

# RZ/A1H Group

User's Manual: Hardware

Renesas Microprocessor  
RZ Family / RZ/A Series

All information contained in these materials, including products and product specifications, represents information on the product at the time of publication and is subject to change by Renesas Electronics Corp. without notice. Please review the latest information published by Renesas Electronics Corp. through various means, including the Renesas Electronics Corp. website (<http://www.renesas.com>).

## Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
3. Renesas Electronics does not assume any liability for infringement of patents, copyrights, or other intellectual property rights of third parties by or arising from the use of Renesas Electronics products or technical information described in this document. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from such alteration, modification, copy or otherwise misappropriation of Renesas Electronics product.
5. Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The recommended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below.  
"Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots etc.  
"High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; and safety equipment etc.  
  
Renesas Electronics products are neither intended nor authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems, surgical implantations etc.), or may cause serious property damages (nuclear reactor control systems, military equipment etc.). You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application for which it is not intended. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for which the product is not intended by Renesas Electronics.
6. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
7. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or systems manufactured by you.
8. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
9. Renesas Electronics products and technology may not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You should not use Renesas Electronics products or technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. When exporting the Renesas Electronics products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations.
10. It is the responsibility of the buyer or distributor of Renesas Electronics products, who distributes, disposes of, or otherwise places the product with a third party, to notify such third party in advance of the contents and conditions set forth in this document, Renesas Electronics assumes no responsibility for any losses incurred by you or third parties as a result of unauthorized use of Renesas Electronics products.
11. This document may not be reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

## General Precautions in the Handling of MPU/MCU Products

The following usage notes are applicable to all MPU/MCU products from Renesas. For detailed usage notes on the products covered by this document, refer to the relevant sections of the document as well as any technical updates that have been issued for the products.

### 1. Handling of Unused Pins

Handle unused pins in accordance with the directions given under Handling of Unused Pins in the manual.

- The input pins of CMOS products are generally in the high-impedance state. In operation with an unused pin in the open-circuit state, extra electromagnetic noise is induced in the vicinity of LSI, an associated shoot-through current flows internally, and malfunctions occur due to the false recognition of the pin state as an input signal become possible. Unused pins should be handled as described under Handling of Unused Pins in the manual.

### 2. Processing at Power-on

The state of the product is undefined at the moment when power is supplied.

- The states of internal circuits in the LSI are indeterminate and the states of register settings and pins are undefined at the moment when power is supplied.

In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the moment when power is supplied until the reset process is completed. In a similar way, the states of pins in a product that is reset by an on-chip power-on reset function are not guaranteed from the moment when power is supplied until the power reaches the level at which resetting has been specified.

### 3. Prohibition of Access to Reserved Addresses

Access to reserved addresses is prohibited.

- The reserved addresses are provided for the possible future expansion of functions. Do not access these addresses; the correct operation of LSI is not guaranteed if they are accessed.

### 4. Clock Signals

After applying a reset, only release the reset line after the operating clock signal has become stable. When switching the clock signal during program execution, wait until the target clock signal has stabilized.

- When the clock signal is generated with an external resonator (or from an external oscillator) during a reset, ensure that the reset line is only released after full stabilization of the clock signal. Moreover, when switching to a clock signal produced with an external resonator (or by an external oscillator) while program execution is in progress, wait until the target clock signal is stable.

### 5. Differences between Products

Before changing from one product to another, i.e. to a product with a different part number, confirm that the change will not lead to problems.

- The characteristics of an MPU or MCU in the same group but having a different part number may differ in terms of the internal memory capacity, layout pattern, and other factors, which can affect the ranges of electrical characteristics, such as characteristic values, operating margins, immunity to noise, and amount of radiated noise. When changing to a product with a different part number, implement a system-evaluation test for the given product.

# Contents

1.	Overview .....	1-1
1.1	Features of This LSI .....	1-1
1.2	Product Lineup .....	1-10
1.3	Block Diagram .....	1-10
1.4	Pin Assignment .....	1-11
1.5	Pin Functions .....	1-14
1.6	List of Pins .....	1-22
2.	CPU .....	2-1
2.1	Features .....	2-1
2.2	Configuration Signals .....	2-2
3.	Boot Mode .....	3-1
3.1	Features .....	3-1
3.2	Boot Mode and Pin Function Setting .....	3-1
3.3	Hardware Used in Each Boot Mode .....	3-2
3.4	Exception Vector Address at a Reset in Each Boot Mode .....	3-3
3.5	Operation .....	3-4
3.5.1	Boot Modes 0 and 1 .....	3-4
3.5.2	Boot Mode 3 .....	3-4
3.5.3	Boot Mode 4 .....	3-5
3.5.4	Boot Mode 5 .....	3-7
3.6	Notes .....	3-9
3.6.1	Boot Related Pins .....	3-9
3.6.2	Operation when an Exception Occurs with the Exception Vector Set to the High Vector Address .....	3-9
4.	Secondary Cache .....	4-1
4.1	Features .....	4-1
4.2	Configuration Signals .....	4-1
5.	LSI Internal Bus .....	5-1
5.1	LSI Internal Bus .....	5-1
5.1.1	Configuration .....	5-1
5.1.2	Operation .....	5-1
5.2	North Main Bus .....	5-2
5.2.1	Configuration .....	5-2
5.2.2	Features .....	5-2
5.2.3	Peripheral Buses .....	5-3
5.3	South Main Bus .....	5-5
5.3.1	Configuration .....	5-5
5.3.2	Features .....	5-5
5.3.3	Connected Buses .....	5-6
5.4	Address Map .....	5-7

5.5	Address Remapping.....	5-10
5.5.1	Overview .....	5-10
5.5.2	Operation .....	5-10
5.6	AXI Interconnect.....	5-11
5.6.1	Configuration.....	5-11
5.6.2	Operation .....	5-11
5.7	Bus Bridges.....	5-11
5.8	AXI Protocol Control Signals.....	5-12
5.9	Write Buffers .....	5-12
5.10	Register Descriptions.....	5-13
5.10.1	Remap Register (RMPR).....	5-14
5.10.2	AXI Bus Control Register 0 (AXIBUSCTL0) .....	5-15
5.10.3	AXI Bus Control Register 1 (AXIBUSCTL1) .....	5-16
5.10.4	AXI Bus Control Register 2 (AXIBUSCTL2) .....	5-17
5.10.5	AXI Bus Control Register 3 (AXIBUSCTL3) .....	5-18
5.10.6	AXI Bus Control Register 4 (AXIBUSCTL4) .....	5-19
5.10.7	AXI Bus Control Register 5 (AXIBUSCTL5) .....	5-20
5.10.8	AXI Bus Control Register 6 (AXIBUSCTL6) .....	5-21
5.10.9	AXI Bus Control Register 7 (AXIBUSCTL7) .....	5-22
5.10.10	AXI Bus Control Register 8 (AXIBUSCTL8) .....	5-23
5.10.11	AXI Bus Control Register 9 (AXIBUSCTL9) .....	5-24
5.10.12	AXI Bus Control Register 10 (AXIBUSCTL10) .....	5-25
5.10.13	AXI Bus Response Error Interrupt Control Register 0 (AXIRERRCTL0).....	5-26
5.10.14	AXI Bus Response Error Interrupt Control Register 1 (AXIRERRCTL1).....	5-27
5.10.15	AXI Bus Response Error Interrupt Control Register 2 (AXIRERRCTL2).....	5-28
5.10.16	AXI Bus Response Error Interrupt Control Register 3 (AXIRERRCTL3).....	5-29
5.10.17	AXI Bus Response Error Status Register 0 (AXIRERRST0).....	5-30
5.10.18	AXI Bus Response Error Status Register 1 (AXIRERRST1).....	5-32
5.10.19	AXI Bus Response Error Status Register 2 (AXIRERRST2).....	5-34
5.10.20	AXI Bus Response Error Status Register 3 (AXIRERRST3).....	5-35
5.10.21	AXI Bus Response Error Clear Register 0 (AXIRERRCLR0).....	5-37
5.10.22	AXI Bus Response Error Clear Register 1 (AXIRERRCLR1).....	5-38
5.10.23	AXI Bus Response Error Clear Register 2 (AXIRERRCLR2).....	5-39
5.10.24	AXI Bus Response Error Clear Register 3 (AXIRERRCLR3).....	5-40
5.11	Interrupt Request .....	5-41
6.	Clock Pulse Generator.....	6-1
6.1	Features.....	6-1
6.2	Input/Output Pins.....	6-4
6.3	Clock Mode .....	6-5
6.4	Register Descriptions.....	6-7

6.4.1	Frequency Control Register (FRQCR) .....	6-7
6.4.2	Frequency Control Register 2 (FRQCR2) .....	6-9
6.5	Changing the Frequency .....	6-10
6.5.1	Changing the Division Ratio .....	6-10
6.6	Usage of the Clock Pins .....	6-11
6.6.1	In the Case of Inputting an External Clock .....	6-11
6.6.2	In the Case of Using a Crystal Resonator .....	6-12
6.6.3	In the Case of Not Using the Clock Pin .....	6-12
6.7	Oscillation Stabilizing Time .....	6-13
6.7.1	Oscillation Stabilizing Time of the On-chip Crystal Oscillator .....	6-13
6.7.2	Oscillation Stabilizing Time of the PLL circuit .....	6-13
6.8	Notes on Board Design .....	6-14
6.8.1	Note on Using a PLL Oscillation Circuit .....	6-14
6.9	Definition of Modulation Rate and Frequency in the SSCG Specification .....	6-15
6.10	Clock Signals .....	6-16
6.10.1	Clock Signals for the System and Realtime Clock .....	6-16
6.10.2	Audio and USB Clock Signals .....	6-17
6.10.3	Video Image Clock Signals (Channel 0) .....	6-18
6.10.4	Video Image Clock Signals (Channel 1) .....	6-19
6.10.5	Other Clock Signals .....	6-19
6.10.6	Internal Clock Signals (1) .....	6-20
6.10.7	Internal Clock Signals (2) .....	6-21
7.	Interrupt Controller .....	7-1
7.1	Features .....	7-1
7.2	Input/Output Pins .....	7-2
7.3	Register Descriptions .....	7-3
7.3.1	Interrupt Control Register 0 (ICR0) .....	7-14
7.3.2	Interrupt Control Register 1 (ICR1) .....	7-15
7.3.3	IRQ Interrupt Request Register (IRQRR) .....	7-16
7.4	Interrupt Sources .....	7-17
7.4.1	NMI Interrupt .....	7-17
7.4.2	IRQ Interrupts .....	7-17
7.4.3	On-Chip Peripheral Module Interrupts .....	7-18
7.4.4	Pin Interrupts .....	7-19
7.5	Interrupt IDs .....	7-19
7.6	Operation .....	7-37
7.6.1	Initial Settings .....	7-37
7.6.2	Flow of Interrupt Operations .....	7-39
7.7	Data Transfer with Interrupt Request Signals .....	7-40

7.7.1	Handling Interrupt Request Signals as Sources for CPU Interrupt but Not Direct Memory Access Controller Activating.....	7-40
7.7.2	Handling Interrupt Request Signals as Sources for Activating Direct Memory Access Controller but Not CPU Interrupt.....	7-40
7.8	Usage Note .....	7-41
7.8.1	Timing to Clear an Interrupt Source.....	7-41
7.8.2	Notes on Selecting IRQ Interrupt Pin Functions.....	7-41
7.8.3	Notes on Reading Interrupt ID Values from ICCIAR.....	7-41
<b>8.</b>	<b>Bus State Controller .....</b>	<b>8-1</b>
8.1	Features.....	8-1
8.2	Input/Output Pins.....	8-3
8.3	Area Overview.....	8-4
8.3.1	Address Map.....	8-4
8.3.2	Data Bus Width and Related Pin Setting for Each Area Depending on Boot Mode.....	8-5
8.4	Register Descriptions.....	8-6
8.4.1	Common Control Register (CMNCR).....	8-7
8.4.2	CSn Space Bus Control Register (CSnBCR) (n = 0 to 5) .....	8-8
8.4.3	CSn Space Wait Control Register (CSnWCR) (n = 0 to 5).....	8-10
8.4.4	SDRAM Control Register (SDCR) .....	8-27
8.4.5	Refresh Timer Control/Status Register (RTCSR) .....	8-29
8.4.6	Refresh Timer Counter (RTCNT) .....	8-30
8.4.7	Refresh Time Constant Register (RTCOR).....	8-30
8.4.8	Timeout Cycle Constant Register (TOSCORn) (n = 0 to 5) .....	8-31
8.4.9	Timeout Status Register (TOSTR) .....	8-32
8.4.10	Timeout Enable Register (TOENR) .....	8-34
8.5	Operation .....	8-35
8.5.1	Access Size and Data Alignment.....	8-35
8.5.2	Normal Space Interface .....	8-37
8.5.3	Access Wait Control.....	8-42
8.5.4	$\overline{\text{CSn}}$ Assert Period Expansion.....	8-44
8.5.5	MPX-I/O Interface.....	8-45
8.5.6	SDRAM Interface.....	8-48
8.5.7	Burst ROM (Clocked Asynchronous) Interface .....	8-77
8.5.8	SRAM Interface with Byte Selection .....	8-78
8.5.9	Burst ROM (Clocked Synchronous) Interface .....	8-83
8.5.10	Wait between Access Cycles.....	8-84
8.5.11	Others.....	8-87
<b>9.</b>	<b>Direct Memory Access Controller .....</b>	<b>9-1</b>
9.1	Features.....	9-1
9.2	Input/Output Pins.....	9-2
9.3	Register Configuration .....	9-2

9.4	Register Descriptions.....	9-4
9.4.1	Next Source Address Register n (N0SA_n, N1SA_n) .....	9-12
9.4.2	Next Destination Address Register n (N0DA_n, N1DA_n).....	9-12
9.4.3	Next Transaction Byte Register n (N0TB_n, N1TB_n).....	9-13
9.4.4	Current Source Address Register (CRSA_n) .....	9-13
9.4.5	Current Destination Address Register (CRDA_n) .....	9-14
9.4.6	Current Transaction Byte Register (CRTB_n) .....	9-14
9.4.7	Channel Status Register n (CHSTAT_n).....	9-15
9.4.8	Channel Control Register n (CHCTRL_n).....	9-18
9.4.9	Channel Configuration Register n (CHCFG_n).....	9-20
9.4.10	Channel Interval Register n (CHITVL_n).....	9-22
9.4.11	Channel Extension Register n (CHEXT_n).....	9-23
9.4.12	Next Link Address Register n (NXLA_n).....	9-23
9.4.13	Current Link Address Register n (CRLA_n).....	9-24
9.4.14	DMA Control Register (DCTRL_0_7, DCTRL_8_15) .....	9-24
9.4.15	DMA Status EN Register (DSTAT_EN_0_7).....	9-25
9.4.16	DMA Status EN Register (DSTAT_EN_8_15).....	9-25
9.4.17	DMA Status ER Register (DSTAT_ER_0_7).....	9-26
9.4.18	DMA Status ER Register (DSTAT_ER_8_15).....	9-26
9.4.19	DMA Status END Register (DSTAT_END_0_7).....	9-27
9.4.20	DMA Status END Register (DSTAT_END_8_15).....	9-27
9.4.21	DMA Status TC Register (DSTAT_TC_0_7).....	9-28
9.4.22	DMA Status TC Register (DSTAT_TC_8_15).....	9-28
9.4.23	DMA Status SUS Register (DSTAT_SUS_0_7).....	9-29
9.4.24	DMA Status SUS Register (DSTAT_SUS_8_15).....	9-29
9.4.25	DMA Extension Resource Selectors 0 to 7 (DMARS0 to DMARS7) .....	9-30
9.5	Operation .....	9-33
9.5.1	Transfer Flow .....	9-33
9.5.2	DMA Transfer Requests.....	9-33
9.6	DMA Mode.....	9-40
9.6.1	Mode Setting.....	9-40
9.6.2	Register Mode.....	9-40
9.6.3	Link Mode .....	9-46
9.7	DMA Transfer .....	9-53
9.7.1	Transfer Mode .....	9-53
9.7.2	Priority Control for DMA Channels.....	9-54
9.7.3	Number of States of an External Bus Cycle .....	9-56
9.7.4	DMA Transfer Request .....	9-56
9.7.5	DMA Acknowledge Output Function .....	9-58
9.7.6	DMA Transfer End Output Function.....	9-60



9.7.7	DMA Transfer End Interrupt.....	9-60
9.7.8	DMA Error Interrupt .....	9-61
9.7.9	Interval Count Function.....	9-61
9.7.10	Difference in Operation Due to the Transfer Size.....	9-62
9.7.11	Transfer Status.....	9-63
9.8	DMA Setting Examples.....	9-67
9.8.1	Setting Example 1 (Register Mode/Hardware Request).....	9-67
9.8.2	Setting Example 2 (Register Mode/Software Request).....	9-69
9.8.3	Setting Example 3 (Register Mode/Continuous Execution).....	9-71
9.8.4	Setting Example 4 (Link Mode).....	9-72
9.8.5	Next Register Set Continuous Execution Setting.....	9-75
9.9	Note .....	9-77
9.9.1	Divided Output of DACK0 and TEND0 .....	9-77
9.9.2	TEND0 Not Output .....	9-78
10.	Multi-Function Timer Pulse Unit 2.....	10-1
10.1	Features.....	10-1
10.2	Input/Output Pins.....	10-5
10.3	Register Descriptions.....	10-6
10.3.1	Timer Control Register (TCR) .....	10-8
10.3.2	Timer Mode Register (TMDR).....	10-11
10.3.3	Timer I/O Control Register (TIOR).....	10-13
10.3.4	Timer Interrupt Enable Register (TIER) .....	10-31
10.3.5	Timer Status Register (TSR) .....	10-33
10.3.6	Timer Buffer Operation Transfer Mode Register (TBTM).....	10-36
10.3.7	Timer Input Capture Control Register (TICCR) .....	10-37
10.3.8	Timer A/D Converter Start Request Control Register (TADCR) .....	10-38
10.3.9	Timer A/D Converter Start Request Cycle Set Registers (TADCORA_4 and TADCORB_4) .....	10-40
10.3.10	Timer A/D Converter Start Request Cycle Set Buffer Registers (TADCOBRA_4 and TADCOBRB_4).....	10-40
10.3.11	Timer Counter (TCNT) .....	10-40
10.3.12	Timer General Register (TGR).....	10-41
10.3.13	Timer Start Register (TSTR).....	10-42
10.3.14	Timer Synchronous Register (TSYR) .....	10-43
10.3.15	Timer Read/Write Enable Register (TRWER).....	10-44
10.3.16	Timer Output Master Enable Register (TOER).....	10-45
10.3.17	Timer Output Control Register 1 (TOCR1) .....	10-46
10.3.18	Timer Output Control Register 2 (TOCR2) .....	10-48
10.3.19	Timer Output Level Buffer Register (TOLBR).....	10-51
10.3.20	Timer Gate Control Register (TGCR).....	10-52
10.3.21	Timer Subcounter (TCNTS).....	10-53

10.3.22	Timer Dead Time Data Register (TDDR)	10-53
10.3.23	Timer Cycle Data Register (TCDR)	10-54
10.3.24	Timer Cycle Buffer Register (TCBR)	10-54
10.3.25	Timer Interrupt Skipping Set Register (TITCR)	10-55
10.3.26	Timer Interrupt Skipping Counter (TITCNT)	10-57
10.3.27	Timer Buffer Transfer Set Register (TBTER)	10-58
10.3.28	Timer Dead Time Enable Register (TDER)	10-59
10.3.29	Timer Waveform Control Register (TWCR)	10-60
10.3.30	Bus Master Interface	10-60
10.4	Operation	10-61
10.4.1	Basic Functions	10-61
10.4.2	Synchronous Operation	10-67
10.4.3	Buffer Operation	10-69
10.4.4	Cascaded Operation	10-73
10.4.5	PWM Modes	10-78
10.4.6	Phase Counting Mode	10-82
10.4.7	Reset-Synchronized PWM Mode	10-88
10.4.8	Complementary PWM Mode	10-91
10.4.9	A/D Converter Start Request Delaying Function	10-126
10.4.10	TCNT Capture at Crest and/or Trough in Complementary PWM Operation	10-129
10.5	Interrupt Sources	10-130
10.5.1	Interrupt Sources and Priorities	10-130
10.5.2	Activation of Direct Memory Access Controller	10-131
10.5.3	A/D Converter Activation	10-131
10.6	Operation Timing	10-133
10.6.1	Input/Output Timing	10-133
10.6.2	Interrupt Signal Timing	10-138
10.7	Usage Notes	10-141
10.7.1	Module Standby Mode Setting	10-141
10.7.2	Input Clock Restrictions	10-141
10.7.3	Caution on Period Setting	10-142
10.7.4	Contention between TCNT Write and Clear Operations	10-142
10.7.5	Contention between TCNT Write and Increment Operations	10-143
10.7.6	Contention between TGR Write and Compare Match	10-143
10.7.7	Contention between Buffer Register Write and Compare Match	10-144
10.7.8	Contention between Buffer Register Write and TCNT Clear	10-145
10.7.9	Contention between TGR Read and Input Capture	10-145
10.7.10	Contention between TGR Write and Input Capture	10-146
10.7.11	Contention between Buffer Register Write and Input Capture	10-146
10.7.12	TCNT2 Write and Overflow/Underflow Contention in Cascade Connection	10-147

10.7.13	Counter Value during Complementary PWM Mode Stop .....	10-148
10.7.14	Buffer Operation Setting in Complementary PWM Mode.....	10-148
10.7.15	Reset Sync PWM Mode Buffer Operation and Compare Match Flag .....	10-149
10.7.16	Overflow Flags in Reset Synchronous PWM Mode .....	10-150
10.7.17	Contention between Overflow/Underflow and Counter Clearing.....	10-150
10.7.18	Contention between TCNT Write and Overflow/Underflow .....	10-151
10.7.19	Cautions on Transition from Normal Operation or PWM Mode 1 to Reset-Synchronized PWM Mode .....	10-151
10.7.20	Output Level in Complementary PWM Mode and Reset-Synchronized PWM Mode .....	10-151
10.7.21	Interrupts in Module Standby Mode.....	10-151
10.7.22	Simultaneous Capture of TCNT_1 and TCNT_2 in Cascade Connection.....	10-152
10.7.23	Notes on Output Waveform Control During Synchronous Counter Clearing in Complementary PWM Mode.....	10-152
10.8	Output Pin Initialization for Multi-Function Timer Pulse Unit 2.....	10-154
10.8.1	Operating Modes .....	10-154
10.8.2	Reset Start Operation.....	10-154
10.8.3	Operation in Case of Re-Setting Due to Error during Operation, etc.....	10-155
10.8.4	Overview of Initialization Procedures and Mode Transitions in Case of Error during Operation, etc.....	10-156
11.	OS Timer.....	11-1
11.1.	Functional Overview .....	11-1
11.1.1.	Features of OSTM .....	11-1
11.2.	Registers .....	11-2
11.2.1.	Registers Overview.....	11-2
11.2.2.	Details of OSTM Registers.....	11-3
11.3.	Functional Description .....	11-7
11.3.1.	Count Clock.....	11-7
11.3.2.	Generation of Interrupt Request.....	11-7
11.3.3.	Starting and Stopping the Timer.....	11-7
11.3.4.	Interval Timer Mode.....	11-8
11.3.5.	Free-Running Comparison Mode .....	11-8
12.	Watchdog Timer.....	12-1
12.1	Features.....	12-1
12.2	Input/Output Pin .....	12-2
12.3	Register Descriptions.....	12-2
12.3.1	Watchdog Timer Counter (WTCNT) .....	12-3
12.3.2	Watchdog Timer Control/Status Register (WTC SR) .....	12-3
12.3.3	Watchdog Reset Control/Status Register (WRCSR).....	12-5
12.3.4	Notes on Register Access .....	12-6
12.4	Usage .....	12-8
12.4.1	Canceling Software Standby Mode.....	12-8

12.4.2	Using Watchdog Timer Mode .....	12-8
12.4.3	Using Interval Timer Mode .....	12-9
12.5	Usage Notes .....	12-10
12.5.1	Timer Variation .....	12-10
12.5.2	Prohibition against Setting H'FF to WTCNT .....	12-10
12.5.3	Interval Timer Overflow Flag.....	12-10
12.5.4	System Reset by $\overline{\text{WDTOVF}}$ Signal .....	12-10
12.5.5	Internal Reset in Watchdog Timer Mode .....	12-10
13.	<b>Realtime Clock</b> .....	<b>13-1</b>
13.1	Features.....	13-1
13.2	Input/Output Pin .....	13-3
13.3	Register Descriptions.....	13-3
13.3.1	64-Hz Counter (R64CNT).....	13-4
13.3.2	Second Counter (RSECCNT).....	13-5
13.3.3	Minute Counter (RMINCNT).....	13-5
13.3.4	Hour Counter (RHRCNT) .....	13-6
13.3.5	Day of Week Counter (RWKCNT).....	13-6
13.3.6	Day Counter (RDAYCNT).....	13-7
13.3.7	Month Counter (RMONCNT).....	13-7
13.3.8	Year Counter (RYRCNT).....	13-8
13.3.9	Second Alarm Register (RSECAR).....	13-8
13.3.10	Minute Alarm Register (RMINAR) .....	13-9
13.3.11	Hour Alarm Register (RHRAR).....	13-9
13.3.12	Day of Week Alarm Register (RWKAR).....	13-10
13.3.13	Day Alarm Register (RDAYAR) .....	13-10
13.3.14	Month Alarm Register (RMONAR).....	13-11
13.3.15	Year Alarm Register (RYRAR) .....	13-11
13.3.16	Control Register 1 (RCR1).....	13-12
13.3.17	Control Register 2 (RCR2).....	13-13
13.3.18	Control Register 3 (RCR3).....	13-14
13.3.19	Control Register 5 (RCR5).....	13-14
13.3.20	Frequency Register H/L (RFRH/L).....	13-15
13.4	Operation .....	13-16
13.4.1	Initial Settings of Registers after Power-On and Oscillation Stabilization Time.....	13-16
13.4.2	Setting Time .....	13-16
13.4.3	Reading Time .....	13-17
13.4.4	Alarm Function.....	13-18
13.5	Usage Notes .....	13-19
13.5.1	Register Writing during Count Operation .....	13-19
13.5.2	Use of Realtime Clock Periodic Interrupts.....	13-19

13.5.3	Transition to Standby Mode after Setting Register .....	13-19
13.5.4	Usage Notes when Writing to and Reading the Register .....	13-19
<b>14.</b>	<b>Serial Communication Interface with FIFO .....</b>	<b>14-1</b>
14.1	Features.....	14-1
14.2	Input/Output Pins.....	14-3
14.3	Register Descriptions.....	14-4
14.3.1	Receive Shift Register (SCRSR).....	14-6
14.3.2	Receive FIFO Data Register (SCFRDR).....	14-6
14.3.3	Transmit Shift Register (SCTSR).....	14-7
14.3.4	Transmit FIFO Data Register (SCFTDR) .....	14-7
14.3.5	Serial Mode Register (SCSMR).....	14-8
14.3.6	Serial Control Register (SCSCR).....	14-10
14.3.7	Serial Status Register (SCFSR).....	14-12
14.3.8	Bit Rate Register (SCBRR).....	14-16
14.3.9	FIFO Control Register (SCFCR).....	14-20
14.3.10	FIFO Data Count Set Register (SCFDR) .....	14-22
14.3.11	Serial Port Register (SCSPTR).....	14-23
14.3.12	Line Status Register (SCLSR).....	14-25
14.3.13	Serial Extension Mode Register (SCEMR).....	14-26
14.4	Operation .....	14-27
14.4.1	Overview .....	14-27
14.4.2	Operation in Asynchronous Mode.....	14-29
14.4.3	Operation in Clock Synchronous Mode .....	14-40
14.5	Interrupts.....	14-48
14.6	Usage Notes .....	14-49
14.6.1	SCFTDR Writing and TDFE Flag.....	14-49
14.6.2	SCFRDR Reading and RDF Flag.....	14-49
14.6.3	Restriction on Direct Memory Controller Usage .....	14-49
14.6.4	Break Detection and Processing.....	14-49
14.6.5	Sending a Break Signal.....	14-50
14.6.6	Receive Data Sampling Timing and Receive Margin (Asynchronous Mode).....	14-50
14.6.7	Selection of Base Clock in Asynchronous Mode .....	14-51
<b>15.</b>	<b>Serial Communications Interface .....</b>	<b>15-1</b>
15.1	Overview .....	15-1
15.2	Register Descriptions.....	15-3
15.2.1	Receive Shift Register (RSR).....	15-4
15.2.2	Receive Data Register (RDR).....	15-4
15.2.3	Transmit Data Register (TDR) .....	15-4
15.2.4	Transmit Shift Register (TSR).....	15-4
15.2.5	Serial Mode Register (SMR).....	15-5

15.2.6	Serial Control Register (SCR) .....	15-8
15.2.7	Serial Status Register (SSR) .....	15-12
15.2.8	Smart Card Mode Register (SCMR) .....	15-16
15.2.9	Bit Rate Register (BRR) .....	15-17
15.2.10	Serial Extended Mode Register (SEMR) .....	15-21
15.2.11	Noise Filter Setting Register (SNFR) .....	15-21
15.2.12	Extended Function Control Register (SECR) .....	15-22
15.3	Operation in Asynchronous Mode .....	15-23
15.3.1	Serial Data Transfer Format .....	15-23
15.3.2	Receive Data Sampling Timing and Reception Margin in Asynchronous Mode .....	15-25
15.3.3	Clock .....	15-26
15.3.4	CTS and RTS Functions .....	15-26
15.3.5	SCI Initialization (Asynchronous Mode) .....	15-27
15.3.6	Serial Data Transmission (Asynchronous Mode) .....	15-28
15.3.7	Serial Data Reception (Asynchronous Mode) .....	15-30
15.4	Multi-Processor Communications Function .....	15-34
15.4.1	Multi-Processor Serial Data Transmission .....	15-35
15.4.2	Multi-Processor Serial Data Reception .....	15-36
15.5	Operation in Clock Synchronous Mode .....	15-39
15.5.1	Clock .....	15-39
15.5.2	CTS and RTS Functions .....	15-39
15.5.3	Initialization (Clock Synchronous Mode) .....	15-40
15.5.4	Serial Data Transmission (Clock Synchronous Mode) .....	15-41
15.5.5	Serial Data Reception (Clock Synchronous Mode) .....	15-43
15.5.6	Simultaneous Serial Data Transmission and Reception (Clock Synchronous Mode) .....	15-46
15.6	Operation in Smart Card Interface Mode .....	15-47
15.6.1	Sample Connection .....	15-47
15.6.2	Data Format (Except in Block Transfer Mode) .....	15-48
15.6.3	Block Transfer Mode .....	15-50
15.6.4	Receive Data Sampling Timing and Reception Margin .....	15-50
15.6.5	Initialization (Smart Card Interface Mode) .....	15-51
15.6.6	Serial Data Transmission (Except in Block Transfer Mode) .....	15-52
15.6.7	Serial Data Reception (Except in Block Transfer Mode) .....	15-55
15.6.8	Clock Output Control .....	15-56
15.7	Noise Cancellation Function .....	15-58
15.8	Interrupt Sources .....	15-59
15.8.1	Interrupts in Serial Communications Interface Mode .....	15-59
15.8.2	Interrupts in Smart Card Interface Mode .....	15-60
15.9	Usage Notes .....	15-61
15.9.1	Setting the Module Standby Function .....	15-61

15.9.2	Break Detection and Processing.....	15-61
15.9.3	The Mark State and Production of Breaks.....	15-61
15.9.4	Receive Error Flags and Transmit Operations (Clock Synchronous Mode Only).....	15-61
15.9.5	Writing Data to TDR.....	15-61
15.9.6	Restrictions on Clock Synchronous Transmission.....	15-61
15.9.7	Restrictions on Using DMAC.....	15-61
15.9.8	Points to Note on Starting Transfer.....	15-61
15.9.9	SCI Operations during Low Power Consumption State.....	15-62
15.9.10	External Clock Input in Clock Synchronous Mode.....	15-64
15.10	IrDA Communications.....	15-65
15.11	IrDA Register Description.....	15-66
15.11.1	IrDA Control Register (IRCR).....	15-66
15.12	IrDA Operation.....	15-67
15.12.1	Flow of IrDA Setting.....	15-67
15.12.2	Transmission.....	15-67
15.12.3	Reception.....	15-68
15.12.4	Selection of High-Level Pulse Width.....	15-68
15.13	Notes on Using the IrDA Module.....	15-69
15.13.1	Shortest Pulse Width in Reception.....	15-69
15.13.2	Asynchronous Basic Clock for Serial Communication Interface.....	15-69
16.	<b>Renesas Serial Peripheral Interface.....</b>	<b>16-1</b>
16.1	Features.....	16-1
16.2	Input/Output Pins.....	16-3
16.3	Register Descriptions.....	16-4
16.3.1	Control Register (SPCR).....	16-7
16.3.2	Slave Select Polarity Register (SSLP).....	16-8
16.3.3	Pin Control Register (SPPCR).....	16-8
16.3.4	Status Register (SPSR).....	16-9
16.3.5	Data Register (SPDR).....	16-11
16.3.6	Sequence Control Register (SPSCR).....	16-11
16.3.7	Sequence Status Register (SPSSR).....	16-12
16.3.8	Bit Rate Register (SPBR).....	16-12
16.3.9	Data Control Register (SPDCR).....	16-13
16.3.10	Clock Delay Register (SPCKD).....	16-14
16.3.11	Slave Select Negation Delay Register (SSLND).....	16-15
16.3.12	Next-Access Delay Register (SPND).....	16-16
16.3.13	Command Register (SPCMD).....	16-17
16.3.14	Buffer Control Register (SPBFCR).....	16-19
16.3.15	Buffer Data Count Setting Register (SPBFDR).....	16-20
16.4	Operation.....	16-21

16.4.1	Overview of Operations .....	16-21
16.4.2	Pin Control.....	16-22
16.4.3	System Configuration Example.....	16-22
16.4.4	Transfer Format .....	16-25
16.4.5	Data Format .....	16-27
16.4.6	Error Detection .....	16-33
16.4.7	Initialization.....	16-35
16.4.8	SPI Operation .....	16-36
16.4.9	Error Handling.....	16-46
16.4.10	Loopback Mode.....	16-47
16.4.11	Interrupt Sources.....	16-47
17.	SPI Multi I/O Bus Controller .....	17-1
17.1	Features.....	17-1
17.2	Block Diagram.....	17-2
17.3	Input/Output Pins.....	17-3
17.4	Register Descriptions.....	17-4
17.4.1	Common Control Register (CMNCR).....	17-6
17.4.2	SSL Delay Register (SSLDR) .....	17-8
17.4.3	Bit Rate Register (SPBCR).....	17-9
17.4.4	Data Read Control Register (DRCR) .....	17-11
17.4.5	Data Read Command Setting Register (DRCMR).....	17-12
17.4.6	Data Read Extended Address Setting Register (DREAR) .....	17-13
17.4.7	Data Read Option Setting Register (DROPR).....	17-14
17.4.8	Data Read Enable Setting Register (DRENDR) .....	17-15
17.4.9	SPI Mode Control Register (SMCR).....	17-17
17.4.10	SPI Mode Command Setting Register (SMCMR) .....	17-18
17.4.11	SPI Mode Address Setting Register (SMADR) .....	17-18
17.4.12	SPI Mode Option Setting Register (SMOPR) .....	17-19
17.4.13	SPI Mode Enable Setting Register (SMENR).....	17-20
17.4.14	SPI Mode Read Data Register 0 (SMRDR0) .....	17-22
17.4.15	SPI Mode Read Data Register 1 (SMRDR1) .....	17-22
17.4.16	SPI Mode Write Data Register 0 (SMWDR0) .....	17-23
17.4.17	SPI Mode Write Data Register 1 (SMWDR1) .....	17-23
17.4.18	Common Status Register (CMNSR).....	17-24
17.4.19	SPI AC Input Characteristics Adjustment Register (CKDLY) .....	17-25
17.4.20	Data Read Dummy Cycle Setting Register (DRDMCR) .....	17-26
17.4.21	Data Read DDR Enable Register (DRDRENDR).....	17-27
17.4.22	SPI Mode Dummy Cycle Setting Register (SMDMCR).....	17-28
17.4.23	SPI Mode DDR Enable Register (SMDRENDR) .....	17-29
17.4.24	SPI AC Output Characteristics Adjustment Register (SPODLY).....	17-30



17.5	Operation .....	17-31
17.5.1	System Configuration .....	17-31
17.5.2	Address Map .....	17-32
17.5.3	32-bit Serial Flash Addresses .....	17-32
17.5.4	Data Alignment .....	17-33
17.5.5	Operating Modes .....	17-34
17.5.6	External Address Space Read Mode .....	17-34
17.5.7	Read Cache .....	17-38
17.5.8	SPI Operating Mode .....	17-39
17.5.9	Transfer Format .....	17-42
17.5.10	Data Format .....	17-43
17.5.11	Data Pin Control .....	17-48
17.5.12	SPBSSL Pin Control .....	17-50
17.5.13	Flags .....	17-50
17.6	Usage Notes .....	17-51
17.6.1	Notes on Transfer to Read Data in SPI Operating Mode .....	17-51
17.6.2	Notes on Starting Transfer from the SPBSSL Retained State in SPI Operating Mode .....	17-51
18.	I <sup>2</sup> C Bus Interface .....	18-1
18.1.	Features .....	18-1
18.1.1.	Channels .....	18-1
18.1.2.	Register Base Addresses .....	18-1
18.1.3.	I/O signals .....	18-2
18.2.	Overview .....	18-3
18.2.1.	Functional Overview .....	18-3
18.2.2.	Block Diagram .....	18-5
18.3.	Registers .....	18-7
18.3.1.	RIICnCR1 — I <sup>2</sup> C Bus Control Register 1 .....	18-7
18.3.2.	RIICnCR2 — I <sup>2</sup> C Bus Control Register 2 .....	18-10
18.3.3.	RIICnMR1 — I <sup>2</sup> C Bus Mode Register 1 .....	18-14
18.3.4.	RIICnMR2 — I <sup>2</sup> C Bus Mode Register 2 .....	18-16
18.3.5.	RIICnMR3 — I <sup>2</sup> C Bus Mode Register 3 .....	18-18
18.3.6.	RIICnFER — I <sup>2</sup> C Bus Function Enable Register .....	18-21
18.3.7.	RIICnSER — I <sup>2</sup> C Bus Status Enable Register .....	18-23
18.3.8.	RIICnIER — I <sup>2</sup> C Bus Interrupt Enable Register .....	18-25
18.3.9.	RIICnSR1 — I <sup>2</sup> C Bus Status Register 1 .....	18-27
18.3.10.	RIICnSR2 — I <sup>2</sup> C Bus Status Register 2 .....	18-30
18.3.11.	RIICnSAR <sub>y</sub> — I <sup>2</sup> C Slave Address Register y (y = 0 to 2) .....	18-35
18.3.12.	RIICnBRL — I <sup>2</sup> C Bus Bit Rate Low-Level Register .....	18-37
18.3.13.	RIICnBRH — I <sup>2</sup> C Bus Bit Rate High-Level Register .....	18-38
18.3.14.	RIICnDRT — I <sup>2</sup> C Bus Transmit Data Register .....	18-41

18.3.15.	RIICnDRR — I <sup>2</sup> C Bus Receive Data Register .....	18-42
18.3.16.	RIICnDRS — I <sup>2</sup> C Bus Shift Register .....	18-43
18.4.	Interrupt Sources.....	18-44
18.5.	Operation .....	18-45
18.5.1.	Communication Data Format .....	18-45
18.5.2.	Initial Settings.....	18-46
18.5.3.	Master Transmit Operation.....	18-47
18.5.4.	Master Receive Operation .....	18-51
18.5.5.	Slave Transmit Operation.....	18-57
18.5.6.	Slave Receive Operation .....	18-60
18.6.	SCL Synchronization Circuit.....	18-63
18.7.	Facility for Delaying SDA Output.....	18-64
18.8.	Digital Noise-Filter Circuits .....	18-65
18.9.	Address Match Detection .....	18-66
18.9.1.	Slave-Address Match Detection .....	18-66
18.9.2.	Detection of the General Call Address .....	18-68
18.9.3.	Device-ID Address Detection.....	18-69
18.9.4.	Host Address Detection.....	18-71
18.10.	Automatically Low-Hold Function for SCL .....	18-72
18.10.1.	Function to Prevent Wrong Transmission of Transmit Data.....	18-72
18.10.2.	NACK Reception Transfer Suspension Function.....	18-73
18.10.3.	Function to Prevent Failure to Receive Data.....	18-74
18.11.	Arbitration-Lost Detection Functions.....	18-76
18.11.1.	Master Arbitration-Lost Detection (MALE Bit) .....	18-76
18.11.2.	Function to Detect Loss of Arbitration during NACK Transmission (NALE Bit) .....	18-78
18.11.3.	Slave Arbitration-Lost Detection (SALE Bit).....	18-79
18.12.	Start Condition/Restart Condition/Stop Condition Issuing Function .....	18-80
18.12.1.	Issuing a Start Condition .....	18-80
18.12.2.	Issuing a Restart Condition.....	18-80
18.12.3.	Issuing a Stop Condition.....	18-82
18.13.	Bus Hanging .....	18-83
18.13.1.	Timeout Function .....	18-83
18.13.2.	Extra SCL Clock Cycle Output Function.....	18-85
18.13.3.	RIIC Reset and Internal Reset .....	18-86
18.14.	SMBus Operation .....	18-87
18.14.1.	SMBus Timeout Measurement.....	18-87
18.14.2.	SMBus Host Notification Protocol/Notify ARP Master .....	18-88
18.15.	Reset States.....	18-89
19.	Serial Sound Interface .....	19-1
19.1	Features.....	19-1

19.2	Input/Output Pins.....	19-3
19.3	Register Description .....	19-4
19.3.1	Control Register (SSICR).....	19-6
19.3.2	Status Register (SSISR).....	19-10
19.3.3	Transmit Data Register (SSITDR) .....	19-12
19.3.4	Receive Data Register (SSIRDR).....	19-12
19.3.5	FIFO Control Register (SSIFCR).....	19-13
19.3.6	FIFO Status Register (SSIFSR).....	19-15
19.3.7	Transmit FIFO Data Register (SSIFTDR).....	19-17
19.3.8	Receive FIFO Data Register (SSIFRDR).....	19-17
19.3.9	TDM Mode Register (SSITDMR).....	19-18
19.3.10	FC Control Register (SSIFCCR) .....	19-19
19.3.11	FC Mode Register (SSIFCMR) .....	19-20
19.3.12	FC Status Register (SSIFCSR).....	19-21
19.4	Operation Description.....	19-22
19.4.1	Bus Format .....	19-22
19.4.2	Non-Compressed Modes .....	19-23
19.4.3	TDM Mode.....	19-31
19.4.4	WS Continue Mode .....	19-32
19.4.5	Operation Modes .....	19-32
19.4.6	Transmit Operation.....	19-33
19.4.7	Receive Operation .....	19-36
19.4.8	Serial Bit Clock Control .....	19-38
19.5	Usage Notes .....	19-38
19.5.1	Limitations from Underflow or Overflow during DMA Operation .....	19-38
19.5.2	Note on Changing Mode from Master Transceiver to Master Receiver .....	19-38
19.5.3	Limits on TDM mode and WS Continue Mode .....	19-38
20.	Media Local Bus .....	20-1
20.1	Features.....	20-1
20.2	Input/Output Pins.....	20-2
20.3	Register Description .....	20-2
21.	CAN Interface .....	21-1
21.1.	Overview .....	21-1
21.1.1.	Units.....	21-1
21.1.2.	Register addresses.....	21-2
21.1.3.	Clock supply .....	21-2
21.1.4.	Interrupts.....	21-3
21.1.5.	I/O signals.....	21-4
21.2.	Function.....	21-4
21.2.1.	Block Diagram.....	21-7

21.3.	Registers .....	21-8
21.3.1.	RSCAN0CmCFG — Channel Configuration Register (m = 0 to 4).....	21-34
21.3.2.	RSCAN0CmCTR — Channel Control Register (m = 0 to 4) .....	21-36
21.3.3.	RSCAN0CmSTS — Channel Status Register (m = 0 to 4).....	21-40
21.3.4.	RSCAN0CmERFL — Channel Error Flag Register (m = 0 to 4).....	21-42
21.3.5.	RSCAN0GCFG — Global Configuration Register.....	21-46
21.3.6.	RSCAN0GCTR — Global Control Register.....	21-49
21.3.7.	RSCAN0GSTS — Global Status Register .....	21-51
21.3.8.	RSCAN0GERFL — Global Error Flag Register .....	21-53
21.3.9.	RSCAN0GTINTSTS0 — Global TX Interrupt Status Register 0.....	21-54
21.3.10.	RSCAN0GTINTSTS1 — Global TX Interrupt Status Register 1.....	21-57
21.3.11.	RSCAN0GTSC — Global Timestamp Counter Register.....	21-59
21.3.12.	RSCAN0GAFLECTR — Receive Rule Entry Control Register .....	21-60
21.3.13.	RSCAN0GAFLCFG0 — Receive Rule Configuration Register 0 .....	21-61
21.3.14.	RSCAN0GAFLCFG1 — Receive Rule Configuration Register 1 .....	21-63
21.3.15.	RSCAN0GAFLIDj — Receive Rule ID Register (j = 0 to 15).....	21-64
21.3.16.	RSCAN0GAFLMj — Receive Rule Mask Register (j = 0 to 15).....	21-66
21.3.17.	RSCAN0GAFLP0j — Receive Rule Pointer 0 Register (j = 0 to 15).....	21-67
21.3.18.	RSCAN0GAFLP1j — Receive Rule Pointer 1 Register (j = 0 to 15).....	21-69
21.3.19.	RSCAN0RMNB — Receive Buffer Number Register .....	21-70
21.3.20.	RSCAN0RMNDy — Receive Buffer New Data Register y (y = 0 to 2).....	21-71
21.3.21.	RSCAN0RMIDq — Receive Buffer ID Register (q = 0 to 79) .....	21-72
21.3.22.	RSCAN0RMPTRq — Receive Buffer Pointer Register (q = 0 to 79).....	21-73
21.3.23.	RSCAN0RMDF0q — Receive Buffer Data Field 0 Register (q = 0 to 79).....	21-74
21.3.24.	RSCAN0RMDF1q — Receive Buffer Data Field 1 Register (q = 0 to 79).....	21-75
21.3.25.	RSCAN0RFCCx — Receive FIFO Buffer Configuration and Control Register (x = 0 to 7) .....	21-76
21.3.26.	RSCAN0RFSTSx — Receive FIFO Buffer Status Register (x = 0 to 7).....	21-78
21.3.27.	RSCAN0RFPCTR <sub>x</sub> — Receive FIFO Buffer Pointer Control Register (x = 0 to 7) .....	21-80
21.3.28.	RSCAN0RFID <sub>x</sub> — Receive FIFO Buffer Access ID Register (x = 0 to 7).....	21-81
21.3.29.	RSCAN0RFPTR <sub>x</sub> — Receive FIFO Buffer Access Pointer Register (x = 0 to 7) .....	21-82
21.3.30.	RSCAN0RFD <sub>0x</sub> — Receive FIFO Buffer Access Data Field 0 Register (x = 0 to 7) .....	21-83
21.3.31.	RSCAN0RFD <sub>1x</sub> — Receive FIFO Buffer Access Data Field 1 Register (x = 0 to 7) .....	21-84
21.3.32.	RSCAN0CFCC <sub>k</sub> — Transmit/receive FIFO buffer Configuration and Control Register k (k = 0 to 14).....	21-85
21.3.33.	RSCAN0CFST <sub>sk</sub> — Transmit/receive FIFO buffer Status Register (k = 0 to 14) .....	21-89
21.3.34.	RSCAN0CFPCTR <sub>k</sub> — Transmit/receive FIFO buffer Pointer Control Register (k = 0 to 14) .....	21-92

21.3.35.	RSCAN0CFIDk — Transmit/receive FIFO buffer Access ID Register (k = 0 to 14) .....	21-94
21.3.36.	RSCAN0CFPTRk — Transmit/receive FIFO buffer Access Pointer Register (k = 0 to 14) .....	21-96
21.3.37.	RSCAN0CFDF0k — Transmit/receive FIFO buffer Access Data Field 0 Register (k = 0 to 14) .....	21-98
21.3.38.	RSCAN0CFDF1k — Transmit/receive FIFO buffer Access Data Field 1 Register (k = 0 to 14) .....	21-99
21.3.39.	RSCAN0FESTS — FIFO Empty Status Register .....	21-100
21.3.40.	RSCAN0FFSTS — FIFO Full Status Register .....	21-102
21.3.41.	RSCAN0FMSTS — FIFO Message Lost Status Register .....	21-104
21.3.42.	RSCAN0RFISTS — Receive FIFO Buffer Interrupt Flag Status Register.....	21-106
21.3.43.	RSCAN0CFRISTS — Transmit/receive FIFO buffer Receive Interrupt Flag Status Register .....	21-107
21.3.44.	RSCAN0CFTISTS — Transmit/receive FIFO buffer Transmit Interrupt Flag Status Register .....	21-108
21.3.45.	RSCAN0TMCp — Transmit Buffer Control Register (p = 0 to 79).....	21-109
21.3.46.	RSCAN0TMSTSp — Transmit Buffer Status Register (p = 0 to 79).....	21-111
21.3.47.	RSCAN0TMTRSTSy — Transmit Buffer Transmit Request Status Register y (y = 0 to 2) .....	21-113
21.3.48.	RSCAN0TMTARSTSy — Transmit Buffer Transmit Abort Request Status Register y (y = 0 to 2) .....	21-115
21.3.49.	RSCAN0TMCSTSy — Transmit Buffer Transmit Complete Status Register y (y = 0 to 2) .....	21-117
21.3.50.	RSCAN0TMTASTSy — Transmit Buffer Transmit Abort Status Register y (y = 0 to 2) .....	21-119
21.3.51.	RSCAN0TMIECy — Transmit Buffer Interrupt Enable Configuration Register y (y = 0 to 2) .....	21-121
21.3.52.	RSCAN0TMIDp — Transmit Buffer ID Register (p = 0 to 79) .....	21-123
21.3.53.	RSCAN0TMPTRp — Transmit Buffer Pointer Register (p= 0 to 79).....	21-125
21.3.54.	RSCAN0TMDf0p — Transmit Buffer Data Field 0 Register (p = 0 to 79).....	21-126
21.3.55.	RSCAN0TMDf1p — Transmit Buffer Data Field 1 Register (p = 0 to 79).....	21-127
21.3.56.	RSCAN0TXQCCm — Transmit Queue Configuration and Control Register (m = 0 to 4) .....	21-128
21.3.57.	RSCAN0TXQSTSm — Transmit Queue Status Register (m = 0 to 4) .....	21-130
21.3.58.	RSCAN0TXQPCTrm — Transmit Queue Pointer Control Register (m = 0 to 4) .....	21-132
21.3.59.	RSCAN0THLCCm — Transmit History Configuration and Control Register (m = 0 to 4) .....	21-133
21.3.60.	RSCAN0THLSTSm — Transmit History Status Register (m = 0 to 4) .....	21-135
21.3.61.	RSCAN0THLACCm — Transmit History Access Register (m = 0 to 4) .....	21-137
21.3.62.	RSCAN0THLPCTrm — Transmit History Pointer Control Register (m = 0 to 4) .....	21-138
21.3.63.	RSCAN0GTSTCFG — Global Test Configuration Register.....	21-139
21.3.64.	RSCAN0GTSTCTR — Global Test Control Register.....	21-141
21.3.65.	RSCAN0GLOCKK — Global Lock Key Register .....	21-142

21.4.	Interrupt Sources.....	21-143
21.5.	RSCAN Modes.....	21-147
21.5.1.	Global Modes.....	21-147
21.5.2.	Channel Modes.....	21-150
21.6.	Reception Function.....	21-155
21.6.1.	Data Processing Using the Receive Rule Table.....	21-155
21.7.	Transmission Functions.....	21-159
21.7.1.	Transmit Priority Determination.....	21-160
21.7.2.	Transmission Using Transmit Buffers.....	21-160
21.7.3.	Transmission Using FIFO Buffers.....	21-161
21.7.4.	Transmission Using Transmit Queues.....	21-164
21.7.5.	Transmit History Function.....	21-164
21.8.	Gateway Function.....	21-166
21.9.	Test Function.....	21-167
21.9.1.	Standard Test Mode.....	21-167
21.9.2.	Listen-Only Mode.....	21-167
21.9.3.	Self-Test Mode (Loopback Mode).....	21-168
21.9.4.	Inter-Channel Communication Test.....	21-170
21.10.	RS-CAN Setting Procedure.....	21-171
21.10.1.	Initial Settings.....	21-171
21.10.2.	Reception Procedure.....	21-177
21.10.3.	Transmission Procedure.....	21-182
21.10.4.	Test Settings.....	21-191
21.11.	Notes on the RS-CAN Module.....	21-193
<b>22.</b>	<b>IEBus Controller.....</b>	<b>22-1</b>
22.1.	IEBB Features.....	22-1
22.2.	Configuration.....	22-3
22.2.1.	Function overview.....	22-3
22.2.2.	Block diagram.....	22-4
22.3.	Registers.....	22-5
22.3.1.	IEBBn register overview.....	22-5
22.3.2.	IEBBn control register details.....	22-6
22.4.	Interrupt Operations.....	22-72
22.4.1.	Interrupt request signals.....	22-72
22.4.2.	Interrupt judgment examples.....	22-77
22.5.	Operation.....	22-79
22.5.1.	FIFO.....	22-79
22.5.2.	Initial settings.....	22-81
22.5.3.	Master transmission (single mode).....	22-82
22.5.4.	Master transmission (FIFO mode).....	22-84

22.5.5.	Master reception (single mode) .....	22-86
22.5.6.	Master reception (FIFO mode) .....	22-88
22.5.7.	Slave transmission (single mode) .....	22-90
22.5.8.	Slave transmission (FIFO mode) .....	22-93
22.5.9.	Slave reception (single mode) .....	22-96
22.5.10.	Slave reception (FIFO mode) .....	22-98
22.6.	Setup Procedures .....	22-99
22.6.1.	Master transmission (single mode) .....	22-99
22.6.2.	Master transmission (FIFO mode) .....	22-100
22.6.3.	Master reception (single mode) .....	22-101
22.6.4.	Master reception (FIFO mode) .....	22-102
22.6.5.	Slave transmission (single mode) .....	22-103
22.6.6.	Slave transmission (FIFO mode) .....	22-106
22.6.7.	Slave reception (single mode) .....	22-109
22.6.8.	Slave reception (FIFO mode) .....	22-110
22.7.	Functions .....	22-111
22.7.1.	IEBus communication protocol .....	22-111
22.7.2.	Determination of bus mastership (arbitration) .....	22-112
22.7.3.	Communication mode .....	22-112
22.7.4.	Communication address .....	22-113
22.7.5.	Broadcast communication .....	22-113
22.7.6.	IEBus transfer format .....	22-114
22.7.7.	Transfer data .....	22-124
22.7.8.	Bit format .....	22-128
23.	<b>Renesas SPDIF Interface</b> .....	<b>23-1</b>
23.1	Overview .....	23-1
23.2	Features .....	23-1
23.3	Functional Block Diagram .....	23-2
23.4	Input/Output Pins .....	23-3
23.5	Renesas SPDIF (IEC60958) Frame Format .....	23-3
23.6	Register .....	23-5
23.7	Register Descriptions .....	23-6
23.7.1	Control Register (CTRL) .....	23-6
23.7.2	Status Register (STAT) .....	23-9
23.7.3	Transmitter Channel 1 Audio Register (TLCA) .....	23-11
23.7.4	Transmitter Channel 2 Audio Register (TRCA) .....	23-12
23.7.5	Transmitter DMA Audio Data Register (TDAD) .....	23-12
23.7.6	Transmitter User Data Register (TUI) .....	23-13
23.7.7	Transmitter Channel 1 Status Register (TLCS) .....	23-14
23.7.8	Transmitter Channel 2 Status Register (TRCS) .....	23-15

23.7.9	Receiver Channel 1 Audio Register (RLCA).....	23-16
23.7.10	Receiver Channel 2 Audio Register (RRCA).....	23-16
23.7.11	Receiver DMA Audio Data (RDAD).....	23-17
23.7.12	Receiver User Data Register (RUI).....	23-18
23.7.13	Receiver Channel 1 Status Register (RLCS).....	23-19
23.7.14	Receiver Channel 2 Status Register (RRCS).....	23-20
23.8	Functional Description—Transmitter.....	23-21
23.8.1	Transmitter Module.....	23-21
23.8.2	Transmitter Module Initialization.....	23-21
23.8.3	Initial Settings for Transmitter Module.....	23-22
23.8.4	Transmitter Module Data Transfer.....	23-22
23.9	Functional Description—Receiver.....	23-24
23.9.1	Receiver Module.....	23-24
23.9.2	Receiver Module Initialization.....	23-24
23.9.3	Receiver Module Data Transfer.....	23-25
23.10	Disabling the Module.....	23-27
23.10.1	Transmitter and Receiver Idle.....	23-27
23.11	Compressed Mode Data.....	23-27
23.12	References.....	23-27
23.13	Usage Notes.....	23-27
23.13.1	Clearing TUIR.....	23-27
23.13.2	Frequency of Clock Input for Audio.....	23-27
24.	CD-ROM Decoder.....	24-1
24.1	Features.....	24-1
24.1.1	Formats Supported by CD-ROM Decoder.....	24-2
24.2	Block Diagrams.....	24-2
24.3	Register Descriptions.....	24-5
24.3.1	Enable Control Register (CROMEN).....	24-7
24.3.2	Sync Code-Based Synchronization Control Register (CROMSY0).....	24-8
24.3.3	Decoding Mode Control Register (CROMCTL0).....	24-9
24.3.4	EDC/ECC Check Control Register (CROMCTL1).....	24-10
24.3.5	Automatic Decoding Stop Control Register (CROMCTL3).....	24-11
24.3.6	Decoding Option Setting Control Register (CROMCTL4).....	24-12
24.3.7	HEAD20 to HEAD22 Representation Control Register (CROMCTL5).....	24-13
24.3.8	Sync Code Status Register (CROMST0).....	24-13
24.3.9	Post-ECC Header Error Status Register (CROMST1).....	24-14
24.3.10	Post-ECC Subheader Error Status Register (CROMST3).....	24-14
24.3.11	Header/Subheader Validity Check Status Register (CROMST4).....	24-15
24.3.12	Mode Determination and Link Sector Detection Status Register (CROMST5).....	24-16
24.3.13	ECC/EDC Error Status Register (CROMST6).....	24-17



24.3.14	Buffer Status Register (CBUFST0).....	24-17
24.3.15	Decoding Stoppage Source Status Register (CBUFST1).....	24-18
24.3.16	Buffer Overflow Status Register (CBUFST2).....	24-18
24.3.17	Pre-ECC Correction Header: Minutes Data Register (HEAD00).....	24-19
24.3.18	Pre-ECC Correction Header: Seconds Data Register (HEAD01).....	24-19
24.3.19	Pre-ECC Correction Header: Frames (1/75 Second) Data Register (HEAD02).....	24-19
24.3.20	Pre-ECC Correction Header: Mode Data Register (HEAD03).....	24-19
24.3.21	Pre-ECC Correction Subheader: File Number (Byte 16) Data Register (SHEAD00).....	24-20
24.3.22	Pre-ECC Correction Subheader: Channel Number (Byte 17) Data Register (SHEAD01).....	24-20
24.3.23	Pre-ECC Correction Subheader: Sub-Mode (Byte 18) Data Register (SHEAD02).....	24-20
24.3.24	Pre-ECC Correction Subheader: Data Type (Byte 19) Data Register (SHEAD03).....	24-21
24.3.25	Pre-ECC Correction Subheader: File Number (Byte 20) Data Register (SHEAD04).....	24-21
24.3.26	Pre-ECC Correction Subheader: Channel Number (Byte 21) Data Register (SHEAD05).....	24-21
24.3.27	Pre-ECC Correction Subheader: Sub-Mode (Byte 22) Data Register (SHEAD06).....	24-22
24.3.28	Pre-ECC Correction Subheader: Data Type (Byte 23) Data Register (SHEAD07).....	24-22
24.3.29	Post-ECC Correction Header: Minutes Data Register (HEAD20).....	24-22
24.3.30	Post-ECC Correction Header: Seconds Data Register (HEAD21).....	24-23
24.3.31	Post-ECC Correction Header: Frames (1/75 Second) Data Register (HEAD22).....	24-23
24.3.32	Post-ECC Correction Header: Mode Data Register (HEAD23).....	24-23
24.3.33	Post-ECC Correction Subheader: File Number (Byte 16) Data Register (SHEAD20).....	24-24
24.3.34	Post-ECC Correction Subheader: Channel Number (Byte 17) Data Register (SHEAD21).....	24-24
24.3.35	Post-ECC Correction Subheader: Sub-Mode (Byte 18) Data Register (SHEAD22).....	24-24
24.3.36	Post-ECC Correction Subheader: Data Type (Byte 19) Data Register (SHEAD23).....	24-25
24.3.37	Post-ECC Correction Subheader: File Number (Byte 20) Data Register (SHEAD24).....	24-25
24.3.38	Post-ECC Correction Subheader: Channel Number (Byte 21) Data Register (SHEAD25).....	24-25
24.3.39	Post-ECC Correction Subheader: Sub-Mode (Byte 22) Data Register (SHEAD26).....	24-26
24.3.40	Post-ECC Correction Subheader: Data Type (Byte 23) Data Register (SHEAD27).....	24-26
24.3.41	Automatic Buffering Setting Control Register (CBUFCTL0).....	24-27
24.3.42	Automatic Buffering Start Sector Setting: Minutes Control Register (CBUFCTL1).....	24-28
24.3.43	Automatic Buffering Start Sector Setting: Seconds Control Register (CBUFCTL2).....	24-28
24.3.44	Automatic Buffering Start Sector Setting: Frames Control Register (CBUFCTL3).....	24-28
24.3.45	ISY Interrupt Source Mask Control Register (CROMST0M).....	24-29
24.3.46	CD-ROM Decoder Reset Control Register (ROMDECRST).....	24-29
24.3.47	CD-ROM Decoder Reset Status Register (RSTSTAT).....	24-30
24.3.48	Serial Sound Interface Data Control Register (SSI).....	24-31
24.3.49	Interrupt Flag Register (INTHOLD).....	24-32
24.3.50	Interrupt Source Mask Control Register (INHINT).....	24-33
24.3.51	CD-ROM Decoder Stream Data Input Register (STRMDIN0).....	24-33
24.3.52	CD-ROM Decoder Stream Data Input Register (STRMDIN2).....	24-34
24.3.53	CD-ROM Decoder Stream Data Output Register (STRMDOUT0).....	24-34

24.4	Operation .....	24-35
24.4.1	Endian Conversion for Data in the Input Stream .....	24-35
24.4.2	Sync Code Maintenance Function .....	24-36
24.4.3	Error Correction.....	24-38
24.4.4	Automatic Decoding Stop Function .....	24-39
24.4.5	Buffering Format .....	24-40
24.4.6	Target-Sector Buffering Function .....	24-41
24.5	Interrupt Sources.....	24-43
24.5.1	Interrupt and DMA Transfer Request Signals.....	24-43
24.5.2	Timing of Status Registers Updates .....	24-44
24.6	Usage Notes.....	24-45
24.6.1	Stopping and Resuming Buffering Alone during Decoding .....	24-45
24.6.2	Setting Sync Code Status Register (CROMST0) .....	24-45
24.6.3	Link Blocks.....	24-45
24.6.4	Stopping and Resuming CD-DSP Operation.....	24-45
24.6.5	Note on Clearing the IREADY Flag.....	24-45
24.6.6	Note on Stream Data Transfer (1) .....	24-46
24.6.7	Note on Stream Data Transfer (2) .....	24-46
25.	LIN Interface .....	25-1
25.1.	RZ/A1H LIN Interface Features.....	25-1
25.1.1.	Channels .....	25-1
25.1.2.	Register addresses.....	25-1
25.1.3.	Clock supply .....	25-1
25.1.4.	Interrupts and DMA.....	25-2
25.1.5.	I/O signals.....	25-2
25.2.	Function.....	25-3
25.2.1.	Block Diagram.....	25-5
25.2.2.	Description of Blocks .....	25-5
25.3.	Registers .....	25-6
25.3.1.	LIN Master Related Registers .....	25-7
25.4.	Interrupt Sources.....	25-33
25.5.	Modes .....	25-34
25.6.	LIN Reset Mode .....	25-36
25.7.	LIN Mode .....	25-37
25.7.1.	LIN Master Mode .....	25-39
25.7.2.	Data Transmission/Reception.....	25-42
25.7.3.	Transmission/Reception Data Buffering .....	25-44
25.7.4.	Wake-up Transmission/Reception.....	25-47
25.7.5.	Status .....	25-49
25.7.6.	Error Status .....	25-50

25.8.	LIN Self-Test Mode.....	25-52
25.8.1.	Change to LIN Self-Test Mode .....	25-54
25.8.2.	Transmission in LIN Master Self-Test Mode.....	25-55
25.8.3.	Reception in LIN Master Self-Test Mode .....	25-56
25.8.4.	Terminating LIN Self-Test Mode.....	25-57
25.9.	Baud Rate Generator .....	25-58
25.9.1.	LIN Master Mode (x.7.1) .....	25-58
25.10.	Noise Filter .....	25-60
<b>26.</b>	<b>Ethernet Controller.....</b>	<b>26-1</b>
26.1	Features.....	26-1
26.2	Input/Output Pins.....	26-3
26.3	Register Descriptions.....	26-4
26.3.1	Software Reset Register (ARSTR).....	26-7
26.3.2	E-MAC Mode Register (ECMR).....	26-8
26.3.3	E-MAC Status Register (ECSR).....	26-10
26.3.4	E-MAC Interrupt Permission Register (ECSIPR).....	26-11
26.3.5	PHY Interface Register (PIR).....	26-12
26.3.6	MAC Address High Register (MAHR).....	26-12
26.3.7	MAC Address Low Register (MALR).....	26-13
26.3.8	Receive Frame Length Register (RFLR).....	26-14
26.3.9	CRC Error Frame Receive Counter Register (CEFCR).....	26-15
26.3.10	Frame Receive Error Counter Register (FRECR).....	26-15
26.3.11	Too-Short Frame Receive Counter Register (TSFRCCR).....	26-16
26.3.12	Too-Long Frame Receive Counter Register (TLFRCCR).....	26-16
26.3.13	Residual-Bit Frame Receive Counter Register (RFCR).....	26-17
26.3.14	Multicast Address Frame Receive Counter Register (MAFCR).....	26-17
26.3.15	Automatic PAUSE Frame Register (APR).....	26-18
26.3.16	Manual PAUSE Frame Register (MPR).....	26-19
26.3.17	Automatic PAUSE Frame Retransmit Count Register (TPAUSER).....	26-20
26.3.18	PAUSE Frame Transmit Counter Register (PFTCR).....	26-20
26.3.19	PAUSE Frame Receive Counter Register (PFRCR).....	26-21
26.3.20	TSU Counter Reset Register (TSU_CTRST).....	26-21
26.3.21	VLANtag Set Register (TSU_VTAG0).....	26-22
26.3.22	CAM Entry Table Busy Register (TSU_ADSBSY).....	26-23
26.3.23	CAM Entry Table Enable Register (TSU_TEN).....	26-24
26.3.24	CAM Entry Table 0H to 31H Registers (TSU_ADRH0 to TSU_ADRH31).....	26-26
26.3.25	CAM Entry Table 0L to 31L Registers (TSU_ADRL0 to TSU_ADRL31).....	26-26
26.3.26	Transmit Frame Counter Register (Normal Transmission Only) (TXNLCR0).....	26-27
26.3.27	Transmit Frame Counter Register (Normal and Erroneous Transmission) (TXALCR0).....	26-27
26.3.28	Receive Frame Counter Register (Normal Reception Only) (RXNLCR0).....	26-28

26.3.29	Receive Frame Counter Register (Normal and Erroneous Reception) (RXALCR0).....	26-28
26.3.30	E-DMAC Start Register (EDSR).....	26-29
26.3.31	E-DMAC Mode Register (EDMR).....	26-30
26.3.32	E-DMAC Transmit Request Register (EDTRR).....	26-31
26.3.33	E-DMAC Receive Request Register (EDRRR).....	26-32
26.3.34	Transmit Descriptor List Start Address Register (TDLAR).....	26-33
26.3.35	Receive Descriptor List Start Address Register (RDLAR).....	26-33
26.3.36	E-MAC/E-DMAC Status Register (EESR).....	26-34
26.3.37	E-MAC/E-DMAC Status Interrupt Permission Register (EESIPR).....	26-37
26.3.38	Transmit/Receive Status Copy Enable Register (TRSCER).....	26-39
26.3.39	Receive Missed-Frame Counter Register (RMFCR).....	26-40
26.3.40	Transmit FIFO Threshold Register (TFTR).....	26-41
26.3.41	FIFO Depth Register (FDR).....	26-42
26.3.42	Receiving Method Control Register (RMCR).....	26-43
26.3.43	Receive Descriptor Fetch Address Register (RDFAR).....	26-43
26.3.44	Receive Descriptor Finished Address Register (RDFXR).....	26-44
26.3.45	Receive Descriptor Final Flag Register (RDFFR).....	26-44
26.3.46	Transmit Descriptor Fetch Address Register (TDFAR).....	26-45
26.3.47	Transmit Descriptor Finished Address Register (TDFXR).....	26-45
26.3.48	Transmit Descriptor Final Flag Register (TDFFR).....	26-46
26.3.49	Overflow Alert FIFO Threshold Register (FCFTR).....	26-47
26.3.50	Receive Data Padding Insert Register (RPADIR).....	26-48
26.3.51	Intelligent Checksum Mode Register (CSMR).....	26-49
26.3.52	Intelligent Checksum Skipped Bytes Monitor Register (CSSBM).....	26-50
26.3.53	Intelligent Checksum Monitor Register (CSSMR).....	26-50
26.4	Operation.....	26-51
26.4.1	Descriptors and Descriptor List.....	26-53
26.4.2	Transmission.....	26-64
26.4.3	Reception.....	26-68
26.4.4	CAM Function.....	26-71
26.4.5	Transmit Processing of Multi-Buffer Frame (Single-Frame/Multi-Descriptor).....	26-72
26.4.6	Padding Insertion in Receive Data.....	26-73
26.4.7	Interrupt Processing.....	26-74
26.4.8	Activation Procedure.....	26-75
26.4.9	Flow Control.....	26-76
26.4.10	Intelligent Checksum Calculation Function.....	26-77
26.5	Connection to PHY-LSI.....	26-79
26.5.1	MII Frame Transmission/Reception Timing.....	26-79
26.5.2	Accessing MII Registers.....	26-80
26.6	Usage Notes.....	26-83

26.6.1	Checksum Calculation of Ethernet Frames .....	26-83
26.6.2	Notes on Using the Intelligent Checksum Function.....	26-84
<b>27.</b>	<b>A/D Converter .....</b>	<b>27-1</b>
27.1	Features.....	27-1
27.2	Input/Output Pins.....	27-3
27.3	Register Descriptions.....	27-4
27.3.1	A/D Data Registers A to H (ADDRA to ADDRH).....	27-5
27.3.2	A/D Comparison Upper Limit Value Registers A to H (ADCMPHA to ADCMPHH).....	27-6
27.3.3	A/D Comparison Lower Limit Value Registers A to H (ADCMPHA to ADCMPLH) .....	27-7
27.3.4	A/D Control/Status Register (ADCSR).....	27-8
27.3.5	A/D Comparison Interrupt Enable Register (ADCMPER) .....	27-10
27.3.6	A/D Comparison Status Register (ADCMPSTR).....	27-11
27.4	Operation .....	27-12
27.4.1	Single Mode.....	27-12
27.4.2	Multi Mode.....	27-14
27.4.3	Scan Mode .....	27-16
27.4.4	A/D Converter Activation by External Trigger or Multi-Function Timer Pulse Unit 2 .....	27-18
27.4.5	Input Sampling and A/D Conversion Time.....	27-18
27.4.6	External Trigger Input Timing .....	27-20
27.5	Interrupt Sources and DMA Transfer Request.....	27-21
27.6	Definitions of A/D Conversion Accuracy .....	27-22
27.7	Usage Notes.....	27-23
27.7.1	Module Standby Mode Setting .....	27-23
27.7.2	Setting Analog Input Voltage.....	27-23
27.7.3	Notes on Board Design.....	27-23
27.7.4	Processing of Analog Input Pins .....	27-24
27.7.5	Permissible Signal Source Impedance.....	27-25
27.7.6	Influences on Absolute Precision .....	27-25
27.7.7	Usage Note on Port Pins.....	27-25
<b>28.</b>	<b>NAND Flash Memory Controller .....</b>	<b>28-1</b>
28.1	Features.....	28-1
28.2	Input/Output Pins.....	28-3
28.3	Register Descriptions.....	28-4
28.3.1	Common Control Register (FLCMNCR).....	28-5
28.3.2	Command Control Register (FLCMDRCR) .....	28-6
28.3.3	Command Code Register (FLCMCDR).....	28-7
28.3.4	Address Register (FLADR).....	28-8
28.3.5	Address Register 2 (FLADR2).....	28-8
28.3.6	Data Counter Register (FLDTCNTR).....	28-9

28.3.7	Data Register (FLDATAR) .....	28-10
28.3.8	Interrupt DMA Control Register (FLINTDMACR).....	28-11
28.3.9	Ready Busy Timeout Setting Register (FLBSYTMR).....	28-13
28.3.10	Ready Busy Timeout Counter (FLBSYCNT) .....	28-13
28.3.11	Data FIFO Register (FLDTFIFO) .....	28-14
28.3.12	Transfer Control Register (FLTRCR) .....	28-14
28.4	Operation .....	28-15
28.4.1	Access Sequence.....	28-15
28.4.2	Register Setting Procedure .....	28-16
28.4.3	Command Access Mode.....	28-17
28.4.4	Status Read .....	28-20
28.5	Interrupt Processing.....	28-21
28.6	DMA Transfer Settings .....	28-21
28.7	Usage Notes .....	28-22
28.7.1	Usage Note for the SNAND Bit.....	28-22
29.	USB2.0 Host/Function Module.....	29-1
29.1	Features.....	29-1
29.2	Input/Output Pins.....	29-3
29.3	Register Descriptions.....	29-4
29.3.1	System Configuration Control Register (SYSCFG0).....	29-10
29.3.2	CPU Bus Wait Setting Register (BUSWAIT).....	29-12
29.3.3	System Configuration Status Register (SYSSTS0) .....	29-13
29.3.4	Device State Control Register 0 (DVSTCTR0) .....	29-14
29.3.5	Test Mode Register (TESTMODE).....	29-17
29.3.6	DMA <sub>n</sub> -FIFO Bus Configuration Registers (D0FBCFG, D1FBCFG).....	29-18
29.3.7	FIFO Port Registers (CFIFO, D0FIFO, D1FIFO).....	29-19
29.3.8	FIFO Port Select Registers (CFIFOSEL, D0FIFOSEL, D1FIFOSEL).....	29-21
29.3.9	FIFO Port Control Registers (CFIFOCTR, D0FIFOCTR, D1FIFOCTR) .....	29-25
29.3.10	Interrupt Enable Register 0 (INTENB0) .....	29-27
29.3.11	Interrupt Enable Register 1 (INTENB1) .....	29-28
29.3.12	BRDY Interrupt Enable Register (BRDYENB).....	29-29
29.3.13	NRDY Interrupt Enable Register (NRDYENB) .....	29-30
29.3.14	BEMP Interrupt Enable Register (BEMPENB) .....	29-30
29.3.15	SOF Output Configuration Register (SOFCFG) .....	29-31
29.3.16	Interrupt Status Register 0 (INTSTS0).....	29-32
29.3.17	Interrupt Status Register 1 (INTSTS1).....	29-35
29.3.18	BRDY Interrupt Status Register (BRDYSTS) .....	29-37
29.3.19	NRDY Interrupt Status Register (NRDYSTS).....	29-38
29.3.20	BEMP Interrupt Status Register (BEMPSTS).....	29-38
29.3.21	Frame Number Register (FRMNUM).....	29-39

29.3.22	μFrame Number Register (UFRMNUM).....	29-40
29.3.23	USB Address Register (USBADDR).....	29-40
29.3.24	USB Request Type Register (USBREQ).....	29-41
29.3.25	USB Request Value Register (USBVAL).....	29-42
29.3.26	USB Request Index Register (USBINDX).....	29-42
29.3.27	USB Request Length Register (USBLENG).....	29-43
29.3.28	DCP Configuration Register (DCPCFG).....	29-43
29.3.29	DCP Maximum Packet Size Register (DCPMAXP).....	29-46
29.3.30	DCP Control Register (DCPCTR).....	29-47
29.3.31	Pipe Window Select Register (PIPESEL).....	29-51
29.3.32	Pipe Configuration Register (PIPECFG).....	29-52
29.3.33	Pipe Buffer Setting Register (PIPEBUF).....	29-56
29.3.34	Pipe Maximum Packet Size Register (PIPEMAXP).....	29-58
29.3.35	Pipe Timing Control Register (PIPEPERI).....	29-59
29.3.36	PIPE <sub>n</sub> Control Registers (PIPE <sub>n</sub> CTR) (n = 1 to F).....	29-60
29.3.37	PIPE <sub>n</sub> Transaction Counter Enable Registers (PIPE <sub>n</sub> TRE) (n = 1 to 5, 9, and A to F).....	29-69
29.3.38	PIPE <sub>n</sub> Transaction Counter Registers (PIPE <sub>n</sub> TRN) (n = 1 to 5, 9, and A to F).....	29-70
29.3.39	Device Address n Configuration Registers (DEVADD <sub>n</sub> ) (n = 0 to 9, and A).....	29-71
29.3.40	UTMI Suspend Mode Register (SUSPMODE).....	29-72
29.3.41	FIFO Continuous Transfer Port Registers (D0FIFOB <sub>n</sub> , D1FIFOB <sub>n</sub> ) (n = 0 to 7).....	29-73
29.4	Operation.....	29-74
29.4.1	System Control and Oscillation Control.....	29-74
29.4.2	Interrupt Functions.....	29-76
29.4.3	Pipe Control.....	29-88
29.4.4	FIFO Buffer Memory.....	29-94
29.4.5	FIFO Port Functions.....	29-96
29.4.6	Control Transfers (DCP).....	29-99
29.4.7	Bulk Transfers (PIPE1 to PIPE5, PIPE9 to PIPE15).....	29-101
29.4.8	Interrupt Transfers (PIPE6 to PIPE9, PIPE10).....	29-103
29.4.9	Isochronous Transfers (PIPE1 and PIPE2).....	29-104
29.4.10	SOF Interpolation Function.....	29-112
30.	Digital Video Decoder.....	30-1
30.1	Features.....	30-1
30.2	Block Diagram.....	30-2
30.3	Input/Output Pins.....	30-3
30.4	Register Descriptions.....	30-4
30.4.1	ADC Control Register 1 (ADCCR1).....	30-10
30.4.2	Timing Generation Control Register 1 (TGCR1).....	30-10
30.4.3	Timing Generation Control Register 2 (TGCR2).....	30-11
30.4.4	Timing Generation Control Register 3 (TGCR3).....	30-11

30.4.5	Sync Separation Control Register 1 (SYNSCR1) .....	30-14
30.4.6	Sync Separation Control Register 2 (SYNSCR2) .....	30-18
30.4.7	Sync Separation Control Register 3 (SYNSCR3) .....	30-19
30.4.8	Sync Separation Control Register 4 (SYNSCR4) .....	30-20
30.4.9	Sync Separation Control Register 5 (SYNSCR5) .....	30-21
30.4.10	Horizontal AFC Control Register 1 (HAFCCR1) .....	30-22
30.4.11	Horizontal AFC Control Register 2 (HAFCCR2) .....	30-24
30.4.12	Horizontal AFC Control Register 3 (HAFCCR3) .....	30-25
30.4.13	Vertical Countdown Control Register 1 (VCDWCR1) .....	30-26
30.4.14	Digital Clamp Control Register 1 (DCPCR1) .....	30-28
30.4.15	Digital Clamp Control Register 2 (DCPCR2) .....	30-29
30.4.16	Digital Clamp Control Register 3 (DCPCR3) .....	30-30
30.4.17	Digital Clamp Control Register 4 (DCPCR4) .....	30-30
30.4.18	Digital Clamp Control Register 5 (DCPCR5) .....	30-31
30.4.19	Digital Clamp Control Register 6 (DCPCR6) .....	30-31
30.4.20	Digital Clamp Control Register 7 (DCPCR7) .....	30-32
30.4.21	Digital Clamp Control Register 8 (DCPCR8) .....	30-32
30.4.22	Noise Detection Control Register (NSDCR) .....	30-33
30.4.23	Burst Lock/Chroma Decoding Control Register (BTLCR) .....	30-34
30.4.24	Burst Gate Pulse Control Register (BTGPCR) .....	30-37
30.4.25	ACC Control Register 1 (ACCCR1) .....	30-38
30.4.26	ACC Control Register 2 (ACCCR2) .....	30-40
30.4.27	ACC Control Register 3 (ACCCR3) .....	30-41
30.4.28	TINT Control Register (TINTCR) .....	30-42
30.4.29	Y/C Delay/Chroma Decoding Control Register (YCDCR) .....	30-43
30.4.30	AGC Control Register 1 (AGCCR1) .....	30-44
30.4.31	AGC Control Register 2 (AGCCR2) .....	30-46
30.4.32	Peak Limiter Control Register (PKLIMITCR) .....	30-47
30.4.33	Over-Range Control Register 1 (RGORCR1) .....	30-49
30.4.34	Over-Range Control Register 2 (RGORCR2) .....	30-50
30.4.35	Over-Range Control Register 3 (RGORCR3) .....	30-50
30.4.36	Over-Range Control Register 4 (RGORCR4) .....	30-51
30.4.37	Over-Range Control Register 5 (RGORCR5) .....	30-51
30.4.38	Over-Range Control Register 6 (RGORCR6) .....	30-52
30.4.39	Over-Range Control Register 7 (RGORCR7) .....	30-53
30.4.40	Feedback Control Register for Horizontal AFC Phase Comparator (AFCPFCR) .....	30-54
30.4.41	Register Update Enable Register (RUPDCR) .....	30-55
30.4.42	Sync Separation Status/Vertical Cycle Read Register (VSYNCSR) .....	30-56
30.4.43	Horizontal Cycle Read Register (HSYNCSR) .....	30-57
30.4.44	Digital Clamp Read Register 1 (DCPSR1) .....	30-57



30.4.45	Digital Clamp Read Register 2 (DCPSR2).....	30-58
30.4.46	Noise Detection Read Register (NSDSR) .....	30-58
30.4.47	Chroma Decoding Read Register 1 (CROMASR1).....	30-59
30.4.48	Chroma Decode Read Register 2 (CROMASR2) .....	30-60
30.4.49	Sync Separation Read Register (SYNCSSR).....	30-61
30.4.50	AGC Control Read Register 1 (AGCCSR1) .....	30-61
30.4.51	AGC Control Read Register 2 (AGCCSR2) .....	30-62
30.4.52	Y/C Separation Control Register 3 (YCSCR3).....	30-63
30.4.53	Y/C Separation Control Register 4 (YCSCR4).....	30-64
30.4.54	Y/C Separation Control Register 5 (YCSCR5).....	30-65
30.4.55	Y/C Separation Control Register 6 (YCSCR6).....	30-65
30.4.56	Y/C Separation Control Register 7 (YCSCR7).....	30-66
30.4.57	Y/C Separation Control Register 8 (YCSCR8).....	30-67
30.4.58	Y/C Separation Control Register 9 (YCSCR9).....	30-68
30.4.59	Y/C Separation Control Register 11 (YCSCR11).....	30-69
30.4.60	Y/C Separation Control Register 12 (YCSCR12).....	30-70
30.4.61	Digital Clamp Control Register 9 (DCPCR9).....	30-71
30.4.62	Chroma Filter TAP Coefficient (WA_F0 to WA_F8) Registers for Y/C Separation (YCTWA_F0 to YCTWA_F8).....	30-72
30.4.63	Chroma Filter TAP Coefficient (WB_F0 to WB_F8) Registers for Y/C Separation (YCTWB_F0 to YCTWB_F8).....	30-73
30.4.64	Chroma Filter TAP Coefficient (NA_F0 to NA_F8) Registers for Y/C Separation (YCTNA_F0 to YCTNA_F8) .....	30-74
30.4.65	Chroma Filter TAP Coefficient (NB_F0 to NB_F8) Registers for Y/C Separation (YCTNB_F0 to YCTNB_F8).....	30-75
30.4.66	Luminance (Y) Signal Gain Control Register (YGAINCR).....	30-76
30.4.67	Color Difference (Cb) Signal Gain Control Register (CBGAINCR).....	30-76
30.4.68	Color Difference (Cr) Signal Gain Control Register (CRGAINCR) .....	30-77
30.4.69	PGA Register Update (PGA_UPDATE).....	30-77
30.4.70	PGA Control Register (PGACR).....	30-78
30.4.71	ADC Control Register 2 (ADCCR2).....	30-78
30.5	Operation .....	30-79
30.5.1	Overview .....	30-79
30.5.2	A/D Converter for Video Signal Input .....	30-80
30.5.3	Sync Separator Circuit.....	30-82
30.5.4	Burst Controlled Oscillator (BCO).....	30-86
30.5.5	Y/C Separator Circuit .....	30-87
30.5.6	Chroma Decoding Circuit.....	30-94
30.5.7	Digital Clamp Circuit .....	30-95
30.5.8	Output Control Circuit.....	30-97
30.6	Recommended Setting.....	30-98

30.7	Connection Example.....	30-103
31.	Video Display Controller 5 (1): Overview .....	31-1
31.1	Features.....	31-1
31.2	Block Diagram.....	31-3
31.3	Input/Output Pins.....	31-4
31.4	Clocks.....	31-5
31.5	Hsync and Vsync Signals .....	31-5
31.5.1	External Input Vsync.....	31-6
31.5.2	Free-Running Vsync.....	31-7
31.5.3	Blending Two Input Video Images .....	31-9
31.5.4	Usage Note on Changing Vsync Signal Selections.....	31-9
32.	Video Display Controller 5 (2): Input Controller .....	32-1
32.1	Input Controller Functions.....	32-1
32.1.1	Overview of Functions .....	32-1
32.1.2	Updating Registers of External Signal Input Block and Sync Signal Adjustment Block .....	32-2
32.1.3	Selecting Input Signals .....	32-2
32.1.4	Controlling Externally Input Video Signals .....	32-3
32.1.5	Selecting Clock Edge for Externally Input Signals .....	32-4
32.1.6	Externally Input Sync Signal Inversion Control.....	32-4
32.1.7	Bit Allocation of Externally Input Video Image Signals .....	32-5
32.1.8	Typical Signal Timing of BT601 Format.....	32-9
32.1.9	Typical Signal Timing of BT656 Format.....	32-12
32.1.10	SAV/EAV Code in BT656 Format.....	32-14
32.1.11	BT656 Progressive Format.....	32-17
32.1.12	BT656/BT601/YCbCr422 Format Setting .....	32-20
32.1.13	YCbCr444/RBG888/666/565 Input Timing.....	32-23
32.1.14	Field Differentiation and Vsync Signal Phase Adjustment.....	32-25
32.1.15	Vsync Signal Delay Adjustment in Line Units .....	32-26
32.1.16	Sync Signal Delay Adjustment.....	32-26
32.1.17	Horizontal Noise Reduction .....	32-27
32.1.18	Color Matrix .....	32-29
32.2	Register Descriptions.....	32-32
32.2.1	External Input Block Register Update Control Register (INP_UPDATE) .....	32-34
32.2.2	Input Select Control Register (INP_SEL_CNT) .....	32-35
32.2.3	External Input Sync Signal Control Register (INP_EXT_SYNC_CNT).....	32-36
32.2.4	Vsync Signal Phase Adjustment Register (INP_VSYNC_PH_ADJ) .....	32-37
32.2.5	Sync Signal Delay Adjustment Register (INP_DLY_ADJ).....	32-37
32.2.6	Image Quality Adjustment Block Register Update Control Register (IMGCNT_UPDATE).....	32-38
32.2.7	NR Control Register 0 (IMGCNT_NR_CNT0).....	32-39

32.2.8	NR Control Register 1 (IMGCNT_NR_CNT1) .....	32-40
32.2.9	Image Quality Adjustment Block Matrix Mode Register (IMGCNT_MTX_MODE) .....	32-41
32.2.10	Image Quality Adjustment Block Matrix YG Adjustment Register 0 (IMGCNT_MTX_YG_ADJ0).....	32-41
32.2.11	Image Quality Adjustment Block Matrix YG Adjustment Register 1 (IMGCNT_MTX_YG_ADJ1).....	32-42
32.2.12	Image Quality Adjustment Block Matrix CBB Adjustment Register 0 (IMGCNT_MTX_CBB_ADJ0).....	32-43
32.2.13	Image Quality Adjustment Block Matrix CBB Adjustment Register 1 (IMGCNT_MTX_CBB_ADJ1).....	32-44
32.2.14	Image Quality Adjustment Block Matrix CRR Adjustment Register 0 (IMGCNT_MTX_CRR_ADJ0).....	32-45
32.2.15	Image Quality Adjustment Block Matrix CRR Adjustment Register 1 (IMGCNT_MTX_CRR_ADJ1).....	32-46
32.2.16	Dynamic Range Compression Register (IMGCNT_DRC_REG) .....	32-46
32.3	Usage Methods .....	32-47
32.3.1	Input Format Adjustment Method .....	32-47
32.3.2	Usage Method of Conversion Color Matrix .....	32-50
33.	<b>Video Display Controller 5 (3): Scaler .....</b>	<b>33-1</b>
33.1	Scaler .....	33-1
33.1.1	Overview of Functions .....	33-1
33.1.2	Register Control.....	33-2
33.1.3	Synchronization Control.....	33-4
33.1.4	Setting Angle of View .....	33-9
33.1.5	Scaling Settings .....	33-13
33.1.6	Horizontal Prefilter .....	33-15
33.1.7	Horizontal Scale-Down .....	33-15
33.1.8	Vertical Scale-Down.....	33-17
33.1.9	Horizontal Scale Up.....	33-19
33.1.10	Vertical Scale-Up .....	33-20
33.1.11	IP Conversion .....	33-22
33.1.12	Control of Interrupt on Specified Image Line before Scaling-down, and Reading of Current Image Line before Scaling-down .....	33-24
33.1.13	Trimming .....	33-25
33.1.14	Screen Synthesis.....	33-26
33.1.15	Selecting Format for Writing Video Image Signals to Frame Buffer .....	33-27
33.1.16	Horizontal Mirroring and Rotation.....	33-28
33.1.17	Writing to Frame Buffer .....	33-28
33.1.18	Selecting a Scaling-up Process or Graphics 0 or 1 Process.....	33-34
33.1.19	Selecting Field for Frame Buffer Reading .....	33-35
33.1.20	Pointer Buffer and Frame Buffer Reading Processing .....	33-36
33.1.21	Cascaded Connection .....	33-38

33.1.22	Blending Two Input Video Images .....	33-38
33.2	Register Descriptions.....	33-39
33.2.1	SCL0 Register Update Control Register (SC0_SCL0_UPDATE).....	33-47
33.2.2	Mask Control Register (SC0_SCL0_FRC1) .....	33-48
33.2.3	Missing Vsync Compensation Control Register (SC0_SCL0_FRC2).....	33-48
33.2.4	Output Sync Select Register (SC0_SCL0_FRC3).....	33-49
33.2.5	Free-Running Period Control Register (SC0_SCL0_FRC4).....	33-50
33.2.6	Output Delay Control Register (SC0_SCL0_FRC5).....	33-50
33.2.7	Full-Screen Vertical Size Register (SC0_SCL0_FRC6).....	33-51
33.2.8	Full-Screen Horizontal Size Register (SC0_SCL0_FRC7).....	33-52
33.2.9	Vsync Detection Register (SC0_SCL0_FRC9).....	33-53
33.2.10	Status Monitor 0 Register (SC0_SCL0_MON0).....	33-53
33.2.11	Interrupt Control Register (SC0_SCL0_INT).....	33-54
33.2.12	Scaling-Down Control Register (SC0_SCL0_DS1).....	33-54
33.2.13	Vertical Capture Size Register (SC0_SCL0_DS2) .....	33-55
33.2.14	Horizontal Capture Size Register (SC0_SCL0_DS3).....	33-56
33.2.15	Horizontal Scale Down Register (SC0_SCL0_DS4) .....	33-57
33.2.16	Initial Vertical Phase Register (SC0_SCL0_DS5).....	33-58
33.2.17	Vertical Scaling Register (SC0_SCL0_DS6).....	33-59
33.2.18	Scaling-Down Control Block Output Size Register (SC0_SCL0_DS7).....	33-60
33.2.19	Scaling-Up Control Register (SC0_SCL0_US1) .....	33-61
33.2.20	Output Image Vertical Size Register (SC0_SCL0_US2).....	33-62
33.2.21	Output Image Horizontal Size Register (SC0_SCL0_US3).....	33-63
33.2.22	Scaling-Up Control Block Input Size Register (SC0_SCL0_US4).....	33-64
33.2.23	Horizontal Scale Up Register (SC0_SCL0_US5) .....	33-64
33.2.24	Horizontal Scale Up Initial Phase Register (SC0_SCL0_US6) .....	33-65
33.2.25	Trimming Register (SC0_SCL0_US7).....	33-65
33.2.26	Frame Buffer Read Select Register (SC0_SCL0_US8) .....	33-66
33.2.27	Background Color Register (SC0_SCL0_OVR1).....	33-66
33.2.28	SCL1 Register Update Control Register (SC0_SCL1_UPDATE).....	33-67
33.2.29	Writing Mode Register (SC0_SCL1_WR1).....	33-68
33.2.30	Write Address Register 1T (SC0_SCL1_WR2).....	33-69
33.2.31	Write Address Register 2T (SC0_SCL1_WR3).....	33-70
33.2.32	Write Address Register 3T (SC0_SCL1_WR4).....	33-71
33.2.33	Frame Sub-Sampling Register (SC0_SCL1_WR5) .....	33-72
33.2.34	Bit Reduction Register (SC0_SCL1_WR6) .....	33-73
33.2.35	Write Detection Register (SC0_SCL1_WR7).....	33-73
33.2.36	Write Address Register 1B (SC0_SCL1_WR8).....	33-74
33.2.37	Write Address Register 2B (SC0_SCL1_WR9).....	33-75
33.2.38	Write Address Register 3B (SC0_SCL1_WR10).....	33-76

33.2.39	Write Detection Register B (SC0_SCL1_WR11) .....	33-76
33.2.40	Status Monitor 1 Register (SC0_SCL1_MON1).....	33-77
33.2.41	Pointer Buffer 0 Register (SC0_SCL1_PBUF0).....	33-77
33.2.42	Pointer Buffer 1 Register (SC0_SCL1_PBUF1).....	33-78
33.2.43	Pointer Buffer 2 Register (SC0_SCL1_PBUF2).....	33-78
33.2.44	Pointer Buffer 3 Register (SC0_SCL1_PBUF3).....	33-79
33.2.45	Pointer Buffer and Field Information Register (SC0_SCL1_PBUF_FLD) .....	33-79
33.2.46	Pointer Buffer Control Register (SC0_SCL1_PBUF_CNT).....	33-80
33.2.47	Graphics 0 Register Update Control Register (GR0_UPDATE) .....	33-81
33.2.48	Frame Buffer Read Control Register (Graphics 0) (GR0_FLM_RD).....	33-81
33.2.49	Frame Buffer Control Register 1 (Graphics 0) (GR0_FLM1) .....	33-82
33.2.50	Frame Buffer Control Register 2 (Graphics 0) (GR0_FLM2) .....	33-83
33.2.51	Frame Buffer Control Register 3 (Graphics 0) (GR0_FLM3) .....	33-84
33.2.52	Frame Buffer Control Register 4 (Graphics 0) (GR0_FLM4) .....	33-85
33.2.53	Frame Buffer Control Register 5 (Graphics 0) (GR0_FLM5) .....	33-86
33.2.54	Frame Buffer Control Register 6 (Graphics 0) (GR0_FLM6) .....	33-87
33.2.55	Alpha Blending Control Register 1 (Graphics 0) (GR0_AB1) .....	33-89
33.2.56	Alpha Blending Control Register 2 (Graphics 0) (GR0_AB2) .....	33-90
33.2.57	Alpha Blending Control Register 3 (Graphics 0) (GR0_AB3) .....	33-90
33.2.58	Alpha Blending Control Register 7 (Graphics 0) (GR0_AB7) .....	33-91
33.2.59	Alpha Blending Control Register 8 (Graphics 0) (GR0_AB8) .....	33-92
33.2.60	Alpha Blending Control Register 9 (Graphics 0) (GR0_AB9) .....	33-92
33.2.61	Alpha Blending Control Register 10 (Graphics 0) (GR0_AB10) .....	33-93
33.2.62	Alpha Blending Control Register 11 (Graphics 0) (GR0_AB11) .....	33-94
33.2.63	Background Color Control Register (Graphics 0) (GR0_BASE) .....	33-94
33.2.64	CLUT Table Control Register (Graphics 0) (GR0_CLUT) .....	33-95
33.2.65	SCL0 Register Update Control Register (SC1_SCL0_UPDATE).....	33-96
33.2.66	Mask Control Register (SC1_SCL0_FRC1) .....	33-97
33.2.67	Missing Vsync Compensation Control Register (SC1_SCL0_FRC2).....	33-97
33.2.68	Output Sync Select Register (SC1_SCL0_FRC3).....	33-98
33.2.69	Free-Running Period Control Register (SC1_SCL0_FRC4).....	33-99
33.2.70	Output Delay Control Register (SC1_SCL0_FRC5).....	33-99
33.2.71	Full-Screen Vertical Size Register (SC1_SCL0_FRC6).....	33-100
33.2.72	Full-Screen Horizontal Size Register (SC1_SCL0_FRC7).....	33-100
33.2.73	Vsync Detection Register (SC1_SCL0_FRC9).....	33-101
33.2.74	Status Monitor 0 Register (SC1_SCL0_MON0).....	33-101
33.2.75	Interrupt Control Register (SC1_SCL0_INT) .....	33-102
33.2.76	Scaling-Down Control Register (SC1_SCL0_DS1).....	33-102
33.2.77	Vertical Capture Size Register (SC1_SCL0_DS2) .....	33-103
33.2.78	Horizontal Capture Size Register (SC1_SCL0_DS3) .....	33-104

33.2.79	Horizontal Scale Down Register (SC1_SCL0_DS4) .....	33-105
33.2.80	Initial Vertical Phase Register (SC1_SCL0_DS5) .....	33-105
33.2.81	Vertical Scaling Register (SC1_SCL0_DS6) .....	33-106
33.2.82	Scaling-Down Control Block Output Size Register (SC1_SCL0_DS7) .....	33-107
33.2.83	Scaling-Up Control Register (SC1_SCL0_US1) .....	33-107
33.2.84	Output Image Vertical Size Register (SC1_SCL0_US2) .....	33-108
33.2.85	Output Image Horizontal Size Register (SC1_SCL0_US3) .....	33-109
33.2.86	Scaling-Up Control Block Input Size Register (SC1_SCL0_US4) .....	33-109
33.2.87	Horizontal Scale Up Register (SC1_SCL0_US5) .....	33-110
33.2.88	Horizontal Scale Up Initial Phase Register (SC1_SCL0_US6) .....	33-111
33.2.89	Trimming Register (SC1_SCL0_US7) .....	33-111
33.2.90	Frame Buffer Read Select Register (SC1_SCL0_US8) .....	33-112
33.2.91	Background Color Register (SC1_SCL0_OVR1) .....	33-113
33.2.92	SCL1 Register Update Control Register (SC1_SCL1_UPDATE) .....	33-114
33.2.93	Writing Mode Register (SC1_SCL1_WR1) .....	33-114
33.2.94	Write Address Register 1T (SC1_SCL1_WR2) .....	33-117
33.2.95	Write Address Register 2T (SC1_SCL1_WR3) .....	33-118
33.2.96	Write Address Register 3T (SC1_SCL1_WR4) .....	33-119
33.2.97	Frame Sub-Sampling Register (SC1_SCL1_WR5) .....	33-120
33.2.98	Bit Reduction Register (SC1_SCL1_WR6) .....	33-121
33.2.99	Write Detection Register (SC1_SCL1_WR7) .....	33-121
33.2.100	Write Address Register 1B (SC1_SCL1_WR8) .....	33-122
33.2.101	Write Address Register 2B (SC1_SCL1_WR9) .....	33-123
33.2.102	Write Address Register 3B (SC1_SCL1_WR10) .....	33-124
33.2.103	Write Detection Register B (SC1_SCL1_WR11) .....	33-125
33.2.104	Status Monitor 1 Register (SC1_SCL1_MON1) .....	33-125
33.2.105	Pointer Buffer 0 Register (SC1_SCL1_PBUF0) .....	33-126
33.2.106	Pointer Buffer 1 Register (SC1_SCL1_PBUF1) .....	33-126
33.2.107	Pointer Buffer 2 Register (SC1_SCL1_PBUF2) .....	33-127
33.2.108	Pointer Buffer 3 Register (SC1_SCL1_PBUF3) .....	33-127
33.2.109	Pointer Buffer and Field Information Register (SC1_SCL1_PBUF_FLD) .....	33-128
33.2.110	Pointer Buffer Control Register (SC1_SCL1_PBUF_CNT) .....	33-129
33.2.111	Graphics 1 Register Update Control Register (GR1_UPDATE) .....	33-129
33.2.112	Frame Buffer Read Control Register (Graphics 1) (GR1_FLM_RD) .....	33-130
33.2.113	Frame Buffer Control Register 1 (Graphics 1) (GR1_FLM1) .....	33-131
33.2.114	Frame Buffer Control Register 2 (Graphics 1) (GR1_FLM2) .....	33-132
33.2.115	Frame Buffer Control Register 3 (Graphics 1) (GR1_FLM3) .....	33-132
33.2.116	Frame Buffer Control Register 4 (Graphics 1) (GR1_FLM4) .....	33-133
33.2.117	Frame Buffer Control Register 5 (Graphics 1) (GR1_FLM5) .....	33-134
33.2.118	Frame Buffer Control Register 6 (Graphics 1) (GR1_FLM6) .....	33-135

33.2.119	Alpha Blending Control Register 1 (Graphics 1) (GR1_AB1) .....	33-137
33.2.120	Alpha Blending Control Register 2 (Graphics 1) (GR1_AB2) .....	33-138
33.2.121	Alpha Blending Control Register 3 (Graphics 1) (GR1_AB3) .....	33-139
33.2.122	Alpha Blending Control Register 4 (Graphics 1) (GR1_AB4) .....	33-139
33.2.123	Alpha Blending Control Register 5 (Graphics 1) (GR1_AB5) .....	33-140
33.2.124	Alpha Blending Control Register 6 (Graphics 1) (GR1_AB6) .....	33-140
33.2.125	Alpha Blending Control Register 7 (Graphics 1) (GR1_AB7) .....	33-141
33.2.126	Alpha Blending Control Register 8 (Graphics 1) (GR1_AB8) .....	33-142
33.2.127	Alpha Blending Control Register 9 (Graphics 1) (GR1_AB9) .....	33-142
33.2.128	Alpha Blending Control Register 10 (Graphics 1) (GR1_AB10) .....	33-143
33.2.129	Alpha Blending Control Register 11 (Graphics 1) (GR1_AB11) .....	33-144
33.2.130	Background Color Control Register (Graphics 1) (GR1_BASE) .....	33-144
33.2.131	CLUT Table Control Register (Graphics 1) (GR1_CLUT) .....	33-145
33.2.132	Status Monitor Register (Graphics 1) (GR1_MON) .....	33-146
33.3	Usage Method .....	33-147
33.3.1	Scaling Setting Example for 525i Video Input and VGA-Size (640 x 480) Video Output .....	33-147
33.3.2	Scaling Setting Example for Graphics Display .....	33-150
33.3.3	Scaling Setting Example for Scaled-up Graphics Display .....	33-152
<b>34.</b>	<b>Video Display Controller 5 (4): Image Quality Improver .....</b>	<b>34-1</b>
34.1	Image Quality Improver .....	34-1
34.1.1	Overview of Functions .....	34-1
34.1.2	Register Update Control .....	34-2
34.1.3	Black Stretch .....	34-2
34.1.4	Enhancer .....	34-3
34.1.5	Color Matrix .....	34-9
34.2	Register Description .....	34-11
34.2.1	Register Update Control Register in Image Quality Improver (ADJ0_UPDATE) .....	34-14
34.2.2	Black Stretch Register (ADJ0_BKSTR_SET) .....	34-15
34.2.3	Enhancer Timing Adjustment Register 1 (ADJ0_ENH_TIM1) .....	34-16
34.2.4	Enhancer Timing Adjustment Register 2 (ADJ0_ENH_TIM2) .....	34-16
34.2.5	Enhancer Timing Adjustment Register 3 (ADJ0_ENH_TIM3) .....	34-17
34.2.6	Enhancer Sharpness Register 1 (ADJ0_ENH_SHP1) .....	34-17
34.2.7	Enhancer Sharpness Register 2 (ADJ0_ENH_SHP2) .....	34-18
34.2.8	Enhancer Sharpness Register 3 (ADJ0_ENH_SHP3) .....	34-19
34.2.9	Enhancer Sharpness Register 4 (ADJ0_ENH_SHP4) .....	34-20
34.2.10	Enhancer Sharpness Register 5 (ADJ0_ENH_SHP5) .....	34-20
34.2.11	Enhancer Sharpness Register 6 (ADJ0_ENH_SHP6) .....	34-21
34.2.12	Enhancer LTI Register 1 (ADJ0_ENH_LTI1) .....	34-22
34.2.13	Enhancer LTI Register 2 (ADJ0_ENH_LTI2) .....	34-23

34.2.14	Matrix Mode Register in Image Quality Improver (ADJ0_MTX_MODE).....	34-24
34.2.15	Matrix YG Control Register 0 in Image Quality Improver (ADJ0_MTX_YG_ADJ0).....	34-24
34.2.16	Matrix YG Control Register 1 in Image Quality Improver (ADJ0_MTX_YG_ADJ1).....	34-25
34.2.17	Matrix CBB Control Register 0 in Image Quality Improver (ADJ0_MTX_CBB_ADJ0).....	34-26
34.2.18	Matrix CBB Control Register 1 in Image Quality Improver (ADJ0_MTX_CBB_ADJ1).....	34-27
34.2.19	Matrix CRR Control Register 0 in Image Quality Improver (ADJ0_MTX_CRR_ADJ0).....	34-28
34.2.20	Matrix CRR Control Register 1 in Image Quality Improver (ADJ0_MTX_CRR_ADJ1).....	34-29
34.2.21	Register Update Control Register in Image Quality Improver (ADJ1_UPDATE).....	34-29
34.2.22	Black Stretch Register (ADJ1_BKSTR_SET).....	34-30
34.2.23	Enhancer Timing Adjustment Register 1 (ADJ1_ENH_TIM1).....	34-31
34.2.24	Enhancer Timing Adjustment Register 2 (ADJ1_ENH_TIM2).....	34-31
34.2.25	Enhancer Timing Adjustment Register 3 (ADJ1_ENH_TIM3).....	34-32
34.2.26	Enhancer Sharpness Register 1 (ADJ1_ENH_SHP1).....	34-33
34.2.27	Enhancer Sharpness Register 2 (ADJ1_ENH_SHP2).....	34-34
34.2.28	Enhancer Sharpness Register 3 (ADJ1_ENH_SHP3).....	34-35
34.2.29	Enhancer Sharpness Register 4 (ADJ1_ENH_SHP4).....	34-36
34.2.30	Enhancer Sharpness Register 5 (ADJ1_ENH_SHP5).....	34-36
34.2.31	Enhancer Sharpness Register 6 (ADJ1_ENH_SHP6).....	34-37
34.2.32	Enhancer LTI Register 1 (ADJ1_ENH_LTI1).....	34-38
34.2.33	Enhancer LTI Register 2 (ADJ1_ENH_LTI2).....	34-39
34.2.34	Matrix Mode Register in Image Quality Improver (ADJ1_MTX_MODE).....	34-40
34.2.35	Matrix YG Control Register 0 in Image Quality Improver (ADJ1_MTX_YG_ADJ0).....	34-40
34.2.36	Matrix YG Control Register 1 in Image Quality Improver (ADJ1_MTX_YG_ADJ1).....	34-41
34.2.37	Matrix CBB Control Register 0 in Image Quality Improver (ADJ1_MTX_CBB_ADJ0).....	34-42
34.2.38	Matrix CBB Control Register 1 in Image Quality Improver (ADJ1_MTX_CBB_ADJ1).....	34-43
34.2.39	Matrix CRR Control Register 0 in Image Quality Improver (ADJ1_MTX_CRR_ADJ0).....	34-44
34.2.40	Matrix CRR Control Register 1 in Image Quality Improver (ADJ1_MTX_CRR_ADJ1).....	34-45
34.3	Usage Method.....	34-46
34.3.1	Black Stretch Usage Method.....	34-46
34.3.2	LTI Processing of Enhancer.....	34-46
34.3.3	Sharpness Processing of Enhancer.....	34-47
34.3.4	Setting Method for Color Matrix Data Conversion.....	34-48
35.	Video Display Controller 5 (5): Image Synthesizer.....	35-1
35.1	Image Synthesizer.....	35-1
35.1.1	Overview of Functions.....	35-1
35.1.2	Graphics Data Read Control.....	35-3
35.1.3	Setting Graphics Display Area.....	35-13
35.1.4	Interrupt Generation at Specified Line.....	35-14
35.1.5	Formats of Frame Buffer Read Signals and Corresponding Alpha Blending Types.....	35-14
35.1.6	Display Selection.....	35-15



35.1.7	Background Color Display Processing.....	35-17
35.1.8	Lower-Layer Graphics Display Processing.....	35-17
35.1.9	Current Graphics Display Processing.....	35-17
35.1.10	Display with Alpha Blending in a Rectangular Area .....	35-18
35.1.11	RGB-Index Chroma-Key Processing .....	35-20
35.1.12	CLUT-Index Chroma-Key Processing.....	35-21
35.1.13	Display with Alpha Blending in One-Pixel Units .....	35-22
35.1.14	Alpha Blending Calculation .....	35-23
35.1.15	CLUT Table.....	35-23
35.1.16	Multiplication Processing with Current Alpha at Alpha Blending in Rectangular Area .....	35-24
35.1.17	Selection of Lower-Layer/Current Graphics in VIN Synthesizer .....	35-24
35.2	Register Descriptions.....	35-25
35.2.1	Graphics 2 Register Update Control Register (GR2_UPDATE) .....	35-32
35.2.2	Frame Buffer Read Control Register (Graphics 2) (GR2_FLM_RD).....	35-32
35.2.3	Frame Buffer Control Register 1 (Graphics 2) (GR2_FLM1) .....	35-33
35.2.4	Frame Buffer Control Register 2 (Graphics 2) (GR2_FLM2) .....	35-34
35.2.5	Frame Buffer Control Register 3 (Graphics 2) (GR2_FLM3) .....	35-34
35.2.6	Frame Buffer Control Register 4 (Graphics 2) (GR2_FLM4) .....	35-35
35.2.7	Frame Buffer Control Register 5 (Graphics 2) (GR2_FLM5) .....	35-35
35.2.8	Frame Buffer Control Register 6 (Graphics 2) (GR2_FLM6) .....	35-36
35.2.9	Alpha Blending Control Register 1 (Graphics 2) (GR2_AB1) .....	35-37
35.2.10	Alpha Blending Control Register 2 (Graphics 2) (GR2_AB2) .....	35-38
35.2.11	Alpha Blending Control Register 3 (Graphics 2) (GR2_AB3) .....	35-39
35.2.12	Alpha Blending Control Register 4 (Graphics 2) (GR2_AB4) .....	35-39
35.2.13	Alpha Blending Control Register 5 (Graphics 2) (GR2_AB5) .....	35-40
35.2.14	Alpha Blending Control Register 6 (Graphics 2) (GR2_AB6) .....	35-40
35.2.15	Alpha Blending Control Register 7 (Graphics 2) (GR2_AB7) .....	35-41
35.2.16	Alpha Blending Control Register 8 (Graphics 2) (GR2_AB8) .....	35-41
35.2.17	Alpha Blending Control Register 9 (Graphics 2) (GR2_AB9) .....	35-42
35.2.18	Alpha Blending Control Register 10 (Graphics 2) (GR2_AB10) .....	35-42
35.2.19	Alpha Blending Control Register 11 (Graphics 2) (GR2_AB11) .....	35-43
35.2.20	Background Color Control Register (Graphics 2) (GR2_BASE) .....	35-43
35.2.21	CLUT Table Control Register (Graphics 2) (GR2_CLUT) .....	35-44
35.2.22	Status Monitor Register (Graphics 2) (GR2_MON) .....	35-44
35.2.23	Graphics 3 Register Update Control Register (GR3_UPDATE) .....	35-45
35.2.24	Frame Buffer Read Control Register (Graphics 3) (GR3_FLM_RD).....	35-45
35.2.25	Frame Buffer Control Register 1 (Graphics 3) (GR3_FLM1) .....	35-46
35.2.26	Frame Buffer Control Register 2 (Graphics 3) (GR3_FLM2) .....	35-47
35.2.27	Frame Buffer Control Register 3 (Graphics 3) (GR3_FLM3) .....	35-47
35.2.28	Frame Buffer Control Register 4 (Graphics 3) (GR3_FLM4) .....	35-48

35.2.29	Frame Buffer Control Register 5 (Graphics 3) (GR3_FLM5) .....	35-48
35.2.30	Frame Buffer Control Register 6 (Graphics 3) (GR3_FLM6) .....	35-49
35.2.31	Alpha Blending Control Register 1 (Graphics 3) (GR3_AB1) .....	35-50
35.2.32	Alpha Blending Control Register 2 (Graphics 3) (GR3_AB2) .....	35-51
35.2.33	Alpha Blending Control Register 3 (Graphics 3) (GR3_AB3) .....	35-51
35.2.34	Alpha Blending Control Register 4 (Graphics 3) (GR3_AB4) .....	35-52
35.2.35	Alpha Blending Control Register 5 (Graphics 3) (GR3_AB5) .....	35-52
35.2.36	Alpha Blending Control Register 6 (Graphics 3) (GR3_AB6) .....	35-53
35.2.37	Alpha Blending Control Register 7 (Graphics 3) (GR3_AB7) .....	35-53
35.2.38	Alpha Blending Control Register 8 (Graphics 3) (GR3_AB8) .....	35-54
35.2.39	Alpha Blending Control Register 9 (Graphics 3) (GR3_AB9) .....	35-54
35.2.40	Alpha Blending Control Register 10 (Graphics 3) (GR3_AB10) .....	35-55
35.2.41	Alpha Blending Control Register 11 (Graphics 3) (GR3_AB11) .....	35-55
35.2.42	Background Color Control Register (Graphics 3) (GR3_BASE) .....	35-56
35.2.43	CLUT Table and Interrupt Control Register (Graphics 3) (GR3_CLUT_INT) .....	35-56
35.2.44	Status Monitor Register (Graphics 3) (GR3_MON) .....	35-57
35.2.45	VIN Synthesizer Register Update Control Register (GR_VIN_UPDATE) .....	35-58
35.2.46	Alpha Blending Control Register 1 (VIN Synthesizer) (GR_VIN_AB1) .....	35-59
35.2.47	Alpha Blending Control Register 2 (VIN Synthesizer) (GR_VIN_AB2) .....	35-60
35.2.48	Alpha Blending Control Register 3 (VIN Synthesizer) (GR_VIN_AB3) .....	35-60
35.2.49	Alpha Blending Control Register 4 (VIN Synthesizer) (GR_VIN_AB4) .....	35-61
35.2.50	Alpha Blending Control Register 5 (VIN Synthesizer) (GR_VIN_AB5) .....	35-61
35.2.51	Alpha Blending Control Register 6 (VIN Synthesizer) (GR_VIN_AB6) .....	35-62
35.2.52	Alpha Blending Control Register 7 (VIN Synthesizer) (GR_VIN_AB7) .....	35-62
35.2.53	Background Color Control Register (VIN Synthesizer) (GR_VIN_BASE) .....	35-63
35.2.54	Status Monitor Register (VIN Synthesizer) (GR_VIN_MON) .....	35-63
35.3	Usage Method .....	35-64
35.3.1	Mute Image .....	35-64
35.3.2	Alpha Blending in Rectangular Area .....	35-64
36.	Video Display Controller 5 (6): Output Image Generator .....	36-1
36.1	Output Image Generation Functions .....	36-1
36.1.1	Overview of Functions .....	36-1
36.1.2	Register Control .....	36-3
36.1.3	Enabling or Disabling Output Image Generator .....	36-4
36.1.4	Synchronization Control .....	36-4
36.1.5	Setting Angle of View .....	36-6
36.1.6	Scaling Settings .....	36-9
36.1.7	Screen Synthesis .....	36-9
36.1.8	Selecting Format for Writing Video Image Signals to Frame Buffer .....	36-10
36.1.9	Writing to Frame Buffer .....	36-11

36.1.10	Selecting an Input Video Image Signal Processing or Graphics (OIR) Processing .....	36-15
36.2	Register Descriptions.....	36-16
36.2.1	SCL0 Register Update Control Register (OIR_SCL0_UPDATE).....	36-20
36.2.2	Mask Control Register (OIR_SCL0_FRC1) .....	36-21
36.2.3	Missing Vsync Compensation Control Register (OIR_SCL0_FRC2).....	36-21
36.2.4	Output Sync Select Register (OIR_SCL0_FRC3).....	36-22
36.2.5	Free-Running Period Control Register (OIR_SCL0_FRC4).....	36-22
36.2.6	Output Delay Control Register (OIR_SCL0_FRC5).....	36-23
36.2.7	Full-Screen Vertical Size Register (OIR_SCL0_FRC6).....	36-24
36.2.8	Full-Screen Horizontal Size Register (OIR_SCL0_FRC7).....	36-25
36.2.9	Scaling-Down Control Register (OIR_SCL0_DS1).....	36-26
36.2.10	Vertical Capture Size Register (OIR_SCL0_DS2) .....	36-27
36.2.11	Horizontal Capture Size Register (OIR_SCL0_DS3) .....	36-28
36.2.12	Scaling-Down Control Block Output Size Register (OIR_SCL0_DS7).....	36-29
36.2.13	Scaling-Up Control Register (OIR_SCL0_US1) .....	36-30
36.2.14	Output Image Vertical Size Register (OIR_SCL0_US2).....	36-31
36.2.15	Output Image Horizontal Size Register (OIR_SCL0_US3).....	36-32
36.2.16	Frame Buffer Read Select Register (OIR_SCL0_US8) .....	36-33
36.2.17	Background Color Register (OIR_SCL0_OVR1).....	36-34
36.2.18	SCL1 Register Update Control Register (OIR_SCL1_UPDATE).....	36-35
36.2.19	Writing Mode Register (OIR_SCL1_WR1).....	36-36
36.2.20	Write Address Register 1 (OIR_SCL1_WR2) .....	36-37
36.2.21	Write Address Register 2 (OIR_SCL1_WR3) .....	36-38
36.2.22	Write Address Register 3 (OIR_SCL1_WR4) .....	36-39
36.2.23	Frame Sub-Sampling Register (OIR_SCL1_WR5) .....	36-40
36.2.24	Bit Reduction Register (OIR_SCL1_WR6) .....	36-41
36.2.25	Write Detection Register (OIR_SCL1_WR7).....	36-41
36.2.26	Graphics (OIR) Register Update Control Register (GR_OIR_UPDATE).....	36-42
36.2.27	Frame Buffer Read Control Register (Graphics (OIR)) (GR_OIR_FLM_RD) .....	36-42
36.2.28	Frame Buffer Control Register 1 (Graphics (OIR)) (GR_OIR_FLM1) .....	36-43
36.2.29	Frame Buffer Control Register 2 (Graphics (OIR)) (GR_OIR_FLM2) .....	36-44
36.2.30	Frame Buffer Control Register 3 (Graphics (OIR)) (GR_OIR_FLM3) .....	36-45
36.2.31	Frame Buffer Control Register 4 (Graphics (OIR)) (GR_OIR_FLM4) .....	36-46
36.2.32	Frame Buffer Control Register 5 (Graphics (OIR)) (GR_OIR_FLM5).....	36-47
36.2.33	Frame Buffer Control Register 6 (Graphics (OIR)) (GR_OIR_FLM6).....	36-48
36.2.34	Alpha Blending Control Register 1 (Graphics (OIR)) (GR_OIR_AB1).....	36-50
36.2.35	Alpha Blending Control Register 2 (Graphics (OIR)) (GR_OIR_AB2).....	36-51
36.2.36	Alpha Blending Control Register 3 (Graphics (OIR)) (GR_OIR_AB3).....	36-52
36.2.37	Alpha Blending Control Register 7 (Graphics (OIR)) (GR_OIR_AB7).....	36-53
36.2.38	Alpha Blending Control Register 8 (Graphics (OIR)) (GR_OIR_AB8).....	36-53

36.2.39	Alpha Blending Control Register 9 (Graphics (OIR)) (GR_OIR_AB9).....	36-54
36.2.40	Alpha Blending Control Register 10 (Graphics (OIR)) (GR_OIR_AB10).....	36-55
36.2.41	Alpha Blending Control Register 11 (Graphics (OIR)) (GR_OIR_AB11).....	36-56
36.2.42	Background Color Control Register (Graphics (OIR)) (GR_OIR_BASE).....	36-57
36.2.43	CLUT Table Control Register (Graphics (OIR)) (GR_OIR_CLUT).....	36-58
36.2.44	Status Monitor Register (GR_OIR_MON).....	36-59
<b>37.</b>	<b>Video Display Controller 5 (7): Output Controller.....</b>	<b>37-1</b>
37.1	Output Controller.....	37-1
37.1.1	Overview of Functions .....	37-1
37.1.2	Register Update Control .....	37-2
37.1.3	Route Selection.....	37-2
37.1.4	Panel Brightness Adjustment.....	37-3
37.1.5	Contrast Adjustment.....	37-3
37.1.6	Gamma Correction .....	37-4
37.1.7	Dither Process.....	37-8
37.1.8	Output Format Conversion .....	37-10
37.1.9	LCD TCON .....	37-16
37.2	Register Descriptions.....	37-26
37.2.1	Register Update Control Register G in Gamma Correction Block (GAM_G_UPDATE).....	37-34
37.2.2	Function Switch Register in Gamma Correction Block (GAM_SW).....	37-35
37.2.3	Table Setting Register G1 to G16 in Gamma Correction Block (GAM_G_LUT1 to GAM_G_LUT16) .....	37-36
37.2.4	Area Setting Register G1 in Gamma Correction Block (GAM_G_AREA1).....	37-38
37.2.5	Area Setting Register G2 in Gamma Correction Block (GAM_G_AREA2).....	37-39
37.2.6	Area Setting Register G3 in Gamma Correction Block (GAM_G_AREA3).....	37-40
37.2.7	Area Setting Register G4 in Gamma Correction Block (GAM_G_AREA4).....	37-41
37.2.8	Area Setting Register G5 in Gamma Correction Block (GAM_G_AREA5).....	37-42
37.2.9	Area Setting Register G6 in Gamma Correction Block (GAM_G_AREA6).....	37-43
37.2.10	Area Setting Register G7 in Gamma Correction Block (GAM_G_AREA7).....	37-44
37.2.11	Area Setting Register G8 in Gamma Correction Block (GAM_G_AREA8).....	37-45
37.2.12	Register Update Control Register B in Gamma Correction Block (GAM_B_UPDATE) .....	37-46
37.2.13	Table Setting Register B1 to B16 in Gamma Correction Block (GAM_B_LUT1 to GAM_B_LUT16).....	37-47
37.2.14	Area Setting Register B1 in Gamma Correction Block (GAM_B_AREA1).....	37-49
37.2.15	Area Setting Register B2 in Gamma Correction Block (GAM_B_AREA2) .....	37-50
37.2.16	Area Setting Register B3 in Gamma Correction Block (GAM_B_AREA3) .....	37-51
37.2.17	Area Setting Register B4 in Gamma Correction Block (GAM_B_AREA4) .....	37-52
37.2.18	Area Setting Register B5 in Gamma Correction Block (GAM_B_AREA5).....	37-53
37.2.19	Area Setting Register B6 in Gamma Correction Block (GAM_B_AREA6).....	37-54
37.2.20	Area Setting Register B7 in Gamma Correction Block (GAM_B_AREA7).....	37-55
37.2.21	Area Setting Register B8 in Gamma Correction Block (GAM_B_AREA8).....	37-56

37.2.22	Register Update Control Register R in Gamma Correction Block (GAM_R_UPDATE) .....	37-57
37.2.23	Table Setting Register R1 to R16 in Gamma Correction Block (GAM_R_LUT1 to GAM_R_LUT16).....	37-58
37.2.24	Area Setting Register R1 in Gamma Correction Block (GAM_R_AREA1) .....	37-60
37.2.25	Area Setting Register R2 in Gamma Correction Block (GAM_R_AREA2) .....	37-61
37.2.26	Area Setting Register R3 in Gamma Correction Block (GAM_R_AREA3) .....	37-62
37.2.27	Area Setting Register R4 in Gamma Correction Block (GAM_R_AREA4) .....	37-63
37.2.28	Area Setting Register R5 in Gamma Correction Block (GAM_R_AREA5) .....	37-64
37.2.29	Area Setting Register R6 in Gamma Correction Block (GAM_R_AREA6) .....	37-65
37.2.30	Area Setting Register R7 in Gamma Correction Block (GAM_R_AREA7) .....	37-66
37.2.31	Area Setting Register R8 in Gamma Correction Block (GAM_R_AREA8) .....	37-67
37.2.32	TCON Register Update Control Register (TCON_UPDATE).....	37-68
37.2.33	TCON Reference Timing Setting Register (TCON_TIM).....	37-68
37.2.34	TCON Vertical Timing Setting Register A1 (TCON_TIM_STVA1).....	37-69
37.2.35	TCON Vertical Timing Setting Register A2 (TCON_TIM_STVA2).....	37-70
37.2.36	TCON Vertical Timing Setting Register B1 (TCON_TIM_STVB1) .....	37-71
37.2.37	TCON Vertical Timing Setting Register B2 (TCON_TIM_STVB2) .....	37-72
37.2.38	TCON Horizontal Timing Setting Register STH1 (TCON_TIM_STH1).....	37-73
37.2.39	TCON Horizontal Timing Setting Register STH2 (TCON_TIM_STH2).....	37-74
37.2.40	TCON Horizontal Timing Setting Register STB1 (TCON_TIM_STB1) .....	37-75
37.2.41	TCON Horizontal Timing Setting Register STB2 (TCON_TIM_STB2) .....	37-76
37.2.42	TCON Horizontal Timing Setting Register CPV1 (TCON_TIM_CPV1) .....	37-77
37.2.43	TCON Horizontal Timing Setting Register CPV2 (TCON_TIM_CPV2) .....	37-78
37.2.44	TCON Horizontal Timing Setting Register POLA1 (TCON_TIM_POLA1).....	37-79
37.2.45	TCON Horizontal Timing Setting Register POLA2 (TCON_TIM_POLA2).....	37-80
37.2.46	TCON Horizontal Timing Setting Register POLB1 (TCON_TIM_POLB1) .....	37-81
37.2.47	TCON Horizontal Timing Setting Register POLB2 (TCON_TIM_POLB2) .....	37-82
37.2.48	TCON Data Enable Polarity Setting Register (TCON_TIM_DE).....	37-83
37.2.49	Register Update Control Register in Output Controller (OUT_UPDATE) .....	37-83
37.2.50	Output Interface Register (OUT_SET).....	37-84
37.2.51	Brightness (DC) Correction Register 1 (OUT_BRIGHT1).....	37-85
37.2.52	Brightness (DC) Correction Register 2 (OUT_BRIGHT2).....	37-86
37.2.53	Contrast (Gain) Correction Register (OUT_CONTRAST).....	37-86
37.2.54	Panel Dither Register (OUT_PDTHA).....	37-87
37.2.55	Output Phase Control Register (OUT_CLK_PHASE).....	37-88
37.3	Usage Methods .....	37-89
37.3.1	Gamma Correction Adjustment Method .....	37-89
37.3.2	Dither Usage Method.....	37-89
37.3.3	Output Format Adjustment Method .....	37-90

38.	Video Display Controller 5 (8):System Controller .....	38-1
38.1	System Controller .....	38-1
38.1.1	Overview of Functions .....	38-1
38.1.2	Interrupt Control .....	38-1
38.1.3	Panel Clock Control.....	38-5
38.1.4	CLUT Table Read Select Signal Status Flag.....	38-6
38.2	Register Descriptions.....	38-7
38.2.1	Interrupt Control Register 1 (SYSCNT_INT1) .....	38-8
38.2.2	Interrupt Control Register 2 (SYSCNT_INT2) .....	38-10
38.2.3	Interrupt Control Register 3 (SYSCNT_INT3).....	38-12
38.2.4	Interrupt Control Register 4 (SYSCNT_INT4).....	38-14
38.2.5	Interrupt Control Register 5 (SYSCNT_INT5).....	38-15
38.2.6	Interrupt Control Register 6 (SYSCNT_INT6).....	38-16
38.2.7	Panel Clock Control Register (SYSCNT_PANEL_CLK) .....	38-17
38.2.8	CLUT Table Read Select Signal Status Register (SYSCNT_CLUT).....	38-18
39.	Dynamic Range Compression .....	39-1
40.	LVDS Output Interface .....	40-1
40.1	Features.....	40-1
40.2	Block Diagram.....	40-1
40.3	Input/Output Pins.....	40-2
40.4	Register Descriptions.....	40-3
40.4.1	LVDS Register Update Control Register (LVDS_UPDATE) .....	40-3
40.4.2	LVDS Format Conversion Register L (LVDSFCL) .....	40-4
40.4.3	LVDS Clock Select Register (LCLKSELR) .....	40-5
40.4.4	LVDS PLL Setting Register (LPLLSETR) .....	40-6
40.4.5	LVDS PHY Characteristics Switching Register (LPHYACC).....	40-7
40.5	Operation .....	40-8
40.5.1	LVDS PLL Settings.....	40-8
40.5.2	LVDS Output Format .....	40-10
40.5.3	Procedures for Register Settings.....	40-12
40.6	Notes.....	40-13
40.6.1	LVDS Output Pin Settings .....	40-13
41.	Image Renderer (IMR-LS2) .....	41-1
42.	Image Renderer for Display (IMR-LSD).....	42-1
43.	Display Out Comparison Unit.....	43-1
43.1	Features.....	43-1
43.2	Block Diagram.....	43-2
43.3	Register Descriptions.....	43-3
43.3.1	Control Register (DOCMCR).....	43-4
43.3.2	Status Register (DOCMSTR).....	43-4

43.3.3	Status Clear Register (DOCMCLSTR) .....	43-5
43.3.4	Interrupt Enable Register (DOCMIENR).....	43-5
43.3.5	Operation Parameter Setting Register (DOCMPPMR).....	43-6
43.3.6	Expected CRC Code Register (DOCMECRCR).....	43-6
43.3.7	Calculated CRC Code Value Register (DOCMCCRCR).....	43-7
43.3.8	Horizontal Start Position Setting Register (DOCMSPXR).....	43-7
43.3.9	Vertical Start Position Setting Register (DOCMSPYR) .....	43-8
43.3.10	Horizontal Size Setting Register (DOCMSZXR).....	43-8
43.3.11	Vertical Size Setting Register (DOCMSZYR).....	43-9
43.3.12	CRC Code Initialization Register (DOCMCRCIR) .....	43-9
43.4	Operation .....	43-10
43.4.1	Overview of Operations.....	43-10
43.4.2	System Configuration .....	43-10
43.4.3	CRC Calculation Method .....	43-11
43.4.4	Graphics Data Selection for CRC Code Generation .....	43-11
43.4.5	Pixel Format .....	43-11
43.4.6	Rectangular Area Settings .....	43-11
43.4.7	CRC Calculation Time Period and Comparison Timing.....	43-13
43.4.8	Register Update Timing.....	43-14
43.4.9	Operation Flow .....	43-15
43.5	Interrupt .....	43-17
43.6	Usage Note .....	43-17
43.6.1	Expected CRC Value.....	43-17
43.6.2	Expansion Control Functionality.....	43-17
44.	Renesas Graphics Processor for OpenVG™ .....	44-1
44.1	Specification .....	44-1
45.	JPEG Codec Unit.....	45-1
45.1	Features.....	45-1
45.2	Register Descriptions.....	45-3
45.2.1	JPEG Code Mode Register (JCMOD).....	45-4
45.2.2	JPEG Code Command Register (JCCMD).....	45-5
45.2.3	JPEG Code Quantization Table Number Register (JCQTN) .....	45-6
45.2.4	JPEG Code Huffman Table Number Register (JCHTN).....	45-7
45.2.5	JPEG Code DRI Upper Register (JCDRIU).....	45-7
45.2.6	JPEG Code DRI Lower Register (JCDRID) .....	45-8
45.2.7	JPEG Code Vertical Size Upper Register (JCVSZU) .....	45-8
45.2.8	JPEG Code Vertical Size Lower Register (JCVSZD).....	45-8
45.2.9	JPEG Code Horizontal Size Upper Register (JCHSZU) .....	45-9
45.2.10	JPEG Coded Horizontal Size Lower Register (JCHSZD).....	45-9
45.2.11	JPEG Code Data Count Upper Register (JCDTCU) .....	45-9

45.2.12	JPEG Code Data Count Middle Register (JCDTCM).....	45-10
45.2.13	JPEG Code Data Count Lower Register (JCDTCD).....	45-10
45.2.14	JPEG Interrupt Enable Register 0 (JINTE0).....	45-11
45.2.15	JPEG Interrupt Status Register 0 (JINTS0).....	45-12
45.2.16	JPEG Code Decode Error Register (JCDERR).....	45-12
45.2.17	JPEG Code Reset Register (JCRST).....	45-13
45.2.18	JPEG Interface Compression Control Register (JIFECNT).....	45-14
45.2.19	JPEG Interface Compression Source Address Register (JIFESA).....	45-15
45.2.20	JPEG Interface Compression Line Offset Register (JIFESOFST).....	45-16
45.2.21	JPEG Interface Compression Destination Address Register (JIFEDA).....	45-16
45.2.22	JPEG Interface Compression Source Line Count Register (JIFESLC).....	45-17
45.2.23	JPEG Interface Compression Destination Count Register (JIFEDDC).....	45-17
45.2.24	JPEG Interface Decompression Control Register (JIFDCNT).....	45-18
45.2.25	JPEG Interface Decompression Source Address Register (JIFDSA).....	45-20
45.2.26	JPEG Interface Decompression Line Offset Register (JIFDDOFST).....	45-20
45.2.27	JPEG Interface Decompression Destination Address Register (JIFDDA).....	45-21
45.2.28	JPEG Interface Decompression Source Data Count Register (JIFSDC).....	45-22
45.2.29	JPEG Interface Decompression Destination Line Count Register (JIFDDL).....	45-22
45.2.30	JPEG Interface Decompression $\alpha$ Set Register (JIFDADT).....	45-23
45.2.31	JPEG Interrupt Enable Register 1 (JINTE1).....	45-24
45.2.32	JPEG Interrupt Status Register 1 (JINTS1).....	45-25
45.2.33	JPEG Input Image Data CbCr Range Setting Register (JIFESVSZ).....	45-26
45.2.34	JPEG Output Image Data CbCr Range Setting Register (JIFESHSZ).....	45-26
45.3	Operation .....	45-27
45.3.1	Compression .....	45-27
45.3.2	Decompression .....	45-33
45.3.3	Output Pixel Format in Decompression .....	45-39
45.3.4	Storing Image Data.....	45-43
45.4	Interrupts.....	45-44
45.4.1	Compression/Decompression Process Interrupt Request (JEDI).....	45-44
45.4.2	Data Transfer Interrupt Request (JDTI).....	45-44
45.5	Bus Reset Processing.....	45-46
46.	Capture Engine Unit.....	46-1
46.1	Features of CEU .....	46-1
46.2	Functional Overview of CEU.....	46-2
46.3	Pin Configuration of CEU .....	46-3
46.4	Register Descriptions of CEU .....	46-4
46.4.1	Capture Start Register (CAPSR) .....	46-5
46.4.2	Capture Control Register (CAPCR).....	46-9
46.4.3	Capture Interface Control Register (CAMCR).....	46-12



46.4.4	Capture Interface Cycle Register (CMCYR).....	46-16
46.4.5	Capture Interface Offset Register (CAMOR).....	46-17
46.4.6	Capture Interface Width Register (CAPWR).....	46-19
46.4.7	Capture Interface Input Format Register (CAIFR).....	46-21
46.4.8	CEU Register Control Register (CRCNTR).....	46-25
46.4.9	CEU Register Forcible Control Register (CRCMPR).....	46-26
46.4.10	Capture Filter Control Register (CFLCR).....	46-27
46.4.11	Capture Filter Size Clip Register (CFSZR).....	46-29
46.4.12	Capture Destination Width Register (CDWDR).....	46-31
46.4.13	Capture Data Address Y Register (CDAYR).....	46-32
46.4.14	Capture Data Address C Register (CDACR).....	46-33
46.4.15	Capture Data Bottom-Field Address Y Register (CDBYR).....	46-35
46.4.16	Capture Data Bottom-Field Address C Register (CDBCR).....	46-36
46.4.17	Capture Bundle Destination Size Register (CBDSR).....	46-37
46.4.18	Capture Low-Pass Filter Control Register (CLFCR).....	46-38
46.4.19	Firewall Operation Control Register (CFWCR).....	46-39
46.4.20	Capture Data Output Control Register (CDOCR).....	46-39
46.4.21	Capture Event Interrupt Enable Register (CEIER).....	46-43
46.4.22	Capture Event Flag Clear Register (CETCR).....	46-45
46.4.23	Capture Status Register (CSTSR).....	46-49
46.4.24	Capture Data Size Register (CDSSR).....	46-51
46.4.25	Capture Data Address Y Register 2 (CDAYR2).....	46-52
46.4.26	Capture Data Address C Register 2 (CDACR2).....	46-54
46.4.27	Capture Data Bottom-Field Address Y Register 2 (CDBYR2).....	46-56
46.4.28	Capture Data Bottom-Field Address C Register 2 (CDBCR2).....	46-57
46.5	Usage Notes for CEU.....	46-58
46.5.1	Conditions for Connection to an External Module.....	46-58
46.5.2	Restrictions on Input/Output Functions.....	46-59
46.5.3	Cooperation with Video Display Controller 5.....	46-59
47.	Pixel Format Converter.....	47-1
47.1	Features.....	47-1
47.2	Register Descriptions.....	47-2
47.2.1	PFV Control Register (PFVCR).....	47-3
47.2.2	PFV Interrupt Control Register (PFVICR).....	47-4
47.2.3	PFV Interrupt Status Register (PFVISR).....	47-5
47.2.4	PFV Input Buffer Register (PFVID).....	47-6
47.2.5	PFV Output Buffer Register (PFVOD).....	47-6
47.2.6	PFV Input FIFO Status Register (PFVIFSR).....	47-7
47.2.7	PFV Output FIFO Status Register (PFVOFSR).....	47-7
47.2.8	PFV $\alpha$ Setting Register (PFVACR).....	47-8

47.2.9	PFV Matrix Mode Register (PFV_MTX_MODE).....	47-8
47.2.10	PFV Matrix YG Adjustment Register 0 (PFV_MTX_YG_ADJ0) .....	47-9
47.2.11	PFV Matrix YG Adjustment Register 1 (PFV_MTX_YG_ADJ1) .....	47-9
47.2.12	PFV Matrix CBB Adjustment Register 0 (PFV_MTX_CBB_ADJ0).....	47-10
47.2.13	PFV Matrix CBB Adjustment Register 1 (PFV_MTX_CBB_ADJ1).....	47-10
47.2.14	PFV Matrix CRR Adjustment Register 0 (PFV_MTX_CRR_ADJ0).....	47-11
47.2.15	PFV Matrix CRR Adjustment Register 1 (PFV_MTX_CRR_ADJ1).....	47-11
47.2.16	PFV Image Size Setting Register (PFVSZR) .....	47-12
47.3	Operation .....	47-13
47.3.1	Overview .....	47-13
47.3.2	Input/Output Data Format .....	47-13
47.3.3	Color Matrix .....	47-14
47.3.4	Bit Reduction.....	47-16
47.3.5	Operating Procedures .....	47-17
47.4	Interrupt Requests.....	47-19
48.	SCUX.....	48-1
48.1	Features.....	48-1
48.2	Input/Output Pins.....	48-3
48.3	Register Descriptions.....	48-4
48.3.1	IPC0_n Initialization Register (IPCIR_IPC0_n) (n = 0, 1, 2, 3) .....	48-11
48.3.2	IPC0_n Pass Select Register (IPSLR_IPC0_n) (n = 0, 1, 2, 3) .....	48-11
48.3.3	OPC0_n Initialization Register (OPCIR_OPC0_n) (n = 0, 1, 2, 3) .....	48-12
48.3.4	OPC0_n Pass Select Register (OPSLR_OPC0_n) (n = 0, 1, 2, 3) .....	48-12
48.3.5	FFD0_n FIFO Download Initialization Register (FFDIR_FFD0_n) (n = 0, 1, 2, 3) .....	48-13
48.3.6	FFD0_n FIFO Download Audio Information Register (FDAIR_FFD0_n) (n = 0, 1, 2, 3) ...	48-13
48.3.7	FFD0_n FIFO Download Request Size Register (DRQSR_FFD0_n) (n = 0, 1, 2, 3) .....	48-14
48.3.8	FFD0_n FIFO Download Pass Register (FFDPR_FFD0_n) (n = 0, 1, 2, 3).....	48-15
48.3.9	FFD0_n FIFO Download Boot Register (FFDBR_FFD0_n) (n = 0, 1, 2, 3).....	48-15
48.3.10	FFD0_n FIFO Download Event Mask Register (DEVMR_FFD0_n) (n = 0, 1, 2, 3).....	48-16
48.3.11	FFD0_n FIFO Download Event Clear Register (DEVCR_FFD0_n) (n = 0, 1, 2, 3).....	48-17
48.3.12	FFU0_n FIFO Upload Initialization Register (FFUIR_FFU0_n) (n = 0, 1, 2, 3) .....	48-18
48.3.13	FFU0_n FIFO Upload Audio Information Register (FUAIR_FFU0_n) (n = 0, 1, 2, 3).....	48-18
48.3.14	FFU0_n FIFO Upload Request Size Register (URQSR_FFU0_n) (n = 0, 1, 2, 3) .....	48-19
48.3.15	FFU0_n FIFO Upload Pass Register (FFUPR_FFU0_n) (n = 0, 1, 2, 3).....	48-20
48.3.16	FFU0_n FIFO Upload Event Mask Register (UEVMR_FFU0_n) (n = 0, 1, 2, 3) .....	48-21
48.3.17	FFU0_n FIFO Upload Event Clear Register (UEVCR_FFU0_n) (n = 0, 1, 2, 3).....	48-22
48.3.18	2SRC0_m Initialization Register p (SRCIRp_2SRC0_m) (m = 0, 1; p = 0, 1) .....	48-23
48.3.19	2SRC0_m Audio Information Register p (SADIRp_2SRC0_m) (m = 0, 1; p = 0, 1) .....	48-24
48.3.20	2SRC0_m Bypass Register p (SRCBRp_2SRC0_m) (m = 0, 1; p = 0, 1).....	48-25
48.3.21	2SRC0_m IFS Control Register p (IFSCRp_2SRC0_m) (m = 0, 1; p = 0, 1) .....	48-25

48.3.22	2SRC0_m IFS Value Setting Register p (IFSVRp_2SRC0_m) (m = 0, 1; p = 0, 1) .....	48-26
48.3.23	2SRC0_m Control Register p (SRCCRp_2SRC0_m) (m = 0, 1; p = 0, 1) .....	48-31
48.3.24	2SRC0_m Minimum FS Setting Register p (MNFSRp_2SRC0_m) (m = 0, 1; p = 0, 1) .....	48-32
48.3.25	2SRC0_m Buffer Size Setting Register p (BFSSRp_2SRC0_m) (m = 0, 1; p = 0, 1) .....	48-32
48.3.26	2SRC0_m SCU2 Status Register p (SC2SRp_2SRC0_m) (m = 0, 1; p = 0, 1) .....	48-33
48.3.27	2SRC0_m Wait Time Setting Register p (WATSRp_2SRC0_m) (m = 0, 1; p = 0, 1) .....	48-33
48.3.28	2SRC0_m Event Mask Register p (SEVMRp_2SRC0_m) (m = 0, 1; p = 0, 1) .....	48-34
48.3.29	2SRC0_m Event Clear Register p (SEVCRp_2SRC0_m) (m = 0, 1; p = 0, 1) .....	48-35
48.3.30	2SRC0_m Initialization Register RIF (SRCIRR_2SRC0_m) (m = 0, 1) .....	48-36
48.3.31	DVU0_n Initialization Register (DVUIR_DVU0_n) (n = 0, 1, 2, 3) .....	48-36
48.3.32	DVU0_n Audio Information Register (VADIR_DVU0_n) (n = 0, 1, 2, 3) .....	48-37
48.3.33	DVU0_n Bypass Register (DVUBR_DVU0_n) (n = 0, 1, 2, 3) .....	48-38
48.3.34	DVU0_n Control Register (DVUCR_DVU0_n) (n = 0, 1, 2, 3) .....	48-38
48.3.35	DVU0_n Zero Cross Mute Control Register (ZCMCR_DVU0_n) (n = 0, 1, 2, 3) .....	48-39
48.3.36	DVU0_n Volume Ramp Control Register (VRCTR_DVU0_n) (n = 0, 1, 2, 3) .....	48-40
48.3.37	DVU0_n Volume Ramp Period Register (VRPDR_DVU0_n) (n = 0, 1, 2, 3) .....	48-41
48.3.38	DVU0_n Volume Ramp Decibel Register (VRDBR_DVU0_n) (n = 0, 1, 2, 3) .....	48-42
48.3.39	DVU0_n Volume Ramp Wait Time Register (VRWTR_DVU0_n) (n = 0, 1, 2, 3) .....	48-43
48.3.40	DVU0_n Volume Value Setting 0 Register (VOL0R_DVU0_n) (n = 0, 1, 2, 3) .....	48-43
48.3.41	DVU0_n Volume Value Setting 1 Register (VOL1R_DVU0_n) (n = 0, 1, 2, 3) .....	48-44
48.3.42	DVU0_n Volume Value Setting 2 Register (VOL2R_DVU0_n) (n = 0, 1, 2, 3) .....	48-45
48.3.43	DVU0_n Volume Value Setting 3 Register (VOL3R_DVU0_n) (n = 0, 1, 2, 3) .....	48-46
48.3.44	DVU0_n Volume Value Setting 4 Register (VOL4R_DVU0_n) (n = 0, 1, 2, 3) .....	48-47
48.3.45	DVU0_n Volume Value Setting 5 Register (VOL5R_DVU0_n) (n = 0, 1, 2, 3) .....	48-48
48.3.46	DVU0_n Volume Value Setting 6 Register (VOL6R_DVU0_n) (n = 0, 1, 2, 3) .....	48-49
48.3.47	DVU0_n Volume Value Setting 7 Register (VOL7R_DVU0_n) (n = 0, 1, 2, 3) .....	48-50
48.3.48	DVU0_n Enable Register (DVUER_DVU0_n) (n = 0, 1, 2, 3) .....	48-50
48.3.49	DVU0_n Status Register (DVUSR_DVU0_n) (n = 0, 1, 2, 3) .....	48-51
48.3.50	DVU0_n Event Mask Register (VEVMR_DVU0_n) (n = 0, 1, 2, 3) .....	48-52
48.3.51	DVU0_n Event Clear Register (VEVCR_DVU0_n) (n = 0, 1, 2, 3) .....	48-54
48.3.52	MIX0_0 Initialization Register (MIXIR_MIX0_0) .....	48-55
48.3.53	MIX0_0 Audio Information Register (MADIR_MIX0_0) .....	48-56
48.3.54	MIX0_0 Bypass Register (MIXBR_MIX0_0) .....	48-57
48.3.55	MIX0_0 Mode Register (MIXMR_MIX0_0) .....	48-57
48.3.56	MIX0_0 Volume Period Register (MVPDR_MIX0_0) .....	48-58
48.3.57	MIX0_0 Decibel A Register (MDBAR_MIX0_0) .....	48-59
48.3.58	MIX0_0 Decibel B Register (MDBBR_MIX0_0) .....	48-59
48.3.59	MIX0_0 Decibel C Register (MDBCR_MIX0_0) .....	48-60
48.3.60	MIX0_0 Decibel D Register (MDBDR_MIX0_0) .....	48-61
48.3.61	MIX0_0 Decibel Enable Register (MDBER_MIX0_0) .....	48-61

48.3.62	MIX0_0 Status Register (MIXSR_MIX0_0) .....	48-62
48.3.63	Software Reset Register (SWRSR_CIM).....	48-62
48.3.64	DMA Control Register (DMACR_CIM) .....	48-63
48.3.65	DMA Transfer Register for FFD0_n (DMATDn_CIM) (n = 0, 1, 2, 3) .....	48-64
48.3.66	DMA Transfer Register for FFU0_n (DMATUn_CIM) (n = 0, 1, 2, 3) .....	48-65
48.3.67	SSI Route Select Register (SSIRSEL_CIM).....	48-66
48.3.68	FFD0_n Timing Select Register (FDTSELn_CIM) (n = 0, 1, 2, 3) .....	48-69
48.3.69	FFU0_n Timing Select Register (FDTSELn_CIM) (n = 0, 1, 2, 3) .....	48-71
48.3.70	SSI Pin Mode Register (SSIPMD_CIM).....	48-73
48.3.71	SSI Control Register (SSICTRL_CIM).....	48-76
48.3.72	SRCn Route Select Register (SRCRSELn_CIM) (n = 0, 1, 2, 3) .....	48-78
48.3.73	MIX Route Select Register (MIXRSEL_CIM).....	48-82
48.4	Operation .....	48-86
48.4.1	Initial Setting Procedure .....	48-86
48.4.2	Transfer Start Procedure and Stop Procedure.....	48-86
48.4.3	Data Rearrangement for Each Channel .....	48-91
48.4.4	Pin Connection Specifications of SSIF .....	48-94
48.4.5	Data Transfer Route.....	48-95
48.4.6	Input Timing Signal and Output Timing Signal.....	48-101
48.4.7	2SRC (SRC) Block.....	48-104
48.4.8	DVU Block.....	48-106
48.4.9	MIX Block.....	48-110
49.	Sound Generator .....	49-1
49.1	Features.....	49-1
49.2	Input/Output Pins.....	49-2
49.3	Register Descriptions.....	49-2
49.3.1	Sound Generator Control Register 1 (SGCR1) .....	49-3
49.3.2	Sound Generator Control Status Register (SGCSR) .....	49-4
49.3.3	Sound Generator Control Register 2 (SGCR2) .....	49-4
49.3.4	Sound Generator Loudness Register (SGLR) .....	49-5
49.3.5	Sound Generator Tone Frequency Register (SGTFR) .....	49-5
49.3.6	Sound Generator Reference Frequency Register (SGSFR).....	49-6
49.4	Operation .....	49-7
49.4.1	Base Operation .....	49-7
49.4.2	Tone Frequency Setting.....	49-10
49.4.3	Auto Attenuator Function.....	49-11
49.4.4	Output Waveform.....	49-11
49.5	Interrupt Source .....	49-12
49.6	Usage Note .....	49-13
49.6.1	Module Stop Mode Settings .....	49-13

50.	SD Host Interface.....	50-1
51.	MMC Host Interface.....	51-1
51.1	Features.....	51-1
51.2	Input/Output Pins.....	51-1
51.3	Register Descriptions.....	51-2
51.3.1	Command Setting Register (CE_CMD_SET).....	51-3
51.3.2	Argument Register (CE_ARG).....	51-5
51.3.3	Argument Register for Automatically-Issued CMD12 (CE_ARG_CMD12).....	51-5
51.3.4	Command Control Register (CE_CMD_CTRL).....	51-6
51.3.5	Transfer Block Setting Register (CE_BLOCK_SET).....	51-6
51.3.6	Clock Control Register (CE_CLK_CTRL).....	51-7
51.3.7	Buffer Access Configuration Register (CE_BUF_ACC).....	51-8
51.3.8	Response Registers 3 to 0 (CE_RESP3 to CE_RESP0).....	51-9
51.3.9	Response Register for Automatically-Issued CMD12 (CE_RESP_CMD12).....	51-10
51.3.10	Data Register (CE_DATA).....	51-10
51.3.11	Interrupt Flag Register (CE_INT).....	51-11
51.3.12	Interrupt Enable Register (CE_INT_EN).....	51-15
51.3.13	Status Register 1 (CE_HOST_STS1).....	51-17
51.3.14	Status Register 2 (CE_HOST_STS2).....	51-18
51.3.15	DMA Mode Setting Register (CE_DMA_MODE).....	51-19
51.3.16	Card Detection/Port Control Register (CE_DETECT).....	51-20
51.3.17	Special Mode Setting Register (CE_ADD_MODE).....	51-21
51.3.18	Version Register (CE_VERSION).....	51-21
51.4	Interrupt Requests.....	51-22
51.5	DMA Specifications.....	51-23
51.5.1	DMA for Buffer Writing.....	51-23
51.5.2	DMA for Buffer Reading.....	51-23
51.6	Operation.....	51-24
51.6.1	Command/Response Formats.....	51-24
51.6.2	Data Block Format.....	51-25
51.6.3	Buffer Structure and Buffer Accesses.....	51-26
51.6.4	Automatic CMD12 Issuance.....	51-27
51.6.5	High Priority Interrupt (HPI).....	51-28
51.6.6	Background Operation.....	51-28
51.6.7	Operation in the Case of Error/Timeout.....	51-28
51.7	Examples of Setting.....	51-29
51.7.1	Legends.....	51-29
51.7.2	Command Transmission.....	51-29
51.7.3	Command Transmission → Response Reception.....	51-30
51.7.4	Command Transmission → Response Reception (with Response Busy).....	51-31

51.7.5	Single-Block Read .....	51-33
51.7.6	Multi-Block Read .....	51-34
51.7.7	Multi-Block Read (with Automatic CMD12 Issuance).....	51-35
51.7.8	Single-Block Write .....	51-36
51.7.9	Multi-Block Write .....	51-37
51.7.10	Multi-Block Write (with Automatic CMD12 Issuance).....	51-38
51.7.11	Forcible Termination .....	51-39
51.7.12	Setting Values of CE_CMD_SET .....	51-40
51.8	Usage Note .....	51-42
51.8.1	Card Detection.....	51-42
51.8.2	Multi-Block Transfer.....	51-42
<b>52.</b>	<b>Motor Control PWM Timer .....</b>	<b>52-1</b>
52.1	Features.....	52-1
52.2	Input/Output Pins.....	52-3
52.3	Register Descriptions.....	52-4
52.3.1	PWM Control Register_n (PWCR_n) (n = 1, 2) .....	52-5
52.3.2	PWM Polarity Register_n (PWPR_n) (n = 1, 2) .....	52-6
52.3.3	PWM Counter_n (PWCNT_n) (n = 1, 2) .....	52-6
52.3.4	PWM Cycle Register_n (PWCYR_n) (n = 1, 2) .....	52-6
52.3.5	PWM Duty Registers_nA, nC, nE, nG (PWDTR_nA, PWDTR_nC, PWDTR_nE, PWDTR_nG) (n = 1, 2).....	52-7
52.3.6	PWM Buffer Registers_nA, nC, nE, nG (PWBFR_nA, PWBFR_nC, PWBFR_nE, PWBFR_nG) (n = 1, 2) .....	52-9
52.3.7	PWM Buffer Transfer Control Register (PWBTCR) .....	52-10
52.4	Bus Master Interface.....	52-11
52.4.1	16-Bit Data Registers.....	52-11
52.4.2	8-Bit Data Registers.....	52-11
52.5	Operation .....	52-12
52.5.1	PWM Operation.....	52-12
52.5.2	Buffer Transfer Control .....	52-13
52.6	Usage Note .....	52-14
52.6.1	Conflict between Buffer Register Write and Compare Match .....	52-14
<b>53.</b>	<b>On-Chip RAM.....</b>	<b>53-1</b>
53.1	Features.....	53-1
53.2	Usage Notes .....	53-3
53.2.1	Page Conflict .....	53-3
53.2.2	Data Retention .....	53-3
<b>54.</b>	<b>Ports.....</b>	<b>54-1</b>
54.1	Features.....	54-1
54.1.1	Port group .....	54-1

54.1.2	Port group index n .....	54-1
54.1.3	Base address .....	54-1
54.2	Functional Overview .....	54-2
54.2.1	Mode of Pin Functions .....	54-2
54.2.2	Pin Data Input/Output.....	54-3
54.3	Register Description .....	54-4
54.3.1	Port Register (Pn) .....	54-4
54.3.2	Port Set and Reset Register (PSRn).....	54-5
54.3.3	Port Pin Read Register (PPRn/JPPR0) .....	54-5
54.3.4	Port Mode Register (PMn) .....	54-6
54.3.5	Port Mode Control Register (PMCn/JPMC0) .....	54-6
54.3.6	Port Function Control Register (PFCn).....	54-7
54.3.7	Port Function Control Expansion Register (PFCEn).....	54-8
54.3.8	Port NOT Register (PNOTn).....	54-8
54.3.9	Port Mode Set and Reset Register (PMSRn).....	54-9
54.3.10	Port Mode Control Set and Reset Register (PMCSRn/JPMCSR0).....	54-10
54.3.11	Port Function Control Additional Expansion Register (PFCAEn).....	54-10
54.3.12	Port Input Buffer Control Register (PIBCn).....	54-11
54.3.13	Port Bidirection Control Register (PBDCn/JPBDC0).....	54-11
54.3.14	Port IP Control Register (PIPCn) .....	54-12
54.3.15	Serial Sound Interface Noise Canceler Control Register (SNCR) .....	54-13
54.4	Port Function .....	54-14
54.5	JTAG Port 0 (JP0) .....	54-14
54.6	Port 0 (P0).....	54-15
54.7	Port 1 (P1).....	54-16
54.8	Port 2 (P2).....	54-18
54.9	Port 3 (P3).....	54-19
54.10	Port 4 (P4).....	54-20
54.11	Port 5 (P5).....	54-21
54.12	Port 6 (P6).....	54-22
54.13	Port 7 (P7).....	54-23
54.14	Port 8 (P8).....	54-25
54.15	Port 9 (P9).....	54-26
54.16	Port 10 (P10).....	54-27
54.17	Port 11 (P11).....	54-28
54.18	Port Control Logical Diagram .....	54-29
54.19	Flowchart Examples of Port Setting .....	54-30
55.	Power-Down Modes.....	55-1
55.1	Features.....	55-1
55.1.1	States of Processing and Power-Down Modes .....	55-1

55.2	Register Descriptions.....	55-3
55.2.1	Standby Control Register 1 (STBCR1).....	55-4
55.2.2	Standby Control Register 2 (STBCR2).....	55-5
55.2.3	Standby Control Register 3 (STBCR3).....	55-6
55.2.4	Standby Control Register 4 (STBCR4).....	55-7
55.2.5	Standby Control Register 5 (STBCR5).....	55-8
55.2.6	Standby Control Register 6 (STBCR6).....	55-9
55.2.7	Standby Control Register 7 (STBCR7).....	55-10
55.2.8	Standby Control Register 8 (STBCR8).....	55-11
55.2.9	Standby Control Register 9 (STBCR9).....	55-12
55.2.10	Standby Control Register 10 (STBCR10).....	55-13
55.2.11	Standby Control Register 11 (STBCR11).....	55-14
55.2.12	Standby Control Register 12 (STBCR12).....	55-15
55.2.13	Standby Control Register 13 (STBCR13).....	55-16
55.2.14	Software Reset Control Register 1 (SWRSTCR1).....	55-17
55.2.15	Software Reset Control Register 2 (SWRSTCR2).....	55-18
55.2.16	Software Reset Control Register 3 (SWRSTCR3).....	55-18
55.2.17	System Control Register 1 (SYSCR1).....	55-19
55.2.18	System Control Register 2 (SYSCR2).....	55-20
55.2.19	System Control Register 3 (SYSCR3).....	55-21
55.2.20	CPU Status Register (CPUSTS).....	55-22
55.2.21	Standby Request Register 1 (STBREQ1).....	55-23
55.2.22	Standby Request Register 2 (STBREQ2).....	55-24
55.2.23	Standby Acknowledgment Register 1 (STBACK1).....	55-25
55.2.24	Standby Acknowledge Register 2 (STBACK2).....	55-26
55.2.25	On-Chip Data-Retention RAM Area Setting Register (RRAMKP).....	55-27
55.2.26	Deep Standby Control Register (DSCTR).....	55-28
55.2.27	Deep Standby Cancel Source Select Register (DSSSR).....	55-29
55.2.28	Deep Standby Cancel Edge Select Register (DSESR).....	55-31
55.2.29	Deep Standby Cancel Source Flag Register (DSFR).....	55-32
55.2.30	XTAL Crystal Oscillator Gain Control Register (XTALCTR).....	55-33
55.3	Operation.....	55-34
55.3.1	Sleep Mode.....	55-34
55.3.2	Software Standby Mode.....	55-34
55.3.3	Software Standby Mode Application Example.....	55-36
55.3.4	Deep Standby Mode.....	55-36
55.3.5	Module Standby Function.....	55-40
55.3.6	Software Reset.....	55-41
55.3.7	Adjustment of XTAL Crystal Oscillator Gain.....	55-42
55.4	Usage Notes.....	55-43



55.4.1	Usage Notes on Setting Registers.....	55-43
55.4.2	Usage Notes when the Realtime Clock is not Used .....	55-43
<b>56.</b>	<b>Debugger Interface .....</b>	<b>56-1</b>
56.1	Features.....	56-1
56.2	Input/Output Pins.....	56-5
56.3	Registers for Boundary-Scan TAP Controller.....	56-6
56.3.1	Bypass Register (BSBPR).....	56-6
56.3.2	Instruction Register (BSIR).....	56-6
56.3.3	Boundary Scan Register (SDBSR).....	56-7
56.3.4	ID Register (BSID).....	56-12
56.4	ICE Registers.....	56-13
56.4.1	Mode Reset Control Register (ICEREGMDRSTCTL).....	56-13
56.4.2	JTAG Trace Select Register (ICEREGJTTRCSEL).....	56-14
56.4.3	Clock Power Control Register (ICEREGCLKPWRCTRL).....	56-15
56.4.4	Lock Access Register (ICEREGLOCKACCESS).....	56-15
56.5	Operation .....	56-16
56.5.1	TAP Controller .....	56-16
56.5.2	Reset Signal Setting.....	56-16
56.6	Boundary Scan.....	56-17
56.6.1	Supported Instructions.....	56-17
56.6.2	Points for Attention .....	56-18
56.7	Usage Notes.....	56-18
<b>57.</b>	<b>List of Registers .....</b>	<b>57-1</b>
57.1	Register Addresses .....	57-1
57.2	Register Bits .....	57-87
57.3	Register States .....	57-333
<b>58.</b>	<b>Electrical Characteristics.....</b>	<b>58-1</b>
58.1	Absolute Maximum Ratings.....	58-1
58.2	Power-On/Power-Off Sequence .....	58-1
58.3	DC Characteristics .....	58-2
58.4	AC Characteristics .....	58-8
58.4.1	Clock Timing.....	58-9
58.4.2	Control Signal Timing .....	58-13
58.4.3	Bus Timing.....	58-14
58.4.4	Direct Memory Access Controller Timing.....	58-40
58.4.5	Multi-Function Timer Pulse Unit 2 Timing .....	58-41
58.4.6	Watchdog Timer Timing .....	58-42
58.4.7	Serial Communication Interface with FIFO Timing .....	58-43
58.4.8	Serial Communication Interface Timing.....	58-44
58.4.9	Renesas Serial Peripheral Interface Timing .....	58-45

58.4.10	SPI Multi I/O Bus Controller Timing.....	58-48
58.4.11	I <sup>2</sup> C Bus Interface Timing .....	58-51
58.4.12	Serial Sound Interface Timing.....	58-52
58.4.13	Media Local Bus Timing.....	58-55
58.4.14	CAN Interface Timing.....	58-56
58.4.15	Ethernet Controller and EthernetAVB Timing.....	58-57
58.4.16	A/D Converter Timing.....	58-60
58.4.17	NAND Type Flash Memory Controller Timing.....	58-61
58.4.18	USB 2.0 Host/Function Module Timing .....	58-65
58.4.19	Video Display Controller 5 Timing.....	58-68
58.4.20	LVDS Timing.....	58-70
58.4.21	Capture Engine Unit Module Signal Timing.....	58-72
58.4.22	MMC Host Interface Timing.....	58-73
58.4.23	General Purpose I/O Ports Timing .....	58-74
58.4.24	Debugger Interface Timing .....	58-74
58.4.25	AC Characteristics Measurement Conditions.....	58-76
58.5	A/D Converter Characteristics.....	58-77
58.6	Video Characteristics of A/D Converter for the Input of Video Signals.....	58-78
<b>59.</b>	<b>States and Handling of Pins.....</b>	<b>59-1</b>
59.1	Pin States .....	59-1
59.2	Treatment of Unused Pins .....	59-8
59.3	Handling of Pins in Deep Standby Mode .....	59-9
59.4	Recommended Combination of Bypass Capacitor.....	59-10
<b>Package</b>	.....	<b>Package-1</b>
A.	Package Dimensions .....	Package-1
B.	Thermal Characteristics.....	Package-4

## 1. Overview

### 1.1 Features of This LSI

This LSI is a single-chip microcontroller that includes an ARM Cortex™-A9 processor along with the integrated peripheral functions required to configure a system.

This LSI includes a 32-Kbyte L1 instruction cache, a 32-Kbyte L1 data cache, and a 128-Kbyte L2 cache. This LSI also includes on-chip peripheral functions necessary for system configuration, such as a 10-Mbyte large-capacity RAM (128 Kbytes are shared by the data-retention RAM), data-retention RAM, multi-function timer pulse unit 2, OS timer, realtime clock, serial communication interface with FIFO, serial communication interface, I<sup>2</sup>C bus interface, serial sound interface, media local bus, SCUX, CAN interface, IEBus™\*1 controller, Renesas SPDIF interface, Renesas serial peripheral interface, SPI multi I/O bus controller, CD-ROM decoder, A/D converter, LIN interface, Ethernet controller, EthernetAVB, NAND flash memory controller, USB 2.0 host/function, digital video decoder, video display controller 5, dynamic range compression, image renderer, image renderer for display, display out comparison unit, Renesas graphics processor for OpenVG™\*2, JPEG codec unit, capture engine unit, pixel format converter, sound generator, SD host interface, MMC host interface, motor control PWM timer, interrupt controller modules, and general I/O ports.

The features of this LSI are listed in Table 1.1.

Note 1. IEBus (Inter Equipment Bus) is a trademark of Renesas Electronics Corporation.

Note 2. OpenVG is a trademark of Khronos Group Inc.

**Table 1.1 RZ/A1H Features**

Items	Specification
CPU	<ul style="list-style-type: none"> <li>• ARM Cortex-A9 processor</li> <li>• Maximum operating frequency: 400 MHz</li> <li>• Instruction cache size: 32 Kbytes</li> <li>• Data cache size: 32 Kbytes</li> <li>• TLB entries: 128 entries</li> <li>• Jazelle architecture extension: Full implementation</li> <li>• Media processing engine with NEON technology</li> </ul>
L2 cache memory	<ul style="list-style-type: none"> <li>• ARM CoreLink™ Level 2 Cache Controller L2C-310</li> <li>• Operating frequency: 133 MHz</li> <li>• Cache size: 128 Kbytes</li> </ul>
Interrupt controller	<ul style="list-style-type: none"> <li>• ARM PrimeCell® Generic Interrupt Controller (PL390)</li> <li>• External interrupt pins (NMI, IRQ7 to IRQ0, and TINT170 to TINT0)</li> <li>• On-chip peripheral interrupts: Priority level set for each module</li> <li>• 32 priority levels available</li> </ul>
Bus state controller	<ul style="list-style-type: none"> <li>• Address space divided into six areas (0 to 5), each a maximum of 64 Mbytes</li> <li>• The following features settable for each area independently <ul style="list-style-type: none"> <li>—Bus size (8, 16, or 32 bits): Available sizes depend on the area.</li> <li>—Number of access wait cycles (different wait cycles can be specified for read and write access cycles in some areas)</li> <li>—Idle wait cycle insertion (between the same area access cycles or different area access cycles)</li> <li>—Specifying the memory to be connected to each area enables direct connection to SRAM, SRAM with byte selection, SDRAM, and burst ROM (clocked synchronous or asynchronous). The address/data multiplexed I/O (MPX) interface is also available.</li> </ul> </li> <li>• Outputs a chip select signal (<math>\overline{CS0}</math> to <math>\overline{CS5}</math>) according to the target area (<math>\overline{CS}</math> assert or negate timing can be selected by software)</li> <li>• SDRAM refresh</li> <li>• Auto refresh or self refresh mode selectable</li> <li>• SDRAM burst access</li> </ul>
Direct memory access controller	<ul style="list-style-type: none"> <li>• Sixteen channels; external requests are available for one of them.</li> <li>• Can be activated by on-chip peripheral modules.</li> <li>• A specific DMA transfer interval can be specified to adjust the bus occupancy.</li> <li>• Link mode (DMA transfer under descriptor control) supported</li> <li>• Transfer information can be automatically reloaded.</li> </ul>
Clock pulse generator	<ul style="list-style-type: none"> <li>• Clock mode: Input clock can be selected from external input (EXTAL or USB_X1) or crystal resonator.</li> <li>• Input clock can be multiplied by 32 (max.) by the internal PLL circuit.</li> <li>• Peak values of EMI noise can be reduced by the on-chip SSCG circuit.</li> <li>• Five types of clocks generated: <ul style="list-style-type: none"> <li>—CPU clock (<math>I\phi</math>): Maximum 400.00 MHz</li> <li>—Image processing clock (<math>G\phi</math>): Maximum 266.67 MHz</li> <li>—Internal bus clock (<math>B\phi</math>): Maximum 133.33 MHz</li> <li>—Peripheral clock 1 (<math>P1\phi</math>): Maximum 66.67 MHz</li> <li>—Peripheral clock 0 (<math>P0\phi</math>): Maximum 33.33 MHz</li> </ul> </li> </ul>
Watchdog timer	<ul style="list-style-type: none"> <li>• On-chip one-channel watchdog timer</li> <li>• A counter overflow can reset the LSI.</li> </ul>
Power-down modes	<ul style="list-style-type: none"> <li>• Four power-down modes provided to reduce the power consumption in this LSI <ul style="list-style-type: none"> <li>—Sleep mode</li> <li>—Software standby mode</li> <li>—Deep standby mode</li> <li>—Module standby mode</li> </ul> </li> </ul>

Items	Specification
Multi-function timer pulse unit 2	<ul style="list-style-type: none"> <li>• Maximum 16 lines of pulse inputs/outputs based on five channels of 16-bit timers</li> <li>• 18 output compare and input capture registers</li> <li>• Input capture function</li> <li>• Pulse output modes</li> <li>• Toggle, PWM, complementary PWM, and reset-synchronized PWM modes</li> <li>• Synchronization of multiple counters</li> <li>• Complementary PWM output mode <ul style="list-style-type: none"> <li>—Non-overlapping waveforms output for 3-phase inverter control</li> <li>—Automatic dead time setting</li> <li>—0% to 100% PWM duty value specifiable</li> <li>—A/D converter start request delaying function</li> <li>—Interrupt skipping at crest or trough</li> </ul> </li> <li>• Reset-synchronized PWM mode <ul style="list-style-type: none"> <li>Three-phase PWM waveforms in positive and negative phases can be output with a required duty value.</li> </ul> </li> <li>• Phase counting mode <ul style="list-style-type: none"> <li>Two-phase encoder pulse counting available</li> </ul> </li> </ul>
OS timer	<ul style="list-style-type: none"> <li>• Two-channel 32-bit counters</li> <li>• Two operating modes: <ul style="list-style-type: none"> <li>—Interval timer mode</li> <li>—Free-running comparison mode</li> </ul> </li> <li>• DMA transfer request or interrupt request can be issued when a compare match occurs.</li> </ul>
Realtime clock	<ul style="list-style-type: none"> <li>• Internal clock, calendar function, alarm function</li> <li>• Interrupts can be generated at intervals of 1/64 s by the 32.768-kHz or 4-MHz on-chip crystal oscillator.</li> </ul>
Serial communication interface with FIFO	<ul style="list-style-type: none"> <li>• Eight channels</li> <li>• Clock synchronous mode or asynchronous mode selectable</li> <li>• Simultaneous transmission and reception (full-duplex communication) supported</li> <li>• Dedicated baud rate generator</li> <li>• Separate 16-byte FIFO registers for transmission and reception</li> <li>• Modem control function (channels 1, 5, and 7 in asynchronous mode)</li> </ul>
Serial communication interface	<ul style="list-style-type: none"> <li>• Two channels</li> <li>• Clock synchronous mode, asynchronous mode, or smart card interface mode is selectable.</li> <li>• Simultaneous transmission and reception (full-duplex communication) supported</li> <li>• Dedicated baud rate generator</li> <li>• LSB first/MSB first selectable</li> <li>• Modem control function</li> <li>• Encoding and decoding of IrDA communications waveforms in accord with version 1.0 of the IrDA standard (on channel 0)</li> </ul>
Renesas serial peripheral interface	<ul style="list-style-type: none"> <li>• Five channels</li> <li>• SPI operation</li> <li>• Master mode and slave mode selectable</li> <li>• Programmable bit length, clock polarity, and clock phase can be selected.</li> <li>• Consecutive transfers</li> <li>• MSB first/LSB first selectable</li> <li>• Maximum transfer rate: 33.33 Mbps</li> </ul>
SPI multi I/O bus controller	<ul style="list-style-type: none"> <li>• Two channels</li> <li>• Up to two serial flash memories with multiple I/O bus sizes (single/dual/quad) can be connected.</li> <li>• External address space read mode (built-in read cache)</li> <li>• SPI operating mode</li> <li>• Clock polarity and clock phase can be selected.</li> <li>• MSB first/LSB first selectable</li> <li>• Maximum transfer rate: 533.33 Mbps (with two serial flash memories connected)</li> </ul>
I <sup>2</sup> C bus interface	<ul style="list-style-type: none"> <li>• Four channels</li> <li>• Master mode and slave mode supported</li> <li>• Support for 7-bit and 10-bit slave address formats</li> <li>• Support for multi-master operation</li> <li>• Timeout detection</li> </ul>

Items	Specification
Serial sound interface	<ul style="list-style-type: none"> <li>• Six-channel bidirectional serial transfer</li> <li>• Duplex communication (channels 0, 1, 3, and 5)</li> <li>• Support of various serial audio formats</li> <li>• Support of master and slave functions</li> <li>• Generation of programmable word clock and bit clock</li> <li>• Multi-channel formats</li> <li>• Support of 8, 16, 18, 20, 22, 24, and 32-bit data formats</li> <li>• Support of eight-stage FIFO for transmission and reception</li> <li>• Support of TDM mode</li> <li>• Support of WS continue mode in which the SSIWS signal is not stopped.</li> <li>• Support of direct transfer to the SCUX module</li> <li>• A change of the sampling frequency can be detected.</li> </ul>
Media local bus	<ul style="list-style-type: none"> <li>• MediaLB (OS62400 manufactured by SSMC) for connection with the INIC incorporated.</li> <li>• Conforms with version 2.0 of the MediaLB standard. Data transfer at up to 50 Mbps is possible.</li> </ul>
SCUX	<ul style="list-style-type: none"> <li>• Sampling rate conversion <ul style="list-style-type: none"> <li>—Asynchronous or synchronous sampling rate conversion is possible.</li> <li>—Sampling rate (synchronous mode) <p>Note: The selectable sampling rates depend on the number of used channels and rate ratio. Input [kHz]: 8, 11.025, 12, 16, 22.05, 24, 32, 44.1, 48, 64, 88.2, or 96 is selectable. Output [kHz]: 8, 16, 24, 44.1, 48, or 96 is selectable.</p> </li> <li>—Sampling rate (asynchronous mode) <p>Note: The selectable sampling rates depend on the number of used channels and rate ratio. Input/output [kHz]: 1 to 96</p> </li> <li>—Data format: 16 or 24 bits</li> </ul> </li> <li>• Digital volume and mute functions <ul style="list-style-type: none"> <li>—The digital volume can be set within the range from a multiple of 0 to 8 (–120 to 18 dB)</li> <li>—Volume ramping supports soft mute, fade-in, and fade-out.</li> <li>—The zero crossing mute function can apply muting at zero-crossing points.</li> </ul> </li> <li>• Mixer <ul style="list-style-type: none"> <li>—Data of two to four source systems can be mixed (added together) into one system.</li> <li>—The ratio to add the sources can be set.</li> <li>—Direct transfer to the serial sound interface module is supported.</li> </ul> </li> </ul>
CAN interface	<ul style="list-style-type: none"> <li>• Five channels</li> <li>• ISO11898-1 compliant</li> <li>• Message buffer: <ul style="list-style-type: none"> <li>—Up to 64 5-channel receive message buffers: shared among all channels.</li> <li>—16 transmit message buffers per channel</li> </ul> </li> </ul>
IEBus™ controller	<ul style="list-style-type: none"> <li>• Conforms with the IEBus protocol (communication modes 1 and 2).</li> <li>• Transfer rates: approximately 17 kbps (in communication mode 1), approximately 26 kbps (in communication mode 2)</li> <li>• Maximum numbers of bytes for transfer: 32 bytes/frame (in communication mode 1), 128 bytes/frame (in communication mode 2)</li> <li>• Operating clock: 8 MHz Note: Input of peripheral clock 0 (P0φ) running at 32 MHz is required.</li> </ul>
Renesas SPDIF interface	<ul style="list-style-type: none"> <li>• Support of IEC60958 standard (stereo and consumer use modes only)</li> <li>• Sampling frequencies of 32 kHz, 44.1 kHz, and 48 kHz</li> <li>• Audio word sizes of 16 to 24 bits per sample</li> <li>• Biphasic mark encoding</li> <li>• Double buffered data</li> <li>• Parity encoded serial data</li> <li>• Simultaneous transmit and receive</li> <li>• Receiver autodetects IEC 61937 compressed mode data.</li> </ul>

Items	Specification
CD-ROM decoder	<ul style="list-style-type: none"> <li>• Support of five formats: Mode 0, mode 1, mode 2, mode 2 form 1, and mode 2 form 2</li> <li>• Sync codes detection and protection</li> <li>• (Protection: When a sync code is not detected, it is automatically inserted.)</li> <li>• Descrambling</li> <li>• ECC correction <ul style="list-style-type: none"> <li>—P, Q, PQ, and QP correction</li> <li>—PQ or QP correction can be repeated up to three times.</li> </ul> </li> <li>• EDC check <ul style="list-style-type: none"> <li>Performed before and after ECC</li> </ul> </li> <li>• Mode and form are automatically detected.</li> <li>• Link sectors are automatically detected.</li> <li>• Buffering data control <ul style="list-style-type: none"> <li>Buffering CD-ROM data including Sync code is transferred in specified format, after the data is descrambled, corrected by ECC, and checked by EDC.</li> </ul> </li> </ul>
LIN interface	<ul style="list-style-type: none"> <li>• Two channels</li> <li>• Conforms with revisions 1.3, 2.0, 2.1, and 2.2 of the LIN protocol and SAEJ 2062.</li> <li>• Master mode supported</li> </ul>
Ethernet controller	<ul style="list-style-type: none"> <li>• Conforms with the Ethernet or the IEEE802.3 MAC (Media Access Control) layer standard</li> <li>• MAC function <ul style="list-style-type: none"> <li>Constructs/deconstructs data frames (frame format conforming to IEEE802.3, 2000 Edition)</li> <li>Supports transfer at 10 and 100 Mbps</li> <li>Supports full-duplex mode</li> <li>Flow control conforming to IEEE802.3x</li> <li>Supports an MII (Media Independent Interface) for connection to a PHY interface in conformance with IEEE 802.3</li> <li>Upward protocol support (checksum) function</li> </ul> </li> <li>• E-DMAC (Direct Memory Access Controller for Ethernet controller) function</li> </ul>
EthernetAVB	<ul style="list-style-type: none"> <li>• Conforms with the Ethernet or the IEEE802.3 MAC (Media Access Control) layer standard</li> <li>• MAC function <ul style="list-style-type: none"> <li>Constructs/deconstructs data frames (frame format conforming to IEEE802.3, 2000 Edition)</li> <li>Supports transfer at 100 Mbps</li> <li>Supports full-duplex mode</li> <li>Flow control conforming to IEEE802.3x</li> <li>Supports an MII (Media Independent Interface) for connection to a PHY interface in conformance with IEEE 802.3</li> <li>Upward protocol support (checksum) function</li> </ul> </li> <li>• AVB-DMAC (DMAC dedicated to EthernetAVB) function <ul style="list-style-type: none"> <li>AVB-DMAC conforms with the following 3 standards; IEEE802.1AS (Clock Synchronization Protocol), IEEE802.1Qav (Realtime Transfer Protocol) and IEEE802.1Qat (Bandwidth Reservation Protocol)</li> </ul> </li> </ul>
NAND flash memory controller	<ul style="list-style-type: none"> <li>• Direct-connected memory interface with NAND-type flash memory</li> <li>• Command access mode</li> <li>• Interrupt request and DMA transfer request</li> <li>• Supports flash memory requiring 5-byte addresses (2 Gbits and more)</li> </ul>
USB 2.0 host/function module	<ul style="list-style-type: none"> <li>• Two channels</li> <li>• Conforms to the Universal Serial Bus Specification Revision 2.0</li> <li>• 480-Mbps, 12-Mbps, and 1.5-Mbps transfer rates provided (host mode)</li> <li>• 480-Mbps and 12-Mbps transfer rates provided (function mode)</li> <li>• On-chip 8-Kbyte RAM as communication buffers</li> </ul>

Items	Specification
Digital video decoder	<ul style="list-style-type: none"> <li>• Two channels</li> <li>• Video input <ul style="list-style-type: none"> <li>Composite video input (CVBS)</li> </ul> </li> <li>• A/D converter for video signal input <ul style="list-style-type: none"> <li>VIN1 and VIN2 pin input selection</li> <li>Low-pass filter (LPF)</li> <li>Sync tip clamp</li> <li>Programmable gain amplifier (PGA) (0 to 6.021 dB)</li> <li>10-bit precision pipelined A/D converter</li> </ul> </li> <li>• Sync separation <ul style="list-style-type: none"> <li>Noise reduction LPF, auto level control sync slicer, horizontal auto frequency control (AFC), vertical count-down, interlace detection, auto gain control (AGC)/peak limiter control</li> </ul> </li> <li>• Y/C separation <ul style="list-style-type: none"> <li>NTSC 2D, PAL 2D, and SECAM 1D supported.</li> </ul> </li> <li>• Chroma-key decoding <ul style="list-style-type: none"> <li>NTSC, PAL, and SECAM supported.</li> <li>Color killer, auto color control (ACC), TINT correction, R-Y axis correction</li> </ul> </li> <li>• Digital clamp <ul style="list-style-type: none"> <li>Pedestal clamp (Y), center clamp (Cb/Cr), noise detection</li> </ul> </li> <li>• Adjustment of output gain <ul style="list-style-type: none"> <li>Contrast: 0 to approximately 2 times</li> <li>Color (Cb/Cr independently): 0 to approximately 2 times</li> </ul> </li> </ul>
Video display controller 5	<ul style="list-style-type: none"> <li>• Two channels</li> <li>• Video input interface: One channel can be selected from the followings. <ul style="list-style-type: none"> <li>BT601, BT656 format (NTSC/PAL) input: Input clock: 27 MHz/54 MHz</li> <li>Digital pin input (channel 0): <ul style="list-style-type: none"> <li>YCbCr422, YCbCr444, RGB888, RGB666, RGB565</li> </ul> </li> <li>Digital pin input size: <ul style="list-style-type: none"> <li>Maximum input video image size to be set*: 1440 pixels × 1024 lines (horizontal × vertical)</li> <li>Note:*Depends on the AC characteristics of the connected device.</li> <li>Examples of input video image size : <ul style="list-style-type: none"> <li>WXGA (1280 × 768)</li> <li>XGA (1024 × 768)</li> <li>SVGA (800 × 600),</li> <li>WVGA (800 × 480),</li> <li>VGA (640 × 480),</li> <li>WQVGA (480 × 240),</li> <li>QVGA (320 × 240, 240 × 320)</li> </ul> </li> </ul> </li> <li>Composite video (CVBS) signal input decoded by the digital video decoder</li> </ul> </li> <li>• Input video control <ul style="list-style-type: none"> <li>Horizontal noise reduction (NR), brightness adjustment and contrast adjustment using matrix operation</li> </ul> </li> <li>• Scaling control <ul style="list-style-type: none"> <li>Vertical and horizontal scaling up or down of input video possible at a desired ratio (scaling up of graphics also possible)</li> <li>Scaling up ratio: 1 to 8; scaling down ratio: 1/8 to 1</li> <li>Interpolation: Hold or linear selectable</li> <li>2D IP conversion: 2D IP conversion through separately setting the initial phases for the top and bottom fields</li> </ul> </li> <li>• Video recording <ul style="list-style-type: none"> <li>Output pixel format: YCbCr444, YCbCr422, RGB888, RGB565</li> <li>Output field rate: 1/1, 1/2, 1/4, 1/8</li> <li>Rotation: Horizontal mirroring and 90/180/270 degree rotation for YCbCr422 and RGB565</li> <li>Maximum video image size to be stored: ×1 size of input video image</li> </ul> </li> <li>• Output video control <ul style="list-style-type: none"> <li>Black stretch: Black area stretched according to Y signal state</li> <li>Enhancer capability: LTI (transient improvement) and sharpness (contour emphasis) for Y signal</li> </ul> </li> </ul>



Items	Specification
Video display controller 5	<ul style="list-style-type: none"> <li>Four graphics layers (two of them also for input video) <ul style="list-style-type: none"> <li>Available input pixel formats <ul style="list-style-type: none"> <li>1 bit/pixel: CLUT1</li> <li>4 bits/pixel: CLUT4</li> <li>8 bits/pixel: CLUT8</li> <li>16 bits/pixel: YCbCr422 (graphics layers 0 and 1), RGB565, ARGB1555, RGBA5551, ARGB4444</li> <li>32 bits/pixel: ARGB8888, RGBA8888, RGB888, YCbCr444 (graphics layers 0 and 1)</li> </ul> </li> </ul> </li> <li>Blending of two input video images <ul style="list-style-type: none"> <li>Two input video images superimposed by alpha blending over a rectangular area can be output.</li> </ul> </li> <li>Superimposition <ul style="list-style-type: none"> <li>Alpha blending in a rectangular area: <ul style="list-style-type: none"> <li>Input video, layer 1, and layer 2 blended according to the transparency percentage <math>\alpha</math> (fade-in and fade-out function available)</li> </ul> </li> <li>Chroma key function: <ul style="list-style-type: none"> <li>Mixing based on transparency percentage <math>\alpha</math> using the specified RGB and CLUT value</li> </ul> </li> <li>Pixel-base alpha blending: <ul style="list-style-type: none"> <li>Alpha blending for each pixel based on transparency percentage <math>\alpha</math></li> <li>Generation of output video images</li> <li>Video images superimposed on graphics layers can be output to memory.</li> </ul> </li> </ul> </li> <li>Panel output control <ul style="list-style-type: none"> <li>Panel output correction: <ul style="list-style-type: none"> <li>Brightness adjustment and contrast adjustment, gamma correction, panel dithering</li> </ul> </li> <li>TCON: <ul style="list-style-type: none"> <li>Various timing output for LCD panel driving provided by a total of seven vertical and horizontal panel driver signals</li> </ul> </li> <li>Panel output pixel format: RGB888, RGB666, RGB565, serial RGB</li> <li>Output video image size: <ul style="list-style-type: none"> <li>Maximum output video image size to be set*: <ul style="list-style-type: none"> <li>1999 pixels <math>\times</math> 2035 lines (horizontal <math>\times</math> vertical)</li> </ul> </li> <li>Note:*Depends on the AC characteristics of the display panel.</li> <li>Examples of output video image size: <ul style="list-style-type: none"> <li>WXGA (1280 <math>\times</math> 768)</li> <li>XGA (1024 <math>\times</math> 768)</li> <li>SVGA (800 <math>\times</math> 600),</li> <li>WVGA (800 <math>\times</math> 480),</li> <li>VGA (640 <math>\times</math> 480),</li> <li>WQVGA (480 <math>\times</math> 240),</li> <li>QVGA (320 <math>\times</math> 240, 240 <math>\times</math> 320)</li> </ul> </li> </ul> </li> </ul> </li> </ul>
Dynamic range compression	<ul style="list-style-type: none"> <li>Two channels</li> <li>Contrast adjustment of captured data</li> <li>Contrast expansion processing optimized per region of the image</li> </ul>
Image renderer (IMR-LS2)	<ul style="list-style-type: none"> <li>Two channels</li> <li>Refers to the video captured data as two-dimensional texture data and draws a shape by performing texture mapping for an arbitrary shape divided into triangular objects.</li> <li>Display list system</li> <li>Drawing functions <ul style="list-style-type: none"> <li>Texture mapping, bilinear filtering, automatic coordinate generation (and relative coordinate input)</li> </ul> </li> <li>Instruction system <ul style="list-style-type: none"> <li>Draw instruction: TRI for drawing a triangle</li> <li>Control instructions: TRAP, INT, NOP, SYNCM, SYNCW, WTL, and WTS</li> </ul> </li> <li>Drawing space <ul style="list-style-type: none"> <li>Destination coordinates: <math>0 \leq X \leq 2,047</math>, <math>0 \leq Y \leq 2,047</math></li> <li>Source coordinates: <math>0 \leq u \leq 1,439</math>, <math>0 \leq v \leq 1,023</math></li> </ul> </li> </ul>
Image renderer for display (IMR-LSD)	<ul style="list-style-type: none"> <li>Refers to the output video image data from the video display controller 5 (channel 0) as two-dimensional texture data and draws shapes by performing texture mapping for an arbitrary shape divided into triangular objects.</li> <li>Display list system</li> <li>Drawing functions <ul style="list-style-type: none"> <li>Texture mapping, bilinear filtering, automatic coordinate generation (and relative coordinate input)</li> </ul> </li> <li>Instruction system <ul style="list-style-type: none"> <li>Draw instruction: TRI for drawing a triangle</li> <li>Control instructions: TRAP, INT, NOP, SYNCM, SYNCW, WTL, and WTS</li> </ul> </li> <li>Drawing space <ul style="list-style-type: none"> <li>Destination coordinates: <math>0 \leq X \leq 2,047</math>, <math>0 \leq Y \leq 2,047</math></li> <li>Source coordinates: <math>0 \leq u \leq 1,439</math>, <math>0 \leq v \leq 1,023</math></li> </ul> </li> </ul>

Items	Specification
Display out comparison unit	<ul style="list-style-type: none"> <li>Two channels</li> <li>Calculates the CRC code of an arbitrary graphics plane and compares it with the pre-calculated CRC code.</li> <li>Specifies a rectangular area in an arbitrary graphics plane selected from among four graphics planes, one plane of the graphics data obtained after <math>\alpha</math> blending, or one plane of the data read from the output video image generator of the video display controller 5.</li> <li>Pixel format 32 bits/pixel: ARGB8888/RGB888/RGB666 16 bits/pixel: RGB565</li> </ul>
Renesas graphics processor for OpenVG™	<ul style="list-style-type: none"> <li>OpenVG™, which is an open 2D vector graphics API, can be processed.</li> <li>Processes can be accelerated in OpenVG™ stage 2 to stage 8 using the dedicated hardware and programmable shader.</li> </ul>
JPEG codec unit	<ul style="list-style-type: none"> <li>Compression and decompression method conforming to the JPEG baseline standard within the range described in this document.</li> <li>Operational precision: Conforming to JPEG Part 2, ISO-IEC10918-2</li> <li>Pixel format: Compression: YCbCr422 Decompression: YCbCr444, YCbCr422, YCbCr411, YCbCr420 Output pixel format to the buffer: YCbCr422, ARGB8888, RGB565</li> <li>Four quantization tables provided</li> <li>Four Huffman tables provided (two tables for AC coefficients and two tables for DC coefficients)</li> <li>Markers supported: SOI, SOF0, SOS, DQT, DHT, DRI, RSTm, and EOI</li> <li>Image data rate: Max. 133.33 Mbytes/s (at 66.67-MHz operation)</li> </ul>
Capture engine unit	<ul style="list-style-type: none"> <li>Examples of input video image size : WXGA (1280 × 768) XGA (1024 × 768) SVGA (800 × 600), WVGA (800 × 480), VGA (640 × 480), WQVGA (480 × 240), QVGA (320 × 240, 240 × 320) Note: Depends on the AC characteristics of the connected device.</li> <li>Input format: 8- or 16-bit YCbCr422 binary data</li> <li>Memory output format: YCbCr422, YCbCr420 Note: The captured data cannot be displayed via the video display controller 5 because the Y data and CbCr data are split when written to memory.</li> </ul>
Pixel format converter	<ul style="list-style-type: none"> <li>Two channels</li> <li>Brightness adjustment, gain adjustment, and YCbCr and RGB mutual conversion.</li> <li>Input pixel data: RGB888, RGB565, YCbCr422</li> <li>Output pixel data: ARGB8888, RGB565, YCbCr422</li> </ul>
Sound generator	<ul style="list-style-type: none"> <li>Four channels</li> <li>Capable of adjusting sound volume using 8-bit PWM output</li> <li>Four types of operating clocks (<math>P0\phi/2</math>, <math>P0\phi/4</math>, <math>P0\phi/8</math>, and <math>P0\phi/16</math>) can be selected.</li> <li>Frequency settings in the 25-Hz to 20-kHz range with precision of 1% or less</li> <li>Automatic attenuator function can be selected.</li> </ul>
SD host interface	<ul style="list-style-type: none"> <li>Two channels</li> <li>SD memory I/O card interface (1-/4-bit SD bus)</li> <li>Error check function: CRC7 (command), CRC16 (data)</li> <li>Interrupt requests <ul style="list-style-type: none"> <li>—Card access interrupt</li> <li>—SDIO access interrupt</li> <li>—Card detect interrupt</li> </ul> </li> <li>DMA transfer requests <ul style="list-style-type: none"> <li>—SD_BUF write</li> <li>—SD_BUF read</li> </ul> </li> <li>Card detection function, write protect supported</li> </ul>
MMC host interface	<ul style="list-style-type: none"> <li>Interface to multi-media card (MMC)</li> <li>Data bus: 1-/4-/8-bit MMC mode</li> <li>Interrupt requests: card detection, error/time-out, and normal operation</li> <li>DMA transfer requests: CE_DATA write and CE_DATA read</li> <li>Card detection function</li> </ul>
General I/O ports	<ul style="list-style-type: none"> <li>256-pin: 115 I/Os, 8 inputs with open-drain outputs, and 16 inputs (input only)</li> <li>324-pin: 147 I/Os, 8 inputs with open-drain outputs, and 16 inputs (input only)</li> <li>Input or output can be selected for each bit.</li> </ul>

Items	Specification
A/D converter	<ul style="list-style-type: none"> <li>• 12-bit resolution</li> <li>• Eight input channels</li> <li>• Minimum conversion time: 5.0 <math>\mu</math>s</li> <li>• A/D conversion request by the external trigger or timer trigger</li> </ul>
Motor control PWM timer	<ul style="list-style-type: none"> <li>• Two 10-bit PWM channels, each with eight outputs</li> </ul>
Debugging interface	<ul style="list-style-type: none"> <li>• ARM CoreSight architecture</li> <li>• JTAG-standard pin assignment</li> </ul>
On-chip RAM	<ul style="list-style-type: none"> <li>• 10-Mbyte large capacity memory for video display/recording and work (128 Kbytes are used for data retention)</li> <li>• 128-Kbyte memory for data retention (16 Kbytes <math>\times</math> 2, 32 Kbytes <math>\times</math> 1, 64 Kbytes <math>\times</math> 1)</li> </ul>
Boot modes	<ul style="list-style-type: none"> <li>• Five boot modes</li> <li>Boot mode 0: Booting from memory (bus width: 16 bits) connected to the CS0 space</li> <li>Boot mode 1: Booting from memory (bus width: 32 bits) connected to the CS0 space</li> <li>Boot mode 2: —</li> <li>Boot mode 3: Booting from a serial flash memory</li> <li>Boot mode 4: Booting from a NAND flash memory with SD controller</li> <li>Boot mode 5: Booting from a NAND flash memory with MMC controller</li> </ul>
Power supply voltage	<ul style="list-style-type: none"> <li>• Vcc: 1.10 to 1.26 V</li> <li>• PVcc: 3.0 to 3.6 V</li> </ul>
Package	<ul style="list-style-type: none"> <li>• PLBG0256KA-B 256-pin BGA, 11-mm square, 0.5-mm pitch JEITA package code: P-LFBGA256-11<math>\times</math>11-0.50 RENESAS code: PLBG0256KA-B</li> <li>• PLQP0256LB-A 256-pin QFP, 28-mm square, 0.4-mm pitch JEITA package code: P-LFQFP256-28<math>\times</math>28-0.40 RENESAS code: PLQP0256LB-A</li> <li>• PRBG0324GA-A 324-pin BGA, 19-mm square, 0.8-mm pitch JEITA package code: P-FBGA324-19<math>\times</math>19-0.80 RENESAS code: PRBG0324GA-A</li> </ul>

## 1.2 Product Lineup

**Table 1.2 Product Lineup**

Group	Part Number	Security	Temperature Range	Quality Level	Package
RZ/A1H	R7S721000VCBG	Not supported	-40 to +85°C	Industry usage etc.	PLBG0256KA-B
	R7S721000VCFP			Industry usage etc.	PLQP0256LB-A
	R7S721000VLFP			Car Accessories	
	R7S721001VCBG			Industry usage etc.	PRBG0324GA-A
	R7S721001VLBG			Car Accessories	

## 1.3 Block Diagram

See section 5, LSI Internal Bus.

1.4 Pin Assignment

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21		
A	PVCC	P6_4	P6_2	P6_0	P9_7	P9_4	P5_10	P5_8	P5_7	P5_3	P5_1	LVDSREF RIN	VIN2B	VDAVSS	VIN1A	P0_2	P1_4	P2_15	P2_12	PVCC	VSS	A	
B	VSS	PVCC	P6_3	P6_1	P0_3	P9_5	P5_9	VSS	P5_6	P5_2	P5_0	VSS	VIN1B	VDAVCC	VIN2A	P1_5	P1_0	P2_14	PVCC	VSS	P2_10	B	
C	VCC	VSS																		P2_11	P2_9	C	
D	P6_8	VCC		PVCC	P9_6	P9_3	LVDSAPV CC	LVDSPLL VCC	P5_5	P5_4	VRP	REXT	P1_7	P1_3	P1_1	P2_13	PVCC	VSS			P2_7	P2_8	D
E	P6_9	P6_7		VSS	PVCC	P9_2	LVDSAPV CC	LVDSAPV CC	VSS	VSS	VCC	VRM	P1_6	P1_2	VSS	PVCC	VSS	P2_4			P2_5	P2_6	E
F	P6_12	P6_10		VCC	VSS													P4_14	P4_15		P2_2	P2_3	F
G	P7_0	P6_11		P6_5	VCC													VCC	P4_11		P4_13	VSS	G
H	P7_2	P6_13		P6_6	VSS													VCC	P2_1		P4_10	P4_12	H
J	P7_4	P7_3		P6_14	VSS													VSS	P2_0		P4_9	P4_8	J
K	P7_8	P7_6		P7_1	P6_15													VSS	P4_3		P4_7	P4_6	K
L	P7_10	P7_9		P7_7	P7_5													P3_15	P4_0		P4_5	P4_4	L
M	P7_14	P7_13		P7_12	P7_11													P3_10	P3_11		P4_2	P4_1	M
N	P8_1	P8_2		P8_0	P7_15													JP0_1	JP0_0		P3_13	P3_14	N
P	P8_4	P8_5		P8_3	P8_6													PVCC	PVCC		VSS	P3_12	P
R	CKIO	P8_7		P8_10	PVCC													PVCC	PVCC		TMS	TCK	R
T	P8_8	P8_9		PVCC	VSS													VSS	BSCANP		P3_9	TRST	T
U	P8_11	P8_12		PVCC	VSS	VCC	VSS	P3_3	RES	VSS	PVCC	PVCC	USB APVCC	PLLVCC	VSS	VCC	VCC	VSS			AUDIO _X2	P3_8	U
V	P8_13	P8_14		VSS	VCC	P3_6	P3_5	P3_1	NMI	P0_4	P0_5	REFRIN	USB AVCC	P0_0	P1_9	P1_10	P1_11	VCC			VIDEO _X2	AUDIO _X1	V
W	P8_15	PVCC																			VSS	VIDEO _X1	W
Y	PVCC	VSS	VCC	P9_0	P3_4	VSS	RTC_X2	VSS	DP1	VBUS1	DP0	VBUS0	USB_X2	VSS	P0_1	P1_8	AVCC	P1_12	P1_14	VCC	VSS	Y	
AA	VSS	VCC	P9_1	P3_7	P3_2	P3_0	RTC_X1	VSS	DM1	VSS	DM0	VSS	USB_X1	EXTAL	XTAL	VSS	AVSS	AVREF	P1_13	P1_15	VCC	AA	

Figure 1.1 Pin Assignment of the 256-Pin BGA (Top Perspective View)

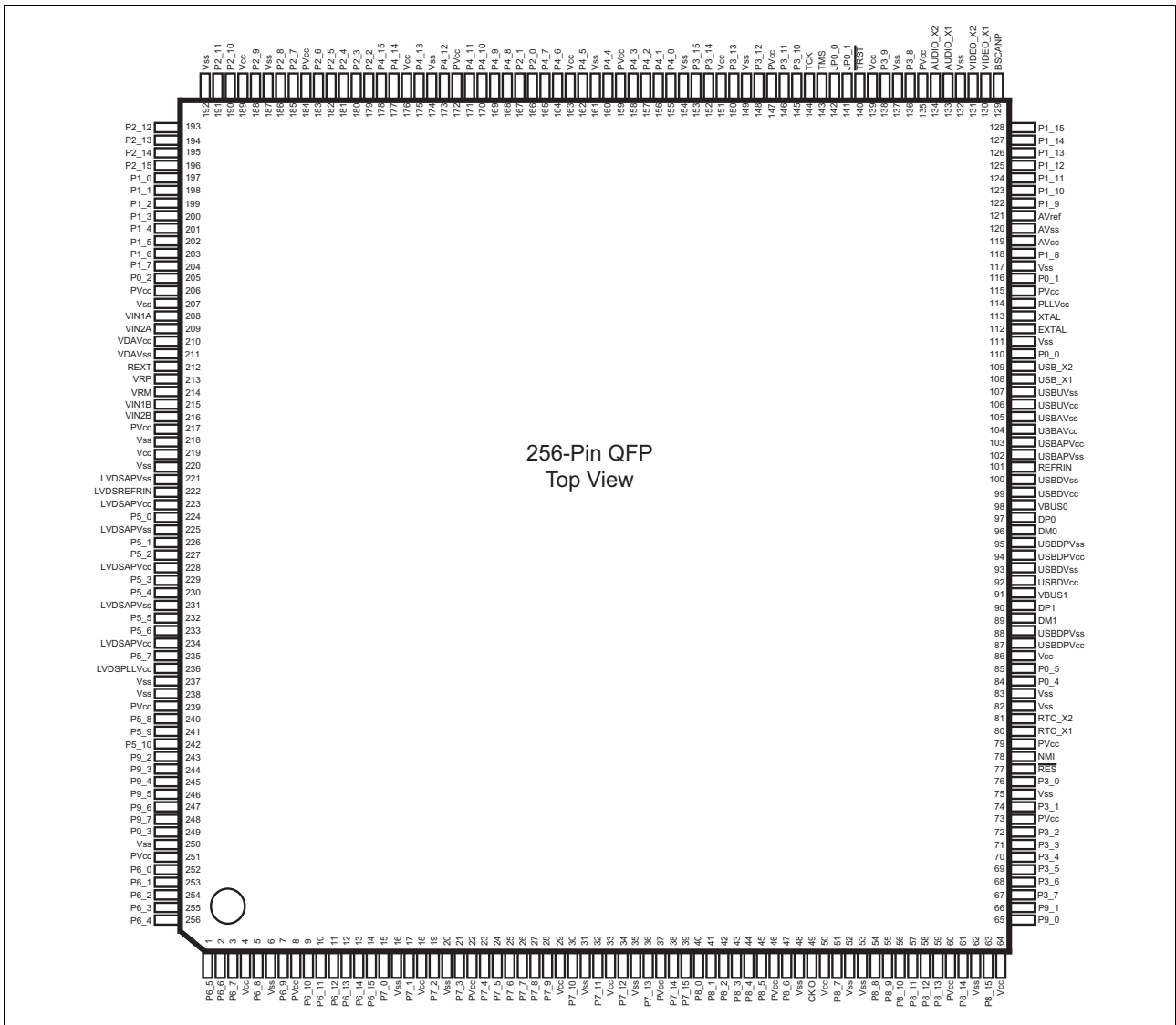


Figure 1.2 Pin Assignment of the 256-Pin QFP (Top Perspective View)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	
A	Vss	P6_4	P0_3	P11_0	P9_6	P9_3	P5_9	Vss	P5_6	P5_2	P5_0	Vss	VIN2B	VDAVss	VIN2A	P0_2	P1_6	P1_3	P1_0	P2_13	P2_12	Vss	A
B	Vcc	Vss	P6_0	P11_2	P9_7	P9_4	P5_10	P5_8	P5_7	P5_3	P5_1	Vss	VIN1B	VDAVcc	VIN1A	P1_7	P1_4	P1_2	P2_15	PVcc	Vss	P2_10	B
C	P6_5	Vcc	Vss	P6_2	P11_3	P11_1	P9_5	P9_2	Vss	P5_5	P5_4	LVDSAPVcc	VRM	REXT	Vss	P1_5	P1_1	P2_14	PVcc	Vss	P2_9	P2_7	C
D	P6_7	P6_6	Vcc	Vss	P6_3	P6_1	PVcc	PVcc	LVDSPLLVcc	Vss	LVDSREFRN	LVDSAPVcc	Vcc	VRP	Vss	PVcc	PVcc	PVcc	Vss	P2_8	P10_15	P10_14	D
E	P6_10	P6_9	P6_8	Vcc															P2_11	P2_6	P10_12	P2_5	E
F	P6_14	P6_13	P6_11	Vcc															P2_4	P10_13	P2_2	P4_15	F
G	P11_13	P11_12	P6_15	P6_12															P4_14	P2_3	P4_13	Vss	G
H	P7_2	P7_1	P11_14	P7_0															P4_11	P10_11	P10_10	P4_12	H
J	P7_5	P7_4	P7_3	P11_15															Vcc	P10_9	P10_8	P4_10	J
K	P7_9	P7_7	P7_6	P7_8															Vcc	P4_8	P4_9	P2_1	K
L	P11_5	P7_11	P7_10	P11_4															PVcc	P4_7	P2_0	P4_6	L
M	P7_12	P11_6	P11_7	Vcc															PVcc	P4_5	P4_4	P10_7	M
N	P7_13	P7_14	P7_15	PVcc															P10_4	P10_5	P10_6	P4_3	N
P	P8_0	P8_1	P8_2	PVcc															Vss	P4_0	P4_2	P4_1	P
R	P8_3	P8_4	P8_5	Vcc															Vcc	P3_15	P3_14	P3_13	R
T	P11_8	P11_9	P11_10	Vcc															Vcc	P3_10	P3_11	P3_12	T
U	Vss	P8_6	P11_11	P8_7															Vcc	JP0_1	TCK	Vss	U
V	CKIO	P8_8	P8_9	P8_13															P3_8	TRST	JP0_0	TMS	V
W	Vss	P8_10	P8_11	PVcc	PVcc	PVcc	Vss	Vss	Vcc	Vcc	Vss	PVcc	PVcc	PLLVcc	Vss	Vss	AVss	AVcc	PVcc	P3_9	AUDIO_X2	AUDIO_X1	W
Y	P8_12	P8_14	PVcc	P3_7	P3_4	P10_2	P3_2	RES	NMI	Vss	VBUS1	VBUS0	USBAPVcc	Vss	P0_0	P0_1	P1_10	P1_13	P1_15	PVcc	VIDEO_X2	VIDEO_X1	Y
AA	P8_15	PVcc	P9_1	P3_5	P10_1	P3_3	P3_1	RTC_X2	P0_5	Vss	DM1	DP0	REFRN	Vss	USB_X2	XTAL	P1_8	P1_11	P1_14	AVcc	PVcc	BSCANP	AA
AB	PVcc	P9_0	P3_6	P10_0	P10_3	P3_0	Vss	RTC_X1	P0_4	Vss	DP1	DM0	Vss	USBAPVcc	USB_X1	EXTAL	Vss	P1_9	P1_12	AVss	AVref	Vss	AB
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	

Figure 1.3 Pin Assignment of the 324-Pin BGA (Top Perspective View)

## 1.5 Pin Functions

**Table 1.3 Pin Functions**

Classification	Symbol	I/O	Name	Function
Power supply	Vcc	I	Power supply	Power supply pins. All the Vcc pins must be connected to the system power supply. This LSI does not operate correctly if there is a pin left open.
	Vss	I	Ground	Ground pins. All the Vss pins must be connected to the system power supply (0 V). This LSI does not operate correctly if there is a pin left open.
	PVcc	I	Power supply for I/O circuits	Power supply for I/O pins. All the PVcc pins must be connected to the system power supply. This LSI does not operate correctly if there is a pin left open.
	PLLvcc	I	Power supply for PLL	Power supply for the on-chip PLL oscillator.
Clock	EXTAL	I	External clock	Connected to a crystal resonator. An external clock signal may also be input to the EXTAL pin.
	XTAL	O	Crystal	Connected to a crystal resonator.
	CKIO	O	System clock output	Supplies the system clock to external devices.
	AUDIO_CLK	I	External clock for audio	Input pin of external clock for audio. A clock input to the divider is selected from an oscillation clock input on this pin or pins AUDIO_X1 and AUDIO_X2.
	AUDIO_X1	I	Crystal resonator/ external clock for audio	Pins connected to a crystal resonator for audio. An external clock can be input on pin AUDIO_X1. A clock input to the divider is selected from an oscillation clock input on these pins or the AUDIO_CLK pin.
	AUDIO_X2	O		
	AUDIO_XOUT	O	AUDIO_X1 clock output	Output for the on-chip crystal oscillator on AUDIO_X1 or the external clock signal.
	AUDIO_XOUT2	O	AUDIO_X1 divided-by-two clock output	Output for the on-chip crystal oscillator on AUDIO_X1 or the external clock signal after frequency division of the selected signal by two.
	AUDIO_XOUT3	O	AUDIO_X1 divided-by-three clock output	Output for the on-chip crystal oscillator on AUDIO_X1 or the external clock signal after frequency division of the selected signal by three.
	Operating mode control	MD_BOOT2, MD_BOOT1, MD_BOOT0	I	Mode set
MD_CLK		I	Clock mode set	Sets the clock operating mode. Do not change the signal levels on this pin while the RES pin is asserted or until the mode is fixed, after the negation.
MD_CLKS		I	SSCG clock mode set	Switches the SSCG circuit on or off. Do not change the signal levels on this pin while the RES pin is asserted or until the mode is fixed, after the negation.
BSCANP		I	Boundary scan set	Boundary scan setting pin. This pin is set to the high level for a boundary scan and to the low level for normal operation.



Classification	Symbol	I/O	Name	Function
System control	$\overline{\text{RES}}$	I	Power-on reset	This LSI enters the power-on reset state when this signal goes low.
	$\overline{\text{WDTOVF}}$	O	Watchdog timer overflow	Outputs an overflow signal from the watchdog timer.
Interrupts	NMI	I	Non-maskable interrupt	Non-maskable interrupt request pin. It is handled as an FIQ exception. Fix it high when not in use.
	IRQ7 to IRQ0	I	Interrupt requests 7 to 0	Maskable interrupt request pins. Level-input or edge-input detection can be selected. When the edge-input detection is selected, the rising edge, falling edge, or both edges can also be selected.
	TINT170 to TINT0	I	Interrupt requests 170 to 0	Maskable interrupt request pins. Detection through input of the high level or a rising edge can be selected.
Address bus	A25 to A0	O	Address bus	Outputs addresses.
Data bus	D31 to D0	I/O	Data bus	Bidirectional data bus.
Bus control	$\overline{\text{CS5}}$ to $\overline{\text{CS0}}$	O	Chip select 5 to 0	Chip-select signals for external memory or devices.
	$\overline{\text{RD}}$	O	Read	Indicates that data is read from an external device.
	$\overline{\text{RD/WR}}$	O	Read/write	Read/write signal.
	$\overline{\text{BS}}$	O	Bus start	Bus-cycle start signal.
	$\overline{\text{AH}}$	O	Address hold	Address hold timing signal for the device that uses the address/data-multiplexed bus.
	$\overline{\text{WAIT}}$	I	Wait	Inserts a wait cycle into the bus cycles during access to the external space.
	$\overline{\text{WE0}}$	O	Byte select	Indicates a write access to bits 7 to 0 of data of external memory or device.
	$\overline{\text{WE1}}$	O	Byte select	Indicates a write access to bits 15 to 8 of data of external memory or device.
	$\overline{\text{WE2}}$	O	Byte select	Indicates a write access to bits 23 to 16 of data of external memory or device.
	$\overline{\text{WE3}}$	O	Byte select	Indicates a write access to bits 31 to 24 of data of external memory or device.
	DQMLL	O	Byte select	Selects bits D7 to D0 when SDRAM is connected.
	DQMLU	O	Byte select	Selects bits D15 to D8 when SDRAM is connected.
	DQMUL	O	Byte select	Selects bits D23 to D16 when SDRAM is connected.
	DQMUU	O	Byte select	Selects bits D31 to D24 when SDRAM is connected.
	$\overline{\text{RAS}}$	O	RAS	Connected to the $\overline{\text{RAS}}$ pin when SDRAM is connected.
	$\overline{\text{CAS}}$	O	CAS	Connected to the $\overline{\text{CAS}}$ pin when SDRAM is connected.
CKE	O	CK enable	Connected to the CKE pin when SDRAM is connected.	

Classification	Symbol	I/O	Name	Function
Direct memory access controller	DREQ0	I	DMA-transfer request	Input pin to receive external requests for DMA transfer.
	DACK0	O	DMA-transfer request accept	Output pin for signals indicating acceptance of external requests from external devices.
	TEND0	O	DMA-transfer end output	Output pin for DMA transfer end.
Multi-function timer pulse unit 2	TCLKA, TCLKB, TCLKC, TCLKD	I	Timer clock input	External clock input pins for the timer.
	TIOC0A, TIOC0B, TIOC0C, TIOC0D	I/O	Input capture/output compare (channel 0)	The TGRA_0 to TGRD_0 input capture input/output compare output/PWM output pins.
	TIOC1A, TIOC1B	I/O	Input capture/output compare (channel 1)	The TGRA_1 and TGRB_1 input capture input/output compare output/PWM output pins.
	TIOC2A, TIOC2B	I/O	Input capture/output compare (channel 2)	The TGRA_2 and TGRB_2 input capture input/output compare output/PWM output pins.
	TIOC3A, TIOC3B, TIOC3C, TIOC3D	I/O	Input capture/output compare (channel 3)	The TGRA_3 to TGRD_3 input capture input/output compare output/PWM output pins.
	TIOC4A, TIOC4B, TIOC4C, TIOC4D	I/O	Input capture/output compare (channel 4)	The TGRA_4 to TGRD_4 input capture input/output compare output/PWM output pins.
	Realtime clock	RTC_X1	I	Crystal resonator for realtime clock/external clock
RTC_X2		O		
RTC_X3		I	Crystal resonator for realtime clock/external clock	Connected to 4-MHz crystal resonator. The RTC_X3 pin can also be used to input an external clock.
RTC_X4		O		
Serial communication interface with FIFO	TxD7 to TxD0	O	Transmit data	Data output pins.
	RxD7 to RxD0	I	Receive data	Data input pins.
	SCK7 to SCK0	I/O	Serial clock	Clock input/output pins.
	$\overline{\text{RTS}}7, \overline{\text{RTS}}5, \overline{\text{RTS}}1$	O	Transmit request	Modem control pins.
	$\overline{\text{CTS}}7, \overline{\text{CTS}}5, \overline{\text{CTS}}1$	I	Transmit enable	Modem control pins.
Serial communication interface	SCI_SCK1, SCI_SCK0	I/O	Serial clock	Clock input/output pins.
	SCI_TXD1, SCI_TXD0	O	Transmit data	Data output pins.
	SCI_RXD1, SCI_RXD0	I	Receive data	Data input pins.
	$\overline{\text{SCI\_CTS}}1/\overline{\text{RTS}}1, \overline{\text{SCI\_CTS}}0/\overline{\text{RTS}}0$	I/O	Transmit and receive start control	I/O pins for controlling the start of transmission and reception.
Renesas serial peripheral interface	MOSI4 to MOSI0	I/O	Data	Data I/O pins.
	MISO4 to MISO0	I/O	Data	Data I/O pins.
	RSPCK4 to RSPCK0	I/O	Clock	Clock I/O pins.
	SSL40 to SSL10	I/O	Slave select	Slave select I/O pins.

Classification	Symbol	I/O	Name	Function
SPI multi I/O bus controller	SPBCLK_1, SPBCLK_0	O	Clock	Clock output pins.
	SPBSSL_1, SPBSSL_0	O	Slave select	Slave select output pins.
	SPBMO0_0/SPBIO00_0, SPBMO0_0/SPBIO10_0, SPBIO20_0, SPBIO30_0, SPBMO1_0/SPBIO01_0, SPBMO1_0/SPBIO11_0, SPBIO21_0, SPBIO31_0	I/O	Data	Data I/O pins for channel 0.
	SPBMO0_1/SPBIO00_1, SPBMO0_1/SPBIO10_1, SPBIO20_1, SPBIO30_1, SPBMO1_1/SPBIO01_1, SPBMO1_1/SPBIO11_1, SPBIO21_1, SPBIO31_1	I/O	Data	Data I/O pins for channel 1.
I <sup>2</sup> C bus interface	RIIC3SCL to RIIC0SCL	I/O	Serial clock pin	Serial clock I/O pins.
	RIIC3SDA to RIIC0SDA	I/O	Serial data pin	Serial data I/O pins.
Serial sound interface	SSITxD5, SSITxD3, SSITxD1, SSITxD0	O	Data output	Serial data output pin.
	SSIRxD5, SSIRxD3, SSIRxD1, SSIRxD0	I	Data input	Serial data input pin.
	SSIDATA4, SSIDATA2	I/O	Data I/O	Serial data I/O pins.
	SSISCK5 to SSISCK0	I/O	SSI clock I/O	I/O pins for serial clocks.
	SSIWS5 to SSIWS0	I/O	SSI clock LR I/O	I/O pins for word selection.
Media local bus	MLB_CLK	I	Clock input	MediaLB clock input pin.
	MLB_SIG	I/O	Signal information I/O	MediaLB signal information I/O pin.
	MLB_DAT	I/O	Data I/O	MediaLB data I/O pin.
CAN interface	CAN_CLK	I	Clock source for CAN communication	Clock source for CAN communication.
	CAN4TX to CAN0TX	O	CAN bus transmit data	Output pins for transmit data on the CAN bus.
	CAN4RX to CAN0RX	I	CAN bus receive data	Input pins for receive data on the CAN bus.
IEBus™ controller	IETxD	O	IEBus™ controller transmit data	Output pin for transmit data on IEBus™ controller.
	IERxD	I	IEBus™ controller receive data	Input pin for receive data on IEBus™ controller.
Renesas SPDIF interface	SPDIF_OUT	O	Output data	Transmit data output pin.
	SPDIF_IN	I	Input data	Receive data input pin.
LIN interface	RLIN31TX, RLIN30TX	O	Output data	Transmit data output pins.
	RLIN31RX, RLIN30RX	I	Input data	Receive data input pins.

Classification	Symbol	I/O	Name	Function
Ethernet controller, EthernetAVB Note: Regarding the switching of pin functions between Ethernet controller and EthernetAVB, refer to section 54, Ports.	ET_TXCLK	I	Transmit clock	Clock pin for transmission.
	ET_TXEN	O	Transmit enable	Transmit data enable pin
	ET_TXD3 to ET_TXD0	O	Transmit data	MII transmit data pins.
	ET_COL	I	Collision detection	Collision detection pin.
	ET_TXER	O	Transmit error	Transmit error output pin.
	ET_RXCLK	I	Receive clock	Receive clock pin
	ET_RXDV	I	Receive enable	Receive data enable pin
	ET_RXD3 to ET_RXD0	I	Receive data	MII receive data pins.
	ET_RXER	I	Receive error	Receive error input pin.
	ET_CRS	I	Carrier detection	Carrier detection pin.
	ET_MDC	O	Management data clock	Clock pin for information transfer via MDIO.
	ET_MDIO	I/O	Management data I/O	Bidirectional pin for exchange of management data
	EthernetAVB	AVB_CAPTURE	I	Timer capture
AVB_GPTP_EXTERN		I	gPTP timer external clock	External clock pin for gPTP timer
NAND flash memory controller	FALE	O	Flash memory address latch enable	Asserted for address output and negated for data I/O.
	$\overline{\text{FRE}}$	O	Flash memory read enable	Reads data at falling edge.
	$\overline{\text{FCE}}$	O	Flash memory chip enable	Enables the flash memory connected to this LSI.
	FCLE	O	Flash memory command latch enable	Asserted at command output.
	FRB	I	Flash memory ready/ busy	High level indicates ready state and low level indicates busy state.
	$\overline{\text{FWE}}$	O	Flash memory write enable	Flash memory latches commands, addresses, and data at falling edge.
	NAF7 to NAF0	I/O	Flash memory data	Data I/O pins.
	USB 2.0 host/ function module	DP1, DP0	I/O	USB 2.0 host/function module D+ data
DM1, DM0		I/O	USB 2.0 host/function module D- data	D- data pins for USB 2.0 host/function module bus.
VBUS1, VBUS0		I	VBUS input	Connected to Vbus on USB 2.0 host/ function module bus.
REFRIN		I	Reference input	Connected to USBAPVss via 5.6-k $\Omega$ $\pm$ 1% resistance.

Classification	Symbol	I/O	Name	Function																																															
USB 2.0 host/ function module	USB_X1	I	Crystal resonator/ external clock for USB 2.0 host/function module	Connected to a crystal resonator for USB 2.0 host/function module. An external clock signal may also be input to the USB_X1 pin.																																															
	USB_X2	O				USBAPVcc	I	Power supply for transceiver analog pins	Power supply for pins.		USBAPVss Note: This pin is not present on products in the BGA package.	I	Ground for transceiver analog pins	Ground for pins.		USBDPVcc Note: This pin is not present on products in the BGA package.	I	Power supply for transceiver digital pins	Power supply for pins.		USBDPVss Note: This pin is not present on products in the BGA package.	I	Ground for transceiver digital pins	Ground for pins.		USBAVcc	I	Power supply for transceiver analog core	Power supply for core.		USBAVss Note: This pin is not present on products in the BGA package.	I	Ground for transceiver analog core	Ground for core.		USBDVcc Note: This pin is not present on products in the BGA package.	I	Power supply for transceiver digital core	Power supply for core.		USBDVss Note: This pin is not present on products in the BGA package.	I	Ground for transceiver digital core	Ground for core.		USBUVcc Note: This pin is not present on products in the BGA package.	I	480-MHz power supply for USB 2.0 host/function module	Power supply for 480-MHz sections		USBUVss Note: This pin is not present on products in the BGA package.
	USBAPVcc	I	Power supply for transceiver analog pins	Power supply for pins.																																															
	USBAPVss Note: This pin is not present on products in the BGA package.	I	Ground for transceiver analog pins	Ground for pins.																																															
	USBDPVcc Note: This pin is not present on products in the BGA package.	I	Power supply for transceiver digital pins	Power supply for pins.																																															
	USBDPVss Note: This pin is not present on products in the BGA package.	I	Ground for transceiver digital pins	Ground for pins.																																															
	USBAVcc	I	Power supply for transceiver analog core	Power supply for core.																																															
	USBAVss Note: This pin is not present on products in the BGA package.	I	Ground for transceiver analog core	Ground for core.																																															
	USBDVcc Note: This pin is not present on products in the BGA package.	I	Power supply for transceiver digital core	Power supply for core.																																															
	USBDVss Note: This pin is not present on products in the BGA package.	I	Ground for transceiver digital core	Ground for core.																																															
	USBUVcc Note: This pin is not present on products in the BGA package.	I	480-MHz power supply for USB 2.0 host/function module	Power supply for 480-MHz sections																																															
	USBUVss Note: This pin is not present on products in the BGA package.	I	480-MHz ground for USB 2.0 host/function module	Ground for 480-MHz sections																																															

Classification	Symbol	I/O	Name	Function
Digital video decoder	VIN1A, VIN2A	I	Composite video signal (CVBS) input	Composite video signal (CVBS) channel 0 input pins.
	VIN1B, VIN2B	I	Composite video signal (CVBS) input	Composite video signal (CVBS) channel 1 input pins.
	VIDEO_X1	I	Crystal resonator/ external clock for digital video decoder	Connected to a crystal resonator for digital video decoder. An external clock signal may also be input to the VIDEO_X1 pin.
	VIDEO_X2	O		
	VRP	O	TOP reference voltage	TOP reference voltage pin for the A/D converter to input video signals. Connected to VDAVss via 0.1- $\mu$ F capacitor.
	VRM	O	BOTTOM reference voltage	BOTTOM reference voltage pin for the A/D converter to input video signals. Connected to VDAVss via 0.1- $\mu$ F capacitor.
	REXT	I	Reference voltage	Reference voltage pin for the A/D converter to input video signals. Connected to VDAVss via 22-k $\Omega$ $\pm$ 1% resistance.
	VDAVcc	I	Analog power supply	Power supply pin for the A/D converter to input video signals.
	VDAVss	I	Analog ground	Ground pin for the A/D converter to input video signals.
Video display controller 5	LCD1_DATA23 to LCD1_DATA0, LCD0_DATA23 to LCD0_DATA0	O	Output data	Data output pins for panel.
	LCD1_TCON6 to LCD1_TCON0, LCD0_TCON6 to LCD0_TCON0	O	Panel timing adjustment output	Output pins for panel timing adjustment
	LCD1_CLK, LCD0_CLK	O	Panel clock	Panel clock output pins.
	LCD1_EXTCLK, LCD0_EXTCLK	I	Panel clock source	Panel clock source input pins.
	DV0_DATA23 to DV0_DATA0, DV1_DATA7 to DV1_DATA0	I	Input data	Data input pins for graphics data.
	DV1_VSYNC, DV0_VSYNC	I	VSYNC input	VSYNC input pins.
	DV1_HSYNC, DV0_HSYNC	I	HSYNC input	HSYNC input pins.
	DV1_CLK, DV0_CLK	I	Input clock	Clock input signal pins for graphics data.
LVDS output interface	TXCLKOUTP, TXCLKOUTM	O	Output clock	LVDS differential clock output pins.
	TXOUT2P to TXOUT0P, TXOUT2M to TXOUT0M	O	Output data	LVDS differential data output pins.
	LVDSREFRIN	I	Reference input	Connected to LVDSAPVss via 5.6-k $\Omega$ $\pm$ 1% resistance.
	LVDSAPVcc	I	LVDS analog power supply	Power supply for LVDS output.
	LVDSAPVss Note: This pin is not present on products in the BGA package.	I	LVDS analog ground	Ground for LVDS output.
	LVDSPLLvcc	I	LVDS PLL power supply	Power supply for LVDS PLL.
Capture engine unit	VIO_D15 to VIO_D0	I	Input data	Graphics data input pins.
	VIO_CLK	I	Input clock	Graphics data clock input pin.
	VIO_VD	I	VSYNC input	VSYNC input pin.
	VIO_HD	I	HSYNC input	HSYNC input pin.
	VIO_FLD	I	FIELD input	Input pin for field information
Sound generator	SGOUT3 to SGOUT0	O	Sound generator output	Sound generator output pins.

Classification	Symbol	I/O	Name	Function
SD host interface	SD_CLK_0, SD_CLK_1	O	SD clock	Output pins for SD clock.
	SD_CMD_0, SD_CMD_1	I/O	SD command	SD command output and response input signals.
	SD_D3_0 to SD_D0_0, SD_D3_1 to SD_D0_1	I/O	SD data	SD data bus signals.
	SD_CD_0, SD_CD_1	I	SD card detection	SD card detection.
	SD_WP_0, SD_WP_1	I	SD write protection	SD write protection signals.
MMC host interface	MMC_CLK	O	MMC clock	Output pin for MMC clock.
	MMC_CMD	I/O	MMC command	MMC command output and response input signal.
	MMC_D7 to MMC_D0	I/O	MMC data	MMC data bus signals.
	MMC_CD	I	MMC card detection	MMC card detection.
Motor control PWM timer	PWM1H to PWM1A PWM2H to PWM2A	O	Timer output	PWM output pins.
A/D converter	AN7 to AN0	I	Analog input pins	Analog input pins.
	ADTRG	I	A/D conversion trigger input	External trigger input pin for starting A/D conversion.
	AVcc	I	Analog power supply	Power supply pin for A/D converter.
	AVss	I	Analog ground	Ground pin for A/D converter.
	AVref	I	Analog reference voltage	Reference voltage pin for A/D converter.
General I/O ports	P2_0 to P2_15, P3_0 to P3_15, P4_0 to P4_15, P5_0 to P5_10, P6_0 to P6_15, P7_0 to P7_15, P8_0 to P8_15, P9_0 to P9_7, P10_0 to P10_15, P11_0 to P11_15	I/O	General port	General I/O port pins.
	P1_0 to P1_7	I/O	General port	8 input port pins with open-drain output.
	JP0_0, JP0_1, P0_0 to P0_5, P1_8 to P1_15	I	General port	16 general input port pins.
Debugging interface	TCK/SWDCLK	I	Test clock	Test-clock input pin. Also used as the input clock pin for serial wire debugging
	TMS/SWDIO	I, I/O	Test mode select	Test-mode select signal input pin. Also used as the I/O data pin for serial wire debugging
	TDI	I	Test data input	Serial input pin for instructions and data.
	TDO	O	Test data output	Serial output pin for instructions and data.
	TRST	I	Test reset	Initialization-signal input pin.
	TRACEDATA3 to TRACEDATA0	O	Data output	Trace data output pins.
	TRACECLK	O	Clock output	Trace clock output pin.
	TRACECTL	O	Enable output	Trace enable output pin.

## 1.6 List of Pins

Table 1.4 List of Pins (256-Pin, BGA)

BGA256 No.	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Function 8		Simplified Circuit Diagram Figure 1.4	
	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O		
A1	PVcc																					
A2	P6_4	I(s)/O	—	—	D4	I(s)/O	LCD1_DA TA12	O	CAN2RX	I(s)	IRQ3	I(s)	RTS5	I(s)/O	—	—	RSPCK1	I(s)/O	DV0_DATA 20	I(s)	(8)	
A3	P6_2	I(s)/O	—	—	D2	I(s)/O	LCD1_DA TA10	O	RLIN31RX	I(s)	IRQ7	I(s)	TCLKA	I(s)	TIOC2A	I(s)/O	RxD2	I(s)	DV0_DATA 18	I(s)	(8)	
A4	P6_0	I(s)/O	—	—	D0	I(s)/O	LCD1_DA TA8	O	RLIN30RX	I(s)	DV0_CLK	I(s)	TIOC1A	I(s)/O	IRQ5	I(s)	RxD3	I(s)	DV0_DATA 16	I(s)	(8)	
A5	P9_7	I(s)/O	—	—	LCD1_DAT A23	O	SPBIO30_0	I(s)/O	SSIDATA2	I(s)/O	TIOC1A	I(s)/O	—	—	—	—	—	—	—	—	(7)	
A6	P9_4	I(s)/O	—	—	LCD1_DAT A20	O	SPBIO00_0	I(s)/O	—	—	RxD1	I(s)	—	—	—	—	—	—	—	—	(7)	
A7	P5_10	I(s)/O	—	—	WE3/ DQM/UU/ AH	O	—	—	DV0_HSY NC	I(s)	—	—	CAN1TX	O	IETxD	O	LCD1_DAT A17	O	—	—	(7)	
A8	P5_8	I(s)/O	—	—	LCD0_EXT CLK	I(s)	IRQ0	I(s)	DV1_CLK	I(s)	—	—	DV0_CLK	I(s)	CS2	O	—	—	—	—	(7)	
A9	P5_7	I(s)/O	—	—	TXOUT0M	O	LCD1_DA TA7	O	LCD0_DA TA23	O	DV1_DAT A7	I(s)	RxD6	I(s)	TIOC0D	I(s)/O	SPDIF_OU T	O	DV0_DATA 15	I(s)	(12), (13)	
A10	P5_3	I(s)/O	—	—	TXOUT2M	O	LCD1_DA TA3	O	LCD0_DA TA19	O	DV1_DAT A3	I(s)	TxD3	O	TIOC3C	I(s)/O	—	—	MISO3	I(s)/O	(12), (13)	
A11	P5_1	I(s)/O	—	—	TXCLK OUTM	O	LCD1_DA TA1	O	LCD0_DA TA17	O	DV1_DAT A1	I(s)	RxD4	I(s)	TIOC0B	I(s)/O	—	—	SSL30	I(s)/O	(12), (13)	
A12	LVDSRE- FRIN	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(12), (13)	
A13	VIN2B	I(a)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	
A14	VDAVss																					
A15	VIN1A	I(a)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	
A16	P0_2	I(s)	MD_CLK	I(s)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(3)
A17	P1_4	I(s)/ O(o)	—	—	RIIC2SCL	I(s)/ O(o)	DV0_CLK	I(s)	CAN1RX	I(s)	IRQ4	I(s)	—	—	—	—	CAN_CLK	I(s)	—	—	(9)	
A18	P2_15	I(s)/O	—	—	D31	I(s)/O	MISO0	I(s)/O	DV0_DAT A15	I(s)	SPBIO31_0	I(s)/O	CAN_CLK	I(s)	RxD0	I(s)	LCD1_DAT A15	O	IRQ1	I(s)	(8)	
A19	P2_12	I(s)/O	—	—	D28	I(s)/O	RSPCK0	I(s)/O	DV0_DAT A12	I(s)	SPBIO01_0	I(s)/O	CAN3RX	I(s)	IRQ6	I(s)	LCD1_DAT A12	O	TIOC1B	I(s)/O	(8)	
A20	PVcc																					
A21	Vss																					
B1	Vss																					
B2	PVcc																					
B3	P6_3	I(s)/O	—	—	D3	I(s)/O	LCD1_DA TA11	O	RLIN31TX	O	IRQ2	I(s)	CTS5	I(s)/O	TIOC2B	I(s)/O	TxD2	O	DV0_DATA 19	I(s)	(8)	
B4	P6_1	I(s)/O	—	—	D1	I(s)/O	LCD1_DA TA9	O	RLIN30TX	O	IRQ4	I(s)	TIOC1B	I(s)/O	SSIDATA4	I(s)/O	TxD3	O	DV0_DATA 17	I(s)	(8)	
B5	P0_3	I(s)	MD_CLKS	I(s)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(3)
B6	P9_5	I(s)/O	—	—	LCD1_DAT A21	O	SPBIO10_0	I(s)/O	SSISCK2	I(s)/O	CTS1	I(s)/O	CS4	O	—	—	—	—	—	—	(7)	
B7	P5_9	I(s)/O	—	—	WE2/ DQMUL	O	ET_MDC	O	DV0_VSY NC	I(s)	IRQ2	I(s)	CAN1RX	I(s)	IERxD	I(s)	LCD1_DAT A16	O	—	—	(7)	
B8	Vss																					(1)
B9	P5_6	I(s)/O	—	—	TXOUT0P	O	LCD1_DA TA6	O	LCD0_DA TA22	O	DV1_DAT A6	I(s)	TxD6	O	IRQ6	I(s)	SPDIF_IN	I(s)	DV0_DATA 14	I(s)	(12), (13)	
B10	P5_2	I(s)/O	—	—	TXOUT2P	O	LCD1_DA TA2	O	LCD0_DA TA18	O	DV1_DAT A2	I(s)	SCK3	I(s)/O	TIOC1B	I(s)/O	—	—	MOSI3	I(s)/O	(12), (13)	
B11	P5_0	I(s)/O	—	—	TXCLK OUTP	O	LCD1_DA TA0	O	LCD0_DA TA16	O	DV1_DAT A0	I(s)	TxD4	O	TIOC0A	I(s)/O	—	—	RSPCK3	I(s)/O	(12), (13)	
B12	Vss																					
B13	VIN1B	I(a)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	
B14	VDAVcc																					
B15	VIN2A	I(a)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	
B16	P1_5	I(s)/ O(o)	—	—	RIIC2SDA	I(s)/ O(o)	DV1_CLK	I(s)	CAN4RX	I(s)	IRQ5	I(s)	VIO_CLK	I(s)	—	—	LCD1_EXT CLK	I(s)	—	—	(9)	
B17	P1_0	I(s)/ O(o)	—	—	RIIC0SCL	I(s)/ O(o)	DV0_DAT A16	I(s)	TCLKA	I(s)	IRQ0	I(s)	VIO_VD	I(s)	DV0_VSYN C	I(s)	—	—	—	—	(9)	
B18	P2_14	I(s)/O	—	—	D30	I(s)/O	MOSI0	I(s)/O	DV0_DAT A14	I(s)	SPBIO21_0	I(s)/O	CAN4RX	I(s)	TxD0	O	LCD1_DAT A14	O	IRQ0	I(s)	(8)	
B19	PVcc																					
B20	Vss																					
B21	P2_10	I(s)/O	—	—	D26	I(s)/O	ET_RXD2	I(s)	DV0_DAT A10	I(s)	SSIRxD0	I(s)	RLIN30TX	O	LCD1_DAT A10	O	VIO_D10	I(s)	MOSI4	I(s)/O	(8)	
C1	Vcc																					
C2	Vss																					
C20	P2_11	I(s)/O	—	—	D27	I(s)/O	ET_RXD3	I(s)	DV0_DAT A11	I(s)	SSITxD0	O	TIOC1A	I(s)/O	LCD1_DAT A11	O	VIO_D11	I(s)	MISO4	I(s)/O	(8)	
C21	P2_9	I(s)/O	—	—	D25	I(s)/O	ET_RXD1	I(s)	DV0_DAT A9	I(s)	SSISW0	I(s)/O	RLIN30RX	I(s)	LCD1_DAT A9	O	VIO_D9	I(s)	SSL40	I(s)/O	(8)	



BGA256 No.	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Function 8		Simplified Circuit Diagram Figure 1.4	
	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O		
D1	P6_8	I(s)O	—	—	D8	I(s)O	DV0_DAT A12	I(s)	—	—	CAN_CLK	I(s)	SCK0	I(s)O	LCD0_DAT A0	O	—	—	—	—	—	(8)
D2	Vcc																					
D4	PVcc																					
D5	P6_6	I(s)O	—	—	LCD1_DAT A22	O	SPBIO20_0	I(s)O	SSWS2	I(s)O	RTS1	I(s)O	CS5	O	—	—	—	—	—	—	—	(7)
D6	P6_3	I(s)O	—	—	LCD1_DAT A19	O	SPBSSL_0	O	—	—	TxD1	O	—	—	—	—	—	—	—	—	—	(7)
D7	LVD SAPVcc																					
D8	LVD SPLLVcc																					
D9	P5_5	I(s)O	—	—	TXOUT1M	O	LCD1_DA TA5	O	LCD0_DA TA21	O	DV1_DAT A5	I(s)	AUDIO_X OUT	O	TIOC0C	I(s)O	FCE	O	DV0_DATA 13	I(s)	(12), (13)	
D10	P5_4	I(s)O	—	—	TXOUT1P	O	LCD1_DA TA4	O	LCD0_DA TA20	O	DV1_DAT A4	I(s)	RxD3	I(s)	TIOC3D	I(s)O	—	—	DV0_DATA 12	I(s)	(12), (13)	
D11	VRP																					
D12	REXT																					
D13	P1_7	I(s)O	—	—	RIIC3SDA	I(s)O	DV1_HSY NC	I(s)	RLIN30RX	I(s)	IRQ7	I(s)	VIO_D13	I(s)	DV0_DATA 13	I(s)	—	—	—	—	—	(9)
D14	P1_3	I(s)O	—	—	RIIC1SDA	I(s)O	DV0_DAT A19	I(s)	ET_COL	I(s)	IRQ3	I(s)	ADTRG	I(s)	—	—	—	—	—	—	—	(9)
D15	P1_1	I(s)O	—	—	RIIC0SDA	I(s)O	DV0_DAT A17	I(s)	TCLKC	I(s)	IRQ1	I(s)	VIO_HD	I(s)	DV0_HSYN C	I(s)	—	—	—	—	—	(9)
D16	P2_13	I(s)O	—	—	D29	I(s)O	SSL00	I(s)O	DV0_DAT A13	I(s)	SPBIO11_0	I(s)O	CANSTX	O	SCK0	I(s)O	LCD1_DAT A13	O	IRQ7	I(s)	(8)	
D17	PVcc																					
D18	Vss																					
D20	P2_7	I(s)O	—	—	D23	I(s)O	ET_TXD3	O	DV0_DAT A7	I(s)	SSITXD5	O	IETxD	O	RTS1	I(s)O	VIO_D7	I(s)	LCD0_DAT A23	O	(8)	
D21	P2_8	I(s)O	—	—	D24	I(s)O	ET_RXD0	I(s)	DV0_DAT A8	I(s)	SSISCK0	I(s)O	LCD0_TC ON6	O	LCD1_DAT A8	O	VIO_D8	I(s)	RSPCK4	I(s)O	(8)	
E1	P6_9	I(s)O	—	—	D9	I(s)O	DV0_DAT A13	I(s)	—	—	—	—	TxD0	O	LCD0_DAT A1	O	—	—	—	—	—	(8)
E2	P6_7	I(s)O	—	—	D7	I(s)O	LCD1_DA TA15	O	—	—	LCD0_TC ON6	O	RxD5	I(s)	—	—	MISO1	I(s)O	DV0_DATA 23	O	(8)	
E4	Vss																					
E5	PVcc																					
E6	P2_2	I(s)O	—	—	LCD1_DAT A18	O	SPBCLK_0	O	RLIN30TX	O	SCK1	I(s)O	A0	O	—	—	—	—	—	—	—	(7)
E7	LVD SAPVcc																					
B8	LVD SAPVcc																					
E9	Vss																					
E10	Vss																					
E11	Vcc																					
E12	VRM																					
E13	P1_6	I(s)O	—	—	RIIC3SCL	I(s)O	DV1_VSY NC	I(s)	IERxD	I(s)	IRQ6	I(s)	VIO_D12	I(s)	DV0_DATA 12	I(s)	—	—	—	—	—	(9)
E14	P1_2	I(s)O	—	—	RIIC1SCL	I(s)O	DV0_DAT A18	I(s)	FRB	I(s)	IRQ2	I(s)	—	—	—	—	LCD1_EXT CLK	I(s)	—	—	—	(9)
E15	Vss																					
E16	PVcc																					
E17	Vss																					
E18	P2_4	I(s)O	—	—	D20	I(s)O	ET_TXD0	O	DV0_DAT A4	I(s)	SSISCK5	I(s)O	SPBCLK_1	O	SCK1	I(s)O	VIO_D4	I(s)	LCD0_DAT A20	O	(8)	
E20	P2_5	I(s)O	—	—	D21	I(s)O	ET_TXD1	O	DV0_DAT A5	I(s)	SSWS5	I(s)O	SPBSSL_1	O	TxD1	O	VIO_D5	I(s)	LCD0_DAT A21	O	(8)	
E21	P2_6	I(s)O	—	—	D22	I(s)O	ET_TXD2	O	DV0_DAT A6	I(s)	SSIRxD5	I(s)	—	—	RxD1	I(s)	VIO_D6	I(s)	LCD0_DAT A22	O	(8)	
F1	P6_12	I(s)O	—	—	D12	I(s)O	DV0_DAT A20	I(s)	—	—	—	—	TxD1	O	LCD0_DAT A4	O	—	—	—	—	—	(8)
F2	P6_10	I(s)O	—	—	D10	I(s)O	DV0_DAT A14	I(s)	—	—	LCD0_TC ON6	O	RxD0	I(s)	LCD0_DAT A2	O	—	—	—	—	—	(8)
F4	Vcc																					
F5	Vss																					
F17	P4_14	I(s)O	—	—	LCD0_DAT A22	O	LCD1_TC ON1	O	SD_D3_0	I(s)O	MMC_D3	I(s)O	SPBIO21_1	I(s)O	SSIRxD3	I(s)	TxD2	O	IRQ6	I(s)	(7)	
F18	P4_15	I(s)O	—	—	LCD0_DAT A23	O	LCD1_TC ON2	O	SD_D2_0	I(s)O	MMC_D2	I(s)O	SPBIO31_1	I(s)O	SSITxD3	O	RxD2	I(s)	IRQ7	I(s)	(7)	
F20	P2_2	I(s)O	—	—	D18	I(s)O	ET_TXEN	O	DV0_DAT A2	I(s)	SPBIO20_1	I(s)O	MLB_SIG	I(s)O	TIOC2B	I(s)O	VIO_D2	I(s)	LCD0_DAT A18	O	(8)	
F21	P2_3	I(s)O	—	—	D19	I(s)O	ET_CRS	I(s)	DV0_DAT A3	I(s)	SPBIO30_1	I(s)O	IERxD	I(s)	CTS1	I(s)O	VIO_D3	I(s)	LCD0_DAT A19	O	(8)	
G1	P7_0	I(s)O	MD_BOOT 2	I(s)	CS0	O	DV0_DAT A16	I(s)	ET_MDC	O	SCK4	I(s)O	RLIN30TX	O	—	—	TIOC0A	I(s)O	—	—	—	(7)
G2	P6_11	I(s)O	—	—	D11	I(s)O	DV0_DAT A15	I(s)	—	—	LCD0_TC ON6	O	SCK1	I(s)O	LCD0_DAT A3	O	—	—	—	—	—	(8)

BGA256 No.	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Function 8		Simplified Circuit Diagram Figure 1.4
	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
G4	P6_5	I(s)O	—	—	D5	I(s)O	LCD1_DA TA13	O	CAN2TX	O	—	—	SCK5	I(s)O	—	—	SSL10	I(s)O	DV0_DATA 21	O	(8)
G5	Vcc																				
G17	Vcc																				
G18	P4_11	I(s)O	—	—	LCD0_DAT A19	O	LCD1_TC ON6	O	SD_D0_0	I(s)O	MMC_D0	I(s)O	SSITx05	O	CAN4TX	O	SCK1	I(s)O	IRQ3	I(s)	(8)
G20	P4_13	I(s)O	—	—	LCD0_DAT A21	O	LCD1_TC ON0	O	SD_CMD_0	I(s)O	MMC_CM D	I(s)O	SPBIO11_1	I(s)O	SSIWS3	I(s)O	RxD1	I(s)	IRQ5	I(s)	(7)
G21	Vss																				
H1	P7_2	I(s)O	—	—	RAS	O	DV0_DAT A18	I(s)	ET_TXER	O	RxD4	I(s)	CAN2RX	I(s)	SSIWS1	I(s)O	TIOC0C	I(s)O	—	—	(7)
H2	P6_13	I(s)O	—	—	D13	I(s)O	DV0_DAT A21	I(s)	—	—	SCK6	I(s)O	RxD1	I(s)	LCD0_DAT A5	O	—	—	IRQ5	I(s)	(8)
H4	P6_6	I(s)O	—	—	D6	I(s)O	LCD1_DA TA14	O	—	—	LCD0_TC ON5	O	TxD5	O	—	—	MOSI1	I(s)O	DV0_DATA 22	O	(8)
H5	Vss																				
H17	Vcc																				
H18	P2_1	I(s)O	—	—	D17	I(s)O	ET_TXER	O	DV0_DAT A1	I(s)	SPBIO10_1	I(s)O	MLB_DAT	I(s)O	TIOC2A	I(s)O	VIO_D1	I(s)	LCD0_DAT A17	O	(8)
H20	P4_10	I(s)O	—	—	LCD0_DAT A18	O	LCD1_TC ON5	O	SD_D1_0	I(s)O	MMC_D1	I(s)O	SSIRx05	I(s)	—	—	RxD0	I(s)	IRQ2	I(s)	(7)
H21	P4_12	I(s)O	—	—	LCD0_DAT A20	O	LCD1_CL K	O	SD_CLK_0	O	MMC_CL K	O	SPBIO11_1	I(s)O	SSISCK3	I(s)O	TxD1	O	IRQ4	I(s)	(7)
J1	P7_4	I(s)O	—	—	CKE	O	DV0_DAT A20	I(s)	ET_TXD0	O	TxD7	O	—	—	SSITxD1	O	TIOC1A	I(s)O	—	—	(7)
J2	P7_3	I(s)O	—	—	CAS	O	DV0_DAT A19	I(s)	ET_TXEN	O	SCK7	I(s)O	CAN2TX	O	SSIRxD1	I(s)	TIOC0D	I(s)O	—	—	(7)
J4	P6_14	I(s)O	—	—	D14	I(s)O	DV0_DAT A22	I(s)	—	—	TxD6	O	—	—	LCD0_DAT A6	O	—	—	IRQ6	I(s)	(8)
J5	Vss																				
J17	Vss																				
J18	P2_0	I(s)O	—	—	D16	I(s)O	ET_TXCL K	I(s)	DV0_DAT A0	I(s)	SPBIO00_1	I(s)O	MLB_CLK	I(s)	IRQ5	I(s)	VIO_D0	I(s)	LCD0_DAT A16	O	(8)
J20	P4_9	I(s)O	—	—	LCD0_DAT A17	O	LCD1_TC ON4	O	SD_WP_0	I(s)	—	—	SSIWS5	I(s)O	CAN2RX	I(s)	TxD0	O	IRQ1	I(s)	(7)
J21	P4_8	I(s)O	—	—	LCD0_DAT A16	O	LCD1_TC ON3	O	SD_CD_0	I(s)	MMC_CD	I(s)	SSISCK5	I(s)O	CAN2TX	O	SCK0	I(s)O	IRQ0	I(s)	(7)
K1	P7_8	I(s)O	—	—	RD	O	SSISCK3	I(s)O	—	—	CAN0RX	I(s)	—	—	—	—	TIOC3A	I(s)O	IRQ1	I(s)	(7)
K2	P7_6	I(s)O	—	—	WE0/ DQMLL	O	DV0_DAT A22	I(s)	ET_TXD2	O	CTS7	I(s)O	—	—	SSIWS2	I(s)O	TIOC2A	I(s)O	—	—	(7)
K4	P7_1	I(s)O	—	—	CS3	O	DV0_DAT A17	I(s)	ET_TXCL K	I(s)	TxD4	O	DV0_CLK	I(s)	SSISCK1	I(s)O	TIOC0B	I(s)O	—	—	(7)
K5	P6_15	I(s)O	—	—	D15	I(s)O	DV0_DAT A23	I(s)	—	—	RxD6	I(s)	—	—	LCD0_DAT A7	O	—	—	IRQ7	I(s)	(8)
K17	Vss																				
K18	P4_3	I(s)O	—	—	LCD0_DAT A11	O	TIOC0D	I(s)O	FWE	O	CAN3TX	O	RxD2	I(s)	—	—	MISO4	I(s)O	MMC_D7	I(s)O	(7)
K20	P4_7	I(s)O	—	—	LCD0_DAT A15	O	MISO1	I(s)O	TIOC4D	I(s)O	PWM2H	O	SSITxD0	O	—	—	DV0_DATA 15	I(s)	—	—	(7)
K21	P4_6	I(s)O	—	—	LCD0_DAT A14	O	MOSI1	I(s)O	TIOC4C	I(s)O	PWM2G	O	SSIRxD0	I(s)	—	—	DV0_DATA 14	I(s)	—	—	(7)
L1	P7_10	I(s)O	—	—	A2	O	SSIRxD3	I(s)	ET_RXD1	I(s)	CAN1TX	O	—	—	—	—	TIOC3C	I(s)O	IRQ2	I(s)	(7)
L2	P7_9	I(s)O	—	—	A1	O	SSIWS3	I(s)O	ET_RXD0	I(s)	CAN0TX	O	—	—	—	—	TIOC3B	I(s)O	IRQ0	I(s)	(7)
L4	P7_7	I(s)O	—	—	WE1/ DQMLU	O	DV0_DAT A23	I(s)	ET_TXD3	O	RTS7	I(s)O	—	—	SSIDATA2	I(s)O	TIOC2B	I(s)O	—	—	(7)
L5	P7_5	I(s)O	—	—	RDWR	O	DV0_DAT A21	I(s)	ET_TXD1	O	RxD7	I(s)	—	—	SSISCK2	I(s)O	TIOC1B	I(s)O	—	—	(7)
L17	P3_15	I(s)O	—	—	LCD0_DAT A7	O	—	—	NAF7	I(s)O	—	—	TRACECT L	O	—	—	SD_D2_1	I(s)O	MMC_D2	I(s)O	(7)
L18	P4_0	I(s)O	—	—	LCD0_DAT A8	O	TIOC0A	I(s)O	FRE	O	—	—	—	—	—	—	RSPCK4	I(s)O	MMC_D4	I(s)O	(7)
L20	P4_5	I(s)O	—	—	LCD0_DAT A13	O	SSL10	I(s)O	TIOC4B	I(s)O	PWM2F	O	SSIWS0	I(s)O	—	—	DV0_DATA 13	I(s)	—	—	(7)
L21	P4_4	I(s)O	—	—	LCD0_DAT A12	O	RSPCK1	I(s)O	TIOC4A	I(s)O	PWM2E	O	SSISCK0	I(s)O	—	—	DV0_DATA 12	I(s)	—	—	(7)
M1	P7_14	I(s)O	—	—	A6	O	SSIDATA4	I(s)O	ET_CRS	I(s)	—	—	—	—	—	—	TIOC4C	I(s)O	IRQ6	I(s)	(7)
M2	P7_13	I(s)O	—	—	A5	O	SSIWS4	I(s)O	ET_MDIO	I(s)O	—	—	—	—	—	—	TIOC4B	I(s)O	IRQ5	I(s)	(7)
M4	P7_12	I(s)O	—	—	A4	O	SSISCK4	I(s)O	ET_RXD3	I(s)	—	—	—	—	—	—	TIOC4A	I(s)O	IRQ4	I(s)	(7)
M5	P7_11	I(s)O	—	—	A3	O	SSITxD3	O	ET_RXD2	I(s)	CAN1RX	I(s)	—	—	—	—	TIOC3D	I(s)O	IRQ3	I(s)	(7)
M17	P3_10	I(s)O	—	—	LCD0_DAT A2	O	—	—	NAF2	I(s)O	—	—	TRACEDA TA2	O	TIOC4C	I(s)O	SD_D1_1	I(s)O	MMC_D1	I(s)O	(7)
M18	P3_11	I(s)O	—	—	LCD0_DAT A3	O	—	—	NAF3	I(s)O	—	—	TRACEDA TA3	O	TIOC4D	I(s)O	SD_D0_1	I(s)O	MMC_D0	I(s)O	(7)
M20	P4_2	I(s)O	—	—	LCD0_DAT A10	O	TIOC0C	I(s)O	FALE	O	CAN3RX	I(s)	TxD2	O	—	—	MOSI4	I(s)O	MMC_D6	I(s)O	(7)
M21	P4_1	I(s)O	—	—	LCD0_DAT A9	O	TIOC0B	I(s)O	FCLE	O	—	—	SCK2	I(s)O	—	—	SSL40	I(s)O	MMC_D5	I(s)O	(7)
N1	P6_1	I(s)O	—	—	A9	O	MOSI0	I(s)O	ET_RXDV	I(s)	TxD5	O	SCI_RXD0	I(s)	—	—	—	—	—	—	(7)

BGA256 No.	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Function 8		Simplified Circuit Diagram Figure 1.4
	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
N2	P8_2	I(s)O	—	—	A10	O	MISO0	I(s)O	AVB_GPT_P_EXTERN	I(s)	RxD5	I(s)	IR00	I(s)	—	—	—	—	—	—	(7)
N4	P8_0	I(s)O	—	—	A8	O	SSL00	I(s)O	ET_RXER	I(s)	SCK5	I(s)O	SCL_SCK0	I(s)O	—	—	—	—	—	—	(7)
N5	P7_15	I(s)O	—	—	A7	O	RSPCK0	I(s)O	ET_RXCLK	I(s)	CTS5	I(s)O	SCL_TXD0	O	—	—	TIOC4D	I(s)O	—	—	(7)
N17	JP0_1	I	—	—	TDO	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(6)
N18	JP0_0	I	—	—	TDI	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(2)
N20	P3_13	I(s)O	—	—	LCD0_DAT A5	O	—	—	NAF5	I(s)O	AUDIO_X_OUT	O	—	—	—	—	SD_CMD_1	I(s)O	MMC_CM D	I(s)O	(7)
N21	P3_14	I(s)O	—	—	LCD0_DAT A6	O	—	—	NAF6	I(s)O	—	—	TRACE CLK	O	—	—	SD_D3_1	I(s)O	MMC_D3	I(s)O	(7)
P1	P8_4	I(s)O	—	—	A12	O	DV1_DAT A1	I(s)	SSL20	I(s)O	—	—	—	—	—	—	IERxD	I(s)	RxD2	I(s)	(7)
P2	P8_5	I(s)O	—	—	A13	O	DV1_DAT A2	I(s)	MOSI2	I(s)O	—	—	—	—	—	—	—	—	—	—	(7)
P4	P8_3	I(s)O	—	—	A11	O	DV1_DAT A0	I(s)	RSPCK2	I(s)O	RTS5	I(s)O	—	—	—	—	IRQ1	I(s)	SCK2	I(s)O	(7)
P5	P8_6	I(s)O	—	—	A14	O	DV1_DAT A3	I(s)	MISO2	I(s)O	—	—	—	—	—	—	IETxD	O	TxD2	O	(7)
P17	PVcc																				
P18	PVcc																				
P20	Vss																				
P21	P3_12	I(s)O	—	—	LCD0_DAT A4	O	—	—	NAF4	I(s)O	—	—	—	—	—	—	SD_CLK_1	O	MMC_CLK	O	(7)
R1	CKIO	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(5)
R2	P8_7	I(s)O	—	—	A15	O	DV1_DAT A4	I(s)	AUDIO_X_OUT	O	IRQ5	I(s)	ET_C0L	I(s)	—	—	—	—	—	—	(7)
R4	P8_10	I(s)O	—	—	A18	O	DV1_DAT A7	I(s)	SPBIO20_1	I(s)O	TIOC3A	I(s)O	CAN4TX	O	PWM1C	O	SGOUT_0	O	SSITxD5	O	(7)
R5	PVcc																				
R17	PVcc																				
R18	PVcc																				
R20	TMS	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(6)
R21	TCK	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(2)
T1	P8_8	I(s)O	—	—	A16	O	DV1_DAT A5	I(s)	SPBIO00_1	I(s)O	SPDIF_IN	I(s)	TIOC1A	I(s)O	PWM1A	O	TxD3	O	SSISCK5	I(s)O	(7)
T2	P8_9	I(s)O	—	—	A17	O	DV1_DAT A6	I(s)	SPBIO10_1	I(s)O	SPDIF_O UT	O	TIOC1B	I(s)O	PWM1B	O	RxD3	I(s)	SSIWS5	I(s)O	(7)
T4	PVcc																				
T5	Vss																				
T17	Vss																				
T18	BSCANP	I(s)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(1)
T20	P3_9	I(s)O	—	—	LCD0_DAT A1	O	—	—	NAF1	I(s)O	—	—	TRACEDA TA1	O	TIOC4B	I(s)O	SD_WP_1	I(s)	IR06	I(s)	(7)
T21	TRST	I(s)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(3)
U1	P8_11	I(s)O	—	—	A19	O	—	—	SPBIO30_1	I(s)O	TIOC3B	I(s)O	RxD5	I(s)	PWM1D	O	SGOUT_1	O	DV0_CLK	I(s)	(7)
U2	P8_12	I(s)O	—	—	A20	O	—	—	SPBCLK_1	O	TIOC3C	I(s)O	SCK5	I(s)O	PWM1E	O	SGOUT_2	O	SSISCK4	I(s)O	(7)
U4	PVcc																				
U5	Vss																				
U6	Vcc																				
U7	Vss																				
U8	P3_3	I(s)O	—	—	LCD0_TC ON2	O	ET_MDIO	I(s)O	IRQ4	I(s)	BS	O	SGI_CTS1/RTS1	I(s)O	DACK0	O	PWM2D	O	MISO3	I(s)O	(7)
U9	RES	I(s)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(1)
U10	Vss																				
U11	PVcc																				
U12	PVcc																				
U13	USBAPVcc																				
U14	PLLvcc																				
U15	Vss																				
U16	Vcc																				
U17	Vcc																				
U18	Vss																				
U20	AUDIO_X2	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(10)
U21	P3_8	I(s)O	—	—	LCD0_DAT A0	O	—	—	NAF0	I(s)O	—	—	TRACEDA TA0	O	TIOC4A	I(s)O	SD_CD_1	I(s)	MMC_CD	I(s)	(7)
V1	P8_13	I(s)O	—	—	A21	O	—	—	SPBSSL_1	O	TIOC3D	I(s)O	TxD5	O	PWM1F	O	SGOUT_3	O	SSIWS4	I(s)O	(7)
V2	P8_14	I(s)O	—	—	A22	O	SPBIO01_0	I(s)O	SPBIO00_1	I(s)O	TIOC2A	I(s)O	RSPCK2	I(s)O	PWM1G	O	TxD4	O	SSIDATA4	I(s)O	(7)
V4	Vss																				

BGA256 No.	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Function 8		Simplified Circuit Diagram Figure 1.4	
	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O		
V5	Vcc																					
V6	P3_6	I(s)/O	--	--	LCDD_TC ON5	O	ET_RXDV	I(s)	SSIRXD1	I(s)	--	--	SCL_RXD0	I(s)	TIOC3C	I(s)/O	RxD3	I(s)	--	--	(7)	
V7	P3_5	I(s)/O	--	--	LCDD_TC ON4	O	ET_RXER	I(s)	SSIWS1	I(s)/O	AUDIO_X OUT3	O	SCL_TXD0	O	TIOC3B	I(s)/O	TxD3	O	--	--	(7)	
V8	P3_1	I(s)/O	--	--	LCDD_TC ON0	O	ET_TXER	O	IRQ6	I(s)	TxD2	O	SCL_TXD1	O	AUDIO_CLK	I(s)	PWM2B	O	SSL30	I(s)/O	(7)	
V9	NMI	I(s)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	(3)	
V10	P0_4	I(s)	--	--	RTC_X3	I	--	--	--	--	--	--	--	--	--	--	--	--	--	--	(3),(10)	
V11	P0_5	I(s)	--	--	RTC_X4	O	--	--	--	--	--	--	--	--	--	--	--	--	--	--	(3),(10)	
V12	REFRIN	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	
V13	USBAVcc																					
V14	P0_0	I(s)	MD_BOOT 0	I(s)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	(3)	
V15	P1_9	I(s)	--	--	AN1	I(a)	--	--	IRQ3	I(s)	--	--	VIO_D15	I(s)	DVO_DATA 15	I(s)	--	--	--	--	(4)	
V16	P1_10	I(s)	--	--	AN2	I(a)	--	--	IRQ4	I(s)	TCLKB	I(s)	--	--	--	--	--	--	--	--	(4)	
V17	P1_11	I(s)	--	--	AN3	I(a)	--	--	IRQ5	I(s)	TCLKD	I(s)	--	--	--	--	--	--	--	--	(4)	
V18	Vcc																					
V20	VIDEO_X2	O	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	(10)	
V21	AUDIO_X1	I	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	(10)	
W1	P8_15	I(s)/O	--	--	A23	O	SPBIO11_0	I(s)/O	SPBIO10_1	I(s)/O	TIOC2B	I(s)/O	SSL20	I(s)/O	PWM1H	O	RxD4	I(s)	--	--	(7)	
W2	PVcc																					
W20	Vss																					
W21	VIDEO_X1	I	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	(10)	
Y1	PVcc																					
Y2	Vss																					
Y3	Vcc																					
Y4	P9_0	I(s)/O	--	--	A24	O	SPBIO21_0	I(s)/O	CAN0TX	O	TCLKC	I(s)	MOSI2	I(s)/O	--	--	--	--	--	--	(7)	
Y5	P3_4	I(s)/O	--	--	LCDD_TC ON3	O	ET_RXCLK	I(s)	SSISCK1	I(s)/O	AUDIO_X OUT2	O	SCL_SCK0	I(s)/O	TIOC3A	I(s)/O	SCK3	I(s)/O	--	--	(7)	
Y6	Vss																					
Y7	RTC_X2	O	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	(11)	
Y8	Vss																					
Y9	DP1	I/O	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	
Y10	VBUS1	I	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	
Y11	DP0	I/O	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	
Y12	VBUS0	I	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	
Y13	USB_X2	O	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	(10)	
Y14	Vss																					
Y15	P0_1	I(s)	MD_BOOT 1	I(s)	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	(3)	
Y16	P1_8	I(s)	--	--	AN0	I(a)	--	--	IRQ2	I(s)	DREQ0	I(s)	VIO_D14	I(s)	DVO_DATA 14	I(s)	--	--	--	--	(4)	
Y17	AVcc																					
Y18	P1_12	I(s)	--	--	AN4	I(a)	DVO_VSY NC	I(s)	--	--	VIO_FLD	I(s)	--	--	--	--	--	--	--	--	(4)	
Y19	P1_14	I(s)	--	--	AN6	I(a)	--	--	--	--	ET_COL	I(s)	--	--	--	--	--	--	--	--	(4)	
Y20	Vcc																					
Y21	Vss																					
AA1	Vss																					
AA2	Vcc																					
AA3	P9_1	I(s)/O	--	--	A25	O	SPBIO31_0	I(s)/O	CAN0RX	I(s)	IRQ0	I(s)	MISO2	I(s)/O	--	--	--	--	--	--	(7)	
AA4	P3_7	I(s)/O	--	--	LCDD_TC ON6	O	--	--	SSITxD1	O	LCD1_EX TCLK	I(s)	SCL_CIS0/ RTS0	I(s)/O	TIOC3D	I(s)/O	CST	O	WDTOVF	O	(7)	
AA5	P3_2	I(s)/O	--	--	LCDD_TC ON1	O	ET_TXEN	O	--	--	RxD2	I(s)	SCL_RXD1	I(s)	TEND0	O	PWM2C	O	MOSI3	I(s)/O	(7)	
AA6	P3_0	I(s)/O	--	--	LCDD_CLK	O	ET_TXCLK	I(s)	IRQ2	I(s)	SCK2	I(s)/O	SCL_SCK1	I(s)/O	TxD2	O	PWM2A	O	RSPCK3	I(s)/O	(7)	
AA7	RTC_X1	I	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	(11)	
AA8	Vss																					
AA9	DM1	I/O	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	
AA10	Vss																					
AA11	DM0	I/O	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	
AA12	Vss																					
AA13	USB_X1	I	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	(10)	
AA14	EXTAL	I	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	(10)	
AA15	XTAL	O	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	(10)	

BGA256 No.	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Function 8		Simplified Circuit Diagram Figure 1.4
	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
AA16	Vss																				
AA17	AVss																				
AA18	AVref																				
AA19	P1_13	I(s)	—	—	AN5	I(a)	DVD_HSY NC	I(s)	—	—	WAIT	I(s)	—	—	—	—	—	—	—	—	(4)
AA20	P1_15	I(s)	—	—	AN7	I(a)	—	—	—	—	AVB_CAP TURE	I(s)	—	—	—	—	—	—	—	—	(4)
AA21	Vcc																				

## [Legend]

(s): Schmitt

(a): Analog

(o): Open drain

Table 1.5 List of Pins (256-Pin, QFP)

No.	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Function 8		Simplified Circuit Diagram Figure 1.4
	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
1	P6_5	I(s)/O	—	—	D5	I(s)/O	LCD1_ DATA13	O	CAN2TX	O	—	—	SCK5	I(s)/O	—	—	SSL10	I(s)/O	DV0_ DATA21	O	(8)
2	P6_6	I(s)/O	—	—	D6	I(s)/O	LCD1_ DATA14	O	—	—	LCD0_ TCON5	O	TxD5	O	—	—	MOSI1	I(s)/O	DV0_ DATA22	O	(8)
3	P6_7	I(s)/O	—	—	D7	I(s)/O	LCD1_ DATA15	O	—	—	LCD0_ TCON6	O	RxD5	I(s)	—	—	MISO1	I(s)/O	DV0_ DATA23	O	(8)
4	Vcc																				
5	P6_8	I(s)/O	—	—	D8	I(s)/O	DV0_ DATA12	I(s)	—	—	CAN_CLK	I(s)	SCK0	I(s)/O	LCD0_ DATA0	O	—	—	IRQ0	I(s)	(8)
6	Vss																				
7	P6_9	I(s)/O	—	—	D9	I(s)/O	DV0_ DATA13	I(s)	—	—	—	—	TxD0	O	LCD0_ DATA1	O	—	—	IRQ1	I(s)	(8)
8	PVcc																				
9	P6_10	I(s)/O	—	—	D10	I(s)/O	DV0_ DATA14	I(s)	—	—	LCD0_ TCON5	O	RxD0	I(s)	LCD0_ DATA2	O	—	—	IRQ2	I(s)	(8)
10	P6_11	I(s)/O	—	—	D11	I(s)/O	DV0_ DATA15	I(s)	—	—	LCD0_ TCON6	O	SCK1	I(s)/O	LCD0_ DATA3	O	—	—	IRQ3	I(s)	(8)
11	P6_12	I(s)/O	—	—	D12	I(s)/O	DV0_ DATA20	I(s)	—	—	—	—	TxD1	O	LCD0_ DATA4	O	—	—	IRQ4	I(s)	(8)
12	P6_13	I(s)/O	—	—	D13	I(s)/O	DV0_ DATA21	I(s)	—	—	SCK6	I(s)/O	RxD1	I(s)	LCD0_ DATA5	O	—	—	IRQ5	I(s)	(8)
13	P6_14	I(s)/O	—	—	D14	I(s)/O	DV0_ DATA22	I(s)	—	—	TxD6	O	—	—	LCD0_ DATA6	O	—	—	IRQ6	I(s)	(8)
14	P6_15	I(s)/O	—	—	D15	I(s)/O	DV0_ DATA23	I(s)	—	—	RxD6	I(s)	—	—	LCD0_ DATA7	O	—	—	IRQ7	I(s)	(8)
15	P7_0	I(s)/O	MD_BOOT2	I(s)	CS0	O	DV0_ DATA16	I(s)	ET_MDC	O	SCK4	I(s)/O	RLIN30TX	O	—	—	TIOC0A	I(s)/O	—	—	(7)
16	Vss																				
17	P7_1	I(s)/O	—	—	CS3	O	DV0_ DATA17	I(s)	ET_TXCLK	I(s)	TxD4	O	DV0_CLK	I(s)	SSISCK1	I(s)/O	TIOC0B	I(s)/O	—	—	(7)
18	Vcc																				
19	P7_2	I(s)/O	—	—	RAS	O	DV0_ DATA18	I(s)	ET_TXER	O	RxD4	I(s)	CAN2RX	I(s)	SSIWS1	I(s)/O	TIOC0C	I(s)/O	—	—	(7)
20	Vss																				
21	P7_3	I(s)/O	—	—	CAS	O	DV0_ DATA19	I(s)	ET_TXEN	O	SCK7	I(s)/O	CAN2TX	O	SSIRxD1	I(s)	TIOC0D	I(s)/O	—	—	(7)
22	PVcc																				
23	P7_4	I(s)/O	—	—	CKE	O	DV0_ DATA20	I(s)	ET_TXD0	O	TxD7	O	—	—	SSITxD1	O	TIOC1A	I(s)/O	—	—	(7)
24	P7_5	I(s)/O	—	—	RD/WR	O	DV0_ DATA21	I(s)	ET_TXD1	O	RxD7	I(s)	—	—	SSISCK2	I(s)/O	TIOC1B	I(s)/O	—	—	(7)
25	P7_6	I(s)/O	—	—	WE0/ DQMLL	O	DV0_ DATA22	I(s)	ET_TXD2	O	CTS7	I(s)/O	—	—	SSIWS2	I(s)/O	TIOC2A	I(s)/O	—	—	(7)
26	P7_7	I(s)/O	—	—	WE1/ DQMLU	O	DV0_ DATA23	I(s)	ET_TXD3	O	RTS7	I(s)/O	—	—	SSIDATA2	I(s)/O	TIOC2B	I(s)/O	—	—	(7)
27	P7_8	I(s)/O	—	—	RD	O	SSISCK3	I(s)/O	—	—	CAN0RX	I(s)	—	—	—	—	TIOC3A	I(s)/O	IRQ1	I(s)	(7)
28	P7_9	I(s)/O	—	—	A1	O	SSIWS3	I(s)/O	ET_RXD0	I(s)	CAN0TX	O	—	—	—	—	TIOC3B	I(s)/O	IRQ0	I(s)	(7)
29	Vcc																				
30	P7_10	I(s)/O	—	—	A2	O	SSIRxD3	I(s)	ET_RXD1	I(s)	CAN1TX	O	—	—	—	—	TIOC3C	I(s)/O	IRQ2	I(s)	(7)
31	Vss																				
32	P7_11	I(s)/O	—	—	A3	O	SSITxD3	O	ET_RXD2	I(s)	CAN1RX	I(s)	—	—	—	—	TIOC3D	I(s)/O	IRQ3	I(s)	(7)
33	Vcc																				
34	P7_12	I(s)/O	—	—	A4	O	SSISCK4	I(s)/O	ET_RXD3	I(s)	—	—	—	—	—	—	TIOC4A	I(s)/O	IRQ4	I(s)	(7)
35	Vss																				
36	P7_13	I(s)/O	—	—	A5	O	SSIWS4	I(s)/O	ET_MDIO	I(s)/O	—	—	—	—	—	—	TIOC4B	I(s)/O	IRQ5	I(s)	(7)
37	PVcc																				
38	P7_14	I(s)/O	—	—	A6	O	SSIDATA4	I(s)/O	ET_CRS	I(s)	—	—	—	—	—	—	TIOC4C	I(s)/O	IRQ6	I(s)	(7)
39	P7_15	I(s)/O	—	—	A7	O	RSPCK0	I(s)/O	ET_RXCLK	I(s)	CTS5	I(s)/O	SCI_TXD0	O	—	—	TIOC4D	I(s)/O	—	—	(7)
40	P8_0	I(s)/O	—	—	A8	O	SSL00	I(s)/O	ET_RXER	I(s)	SCK5	I(s)/O	SCI_SCK0	I(s)/O	—	—	—	—	—	—	(7)

No.	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Function 8		Simplified Circuit Diagram Figure 1.4
	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
41	P8_1	I(s)/O	—	—	A9	O	MOSI0	I(s)/O	ET_RXDV	I(s)	TxD5	O	SCI_RXD0	I(s)	—	—	—	—	—	—	(7)
42	P8_2	I(s)/O	—	—	A10	O	MISO0	I(s)/O	AVB_GPTP EXTERN	I(s)	RxD5	I(s)	IRQ0	I(s)	—	—	—	—	—	—	(7)
43	P8_3	I(s)/O	—	—	A11	O	DV1 DATA0	I(s)	RSPCK2	I(s)/O	RTS5	I(s)/O	—	—	IRQ1	I(s)	SCK2	I(s)/O	—	—	(7)
44	P8_4	I(s)/O	—	—	A12	O	DV1 DATA1	I(s)	SSL20	I(s)/O	—	—	—	—	IERxD	I(s)	RxD2	I(s)	—	—	(7)
45	P8_5	I(s)/O	—	—	A13	O	DV1 DATA2	I(s)	MOSI2	I(s)/O	—	—	—	—	—	—	—	—	—	—	(7)
46	PVcc																				
47	P8_6	I(s)/O	—	—	A14	O	DV1 DATA3	I(s)	MISO2	I(s)/O	—	—	—	—	IETxD	O	TxD2	O	—	—	(7)
48	Vss																				
49	CKIO	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(5)
50	Vcc																				
51	P8_7	I(s)/O	—	—	A15	O	DV1 DATA4	I(s)	AUDIO_ XOUT	O	IRQ5	I(s)	ET_COL	I(s)	—	—	—	—	—	—	(7)
52	Vss																				
53	Vss																				
54	P8_8	I(s)/O	—	—	A16	O	DV1 DATA5	I(s)	SPBIO0_1	I(s)/O	SPDIF_IN	I(s)	TIOC1A	I(s)/O	PWM1A	O	TxD3	O	SSISCK5	I(s)/O	(7)
55	P8_9	I(s)/O	—	—	A17	O	DV1 DATA6	I(s)	SPBIO10_1	I(s)/O	SPDIF_OUT	O	TIOC1B	I(s)/O	PWM1B	O	RxD3	I(s)	SSIWS5	I(s)/O	(7)
56	P8_10	I(s)/O	—	—	A18	O	DV1 DATA7	I(s)	SPBIO20_1	I(s)/O	TIOC3A	I(s)/O	CAN4TX	O	PWM1C	O	SGOUT_0	O	SSITxD5	O	(7)
57	P8_11	I(s)/O	—	—	A19	O	—	—	SPBIO30_1	I(s)/O	TIOC3B	I(s)/O	RxD5	I(s)	PWM1D	O	SGOUT_1	O	DV0_CLK	I(s)	(7)
58	P8_12	I(s)/O	—	—	A20	O	—	—	SPBCLK_1	O	TIOC3C	I(s)/O	SCK5	I(s)/O	PWM1E	O	SGOUT_2	O	SSISCK4	I(s)/O	(7)
59	P8_13	I(s)/O	—	—	A21	O	—	—	SPBSSL_1	O	TIOC3D	I(s)/O	TxD5	O	PWM1F	O	SGOUT_3	O	SSIWS4	I(s)/O	(7)
60	PVcc																				
61	P8_14	I(s)/O	—	—	A22	O	SPBIO01_0	I(s)/O	SPBIO00_1	I(s)/O	TIOC2A	I(s)/O	RSPCK2	I(s)/O	PWM1G	O	TxD4	O	SSIDATA4	I(s)/O	(7)
62	Vss																				
63	P8_15	I(s)/O	—	—	A23	O	SPBIO11_0	I(s)/O	SPBIO10_1	I(s)/O	TIOC2B	I(s)/O	SSL20	I(s)/O	PWM1H	O	RxD4	I(s)	—	—	(7)
64	Vcc																				
65	P9_0	I(s)/O	—	—	A24	O	SPBIO21_0	I(s)/O	CAN0TX	O	TCLKC	I(s)	MOSI2	I(s)/O	—	—	—	—	—	—	(7)
66	P9_1	I(s)/O	—	—	A25	O	SPBIO31_0	I(s)/O	CAN0RX	I(s)	IRQ0	I(s)	MISO2	I(s)/O	—	—	—	—	—	—	(7)
67	P3_7	I(s)/O	—	—	LCD0_ TCON6	O	—	—	SSITxD1	O	LCD1_ EXTCLK	I(s)	SCI_CTS0/ RTS0	I(s)/O	TIOC3D	I(s)/O	CS1	O	WDTOVF	O	(7)
68	P3_6	I(s)/O	—	—	LCD0_ TCON5	O	ET_RXDV	I(s)	SSIRxD1	I(s)	—	—	SCI_RXD0	I(s)	TIOC3C	I(s)/O	RxD3	I(s)	—	—	(7)
69	P3_5	I(s)/O	—	—	LCD0_ TCON4	O	ET_RXER	I(s)	SSIWS1	I(s)/O	AUDIO_ XOUT3	O	SCI_TXD0	O	TIOC3B	I(s)/O	TxD3	O	—	—	(7)
70	P3_4	I(s)/O	—	—	LCD0_ TCON3	O	ET_RXCLK	I(s)	SSISCK1	I(s)/O	AUDIO_ XOUT2	O	SCI_SCK0	I(s)/O	TIOC3A	I(s)/O	SCK3	I(s)/O	—	—	(7)
71	P3_3	I(s)/O	—	—	LCD0_ TCON2	O	ET_MDIO	I(s)/O	IRQ4	I(s)	BS	O	SCI_CTS1/ RTS1	I(s)/O	DACK0	O	PWM2D	O	MISO3	I(s)/O	(7)
72	P3_2	I(s)/O	—	—	LCD0_ TCON1	O	ET_TXEN	O	—	—	RxD2	I(s)	SCI_RXD1	I(s)	TEND0	O	PWM2C	O	MOSI3	I(s)/O	(7)
73	PVcc																				
74	P3_1	I(s)/O	—	—	LCD0_ TCON0	O	ET_TXER	O	IRQ6	I(s)	TxD2	O	SCI_TXD1	O	AUDIO_CLK	I(s)	PWM2B	O	SSL30	I(s)/O	(7)
75	Vss																				
76	P3_0	I(s)/O	—	—	LCD0_CLK	O	ET_TXCLK	I(s)	IRQ2	I(s)	SCK2	I(s)/O	SCI_SCK1	I(s)/O	TxD2	O	PWM2A	O	RSPCK3	I(s)/O	(7)
77	RES	I(s)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(1)
78	NMI	I(s)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(3)
79	PVcc																				
80	RTC_X1	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(11)
81	RTC_X2	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(11)
82	Vss																				

No.	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Function 8		Simplified Circuit Diagram Figure 1.4	
	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O		
83	Vss																					
84	P0_4	I(s)	—	—	RTC_X3	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(3), (10)
85	P0_5	I(s)	—	—	RTC_X4	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(3), (10)
86	Vcc																					
87	USBDPVcc																					
88	USBDPVss																					
89	DM1	I/O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	
90	DP1	I/O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	
91	VBUS1	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	
92	USBDVcc																					
93	USBDVss																					
94	USBDPVcc																					
95	USBDPVss																					
96	DM0	I/O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	
97	DP0	I/O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	
98	VBUS0	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	
99	USBDVcc																					
100	USBDVss																					
101	REFRIN	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	
102	USBAPVss																					
103	USBAPVcc																					
104	USBVcc																					
105	USBVss																					
106	USBVcc																					
107	USBVss																					
108	USB_X1	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(10)
109	USB_X2	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(10)
110	P0_0	I(s)	MD_BOOT0	I(s)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(3)
111	Vss																					
112	EXTAL	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(10)
113	XTAL	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(10)
114	PLLVcc																					
115	PVcc																					
116	P0_1	I(s)	MD_BOOT1	I(s)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(3)
117	Vss																					
118	P1_8	I(s)	—	—	AN0	I(a)	—	—	IRQ2	I(s)	DREQ0	I(s)	VIO_D14	I(s)	DV0_DATA14	I(s)	—	—	—	—	—	(4)
119	AVcc																					
120	AVss																					
121	AVref																					
122	P1_9	I(s)	—	—	AN1	I(a)	—	—	IRQ3	I(s)	—	—	VIO_D15	I(s)	DV0_DATA15	I(s)	—	—	—	—	—	(4)
123	P1_10	I(s)	—	—	AN2	I(a)	—	—	IRQ4	I(s)	TCLKB	I(s)	—	—	—	—	—	—	—	—	—	(4)
124	P1_11	I(s)	—	—	AN3	I(a)	—	—	IRQ5	I(s)	TCLKD	I(s)	—	—	—	—	—	—	—	—	—	(4)
125	P1_12	I(s)	—	—	AN4	I(a)	DV0_VSYNC	I(s)	—	—	VIO_FLD	I(s)	—	—	—	—	—	—	—	—	—	(4)
126	P1_13	I(s)	—	—	AN5	I(a)	DV0_HSYNC	I(s)	—	—	WAIT	I(s)	—	—	—	—	—	—	—	—	—	(4)
127	P1_14	I(s)	—	—	AN6	I(a)	—	—	—	—	ET_COL	I(s)	—	—	—	—	—	—	—	—	—	(4)
128	P1_15	I(s)	—	—	AN7	I(a)	—	—	—	—	AVB_CAPTURE	I(s)	—	—	—	—	—	—	—	—	—	(4)
129	BSCANP	I(s)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(1)
130	VIDEO_X1	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(10)
131	VIDEO_X2	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(10)



No.	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Function 8		Simplified Circuit Diagram Figure 1.4
	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
132	Vss																				
133	AUDIO_X1	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(10)
134	AUDIO_X2	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(10)
135	PVcc																				
136	P3_8	I(s)/O	—	—	LCD0_ DATA0	O	—	—	NAF0	I(s)/O	—	—	TRACE DATA0	O	TIOC4A	I(s)/O	SD_CD_1	I(s)	MMC_CD	I(s)	(7)
137	Vss																				
138	P3_9	I(s)/O	—	—	LCD0_ DATA1	O	—	—	NAF1	I(s)/O	—	—	TRACE DATA1	O	TIOC4B	I(s)/O	SD_WP_1	I(s)	IRQ6	I(s)	(7)
139	Vcc																				
140	TRST	I(s)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(3)
141	JP0_1	I	—	—	TDO	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(6)
142	JP0_0	I	—	—	TDI	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(2)
143	TMS	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(6)
144	TCK	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(2)
145	P3_10	I(s)/O	—	—	LCD0_ DATA2	O	—	—	NAF2	I(s)/O	—	—	TRACE DATA2	O	TIOC4C	I(s)/O	SD_D1_1	I(s)/O	MMC_D1	I(s)/O	(7)
146	P3_11	I(s)/O	—	—	LCD0_ DATA3	O	—	—	NAF3	I(s)/O	—	—	TRACE DATA3	O	TIOC4D	I(s)/O	SD_D0_1	I(s)/O	MMC_D0	I(s)/O	(7)
147	PVcc																				
148	P3_12	I(s)/O	—	—	LCD0_ DATA4	O	—	—	NAF4	I(s)/O	—	—	—	—	—	—	SD_CLK_1	O	MMC_CLK	O	(7)
149	Vss																				
150	P3_13	I(s)/O	—	—	LCD0_ DATA5	O	—	—	NAF5	I(s)/O	AUDIO_XOUT	O	—	—	—	—	SD_CMD_1	I(s)/O	MMC_CMD	I(s)/O	(7)
151	Vcc																				
152	P3_14	I(s)/O	—	—	LCD0_ DATA6	O	—	—	NAF6	I(s)/O	—	—	TRACECLK	O	—	—	SD_D3_1	I(s)/O	MMC_D3	I(s)/O	(7)
153	P3_15	I(s)/O	—	—	LCD0_ DATA7	O	—	—	NAF7	I(s)/O	—	—	TRACECTL	O	—	—	SD_D2_1	I(s)/O	MMC_D2	I(s)/O	(7)
154	Vss																				
155	P4_0	I(s)/O	—	—	LCD0_ DATA8	O	TIOC0A	I(s)/O	FRE	O	—	—	—	—	—	—	RSPCK4	I(s)/O	MMC_D4	I(s)/O	(7)
156	P4_1	I(s)/O	—	—	LCD0_ DATA9	O	TIOC0B	I(s)/O	FCLE	O	—	—	SCK2	I(s)/O	—	—	SSL40	I(s)/O	MMC_D5	I(s)/O	(7)
157	P4_2	I(s)/O	—	—	LCD0_ DATA10	O	TIOC0C	I(s)/O	FALE	O	CAN3RX	I(s)	TxD2	O	—	—	MOSI4	I(s)/O	MMC_D6	I(s)/O	(7)
158	P4_3	I(s)/O	—	—	LCD0_ DATA11	O	TIOC0D	I(s)/O	FWE	O	CAN3TX	O	RxD2	I(s)	—	—	MISO4	I(s)/O	MMC_D7	I(s)/O	(7)
159	PVcc																				
160	P4_4	I(s)/O	—	—	LCD0_ DATA12	O	RSPCK1	I(s)/O	TIOC4A	I(s)/O	PWM2E	O	SSISCK0	I(s)/O	—	—	DV0_ DATA12	I(s)	—	—	(7)
161	Vss																				
162	P4_5	I(s)/O	—	—	LCD0_ DATA13	O	SSL10	I(s)/O	TIOC4B	I(s)/O	PWM2F	O	SSIWS0	I(s)/O	—	—	DV0_ DATA13	I(s)	—	—	(7)
163	Vcc																				
164	P4_6	I(s)/O	—	—	LCD0_ DATA14	O	MOSI1	I(s)/O	TIOC4C	I(s)/O	PWM2G	O	SSIRxD0	I(s)	—	—	DV0_ DATA14	I(s)	—	—	(7)
165	P4_7	I(s)/O	—	—	LCD0_ DATA15	O	MISO1	I(s)/O	TIOC4D	I(s)/O	PWM2H	O	SSITxD0	O	—	—	DV0_ DATA15	I(s)	—	—	(7)
166	P2_0	I(s)/O	—	—	D16	I(s)/O	ET_TXCLK	I(s)	DV0_ DATA0	I(s)	SPBIO00_1	I(s)/O	MLB_CLK	I(s)	IRQ5	I(s)	VIO_D0	I(s)	LCD0_ DATA16	O	(8)
167	P2_1	I(s)/O	—	—	D17	I(s)/O	ET_TXER	O	DV0_ DATA1	I(s)	SPBIO10_1	I(s)/O	MLB_DAT	I(s)/O	TIOC2A	I(s)/O	VIO_D1	I(s)	LCD0_ DATA17	O	(8)
168	P4_8	I(s)/O	—	—	LCD0_ DATA16	O	LCD1_ TCON3	O	SD_CD_0	I(s)	MMC_CD	I(s)	SSISCK5	I(s)/O	CAN2TX	O	SCK0	I(s)/O	IRQ0	I(s)	(7)
169	P4_9	I(s)/O	—	—	LCD0_ DATA17	O	LCD1_ TCON4	O	SD_WP_0	I(s)	—	—	SSIWS5	I(s)/O	CAN2RX	I(s)	TxD0	O	IRQ1	I(s)	(7)
170	P4_10	I(s)/O	—	—	LCD0_ DATA18	O	LCD1_ TCON5	O	SD_D1_0	I(s)/O	MMC_D1	I(s)/O	SSIRxD5	I(s)	—	—	RxD0	I(s)	IRQ2	I(s)	(7)

No.	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Function 8		Simplified Circuit Diagram Figure 1.4
	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
171	P4_11	I(s)/O	—	—	LCD0_DATA19	O	LCD1_TCON6	O	SD_D0_0	I(s)/O	MMC_D0	I(s)/O	SSITxD5	O	CAN4TX	O	SCK1	I(s)/O	IRQ3	I(s)	(7)
172	PVcc																				
173	P4_12	I(s)/O	—	—	LCD0_DATA20	O	LCD1_CLK	O	SD_CLK_0	O	MMC_CLK	O	SPBIO01_1	I(s)/O	SSISCK3	I(s)/O	TxD1	O	IRQ4	I(s)	(7)
174	Vss																				
175	P4_13	I(s)/O	—	—	LCD0_DATA21	O	LCD1_TCON0	O	SD_CMD_0	I(s)/O	MMC_CMD	I(s)/O	SPBIO11_1	I(s)/O	SSIWS3	I(s)/O	RxD1	I(s)	IRQ5	I(s)	(7)
176	Vcc																				
177	P4_14	I(s)/O	—	—	LCD0_DATA22	O	LCD1_TCON1	O	SD_D3_0	I(s)/O	MMC_D3	I(s)/O	SPBIO21_1	I(s)/O	SSIRxD3	I(s)	TxD2	O	IRQ6	I(s)	(7)
178	P4_15	I(s)/O	—	—	LCD0_DATA23	O	LCD1_TCON2	O	SD_D2_0	I(s)/O	MMC_D2	I(s)/O	SPBIO31_1	I(s)/O	SSITxD3	O	RxD2	I(s)	IRQ7	I(s)	(7)
179	P2_2	I(s)/O	—	—	D18	I(s)/O	ET_TXEN	O	DV0_DATA2	I(s)	SPBIO20_1	I(s)/O	MLB_SIG	I(s)/O	TIOC2B	I(s)/O	VIO_D2	I(s)	LCD0_DATA18	O	(8)
180	P2_3	I(s)/O	—	—	D19	I(s)/O	ET_CRS	I(s)	DV0_DATA3	I(s)	SPBIO30_1	I(s)/O	IERxD	I(s)	CTS1	I(s)/O	VIO_D3	I(s)	LCD0_DATA19	O	(8)
181	P2_4	I(s)/O	—	—	D20	I(s)/O	ET_TXD0	O	DV0_DATA4	I(s)	SSISCK5	I(s)/O	SPBCLK_1	O	SCK1	I(s)/O	VIO_D4	I(s)	LCD0_DATA20	O	(8)
182	P2_5	I(s)/O	—	—	D21	I(s)/O	ET_TXD1	O	DV0_DATA5	I(s)	SSIWS5	I(s)/O	SPBSSL_1	O	TxD1	O	VIO_D5	I(s)	LCD0_DATA21	O	(8)
183	P2_6	I(s)/O	—	—	D22	I(s)/O	ET_TXD2	O	DV0_DATA6	I(s)	SSIRxD5	I(s)	—	—	RxD1	I(s)	VIO_D6	I(s)	LCD0_DATA22	O	(8)
184	PVcc																				
185	P2_7	I(s)/O	—	—	D23	I(s)/O	ET_TXD3	O	DV0_DATA7	I(s)	SSITxD5	O	IETxD	O	RTS1	I(s)/O	VIO_D7	I(s)	LCD0_DATA23	O	(8)
186	P2_8	I(s)/O	—	—	D24	I(s)/O	ET_RXD0	I(s)	DV0_DATA8	I(s)	SSISCK0	I(s)/O	LCD0_TCON6	O	LCD1_DATA8	O	VIO_D8	I(s)	RSPCK4	I(s)/O	(8)
187	Vss																				
188	P2_9	I(s)/O	—	—	D25	I(s)/O	ET_RXD1	I(s)	DV0_DATA9	I(s)	SSIWS0	I(s)/O	RLIN30RX	I(s)	LCD1_DATA9	O	VIO_D9	I(s)	SSL40	I(s)/O	(8)
189	Vcc																				
190	P2_10	I(s)/O	—	—	D26	I(s)/O	ET_RXD2	I(s)	DV0_DATA10	I(s)	SSIRxD0	I(s)	RLIN30TX	O	LCD1_DATA10	O	VIO_D10	I(s)	MOSI4	I(s)/O	(8)
191	P2_11	I(s)/O	—	—	D27	I(s)/O	ET_RXD3	I(s)	DV0_DATA11	I(s)	SSITxD0	O	TIOC1A	I(s)/O	LCD1_DATA11	O	VIO_D11	I(s)	MISO4	I(s)/O	(8)
192	Vss																				
193	P2_12	I(s)/O	—	—	D28	I(s)/O	RSPCK0	I(s)/O	DV0_DATA12	I(s)	SPBIO01_0	I(s)/O	CAN3RX	I(s)	IRQ6	I(s)	LCD1_DATA12	O	TIOC1B	I(s)/O	(8)
194	P2_13	I(s)/O	—	—	D29	I(s)/O	SSL00	I(s)/O	DV0_DATA13	I(s)	SPBIO11_0	I(s)/O	CAN3TX	O	SCK0	I(s)/O	LCD1_DATA13	O	IRQ7	I(s)	(8)
195	P2_14	I(s)/O	—	—	D30	I(s)/O	MOSI0	I(s)/O	DV0_DATA14	I(s)	SPBIO21_0	I(s)/O	CAN4RX	I(s)	TxD0	O	LCD1_DATA14	O	IRQ0	I(s)	(8)
196	P2_15	I(s)/O	—	—	D31	I(s)/O	MISO0	I(s)/O	DV0_DATA15	I(s)	SPBIO31_0	I(s)/O	CAN_CLK	I(s)	RxD0	I(s)	LCD1_DATA15	O	IRQ1	I(s)	(8)
197	P1_0	I(s)/O(o)	—	—	RIIC0SCL	I(s)/O(o)	DV0_DATA16	I(s)	TCLKA	I(s)	IRQ0	I(s)	VIO_VD	I(s)	DV0_VSYNC	I(s)	—	—	—	—	(9)
198	P1_1	I(s)/O(o)	—	—	RIIC0SDA	I(s)/O(o)	DV0_DATA17	I(s)	TCLKC	I(s)	IRQ1	I(s)	VIO_HD	I(s)	DV0_HSYNC	I(s)	—	—	—	—	(9)
199	P1_2	I(s)/O(o)	—	—	RIIC1SCL	I(s)/O(o)	DV0_DATA18	I(s)	FRB	I(s)	IRQ2	I(s)	—	—	—	—	LCD1_EXTCLK	I(s)	—	—	(9)
200	P1_3	I(s)/O(o)	—	—	RIIC1SDA	I(s)/O(o)	DV0_DATA19	I(s)	ET_COL	I(s)	IRQ3	I(s)	ADTRG	I(s)	—	—	—	—	—	—	(9)
201	P1_4	I(s)/O(o)	—	—	RIIC2SCL	I(s)/O(o)	DV0_CLK	I(s)	CAN1RX	I(s)	IRQ4	I(s)	—	—	—	—	CAN_CLK	I(s)	—	—	(9)
202	P1_5	I(s)/O(o)	—	—	RIIC2SDA	I(s)/O(o)	DV1_CLK	I(s)	CAN4RX	I(s)	IRQ5	I(s)	VIO_CLK	I(s)	—	—	LCD1_EXTCLK	I(s)	—	—	(9)
203	P1_6	I(s)/O(o)	—	—	RIIC3SCL	I(s)/O(o)	DV1_VSYNC	I(s)	IERxD	I(s)	IRQ6	I(s)	VIO_D12	I(s)	DV0_DATA12	I(s)	—	—	—	—	(9)
204	P1_7	I(s)/O(o)	—	—	RIIC3SDA	I(s)/O(o)	DV1_HSYNC	I(s)	RLIN30RX	I(s)	IRQ7	I(s)	VIO_D13	I(s)	DV0_DATA13	I(s)	—	—	—	—	(9)
205	P0_2	I(s)	MD_CLK	I(s)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(3)
206	PVcc																				
207	Vss																				

No.	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Function 8		Simplified Circuit Diagram Figure 1.4
	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
208	VIN1A	I(a)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	
209	VIN2A	I(a)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	
210	VDAVcc																				
211	VDAVss																				
212	REXT	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	
213	VRP	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	
214	VRM	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	
215	VIN1B	I(a)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	
216	VIN2B	I(a)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	
217	PVcc																				
218	Vss																				
219	Vcc																				
220	Vss																				
221	LVDSAPVss																				
222	LVDSREFRI N	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	
223	LVDSAPVcc																				
224	P5_0	I(s)/O	—	—	TXCLK OUTP	O	LCD1_ DATA0	O	LCD0_ DATA16	O	DV1_DATA0	I(s)	TxD4	O	TIOC0A	I(s)/O	—	—	RSPCK3	I(s)/O	(12), (13)
225	LVDSAPVss																				
226	P5_1	I(s)/O	—	—	TXCLK OUTM	O	LCD1_ DATA1	O	LCD0_ DATA17	O	DV1_DATA1	I(s)	RxD4	I(s)	TIOC0B	I(s)/O	—	—	SSL30	I(s)/O	(12), (13)
227	P5_2	I(s)/O	—	—	TXOUT2P	O	LCD1_ DATA2	O	LCD0_ DATA18	O	DV1_DATA2	I(s)	SCK3	I(s)/O	TIOC1B	I(s)/O	—	—	MOSI3	I(s)/O	(12), (13)
228	LVDSAPVcc																				
229	P5_3	I(s)/O	—	—	TXOUT2M	O	LCD1_ DATA3	O	LCD0_ DATA19	O	DV1_DATA3	I(s)	TxD3	O	TIOC3C	I(s)/O	—	—	MISO3	I(s)/O	(12), (13)
230	P5_4	I(s)/O	—	—	TXOUT1P	O	LCD1_ DATA4	O	LCD0_ DATA20	O	DV1_DATA4	I(s)	RxD3	I(s)	TIOC3D	I(s)/O	—	—	DV0_ DATA12	I(s)	(12), (13)
231	LVDSAPVss																				
232	P5_5	I(s)/O	—	—	TXOUT1M	O	LCD1_ DATA5	O	LCD0_ DATA21	O	DV1_DATA5	I(s)	AUDIO_ XOUT	O	TIOC0C	I(s)/O	FCE	O	DV0_ DATA13	I(s)	(12), (13)
233	P5_6	I(s)/O	—	—	TXOUT0P	O	LCD1_ DATA6	O	LCD0_ DATA22	O	DV1_DATA6	I(s)	TxD6	O	IRQ6	I(s)	SPDIF_IN	I(s)	DV0_ DATA14	I(s)	(12), (13)
234	LVDSAPVcc																				
235	P5_7	I(s)/O	—	—	TXOUT0M	O	LCD1_ DATA7	O	LCD0_ DATA23	O	DV1_DATA7	I(s)	RxD6	I(s)	TIOC0D	I(s)/O	SPDIF_OUT	O	DV0_ DATA15	I(s)	(12), (13)
236	LVDSPLLvcc																				
237	Vss																				
238	Vss																				
239	PVcc																				
240	P5_8	I(s)/O	—	—	LCD0_ EXTCLK	I(s)	IRQ0	I(s)	DV1_CLK	I(s)	—	—	DV0_CLK	I(s)	CS2	O	—	—	—	—	(7)
241	P5_9	I(s)/O	—	—	WE2/ DQMUL	O	ET_MDC	O	DV0_ VSYNC	I(s)	IRQ2	I(s)	CAN1RX	I(s)	IERxD	I(s)	LCD1_ DATA16	O	—	—	(7)
242	P5_10	I(s)/O	—	—	WE3/ DQMUU/AH	O	—	—	DV0_ HSYNC	I(s)	—	—	CAN1TX	O	IETxD	O	LCD1_ DATA17	O	—	—	(7)
243	P9_2	I(s)/O	—	—	LCD1_ DATA18	O	SPBCLK_0	O	RLIN30TX	O	SCK1	I(s)/O	A0	O	—	—	—	—	—	—	(7)
244	P9_3	I(s)/O	—	—	LCD1_ DATA19	O	SPBSSL_0	O	—	—	TxD1	O	—	—	—	—	—	—	—	—	(7)
245	P9_4	I(s)/O	—	—	LCD1_ DATA20	O	SPBIO00_0	I(s)/O	—	—	RxD1	I(s)	—	—	—	—	—	—	—	—	(7)
246	P9_5	I(s)/O	—	—	LCD1_ DATA21	O	SPBIO10_0	I(s)/O	SSISCK2	I(s)/O	CTS1	I(s)/O	CS4	O	—	—	—	—	—	—	(7)
247	P9_6	I(s)/O	—	—	LCD1_ DATA22	O	SPBIO20_0	I(s)/O	SSISW2	I(s)/O	RTS1	I(s)/O	CS5	O	—	—	—	—	—	—	(7)
248	P9_7	I(s)/O	—	—	LCD1_ DATA23	O	SPBIO30_0	I(s)/O	SSIDATA2	I(s)/O	TIOC1A	I(s)/O	—	—	—	—	—	—	—	—	(7)

No.	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Function 8		Simplified Circuit Diagram Figure 1.4
	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
249	P0_3	I(s)	MD_CLKS	I(s)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(3)
250	Vss																				
251	PVcc																				
252	P6_0	I(s)/O	—	—	D0	I(s)/O	LCD1_ DATA8	O	RLIN30RX	I(s)	DV0_CLK	I(s)	TIOC1A	I(s)/O	IRQ5	I(s)	RxD3	I(s)	DV0_ DATA16	I(s)	(8)
253	P6_1	I(s)/O	—	—	D1	I(s)/O	LCD1_ DATA9	O	RLIN30TX	O	IRQ4	I(s)	TIOC1B	I(s)/O	SSIDATA4	I(s)/O	TxD3	O	DV0_ DATA17	I(s)	(8)
254	P6_2	I(s)/O	—	—	D2	I(s)/O	LCD1_ DATA10	O	RLIN31RX	I(s)	IRQ7	I(s)	TCLKA	I(s)	TIOC2A	I(s)/O	RxD2	I(s)	DV0_ DATA18	I(s)	(8)
255	P6_3	I(s)/O	—	—	D3	I(s)/O	LCD1_ DATA11	O	RLIN31TX	O	IRQ2	I(s)	CTS5	I(s)/O	TIOC2B	I(s)/O	TxD2	O	DV0_ DATA19	I(s)	(8)
256	P6_4	I(s)/O	—	—	D4	I(s)/O	LCD1_ DATA12	O	CAN2RX	I(s)	IRQ3	I(s)	RTS5	I(s)/O	—	—	RSPCK1	I(s)/O	DV0_ DATA20	I(s)	(8)

[Legend]

(s): Schmitt

(a): Analog

(o): Open drain

Table 1.6 List of Pins (324-Pin, BGA)

No.	Ball Number	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Function 8		Simplified Circuit Diagram Figure 1.4
		Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
1	A1	Vss																				
2	A2	P6_4	I(s)/O	—	—	D4	I(s)/O	LCD1_ DATA12	O	CAN2RX	I(s)	IRQ3	I(s)	RTS5	I(s)/O	—	—	RSPCK1	I(s)/O	DV0_ DATA20	I(s)	(8)
3	A3	P0_3	I(s)		MD_ CLKS	I(s)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(3)
4	A4	P11_0	I(s)/O	—	—	DV0_ DATA12	I(s)	TIOC4A	I(s)/O	—	—	SCK6	I(s)/O	LCD0_ DATA7	O	VIO_D12	I(s)	—	—	—	—	(7)
5	A5	P9_6	I(s)/O	—	—	LCD1_ DATA22	O	SPBIO 20_0	I(s)/O	SSIWS2	I(s)/O	RTS1	I(s)/O	CS5	O	—	—	—	—	—	—	(7)
6	A6	P9_3	I(s)/O	—	—	LCD1_ DATA19	O	SPBSSL_ 0	O	—	—	TxD1	O	—	—	—	—	—	—	—	—	(7)
7	A7	P5_9	I(s)/O	—	—	WE2/ DQMUL	O	ET_ MDC	O	DV0_ VSYNC	I(s)	IRQ2	I(s)	CAN1RX	I(s)	IERxD	I(s)	LCD1_ DATA16	O	—	—	(7)
8	A8	Vss																				
9	A9	P5_6	I(s)/O	—	—	TXOUT 0P	O	LCD1_ DATA6	O	LCD0_ DATA22	O	DV1_ DATA6	I(s)	TxD6	O	IRQ6	I(s)	SPDIF_ IN	I(s)	DV0_ DATA14	I(s)	(12), (13)
10	A10	P5_2	I(s)/O	—	—	TXOUT 2P	O	LCD1_ DATA2	O	LCD0_ DATA18	O	DV1_ DATA2	I(s)	SCK3	I(s)/O	TIOC1B	I(s)/O	—	—	MOSI3	I(s)/O	(12), (13)
11	A11	P5_0	I(s)/O	—	—	TXCLK OUTP	O	LCD1_ DATA0	O	LCD0_ DATA16	O	DV1_ DATA0	I(s)	TxD4	O	TIOC0A	I(s)/O	—	—	RSPCK3	I(s)/O	(12), (13)
12	A12	Vss																				
13	A13	VIN2B	I(a)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	
14	A14	VDAVss																				
15	A15	VIN2A	I(a)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	
16	A16	P0_2	I(s)		MD_ CLK	I(s)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(3)
17	A17	P1_6	I(s)/O(o)	—	—	RIIC3 SCL	I(s)/O(o)	DV1_ VSYNC	I(s)	IERxD	I(s)	IRQ6	I(s)	VIO_D12	I(s)	DV0_ DATA12	I(s)	—	—	—	—	(9)
18	A18	P1_3	I(s)/O(o)	—	—	RIIC1 SDA	I(s)/O(o)	DV0_ DATA19	I(s)	ET_ COL	I(s)	IRQ3	I(s)	ADTRG	I(s)	—	—	—	—	—	—	(9)
19	A19	P1_0	I(s)/O(o)	—	—	RIIC0 SCL	I(s)/O(o)	DV0_ DATA16	I(s)	TCLKA	I(s)	IRQ0	I(s)	VIO_VD	I(s)	DV0_ VSYNC	I(s)	—	—	—	—	(9)
20	A20	P2_13	I(s)/O	—	—	D29	I(s)/O	SSL00	I(s)/O	DV0_ DATA13	I(s)	SPBIO 11_0	I(s)/O	CAN3TX	O	SCK0	I(s)/O	LCD1_ DATA13	O	IRQ7	I(s)	(8)
21	A21	P2_12	I(s)/O	—	—	D28	I(s)/O	RSPCK0	I(s)/O	DV0_ DATA12	I(s)	SPBIO 01_0	I(s)/O	CAN3RX	I(s)	IRQ6	I(s)	LCD1_ DATA12	O	TIOC1B	I(s)/O	(8)
22	A22	Vss																				
23	B1	Vcc																				
24	B2	Vss																				
25	B3	P6_0	I(s)/O	—	—	D0	I(s)/O	LCD1_ DATA8	O	RLIN30 RX	I(s)	DV0_ CLK	I(s)	TIOC1A	I(s)/O	IRQ5	I(s)	RxD3	I(s)	DV0_ DATA16	I(s)	(8)
26	B4	P11_2	I(s)/O	—	—	DV0_ DATA14	I(s)	TIOC4C	I(s)/O	—	—	RxD6	I(s)	LCD0_ DATA5	O	VIO_D14	I(s)	—	—	—	—	(7)
27	B5	P9_7	I(s)/O	—	—	LCD1_ DATA23	O	SPBIO 30_0	I(s)/O	SSI DATA2	I(s)/O	TIOC1A	I(s)/O	—	—	—	—	—	—	—	—	(7)
28	B6	P9_4	I(s)/O	—	—	LCD1_ DATA20	O	SPBIO 00_0	I(s)/O	—	—	RxD1	I(s)	—	—	—	—	—	—	—	—	(7)
29	B7	P5_10	I(s)/O	—	—	WE3/ DQMUL/ AH	O	—	—	DV0_ HSYNC	I(s)	—	—	CAN1TX	O	IERxD	O	LCD1_ DATA17	O	—	—	(7)
30	B8	P5_8	I(s)/O	—	—	LCD0_ EXTCLK	I(s)	IRQ0	I(s)	DV1_ CLK	I(s)	—	—	DV0_ CLK	I(s)	CS2	O	—	—	—	—	(7)
31	B9	P5_7	I(s)/O	—	—	TXOUT 0M	O	LCD1_ DATA7	O	LCD0_ DATA23	O	DV1_ DATA7	I(s)	RxD6	I(s)	TIOC0D	I(s)/O	SPDIF_ OUT	O	DV0_ DATA15	I(s)	(12), (13)
32	B10	P5_3	I(s)/O	—	—	TXOUT 2M	O	LCD1_ DATA3	O	LCD0_ DATA19	O	DV1_ DATA3	I(s)	TxD3	O	TIOC3C	I(s)/O	—	—	MISO3	I(s)/O	(12), (13)
33	B11	P5_1	I(s)/O	—	—	TXCLK OUTM	O	LCD1_ DATA1	O	LCD0_ DATA17	O	DV1_ DATA1	I(s)	RxD4	I(s)	TIOC0B	I(s)/O	—	—	SSL30	I(s)/O	(12), (13)
34	B12	Vss																				
35	B13	VIN1B	I(a)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	
36	B14	VDAVcc																				
37	B15	VIN1A	I(a)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	
38	B16	P1_7	I(s)/O(o)	—	—	RIIC3 SDA	I(s)/O(o)	DV1_ HSYNC	I(s)	RLIN30 RX	I(s)	IRQ7	I(s)	VIO_D13	I(s)	DV0_ DATA13	I(s)	—	—	—	—	(9)

No.	Ball Number	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Function 8		Simplified Circuit Diagram Figure 1.4
		Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
39	B17	P1_4	I(s)/O(o)	—	—	RIIC2 SCL	I(s)/O(o)	DV0_ CLK	I(s)	CAN1RX	I(s)	IRQ4	I(s)	—	—	—	—	CAN_ CLK	I(s)	—	—	(9)
40	B18	P1_2	I(s)/O(o)	—	—	RIIC1 SCL	I(s)/O(o)	DV0_ DATA18	I(s)	FRB	I(s)	IRQ2	I(s)	—	—	—	—	LCD1_ EXTCLK	I(s)	—	—	(9)
41	B19	P2_15	I(s)/O	—	—	D31	I(s)/O	MISO0	I(s)/O	DV0_ DATA15	I(s)	SPBIO 31_0	I(s)/O	CAN_ CLK	I(s)	RxD0	I(s)	LCD1_ DATA15	O	IRQ1	I(s)	(8)
42	B20	PVcc																				
43	B21	Vss																				
44	B22	P2_10	I(s)/O	—	—	D26	I(s)/O	ET_ RXD2	I(s)	DV0_ DATA10	I(s)	SSIRxD0	I(s)	RLIN30 TX	O	LCD1_ DATA10	O	VIO_D10	I(s)	MOSI4	I(s)/O	(8)
45	C1	P6_5	I(s)/O	—	—	D5	I(s)/O	LCD1_ DATA13	O	CAN2TX	O	—	—	SCK5	I(s)/O	—	—	SSL10	I(s)/O	DV0_ DATA21	O	(8)
46	C2	Vcc																				
47	C3	Vss																				
48	C4	P6_2	I(s)/O	—	—	D2	I(s)/O	LCD1_ DATA10	O	RLIN31 RX	I(s)	IRQ7	I(s)	TCLKA	I(s)	TIOC2A	I(s)/O	RxD2	I(s)	DV0_ DATA18	I(s)	(8)
49	C5	P11_3	I(s)/O	—	—	DV0_ DATA15	I(s)	TIOC4D	I(s)/O	—	—	—	—	LCD0_ DATA4	O	VIO_D15	I(s)	—	—	—	—	(7)
50	C6	P11_1	I(s)/O	—	—	DV0_ DATA13	I(s)	TIOC4B	I(s)/O	—	—	TxD6	O	LCD0_ DATA6	O	VIO_D13	I(s)	—	—	—	—	(7)
51	C7	P9_5	I(s)/O	—	—	LCD1_ DATA21	O	SPBIO 10_0	I(s)/O	SSISCK2	I(s)/O	CTS1	I(s)/O	CS4	O	—	—	—	—	—	—	(7)
52	C8	P9_2	I(s)/O	—	—	LCD1_ DATA18	O	SPBCLK 0	O	RLIN30 TX	O	SCK1	I(s)/O	A0	O	—	—	—	—	—	—	(7)
53	C9	Vss																				
54	C10	P5_5	I(s)/O	—	—	TXOUT 1M	O	LCD1_ DATA5	O	LCD0_ DATA21	O	DV1_ DATA5	I(s)	AUDIO_ XOUT	O	TIOC0C	I(s)/O	FCE	O	DV0_ DATA13	I(s)	(12), (13)
55	C11	P5_4	I(s)/O	—	—	TXOUT 1P	O	LCD1_ DATA4	O	LCD0_ DATA20	O	DV1_ DATA4	I(s)	RxD3	I(s)	TIOC3D	I(s)/O	—	—	DV0_ DATA12	I(s)	(12), (13)
56	C12	LVDS APVcc																				
57	C13	VRM	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	
58	C14	REXT	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	
59	C15	Vss																				
60	C16	P1_5	I(s)/O(o)	—	—	RIIC2 SDA	I(s)/O(o)	DV1_ CLK	I(s)	CAN4RX	I(s)	IRQ5	I(s)	VIO_ CLK	I(s)	—	—	LCD1_ EXTCLK	I(s)	—	—	(9)
61	C17	P1_1	I(s)/O(o)	—	—	RIIC0 SDA	I(s)/O(o)	DV0_ DATA17	I(s)	TCLKC	I(s)	IRQ1	I(s)	VIO_HD	I(s)	DV0_ HSYNC	I(s)	—	—	—	—	(9)
62	C18	P2_14	I(s)/O	—	—	D30	I(s)/O	MOSI0	I(s)/O	DV0_ DATA14	I(s)	SPBIO 21_0	I(s)/O	CAN4RX	I(s)	TxD0	O	LCD1_ DATA14	O	IRQ0	I(s)	(8)
63	C19	PVcc																				
64	C20	Vss																				
65	C21	P2_9	I(s)/O	—	—	D25	I(s)/O	ET_ RXD1	I(s)	DV0_ DATA9	I(s)	SSIWS0	I(s)/O	RLIN30 RX	I(s)	LCD1_ DATA9	O	VIO_D9	I(s)	SSL40	I(s)/O	(8)
66	C22	P2_7	I(s)/O	—	—	D23	I(s)/O	ET_ TXD3	O	DV0_ DATA7	I(s)	SSITxD5	O	IETxD	O	RTS1	I(s)/O	VIO_D7	I(s)	LCD0_ DATA23	O	(8)
67	D1	P6_7	I(s)/O	—	—	D7	I(s)/O	LCD1_ DATA15	O	—	—	LCD0_ TCON6	O	RxD5	I(s)	—	—	MISO1	I(s)/O	DV0_ DATA23	O	(8)
68	D2	P6_6	I(s)/O	—	—	D6	I(s)/O	LCD1_ DATA14	O	—	—	LCD0_ TCON5	O	TxD5	O	—	—	MOSI1	I(s)/O	DV0_ DATA22	O	(8)
69	D3	Vcc																				
70	D4	Vss																				
71	D5	P6_3	I(s)/O	—	—	D3	I(s)/O	LCD1_ DATA11	O	RLIN31 TX	O	IRQ2	I(s)	CTS5	I(s)/O	TIOC2B	I(s)/O	TxD2	O	DV0_ DATA19	I(s)	(8)
72	D6	P6_1	I(s)/O	—	—	D1	I(s)/O	LCD1_ DATA9	O	RLIN30 TX	O	IRQ4	I(s)	TIOC1B	I(s)/O	SSI DATA4	I(s)/O	TxD3	O	DV0_ DATA17	I(s)	(8)
73	D7	PVcc																				
74	D8	PVcc																				
75	D9	LVDS PLLVcc																				
76	D10	Vss																				
77	D11	LVDS REFRIN	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	

No.	Ball Number	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Function 8		Simplified Circuit Diagram Figure 1.4
		Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
78	D12	LVDS APVcc																				
79	D13	Vcc																				
80	D14	VRP	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	
81	D15	Vss																				
82	D16	PVcc																				
83	D17	PVcc																				
84	D18	PVcc																				
85	D19	Vss																				
86	D20	P2_8	I(s)O	—	—	D24	I(s)O	ET_ RXD0	I(s)	DV0_ DATA8	I(s)	SSISCK0	I(s)O	LCD0_ TCON6	O	LCD1_ DATA8	O	VIO_D8	I(s)	RSPCK4	I(s)O	(8)
87	D21	P10_15	I(s)O	—	—	DV0_ DATA11	I(s)	SSITXD1	O	—	—	MISO0	I(s)O	LCD0_ DATA8	O	VIO_D11	I(s)	—	—	—	—	(7)
88	D22	P10_14	I(s)O	—	—	DV0_ DATA10	I(s)	SSIRxD1	I(s)	—	—	MOSI0	I(s)O	LCD0_ DATA9	O	VIO_D10	I(s)	—	—	—	—	(7)
89	E1	P6_10	I(s)O	—	—	D10	I(s)O	DV0_ DATA14	I(s)	—	—	LCD0_ TCON5	O	RxD0	I(s)	LCD0_ DATA2	O	—	—	IRQ2	I(s)	(8)
90	E2	P6_9	I(s)O	—	—	D9	I(s)O	DV0_ DATA13	I(s)	—	—	—	—	TxD0	O	LCD0_ DATA1	O	—	—	IRQ1	I(s)	(8)
91	E3	P6_8	I(s)O	—	—	D8	I(s)O	DV0_ DATA12	I(s)	—	—	CAN_ CLK	I(s)	SCK0	I(s)O	LCD0_ DATA0	O	—	—	IRQ0	I(s)	(8)
92	E4	Vcc																				
93	E19	P2_11	I(s)O	—	—	D27	I(s)O	ET_ RXD3	I(s)	DV0_ DATA11	I(s)	SSITxD0	O	TIOC1A	I(s)O	LCD1_ DATA11	O	VIO_D11	I(s)	MISO4	I(s)O	(8)
94	E20	P2_6	I(s)O	—	—	D22	I(s)O	ET_ TXD2	O	DV0_ DATA6	I(s)	SSIRxD5	I(s)	—	—	RxD1	I(s)	VIO_D6	I(s)	LCD0_ DATA22	O	(8)
95	E21	P10_12	I(s)O	—	—	DV0_ DATA8	I(s)	SSISCK1	I(s)O	—	—	RSPCK0	I(s)O	LCD0_ DATA11	O	VIO_D8	I(s)	—	—	—	—	(7)
96	E22	P2_5	I(s)O	—	—	D21	I(s)O	ET_ TXD1	O	DV0_ DATA5	I(s)	SSIWS5	I(s)O	SPBSSL_ 1	O	TxD1	O	VIO_D5	I(s)	LCD0_ DATA21	O	(8)
97	F1	P6_14	I(s)O	—	—	D14	I(s)O	DV0_ DATA22	I(s)	—	—	TxD6	O	—	—	LCD0_ DATA6	O	—	—	IRQ6	I(s)	(8)
98	F2	P6_13	I(s)O	—	—	D13	I(s)O	DV0_ DATA21	I(s)	—	—	SCK6	I(s)O	RxD1	I(s)	LCD0_ DATA5	O	—	—	IRQ5	I(s)	(8)
99	F3	P6_11	I(s)O	—	—	D11	I(s)O	DV0_ DATA15	I(s)	—	—	LCD0_ TCON6	O	SCK1	I(s)O	LCD0_ DATA3	O	—	—	IRQ3	I(s)	(8)
100	F4	Vcc																				
101	F19	P2_4	I(s)O	—	—	D20	I(s)O	ET_ TXD0	O	DV0_ DATA4	I(s)	SSISCK5	I(s)O	SPBCLK_ 1	O	SCK1	I(s)O	VIO_D4	I(s)	LCD0_ DATA20	O	(8)
102	F20	P10_13	I(s)O	—	—	DV0_ DATA9	I(s)	SSIWS1	I(s)O	—	—	SSL00	I(s)O	LCD0_ DATA10	O	VIO_D9	I(s)	—	—	—	—	(7)
103	F21	P2_2	I(s)O	—	—	D18	I(s)O	ET_ TXEN	O	DV0_ DATA2	I(s)	SPBIO_ 20_1	I(s)O	MLB_ SIG	I(s)O	TIOC2B	I(s)O	VIO_D2	I(s)	LCD0_ DATA18	O	(8)
104	F22	P4_15	I(s)O	—	—	LCD0_ DATA23	O	LCD1_ TCON2	O	SD_ D2_0	I(s)O	MMC_ D2	I(s)O	SPBIO_ 31_1	I(s)O	SSITxD3	O	RxD2	I(s)	IRQ7	I(s)	(7)
105	G1	P11_13	I(s)O	—	—	CAN1TX	O	SSL10	I(s)O	LCD0_ TCON4	O	MMC_ D5	I(s)O	LCD0_ TCON1	O	—	—	—	—	—	—	(7)
106	G2	P11_12	I(s)O	—	—	CAN1RX	I(s)	RSPCK1	I(s)O	IRQ3	I(s)	MMC_ D4	I(s)O	LCD0_ TCON2	O	—	—	—	—	—	—	(7)
107	G3	P6_15	I(s)O	—	—	D15	I(s)O	DV0_ DATA23	I(s)	—	—	RxD6	I(s)	—	—	LCD0_ DATA7	O	—	—	IRQ7	I(s)	(8)
108	G4	P6_12	I(s)O	—	—	D12	I(s)O	DV0_ DATA20	I(s)	—	—	—	—	TxD1	O	LCD0_ DATA4	O	—	—	IRQ4	I(s)	(8)
109	G19	P4_14	I(s)O	—	—	LCD0_ DATA22	O	LCD1_ TCON1	O	SD_ D3_0	I(s)O	MMC_ D3	I(s)O	SPBIO_ 21_1	I(s)O	SSIRxD3	I(s)	TxD2	O	IRQ6	I(s)	(7)
110	G20	P2_3	I(s)O	—	—	D19	I(s)O	ET_ CRS	I(s)	DV0_ DATA3	I(s)	SPBIO_ 30_1	I(s)O	IERxD	I(s)	CTS1	I(s)O	VIO_D3	I(s)	LCD0_ DATA19	O	(8)
111	G21	P4_13	I(s)O	—	—	LCD0_ DATA21	O	LCD1_ TCON0	O	SD_ CMD_0	I(s)O	MMC_ CMD	I(s)O	SPBIO_ 11_1	I(s)O	SSIWS3	I(s)O	RxD1	I(s)	IRQ5	I(s)	(7)
112	G22	Vss																				
113	H1	P7_2	I(s)O	—	—	RAS	O	DV0_ DATA18	I(s)	ET_ TXER	O	RxD4	I(s)	CAN2RX	I(s)	SSIWS1	I(s)O	TIOC0C	I(s)O	—	—	(7)
114	H2	P7_1	I(s)O	—	—	CS3	O	DV0_ DATA17	I(s)	ET_ TXCLK	I(s)	TxD4	O	DV0_ CLK	I(s)	SSISCK1	I(s)O	TIOC0B	I(s)O	—	—	(7)
115	H3	P11_14	I(s)O	—	—	SPDIF_ IN	I(s)	MOSI1	I(s)O	LCD0_ TCON5	O	MMC_ D6	I(s)O	LCD0_ TCON0	O	—	—	—	—	—	—	(7)

No.	Ball Number	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Function 8		Simplified Circuit Diagram Figure 1.4
		Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
116	H4	P7_0	I(s)O	MD_BOOT2	I(s)	CS0	O	DV0_DATA16	I(s)	ET_MDC	O	SCK4	I(s)O	RLIN30_TX	O	—	—	TIOC0A	I(s)O	—	—	(7)
117	H19	P4_11	I(s)O	—	—	LCD0_DATA19	O	LCD1_TCON6	O	SD_D0_0	I(s)O	MMC_D0	I(s)O	SSIxD5	O	CAN4TX	O	SCK1	I(s)O	IRQ3	I(s)	(7)
118	H20	P10_11	I(s)O	—	—	DV0_DATA7	I(s)	TIOC2B	I(s)O	—	—	ET_RXD3	I(s)	LCD0_DATA12	O	VIO_D7	I(s)	—	—	—	—	(7)
119	H21	P10_10	I(s)O	—	—	DV0_DATA6	I(s)	TIOC2A	I(s)O	—	—	ET_RXD2	I(s)	LCD0_DATA13	O	VIO_D6	I(s)	—	—	—	—	(7)
120	H22	P4_12	I(s)O	—	—	LCD0_DATA20	O	LCD1_CLK	O	SD_CLK_0	O	MMC_CLK	O	SPBIO_01_1	I(s)O	SSISCK3	I(s)O	TxD1	O	IRQ4	I(s)	(7)
121	J1	P7_5	I(s)O	—	—	RDWR	O	DV0_DATA21	I(s)	ET_TXD1	O	RxD7	I(s)	—	—	SSISCK2	I(s)O	TIOC1B	I(s)O	—	—	(7)
122	J2	P7_4	I(s)O	—	—	CKE	O	DV0_DATA20	I(s)	ET_TXD0	O	TxD7	O	—	—	SSITxD1	O	TIOC1A	I(s)O	—	—	(7)
123	J3	P7_3	I(s)O	—	—	CAS	O	DV0_DATA19	I(s)	ET_TXEN	O	SCK7	I(s)O	CAN2TX	O	SSIRxD1	I(s)	TIOC0D	I(s)O	—	—	(7)
124	J4	P11_15	I(s)O	—	—	SPDIF_OUT	O	MISO1	I(s)O	IRQ1	I(s)	MMC_D7	I(s)O	LCD0_CLK	O	—	—	—	—	—	—	(7)
125	J9	Vss																				
126	J10	Vss																				
127	J11	Vss																				
128	J12	Vss																				
129	J13	Vss																				
130	J14	Vss																				
131	J19	Vcc																				
132	J20	P10_9	I(s)O	—	—	DV0_DATA5	I(s)	TIOC1B	I(s)O	—	—	ET_RXD1	I(s)	LCD0_DATA14	O	VIO_D5	I(s)	—	—	—	—	(7)
133	J21	P10_8	I(s)O	—	—	DV0_DATA4	I(s)	TIOC1A	I(s)O	—	—	ET_RXD0	I(s)	LCD0_DATA15	O	VIO_D4	I(s)	—	—	—	—	(7)
134	J22	P4_10	I(s)O	—	—	LCD0_DATA18	O	LCD1_TCON5	O	SD_D1_0	I(s)O	MMC_D1	I(s)O	SSIRxD5	I(s)	—	—	RxD0	I(s)	IRQ2	I(s)	(7)
135	K1	P7_9	I(s)O	—	—	A1	O	SSIWS3	I(s)O	ET_RXD0	I(s)	CAN0TX	O	—	—	—	—	TIOC3B	I(s)O	IRQ0	I(s)	(7)
136	K2	P7_7	I(s)O	—	—	WE1/DQMLU	O	DV0_DATA23	I(s)	ET_TXD3	O	RTS7	I(s)O	—	—	SSI_DATA2	I(s)O	TIOC2B	I(s)O	—	—	(7)
137	K3	P7_6	I(s)O	—	—	WE0/DQMLL	O	DV0_DATA22	I(s)	ET_TXD2	O	CTS7	I(s)O	—	—	SSIWS2	I(s)O	TIOC2A	I(s)O	—	—	(7)
138	K4	P7_8	I(s)O	—	—	RD	O	SSISCK3	I(s)O	—	—	CAN0RX	I(s)	—	—	—	—	TIOC3A	I(s)O	IRQ1	I(s)	(7)
139	K9	Vss																				
140	K10	Vss																				
141	K11	Vss																				
142	K12	Vss																				
143	K13	Vss																				
144	K14	Vss																				
145	K19	Vcc																				
146	K20	P4_8	I(s)O	—	—	LCD0_DATA16	O	LCD1_TCON3	O	SD_CD_0	I(s)	MMC_CD	I(s)	SSISCK5	I(s)O	CAN2TX	O	SCK0	I(s)O	IRQ0	I(s)	(7)
147	K21	P4_9	I(s)O	—	—	LCD0_DATA17	O	LCD1_TCON4	O	SD_WP_0	I(s)	—	—	SSIWS5	I(s)O	CAN2RX	I(s)	TxD0	O	IRQ1	I(s)	(7)
148	K22	P2_1	I(s)O	—	—	D17	I(s)O	ET_TXER	O	DV0_DATA1	I(s)	SPBIO_10_1	I(s)O	MLB_DAT	I(s)O	TIOC2A	I(s)O	VIO_D1	I(s)	LCD0_DATA17	O	(8)
149	L1	P11_5	I(s)O	—	—	DV0_DATA17	I(s)	SD_WP_0	I(s)	SSIWS4	I(s)O	—	—	LCD0_DATA2	O	—	—	—	—	—	—	(7)
150	L2	P7_11	I(s)O	—	—	A3	O	SSITxD3	O	ET_RXD2	I(s)	CAN1RX	I(s)	—	—	—	—	TIOC3D	I(s)O	IRQ3	I(s)	(7)
151	L3	P7_10	I(s)O	—	—	A2	O	SSIRxD3	I(s)	ET_RXD1	I(s)	CAN1TX	O	—	—	—	—	TIOC3C	I(s)O	IRQ2	I(s)	(7)
152	L4	P11_4	I(s)O	—	—	DV0_DATA16	I(s)	SD_CD_0	I(s)	SSISCK4	I(s)O	MMC_CD	I(s)	LCD0_DATA3	O	—	—	—	—	—	—	(7)
153	L9	Vss																				
154	L10	Vss																				
155	L11	Vss																				



No.	Ball Number	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Function 8		Simplified Circuit Diagram Figure 1.4
		Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
156	L12	Vss																				
157	L13	Vss																				
158	L14	Vss																				
159	L19	PVcc																				
160	L20	P4_7	I(s)O	—	—	LCD0_ DATA15	O	MISO1	I(s)O	TIOC4D	I(s)O	PWM2H	O	SSITxD0	O	—	—	DV0_ DATA15	I(s)	—	—	(7)
161	L21	P2_0	I(s)O	—	—	D16	I(s)O	ET_ TXCLK	I(s)	DV0_ DATA0	I(s)	SPBIO 00_1	I(s)O	MLB_ CLK	I(s)	IRQ5	I(s)	VIO_D0	I(s)	LCD0_ DATA16	O	(8)
162	L22	P4_6	I(s)O	—	—	LCD0_ DATA14	O	MOSI1	I(s)O	TIOC4C	I(s)O	PWM2G	O	SSIRxD0	I(s)	—	—	DV0_ DATA14	I(s)	—	—	(7)
163	M1	P7_12	I(s)O	—	—	A4	O	SSISCK4	I(s)O	ET_ RXD3	I(s)	—	—	—	—	—	—	TIOC4A	I(s)O	IRQ4	I(s)	(7)
164	M2	P11_6	I(s)O	—	—	DV0_ DATA18	I(s)	SD_ D1_0	I(s)O	SSI DATA4	I(s)O	MMC_ D1	I(s)O	LCD0_ DATA1	O	—	—	—	—	—	—	(7)
165	M3	P11_7	I(s)O	—	—	DV0_ DATA19	I(s)	SD_ D0_0	I(s)O	CTS5	I(s)O	MMC_ D0	I(s)O	LCD0_ DATA0	O	—	—	—	—	—	—	(7)
166	M4	Vcc																				
167	M9	Vss																				
168	M10	Vss																				
169	M11	Vss																				
170	M12	Vss																				
171	M13	Vss																				
172	M14	Vss																				
173	M19	PVcc																				
174	M20	P4_5	I(s)O	—	—	LCD0_ DATA13	O	SSL10	I(s)O	TIOC4B	I(s)O	PWM2F	O	SSIWS 0	I(s)O	—	—	DV0_ DATA13	I(s)	—	—	(7)
175	M21	P4_4	I(s)O	—	—	LCD0_ DATA12	O	RSPCK1	I(s)O	TIOC4A	I(s)O	PWM2E	O	SSISCK0	I(s)O	—	—	DV0_ DATA12	I(s)	—	—	(7)
176	M22	P10_7	I(s)O	—	—	DV0_ DATA3	I(s)	TIOC0D	I(s)O	PWM2H	O	ET_ TXD3	O	LCD0_ DATA16	O	VIO_D3	I(s)	—	—	—	—	(7)
177	N1	P7_13	I(s)O	—	—	A5	O	SSIWS4	I(s)O	ET_ MDIO	I(s)O	—	—	—	—	—	—	TIOC4B	I(s)O	IRQ5	I(s)	(7)
178	N2	P7_14	I(s)O	—	—	A6	O	SSI DATA4	I(s)O	ET_ CRS	I(s)	—	—	—	—	—	—	TIOC4C	I(s)O	IRQ6	I(s)	(7)
179	N3	P7_15	I(s)O	—	—	A7	O	RSPCK0	I(s)O	ET_ RXCLK	I(s)	CTS5	I(s)O	SCL_ TXD0	O	—	—	TIOC4D	I(s)O	—	—	(7)
180	N4	PVcc																				
181	N9	Vss																				
182	N10	Vss																				
183	N11	Vss																				
184	N12	Vss																				
185	N13	Vss																				
186	N14	Vss																				
187	N19	P10_4	I(s)O	—	—	DV0_ DATA0	I(s)	TIOC0A	I(s)O	PWM2E	O	ET_ TXD0	O	LCD0_ DATA19	O	VIO_D0	I(s)	—	—	—	—	(7)
188	N20	P10_5	I(s)O	—	—	DV0_ DATA1	I(s)	TIOC0B	I(s)O	PWM2F	O	ET_ TXD1	O	LCD0_ DATA18	O	VIO_D1	I(s)	—	—	—	—	(7)
189	N21	P10_6	I(s)O	—	—	DV0_ DATA2	I(s)	TIOC0C	I(s)O	PWM2G	O	ET_ TXD2	O	LCD0_ DATA17	O	VIO_D2	I(s)	—	—	—	—	(7)
190	N22	P4_3	I(s)O	—	—	LCD0_ DATA11	O	TIOC0D	I(s)O	FWE	O	CAN3TX	O	RxD2	I(s)	—	—	MISO4	I(s)O	MMC_ D7	I(s)O	(7)
191	P1	P8_0	I(s)O	—	—	A8	O	SSL00	I(s)O	ET_ RXER	I(s)	SCK5	I(s)O	SCL_ SCK0	I(s)O	—	—	—	—	—	—	(7)
192	P2	P8_1	I(s)O	—	—	A9	O	MOSI0	I(s)O	ET_ RXDV	I(s)	TxD5	O	SCL_ RXD0	I(s)	—	—	—	—	—	—	(7)
193	P3	P8_2	I(s)O	—	—	A10	O	MISO0	I(s)O	AVB_ GPTP_ EXTERN	I(s)	RxD5	I(s)	IRQ0	I(s)	—	—	—	—	—	—	(7)
194	P4	PVcc																				
195	P9	Vss																				
196	P10	Vss																				

No.	Ball Number	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Function 8		Simplified Circuit Diagram Figure 1.4
		Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
197	P11	Vss																				
198	P12	Vss																				
199	P13	Vss																				
200	P14	Vss																				
201	P19	Vss																				
202	P20	P4_0	I(s)/O	—	—	LCD0_ DATA8	O	TIOC0A	I(s)/O	FRE	O	—	—	—	—	—	—	RSPCK4	I(s)/O	MMC_ D4	I(s)/O	(7)
203	P21	P4_2	I(s)/O	—	—	LCD0_ DATA10	O	TIOC0C	I(s)/O	FALE	O	CAN3RX	I(s)	TxD2	O	—	—	MOSI4	I(s)/O	MMC_ D6	I(s)/O	(7)
204	P22	P4_1	I(s)/O	—	—	LCD0_ DATA9	O	TIOC0B	I(s)/O	FCLE	O	—	—	SCK2	I(s)/O	—	—	SSL40	I(s)/O	MMC_ D5	I(s)/O	(7)
205	R1	P8_3	I(s)/O	—	—	A11	O	DV1_ DATA0	I(s)	RSPCK2	I(s)/O	RTS5	I(s)/O	—	—	IRQ1	I(s)	SCK2	I(s)/O	—	—	(7)
206	R2	P8_4	I(s)/O	—	—	A12	O	DV1_ DATA1	I(s)	SSL20	I(s)/O	—	—	—	—	IERxD	I(s)	RxD2	I(s)	—	—	(7)
207	R3	P8_5	I(s)/O	—	—	A13	O	DV1_ DATA2	I(s)	MOSI2	I(s)/O	—	—	—	—	—	—	—	—	—	—	(7)
208	R4	Vcc																				
209	R19	Vcc																				
210	R20	P3_15	I(s)/O	—	—	LCD0_ DATA7	O	—	—	NAF7	I(s)/O	—	—	TRACE CTL	O	—	—	SD_ D2_1	I(s)/O	MMC_ D2	I(s)/O	(7)
211	R21	P3_14	I(s)/O	—	—	LCD0_ DATA6	O	—	—	NAF6	I(s)/O	—	—	TRACE CLK	O	—	—	SD_ D3_1	I(s)/O	MMC_ D3	I(s)/O	(7)
212	R22	P3_13	I(s)/O	—	—	LCD0_ DATA5	O	—	—	NAF5	I(s)/O	AUDIO_ XOUT	O	—	—	—	—	SD_ CMD_1	I(s)/O	MMC_ CMD	I(s)/O	(7)
213	T1	P11_8	I(s)/O	—	—	DV0_ DATA20	I(s)	SD_ CLK_0	O	RTS5	I(s)/O	MMC_ CLK	O	LCD0_ TCON6	O	—	—	—	—	—	—	(7)
214	T2	P11_9	I(s)/O	—	—	DV0_ DATA21	I(s)	SD_ CMD_0	I(s)/O	SCK5	I(s)/O	MMC_ CMD	I(s)/O	LCD0_ TCON5	O	—	—	—	—	—	—	(7)
215	T3	P11_10	I(s)/O	—	—	DV0_ DATA22	I(s)	SD_ D3_0	I(s)/O	TxD5	O	MMC_ D3	I(s)/O	LCD0_ TCON4	O	—	—	—	—	—	—	(7)
216	T4	Vcc																				
217	T19	Vcc																				
218	T20	P3_10	I(s)/O	—	—	LCD0_ DATA2	O	—	—	NAF2	I(s)/O	—	—	TRACE DATA2	O	TIOC4C	I(s)/O	SD_ D1_1	I(s)/O	MMC_ D1	I(s)/O	(7)
219	T21	P3_11	I(s)/O	—	—	LCD0_ DATA3	O	—	—	NAF3	I(s)/O	—	—	TRACE DATA3	O	TIOC4D	I(s)/O	SD_ D0_1	I(s)/O	MMC_ D0	I(s)/O	(7)
220	T22	P3_12	I(s)/O	—	—	LCD0_ DATA4	O	—	—	NAF4	I(s)/O	—	—	—	—	—	—	SD_ CLK_1	O	MMC_ CLK	O	(7)
221	U1	Vss																				
222	U2	P8_6	I(s)/O	—	—	A14	O	DV1_ DATA3	I(s)	MISO2	I(s)/O	—	—	—	—	IETxD	O	TxD2	O	—	—	(7)
223	U3	P11_11	I(s)/O	—	—	DV0_ DATA23	I(s)	SD_ D2_0	I(s)/O	RxD5	I(s)	MMC_ D2	I(s)/O	LCD0_ TCON3	O	—	—	—	—	—	—	(7)
224	U4	P8_7	I(s)/O	—	—	A15	O	DV1_ DATA4	I(s)	AUDIO_ XOUT	O	IRQ5	I(s)	ET_ COL	I(s)	—	—	—	—	—	—	(7)
225	U19	Vcc																				
226	U20	JP0_1	I	—	—	TDO	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(6)
227	U21	TCK	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(2)
228	U22	Vss																				
229	V1	CKIO	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(5)
230	V2	P8_8	I(s)/O	—	—	A16	O	DV1_ DATA5	I(s)	SPBIO_ 00_1	I(s)/O	SPDIF_ IN	I(s)	TIOC1A	I(s)/O	PWM1A	O	TxD3	O	SSISCK5	I(s)/O	(7)
231	V3	P8_9	I(s)/O	—	—	A17	O	DV1_ DATA6	I(s)	SPBIO_ 10_1	I(s)/O	SPDIF_ OUT	O	TIOC1B	I(s)/O	PWM1B	O	RxD3	I(s)	SSIWS5	I(s)/O	(7)
232	V4	P8_13	I(s)/O	—	—	A21	O	—	—	SPBSSL_ 1	O	TIOC3D	I(s)/O	TxD5	O	PWM1F	O	SGOUT_3	O	SSIWS4	I(s)/O	(7)
233	V19	P3_8	I(s)/O	—	—	LCD0_ DATA0	O	—	—	NAF0	I(s)/O	—	—	TRACE DATA0	O	TIOC4A	I(s)/O	SD_ CD_1	I(s)	MMC_ CD	I(s)	(7)
234	V20	TRST	I(s)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(3)
235	V21	JP0_0	I	—	—	TDI	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(2)
236	V22	TMS	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(6)

No.	Ball Number	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Function 8		Simplified Circuit Diagram Figure 1.4
		Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
237	W1	Vss																				
238	W2	P8_10	I(s)/O	—	—	A18	O	DV1_DATA7	I(s)	SPBIO_20_1	I(s)/O	TIOC3A	I(s)/O	CAN4TX	O	PWM1C	O	SGOUT_0	O	SSITxD5	O	(7)
239	W3	P8_11	I(s)/O	—	—	A19	O	—	—	SPBIO_30_1	I(s)/O	TIOC3B	I(s)/O	RxD5	I(s)	PWM1D	O	SGOUT_1	O	DV0_CLK	I(s)	(7)
240	W4	PVcc																				
241	W5	PVcc																				
242	W6	PVcc																				
243	W7	Vss																				
244	W8	Vss																				
245	W9	Vcc																				
246	W10	Vcc																				
247	W11	Vss																				
248	W12	PVcc																				
249	W13	PVcc																				
250	W14	PLLVcc																				
251	W15	Vss																				
252	W16	Vss																				
253	W17	AVss																				
254	W18	AVcc																				
255	W19	PVcc																				
256	W20	P3_9	I(s)/O	—	—	LCD0_DATA1	O	—	—	NAF1	I(s)/O	—	—	TRACE DATA1	O	TIOC4B	I(s)/O	SD_WP_1	I(s)	IRQ6	I(s)	(7)
257	W21	AUDIO_X2	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(10)
258	W22	AUDIO_X1	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(10)
259	Y1	P8_12	I(s)/O	—	—	A20	O	—	—	SPBCLK_1	O	TIOC3C	I(s)/O	SCK5	I(s)/O	PWM1E	O	SGOUT_2	O	SSISCK4	I(s)/O	(7)
260	Y2	P8_14	I(s)/O	—	—	A22	O	SPBIO_01_0	I(s)/O	SPBIO_00_1	I(s)/O	TIOC2A	I(s)/O	RSPCK2	I(s)/O	PWM1G	O	TxD4	O	SSI_DATA4	I(s)/O	(7)
261	Y3	PVcc																				
262	Y4	P3_7	I(s)/O	—	—	LCD0_TCON6	O	—	—	SSITxD1	O	LCD1_EXTCLK	I(s)	SCL_CTS0/RTS0	I(s)/O	TIOC3D	I(s)/O	CS1	O	WDT_OVF	O	(7)
263	Y5	P3_4	I(s)/O	—	—	LCD0_TCON3	O	ET_RXCLK	I(s)	SSISCK1	I(s)/O	AUDIO_XOUT2	O	SCL_SCK0	I(s)/O	TIOC3A	I(s)/O	SCK3	I(s)/O	—	—	(7)
264	Y6	P10_2	I(s)/O	—	—	DV0_HSYNC	I(s)	TCLKC	I(s)	PWM2C	O	ET_TXEN	O	LCD0_DATA21	O	VIO_HD	I(s)	—	—	—	—	(7)
265	Y7	P3_2	I(s)/O	—	—	LCD0_TCON1	O	ET_TXEN	O	—	—	RxD2	I(s)	SCL_RXD1	I(s)	TEND0	O	PWM2C	O	MOSI3	I(s)/O	(7)
266	Y8	RES	I(s)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(1)
267	Y9	NMI	I(s)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(3)
268	Y10	Vss																				
269	Y11	VBUS1	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	#N/A
270	Y12	VBUS0	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	#N/A
271	Y13	USB AVcc																				
272	Y14	Vss																				
273	Y15	P0_0	I(s)	MD_BOOT0	I(s)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(3)
274	Y16	P0_1	I(s)	MD_BOOT1	I(s)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(3)
275	Y17	P1_10	I(s)	—	—	AN2	I(a)	—	—	IRQ4	I(s)	TCLKB	I(s)	—	—	—	—	—	—	—	—	(4)
276	Y18	P1_13	I(s)	—	—	AN5	I(a)	DV0_HSYNC	I(s)	—	—	WAIT	I(s)	—	—	—	—	—	—	—	—	(4)
277	Y19	P1_15	I(s)	—	—	AN7	I(a)	—	—	—	—	AVB_CAPTURE	I(s)	—	—	—	—	—	—	—	—	(4)
278	Y20	PVcc																				

No.	Ball Number	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Function 8		Simplified Circuit Diagram Figure 1.4
		Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
279	Y21	VIDEO_X2	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(10)
280	Y22	VIDEO_X1	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(10)
281	AA1	P8_15	I(s)O	—	—	A23	O	SPBIO_11_0	I(s)O	SPBIO_10_1	I(s)O	TIOC2B	I(s)O	SSL20	I(s)O	PWM1H	O	RxD4	I(s)	—	—	(7)
282	AA2	PVcc																				
283	AA3	P9_1	I(s)O	—	—	A25	O	SPBIO_31_0	I(s)O	CAN0RX	I(s)	IRQ0	I(s)	MISO2	I(s)O	—	—	—	—	—	—	(7)
284	AA4	P3_5	I(s)O	—	—	LCD0_TCON4	O	ET_RXER	I(s)	SSIWS1	I(s)O	AUDIO_XOUT3	O	SCL_TXD0	O	TIOC3B	I(s)O	TxD3	O	—	—	(7)
285	AA5	P10_1	I(s)O	—	—	DV0_VSYNC	I(s)	TCLKB	I(s)	PWM2B	O	ET_TXER	O	LCD0_DATA22	O	VIO_VD	I(s)	—	—	—	—	(7)
286	AA6	P3_3	I(s)O	—	—	LCD0_TCON2	O	ET_MDIO	I(s)O	IRQ4	I(s)	BS	O	SCL_CTS1/RTS1	I(s)O	DACK0	O	PWM2D	O	MISO3	I(s)O	(7)
287	AA7	P3_1	I(s)O	—	—	LCD0_TCON0	O	ET_TXER	O	IRQ6	I(s)	TxD2	O	SCL_TXD1	O	AUDIO_CLK	I(s)	PWM2B	O	SSL30	I(s)O	(7)
288	AA8	RTC_X2	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(11)
289	AA9	P0_5	I(s)	—	—	RTC_X4	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(3), (10)
290	AA10	Vss																				
291	AA11	DM1	I/O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	
292	AA12	DP0	I/O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	
293	AA13	REFRIN	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	
294	AA14	Vss																				
295	AA15	USB_X2	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(10)
296	AA16	XTAL	O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(10)
297	AA17	P1_8	I(s)	—	—	AN0	I(a)	—	—	IRQ2	I(s)	DREQ0	I(s)	VIO_D14	I(s)	DV0_DATA14	I(s)	—	—	—	—	(4)
298	AA18	P1_11	I(s)	—	—	AN3	I(a)	—	—	IRQ5	I(s)	TCLKD	I(s)	—	—	—	—	—	—	—	—	(4)
299	AA19	P1_14	I(s)	—	—	AN6	I(a)	—	—	—	—	ET_COL	I(s)	—	—	—	—	—	—	—	—	(4)
300	AA20	AVcc																				
301	AA21	PVcc																				
302	AA22	BSCANP	I(s)	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(1)
303	AB1	PVcc																				
304	AB2	P9_0	I(s)O	—	—	A24	O	SPBIO_21_0	I(s)O	CAN0TX	O	TCLKC	I(s)	MOSI2	I(s)O	—	—	—	—	—	—	(7)
305	AB3	P3_6	I(s)O	—	—	LCD0_TCON5	O	ET_RXDV	I(s)	SSIRxD1	I(s)	—	—	SCL_RXD0	I(s)	TIOC3C	I(s)O	RxD3	I(s)	—	—	(7)
306	AB4	P10_0	I(s)O	—	—	DV0_CLK	I(s)	TCLKA	I(s)	PWM2A	O	ET_TXCLK	I(s)	LCD0_DATA23	O	VIO_CLK	I(s)	—	—	—	—	(7)
307	AB5	P10_3	I(s)O	—	—	—	—	TCLKD	I(s)	PWM2D	O	ET_CRS	I(s)	LCD0_DATA20	O	VIO_FLD	I(s)	—	—	—	—	(7)
308	AB6	P3_0	I(s)O	—	—	LCD0_CLK	O	ET_TXCLK	I(s)	IRQ2	I(s)	SCK2	I(s)O	SCL_SCK1	I(s)O	TxD2	O	PWM2A	O	RSPCK3	I(s)O	(7)
309	AB7	Vss																				
310	AB8	RTC_X1	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(11)
311	AB9	P0_4	I(s)	—	—	RTC_X3	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(3), (10)
312	AB10	Vss																				
313	AB11	DP1	I/O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	#N/A
314	AB12	DM0	I/O	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	#N/A
315	AB13	Vss																				
316	AB14	USB APVcc																				
317	AB15	USB_X1	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(10)
318	AB16	EXTAL	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	(10)
319	AB17	Vss																				

No.	Ball Number	Port Function/ Dedicated Function		Mode Function		Function 1		Function 2		Function 3		Function 4		Function 5		Function 6		Function 7		Function 8		Simplified Circuit Diagram Figure 1.4
		Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	Symbol	I/O	
320	AB18	P1_9	I(s)	—	—	AN1	I(a)	—	—	IRQ3	I(s)	—	—	VIO_D15	I(s)	DV0_DATA15	I(s)	—	—	—	—	(4)
321	AB19	P1_12	I(s)	—	—	AN4	I(a)	DV0_VSYNC	I(s)	—	—	VIO_FLD	I(s)	—	—	—	—	—	—	—	—	(4)
322	AB20	AVss																				
323	AB21	AVref																				
324	AB22	Vss																				

[Legend]

(s): Schmitt

(a): Analog

(o): Open drain

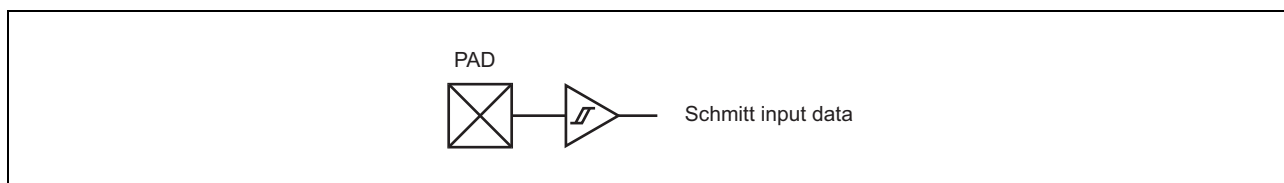


Figure 1.4 (1) Simplified Circuit Diagram (Schmitt Input Buffer)

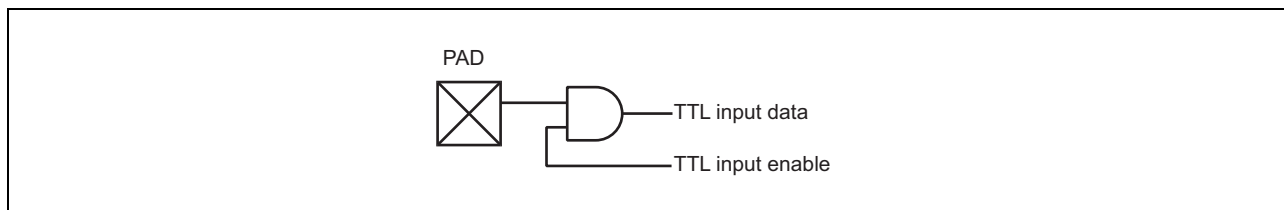


Figure 1.4 (2) Simplified Circuit Diagram (TTL AND Input Buffer)

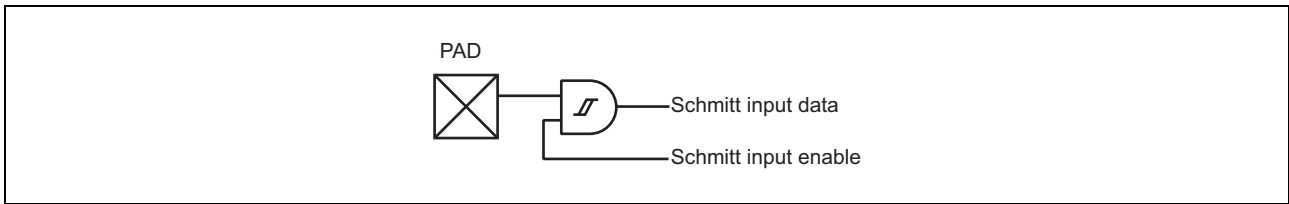


Figure 1.4 (3) Simplified Circuit Diagram (Schmitt AND Input Buffer)

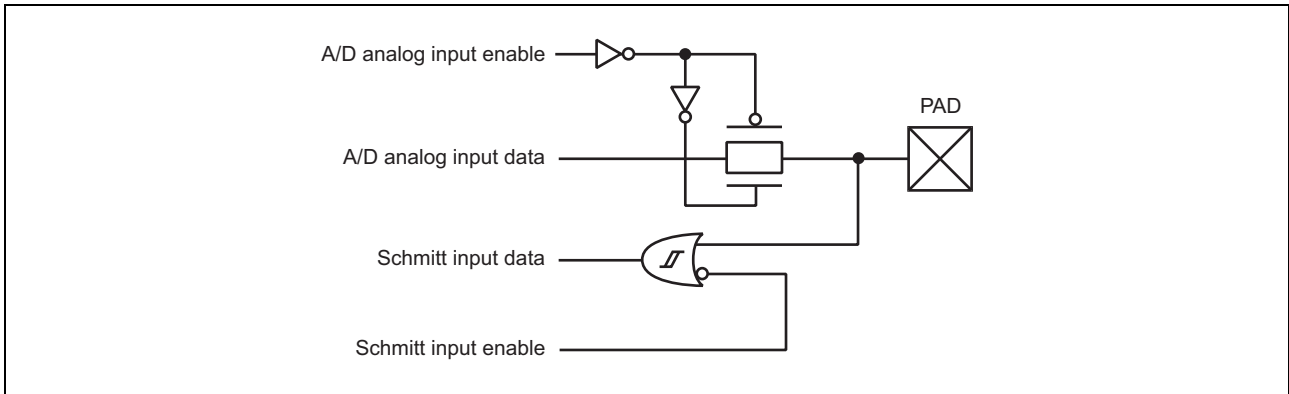


Figure 1.4 (4) Simplified Circuit Diagram (Schmitt OR Input and A/D Input Buffer)

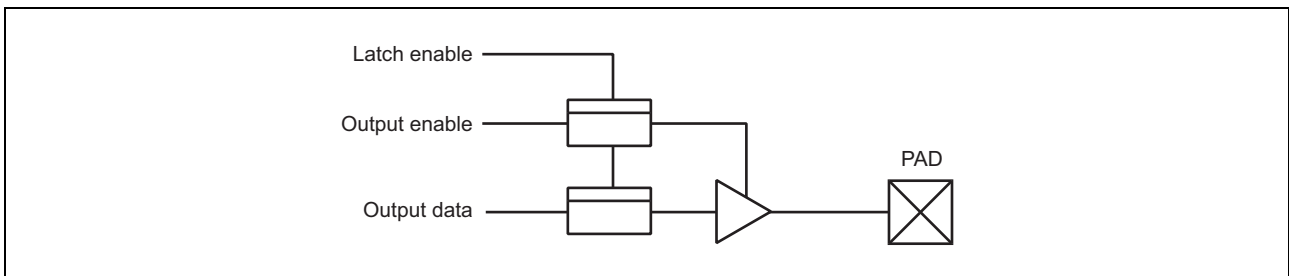


Figure 1.4 (5) Simplified Circuit Diagram (Output Buffer with Enable, with Latch)

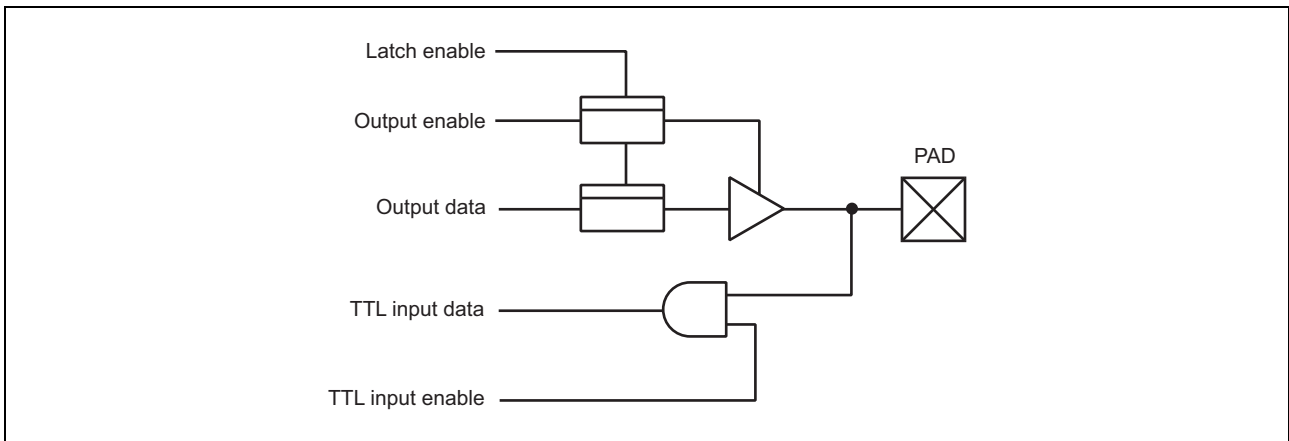


Figure 1.4 (6) Simplified Circuit Diagram (Bidirectional Buffer, TTL AND Input, with Latch)

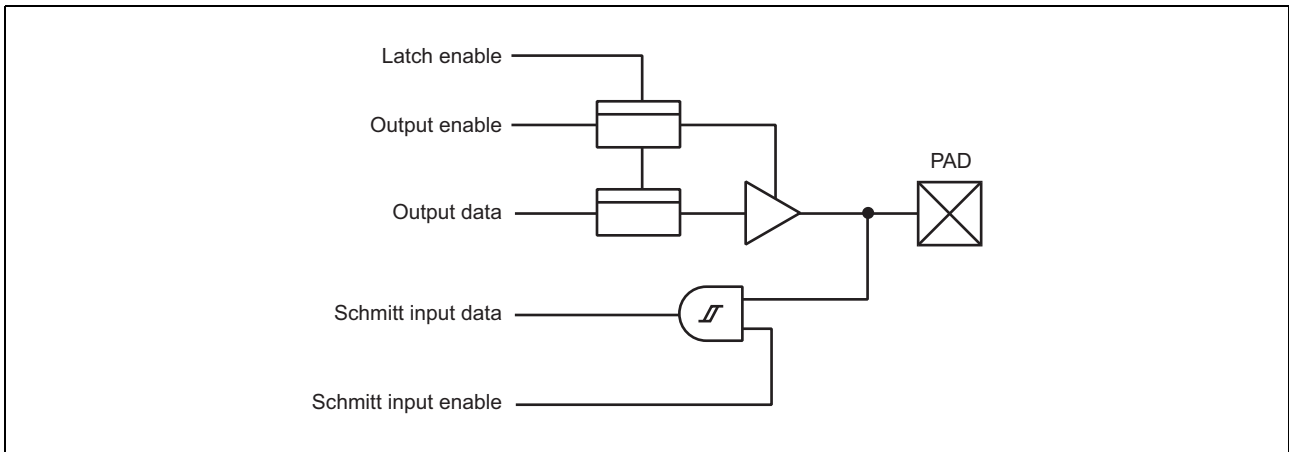


Figure 1.4 (7) Simplified Circuit Diagram (Bidirectional Buffer, Schmitt AND Input, with Latch)

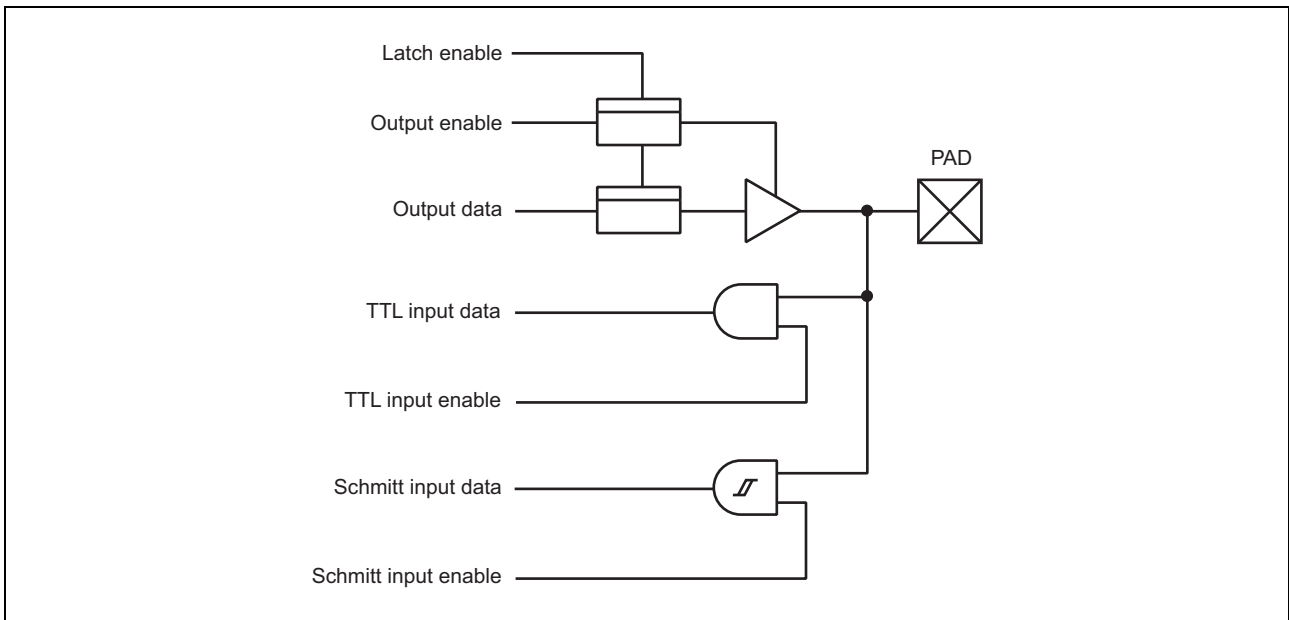


Figure 1.4 (8) Simplified Circuit Diagram (Bidirectional Buffer, TTL AND Input, Schmitt AND Input, with Latch)

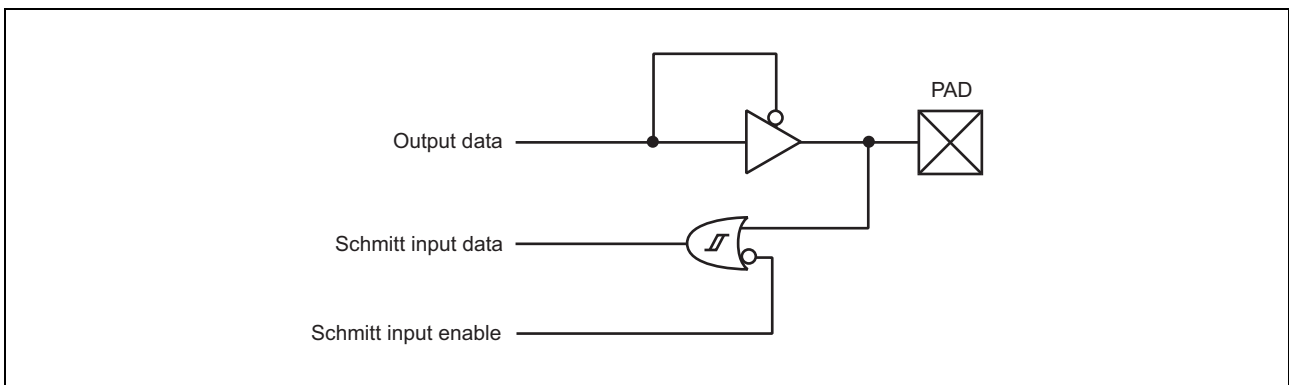


Figure 1.4 (9) Simplified Circuit Diagram (Open Drain Output and Schmitt OR Input Buffer)

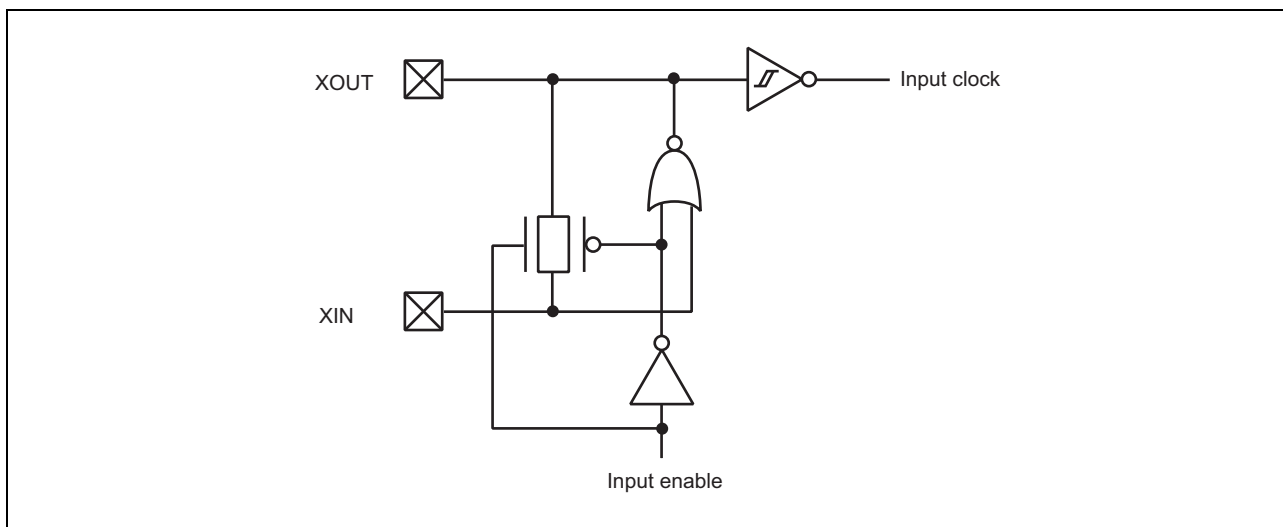


Figure 1.4 (10) Simplified Circuit Diagram (Oscillation Buffer 1)

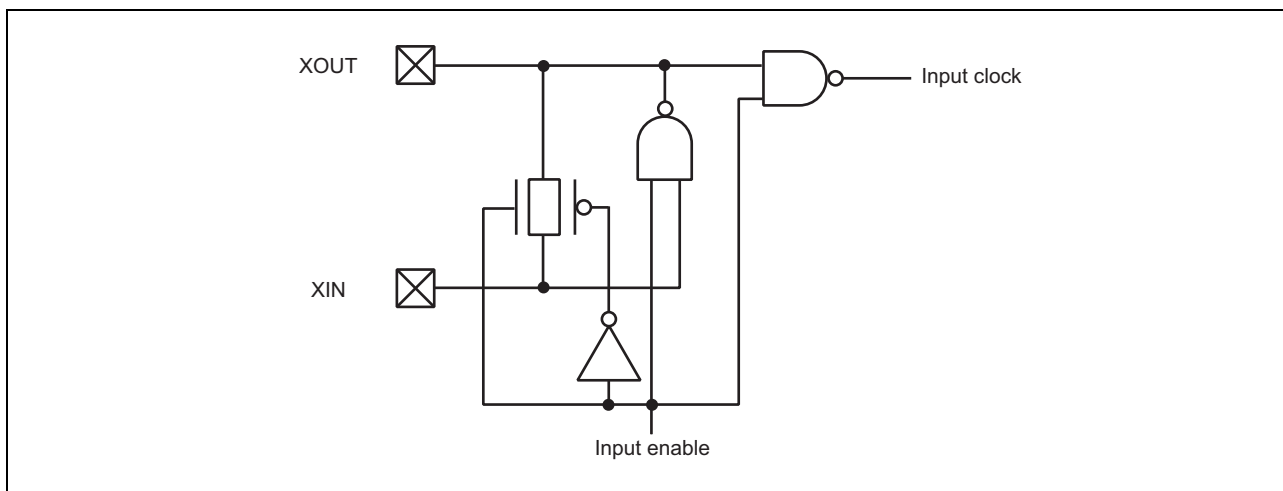


Figure 1.4 (11) Simplified Circuit Diagram (Oscillation Buffer 2)

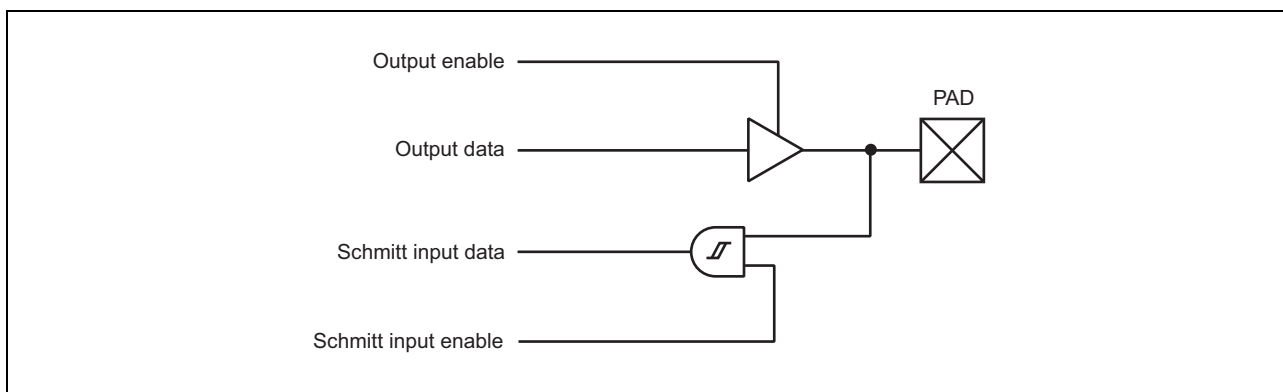


Figure 1.4 (12) Simplified Circuit Diagram (Bidirectional Buffer, Schmitt AND Input Buffer, No Latch)



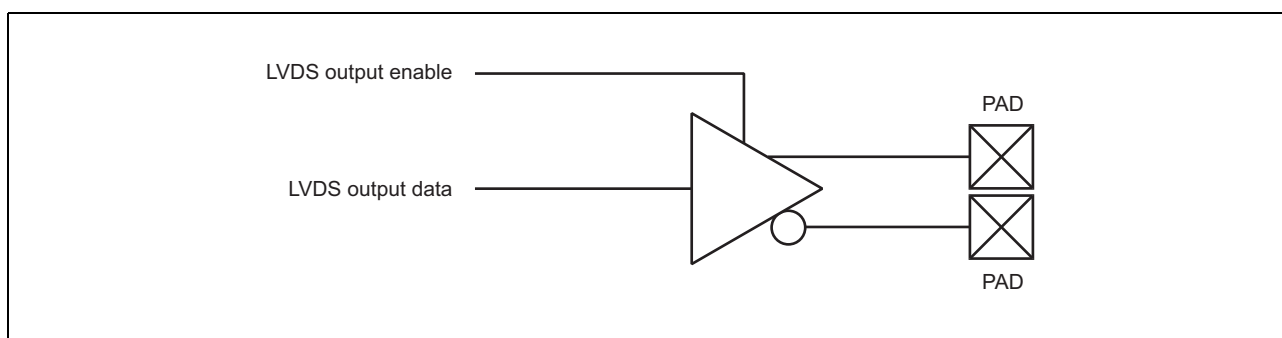


Figure 1.4 (13) Simplified Circuit Diagram (LVDS Output Buffer)

## 2. CPU

This product incorporates the ARM single-core Cortex-A9 MPCore, where the IP version is r3p0.

### 2.1 Features

- Instruction cache size: 32 Kbytes
- Data cache size: 32 Kbytes
- TLB entries: 128 entries
- Jazelle architecture extension: Full
- Media processing engine with NEON technology: Included
- FPU: Included
- PTM interface: Included
- Wrappers to support for power off and dormant mode: Not included
- Preload engine: Not included
- Number of interrupts: 0 (On-chip interrupt controller is not used.)
- Accelerator Coherence Port: Not included

For details, refer to Cortex-A9 MPCore Technical Reference Manual issued by ARM Ltd.

## 2.2 Configuration Signals

Table 2.1 shows the Cortex-A9 configuration signals and the settings.

**Table 2.1 Cortex-A9 Configuration Signal Settings**

Configuration Signal	Boot Mode
CFGEND	1'b0
CFGNMIF	1'b1
CLUSTERID	1'b0
FILTEREN*1	1'b1
FILTERSTART[31:20]*1	12'hE00
FILTEREND[31:20]*1	12'hFFF
PERIPBASE[31:13]*2	19'b111_1000_0000_0000_0000
TEINIT	1'b0
VINITHI	NOR, SRAM boot: 1'b0; on-chip ROM boot: 1'b1

Note 1. Do not change the initial settings of these signals by software.

Note 2. The base address for the private memory area in the Cortex-A9 processor is H'F0000000. For details and overview of the registers located in the addresses relative to this base address, refer to Cortex-A9 MPCore Technical Reference Manual issued by ARM Ltd.

## 3. Boot Mode

This LSI can be booted from the memory connected to the CS0 space, the serial flash memory, the NAND flash memory with an SD controller, and the NAND flash memory with an MMC controller.

### 3.1 Features

- Five boot modes
  - Boot mode 0: Boots the LSI from the memory (bus width: 16 bits) connected to the CS0 space
  - Boot mode 1: Boots the LSI from the memory (bus width: 32 bits) connected to the CS0 space
  - Boot mode 3: Boots the LSI from the serial flash memory connected to the SPI multi I/O bus space
  - Boot mode 4: Boots the LSI from the NAND flash memory with the SD controller\*<sup>1</sup>
  - Boot mode 5: Boots the LSI from the NAND flash memory with the MMC controller\*<sup>2</sup>

Note 1. It is possible to boot the LSI from the embedded SD (eSD) defined by the SD specification part 1 eSD addendum version 2.10 standard.

Note 2. It is possible to boot the LSI from the eMMC device corresponding to the boot operating mode of the JEDEC standard JESD84 A44 (MMCA 4.4) Standard. (It is not possible to boot the LSI from the MMC card.)

### 3.2 Boot Mode and Pin Function Setting

This LSI can determine the boot mode using external pins when  $\overline{\text{RES}}$  is low. The external pin settings for selecting the boot mode are shown in Table 3.1.

**Table 3.1 External Pin (MD\_BOOT2 to MD\_BOOT0) Settings and Corresponding Boot Modes**

MD_BOOT2	MD_BOOT1	MD_BOOT0	Boot Mode
*	0	0	Boot Mode 0 (CS0-space 16-bit booting) Boots the LSI from the memory (bus width: 16 bits) connected to the CS0 space.
*	1	0	Boot Mode 1 (CS0-space 32-bit booting) Boots the LSI from the memory (bus width: 32 bits) connected to the CS0 space.
1	0	1	Boot Mode 3 (serial flash booting) Boots the LSI from the serial flash memory connected to the SPI multi I/O bus space. The only way of booting this LSI chip is from channel 0 (P9_2 to P9_5) in this mode.
0	1	1	Boot Mode 4 (eSD booting) Boots the LSI from the NAND flash memory with the SD controller. The only way of booting this LSI chip is from channel 0 (P4_10 to P4_15) in this mode.
1	1	1	Boot Mode 5 (eMMC booting) Boots the LSI from the NAND flash memory with the MMC controller. The only way of booting this LSI chip is from channel 0 (P3_10 to P3_15) in this mode.

### 3.3 Hardware Used in Each Boot Mode

Table 3.2 gives information about the hardware used in each boot mode.

**Table 3.2 Hardware Used in Each Boot Mode**

Boot Mode	Peripheral Module	Pins Used	Remarks
Boot Mode 0 (CS0-space 16-bit booting)	Bus state controller	A[20:1] D[15:0] CS0 RD CKIO	—
Boot Mode 1 (CS0-space 32-bit booting)	Bus state controller	A[20:2] D[31:0] CS0 RD CKIO	—
Boot Mode 3 (Serial flash booting)	SPI multi I/O bus controller	SPBCLK_0 SPBSSL_0 SPBMO0_0 SPBMIO_0	The internal baud rate generator generates SPBCLK_0 by dividing B $\phi$ by 8.
Boot Mode 4 (eSD booting)	SD host interface	SD_CLK0 SD_CMD0 SD_D[3:0]0	The SD clock frequency (SD_CLK0) is generated by dividing P1 $\phi$ by 4.
Boot Mode 5 (eMMC booting)	MMC host interface	MMC_CLK MMC_CMD MMC_D[3:0]	The clock frequency (MMC_CLK) is generated by dividing P1 $\phi$ by 4.

### 3.4 Exception Vector Address at a Reset in Each Boot Mode

In this LSI, the exception vector address at a reset differs depending on the boot mode.

In this LSI, the exception vector at a reset starts from H'0000\_0000 (low vector) in boot mode 0 or 1 and from H'FFFF\_0000 (high vector) in boot mode 3 to 5.

In this LSI, an on-chip ROM is allocated in area H'FFFF\_0000 to H'FFFF\_FFFF. The on-chip ROM has a boot program which executes processing corresponding to the boot mode set by the MD\_BOOT2 to MD\_BOOT0 external pins.

Table 3.3 lists the exception vector address at a reset for each boot mode.

**Table 3.3 Exception Vector Address at a Reset in Each Boot Mode**

<b>Boot Mode</b>	<b>Exception Vector Address at a Rest</b>	<b>Memory Allocated at the Exception Vector Address</b>
Boot Mode 0 (CS0-space 16-bit booting)	H'0000 0000 (low vector)	Memory connected to the CS0 space
Boot Mode 1 (CS0-space 32-bit booting)	H'0000 0000 (low vector)	Memory connected to the CS0 space
Boot Mode 3 (Serial flash booting)	H'FFFF 0000 (high vector)	On-chip ROM (boot program)
Boot Mode 4 (eSD booting)	H'FFFF 0000 (high vector)	On-chip ROM (boot program)
Boot Mode 5 (eMMC booting)	H'FFFF 0000 (high vector)	On-chip ROM (boot program)

## 3.5 Operation

### 3.5.1 Boot Modes 0 and 1

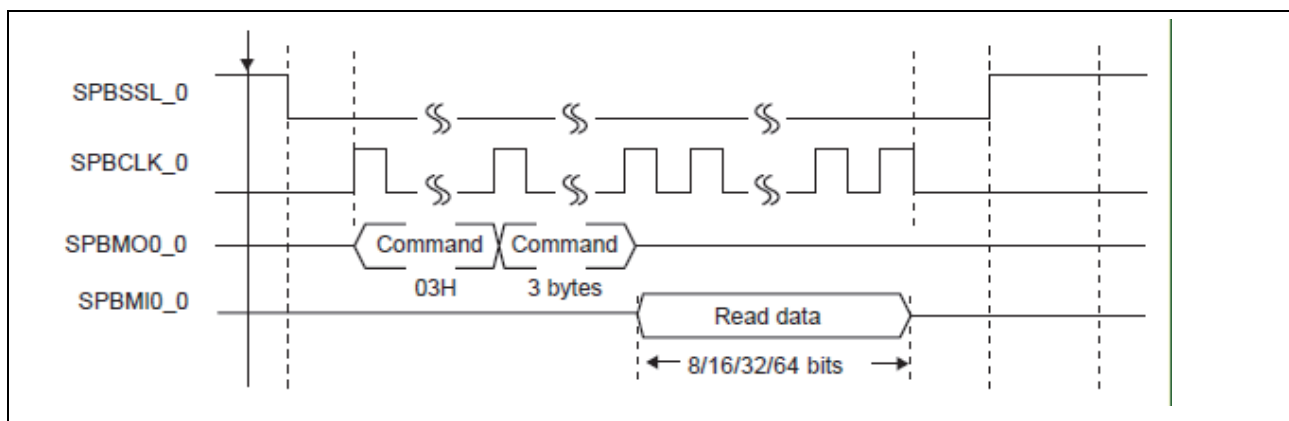
In boot modes 0 and 1, this LSI is booted from the memory connected to the CS0 space. In these modes, this LSI operates as follows:

After the power-on reset is canceled, program execution is started from H'0000\_0000 in the memory connected to the CS0 space.

### 3.5.2 Boot Mode 3

In boot mode 3, booting up is from the serial flash memory connected to the SPI multi I/O bus space. In this mode, this LSI operates as follows: After the power-on reset is canceled, the boot program stored in the on-chip ROM (starting from H'FFFF\_0000) is executed.

The boot program configures the SPI multi I/O bus controller in external address space read mode. With this configuration, this LSI converts reads from the SPI multi I/O bus space to SPI communications and is ready to read directly from the connected serial flash memory. The boot program configures a read command (opcode: 03H, address: 3 bytes, dummy cycle: none) as the command to the serial flash memory used for SPI communication conversion. Figure 3.1 shows the control signals output to the serial flash memory through SPI communication conversion.



**Figure 3.1 Control Signals Output to the Serial Flash Memory Through SPI Communication Conversion**

The boot program uses the area at H'2002\_0000 to H'2002\_3FFF as work memory. It branches to H'1800\_0000 (SPI multi I/O bus space) at the end of the processing. At this time, the I bit, F bit, T bit and bits Mode[4:0] in CPSR are set to the initial states with the I bit set to 1'b1 (IRQ masked state), the F bit set to 1'b1 (FIQ masked state), the T bit set to 1'b0 (ARM state) and bits Mode[4:0] set to 5'b10011 (supervisor mode).

### 3.5.3 Boot Mode 4

In boot mode 4, booting up is from the NAND flash memory with the SD controller, which is connected to channel 0 of the SD host interface. The flow of operation of this LSI in boot mode 4 is as described below.

After a power-on reset, this LSI executes the boot program stored in the on-chip ROM (starting from H'FFFF\_0000). The boot program transfers 28 Kbytes of program from the NAND flash memory with the SD controller connected to channel 0 of the SD host interface to the address range from H'2002\_4000 to H'2002\_AFFF of the large-capacity on-chip RAM.

The program (28 Kbytes) that the boot program transfers to the large-capacity on-chip RAM is called a loader program.

The boot program uses an area from H'2002\_0000 to H'2002\_3FFF as work memory. Note that the loader program must be stored in the NAND flash memory with the SD controller according to the loader program storage specifications.\*<sup>1</sup>

Note 1. For the storage specifications of the loader program, contact Renesas Electronics Corporation's sales office.

After the boot program finishes processing, it branches to H'2002\_4000 (large-capacity on-chip RAM). At this time, the I bit, F bit, T bit and bits Mode[4:0] in CPSR are set to the initial states with the I bit set to 1'b1 (IRQ masked state), the F bit set to 1'b1 (FIQ masked state), the T bit set to 1'b0 (ARM state) and bits Mode[4:0] set to 5'b10011 (supervisor mode).

The size of the loader program that the boot program transfers to the large-capacity on-chip RAM is fixed to 28 Kbytes. If your loader program exceeds this limit, use the loader program to transfer your program (application program) from the NAND flash memory with the SD controller to the large-capacity on-chip or external RAM using channel 0 of the SD host interface. Note that you must design a loader program.

#### (1) Initiation of the SD Host Interface Channel 0

#### (2) Transfer of the Loader Program

The 28-KB loader program is transferred from flash memory with the SD controller, which is connected to channel 0 of the SD host interface, to the address range from H'2002\_4000 to H'2002\_AFFF (page 0) of the large-capacity on-chip RAM. The address range from H'2002\_0000 to H'2002\_3FFF (page 0) of the large-capacity on-chip RAM is also used as the work memory for boot process.

Once transfer of the loader program has been completed, execution by the CPU jumps to page 0 of the large-capacity on-chip RAM so that it can start executing the transferred loader program.

#### (3) Transfer of an Application Program (as Desired)

The loader program employs the SD host interface to transfer the data to be deployed from flash memory with the SD controller to on-chip RAM or external RAM.



Figure 3.2 is a schematic view of the specification for boot mode 4.

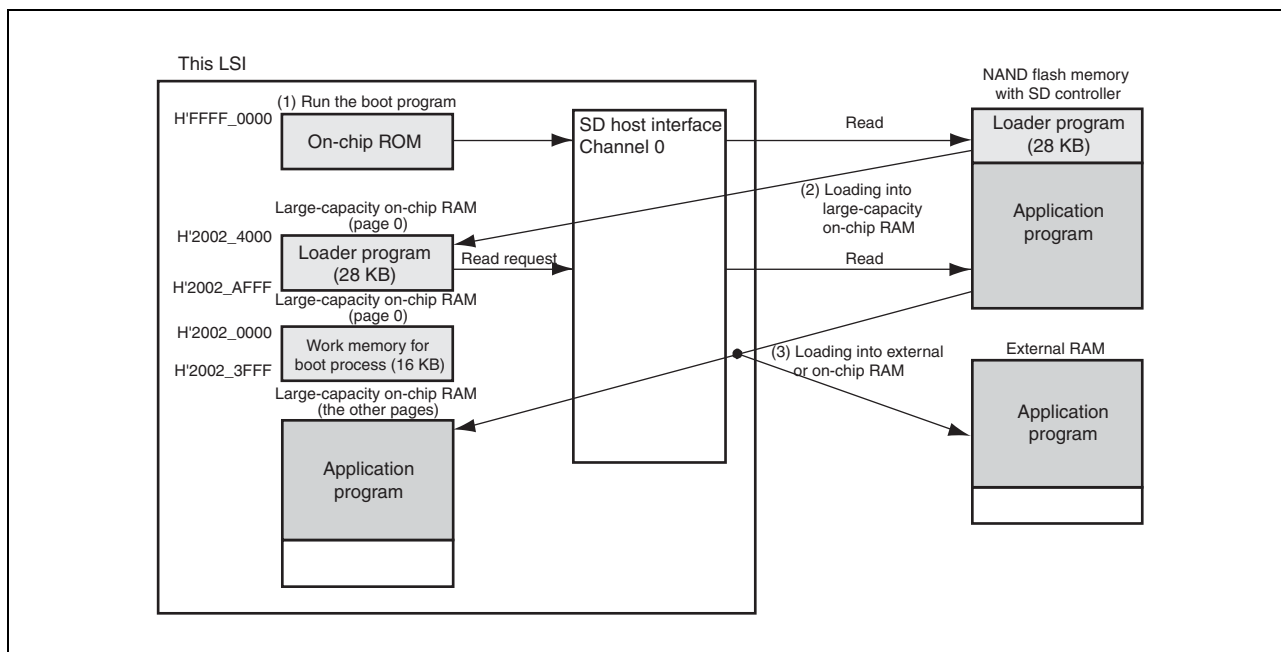


Figure 3.2 Schematic View of Specification for Boot Mode 4

### 3.5.4 Boot Mode 5

In boot mode 5, booting up is from the NAND flash memory with the MMC controller, which is connected to the MMC host interface. The flow of operation of this LSI in boot mode 5 is as described below.

After a power-on reset, this LSI executes the boot program stored in the on-chip ROM (starting from H'FFFF\_0000). The boot program transfers 28 Kbytes of program from the NAND flash memory with the MMC controller connected to the MMC host interface to the address range from H'2002\_4000 to H'2002\_AFFF of the large-capacity on-chip RAM. The program (28 Kbytes) that the boot program transfers to the large-capacity on-chip RAM is called a loader program.

The boot program uses an area from H'2002\_0000 to H'2002\_3FFF as work memory. Note that the loader program must be stored in the NAND flash memory with the MMC controller according to the loader program storage specifications.\*1  
Note 1. For the storage specifications of the loader program, contact Renesas Electronics Corporation's sales office.

After the boot program finishes processing, it branches to H'2002\_4000 (large-capacity on-chip RAM). At this time, the I bit, F bit, T bit and bits Mode[4:0] in CPSR are set to the initial states with the I bit set to 1'b1 (IRQ masked state), the F bit set to 1'b1 (FIQ masked state), the T bit set to 1'b0 (ARM state) and bits Mode[4:0] set to 5'b10011 (supervisor mode).

The size of the loader program that the boot program transfers to the large-capacity on-chip RAM is fixed to 28 Kbytes. If your loader program exceeds this limit, use the loader program to transfer your program (application program) from the NAND flash memory with the MMC controller to the large-capacity on-chip or external RAM using the MMC host interface. Note that you must design a loader program.

#### (1) Initiation of the MMC Host Interface

#### (2) Transfer of the Loader Program

The 28-KB loader program is transferred from flash memory with the MMC controller, which is connected to the MMC host interface, to the address range from H'2002\_4000 to H'2002\_AFFF (page 0) of the large-capacity on-chip RAM with the MMC data bus width of 4 bits. The address range from H'2002\_0000 to H'2002\_3FFF (page 0) of the large-capacity on-chip RAM is also used as the work memory for boot process.

Once transfer of the loader program has been completed, execution by the CPU jumps to page 0 of the large-capacity on-chip RAM so that it can start executing the transferred loader program.

#### (3) Transfer of an Application Program (as Desired)

The loader program employs the MMC host interface to transfer the data to be deployed from flash memory with the MMC controller to on-chip RAM or external RAM.

Figure 3.3 is a schematic view of the specification for boot mode 5.

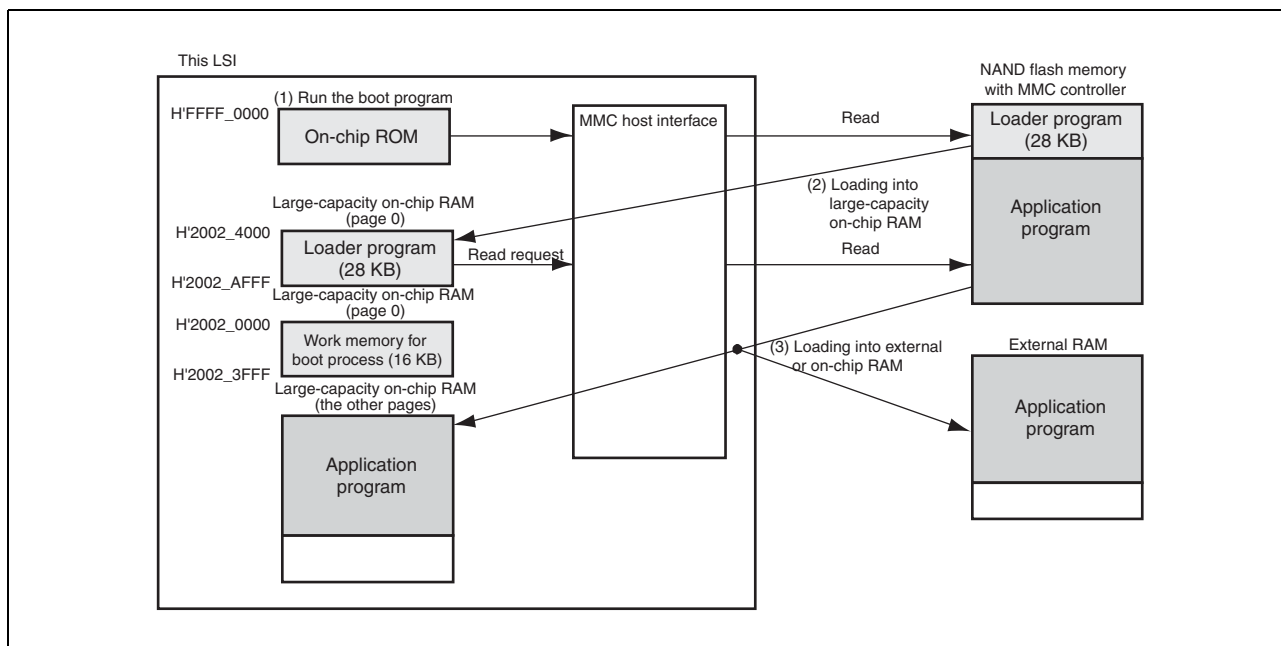


Figure 3.3 Schematic View of Specification for Boot Mode 5

## 3.6 Notes

### 3.6.1 Boot Related Pins

The initial states and output states in deep standby mode of the pins related to CS0 space memory read, SPI multi I/O bus space memory read, channel 0 of the SD host interface, and the MMC host interface are different in each boot mode. For details, refer to section 8, Bus State Controller, section 54, Ports, and section 55, Power-Down Modes.

### 3.6.2 Operation when an Exception Occurs with the Exception Vector Set to the High Vector Address

In this LSI, the program counter loops to its own address (exception vector address) in the on-chip ROM if an exception (except for a reset) occurs when the exception vector is set to the high vector address. In boot mode 3, 4 or 5, set the V bit in SCTLR to 0 to set the exception vector to the low vector address before an exception (except for a reset) occurs. For the details about the CP15 system control register (SCTLR), refer to the ARM Architecture Reference Manual.

## 4. Secondary Cache

This product incorporates ARM's PL310 as a secondary cache. The IP version is r3p2.

### 4.1 Features

- Total cache size: 128 Kbytes
- Number of cache ways: 8 ways
- Number of master ports : 2
- Number of slave ports: 2
- Lockdown by master: No
- Lockdown by line Defined
- Speculative read: No
- Sideband signal from CA9: No

For details, see CoreLink™ Level 2 Cache Controller L2C-310 Technical Reference Manual issued by ARM Ltd.

### 4.2 Configuration Signals

The setting values of the configuration signals are shown in Table 4.1.

**Table 4.1 Setting Values of Configuration Signals**

Configuration Signals	Setting Values
ASSOCIATIVITY*1	1'b1 (8 ways)
CACHEID[5:0]	6'b000000
CFGADDRFILTEN*1	1'b1
CFGADDRFILTEEND[11:0]*1	12'h3FFF
CFGADDRFILTSTART[11:0]*1	12'h180
CFGBIGEND	1'b0
DATAREADLAT[2:0]*1	3'b000
DATASETUPLAT[2:0]*1	3'b000
DATAWRITELAT[2:0]*1	3'b000
REGFILEBASE[19:0]*2	20'h3FFFF
TAGREADLAT[2:0]*1	3'b000
TAGSETUPLAT[2:0]*1	3'b000
TAGWRITELAT[2:0]*1	3'b000
WAYSIZESIZE[2:0]*1	3'b001 (16 Kbytes)

Note 1. Do not change the initial settings of these signals by software.

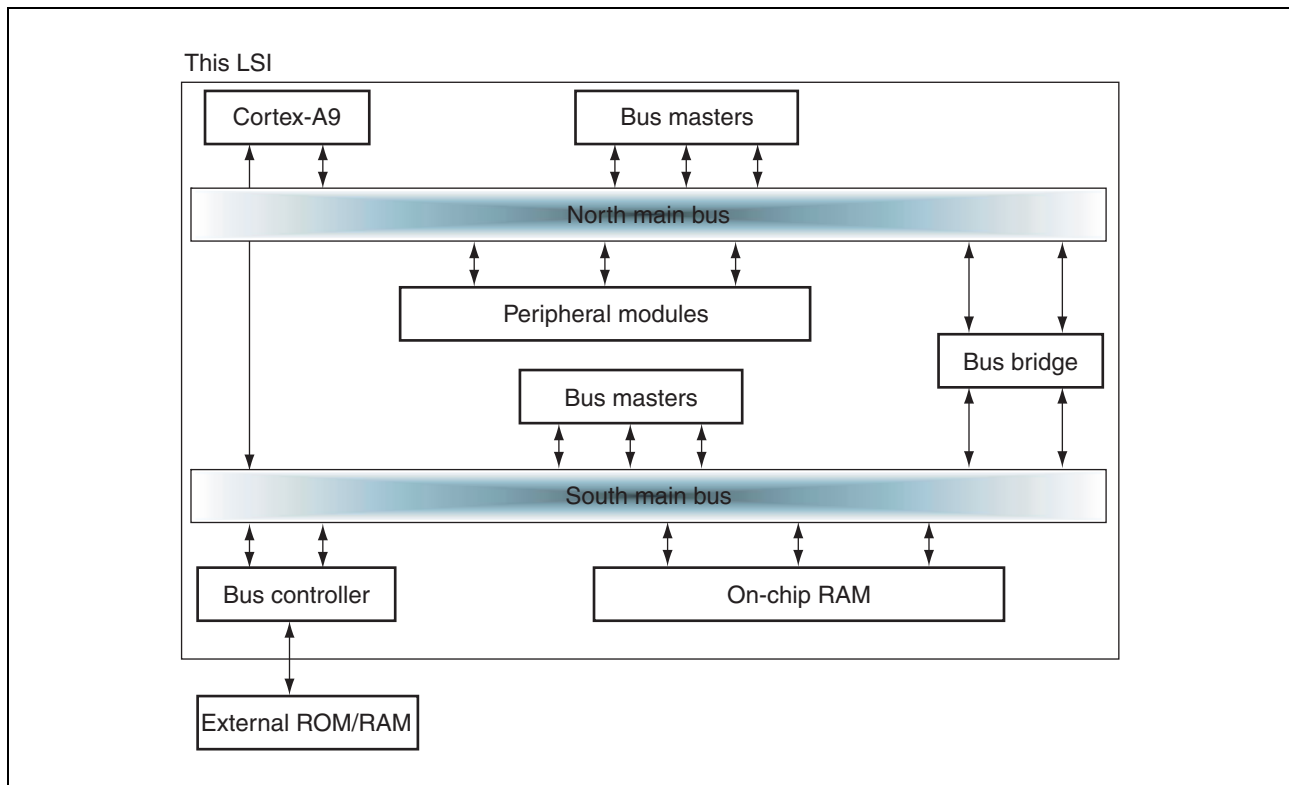
Note 2. The base address for the PL310 registers is H'3FFFF000. For the details and overview of the registers, see CoreLink™ Level 2 Cache Controller L2C-310 Technical Reference Manual issued by ARM Ltd. The external ROM/RAM mirror area (0x4000\_0000 to 0x5FFF\_FFFF) is mirrored before the secondary cache. Accordingly, when a cache maintenance operation is to be executed for the external ROM/RAM mirror area, treat this as the normal external ROM/RAM area in the physical addresses range from 0x0000\_0000 to 0x1FFF\_FFFF.

## 5. LSI Internal Bus

### 5.1 LSI Internal Bus

#### 5.1.1 Configuration

This LSI has two main buses: the north main bus where peripheral modules are connected and the south main bus where on-chip RAM and external ROM and RAM are connected. Figure 5.1 is a schematic diagram of the internal buses.



**Figure 5.1 Schematic Diagram of LSI Internal Bus**

#### 5.1.2 Operation

Cortex-A9 has separate interfaces for the north main bus and south main bus. The addresses assigned to the north main bus are accessed through the north main bus interface, and those assigned to the south main bus are accessed through the south main bus interface.

When a bus master connected to the north main bus, except for Cortex-A9, accesses the on-chip RAM or external ROM or RAM, access is executed through the bus bridge for access from the north main bus to the south main bus. The bus masters connected to the south main bus cannot access an address assigned to the north main bus. The internal bus of this LSI operates in little endian.

## 5.2 North Main Bus

### 5.2.1 Configuration

Various peripheral modules are connected to the north main bus. Figure 5.2 shows the configuration of the north main bus.

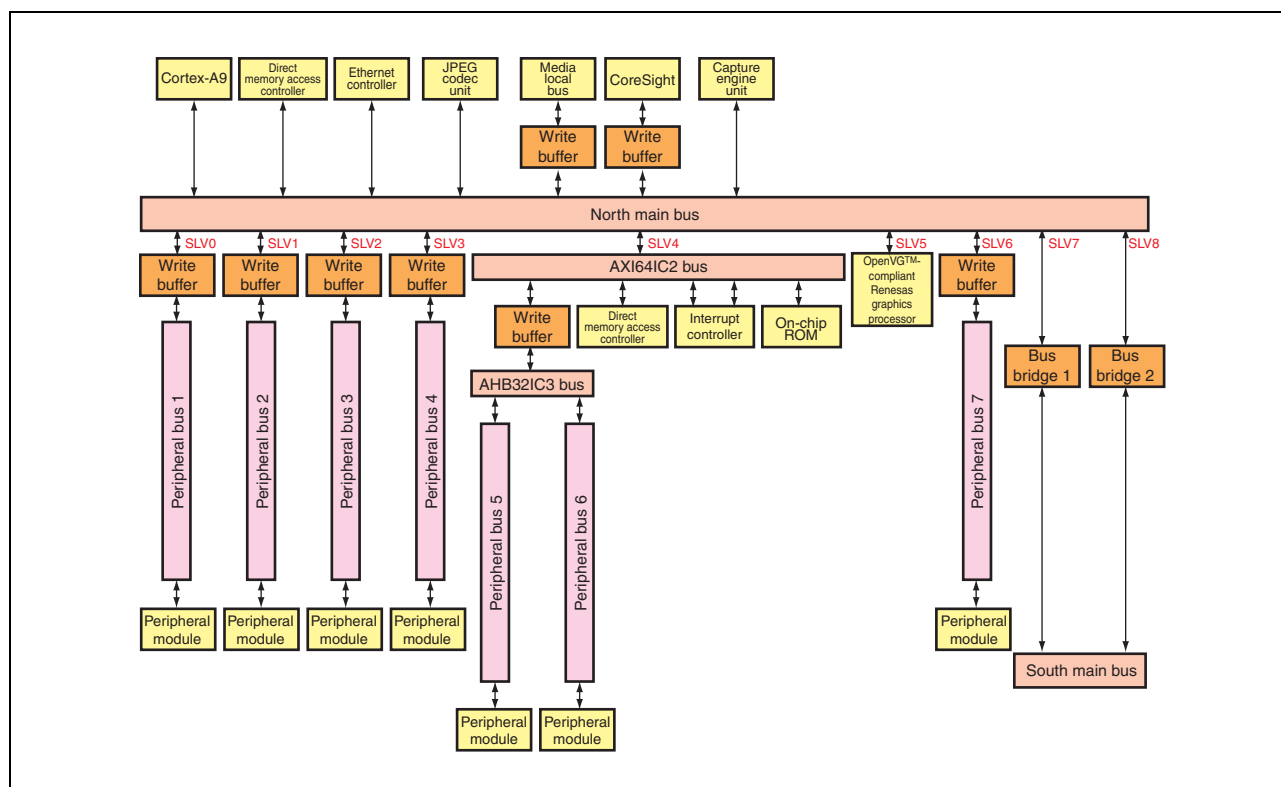


Figure 5.2 North Main Bus Configuration

### 5.2.2 Features

Table 5.1 shows the features of the north main bus.

Table 5.1 North Main Bus

Item	Description
Bus protocol	AMBA AXI protocol
Bus system configuration	AXI interconnect with multi-layer configuration for all channels
Bus clock frequency	B $\phi$
Bus width	64 bits
Arbitration	Round robin

### 5.2.3 Peripheral Buses

Table 5.2 is a list of the peripheral buses connected to the north main bus.

**Table 5.2 List of Peripheral Buses**

Item	Description
Peripheral bus 1	
Bus clock frequency	P0φ
Bus width	32 bits
Connected peripheral modules	Multi-function timer pulse unit 2
	Realtime clock
	NAND flash memory controller
	Digital video decoder channels 0 and 1
	Video display controller 5 channels 0 and 1
	Image renderer (IMR-LS2) channels 0 and 1
	Image renderer for display (IMR-LSD)
	Display out comparison unit channels 0 and 1
	Sound generator channels 0 to 3
	Motor control PWM timer
Peripheral bus 2	
Bus clock frequency	P0φ
Bus width	32 bits
Connected peripheral modules	Clock pulse generator
	Interrupt controller
	Direct memory access controller
	OS timer channels 0 and 1
	I <sup>2</sup> C bus interface channels 0 to 3
	IEBus controller
	LIN interface channels 0 and 1
	General I/O ports
Peripheral bus 3	
Bus clock frequency	P1φ
Bus width	32 bits
Connected peripheral modules	CAN interface
	Media local bus
	Dynamic range compression channels 0 and 1
	SD host interface channels 0 and 1
	MMC host interface
Peripheral bus 4	
Bus clock frequency	P1φ
Bus width	32 bits



**Table 5.2 List of Peripheral Buses**

Item	Description
Connected peripheral modules	Serial communication interface with FIFO channels 0 to 7
	Serial communication interface channels 0 and 1
	Renesas serial peripheral interface channels 0 to 4
	Renesas SPDIF interface
	CD-ROM decoder
	A/D converter
	USB2.0 host/function module channel 0
	JPEG codec unit
AXI64IC2 bus	
Bus protocol	AMBA AXI protocol
Bus system configuration	AXI interconnect with multi-layer configuration for all channels
Bus clock frequency	B $\phi$
Bus width	64 bits
Arbitration	Round robin
AHB32IC3 bus	
Bus protocol	AMBA AHB protocol
Bus clock frequency	B $\phi$
Bus width	32 bits
Peripheral bus 5	
Bus clock frequency	P1 $\phi$
Bus width	32 bits
Connected peripheral modules	Serial sound interface channels 0 to 5
	USB2.0 host/function module channel 1
	SCUX
Peripheral bus 6	
Bus clock frequency	B $\phi$
Bus width	32 bits
Connected peripheral modules	Ethernet controller
	Pixel format converter channels 0 and 1
	Capture engine unit
Peripheral bus 7	
Bus clock frequency	P1 $\phi$
Bus width	32 bits
Connected peripheral modules	CoreSight

## 5.3 South Main Bus

### 5.3.1 Configuration

On-chip RAM and external ROM and RAM are connected to the south main bus. Figure 5.3 shows the configuration of the south main bus.

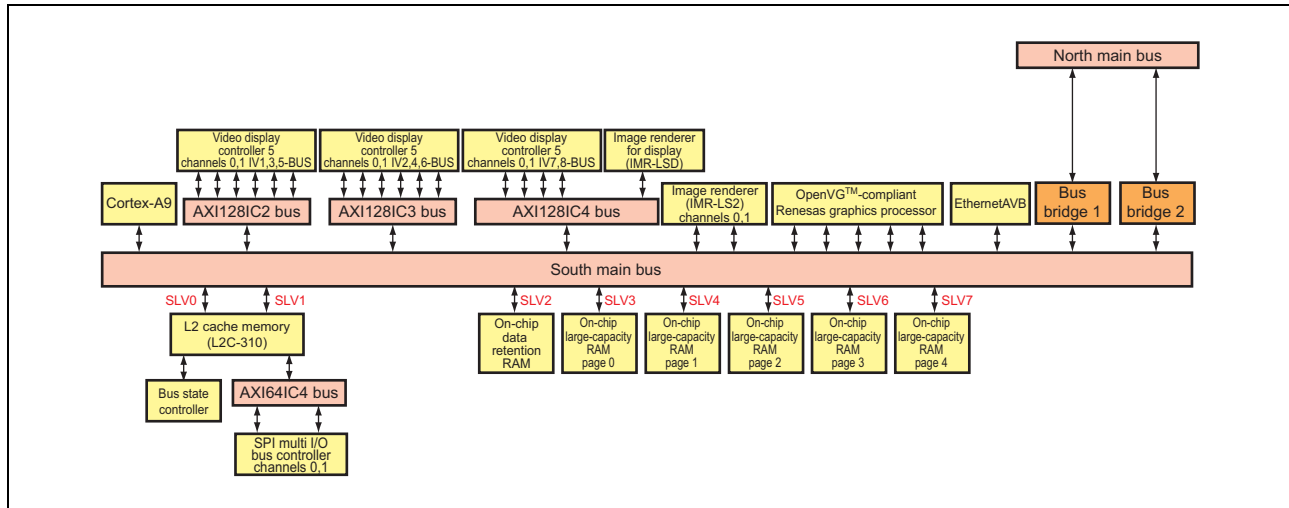


Figure 5.3 South Main Bus Configuration

### 5.3.2 Features

Table 5.3 shows the features of the south main bus.

Table 5.3 South Main Bus

Item	Description
Bus protocol	AMBA AXI protocol
Bus system configuration	AXI interconnect with multi-layer configuration for all channels
Bus clock frequency	B $\phi$
Bus width	128 bits
Arbitration	Round robin

### 5.3.3 Connected Buses

Table 5.4 is a list of the buses connected to the south main bus and their features.

**Table 5.4 List of Buses Connected to South Main Bus and their Features**

Item	Description
AXI128IC2, AXI128IC3, and AXI128IC4 buses	
Bus protocol	AMBA AXI protocol
Bus system configuration	AXI interconnect with multi-layer configuration for all channels
Bus clock frequency	B $\phi$
Bus width	128 bits
Arbitration	Round robin
AXI64IC4 bus	
Bus protocol	AMBA AXI protocol
Bus system configuration	AXI interconnect with multi-layer configuration for all channels
Bus clock frequency	B $\phi$
Bus width	64 bits
Arbitration	Round robin

## 5.4 Address Map

Table 5.5 shows the address map of this LSI.

**Table 5.5 Address Map**

Address	Area	Slave Area Viewed from North Main Bus Masters	Slave Area Viewed from South Main Bus Masters
0xFFFF_0000 to 0xFFFF_FFFF	I/O area	ROM in SLV4*1	—
0xFD00_0000 to 0xFFFE_FFFF	Reserved area	—	—
0xFCFF_0000 to 0xFCFF_FFFF	I/O area	SLV0 *1	—
0xFCFE_0000 to 0xFCFE_FFFF	I/O area	SLV1*1	—
0xFC08_0000 to 0xFCFD_FFFF	Reserved area	—	—
0xFC00_0000 to 0xFC07_FFFF	I/O area	SLV6*2, *4	—
0xF000_2000 to 0xFBFF_FFFF	Reserved area	—	—
0xF000_0000 to 0xF000_1FFF	Cortex-A9 private area	—	—
0xE823_0000 to 0xEFFF_FFFF	Reserved area	—	—
0xE820_0000 to 0xE822_FFFF	I/O area	SLV4 *1, *3	—
0xE814_0000 to 0xE81F_FFFF	Reserved area	—	—
0xE810_0000 to 0xE813_FFFF	I/O area	SLV5 *2, *5	—
0xE805_0000 to 0xE80F_FFFF	Reserved area	—	—
0xE803_0000 to 0xE804_FFFF	I/O area	SLV2 *1	—
0xE802_0000 to 0xE802_FFFF	Reserved area	—	—
0xE800_0000 to 0xE801_FFFF	I/O area	SLV3 *1	—
0xE000_0000 to 0xE7FF_FFFF	Reserved area	—	—
0x60A0_0000 to 0xDFFF_FFFF	Reserved area	—	—
0x6080_0000 to 0x609F_FFFF	On-chip large-capacity RAM page 4 mirror area (2 Mbytes)	SLV7	SLV7
0x6060_0000 to 0x607F_FFFF	On-chip large-capacity RAM page 3 mirror area (2 Mbytes)	—	SLV6
0x6040_0000 to 0x605F_FFFF	On-chip large-capacity RAM page 2 mirror area (2 Mbytes)	SLV8	SLV5
0x6020_0000 to 0x603F_FFFF	On-chip large-capacity RAM page 1 mirror area (2 Mbytes)	—	SLV4
0x6002_0000 to 0x601F_FFFF	On-chip large-capacity RAM page 0, (including on-chip data retention RAM) mirror area (2 Mbytes)	—	SLV3
0x6000_0000 to 0x6001_FFFF	—	—	SLV2

**Table 5.5 Address Map**

Address	Area	Slave Area Viewed from North Main Bus Masters	Slave Area Viewed from South Main Bus Masters
0x5C00_0000 to 0x5FFFF_FFFF	SPI multi I/O bus area channel 1 mirror area (64 Mbytes)	SLV7	SLV0
0x5800_0000 to 0x5BFF_FFFF	SPI multi I/O bus area channel 0 mirror area (64 Mbytes)		SLV1
0x5400_0000 to 0x57FF_FFFF	CS5 space mirror area (64 Mbytes)		SLV0
0x5000_0000 to 0x53FF_FFFF	CS4 space mirror area (64 Mbytes)		
0x4C00_0000 to 0x4FFF_FFFF	CS3 space mirror area (64 Mbytes)		
0x4800_0000 to 0x4BFF_FFFF	CS2 space mirror area (64 Mbytes)		
0x4400_0000 to 0x47FF_FFFF	CS1 space mirror area (64 Mbytes)		
0x4000_0000 to 0x43FF_FFFF	CS0 space mirror area (64 Mbytes)		
0x3FFF_C000 to 0x3FFF_FFFF	I/O area	SLV8	SLV0
0x3FEF_C000 to 0x3FFF_BFFF	Reserved area	—	—
0x3FEF_B000 to 0x3FEF_BFFF	I/O area	SLV8	SLV0
0x3FEF_A000 to 0x3FEF_AFFF	I/O area		SLV1
0x20A0_0000 to 0x3FEF_9FFF	Reserved area	—	—
0x2090_0000 to 0x209F_FFFF	On-chip large-capacity RAM page 4 lower area (1 Mbyte)	SLV7	SLV7
0x2080_0000 to 0x208F_FFFF	On-chip large-capacity RAM page 3 lower area (1 Mbyte)		SLV6
0x2070_0000 to 0x207F_FFFF	On-chip large-capacity RAM page 2 lower area (1 Mbyte)	SLV8	SLV5
0x2060_0000 to 0x206F_FFFF	On-chip large-capacity RAM page 1 lower area (1 Mbyte)		SLV4
0x2050_0000 to 0x205F_FFFF	On-chip large-capacity RAM page 0 lower area (1 Mbyte)		SLV3
0x2040_0000 to 0x204F_FFFF	On-chip large-capacity RAM page 4 upper area (1 Mbyte)	SLV7	SLV7
0x2030_0000 to 0x203F_FFFF	On-chip large-capacity RAM page 3 upper area (1 Mbyte)		SLV6
0x2020_0000 to 0x202F_FFFF	On-chip large-capacity RAM page 2 upper area (1 Mbyte)	SLV8	SLV5
0x2010_0000 to 0x201F_FFFF	On-chip large-capacity RAM page 1 upper area (1 Mbyte)		SLV4
0x2002_0000 to 0x200F_FFFF	On-chip large-capacity RAM page 0 upper area (including on-chip data retention RAM)		SLV3
0x2000_0000 to 0x2001_FFFF	(1 Mbyte)		SLV2

**Table 5.5 Address Map**

Address	Area	Slave Area Viewed from North Main Bus Masters	Slave Area Viewed from South Main Bus Masters
0x1C00_0000 to 0x1FFF_FFFF	SPI multi I/O bus area channel 1 (64 Mbytes)	SLV8	SLV0
0x1800_0000 to 0x1BFF_FFFF	SPI multi I/O bus area channel 0 (64 Mbytes)		SLV1
0x1400_0000 to 0x17FF_FFFF	CS5 space (64 Mbytes)		SLV0
0x1000_0000 to 0x13FF_FFFF	CS4 space (64 Mbytes)		
0x0C00_0000 to 0x0FFF_FFFF	CS3 space (64 Mbytes)		
0x0800_0000 to 0x0BFF_FFFF	CS2 space (64 Mbytes)		
0x0400_0000 to 0x07FF_FFFF	CS1 space (64 Mbytes)		
0x0000_0000 to 0x03FF_FFFF	CS0 space (64 Mbytes)		

Note 1. Only Cortex-A9, the direct memory access controller, and CoreSight can access this area. If any other north main bus master accesses this area, a decode error will occur.

Note 2. Only Cortex-A9 and CoreSight can access this area. If any other north main bus master accesses this area, a decode error will occur.

Note 3. If any address from 0xE821\_4800 to 0xE822\_FFFF is accessed, a decode error or slave error will occur.

Note 4. A slave error may occur depending on the CoreSight state.

Note 5. If this area is accessed while the OpenVG™-compliant Renesas graphics processor is in module standby state, a slave error will occur.

Note 6. If the on-chip large-capacity RAM is accessed while access is disabled, a slave error will occur.

Note 7. If the area indicated as "—" is accessed, a decode error or a slave error will occur.

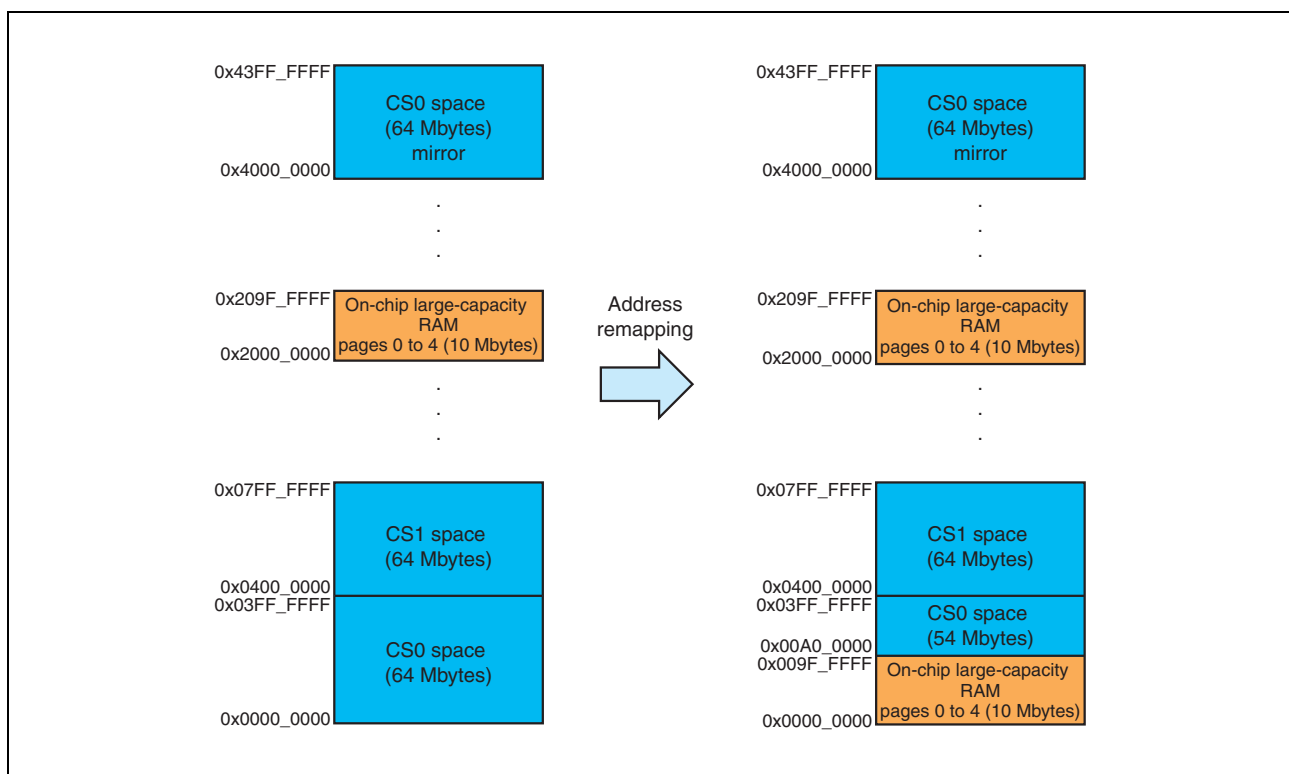
Note 8. I/O areas should be accessed in the size specified for each slave module.

## 5.5 Address Remapping

### 5.5.1 Overview

Execution in Cortex-A9 jumps to an exception vector placed in addresses 0x0000\_0000 to 0x0000\_001C when an exception such as a reset or an interrupt occurs. The interrupt response time depends on the time to access the memory connected to this area, and when low-speed memory is connected, the overhead is large. To avoid this, the exception vectors can be remapped to the on-chip high-speed RAM by using the MMU or vector base address register, or the address remapping function can be used to allocate the addresses where the exception vectors are placed to the on-chip high-speed RAM.

Figure 5.4 show the address maps before and after address remapping.



**Figure 5.4** Address Remapping

### 5.5.2 Operation

Addresses are remapped by setting the AXI128 bit in the remap register to 0. After address remapping, pages 0 to 4 of the on-chip large-capacity RAM are allocated to addresses 0x0000\_00000 to 0x009F\_FFFF. To access the CS0 space after address remapping, use the mirror area for the CS0 space.

During address remapping, access to addresses 0x0000\_0000 to 0x009F\_FFFF is prohibited. Accordingly, to modify the remap register, use the following steps.

- (1) Stop the bus masters except for Cortex-A9,  
or make settings so that addresses 0x0000\_00000 to 0x009F\_FFFF are never accessed.
- (2) Execute a program outside addresses 0x0000\_0000 to 0x009F\_FFFF.
- (3) After modifying the value of the remap register, execute a dummy read of the remap register.

## 5.6 AXI Interconnect

### 5.6.1 Configuration

The AXI interconnect in this LSI has a multi-layer configuration in all channels (five channels). Figure 5.5 shows a conceptual diagram of the AXI interconnect configuration.

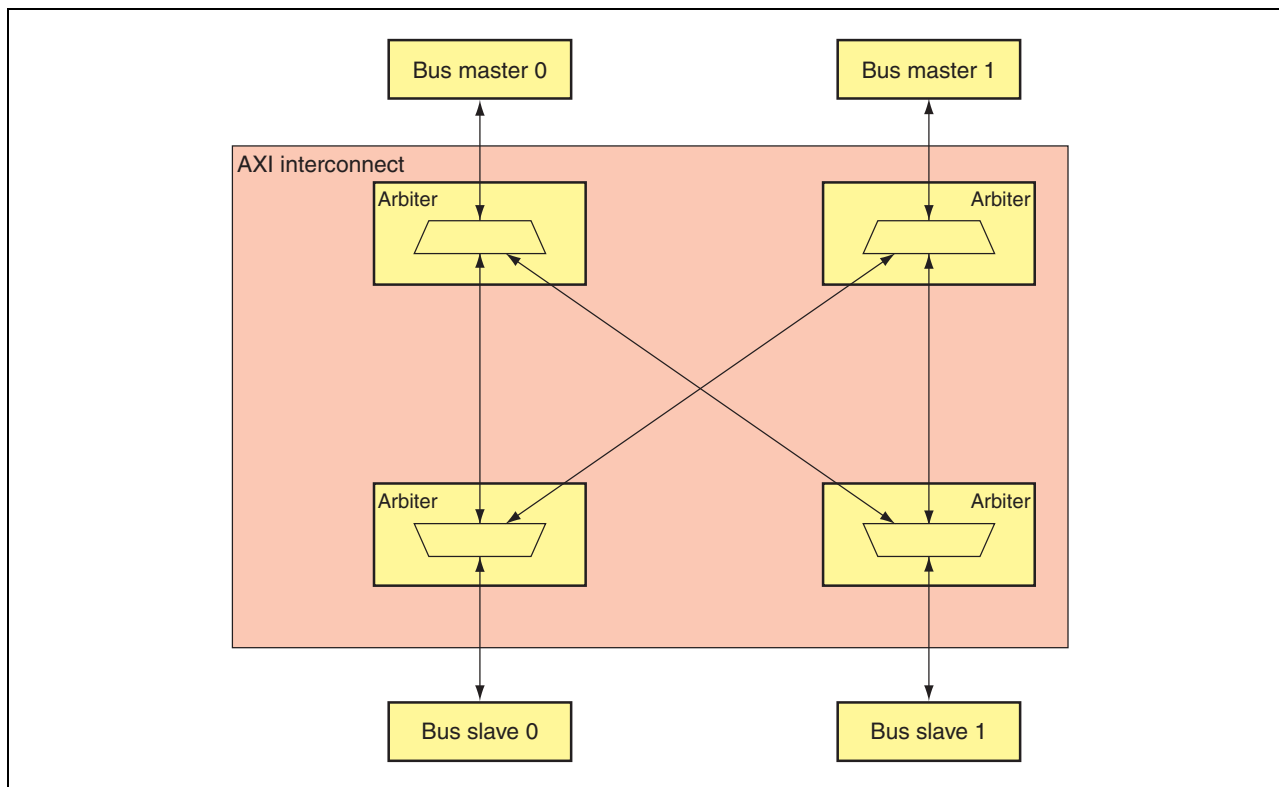


Figure 5.5 Conceptual Diagram of AXI Interconnect Configuration

### 5.6.2 Operation

In the AXI interconnect, the necessary wiring is prepared for connection between all bus masters and all bus slaves in all channels. When bus masters and slaves access bus slaves and masters, transfer will proceed after bus arbitration by the arbiter. The bus mastership priority changes in a round-robin manner. When multiple bus masters or slaves access different bus slaves or masters, multiple accesses can be executed in parallel. However, when multiple bus masters or slaves access a single bus slave or master at the same time, the bus arbiter executes bus arbitration. When a bus master or slave cannot obtain the bus mastership, it enters a wait state until the bus master or slave that has the bus mastership completes transfer unless the access destination bus master or slave cannot accept multiple transfers. When the destination bus master or slave can accept multiple transfers, bus arbitration is done again with the next transfer timing.

## 5.7 Bus Bridges

Access from the north main bus to the south main bus is executed through a bus bridge. There are two bus bridges and which bus bridge is used is determined depending on the slave area to be accessed. For assignment of the slave areas to be accessed, see Table 5.5, Address Map.

Each bus bridge can accept up to eight transfers at the same time. Out-of-order transfer is also supported. Therefore, when access to low-speed external ROM and access to on-chip high-speed RAM from different bus masters occur sequentially in this order, the on-chip RAM access can be done without waiting for completion of the previous external ROM access completion.



## 5.8 AXI Protocol Control Signals

The AXI protocol control signals can be set as desired for each master module. For details of the AXI protocol control signals, refer to the AMBA AXI Protocol Specification prepared by ARM Ltd.

### (1) Cache Control Signals (ARCACHE[3:0] and AWCACHE[3:0])

Use the AXI bus control register (AXIBUSCTL) to make settings of the ARCACHE[3:0] and AWCACHE[3:0] signals for each master module. Be sure to make settings while the target master module does not use the AXI bus.

### (2) Reponse Signals (RRESP[1:0] and BRESP[1:0])

Use the AXI bus response error status register (AXIRERRST) to read the RRESP[1:0] and BRESP[1:0] signals received by each master module. The register value is updated when a response error occurs. The status register value can be cleared to 00 through the AXI bus response error clear register (AXIRERRCLR).

In addition, enabling interrupts through the AXI bus response error interrupt control register (AXIRERRCTL) allows an interrupt to be generated when a response error occurs.

This interrupt should be used only for debugging purposes. Make sure that no response error occurs during system operation.

## 5.9 Write Buffers

A write buffer is provided at each connection between the north main bus and a peripheral bus and at each connection between the media local bus and the north main bus, and CoreSight and the north main bus. When the AWCACHE[1:0] cache control signals are set to cache-enabled or buffer-enabled (either of the AWCACHE[1:0] signals is set to 1), the write buffer sends a write completion response to the bus master before accessing the slave area under the write buffer. At this time, even if a slave error response is returned from the slave area to be accessed, it is ignored.

## 5.10 Register Descriptions

Table 5.6 shows the registers related to the internal bus.

**Table 5.6 Register Configuration**

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Remap register	RMPR	R/W	H'0000_0003	H'FCFE_1A00	32
AXI bus control register 0	AXIBUSCTL0	R/W	H'0000_0000	H'FCFE_1A04	32
AXI bus control register 1	AXIBUSCTL1	R/W	H'0000_0000	H'FCFE_1A08	32
AXI bus control register 2	AXIBUSCTL2	R/W	H'0000_0000	H'FCFE_1A0C	32
AXI bus control register 3	AXIBUSCTL3	R/W	H'0000_0000	H'FCFE_1A10	32
AXI bus control register 4	AXIBUSCTL4	R/W	H'0000_0000	H'FCFE_1A14	32
AXI bus control register 5	AXIBUSCTL5	R/W	H'0000_0000	H'FCFE_1A18	32
AXI bus control register 6	AXIBUSCTL6	R/W	H'0000_0000	H'FCFE_1A1C	32
AXI bus control register 7	AXIBUSCTL7	R/W	H'0000_0000	H'FCFE_1A20	32
AXI bus control register 8	AXIBUSCTL8	R/W	H'0000_0000	H'FCFE_1A24	32
AXI bus control register 9	AXIBUSCTL9	R/W	H'0000_0000	H'FCFE_1A28	32
AXI bus control register 10	AXIBUSCTL10	R/W	H'0000_0000	H'FCFE_1A2C	32
AXI bus response error interrupt control register 0	AXIRERRCTL0	R/W	H'0000_0000	H'FCFE_1A30	32
AXI bus response error interrupt control register 1	AXIRERRCTL1	R/W	H'0000_0000	H'FCFE_1A34	32
AXI bus response error interrupt control register 2	AXIRERRCTL2	R/W	H'0000_0000	H'FCFE_1A38	32
AXI bus response error interrupt control register 3	AXIRERRCTL3	R/W	H'0000_0000	H'FCFE_1A3C	32
AXI bus response error status register 0	AXIRERRST0	R/W	H'0000_0000	H'FCFE_1A40	32
AXI bus response error status register 1	AXIRERRST1	R/W	H'0000_0000	H'FCFE_1A44	32
AXI bus response error status register 2	AXIRERRST2	R/W	H'0000_0000	H'FCFE_1A48	32
AXI bus response error status register 3	AXIRERRST3	R/W	H'0000_0000	H'FCFE_1A4C	32
AXI bus response error clear register 0	AXIRERRCLR0	R/W	H'0000_0000	H'FCFE_1A50	32
AXI bus response error clear register 1	AXIRERRCLR1	R/W	H'0000_0000	H'FCFE_1A54	32
AXI bus response error clear register 2	AXIRERRCLR2	R/W	H'0000_0000	H'FCFE_1A58	32
AXI bus response error clear register 3	AXIRERRCLR3	R/W	H'0000_0000	H'FCFE_1A5C	32

### 5.10.1 Remap Register (RMPR)

This register controls the address remapping function.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	AXI128	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	AXI128	1	R/W	AXI128 Address Remapping This bit enables or disables allocation of addresses H'0000_0000 to H'009F_FFFF to on-chip RAM pages 0 to 4. 0: Address remapping is enabled. 1: Address remapping is disabled.
0	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.

## 5.10.2 AXI Bus Control Register 0 (AXIBUSCTL0)

This register controls the cache operation for the JPEG codec unit and Ethernet controller.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	JCUARCACHE[3:0]				—	—	—	—	JCUAWCACHE[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	ETHARCACHE[3:0]				—	—	—	—	ETHAWCACHE[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 24	JCUARCA CHE [3:0]	0000	R/W	ARCACHE[3:0] Signals for JPEG Codec Unit These bits specify the system cache operation when the JPEG codec unit performs read access. The values of these bits are used as the ARCACHE[3:0] signals for the JPEG codec unit. Modify the values of these bits only while the JPEG codec unit does not use the internal bus.
23 to 20	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
19 to 16	JCUAWCA CHE [3:0]	0000	R/W	AWCACHE[3:0] Signals for JPEG Codec Unit These bits specify the system cache operation when the JPEG codec unit performs write access. The values of these bits are used as the AWCACHE[3:0] signals for the JPEG codec unit. Modify the values of these bits only while the JPEG codec unit does not use the internal bus.
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 8	ETHARCA CHE [3:0]	0000	R/W	ARCACHE[3:0] Signals for Ethernet Controller These bits specify the system cache operation when the Ethernet controller performs read access. The values of these bits are used as the ARCACHE[3:0] signals for the Ethernet controller. Modify the values of these bits only while the Ethernet controller does not use the internal bus.
7 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	ETHAWCA CHE [3:0]	0000	R/W	AWCACHE[3:0] Signals for Ethernet Controller These bits specify the system cache operation when the Ethernet controller performs write access. The values of these bits are used as the AWCACHE[3:0] signals for the Ethernet controller. Modify the values of these bits only while the Ethernet controller does not use the internal bus.

### 5.10.3 AXI Bus Control Register 1 (AXIBUSCTL1)

This register controls the cache operation for the image renderer (IMR-LS2).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	IMR20ARCACHE[3:0]				—	—	—	—	IMR20AWCACHE[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	IMR21ARCACHE[3:0]				—	—	—	—	IMR21AWCACHE[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 24	IMR20ARCACHE[3:0]	0000	R/W	ARCACHE[3:0] Signals for Image Renderer (IMR-LS2) Channel 0 These bits specify the system cache operation when image renderer (IMR-LS2) channel 0 performs read access. The values of these bits are used as the ARCACHE[3:0] signals for image renderer (IMR-LS2) channel 0. Modify the values of these bits only while image renderer (IMR-LS2) channel 0 does not use the internal bus.
23 to 20	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
19 to 16	IMR20AWCACHE[3:0]	0000	R/W	AWCACHE[3:0] Signals for Image Renderer (IMR-LS2) Channel 0 These bits specify the system cache operation when image renderer (IMR-LS2) channel 0 performs write access. The values of these bits are used as the AWCACHE[3:0] signals for image renderer (IMR-LS2) channel 0. Modify the values of these bits only while image renderer (IMR-LS2) channel 0 does not use the internal bus.
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 8	IMR21ARCACHE[3:0]	0000	R/W	ARCACHE[3:0] Signals for Image Renderer (IMR-LS2) Channel 1 These bits specify the system cache operation when image renderer (IMR-LS2) channel 1 performs read access. The values of these bits are used as the ARCACHE[3:0] signals for image renderer (IMR-LS2) channel 1. Modify the values of these bits only while image renderer (IMR-LS2) channel 1 does not use the internal bus.
7 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	IMR21AWCACHE[3:0]	0000	R/W	AWCACHE[3:0] Signals for Image Renderer (IMR-LS2) Channel 1 These bits specify the system cache operation when image renderer (IMR-LS2) channel 1 performs write access. The values of these bits are used as the AWCACHE[3:0] signals for image renderer (IMR-LS2) channel 1. Modify the values of these bits only while image renderer (IMR-LS2) channel 1 does not use the internal bus.

### 5.10.4 AXI Bus Control Register 2 (AXIBUSCTL2)

This register controls the cache operation for the image renderer for display (IMR-LSD) and capture engine unit.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	IMRDARCACHE[3:0]				—	—	—	—	IMRDAWCACHE[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	CEUAWCACHE[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 24	IMRDARCA CHE [3:0]	0000	R/W	ARCACHE[3:0] Signals for Image Renderer for Display (IMR-LSD) These bits specify the system cache operation when the image renderer for display (IMR-LSD) performs read access. The values of these bits are used as the ARCACHE[3:0] signals for the image renderer for display (IMR-LSD). Modify the values of these bits only while the image renderer for display (IMR-LSD) does not use the internal bus.
23 to 20	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
19 to 16	IMRDAWC ACHE [3:0]	0000	R/W	AWCACHE[3:0] Signals for Image Renderer for Display (IMR-LSD) These bits specify the system cache operation when the image renderer for display (IMR-LSD) performs write access. The values of these bits are used as the AWCACHE[3:0] signals for the image renderer for display (IMR-LSD). Modify the values of these bits only while the image renderer for display (IMR-LSD) does not use the internal bus.
15 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	CEUAWCA CHE [3:0]	0000	R/W	AWCACHE[3:0] Signals for Capture Engine Unit These bits specify the system cache operation when capture engine unit performs write access. The values of these bits are used as the AWCACHE[3:0] signals for capture engine unit. Modify the values of these bits only while capture engine unit does not use the internal bus.

### 5.10.5 AXI Bus Control Register 3 (AXIBUSCTL3)

This register controls the cache operation for the OpenVG™-compliant Renesas graphics processor.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	
	—	—	—	—	RGP640ARCACHE[3:0]				—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	—	—	—	—	RGP641ARCACHE[3:0]				—	—	—	—	RGP641AWCACHE[3:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W	

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 24	RGP640AR CACHE [3:0]	0000	R/W	ARCACHE[3:0] Signals for OpenVG™-Compliant Renesas Graphics Processor AXI640 Bus These bits specify the system cache operation when the AXI640 bus of the OpenVG™-compliant Renesas graphics processor performs read access. The values of these bits are used as the ARCACHE[3:0] signals for the AXI640 bus of the OpenVG™-compliant Renesas graphics processor. Modify the values of these bits only while the OpenVG™-compliant Renesas graphics processor does not use the internal bus.
23 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 8	RGP641AR CACHE [3:0]	0000	R/W	ARCACHE[3:0] Signals for OpenVG™-Compliant Renesas Graphics Processor AXI641 Bus These bits specify the system cache operation when the AXI641 bus of the OpenVG™-compliant Renesas graphics processor performs read access. The values of these bits are used as the ARCACHE[3:0] signals for the AXI641 bus of the OpenVG™-compliant Renesas graphics processor. Modify the values of these bits only while the OpenVG™-compliant Renesas graphics processor does not use the internal bus.
7 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	RGP641A WCACHE [3:0]	0000	R/W	AWCACHE[3:0] Signals for OpenVG™-Compliant Renesas Graphics Processor AXI641 Bus These bits specify the system cache operation when the AXI641 bus of the OpenVG™-compliant Renesas graphics processor performs write access. The values of these bits are used as the AWCACHE[3:0] signals for the AXI641 bus of the OpenVG™-compliant Renesas graphics processor. Modify the values of these bits only while the OpenVG™-compliant Renesas graphics processor does not use the internal bus.

### 5.10.6 AXI Bus Control Register 4 (AXIBUSCTL4)

This register controls the cache operation for the OpenVG™-compliant Renesas graphics processor.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	RGB1282ARCACHE[3:0]				—	—	—	—	RGB1282AWCACHE[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	RGP1280ARCACHE[3:0]				—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 24	RGB1282 ARCACHE [3:0]	0000	R/W	ARCACHE[3:0] Signals for OpenVG™-Compliant Renesas Graphics Processor AXI1282 Bus These bits specify the system cache operation when the AXI1282 bus of the OpenVG™-compliant Renesas graphics processor performs read access. The values of these bits are used as the ARCACHE[3:0] signals for the AXI1282 bus of the OpenVG™-compliant Renesas graphics processor. Modify the values of these bits only while the OpenVG™-compliant Renesas graphics processor does not use the internal bus.
23 to 20	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
19 to 16	RGB1282 AWCACHE [3:0]	0000	R/W	AWCACHE[3:0] Signals for OpenVG™-Compliant Renesas Graphics Processor AXI1282 Bus These bits specify the system cache operation when the AXI1282 bus of the OpenVG™-compliant Renesas graphics processor performs write access. The values of these bits are used as the AWCACHE[3:0] signals for the AXI1282 bus of the OpenVG™-compliant Renesas graphics processor. Modify the values of these bits only while the OpenVG™-compliant Renesas graphics processor does not use the internal bus.
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 8	RGP1280A RCACHE [3:0]	0000	R/W	ARCACHE[3:0] Signals for OpenVG™-Compliant Renesas Graphics Processor AXI1280 Bus These bits specify the system cache operation when the AXI1280 bus of the OpenVG™-compliant Renesas graphics processor performs read access. The values of these bits are used as the ARCACHE[3:0] signals for the AXI1280 bus of the OpenVG™-compliant Renesas graphics processor. Modify the values of these bits only while the OpenVG™-compliant Renesas graphics processor does not use the internal bus.
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.



### 5.10.7 AXI Bus Control Register 5 (AXIBUSCTL5)

This register controls the cache operation for the OpenVG™-compliant Renesas graphics processor and media local bus.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	RGP1281ARCACHE[3:0]				—	—	—	—	RGP1281AWCACHE[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MLBAXCACHE [1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 24	RGP1281A RCACHE [3:0]	0000	R/W	ARCACHE[3:0] Signals for OpenVG™-Compliant Renesas Graphics Processor AXI1281 Bus These bits specify the system cache operation when the AXI1281 bus of the OpenVG™-compliant Renesas graphics processor performs read access. The values of these bits are used as the ARCACHE[3:0] signals for the AXI1281 bus of the OpenVG™-compliant Renesas graphics processor. Modify the values of these bits only while the OpenVG™-compliant Renesas graphics processor does not use the internal bus.
23 to 20	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
19 to 16	RGP1281A WCACHE [3:0]	0000	R/W	AWCACHE[3:0] Signals for OpenVG™-Compliant Renesas Graphics Processor AXI1281 Bus These bits specify the system cache operation when the AXI1281 bus of the OpenVG™-compliant Renesas graphics processor performs write access. The values of these bits are used as the AWCACHE[3:0] signals for the AXI1281 bus of the OpenVG™-compliant Renesas graphics processor. Modify the values of these bits only while the OpenVG™-compliant Renesas graphics processor does not use the internal bus.
15 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	MLBAXCA CHE [1:0]	00	R/W	AWCACHE[3:0] and ARCACHE[3:0] Signals for Media Local Bus These bits specify the system cache operation when the media local bus performs read or write access. The values of these bits are used as the AWCACHE[3:0] and ARCACHE[3:0] signals for the media local bus. The MLBAXCACHE[0] value is used as ARCACHE[0] and AWCACHE[0] without change. When MLBAXCACHE[1] = 0, the ARCACHE[3:1] and AWCACHE[3:1] signals are all set to 0. When MLBAXCACHE[1] = 1, the ARCACHE[3:1] and AWCACHE[3:1] signals are all set to 1. Modify the values of these bits only while the media local bus does not use the internal bus.

### 5.10.8 AXI Bus Control Register 6 (AXIBUSCTL6)

This register controls the cache operation for video display controller 5.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	VDC501ARCACHE[3:0]				—	—	—	—	VDC501AWCACHE[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	VDC502ARCACHE[3:0]				—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 24	VDC501AR CACHE [3:0]	0000	R/W	ARCACHE[3:0] Signals for Video Display Controller 5 Channel 0 IV3-BUS These bits specify the system cache operation when the IV3-BUS in channel 0 of video display controller 5 performs read access. The values of these bits are used as the ARCACHE[3:0] signals for the IV3-BUS in channel 0 of video display controller 5. Modify the values of these bits only while channel 0 of video display controller 5 does not use the internal bus.
23 to 20	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
19 to 16	VDC501AW CACHE [3:0]	0000	R/W	AWCACHE[3:0] Signals for Video Display Controller 5 Channel 0 IV1-BUS These bits specify the system cache operation when the IV1-BUS in channel 0 of video display controller 5 performs write access. The values of these bits are used as the AWCACHE[3:0] signals for the IV1-BUS in channel 0 of video display controller 5. Modify the values of these bits only while channel 0 of video display controller 5 does not use the internal bus.
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 8	VDC502AR CACHE [3:0]	0000	R/W	ARCACHE[3:0] Signals for Video Display Controller 5 Channel 0 IV5-BUS These bits specify the system cache operation when the IV5-BUS in channel 0 of video display controller 5 performs read access. The values of these bits are used as the ARCACHE[3:0] signals for the IV5-BUS in channel 0 of video display controller 5. Modify the values of these bits only while channel 0 of video display controller 5 does not use the internal bus.
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

### 5.10.9 AXI Bus Control Register 7 (AXIBUSCTL7)

This register controls the cache operation for video display controller 5.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	VDC503ARCACHE[3:0]				—	—	—	—	VDC503AWCACHE[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	VDC504ARCACHE[3:0]				—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 24	VDC503AR CACHE [3:0]	0000	R/W	ARCACHE[3:0] Signals for Video Display Controller 5 Channel 0 IV4-BUS These bits specify the system cache operation when the IV4-BUS in channel 0 of video display controller 5 performs read access. The values of these bits are used as the ARCACHE[3:0] signals for the IV4-BUS in channel 0 of video display controller 5. Modify the values of these bits only while channel 0 of video display controller 5 does not use the internal bus.
23 to 20	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
19 to 16	VDC503AW CACHE [3:0]	0000	R/W	AWCACHE[3:0] Signals for Video Display Controller 5 Channel 0 IV2-BUS These bits specify the system cache operation when the IV2-BUS in channel 0 of video display controller 5 performs write access. The values of these bits are used as the AWCACHE[3:0] signals for the IV2-BUS in channel 0 of video display controller 5. Modify the values of these bits only while channel 0 of video display controller 5 does not use the internal bus.
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 8	VDC504AR CACHE [3:0]	0000	R/W	ARCACHE[3:0] Signals for Video Display Controller 5 Channel 0 IV6-BUS These bits specify the system cache operation when the IV6-BUS in channel 0 of video display controller 5 performs read access. The values of these bits are used as the ARCACHE[3:0] signals for the IV6-BUS in channel 0 of video display controller 5. Modify the values of these bits only while channel 0 of video display controller 5 does not use the internal bus.
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

### 5.10.10 AXI Bus Control Register 8 (AXIBUSCTL8)

This register controls the cache operation for video display controller 5.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	VDC505ARCACHE[3:0]				—	—	—	—	VDC505AWCACHE[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	VDC511ARCACHE[3:0]				—	—	—	—	VDC511AWCACHE[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 24	VDC505AR CACHE [3:0]	0000	R/W	ARCACHE[3:0] Signals for Video Display Controller 5 Channel 0 IV8-BUS These bits specify the system cache operation when the IV8-BUS in channel 0 of video display controller 5 performs read access. The values of these bits are used as the ARCACHE[3:0] signals for the IV8-BUS in channel 0 of video display controller 5. Modify the values of these bits only while channel 0 of video display controller 5 does not use the internal bus.
23 to 20	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
19 to 16	VDC505AW CACHE [3:0]	0000	R/W	AWCACHE[3:0] Signals for Video Display Controller 5 Channel 0 IV7-BUS These bits specify the system cache operation when the IV7-BUS in channel 0 of video display controller 5 performs write access. The values of these bits are used as the AWCACHE[3:0] signals for the IV7-BUS in channel 0 of video display controller 5. Modify the values of these bits only while channel 0 of video display controller 5 does not use the internal bus.
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 8	VDC511AR CACHE [3:0]	0000	R/W	ARCACHE[3:0] Signals for Video Display Controller 5 Channel 1 IV3-BUS These bits specify the system cache operation when the IV3-BUS in channel 1 of video display controller 5 performs read access. The values of these bits are used as the ARCACHE[3:0] signals for the IV3-BUS in channel 1 of video display controller 5. Modify the values of these bits only while channel 1 of video display controller 5 does not use the internal bus.
7 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	VDC511AW CACHE [3:0]	0000	R/W	AWCACHE[3:0] Signals for Video Display Controller 5 Channel 1 IV1-BUS These bits specify the system cache operation when the IV1-BUS in channel 1 of video display controller 5 performs write access. The values of these bits are used as the AWCACHE[3:0] signals for the IV1-BUS in channel 1 of video display controller 5. Modify the values of these bits only while channel 1 of video display controller 5 does not use the internal bus.

### 5.10.11 AXI Bus Control Register 9 (AXIBUSCTL9)

This register controls the cache operation for video display controller 5.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	
	—	—	—	—	VDC512ARCACHE[3:0]				—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	—	—	—	—	VDC513ARCACHE[3:0]				—	—	—	—	VDC513AWCACHE[3:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W	

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 24	VDC512AR CACHE [3:0]	0000	R/W	ARCACHE[3:0] Signals for Video Display Controller 5 Channel 1 IV5-BUS These bits specify the system cache operation when the IV5-BUS in channel 1 of video display controller 5 performs read access. The values of these bits are used as the ARCACHE[3:0] signals for the IV5-BUS in channel 1 of video display controller 5. Modify the values of these bits only while channel 1 of video display controller 5 does not use the internal bus.
23 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 8	VDC513AR CACHE [3:0]	0000	R/W	ARCACHE[3:0] Signals for Video Display Controller 5 Channel 1 IV4-BUS These bits specify the system cache operation when the IV4-BUS in channel 1 of video display controller 5 performs read access. The values of these bits are used as the ARCACHE[3:0] signals for the IV4-BUS in channel 1 of video display controller 5. Modify the values of these bits only while channel 1 of video display controller 5 does not use the internal bus.
7 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	VDC513AW CACHE [3:0]	0000	R/W	AWCACHE[3:0] Signals for Video Display Controller 5 Channel 1 IV2-BUS These bits specify the system cache operation when the IV2-BUS in channel 1 of video display controller 5 performs write access. The values of these bits are used as the AWCACHE[3:0] signals for the IV2-BUS in channel 1 of video display controller 5. Modify the values of these bits only while channel 1 of video display controller 5 does not use the internal bus.

### 5.10.12 AXI Bus Control Register 10 (AXIBUSCTL10)

This register controls the cache operation for video display controller 5.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	
	—	—	—	—	VDC514ARCACHE[3:0]				—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	—	—	—	—	VDC515ARCACHE[3:0]				—	—	—	—	VDC515AWCACHE[3:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W	

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 24	VDC514AR CACHE [3:0]	0000	R/W	ARCACHE[3:0] Signals for Video Display Controller 5 Channel 1 IV6-BUS These bits specify the system cache operation when the IV6-BUS in channel 1 of video display controller 5 performs read access. The values of these bits are used as the ARCACHE[3:0] signals for the IV6-BUS in channel 1 of video display controller 5. Modify the values of these bits only while channel 1 of video display controller 5 does not use the internal bus.
23 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 8	VDC515AR CACHE [3:0]	0000	R/W	ARCACHE[3:0] Signals for Video Display Controller 5 Channel 1 IV8-BUS These bits specify the system cache operation when the IV8-BUS in channel 1 of video display controller 5 performs read access. The values of these bits are used as the ARCACHE[3:0] signals for the IV8-BUS in channel 1 of video display controller 5. Modify the values of these bits only while channel 1 of video display controller 5 does not use the internal bus.
7 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	VDC515AW CACHE [3:0]	0000	R/W	AWCACHE[3:0] Signals for Video Display Controller 5 Channel 1 IV7-BUS These bits specify the system cache operation when the IV7-BUS in channel 1 of video display controller 5 performs write access. The values of these bits are used as the AWCACHE[3:0] signals for the IV7-BUS in channel 1 of video display controller 5. Modify the values of these bits only while channel 1 of video display controller 5 does not use the internal bus.

### 5.10.13 AXI Bus Response Error Interrupt Control Register 0 (AXIRERRCTL0)

This register controls AXI bus response error interrupts.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	JCUR ERREN	—	—	—	ETHR ERREN	—	—	—	IMR20R ERREN	—	—	—	IMR21R ERREN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	IMDR ERREN	—	—	—	CEUR ERREN	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	JCURERR EN	0	R/W	Response Error Interrupt Enable for JPEG Codec Unit Enables or disables interrupt requests when access from the JPEG codec unit generates a response error. 0: Interrupt requests are disabled. 1: Interrupt requests are enabled.
27 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	ETHRERR EN	0	R/W	Response Error Interrupt Enable for Ethernet Controller Enables or disables interrupt requests when access from the Ethernet controller generates a response error. 0: Interrupt requests are disabled. 1: Interrupt requests are enabled.
23 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	IMR20RER REN	0	R/W	Response Error Interrupt Enable for Image Renderer (IMR-LS2) Channel 0 Enables or disables interrupt requests when access from image renderer (IMR-LS2) channel 0 generates a response error. 0: Interrupt requests are disabled. 1: Interrupt requests are enabled.
19 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	IMR21RER REN	0	R/W	Response Error Interrupt Enable for Image Renderer (IMR-LS2) Channel 1 Enables or disables interrupt requests when access from image renderer (IMR-LS2) channel 1 generates a response error. 0: Interrupt requests are disabled. 1: Interrupt requests are enabled.
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	IMDRRERR EN	0	R/W	Response Error Interrupt Enable for Image Renderer for Display (IMR-LSD) Enables or disables interrupt requests when access from the image renderer for display (IMR-LSD) generates a response error. 0: Interrupt requests are disabled. 1: Interrupt requests are enabled.
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	CEURERR EN	0	R/W	Response Error Interrupt Enable for Capture Engine Unit Enables or disables interrupt requests when access from capture engine unit generates a response error. 0: Interrupt requests are disabled. 1: Interrupt requests are enabled.
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

### 5.10.14 AXI Bus Response Error Interrupt Control Register 1 (AXIRERRCTL1)

This register controls AXI bus response error interrupts.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	RGP640 RERREN	—	—	—	RGP641 RERREN	—	—	—	RGP1282 RERREN	—	—	—	RGP1280 RERREN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	RGP1281 RERREN	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	RGP640 RERREN	0	R/W	Response Error Interrupt Enable for OpenVG™-Compliant Renesas Graphics Processor AXI640 Bus Enables or disables interrupt requests when access from the AXI640 bus of the OpenVG™-compliant Renesas graphics processor generates a response error. 0: Interrupt requests are disabled. 1: Interrupt requests are enabled.
27 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	RGP641 RERREN	0	R/W	Response Error Interrupt Enable for OpenVG™-Compliant Renesas Graphics Processor AXI641 Bus Enables or disables interrupt requests when access from the AXI641 bus of the OpenVG™-compliant Renesas graphics processor generates a response error. 0: Interrupt requests are disabled. 1: Interrupt requests are enabled.
23 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	RGP1282 RERREN	0	R/W	Response Error Interrupt Enable for OpenVG™-Compliant Renesas Graphics processor AXI1282 Bus Enables or disables interrupt requests when access from the AXI1282 bus of the OpenVG™-compliant Renesas graphics processor generates a response error. 0: Interrupt requests are disabled. 1: Interrupt requests are enabled.
19 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	RGP1280 RERREN	0	R/W	Response Error Interrupt Enable for OpenVG™-Compliant Renesas Graphics Processor AXI1280 Bus Enables or disables interrupt requests when access from the AXI1280 bus of the OpenVG™-compliant Renesas graphics processor generates a response error. 0: Interrupt requests are disabled. 1: Interrupt requests are enabled.
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	RGP1281 RERREN	0	R/W	Response Error Interrupt Enable for OpenVG™-Compliant Renesas Graphics Processor AXI1281 Bus Enables or disables interrupt requests when access from the AXI1281 bus of the OpenVG™-compliant Renesas graphics processor generates a response error. 0: Interrupt requests are disabled. 1: Interrupt requests are enabled.
11 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.



### 5.10.15 AXI Bus Response Error Interrupt Control Register 2 (AXIRERRCTL2)

This register controls AXI bus response error interrupts.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	VDC501 RERREN	—	—	—	VDC502 RERREN	—	—	—	VDC503 RERREN	—	—	—	VDC504 RERREN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	VDC505 RERREN	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	VDC501 RERREN	0	R/W	Response Error Interrupt Enable for Video Display Controller 5 Channel 0 IV1/3-BUS Enables or disables interrupt requests when access from the IV1-BUS or IV3-BUS in channel 0 of video display controller 5 generates a response error. 0: Interrupt requests are disabled. 1: Interrupt requests are enabled.
27 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	VDC502 RERREN	0	R/W	Response Error Interrupt Enable for Video Display Controller 5 Channel 0 IV5-BUS Enables or disables interrupt requests when access from the IV5-BUS in channel 0 of video display controller 5 generates a response error. 0: Interrupt requests are disabled. 1: Interrupt requests are enabled.
23 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	VDC503 RERREN	0	R/W	Response Error Interrupt Enable for Video Display Controller 5 Channel 0 IV2/4-BUS Enables or disables interrupt requests when access from the IV2-BUS or IV4-BUS in channel 0 of video display controller 5 generates a response error. 0: Interrupt requests are disabled. 1: Interrupt requests are enabled.
19 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	VDC504 RERREN	0	R/W	Response Error Interrupt Enable for Video Display Controller 5 Channel 0 IV6-BUS Enables or disables interrupt requests when access from the IV6-BUS in channel 0 of video display controller 5 generates a response error. 0: Interrupt requests are disabled. 1: Interrupt requests are enabled.
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	VDC505 RERREN	0	R/W	Response Error Interrupt Enable for Video Display Controller 5 Channel 0 IV7/8-BUS Enables or disables interrupt requests when access from the IV7-BUS or IV8-BUS in channel 0 of video display controller 5 generates a response error. 0: Interrupt requests are disabled. 1: Interrupt requests are enabled.
11 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

### 5.10.16 AXI Bus Response Error Interrupt Control Register 3 (AXIRERRCTL3)

This register controls AXI bus response error interrupts.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	VDC511 RERREN	—	—	—	VDC512 RERREN	—	—	—	VDC513 RERREN	—	—	—	VDC514 RERREN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	VDC515 RERREN	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	VDC511 RERREN	0	R/W	Response Error Interrupt Enable for Video Display Controller 5 Channel 1 IV1/3-BUS Enables or disables interrupt requests when access from the IV1-BUS or IV3-BUS in channel 1 of video display controller 5 generates a response error. 0: Interrupt requests are disabled. 1: Interrupt requests are enabled.
27 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	VDC512 RERREN	0	R/W	Response Error Interrupt Enable for Video Display Controller 5 Channel 1 IV5-BUS Enables or disables interrupt requests when access from the IV5-BUS in channel 1 of video display controller 5 generates a response error. 0: Interrupt requests are disabled. 1: Interrupt requests are enabled.
23 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	VDC513 RERREN	0	R/W	Response Error Interrupt Enable for Video Display Controller 5 Channel 1 IV2/4-BUS Enables or disables interrupt requests when access from the IV2-BUS or IV4-BUS in channel 1 of video display controller 5 generates a response error. 0: Interrupt requests are disabled. 1: Interrupt requests are enabled.
19 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	VDC514 RERREN	0	R/W	Response Error Interrupt Enable for Video Display Controller 5 Channel 1 IV6-BUS Enables or disables interrupt requests when access from the IV6-BUS in channel 1 of video display controller 5 generates a response error. 0: Interrupt requests are disabled. 1: Interrupt requests are enabled.
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	VDC515 RERREN	0	R/W	Response Error Interrupt Enable for Video Display Controller 5 Channel 1 IV7/8-BUS Enables or disables interrupt requests when access from the IV7-BUS or IV8-BUS in channel 1 of video display controller 5 generates a response error. 0: Interrupt requests are disabled. 1: Interrupt requests are enabled.
11 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

### 5.10.17 AXI Bus Response Error Status Register 0 (AXIRERRST0)

This register indicates occurrence of AXI bus response errors.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	JCURRESP [1:0]	JCUBRESP [1:0]	ETHRRESP [1:0]	ETHBRESP [1:0]	IMR20RRESP [1:0]	IMR20BRESP [1:0]	IMR21RRESP [1:0]	IMR21BRESP [1:0]								
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IMRDRRESP [1:0]	IMRDBRESP [1:0]	—	—	CEUBRESP [1:0]	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	JCURRESP [1:0]	00	R	RRESP[1:0] Signals for JPEG Codec Unit These bits indicate the RRESP[1:0] signals received by the JPEG codec unit. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
29, 28	JCUBRESP [1:0]	00	R	BRESP[1:0] Signals for JPEG Codec Unit These bits indicate the BRESP[1:0] signals received by the JPEG codec unit. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
27, 26	ETHRRESP [1:0]	00	R	RRESP[1:0] Signals for Ethernet Controller These bits indicate the RRESP[1:0] signals received by the Ethernet controller. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
25, 24	ETHBRESP [1:0]	00	R	BRESP[1:0] Signals for Ethernet Controller These bits indicate the BRESP[1:0] signals received by the Ethernet controller. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
23, 22	IMR20RRESP [1:0]	00	R	RRESP[1:0] Signals for Image Renderer (IMR-LS2) Channel 0 These bits indicate the RRESP[1:0] signals received by image renderer (IMR-LS2) channel 0. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
21, 20	IMR20BRESP [1:0]	00	R	BRESP[1:0] Signals for Image Renderer (IMR-LS2) Channel 0 These bits indicate the BRESP[1:0] signals received by image renderer (IMR-LS2) channel 0. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
19, 18	IMR21RRESP [1:0]	00	R	RRESP[1:0] Signals for Image Renderer (IMR-LS2) Channel 1 These bits indicate the RRESP[1:0] signals received by image renderer (IMR-LS2) channel 1. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR

Bit	Bit Name	Initial Value	R/W	Description
17, 16	IMR21BRESP[1:0]	00	R	BRESP[1:0] Signals for Image Renderer (IMR-LS2) Channel 1 These bits indicate the BRESP[1:0] signals received by image renderer (IMR-LS2) channel 1. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
15, 14	IMRDRRESP[1:0]	00	R	RRESP[1:0] Signals for Image Renderer for Display (IMR-LSD) These bits indicate the RRESP[1:0] signals received by the image renderer for display (IMR-LSD). The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
13, 12	IMRDBRESP[1:0]	00	R	BRESP[1:0] Signals for Image Renderer for Display (IMR-LSD) These bits indicate the BRESP[1:0] signals received by the image renderer for display (IMR-LSD). The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	CEUBRESP[1:0]	00	R	BRESP[1:0] Signals for Capture Engine Unit These bits indicate the BRESP[1:0] signals received by capture engine unit. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

### 5.10.18 AXI Bus Response Error Status Register 1 (AXIRERRST1)

This register indicates occurrence of AXI bus response errors.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RGP640RRESP [1:0]		—	—	RGP641RRESP [1:0]		RGP641BRESP [1:0]		RGP1282RRESP [1:0]		RGP1282BRESP [1:0]		RGP1280RRESP [1:0]		—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RGP1281RRESP [1:0]		RGP1281BRESP [1:0]		—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	RGP640RR ESP [1:0]	00	R	RRESP[1:0] Signals for OpenVG™-Compliant Renesas Graphics Processor AXI640 Bus These bits indicate the RRESP[1:0] signals received by the AXI640 bus of the OpenVG™-compliant Renesas graphics processor. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
29, 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27, 26	RGP641RR ESP [1:0]	00	R	RRESP[1:0] Signals for OpenVG™-Compliant Renesas Graphics Processor AXI641 Bus These bits indicate the RRESP[1:0] signals received by the AXI641 bus of the OpenVG™-compliant Renesas graphics processor. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
25, 24	RGP641BR ESP [1:0]	00	R	BRESP[1:0] Signals for OpenVG™-Compliant Renesas Graphics Processor AXI641 Bus These bits indicate the BRESP[1:0] signals received by the AXI641 bus of the OpenVG™-compliant Renesas graphics processor. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
23, 22	RGP1282 RRESP[1:0]	00	R	RRESP[1:0] Signals for OpenVG™-Compliant Renesas Graphics Processor AXI1282 Bus These bits indicate the RRESP[1:0] signals received by the AXI1282 bus of the OpenVG™-compliant Renesas graphics processor. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
21, 20	RGP1282 BRESP[1:0]	00	R	BRESP[1:0] Signals for OpenVG™-Compliant Renesas Graphics Processor AXI1282 Bus These bits indicate the BRESP[1:0] signals received by the AXI1282 bus of the OpenVG™-compliant Renesas graphics processor. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR

Bit	Bit Name	Initial Value	R/W	Description
19, 18	RGP1280R RESP[1:0]	00	R	RRESP[1:0] Signals for OpenVG™-Compliant Renesas Graphics Processor AXI1280 Bus These bits indicate the RRESP[1:0] signals received by the AXI1280 bus of the OpenVG™-compliant Renesas graphics processor. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
17, 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15, 14	RGP1281R RESP[1:0]	00	R	RRESP[1:0] Signals for OpenVG™-Compliant Renesas Graphics Processor AXI1281 Bus These bits indicate the RRESP[1:0] signals received by the AXI1281 bus of the OpenVG™-compliant Renesas graphics processor. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
13, 12	RGP1281B RESP[1:0]	00	R	BRESP[1:0] Signals for OpenVG™-Compliant Renesas Graphics Processor AXI1281 Bus These bits indicate the BRESP[1:0] signals received by the AXI1281 bus of the OpenVG™-compliant Renesas graphics processor. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
11 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

### 5.10.19 AXI Bus Response Error Status Register 2 (AXIRERRST2)

This register indicates occurrence of AXI bus response errors.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	VDC501RRESP [1:0]	VDC501BRESP [1:0]	VDC502RRESP [1:0]	—	—	—	—	—	VDC503RRESP [1:0]	VDC503BRESP [1:0]	VDC504RRESP [1:0]	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VDC505RRESP [1:0]	VDC505BRESP [1:0]	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	VDC501RR ESP [1:0]	00	R	RRESP[1:0] Signals for Video Display Controller 5 Channel 0 IV3-BUS These bits indicate the RRESP[1:0] signals received by the IV3-BUS in channel 0 of video display controller 5. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
29, 28	VDC501BR ESP [1:0]	00	R	BRESP[1:0] Signals for Video Display Controller 5 Channel 0 IV1-BUS These bits indicate the BRESP[1:0] signals received by the IV1-BUS in channel 0 of video display controller 5. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
27, 26	VDC502RR ESP [1:0]	00	R	RRESP[1:0] Signals for Video Display Controller 5 Channel 0 IV5-BUS These bits indicate the RRESP[1:0] signals received by the IV5-BUS in channel 0 of video display controller 5. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
25, 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23, 22	VDC503RR ESP [1:0]	00	R	RRESP[1:0] Signals for Video Display Controller 5 Channel 0 IV4-BUS These bits indicate the RRESP[1:0] signals received by the IV4-BUS in channel 0 of video display controller 5. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
21, 20	VDC503BR ESP [1:0]	00	R	BRESP[1:0] Signals for Video Display Controller 5 Channel 0 IV2-BUS These bits indicate the BRESP[1:0] signals received by the IV2-BUS in channel 0 of video display controller 5. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
19, 18	VDC504RR ESP [1:0]	00	R	RRESP[1:0] Signals for Video Display Controller 5 Channel 0 IV6-BUS These bits indicate the RRESP[1:0] signals received by the IV6-BUS in channel 0 of video display controller 5. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
17, 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
15, 14	VDC505RR ESP [1:0]	00	R	RRESP[1:0] Signals for Video Display Controller 5 Channel 0 IV8-BUS These bits indicate the RRESP[1:0] signals received by the IV8-BUS in channel 0 of video display controller 5. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
13, 12	VDC505BR ESP [1:0]	00	R	BRESP[1:0] Signals for Video Display Controller 5 Channel 0 IV7-BUS These bits indicate the BRESP[1:0] signals received by the IV7-BUS in channel 0 of video display controller 5. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
11 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

### 5.10.20 AXI Bus Response Error Status Register 3 (AXIRERRST3)

This register indicates occurrence of AXI bus response errors.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	VDC511RRESP [1:0]	VDC511BRESP [1:0]	VDC512RRESP [1:0]	—	—	—	—	—	VDC513RRESP [1:0]	VDC513BRESP [1:0]	VDC514RRESP [1:0]	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VDC515RRESP [1:0]	VDC515BRESP [1:0]	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	VDC511RR ESP [1:0]	00	R	RRESP[1:0] Signals for Video Display Controller 5 Channel 1 IV3-BUS These bits indicate the RRESP[1:0] signals received by the IV3-BUS in channel 1 of video display controller 5. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
29, 28	VDC511BR ESP [1:0]	00	R	BRESP[1:0] Signals for Video Display Controller 5 Channel 1 IV1-BUS These bits indicate the BRESP[1:0] signals received by the IV1-BUS in channel 1 of video display controller 5. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
27, 26	VDC512RR ESP [1:0]	00	R	RRESP[1:0] Signals for Video Display Controller 5 Channel 1 IV5-BUS These bits indicate the RRESP[1:0] signals received by the IV5-BUS in channel 1 of video display controller 5. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
25, 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.



Bit	Bit Name	Initial Value	R/W	Description
23, 22	VDC513RR ESP [1:0]	00	R	RRESP[1:0] Signals for Video Display Controller 5 Channel 1 IV4-BUS These bits indicate the RRESP[1:0] signals received by the IV4-BUS in channel 1 of video display controller 5. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
21, 20	VDC513BR ESP [1:0]	00	R	BRESP[1:0] Signals for Video Display Controller 5 Channel 1 IV2-BUS These bits indicate the BRESP[1:0] signals received by the IV2-BUS in channel 1 of video display controller 5. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
19, 18	VDC514RR ESP [1:0]	00	R	RRESP[1:0] Signals for Video Display Controller 5 Channel 1 IV6-BUS These bits indicate the RRESP[1:0] signals received by the IV6-BUS in channel 1 of video display controller 5. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
17, 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15, 14	VDC515RR ESP [1:0]	00	R	RRESP[1:0] Signals for Video Display Controller 5 Channel 1 IV8-BUS These bits indicate the RRESP[1:0] signals received by the IV8-BUS in channel 1 of video display controller 5. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
13, 12	VDC515BR ESP [1:0]	00	R	BRESP[1:0] Signals for Video Display Controller 5 Channel 1 IV7-BUS These bits indicate the BRESP[1:0] signals received by the IV7-BUS in channel 1 of video display controller 5. The values of these bits are updated when a response error occurs. 00: OKAY 10: SLVERR 11: DECERR
11 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

### 5.10.21 AXI Bus Response Error Clear Register 0 (AXIRERRCLR0)

This register clears the AXI bus response error status.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	JCU RRESP CLR	—	JCU BRESP CLR	—	ETH RRESP CLR	—	ETH BRESP CLR	—	IMR20 RRESP CLR	—	IMR20 BRESP CLR	—	IMR21 RRESP CLR	—	IMR21 BRESP CLR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R/W	R	R/W	R	R/W	R	R/W	R	R/W	R	R/W	R	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	IMRD RRESP CLR	—	IMRD BRESP CLR	—	—	—	—	CEU BRESP CLR	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R/W	R	R	R	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30	JCURRESP CLR	0	R/W	JCURRESP[1:0] Clear Writing 1 to this bit clears the JCURRESP[1:0] bits to 00. This bit is always read as 0.
29	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
28	JCUBRESP CLR	0	R/W	JCUBRESP[1:0] Clear Writing 1 to this bit clears the JCUBRESP[1:0] bits to 00. This bit is always read as 0.
27	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
26	ETHRRESP CLR	0	R/W	ETHRRESP[1:0] Clear Writing 1 to this bit clears the ETHRRESP[1:0] bits to 00. This bit is always read as 0.
25	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
24	ETHBRESP CLR	0	R/W	ETHBRESP[1:0] Clear Writing 1 to this bit clears the ETHBRESP[1:0] bits to 00. This bit is always read as 0.
23	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
22	IMR20RRES PCLR	0	R/W	IMR20RRESP[1:0] Clear Writing 1 to this bit clears the IMR20RRESP[1:0] bits to 00. This bit is always read as 0.
21	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
20	IMR20BRES PCLR	0	R/W	IMR20BRESP[1:0] Clear Writing 1 to this bit clears the IMR20BRESP[1:0] bits to 00. This bit is always read as 0.
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
18	IMR21RRES PCLR	0	R/W	IMR21RRESP[1:0] Clear Writing 1 to this bit clears the IMR21RRESP[1:0] bits to 00. This bit is always read as 0.
17	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
16	IMR21BRES PCLR	0	R/W	IMR21BRESP[1:0] Clear Writing 1 to this bit clears the IMR21BRESP[1:0] bits to 00. This bit is always read as 0.
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
14	IMDRRES PCLR	0	R/W	IMDRRESP[1:0] Clear Writing 1 to this bit clears the IMDRRESP[1:0] bits to 00. This bit is always read as 0.
13	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
12	IMRDBRES PCLR	0	R/W	IMRDBRESP[1:0] Clear Writing 1 to this bit clears the IMRDBRESP[1:0] bits to 00. This bit is always read as 0.
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	CEUBRESP CLR	0	R/W	CEUBRESP[1:0] Clear Writing 1 to this bit clears the CEUBRESP[1:0] bits to 00. This bit is always read as 0.
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

### 5.10.22 AXI Bus Response Error Clear Register 1 (AXIRERRCLR1)

This register clears the AXI bus response error status.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	RGP640 RRESP CLR	—	—	—	RGP641 RRESP CLR	—	RGP641 BRESP CLR	—	RGP1282 RRESP CLR	—	RGP1282 BRESP CLR	—	RGP1280 RRESP CLR	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R	R	R/W	R	R/W	R	R/W	R	R/W	R	R/W	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	RGP1281 RRESP CLR	—	RGP1281 BRESP CLR	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30	RGP640 RRESP CLR	0	R/W	RGP640RRESP[1:0] Clear Writing 1 to this bit clears the RGP640RRESP[1:0] bits to 00. This bit is always read as 0.
29 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26	RGP641 RRESP CLR	0	R/W	RGP641RRESP[1:0] Clear Writing 1 to this bit clears the RGP641RRESP[1:0] bits to 00. This bit is always read as 0.
25	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
24	RGP641 BRESP CLR	0	R/W	RGP641BRESP[1:0] Clear Writing 1 to this bit clears the RGP641BRESP[1:0] bits to 00. This bit is always read as 0.
23	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
22	RGP1282 RRESP CLR	0	R/W	RGP1282RRESP[1:0] Clear Writing 1 to this bit clears the RGP1282RRESP[1:0] bits to 00. This bit is always read as 0.
21	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
20	RGP1282 BRESP CLR	0	R/W	RGP1282BRESP[1:0] Clear Writing 1 to this bit clears the RGP1282BRESP[1:0] bits to 00. This bit is always read as 0.
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
18	RGP1280 RRESP CLR	0	R/W	RGP1280RRESP[1:0] Clear Writing 1 to this bit clears the RGP1280RRESP[1:0] bits to 00. This bit is always read as 0.
17 to 15	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
14	RGP1281 RRESP CLR	0	R/W	RGP1281RRESP[1:0] Clear Writing 1 to this bit clears the RGP1281RRESP[1:0] bits to 00. This bit is always read as 0.
13	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
12	RGP1281 BRESP CLR	0	R/W	RGP1281BRESP[1:0] Clear Writing 1 to this bit clears the RGP1281BRESP[1:0] bits to 00. This bit is always read as 0.
11 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

### 5.10.23 AXI Bus Response Error Clear Register 2 (AXIRERRCLR2)

This register clears the AXI bus response error status.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	VDC501 RRESP CLR	—	VDC501 BRESP CLR	—	VDC502 RRESP CLR	—	—	—	VDC503 RRESP CLR	—	VDC503 BRESP CLR	—	VDC504 RRESP CLR	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R/W	R	R/W	R	R	R	R/W	R	R/W	R	R/W	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	VDC505 RRESP CLR	—	VDC505 BRESP CLR	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30	VDC501 RRESP CLR	0	R/W	VDC501RRESP[1:0] Clear Writing 1 to this bit clears the VDC501RRESP[1:0] bits to 00. This bit is always read as 0.
29	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
28	VDC501 BRESP CLR	0	R/W	VDC501BRESP[1:0] Clear Writing 1 to this bit clears the VDC501BRESP[1:0] bits to 00. This bit is always read as 0.
27	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
26	VDC502 RRESP CLR	0	R/W	VDC502RRESP[1:0] Clear Writing 1 to this bit clears the VDC502RRESP[1:0] bits to 00. This bit is always read as 0.
25 to 23	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
22	VDC503RRESP CLR	0	R/W	VDC503RRESP[1:0] Clear Writing 1 to this bit clears the VDC503RRESP[1:0] bits to 00. This bit is always read as 0.
21	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
20	VDC503BRESP CLR	0	R/W	VDC503BRESP[1:0] Clear Writing 1 to this bit clears the VDC503BRESP[1:0] bits to 00. This bit is always read as 0.
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
18	VDC504RRESP CLR	0	R/W	VDC504RRESP[1:0] Clear Writing 1 to this bit clears the VDC504RRESP[1:0] bits to 00. This bit is always read as 0.
17 to 15	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
14	VDC505RRESP CLR	0	R/W	VDC505RRESP[1:0] Clear Writing 1 to this bit clears the VDC505RRESP[1:0] bits to 00. This bit is always read as 0.
13	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
12	VDC505BRESP CLR	0	R/W	VDC505BRESP[1:0] Clear Writing 1 to this bit clears the VDC505BRESP[1:0] bits to 00. This bit is always read as 0.
11 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

### 5.10.24 AXI Bus Response Error Clear Register 3 (AXIRERRCLR3)

This register clears the AXI bus response error status.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	VDC511RRESP CLR	—	VDC511BRESP CLR	—	VDC512RRESP CLR	—	—	—	VDC513RRESP CLR	—	VDC513BRESP CLR	—	VDC514RRESP CLR	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R/W	R	R/W	R	R	R	R/W	R	R/W	R	R/W	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	VDC515RRESP CLR	—	VDC515BRESP CLR	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30	VDC511RRESP CLR	0	R/W	VDC511RRESP[1:0] Clear Writing 1 to this bit clears the VDC511RRESP[1:0] bits to 00. This bit is always read as 0.
29	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
28	VDC511BRESP CLR	0	R/W	VDC511BRESP[1:0] Clear Writing 1 to this bit clears the VDC511BRESP[1:0] bits to 00. This bit is always read as 0.
27	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
26	VDC512 RRESP CLR	0	R/W	VDC512RRESP[1:0] Clear Writing 1 to this bit clears the VDC512RRESP[1:0] bits to 00. This bit is always read as 0.
25 to 23	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
22	VDC513 RRESP CLR	0	R/W	VDC513RRESP[1:0] Clear Writing 1 to this bit clears the VDC513RRESP[1:0] bits to 00. This bit is always read as 0.
21	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
20	VDC513 BRESP CLR	0	R/W	VDC513BRESP[1:0] Clear Writing 1 to this bit clears the VDC513BRESP[1:0] bits to 00. This bit is always read as 0.
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
18	VDC514 RRESP CLR	0	R/W	VDC514RRESP[1:0] Clear Writing 1 to this bit clears the VDC514RRESP[1:0] bits to 00. This bit is always read as 0.
17 to 15	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
14	VDC515 RRESP CLR	0	R/W	VDC515RRESP[1:0] Clear Writing 1 to this bit clears the VDC515RRESP[1:0] bits to 00. This bit is always read as 0.
13	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
12	VDC515 BRESP CLR	0	R/W	VDC515BRESP[1:0] Clear Writing 1 to this bit clears the VDC515BRESP[1:0] bits to 00. This bit is always read as 0.
11 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

## 5.11 Interrupt Request

When a decode error or a slave error occurs, an AXI bus response error interrupt request (PRRI) is issued.

An interrupt request is issued when a response is returned from the bus for which interrupt requests are enabled through the AXI bus response error interrupt control register (AXIRERRCTL). To check the response error type, read the AXI bus response error status register (AXIRERRST). To clear the interrupt request, clear the AXI bus response error status register through the AXI bus response error clear register (AXIRERRCLR).

This interrupt should be used only for debugging purposes. Make sure that no response error occurs during system operation.

## 6. Clock Pulse Generator

This LSI has a clock pulse generator that generates a CPU clock ( $I\phi$ ), image processing clock ( $G\phi$ ), internal bus clock ( $B\phi$ ), peripheral clock 1 ( $P1\phi$ ), and peripheral clock 0 ( $P0\phi$ ). The clock pulse generator consists of a crystal oscillator, PLL circuits, and divider circuits.

### 6.1 Features

- Clock types  
A CPU clock ( $I\phi$ ); an image processing clock ( $G\phi$ ); an internal bus clock ( $B\phi$ ); peripheral clock 1 ( $P1\phi = CKIO$ ) for the external bus interface; peripheral clock 0 ( $P0\phi$ ) for the on-chip peripheral modules
- Frequency change function  
CPU and image processing clock frequencies can be changed independently using the PLL (phase locked loop) circuits and divider circuits within this module. Frequencies are changed by software using frequency control register (FRQCR, FRQCR2) settings.
- Power-down mode control  
The clock can be stopped in sleep mode, software standby mode, and deep standby mode, and specific modules can be stopped using the module standby function. For details on clock control in the power-down modes, see **section 55, Power-Down Modes**.
- SSCG function  
The CPU's internal PLL (phase locked loop) circuit includes an SSCG (spread spectrum clock generator). The SSCG can be used to decrease the peak value of EMI (electromagnetic interference) noise by frequency modulation, that is, by slightly modulating the output frequency.  
The specification of the SSCG for this LSI is as follows.  
—Specification of SSCG
  - (1) Modulation waveform (modulation profile): Triangle wave
  - (2) Type of spreading: Down-spreading
  - (3) Modulation rate: -3.3% (clock mode 0), -3.1% (clock mode 1)
  - (4) Modulation frequency: 20.00 to 26.67 kHz (frequency on the EXTAL pin  $\div$  500)  
24.00 kHz (frequency on the USB\_X1 pin  $\times$  (1/4)  $\div$  500)

Figure 6.1 shows a block diagram of the clock pulse generator.

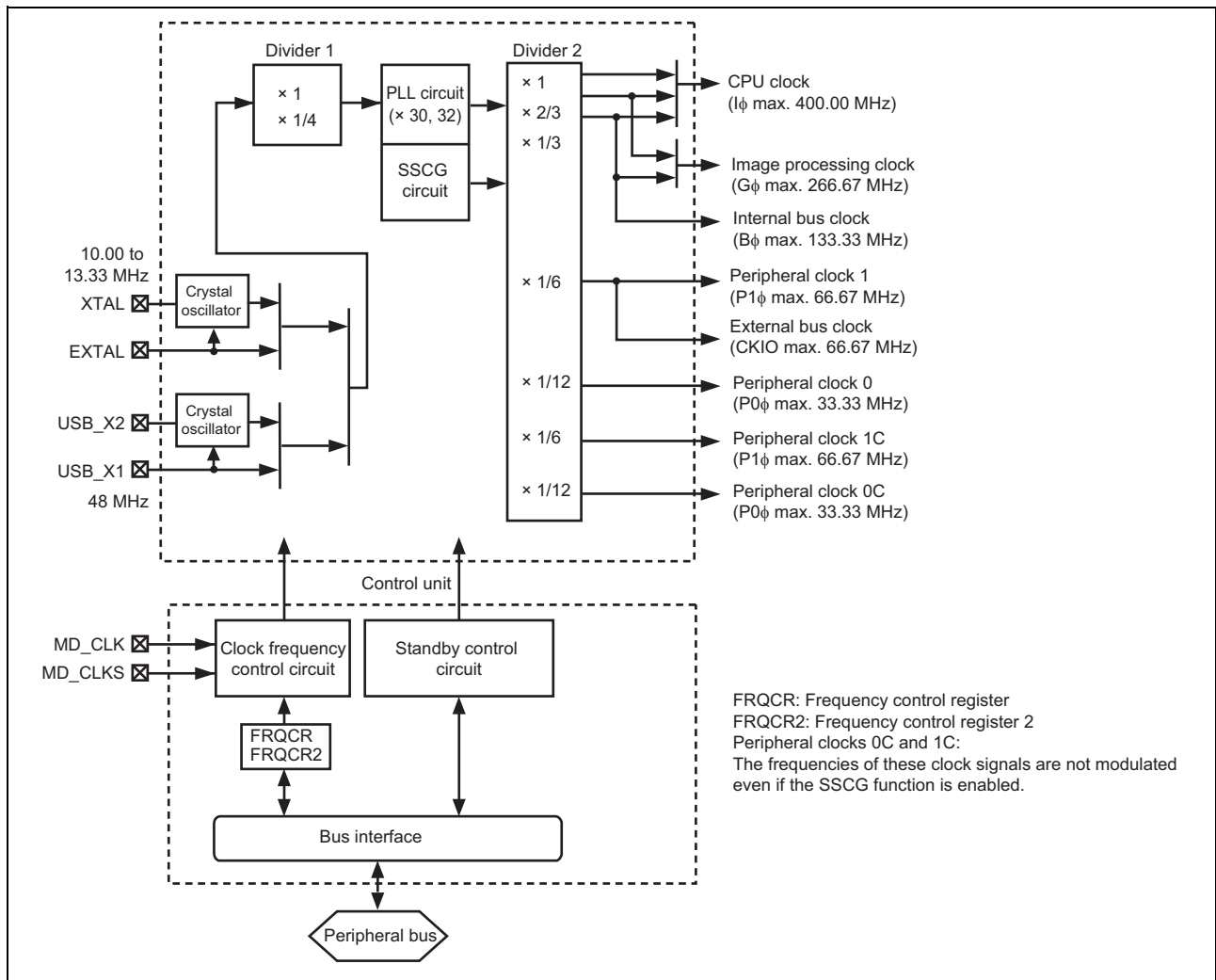


Figure 6.1 Block Diagram



The blocks of this module function as follows:

#### (1) Crystal Oscillator

A crystal oscillator is connected to the XTAL and EXTAL pins or to the USB\_X2 and USB\_X1 pins. Either the EXTAL or USB\_X1 pin is selected by the clock mode settings.

#### (2) PLL Circuit

The PLL circuit is capable of multiplying the frequency of the input clock signal from the EXTAL pin by 30. If the input clock signal from the USB\_X1 pin is selected, the frequency is multiplied by 32.

#### (3) Divider 1 and Divider 2

The ratio for frequency division by divider 1 is fixed to 1/1 for the input from the EXTAL pin and 1/4 for the input from the USB\_X1 pin.

Divider 2 generates a clock signal whose operating frequency can be used for the CPU clock, image processing clock, internal bus clock, peripheral clock 1, and peripheral clock 0. The division ratio of the CPU clock and the image processing clock is set by the frequency control register (FRQCR, FRQCR2). The division ratios for peripheral clocks 1 and 0 are fixed to 1/6 and 1/12, respectively.

#### (4) Clock Frequency Control Circuit

The clock frequency control circuit controls the clock frequency using the frequency control register (FRQCR, FRQCR2).

#### (5) Standby Control Circuit

The standby control circuit controls the states of the on-chip oscillation circuit and other modules during clock switching, or, software standby or deep standby mode.

In addition, the standby control register is provided to control the power-down mode of other modules. For details on the standby control register, see section 55, Power-Down Modes.

#### (6) Frequency Control Register (FRQCR, FRQCR2)

The frequency control register (FRQCR, FRQCR2) has control bits assigned for the following functions: clock output/non-output from the CKIO pin during software standby mode or deep standby mode and the frequency division ratio of the CPU clock ( $I\phi$ ) and the image processing clock ( $G\phi$ ).

#### (7) SSCG Circuit

Operation of the SSCG circuit is switched on or off (enabled or disabled) by the MD\_CLKS pin. When the SSCG function is disabled, all of the internal clock frequencies are fixed, i.e. not modulated. When the SSCG function is enabled, the frequencies of clock signals supplied to peripheral modules other than those listed below are modulated.

Peripheral modules to which non-modulated clock signals are supplied:

IEBus™ controller, multi-function timer pulse unit 2, serial communications interface with FIFO, CAN interface, OS timer, motor control PWM timer, sound generator, LIN interface, and serial communication interface.

## 6.2 Input/Output Pins

Table 6.1 lists the clock pulse generator pins and their functions.

**Table 6.1 Pin Configuration and Functions of the Clock Pulse Generator**

Pin Name	Symbol	I/O	Function
Mode control pin	MD_CLK	Input	Switches between the EXTAL input and the USB_X1 input.
	MD_CLKS	Input	Enables or disables the SSCG circuit.
Crystal input/output pins (clock input pins)	XTAL	Output	Connected to the crystal resonator. (Leave this pin open when the crystal resonator is not in use.)
	EXTAL	Input	Connected to the crystal resonator or used to input external clock.
	USB_X2	Output	Connected to the crystal resonator. (Leave this pin open when the crystal resonator is not in use.)
	USB_X1	Input	Connected to the crystal resonator or used to input external clock.
Clock output pin	CKIO	Output	Clock output pin.

### 6.3 Clock Mode

Table 6.2 indicates the input/output clock frequency. Table 6.3 shows the usable frequency ranges.

**Table 6.2 Input/Output Clock Frequency**

Mode	MD_CLK0 Pin Setting	Clock I/O Source	Output	Divider 1	PLL Circuit	CKIO Frequency
0	0	EXTAL/crystal resonator	CKIO	1	ON (x 30)	(EXTAL/crystal resonator) x 5
1	1	USB_X1/crystal resonator	CKIO	1/4	ON (x 32)	(USB_X1/crystal resonator) x 4/3

In clock mode 0, the clock signal is the input from the EXTAL pin or the crystal oscillator. The PLL circuit shapes waveforms and multiplies the frequency, and then supplies the clock to the LSI. The oscillating frequency for the crystal resonator and EXTAL pin input clock ranges from 10 to 13.33 MHz. The frequency range of CKIO is from 50 to 66.67 MHz.

In clock mode 1, the clock signal is the input from the USB\_X1 pin or the crystal oscillator. The PLL circuit shapes waveforms and multiplies the frequency, and then supplies the clock to the LSI. The oscillating frequency for the crystal resonator and USB\_X1 pin input clock is 48 MHz. The frequency of CKIO is 64 MHz.

When changing the frequency, be sure to set the standby\_mode\_en bit of the power control register in the PL310. For details on the register, see CoreLink Level 2 Cache Controller L2C-310 Technical Reference Manual issued by ARM Ltd.

**Table 6.3 Settable Frequency Ranges**

Mode	FRQCR Setting *1	FRQC R2 Setting *2	PLL Frequency Multiplier	PLL Circuit	Ratio of Internal Clock Frequencies (I : G : B : P1 : P0)*3	Selectable Frequency Range (MHz)						
						Input Clock*4	Output Clock (CKIO Pin)	CPU Clock (I $\phi$ )	Image processing clock (G $\phi$ )	Internal Bus Clock (B $\phi$ )	Peripheral Clock 1 (P1 $\phi$ )	Peripheral Clock 0 (P0 $\phi$ )
0	H'x035	H'0001	ON (x 30)		30 : 20 : 10 : 5 : 5/2	10.00 to 13.33	50.00 to 66.67	300.00 to 400.00	200.00 to 266.67	100.00 to 133.33	50.00 to 66.67	25.00 to 33.33
	H'x135	H'0001			20 : 20 : 10 : 5 : 5/2		200.00 to 266.67					
	H'x035	H'0003			30 : 10 : 10 : 5 : 5/2		300.00 to 400.00	100.00 to 133.33				
	H'x135	H'0003			20 : 10 : 10 : 5 : 5/2		200.00 to 266.67					
	H'x335	H'0003			10 : 10 : 10 : 5 : 5/2		100.00 to 133.33					
1	H'x035	H'0001	ON (x 32)		8 : 16/3 : 8/3 : 4/3 : 2/3	48.00	64.00	384.00	256.00	128.00	64.00	32.00
	H'x135	H'0001			16/3 : 16/3 : 8/3 : 4/3 : 2/3		256.00					
	H'x035	H'0003			8 : 8/3 : 8/3 : 4/3 : 2/3		384.00	128.00				
	H'x135	H'0003			16/3 : 8/3 : 8/3 : 4/3 : 2/3		256.00					
	H'x335	H'0003			8/3 : 8/3 : 8/3 : 4/3 : 2/3		128.00					

Note 1. x in the FRQCR register setting depends on the set value in bits 12, 13, and 14.

Note 2. When using an image renderer, image renderer for display, or Renesas graphics processor for OpenVG™, be sure to set the FRQCR2 register to H'0001.

Note 3. The ratio of clock frequencies, where the input clock frequency is assumed to be 1.

Note 4. In clock mode 0, the frequency of the EXTAL pin input clock or the crystal resonator.  
In clock mode 1, the frequency of the USB\_X1 pin input clock or the crystal resonator.

Caution: Do not use this LSI for frequency settings other than those in Table 6.3.

The clock source of the chip is switched by the setting of the MD\_CLK pin while the  $\overline{\text{RES}}$  pin is being held low. The following table shows the correspondence between clock source and pin settings.

**Table 6.4 Clock Source Selection**

MD_CLK Pin Setting	Clock Source
0	EXTAL/crystal resonator
1	USB_X1/crystal resonator

The SSCG function of the chip is switched on or off by the setting of the MD\_CLKS pin while the RES pin is being held low. The following table shows the correspondence between SSCG operation and pin settings. Note that the pin setting does not affect the PLL frequency multipliers and division ratios for individual clock signals.

**Table 6.5 SSCG Operation Setting**

MD_CLKS Pin Setting	SSCG Operation
0	Off
1	On

## 6.4 Register Descriptions

Table 6.6 shows the register configuration of the clock pulse generator.

**Table 6.6 Register Configuration**

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Frequency control register	FRQCR	R/W	H'0335	H'FCFE0010	16
	FRQCR2	R/W	H'0003	H'FCFE0014	16

### 6.4.1 Frequency Control Register (FRQCR)

FRQCR is a 16-bit readable/writable register used to specify whether a clock is output from the CKIO pin during normal operation mode, change of gain of crystal oscillator for the XTAL pin, software standby mode, deep standby mode, and standby mode cancellation. The register specifies the frequency division ratio for the CPU clock ( $I\phi$ ). FRQCR can be accessed in 16-bit units.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	CKO EN2	CKOEN[1:0]	-	-	IFC[1:0]	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	1	1	0	0	1	1	0	1	0	1
R/W:	R	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	CKOEN2	0	R/W	Clock Output Enable 2 Specifies whether the CKIO pin outputs clock signals or is fixed to the low level when the gain of the crystal oscillator for the XTAL pin is changed. If this bit is set to 1, the CKIO pin is fixed to the low level when the gain of the crystal oscillator for the XTAL pin is changed. Therefore, the malfunction of an external circuit caused by an unstable CKIO clock while changing the gain of the crystal oscillator for the XTAL pin can be prevented. 0: Unstable clock output 1: Low-level output
13, 12	CKOEN[1:0]	00	R/W	Clock Output Enable These bits specify whether the CKIO pin outputs clock signals, or is set to a fixed level or high impedance (Hi-Z) during normal operation mode, deep standby mode, software standby mode, or cancellation of standby mode. If these bits are set to 01, the CKIO pin is fixed at low during deep standby mode, software standby mode, or cancellation of software standby mode. Therefore, the malfunction of an external circuit caused by an unstable CKIO clock during cancellation of software standby mode can be prevented. Table 6.6 lists CKOEN[1:0] settings.
11, 10	—	00	R	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	IFC[1:0]	11	R/W	CPU Clock Frequency Division Ratio These bits specify the frequency division ratio of the CPU clock with respect to the output frequency of PLL circuit. Note: See section 6.5.1. 00: 1/1 time 01: 2/3 time 10: Reserved (setting prohibited) 11: 1/3 time
7, 6	—	00	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
5, 4	—	11	R	Reserved These bits are always read as 1. The write value should always be 1.
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
1	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
0	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.

**Table 6.7 CKOEN[1:0] Settings**

Setting	Normal Operation	Software Standby Mode	Deep Standby Mode*
00	Output	Output off (Hi-Z)	Output off (Hi-Z)
01	Output	Low-level output	Low-level output
10	Output	Output (unstable clock output)	Low-level or high-level output
11	Output off (Hi-Z)	Output off (Hi-Z)	Output off (Hi-Z)

Note: \* Note that the first cycle of the output CKIO clock may be missing after release from deep standby.

### 6.4.2 Frequency Control Register 2 (FRQCR2)

FRQCR2 is a 16-bit readable/writable register used to specify the frequency division ratio for the image processing clock (G $\phi$ ). FRQCR2 can be accessed in 16-bit units.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	GFC[1:0]	
initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 2	—	All 0	R	Reserved This bit is always read as 0. The write value should always be 0.
1, 0	GFC[1:0]	11	R/W	Image Processing Clock Frequency Division Ratio These bits specify the frequency division ratio of the image processing clock with respect to the output frequency of PLL circuit. Note: See section 6.5.1. 00: Reserved (setting prohibited) 01: 2/3 time 10: Reserved (setting prohibited) 11: 1/3 time

## 6.5 Changing the Frequency

The frequency of the CPU clock ( $I\phi$ ) and image processing clock ( $G\phi$ ) can be changed by changing the division rate of divider. The division rate can be changed by software through the frequency control register (FRQCR, FRQCR2).

### 6.5.1 Changing the Division Ratio

The division rate of divider can be changed by the following operation.

1. In the initial state, IFC[1:0] = B'11 and GFC[1:0] = B'11.
2. Set the desired value in the IFC[1:0] and GFC[1:0] bits. Note that if the wrong value is set, this LSI will malfunction. If the setting of FRQCR2 is H'0001, change FRQCR to H'x035 or H'x135 before making these settings. In this case, setting FRQCR to H'x335 is prohibited. The settings should be such that the clock frequency of  $G\phi$  is no higher than that of  $I\phi$ .
3. After the register bits (IFC[1:0] and GFC[1:0]) have been set, the clock is supplied of the new division ratio.

Note: When executing the WFI instruction after changing the frequency, be sure to read the frequency control registers (FRQCR and FRQCR2) to confirm that the new settings are in place and read the ISBUSY0 bit in the CPU status register (CPUSTS) to confirm that it is set to 0 beforehand.  
For the CPUSTS register, see section 55, Power-Down Modes.



## 6.6 Usage of the Clock Pins

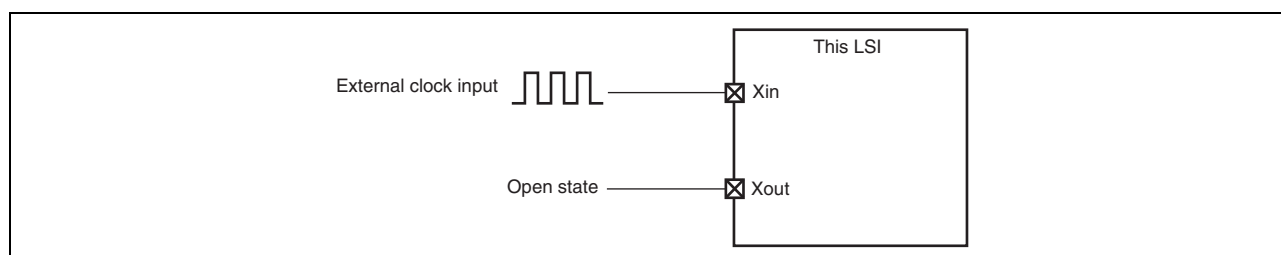
For the connection of a crystal resonator or the input of a clock signal, this LSI circuit has the pins listed in Table 6.8. With regard to these pins, take care on the following points. Furthermore, Xin pin and Xout pin are used in this section to refer to the pins listed in the table.

**Table 6.8 Clock Pins**

<b>Xin Pins (Used for Connection of a Crystal Resonator and Input of External Clock Signals)</b>	<b>Xout Pins (Used for Connection of a Crystal Resonator)</b>
EXTAL	XTAL
USB_X1	USB_X2
AUDIO_X1	AUDIO_X2
RTC_X1	RTC_X2
RTC_X3	RTC_X4
VIDEO_X1	VIDEO_X2

### 6.6.1 In the Case of Inputting an External Clock

An example of the connection of an external clock is shown in Figure 6.2. In cases where the Xout pin is left open state, take the parasitic capacitance as less than 10 pF.



**Figure 6.2 Example of the Connection of an External Clock**

### 6.6.2 In the Case of Using a Crystal Resonator

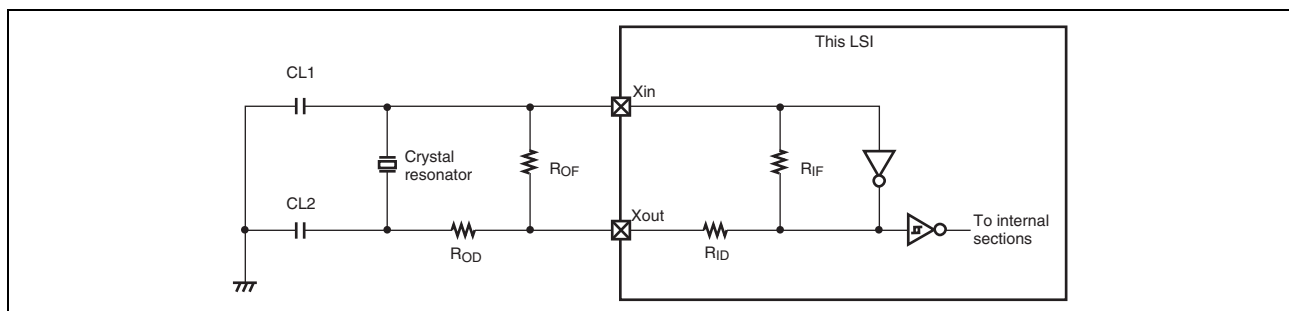
An example of the connection of crystal resonator is shown in Figure 6.3.

Place the crystal resonator and capacitors (CL1 and CL2) as close to pins Xin and Xout as possible. Furthermore, to avoid inductance so that oscillation is correct, use the points where the capacitors are connected to the crystal resonator in common and do not place wiring patterns close to these components.

Since the design of the user board is closely connected with the effective characteristics of the crystal resonator, refer to the example of connection of the crystal resonator that is introduced in this section and perform thorough evaluation on the user side as well. The rated value of the crystal resonator will vary with the floating capacitances and so on of the crystal resonator and mounted circuit, so proceed with decisions on the basis of full discussions with the maker of the crystal resonator. Ensure that voltages applied to the clock pins do not exceed the maximum rated values.

Although the feedback resistor is included in this LSI, an external feedback resistor may be required in some cases. This depends on the characteristics of the crystal resonator.

Set the parameters (of resistors and capacitors) with thorough evaluation on the user side.



**Figure 6.3** Example of the Connection of a Crystal Resonator

### 6.6.3 In the Case of Not Using the Clock Pin

In cases where the pins are not in use, fix the level on the Xin pin (pull it up or down, or connect it to the power-supply or ground level), and leave the Xout pin open state.

## 6.7 Oscillation Stabilizing Time

### 6.7.1 Oscillation Stabilizing Time of the On-chip Crystal Oscillator

In the case of using a crystal resonator, please wait longer than the oscillation stabilizing time at the following cases, to keep the oscillation stabilizing time of the on-chip crystal oscillator (In the case of inputting an external clock input, it is not necessary).

- Power on
- Releasing the software standby mode or deep standby mode by  $\overline{\text{RES}}$  pin
- Changing from halting oscillation to running oscillation by power-on reset or register setting (AUDIO\_X1)
- Changing from halting oscillation to running oscillation by register setting (RTC\_X1, RTC\_X3)
- Changing the gain of the on-chip crystal oscillator by  $\overline{\text{RES}}$  pin (EXTAL, RTC\_X3)

### 6.7.2 Oscillation Stabilizing Time of the PLL circuit

The clock from EXTAL is supplied to the PLL circuit. So, regardless of whether using a crystal resonator or inputting an external clock from EXTAL, please wait longer than the oscillation stabilizing time at the following cases, to keep the oscillation stabilizing time of the PLL circuit.

- Power on (in the case of using the crystal resonator)/start inputting external clock (in the case of inputting the external clock)
- Releasing the software standby mode or deep standby mode by  $\overline{\text{RES}}$  pin

#### [Remarks]

The oscillation stabilizing time is kept by the counter running in the LSI at the following cases.

- Releasing the software standby mode or deep standby mode by the other than  $\overline{\text{RES}}$  pin
- Changing the gain of the on-chip crystal oscillator by the register setting (EXTAL, RTC\_X3)

## 6.8 Notes on Board Design

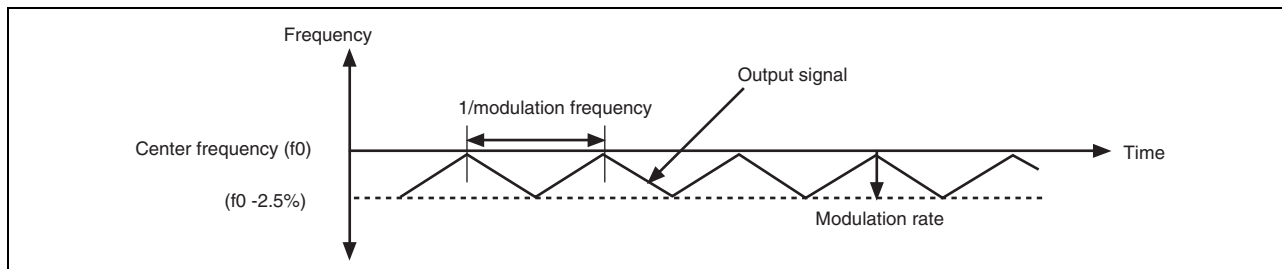
### 6.8.1 Note on Using a PLL Oscillation Circuit

In the PLLVcc connection pattern for the PLL, signal lines from the board power supply pins must be as short as possible and pattern width must be as wide as possible to reduce inductive interferences.

Since the analog power supply pins of the PLL are sensitive to the noise, the system may malfunction due to inductive interference at the other power supply pins. To prevent such malfunction, the analog power supply pins and the digital power supply pins Vcc and PVcc should not supply the same resources on the board if at all possible.

## 6.9 Definition of Modulation Rate and Frequency in the SSCG Specification

The SSCG circuit can be used to decrease the peak value of electromagnetic interference noise by frequency modulation, i.e. by slightly modulating the output frequency. In this case, the rate of change in the frequency and the size of the change to the input clock frequency are defined as the modulation rate and modulation frequency, respectively. Figure 6.4 shows the modulation rate and modulation frequency.



**Figure 6.4** Definition of SSCG Modulation Rate and Frequency

## 6.10 Clock Signals

### 6.10.1 Clock Signals for the System and Realtime Clock

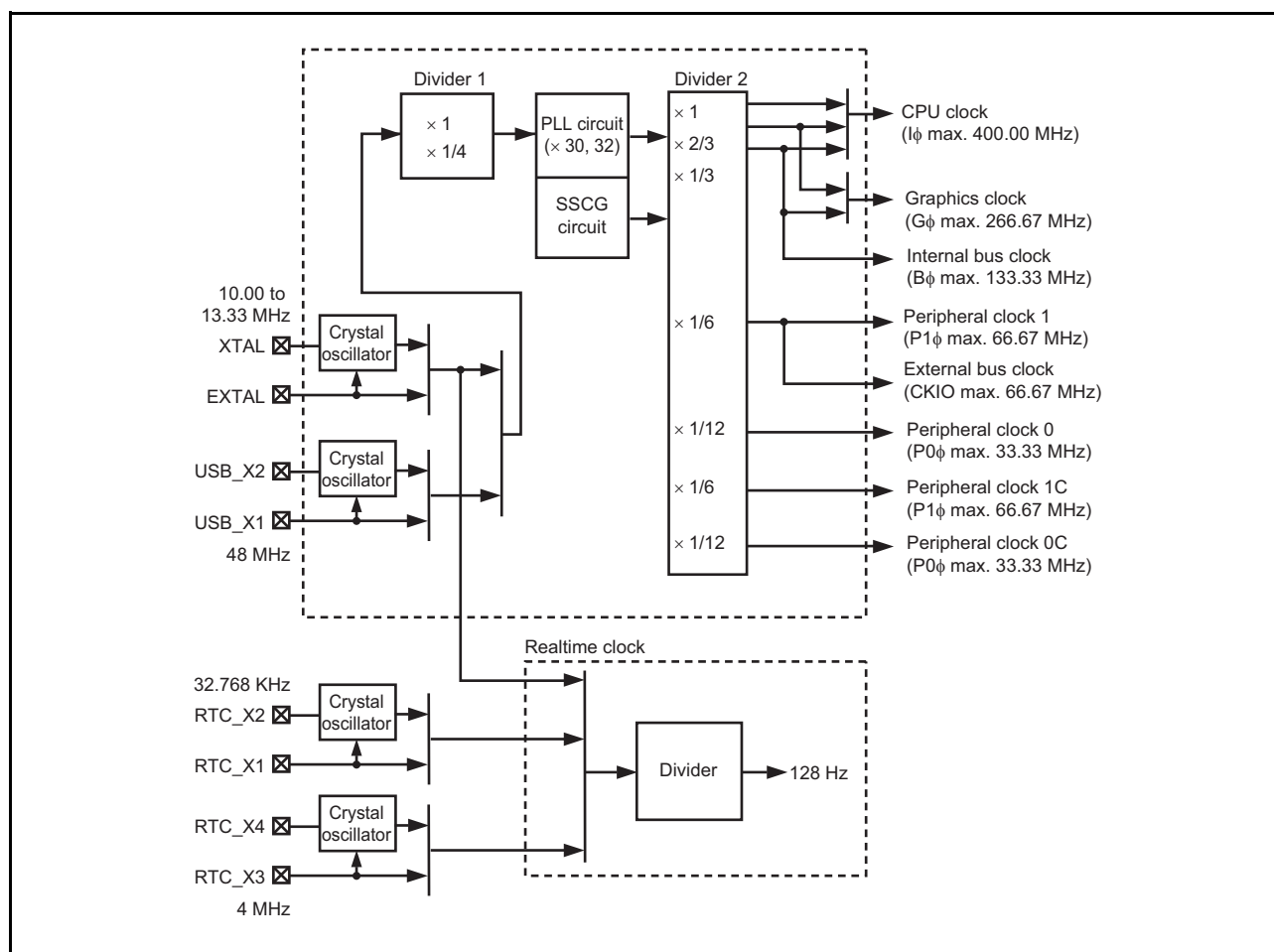


Figure 6.5 Clock Signals for the System and Realtime Clock

6.10.2 Audio and USB Clock Signals

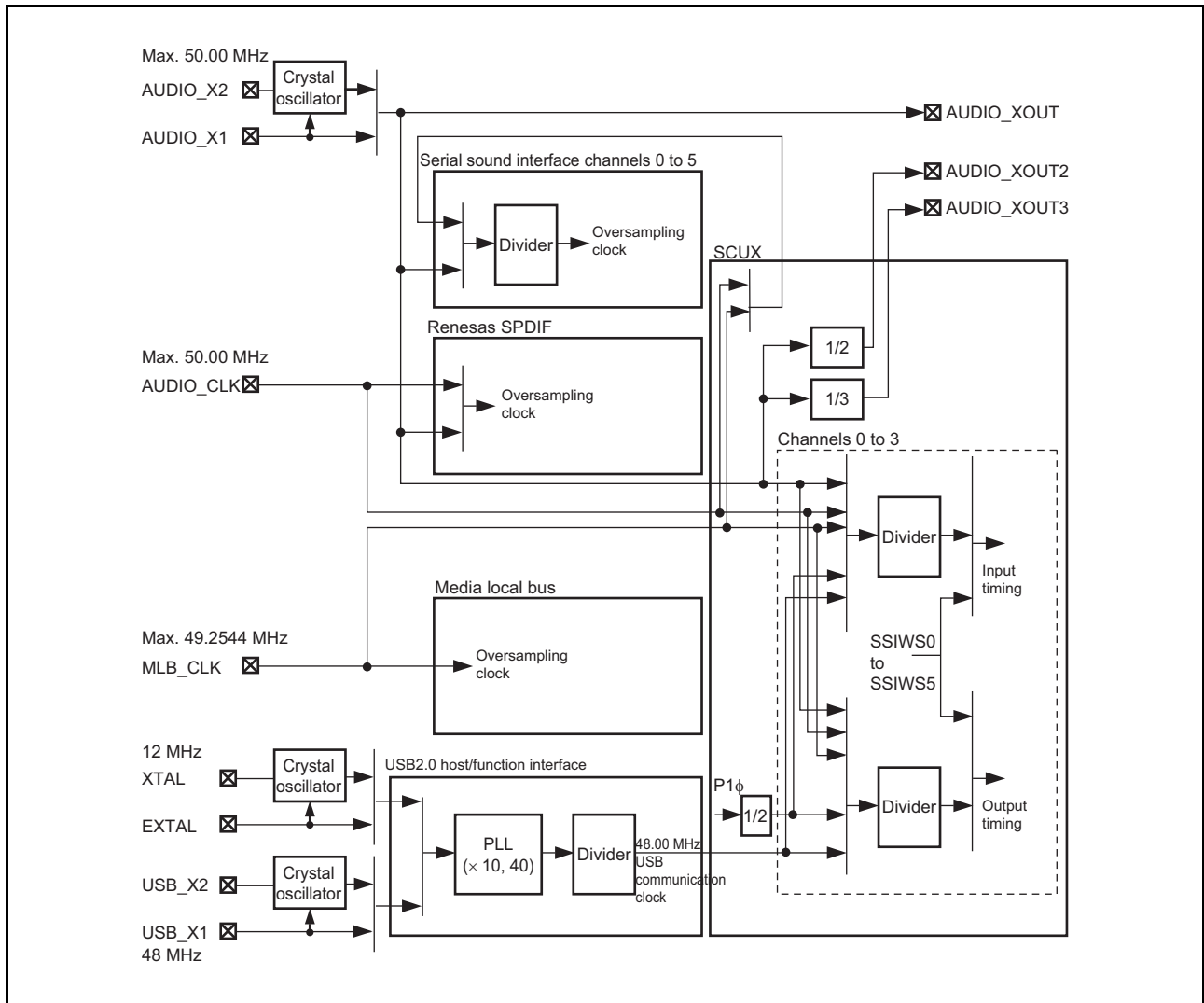


Figure 6.6 Audio and USB Clock Signals

6.10.3 Video Image Clock Signals (Channel 0)

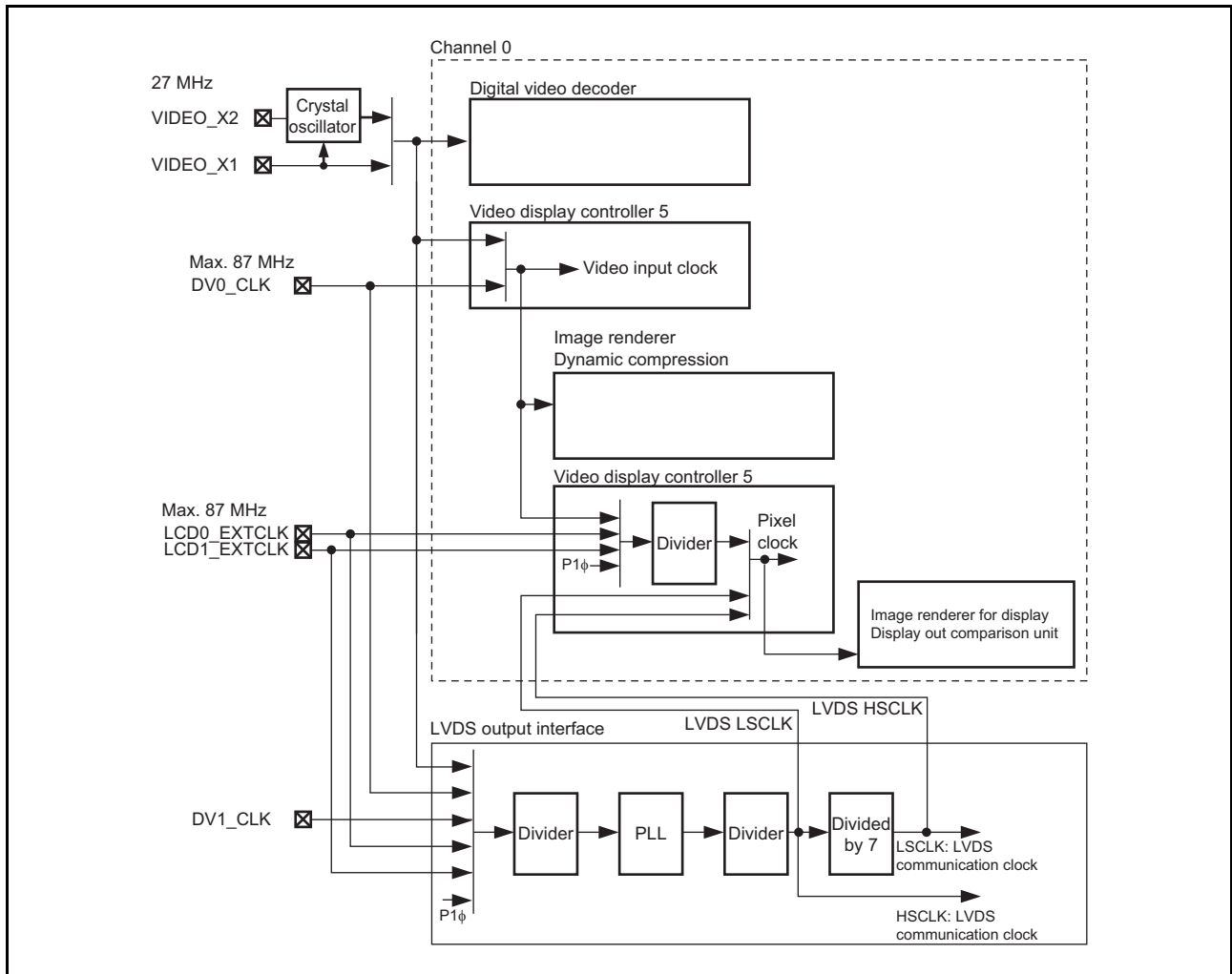


Figure 6.7 Video Image Clock Signals (Channel 0)



### 6.10.4 Video Image Clock Signals (Channel 1)

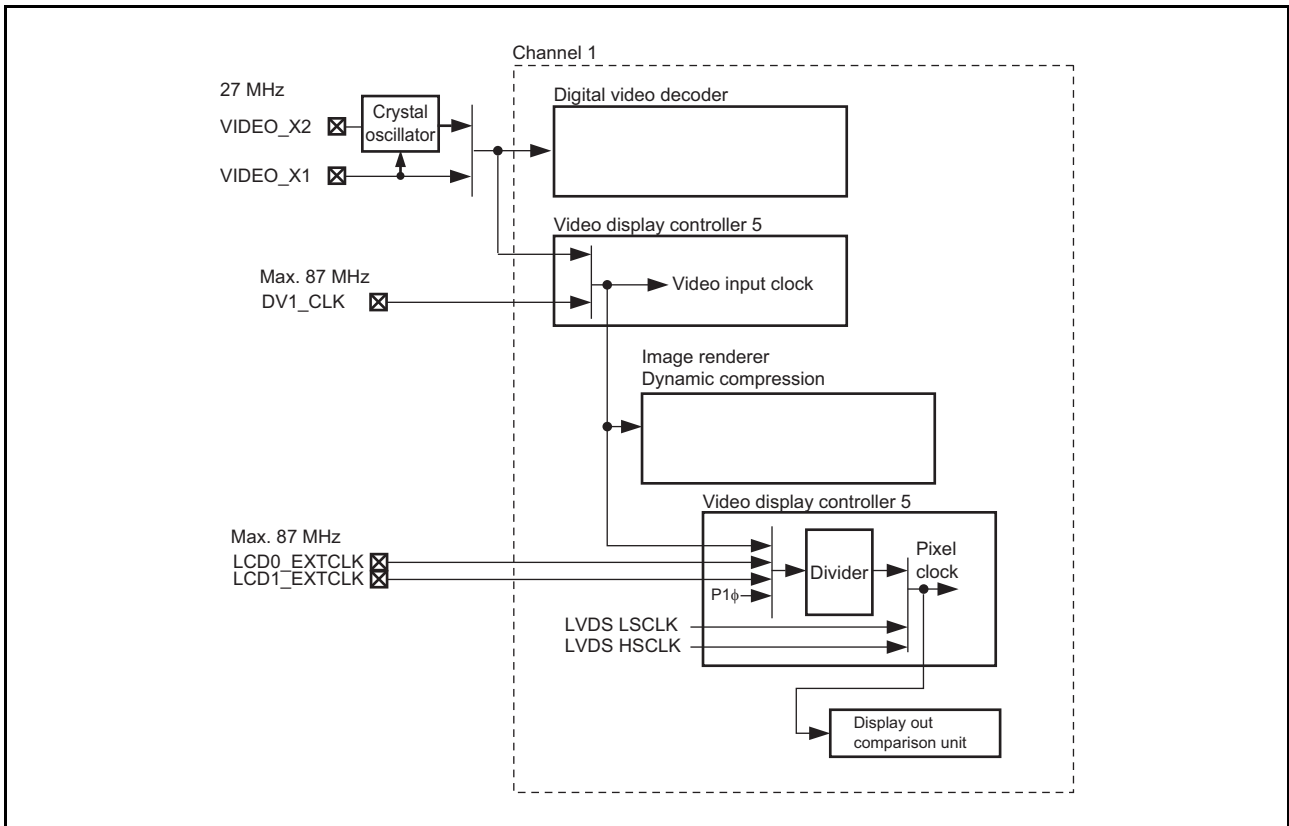


Figure 6.8 Video Image Clock Signals (Channel 1)

### 6.10.5 Other Clock Signals

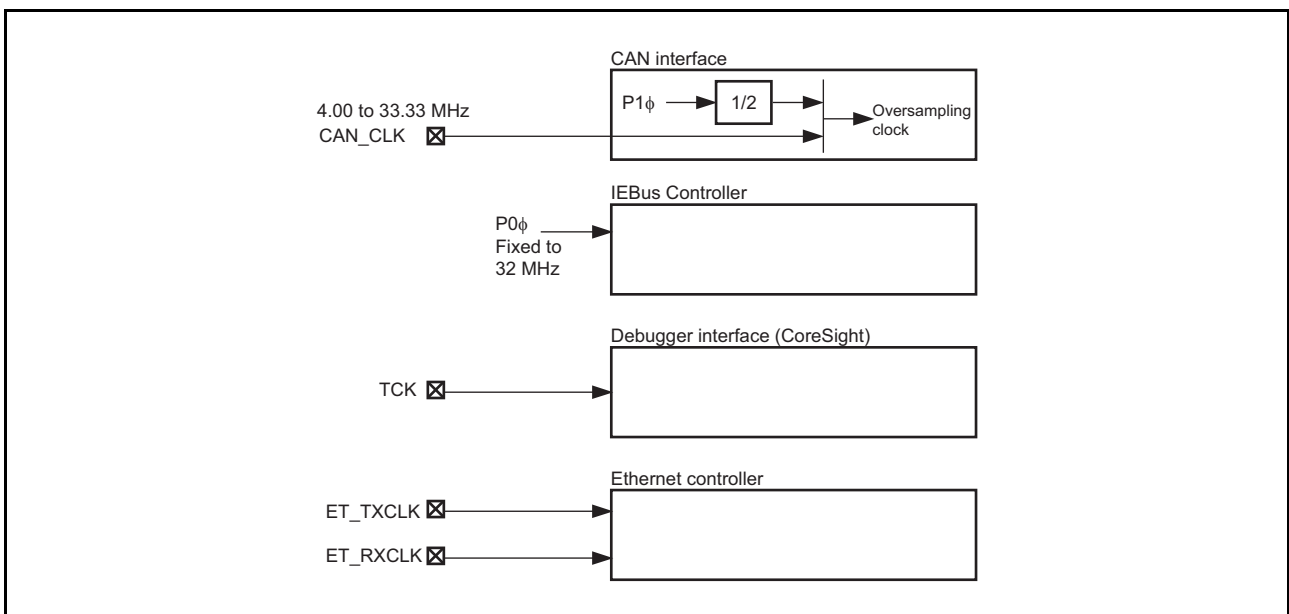


Figure 6.9 Clock Signals for Other Modules

## 6.10.6 Internal Clock Signals (1)

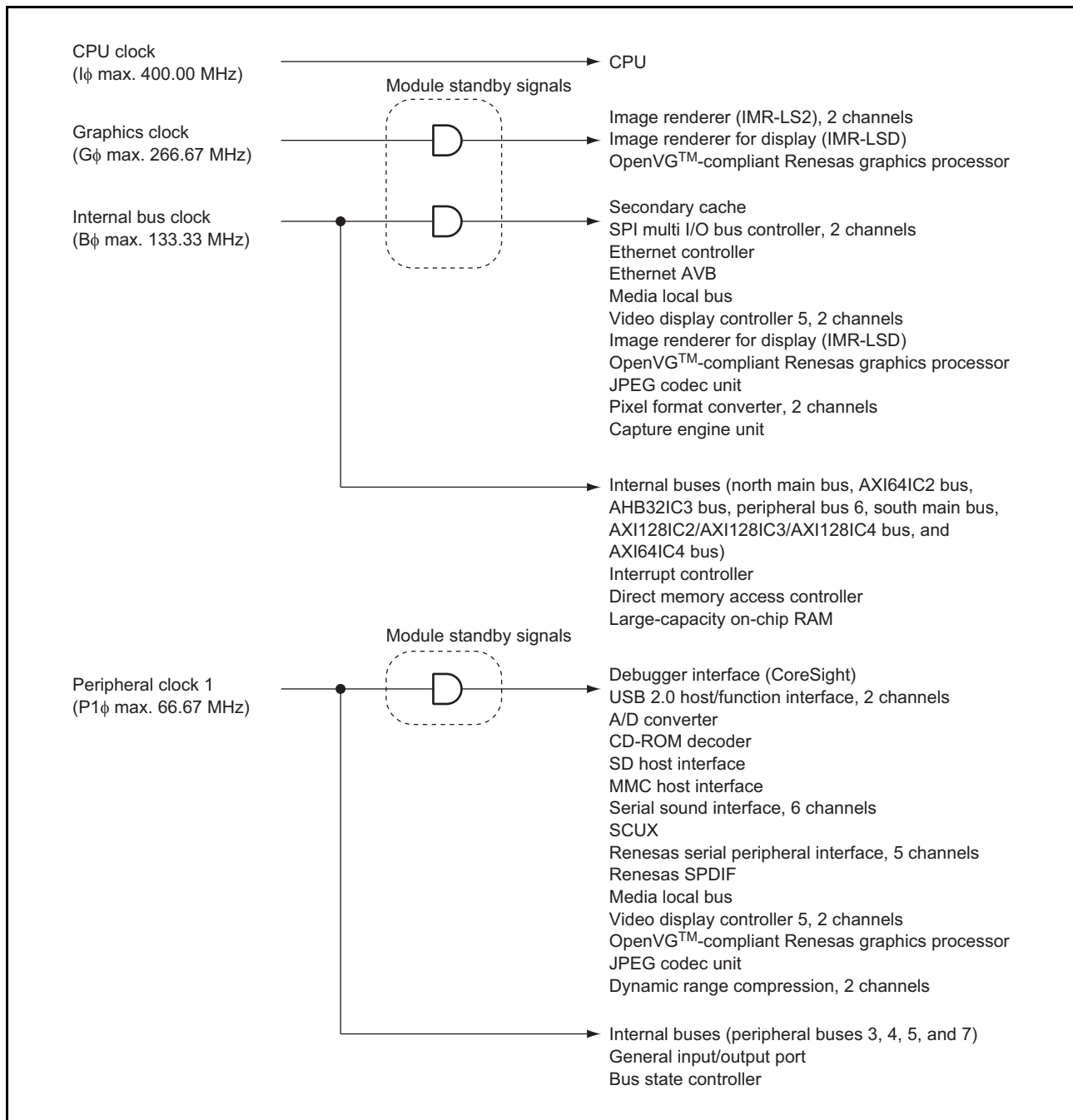


Figure 6.10 Distribution of Internal Clock Signals

## 6.10.7 Internal Clock Signals (2)

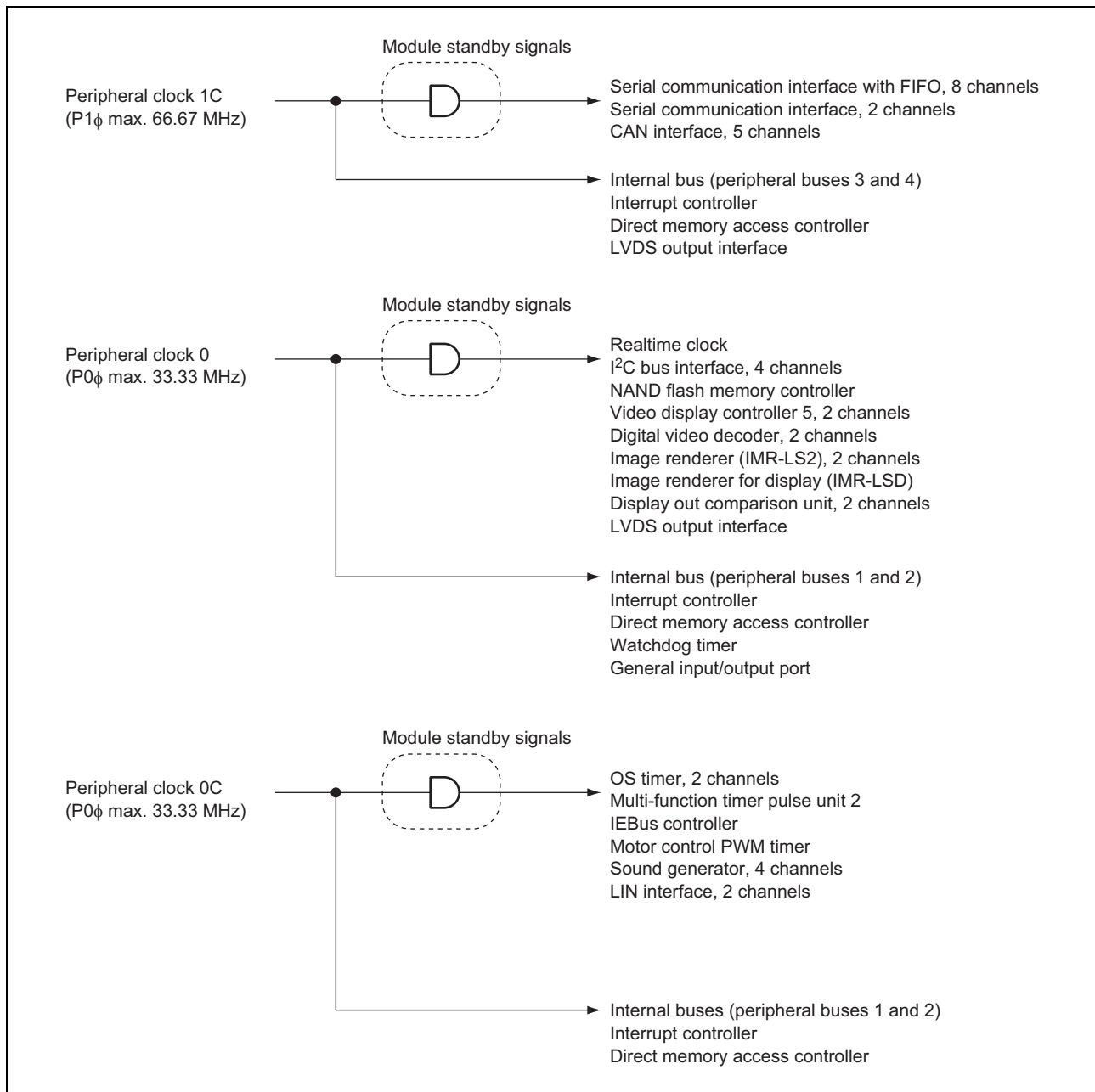


Figure 6.11 Distribution of Internal Clock Signals (2)

## 7. Interrupt Controller

The interrupt controller ascertains the priority of interrupt sources and controls interrupt requests to the CPU. The interrupt controller registers set the order of priority of each interrupt, allowing the user to process interrupt requests according to the user-set priority.

### 7.1 Features

- 32 levels of interrupt priority can be set.  
By setting the interrupt priority registers, the priorities of IRQ interrupts, on-chip peripheral module interrupts, and pin interrupts can be selected from 16 levels for request sources.
- NMI noise canceler function  
An NMI input-level bit indicates the NMI pin state. By reading this bit in the interrupt exception service routine, the pin state can be checked, enabling it to be used as the noise canceler function.

Figure 7.1 shows a block diagram.

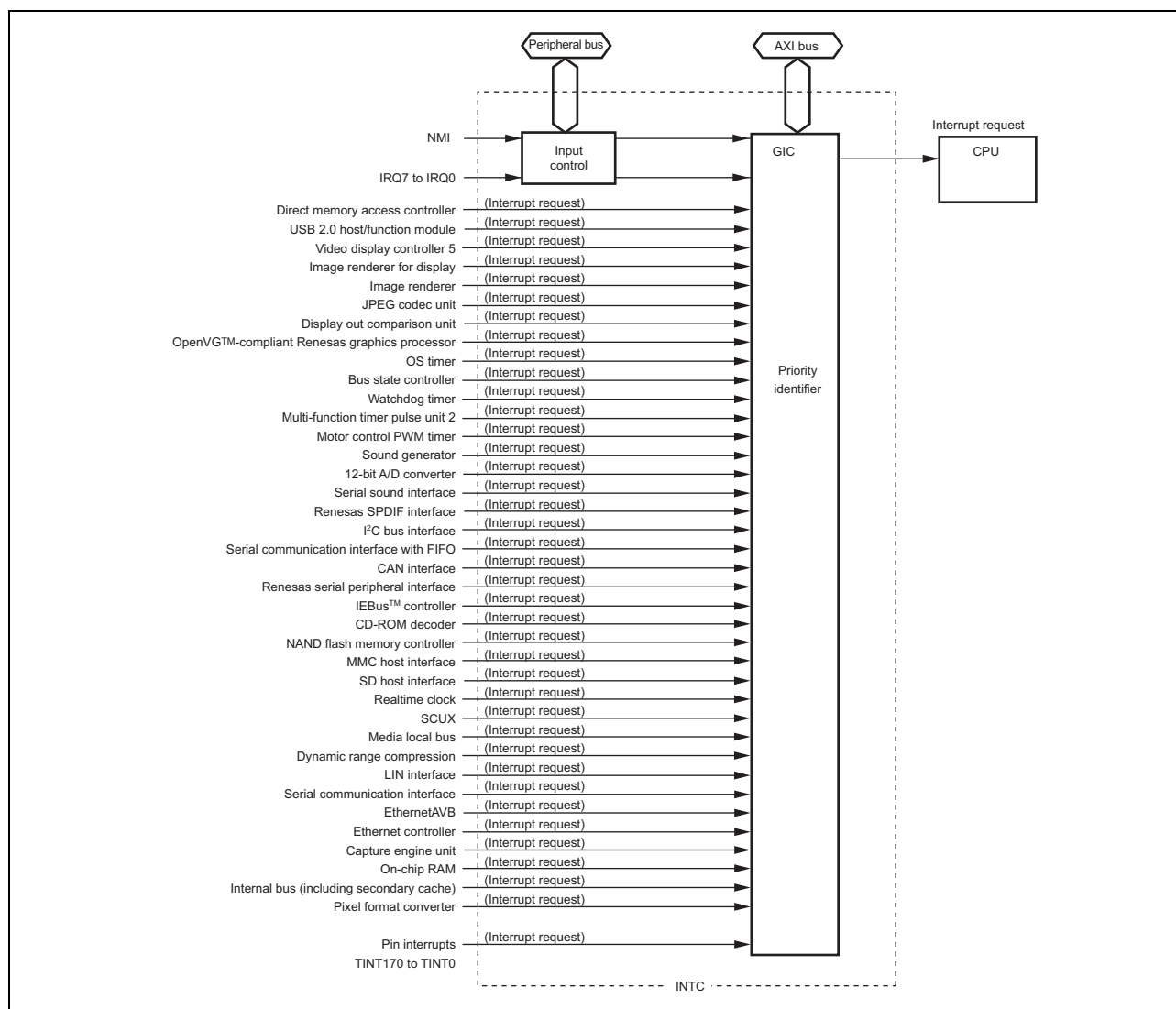


Figure 7.1 Block Diagram

## 7.2 Input/Output Pins

Table 7.1 shows the pin configuration.

Table 7.1 Pin Configuration

Pin Name	Symbol	I/O	Function
Nonmaskable interrupt input pin	NMI	Input	Input of nonmaskable interrupt request signal
Interrupt request input pins	IRQ7 to IRQ0	Input	Input of maskable interrupt request signals
	TINT170 to TINT0	Input	Input of maskable interrupt request signals

### 7.3 Register Descriptions

Table 7.2 shows the register configuration. These registers are used to set the interrupt priorities and control detection of the external interrupt input signal.

For a description of the registers other than interrupt control register 0, interrupt control register 1, and IRQ interrupt request register, see the GIC architecture specification.

**Table 7.2 Register Configuration**

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Interrupt control register 0	ICR0	R/W	*1	H'FCFEF800	16
Interrupt control register 1	ICR1	R/W	H'0000	H'FCFEF802	16
IRQ interrupt request register	IRQRR	R/(W) <sup>2</sup>	H'0000	H'FCFEF804	16
Distributor control register	ICDDCR	RW	H'00000000	H'E8201000	32
Interrupt controller type register	ICDICTR	R	H'0000FC31	H'E8201004	32
Distributor implementer identification register	ICDIIDR	R	H'0000043B	H'E8201008	32
Interrupt security register 0	ICDISR0	R/W	H'00000000	H'E8201080	32
Interrupt security register 1	ICDISR1	R/W	H'00000000	H'E8201084	32
Interrupt security register 2	ICDISR2	R/W	H'00000000	H'E8201088	32
Interrupt security register 3	ICDISR3	R/W	H'00000000	H'E820108C	32
Interrupt security register 4	ICDISR4	R/W	H'00000000	H'E8201090	32
Interrupt security register 5	ICDISR5	R/W	H'00000000	H'E8201094	32
Interrupt security register 6	ICDISR6	R/W	H'00000000	H'E8201098	32
Interrupt security register 7	ICDISR7	R/W	H'00000000	H'E820109C	32
Interrupt security register 8	ICDISR8	R/W	H'00000000	H'E82010A0	32
Interrupt security register 9	ICDISR9	R/W	H'00000000	H'E82010A4	32
Interrupt security register 10	ICDISR10	R/W	H'00000000	H'E82010A8	32
Interrupt security register 11	ICDISR11	R/W	H'00000000	H'E82010AC	32
Interrupt security register 12	ICDISR12	R/W	H'00000000	H'E82010B0	32
Interrupt security register 13	ICDISR13	R/W	H'00000000	H'E82010B4	32
Interrupt security register 14	ICDISR14	R/W	H'00000000	H'E82010B8	32
Interrupt security register 15	ICDISR15	R/W	H'00000000	H'E82010BC	32
Interrupt security register 16	ICDISR16	R/W	H'00000000	H'E82010C0	32
Interrupt security register 17	ICDISR17	R/W	H'00000000	H'E82010C4	32
Interrupt security register 18	ICDISR18	R/W	H'00000000	H'E82010C8	32
Interrupt set-enable register 0	ICDISER0	R/W	H'00000000	H'E8201100	32
Interrupt set-enable register 1	ICDISER1	R/W	H'00000000	H'E8201104	32
Interrupt set-enable register 2	ICDISER2	R/W	H'00000000	H'E8201108	32
Interrupt set-enable register 3	ICDISER3	R/W	H'00000000	H'E820110C	32
Interrupt set-enable register 4	ICDISER4	R/W	H'00000000	H'E8201110	32
Interrupt set-enable register 5	ICDISER5	R/W	H'00000000	H'E8201114	32
Interrupt set-enable register 6	ICDISER6	R/W	H'00000000	H'E8201118	32
Interrupt set-enable register 7	ICDISER7	R/W	H'00000000	H'E820111C	32
Interrupt set-enable register 8	ICDISER8	R/W	H'00000000	H'E8201120	32
Interrupt set-enable register 9	ICDISER9	R/W	H'00000000	H'E8201124	32
Interrupt set-enable register 10	ICDISER10	R/W	H'00000000	H'E8201128	32
Interrupt set-enable register 11	ICDISER11	R/W	H'00000000	H'E820112C	32
Interrupt set-enable register 12	ICDISER12	R/W	H'00000000	H'E8201130	32
Interrupt set-enable register 13	ICDISER13	R/W	H'00000000	H'E8201134	32

Table 7.2 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Interrupt set-enable register 14	ICDISER14	R/W	H'00000000	H'E8201138	32
Interrupt set-enable register 15	ICDISER15	R/W	H'00000000	H'E820113C	32
Interrupt set-enable register 16	ICDISER16	R/W	H'00000000	H'E8201140	32
Interrupt set-enable register 17	ICDISER17	R/W	H'00000000	H'E8201144	32
Interrupt set-enable register 18	ICDISER18	R/W	H'00000000	H'E8201148	32
Interrupt clear-enable register 0	ICDICER0	R/W	H'00000000	H'E8201180	32
Interrupt clear-enable register 1	ICDICER1	R/W	H'00000000	H'E8201184	32
Interrupt clear-enable register 2	ICDICER2	R/W	H'00000000	H'E8201188	32
Interrupt clear-enable register 3	ICDICER3	R/W	H'00000000	H'E820118C	32
Interrupt clear-enable register 4	ICDICER4	R/W	H'00000000	H'E8201190	32
Interrupt clear-enable register 5	ICDICER5	R/W	H'00000000	H'E8201194	32
Interrupt clear-enable register 6	ICDICER6	R/W	H'00000000	H'E8201198	32
Interrupt clear-enable register 7	ICDICER7	R/W	H'00000000	H'E820119C	32
Interrupt clear-enable register 8	ICDICER8	R/W	H'00000000	H'E82011A0	32
Interrupt clear-enable register 9	ICDICER9	R/W	H'00000000	H'E82011A4	32
Interrupt clear-enable register 10	ICDICER10	R/W	H'00000000	H'E82011A8	32
Interrupt clear-enable register 11	ICDICER11	R/W	H'00000000	H'E82011AC	32
Interrupt clear-enable register 12	ICDICER12	R/W	H'00000000	H'E82011B0	32
Interrupt clear-enable register 13	ICDICER13	R/W	H'00000000	H'E82011B4	32
Interrupt clear-enable register 14	ICDICER14	R/W	H'00000000	H'E82011B8	32
Interrupt clear-enable register 15	ICDICER15	R/W	H'00000000	H'E82011BC	32
Interrupt clear-enable register 16	ICDICER16	R/W	H'00000000	H'E82011C0	32
Interrupt clear-enable register 17	ICDICER17	R/W	H'00000000	H'E82011C4	32
Interrupt clear-enable register 18	ICDICER18	R/W	H'00000000	H'E82011C8	32
Interrupt set-pending register 0	ICDISPR0	R/W	H'00000000	H'E8201200	32
Interrupt set-pending register 1	ICDISPR1	R/W	H'00000000	H'E8201204	32
Interrupt set-pending register 2	ICDISPR2	R/W	H'00000000	H'E8201208	32
Interrupt set-pending register 3	ICDISPR3	R/W	H'00000000	H'E820120C	32
Interrupt set-pending register 4	ICDISPR4	R/W	H'00000000	H'E8201210	32
Interrupt set-pending register 5	ICDISPR5	R/W	H'00000000	H'E8201214	32
Interrupt set-pending register 6	ICDISPR6	R/W	H'00000000	H'E8201218	32
Interrupt set-pending register 7	ICDISPR7	R/W	H'00000000	H'E820121C	32
Interrupt set-pending register 8	ICDISPR8	R/W	H'00000000	H'E8201220	32
Interrupt set-pending register 9	ICDISPR9	R/W	H'00000000	H'E8201224	32
Interrupt set-pending register 10	ICDISPR10	R/W	H'00000000	H'E8201228	32
Interrupt set-pending register 11	ICDISPR11	R/W	H'00000000	H'E820122C	32
Interrupt set-pending register 12	ICDISPR12	R/W	H'00000000	H'E8201230	32
Interrupt set-pending register 13	ICDISPR13	R/W	H'00000000	H'E8201234	32
Interrupt set-pending register 14	ICDISPR14	R/W	H'00000000	H'E8201238	32
Interrupt set-pending register 15	ICDISPR15	R/W	H'00000000	H'E820123C	32
Interrupt set-pending register 16	ICDISPR16	R/W	H'00000000	H'E8201240	32
Interrupt set-pending register 17	ICDISPR17	R/W	H'00000000	H'E8201244	32
Interrupt set-pending register 18	ICDISPR18	R/W	H'00000000	H'E8201248	32
Interrupt clear-pending register 0	ICDICPR0	R/W	H'00000000	H'E8201280	32
Interrupt clear-pending register 1	ICDICPR1	R/W	H'00000000	H'E8201284	32

Table 7.2 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Interrupt clear-pending register 2	ICDICPR2	R/W	H'00000000	H'E8201288	32
Interrupt clear-pending register 3	ICDICPR3	R/W	H'00000000	H'E820128C	32
Interrupt clear-pending register 4	ICDICPR4	R/W	H'00000000	H'E8201290	32
Interrupt clear-pending register 5	ICDICPR5	R/W	H'00000000	H'E8201294	32
Interrupt clear-pending register 6	ICDICPR6	R/W	H'00000000	H'E8201298	32
Interrupt clear-pending register 7	ICDICPR7	R/W	H'00000000	H'E820129C	32
Interrupt clear-pending register 8	ICDICPR8	R/W	H'00000000	H'E82012A0	32
Interrupt clear-pending register 9	ICDICPR9	R/W	H'00000000	H'E82012A4	32
Interrupt clear-pending register 10	ICDICPR10	R/W	H'00000000	H'E82012A8	32
Interrupt clear-pending register 11	ICDICPR11	R/W	H'00000000	H'E82012AC	32
Interrupt clear-pending register 12	ICDICPR12	R/W	H'00000000	H'E82012B0	32
Interrupt clear-pending register 13	ICDICPR13	R/W	H'00000000	H'E82012B4	32
Interrupt clear-pending register 14	ICDICPR14	R/W	H'00000000	H'E82012B8	32
Interrupt clear-pending register 15	ICDICPR15	R/W	H'00000000	H'E82012BC	32
Interrupt clear-pending register 16	ICDICPR16	R/W	H'00000000	H'E82012C0	32
Interrupt clear-pending register 17	ICDICPR17	R/W	H'00000000	H'E82012C4	32
Interrupt clear-pending register 18	ICDICPR18	R/W	H'00000000	H'E82012C8	32
Active bit register 0	ICDABR0	R/W	H'00000000	H'E8201300	32
Active bit register 1	ICDABR1	R/W	H'00000000	H'E8201304	32
Active bit register 2	ICDABR2	R/W	H'00000000	H'E8201308	32
Active bit register 3	ICDABR3	R/W	H'00000000	H'E820130C	32
Active bit register 4	ICDABR4	R/W	H'00000000	H'E8201310	32
Active bit register 5	ICDABR5	R/W	H'00000000	H'E8201314	32
Active bit register 6	ICDABR6	R/W	H'00000000	H'E8201318	32
Active bit register 7	ICDABR7	R/W	H'00000000	H'E820131C	32
Active bit register 8	ICDABR8	R/W	H'00000000	H'E8201320	32
Active bit register 9	ICDABR9	R/W	H'00000000	H'E8201324	32
Active bit register 10	ICDABR10	R/W	H'00000000	H'E8201328	32
Active bit register 11	ICDABR11	R/W	H'00000000	H'E820132C	32
Active bit register 12	ICDABR12	R/W	H'00000000	H'E8201330	32
Active bit register 13	ICDABR13	R/W	H'00000000	H'E8201334	32
Active bit register 14	ICDABR14	R/W	H'00000000	H'E8201338	32
Active bit register 15	ICDABR15	R/W	H'00000000	H'E820133C	32
Active bit register 16	ICDABR16	R/W	H'00000000	H'E8201340	32
Active bit register 17	ICDABR17	R/W	H'00000000	H'E8201344	32
Active bit register 18	ICDABR18	R/W	H'00000000	H'E8201348	32
Interrupt priority register 0	ICDIPR0	R/W	H'00000000	H'E8201400	32
Interrupt priority register 1	ICDIPR1	R/W	H'00000000	H'E8201404	32
Interrupt priority register 2	ICDIPR2	R/W	H'00000000	H'E8201408	32
Interrupt priority register 3	ICDIPR3	R/W	H'00000000	H'E820140C	32
Interrupt priority register 4	ICDIPR4	R/W	H'00000000	H'E8201410	32
Interrupt priority register 5	ICDIPR5	R/W	H'00000000	H'E8201414	32
Interrupt priority register 6	ICDIPR6	R/W	H'00000000	H'E8201418	32
Interrupt priority register 7	ICDIPR7	R/W	H'00000000	H'E820141C	32
Interrupt priority register 8	ICDIPR8	R/W	H'00000000	H'E8201420	32



**Table 7.2 Register Configuration**

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Interrupt priority register 9	ICDIPR9	R/W	H'00000000	H'E8201424	32
Interrupt priority register 10	ICDIPR10	R/W	H'00000000	H'E8201428	32
Interrupt priority register 11	ICDIPR11	R/W	H'00000000	H'E820142C	32
Interrupt priority register 12	ICDIPR12	R/W	H'00000000	H'E8201430	32
Interrupt priority register 13	ICDIPR13	R/W	H'00000000	H'E8201434	32
Interrupt priority register 14	ICDIPR14	R/W	H'00000000	H'E8201438	32
Interrupt priority register 15	ICDIPR15	R/W	H'00000000	H'E820143C	32
Interrupt priority register 16	ICDIPR16	R/W	H'00000000	H'E8201440	32
Interrupt priority register 17	ICDIPR17	R/W	H'00000000	H'E8201444	32
Interrupt priority register 18	ICDIPR18	R/W	H'00000000	H'E8201448	32
Interrupt priority register 19	ICDIPR19	R/W	H'00000000	H'E820144C	32
Interrupt priority register 20	ICDIPR20	R/W	H'00000000	H'E8201450	32
Interrupt priority register 21	ICDIPR21	R/W	H'00000000	H'E8201454	32
Interrupt priority register 22	ICDIPR22	R/W	H'00000000	H'E8201458	32
Interrupt priority register 23	ICDIPR23	R/W	H'00000000	H'E820145C	32
Interrupt priority register 24	ICDIPR24	R/W	H'00000000	H'E8201460	32
Interrupt priority register 25	ICDIPR25	R/W	H'00000000	H'E8201464	32
Interrupt priority register 26	ICDIPR26	R/W	H'00000000	H'E8201468	32
Interrupt priority register 27	ICDIPR27	R/W	H'00000000	H'E820146C	32
Interrupt priority register 28	ICDIPR28	R/W	H'00000000	H'E8201470	32
Interrupt priority register 29	ICDIPR29	R/W	H'00000000	H'E8201474	32
Interrupt priority register 30	ICDIPR30	R/W	H'00000000	H'E8201478	32
Interrupt priority register 31	ICDIPR31	R/W	H'00000000	H'E820147C	32
Interrupt priority register 32	ICDIPR32	R/W	H'00000000	H'E8201480	32
Interrupt priority register 33	ICDIPR33	R/W	H'00000000	H'E8201484	32
Interrupt priority register 34	ICDIPR34	R/W	H'00000000	H'E8201488	32
Interrupt priority register 35	ICDIPR35	R/W	H'00000000	H'E820148C	32
Interrupt priority register 36	ICDIPR36	R/W	H'00000000	H'E8201490	32
Interrupt priority register 37	ICDIPR37	R/W	H'00000000	H'E8201494	32
Interrupt priority register 38	ICDIPR38	R/W	H'00000000	H'E8201498	32
Interrupt priority register 39	ICDIPR39	R/W	H'00000000	H'E820149C	32
Interrupt priority register 40	ICDIPR40	R/W	H'00000000	H'E82014A0	32
Interrupt priority register 41	ICDIPR41	R/W	H'00000000	H'E82014A4	32
Interrupt priority register 42	ICDIPR42	R/W	H'00000000	H'E82014A8	32
Interrupt priority register 43	ICDIPR43	R/W	H'00000000	H'E82014AC	32
Interrupt priority register 44	ICDIPR44	R/W	H'00000000	H'E82014B0	32
Interrupt priority register 45	ICDIPR45	R/W	H'00000000	H'E82014B4	32
Interrupt priority register 46	ICDIPR46	R/W	H'00000000	H'E82014B8	32
Interrupt priority register 47	ICDIPR47	R/W	H'00000000	H'E82014BC	32
Interrupt priority register 48	ICDIPR48	R/W	H'00000000	H'E82014C0	32
Interrupt priority register 49	ICDIPR49	R/W	H'00000000	H'E82014C4	32
Interrupt priority register 50	ICDIPR50	R/W	H'00000000	H'E82014C8	32
Interrupt priority register 51	ICDIPR51	R/W	H'00000000	H'E82014CC	32
Interrupt priority register 52	ICDIPR52	R/W	H'00000000	H'E82014D0	32
Interrupt priority register 53	ICDIPR53	R/W	H'00000000	H'E82014D4	32

Table 7.2 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Interrupt priority register 54	ICDIPR54	R/W	H'00000000	H'E82014D8	32
Interrupt priority register 55	ICDIPR55	R/W	H'00000000	H'E82014DC	32
Interrupt priority register 56	ICDIPR56	R/W	H'00000000	H'E82014E0	32
Interrupt priority register 57	ICDIPR57	R/W	H'00000000	H'E82014E4	32
Interrupt priority register 58	ICDIPR58	R/W	H'00000000	H'E82014E8	32
Interrupt priority register 59	ICDIPR59	R/W	H'00000000	H'E82014EC	32
Interrupt priority register 60	ICDIPR60	R/W	H'00000000	H'E82014F0	32
Interrupt priority register 61	ICDIPR61	R/W	H'00000000	H'E82014F4	32
Interrupt priority register 62	ICDIPR62	R/W	H'00000000	H'E82014F8	32
Interrupt priority register 63	ICDIPR63	R/W	H'00000000	H'E82014FC	32
Interrupt priority register 64	ICDIPR64	R/W	H'00000000	H'E8201500	32
Interrupt priority register 65	ICDIPR65	R/W	H'00000000	H'E8201504	32
Interrupt priority register 66	ICDIPR66	R/W	H'00000000	H'E8201508	32
Interrupt priority register 67	ICDIPR67	R/W	H'00000000	H'E820150C	32
Interrupt priority register 68	ICDIPR68	R/W	H'00000000	H'E8201510	32
Interrupt priority register 69	ICDIPR69	R/W	H'00000000	H'E8201514	32
Interrupt priority register 70	ICDIPR70	R/W	H'00000000	H'E8201518	32
Interrupt priority register 71	ICDIPR71	R/W	H'00000000	H'E820151C	32
Interrupt priority register 72	ICDIPR72	R/W	H'00000000	H'E8201520	32
Interrupt priority register 73	ICDIPR73	R/W	H'00000000	H'E8201524	32
Interrupt priority register 74	ICDIPR74	R/W	H'00000000	H'E8201528	32
Interrupt priority register 75	ICDIPR75	R/W	H'00000000	H'E820152C	32
Interrupt priority register 76	ICDIPR76	R/W	H'00000000	H'E8201530	32
Interrupt priority register 77	ICDIPR77	R/W	H'00000000	H'E8201534	32
Interrupt priority register 78	ICDIPR78	R/W	H'00000000	H'E8201538	32
Interrupt priority register 79	ICDIPR79	R/W	H'00000000	H'E820153C	32
Interrupt priority register 80	ICDIPR80	R/W	H'00000000	H'E8201540	32
Interrupt priority register 81	ICDIPR81	R/W	H'00000000	H'E8201544	32
Interrupt priority register 82	ICDIPR82	R/W	H'00000000	H'E8201548	32
Interrupt priority register 83	ICDIPR83	R/W	H'00000000	H'E820154C	32
Interrupt priority register 84	ICDIPR84	R/W	H'00000000	H'E8201550	32
Interrupt priority register 85	ICDIPR85	R/W	H'00000000	H'E8201554	32
Interrupt priority register 86	ICDIPR86	R/W	H'00000000	H'E8201558	32
Interrupt priority register 87	ICDIPR87	R/W	H'00000000	H'E820155C	32
Interrupt priority register 88	ICDIPR88	R/W	H'00000000	H'E8201560	32
Interrupt priority register 89	ICDIPR89	R/W	H'00000000	H'E8201564	32
Interrupt priority register 90	ICDIPR90	R/W	H'00000000	H'E8201568	32
Interrupt priority register 91	ICDIPR91	R/W	H'00000000	H'E820156C	32
Interrupt priority register 92	ICDIPR92	R/W	H'00000000	H'E8201570	32
Interrupt priority register 93	ICDIPR93	R/W	H'00000000	H'E8201574	32
Interrupt priority register 94	ICDIPR94	R/W	H'00000000	H'E8201578	32
Interrupt priority register 95	ICDIPR95	R/W	H'00000000	H'E820157C	32
Interrupt priority register 96	ICDIPR96	R/W	H'00000000	H'E8201580	32
Interrupt priority register 97	ICDIPR97	R/W	H'00000000	H'E8201584	32
Interrupt priority register 98	ICDIPR98	R/W	H'00000000	H'E8201588	32

**Table 7.2 Register Configuration**

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Interrupt priority register 99	ICDIPR99	R/W	H'00000000	H'E820158C	32
Interrupt priority register 100	ICDIPR100	R/W	H'00000000	H'E8201590	32
Interrupt priority register 101	ICDIPR101	R/W	H'00000000	H'E8201594	32
Interrupt priority register 102	ICDIPR102	R/W	H'00000000	H'E8201598	32
Interrupt priority register 103	ICDIPR103	R/W	H'00000000	H'E820159C	32
Interrupt priority register 104	ICDIPR104	R/W	H'00000000	H'E82015A0	32
Interrupt priority register 105	ICDIPR105	R/W	H'00000000	H'E82015A4	32
Interrupt priority register 106	ICDIPR106	R/W	H'00000000	H'E82015A8	32
Interrupt priority register 107	ICDIPR107	R/W	H'00000000	H'E82015AC	32
Interrupt priority register 108	ICDIPR108	R/W	H'00000000	H'E82015B0	32
Interrupt priority register 109	ICDIPR109	R/W	H'00000000	H'E82015B4	32
Interrupt priority register 110	ICDIPR110	R/W	H'00000000	H'E82015B8	32
Interrupt priority register 111	ICDIPR111	R/W	H'00000000	H'E82015BC	32
Interrupt priority register 112	ICDIPR112	R/W	H'00000000	H'E82015C0	32
Interrupt priority register 113	ICDIPR113	R/W	H'00000000	H'E82015C4	32
Interrupt priority register 114	ICDIPR114	R/W	H'00000000	H'E82015C8	32
Interrupt priority register 115	ICDIPR115	R/W	H'00000000	H'E82015CC	32
Interrupt priority register 116	ICDIPR116	R/W	H'00000000	H'E82015D0	32
Interrupt priority register 117	ICDIPR117	R/W	H'00000000	H'E82015D4	32
Interrupt priority register 118	ICDIPR118	R/W	H'00000000	H'E82015D8	32
Interrupt priority register 119	ICDIPR119	R/W	H'00000000	H'E82015DC	32
Interrupt priority register 120	ICDIPR120	R/W	H'00000000	H'E82015E0	32
Interrupt priority register 121	ICDIPR121	R/W	H'00000000	H'E82015E4	32
Interrupt priority register 122	ICDIPR122	R/W	H'00000000	H'E82015E8	32
Interrupt priority register 123	ICDIPR123	R/W	H'00000000	H'E82015EC	32
Interrupt priority register 124	ICDIPR124	R/W	H'00000000	H'E82015F0	32
Interrupt priority register 125	ICDIPR125	R/W	H'00000000	H'E82015F4	32
Interrupt priority register 126	ICDIPR126	R/W	H'00000000	H'E82015F8	32
Interrupt priority register 127	ICDIPR127	R/W	H'00000000	H'E82015FC	32
Interrupt priority register 128	ICDIPR128	R/W	H'00000000	H'E8201600	32
Interrupt priority register 129	ICDIPR129	R/W	H'00000000	H'E8201604	32
Interrupt priority register 130	ICDIPR130	R/W	H'00000000	H'E8201608	32
Interrupt priority register 131	ICDIPR131	R/W	H'00000000	H'E820160C	32
Interrupt priority register 132	ICDIPR132	R/W	H'00000000	H'E8201610	32
Interrupt priority register 133	ICDIPR133	R/W	H'00000000	H'E8201614	32
Interrupt priority register 134	ICDIPR134	R/W	H'00000000	H'E8201618	32
Interrupt priority register 135	ICDIPR135	R/W	H'00000000	H'E820161C	32
Interrupt priority register 136	ICDIPR136	R/W	H'00000000	H'E8201620	32
Interrupt priority register 137	ICDIPR137	R/W	H'00000000	H'E8201624	32
Interrupt priority register 138	ICDIPR138	R/W	H'00000000	H'E8201628	32
Interrupt priority register 139	ICDIPR139	R/W	H'00000000	H'E820162C	32
Interrupt priority register 140	ICDIPR140	R/W	H'00000000	H'E8201630	32
Interrupt priority register 141	ICDIPR141	R/W	H'00000000	H'E8201634	32
Interrupt priority register 142	ICDIPR142	R/W	H'00000000	H'E8201638	32
Interrupt priority register 143	ICDIPR143	R/W	H'00000000	H'E820163C	32

Table 7.2 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Interrupt priority register 144	ICDIPR144	R/W	H'00000000	H'E8201640	32
Interrupt priority register 145	ICDIPR145	R/W	H'00000000	H'E8201644	32
Interrupt priority register 146	ICDIPR146	R/W	H'00000000	H'E8201648	32
Interrupt processor target register 0	ICDIPTR0	R	H'00000000	H'E8201800	32
Interrupt processor target register 1	ICDIPTR1	R	H'00000000	H'E8201804	32
Interrupt processor target register 2	ICDIPTR2	R	H'00000000	H'E8201808	32
Interrupt processor target register 3	ICDIPTR3	R	H'00000000	H'E820180C	32
Interrupt processor target register 4	ICDIPTR4	R	H'00000000	H'E8201810	32
Interrupt processor target register 5	ICDIPTR5	R	H'00000000	H'E8201814	32
Interrupt processor target register 6	ICDIPTR6	R	H'00000000	H'E8201818	32
Interrupt processor target register 7	ICDIPTR7	R	H'00000000	H'E820181C	32
Interrupt processor target register 8	ICDIPTR8	R/W	H'00000000	H'E8201820	32
Interrupt processor target register 9	ICDIPTR9	R/W	H'00000000	H'E8201824	32
Interrupt processor target register 10	ICDIPTR10	R/W	H'00000000	H'E8201828	32
Interrupt processor target register 11	ICDIPTR11	R/W	H'00000000	H'E820182C	32
Interrupt processor target register 12	ICDIPTR12	R/W	H'00000000	H'E8201830	32
Interrupt processor target register 13	ICDIPTR13	R/W	H'00000000	H'E8201834	32
Interrupt processor target register 14	ICDIPTR14	R/W	H'00000000	H'E8201838	32
Interrupt processor target register 15	ICDIPTR15	R/W	H'00000000	H'E820183C	32
Interrupt processor target register 16	ICDIPTR16	R/W	H'00000000	H'E8201840	32
Interrupt processor target register 17	ICDIPTR17	R/W	H'00000000	H'E8201844	32
Interrupt processor target register 18	ICDIPTR18	R/W	H'00000000	H'E8201848	32
Interrupt processor target register 19	ICDIPTR19	R/W	H'00000000	H'E820184C	32
Interrupt processor target register 20	ICDIPTR20	R/W	H'00000000	H'E8201850	32
Interrupt processor target register 21	ICDIPTR21	R/W	H'00000000	H'E8201854	32
Interrupt processor target register 22	ICDIPTR22	R/W	H'00000000	H'E8201858	32
Interrupt processor target register 23	ICDIPTR23	R/W	H'00000000	H'E820185C	32
Interrupt processor target register 24	ICDIPTR24	R/W	H'00000000	H'E8201860	32
Interrupt processor target register 25	ICDIPTR25	R/W	H'00000000	H'E8201864	32
Interrupt processor target register 26	ICDIPTR26	R/W	H'00000000	H'E8201868	32
Interrupt processor target register 27	ICDIPTR27	R/W	H'00000000	H'E820186C	32
Interrupt processor target register 28	ICDIPTR28	R/W	H'00000000	H'E8201870	32
Interrupt processor target register 29	ICDIPTR29	R/W	H'00000000	H'E8201874	32
Interrupt processor target register 30	ICDIPTR30	R/W	H'00000000	H'E8201878	32
Interrupt processor target register 31	ICDIPTR31	R/W	H'00000000	H'E820187C	32
Interrupt processor target register 32	ICDIPTR32	R/W	H'00000000	H'E8201880	32
Interrupt processor target register 33	ICDIPTR33	R/W	H'00000000	H'E8201884	32
Interrupt processor target register 34	ICDIPTR34	R/W	H'00000000	H'E8201888	32
Interrupt processor target register 35	ICDIPTR35	R/W	H'00000000	H'E820188C	32
Interrupt processor target register 36	ICDIPTR36	R/W	H'00000000	H'E8201890	32
Interrupt processor target register 37	ICDIPTR37	R/W	H'00000000	H'E8201894	32
Interrupt processor target register 38	ICDIPTR38	R/W	H'00000000	H'E8201898	32
Interrupt processor target register 39	ICDIPTR39	R/W	H'00000000	H'E820189C	32
Interrupt processor target register 40	ICDIPTR40	R/W	H'00000000	H'E82018A0	32
Interrupt processor target register 41	ICDIPTR41	R/W	H'00000000	H'E82018A4	32

Table 7.2 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Interrupt processor target register 42	ICDIPTR42	R/W	H'00000000	H'E82018A8	32
Interrupt processor target register 43	ICDIPTR43	R/W	H'00000000	H'E82018AC	32
Interrupt processor target register 44	ICDIPTR44	R/W	H'00000000	H'E82018B0	32
Interrupt processor target register 45	ICDIPTR45	R/W	H'00000000	H'E82018B4	32
Interrupt processor target register 46	ICDIPTR46	R/W	H'00000000	H'E82018B8	32
Interrupt processor target register 47	ICDIPTR47	R/W	H'00000000	H'E82018BC	32
Interrupt processor target register 48	ICDIPTR48	R/W	H'00000000	H'E82018C0	32
Interrupt processor target register 49	ICDIPTR49	R/W	H'00000000	H'E82018C4	32
Interrupt processor target register 50	ICDIPTR50	R/W	H'00000000	H'E82018C8	32
Interrupt processor target register 51	ICDIPTR51	R/W	H'00000000	H'E82018CC	32
Interrupt processor target register 52	ICDIPTR52	R/W	H'00000000	H'E82018D0	32
Interrupt processor target register 53	ICDIPTR53	R/W	H'00000000	H'E82018D4	32
Interrupt processor target register 54	ICDIPTR54	R/W	H'00000000	H'E82018D8	32
Interrupt processor target register 55	ICDIPTR55	R/W	H'00000000	H'E82018DC	32
Interrupt processor target register 56	ICDIPTR56	R/W	H'00000000	H'E82018E0	32
Interrupt processor target register 57	ICDIPTR57	R/W	H'00000000	H'E82018E4	32
Interrupt processor target register 58	ICDIPTR58	R/W	H'00000000	H'E82018E8	32
Interrupt processor target register 59	ICDIPTR59	R/W	H'00000000	H'E82018EC	32
Interrupt processor target register 60	ICDIPTR60	R/W	H'00000000	H'E82018F0	32
Interrupt processor target register 61	ICDIPTR61	R/W	H'00000000	H'E82018F4	32
Interrupt processor target register 62	ICDIPTR62	R/W	H'00000000	H'E82018F8	32
Interrupt processor target register 63	ICDIPTR63	R/W	H'00000000	H'E82018FC	32
Interrupt processor target register 64	ICDIPTR64	R/W	H'00000000	H'E8201900	32
Interrupt processor target register 65	ICDIPTR65	R/W	H'00000000	H'E8201904	32
Interrupt processor target register 66	ICDIPTR66	R/W	H'00000000	H'E8201908	32
Interrupt processor target register 67	ICDIPTR67	R/W	H'00000000	H'E820190C	32
Interrupt processor target register 68	ICDIPTR68	R/W	H'00000000	H'E8201910	32
Interrupt processor target register 69	ICDIPTR69	R/W	H'00000000	H'E8201914	32
Interrupt processor target register 70	ICDIPTR70	R/W	H'00000000	H'E8201918	32
Interrupt processor target register 71	ICDIPTR71	R/W	H'00000000	H'E820191C	32
Interrupt processor target register 72	ICDIPTR72	R/W	H'00000000	H'E8201920	32
Interrupt processor target register 73	ICDIPTR73	R/W	H'00000000	H'E8201924	32
Interrupt processor target register 74	ICDIPTR74	R/W	H'00000000	H'E8201928	32
Interrupt processor target register 75	ICDIPTR75	R/W	H'00000000	H'E820192C	32
Interrupt processor target register 76	ICDIPTR76	R/W	H'00000000	H'E8201930	32
Interrupt processor target register 77	ICDIPTR77	R/W	H'00000000	H'E8201934	32
Interrupt processor target register 78	ICDIPTR78	R/W	H'00000000	H'E8201938	32
Interrupt processor target register 79	ICDIPTR79	R/W	H'00000000	H'E820193C	32
Interrupt processor target register 80	ICDIPTR80	R/W	H'00000000	H'E8201940	32
Interrupt processor target register 81	ICDIPTR81	R/W	H'00000000	H'E8201944	32
Interrupt processor target register 82	ICDIPTR82	R/W	H'00000000	H'E8201948	32
Interrupt processor target register 83	ICDIPTR83	R/W	H'00000000	H'E820194C	32
Interrupt processor target register 84	ICDIPTR84	R/W	H'00000000	H'E8201950	32
Interrupt processor target register 85	ICDIPTR85	R/W	H'00000000	H'E8201954	32
Interrupt processor target register 86	ICDIPTR86	R/W	H'00000000	H'E8201958	32

Table 7.2 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Interrupt processor target register 87	ICDIPTR87	R/W	H'00000000	H'E820195C	32
Interrupt processor target register 88	ICDIPTR88	R/W	H'00000000	H'E8201960	32
Interrupt processor target register 89	ICDIPTR89	R/W	H'00000000	H'E8201964	32
Interrupt processor target register 90	ICDIPTR90	R/W	H'00000000	H'E8201968	32
Interrupt processor target register 91	ICDIPTR91	R/W	H'00000000	H'E820196C	32
Interrupt processor target register 92	ICDIPTR92	R/W	H'00000000	H'E8201970	32
Interrupt processor target register 93	ICDIPTR93	R/W	H'00000000	H'E8201974	32
Interrupt processor target register 94	ICDIPTR94	R/W	H'00000000	H'E8201978	32
Interrupt processor target register 95	ICDIPTR95	R/W	H'00000000	H'E820197C	32
Interrupt processor target register 96	ICDIPTR96	R/W	H'00000000	H'E8201980	32
Interrupt processor target register 97	ICDIPTR97	R/W	H'00000000	H'E8201984	32
Interrupt processor target register 98	ICDIPTR98	R/W	H'00000000	H'E8201988	32
Interrupt processor target register 99	ICDIPTR99	R/W	H'00000000	H'E820198C	32
Interrupt processor target register 100	ICDIPTR100	R/W	H'00000000	H'E8201990	32
Interrupt processor target register 101	ICDIPTR101	R/W	H'00000000	H'E8201994	32
Interrupt processor target register 102	ICDIPTR102	R/W	H'00000000	H'E8201998	32
Interrupt processor target register 103	ICDIPTR103	R/W	H'00000000	H'E820199C	32
Interrupt processor target register 104	ICDIPTR104	R/W	H'00000000	H'E82019A0	32
Interrupt processor target register 105	ICDIPTR105	R/W	H'00000000	H'E82019A4	32
Interrupt processor target register 106	ICDIPTR106	R/W	H'00000000	H'E82019A8	32
Interrupt processor target register 107	ICDIPTR107	R/W	H'00000000	H'E82019AC	32
Interrupt processor target register 108	ICDIPTR108	R/W	H'00000000	H'E82019B0	32
Interrupt processor target register 109	ICDIPTR109	R/W	H'00000000	H'E82019B4	32
Interrupt processor target register 110	ICDIPTR110	R/W	H'00000000	H'E82019B8	32
Interrupt processor target register 111	ICDIPTR111	R/W	H'00000000	H'E82019BC	32
Interrupt processor target register 112	ICDIPTR112	R/W	H'00000000	H'E82019C0	32
Interrupt processor target register 113	ICDIPTR113	R/W	H'00000000	H'E82019C4	32
Interrupt processor target register 114	ICDIPTR114	R/W	H'00000000	H'E82019C8	32
Interrupt processor target register 115	ICDIPTR115	R/W	H'00000000	H'E82019CC	32
Interrupt processor target register 116	ICDIPTR116	R/W	H'00000000	H'E82019D0	32
Interrupt processor target register 117	ICDIPTR117	R/W	H'00000000	H'E82019D4	32
Interrupt processor target register 118	ICDIPTR118	R/W	H'00000000	H'E82019D8	32
Interrupt processor target register 119	ICDIPTR119	R/W	H'00000000	H'E82019DC	32
Interrupt processor target register 120	ICDIPTR120	R/W	H'00000000	H'E82019E0	32
Interrupt processor target register 121	ICDIPTR121	R/W	H'00000000	H'E82019E4	32
Interrupt processor target register 122	ICDIPTR122	R/W	H'00000000	H'E82019E8	32
Interrupt processor target register 123	ICDIPTR123	R/W	H'00000000	H'E82019EC	32
Interrupt processor target register 124	ICDIPTR124	R/W	H'00000000	H'E82019F0	32
Interrupt processor target register 125	ICDIPTR125	R/W	H'00000000	H'E82019F4	32
Interrupt processor target register 126	ICDIPTR126	R/W	H'00000000	H'E82019F8	32
Interrupt processor target register 127	ICDIPTR127	R/W	H'00000000	H'E82019FC	32
Interrupt processor target register 128	ICDIPTR128	R/W	H'00000000	H'E8201A00	32
Interrupt processor target register 129	ICDIPTR129	R/W	H'00000000	H'E8201A04	32
Interrupt processor target register 130	ICDIPTR130	R/W	H'00000000	H'E8201A08	32
Interrupt processor target register 131	ICDIPTR131	R/W	H'00000000	H'E8201A0C	32

Table 7.2 Register Configuration

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Interrupt processor target register 132	ICDIPTR132	R/W	H'00000000	H'E8201A10	32
Interrupt processor target register 133	ICDIPTR133	R/W	H'00000000	H'E8201A14	32
Interrupt processor target register 134	ICDIPTR134	R/W	H'00000000	H'E8201A18	32
Interrupt processor target register 135	ICDIPTR135	R/W	H'00000000	H'E8201A1C	32
Interrupt processor target register 136	ICDIPTR136	R/W	H'00000000	H'E8201A20	32
Interrupt processor target register 137	ICDIPTR137	R/W	H'00000000	H'E8201A24	32
Interrupt processor target register 138	ICDIPTR138	R/W	H'00000000	H'E8201A28	32
Interrupt processor target register 139	ICDIPTR139	R/W	H'00000000	H'E8201A2C	32
Interrupt processor target register 140	ICDIPTR140	R/W	H'00000000	H'E8201A30	32
Interrupt processor target register 141	ICDIPTR141	R/W	H'00000000	H'E8201A34	32
Interrupt processor target register 142	ICDIPTR142	R/W	H'00000000	H'E8201A38	32
Interrupt processor target register 143	ICDIPTR143	R/W	H'00000000	H'E8201A3C	32
Interrupt processor target register 144	ICDIPTR144	R/W	H'00000000	H'E8201A40	32
Interrupt processor target register 145	ICDIPTR145	R/W	H'00000000	H'E8201A44	32
Interrupt processor target register 146	ICDIPTR146	R/W	H'00000000	H'E8201A48	32
Interrupt configuration register 0	ICDICFR0	R	H'AAAAAAAA	H'E8201C00	32
Interrupt configuration register 1	ICDICFR1	R/W	H'55555555	H'E8201C04	32
Interrupt configuration register 2	ICDICFR2	R/W	H'55555555	H'E8201C08	32
Interrupt configuration register 3	ICDICFR3	R/W	H'55555555	H'E8201C0C	32
Interrupt configuration register 4	ICDICFR4	R/W	H'55555555	H'E8201C10	32
Interrupt configuration register 5	ICDICFR5	R/W	H'55555555	H'E8201C14	32
Interrupt configuration register 6	ICDICFR6	R/W	H'55555555	H'E8201C18	32
Interrupt configuration register 7	ICDICFR7	R/W	H'55555555	H'E8201C1C	32
Interrupt configuration register 8	ICDICFR8	R/W	H'55555555	H'E8201C20	32
Interrupt configuration register 9	ICDICFR9	R/W	H'55555555	H'E8201C24	32
Interrupt configuration register 10	ICDICFR10	R/W	H'55555555	H'E8201C28	32
Interrupt configuration register 11	ICDICFR11	R/W	H'55555555	H'E8201C2C	32
Interrupt configuration register 12	ICDICFR12	R/W	H'55555555	H'E8201C30	32
Interrupt configuration register 13	ICDICFR13	R/W	H'55555555	H'E8201C34	32
Interrupt configuration register 14	ICDICFR14	R/W	H'55555555	H'E8201C38	32
Interrupt configuration register 15	ICDICFR15	R/W	H'55555555	H'E8201C3C	32
Interrupt configuration register 16	ICDICFR16	R/W	H'55555555	H'E8201C40	32
Interrupt configuration register 17	ICDICFR17	R/W	H'55555555	H'E8201C44	32
Interrupt configuration register 18	ICDICFR18	R/W	H'55555555	H'E8201C48	32
Interrupt configuration register 19	ICDICFR19	R/W	H'55555555	H'E8201C4C	32
Interrupt configuration register 20	ICDICFR20	R/W	H'55555555	H'E8201C50	32
Interrupt configuration register 21	ICDICFR21	R/W	H'55555555	H'E8201C54	32
Interrupt configuration register 22	ICDICFR22	R/W	H'55555555	H'E8201C58	32
Interrupt configuration register 23	ICDICFR23	R/W	H'55555555	H'E8201C5C	32
Interrupt configuration register 24	ICDICFR24	R/W	H'55555555	H'E8201C60	32
Interrupt configuration register 25	ICDICFR25	R/W	H'55555555	H'E8201C64	32
Interrupt configuration register 26	ICDICFR26	R/W	H'55555555	H'E8201C68	32
Interrupt configuration register 27	ICDICFR27	R/W	H'55555555	H'E8201C6C	32
Interrupt configuration register 28	ICDICFR28	R/W	H'55555555	H'E8201C70	32
Interrupt configuration register 29	ICDICFR29	R/W	H'55555555	H'E8201C74	32

**Table 7.2 Register Configuration**

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Interrupt configuration register 30	ICDICFR30	R/W	H'55555555	H'E8201C78	32
Interrupt configuration register 31	ICDICFR31	R/W	H'55555555	H'E8201C7C	32
Interrupt configuration register 32	ICDICFR32	R/W	H'55555555	H'E8201C80	32
Interrupt configuration register 33	ICDICFR33	R/W	H'55555555	H'E8201C84	32
Interrupt configuration register 34	ICDICFR34	R/W	H'55555555	H'E8201C88	32
Interrupt configuration register 35	ICDICFR35	R/W	H'55555555	H'E8201C8C	32
Interrupt configuration register 36	ICDICFR36	R/W	H'55555555	H'E8201C90	32
PPI status register	ppi_status	R	H'00000000	H'E8201D00	32
SPI status register 0	spi_status0	R	H'00000000	H'E8201D04	32
SPI status register 1	spi_status1	R	H'00000000	H'E8201D08	32
SPI status register 2	spi_status2	R	H'00000000	H'E8201D0C	32
SPI status register 3	spi_status3	R	H'00000000	H'E8201D10	32
SPI status register 4	spi_status4	R	H'00000000	H'E8201D14	32
SPI status register 5	spi_status5	R	H'00000000	H'E8201D18	32
SPI status register 6	spi_status6	R	H'00000000	H'E8201D1C	32
SPI status register 7	spi_status7	R	H'00000000	H'E8201D20	32
SPI status register 8	spi_status8	R	H'00000000	H'E8201D24	32
SPI status register 9	spi_status9	R	H'00000000	H'E8201D28	32
SPI status register 10	spi_status10	R	H'00000000	H'E8201D2C	32
SPI status register 11	spi_status11	R	H'00000000	H'E8201D30	32
SPI status register 12	spi_status12	R	H'00000000	H'E8201D34	32
SPI status register 13	spi_status13	R	H'00000000	H'E8201D38	32
SPI status register 14	spi_status14	R	H'00000000	H'E8201D3C	32
SPI status register 15	spi_status15	R	H'00000000	H'E8201D40	32
SPI status register 16	spi_status16	R	H'00000000	H'E8201D44	32
Software generation interrupt register	ICDSGIR	W	H'00000000	H'E8201F00	32
CPU interface control register	ICCICR	R/W	H'00000000	H'E8202000	32
Interrupt priority mask register	ICCPMR	R/W	H'00000000	H'E8202004	32
Binary point register	ICCBPR	R/W	H'00000002	H'E8202008	32
Interrupt acknowledge register	ICCIAR	R	H'000003FF	H'E820200C	32
End-of-interrupt register	ICCEOIR	W	-	H'E8202010	32
Running priority register	ICCRPR	R	H'000000FF	H'E8202014	32
Highest pending interrupt register	ICCHPIR	R	H'000003FF	H'E8202018	32
Aliased binary point register	ICCABPR	R/W	H'00000003	H'E820201C	32
CPU interface implementer identification register	ICCIIDR	R	H'3901043B	H'E82020FC	32

Note 1. When the NMI pin is high, becomes H'8001; when low, becomes H'0001.

Note 2. Only 0 can be written after reading 1, to clear the flag.



### 7.3.1 Interrupt Control Register 0 (ICR0)

ICR0 is a 16-bit register that sets the input signal detection mode for the external interrupt input pin NMI, and indicates the input level at the NMI pin.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NMIL	-	-	-	-	-	-	NMIE	-	-	-	-	-	-	NMIF	-
Initial value:	*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R/(W)*2	R

Notes: 1. 1 when the NMI pin is high, and 0 when the NMI pin is low.  
2. Only 0 can be written after reading 1, to clear the flag.

Bit	Bit Name	Initial Value	R/W	Description
15	NMIL	*1	R	<b>NMI Input Level</b> Sets the level of the signal input at the NMI pin. The NMI pin level can be obtained by reading this bit. This bit cannot be modified. 0: Low level is input to NMI pin 1: High level is input to NMI pin
14 to 9	—	All 0	R	<b>Reserved</b> These bits are always read as 0. The write value should always be 0.
8	NMIE	0	R/W	<b>NMI Edge Select</b> Selects whether the falling or rising edge of the interrupt request signal on the NMI pin is detected. 0: Interrupt request is detected on falling edge of NMI input 1: Interrupt request is detected on rising edge of NMI input
7 to 2	—	All 0	R	<b>Reserved</b> These bits are always read as 0. The write value should always be 0.
1	NMIF	0	R/(W)*2	<b>NMI Interrupt Request</b> This bit indicates the status of the NMI interrupt request. This bit cannot be modified. 0: NMI interrupt request has not occurred [Clearing conditions] • Cleared by changing NMIE of ICR0 • Cleared by reading NMIF while NMIF = 1, then writing 0 to NMIF 1: NMI interrupt request is detected [Setting condition] • Edge corresponding to NMIE of ICR0 has occurred at NMI pin
0	—	0	R	<b>Reserved</b> This bit is always read as 0. The write value should always be 0.

### 7.3.2 Interrupt Control Register 1 (ICR1)

ICR1 is a 16-bit register that specifies the detection mode for external interrupt input pins IRQ7 to IRQ0 individually: low level, falling edge, rising edge, or both edges.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IRQ71S	IRQ70S	IRQ61S	IRQ60S	IRQ51S	IRQ50S	IRQ41S	IRQ40S	IRQ31S	IRQ30S	IRQ21S	IRQ20S	IRQ11S	IRQ10S	IRQ01S	IRQ00S
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	IRQ71S	0	R/W	IRQ Sense Select
14	IRQ70S	0	R/W	These bits select whether interrupt signals corresponding to pins IRQ7 to IRQ0 are detected by a low level, falling edge, rising edge, or both edges. 00: Interrupt request is detected on low level of IRQn input 01: Interrupt request is detected on falling edge of IRQn input 10: Interrupt request is detected on rising edge of IRQn input 11: Interrupt request is detected on both edges of IRQn input
13	IRQ61S	0	R/W	
12	IRQ60S	0	R/W	
11	IRQ51S	0	R/W	
10	IRQ50S	0	R/W	
9	IRQ41S	0	R/W	
8	IRQ40S	0	R/W	
7	IRQ31S	0	R/W	
6	IRQ30S	0	R/W	
5	IRQ21S	0	R/W	
4	IRQ20S	0	R/W	
3	IRQ11S	0	R/W	
2	IRQ10S	0	R/W	
1	IRQ01S	0	R/W	
0	IRQ00S	0	R/W	

[Legend]

n = 7 to 0

### 7.3.3 IRQ Interrupt Request Register (IRQRR)

IRQRR is a 16-bit register that indicates interrupt requests from external input pins IRQ7 to IRQ0. If edge detection is set for the IRQ7 to IRQ0 interrupts, writing 0 to the IRQ7F to IRQ0F bits after reading IRQ7F to IRQ0F = 1 cancels the retained interrupts.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	IRQ7F	IRQ6F	IRQ5F	IRQ4F	IRQ3F	IRQ2F	IRQ1F	IRQ0F
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*

Note: \* Only 0 can be written to clear the flag after 1 is read.

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	IRQ7F	0	R/(W)*	IRQ Interrupt Request
6	IRQ6F	0	R/(W)*	These bits indicate the status of the IRQ7 to IRQ0 interrupt requests.
5	IRQ5F	0	R/(W)*	Level detection: 0: IRQn interrupt request has not occurred [Clearing condition]
4	IRQ4F	0	R/(W)*	• IRQn input is high
3	IRQ3F	0	R/(W)*	1: IRQn interrupt has occurred
2	IRQ2F	0	R/(W)*	[Setting condition]
1	IRQ1F	0	R/(W)*	• IRQn input is low
0	IRQ0F	0	R/(W)*	Edge detection: 0: IRQn interrupt request is not detected [Clearing conditions] • Cleared by reading IRQnF while IRQnF = 1, then writing 0 to IRQnF • Cleared by executing IRQn interrupt exception handling 1: IRQn interrupt request is detected [Setting condition] • Edge corresponding to IRQn1S or IRQn0S of ICR1 has occurred at IRQn pin

## 7.4 Interrupt Sources

There are four types of interrupt sources: NMI, IRQ, on-chip peripheral modules, and pin interrupts. Each interrupt has a priority level (0 to 32), with 0 the highest and 32 the lowest.

### 7.4.1 NMI Interrupt

The NMI interrupt with the highest priority is accepted by the CPU as an FIQ exception all times. NMI interrupt requests are edge-detected, and the NMI edge select bit (NMIE) in ICR0 selects whether the rising edge or falling edge is detected. The status of the interrupt request can be checked by reading the NMI interrupt request bit (NMIF) in the ICR0. When the NME bit is changed, the NMI interrupt request that is retained is cleared.

When the NMI pin keeps level (low level after the falling edge or high level after the rising edge) in software standby mode until software standby mode is canceled by cancel source other than NMI (until the clock is initiated after the oscillation settling), that edge of the NMI in software standby mode can be detected.

When deep standby mode is entered, deep standby mode is canceled by the NMI interrupt.

### 7.4.2 IRQ Interrupts

IRQ interrupts are input from pins IRQ7 to IRQ0. For the IRQ interrupts, low-level, falling-edge, rising-edge, or both-edge detection can be selected individually for each pin by the IRQ sense select bits (IRQ71S to IRQ01S and IRQ70S to IRQ00S) in interrupt control register 1 (ICR1).

When using low-level sensing for IRQ interrupts, an interrupt request signal is sent to the interrupt controller while the IRQ7 to IRQ0 pins are low. An interrupt request signal is stopped being sent to the interrupt controller when the IRQ7 to IRQ0 pins are driven high. The status of the interrupt requests can be checked by reading the IRQ interrupt request bits (IRQ7F to IRQ0F) in the IRQ interrupt request register (IRQRR).

When using edge-sensing for IRQ interrupts, an interrupt request is detected due to change of the IRQ7 to IRQ0 pin states, and an interrupt request signal is sent to the interrupt controller. The result of IRQ interrupt request detection is retained until that interrupt request is accepted. Whether IRQ interrupt requests have been detected or not can be checked by reading the IRQ7F to IRQ0F bits in IRQRR. Writing 0 to these bits after reading them as 1 clears the result of IRQ interrupt request detection.

When returning from IRQ interrupt exception service routine, execute the return instruction after confirming that the interrupt request has been cleared by the IRQ interrupt request register (IRQRR) so as not to accidentally receive the interrupt request again.

### 7.4.3 On-Chip Peripheral Module Interrupts

On-chip peripheral module interrupts are generated by the following on-chip peripheral modules:

- Direct memory access controller
- USB 2.0 host/function module
- Video display controller 5
- Image renderer for display
- Image renderer
- JPEG codec unit
- Display out comparison unit
- OpenVG™-compliant Renesas graphics processor
- OS timer
- Bus state controller
- Watchdog timer
- Multi-function timer pulse unit 2
- Motor control PWM timer
- Sound generator
- 12-bit A/D converter
- Serial sound interface
- Renesas SPDIF interface
- I<sup>2</sup>C bus interface
- Serial communication interface with FIFO
- Clock-synchronized serial I/O module with FIFO
- CAN interface
- Renesas serial peripheral interface
- IEBus™ controller
- CD-ROM decoder
- NAND flash memory controller
- MMC host interface
- SD host interface
- Realtime clock
- SCUX
- Media local bus
- Dynamic range compression
- LIN interface
- Serial communication interface
- EthernetAVB
- Ethernet controller
- Capture engine unit
- On-chip RAM
- Internal bus
- Pixel format converter

When returning from the interrupt exception service routine for an interrupt request at the peripheral-module level, only

execute the return instruction after confirming that the interrupt source has been cleared so that the interrupt request is not accidentally received again.

#### 7.4.4 Pin Interrupts

Pin interrupts are input from pins TINT170 to TINT0. For the pin interrupts, high-level or rising-edge detection can be selected individually for each pin by the interrupt configuration registers (ICDICFRn).

For a description of the interrupt configuration registers (ICDICFRn), see the GIC architecture specification.

### 7.5 Interrupt IDs

Table 7.3 lists the interrupt sources and their interrupt IDs, and the registers for setting the interrupt sources.

Each interrupt source is allocated a different interrupt ID. To control notification of the interrupt source to the CPU and reference its status, it is necessary to set and reference the following registers which correspond to given interrupt IDs.

- |                                       |          |
|---------------------------------------|----------|
| • Interrupt security register         | ICDISRn  |
| • Interrupt set-enable register       | ICDISERn |
| • Interrupt clear-enable register     | ICDICERn |
| • Interrupt set-pending register      | ICDISPRn |
| • Interrupt clear-pending register    | ICDICPRn |
| • Active bit register                 | ICDABRn  |
| • Interrupt configuration register    | ICDICFRn |
| • Interrupt priority register         | ICDIPRn  |
| • Interrupt processor target register | ICDIPTRn |

Section 7.6.7, Initial Settings, gives a general example of how to set the registers. For details on individual registers, see the GIC architecture specification.

Table 7.3 List of Interrupt IDs

Interrupt Source				Register Allocation							
Module	Channel	Request Source Name	Interrupt Request Edge/Level	Inter-rupt ID	ICDISRn ICDISERn ICDICERn ICDISPRn ICDICPRn ICDABRn	Bit	ICDICFRn	Bit	ICDIPRn ICDIPTRn	Bit	
GIC software interrupt	-	-	-	0	0	0	0	1 to 0	0	7 to 0	
				1	1	3 to 2	15 to 8				
				2	2	5 to 4	23 to 16				
				3	3	7 to 6	31 to 24				
				4	4	9 to 8	1	7 to 0			
				5	5	11 to 10	15 to 8				
				6	6	13 to 12	23 to 16				
				7	7	15 to 14	31 to 24				
				8	8	17 to 16	2	7 to 0			
				9	9	19 to 18	15 to 8				
				10	10	21 to 20	23 to 16				
				11	11	23 to 22	31 to 24				
				12	12	25 to 24	3	7 to 0			
				13	13	27 to 26	15 to 8				
				14	14	29 to 28	23 to 16				
15	15	31 to 30	31 to 24								
CPU	0	PMUIRQ0	Level	16	16	1	1 to 0	4	7 to 0		
		COMMRX0	Level	17	17	3 to 2	15 to 8				
		COMMTX0	Level	18	18	5 to 4	23 to 16				
		CTIIRQ0	Level	19	19	7 to 6	31 to 24				
IRQ	-	IRQ0	Level	32	1	0	2	1 to 0	8	7 to 0	
		IRQ1	Level	33	1	3 to 2	15 to 8				
		IRQ2	Level	34	2	5 to 4	23 to 16				
		IRQ3	Level	35	3	7 to 6	31 to 24				
		IRQ4	Level	36	4	9 to 8	9	7 to 0			
		IRQ5	Level	37	5	11 to 10	15 to 8				
		IRQ6	Level	38	6	13 to 12	23 to 16				
		IRQ7	Level	39	7	15 to 14	31 to 24				
Secondary cache	-	PL310ERR	Level	40	8	17 to 16	10	7 to 0			

Table 7.3 List of Interrupt IDs

Interrupt Source				Register Allocation							
Module	Channel	Request Source Name	Interrupt Request Edge/Level	Inter-rupt ID	ICDISRn ICDISERn ICDICERn ICDISPRn ICDICPRn ICDABRn	Bit	ICDICFRn	Bit	ICDIPRn ICDIPTRn	Bit	
Direct memory access controller	0	DMAINT0	Edge	41	1	9	2	19 to 18	10	15 to 8	
	1	DMAINT1	Edge	42		10		21 to 20		23 to 16	
	2	DMAINT2	Edge	43		11		23 to 22		31 to 24	
	3	DMAINT3	Edge	44		12		25 to 24	11	7 to 0	
	4	DMAINT4	Edge	45		13		27 to 26		15 to 8	
	5	DMAINT5	Edge	46		14		29 to 28		23 to 16	
	6	DMAINT6	Edge	47		15		31 to 30		31 to 24	
	7	DMAINT7	Edge	48		16	3	1 to 0	12	7 to 0	
	8	DMAINT8	Edge	49		17		3 to 2		15 to 8	
	9	DMAINT9	Edge	50		18		5 to 4		23 to 16	
	10	DMAINT10	Edge	51		19		7 to 6		31 to 24	
	11	DMAINT11	Edge	52		20		9 to 8	13	7 to 0	
	12	DMAINT12	Edge	53		21		11 to 10		15 to 8	
	13	DMAINT13	Edge	54		22		13 to 12		23 to 16	
	14	DMAINT14	Edge	55		23		15 to 14		31 to 24	
		-	DMAERR	Edge	56		24		17 to 16	14	7 to 0
Reserved				57		25		19 to 18		15 to 8	
				58		26		21 to 20		23 to 16	
				59		27		23 to 22		31 to 24	
				60		28		25 to 24	15	7 to 0	
				61		29		27 to 26		15 to 8	
				62		30		29 to 28		23 to 16	
				63		31		31 to 30		31 to 24	
				64	2	0	4	1 to 0	16	7 to 0	
				65		1		3 to 2		15 to 8	
				66		2		5 to 4		23 to 16	
				67		3		7 to 6		31 to 24	
				68		4		9 to 8	17	7 to 0	
				69		5		11 to 10		15 to 8	
				70		6		13 to 12		23 to 16	
				71		7		15 to 14		31 to 24	
				72		8		17 to 16	18	7 to 0	
USB 2.0 host/function module	0	USBIO	Level	73		9		19 to 18		15 to 8	
	1	USB11	Level	74		10		21 to 20		23 to 16	



Table 7.3 List of Interrupt IDs

Interrupt Source		Register Allocation								
Module	Channel	Request Source Name	Interrupt Request Edge/Level	Inter-rupt ID	ICDISRn ICDISERn ICDICERn ICDISPRn ICDICPRn ICDABRn	ICDICFRn		ICDIPRn		Bit
						Bit	Bit	Bit	Bit	
Video display controller 5	0	S0_VI_VSYNC0	Level	75	2	11	4	23 to 22	18	31 to 24
		S0_LO_VSYNC0	Level	76		12		25 to 24	19	7 to 0
		S0_VSYNCERR0	Level	77		13		27 to 26		15 to 8
		GR3_VLINE0	Level	78		14		29 to 28		23 to 16
		S0_VFIELD0	Level	79		15		31 to 30		31 to 24
		IV1_VBUFERR0	Level	80		16	5	1 to 0	20	7 to 0
		IV3_VBUFERR0	Level	81		17		3 to 2		15 to 8
		IV5_VBUFERR0	Level	82		18		5 to 4		23 to 16
		IV6_VBUFERR0	Level	83		19		7 to 6		31 to 24
		S0_WLINE0	Level	84		20		9 to 8	21	7 to 0
		S1_VI_VSYNC0	Level	85		21		11 to 10		15 to 8
		S1_LO_VSYNC0	Level	86		22		13 to 12		23 to 16
		S1_VSYNCERR0	Level	87		23		15 to 14		31 to 24
		S1_VFIELD0	Level	88		24		17 to 16	22	7 to 0
		IV2_VBUFERR0	Level	89		25		19 to 18		15 to 8
		IV4_VBUFERR0	Level	90		26		21 to 20		23 to 16
		S1_WLINE0	Level	91		27		23 to 22		31 to 24
		OIR_VI_VSYNC0	Level	92		28		25 to 24	23	7 to 0
		OIR_LO_VSYNC0	Level	93		29		27 to 26		15 to 8
		OIR_VSYNCERR0	Level	94		30		29 to 28		23 to 16
OIR_VFIELD0	Level	95		31		31 to 30		31 to 24		
		IV7_VBUFERR0	Level	96	3	0	6	1 to 0	24	7 to 0
		IV8_VBUFERR0	Level	97		1		3 to 2		15 to 8
Reserved				98		2		5 to 4		23 to 16
Video display controller 5	1	S0_VI_VSYNC1	Level	99		3		7 to 6		31 to 24
		S0_LO_VSYNC1	Level	100		4		9 to 8	25	7 to 0
		S0_VSYNCERR1	Level	101		5		11 to 10		15 to 8
		GR3_VLINE1	Level	102		6		13 to 12		23 to 16
		S0_VFIELD1	Level	103		7		15 to 14		31 to 24
		IV1_VBUFERR1	Level	104		8		17 to 16	26	7 to 0
		IV3_VBUFERR1	Level	105		9		19 to 18		15 to 8
		IV5_VBUFERR1	Level	106		10		21 to 20		23 to 16
		IV6_VBUFERR1	Level	107		11		23 to 22		31 to 24
		S0_WLINE1	Level	108		12		25 to 24	27	7 to 0
		S1_VI_VSYNC1	Level	109		13		27 to 26		15 to 8
		S1_LO_VSYNC1	Level	110		14		29 to 28		23 to 16
S1_VSYNCERR1	Level	111		15		31 to 30		31 to 24		

Table 7.3 List of Interrupt IDs

Interrupt Source			Register Allocation								
Module	Channel	Request Source Name	Interrupt Request Edge/Level	Inter-rupt ID	ICDISRn ICDISERn ICDICERn ICDISPRn ICDICPRn ICDABRn	Bit	ICDICFRn	Bit	ICDIPRn ICDIPTRn	Bit	
Video display controller 5	1	S1_VFIELD1	Level	112	3	16	7	1 to 0	28	7 to 0	
		IV2_VBUFERR1	Level	113		17		3 to 2		15 to 8	
		IV4_VBUFERR1	Level	114		18		5 to 4		23 to 16	
		S1_WLINE1	Level	115		19		7 to 6		31 to 24	
		OIR_VI_VSYNC1	Level	116		20		9 to 8		29	7 to 0
		OIR_LO_VSYNC1	Level	117		21		11 to 10		15 to 8	
		OIR_VLINE1	Level	118		22		13 to 12		23 to 16	
		OIR_VFIELD1	Level	119		23		15 to 14		31 to 24	
		IV7_VBUFERR1	Level	120		24		17 to 16		30	7 to 0
		IV8_VBUFERR1	Level	121	25	19 to 18	15 to 8				
Reserved				122	26	21 to 20	23 to 16				
Image renderer for display	-	IMRDI	Level	123	27	23 to 22	31 to 24				
Image renderer	0	IMR2I0	Level	124	28	25 to 24	31	7 to 0			
	1	IMR2I1	Level	125	29	27 to 26	15 to 8				
JPEG codec unit	-	JEDI	Level	126	30	29 to 28	23 to 16				
		JDTI	Level	127	31	31 to 30	31 to 24				
Display out comparison unit	0	CMP0	Level	128	4	0	8	1 to 0	32	7 to 0	
	1	CMP1	Level	129		1		3 to 2		15 to 8	
OpenVG™-compliant Renesas graphics processor	-	INT0	Level	130	2	5 to 4	23 to 16				
		INT1	Level	131	3	7 to 6	31 to 24				
		INT2	Level	132	4	9 to 8	33	7 to 0			
		INT3	Level	133	5	11 to 10	15 to 8				
OS timer	0	OSTM0TINT	Edge	134	6	13 to 12	23 to 16				
	1	OSTM1TINT	Edge	135	7	15 to 14	31 to 24				
Bus state controller	-	CMI	Level	136	8	17 to 16	34	7 to 0			
		WTOUT	Level	137	9	19 to 18	15 to 8				
Watchdog timer	-	ITI	Level	138	10	21 to 20	23 to 16				

Table 7.3 List of Interrupt IDs

Interrupt Source			Register Allocation							
Module	Channel	Request Source Name	Interrupt Request Edge/Level	Inter-rupt ID	ICDISRn ICDISERn ICDICERn ICDISPRn ICDICPRn ICDABRn	Bit	ICDICFRn	Bit	ICDIPRn ICDIPTRn	Bit
Multi-function timer pulse unit 2	0	TGI0A	Level	139	4	11	8	23 to 22	34	31 to 24
		TGI0B	Level	140		12		25 to 24		7 to 0
		TGI0C	Level	141		13		27 to 26		15 to 8
		TGI0D	Level	142		14		29 to 28		23 to 16
		TGI0V	Level	143		15		31 to 30		31 to 24
		TGI0E	Level	144		16		9		1 to 0
	TGI0F	Level	145	17	3 to 2	15 to 8				
	1	TGI1A	Level	146	18	5 to 4	23 to 16			
		TGI1B	Level	147	19	7 to 6	31 to 24			
		TGI1V	Level	148	20	9 to 8	37	7 to 0		
		TGI1U	Level	149	21	11 to 10		15 to 8		
	2	TGI2A	Level	150	22	13 to 12	23 to 16			
		TGI2B	Level	151	23	15 to 14	31 to 24			
		TGI2V	Level	152	24	17 to 16	38	7 to 0		
		TGI2U	Level	153	25	19 to 18		15 to 8		
	3	TGI3A	Level	154	26	21 to 20	23 to 16			
		TGI3B	Level	155	27	23 to 22	31 to 24			
		TGI3C	Level	156	28	25 to 24	39	7 to 0		
		TGI3D	Level	157	29	27 to 26		15 to 8		
		TGI3V	Level	158	30	29 to 28	23 to 16			
4	TGI4A	Level	159	31	31 to 30	31 to 24				
	TGI4B	Level	160	5	0	10	1 to 0	40	7 to 0	
	TGI4C	Level	161		1		3 to 2		15 to 8	
	TGI4D	Level	162		2		5 to 4		23 to 16	
	TGI4V	Level	163		3		7 to 6		31 to 24	
Motor control PWM timer	-	CMI1	Level	164	4	9 to 8	41	7 to 0		
		CMI2	Level	165	5	11 to 10		15 to 8		
Sound generator	0	SGDEI0	Level	166	6	13 to 12	23 to 16			
	1	SGDEI1	Level	167	7	15 to 14	31 to 24			
	2	SGDEI2	Level	168	8	17 to 16	42	7 to 0		
	3	SGDEI3	Level	169	9	19 to 18		15 to 8		
12-bit A/D converter	-	ADI	Level	170	10	21 to 20	23 to 16			
		LMTI	Level	171	11	23 to 22	31 to 24			

Table 7.3 List of Interrupt IDs

Interrupt Source				Register Allocation									
Module	Channel	Request Source Name	Interrupt Request Edge/Level	Inter-rupt ID	ICDISRn ICDISERn ICDICERn ICDISPRn ICDICPRn ICDABRn	Bit	ICDICFRn	Bit	ICDIPRn ICDIPTRn	Bit			
Serial sound interface	0	SSII0	Level	172	5	12	10	25 to 24	43	7 to 0			
		SSIRXI0	Level	173		13		27 to 26		15 to 8			
		SSITXI0	Level	174		14		29 to 28		23 to 16			
	1	SSII1	Level	175		15		31 to 30		31 to 24			
		SSIRXI1	Level	176		16		11		1 to 0	44	7 to 0	
		SSITXI1	Level	177		17		3 to 2		15 to 8			
	2	SSII2	Level	178		18		5 to 4		23 to 16			
		SSIRTI2	Level	179		19		7 to 6		31 to 24			
	3	SSII3	Level	180		20		9 to 8	45	7 to 0			
		SSIRXI3	Level	181		21		11 to 10		15 to 8			
		SSITXI3	Level	182		22		13 to 12		23 to 16			
	4	SSII4	Level	183		23		15 to 14		31 to 24			
		SSIRTI4	Level	184		24		17 to 16		46	7 to 0		
	5	SSII5	Level	185		25		19 to 18		15 to 8			
		SSIRXI5	Level	186		26		21 to 20		23 to 16			
SSITXI5		Level	187	27		23 to 22		31 to 24					
Renesas SPDIF interface	-	SPDIFI	Level	188		28	11	25 to 24	47	7 to 0			
I <sup>2</sup> C bus interface	0	INTIICTEI0	Level	189		29		27 to 26		15 to 8			
		INTIICRI0	Edge	190		30		29 to 28		23 to 16			
		INTIICTI0	Edge	191		31		31 to 30		31 to 24			
		INTIICSPI0	Level	192		6		0		12	1 to 0	48	7 to 0
		INTIICSTI0	Level	193				1			3 to 2		15 to 8
		INTIICNAKI0	Level	194				2			5 to 4		23 to 16
		INTIICALI0	Level	195				3			7 to 6		31 to 24
		INTIICTMOI0	Level	196		4		9 to 8		49	7 to 0		
	1	INTIICTEI1	Level	197		5		11 to 10		15 to 8			
		INTIICRI1	Edge	198		6		13 to 12		23 to 16			
		INTIICTI1	Edge	199		7		15 to 14		31 to 24			
		INTIICSPI1	Level	200		8		17 to 16		50	7 to 0		
		INTIICSTI1	Level	201		9		19 to 18		15 to 8			
		INTIICNAKI1	Level	202		10		21 to 20		23 to 16			
		INTIICALI1	Level	203		11		23 to 22		31 to 24			
INTIICTMOI1	Level	204	12	25 to 24	51	7 to 0							

Table 7.3 List of Interrupt IDs

Interrupt Source				Register Allocation						
Module	Channel	Request Source Name	Interrupt Request Edge/Level	Inter-rupt ID	ICDISRn		ICDISERn		ICDICERn	
					ICDISPRn	ICDICPRn	ICDISPRn	ICDICPRn	ICDISPRn	ICDICPRn
					Bit	ICDICFRn	Bit	ICDIPRn	ICDIPTRn	Bit
I <sup>2</sup> C bus interface	2	INTIICTEI2	Level	205	6	13	12	27 to 26	51	15 to 8
		INTIICRI2	Edge	206		14		29 to 28		23 to 16
		INTIICTI2	Edge	207		15		31 to 30		31 to 24
		INTIICSPI2	Level	208		16	13	1 to 0	52	7 to 0
		INTIICSTI2	Level	209		17		3 to 2		15 to 8
		INTIICNAKI2	Level	210		18		5 to 4		23 to 16
		INTIICALI2	Level	211		19		7 to 6		31 to 24
		INTIICTMOI2	Level	212		20		9 to 8	53	7 to 0
	3	INTIICTEI3	Level	213		21		11 to 10		15 to 8
		INTIICRI3	Edge	214		22		13 to 12		23 to 16
		INTIICTI3	Edge	215		23		15 to 14		31 to 24
		INTIICSPI3	Level	216		24		17 to 16	54	7 to 0
		INTIICSTI3	Level	217		25		19 to 18		15 to 8
		INTIICNAKI3	Level	218		26		21 to 20		23 to 16
		INTIICALI3	Level	219		27		23 to 22		31 to 24
		INTIICTMOI3	Level	220		28		25 to 24	55	7 to 0
Serial communication interface with FIFO	0	BRI0	Level	221		29		27 to 26		15 to 8
		ERI0	Level	222		30		29 to 28		23 to 16
		RXI0	Level	223		31		31 to 30		31 to 24
		TXI0	Level	224	7	0	14	1 to 0	56	7 to 0
	1	BRI1	Level	225		1		3 to 2		15 to 8
		ERI1	Level	226		2		5 to 4		23 to 16
		RXI1	Level	227		3		7 to 6		31 to 24
		TXI1	Level	228		4		9 to 8	57	7 to 0
	2	BRI2	Level	229		5		11 to 10		15 to 8
		ERI2	Level	230		6		13 to 12		23 to 16
		RXI2	Level	231		7		15 to 14		31 to 24
		TXI2	Level	232		8		17 to 16	58	7 to 0
	3	BRI3	Level	233		9		19 to 18		15 to 8
		ERI3	Level	234		10		21 to 20		23 to 16
		RXI3	Level	235		11		23 to 22		31 to 24
		TXI3	Level	236		12		25 to 24	59	7 to 0
	4	BRI4	Level	237		13		27 to 26		15 to 8
		ERI4	Level	238		14		29 to 28		23 to 16
		RXI4	Level	239		15		31 to 30		31 to 24

Table 7.3 List of Interrupt IDs

Interrupt Source				Register Allocation							
Module	Channel	Request Source Name	Interrupt Request Edge/Level	Inter-rupt ID	ICDISRn ICDISERn ICDICERn ICDISPRn ICDICPRn ICDABRn	Bit	ICDICFRn	Bit	ICDIPRn ICDIPTRn	Bit	
Serial communication interface with FIFO	4	TXI4	Level	240	7	16	15	1 to 0	60	7 to 0	
		5	BRI5	Level		241		17		3 to 2	15 to 8
			ERI5	Level		242		18		5 to 4	23 to 16
			RXI5	Level		243		19		7 to 6	31 to 24
	6	TXI5	Level	244	20	9 to 8	61	7 to 0			
		BRI6	ERI6	Level	245	21		11 to 10	15 to 8		
			ERI6	Level	246	22		13 to 12	23 to 16		
			RXI6	Level	247	23		15 to 14	31 to 24		
	7	TXI6	Level	248	24	17 to 16	62	7 to 0			
		BRI7	ERI7	Level	249	25		19 to 18	15 to 8		
			ERI7	Level	250	26		21 to 20	23 to 16		
			RXI7	Level	251	27		23 to 22	31 to 24		
	CAN interface	Common	INTRCANGERR	Level	253	8	29	16	27 to 26	63	15 to 8
			INTRCANGRECC	Level	254		30		29 to 28		23 to 16
		0	INTRCAN0REC	Level	255	31	31 to 30	31 to 24			
			INTRCAN0ERR	Level	256	0	1 to 0	64	7 to 0		
INTRCAN0TRX			Level	257	1	3 to 2	15 to 8				
1		INTRCAN1REC	Level	258	2	5 to 4	23 to 16				
		INTRCAN1ERR	Level	259	3	7 to 6	31 to 24				
		INTRCAN1TRX	Level	260	4	9 to 8	65	7 to 0			
2		INTRCAN2REC	Level	261	5	11 to 10		15 to 8			
		INTRCAN2ERR	Level	262	6	13 to 12		23 to 16			
		INTRCAN2TRX	Level	263	7	15 to 14	31 to 24				
3		INTRCAN3REC	Level	264	8	17 to 16	66	7 to 0			
		INTRCAN3ERR	Level	265	9	19 to 18		15 to 8			
	INTRCAN3TRX	Level	266	10	21 to 20	23 to 16					
4	INTRCAN4REC	Level	267	11	23 to 22	67	31 to 24				
	INTRCAN4ERR	Level	268	12	25 to 24		7 to 0				
	INTRCAN4TRX	Level	269	13	27 to 26		15 to 8				

Table 7.3 List of Interrupt IDs

Interrupt Source			Register Allocation							
Module	Channel	Request Source Name	Interrupt Request Edge/Level	Inter-rupt ID	ICDISRn ICDISERn ICDICERn ICDISPRn ICDICPRn ICDABRn	Bit	ICDICFRn	Bit	ICDIPRn ICDIPTRn	Bit
Renesas serial peripheral interface	0	SPEI0	Level	270	8	14	16	29 to 28	67	23 to 16
		SPRI0	Level	271		15		31 to 30		31 to 24
		SPTI0	Level	272		16		17		1 to 0
	1	SPEI1	Level	273	17	3 to 2	15 to 8			
		SPRI1	Level	274	18	5 to 4	23 to 16			
		SPTI1	Level	275	19	7 to 6	31 to 24			
	2	SPEI2	Level	276	20	9 to 8	69	7 to 0		
		SPRI2	Level	277	21	11 to 10	15 to 8			
		SPTI2	Level	278	22	13 to 12	23 to 16			
	3	SPEI3	Level	279	23	15 to 14	31 to 24			
		SPRI3	Level	280	24	17 to 16	70	7 to 0		
		SPTI3	Level	281	25	19 to 18	15 to 8			
4	SPEI4	Level	282	26	21 to 20	23 to 16				
	SPRI4	Level	283	27	23 to 22	31 to 24				
	SPTI4	Level	284	28	25 to 24	71	7 to 0			
IEBus™ controller	-	IEBBTD	Edge	285	29	27 to 26	15 to 8			
		IEBBTERR	Edge	286	30	29 to 28	23 to 16			
		IEBBTSTA	Edge	287	31	31 to 30	31 to 24			
		IEBBTV	Edge	288	9	0	18	1 to 0	72	7 to 0
CD-ROM decoder	-	ISY	Level	289	1	3 to 2	15 to 8			
		IERR	Level	290	2	5 to 4	23 to 16			
		ITARG	Level	291	3	7 to 6	31 to 24			
		ISEC	Level	292	4	9 to 8	73	7 to 0		
		IBUF	Level	293	5	11 to 10	15 to 8			
		IREADY	Level	294	6	13 to 12	23 to 16			
NAND flash memory controller	-	STERB BTOERB	Level	295	7	15 to 14	31 to 24			
		FLTENDI	Level	296	8	17 to 16	74	7 to 0		
		FLTREQ0I	Level	297	9	19 to 18	15 to 8			
MMC host interface	-	MMC0	Level	299	11	23 to 22	31 to 24			
		MMC1	Level	300	12	25 to 24	75	7 to 0		
		MMC2	Level	301	13	27 to 26	15 to 8			

Table 7.3 List of Interrupt IDs

Interrupt Source				Register Allocation						
Module	Channel	Request Source Name	Interrupt Request Edge/Level	Inter-rupt ID	ICDISRn ICDISERn ICDICERn ICDISPRn ICDICPRn ICDABRn	Bit	ICDICFRn	Bit	ICDIPRn ICDIPTRn	Bit
SD host interface	0	SDHI0_3	Level	302	9	14	18	29 to 28	75	23 to 16
		SDHI0_0	Level	303		15		31 to 30		31 to 24
		SDHI0_1	Level	304		16		19		1 to 0
	1	SDHI1_3	Level	305		17	3 to 2	15 to 8		
		SDHI1_0	Level	306		18	5 to 4	23 to 16		
		SDHI1_1	Level	307		19	7 to 6	31 to 24		
Realtime clock	-	ARM	Level	308	20	9 to 8	77	7 to 0		
		PRD	Level	309	21	11 to 10	15 to 8			
		CUP	Level	310	22	13 to 12	23 to 16			
SCUX	-	SCUAI0	Level	311	23	15 to 14	31 to 24			
		SCUAI1	Level	312	24	17 to 16	78	7 to 0		
		SCUFDI0	Level	313	25	19 to 18	15 to 8			
		SCUFDI1	Level	314	26	21 to 20	23 to 16			
		SCUFDI2	Level	315	27	23 to 22	31 to 24			
		SCUFDI3	Level	316	28	25 to 24	79	7 to 0		
		SCUFUI0	Level	317	29	27 to 26	15 to 8			
		SCUFUI1	Level	318	30	29 to 28	23 to 16			
		SCUFUI2	Level	319	31	31 to 30	31 to 24			
		SCUFUI3	Level	320	10	0	20	1 to 0	80	7 to 0
		SCUDVI0	Level	321		1		3 to 2		15 to 8
		SCUDVI1	Level	322		2		5 to 4		23 to 16
				SCUDVI2	Level	323	3	7 to 6	31 to 24	
				SCUDVI3	Level	324	4	9 to 8	81	7 to 0
		Media local bus	-	MLB_CINT	Level	325	5	11 to 10	15 to 8	
MLB_SINT	Level			326	6	13 to 12	23 to 16			
Dynamic range compression	0	DRC1	Level	327	7	15 to 14	31 to 24			
	1	DRC1	Level	328	8	17 to 16	82	7 to 0		
Reserved				329	9	19 to 18	15 to 8			
				330	10	21 to 20	23 to 16			
LIN interface	0	LIN0_INT_T	Edge	331	11	23 to 22	31 to 24			
		LIN0_INT_R	Edge	332	12	25 to 24	83	7 to 0		
		LIN0_INT_S	Edge	333	13	27 to 26	15 to 8			
		LIN0_INT_M	Edge	334	14	29 to 28	23 to 16			
	1	LIN1_INT_T	Edge	335	15	31 to 30	31 to 24			
		LIN1_INT_R	Edge	336	16	21	1 to 0	84	7 to 0	
		LIN1_INT_S	Edge	337	17	3 to 2	15 to 8			
		LIN1_INT_M	Edge	338	18	5 to 4	23 to 16			



Table 7.3 List of Interrupt IDs

Interrupt Source				Register Allocation						
Module	Channel	Request Source Name	Interrupt Request Edge/Level	Inter-rupt ID	ICDISRn ICDISERn ICDICERn ICDISPRn ICDICPRn ICDABRn	Bit	ICDICFRn	Bit	ICDIPRn ICDIPTRn	Bit
Reserved				339	10	19	21	7 to 6	84	31 to 24
				340		20		9 to 8	85	7 to 0
				341		21		11 to 10		15 to 8
				342		22		13 to 12		23 to 16
				343		23		15 to 14		31 to 24
				344		24		17 to 16	86	7 to 0
				345		25		19 to 18		15 to 8
Serial communication interface	0	ERI0	Level	347		27		23 to 22		31 to 24
		RXI0	Edge	348		28		25 to 24	87	7 to 0
		TXI0	Edge	349		29		27 to 26		15 to 8
		TEI0	Level	350		30		29 to 28		23 to 16
	1	ERI1	Level	351		31		31 to 30		31 to 24
		RXI1	Edge	352	11	0	22	1 to 0	88	7 to 0
		TXI1	Edge	353		1		3 to 2		15 to 8
		TEI1	Level	354		2	5 to 4		23 to 16	
Ethernet AVB	-	AVBI_DATA	Level	355		3		7 to 6		31 to 24
		AVBI_ERROR	Level	356		4		9 to 8	89	7 to 0
		AVBI_MANAGE	Level	357		5		11 to 10		15 to 8
		AVBI_MAC	Level	358		6		13 to 12		23 to 16
Ethernet controller	-	ETHERI	Level	359		7	15 to 14		31 to 24	
Reserved				360		8		17 to 16	90	7 to 0
				361		9		19 to 18		15 to 8
				362		10		21 to 20		23 to 16
				363		11		23 to 22		31 to 24
Capture engine unit	-	CEUI	Level	364		12	25 to 24	91	7 to 0	
Reserved				365		13		27 to 26		15 to 8
				366		14		29 to 28		23 to 16
				367		15		31 to 30		31 to 24
				368		16	23	1 to 0	92	7 to 0
				369		17		3 to 2		15 to 8
				370		18		5 to 4		23 to 16
				371		19		7 to 6		31 to 24

Table 7.3 List of Interrupt IDs

Interrupt Source			Register Allocation							
Module	Channel	Request Source Name	Interrupt Request Edge/Level	Inter-rupt ID	ICDISRn ICDISERn ICDICERn ICDISPRn ICDICPRn ICDABRn	Bit	ICDICFRn	Bit	ICDIPRn ICDIPTRn	Bit
Reserved				372	11	20	23	9 to 8	93	7 to 0
				373		21		11 to 10		15 to 8
				374		22		13 to 12		23 to 16
				375		23		15 to 14		31 to 24
				376		24		17 to 16	94	7 to 0
				377		25		19 to 18		15 to 8
				378		26		21 to 20		23 to 16
				379		27		23 to 22		31 to 24
				380		28		25 to 24	95	7 to 0
Internal bus		H2XMLB_ERRINT	Level	381		29		27 to 26		15 to 8
		H2XIC1_ERRINT	Level	382		30		29 to 28		23 to 16
		X2HPERI1_ERRINT	Level	383		31		31 to 30		31 to 24
		X2HPERI2_ERRINT	Level	384	12	0	24	1 to 0	96	7 to 0
		X2HPERI34_ERRINT	Level	385		1		3 to 2		15 to 8
		X2HPERI5_ERRINT	Level	386		2		5 to 4		23 to 16
		X2HPERI67_ERRINT	Level	387		3		7 to 6		31 to 24
		X2HDBGR_ERRINT	Level	388		4		9 to 8	97	7 to 0
		X2HBSC_ERRINT		389		5		11 to 10		15 to 8
		X2HSPI1_ERRINT		390		6		13 to 12		23 to 16
		X2HSPI2_ERRINT		391		7		15 to 14		31 to 24
		PRRI		392		8		17 to 16	98	7 to 0
Pixel format converter	0	IFEI0		393		9		19 to 18		15 to 8
		OFFI0		394		10		21 to 20		23 to 16
		PFVEI0		395		11		23 to 22		31 to 24
	1	IFEI1		396		12		25 to 24	99	7 to 0
		OFFI1		397		13		27 to 26		15 to 8
		PFVEI1		398		14		29 to 28		23 to 16
Reserved				399		15		31 to 30		31 to 24
				400		16	25	1 to 0	100	7 to 0
				401		17		3 to 2		15 to 8
				402		18		5 to 4		23 to 16
				403		19		7 to 6		31 to 24
				404		20		9 to 8	101	7 to 0
				405		21		11 to 10		15 to 8
				406		22		13 to 12		23 to 16
				407		23		15 to 14		31 to 24

Table 7.3 List of Interrupt IDs

Interrupt Source				Register Allocation							
Module	Channel	Request Source Name	Interrupt Request Edge/Level	Inter-rupt ID	ICDISRn ICDISERn ICDICERn ICDISPRn ICDICPRn ICDABRn	Bit	ICDICFRn	Bit	ICDIPRn ICDIPTRn	Bit	
Reserved				408	12	24	25	17 to 16	102	7 to 0	
				409		25		19 to 18		15 to 8	
				410		26		21 to 20		23 to 16	
				411		27		23 to 22		31 to 24	
				412		28		25 to 24		103	7 to 0
				413		29		27 to 26			15 to 8
				414		30		29 to 28		23 to 16	
				415		31		31 to 30		31 to 24	
Pin interrupts Note: The port name is given in the Channel column for the pin interrupts.	JP0_0	TINT0	Edge/Level	416	13	0	26	1 to 0	104	7 to 0	
	JP0_1	TINT1	Edge/Level	417		1		3 to 2		15 to 8	
	P0_0	TINT2	Edge/Level	418		2		5 to 4		23 to 16	
	P0_1	TINT3	Edge/Level	419		3		7 to 6		31 to 24	
	P0_2	TINT4	Edge/Level	420		4		9 to 8		105	7 to 0
	P0_3	TINT5	Edge/Level	421		5		11 to 10			15 to 8
	P0_4	TINT6	Edge/Level	422		6		13 to 12		23 to 16	
	P0_5	TINT7	Edge/Level	423		7		15 to 14		31 to 24	
	P1_0	TINT8	Edge/Level	424		8		17 to 16		106	7 to 0
	P1_1	TINT9	Edge/Level	425		9		19 to 18			15 to 8
	P1_2	TINT10	Edge/Level	426		10		21 to 20		23 to 16	
	P1_3	TINT11	Edge/Level	427		11		23 to 22		31 to 24	
	P1_4	TINT12	Edge/Level	428		13		12		25 to 24	107
	P1_5	TINT13	Edge/Level	429	13		27 to 26	15 to 8			
	P1_6	TINT14	Edge/Level	430	14		29 to 28	23 to 16			
	P1_7	TINT15	Edge/Level	431	15		31 to 30	31 to 24			
	P1_8	TINT16	Edge/Level	432	16		27	1 to 0	108	7 to 0	
	P1_9	TINT17	Edge/Level	433	17			3 to 2		15 to 8	
	P1_10	TINT18	Edge/Level	434	18		5 to 4	23 to 16			
	P1_11	TINT19	Edge/Level	435	19		7 to 6	31 to 24			
	P1_12	TINT20	Edge/Level	436	20		9 to 8	109	7 to 0		
	P1_13	TINT21	Edge/Level	437	21		11 to 10		15 to 8		
	P1_14	TINT22	Edge/Level	438	22		13 to 12	23 to 16			
	P1_15	TINT23	Edge/Level	439	23		15 to 14	31 to 24			
	P2_0	TINT24	Edge/Level	440	24		17 to 16	110	7 to 0		
	P2_1	TINT25	Edge/Level	441	25	19 to 18	15 to 8				
	P2_2	TINT26	Edge/Level	442	26	21 to 20	23 to 16				
P2_3	TINT27	Edge/Level	443	27	23 to 22	31 to 24					

Table 7.3 List of Interrupt IDs

Interrupt Source			Register Allocation										
Module	Channel	Request Source Name	Interrupt Request Edge/Level	Inter-rupt ID	ICDISRn		ICDISERn		ICDICERn		ICDISPRn		
					ICDICPRn	ICDABRn	Bit	ICDICFRn	Bit	ICDIPRn	ICDIPTRn	Bit	
Pin interrupts Note: The port name is given in the Channel column for the pin interrupts.	P2_4	TINT28	Edge/Level	444	13	28	27	25 to 24		111	7 to 0		
	P2_5	TINT29	Edge/Level	445		29		27 to 26			15 to 8		
	P2_6	TINT30	Edge/Level	446		30		29 to 28			23 to 16		
	P2_7	TINT31	Edge/Level	447	14	31	28	31 to 30		112	31 to 24		
	P2_8	TINT32	Edge/Level	448		0		1 to 0			7 to 0		
	P2_9	TINT33	Edge/Level	449		1		3 to 2			15 to 8		
	P2_10	TINT34	Edge/Level	450		2		5 to 4			23 to 16		
	P2_11	TINT35	Edge/Level	451		3		7 to 6			31 to 24		
	P2_12	TINT36	Edge/Level	452		4		9 to 8			113	7 to 0	
	P2_13	TINT37	Edge/Level	453		5		11 to 10				15 to 8	
	P2_14	TINT38	Edge/Level	454	6	13 to 12		23 to 16					
	P2_15	TINT39	Edge/Level	455	7	15 to 14		31 to 24					
	P3_0	TINT40	Edge/Level	456	8	17 to 16		114	7 to 0				
	P3_1	TINT41	Edge/Level	457	9	19 to 18			15 to 8				
	P3_2	TINT42	Edge/Level	458	10	21 to 20			23 to 16				
	P3_3	TINT43	Edge/Level	459	11	23 to 22		31 to 24					
	P3_4	TINT44	Edge/Level	460	12	25 to 24		115	7 to 0				
	P3_5	TINT45	Edge/Level	461	13	27 to 26			15 to 8				
	P3_6	TINT46	Edge/Level	462	14	29 to 28			23 to 16				
	P3_7	TINT47	Edge/Level	463	15	31 to 30		31 to 24					
	P3_8	TINT48	Edge/Level	464	16	29	1 to 0		116	7 to 0			
	P3_9	TINT49	Edge/Level	465	17		3 to 2			15 to 8			
	P3_10	TINT50	Edge/Level	466	18		5 to 4			23 to 16			
	P3_11	TINT51	Edge/Level	467	19		7 to 6			31 to 24			
	P3_12	TINT52	Edge/Level	468	20		9 to 8			117	7 to 0		
	P3_13	TINT53	Edge/Level	469	21		11 to 10				15 to 8		
	P3_14	TINT54	Edge/Level	470	22		13 to 12				23 to 16		
	P3_15	TINT55	Edge/Level	471	23	15 to 14		31 to 24					
	P4_0	TINT56	Edge/Level	472	24	17 to 16		118	7 to 0				
	P4_1	TINT57	Edge/Level	473	25	19 to 18			15 to 8				
	P4_2	TINT58	Edge/Level	474	26	21 to 20			23 to 16				
	P4_3	TINT59	Edge/Level	475	27	23 to 22		31 to 24					
P4_4	TINT60	Edge/Level	476	28	25 to 24		119	7 to 0					
P4_5	TINT61	Edge/Level	477	29	27 to 26			15 to 8					
P4_6	TINT62	Edge/Level	478	30	29 to 28			23 to 16					
P4_7	TINT63	Edge/Level	479	31	31 to 30		31 to 24						
P4_8	TINT64	Edge/Level	480	15	0	30	1 to 0		120	7 to 0			
P4_9	TINT65	Edge/Level	481		1		3 to 2			15 to 8			
P4_10	TINT66	Edge/Level	482		2		5 to 4			23 to 16			

Table 7.3 List of Interrupt IDs

Interrupt Source			Register Allocation							
Module	Channel	Request Source Name	Interrupt Request Edge/Level	Inter-rupt ID	ICDISRn ICDISERn ICDICERn ICDISPRn ICDICPRn ICDABRn	Bit	ICDICFRn	Bit	ICDIPRn ICDIPTRn	Bit
Pin interrupts Note: The port name is given in the Channel column for the pin interrupts.	P4_11	TINT67	Edge/Level	483	15	3	30	7 to 6	120	31 to 24
	P4_12	TINT68	Edge/Level	484		4		9 to 8	121	7 to 0
	P4_13	TINT69	Edge/Level	485		5		11 to 10		15 to 8
	P4_14	TINT70	Edge/Level	486		6		13 to 12		23 to 16
	P4_15	TINT71	Edge/Level	487		7		15 to 14		31 to 24
	P5_0	TINT72	Edge/Level	488		8		17 to 16	122	7 to 0
	P5_1	TINT73	Edge/Level	489		9		19 to 18		15 to 8
	P5_2	TINT74	Edge/Level	490		10		21 to 20		23 to 16
	P5_3	TINT75	Edge/Level	491		11		23 to 22		31 to 24
	P5_4	TINT76	Edge/Level	492		12		25 to 24	123	7 to 0
	P5_5	TINT77	Edge/Level	493		13		27 to 26		15 to 8
	P5_6	TINT78	Edge/Level	494		14		29 to 28		23 to 16
	P5_7	TINT79	Edge/Level	495		15		31 to 30		31 to 24
	P5_8	TINT80	Edge/Level	496		16	31	1 to 0	124	7 to 0
	P5_9	TINT81	Edge/Level	497		17		3 to 2		15 to 8
	P5_10	TINT82	Edge/Level	498		18		5 to 4		23 to 16
	P6_0	TINT83	Edge/Level	499		19		7 to 6		31 to 24
	P6_1	TINT84	Edge/Level	500		20		9 to 8	125	7 to 0
	P6_2	TINT85	Edge/Level	501		21		11 to 10		15 to 8
	P6_3	TINT86	Edge/Level	502		22		13 to 12		23 to 16
	P6_4	TINT87	Edge/Level	503		23		15 to 14		31 to 24
	P6_5	TINT88	Edge/Level	504		24		17 to 16	126	7 to 0
	P6_6	TINT89	Edge/Level	505		25		19 to 18		15 to 8
	P6_7	TINT90	Edge/Level	506		26		21 to 20		23 to 16
	P6_8	TINT91	Edge/Level	507		27		23 to 22		31 to 24
	P6_9	TINT92	Edge/Level	508		28		25 to 24	127	7 to 0
	P6_10	TINT93	Edge/Level	509		29		27 to 26		15 to 8
	P6_11	TINT94	Edge/Level	510		30		29 to 28		23 to 16
	P6_12	TINT95	Edge/Level	511		31		31 to 30		31 to 24
	P6_13	TINT96	Edge/Level	512	16	0	32	1 to 0	128	7 to 0
	P6_14	TINT97	Edge/Level	513		1		3 to 2		15 to 8
	P6_15	TINT98	Edge/Level	514		2		5 to 4		23 to 16
P7_0	TINT99	Edge/Level	515		3		7 to 6		31 to 24	
P7_1	TINT100	Edge/Level	516		4		9 to 8	129	7 to 0	
P7_2	TINT101	Edge/Level	517		5		11 to 10		15 to 8	
P7_3	TINT102	Edge/Level	518		6		13 to 12		23 to 16	
P7_4	TINT103	Edge/Level	519		7		15 to 14		31 to 24	

Table 7.3 List of Interrupt IDs

Interrupt Source			Register Allocation									
Module	Channel	Request Source Name	Interrupt Request Edge/Level	Inter-rupt ID	ICDISRn ICDISERn ICDICERn ICDISPRn ICDICPRn ICDABRn	Bit	ICDICFRn	Bit	ICDIPRn ICDIPTRn	Bit		
Pin interrupts Note: The port name is given in the Channel column for the pin interrupts.	P7_5	TINT104	Edge/Level	520	16	8	32	17 to 16	130	7 to 0		
	P7_6	TINT105	Edge/Level	521		9		19 to 18		15 to 8		
	P7_7	TINT106	Edge/Level	522		10		21 to 20		23 to 16		
	P7_8	TINT107	Edge/Level	523		11		23 to 22		31 to 24		
	P7_9	TINT108	Edge/Level	524		12		25 to 24		131	7 to 0	
	P7_10	TINT109	Edge/Level	525		13		27 to 26		15 to 8		
	P7_11	TINT110	Edge/Level	526		14		29 to 28		23 to 16		
	P7_12	TINT111	Edge/Level	527		15		31 to 30		31 to 24		
	P7_13	TINT112	Edge/Level	528		16		33		1 to 0	132	7 to 0
	P7_14	TINT113	Edge/Level	529		17				3 to 2		15 to 8
	P7_15	TINT114	Edge/Level	530		18				5 to 4		23 to 16
	P8_0	TINT115	Edge/Level	531		19		7 to 6		31 to 24		
	P8_1	TINT116	Edge/Level	532		20		9 to 8		133	7 to 0	
	P8_2	TINT117	Edge/Level	533		21		11 to 10			15 to 8	
	P8_3	TINT118	Edge/Level	534		22		13 to 12			23 to 16	
	P8_4	TINT119	Edge/Level	535	23	15 to 14	31 to 24					
	P8_5	TINT120	Edge/Level	536	24	17 to 16	134	7 to 0				
	P8_6	TINT121	Edge/Level	537	25	19 to 18		15 to 8				
	P8_7	TINT122	Edge/Level	538	26	21 to 20		23 to 16				
	P8_8	TINT123	Edge/Level	539	27	23 to 22	31 to 24					
	P8_9	TINT124	Edge/Level	540	28	25 to 24	135	7 to 0				
	P8_10	TINT125	Edge/Level	541	29	27 to 26		15 to 8				
	P8_11	TINT126	Edge/Level	542	30	29 to 28		23 to 16				
	P8_12	TINT127	Edge/Level	543	31	31 to 30	31 to 24					
	P8_13	TINT128	Edge/Level	544	17	0	34	1 to 0	136	7 to 0		
	P8_14	TINT129	Edge/Level	545		1		3 to 2		15 to 8		
	P8_15	TINT130	Edge/Level	546		2		5 to 4		23 to 16		
	P9_0	TINT131	Edge/Level	547		3		7 to 6		31 to 24		
	P9_1	TINT132	Edge/Level	548		4		9 to 8		137	7 to 0	
	P9_2	TINT133	Edge/Level	549		5		11 to 10			15 to 8	
	P9_3	TINT134	Edge/Level	550		6		13 to 12			23 to 16	
	P9_4	TINT135	Edge/Level	551		7		15 to 14		31 to 24		
	P9_5	TINT136	Edge/Level	552		8		17 to 16		138	7 to 0	
	P9_6	TINT137	Edge/Level	553		9		19 to 18			15 to 8	
	P9_7	TINT138	Edge/Level	554		10		21 to 20			23 to 16	
P10_0	TINT139	Edge/Level	555	11		23 to 22		31 to 24				

Table 7.3 List of Interrupt IDs

Interrupt Source			Register Allocation									
Module	Channel	Request Source Name	Interrupt Request Edge/Level	Inter-rupt ID	ICDISRn ICDISERn ICDICERn ICDISPRn ICDICPRn ICDABRn	Bit	ICDICFRn	Bit	ICDIPRn ICDIPTRn	Bit		
Pin interrupts Note: The port name is given in the Channel column for the pin interrupts.	P10_1	TINT140	Edge/Level	556	17	12	34	25 to 24	139	7 to 0		
	P10_2	TINT141	Edge/Level	557		13		27 to 26		15 to 8		
	P10_3	TINT142	Edge/Level	558		14		29 to 28		23 to 16		
	P10_4	TINT143	Edge/Level	559		15		31 to 30		31 to 24		
	P10_5	TINT144	Edge/Level	560		16		35		1 to 0	140	7 to 0
	P10_6	TINT145	Edge/Level	561		17				3 to 2		15 to 8
	P10_7	TINT146	Edge/Level	562		18				5 to 4		23 to 16
	P10_8	TINT147	Edge/Level	563		19		7 to 6		31 to 24		
	P10_9	TINT148	Edge/Level	564		20		9 to 8		141	7 to 0	
	P10_10	TINT149	Edge/Level	565		21		11 to 10			15 to 8	
	P10_11	TINT150	Edge/Level	566		22		13 to 12			23 to 16	
	P10_12	TINT151	Edge/Level	567		23		15 to 14		31 to 24		
	P10_13	TINT152	Edge/Level	568		24		17 to 16		142	7 to 0	
	P10_14	TINT153	Edge/Level	569		25		19 to 18			15 to 8	
	P10_15	TINT154	Edge/Level	570		26		21 to 20			23 to 16	
	P11_0	TINT155	Edge/Level	571	27	23 to 22	31 to 24					
	P11_1	TINT156	Edge/Level	572	28	25 to 24	143	7 to 0				
	P11_2	TINT157	Edge/Level	573	29	27 to 26		15 to 8				
	P11_3	TINT158	Edge/Level	574	30	29 to 28		23 to 16				
	P11_4	TINT159	Edge/Level	575	31	31 to 30	31 to 24					
	P11_5	TINT160	Edge/Level	576	18	0	36	1 to 0	144	7 to 0		
	P11_6	TINT161	Edge/Level	577		1		3 to 2		15 to 8		
	P11_7	TINT162	Edge/Level	578		2		5 to 4		23 to 16		
	P11_8	TINT163	Edge/Level	579		3		7 to 6		31 to 24		
	P11_9	TINT164	Edge/Level	580		4		9 to 8		145	7 to 0	
	P11_10	TINT165	Edge/Level	581		5		11 to 10			15 to 8	
	P11_11	TINT166	Edge/Level	582		6		13 to 12			23 to 16	
	P11_12	TINT167	Edge/Level	583		7		15 to 14		31 to 24		
	P11_13	TINT168	Edge/Level	584		8		17 to 16		146	7 to 0	
	P11_14	TINT169	Edge/Level	585		9		19 to 18			15 to 8	
P11_15	TINT170	Edge/Level	586	10		21 to 20		23 to 16				

## 7.6 Operation

### 7.6.1 Initial Settings

For details on the registers for making initial settings and the procedures for settings in general, see the specification for the GCI interrupt architecture. Figure 7.2 illustrates the flow of initial settings.

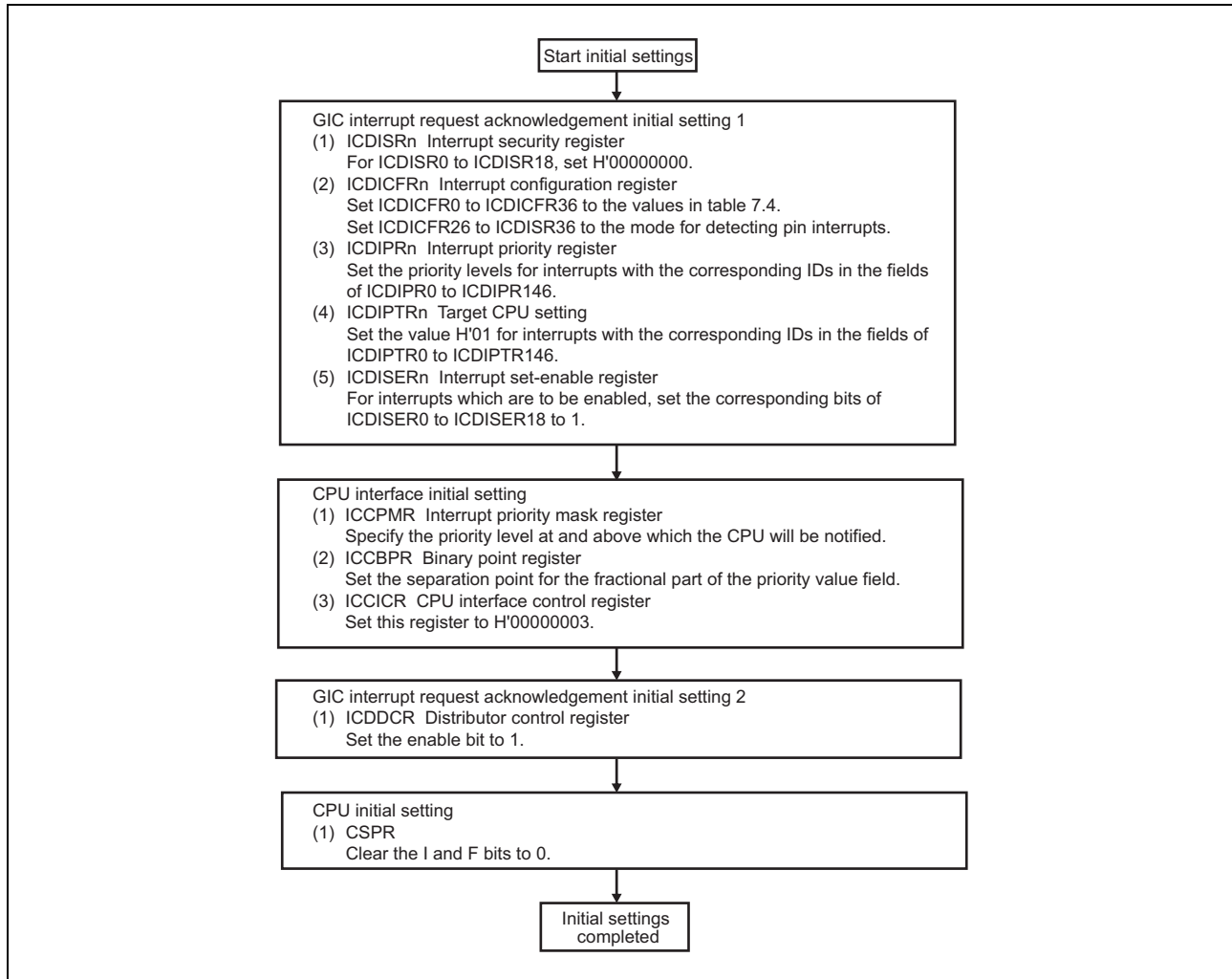


Figure 7.2 Flow of Initial Settings



**Table 7.4 CDICFRn Interrupt Configuration Register Settings**

Register Name	Setting	Interrupt ID
ICDICFR0	H'AAAAAAAA	15 to 0
ICDICFR1	H'00000055	31 to 16
ICDICFR2	H'FFFD5555	47 to 32
ICDICFR3	H'555FFFFF	63 to 48
ICDICFR4	H'55555555	79 to 64
ICDICFR5	H'55555555	95 to 80
ICDICFR6	H'55555555	111 to 96
ICDICFR7	H'55555555	127 to 112
ICDICFR8	H'5555F555	143 to 128
ICDICFR9	H'55555555	159 to 144
ICDICFR10	H'55555555	175 to 160
ICDICFR11	H'F5555555	191 to 176
ICDICFR12	H'F555F555	207 to 192
ICDICFR13	H'5555F555	223 to 208
ICDICFR14	H'55555555	239 to 224
ICDICFR15	H'55555555	255 to 240
ICDICFR16	H'55555555	271 to 256
ICDICFR17	H'FD555555	287 to 272
ICDICFR18	H'55555557	303 to 288
ICDICFR19	H'55555555	319 to 304
ICDICFR20	H'FFD55555	335 to 320
ICDICFR21	H'5F55557F	351 to 336
ICDICFR22	H'FD55555F	367 to 352
ICDICFR23	H'55555557	383 to 368
ICDICFR24	H'55555555	399 to 384
ICDICFR25	H'55555555	415 to 400
ICDICFR26*1	H'55555555	431 to 416
ICDICFR27*1	H'55555555	447 to 432
ICDICFR28*1	H'55555555	463 to 448
ICDICFR29*1	H'55555555	479 to 464
ICDICFR30*1	H'55555555	495 to 480
ICDICFR31*1	H'55555555	511 to 496
ICDICFR32*1	H'55555555	527 to 512
ICDICFR33*1	H'55555555	543 to 528
ICDICFR34*1	H'55555555	559 to 544
ICDICFR35*1	H'55555555	575 to 560
ICDICFR36*1	H'55555555	586 to 576

Note 1. Edge or level detection can be selected for IDs corresponding to pin interrupts. The settings in the above table select level detection.

### 7.6.2 Flow of Interrupt Operations

For details on operation involved in interrupt generation, see the GIC architecture specification.

Figure 7.3 shows the flow of interrupt operations.

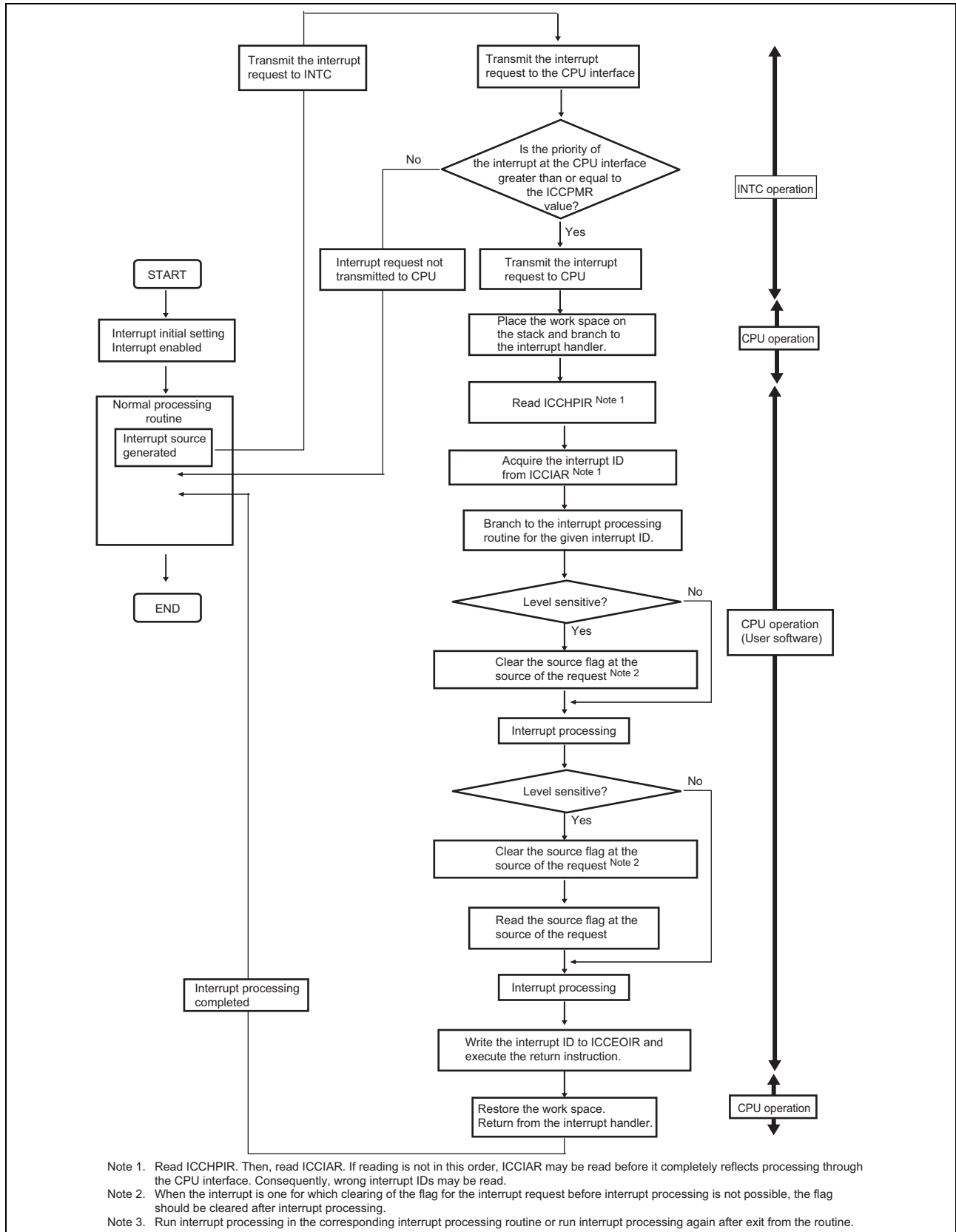


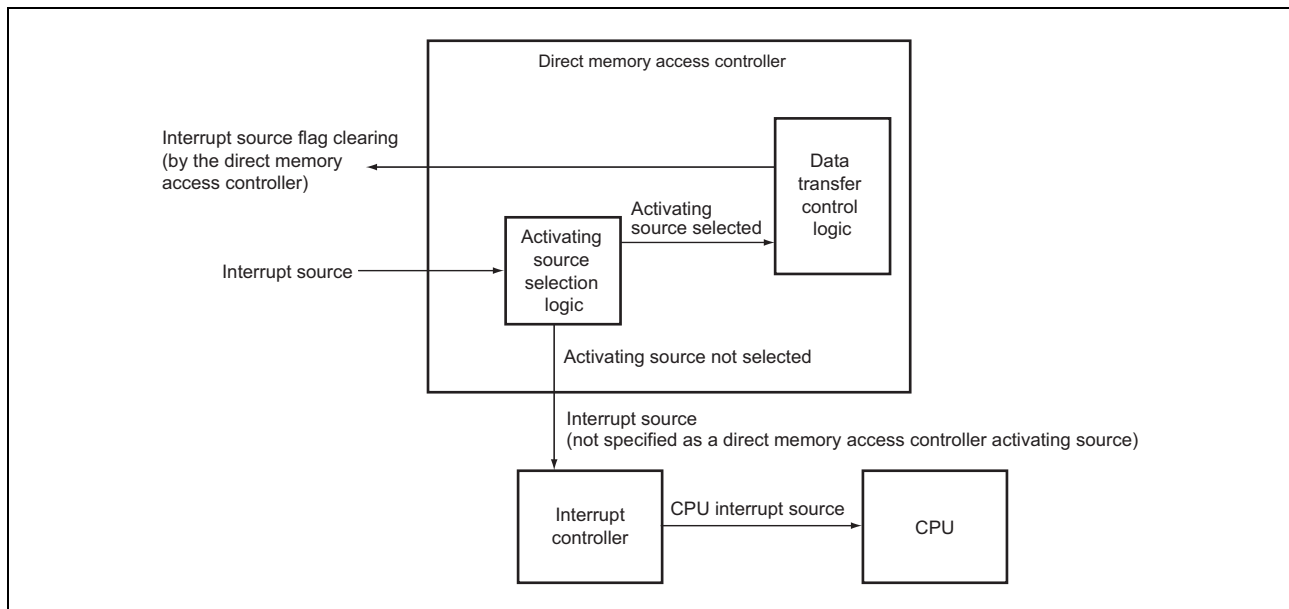
Figure 7.3 Flow of Interrupt Operations 1

## 7.7 Data Transfer with Interrupt Request Signals

Interrupt request signals can be used to activate the direct memory access controller and transfer data.

Interrupt sources for which the direct memory access controller is designated as the destination by DMA extension resource selectors 0 to 7 are masked and requests from them are not input to the interrupt controller.

Figure 7.4 shows a block diagram of interrupt control. For details, see section 9, Direct Memory Access Controller.



**Figure 7.4** Interrupt Control Block Diagram

### 7.7.1 Handling Interrupt Request Signals as Sources for CPU Interrupt but Not Direct Memory Access Controller Activating

1. Do not select direct memory access controller activating sources.
2. When interrupts occur, interrupt requests are sent to the CPU.
3. The CPU clears the interrupt source and performs the necessary processing in the interrupt exception service routine.

### 7.7.2 Handling Interrupt Request Signals as Sources for Activating Direct Memory Access Controller but Not CPU Interrupt

1. Select direct memory access controller activating sources.
2. Activating sources are applied to the direct memory access controller when interrupts occur.
3. The direct memory access controller clears the interrupt sources when starting transfer.

## 7.8 Usage Note

### 7.8.1 Timing to Clear an Interrupt Source

The interrupt source flags should be cleared in the interrupt exception service routine. After clearing the interrupt source flag, time is required before the interrupt source sent to the CPU is actually cancelled. To ensure that an interrupt request that should have been cleared is not inadvertently accepted again, read the interrupt source flag after it has been cleared, and then execute an RFE instruction.

### 7.8.2 Notes on Selecting IRQ Interrupt Pin Functions

While the interrupt control register 1 (ICR1) is set so that an interrupt request is detected on the falling edge of an IRQn input and the current input on the pin is at the low level, the relevant edge will be detected when the pin function is switched to the IRQ interrupt function.

### 7.8.3 Notes on Reading Interrupt ID Values from ICCIAR

If an interrupt ID value read from ICCIAR is 0, the interrupt notice may be wrong. At that time, confirm the interrupt state before proceeding with interrupt processing.

When the interrupt ID is read as 0, the interrupt state can be confirmed by using bit 0 in ICDABR0.

If the interrupt state is inactive, the interrupt notice is wrong and no interrupt processing is required. Return from the interrupt processing after writing the same value as the setting value to ICDIPR0.

If the interrupt state is active, the interrupt notice is correct. Proceed with interrupt processing.

## 8. Bus State Controller

The bus state controller outputs control signals for various types of memory and external devices that are connected to the external address space. The functions of this module enable this LSI to connect directly with SRAM, SDRAM, and other memory storage devices, and external devices.

### 8.1 Features

1. External address space
  - A maximum of 64 Mbytes for each of areas CS0 to CS5.
  - Can specify the normal space interface, SRAM interface with byte selection, burst ROM (clocked synchronous or asynchronous), MPX-I/O, and SDRAM memory type for each address space.
  - Can select the data bus width (8, 16, or 32 bits) for each of address spaces.
  - Controls insertion of wait cycles for each address space.
  - Controls insertion of wait cycles for each read access and write access.
  - Can set independent idle cycles during the continuous access for five cases: read-write (in same space/different spaces), read-read (in same space/different spaces), the first cycle is a write access.
2. Normal space interface
  - Supports the interface that can directly connect to the SRAM.
3. Burst ROM interface (clocked asynchronous)
  - High-speed access to the ROM that has the page mode function.
4. MPX-I/O interface
  - Can directly connect to a peripheral LSI that needs an address/data multiplexing.
5. SDRAM interface
  - Can set the SDRAM in up to two areas.
  - Multiplex output for row address/column address.
  - Efficient access by single read/single write.
  - High-speed access in bank-active mode.
  - Supports an auto-refresh and self-refresh.
  - Supports a power-down mode.
  - Issues MRS and EMRS commands.
6. SRAM interface with byte selection
  - Can connect directly to a SRAM with byte selection.
7. Burst ROM interface (clocked synchronous)
  - Can connect directly to a burst ROM of the clocked synchronous type.
8. Refresh function
  - Supports the auto-refresh and self-refresh functions.
  - Specifies the refresh interval using the refresh counter and clock selection.
  - Can execute concentrated refresh by specifying the refresh counts (1, 2, 4, 6, or 8).
9. Usage as interval timer for refresh counter
  - Generates an interrupt request at compare match.
10. Detection of long wait state for access by the signal on the external WAIT pin.
  - A timeout detection condition is specifiable per CS space.
  - Once timeout is detected, the external WAIT function is disabled and a timeout detection interrupt request is issued.

Figure 8.1 shows a block diagram of this module.

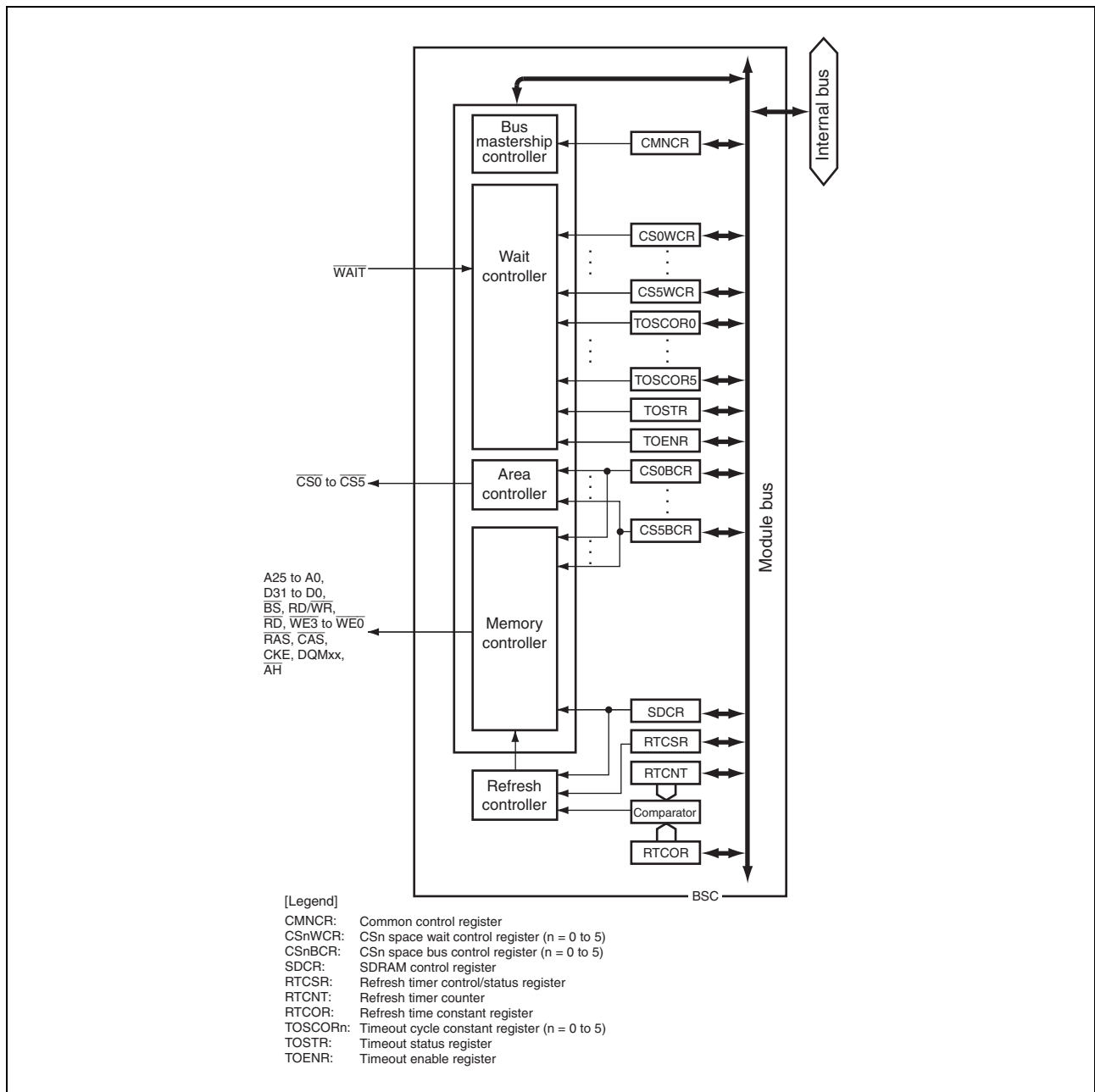


Figure 8.1 Block Diagram of Bus State Controller

## 8.2 Input/Output Pins

Table 8.1 shows the pin configuration.

**Table 8.1 Pin Configuration**

Name	I/O	Function
A25 to A0	Output	Address bus
D31 to D0	I/O	Data bus
BS	Output	Bus cycle start
$\overline{CS0}$ to $\overline{CS5}$	Output	Chip select
$\overline{RD}/\overline{WR}$	Output	Read/write Connects to $\overline{WE}$ pins when SDRAM or SRAM with byte selection is connected.
$\overline{RD}$	Output	Read pulse signal (read data output enable signal)
$\overline{WE3}/\overline{DQMUU}/\overline{AH}$	Output	Indicates that D31 to D24 are being written to. Connected to the byte select signal when a SRAM with byte selection is connected. Functions as the select signals for D31 to D24 when SDRAM is connected. Functions as the address hold signal when the MPX-I/O is used.
$\overline{WE2}/\overline{DQMUL}$	Output	Indicates that D23 to D16 are being written to. Connected to the byte select signal when a SRAM with byte selection is connected. Functions as the select signals for D23 to D16 when SDRAM is connected.
$\overline{WE1}/\overline{DQMLU}$	Output	Indicates that D15 to D8 are being written to. Connected to the byte select signal when a SRAM with byte selection is connected. Functions as the select signals for D15 to D8 when SDRAM is connected.
$\overline{WE0}/\overline{DQMLL}$	Output	Indicates that D7 to D0 are being written to. Connected to the byte select signal when a SRAM with byte selection is connected. Functions as the select signals for D7 to D0 when SDRAM is connected.
$\overline{RAS}$	Output	Connects to $\overline{RAS}$ pin when SDRAM is connected.
$\overline{CAS}$	Output	Connects to $\overline{CAS}$ pin when SDRAM is connected.
$\overline{CKE}$	Output	Connects to $\overline{CKE}$ pin when SDRAM is connected.
$\overline{WAIT}$	Input	External wait input

## 8.3 Area Overview

### 8.3.1 Address Map

In the architecture, this LSI has a 32-bit address space, which is divided into external memory spaces (SPI multi I/O bus space, large-capacity on-chip RAM, hold on-chip RAM, on-chip peripheral modules, and reserved areas) according to the upper bits of the address.

See section 5, LSI Internal Bus (including Secondary Cache) for how to enable or disable caching for the CS0 to CS5 external address spaces.

The kind of memory to be connected and the data bus width are specified in each partial space. The address map for the external address space is listed below.

**Table 8.2 Address Map**

Internal Address	Space	Memory to be Connected
H'00000000 to H'03FFFFFF	CS0	Normal space, SRAM with byte selection, burst ROM (asynchronous or synchronous)
H'04000000 to H'07FFFFFF	CS1	Normal space, SRAM with byte selection
H'08000000 to H'0BFFFFFF	CS2	Normal space, SRAM with byte selection, SDRAM
H'0C000000 to H'0FFFFFFF	CS3	Normal space, SRAM with byte selection, SDRAM
H'10000000 to H'13FFFFFF	CS4	Normal space, SRAM with byte selection, burst ROM (asynchronous)
H'14000000 to H'17FFFFFF	CS5	Normal space, SRAM with byte selection, MPX-I/O
H'18000000 to H'3FFFFFFF	Other	SPI multi I/O bus space, large-capacity on-chip RAM, hold on-chip RAM, on-chip peripheral modules, reserved area*1
H'40000000 to H'43FFFFFF	CS0 mirror	Normal space, SRAM with byte selection, burst ROM (asynchronous or synchronous)
H'44000000 to H'47FFFFFF	CS1 mirror	Normal space, SRAM with byte selection
H'48000000 to H'4BFFFFFF	CS2 mirror	Normal space, SRAM with byte selection, SDRAM
H'4C000000 to H'4FFFFFFF	CS3 mirror	Normal space, SRAM with byte selection, SDRAM
H'50000000 to H'53FFFFFF	CS4 mirror	Normal space, SRAM with byte selection, burst ROM (asynchronous)
H'54000000 to H'57FFFFFF	CS5 mirror	Normal space, SRAM with byte selection, MPX-I/O
H'58000000 to H'FFFFFFF	Other	SPI multi I/O bus space (mirror), large-capacity on-chip RAM (mirror), hold on-chip RAM (mirror), on-chip peripheral modules, reserved area*1

Note 1. For the large-capacity on-chip RAM space and hold on-chip RAM space, access the addresses shown in section 53, On-Chip RAM. For the on-chip peripheral module space, access the addresses shown in section 57, List of Registers. Do not access addresses which are not described in these sections. Otherwise, the correct operation cannot be guaranteed.



### 8.3.2 Data Bus Width and Related Pin Setting for Each Area Depending on Boot Mode

The initial state of data bus width and settings of the pins related to this module depends on boot mode. For boot mode, refer to section 3, Boot Mode.

In boot modes 0 and 1, the state of area 0 is fixed to the state with bus width of 16 or 32 bits, because this LSI is started up by the program stored in the ROM connected to area 0. The initial states of areas 1 to 5 are the same as that of area 0, but the bus width can be changed by the program. Immediately after a power-on reset in these modes, some of the address and data-bus signals and the  $\overline{CS0}$  and  $\overline{RD}$  signals are automatically selected by default as the functions of the corresponding pins, since these signals are required to read ROM data from area 0. With the exception of these pins, the general purpose pin function is selected by default, and other required pin functions must be specified by the program. Read access to area 0 is only permitted before the pin settings are completed.

In boot modes 3 to 5, the state of areas 0 to 5 can be changed from the initial state by the program, because the LSI is started by the program stored in the SPI serial memory, the NAND flash memory with the SD controller, or the NAND flash memory with the MMC controller. Since pin functions related to this module are not set automatically, they need to be set by the program. Do not access external address spaces before the pin settings are completed.

Table 8.3 shows the initial state by areas 0 to 5 in boot modes 0, 1, and 2 to 5.

The sample access waveforms shown in this section include the pins such as  $\overline{BS}$ ,  $\overline{RD}/\overline{WR}$ , and  $\overline{WEn}$ . They are the waveforms when pin functions are assigned to the general I/O ports. For example, when 16-bit bus width is used in boot mode 1, setting for pin A1 is needed. When 8-bit bus width is used, setting for pins A1 and A0 is also needed.

For details on pin function settings, see section 54, Ports.

**Table 8.3 Initial States by Areas in Boot Modes 0, 1, and 2 to 5**

Boot Mode	Item	Area 0	Areas 1 to 5
0	Data bus width	Fixed to 16 bits. Not changeable.	16 bits. Can be changed by program.
	Settings of pins related to this module	Pins A20 to A1, D15 to D0, $\overline{CS0}$ , and $\overline{RD}$ are set automatically. Other pins need to be set by program.	
1	Data bus width	Fixed to 32 bits. Not changeable.	32 bits. Can be changed by program.
	Settings of pins related to this module	Pins A20 to A2, D31 to D0, $\overline{CS0}$ , and $\overline{RD}$ are set automatically. Other pins need to be set by program.	
3 to 5	Data bus width	32 bits. Can be changed by program.	
	Settings of pins related to this module	General I/O function. For external bus access, all the necessary pins need to be set by program.	

Note 1. In boot mode 0 or 1, if a boot ROM that uses higher-order address lines than A21 is connected, the circuit board must include pull-down resistors for those address lines.

Note 2. The data-bus width may be limited by the type of memory in use. For details, see section 8.4.2, CSn Space Bus Control Register (CSnBCR) (n = 0 to 5).

## 8.4 Register Descriptions

Table 8.4 shows the register configuration of this module.

Do not access the areas until settings of the connected memory interface are completed.

**Table 8.4 Register Configuration**

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Common control register	CMNCR	R/W	H'00001018	H'3FFFC000	32
CS0 space bus control register	CS0BCR	R/W	H'36DB0C00*1	H'3FFFC004	32
CS1 space bus control register	CS1BCR	R/W	H'36DB0C00*1	H'3FFFC008	32
CS2 space bus control register	CS2BCR	R/W	H'36DB0C00*1	H'3FFFC00C	32
CS3 space bus control register	CS3BCR	R/W	H'36DB0C00*1	H'3FFFC010	32
CS4 space bus control register	CS4BCR	R/W	H'36DB0C00*1	H'3FFFC014	32
CS5 space bus control register	CS5BCR	R/W	H'36DB0C00*1	H'3FFFC018	32
CS0 space wait control register	CS0WCR	R/W	H'00000500	H'3FFFC028	32
CS1 space wait control register	CS1WCR	R/W	H'00000500	H'3FFFC02C	32
CS2 space wait control register	CS2WCR	R/W	H'00000500	H'3FFFC030	32
CS3 space wait control register	CS3WCR	R/W	H'00000500	H'3FFFC034	32
CS4 space wait control register	CS4WCR	R/W	H'00000500	H'3FFFC038	32
CS5 space wait control register	CS5WCR	R/W	H'00000500	H'3FFFC03C	32
SDRAM control register	SDCR	R/W	H'00000000	H'3FFFC04C	32
Refresh timer control/status register	RTCSR	R/W	H'00000000	H'3FFFC050	32
Refresh timer counter	RTCNT	R/W	H'00000000	H'3FFFC054	32
Refresh time constant register	RTCOR	R/W	H'00000000	H'3FFFC058	32
Timeout cycle constant register 0	TOSCOR0	R/W	H'00000000	H'3FFFC060	32
Timeout cycle constant register 1	TOSCOR1	R/W	H'00000000	H'3FFFC064	32
Timeout cycle constant register 2	TOSCOR2	R/W	H'00000000	H'3FFFC068	32
Timeout cycle constant register 3	TOSCOR3	R/W	H'00000000	H'3FFFC06C	32
Timeout cycle constant register 4	TOSCOR4	R/W	H'00000000	H'3FFFC070	32
Timeout cycle constant register 5	TOSCOR5	R/W	H'00000000	H'3FFFC074	32
Timeout status register	TOSTR	R/W	H'00000000	H'3FFFC080	32
Timeout enable register	TOENR	R/W	H'00000000	H'3FFFC084	32

Note 1. H'36DB0C00 in boot mode 0; H'36DB0E00 in boot modes 1 and 3 to 5

### 8.4.1 Common Control Register (CMNCR)

CMNCR is a 32-bit register that controls the common items for each area.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	TL0	-	-	-	AL0	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	DPRTY[1:0]		-	-	-	-	-	-	-	HIZ MEM	HIZ CNT*
Initial value:	0	0	0	1	0	0	0	0	0	0	0	1	1	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	TL0	0	RW	Transfer End Level Specifies the TEND0 signal output is high active or low active. 0: Low-active output from TEND0 1: High-active output from TEND0
27 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	AL0	0	RW	Specifies the DACK0 (acknowledge) signal output is high active or low active. 0: Low-active output from DACK0 1: High-active output from DACK0
23 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10, 9	DPRTY[1:0]	00	R/W	DMA Burst Transfer Priority Specify the priority for a refresh request during DMA burst transfer. 0*: Accepts a refresh request during DMA burst transfer. 10: Does not accept a refresh request during DMA burst transfer. 11: Reserved (setting prohibited)
8 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4, 3	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
2	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
1	HIZMEM	0	R/W	High-Z Memory Control Specifies the pin state in software standby mode or deep standby mode for A25 to A0, $\overline{BS}$ , $\overline{CSn}$ , $\overline{RD}/\overline{WR}$ , $\overline{WEn}/\overline{DQMxx}/\overline{AH}$ , and $\overline{RD}$ . 0: High impedance in software standby mode or deep standby mode. 1: Driven in software standby mode or deep standby mode
0	HIZCNT*	0	R/W	High-Z Control Specifies the state in software standby mode or deep standby mode for CKE, $\overline{RAS}$ , and $\overline{CAS}$ . 0: High impedance in software standby mode or deep standby mode for CKE, $\overline{RAS}$ , and $\overline{CAS}$ . 1: Driven in software standby mode or deep standby mode for CKE, $\overline{RAS}$ , and $\overline{CAS}$ .

Note: \* For High-Z control of CKIO, see section 6, Clock Pulse Generator.

## 8.4.2 CSn Space Bus Control Register (CSnBCR) (n = 0 to 5)

CSnBCR is a 32-bit readable/writable register that specifies the memory connected to each space, the number of idle cycles between bus cycles, and the bus width.

Do not access external memory for the corresponding area until CSnBCR initial setting and pin setting are completed. Idle cycles may be inserted even when they are not specified. For details, see section 8.5.10, Wait between Access Cycles.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	IWW[2:0]			IWRWD[2:0]			IWRWS[2:0]			IWRRD[2:0]			IWRRS[2:0]		
Initial value:	0	0	1	1	0	1	1	0	1	1	0	1	1	0	1	1
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	TYPE[2:0]			-	BSZ[1:0]		-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	1	1*	0*	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R	R	R	R	R	R	R	R	R

Note: \* B'10 in boot mode 0; B'11 in boot mode 1 and 3 to 5.

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30 to 28	IWW[2:0]	011	R/W	Idle Cycles between Write-Read Cycles and Write-Write Cycles These bits specify the number of idle cycles to be inserted after the access to a memory that is connected to the space. The target access cycles are the write-read cycle and write-write cycle. 000: No idle cycle inserted 001: 1 idle cycle inserted 010: 2 idle cycles inserted 011: 4 idle cycles inserted 100: 6 idle cycles inserted 101: 8 idle cycles inserted 110: 10 idle cycles inserted 111: 12 idle cycles inserted
27 to 25	IWRWD[2:0]	011	R/W	Idle Cycles for Another Space Read-Write Specify the number of idle cycles to be inserted after the access to a memory that is connected to the space. The target access cycle is a read-write one in which continuous access cycles switch between different spaces. 000: No idle cycle inserted 001: 1 idle cycle inserted 010: 2 idle cycles inserted 011: 4 idle cycles inserted 100: 6 idle cycles inserted 101: 8 idle cycles inserted 110: 10 idle cycles inserted 111: 12 idle cycles inserted
24 to 22	IWRWS[2:0]	011	R/W	Idle Cycles for Read-Write in the Same Space Specify the number of idle cycles to be inserted after the access to a memory that is connected to the space. The target cycle is a read-write cycle of which continuous access cycles are for the same space. 000: No idle cycle inserted 001: 1 idle cycle inserted 010: 2 idle cycles inserted 011: 4 idle cycles inserted 100: 6 idle cycles inserted 101: 8 idle cycles inserted 110: 10 idle cycles inserted 111: 12 idle cycles inserted

Bit	Bit Name	Initial Value	R/W	Description
21 to 19	IWRRD[2:0]	011	R/W	<p>Idle Cycles for Read-Read in Another Space</p> <p>Specify the number of idle cycles to be inserted after the access to a memory that is connected to the space. The target cycle is a read-read cycle of which continuous access cycles switch between different space.</p> <p>000: No idle cycle inserted            001: 1 idle cycle inserted            010: 2 idle cycles inserted            011: 4 idle cycles inserted            100: 6 idle cycles inserted            101: 8 idle cycles inserted            110: 10 idle cycles inserted            111: 12 idle cycles inserted</p>
18 to 16	IWRRS[2:0]	011	R/W	<p>Idle Cycles for Read-Read in the Same Space</p> <p>Specify the number of idle cycles to be inserted after the access to a memory that is connected to the space. The target cycle is a read-read cycle of which continuous access cycles are for the same space.</p> <p>000: No idle cycle inserted            001: 1 idle cycle inserted            010: 2 idle cycles inserted            011: 4 idle cycles inserted            100: 6 idle cycles inserted            101: 8 idle cycles inserted            110: 10 idle cycles inserted            111: 12 idle cycles inserted</p>
15	—	0	R	<p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p>
14 to 12	TYPE[2:0]	000	R/W	<p>Specify the type of memory connected to a space.</p> <p>000: Normal space            001: Burst ROM (clock asynchronous)            010: MPX-I/O            011: SRAM with byte selection            100: SDRAM            101: Reserved (setting prohibited)            110: Reserved (setting prohibited)            111: Burst ROM (clock synchronous)</p> <p>For details for memory type in each area, see Table 8.2.</p> <p>Note: When connecting the burst ROM to the CS0 space in boot modes 0 and 1, change the CS0WCR register to the settings by the burst ROM CS0WCR uses and then set TYPE[2:0] to the burst ROM setting. In boot modes 2 to 5, memory access should be performed after setting CS0BCR and CS0WCR.</p>
11	—	1	R	<p>Reserved</p> <p>This bit is always read as 1. The write value should always be 1.</p>
10, 9	BSZ[1:0]	10*	R/W	<p>Data Bus Width Specification</p> <p>Specify the data bus widths of spaces.</p> <p>00: Reserved (setting prohibited)            01: 8-bit size            10: 16-bit size            11: 32-bit size</p> <p>For MPX-I/O, selects bus width by address</p> <p>Notes: 1. If area 5 is specified as MPX-I/O, the bus width can be specified as 8 bits or 16 bits by the address according to the SZSEL bit in CS5WCR by specifying the BSZ[1:0] bits to 11. The fixed bus width can be specified as 8 bits or 16 bits            2. In boot modes 0 and 1, the BSZ[1:0] bits settings in CS0BCR are ignored.            3. If area 2 or area 3 is specified as SDRAM space, the bus width can be specified as either 16 bits or 32 bits.            4. If area 0 is specified as clocked synchronous burst ROM space, the bus width can be specified as either 16 bits or 32 bits.</p>
8 to 0	—	All 0	R	<p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p>

Note: \* B'10 in boot mode 0; B'11 in boot modes 1 and 3 to 5.

### 8.4.3 CSn Space Wait Control Register (CSnWCR) (n = 0 to 5)

CSnWCR specifies various wait cycles for memory access. The bit configuration of this register varies as shown below according to the memory type (TYPE2 to TYPE0) specified by the CSn space bus control register (CSnBCR). Specify CSnWCR before accessing the target area. Specify CSnBCR first, then specify CSnWCR.

#### (1) Normal Space, SRAM with Byte Selection, and MPX-I/O

- CS0WCR

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	—*	BAS	-	-	—*	—*
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	SW[1:0]	WR[3:0]			WM	-	-	-	-	-	-	HW[1:0]	-
Initial value:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
21	—*	0	R/W	Reserved Set this bit to 0 when the interfaces for normal space or for SRAM with byte selection are used.
20	BAS	0	R/W	SRAM with Byte Selection Byte Access Select Specifies the $\overline{WEn}$ and $\overline{RD}/\overline{WR}$ signal timing when the SRAM interface with byte selection is used. 0: Asserts the $\overline{WEn}$ signal at the read/write timing and asserts the $\overline{RD}/\overline{WR}$ signal during the write access cycle. 1: Asserts the $\overline{WEn}$ signal during the read/write access cycle and asserts the $\overline{RD}/\overline{WR}$ signal at the write timing.
19, 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17, 16	—*	All 0	R/W	Reserved Set these bits to 0 when the interfaces for normal space or for SRAM with byte selection are used.
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12, 11	SW[1:0]	00	R/W	Number of Delay Cycles from Address, $\overline{CS0}$ Assertion to $\overline{RD}$ , $\overline{WEn}$ Assertion Specify the number of delay cycles from address and $\overline{CS0}$ assertion to $\overline{RD}$ and $\overline{WEn}$ assertion. 00: 0.5 cycles 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles

Bit	Bit Name	Initial Value	R/W	Description
10 to 7	WR[3:0]	1010	R/W	Number of Access Wait Cycles Specify the number of cycles that are necessary for read/write access. 0000: No cycle 0001: 1 cycle 0010: 2 cycles 0011: 3 cycles 0100: 4 cycles 0101: 5 cycles 0110: 6 cycles 0111: 8 cycles 1000: 10 cycles 1001: 12 cycles 1010: 14 cycles 1011: 18 cycles 1100: 24 cycles 1101: Reserved (setting prohibited) 1110: Reserved (setting prohibited) 1111: Reserved (setting prohibited)
6	WM	0	R/W	External Wait Mask Specification Specifies whether or not the external wait input is valid. The specification by this bit is valid even when the number of access wait cycle is 0. 0: External wait input is valid 1: External wait input is ignored
5 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	HW[1:0]	00	R/W	Delay Cycles from $\overline{RD}$ , $\overline{WEn}$ Negation to $\overline{Address}$ , $\overline{CS0}$ Negation Specify the number of delay cycles from $\overline{RD}$ and $\overline{WEn}$ negation to address and $\overline{CS0}$ negation. 00: 0.5 cycles 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles

Note: \* In boot modes 0 and 1, to connect the burst ROM to the CS0 space and switch to burst ROM interface after activation, set the TYPE[2:0] bits in CS0BCR after setting the burst number by the bits 20 and 21 and the burst wait cycle number by the bits 16 and 17. Do not write 1 to the reserved bits other than above bits.

- CS1WCR

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	BAS	-	WW[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	SW[1:0]		WR[3:0]				WM	-	-	-	-	HW[1:0]	
Initial value:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	BAS	0	R/W	SRAM with <u>Byte Selection</u> <u>Byte Access Select</u> Specifies the $\overline{WEn}$ and $\overline{RD}/\overline{WR}$ signal timing when the SRAM interface with byte selection is used. 0: Asserts the $\overline{WEn}$ signal at the read/write timing and asserts the $\overline{RD}/\overline{WR}$ signal during the write access cycle. 1: Asserts the $\overline{WEn}$ signal during the read/write access cycle and asserts the $\overline{RD}/\overline{WR}$ signal at the write timing.
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
18 to 16	WW[2:0]	000	R/W	Number of Write Access Wait Cycles Specify the number of cycles that are necessary for write access. 000: The same cycles as WR[3:0] setting (number of read access wait cycles) 001: No cycle 010: 1 cycle 011: 2 cycles 100: 3 cycles 101: 4 cycles 110: 5 cycles 111: 6 cycles
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12, 11	SW[1:0]	00	R/W	Number of Delay Cycles from Address, $\overline{CSn}$ Assertion to $\overline{RD}$ , $\overline{WEn}$ Assertion Specify the number of delay cycles from address and $\overline{CSn}$ assertion to $\overline{RD}$ and $\overline{WEn}$ assertion. 00: 0.5 cycles 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles



Bit	Bit Name	Initial Value	R/W	Description
10 to 7	WR[3:0]	1010	R/W	Number of Read Access Wait Cycles Specify the number of cycles that are necessary for read access. 0000: No cycle 0001: 1 cycle 0010: 2 cycles 0011: 3 cycles 0100: 4 cycles 0101: 5 cycles 0110: 6 cycles 0111: 8 cycles 1000: 10 cycles 1001: 12 cycles 1010: 14 cycles 1011: 18 cycles 1100: 24 cycles 1101: Reserved (setting prohibited) 1110: Reserved (setting prohibited) 1111: Reserved (setting prohibited)
6	WM	0	R/W	External Wait Mask Specification Specifies whether or not the external wait input is valid. The specification by this bit is valid even when the number of access wait cycle is 0. 0: External wait input is valid 1: External wait input is ignored
5 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	HW[1:0]	00	R/W	Delay Cycles from RD, WEn Negation to Address, CSn Negation Specify the number of delay cycles from RD and WEn negation to address and CSn negation. 00: 0.5 cycles 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles

- CS2WCR, CS3WCR

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	BAS	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	WR[3:0]				WM	-	-	-	-	-	-
Initial value:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	BAS	0	R/W	SRAM with <u>Byte Selection</u> <u>Byte Access Select</u> Specifies the $\overline{WEn}$ and RD/WR signal timing when the SRAM interface with <u>byte selection</u> is used. 0: Asserts the $\overline{WEn}$ signal at the read timing and asserts the RD/WR signal during the write access cycle. 1: Asserts the $\overline{WEn}$ signal during the read access cycle and asserts the RD/WR signal at the write timing.
19 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 7	WR[3:0]	1010	R/W	Number of Access Wait Cycles Specify the number of cycles that are necessary for read/write access. 0000: No cycle 0001: 1 cycle 0010: 2 cycles 0011: 3 cycles 0100: 4 cycles 0101: 5 cycles 0110: 6 cycles 0111: 8 cycles 1000: 10 cycles 1001: 12 cycles 1010: 14 cycles 1011: 18 cycles 1100: 24 cycles 1101: Reserved (setting prohibited) 1110: Reserved (setting prohibited) 1111: Reserved (setting prohibited)
6	WM	0	R/W	External Wait Mask Specification Specifies whether or not the external wait input is valid. The specification by this bit is valid even when the number of access wait cycle is 0. 0: External wait input is valid 1: External wait input is ignored
5 to 0	¾	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

- CS4WCR

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	BAS	-	WW[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	SW[1:0]		WR[3:0]			WM	-	-	-	-	HW[1:0]		
Initial value:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	BAS	0	R/W	SRAM with Byte Selection Byte Access Select Specifies the $\overline{WEn}$ and $\overline{RD/WR}$ signal timing when the SRAM interface with byte selection is used. 0: Asserts the $\overline{WEn}$ signal at the read timing and asserts the $\overline{RD/WR}$ signal during the write access cycle. 1: Asserts the $\overline{WEn}$ signal during the read access cycle and asserts the $\overline{RD/WR}$ signal at the write timing.
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
18 to 16	WW[2:0]	000	R/W	Number of Write Access Wait Cycles Specify the number of cycles that are necessary for write access. 000: The same cycles as WR[3:0] setting (number of read access wait cycles) 001: No cycle 010: 1 cycle 011: 2 cycles 100: 3 cycles 101: 4 cycles 110: 5 cycles 111: 6 cycles
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12, 11	SW[1:0]	00	R/W	Number of Delay Cycles from Address, $\overline{CS4}$ Assertion to $\overline{RD}$ , $\overline{WE}$ Assertion Specify the number of delay cycles from address and $\overline{CS4}$ assertion to $\overline{RD}$ and $\overline{WE}$ assertion. 00: 0.5 cycles 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles

Bit	Bit Name	Initial Value	R/W	Description
10 to 7	WR[3:0]	1010	R/W	Number of Read Access Wait Cycles Specify the number of cycles that are necessary for read access. 0000: No cycle 0001: 1 cycle 0010: 2 cycles 0011: 3 cycles 0100: 4 cycles 0101: 5 cycles 0110: 6 cycles 0111: 8 cycles 1000: 10 cycles 1001: 12 cycles 1010: 14 cycles 1011: 18 cycles 1100: 24 cycles 1101: Reserved (setting prohibited) 1110: Reserved (setting prohibited) 1111: Reserved (setting prohibited)
6	WM	0	R/W	External Wait Mask Specification Specifies whether or not the external wait input is valid. The specification by this bit is valid even when the number of access wait cycle is 0. 0: External wait input is valid 1: External wait input is ignored
5 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	HW[1:0]	00	R/W	Delay Cycles from $\overline{RD}$ , $\overline{WEn}$ Negation to Address, $\overline{CS4}$ Negation Specify the number of delay cycles from $\overline{RD}$ and $\overline{WEn}$ negation to address and $\overline{CS4}$ negation. 00: 0.5 cycles 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles

- CS5WCR

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	SZSEL	MPXW/ BAS	-	WW[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	SW[1:0]		WR[3:0]			WM	-	-	-	-	HW[1:0]		
Initial value:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description																				
31 to 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.																				
21	SZSEL	0	R/W	MPX-I/O Interface Bus Width Specification Specifies an address to select the bus width when the BSZ[1:0] of CS5BCR are specified as 11. This bit is valid only when area 5 is specified as MPX-I/O. 0: Selects the bus width by address A14 1: Selects the bus width by address A21 The relationship between the SZSEL bit and bus width selected by A14 or A21 are summarized below. <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>SZSEL</th> <th>A14</th> <th>A21</th> <th>Bus Width</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>Not affected</td> <td>8 bits</td> </tr> <tr> <td>0</td> <td>1</td> <td>Not affected</td> <td>16 bits</td> </tr> <tr> <td>1</td> <td>Not affected</td> <td>0</td> <td>8 bits</td> </tr> <tr> <td>1</td> <td>Not affected</td> <td>1</td> <td>16 bits</td> </tr> </tbody> </table>	SZSEL	A14	A21	Bus Width	0	0	Not affected	8 bits	0	1	Not affected	16 bits	1	Not affected	0	8 bits	1	Not affected	1	16 bits
SZSEL	A14	A21	Bus Width																					
0	0	Not affected	8 bits																					
0	1	Not affected	16 bits																					
1	Not affected	0	8 bits																					
1	Not affected	1	16 bits																					
20	MPXW	0	R/W	MPX-I/O Interface Address Wait This bit setting is valid only when area 5 is specified as MPX-I/O. Specifies the address cycle insertion wait for MPX-I/O interface. 0: Inserts no wait cycle 1: Inserts 1 wait cycle																				
20	BAS	0	R/W	SRAM with Byte Selection Byte Access Select This bit setting is valid only when area 5 is specified as SRAM with byte selection. Specifies the $\overline{WEn}$ and $\overline{RD/WR}$ signal timing when the SRAM interface with byte selection is used. 0: Asserts the $\overline{WEn}$ signal at the read timing and asserts the $\overline{RD/WR}$ signal during the write access cycle. 1: Asserts the $\overline{WEn}$ signal during the read access cycle and asserts the $\overline{RD/WR}$ signal at the write timing.																				
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.																				
18 to 16	WW[2:0]	000	R/W	Number of Write Access Wait Cycles Specify the number of cycles that are necessary for write access. 000: The same cycles as WR[3:0] setting (number of read access wait cycles) 001: No cycle 010: 1 cycle 011: 2 cycles 100: 3 cycles 101: 4 cycles 110: 5 cycles 111: 6 cycles																				
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.																				

Bit	Bit Name	Initial Value	R/W	Description
12, 11	SW[1:0]	00	R/W	<p>Number of Delay Cycles from Address, <math>\overline{CS5}</math> Assertion to <math>\overline{RD}</math>, <math>\overline{WE}</math> Assertion</p> <p>Specify the number of delay cycles from address and <math>\overline{CS5}</math> assertion to <math>\overline{RD}</math> and <math>\overline{WEn}</math> assertion when area 5 is specified as normal space or SRAM with byte selection. Specify the number of delay cycles from the end of address cycle (<math>Ta3</math>) to <math>\overline{RD}</math> and <math>\overline{WEn}</math> assertion when area 5 is specified as MPx-I/O.</p> <p>00: 0.5 cycles 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles</p>
10 to 7	WR[3:0]	1010	R/W	<p>Number of Read Access Wait Cycles</p> <p>Specify the number of cycles that are necessary for read access.</p> <p>0000: No cycle 0001: 1 cycle 0010: 2 cycles 0011: 3 cycles 0100: 4 cycles 0101: 5 cycles 0110: 6 cycles 0111: 8 cycles 1000: 10 cycles 1001: 12 cycles 1010: 14 cycles 1011: 18 cycles 1100: 24 cycles 1101: Reserved (setting prohibited) 1110: Reserved (setting prohibited) 1111: Reserved (setting prohibited)</p>
6	WM	0	R/W	<p>External Wait Mask Specification</p> <p>Specifies whether or not the external wait input is valid. The specification by this bit is valid even when the number of access wait cycle is 0.</p> <p>0: External wait input is valid 1: External wait input is ignored</p>
5 to 2	—	All 0	R	<p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p>
1, 0	HW[1:0]	00	R/W	<p>Delay Cycles from <math>\overline{RD}</math>, <math>\overline{WEn}</math> Negation to Address, <math>\overline{CS5}</math> Negation</p> <p>Specify the number of delay cycles from <math>\overline{RD}</math> and <math>\overline{WEn}</math> negation to address and <math>\overline{CS5}</math> negation when area 5 is specified as normal space or SRAM with byte selection. Specify the number of delay cycles from <math>\overline{RD}</math> and <math>\overline{WEn}</math> negation to <math>\overline{CS5}</math> negation when area 5 is specified as MPx-I/O.</p> <p>00: 0.5 cycles 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles</p>

## (2) Burst ROM (Clocked Asynchronous)

## • CS0WCR

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	BST[1:0]	-	-	-	-	BW[1:0]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	W[3:0]			WM	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description																		
31 to 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.																		
21, 20	BST[1:0]	00	R/W	Burst Count Specification Specify the burst count for 16-byte or more access. These bits must not be set to B'11, because B'11 setting is reserved. <table border="1"> <thead> <tr> <th>Bus Width</th> <th>BST[1:0]</th> <th>Burst count (16-byte access)</th> </tr> </thead> <tbody> <tr> <td rowspan="2">8 bits</td> <td>00</td> <td>16 burst × one time</td> </tr> <tr> <td>01</td> <td>4 burst × four times</td> </tr> <tr> <td rowspan="3">16 bits</td> <td>00</td> <td>8 burst × one time</td> </tr> <tr> <td>01</td> <td>2 burst × four times</td> </tr> <tr> <td>10</td> <td>4-4 or 2-4-2 burst</td> </tr> <tr> <td>32 bits</td> <td>xx</td> <td>4 burst × one time</td> </tr> </tbody> </table> Note: • For details, see table 8.17, Relationship between Bus Width, Access Size, and Number of Bursts.	Bus Width	BST[1:0]	Burst count (16-byte access)	8 bits	00	16 burst × one time	01	4 burst × four times	16 bits	00	8 burst × one time	01	2 burst × four times	10	4-4 or 2-4-2 burst	32 bits	xx	4 burst × one time
Bus Width	BST[1:0]	Burst count (16-byte access)																				
8 bits	00	16 burst × one time																				
	01	4 burst × four times																				
16 bits	00	8 burst × one time																				
	01	2 burst × four times																				
	10	4-4 or 2-4-2 burst																				
32 bits	xx	4 burst × one time																				
19, 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.																		
17, 16	BW[1:0]	00	R/W	Number of Burst Wait Cycles Specify the number of wait cycles to be inserted between the second or subsequent access cycles in burst access. 00: No cycle 01: 1 cycle 10: 2 cycles 11: 3 cycles																		
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.																		
10 to 7	W[3:0]	1010	R/W	Number of Access Wait Cycles Specify the number of wait cycles to be inserted in the first access cycle. 0000: No cycle 0001: 1 cycle 0010: 2 cycles 0011: 3 cycles 0100: 4 cycles 0101: 5 cycles 0110: 6 cycles 0111: 8 cycles 1000: 10 cycles 1001: 12 cycles 1010: 14 cycles 1011: 18 cycles 1100: 24 cycles 1101: Reserved (setting prohibited) 1110: Reserved (setting prohibited) 1111: Reserved (setting prohibited)																		

---

Bit	Bit Name	Initial Value	R/W	Description
6	WM	0	R/W	External Wait Mask Specification Specifies whether or not the external wait input is valid. The specification by this bit is valid even when the number of access wait cycle is 0. 0: External wait input is valid 1: External wait input is ignored
5 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

---



- CS4WCR

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	BST[1:0]	-	-	-	BW[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	SW[1:0]	W[3:0]			WM	-	-	-	-	-	-	-	HW[1:0]
Initial value:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description																		
31 to 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.																		
21, 20	BST[1:0]	00	R/W	Burst Count Specification Specify the burst count for 16-byte or more access. These bits must not be set to B'11, because B'11 setting is reserved. <table border="1"> <thead> <tr> <th>Bus Width</th> <th>BST[1:0]</th> <th>Burst count (16-byte access)</th> </tr> </thead> <tbody> <tr> <td rowspan="2">8 bits</td> <td>00</td> <td>16 burst × one time</td> </tr> <tr> <td>01</td> <td>4 burst × four times</td> </tr> <tr> <td rowspan="3">16 bits</td> <td>00</td> <td>8 burst × one time</td> </tr> <tr> <td>01</td> <td>2 burst × four times</td> </tr> <tr> <td>10</td> <td>4-4 or 2-4-2 burst</td> </tr> <tr> <td>32 bits</td> <td>xx</td> <td>4 burst × one time</td> </tr> </tbody> </table> Note: • For details, see table 8.17, Relationship between Bus Width, Access Size, and Number of Bursts.	Bus Width	BST[1:0]	Burst count (16-byte access)	8 bits	00	16 burst × one time	01	4 burst × four times	16 bits	00	8 burst × one time	01	2 burst × four times	10	4-4 or 2-4-2 burst	32 bits	xx	4 burst × one time
Bus Width	BST[1:0]	Burst count (16-byte access)																				
8 bits	00	16 burst × one time																				
	01	4 burst × four times																				
16 bits	00	8 burst × one time																				
	01	2 burst × four times																				
	10	4-4 or 2-4-2 burst																				
32 bits	xx	4 burst × one time																				
19, 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.																		
17, 16	BW[1:0]	00	R/W	Number of Burst Wait Cycles Specify the number of wait cycles to be inserted between the second or subsequent access cycles in burst access. 00: No cycle 01: 1 cycle 10: 2 cycles 11: 3 cycles																		
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.																		
12, 11	SW[1:0]	00	R/W	Number of Delay Cycles from Address, $\overline{CS4}$ Assertion to $\overline{RD}$ , $\overline{WEN}$ Assertion Specify the number of delay cycles from address and $\overline{CS4}$ assertion to $\overline{RD}$ and $\overline{WEN}$ assertion. 00: 0.5 cycles 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles																		

Bit	Bit Name	Initial Value	R/W	Description
10 to 7	W[3:0]	1010	R/W	Number of Access Wait Cycles Specify the number of wait cycles to be inserted in the first access cycle. 0000: No cycle 0001: 1 cycle 0010: 2 cycles 0011: 3 cycles 0100: 4 cycles 0101: 5 cycles 0110: 6 cycles 0111: 8 cycles 1000: 10 cycles 1001: 12 cycles 1010: 14 cycles 1011: 18 cycles 1100: 24 cycles 1101: Reserved (setting prohibited) 1110: Reserved (setting prohibited) 1111: Reserved (setting prohibited)
6	WM	0	R/W	External Wait Mask Specification Specifies whether or not the external wait input is valid. The specification by this bit is valid even when the number of access wait cycle is 0. 0: External wait input is valid 1: External wait input is ignored
5 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	HW[1:0]	00	R/W	Delay Cycles from $\overline{RD}$ , $\overline{WEn}$ Negation to Address, $\overline{CS4}$ Negation Specify the number of delay cycles from $\overline{RD}$ and $\overline{WEn}$ negation to address and $\overline{CS4}$ negation. 00: 0.5 cycles 01: 1.5 cycles 10: 2.5 cycles 11: 3.5 cycles

## (3) SDRAM\*

## • CS2WCR

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	A2CL[1:0]	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
9	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
8, 7	A2CL[1:0]	10	R/W	CAS Latency for Area 2 Specify the CAS latency for area 2. 00: 1 cycle 01: 2 cycles 10: 3 cycles 11: 4 cycles
6 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note: \* If only one area is connected to the SDRAM, specify area 3. In this case, specify area 2 as normal space or SRAM with byte selection.

- CS3WCR

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	WTRP[1:0]*	-	WTRCD[1:0]*	-	A3CL[1:0]	-	-	TRWL[1:0]*	-	WTRC[1:0]*	-	-	-	-	-
Initial value:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R	R/W	R/W	R	R/W	R/W	R	R	R/W	R/W	R	R/W	R/W

Note: \* If both areas 2 and 3 are specified as SDRAM, WTRP[1:0], WTRCD[1:0], TRWL[1:0], and WTRC[1:0] bit settings are used in both areas in common.

Bit	Bit Name	Initial Value	R/W	Description
31 to 15	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
14, 13	WTRP[1:0]*	00	R/W	Number of Auto-Precharge Completion Wait Cycles Specify the number of minimum precharge completion wait cycles as shown below. <ul style="list-style-type: none"> <li>From the start of auto-precharge and issuing of ACTV command for the same bank</li> <li>From issuing of the PRE/PALL command to issuing of the ACTV command for the same bank</li> <li>Till entering the power-down mode or deep power-down mode</li> <li>From the issuing of PALL command to issuing REF command in auto refresh mode</li> <li>From the issuing of PALL command to issuing SELF command in self refresh mode</li> </ul> The setting for areas 2 and 3 is common. 00: No cycle 01: 1 cycle 10: 2 cycles 11: 3 cycles
12	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
11, 10	WTRCD [1:0]*	01	R/W	Number of Wait Cycles between ACTV Command and READ(A)/WRIT(A) Command Specify the minimum number of wait cycles from issuing the ACTV command to issuing the READ(A)/WRIT(A) command. The setting for areas 2 and 3 is common. 00: No cycle 01: 1 cycle 10: 2 cycles 11: 3 cycles
9	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
8, 7	A3CL[1:0]	10	R/W	CAS Latency for Area 3 Specify the CAS latency for area 3. 00: 1 cycle 01: 2 cycles 10: 3 cycles 11: 4 cycles
6, 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
4, 3	TRWL[1:0]*	00	R/W	<p>Number of Auto-Precharge Startup Wait Cycles Specify the number of minimum auto-precharge startup wait cycles as shown below.</p> <ul style="list-style-type: none"> <li>• Cycle number from the issuance of the WRITA command by this LSI until the completion of auto-precharge in the SDRAM. Equivalent to the cycle number from the issuance of the WRITA command until the issuance of the ACTV command. Confirm that how many cycles are required between the WRITA command receive in the SDRAM and the auto-precharge activation, referring to each SDRAM data sheet. And set the cycle number so as not to exceed the cycle number specified by this bit.</li> <li>• Cycle number from the issuance of the WRIT command until the issuance of the PRE command. This is the case when accessing another low address in the same bank in bank active mode.</li> </ul> <p>The setting for areas 2 and 3 is common. 00: No cycle 01: 1 cycle 10: 2 cycles 11: 3 cycles</p>
2	—	0	R	<p>Reserved This bit is always read as 0. The write value should always be 0.</p>
1, 0	WTRC[1:0]*	00	R/W	<p>Number of Idle Cycles from REF Command/Self-Refresh Release to ACTV/REF/MRS Command Specify the number of minimum idle cycles in the periods shown below.</p> <ul style="list-style-type: none"> <li>• From the issuance of the REF command until the issuance of the ACTV/REF/MRS command</li> <li>• From releasing self-refresh until the issuance of the ACTV/REF/MRS command.</li> </ul> <p>The setting for areas 2 and 3 is common. 00: 2 cycles 01: 3 cycles 10: 5 cycles 11: 8 cycles</p>

Note: \* If both areas 2 and 3 are specified as SDRAM, WTRP[1:0], WTRCD[1:0], TRWL[1:0], and WTRC[1:0] bit settings are used in both areas in common.

If only one area is connected to the SDRAM, specify area 3. In this case, specify area 2 as normal space or SRAM with byte selection.

## (4) Burst ROM (Clocked Synchronous)

## • CS0WCR

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	BW[1:0]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	W[3:0]				WM	-	-	-	-	-	-
Initial value:	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17, 16	BW[1:0]	00	R/W	Number of Burst Wait Cycles Specify the number of wait cycles to be inserted between the second or subsequent access cycles in burst access. 00: No cycle 01: 1 cycle 10: 2 cycles 11: 3 cycles
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 7	W[3:0]	1010	R/W	Number of Access Wait Cycles Specify the number of wait cycles to be inserted in the first access cycle. 0000: No cycle 0001: 1 cycle 0010: 2 cycles 0011: 3 cycles 0100: 4 cycles 0101: 5 cycles 0110: 6 cycles 0111: 8 cycles 1000: 10 cycles 1001: 12 cycles 1010: 14 cycles 1011: 18 cycles 1100: 24 cycles 1101: Reserved (setting prohibited) 1110: Reserved (setting prohibited) 1111: Reserved (setting prohibited)
6	WM	0	R/W	External Wait Mask Specification Specifies whether or not the external wait input is valid. The specification by this bit is valid even when the number of access wait cycle is 0. 0: External wait input is valid 1: External wait input is ignored
5 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

### 8.4.4 SDRAM Control Register (SDCR)

SDCR specifies the method to refresh and access SDRAM, and the types of SDRAMs to be connected.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	A2ROW[1:0]	-	-	A2COL[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	DEEP	-	RFSH	RMODE	PDOWN	BACTV	-	-	-	A3ROW[1:0]	-	-	A3COL[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20, 19	A2ROW[1:0]	00	R/W	Number of Bits of Row Address for Area 2 Specify the number of bits of row address for area 2. 00: 11 bits 01: 12 bits 10: 13 bits 11: Reserved (setting prohibited)
18	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
17, 16	A2COL[1:0]	00	R/W	Number of Bits of Column Address for Area 2 Specify the number of bits of column address for area 2. 00: 8 bits 01: 9 bits 10: 10 bits 11: Reserved (setting prohibited)
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13	DEEP	0	R/W	Deep Power-Down Mode This bit is valid for low-power SDRAM. If the RFSH or RMODE bit is set to 1 while this bit is set to 1, the deep power-down entry command is issued and the low-power SDRAM enters the deep power-down mode. 0: Self-refresh mode 1: Deep power-down mode
12	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
11	RFSH	0	R/W	Refresh Control Specifies whether or not the refresh operation of the SDRAM is performed. 0: No refresh 1: Refresh
10	RMODE	0	R/W	Refresh Control Specifies whether to perform auto-refresh or self-refresh when the RFSH bit is 1. When the RFSH bit is 1 and this bit is 1, self-refresh starts immediately. When the RFSH bit is 1 and this bit is 0, auto-refresh starts according to the contents that are set in registers RTCSR, RTCNT, and RTCOR. 0: Auto-refresh is performed 1: Self-refresh is performed

Bit	Bit Name	Initial Value	R/W	Description
9	PDOWN	0	R/W	<p>Power-Down Mode</p> <p>Specifies whether the SDRAM will enter the power-down mode after the access to the SDRAM. With this bit being set to 1, after the SDRAM is accessed, the CKE signal is driven low and the SDRAM enters the power-down mode.</p> <p>0: The SDRAM does not enter the power-down mode after being accessed.</p> <p>1: The SDRAM enters the power-down mode after being accessed.</p>
8	BACTV	0	R/W	<p>Bank Active Mode</p> <p>Specifies to access whether in auto-precharge mode (using READA and WRITA commands) or in bank active mode (using READ and WRIT commands).</p> <p>0: Auto-precharge mode (using READA and WRITA commands)</p> <p>1: Bank active mode (using READ and WRIT commands)</p> <p>Note: Bank active mode can be set only for area 3. When both areas 2 and 3 are set to SDRAM, specify the auto-precharge mode.</p>
7 to 5	—	All 0	R	<p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p>
4, 3	A3ROW[1:0]	00	R/W	<p>Number of Bits of Row Address for Area 3</p> <p>Specify the number of bits of the row address for area 3.</p> <p>00: 11 bits</p> <p>01: 12 bits</p> <p>10: 13 bits</p> <p>11: Reserved (setting prohibited)</p>
2	—	0	R	<p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p>
1, 0	A3COL[1:0]	00	R/W	<p>Number of Bits of Column Address for Area 3</p> <p>Specify the number of bits of the column address for area 3.</p> <p>00: 8 bits</p> <p>01: 9 bits</p> <p>10: 10 bits</p> <p>11: Reserved (setting prohibited)</p>



### 8.4.5 Refresh Timer Control/Status Register (RTCSR)

RTCSR specifies various items about refresh for SDRAM.

When RTCSR is written, the upper 16 bits of the write data must be H'A55A to cancel write protection.

The phase of the clock for incrementing the count in the refresh timer counter (RTCNT) is adjusted only by a power-on reset. Note that there is an error in the time until the compare match flag is set for the first time after the timer is started with the CKS[2:0] bits being set to a value other than B'000.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	CMF	CMIE	CKS[2:0]			RRC[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0.
7	CMF	0	R/W	Compare Match Flag Indicates that a compare match occurs between the refresh timer counter (RTCNT) and refresh time constant register (RTCOR). This bit is set or cleared in the following conditions. 0: Clearing condition: When 0 is written in CMF after reading out RTCSR during CMF = 1. 1: Setting condition: When the condition RTCNT = RTCOR is satisfied.
6	CMIE	0	R/W	Compare Match Interrupt Enable Enables or disables CMF interrupt requests when the CMF bit in RTCSR is set to 1. 0: Disables CMF interrupt requests. 1: Enables CMF interrupt requests.
5 to 3	CKS[2:0]	000	R/W	Clock Select Select the clock input to count-up the refresh timer counter (RTCNT). 000: Stop the counting-up 001: CKIOφ/4 010: CKIOφ/16 011: CKIOφ/64 100: CKIOφ/256 101: CKIOφ/1024 110: CKIOφ/2048 111: CKIOφ/4096
2 to 0	RRC[2:0]	000	R/W	Refresh Count Specify the number of continuous refresh cycles, when the refresh request occurs after the coincidence of the values of the refresh timer counter (RTCNT) and the refresh time constant register (RTCOR). These bits can make the period of occurrence of refresh long. 000: 1 time 001: 2 times 010: 4 times 011: 6 times 100: 8 times 101: Reserved (setting prohibited) 110: Reserved (setting prohibited) 111: Reserved (setting prohibited)

### 8.4.6 Refresh Timer Counter (RTCNT)

RTCNT is an 8-bit counter that increments using the clock selected by bits CKS[2:0] in RTCSR. When RTCNT matches RTCOR, RTCNT is cleared to 0. The value in RTCNT returns to 0 after counting up to 255. When the RTCNT is written, the upper 16 bits of the write data must be H'A55A to cancel write protection.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-								
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0.
7 to 0		All 0	R/W	8-Bit Counter

### 8.4.7 Refresh Time Constant Register (RTCOR)

RTCOR is an 8-bit register. When RTCOR matches RTCNT, the CMF bit in RTCSR is set to 1 and RTCNT is cleared to 0.

When the RFSH bit in SDCR is 1, a memory refresh request is issued by this matching signal. This request is maintained until the refresh operation is performed. If the request is not processed when the next matching occurs, the previous request is ignored.

When the CMIE bit in RTCSR is set to 1, an interrupt request is issued by this matching signal. The request continues to be output until the CMF bit in RTCSR is cleared. Clearing the CMF bit only affects the interrupt request and does not clear the refresh request. Therefore, a combination of refresh request and interval timer interrupt can be specified so that the number of refresh requests are counted by using timer interrupts while refresh is performed periodically.

When RTCOR is written, the upper 16 bits of the write data must be H'A55A to cancel write protection.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-								
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0.
7 to 0		All 0	R/W	8-Bit Counter

### 8.4.8 Timeout Cycle Constant Register (TOSCORn) (n = 0 to 5)

TOSCORn is a 16-bit register the value of which is effective when the WM bit in the CSn space wait control register (CSnWCR) is 0 and the corresponding bit in the timeout enable register (TOENR) is 1. When the number of cycles of waiting due to the signal on the external wait input pin matches the setting of TOSCORn, external wait input is disabled to end the cycle of access, the timeout status flag for the corresponding space in the timeout status register (TOSTR) is set, and a timeout detection interrupt request is generated. The timeout detection interrupt request is retained until the corresponding bit in TOENR is set to 0 or 0 is written to the timeout status flag for the corresponding space. Note that timeout detection is enabled even while the timeout status flag for the corresponding space in TOSTR is 1, and external wait input is disabled in response to a further timeout.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0.
15 to 0		All 0	R/W	16-Bit Register H'0000: 65536 cycles H'0001: 1 cycle : H'FFFF: 65535 cycles

### 8.4.9 Timeout Status Register (TOSTR)

TOSTR is an 8-bit register that holds the timeout status flags for the CS spaces. When the WM bit in the CSn space wait control register (CSnWCR) is 0 and the corresponding bit in the timeout enable register (TOENR) is 1 and the number of cycles of waiting in response to the signal on the external wait input matches the setting of TOSCORn, the timeout status flag for the corresponding space is set and a timeout detection interrupt request is generated. The only writable value for the timeout status flags is 0, which clears the flag. Writing 1 to a flag is ignored.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	CS5TOSTF	CS4TOSTF	CS3TOSTF	CS2TOSTF	CS1TOSTF	CS0TOSTF
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved These bits are always read as 0.
5	CS5TOSTF	0	R/W	CS5 Space Timeout Status Flag Status flag that indicates that the number of cycles of waiting due to the input on the external wait pin during access to the CS5 space has matched the setting of the CS5 space timeout cycle constant register (TOSCOR5). This bit is set or cleared in the following conditions. 0: Clearing condition When 0 is written in CS5TOSTF. 1: Setting condition When the WM bit in the CS5 space wait control register (CS5WCR) is 0 and the CS5TOEN bit in the timeout enable register (TOENR) is 1, the number of cycles of waiting due to the input on the external wait pin during access to the CS5 space has matched the setting of TOSCOR5.
4	CS4TOSTF	0	R/W	CS4 Space Timeout Status Flag Status flag that indicates that the number of cycles of waiting due to the input on the external wait pin during access to the CS4 space has matched the setting of the CS4 space timeout cycle constant register (TOSCOR4). For the condition to set or clear this bit, refer to the description of CS5TOSTF.
3	CS3TOSTF	0	R/W	CS3 Space Timeout Status Flag Status flag that indicates that the number of cycles of waiting due to the input on the external wait pin during access to the CS3 space has matched the setting of the CS3 space timeout cycle constant register (TOSCOR3). For the condition to set or clear this bit, refer to the description of CS5TOSTF.
2	CS2TOSTF	0	R/W	CS2 Space Timeout Status Flag Status flag that indicates that the number of cycles of waiting due to the input on the external wait pin during access to the CS2 space has matched the setting of the CS2 space timeout cycle constant register (TOSCOR2). For the condition to set or clear this bit, refer to the description of CS5TOSTF.
1	CS1TOSTF	0	R/W	CS1 Space Timeout Status Flag Status flag that indicates that the number of cycles of waiting due to the input on the external wait pin during access to the CS1 space has matched the setting of the CS1 space timeout cycle constant register (TOSCOR1). For the condition to set or clear this bit, refer to the description of CS5TOSTF.

---

Bit	Bit Name	Initial Value	R/W	Description
0	CS0TOSTF	0	R/W	CS0 Space Timeout Status Flag Status flag that indicates that the number of cycles of waiting due to the input on the external wait pin during access to the CS0 space has matched the setting of the CS0 space timeout cycle constant register (TOSCOR0). For the condition to set or clear this bit, refer to the description of CS5TOSTF.

---

### 8.4.10 Timeout Enable Register (TOENR)

TOENR is an 8-bit register that specifies enabling or disabling the detection of timeout for waiting in each of the CS spaces.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	CS5 TOEN	CS4 TOEN	CS3 TOEN	CS2 TOEN	CS1 TOEN	CS0 TOEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved These bits are always read as 0.
5	CS5TOEN	0	R/W	CS5 Space Timeout Detection Enable Specifies enabling or disabling the detection of timeout for waiting in the CS5 space. 1: The timeout detection is disabled. 0: The timeout detection is enabled.
4	CS4TOEN	0	R/W	CS4 Space Timeout Detection Enable Specifies enabling or disabling the detection of timeout for waiting in the CS4 space. 1: The timeout detection is disabled. 0: The timeout detection is enabled.
3	CS3TOEN	0	R/W	CS3 Space Timeout Detection Enable Specifies enabling or disabling the detection of timeout for waiting in the CS3 space. 1: The timeout detection is disabled. 0: The timeout detection is enabled.
2	CS2TOEN	0	R/W	CS2 Space Timeout Detection Enable Specifies enabling or disabling the detection of timeout for waiting in the CS2 space. 1: The timeout detection is disabled. 0: The timeout detection is enabled.
1	CS1TOEN	0	R/W	CS1 Space Timeout Detection Enable Specifies enabling or disabling the detection of timeout for waiting in the CS1 space. 1: The timeout detection is disabled. 0: The timeout detection is enabled.
0	CS0TOEN	0	R/W	CS0 Space Timeout Detection Enable Specifies enabling or disabling the detection of timeout for waiting in the CS0 space. 1: The timeout detection is disabled. 0: The timeout detection is enabled.

## 8.5 Operation

### 8.5.1 Access Size and Data Alignment

This LSI supports little endian, in which the least significant byte (LSB) is that in the direction of the 0th address.

Data bus width can be selected from 8 bits, 16 bits, and 32 bits for the normal memory and SRAM with byte selection.

Data bus width can be selected from 16 bits and 32 bits for SDRAM. For MPX-I/O, the data bus width is fixed to either 8 or 16 bits, or made selectable as 8 bits or 16 bits by one of the address lines.

Data bus width varies depending on boot mode. For details, refer to section 8.3.2, Data Bus Width and Related Pin Setting for Each Area Depending on Boot Mode.

Data alignment is performed in accordance with the data bus width selected for the device. This also means that four read operations are required to read 32-bit data from a byte-width device. In this LSI, data alignment and conversion of data length is performed automatically between the respective interfaces.

Table 8.5 to Table 8.8 show the relationship between device data width and access unit.

**Table 8.5 32-Bit External Device Access and Data Alignment in Little Endian**

Operation	Data Bus				Strobe Signals			
	D31 to D24	D23 to D16	D15 to D8	D7 to D0	WE3, DQMUU	WE2, DQMUL	WE1, DQMLU	WE0, DQMLL
8-bit access at address 0	—	—	—	Data 7 to 0	—	—	—	Assert
8-bit access at address 1	—	—	Data 7 to 0	—	—	—	Assert	—
8-bit access at address 2	—	Data 7 to 0	—	—	—	Assert	—	—
8-bit access at address 3	Data 7 to 0	—	—	—	Assert	—	—	—
16-bit access at address 0	—	—	Data 15 to 8	Data 7 to 0	—	—	Assert	Assert
16-bit access at address 2	Data 15 to 8	Data 7 to 0	—	—	Assert	Assert	—	—
32-bit access at address 0	Data 31 to 24	Data 23 to 16	Data 15 to 8	Data 7 to 0	Assert	Assert	Assert	Assert

**Table 8.6 16-Bit External Device Access and Data Alignment in Little Endian**

Operation	Data Bus				Strobe Signals			
	D31 to D24	D23 to D16	D15 to D8	D7 to D0	WE3, DQMUU	WE2, DQMUL	WE1, DQMLU	WE0, DQMLL
8-bit access at address 0	—	—	—	Data 7 to 0	—	—	—	Assert
8-bit access at address 1	—	—	Data 7 to 0	—	—	—	Assert	—
8-bit access at address 2	—	—	—	Data 7 to 0	—	—	—	Assert
8-bit access at address 3	—	—	Data 7 to 0	—	—	—	Assert	—
16-bit access at address 0	—	—	Data 15 to 8	Data 7 to 0	—	—	Assert	Assert
16-bit access at address 2	—	—	Data 15 to 8	Data 7 to 0	—	—	Assert	Assert

**Table 8.6 16-Bit External Device Access and Data Alignment in Little Endian**

Operation	Data Bus				Strobe Signals				
	D31 to D24	D23 to D16	D15 to D8	D7 to D0	WE3, DQMUU	WE2, DQMUL	WE1, DQMLU	WE0, DQMLL	
32-bit access at address 0	1st access at address 0	—	—	Data 15 to 8	Data 7 to 0	—	—	Assert	Assert
	2nd access at address 2	—	—	Data 31 to 24	Data 23 to 16	—	—	Assert	Assert

**Table 8.7 8-Bit External Device Access and Data Alignment in Little Endian**

Operation	Data Bus				Strobe Signals			
	D31 to D24	D23 to D16	D15 to D8	D7 to D0	WE3, DQMUU	WE2, DQMUL	WE1, DQMLU	WE0, DQMLL
8-bit access at address 0	—	—	—	Data 7 to 0	—	—	—	Assert
8-bit access at address 1	—	—	—	Data 7 to 0	—	—	—	Assert
8-bit access at address 2	—	—	—	Data 7 to 0	—	—	—	Assert
8-bit access at address 3	—	—	—	Data 7 to 0	—	—	—	Assert
16-bit access at address 0	1st access at address 0	—	—	Data 7 to 0	—	—	—	Assert
	2nd access at address 1	—	—	Data 15 to 8	—	—	—	Assert
16-bit access at address 2	1st access at address 0	—	—	Data 7 to 0	—	—	—	Assert
	2nd access at address 1	—	—	Data 15 to 8	—	—	—	Assert
32-bit access at address 0	1st access at address 0	—	—	Data 7 to 0	—	—	—	Assert
	2nd access at address 1	—	—	Data 15 to 8	—	—	—	Assert
	3rd access at address 2	—	—	Data 23 to 16	—	—	—	Assert
	4th access at address 3	—	—	Data 31 to 24	—	—	—	Assert



## 8.5.2 Normal Space Interface

### (1) Basic Timing

For access to a normal space, this LSI uses strobe signal output in consideration of the fact that mainly static RAM will be directly connected. When using SRAM with a byte-selection pin, see section 8.5.8, SRAM Interface with Byte Selection. Figure 8.2 shows the basic timings of normal space access. A no-wait normal access is completed in two cycles. The  $\overline{BS}$  signal is asserted for one cycle to indicate the start of a bus cycle.

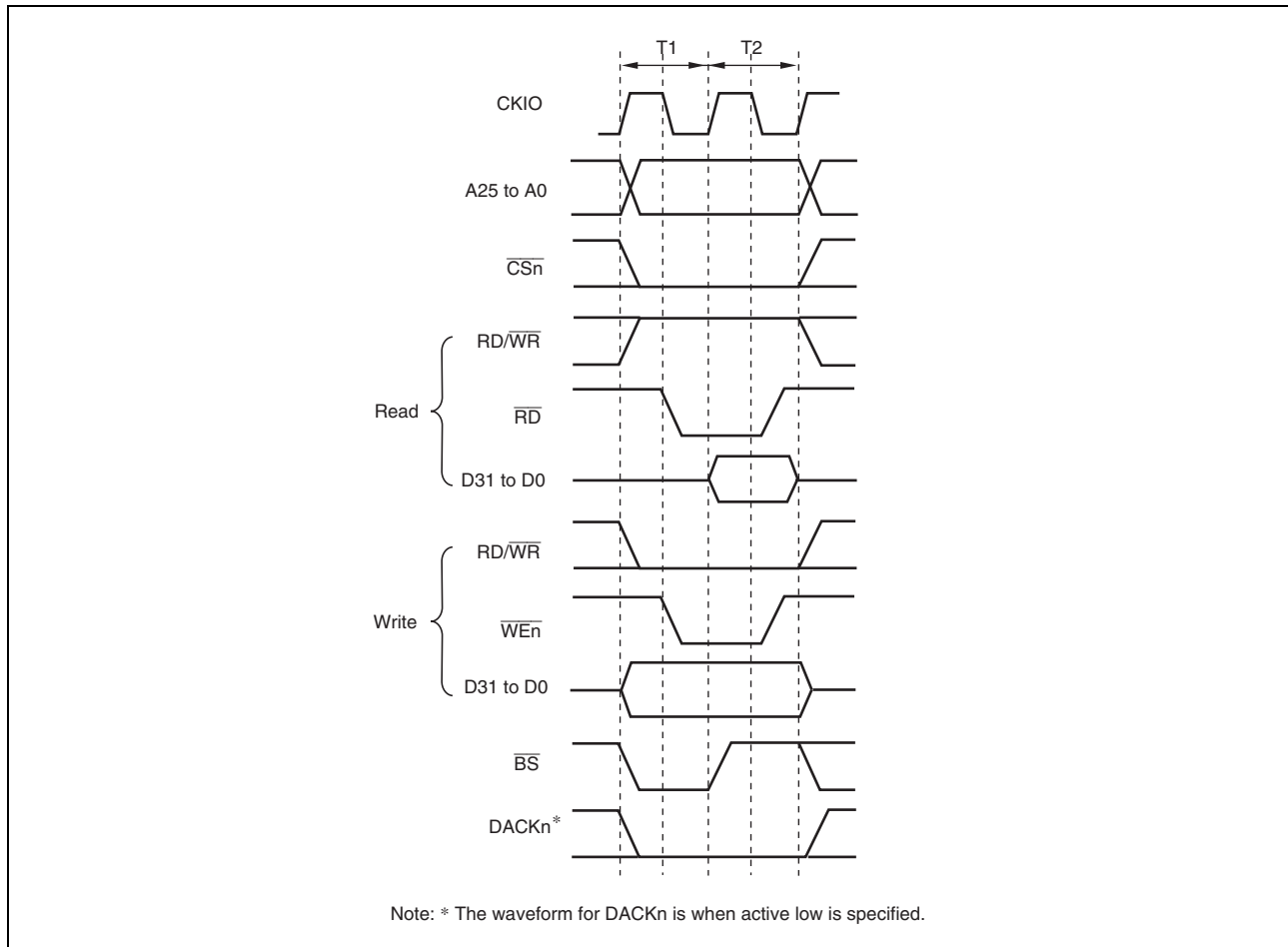
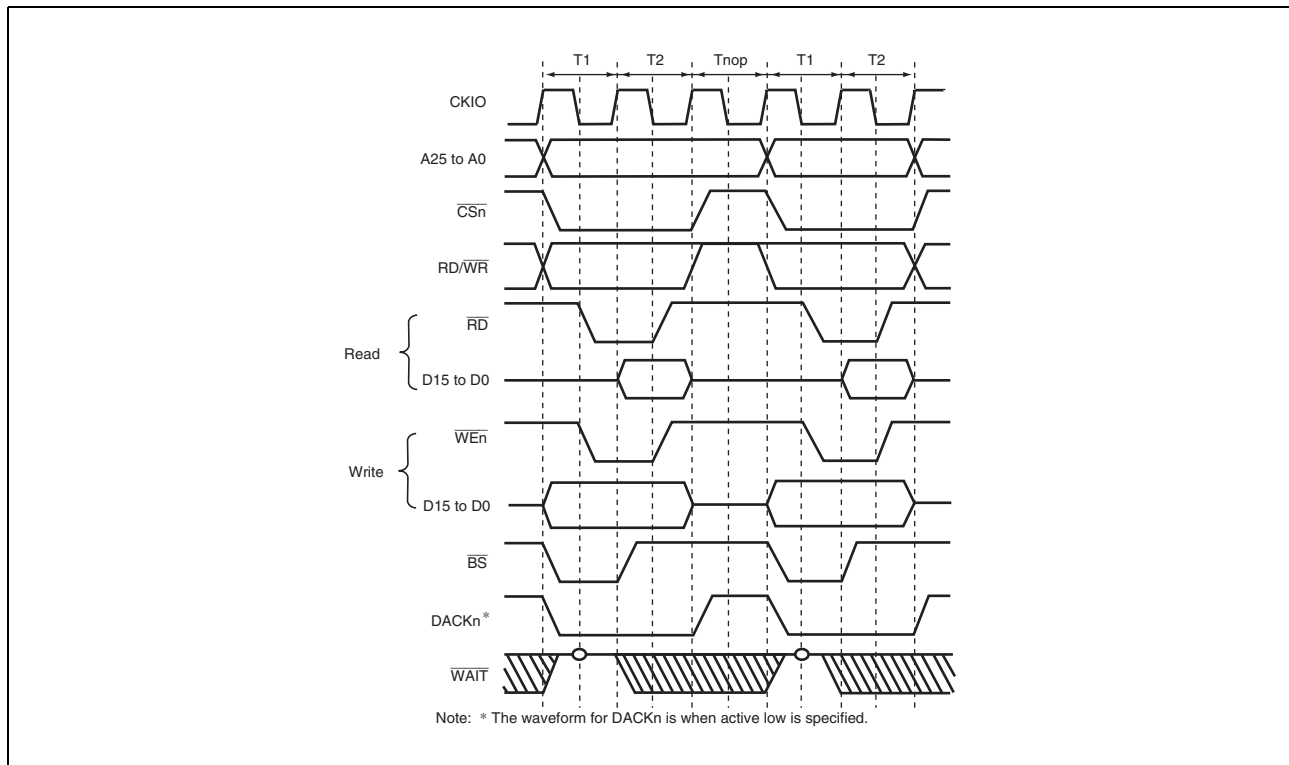


Figure 8.2 Normal Space Basic Access Timing (Access Wait 0)

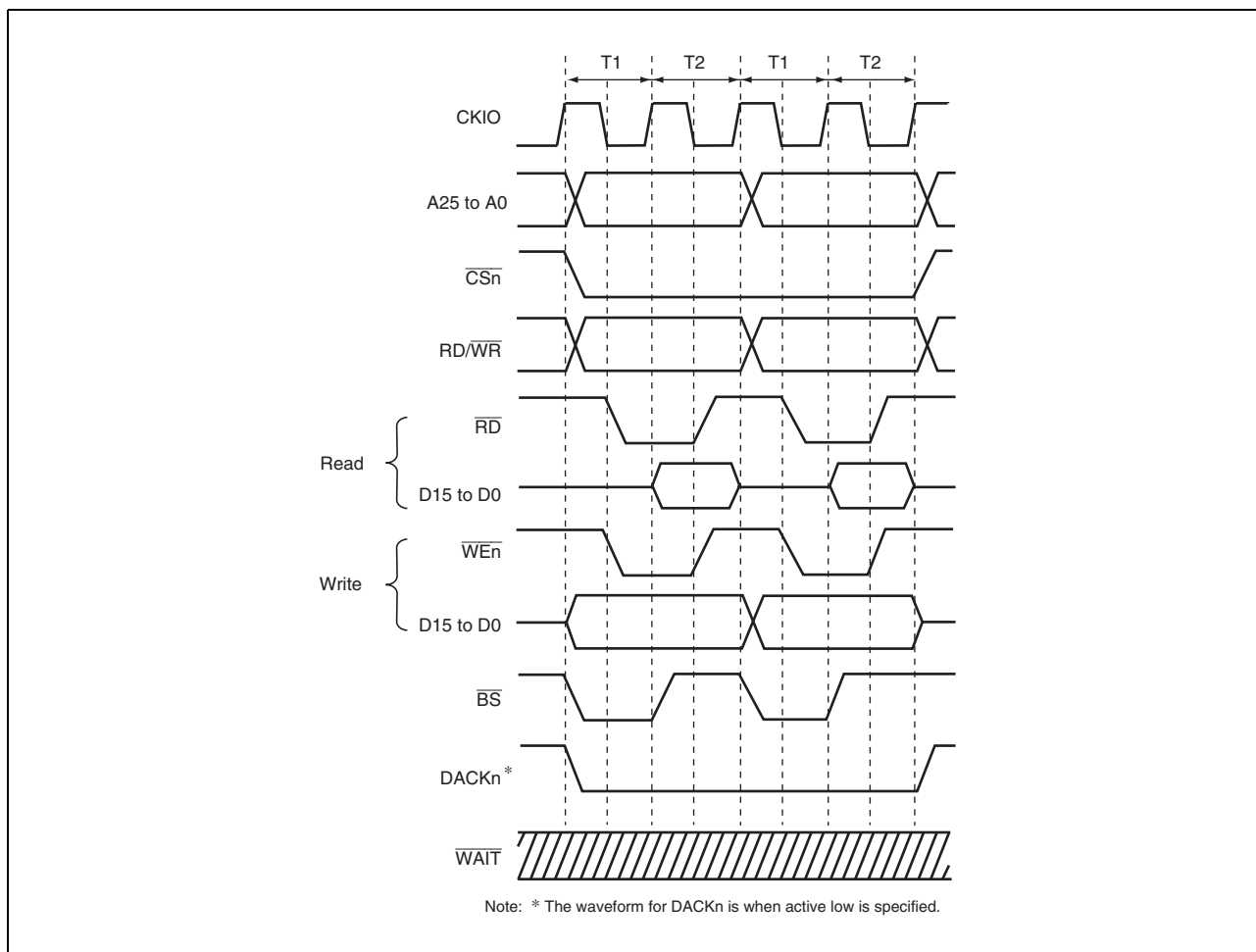
There is no access size specification when reading. The correct access start address is output in the least significant bit of the address, but since there is no access size specification, 32 bits are always read in case of a 32-bit device. 16 bits are always read in case of a 16-bit device. When writing, only the  $\overline{WEn}$  signal for the byte to be written is asserted.

It is necessary to output the data that has been read using  $\overline{RD}$  when a buffer is established in the data bus. The  $\overline{RD}/\overline{WR}$  signal is in a read state (high output) when no access has been carried out. Therefore, care must be taken when controlling the external data buffer with this signal, to avoid output collision.

Figure 8.3 and Figure 8.4 show the basic timings in continuous access to normal space. If the WM bit in CSnWCR is cleared to 0, a Tnop cycle is inserted after the CSn space access to evaluate the external wait (Figure 8.3). If the WM bit in CSnWCR is set to 1, external waits are ignored and no Tnop cycle is inserted (Figure 8.4).



**Figure 8.3** Continuous Access to Normal Space (1) Bus Width = 16 Bits, 32-Bit Access, CSnWCR.WM Bit = 0 (Access Wait = 0, Cycle Wait = 0)



**Figure 8.4** Continuous Access to Normal Space (2) Bus Width = 16 Bits, 32-Bit Access, CSnWCR.WM Bit = 1 (Access Wait = 0, Cycle Wait = 0)

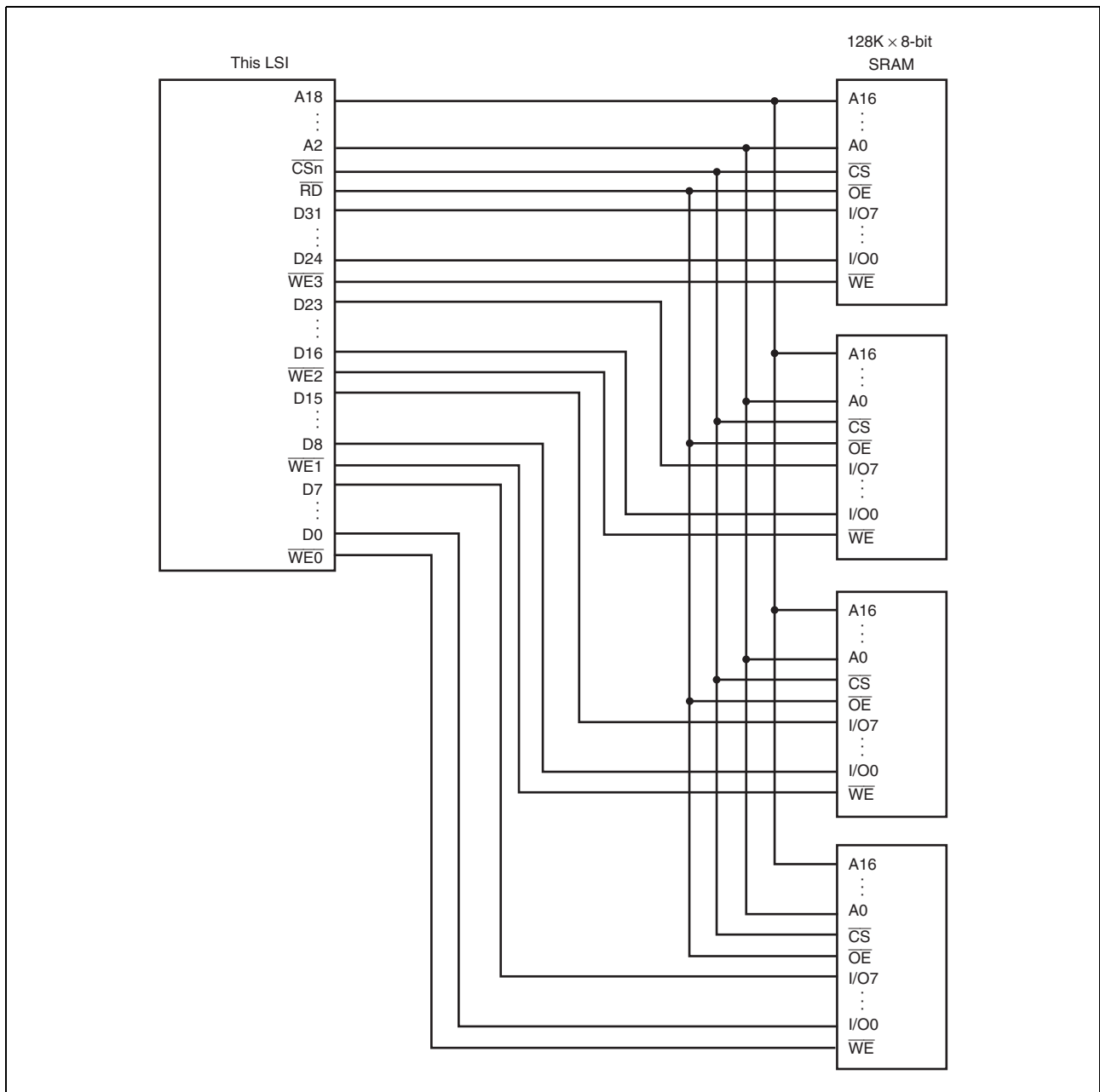


Figure 8.5 Example of 32-Bit Data-Width SRAM Connection

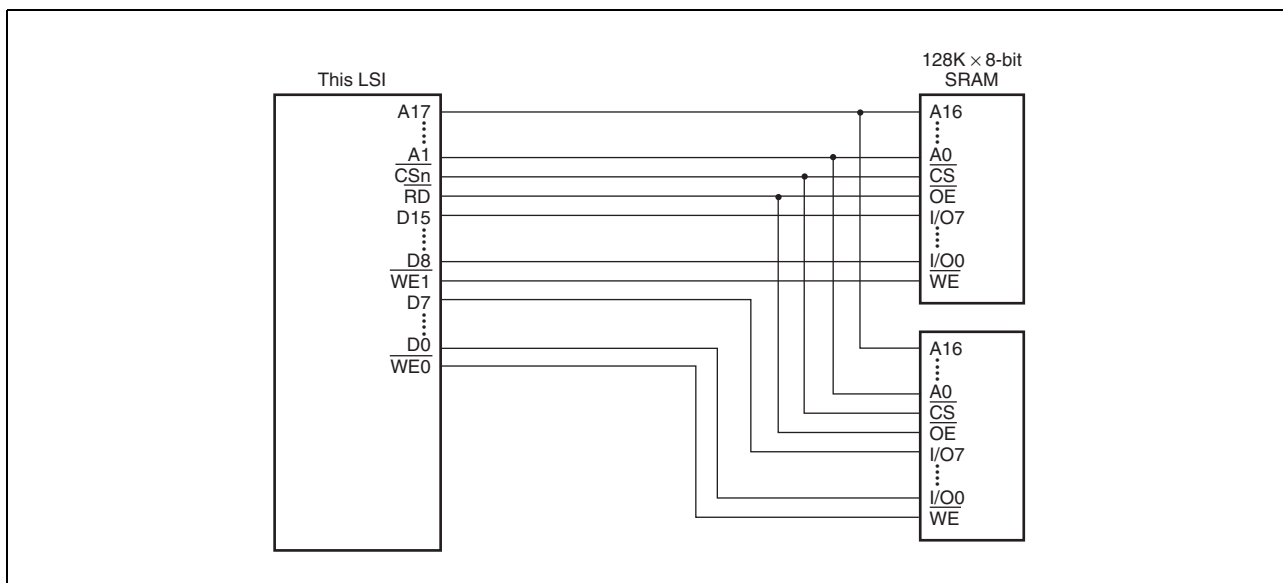


Figure 8.6 Example of 16-Bit Data-Width SRAM Connection

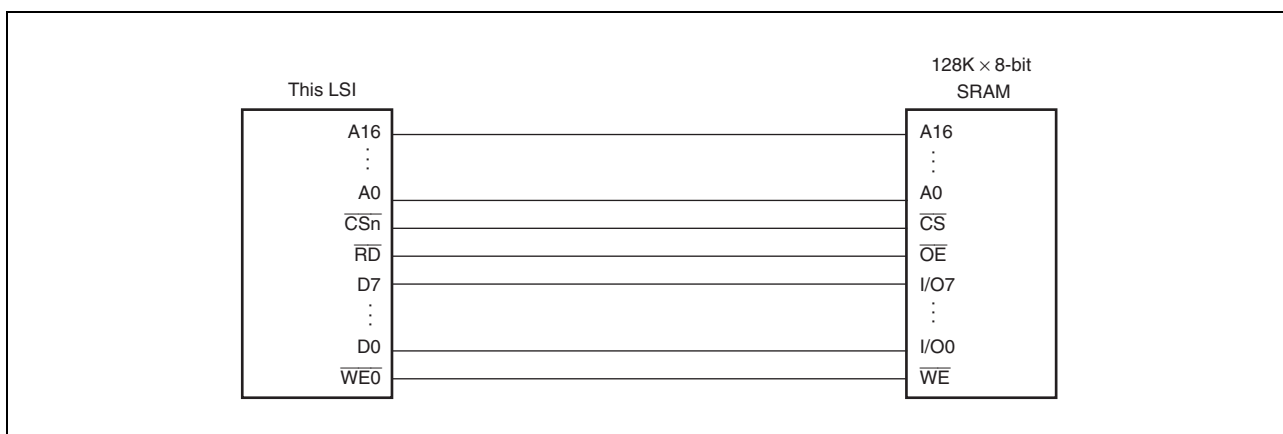
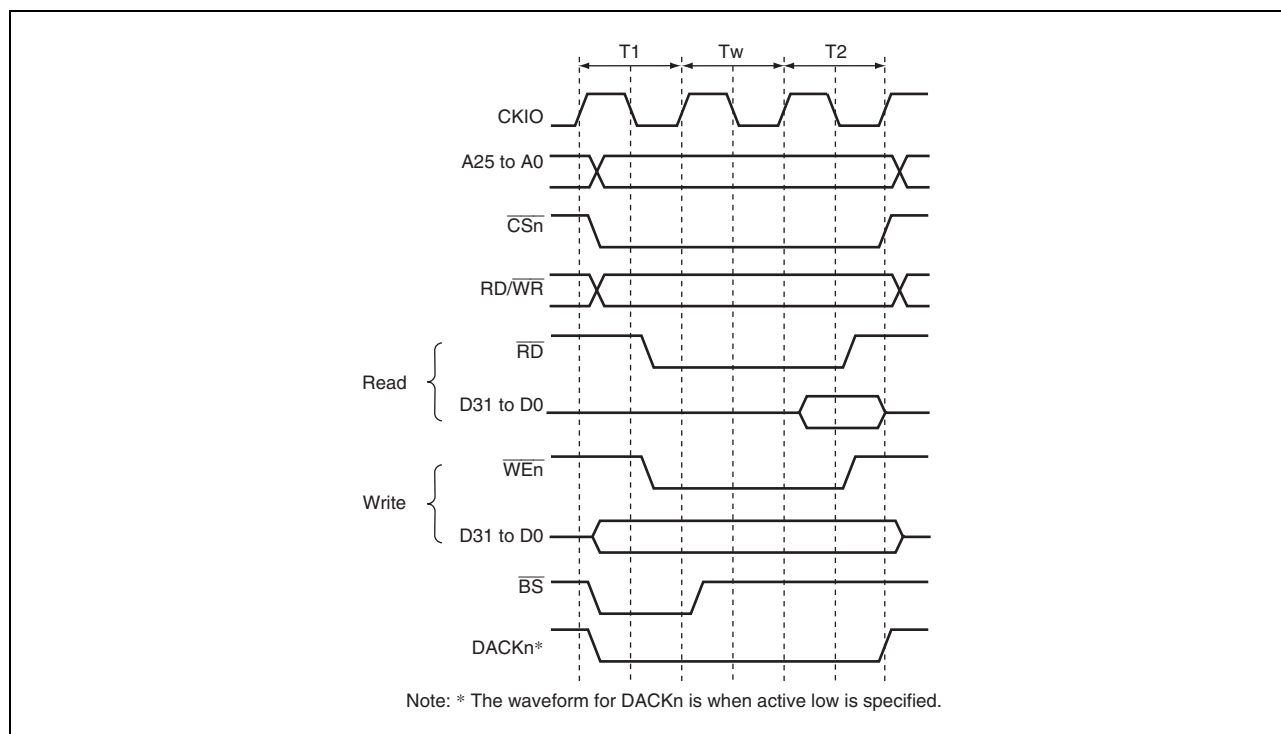


Figure 8.7 Example of 8-Bit Data-Width SRAM Connection

### 8.5.3 Access Wait Control

Wait cycle insertion on a normal space access can be controlled by the settings of bits WR3 to WR0 in CSnWCR. It is possible for areas 1, 4, and 5 to insert wait cycles independently in read access and in write access. Areas 0, 2, and 3 have common access wait for read cycle and write cycle. The specified number of  $T_w$  cycles are inserted as wait cycles in a normal space access shown in Figure 8.8.



**Figure 8.8 Wait Timing for Normal Space Access (Software Wait Only)**

When the WM bit in CSnWCR is cleared to 0, the external wait input  $\overline{\text{WAIT}}$  signal is also sampled.  $\overline{\text{WAIT}}$  pin sampling is shown in Figure 8.9. A 2-cycle wait is specified as a software wait. The  $\overline{\text{WAIT}}$  signal is sampled on the falling edge of CKIO at the transition from the T1 or Tw cycle to the T2 cycle.

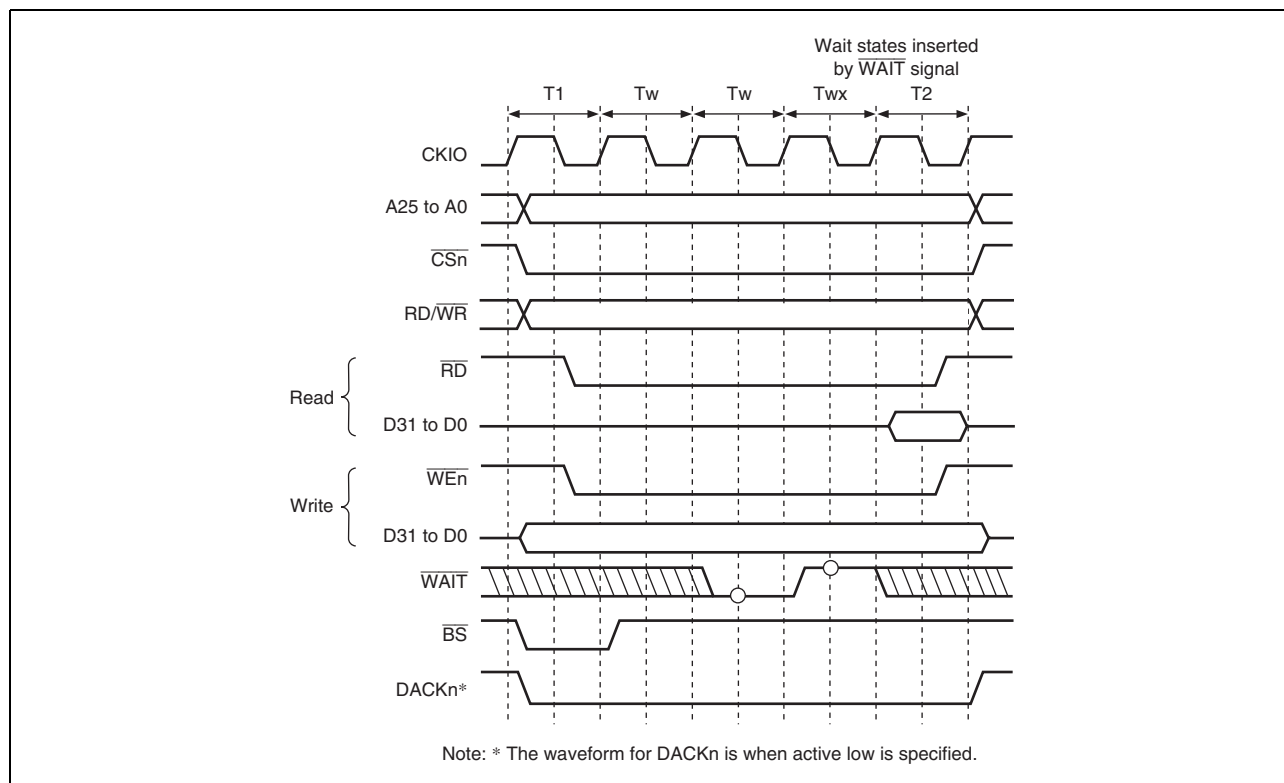


Figure 8.9 Wait Cycle Timing for Normal Space Access (Wait Cycle Insertion Using WAIT Signal)

### 8.5.4 $\overline{\text{CSn}}$ Assert Period Expansion

The number of cycles from  $\overline{\text{CSn}}$  assertion to  $\overline{\text{RD}}$ ,  $\overline{\text{WEn}}$  assertion can be specified by setting bits SW1 and SW0 in CSnWCR. The number of cycles from  $\overline{\text{RD}}$ ,  $\overline{\text{WEn}}$  negation to  $\overline{\text{CSn}}$  negation can be specified by setting bits HW1 and HW0. Therefore, a flexible interface to an external device can be obtained. Figure 8.10 shows an example. A  $T_h$  cycle and a  $T_f$  cycle are added before and after an ordinary cycle, respectively. In these cycles,  $\overline{\text{RD}}$  and  $\overline{\text{WEn}}$  are not asserted, while other signals are asserted. The data output is prolonged to the  $T_f$  cycle, and this prolongation is useful for devices with slow writing operations.

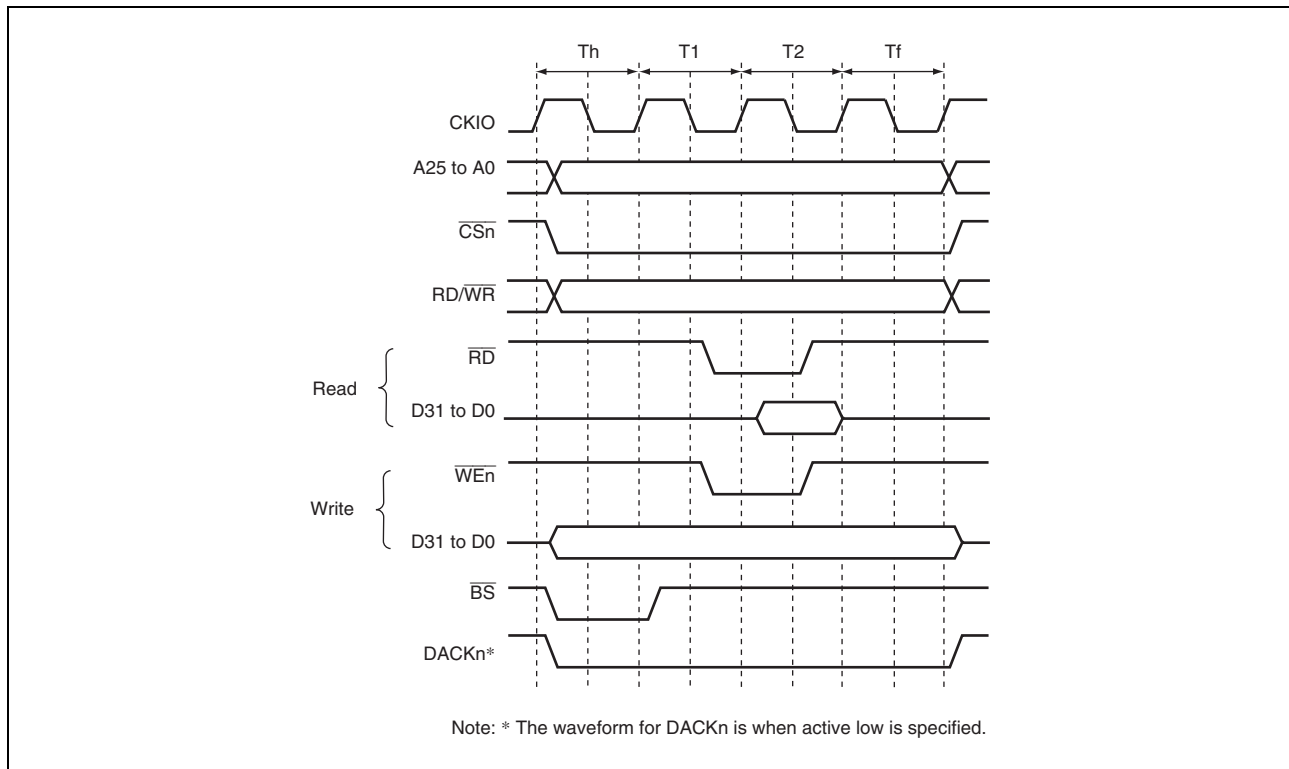


Figure 8.10  $\overline{\text{CSn}}$  Assert Period Expansion



### 8.5.5 MPX-I/O Interface

Access timing for the MPX space is shown below. In the MPX space,  $\overline{CS5}$ ,  $\overline{AH}$ ,  $\overline{RD}$ , and  $\overline{WE_n}$  signals control the accessing. The basic access for the MPX space consists of 2 cycles of address output followed by an access to a normal space. The bus width for the address output cycle or the data input/output cycle is fixed to 8 bits or 16 bits. Alternatively, it can be 8 bits or 16 bits depending on the address to be accessed.

Output of the addresses D15 to D0 or D7 to D0 is performed from cycle Ta2 to cycle Ta3. Because cycle Ta1 has a high-impedance state, collisions of addresses and data can be avoided without inserting idle cycles, even in continuous access cycles. Address output is increased to 3 cycles by setting the MPXW bit in CS5WCR to 1.

The  $\overline{RD}/\overline{WR}$  signal is output at the same time as the  $\overline{CS5}$  signal; it is high in the read cycle and low in the write cycle.

The data cycle is the same as that in a normal space access.

The delay cycles the number of which is specified by SW[1:0] are inserted between cycle Ta3 and cycle T1. The delay cycles the number of which is specified by HW[1:0] are added after cycle T2.

Timing charts are shown in Figure 8.11 to Figure 8.13.

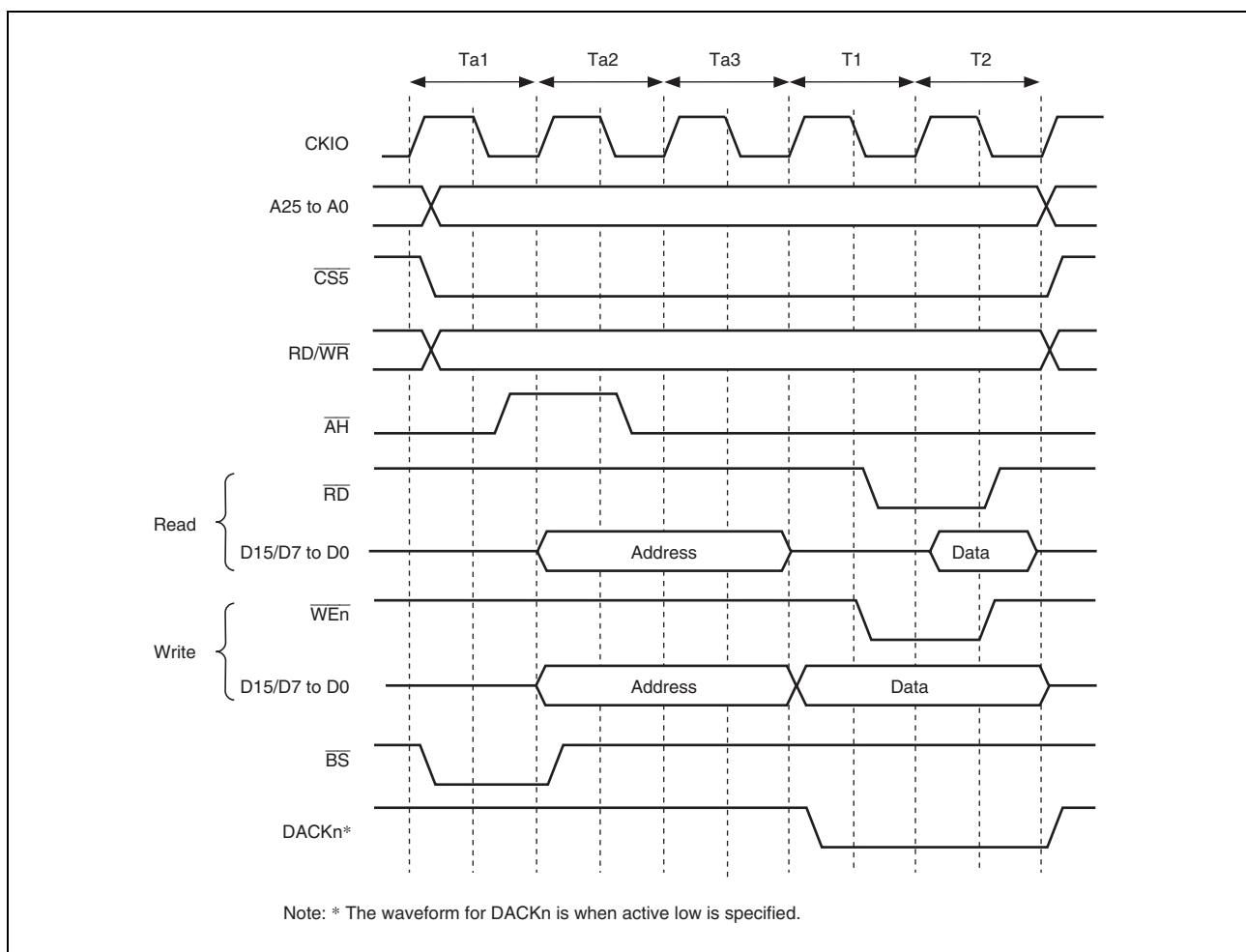


Figure 8.11 (1) Access Timing for MPX Space (Address Cycle No Wait, Data Cycle No Wait)

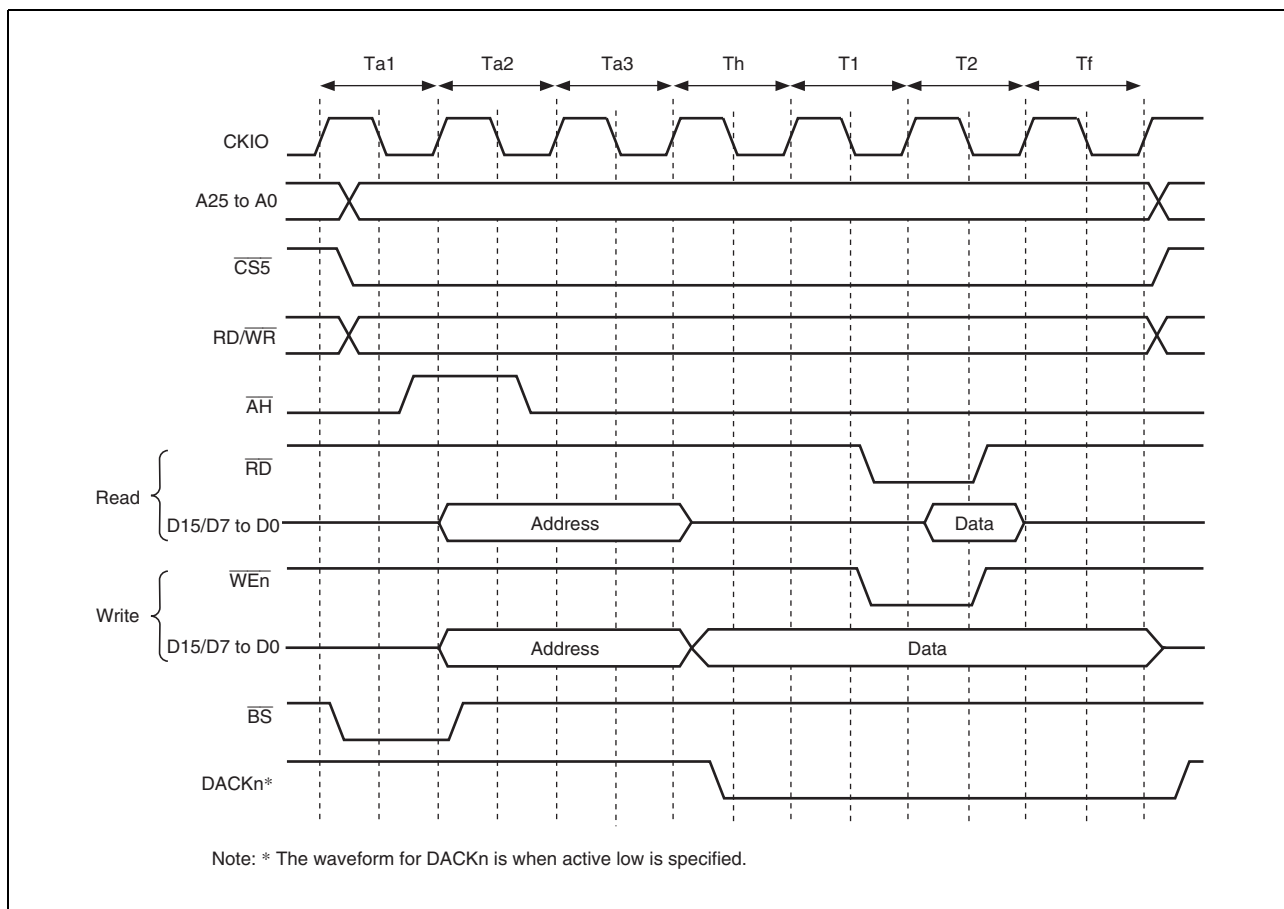


Figure 8.11 (2) Access Timing for MPX Space (Address Cycle No Wait, Extended Assertion Cycle 1.5, Data Cycle No Wait, Extended Negation Cycle 1.5)

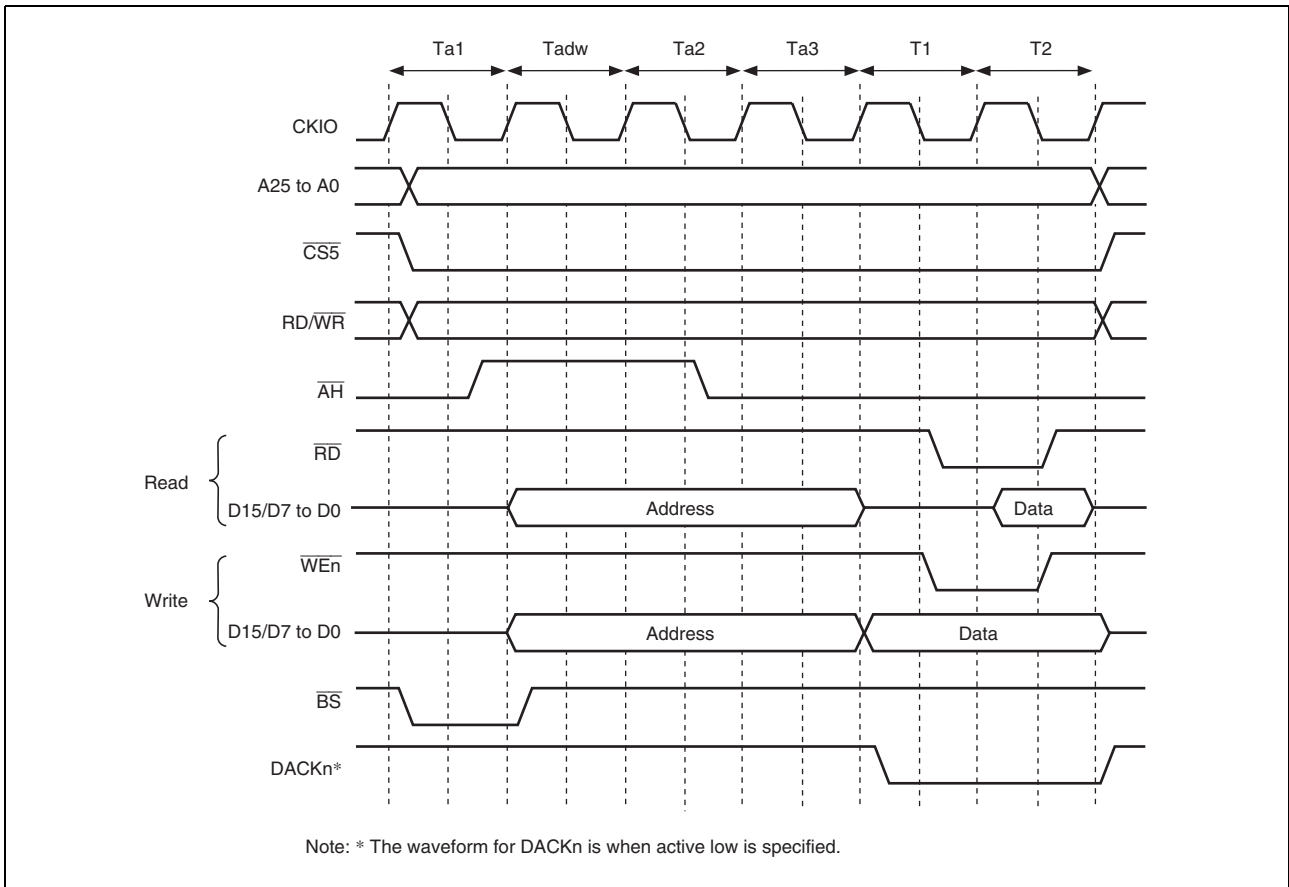


Figure 8.12 Access Timing for MPX Space (Address Cycle Wait 1, Data Cycle No Wait)

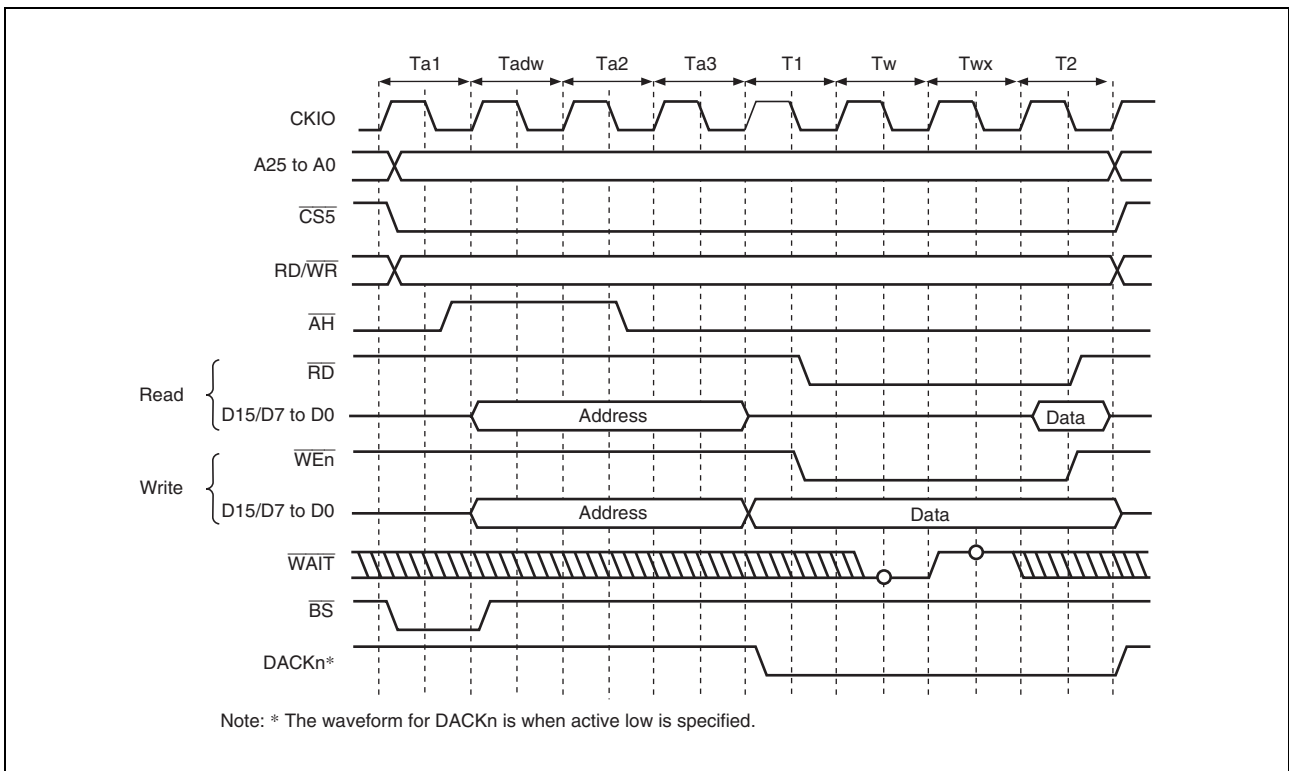


Figure 8.13 Access Timing for MPX Space (Address Cycle Access Wait 1, Data Cycle Wait 1, External Wait 1)

## 8.5.6 SDRAM Interface

### (1) SDRAM Direct Connection

The SDRAM that can be connected to this LSI is a product that has 11/12/13 bits of row address, 8/9/10 bits of column address, 4 or less banks, and uses the A10 pin for setting precharge mode in read and write command cycles.

The control signals for direct connection of SDRAM are  $\overline{RAS}$ ,  $\overline{CAS}$ ,  $RD/\overline{WR}$ , DQM0, DQM1, DQML0, DQML1, CKE,  $\overline{CS2}$ , and  $\overline{CS3}$ . All the signals other than  $\overline{CS2}$  and  $\overline{CS3}$  are common to all areas, and signals other than CKE are valid only when  $\overline{CS2}$  or  $\overline{CS3}$  is asserted. SDRAM can be connected to up to 2 spaces. The data bus width of the area that is connected to SDRAM is 16 bits or 32 bits.

Burst read/single write (burst length 1) and burst read/burst write (burst length 1) are supported as the SDRAM operating mode.

Commands for SDRAM can be specified by  $\overline{RAS}$ ,  $\overline{CAS}$ ,  $RD/\overline{WR}$ , and specific address signals. These commands supports:

- NOP
- Auto-refresh (REF)
- Self-refresh (SELF)
- All banks pre-charge (PALL)
- Specified bank pre-charge (PRE)
- Bank active (ACTV)
- Read (READ)
- Read with pre-charge (READA)
- Write (WRIT)
- Write with pre-charge (WRITA)
- Write mode register (MRS, EMRS)

The byte to be accessed is specified by DQM0, DQM1, DQML0, and DQML1. Reading or writing is performed for a byte whose corresponding DQMxx is low. For details on the relationship between DQMxx and the byte to be accessed, see section 8.5.1, Access Size and Data Alignment.

Figure 8.14 and Figure 8.15 show examples of the connection of the SDRAM with the LSI.

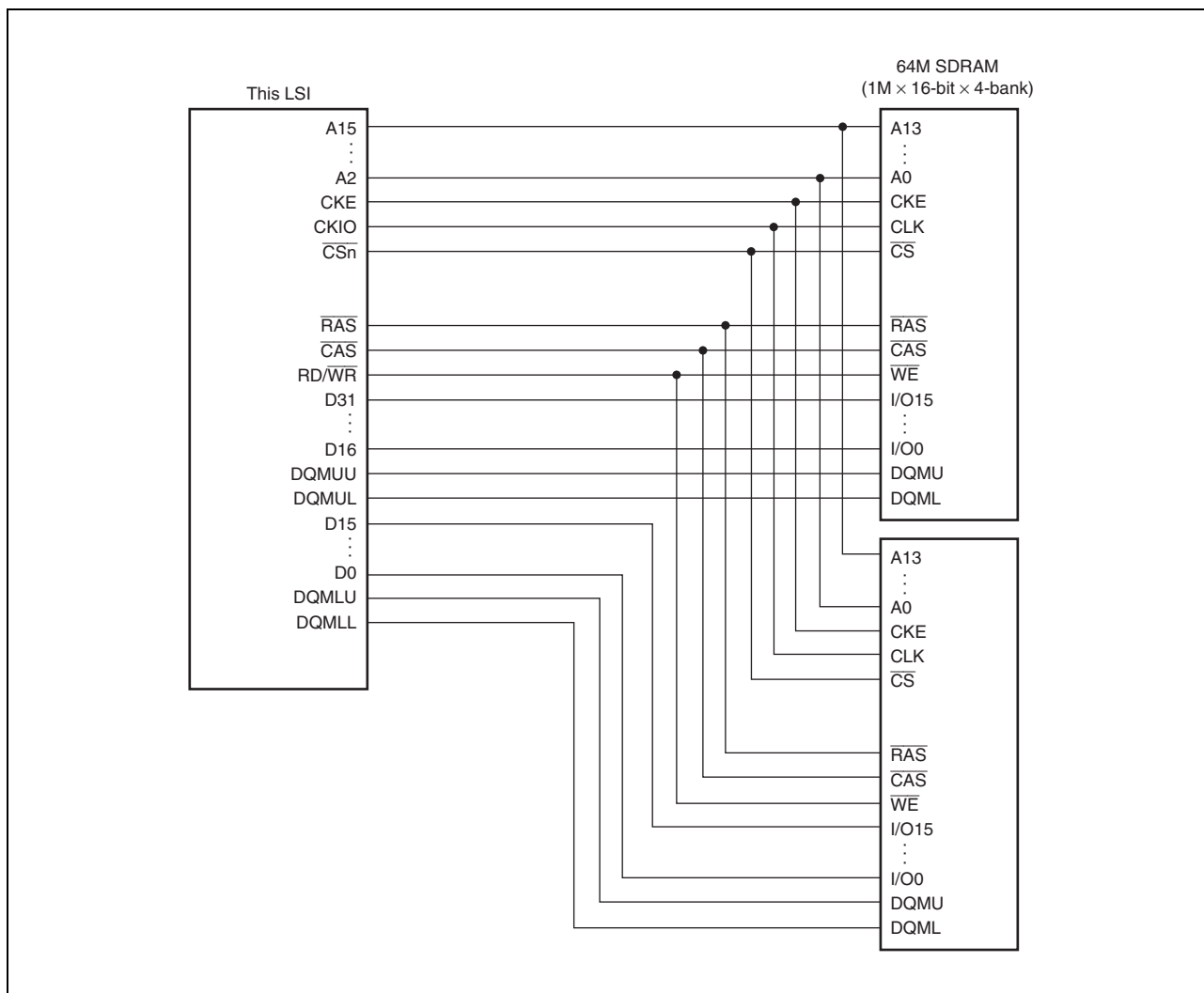


Figure 8.14 Example of 32-Bit Data Width SDRAM Connection

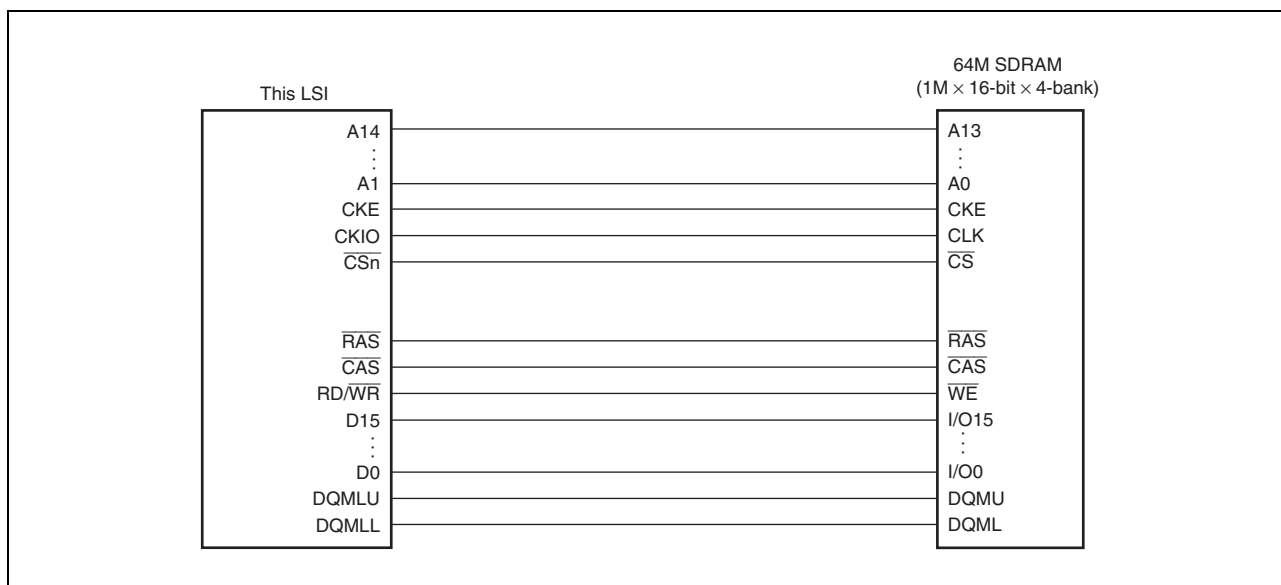


Figure 8.15 Example of 16-Bit Data Width SDRAM Connection

## (2) Address Multiplexing

An address multiplexing is specified so that SDRAM can be connected without external multiplexing circuitry according to the setting of bits BSZ[1:0] in CSnBCR and bits A2ROW[1:0], A2COL[1:0], A3ROW[1:0], and A3COL[1:0] in SDCR. Table 8.9 to Table 8.12 show the relationship between the settings of bits BSZ[1:0], A2ROW[1:0], A2COL[1:0], A3ROW[1:0], and A3COL[1:0] and the bits output at the address pins. Do not specify those bits in the manner other than this table, otherwise the operation of this LSI is not guaranteed. A25 to A18 are not multiplexed and the original values of address are always output at these pins.

When the data bus width is 16 bits (BSZ1 and BSZ0 = B'10), A0 of SDRAM specifies a 16-bit address. Therefore, connect this A0 pin of SDRAM to the A1 pin of the LSI; the A1 pin of SDRAM to the A2 pin of the LSI, and so on. When the data bus width is 32 bits (BSZ1 and BSZ0 = B'11), A0 of SDRAM specifies a 32-bit address. Therefore, connect this A0 pin of SDRAM to the A2 pin of the LSI; the A1 pin of SDRAM to the A3 pin of the LSI, and so on.

**Table 8.8 Relationship between BSZ[1:0], A2/3ROW[1:0], A2/3COL[1:0], and Address Multiplex Output (1)-1**

Setting			SDRAM Pin	Function
BSZ [1:0]	A2/3 ROW [1:0]	A2/3 COL [1:0]		
11 (32 bits)	00 (11 bits)	00 (8 bits)		
Output Pin of This LSI	Row Address Output Cycle	Column Address Output Cycle		
A17	A25	A17		Unused
A16	A24	A16		
A15	A23	A15		
A14	A22 <sup>*2</sup>	A22 <sup>*2</sup>	A12 (BA1)	Specifies bank
A13	A21 <sup>*2</sup>	A21 <sup>*2</sup>	A11 (BA0)	
A12	A20	L/H <sup>*1</sup>	A10/AP	Specifies address/ precharge
A11	A19	A11	A9	Address
A10	A18	A10	A8	
A9	A17	A9	A7	
A8	A16	A8	A6	
A7	A15	A7	A5	
A6	A14	A6	A4	
A5	A13	A5	A3	
A4	A12	A4	A2	
A3	A11	A3	A1	
A2	A10	A2	A0	
A1	A9	A1		Unused
A0	A8	A0		
Example of connected memory				
64-Mbit product (512 Kwords × 32 bits × 4 banks, column 8 bits product): 1				
16-Mbit product (512 Kwords × 16 bits × 2 banks, column 8 bits product): 2				

Note 1. L/H is a bit used in the command specification; it is fixed at L or H according to the access mode.

Note 2. Bank address specification

**Table 8.8 Relationship between BSZ[1:0], A2/3ROW[1:0], A2/3COL[1:0], and Address Multiplex Output (1)-2**

Setting			SDRAM Pin	Function
BSZ [1:0] 11 (32 bits)	A2/3 ROW [1:0] 01 (12 bits)	A2/3 COL [1:0] 00 (8 bits)		
Output Pin of This LSI	Row Address Output Cycle	Column Address Output Cycle		
A17	A25	A17		Unused
A16	A24	A16		
A15	A23*2	A23*2	A13 (BA1)	Specifies bank
A14	A22*2	A22*2	A12 (BA0)	
A13	A21	A13	A11	Address
A12	A20	L/H*1	A10/AP	Specifies address/ precharge
A11	A19	A11	A9	Address
A10	A18	A10	A8	
A9	A17	A9	A7	
A8	A16	A8	A6	
A7	A15	A7	A5	
A6	A14	A6	A4	
A5	A13	A5	A3	
A4	A12	A4	A2	
A3	A11	A3	A1	
A2	A10	A2	A0	
A1	A9	A1		Unused
A0	A8	A0		

Example of connected memory

128-Mbit product (1 Mwords × 32 bits × 4 banks, column 8 bits product): 1

64-Mbit product (1 Mwords × 16 bits × 4 banks, column 8 bits product): 2

Note 1. L/H is a bit used in the command specification; it is fixed at L or H according to the access mode.

Note 2. Bank address specification

**Table 8.9 Relationship between BSZ[1:0], A2/3ROW[1:0], A2/3COL[1:0], and Address Multiplex Output (2)-1**

Setting			SDRAM Pin	Function
BSZ [1:0] 11 (32 bits) Output Pin of This LSI	A2/3 ROW [1:0] 01 (12 bits) Row Address Output Cycle	A2/3 COL [1:0] 01 (9 bits) Column Address Output Cycle		
A17	A26	A17		Unused
A16	A25	A16		
A15	A24*2	A24*2	A13 (BA1)	Specifies bank
A14	A23*2	A23*2	A12 (BA0)	
A13	A22	A13	A11	Address
A12	A21	L/H*1	A10/AP	Specifies address/ precharge
A11	A20	A11	A9	Address
A10	A19	A10	A8	
A9	A18	A9	A7	
A8	A17	A8	A6	
A7	A16	A7	A5	
A6	A15	A6	A4	
A5	A14	A5	A3	
A4	A13	A4	A2	
A3	A12	A3	A1	
A2	A11	A2	A0	
A1	A10	A1		Unused
A0	A9	A0		
Example of connected memory				
256-Mbit product (2 Mwords × 32 bits × 4 banks, column 9 bits product): 1				
128-Mbit product (2 Mwords × 16 bits × 4 banks, column 9 bits product): 2				

Note 1. L/H is a bit used in the command specification; it is fixed at L or H according to the access mode.

Note 2. Bank address specification



**Table 8.9 Relationship between BSZ[1:0], A2/3ROW[1:0], A2/3COL[1:0], and Address Multiplex Output (2)-2**

Setting			SDRAM Pin	Function
BSZ [1:0] 11 (32 bits) Output Pin of This LSI	A2/3 ROW [1:0] 01 (12 bits) Row Address Output Cycle	A2/3 COL [1:0] 10 (10 bits) Column Address Output Cycle		
A17	A27	A17		Unused
A16	A26	A16		
A15	A25*2	A25*2	A13 (BA1)	Specifies bank
A14	A24*2	A24*2	A12 (BA0)	
A13	A23	A13	A11	Address
A12	A22	L/H*1	A10/AP	Specifies address/ precharge
A11	A21	A11	A9	Address
A10	A20	A10	A8	
A9	A19	A9	A7	
A8	A18	A8	A6	
A7	A17	A7	A5	
A6	A16	A6	A4	
A5	A15	A5	A3	
A4	A14	A4	A2	
A3	A13	A3	A1	
A2	A12	A2	A0	
A1	A11	A1		Unused
A0	A10	A0		
Example of connected memory				
512-Mbit product (4 Mwords × 32 bits × 4 banks, column 10 bits product): 1				
256-Mbit product (4 Mwords × 16 bits × 4 banks, column 10 bits product): 2				

Note 1. L/H is a bit used in the command specification; it is fixed at L or H according to the access mode.

Note 2. Bank address specification

**Table 8.10 Relationship between BSZ[1:0], A2/3ROW[1:0], A2/3COL[1:0], and Address Multiplex Output (3)**

Setting				
BSZ [1:0] 11 (32 bits)	A2/3 ROW [1:0] 10 (13 bits)	A2/3 COL [1:0] 01 (9 bits)		
Output Pin of This LSI	Row Address Output Cycle	Column Address Output Cycle	SDRAM Pin	Function
A17	A26	A17		Unused
A16	A25* <sup>2</sup>	A25* <sup>2</sup>	A14 (BA1)	Specifies bank
A15	A24* <sup>2</sup>	A24* <sup>2</sup>	A13 (BA0)	
A14	A23	A14	A12	Address
A13	A22	A13	A11	
A12	A21	L/H* <sup>1</sup>	A10/AP	Specifies address/ precharge
A11	A20	A11	A9	Address
A10	A19	A10	A8	
A9	A18	A9	A7	
A8	A17	A8	A6	
A7	A16	A7	A5	
A6	A15	A6	A4	
A5	A14	A5	A3	
A4	A13	A4	A2	
A3	A12	A3	A1	
A2	A11	A2	A0	
A1	A10	A1		Unused
A0	A9	A0		
Example of connected memory				
512-Mbit product (4 Mwords × 32 bits × 4 banks, column 9 bits product): 1				
256-Mbit product (4 Mwords × 16 bits × 4 banks, column 9 bits product): 2				

Note 1. L/H is a bit used in the command specification; it is fixed at L or H according to the access mode.

Note 2. Bank address specification

**Table 8.11 Relationship between BSZ[1:0], A2/3ROW[1:0], A2/3COL[1:0], and Address Multiplex Output (4)-1**

Setting				
BSZ [1:0] 10 (16 bits)	A2/3 ROW [1:0] 00 (11 bits)	A2/3 COL [1:0] 00 (8 bits)		
Output Pin of This LSI	Row Address Output Cycle	Column Address Output Cycle	SDRAM Pin	Function
A17	A25	A17		Unused
A16	A24	A16		
A15	A23	A15		
A14	A22	A14		
A13	A21	A21		
A12	A20 <sup>*2</sup>	A20 <sup>*2</sup>	A11 (BA0)	Specifies bank
A11	A19	L/H <sup>*1</sup>	A10/AP	Specifies address/ precharge
A10	A18	A10	A9	Address
A9	A17	A9	A8	
A8	A16	A8	A7	
A7	A15	A7	A6	
A6	A14	A6	A5	
A5	A13	A5	A4	
A4	A12	A4	A3	
A3	A11	A3	A2	
A2	A10	A2	A1	
A1	A9	A1	A0	
A0	A8	A0		Unused
Example of connected memory				
16-Mbit product (512 Kwords × 16 bits × 2 banks, column 8 bits product): 1				

Note 1. L/H is a bit used in the command specification; it is fixed at L or H according to the access mode.

Note 2. Bank address specification

**Table 8.11 Relationship between BSZ[1:0], A2/3ROW[1:0], A2/3COL[1:0], and Address Multiplex Output (4)-2**

Setting			SDRAM Pin	Function
BSZ [1:0] 10 (16 bits) Output Pin of This LSI	A2/3 ROW [1:0] 01 (12 bits) Row Address Output Cycle	A2/3 COL [1:0] 00 (8 bits) Column Address Output Cycle		
A17	A25	A17		Unused
A16	A24	A16		
A15	A23	A15		
A14	A22*2	A22*2	A13 (BA1)	Specifies bank
A13	A21*2	A21*2	A12 (BA0)	
A12	A20	A12	A11	Address
A11	A19	L/H*1	A10/AP	Specifies address/ precharge
A10	A18	A10	A9	Address
A9	A17	A9	A8	
A8	A16	A8	A7	
A7	A15	A7	A6	
A6	A14	A6	A5	
A5	A13	A5	A4	
A4	A12	A4	A3	
A3	A11	A3	A2	
A2	A10	A2	A1	
A1	A9	A1	A0	
A0	A8	A0		Unused
Example of connected memory				
64-Mbit product (1 Mwords × 16 bits × 4 banks, column 8 bits product): 1				

Note 1. L/H is a bit used in the command specification; it is fixed at L or H according to the access mode.

Note 2. Bank address specification

**Table 8.12 Relationship between BSZ[1:0], A2/3ROW[1:0], A2/3COL[1:0], and Address Multiplex Output (5)-1**

Setting			SDRAM Pin	Function
BSZ [1:0] 10 (16 bits) Output Pin of This LSI	A2/3 ROW [1:0] 01 (12 bits) Row Address Output Cycle	A2/3 COL [1:0] 01 (9 bits) Column Address Output Cycle		
A17	A26	A17		Unused
A16	A25	A16		
A15	A24	A15		
A14	A23 <sup>*2</sup>	A23 <sup>*2</sup>	A13 (BA1)	Specifies bank
A13	A22 <sup>*2</sup>	A22 <sup>*2</sup>	A12 (BA0)	
A12	A21	A12	A11	Address
A11	A20	L/H <sup>*1</sup>	A10/AP	Specifies address/ precharge
A10	A19	A10	A9	Address
A9	A18	A9	A8	
A8	A17	A8	A7	
A7	A16	A7	A6	
A6	A15	A6	A5	
A5	A14	A5	A4	
A4	A13	A4	A3	
A3	A12	A3	A2	
A2	A11	A2	A1	
A1	A10	A1	A0	
A0	A9	A0		Unused
Example of connected memory				
128-Mbit product (2 Mwords × 16 bits × 4 banks, column 9 bits product): 1				

Note 1. L/H is a bit used in the command specification; it is fixed at L or H according to the access mode.

Note 2. Bank address specification

**Table 8.12 Relationship between BSZ[1:0], A2/3ROW[1:0], A2/3COL[1:0], and Address Multiplex Output (5)-2**

Setting			SDRAM Pin	Function
BSZ [1:0] 10 (16 bits) Output Pin of This LSI	A2/3 ROW [1:0] 01 (12 bits) Row Address Output Cycle	A2/3 COL [1:0] 10 (10 bits) Column Address Output Cycle		
A17	A27	A17		Unused
A16	A26	A16		
A15	A25	A15		
A14	A24*2	A24*2	A13 (BA1)	Specifies bank
A13	A23*2	A23*2	A12 (BA0)	
A12	A22	A12	A11	Address
A11	A21	L/H*1	A10/AP	Specifies address/ precharge
A10	A20	A10	A9	Address
A9	A19	A9	A8	
A8	A18	A8	A7	
A7	A17	A7	A6	
A6	A16	A6	A5	
A5	A15	A5	A4	
A4	A14	A4	A3	
A3	A13	A3	A2	
A2	A12	A2	A1	
A1	A11	A1	A0	
A0	A10	A0		Unused
Example of connected memory				
256-Mbit product (4 Mwords × 16 bits × 4 banks, column 10 bits product): 1				

Note 1. L/H is a bit used in the command specification; it is fixed at L or H according to the access mode.

Note 2. Bank address specification

**Table 8.13 Relationship between BSZ[1:0], A2/3ROW[1:0], A2/3COL[1:0], and Address Multiplex Output (6)-1**

Setting				
BSZ [1:0] 10 (16 bits)	A2/3 ROW [1:0] 10 (13 bits)	A2/3 COL [1:0] 01 (9 bits)		
Output Pin of This LSI	Row Address Output Cycle	Column Address Output Cycle	SDRAM Pin	Function
A17	A26	A17		Unused
A16	A25	A16		
A15	A24*2	A24*2	A14 (BA1)	Specifies bank
A14	A23*2	A23*2	A13 (BA0)	
A13	A22	A13	A12	Address
A12	A21	A12	A11	
A11	A20	L/H*1	A10/AP	Specifies address/ precharge
A10	A19	A10	A9	Address
A9	A18	A9	A8	
A8	A17	A8	A7	
A7	A16	A7	A6	
A6	A15	A6	A5	
A5	A14	A5	A4	
A4	A13	A4	A3	
A3	A12	A3	A2	
A2	A11	A2	A1	
A1	A10	A1	A0	
A0	A9	A0		Unused
Example of connected memory				
256-Mbit product (4 Mwords ´ 16 bits ´ 4 banks, column 9 bits product): 1				

Note 1. L/H is a bit used in the command specification; it is fixed at L or H according to the access mode.

Note 2. Bank address specification

**Table 8.13 Relationship between BSZ[1:0], A2/3ROW[1:0], A2/3COL[1:0], and Address Multiplex Output (6)-2**

Setting			SDRAM Pin	Function
BSZ [1:0] 10 (16 bits) Output Pin of This LSI	A2/3 ROW [1:0] 10 (13 bits) Row Address Output Cycle	A2/3 COL [1:0] 10 (10 bits) Column Address Output Cycle		
A17	A27	A17		Unused
A16	A26	A16		
A15	A25*2	A25*2	A14 (BA1)	Specifies bank
A14	A24*2	A24*2	A13 (BA0)	
A13	A23	A13	A12	Address
A12	A22	A12	A11	
A11	A21	L/H*1	A10/AP	Specifies address/ precharge
A10	A20	A10	A9	Address
A9	A19	A9	A8	
A8	A18	A8	A7	
A7	A17	A7	A6	
A6	A16	A6	A5	
A5	A15	A5	A4	
A4	A14	A4	A3	
A3	A13	A3	A2	
A2	A12	A2	A1	
A1	A11	A1	A0	
A0	A10	A0		Unused
Example of connected memory				
512-Mbit product (8 Mwords × 16 bits × 4 banks, column 10 bits product): 1				

Note 1. L/H is a bit used in the command specification; it is fixed at L or H according to the access mode.

Note 2. Bank address specification



### (3) Burst Read

A burst read occurs in the following cases with this LSI.

- Access size in reading is larger than data bus width.
- 16-, 32- or 64-byte transfer

This LSI always accesses the SDRAM with burst length 1. For example, read access of burst length 1 is performed consecutively 8 times to read 16-byte continuous data from the SDRAM that is connected to a 16-bit data bus. This access is called the burst read with the burst number 8. Table 8.11 shows the relationship between the access size and the number of bursts.

**Table 8.14 Relationship between Access Size and Number of Bursts**

Bus Width	Access Size	Number of Bursts
16 bits	8 bits	1
	16 bits	1
	32 bits	2
	16 bytes	8
	32 bytes	16
	64 bytes	32
32 bits	8 bits	1
	16 bits	1
	32 bits	1
	16 bytes	4
	32 bytes	8
	64 bytes	16

Figure 8.16 and Figure 8.17 show timing charts in burst read. In burst read, an ACTV command is output in the Tr cycle, the READ command is issued in the Tc1, Tc2, and Tc3 cycles, the READA command is issued in the Tc4 cycle, and the read data is received at the rising edge of the external clock (CKIO) in the Td1 to Td4 cycles. The Tap cycle is used to wait for the completion of an auto-precharge induced by the READA command in the SDRAM. In the Tap cycle, a new command will not be issued to the same bank. However, access to another CS space or another bank in the same SDRAM space is enabled. The number of Tap cycles is specified by the WTRP1 and WTRP0 bits in CS3WCR. In this LSI, wait cycles can be inserted by specifying each bit in CS3WCR to connect the SDRAM in variable frequencies. Figure 8.17 shows an example in which wait cycles are inserted. The number of cycles from the Tr cycle where the ACTV command is output to the Tc1 cycle where the READ command is output can be specified using the WTRCD1 and WTRCD0 bits in CS3WCR. If the WTRCD1 and WTRCD0 bits specify one cycle or more, a Trw cycle where the NOP command is issued is inserted between the Tr cycle and Tc1 cycle. The number of cycles from the Tc1 cycle where the READ command is output to the Td1 cycle where the read data is latched can be specified for the CS2 and CS3 spaces independently, using the A2CL1 and A2CL0 bits in CS2WCR or the A3CL1 and A3CL0 bits in CS3WCR. The number of cycles from Tc1 to Td1 corresponds to the SDRAM CAS latency. The CAS latency for the SDRAM is normally defined as up to three cycles. However, the CAS latency in this LSI can be specified as 1 to 4 cycles. This CAS latency can be achieved by connecting a latch circuit between this LSI and the SDRAM. A Tde cycle is an idle cycle required to transfer the read data into this LSI and occurs once for every burst read or every single read.

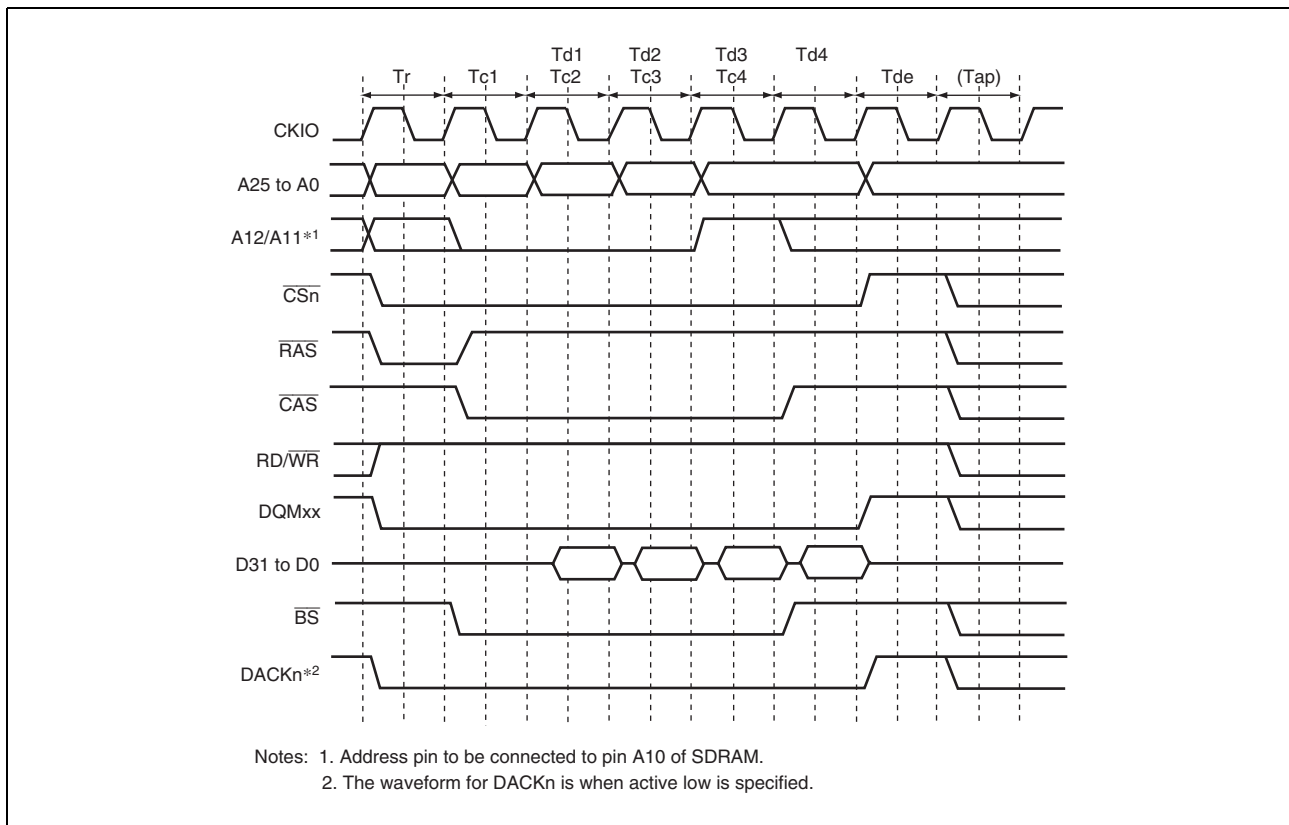


Figure 8.16 Burst Read Basic Timing (CAS Latency 1, Auto Pre-Charge)

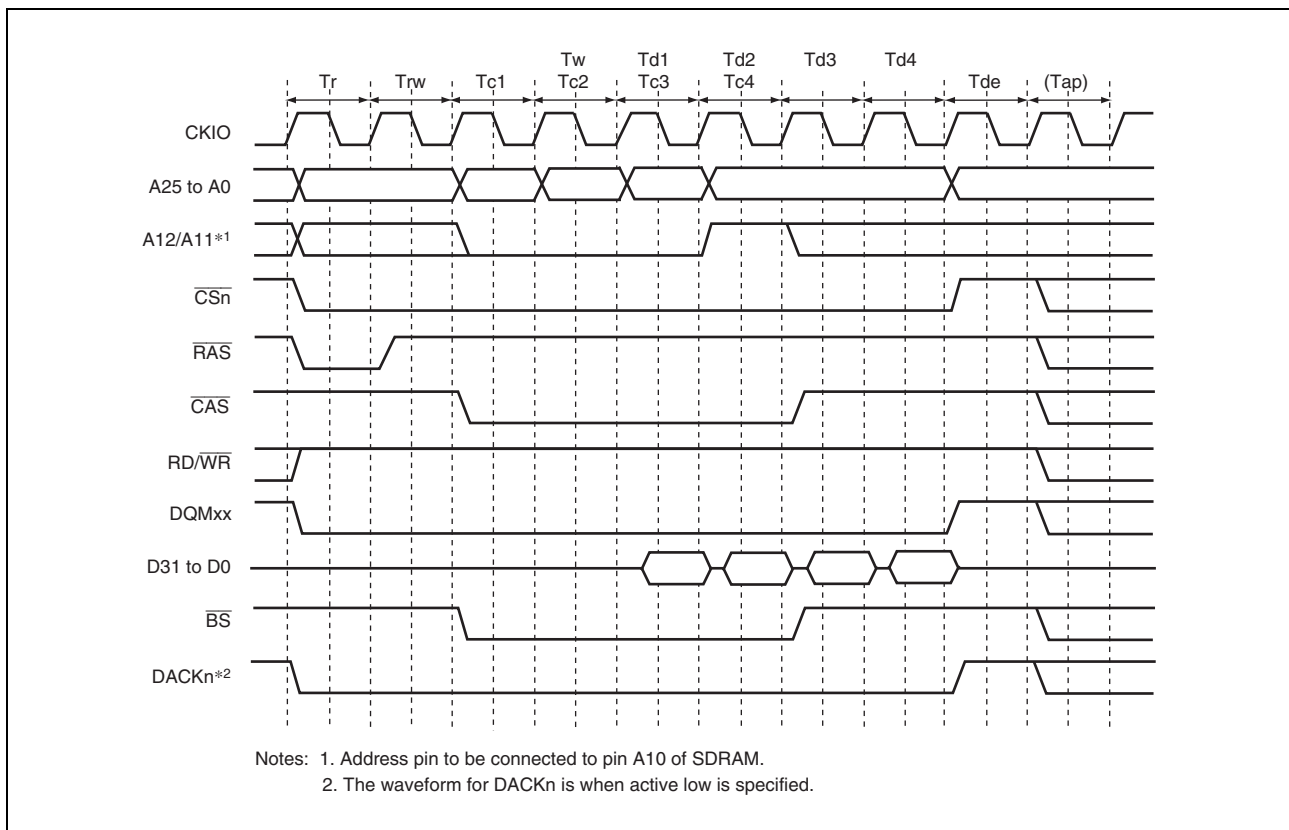
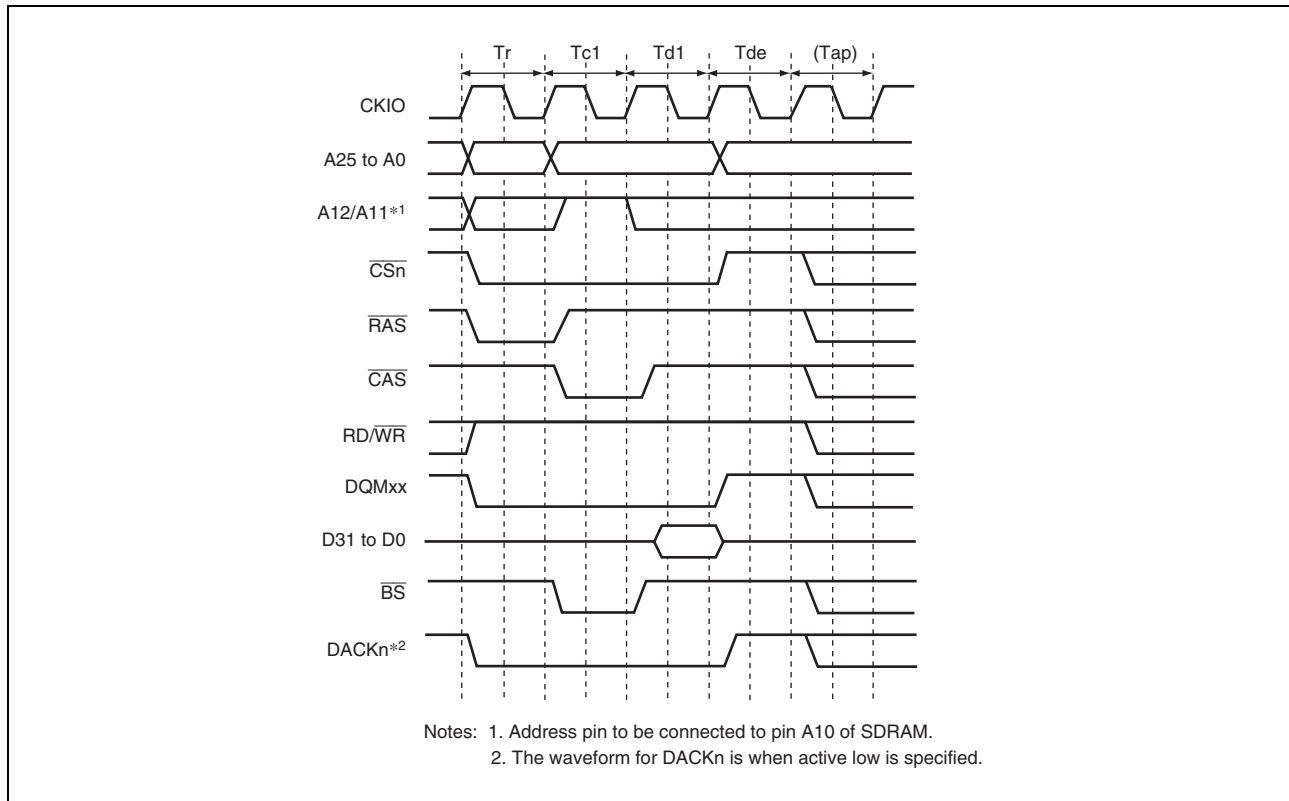


Figure 8.17 Burst Read Wait Specification Timing (CAS Latency 2, WTRCD[1:0] = 1 Cycle, Auto Pre-Charge)

#### (4) Single Read

A read access ends in one cycle when the data bus width is larger than or equal to the access size. As the SDRAM is set to the burst read with the burst length 1, only the required data is output. A read access that ends in one cycle is called single read.

Figure 8.18 shows the single read basic timing.



**Figure 8.18 Basic Timing for Single Read (CAS Latency 1, Auto Pre-Charge)**

### (5) Burst Write

A burst write occurs in the following cases in this LSI.

- Access size in writing is larger than data bus width.
- 16-, 32- or 64-byte transfer

This LSI always accesses SDRAM with burst length 1. For example, write access of burst length 1 is performed continuously 8 times to write 16-byte continuous data to the SDRAM that is connected to a 16-bit data bus. This access is called burst write with the burst number 8. The relationship between the access size and the number of bursts is shown in Table 8.13. Figure 8.19 shows a timing chart for burst writes. In burst write, an ACTV command is output in the Tr cycle, the WRIT command is issued in the Tc1, Tc2, and Tc3 cycles, and the WRITA command is issued to execute an auto-precharge in the Tc4 cycle. In the write cycle, the write data is output simultaneously with the write command. After the write command with the auto-precharge is output, the Trw1 cycle that waits for the auto-precharge initiation is followed by the Tap cycle that waits for completion of the auto-precharge induced by the WRITA command in the SDRAM. Between the Trw1 and the Tap cycle, a new command will not be issued to the same bank. However, access to another CS space or another bank in the same SDRAM space is enabled. The number of Trw1 cycles is specified by the TRWL1 and TRWL0 bits in CS3WCR. The number of Tap cycles is specified by the WTRP1 and WTRP0 bits in CS3WCR.

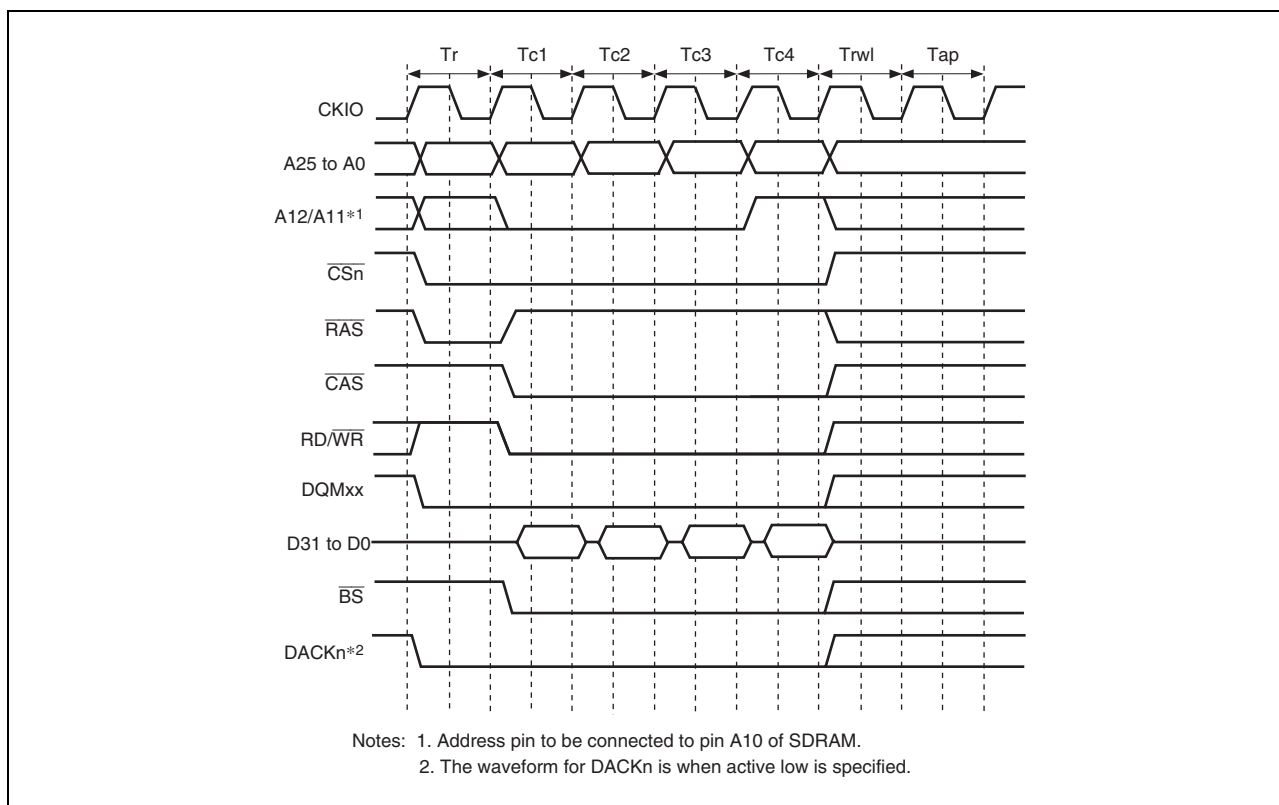
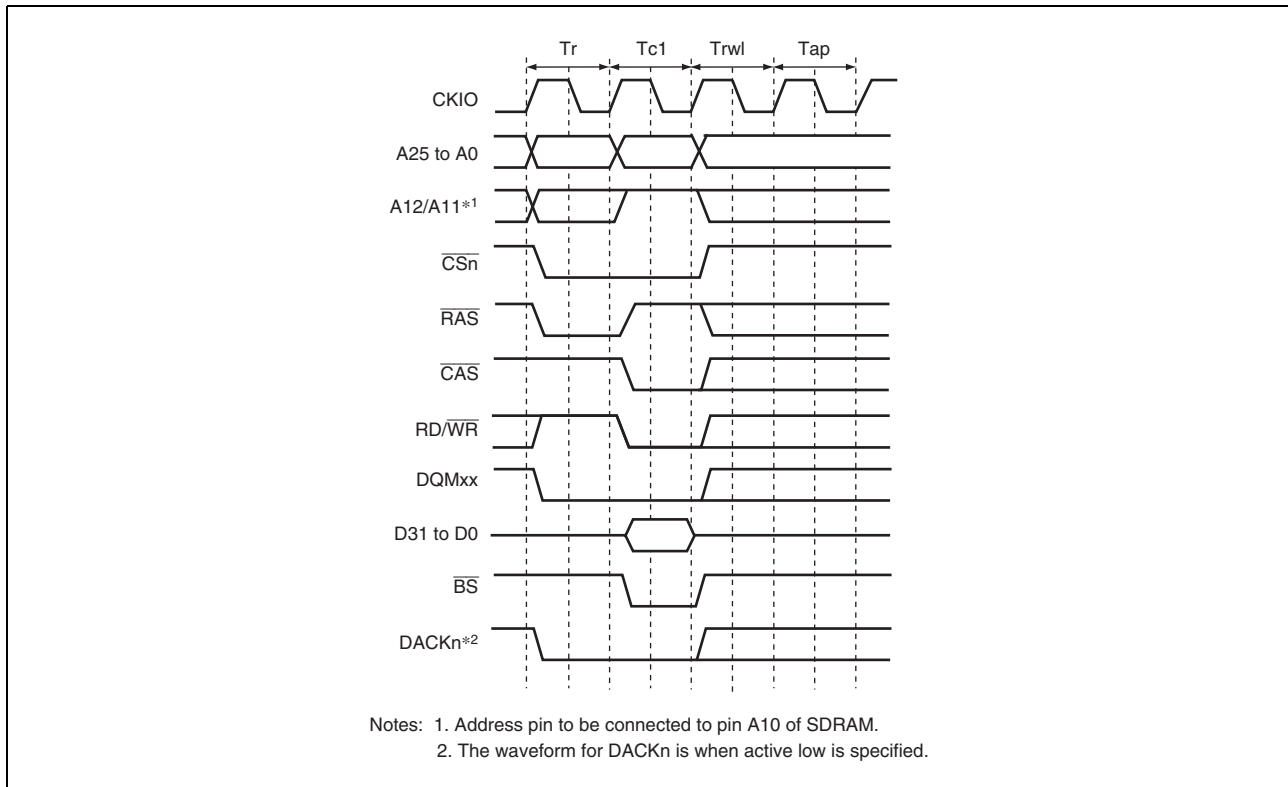


Figure 8.19 Basic Timing for Burst Write (Auto Pre-Charge)

### (6) Single Write

A write access ends in one cycle when the data bus width is larger than or equal to access size. As a single write or burst write with burst length 1 is set in SDRAM, only the required data is output. The write access that ends in one cycle is called single write. Figure 8.20 shows the single write basic timing.



**Figure 8.20 Single Write Basic Timing (Auto-Precharge)**

## (7) Bank Active

The SDRAM bank function can be used to support high-speed access to the same row address. When the BACTV bit in SDCR is 1, access is performed using commands without auto-precharge (READ or WRIT). This function is called bank-active function. This function is valid only for area 3. When area 3 is set to bank-active mode, area 2 should be set to normal space or SRAM with byte selection. When areas 2 and 3 are both set to SDRAM, auto precharge mode must be set.

When the bank-active function is used, precharging is not performed when the access ends. When accessing the same row address in the same bank, it is possible to issue the READ or WRIT command immediately, without issuing an ACTV command. As SDRAM is internally divided into several banks, it is possible to activate one row address in each bank. If the next access is to a different row address, a PRE command is first issued to precharge the relevant bank, then when precharging is completed, the access is performed by issuing an ACTV command followed by a READ or WRIT command. If this is followed by an access to a different row address, the access time will be longer because of the precharging performed after the access request is issued. The number of cycles between issuance of the PRE command and the ACTV command is determined by the WTRP1 and WTPR0 bits in CS3WCR.

In a write, when an auto-precharge is performed, a command cannot be issued to the same bank for a period of  $Trwl + Tap$  cycles after issuance of the WRITA command. When bank active mode is used, READ or WRIT commands can be issued successively if the row address is the same. The number of cycles can thus be reduced by  $Trwl + Tap$  cycles for each write.

There is a limit on tRAS, the time for placing each bank in the active state. If there is no guarantee that there will not be a cache hit and another row address will be accessed within the period in which this value is maintained by program execution, it is necessary to set auto-refresh and set the refresh cycle to no more than the maximum value of tRAS.

A burst read cycle without auto-precharge is shown in Figure 8.21, a burst read cycle for the same row address in Figure 8.22, and a burst read cycle for different row addresses in Figure 8.23. Similarly, a single write cycle without auto-precharge is shown in Figure 8.24, a single write cycle for the same row address in Figure 8.25, and a single write cycle for different row addresses in Figure 8.26.

In Figure 8.22, a Tnop cycle in which no operation is performed is inserted before the Tc cycle that issues the READ command. The Tnop cycle is inserted to acquire two cycles of CAS latency for the DQMxx signal that specifies the read byte in the data read from the SDRAM. If the CAS latency is specified as two cycles or more, the Tnop cycle is not inserted because the two cycles of latency can be acquired even if the DQMxx signal is asserted after the Tc cycle. When bank active mode is set, if only access cycles to the respective banks in the area 3 space are considered, as long as access cycles to the same row address continue, the operation starts with the cycle in Figure 8.21 or Figure 8.24, followed by repetition of the cycle in Figure 8.22 or Figure 8.25. An access to a different area during this time has no effect. If there is an access to a different row address in the bank active state, the bus cycle in Figure 8.23 or Figure 8.26 is executed instead of that in Figure 8.22 or Figure 8.25. In bank active mode, too, all banks become inactive after a refresh cycle.

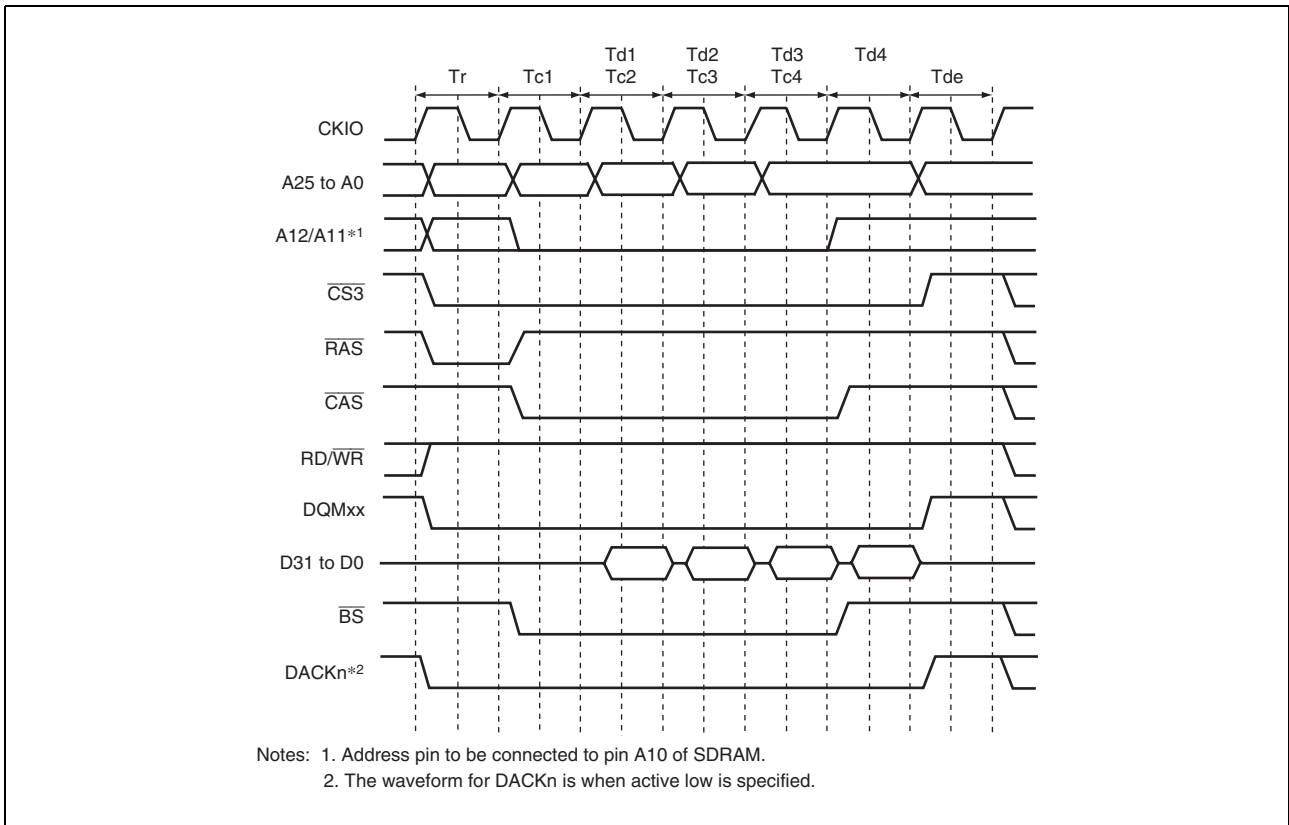


Figure 8.21 Burst Read Timing (Bank Active, Different Bank, CAS Latency 1)

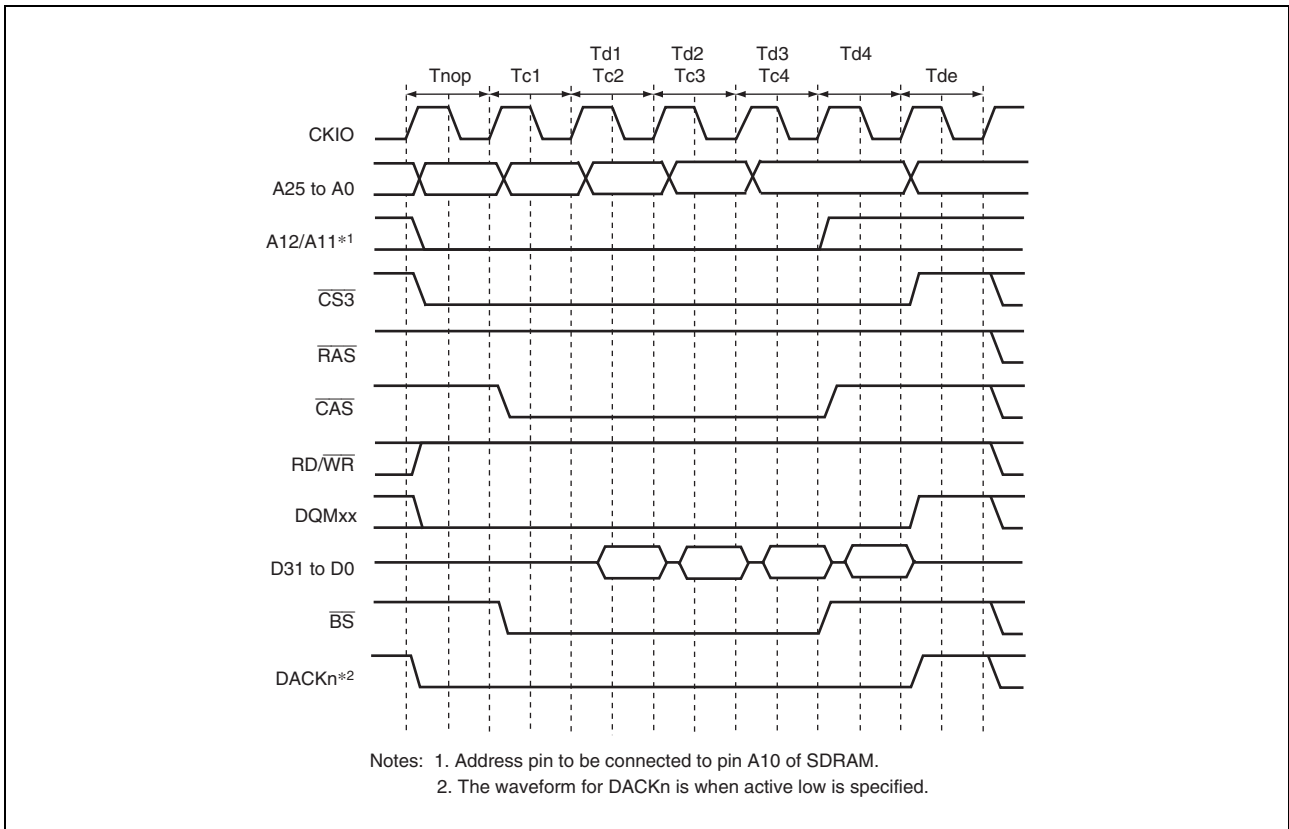
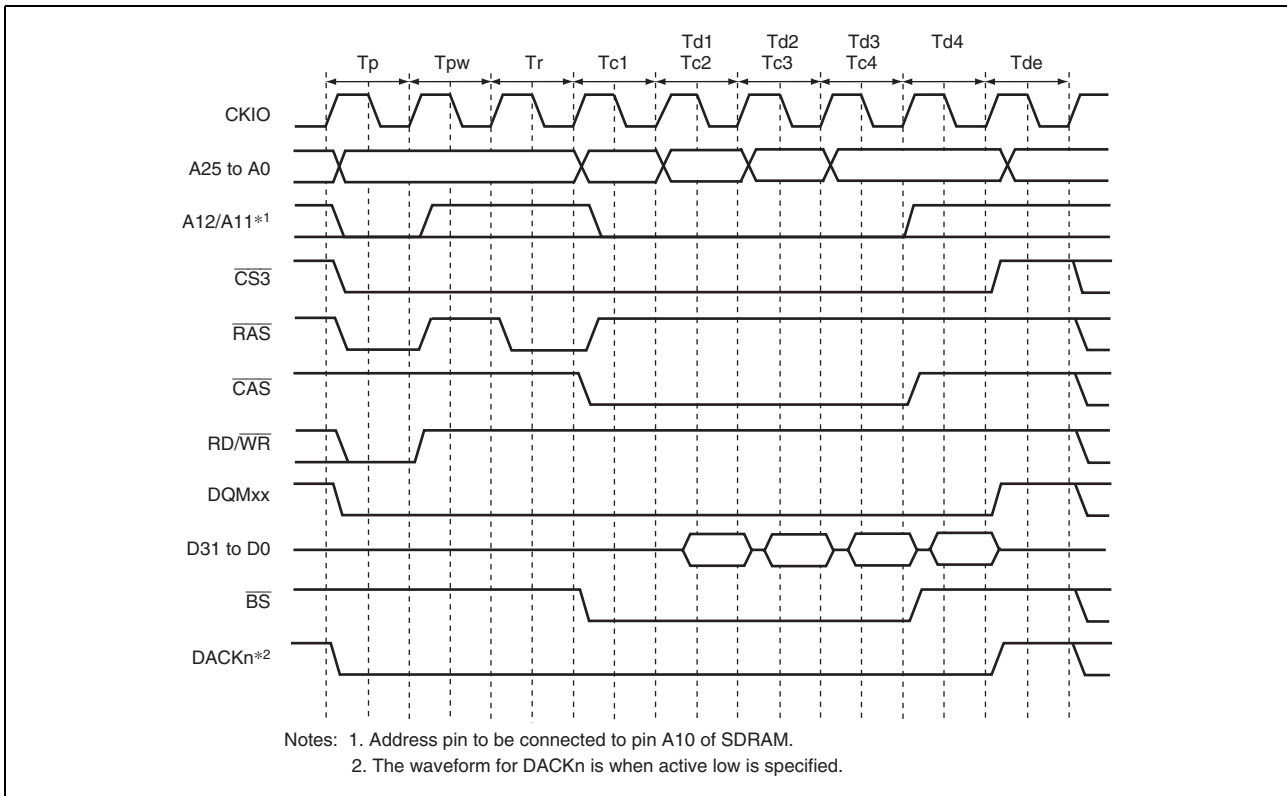
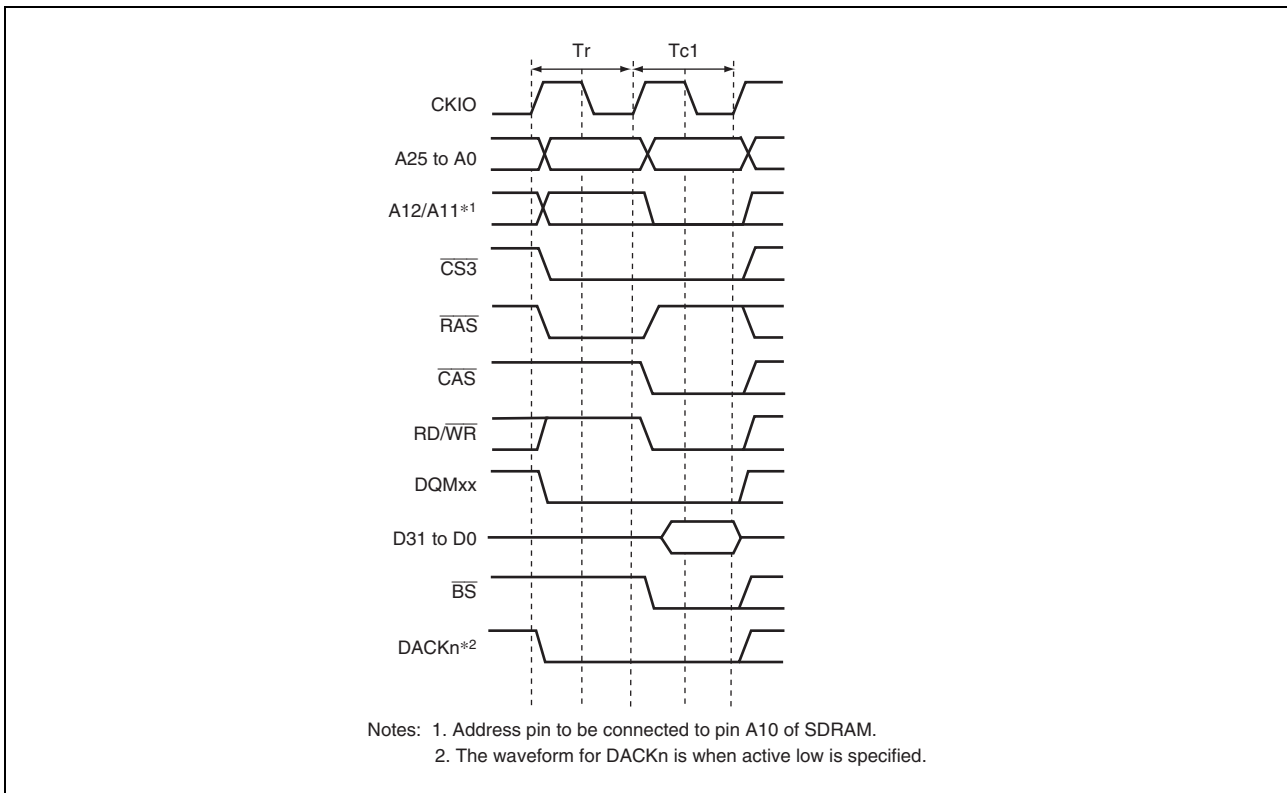


Figure 8.22 Burst Read Timing (Bank Active, Same Row Addresses in the Same Bank, CAS Latency 1)



**Figure 8.23 Burst Read Timing (Bank Active, Different Row Addresses in the Same Bank, CAS Latency 1)**



**Figure 8.24 Single Write Timing (Bank Active, Different Bank)**



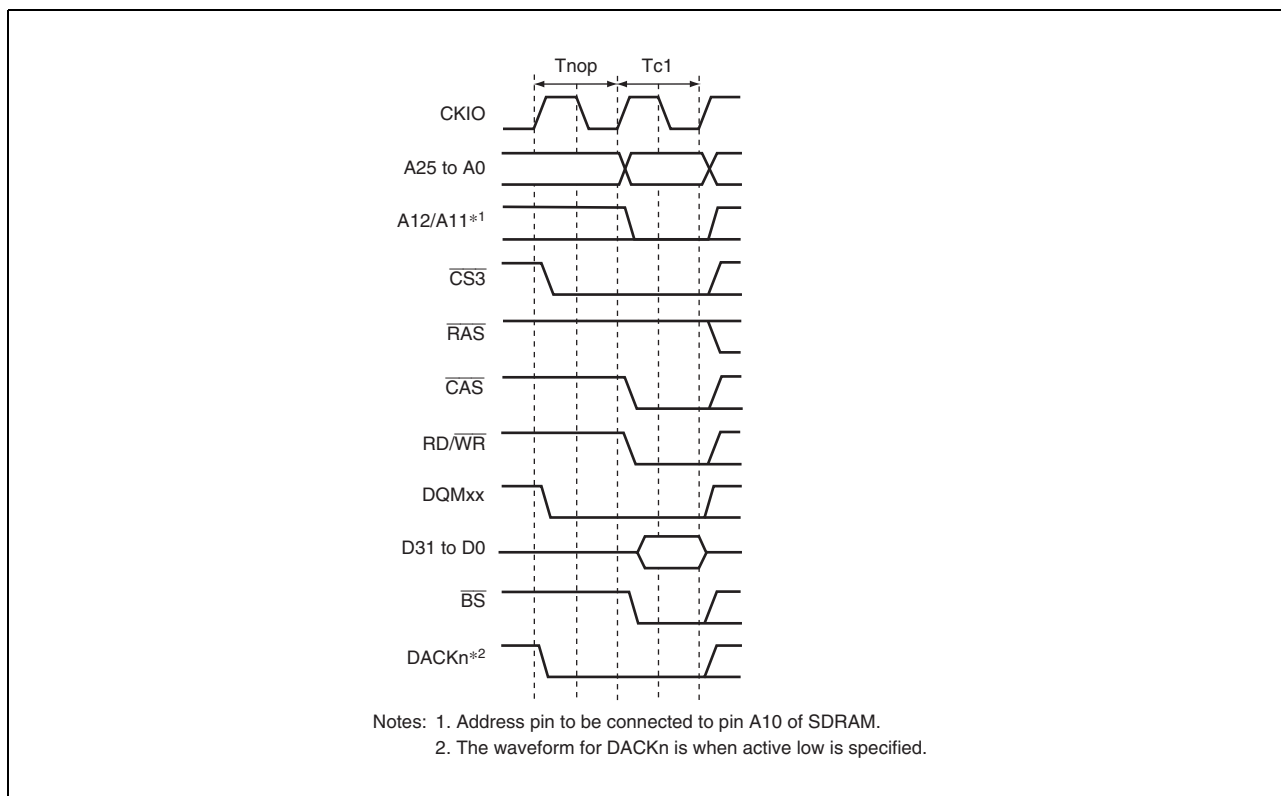


Figure 8.25 Single Write Timing (Bank Active, Same Row Addresses in the Same Bank)

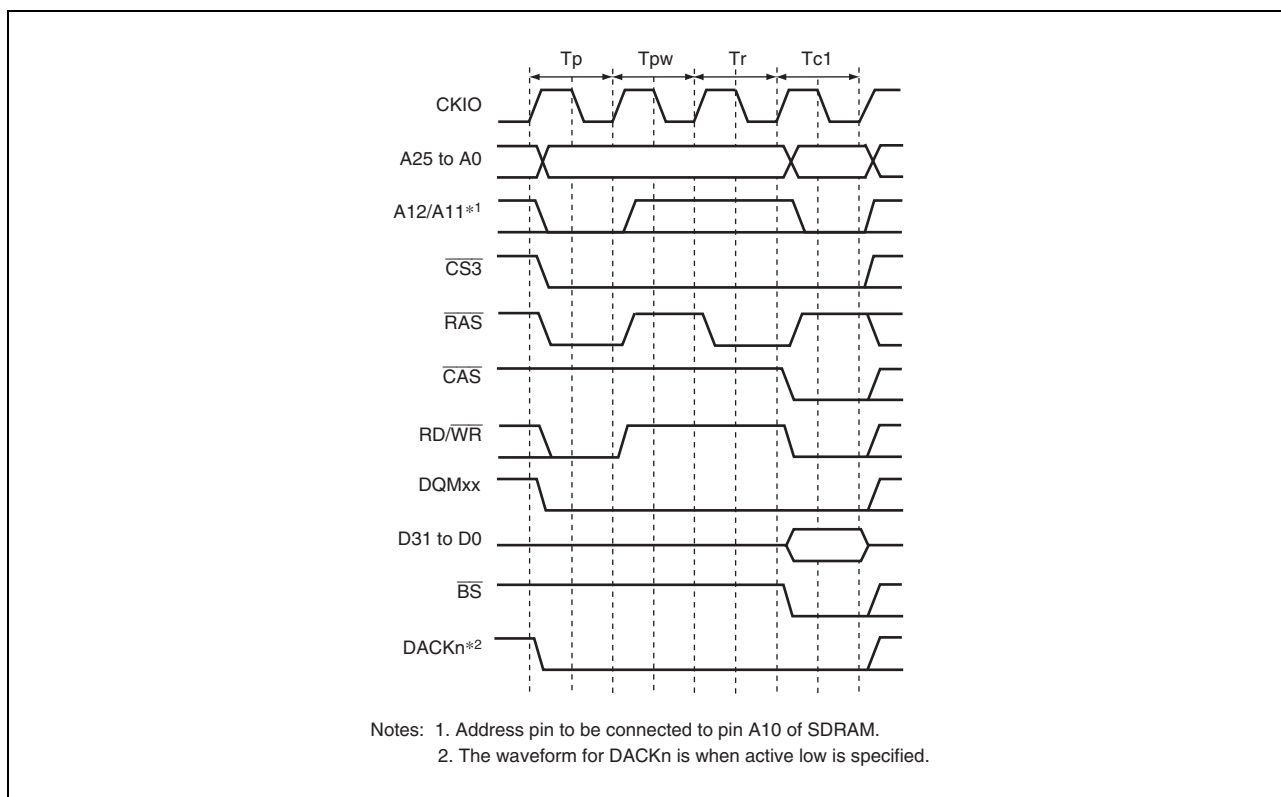


Figure 8.26 Single Write Timing (Bank Active, Different Row Addresses in the Same Bank)

## (8) Refreshing

This module has a function for controlling SDRAM refreshing. Auto-refreshing can be performed by clearing the RMODE bit to 0 and setting the RFSH bit to 1 in SDCR. A continuous refreshing can be performed by setting the RRC2 to RRC0 bits in RTCSR. If SDRAM is not accessed for a long period, self-refresh mode, in which the power consumption for data retention is low, can be activated by setting both the RMODE bit and the RFSH bit to 1.

### (a) Auto-refreshing

Refreshing is performed at intervals determined by the input clock selected by bits CKS2 to CKS0 in RTCSR, and the value set by in RTCOR. The value of bits CKS2 to CKS0 in RTCOR should be set so as to satisfy the refresh interval stipulation for the SDRAM used. First make the settings for RTCOR, RTCNT, and the RMODE and RFSH bits in SDCR, and then make the CKS2 to CKS0 and RRC2 to RRC0 settings. When the clock is selected by bits CKS2 to CKS0, RTCNT starts counting up from the value at that time. The RTCNT value is constantly compared with the RTCOR value, and if the two values are the same, a refresh request is generated and an auto-refresh is performed for the number of times specified by the RRC2 to RRC0. At the same time, RTCNT is cleared to zero and the count-up is restarted.

Figure 8.27 shows the auto-refresh cycle timing. After starting the auto refreshing, PALL command is issued in the  $T_p$  cycle to make all the banks to pre-charged state from active state when some bank is being pre-charged. Then REF command is issued in the  $T_{rr}$  cycle after inserting idle cycles of which number is specified by the WTRP1 and WTRP0 bits in CS3WCR. A new command is not issued for the duration of the number of cycles specified by the WTRC1 and WTRC0 bits in CS3WCR after the  $T_{rr}$  cycle. The WTRC1 and WTRC0 bits must be set so as to satisfy the SDRAM refreshing cycle time stipulation ( $t_{RC}$ ). An idle cycle is inserted between the  $T_p$  cycle and  $T_{rr}$  cycle when the setting value of the WTRP1 and WTRP0 bits in CS3WCR is longer than or equal to 1 cycle.

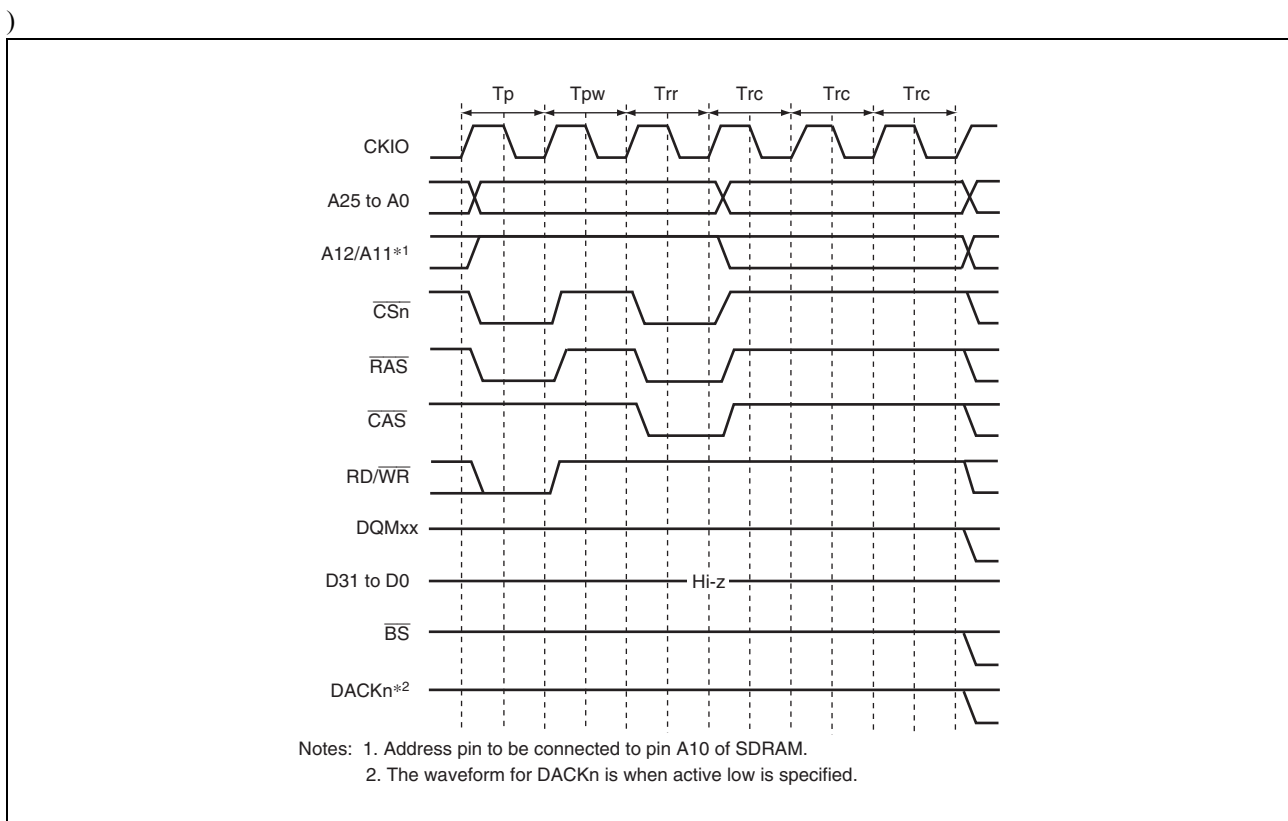


Figure 8.27 Auto-Refresh Timing

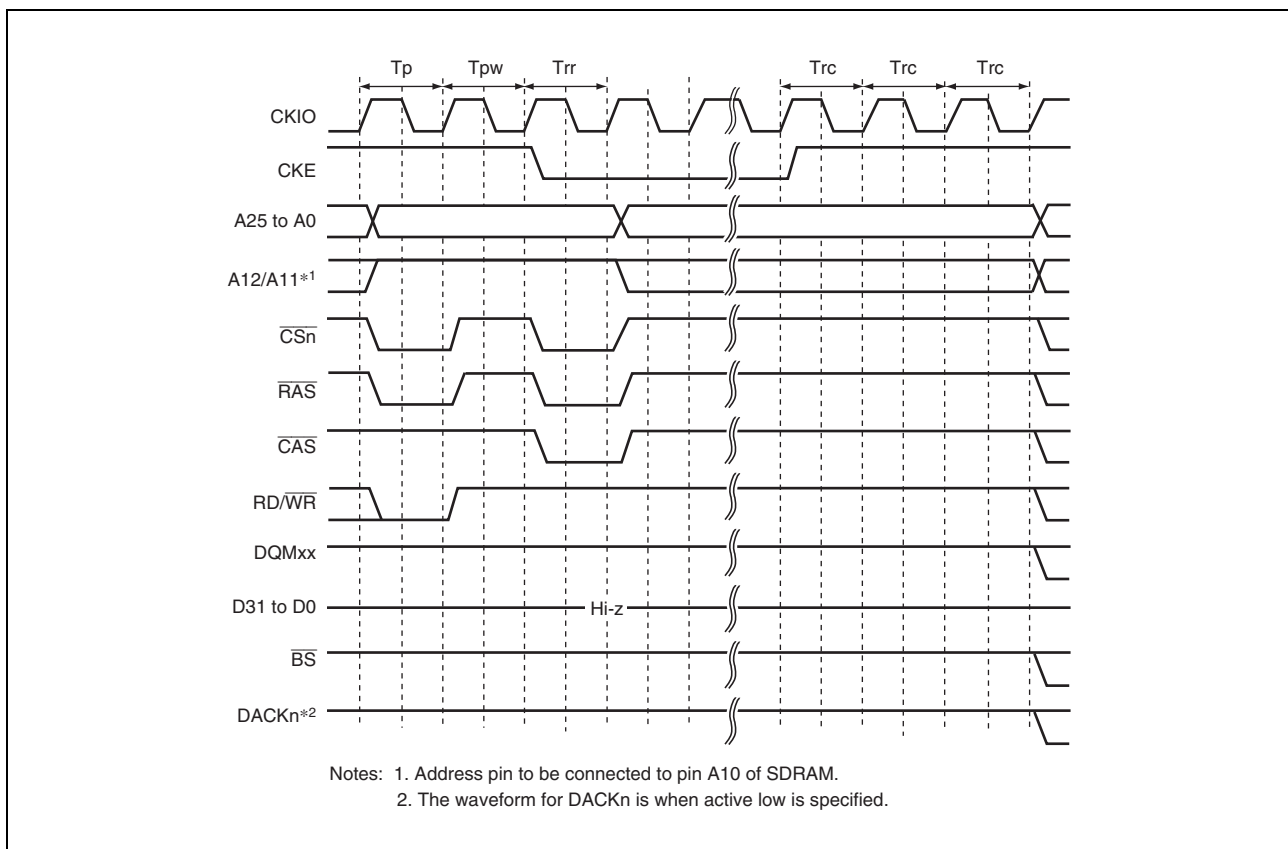
**(b) Self-refreshing**

Self-refresh mode is a kind of standby mode, in which the refresh timing and refresh addresses are generated within the SDRAM. Self-refreshing is activated by setting both the RMODE bit and the RFSH bit in SDCR to 1. After starting the self-refreshing, PALL command is issued in  $T_p$  cycle after the completion of the pre-charging bank. A SELF command is then issued after inserting idle cycles of which number is specified by the WTRP1 and WTRP0 bits in CS3WSR. SDRAM cannot be accessed while in the self-refresh state. Self-refresh mode is cleared by clearing the RMODE bit to 0. After self-refresh mode has been cleared, command issuance is disabled for the number of cycles specified by the WTRC1 and WTRC0 bits in CS3WCR.

Self-refresh timing is shown in figure 8.28. Settings must be made so that self-refresh clearing and data retention are performed correctly, and auto-refreshing is performed at the correct intervals. When self-refreshing is activated from the state in which auto-refreshing is set, auto-refreshing is restarted if the RFSH bit is set to 1 and the RMODE bit is cleared to 0 when self-refresh mode is cleared. If the transition from clearing of self-refresh mode to the start of auto-refreshing takes time, this time should be taken into consideration when setting the initial value of RTCNT. Making the RTCNT value 1 less than the RTCOR value will enable refreshing to be started immediately.

After self-refreshing has been set, the self-refresh state continues even if the chip standby state is entered using the LSI standby function, and is maintained even after recovery from standby mode due to an interrupt. Note that the necessary signals such as CKE must be driven even in standby state by setting the HIZCNT bit in CMNCR to 1.

In case of a power-on reset, the bus state controller's registers are initialized, and therefore the self-refresh state is cleared.



**Figure 8.28 Self-Refresh Timing**

### (9) Relationship between Refresh Requests and Bus Cycles

If a refresh request occurs during bus cycle execution, the refresh cycle must wait for the bus cycle to be completed. If a new refresh request occurs while waiting for the previous refresh request, the previous refresh request is deleted. To refresh correctly, a bus cycle longer than the refresh interval must be prevented from occurring.

### (10) Power-Down Mode

If the PDOWN bit in SDCR is set to 1, the SDRAM is placed in power-down mode by bringing the CKE signal to the low level in the non-access cycle. This power-down mode can effectively lower the power consumption in the non-access cycle. However, please note that if an access occurs in power-down mode, a cycle of overhead occurs because a cycle is needed to assert the CKE in order to cancel the power-down mode.

Figure 8.29 shows the access timing in power-down mode.

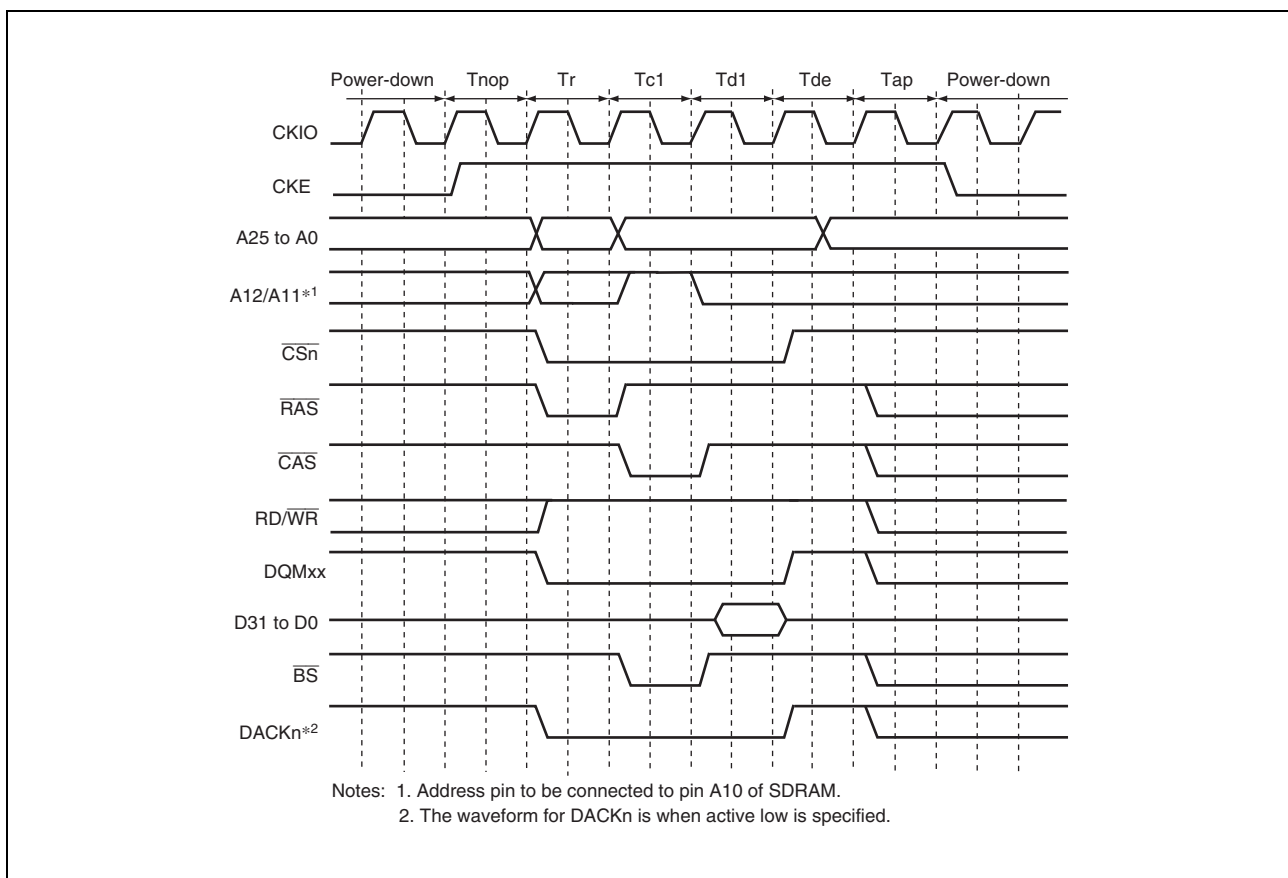


Figure 8.29 Power-Down Mode Access Timing

## (11) Power-On Sequence

In order to use SDRAM, mode setting must first be made for SDRAM after the pose interval specified for the SDRAM to be used after powering on. The pose interval should be obtained by a power-on reset generating circuit or software.

To perform SDRAM initialization correctly, the registers of this module must first be set, followed by a write to the SDRAM mode register. In SDRAM mode register setting, the address signal value at that time is latched by a combination of the  $\overline{CSn}$ ,  $\overline{RAS}$ ,  $\overline{CAS}$ , and  $RD/\overline{WR}$  signals. If the value to be set is X, the bus state controller provides for value X to be written to the SDRAM mode register by performing a 16-bit write to address H'3FFFD000 + X for area 2 SDRAM, and to address H'3FFFE000 + X for area 3 SDRAM. In this operation the data is ignored, but the mode write is performed as a byte-size access. To set burst read/single write or burst read/burst write (CAS latency 2 to 3, wrap type = sequential, and burst length 1) supported by the LSI, arbitrary data is written in 16 bits to the access addresses shown in table 8.15. In this time 0 is output at the external address pins of A12 or later.

**Table 8.15 Access Address in SDRAM Mode Register Write**

- Setting for Area 2  
Burst read/single write (burst length 1):

Data Bus Width	CAS Latency	Access Address	External Address Pin
16 bits	2	H'3FFFD440	H'0000440
	3	H'3FFFD460	H'0000460
32 bits	2	H'3FFFD880	H'0000880
	3	H'3FFFD8C0	H'00008C0

Burst read/burst write (burst length 1):

Data Bus Width	CAS Latency	Access Address	External Address Pin
16 bits	2	H'3FFFD040	H'0000040
	3	H'3FFFD060	H'0000060
32 bits	2	H'3FFFD080	H'0000080
	3	H'3FFFD0C0	H'00000C0

- Setting for Area 3  
Burst read/single write (burst length 1):

Data Bus Width	CAS Latency	Access Address	External Address Pin
16 bits	2	H'3FFFE440	H'0000440
	3	H'3FFFE460	H'0000460
32 bits	2	H'3FFFE880	H'0000880
	3	H'3FFFE8C0	H'00008C0

Burst read/burst write (burst length 1):

Data Bus Width	CAS Latency	Access Address	External Address Pin
16 bits	2	H'3FFFE040	H'0000040
	3	H'3FFFE060	H'0000060
32 bits	2	H'3FFFE080	H'0000080
	3	H'3FFFE0C0	H'00000C0

Mode register setting timing is shown in Figure 8.30. A PALL command (all bank pre-charge command) is firstly issued. A REF command (auto refresh command) is then issued 8 times. An MRS command (mode register write command) is finally issued. Idle cycles, of which number is specified by the WTRP1 and WTRP0 bits in CS3WCR, are inserted between the PALL and the first REF. Idle cycles, of which number is specified by the WTRC1 and WTRC0 bits in CS3WCR, are inserted between REF and REF, and between the 8th REF and MRS. One or more idle cycles are inserted between the MRS and a command to be issued next.

It is necessary to keep idle time of certain cycles for SDRAM before issuing PALL command after power-on. Refer to the manual of the SDRAM for the idle time to be needed. When the pulse width of the reset signal is longer than the idle time, mode register setting can be started immediately after the reset, but care should be taken when the pulse width of the reset signal is shorter than the idle time.

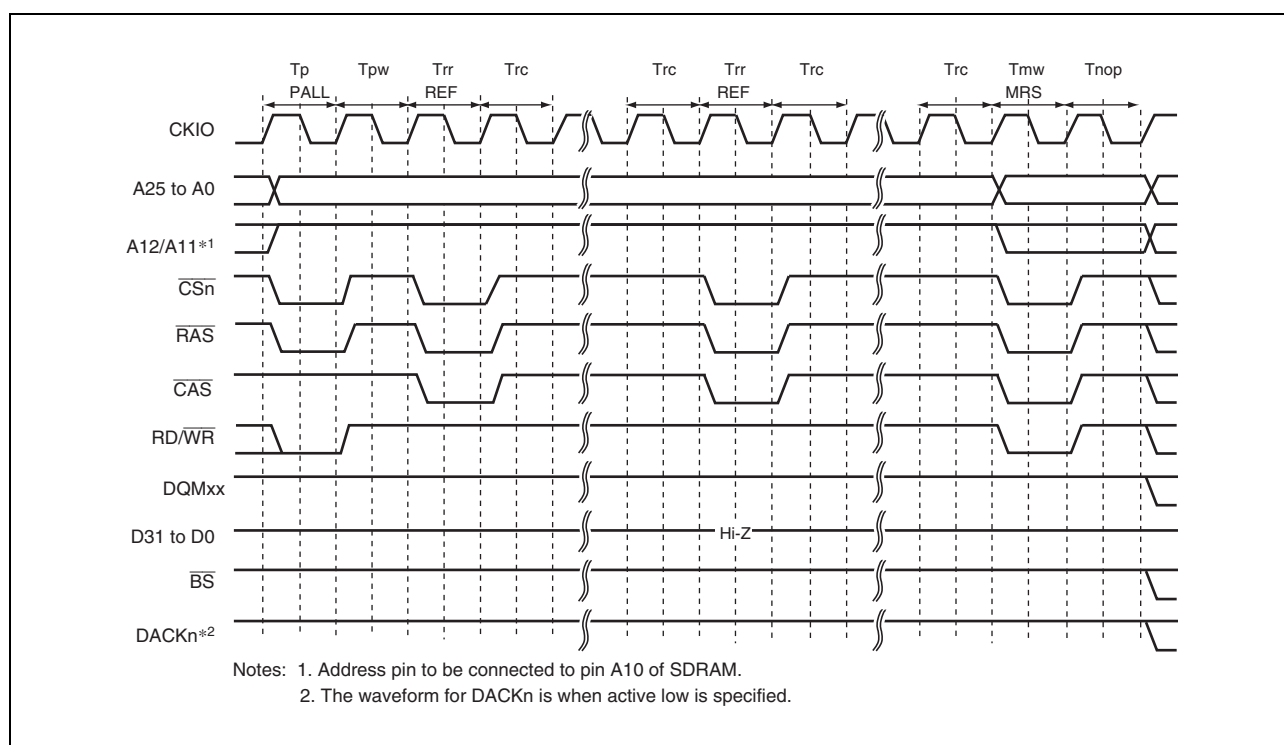


Figure 8.30 SDRAM Mode Write Timing (Based on JEDEC)

## (12) Low-Power SDRAM

The low-power SDRAM can be accessed using the same protocol as the normal SDRAM.

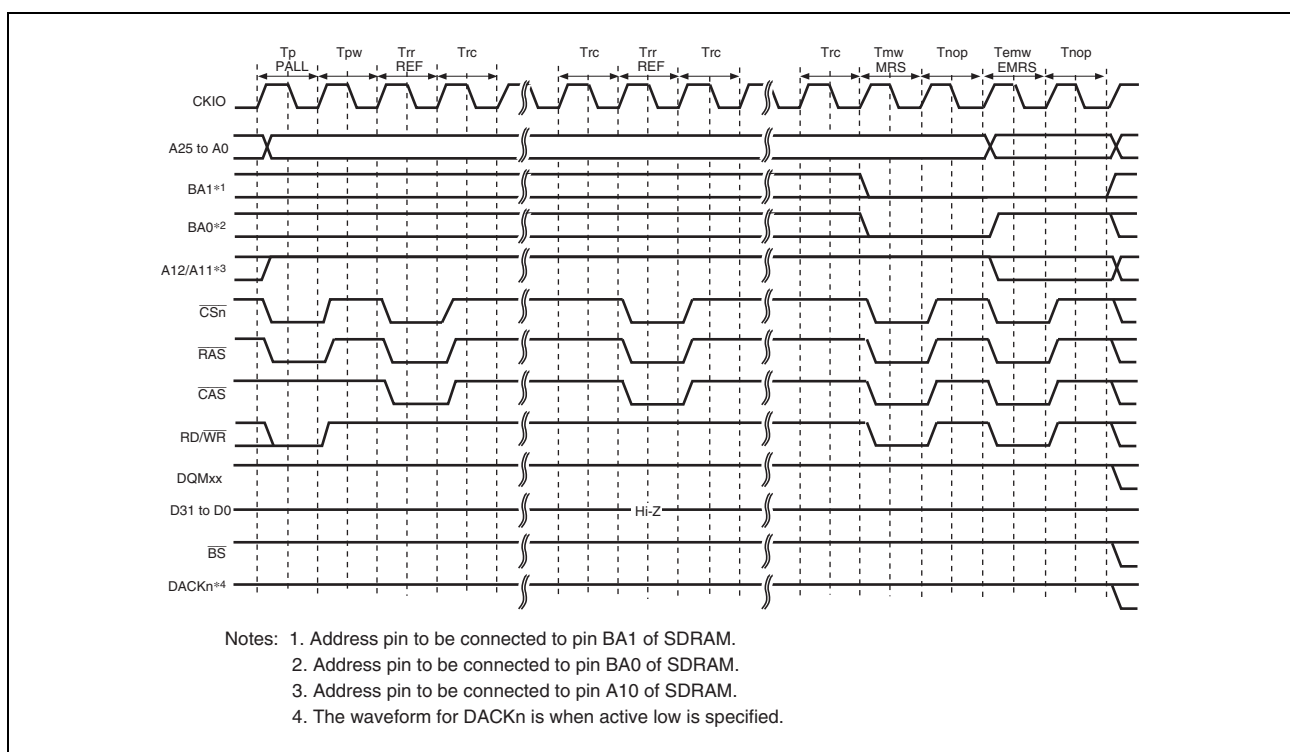
The differences between the low-power SDRAM and normal SDRAM are that partial refresh takes place that puts only a part of the SDRAM in the self-refresh state during the self-refresh function, and that power consumption is low during refresh under user conditions such as the operating temperature. The partial refresh is effective in systems in which the data in a work area other than the specific area can be lost without severe repercussions. For details, please refer to the Data Sheet for the low-power SDRAM to be used.

The low-power SDRAM supports the extension mode register in addition to the mode registers as the normal SDRAM. This LSI supports issuing of the extension mode register write command (EMRS).

The EMRS command is issued according to the conditions specified in table below. For example, if data H'0YYYYYYY is written to address H'3FFFEXX0 in 32 bits, the commands are issued to the CS3 space in the following sequence: PALL -> REF \* 8 -> MRS -> EMRS. In this case, the MRS and EMRS issue addresses are H'0000XX0 and H'YYYYYYY, respectively. If data H'1YYYYYYY is written to address H'3FFFEXX0 in 32 bits, the commands are issued to the CS3 space in the following sequence: PALL -> MRS -> EMRS.

**Table 8.16 Output Addresses when EMRS Command Is Issued**

Command to be Issued	Access Address	Access Data	Write Access Size	MRS Command Issue Address	EMRS Command Issue Address
CS2 MRS	H'3FFFDXX0	H'*****	16 bits	H'0000XX0	—
CS3 MRS	H'3FFFEXX0	H'*****	16 bits	H'0000XX0	—
CS2 MRS + EMRS (with refresh)	H'3FFFDXX0	H'0YYYYYYY	32 bits	H'0000XX0	H'YYYYYYY
CS3 MRS + EMRS (with refresh)	H'3FFFEXX0	H'0YYYYYYY	32 bits	H'0000XX0	H'YYYYYYY
CS2 MRS + EMRS (without refresh)	H'3FFFDXX0	H'1YYYYYYY	32 bits	H'0000XX0	H'YYYYYYY
CS3 MRS + EMRS (without refresh)	H'3FFFEXX0	H'1YYYYYYY	32 bits	H'0000XX0	H'YYYYYYY

**Figure 8.31 EMRS Command Issue Timing**

- Deep power-down mode

The low-power SDRAM supports the deep power-down mode as a low-power consumption mode. In the partial self-refresh function, self-refresh is performed on a specific area. In the deep power-down mode, self-refresh will not be performed on any memory area. This mode is effective in systems where all of the system memory areas are used as work areas.

If the RMODE bit in the SDCR is set to 1 while the DEEP and RFSH bits in the SDCR are set to 1, the low-power SDRAM enters the deep power-down mode. If the RMODE bit is cleared to 0, the CKE signal is pulled high to cancel the deep power-down mode. Before executing an access after returning from the deep power-down mode, the power-up sequence must be re-executed.

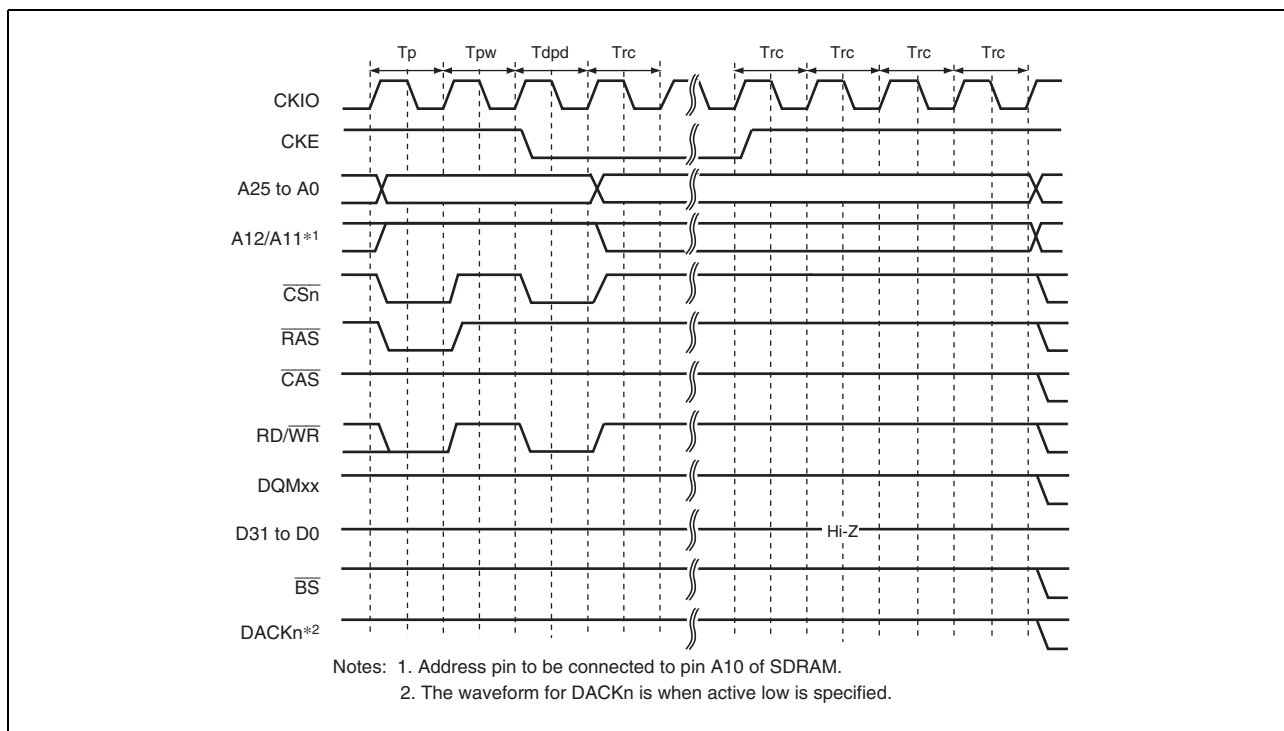


Figure 8.32 Deep Power-Down Mode Transition Timing



### 8.5.7 Burst ROM (Clocked Asynchronous) Interface

The burst ROM (clocked asynchronous) interface is used to access a memory with a high-speed read function using a method of address switching called the burst mode or page mode. In a burst ROM (clocked asynchronous) interface, basically the same access as the normal space is performed, but the 2nd and subsequent access cycles are performed only by changing the address, without negating the  $\overline{RD}$  signal at the end of the 1st cycle. In the 2nd and subsequent access cycles, addresses are changed at the falling edge of the CKIO.

For the 1st access cycle, the number of wait cycles specified by the W3 to W0 bits in CSnWCR is inserted. For the 2nd and subsequent access cycles, the number of wait cycles specified by the BW1 and BW0 bits in CSnWCR is inserted. In the access to the burst ROM (clocked asynchronous), the  $\overline{BS}$  signal is asserted only to the first access cycle. An external wait input is valid only to the first access cycle.

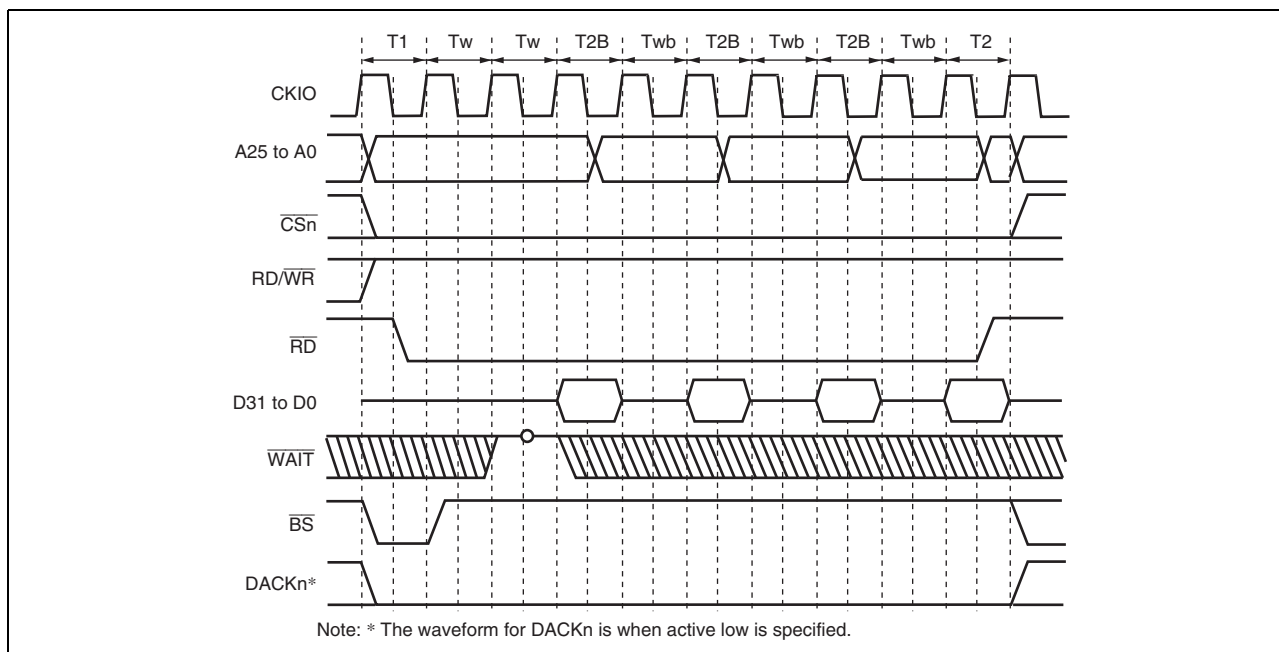
In the single access or write access that does not perform the burst operation in the burst ROM (clocked asynchronous) interface, access timing is same as a normal space.

Table 8.13 lists a relationship between bus width, access size, and the number of bursts. Figure 8.33 shows a timing chart.

**Table 8.17 Relationship between Bus Width, Access Size, and Number of Bursts**

Bus Width	Access Size	CSnWCR. BST[1:0] Bits	Number of Bursts	Access Count				
8 bits	8 bits	Not affected	1	1				
	16 bits	Not affected	2	1				
	32 bits	Not affected	4	1				
	16 bytes	00	00	16	1			
			01	4	4			
	32 bytes	00	00	16	2			
			01	4	8			
	64 bytes	00	00	16	4			
			01	4	16			
	16 bits	8 bits	Not affected	1	1			
16 bits		Not affected	1	1				
32 bits		Not affected	2	1				
16 bytes		00	00	8	1			
			01	2	4			
			10*1	4	2			
				2, 4, 2	3			
				32 bytes	00	00	8	2
						01	2	8
10*1		4	4					
				2, 4, 2	6			
				64 bytes	00	00	8	4
						01	2	16
10*1		4	8					
				2, 4, 2	12			
				8 bits	Not affected	1	1	
	16 bits			Not affected	1	1		
32 bits	32 bits	Not affected	1	1				
	16 bytes	Not affected	4	1				
	32 bytes	Not affected	4	2				
	64 bytes	Not affected	4	4				

Note 1. When the bus width is 16 bits, the access size is 16 bits or more, and the BST[1:0] bits in CSnWCR are 10, the number of bursts and access count depend on the access start address. At address H'xxx0 or H'xxx8, 4-4 burst access is performed. At address H'xxx4 or H'xxxC, 2-4-2 burst access is performed.



**Figure 8.33 Burst ROM Access Timing (Clocked Asynchronous) (Bus Width = 32 Bits, 16-Byte Transfer (Number of Burst 4), Wait Cycles Inserted in First Access = 2, Wait Cycles Inserted in Second and Subsequent Access Cycles = 1)**

### 8.5.8 SRAM Interface with Byte Selection

The SRAM interface with byte selection is a memory interface that outputs the byte selection signal ( $\overline{WEn}$ ) in both read and write bus cycles. This interface has 16-bit data pins and accesses SRAMs having upper and lower byte selection pins, such as UB and LB.

When the BAS bit in CSnWCR is cleared to 0 (initial value), the write access timing of the SRAM interface with byte selection is the same as that for the normal space interface. While in read access of a byte-selection SRAM interface, the byte-selection signal is output from the  $\overline{WEn}$  pin, which is different from that for the normal space interface. The basic access timing is shown in figure 8.34. In write access, data is written to the memory according to the timing of the byte-selection pin ( $\overline{WEn}$ ). For details, please refer to the Data Sheet for the corresponding memory.

If the BAS bit in CSnWCR is set to 1, the  $\overline{WEn}$  pin and RD/ $\overline{WR}$  pin timings change. Figure 8.35 shows the basic access timing. In write access, data is written to the memory according to the timing of the write enable pin (RD/ $\overline{WR}$ ). The data hold timing from RD/ $\overline{WR}$  negation to data write must be acquired by setting the HW1 and HW0 bits in CSnWCR.

Figure 8.36 shows the access timing when a software wait is specified.

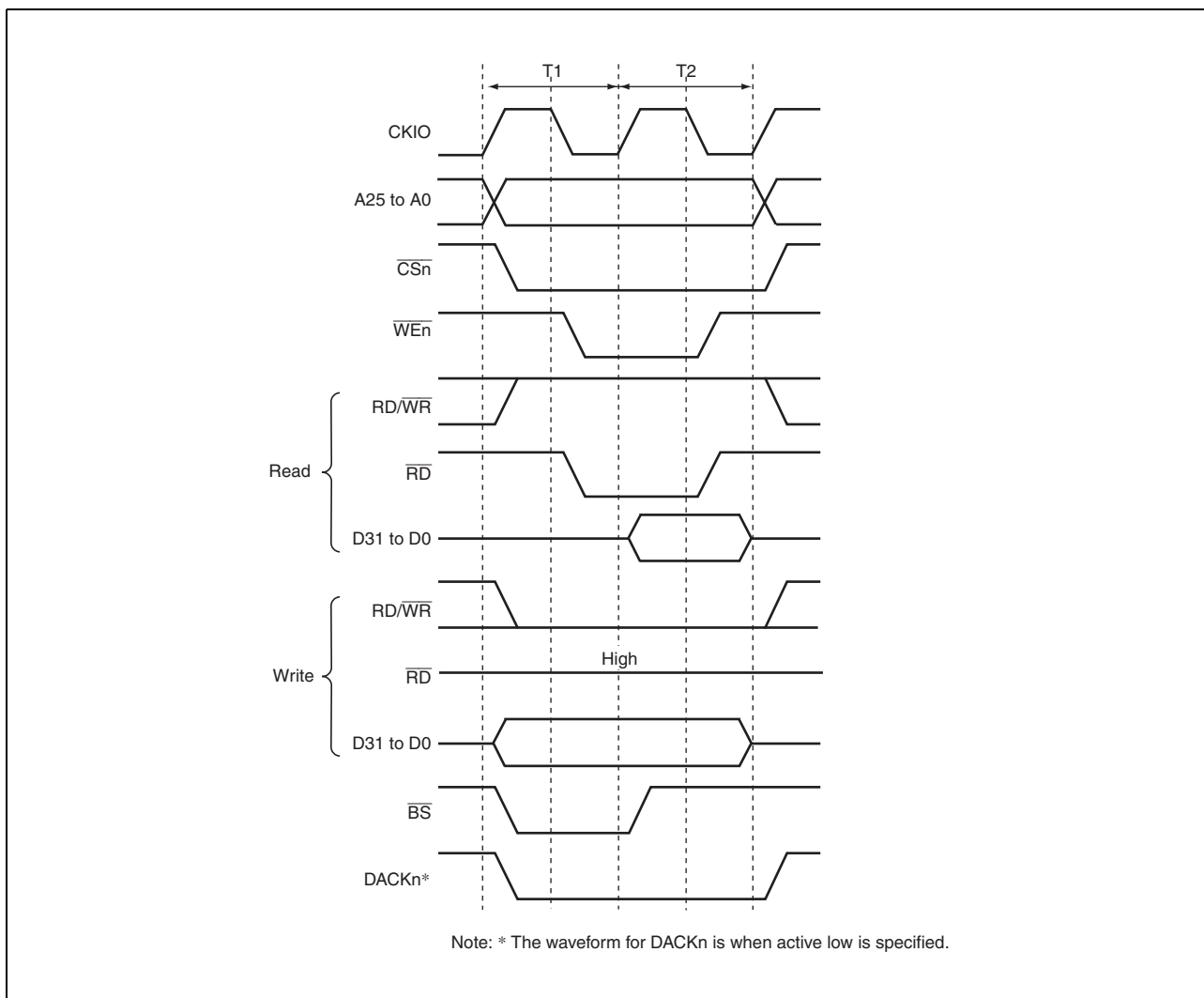


Figure 8.34 Basic Access Timing for SRAM with Byte Selection (BAS = 0)

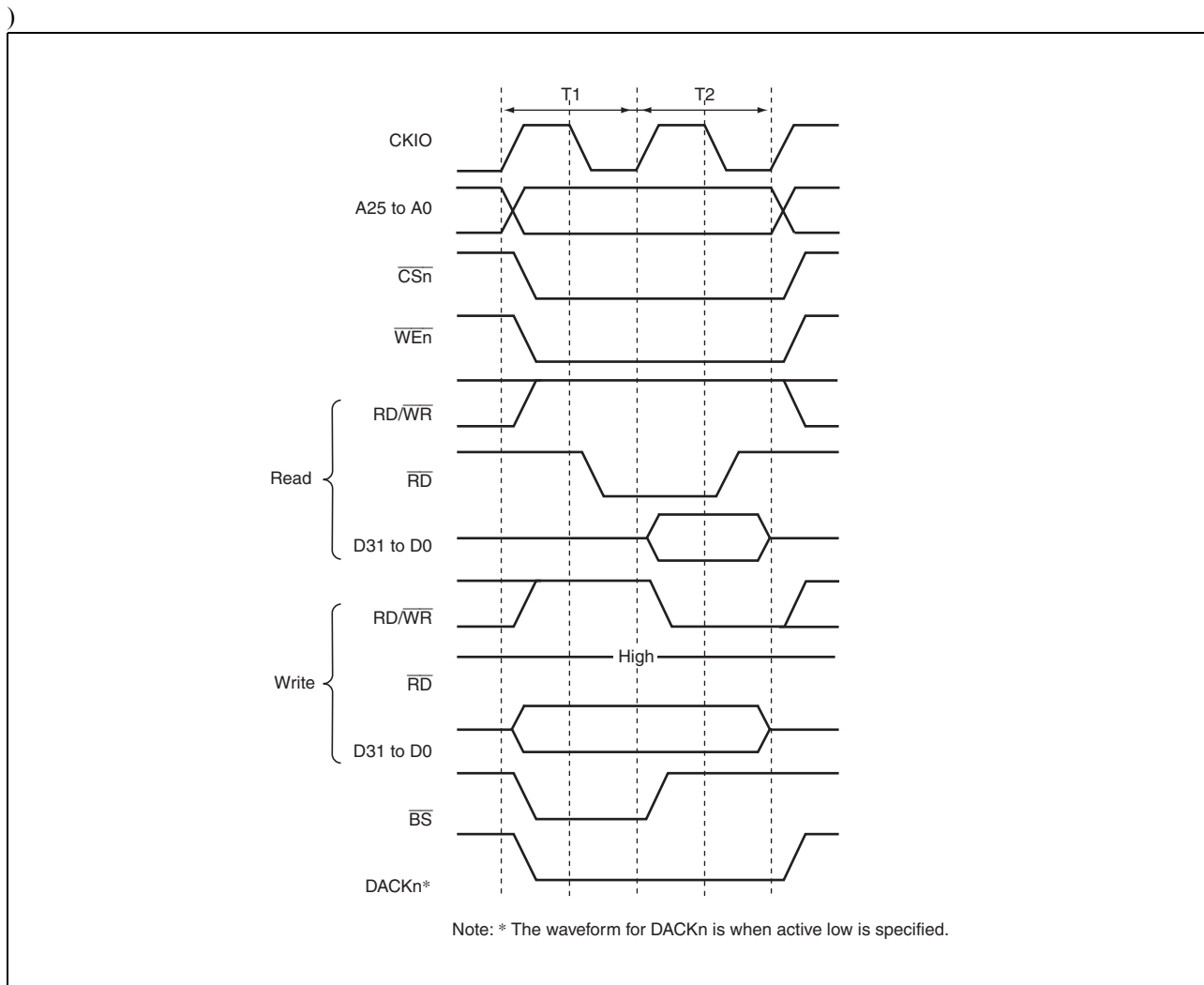


Figure 8.35 Basic Access Timing for SRAM with Byte Selection (BAS = 1)

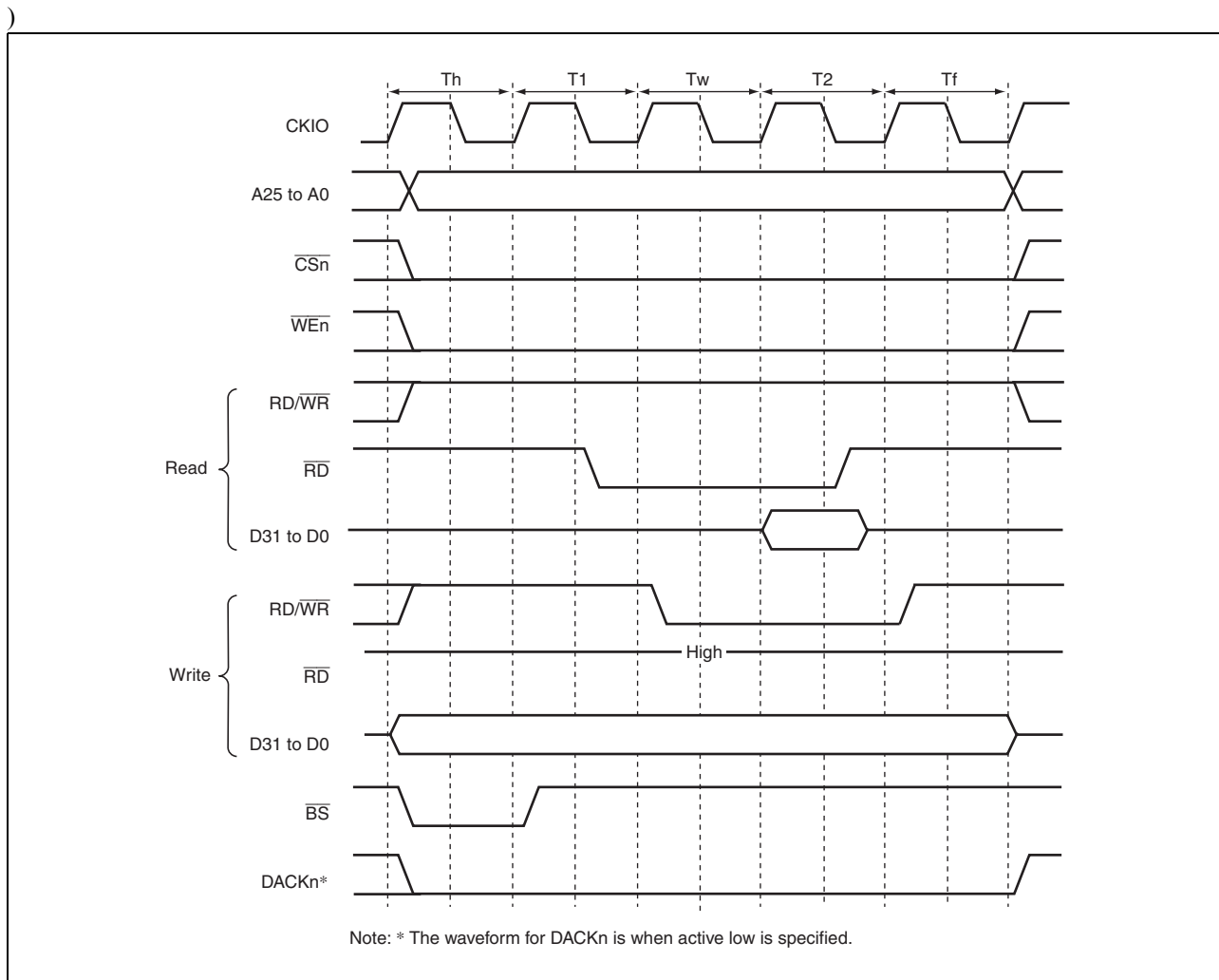


Figure 8.36 Wait Timing for SRAM with Byte Selection (BAS = 1) (SW[1:0] = 01, WR[3:0] = 0001, HW[1:0] = 01)

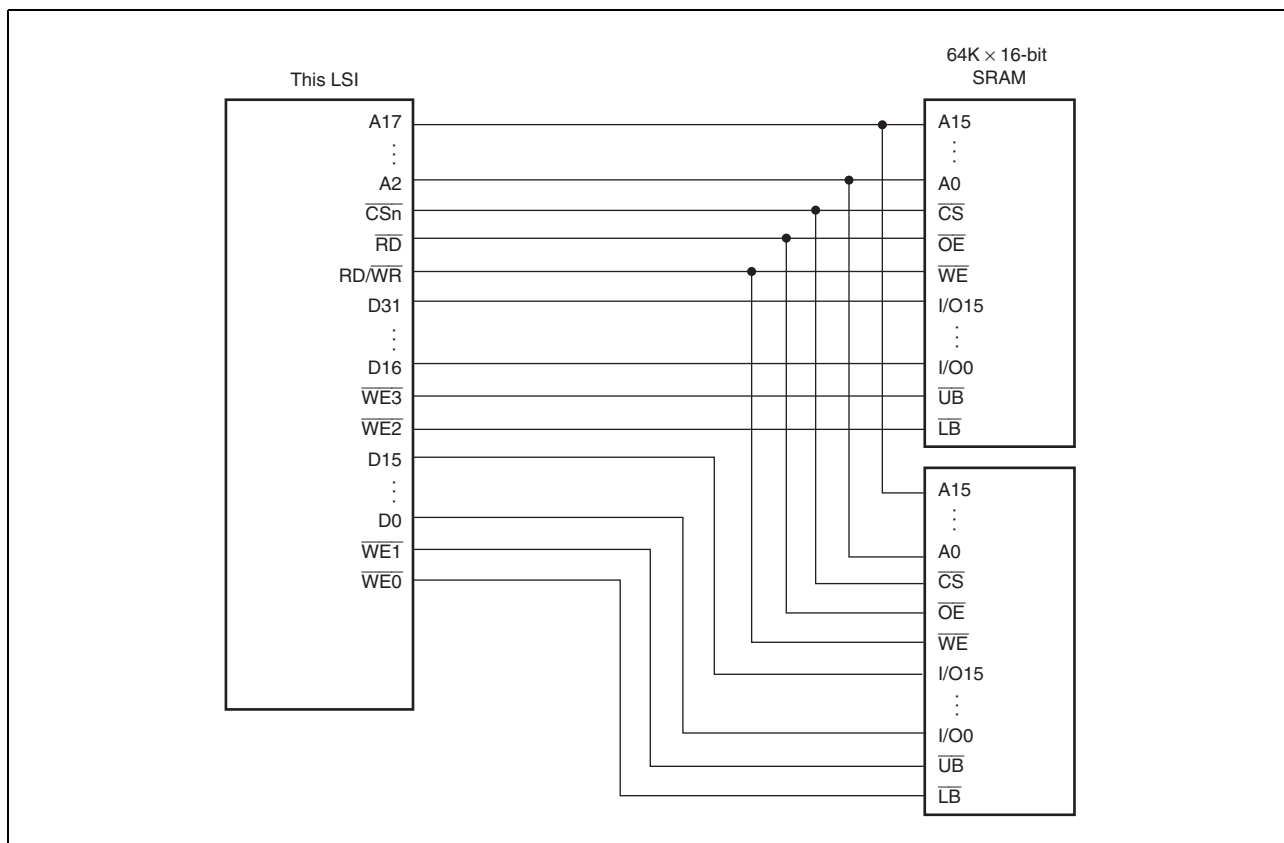


Figure 8.37 Example of Connection with 32-Bit Data-Width SRAM with Byte Selection

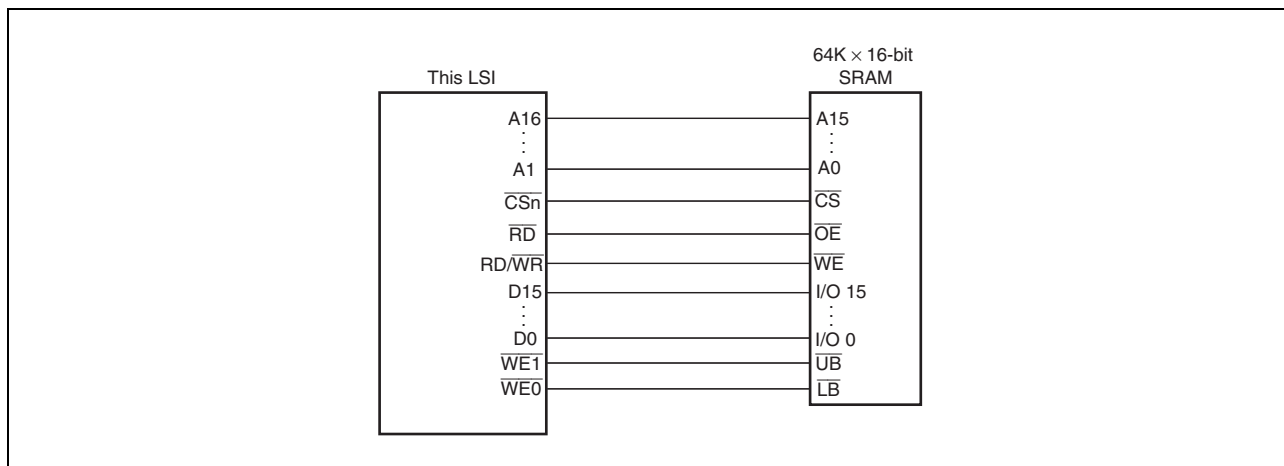


Figure 8.38 Example of Connection with 16-Bit Data-Width SRAM with Byte Selection

### 8.5.9 Burst ROM (Clocked Synchronous) Interface

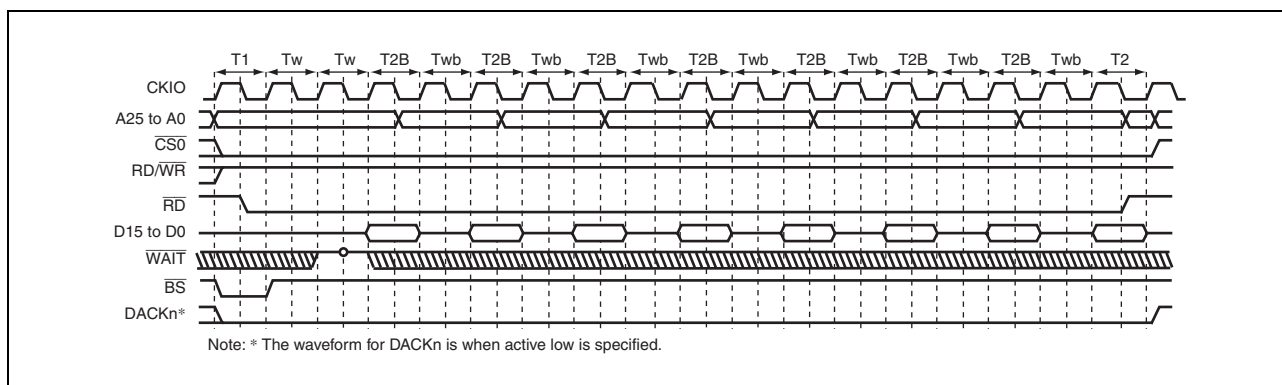
The burst ROM (clocked synchronous) interface is supported to access a ROM with a synchronous burst function at high speed. The burst ROM interface accesses the burst ROM in the same way as a normal space. This interface is valid only for area 0.

In the first access cycle, wait cycles are inserted. In this case, the number of wait cycles to be inserted is specified by the W3 to W0 bits in CS0WCR. In the second and subsequent cycles, the number of wait cycles to be inserted is specified by the BW1 and BW0 bits in CS0WCR.

While the burst ROM (clocked synchronous) is accessed, the  $\overline{BS}$  signal is asserted only for the first access cycle and an external wait input is also valid for the first access cycle.

When the bus width is 16 bits, the burst length must be specified as 8. When the bus width is 32 bits, the burst length must be specified as 4. The burst ROM interface does not support the 8-bit bus width for the burst ROM.

The burst ROM interface performs burst operations for all read access. For example, in a 32-bit access over a 16-bit bus, valid 16-bit data is read two times and invalid 16-bit data is read six times. These invalid data read cycles increase the memory access time and degrade the program execution speed and DMA transfer speed. To prevent this problem, it is recommended using a read in a 16-byte or more access size. The burst ROM interface performs write access in the same way as normal space access.



**Figure 8.39 Burst ROM Access Timing (Clocked Synchronous) (Burst Length = 8, Wait Cycles Inserted in First Access = 2, Wait Cycles Inserted in Second and Subsequent Access Cycles = 1)**

### 8.5.10 Wait between Access Cycles

As the operating frequency of LSIs becomes higher, the off-operation of the data buffer often collides with the next data access when the read operation from devices with slow access speed is completed. As a result of these collisions, the reliability of the device is low and malfunctions may occur. A function that avoids data collisions by inserting idle (wait) cycles between continuous access cycles has been newly added.

The number of wait cycles between access cycles can be set by the WM bit in CSnWCR, bits IWW2 to IWW0, IWRWD2 to IWRWD0, IWRWS2 to IWRWS0, IWRRD2 to IWRRD0, and IWRRS2 to IWRRS 0 in CSnBCR. The conditions for setting the idle cycles between access cycles are shown below.

1. Continuous access cycles are write-read or write-write
2. Continuous access cycles are read-write for different spaces
3. Continuous access cycles are read-write for the same space
4. Continuous access cycles are read-read for different spaces
5. Continuous access cycles are read-read for the same space

For the specification of the number of idle cycles between access cycles described above, refer to the description of each register.

Besides the idle cycles between access cycles specified by the registers, idle cycles must be inserted to interface with the internal bus or to obtain the minimum pulse width for a multiplexed pin ( $\overline{WEn}$ ). The following gives detailed information about the idle cycles and describes how to estimate the number of idle cycles.

The number of idle cycles on the external bus from  $\overline{CSn}$  negation to  $\overline{CSn}$  or  $\overline{CSm}$  assertion is described below.

There are seven conditions that determine the number of idle cycles on the external bus as shown in Table 8.13. The effects of these conditions are shown in Figure 8.40.



**Table 8.18 Conditions for Determining Number of Idle Cycles**

No.	Condition	Description	Range	Note
[1]	IW***[2:0] in CSnBCR	These bits specify the number of idle cycles for access. The number of idle cycles can be specified independently for each combination of the previous and next cycles. For example, in the case where reading CS1 space followed by reading other CS space, the bits IWRRD[2:0] in CS1BCR should be set to B'100 to specify six or more idle cycles. This condition is effective only for access cycles other than single address transfer and generates idle cycles after the access is completed.	0 to 12	Do not set 0 for the number of idle cycles between memory types which are not allowed to be accessed successively.
[2]	SDRAM-related bits in CSnWCR	These bits specify precharge completion and startup wait cycles and idle cycles between commands for SDRAM access. This condition is effective only for SDRAM access and generates idle cycles after the access is completed	0 to 3	Specify these bits in accordance with the specification of the target SDRAM.
[3]	WM in CSnWCR	This bit enables or disables external $\overline{\text{WAIT}}$ pin input for the memory types other than SDRAM. When this bit is cleared to 0 (external $\overline{\text{WAIT}}$ enabled), one idle cycle is inserted to check the external $\overline{\text{WAIT}}$ pin input after the access is completed. When this bit is set to 1 (disabled), no idle cycle is generated.	0 or 1	
[4]	Read data transfer cycle	One idle cycle is inserted after a read access is completed. This idle cycle is not generated for the first or middle cycles in divided access cycles. This is neither generated when the HW[1:0] bits in CSnWCR are not B'00.	0 or 1	One idle cycle is always generated after a read cycle with SDRAM.
[5]	Internal bus idle cycles, etc.	External bus access requests from the CPU or the direct memory access controller and their results are passed through the internal bus. The external bus enters idle state during internal bus idle cycles or while a bus other than the external bus is being accessed. This condition is not effective for divided access cycles, which are generated by the bus state controller when the access size is larger than the external data bus width.	0 or larger	The number of internal bus idle cycles may not become 0 depending on the CPU: internal bus: CKIO
[6]	Write data wait cycles	During write access, a write cycle is executed on the external bus only after the write data becomes ready. This write data wait period generates idle cycles before the write cycle. Note that when the previous cycle is a write cycle and the internal bus idle cycles are shorter than the previous write cycle, write data can be prepared in parallel with the previous write cycle and therefore, no idle cycle is generated (write buffer effect).	0 or 1	For write → write or write → read access cycles, successive access cycles without idle cycles may be available due to the write buffer effect described in the left column. If successive access cycles without idle cycles are not allowed, specify the minimum number of idle cycles between access cycles through CSnBCR.
[7]	Idle cycles between different memory types	To ensure the minimum pulse width on the signal-multiplexed pins, idle cycles may be inserted before access after memory types are switched. For some memory types, idle cycles are inserted even when memory types are not switched.	0 to 2	The number of idle cycles depends on the target memory types. See table 8.19.

In the above conditions, a total of four conditions, that is, condition [1], condition [2] or [3] (either one is effective), a set of conditions [4] to [6] (these are generated successively, and therefore the sum of them should be taken as one set of idle cycles), and condition [7] are generated at the same time. The maximum number of idle cycles among these four conditions become the number of idle cycles on the external bus. To ensure the minimum idle cycles, be sure to make register settings for condition [1].

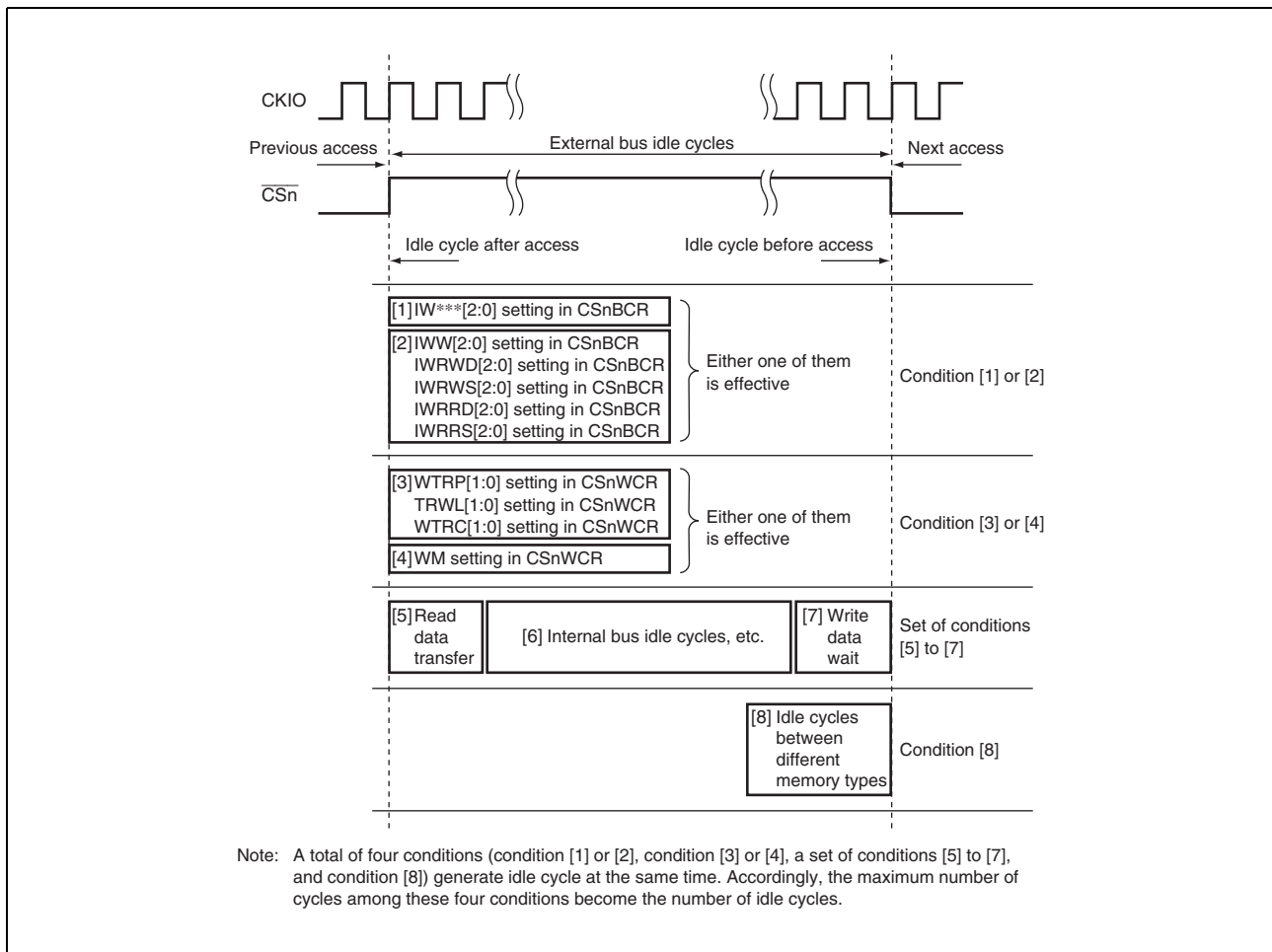


Figure 8.40 Idle Cycle Conditions

Table 8.19 Number of Idle Cycles Inserted between Access Cycles to Different Memory Types

Previous Cycle	Next Cycle						
	SRAM	Burst ROM (Asynchronous)	MPX-I/O	Byte SRAM (BAS = 0)	Byte SRAM (BAS = 1)	SDRAM	Burst ROM (Synchronous)
SRAM	0	0	1	0	0/1*1	0/1*1	0
Burst ROM (asynchronous)	0	0	1	0	0/1*1	0/1*1	0
MPX-I/O	1	1	0	1	1	1	1
Byte SRAM (BAS = 0)	0	0	1	0	0/1*1	0/1*1	0
Byte SRAM (BAS = 1)	0/1*1	0/1*1	1/2*1	0/1*1	0	0	0/1*1
SDRAM	1	1	2	1	0	0	1
Burst ROM (synchronous)	0	0	1	0	1	1	0

Note 1. The number of idle cycles is determined by the setting of bits HW[1:0] in the CSnWCR register for the previous cycle. The values on the left and right sides of the virgules show the numbers of idle cycles when HW[1:0] ≠ B'00 and HW[1:0] = B'00, respectively. If the memory connected to the CSn space in the previous cycle is of a type for which bits HW[1:0] in the CSnWCR register are ineffective, the number of idle cycles will be the value on the right side.

### 8.5.11 Others

#### (1) Reset

This module can be initialized completely only at power-on reset. At power-on reset, all signals are negated and data output buffers are turned off regardless of the bus cycle state after the internal reset is synchronized with the internal clock. All control registers are initialized. In software standby and sleep, control registers of the bus state controller are not initialized.

#### (2) Caution on Write Buffer

Since the bus state controller incorporates a one-stage write buffer, it can execute an access via the internal bus before the previous external bus cycle is completed in a write cycle. If the on-chip module is read or written after the external low-speed memory is written, the on-chip module can be accessed before the completion of the external low-speed memory write cycle.

In read cycles, the CPU is placed in the wait state until read operation has been completed. To continue the process after the data write to the device has been completed, perform a dummy read to the same address to check for completion of the write before the next process to be executed.

The write buffer of the bus state controller functions in the same way for an access by a bus master other than the CPU such as the direct memory access controller. Accordingly, to perform DMA transfers, the next read cycle is initiated before the previous write cycle is completed. Note, however, that if both the DMA source and destination addresses exist in external memory space, the next read cycle will not be initiated until the previous write cycle is completed.

Changing the registers in this module while the write buffer is operating may disrupt correct write access. Therefore, do not change the registers in this module immediately after a write access. If this change becomes necessary, do it after executing a dummy read of the write data.

#### (3) On-Chip Peripheral Module Access

To access an on-chip module register, two or more peripheral module clock (P0f or P1f) cycles are required. Care must be taken in system design.

When the CPU writes data to the internal peripheral registers, the CPU performs the succeeding instructions without waiting for the completion of writing to registers.

For example, a case is described here in which the system is transferring to the software standby mode for power savings. To make this transition, the WFI instruction must be performed after setting the STBY bit in the STBCR1 register to 1. However a dummy read of the STBCR1 register is required before executing the WFI instruction. If a dummy read is omitted, the CPU executes the WFI instruction before the STBY bit is set to 1, thus the system enters sleep mode not software standby mode. A dummy read of the STBCR1 register is indispensable to complete writing to the STBY bit. To reflect the change by internal peripheral registers while performing the succeeding instructions, execute a dummy read of registers to which write instruction is given and then perform the succeeding instructions.

## 9. Direct Memory Access Controller

The direct memory access controller can be used in place of the CPU to perform high-speed transfers between external devices that have DACK (transfer request acknowledge signal), external memory, on-chip memory, memory-mapped external devices, and on-chip peripheral modules.

### 9.1 Features

- Number of channels selectable: 16 channels (CH0 to CH15). Only the CH0 channel can receive external requests.
- 4-Gbyte address space (according to the architecture)
- Transfer data size: Byte, two bytes, four bytes, eight bytes, 16 bytes, 32 bytes, 64 bytes, and 128 bytes
- Maximum transfer count:  $2^{32} - 1$  bytes
- Address mode: Dual address mode
- Transfer requests: Can be selected from the three types of external request, on-chip peripheral module request, and auto request (software trigger)
- The following modules can issue on-chip peripheral module requests.
  - Serial communication interface with FIFO: 16 sources
  - A/D converter: 1 source
  - Multi-function timer pulse unit 2: 5 sources
  - USB2.0 host/function module: 4 sources
  - NAND flash memory controller: 1 source
  - Serial sound interface: 10 sources
  - Sound generator: 4 sources
  - Renesas SPDIF interface: 2 sources
  - CD-ROM decoder: 1 source
  - SD host interface: 4 sources
  - MMC host interface: 2 sources
  - Renesas serial peripheral interface: 10 sources
  - Motor control PWM timer: 2 sources
  - IEBus™ controller: 2 sources
  - OS timer: 2 sources
  - SCUX: 8 sources
  - Media local bus: 1 source
  - Serial communication interface: 4 sources
  - I<sup>2</sup>C bus interface: 8 sources
  - LIN interface: 4 sources
  - Pixel format converter: 4 sources
- Transfer mode: Single transfer mode and block transfer mode are selectable.
- Priority: The channel priority levels within channels 0 to 7 and within channels 8 to 15 are selectable between fixed mode and round-robin mode (the channel priority level between the group of channels 0 to 7 and the group of channels 8 to 15 is round-robin mode).
- Interrupt request: An interrupt request can be sent to the CPU on completion of data transfer (DMA transfer end interrupt per channel) or on occurrence of a transfer error (DMA error interrupt).
- External request detection: Low level detection, high level detection, rising edge detection, and falling edge detection are selectable for DREQ input detection.
- The DMA registers have a continuous execution function that allows the next DMA transfer to be executed continuously by making settings for the next DMA transfer during execution of the current DMA transfer. This continuous execution function can be enabled or disabled independently in each channel.
- Link mode: In this mode, the setting data (descriptor data) located in the memory by the CPU is automatically

retrieved by the DMAC, and DMA transfer is performed according to those values.

- Buffer sweep: If an ongoing DMA transfer is forced to end, the data already retrieved into the buffer can be output before DMA transfer ends.
- Interval: A specific DMA transfer interval can be specified to adjust the bus occupancy.

## 9.2 Input/Output Pins

Table 9.1 lists the pin configuration. This module has pins for a single channel (CH0) as the external bus use.

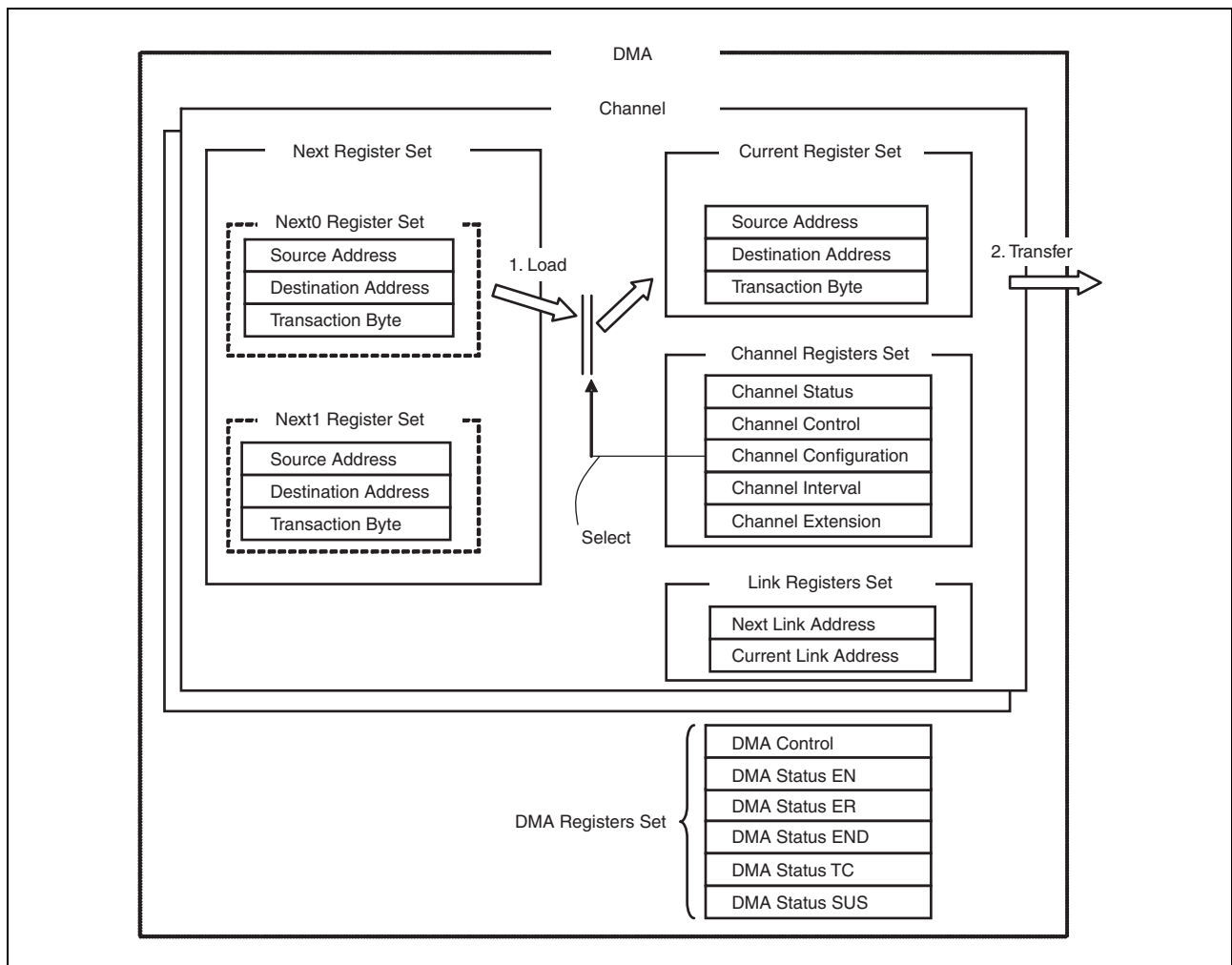
**Table 9.1 Pin Configuration**

Channel	Name	Pin Name	I/O	Function
0	DMA transfer request	DREQ0	Input	DMA transfer request input from an external device to channel 0
	DMA transfer request acknowledge	DACK0	Output	DMA transfer request acknowledge output from channel 0 of this module
	DMA transfer end	TEND0	Output	DMA transfer end output for channel 0 of this module

Note 1. For the active level of DACK0 and TEND0, refer to section 8, Bus State Controller.

## 9.3 Register Configuration

The register configuration is shown in the figure below.



**Figure 9.1 Register Configuration**

**(a) Next Register Set**

This register set is used to set the source address, destination address, and transfer byte count of the DMA transaction to be executed next.

It consists of the Next0 register set and the Next1 register set.

In register mode, set this register set by using software. In link mode, the descriptor read data is automatically set in the Next0 register set.

These register set values are loaded to the Current Register Set and used for DMA transfer.

**(b) Current Register Set**

This register set indicates the source address, destination address, and transfer byte count of the currently executed DMA transaction.

The values are loaded from the Next0/1 register set (register mode) or from the descriptor read data (link mode). The user cannot write directly to this register set.

The register set is automatically updated each time a DMA transaction is executed.

**(c) Channel Register Set**

This register set is used to make the DMA transfer settings.

The settings to be made with this register set include channel status indication, channel control, DMA transaction setting, and DMA transaction interval.

**(d) Link Register Set**

This register set consists of a register that sets the address of the descriptor to be loaded next in link mode (Next Link Address Register) and a register that indicates the address of the currently executed descriptor (Current Link Address Register).

The Current Link Address Register is automatically updated when a descriptor is read. The user cannot write directly to this register set.

**(e) DMA Register Set**

This register set consists of a register that controls DMA as a whole and registers that indicate the status of the corresponding channels. It enables channel priority control as well as the monitoring of the channel status (EN, ER, END, TC, and SUS).

**(f) Extended Resource Selector Register Set**

This register set is used to select the on-chip peripheral module to perform DMA transfer and the external request.

## 9.4 Register Descriptions

Table 9.2 lists the register configuration. There are eleven control registers and five status registers for each channel, and twelve common control registers are used by all channels. In addition, there is one extension resource selector per two channels. Each channel number is expressed in the register names, as in N0SA\_0 for N0SA in channel 0.

**Table 9.2 Register Configuration**

Channel	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
0	Next0 source address register 0	N0SA_0	RW	H'00000000	H'E8200000	32
	Next0 destination address register 0	N0DA_0	RW	H'00000000	H'E8200004	32
	Next0 transaction byte register 0	N0TB_0	RW	H'00000000	H'E8200008	32
	Next1 source address register 0	N1SA_0	RW	H'00000000	H'E820000C	32
	Next1 destination address register 0	N1DA_0	RW	H'00000000	H'E8200010	32
	Next1 transaction byte register 0	N1TB_0	RW	H'00000000	H'E8200014	32
	Current source address register 0	CRSA_0	R	H'00000000	H'E8200018	32
	Current destination address register 0	CRDA_0	R	H'00000000	H'E820001C	32
	Current transaction byte register 0	CRTB_0	R	H'00000000	H'E8200020	32
	Channel status register 0	CHSTAT_0	R	H'00000000	H'E8200024	32
	Channel control register 0	CHCTRL_0	RW	H'00000000	H'E8200028	32
	Channel configuration register 0	CHCFG_0	RW	H'00000000	H'E820002C	32
	Channel interval register 0	CHITVL_0	RW	H'00000000	H'E8200030	32
	Channel extension register 0	CHEXT_0	RW	H'00000000	H'E8200034	32
	Next link address register 0	NXLA_0	RW	H'00000000	H'E8200038	32
	Current link address register 0	CRLA_0	R	H'00000000	H'E820003C	32
1	Next0 source address register 1	N0SA_1	RW	H'00000000	H'E8200040	32
	Next0 destination address register 1	N0DA_1	RW	H'00000000	H'E8200044	32
	Next0 transaction byte register 1	N0TB_1	RW	H'00000000	H'E8200048	32
	Next1 source address register 1	N1SA_1	RW	H'00000000	H'E820004C	32
	Next1 destination address register 1	N1DA_1	RW	H'00000000	H'E8200050	32
	Next1 transaction byte register 1	N1TB_1	RW	H'00000000	H'E8200054	32
	Current source address register 1	CRSA_1	R	H'00000000	H'E8200058	32
	Current destination address register 1	CRDA_1	R	H'00000000	H'E820005C	32
	Current transaction byte register 1	CRTB_1	R	H'00000000	H'E8200060	32
	Channel status register 1	CHSTAT_1	R	H'00000000	H'E8200064	32
	Channel control register 1	CHCTRL_1	RW	H'00000000	H'E8200068	32
	Channel configuration register 1	CHCFG_1	RW	H'00000000	H'E820006C	32
	Channel interval register 1	CHITVL_1	RW	H'00000000	H'E8200070	32
	Channel extension register 1	CHEXT_1	RW	H'00000000	H'E8200074	32
	Next link address register 1	NXLA_1	RW	H'00000000	H'E8200078	32
	Current link address register 1	CRLA_1	R	H'00000000	H'E820007C	32

**Table 9.2 Register Configuration**

Channel	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
2	Next0 source address register 2	N0SA_2	RW	H'00000000	H'E8200080	32
	Next0 destination address register 2	N0DA_2	RW	H'00000000	H'E8200084	32
	Next0 transaction byte register 2	N0TB_2	RW	H'00000000	H'E8200088	32
	Next1 source address register 2	N1SA_2	RW	H'00000000	H'E820008C	32
	Next1 destination address register 2	N1DA_2	RW	H'00000000	H'E8200090	32
	Next1 transaction byte register 2	N1TB_2	RW	H'00000000	H'E8200094	32
	Current source address register 2	CRSA_2	R	H'00000000	H'E8200098	32
	Current destination address register 2	CRDA_2	R	H'00000000	H'E820009C	32
	Current transaction byte register 2	CRTB_2	R	H'00000000	H'E82000A0	32
	Channel status register 2	CHSTAT_2	R	H'00000000	H'E82000A4	32
	Channel control register 2	CHCTRL_2	RW	H'00000000	H'E82000A8	32
	Channel configuration register 2	CHCFG_2	RW	H'00000000	H'E82000AC	32
	Channel interval register 2	CHITVL_2	RW	H'00000000	H'E82000B0	32
	Channel extension register 2	CHEXT_2	RW	H'00000000	H'E82000B4	32
	Next link address register 2	NXLA_2	RW	H'00000000	H'E82000B8	32
	Current link address register 2	CRLA_2	R	H'00000000	H'E82000BC	32
3	Next0 source address register 3	N0SA_3	RW	H'00000000	H'E82000C0	32
	Next0 destination address register 3	N0DA_3	RW	H'00000000	H'E82000C4	32
	Next0 transaction byte register 3	N0TB_3	RW	H'00000000	H'E82000C8	32
	Next1 source address register 3	N1SA_3	RW	H'00000000	H'E82000CC	32
	Next1 destination address register 3	N1DA_3	RW	H'00000000	H'E82000D0	32
	Next1 transaction byte register 3	N1TB_3	RW	H'00000000	H'E82000D4	32
	Current source address register 3	CRSA_3	R	H'00000000	H'E82000D8	32
	Current destination address register 3	CRDA_3	R	H'00000000	H'E82000DC	32
	Current transaction byte register 3	CRTB_3	R	H'00000000	H'E82000E0	32
	Channel status register 3	CHSTAT_3	R	H'00000000	H'E82000E4	32
	Channel control register 3	CHCTRL_3	RW	H'00000000	H'E82000E8	32
	Channel configuration register 3	CHCFG_3	RW	H'00000000	H'E82000EC	32
	Channel interval register 3	CHITVL_3	RW	H'00000000	H'E82000F0	32
	Channel extension register 3	CHEXT_3	RW	H'00000000	H'E82000F4	32
	Next link address register 3	NXLA_3	RW	H'00000000	H'E82000F8	32
	Current link address register 3	CRLA_3	R	H'00000000	H'E82000FC	32



**Table 9.2 Register Configuration**

Channel	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
4	Next0 source address register 4	N0SA_4	RW	H'00000000	H'E8200100	32
	Next0 destination address register 4	N0DA_4	RW	H'00000000	H'E8200104	32
	Next0 transaction byte register 4	N0TB_4	RW	H'00000000	H'E8200108	32
	Next1 source address register 4	N1SA_4	RW	H'00000000	H'E820010C	32
	Next1 destination address register 4	N1DA_4	RW	H'00000000	H'E8200110	32
	Next1 transaction byte register 4	N1TB_4	RW	H'00000000	H'E8200114	32
	Current source address register 4	CRSA_4	R	H'00000000	H'E8200118	32
	Current destination address register 4	CRDA_4	R	H'00000000	H'E820011C	32
	Current transaction byte register 4	CRTB_4	R	H'00000000	H'E8200120	32
	Channel status register 4	CHSTAT_4	R	H'00000000	H'E8200124	32
	Channel control register 4	CHCTRL_4	RW	H'00000000	H'E8200128	32
	Channel configuration register 4	CHCFG_4	RW	H'00000000	H'E820012C	32
	Channel interval register 4	CHITVL_4	RW	H'00000000	H'E8200130	32
	Channel extension register 4	CHEXT_4	RW	H'00000000	H'E8200134	32
	Next link address register 4	NXLA_4	RW	H'00000000	H'E8200138	32
	Current link address register 4	CRLA_4	R	H'00000000	H'E820013C	32
5	Next0 source address register 5	N0SA_5	RW	H'00000000	H'E8200140	32
	Next0 destination address register 5	N0DA_5	RW	H'00000000	H'E8200144	32
	Next0 transaction byte register 5	N0TB_5	RW	H'00000000	H'E8200148	32
	Next1 source address register 5	N1SA_5	RW	H'00000000	H'E820014C	32
	Next1 destination address register 5	N1DA_5	RW	H'00000000	H'E8200150	32
	Next1 transaction byte register 5	N1TB_5	RW	H'00000000	H'E8200154	32
	Current source address register 5	CRSA_5	R	H'00000000	H'E8200158	32
	Current destination address register 5	CRDA_5	R	H'00000000	H'E820015C	32
	Current transaction byte register 5	CRTB_5	R	H'00000000	H'E8200160	32
	Channel status register 5	CHSTAT_5	R	H'00000000	H'E8200164	32
	Channel control register 5	CHCTRL_5	RW	H'00000000	H'E8200168	32
	Channel configuration register 5	CHCFG_5	RW	H'00000000	H'E820016C	32
	Channel interval register 5	CHITVL_5	RW	H'00000000	H'E8200170	32
	Channel extension register 5	CHEXT_5	RW	H'00000000	H'E8200174	32
	Next link address register 5	NXLA_5	RW	H'00000000	H'E8200178	32
	Current link address register 5	CRLA_5	R	H'00000000	H'E820017C	32

**Table 9.2 Register Configuration**

Channel	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
6	Next0 source address register 6	N0SA_6	RW	H'00000000	H'E8200180	32
	Next0 destination address register 6	N0DA_6	RW	H'00000000	H'E8200184	32
	Next0 transaction byte register 6	N0TB_6	RW	H'00000000	H'E8200188	32
	Next1 source address register 6	N1SA_6	RW	H'00000000	H'E820018C	32
	Next1 destination address register 6	N1DA_6	RW	H'00000000	H'E8200190	32
	Next1 transaction byte register 6	N1TB_6	RW	H'00000000	H'E8200194	32
	Current source address register 6	CRSA_6	R	H'00000000	H'E8200198	32
	Current destination address register 6	CRDA_6	R	H'00000000	H'E820019C	32
	Current transaction byte register 6	CRTB_6	R	H'00000000	H'E82001A0	32
	Channel status register 6	CHSTAT_6	R	H'00000000	H'E82001A4	32
	Channel control register 6	CHCTRL_6	RW	H'00000000	H'E82001A8	32
	Channel configuration register 6	CHCFG_6	RW	H'00000000	H'E82001AC	32
	Channel interval register 6	CHITVL_6	RW	H'00000000	H'E82001B0	32
	Channel extension register 6	CHEXT_6	RW	H'00000000	H'E82001B4	32
	Next link address register 6	NXLA_6	RW	H'00000000	H'E82001B8	32
Current link address register 6	CRLA_6	R	H'00000000	H'E82001BC	32	
7	Next0 source address register 7	N0SA_7	RW	H'00000000	H'E82001C0	32
	Next0 destination address register 7	N0DA_7	RW	H'00000000	H'E82001C4	32
	Next0 transaction byte register 7	N0TB_7	RW	H'00000000	H'E82001C8	32
	Next1 source address register 7	N1SA_7	RW	H'00000000	H'E82001CC	32
	Next1 destination address register 7	N1DA_7	RW	H'00000000	H'E82001D0	32
	Next1 transaction byte register 7	N1TB_7	RW	H'00000000	H'E82001D4	32
	Current source address register 7	CRSA_7	R	H'00000000	H'E82001D8	32
	Current destination address register 7	CRDA_7	R	H'00000000	H'E82001DC	32
	Current transaction byte register 7	CRTB_7	R	H'00000000	H'E82001E0	32
	Channel status register 7	CHSTAT_7	R	H'00000000	H'E82001E4	32
	Channel control register 7	CHCTRL_7	RW	H'00000000	H'E82001E8	32
	Channel configuration register 7	CHCFG_7	RW	H'00000000	H'E82001EC	32
	Channel interval register 7	CHITVL_7	RW	H'00000000	H'E82001F0	32
	Channel extension register 7	CHEXT_7	RW	H'00000000	H'E82001F4	32
	Next link address register 7	NXLA_7	RW	H'00000000	H'E82001F8	32
Current link address register 7	CRLA_7	R	H'00000000	H'E82001FC	32	
Common for 0 to 7	DMA control registers 0 to 7	DCTRL_0_7	R/W	H'00000000	H'E8200300	32
	DMA status EN registers 0 to 7	DSTAT_EN_0_7	R	H'00000000	H'E8200310	32
	DMA status ER registers 0 to 7	DSTAT_ER_0_7	R	H'00000000	H'E8200314	32
	DMA status END registers 0 to 7	DSTAT_END_0_7	R	H'00000000	H'E8200318	32
	DMA status TC registers 0 to 7	DSTAT_TC_0_7	R	H'00000000	H'E820031C	32
	DMA status SUS registers 0 to 7	DSTAT_SUS_0_7	R	H'00000000	H'E8200320	32

**Table 9.2 Register Configuration**

Channel	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
8	Next0 source address register 8	N0SA_8	RW	H'00000000	H'E8200400	32
	Next0 destination address register 8	N0DA_8	RW	H'00000000	H'E8200404	32
	Next0 transaction byte register 8	N0TB_8	RW	H'00000000	H'E8200408	32
	Next1 source address register 8	N1SA_8	RW	H'00000000	H'E820040C	32
	Next1 destination address register 8	N1DA_8	RW	H'00000000	H'E8200410	32
	Next1 transaction byte register 8	N1TB_8	RW	H'00000000	H'E8200414	32
	Current source address register 8	CRSA_8	R	H'00000000	H'E8200418	32
	Current destination address register 8	CRDA_8	R	H'00000000	H'E820041C	32
	Current transaction byte register 8	CRTB_8	R	H'00000000	H'E8200420	32
	Channel status register 8	CHSTAT_8	R	H'00000000	H'E8200424	32
	Channel control register 8	CHCTRL_8	RW	H'00000000	H'E8200428	32
	Channel configuration register 8	CHCFG_8	RW	H'00000000	H'E820042C	32
	Channel interval register 8	CHITVL_8	RW	H'00000000	H'E8200430	32
	Channel extension register 8	CHEXT_8	RW	H'00000000	H'E8200434	32
	Next link address register 8	NXLA_8	RW	H'00000000	H'E8200438	32
	Current link address register 8	CRLA_8	R	H'00000000	H'E820043C	32
9	Next0 source address register 9	N0SA_9	RW	H'00000000	H'E8200440	32
	Next0 destination address register 9	N0DA_9	RW	H'00000000	H'E8200444	32
	Next0 transaction byte register 9	N0TB_9	RW	H'00000000	H'E8200448	32
	Next1 source address register 9	N1SA_9	RW	H'00000000	H'E820044C	32
	Next1 destination address register 9	N1DA_9	RW	H'00000000	H'E8200450	32
	Next1 transaction byte register 9	N1TB_9	RW	H'00000000	H'E8200454	32
	Current source address register 9	CRSA_9	R	H'00000000	H'E8200458	32
	Current destination address register 9	CRDA_9	R	H'00000000	H'E820045C	32
	Current transaction byte register 9	CRTB_9	R	H'00000000	H'E8200460	32
	Channel status register 9	CHSTAT_9	R	H'00000000	H'E8200464	32
	Channel control register 9	CHCTRL_9	RW	H'00000000	H'E8200468	32
	Channel configuration register 9	CHCFG_9	RW	H'00000000	H'E820046C	32
	Channel interval register 9	CHITVL_9	RW	H'00000000	H'E8200470	32
	Channel extension register 9	CHEXT_9	RW	H'00000000	H'E8200474	32
	Next link address register 9	NXLA_9	RW	H'00000000	H'E8200478	32
	Current link address register 9	CRLA_9	R	H'00000000	H'E820047C	32

**Table 9.2 Register Configuration**

Channel	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
10	Next0 source address register 10	N0SA_10	RW	H'00000000	H'E8200480	32
	Next0 destination address register 10	N0DA_10	RW	H'00000000	H'E8200484	32
	Next0 transaction byte register 10	N0TB_10	RW	H'00000000	H'E8200488	32
	Next1 source address register 10	N1SA_10	RW	H'00000000	H'E820048C	32
	Next1 destination address register 10	N1DA_10	RW	H'00000000	H'E8200490	32
	Next1 transaction byte register 10	N1TB_10	RW	H'00000000	H'E8200494	32
	Current source address register 10	CRSA_10	R	H'00000000	H'E8200498	32
	Current destination address register 10	CRDA_10	R	H'00000000	H'E820049C	32
	Current transaction byte register 10	CRTB_10	R	H'00000000	H'E82004A0	32
	Channel status register 10	CHSTAT_10	R	H'00000000	H'E82004A4	32
	Channel control register 10	CHCTRL_10	RW	H'00000000	H'E82004A8	32
	Channel configuration register 10	CHCFG_10	RW	H'00000000	H'E82004AC	32
	Channel interval register 10	CHITVL_10	RW	H'00000000	H'E82004B0	32
	Channel extension register 10	CHEXT_10	RW	H'00000000	H'E82004B4	32
	Next link address register 10	NXLA_10	RW	H'00000000	H'E82004B8	32
	Current link address register 10	CRLA_10	R	H'00000000	H'E82004BC	32
11	Next0 source address register 11	N0SA_11	RW	H'00000000	H'E82004C0	32
	Next0 destination address register 11	N0DA_11	RW	H'00000000	H'E82004C4	32
	Next0 transaction byte register 11	N0TB_11	RW	H'00000000	H'E82004C8	32
	Next1 source address register 11	N1SA_11	RW	H'00000000	H'E82004CC	32
	Next1 destination address register 11	N1DA_11	RW	H'00000000	H'E82004D0	32
	Next1 transaction byte register 11	N1TB_11	RW	H'00000000	H'E82004D4	32
	Current source address register 11	CRSA_11	R	H'00000000	H'E82004D8	32
	Current destination address register 11	CRDA_11	R	H'00000000	H'E82004DC	32
	Current transaction byte register 11	CRTB_11	R	H'00000000	H'E82004E0	32
	Channel status register 11	CHSTAT_11	R	H'00000000	H'E82004E4	32
	Channel control register 11	CHCTRL_11	RW	H'00000000	H'E82004E8	32
	Channel configuration register 11	CHCFG_11	RW	H'00000000	H'E82004EC	32
	Channel interval register 11	CHITVL_11	RW	H'00000000	H'E82004F0	32
	Channel extension register 11	CHEXT_11	RW	H'00000000	H'E82004F4	32
	Next link address register 11	NXLA_11	RW	H'00000000	H'E82004F8	32
	Current link address register 11	CRLA_11	R	H'00000000	H'E82004FC	32

**Table 9.2 Register Configuration**

Channel	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
12	Next0 source address register 12	N0SA_12	RW	H'00000000	H'E8200500	32
	Next0 destination address register 12	N0DA_12	RW	H'00000000	H'E8200504	32
	Next0 transaction byte register 12	N0TB_12	RW	H'00000000	H'E8200508	32
	Next1 source address register 12	N1SA_12	RW	H'00000000	H'E820050C	32
	Next1 destination address register 12	N1DA_12	RW	H'00000000	H'E8200510	32
	Next1 transaction byte register 12	N1TB_12	RW	H'00000000	H'E8200514	32
	Current source address register 12	CRSA_12	R	H'00000000	H'E8200518	32
	Current destination address register 12	CRDA_12	R	H'00000000	H'E820051C	32
	Current transaction byte register 12	CRTB_12	R	H'00000000	H'E8200520	32
	Channel status register 12	CHSTAT_12	R	H'00000000	H'E8200524	32
	Channel control register 12	CHCTRL_12	RW	H'00000000	H'E8200528	32
	Channel configuration register 12	CHCFG_12	RW	H'00000000	H'E820052C	32
	Channel interval register 12	CHITVL_12	RW	H'00000000	H'E8200530	32
	Channel extension register 12	CHEXT_12	RW	H'00000000	H'E8200534	32
	Next link address register 12	NXLA_12	RW	H'00000000	H'E8200538	32
	Current link address register 12	CRLA_12	R	H'00000000	H'E820053C	32
13	Next0 source address register 13	N0SA_13	RW	H'00000000	H'E8200540	32
	Next0 destination address register 13	N0DA_13	RW	H'00000000	H'E8200544	32
	Next0 transaction byte register 13	N0TB_13	RW	H'00000000	H'E8200548	32
	Next1 source address register 13	N1SA_13	RW	H'00000000	H'E820054C	32
	Next1 destination address register 13	N1DA_13	RW	H'00000000	H'E8200550	32
	Next1 transaction byte register 13	N1TB_13	RW	H'00000000	H'E8200554	32
	Current source address register 13	CRSA_13	R	H'00000000	H'E8200558	32
	Current destination address register 13	CRDA_13	R	H'00000000	H'E820055C	32
	Current transaction byte register 13	CRTB_13	R	H'00000000	H'E8200560	32
	Channel status register 13	CHSTAT_13	R	H'00000000	H'E8200564	32
	Channel control register 13	CHCTRL_13	RW	H'00000000	H'E8200568	32
	Channel configuration register 13	CHCFG_13	RW	H'00000000	H'E820056C	32
	Channel interval register 13	CHITVL_13	RW	H'00000000	H'E8200570	32
	Channel extension register 13	CHEXT_13	RW	H'00000000	H'E8200574	32
	Next link address register 13	NXLA_13	RW	H'00000000	H'E8200578	32
	Current link address register 13	CRLA_13	R	H'00000000	H'E820057C	32

**Table 9.2 Register Configuration**

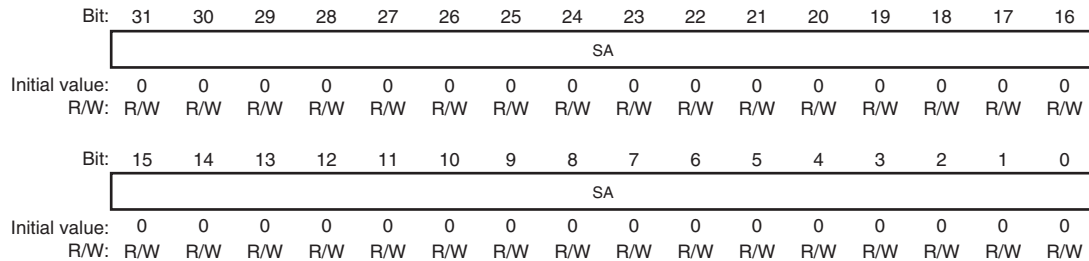
Channel	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
14	Next0 source address register 14	N0SA_14	RW	H'00000000	H'E8200580	32
	Next0 destination address register 14	N0DA_14	RW	H'00000000	H'E8200584	32
	Next0 transaction byte register 14	N0TB_14	RW	H'00000000	H'E8200588	32
	Next1 source address register 14	N1SA_14	RW	H'00000000	H'E820058C	32
	Next1 destination address register 14	N1DA_14	RW	H'00000000	H'E8200590	32
	Next1 transaction byte register 14	N1TB_14	RW	H'00000000	H'E8200594	32
	Current source address register 14	CRSA_14	R	H'00000000	H'E8200598	32
	Current destination address register 14	CRDA_14	R	H'00000000	H'E820059C	32
	Current transaction byte register 14	CRTB_14	R	H'00000000	H'E82005A0	32
	Channel status register 14	CHSTAT_14	R	H'00000000	H'E82005A4	32
	Channel control register 14	CHCTRL_14	RW	H'00000000	H'E82005A8	32
	Channel configuration register 14	CHCFG_14	RW	H'00000000	H'E82005AC	32
	Channel interval register 14	CHITVL_14	RW	H'00000000	H'E82005B0	32
	Channel extension register 14	CHEXT_14	RW	H'00000000	H'E82005B4	32
	Next link address register 14	NXLA_14	RW	H'00000000	H'E82005B8	32
Current link address register 14	CRLA_14	R	H'00000000	H'E82005BC	32	
15	Next0 source address register 15	N0SA_15	RW	H'00000000	H'E82005C0	32
	Next0 destination address register 15	N0DA_15	RW	H'00000000	H'E82005C4	32
	Next0 transaction byte register 15	N0TB_15	RW	H'00000000	H'E82005C8	32
	Next1 source address register 15	N1SA_15	RW	H'00000000	H'E82005CC	32
	Next1 destination address register 15	N1DA_15	RW	H'00000000	H'E82005D0	32
	Next1 transaction byte register 15	N1TB_15	RW	H'00000000	H'E82005D4	32
	Current source address register 15	CRSA_15	R	H'00000000	H'E82005D8	32
	Current destination address register 15	CRDA_15	R	H'00000000	H'E82005DC	32
	Current transaction byte register 15	CRTB_15	R	H'00000000	H'E82005E0	32
	Channel status register 15	CHSTAT_15	R	H'00000000	H'E82005E4	32
	Channel control register 15	CHCTRL_15	RW	H'00000000	H'E82005E8	32
	Channel configuration register 15	CHCFG_15	RW	H'00000000	H'E82005EC	32
	Channel interval register 15	CHITVL_15	RW	H'00000000	H'E82005F0	32
	Channel extension register 15	CHEXT_15	RW	H'00000000	H'E82005F4	32
	Next link address register 15	NXLA_15	RW	H'00000000	H'E82005F8	32
Current link address register 15	CRLA_15	R	H'00000000	H'E82005FC	32	
Common for 8 to 15	DMA control registers 8 to 15	DCTRL_8_15	R/W	H'00000000	H'E8200700	32
	DMA status EN registers 8 to 15	DSTAT_EN_8_15	R	H'00000000	H'E8200710	32
	DMA status ER registers 8 to 15	DSTAT_ER_8_15	R	H'00000000	H'E8200714	32
	DMA status END registers 8 to 15	DSTAT_END_8_15	R	H'00000000	H'E8200718	32
	DMA status TC registers 8 to 15	DSTAT_TC_8_15	R	H'00000000	H'E820071C	32
	DMA status SUS registers 8 to 15	DSTAT_SUS_8_15	R	H'00000000	H'E8200720	32
0/1	DMA extended resource selector 0	DMARS0	R/W	H'00000000	H'FCFE1000	32
2/3	DMA extended resource selector 1	DMARS1	R/W	H'00000000	H'FCFE1004	32
4/5	DMA extended resource selector 2	DMARS2	R/W	H'00000000	H'FCFE1008	32
6/7	DMA extended resource selector 3	DMARS3	R/W	H'00000000	H'FCFE100C	32
8/9	DMA extended resource selector 4	DMARS4	R/W	H'00000000	H'FCFE1010	32
10/11	DMA extended resource selector 5	DMARS5	R/W	H'00000000	H'FCFE1014	32
12/13	DMA extended resource selector 6	DMARS6	R/W	H'00000000	H'FCFE1018	32
14/15	DMA extended resource selector 7	DMARS7	R/W	H'00000000	H'FCFE101C	32

### 9.4.1 Next Source Address Register n (N0SA\_n, N1SA\_n)

This register sets the DMA transfer source address (32 bits) of DMA channel n (n = 0 to 15) which is to be executed next. N0SA\_n is for the Next0 Register Set, and N1SA\_n is for the Next1 Register Set.

In register mode, set this register set by using software. In link mode, the descriptor read data is automatically set in the Next0 register set.

These register set values are loaded to the Current Register Set and used for DMA transfer.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SA	All 0	R/W	Source Address Sets the start address of the DMA transfer source.

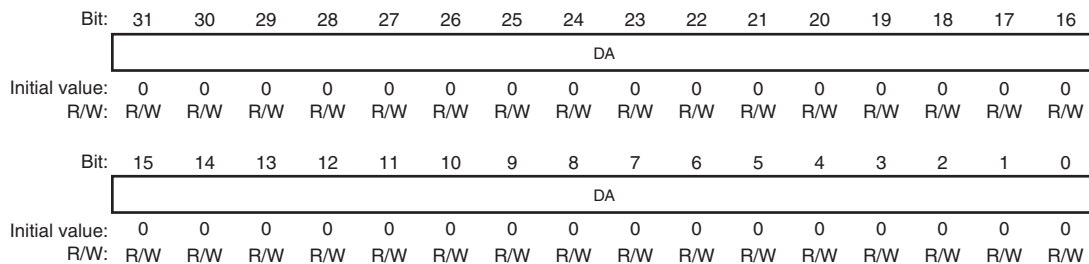
### 9.4.2 Next Destination Address Register n (N0DA\_n, N1DA\_n)

This register sets the DMA transfer destination address (32 bits) of DMA channel n (n = 0 to 15) which is to be executed next.

N0DA\_n is for the Next0 Register Set, and N1DA\_n is for the Next1 Register Set.

In register mode, set this register set by using software. In link mode, the descriptor read data is automatically set in the Next0 register set.

These register set values are loaded to the Current Register Set and used for DMA transfer.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DA	All 0	R/W	Destination Address Sets the start address of the DMA transfer destination.

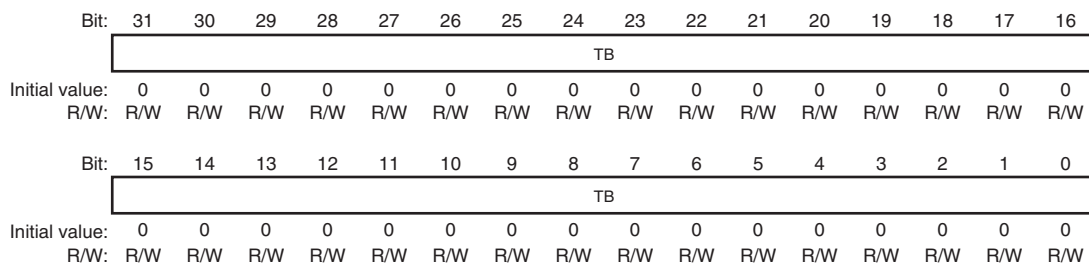
### 9.4.3 Next Transaction Byte Register n (N0TB\_n, N1TB\_n)

This register sets the total transfer byte count (DMA transaction) of DMA channel (n = 0 to 15) which is to be executed next.

N0TB\_n is for the Next0 Register Set, and N1TB\_n is for the Next1 Register Set.

In register mode, set this register set by using software. In link mode, the descriptor read data is automatically set in the Next0 register set.

These register set values are loaded to the Current Register Set and used for DMA transfer.

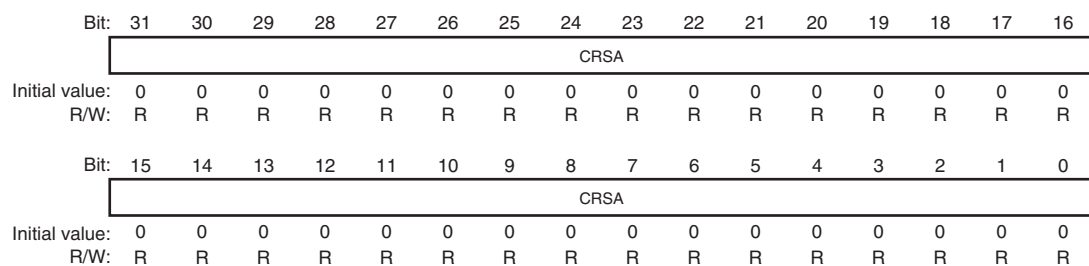


Bit	Bit Name	Initial Value	R/W	Description
31 to 0	TB	All 0	R/W	Transaction Byte Sets the total transfer byte count. Caution: Do not start a DMA transaction with 0 set in this register.

### 9.4.4 Current Source Address Register (CRSA\_n)

This register indicates the DMA transfer source address of DMA channel n (n = 0 to 15).

The values are loaded from the Next0/1 register set in register mode or from the descriptor read data in link mode. This register cannot be written by software.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CRSA	All 0	R	Current Source Address Register Indicates the read address of the next DMA transaction. The value automatically increments during the DMA transaction. (The value is fixed when 1 is set in SAD of the CHCFG_n register.) The value increments when a read transfer starts. Read this register after DMA stops (0 is set in EN of the CHSTAT_n register). (Any value obtained during the DMA operation should be handled as a reference value.)



### 9.4.5 Current Destination Address Register (CRDA\_n)

This register indicates the DMA transfer destination address of DMA channel n (n = 0 to 15).

The values are loaded from the Next0/1 register set in register mode or from the descriptor read data in link mode. This register cannot be written by software.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRDA															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CRDA															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CRDA	All 0	R	Current Destination Address Register Indicates the write address of the next DMA transaction. The value automatically increments during the DMA transaction. (The value is fixed when 1 is set in DAD of the CHCFG_n register.) The value increments when a write transfer starts. Read this register after DMA stops (0 is set in EN of the CHSTAT_n register). (Any value obtained during the DMA operation should be handled as a reference value.)

### 9.4.6 Current Transaction Byte Register (CRTB\_n)

This register indicates the total transfer byte count of DMA channel n (n = 0 to 15). The value of this register becomes 0 when the transaction ends.

The values are loaded from the Next0/1 register set in register mode or from the descriptor read data in link mode. This register cannot be written by software.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRTB															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CRTB															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CRTB	All 0	R	Current Transaction Byte Register Indicates the remaining transfer byte count of the currently executed DMA transaction. The value automatically decrements during the DMA transaction. The value decrements when a write transfer is completed. Read this register after DMA stops (0 is set in EN of the CHSTAT_n register). (Any value obtained during the DMA operation should be handled as a reference value.)

### 9.4.7 Channel Status Register n (CHSTAT\_n)

This register indicates the status of DMA channel n (n = 0 to 15).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	INTMSK
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	MODE	DER	DW	DL	SR	TC	END	ER	SUS	TACT	RQST	EN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved area. Set 0. A read operation results in 0 being read.
16	INTMSK	0	R	Indicates the temporary mask status of the DMA transfer end interrupt. 1: Masked temporarily 0: Unmasked temporarily  Set condition(s): • When SETINTMSK (CHCTRL_n) is set to 1 Reset condition(s): • When CLRINTMSK (CHCTRL_n) is set to 1 • When SWRST (CHCTRL_n) is set to 1
15 to 12	—	All 0	R	Reserved area. Set 0. A read operation results in 0 being read.
11	MODE	0	R	DMA Mode Indicates the DMA mode. It corresponds to the value set in the DMS bit of the CHCFG_n register. 0: Register mode 1: Link mode
10	DER	0	R	Descriptor Error Indicates whether the link valid value of the read descriptor is invalid (LV = 0) (this is not dependent on the DIM level of the descriptor). If a descriptor error has occurred, the transfer is stopped but no DMA error interrupt occurs. 0: Descriptor Error not detected 1: Descriptor Error detected  Set condition(s): • When the LV value loaded with the descriptor in link mode is 0 Reset condition(s): • When SWRST (CHCTRL_n) is set to 1
9	DW	0	R	Descriptor WriteBack Indicates the descriptor writeback status. The bit maintains 1 if a bus error is received during descriptor writeback. 0: Operation other than writeback is being performed for the header in link mode. 1: (ER = 0) Writeback is being performed for the header in link mode. (ER = 1) A bus error occurs during writeback for the header in link mode.  Set condition(s): • When header writeback in link mode starts Reset condition(s): • When header writeback in link mode ends with an OK response • When SWRST (CHCTRL_n) is set to 1

Bit	Bit Name	Initial Value	R/W	Description
8	DL	0	R	<p><b>Descriptor Load</b> Indicates whether the descriptor is being loaded. The bit maintains 1 if a bus error is received during descriptor load. 0: Operation other than descriptor load 1: (ER = 0) Descriptor load is in progress in link mode. (ER = 1) A bus error occurs during descriptor load in link mode.</p> <p>Set condition(s):</p> <ul style="list-style-type: none"> <li>When descriptor load in link mode starts</li> </ul> <p>Reset condition(s):</p> <ul style="list-style-type: none"> <li>When descriptor load in link mode ends with an OK response</li> <li>When SWRST (CHCTRL_n) is set to 1</li> </ul>
7	SR	0	R	<p><b>Selected Register Set</b> Indicates the register set currently selected in register mode. 0: Next0 Register Set 1: Next1 Register Set</p> <p>Set condition(s):</p> <ul style="list-style-type: none"> <li>When RSEL (CHCFG_n) is set to 1</li> </ul> <p>Reset condition(s):</p> <ul style="list-style-type: none"> <li>When RSEL (CHCFG_n) is set to 0</li> </ul>
6	TC	0	R	<p><b>Terminal Count</b> Indicates whether the DMA transaction is completed. 0: DMA transfer not completed 1: DMA transfer completed</p> <p>Set condition(s):</p> <ul style="list-style-type: none"> <li>When data equivalent to the total transfer byte count set in the CRTB register has been transferred in register mode</li> <li>When data equivalent to the total transfer byte count set in the CRTB register has been transferred in link mode, with 1 set in WBD of the descriptor header</li> <li>When descriptor writeback is completed in link mode, with 0 set in WBD of the descriptor header</li> </ul> <p>Clear condition(s):</p> <ul style="list-style-type: none"> <li>When the CLRTC (CHCTRL_n) bit is set to 1</li> <li>When the SWRST (CHCTRL_n) bit is set to 1</li> </ul>
5	END	0	R	<p><b>DMAEND Interrupted</b> Indicates whether the DMA transaction is completed and whether the DMA transfer end interrupt has occurred. 0: DMA transfer not completed 1: DMA transfer completed</p> <p>Set condition(s):</p> <ul style="list-style-type: none"> <li>When one of the set conditions for the TC bit is met and 0 is set in DEM of the CHCFG_n register</li> <li>When the descriptor is read in link mode and both LV of the header and DIM are set to 0</li> </ul> <p>Clear condition(s):</p> <ul style="list-style-type: none"> <li>When CLREND (CHCTRL_n) is set to 1</li> <li>When SWRST (CHCTRL_n) is set to 1</li> </ul>
4	ER	0	R	<p><b>Error bit</b> Indicates that a DMA error interrupt has occurred because an error response has been received from the transfer source or destination and a bus error has occurred during the DMA transfer. 0: No bus error has occurred 1: A DMA error interrupt has occurred due to a bus error</p> <p>Set condition(s):</p> <ul style="list-style-type: none"> <li>When a bus error has occurred during a bus cycle</li> </ul> <p>Clear condition(s):</p> <ul style="list-style-type: none"> <li>When SWRST (CHCTRL_n) is set to 1</li> </ul>

Bit	Bit Name	Initial Value	R/W	Description
3	SUS	0	R	<p>Suspend Indicates whether the channel is suspended. 0: Channel_n not suspended 1: Channel_n suspended</p> <p>Set condition(s):</p> <ul style="list-style-type: none"> <li>When SETSUS (CHCTRL_n) is set to 1 during a DMA transfer on Channel_n, creating a SUSPEND status internally</li> </ul> <p>Clear condition(s):</p> <ul style="list-style-type: none"> <li>When CLRSUS (CHCTRL_n) is set to 1</li> <li>When CLREN (CHCTRL_n) is set to 1</li> </ul>
2	TACT	0	R	<p>Transaction Active Indicates whether the DMAC is active. This bit is intended to check that the channel is completely inactive. 0: DMA on Channel_n inactive 1: DMA on Channel_n active</p> <p>Set condition(s):</p> <ul style="list-style-type: none"> <li>When a DMA transaction starts on Channel_n</li> </ul> <p>Clear condition(s):</p> <ul style="list-style-type: none"> <li>When a DMA transaction is completed</li> </ul>
1	RQST	0	R	<p>Request Indicates whether a transfer request is being received. 0: DMA transfer request not being received 1: DMA transfer request being received</p> <p>Set condition(s):</p> <ul style="list-style-type: none"> <li>When the STG bit (CHCTRL_n) is set to 1 (auto request)</li> <li>When a transfer request is received from the DMA request source set in the CHCFG_n register</li> </ul> <p>Clear condition(s):</p> <ul style="list-style-type: none"> <li>When SWRST (CHCTRL_n) is set to 1</li> <li>When CLRRQ (CHCTRL_n) is set to 1</li> <li>When a transfer is executed on the side specified by REQD (CHCFG_n) in single transfer mode (TM = 0).</li> <li>When all DMA transactions are completed in register mode (the transaction ends with REN set to 0)</li> <li>When the DMA transfer of the last descriptor (LE = 1) is completed in link mode</li> <li>When descriptor read stops (LV = 0) in link mode</li> <li>When a bus error is received due to an error response</li> </ul>
0	EN	0	R	<p>Enable Indicates whether the operation of DMA channel n is enabled or disabled. 0: Operation disabled 1: Operation enabled</p> <p>Set condition(s):</p> <ul style="list-style-type: none"> <li>When SETEN (CHCTRL_n) is set to 1</li> </ul> <p>Clear condition(s):</p> <ul style="list-style-type: none"> <li>When SWRST (CHCTRL_n) is set to 1</li> <li>When CLREN (CHCTRL_n) is set to 1</li> <li>When a bus error is received due to an error response during the transfer</li> <li>When all DMA transactions are completed in register mode (the transaction ends with REN set to 0)</li> <li>When the DMA transfer of the last descriptor (LE = 1) is completed in link mode (writeback when WBD is set to 0)</li> <li>When descriptor read stops (LV = 0) in link mode</li> </ul>

<Caution> If the ER bit is set to 1 for any transfer, the whole transfer should be handled as invalid.

To suspend a DMA transaction, mask or clear the transfer request or clear the Enable bit (for the procedure, see section 9.7.11 (2) Transfer Stop).

If a transfer request from an on-chip peripheral module or the external DREQ input is made concurrently with an auto request (by setting 1 in the STG bit) for the same one channel, the trigger source that takes effect cannot be identified. Make sure that only one of these transfer requests is used in the system.

When transfer is requested by an auto request, wait for the last requested DMA transfer to complete (use the Current Register or other data to check the status) before setting the STG bit for the next transfer request.

### 9.4.8 Channel Control Register n (CHCTRL\_n)

This register controls the DMA transfer operation on DMA channel n (n = 0 to 15).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CLRINT MSK	SETINT MSK
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	CLR SUS	SET SUS	-	CLRTC	CLR END	CLRRQ	SWRST	STG	CLREN	SETEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved area. Set 0. A read operation results in 0 being read.
17	CLRINTMSK	0	R/W	When this bit is set to 1, the mask of the DMA transfer end interrupt is cleared. Also, the INTMSK bit of the CHSTAT_n register is set to 0. If the mask is cleared when 1 is set in both LVINT of the DCTRL register and END of the CHSTAT_n register, the DMA transfer end interrupt becomes active. (It does not become active when 0 is set in LVINT.) A read operation results in 0 being read. 1: Clears the mask set by SETINTMSK. 0: Does not affect the operation.
16	SETINTMSK	0	R/W	When this bit is set to 1, the DMA transfer end interrupt is temporarily masked. Also, the INTMSK bit of the CHSTAT_n register is set to 1. A read operation results in 0 being read. 1: Masks the DMA transfer end interrupt. 0: Does not affect the operation.
15 to 10	—	All 0	R	Reserved area. Set 0. A read operation results in 0 being read.
9	CLRSUS	0	R/W	Clear Suspend Clears the suspend status. Setting this bit to 1 when 1 is set in SUS of the CHSTAT_n register can clear the suspend status. An attempt to read this bit results in 0 being read. 1: Clears the suspend status of the current DMA transfer. 0: Does not affect the operation.
8	SETSUS	0	R/W	Set Suspend Suspends the current DMA transfer. Setting this bit to 1 when 1 is set in EN of the CHSTAT_n register can suspend the current DMA transfer. An attempt to read this bit results in 0 being read. 1: Suspends the current DMA transfer. 0: Does not affect the operation.
7	—	0	R	Reserved area. Set 0. A read operation results in 0 being read.
6	CLRTC	0	R/W	Clear TC bit Setting this bit to 1 can clear the TC bit of the CHSTAT_n register. An attempt to read this bit results in 0 being read. 1: Clears the TC bit. 0: Does not affect the operation.
5	CLREND	0	R/W	Clear End bit Setting this bit to 1 can clear the END bit of the CHSTAT_n register. Also, the DMA transfer end interrupt is cleared. An attempt to read this bit results in 0 being read. 1: Clears the END bit. 0: Does not affect the operation.
4	CLRRQ	0	R/W	Clear Request bit Setting this bit to 1 can clear the RQST bit of the CHSTAT_n register. An attempt to read this bit results in 0 being read. 1: Clears the RQST bit. 0: Does not affect the operation.

Bit	Bit Name	Initial Value	R/W	Description
3	SWRST	0	R/W	<p>Software Reset</p> <p>Setting this bit to 1 can clear the channel status register (CHSTAT_n). When setting this bit to 1, make sure that both the EN bit and TACT bit are set to 0.</p> <p>An attempt to read this bit results in 0 being read.</p> <p>1: Resets the channel status register. 0: Does not affect the operation.</p>
2	STG	0	R/W	<p>Software Trigger</p> <p>Setting this bit to 1 sets an auto request. If this bit is set at the same time the SWRST bit is set, the clear operation by the SWRST bit takes precedence.</p> <p>An attempt to read this bit results in 0 being read.</p> <p>1: Sets a transfer request triggered by an auto request (sets 1 in the RQST bit). 0: Does not affect the operation.</p>
1	CLREN	0	R/W	<p>Clear Enable</p> <p>Setting this bit to 1 can clear the EN bit (for details, see section 9.7.11 (2) Transfer Stop).</p> <p>An attempt to read this bit results in 0 being read.</p> <p>1: Stops the DMA transfer (clears the EN bit). 0: Does not affect the operation.</p>
0	SETEN	0	R/W	<p>Set Enable</p> <p>Enables a DMA transfer on DMA channel n. If this bit is set at the same time the SWRST bit is set, the clear operation by the SWRST bit takes precedence and the transfer does not start.</p> <p>An attempt to read this bit results in 0 being read.</p> <p>1: Enables a DMA transfer (sets 1 in the EN bit). 0: Does not affect the operation.</p>

### 9.4.9 Channel Configuration Register n (CHCFG\_n)

This register controls the DMA transfer operation on DMA channel n (n = 0 to 15).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DMS	REN	RSW	RSEL	SBE	-	-	DEM	-	TM	DAD	SAD	DDS[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDS[3:0]			-	AM[2:0]			-	LVL	HIEN	LOEN	REQD	SEL[2:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	DMS	0	R/W	<b>DMA Mode Select</b> Sets the DMA mode. 0: Register mode (initial value) 1: Link mode
30	REN	0	R/W	<b>Register Set Enable</b> After a DMA transaction is completed, DMA transfers are continued using the Next register set selected by RSEL. This bit is valid only in register mode. 0: Does not continue DMA transfers. 1: Continues DMA transfers.  Set condition(s): <ul style="list-style-type: none"> <li>When 1 is written to this bit</li> </ul> Clear condition(s): <ul style="list-style-type: none"> <li>When 0 is written to this bit</li> <li>When a DMA transaction is completed, with REN set to 1</li> </ul>
29	RSW	0	R/W	<b>Register Select Switch</b> Inverts RSEL automatically after a DMA transaction is completed. This bit is valid only in register mode. 0: Does not invert RSEL automatically after a DMA transaction (initial value). 1: Inverts RSEL automatically after a DMA transaction.
28	RSEL	0	R/W	<b>Register Set Select</b> Selects the Next register set to be executed next. This bit is valid only in register mode. When RSW is set to 1, this bit is inverted automatically when a DMA transaction is completed. 0: Executes the Next0 Register Set (initial value). 1: Executes the Next1 Register Set.  Transition condition(s): <ul style="list-style-type: none"> <li>When a DMA transaction is completed, with RSW set to 1</li> </ul>
27	SBE	0	R/W	<b>Sweep Buffer Enable</b> Selects whether to sweep (write) the data already read into the buffer and stop the DMA transfer if the Enable bit is cleared to 0 during a DMA transaction. The sweep mode is available only when REQD is set to 0. 0: Stops the DMA transfer without sweeping the buffer (initial value). 1: Stops the DMA transfer after sweeping the buffer.
26, 25	—	0	R	Reserved area. Set 0. A read operation results in 0 being read.
24	DEM	0	R/W	<b>DMA Transfer End Interrupt Mask</b> Masks the DMA transfer end interrupt for register mode transfer. If 1 is set in this bit when a DMA transfer end interrupt is output, the DMA transfer end interrupt signal is not asserted. In this case, DEM is cleared to 0 automatically. 0: Does not mask the DMA transfer end interrupt (initial value). 1: Masks the DMA transfer end interrupt.  Clear condition(s): <ul style="list-style-type: none"> <li>When a DMA transaction is completed with DEM set to 1</li> </ul>
23	—	0	R	Reserved area. Set 0. A read operation results in 0 being read.

Bit	Bit Name	Initial Value	R/W	Description
22	TM	0	R/W	Transfer Mode Sets the DMA transfer mode. 0: Single transfer mode (initial value) 1: Block transfer mode
21	DAD	0	R/W	Sets the destination address counting direction of DMA channel n. 0: Increment (initial value) 1: Fixed
20	SAD	0	R/W	Sets the source address counting direction of DMA channel n. 0: Increment (initial value) 1: Fixed
19 to 16	DDS [3:0]	0000	R/W	Destination Data Size Sets the DMA transfer size of the transfer destination.
				<b>Value</b>
				<b>Size</b>
				<b>Remark</b>
				0000
				8 bits
				Initial value
				0001
				16 bits
				0010
				32 bits
				0011
				64 bits
				0100
				128 bits
				0101
				256 bits
				0110
				512 bits
				0111
				1024 bits
				Other than the above
				—
				Setting prohibited
15 to 12	SDS [3:0]	0000	R/W	Source Data Size Sets the DMA transfer size of the transfer source.
				<b>Value</b>
				<b>Size</b>
				<b>Remark</b>
				0000
				8 bits
				Initial value
				0001
				16 bits
				0010
				32 bits
				0011
				64 bits
				0100
				128 bits
				0101
				256 bits
				0110
				512 bits
				0111
				1024 bits
				Other than the above
				—
				Setting prohibited
11	—	0	R	Reserved area. Set 0. A read operation results in 0 being read.
10 to 8	AM [2:0]	000	R/W	ACK Mode Sets the DMAACK output mode. 000: (initial value) 001: Level mode (active until the transfer request from an on-chip peripheral module or the external DREQ input becomes inactive) 01x: Bus cycle mode (active while the DMA transfer is in a bus cycle) 1xx: DMAACK not to be output (this setting should be made when an auto request is made by STG (CHCTRL_n))
7	—	0	R	Reserved area. Set 0. A read operation results in 0 being read.
6	LVL	0	R/W	Level Selects whether to detect a DMA request based on the level or edge of the signal. 0: Detects based on the edge (initial value). 1: Detects based on the level.

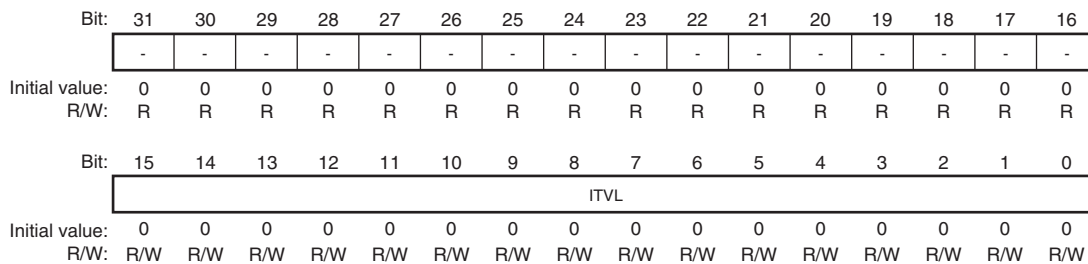


Bit	Bit Name	Initial Value	R/W	Description
5	HIEN	0	R/W	<b>High Enable</b> Selects whether to detect a DMA request using the High level or rising edge of the signal. When LVL = 0: HIEN = 1: Detects a request in response to the rising edge of the signal. HIEN = 0: Does not detect a request in response to the rising edge of the signal (initial value). When LVL = 1: HIEN = 1: Detects a request when the signal is at the High level. HIEN = 0: Does not detect a request even when the signal is at the High level (initial value).
4	LOEN	0	R/W	<b>Low Enable</b> Selects whether to detect a DMA request using the Low level or falling edge of the signal. When LVL = 0: LOEN = 1: Detects a request in response to the falling edge of the signal. LOEN = 0: Does not detect a request in response to the falling edge of the signal (initial value). When LVL = 1: LOEN = 1: Detects a request when the signal is at the Low level. LOEN = 0: Does not detect a request even when the signal is at the Low level (initial value).
3	REQD	0	R/W	<b>Request Direction</b> Selects whether DMAREQ selected by the SEL bit is the source or destination. This bit is also used to define when DMAACK is to become active. 0: Source; DMAACK is to become active when read (initial value). 1: Destination; DMAACK is to become active when written.
2 to 0	SEL[2:0]	000	R/W	These bits are used to set a DMAC channel. Set one of the following values so that the channel set by the SEL bits matches the CHCFG_n channel. 000: CH0/CH8 001: CH1/CH9 010: CH2/CH10 011: CH3/CH11 100: CH4/CH12 101: CH5/CH13 110: CH6/CH14 111: CH7/CH15

#### 9.4.10 Channel Interval Register n (CHITVL\_n)

This register sets the transfer interval for DMA channel n (n = 0 to 15).

For details, see section 9.7.9, Interval Count Function.



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved area. Set 0. A read operation results in 0 being read.
15 to 0	ITVL	All 0	R/W	Sets the channel transfer interval.

### 9.4.11 Channel Extension Register n (CHEXT\_n)

This is an extension register for DMA channel n (n = 0 to 15).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCA[3:0]				-	-	-	-	SCA[3:0]				-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Set 0. A read operation results in 0 being read.
15 to 12	DCA[3:0]	0000	R/W	Destination CACHE Sets the value to be output to AWCACHE[3:0] for DMA write transfer. See Note below.
11	—	0	R	Reserved area. Set 0. The initial value is 0.
10 to 8	—	000	R	Reserved area. Set 0. The initial value is 0.
7 to 4	SCA[3:0]	0000	R/W	Source CACHE Sets the value to be output to ARCACHE[3:0] for DMA read transfer. See Note below.
3	—	0	R	Reserved area. Set 0. The initial value is 0.
2 to 0	—	000	R	Reserved area. Set 0. The initial value is 0.

Note 1. When the transfer destination or source is not in the external bus space, set these bits to 0000.

Even when the transfer destination or source is in the external bus space but the secondary cache is not in use, set these bits to 0000. In this case, the DACK0 output and TEND0 output are issued in response to the DREQ0 transfer request.

When the secondary cache is in use in the external bus space, set CACHE[3:0].

For the setting value, see AMBA AXI Protocol Specification from ARM Limited.

### 9.4.12 Next Link Address Register n (NXLA\_n)

This is a 32-bit register that sets the link address of DMA channel n (n = 0 to 15).

For information about the link mode, see section 9.6.3, Link Mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	NXLA															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NXLA															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	NXLA	All 0	R/W	Sets a link address. The low-order 2 bits are masked with 0s. Only an address aligned with a 4-byte boundary can be set.

### 9.4.13 Current Link Address Register n (CRLA\_n)

This is a 32-bit register that indicates the link address of DMA channel n (n = 0 to 15).

For information about the link mode, see section 9.6.3, Link Mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRLA															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CRLA															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CRLA	All 0	R	Indicates the address of the currently executed descriptor.

### 9.4.14 DMA Control Register (DCTRL\_0\_7, DCTRL\_8\_15)

This register sets the transfer type for descriptor access and the arbitration between channels.

(DCTRL\_0\_7 is common for channels 0 to 7 and DCTRL\_8\_15 is common for channels 8 to 15.)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LWCA				-	-	-	-	LDCA				-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LVINT	PR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	LWCA	0000	R/W	Link WriteBack CACHE Sets the value to be output to AWCACHE[3:0] during descriptor writeback in link mode. For the setting value, see Note in section 9.4.11, Channel Extension Register n (CHEXT_n).
27 to 24	—	0	R	Reserved area. Set 0. The initial value is 0.
23 to 20	LDCA	0000	R/W	Link Descriptor CACHE Sets the value to be output to ARCACHE[3:0] during descriptor load in link mode. For the setting value, see Note in section 9.4.11, Channel Extension Register n (CHEXT_n).
19 to 16	—	All 0	R	Reserved area. Set 0. The initial value is 0.
15 to 2	—	All 0	R	Reserved area. Set 0. A read operation results in 0 being read.
1	LVINT	0	R/W	Sets whether to use pulse output or level output for the DMA transfer end interrupt and DMA error interrupt. Set pulse output for this product. 0: Pulse output (initial value) 1: Level output
0	PR	0	R/W	Sets the transfer priority control mode between channels (see section 9.7.2, Priority Control for DMA Channels). 0: Fixed priority mode (initial value) 1: Round robin mode

### 9.4.15 DMA Status EN Register (DSTAT\_EN\_0\_7)

This register indicates the EN bit status of the CHSTAT\_n register (n = 0 to 7).

Writing to this register does not affect the values of the bits.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	EN7	EN6	EN5	EN4	EN3	EN2	EN1	EN0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved area. Set 0. A read operation results in 0 being read.
7	EN7	0	R	Indicates the EN bit status of DMA channel 7.
6	EN6	0	R	Indicates the EN bit status of DMA channel 6.
5	EN5	0	R	Indicates the EN bit status of DMA channel 5.
4	EN4	0	R	Indicates the EN bit status of DMA channel 4.
3	EN3	0	R	Indicates the EN bit status of DMA channel 3.
2	EN2	0	R	Indicates the EN bit status of DMA channel 2.
1	EN1	0	R	Indicates the EN bit status of DMA channel 1.
0	EN0	0	R	Indicates the EN bit status of DMA channel 0.

### 9.4.16 DMA Status EN Register (DSTAT\_EN\_8\_15)

This register indicates the EN bit status of the CHSTAT\_n register (n = 8 to 15).

Writing to this register does not affect the values of the bits.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	EN15	EN14	EN13	EN12	EN11	EN10	EN9	EN8
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved area. Set 0. A read operation results in 0 being read.
7	EN15	0	R	Indicates the EN bit status of DMA channel 15.
6	EN14	0	R	Indicates the EN bit status of DMA channel 14.
5	EN13	0	R	Indicates the EN bit status of DMA channel 13.
4	EN12	0	R	Indicates the EN bit status of DMA channel 12.
3	EN11	0	R	Indicates the EN bit status of DMA channel 11.
2	EN10	0	R	Indicates the EN bit status of DMA channel 10.
1	EN9	0	R	Indicates the EN bit status of DMA channel 9.
0	EN8	0	R	Indicates the EN bit status of DMA channel 8.

### 9.4.17 DMA Status ER Register (DSTAT\_ER\_0\_7)

This register indicates the ER bit status of the CHSTAT\_n register (n = 0 to 7).

Writing to this register does not affect the values of the bits.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	ER7	ER6	ER5	ER4	ER3	ER2	ER1	ER0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved area. Set 0. A read operation results in 0 being read.
7	ER7	0	R	Indicates the ER bit status of DMA channel 7.
6	ER6	0	R	Indicates the ER bit status of DMA channel 6.
5	ER5	0	R	Indicates the ER bit status of DMA channel 5.
4	ER4	0	R	Indicates the ER bit status of DMA channel 4.
3	ER3	0	R	Indicates the ER bit status of DMA channel 3.
2	ER2	0	R	Indicates the ER bit status of DMA channel 2.
1	ER1	0	R	Indicates the ER bit status of DMA channel 1.
0	ER0	0	R	Indicates the ER bit status of DMA channel 0.

### 9.4.18 DMA Status ER Register (DSTAT\_ER\_8\_15)

This register indicates the ER bit status of the CHSTAT\_n register (n = 8 to 15).

Writing to this register does not affect the values of the bits.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	ER15	ER14	ER13	ER12	ER11	ER10	ER9	ER8
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved area. Set 0. A read operation results in 0 being read.
7	ER15	0	R	Indicates the ER bit status of DMA channel 15.
6	ER14	0	R	Indicates the ER bit status of DMA channel 14.
5	ER13	0	R	Indicates the ER bit status of DMA channel 13.
4	ER12	0	R	Indicates the ER bit status of DMA channel 12.
3	ER11	0	R	Indicates the ER bit status of DMA channel 11.
2	ER10	0	R	Indicates the ER bit status of DMA channel 10.
1	ER9	0	R	Indicates the ER bit status of DMA channel 9.
0	ER8	0	R	Indicates the ER bit status of DMA channel 8.

### 9.4.19 DMA Status END Register (DSTAT\_END\_0\_7)

This register indicates the END bit status of the CHSTAT\_n register (n = 0 to 7).

Writing to this register does not affect the values of the bits.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	END7	END6	END5	END4	END3	END2	END1	END0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved area. Set 0. A read operation results in 0 being read.
7	END7	0	R	Indicates the END bit status of DMA channel 7.
6	END6	0	R	Indicates the END bit status of DMA channel 6.
5	END5	0	R	Indicates the END bit status of DMA channel 5.
4	END4	0	R	Indicates the END bit status of DMA channel 4.
3	END3	0	R	Indicates the END bit status of DMA channel 3.
2	END2	0	R	Indicates the END bit status of DMA channel 2.
1	END1	0	R	Indicates the END bit status of DMA channel 1.
0	END0	0	R	Indicates the END bit status of DMA channel 0.

### 9.4.20 DMA Status END Register (DSTAT\_END\_8\_15)

This register indicates the END bit status of the CHSTAT\_n register (n = 8 to 15).

Writing to this register does not affect the values of the bits.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	END15	END14	END13	END12	END11	END10	END9	END8
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved area. Set 0. A read operation results in 0 being read.
7	END15	0	R	Indicates the END bit status of DMA channel 15.
6	END14	0	R	Indicates the END bit status of DMA channel 14.
5	END13	0	R	Indicates the END bit status of DMA channel 13.
4	END12	0	R	Indicates the END bit status of DMA channel 12.
3	END11	0	R	Indicates the END bit status of DMA channel 11.
2	END10	0	R	Indicates the END bit status of DMA channel 10.
1	END9	0	R	Indicates the END bit status of DMA channel 9.
0	END8	0	R	Indicates the END bit status of DMA channel 8.

### 9.4.21 DMA Status TC Register (DSTAT\_TC\_0\_7)

This register indicates the TC bit status of the CHSTAT\_n register (n = 0 to 7).

Writing to this register does not affect the values of the bits.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	TC7	TC6	TC5	TC4	TC3	TC2	TC1	TC0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved area. Set 0. A read operation results in 0 being read.
7	TC7	0	R	Indicates the TC bit status of DMA channel 7.
6	TC6	0	R	Indicates the TC bit status of DMA channel 6.
5	TC5	0	R	Indicates the TC bit status of DMA channel 5.
4	TC4	0	R	Indicates the TC bit status of DMA channel 4.
3	TC3	0	R	Indicates the TC bit status of DMA channel 3.
2	TC2	0	R	Indicates the TC bit status of DMA channel 2.
1	TC1	0	R	Indicates the TC bit status of DMA channel 1.
0	TC0	0	R	Indicates the TC bit status of DMA channel 0.

### 9.4.22 DMA Status TC Register (DSTAT\_TC\_8\_15)

This register indicates the TC bit status of the CHSTAT\_n register (n = 8 to 15).

Writing to this register does not affect the values of the bits.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	TC15	TC14	TC13	TC12	TC11	TC10	TC9	TC8
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved area. Set 0. A read operation results in 0 being read.
7	TC15	0	R	Indicates the TC bit status of DMA channel 15.
6	TC14	0	R	Indicates the TC bit status of DMA channel 14.
5	TC13	0	R	Indicates the TC bit status of DMA channel 13.
4	TC12	0	R	Indicates the TC bit status of DMA channel 12.
3	TC11	0	R	Indicates the TC bit status of DMA channel 11.
2	TC10	0	R	Indicates the TC bit status of DMA channel 10.
1	TC9	0	R	Indicates the TC bit status of DMA channel 9.
0	TC8	0	R	Indicates the TC bit status of DMA channel 8.

### 9.4.23 DMA Status SUS Register (DSTAT\_SUS\_0\_7)

This register indicates the SUS bit status of the CHSTAT<sub>n</sub> register (n = 0 to 7).

Writing to this register does not affect the values of the bits.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	SUS7	SUS6	SUS5	SUS4	SUS3	SUS2	SUS1	SUS0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved area. Set 0. A read operation results in 0 being read.
7	SUS7	0	R	Indicates the SUS bit status of DMA channel 7.
6	SUS6	0	R	Indicates the SUS bit status of DMA channel 6.
5	SUS5	0	R	Indicates the SUS bit status of DMA channel 5.
4	SUS4	0	R	Indicates the SUS bit status of DMA channel 4.
3	SUS3	0	R	Indicates the SUS bit status of DMA channel 3.
2	SUS2	0	R	Indicates the SUS bit status of DMA channel 2.
1	SUS1	0	R	Indicates the SUS bit status of DMA channel 1.
0	SUS0	0	R	Indicates the SUS bit status of DMA channel 0.

### 9.4.24 DMA Status SUS Register (DSTAT\_SUS\_8\_15)

This register indicates the SUS bit status of the CHSTAT<sub>n</sub> register (n = 8 to 15).

Writing to this register does not affect the values of the bits.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	SUS15	SUS14	SUS13	SUS12	SUS11	SUS10	SUS9	SUS8
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved area. Set 0. A read operation results in 0 being read.
7	SUS15	0	R	Indicates the SUS bit status of DMA channel 15.
6	SUS14	0	R	Indicates the SUS bit status of DMA channel 14.
5	SUS13	0	R	Indicates the SUS bit status of DMA channel 13.
4	SUS12	0	R	Indicates the SUS bit status of DMA channel 12.
3	SUS11	0	R	Indicates the SUS bit status of DMA channel 11.
2	SUS10	0	R	Indicates the SUS bit status of DMA channel 10.
1	SUS9	0	R	Indicates the SUS bit status of DMA channel 9.
0	SUS8	0	R	Indicates the SUS bit status of DMA channel 8.



### 9.4.25 DMA Extension Resource Selectors 0 to 7 (DMARS0 to DMARS7)

DMARS are 32-bit readable/writable registers that specify the DMA transfer sources from peripheral modules in each channel. DMARS0 is for channels 0 and 1, DMARS1 is for channels 2 and 3, and so on.

Table 9.4 shows the specifiable combinations.

DMARS can specify transfer requests to be accepted for the following triggers.

The following modules can issue on-chip peripheral module requests.

Serial communication interface with FIFO: 16 sources

A/D converter: 1 source

Multi-function timer pulse unit 2: 5 sources

USB2.0 host/function module: 4 sources

NAND flash memory controller: 1 source

Serial sound interface: 10 sources

Sound generator: 4 sources

Renesas SPDIF interface: 2 sources

CD-ROM decoder: 1 source

SD host interface: 4 sources

MMC host interface: 2 sources

Renesas serial peripheral interface: 10 sources

Motor control PWM timer: 2 sources

IEBus™ controller: 2 sources

OS timer: 2 sources

SCUX: 8 sources

Media local bus: 1 source

Serial communication interface: 4 sources

I<sup>2</sup>C bus interface: 8 sources

LIN interface: 4 sources

Pixel format converter: 4 sources

Some on-chip peripheral modules in this product use the same signal both for an interrupt request and for a DMA transfer request. If such a module is selected by a DMARS register, the signal works as a DMA transfer request signal and interrupt requests to the interrupt controller are masked. To enable the interrupt, clear the setting of DMARS (set all MID[6:0] and RID[1:0] to 0).

- DMARS0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	CH1 MID[6:0]						CH1 RID[1:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	CH0 MID[6:0]						CH0 RID[1:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- DMARS1

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	CH3 MID[6:0]						CH3 RID[1:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	CH2 MID[6:0]						CH2 RID[1:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- DMARS2

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	
	-	-	-	-	-	-	-	CH5 MID[6:0]						CH5 RID[1:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Bit:	15	-	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-		-	-	-	-	-	CH4 MID[6:0]						CH4 RID[1:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- DMARS3

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	CH7 MID[6:0]						CH7 RID[1:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	CH6 MID[6:0]						CH6 RID[1:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- DMARS4

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	CH9 MID[6:0]						CH9 RID[1:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	CH8 MID[6:0]						CH8 RID[1:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- DMARS5

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	CH11 MID[6:0]						CH11 RID[1:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	CH10 MID[6:0]						CH10 RID[1:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- DMARS6

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	CH13 MID[6:0]						CH13 RID[1:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	CH12 MID[6:0]						CH12 RID[1:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

- DMARS7

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	CH15 MID[6:0]						CH15 RID[1:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	CH14 MID[6:0]						CH14 RID[1:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

## 9.5 Operation

When there is a DMA transfer request, the DMAC starts the transfer according to the predetermined channel priority order; when the transfer end conditions are satisfied, it ends the transfer. Transfers can be requested in three modes: auto request, external request, and on-chip peripheral module request.

### 9.5.1 Transfer Flow

After the next source address register (N0SA\_n/N1SA\_n), next destination address register (N0DA\_n/N1DA\_n), next transaction byte register (N0TB\_n/N1TB\_n), channel control register (CHCTRL\_n), channel configuration register (CHCFG\_n), channel extension register (CHEXT\_n), DMA control register (DCTRL\_0\_7/DCTRL\_8\_15), and DMA extension resource selector (DMARS) are set for the target transfer conditions, the DMAC transfers data according to the following procedure:

1. Checks to see if transfer is enabled (EN = 0 and TACT = 0 in channel status register).
2. Clears the channel status register (set 1 in the SWRST bit of the channel control register).
3. Enables DMA transfer (set 1 in the SETEN bit of the channel control register).
4. When a transfer request comes and transfer is enabled, the DMAC transfers one transfer unit of data (depending on the DDS[3:0] and SDS[3:0] bit settings). For an auto request, the transfer begins automatically when 1 is set in the STG bit of the channel control register. The CRTB\_n value will be decremented by 1 for each transfer.
5. If 0 is set in the REN bit of the channel configuration register when transfer has been completed for the specified count (when CRTB\_n reaches 0), transfer ends normally. If the DEM bit of the channel configuration register is set to 0 at this time, a DMA transfer end interrupt is sent to the CPU. If the REN bit is 1 when CRTB\_n reaches 0, transfer operations are continued with the values of N0SA\_n/N1SA\_n, N0DA\_n/N1DA\_n, and N0TB\_n/N1TB\_n set by the RSEL bit of the channel configuration register until there are no more transfer requests.
6. When an address error in the DMAC is generated, the transfer is stopped. Transfers are also stopped when 1 is set in the CLREN bit of CHCTRL\_n.

### 9.5.2 DMA Transfer Requests

DMA transfer requests are basically generated in either the data transfer source or destination, but they can also be generated in external devices and on-chip peripheral modules that are neither the transfer source nor destination. Transfers can be requested in three modes: auto request, external request, and on-chip peripheral module request. External request or on-chip peripheral module request is selected by the DMARS0 to DMARS7 registers.

#### (1) Auto-Request Mode

When there is no transfer request signal from an external source, as in a memory-to-memory transfer or a transfer between memory and an on-chip peripheral module unable to request a transfer, auto-request mode allows the DMAC to automatically generate a transfer request signal internally. When the STG bit in channel control register n is set to 1, the transfer begins so long as the TACT bit in channel status register is 0.

#### (2) External Request Mode

In this mode a transfer is performed at the transfer request signal (DREQ0) of an external device of the LSI. When the DMA transfer is enabled, DMA transfer is performed upon a DREQ input.

Choose to detect DREQ0 by either the edge or level of the signal input with the LVL, HIEN, and LOEN bits in channel configuration register 0 as shown below. The source of the transfer request does not have to be the data transfer source or destination.

For the output level settings of the DACK0 and TEND0 pins, refer to section 8, Bus State Controller.

**Table 9.3 Settings for External Request Detection**

CHCFG_0			Detection of External Request
LVL	HIEN	LOEN	
0	0	1	Falling edge detection
	1	0	Rising edge detection
1	0	1	Low level detection
	1	0	High level detection

When DREQ0 is accepted, the DREQ0 pin enters the request accept disabled state (non-sensitive period). After issuing an acknowledge DACK0 signal for the accepted DREQ0, the DREQ0 pin again enters the request accept enabled state.

### (3) On-Chip Peripheral Module Request Mode

In this mode, the transfer is performed in response to the DMA transfer request signal from an on-chip peripheral module.

When a transfer request signal is sent in on-chip peripheral module request mode while DMA transfer is enabled, the DMA transfer is performed.

The DMA transfer request signals to be sent from on-chip peripheral modules or external pin input are listed in Table 9.4.

The transfer source or destination is fixed for some on-chip peripheral module requests. For details, see Table 9.4.

**Table 9.4 On-Chip Peripheral Module Requests**

DMA Transfer Request Source	DMA Transfer Request Signal	Transfer Source	Transfer Destination	DMARS		CHCFG_n										
				MID	RID	TM	AM [2:0]	LVL	HIEN	LO EN	RE QD	SEL[2:0]				
OS timer channel 0	OSTM0TINT (compare match)	Arbitrary	Arbitrary	000_1000	11	0/1	010	0	1	0	0/1	Ch0: 000 Ch1: 001 Ch2: 010 Ch3: 011 Ch4: 100 Ch5: 101 Ch6: 110 Ch7: 111 Ch8: 000 Ch9: 001 Ch10: 010 Ch11: 011 Ch12: 100 Ch13: 101 Ch14: 110 Ch15: 111				
OS timer channel 1	OSTM1TINT (compare match)	Arbitrary	Arbitrary	000_1001	11											
Multi-function timer pulse unit 2 channel 0	TGIA_0 (input capture/compare match)	Arbitrary	Arbitrary	001_0000	11								0/1	001	1	0/1
Multi-function timer pulse unit 2 channel 1	TGIA_1 (input capture/compare match)	Arbitrary	Arbitrary	001_0001	11											
Multi-function timer pulse unit 2 channel 2	TGIA_2 (input capture/compare match)	Arbitrary	Arbitrary	001_0010	11											
Multi-function timer pulse unit 2 channel 3	TGIA_3 (input capture/compare match)	Arbitrary	Arbitrary	001_0011	11											
Multi-function timer pulse unit 2 channel 4	TGIA_4 (input capture/compare match)	Arbitrary	Arbitrary	001_0100	11											
Serial communication interface with FIFO channel 0	TXI0 (transmit empty)	Arbitrary	SCFTDR_0	001_1000	01	0	010	1			1					
	RXI0 (receive data full)	SCFRDR_0	Arbitrary		10						0					
Serial communication interface with FIFO channel 1	TXI1 (transmit empty)	Arbitrary	SCFTDR_1	001_1001	01						1					
	RXI1 (receive data full)	SCFRDR_1	Arbitrary		10						0					
Serial communication interface with FIFO channel 2	TXI2 (transmit empty)	Arbitrary	SCFTDR_2	001_1010	01						1					
	RXI2 (receive data full)	SCFRDR_2	Arbitrary		10						0					

Table 9.4 On-Chip Peripheral Module Requests

DMA Transfer Request Source	DMA Transfer Request Signal	Transfer Source	Transfer Destination	DMARS		CHCFG_n						SEL[2:0]					
				MID	RID	TM	AM [2:0]	LVL	HIEN	LO EN	RE QD						
Serial communication interface with FIFO channel 3	TXI3 (transmit empty)	Arbitrary	SCFTDR_3	001_1011	01	0	010	1	1	0	1	Ch0: 000 Ch1: 001 Ch2: 010 Ch3: 011 Ch4: 100 Ch5: 101 Ch6: 110 Ch7: 111 Ch8: 000 Ch9: 001 Ch10: 010 Ch11: 011 Ch12: 100 Ch13: 101 Ch14: 110 Ch15: 111					
	RXI3 (receive data full)	SCFRDR_3	Arbitrary		10						0						
Serial communication interface with FIFO channel 4	TXI4 (transmit empty)	Arbitrary	SCFTDR_4	001_1100	01												
	RXI4 (receive data full)	SCFRDR_4	Arbitrary		10						0						
Serial communication interface with FIFO channel 5	TXI5 (transmit empty)	Arbitrary	SCFTDR_5	001_1101	01												
	RXI5 (receive data full)	SCFRDR_5	Arbitrary		10						0						
Serial communication interface with FIFO channel 6	TXI6 (transmit empty)	Arbitrary	SCFTDR_6	001_1110	01												
	RXI6 (receive data full)	SCFRDR_6	Arbitrary		10						0						
Serial communication interface with FIFO channel 7	TXI7 (transmit empty)	Arbitrary	SCFTDR_7	001_1111	01												
	RXI7 (receive data full)	SCFRDR_7	Arbitrary		10						0						
USB2.0 host/function module channel 0	USB0_DMA0 (channel 0 transmit FIFO empty)	Arbitrary	D0FIFO_0 D0FIFOBn_0 (n = 0 to 7)	010_0000	11						010		1				1
	USB0_DMA0 (channel 0 receive FIFO full)	D0FIFO_0 D0FIFOBn_0 (n = 0 to 7)	Arbitrary														0
	USB0_DMA1 (channel 1 transmit FIFO empty)	Arbitrary	D1FIFO_0 D1FIFOBn_0 (n = 0 to 7)	010_0001	11												1
	USB0_DMA1 (channel 1 receive FIFO full)	D1FIFO_0 D1FIFOBn_0 (n = 0 to 7)	Arbitrary														0

Table 9.4 On-Chip Peripheral Module Requests

DMA Transfer Request Source	DMA Transfer Request Signal	Transfer Source	Transfer Destination	DMARS		CHCFG_n						SEL[2:0]	
				MID	RID	TM	AM [2:0]	LVL	HIEN	LO EN	RE QD		
USB2.0 host/function module channel 1	USB1_DMA0 (channel 0 transmit FIFO empty)	Arbitrary	D0FIFO_1 D0FIFO <sub>n-1</sub> (n = 0 to 7)	010_0010	11	0	001	1	1	0	1	Ch0: 000 Ch1: 001 Ch2: 010 Ch3: 011 Ch4: 100 Ch5: 101 Ch6: 110 Ch7: 111 Ch8: 000 Ch9: 001 Ch10: 010 Ch11: 011 Ch12: 100 Ch13: 101 Ch14: 110 Ch15: 111	
	USB1_DMA0 (channel 0 receive FIFO full)	D0FIFO_1 D0FIFO <sub>n-1</sub> (n = 0 to 7)	Arbitrary								0		
	USB1_DMA1 (channel 1 transmit FIFO empty)	Arbitrary	D1FIFO_1 D1FIFO <sub>n-1</sub> (n = 0 to 7)	010_0011	11						1		
	USB1_DMA1 (channel 1 receive FIFO full)	D1FIFO_1 D1FIFO <sub>n-1</sub> (n = 0 to 7)	Arbitrary								0		
A/D converter	ADI (A/D conversion end)	ADDR	Arbitrary	010_0100	11		001	1			0		
IEBus™ controller	IEBBTD (data interrupt) (during transmission in single mode)	Arbitrary	IEBB0DR	010_1000	11	0	010	0	1	0			
	IEBBTD (data interrupt) (during reception in single mode)	IEBB0DR	Arbitrary										
	IEBBTD (data interrupt) (during transmission in FIFO mode)	Arbitrary	IEBB0DR										
	IEBBTV (vector interrupt) (during reception in FIFO mode)	IEBB0DR	Arbitrary	010_1001	11								
CD-ROM decoder	IREADY (decoding end)	STRMDOU T0	Arbitrary	010_1010	11		010	1			0		
NAND flash memory controller	Data unit: Transmit data empty	Arbitrary	FLDTFIFO	010_1100	11		010	1			1		
	Data unit: Receive data full	FLDTFIFO	Arbitrary								0		
SD host interface 0	SDHI_0 transmission	Arbitrary	Data register	011_0000	01		010	1			1		
	SDHI_0 reception	Data register	Arbitrary		10	0							
SD host interface 1	SDHI_1 transmission	Arbitrary	Data register	011_0001	01						1		
	SDHI_1 reception	Data register	Arbitrary		10	0							
MMC host interface	Transmit data empty	Arbitrary	Data register	011_0010	01		010	1			1		
	Receive data full	Data register	Arbitrary		10	0							

Table 9.4 On-Chip Peripheral Module Requests

DMA Transfer Request Source	DMA Transfer Request Signal	Transfer Source	Transfer Destination	DMARS		CHCFG_n								
				MID	RID	TM	AM [2:0]	LVL	HIEN	LO EN	RE QD	SEL[2:0]		
Serial sound interface channel 0	SSITX10 (transmit data empty)	Arbitrary	SSIFTDR_0	011_1000	01	0	010	1	1	0	1	Ch0: 000 Ch1: 001 Ch2: 010 Ch3: 011 Ch4: 100 Ch5: 101 Ch6: 110 Ch7: 111 Ch8: 000 Ch9: 001 Ch10: 010 Ch11: 011 Ch12: 100 Ch13: 101 Ch14: 110 Ch15: 111		
	SSIRX10 (receive data full)	SSIFRDR_0	Arbitrary		10						0			
Serial sound interface channel 1	SSITX11 (transmit data empty)	Arbitrary	SSIFTDR_1	011_1001	01						1			
	SSIRX11 (receive data full)	SSIFRDR_1	Arbitrary		10						0			
Serial sound interface channel 2	SSIRT12 (transmit data empty)	Arbitrary	SSIFRDR_2	011_1010	11						1			
	SSIRT12 (receive data full)	SSIFRDR_2	Arbitrary								0			
Serial sound interface channel 3	SSITX13 (transmit data empty)	Arbitrary	SSIFTDR_3	011_1011	01						1			
	SSIRX13 (receive data full)	SSIFRDR_3	Arbitrary		10						0			
Serial sound interface channel 4	SSIRT14 (transmit data empty)	Arbitrary	SSIFTDR_4	011_1100	11						1			
	SSIRT14 (receive data full)	SSIFRDR_4	Arbitrary								0			
Serial sound interface channel 5	SSITX15 (transmit data empty)	Arbitrary	SSIFTDR_5	011_1101	01						1			
	SSIRX15 (receive data full)	SSIFRDR_5	Arbitrary		10						0			
SCUX	SCUTX10 (FFD0_0 request)	Arbitrary	DMATD0_CIM	100_0000	01						001		1	1
	SCURX10 (FFU0_0 request)	DMATU0_CIM	Arbitrary		10									0
	SCUTX11 (FFD0_1 request)	Arbitrary	DMATD1_CIM	100_0001	01									1
	SCURX11 (FFU0_1 request)	DMATU1_CIM	Arbitrary		10	0								
	SCUTX12 (FFD0_2 request)	Arbitrary	DMATD2_CIM	100_0010	01	1								
	SCURX12 (FFU0_2 request)	DMATU2_CIM	Arbitrary		10	0								
	SCUTX13 (FFD0_3 request)	Arbitrary	DMATD3_CIM	100_0011	01	1								
	SCURX13 (FFU0_3 request)	DMATU3_CIM	Arbitrary		10	0								



Table 9.4 On-Chip Peripheral Module Requests

DMA Transfer Request Source	DMA Transfer Request Signal	Transfer Source	Transfer Destination	DMARS		CHCFG_n											
				MID	RID	TM	AM [2:0]	LVL	HIEN	LO EN	RE QD	SEL[2:0]					
Renesas serial peripheral interface channel 0	SPTI0 (transmit data empty)	Arbitrary	SPDR_0	100_1000	01	0	010	1	1	0	1	Ch0: 000 Ch1: 001 Ch2: 010 Ch3: 011 Ch4: 100 Ch5: 101 Ch6: 110 Ch7: 111 Ch8: 000 Ch9: 001 Ch10: 010 Ch11: 011 Ch12: 100 Ch13: 101 Ch14: 110 Ch15: 111					
	SPRI0 (receive data full)	SPDR_0	Arbitrary		10						0						
Renesas serial peripheral interface channel 1	SPTI1 (transmit data empty)	Arbitrary	SPDR_1	100_1001	01						1						
	SPRI1 (receive data full)	SPDR_1	Arbitrary		10						0						
Renesas serial peripheral interface channel 2	SPTI2 (transmit data empty)	Arbitrary	SPDR_2	100_1010	01						1						
	SPRI2 (receive data full)	SPDR_2	Arbitrary		10						0						
Renesas serial peripheral interface channel 3	SPTI3 (transmit data empty)	Arbitrary	SPDR_3	100_1011	01						010		1				
	SPRI3 (receive data full)	SPDR_3	Arbitrary		10						0						
Renesas serial peripheral interface channel 4	SPTI4 (transmit data empty)	Arbitrary	SPDR_4	100_1100	01						010		1				
	SPRI4 (receive data full)	SPDR_4	Arbitrary		10						0						
Renesas SPDIF interface	SPDIFTXI	Arbitrary	TDAD	101_0000	01						010		1				
	SPDIFRXI	RDAD	Arbitrary		10						0						
Motor control PWM timer Channel 1	CMI1	Arbitrary	PWBFR1	101_0001	11						0		001	1	1	0	1
Motor control PWM timer Channel 2	CMI2	Arbitrary	PWBFR2	101_0010	11												1
Media local bus	MLB_CINT (MLB channel write)	Arbitrary	Local Channel buffer	101_0011	11												010
	MLB_CINT (MLB channel read)	Local Channel buffer	Arbitrary		0												

Table 9.4 On-Chip Peripheral Module Requests

DMA Transfer Request Source	DMA Transfer Request Signal	Transfer Source	Transfer Destination	DMARS		CHCFG_n						
				MID	RID	TM	AM [2:0]	LVL	HIEN	LO EN	RE QD	SEL[2:0]
Sound generator 0	SGDEI0	Arbitrary	SGLR_0	101_0100	11	0	001	1	1	0	1	Ch0: 000 Ch1: 001 Ch2: 010 Ch3: 011 Ch4: 100 Ch5: 101 Ch6: 110 Ch7: 111 Ch8: 000 Ch9: 001 Ch10: 010 Ch11: 011 Ch12: 100 Ch13: 101 Ch14: 110 Ch15: 111
Sound generator 1	SGDEI1	Arbitrary	SGLR_1	101_0101	11						1	
Sound generator 2	SGDEI2	Arbitrary	SGLR_2	101_0110	11						1	
Sound generator 3	SGDEI3	Arbitrary	SGLR_3	101_0111	11		1					
Serial communication interface channel 0	TXI0	Arbitrary	TDR0	101_1010	01		010	0			1	
	RXI0	RDR0	Arbitrary		10						0	
Serial communication interface channel 1	TXI1	Arbitrary	TDR1	101_1011	01		010	0			1	
	RXI1	RDR1	Arbitrary		10						0	
I <sup>2</sup> C bus interface channel 0	INTRIIC_TI0 (transmit data empty)	Arbitrary	RIIC0DRT	110_0000	01		010	0			1	
	INTRIIC_RI0 (receive data full)	RIIC0DRR	Arbitrary		10						0	
I <sup>2</sup> C bus interface channel 1	INTRIIC_TI1 (transmit data empty)	Arbitrary	RIIC1DRT	110_0001	01		010	0			1	
	INTRIIC_RI1 (receive data full)	RIIC1DRR	Arbitrary		10						0	
I <sup>2</sup> C bus interface channel 2	INTRIIC_TI2 (transmit data empty)	Arbitrary	RIIC2DRT	110_0010	01		010	0			1	
	INTRIIC_RI2 (receive data full)	RIIC2DRR	Arbitrary		10						0	
I <sup>2</sup> C bus interface channel 3	INTRIIC_TI3 (transmit data empty)	Arbitrary	RIIC3DRT	110_0011	01		010	0			1	
	INTRIIC_RI3 (receive data full)	RIIC3DRR	Arbitrary		10	0						
LIN interface channel 0	LIN0_INT_T	Arbitrary	RLN30LDBRm (m = 1 to 8)	110_1000	01	0	010	0	1	0	1	
	LIN0_INT_R	RLN30LDBRm (m = 1 to 8)	Arbitrary		10							
LIN interface channel 1	LIN1_INT_T	Arbitrary	RLN31LDBRm (m = 1 to 8)	110_1001	01		010	1			1	
	LIN1_INT_R	RLN31LDBRm (m = 1 to 8)	Arbitrary		10							
Pixel format converter channel 0	IFEI0	Arbitrary	PFVID	110_1100	01		010	1			1	
	OFFI0	PFVOD	Arbitrary		10							0
Pixel format converter channel 1	IFEI1	Arbitrary	PFVID	110_1101	01		010	1			1	
	OFFI1	PFVOD	Arbitrary		10							0
External request	DREQ0	Arbitrary	Arbitrary	000_0000	11		0/1	001/ 010/ 100	001: Falling edge detection 010: Rising edge detection 101: Low level detection 110: High level detection	0/1	000	

Note: • CHCFG\_n setting value

- TM 0: Single transfer  
1: Block transfer
- AM 001: ACK level output  
010: ACK bus cycle output  
100: No ACK
- LVL 0: REQ edge detection  
1: REQ level detection
- REQD 0: ACK output at read  
1: ACK output at write

## 9.6 DMA Mode

### 9.6.1 Mode Setting

The DMS field of the CHCFG\_n register can be used to toggle between register mode and link mode.

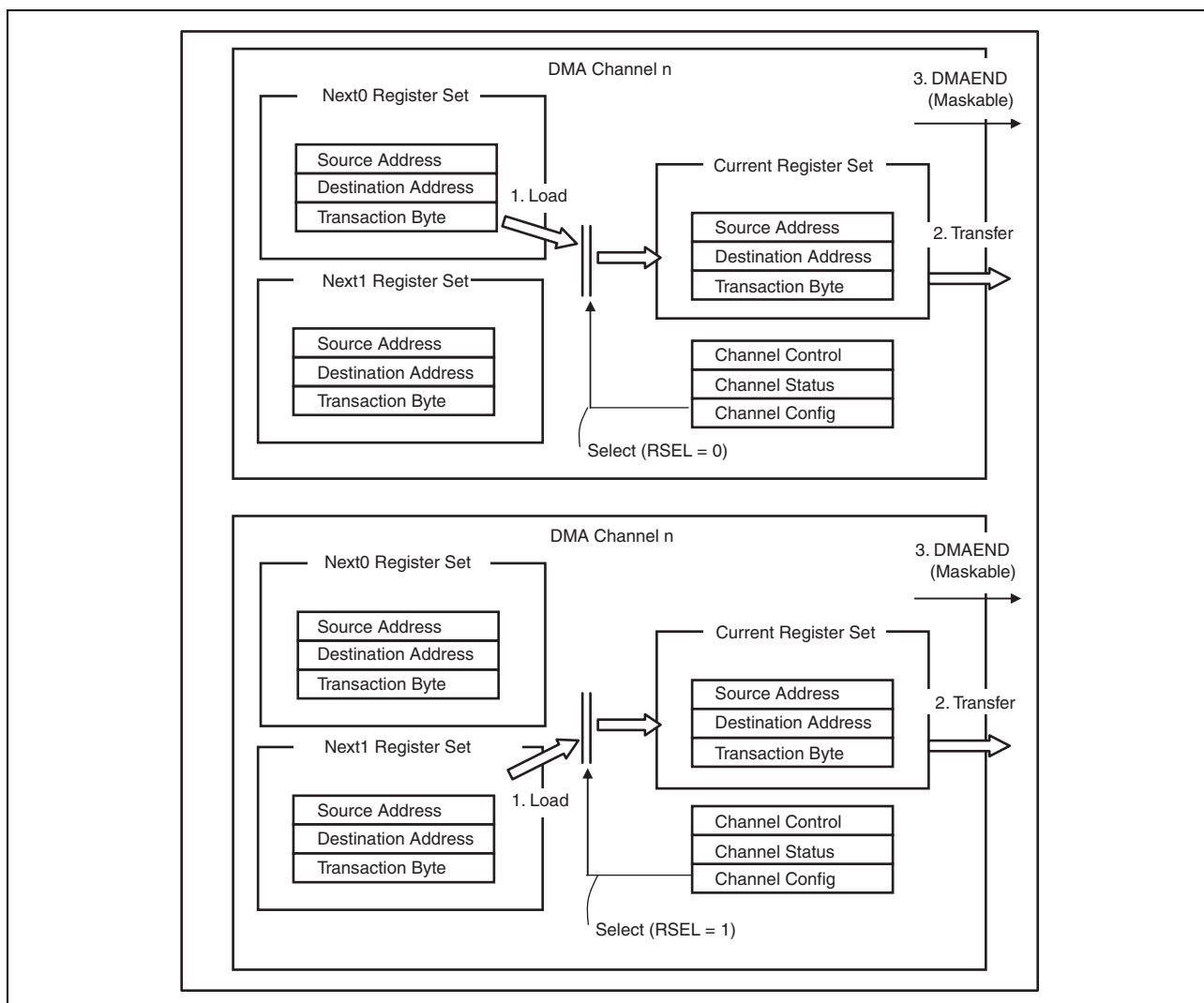
**Table 9.5 DMA Mode Setting**

DMS (CHCFG_n)	Mode	Description
0	Register mode	A DMA transfer is executed using the values set in the Next Register Set.
1	Link mode	A DMA transfer is executed using the descriptor set in the Current register. The DMAC repeatedly loads the descriptor and executes the DMA transfer unless otherwise set by the descriptor or stopped by the control register.

### 9.6.2 Register Mode

In register mode, a DMA transfer is executed using the values set in the internal registers.

Two sets of the source address, destination address, and transfer byte count (Next0 Register Set and Next1 Register Set) can be set. It is possible to select the Next register set to be used for the DMA transfer, as well as to execute two Next register sets continuously for the DMA transfer.



**Figure 9.2 Outline of Normal Register Mode**

The above figure shows how the transfer is executed when the Next0 Register Set is used (upper part of the figure) and when the Next1 Register Set is used (lower part of the figure).

(1) Operation Flow

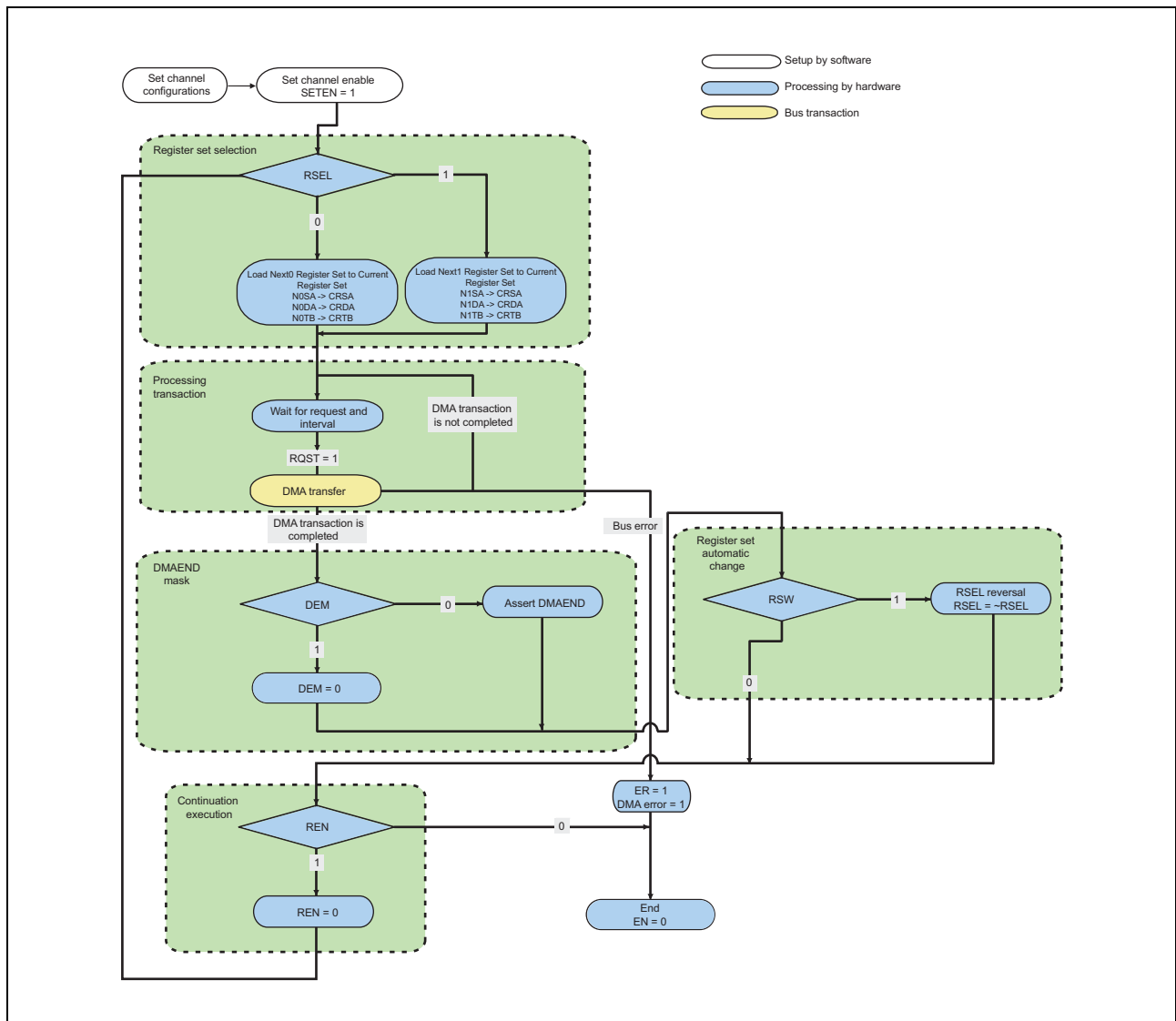


Figure 9.3 Register Mode Flow

<Explanation of the register mode flow>

1. Channel setting (set channel configuration)

The Next0 or Next1 register set (destination address, source address, and total transfer byte count) is set.  
In the Channel register set, the DMA register set (REQ, DMAACK, transfer size, etc.) is set. (See section 9.7, DMA Transfer.)

2. Register set selection (register set selection)

When 1 is set in EN, the values set in the Next register set selected by RSEL are loaded to the Current register set.

3. DMA transaction (processing transaction)

A DMA transfer is executed according to the set values. For details of the transfer, see section 9.7, DMA Transfer.

4. DMA transfer end interrupt mask (DMAINT mask)

The DMA transfer end interrupt is masked according to the value set in the DEM bit of CHCFG\_n. When 1 is set in DEM, the DMA transfer end interrupt is not output. Also, immediately after that, DEM is automatically cleared to 0.

5. Automatic register set change (register set automatic change)

Whether to use the other Next register set is determined by the value set in the RSW bit of CHCFG\_n.

6. Continuation of execution (continuation execution)

Whether to continue the execution of the DMA transfer is determined by the value set in the REN bit of CHCFG\_n. When 1 is set in REN, the execution of the DMA transfer is continued. Also, immediately after that, REN is automatically cleared to 0.

## (2) Register Setting

## (a) Register mode setting

Select the register set to be executed.

**Table 9.6 Register Mode Setting**

DMS (CHCFG_n)	RSEL (CHCFG_n)	Description
0	0	Executes the Next0 Register Set.
	1	Executes the Next1 Register Set.

## (b) DMA transfer end interrupt mask setting

The DMA transfer end interrupt can be masked individually for each register set.

**Table 9.7 DMAINT Mask Setting**

DEM (CHCFG_n)	Operation	Remark
0	When the DMA transaction is completed, a DMA transfer end interrupt is issued.	
1	Even when the DMA transaction is completed, a DMA transfer end interrupt is not issued. After the DMA transaction is completed, DEM is cleared to 0 by hardware.	

## (c) Automatic register set execution setting

After DMA transfers, the DMA transaction of the selected register set is automatically executed.

**Table 9.8 Automatic Register Set Execution Setting**

REN (CHCFG_n)	Operation	Remark
0	When the DMA transaction of the register set selected by RSEL is completed, the EN bit is cleared and the DMA operation ends.	Set this value to execute a DMA transaction once.
1	When a DMA transaction is completed, the DMAC continues to execute a DMA transfer by using the data set in the selected register set. When continuous transfers are successful, REN is cleared to 0.	Set this value to continuously execute DMA transfers by using the data set in separate register sets.

(d) Automatic register set change setting

When 1 is set in REN, the DMAC can automatically change to the register set to be executed next, after a DMA transaction is completed.

**Table 9.9 Automatic Register Set Change Setting**

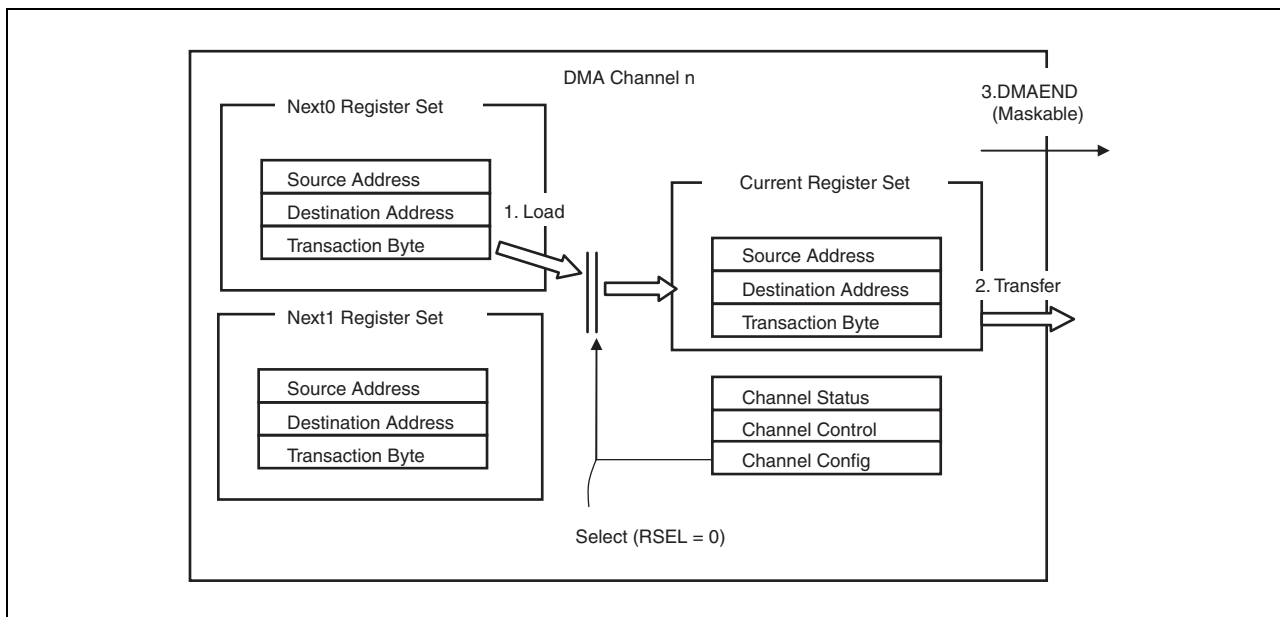
RSW (CHCFG_n)	Operation	Remark
0	If 1 is set in REN when a DMA transaction is completed, the register set is not changed.	Set this value to use only one register set.
1	If 1 is set in REN when a DMA transaction is completed, the value of RSEL is automatically inverted and the other register set is selected.	Set this value to change the register set.

(3) Setting Examples

(a) When only the Next0 register set is used

**Table 9.10 Register Mode Setting Example 1**

DMS (CHCFG_n)	RSEL (CHCFG_n)	DEM (CHCFG_n)	RSW (CHCFG_n)	REN (CHCFG_n)
0 (Register mode)	0 (Next0)	0 (not masked)	0 (not switched)	0 (not continuously executed)



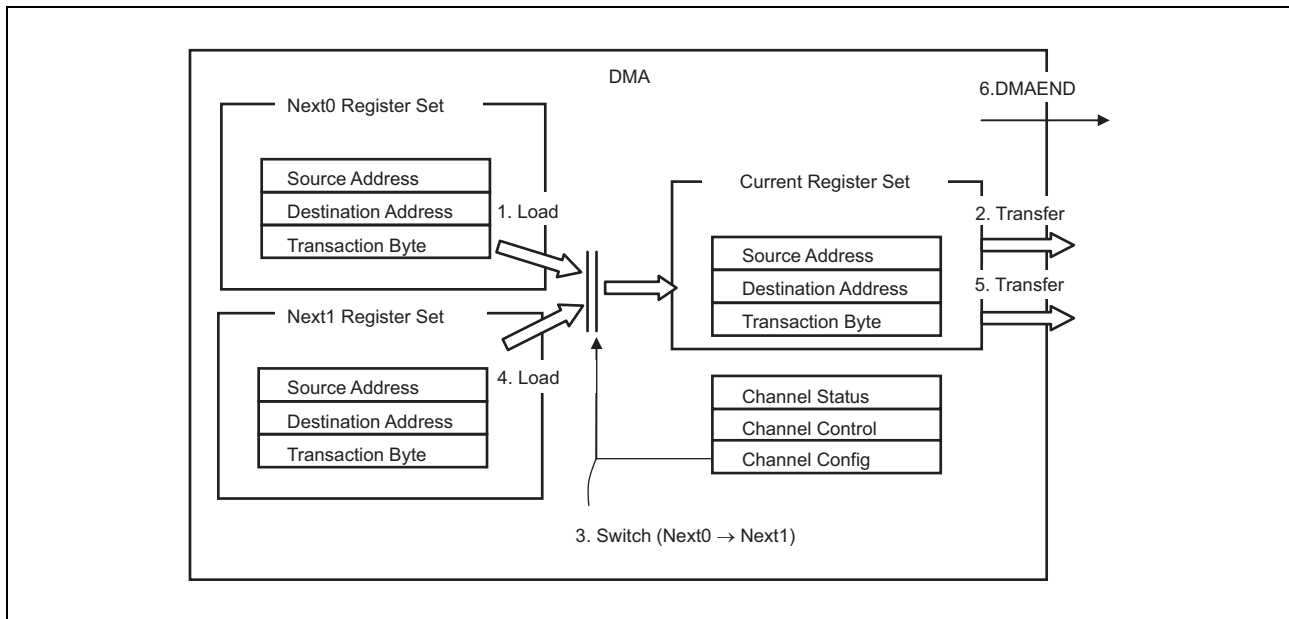
**Figure 9.4 Register Mode Setting Example 1**

- 1 is set in EN (SETEN = 1), and the Next0 register set is loaded to the Current register set.
- A DMA transaction is executed according to the values set in the Current register set and Channel register set.
- Because 0 is set in DEM, the DMA transfer end interrupt is issued after the DMA transaction is completed.
- Because 0 is set in REN, EN is cleared to 0 and the DMA transaction ends.

(b) When two register sets are used continuously

**Table 9.11 Automatic Register Set Execution Setting**

DMS (CHCFG_n)	RSEL (CHCFG_n)	DEM (CHCFG_n)	RSW (CHCFG_n)	REN (CHCFG_n)
0 (Register mode)	0 (Next0)	1 (masked)	1 (switched)	1 (continuously executed)



**Figure 9.5 Register Mode Setting Example 2**

1. 1 is set in EN (SETEN = 1), and the Next0 register set is loaded to the Current register set.
2. A DMA transaction is executed according to the values set in the Current register set and Channel register set.
3. Because 1 is set in DEM, DMA transfer end interrupt is not output after the DMA transaction is completed. Also, DEM is automatically cleared to 0.
4. Because 1 is set in REN, the execution is continued. Also, REN is automatically cleared to 0.
5. Because 1 is set in RSW, the register set to be executed next is switched (RSEL = 0 → 1).
6. The Next1 register set is loaded to the Current register set.
7. A DMA transaction is executed according to the values set in the Current register set and Channel register set.
8. Because 0 is set in DEM, the DMA transfer end interrupt is issued after the DMA transaction is completed.
9. Because 0 is set in REN, EN is cleared to 0 and the DMA transaction ends.



### 9.6.3 Link Mode

In link mode, a descriptor stored in external memory is loaded as set values and a DMA transaction is executed using the loaded values. The DMAC contains a Next Link address and a Current Link address for each channel, and these addresses are used to set the descriptor address to be executed next and to indicate the descriptor address of the currently executed DMA transaction, respectively.

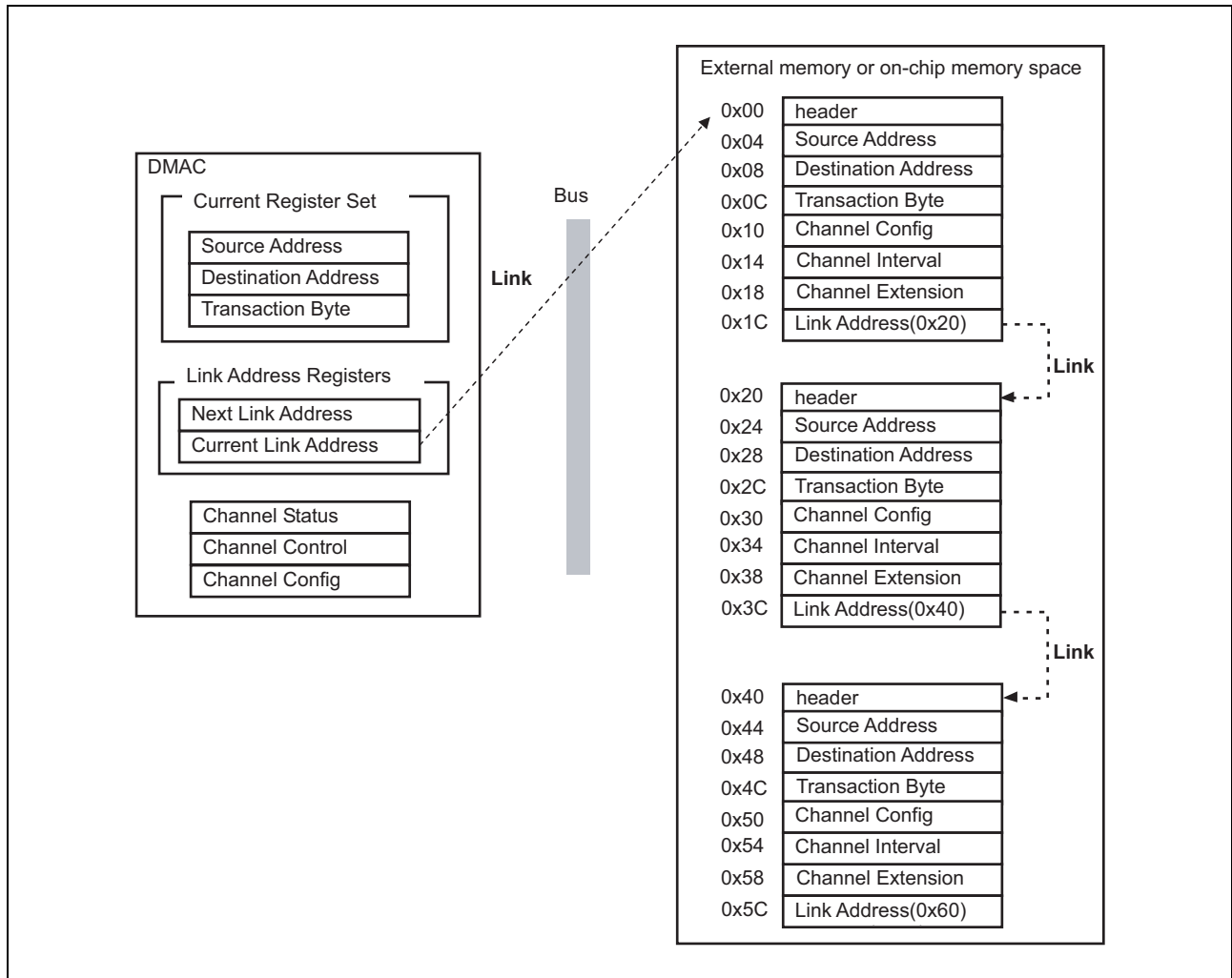


Figure 9.6 Link Mode Outline

(1) Operation Flow

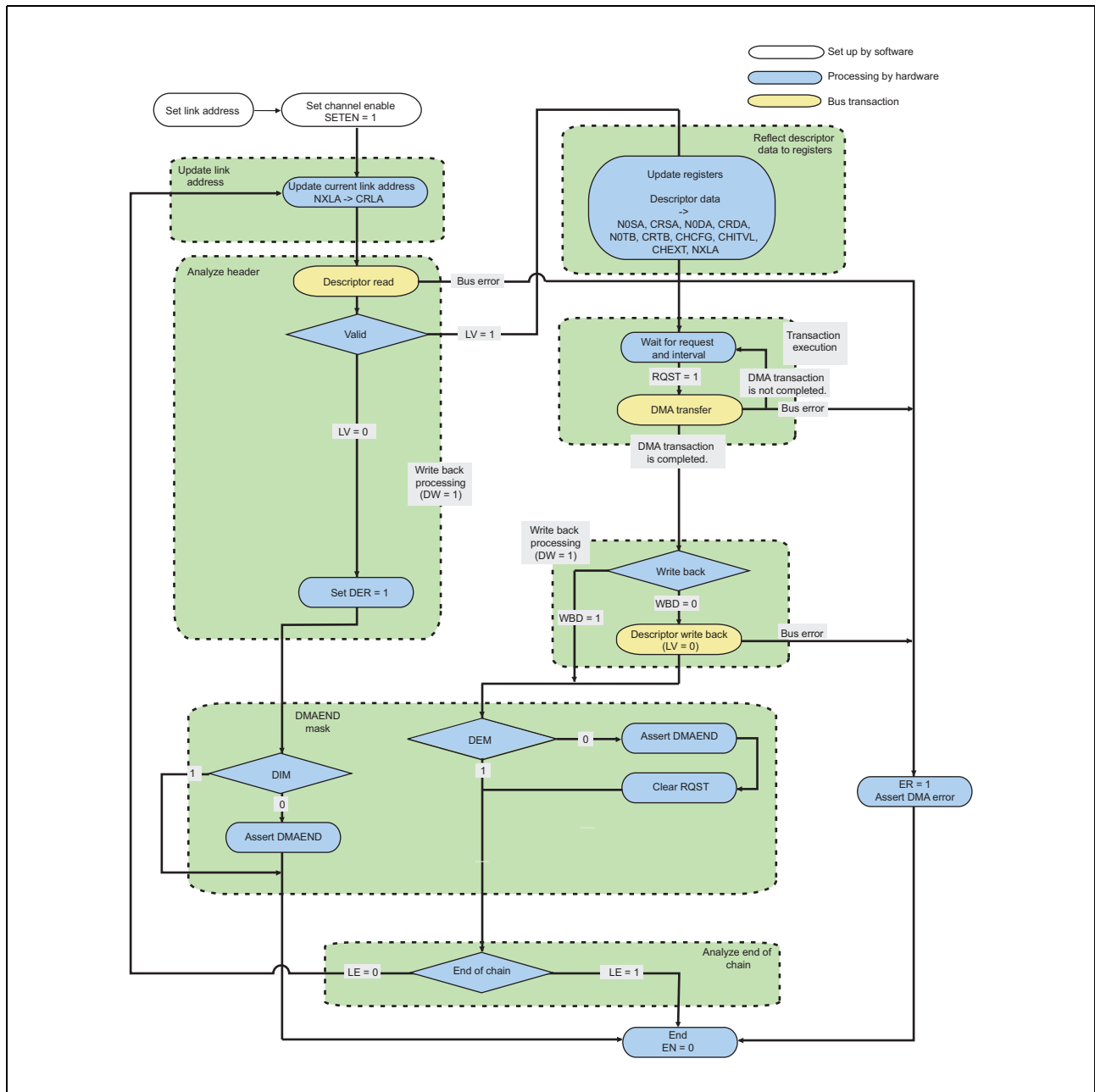


Figure 9.7 Link Mode Flow

<Explanation of the link mode flow>

1. Channel setting  
The start address of the link destination is set in NXLA\_n.
2. Link address update  
When 1 is set in EN (1 is set in SETEN), the Link address set in NXLA\_n is loaded to CRLA\_n.
3. Descriptor load and header analysis  
The DMAC begins to load the descriptor and then analyzes the content of the header. When LV is 0, the DMAC discards the loaded descriptor and sets 1 in DER to end the operation (EN = 0). In this case, if 0 is set in DIM of the header, DMAEND is issued.
4. Descriptor setting  
The loaded descriptor is set in the Current register set and Channel register set. Also, the next link address is set in NXLA\_n.
5. DMA transaction  
A DMA transaction is executed according to the set values.
6. Header writeback  
When 0 is set in WBD of the header, the DMAC writes back the header with 0 set in its LV bit.
7. DMAINT mask  
When 0 is set in the DEM bit of CHCFG\_n, the DMA transfer end interrupt is issued.
8. Link end analysis  
When 1 is set in LE of the header, the operation is ended by clearing EN to 0, after transfer using the settings of the descriptor is completed. If the setting of LE is 0, the Current registers are then updated and loading of the next descriptor begins. The TEND signal is issued after the transfer of each descriptor.

## (2) Register Setting

### (a) Link mode setting

To use the link mode, set 1 in the DMS bit of the CHCFG\_n register.

**Table 9.12 Link Mode Setting**

DMS (CHCFG_n)	Description
1	Operates in link mode. This bit cannot be changed using a descriptor.

### (b) Link address setting

There are two registers that indicate a link address: Next Link address register and Current Link address register.

To start the link mode, set a link address in the Next Link address register.

The Next Link address indicates the next link address after a descriptor is loaded. The Current Link address indicates the currently executed link address.

**Table 9.13 Link Address Register Set**

Register	Description
Next Link Address Register (NXLA_n)	Sets and indicates the next link address. Before starting the link mode, set a link address in this register.
Current Link Address Register (CRLA_n)	Indicates the currently executed link address. This register is read-only.

<Caution> In link mode, the settings can be changed by reading a descriptor. It is not possible, however, to synchronize the change of the settings with a peripheral module request or external request. Therefore, when using a peripheral module request or external request, set AM, LVL, HIEN, LOEN, and SEL of the CHCFG\_n register before setting Enable and do not change any of these bits in the descriptor.

## (3) Descriptor Setting

In a link address, prepare a descriptor with data arranged in the order shown below.

The DMAC reads the descriptor in burst mode.

### (a) Descriptor data arrangement

**Table 9.14 Descriptor Data Arrangement**

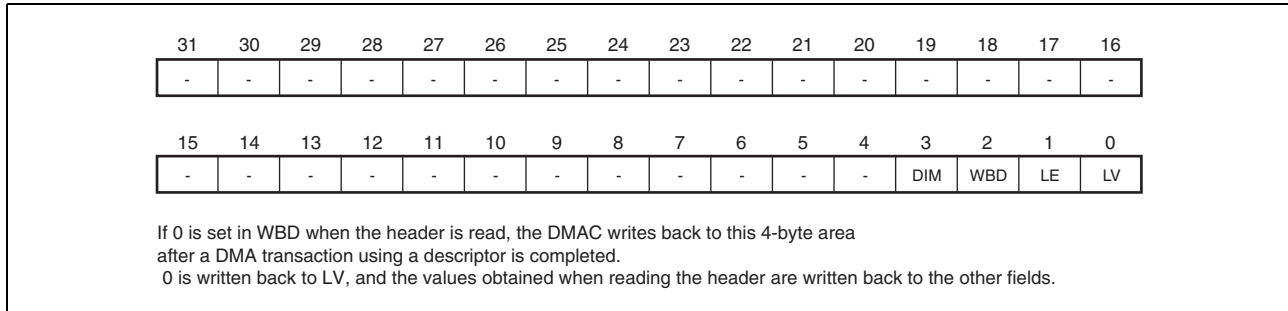
Address	Data	Remark
Link address + 00H	header	
Link address + 04H	Source Address	
Link address + 08H	Destination Address	
Link address + 0CH	Transaction Byte	
Link address + 10H	Config	The register mode cannot be set.
Link address + 14H	Interval	
Link address + 18H	Extension	
Link address + 1CH	Next Link Address	

Remark: As a link address, set an address aligned along the 32-bit boundary.

## (b) header

The header indicates the status of the descriptor, as shown below.

The DMAC reads this area when a DMA transfer is started in link mode. Also, after a DMA transaction is completed, the DMAC writes back the transfer status to the area.



**Figure 9.8** header Area

**Table 9.15** Header Area

Bit Position	Bit Name	Meaning
31 to 4	—	—
3	DIM	Descriptor Interrupt Mask Sets whether to mask the DMA transfer end interrupt if 0 is set in LV when the header is loaded. 0: Issues a DMA transfer end interrupt. 1: Does not issue a DMA transfer end interrupt.
2	WBD	Write Back Disable Sets whether to mask LV bit writeback. When 1 is set in this bit, the DMAC does not perform writeback. 0: Writes the LV bit back to 0. 1: Does not write back the LV bit.
1	LE	Link End Indicates whether the link ends with the DMA transaction of this descriptor. Set 1 in this bit to indicate the end of the link. 0: The link continues. 1: The link ends.
0	LV	Link Valid Indicates whether this descriptor is valid. If 0 is set in WBD, the DMAC writes 0 in this bit after the DMA transaction written in the descriptor is executed. When setting the header, set 1 in this bit. 0: Descriptor invalid 1: Descriptor valid

## (c) Descriptor data other than the header

The data items of the descriptor other than the header are the same as defined in the internal register specifications (note that the DMS bit of the CHCFG\_n register cannot be changed using the descriptor). For information about the internal register specifications, see section 9.4, Register Descriptions.

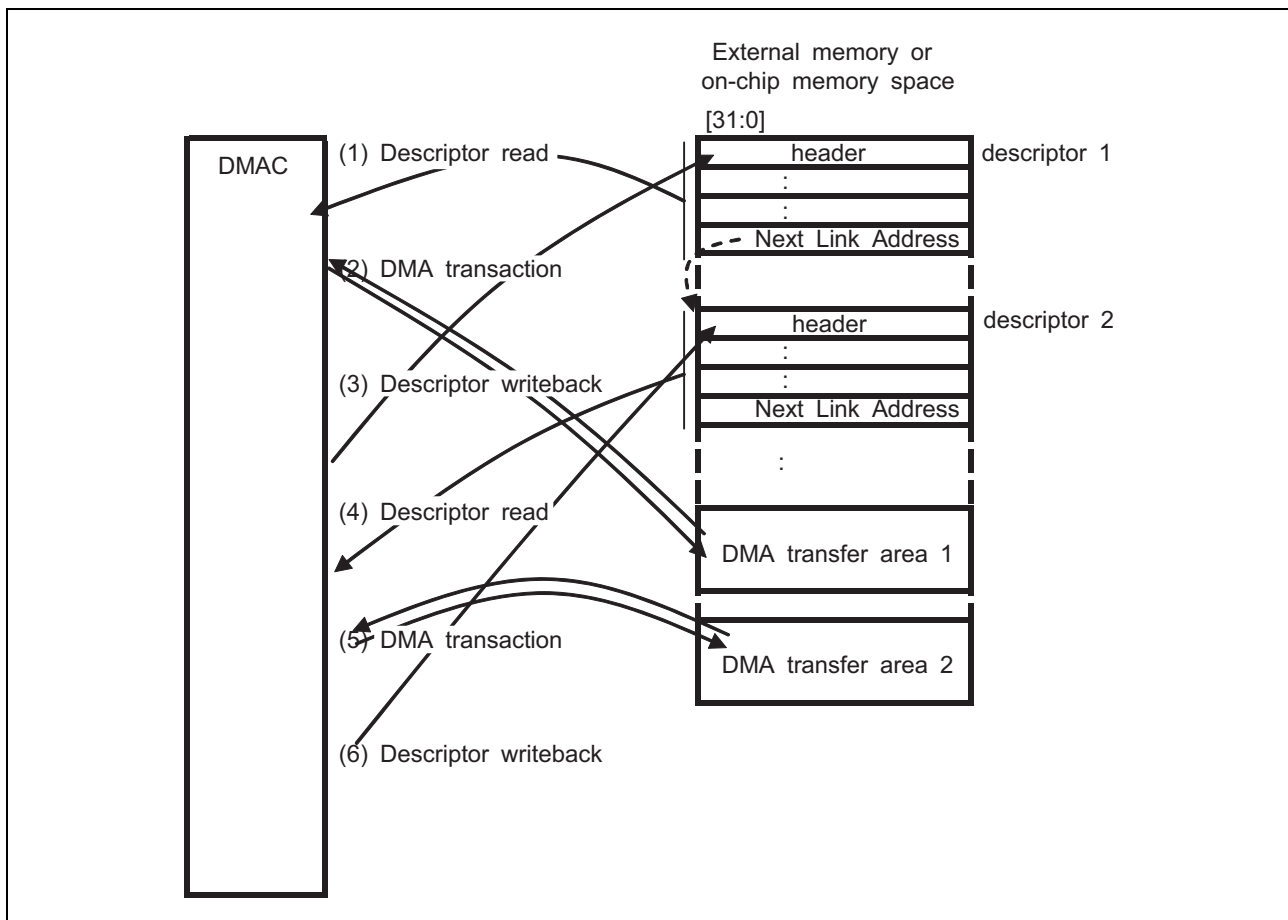
For descriptor setting examples, see section 9.8, DMA Setting Examples.

## (d) CACHE settings for descriptor access

The CACHE settings for descriptor access can be set in LWCA and LDCA of the DMA control register (DCTRL\_0\_7, DCTRL\_8\_15). Make these settings as appropriate for the access destination in which the descriptor is prepared.

## (e) Descriptor area and DMA transfer area

The following figure outlines the descriptor area and DMA transfer area that are accessed by the DMAC.



**Figure 9.9 Outline of the Descriptor Area and DMA Transfer Area**

1. Descriptor read  
The values set in the internal Next Link Address register are loaded to the Current Link Address register, and a descriptor is read from the external memory space (descriptor1) pointed to by the Current Link Address register.
2. DMA transfer  
When 1 is set in the LV bit of the header in the descriptor, a DMA transfer is executed according to the descriptor data.
3. Descriptor writeback  
When 0 is set in the WBD bit of the header after the DMA transfer of the set number of bytes is completed, the DMAC writes back data in word size to the header of descriptor1, with 0 set in LV and the other bits containing the values read in <1>.
4. Descriptor read  
When 0 is set in the LE bit of the header in the last read descriptor (<1>), the next descriptor is read from the address (descriptor2) indicated by Next Link Address in the descriptor.
5. DMA transfer  
When 1 is set in the LV bit of the header in the descriptor, a DMA transfer is executed according to the descriptor data.
6. Descriptor writeback  
When 0 is set in the WBD bit of the header after the DMA transfer of the set number of bytes is completed, the DMAC writes back data in word size to the header of descriptor2, with 0 set in LV and the other bits containing the values read in <4>.

4 through 6 are repeated.

When the header contains 1 in LE and 0 in WBD, the DMAC executes a DMA transfer using the settings of that descriptor, writes back data with 0 set in the LV bit of the header and ends the operation.

When the header contains 1 in both LE and WBD, the DMAC executes a DMA transfer using the settings of that descriptor and ends the operation (without writing back).

When the header contains 0 in LV, the DMAC ends the operation (without executing a DMA transfer).

#### (4) Descriptor Configuration Examples

In link mode, a descriptor can be configured as shown below.

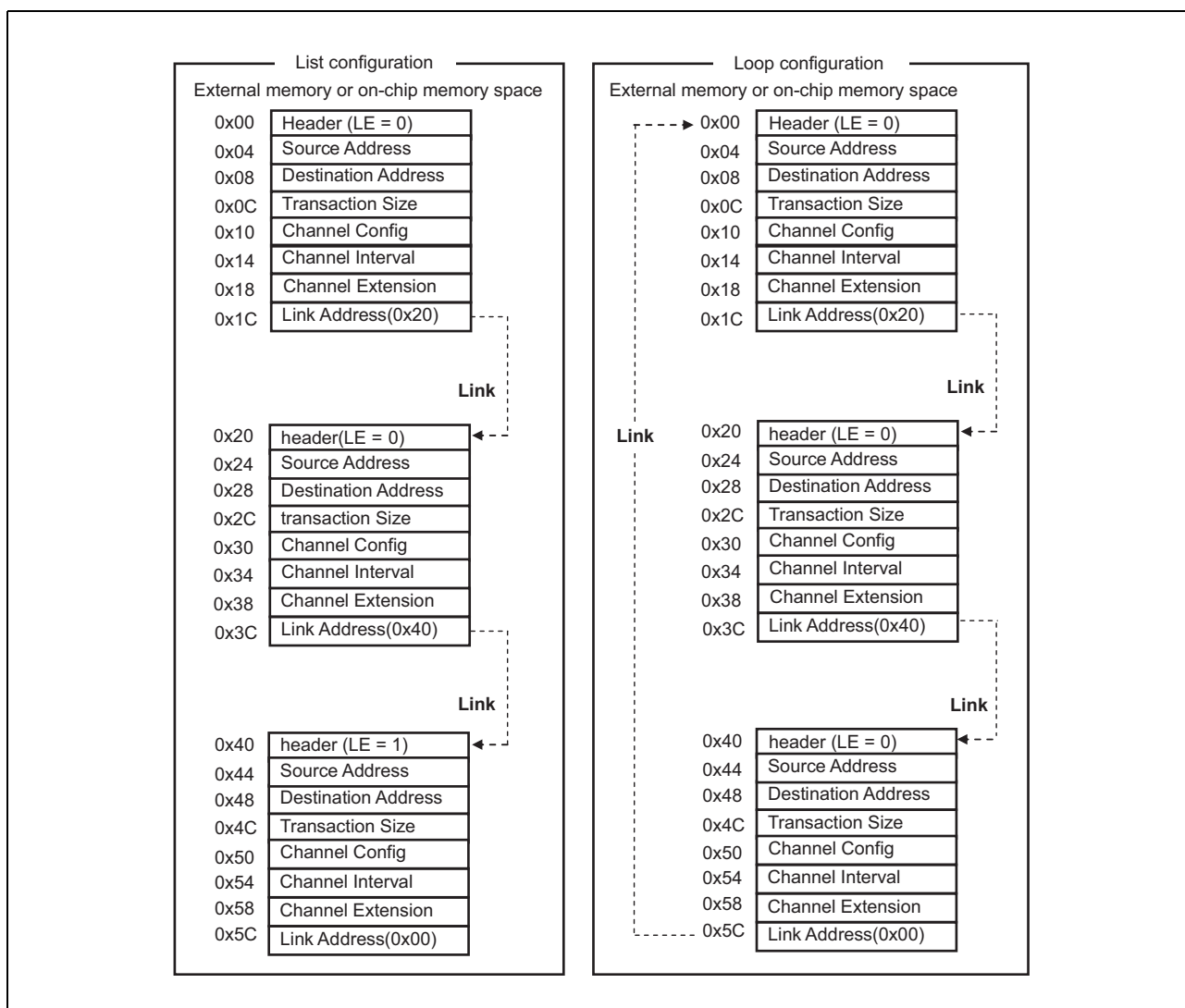


Figure 9.10 Descriptor Configuration Examples

- List configuration

The link is ended by setting 1 in the LE bit of the header in the last descriptor.

- Loop configuration

A descriptor can be created with a loop configuration, by setting the address of the top descriptor in the link address of the last descriptor. To end the loop, change the value of the LE bit of the header to 1 before the DMAC reads the descriptor, or follow the transfer suspension procedure.

## 9.7 DMA Transfer

The basic operation of DMA transfer is described here.

### 9.7.1 Transfer Mode

Two transfer modes are supported: single transfer mode and block transfer mode.

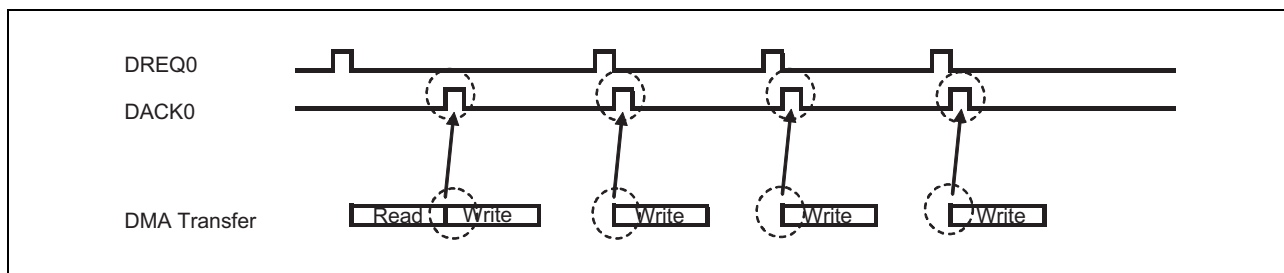
To select a transfer mode, set the TM bit of CHCFG\_n for each channel.

**Table 9.16 Basic Transfer Setting**

Transfer Mode	TM (CHCFG_n)	Function
Single transfer	0	A single DMA transfer is executed in response to a DMAREQ.
Block transfer	1	In response to a DMAREQ, the DMAC continues to execute the transfer until the DMA transaction is completed.

#### (1) Single Transfer Mode

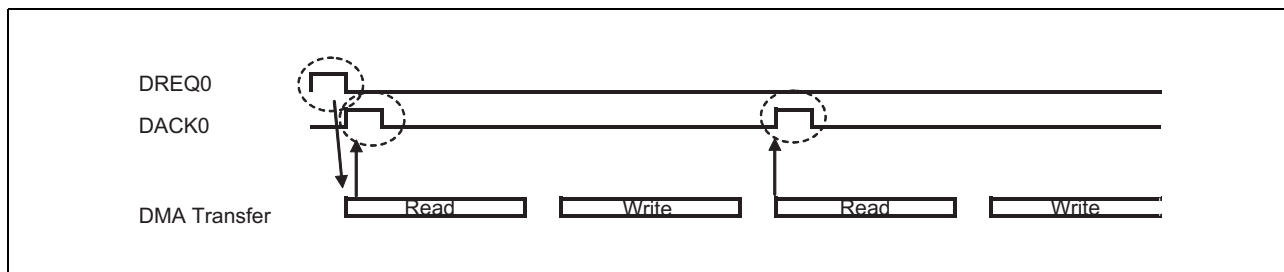
When a DMA transfer request is received, a DMA transfer is executed once in the direction indicated by REQD (source or destination). A DMA transfer is executed once each time a transfer request is received, and this operation continues until the number of bytes loaded to CRTB\_n is reached (arbitration between channels is accomplished for each DMA transfer).



**Figure 9.11 Single Transfer Mode (REQD = 1, SDS > DDS)**

#### (2) Block Transfer Mode

Once a DMA transfer request is received, the DMAC continues to execute the transfer until data equivalent to the number of bytes loaded to the DMA transfer byte register (CRTB\_n register) is transferred (the DMA transaction is completed) (arbitration between channels is accomplished for each DMA transfer).



**Figure 9.12 Block Transfer Mode (REQD = 0, SDS < DDS)**



## 9.7.2 Priority Control for DMA Channels

Within channels 0 to 7 and 8 to 15, two priority control modes are supported: fixed priority mode and round robin mode. Only round robin mode is supported for priority control between the group of channels 0 to 7 and the group of channels 8 to 15. To select a priority control mode, use the PR bit of the DMA control register (DCTRL register). The fixed priority mode is selected when 0 is set in the PR bit, and the round robin mode is selected when 1 is set.

Read priority and write priority are controlled independently.

The DMAC issues transfer requests to different channels concurrently without waiting for the completion of any particular transfer and processes responses in the order it receives them. Therefore, the order in which the channels start transactions is not necessarily consistent with the order in which the transactions end.

**Table 9.17 Priority Control Setting**

Mode	PR (DCTRL)	Function	Purpose
Fixed priority	0	Requests are controlled based on the fixed order of priority for channels 0 to 7 and 8 to 15 (High: CH0 (CH8) > CH1 (CH9) > CH2 (CH10) > CH3 (CH11) > CH4 (CH12) > CH5 (CH13) > CH6 (CH14) > CH7 (CH15): Low).	Use this mode when the channels have a specific order of priority.
Round robin	1	Requests are controlled in a round robin fashion.	Use this mode to execute all requests evenly.

### (1) Fixed Priority Mode

In fixed priority mode, the channels have a fixed order of priority in channels 0 to 7 and 8 to 15. Round robin mode is used to determine the priority between the group of channels 0 to 7 and the group of channels 8 to 15.

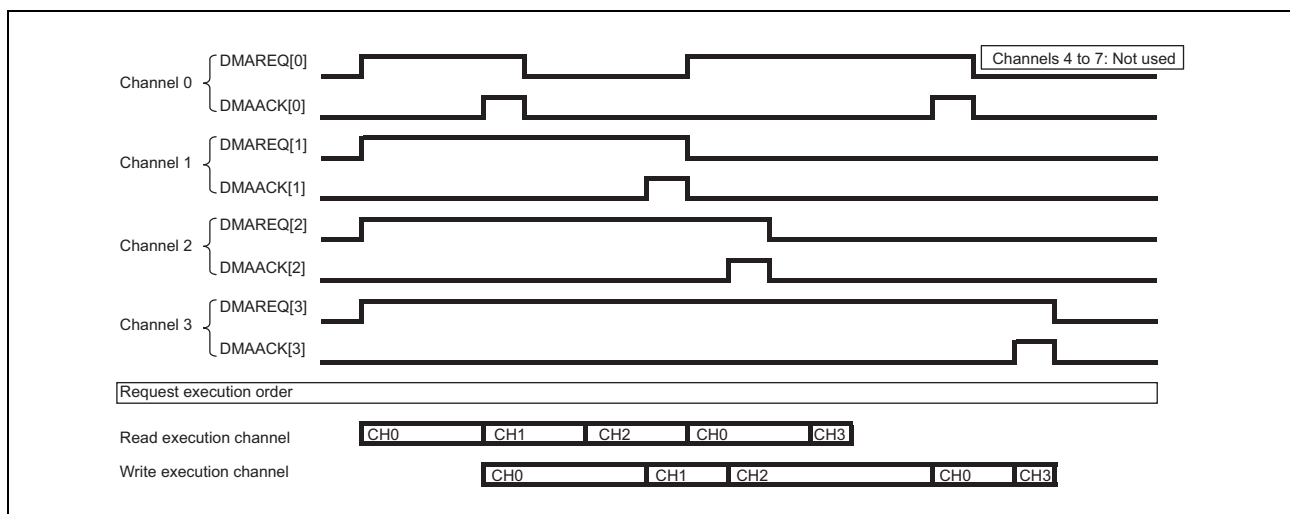
Immediately after a reset, the order of priority is as follows.

High CH0 > CH8 > CH1 > CH9 > CH2 > CH10 > CH3 > CH11 > CH4 > CH12 > CH5 > CH13 > CH6 > CH14 > CH7 > CH15 Low

If there is a transfer request from DMA channel 0 in this state, a transfer is executed on DMA channel 0. After the transfer is completed, the order of priority is as follows.

High CH8 > CH0 > CH9 > CH1 > CH10 > CH2 > CH11 > CH3 > CH12 > CH4 > CH13 > CH5 > CH14 > CH6 > CH15 > CH7 Low

If a DMA transfer request occurs on multiple channels simultaneously, the DMA transfer request of the channel having the smallest channel number is given priority. The following figure shows an example where a DMA transfer request occurs on a channel having a higher priority while a DMA transfer is being executed in fixed priority mode.



**Figure 9.13 Fixed Priority Mode (Number of Channels = 4, REQD = 1)**

(2) Round Robin Mode

In round robin mode, each time a transfer request is received from a channel in the group of channels 0 to 7 and the group of channels 8 to 15, the order of priority is changed in such a way that the channel that executed a transfer last has the lowest priority.

Round robin mode is used to determine the priority between the group of channels 0 to 7 and the group of channels 8 to 15.

Immediately after a reset, the order of priority is the same as that of the fixed priority mode, which is as follows.

High CH0 > CH8 > CH1 > CH9 > CH2 > CH10 > CH3 > CH11 > CH4 > CH12 > CH5 > CH13 > CH6 > CH14 > CH7 > CH15 Low

If a transfer request is received from DMA channel 2 in this state, a transfer is executed on DMA channel 2. After the transfer is completed, the order of priority is as follows.

High CH8 > CH3 > CH9 > CH4 > CH10 > CH5 > CH11 > CH6 > CH12 > CH7 > CH13 > CH0 > CH14 > CH1 > CH15 > CH2 Low

The following figure shows an example where DMA transfers are executed in round robin mode.

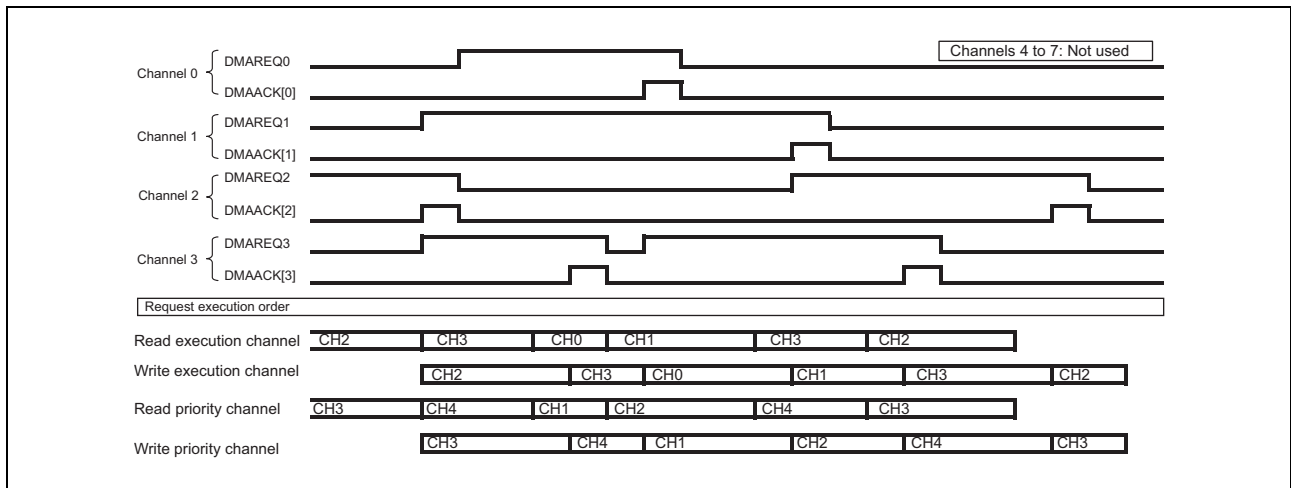


Figure 9.14 Round Robin Mode (Number of Channels = 4, REQD = 0)

The channel whose channel number is the number of the currently transferring channel + 1 gets to execute a DMA transfer next. If there is no transfer request from this channel, the channel whose channel number is the number of this channel + 1 gets to execute a DMA transfer.

### 9.7.3 Number of States of an External Bus Cycle

When this module is the bus master, the number of states of an external bus cycle is controlled by the bus state controller as when the CPU is the bus master. For details, refer to section 8, Bus State Controller.

### 9.7.4 DMA Transfer Request

Edge detection or level detection can be selected using the LVL bit of the CHCFG\_n register.

The HIEN and LOEN bits of the CHCFG\_n register are used to select either the rising edge or falling edge in the case of edge detection or either the high level or low level in the case of level detection.

When the transfer request is by an on-chip peripheral module, set the CHCFG\_n register according to Table 9.4, On-Chip Peripheral Module Requests.

When the transfer request is by the external pin (DREQ0), set the detection conditions (rising/falling edge and high/low level) according to Table 9.18, Setting for Detection of External Pin Request.

**Table 9.18 Setting for Detection of External Pin Request**

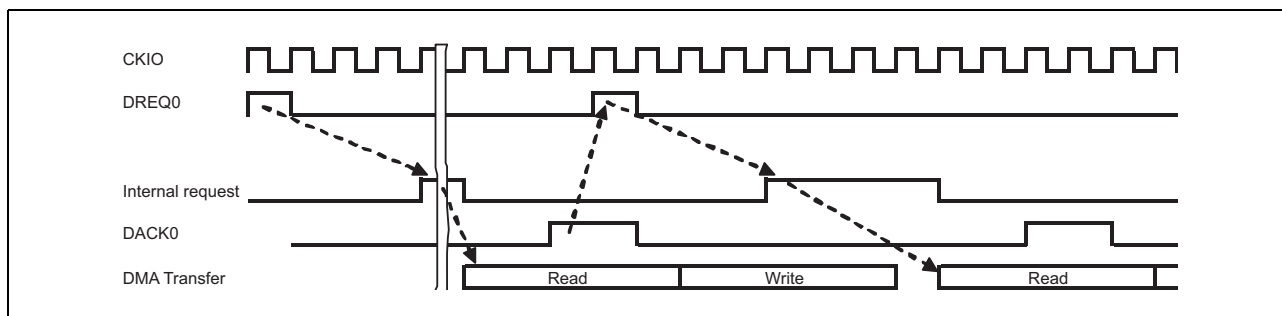
Mode	LVL (CHCFG_n)	HIEN (CHCFG_n)	LOEN (CHCFG_n)	Function
Edge detection	0	0	0	Specify this value when auto request triggers are in use.
			1	Detects external pin request (DREQ0) at its falling edge.
		1	0	Detects external pin request (DREQ0) at its rising edge.
			1	Setting prohibited
Level detection	1	0	0	Setting prohibited
			1	Detects external pin request (DREQ0) in Low level mode.
		1	0	Detects external pin request (DREQ0) in High level mode.
			1	Setting prohibited

#### (1) Edge Detection

Setting 0 in the LVL bit of the CHCFG\_n register enables edge detection.

When 1 is set in the HIEN bit of the CHCFG\_n register, rising edge detection is enabled. When 1 is set in the LOEN bit, falling edge detection is enabled.

Wait for DACK0 to be detected, before issuing the next DREQ0 request.



**Figure 9.15 Edge Detection Timing (HIEN = 1, REQD = 0)**

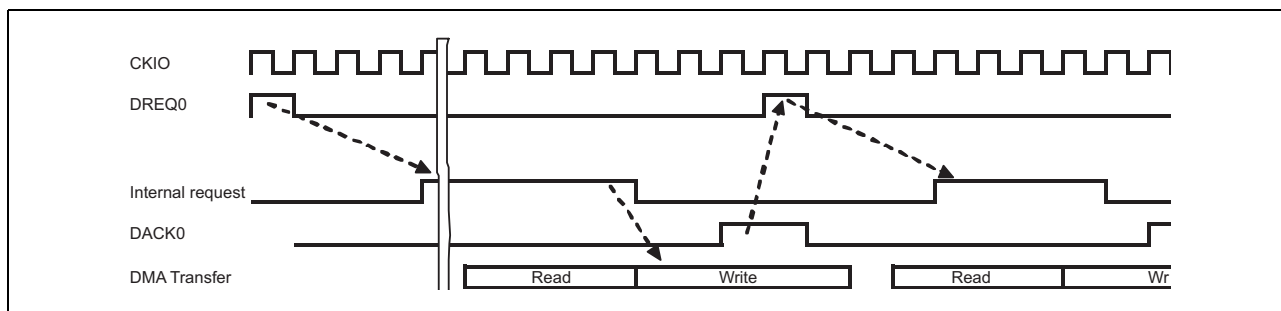


Figure 9.16 Edge Detection Timing (HIEN = 1, REQD = 1)

(2) Level Detection

Setting 1 in the LVL bit of the CHCFG\_n register enables level detection.

DREQ0 is regarded as valid when it remains active for two consecutive clock cycles or more (depending on the HIEN and LOEN settings).

When the level mode is selected for DACK0, it remains at the High level until DREQ0 is deasserted.

When the next DMA transfer request is to be issued, DACK0 needs to be deasserted before that DREQ0 can be asserted.

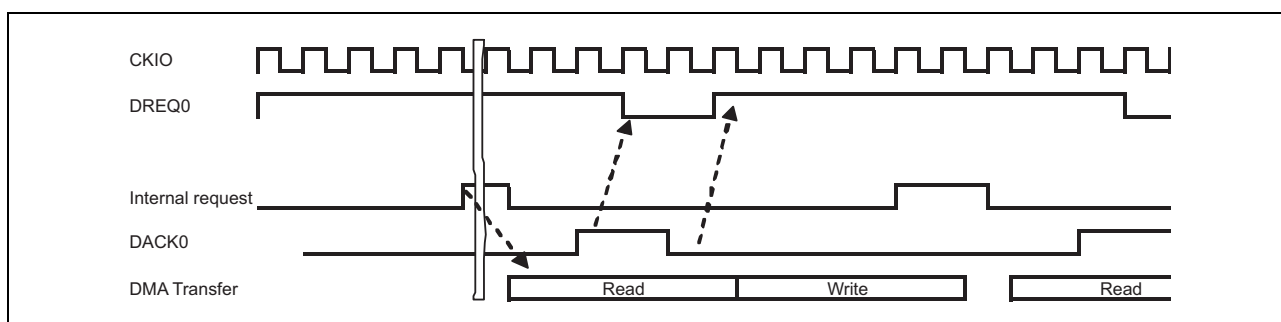


Figure 9.17 Level Detection Timing (HIEN = 1, REQD = 0, AM[2:0] = 001)

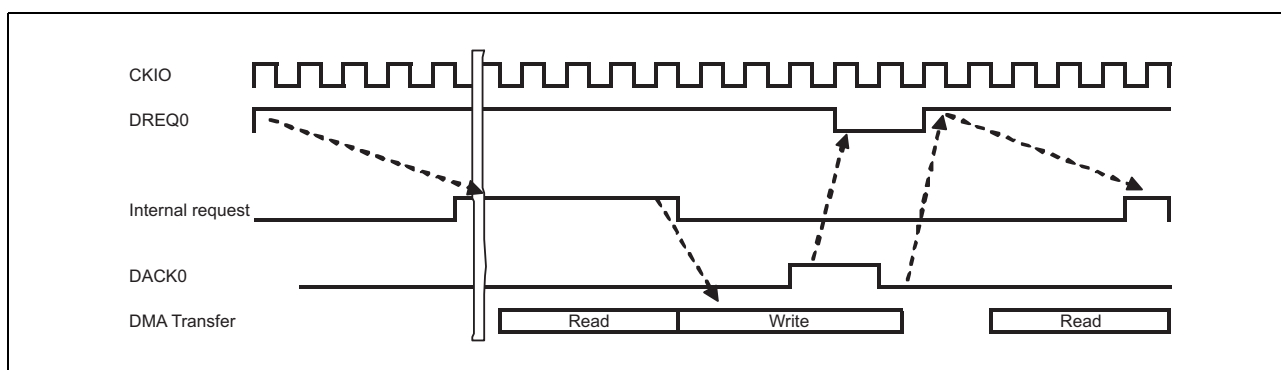


Figure 9.18 Level Detection Timing (HIEN = 1, REQD = 1, AM[2:0] = 001)

### 9.7.5 DMA Acknowledge Output Function

DACK0 is an acknowledge signal that is sent to DREQ0. Level output and bus cycle output settings are supported as the DACK0 output mode. DACK0 is asserted at the same time as CS assertion except for the MPX-IO interface. For details, refer to section 8, Bus State Controller.

#### (1) DMA Acknowledge Signal Output Timing Setting

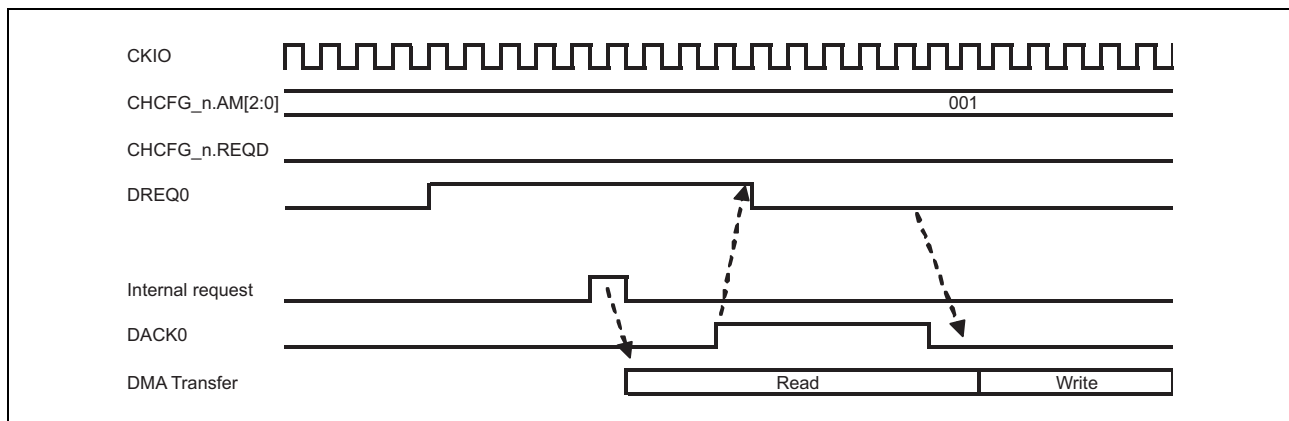
Upon receiving a DMA transfer request, the DACK0 pin becomes active (High level output). By using the REQD and AM[2:0] bits of the CHCFG\_n register, the DACK0 output timing can be set as shown below.

**Table 9.19 DACK0 Output Timing Setting**

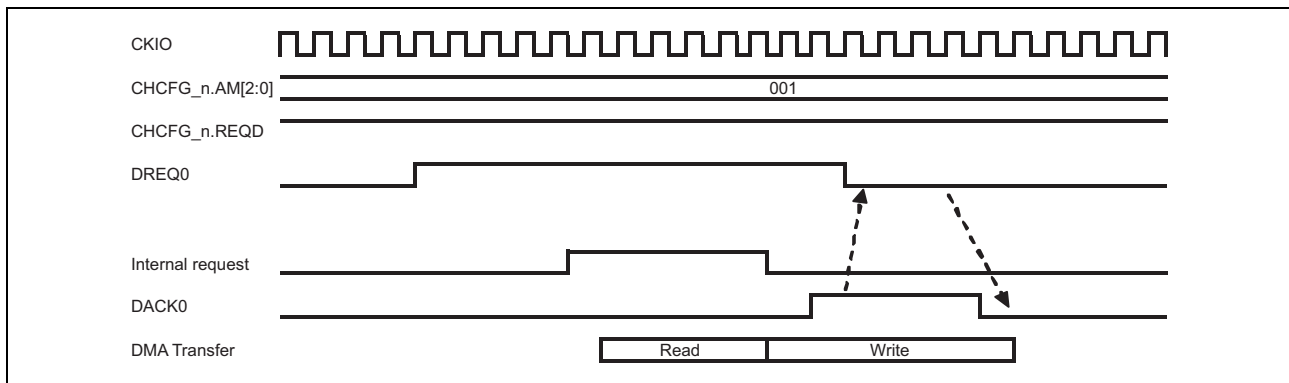
Mode	AM[2] (CHCFG_n)	AM[1:0] (CHCFG_n)	REQD (CHCFG_n)	Purpose
Pulse	0	00	0 1	Setting prohibited
Level	0	01	0 (Active during read) 1 (Active during write)	DACK0 is output as a level. DACK0 remains asserted until DREQ0 is deasserted.
Bus cycle	0	10 11	0 (Active during read) 1 (Active during write)	DACK0 is output for the duration of a bus cycle. Use this mode to keep DACK0 asserted until the end of the bus cycle.
Mask	1	—	—	Make this setting when using auto request trigger.

#### (2) Level Output

Setting 001 in the AM bits of the CHCFG\_n register enables level output. DACK0 remains asserted until DREQ0 is deasserted.



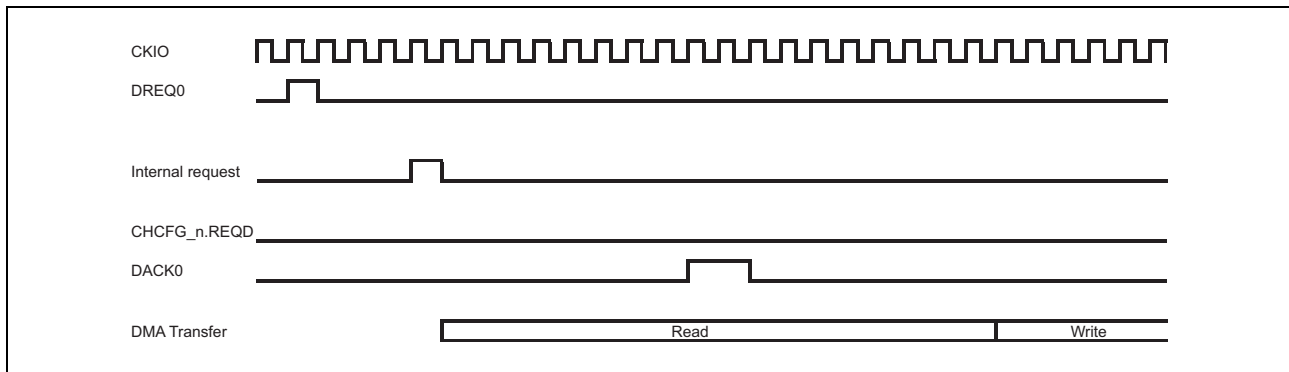
**Figure 9.19 DACK0 Output Timing (AM[2:0] = 001, REQD = 0)**



**Figure 9.20 DACK0 Output Timing (AM[2:0] = 001, REQD = 1)**

### (3) Bus Cycle Output

Setting 010 in the AM bits of the CHCTRL\_n register enables bus cycle output. DACK0 remains active for the duration of a bus cycle.



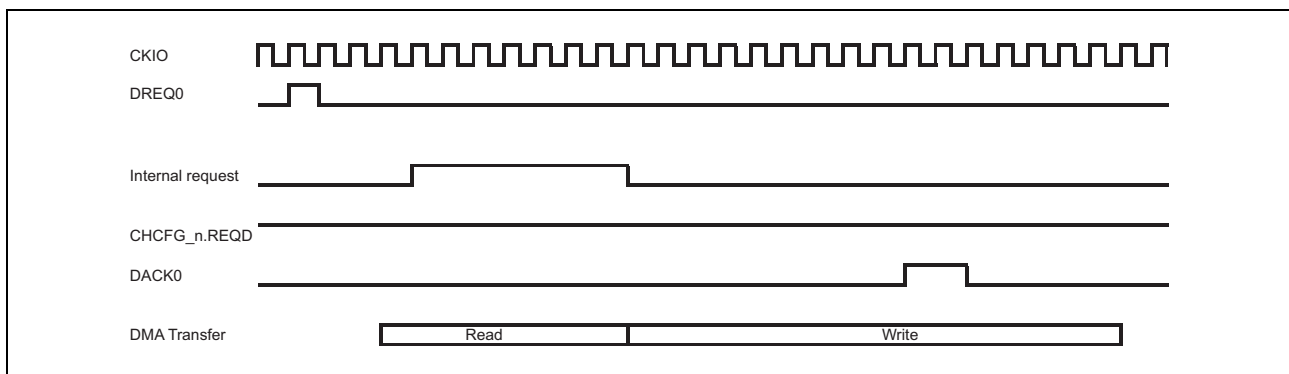
**Figure 9.21 Bus Cycle Output Timing (REQD = 0)**

- In the read active mode (REQD = 0), DACK0 remains active from the time when a read request is output on the bus until one cycle after the final read data.
- When level detection is selected for DREQ0, DREQ0 remains disabled until the cycle following the end of the bus cycle.

The following signals trigger the rise and fall of DACK0:

Rise: Transfer start (MARVALID = 1)

Fall: Transfer end (MRLAST & MRREADY = 1)



**Figure 9.22 Bus Cycle Output Timing (REQD = 1)**

- In the write active mode (REQD = 1), DACK0 remains active from the time when a write request is output until one cycle after the response to the final data is returned.
- When level detection is selected for DREQ0, DREQ0 remains disabled until the cycle following the end of the bus cycle.

The following signals trigger the rise and fall of DACK0:

Rise: Transfer start (MAWVALID = 1)

Fall: Transfer end (MBVALID & MBREADY = 1)

### 9.7.6 DMA Transfer End Output Function

TEND0 is a transaction completion signal that is sent to the source of a DMA transfer request. TEND0 is asserted as the same time as DACK0 for the last transfer transaction. Figure 9.23 shows the TEND0 output timing.

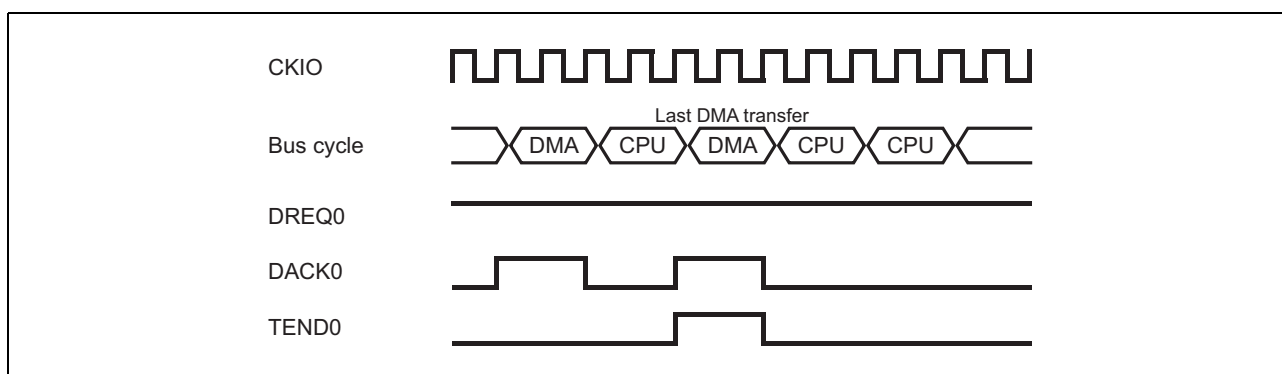


Figure 9.23 TEND0 Output Timing

### 9.7.7 DMA Transfer End Interrupt

The DMA transfer end interrupt is an interrupt request signal that indicates that a DMA transaction is completed.

There is an independent DMA transfer end interrupt for each channel.

When the transfer of data equivalent to the total transfer byte count loaded to the CRTB (Current Transaction Byte) is completed, 1 is set in END of the CHSTAT\_n register. In this case, when 0 is set in DEM of the CHCFG\_n register, the DMA transfer end interrupt is output (n = 0 to 15). (When writeback is performed in link mode, the signal is output after the writeback operation.)

When 0 is set in LV of the header in the read descriptor in link mode, 1 is set in DER of the CHSTAT\_n register. In this case, when 0 is set in DIM of the header, the DMA transfer end interrupt is output.

Table 9.20 Assertion Conditions of DMA Transfer End Interrupt

Source	Condition	DMA Transfer End Interrupt Mask Signal
DMA transaction end	When the transfer of data equivalent to the total transfer byte count loaded to the CRTB (Current Transaction Byte) is completed with an OKAY response (or after the writeback operation when writeback is performed in link mode)	DEM bit of the CHCFG_n register
Descriptor invalid	When 0 is set in LV of the header in the read descriptor in link mode while 0 is set in DIM of the header	DIM bit of the header

### 9.7.8 DMA Error Interrupt

If an error response is received for a DMA transfer or descriptor access, the DMAC regards it as an error and stops the transfer. Upon receiving an error response, the EN bit of the CHSTAT\_n register of transferring channel n is cleared to 0 and 1 is set in the ER bit (n = 0 to 15). Also, the DMA error interrupt is output.

The DMA error interrupt cannot be masked.

Once an error occurs, the data of the whole transfer cannot be guaranteed. Be sure to start the transaction again from the beginning by following the procedure below.

1. Set 1 in the SWRST bit of the CHCTRL\_n register.
2. Set each register again.

### 9.7.9 Interval Count Function

The interval at which a DMA transfer is executed can be adjusted by setting the ITVL bit of the channel interval register (CHITVL\_n). This function is intended to prevent the DMA controller from occupying the bus all the time.

When a read or write operation is completed, a countdown starts from the value set in CHITVL\_n. The next internal request is not executed until the count value reaches 0.

The following figure shows an example of how this works.

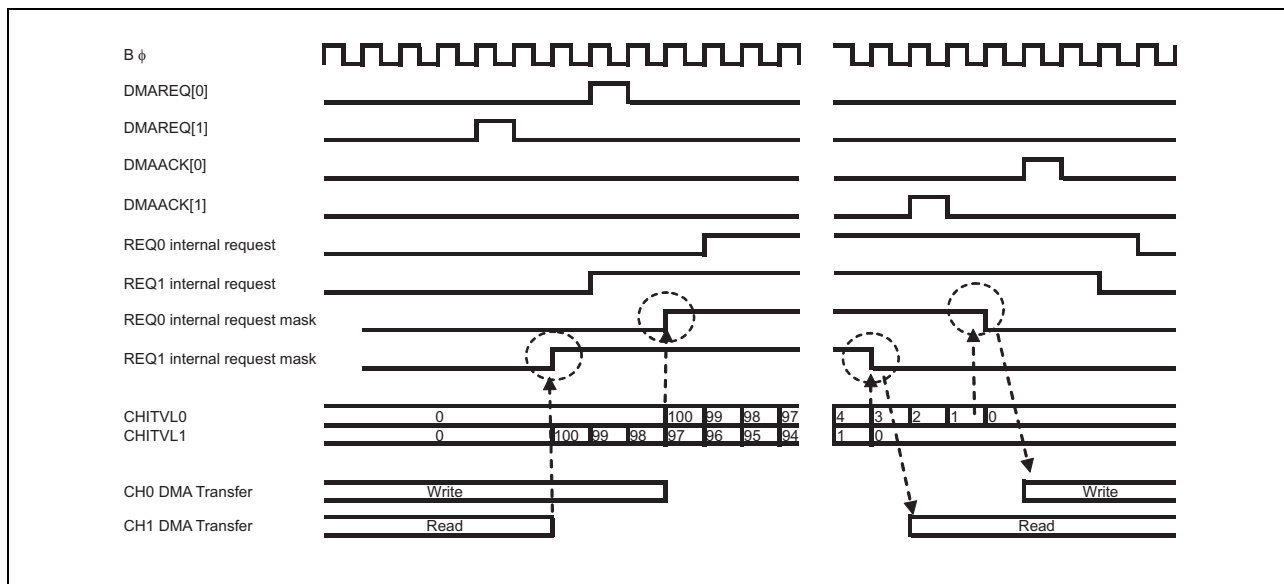


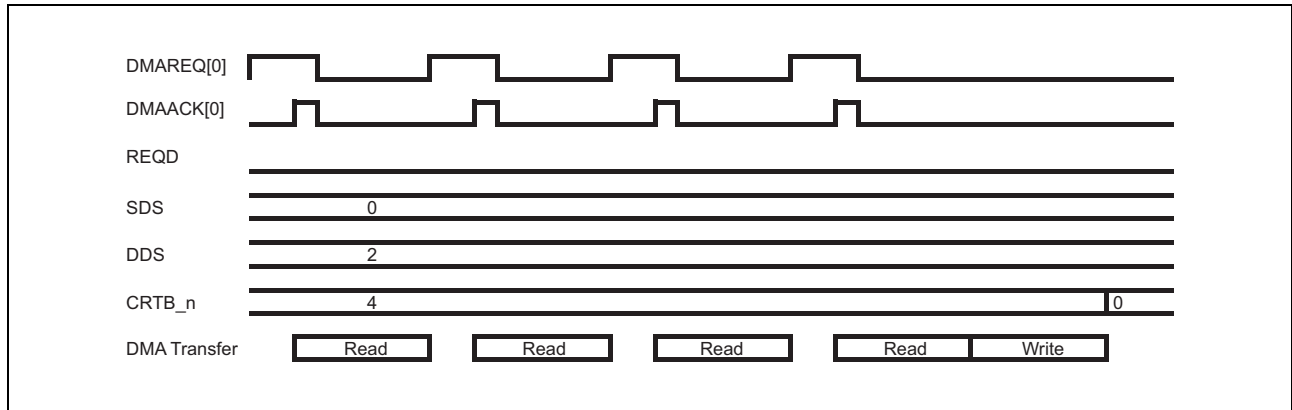
Figure 9.24 Interval Count



### 9.7.10 Difference in Operation Due to the Transfer Size

#### (1) When the Source Transfer Size Is Smaller

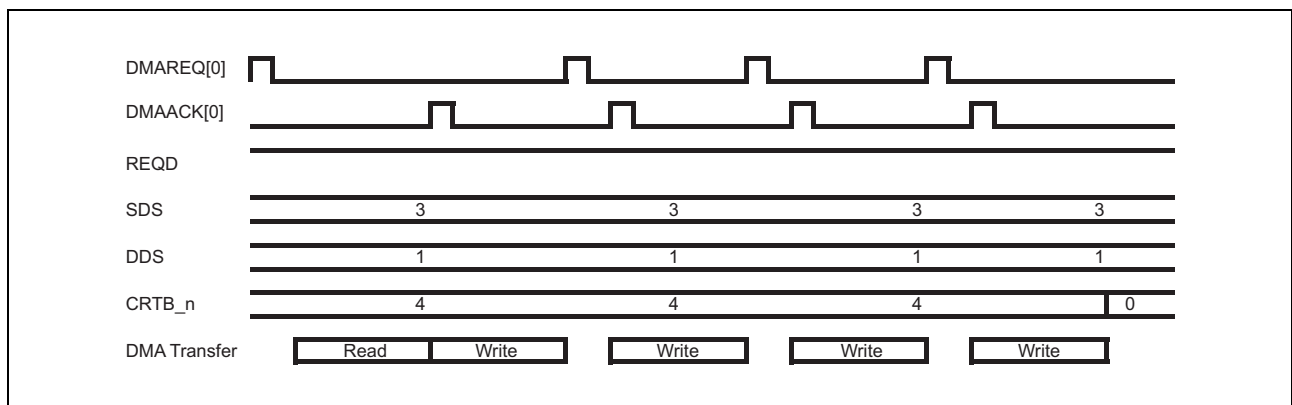
When the read of data equivalent to the destination data size is completed, the data is written to the destination. The following figure shows a timing chart where the source transfer size is 8 bits and the destination transfer size is 32 bits (in the case of rising edge detection).



**Figure 9.25** When the Source Transfer Size Is Smaller (LVL = 0, HIEN = 1, REQD = 0, SDS < DDS in CHCFG\_n)

#### (2) When the Destination Transfer Size Is Smaller

Since the source transfer size is larger, multiple destination writes occur after a single source read. The following figure shows a timing chart where the source transfer size is 64 bits and the destination transfer size is 16 bits (in the case of rising edge detection) (1 is set in REQD of the CHCFG\_n register).

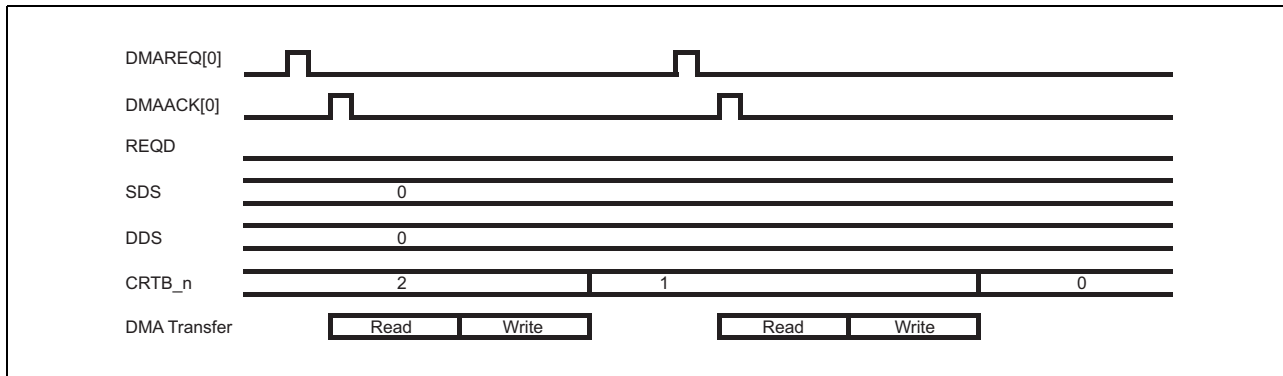


**Figure 9.26** When the Destination Transfer Size Is Smaller (LVL = 0, HIEN = 1, REQD = 1, SDS > DDS in CHCFG\_n)

(3) When the Source Transfer Size Is the Same as the Destination Transfer Size

Every time a DMA transfer request is detected, a source read and a destination write occur.

The following figure shows a timing chart where the source transfer size and the destination transfer size are both 8 bits (in the case of rising edge detection, with 1 set in REQD of the CHCFG\_n register).



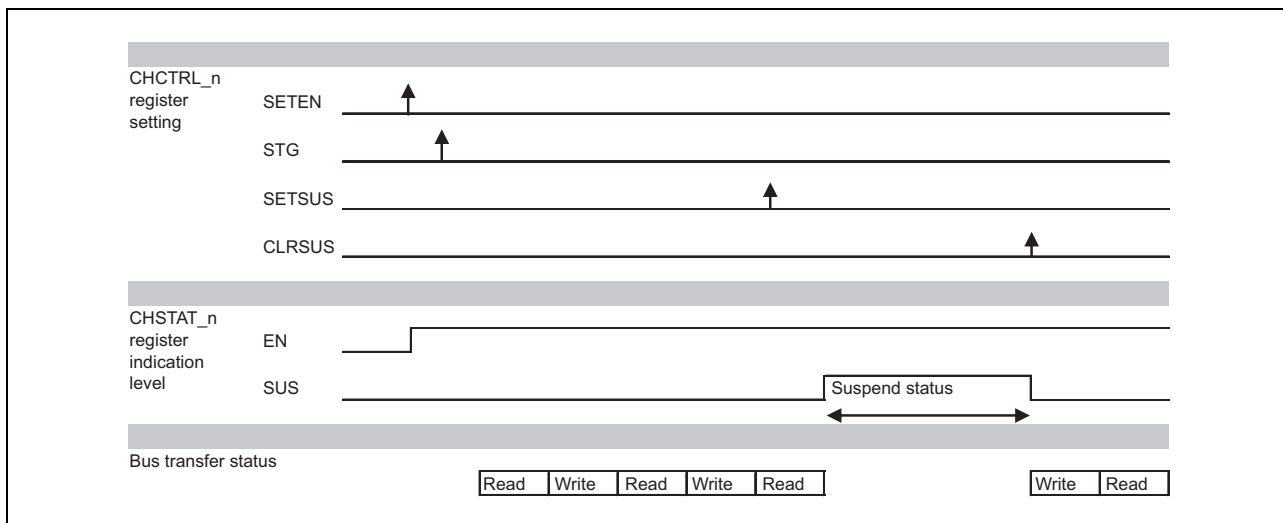
**Figure 9.27** When the Source Transfer Size Is the Same as the Destination Transfer Size (LVL = 0, HIEN = 1, REQD = 0, SDS = DDS in CHCFG\_n)

9.7.11 Transfer Status

The channel status register indicates the status of DMA transfer execution on a channel.

(1) Suspend

A DMA transfer can be suspended by using the SETSUS bit of CHCTRL\_n. In this case, if an ongoing bus cycle exists, the DMAC waits for that cycle to end before suspending the transfer. Writing 1 in the CLRSUS bit restores the DMA transfer from the suspend status.



**Figure 9.28** DMAC Suspend Status (Auto Request/Block Transfer)

In the above case, the DMA transfer is suspended after the read transfer is completed.

If there is any ongoing DMA transfer, the suspend status starts when that transfer is completed. To make sure that the transfer is suspended, read the CHSTAT or DSTAT\_SUS register, after setting SETSUS, and check that 1 is set in the SUS bit for the relevant channel.

(2) Transfer Stop

If 1 is written to CLREN while a DMA transaction is in progress, the DMA transaction for the corresponding channel can be stopped. For the post-stop processing, two modes are supported: one sweeps out the data remaining in the buffer when the transaction is stopped (SBE = 1) and the other does not (SBE = 0). One of these modes can be selected using the SBE bit of the CHCFG\_n register. By default, SBE is set to 0.

When this sweep mode is enabled and CLREN is set to 1, and if a DMA transaction is stopped with data remaining in the DMAC buffer, the transaction is completed after the DMAC sweeps the data.

(a) Transfer Stop (Buffer Sweep Disabled - SBE = 0)

If 1 is set in CLREN during a DMA transfer, the DMA transfer is stopped. The stop timing depends on the value set in REQD. After stopping a DMA transfer, be sure to set 1 in SWRST to clear the DMA internal status before setting the next transfer.

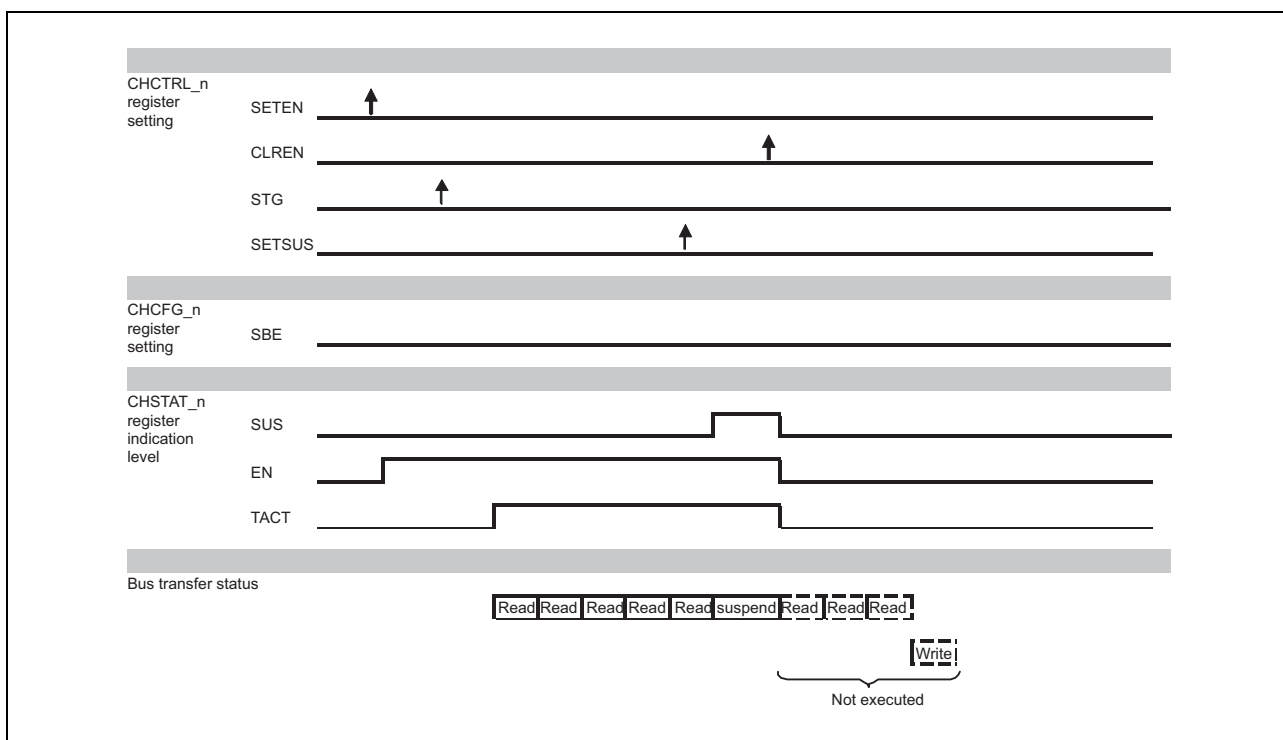
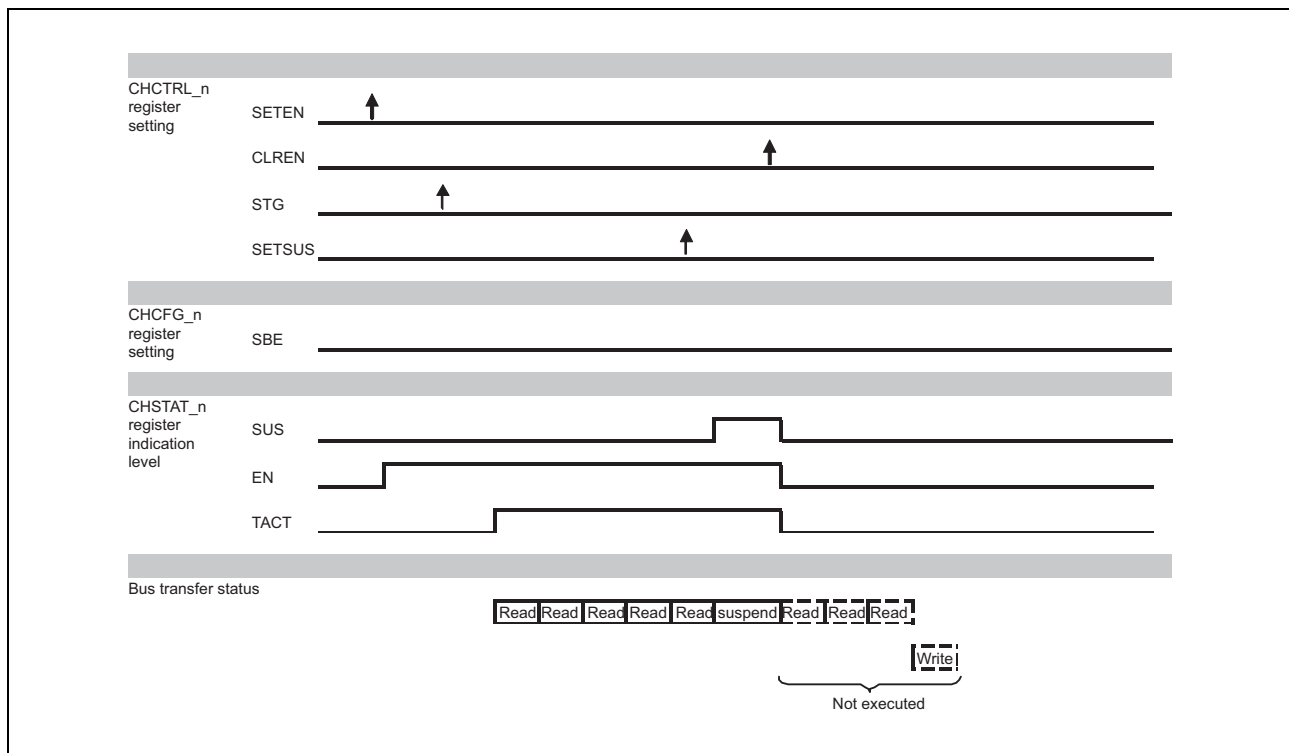


Figure 9.29 DMAC Transfer Stop

- The setting of the TACT bit being 0 indicates that the channel has been brought to a complete stop.
- If an ongoing DMA transfer is stopped before it is completed, the DMA transfer end interrupt is not asserted.
- If 0 is set in REQD, the DMA transfer is stopped when the next read is completed. (If the buffer contains any data that can be written, the DMA transfer is stopped after the data is written.)
- If 1 is set in REQD, the DMA transfer is stopped when the next write is completed.

**(b) Transfer Stop (Buffer Sweep Enabled - SBE = 1)**

If 1 is set in CLREN during a DMA transfer, the DMA transfer is stopped. When 0 is set in REQD, the DMA transfer is stopped after the DMAC sweeps (writes) the already read data. If 1 is set in REQD to use hardware requests, do not use the sweep mode. After stopping a DMA transfer, be sure to set 1 in SWRST to clear the DMAC internal status before setting the next transfer.



**Figure 9.30 DMA Transfer Stop (Buffer Sweep Mode)**

- The setting of the TACT bit being 0 indicates that the channel has been brought to a complete stop.
- If a transfer is stopped in sweep mode (SBE = 1) during the fifth read transfer by setting SETSUS and then CLREN, the read data is written before the DMA transfer is stopped.

**(c) Channel Stop Check Method**

Even when the EN bit is cleared to 0, the DMA transfer cannot be stopped immediately, if the bus is already executing the transfer. Therefore, in order to make sure that the DMAC has been brought to a complete stop, check that the EN bit and TACT bit are both set to 0.

**(d) Transfer Stop Procedure**

The transfer stop procedure is described below.

1. Set 1 in SETSUS of CHCTRL\_n.
2. Repeat polling until the SUS bit of CHSTAT\_n is set to 1. (If EN is already set to 0, the DMAC has already been stopped. Go to step 6.)
3. Set 1 in CLREN of CHCTRL\_n.
4. When 0 is set in SBE, the transfer is stopped according to the value of REQD. When 1 is set in SBE, the sweep mode is enabled. When 1 is set in SBE, set 0 in REQD.
5. Read CHSTAT\_n to check that 0 is set in the TACT bit. When TACT is set to 0, it means that the DMAC has been brought to a complete stop. When TACT is set to 1, repeat polling until this bit is set to 0.

- To execute the next DMA transfer after stopping a transfer, be sure to set 1 in the SWRST (software reset) bit of CHCTRL\_n before the next DMA transfer starts.

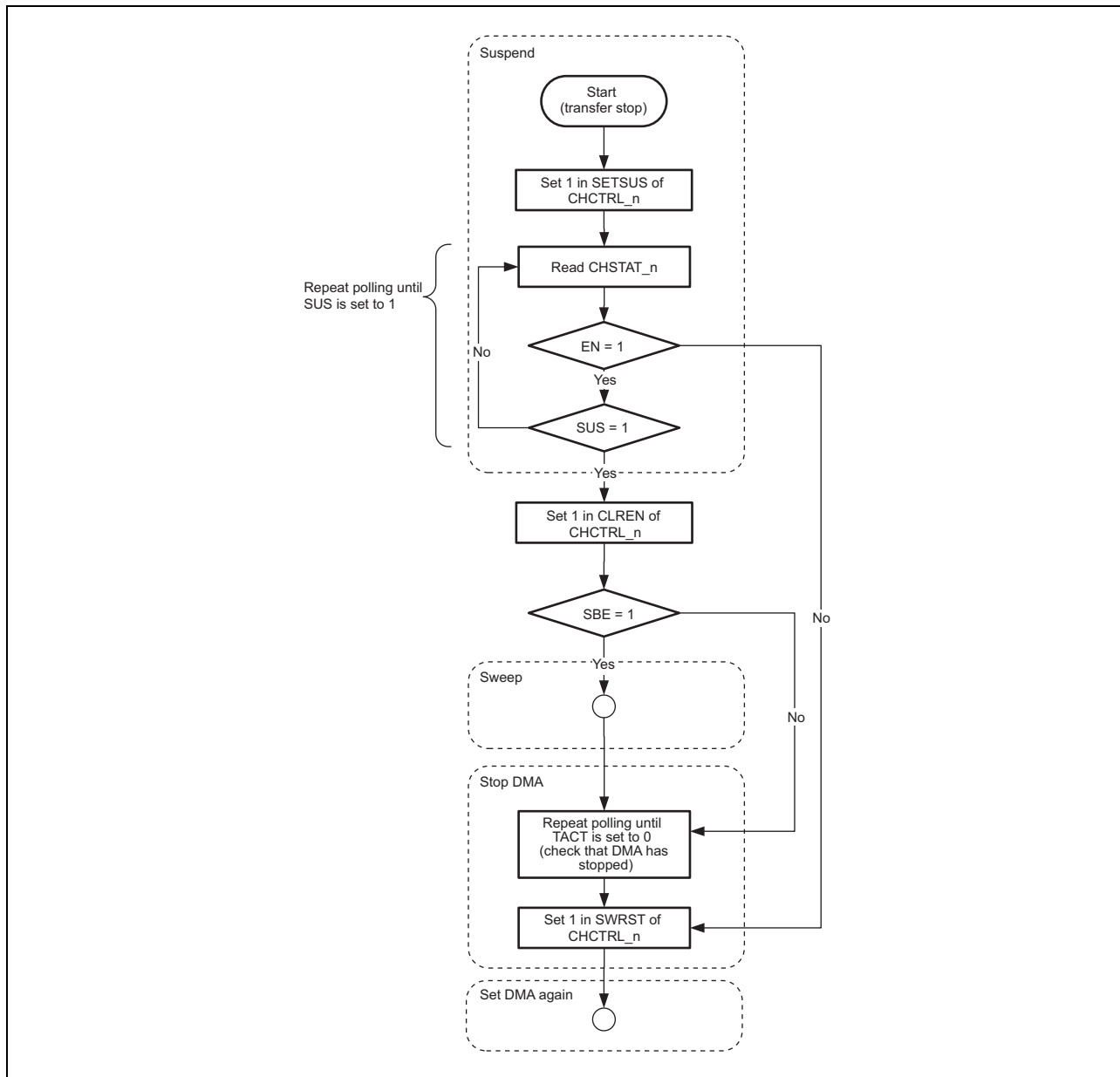


Figure 9.31 Transfer Stop Flow

## 9.8 DMA Setting Examples

Setting examples applicable when DMA transfer is executed using the direct memory access controller are shown in the following.

The transfer conditions for these setting examples are as follows.

**Table 9.21 Transfer Condition List for DMA Transfer Setting Examples**

	DMA Mode	Transfer Mode	Transfer Request
Setting example 1	Register	Single	Hardware
Setting example 2	Register	Block	Software
Setting example 3	Register (continuous execution)	Block	Software
Setting example 4	Link	Block	Software

For details of the settings, see the individual setting examples.

### 9.8.1 Setting Example 1 (Register Mode/Hardware Request)

The following table shows a setting example applicable when DMA transfer is executed using the settings shown below.

**Table 9.22 DMA Transfer Setting Example 1**

Item	Description	
Channel used	3	
DMA mode	Register	
Transfer mode	Single transfer	
Register set used	Next0	
Source/destination	Source	Destination
Start address	1111_0000H	2222_0000H
Address direction	Increment	Increment
Data size	32 bits	32 bits
DMA transfer byte count	64 bytes	
DMA transfer request	Rising edge detection by hardware	
DMAACK signal	Level output during read	
DMA transfer end interrupt mask	Not masked	
CACHE setting	Default value	

#### Setting example 1

N0SA = 1111\_0000H (source address)

N0DA = 2222\_0000H (destination address)

N0TB = 0000\_0040H (transfer byte count)

CHCFG = 0002\_2123H (configuration)

CHITVL = 0000\_0000H (interval)

CHEXT = 0000\_0000H (CACHE setting)

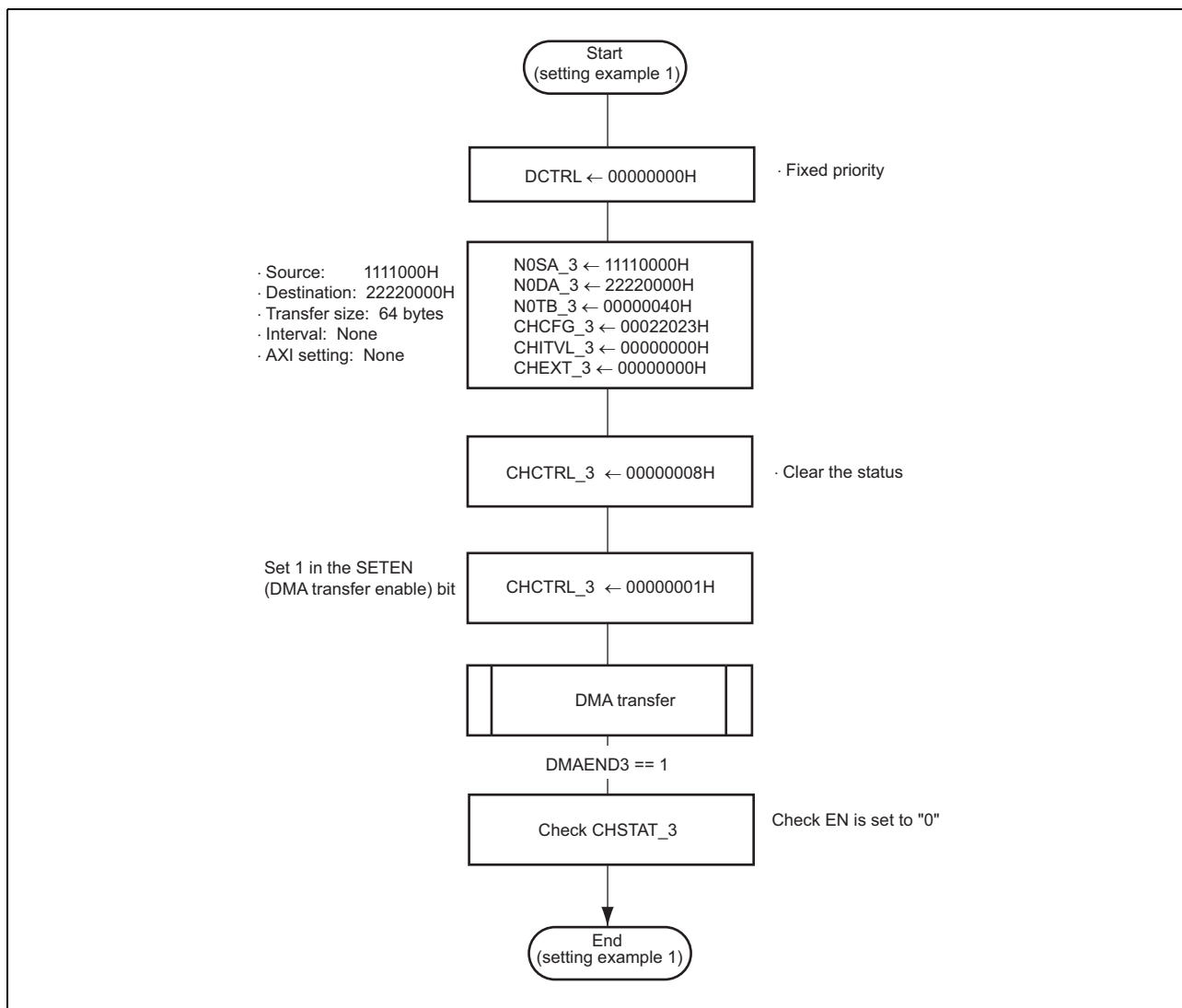


Figure 9.32 Setting Example 1

## 9.8.2 Setting Example 2 (Register Mode/Software Request)

The following table shows a setting example applicable when DMA transfer is executed using the settings shown below.

**Table 9.23 DMA Transfer Setting Example 2**

Item	Description	
Channel used	2	
Priority control	Round robin	
DMA mode	Register	
Transfer mode	Block transfer	
Register set used	Next1	
Source/destination	Source	Destination
Start address	0FFF_E000H	3333_0000H
Address direction	Increment	Increment
Data size	8 bits	256 bits
DMA transfer byte count	128 bytes	
DMA transfer request	Auto request	
DMAACK signal	Masked	
DMA transfer end interrupt mask	Not masked	
CACHE setting	Default value	

### Setting example 2

DCTRL = 0000\_0001H (DMA setting)

N1SA = 0FFF\_E000H (source address)

N1DA = 3333\_0000H (destination address)

N1TB = 0000\_0080H (transfer byte count)

CHCFG = 1045\_0402H (configuration)

CHITVL = 0000\_0000H (interval)

CHEXT = 0000\_0000H (CACHE setting)



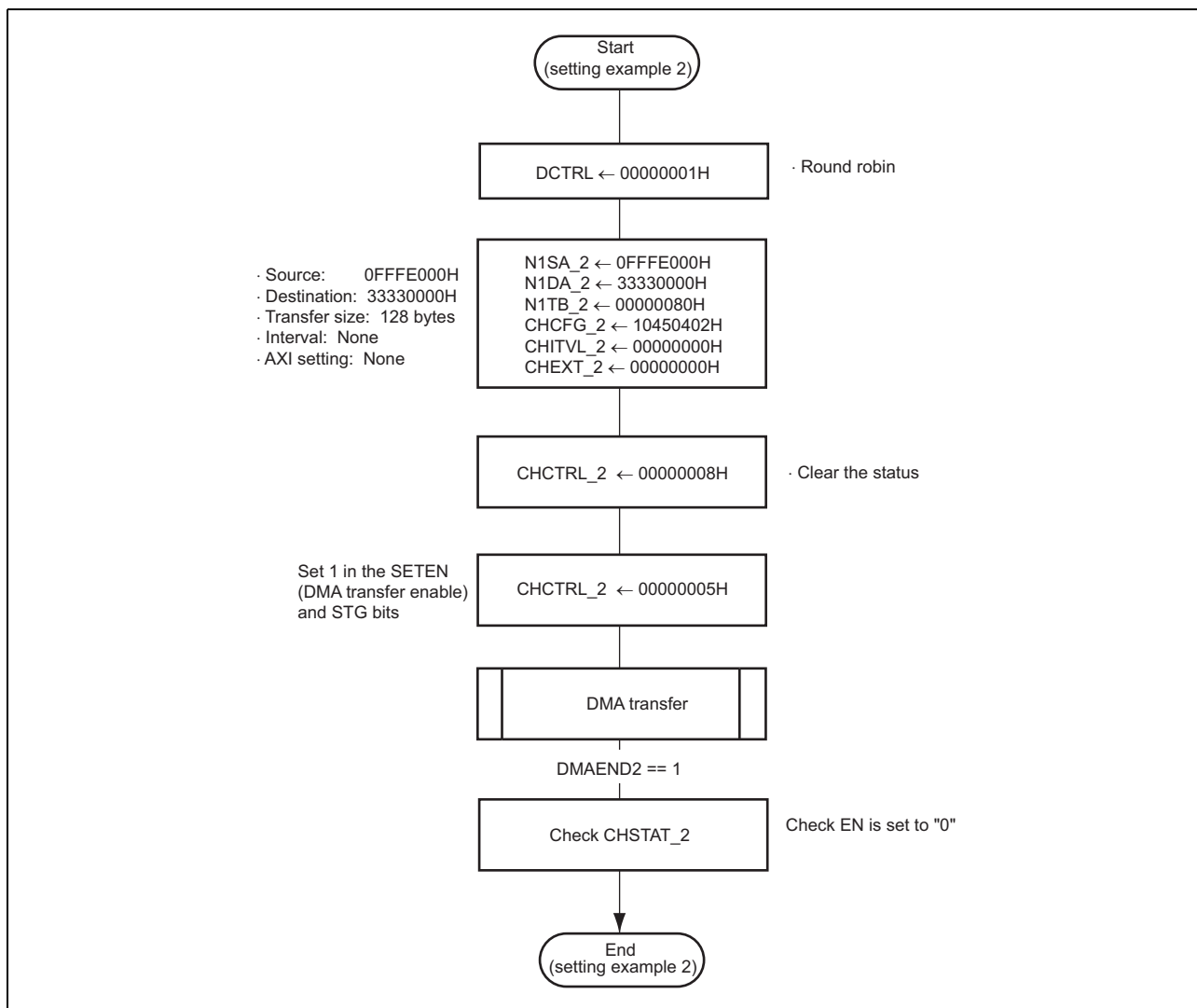


Figure 9.33 Setting Example 2

### 9.8.3 Setting Example 3 (Register Mode/Continuous Execution)

The following table shows a setting example applicable when DMA transfer is executed using the settings shown below.

**Table 9.24 DMA Transfer Setting Example 3**

Item	Description	
Channel used	1	
Priority control	Round robin	
DMA mode	Register	
Transfer mode	Block transfer	
Register set used	Use Next0 and then Next1 continuously	
Next0	Source	Destination
Start address	1111_0000H	3333_0000H
Address direction	Fixed	Fixed
Data size	32 bits	512 bits
DMA transfer byte count	512 bytes	
Next1	Source	Destination
Start address	2222_0000H	4444_0000H
Address direction	Fixed	Fixed
Data size	32 bits	512 bits
DMA transfer byte count	2048 bytes	
DMA transfer request	Auto request	
DMAACK signal	Not output	
DMA transfer end interrupt mask	Mask the DMA transfer end interrupt upon completion of Next0	
CACHE setting	Default value	

#### Setting example 3

DCTRL = 0000\_0001H (DMA setting)

N0SA = 1111\_0000H (source address)

N0DA = 3333\_0000H (destination address)

N0TB = 0000\_0200H (transfer byte count)

N1SA = 2222\_0000H (source address)

N1DA = 4444\_0000H (destination address)

N1TB = 0000\_0800H (transfer byte count)

CHCFG = 6176\_2001H (configuration)

CHITVL = 0000\_0000H (interval)

CHEXT = 0000\_0000H (CACHE setting)

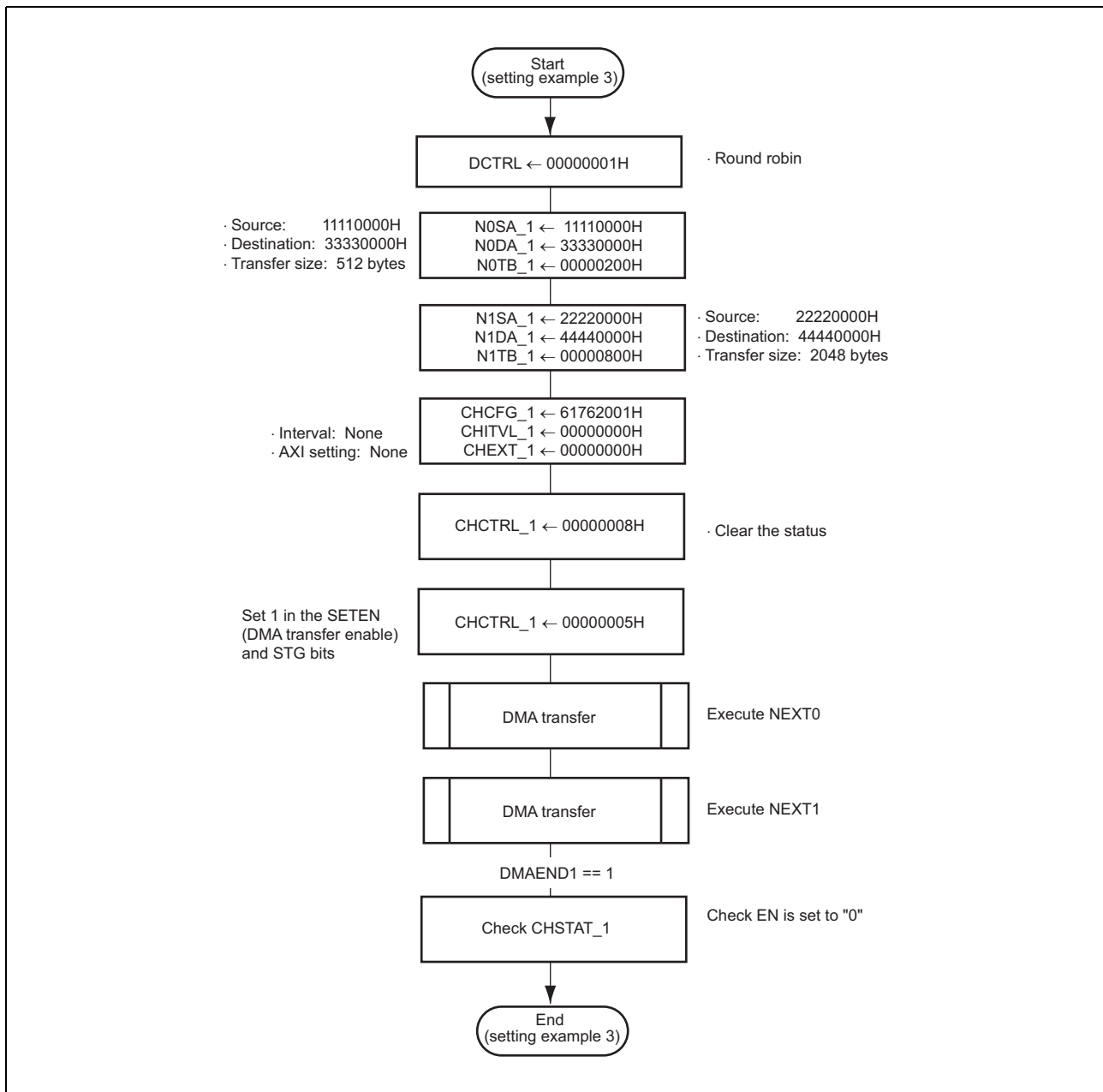


Figure 9.34 Setting Example 3

### 9.8.4 Setting Example 4 (Link Mode)

The following table shows a setting example applicable when DMA transfer is executed using the settings shown below.

Table 9.25 DMA Transfer Setting Example 4

Item	Description
Channel used	0
Priority control	Round robin
DMA mode	Link
Transfer mode	Block transfer
Register set used	—
Descriptor start address	0000_1000H

**Table 9.26 DMA Transfer Setting Example 4 (Descriptor 1)**

Item	Description	
Descriptor start address	0000_1000H	
Next descriptor start address	0000_2000H	
Transfer mode	Block transfer	
Next0	Source	Destination
Start address	1111_0000H	3333_0000H
Address direction	Increment	Increment
Data size	32 bits	32 bits
DMA transfer byte count	2048 bytes	
DMA transfer request	Auto request trigger (STG)	
DMAACK signal	Not output	
DMA transfer end interrupt mask	Masked	
CACHE setting	Default value	
header		
DMA interrupt when LV = 1	Issued (DIM = 0)	
LV writeback	Done (WBD = 0)	
Next link address	Available (LE = 0)	
Descriptor valid	Valid (LV = 1)	

**Table 9.27 DMA Transfer Setting Example 4 (Descriptor 2)**

Item	Description	
Descriptor start address	0000_2000H	
Next descriptor start address	0000_5000H	
Transfer mode	Block transfer	
Next0	Source	Destination
Start address	4444_0000H	5555_0000H
Address direction	Increment	Increment
Data size	64 bits	256 bits
DMA transfer byte count	1024 bytes	
DMA transfer request	Auto request trigger (STG)	
DMAACK signal	Not output	
DMA transfer end interrupt mask	Masked	
CACHE setting	Default value	
header		
DMA interrupt when LV = 1	Issued (DIM = 0)	
LV writeback	Done (WBD = 0)	
Next link address	Available (LE = 0)	
Descriptor valid	Valid (LV = 1)	

**Table 9.28 DMA Transfer Setting Example 4 (Descriptor 3)**

Item	Description	
Descriptor start address	0000_5000H	
Next descriptor start address	—	
Transfer mode	Block transfer	
Next0	Source	Destination
Start address	7777_0000H	AAAA_0000H
Address direction	Increment	Increment
Data size	512 bits	512 bits
DMA transfer byte count	4096 bytes	
DMA transfer request	Auto request trigger (STG)	
DMAACK signal	Not output	
DMA transfer end interrupt mask	Not masked	
CACHE setting	Default value	
header		
DMA interrupt when LV = 1	Issued (DIM = 0)	
LV writeback	Done (WBD = 0)	
Next link address	Not available (LE = 1)	
Descriptor valid	Valid (LV = 1)	

**Setting example 4**

DCTRL= 0000\_0001H (DMA setting)

NXLA = 0000\_1000H (descriptor start address)

CHCFG = 8000\_0000H (configuration)

**Table 9.29 Descriptor Setting**

	Descriptor 1	Descriptor 2	Descriptor 3
header	0000_0001H	0000_0001H	0000_0003H
SA (Source Address)	1111_0000H	4444_0000H	7777_0000H
DA (Destination Address)	3333_0000H	5555_0000H	AAAA_0000H
TB (Transaction Byte)	0000_0800H	0000_0400H	0000_1000H
CFG (Configuration)	8142_2008H	8145_3008H	8046_6008H
ITVL (Interval)	0000_0000H	0000_0000H	0000_0000H
EXT (Extension)	0000_0000H	0000_0000H	0000_0000H
NXLA (Next Link Address)	0000_2000H	0000_5000H	0000_0000H

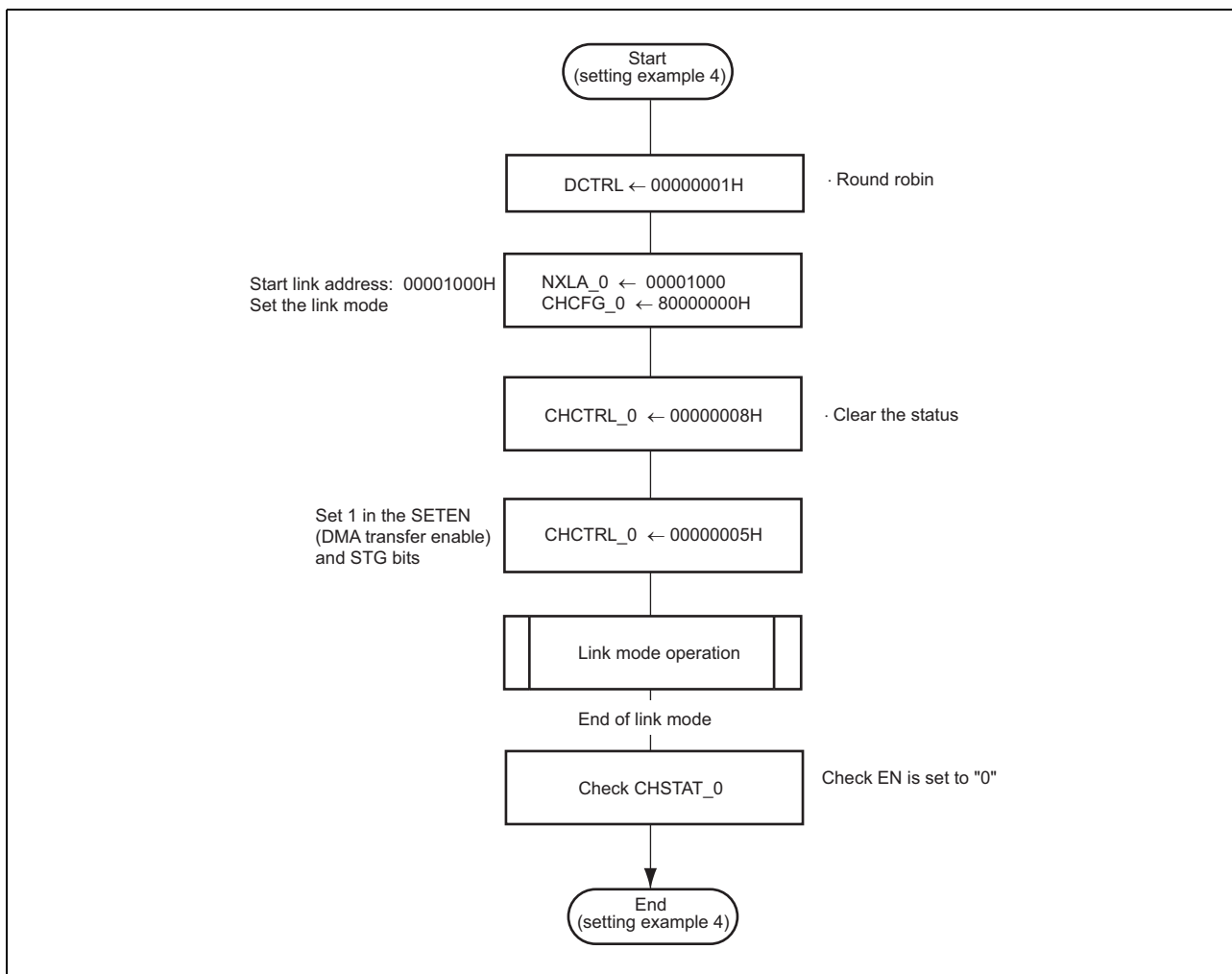


Figure 9.35 Setting Example 4

### 9.8.5 Next Register Set Continuous Execution Setting

The following figure shows the flowchart for executing DMA transfers continuously by using two Next register sets in register mode. While a DMA transaction is being executed using one Next register set, the other Next register set is set in order to continue to execute DMA transfers.

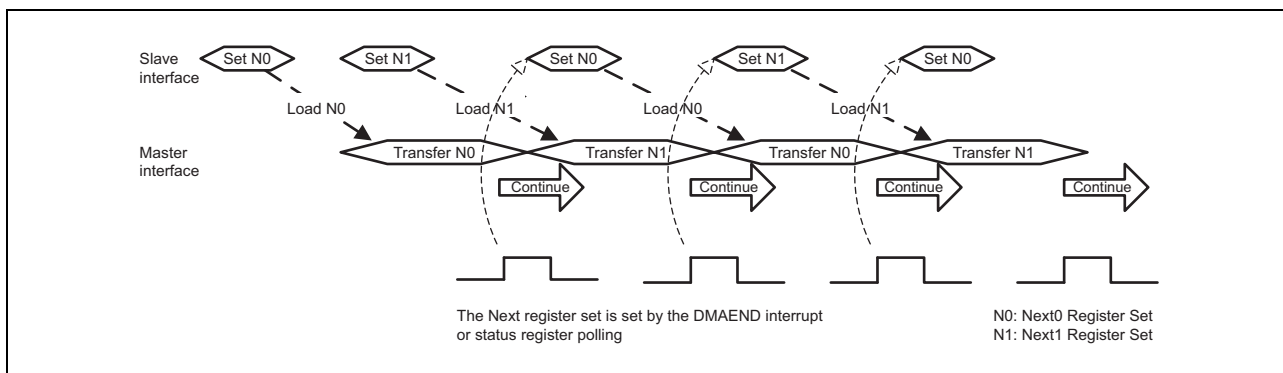


Figure 9.36 Image of Next Register Set Continuous Execution

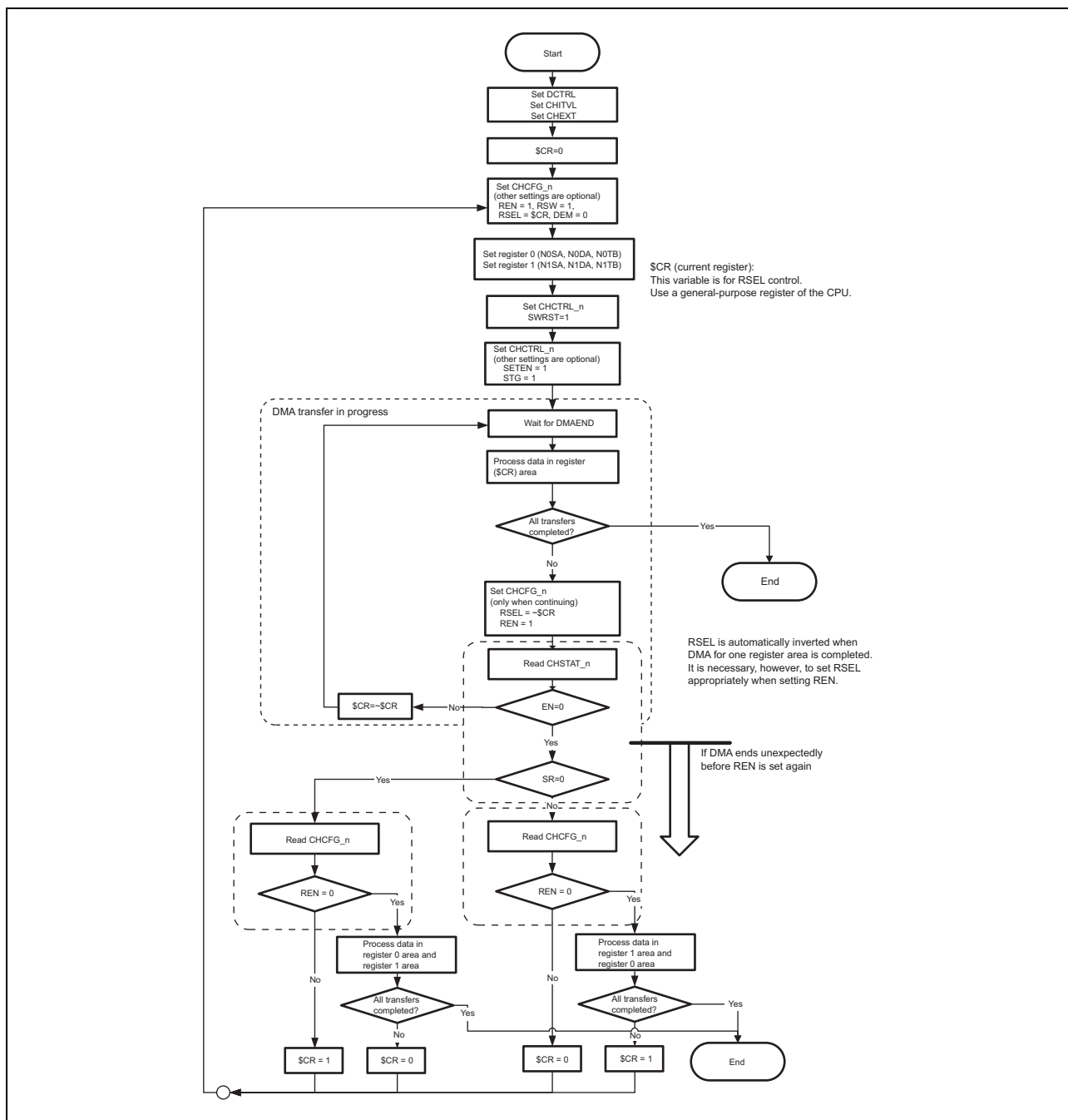


Figure 9.37 Example of Continuous DMA Execution by Using a Next Register Set

- Supplementary information

First, save the data of the register sets to be used for DMA transfers (0 (N0SA, N0DA, and N0TB) and 1 (N1SA, N1DA, and N1TB)) to a general-purpose register of the CPU (the values of this register is referred to as \$SCR for the sake of convenience).

Each time the DMA transfers for one register set are completed (the DMA transfer end interrupt is output), REN is automatically cleared to 0. In order to continue to execute DMA transfers, it is necessary to set REN of the CHCFG\_n register every time the DMA transfer end interrupt is asserted. This register also contains the RSEL bit, and the value of this bit needs to be set appropriately as well. Therefore, use \$SCR.

In this mode, two Next register sets are executed continuously. However, if CLREN is not set before the DMA transaction is completed (the next DMA transfer end interrupt is output), continuous execution stops. In this case, how much of data has been transferred can be checked by reading the SR and EN bits of the CHSTAT\_n register and the REN bit of the CHCFG\_n register. To restart the DMA transaction, follow the flowchart shown above.

9.9 Note

9.9.1 Divided Output of DACK0 and TEND0

When transferring 4 bytes or more to an 8-bit or 16-bit external device or transferring 2 bytes or more to an 8-bit external device, each DMA transfer unit is divided into multiple bus cycles. Note that, if the setting is such that DMA transfer is divided into multiple bus cycles and CS is negated between bus cycles, the DACK0 output and the TEND0 output is divided to align data as with CS. Figure 9.38 shows an example.

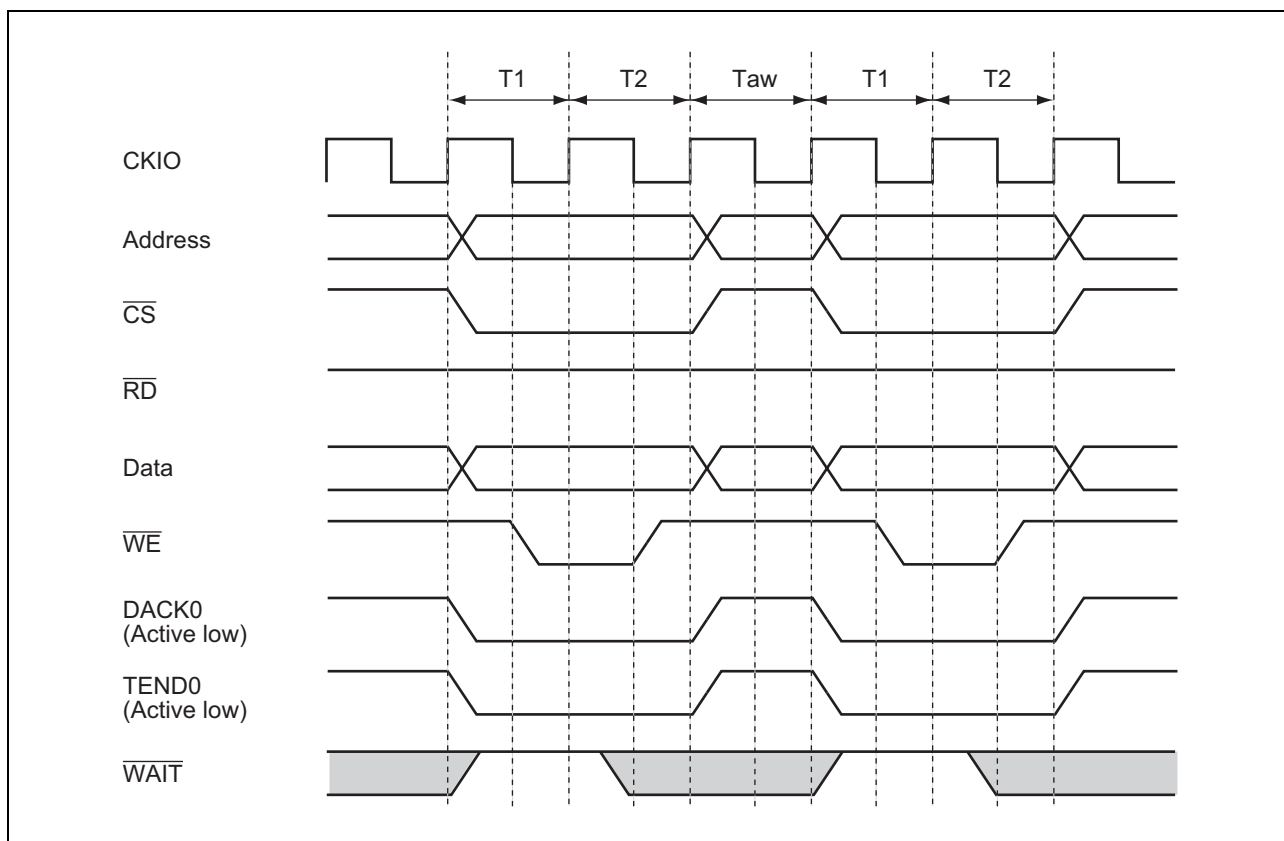


Figure 9.38 Example of TEND0 Divided Output Timing



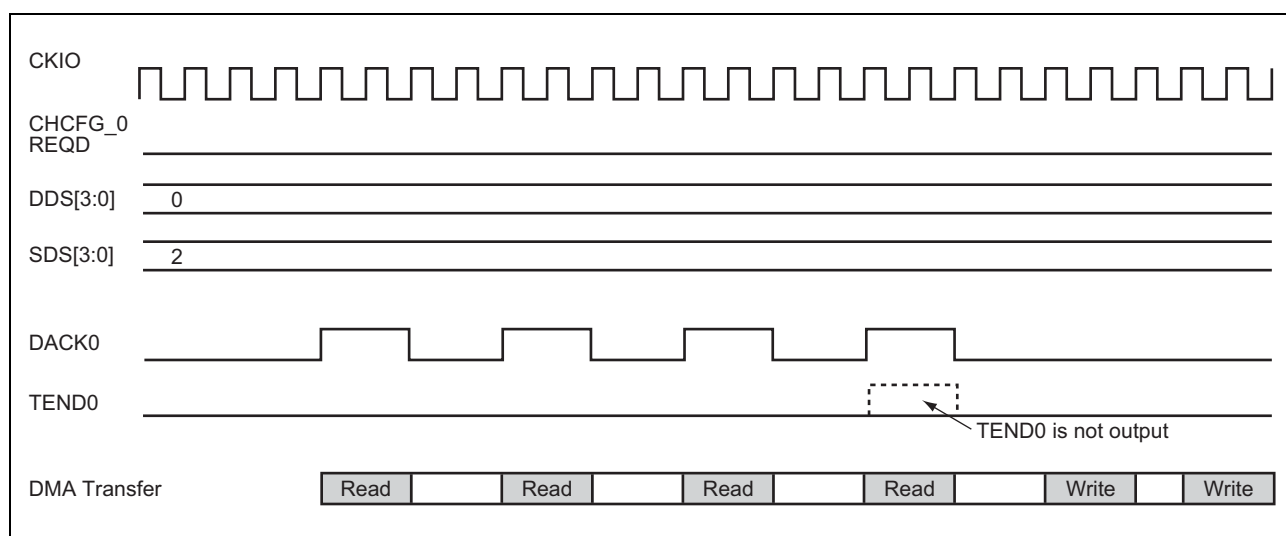
### 9.9.2 TEND0 Not Output

Note that TEND0 may not be output depending on the combination of the bits DDS[3:0], SDS[3:0] and REQD in the CHCFG\_0 register.

Table 9.30 shows when TEND0 is not output and Figure 9.39 shows an operation example.

**Table 9.30 Bit Combination when TEND0 Is Not Output**

CHCFG_0 Register			
REQD	DDS	SDS	TEND0 Output
1	—	—	Output
0	DDS > SDS		Output
	DDS = SDS		Output
	DDS < SDS		Not output



**Figure 9.39 TEND0 Not Output**

## 10. Multi-Function Timer Pulse Unit 2

This LSI has an on-chip multi-function timer pulse unit 2 that comprises five 16-bit timer channels.

### 10.1 Features

- Maximum 16 pulse input/output lines
- Selection of eight counter input clocks for each channel
- The following operations can be set:
  - Waveform output at compare match
  - Input capture function
  - Counter clear operation
  - Multiple timer counters (TCNT) can be written to simultaneously
  - Simultaneous clearing by compare match and input capture is possible
  - Register simultaneous input/output is possible by synchronous counter operation
  - A maximum 12-phase PWM output is possible in combination with synchronous operation
- Buffer operation settable for channels 0, 3, and 4
- Phase counting mode settable independently for each of channels 1 and 2
- Cascade connection operation
- Fast access via internal 16-bit bus
- 25 interrupt sources
- Automatic transfer of register data
- A/D converter start trigger can be generated
- Module standby mode can be settable
- A total of six-phase waveform output, which includes complementary PWM output, and positive and negative phases of reset PWM output by interlocking operation of channels 3 and 4, is possible.
- AC synchronous motor (brushless DC motor) drive mode using complementary PWM output and reset PWM output is settable by interlocking operation of channels 0, 3, and 4, and the selection of two types of waveform outputs (chopping and level) is possible.
- In complementary PWM mode, interrupts at the crest and trough of the counter value and A/D converter start triggers can be skipped.

Table 10.1 Functions of Multi-Function Timer Pulse Unit 2

Item	Channel 0	Channel 1	Channel 2	Channel 3	Channel 4
Count clock	P0 $\phi$ /1	P0 $\phi$ /1	P0 $\phi$ /1	P0 $\phi$ /1	P0 $\phi$ /1
	P0 $\phi$ /4	P0 $\phi$ /4	P0 $\phi$ /4	P0 $\phi$ /4	P0 $\phi$ /4
	P0 $\phi$ /16	P0 $\phi$ /16	P0 $\phi$ /16	P0 $\phi$ /16	P0 $\phi$ /16
	P0 $\phi$ /64	P0 $\phi$ /64	P0 $\phi$ /64	P0 $\phi$ /64	P0 $\phi$ /64
	TCLKA	P0 $\phi$ /256	P0 $\phi$ /1024	P0 $\phi$ /256	P0 $\phi$ /256
	TCLKB	TCLKA	TCLKA	P0 $\phi$ /1024	P0 $\phi$ /1024
	TCLKC	TCLKB	TCLKB	TCLKA	TCLKA
	TCLKD		TCLKC	TCLKB	TCLKB
General registers	TGRA_0	TGRA_1	TGRA_2	TGRA_3	TGRA_4
	TGRB_0	TGRB_1	TGRB_2	TGRB_3	TGRB_4
	TGRE_0				
General registers/ buffer registers	TGRC_0	—	—	TGRC_3	TGRC_4
	TGRD_0			TGRD_3	TGRD_4
	TGRF_0				
I/O pins	TIOC0A	TIOC1A	TIOC2A	TIOC3A	TIOC4A
	TIOC0B	TIOC1B	TIOC2B	TIOC3B	TIOC4B
	TIOC0C			TIOC3C	TIOC4C
	TIOC0D			TIOC3D	TIOC4D
Counter clear function	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture
Compare match output	0 output	√	√	√	√
	1 output	√	√	√	√
	Toggle output	√	√	√	√
Input capture function	√	√	√	√	√
Synchronous operation	√	√	√	√	√
PWM mode 1	√	√	√	√	√
PWM mode 2	√	√	√	—	—
Complementary PWM mode	—	—	—	√	√
Reset PWM mode	—	—	—	√	√
AC synchronous motor drive mode	√	—	—	√	√
Phase counting mode	—	√	√	—	—
Buffer operation	√	—	—	√	√
Activation of direct memory access controller	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture	TGR compare match or input capture and TCNT overflow or underflow
A/D converter start trigger	TGRA_0 compare match or input capture TGRE_0 compare match	TGRA_1 compare match or input capture	TGRA_2 compare match or input capture	TGRA_3 compare match or input capture	TGRA_4 compare match or input capture TCNT_4 underflow (trough) in complementary PWM mode

Item	Channel 0	Channel 1	Channel 2	Channel 3	Channel 4
Interrupt sources	7 sources <ul style="list-style-type: none"> <li>• Compare match or input capture 0A</li> <li>• Compare match or input capture 0B</li> <li>• Compare match or input capture 0C</li> <li>• Compare match or input capture 0D</li> <li>• Compare match 0E</li> <li>• Compare match 0F</li> <li>• Overflow</li> </ul>	4 sources <ul style="list-style-type: none"> <li>• Compare match or input capture 1A</li> <li>• Compare match or input capture 1B</li> <li>• Overflow</li> <li>• Underflow</li> </ul>	4 sources <ul style="list-style-type: none"> <li>• Compare match or input capture 2A</li> <li>• Compare match or input capture 2B</li> <li>• Overflow</li> <li>• Underflow</li> </ul>	5 sources <ul style="list-style-type: none"> <li>• Compare match or input capture 3A</li> <li>• Compare match or input capture 3B</li> <li>• Compare match or input capture 3C</li> <li>• Compare match or input capture 3D</li> <li>• Overflow</li> </ul>	5 sources <ul style="list-style-type: none"> <li>• Compare match or input capture 4A</li> <li>• Compare match or input capture 4B</li> <li>• Compare match or input capture 4C</li> <li>• Compare match or input capture 4D</li> <li>• Overflow or underflow</li> </ul>
A/D converter start request delaying function	—	—	—	—	<ul style="list-style-type: none"> <li>• A/D converter start request at a match between TADCORA_4 and TCNT_4</li> <li>• A/D converter start request at a match between TADCORB_4 and TCNT_4</li> </ul>
Interrupt skipping function	—	—	—	<ul style="list-style-type: none"> <li>• Skips TGRA_3 compare match interrupts</li> </ul>	<ul style="list-style-type: none"> <li>• Skips TCIV_4 interrupts</li> </ul>

## [Legend]

- √: Available  
 —: Not available

Figure 10.1 shows a block diagram.

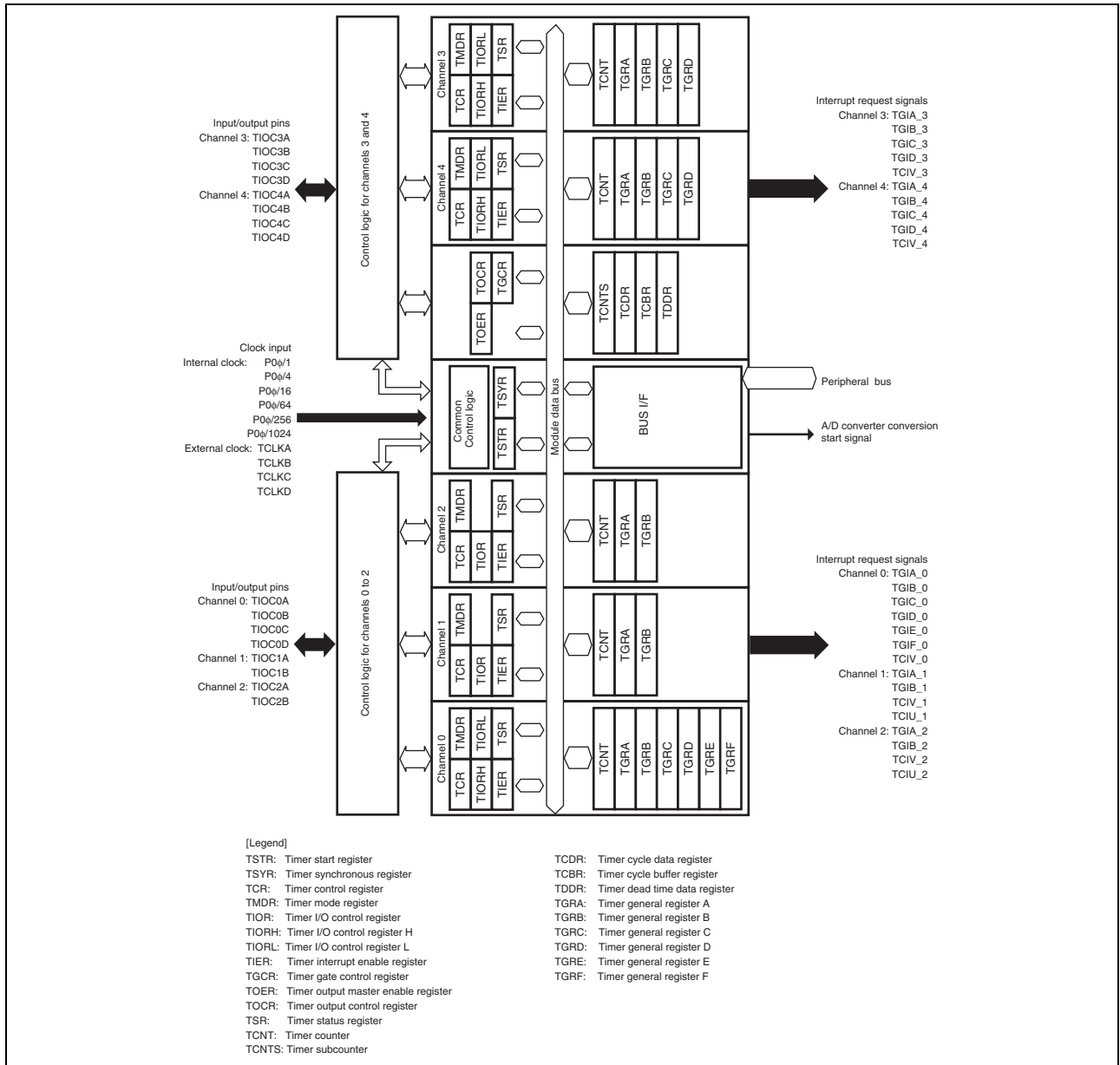


Figure 10.1 Block Diagram

## 10.2 Input/Output Pins

Table 10.2 shows the pin configuration

**Table 10.2 Pin Configuration**

Channel	Pin Name	I/O	Function
Common	TCLKA	Input	External clock A input pin (Channel 1 phase counting mode A phase input)
	TCLKB	Input	External clock B input pin (Channel 1 phase counting mode B phase input)
	TCLKC	Input	External clock C input pin (Channel 2 phase counting mode A phase input)
	TCLKD	Input	External clock D input pin (Channel 2 phase counting mode B phase input)
0	TIOC0A	I/O	TGRA_0 input capture input/output compare output/PWM output pin
	TIOC0B	I/O	TGRB_0 input capture input/output compare output/PWM output pin
	TIOC0C	I/O	TGRC_0 input capture input/output compare output/PWM output pin
	TIOC0D	I/O	TGRD_0 input capture input/output compare output/PWM output pin
1	TIOC1A	I/O	TGRA_1 input capture input/output compare output/PWM output pin
	TIOC1B	I/O	TGRB_1 input capture input/output compare output/PWM output pin
2	TIOC2A	I/O	TGRA_2 input capture input/output compare output/PWM output pin
	TIOC2B	I/O	TGRB_2 input capture input/output compare output/PWM output pin
3	TIOC3A	I/O	TGRA_3 input capture input/output compare output/PWM output pin
	TIOC3B	I/O	TGRB_3 input capture input/output compare output/PWM output pin
	TIOC3C	I/O	TGRC_3 input capture input/output compare output/PWM output pin
	TIOC3D	I/O	TGRD_3 input capture input/output compare output/PWM output pin
4	TIOC4A	I/O	TGRA_4 input capture input/output compare output/PWM output pin
	TIOC4B	I/O	TGRB_4 input capture input/output compare output/PWM output pin
	TIOC4C	I/O	TGRC_4 input capture input/output compare output/PWM output pin
	TIOC4D	I/O	TGRD_4 input capture input/output compare output/PWM output pin

Note: For the pin configuration in complementary PWM mode, see Table 10.52 in section 10.4.8, Complementary PWM Mode.

### 10.3 Register Descriptions

Table 10.3 shows the register configuration. To distinguish registers in each channel, an underscore and the channel number are added as a suffix to the register name; TCR for channel 0 is expressed as TCR\_0.

**Table 10.3 Register Configuration**

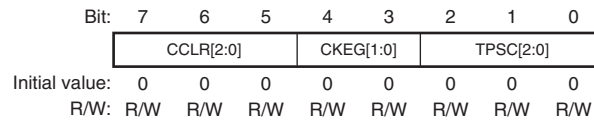
Channel	Register Name	Abbreviation	R/W	Initial value	Address	Access Size
0	Timer control register_0	TCR_0	R/W	H'00	H'FCFF0300	8
	Timer mode register_0	TMDR_0	R/W	H'00	H'FCFF0301	8
	Timer I/O control register H_0	TIORH_0	R/W	H'00	H'FCFF0302	8
	Timer I/O control register L_0	TIORL_0	R/W	H'00	H'FCFF0303	8
	Timer interrupt enable register_0	TIER_0	R/W	H'00	H'FCFF0304	8
	Timer status register_0	TSR_0	R/W	H'C0	H'FCFF0305	8
	Timer counter_0	TCNT_0	R/W	H'0000	H'FCFF0306	16
	Timer general register A_0	TGRA_0	R/W	H'FFFF	H'FCFF0308	16
	Timer general register B_0	TGRB_0	R/W	H'FFFF	H'FCFF030A	16
	Timer general register C_0	TGRC_0	R/W	H'FFFF	H'FCFF030C	16
	Timer general register D_0	TGRD_0	R/W	H'FFFF	H'FCFF030E	16
	Timer general register E_0	TGRE_0	R/W	H'FFFF	H'FCFF0320	16
	Timer general register F_0	TGRF_0	R/W	H'FFFF	H'FCFF0322	16
	Timer interrupt enable register 2_0	TIER2_0	R/W	H'00	H'FCFF0324	8
	Timer status register 2_0	TSR2_0	R/W	H'C0	H'FCFF0325	8
	Timer buffer operation transfer mode register_0	TBTM_0	R/W	H'00	H'FCFF0326	8
1	Timer control register_1	TCR_1	R/W	H'00	H'FCFF0380	8
	Timer mode register_1	TMDR_1	R/W	H'00	H'FCFF0381	8
	Timer I/O control register_1	TIOR_1	R/W	H'00	H'FCFF0382	8
	Timer interrupt enable register_1	TIER_1	R/W	H'00	H'FCFF0384	8
	Timer status register_1	TSR_1	R/W	H'C0	H'FCFF0385	8
	Timer counter_1	TCNT_1	R/W	H'0000	H'FCFF0386	16
	Timer general register A_1	TGRA_1	R/W	H'FFFF	H'FCFF0388	16
	Timer general register B_1	TGRB_1	R/W	H'FFFF	H'FCFF038A	16
	Timer input capture control register	TICCR	R/W	H'00	H'FCFF0390	8
2	Timer control register_2	TCR_2	R/W	H'00	H'FCFF0000	8
	Timer mode register_2	TMDR_2	R/W	H'00	H'FCFF0001	8
	Timer I/O control register_2	TIOR_2	R/W	H'00	H'FCFF0002	8
	Timer interrupt enable register_2	TIER_2	R/W	H'00	H'FCFF0004	8
	Timer status register_2	TSR_2	R/W	H'C0	H'FCFF0005	8
	Timer counter_2	TCNT_2	R/W	H'0000	H'FCFF0006	16
	Timer general register A_2	TGRA_2	R/W	H'FFFF	H'FCFF0008	16
	Timer general register B_2	TGRB_2	R/W	H'FFFF	H'FCFF000A	16
3	Timer control register_3	TCR_3	R/W	H'00	H'FCFF0200	8
	Timer mode register_3	TMDR_3	R/W	H'00	H'FCFF0202	8
	Timer I/O control register H_3	TIORH_3	R/W	H'00	H'FCFF0204	8
	Timer I/O control register L_3	TIORL_3	R/W	H'00	H'FCFF0205	8
	Timer interrupt enable register_3	TIER_3	R/W	H'00	H'FCFF0208	8
	Timer status register_3	TSR_3	R/W	H'C0	H'FCFF022C	8
	Timer counter_3	TCNT_3	R/W	H'0000	H'FCFF0210	16
	Timer general register A_3	TGRA_3	R/W	H'FFFF	H'FCFF0218	16
Timer general register B_3	TGRB_3	R/W	H'FFFF	H'FCFF021A	16	

Channel	Register Name	Abbreviation	R/W	Initial value	Address	Access Size	
3	Timer general register C_3	TGRC_3	R/W	H'FFFF	H'FCFF0224	16	
	Timer general register D_3	TGRD_3	R/W	H'FFFF	H'FCFF0226	16	
	Timer buffer operation transfer mode register_3	TBTM_3	R/W	H'00	H'FCFF0238	8	
4	Timer control register_4	TCR_4	R/W	H'00	H'FCFF0201	8	
	Timer mode register_4	TMDR_4	R/W	H'00	H'FCFF0203	8	
	Timer I/O control register H_4	TIORH_4	R/W	H'00	H'FCFF0206	8	
	Timer I/O control register L_4	TIORL_4	R/W	H'00	H'FCFF0207	8	
	Timer interrupt enable register_4	TIER_4	R/W	H'00	H'FCFF0209	8	
	Timer status register_4	TSR_4	R/W	H'C0	H'FCFF022D	8	
	Timer counter_4	TCNT_4	R/W	H'0000	H'FCFF0212	16	
	Timer general register A_4	TGRA_4	R/W	H'FFFF	H'FCFF021C	16	
	Timer general register B_4	TGRB_4	R/W	H'FFFF	H'FCFF021E	16	
	Timer general register C_4	TGRC_4	R/W	H'FFFF	H'FCFF0228	16	
	Timer general register D_4	TGRD_4	R/W	H'FFFF	H'FCFF022A	16	
	Timer buffer operation transfer mode register_4	TBTM_4	R/W	H'00	H'FCFF0239	8	
	Timer A/D converter start request control register	TADCR	R/W	H'0000	H'FCFF0240	16	
	Timer A/D converter start request cycle set register A_4	TADCORA_4	R/W	H'FFFF	H'FCFF0244	16	
	Timer A/D converter start request cycle set register B_4	TADCORB_4	R/W	H'FFFF	H'FCFF0246	16	
	Timer A/D converter start request cycle set buffer register A_4	TADCOBRA_4	R/W	H'FFFF	H'FCFF0248	16	
	Timer A/D converter start request cycle set buffer register B_4	TADCOBRB_4	R/W	H'FFFF	H'FCFF024A	16	
	Common	Timer start register	TSTR	R/W	H'00	H'FCFF0280	8
		Timer synchronous register	TSYR	R/W	H'00	H'FCFF0281	8
Timer read/write enable register		TRWER	R/W	H'01	H'FCFF0284	8	
Common to 3 and 4	Timer output master enable register	TOER	R/W	H'C0	H'FCFF020A	8	
	Timer output control register 1	TOCR1	R/W	H'00	H'FCFF020E	8	
	Timer output control register 2	TOCR2	R/W	H'00	H'FCFF020F	8	
	Timer gate control register	TGCR	R/W	H80	H'FCFF020D	8	
	Timer cycle data register	TCDR	R/W	H'FFFF	H'FCFF0214	16	
	Timer dead time data register	TDDR	R/W	H'FFFF	H'FCFF0216	16	
	Timer subcounter	TCNTS	R	H'0000	H'FCFF0220	16	
	Timer cycle buffer register	TCBR	R/W	H'FFFF	H'FCFF0222	16	
	Timer interrupt skipping set register	TITCR	R/W	H'00	H'FCFF0230	8	
	Timer interrupt skipping counter	TITCNT	R	H'00	H'FCFF0231	8	
	Timer buffer transfer set register	TBTER	R/W	H'00	H'FCFF0232	8	
	Timer dead time enable register	TDER	R/W	H'01	H'FCFF0234	8	
	Timer waveform control register	TWCR	R/W	H'00	H'FCFF0260	8	
	Timer output level buffer register	TOLBR	R/W	H'00	H'FCFF0236	8	



### 10.3.1 Timer Control Register (TCR)

The TCR registers are 8-bit readable/writable registers that control the TCNT operation for each channel. This module has a total of five TCR registers, one each for channels 0 to 4. TCR register settings should be conducted only when TCNT operation is stopped.



Bit	Bit Name	Initial Value	R/W	Description
7 to 5	CCLR[2:0]	000	R/W	Counter Clear 0 to 2 These bits select the TCNT counter clearing source. See Table 10.4 and Table 10.5 for details.
4, 3	CKEG[1:0]	00	R/W	Clock Edge 0 and 1 These bits select the input clock edge. When the input clock is counted using both edges, the input clock period is halved (e.g. P0φ/4 both edges = P0φ/2 rising edge). If phase counting mode is used on channels 1 and 2, this setting is ignored and the phase counting mode setting has priority. Internal clock edge selection is valid when the input clock is P0φ/4 or slower. When P0φ/1, or the overflow/underflow of another channel is selected for the input clock, although values can be written, counter operation compiles with the initial value. 00: Count at rising edge 01: Count at falling edge 1x: Count at both edges
2 to 0	TPSC[2:0]	000	R/W	Time Prescaler 0 to 2 These bits select the TCNT counter clock. The clock source can be selected independently for each channel. See Table 10.6 to Table 10.9 for details.

[Legend]

x: Don't care

**Table 10.4 CCLR0 to CCLR2 (Channels 0, 3, and 4)**

Channel	Bit 7 CCLR2	Bit 6CCLR1	Bit 5CCLR0	Description
0, 3, 4	0	0	0	TCNT clearing disabled
			1	TCNT cleared by TGRA compare match/input capture
			0	TCNT cleared by TGRB compare match/input capture
			1	TCNT cleared by counter clearing for another channel performing synchronous clearing/synchronous operation*1
1	0	1	0	TCNT clearing disabled
			1	TCNT cleared by TGRC compare match/input capture*2
			0	TCNT cleared by TGRD compare match/input capture*2
			1	TCNT cleared by counter clearing for another channel performing synchronous clearing/synchronous operation*1

Note 1. Synchronous operation is set by setting the SYNC bit in TSYR to 1.

Note 2. When TGRC or TGRD is used as a buffer register, TCNT is not cleared because the buffer register setting has priority, and compare match/input capture does not occur.

**Table 10.5 CCLR0 to CCLR2 (Channels 1 and 2)**

Channel	Bit 7 Reserved*2	Bit 6 CCLR1	Bit 5 CCLR0	Description
1, 2	0	0	0	TCNT clearing disabled
			1	TCNT cleared by TGRA compare match/input capture
		1	0	TCNT cleared by TGRB compare match/input capture
			1	TCNT cleared by counter clearing for another channel performing synchronous clearing/synchronous operation*1

Note 1. Synchronous operation is selected by setting the SYNC bit in TSYR to 1.

Note 2. Bit 7 is reserved in channels 1 and 2. It is always read as 0 and cannot be modified.

**Table 10.6 TPSC0 to TPSC2 (Channel 0)**

Channel	Bit 2 TPSC2	Bit 1 TPSC1	Bit 0 TPSC0	Description	
0	0	0	0	Internal clock: counts on P0 $\phi$ /1	
			1	Internal clock: counts on P0 $\phi$ /4	
			1	Internal clock: counts on P0 $\phi$ /16	
			1	Internal clock: counts on P0 $\phi$ /64	
	1	0	0	External clock: counts on TCLKA pin input	
			1	External clock: counts on TCLKB pin input	
			1	0	External clock: counts on TCLKC pin input
				1	External clock: counts on TCLKD pin input

**Table 10.7 TPSC0 to TPSC2 (Channel 1)**

Channel	Bit 2 TPSC2	Bit 1 TPSC1	Bit 0 TPSC0	Description	
1	0	0	0	Internal clock: counts on P0 $\phi$ /1	
			1	Internal clock: counts on P0 $\phi$ /4	
			1	Internal clock: counts on P0 $\phi$ /16	
			1	Internal clock: counts on P0 $\phi$ /64	
	1	0	0	External clock: counts on TCLKA pin input	
			1	External clock: counts on TCLKB pin input	
			1	0	Internal clock: counts on P0 $\phi$ /256
				1	Counts on TCNT_2 overflow/underflow

Note: This setting is ignored when channel 1 is in phase counting mode.

**Table 10.8 TPSC0 to TPSC2 (Channel 2)**

Channel	Bit 2 TPSC2	Bit 1 TPSC1	Bit 0 TPSC0	Description	
2	0	0	0	Internal clock: counts on P0 $\phi$ /1	
			1	Internal clock: counts on P0 $\phi$ /4	
			1	Internal clock: counts on P0 $\phi$ /16	
			1	Internal clock: counts on P0 $\phi$ /64	
	1	0	0	External clock: counts on TCLKA pin input	
			1	External clock: counts on TCLKB pin input	
			1	0	External clock: counts on TCLKC pin input
				1	Internal clock: counts on P0 $\phi$ /1024

Note: This setting is ignored when channel 2 is in phase counting mode.

**Table 10.9 TPSC0 to TPSC2 (Channels 3 and 4)**

Channel	Bit 2 TPSC2	Bit 1 TPSC1	Bit 0 TPSC0	Description
3, 4	0	0	0	Internal clock: counts on P0 $\phi$ /1
			1	Internal clock: counts on P0 $\phi$ /4
	1	1	0	Internal clock: counts on P0 $\phi$ /16
			1	Internal clock: counts on P0 $\phi$ /64
1	0	0	0	Internal clock: counts on P0 $\phi$ /256
			1	Internal clock: counts on P0 $\phi$ /1024
	1	1	0	External clock: counts on TCLKA pin input
			1	External clock: counts on TCLKB pin input

### 10.3.2 Timer Mode Register (TMDR)

The TMDR registers are 8-bit readable/writable registers that are used to set the operating mode of each channel. This module has five TMDR registers, one each for channels 0 to 4. TMDR register settings should be changed only when TCNT operation is stopped.

Bit:	7	6	5	4	3	2	1	0
	-	BFE	BFB	BFA	MD[3:0]			
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	BFE	0	R/W	Buffer Operation E Specifies whether TGRE_0 and TGRF_0 are to operate in the normal way or to be used together for buffer operation. TGRF compare match is generated when TGRF is used as the buffer register. In channels 1 to 4, this bit is reserved. It is always read as 0 and the write value should always be 0. 0: TGRE_0 and TGRF_0 operate normally 1: TGRE_0 and TGRF_0 used together for buffer operation
5	BFB	0	R/W	Buffer Operation B Specifies whether TGRB is to operate in the normal way, or TGRB and TGRD are to be used together for buffer operation. When TGRD is used as a buffer register, TGRD input capture/output compare is not generated in a mode other than complementary PWM. In channels 1 and 2, which have no TGRD, bit 5 is reserved. It is always read as 0 and cannot be modified. 0: TGRB and TGRD operate normally 1: TGRB and TGRD used together for buffer operation
4	BFA	0	R/W	Buffer Operation A Specifies whether TGRA is to operate in the normal way, or TGRA and TGRC are to be used together for buffer operation. When TGRC is used as a buffer register, TGRC input capture/output compare is not generated in a mode other than complementary PWM. TGRC compare match is generated when in complementary PWM mode. When compare match for channel 4 occurs during the Tb period in complementary PWM mode, TGFC is set. Therefore, set the TGIEC bit in the timer interrupt enable register 4 (TIER_4) to 0. In channels 1 and 2, which have no TGRC, bit 4 is reserved. It is always read as 0 and cannot be modified. 0: TGRA and TGRC operate normally 1: TGRA and TGRC used together for buffer operation
3 to 0	MD[3:0]	0000	R/W	Modes 0 to 3 These bits are used to set the timer operating mode. See Table 10.10 for details.

**Table 10.10 Setting of Operation Mode by Bits MD0 to MD3**

Bit 3 MD3	Bit 2 MD2	Bit 1 MD1	Bit 0 MD0	Description
0	0	0	0	Normal operation
			1	Setting prohibited
		1	0	PWM mode 1
			1	PWM mode 2*1
	1	0	0	Phase counting mode 1*2
			1	Phase counting mode 2*2
		1	0	Phase counting mode 3*2
			1	Phase counting mode 4*2
1	0	0	0	Reset synchronous PWM mode*3
			1	Setting prohibited
		1	X	Setting prohibited
			1	0
	1	Complementary PWM mode 1 (transmit at crest)*3		
	1	0		Complementary PWM mode 2 (transmit at trough)*3
		1		Complementary PWM mode 2 (transmit at crest and trough)*3

## [Legend]

X: Don't care

Note 1. Notes:1.PWM mode 2 cannot be set for channels 3 and 4.

Note 2. Phase counting mode cannot be set for channels 0, 3, and 4.

Note 3. Reset synchronous PWM mode, complementary PWM mode can only be set for channel 3. When channel 3 is set to reset synchronous PWM mode or complementary PWM mode, the channel 4 settings become ineffective and automatically conform to the channel 3 settings. However, do not set channel 4 to reset synchronous PWM mode or complementary PWM mode. Reset synchronous PWM mode and complementary PWM mode cannot be set for channels 0, 1, and 2.

### 10.3.3 Timer I/O Control Register (TIOR)

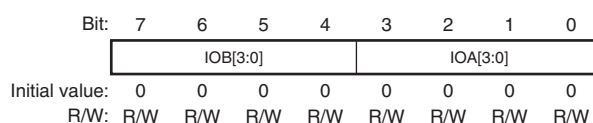
The TIOR registers are 8-bit readable/writable registers that control the TGR registers. This module has a total of eight TIOR registers, two each for channels 0, 3, and 4, one each for channels 1 and 2.

TIOR should be set while TMDR is set in normal operation, PWM mode, or phase counting mode.

The initial output specified by TIOR is valid when the counter is stopped (the CST bit in TSTR is cleared to 0). Note also that, in PWM mode 2, the output at the point at which the counter is cleared to 0 is specified.

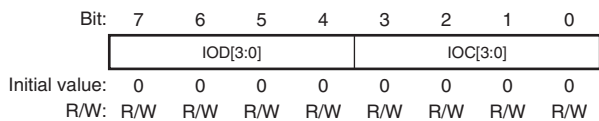
When TGRC or TGRD is designated for buffer operation, this setting is invalid and the register operates as a buffer register.

- TIORH\_0, TIOR\_1, TIOR\_2, TIORH\_3, TIORH\_4



Bit	Bit Name	Initial Value	R/W	Description
7 to 4	IOB[3:0]	0000	R/W	I/O Control B0 to B3 Specify the function of TGRB. See the following tables. TIORH_0: Table 10.11 TIOR_1: Table 10.13 TIOR_2: Table 10.14 TIORH_3: Table 10.15 TIORH_4: Table 10.17
3 to 0	IOA[3:0]	0000	R/W	I/O Control A0 to A3 Specify the function of TGRA. See the following tables. TIORH_0: Table 10.19 TIOR_1: Table 10.21 TIOR_2: Table 10.22 TIORH_3: Table 10.23 TIORH_4: Table 10.25

- TIORL\_0, TIORL\_3, TIORL\_4



Bit	Bit Name	Initial Value	R/W	Description
7 to 4	IOD[3:0]	0000	R/W	I/O Control D0 to D3 Specify the function of TGRD. See the following tables. TIORL_0: Table 10.12 TIORL_3: Table 10.16 TIORL_4: Table 10.18
3 to 0	IOC[3:0]	0000	R/W	I/O Control C0 to C3 Specify the function of TGRC. See the following tables. TIORL_0: Table 10.20 TIORL_3: Table 10.24 TIORL_4: Table 10.26

Table 10.11 TIORH\_0 (Channel 0)

Bit 7 IOB3	Bit 6 IOB2	Bit 5 IOB1	Bit 4 IOB0	Description	
				TGRB_0 Function	TIOC0B Pin Function
0	0	0	0	Output compare register	Output retained*
			1		Initial output is 0 0 output at compare match
			0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
			0		Output retained
			1		Initial output is 1 0 output at compare match
			0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
			0		Input capture at rising edge
			1		Input capture at falling edge
			X		Input capture at both edges
			X		Capture input source is channel 1/count clock Input capture at TCNT_1 count-up/count-down

[Legend]

X: Don't care

Note: \* After power-on reset, 0 is output until TIOR is set.



Table 10.12 TIORL\_0 (Channel 0)

Bit 7 IOD3	Bit 6 IOD2	Bit 5 IOD1	Bit 4 IOD0	Description	
				TGRD_0 Function	TIOC0D Pin Function
0	0	0	0	Output compare register*2	Output retained*1
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
		1	0		Output retained
			1		Initial output is 1 0 output at compare match
	1	0	0	Input capture register*2	Input capture at rising edge
			1		Input capture at falling edge
		1	Input capture at both edges		
		X	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
		1	X		Capture input source is channel 1/count clock Input capture at TCNT_1 count-up/count-down

[Legend]

X: Don't care

Note 1. After power-on reset, 0 is output until TIOR is set.

Note 2. When the BFB bit in TMDR\_0 is set to 1 and TGRD\_0 is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

**Table 10.13 TIOR\_1 (Channel 1)**

Bit 7 IOB3	Bit 6 IOB2	Bit 5 IOB1	Bit 4 IOB0	Description	
				TGRB_1 Function	TIOC1B Pin Function
0	0	0	0	Output compare register	Output retained*
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
		1	0		Output retained
			1		Initial output is 1 0 output at compare match
	1	0	Initial output is 1 1 output at compare match		
		1	Initial output is 1 Toggle output at compare match		
	1	0	0	Input capture register	Input capture at rising edge
			1		Input capture at falling edge
			X		Input capture at both edges
		1	X	X	Input capture at generation of TGRC_0 compare match/input capture

[Legend]

X: Don't care

Note: \* After power-on reset, 0 is output until TIOR is set.

Table 10.14 TIOR\_2 (Channel 2)

Bit 7 IOB3	Bit 6 IOB2	Bit 5 IOB1	Bit 4 IOB0	Description	
				TGRB_2 Function	TIOC2B Pin Function
0	0	0	0	Output compare register	Output retained*
			1		Initial output is 0 0 output at compare match
			0		Initial output is 0 1 output at compare match
		1	0		Initial output is 0 Toggle output at compare match
			1		Output retained
			0		Initial output is 1 0 output at compare match
	1	0	0	Input capture register	Initial output is 1 1 output at compare match
			1		Initial output is 1 1 output at compare match
			0		Initial output is 1 Toggle output at compare match
		1	0		Input capture at rising edge
			1		Input capture at falling edge
			X		Input capture at both edges

[Legend]

X: Don't care

Note: \* After power-on reset, 0 is output until TIOR is set.

Table 10.15 TIORH\_3 (Channel 3)

Bit 7 IOB3	Bit 6 IOB2	Bit 5 IOB1	Bit 4 IOB0	Description		
				TGRB_3 Function	TIOC3B Pin Function	
0	0	0	0	Output compare register	Output retained*	
			1		Initial output is 0 0 output at compare match	
		1	0		Initial output is 0 1 output at compare match	
			1		Initial output is 0 Toggle output at compare match	
		1	0		Output retained	
			1		Initial output is 1 0 output at compare match	
	1	0	0	Input capture register	Input capture at rising edge	
			1		Input capture at falling edge	
		1	0		Initial output is 1 1 output at compare match	
			1		Initial output is 1 Toggle output at compare match	
		X	0		0	Input capture at rising edge
					1	Input capture at falling edge
1	X	0	Input capture at rising edge			
		1	Input capture at falling edge			

[Legend]

X: Don't care

Note: \* After power-on reset, 0 is output until TIOR is set.

**Table 10.16 TIORL\_3 (Channel 3)**

Bit 7 IOD3	Bit 6 IOD2	Bit 5 IOD1	Bit 4 IOD0	Description	
				TGRD_3 Function	TIOC3D Pin Function
0	0	0	0	Output compare register*2	Output retained*1
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	X	0	0	Input capture register*2	Input capture at rising edge
			1	Input capture at falling edge	
		1	X	Input capture at both edges	

[Legend]

X: Don't care

Note 1. After power-on reset, 0 is output until TIOR is set.

Note 2. When the BFB bit in TMDR\_3 is set to 1 and TGRD\_3 is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 10.17 TIORH\_4 (Channel 4)

Bit 7 IOB3	Bit 6 IOB2	Bit 5 IOB1	Bit 4 IOB0	Description		
				TGRB_4 Function	TIOC4B Pin Function	
0	0	0	0	Output compare register	Output retained*	
			1		Initial output is 0 0 output at compare match	
		1	0		Initial output is 0 1 output at compare match	
			1		Initial output is 0 Toggle output at compare match	
		1	0		Output retained	
			1		Initial output is 1 0 output at compare match	
	1	0	0	Input capture register	Input capture at rising edge	
			1		Input capture at falling edge	
		1	0		Initial output is 1 1 output at compare match	
			1		Initial output is 1 Toggle output at compare match	
		1	X		0	Input capture at rising edge
					1	Input capture at falling edge
1	X	X	Input capture at both edges			

[Legend]

X: Don't care

Note: \* After power-on reset, 0 is output until TIOR is set.

Table 10.18 TIORL\_4 (Channel 4)

Bit 7 IOD3	Bit 6 IOD2	Bit 5 IOD1	Bit 4 IOD0	Description	
				TGRD_4 Function	TIOC4D Pin Function
0	0	0	0	Output compare register*2	Output retained*1
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
		1	Initial output is 0 Toggle output at compare match		
	1	0	0	Output compare register*2	Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
		1	Initial output is 1 Toggle output at compare match		
1	X	0	0	Input capture register*2	Input capture at rising edge
			1		Input capture at falling edge
		1	X		Input capture at both edges

[Legend]

X: Don't care

Note 1. After power-on reset, 0 is output until TIOR is set.

Note 2. When the BFB bit in TMDR\_4 is set to 1 and TGRD\_4 is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

**Table 10.19** TIORH\_0 (Channel 0)

Bit 3 IOA3	Bit 2 IOA2	Bit 1 IOA1	Bit 0 IOA0	Description	
				TGRA_0 Function	TIOC0A Pin Function
0	0	0	0	Output compare register	Output retained*
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0	Output retained	
			1	Initial output is 1 0 output at compare match	
		1	0	Initial output is 1 1 output at compare match	
			1	Initial output is 1 Toggle output at compare match	
1	0	0	0	Input capture register	Input capture at rising edge
			1	Input capture at falling edge	
		1	X	Input capture at both edges	
			X	Capture input source is channel 1/count clock Input capture at TCNT_1 count-up/count-down	

[Legend]

X: Don't care

Note: \* After power-on reset, 0 is output until TIOR is set.



Table 10.20 TIORL\_0 (Channel 0)

Bit 3 IOC3	Bit 2 IOC2	Bit 1 IOC1	Bit 0 IOC0	Description	
				TGRC_0 Function	TIOC0C Pin Function
0	0	0	0	Output compare register*2	Output retained*1
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0	Output retained	
			1	Initial output is 1 0 output at compare match	
		1	0	Initial output is 1 1 output at compare match	
			1	Initial output is 1 Toggle output at compare match	
1	0	0	0	Input capture register*2	Input capture at rising edge
			1	Input capture at falling edge	
		1	X	Input capture at both edges	
			X	Capture input source is channel 1/count clock Input capture at TCNT_1 count-up/count-down	

[Legend]

X: Don't care

Note 1. After power-on reset, 0 is output until TIOR is set.

Note 2. When the BFA bit in TMDR\_0 is set to 1 and TGRC\_0 is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

**Table 10.21** TIOR\_1 (Channel 1)

Bit 3 IOA3	Bit 2 IOA2	Bit 1 IOA1	Bit 0 IOA0	Description	
				TGRA_1 Function	TIOC1A Pin Function
0	0	0	0	Output compare register	Output retained*
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	0	0	Input capture register	Input capture at rising edge	
		1		Input capture at falling edge	
	1	X		Input capture at both edges	
		X		Input capture at generation of channel 0/TGRA_0 compare match/input capture	

[Legend]

X: Don't care

Note: \* After power-on reset, 0 is output until TIOR is set.

Table 10.22 TIOR\_2 (Channel 2)

Bit 3 IOA3	Bit 2 IOA2	Bit 1 IOA1	Bit 0 IOA0	Description	
				TGRA_2 Function	TIOC2A Pin Function
0	0	0	0	Output compare register	Output retained*
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
		1	0		Output retained
			1		Initial output is 1 0 output at compare match
	1	0	Initial output is 1 1 output at compare match		
		1	Initial output is 1 Toggle output at compare match		
	1	X	0	Input capture register	Input capture at rising edge
			1		Input capture at falling edge
			X		Input capture at both edges

[Legend]

X: Don't care

Note: \* After power-on reset, 0 is output until TIOR is set.

Table 10.23 TIORH\_3 (Channel 3)

Bit 3 IOA3	Bit 2 IOA2	Bit 1 IOA1	Bit 0 IOA0	Description	
				TGRA_3 Function	TIOC3A Pin Function
0	0	0	0	Output compare register	Output retained*
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
		1	0		Output retained
			1		Initial output is 1 0 output at compare match
	1	0	Initial output is 1 1 output at compare match		
		1	Initial output is 1 Toggle output at compare match		
	1	X	0	Input capture register	Input capture at rising edge
			1		Input capture at falling edge
			X		Input capture at both edges

[Legend]

X: Don't care

Note: \* After power-on reset, 0 is output until TIOR is set.

Table 10.24 TIORL\_3 (Channel 3)

Bit 3 IOC3	Bit 2 IOC2	Bit 1 IOC1	Bit 0 IOC0	Description	
				TGRC_3 Function	TIOC3C Pin Function
0	0	0	0	Output compare register*2	Output retained*1
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0	Input capture register*2	Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	X	0	Input capture register*2	Input capture at rising edge	
		1		Input capture at falling edge	
		1		Input capture at both edges	

[Legend]

X: Don't care

Note 1. After power-on reset, 0 is output until TIOR is set.

Note 2. When the BFA bit in TMDR\_3 is set to 1 and TGRC\_3 is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

Table 10.25 TIORH\_4 (Channel 4)

Bit 3 IOA3	Bit 2 IOA2	Bit 1 IOA1	Bit 0 IOA0	Description	
				TGRA_4 Function	TIOC4A Pin Function
0	0	0	0	Output compare register	Output retained*
			1		Initial output is 0 0 output at compare match
		1	0		Initial output is 0 1 output at compare match
			1		Initial output is 0 Toggle output at compare match
	1	0	0		Output retained
			1		Initial output is 1 0 output at compare match
		1	0		Initial output is 1 1 output at compare match
			1		Initial output is 1 Toggle output at compare match
1	X	0	Input capture register	Input capture at rising edge	
		1		Input capture at falling edge	
		1		Input capture at both edges	

[Legend]

X: Don't care

Note: \* After power-on reset, 0 is output until TIOR is set.

Table 10.26 TIORL\_4 (Channel 4)

Bit 3 IOC3	Bit 2 IOC2	Bit 1 IOC1	Bit 0 IOC0	Description		
				TGRC_4 Function	TIOC4C Pin Function	
0	0	0	0	Output compare register*2	Output retained*1	
			1		Initial output is 0 0 output at compare match	
		1	0		Initial output is 0 1 output at compare match	
			1		Initial output is 0 Toggle output at compare match	
		1	0		Output retained	
			1		Initial output is 1 0 output at compare match	
	1	0	Initial output is 1 1 output at compare match			
		1	Initial output is 1 Toggle output at compare match			
	1	X	0	0	Input capture register*2	Input capture at rising edge
				1		Input capture at falling edge
			1	Input capture at both edges		
		1	X	X		X

[Legend]

X: Don't care

Note 1. After power-on reset, 0 is output until TIOR is set.

Note 2. When the BFA bit in TMDR\_4 is set to 1 and TGRC\_4 is used as a buffer register, this setting is invalid and input capture/output compare is not generated.

### 10.3.4 Timer Interrupt Enable Register (TIER)

The TIER registers are 8-bit readable/writable registers that control enabling or disabling of interrupt requests for each channel. This module has six TIER registers, two for channel 0 and one each for channels 1 to 4.

- TIER\_0, TIER\_1, TIER\_2, TIER\_3, TIER\_4

Bit:	7	6	5	4	3	2	1	0
	TTGE	TTGE2	TCIEU	TCIEV	TGIED	TGIEC	TGIEB	TGIEA
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	TTGE	0	R/W	A/D Converter Start Request Enable Enables or disables generation of A/D converter start requests by TGRA input capture/compare match. 0: A/D converter start request generation disabled 1: A/D converter start request generation enabled
6	TTGE2	0	R/W	A/D Converter Start Request Enable 2 Enables or disables generation of A/D converter start requests by TCNT_4 underflow (trough) in complementary PWM mode. In channels 0 to 3, bit 6 is reserved. It is always read as 0 and the write value should always be 0. 0: A/D converter start request generation by TCNT_4 underflow (trough) disabled 1: A/D converter start request generation by TCNT_4 underflow (trough) enabled
5	TCIEU	0	R/W	Underflow Interrupt Enable Enables or disables interrupt requests (TCIU) by the TCFU flag when the TCFU flag in TSR is set to 1 in channels 1 and 2. In channels 0, 3, and 4, bit 5 is reserved. It is always read as 0 and the write value should always be 0. 0: Interrupt requests (TCIU) by TCFU disabled 1: Interrupt requests (TCIU) by TCFU enabled
4	TCIEV	0	R/W	Overflow Interrupt Enable Enables or disables interrupt requests (TCIV) by the TCFV flag when the TCFV flag in TSR is set to 1. 0: Interrupt requests (TCIV) by TCFV disabled 1: Interrupt requests (TCIV) by TCFV enabled
3	TGIED	0	R/W	TGR Interrupt Enable D Enables or disables interrupt requests (TGID) by the TGFD bit when the TGFD bit in TSR is set to 1 in channels 0, 3, and 4. In channels 1 and 2, bit 3 is reserved. It is always read as 0 and the write value should always be 0. 0: Interrupt requests (TGID) by TGFD bit disabled 1: Interrupt requests (TGID) by TGFD bit enabled
2	TGIEC	0	R/W	TGR Interrupt Enable C Enables or disables interrupt requests (TGIC) by the TGFC bit when the TGFC bit in TSR is set to 1 in channels 0, 3, and 4. In channels 1 and 2, bit 2 is reserved. It is always read as 0 and the write value should always be 0. 0: Interrupt requests (TGIC) by TGFC bit disabled 1: Interrupt requests (TGIC) by TGFC bit enabled
1	TGIEB	0	R/W	TGR Interrupt Enable B Enables or disables interrupt requests (TGIB) by the TGFB bit when the TGFB bit in TSR is set to 1. 0: Interrupt requests (TGIB) by TGFB bit disabled 1: Interrupt requests (TGIB) by TGFB bit enabled



Bit	Bit Name	Initial Value	R/W	Description
0	TGIEA	0	R/W	TGR Interrupt Enable A Enables or disables interrupt requests (TGIA) by the TGFA bit when the TGFA bit in TSR is set to 1. 0: Interrupt requests (TGIA) by TGFA bit disabled 1: Interrupt requests (TGIA) by TGFA bit enabled

- TIER2\_0

Bit:	7	6	5	4	3	2	1	0
	TTGE2	-	-	-	-	-	TGIEF	TGIEE
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	TTGE2	0	R/W	A/D Converter Start Request Enable 2 Enables or disables generation of A/D converter start requests by compare match between TCNT_0 and TGRE_0. 0: A/D converter start request generation by compare match between TCNT_0 and TGRE_0 disabled 1: A/D converter start request generation by compare match between TCNT_0 and TGRE_0 enabled
6 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	TGIEF	0	R/W	TGR Interrupt Enable F Enables or disables interrupt requests by compare match between TCNT_0 and TGRF_0. 0: Interrupt requests (TGIF) by TGFE bit disabled 1: Interrupt requests (TGIF) by TGFE bit enabled
0	TGIEE	0	R/W	TGR Interrupt Enable E Enables or disables interrupt requests by compare match between TCNT_0 and TGRE_0. 0: Interrupt requests (TGIE) by TGEE bit disabled 1: Interrupt requests (TGIE) by TGEE bit enabled

### 10.3.5 Timer Status Register (TSR)

The TSR registers are 8-bit readable/writable registers that indicate the status of each channel. This module has six TSR registers, two for channel 0 and one each for channels 1 to 4.

- TSR\_0, TSR\_1, TSR\_2, TSR\_3, TSR\_4

Bit:	7	6	5	4	3	2	1	0
	TCFD	-	TCFU	TCFV	TGFD	TGFC	TGFB	TGFA
Initial value:	1	1	0	0	0	0	0	0
R/W:	R	R	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1

Note: 1. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

Bit	Bit Name	Initial Value	R/W	Description
7	TCFD	1	R	Count Direction Flag Status flag that shows the direction in which TCNT counts in channels 1 to 4. In channel 0, bit 7 is reserved. It is always read as 1 and the write value should always be 1. 0: TCNT counts down 1: TCNT counts up
6	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
5	TCFU	0	R/(W)*1	Underflow Flag Status flag that indicates that TCNT underflow has occurred when channels 1 and 2 are set to phase counting mode. Only 0 can be written, for flag clearing. In channels 0, 3, and 4, bit 5 is reserved. It is always read as 0 and the write value should always be 0. [Clearing condition] • When 0 is written to TCFU after reading TCFU = 1*2 [Setting condition] • When the TCNT value underflows (changes from H'0000 to H'FFFF)
4	TCFV	0	R/(W)*1	Overflow Flag Status flag that indicates that TCNT overflow has occurred. Only 0 can be written, for flag clearing. [Clearing condition] • When 0 is written to TCFV after reading TCFV = 1*2 [Setting condition] • When the TCNT value overflows (changes from H'FFFF to H'0000) In channel 4, when the TCNT_4 value underflows (changes from H'0001 to H'0000) in complementary PWM mode, this flag is also set.
3	TGFD	0	R/(W)*1	Input Capture/Output Compare Flag D Status flag that indicates the occurrence of TGRD input capture or compare match in channels 0, 3, and 4. Only 0 can be written, for flag clearing. In channels 1 and 2, bit 3 is reserved. It is always read as 0 and the write value should always be 0. [Clearing condition] • When 0 is written to TGFD after reading TGFD = 1*2 [Setting conditions] • When TCNT = TGRD and TGRD is functioning as output compare register • When TCNT value is transferred to TGRD by input capture signal and TGRD is functioning as input capture register

Bit	Bit Name	Initial Value	R/W	Description
2	TGFC	0	R/(W)*1	<p>Input Capture/Output Compare Flag C</p> <p>Status flag that indicates the occurrence of TGRC input capture or compare match in channels 0, 3, and 4. Only 0 can be written, for flag clearing. In channels 1 and 2, bit 2 is reserved. It is always read as 0 and the write value should always be 0.</p> <p>[Clearing condition]</p> <ul style="list-style-type: none"> <li>When 0 is written to TGFC after reading</li> </ul> <p>TGFC = 1*2</p> <p>[Setting conditions]</p> <ul style="list-style-type: none"> <li>When TCNT = TGRC and TGRC is functioning as output compare register</li> <li>When TCNT value is transferred to TGRC by input capture signal and TGRC is functioning as input capture register</li> </ul>
1	TGFB	0	R/(W)*1	<p>Input Capture/Output Compare Flag B</p> <p>Status flag that indicates the occurrence of TGRB input capture or compare match. Only 0 can be written, for flag clearing.</p> <p>[Clearing condition]</p> <ul style="list-style-type: none"> <li>When 0 is written to TGFB after reading</li> </ul> <p>TGFB = 1*2</p> <p>[Setting conditions]</p> <ul style="list-style-type: none"> <li>When TCNT = TGRB and TGRB is functioning as output compare register</li> <li>When TCNT value is transferred to TGRB by input capture signal and TGRB is functioning as input capture register</li> </ul>
0	TGFA	0	R/(W)*1	<p>Input Capture/Output Compare Flag A</p> <p>Status flag that indicates the occurrence of TGRA input capture or compare match. Only 0 can be written, for flag clearing.</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> <li>When the direct memory access controller is activated by TGIA interrupt</li> <li>When 0 is written to TGFA after reading</li> </ul> <p>TGFA = 1*2</p> <p>[Setting conditions]</p> <ul style="list-style-type: none"> <li>When TCNT = TGRA and TGRA is functioning as output compare register</li> <li>When TCNT value is transferred to TGRA by input capture signal and TGRA is functioning as input capture register</li> </ul>

Note 1. Writing 0 to this bit after reading it as 1 clears the flag.

Note 2. If the next flag is set before TGFA is cleared to 0 after reading TGFA = 1, TGFA remains 1 even when 0 is written to. In this case, read TGFA = 1 again to clear TGFA to 0.

- TSR2\_0

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	TGFF	TGFE
Initial value:	1	1	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/(W)*1	R/(W)*1

Note: 1. Writing 0 to this bit after reading it as 1 clears the flag and is the only allowed way.

Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
5 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	TGFF	0	R/(W)*1	Compare Match Flag F Status flag that indicates the occurrence of compare match between TCNT_0 and TGRF_0. [Clearing condition] <ul style="list-style-type: none"> <li>• When 0 is written to TGFF after reading TGFF = 1*2</li> </ul> [Setting condition] <ul style="list-style-type: none"> <li>• When TCNT_0 = TGRF_0 and TGRF_0 is functioning as compare register</li> </ul>
0	TGFE	0	R/(W)*1	Compare Match Flag E Status flag that indicates the occurrence of compare match between TCNT_0 and TGRE_0. [Clearing condition] <ul style="list-style-type: none"> <li>• When 0 is written to TGFE after reading TGFE = 1*2</li> </ul> [Setting condition] <ul style="list-style-type: none"> <li>• When TCNT_0 = TGRE_0 and TGRE_0 is functioning as compare register</li> </ul>

Note 1. Writing 0 to this bit after reading it as 1 clears the flag.

Note 2. If the next flag is set before TGFA is cleared to 0 after reading TGFA = 1, TGFA remains 1 even when 0 is written to. In this case, read TGFA = 1 again to clear TGFA to 0.

### 10.3.6 Timer Buffer Operation Transfer Mode Register (TBTM)

The TBTM registers are 8-bit readable/writable registers that specify the timing for transferring data from the buffer register to the timer general register in PWM mode. This module has three TBTM registers, one each for channels 0, 3, and 4.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	TTSE	TTSB	TTSA
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	TTSE	0	R/W	Timing Select E Specifies the timing for transferring data from TGRF_0 to TGRE_0 when they are used together for buffer operation. In channels 3 and 4, bit 2 is reserved. It is always read as 0 and the write value should always be 0. 0: When compare match E occurs in channel 0 1: When TCNT_0 is cleared
1	TTSB	0	R/W	Timing Select B Specifies the timing for transferring data from TGRD to TGRB in each channel when they are used together for buffer operation. 0: When compare match B occurs in each channel 1: When TCNT is cleared in each channel
0	TTSA	0	R/W	Timing Select A Specifies the timing for transferring data from TGRC to TGRA in each channel when they are used together for buffer operation. 0: When compare match A occurs in each channel 1: When TCNT is cleared in each channel

### 10.3.7 Timer Input Capture Control Register (TICCR)

TICCR is an 8-bit readable/writable register that specifies input capture conditions when TCNT\_1 and TCNT\_2 are cascaded. This module has one TICCR in channel 1.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	I2BE	I2AE	I1BE	I1AE
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	I2BE	0	R/W	Input Capture Enable Specifies whether to include the TIOC2B pin in the TGRB_1 input capture conditions. 0: Does not include the TIOC2B pin in the TGRB_1 input capture conditions 1: Includes the TIOC2B pin in the TGRB_1 input capture conditions
2	I2AE	0	R/W	Input Capture Enable Specifies whether to include the TIOC2A pin in the TGRA_1 input capture conditions. 0: Does not include the TIOC2A pin in the TGRA_1 input capture conditions 1: Includes the TIOC2A pin in the TGRA_1 input capture conditions
1	I1BE	0	R/W	Input Capture Enable Specifies whether to include the TIOC1B pin in the TGRB_2 input capture conditions. 0: Does not include the TIOC1B pin in the TGRB_2 input capture conditions 1: Includes the TIOC1B pin in the TGRB_2 input capture conditions
0	I1AE	0	R/W	Input Capture Enable Specifies whether to include the TIOC1A pin in the TGRA_2 input capture conditions. 0: Does not include the TIOC1A pin in the TGRA_2 input capture conditions 1: Includes the TIOC1A pin in the TGRA_2 input capture conditions

### 10.3.8 Timer A/D Converter Start Request Control Register (TADCR)

TADCR is a 16-bit readable/writable register that enables or disables A/D converter start requests and specifies whether to link A/D converter start requests with interrupt skipping operation. This module has one TADCR in channel 4.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BF[1:0]		-	-	-	-	-	-	UT4AE	DT4AE	UT4BE	DT4BE	ITA3AE	ITA4VE	ITB3AE	ITB4VE
Initial value:	0	0	0	0	0	0	0	0	0	0*	0	0*	0*	0*	0*	0*
R/W:	R/W	R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: \* Do not set to 1 when complementary PWM mode is not selected.

Bit	Bit Name	Initial Value	R/W	Description
15, 14	BF[1:0]	00	R/W	TADCOBRA_4/TADCOBRB_4 Transfer Timing Select Select the timing for transferring data from TADCOBRA_4 and TADCOBRB_4 to TADCORA_4 and TADCORB_4. For details, see Table 10.27.
13 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	UT4AE	0	R/W	Up-Count TRG4AN Enable Enables or disables A/D converter start requests (TRG4AN) during TCNT_4 up-count operation. 0: A/D converter start requests (TRG4AN) disabled during TCNT_4 up-count operation 1: A/D converter start requests (TRG4AN) enabled during TCNT_4 up-count operation
6	DT4AE	0*	R/W	Down-Count TRG4AN Enable Enables or disables A/D converter start requests (TRG4AN) during TCNT_4 down-count operation. 0: A/D converter start requests (TRG4AN) disabled during TCNT_4 down-count operation 1: A/D converter start requests (TRG4AN) enabled during TCNT_4 down-count operation
5	UT4BE	0	R/W	Up-Count TRG4BN Enable Enables or disables A/D converter start requests (TRG4BN) during TCNT_4 up-count operation. 0: A/D converter start requests (TRG4BN) disabled during TCNT_4 up-count operation 1: A/D converter start requests (TRG4BN) enabled during TCNT_4 up-count operation
4	DT4BE	0*	R/W	Down-Count TRG4BN Enable Enables or disables A/D converter start requests (TRG4BN) during TCNT_4 down-count operation. 0: A/D converter start requests (TRG4BN) disabled during TCNT_4 down-count operation 1: A/D converter start requests (TRG4BN) enabled during TCNT_4 down-count operation
3	ITA3AE	0*	R/W	TGIA_3 Interrupt Skipping Link Enable Select whether to link A/D converter start requests (TRG4AN) with TGIA_3 interrupt skipping operation. 0: Does not link with TGIA_3 interrupt skipping 1: Links with TGIA_3 interrupt skipping
2	ITA4VE	0*	R/W	TCIV_4 Interrupt Skipping Link Enable Select whether to link A/D converter start requests (TRG4AN) with TCIV_4 interrupt skipping operation. 0: Does not link with TCIV_4 interrupt skipping 1: Links with TCIV_4 interrupt skipping
1	ITB3AE	0*	R/W	TGIA_3 Interrupt Skipping Link Enable Select whether to link A/D converter start requests (TRG4BN) with TGIA_3 interrupt skipping operation. 0: Does not link with TGIA_3 interrupt skipping 1: Links with TGIA_3 interrupt skipping

Bit	Bit Name	Initial Value	R/W	Description
0	ITB4VE	0*	R/W	TCIV_4 Interrupt Skipping Link Enable Select whether to link A/D converter start requests (TRG4BN) with TCIV_4 interrupt skipping operation. 0: Does not link with TCIV_4 interrupt skipping 1: Links with TCIV_4 interrupt skipping

Note 1. TADCR must not be accessed in eight bits; it should always be accessed in 16 bits.

Note 2. When interrupt skipping is disabled (the T3AEN and T4VEN bits in the timer interrupt skipping set register (TITCR) are cleared to 0 or the skipping count set bits (3ACOR and 4VCOR) in TITCR are cleared to 0), do not link A/D converter start requests with interrupt skipping operation (clear the ITA3AE, ITA4VE, ITB3AE, and ITB4VE bits in the timer A/D converter start request control register (TADCR) to 0).

Note 3. If link with interrupt skipping is enabled while interrupt skipping is disabled, A/D converter start requests will not be issued.

Note: \* Do not set to 1 when complementary PWM mode is not selected.

**Table 10.27 Setting of Transfer Timing by Bits BF1 and BF0**

Bit 7	Bit 6	Description
BF1	BF0	
0	0	Does not transfer data from the cycle set buffer register to the cycle set register.
0	1	Transfers data from the cycle set buffer register to the cycle set register at the crest of the TCNT_4 count.*1
1	0	Transfers data from the cycle set buffer register to the cycle set register at the trough of the TCNT_4 count.*2
1	1	Transfers data from the cycle set buffer register to the cycle set register at the crest and trough of the TCNT_4 count.*2

Note 1. Data is transferred from the cycle set buffer register to the cycle set register when the crest of the TCNT\_4 count is reached in complementary PWM mode, when compare match occurs between TCNT\_3 and TGRA\_3 in reset-synchronized PWM mode, or when compare match occurs between TCNT\_4 and TGRA\_4 in PWM mode 1 or normal operation mode.

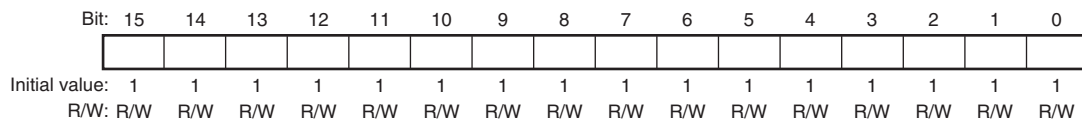
Note 2. These settings are prohibited when complementary PWM mode is not selected.



### 10.3.9 Timer A/D Converter Start Request Cycle Set Registers (TADCORA\_4 and TADCORB\_4)

TADCORA\_4 and TADCORB\_4 are 16-bit readable/writable registers. When the TCNT\_4 count reaches the value in TADCORA\_4 or TADCORB\_4, a corresponding A/D converter start request will be issued.

TADCORA\_4 and TADCORB\_4 are initialized to H'FFFF.

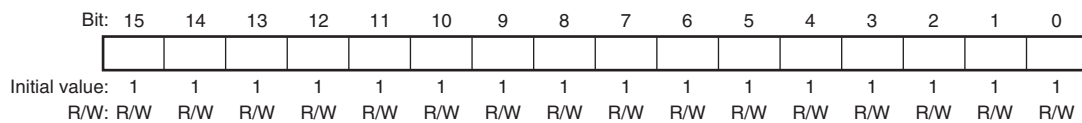


Note: TADCORA\_4 and TADCORB\_4 must not be accessed in eight bits; they should always be accessed in 16 bits.

### 10.3.10 Timer A/D Converter Start Request Cycle Set Buffer Registers (TADCOBRA\_4 and TADCOBRB\_4)

TADCOBRA\_4 and TADCOBRB\_4 are 16-bit readable/writable registers. When the crest or trough of the TCNT\_4 count is reached, these register values are transferred to TADCORA\_4 and TADCORB\_4, respectively.

TADCOBRA\_4 and TADCOBRB\_4 are initialized to H'FFFF.

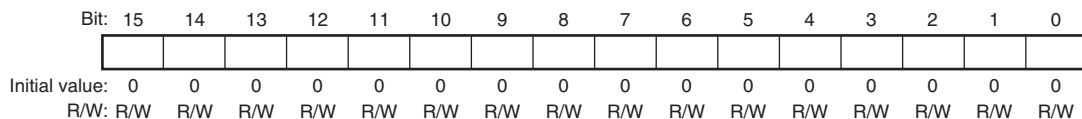


Note: TADCOBRA\_4 and TADCOBRB\_4 must not be accessed in eight bits; they should always be accessed in 16 bits.

### 10.3.11 Timer Counter (TCNT)

The TCNT counters are 16-bit readable/writable counters. This module has five TCNT counters, one each for channels 0 to 4.

The TCNT counters must not be accessed in eight bits; they should always be accessed in 16 bits.



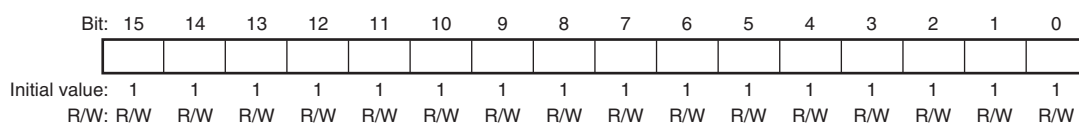
Note: The TCNT counters must not be accessed in eight bits; they should always be accessed in 16 bits.

### 10.3.12 Timer General Register (TGR)

The TGR registers are 16-bit readable/writable registers. This module has eighteen TGR registers, six for channel 0, two each for channels 1 and 2, four each for channels 3 and 4.

TGRA, TGRB, TGRC, and TGRD function as either output compare or input capture registers. TGRC and TGRD for channels 0, 3, and 4 can also be designated for operation as buffer registers. TGR buffer register combinations are TGRA and TGRC, and TGRB and TGRD.

TGRE\_0 and TGRF\_0 function as compare registers. When the TCNT\_0 count matches the TGRE\_0 value, an A/D converter start request can be issued. TGRF can also be designated for operation as a buffer register. TGR buffer register combination is TGRE and TGRF.



Note: The TGR registers must not be accessed in eight bits; they should always be accessed in 16 bits.  
TGR registers are initialized to H'FFFF.

### 10.3.13 Timer Start Register (TSTR)

TSTR is an 8-bit readable/writable register that selects operation/stoppage of TCNT for channels 0 to 4.

When setting the operating mode in TMDR or setting the count clock in TCR, first stop the TCNT counter.

Bit:	7	6	5	4	3	2	1	0
	CST4	CST3	-	-	-	CST2	CST1	CST0
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	CST4	0	R/W	Counter Start 4 and 3
6	CST3	0	R/W	These bits select operation or stoppage for TCNT. If 0 is written to the CST bit during operation with the TIOC pin designated for output, the counter stops but the TIOC pin output compare output level is retained. If TIOR is written to when the CST bit is cleared to 0, the pin output level will be changed to the set initial output value. 0: TCNT_4 and TCNT_3 count operation is stopped 1: TCNT_4 and TCNT_3 performs count operation
5 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	CST2	0	R/W	Counter Start 2 to 0
1	CST1	0	R/W	These bits select operation or stoppage for TCNT.
0	CST0	0	R/W	If 0 is written to the CST bit during operation with the TIOC pin designated for output, the counter stops but the TIOC pin output compare output level is retained. If TIOR is written to when the CST bit is cleared to 0, the pin output level will be changed to the set initial output value. 0: TCNT_2 to TCNT_0 count operation is stopped 1: TCNT_2 to TCNT_0 performs count operation

### 10.3.14 Timer Synchronous Register (TSYR)

TSYR is an 8-bit readable/writable register that selects independent operation or synchronous operation for the channel 0 to 4 TCNT counters. A channel performs synchronous operation when the corresponding bit in TSYR is set to 1.

Bit:	7	6	5	4	3	2	1	0
	SYNC4	SYNC3	-	-	-	SYNC2	SYNC1	SYNC0
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	SYNC4	0	R/W	Timer Synchronous operation 4 and 3
6	SYNC3	0	R/W	These bits are used to select whether operation is independent of or synchronized with other channels. When synchronous operation is selected, the TCNT synchronous presetting of multiple channels, and synchronous clearing by counter clearing on another channel, are possible. To set synchronous operation, the SYNC bits for at least two channels must be set to 1. To set synchronous clearing, in addition to the SYNC bit, the TCNT clearing source must also be set by means of bits CCLR0 to CCLR2 in TCR. 0: TCNT_4 and TCNT_3 operate independently (TCNT presetting/clearing is unrelated to other channels) 1: TCNT_4 and TCNT_3 performs synchronous operation TCNT synchronous presetting/synchronous clearing is possible
5 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	SYNC2	0	R/W	Timer Synchronous operation 2 to 0
1	SYNC1	0	R/W	These bits are used to select whether operation is independent of or synchronized with other channels.
0	SYNC0	0	R/W	When synchronous operation is selected, the TCNT synchronous presetting of multiple channels, and synchronous clearing by counter clearing on another channel, are possible. To set synchronous operation, the SYNC bits for at least two channels must be set to 1. To set synchronous clearing, in addition to the SYNC bit, the TCNT clearing source must also be set by means of bits CCLR0 to CCLR2 in TCR. 0: TCNT_2 to TCNT_0 operates independently (TCNT presetting /clearing is unrelated to other channels) 1: TCNT_2 to TCNT_0 performs synchronous operation TCNT synchronous presetting/synchronous clearing is possible

### 10.3.15 Timer Read/Write Enable Register (TRWER)

TRWER is an 8-bit readable/writable register that enables or disables access to the registers and counters which have write-protection capability against accidental modification in channels 3 and 4.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	RWE
Initial value:	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	RWE	1	R/W	Read/Write Enable Enables or disables access to the registers which have write-protection capability against accidental modification. 0: Disables read/write access to the registers 1: Enables read/write access to the registers [Clearing condition] <ul style="list-style-type: none"> <li>When 0 is written to the RWE bit after reading RWE = 1</li> </ul>

- Registers and counters having write-protection capability against accidental modification  
22 registers: TCR\_3, TCR\_4, TMDR\_3, TMDR\_4, TIORH\_3, TIORH\_4, TIORL\_3, TIORL\_4, TIER\_3, TIER\_4, TGRA\_3, TGRA\_4, TGRB\_3, TGRB\_4, TOER, TOCR1, TOCR2, TGCR, TCDR, TDDR, TCNT\_3, and TCNT4.

### 10.3.16 Timer Output Master Enable Register (TOER)

TOER is an 8-bit readable/writable register that enables/disables output settings for output pins TIOC4D, TIOC4C, TIOC3D, TIOC4B, TIOC4A, and TIOC3B. These pins do not output correctly if the TOER bits have not been set. Set TOER of CH3 and CH4 prior to setting TIOR of CH3 and CH4.

Make settings of the TOER while counting by the TCNT registers of channels 3 and 4 is stopped.

Bit:	7	6	5	4	3	2	1	0
	-	-	OE4D	OE4C	OE3D	OE4B	OE4A	OE3B
Initial value:	1	1	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
5	OE4D	0	R/W	Master Enable TIOC4D This bit enables/disables the TIOC4D pin output for this module. 0: Output for this module is disabled (inactive level)* 1: Output for this module is enabled
4	OE4C	0	R/W	Master Enable TIOC4C This bit enables/disables the TIOC4C pin output for this module. 0: Output for this module is disabled (inactive level)* 1: Output for this module is enabled
3	OE3D	0	R/W	Master Enable TIOC3D This bit enables/disables the TIOC3D pin output for this module. 0: Output for this module is disabled (inactive level)* 1: Output for this module is enabled
2	OE4B	0	R/W	Master Enable TIOC4B This bit enables/disables the TIOC4B pin output for this module. 0: Output for this module is disabled (inactive level)* 1: Output for this module is enabled
1	OE4A	0	R/W	Master Enable TIOC4A This bit enables/disables the TIOC4A pin output for this module. 0: Output for this module is disabled (inactive level)* 1: Output for this module is enabled
0	OE3B	0	R/W	Master Enable TIOC3B This bit enables/disables the TIOC3B pin output for this module. 0: Output for this module is disabled (inactive level)* 1: Output for this module is enabled

Note: \* The inactive level is determined by the settings in timer output control registers 1 and 2 (TOCR1 and TOCR2). For details, refer to section 10.3.17, Timer Output Control Register 1 (TOCR1), and section 10.3.18, Timer Output Control Register 2 (TOCR2). Set these bits to 1 to enable output for this module in other than complementary PWM or reset-synchronized PWM mode. When these bits are set to 0, low level is output.

### 10.3.17 Timer Output Control Register 1 (TOCR1)

TOCR1 is an 8-bit readable/writable register that enables/disables PWM synchronized toggle output in complementary PWM mode/reset synchronized PWM mode, and controls output level inversion of PWM output.

Bit:	7	6	5	4	3	2	1	0
	-	PSYE	-	-	TOCL	TOCS	OLSN	OLSP
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R	R/(W)* <sup>3</sup>	R/W	R/W	R/W

Bit	Bit Name	Initial value	R/W	Description
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	PSYE	0	R/W	PWM Synchronous Output Enable This bit selects the enable/disable of toggle output synchronized with the PWM period. 0: Toggle output is disabled 1: Toggle output is enabled
5, 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	TOCL	0	R/(W)* <sup>3</sup>	TOC Register Write Protection* <sup>1</sup> This bit selects the enable/disable of write access to the TOCS, OLSN, and OLSP bits in TOCR1. 0: Write access to the TOCS, OLSN, and OLSP bits is enabled 1: Write access to the TOCS, OLSN, and OLSP bits is disabled
2	TOCS	0	R/W	TOC Select This bit selects either the TOCR1 or TOCR2 setting to be used for the output level in complementary PWM mode and reset-synchronized PWM mode. 0: TOCR1 setting is selected 1: TOCR2 setting is selected
1	OLSN	0	R/W	Output Level Select N* <sup>2</sup> * <sup>4</sup> This bit selects the negative phase output level in reset-synchronized PWM mode/complementary PWM mode. See Table 10.28.
0	OLSP	0	R/W	Output Level Select P* <sup>2</sup> This bit selects the positive phase output level in reset-synchronized PWM mode/complementary PWM mode. See Table 10.29.

Note 1. Setting the TOCL bit to 1 prevents accidental modification when the CPU goes out of control.

Note 2. Clearing the TOCS0 bit to 0 makes this bit setting valid.

Note 3. After power-on reset, 1 can be written only once. After 1 has been written, 0 cannot be written.

Note 4. If the dead-time is not generated, the negative-phase output will be the exact inverse of the positive-phase output. Furthermore, set OLSP and OLSN to the same value.

**Table 10.28 Output Level Select Function**

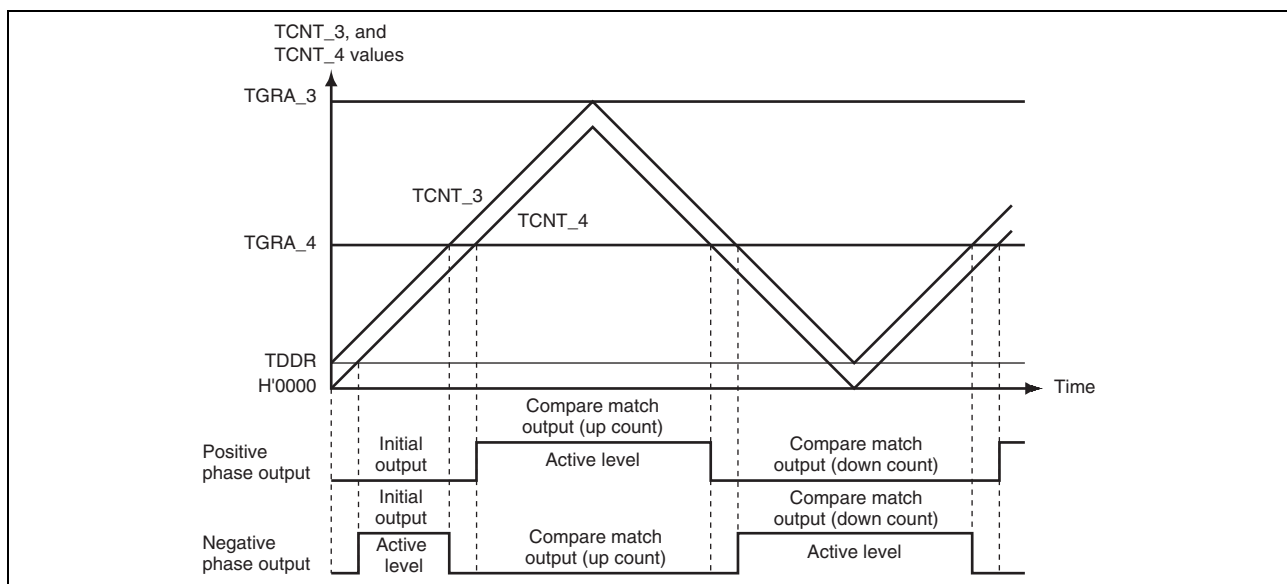
Bit 1	Function			
	Initial Output	Active Level	Up Count	Down Count
0	High level	Low level	High level	Low level
1	Low level	High level	Low level	High level

Note: The negative phase waveform initial output value changes to active level after elapse of the dead time after count start.

**Table 10.29 Output Level Select Function**

Bit 0	Function			
	Initial Output	Active Level	Up Count	Down Count
0	High level	Low level	Low level	High level
1	Low level	High level	High level	Low level

Figure 10.2 shows an example of complementary PWM mode output (1 phase) when OLSN = 1, OLSP = 1.



**Figure 10.2 Complementary PWM Mode Output Level Example**



### 10.3.18 Timer Output Control Register 2 (TOCR2)

TOCR2 is an 8-bit readable/writable register that controls output level inversion of PWM output in complementary PWM mode and reset-synchronized PWM mode.

Bit:	7	6	5	4	3	2	1	0
	BF[1:0]	OLS3N	OLS3P	OLS2N	OLS2P	OLS1N	OLS1P	
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial value	R/W	Description
7, 6	BF[1:0]	00	R/W	TOLBR Buffer Transfer Timing Select These bits select the timing for transferring data from TOLBR to TOCR2. For details, see Table 10.30.
5	OLS3N	0	R/W	Output Level Select 3N* This bit selects the output level on TIOC4D in reset-synchronized PWM mode/ complementary PWM mode. See Table 10.31.
4	OLS3P	0	R/W	Output Level Select 3P* This bit selects the output level on TIOC4B in reset-synchronized PWM mode/ complementary PWM mode. See Table 10.32.
3	OLS2N	0	R/W	Output Level Select 2N* This bit selects the output level on TIOC4C in reset-synchronized PWM mode/ complementary PWM mode. See Table 10.33.
2	OLS2P	0	R/W	Output Level Select 2P* This bit selects the output level on TIOC4A in reset-synchronized PWM mode/ complementary PWM mode. See Table 10.34.
1	OLS1N	0	R/W	Output Level Select 1N* This bit selects the output level on TIOC3D in reset-synchronized PWM mode/ complementary PWM mode. See Table 10.35.
0	OLS1P	0	R/W	Output Level Select 1P* This bit selects the output level on TIOC3B in reset-synchronized PWM mode/ complementary PWM mode. See Table 10.36.

Note: \* Setting the TOCS bit in TOCR1 to 1 makes this bit setting valid.  
If the dead-time is not generated, the negative-phase output will be the exact inverse of the positive-phase output.  
Furthermore, set OLSiP and OLSiN (i = 1, 2, 3) to the same value.

**Table 10.30 Setting of Bits BF1 and BF0**

Bit 7	Bit 6	Description	
BF1	BF0	Complementary PWM Mode	Reset-Synchronized PWM Mode
0	0	Does not transfer data from the buffer register (TOLBR) to TOCR2.	Does not transfer data from the buffer register (TOLBR) to TOCR2.
0	1	Transfers data from the buffer register (TOLBR) to TOCR2 at the crest of the TCNT_4 count.	Transfers data from the buffer register (TOLBR) to TOCR2 when TCNT_3/TCNT_4 is cleared
1	0	Transfers data from the buffer register (TOLBR) to TOCR2 at the trough of the TCNT_4 count.	Setting prohibited
1	1	Transfers data from the buffer register (TOLBR) to TOCR2 at the crest and trough of the TCNT_4 count.	Setting prohibited

**Table 10.31 TIOC4D Output Level Select Function**

Bit 5	Function			
	Initial Output	Active Level	Compare Match Output	
Up Count			Down Count	
0	High level	Low level	High level	Low level
1	Low level	High level	Low level	High level

Note: The negative phase waveform initial output value changes to the active level after elapse of the dead time after count start.

**Table 10.32 TIOC4B Output Level Select Function**

Bit 4	Function			
	Initial Output	Active Level	Compare Match Output	
Up Count			Down Count	
0	High level	Low level	Low level	High level
1	Low level	High level	High level	Low level

**Table 10.33 TIOC4C Output Level Select Function**

Bit 3	Function			
	Initial Output	Active Level	Compare Match Output	
Up Count			Down Count	
0	High level	Low level	High level	Low level
1	Low level	High level	Low level	High level

Note: The negative phase waveform initial output value changes to the active level after elapse of the dead time after count start.

**Table 10.34 TIOC4A Output Level Select Function**

Bit 2	Function			
	Initial Output	Active Level	Compare Match Output	
Up Count			Down Count	
0	High level	Low level	Low level	High level
1	Low level	High level	High level	Low level

**Table 10.35 TIOC3D Output Level Select Function**

Bit 1	Function			
	Initial Output	Active Level	Compare Match Output	
Up Count			Down Count	
0	High level	Low level	High level	Low level
1	Low level	High level	Low level	High level

Note: The negative phase waveform initial output value changes to the active level after elapse of the dead time after count start.

**Table 10.36 TIOC4B Output Level Select Function**

Bit 0	Function			
	Initial Output	Active Level	Compare Match Output	
Up Count			Down Count	
0	High level	Low level	Low level	High level
1	Low level	High level	High level	Low level

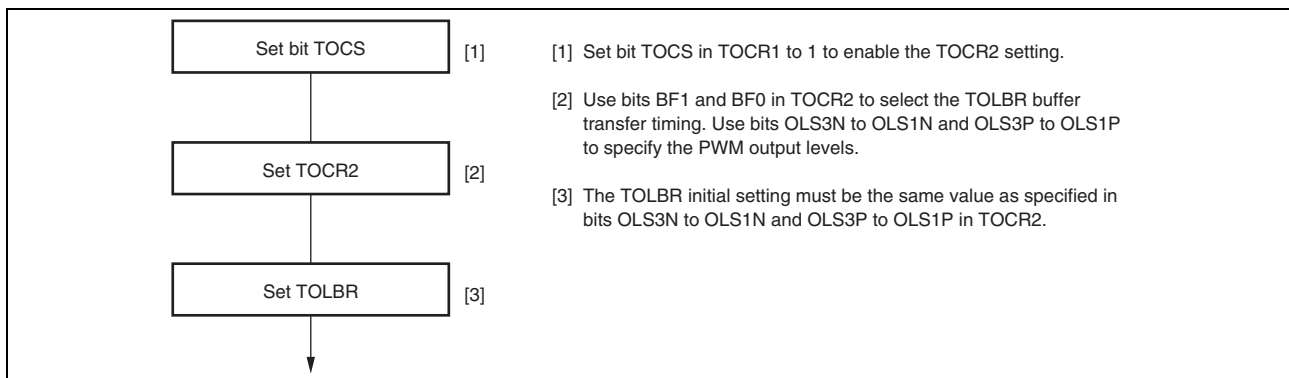
### 10.3.19 Timer Output Level Buffer Register (TOLBR)

TOLBR is an 8-bit readable/writable register that functions as a buffer for TOCR2 and specifies the PWM output level in complementary PWM mode and reset-synchronized PWM mode.

Bit:	7	6	5	4	3	2	1	0
	-	-	OLS3N	OLS3P	OLS2N	OLS2P	OLS1N	OLS1P
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial value	R/W	Description
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	OLS3N	0	R/W	Specifies the buffer value to be transferred to the OLS3N bit in TOCR2.
4	OLS3P	0	R/W	Specifies the buffer value to be transferred to the OLS3P bit in TOCR2.
3	OLS2N	0	R/W	Specifies the buffer value to be transferred to the OLS2N bit in TOCR2.
2	OLS2P	0	R/W	Specifies the buffer value to be transferred to the OLS2P bit in TOCR2.
1	OLS1N	0	R/W	Specifies the buffer value to be transferred to the OLS1N bit in TOCR2.
0	OLS1P	0	R/W	Specifies the buffer value to be transferred to the OLS1P bit in TOCR2.

Figure 10.3 shows an example of the PWM output level setting procedure in buffer operation.



**Figure 10.3 PWM Output Level Setting Procedure in Buffer Operation**

### 10.3.20 Timer Gate Control Register (TGCR)

TGCR is an 8-bit readable/writable register that controls the waveform output necessary for brushless DC motor control in reset-synchronized PWM mode/complementary PWM mode. These register settings are ineffective for anything other than complementary PWM mode/reset-synchronized PWM mode.

Bit:	7	6	5	4	3	2	1	0
	-	BDC	N	P	FB	WF	VF	UF
Initial value:	1	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial value	R/W	Description
7	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
6	BDC	0	R/W	Brushless DC Motor This bit selects whether to make the functions of this register (TGCR) effective or ineffective. 0: Ordinary output 1: Functions of this register are made effective
5	N	0	R/W	Negative Phase Output (N) Control This bit selects whether the level output or the reset-synchronized PWM/complementary PWM output while the reverse pins (TIOC3D, TIOC4C, and TIOC4D) are output. 0: Level output 1: Reset synchronized PWM/complementary PWM output
4	P	0	R/W	Positive Phase Output (P) Control This bit selects whether the level output or the reset-synchronized PWM/complementary PWM output while the positive pin (TIOC3B, TIOC4A, and TIOC4B) are output. 0: Level output 1: Reset synchronized PWM/complementary PWM output
3	FB	0	R/W	External Feedback Signal Enable This bit selects whether the switching of the output of the positive/negative phase is carried out automatically with channel-0 TGRA, TGRB, TGRC input capture signals or by writing 0 or 1 to bits 2 to 0 in TGCR. 0: Output switching is external input (Input sources are channel 0 TGRA, TGRB, TGRC input capture signal) 1: Output switching is carried out by software (setting values of UF, VF, and WF in TGCR).
2	WF	0	R/W	Output Phase Switch 2 to 0
1	VF	0	R/W	These bits set the positive phase/negative phase output phase on or off state. The setting of these bits is valid only when the FB bit in this register is set to 1. In this case, the setting of bits 2 to 0 is a substitute for external input. See Table 10.37.
0	UF	0	R/W	

**Table 10.37 Output level Select Function**

Bit 2	Bit 1	Bit 0	Function					
			TIOC3B	TIOC4A	TIOC4B	TIOC3D	TIOC4C	TIOC4D
WF	VF	UF	U Phase	V Phase	W Phase	U Phase	V Phase	W Phase
0	0	0	OFF	OFF	OFF	OFF	OFF	OFF
		1	ON	OFF	OFF	OFF	OFF	ON
	1	0	OFF	ON	OFF	ON	OFF	OFF
		1	OFF	ON	OFF	OFF	OFF	ON
1	0	0	OFF	OFF	ON	OFF	ON	OFF
		1	ON	OFF	OFF	OFF	ON	OFF
	1	0	OFF	OFF	ON	ON	OFF	OFF
		1	OFF	OFF	OFF	OFF	OFF	OFF

### 10.3.21 Timer Subcounter (TCNTS)

TCNTS is a 16-bit read-only counter that is used only in complementary PWM mode.

The initial value of TCNTS is H'0000.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Note: Accessing the TCNTS in 8-bit units is prohibited. Always access in 16-bit units.

### 10.3.22 Timer Dead Time Data Register (TDDR)

TDDR is a 16-bit register, used only in complementary PWM mode that specifies the TCNT\_3 and TCNT\_4 counter offset values. In complementary PWM mode, when the TCNT\_3 and TCNT\_4 counters are cleared and then restarted, the TDDR register value is loaded into the TCNT\_3 counter and the count operation starts.

The initial value of TDDR is H'FFFF.

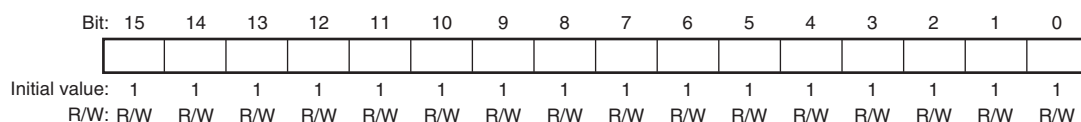
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: Accessing the TDDR in 8-bit units is prohibited. Always access in 16-bit units.

### 10.3.23 Timer Cycle Data Register (TCDR)

TCDR is a 16-bit register used only in complementary PWM mode. Set half the PWM carrier sync value (note that this value should be at least double the value specified in TDDR + 3) as the TCDR register value. This register is constantly compared with the TCNTS counter in complementary PWM mode, and when a match occurs, the TCNTS counter switches direction (decrement to increment).

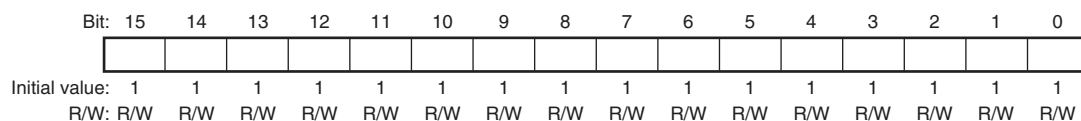
The initial value of TCDR is H'FFFF.



Note: Accessing the TCDR in 8-bit units is prohibited. Always access in 16-bit units.

### 10.3.24 Timer Cycle Buffer Register (TCBR)

TCBR is a 16-bit register used only in complementary PWM mode. It functions as a buffer register for the TCDR register. The TCBR register values are transferred to the TCDR register with the transfer timing set in the TMDR register. The initial value of TCBR is H'FFFF.



Note: Accessing the TCBR in 8-bit units is prohibited. Always access in 16-bit units.

### 10.3.25 Timer Interrupt Skipping Set Register (TITCR)

TITCR is an 8-bit readable/writable register that enables or disables interrupt skipping and specifies the interrupt skipping count. This module has one TITCR.

Bit:	7	6	5	4	3	2	1	0
	T3AEN	3ACOR[2:0]			T4VEN	4VCOR[2:0]		
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial value	R/W	Description
7	T3AEN	0	R/W	T3AEN Enables or disables TGIA_3 interrupt skipping. 0: TGIA_3 interrupt skipping disabled 1: TGIA_3 interrupt skipping enabled
6 to 4	3ACOR[2:0]	000	R/W	These bits specify the TGIA_3 interrupt skipping count within the range from 0 to 7.* For details, see Table 10.38.
3	T4VEN	0	R/W	T4VEN Enables or disables TCIV_4 interrupt skipping. 0: TCIV_4 interrupt skipping disabled 1: TCIV_4 interrupt skipping enabled
2 to 0	4VCOR[2:0]	000	R/W	These bits specify the TCIV_4 interrupt skipping count within the range from 0 to 7.* For details, see Table 10.39.

Note: \* When 0 is specified for the interrupt skipping count, no interrupt skipping will be performed. Before changing the interrupt skipping count, be sure to clear the T3AEN and T4VEN bits to 0 to clear the skipping counter (TICNT).



**Table 10.38 Setting of Interrupt Skipping Count by Bits 3ACOR2 to 3ACOR0**

Bit 6	Bit 5	Bit 4	Description
3ACOR2	3ACOR1	3ACOR0	
0	0	0	Does not skip TGIA_3 interrupts.
0	0	1	Sets the TGIA_3 interrupt skipping count to 1.
0	1	0	Sets the TGIA_3 interrupt skipping count to 2.
0	1	1	Sets the TGIA_3 interrupt skipping count to 3.
1	0	0	Sets the TGIA_3 interrupt skipping count to 4.
1	0	1	Sets the TGIA_3 interrupt skipping count to 5.
1	1	0	Sets the TGIA_3 interrupt skipping count to 6.
1	1	1	Sets the TGIA_3 interrupt skipping count to 7.

**Table 10.39 Setting of Interrupt Skipping Count by Bits 4VCOR2 to 4VCOR0**

Bit 2	Bit 1	Bit 0	Description
4VCOR2	4VCOR1	4VCOR0	
0	0	0	Does not skip TCIV_4 interrupts.
0	0	1	Sets the TCIV_4 interrupt skipping count to 1.
0	1	0	Sets the TCIV_4 interrupt skipping count to 2.
0	1	1	Sets the TCIV_4 interrupt skipping count to 3.
1	0	0	Sets the TCIV_4 interrupt skipping count to 4.
1	0	1	Sets the TCIV_4 interrupt skipping count to 5.
1	1	0	Sets the TCIV_4 interrupt skipping count to 6.
1	1	1	Sets the TCIV_4 interrupt skipping count to 7.

### 10.3.26 Timer Interrupt Skipping Counter (TITCNT)

TITCNT is an 8-bit readable/writable counter. This module has one TITCNT. TITCNT retains its value even after stopping the count operation of TCNT\_3 and TCNT\_4.

Bit:	7	6	5	4	3	2	1	0
	-	3ACNT[2:0]			-	4VCNT[2:0]		
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	—	0	R	Reserved This bit is always read as 0.
6 to 4	3ACNT[2:0]	000	R	TGI <sub>A</sub> _3 Interrupt Counter While the T3AEN bit in TITCR is set to 1, the count in these bits is incremented every time a TGI <sub>A</sub> _3 interrupt occurs. [Clearing conditions] <ul style="list-style-type: none"> <li>• When the 3ACNT2 to 3ACNT0 value in TITCNT matches the 3ACOR2 to 3ACOR0 value in TITCR</li> <li>• When the T3AEN bit in TITCR is cleared to 0</li> <li>• When the 3ACOR2 to 3ACOR0 bits in TITCR are cleared to 0</li> </ul>
3	—	0	R	Reserved This bit is always read as 0.
2 to 0	4VCNT[2:0]	000	R	TCIV <sub>4</sub> Interrupt Counter While the T4VEN bit in TITCR is set to 1, the count in these bits is incremented every time a TCIV <sub>4</sub> interrupt occurs. [Clearing conditions] <ul style="list-style-type: none"> <li>• When the 4VCNT2 to 4VCNT0 value in TITCNT matches the 4VCOR2 to 4VCOR2 value in TITCR</li> <li>• When the T4VEN bit in TITCR is cleared to 0</li> <li>• When the 4VCOR2 to 4VCOR2 bits in TITCR are cleared to 0</li> </ul>

Note: To clear the TITCNT, clear the bits T3AEN and T4VEN in TITCR to 0.

### 10.3.27 Timer Buffer Transfer Set Register (TBTER)

TBTER is an 8-bit readable/writable register that enables or disables transfer from the buffer registers\* used in complementary PWM mode to the temporary registers and specifies whether to link the transfer with interrupt skipping operation. This module has one TBTER.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	BTE[1:0]	
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	BTE[1:0]	00	R/W	These bits enable or disable transfer from the buffer registers* used in complementary PWM mode to the temporary registers and specify whether to link the transfer with interrupt skipping operation. For details, see Table 10.40.

Note: \* Applicable buffer registers:  
TGRC\_3, TGRD\_3, TGRC\_4, TGRD\_4, and TCBR

**Table 10.40 Setting of Bits BTE1 and BTE0**

Bit 1	Bit 0	Description
BTE1	BTE0	Description
0	0	Enables transfer from the buffer registers to the temporary registers*1 and does not link the transfer with interrupt skipping operation.
0	1	Disables transfer from the buffer registers to the temporary registers.
1	0	Links transfer from the buffer registers to the temporary registers with interrupt skipping operation.*2
1	1	Setting prohibited

Note 1. Data is transferred according to the MD3 to MD0 bit setting in TMDR. For details, refer to section 10.4.8, Complementary PWM Mode.

Note 2. When interrupt skipping is disabled (the T3AEN and T4VEN bits are cleared to 0 in the timer interrupt skipping set register (TITCR) or the skipping count set bits (3ACOR and 4VCOR) in TITCR are cleared to 0), be sure to disable link of buffer transfer with interrupt skipping (clear the BTE1 bit in the timer buffer transfer set register (TBTER) to 0). If link with interrupt skipping is enabled while interrupt skipping is disabled, buffer transfer will not be performed.

### 10.3.28 Timer Dead Time Enable Register (TDER)

TDER is an 8-bit readable/writable register that controls dead time generation in complementary PWM mode. This module has one TDER in channel 3. TDER must be modified only while TCNT stops.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	TDER
Initial value:	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R/(W)

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	TDER	1	R/(W)	Dead Time Enable Specifies whether to generate dead time. 0: Does not generate dead time 1: Generates dead time* [Clearing condition] • When 0 is written to TDER after reading TDER = 1

Note: \* TDDR must be set to 1 or a larger value.

### 10.3.29 Timer Waveform Control Register (TWCR)

TWCR is an 8-bit readable/writable register that controls the waveform when synchronous counter clearing occurs in TCNT\_3 and TCNT\_4 in complementary PWM mode and specifies whether to clear the counters at TGRA\_3 compare match. The CCE bit and WRE bit in TWCR must be modified only while TCNT stops.

Bit:	7	6	5	4	3	2	1	0
	CCE	-	-	-	-	-	-	WRE
Initial value:	0*	0	0	0	0	0	0	0
R/W:	R/(W)	R	R	R	R	R	R	R/(W)

Note: \* Do not set to 1 when complementary PWM mode is not selected.

Bit	Bit Name	Initial Value	R/W	Description
7	CCE	0*	R/(W)	Compare Match Clear Enable Specifies whether to clear counters at TGRA_3 compare match in complementary PWM mode. 0: Does not clear counters at TGRA_3 compare match 1: Clears counters at TGRA_3 compare match [Setting condition] • When 1 is written to CCE after reading CCE = 0
6 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	WRE	0	R/(W)	Initial Output Suppression Enable Selects the waveform output when synchronous counter clearing occurs in complementary PWM mode. The initial output is suppressed only when synchronous clearing occurs within the Tb interval at the trough in complementary PWM mode. When synchronous clearing occurs outside this interval, the initial value specified in TOCR is output regardless of the WRE bit setting. The initial value is also output when synchronous clearing occurs in the Tb interval at the trough immediately after TCNT_3 and TCNT_4 start operation. For the Tb interval at the trough in complementary PWM mode, see Figure 10.40. 0: Outputs the initial value specified in TOCR 1: Suppresses initial output [Setting condition] • When 1 is written to WRE after reading WRE = 0

Note: \* Do not set to 1 when complementary PWM mode is not selected.

### 10.3.30 Bus Master Interface

The timer counters (TCNT), general registers (TGR), timer subcounter (TCNTS), timer cycle buffer register (TCBR), timer dead time data register (TDDR), timer cycle data register (TCDR), timer A/D converter start request control register (TADCR), timer A/D converter start request cycle set registers (TADCOR), and timer A/D converter start request cycle set buffer registers (TADCOBR) are 16-bit registers. 16-bit read/writes is possible. 8-bit read/write is not possible. Always access in 16-bit units.

All registers other than the above registers are 8-bit registers. 8-bit read/writes is possible. 16-bit read/writes is not possible. Always access in 8-bit units.

## 10.4 Operation

### 10.4.1 Basic Functions

Each channel has a TCNT and TGR register. TCNT performs up-counting, and is also capable of free-running operation, cycle counting, and external event counting.

Each TGR can be used as an input capture register or output compare register.

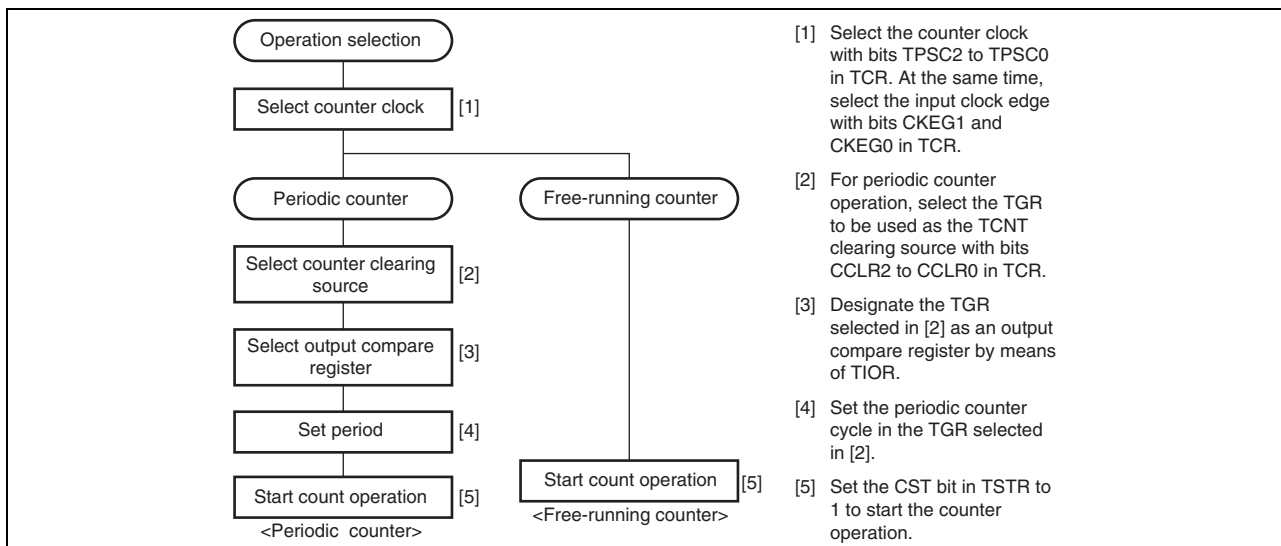
Always select functions for external pins of this module using the general I/O ports.

#### (1) Counter Operation

When one of bits CST0 to CST4 in TSTR is set to 1, the TCNT counter for the corresponding channel begins counting. TCNT can operate as a free-running counter, periodic counter, for example.

#### (a) Example of Count Operation Setting Procedure

Figure 10.4 shows an example of the count operation setting procedure.

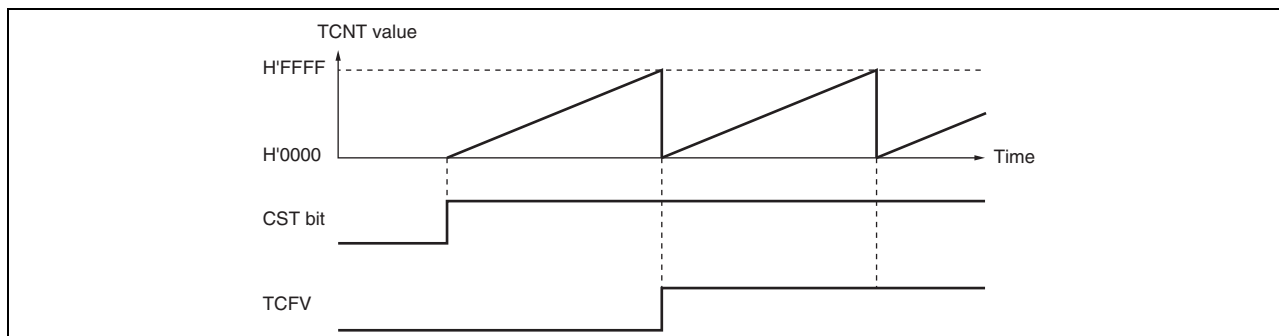


**Figure 10.4 Example of Counter Operation Setting Procedure**

**(b) Free-Running Count Operation and Periodic Count Operation:**

Immediately after a reset, the TCNT counters of this module are all designated as free-running counters. When the relevant bit in TSTR is set to 1 the corresponding TCNT counter starts up-count operation as a free-running counter. When TCNT overflows (from H'FFFF to H'0000), the TCFV bit in TSR is set to 1. If the value of the corresponding TCIEV bit in TIER is 1 at this point, this module requests an interrupt. After overflow, TCNT starts counting up again from H'0000.

Figure 10.5 illustrates free-running counter operation.

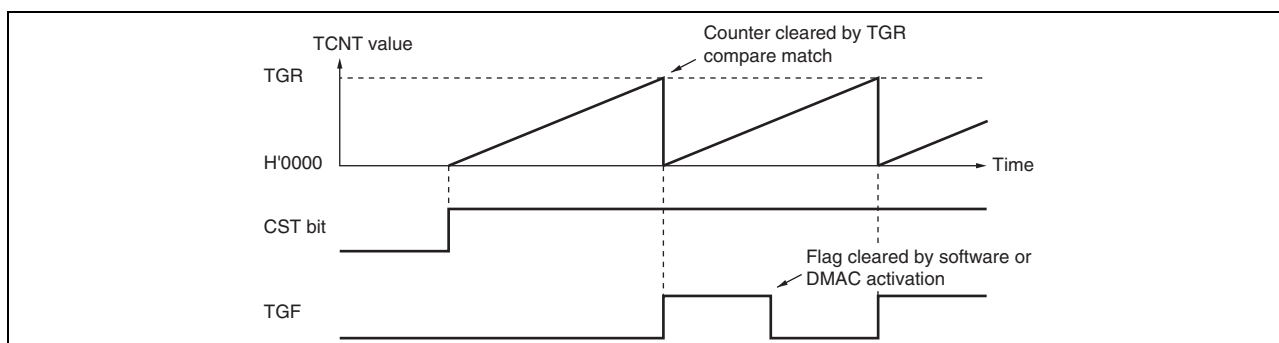


**Figure 10.5 Free-Running Counter Operation**

When compare match is selected as the TCNT clearing source, the TCNT counter for the relevant channel performs periodic count operation. The TGR register for setting the period is designated as an output compare register, and counter clearing by compare match is selected by means of bits CCLR0 to CCLR2 in TCR. After the settings have been made, TCNT starts up-count operation as a periodic counter when the corresponding bit in TSTR is set to 1. When the count value matches the value in TGR, the TGF bit in TSR is set to 1 and TCNT is cleared to H'0000.

If the value of the corresponding TGIE bit in TIER is 1 at this point, this module requests an interrupt. After a compare match, TCNT starts counting up again from H'0000.

Figure 10.6 illustrates periodic counter operation.



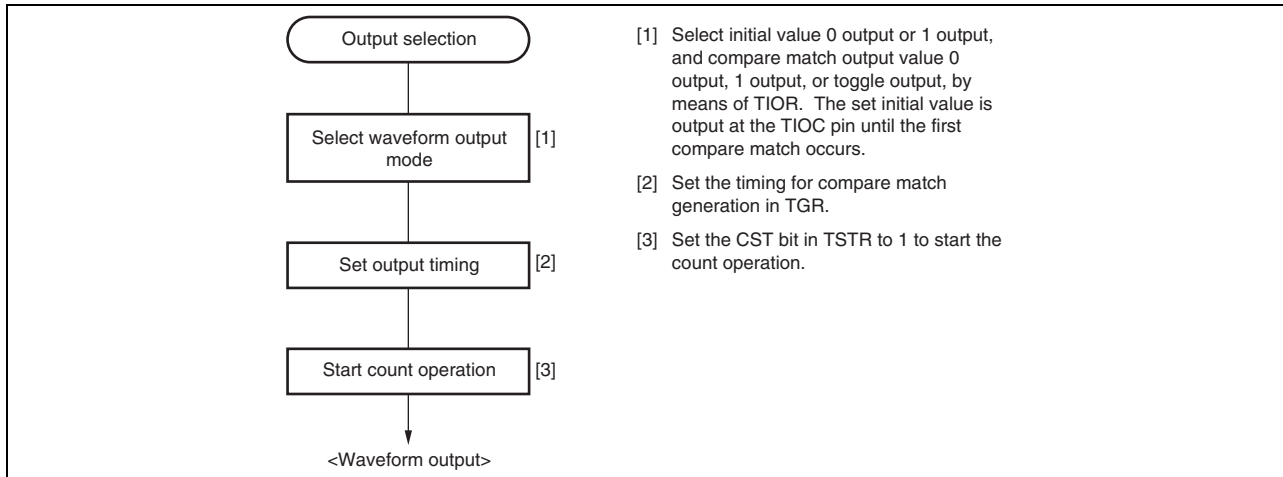
**Figure 10.6 Periodic Counter Operation**

## (2) Waveform Output by Compare Match

This module can perform 0, 1, or toggle output from the corresponding output pin using compare match.

### (a) Example of Setting Procedure for Waveform Output by Compare Match

Figure 10.7 shows an example of the setting procedure for waveform output by compare match.

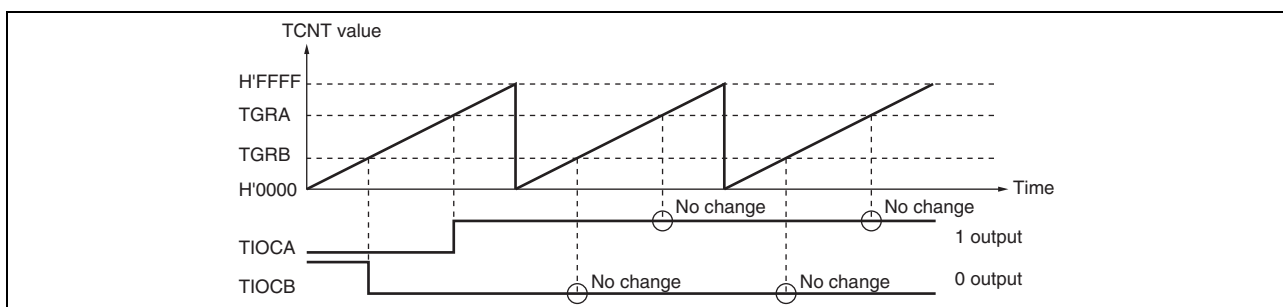


**Figure 10.7** Example of Setting Procedure for Waveform Output by Compare Match

### (b) Examples of Waveform Output Operation:

Figure 10.8 shows an example of 0 output/1 output.

In this example TCNT has been designated as a free-running counter, and settings have been made such that 1 is output by compare match A, and 0 is output by compare match B. When the set level and the pin level coincide, the pin level does not change.

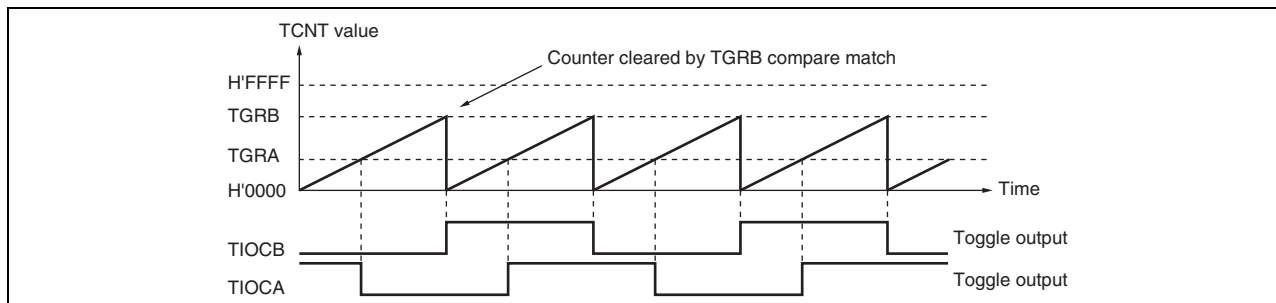


**Figure 10.8** Example of 0 Output/1 Output Operation



Figure 10.9 shows an example of toggle output.

In this example, TCNT has been designated as a periodic counter (with counter clearing on compare match B), and settings have been made such that the output is toggled by both compare match A and compare match B.



**Figure 10.9 Example of Toggle Output Operation**

### (3) Input Capture Function

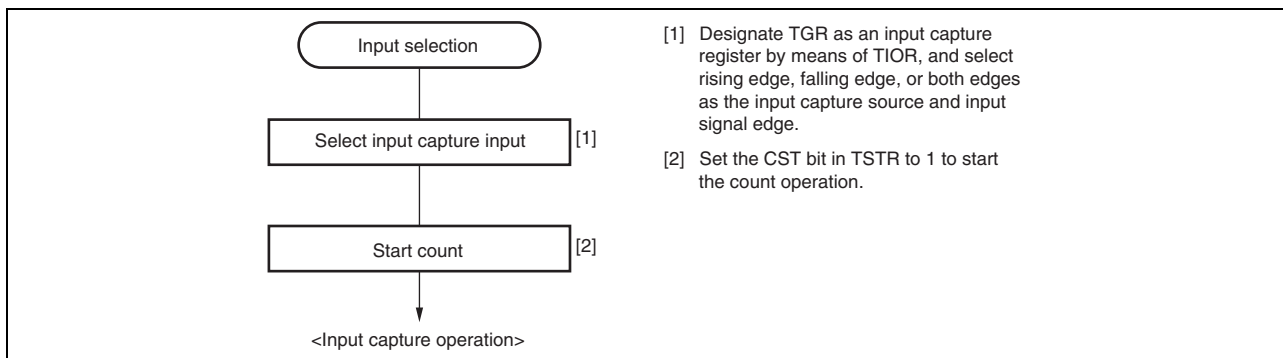
The TCNT value can be transferred to TGR on detection of the TIOC pin input edge.

Rising edge, falling edge, or both edges can be selected as the detected edge. For channels 0 and 1, it is also possible to specify another channel's counter input clock or compare match signal as the input capture source.

**Note:** When another channel's counter input clock is used as the input capture input for channels 0 and 1, P0 $\phi$ /1 should not be selected as the counter input clock used for input capture input. Input capture will not be generated if P0 $\phi$ /1 is selected.

#### (a) Example of Input Capture Operation Setting Procedure

Figure 10.10 shows an example of the input capture operation setting procedure.



**Figure 10.10** Example of Input Capture Operation Setting Procedure

(b) Example of Input Capture Operation

Figure 10.11 shows an example of input capture operation.

In this example both rising and falling edges have been selected as the TIOCA pin input capture input edge, the falling edge has been selected as the TIOCB pin input capture input edge, and counter clearing by TGRB input capture has been designated for TCNT.

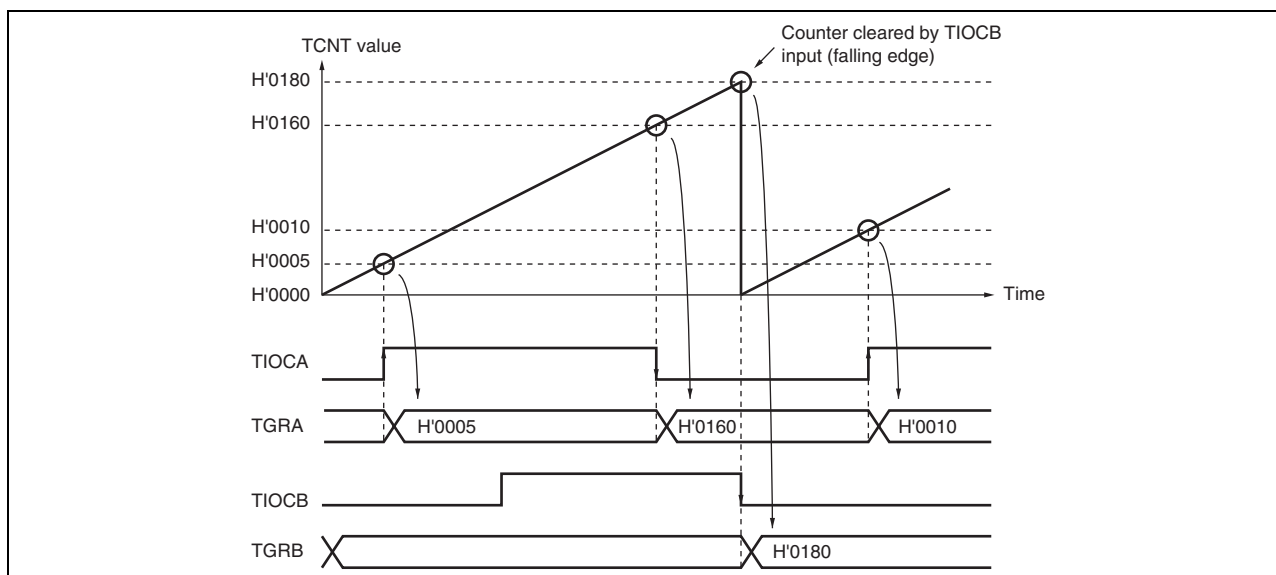


Figure 10.11 Example of Input Capture Operation

## 10.4.2 Synchronous Operation

In synchronous operation, the values in a number of TCNT counters can be rewritten simultaneously (synchronous presetting). Also, a number of TCNT counters can be cleared simultaneously by making the appropriate setting in TCR (synchronous clearing).

Synchronous operation enables TGR to be incremented with respect to a single time base.

Channels 0 to 4 can all be designated for synchronous operation.

### (1) Example of Synchronous Operation Setting Procedure

Figure 10.12 shows an example of the synchronous operation setting procedure.

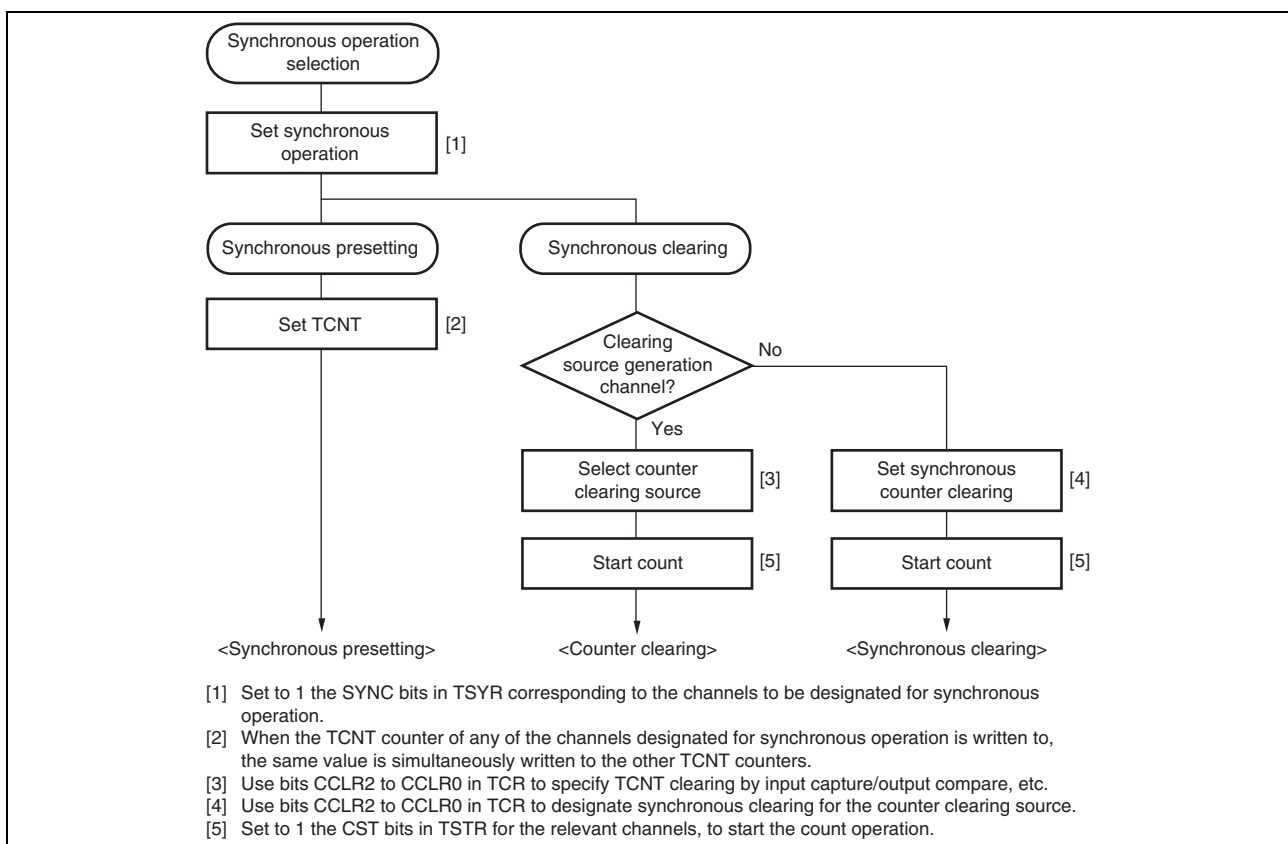


Figure 10.12 Example of Synchronous Operation Setting Procedure

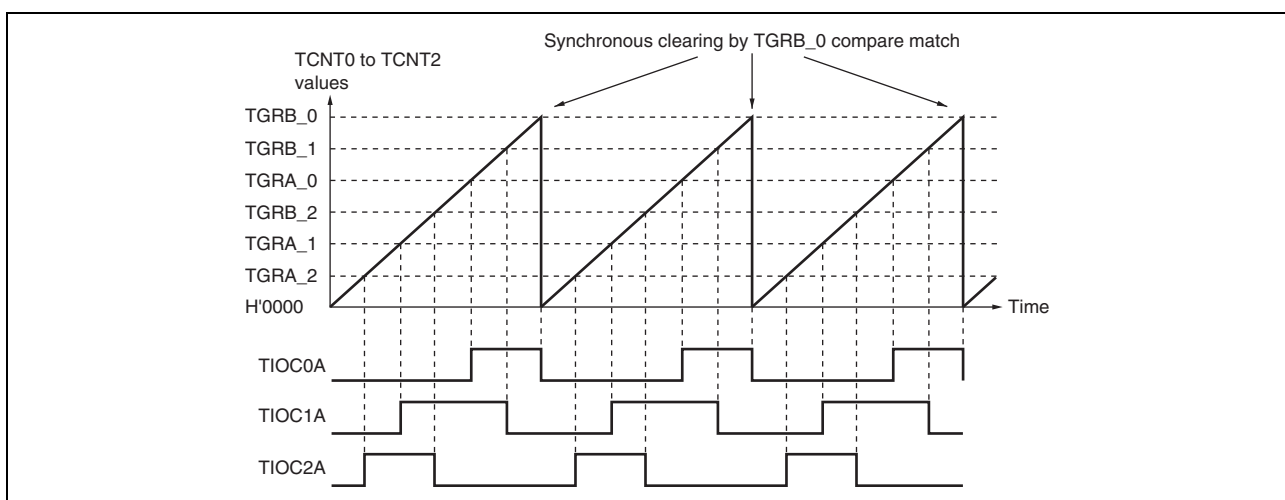
## (2) Example of Synchronous Operation

Figure 10.13 shows an example of synchronous operation.

In this example, synchronous operation and PWM mode 1 have been designated for channels 0 to 2, TGRB\_0 compare match has been set as the channel 0 counter clearing source, and synchronous clearing has been set for the channel 1 and 2 counter clearing source.

Three-phase PWM waveforms are output from pins TIOC0A, TIOC1A, and TIOC2A. At this time, synchronous presetting, and synchronous clearing by TGRB\_0 compare match, are performed for channel 0 to 2 TCNT counters, and the data set in TGRB\_0 is used as the PWM cycle.

For details of PWM modes, see section 10.4.5, PWM Modes.



**Figure 10.13 Example of Synchronous Operation**

### 10.4.3 Buffer Operation

Buffer operation, provided for channels 0, 3, and 4, enables TGRC and TGRD to be used as buffer registers. In channel 0, TGRF can also be used as a buffer register.

Buffer operation differs depending on whether TGR has been designated as an input capture register or as a compare match register.

Note: TGRE\_0 cannot be designated as an input capture register and can only operate as a compare match register.

Table 10.41 shows the register combinations used in buffer operation.

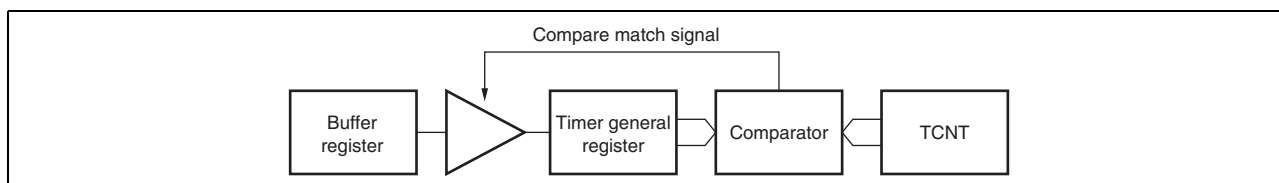
**Table 10.41 Register Combinations in Buffer Operation**

Channel	Timer General Register	Buffer Register
0	TGRA_0	TGRC_0
	TGRB_0	TGRD_0
	TGRE_0	TGRF_0
3	TGRA_3	TGRC_3
	TGRB_3	TGRD_3
4	TGRA_4	TGRC_4
	TGRB_4	TGRD_4

- When TGR is an output compare register

When a compare match occurs, the value in the buffer register for the corresponding channel is transferred to the timer general register.

This operation is illustrated in Figure 10.14.

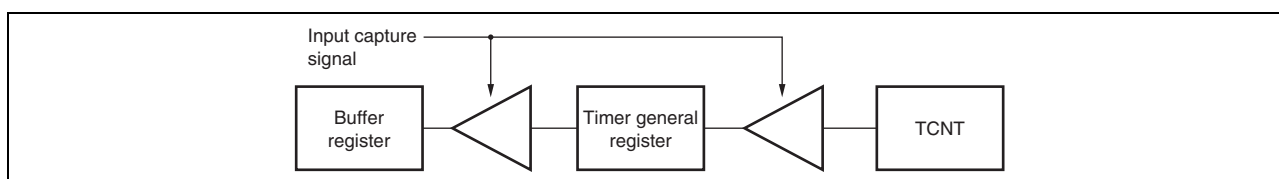


**Figure 10.14 Compare Match Buffer Operation**

- When TGR is an input capture register

When input capture occurs, the value in TCNT is transferred to TGR and the value previously held in the timer general register is transferred to the buffer register.

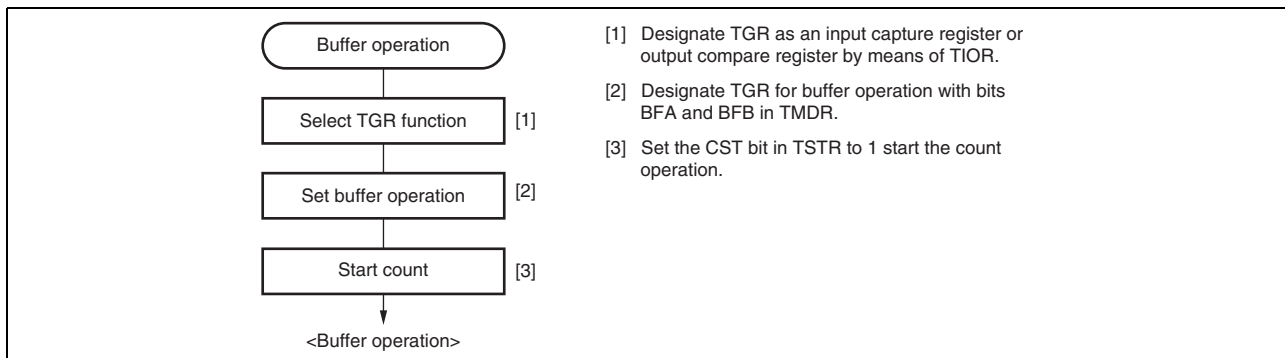
This operation is illustrated in Figure 10.15.



**Figure 10.15 Input Capture Buffer Operation**

### (1) Example of Buffer Operation Setting Procedure

Figure 10.16 shows an example of the buffer operation setting procedure.



**Figure 10.16** Example of Buffer Operation Setting Procedure

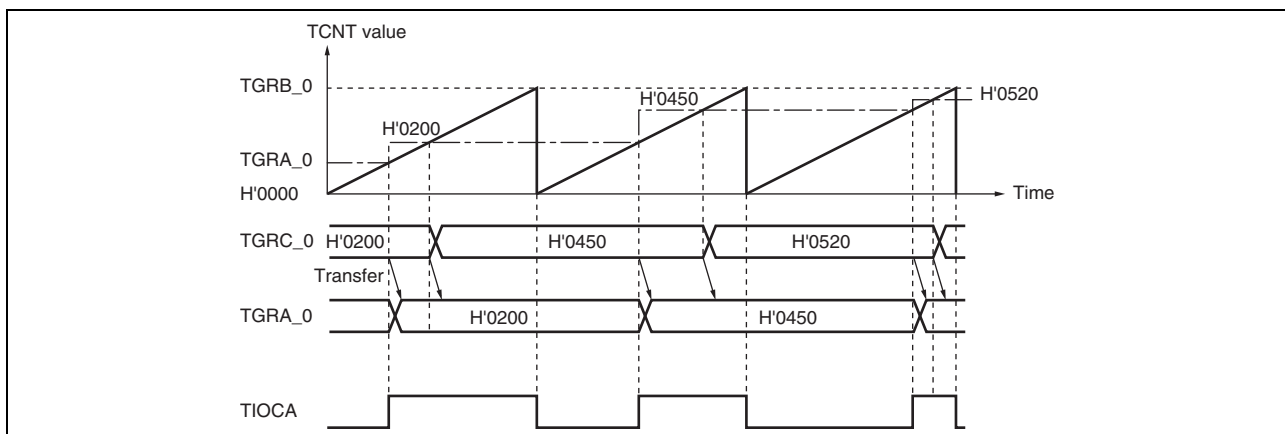
### (2) Examples of Buffer Operation

#### (a) When TGR is an output compare register

Figure 10.17 shows an operation example in which PWM mode 1 has been designated for channel 0, and buffer operation has been designated for TGRA and TGRC. The settings used in this example are TCNT clearing by compare match B, 1 output at compare match A, and 0 output at compare match B. In this example, the TTSA bit in TBTM is cleared to 0.

As buffer operation has been set, when compare match A occurs the output changes and the value in buffer register TGRC is simultaneously transferred to timer general register TGRA. This operation is repeated each time that compare match A occurs.

For details of PWM modes, see section 10.4.5, PWM Modes.



**Figure 10.17** Example of Buffer Operation (1)

(b) When TGR is an input capture register

Figure 10.18 shows an operation example in which TGRA has been designated as an input capture register, and buffer operation has been designated for TGRA and TGRC.

Counter clearing by TGRA input capture has been set for TCNT, and both rising and falling edges have been selected as the TIOCA pin input capture input edge.

As buffer operation has been set, when the TCNT value is stored in TGRA upon the occurrence of input capture A, the value previously stored in TGRA is simultaneously transferred to TGRC.

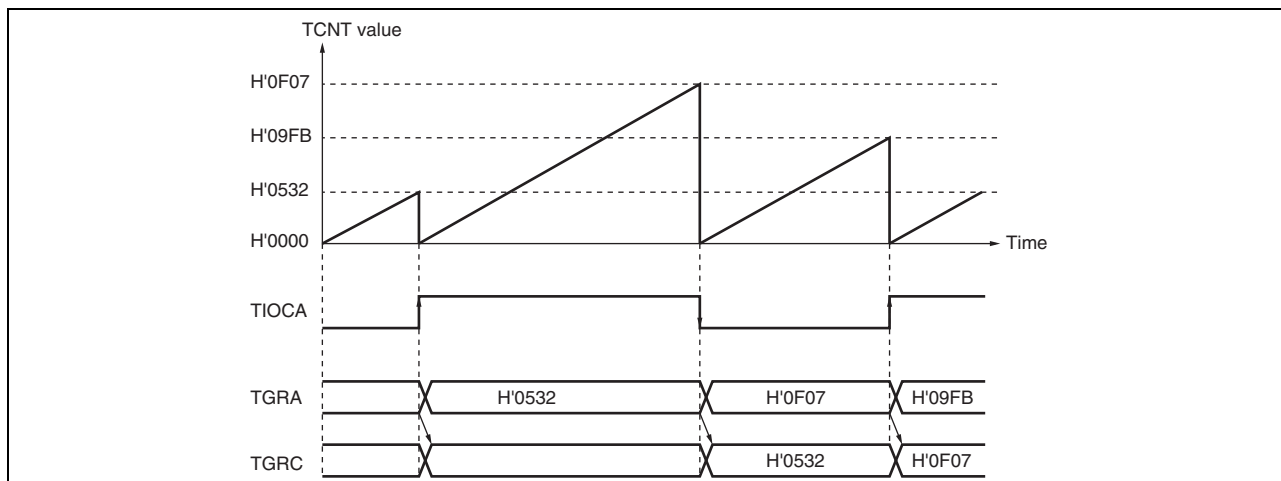


Figure 10.18 Example of Buffer Operation (2)



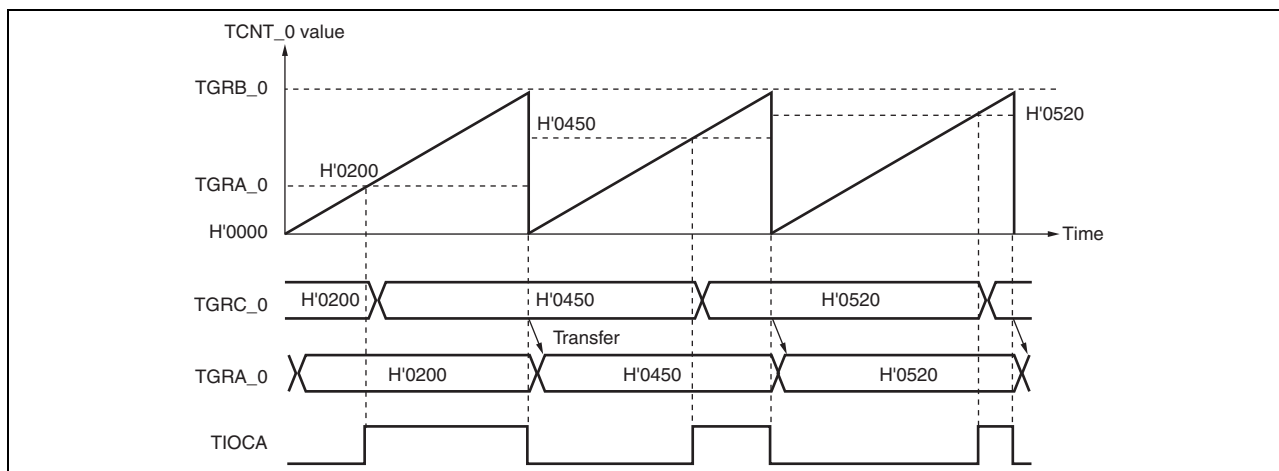
### (3) Selecting Timing for Transfer from Buffer Registers to Timer General Registers in Buffer Operation

The timing for transfer from buffer registers to timer general registers can be selected in PWM mode 1 or 2 for channel 0 or in PWM mode 1 for channels 3 and 4 by setting the buffer operation transfer mode registers (TBTM\_0, TBTM\_3, and TBTM\_4). Either compare match (initial setting) or TCNT clearing can be selected for the transfer timing. TCNT clearing as transfer timing is one of the following cases.

- When TCNT overflows (H'FFFF to H'0000)
- When H'0000 is written to TCNT during counting
- When TCNT is cleared to H'0000 under the condition specified in the CCLR2 to CCLR0 bits in TCR

Note: TBTM must be modified only while TCNT stops.

Figure 10.19 shows an operation example in which PWM mode 1 is designated for channel 0 and buffer operation is designated for TGRA\_0 and TGRC\_0. The settings used in this example are TCNT\_0 clearing by compare match B, 1 output at compare match A, and 0 output at compare match B. The TTSA bit in TBTM\_0 is set to 1.



**Figure 10.19** Example of Buffer Operation When TCNT\_0 Clearing is Selected for TGRC\_0 to TGRA\_0 Transfer Timing

### 10.4.4 Cascaded Operation

In cascaded operation, two 16-bit counters for different channels are used together as a 32-bit counter.

This function works by counting the channel 1 counter clock upon overflow/underflow of TCNT\_2 as set in bits TPSC0 to TPSC2 in TCR.

Underflow occurs only when the lower 16-bit TCNT is in phase-counting mode.

Table 10.42 shows the register combinations used in cascaded operation.

Note: When phase counting mode is set for channel 1, the counter clock setting is invalid and the counters operates independently in phase counting mode.

**Table 10.42 Cascaded Combinations**

Combination	Upper 16 Bits	Lower 16 Bits
Channels 1 and 2	TCNT_1	TCNT_2

For simultaneous input capture of TCNT\_1 and TCNT\_2 during cascaded operation, additional input capture input pins can be specified by the input capture control register (TICCR). The condition for input capture is the detection of an edge in the signal obtained from the logical OR of the signal on the main input pin and the signal on the additional input pin. For details, see (4), Cascaded Operation Example (c). For input capture in cascade connection, refer to section 10.7.22, Simultaneous Capture of TCNT\_1 and TCNT\_2 in Cascade Connection.

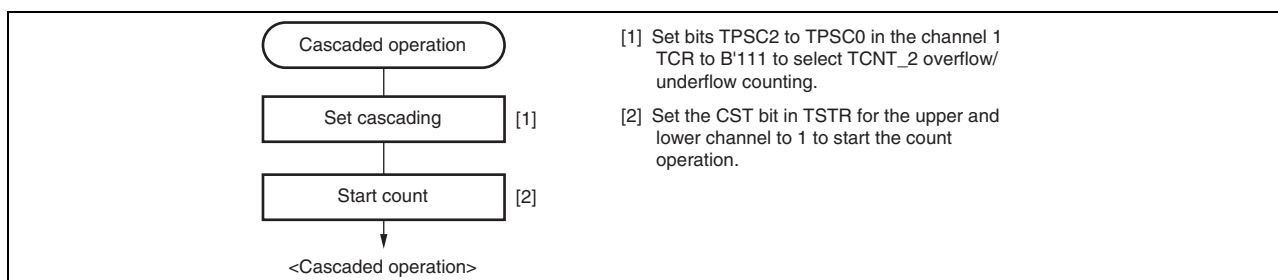
Table 10.43 show the TICCR setting and input capture input pins.

**Table 10.43 TICCR Setting and Input Capture Input Pins**

Target Input Capture	TICCR Setting	Input Capture Input Pins
Input capture from TCNT_1 to TGRA_1	I2AE bit = 0 (initial value)	TIOC1A
	I2AE bit = 1	TIOC1A, TIOC2A
Input capture from TCNT_1 to TGRB_1	I2BE bit = 0 (initial value)	TIOC1B
	I2BE bit = 1	TIOC1B, TIOC2B
Input capture from TCNT_2 to TGRA_2	I1AE bit = 0 (initial value)	TIOC2A
	I1AE bit = 1	TIOC2A, TIOC1A
Input capture from TCNT_2 to TGRB_2	I1BE bit = 0 (initial value)	TIOC2B
	I1BE bit = 1	TIOC2B, TIOC1B

(1) Example of Cascaded Operation Setting Procedure

Figure 10.20 shows an example of the setting procedure for cascaded operation.

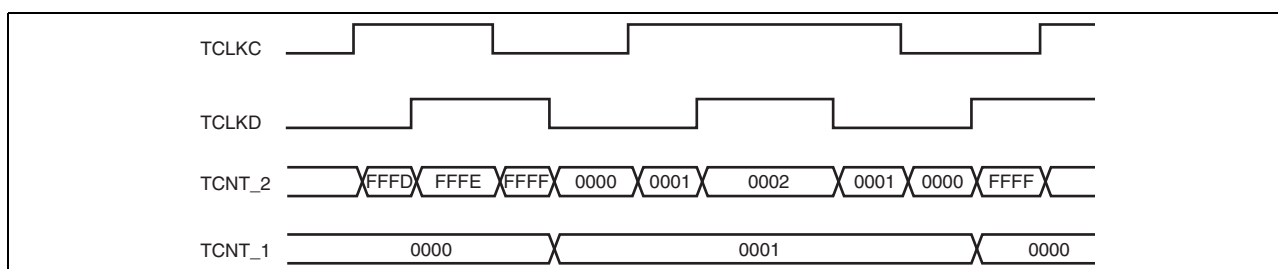


**Figure 10.20 Cascaded Operation Setting Procedure**

(2) Cascaded Operation Example (a)

Figure 10.21 illustrates the operation when TCNT\_2 overflow/underflow counting has been set for TCNT\_1 and phase counting mode has been designated for channel 2.

TCNT\_1 is incremented by TCNT\_2 overflow and decremented by TCNT\_2 underflow.



**Figure 10.21 Cascaded Operation Example (a)**

(3) Cascaded Operation Example (b)

Figure 10.22 illustrates the operation when TCNT\_1 and TCNT\_2 have been cascaded and the I2AE bit in TICCR has been set to 1 to include the TIOC2A pin in the TGRA\_1 input capture conditions. In this example, the IOA0 to IOA3 bits in TIOR\_1 have selected the TIOC1A rising edge for the input capture timing while the IOA0 to IOA3 bits in TIOR\_2 have selected the TIOC2A rising edge for the input capture timing.

Under these conditions, the rising edge of both TIOC1A and TIOC2A is used for the TGRA\_1 input capture condition. For the TGRA\_2 input capture condition, the TIOC2A rising edge is used.

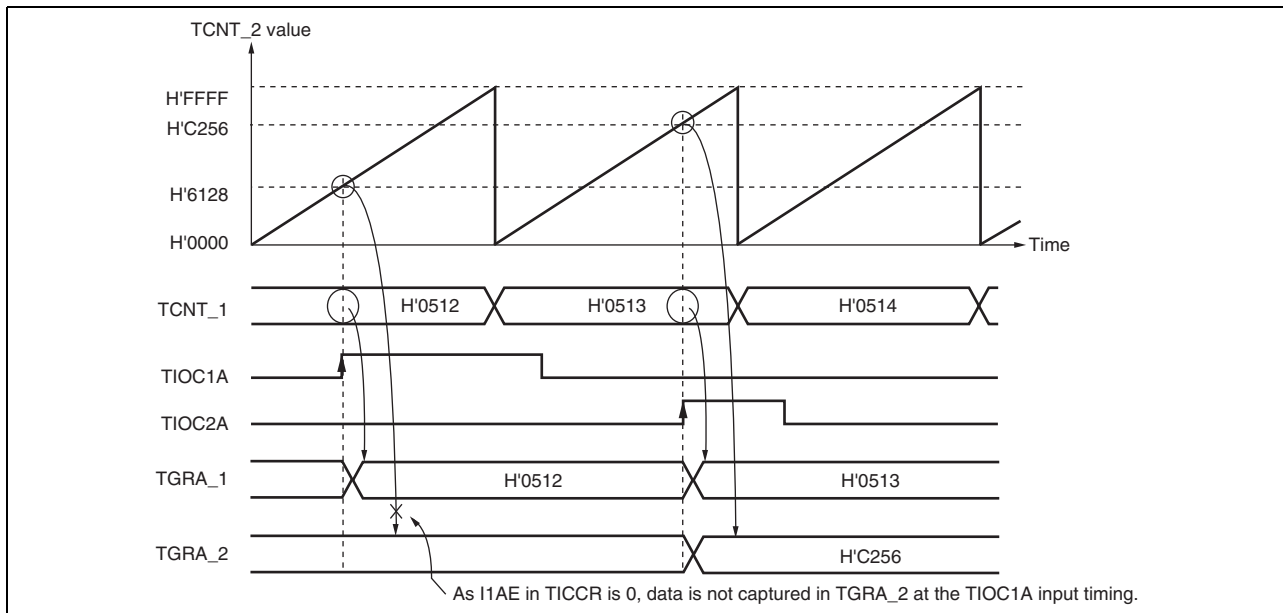


Figure 10.22 Cascaded Operation Example (b)

(4) Cascaded Operation Example (c)

Figure 10.23 illustrates the operation when TCNT\_1 and TCNT\_2 have been cascaded and the I2AE and I1AE bits in TICC1R have been set to 1 to include the TIOC2A and TIOC1A pins in the TGRA\_1 and TGRA\_2 input capture conditions, respectively. In this example, the IOA0 to IOA3 bits in both TIOR\_1 and TIOR\_2 have selected both the rising and falling edges for the input capture timing. Under these conditions, the ORed result of TIOC1A and TIOC2A input is used for the TGRA\_1 and TGRA\_2 input capture conditions.

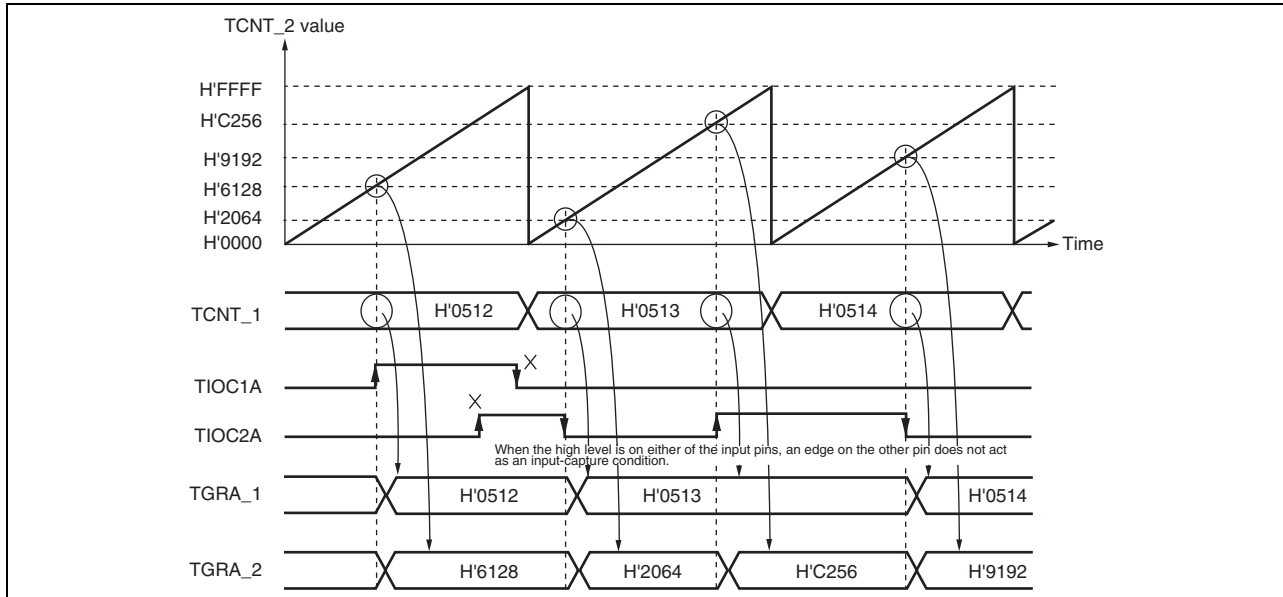


Figure 10.23 Cascaded Operation Example (c)

(5) Cascaded Operation Example (d)

Figure 10.24 illustrates the operation when TCNT\_1 and TCNT\_2 have been cascaded and the I2AE bit in TICCR has been set to 1 to include the TIOC2A pin in the TGRA\_1 input capture conditions. In this example, the IOA0 to IOA3 bits in TIOR\_1 have selected TGRA\_0 compare match or input capture occurrence for the input capture timing while the IOA0 to IOA3 bits in TIOR\_2 have selected the TIOC2A rising edge for the input capture timing.

Under these conditions, as TIOR\_1 has selected TGRA\_0 compare match or input capture occurrence for the input capture timing, the TIOC2A edge is not used for TGRA\_1 input capture condition although the I2AE bit in TICCR has been set to 1.

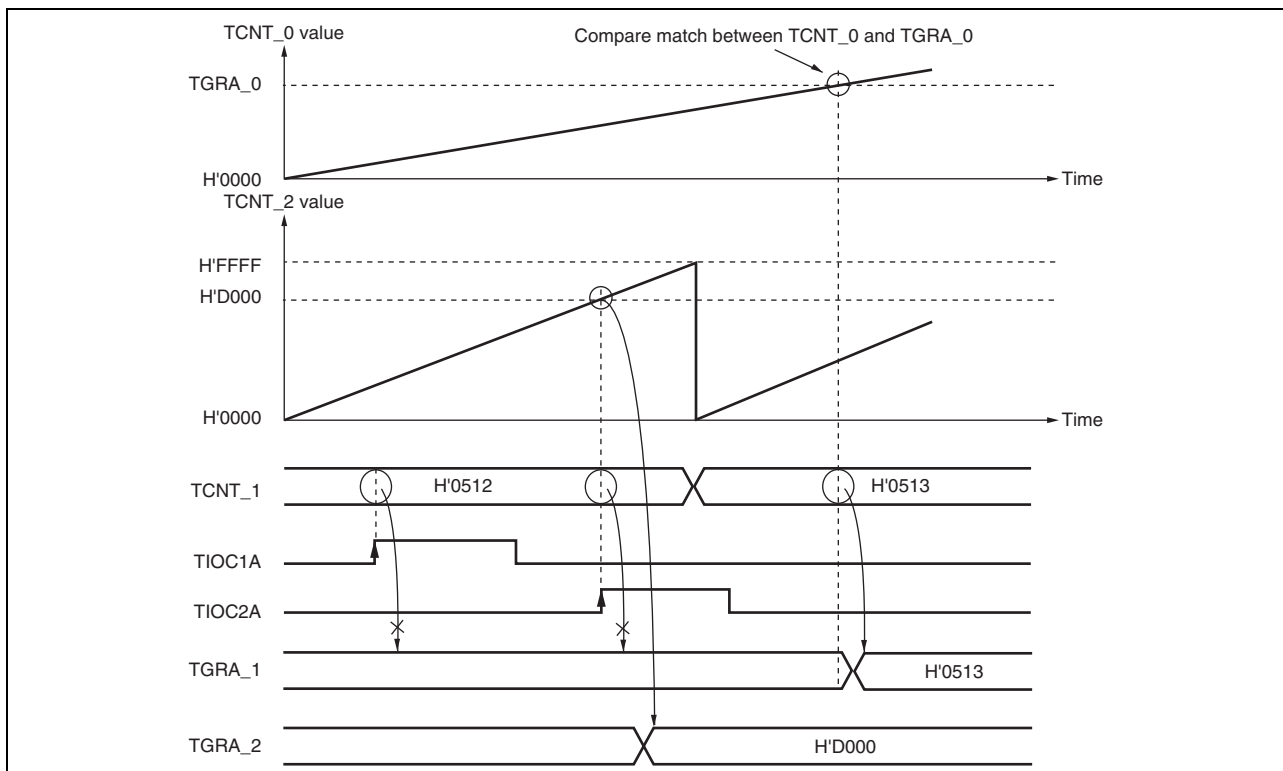


Figure 10.24 Cascaded Operation Example (d)

### 10.4.5 PWM Modes

In PWM mode, PWM waveforms are output from the output pins. The output level can be selected as 0, 1, or toggle output in response to a compare match of each TGR.

TGR registers settings can be used to output a PWM waveform in the range of 0% to 100% duty.

Designating TGR compare match as the counter clearing source enables the period to be set in that register. All channels can be designated for PWM mode independently. Synchronous operation is also possible.

There are two PWM modes, as described below.

- PWM mode 1

PWM output is generated from the TIOCA and TIOCC pins by pairing TGRA with TGRB and TGRC with TGRD. The output specified by bits IOA0 to IOA3 and IOC0 to IOC3 in TIOR is output from the TIOCA and TIOCC pins at compare matches A and C, and the output specified by bits IOB0 to IOB3 and IOD0 to IOD3 in TIOR is output at compare matches B and D. The initial output value is the value set in TGRA or TGRC. If the set values of paired TGRs are identical, the output value does not change when a compare match occurs.

In PWM mode 1, a maximum 8-phase PWM output is possible.

- PWM mode 2

PWM output is generated using one TGR as the cycle register and the others as duty registers. The output specified in TIOR is performed by means of compare matches. Upon counter clearing by a cycle register compare match, the output value of each pin is the initial value set in TIOR. If the set values of the cycle and duty registers are identical, the output value does not change when a compare match occurs.

In PWM mode 2, a maximum 8-phase PWM output is possible in combination use with synchronous operation.

The correspondence between PWM output pins and registers is shown in Table 10.44.

**Table 10.44 PWM Output Registers and Output Pins**

Channel	Registers	Output Pins	
		PWM Mode 1	PWM Mode 2
0	TGRA_0	TIOC0A	TIOC0A
	TGRB_0		TIOC0B
	TGRC_0	TIOC0C	TIOC0C
	TGRD_0		TIOC0D
1	TGRA_1	TIOC1A	TIOC1A
	TGRB_1		TIOC1B
2	TGRA_2	TIOC2A	TIOC2A
	TGRB_2		TIOC2B
3	TGRA_3	TIOC3A	Cannot be set
	TGRB_3		Cannot be set
	TGRC_3	TIOC3C	Cannot be set
	TGRD_3		Cannot be set
4	TGRA_4	TIOC4A	Cannot be set
	TGRB_4		Cannot be set
	TGRC_4	TIOC4C	Cannot be set
	TGRD_4		Cannot be set

Note: In PWM mode 2, PWM output is not possible for the TGR register in which the period is set.

(1) Example of PWM Mode Setting Procedure

Figure 10.25 shows an example of the PWM mode setting procedure.

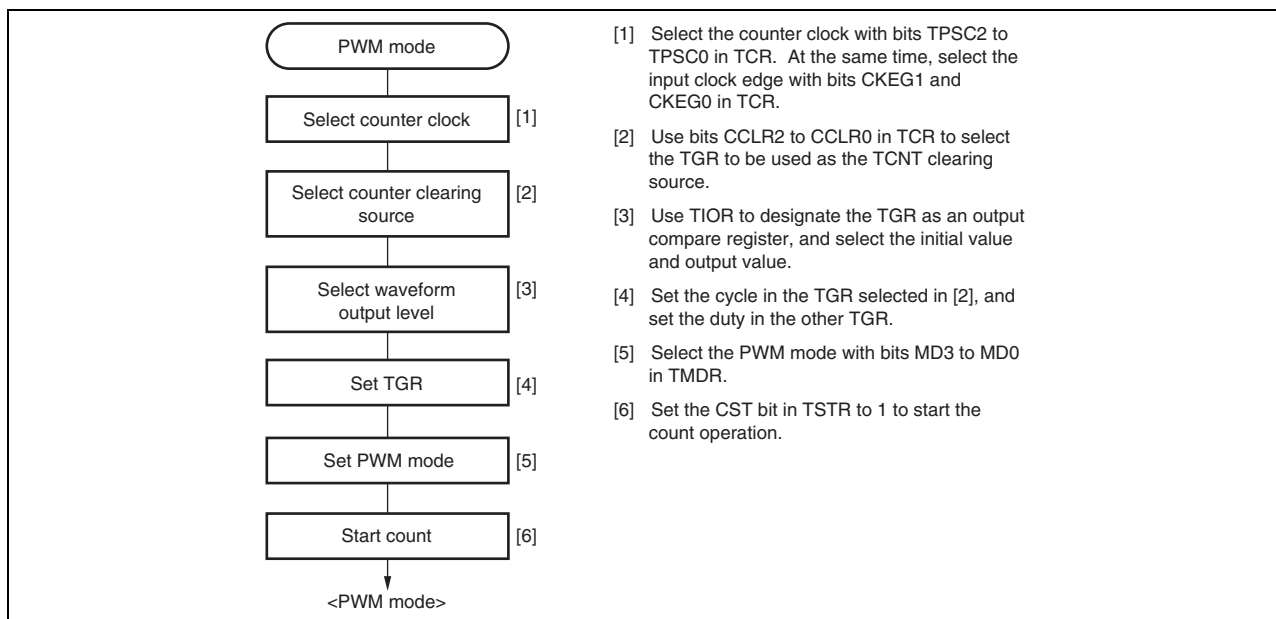


Figure 10.25 Example of PWM Mode Setting Procedure

(2) Examples of PWM Mode Operation

Figure 10.26 shows an example of PWM mode 1 operation.

In this example, TGRA compare match is set as the TCNT clearing source, 0 is set for the TGRA initial output value and output value, and 1 is set as the TGRB output value.

In this case, the value set in TGRA is used as the period, and the values set in the TGRB registers are used as the duty levels.

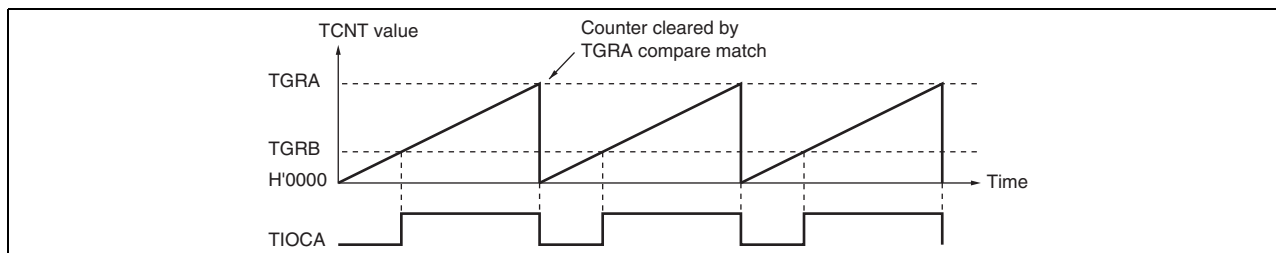


Figure 10.26 Example of PWM Mode Operation (1)



Figure 10.27 shows an example of PWM mode 2 operation.

In this example, synchronous operation is designated for channels 0 and 1, TGRB\_1 compare match is set as the TCNT clearing source, and 0 is set for the initial output value and 1 for the output value of the other TGR registers (TGRA\_0 to TGRD\_0, TGRA\_1), outputting a 5-phase PWM waveform.

In this case, the value set in TGRB\_1 is used as the cycle, and the values set in the other TGRs are used as the duty levels.

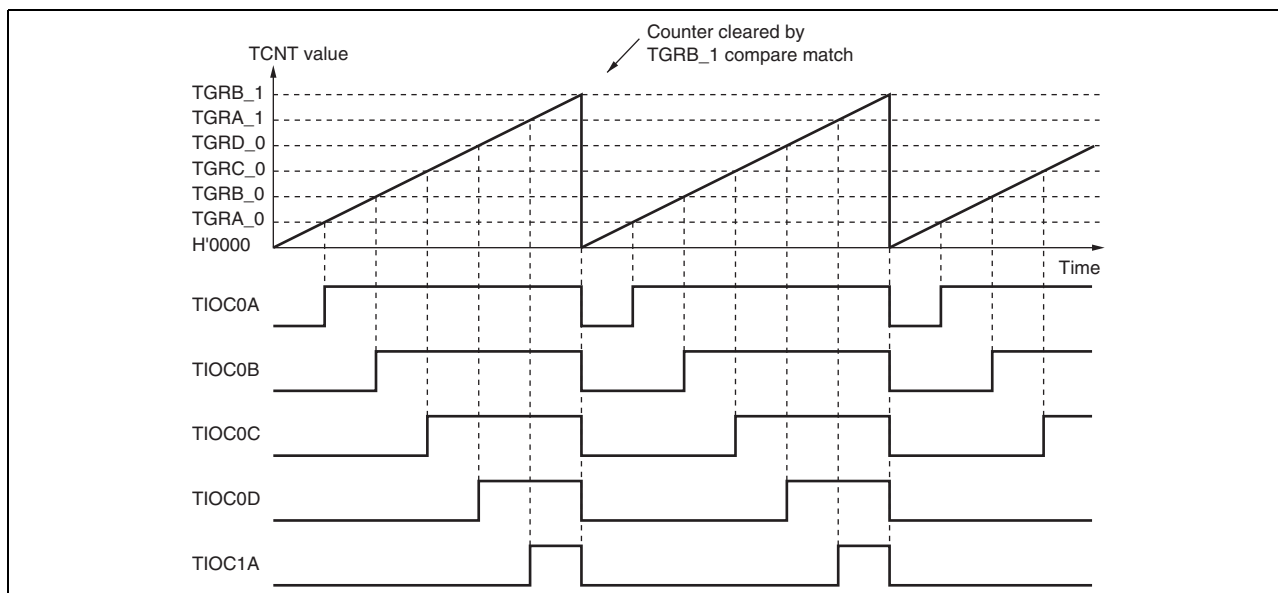


Figure 10.27 Example of PWM Mode Operation (2)

Figure 10.28 shows examples of PWM waveform output with 0% duty and 100% duty in PWM mode.

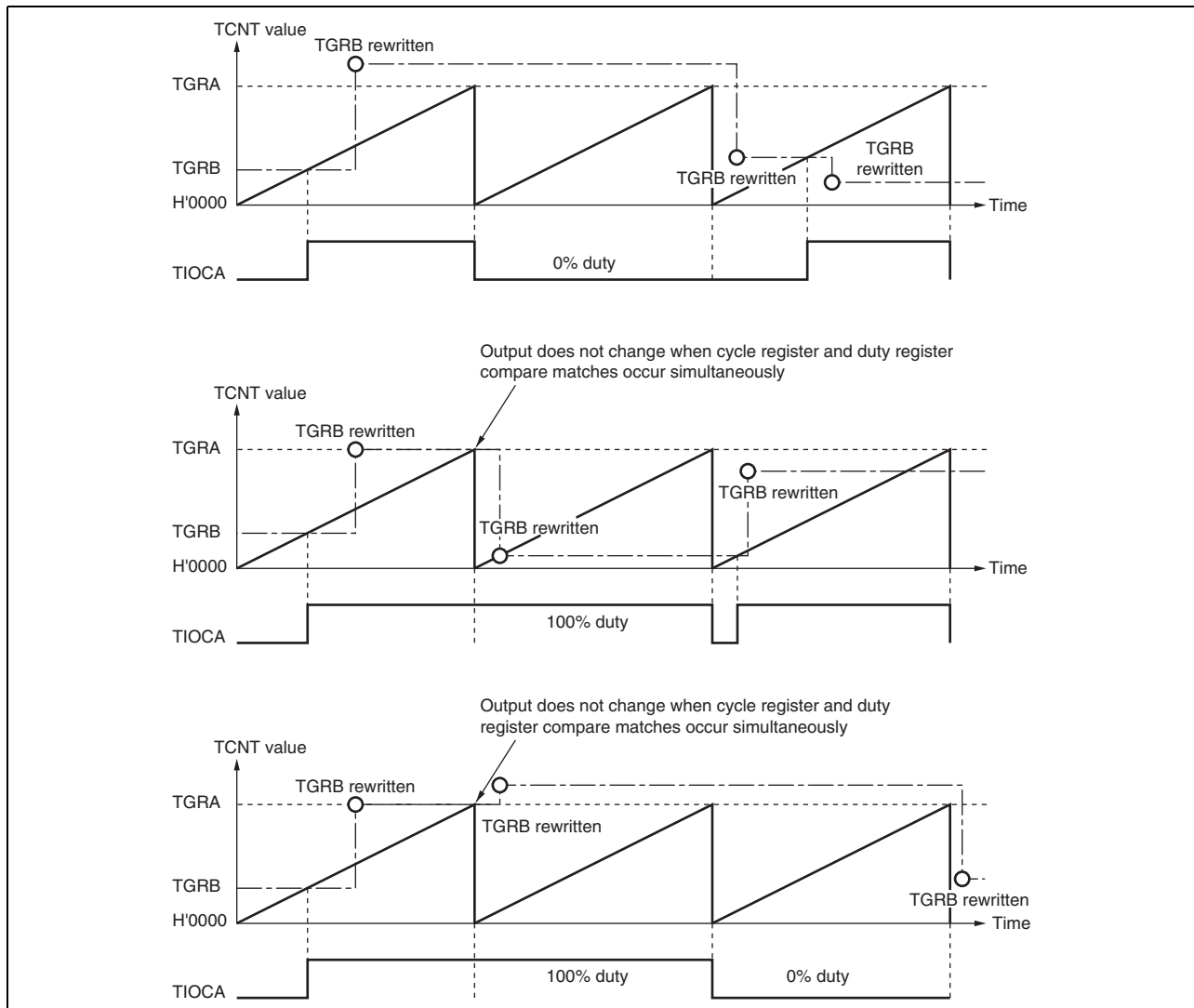


Figure 10.28 Example of PWM Mode Operation (3)

### 10.4.6 Phase Counting Mode

In phase counting mode, the phase difference between two external clock inputs is detected and TCNT is incremented/decremented accordingly. This mode can be set for channels 1 and 2.

When phase counting mode is set, an external clock is selected as the counter input clock and TCNT operates as an up/down-counter regardless of the setting of bits TPSC0 to TPSC2 and bits CKEG0 and CKEG1 in TCR. However, the functions of bits CCLR0 and CCLR1 in TCR, and of TIOR, TIER, and TGR, are valid, and input capture/compare match and interrupt functions can be used.

This can be used for two-phase encoder pulse input.

If overflow occurs when TCNT is counting up, the TCFV flag in TSR is set; if underflow occurs when TCNT is counting down, the TCFU flag is set.

The TCFD bit in TSR is the count direction flag. Reading the TCFD flag reveals whether TCNT is counting up or down.

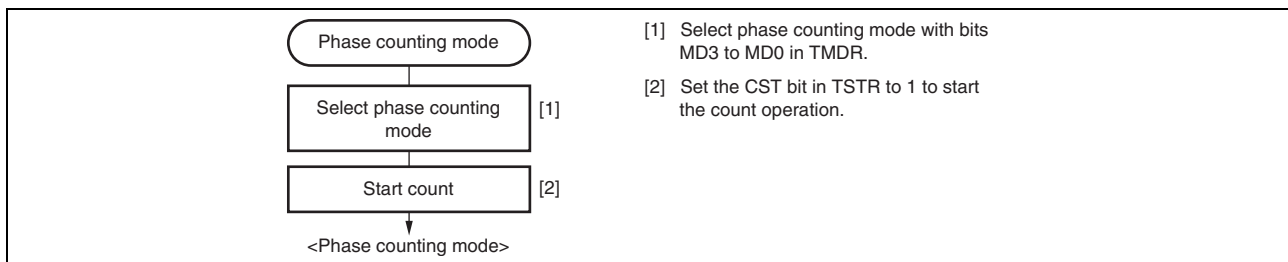
Table 10.45 shows the correspondence between external clock pins and channels.

**Table 10.45 Phase Counting Mode Clock Input Pins**

Channels	External Clock Pins	
	A-Phase	B-Phase
When channel 1 is set to phase counting mode	TCLKA	TCLKB
When channel 2 is set to phase counting mode	TCLKC	TCLKD

#### (1) Example of Phase Counting Mode Setting Procedure

Figure 10.29 shows an example of the phase counting mode setting procedure.



**Figure 10.29 Example of Phase Counting Mode Setting Procedure**

(2) Examples of Phase Counting Mode Operation

In phase counting mode, TCNT counts up or down according to the phase difference between two external clocks. There are four modes, according to the count conditions.

(a) Phase counting mode 1

Figure 10.30 shows an example of phase counting mode 1 operation, and Table 10.46 summarizes the TCNT up/down-count conditions.

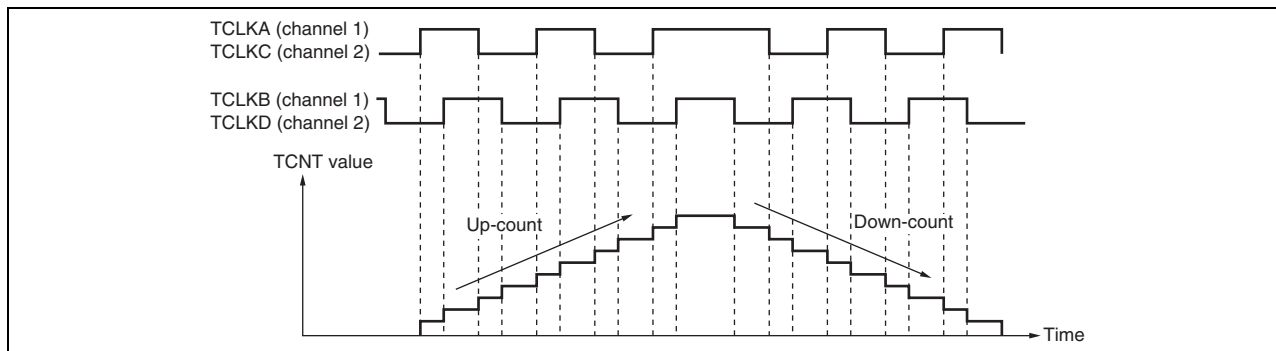


Figure 10.30 Example of Phase Counting Mode 1 Operation

Table 10.46 Up/Down-Count Conditions in Phase Counting Mode 1

TCLKA (Channel 1) TCLKC (Channel 2)	TCLKB (Channel 1) TCLKD (Channel 2)	Operation
High level		Up-count
Low level		
	Low level	
	High level	
High level		Down-count
Low level		
	High level	
	Low level	

[Legend]

- :Rising edge
- :Falling edge

(b) Phase counting mode 2

Figure 10.31 shows an example of phase counting mode 2 operation, and Table 10.47 summarizes the TCNT up/down-count conditions.

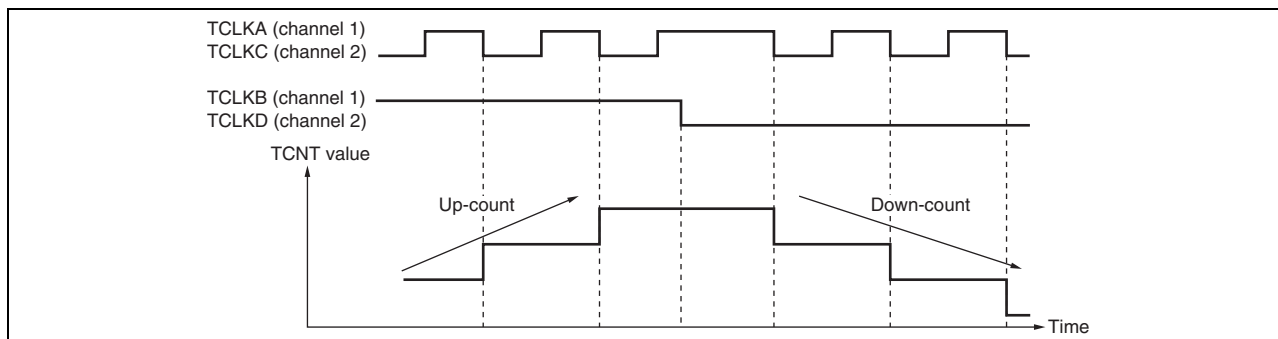


Figure 10.31 Example of Phase Counting Mode 2 Operation

Table 10.47 Up/Down-Count Conditions in Phase Counting Mode 2

TCLKA (Channel 1) TCLKC (Channel 2)	TCLKB (Channel 1) TCLKD (Channel 2)	Operation
High level		Don't care
Low level		Don't care
	Low level	Don't care
	High level	Up-count
High level		Don't care
Low level		Don't care
	High level	Don't care
	Low level	Down-count

[Legend]

- :Rising edge
- :Falling edge

(c) Phase counting mode 3

Figure 10.32 shows an example of phase counting mode 3 operation, and Table 10.48 summarizes the TCNT up/down-count conditions.

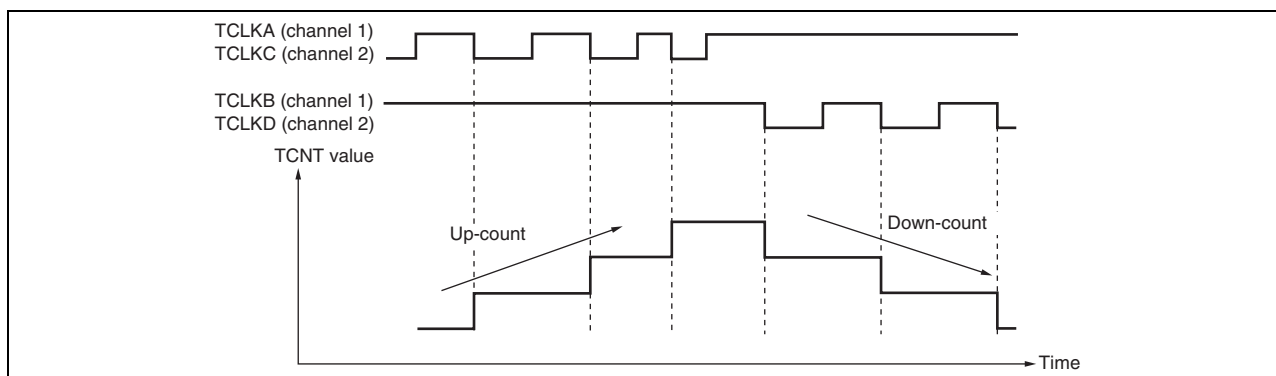


Figure 10.32 Example of Phase Counting Mode 3 Operation

Table 10.48 Up/Down-Count Conditions in Phase Counting Mode 3

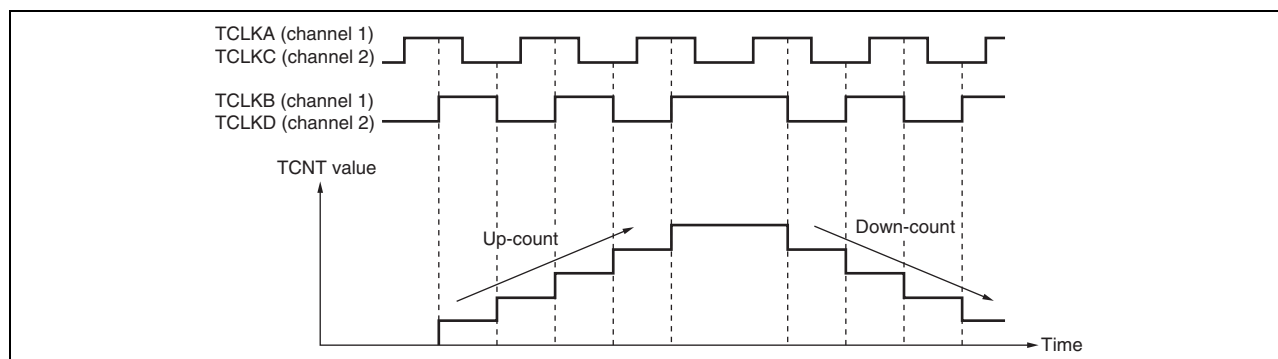
TCLKA (Channel 1) TCLKC (Channel 2)	TCLKB (Channel 1) TCLKD (Channel 2)	Operation
High level		Don't care
Low level		Don't care
	Low level	Don't care
	High level	Up-count
High level		Down-count
Low level		Don't care
	High level	Don't care
	Low level	Don't care

[Legend]

- :Rising edge
- :Falling edge

## (d) Phase counting mode 4

Figure 10.33 shows an example of phase counting mode 4 operation, and Table 10.49 summarizes the TCNT up/down-count conditions.



**Figure 10.33 Example of Phase Counting Mode 4 Operation**

**Table 10.49 Up/Down-Count Conditions in Phase Counting Mode 4**

TCLKA (Channel 1) TCLKC (Channel 2)	TCLKB (Channel 1) TCLKD (Channel 2)	Operation
High level		Up-count
Low level		Up-count
	Low level	Don't care
	High level	Don't care
High level		Down-count
Low level		Down-count
	High level	Don't care
	Low level	Don't care

[Legend]

:Rising edge  
 :Falling edge

### (3) Phase Counting Mode Application Example

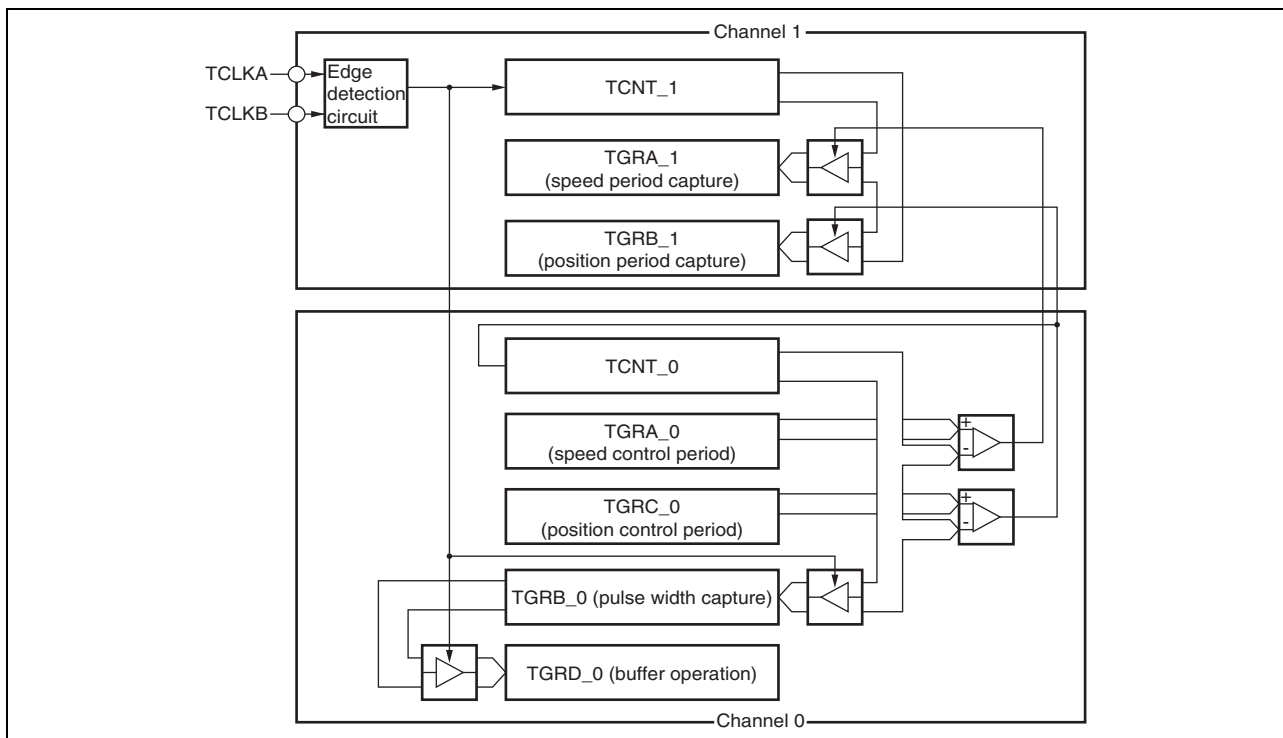
Figure 10.34 shows an example in which channel 1 is in phase counting mode, and channel 1 is coupled with channel 0 to input servo motor 2-phase encoder pulses in order to detect position or speed.

Channel 1 is set to phase counting mode 1, and the encoder pulse A-phase and B-phase are input to TCLKA and TCLKB.

Channel 0 operates with TCNT counter clearing by TGRC\_0 compare match; TGRA\_0 and TGRC\_0 are used for the compare match function and are set with the speed control period and position control period. TGRB\_0 is used for input capture, with TGRB\_0 and TGRD\_0 operating in buffer mode. The channel 1 counter input clock is designated as the TGRB\_0 input capture source, and the pulse widths of 2-phase encoder 4-multiplication pulses are detected.

TGRA\_1 and TGRB\_1 for channel 1 are designated for input capture, and channel 0 TGRA\_0 and TGRC\_0 compare matches are selected as the input capture source and store the up/down-counter values for the control periods.

This procedure enables the accurate detection of position and speed.



**Figure 10.34 Phase Counting Mode Application Example**



### 10.4.7 Reset-Synchronized PWM Mode

In the reset-synchronized PWM mode, three-phase output of positive and negative PWM waveforms that share a common wave transition point can be obtained by combining channels 3 and 4.

When set for reset-synchronized PWM mode, the TIOC3B, TIOC3D, TIOC4A, TIOC4C, TIOC4B, and TIOC4D pins function as PWM output pins and TCNT3 functions as an upcounter.

Table 10.50 shows the PWM output pins used. Table 10.51 shows the settings of the registers.

**Table 10.50 Output Pins for Reset-Synchronized PWM Mode**

Channel	Output Pin	Description
3	TIOC3B	PWM output pin 1
	TIOC3D	PWM output pin 1' (negative-phase waveform of PWM output 1)
4	TIOC4A	PWM output pin 2
	TIOC4C	PWM output pin 2' (negative-phase waveform of PWM output 2)
	TIOC4B	PWM output pin 3
	TIOC4D	PWM output pin 3' (negative-phase waveform of PWM output 3)

**Table 10.51 Register Settings for Reset-Synchronized PWM Mode**

Register	Description of Setting
TCNT_3	Initial setting of H'0000
TCNT_4	Initial setting of H'0000
TGRA_3	Set count cycle for TCNT_3
TGRB_3	Sets the turning point for PWM waveform output by the TIOC3B and TIOC3D pins
TGRA_4	Sets the turning point for PWM waveform output by the TIOC4A and TIOC4C pins
TGRB_4	Sets the turning point for PWM waveform output by the TIOC4B and TIOC4D pins

## (1) Procedure for Selecting the Reset-Synchronized PWM Mode

Figure 10.35 shows an example of procedure for selecting the reset synchronized PWM mode.

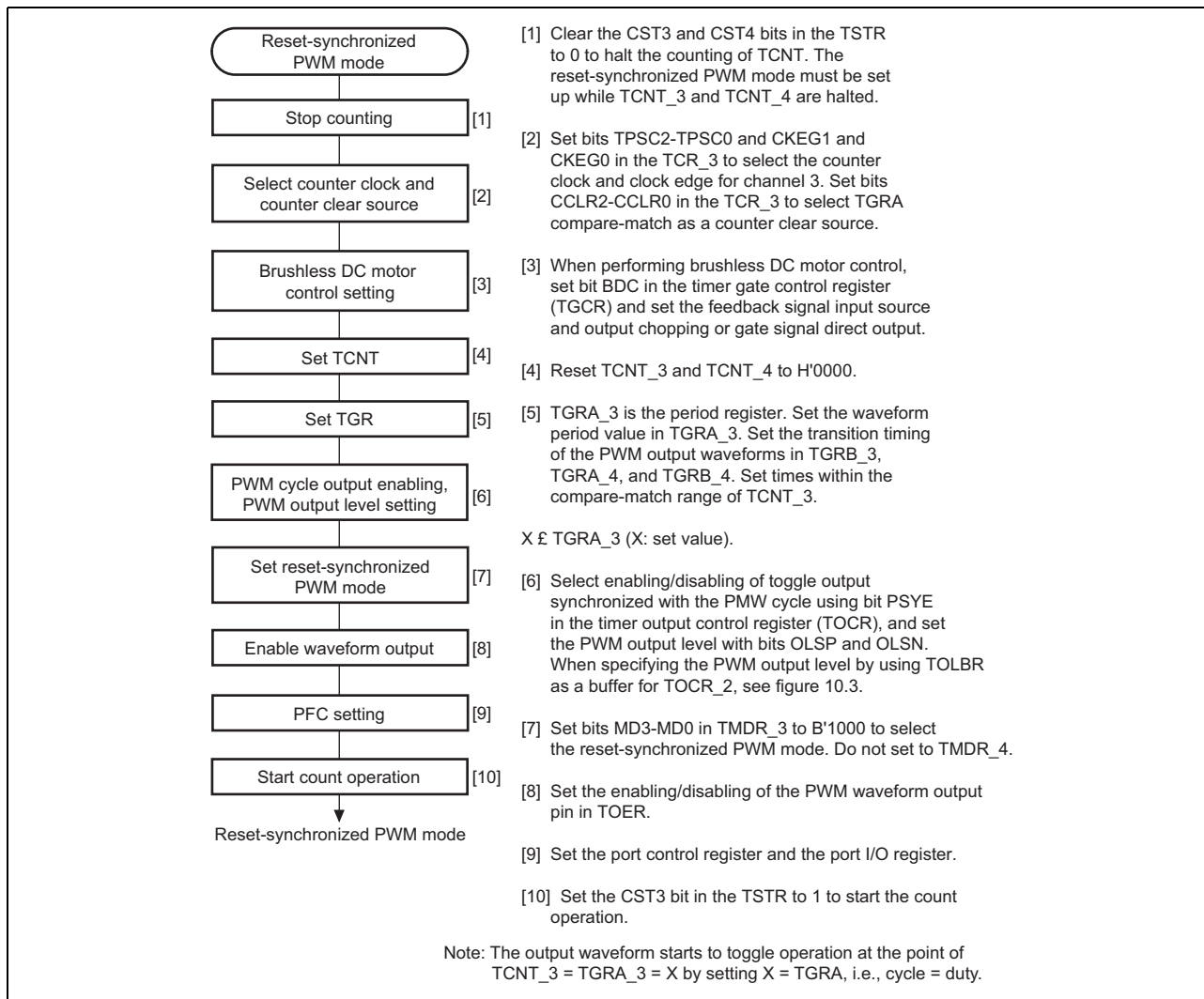


Figure 10.35 Procedure for Selecting Reset-Synchronized PWM Mode

(2) Reset-Synchronized PWM Mode Operation

Figure 10.36 shows an example of operation in the reset-synchronized PWM mode. TCNT\_3 and TCNT\_4 operate as upcounters. The counter is cleared when a TCNT\_3 and TGRA\_3 compare-match occurs, and then begins incrementing from H'0000. The PWM output pin output toggles with each occurrence of a TGRB\_3, TGRA\_4, TGRB\_4 compare-match, and upon counter clears.

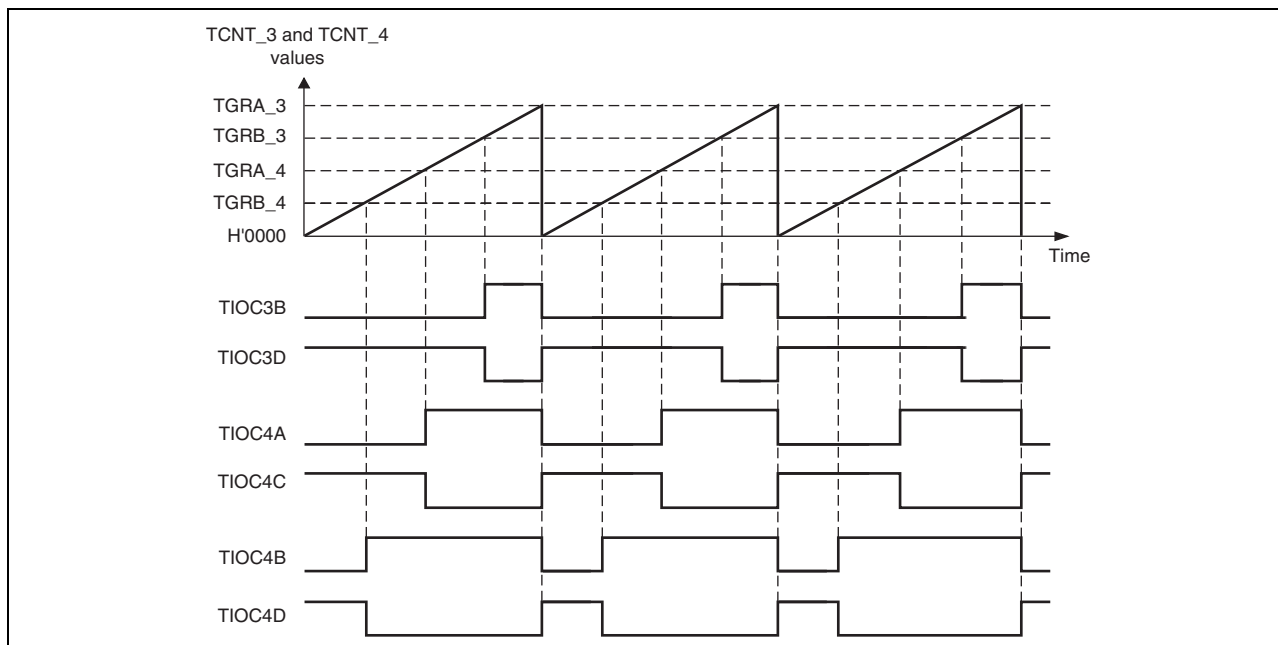


Figure 10.36 Reset-Synchronized PWM Mode Operation Example (When TOCR's OLSN = 1 and OLSP = 1)

### 10.4.8 Complementary PWM Mode

In the complementary PWM mode, three-phase output of non-overlapping positive and negative PWM waveforms can be obtained by combining channels 3 and 4. PWM waveforms without non-overlapping interval are also available.

In complementary PWM mode, TIOC3B, TIOC3D, TIOC4A, TIOC4B, TIOC4C, and TIOC4D pins function as PWM output pins, the TIOC3A pin can be set for toggle output synchronized with the PWM period. TCNT\_3 and TCNT\_4 function as up/down counters.

Table 10.52 shows the PWM output pins used. Table 10.53 shows the settings of the registers used.

**Table 10.52 Output Pins for Complementary PWM Mode**

Channel	Output Pin	Description
3	TIOC3A	Toggle output synchronized with PWM period (or I/O port)
	TIOC3B	PWM output pin 1
	TIOC3C	I/O port*
	TIOC3D	PWM output pin 1' (non-overlapping negative-phase waveform of PWM output 1; PWM output without non-overlapping interval is also available)
4	TIOC4A	PWM output pin 2
	TIOC4B	PWM output pin 3
	TIOC4C	PWM output pin 2' (non-overlapping negative-phase waveform of PWM output 2; PWM output without non-overlapping interval is also available)
	TIOC4D	PWM output pin 3' (non-overlapping negative-phase waveform of PWM output 3; PWM output without non-overlapping interval is also available)

Note: \* Avoid setting the TIOC3C pin as a timer I/O pin in the complementary PWM mode.

**Table 10.53 Register Settings for Complementary PWM Mode**

Channel	Counter/Register	Description	Read/Write from CPU
3	TCNT_3	Start of up-count from value set in dead time register	Maskable by TRWER setting*
	TGRA_3	Set TCNT_3 upper limit value (1/2 carrier cycle + dead time)	Maskable by TRWER setting*
	TGRB_3	PWM output 1 compare register	Maskable by TRWER setting*
	TGRC_3	TGRA_3 buffer register	Always readable/writable
	TGRD_3	PWM output 1/TGRB_3 buffer register	Always readable/writable
4	TCNT_4	Up-count start, initialized to H'0000	Maskable by TRWER setting*
	TGRA_4	PWM output 2 compare register	Maskable by TRWER setting*
	TGRB_4	PWM output 3 compare register	Maskable by TRWER setting*
	TGRC_4	PWM output 2/TGRA_4 buffer register	Always readable/writable
	TGRD_4	PWM output 3/TGRB_4 buffer register	Always readable/writable
Timer dead time data register (TDDR)		Set TCNT_4 and TCNT_3 offset value (dead time value)	Maskable by TRWER setting*
Timer cycle data register (TCDR)		Set TCNT_4 upper limit value (1/2 carrier cycle)	Maskable by TRWER setting*
Timer cycle buffer register (TCBR)		TCDR buffer register	Always readable/writable
Subcounter (TCNTS)		Subcounter for dead time generation	Read-only
Temporary register 1 (TEMP1)		PWM output 1/TGRB_3 temporary register	Not readable/writable
Temporary register 2 (TEMP2)		PWM output 2/TGRA_4 temporary register	Not readable/writable
Temporary register 3 (TEMP3)		PWM output 3/TGRB_4 temporary register	Not readable/writable

Note: \* Access can be enabled or disabled according to the setting of bit 0 (RWE) in TRWER (timer read/write enable register).

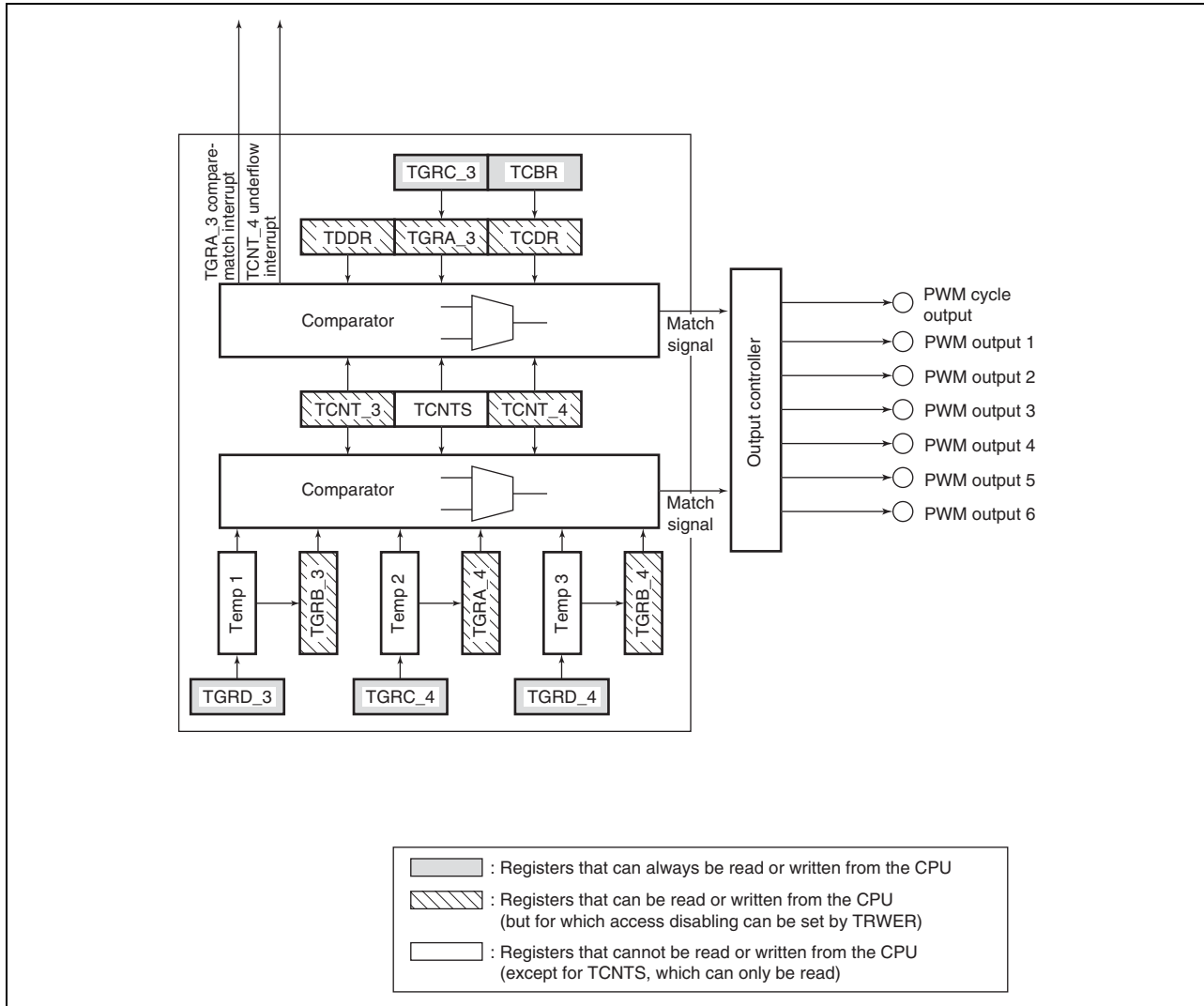


Figure 10.37 Block Diagram of Channels 3 and 4 in Complementary PWM Mode

(1) Example of Complementary PWM Mode Setting Procedure

An example of the complementary PWM mode setting procedure is shown in Figure 10.38.

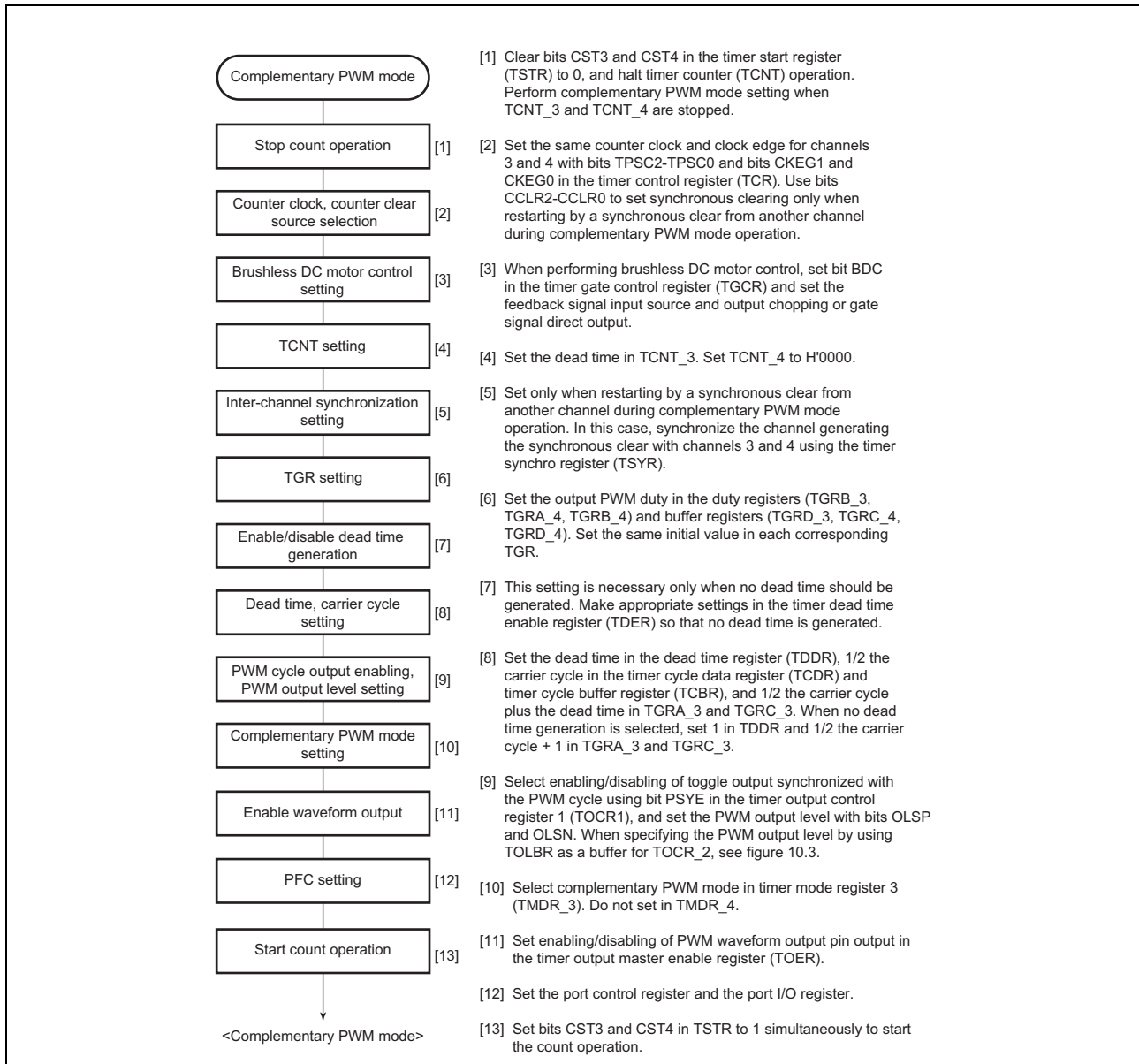


Figure 10.38 Example of Complementary PWM Mode Setting Procedure

## (2) Outline of Complementary PWM Mode Operation

In complementary PWM mode, 6-phase PWM output is possible. Figure 10.39 illustrates counter operation in complementary PWM mode, and Figure 10.40 shows an example of complementary PWM mode operation.

### (a) Counter Operation

In complementary PWM mode, three counters—TCNT\_3, TCNT\_4, and TCNTS—perform up/down-count operations.

TCNT\_3 is automatically initialized to the value set in TDDR when complementary PWM mode is selected and the CST bit in TSTR is 0.

When the CST bit is set to 1, TCNT\_3 counts up to the value set in TGRA\_3, then switches to down-counting when it matches TGRA\_3. When the TCNT3 value matches TDDR, the counter switches to up-counting, and the operation is repeated in this way.

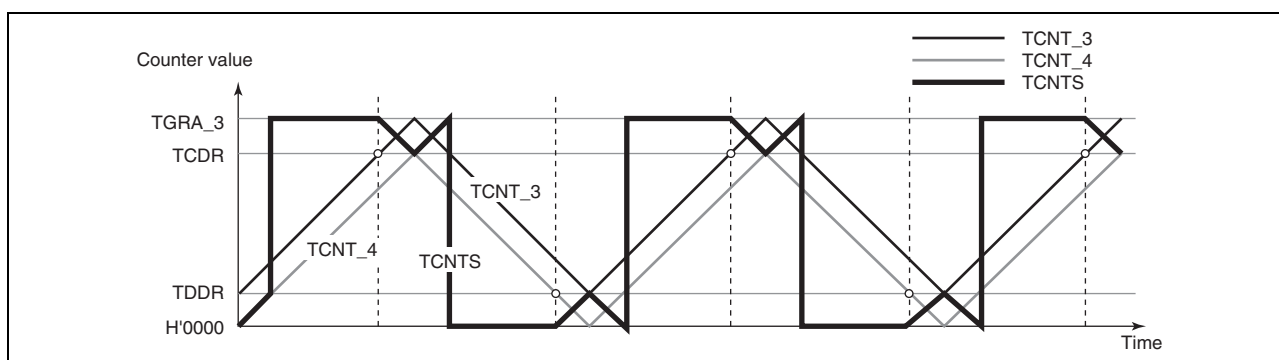
TCNT\_4 is initialized to H'0000.

When the CST bit is set to 1, TCNT4 counts up in synchronization with TCNT\_3, and switches to down-counting when it matches TCDR. On reaching H'0000, TCNT4 switches to up-counting, and the operation is repeated in this way.

TCNTS is a read-only counter. It need not be initialized.

When TCNT\_3 matches TCDR during TCNT\_3 and TCNT\_4 up/down-counting, down-counting is started, and when TCNTS matches TCDR, the operation switches to up-counting. When TCNTS matches TGRA\_3, it is cleared to H'0000. When TCNT\_4 matches TDDR during TCNT\_3 and TCNT\_4 down-counting, up-counting is started, and when TCNTS matches TDDR, the operation switches to down-counting. When TCNTS reaches H'0000, it is set with the value in TGRA\_3.

TCNTS is compared with the compare register and temporary register in which the PWM duty is set during the count operation only.



**Figure 10.39** Complementary PWM Mode Counter Operation



### (b) Register Operation

In complementary PWM mode, nine registers are used, comprising compare registers, buffer registers, and temporary registers. Figure 10.40 shows an example of complementary PWM mode operation.

The registers which are constantly compared with the counters to perform PWM output are TGRB\_3, TGRA\_4, and TGRB\_4. When these registers match the counter, the value set in bits OLSN and OLSP in the timer output control register (TOCR) is output.

The buffer registers for these compare registers are TGRD\_3, TGRC\_4, and TGRD\_4.

Between a buffer register and compare register there is a temporary register. The temporary registers cannot be accessed by the CPU.

Data in a compare register is changed by writing the new data to the corresponding buffer register. The buffer registers can be read or written at any time.

The data written to a buffer register is constantly transferred to the temporary register in the Ta interval. Data is not transferred to the temporary register in the Tb interval. Data written to a buffer register in this interval is transferred to the temporary register at the end of the Tb interval.

The value transferred to a temporary register is transferred to the compare register when TCNTS for which the Tb interval ends matches TGRA\_3 when counting up, or H'0000 when counting down. The timing for transfer from the temporary register to the compare register can be selected with bits MD3 to MD0 in the timer mode register (TMDR). Figure 10.40 shows an example in which the mode is selected in which the change is made in the trough.

In the tb interval (tb1 in Figure 10.40) in which data transfer to the temporary register is not performed, the temporary register has the same function as the compare register, and is compared with the counter. In this interval, therefore, there are two compare match registers for one-phase output, with the compare register containing the pre-change data, and the temporary register containing the new data. In this interval, the three counters—TCNT\_3, TCNT\_4, and TCNTS—and two registers—compare register and temporary register—are compared, and PWM output controlled accordingly.

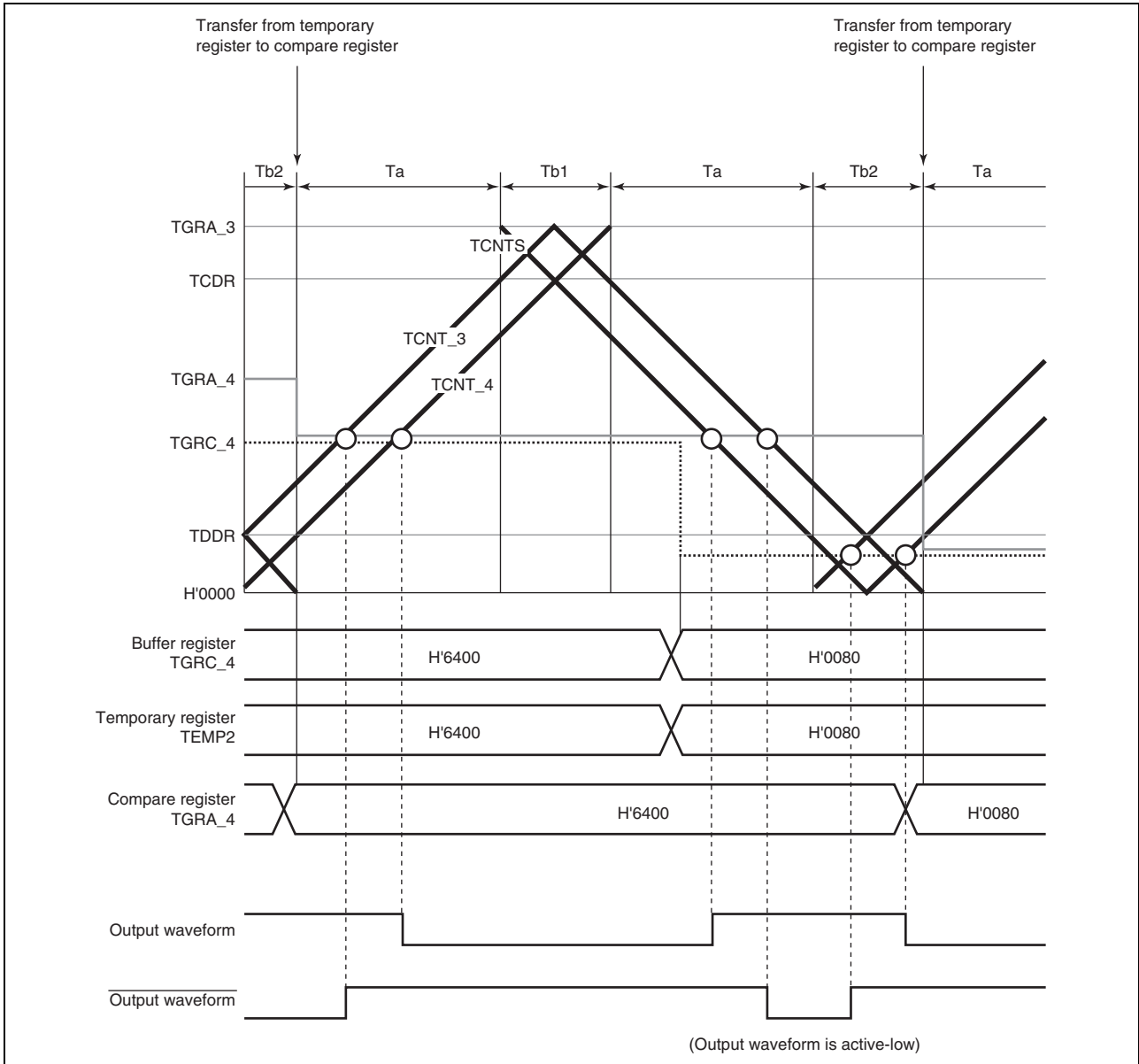


Figure 10.40 Example of Complementary PWM Mode Operation

**(c) Initialization**

In complementary PWM mode, there are six registers that must be initialized. In addition, there is a register that specifies whether to generate dead time (it should be used only when dead time generation should be disabled).

Before setting complementary PWM mode with bits MD3 to MD0 in the timer mode register (TMDR), the following initial register values must be set.

TGRC\_3 operates as the buffer register for TGRA\_3, and should be set with 1/2 the PWM carrier cycle + dead time Td. The timer cycle buffer register (TCBR) operates as the buffer register for the timer cycle data register (TCDR), and should be set with 1/2 the PWM carrier cycle. Set dead time Td in the timer dead time data register (TDDR).

When dead time is not needed, the TDER bit in the timer dead time enable register (TDER) should be cleared to 0, TGRC\_3 and TGRA\_3 should be set to 1/2 the PWM carrier cycle + 1, and TDDR should be set to 1.

Set the respective initial PWM duty values in buffer registers TGRD\_3, TGRC\_4, and TGRD\_4.

The values set in the five buffer registers excluding TDDR are transferred simultaneously to the corresponding compare registers when complementary PWM mode is set.

Set TCNT\_4 to H'0000 before setting complementary PWM mode.

**Table 10.54 Registers and Counters Requiring Initialization**

Register/Counter	Set Value
TGRC_3	1/2 PWM carrier cycle + dead time Td (1/2 PWM carrier cycle + 1 when dead time generation is disabled by TDER)
TDDR	Dead time Td (1 when dead time generation is disabled by TDER)
TCBR	1/2 PWM carrier cycle
TGRD_3, TGRC_4, TGRD_4	Initial PWM duty value for each phase
TCNT_4	H'0000

Note: The TGRC\_3 set value must be the sum of 1/2 the PWM carrier cycle set in TCBR and dead time Td set in TDDR. When dead time generation is disabled by TDER, TGRC\_3 must be set to 1/2 the PWM carrier cycle + 1.

#### (d) PWM Output Level Setting

In complementary PWM mode, the PWM pulse output level is set with bits OLSN and OLSP in timer output control register 1 (TOCR1) or bits OLS1P to OLS3P and OLS1N to OLS3N in timer output control register 2 (TOCR2).

The output level can be set for each of the three positive phases and three negative phases of 6-phase output.

Complementary PWM mode should be cleared before setting or changing output levels.

#### (e) Dead Time Setting

In complementary PWM mode, PWM pulses are output with a non-overlapping relationship between the positive and negative phases. This non-overlap time is called the dead time.

The non-overlap time is set in the timer dead time data register (TDDR). The value set in TDDR is used as the TCNT\_3 counter start value, and creates non-overlap between TCNT\_3 and TCNT\_4. Complementary PWM mode should be cleared before changing the contents of TDDR.

#### (f) Dead Time Suppressing

Dead time generation is suppressed by clearing the TDER bit in the timer dead time enable register (TDER) to 0. TDER can be cleared to 0 only when 0 is written to it after reading TDER = 1.

TGRA\_3 and TGRC\_3 should be set to 1/2 PWM carrier cycle + 1 and the timer dead time data register (TDDR) should be set to 1.

By the above settings, PWM waveforms without dead time can be obtained. Figure 10.41 shows an example of operation without dead time.

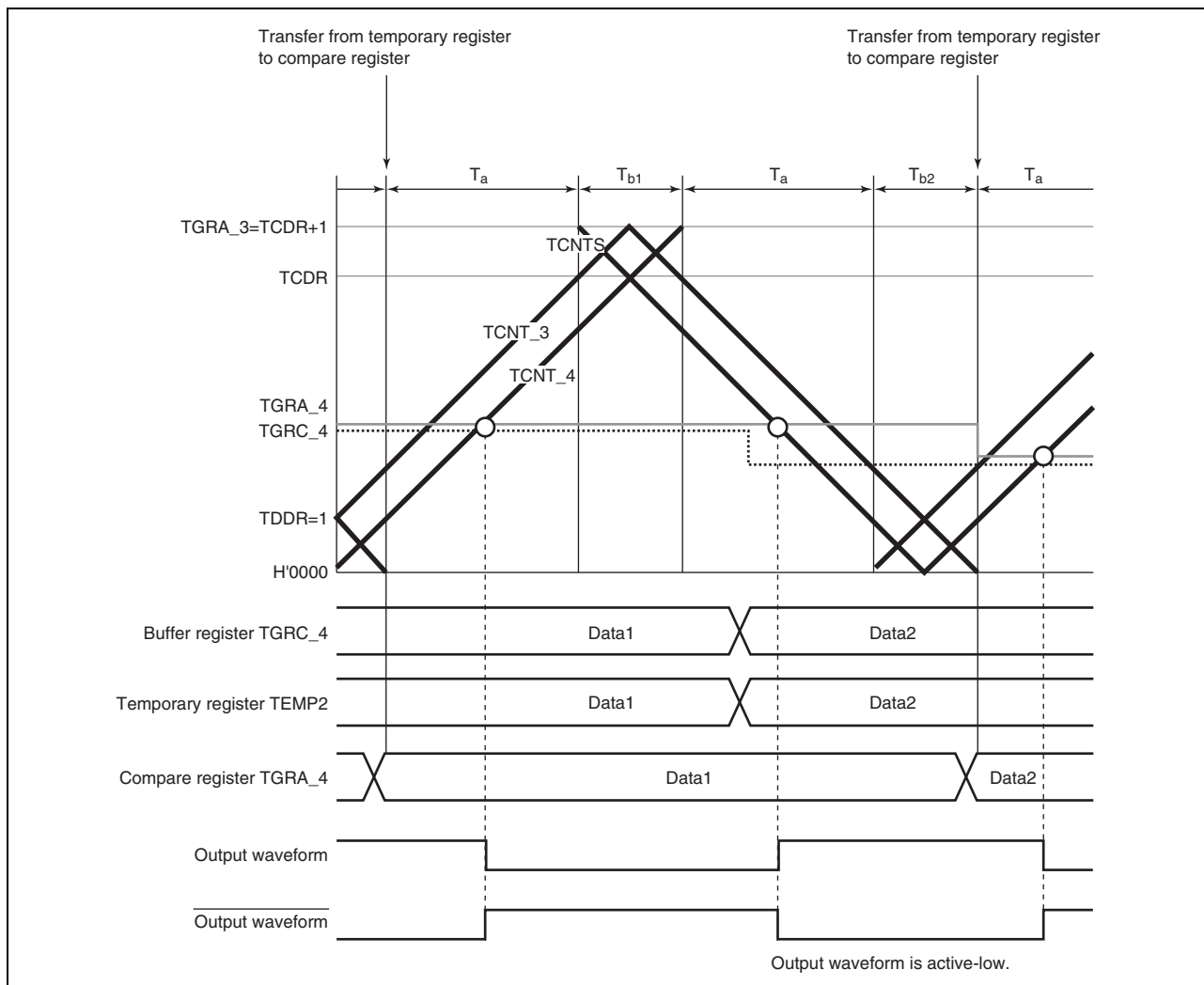


Figure 10.41 Example of Operation without Dead Time

**(g) PWM Cycle Setting**

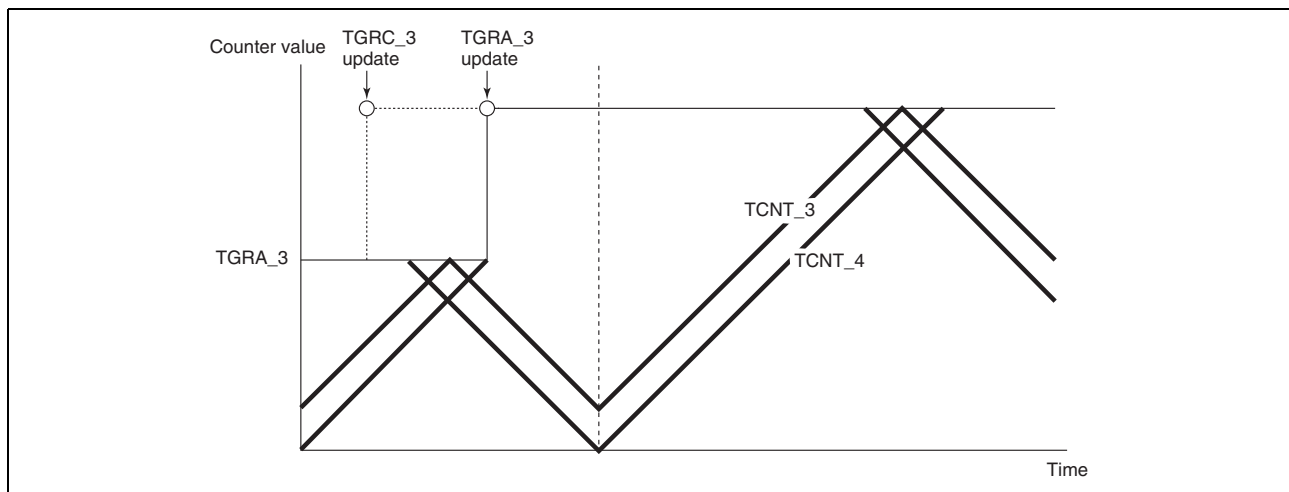
In complementary PWM mode, the PWM pulse cycle is set in two registers—TGRA\_3, in which the TCNT\_3 upper limit value is set, and TCDR, in which the TCNT\_4 upper limit value is set. The settings should be made so as to achieve the following relationship between these two registers:

$$\begin{aligned} \text{With dead time: } & \text{TGRA\_3 set value} = \text{TCDR set value} + \text{TDDR set value} \\ & \text{TCDR set value} > \text{Double the TDDR set value} + 2 \\ \text{Without dead time: } & \text{TGRA\_3 set value} = \text{TCDR set value} + 1 \\ & \text{TCDR set value} > 4 \end{aligned}$$

The TGRA\_3 and TCDR settings are made by setting the values in buffer registers TGRC\_3 and TCBR. The values set in TGRC\_3 and TCBR are transferred simultaneously to TGRA\_3 and TCDR in accordance with the transfer timing selected with bits MD3 to MD0 in the timer mode register (TMDR).

The updated PWM cycle is reflected from the next cycle when the data update is performed at the crest, and from the current cycle when performed in the trough. Figure 10.42 illustrates the operation when the PWM cycle is updated at the crest.

See (h) Register Data Updating, for the method of updating the data in each buffer register.



**Figure 10.42** Example of PWM Cycle Updating

### (h) Register Data Updating

In complementary PWM mode, the buffer register is used to update the data in a compare register. The update data can be written to the buffer register at any time. There are five PWM duty and carrier cycle registers that have buffer registers and can be updated during operation.

There is a temporary register between each of these registers and its buffer register. When subcounter TCNTS is not counting, if buffer register data is updated, the temporary register value is also rewritten. Transfer is not performed from buffer registers to temporary registers when TCNTS is counting; in this case, the value written to a buffer register is transferred after TCNTS halts.

The temporary register value is transferred to the compare register at the data update timing set with bits MD3 to MD0 in the timer mode register (TMDR). **Figure 10.43** shows an example of data updating in complementary PWM mode. This example shows the mode in which data updating is performed at both the counter crest and trough.

When rewriting buffer register data, a write to TGRD\_4 must be performed at the end of the update. Data transfer from the buffer registers to the temporary registers is performed simultaneously for all five registers after the write to TGRD\_4.

A write to TGRD\_4 must be performed after writing data to the registers to be updated, even when not updating all five registers, or when updating the TGRD\_4 data. In this case, the data written to TGRD\_4 should be the same as the data prior to the write operation.

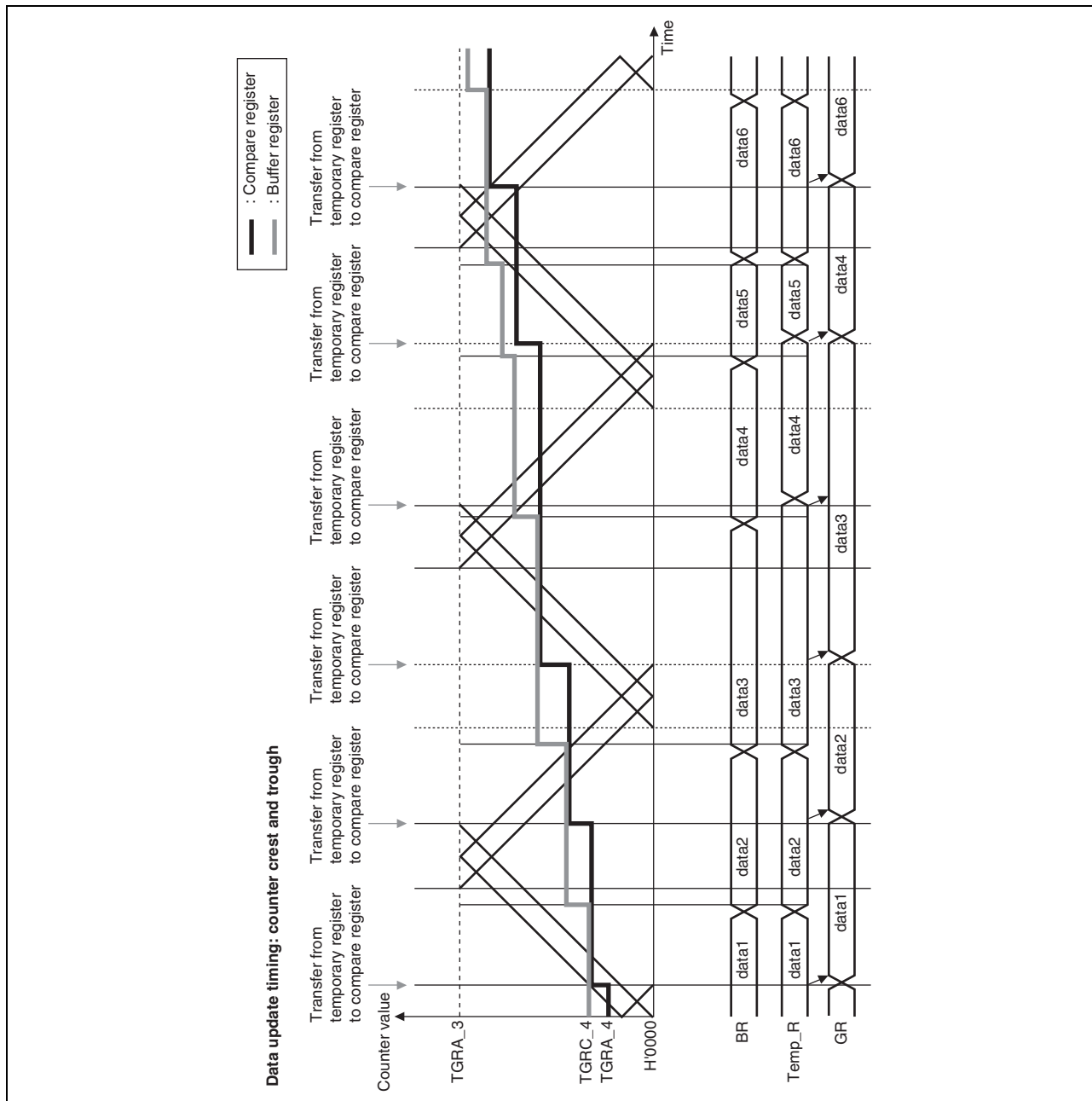


Figure 10.43 Example of Data Update in Complementary PWM Mode



## (i) Initial Output in Complementary PWM Mode

In complementary PWM mode, the initial output is determined by the setting of bits OLSN and OLSP in timer output control register 1 (TOCR1) or bits OLS1N to OLS3N and OLS1P to OLS3P in timer output control register 2 (TOCR2).

This initial output is the PWM pulse non-active level, and is output from when complementary PWM mode is set with the timer mode register (TMDR) until TCNT\_4 exceeds the value set in the dead time register (TDDR). Figure 10.44 shows an example of the initial output in complementary PWM mode.

An example of the waveform when the initial PWM duty value is smaller than the TDDR value is shown in Figure 10.45.

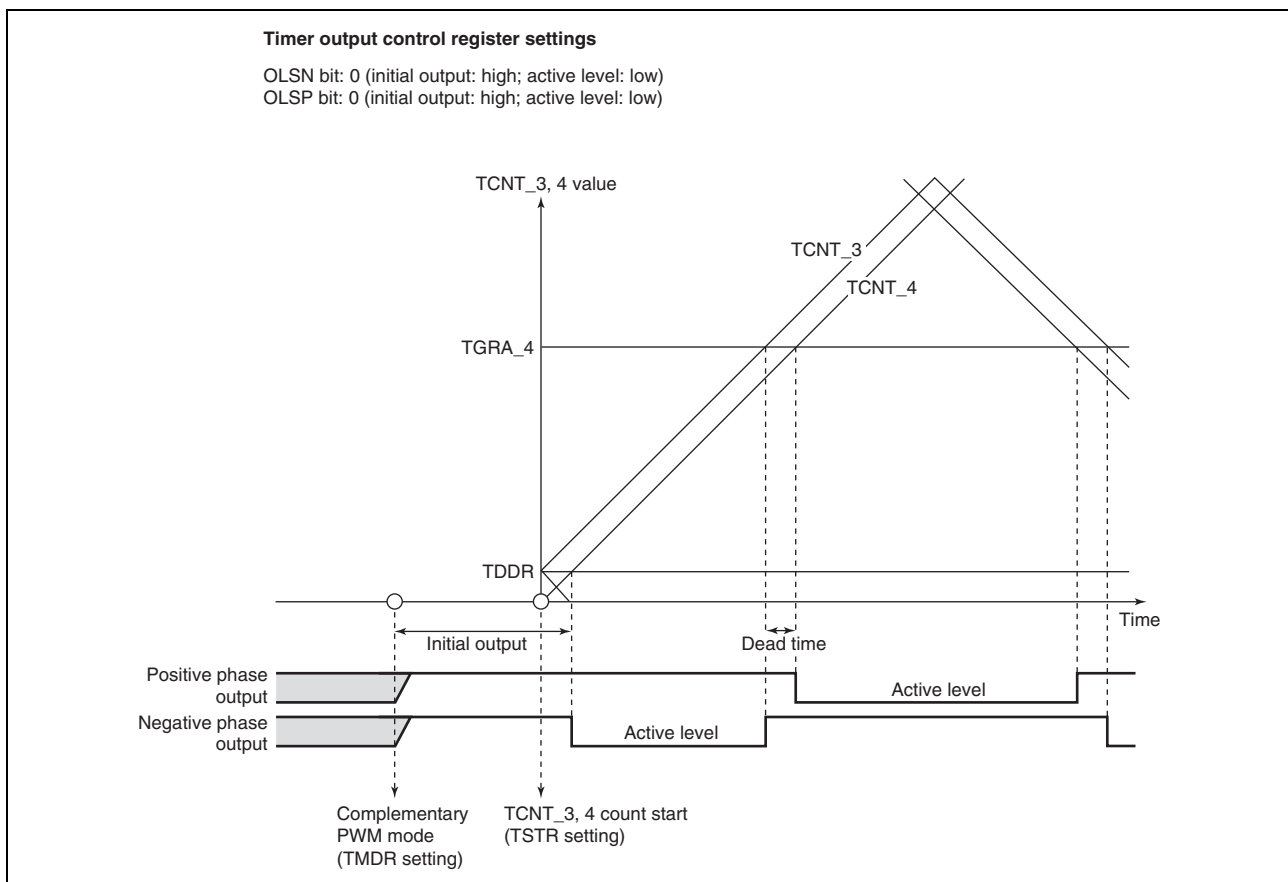
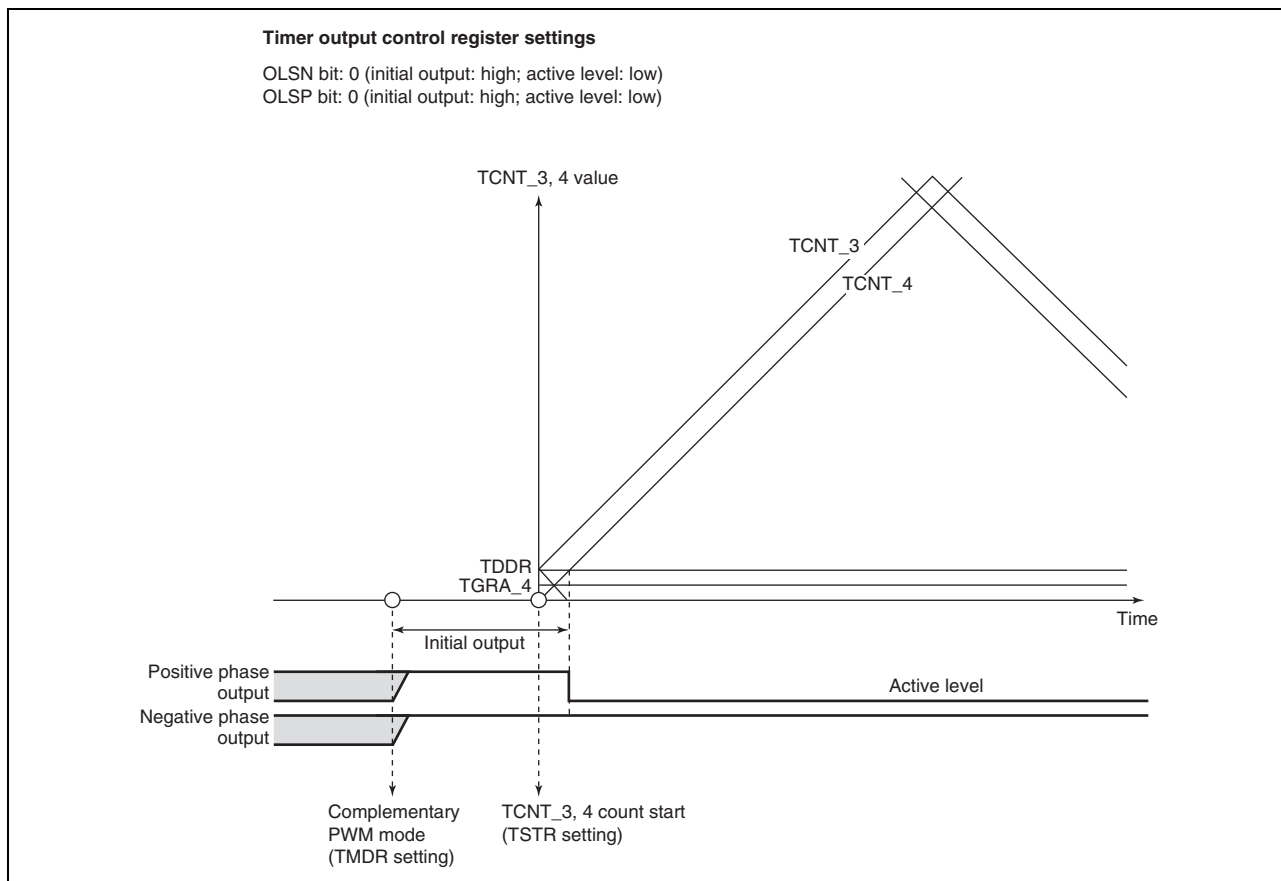


Figure 10.44 Example of Initial Output in Complementary PWM Mode (1)



**Figure 10.45 Example of Initial Output in Complementary PWM Mode (2)**

### (j) Complementary PWM Mode PWM Output Generation Method

In complementary PWM mode, 3-phase output is performed of PWM waveforms with a non-overlap time between the positive and negative phases. This non-overlap time is called the dead time.

A PWM waveform is generated by output of the output level selected in the timer output control register in the event of a compare-match between a counter and compare register. While TCNTS is counting, compare register and temporary register values are simultaneously compared to create consecutive PWM pulses from 0 to 100%. The relative timing of on and off compare-match occurrence may vary, but the compare-match that turns off each phase takes precedence to secure the dead time and ensure that the positive phase and negative phase on times do not overlap. Figure 10.46 to Figure 10.48 show examples of waveform generation in complementary PWM mode.

The positive phase/negative phase off timing is generated by a compare-match with the solid-line counter, and the on timing by a compare-match with the dotted-line counter operating with a delay of the dead time behind the solid-line counter. In the T1 period, compare-match **a** that turns off the negative phase has the highest priority, and compare-matches occurring prior to **a** are ignored. In the T2 period, compare-match **c** that turns off the positive phase has the highest priority, and compare-matches occurring prior to **c** are ignored.

In normal cases, compare-matches occur in the order **a** → **b** → **c** → **d** (or **c** → **d** → **a'** → **b'**), as shown in figure 10.46.

If compare-matches deviate from the **a** → **b** → **c** → **d** order, since the time for which the negative phase is off is less than twice the dead time, the figure shows the positive phase is not being turned on. If compare-matches deviate from the **c** → **d** → **a'** → **b'** order, since the time for which the positive phase is off is less than twice the dead time, the figure shows the negative phase is not being turned on.

If compare-match **c** occurs first following compare-match **a**, as shown in Figure 10.47, compare-match **b** is ignored, and the negative phase is turned on by compare-match **d**. This is because turning off of the positive phase has priority due to the occurrence of compare-match **c** (positive phase off timing) before compare-match **b** (positive phase on timing) (consequently, the waveform does not change since the positive phase goes from off to off).

Similarly, in the example in Figure 10.48, compare-match **a'** with the new data in the temporary register occurs before compare-match **c**, but other compare-matches occurring up to **c**, which turns off the positive phase, are ignored. As a result, the negative phase is not turned on.

Thus, in complementary PWM mode, compare-matches at turn-off timings take precedence, and turn-on timing compare-matches that occur before a turn-off timing compare-match are ignored.

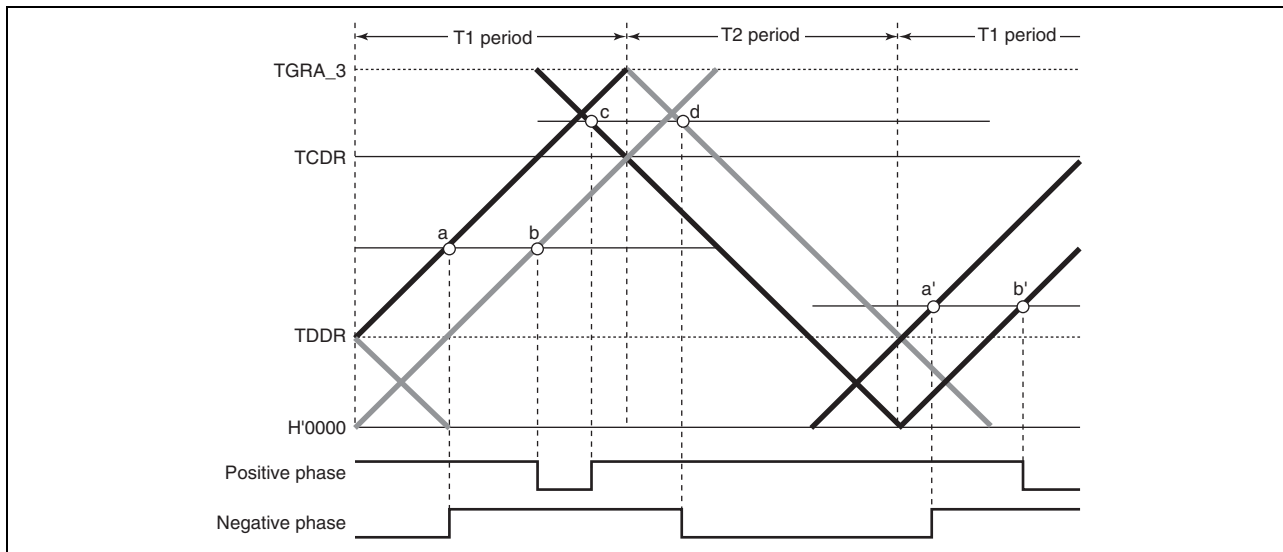


Figure 10.46 Example of Complementary PWM Mode Waveform Output (1)

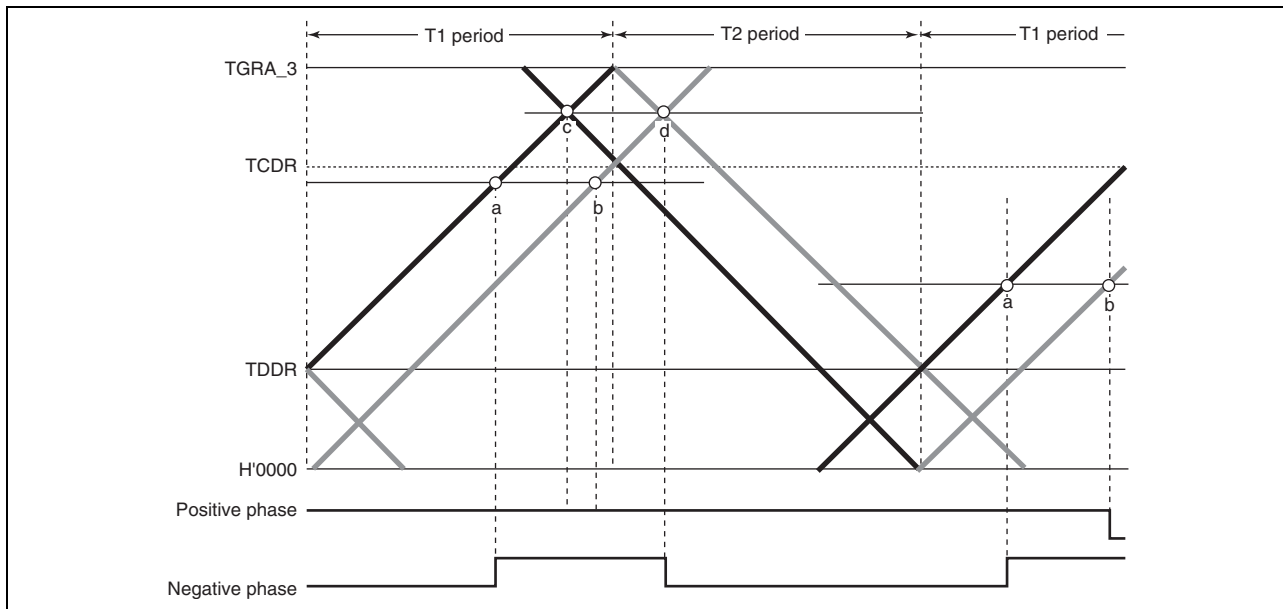


Figure 10.47 Example of Complementary PWM Mode Waveform Output (2)

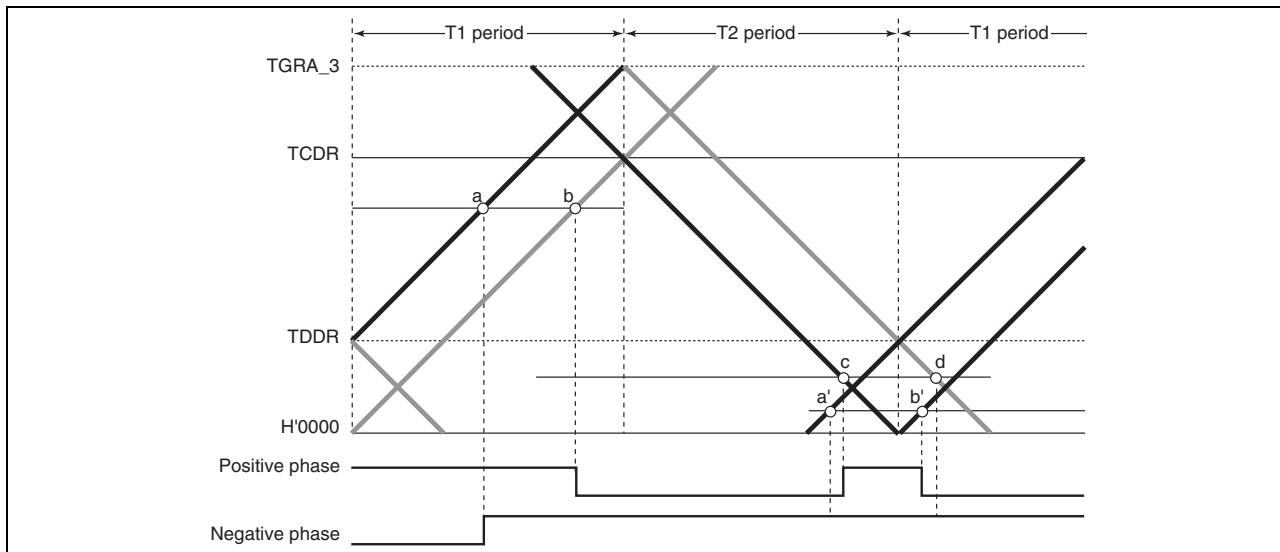


Figure 10.48 Example of Complementary PWM Mode Waveform Output (3)

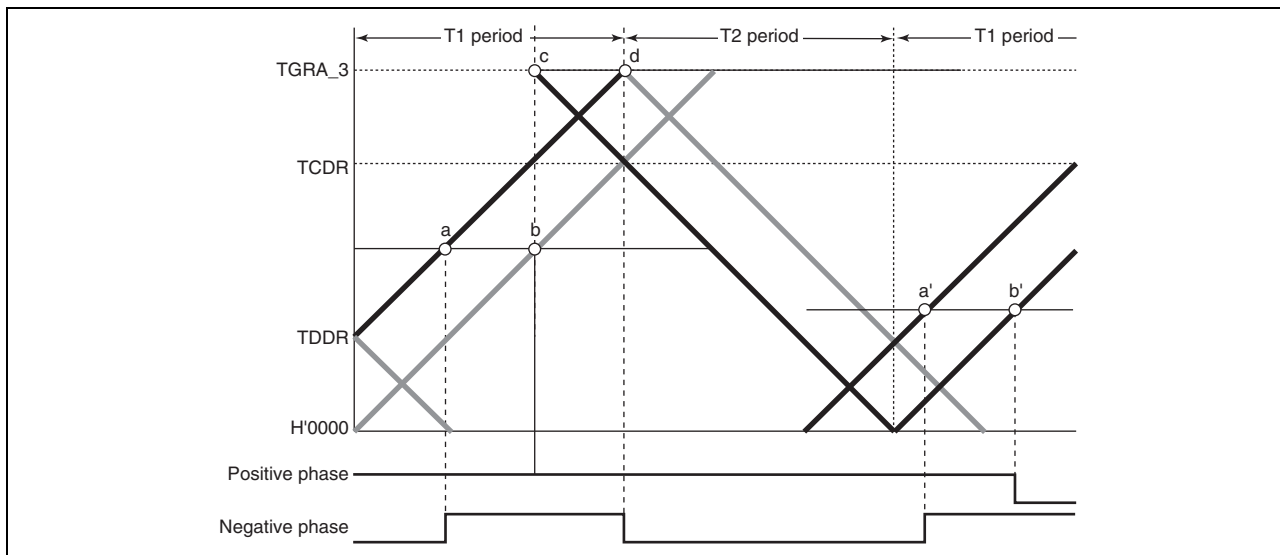


Figure 10.49 Example of Complementary PWM Mode 0% and 100% Waveform Output (1)

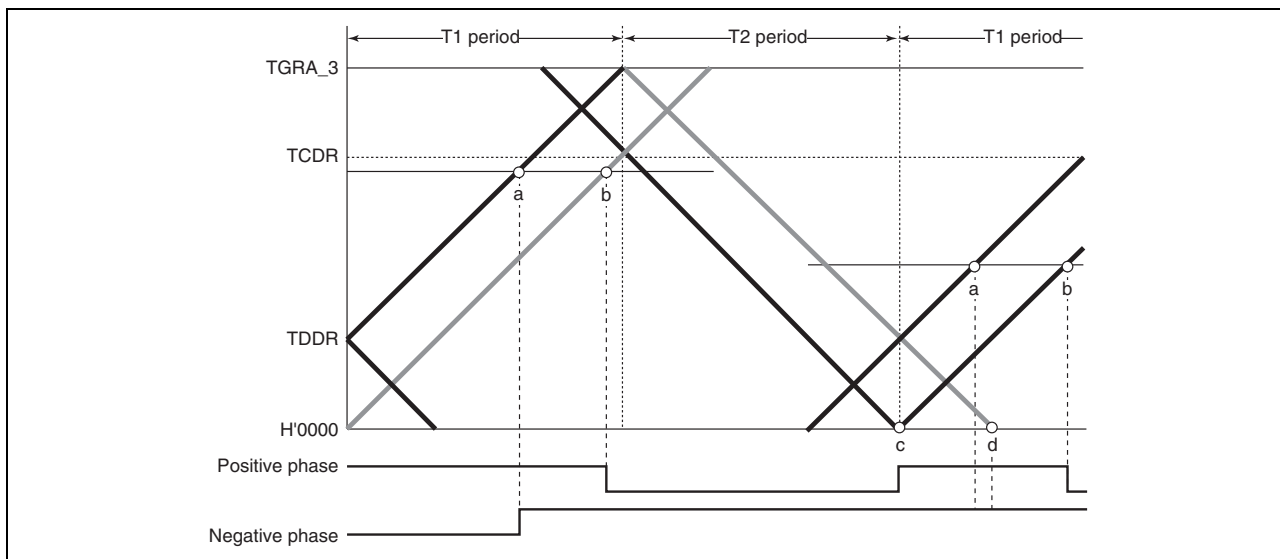


Figure 10.50 Example of Complementary PWM Mode 0% and 100% Waveform Output (2)

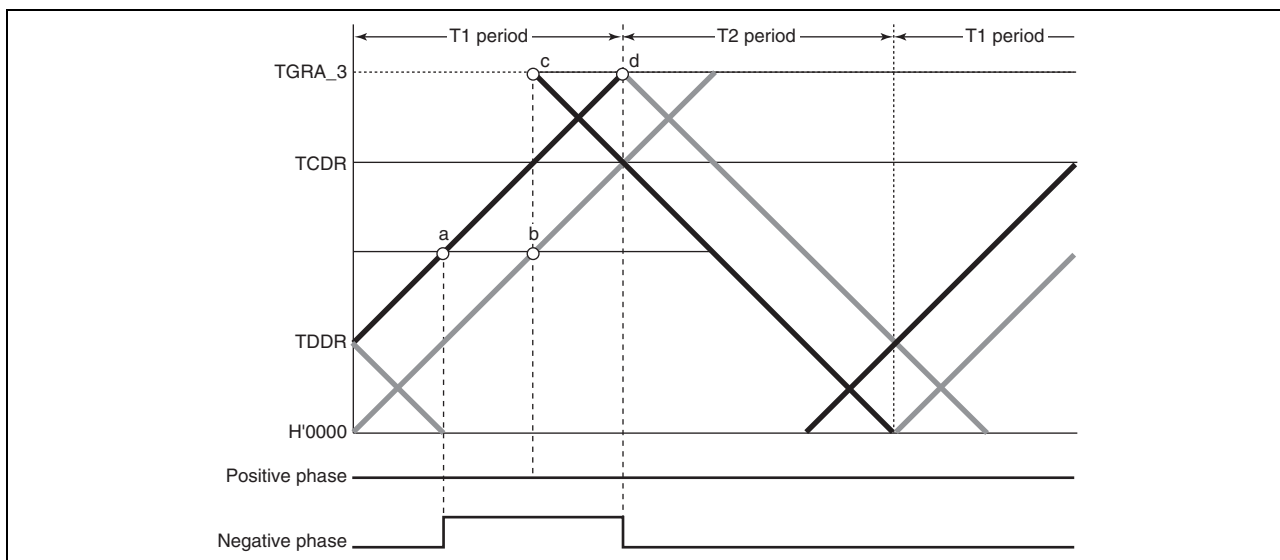


Figure 10.51 Example of Complementary PWM Mode 0% and 100% Waveform Output (3)

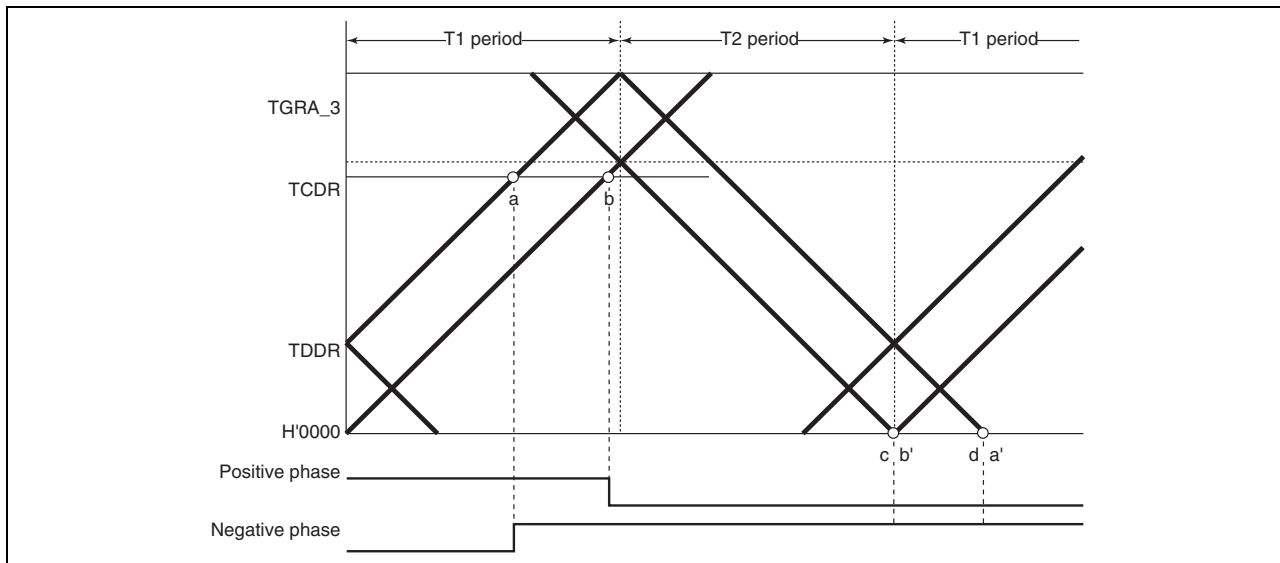


Figure 10.52 Example of Complementary PWM Mode 0% and 100% Waveform Output (4)

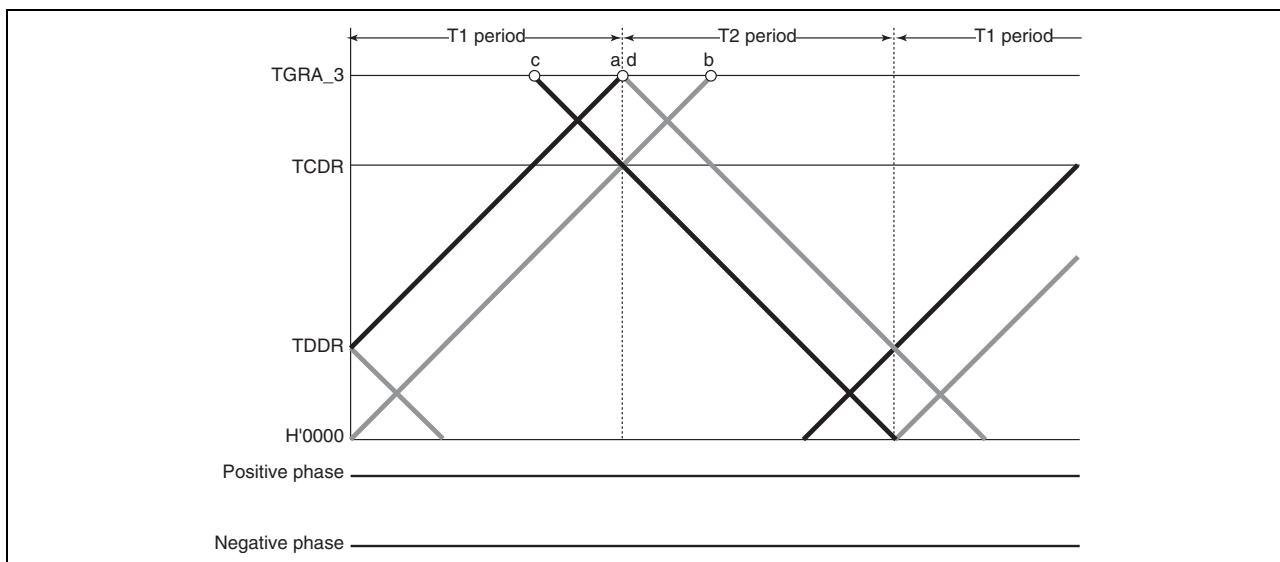


Figure 10.53 Example of Complementary PWM Mode 0% and 100% Waveform Output (5)

### (k) Complementary PWM Mode 0% and 100% Duty Output

In complementary PWM mode, 0% and 100% duty cycles can be output as required. Figure 10.49 to Figure 10.53 show output examples.

100% duty output is performed when the compare register value is set to H'0000. The waveform in this case has a positive phase with a 100% on-state. 0% duty output is performed when the compare register value is set to the same value as TGRA\_3. The waveform in this case has a positive phase with a 100% off-state.

On and off compare-matches occur simultaneously, but if a turn-on compare-match and turn-off compare-match for the same phase occur simultaneously, both compare-matches are ignored and the waveform does not change.

### (l) Toggle Output Synchronized with PWM Cycle

In complementary PWM mode, toggle output can be performed in synchronization with the PWM carrier cycle by setting the PSYE bit to 1 in the timer output control register (TOCR). An example of a toggle output waveform is shown in Figure 10.54.

This output is toggled by a compare-match between TCNT\_3 and TGRA\_3 and a compare-match between TCNT4 and H'0000.

The output pin for this toggle output is the TIOC3A pin. The initial output is 1.

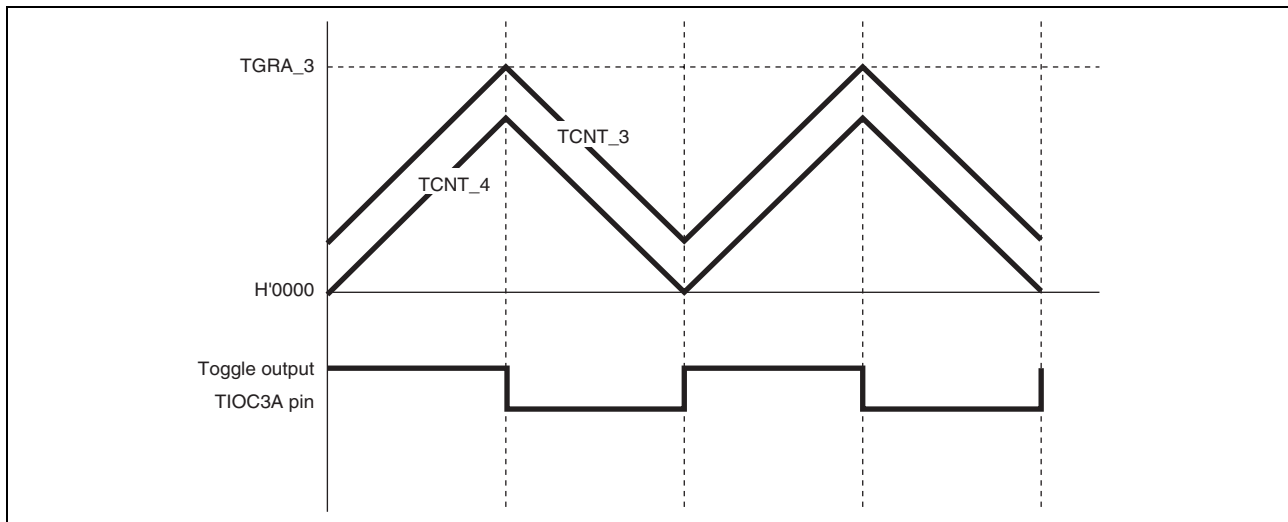


Figure 10.54 Example of Toggle Output Waveform Synchronized with PWM Output



(m) Counter Clearing by Another Channel

In complementary PWM mode, by setting a mode for synchronization with another channel by means of the timer synchronous register (TSYR), and selecting synchronous clearing with bits CCLR2 to CCLR0 in the timer control register (TCR), it is possible to have TCNT\_3, TCNT\_4, and TCNTS cleared by another channel.

Figure 10.55 illustrates the operation.

Use of this function enables counter clearing and restarting to be performed by means of an external signal.

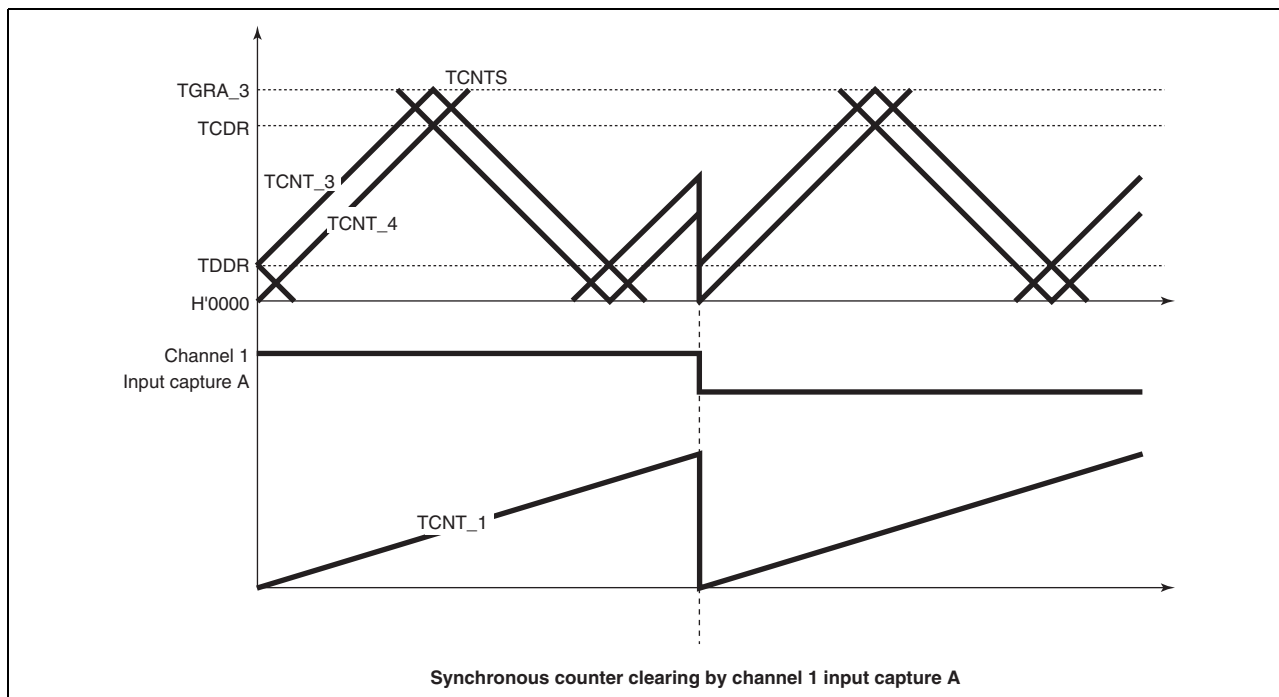


Figure 10.55 Counter Clearing Synchronized with Another Channel

(n) Output Waveform Control at Synchronous Counter Clearing in Complementary PWM Mode

Setting the WRE bit in TWCR to 1 suppresses initial output when synchronous counter clearing occurs in the  $T_b$  interval at the trough in complementary PWM mode and controls abrupt change in duty cycle at synchronous counter clearing.

Initial output suppression is applicable only when synchronous clearing occurs in the  $T_b$  interval at the trough as indicated by (10) or (11) in Figure 10.56. When synchronous clearing occurs outside that interval, the initial value specified by the OLS bits in TOCR is output. Even in the  $T_b$  interval at the trough, if synchronous clearing occurs in the initial value output period (indicated by (1) in Figure 10.56) immediately after the counters start operation, initial value output is not suppressed.

When using the initial output suppression function, make sure to set compare registers TGRB\_3, TGRA\_4, and TGRB\_4 to a value twice or more the setting of dead time data register TDDR. If synchronous clearing occurs with the compare registers set to a value less than twice the setting of TDDR, the PWM output dead time may be too short (or nonexistent) or illegal active-level PWM negative-phase output may occur during the initial output suppression interval. For details, see section 10.7.23, Notes on Output Waveform Control During Synchronous Counter Clearing in Complementary PWM Mode.

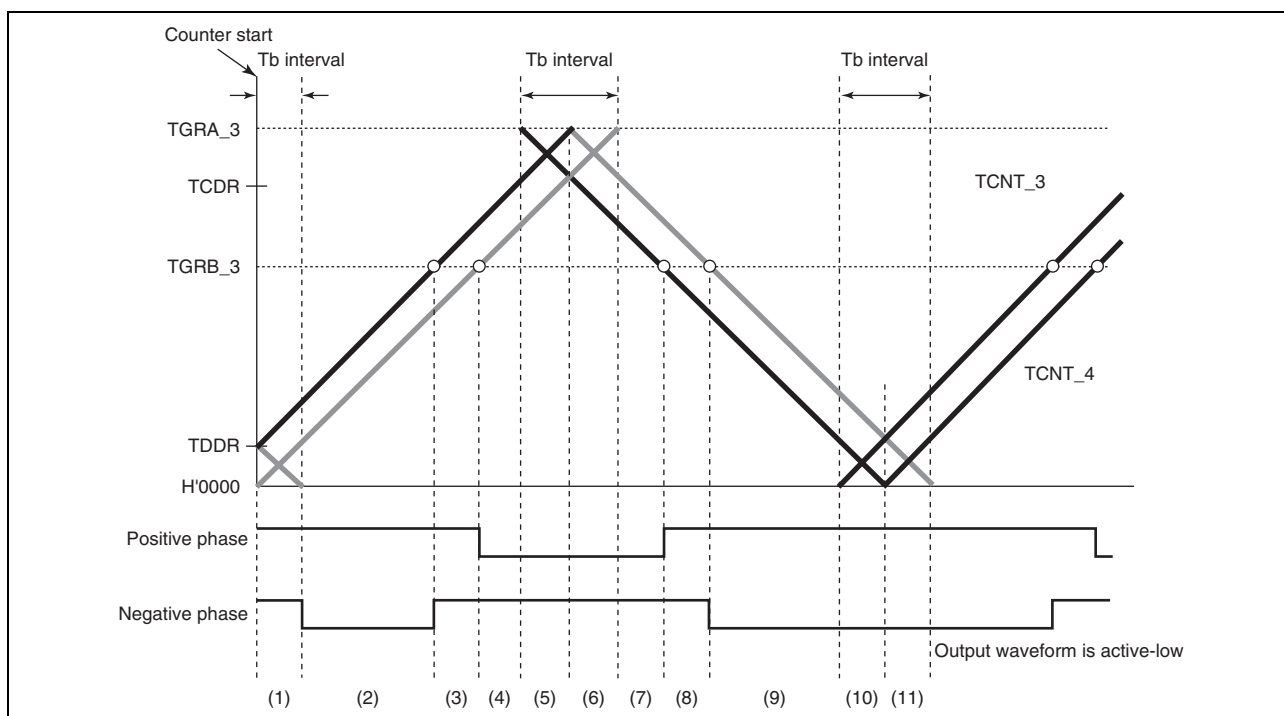
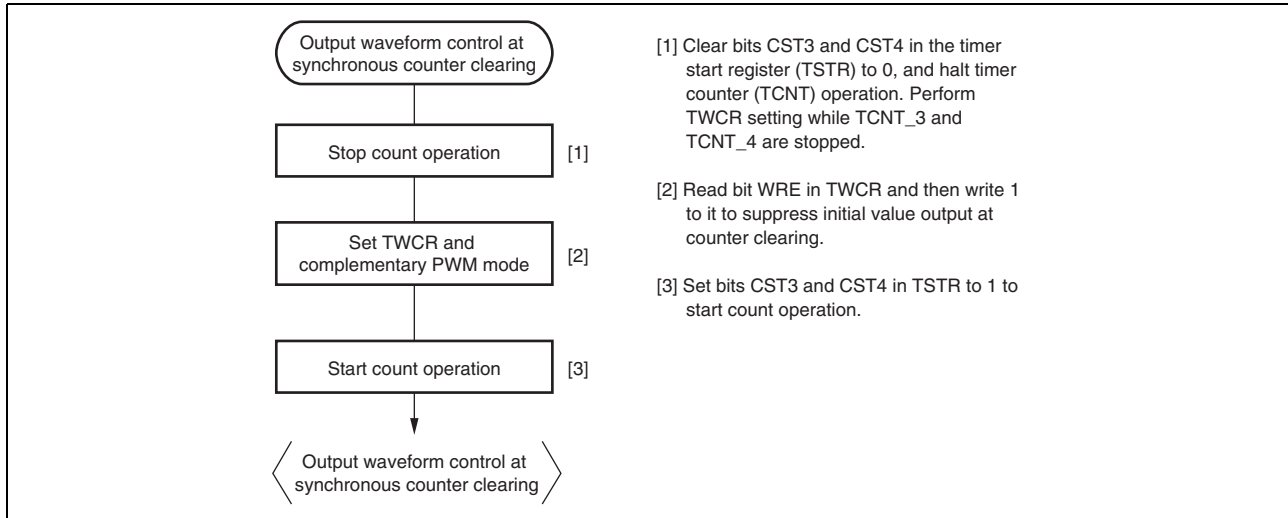


Figure 10.56 Timing for Synchronous Counter Clearing

- Example of Procedure for Setting Output Waveform Control at Synchronous Counter Clearing in Complementary PWM Mode

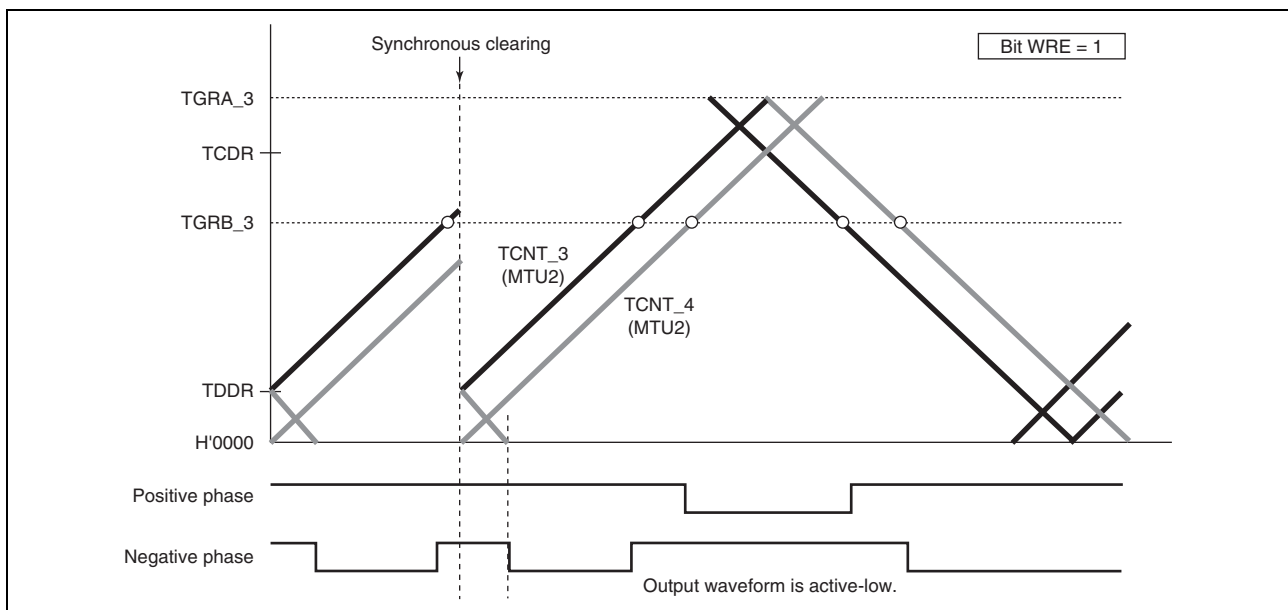
An example of the procedure for setting output waveform control at synchronous counter clearing in complementary PWM mode is shown in Figure 10.57.



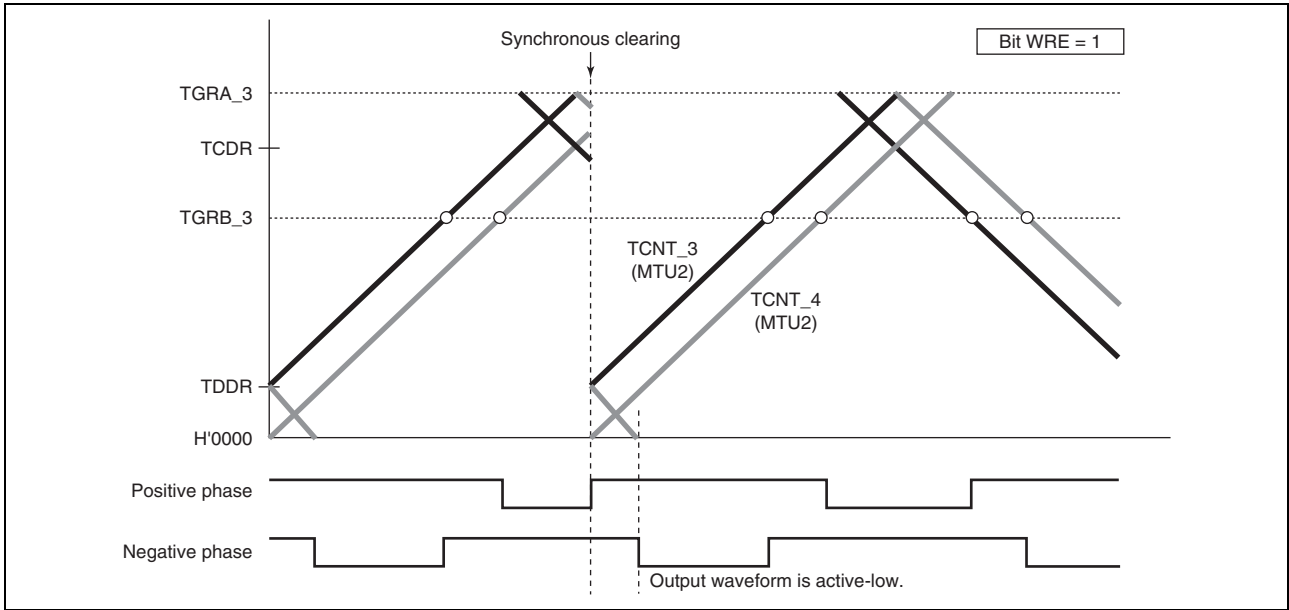
**Figure 10.57 Example of Procedure for Setting Output Waveform Control at Synchronous Counter Clearing in Complementary PWM Mode**

- Examples of Output Waveform Control at Synchronous Counter Clearing in Complementary PWM Mode

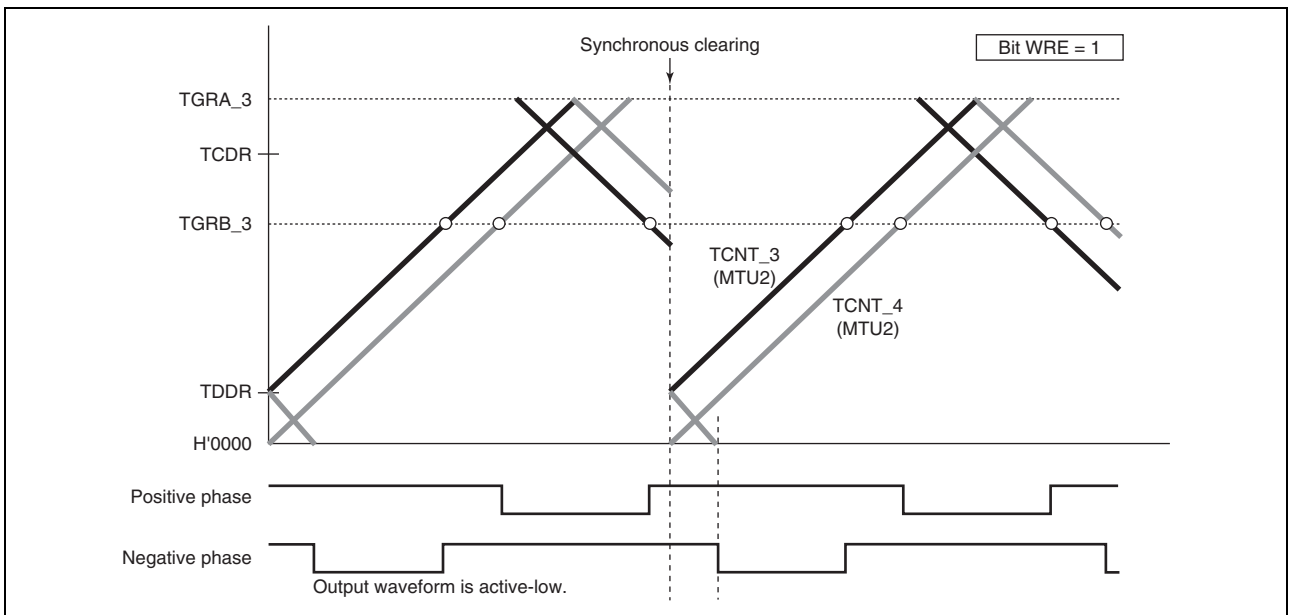
Figure 10.58 to Figure 10.61 show examples of output waveform control in which this module operates in complementary PWM mode and synchronous counter clearing is generated while the WRE bit in TWCR is set to 1. In the examples shown in Figure 10.58 to Figure 10.61, synchronous counter clearing occurs at timing (3), (6), (8), and (11) shown in Figure 10.56, respectively.



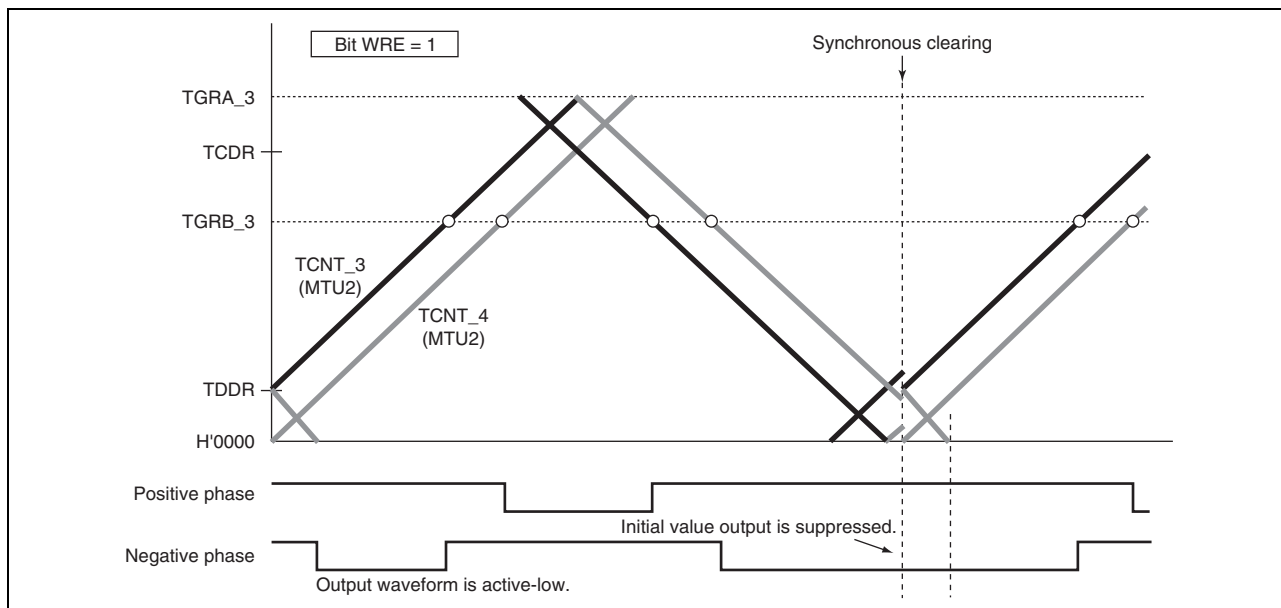
**Figure 10.58 Example of Synchronous Clearing in Dead Time during Up-Counting (Timing (3) in Figure 10.56; Bit WRE of TWCR is 1)**



**Figure 10.59** Example of Synchronous Clearing in Interval Tb at Crest (Timing (6) in Figure 10.56; Bit WRE of TWCR is 1)



**Figure 10.60** Example of Synchronous Clearing in Dead Time during Down-Counting (Timing (8) in Figure 10.56; Bit WRE of TWCR is 1)



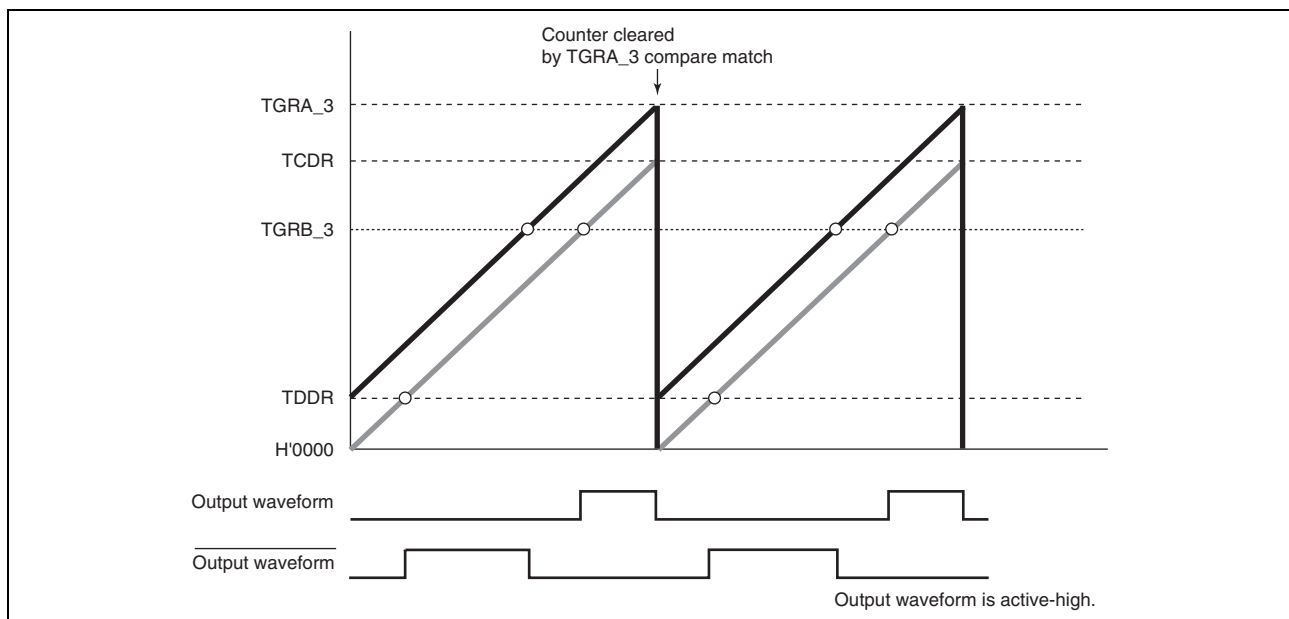
**Figure 10.61 Example of Synchronous Clearing in Interval Tb at Trough (Timing (11) in Figure 10.56; Bit WRE of TWCR is 1)**

### (o) Counter Clearing by TGRA\_3 Compare Match

In complementary PWM mode, by setting the CCE bit in the timer waveform control register (TWCR), it is possible to have TCNT\_3, TCNT\_4, and TCNTS cleared by TGRA\_3 compare match.

Figure 10.62 illustrates an operation example.

- Note 1. Use this function only in complementary PWM mode 1 (transfer at crest)
- Note 2. Do not specify synchronous clearing by another channel (do not set the SYNC0 to SYNC4 bits in the timer synchronous register (TSYR) to 1).
- Note 3. Do not set the PWM duty value to H'0000.
- Note 4. Do not set the PSYE bit in timer output control register 1 (TOCR1) to 1.



**Figure 10.62 Example of Counter Clearing Operation by TGRA\_3 Compare Match**

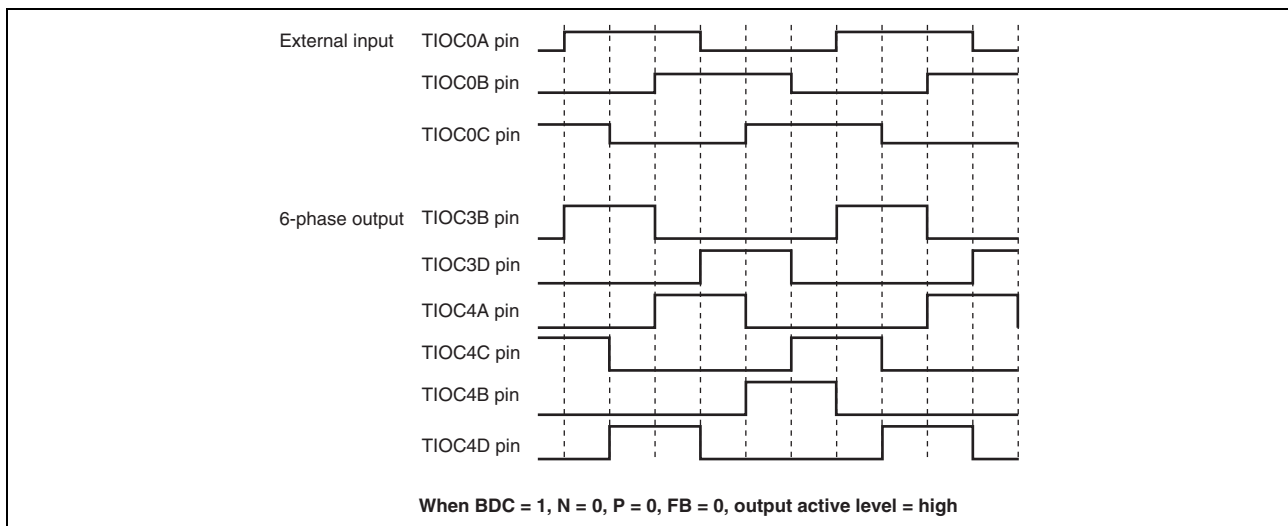
### (p) Example of AC Synchronous Motor (Brushless DC Motor) Drive Waveform Output

In complementary PWM mode, a brushless DC motor can easily be controlled using the timer gate control register (TGCR). Figure 10.63 to Figure 10.66 show examples of brushless DC motor drive waveforms created using TGCR.

When output phase switching for a 3-phase brushless DC motor is performed by means of external signals detected with a Hall element, etc., clear the FB bit in TGCR to 0. In this case, the external signals indicating the polarity position are input to channel 0 timer input pins TIOC0A, TIOC0B, and TIOC0C (set with the general I/O ports). When an edge is detected at pin TIOC0A, TIOC0B, or TIOC0C, the output on/off state is switched automatically.

When the FB bit is 1, the output on/off state is switched when the UF, VF, or WF bit in TGCR is cleared to 0 or set to 1. The drive waveforms are output from the complementary PWM mode 6-phase output pins. With this 6-phase output, in the case of on output, it is possible to use complementary PWM mode output and perform chopping output by setting the N bit or P bit to 1. When the N bit or P bit is 0, level output is selected.

The 6-phase output active level (on output level) can be set with the OLSN and OLSP bits in the timer output control register (TOCR) regardless of the setting of the N and P bits.



**Figure 10.63** Example of Output Phase Switching by External Input (1)

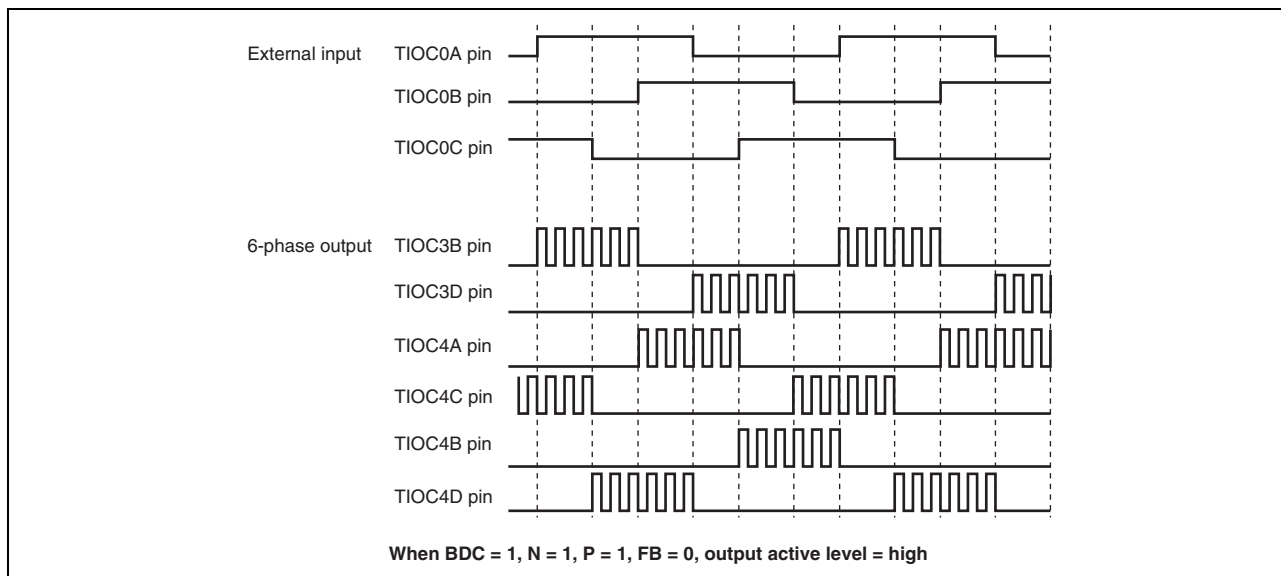


Figure 10.64 Example of Output Phase Switching by External Input (2)

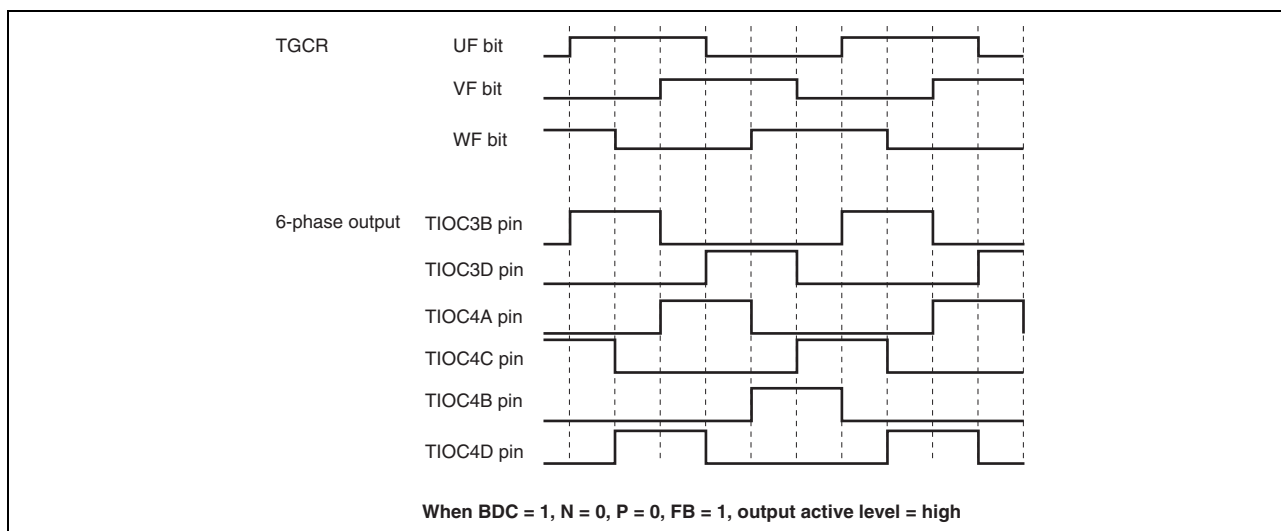
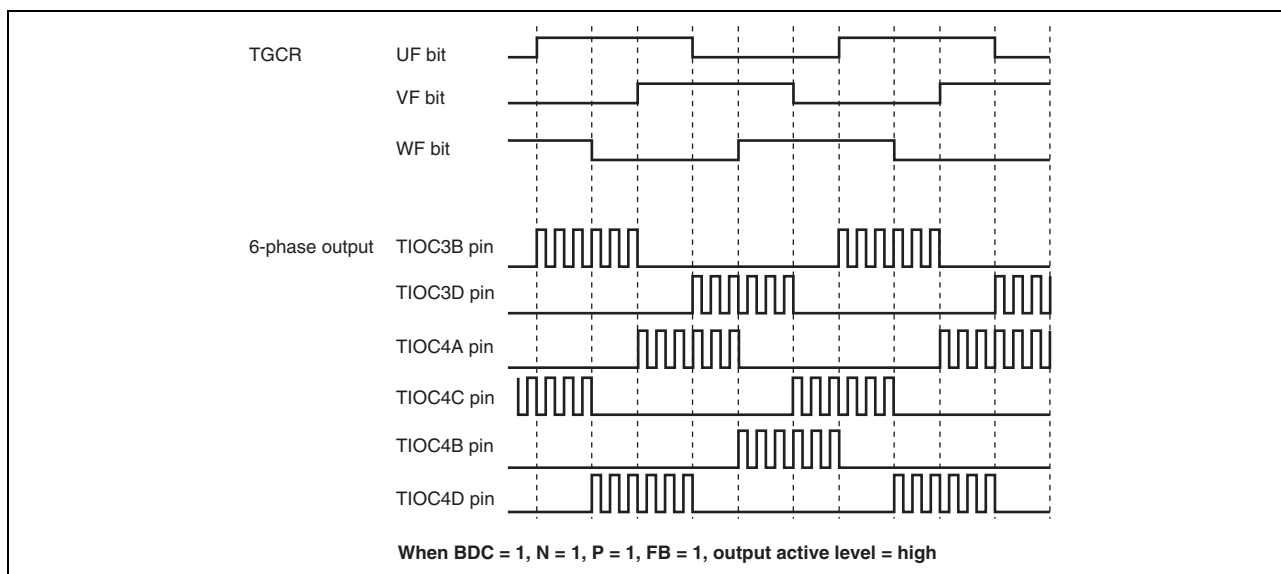


Figure 10.65 Example of Output Phase Switching by Means of UF, VF, WF Bit Settings (1)





**Figure 10.66 Example of Output Phase Switching by Means of UF, VF, WF Bit Settings (2)**

**(q) A/D Converter Start Request Setting**

In complementary PWM mode, an A/D converter start request can be issued using a TGRA\_3 compare-match, TCNT\_4 underflow (trough), or compare-match on a channel other than channels 3 and 4.

When start requests using a TGRA\_3 compare-match are specified, A/D conversion can be started at the crest of the TCNT\_3 count.

A/D converter start requests can be set by setting the TTGE bit to 1 in the timer interrupt enable register (TIER). To issue an A/D converter start request at a TCNT\_4 underflow (trough), set the TTGE2 bit in TIER\_4 to 1.

### (3) Interrupt Skipping in Complementary PWM Mode

Interrupts TGIA\_3 (at the crest) and TCIV\_4 (at the trough) in channels 3 and 4 can be skipped up to seven times by making settings in the timer interrupt skipping set register (TITCR).

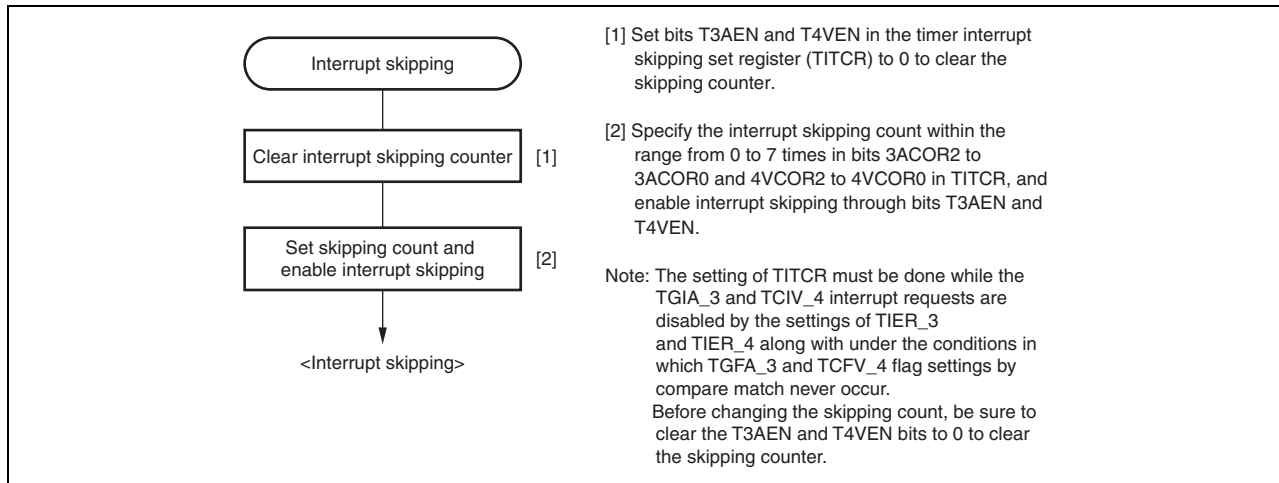
Transfers from a buffer register to a temporary register or a compare register can be skipped in coordination with interrupt skipping by making settings in the timer buffer transfer register (TBTER). For the linkage with buffer registers, refer to description (c), Buffer Transfer Control Linked with Interrupt Skipping, below.

A/D converter start requests generated by the A/D converter start request delaying function can also be skipped in coordination with interrupt skipping by making settings in the timer A/D converter request control register (TADCR). For the linkage with the A/D converter start request delaying function, refer to section 10.4.9, A/D Converter Start Request Delaying Function.

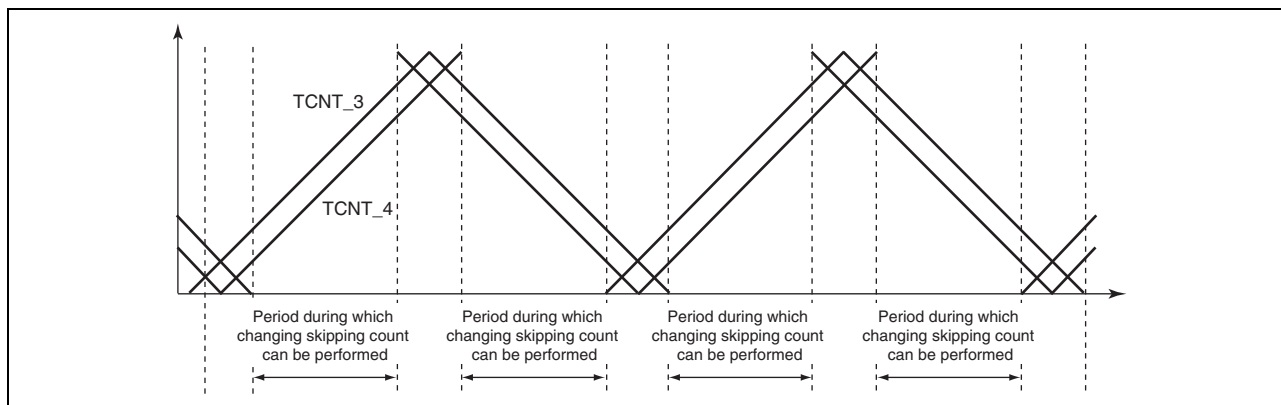
The setting of the timer interrupt skipping setting register (TITCR) must be done while the TGIA\_3 and TCIV\_4 interrupt requests are disabled by the settings of TIER\_3 and TIER\_4 along with under the conditions in which TGFA\_3 and TCFV\_4 flag settings by compare match never occur. Before changing the skipping count, be sure to clear the T3AEN and T4VEN bits to 0 to clear the skipping counter.

#### (a) Example of Interrupt Skipping Operation Setting Procedure

Figure 10.67 shows an example of the interrupt skipping operation setting procedure. Figure 10.68 shows the periods during which interrupt skipping count can be changed.



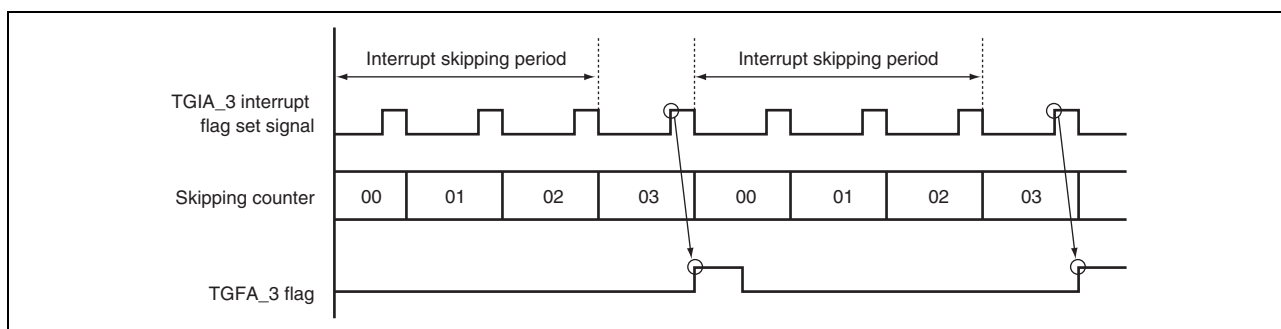
**Figure 10.67 Example of Interrupt Skipping Operation Setting Procedure**



**Figure 10.68** Periods during which Interrupt Skipping Count can be Changed

(b) Example of Interrupt Skipping Operation

Figure 10.69 shows an example of TGIA\_3 interrupt skipping in which the interrupt skipping count is set to three by the 3ACOR bit and the T3AEN bit is set to 1 in the timer interrupt skipping set register (TITCR).



**Figure 10.69** Example of Interrupt Skipping Operation

(c) Buffer Transfer Control Linked with Interrupt Skipping

In complementary PWM mode, whether to transfer data from a buffer register to a temporary register and whether to link the transfer with interrupt skipping can be specified with the BTE1 and BTE0 bits in the timer buffer transfer set register (TBTER).

Figure 10.70 shows an example of operation when buffer transfer is suppressed (BTE1 = 0 and BTE0 = 1). While this setting is valid, data is not transferred from the buffer register to the temporary register.

Figure 10.71 shows an example of operation when buffer transfer is linked with interrupt skipping (BTE1 = 1 and BTE0 = 0). While this setting is valid, data is not transferred from the buffer register to the temporary register outside the buffer transfer-enabled period. Depending on the rewrite timing from the interrupt generation to the buffer register, there are two types of the transfer timing such as from the buffer register to the temporary register and from the temporary register to the general register.

Note that the buffer transfer-enabled period depends on the T3AEN and T4VEN bit settings in the timer interrupt skipping set register (TITCR). Figure 10.72 shows the relationship between the T3AEN and T4VEN bit settings in TITCR and buffer transfer-enabled period.

Note: This function must always be used in combination with interrupt skipping. When interrupt skipping is disabled (the T3AEN and T4VEN bits in the timer interrupt skipping set register (TITCR) are cleared to 0 or the skipping count set bits (3ACOR and 4VCOR) in TITCR are cleared to 0), make sure that buffer transfer is not linked with interrupt skipping (clear the BTE1 bit in the timer buffer transfer set register (TBTER) to 0). If buffer transfer is linked with interrupt skipping while interrupt skipping is disabled, buffer transfer is never performed.

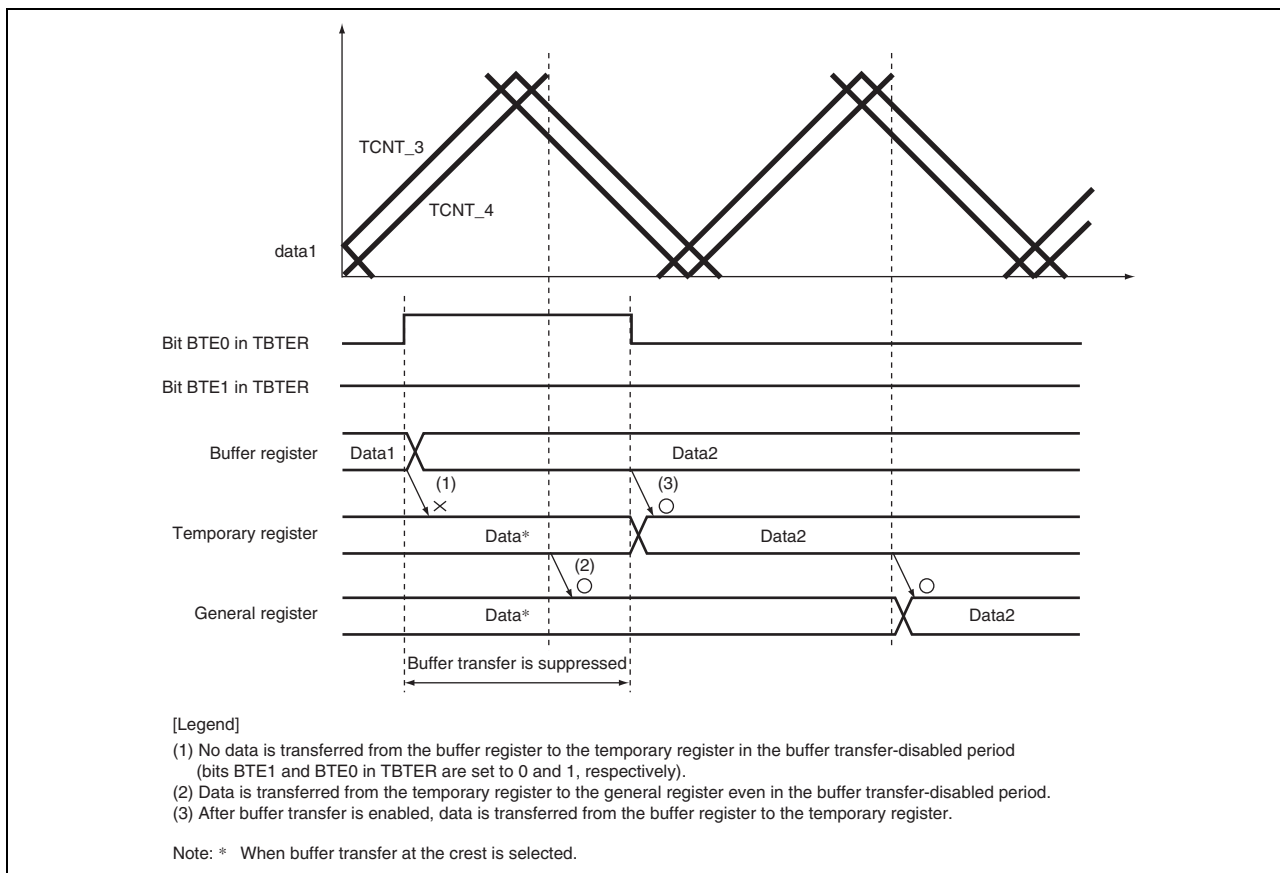
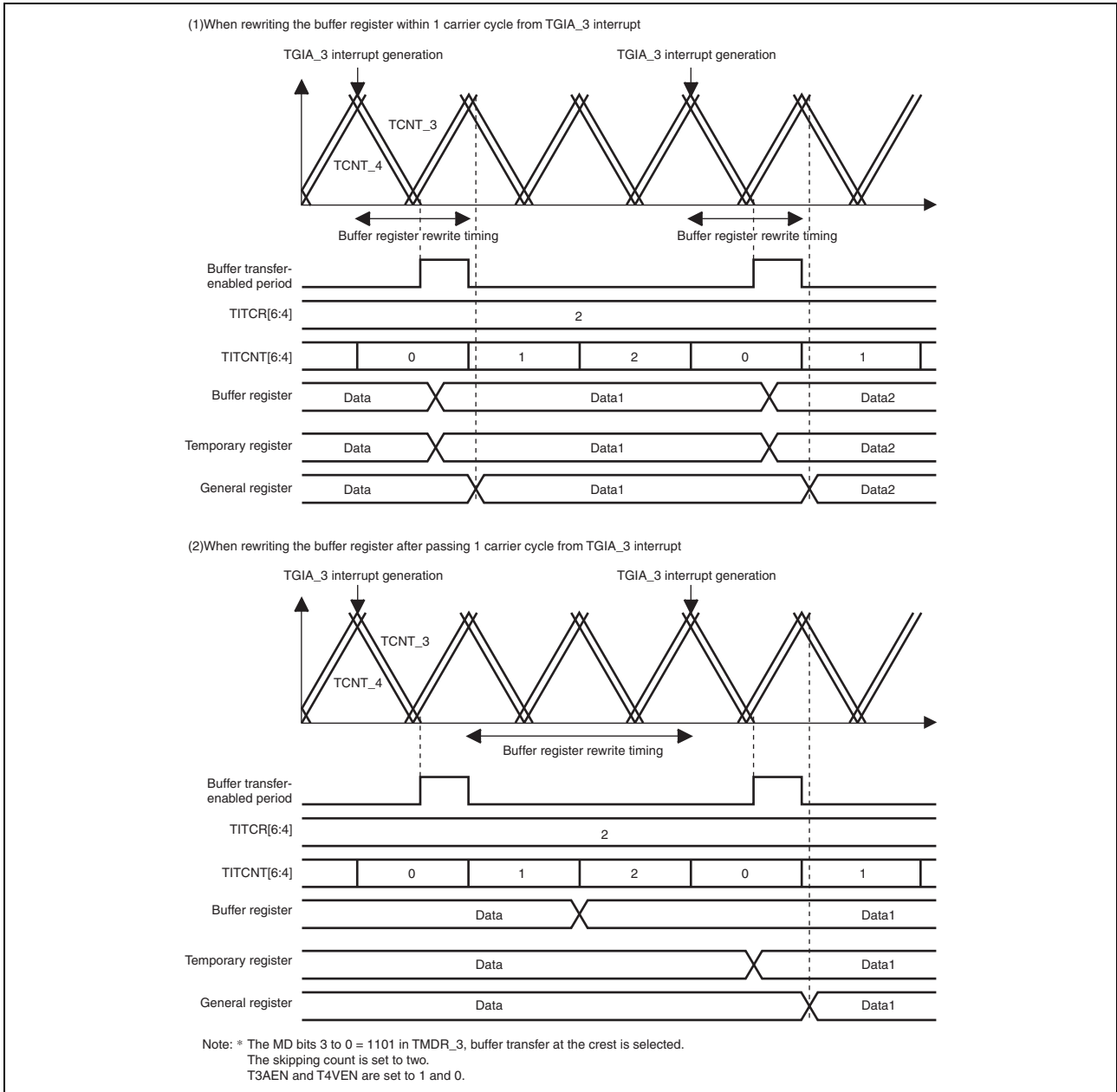
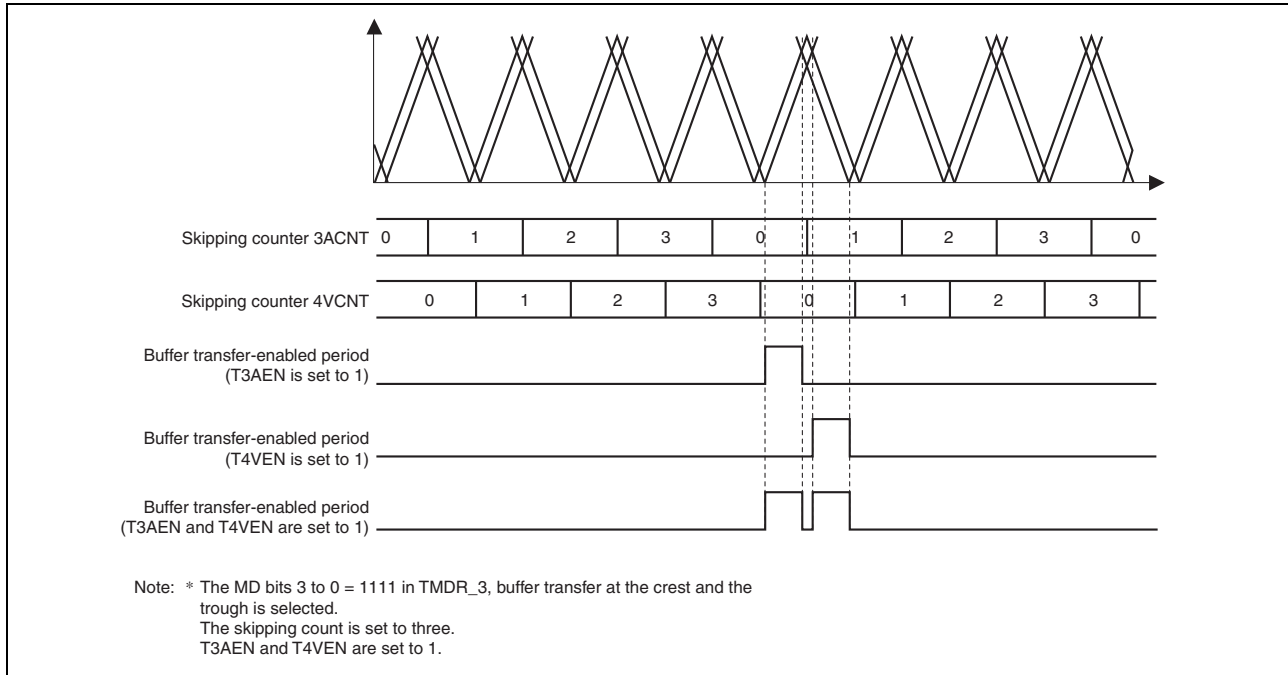


Figure 10.70 Example of Operation when Buffer Transfer is Suppressed (BTE1 = 0 and BTE0 = 1)



**Figure 10.71 Example of Operation when Buffer Transfer is Linked with Interrupt Skipping (BTE1 = 1 and BTE0 = 0)**



**Figure 10.72 Relationship between Bits T3AEN and T4VEN in TITCR and Buffer Transfer-Enabled Period**

#### (4) Complementary PWM Mode Output Protection Function

Complementary PWM mode output has the following protection function.

##### (a) Register and counter miswrite prevention function

With the exception of the buffer registers, which can be rewritten at any time, access by the CPU can be enabled or disabled for the mode registers, control registers, compare registers, and counters used in complementary PWM mode by means of the RWE bit in the timer read/write enable register (TRWER). The applicable registers are some (21 in total) of the registers in channels 3 and 4 shown in the following:

- TCR\_3 and TCR\_4, TMDR\_3 and TMDR\_4, TIORH\_3 and TIORH\_4, TIORL\_3 and TIORL\_4, TIER\_3 and TIER\_4, TCNT\_3 and TCNT\_4, TGRA\_3 and TGRA\_4, TGRB\_3 and TGRB\_4, TOER, TOCR, TGCR, TCDR, and TDDR.

This function enables miswriting due to CPU runaway to be prevented by disabling CPU access to the mode registers, control registers, and counters. When the applicable registers are read in the access-disabled state, undefined values are returned. Writing to these registers is ignored.

### 10.4.9 A/D Converter Start Request Delaying Function

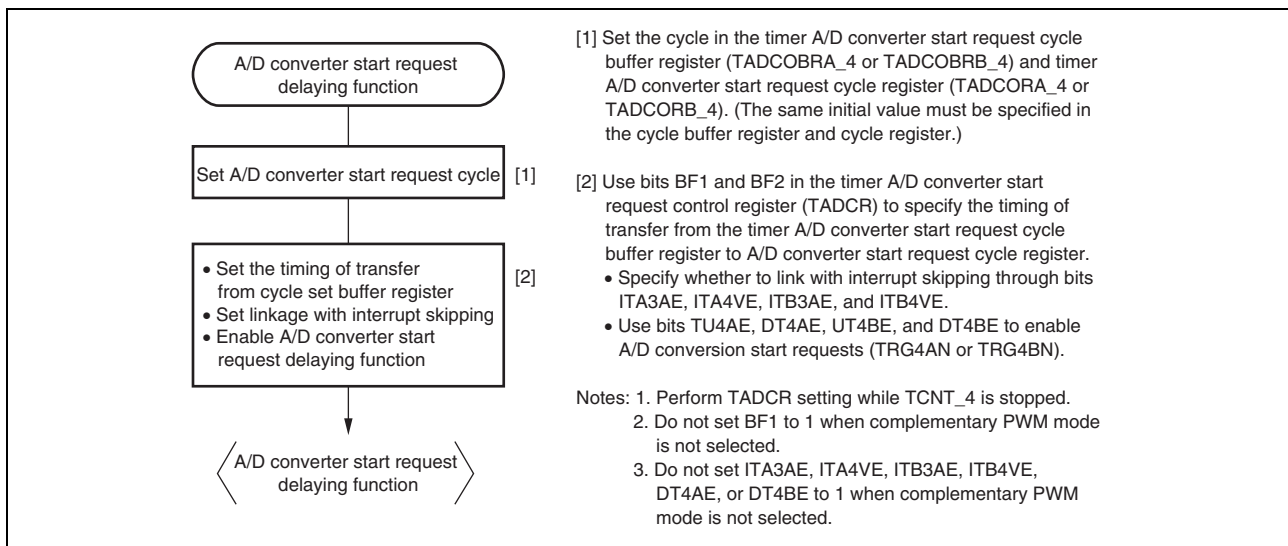
A/D converter start requests can be issued in channel 4 by making settings in the timer A/D converter start request control register (TADCR), timer A/D converter start request cycle set registers (TADCORA\_4 and TADCORB\_4), and timer A/D converter start request cycle set buffer registers (TADCOBRA\_4 and TADCOBRB\_4).

The A/D converter start request delaying function compares TCNT\_4 with TADCORA\_4 or TADCORB\_4, and when their values match, the function issues a respective A/D converter start request (TRG4AN or TRG4BN).

A/D converter start requests (TRG4AN and TRG4BN) can be skipped in coordination with interrupt skipping by setting the ITA3AE, ITA4VE, ITB3AE, and ITB4VE bits in TADCR.

- Example of Procedure for Specifying A/D Converter Start Request Delaying Function

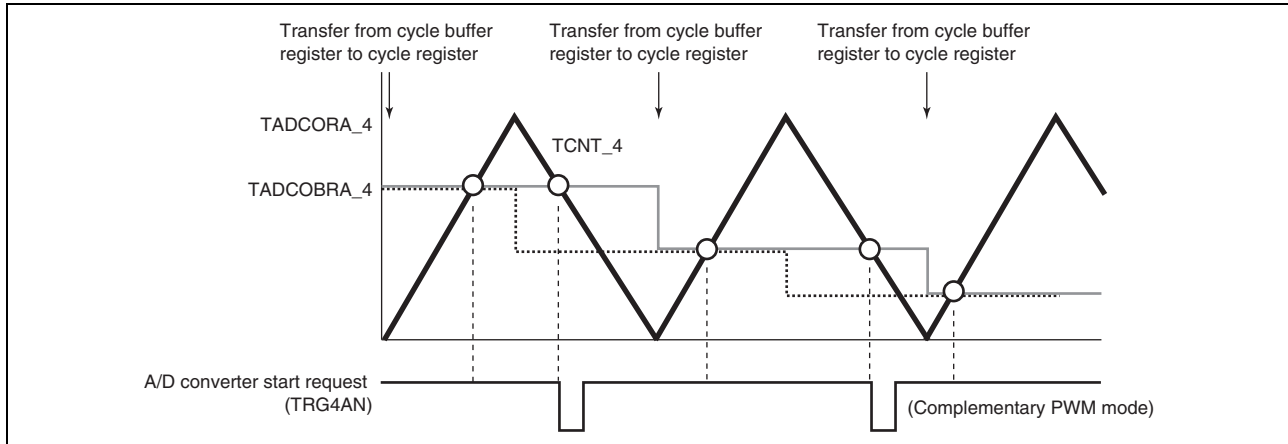
Figure 10.73 shows an example of procedure for specifying the A/D converter start request delaying function.



**Figure 10.73 Example of Procedure for Specifying A/D Converter Start Request Delaying Function**

- Basic Operation Example of A/D Converter Start Request Delaying Function

Figure 10.74 shows a basic example of A/D converter request signal (TRG4AN) operation when the trough of TCNT\_4 is specified for the buffer transfer timing and an A/D converter start request signal is output during TCNT\_4 down-counting.



**Figure 10.74 Basic Example of A/D Converter Start Request Signal (TRG4AN) Operation**

- Buffer Transfer

The data in the timer A/D converter start request cycle set registers (TADCORA\_4 and TADCORB\_4) is updated by writing data to the timer A/D converter start request cycle set buffer registers (TADCOBRA\_4 and TADCOBRB\_4). Data is transferred from the buffer registers to the respective cycle set registers at the timing selected with the BF1 and BF0 bits in the timer A/D converter start request control register (TADCR\_4).

- A/D Converter Start Request Delaying Function Linked with Interrupt Skipping

A/D converter start requests (TRG4AN and TRG4BN) can be issued in coordination with interrupt skipping by making settings in the ITA3AE, ITA4VE, ITB3AE, and ITB4VE bits in the timer A/D converter start request control register (TADCR).

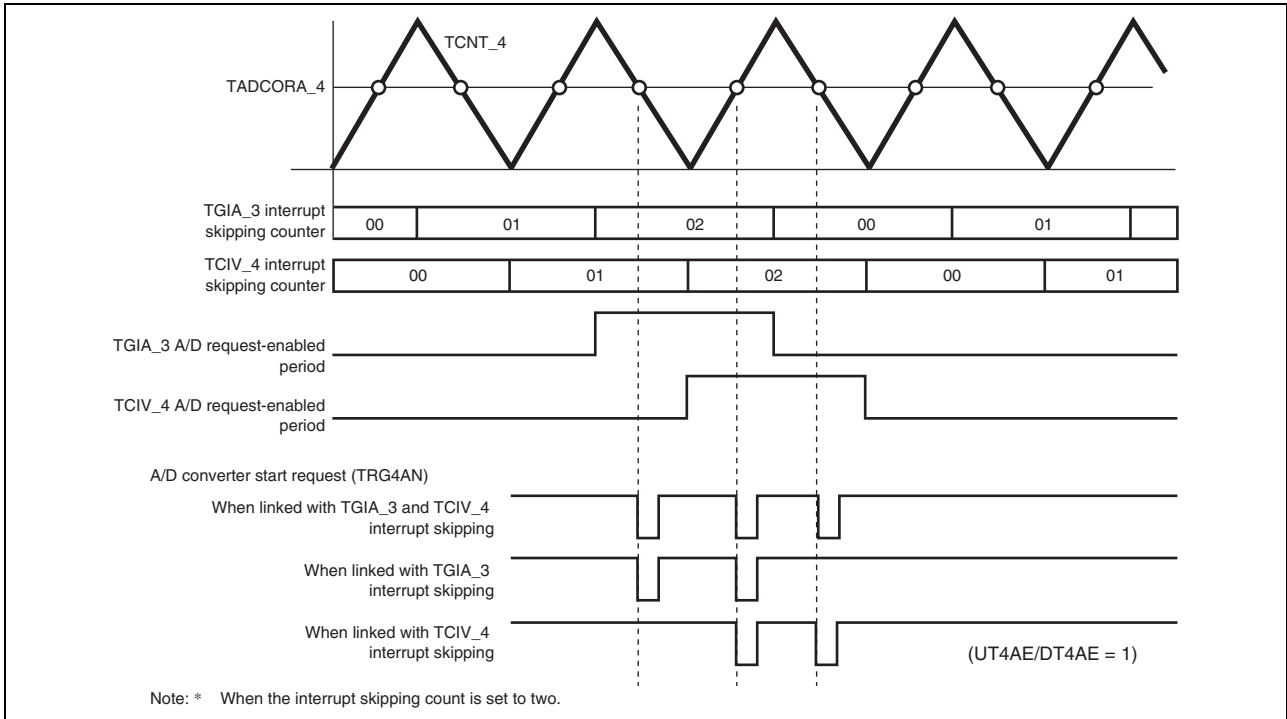
Figure 10.75 shows an example of A/D converter start request signal (TRG4AN) operation when TRG4AN output is enabled during TCNT\_4 up counting and down counting and A/D converter start requests are linked with interrupt skipping.

Figure 10.76 shows another example of A/D converter start request signal (TRG4AN) operation when TRG4AN output is enabled during TCNT\_4 up counting and A/D converter start requests are linked with interrupt skipping.

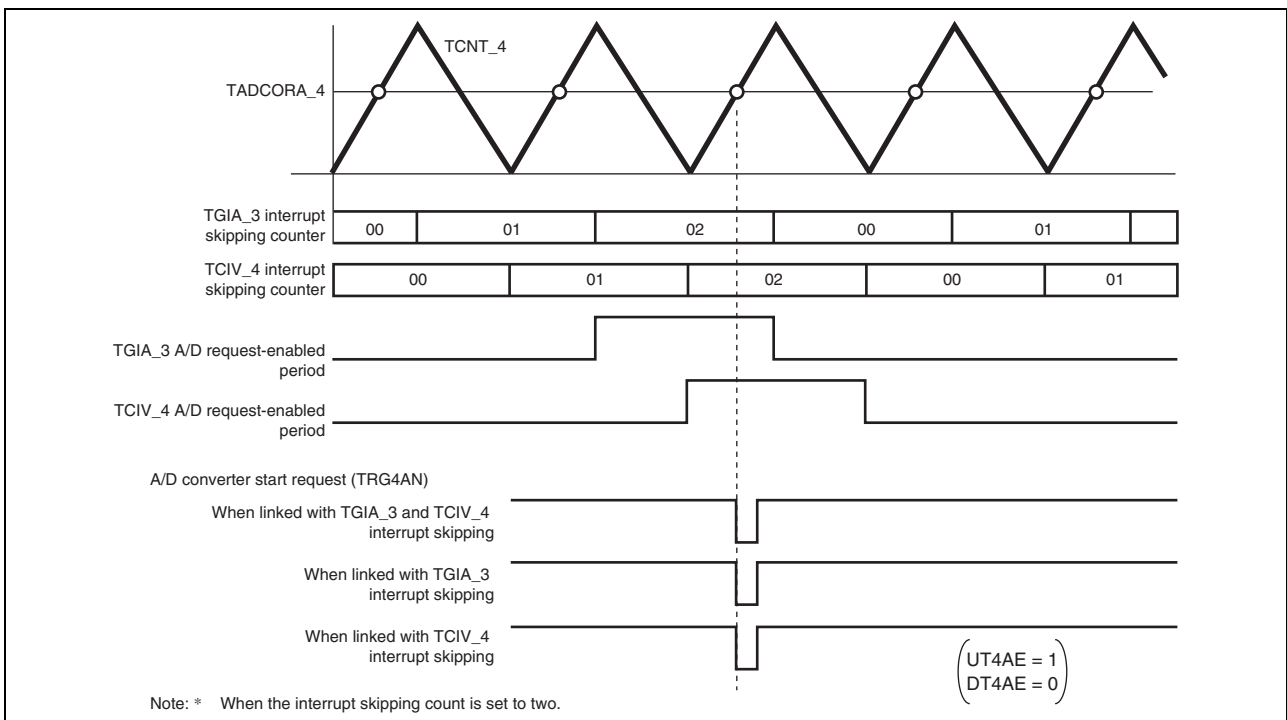
**Note:** This function must be used in combination with interrupt skipping.

When interrupt skipping is disabled (the T3AEN and T4VEN bits in the timer interrupt skipping set register (TITCR) are cleared to 0 or the skipping count set bits (3ACOR and 4VCOR) in TITCR are cleared to 0), make sure that A/D converter start requests are not linked with interrupt skipping (clear the ITA3AE, ITA4VE, ITB3AE, and ITB4VE bits in the timer A/D converter start request control register (TADCR) to 0).





**Figure 10.75 Example of A/D Converter Start Request Signal (TRG4AN) Operation Linked with Interrupt Skipping(1)**

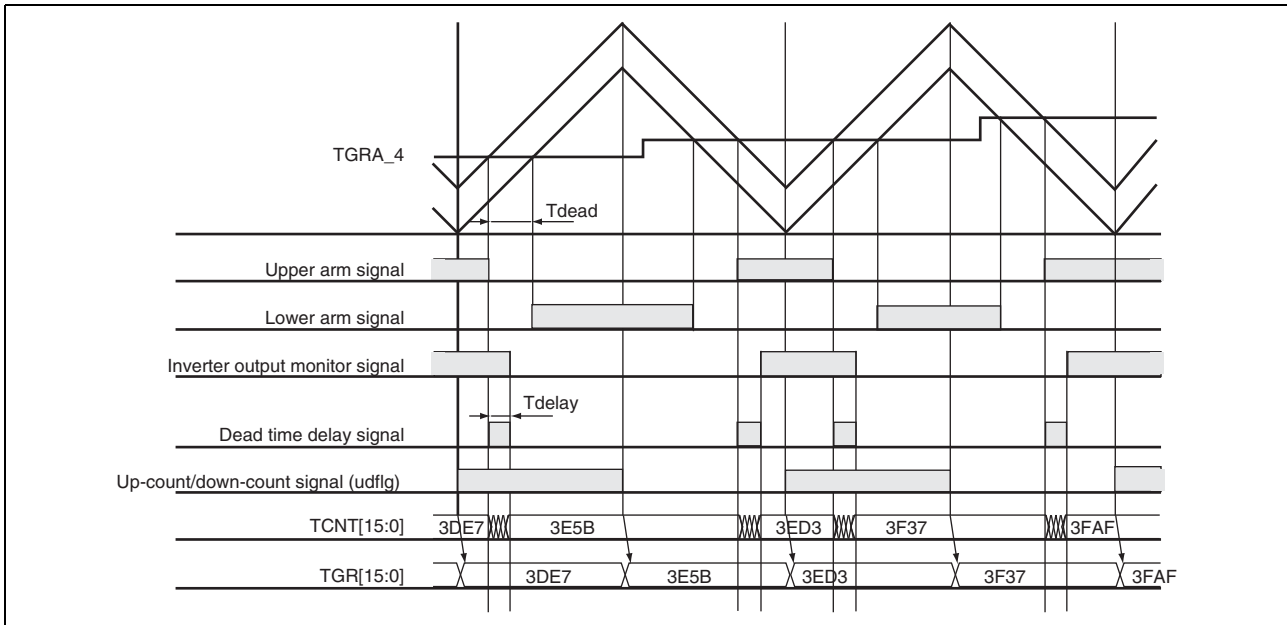


**Figure 10.76 Example of A/D Converter Start Request Signal (TRG4AN) Operation Linked with Interrupt Skipping(2)**

### 10.4.10 TCNT Capture at Crest and/or Trough in Complementary PWM Operation

The TCNT value is captured in TGR at either the crest or trough or at both the crest and trough during complementary PWM operation. The timing for capturing in TGR can be selected by TIOR.

Figure 10.77 shows an example in which TCNT is used as a free-running counter without being cleared, and the TCNT value is captured in TGR at the specified timing (either crest or trough, or both crest and trough).



**Figure 10.77 TCNT Capturing at Crest and/or Trough in Complementary PWM Operation**

## 10.5 Interrupt Sources

### 10.5.1 Interrupt Sources and Priorities

This module has three kinds of interrupt sources; TGR input capture/compare match, TCNT overflow, and TCNT underflow. Each interrupt source has its own status flag and enable/disable bit, allowing the generation of interrupt request signals to be enabled or disabled individually.

When an interrupt request is generated, the corresponding status flag in TSR is set to 1. If the corresponding enable/disable bit in TIER is set to 1 at this time, an interrupt is requested. The interrupt request is cleared by clearing the status flag to 0.

Relative channel priorities can be changed by the interrupt controller, however the priority order within a channel is fixed. For details, see section 7, Interrupt Controller.

Table 10.55 lists the interrupt sources of this module.

**Table 10.55 Interrupts of Multi-Function Timer Pulse Unit 2**

Channel	Name	Interrupt Source	Interrupt Flag	Activation of Direct Memory Access Controller	Priority	
0	TGIA_0	TGRA_0 input capture/compare match	TGFA_0	Possible	High ↑	
	TGIB_0	TGRB_0 input capture/compare match	TGFB_0	Not possible		
	TGIC_0	TGRC_0 input capture/compare match	TGFC_0	Not possible		
	TGID_0	TGRD_0 input capture/compare match	TGFD_0	Not possible		
	TCIV_0	TCNT_0 overflow	TCFV_0	Not possible		
	TGIE_0	TGRE_0 compare match	TGFE_0	Not possible		
	TGIF_0	TGRF_0 compare match	TGFF_0	Not possible		
1	TGIA_1	TGRA_1 input capture/compare match	TGFA_1	Possible	↓	
	TGIB_1	TGRB_1 input capture/compare match	TGFB_1	Not possible		
	TCIV_1	TCNT_1 overflow	TCFV_1	Not possible		
	TCIU_1	TCNT_1 underflow	TCFU_1	Not possible		
2	TGIA_2	TGRA_2 input capture/compare match	TGFA_2	Possible		
	TGIB_2	TGRB_2 input capture/compare match	TGFB_2	Not possible		
	TCIV_2	TCNT_2 overflow	TCFV_2	Not possible		
	TCIU_2	TCNT_2 underflow	TCFU_2	Not possible		
3	TGIA_3	TGRA_3 input capture/compare match	TGFA_3	Possible		
	TGIB_3	TGRB_3 input capture/compare match	TGFB_3	Not possible		
	TGIC_3	TGRC_3 input capture/compare match	TGFC_3	Not possible		
	TGID_3	TGRD_3 input capture/compare match	TGFD_3	Not possible		
	TCIV_3	TCNT_3 overflow	TCFV_3	Not possible		
4	TGIA_4	TGRA_4 input capture/compare match	TGFA_4	Possible		Low
	TGIB_4	TGRB_4 input capture/compare match	TGFB_4	Not possible		
	TGIC_4	TGRC_4 input capture/compare match	TGFC_4	Not possible		
	TGID_4	TGRD_4 input capture/compare match	TGFD_4	Not possible		
	TCIV_4	TCNT_4 overflow/underflow	TCFV_4	Not possible		

Note: This table shows the initial state immediately after a reset. The relative channel priorities can be changed by the interrupt controller.

### (1) Input Capture/Compare Match Interrupt

An interrupt is requested if the TGIE bit in TIER is set to 1 when the TGF flag in TSR is set to 1 by the occurrence of a TGR input capture/compare match on a particular channel. The interrupt request is cleared by clearing the TGF flag to 0. This module has eighteen input capture/compare match interrupts, six for channel 0, four each for channels 3 and 4, and two each for channels 1 and 2. The TGFE\_0 and TGFF\_0 flags in channel 0 are not set by the occurrence of an input capture.

### (2) Overflow Interrupt

An interrupt is requested if the TCIEV bit in TIER is set to 1 when the TCFV flag in TSR is set to 1 by the occurrence of TCNT overflow on a channel. The interrupt request is cleared by clearing the TCFV flag to 0. This module has five overflow interrupts, one for each channel.

### (3) Underflow Interrupt

An interrupt is requested if the TCIEU bit in TIER is set to 1 when the TCFU flag in TSR is set to 1 by the occurrence of TCNT underflow on a channel. The interrupt request is cleared by clearing the TCFU flag to 0. This module has two underflow interrupts, one each for channels 1 and 2.

## 10.5.2 Activation of Direct Memory Access Controller

The direct memory access controller can be activated by the TGRA input capture/compare match interrupt in each channel. For details, see section 9, Direct Memory Access Controller.

In this module, a total of five TGRA input capture/compare match interrupts can be used as direct memory access controller activation sources, one each for channels 0 to 4.

## 10.5.3 A/D Converter Activation

The A/D converter can be activated by one of the following three methods in this module. Table 10.56 shows the relationship between interrupt sources and A/D converter start request signals.

### (1) A/D Converter Activation by TGRA Input Capture/Compare Match or at TCNT\_4 Trough in Complementary PWM Mode

The A/D converter can be activated by the occurrence of a TGRA input capture/compare match in each channel. In addition, if complementary PWM operation is performed while the TTGE2 bit in TIER\_4 is set to 1, the A/D converter can be activated at the trough of TCNT\_4 count (TCNT\_4 = H'0000).

A/D converter start request signal TRGAN is issued to the A/D converter under either one of the following conditions.

- When the TGFA flag in TSR is set to 1 by the occurrence of a TGRA input capture/compare match on a particular channel while the TTGE bit in TIER is set to 1
- When the TCNT\_4 count reaches the trough (TCNT\_4 = H'0000) during complementary PWM operation while the TTGE2 bit in TIER\_4 is set to 1

When either condition is satisfied, if A/D converter start signal TRGAN from this module is selected as the trigger in the A/D converter, A/D conversion will start.

### (2) A/D Converter Activation by Compare Match between TCNT\_0 and TGRE\_0

The A/D converter can be activated by generating A/D converter start request signal TRG0N when a compare match occurs between TCNT\_0 and TGRE\_0 in channel 0.

When the TGFE flag in TSR2\_0 is set to 1 by the occurrence of a compare match between TCNT\_0 and TGRE\_0 in channel 0 while the TTGE2 bit in TIER2\_0 is set to 1, A/D converter start request TGR0N is issued to the A/D converter. If A/D converter start signal TGR0N from this module is selected as the trigger in the A/D converter, A/D conversion will start.

### (3) A/D Converter Activation by A/D Converter Start Request Delaying Function

The A/D converter can be activated by generating A/D converter start request signal TRG4AN or TRG4BN when the TCNT\_4 count matches the TADCORA or TADCORB value if the UT4AE, DT4AE, UT4BE, or DT4BE bit in the A/D converter start request control register (TADCR) is set to 1. For details, refer to section 10.4.9, A/D Converter Start Request Delaying Function.

A/D conversion will start if A/D converter start signal TRG4AN from this module is selected as the trigger in the A/D converter when TRG4AN is generated or if TRG4BN from this module is selected as the trigger in the A/D converter when TRG4BN is generated.

**Table 10.56 Interrupt Sources and A/D Converter Start Request Signals**

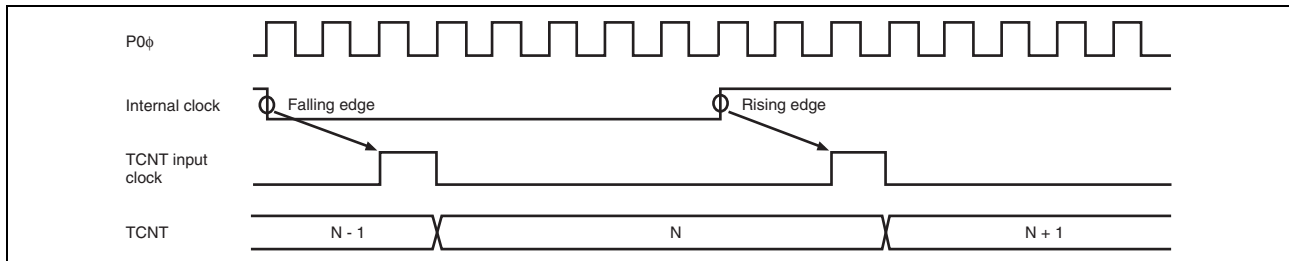
Target Registers	Interrupt Source	A/D Converter Start Request Signal
TGRA_0 and TCNT_0	Input capture/compare match	TRGAN
TGRA_1 and TCNT_1		
TGRA_2 and TCNT_2		
TGRA_3 and TCNT_3		
TGRA_4 and TCNT_4		
TCNT_4	TCNT_4 Trough in complementary PWM mode	
TGRE_0 and TCNT_0	Compare match	TRG0N
TADCORA and TCNT_4		TRG4AN
TADCORB and TCNT_4		TRG4BN

## 10.6 Operation Timing

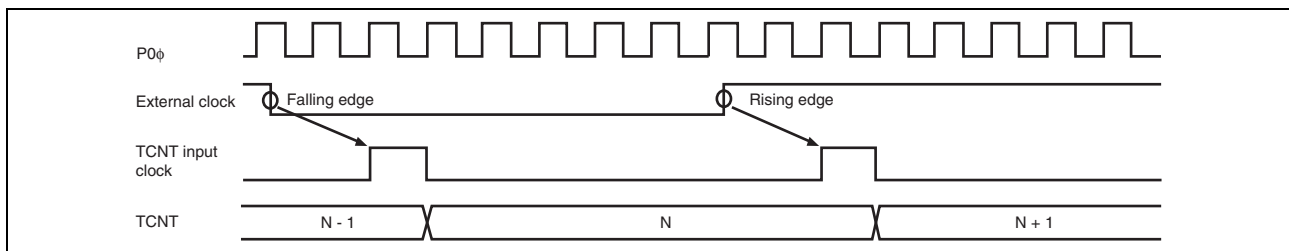
### 10.6.1 Input/Output Timing

#### (1) TCNT Count Timing

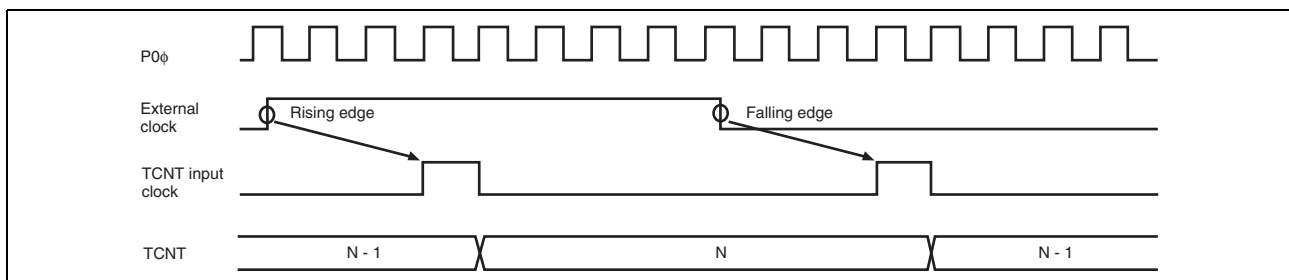
Figure 10.78 shows TCNT count timing in internal clock operation, and Figure 10.79 shows TCNT count timing in external clock operation (normal mode), and Figure 10.80 shows TCNT count timing in external clock operation (phase counting mode).



**Figure 10.78** Count Timing in Internal Clock Operation



**Figure 10.79** Count Timing in External Clock Operation

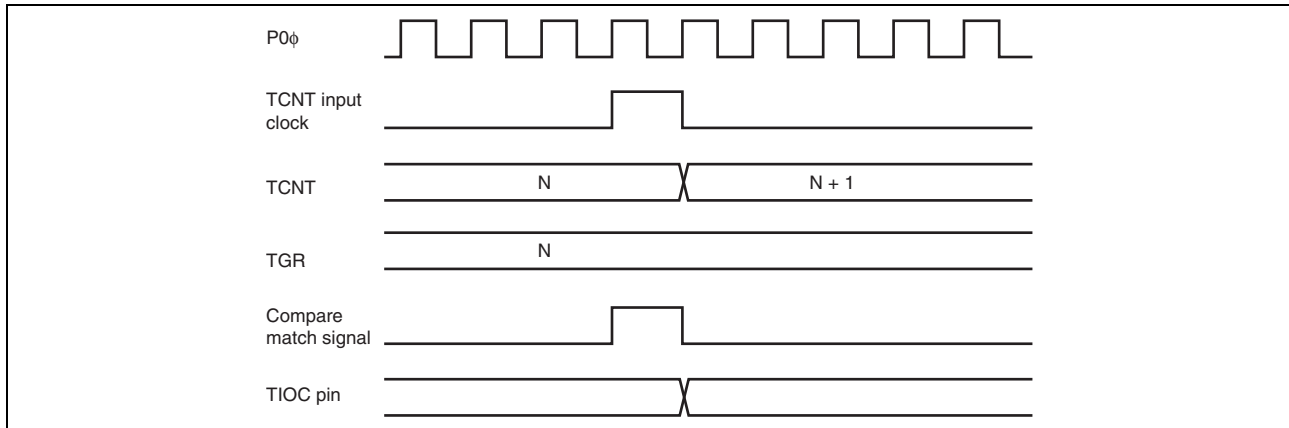


**Figure 10.80** Count Timing in External Clock Operation (Phase Counting Mode)

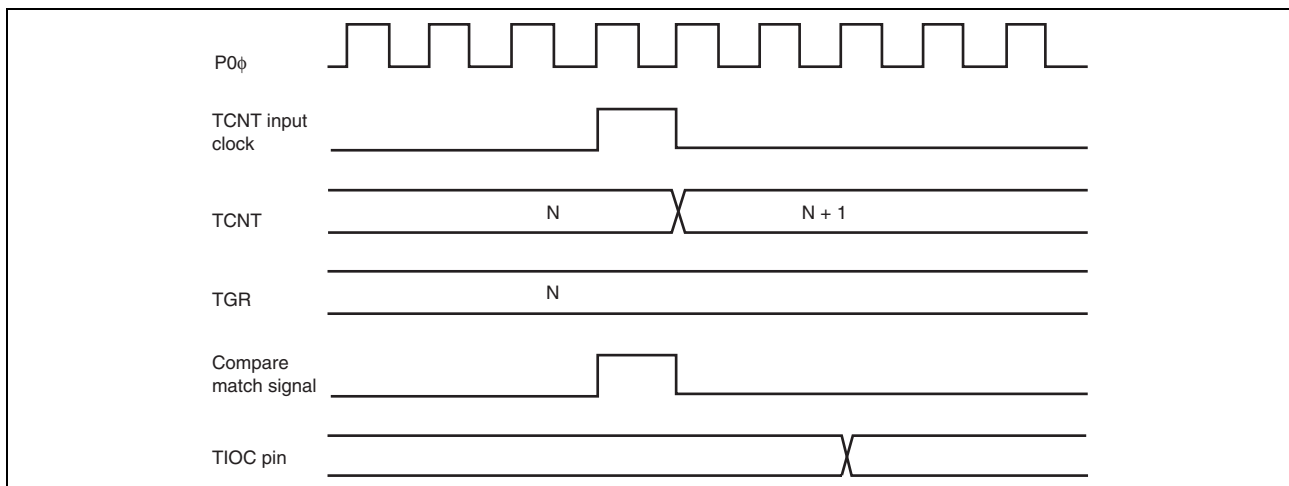
## (2) Output Compare Output Timing

A compare match signal is generated in the final state in which TCNT and TGR match (the point at which the count value matched by TCNT is updated). When a compare match signal is generated, the output value set in TIOR is output at the output compare output pin (TIOC pin). After a match between TCNT and TGR, the compare match signal is not generated until the TCNT input clock is generated.

Figure 10.81 shows output compare output timing (normal mode and PWM mode) and Figure 10.82 shows output compare output timing (complementary PWM mode and reset synchronous PWM mode).



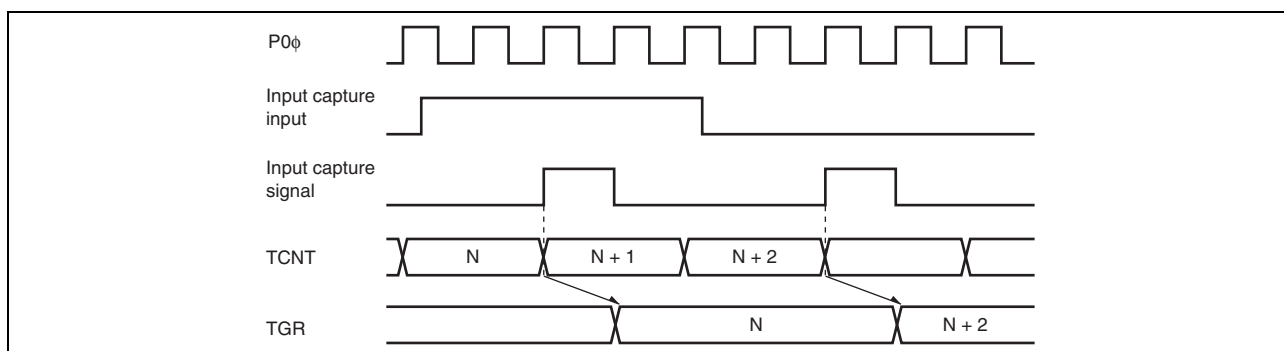
**Figure 10.81 Output Compare Output Timing (Normal Mode/PWM Mode)**



**Figure 10.82 Output Compare Output Timing (Complementary PWM Mode/Reset Synchronous PWM Mode)**

### (3) Input Capture Signal Timing

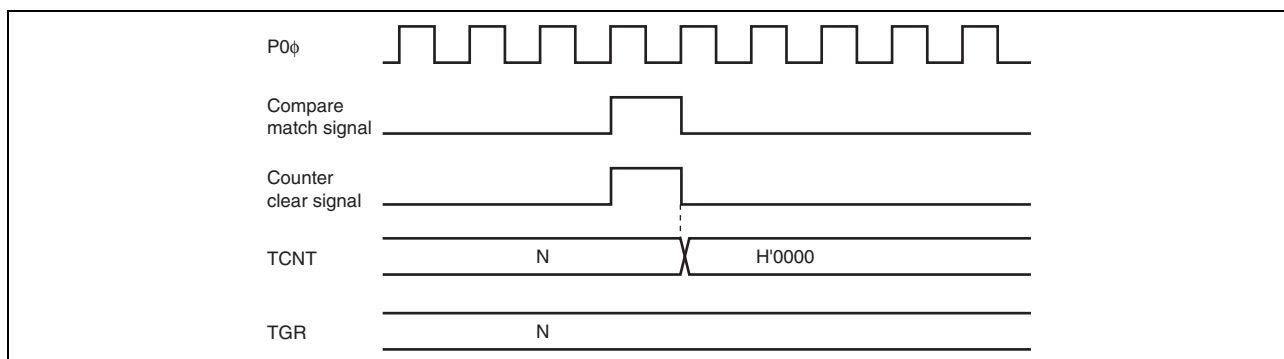
Figure 10.83 shows input capture signal timing.



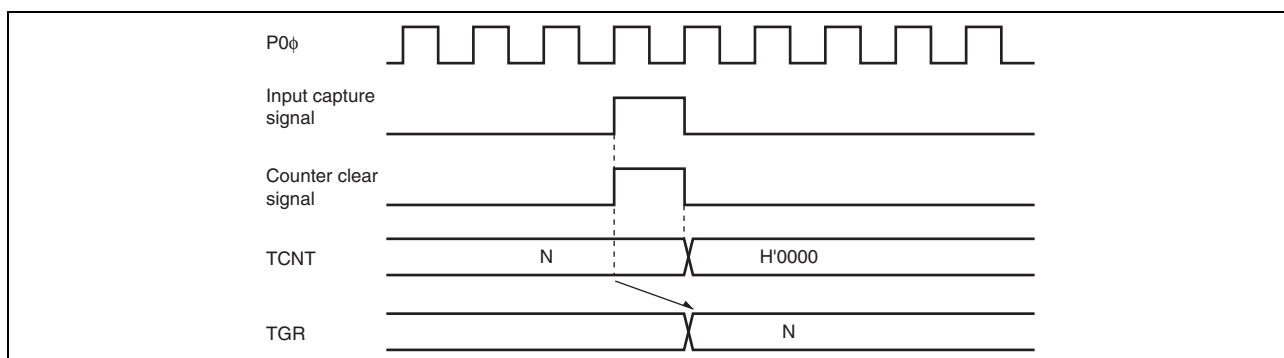
**Figure 10.83 Input Capture Input Signal Timing**

### (4) Timing for Counter Clearing by Compare Match/Input Capture

Figure 10.84 shows the timing when counter clearing on compare match is specified, and Figure 10.85 shows the timing when counter clearing on input capture is specified.



**Figure 10.84 Counter Clear Timing (Compare Match)**



**Figure 10.85 Counter Clear Timing (Input Capture)**



(5) Buffer Operation Timing

Figure 10.86 to Figure 10.88 show the timing in buffer operation.

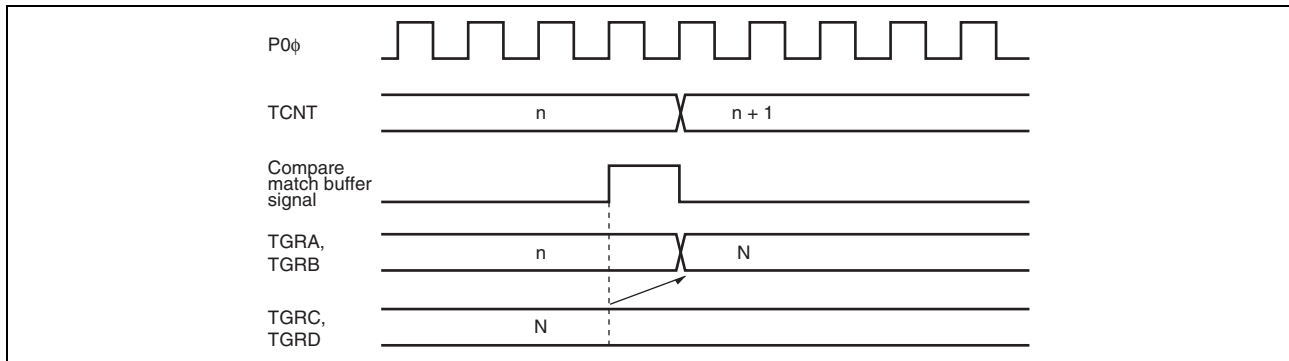


Figure 10.86 Buffer Operation Timing (Compare Match)

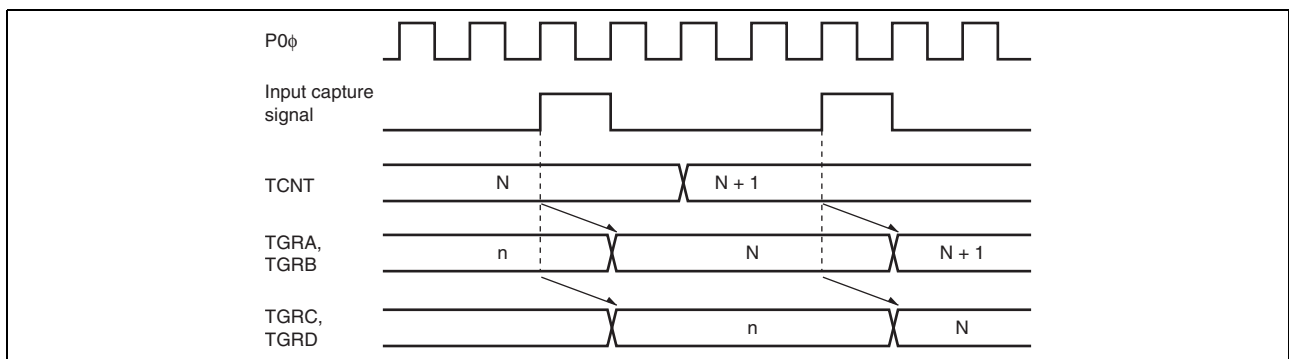


Figure 10.87 Buffer Operation Timing (Input Capture)

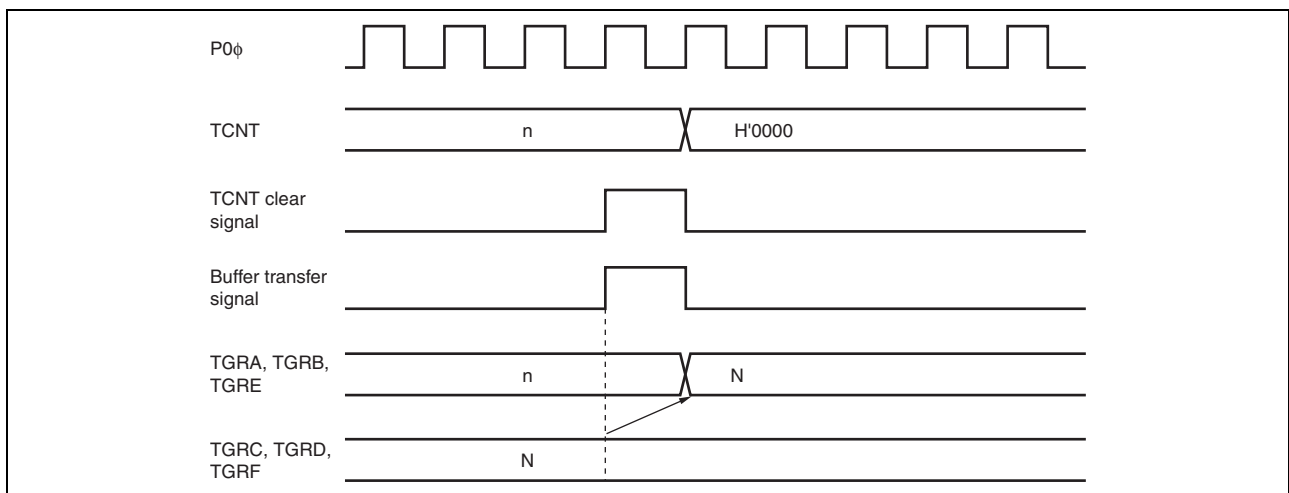


Figure 10.88 Buffer Transfer Timing (when TCNT Cleared)

(6) Buffer Transfer Timing (Complementary PWM Mode)

Figure 10.89 to Figure 10.91 show the buffer transfer timing in complementary PWM mode.

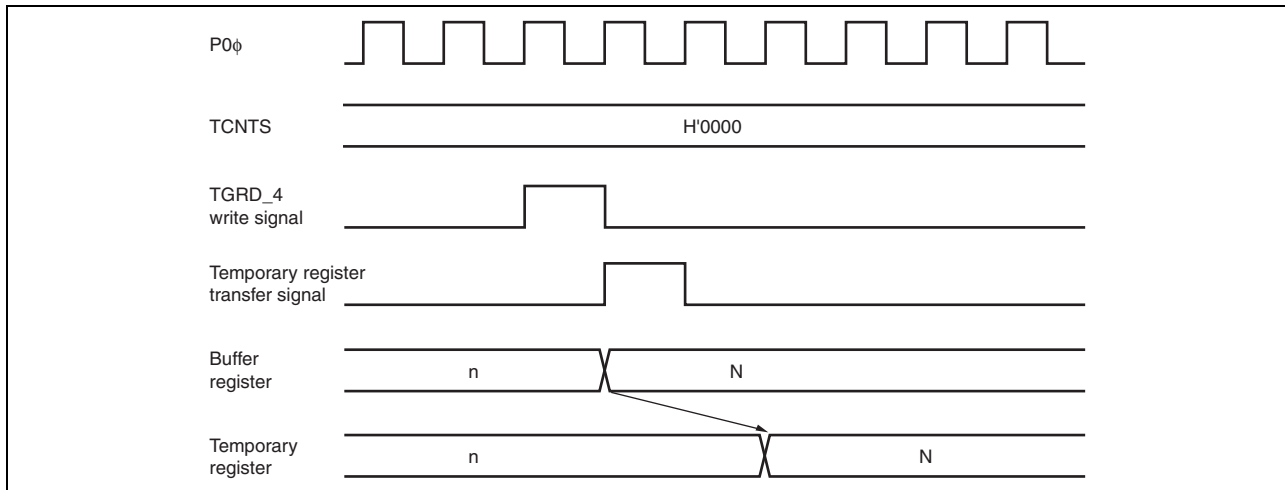


Figure 10.89 Transfer Timing from Buffer Register to Temporary Register (TCNTS Stop)

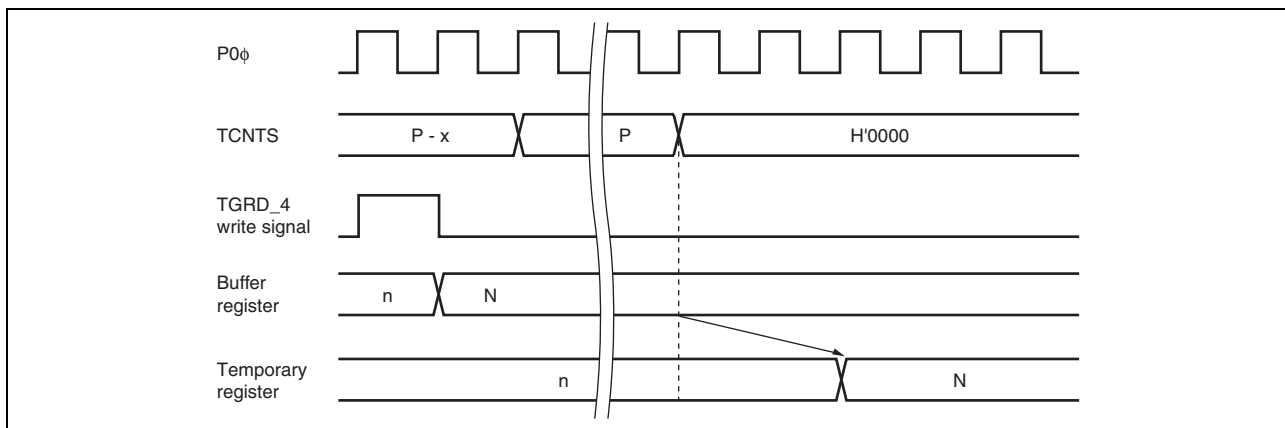


Figure 10.90 Transfer Timing from Buffer Register to Temporary Register (TCNTS Operating)

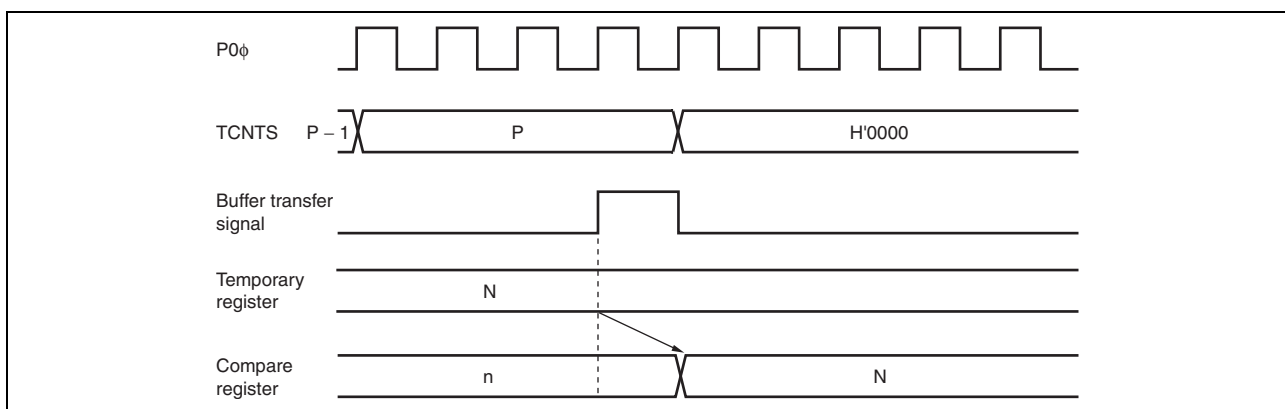
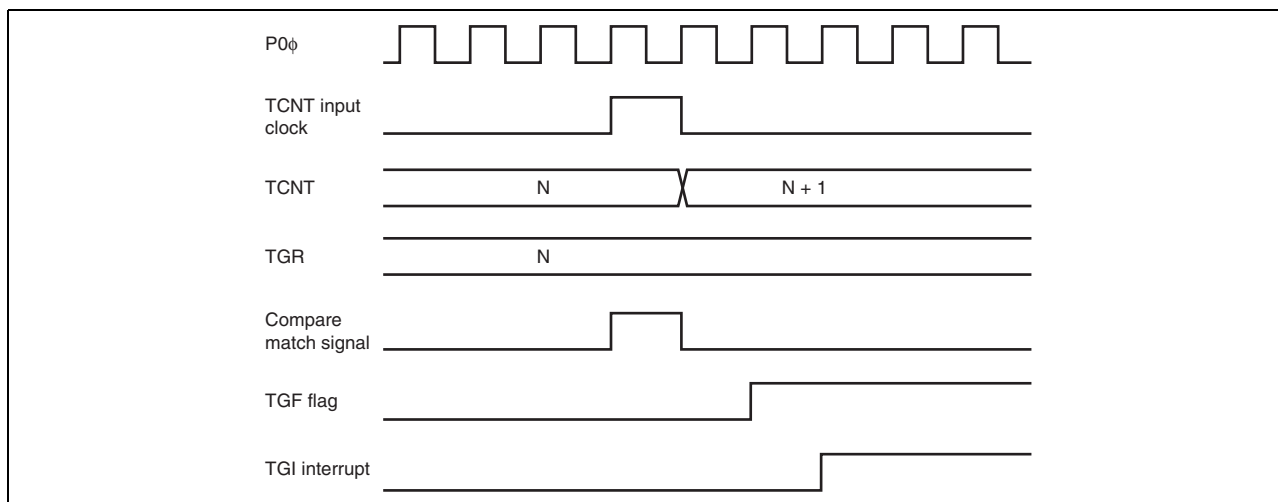


Figure 10.91 Transfer Timing from Temporary Register to Compare Register

### 10.6.2 Interrupt Signal Timing

#### (1) TGF Flag Setting Timing in Case of Compare Match

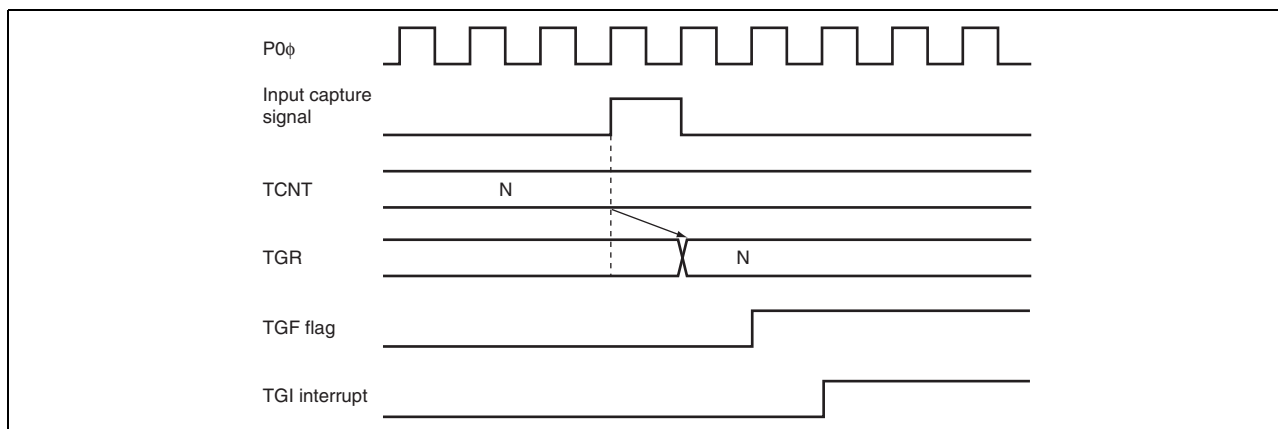
Figure 10.92 shows the timing for setting of the TGF flag in TSR on compare match, and TGI interrupt request signal timing.



**Figure 10.92 TGI Interrupt Timing (Compare Match)**

#### (2) TGF Flag Setting Timing in Case of Input Capture

Figure 10.93 shows the timing for setting of the TGF flag in TSR on input capture, and TGI interrupt request signal timing.



**Figure 10.93 TGI Interrupt Timing (Input Capture)**

(3) TCFV Flag/TCFU Flag Setting Timing

Figure 10.94 shows the timing for setting of the TCFV flag in TSR on overflow, and TCIV interrupt request signal timing.

Figure 10.95 shows the timing for setting of the TCFU flag in TSR on underflow, and TCIU interrupt request signal timing.

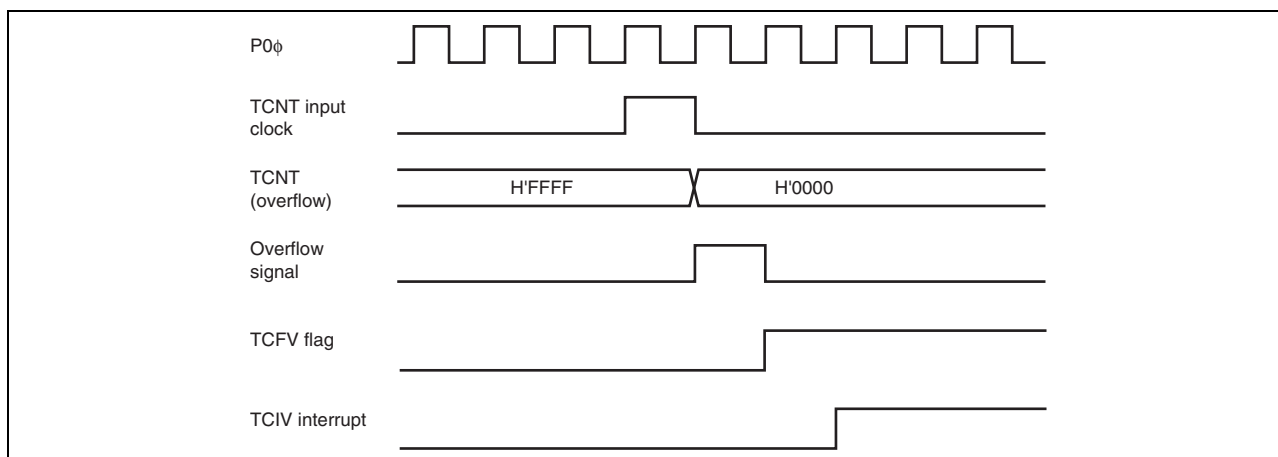


Figure 10.94 TCIV Interrupt Setting Timing

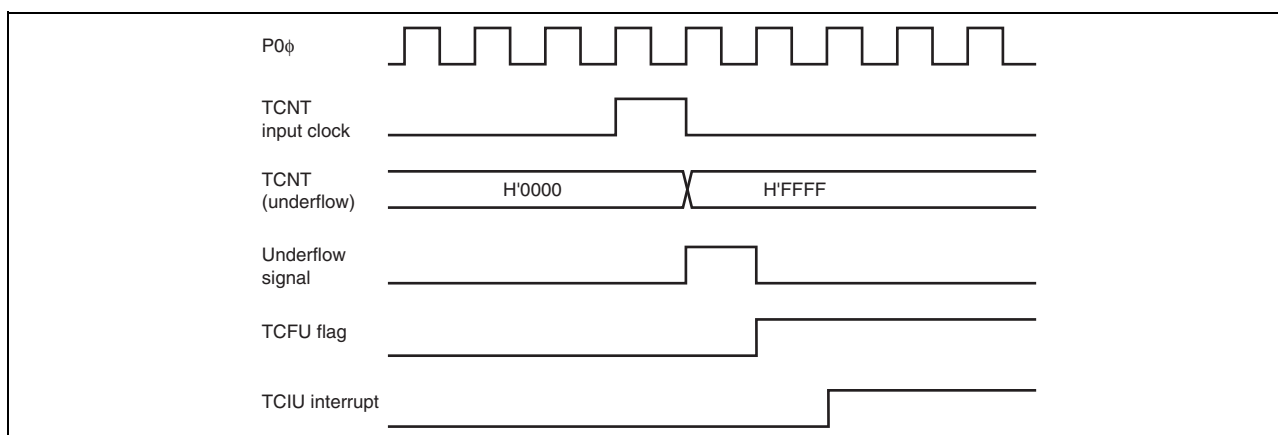
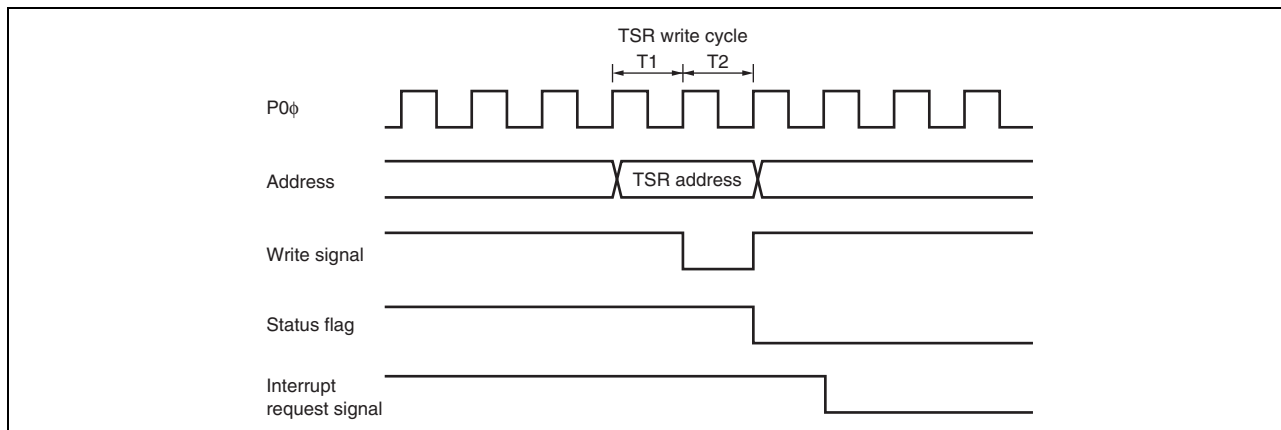


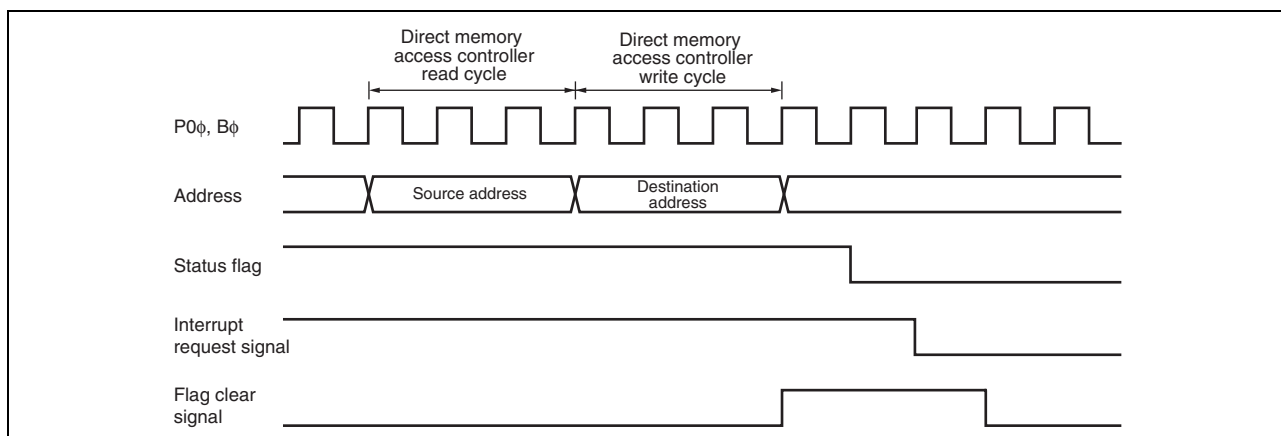
Figure 10.95 TCIU Interrupt Setting Timing

#### (4) Status Flag Clearing Timing

After a status flag is read as 1 by the CPU, it is cleared by writing 0 to it. When the direct memory access controller is activated, the flag is cleared automatically. Figure 10.96 shows the timing for status flag clearing by the CPU, and Figure 10.97 shows the timing for status flag clearing by the direct memory access controller.



**Figure 10.96** Timing for Status Flag Clearing by CPU



**Figure 10.97** Timing for Status Flag Clearing by Direct Memory Access Controller Activation

## 10.7 Usage Notes

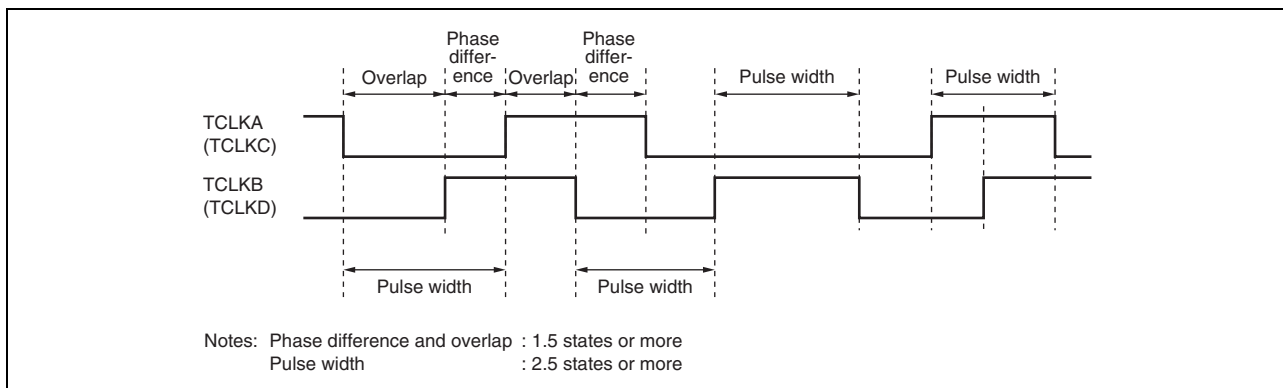
### 10.7.1 Module Standby Mode Setting

Operation of this module can be disabled or enabled using the standby control register. The initial setting is for the operation to be halted. Register access is enabled by clearing module standby mode. For details, refer to [section 55, Power-Down Modes](#).

### 10.7.2 Input Clock Restrictions

The input clock pulse width must be at least 1.5 states in the case of single-edge detection, and at least 2.5 states in the case of both-edge detection. This module will not operate properly at narrower pulse widths.

In phase counting mode, the phase difference and overlap between the two input clocks must be at least 1.5 states, and the pulse width must be at least 2.5 states. [Figure 10.98](#) shows the input clock conditions in phase counting mode.



**Figure 10.98** Phase Difference, Overlap, and Pulse Width in Phase Counting Mode

### 10.7.3 Caution on Period Setting

When counter clearing on compare match is set, TCNT is cleared in the final state in which it matches the TGR value (the point at which the count value matched by TCNT is updated). Consequently, the actual counter frequency is given by the following formula:

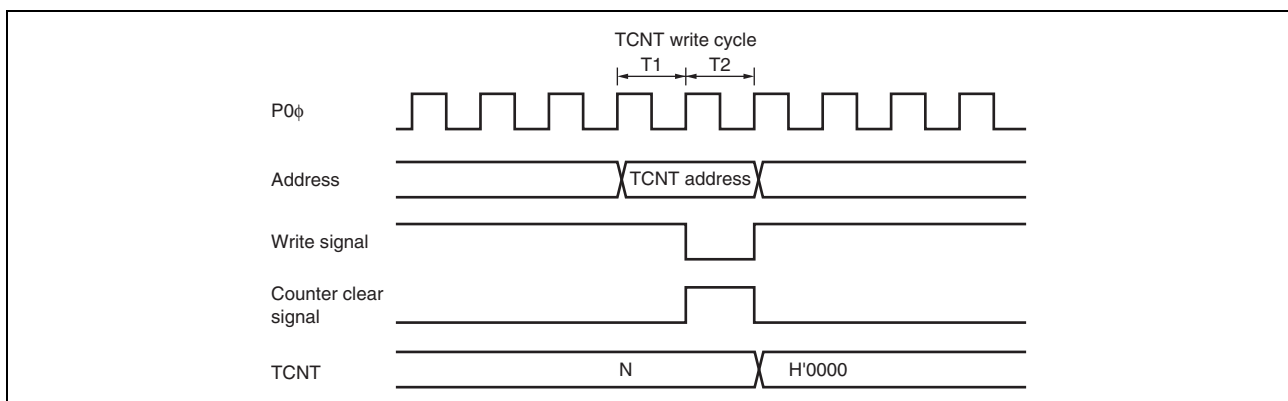
$$f = \frac{P0\phi}{(N+1)}$$

Where f: Counter frequency  
 P0φ: Peripheral clock operating frequency  
 N: TGR set value

### 10.7.4 Contention between TCNT Write and Clear Operations

If the counter clear signal is generated in the T2 state of a TCNT write cycle, TCNT clearing takes precedence and the TCNT write is not performed.

Figure 10.99 shows the timing in this case.

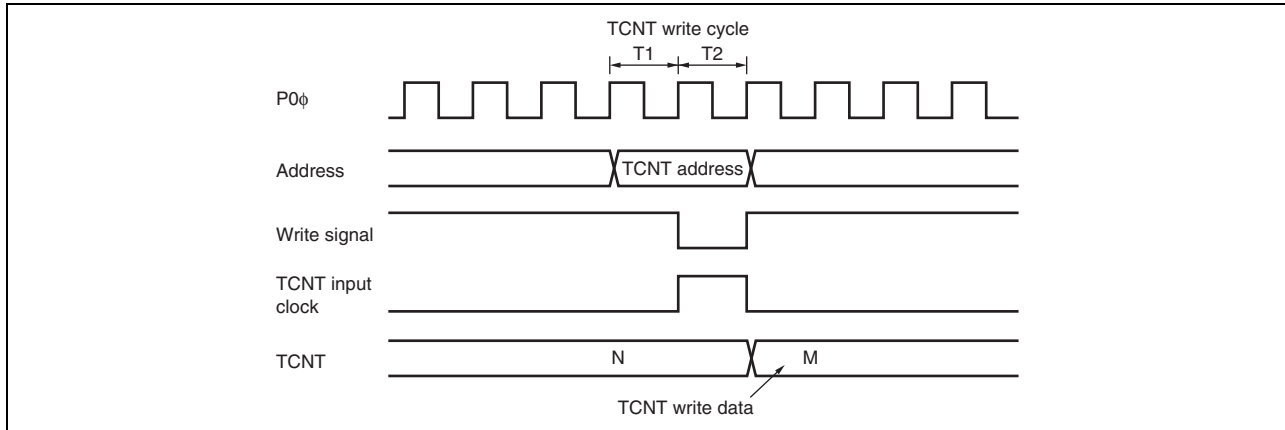


**Figure 10.99** Contention between TCNT Write and Clear Operations

### 10.7.5 Contention between TCNT Write and Increment Operations

If incrementing occurs in the T2 state of a TCNT write cycle, the TCNT write takes precedence and TCNT is not incremented.

Figure 10.100 shows the timing in this case.

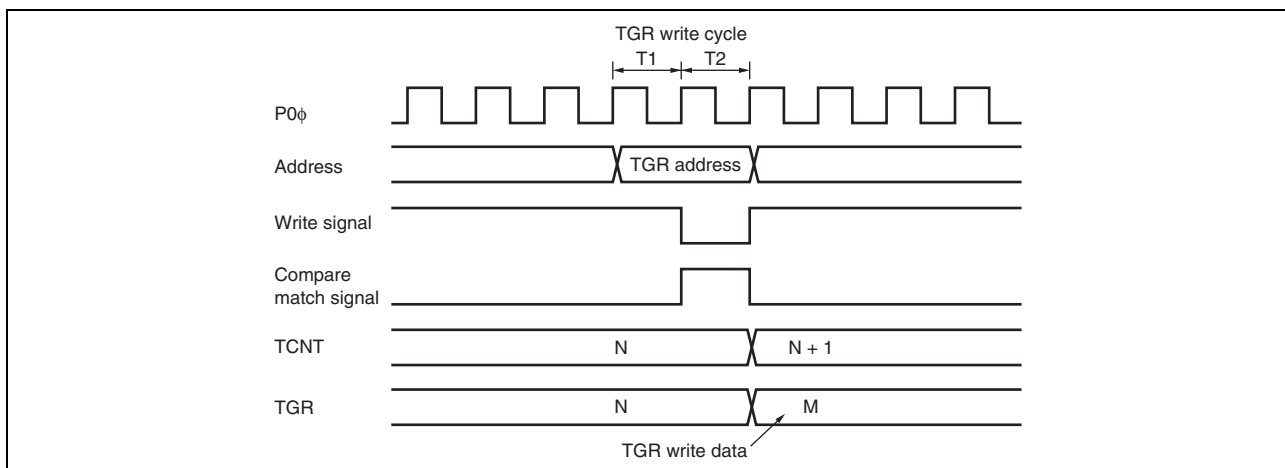


**Figure 10.100 Contention between TCNT Write and Increment Operations**

### 10.7.6 Contention between TGR Write and Compare Match

If a compare match occurs in the T2 state of a TGR write cycle, the TGR write is executed and the compare match signal is also generated.

Figure 10.101 shows the timing in this case.



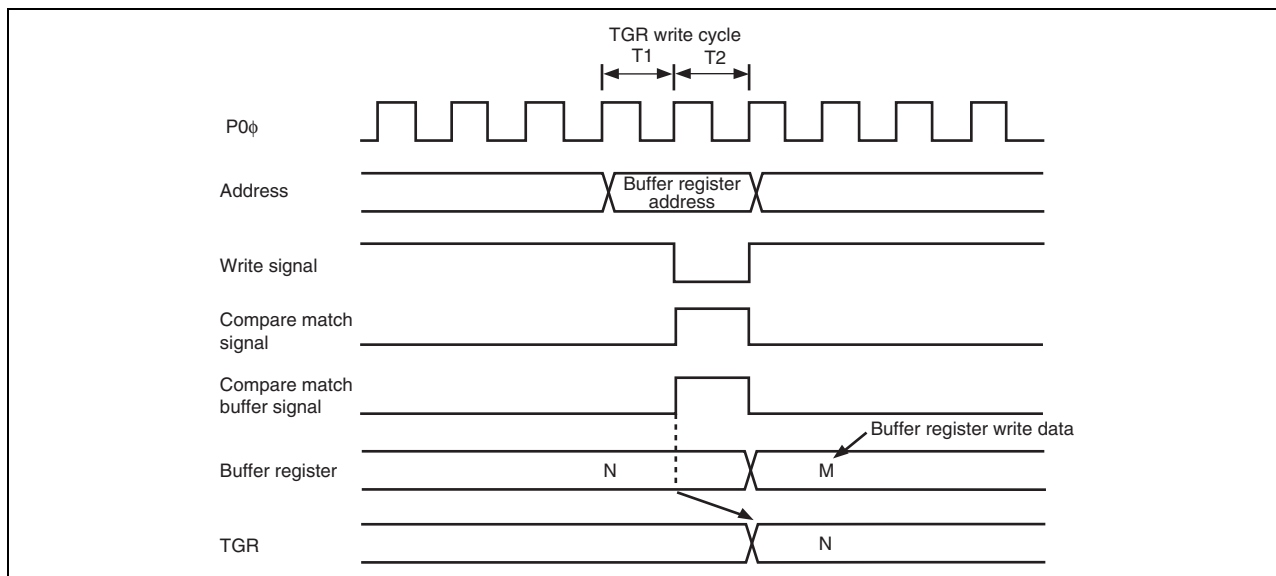
**Figure 10.101 Contention between TGR Write and Compare Match**



### 10.7.7 Contention between Buffer Register Write and Compare Match

If a compare match occurs in the T2 state of a TGR write cycle, the data that is transferred to TGR by the buffer operation is the data after write.

Figure 10.102 shows the timing in this case.

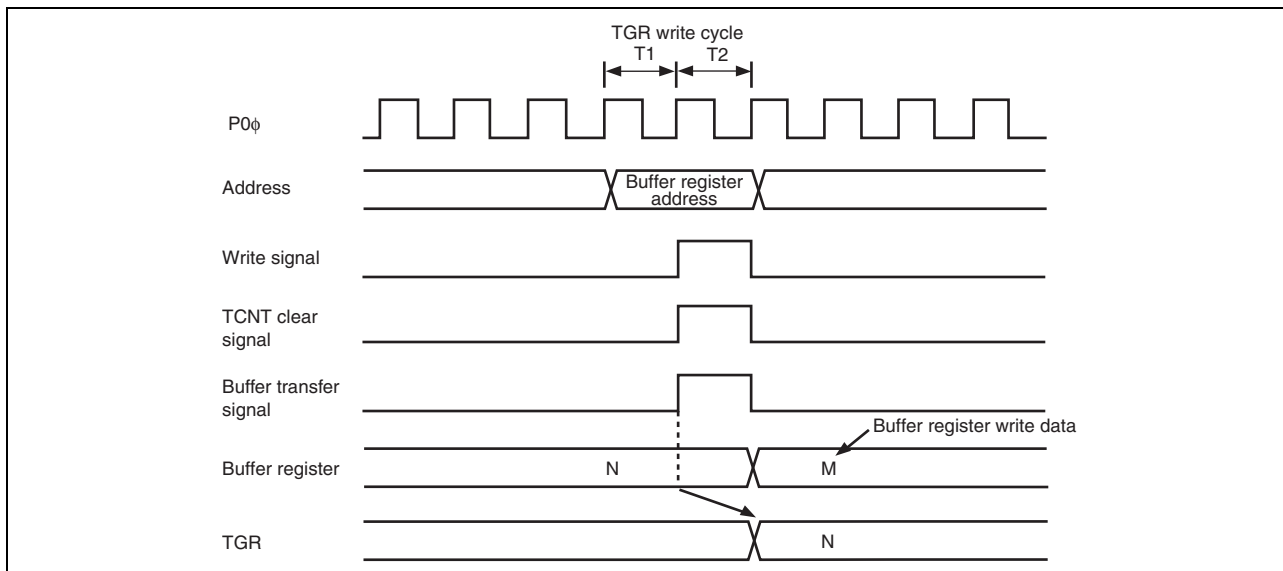


**Figure 10.102 Contention between Buffer Register Write and Compare Match**

### 10.7.8 Contention between Buffer Register Write and TCNT Clear

When the buffer transfer timing is set at the TCNT clear by the buffer transfer mode register (TBTM), if TCNT clear occurs in the T2 state of a TGR write cycle, the data that is transferred to TGR by the buffer operation is the data before write.

Figure 10.103 shows the timing in this case.

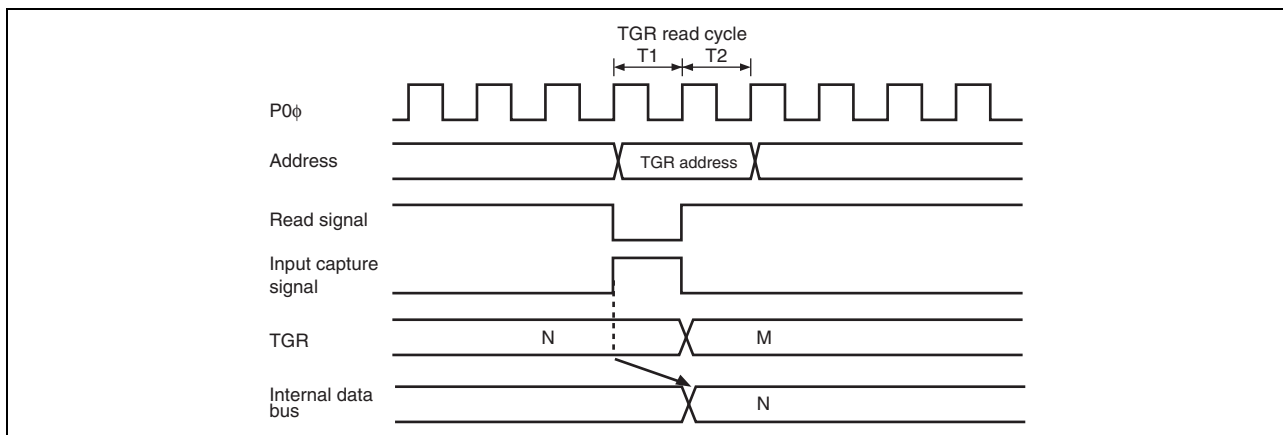


**Figure 10.103 Contention between Buffer Register Write and TCNT Clear**

### 10.7.9 Contention between TGR Read and Input Capture

If an input capture signal is generated in the T1 state of a TGR read cycle, the data that is read will be the data in the buffer before input capture transfer.

Figure 10.104 shows the timing in this case.

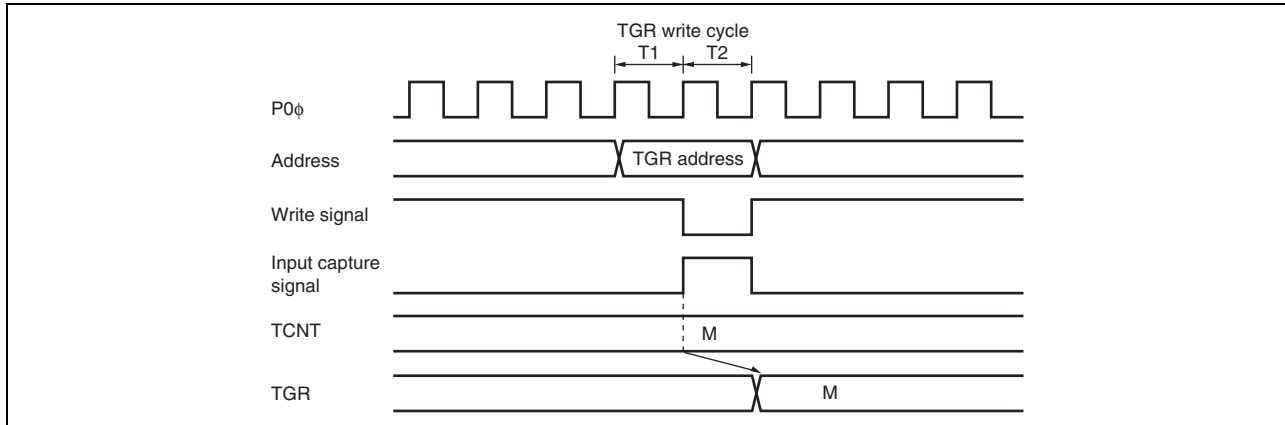


**Figure 10.104 Contention between TGR Read and Input Capture**

### 10.7.10 Contention between TGR Write and Input Capture

If an input capture signal is generated in the T2 state of a TGR write cycle, the input capture operation takes precedence and the write to TGR is not performed.

Figure 10.105 shows the timing in this case.

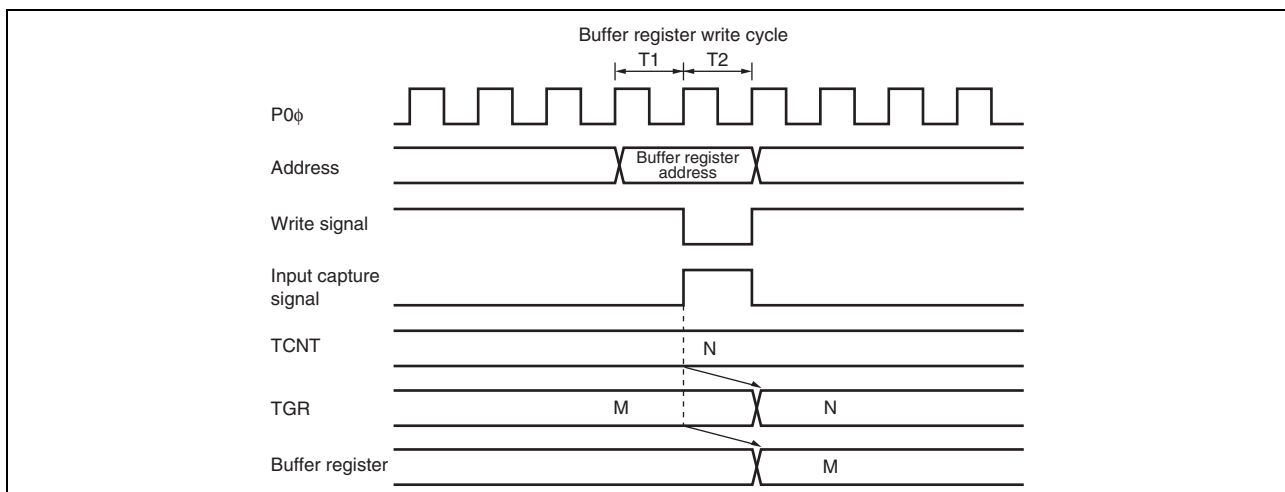


**Figure 10.105 Contention between TGR Write and Input Capture**

### 10.7.11 Contention between Buffer Register Write and Input Capture

If an input capture signal is generated in the T2 state of a buffer register write cycle, the buffer operation takes precedence and the write to the buffer register is not performed.

Figure 10.106 shows the timing in this case.



**Figure 10.106 Contention between Buffer Register Write and Input Capture**

### 10.7.12 TCNT2 Write and Overflow/Underflow Contention in Cascade Connection

With timer counters TCNT1 and TCNT2 in a cascade connection, when a contention occurs during TCNT\_1 count (during a TCNT\_2 overflow/underflow) in the T<sub>2</sub> state of the TCNT\_2 write cycle, the write to TCNT\_2 is conducted, and the TCNT\_1 count signal is disabled. At this point, if there is match with TGRA\_1 and the TCNT\_1 value, a compare signal is issued. Furthermore, when the TCNT\_1 count clock is selected as the input capture source of channel 0, TGRA\_0 to D\_0 carry out the input capture operation. In addition, when the compare match/input capture is selected as the input capture source of TGRB\_1, TGRB\_1 carries out input capture operation. The timing is shown in Figure 10.107.

For cascade connections, be sure to synchronize settings for channels 1 and 2 when setting TCNT clearing.

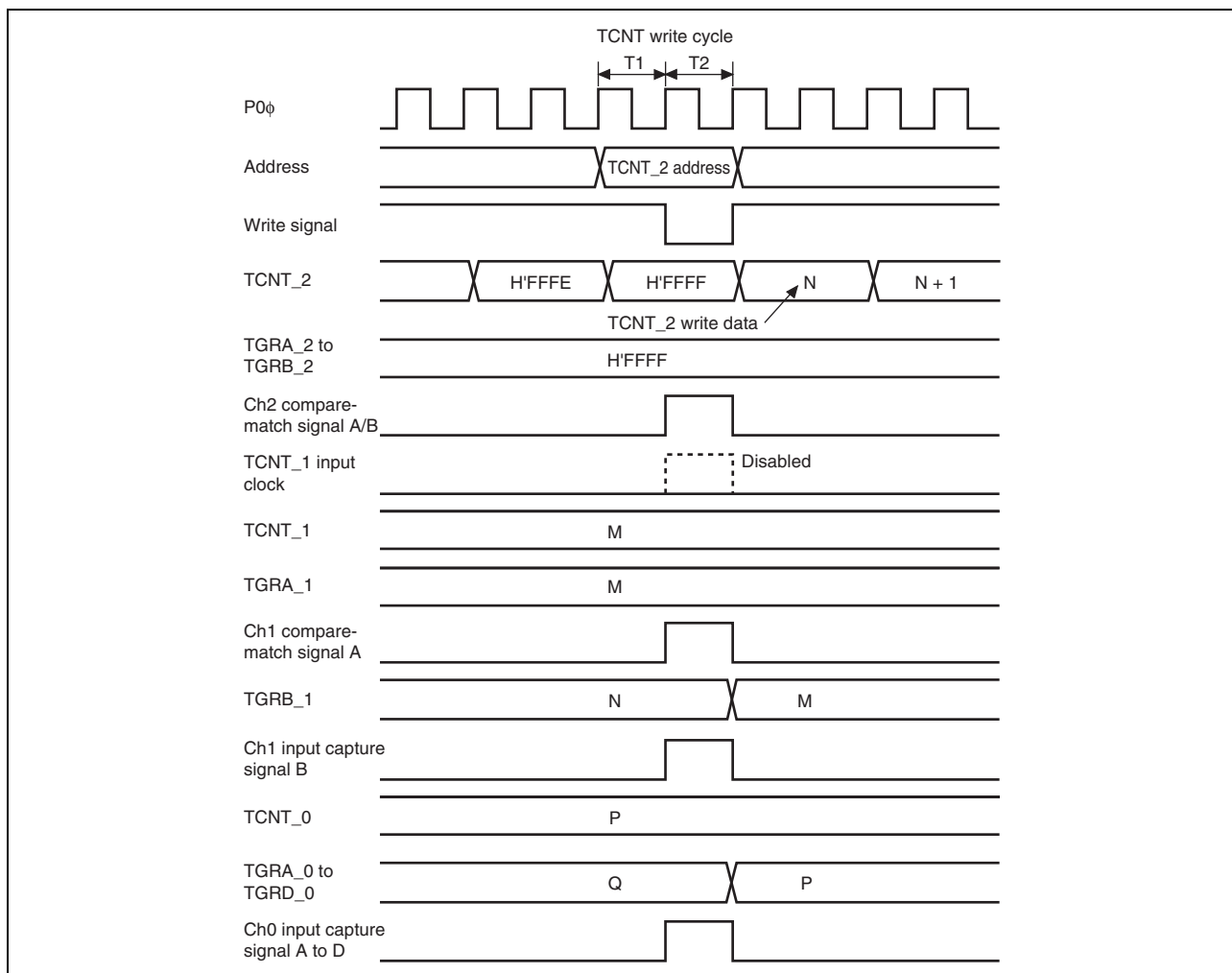


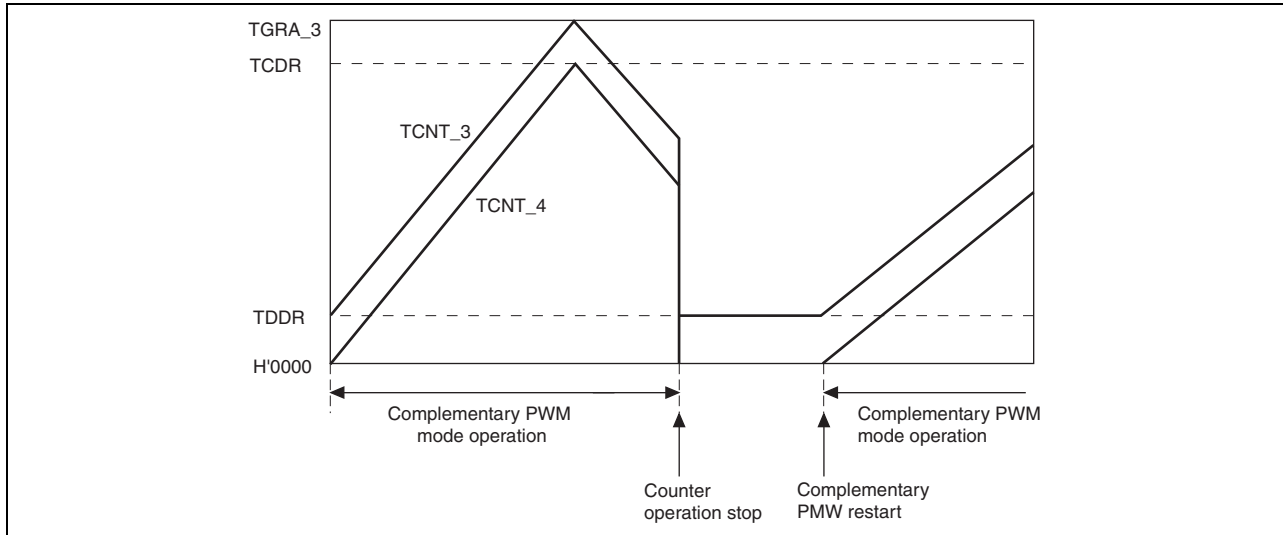
Figure 10.107 TCNT\_2 Write and Overflow/Underflow Contention with Cascade Connection

### 10.7.13 Counter Value during Complementary PWM Mode Stop

When counting operation is suspended with TCNT\_3 and TCNT\_4 in complementary PWM mode, TCNT\_3 has the timer dead time register (TDDR) value, and TCNT\_4 is held at H'0000.

When restarting complementary PWM mode, counting begins automatically from the initialized state. This explanatory diagram is shown in Figure 10.108.

When counting begins in another operating mode, be sure that TCNT\_3 and TCNT\_4 are set to the initial values.



**Figure 10.108 Counter Value during Complementary PWM Mode Stop**

### 10.7.14 Buffer Operation Setting in Complementary PWM Mode

In complementary PWM mode, conduct rewrites by buffer operation for the PWM cycle setting register (TGRA\_3), timer cycle data register (TCDR), and duty setting registers (TGRB\_3, TGRA\_4, and TGRB\_4).

In complementary PWM mode, channel 3 and channel 4 buffers operate in accordance with bit settings BFA and BFB of TMDR\_3. When TMDR\_3's BFA bit is set to 1, TGRC\_3 functions as a buffer register for TGRA\_3. At the same time, TGRC\_4 functions as the buffer register for TGRA\_4, and TCBR functions as the TCDR's buffer register.

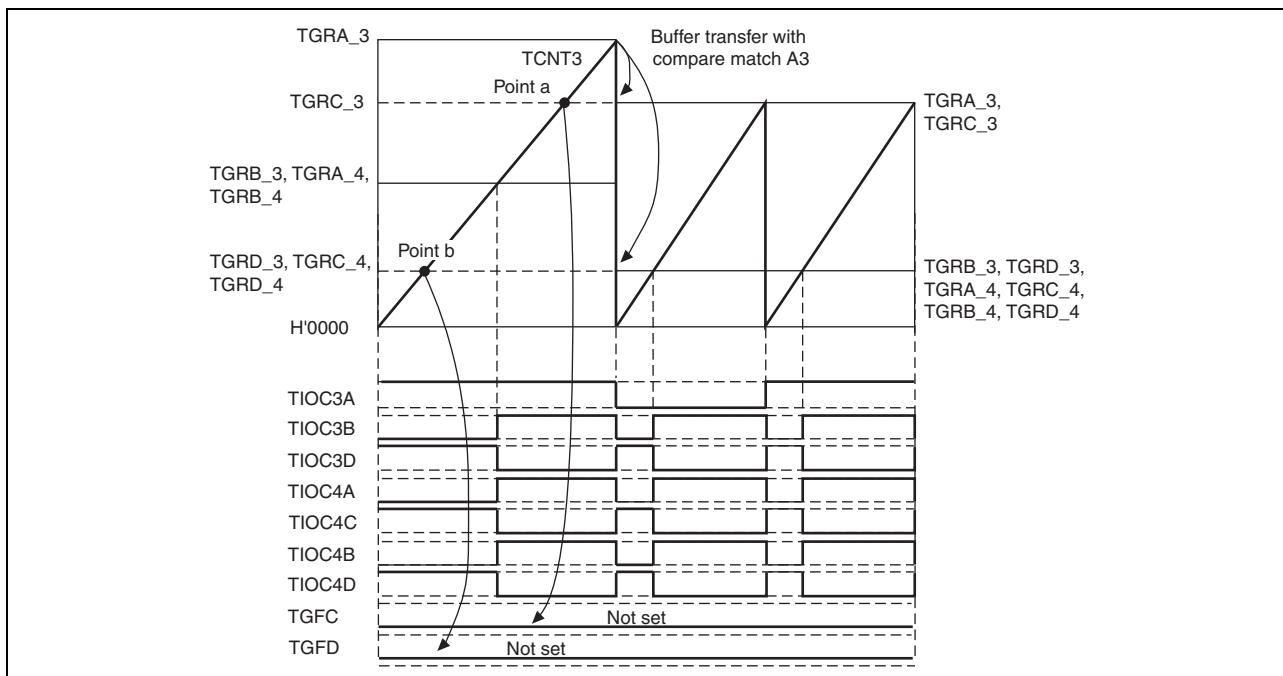
### 10.7.15 Reset Sync PWM Mode Buffer Operation and Compare Match Flag

When setting buffer operation for reset sync PWM mode, set the BFA and BFB bits of TMDR\_4 to 0. The TIOC4C pin will be unable to produce its waveform output if the BFA bit of TMDR\_4 is set to 1.

In reset sync PWM mode, the channel 3 and channel 4 buffers operate in accordance with the BFA and BFB bit settings of TMDR\_3. For example, if the BFA bit of TMDR\_3 is set to 1, TGRC\_3 functions as the buffer register for TGRA\_3. At the same time, TGRC\_4 functions as the buffer register for TGRA\_4.

The TGFC bit and TGFD bit of TSR\_3 and TSR\_4 are not set when TGRC\_3 and TGRD\_3 are operating as buffer registers.

Figure 10.109 shows an example of operations for TGR\_3, TGR\_4, TIOC3, and TIOC4, with TMDR\_3's BFA and BFB bits set to 1, and TMDR\_4's BFA and BFB bits set to 0.



**Figure 10.109 Buffer Operation and Compare-Match Flags in Reset Synchronous PWM Mode**

### 10.7.16 Overflow Flags in Reset Synchronous PWM Mode

When set to reset synchronous PWM mode, TCNT\_3 and TCNT\_4 start counting when the CST3 bit of TSTR is set to 1. At this point, TCNT\_4's count clock source and count edge obey the TCR\_3 setting.

In reset synchronous PWM mode, with cycle register TGRA\_3's set value at H'FFFF, when specifying TGR3A compare-match for the counter clear source, TCNT\_3 and TCNT\_4 count up to H'FFFF, then a compare-match occurs with TGRA\_3, and TCNT\_3 and TCNT\_4 are both cleared. At this point, TSR's overflow flag TCFV bit is not set.

Figure 10.110 shows a TCFV bit operation example in reset synchronous PWM mode with a set value for cycle register TGRA\_3 of H'FFFF, when a TGRA\_3 compare-match has been specified without synchronous setting for the counter clear source.

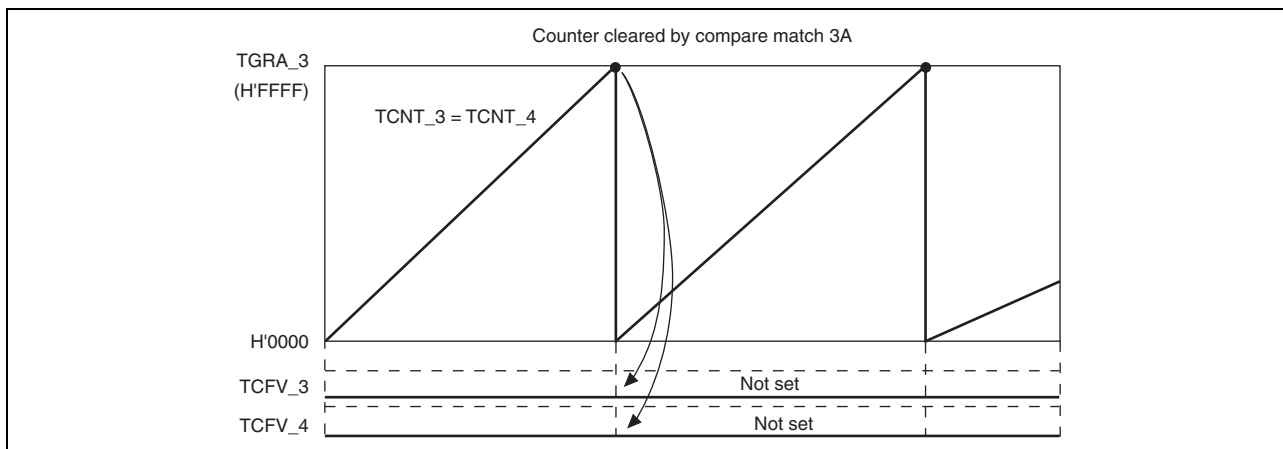


Figure 10.110 Reset Synchronous PWM Mode Overflow Flag

### 10.7.17 Contention between Overflow/Underflow and Counter Clearing

If overflow/underflow and counter clearing occur simultaneously, the TCFV/TCFU flag in TSR is not set and TCNT clearing takes precedence.

Figure 10.111 shows the operation timing when a TGR compare match is specified as the clearing source, and when H'FFFF is set in TGR.

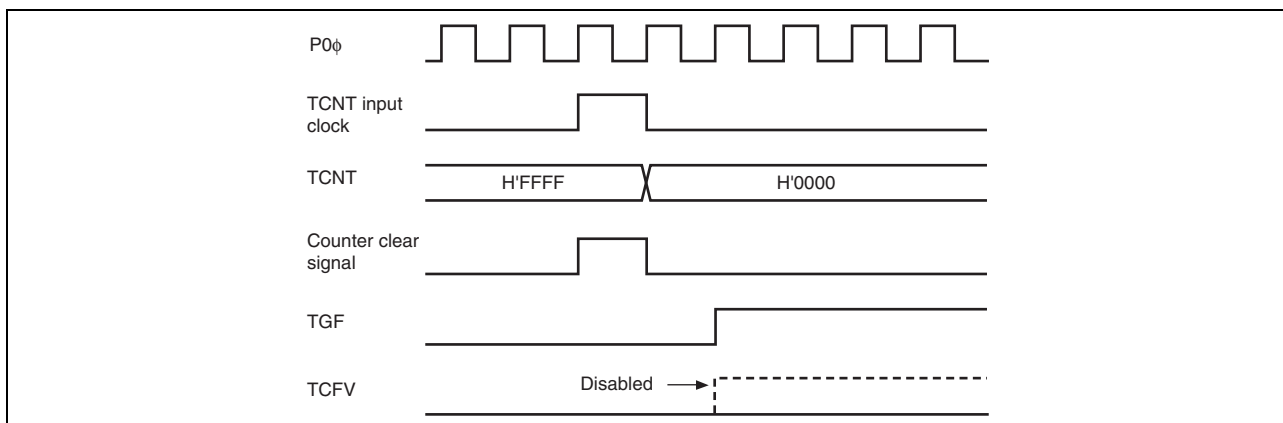
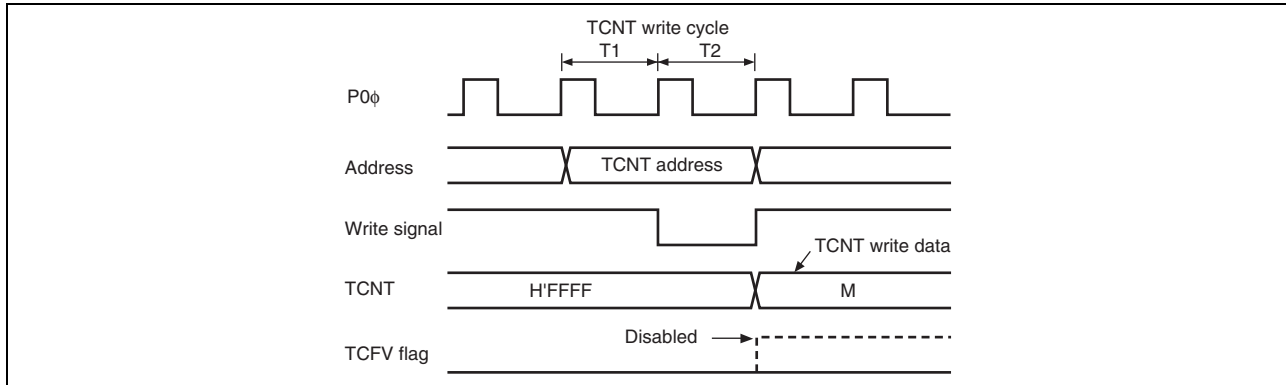


Figure 10.111 Contention between Overflow and Counter Clearing

### 10.7.18 Contention between TCNT Write and Overflow/Underflow

If there is an up-count or down-count in the T2 state of a TCNT write cycle, and overflow/underflow occurs, the TCNT write takes precedence and the TCFV/TCFU flag in TSR is not set.

Figure 10.112 shows the operation timing when there is contention between TCNT write and overflow.



**Figure 10.112 Contention between TCNT Write and Overflow**

### 10.7.19 Cautions on Transition from Normal Operation or PWM Mode 1 to Reset-Synchronized PWM Mode

When making a transition from channel 3 or 4 normal operation or PWM mode 1 to reset-synchronized PWM mode, if the counter is halted with the output pins (TIOC3B, TIOC3D, TIOC4A, TIOC4C, TIOC4B, TIOC4D) in the high-level state, followed by the transition to reset-synchronized PWM mode and operation in that mode, the initial pin output will not be correct.

When making a transition from normal operation to reset-synchronized PWM mode, write H'11 to registers TIORH\_3, TIORL\_3, TIORH\_4, and TIORL\_4 to initialize the output pins to low level output, then set an initial register value of H'00 before making the mode transition.

When making a transition from PWM mode 1 to reset-synchronized PWM mode, first switch to normal operation, then initialize the output pins to low level output and set an initial register value of H'00 before making the transition to reset-synchronized PWM mode.

### 10.7.20 Output Level in Complementary PWM Mode and Reset-Synchronized PWM Mode

When channels 3 and 4 are in complementary PWM mode or reset-synchronized PWM mode, the PWM waveform output level is set with the OLSP and OLSN bits in the timer output control register (TOCR). In the case of complementary PWM mode or reset-synchronized PWM mode, TIOR should be set to H'00.

### 10.7.21 Interrupts in Module Standby Mode

If module standby mode is entered when an interrupt has been requested, it will not be possible to clear the CPU interrupt source or the direct memory access controller activation source. Interrupts should therefore be disabled before entering module standby mode.



### 10.7.22 Simultaneous Capture of TCNT\_1 and TCNT\_2 in Cascade Connection

When timer counters 1 and 2 (TCNT\_1 and TCNT\_2) are operated as a 32-bit counter in cascade connection, the cascade counter value cannot be captured successfully even if input-capture input is simultaneously done to TIOC1A and TIOC2A or to TIOC1B and TIOC2B. This is because the input timing of TIOC1A and TIOC2A or of TIOC1B and TIOC2B may not be the same when external input-capture signals to be input into TCNT\_1 and TCNT\_2 are taken in synchronization with the internal clock. For example, TCNT\_1 (the counter for upper 16 bits) does not capture the count-up value by overflow from TCNT\_2 (the counter for lower 16 bits) but captures the count value before the count-up. In this case, the values of TCNT\_1 = H'FFF1 and TCNT\_2 = H'0000 should be transferred to TGRA\_1 and TGRA\_2 or to TGRB\_1 and TGRB\_2, but the values of TCNT\_1 = H'FFF0 and TCNT\_2 = H'0000 are erroneously transferred.

### 10.7.23 Notes on Output Waveform Control During Synchronous Counter Clearing in Complementary PWM Mode

In complementary PWM mode, when output waveform control during synchronous counter clearing is enabled (WRE in the TWCR register set to 1), the following problems may occur when condition (1) or condition (2), below, is satisfied.

- Dead time for the PWM output pins may be too short (or nonexistent).
- Active-level output from the PWM negative-phase pins may occur outside the correct active-level output interval

Condition (1): When synchronous clearing occurs in the PWM output dead time interval within initial output suppression interval (10) (Figure 10.113).

Condition (2): When synchronous clearing occurs within initial output suppression interval (10) or (11) and  $TGRB_3 \leq TDDR$ ,  $TGRA_4 \leq TDDR$ , or  $TGRB_4 \leq TDDR$  is true (Figure 10.114)

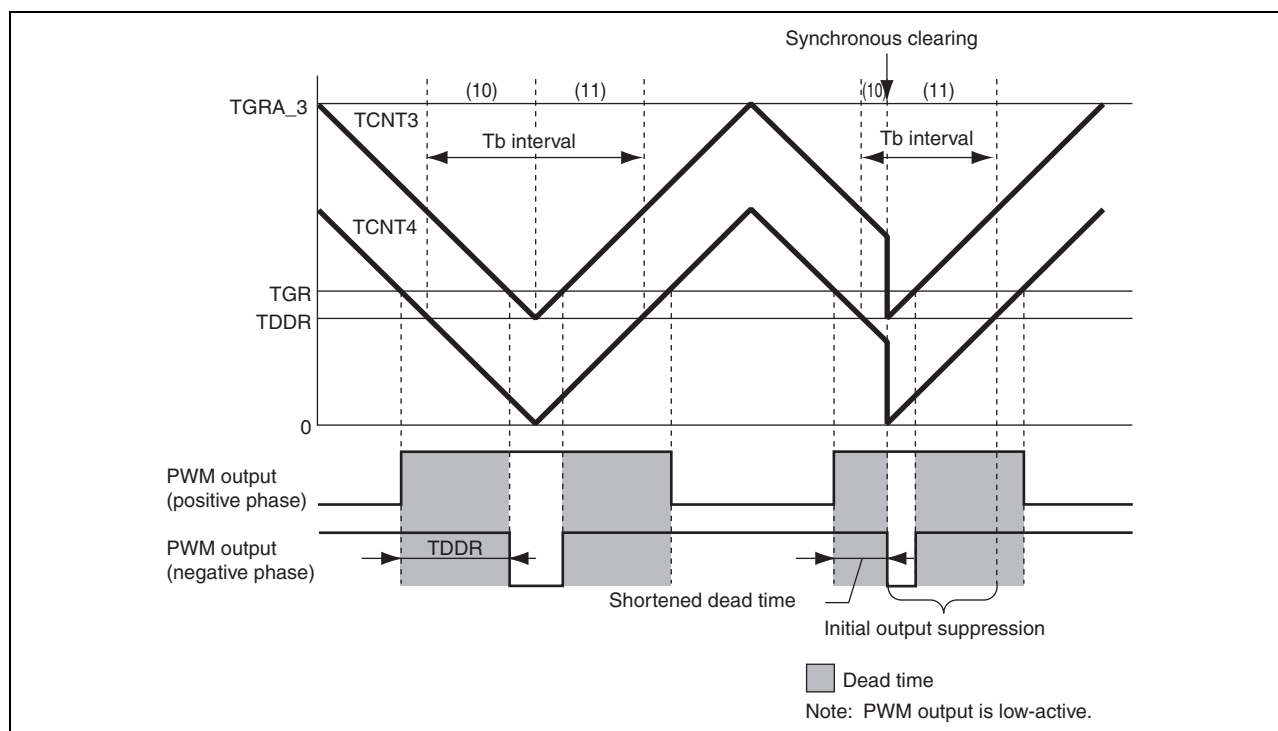
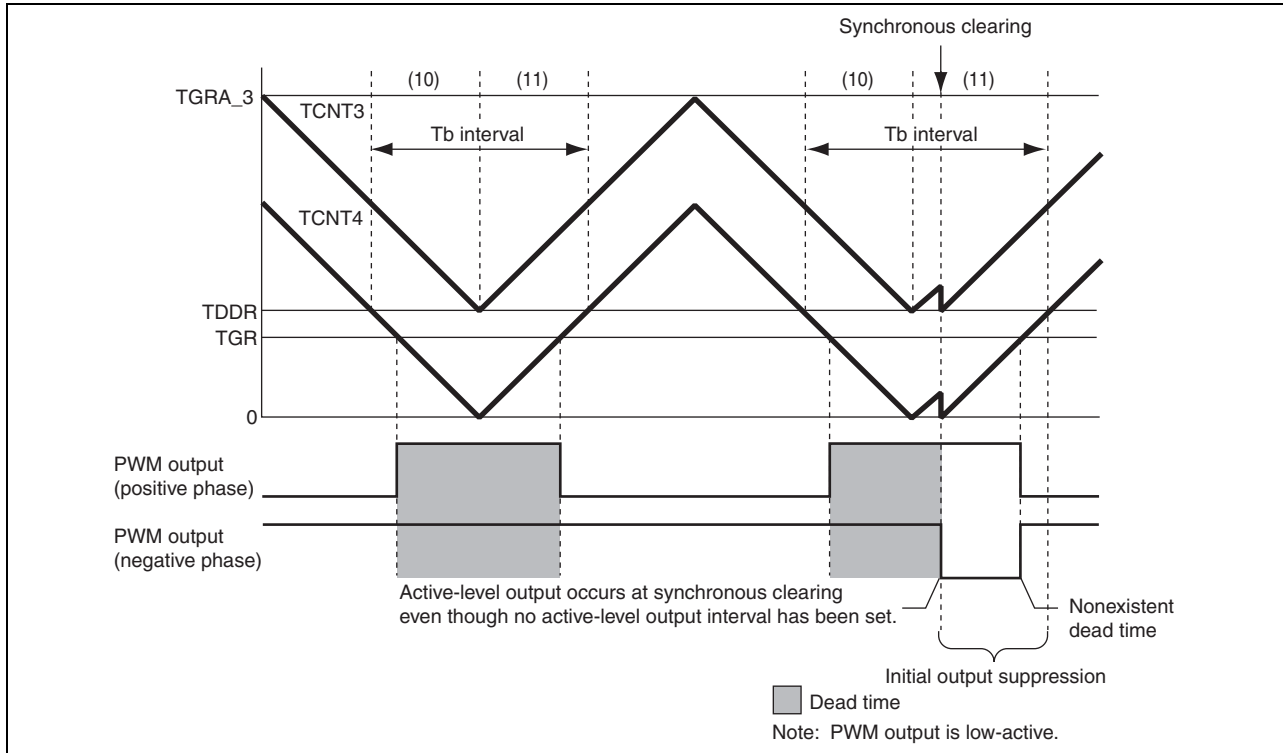


Figure 10.113 Condition (1) Synchronous Clearing Example



**Figure 10.114 Condition (2) Synchronous Clearing Example**

The following workaround can be used to avoid these problems.

When using synchronous clearing, make sure to set compare registers TGRB\_3, TGRA\_4, and TGRB\_4 to a value twice or more the setting of dead time data register TDDR.

## 10.8 Output Pin Initialization for Multi-Function Timer Pulse Unit 2

### 10.8.1 Operating Modes

This module has the following six operating modes. Waveform output is possible in all of these modes.

- Normal mode (channels 0 to 4)
- PWM mode 1 (channels 0 to 4)
- PWM mode 2 (channels 0 to 2)
- Phase counting modes 1 to 4 (channels 1 and 2)
- Complementary PWM mode (channels 3 and 4)
- Reset-synchronized PWM mode (channels 3 and 4)

The output pin initialization method for each of these modes is described in this section.

### 10.8.2 Reset Start Operation

The output pins of this module (TIOC\*) are initialized low by a reset and in standby mode. Since the pin functions are selected using the general I/O ports, when the general I/O port is set, the pin states at that point are output to the ports. When this module output is selected by the general I/O port immediately after a reset, the initial output level, low, is output directly at the port. When the active level is low, the system will operate at this point, and therefore the general I/O port setting should be made after the initialization of the output pins is completed.

Note: Channel number and port notation are substituted for \*.

### 10.8.3 Operation in Case of Re-Setting Due to Error during Operation, etc.

If an error occurs during operation of this module, the module output should be cut by the system. Cutoff is performed by switching the pin output to port output with the general I/O port and outputting the inverse of the active level. The pin initialization procedures for re-setting due to an error during operation, etc., and the procedures for restarting in a different mode after re-setting, are shown below.

This module has six operating modes, as stated above. There are thus 36 mode transition combinations, but some transitions are not available with certain channel and mode combinations. Possible mode transition combinations are shown in Table 10.57.

**Table 10.57 Mode Transition Combinations**

Before	After					
	Normal	PWM1	PWM2	PCM	CPWM	RPWM
Normal	(1)	(2)	(3)	(4)	(5)	(6)
PWM1	(7)	(8)	(9)	(10)	(11)	(12)
PWM2	(13)	(14)	(15)	(16)	None	None
PCM	(17)	(18)	(19)	(20)	None	None
CPWM	(21)	(22)	None	None	(23) (24)	(25)
RPWM	(26)	(27)	None	None	(28)	(29)

[Legend]

Normal: Normal mode

PWM1: PWM mode 1

PWM2: PWM mode 2

PCM: Phase counting modes 1 to 4

CPWM: Complementary PWM mode

RPWM: Reset-synchronized PWM mode

#### 10.8.4 Overview of Initialization Procedures and Mode Transitions in Case of Error during Operation, etc.

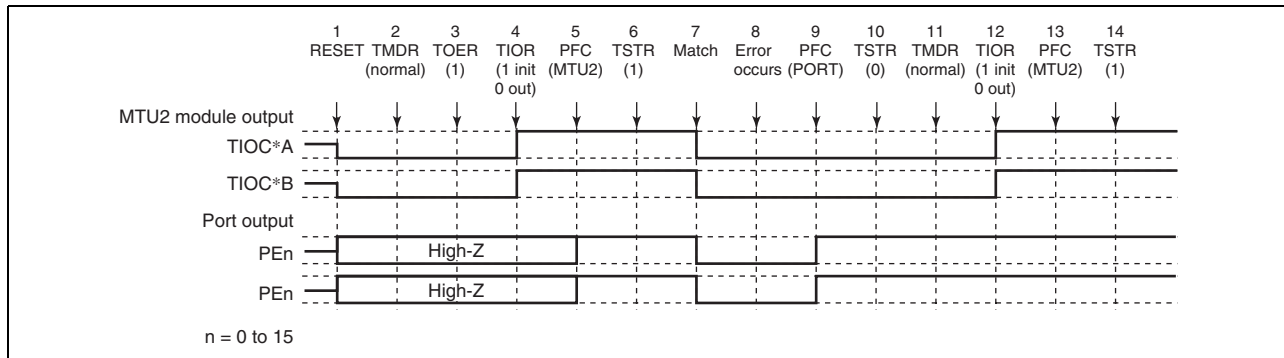
- When making a transition to a mode (Normal, PWM1, PWM2, PCM) in which the pin output level is selected by the timer I/O control register (TIOR) setting, initialize the pins by means of a TIOR setting.
- In PWM mode 1, since a waveform is not output to the TIOC\*B (TIOC \*D) pin, setting TIOR will not initialize the pins. If initialization is required, carry it out in normal mode, then switch to PWM mode 1.
- In PWM mode 2, since a waveform is not output to the cycle register pin, setting TIOR will not initialize the pins. If initialization is required, carry it out in normal mode, then switch to PWM mode 2.
- In normal mode or PWM mode 2, if TGRC and TGRD operate as buffer registers, setting TIOR will not initialize the buffer register pins. If initialization is required, clear buffer mode, carry out initialization, then set buffer mode again.
- In PWM mode 1, if either TGRC or TGRD operates as a buffer register, setting TIOR will not initialize the TGRC pin. To initialize the TGRC pin, clear buffer mode, carry out initialization, then set buffer mode again.
- When making a transition to a mode (CPWM, RPWM) in which the pin output level is selected by the timer output control register (TOCR) setting, switch to normal mode and perform initialization with TIOR, then restore TIOR to its initial value, and temporarily disable channel 3 and 4 output with the timer output master enable register (TOER). Then operate the unit in accordance with the mode setting procedure (TOCR setting, TMDR setting, TOER setting).

Note: Channel number is substituted for \* indicated in this article.

Pin initialization procedures are described below for the numbered combinations in Table 10.57. The active level is assumed to be low.

### (1) Operation when Error Occurs during Normal Mode Operation, and Operation is Restarted in Normal Mode

Figure 10.115 shows an explanatory diagram of the case where an error occurs in normal mode and operation is restarted in normal mode after re-setting.

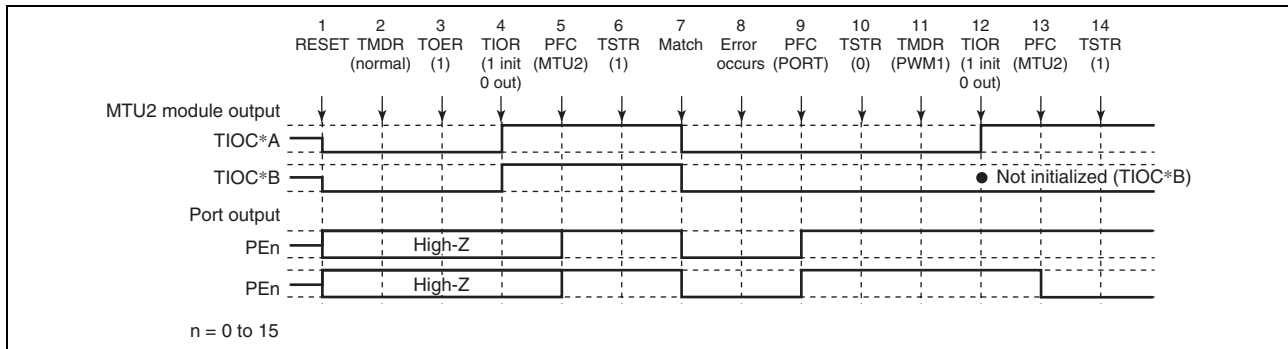


**Figure 10.115 Error Occurrence in Normal Mode, Recovery in Normal Mode**

1. After a reset, the module output is low and ports are in the high-impedance state.
2. After a reset, the TMDR setting is for normal mode.
3. For channels 3 and 4, enable output with TOER before initializing the pins with TIOR.
4. Initialize the pins with TIOR. (The example shows initial high output, with low output on compare-match occurrence.)
5. Set the multi-function timer pulse unit 2 output with the general I/O port.
6. The count operation is started by TSTR.
7. Output goes low on compare-match occurrence.
8. An error occurs.
9. Set port output with the general I/O port and output the inverse of the active level.
10. The count operation is stopped by TSTR.
11. Not necessary when restarting in normal mode.
12. Initialize the pins with TIOR.
13. Set the multi-function timer pulse unit 2 output with the general I/O port.
14. Operation is restarted by TSTR.

## (2) Operation when Error Occurs during Normal Mode Operation, and Operation is Restarted in PWM Mode 1

Figure 10.116 shows an explanatory diagram of the case where an error occurs in normal mode and operation is restarted in PWM mode 1 after re-setting.



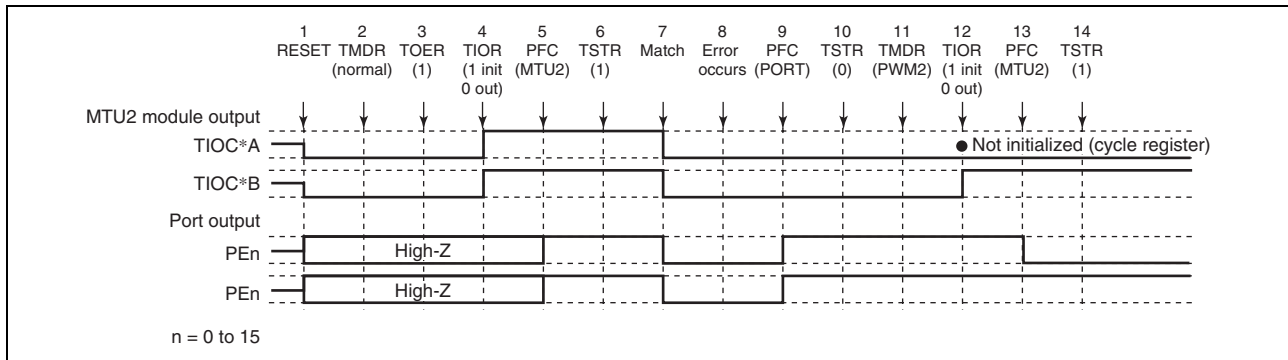
**Figure 10.116 Error Occurrence in Normal Mode, Recovery in PWM Mode 1**

1 to 10 are the same as in Figure 10.115.

11. Set PWM mode 1.
12. Initialize the pins with TIOR. (In PWM mode 1, the TIOC\*B side is not initialized. If initialization is required, initialize in normal mode, and then switch to PWM mode 1.)
13. Set the multi-function timer pulse unit 2 output with the general I/O port.
14. Operation is restarted by TSTR.

### (3) Operation when Error Occurs during Normal Mode Operation, and Operation is Restarted in PWM Mode 2

Figure 10.117 shows an explanatory diagram of the case where an error occurs in normal mode and operation is restarted in PWM mode 2 after re-setting.



**Figure 10.117 Error Occurrence in Normal Mode, Recovery in PWM Mode 2**

1 to 10 are the same as in Figure 10.115.

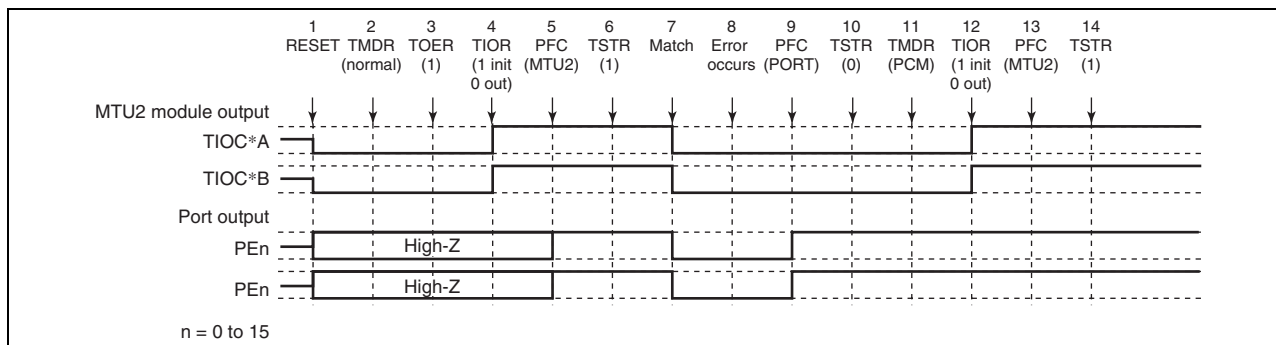
- Set PWM mode 2.
- Initialize the pins with TIOR. (In PWM mode 2, the cycle register pins are not initialized. If initialization is required, initialize in normal mode, and then switch to PWM mode 2.)
- Set the multi-function timer pulse unit 2 output with the general I/O port.
- Operation is restarted by TSTR.

Note: PWM mode 2 can only be set for channels 0 to 2, and therefore TOER setting is not necessary.



(4) Operation when Error Occurs during Normal Mode Operation, and Operation is Restarted in Phase Counting Mode

Figure 10.118 shows an explanatory diagram of the case where an error occurs in normal mode and operation is restarted in phase counting mode after re-setting.



**Figure 10.118 Error Occurrence in Normal Mode, Recovery in Phase Counting Mode**

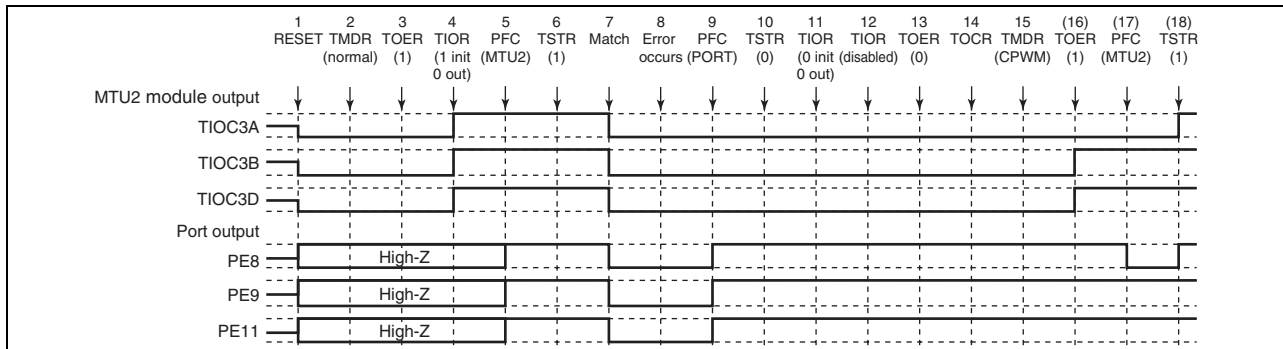
1 to 10 are the same as in Figure 10.115.

11. Set phase counting mode.
12. Initialize the pins with TIOR.
13. Set the multi-function timer pulse unit 2 output with the general I/O port.
14. Operation is restarted by TSTR.

Note: Phase counting mode can only be set for channels 1 and 2, and therefore TOER setting is not necessary.

### (5) Operation when Error Occurs during Normal Mode Operation, and Operation is Restarted in Complementary PWM Mode

Figure 10.119 shows an explanatory diagram of the case where an error occurs in normal mode and operation is restarted in complementary PWM mode after re-setting.



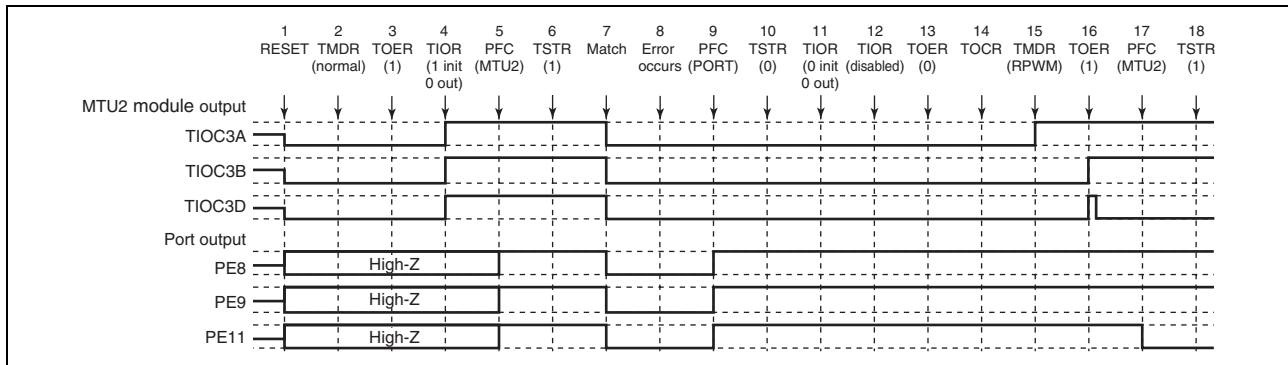
**Figure 10.119 Error Occurrence in Normal Mode, Recovery in Complementary PWM Mode**

1 to 10 are the same as in Figure 10.115.

11. Initialize the normal mode waveform generation section with TIOR.
12. Disable operation of the normal mode waveform generation section with TIOR.
13. Disable channel 3 and 4 output with TOER.
14. Select the complementary PWM output level and cyclic output enabling/disabling with TOCR.
15. Set complementary PWM.
16. Enable channel 3 and 4 output with TOER.
17. Set the multi-function timer pulse unit 2 output with the general I/O port.
18. Operation is restarted by TSTR.

### (6) Operation when Error Occurs during Normal Mode Operation, and Operation is Restarted in Reset-Synchronized PWM Mode

Figure 10.120 shows an explanatory diagram of the case where an error occurs in normal mode and operation is restarted in reset-synchronized PWM mode after re-setting.



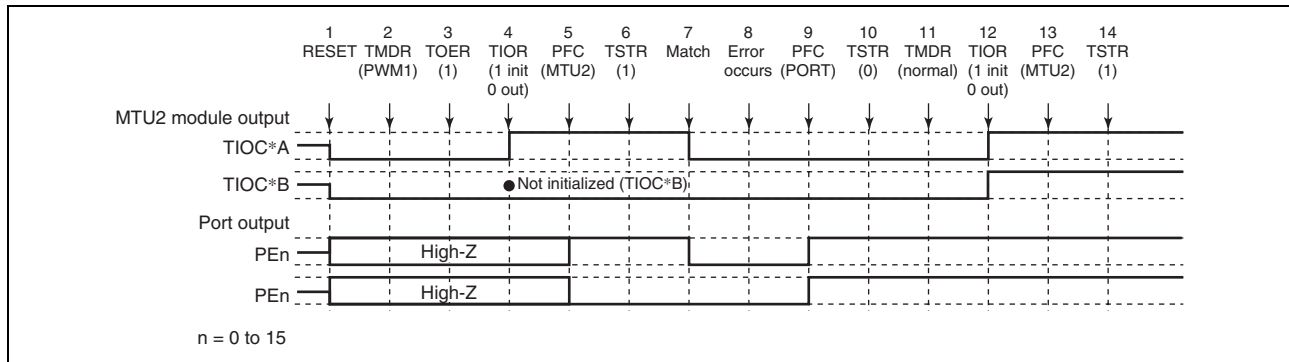
**Figure 10.120 Error Occurrence in Normal Mode, Recovery in Reset-Synchronized PWM Mode**

1 to 13 are the same as in Figure 10.115.

14. Select the reset-synchronized PWM output level and cyclic output enabling/disabling with TOCR.
15. Set reset-synchronized PWM.
16. Enable channel 3 and 4 output with TOER.
17. Set the multi-function timer pulse unit 2 output with the general I/O port.
18. Operation is restarted by TSTR.

### (7) Operation when Error Occurs during PWM Mode 1 Operation, and Operation is Restarted in Normal Mode

Figure 10.121 shows an explanatory diagram of the case where an error occurs in PWM mode 1 and operation is restarted in normal mode after re-setting.

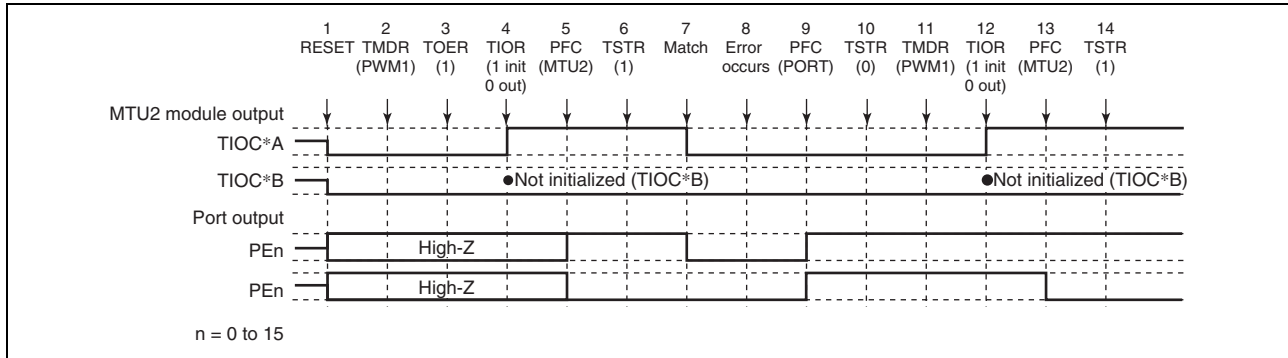


**Figure 10.121 Error Occurrence in PWM Mode 1, Recovery in Normal Mode**

1. After a reset, the module output is low and ports are in the high-impedance state.
2. Set PWM mode 1.
3. For channels 3 and 4, enable output with TOER before initializing the pins with TIOR.
4. Initialize the pins with TIOR. (The example shows initial high output, with low output on compare-match occurrence. In PWM mode 1, the TIOC\*B side is not initialized.)
5. Set the multi-function timer pulse unit 2 output with the general I/O port.
6. The count operation is started by TSTR.
7. Output goes low on compare-match occurrence.
8. An error occurs.
9. Set port output with the general I/O port and output the inverse of the active level.
10. The count operation is stopped by TSTR.
11. Set normal mode.
12. Initialize the pins with TIOR.
13. Set the multi-function timer pulse unit 2 output with the general I/O port.
14. Operation is restarted by TSTR.

### (8) Operation when Error Occurs during PWM Mode 1 Operation, and Operation is Restarted in PWM Mode 1

Figure 10.122 shows an explanatory diagram of the case where an error occurs in PWM mode 1 and operation is restarted in PWM mode 1 after re-setting.



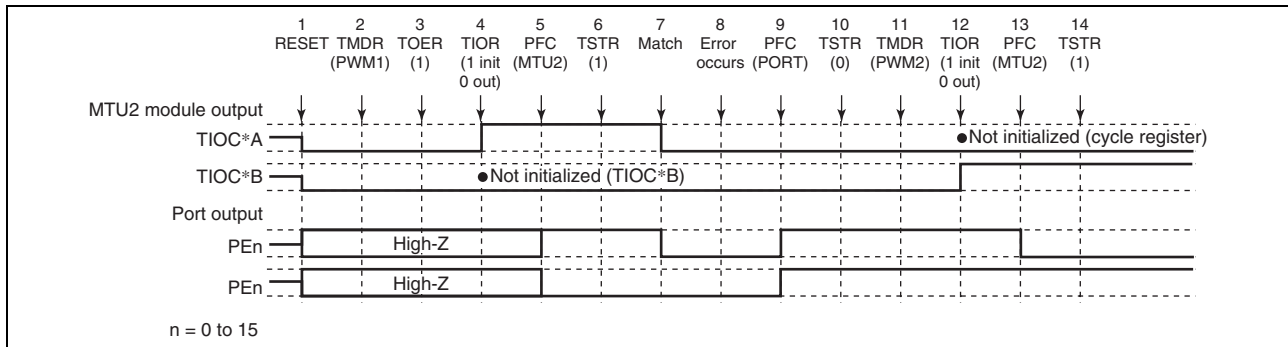
**Figure 10.122 Error Occurrence in PWM Mode 1, Recovery in PWM Mode 1**

1 to 10 are the same as in Figure 10.121.

11. Not necessary when restarting in PWM mode 1.
12. Initialize the pins with TIOR. (In PWM mode 1, the TIOC\*B side is not initialized.)
13. Set the multi-function timer pulse unit 2 output with the general I/O port.
14. Operation is restarted by TSTR.

### (9) Operation when Error Occurs during PWM Mode 1 Operation, and Operation is Restarted in PWM Mode 2

Figure 10.123 shows an explanatory diagram of the case where an error occurs in PWM mode 1 and operation is restarted in PWM mode 2 after re-setting.



**Figure 10.123 Error Occurrence in PWM Mode 1, Recovery in PWM Mode 2**

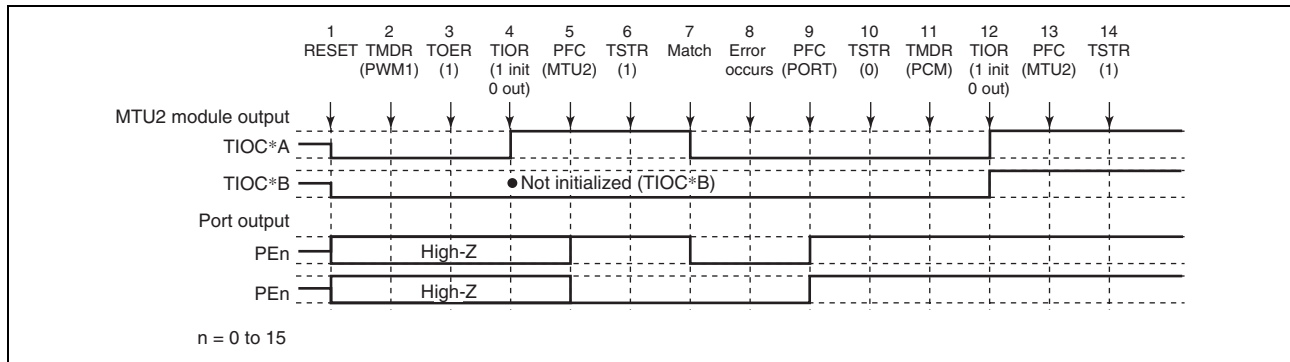
1 to 10 are the same as in Figure 10.121.

11. Set PWM mode 2.
12. Initialize the pins with TIOR. (In PWM mode 2, the cycle register pins are not initialized.)
13. Set the multi-function timer pulse unit 2 output with the general I/O port.
14. Operation is restarted by TSTR.

Note: PWM mode 2 can only be set for channels 0 to 2, and therefore TOER setting is not necessary.

### (10) Operation when Error Occurs during PWM Mode 1 Operation, and Operation is Restarted in Phase Counting Mode

Figure 10.124 shows an explanatory diagram of the case where an error occurs in PWM mode 1 and operation is restarted in phase counting mode after re-setting.



**Figure 10.124 Error Occurrence in PWM Mode 1, Recovery in Phase Counting Mode**

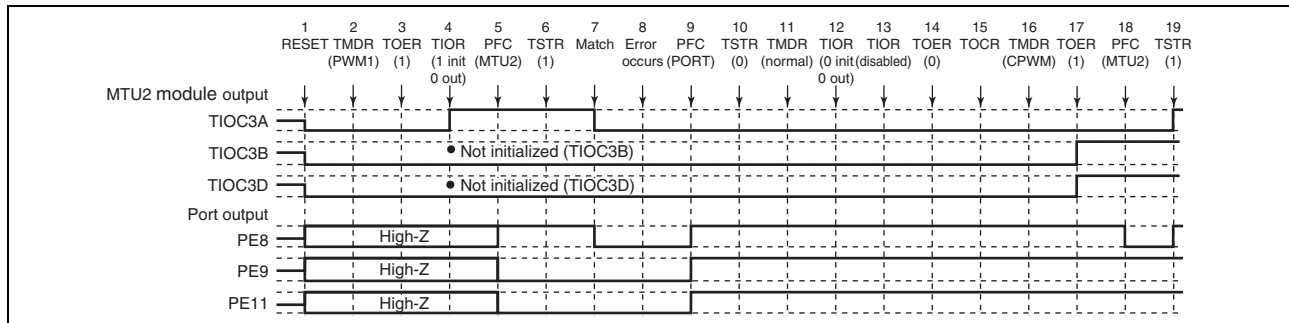
1 to 10 are the same as in Figure 10.121.

11. Set phase counting mode.
12. Initialize the pins with TIOR.
13. Set the multi-function timer pulse unit 2 output with the general I/O port.
14. Operation is restarted by TSTR.

Note: Phase counting mode can only be set for channels 1 and 2, and therefore TOER setting is not necessary.

### (11) Operation when Error Occurs during PWM Mode 1 Operation, and Operation is Restarted in Complementary PWM Mode

Figure 10.125 shows an explanatory diagram of the case where an error occurs in PWM mode 1 and operation is restarted in complementary PWM mode after re-setting.



**Figure 10.125 Error Occurrence in PWM Mode 1, Recovery in Complementary PWM Mode**

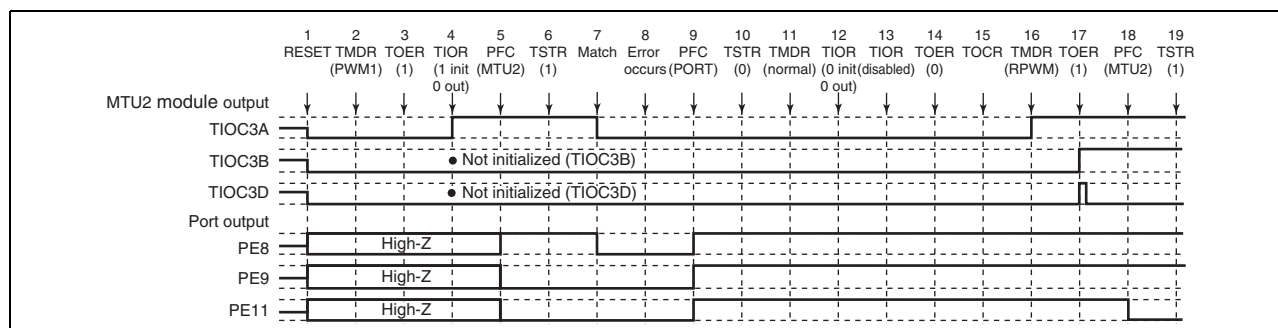
1 to 10 are the same as in Figure 10.121.

11. Set normal mode for initialization of the normal mode waveform generation section.
12. Initialize the PWM mode 1 waveform generation section with TIOR.
13. Disable operation of the PWM mode 1 waveform generation section with TIOR.
14. Disable channel 3 and 4 output with TOER.
15. Select the complementary PWM output level and cyclic output enabling/disabling with TOCR.
16. Set complementary PWM.
17. Enable channel 3 and 4 output with TOER.
18. Set the multi-function timer pulse unit 2 output with the general I/O port.
19. Operation is restarted by TSTR.



### (12) Operation when Error Occurs during PWM Mode 1 Operation, and Operation is Restarted in Reset-Synchronized PWM Mode

Figure 10.126 shows an explanatory diagram of the case where an error occurs in PWM mode 1 and operation is restarted in reset-synchronized PWM mode after re-setting.



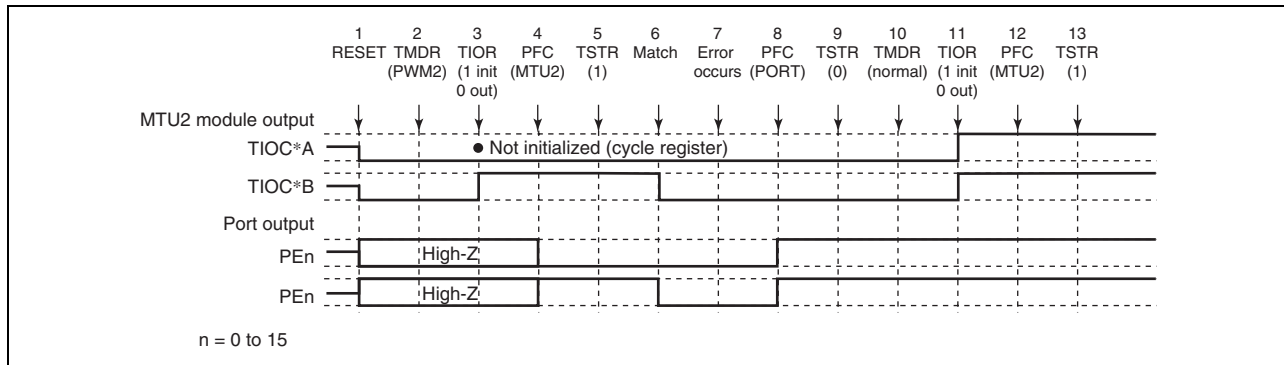
**Figure 10.126 Error Occurrence in PWM Mode 1, Recovery in Reset-Synchronized PWM Mode**

1 to 14 are the same as in Figure 10.125.

15. Select the reset-synchronized PWM output level and cyclic output enabling/disabling with TOCR.
16. Set reset-synchronized PWM.
17. Enable channel 3 and 4 output with TOER.
18. Set the multi-function timer pulse unit 2 output with the general I/O port.
19. Operation is restarted by TSTR.

### (13) Operation when Error Occurs during PWM Mode 2 Operation, and Operation is Restarted in Normal Mode

Figure 10.127 shows an explanatory diagram of the case where an error occurs in PWM mode 2 and operation is restarted in normal mode after re-setting.

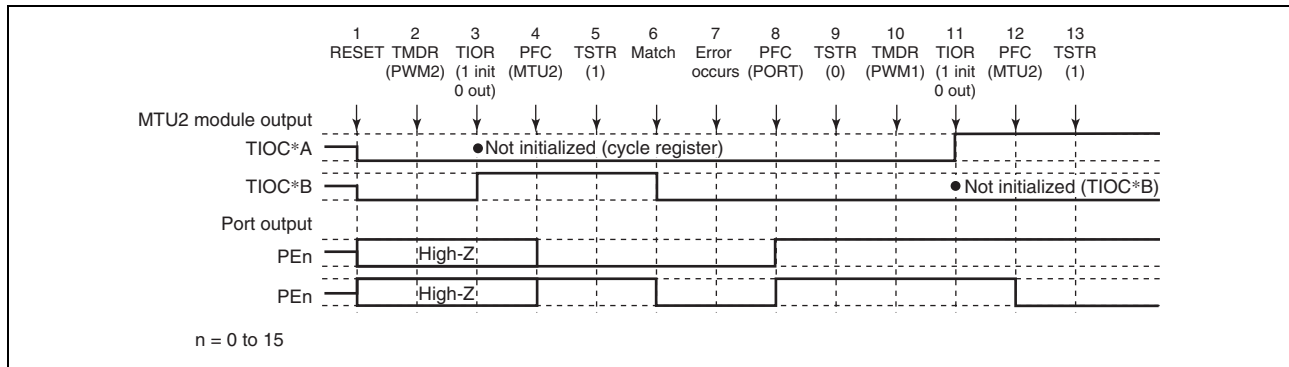


**Figure 10.127 Error Occurrence in PWM Mode 2, Recovery in Normal Mode**

1. After a reset, the module output is low and ports are in the high-impedance state.
2. Set PWM mode 2.
3. Initialize the pins with TIOR. (The example shows initial high output, with low output on compare-match occurrence. In PWM mode 2, the cycle register pins are not initialized. In the example, TIOC \*A is the cycle register.)
4. Set the multi-function timer pulse unit 2 output with the general I/O port.
5. The count operation is started by TSTR.
6. Output goes low on compare-match occurrence.
7. An error occurs.
8. Set port output with the general I/O port and output the inverse of the active level.
9. The count operation is stopped by TSTR.
10. Set normal mode.
11. Initialize the pins with TIOR.
12. Set the multi-function timer pulse unit 2 output with the general I/O port.
13. Operation is restarted by TSTR.

### (14) Operation when Error Occurs during PWM Mode 2 Operation, and Operation is Restarted in PWM Mode 1

Figure 10.128 shows an explanatory diagram of the case where an error occurs in PWM mode 2 and operation is restarted in PWM mode 1 after re-setting.



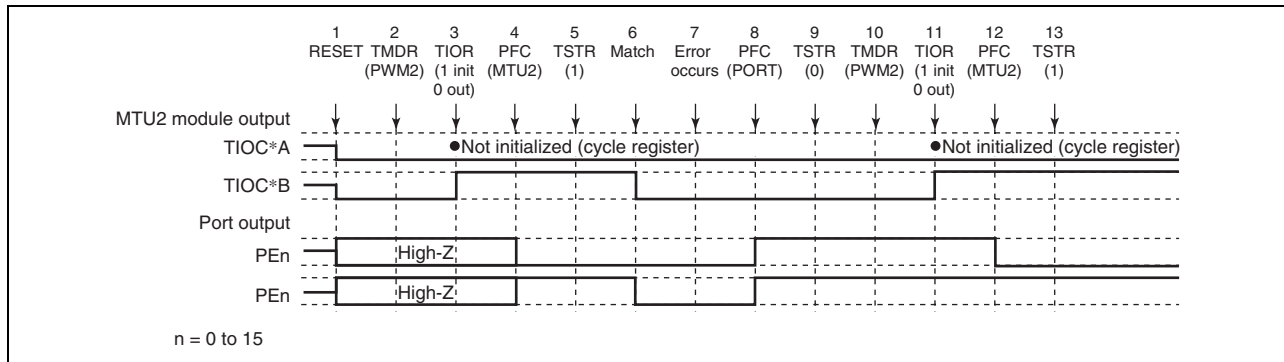
**Figure 10.128 Error Occurrence in PWM Mode 2, Recovery in PWM Mode 1**

1 to 9 are the same as in Figure 10.127.

10. Set PWM mode 1.
11. Initialize the pins with TIOR. (In PWM mode 1, the TIOC\*B side is not initialized.)
12. Set the multi-function timer pulse unit 2 output with the general I/O port.
13. Operation is restarted by TSTR.

### (15) Operation when Error Occurs during PWM Mode 2 Operation, and Operation is Restarted in PWM Mode 2

Figure 10.129 shows an explanatory diagram of the case where an error occurs in PWM mode 2 and operation is restarted in PWM mode 2 after re-setting.



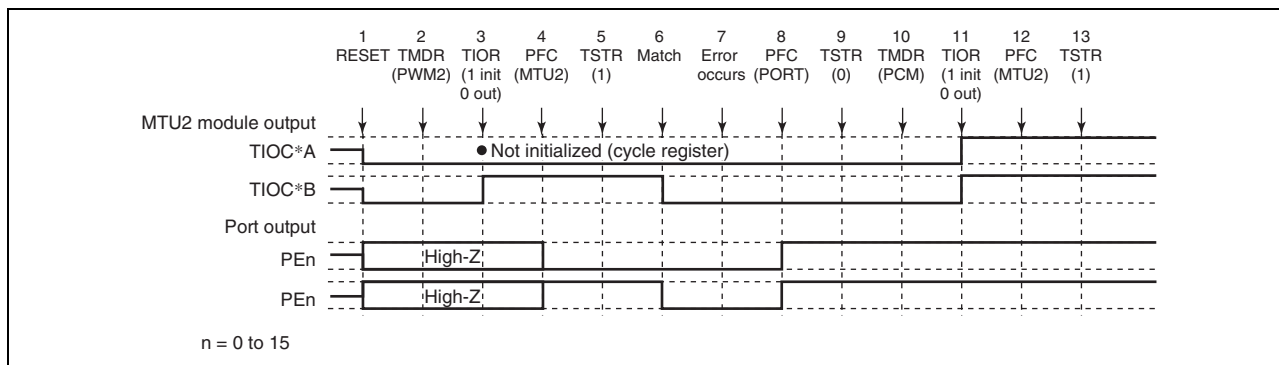
**Figure 10.129 Error Occurrence in PWM Mode 2, Recovery in PWM Mode 2**

1 to 9 are the same as in Figure 10.127.

10. Not necessary when restarting in PWM mode 2.
11. Initialize the pins with TIOR. (In PWM mode 2, the cycle register pins are not initialized.)
12. Set the multi-function timer pulse unit 2 output with the general I/O port.
13. Operation is restarted by TSTR.

(16) Operation when Error Occurs during PWM Mode 2 Operation, and Operation is Restarted in Phase Counting Mode

Figure 10.130 shows an explanatory diagram of the case where an error occurs in PWM mode 2 and operation is restarted in phase counting mode after re-setting.



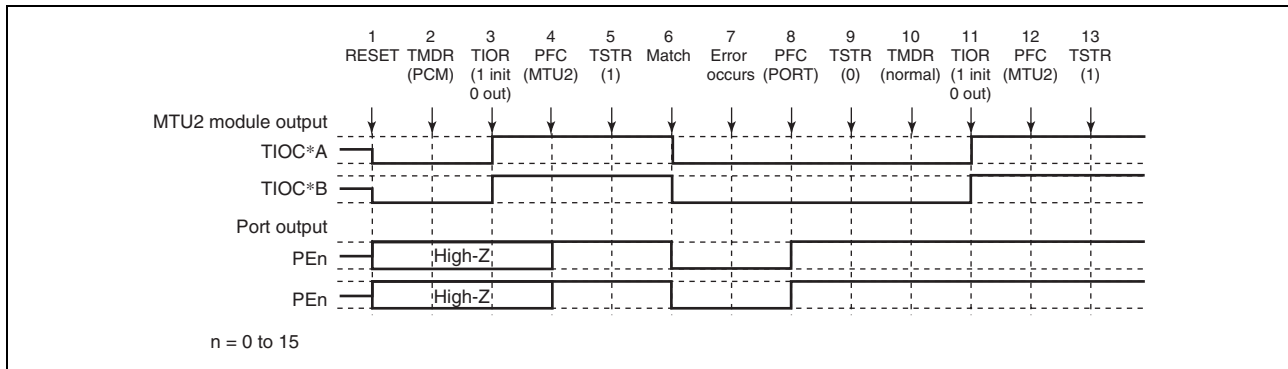
**Figure 10.130 Error Occurrence in PWM Mode 2, Recovery in Phase Counting Mode**

1 to 9 are the same as in Figure 10.127.

- 10. Set phase counting mode.
- 11. Initialize the pins with TIOR.
- 12. Set the multi-function timer pulse unit 2 output with the general I/O port.
- 13. Operation is restarted by TSTR.

### (17) Operation when Error Occurs during Phase Counting Mode Operation, and Operation is Restarted in Normal Mode

Figure 10.131 shows an explanatory diagram of the case where an error occurs in phase counting mode and operation is restarted in normal mode after re-setting.

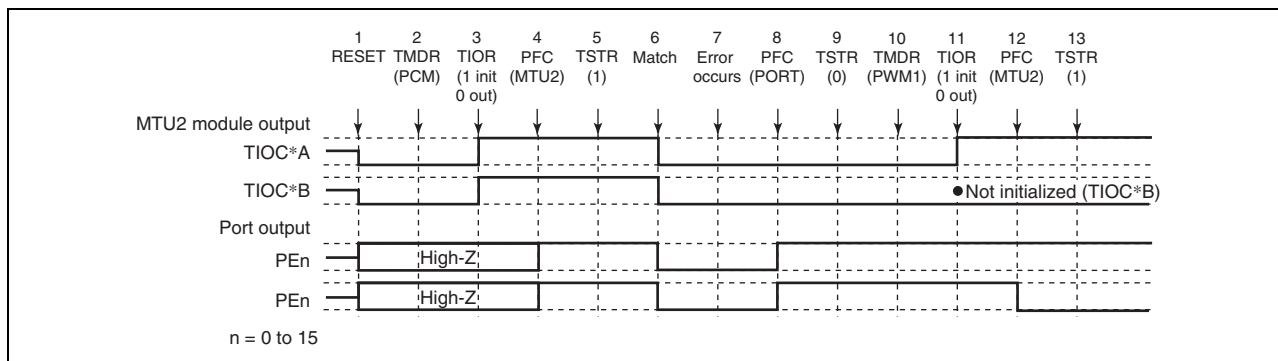


**Figure 10.131 Error Occurrence in Phase Counting Mode, Recovery in Normal Mode**

1. After a reset, the module output is low and ports are in the high-impedance state.
2. Set phase counting mode.
3. Initialize the pins with TIOR. (The example shows initial high output, with low output on compare-match occurrence.)
4. Set the multi-function timer pulse unit 2 output with the general I/O port.
5. The count operation is started by TSTR.
6. Output goes low on compare-match occurrence.
7. An error occurs.
8. Set port output with the general I/O port and output the inverse of the active level.
9. The count operation is stopped by TSTR.
10. Set in normal mode.
11. Initialize the pins with TIOR.
12. Set the multi-function timer pulse unit 2 output with the general I/O port.
13. Operation is restarted by TSTR.

(18) Operation when Error Occurs during Phase Counting Mode Operation, and Operation is Restarted in PWM Mode 1

Figure 10.132 shows an explanatory diagram of the case where an error occurs in phase counting mode and operation is restarted in PWM mode 1 after re-setting.



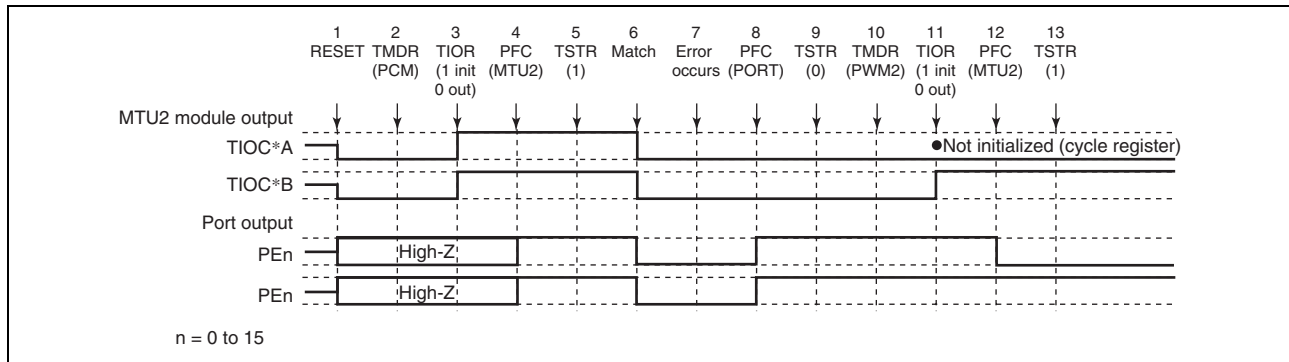
**Figure 10.132 Error Occurrence in Phase Counting Mode, Recovery in PWM Mode 1**

1 to 9 are the same as in Figure 10.131.

- 10. Set PWM mode 1.
- 11. Initialize the pins with TIOR. (In PWM mode 1, the TIOC \*B side is not initialized.)
- 12. Set the multi-function timer pulse unit 2 output with the general I/O port.
- 13. Operation is restarted by TSTR.

### (19) Operation when Error Occurs during Phase Counting Mode Operation, and Operation is Restarted in PWM Mode 2

Figure 10.133 shows an explanatory diagram of the case where an error occurs in phase counting mode and operation is restarted in PWM mode 2 after re-setting.



**Figure 10.133 Error Occurrence in Phase Counting Mode, Recovery in PWM Mode 2**

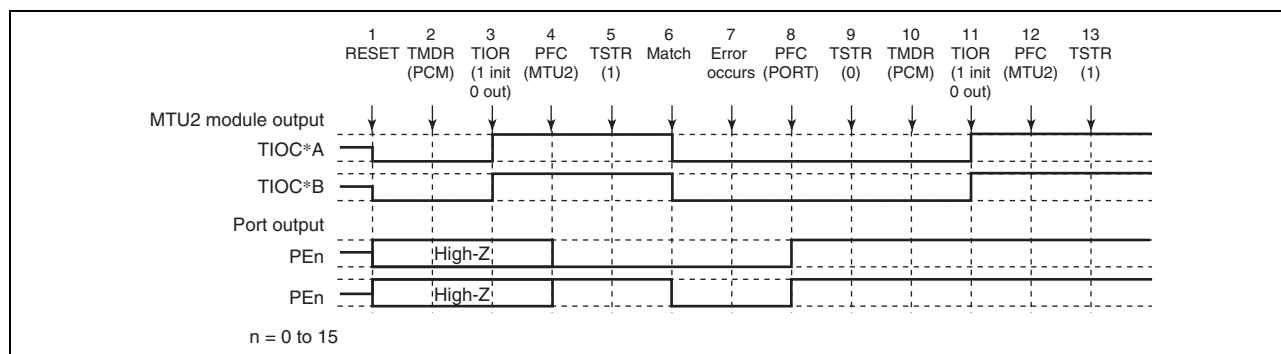
1 to 9 are the same as in Figure 10.131.

10. Set PWM mode 2.
11. Initialize the pins with TIOR. (In PWM mode 2, the cycle register pins are not initialized.)
12. Set the multi-function timer pulse unit 2 output with the general I/O port.
13. Operation is restarted by TSTR.



## (20) Operation when Error Occurs during Phase Counting Mode Operation, and Operation is Restarted in Phase Counting Mode

Figure 10.134 shows an explanatory diagram of the case where an error occurs in phase counting mode and operation is restarted in phase counting mode after re-setting.



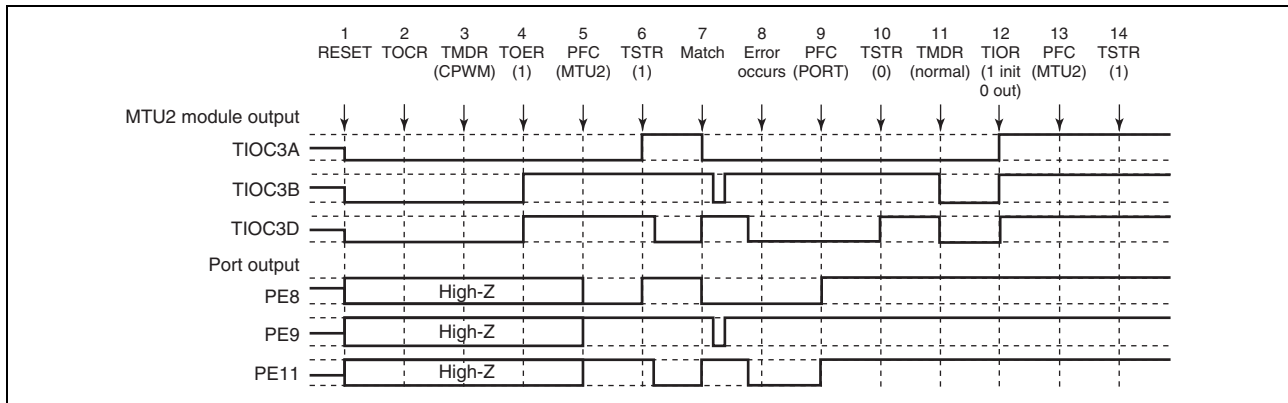
**Figure 10.134 Error Occurrence in Phase Counting Mode, Recovery in Phase Counting Mode**

1 to 9 are the same as in Figure 10.131.

- 10. Not necessary when restarting in phase counting mode.
- 11. Initialize the pins with TIOR.
- 12. Set the multi-function timer pulse unit 2 output with the general I/O port.
- 13. Operation is restarted by TSTR.

### (21) Operation when Error Occurs during Complementary PWM Mode Operation, and Operation is Restarted in Normal Mode

Figure 10.135 shows an explanatory diagram of the case where an error occurs in complementary PWM mode and operation is restarted in normal mode after re-setting.

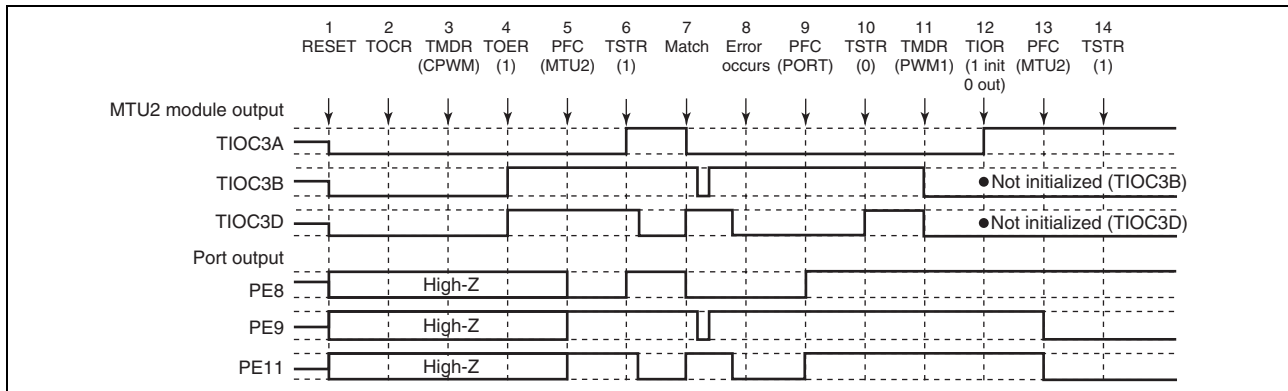


**Figure 10.135 Error Occurrence in Complementary PWM Mode, Recovery in Normal Mode**

1. After a reset, the module output is low and ports are in the high-impedance state.
2. Select the complementary PWM output level and cyclic output enabling/disabling with TOCR.
3. Set complementary PWM.
4. Enable channel 3 and 4 output with TOER.
5. Set the multi-function timer pulse unit 2 output with the general I/O port.
6. The count operation is started by TSTR.
7. The complementary PWM waveform is output on compare-match occurrence.
8. An error occurs.
9. Set port output with the general I/O port and output the inverse of the active level.
10. The count operation is stopped by TSTR. (This module outputs the same value as the complementary PWM output initial value.)
11. Set normal mode. (This module outputs a low-level signal.)
12. Initialize the pins with TIOR.
13. Set the multi-function timer pulse unit 2 output with the general I/O port.
14. Operation is restarted by TSTR.

## (22) Operation when Error Occurs during Complementary PWM Mode Operation, and Operation is Restarted in PWM Mode 1

Figure 10.136 shows an explanatory diagram of the case where an error occurs in complementary PWM mode and operation is restarted in PWM mode 1 after re-setting.



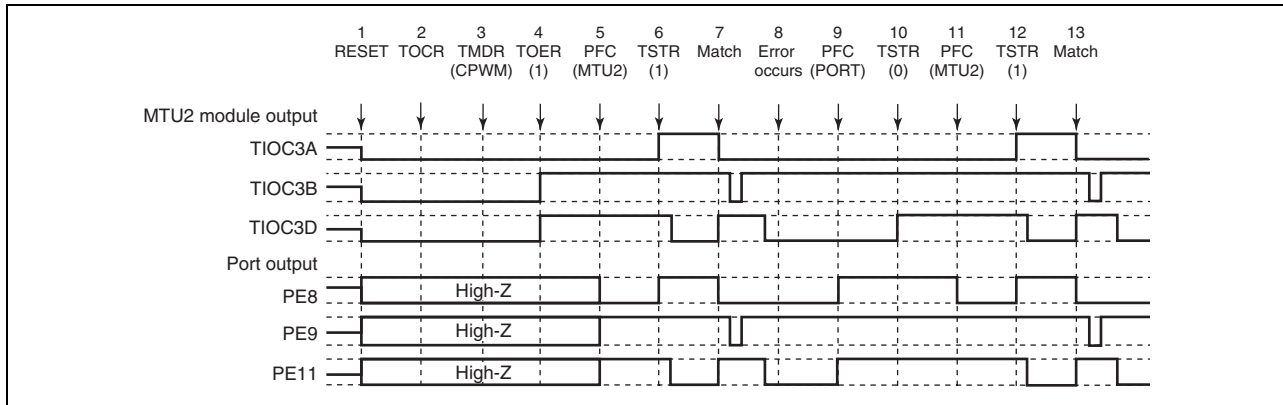
**Figure 10.136 Error Occurrence in Complementary PWM Mode, Recovery in PWM Mode 1**

1 to 10 are the same as in Figure 10.135.

11. Set PWM mode 1. (This module outputs a low-level signal.)
12. Initialize the pins with TIOR. (In PWM mode 1, the TIOC \*B side is not initialized.)
13. Set the multi-function timer pulse unit 2 output with the general I/O port.
14. Operation is restarted by TSTR.

### (23) Operation when Error Occurs during Complementary PWM Mode Operation, and Operation is Restarted in Complementary PWM Mode

Figure 10.137 shows an explanatory diagram of the case where an error occurs in complementary PWM mode and operation is restarted in complementary PWM mode after re-setting (when operation is restarted using the cycle and duty settings at the time the counter was stopped).



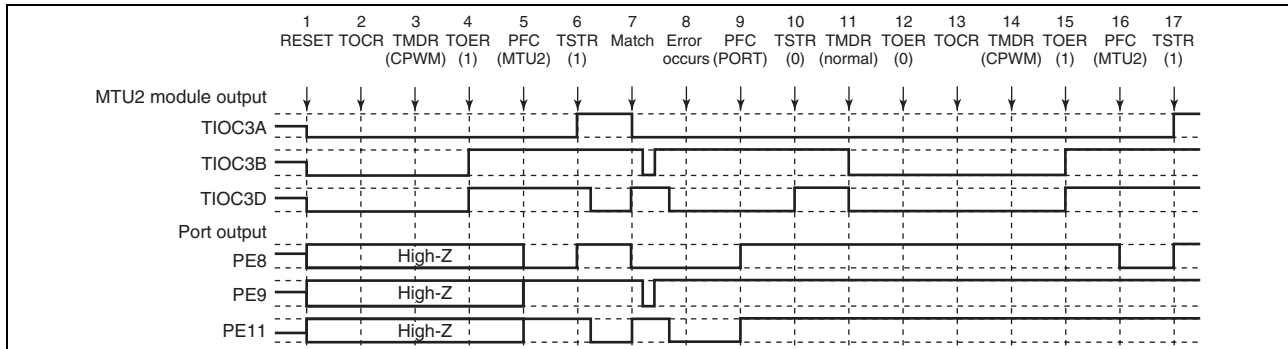
**Figure 10.137 Error Occurrence in Complementary PWM Mode, Recovery in Complementary PWM Mode**

1 to 10 are the same as in Figure 10.135.

11. Set the multi-function timer pulse unit 2 output with the general I/O port.
12. Operation is restarted by TSTR.
13. The complementary PWM waveform is output on compare-match occurrence.

### (24) Operation when Error Occurs during Complementary PWM Mode Operation, and Operation is Restarted in Complementary PWM Mode

Figure 10.138 shows an explanatory diagram of the case where an error occurs in complementary PWM mode and operation is restarted in complementary PWM mode after re-setting (when operation is restarted using completely new cycle and duty settings).



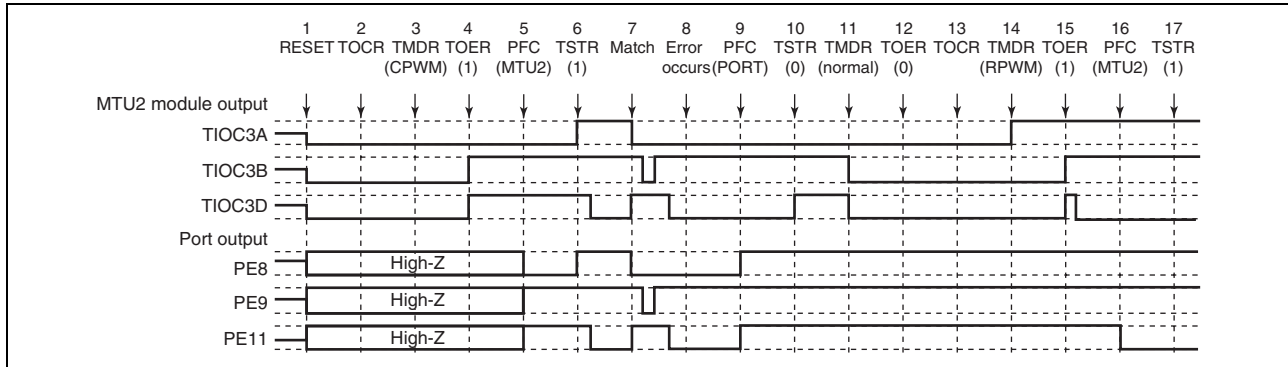
**Figure 10.138 Error Occurrence in Complementary PWM Mode, Recovery in Complementary PWM Mode**

1 to 10 are the same as in Figure 10.135.

11. Set normal mode and make new settings. (This module outputs a low-level signal.)
12. Disable channel 3 and 4 output with TOER.
13. Select the complementary PWM mode output level and cyclic output enabling/disabling with TOCR.
14. Set complementary PWM.
15. Enable channel 3 and 4 output with TOER.
16. Set the multi-function timer pulse unit 2 output with the general I/O port.
17. Operation is restarted by TSTR.

### (25) Operation when Error Occurs during Complementary PWM Mode Operation, and Operation is Restarted in Reset-Synchronized PWM Mode

Figure 10.139 shows an explanatory diagram of the case where an error occurs in complementary PWM mode and operation is restarted in reset-synchronized PWM mode.



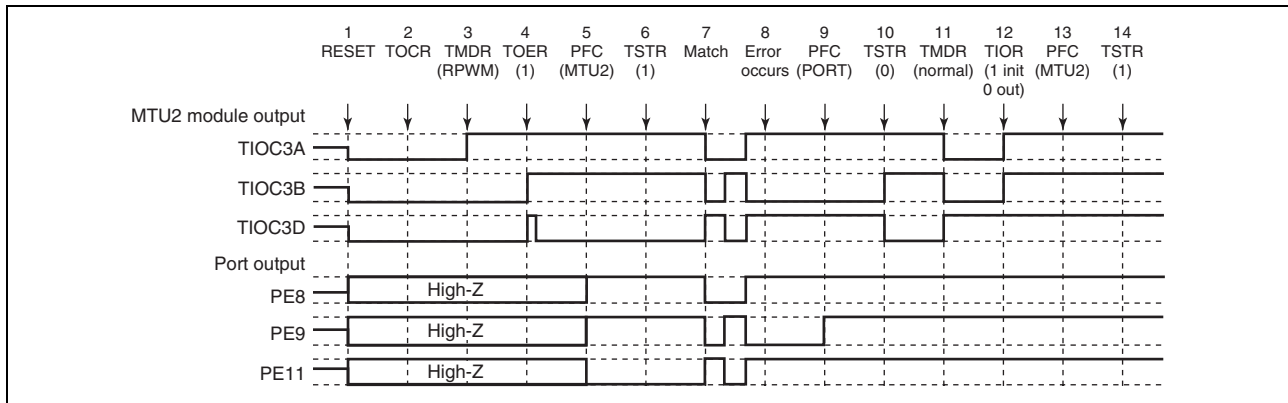
**Figure 10.139 Error Occurrence in Complementary PWM Mode, Recovery in Reset-Synchronized PWM Mode**

1 to 10 are the same as in Figure 10.135.

11. Set normal mode. (This module outputs a low-level signal.)
12. Disable channel 3 and 4 output with TOER.
13. Select the reset-synchronized PWM mode output level and cyclic output enabling/disabling with TOCR.
14. Set reset-synchronized PWM.
15. Enable channel 3 and 4 output with TOER.
16. Set the multi-function timer pulse unit 2 output with the general I/O port.
17. Operation is restarted by TSTR.

## (26) Operation when Error Occurs during Reset-Synchronized PWM Mode Operation, and Operation is Restarted in Normal Mode

Figure 10.140 shows an explanatory diagram of the case where an error occurs in reset-synchronized PWM mode and operation is restarted in normal mode after re-setting.

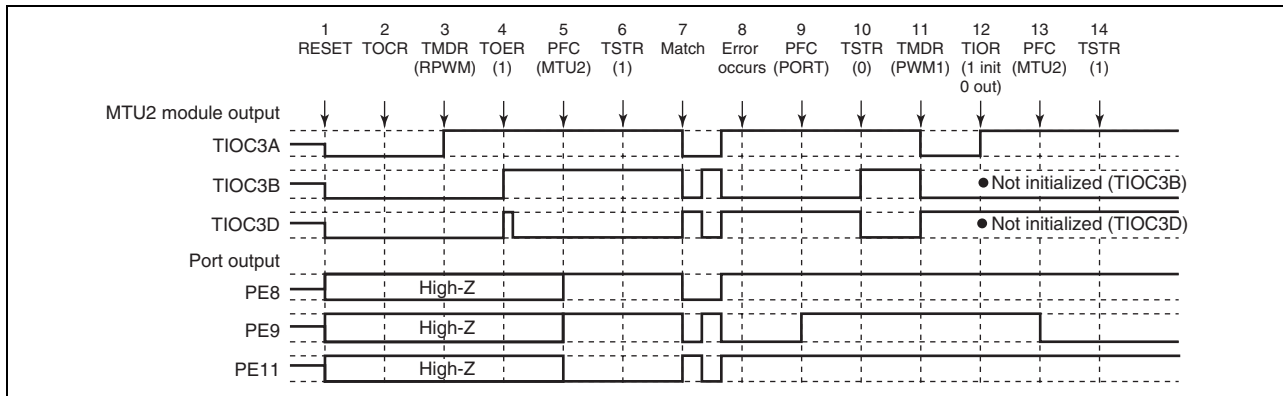


**Figure 10.140 Error Occurrence in Reset-Synchronized PWM Mode, Recovery in Normal Mode**

1. After a reset, the module output is low and ports are in the high-impedance state.
2. Select the reset-synchronized PWM output level and cyclic output enabling/disabling with TOCR.
3. Set reset-synchronized PWM.
4. Enable channel 3 and 4 output with TOER.
5. Set the multi-function timer pulse unit 2 output with the general I/O port.
6. The count operation is started by TSTR.
7. The reset-synchronized PWM waveform is output on compare-match occurrence.
8. An error occurs.
9. Set port output with the general I/O port and output the inverse of the active level.
10. The count operation is stopped by TSTR. (This module outputs the same value as the reset-synchronized PWM output initial value.)
11. Set normal mode. (The positive phase output from this module is low, and negative phase output is high.)
12. Initialize the pins with TIOR.
13. Set the multi-function timer pulse unit 2 output with the general I/O port.
14. Operation is restarted by TSTR.

### (27) Operation when Error Occurs during Reset-Synchronized PWM Mode Operation, and Operation is Restarted in PWM Mode 1

Figure 10.141 shows an explanatory diagram of the case where an error occurs in reset-synchronized PWM mode and operation is restarted in PWM mode 1 after re-setting.



**Figure 10.141 Error Occurrence in Reset-Synchronized PWM Mode, Recovery in PWM Mode 1**

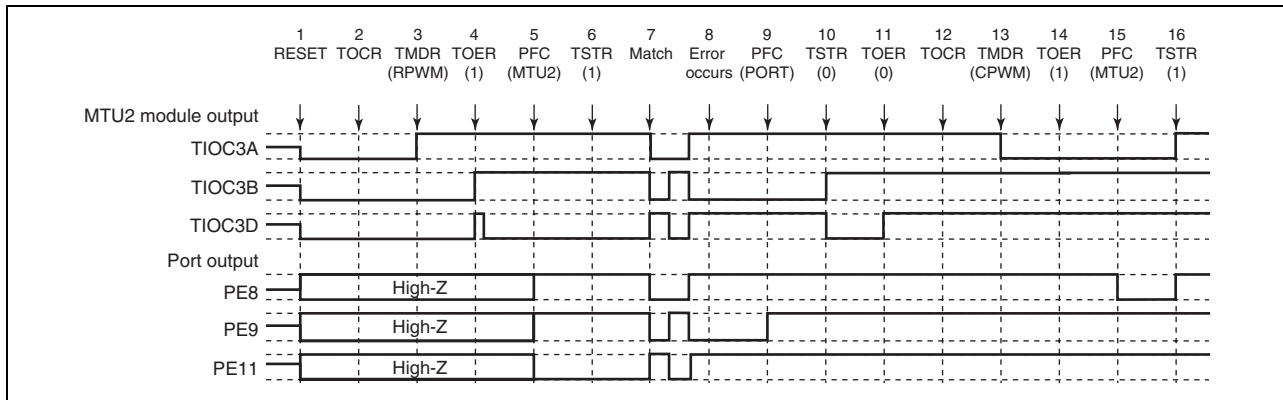
1 to 10 are the same as in Figure 10.140.

11. Set PWM mode 1. (The positive phase output from this module is low, and negative phase output is high.)
12. Initialize the pins with TIOR. (In PWM mode 1, the TIOC \*B side is not initialized.)
13. Set the multi-function timer pulse unit 2 output with the general I/O port.
14. Operation is restarted by TSTR.



### (28) Operation when Error Occurs during Reset-Synchronized PWM Mode Operation, and Operation is Restarted in Complementary PWM Mode

Figure 10.142 shows an explanatory diagram of the case where an error occurs in reset-synchronized PWM mode and operation is restarted in complementary PWM mode after re-setting.



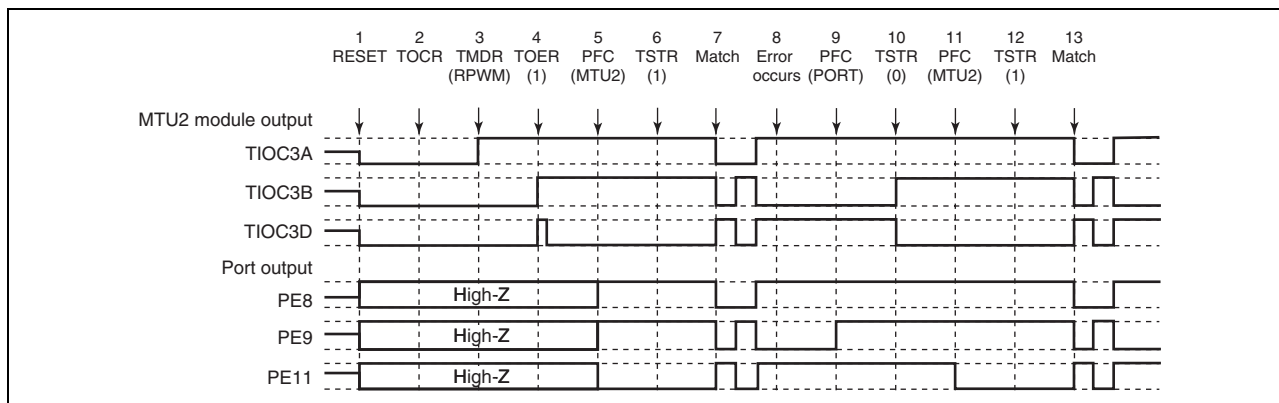
**Figure 10.142 Error Occurrence in Reset-Synchronized PWM Mode, Recovery in Complementary PWM Mode**

1 to 10 are the same as in Figure 10.140.

11. Disable channel 3 and 4 output with TOER.
12. Select the complementary PWM output level and cyclic output enabling/disabling with TOCR.
13. Set complementary PWM. (The cyclic output pin of this module outputs a low-level signal.)
14. Enable channel 3 and 4 output with TOER.
15. Set the multi-function timer pulse unit 2 output with the general I/O port.
16. Operation is restarted by TSTR.

### (29) Operation when Error Occurs during Reset-Synchronized PWM Mode Operation, and Operation is Restarted in Reset-Synchronized PWM Mode

Figure 10.143 shows an explanatory diagram of the case where an error occurs in reset-synchronized PWM mode and operation is restarted in reset-synchronized PWM mode after re-setting.



**Figure 10.143 Error Occurrence in Reset-Synchronized PWM Mode, Recovery in Reset-Synchronized PWM Mode**

1 to 10 are the same as in Figure 10.140.

11. Set the multi-function timer pulse unit 2 output with the general I/O port.
12. Operation is restarted by TSTR.
13. The reset-synchronized PWM waveform is output on compare-match occurrence.

## 11. OS Timer

### 11.1 Functional Overview

The OS timer has the following features.

- Two operating modes
  - Interval timer mode
  - Free-running comparison mode
- Choice between startup of DMA by compare match and generation of interrupt

#### 11.1.1 Features of OSTM

##### Channels

This product has the following number of channels of the OS timer.

**Table 11.58 Channels of OS timer**

OS Timer	
Number of Channels	2
Name	OSTMn

##### Meaning of n

Throughout this section, the individual channels of the OS timer are identified by the index "n" (n = 0, 1), for example OSTMnTO for the OS timer n output register.

##### Register address

All OS timer register addresses are given as address offsets from the individual base addresses <OSTMn\_base>.

The register base addresses of each OSTMn are listed in the following table.

**Table 11.59 Register Base Addresses**

OSTMn	<OSTMn_base> Address
OSTM0	FCFEC000
OSTM1	FCFEC400

##### Interrupts

The OS timers can generate the following interrupt requests.

**Table 11.60 OSTMn Interrupt Requests**

OSTMn Signal	Function	Startup of Direct Memory Access Controller
OSTM0TINT	OSTM0 interrupt	√
OSTM1TINT	OSTM1 interrupt	√

## 11.2 Registers

The OS timer is controlled and operated by the following registers.

### 11.2.1 Registers Overview

The list of OSTMn (n = 0, 1) registers and the memory addresses are as follows.

For the base addresses, see the **Table 11.59**.

For the actual addresses, the offset values indicated in the following table are added to the base addresses.

Register Name	Function	R/W	Reset Value	Access Unit (bit)			Offset Address (PADDR5-PADDR1)
				8	16	32	
OSTMnCMP	OSTM compare register	R/W	00000000H	—	—	√	00H
OSTMnCNT	OSTM counter register	R	FFFFFFFFH	—	—	√	04H
OSTMnTE	OSTM count enable status register	R	00H	√	—	—	10H
OSTMnTS	OSTM count start trigger register	W	00H	√	—	—	14H
OSTMnTT	OSTM count stop trigger register	W	00H	√	—	—	18H
OSTMnCTL	OSTM control register	R/W	00H	√	—	—	20H

## 11.2.2 Details of OSTM Registers

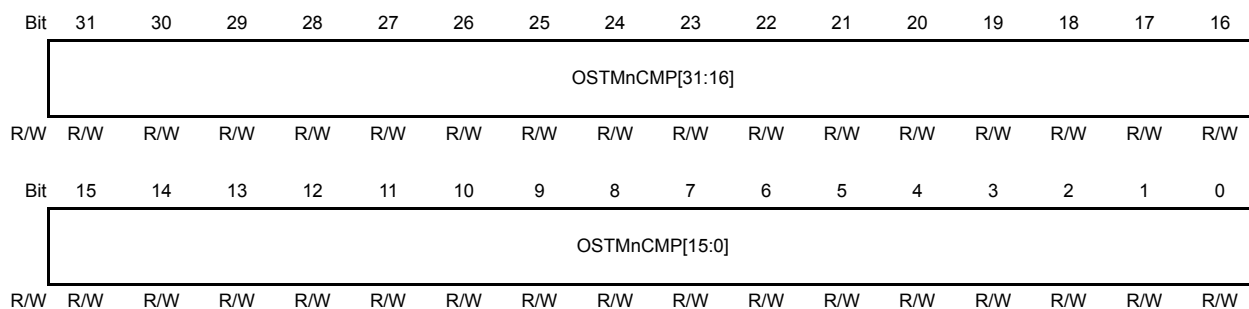
### 11.2.2.1 OSTMnCMP - OSTM Compare Register

Depending on the mode of operation, this register holds the start value for the down-counter or the value for comparison with that of the counter.

**Access:** This register is readable/writable in 32-bit units.

**Address:** <OSTMn\_base>

**Initial value:** 0000 0000<sub>H</sub>



**Table 11.61 OSTMnCMP register contents**

Bit Position	Bit Name	Function
31 to 0	OSTMnCMP [31:0]	<ul style="list-style-type: none"> <li>In interval timer mode: start value of the down-counter</li> <li>In free-running comparison mode: value for comparison</li> </ul>

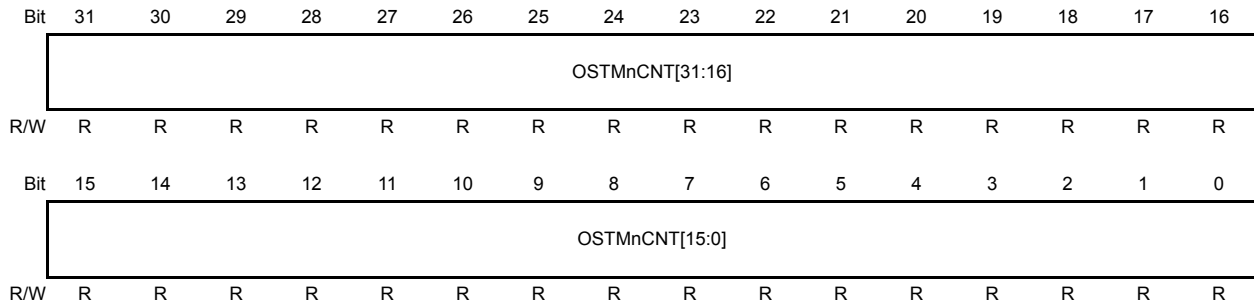
### 11.2.2.2 OSTMnCNT - OSTM Counter Register

This register indicates the counter value of the timer.

**Access:** This register is readable in 32-bit units.

**Address:** OSTMn\_base> + 4<sub>H</sub>

**Initial value:** The initial value depends on the operating mode of the OS timer. Refer to **Table 11.63, Correspondence between Operating Mode, Counting Direction and Initial Value.**



**Table 11.62 OSTMnCNT register contents**

Bit Position	Bit Name	Function
31 to 0	OSTMnCNT [31:0]	32-bit counter value

The following table shows the correspondence between operating mode, counting direction and initial value. The initial value is the value read from the counter after a change to the operating mode.

**Table 11.63 Correspondence between Operating Mode, Counting Direction and Initial Value**

Timer Operating Mode	OSTMnCTL.OSTMnMD1	Counting Direction	Initial value
Interval timer mode	0 <sup>*1</sup>	Down	FFFF FFFF <sub>H</sub>
Free-running comparison mode	1	Up	0000 0000 <sub>H</sub>

Note 1. Value after reset

### 11.2.2.3 OSTMnTE - OSTM Count Enable Status Register

This register indicates whether the counter is enabled or disabled.

**Access:** This register is readable in 8-bit units.

**Address:** <OSTMn\_base>+ 10<sub>H</sub>

**Initial value:** 00<sub>H</sub>

Bit	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	OSTMnTE
R/W	R	R	R	R	R	R	R	R

**Table 11.64 OSTMnTE register contents**

Bit Position	Bit Name	Function
7 to 1	Reserved	Reserved These bits are always read as 0. The write value should always be 0.
0	OSTMnTE	This bit indicates whether the counter is enabled or disabled. 0: Counter disabled 1: Counter enabled This bit is set to 1 in response to OSTMnTS.OSTMnTS being set to 1. This bit is reset to 0 in response to OSTMnTT.OSTMnTT being set to 1.

#### NOTE

When OSTMnTE = 0, the counter retains its value.

If the counter is restarted, it

- restarts counting down from the value in the OSTMnCMP register if it is in interval timer mode or
- restarts counting up from the counter value 0000 0000H if it is in free running comparison mode.

### 11.2.2.4 OSTMnTS - OSTM Count Start Trigger Register

This register starts the counter.

**Access:** This register is writable in 8-bit units. It is always read as 00<sub>H</sub>.

**Address:** <OSTMn\_base>+ 14<sub>H</sub>

**Initial value:** 00<sub>H</sub>

Bit	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	OSTMnTS
R/W	R	R	R	R	R	R	R	W

**Table 11.65 OSTMnTS register contents**

Bit Position	Bit Name	Function
7 to 1	Reserved	Reserved These bits are always read as 0. The write value should always be 0.
0	OSTMnTS	This bit starts the counter. 0: No function 1: Starts the counter and sets OSTMnTE.OSTMnTE = 1. <ul style="list-style-type: none"> <li>• In interval timer mode, a forced restart is executed if this bit is set while OSTMnTE.OSTMnTE = 1.</li> <li>• In free-running comparison mode, setting this bit is ignored as long as OSTMnTE.OSTMnTE = 1.</li> </ul>

### 11.2.2.5 OSTMnTT- OSTM Count Stop Trigger Register

This register stops the counter.

**Access:** This register is writable in 8-bit units. It is always read as 00<sub>H</sub>.

**Address:** <OSTMn\_base>+ 18<sub>H</sub>

**Initial value:** 00<sub>H</sub>

Bit	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	0	OSTMnTT
R/W	R	R	R	R	R	R	R	W

**Table 11.66** OSTMnTT register contents

Bit Position	Bit Name	Function
7 to 1	Reserved	Reserved These bits are always read as 0. The write value should always be 0.
0	OSTMnTT	Stops the counter. 0: No function 1: Stops the counter and clears the OSTMnTE.OSTMnTE bit.

### 11.2.2.6 OSTMnCTL - OSTM Control Register

This register specifies the operating mode for the counter and controls enabling/disabling of OSTMnTINT interrupt requests when counting starts.

**Access:** This register is readable/writable in 8-bit units. Writing to this register is only possible if the counter is disabled (OSTMnTOE.OSTMnTOE = 0).

**Address:** <OSTMn\_base>+ 20<sub>H</sub>

**Initial value:** 00<sub>H</sub>

Bit	7	6	5	4	3	2	1	0
	0	0	0	0	0	0	OSTMnMD1	OSTMnMD0
R/W	R	R	R	R	R	R	R/W	R/W

**Table 11.67** OSTMnCTL register contents

Bit Position	Bit Name	Function
7 to 2	Reserved	Reserved These bits are always read as 0. The write value should always be 0.
1	OSTMnMD1	Specifies the operating mode for the counter. 0: Interval timer mode 1: Free-running comparison mode
0	OSTMnMD0	Controls enabling/disabling of OSTMnTINT interrupt requests when counting starts. 0: Disables the interrupts when counting starts. 1: Enables the interrupts when counting starts.



## 11.3 Functional Description

Each OS timer is a 32-bit timer/counter.

The settings for operating mode specify the direction of counting (up or down) and the generation of interrupt requests.

### 11.3.1 Count Clock

The count clock of OSTMn is P0φ.

### 11.3.2 Generation of Interrupt Request

An OSTMnTINT interrupt request is generated whenever the counter reaches 0000 0000H (in interval timer mode) or matches the comparison value (in free-running comparison mode).

An interrupt request can also be generated on starting and restarting of the counter. This is controlled by the OSTMnCTL.OSTMnMD0 bit.

### 11.3.3 Starting and Stopping the Timer

The OS timer is started and stopped as follows.

#### Starting the timer

The timer is started in either of the following way:

- setting the OSTMnTS.OSTMnTSF bit to 1

Status bit OSTMnTE.OSTMnTE is set to 1.

The counter starts to count up or down in accord with the settings for operating mode.

#### Stopping the timer

Setting the OSTMnTT.OSTMnTT bit to 1 stops the timer.

This also clears the OSTMnTE.OSTMnTE status flag.

#### Initialization

To initialize interval timer mode and free-running comparison mode after reset release, take the following steps.

1. Set the value where the down-counter is to start counting or the value for comparison in the OSTMnCMP register.
2. Set the OSTMnCTL.OSTMnMD1 bit to select interval timer mode or free running comparison mode.
3. Select the interrupt mode for the start of counting (OSTMnCTL.OSTMnMD0 = 1 or 0).

### 11.3.4 Interval Timer Mode

Select the interval timer mode when an OS timer is to be used as a reference timer for generating interrupt requests at a fixed interval.

#### 11.3.4.1 Basic Operation in Interval Timer Mode

In interval timer mode, the timer counts down from the value specified in the OSTMnCMP register. An OSTMnTINT interrupt request is generated when the counter reaches 0000 0000H.

Select interval timer mode by setting OSTMnCTL.OSTMnMD1 = 0.

New values can be written to the OSTMnCMP register at any time. If it is rewritten during count operation, the counter loads the new OSTMnCMP value when the next 0000 0000H is reached.

#### Cycles of OSTMnTINT output

The cycle of OSTMnTINT output is as follows.

- OSTMnTINT generation cycle = counter-clock cycle × (OSTMnCMP + 1)

#### Forced restart

The counter is forcibly restarted by setting OSTMnTS.OSTMnTS = 1 during counting.

The counter loads the initial value from the OSTMnCMP register and continues counting down.

#### 11.3.4.2 Operation when OSTMnCMP = 0000 0000H

When OSTMnCMP = 0000 0000H, the OS timer behaves as follows.

- The OSTMnTINT interrupt request signal stays fixed to the high level from the start to the end of counting.

### 11.3.5 Free-Running Comparison Mode

#### 11.3.5.1 Basic Operation in Free-Running Comparison Mode

In free-running comparison mode, the counter counts up from 0000 0000H to FFFF FFFFH. An OSTMnTINT interrupt request is output when the current value of the counter matches the value of the OSTMnCMP register. The free-running comparison mode is selected by setting the OSTMnCTL.OSTMnMD1 bit to 1.

New values can be written to the OSTMnCMP register at any time.

#### Forced restart

Forced restarting does not proceed during counting even if the OSTMnTS.OSTMnTS bit is set. The counter ignores the attempted setting and continues counting.

## 12. Watchdog Timer

This LSI includes the watchdog timer, which externally outputs an overflow signal ( $\overline{\text{WDTOVF}}$ ) on overflow of the counter when the value of the counter has not been updated because of a system malfunction. This module can simultaneously generate an internal reset signal for the entire LSI.

This module is a single channel timer that counts up the clock oscillation settling period when the system leaves software standby mode. It can also be used as a general watchdog timer or interval timer.

### 12.1 Features

- Can be used to ensure the clock oscillation settling time  
This module is used in leaving software standby mode.
- Can switch between watchdog timer mode and interval timer mode.
- Outputs  $\overline{\text{WDTOVF}}$  signal in watchdog timer mode  
When the counter overflows in watchdog timer mode, the  $\overline{\text{WDTOVF}}$  signal is output externally. It is possible to select whether to reset the LSI internally when this happens. The power-on reset signal is used as the internal reset signal.
- Interrupt generation in interval timer mode  
An interval timer interrupt is generated when the counter overflows.
- Choice of eight counter input clocks  
Eight clocks ( $\text{P0}\phi \times 1$  to  $\text{P0}\phi \times 1/16384$ ) that are obtained by dividing the peripheral clock can be selected.

Figure 12.1 shows a block diagram.

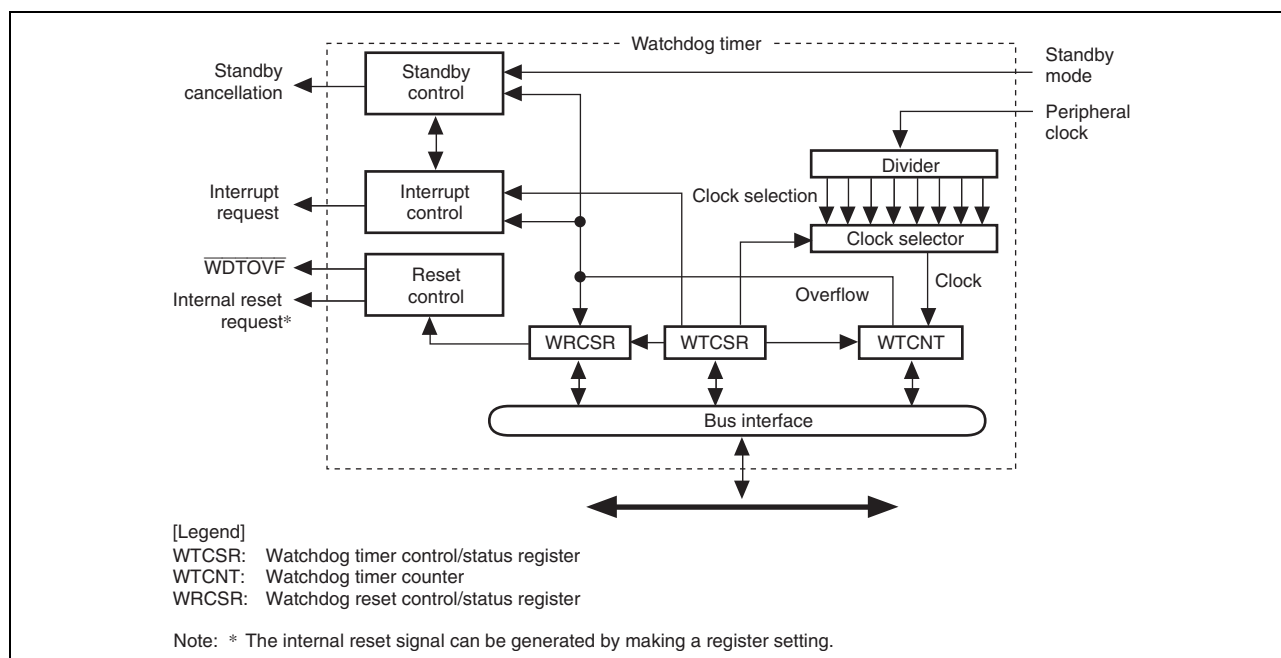


Figure 12.1 Block Diagram

## 12.2 Input/Output Pin

Table 12.1 shows the pin configuration.

**Table 12.1 Pin Configuration**

Pin Name	Symbol	I/O	Function
Watchdog timer overflow	$\overline{\text{WDTOVF}}$	Output	Outputs the counter overflow signal in watchdog timer mode

## 12.3 Register Descriptions

Table 12.2 shows the register configuration.

**Table 12.2 Register Configuration**

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Watchdog timer counter	WTCNT	R/W	H'00	H'FCFE0002	16*
Watchdog timer control/status register	WTCSR	R/W	H'18	H'FCFE0000	16*
Watchdog reset control/status register	WRCSR	R/W	H'1F	H'FCFE0004	16*

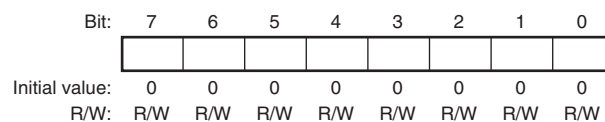
Note: \* For the access size, see section 12.3.4, Notes on Register Access.

### 12.3.1 Watchdog Timer Counter (WTCNT)

WTCNT is an 8-bit readable/writable register that is incremented by cycles of the selected clock signal. When an overflow occurs, it generates a watchdog timer overflow signal ( $\overline{\text{WDTOVF}}$ ) in watchdog timer mode and an interrupt in interval timer mode.

Use 16-bit access to write to WTCNT, writing H'5A in the upper byte. Use 8-bit access to read from WTCNT.

**Note:** The method for writing to WTCNT differs from that for other registers to prevent erroneous writes. See section 12.3.4, Notes on Register Access for details.



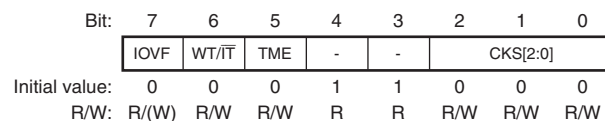
### 12.3.2 Watchdog Timer Control/Status Register (WTCSR)

WTCSR is an 8-bit readable/writable register composed of bits to select the clock used for the count, overflow flags, and timer enable bit.

When used to count the clock oscillation settling time for canceling software standby mode, it retains its value after counter overflow.

Use 16-bit access to write to WTCSR, writing H'A5 in the upper byte. Use 8-bit access to read from WTCSR.

**Note:** The method for writing to WTCSR differs from that for other registers to prevent erroneous writes. See section 12.3.4, Notes on Register Access for details.



Bit	Bit Name	Initial Value	R/W	Description
7	IOVF	0	R/(W)	Interval Timer Overflow Indicates that WTCNT has overflowed in interval timer mode. This flag is not set in watchdog timer mode. 0: No overflow 1: WTCNT overflow in interval timer mode [Clearing condition] • When 0 is written to IOVF after reading IOVF
6	WT/IT	0	R/W	Timer Mode Select Selects whether to use this module as a watchdog timer or an interval timer. 0: Use as interval timer 1: Use as watchdog timer <b>Note:</b> When the WTCNT overflows in watchdog timer mode, the $\overline{\text{WDTOVF}}$ signal is output externally. If this bit is modified when this module is running, the up-count may not be performed correctly.

Bit	Bit Name	Initial Value	R/W	Description
5	TME	0	R/W	<p>Timer Enable</p> <p>Starts and stops timer operation. Clear this bit to 0 when using this module in software standby mode or when changing the clock frequency.</p> <p>0: Timer disabled            Count-up stops and WTCNT value is retained            1: Timer enabled</p>
4,3	—	All 1	R	<p>Reserved</p> <p>These bits are always read as 1. The write value should always be 1.</p>
2 to 0	CKS[2:0]	000	R/W	<p>Clock Select</p> <p>These bits select the clock to be used for the WTCNT count from the eight types obtainable by dividing the peripheral clock (<math>P0\phi</math>). The overflow period that is shown inside the parenthesis in the table is the value when the peripheral clock (<math>P0\phi</math>) is 33.33 MHz.</p>

Bits 2 to 0	Clock Ratio	Overflow Cycle
000:	1 x $P0f$	7.7 ms
001:	1/64 x $P0f$	490 ms
010:	1/128 x $P0f$	979 ms
011:	1/256 x $P0f$	2.0 ms
100:	1/512 x $P0f$	3.9 ms
101:	1/1024 x $P0f$	7.8 ms
110:	1/4096 x $P0f$	31 ms
111:	1/16384 x $P0f$	125 ms

Note: If bits CKS[2:0] are modified when this module is running, the up-count may not be performed correctly. Ensure that these bits are modified only when this module is not running.

### 12.3.3 Watchdog Reset Control/Status Register (WRCSR)

WRCSR is an 8-bit readable/writable register that controls output of the internal reset signal generated by watchdog timer counter (WTCNT) overflow.

Note: The method for writing to WRCSR differs from that for other registers to prevent erroneous writes. See section 12.3.4, Notes on Register Access for details.

Bit:	7	6	5	4	3	2	1	0
	WOVF	RSTE	RSTS	-	-	-	-	-
Initial value:	0	0	0	1	1	1	1	1
R/W:	R/(W)	R/W	R/W	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	WOVF	0	R/(W)	Watchdog Timer Overflow Indicates that the WTCNT has overflowed in watchdog timer mode. This bit is not set in interval timer mode. 0: No overflow 1: WTCNT has overflowed in watchdog timer mode [Clearing condition] • When 0 is written to WOVF after reading WOVF
6	RSTE	0	R/W	Reset Enable Selects whether to generate a signal to reset the LSI internally if WTCNT overflows in watchdog timer mode. In interval timer mode, this setting is ignored. 0: Not reset when WTCNT overflows* 1: Reset when WTCNT overflows Note: * LSI not reset internally, but WTCNT and WTCSR reset within this module.
5	RSTS	0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
4 to 0	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.

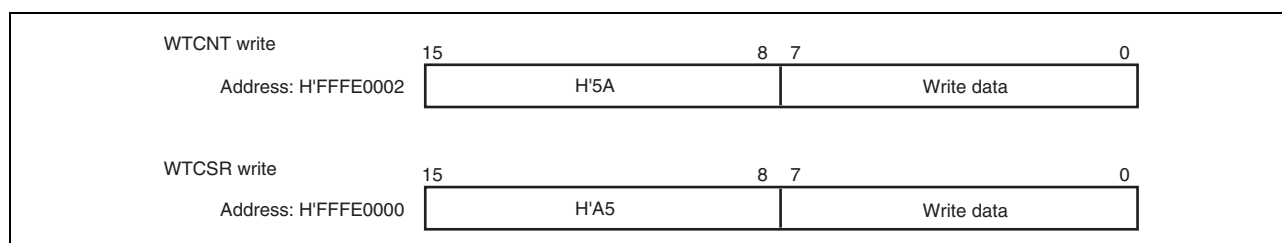
### 12.3.4 Notes on Register Access

The watchdog timer counter (WTCNT), watchdog timer control/status register (WTCSR), and watchdog reset control/status register (WRCSR) are more difficult to write to than other registers. The procedures for reading or writing to these registers are given below.

#### (1) Writing to WTCNT and WTCSR

These registers must be written by a 16-bit transfer instruction. They cannot be written by an 8- or 32-bit transfer instruction.

When writing to WTCNT, set the upper byte to H'5A and transfer the lower byte as the write data, as shown in Figure 12.2. When writing to WTCSR, set the upper byte to H'A5 and transfer the lower byte as the write data. This transfer procedure writes the lower byte data to WTCNT or WTCSR.



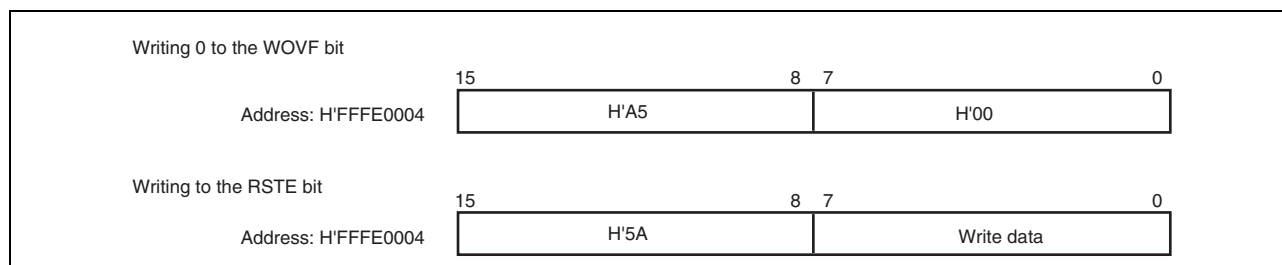
**Figure 12.2** Writing to WTCNT and WTCSR

#### (2) Writing to WRCSR

WRCSR must be written by a 16-bit access to address H'FCFE0004. It cannot be written by 8- or 32-bit transfer instructions.

Procedures for writing 0 to WOVF (bit 7) and for writing to RSTE (bit 6) and RSTS (bit 5) are different, as shown in Figure 12.3.

To write 0 to the WOVF bit, the write data must be H'A5 in the upper byte and H'00 in the lower byte. This clears the WOVF bit to 0. The RSTE and RSTS bits are not affected. To write to the RSTE and RSTS bits, the upper byte must be H'5A and the lower byte must be the write data. The values of bits 6 and 5 of the lower byte are transferred to the RSTE and RSTS bits, respectively. The WOVF bit is not affected.



**Figure 12.3** Writing to WRCSR



### (3) Reading from WTCNT, WTCSR, and WRCSR

WTCNT, WTCSR, and WRCSR are read in a method similar to other registers. WTCSR is allocated to address H'FCFE0000, WTCNT to address H'FCFE0002, and WRCSR to address H'FCFE0004. Eight-bit transfer instructions must be used for reading from these registers.

## 12.4 Usage

### 12.4.1 Canceling Software Standby Mode

This module can be used to cancel software standby mode with an interrupt such as an NMI interrupt. The procedure is described below. (This module does not operate when resets are used for canceling, so keep the  $\overline{\text{RES}}$  or  $\overline{\text{MRES}}$  pin low until clock oscillation settles.)

1. Before making a transition to software standby mode, always clear the TME bit in WTCSR to 0. When the TME bit is 1, an erroneous reset or interval timer interrupt may be generated when the count overflows.
2. Set the type of count clock used in the CKS[2:0] bits in WTCSR and the initial value of the counter in WTCNT. These values should ensure that the time till count overflow is longer than the clock oscillation settling time.
3. After setting the STBY and DEEP bits of the standby control register 1 (STBCR1: see section 55, Power-Down Modes) to 1 and 0 respectively, the execution of a WFI instruction puts the system in software standby mode and clock operation then stops.
4. This module starts counting by detecting the edge change of the NMI signal.
5. When the module count overflows, the clock pulse generator starts supplying the clock and this LSI resumes operation. The WOVF flag in WRCSR is not set when this happens.

### 12.4.2 Using Watchdog Timer Mode

1. Set the  $\overline{\text{WT/IT}}$  bit in WTCSR to 1, the type of count clock in the CKS[2:0] bits in WTCSR, whether this LSI is to be reset internally or not in the RSTE bit in WRCSR, the reset type if it is generated in the RSTS bit in WRCSR, and the initial value of the counter in WTCNT.
2. Set the TME bit in WTCSR to 1 to start the count in watchdog timer mode.
3. While operating in watchdog timer mode, rewrite the counter periodically to H'00 to prevent the counter from overflowing.
4. When the counter overflows, this module sets the WOVF flag in WRCSR to 1, and the  $\overline{\text{WDTOVF}}$  signal is output externally (Figure 12.4). The  $\overline{\text{WDTOVF}}$  signal can be used to reset the system. The  $\overline{\text{WDTOVF}}$  signal is output for  $64 \times P0\phi$  clock cycles.
5. If the RSTE bit in WRCSR is set to 1, a signal to reset the inside of this LSI can be generated simultaneously with the  $\overline{\text{WDTOVF}}$  signal. Either power-on reset or manual reset can be selected for this interrupt by the RSTS bit in WRCSR. The internal reset signal is output for  $128 \times P0\phi$  clock cycles.
6. When an overflow reset of this module is generated simultaneously with a reset input on the  $\overline{\text{RES}}$  pin, the  $\overline{\text{RES}}$  pin reset takes priority, and the WOVF bit in WRCSR is cleared to 0.

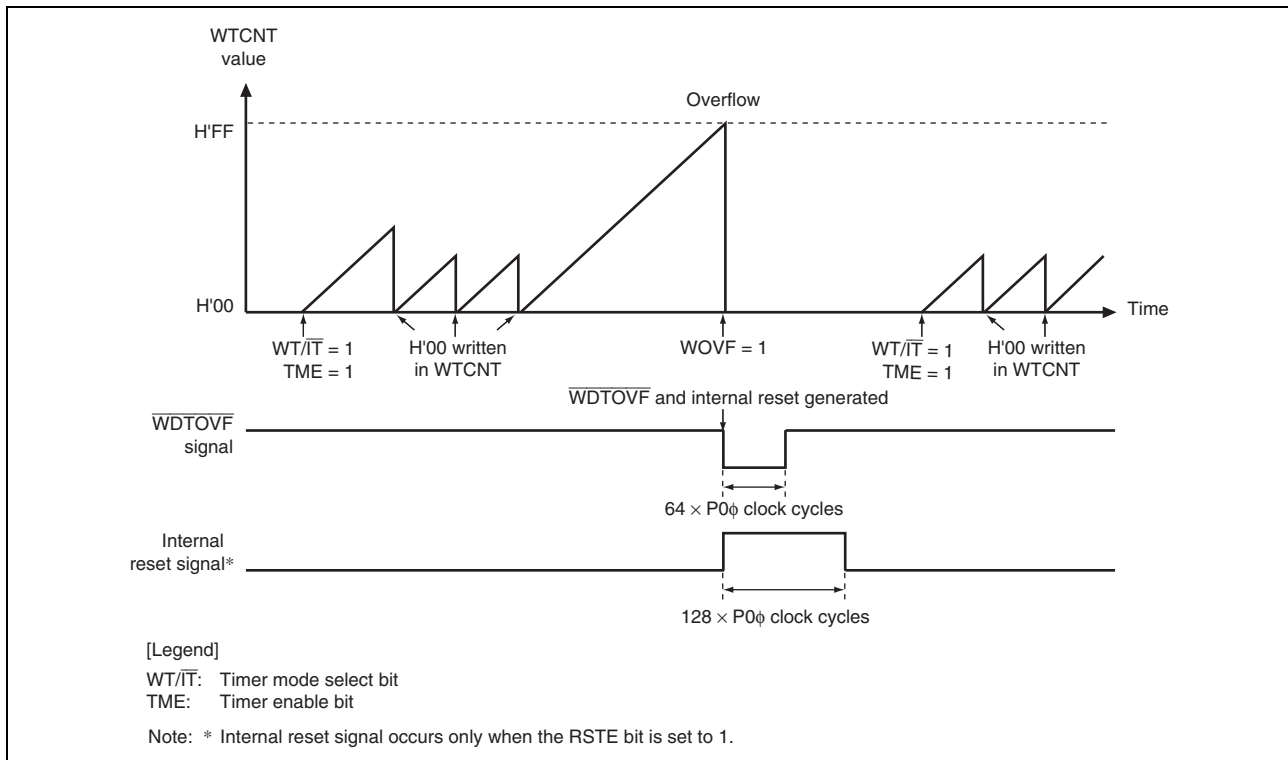


Figure 12.4 Operation in Watchdog Timer Mode

### 12.4.3 Using Interval Timer Mode

When operating in interval timer mode, interval timer interrupts are generated at every overflow of the counter. This enables interrupts to be generated at set periods.

1. Clear the WT/IT bit in WTCSR to 0, set the type of count clock in the CKS[2:0] bits in WTCSR, and set the initial value of the counter in WTCNT.
2. Set the TME bit in WTCSR to 1 to start the count in interval timer mode.
3. When the counter overflows, this module sets the IOVF bit in WTCSR to 1 and an interval timer interrupt request is sent to the interrupt controller. The counter then resumes counting.

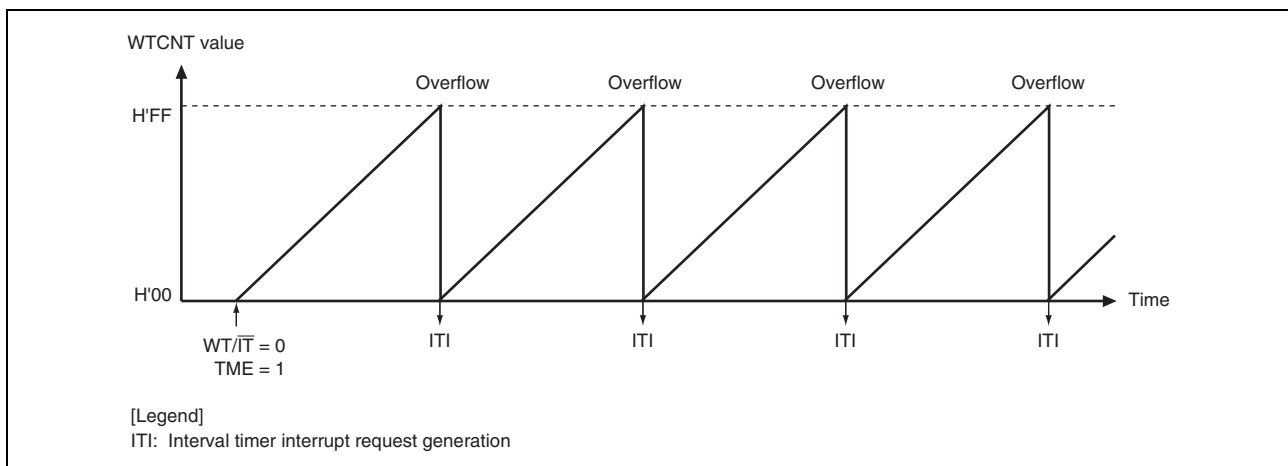


Figure 12.5 Operation in Interval Timer Mode

## 12.5 Usage Notes

Pay attention to the following points when using this module in either the interval timer or watchdog timer mode.

### 12.5.1 Timer Variation

After timer operation has started, the period from the power-on reset point to the first count up timing of WTCNT varies depending on the time period that is set by the TME bit of WTCSR. The shortest such time period is thus one cycle of the peripheral clock, P0φ, while the longest is the result of frequency division according to the value in the CKS[2:0] bits. The timing of subsequent incrementation is in accord with the selected frequency division ratio. Accordingly, this time difference is referred to as timer variation.

This also applies to the timing of the first incrementation after WTCNT has been written to during timer operation.

### 12.5.2 Prohibition against Setting H'FF to WTCNT

When the value in WTCNT reaches H'FF, this module assumes that an overflow has occurred. Accordingly, when H'FF is set in WTCNT, an interval timer interrupt or reset will occur immediately, regardless of the current clock selection by the CKS[2:0] bits.

### 12.5.3 Interval Timer Overflow Flag

When the value in WTCNT is H'FF, the IOVF flag in WTCSR cannot be cleared.

Only clear the IOVF flag when the value in WTCNT has either become H'00 or been changed to a value other than H'FF.

### 12.5.4 System Reset by $\overline{\text{WDTOVF}}$ Signal

If the  $\overline{\text{WDTOVF}}$  signal is input to the  $\overline{\text{RES}}$  pin of this LSI, this LSI cannot be initialized correctly.

Avoid input of the  $\overline{\text{WDTOVF}}$  signal to the  $\overline{\text{RES}}$  pin of this LSI through glue logic circuits. To reset the entire system with the  $\overline{\text{WDTOVF}}$  signal, use the circuit shown in Figure 12.6.

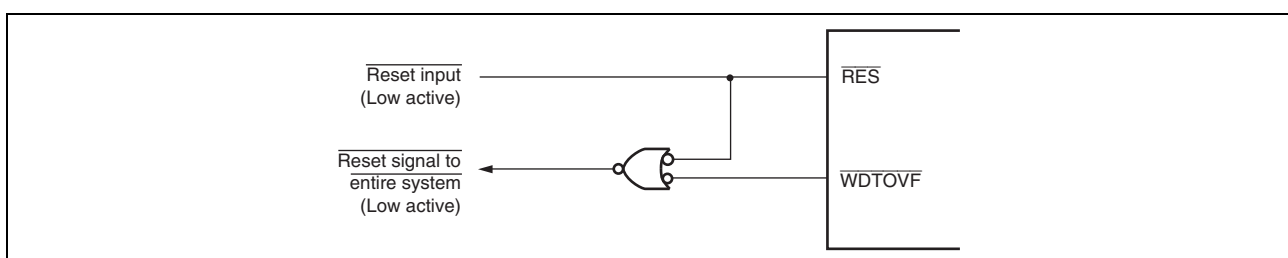


Figure 12.6 Example of System Reset Circuit Using  $\overline{\text{WDTOVF}}$  Signal

### 12.5.5 Internal Reset in Watchdog Timer Mode

When an internal reset is generated due to an overflow of the watchdog timer counter (WTCNT) in watchdog timer mode, the watchdog reset control/status register (WRCSR) is not initialized, so the WOVF bit retains the value 1. As long as the WOVF bit is 1, an internal reset will not be generated even if the WTCNT overflows again.

## 13. Realtime Clock

This LSI has a realtime clock and 32.768-kHz and 4-MHz crystal oscillators.

### 13.1 Features

- Clock and calendar functions (BCD format): Seconds, minutes, hours, day of the week, day, month, and year.
- 1-Hz to 64-Hz timer (binary format)  
64-Hz counter indicates the state of the divider circuit between 64 Hz and 1 Hz
- Start/stop function
- 30-second adjust function
- Alarm interrupt: Comparison with seconds, minutes, hours, day of the week, day, month, or year can be selected as a condition for the alarm interrupt
- Periodic interrupt: the interrupt cycle may be 1/64 second, 1/16 second, 1/4 second, 1/2 second, 1 second, or 2 seconds
- Carry interrupt: a carry interrupt indicates that a second counter carry is generated or a 64-Hz counter carry is generated reading the 64-Hz counter
- Automatic leap year adjustment
- The external clock signal input for internal operation or the external clock signal input dedicated to the clock operation can be selected as the operating clock signal for the clock function.
- Recovery from deep standby mode can be performed by an alarm interrupt.

Figure 13.1 shows the block diagram.

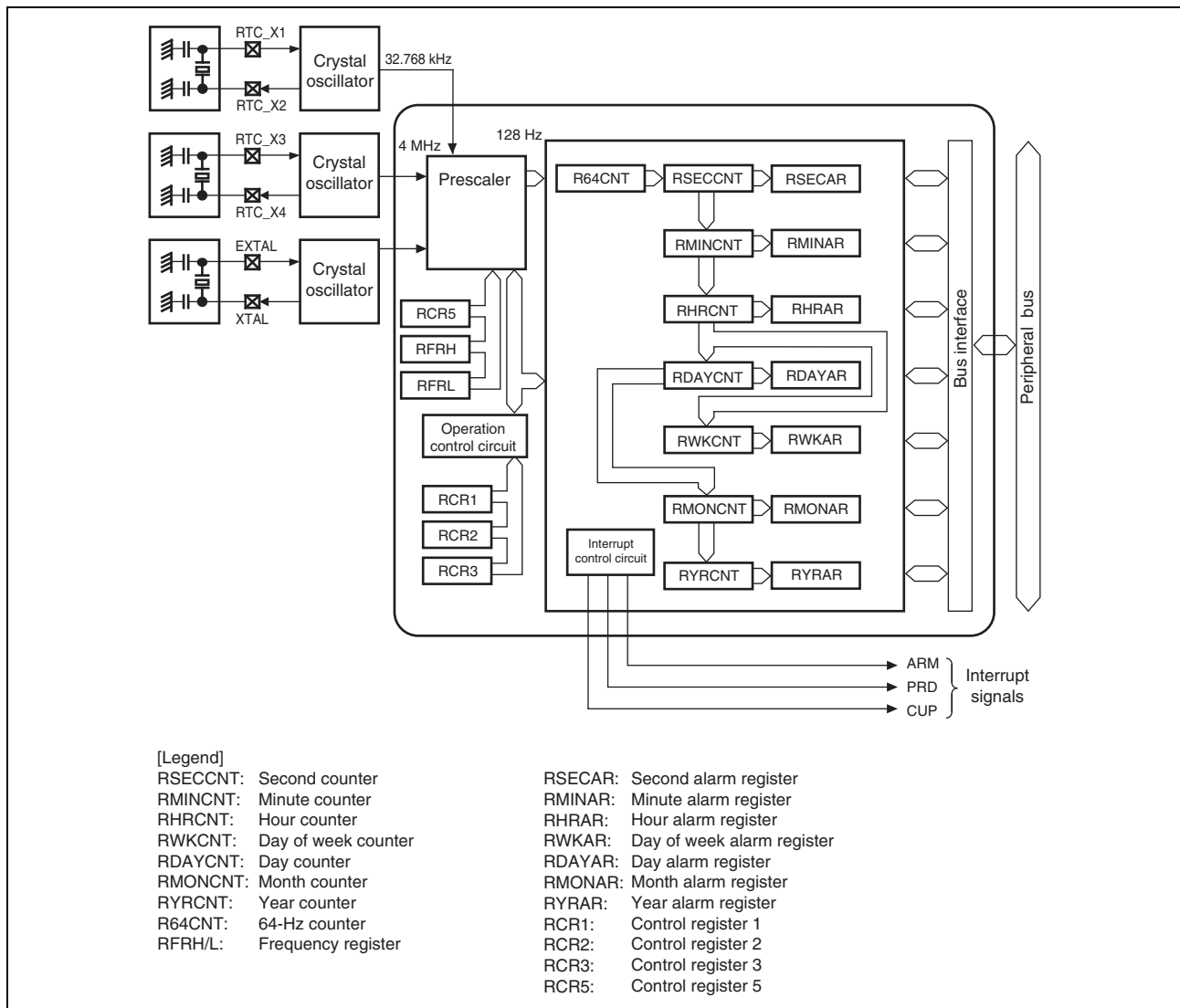


Figure 13.1 Block Diagram

## 13.2 Input/Output Pin

Table 13.1 shows the pin configuration.

**Table 13.1 Pin Configuration**

Pin Name	Symbol	I/O	Description
Realtime clock crystal resonator pin/external clock	RTC_X1	Input	Connects a 32.768-kHz crystal resonator for this module. External clock can be input to the RTC_X1 pin.
	RTC_X2	Output	
	RTC_X3	Input	Connects a 4-MHz crystal resonator for this module. External clock can be input to the RTC_X3 pin.
	RTC_X4	Output	
Internal clock crystal resonator/external clock	EXTAL	Input	Connects crystal resonator used for internal operation. For details, see section 6, Clock Pulse Generator.
	XTAL	Output	

## 13.3 Register Descriptions

Table 13.2 shows the register configuration.

**Table 13.2 Register Configuration**

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
64-Hz counter	R64CNT	R	H'xx	H'FCFF1000	8
Second counter	RSECCNT	R/W	H'xx	H'FCFF1002	8
Minute counter	RMINCNT	R/W	H'xx	H'FCFF1004	8
Hour counter	RHRCNT	R/W	H'xx	H'FCFF1006	8
Day of week counter	RWKCNT	R/W	H'0x	H'FCFF1008	8
Day counter	RDAYCNT	R/W	H'xx	H'FCFF100A	8
Month counter	RMONCNT	R/W	H'xx	H'FCFF100C	8
Year counter	RYRCNT	R/W	H'xxxx	H'FCFF100E	16
Second alarm register	RSECAR	R/W	H'xx	H'FCFF1010	8
Minute alarm register	RMINAR	R/W	H'xx	H'FCFF1012	8
Hour alarm register	RHRAR	R/W	H'xx	H'FCFF1014	8
Day of week alarm register	RWKAR	R/W	H'xx	H'FCFF1016	8
Day alarm register	RDAYAR	R/W	H'xx	H'FCFF1018	8
Month alarm register	RMONAR	R/W	H'xx	H'FCFF101A	8
Year alarm register	RYRAR	R/W	H'xxxx	H'FCFF1020	16
Control register 1	RCR1	R/W	H'xx	H'FCFF101C	8
Control register 2	RCR2	R/W	H'01	H'FCFF101E	8
Control register 3	RCR3	R/W	H'x0	H'FCFF1024	8
Control register 5	RCR5	R/W	H'0x	H'FCFF1026	8
Frequency register	RFRH	R/W	H'xxxx	H'FCFF102A	16
	RFRL	R/W	H'xxxx	H'FCFF102C	16

### 13.3.1 64-Hz Counter (R64CNT)

R64CNT indicates the state of the divider circuit between 64 Hz and 1 Hz.

Reading this register, when carry from 128-Hz divider stage is generated, sets the CF bit in the control register 1 (RCR1) to 1, which indicates that the carrying and reading the 64-Hz counter are performed at the same time. In this case, the R64CNT should be read again after writing 0 to the CF bit in RCR1 since the read value is not valid.

Setting the RESET or ADJ bit in the control register 2 (RCR2) to 1 initializes the divider circuit and the R64CNT.

Bit:	7	6	5	4	3	2	1	0
	-	1Hz	2Hz	4Hz	8Hz	16Hz	32Hz	64Hz
Initial value:	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W:	R	R	R	R	R	R	R	R

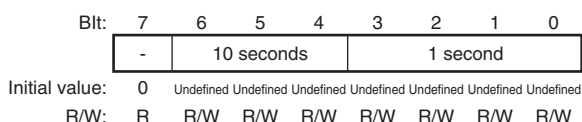
Bit	Bit Name	Initial Value	R/W	Description
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	1 Hz	Undefined	R	Indicate the state of the divider circuit between 64 Hz and 1 Hz.
5	2 Hz	Undefined	R	
4	4 Hz	Undefined	R	
3	8 Hz	Undefined	R	
2	16 Hz	Undefined	R	
1	32 Hz	Undefined	R	
0	64 Hz	Undefined	R	



### 13.3.2 Second Counter (RSECCNT)

RSECCNT is the counter used for setting/counting the BCD-coded second value. The count operation is performed by a carry for each second of the 64-Hz counter.

The assignable range is from 00 through 59 (practically in BCD); otherwise an operation error will occur. Carry out write processing after stopping the count operation through the setting of the START bit in RCR2.

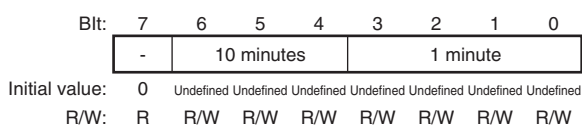


Bit	Bit Name	Initial Value	R/W	Description
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6 to 4	10 seconds	Undefined	R/W	Counting Ten's Position of Seconds Counts from 0 to 5 for 60-seconds counting.
3 to 0	1 second	Undefined	R/W	Counting One's Position of Seconds Counts from 0 to 9, one number per second. When a carry is generated, 1 is added to the ten's position.

### 13.3.3 Minute Counter (RMINCNT)

RMINCNT is the counter used for setting/counting the BCD-coded minute value. The count operation is performed by a carry for each minute of the second counter.

The assignable range is from 00 through 59 (practically in BCD); otherwise an operation error will occur. Carry out write processing after stopping the count operation through the setting of the START bit in RCR2.



Bit	Bit Name	Initial Value	R/W	Description
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6 to 4	10 minutes	Undefined	R/W	Counting Ten's Position of Minutes Counts from 0 to 5 for 60-minutes counting.
3 to 0	1 minute	Undefined	R/W	Counting One's Position of Minutes Counts from 0 to 9, one number per minute. When a carry is generated, 1 is added to the ten's position.

### 13.3.4 Hour Counter (RHRCNT)

RHRCNT is the counter used for setting/counting the BCD-coded hour value. The count operation is performed by a carry for each 1 hour of the minute counter.

The assignable range is from 00 through 23 (practically in BCD); otherwise an operation error will occur. Carry out write processing after stopping the count operation through the setting of the START bit in RCR2.

Bit:	7	6	5	4	3	2	1	0
	-	-	10 hours	1 hour				
Initial value:	0	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5, 4	10 hours	Undefined	R/W	Counting Ten's Position of Hours Counts from 0 to 2 for ten's position of hours.
3 to 0	1 hour	Undefined	R/W	Counting One's Position of Hours Counts from 0 to 9, one number per hour. When a carry is generated, 1 is added to the ten's position.

### 13.3.5 Day of Week Counter (RWKCNT)

RWKCNT is the counter used for setting/counting the BCD-coded day-of-week value. The count operation is performed by a carry for each day of the hour counter.

The assignable range is from 0 through 6 (practically in BCD); otherwise an operation error will occur. Carry out write processing after stopping the count operation through the setting of the START bit in RCR2.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	Day		
Initial value:	0	0	0	0	0	Undefined	Undefined	Undefined
R/W:	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	Day	Undefined	R/W	Day-of-Week Counting Day-of-week is indicated with a binary code. 000: Sunday 001: Monday 010: Tuesday 011: Wednesday 100: Thursday 101: Friday 110: Saturday 111: Reserved (setting prohibited)

### 13.3.6 Day Counter (RDAYCNT)

RDAYCNT is the counter used for setting/counting the BCD-coded day value. The count operation is performed by a carry for each day of the hour counter.

The assignable range is from 01 through 31 (practically in BCD); otherwise an operation error will occur. Carry out write processing after stopping the count operation through the setting of the START bit in RCR2.

The assignable range changes depending on the month and in leap years. Confirm the correct setting. A leap year is determined by checking if the year counter (RYRCNT) value is divisible by 400, 100, and 4.

Bit:	7	6	5	4	3	2	1	0
	-	-	10 days		1 day			
Initial value:	0	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5, 4	10 days	Undefined	R/W	Counting Ten's Position of Days
3 to 0	1 day	Undefined	R/W	Counting One's Position of Days Counts from 0 to 9, one number per day. When a carry is generated, 1 is added to the ten's position.

### 13.3.7 Month Counter (RMONCNT)

RMONCNT is the counter used for setting/counting the BCD-coded month value. The count operation is performed by a carry for each month of the date counter.

The assignable range is from 01 through 12 (practically in BCD); otherwise an operation error will occur. Carry out write processing after stopping the count operation through the setting of the START bit in RCR2.

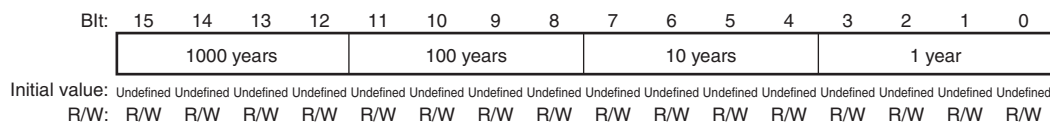
Bit:	7	6	5	4	3	2	1	0
	-	-	-	10 months	1 month			
Initial value:	0	0	0	Undefined	Undefined	Undefined	Undefined	Undefined
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	10 months	Undefined	R/W	Counting Ten's Position of Months
3 to 0	1 month	Undefined	R/W	Counting One's Position of Months Counts from 0 to 9, one number per month. When a carry is generated, 1 is added to the ten's position.

### 13.3.8 Year Counter (RYRCNT)

RYRCNT is the counter used for setting/counting the BCD-coded year value. The count operation is performed by a carry for each year of the month counter.

The assignable range is from 0000 through 9999 (practically in BCD); otherwise an operation error will occur. Carry out write processing after stopping the count operation through the setting of the START bit in RCR2.

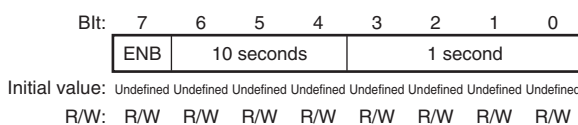


Bit	Bit Name	Initial Value	R/W	Description
15 to 12	1000 years	Undefined	R/W	Counting Thousand's Position of Years
11 to 8	100 years	Undefined	R/W	Counting Hundred's Position of Years
7 to 4	10 years	Undefined	R/W	Counting Ten's Position of Years
3 to 0	1 year	Undefined	R/W	Counting One's Position of Years

### 13.3.9 Second Alarm Register (RSECAR)

RSECAR is an alarm register corresponding to the BCD-coded second counter RSECCNT. When the ENB bit is set to 1, a comparison with the RSECCNT value is performed. From among RSECAR/RMINAR/RHRAR/RWKAR/RDAYAR/RMONAR/RCR3, the counter and alarm register comparison is performed only on those with ENB bits set to 1, and if each of those coincides, an alarm flag of RCR1 is set to 1.

The assignable range is from 00 through 59 + ENB bits (practically in BCD); otherwise an operation error will occur.

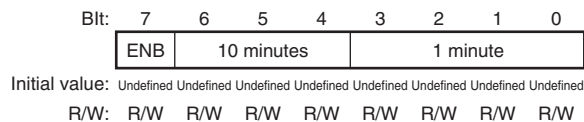


Bit	Bit Name	Initial Value	R/W	Description
7	ENB	Undefined	R/W	When this bit is set to 1, a comparison with the RSECCNT value is performed.
6 to 4	10 seconds	Undefined	R/W	Ten's position of seconds setting value
3 to 0	1 second	Undefined	R/W	One's position of seconds setting value

### 13.3.10 Minute Alarm Register (RMINAR)

RMINAR is an alarm register corresponding to the BCD-coded minute counter RMINCNT. When the ENB bit is set to 1, a comparison with the RMINCNT value is performed. From among RSECAR/RMINAR/RHRAR/RWKAR/RDAYAR/RMONAR/RCR3, the counter and alarm register comparison is performed only on those with ENB bits set to 1, and if each of those coincides, an alarm flag of RCR1 is set to 1.

The assignable range is from 00 through 59 + ENB bits (practically in BCD); otherwise an operation error will occur.

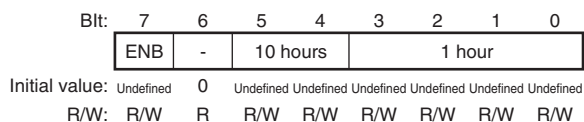


Bit	Bit Name	Initial Value	R/W	Description
7	ENB	Undefined	R/W	When this bit is set to 1, a comparison with the RMINCNT value is performed.
6 to 4	10 minutes	Undefined	R/W	Ten's position of minutes setting value
3 to 0	1 minute	Undefined	R/W	One's position of minutes setting value

### 13.3.11 Hour Alarm Register (RHRAR)

RHRAR is an alarm register corresponding to the BCD-coded hour counter RHRCNT. When the ENB bit is set to 1, a comparison with the RHRCNT value is performed. From among RSECAR/RMINAR/RHRAR/RWKAR/RDAYAR/RMONAR/RCR3, the counter and alarm register comparison is performed only on those with ENB bits set to 1, and if each of those coincides, an alarm flag of RCR1 is set to 1.

The assignable range is from 00 through 23 + ENB bits (practically in BCD); otherwise an operation error will occur.



Bit	Bit Name	Initial Value	R/W	Description
7	ENB	Undefined	R/W	When this bit is set to 1, a comparison with the RHRCNT value is performed.
6	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
5, 4	10 hours	Undefined	R/W	Ten's position of hours setting value
3 to 0	1 hour	Undefined	R/W	One's position of hours setting value

### 13.3.12 Day of Week Alarm Register (RWKAR)

RWKAR is an alarm register corresponding to the BCD-coded day of week counter RWKCNT. When the ENB bit is set to 1, a comparison with the RWKCNT value is performed. From among RSECAR/RMINAR/RHRAR/RWKAR/RDAYAR/RMONAR/RCR3, the counter and alarm register comparison is performed only on those with ENB bits set to 1, and if each of those coincides, an alarm flag of RCR1 is set to 1.

The assignable range is from 0 through 6 + ENB bits (practically in BCD); otherwise an operation error will occur.

Bit:	7	6	5	4	3	2	1	0
	ENB	-	-	-	-	Day		
Initial value:	Undefined	0	0	0	0	Undefined	Undefined	Undefined
R/W:	R/W	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	ENB	Undefined	R/W	When this bit is set to 1, a comparison with the RWKCNT value is performed.
6 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	Day	Undefined	R/W	Day of Week Setting Value 000: Sunday 001: Monday 010: Tuesday 011: Wednesday 100: Thursday 101: Friday 110: Saturday 111: Reserved (setting prohibited)

### 13.3.13 Day Alarm Register (RDAYAR)

RDAYAR is an alarm register corresponding to the BCD-coded day counter RDAYCNT. When the ENB bit is set to 1, a comparison with the RDAYCNT value is performed. From among RSECAR/RMINAR/RHRAR/RWKAR/RDAYAR/RMONAR/RCR3, the counter and alarm register comparison is performed only on those with ENB bits set to 1, and if each of those coincides, an alarm flag of RCR1 is set to 1.

The assignable range is from 01 through 31 + ENB bits (practically in BCD); otherwise an operation error will occur.

Bit:	7	6	5	4	3	2	1	0
	ENB	-	10 days		1 day			
Initial value:	Undefined	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W:	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	ENB	Undefined	R/W	When this bit is set to 1, a comparison with the RDAYCNT value is performed.
6	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
5, 4	10 days	Undefined	R/W	Ten's position of days setting value
3 to 0	1 day	Undefined	R/W	One's position of days setting value

### 13.3.14 Month Alarm Register (RMONAR)

RMONAR is an alarm register corresponding to the BCD-coded month counter RMONCNT. When the ENB bit is set to 1, a comparison with the RMONCNT value is performed. From among RSECAR/RMINAR/RHRAR/RWKAR/RDAYAR/RMONAR/RRCR3, the counter and alarm register comparison is performed only on those with ENB bits set to 1, and if each of those coincides, an alarm flag of RCR1 is set to 1.

The assignable range is from 01 through 12 + ENB bits (practically in BCD); otherwise an operation error will occur.

Bit:	7	6	5	4	3	2	1	0
	ENB	-	-	10 months	1 month			
Initial value:	Undefined	0	0	Undefined	Undefined	Undefined	Undefined	Undefined
R/W:	R/W	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	ENB	Undefined	R/W	When this bit is set to 1, a comparison with the RMONCNT value is performed.
6, 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	10 months	Undefined	R/W	Ten's position of months setting value
3 to 0	1 month	Undefined	R/W	One's position of months setting valu

### 13.3.15 Year Alarm Register (RYRAR)

RYRAR is an alarm register corresponding to the BCD-coded year counter RYRCNT. The assignable range is from 0000 through 9999 (practically in BCD); otherwise an operation error will occur.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	1000 years				100 years				10 years				1 year			
Initial value:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	1000 years	Undefined	R/W	Thousand's position of years setting value
11 to 8	100 years	Undefined	R/W	Hundred's position of years setting value
7 to 4	10 years	Undefined	R/W	Ten's position of years setting value
3 to 0	1 year	Undefined	R/W	One's position of years setting value

### 13.3.16 Control Register 1 (RCR1)

RCR1 is a register that affects carry flags and alarm flags. It also selects whether to generate interrupts for each flag. The CF flag remains undefined until the divider circuit is reset (the RESET and ADJ bits in RCR2 are set to 1). When using the CF flag, make sure to reset the divider circuit beforehand.

The AF flag remains undefined until the value is set to an alarm register and a counter. When using the AF flag, make sure to set the alarm register and counter beforehand.

Bit:	7	6	5	4	3	2	1	0
	CF	-	-	CIE	AIE	-	-	AF
Initial value:	Undefined	0	0	0	0	0	0	Undefined
R/W:	R/W	R	R	R/W	R/W	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	CF	Undefined	R/W	<b>Carry Flag</b> Status flag that indicates that a carry has occurred. CF is set to 1 when a second counter carry is generated or a 64-Hz counter carry is generated while reading the 64-Hz counter A value read from the count register at this time cannot be guaranteed; another read is required. 0: No carry of second counter or no carry of 64-Hz counter while reading 64-Hz counter [Clearing condition] When 0 is written to CF 1: Carry of second counter or carry of 64-Hz counter while reading 64-Hz counter [Setting condition] When a second counter carry is generated or a 64-Hz counter carry is generated while reading the 64-Hz counter, or 1 is written to CF.
6, 5	—	All 0	R	<b>Reserved</b> These bits are always read as 0. The write value should always be 0.
4	CIE	0	R/W	<b>Carry Interrupt Enable Flag</b> When the carry flag (CF) is set to 1, the CIE bit enables an interrupt. 0: A carry interrupt is not generated when the CF flag is set to 1 1: A carry interrupt is generated when the CF flag is set to 1
3	AIE	0	R/W	<b>Alarm Interrupt Enable Flag</b> When the alarm flag (AF) is set to 1, the AIE bit enables an interrupt. 0: An alarm interrupt is not generated when the AF flag is set to 1 1: An alarm interrupt is generated when the AF flag is set to 1
2, 1	—	All 0	R	<b>Reserved</b> These bits are always read as 0. The write value should always be 0.
0	AF	Undefined	R/W	<b>Alarm Flag</b> The AF flag is set when the alarm time, which is set by the alarm registers (RSECAR, RMINAR, RHRAR, RWKAR, RDAYAR, RMONAR, or RYRAR whose ENB bit is set to 1), and counters match. 0: Alarm registers and counters not match [Clearing condition] When 0 is written to AF 1: Alarm registers and counters match* [Setting condition] When alarm registers (only the registers with ENB bit set to 1) and counters match Note: * Writing 1 holds previous value.



### 13.3.17 Control Register 2 (RCR2)

RCR2 is a register for periodic interrupt control, 30-second adjustment, divider circuit RESET, and count control.

RCR2 is initialized by a power-on reset or in deep standby mode. The RTCEN bit is only initialized by a power-on reset using the  $\overline{\text{RES}}$  pin.

Bit:	7	6	5	4	3	2	1	0
	PEF	PES[2:0]			RTCEN	ADJ	RESET	START
Initial value:	0	0	0	0	1	0	0	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	PEF	0	R/W	<p>Periodic Interrupt Flag</p> <p>Indicates that an interrupt is generated with the period designated by the PES2 to PES0 bits. When set to 1, PEF generates periodic interrupts.</p> <p>0: Interrupts not generated with the period designated by the bits PES2 to PES0. [Clearing condition]</p> <p>When 0 is written to PEF</p> <p>1: Interrupts generated with the period designated by the PES2 to PES0 bits. [Setting condition]</p> <p>When an interrupt is generated with the period designated by the bits PES0 to PES2 or when 1 is written to the PEF flag</p>
6 to 4	PES[2:0]	000	R/W	<p>Interrupt Enable Flags</p> <p>These bits specify the period of the periodic interrupt.</p> <p>000: No periodic interrupts generated</p> <p>001: Setting prohibited</p> <p>010: Periodic interrupt generated every 1/64 second</p> <p>011: Periodic interrupt generated every 1/16 second</p> <p>100: Periodic interrupt generated every 1/4 second</p> <p>101: Periodic interrupt generated every 1/2 second</p> <p>110: Periodic interrupt generated every 1 second</p> <p>111: Periodic interrupt generated every 2 seconds</p>
3	RTCEN	1	R/W	<p>RTC_X1 and RTC_X3 Clock Control</p> <p>Controls the function of the RTC_X1 and RTC_X3 pins.</p> <p>0: Halts the on-chip crystal oscillator/disables the external clock input.</p> <p>1: Runs the on-chip crystal oscillator/enables the external clock input.</p> <p>Note: The on-chip crystal oscillator selected with the RCKSEL[1:0] bits in the RCR5 register runs. This bit must not be set to 1 when the RCKSEL[1:0] bits are set to 01.</p>
2	ADJ	0	R/W	<p>30-Second Adjustment</p> <p>When 1 is written to the ADJ bit, times of 29 seconds or less will be rounded down to 00 seconds and 30 seconds or more up to 1 minute. The divider circuit (prescaler and R64CNT) will be simultaneously reset. The ADJ bit is automatically reset to 0; there is no need to write 0 to this bit. This bit is always read as 0.</p> <p>0: Normal clock operation</p> <p>1: 30-second adjustment</p>
1	RESET	0	R/W	<p>Reset</p> <p>Writing 1 to this bit initializes the divider circuit, the R64CNT register, the alarm register, the RCR3 register, bits CF and AF in RCR1, and bit PEF in RCR2. In this case, the RESET bit is automatically reset to 0 after 1 is written to and the above registers are reset. Thus, there is no need to write 0 to this bit. This bit is always read as 0.</p> <p>0: Normal clock operation</p> <p>1: Divider circuit is reset.</p>
0	START	1	R/W	<p>Start</p> <p>Halts and restarts the counter (clock).</p> <p>0: Second, minute, hour, day, week, month, and year counters halt.</p> <p>1: Second, minute, hour, day, week, month, and year counters run normally</p>

### 13.3.18 Control Register 3 (RCR3)

When the ENB bit is set to 1, RCR3 performs a comparison with the RYRCNT. From among RSECAR/RMINAR/RHRAR/RWKAR/RDAYAR/RMONAR/RCR3, the counter and alarm register comparison is performed only on those with ENB bits set to 1, and if each of those coincides, an alarm flag of RCR1 is set to 1.

Bit:	7	6	5	4	3	2	1	0
	ENB	-	-	-	-	-	-	-
Initial value:	Undefined	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	ENB	Undefined	R/W	When this bit is set to 1, comparison of the year alarm register (RYRAR) and the year counter (RYRCNT) is performed.
6 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

### 13.3.19 Control Register 5 (RCR5)

When the RCKSEL[1:0] bits in RCR5 are set to 00, 01, and 10, the RTC\_X1 clock pulses (32.768 kHz), the EXTAL clock pulses, and the RTC\_X3 clock pulses are used for clock counting, respectively.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	RCKSEL[1:0]	
Initial value:	0	0	0	0	0	0	Undefined	Undefined
R/W:	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	RCKSEL[1:0]	Undefined	R/W	Operation clock select Operation clock can be selected from RTC_X1, EXTAL, and RTC_X3. The setting of these bits should not be switched during operation. 00: Selects RTC_X1 (32.768 kHz). 01: Selects EXTAL. 10: Selects RTC_X3. 11: Setting prohibited.

### 13.3.20 Frequency Register H/L (RFRH/L)

RFRH/L is a 16-bit readable/writable register.

The "frequency comparison value" is set in RFC[18:0] so that a 128-Hz clock is generated when the realtime clock operates at the EXTAL or RTC\_X3 clock frequency.

Change the "frequency comparison value" according to the EXTAL clock frequency. The calculation method is shown below. When the RCKSEL bits in RCR5 are set to 00, setting this register is not necessary.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SEL64	-	-	-	-	-	-	-	-	-	-	-	-	RFC[18:16]		
Initial value:	Undefined	0	0	0	0	0	0	0	0	0	0	0	0	Undefined	Undefined	Undefined
R/W:	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFC[15:0]															
Initial value:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	SEL64	Undefined	R/W	64-Hz Divider Select Indicates the EXTAL or RTC_X3 clock frequency is not dividable by 128 Hz but is dividable by 64 Hz. 0: EXTAL or RTC_X3 clock frequency is dividable by 128 Hz. 1: EXTAL or RTC_X3 clock frequency is not dividable by 128 Hz but is dividable by 64 Hz.
30 to 19	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
18 to 0	RFC[18:0]	Undefined	R/W	Frequency comparison value Sets the comparison value to generate operation clock from the EXTAL or RTC_X3 clock frequency.

#### (1) Method for calculating "frequency comparison value".

- When EXTAL clock frequency is dividable by 128 Hz:  

$$\text{RFC}[18:0] = (\text{EXTAL or RTC\_X3 clock frequency})/128$$
 Clear the SEL64 bit to 0.
- When EXTAL clock frequency is not dividable by 128 Hz but is dividable by 64 Hz:  

$$\text{RFC}[18:0] = (\text{EXTAL or RTC\_X3 clock frequency})/64$$
 Set the SEL64 bit to 1.

#### (2) Setting Example

**Table 13.3 Setting Example**

Clock Frequency		SEL64 Setting Value	RFC Setting Value
EXTAL	10 MHz	0	H'1312D
	11 MHz	1	H'29F63
	12 MHz	0	H'16E36
	13 MHz	1	H'31975
RTC_X3	4 MHz	0	H'07A12

## 13.4 Operation

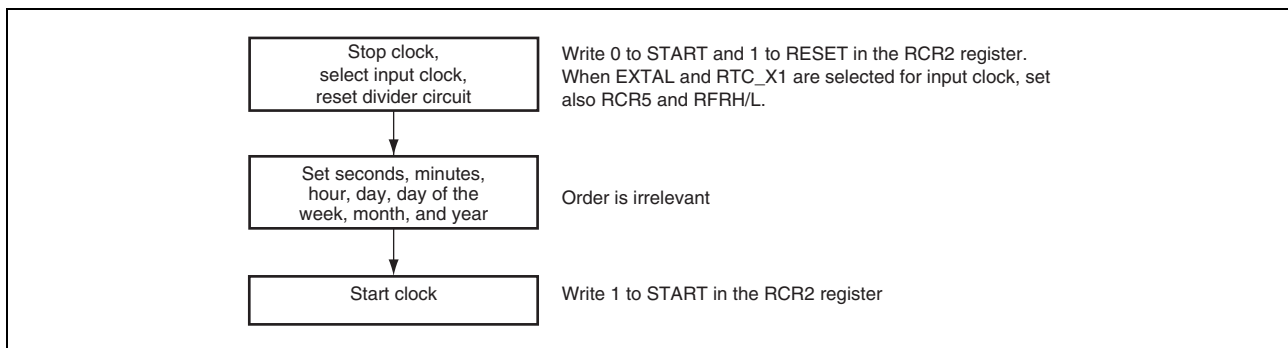
A usage example of this module is shown below.

### 13.4.1 Initial Settings of Registers after Power-On and Oscillation Stabilization Time

All the registers should be initialized after the power is turned on. When the 4-MHz crystal oscillator is used, oscillation stabilization time is necessary after changing the RTCEN bit in RCR2 from 0 to 1. During oscillation stabilization time, various configurations for or operation of the real time clock must not be performed. For details on oscillation stabilization time, refer to section 58, Electrical Characteristics.

### 13.4.2 Setting Time

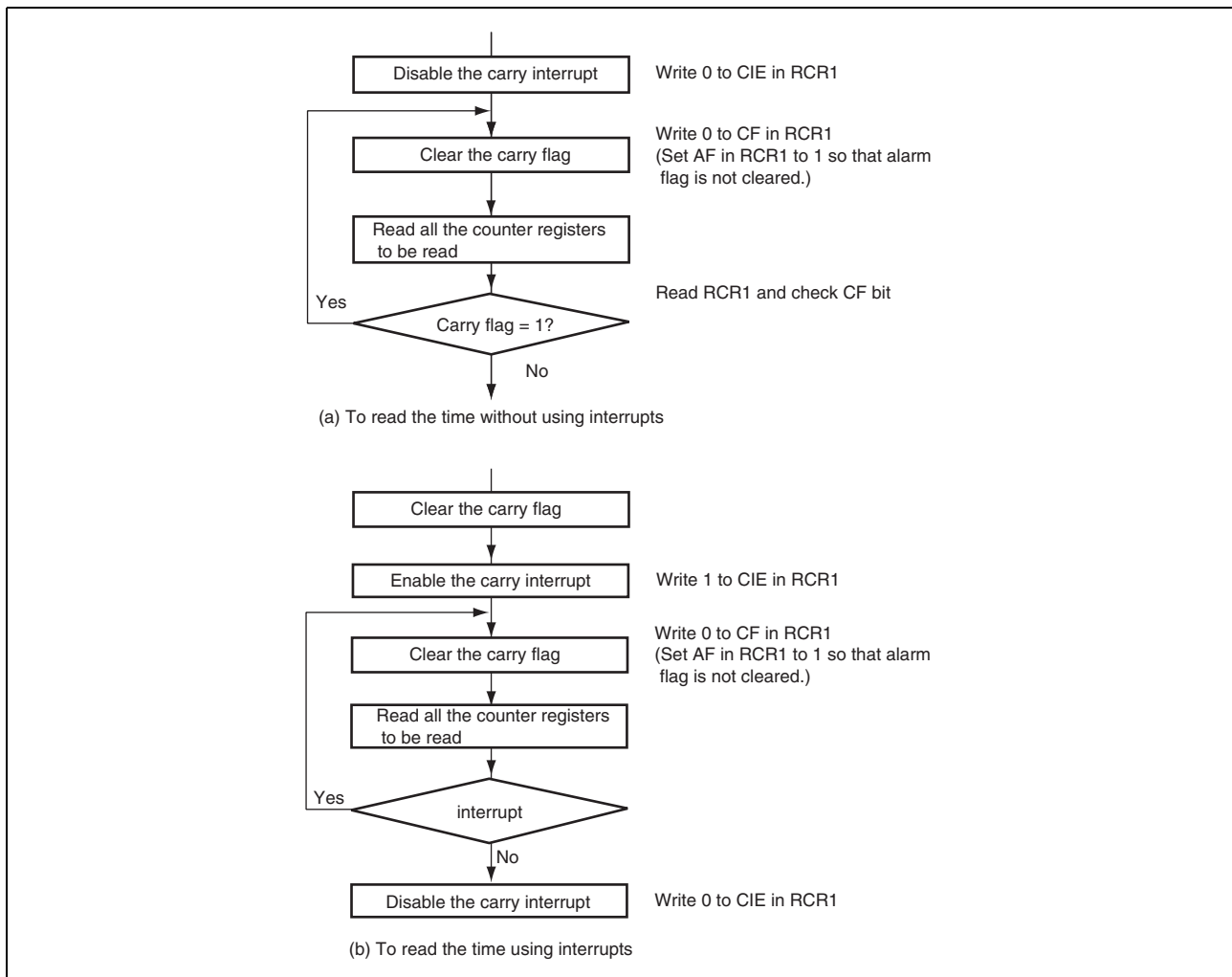
Figure 13.2 shows how to set the time when the clock is stopped.



**Figure 13.2** Setting Time

### 13.4.3 Reading Time

Figure 13.3 shows how to read the time.

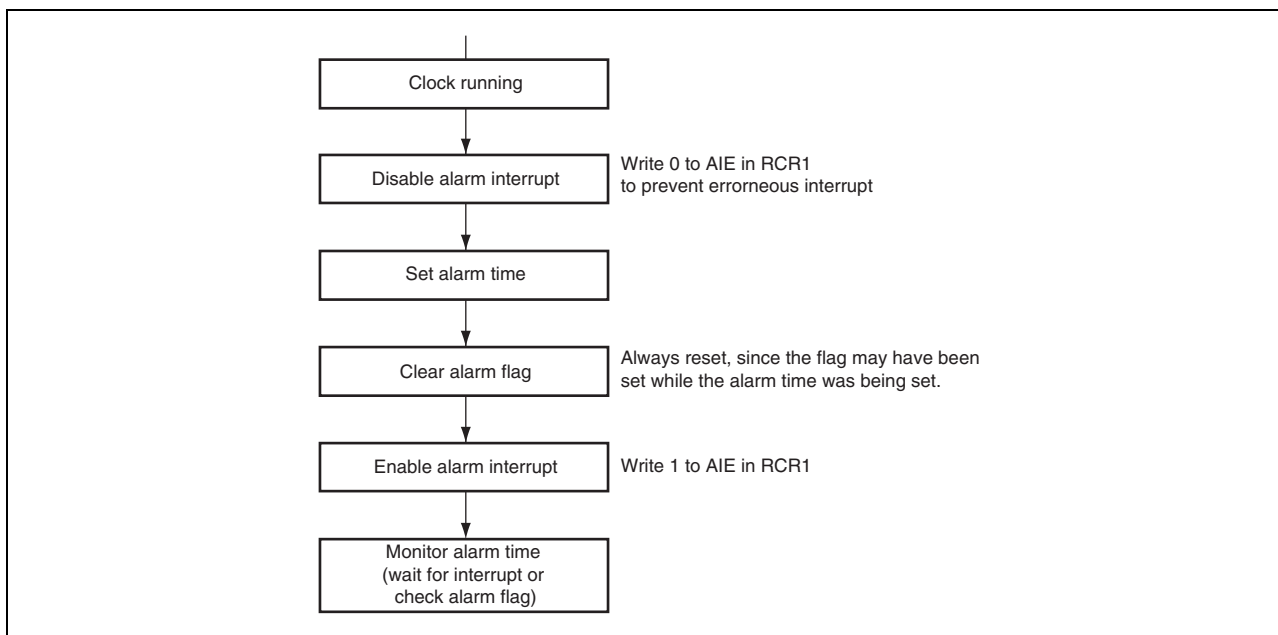


**Figure 13.3 Reading Time**

If a carry occurs while reading the time, the correct time will not be obtained, so it must be read again. Part (a) in Figure 13.3 shows the method of reading the time without using interrupts; part (b) in Figure 13.3 shows the method using carry interrupts. To keep programming simple, method (a) should normally be used.

### 13.4.4 Alarm Function

Figure 13.4 shows how to use the alarm function.



**Figure 13.4 Using Alarm Function**

Alarms can be generated using seconds, minutes, hours, day of the week, day, month, year, or any combination of these. Set the ENB bit in the register on which the alarm is placed to 1, and then set the alarm time in the lower bits. Clear the ENB bit in the register on which the alarm is not placed to 0.

When the clock and alarm times match, 1 is set in the AF bit in RCR1. Alarm detection can be checked by reading this bit, but normally it is done by interrupt. If 1 is set in the AIE bit in RCR1, an interrupt is generated when an alarm occurs.

The alarm flag is set when the clock and alarm times match. However, the alarm flag can be cleared by writing 0.

## 13.5 Usage Notes

### 13.5.1 Register Writing during Count Operation

The following registers cannot be written to during count operation (while the START bit = 1 in RCR2).

RSECCNT, RMINCNT, RHRCNT, RDAYCNT, RWKCNT, RMONCNT, RYRCONT

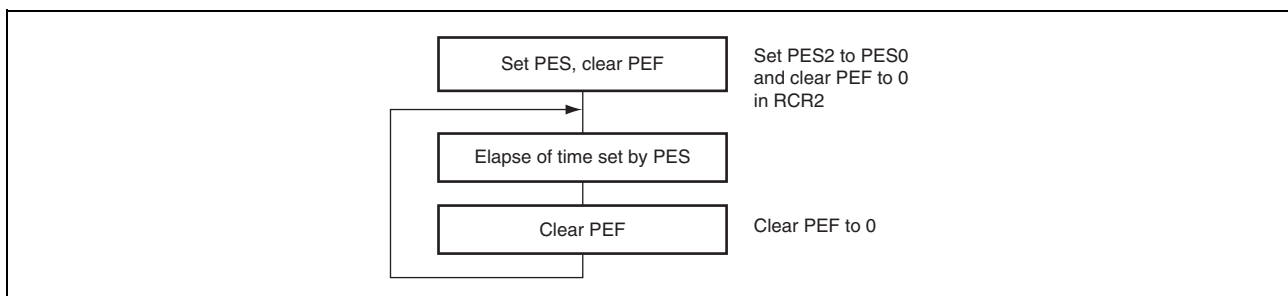
Count operation must be stopped before writing to any of the above registers.

### 13.5.2 Use of Realtime Clock Periodic Interrupts

The method of using the periodic interrupt function is shown in Figure 13.5.

A periodic interrupt can be generated periodically at the interval set by bits PES2 to PES0 in RCR2. When the time set by bits PES2 to PES0 has elapsed, the PEF is set to 1.

The PEF is cleared to 0 upon periodic interrupt generation or when bits PES2 to PES0 are set. Periodic interrupt generation can be confirmed by reading this bit, but normally the interrupt function is used.



**Figure 13.5 Using Periodic Interrupt Function**

### 13.5.3 Transition to Standby Mode after Setting Register

When a transition to standby mode is made after registers in this module are set, sometimes counting is not performed correctly. After the registers are set, be sure to perform one dummy read of the registers before making a transition to standby mode.

### 13.5.4 Usage Notes when Writing to and Reading the Register

- After writing to the RCR2 register or any of the counters such as the second counter, dummy-read the register twice before reading the actual value. The register contents before the write are returned by the two dummy reads, and the third read returns the register contents reflecting the write.
- Registers other than the above can be read immediately after a write and the written value is reflected.

## 14. Serial Communication Interface with FIFO

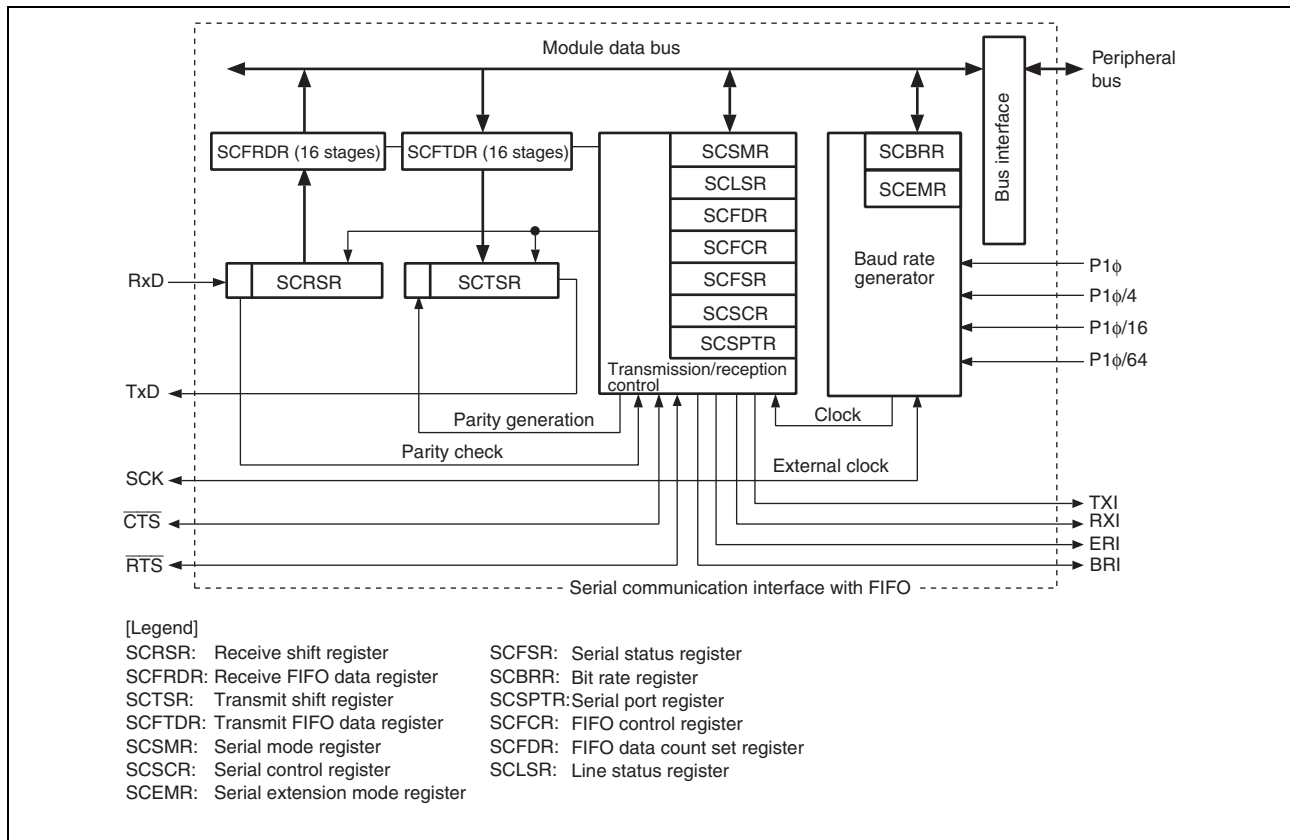
This LSI has an eight-channel serial communication interface with FIFO that supports both asynchronous and clock synchronous serial communication. It also has 16-stage FIFO registers for both transmission and reception independently for each channel that enable this LSI to perform efficient high-speed continuous communication.

### 14.1 Features

- Asynchronous serial communication:
  - Serial data communication is performed by start-stop in character units. This module can communicate with a universal asynchronous receiver/transmitter (UART), an asynchronous communication interface adapter (ACIA), or any other communications chip that employs a standard asynchronous serial system. There are eight selectable serial data communication formats.
  - Data length: 7 or 8 bits
  - Stop bit length: 1 or 2 bits
  - Parity: Even, odd, or none
  - Receive error detection: Parity, framing, and overrun errors
  - Break detection: Break is detected when a framing error is followed by at least one frame at the space 0 level (low level). It is also detected by reading the RxD pin level directly from the serial port register when a framing error occurs.
- Clock synchronous serial communication:
  - Serial data communication is synchronized with a clock signal. This module can communicate with other chips having a clock synchronous communication function. There is one serial data communication format.
  - Data length: 8 bits
  - Receive error detection: Overrun errors
- Full duplex communication: The transmitting and receiving sections are independent, so this module can transmit and receive simultaneously. Both sections use 16-stage FIFO buffering, so high-speed continuous data transfer is possible in both the transmit and receive directions.
- On-chip baud rate generator with selectable bit rates
- Internal or external transmit/receive clock source: From either baud rate generator (internal) or SCK pin (external)
- Four types of interrupts: Transmit-FIFO-data-empty interrupt, break interrupt, receive-FIFO-data-full interrupt, and receive-error interrupt are requested independently.
- When this module is not in use, it can be stopped by halting the clock supplied to it, saving power.
- In asynchronous mode, on-chip modem control functions ( $\overline{\text{RTS}}$  and  $\overline{\text{CTS}}$ ) (only channels 1, 5, and 7).
- The quantity of data in the transmit and receive FIFO data registers and the number of receive errors of the receive data in the receive FIFO data register can be ascertained.
- A time-out error (DR) can be detected when receiving in asynchronous mode.
- In asynchronous mode, the base clock frequency can be either 16 or 8 times the bit rate.
- When an internal clock is selected as a clock source and the SCK pin is used as an input pin in asynchronous mode, either normal mode or double-speed mode can be selected for the baud rate generator.



Figure 14.1 shows a block diagram. However, certain channels do not have the  $\overline{\text{CTS}}$  and  $\overline{\text{RTS}}$  pins.



**Figure 14.1 Block Diagram**

## 14.2 Input/Output Pins

Table 14.1 shows the pin configuration.

**Table 14.1 Pin Configuration**

Channel	Pin Name	Symbol	I/O	Function
0 to 7	Serial clock pins	SCK0 to SCK7	I/O	Clock I/O
	Receive data pins	RxD0 to RxD7	Input	Receive data input
	Transmit data pins	TxD0 to TxD7	Output	Transmit data output
1, 5, 7	Request to send pin	$\overline{\text{RTS1}}, \overline{\text{RTS5}}, \overline{\text{RTS7}}$	I/O	Request to send
	Clear to send pin	$\overline{\text{CTS1}}, \overline{\text{CTS5}}, \overline{\text{CTS7}}$	I/O	Clear to send

### 14.3 Register Descriptions

This module has the following registers.

**Table 14.2 Register Configuration**

Channel	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
0	Serial mode register_0	SCSMR_0	R/W	H'0000	H'E8007000	16
	Bit rate register_0	SCBRR_0	R/W	H'FF	H'E8007004	8
	Serial control register_0	SCSCR_0	R/W	H'0000	H'E8007008	16
	Transmit FIFO data register_0	SCFTDR_0	W	Undefined	H'E800700C	8
	Serial status register_0	SCFSR_0	R/(W)*1	H'0060	H'E8007010	16
	Receive FIFO data register_0	SCFRDR_0	R	Undefined	H'E8007014	8
	FIFO control register_0	SCFCR_0	R/W	H'0000	H'E8007018	16
	FIFO data count set register_0	SCFDR_0	R	H'0000	H'E800701C	16
	Serial port register_0	SCSPTR_0	R/W	H'0050	H'E8007020	16
	Line status register_0	SCLSR_0	R/(W)*2	H'0000	H'E8007024	16
	Serial extension mode register_0	SCEMR_0	R/W	H'0000	H'E8007028	16
1	Serial mode register_1	SCSMR_1	R/W	H'0000	H'E8007800	16
	Bit rate register_1	SCBRR_1	R/W	H'FF	H'E8007804	8
	Serial control register_1	SCSCR_1	R/W	H'0000	H'E8007808	16
	Transmit FIFO data register_1	SCFTDR_1	W	Undefined	H'E800780C	8
	Serial status register_1	SCFSR_1	R/(W)*1	H'0060	H'E8007810	16
	Receive FIFO data register_1	SCFRDR_1	R	Undefined	H'E8007814	8
	FIFO control register_1	SCFCR_1	R/W	H'0000	H'E8007818	16
	FIFO data count set register_1	SCFDR_1	R	H'0000	H'E800781C	16
	Serial port register_1	SCSPTR_1	R/W	H'0050	H'E8007820	16
	Line status register_1	SCLSR_1	R/(W)*2	H'0000	H'E8007824	16
	Serial extension mode register_1	SCEMR_1	R/W	H'0000	H'E8007828	16
2	Serial mode register_2	SCSMR_2	R/W	H'0000	H'E8008000	16
	Bit rate register_2	SCBRR_2	R/W	H'FF	H'E8008004	8
	Serial control register_2	SCSCR_2	R/W	H'0000	H'E8008008	16
	Transmit FIFO data register_2	SCFTDR_2	W	Undefined	H'E800800C	8
	Serial status register_2	SCFSR_2	R/(W)*1	H'0060	H'E8008010	16
	Receive FIFO data register_2	SCFRDR_2	R	Undefined	H'E8008014	8
	FIFO control register_2	SCFCR_2	R/W	H'0000	H'E8008018	16
	FIFO data count set register_2	SCFDR_2	R	H'0000	H'E800801C	16
	Serial port register_2	SCSPTR_2	R/W	H'0050	H'E8008020	16
	Line status register_2	SCLSR_2	R/(W)*2	H'0000	H'E8008024	16
	Serial extension mode register_2	SCEMR_2	R/W	H'0000	H'E8008028	16
3	Serial mode register_3	SCSMR_3	R/W	H'0000	H'E8008800	16
	Bit rate register_3	SCBRR_3	R/W	H'FF	H'E8008804	8
	Serial control register_3	SCSCR_3	R/W	H'0000	H'E8008808	16
	Transmit FIFO data register_3	SCFTDR_3	W	Undefined	H'E800880C	8
	Serial status register_3	SCFSR_3	R/(W)*1	H'0060	H'E8008810	16
	Receive FIFO data register_3	SCFRDR_3	R	Undefined	H'E8008814	8
	FIFO control register_3	SCFCR_3	R/W	H'0000	H'E8008818	16
	FIFO data count set register_3	SCFDR_3	R	H'0000	H'E800881C	16
	Serial port register_3	SCSPTR_3	R/W	H'0050	H'E8008820	16
	Line status register_3	SCLSR_3	R/(W)*2	H'0000	H'E8008824	16
	Serial extension mode register_3	SCEMR_3	R/W	H'0000	H'E8008828	16

Channel	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
4	Serial mode register_4	SCSMR_4	R/W	H'0000	H'E8009000	16
	Bit rate register_4	SCBRR_4	R/W	H'FF	H'E8009004	8
	Serial control register_4	SCSCR_4	R/W	H'0000	H'E8009008	16
	Transmit FIFO data register_4	SCFTDR_4	W	Undefined	H'E800900C	8
	Serial status register_4	SCFSR_4	R/(W)*1	H'0060	H'E8009010	16
	Receive FIFO data register_4	SCFRDR_4	R	Undefined	H'E8009014	8
	FIFO control register_4	SCFCR_4	R/W	H'0000	H'E8009018	16
	FIFO data count set register_4	SCFDR_4	R	H'0000	H'E800901C	16
	Serial port register_4	SCSPTR_4	R/W	H'0050	H'E8009020	16
	Line status register_4	SCLSR_4	R/(W)*2	H'0000	H'E8009024	16
	Serial extension mode register_4	SCEMR_4	R/W	H'0000	H'E8009028	16
5	Serial mode register_5	SCSMR_5	R/W	H'0000	H'E8009800	16
	Bit rate register_5	SCBRR_5	R/W	H'FF	H'E8009804	8
	Serial control register_5	SCSCR_5	R/W	H'0000	H'E8009808	16
	Transmit FIFO data register_5	SCFTDR_5	W	Undefined	H'E800980C	8
	Serial status register_5	SCFSR_5	R/(W)*1	H'0060	H'E8009810	16
	Receive FIFO data register_5	SCFRDR_5	R	Undefined	H'E8009814	8
	FIFO control register_5	SCFCR_5	R/W	H'0000	H'E8009818	16
	FIFO data count set register_5	SCFDR_5	R	H'0000	H'E800981C	16
	Serial port register_5	SCSPTR_5	R/W	H'0050	H'E8009820	16
	Line status register_5	SCLSR_5	R/(W)*2	H'0000	H'E8009824	16
	Serial extension mode register_5	SCEMR_5	R/W	H'0000	H'E8009828	16
6	Serial mode register_6	SCSMR_6	R/W	H'0000	H'E800A000	16
	Bit rate register_6	SCBRR_6	R/W	H'FF	H'E800A004	8
	Serial control register_6	SCSCR_6	R/W	H'0000	H'E800A008	16
	Transmit FIFO data register_6	SCFTDR_6	W	Undefined	H'E800A00C	8
	Serial status register_6	SCFSR_6	R/(W)*1	H'0060	H'E800A010	16
	Receive FIFO data register_6	SCFRDR_6	R	Undefined	H'E800A014	8
	FIFO control register_6	SCFCR_6	R/W	H'0000	H'E800A018	16
	FIFO data count set register_6	SCFDR_6	R	H'0000	H'E800A01C	16
	Serial port register_6	SCSPTR_6	R/W	H'0050	H'E800A020	16
	Line status register_6	SCLSR_6	R/(W)*2	H'0000	H'E800A024	16
	Serial extension mode register_6	SCEMR_6	R/W	H'0000	H'E800A028	16
7	Serial mode register_7	SCSMR_7	R/W	H'0000	H'E800A800	16
	Bit rate register_7	SCBRR_7	R/W	H'FF	H'E800A804	8
	Serial control register_7	SCSCR_7	R/W	H'0000	H'E800A808	16
	Transmit FIFO data register_7	SCFTDR_7	W	Undefined	H'E800A80C	8
	Serial status register_7	SCFSR_7	R/(W)*1	H'0060	H'E800A810	16
	Receive FIFO data register_7	SCFRDR_7	R	Undefined	H'E800A814	8
	FIFO control register_7	SCFCR_7	R/W	H'0000	H'E800A818	16
	FIFO data count set register_7	SCFDR_7	R	H'0000	H'E800A81C	16
	Serial port register_7	SCSPTR_7	R/W	H'0050	H'E800A820	16
	Line status register_7	SCLSR_7	R/(W)*2	H'0000	H'E800A824	16
	Serial extension mode register_7	SCEMR_7	R/W	H'0000	H'E800A828	16

Note 1. Only 0 can be written to clear the flag. Bits 15 to 8, 3, and 2 are read-only bits that cannot be modified.

Note 2. Only 0 can be written to clear the flag. Bits 15 to 1 are read-only bits that cannot be modified.

### 14.3.1 Receive Shift Register (SCRSR)

SCRSR receives serial data. Data input at the RxD pin is loaded into SCRSR in the order received, LSB (bit 0) first, being converted to parallel form. When one byte has been received, it is automatically transferred to the receive FIFO data register (SCFRDR).

The CPU cannot read from or write to SCRSR directly.

Bit:	7	6	5	4	3	2	1	0
Initial value:	-	-	-	-	-	-	-	-
R/W:	-	-	-	-	-	-	-	-

### 14.3.2 Receive FIFO Data Register (SCFRDR)

SCFRDR is a 16-stage FIFO register that stores serial receive data. The reception of one byte of serial data is complete when the received data is moved from the receive shift register (SCRSR) to SCFRDR for storage. Continuous reception is possible until 16 bytes are stored. The CPU can read but not write to SCFRDR. If data is read when there is no receive data in the SCFRDR, the value is undefined.

When SCFRDR is full of receive data, subsequent serial data is lost.

Bit:	7	6	5	4	3	2	1	0
Initial value:	-	-	-	-	-	-	-	-
R/W:	R	R	R	R	R	R	R	R

### 14.3.3 Transmit Shift Register (SCTSR)

SCTSR transmits serial data. Transmit data is loaded from the transmit FIFO data register (SCFTDR) into SCTSR, then the data is transmitted serially from the TxD pin, LSB (bit 0) first. After one data byte has been transmitted, the next transmit data is automatically loaded from SCFTDR into SCTSR and transmission is started again.

The CPU cannot read from or write to SCTSR directly.

Bit:	7	6	5	4	3	2	1	0									
	<table border="1" style="width: 100%; height: 15px;"> <tr> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> </tr> </table>																
Initial value:	-	-	-	-	-	-	-	-									
R/W:	-	-	-	-	-	-	-	-									

### 14.3.4 Transmit FIFO Data Register (SCFTDR)

SCFTDR is a 16-stage FIFO register that stores data for serial transmission. When the transmit shift register (SCTSR) empty is detected, transmit data written in the SCFTDR is moved to SCTSR and serial transmission is started. Continuous serial transmission is performed until there is no transmit data left in SCFTDR. The CPU can write to SCFTDR at all times.

When SCFTDR is full of transmit data (16 bytes), no more data can be written. If writing of next data is attempted, the data is ignored.

Bit:	7	6	5	4	3	2	1	0									
	<table border="1" style="width: 100%; height: 15px;"> <tr> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> <td style="width: 12.5%;"></td> </tr> </table>																
Initial value:	-	-	-	-	-	-	-	-									
R/W:	W	W	W	W	W	W	W	W									

### 14.3.5 Serial Mode Register (SCSMR)

SCSMR specifies the serial communication format and selects the clock source for the baud rate generator.

The CPU can always read from and write to SCSMR.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	C/ $\bar{A}$	CHR	PE	O/ $\bar{E}$	STOP	-	CKS[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	C/ $\bar{A}$	0	R/W	Communication Mode Selects operating mode from asynchronous and clock synchronous modes. 0: Asynchronous mode 1: Clock synchronous mode
6	CHR	0	R/W	Character Length Selects 7-bit or 8-bit data length in asynchronous mode. In the clock synchronous mode, the data length is always 8 bits, regardless of the CHR setting. 0: 8-bit data 1: 7-bit data* Note: * When 7-bit data is selected, the MSB (bit 7) of the transmit FIFO data register is not transmitted.
5	PE	0	R/W	Parity Enable Selects whether to add a parity bit to transmit data and to check the parity of receive data, in asynchronous mode. In clock synchronous mode, a parity bit is neither added nor checked, regardless of the PE setting. 0: Parity bit not added or checked 1: Parity bit added and checked* Note: * When PE is set to 1, an even or odd parity bit is added to transmit data, depending on the parity mode (O/ $\bar{E}$ ) setting. Receive data parity is checked according to the even/odd (O/ $\bar{E}$ ) mode setting.
4	O/ $\bar{E}$	0	R/W	Parity Mode Selects even or odd parity when parity bits are added and checked. The O/ $\bar{E}$ setting is used only in asynchronous mode and only when the parity enable bit (PE) is set to 1 to enable parity addition and checking. The O/ $\bar{E}$ setting is ignored in clock synchronous mode or in asynchronous mode when parity addition and checking is disabled. 0: Even parity*1 1: Odd parity*2 Note: 1. If even parity is selected, the parity bit is added to transmit data to make an even number of 1s in the transmitted character and parity bit combined. Receive data is checked to see if it has an even number of 1s in the received character and parity bit combined. 2. If odd parity is selected, the parity bit is added to transmit data to make an odd number of 1s in the transmitted character and parity bit combined. Receive data is checked to see if it has an odd number of 1s in the received character and parity bit combined.

Bit	Bit Name	Initial Value	R/W	Description
3	STOP	0	R/W	<p>Stop Bit Length</p> <p>Selects one or two bits as the stop bit length in asynchronous mode. This setting is used only in asynchronous mode. It is ignored in clock synchronous mode because no stop bits are added.</p> <p>When receiving, only the first stop bit is checked, regardless of the STOP bit setting. If the second stop bit is 1, it is treated as a stop bit, but if the second stop bit is 0, it is treated as the start bit of the next incoming character.</p> <p>0: One stop bit When transmitting, a single 1-bit is added at the end of each transmitted character.</p> <p>1: Two stop bits When transmitting, two 1 bits are added at the end of each transmitted character.</p>
2	—	0	R	<p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p>
1, 0	CKS[1:0]	00	R/W	<p>Clock Select</p> <p>Select the internal clock source of the on-chip baud rate generator. For further information on the clock source, bit rate register settings, and baud rate, see section 14.3.8, Bit Rate Register (SCBRR).</p> <p>00: P1<math>\phi</math> 01: P1<math>\phi</math>/4 10: P1<math>\phi</math>/16 11: P1<math>\phi</math>/64 Note: P1<math>\phi</math>: Peripheral clock</p>



### 14.3.6 Serial Control Register (SCSCR)

SCSCR enables/disables the transmitter/receiver operation and interrupt requests, and selects the transmit/receive clock source. The CPU can always read and write to SCSCR.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	TIE	RIE	TE	RE	REIE	-	CKE[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	TIE	0	R/W	Transmit Interrupt Enable Enables or disables the transmit-FIFO-data-empty interrupt (TXI) requested when the serial transmit data is transferred from the transmit FIFO data register (SCFTDR) to the transmit shift register (SCTSR), the quantity of data in the transmit FIFO register becomes less than the specified number of transmission triggers, and then the TDFE flag in the serial status register (SCFCSR) is set to 1. 0: Transmit-FIFO-data-empty interrupt request (TXI) is disabled 1: Transmit-FIFO-data-empty interrupt request (TXI) is enabled* Note: * The TXI interrupt request can be cleared by writing a greater quantity of transmit data than the specified transmission trigger number to SCFTDR and clearing TDFE to 0 after reading 1 from TDFE, or by clearing TIE to 0.
6	RIE	0	R/W	Receive Interrupt Enable Enables or disables the receive FIFO data full (RXI) interrupts requested when the RDF flag or DR flag in serial status register (SCFCSR) is set to 1, receive-error (ERI) interrupts requested when the ER flag in SCFCSR is set to 1, and break (BRI) interrupts requested when the BRK flag in SCFCSR or the ORER flag in line status register (SCLSR) is set to 1. 0: Receive FIFO data full interrupt (RXI), receive-error interrupt (ERI), and break interrupt (BRI) requests are disabled 1: Receive FIFO data full interrupt (RXI), receive-error interrupt (ERI), and break interrupt (BRI) requests are enabled* Note: * RXI interrupt requests can be cleared by reading the DR or RDF flag after it has been set to 1, then clearing the flag to 0, or by clearing RIE to 0. ERI or BRI interrupt requests can be cleared by reading the ER, BRK or ORER flag after it has been set to 1, then clearing the flag to 0, or by clearing RIE and REIE to 0.
5	TE	0	R/W	Transmit Enable Enables or disables serial transmission. 0: Serial transmission disabled 1: Serial transmission enabled* Note: * When this bit is set to 1, serial transmission starts after writing of transmit data into SCFTDR. Be sure to select the transmit format in SCSMR and SCFCR and reset the transmit FIFO before setting TE to 1.
4	RE	0	R/W	Receive Enable Enables or disables serial reception. 0: Serial reception disabled*1 1: Serial reception enabled*2 Notes: 1. Clearing RE to 0 does not affect the receive flags (DR, ER, BRK, RDF, FER, PER, and ORER). These flags retain their previous values. 2. When this bit is set to 1, serial reception starts when a start bit is detected in asynchronous mode, or synchronous clock is detected in clock synchronous mode. Be sure to select the receive format in SCSMR and SCFCR and reset the receive FIFO before setting RE to 1.

Bit	Bit Name	Initial Value	R/W	Description
3	REIE	0	R/W	<p>Receive Error Interrupt Enable</p> <p>Enables or disables the receive-error (ERI) interrupts and break (BRI) interrupts. The setting of REIE bit is valid only when RIE bit is set to 0.</p> <p>0: Receive-error interrupt (ERI) and break interrupt (BRI) requests are disabled</p> <p>1: Receive-error interrupt (ERI) and break interrupt (BRI) requests are enabled*</p> <p>Note: * ERI or BRI interrupt requests can be cleared by reading the ER, BRK or ORER flag after it has been set to 1, then clearing the flag to 0, or by clearing RIE and REIE to 0. Even if RIE is set to 0, when REIE is set to 1, ERI or BRI interrupt requests are enabled.</p>
2	—	0	R	<p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p>
1, 0	CKE[1:0]	00	R/W	<p>Clock Enable</p> <p>Select the clock source and enable or disable clock output from the SCK pin. Depending on CKE[1:0], the SCK pin can be used for serial clock output or serial clock input. If synchronous clock output is set in clock synchronous mode, set the C/A bit in SCSMR to 1, and then set CKE[1:0].</p> <ul style="list-style-type: none"> <li>• Asynchronous mode</li> <li>00: Internal clock, SCK pin used for input (input signal is ignored)</li> <li>01: Internal clock, SCK pin used for clock output (The output clock frequency is either 16 or 8 times the bit rate.)</li> <li>10: External clock, SCK pin used for clock input (The input clock frequency is either 16 or 8 times the bit rate.)</li> <li>11: Setting prohibited</li> <li>• Clock synchronous mode</li> <li>00: Internal clock, SCK pin used for synchronous clock output</li> <li>01: Internal clock, SCK pin used for synchronous clock output</li> <li>10: External clock, SCK pin used for synchronous clock input</li> <li>11: Setting prohibited</li> </ul>

### 14.3.7 Serial Status Register (SCFSR)

SCFSR is a 16-bit register. The upper 8 bits indicate the number of receive errors in the receive FIFO data register, and the lower 8 bits indicate the status flag indicating operating state.

The CPU can always read from and write to SCFSR, but cannot write 1 to the status flags (ER, TEND, TDFE, BRK, RDF, and DR). These flags can be cleared to 0 only if they have first been read (after being set to 1). The PER flag (bits 15 to 12 and bit 2) and the FER flag (bits 11 to 8 and bit 3) are read-only bits that cannot be written.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PER[3:0]				FER[3:0]				ER	TEND	TDFE	BRK	FER	PER	RDF	DR
Initial value:	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R	R	R/(W)*	R/(W)*

Note: \* Only 0 can be written to clear the flag after 1 is read.

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	PER[3:0]	0000	R	<p>Number of Parity Errors</p> <p>Indicate the quantity of data including a parity error in the receive data stored in the receive FIFO data register (SCFRDR). The value indicated by bits 15 to 12 after the ER bit in SCFSR is set, represents the number of parity errors in SCFRDR. When parity errors have occurred in all 16-byte receive data in SCFRDR, PER[3:0] shows 0000.</p>
11 to 8	FER[3:0]	0000	R	<p>Number of Framing Errors</p> <p>Indicate the quantity of data including a framing error in the receive data stored in SCFRDR. The value indicated by bits 11 to 8 after the ER bit in SCFSR is set, represents the number of framing errors in SCFRDR. When framing errors have occurred in all 16-byte receive data in SCFRDR, FER[3:0] shows 0000.</p>
7	ER	0	R/(W)*	<p>Receive Error</p> <p>Indicates the occurrence of a framing error, or of a parity error when receiving data that includes parity.*1</p> <p>0: Receiving is in progress or has ended normally [Clearing conditions]</p> <ul style="list-style-type: none"> <li>ER is cleared to 0 by a power-on reset</li> <li>ER is cleared to 0 when 0 is written to after 1 is read from ER</li> </ul> <p>1: A framing error or parity error has occurred. [Setting conditions]</p> <ul style="list-style-type: none"> <li>ER is set to 1 when the stop bit is 0 after checking whether or not the last stop bit of the received data is 1 at the end of one data receive operation*2</li> <li>ER is set to 1 when the total number of 1s in the receive data plus parity bit does not match the even/odd parity specified by the O/E bit in SCSMR</li> </ul> <p>Note: 1. Clearing the RE bit to 0 in SCSCR does not affect the ER bit, which retains its previous value. Even if a receive error occurs, the receive data is transferred to SCFRDR and the receive operation is continued. Whether or not the data read from SCFRDR includes a receive error can be detected by the FER and PER bits in SCFSR.</p> <p>2. In two stop bits mode, only the first stop bit is checked; the second stop bit is not checked.</p>

Bit	Bit Name	Initial Value	R/W	Description
6	TEND	1	R/(W)*	<p>Transmit End</p> <p>Indicates that when the last bit of a serial character was transmitted, SCFTDR did not contain valid data, so transmission has ended.</p> <p>0: Transmission is in progress</p> <p>[Clearing condition]</p> <ul style="list-style-type: none"> <li>TEND is cleared to 0 when 0 is written after 1 is read from TEND after transmit data is written in SCFTDR*1</li> </ul> <p>1: End of transmission</p> <p>[Setting conditions]</p> <ul style="list-style-type: none"> <li>TEND is set to 1 by a power-on reset</li> <li>TEND is set to 1 when TE is cleared to 0 in the serial control register (SCSCR)</li> <li>TEND is set to 1 when SCFTDR does not contain transmit data when the last bit of a one-byte serial character is transmitted</li> </ul> <p>Note: 1. Do not use this bit as a transmit end flag when the direct memory access controller writes data to SCFTDR due to a TXI interrupt request.</p>
5	TDFE	1	R/(W)*	<p>Transmit FIFO Data Empty</p> <p>Indicates that data has been transferred from the transmit FIFO data register (SCFTDR) to the transmit shift register (SCTSR), the quantity of data in SCFTDR has become less than the transmission trigger number specified by the TTRG[1:0] bits in the FIFO control register (SCFCR), and writing of transmit data to SCFTDR is enabled.</p> <p>0: The quantity of transmit data written to SCFTDR is greater than the specified transmission trigger number</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> <li>TDFE is cleared to 0 when data exceeding the specified transmission trigger number is written to SCFTDR after 1 is read from TDFE and then 0 is written to TDFE</li> <li>TDFE is cleared to 0 when direct memory access controller is activated by transmit FIFO data empty interrupt (TXI) and data exceeding the specified transmission trigger number is written to SCFTDR</li> </ul> <p>1: The quantity of transmit data in SCFTDR is less than or equal to the specified transmission trigger number*1</p> <p>[Setting conditions]</p> <ul style="list-style-type: none"> <li>TDFE is set to 1 by a power-on reset</li> <li>TDFE is set to 1 when the quantity of transmit data in SCFTDR becomes less than or equal to the specified transmission trigger number as a result of transmission</li> </ul> <p>Note: 1. Since SCFTDR is a 16-byte FIFO register, the maximum quantity of data that can be written when TDFE is 1 is "16 minus the specified transmission trigger number". If an attempt is made to write additional data, the data is ignored. The quantity of data in SCFTDR is indicated by the upper 8 bits of SCFDR.</p>
4	BRK	0	R/(W)*	<p>Break Detection</p> <p>Indicates that a break signal has been detected in receive data.</p> <p>0: No break signal received</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> <li>BRK is cleared to 0 by a power-on reset</li> <li>BRK is cleared to 0 when software reads BRK after it has been set to 1, then writes 0 to BRK</li> </ul> <p>1: Break signal received*1</p> <p>[Setting condition]</p> <ul style="list-style-type: none"> <li>BRK is set to 1 when data including a framing error is received, followed by at least one frame at the space 0 level (low level)</li> </ul> <p>Note: 1. When a break is detected, transfer of the receive data (H'00) to SCFRDR stops after detection. When the break ends and the receive signal becomes mark 1, the transfer of receive data resumes.</p>

Bit	Bit Name	Initial Value	R/W	Description
3	FER	0	R	<p>Framing Error Indication</p> <p>Indicates a framing error in the data read from the receive FIFO data register (SCFRDR) in asynchronous mode.</p> <p>0: No receive framing error occurred in the next data read from SCFRDR [Clearing conditions]</p> <ul style="list-style-type: none"> <li>• FER is cleared to 0 by a power-on reset</li> <li>• FER is cleared to 0 when no framing error is present in the next data read from SCFRDR</li> </ul> <p>1: A receive framing error occurred in the next data read from SCFRDR [Setting condition]</p> <ul style="list-style-type: none"> <li>• FER is set to 1 when a framing error is present in the next data read from SCFRDR</li> </ul>
2	PER	0	R	<p>Parity Error Indication</p> <p>Indicates a parity error in the data read from the receive FIFO data register (SCFRDR) in asynchronous mode.</p> <p>0: No receive parity error occurred in the next data read from SCFRDR [Clearing conditions]</p> <ul style="list-style-type: none"> <li>• PER is cleared to 0 by a power-on reset</li> <li>• PER is cleared to 0 when no parity error is present in the next data read from SCFRDR</li> </ul> <p>1: A receive parity error occurred in the next data read from SCFRDR [Setting condition]</p> <ul style="list-style-type: none"> <li>• PER is set to 1 when a parity error is present in the next data read from SCFRDR</li> </ul>
1	RDF	0	R/(W)*	<p>Receive FIFO Data Full</p> <p>Indicates that receive data has been transferred to the receive FIFO data register (SCFRDR), and the quantity of data in SCFRDR has become more than the receive trigger number specified by the RTRG[1:0] bits in the FIFO control register (SCFCR).</p> <p>0: The quantity of transmit data written to SCFRDR is less than the specified receive trigger number [Clearing conditions]</p> <ul style="list-style-type: none"> <li>• RDF is cleared to 0 by a power-on reset</li> <li>• RDF is cleared to 0 when the SCFRDR is read until the quantity of receive data in SCFRDR becomes less than the specified receive trigger number after 1 is read from RDF, and then 0 is written</li> <li>• RDF is cleared to 0 when the direct memory access controller is activated by receive FIFO data full interrupt (RXI) and SCFRDR is read until the quantity of receive data in it becomes less than the specified receive trigger number</li> </ul> <p>1: The quantity of receive data in SCFRDR is more than the specified receive trigger number [Setting condition]</p> <ul style="list-style-type: none"> <li>• RDF is set to 1 when a quantity of receive data more than the specified receive trigger number is stored in SCFRDR*1</li> </ul> <p>Note: 1. As SCFRDR is a 16-byte FIFO register, the maximum quantity of data that can be read when RDF is 1 is the specified receive trigger number. If an attempt is made to read after all the data in SCFRDR has been read, the data is undefined. The quantity of receive data in SCFRDR is indicated by the lower 8 bits of SCFCR.</p>

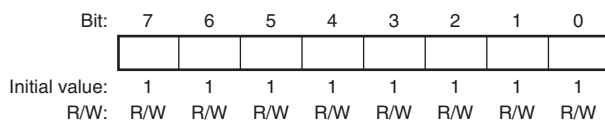
Bit	Bit Name	Initial Value	R/W	Description
0	DR	0	R/(W)*	<p>Receive Data Ready</p> <p>Indicates that the quantity of data in the receive FIFO data register (SCFRDR) is less than the specified receive trigger number, and that the next data has not yet been received after the elapse of 15 ETU from the last stop bit in asynchronous mode. In clock synchronous mode, this bit is not set to 1.</p> <p>0: Receiving is in progress, or no receive data remains in SCFRDR after receiving ended normally</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> <li>• DR is cleared to 0 by a power-on reset</li> <li>• DR is cleared to 0 when all receive data are read from SCFRDR after 1 is read from DR, and then 0 is written.</li> <li>• DR is cleared to 0 when all receive data are read from SCFRDR after the direct memory access controller is activated by receive FIFO data full interrupt (RXI).</li> </ul> <p>1: Next receive data has not been received</p> <p>[Setting condition]</p> <ul style="list-style-type: none"> <li>• DR is set to 1 when SCFRDR contains less data than the specified receive trigger number, and the next data has not yet been received after the elapse of 15 ETU from the last stop bit.*1</li> </ul> <p>Note: 1. This is equivalent to 1.5 frames with the 8-bit, 1-stop-bit format. (ETU: elementary time unit)</p>

Note: \* Only 0 can be written to clear the flag after 1 is read.

### 14.3.8 Bit Rate Register (SCBRR)

SCBRR is an 8-bit register that is used with the CKS1 and CKS0 bits in the serial mode register (SCSMR) and the BGDM and ABCS bits in the serial extension mode register (SCEMR) to determine the serial transmit/receive bit rate.

The CPU can always read from and write to SCBRR. SCBRR is initialized to H'FF by a power-on reset. Each channel has independent baud rate generator control, so different values can be set in eight channels.



The SCBRR setting is calculated as follows:

- Asynchronous mode:

When baud rate generator operates in normal mode (when the BGDM bit of SCEMR is 0):

$$N = \frac{P1\phi}{64 \times 2^{2n-1} \times B} \times 10^6 - 1 \quad (\text{Operation on a base clock with a frequency of 16 times the bit rate})$$

$$N = \frac{P1\phi}{32 \times 2^{2n-1} \times B} \times 10^6 - 1 \quad (\text{Operation on a base clock with a frequency of 8 times the bit rate})$$

When baud rate generator operates in double speed mode (when the BGDM bit of SCEMR is 1):

$$N = \frac{P1\phi}{32 \times 2^{2n-1} \times B} \times 10^6 - 1 \quad (\text{Operation on a base clock with a frequency of 16 times the bit rate})$$

$$N = \frac{P1\phi}{16 \times 2^{2n-1} \times B} \times 10^6 - 1 \quad (\text{Operation on a base clock with a frequency of 8 times the bit rate})$$

- Clock synchronous mode:

$$N = \frac{P1\phi}{8 \times 2^{2n-1} \times B} \times 10^6 - 1$$

B: Bit rate (bits/s)

N: SCBRR setting for baud rate generator ( $0 \leq N \leq 255$ )  
(The setting must satisfy the electrical characteristics.)

P1φ: Operating frequency for peripheral modules (MHz)

n: Baud rate generator clock source ( $n = 0, 1, 2, 3$ ) (for the clock sources and values of n, see Table 14.3.)

**Table 14.3 SCSMR Settings**

n	Clock Source	SCSMR Settings	
		CKS[1]	CKS[0]
0	P1φ	0	0
1	P1φ/4	0	1
2	P1φ/16	1	0
3	P1φ/64	1	1

The bit rate error in asynchronous mode is given by the following formula:

When baud rate generator operates in normal mode (the BGDM bit of SCEMR is 0):

$$\text{Error (\%)} = \left\{ \frac{P1\phi \times 10^6}{(N + 1) \times B \times 64 \times 2^{2n-1}} - 1 \right\} \times 100 \quad \text{(Operation on a base clock with a frequency of 16 times the bit rate)}$$

$$\text{Error (\%)} = \left\{ \frac{P1\phi \times 10^6}{(N + 1) \times B \times 32 \times 2^{2n-1}} - 1 \right\} \times 100 \quad \text{(Operation on a base clock with a frequency of 8 times the bit rate)}$$

When baud rate generator operates in double speed mode (the BGDM bit of SCEMR is 1):

$$\text{Error (\%)} = \left\{ \frac{P1\phi \times 10^6}{(N + 1) \times B \times 32 \times 2^{2n-1}} - 1 \right\} \times 100 \quad \text{(Operation on a base clock with a frequency of 16 times the bit rate)}$$

$$\text{Error (\%)} = \left\{ \frac{P1\phi \times 10^6}{(N + 1) \times B \times 16 \times 2^{2n-1}} - 1 \right\} \times 100 \quad \text{(Operation on a base clock with a frequency of 8 times the bit rate)}$$



Table 14.4 lists the sample SCBRR settings in asynchronous mode in which a base clock frequency is 16 times the bit rate (the ABCS bit in SCEMR is 0) and the baud rate generator operates in normal mode (the BGDM bit in SCEMR is 0), and Table 14.5 lists the sample SCBRR settings in clock synchronous mode.

**Table 14.4 Bit Rates and SCBRR Settings (Asynchronous Mode, BGDM = 0, ABCS = 0)**

Bit Rate (bits/s)	P1 $\phi$ (MHz)					
	50			66.67		
	n	N	Error (%)	n	N	Error (%)
110	3	221	-0.02			
150	3	162	-0.15	3	216	0.01
300	3	80	0.47	3	108	-0.45
600	2	162	-0.15	2	216	0.01
1200	2	80	0.47	2	108	-0.45
2400	1	162	-0.15	1	216	0.01
4800	1	80	0.47	1	108	-0.45
9600	0	162	-0.15	0	216	0.01
19200	0	80	0.47	0	108	-0.45
31250	0	49	0.00	0	66	-0.50
38400	0	40	-0.76	0	53	0.47

Note: The error rate should be  $\leq 1\%$ .

[Legend]

Blank space: Setting impossible

**Table 14.5 Bit Rates and SCBRR Settings (Clock Synchronous Mode)**

Bit Rate (bits/s)	P1 $\phi$ (MHz)			
	50		66.67	
	n	N	n	N
500	—	—		
1000	3	194	—	—
2500	3	77	3	103
5000	2	155	2	207
10000	2	77	2	103
25000	1	124	1	166
50000	1	62	1	82
100000	0	124	0	166
250000	0	49	0	66
500000	0	24	—	—
1000000	—	—	—	—
2000000	—	—	—	—

[Legend]

Blank space: Setting impossible

—: Setting possible, but error occurs

Table 14.6 indicates the maximum bit rates in asynchronous mode when the baud rate generator is used. Table 14.7 lists the maximum bit rates in asynchronous mode when the external clock input is used. Table 14.8 lists the maximum bit rates in clock synchronous mode when the external clock input is used (when  $t_{Scyc} = 12t_{pcyc}^*$ ).

Note: \* Make sure that the electrical characteristics of this LSI and that of a connected LSI are satisfied.

**Table 14.6 Maximum Bit Rates for Various Frequencies with Baud Rate Generator (Asynchronous Mode)**

P1 $\phi$ (MHz)	Settings				Maximum Bit Rate (bits/s)
	BGDM	ABCS	n	N	
50	0	0	0	0	1562500
		1	0	0	3125000
	1	0	0	0	3125000
		1	0	0	6250000
66.67	0	0	0	0	2083333
		1	0	0	4166667
	1	0	0	0	4166667
		1	0	0	8333333

**Table 14.7 Maximum Bit Rates with External Clock Input (Asynchronous Mode)**

P1 $\phi$ (MHz)	External Input Clock (MHz)	Settings	Maximum Bit Rate (bits/s)
		ABCS	
50	12.5000	0	781250
		1	1562500
66.67	16.6667	0	1041667
		1	2083333

**Table 14.8 Maximum Bit Rates with External Clock Input (Clock Synchronous Mode,  $t_{Scyc} = 12 t_{pcyc}$ )**

P1 $\phi$ (MHz)	External Input Clock (MHz)	Maximum Bit Rate (bits/s)
50	4.1667	4166666.7
66.67	5.5556	5555555.5

### 14.3.9 FIFO Control Register (SCFCR)

SCFCR resets the quantity of data in the transmit and receive FIFO data registers, sets the trigger data quantity, and contains an enable bit for loop-back testing. SCFCR can always be read and written to by the CPU.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	RSTRG[2:0]			RTRG[1:0]		TTRG[1:0]		MCE	TFRST	RFRST	LOOP
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 8	RSTRG[2:0]	000	R/W	RTS Output Active Trigger When the quantity of receive data in receive FIFO data register (SCFRDR) becomes more than the number shown below, $\overline{\text{RTS}}$ signal is set to high. 000: 15 001: 1 010: 4 011: 6 100: 8 101: 10 110: 12 111: 14
7, 6	RTRG[1:0]	00	R/W	Receive FIFO Data Trigger Set the quantity of receive data which sets the receive data full (RDF) flag in the serial status register (SCFSR). The RDF flag is set to 1 when the quantity of receive data stored in the receive FIFO data register (SCFRDR) is increased more than the set trigger number shown below. <ul style="list-style-type: none"> <li>• Asynchronous mode</li> <li>00: 1</li> <li>01: 4</li> <li>10: 8</li> <li>11: 14</li> <li>• Clock synchronous mode</li> <li>00: 1</li> <li>01: 2</li> <li>10: 8</li> <li>11: 14</li> </ul> Note: In clock synchronous mode, to transfer the receive data using the direct memory access controller, set the receive trigger number to 1. If set to other than 1, CPU must read the receive data left in SCFRDR.
5, 4	TTRG[1:0]	00	R/W	Transmit FIFO Data Trigger Set the quantity of remaining transmit data which sets the transmit FIFO data register empty (TDFE) flag in the serial status register (SCFSR). The TDFE flag is set to 1 when the quantity of transmit data in the transmit FIFO data register (SCFTDR) becomes less than the set trigger number shown below. 00: 8 (8)* 01: 4 (12)* 10: 2 (14)* 11: 0 (16)* Note: * Values in parentheses mean the number of empty bytes in SCFTDR when the TDFE flag is set to 1.
3	MCE	0	R/W	Modem Control Enable Enables modem control signals $\overline{\text{CTS}}$ and $\overline{\text{RTS}}$ . For channels 0, 2 to 4, and 6 in clock synchronous mode, MCE bit should always be 0. 0: Modem signal disabled* 1: Modem signal enabled Note: * $\overline{\text{CTS}}$ is fixed at active 0 regardless of the input value, and $\overline{\text{RTS}}$ is also fixed at 0.

Bit	Bit Name	Initial Value	R/W	Description
2	TFRST	0	R/W	<p>Transmit FIFO Data Register Reset</p> <p>Disables the transmit data in the transmit FIFO data register and resets the register to the empty state.</p> <p>0: Reset operation disabled*</p> <p>1: Reset operation enabled</p> <p>Note: * Reset operation is executed by a power-on reset.</p>
1	RFRST	0	R/W	<p>Receive FIFO Data Register Reset</p> <p>Disables the receive data in the receive FIFO data register and resets the register to the empty state.</p> <p>0: Reset operation disabled*</p> <p>1: Reset operation enabled</p> <p>Note: * Reset operation is executed by a power-on reset.</p>
0	LOOP	0	R/W	<p>Loop-Back Test</p> <p>Internally connects the transmit output pin (TxD) and receive input pin (RxD) and internally connects the RTS pin and CTS pin and enables loop-back testing.</p> <p>0: Loop back test disabled</p> <p>1: Loop back test enabled</p>

### 14.3.10 FIFO Data Count Set Register (SCFDR)

SCFDR is a 16-bit register which indicates the quantity of data stored in the transmit FIFO data register (SCFTDR) and the receive FIFO data register (SCFRDR).

It indicates the quantity of transmit data in SCFTDR with the upper 8 bits, and the quantity of receive data in SCFRDR with the lower 8 bits. SCFDR can always be read by the CPU.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	T[4:0]				-	-	-	R[4:0]					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 8	T[4:0]	00000	R	T4 to T0 bits indicate the quantity of non-transmitted data stored in SCFTDR. H'00 means no transmit data, and H'10 means that all transmit data is stored in SCFTDR.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4 to 0	R[4:0]	00000	R	R4 to R0 bits indicate the quantity of receive data stored in SCFRDR. H'00 means no receive data, and H'10 means that all receive data is stored in SCFRDR.

### 14.3.11 Serial Port Register (SCSPTR)

SCSPTR controls input/output and data of pins multiplexed to the functions of this module. Bits 7 and 6 can control input/output data of  $\overline{\text{RTS}}$  pin. Bits 5 and 4 can control input/output data of  $\overline{\text{CTS}}$  pin. Bits 3 and 2 can control input/output data of SCK pin. Bits 1 and 0 can input data from RxD pin and output data to TxD pin, so they control break of serial transmitting/receiving.

The CPU can always read and write to SCSPTR.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	RTSIO	RTSDT	CTSIO	CTSDT	SCKIO	SCKDT	SPB2IO	SPB2DT
Initial value:	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	RTSIO	0	R/W	$\overline{\text{RTS}}$ Port Input/Output Specifies input or output for the serial port $\overline{\text{RTS}}$ pin. When the $\overline{\text{RTS}}$ pin is actually used as a port outputting the RTSDT bit value, the MCE bit in SCFCR should be cleared to 0. 0: RTSDT bit value not output to $\overline{\text{RTS}}$ pin 1: RTSDT bit value output to $\overline{\text{RTS}}$ pin
6	RTSDT	1	R/W	$\overline{\text{RTS}}$ Port Data Specifies the input/output data of the serial port $\overline{\text{RTS}}$ pin. Input/output is specified by the RTSIO bit. For output, the RTSDT bit value is output to the $\overline{\text{RTS}}$ pin. The $\overline{\text{RTS}}$ pin status is read from the RTSDT bit regardless of the RTSIO bit setting. However, $\overline{\text{RTS}}$ input/output must be set in the general purpose I/O ports. 0: Input/output data is low level 1: Input/output data is high level
5	CTSIO	0	R/W	$\overline{\text{CTS}}$ Port Input/Output Specifies input or output for the serial port $\overline{\text{CTS}}$ pin. When the $\overline{\text{CTS}}$ pin is actually used as a port outputting the CTSDT bit value, the MCE bit in SCFCR should be cleared to 0. 0: CTSDT bit value not output to $\overline{\text{CTS}}$ pin 1: CTSDT bit value output to $\overline{\text{CTS}}$ pin
4	CTSDT	1	R/W	$\overline{\text{CTS}}$ Port Data Specifies the input/output data of the serial port $\overline{\text{CTS}}$ pin. Input/output is specified by the CTSIO bit. For output, the CTSDT bit value is output to the $\overline{\text{CTS}}$ pin. The $\overline{\text{CTS}}$ pin status is read from the CTSDT bit regardless of the CTSIO bit setting. However, $\overline{\text{CTS}}$ input/output must be set in the general purpose I/O ports. 0: Input/output data is low level 1: Input/output data is high level
3	SCKIO	0	R/W	SCK Port Input/Output Specifies input or output for the serial port SCK pin. When the SCK pin is actually used as a port outputting the SCKDT bit value, the CKE[1:0] bits in SCSCR should be cleared to 0. 0: SCKDT bit value not output to SCK pin 1: SCKDT bit value output to SCK pin
2	SCKDT	0	R/W	SCK Port Data Specifies the input/output data of the serial port SCK pin. Input/output is specified by the SCKIO bit. For output, the SCKDT bit value is output to the SCK pin. The SCK pin status is read from the SCKDT bit regardless of the SCKIO bit setting. However, SCK input/output must be set in the general purpose I/O ports. 0: Input/output data is low level 1: Input/output data is high level

---

Bit	Bit Name	Initial Value	R/W	Description
1	SPB2IO	0	R/W	<b>Serial Port Break Input/Output</b> Specifies input or output for the serial port TxD pin. When the TxD pin is actually used as a port outputting the SPB2DT bit value, the TE bit in SCSCR should be cleared to 0. 0: SPB2DT bit value not output to TxD pin 1: SPB2DT bit value output to TxD pin
0	SPB2DT	0	R/W	<b>Serial Port Break Data</b> Specifies the input data of the RxD pin and the output data of the TxD pin used as serial ports. Input/output is specified by the SPB2IO bit. When the TxD pin is set to output, the SPB2DT bit value is output to the TxD pin. The RxD pin status is read from the SPB2DT bit regardless of the SPB2IO bit setting. However, RxD input and TxD output must be set in the general purpose I/O ports. 0: Input/output data is low level 1: Input/output data is high level

---

### 14.3.12 Line Status Register (SCLSR)

The CPU can always read or write to SCLSR, but cannot write 1 to the ORER flag. This flag can be cleared to 0 only if it has first been read (after being set to 1).

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	ORER
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/(W)*

Note: \* Only 0 can be written to clear the flag after 1 is read.

Bit	Bit Name	Initial Value	R/W	Description
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	ORER	0	R/(W)*	<p>Overrun Error</p> <p>Indicates the occurrence of an overrun error during reception.</p> <p>0: Receiving is in progress or has ended normally*1 [Clearing conditions]</p> <ul style="list-style-type: none"> <li>ORER is cleared to 0 by a power-on reset</li> <li>ORER is cleared to 0 when 0 is written after 1 is read from ORER.</li> </ul> <p>1: An overrun error has occurred during reception*2 [Setting condition]</p> <ul style="list-style-type: none"> <li>ORER is set to 1 when the next serial receiving is finished while the receive FIFO is full of 16-byte receive data.</li> </ul> <p>Note: 1. Clearing the RE bit to 0 in SCSCR does not affect the ORER bit, which retains its previous value. 2. The receive FIFO data register (SCFRDR) retains the data before an overrun error has occurred, and the next received data is discarded. When the ORER bit is set to 1, the next serial reception cannot be continued.</p>



### 14.3.13 Serial Extension Mode Register (SCEMR)

The CPU can always read from or write to SCEMR. Setting the BGDM bit in this register to 1 allows the baud rate generator in this module to operate in double-speed mode when asynchronous mode is selected (by setting the  $C/\bar{A}$  bit in SCSMR to 0) and an internal clock is selected as a clock source and the SCK pin is set as an input pin (by setting the CKE[1:0] bits in SCSCR to 00).

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	BGDM	-	-	-	-	-	-	ABCS
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	BGDM	0	R/W	Baud Rate Generator Double-Speed Mode When the BGDM bit is set to 1, the baud rate generator in this module operates in double-speed mode. This bit is valid only when asynchronous mode is selected by setting the $C/\bar{A}$ bit in SCSMR to 0 and an internal clock is selected as a clock source and the SCK pin is set as an input pin by setting the CKE[1:0] bits in SCSCR to 00. In other settings, use normal mode. 0: Normal mode 1: Double-speed mode
6 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	ABCS	0	R/W	Base Clock Select in Asynchronous Mode This bit selects the base clock frequency within a bit period in asynchronous mode. This bit is valid only in asynchronous mode (when the $C/\bar{A}$ bit in SCSMR is 0). 0: Base clock frequency is 16 times the bit rate 1: Base clock frequency is 8 times the bit rate

## 14.4 Operation

### 14.4.1 Overview

For serial communication, this module has an asynchronous mode in which characters are synchronized individually, and a clock synchronous mode in which communication is synchronized with clock pulses.

This module has a 16-stage FIFO buffer for both transmission and reception, reducing the overhead of the CPU, and enabling continuous high-speed communication. Furthermore, channels 1, 5, and 7 have  $\overline{\text{RTS}}$  and  $\overline{\text{CTS}}$  signals to be used as modem control signals.

The transmission/reception format is selected in the serial mode register (SCSMR), as shown in Table 14.9. The clock source is selected by the combination of the  $C/\overline{A}$  bit in SCSMR and the CKE1 and CKE0 bits in the serial control register (SCSCR), as shown in Table 14.10.

#### (1) Asynchronous Mode

- Data length is selectable: 7 or 8 bits
- Parity bit is selectable. So is the stop bit length (1 or 2 bits). The combination of the preceding selections constitutes the communication format and character length.
- In receiving, it is possible to detect framing errors, parity errors, receive FIFO data full, overrun errors, receive data ready, and breaks.
- The number of stored data bytes is indicated for both the transmit and receive FIFO registers.
- An internal or external clock can be selected as the clock source.
  - When an internal clock is selected, this module operates using the clock of on-chip baud rate generator.
  - When an external clock is selected, the external clock input must have a frequency 16 or 8 times the bit rate. (The on-chip baud rate generator is not used.)

## (2) Clock Synchronous Mode

- The transmission/reception format has a fixed 8-bit data length.
- In receiving, it is possible to detect overrun errors (ORER).
- An internal or external clock can be selected as the clock source.
  - When an internal clock is selected, this module operates using the clock of the on-chip baud rate generator, and outputs this clock to external devices as the synchronous clock.
  - When an external clock is selected, this module operates on the input external synchronous clock not using the on-chip baud rate generator.

**Table 14.9 SCSMR Settings and Communication Formats**

SCSMR Settings				Communication Format						
Bit 7 C/A	Bit 6 CHR	Bit 5 PE	Bit 3 STOP	Mode	Data Length	Parity Bit	Stop Bit Length			
0	0	0	0	Asynchronous	8 bits	Not set	1 bit			
			1				2 bits			
			0				Set	1 bit		
			1					2 bits		
		1	0	0		Clock synchronous	7 bits	Not set	1 bit	
				1					2 bits	
				0					Set	1 bit
				1						2 bits
1	x	x	x	Clock synchronous	8 bits	Not set		None		

[Legend]

x: Don't care

**Table 14.10 SCSMR and SCSCR Settings and Clock Source Selection**

SCSMR	SCSCR	Mode	Clock Source	SCK Pin Function	
Bit 7 C/A	Bit 1, 0 CKE[1:0]				
0	00	Asynchronous	Internal	This module does not use the SCK pin.	
	01			Outputs a clock with a frequency 16/8 times the bit rate	
	10			External	Inputs a clock with frequency 16/8 times the bit rate
	11			Setting prohibited	
1	0x	Clock synchronous	Internal	Outputs the synchronous clock	
	10		External	Inputs the synchronous clock	
	11		Setting prohibited		

[Legend]

x: Don't care

Note: When using the baud rate generator in double-speed mode (BGMD = 1), select asynchronous mode by setting the  $\overline{C/A}$  bit to 0, and select an internal clock as a clock source and the SCK pin is not used (the CKE[1:0] bits set to 00).

### 14.4.2 Operation in Asynchronous Mode

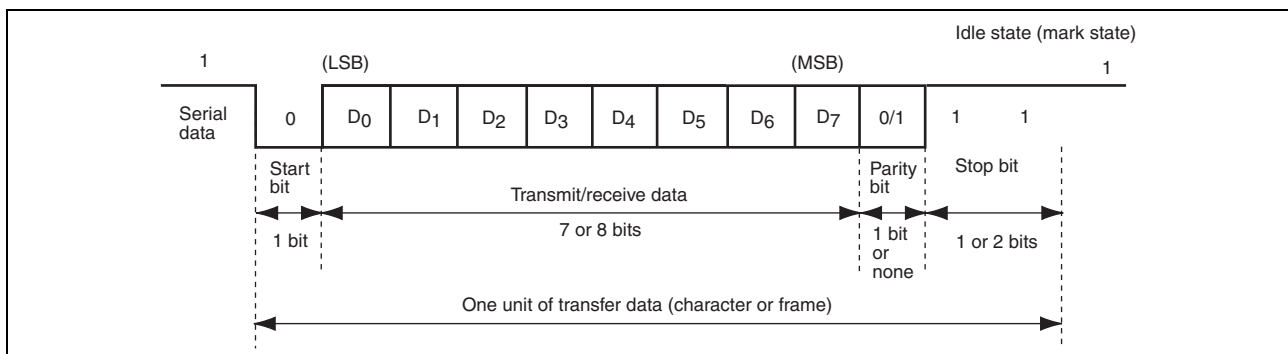
In asynchronous mode, each transmitted or received character begins with a start bit and ends with a stop bit. Serial communication is synchronized one character at a time.

The transmitting and receiving sections in this module are independent, so full duplex communication is possible. The transmitter and receiver are 16-stage FIFO buffered, so data can be written and read while transmitting and receiving are in progress, enabling continuous transmitting and receiving.

Figure 14.2 shows the general format of asynchronous serial communication.

In asynchronous serial communication, the communication line is normally held in the mark (high) state. This module monitors the line and starts serial communication when the line goes to the space (low) state, indicating a start bit.

One serial character consists of a start bit (low), data (LSB first), parity bit (high or low), and stop bit (high), in that order. When receiving in asynchronous mode, this module synchronizes at the falling edge of the start bit. This module samples each data bit on the eighth or fourth pulse of a clock with a frequency 16 or 8 times the bit rate; receive data is latched at the center of each bit.



**Figure 14.2 Example of Data Format in Asynchronous Communication (8-Bit Data with Parity and Two Stop Bits)**

## (1) Transmit/Receive Formats

Table 14.11 lists the eight communication formats that can be selected in asynchronous mode. The format is selected by settings in the serial mode register (SCSMR).

**Table 14.11 Serial Communication Formats (Asynchronous Mode)**

SCSMR Bits			Serial Transmit/Receive Format and Frame Length												
CHR	PE	STOP	1	2	3	4	5	6	7	8	9	10	11	12	
0	0	0	START	8-bit data							STOP				
0	0	1	START	8-bit data							STOP	STOP			
0	1	0	START	8-bit data							P	STOP			
0	1	1	START	8-bit data							P	STOP	STOP		
1	0	0	START	7-bit data						STOP					
1	0	1	START	7-bit data						STOP	STOP				
1	1	0	START	7-bit data						P	STOP				
1	1	1	START	7-bit data						P	STOP	STOP			

[Legend]

START: Start bit

STOP: Stop bit

P: Parity bit

## (2) Clock

An internal clock generated by the on-chip baud rate generator or an external clock input from the SCK pin can be selected as the transmit/receive clock. The clock source is selected by the  $C/\bar{A}$  bit in the serial mode register (SCSMR) and the CKE1 and CKE0 bits in the serial control register (SCSCR). For clock source selection, refer to Table 14.10, SCSMR and SCSCR Settings and Clock Source Selection.

When an external clock is input at the SCK pin, it must have a frequency equal to 16 or 8 times the desired bit rate. When this module operates on an internal clock, it can output a clock signal on the SCK pin. The frequency of this output clock is 16 or 8 times the desired bit rate.

## (3) Transmitting and Receiving Data

- Initialization (Asynchronous Mode)

Before transmitting or receiving, clear the TE and RE bits to 0 in the serial control register (SCSCR), then initialize this module as follows.

When changing the operation mode or the communication format, always clear the TE and RE bits to 0 before following the procedure given below. Clearing TE to 0 initializes the transmit shift register (SCTSR). Clearing TE and RE to 0, however, does not initialize the serial status register (SCFSR), transmit FIFO data register (SCFTDR), or receive FIFO data register (SCFRDR), which retain their previous contents. Clear TE to 0 after all transmit data has been transmitted and the TEND flag in the SCFSR is set. The TE bit can be cleared to 0 during transmission, but the transmit data goes to the Mark state after the bit is cleared to 0. Set the TFRST bit in SCFCR to 1 and reset SCFTDR before TE is set again to start transmission.

When an external clock is used, the clock should not be stopped during initialization or subsequent operation. The operation becomes unreliable if the clock is stopped.

Figure 14.3 shows a sample flowchart for initialization.

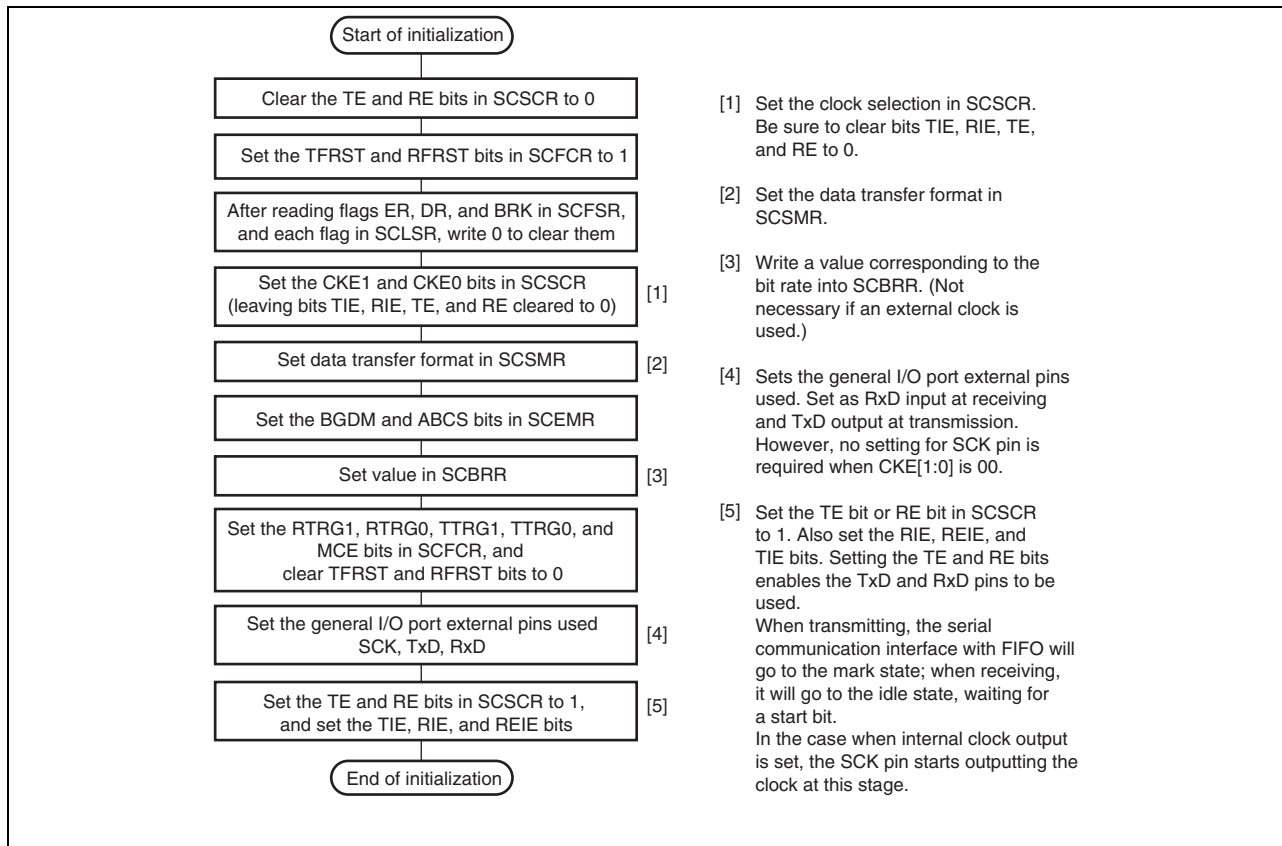


Figure 14.3 Sample Flowchart for Initialization

- Transmitting Serial Data (Asynchronous Mode)

Figure 14.4 shows a sample flowchart for serial transmission.

Use the following procedure for serial data transmission after enabling transmission.

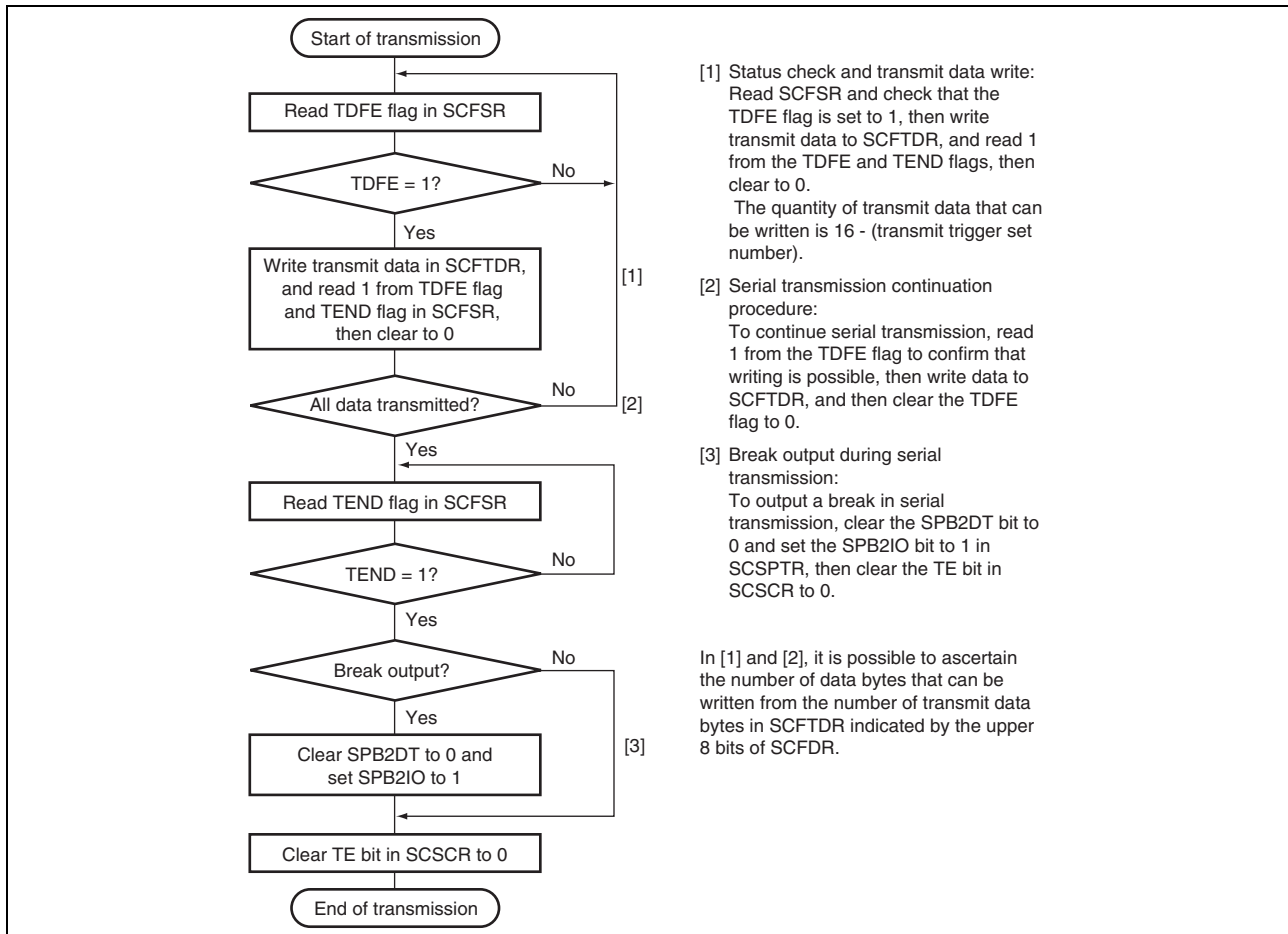


Figure 14.4 Sample Flowchart for Transmitting Serial Data



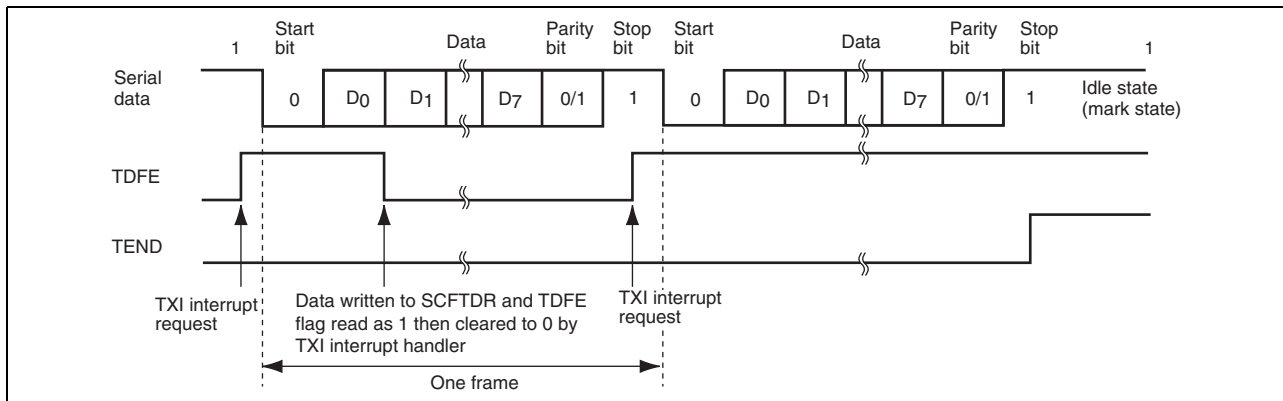
In serial transmission, this module operates as described below.

1. When data is written into the transmit FIFO data register (SCFTDR), the data is transferred from SCFTDR to the transmit shift register (SCTSR). Confirm that the TDFE flag in the serial status register (SCFSR) is set to 1 before writing transmit data to SCFTDR. The number of data bytes that can be written is (16 – transmit trigger setting).
2. When data is transferred from SCFTDR to SCTSR and transmission is started, consecutive transmit operations are performed until there is no transmit data left in SCFTDR. When the number of transmit data bytes in SCFTDR falls below the transmit trigger number set in the FIFO control register (SCFCR), the TDFE flag is set. If the TIE bit in the serial control register (SCSR) is set to 1 at this time, a transmit-FIFO-data-empty interrupt (TXI) request is generated.

The serial transmit data is sent from the TxD pin in the following order.

- A. Start bit: One-bit 0 is output.
  - B. Transmit data: 8-bit or 7-bit data is output in LSB-first order.
  - C. Parity bit: One parity bit (even or odd parity) is output. (A format in which a parity bit is not output can also be selected.)
  - D. Stop bit(s): One or two 1 bits (stop bits) are output.
  - E. Mark state: 1 is output continuously until the start bit that starts the next transmission is sent.
3. The SCFTDR transmit data is checked at the timing for sending the stop bit. If data is present, the data is transferred from SCFTDR to SCTSR, the stop bit is sent, and then serial transmission of the next frame is started.

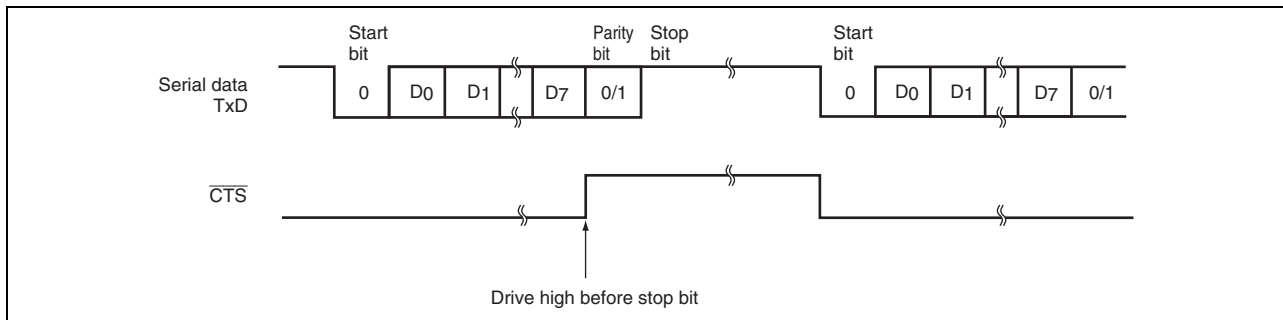
Figure 14.5 shows an example of the operation for transmission.



**Figure 14.5 Example of Transmit Operation (8-Bit Data, Parity, 1 Stop Bit)**

- When modem control is enabled in channels 1, 5, and 7, transmission can be stopped and restarted in accordance with the  $\overline{\text{CTS}}$  input value. When  $\overline{\text{CTS}}$  is set to 1, if transmission is in progress, the line goes to the mark state after transmission of one frame. When  $\overline{\text{CTS}}$  is set to 0, the next transmit data is output starting from the start bit.

Figure 14.6 shows an example of the operation when modem control is used.



**Figure 14.6 Example of Operation Using Modem Control ( $\overline{\text{CTS}}$ )**

- Receiving Serial Data (Asynchronous Mode)

Figure 14.7 and Figure 14.8 show sample flowcharts for serial reception.

Use the following procedure for serial data reception after enabling reception.

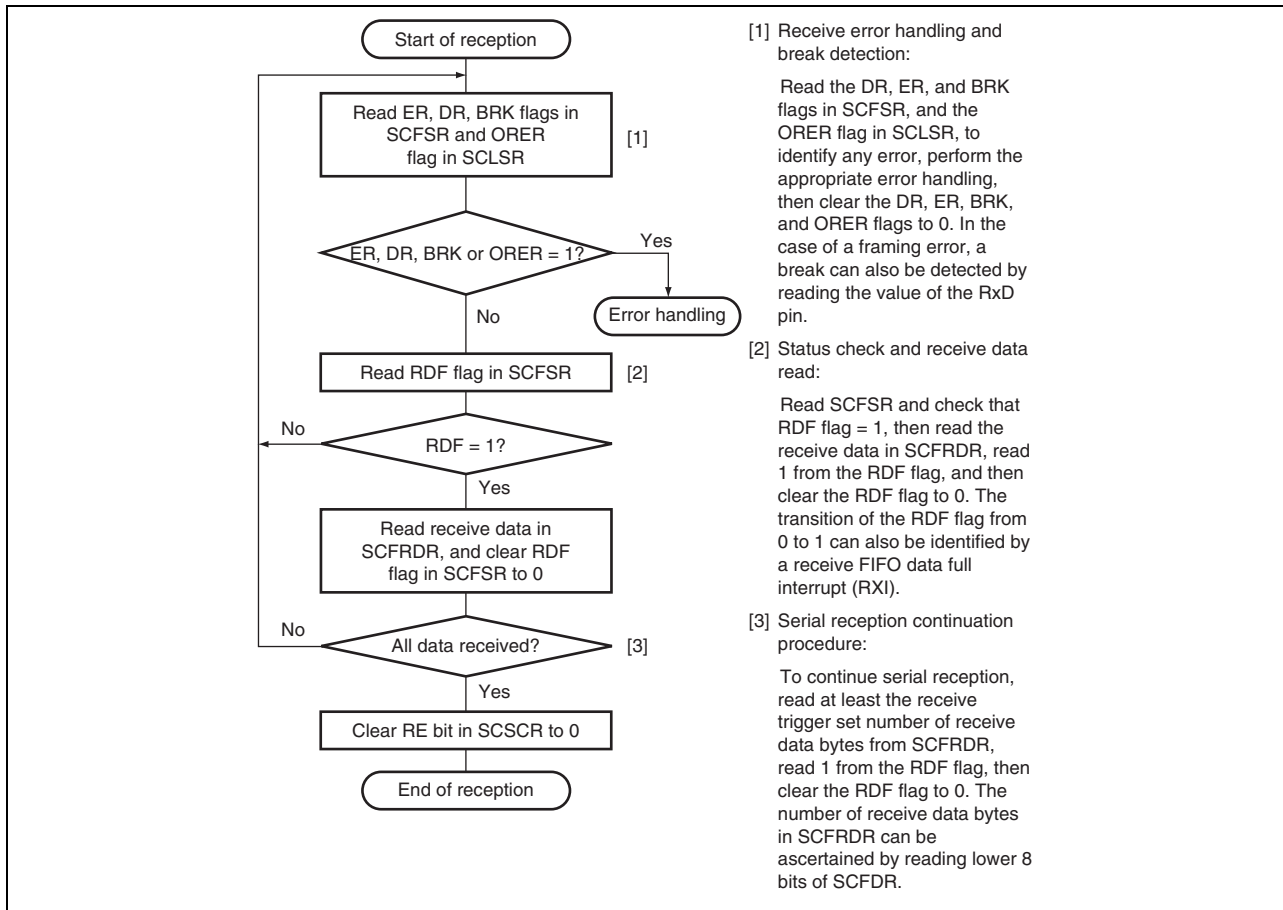
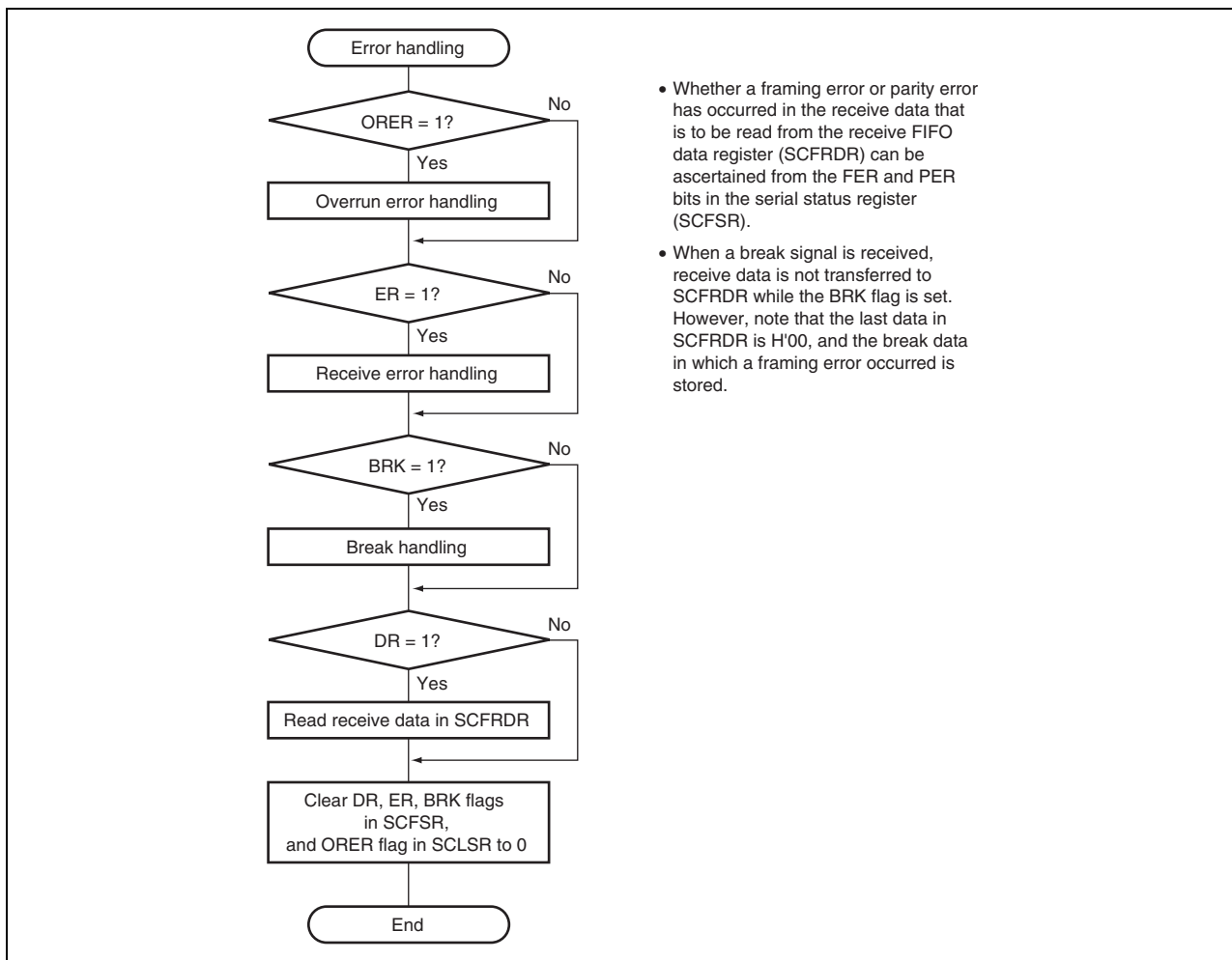


Figure 14.7 Sample Flowchart for Receiving Serial Data (1)



- Whether a framing error or parity error has occurred in the receive data that is to be read from the receive FIFO data register (SCFRDR) can be ascertained from the FER and PER bits in the serial status register (SCFSR).
- When a break signal is received, receive data is not transferred to SCFRDR while the BRK flag is set. However, note that the last data in SCFRDR is H'00, and the break data in which a framing error occurred is stored.

Figure 14.8 Sample Flowchart for Receiving Serial Data (2)

In serial reception, this module operates as described below.

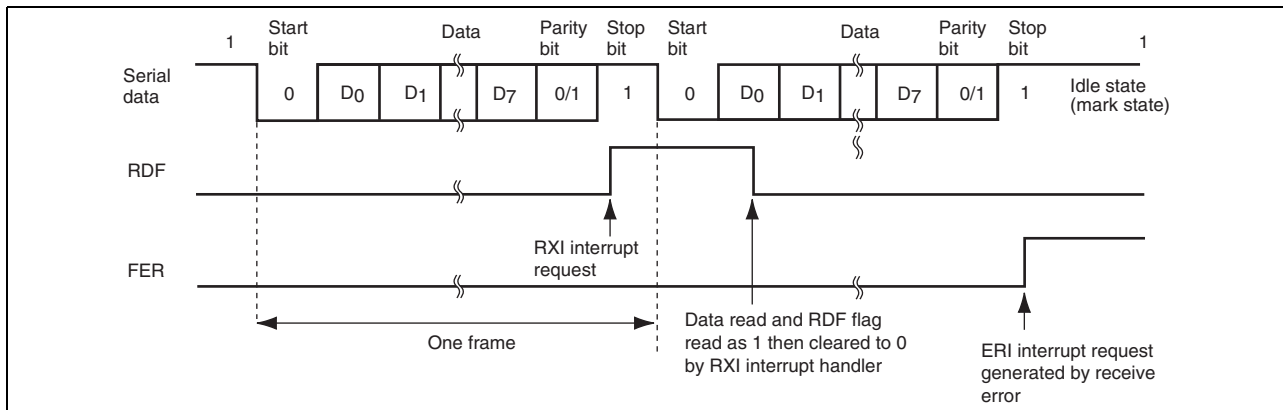
1. The transmission line is monitored, and if a 0 start bit is detected, internal synchronization is performed and reception is started.
2. The received data is stored in SCRSR in LSB-to-MSB order.
3. The parity bit and stop bit are received.  
After receiving these bits, this module carries out the following checks.
  - A. Stop bit check: Checks whether the stop bit is 1. If there are two stop bits, only the first is checked.
  - B. Checks whether receive data can be transferred from the receive shift register (SCRSR) to SCFRDR.
  - C. Overrun check: Checks that the ORER flag is 0, indicating that the overrun error has not occurred.
  - D. Break check: Checks that the BRK flag is 0, indicating that the break state is not set.

If all the above checks are passed, the receive data is stored in SCFRDR.

Note: When a parity error or a framing error occurs, reception is not suspended.

4. If the RIE bit in SCSCR is set to 1 when the RDF or DR flag changes to 1, a receive-FIFO-data-full interrupt (RXI) request is generated. If the RIE bit or the REIE bit in SCSCR is set to 1 when the ER flag changes to 1, a receive-error interrupt (ERI) request is generated. If the RIE bit or the REIE bit in SCSCR is set to 1 when the BRK or ORER flag changes to 1, a break reception interrupt (BRI) request is generated.

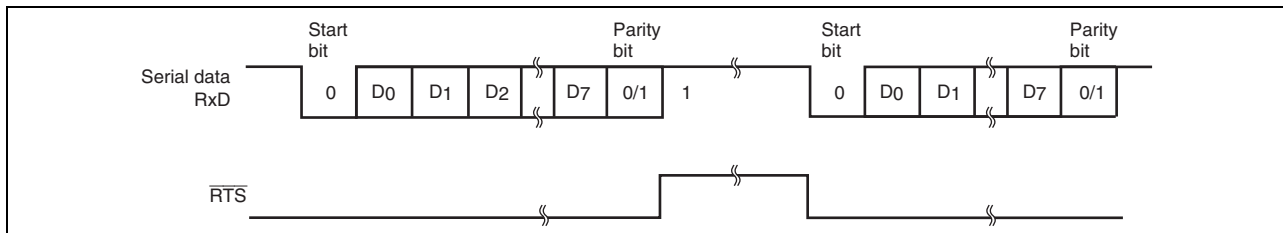
Figure 14.9 shows an example of the operation for reception in asynchronous mode.



**Figure 14.9 Example of Receive Operation(8-Bit Data, Parity, 1 Stop Bit)**

- When modem control is enabled in channels 1, 5, and 7, the  $\overline{\text{RTS}}$  signal is output when SCFRDR is empty. When  $\overline{\text{RTS}}$  is 0, reception is possible. When  $\overline{\text{RTS}}$  is 1, this indicates that SCFRDR exceeds the number set for the RTS output active trigger.

Figure 14.10 shows an example of the operation when modem control is used.



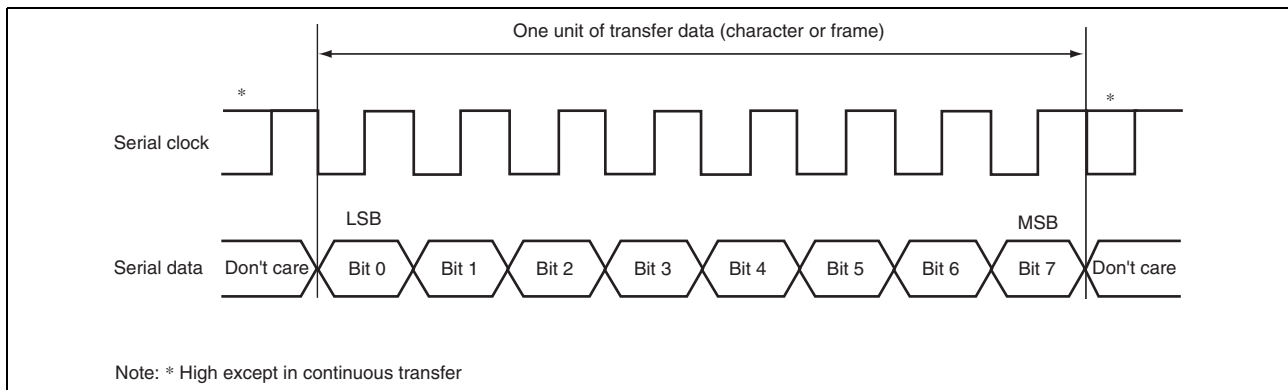
**Figure 14.10 Example of Operation Using Modem Control ( $\overline{\text{RTS}}$ )**

### 14.4.3 Operation in Clock Synchronous Mode

In clock synchronous mode, data is transmitted and received in synchronization with clock pulses. This mode is suitable for high-speed serial communication.

The transmitter and receiver in this module are independent, so full-duplex communication is possible while sharing the same clock. The transmitter and receiver are also 16-stage FIFO buffered, so continuous transmitting or receiving is possible by reading or writing data while transmitting or receiving is in progress.

Figure 14.11 shows the general format in clock synchronous serial communication.



**Figure 14.11 Data Format in Clock Synchronous Communication**

In clock synchronous serial communication, each data bit is output on the communication line from one falling edge of the serial clock to the next. Data is guaranteed valid at the rising edge of the serial clock.

In each character, the serial data bits are transmitted in order from the LSB (first) to the MSB (last). After output of the MSB, the communication line remains in the state of the MSB.

In clock synchronous mode, data is received in synchronization with the rising edge of the synchronous clock.

### (1) Transmit/Receive Formats

The data length is fixed at eight bits. No parity bit can be added.

### (2) Clock

An internal clock generated by the on-chip baud rate generator or an external synchronous clock input from the SCK pin can be selected by setting the  $C/\bar{A}$  bit in SCSMR and the  $CKE[1:0]$  bits in SCSCR,.

When this module operates on an internal clock, it outputs the synchronous clock signal at the SCK pin. Eight clock pulses are output per one character transmission or reception. When transmission or reception is not performed, the clock signal remains in the high state. When only receiving is performed with the internal clock selected, the clock signal pulses are output while the RE bit of SCSCR is 1 and the number of data in receive FIFO is more than the receive FIFO data trigger number.

### (3) Transmitting and Receiving Data

- Initialization (Clock Synchronous Mode)

Before transmitting, receiving, or changing the mode or communication format, the software must clear the TE and RE bits to 0 in the serial control register (SCSCR), and then initialize this module following the procedure described below.

Clearing TE to 0 initializes the transmit shift register (SCTSR). Clearing RE to 0, however, does not initialize the RDF, PER, FER, and ORER flags and receive data register (SCRDR), which retain their previous contents.



Figure 14.12 shows a sample flowchart for initialization.

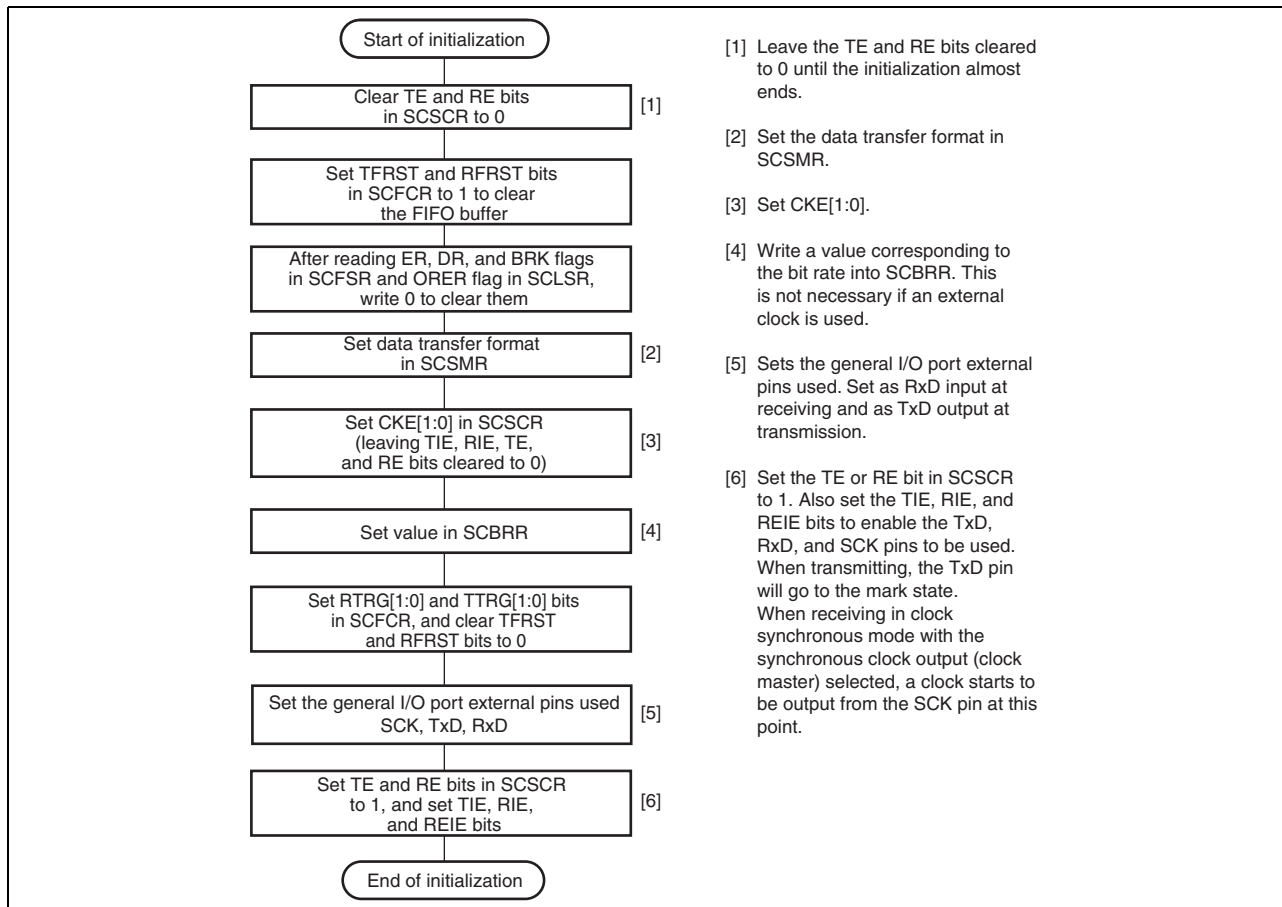


Figure 14.12 Sample Flowchart for Initialization

- Transmitting Serial Data (Clock Synchronous Mode)

Figure 14.13 shows a sample flowchart for transmitting serial data.

Use the following procedure for serial data transmission after enabling transmit operation.

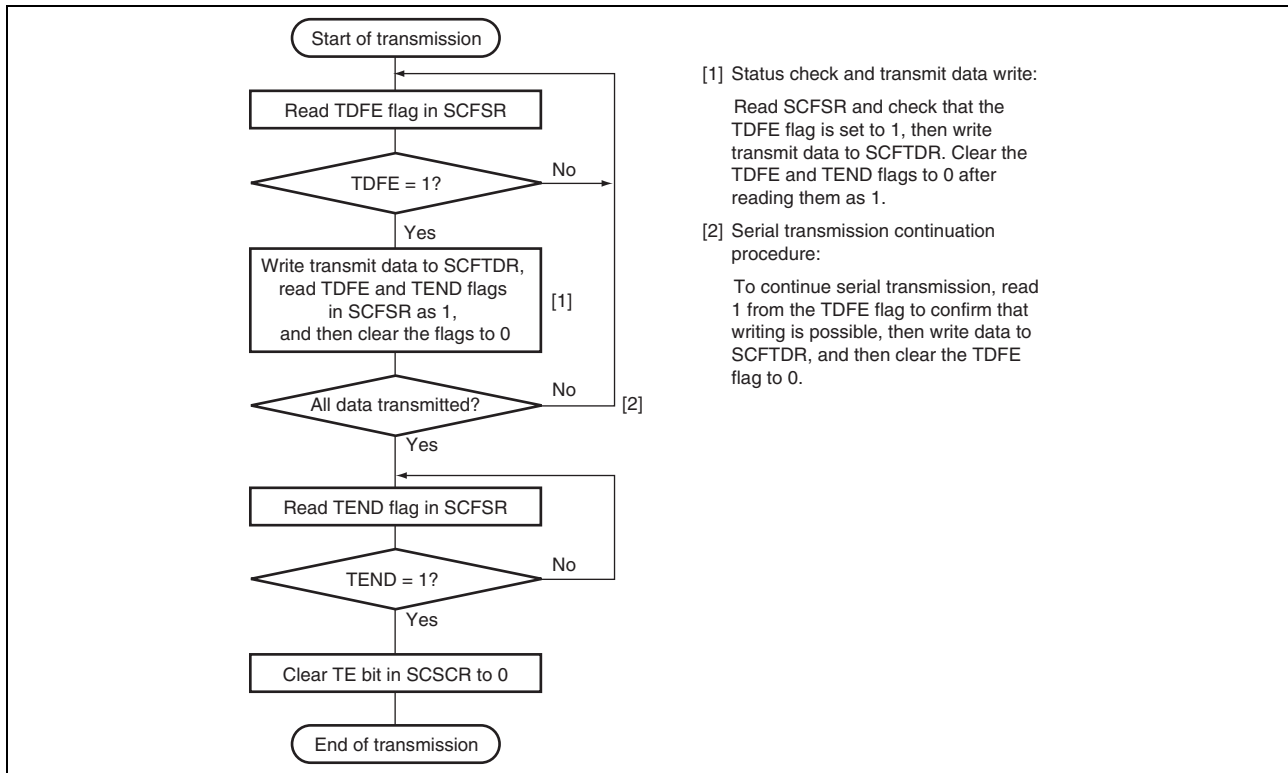
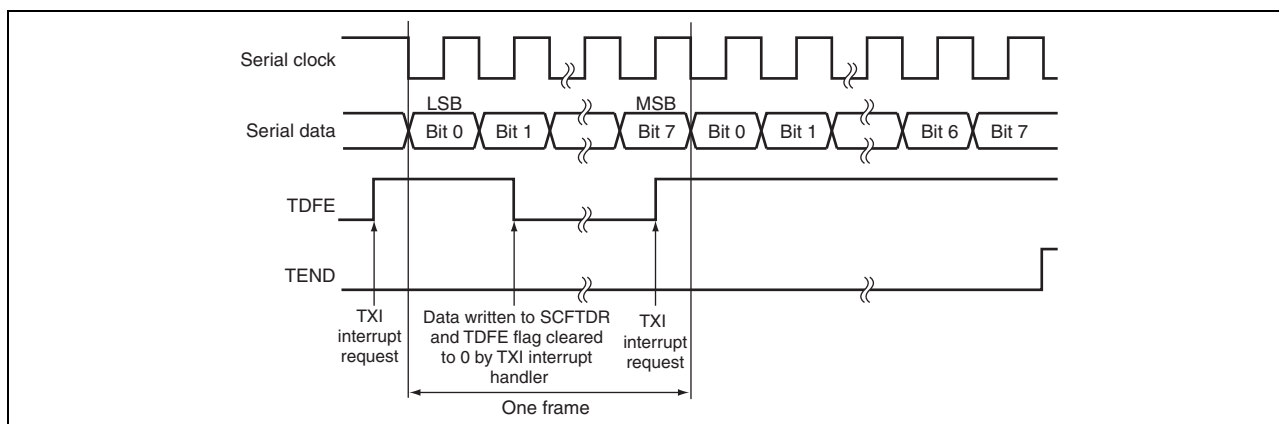


Figure 14.13 Sample Flowchart for Transmitting Serial Data

In serial transmission, this module operates as described below.

1. When data is written into the transmit FIFO data register (SCFTDR), the data is transferred from SCFTDR to the transmit shift register (SCTSR). Confirm that the TDFE flag in the serial status register (SCFSR) is set to 1 before writing transmit data to SCFTDR. The number of transmit data bytes that can be written is (16 – transmit trigger setting).
2. When data is transferred from SCFTDR to SCTSR and transmission is started, consecutive transmit operations are performed until there is no transmit data left in SCFTDR. When the number of transmit data bytes in SCFTDR falls below the transmit trigger number set in the FIFO control register (SCFCR), the TDFE flag is set. If the TIE bit in the serial control register (SCSCR) is set to 1 at this time, a transmit-FIFO-data-empty interrupt (TXI) request is generated.  
If clock output mode is selected, eight synchronous clock pulses are output. If an external clock source is selected, data is output in synchronization with the input clock. Data is output from the TxD pin in order from the LSB (bit 0) to the MSB (bit 7).
3. The SCFTDR transmit data is checked at the timing for sending the MSB (bit 7). If data is present, the data is transferred from SCFTDR to SCTSR, and then serial transmission of the next frame is started. If there is no data, the TxD pin holds the state after the TEND flag in SCFSR is set to 1 and the MSB (bit 7) is sent.
4. After the end of serial transmission, the SCK pin is held in the high state.

Figure 14.14 shows an example of transmit operation.



**Figure 14.14** Example of Transmit Operation

- Receiving Serial Data (Clock Synchronous Mode)

Figure 14.15 and Figure 14.16 show sample flowcharts for receiving serial data. Use the following procedure for serial data reception after enabling receive operation. When switching from asynchronous mode to clock synchronous mode without initialization, make sure that ORER, PER, and FER are cleared to 0.

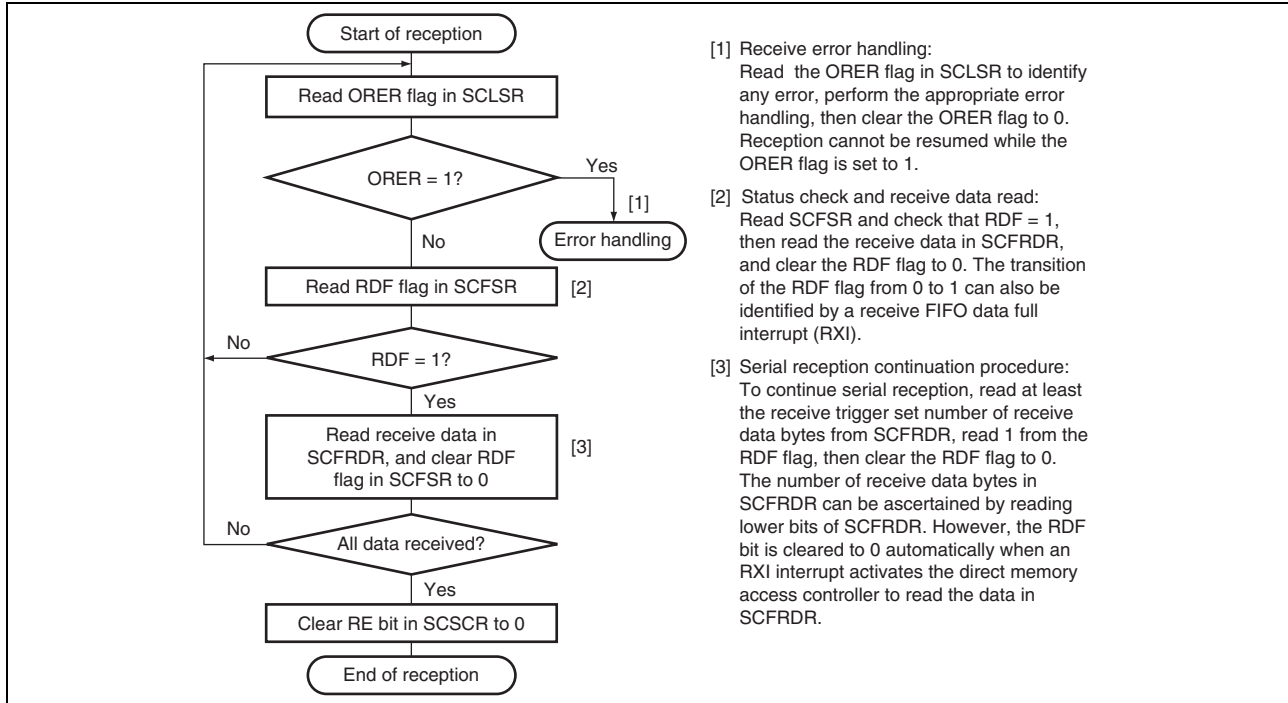


Figure 14.15 Sample Flowchart for Receiving Serial Data (1)

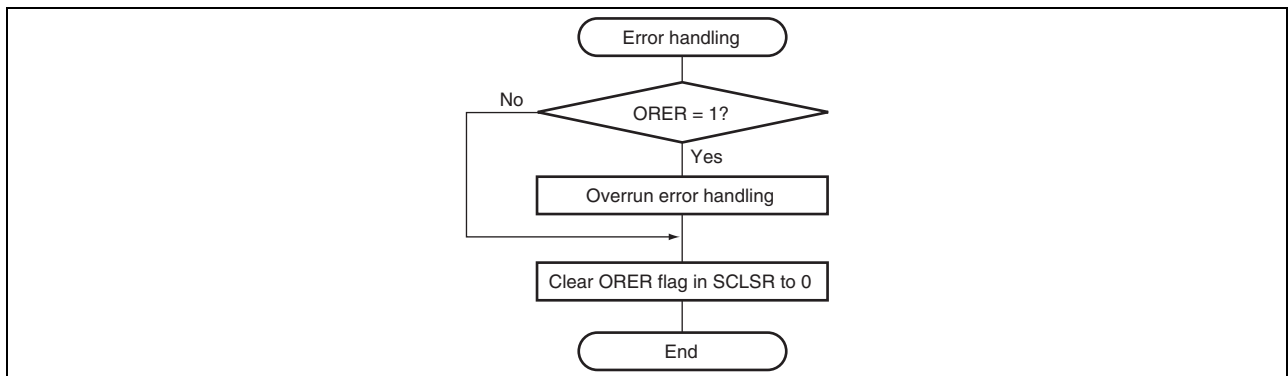
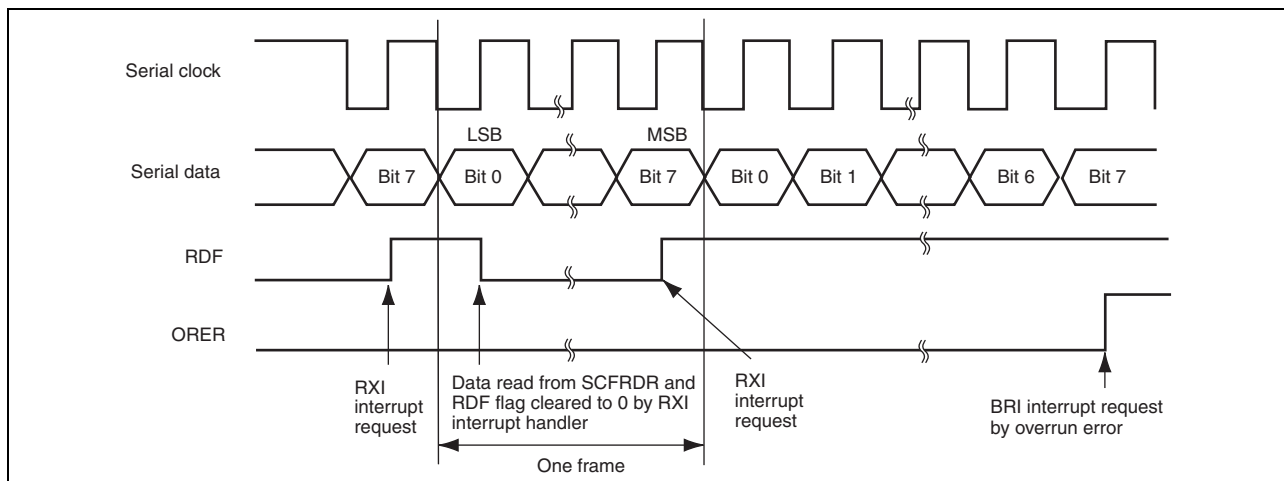


Figure 14.16 Sample Flowchart for Receiving Serial Data (2)

In serial reception, this module operates as described below.

1. Reception is started in synchronization with synchronous clock input or output.
2. Receive data is shifted into SCRSR in order from the LSB to the MSB. After the data reception, whether the receive data can be loaded from SCRSR into SCFRDR or not is checked. If this check is passed, the RDF flag is set to 1 and the received data is stored in SCFRDR. If the check is not passed (overrun error is detected), further reception is prevented.
3. After setting RDF to 1, if the receive interrupt enable bit (RIE) is set to 1 in SCSCR, a receive FIFO data full interrupt (RXI) request is generated. If the ORER bit is set to 1 and the receive interrupt enable bit (RIE) or the receive error interrupt enable bit (REIE) in SCSCR is also set to 1, a break interrupt (BRI) request is generated.

Figure 14.17 shows an example of receive operation.



**Figure 14.17 Example of Receive Operation**

• Transmitting and Receiving Serial Data Simultaneously (Clock Synchronous Mode)

Figure 14.18 shows a sample flowchart for transmitting and receiving serial data simultaneously.

Use the following procedure for the simultaneous transmission/reception of serial data, after enabling transmit/receive operation.

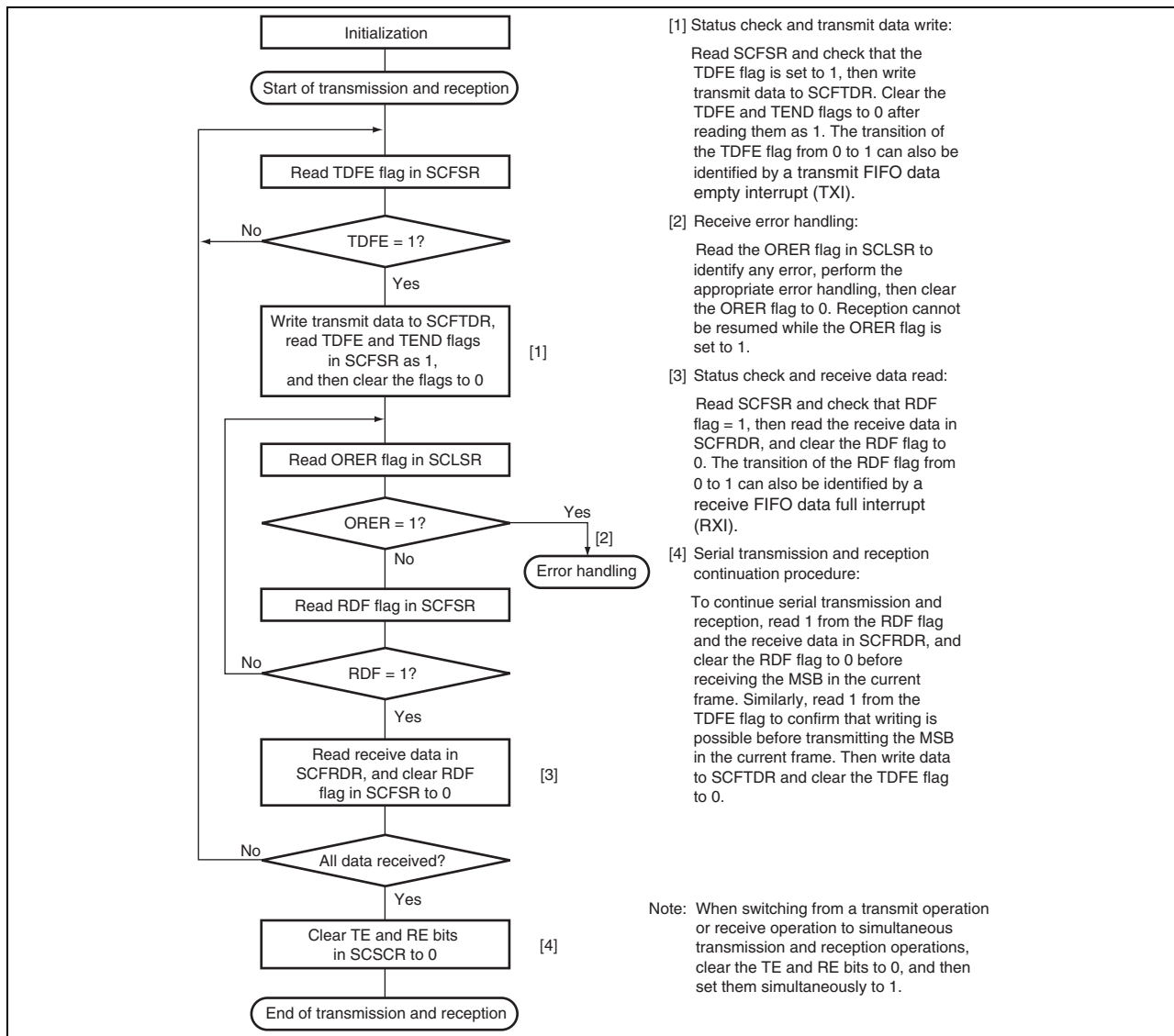


Figure 14.18 Sample Flowchart for Transmitting/Receiving Serial Data

## 14.5 Interrupts

This module has four interrupt sources: transmit-FIFO-data-empty (TXI), receive-error (ERI), receive FIFO data full (RXI), and break (BRI).

Table 14.12 shows the interrupt sources and their order of priority. The interrupt sources are enabled or disabled by means of the TIE, RIE, and REIE bits in SCSCR. A separate interrupt request is sent to the interrupt controller for each of these interrupt sources.


When a TXI request is enabled by the TIE bit and the TDFE flag in the serial status register (SCFSR) is set to 1, a TXI interrupt request is generated. The direct memory access controller can be activated and data transfer performed by this TXI interrupt request. At this time, an interrupt request is not sent to the CPU.

When an RXI request is enabled by the RIE bit and the RDF flag or the DR flag in SCFSR is set to 1, an RXI interrupt request is generated. The direct memory access controller can be activated and data transfer performed by this RXI interrupt request. At this time, an interrupt request is not sent to the CPU. The RXI interrupt request caused by the DR flag is generated only in asynchronous mode.

When the RIE bit is set to 0 and the REIE bit is set to 1, this module requests only an ERI or a BRI interrupt without requesting an RXI interrupt.

The TXI indicates that transmit data can be written, and the RXI indicates that there is receive data in SCFRDR.

**Table 14.12 Interrupt Sources**

Interrupt Source	Description	Direct Memory Access Controller Activation	Priority on Reset Release
BRI	Interrupt initiated by break (BRK) or overrun error (ORER)	Not possible	High
ERI	Interrupt initiated by receive error (ER)	Not possible	
RXI	Interrupt initiated by receive FIFO data full (RDF) or data ready (DR)	Possible	
TXI	Interrupt initiated by transmit FIFO data empty (TDFE)	Possible	

## 14.6 Usage Notes

Note the following when using this module.

### 14.6.1 SCFTDR Writing and TDFE Flag

The TDFE flag in the serial status register (SCFSR) is set when the number of transmit data bytes written in the transmit FIFO data register (SCFTDR) has fallen below the transmit trigger number set by bits TTRG[1:0] in the FIFO control register (SCFCR). After the TDFE flag is set, transmit data up to the number of empty bytes in SCFTDR can be written, allowing efficient continuous transmission.

However, if the number of data bytes written in SCFTDR is equal to or less than the transmit trigger number, the TDFE flag will be set to 1 again after being read as 1 and cleared to 0. The TDFE flag should therefore be cleared to 0 after being read as 1 when SCFTDR contains more than the transmit trigger number of transmit data bytes.

The number of transmit data bytes in SCFTDR can be found from the upper 8 bits of the FIFO data count register (SCFDR).

### 14.6.2 SCFRDR Reading and RDF Flag

The RDF flag in the serial status register (SCFSR) is set when the number of receive data bytes in the receive FIFO data register (SCFRDR) has become equal to or greater than the receive trigger number set by bits RTRG[1:0] in the FIFO control register (SCFCR). After RDF flag is set, receive data equivalent to the trigger number can be read from SCFRDR, allowing efficient continuous reception.

However, if the number of data bytes in SCFRDR exceeds the trigger number, the RDF flag will be set to 1 again after being read as 1 and then cleared to 0. The RDF flag should therefore be cleared to 0 after being read as 1 after reading the number of the received data in the receive FIFO data register (SCFRDR) which is less than the trigger number.

The number of receive data bytes in SCFRDR can be found from the lower 8 bits of the FIFO data count register (SCFDR).

### 14.6.3 Restriction on Direct Memory Controller Usage

When the direct memory access controller writes data to SCFTDR due to a TXI interrupt request, the state of the TEND flag becomes undefined. Therefore, the TEND flag should not be used as the transfer end flag in such a case.

### 14.6.4 Break Detection and Processing

Break signals can be detected by reading the RxD pin directly when a framing error (FER) is detected. In the break state the input from the RxD pin consists of all 0s, so the FER flag is set and the parity error flag (PER) may also be set.

Note that, although transfer of receive data to SCFRDR is halted in the break state, the receive operation is continued.



### 14.6.5 Sending a Break Signal

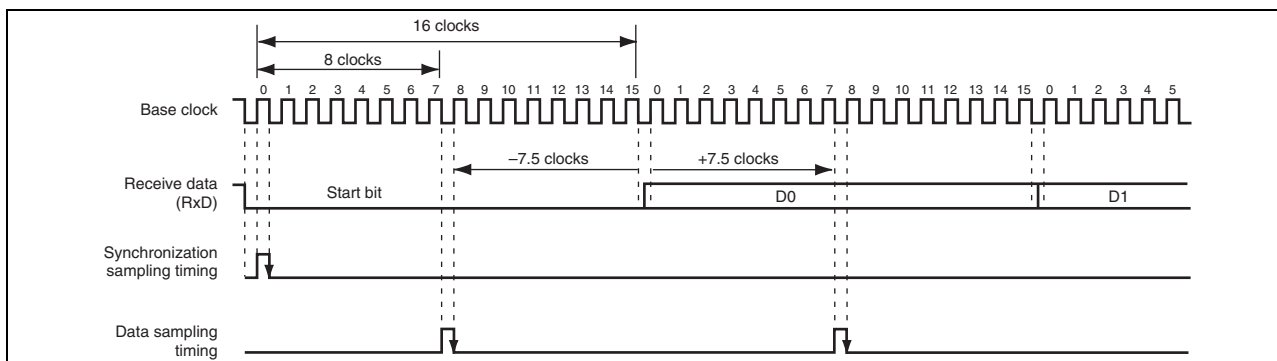
The I/O condition and level of the TxD pin are determined by the SPB2IO and SPB2DT bits in the serial port register (SCSPTR). This feature can be used to send a break signal.

Until TE bit is set to 1 (enabling transmission) after initializing, the TxD pin does not work. During the period, mark status is performed by the SPB2DT bit. Therefore, the SPB2IO and SPB2DT bits should be set to 1 (high level output).

To send a break signal during serial transmission, clear the SPB2DT bit to 0 (designating low level), then clear the TE bit to 0 (halting transmission). When the TE bit is cleared to 0, the transmitter is initialized regardless of the current transmission state, and 0 is output from the TxD pin.

### 14.6.6 Receive Data Sampling Timing and Receive Margin (Asynchronous Mode)

This module operates on a base clock with a frequency 16 or 8 times the bit rate. In reception, the falling edge of the start bit is sampled at the base clock to perform synchronization internally. Receive data is latched at the rising edge of the eighth or fourth base clock pulse. When this module operates on a base clock with a frequency 16 times the bit rate, the receive data is sampled at the timing shown in Figure 14.19.



**Figure 14.19 Receive Data Sampling Timing in Asynchronous Mode (Operation on a Base Clock with a Frequency 16 Times the Bit Rate)**

The receive margin in asynchronous mode can therefore be expressed as shown in equation 1.

Equation 1:

$$M = \left| \left( 0.5 - \frac{1}{2N} \right) - (L - 0.5)F - \frac{|D - 0.5|}{N} (1 + F) \right| \times 100 \%$$

Where: M: Receive margin (%)  
 N: Ratio of clock frequency to bit rate (N = 16 or 8)  
 D: Clock duty (D = 0 to 1.0)  
 L: Frame length (L = 9 to 12)  
 F: Absolute deviation of clock frequency

From equation 1, if F = 0, D = 0.5 and N = 16, the receive margin is 46.875%, as given by equation 2.

**Equation 2:**

When  $D = 0.5$  and  $F = 0$ :

$$\begin{aligned} M &= (0.5 - 1/(2 \times 16)) \times 100\% \\ &= 46.875\% \end{aligned}$$

This is a theoretical value. A reasonable margin to allow in system designs is 20% to 30%.

### 14.6.7 Selection of Base Clock in Asynchronous Mode

In this LSI, when asynchronous mode is selected, the base clock frequency within a bit period can be set to the frequency 16 or 8 times the bit rate by setting the ABCS bit in SCEMR.

Note that, however, if the base clock frequency 8 times the bit rate is used, receive margin is decreased as calculated using equation 1 in section 14.6.6, Receive Data Sampling Timing and Receive Margin (Asynchronous Mode).

If the desired bit rate can be set simply by setting SCBRR and the CKS1 and CKS0 bits in SCSMR, it is recommended to use the base clock frequency within a bit period 16 times the bit rate (by setting the ABCS bit in SCEMR to 0). If an internal clock is selected as a clock source and the SCK pin is not used, the bit rate can be increased without decreasing receive margin by selecting double-speed mode for the baud rate generator (setting the BGDM bit in SCEMR to 1).

## 15. Serial Communications Interface

This LSI has two independent serial communications interface (SCI) channels.

The SCI can handle both asynchronous and clock synchronous serial communications.

Asynchronous serial data communications can be carried out with standard asynchronous communications chips such as a Universal Asynchronous Receiver/Transmitter (UART) or Asynchronous Communications Interface Adapter (ACIA).

As an extended function in asynchronous communications mode, the SCI also supports smart card (IC card) interfaces conforming to ISO/IEC 7816-3 (standard for Identification Cards).

### 15.1 Overview

Table 15.1 lists the specifications of the SCI module.

Figure 15.1 is a block diagram of the SCI module.

**Table 15.1 Specifications of SCI**

Item		Specifications
Serial communications mode		<ul style="list-style-type: none"> <li>Asynchronous operation</li> <li>Clock synchronous operation</li> <li>Smart card interface</li> </ul>
Transfer speed		Bit rate specifiable with on-chip baud rate generator.
Full-duplex communications		Transmitter: Enables continuous transmission by double-buffering. Receiver: Enables continuous reception by double-buffering.
I/O pins		See Table 15.2.
Data transfer		Selectable as LSB-first or MSB-first transfer
Interrupt sources		Transmit-end, transmit-data-empty, receive-data-full, and receive error
Power consumption reduction function		Module-standby state can be set for each channel.
Asynchronous mode	Data length	7 or 8 bits
	Transmission stop bit	1 or 2 bits
	Parity	Even, odd, or none
	Receive error detection	Parity, overrun, and framing errors
	Hardware flow control	CTS <sub>n</sub> and RTS <sub>n</sub> pins can be used in transfer control.
	Break detection	Break can be detected by reading RXD <sub>n</sub> pin level directly in case of a framing error
	Clock source	Selectable from internal or external clock Enables transfer rate clock input from TMR (SCI5 and SCI6)
	Multi-processor communications function	Serial communication among multiple processors
Clock synchronous mode	Data length	8 bits
	Receive error detection	Overrun errors
	Hardware flow control	CTS <sub>n</sub> and RTS <sub>n</sub> pins can be used in transfer control.
Smart card interface mode	Error processing	An error signal can be automatically transmitted on detection of a parity error during reception
		Data can be automatically re-transmitted on receiving an error signal during transmission
	Data type	Both direct convention and inverse convention are supported.

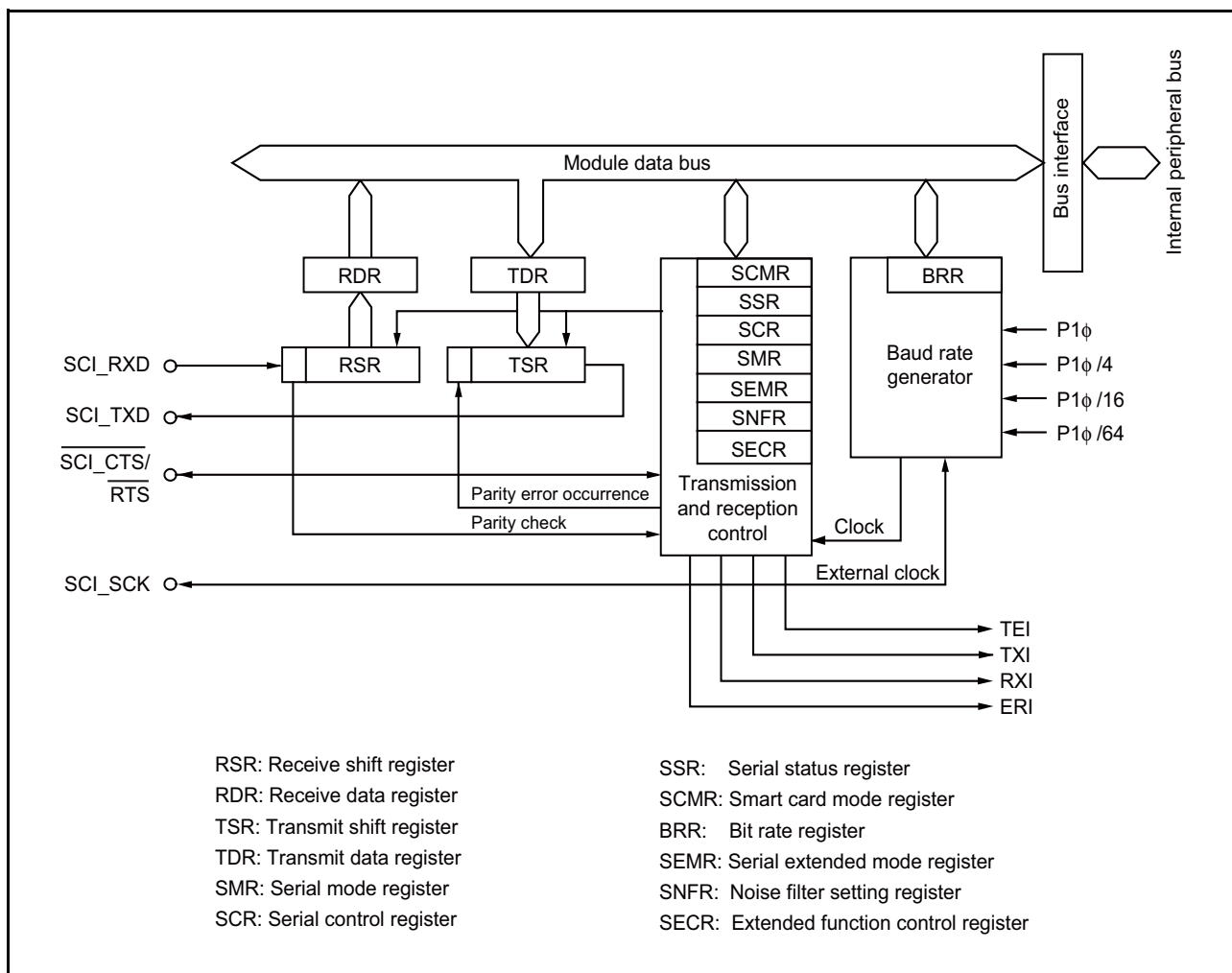


Figure 15.1 Block Diagram of SCI

Table 15.2 lists the pin configuration of the SCIs.

Table 15.2 Input and Output Pins of the SCIs

Channel	Pin Name	I/O	Function
SCI0	SCI_SCK0	I/O	SCI0 clock input/output
	SCI_RXD0	Input	SCI0 receive data input
	SCI_TXD0	Output	SCI0 transmit data output
	SCI_CTS0/RTS0	I/O	SCI0 transfer start control input/output
SCI1	SCI_SCK1	I/O	SCI1 clock input/output
	SCI_RXD1	Input	SCI1 receive data input
	SCI_TXD1	Output	SCI1 transmit data output
	SCI_CTS1/RTS1	I/O	SCI1 transfer start control input/output

Note: • These pins are referred to as SCKn, TXDn, RTSn#, and CTSn# in the text in this section.

## 15.2 Register Descriptions

Table 15.3 is a list of registers.

**Table 15.3 List of Registers**

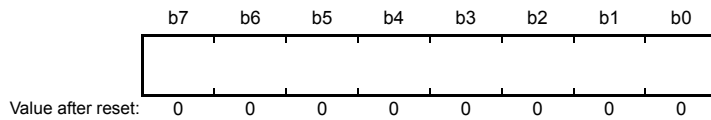
Channel	Register Name	Symbol	Value after Reset	Address	Access size
0	Serial mode register 0	SMR0	H'00	H'E800B000	8
	Bit rate register 0	BRR0	H'FF	H'E800B001	8
	Serial control register 0	SCR0	H'00	H'E800B002	8
	Transmit data register 0	TDR0	H'FF	H'E800B003	8
	Serial status register 0	SSR0	H'84	H'E800B004	8
	Receive data register 0	RDR0	H'00	H'E800B005	8
	Smart card mode register 0	SCMR0	H'F2	H'E800B006	8
	Serial extended mode register 0	SEMR0	H'00	H'E800B007	8
	Noise filter setting register 0	SNFR0	H'00	H'E800B008	8
	Extended function control register 0	SECR0	H'00	H'E800B00D	8
1	Serial mode register 1	SMR1	H'00	H'E800B800	8
	Bit rate register 1	BRR1	H'FF	H'E800B801	8
	Serial control register 1	SCR1	H'00	H'E800B802	8
	Transmit data register 1	TDR1	H'FF	H'E800B803	8
	Serial status register 1	SSR1	H'84	H'E800B804	8
	Receive data register 1	RDR1	H'00	H'E800B805	8
	Smart card mode register 1	SCMR1	H'F2	H'E800B806	8
	Serial extended mode register 1	SEMR1	H'00	H'E800B807	8
	Noise filter setting register 1	SNFR1	H'00	H'E800B808	8
	Extended function control register 1	SECR1	H'00	H'E800B80D	8

### 15.2.1 Receive Shift Register (RSR)

RSR is a shift register which is used to receive serial data input from the RXDn pin and converts it into parallel data. When one frame of data has been received, it is transferred to RDR automatically.

RSR cannot be directly accessed by the CPU.

### 15.2.2 Receive Data Register (RDR)



RDR is an 8-bit register that stores receive data.

When the SCI has received one frame of serial data, it transfers the received serial data from RSR to RDR where it is stored. This allows RSR to receive the next data.

Since RSR and RDR function as a double buffer in this way, continuous receive operations can be performed.

Read RDR only once after a receive data full interrupt (RXI) has occurred. Note that if next one frame of data is received before reading receive data from RDR, an overrun error occurs.

RDR cannot be written to by the CPU.

### 15.2.3 Transmit Data Register (TDR)



TDR is an 8-bit register that stores transmit data.

When the SCI detects that TSR is empty, it transfers the transmit data written in TDR to TSR and starts transmission.

The double-buffered structures of TDR and TSR enable continuous serial transmission. If the next transmit data has already been written to TDR when one frame of data is transmitted, the SCI transfers the written data to TSR to continue transmission.

The CPU is able to read from or write to TDR at any time. Only write data for transmission to TDR once after each instance of the transmit data empty interrupt (TXI).

### 15.2.4 Transmit Shift Register (TSR)

TSR is a shift register that transmits serial data.

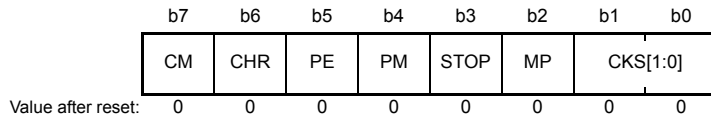
To perform serial data transmission, the SCI first automatically transfers transmit data from TDR to TSR, and then sends the data to the TXDn pin.

TSR cannot be directly accessed by the CPU.

## 15.2.5 Serial Mode Register (SMR)

Note: • Some bits in SMR have different functions in serial communications interface mode and smart card interface mode.

### (1) Serial Communications Interface Mode (SMIF in SCMR = 0)



Bit	Symbol	Bit Name	Description	R/W
b1, b0	CKS[1:0]	Clock Select	b1 b0 0 0: P1 $\phi$ clock (n = 0)*1 0 1: P1 $\phi$ /4 clock (n = 1)*1 1 0: P1 $\phi$ /16 clock (n = 2)*1 1 1: P1 $\phi$ /64 clock (n = 3)*1	R/W*4
b2	MP	Multi-Processor Mode	(Valid only in asynchronous mode) 0: Multi-processor communications function is disabled 1: Multi-processor communications function is enabled	R/W*4
b3	STOP	Stop Bit Length	(Valid only in asynchronous mode) 0: 1 stop bit 1: 2 stop bits	R/W*4
b4	PM	Parity Mode	(Valid only when the PE bit is 1 in asynchronous mode) 0: Selects even parity 1: Selects odd parity	R/W*4
b5	PE	Parity Enable	(Valid only in asynchronous mode) <ul style="list-style-type: none"> <li>• When transmitting               <ul style="list-style-type: none"> <li>0: Parity bit addition is not performed</li> <li>1: The parity bit is added</li> </ul> </li> <li>• When receiving               <ul style="list-style-type: none"> <li>0: Parity bit checking is not performed</li> <li>1: The parity bit is checked</li> </ul> </li> </ul>	R/W*4
b6	CHR	Character Length	(Valid only in asynchronous mode) 0: Selects 8 bits as the data length*2 1: Selects 7 bits as the data length*3	R/W*4
b7	CM	Communications Mode	0: Asynchronous mode 1: Clock synchronous mode	R/W*4

Note 1. n is the decimal notation of the value of n in BRR (see section 15.2.9, Bit Rate Register (BRR)).

Note 2. In clock synchronous mode, this bit setting is invalid and a fixed data length of 8 bits is used.

Note 3. LSB-first is fixed and the MSB (bit 7) in TDR is not transmitted in transmission.

Note 4. Writable only when TE in SCR = 0 and RE in SCR = 0 (both serial transmission and reception are disabled).

#### CKS[1:0] Bits (Clock Select)

These bits select the clock source for the on-chip baud rate generator.

For the relation between the settings of these bits and the baud rate, see section 15.2.9, Bit Rate Register (BRR).

#### MP Bit (Multi-Processor Mode)

Disables/enables the multi-processor communications function. The settings of the PE bit and PM bit are invalid in multi-processor mode.

#### STOP Bit (Stop Bit Length)

Selects the stop bit length in transmission.

In reception, only the first stop bit is checked regardless of this bit setting. If the second stop bit is 0, it is treated as the start bit of the next transmit frame.

**PM Bit (Parity Mode)**

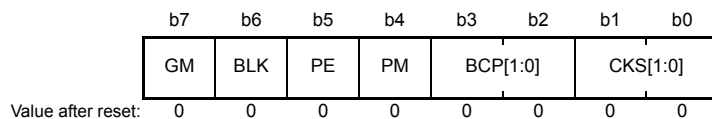
Selects the parity mode (even or odd) for transmission and reception.  
The setting of the PM bit is invalid in multi-processor mode.

**PE Bit (Parity Enable)**

When this bit is set to 1, the parity bit is added to transmit data, and the parity bit is checked in reception.  
Irrespective of the setting of the PE bit, the parity bit is not added or checked in multi-processor format.

**CHR Bit (Character Length)**

Selects the data length for transmission and reception.  
In clock synchronous mode, a fixed data length of 8 bits is used.

**(2) Smart Card Interface Mode (SMIF in SCMR = 1)**

Bit	Symbol	Bit Name	Description	R/W
b1, b0	CKS[1:0]	Clock Select	b1 b0 0 0: P1 $\phi$ clock (n = 0)*1 0 1: P1 $\phi$ /4 clock (n = 1)*1 1 0: P1 $\phi$ /16 clock (n = 2)*1 1 1: P1 $\phi$ /64 clock (n = 3)*1	R/W*3
b3, b2	BCP[1:0]	Base Clock Pulse	Selects the number of base clock cycles in combination with the BCP2 bit in SCMR. Setting values in BCP2 bit in SCMR and BCP[1:0] bits in SMR: BCP2 b3 b2 0 0 0: 93 clock cycles (S = 93)*2 0 0 1: 128 clock cycles (S = 128)*2 0 1 0: 186 clock cycles (S = 186)*2 0 1 1: 512 clock cycles (S = 512)*2 1 0 0: 32 clock cycles (S = 32)*2 (Initial value) 1 0 1: 64 clock cycles (S = 64)*2 1 1 0: 372 clock cycles (S = 372)*2 1 1 1: 256 clock cycles (S = 256)*2	R/W*3
b4	PM	Parity Mode	(Valid only when the PE bit is 1 in asynchronous mode) 0: Selects even parity 1: Selects odd parity	R/W*3
b5	PE	Parity Enable	When this bit is set to 1, a parity bit is added to data for transmission, and the parity of received data is checked. Set this bit to 1 in smart card interface mode.	R/W*3
b6	BLK	Block Transfer Mode	0: Normal mode operation 1: Block transfer mode operation	R/W*3
b7	GM	GSM Mode	0: Normal mode operation 1: GSM mode operation	R/W*3

Note 1. n is the decimal notation of the value of n in BRR (see section 15.2.9, Bit Rate Register (BRR)).

Note 2. S is the value of S in BRR (see section 15.2.9, Bit Rate Register (BRR)).

Note 3. Writable only when TE in SCR = 0 and RE in SCR = 0 (both serial transmission and reception are disabled).

**CKS[1:0] Bits (Clock Select)**

These bits select the clock source for the on-chip baud rate generator.

For the relationship between the settings of these bits and the baud rate, see section 15.2.9, Bit Rate Register (BRR).



**BCP[1:0] Bits (Base Clock Pulse)**

These bits select the number of base clock cycles in a 1-bit data transfer time in smart card interface mode.

Set these bits in combination with the BCP2 bit in SCMR.

For details, see section 15.6.4, Receive Data Sampling Timing and Reception Margin.

**PM Bit (Parity Mode)**

Selects the parity mode for transmission and reception (even or odd).

For details on the usage of this bit in smart card interface mode, see section 15.6.2, Data Format (Except in Block Transfer Mode).

**PE Bit (Parity Enable)**

Set the PE bit to 1.

The parity bit is added to transmit data before transmission, and the parity bit is checked in reception.

**BLK Bit (Block Transfer Mode)**

Setting this bit to 1 allows block transfer mode operation.

For details, see section 15.6.3, Block Transfer Mode.

**GM Bit (GSM Mode)**

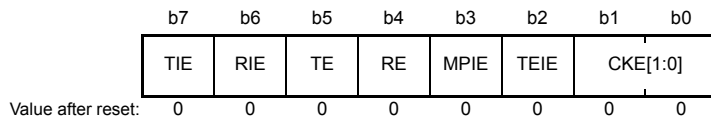
Setting this bit to 1 allows GSM mode operation.

In GSM mode, the SSR.TEND flag set timing is put forward to 11.0 etu (elementary time unit = 1-bit transfer time) from the start and the clock output control function is appended. For details, see section 15.6.6, Serial Data Transmission (Except in Block Transfer Mode) and section 15.6.8, Clock Output Control.

## 15.2.6 Serial Control Register (SCR)

Note: • Some bits in SCR have different functions in serial communications interface mode and smart card interface mode.

### (1) Serial Communications Interface Mode (SMIF in SCMR = 0)



Bit	Symbol	Bit Name	Description	R/W
b1, b0	CKE[1:0]	Clock Enable	(Asynchronous mode) b1 b0 0 0: On-chip baud rate generator The SCKn pin functions as general-purpose I/O port. 0 1: On-chip baud rate generator The clock with the same frequency as the bit rate is output from the SCKn pin. 1 x: External clock SEMR.ABCS bit is 0: The clock with a frequency 16 times the bit rate should be input from the SCKn pin. SEMR.ABCS bit is 1: The clock with a frequency eight times the bit rate should be input from the SCKn pin.  (Clock synchronous mode) b1 b0 0 x: Internal clock The SCKn pin functions as the clock output pin. 1 x: External clock The SCKn pin functions as the clock input pin.	R/W*1
b2	TEIE	Transmit End Interrupt Enable	0: A TEI interrupt request is disabled 1: A TEI interrupt request is enabled	R/W
b3	MPIE	Multi-Processor Interrupt Enable	(Valid in asynchronous mode when SMR.MP = 1) 0: Normal reception 1: When the data with the multi-processor bit set to 0 is received, the data is not read, and setting the status flags ORER and FER in SSR to 1 is disabled. When the data with the multi-processor bit set to 1 is received, the MPIE bit is automatically cleared to 0, and normal reception is resumed.	R/W
b4	RE	Receive Enable	0: Serial reception is disabled 1: Serial reception is enabled	R/W*2
b5	TE	Transmit Enable	0: Serial transmission is disabled 1: Serial transmission is enabled	R/W*2
b6	RIE	Receive Interrupt Enable	0: RXI and ERI interrupt requests are disabled 1: RXI and ERI interrupt requests are enabled	R/W
b7	TIE	Transmit Interrupt Enable	0: A TXI interrupt request is disabled 1: A TXI interrupt request is enabled	R/W

x: Don't care

Note 1. Writable only when TE = 0 and RE = 0.

Note 2. A 1 can be written only when TE = 0 and RE = 0, while the SMR.CM bit is 1. After setting TE or RE to 1, only 0 can be written in TE and RE. While the SMR.CM bit is 0, writing is enabled under any condition.

**CKE[1:0] Bits (Clock Enable)**

These bits select the clock source and SCKn pin function.

**TEIE Bit (Transmit End Interrupt Enable)**

Enables or disables a TEI interrupt request.

A TEI interrupt request is disabled by clearing the TEIE bit to 0.

**MPIE Bit (Multi-Processor Interrupt Enable)**

When this bit is set to 1 and the data with the multi-processor bit set to 0 is received, the data is not read and setting the status flags ORER and FER in SSR to 1 is disabled. When the data with the multi-processor bit set to 1 is received, the MPIE is automatically cleared to 0, and normal reception is resumed. For details, see section 15.4, Multi-Processor Communications Function.

When the receive data includes the MPB bit is SSR set to 0, the receive data is not transferred from the RSR to the RDR, a receive error is not detected, and setting the flags ORER and FER to 1 is disabled.

When the receive data includes the MPB bit set to 1, the MPB bit is set to 1, the MPIE bit is automatically cleared to 0, the RXI and ERI interrupt requests are enabled (if the RIE bit in SCR is set to 1), and setting the flags ORER and FER to 1 is enabled.

MPIE should be set to 0 if multi-processor communications function is not to be used.

**RE Bit (Receive Enable)**

Enables or disables serial reception.

When this bit is set to 1, serial reception is started by detecting the start bit in asynchronous mode or the synchronous clock input in clock synchronous mode. Note that SMR should be set prior to setting the RE bit to 1 in order to designate the reception format.

Even if reception is halted by clearing the RE bit to 0, the ORER, FER, and PER flags in SSR are not affected and the previous value is retained.

**TE Bit (Transmit Enable)**

Enables or disables serial transmission.

When this bit is set to 1, serial transmission is started by writing transmit data to TDR. Note that SMR should be set prior to setting the TE bit to 1 in order to designate the transmission format.

**RIE Bit (Receive Interrupt Enable)**

Enables or disables RXI and ERI interrupt requests.

An RXI interrupt request is disabled by clearing the RIE bit to 0.

An ERI interrupt request can be cancelled by reading 1 from the ORER, FER, or PER flag in SSR and then clearing the flag to 0, or clearing the RIE bit to 0.

**TIE Bit (Transmit Interrupt Enable)**

Enables or disables notification of a TXI interrupt request.

Notification of a TXI interrupt request is disabled by clearing the TIE bit to 0.

## (2) Smart Card Interface Mode (SMIF in SCMR = 1)

b7	b6	b5	b4	b3	b2	b1	b0
TIE	RIE	TE	RE	MPIE	TEIE	CKE[1:0]	
Value after reset:	0	0	0	0	0	0	0

Bit	Symbol	Bit Name	Description	R/W
b1, b0	CKE[1:0]	Clock Enable	<ul style="list-style-type: none"> <li>When GM in SMR = 0           <ul style="list-style-type: none"> <li>b1 b0</li> <li>0 0: Output disabled (The SCKn pin is available for use as an I/O port in accord with the general-purpose I/O port settings.)</li> <li>0 1: Clock output</li> <li>1 x: (Setting prohibited)</li> </ul> </li> <li>When GM in SMR = 1           <ul style="list-style-type: none"> <li>b1 b0</li> <li>0 0: Output fixed low</li> <li>x 1: Clock output</li> <li>1 0: Output fixed high</li> </ul> </li> </ul>	R/W*1
b2	TEIE	Transmit End Interrupt Enable	This bit should be 0 in smart card interface mode.	R/W
b3	MPIE	Multi-Processor Interrupt Enable	This bit should be 0 in smart card interface mode.	R/W
b4	RE	Receive Enable	0: Serial reception is disabled 1: Serial reception is enabled	R/W*2
b5	TE	Transmit Enable	0: Serial transmission is disabled 1: Serial transmission is enabled	R/W*2
b6	RIE	Receive Interrupt Enable	0: RXI and ERI interrupt requests are disabled 1: RXI and ERI interrupt requests are enabled	R/W
b7	TIE	Transmit Interrupt Enable	0: A TXI interrupt request is disabled 1: A TXI interrupt request is enabled	R/W

x: Don't care

Note 1. Writable only when TE = 0 and RE = 0.

Note 2. A 1 can be written only when TE = 0 and RE = 0, while the SMR.CM bit is 1. After setting TE or RE to 1, only 0 can be written in TE and RE. While the SMR.CM bit is 0, writing is enabled under any condition.

For details on interrupt requests, see section 15.8, Interrupt Sources.

**CKE[1:0] Bits (Clock Enable)**

These bits control the clock output from the SCKn pin.

In GSM mode, clock output can be dynamically switched. For details, see section 15.6.8, Clock Output Control.

**TEIE Bit (Transmit End Interrupt Enable)**

This bit should be 0 in smart card interface mode.

**MPIE Bit (Multi-Processor Interrupt Enable)**

This bit should be 0 in smart card interface mode.

**RE Bit (Receive Enable)**

Enables or disables serial reception.

When this bit is set to 1, serial reception is started by detecting the start bit. Note that SMR should be set prior to setting the RE bit to 1 in order to designate the reception format.

Even if reception is halted by clearing the RE bit to 0, the ORER, FER, and PER flags in SSR are not affected and the previous value is retained.

**TE Bit (Transmit Enable)**

Enables or disables serial transmission.

When this bit is set to 1, serial transmission is started by writing transmit data to TDR. Note that SMR should be set prior to setting the TE bit to 1 in order to designate the transmission format.

**RIE Bit (Receive Interrupt Enable)**

Enables or disables RXI and ERI interrupt requests.

An RXI interrupt request is disabled by clearing the RIE bit to 0.

An ERI interrupt request can be cancelled by reading 1 from the ORER, FER, or PER flag in SSR and then clearing the flag to 0, or clearing the RIE bit to 0.

**TIE Bit (Transmit Interrupt Enable)**

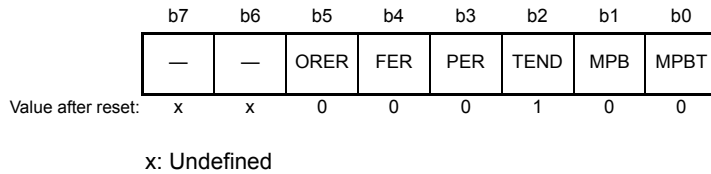
Enables or disables notification of a TXI interrupt request.

Notification of a TXI interrupt request is disabled by clearing the TIE bit to 0.

## 15.2.7 Serial Status Register (SSR)

Note: • Some bits in SSR have different functions in serial communications interface mode and smart card interface mode.

### (1) Serial Communications Interface Mode (SMIF in SCMR = 0)



Bit	Symbol	Bit Name	Description	R/W
b0	MPBT	Multi-Processor Bit Transfer	Sets the multi-processor bit for adding to the transmission frame 0: Data transmission cycles 1: ID transmission cycles	R/W
b1	MPB	Multi-Processor	Value of the multi-processor bit in the reception frame 0: Data transmission cycles 1: ID transmission cycles	R
b2	TEND	Transmit End Flag	0: A character is being transmitted. 1: Character transfer has been completed.	R
b3	PER	Parity Error Flag	0: No parity error occurred 1: A parity error has occurred	R/(W) *1
b4	FER	Framing Error Flag	0: No framing error occurred 1: A framing error has occurred	R/(W) *1
b5	ORER	Overrun Error Flag	0: No overrun error occurred 1: An overrun error has occurred	R/(W) *1
b7, b6	—	Reserved	The read value is undefined. The write value should be 1.	R

Note 1. Only 0 can be written to this bit, to clear the flag.

#### MPB Bit (Multi-Processor)

Holds the value of the multi-processor bit in the reception frame. This bit does not change when the RE bit in SCR is 0.

#### TEND Flag (Transmission End Flag)

Indicates completion of transmission.

[Setting conditions]

- Clearing of the SCR.TE bit to 0 (disabling serial transmission operations)  
When the SCR.TE bit is changed from 0 to 1, the TEND flag is not affected and retains the value 1.
- The TDR is not updated at the time of transmission of the tail-end bit of a character being transmitted

[Clearing condition]

- When data for transmission are written to the TDR while the SCR.TE is 1  
When the TEND flag is cleared in response to writing of data for transmission to the TDR, dummy read the TEND flag before return from interrupt exception processing for TEI interrupt requests when the latter are enabled.

**PER Bit (Parity Error Flag)**

Indicates that a parity error has occurred during reception in asynchronous mode and the reception ends abnormally.

[Setting condition]

- When a parity error is detected during reception  
Although receive data when the parity error occurs is transferred to RDR, no RXI interrupt request occurs. Note that when the PER flag is being set to 1, the subsequent receive data is not transferred to RDR.

[Clearing condition]

- When 0 is written to PER after reading PER = 1 (after writing 0 to it, read the PER bit to check that it has actually been cleared to 0.)  
Even when the RE bit in SCR is cleared to 0 (which indicates that serial reception is disabled), the PER flag is not affected and retains its previous value.

**FER Bit (Framing Error Flag)**

Indicates that a framing error has occurred during reception in asynchronous mode and the reception ends abnormally.

[Setting condition]

- When the stop bit is 0  
In 2-stop-bit mode, only the first stop bit is checked whether it is 1 but the second stop bit is not checked. Note that although receive data when the framing error occurs is transferred to RDR, no RXI interrupt request occurs. In addition, when the FER flag is being set to 1, the subsequent receive data is not transferred to RDR.

[Clearing condition]

- When 0 is written to FER after reading FER = 1 (after writing 0 to it, read the FER bit to check that it has actually been cleared to 0.)  
Even when the RE bit in SCR is cleared to 0, the FER flag is not affected and retains its previous value.

**ORER Bit (Overrun Error Flag)**

Indicates that an overrun error has occurred during reception and the reception ends abnormally.

[Setting condition]

- When the next data is received before receive data is read from RDR  
In RDR, receive data prior to an overrun error occurrence is retained, but data received after the overrun error occurrence is lost. When the ORER flag is set to 1, subsequent serial reception cannot be performed. Note that, in clock synchronous mode, serial transmission also cannot continue.

[Clearing condition]

- When a 0 is written to ORER after reading ORER = 1 (after writing a 0 to it, read the ORER bit to check that it has actually been cleared to 0.)  
Even when the RE bit in SCR is cleared to 0, the ORER flag is not affected and retains its previous value.

## (2) Smart Card Interface Mode (SMIF in SCMR = 1)

	b7	b6	b5	b4	b3	b2	b1	b0
	—	—	ORER	ERS	PER	TEND	MPB	MPBT
Value after reset:	x	x	0	0	0	1	0	0

x: Undefined

Bit	Symbol	Bit Name	Description	R/W
b0	MPBT	Multi-Processor Bit Transfer	This bit should be set to 0 in smart card interface mode.	R/W
b1	MPB	Multi-Processor	This bit is not used in smart card interface mode. It should be set to 0.	R
b2	TEND	Transmit End Flag	0: A character is being transmitted. 1: Character transfer has been completed.	R
b3	PER	Parity Error Flag	0: No parity error occurred 1: A parity error has occurred	R/(W) *1
b4	ERS	Error Signal Status Flag	0: Low error signal not responded 1: Low error signal responded	R/(W) *1
b5	ORER	Overflow Error Flag	0: No overflow error occurred 1: An overflow error has occurred	R/(W) *1
b7, b6	—	Reserved	The read value is undefined. The write value should be 1.	R

Note 1. Only 0 can be written to this bit, to clear the flag.

**MPBT Bit (Multi-Processor Bit Transfer)**

This bit should be set to 0 in smart card interface mode.

**MPB Bit (Multi-Processor)**

This bit is not used in smart card interface mode. It should be set to 0.

**TEND Flag (Transmission End Flag)**

With no error signal from the receiving side, this bit is set to 1 when further data for transfer is ready to be transferred to the TDR register.

[Setting conditions]

- When SCR.TE bit = 0 (disabling serial transmission operations)  
When the SCR.TE bit is changed from 0 to 1, the TEND flag is not affected and retains the value 1.
- When a specified period has elapsed after the latest transmission of one byte, the ERS flag is 0, and the TDR register is not updated  
The set timing is determined by register settings as listed below.  
When SMR.GM = 0 and SMR.BLK = 0, 12.5 etu after the start of transmission  
When SMR.GM = 0 and SMR.BLK = 1, 11.5 etu after the start of transmission  
When SMR.GM = 1 and SMR.BLK = 0, 11.0 etu after the start of transmission  
When SMR.GM = 1 and SMR.BLK = 1, 11.0 etu after the start of transmission

[Clearing condition]

- When data for transmission are written to the TDR while the SCR.TE is 1



**PER Flag (Parity Error Flag)**

Indicates that a parity error has occurred during reception in asynchronous mode and the reception ends abnormally.

[Setting condition]

- When a parity error is detected during reception

Although receive data when the parity error occurs is transferred to RDR, no RXI interrupt request occurs. Note that when the PER flag is being set to 1, the subsequent receive data is not transferred to RDR.

[Clearing condition]

- When 0 is written to PER after reading PER = 1 (After writing 0 to it, read the PER bit to check that it has actually been cleared to 0.)

Even when the RE bit in SCR is cleared to 0 (which indicates that serial reception is disabled), the PER flag is not affected and retains its previous value.

**ERS Flag (Error Signal Status Flag)**

[Setting condition]

- When a low error signal is sampled

[Clearing condition]

- When 0 is written to ERS after reading ERS = 1

**ORER Flag (Overrun Error Flag)**

Indicates that an overrun error has occurred during reception and the reception ends abnormally.

[Setting condition]

- When the next data is received before receive data is read from RDR

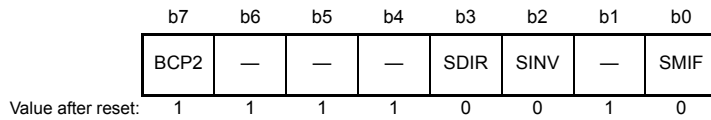
In RDR, the receive data prior to an overrun error occurrence is retained, but data received following the overrun error occurrence is lost. When the ORER flag is set to 1, subsequent serial reception cannot be performed.

[Clearing condition]

- When 0 is written to ORER after reading ORER = 1 (After writing 0 to it, read the ORER bit to check that it has actually been cleared to 0.)

Even when the RE bit in SCR is cleared to 0, the ORER flag is not affected and retains its previous value.

### 15.2.8 Smart Card Mode Register (SCMR)



Bit	Symbol	Bit Name	Description	R/W
b0	SMIF	Smart Card Interface Mode Select	0: Serial communications interface mode 1: Smart card interface mode	R/W*1
b1	—	Reserved	This bit is read as 1. The write value should be 1.	R
b2	SINV	Transmitted/Received Data Invert	0: TDR contents are transmitted as they are. Receive data is stored as it is in RDR. 1: TDR contents are inverted before being transmitted. Receive data is stored in inverted form in RDR.	R/W*1
b3	SDIR	Transmitted/Received Data Transfer Direction	0: Transfer with LSB-first 1: Transfer with MSB-first	R/W*1
b6 to b4	—	Reserved	This bit is read as 1. The write value should be 1.	R
b7	BCP2	Base Clock Pulse 2	Selects the number of base clock cycles in combination with the SMR.BCP[1:0] bits. Setting values in the SCMR.BCP2 bit and SMR.BCP[1:0] bits BCP2 BCP1 BCP0 0 0 0: 93 clock cycles (S = 93)*2 0 0 1: 128 clock cycles (S = 128)*2 0 1 0: 186 clock cycles (S = 186)*2 0 1 1: 512 clock cycles (S = 512)*2 1 0 0: 32 clock cycles (S = 32)*2 (Initial Value) 1 0 1: 64 clock cycles (S = 64)*2 1 1 0: 372 clock cycles (S = 372)*2 1 1 1: 256 clock cycles (S = 256)*2	R/W*1

Note 1. Writable only when TE in SCR = 0 and RE in SCR = 0 (both serial transmission and reception are disabled).

Note 2. S is the value of S in BRR (see section 15.2.9, Bit Rate Register (BRR)).

#### SMIF Bit (Smart Card Interface Mode Select)

When this bit is set to 1, smart card interface mode is selected.

When this bit is set to 0, asynchronous or clock synchronous mode is selected.

#### SINV Bit (Transmitted/Received Data Invert)

Inverts the transmit/receive data logic level. This bit does not affect the logic level of the parity bit. To invert the parity bit, invert the PM bit in SMR.

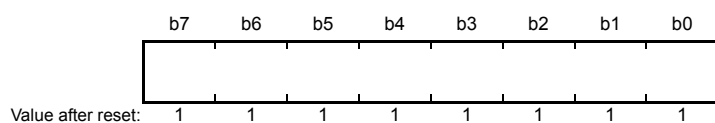
#### SDIR Bit (Transmitted/Received Data Transfer Direction)

Selects the serial/parallel conversion format.

#### BCP2 Bit (Base Clock Pulse 2)

Selects the number of base clock cycles in a 1-bit data transfer time in smart card interface mode. Set this bit in combination with the SMR.BCP[1:0] bits.

### 15.2.9 Bit Rate Register (BRR)



BRR is an 8-bit register that adjusts the bit rate.

As each SCI channel has independent baud-rate generator control, different bit rates can be set for each. Table 15.4 lists the relationships between the setting (N) in the BRR and the bit rate (B) for normal asynchronous mode, multi-processor transfer, clock synchronous mode, and smart card interface mode.

The initial value of BRR is FFh.

BRR can be read from by the CPU at all times, but it can be written to only when the TE and RE bits in SCR are 0.

**Table 15.4 Relationships between N Setting in BRR and Bit Rate B**

Mode	ABCS Bit in SEMR	BRR Setting	Error
Asynchronous, multi-processor transfer	0	$N = \frac{P1\phi \times 10^6}{64 \times 2^{2n-1} \times B} - 1$	$\text{Error (\%)} = \left\{ \frac{P1\phi \times 10^6}{B \times 64 \times 2^{2n-1} \times (N+1)} - 1 \right\} \times 100$
	1	$N = \frac{P1\phi \times 10^6}{32 \times 2^{2n-1} \times B} - 1$	$\text{Error (\%)} = \left\{ \frac{P1\phi \times 10^6}{B \times 32 \times 2^{2n-1} \times (N+1)} - 1 \right\} \times 100$
Clock synchronous		$N = \frac{P1\phi \times 10^6}{8 \times 2^{2n-1} \times B} - 1$	
Smart card interface		$N = \frac{P1\phi \times 10^6}{S \times 2^{2n+1} \times B} - 1$	$\text{Error (\%)} = \left\{ \frac{P1\phi \times 10^6}{B \times S \times 2^{2n+1} \times (N+1)} - 1 \right\} \times 100$

B: Bit rate (bps)

N: BRR setting for baud rate generator ( $0 \leq N \leq 255$ )

P1 $\phi$ : Operating frequency (MHz)

n and S: Determined by the SMR setting listed in the following table.

**Table 15.5 Clock Source Settings**

SMR Setting		
CKS[1:0] Bits	Clock Source	n
0 0	P1 $\phi$ clock	0
0 1	P1 $\phi$ /4 clock	1
1 0	P1 $\phi$ /16 clock	2
1 1	P1 $\phi$ /64 clock	3

**Table 15.6 Base Clock Settings in Smart Card Interface Mode**

SCMR Setting	SMR Setting	Base Clock Cycles for	
BCP2 Bit	BCP[1:0] Bits	One-bit Period	S
0	0 0	93 clock cycles	93
0	0 1	128 clock cycles	128
0	1 0	186 clock cycles	186
0	1 1	512 clock cycles	512
1	0 0	32 clock cycles	32
1	0 1	64 clock cycles	64
1	1 0	372 clock cycles	372
1	1 1	256 clock cycles	256

Table 15.7 lists sample N settings in BRR in normal asynchronous mode. Table 15.8 lists the maximum bit rate settable for each operating frequency. Examples of BRR (N) settings in clock synchronous mode are listed in Table 15.10. Examples of BRR (N) settings in smart card interface mode are listed in Table 15.12. In smart card interface mode, the number of base clock cycles S in a 1-bit data transfer time can be selected. For details, see section 15.6.4, Receive Data Sampling Timing and Reception Margin. Table 15.9 and Table 15.11 list the maximum bit rates with external clock input.

When the asynchronous mode base clock select bit (ABCS) in the serial extended mode register (SEMR) is set to 1 in asynchronous mode, the bit rate is two times that of listed in Table 15.7.

**Table 15.7 Examples of BRR Settings for Various Bit Rates (Asynchronous Mode)**

Bit Rate (bps)	Operating Frequency P1 $\phi$ (MHz)								
	50			64			66.67		
	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
110	3	221	-0.02						
150	3	162	-0.15	3	207	0.16	3	216	0.01
300	3	80	0.47	3	103	0.16	3	108	-0.45
600	2	162	-0.15	2	207	0.16	2	216	0.01
1200	2	80	0.47	2	103	0.16	2	108	-0.45
2400	1	162	-0.15	1	207	0.16	1	216	0.01
4800	1	80	0.47	1	103	0.16	1	108	-0.45
9600	0	162	-0.15	1	51	0.16	0	216	0.01
19200	0	80	0.47	0	103	0.16	0	108	-0.45
31250	0	49	0.00	0	63	0.00	0	66	-0.50
38400	0	40	-0.76	0	51	0.16	0	53	0.47

[Legend]

Space: Setting prohibited.

-: Can be set, but there will be error.

Note: • This is an example when the ABCS bit in SEMR is 0.  
When the ABCS bit is set to 1, the bit rate is two times.

**Table 15.8 Maximum Bit Rate for Each Operating Frequency (Asynchronous Mode)**

P1 $\phi$ (MHz)	Maximum Bit Rate(bps)	n	N
50	1562500	0	0
64	2000000	0	0
66.67	2083333	0	0

Note: • When the ABCS bit in SEMR is set to 1, the bit rate is two times.

**Table 15.9 Maximum Bit Rate with External Clock Input (Asynchronous Mode)**

P1 $\phi$ (MHz)	External Input Clock(MHz)	Maximum Bit Rate(bps)	
		SEMR.ABCS bit = 0	SEMR.ABCS bit = 1
50	12.5	781250	1562500
64	16	1000000	2000000
66.67	16.667	1041667	2083333

**Table 15.10 BRR Settings for Various Bit Rates (Clock Synchronous Mode)**

Bit Rate (bps)	Operating Frequency P1 $\phi$ (MHz)					
	50		64		66.67	
	n	N	n	N	n	N
110						
250						
500						
1k	3	194	3	249		
2.5k	3	77	3	99	3	103
5k	2	155	2	199	2	207
10k	2	77	2	99	2	103
25k	1	124	1	159	1	166
50k	1	62	1	79	1	82
100k	0	124	0	159	0	166
250k	0	49	0	63	0	66
500k	0	24	0	31	0	32
1M	—	—	0	15	0	16
2.5M	0	4	—	—	—	—
5M	—	—	—	—	—	—

Space: Setting prohibited.

—: Can be set, but there will be error.

**Table 15.11 Maximum Bit Rate with External Clock Input (Clock Synchronous Mode)**

P1 $\phi$ (MHz)	External Input Clock (MHz)	Maximum Bit Rate (bps)
50	8.3333	8333333.3
64	10.6667	10666666.7
66.67	11.1111	11111100

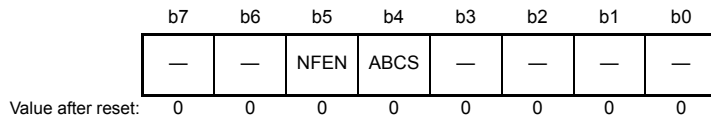
**Table 15.12 BRR Settings for Various Bit Rates (Smart Card Interface Mode, n = 0, S = 372)**

Bit Rate (bps)	P1 $\phi$ (MHz)								
	50			64			66.67		
	n	N	Error (%)	n	N	Error (%)	n	N	Error (%)
9600	0	6	0.01	0	8	0.44	0	8	3.72

**Table 15.13 Maximum Bit Rate for Each Operating Frequency (Smart Card Interface Mode, S = 372)**

P1 $\phi$ (MHz)	Maximum Bit Rate (bps)	n	N
50	67204	0	0
64	86022	0	0
66.67	89610	0	0

### 15.2.10 Serial Extended Mode Register (SEMR)



Bit	Symbol	Bit Name	Description	R/W
b3 to b0	—	Reserved	These bits are read as 0. The write value should be 0.	R
b4	ABCS	Asynchronous Mode Base Clock Select	(Valid only in asynchronous mode) 0: Selects 16 base clock cycles for 1-bit period 1: Selects 8 base clock cycles for 1-bit period	R/W*1
b5	NFEN	Digital Noise Filter Function Enable	(In asynchronous mode) 0: Noise cancellation function for the RXDn input signal is disabled. 1: Noise cancellation function for the RXDn input signal is enabled. The NFEN bit should be 0 in any mode other than above.	R/W*1
b7, b6	—	Reserved	These bits are read as 0. The write value should be 0.	R

Note 1. Writable only when TE in SCR = 0 and RE in SCR = 0 (both serial transmission and reception are disabled).

#### NFEN Bit (Digital Noise Filter Function Enable)

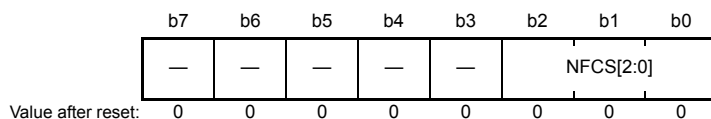
This bit enables or disables the digital noise filter function.

When the function is enabled, noise cancellation is applied to the RXDn input signal in asynchronous mode.

In any mode other than above, set the NFEN bit to 0 to disable the digital noise filter function.

When the function is disabled, input signals are transferred as is, as internal signals.

### 15.2.11 Noise Filter Setting Register (SNFR)



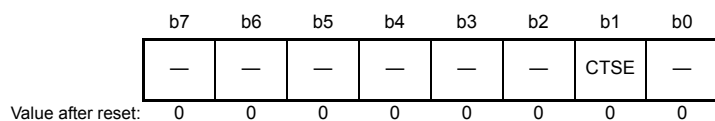
Bit	Symbol	Bit Name	Description	R/W
b2 to b0	NFCS[2:0]	Noise Filter Clock Select	In asynchronous mode, the standard setting for the base clock is as follows. b2 b0 0 0 0: The clock signal divided by 1 is used with the noise filter.  Other values: Do not make settings other than those listed above.	R/W*1
b7 to b3	—	Reserved	These bits are read as 0. The write value should be 0.	R

Note 1. Writing to these bits is only possible when the RE and TE bits in the SCR are 0. (both serial transmission and reception are disabled.)

#### NFCS[2:0] Bits (Noise Filter Clock Select)

These bits select the sampling clock for the digital noise filter. To use the noise filter in asynchronous mode, set these bits to 000b.

### 15.2.12 Extended Function Control Register (SECR)



Bit	Symbol	Bit Name	Description	R/W
b0	—	Reserved	This bit is read as 0. The write value should be 0.	R
b1	CTSE	CTS Enable	0: CTS pin function is disabled (RTS output function is enabled). 1: CTS pin function is enabled	R/W*1
b7 to b2	—	Reserved	These bits are read as 0. The write value should be 0.	R

Note 1. Writing to this bit is only possible when the RE and TE bits in the SCR are 0 (both serial transmission and reception are disabled).

SECR is used to select the extension settings in asynchronous and clock-synchronous modes.

#### CTSE Bit (CTS Enable)

Set this bit to 1 if the CTS control signal is used for control of transmission and reception. The RTS signal is output when this bit is set to 0. Set this bit to 0 in smart card interface mode.



### 15.3 Operation in Asynchronous Mode

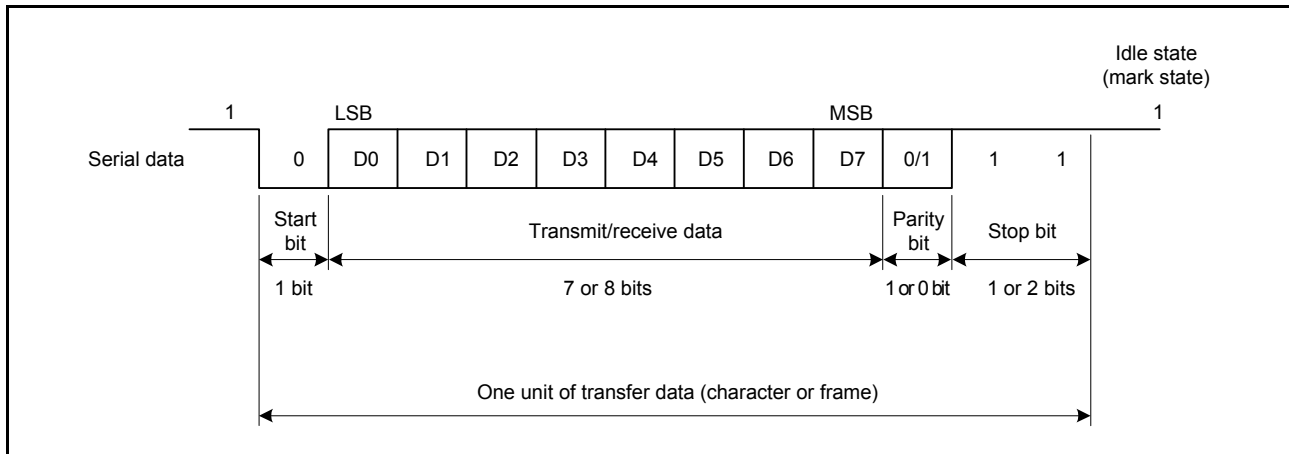
Figure 15.2 shows the general format for asynchronous serial communications.

One frame consists of a start bit (low level), transmit/receive data, a parity bit, and stop bits (high level).

In asynchronous serial communications, the communications line is usually held in the mark state (high level).

The SCI monitors the communications line, and when it goes to the space state (low level), recognizes a start bit and starts serial communications.

Inside the SCI, the transmitter and receiver are independent units, enabling full-duplex communications. Both the transmitter and the receiver also have a double-buffered structure, so that data can be read or written during transmission or reception, enabling continuous data transmission and reception.



**Figure 15.2 Data Format in Asynchronous Serial Communications  
(Example with 8-Bit Data, Parity, Two Stop Bits)**

#### 15.3.1 Serial Data Transfer Format

Table 15.14 lists the serial data transfer formats that can be used in asynchronous mode.

Any of 12 transfer formats can be selected according to the SMR setting. For details of multi-processor function, see section 15.4, Multi-Processor Communications Function.

Table 15.14 Serial Transfer Formats (Asynchronous Mode)

SMR Setting				Serial Transfer Format and Frame Length												
CHR	PE	MP	STOP	1	2	3	4	5	6	7	8	9	10	11	12	
0	0	0	0	S	8-bit data								STOP			
0	0	0	1	S	8-bit data								STOP	STOP		
0	1	0	0	S	8-bit data								P	STOP		
0	1	0	1	S	8-bit data								P	STOP	STOP	
1	0	0	0	S	7-bit data							STOP				
1	0	0	1	S	7-bit data							STOP	STOP			
1	1	0	0	S	7-bit data							P	STOP			
1	1	0	1	S	7-bit data							P	STOP	STOP		
0	—	1	0	S	8-bit data								MPB	STOP		
0	—	1	1	S	8-bit data								MPB	STOP	STOP	
1	—	1	0	S	7-bit data							MPB	STOP			
1	—	1	1	S	7-bit data							MPB	STOP	STOP		

S: Start bit

STOP: Stop bit

P: Parity bit

MPB: Multi-processor bit

### 15.3.2 Receive Data Sampling Timing and Reception Margin in Asynchronous Mode

In asynchronous mode, the SCI operates on a base clock with a frequency of 16 times\*1 the bit rate.

In reception, the SCI samples the falling edge of the start bit using the base clock, and performs internal synchronization. Since receive data is sampled at the rising edge of the 8th pulse\*1 of the base clock, data is latched at the middle of each bit, as shown in Figure 15.3. Thus the reception margin in asynchronous mode is determined by formula (1) below.

$$M = \left| \left( 0.5 - \frac{1}{2N} \right) - (L - 0.5) F - \frac{|D - 0.5|}{N} (1 + F) \right| \times 100 [\%] \quad \cdots \text{Formula (1)}$$

M: Reception margin

N: Ratio of bit rate to clock (N = 16 when ABCS in SEMR = 0, N = 8 when ABCS in SEMR = 1)

D: Duty cycle of clock (D = 0.5 to 1.0)

L: Frame length (L = 9 to 12)

F: Absolute value of clock frequency deviation

Assuming values of F = 0 and D = 0.5 in formula (1), the reception margin is determined by the formula below.

$$M = \{0.5 - 1/(2 \times 16)\} \times 100 (\%) = 46.875\%$$

However, this is only the computed value, and a margin of 20% to 30% should be allowed in system design.

Note 1. This is an example when the ABCS bit in SEMR is 0. When the ABCS bit is 1, a frequency of 8 times the bit rate is used as a base clock and receive data is sampled at the rising edge of the 4th pulse of the base clock

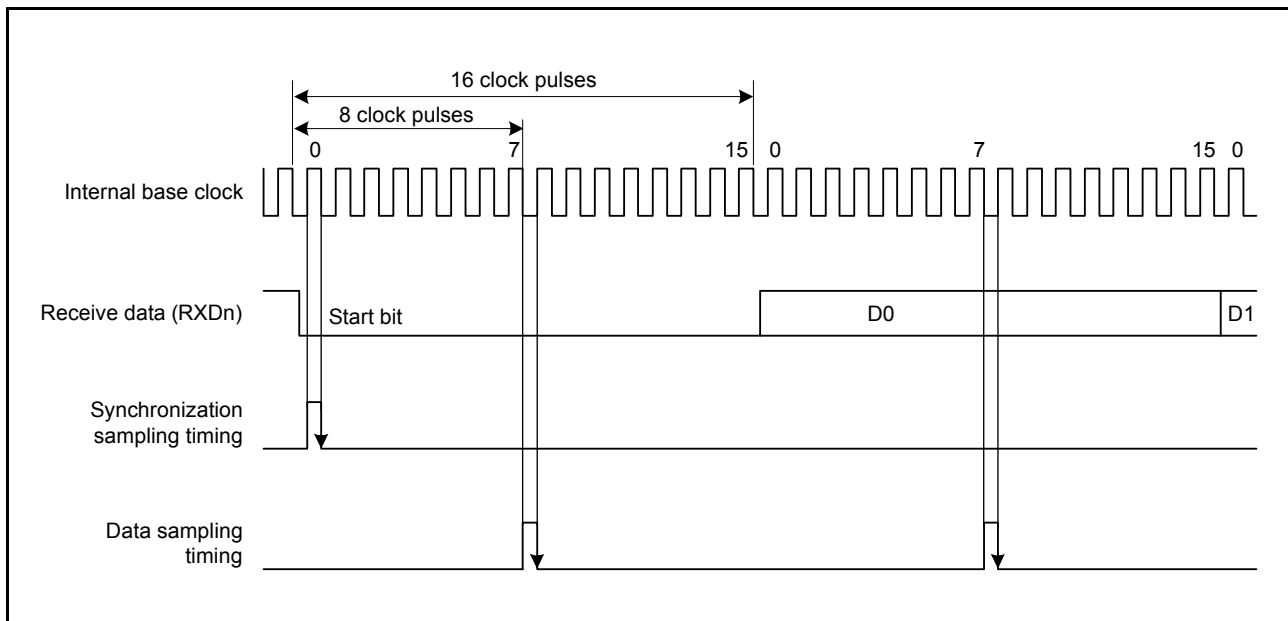


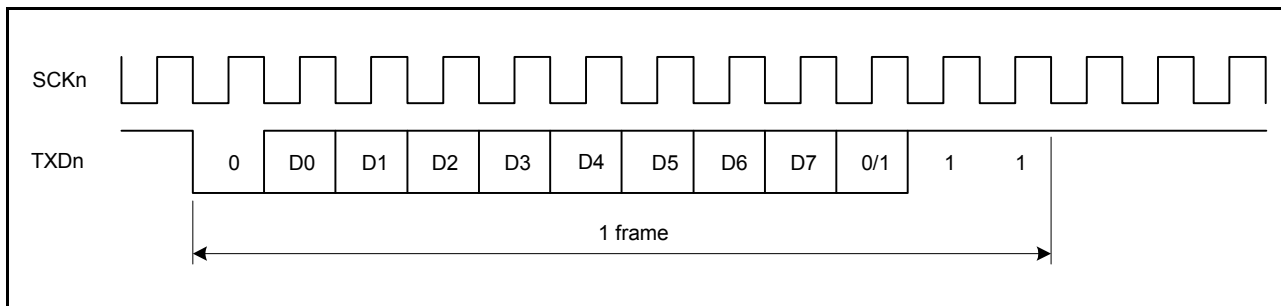
Figure 15.3 Receive Data Sampling Timing in Asynchronous Mode

### 15.3.3 Clock

Either an internal clock generated by the on-chip baud rate generator or an external clock input to the SCKn pin can be selected as the SCI's transfer clock, according to the setting of the CM bit in SMR and the CKE[1:0] bits in SCR.

When an external clock is input to the SCKn pin, the clock frequency should be 16 times the bit rate (when ABCS in SEMR = 0) and 8 times the bit rate (when ABCS in SEMR = 1).

When the SCI is operated on an internal clock, the clock can be output from the SCKn pin. The frequency of the clock output in this case is equal to the bit rate, and the phase is such that the rising edge of the clock is in the middle of the transmit data, as shown in Figure 15.4.



**Figure 15.4** Phase Relationship between Output Clock and Transmit Data  
(Asynchronous Mode: SMR.CHR = 0, PE = 1, MP = 0, STOP = 1)

### 15.3.4 CTS and RTS Functions

The CTS function is the use of input on the CTSn# pin in transmission control. Setting the SECR.CTSE bit to 1 enables the CTS function. When the CTS function is enabled, placing the low level on the CTSn# pin causes transmission to start.

Applying the low level to the CTS# pin while transmission is in progress does not affect transmission of the current frame, which continues.

In the RTS function, by using the function of output on the RTSn# pin, a low level is output when reception becomes possible. Conditions for output of the low and high level are shown below.

[Conditions for low-level output]

Satisfaction of all conditions listed below

- The value of the RE bit in the SCR is 1
- Reception is not in progress
- There are no received data yet to be read
- The ORER, FER, and PER flags in the SSR are all 0

[Condition for high-level output]

- Any of the conditions for the low level not being satisfied

### 15.3.5 SCI Initialization (Asynchronous Mode)

Before transmitting and receiving data, start by writing the initial value “00h” to SCR and then continue through the procedure for SCI given in the sample flowchart (Figure 15.5). Whenever the operating mode or transfer format is changed, SCR must be initialized before the change is made.

When the external clock is used in asynchronous mode, ensure that the clock signal is supplied even during initialization. Note that clearing the SCR.RE bit to 0 initializes neither the ORER, FER, and PER flags in SSR nor RDR.

Moreover, note that switching the value of the SCR.TE bit from 1 to 0 or 0 to 1 while the SCR.TIE bit is 1 leads to the generation of a TXI interrupt request.

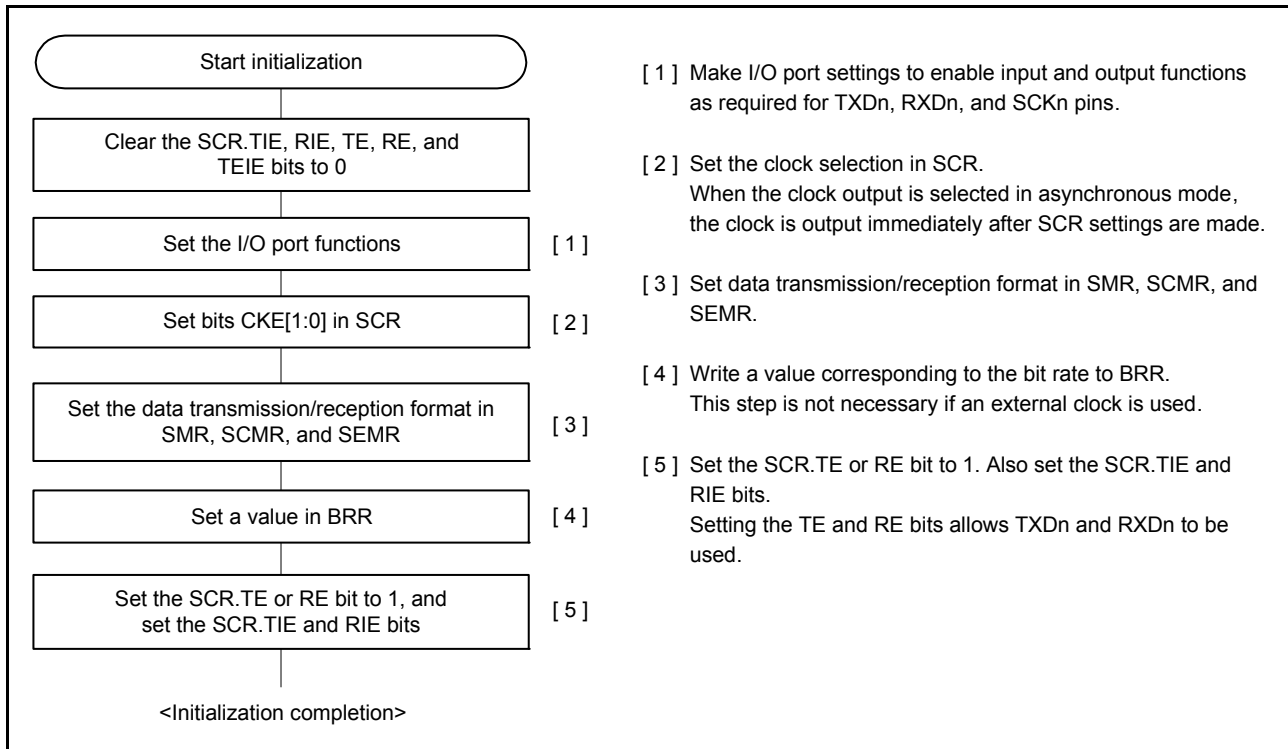


Figure 15.5 Sample SCI Initialization Flowchart (Asynchronous Mode)

### 15.3.6 Serial Data Transmission (Asynchronous Mode)

Figure 15.6 shows an example of the operation for serial transmission in asynchronous mode. In serial transmission, the SCI operates as described below.

1. The SCI transfers data from TDR to TSR when data is written to TDR in the TXI interrupt processing routine. The TXI interrupt request at the beginning of transmission is generated when the TE bit in SCR is set to 1 after the TIE bit in SCR is set to 1 or when these two bits are set to 1 simultaneously by a single instruction.
2. Transmission starts after the CTSE bit in SECR is set to 0 (disabling the CTS function) and a low level on the CTS# pin causes data transfer from TDR to TSR. If the TIE bit in SCR is 1 at this time, a TXI interrupt request is generated. Continuous transmission is obtainable by writing the next data for transmission to TDR in the TXI interrupt processing routine before transmission of the current data for transmission is completed. When TEI interrupt requests are in use, set the SCR.TIE bit to 0 (disabling TXI requests) and the SCR.TEIE bit to 1 (enabling TEI requests) after the last of the data to be transmitted are written to the TDR from the processing routine for TXI requests.
3. Data is sent from the TXDn pin in the following order: start bit, transmit data, parity bit or multi-processor bit (may be omitted depending on the format), and stop bit.
4. The SCI checks for updating of (writing to) TDR at the time of stop bit output.
5. When TDR is updated, setting of the CTSE bit in SECR to 0 (CTS function disabled) or a low level input on the CTSn# pin causes the next transfer of the next data for transmission from TDR to TSR and sending of the stop bit, after which serial transmission of the next frame starts.
6. If TDR is not updated, the TEND flag in SSR is set to 1, the stop bit is sent, and then the mark state is entered in which 1 is output. If the TEIE flag in SCR is 1 at this time, the TEND flag in SSR is set to 1 and a TEI interrupt request is generated.

Figure 15.7 shows a sample flowchart for serial transmission in asynchronous mode.

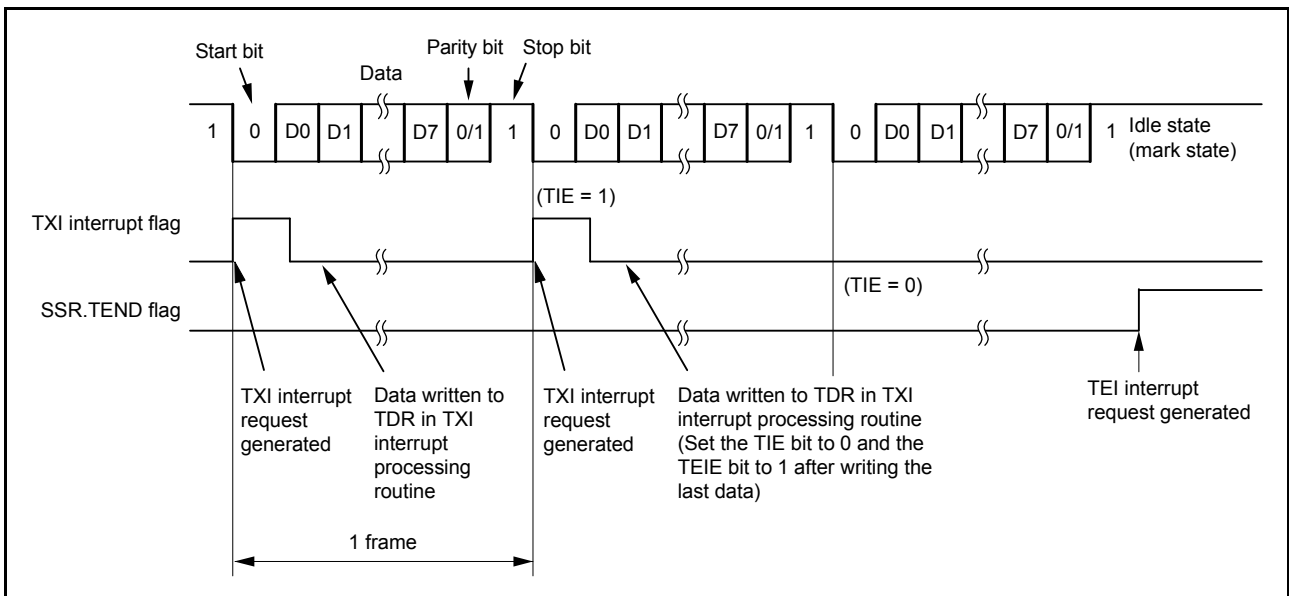


Figure 15.6 Example of Operation for Serial Transmission in Asynchronous Mode (from the Middle of Transmission until Transmission Completion) (Example with 8-Bit Data, Parity, One Stop Bit)

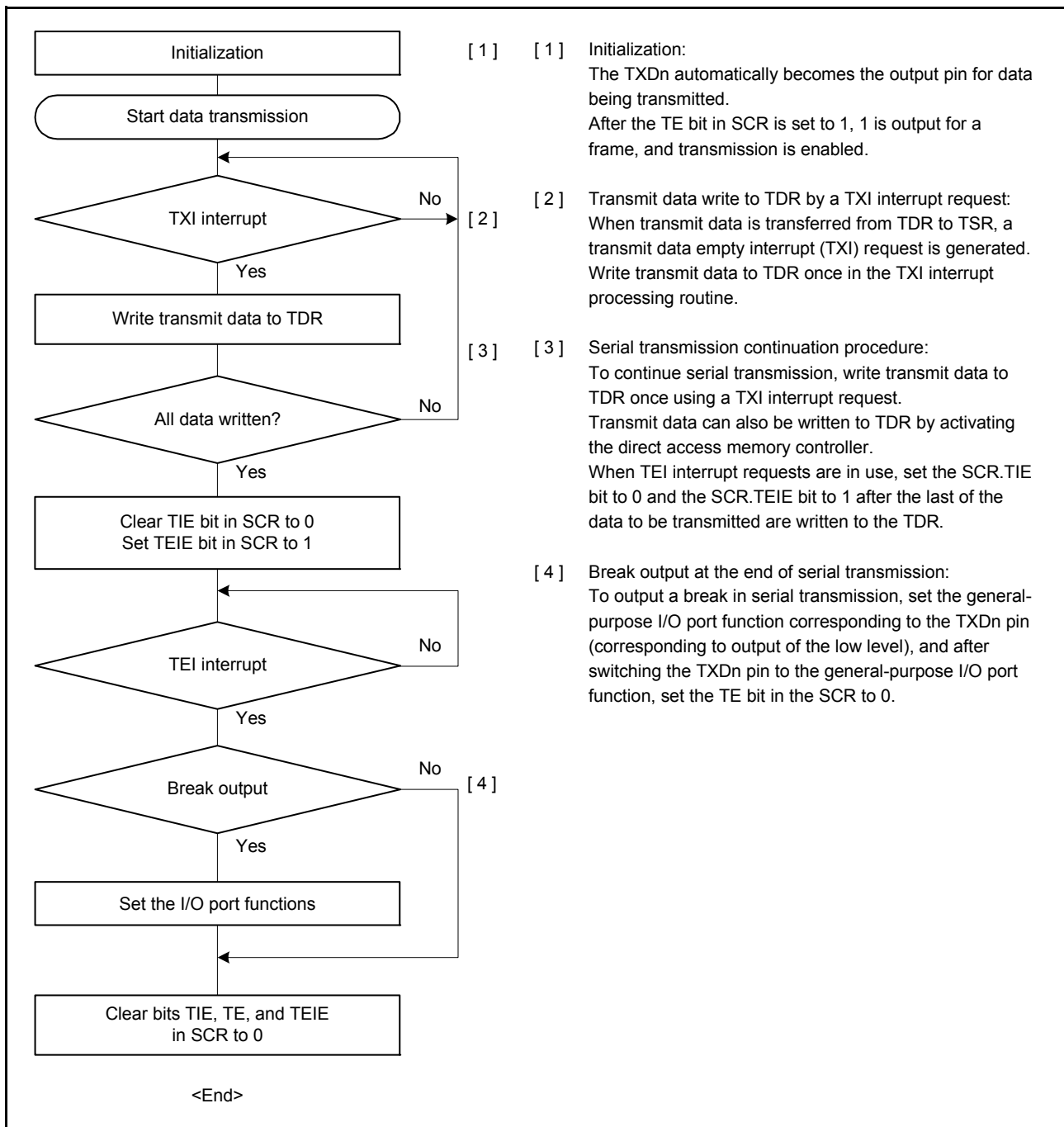
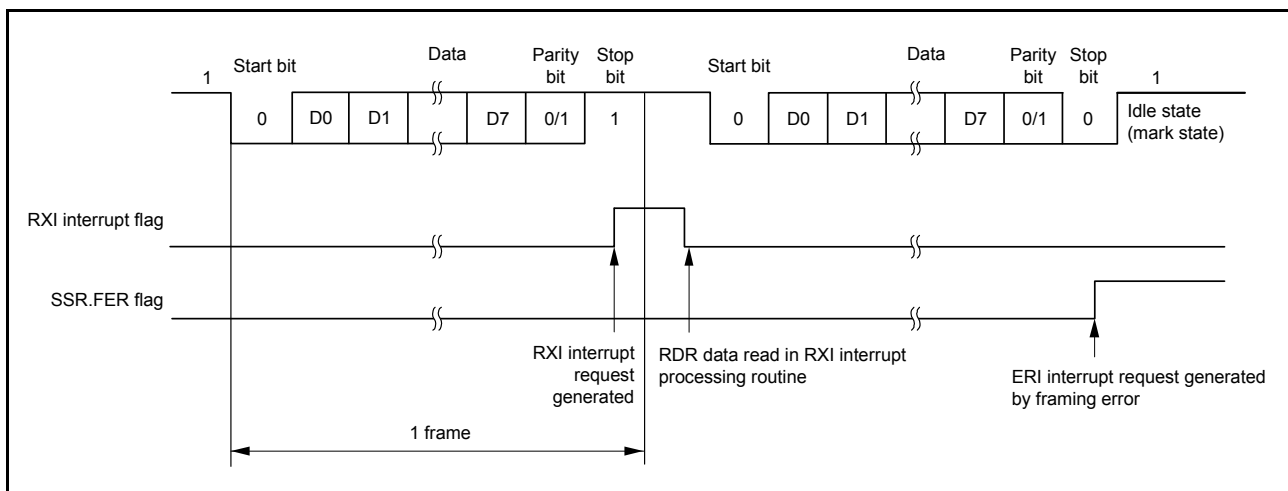


Figure 15.7 Example of Serial Transmission Flowchart in Asynchronous Mode

### 15.3.7 Serial Data Reception (Asynchronous Mode)

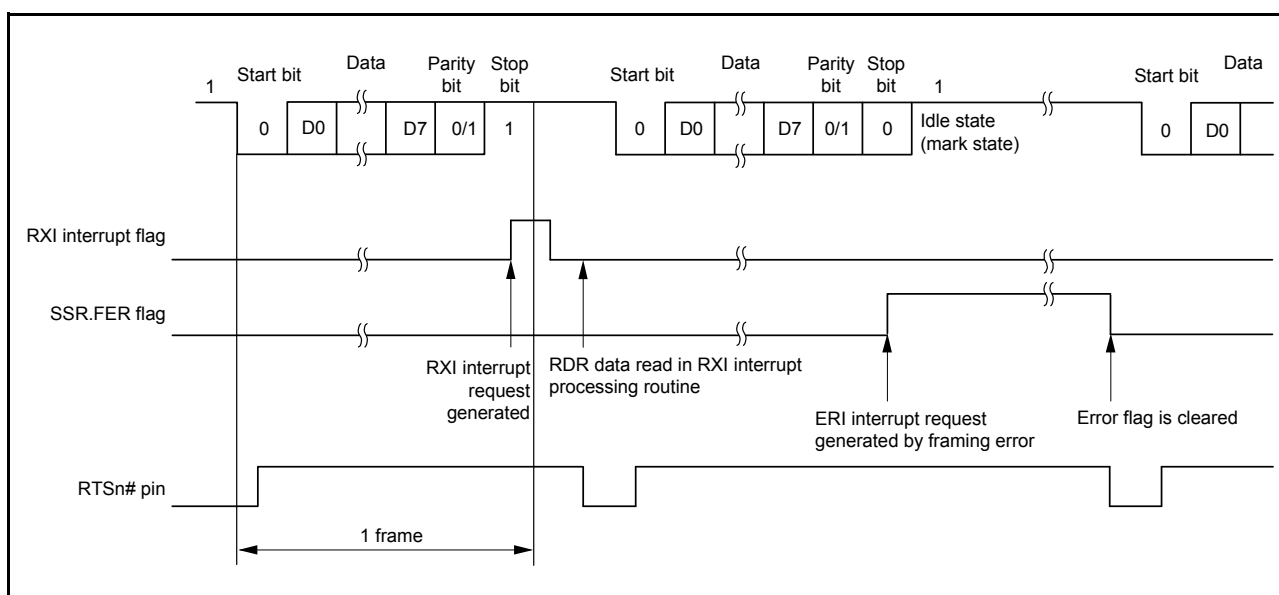
Figure 15.8 and Figure 15.9 show examples of the operation for serial data reception in asynchronous mode. In serial data reception, the SCI operates as described below.

1. When the value of the RE bit in SCR becomes 1, the output signal on the RTSn# pin goes to the low level.
2. When the SCI monitors the communications line and detects a start bit, it performs internal synchronization, stores receive data in RSR, and checks the parity bit and stop bit.
3. If an overrun error occurs, the ORER flag in SSR is set to 1. If the RIE bit in SCR is 1 at this time, an ERI interrupt request is generated. Receive data is not transferred to RDR.
4. If a parity error is detected, the PER bit in SSR is set to 1 and receive data is transferred to RDR. If the RIE bit in SCR is 1 at this time, an ERI interrupt request is generated.
5. If a framing error (when the stop bit is 0) is detected, the FER bit in SSR is set to 1 and receive data is transferred to RDR. If the RIE bit in SCR is 1 at this time, an ERI interrupt request is generated.
6. When reception finishes successfully, receive data is transferred to RDR. If the RIE bit in SCR is 1 at this time, an RXI interrupt request is generated. Continuous reception is enabled by reading the receive data transferred to RDR in this RXI interrupt processing routine before reception of the next receive data is completed. Reading out the received data that have been transferred to RDR causes the RTSn# pin to output the low level.



**Figure 15.8 Example of SCI Operation for Serial Reception in Asynchronous Mode (1) (when RTS Function is not Used) (Example with 8-Bit Data, Parity, One Stop Bit)**





**Figure 15.9 Example of SCI Operation for Serial Reception in Asynchronous Mode (2) (when RTS Function Is Used) (Example with 8-Bit Data, Parity, One Stop Bit)**

Table 15.15 lists the states of the SSR status flags and receive data handling when a receive error is detected. If a receive error is detected, an ERI interrupt request is generated but an RXI interrupt request is not generated. Data reception cannot be resumed while the receive error flag is 1. Accordingly, clear the ORER, FER, and PER bits to 0 before resuming reception. Moreover, be sure to read the RDR during overrun error processing. Figure 15.10 and Figure 15.11 show samples of flowcharts for serial data reception.

**Table 15.15 SSR Status Flags and Receive Data Handling**

SSR Status Flag			Receive Data	Receive Error Type
ORER	FER	PER		
1	0	0	Lost	Overrun error
0	1	0	Transferred to RDR	Framing error
0	0	1	Transferred to RDR	Parity error
1	1	0	Lost	Overrun error + framing error
1	0	1	Lost	Overrun error + parity error
0	1	1	Transferred to RDR	Framing error + parity error
1	1	1	Lost	Overrun error + framing error + parity error

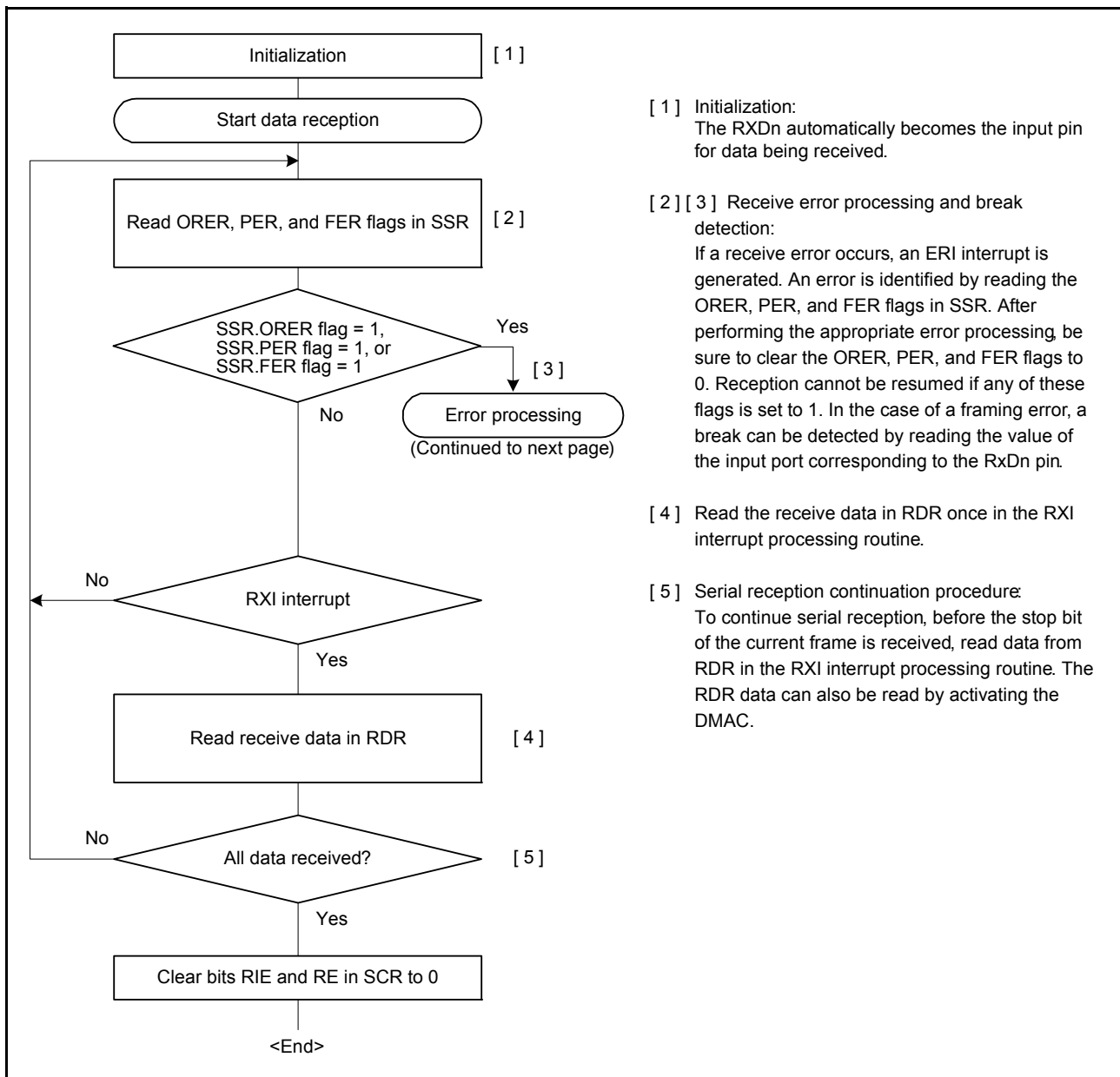


Figure 15.10 Example of Serial Reception Flowchart (1) (Asynchronous Mode)

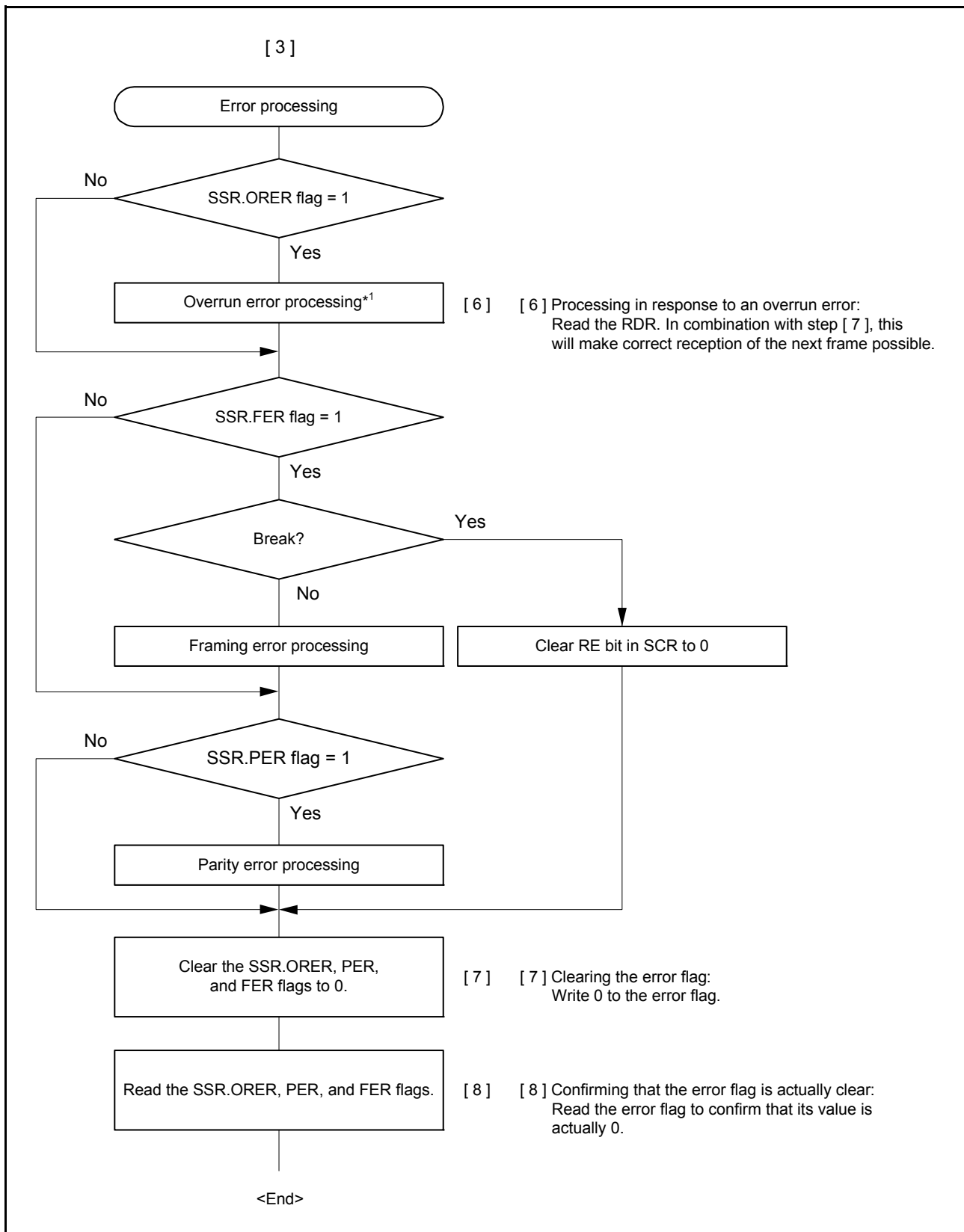
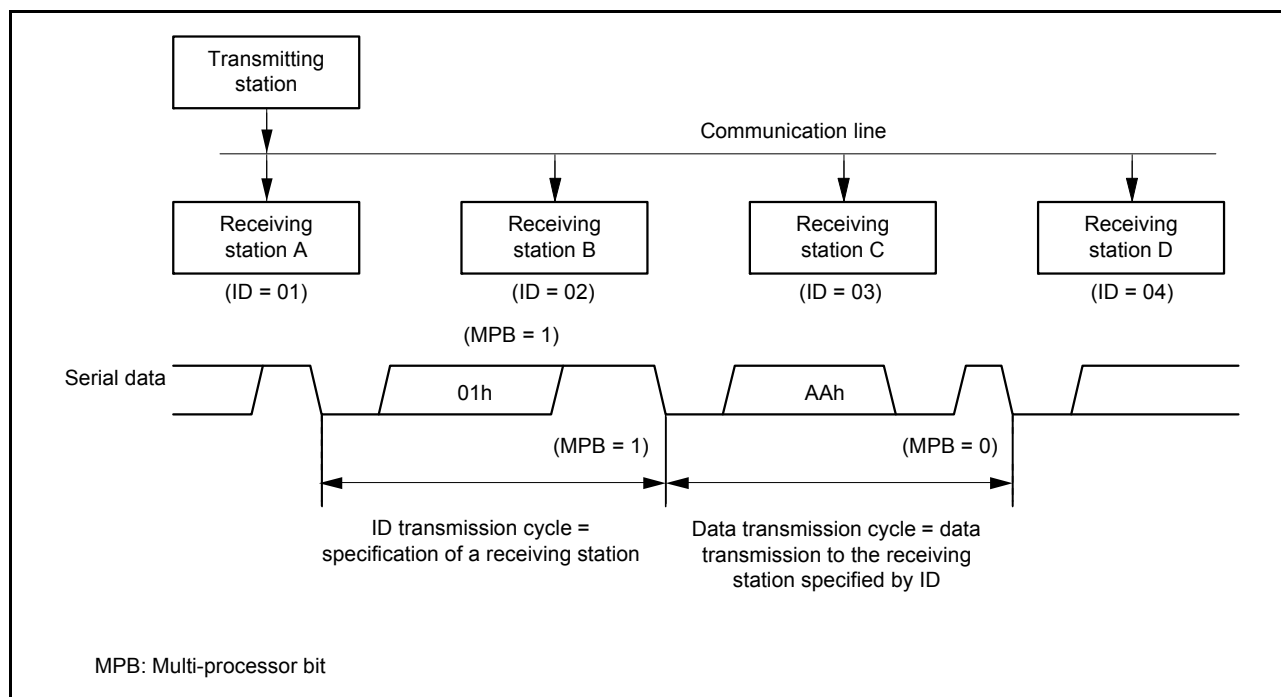


Figure 15.11 Example of Serial Reception Flowchart (2) (Asynchronous Mode)

## 15.4 Multi-Processor Communications Function

Using the multi-processor communication functions enables to transmit and receive data by sharing a communication line between multiple processors by using asynchronous serial communication in which the multi-processor bit is added. In multi-processor communication, a unique ID code is allocated to each receiving station. Serial communication cycles consist of an ID transmission cycle to specify the receiving station and a data transmission cycle to transmit data to the specified receiving station. The multi-processor bit is used to distinguish between the ID transmission cycle and the data transmission cycle. When the multi-processor bit is set to 1, it indicates the ID transmission cycle and when the multi-processor bit is set to 0, it indicates the data transmission cycle. Figure 15.12 shows an example of communication between processors by using a multi-processor format. First, a transmitting station transmits communication data in which the multi-processor bit set to 1 is added to the ID code of the receiving station. Next, the transmitting station transmits the communication data in which the multi-processor bit set to 0 is added to the transmission data. Upon receiving the communication data in which the multi-processor bit is set to 1, the receiving station compares the received ID with the ID of the receiving station itself and if the two match, receives the communication data that is subsequently transmitted. If the received ID does not match with the ID of the receiving station, the receiving station skips the communication data until again receiving the communication data in which the multi-processor bit is set to 1. For supporting this function, the SCI provides the MPIE bit in SCR. When the MPIE bit in SCR is set to 1, transfer of receive data from the RSR to the RDR, detection of a reception error, and setting the respective status flags ORER and FER in SSR are disabled until reception of data in which the multi-processor bit is set to 1. Upon receiving a reception character in which the multi-processor bit is set to 1, the MPBT bit in SSR is set to 1 and the MPIE bit in SCR is automatically cleared, thus returning to a normal reception operation. During this time, an RXI interrupt is generated if the RIE bit in SCR is set.

When the multi-processor format is specified, specification of the parity bit is disabled. Apart from this, there is no difference from the operation in the normal asynchronous mode. A clock which is used for the multi-processor communication is also the same as the clock used in the normal asynchronous mode.



**Figure 15.12** An Example of Communication using the Multi-Processor Format  
(Example of Transmission of Data AAh to Receiving Station A)

### 15.4.1 Multi-Processor Serial Data Transmission

Figure 15.13 is a sample flowchart of multi-processor data transmission. In the ID transmission cycle, the ID should be transmitted with the MPBT bit in SSR set to 1. In the data transmission cycle, the data should be transmitted with the MPBT bit set to 0. The other operations are the same as the operations in asynchronous mode.

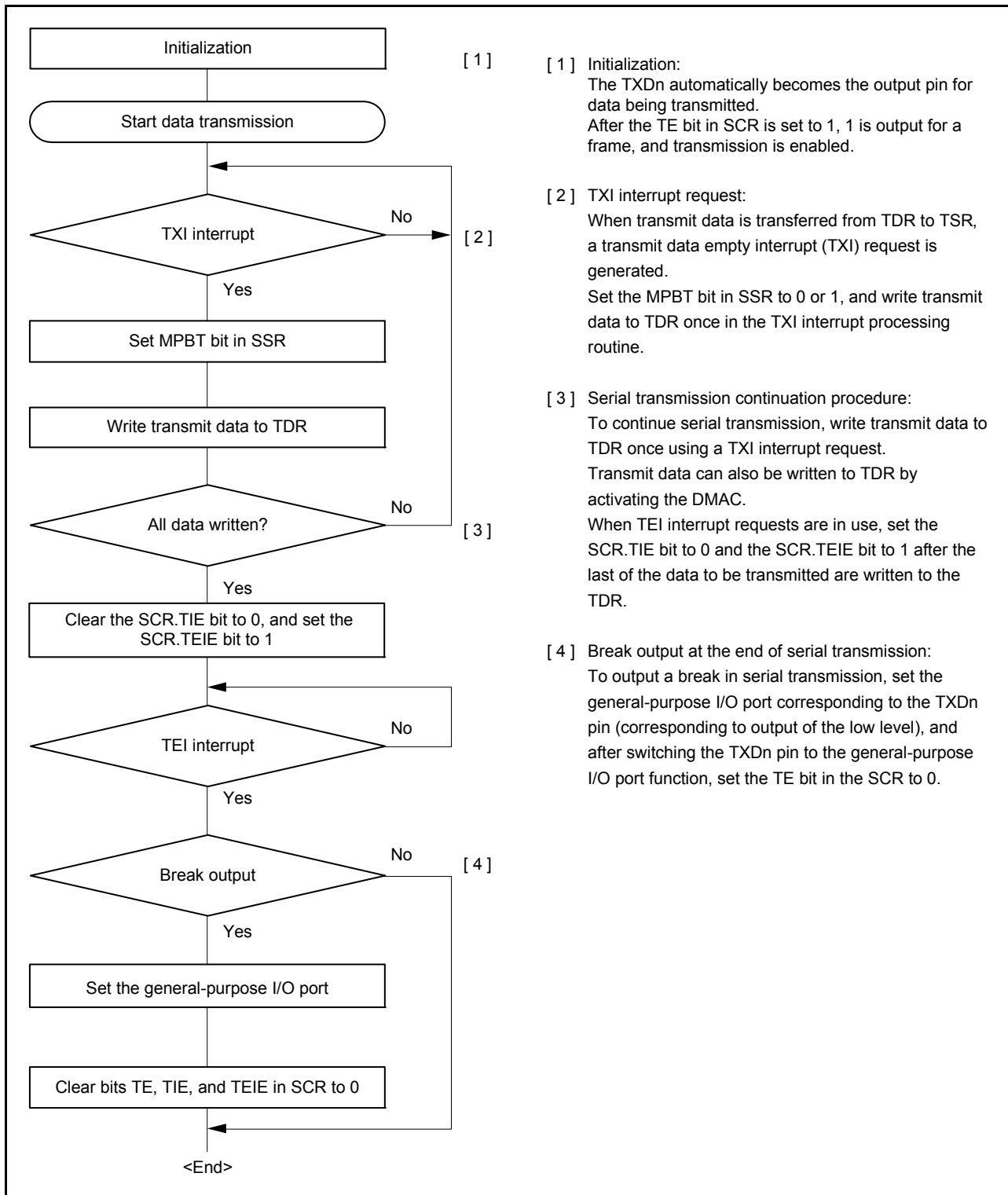


Figure 15.13 Example of Multi-Processor Serial Transmission Flowchart

### 15.4.2 Multi-Processor Serial Data Reception

Figure 15.15 and Figure 15.16 are sample flowcharts of multi-processor data reception. When the MPIE bit in SCR is set to 1, reading the communication data is skipped until reception of the communication data in which the multi-processor bit is set to 1. When the communication data in which the multi-processor bit is set to 1 is received, the received data is transferred to RDR. During this time, the RXI interrupt request is generated. The other operations are the same as the operations in asynchronous mode.

Figure 15.14 is the example of operation for reception.

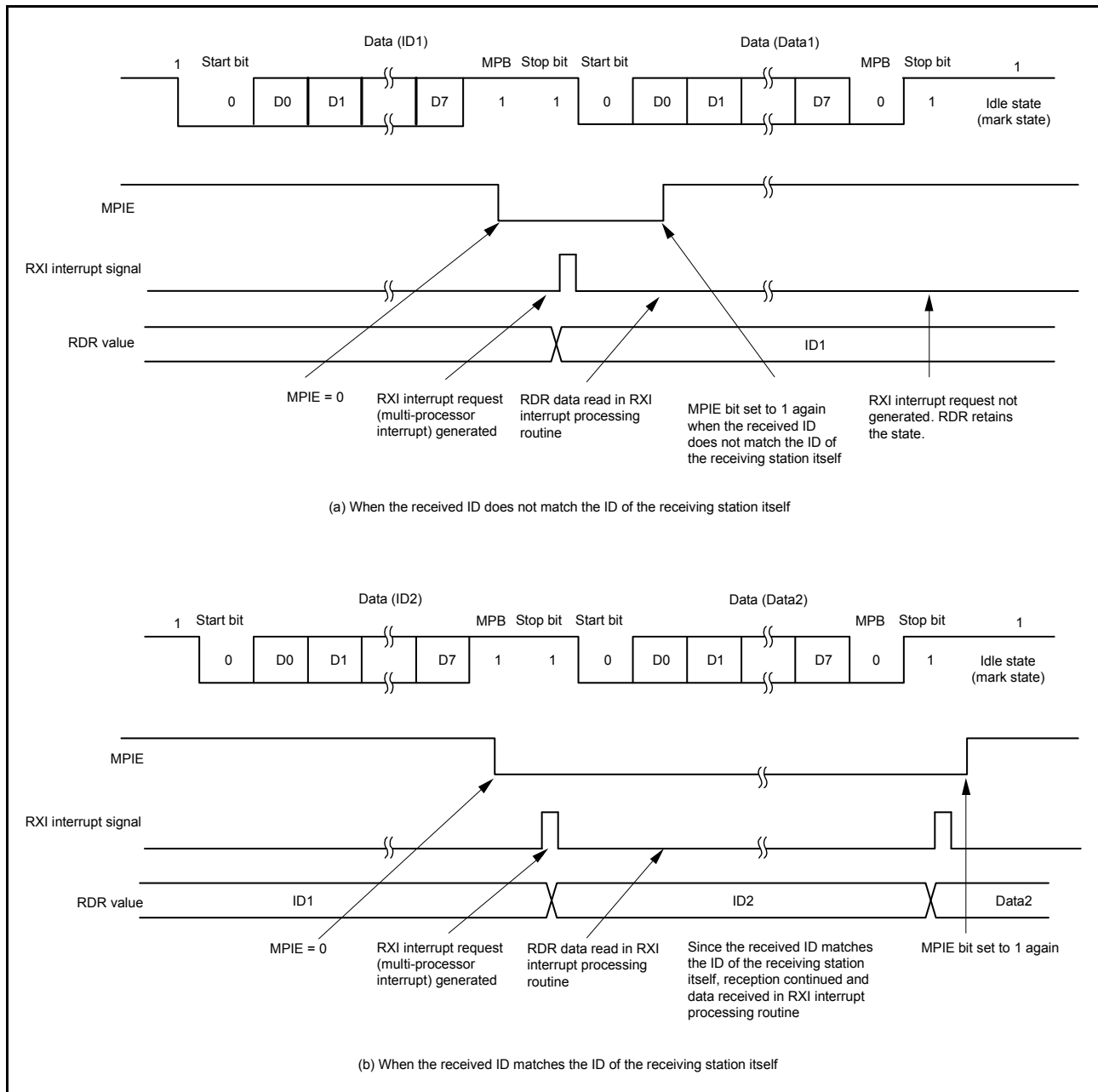


Figure 15.14 Example of Reception (8-Bit Data/Multi-Processor Bit/One Stop Bit)

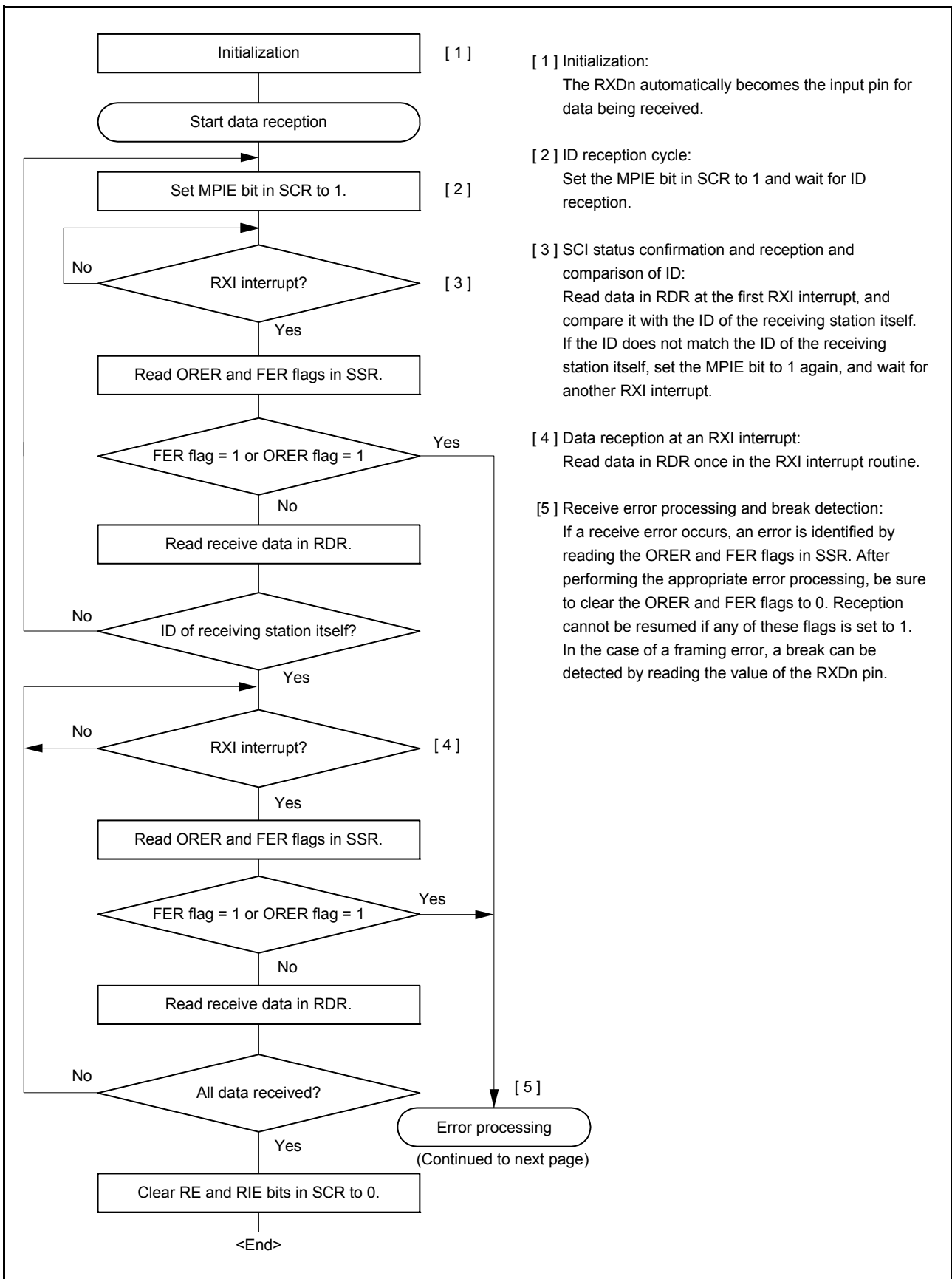


Figure 15.15 Example of Multi-Processor Serial Reception Flowchart (1)

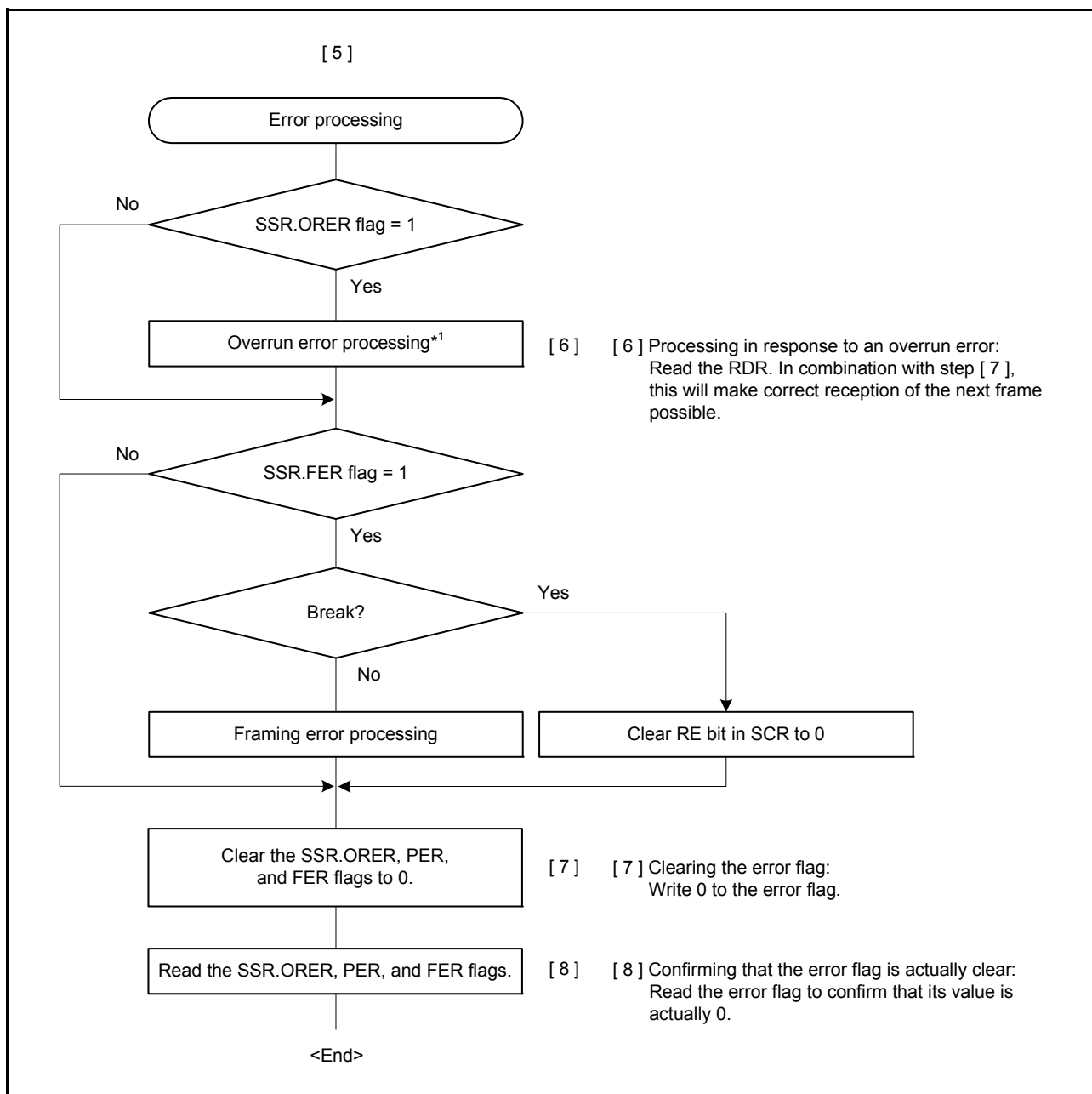


Figure 15.16 Example of Multi-Processor Serial Reception Flowchart (2)



## 15.5 Operation in Clock Synchronous Mode

Figure 15.17 shows the data format for clock synchronous serial data communications.

In clock synchronous mode, data is transmitted or received in synchronization with clock pulses. One character in transfer data consists of 8-bit data. In clock synchronous mode, no parity bit can be added.

In data transmission, the SCI outputs data from one falling edge of the synchronization clock to the next. In data reception, the SCI receives data in synchronization with the rising edge of the synchronization clock. After 8-bit data is output, the transmission line holds the last bit output state.

Inside the SCI, the transmitter and receiver are independent units, enabling full-duplex communications by use of a common clock. Both the transmitter and the receiver also have a double-buffered structure, so that the next transmit data can be written during transmission or the previous receive data can be read during reception, enabling continuous data transfer.

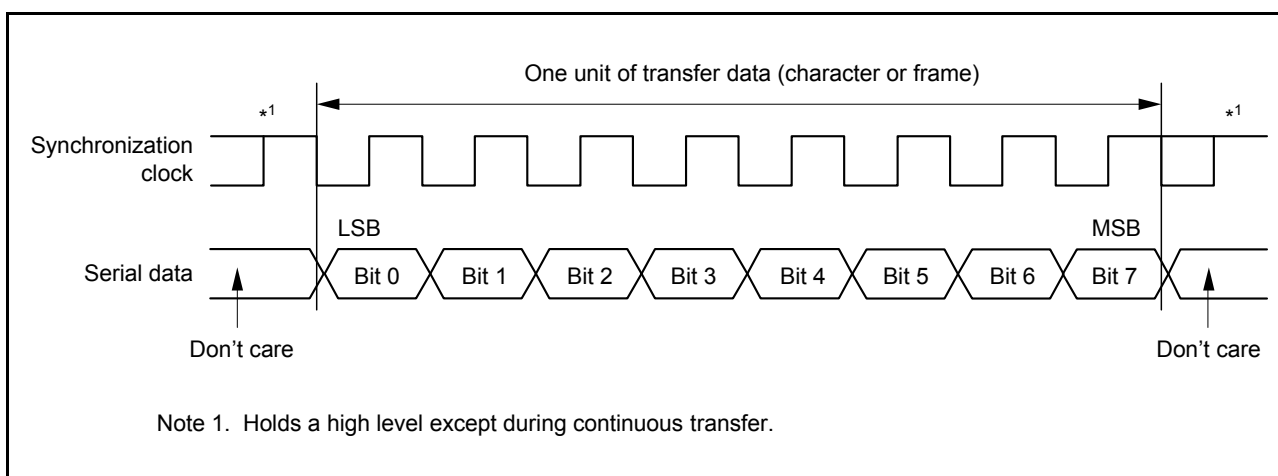


Figure 15.17 Data Format in Clock Synchronous Serial Communications (LSB-First)

### 15.5.1 Clock

Either an internal clock generated by the on-chip baud rate generator or an external synchronization clock input at the SCKn pin can be selected, according to the setting of the CKEn[1:0] bits in SCR.

When the SCI is operated on an internal clock, the synchronization clock is output from the SCKn pin. Eight synchronization clock pulses are output in the transfer of one character, and when no transfer is performed the clock is held high. However, when only data reception is performed, output of the synchronizing clock signal continues until the CTS function is enabled and the high level is input on the CTSn# pin, an overflow error occurs, or the RE bit in SCR is set to 0. When the CTS function is enabled, the synchronous clock signal output is stopped if the CTSn# pin input is high on completion of the frame reception.

### 15.5.2 CTS and RTS Functions

In the CTS function, CTSn# pin input is used to control reception/transmission start when the clock source is the internal clock. Setting the SECR.CTSE bit to 1 enables the CTS function.

When the CTS function is enabled, placing the low level on the CTSn# pin causes reception/transmission to start.

In the RTS function, RTSn# pin output is used to request reception/transmission start when the clock source is an external synchronizing clock. A low level is output when serial communications become possible. Conditions for output of the low and high level are shown below.

[Conditions for low-level output]

Satisfaction of all conditions listed below

- The value of the RE or TE bit in the SCR is 1
- Neither transmission nor reception is in progress
- There are no received data yet to be read (when the SCR.RE bit is 1)
- Transmit data has been written (when the SCR.TE bit is 1)
- ORER flag in SSR is 0

[Condition for high-level output]

Any of the conditions for the low level not being satisfied

### 15.5.3 Initialization (Clock Synchronous Mode)

Before transmitting and receiving data, start by writing the initial value “00h” to the SCR and then continue through the procedure for SCI given in the sample flowchart (Figure 15.18). Whenever the operating mode or transfer format is changed, the SCR must be initialized before the change is made.

Note that clearing the SCR.RE bit to 0 initializes neither the ORER, FER, and PER flags in SSR nor RDR.

Moreover, note that switching the value of the SCR.TE bit from 1 to 0 or 0 to 1 while the SCR.TIE bit is 1 leads to the generation of a TXI interrupt request.

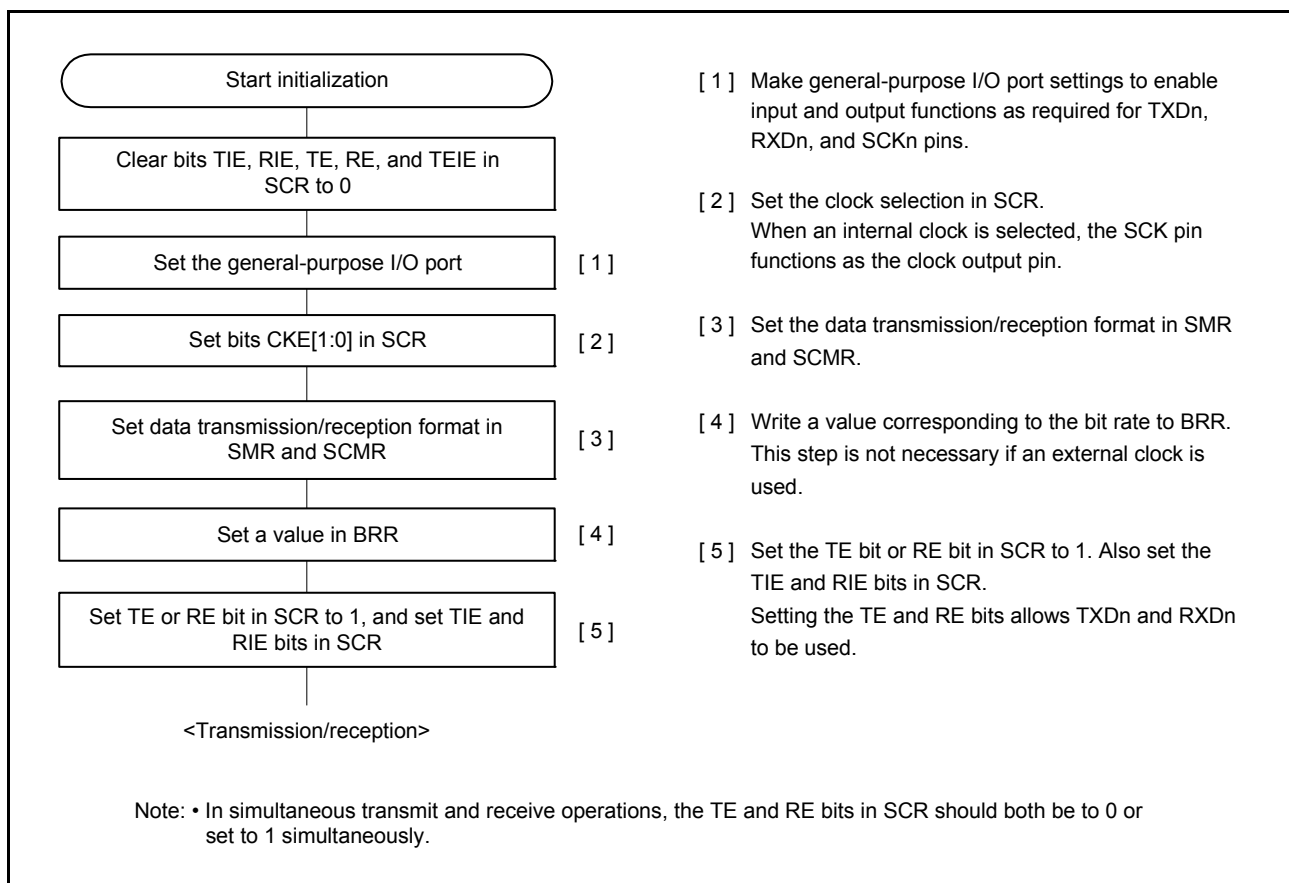


Figure 15.18 Example of Initialization Flowchart (Clock Synchronous Mode)

### 15.5.4 Serial Data Transmission (Clock Synchronous Mode)

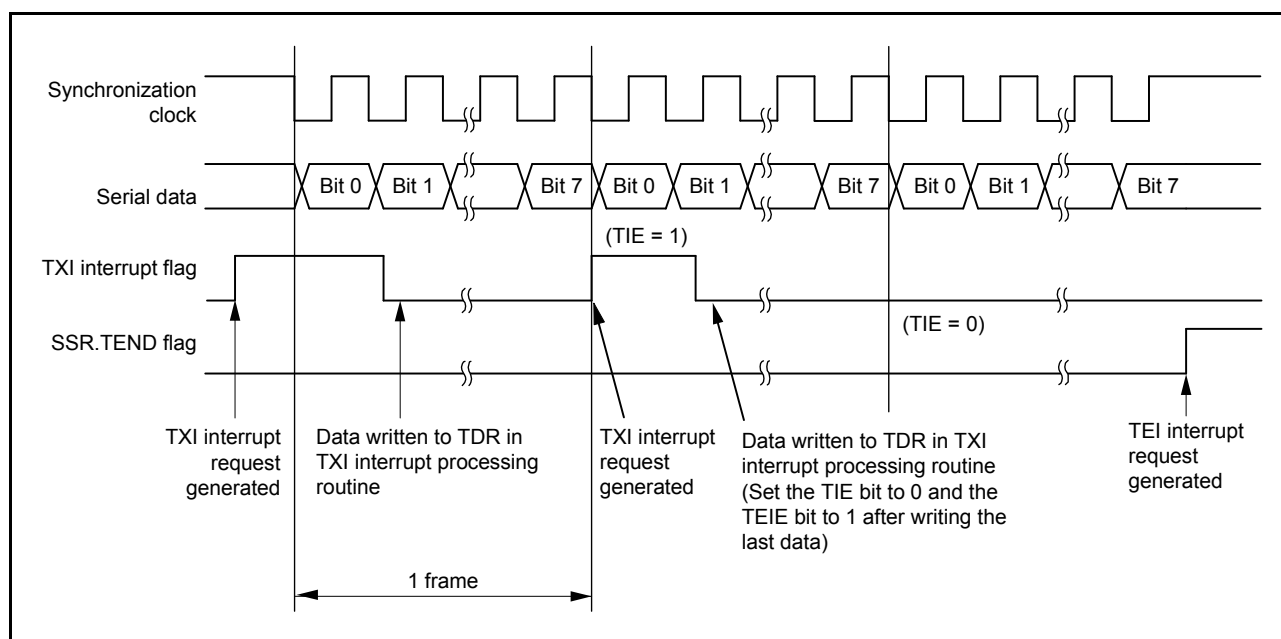
Figure 15.19 shows an example of the operation for serial transmission in clock synchronous mode.

In serial data transmission, the SCI operates as described below.

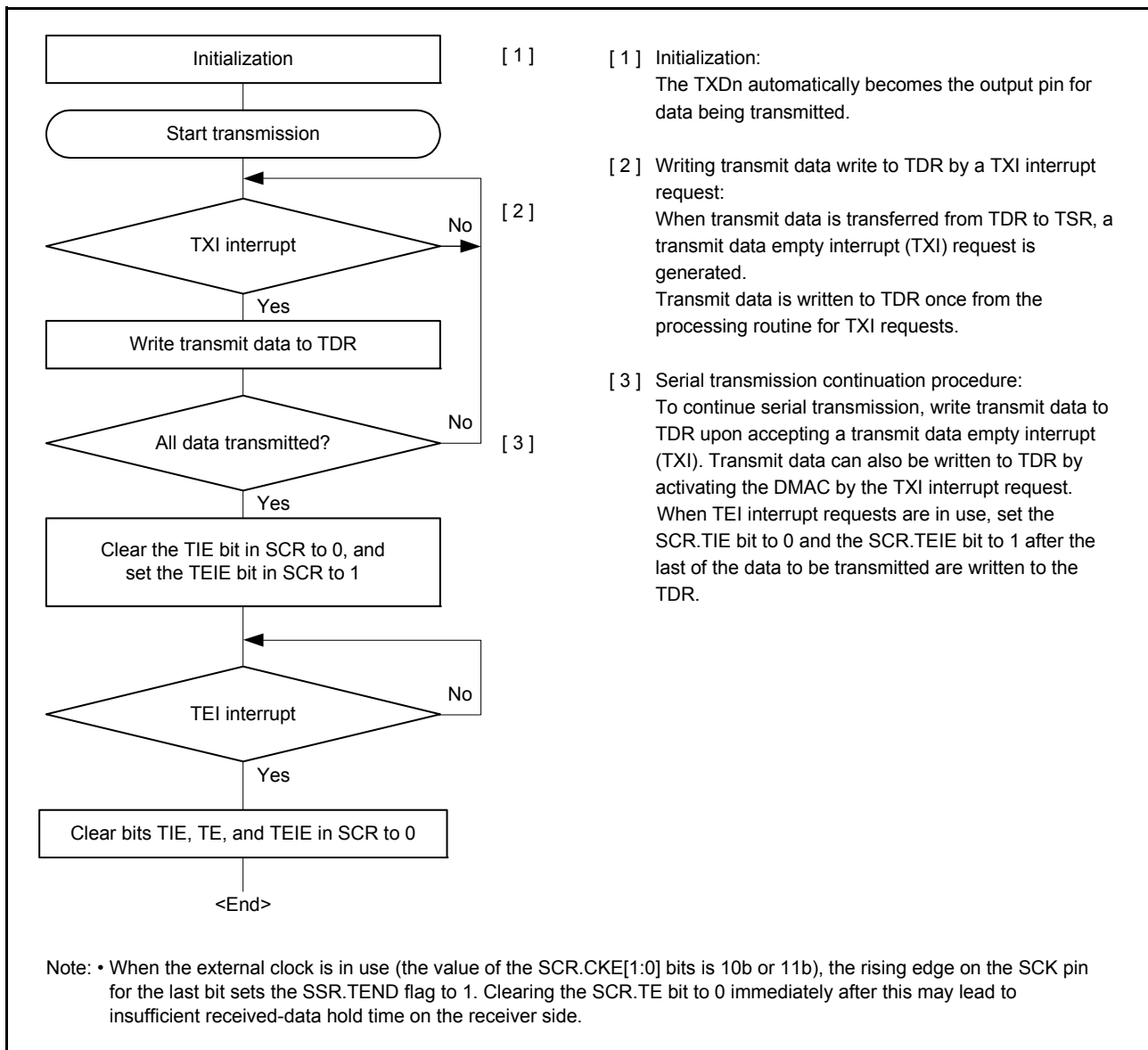
1. The SCI transfers data from TDR to TSR when data is written to TDR in the TXI interrupt processing routine. The TXI interrupt request at the beginning of transmission is generated when the TE bit in SCR is set to 1 after the TIE bit in SCR is set to 1 or when these two bits are set to 1 simultaneously by a single instruction.
2. After transferring data from TDR to TSR, the SCI starts transmission. When the SCR.TIE bit is set to 1 at this time, a TXI interrupt request is generated. Continuous transmission is enabled by writing the next transmit data to TDR in this TXI interrupt processing routine before transmission of the current transmit data has finished. When TEI interrupt requests are in use, set the SCR.TIE bit to 0 (disabling TXI requests) and the SCR.TEIE bit to 1 (enabling TEI requests) after the last of the data to be transmitted are written to the TDR from the processing routine for TXI requests.
3. 8-bit data is sent from the TXDn pin in synchronization with the output clock when clock output mode has been specified and in synchronization with the input clock when use of an external clock has been specified. Output of the clock signal is suspended until the input CTS signal is at the low level while the CTSE bit in SECR is 1 (enabling the CTS function).
4. The SCI checks for updating of (writing to) the TDR at the time of the last bit output.
5. When TDR is updated, the next transmit data is transferred from TDR to TSR, and serial transmission of the next frame is started.
6. If TDR is not updated, set the SSR flag in TEND to 1 and the TXDn pin retains the output state of the last bit. If the TEIE bit in SCR is 1 at this time, a TEI interrupt request is generated. The SCKn pin is held high.

Figure 15.20 shows a sample flowchart of serial data transmission.

Transmission will not start while a receive error flag (ORER, FER, or PER in SSR) is set to 1. Be sure to clear the receive error flags to 0 before starting transmission. Note that clearing the RE bit in SCR to 0 does not clear the receive error flags.



**Figure 15.19 Example of Operation for Serial Transmission in Clock Synchronous Mode (from the Middle of Transmission until Transmission Completion)**

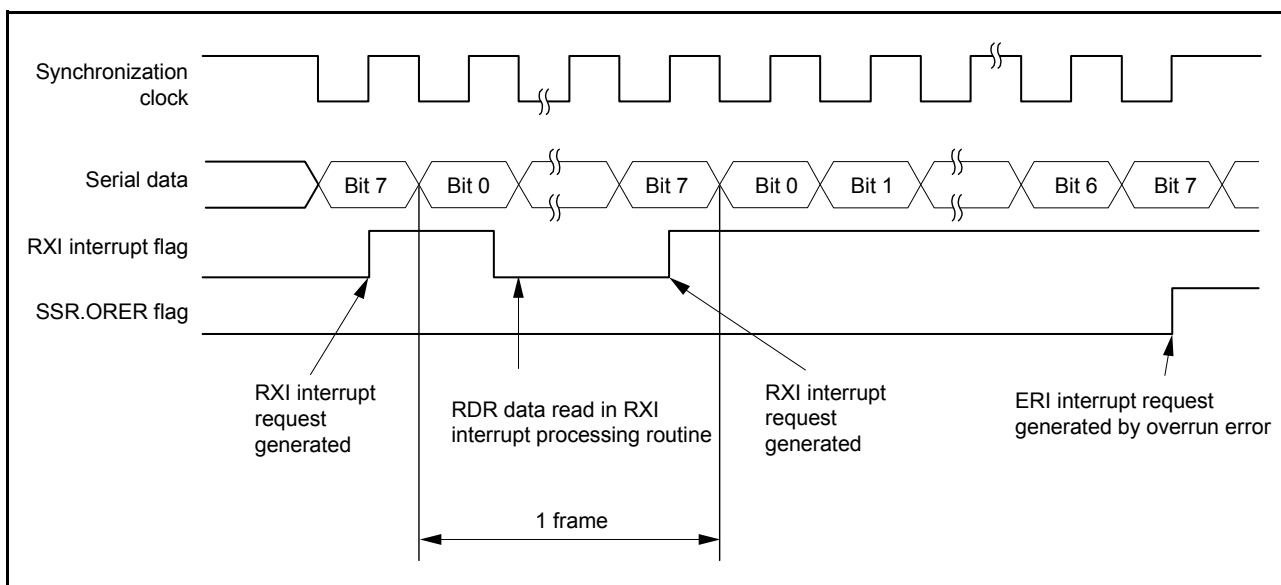


**Figure 15.20 Example of Serial Transmission Flowchart (Clock Synchronous Mode)**

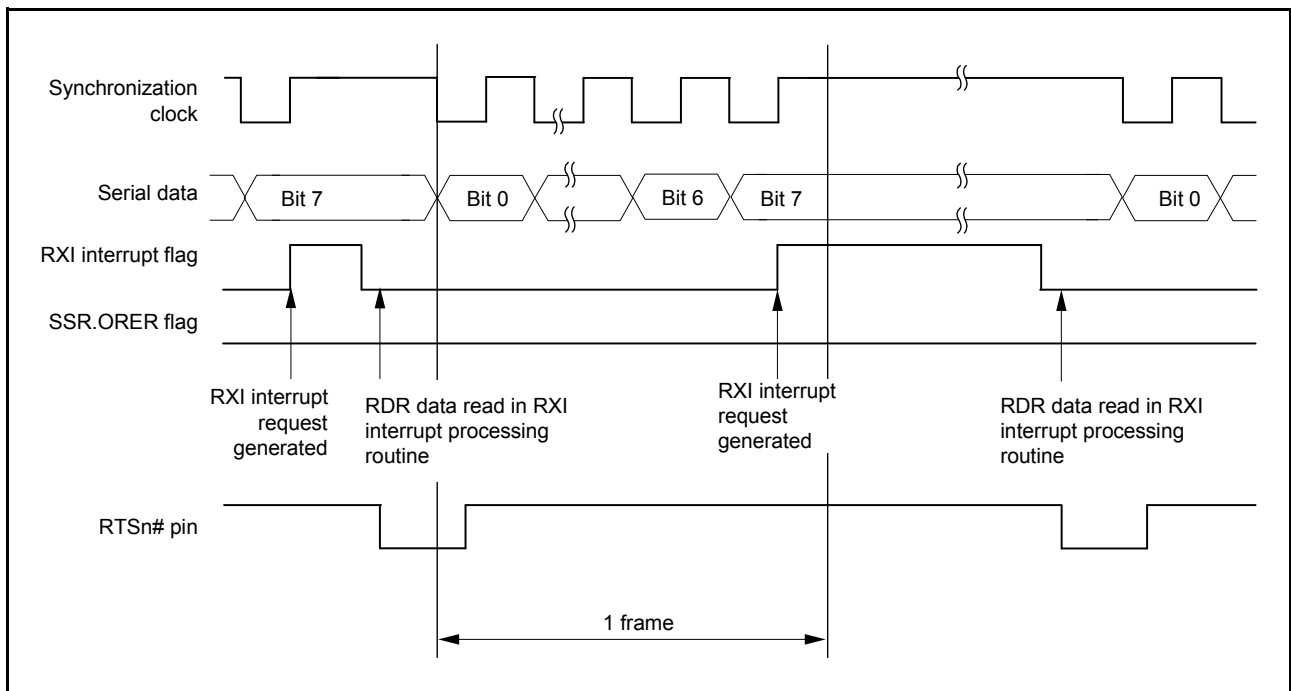
### 15.5.5 Serial Data Reception (Clock Synchronous Mode)

Figure 15.21 and Figure 15.22 show examples of SCI operation for serial reception in clock synchronous mode. In serial data reception, the SCI operates as described below.

1. The value of the RE bit in SCR becoming 1 places the signal output on the RTS pin at the low level (when the RTS function is in use).
2. The SCI performs internal initialization and starts receiving data in synchronization with a synchronization clock input or output, and stores the receive data in RSR.
3. If an overrun error occurs, the ORER bit in SSR is set to 1. If the RIE bit in SCR is 1 at this time, an ERI interrupt request is generated. Receive data is not transferred to RDR.
4. When reception finishes successfully, receive data is transferred to RDR. If the RIE bit in SCR is 1 at this time, an RXI interrupt request is generated. Continuous reception is enabled by reading the receive data transferred to RDR in this RXI interrupt processing routine before reception of the next receive data is completed. Reading out the received data that have been transferred to RDR causes the RTSn# pin to output the low level (when the RTS function is in use).



**Figure 15.21** Example of Operation for Serial Reception in Clock Synchronous Mode (1) (when RTS Function is not Used)



**Figure 15.22 Example of Operation for Serial Reception in Clock Synchronous Mode (2) (when RTS Function is Used)**

Data transfer cannot be resumed while a receive error flag is 1. Accordingly, clear the ORER, FER, and PER bits in SSR to 0 before resuming reception. Moreover, be sure to read the RDR during overrun error processing. Figure 15.23 shows a sample flowchart for serial data reception.

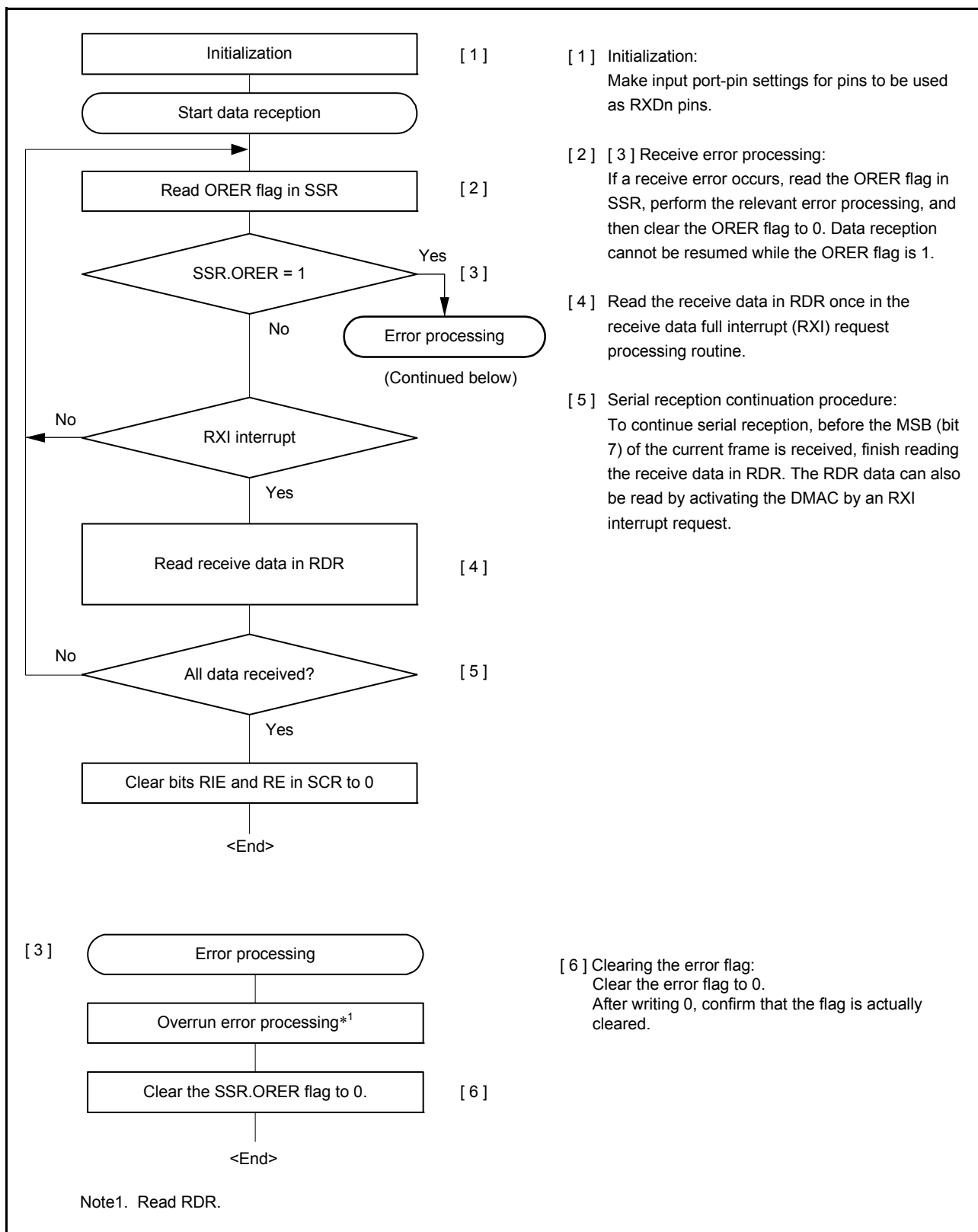


Figure 15.23 Example of Serial Reception Flowchart (Clock Synchronous Mode)

### 15.5.6 Simultaneous Serial Data Transmission and Reception (Clock Synchronous Mode)

Figure 15.24 shows a sample flowchart for simultaneous serial transmit and receive operations in clock synchronous mode.

After initializing the SCI, the following procedure should be used for simultaneous serial data transmit and receive operations.

To switch from transmit mode to simultaneous transmit and receive mode, check that the SCI has finished transmission by reading that the TEND flag in SSR is 1, and then initialize the SCR register. Then set the TIE, RIE, TE, and RE bits in SCR to 1 simultaneously by a single instruction.

To switch from receive mode to simultaneous transmit and receive mode, check that the SCI has finished reception, and then clear the RIE and RE bits to 0. Then check that the receive error flags (ORER, FER, and PER in SSR) are 0, and then set the TIE, RIE, TE, and RE bits in SCR to 1 simultaneously by a single instruction.

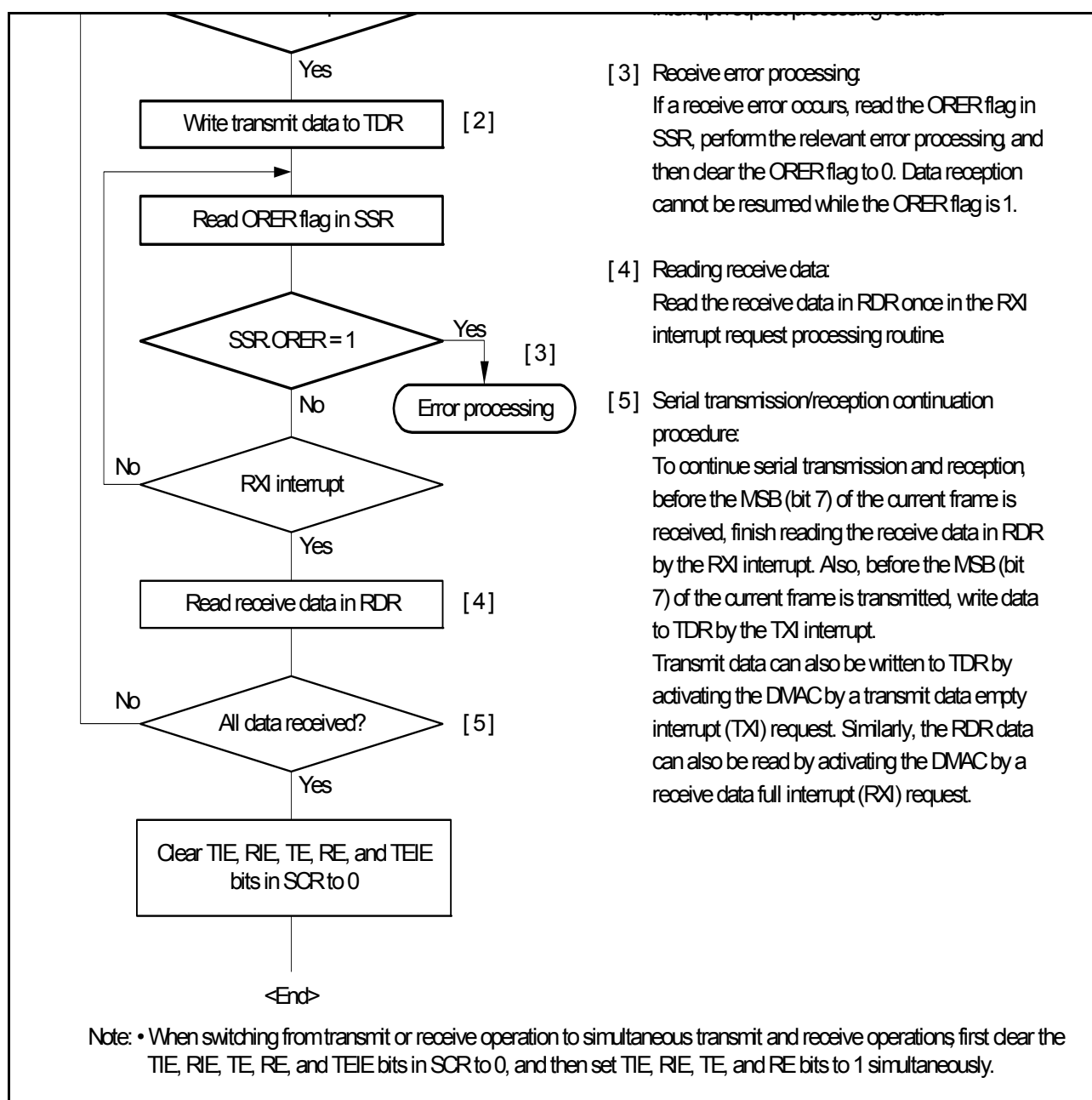


Figure 15.24 Example of Simultaneous Serial Transmission and Reception Flowchart (Clock Synchronous Mode)



## 15.6 Operation in Smart Card Interface Mode

The SCI supports the smart card (IC card) interface conforming to the ISO/IEC 7816-3 (Identification Card) standard. Smart card interface mode can be selected using the appropriate register.

### 15.6.1 Sample Connection

Figure 15.25 shows a sample connection between a smart card (IC card) and this LSI.

As in the figure, since this LSI communicates with an IC card using a single transmission line, interconnect the TXDn and RXDn pins and pull up the data transmission line to Vcc using a resistor.

Setting the TE and RE bits in SCR to 1 with an IC card disconnected enables closed transmission/reception allowing self-diagnosis.

To supply an IC card with the clock pulses generated by the SCI, input the SCKn pin output to the CLK pin of an IC card. The output port of this LSI can be used to output a reset signal.

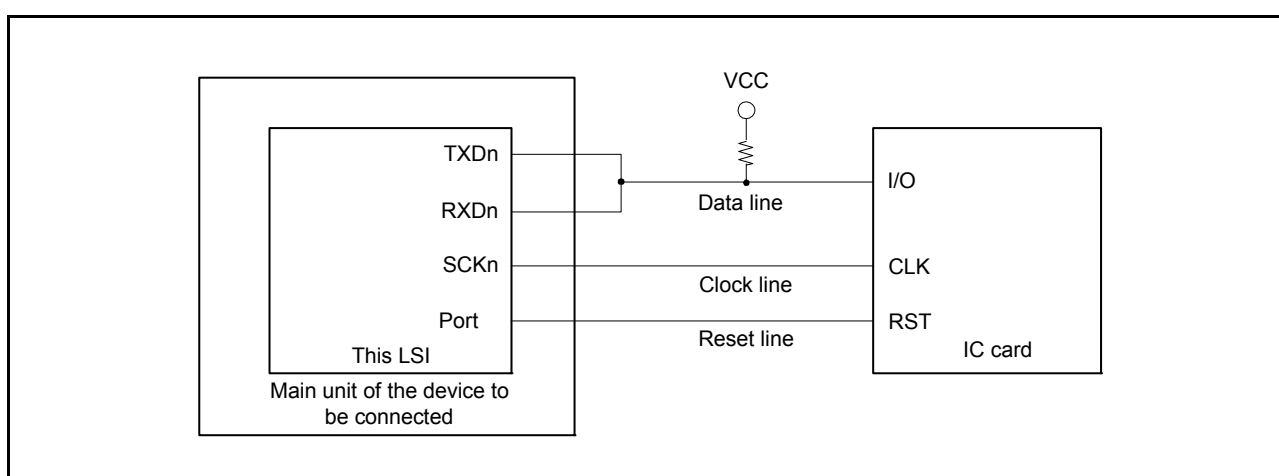
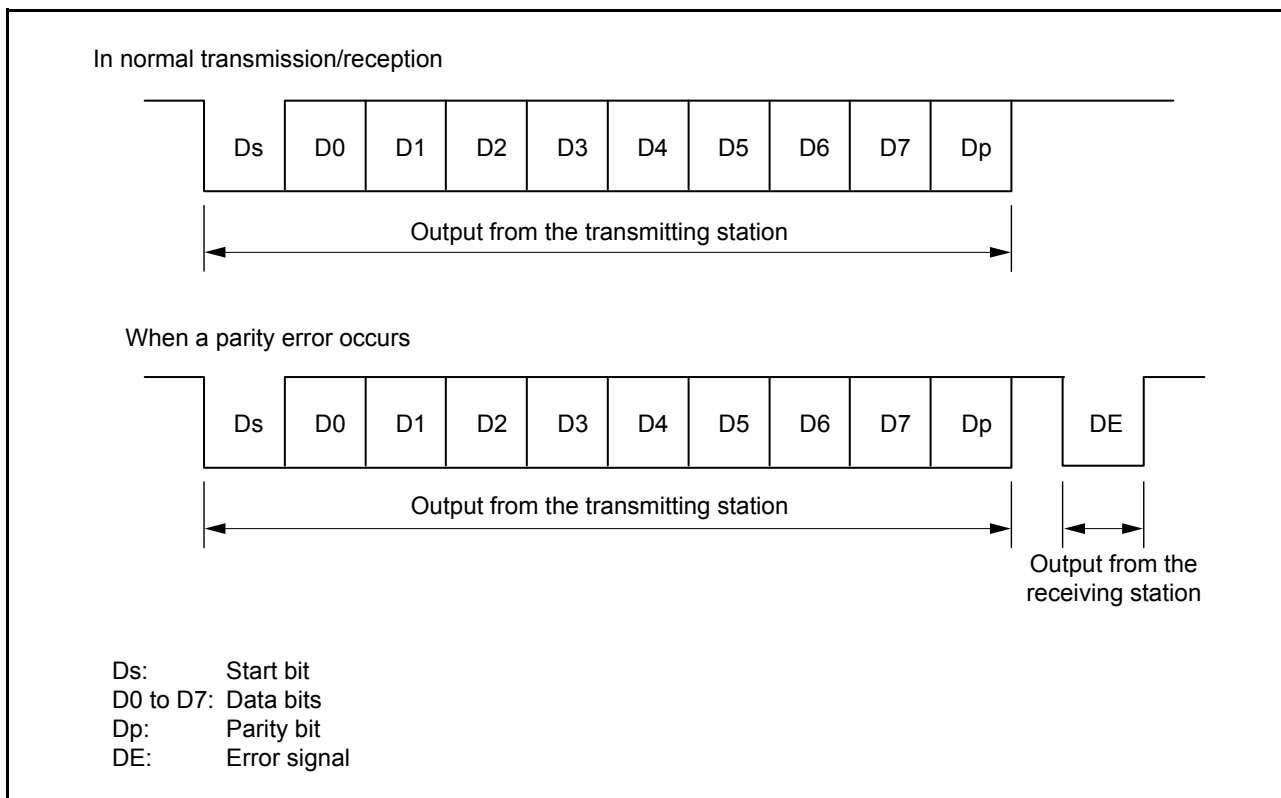


Figure 15.25 Sample Connection with a Smart Card (IC Card)

### 15.6.2 Data Format (Except in Block Transfer Mode)

Figure 15.26 shows the data transfer formats in smart card interface mode.

- One frame consists of 8-bit data and a parity bit in asynchronous mode.
- During transmission, at least 2 etu (elementary time unit: time required for transferring one bit) is secured as a guard time from the end of the parity bit until the start of the next frame.
- If a parity error is detected during reception, a low-level error signal is output for 1 etu after 10.5 etu has passed from the start bit.
- If an error signal is sampled during transmission, the same data is automatically re-transmitted after at least 2 etu.



**Figure 15.26 Data Formats in Smart Card Interface Mode**

For communications with IC cards of the direct convention type and inverse convention type, follow the procedure below.

### (1) Direct Convention Type

For the direct convention type, logic levels 1 and 0 correspond to states Z and A, respectively, and data is transferred with LSB-first as the start character, as shown in Figure 15.27. Therefore, data in the start character in the figure is 3Bh. When using the direct convention type, write 0 to both the SDIR and SINV bits in SCMR. Write 0 to the PM bit in SMR in order to use even parity, which is prescribed by the smart card standard.

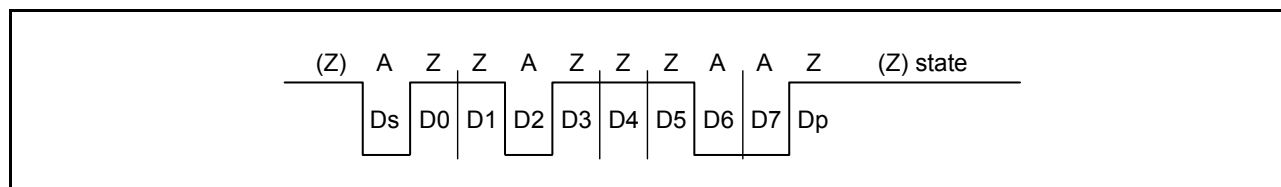


Figure 15.27 Direct Convention (SDIR in SCMR = 0, SINV in SCMR = 0, PM in SMR = 0)

### (2) Inverse Convention Type

For the inverse convention type, logic levels 1 and 0 correspond to states A and Z, respectively and data is transferred with MSB-first as the start character, as shown in Figure 15.28. Therefore, data in the start character in the figure is 3Fh. When using the inverse convention type, write 1 to both the SDIR and SINV bits in SCMR. The parity bit is logic level 0 to produce even parity, which is prescribed by the smart card standard, and corresponds to state Z. Since the SCMR.SINV bit of this LSI only inverts data bits D7 to D0, write 1 to the PM bit in SMR to invert the parity bit for both transmission and reception.

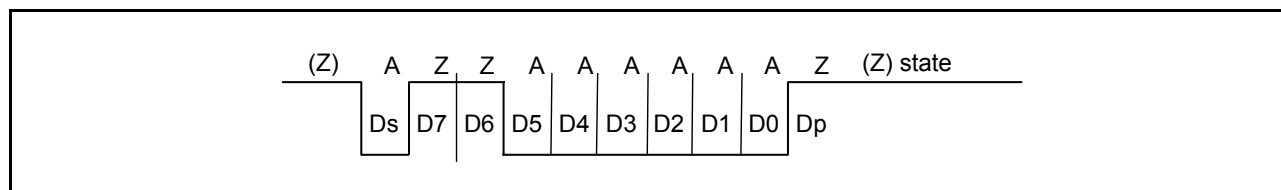


Figure 15.28 Inverse Convention (SDIR in SCMR = 1, SINV in SCMR = 1, PM in SMR = 1)

### 15.6.3 Block Transfer Mode

Block transfer mode is different from normal smart card interface mode in the following respects.

- Even if a parity error is detected during reception, no error signal is output. Since the PER bit in SSR is set by error detection, clear the PER bit before receiving the parity bit of the next frame.
- During transmission, at least 1 etu is secured as a guard time from the end of the parity bit until the start of the next frame.
- Since the same data is not re-transmitted during transmission, the TEND flag in SSR is set 11.5 etu after transmission start.
- In block transfer mode, the ERS flag in SSR indicates the error signal status as in normal smart card interface mode, but the flag is always read as 0 because no error signal is transferred.

### 15.6.4 Receive Data Sampling Timing and Reception Margin

Only the internal clock generated by the on-chip baud rate generator can be used as a transfer clock in smart card interface mode.

In this mode, the SCI can operate on a base clock with a frequency of 32, 64, 372, 256, 93, 128, 186, or 512 times the bit rate according to the settings of the BCP2 bit in SCMR and the BCP[1:0] bits in SMR (the frequency is always 16 times the bit rate in normal asynchronous mode).

For data reception, the falling edge of the start bit is sampled with the base clock to perform internal synchronization. Receive data is sampled on the 16th, 32nd, 186th, 128th, 46th, 64th, 93rd, and 256th rising edges of the base clock so that it can be latched at the middle of each bit as shown in Figure 15.29. The reception margin here is determined by the following formula.

$$M = \left| \left( 0.5 - \frac{1}{2N} \right) - (L - 0.5) F - \frac{|D - 0.5|}{N} (1+F) \right| \times 100 \text{ [%]}$$

M: Reception margin (%)

N: Ratio of bit rate to clock (N = 32, 64, 372, 256)

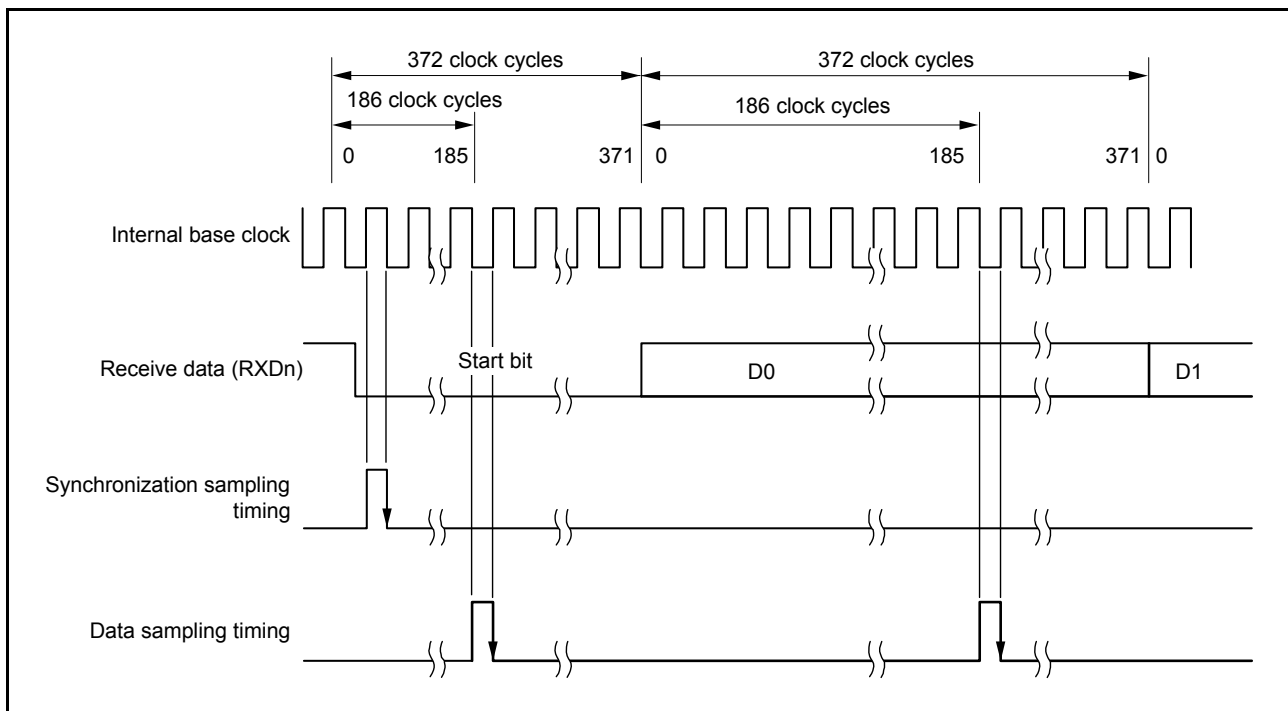
D: Duty cycle of clock (D = 0 to 1.0)

L: Frame length (L = 10)

F: Absolute value of clock frequency deviation

Assuming values of F = 0, D = 0.5, and N = 372 in the above formula, the reception margin is determined by the formula below.

$$M = \{0.5 - 1/(2 \times 372)\} \times 100 \text{ [%]} = 49.866\%$$



**Figure 15.29 Receive Data Sampling Timing in Smart Card Interface Mode**  
(When Clock Frequency is 372 Times the Bit Rate)

### 15.6.5 Initialization (Smart Card Interface Mode)

Before transmitting and receiving data, initialize the SCI using the following procedure. Initialization is also necessary before switching from transmission to reception and vice versa.

1. Write the initial value “00h” to the SCR.
2. Make general-purpose I/O port settings to enable input and output functions as required for TXDn, RXDn, and SCKn pins.
3. Set the error flags ORER, ERS, and PER in SSR to 0.
4. Set bits GM, BLK, PM, BCP[1:0], and CKS[1:0] in SMR and the BCP2 bit in SCMR appropriately. Also set the PE bit in SMR to 1.
5. Set bits SDIR, SINV, and SMIF in SCMR appropriately. Then, the TXDn and RXDn pins are placed in the high impedance state.
6. Set the value corresponding to the bit rate in BRR.
7. Set the CKE[1:0] bits in SCR appropriately, and set bits TIE, RIE, TE, RE, and TEIE in SCR to 0 at the same time. When the CKE[1:0] bit in SCR is set to 1, the SCKn pin is allowed to output clock pulses.
8. Set the TIE, RIE, TE, and RE bits in SCR to 1. Setting the TE and RE bits to 1 simultaneously is prohibited except for self-diagnosis.

To change reception mode to transmission mode, first check that reception has completed, and then initialize the SCI. At the end of initialization, set SCR.TE = 1 and SCR.RE = 0. Reception completion can be verified by reading the RXI request, ORER, or PER flag in SSR.

To change transmission mode to reception mode, first check that transmission has completed, and then initialize the SCI. At the end of initialization, set SCR.TE = 0 and SCR.RE = 1. Transmission completion can be verified by reading the TEND flag in SSR.

### 15.6.6 Serial Data Transmission (Except in Block Transfer Mode)

Serial data transmission in smart card interface mode (except in block transfer mode), in that an error signal is sampled and data can be re-transmitted, is different from that in normal serial communications interface mode. Figure 15.30 shows the data retransfer operation during transmission.

1. When an error signal from the receiver end is sampled after one-frame data has been transmitted, the ERS flag in SSR is set to 1. If the RIE bit in SCR is 1 at this time, an ERI interrupt request is generated. Clear the ERS flag in SSR to 0 before the next parity bit is sampled.
2. For a frame in which an error signal is received, the TEND flag in SSR is not set. Data is retransferred from TDR to TSR allowing automatic data retransmission.
3. If no error signal is returned from the receiver, the ERS flag in SSR is not set to 1.
4. In this case, the SCI judges that transmission of one-frame data (including retransfer) has been completed, and the TEND flag is set. If the TIE bit in SCR is 1 at this time, a TXI interrupt request is generated. Writing transmit data to TDR starts transmission of the next data.

Figure 15.32 shows a sample flowchart of serial transmission. All the processing steps are automatically performed using a TXI interrupt request to activate the DMAC.

When the TEND flag in SSR is set to 1 in transmission, if the TIE bit in SCR is 1, a TXI interrupt request is generated. The DMAC is activated by a TXI interrupt request if the TXI interrupt request is specified as a source of DMAC activation beforehand, allowing transfer of transmit data. The TEND flag in SSR is automatically cleared to 0 when the DMAC transfers the data.

If an error occurs, the SCI automatically re-transmits the same data. During this retransmission, the TEND flag in SSR is kept to 0 and the DMAC is not activated. Therefore, the SCI and DMAC automatically transmit the specified number of bytes, including retransmission in the case of error occurrence. However, since the ERS flag in SSR is not automatically cleared, set the RIE bit in SCR to 1 beforehand to enable an ERI interrupt request to be generated at error occurrence, and clear the ERS flag in SSR to 0.

When transmitting/receiving data using the DMAC, be sure to make settings to enable the DMAC before making SCI settings.

For DMAC settings, see section 9, Direct Memory Access Controller.

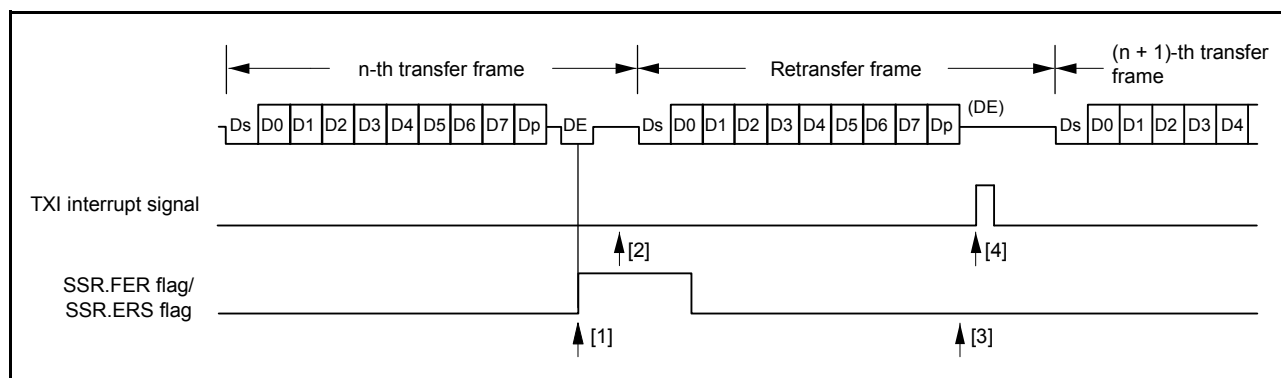


Figure 15.30 Data Retransfer Operation in Transmission Mode

Note that the SSR.TEND flag is set in different timings depending on the GM bit setting in SMR. Figure 15.31 shows the SSR.TEND flag generation timing.

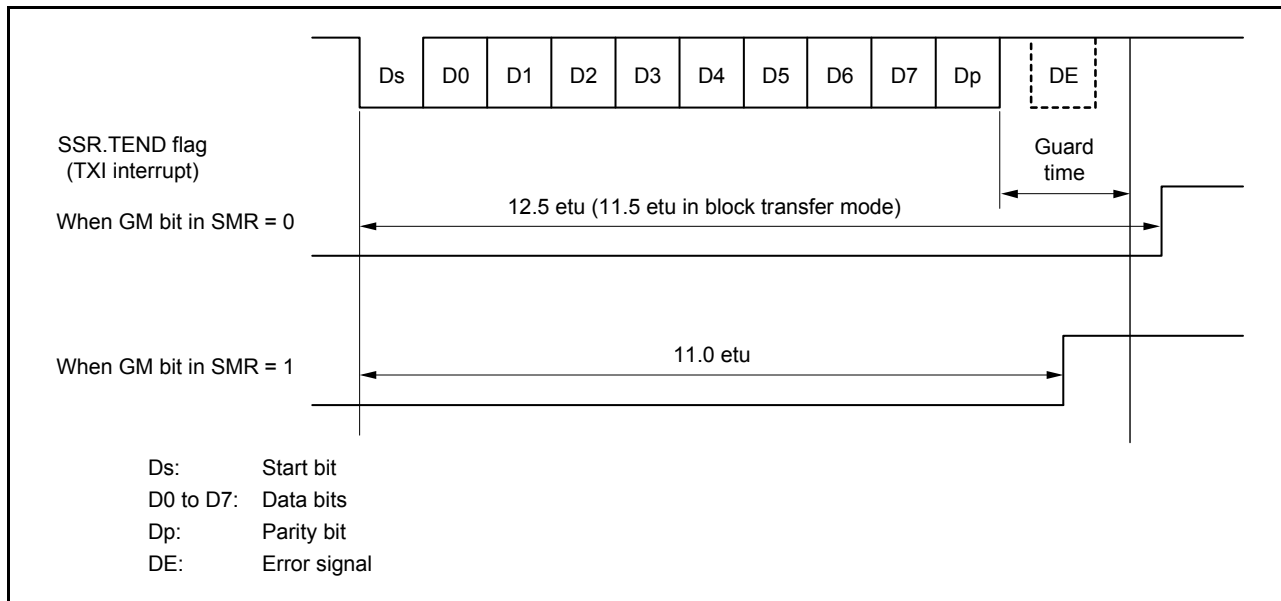


Figure 15.31 SSR.TEND Flag Generation Timing during Transmission

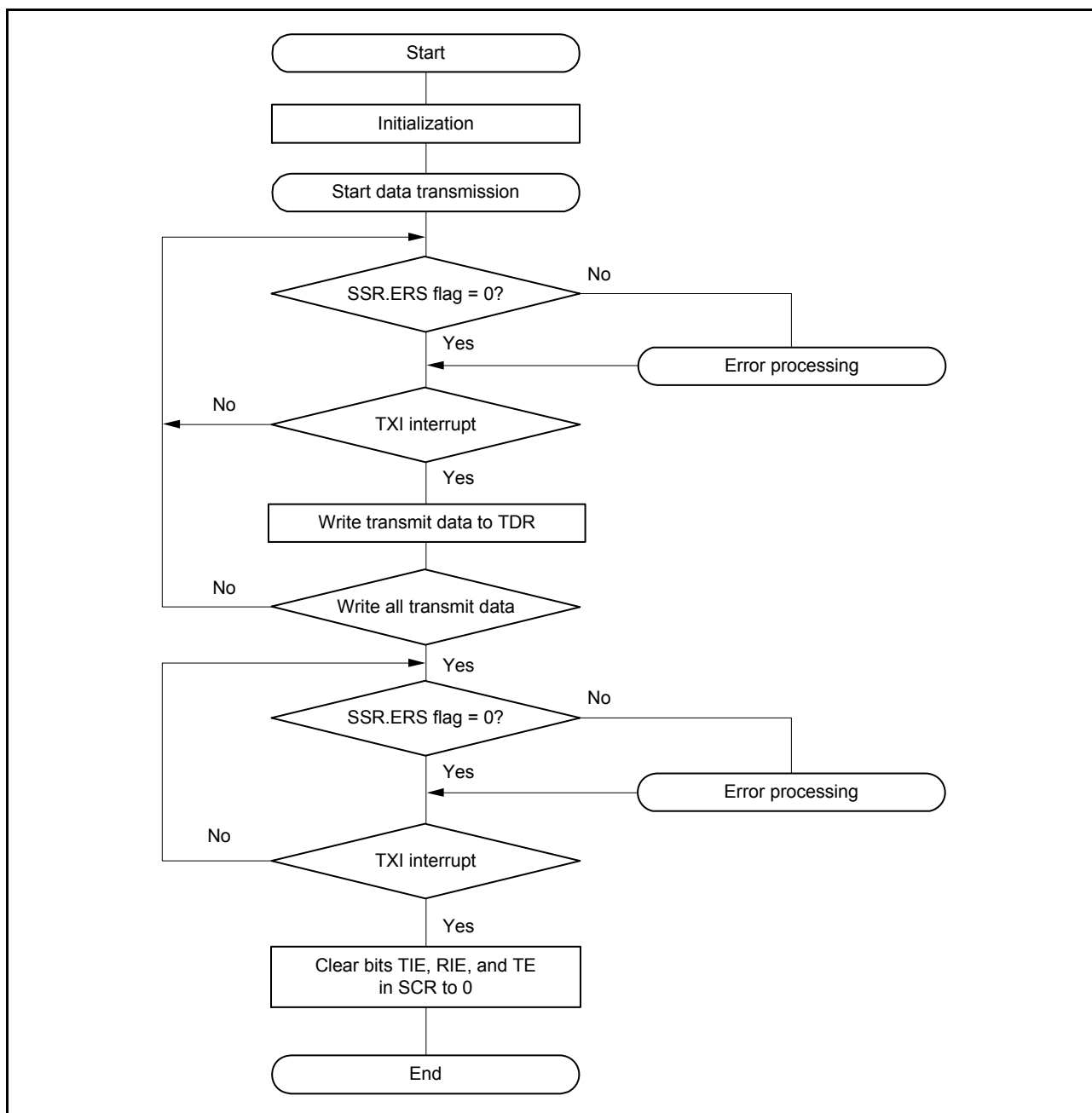


Figure 15.32 Sample Smart Card Interface Transmission Flowchart



### 15.6.7 Serial Data Reception (Except in Block Transfer Mode)

Serial data reception in smart card interface mode is similar to that in serial communications interface mode. Figure 15.33 shows the data retransfer operation in reception mode.

1. If a parity error is detected in receive data, the PER flag in SSR is set to 1. When the RIE bit in SCR is 1 at this time, an ERI interrupt request is generated. Clear the PER flag in SSR to 0 before the next parity bit is sampled.
2. For a frame in which a parity error is detected, no RXI interrupt is generated.
3. When no parity error is detected, the PER flag in SSR is not set to 1.
4. In this case, data is determined to have been received successfully. When the RIE bit in SCR is 1, an RXI interrupt request is generated.

Figure 15.34 shows a sample flowchart for serial data reception. All the processing steps are automatically performed using an RXI interrupt request to activate the DMAC.

In reception, setting the RIE bit in SCR to 1 allows an RXI interrupt request to be generated. The DMAC is activated by an RXI interrupt request if the RXI interrupt request is specified as a source of DMAC activation beforehand, allowing transfer of receive data.

If an error occurs during reception and either the ORER or PER flag in SSR is set to 1, a receive error interrupt (ERI) request is generated. Clear the error flag after the error occurrence. If an error occurs, the DMAC is not activated and receive data is skipped. Therefore, the number of bytes of receive data specified in the DMAC is transferred.

Even if a parity error occurs and the PER flag in SSR is set to 1 during reception, receive data is transferred to RDR, thus allowing the data to be read.

Note 1. For operations in block transfer mode, see section 15.3, Operation in Asynchronous Mode.

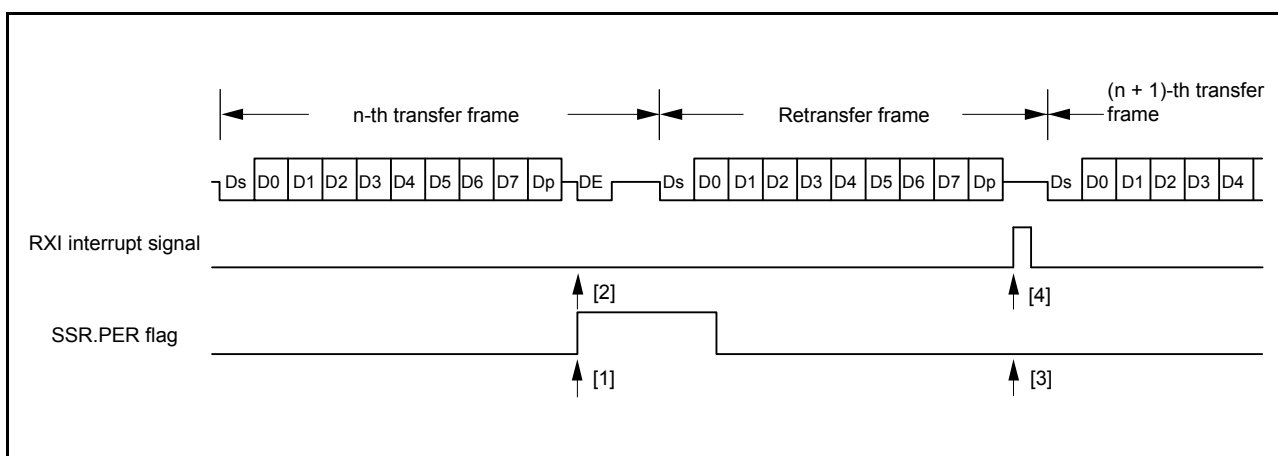


Figure 15.33 Data Retransfer Operation in Reception Mode

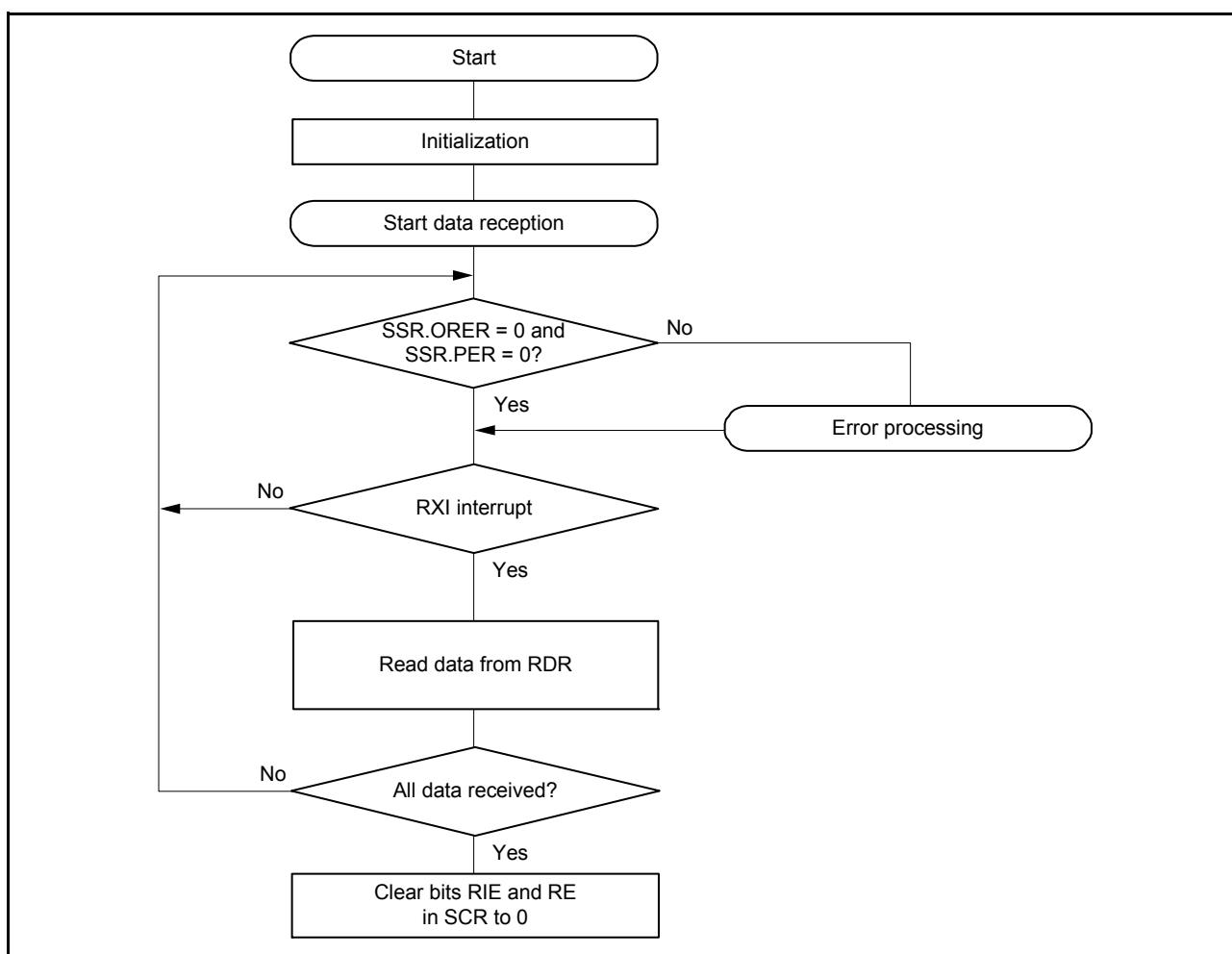


Figure 15.34 Sample Smart Card Interface Reception Flowchart

### 15.6.8 Clock Output Control

Clock output can be fixed using the CKE[1:0] bits in SCR when the GM bit in SMR is 1. Specifically, the minimum width of a clock pulse can be specified.

Figure 15.35 shows an example of clock output fixing timing when the CKE0 bit is controlled with SMR.GM = 1 and SCR.CKE1 = 0.

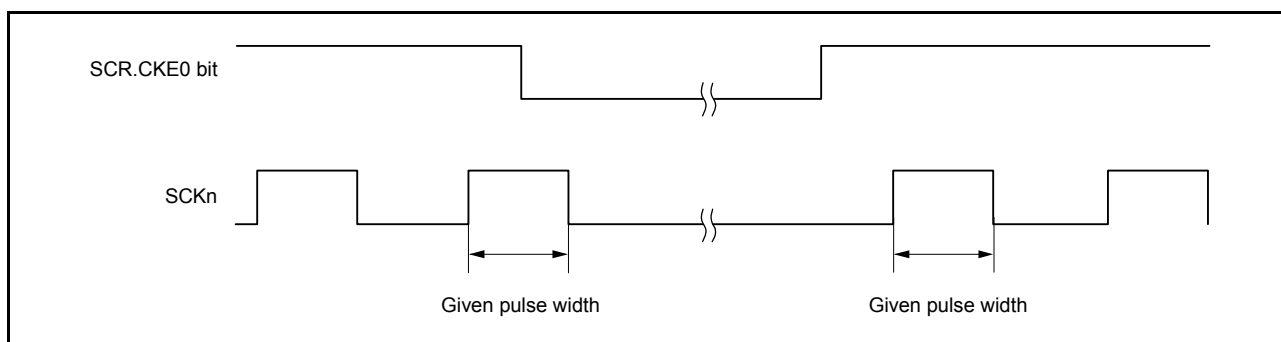


Figure 15.35 Clock Output Fixing Timing

At power-on and transitions to/from software standby mode, use the following procedure to secure the appropriate clock duty cycle.

### (1) At Power-On

To secure the appropriate clock duty cycle simultaneously with power-on, use the following procedure.

1. Initially, port input is enabled in the high-impedance state. To fix the potential level, use a pull-up or pull-down resistor.
2. Fix the SCKn pin to the specified output by setting the SCR.CKE[1] bit and general-purpose I/O port.
3. Set SMR and SCMR to enable smart card interface mode.
4. Set the SCR.CKE[0] bit to 1 to start clock output.

### (2) At Mode Switching

#### (a) At transition from smart card interface mode to software standby mode

1. Set low power consumption mode to make the SCKn pin fixed with a desired output value in software standby mode.
2. Write 0 to the TE and RE bits in SCR to stop transmission/reception.  
Simultaneously, set the SCR.CKE[1] bit to the value for the output fixed state in software standby mode.
3. Write 0 to the SCR.CKE[0] bit to stop the clock.
4. Wait for one cycle of the serial clock. In the mean time, the clock output is fixed to the specified level with the duty cycle retained.
5. After switching the SCKn pin to the general-purpose I/O port function, make a transition to software standby mode.

#### (b) Return from software standby mode to smart card interface mode

6. Cancel software standby mode.
7. Set the SCR.CKE[0] bit to 1 to start clock output. A clock signal with the appropriate duty cycle is then generated.

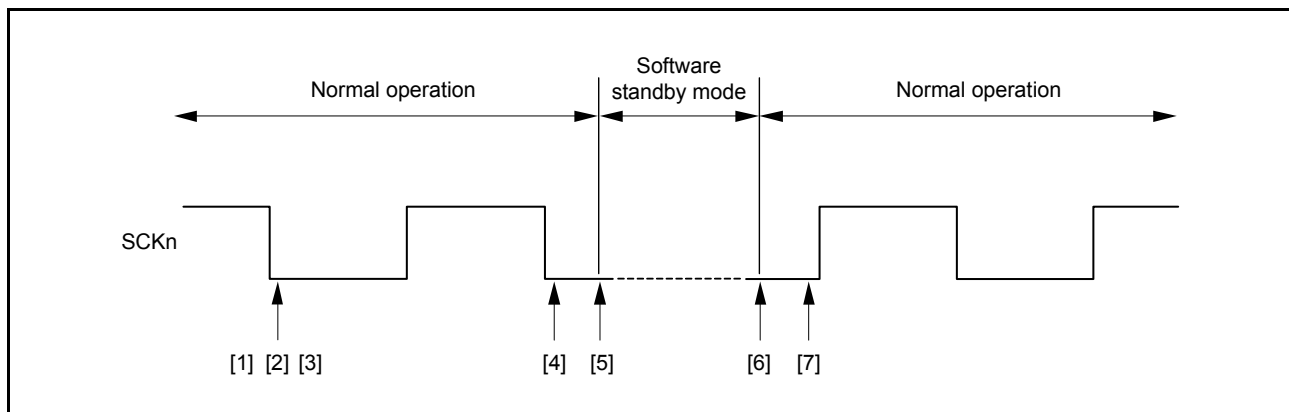


Figure 15.36 Clock Stop and Restart Procedure

## 15.7 Noise Cancellation Function

Figure 15.37 shows the configuration of the noise filter used for noise cancellation. The noise filter consists of two stages of flip-flop circuits and a match-detection circuit. When the level on the pin matches in three consecutive samples taken at the set sampling interval, the matching level continues to be conveyed internally until the level on the pin again matches in three consecutive samples.

In asynchronous mode, the noise cancellation function can be applied on the RXDn input signal. The period of the base clock ( $1/16$ th of a bit-period when SEMR.ABCS = 0 and  $1/8$ th of a bit-period when SEMR.ABCS = 1) is the sampling interval.

If the base clock is stopped with the noise filter enabled and then the clock input is started again, the noise filter operation resumes from where the clock was stopped. If SCR.TE and SCR.RE are set to 0 during base clock input, all of the noise filter flip-flop values are initialized to 1. Accordingly, if the input data is 1 when reception operation resumes, it is determined that a level match is detected and is conveyed to the internal signal. When the level being input corresponds to 0, the initial output of the noise filter is retained until the level matches in three consecutive samples.

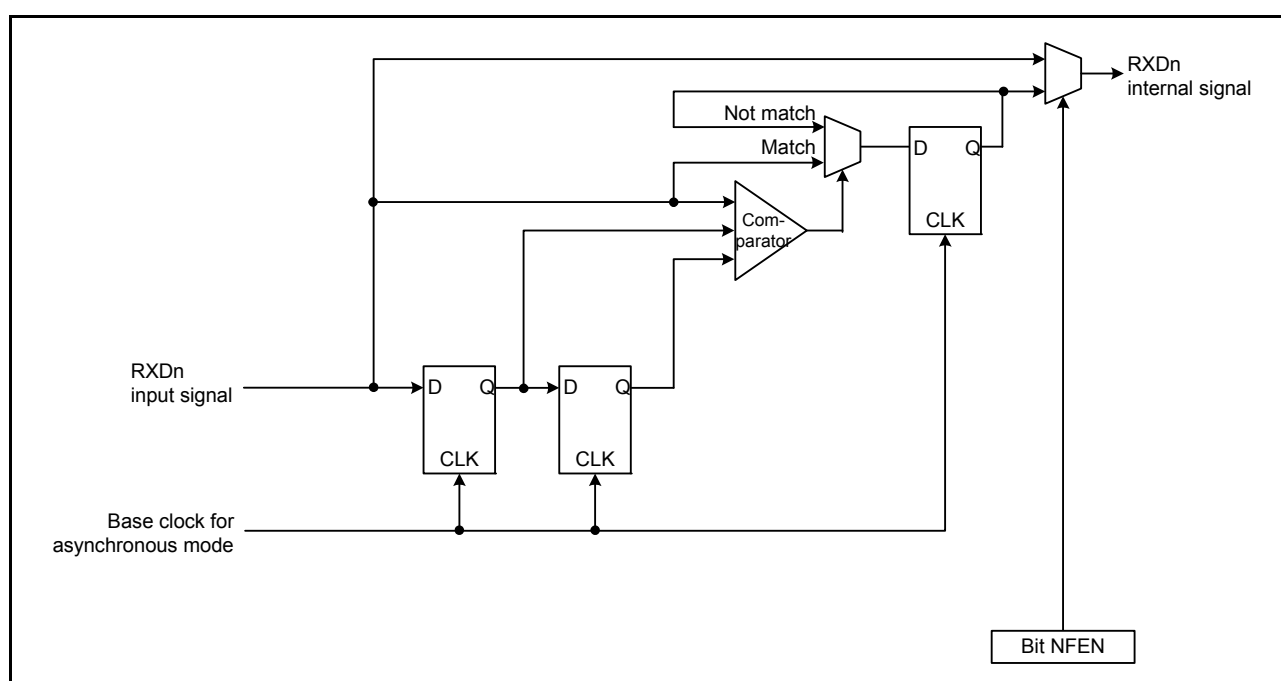


Figure 15.37 Block Diagram of Digital Noise Filter Circuit

## 15.8 Interrupt Sources

### 15.8.1 Interrupts in Serial Communications Interface Mode

Table 15.16 lists interrupt sources in serial communication interface mode. Individual interrupt sources can be enabled or disabled with the enable bits in SCR.

If the SCR.TIE bit is 1, a TXI interrupt request is generated when data for transmission are transferred from the TDR to the TSR. A TXI interrupt request can also be generated by setting the SCR.TE bit to 1 after setting the SCR.TIE bit to 1 or by using a single instruction to set the SCR.TE and SCR.TIE bit to 1 at the same time. A TXI interrupt request can activate the DMAC to handle data transfer.

A TXI interrupt request is not generated by setting the SCR.TE bit to 1 while the setting of the SCR.TIE bit is 0 or by setting the SCR.TIE bit to 1 while the setting of the SCR.TE bit is 1.\*<sup>1</sup>

When new data are not written by the time of transmission of the last bit of the current data for transmission and the setting of the SCR.TEIE bit is 1, the SSR.TEND flag becomes 1 and a TEI interrupt request is generated. Furthermore, when the setting of the SCR.TE bit is 1, the SSR.TEND flag retains the value 1 until further data for transmission are written to the TDR, and setting the SCR.TEIE bit to 1 leads to the generation of a TEI interrupt request.

Writing data to the TDR leads to clearing of the SSR.TEND flag and, after a certain time, discarding of the TEI interrupt request.

If the SCR.RIE bit is 1, an RXI interrupt request is generated when received data are stored in the RDR. An RXI interrupt request can activate the DMAC to handle data transfer.

Setting of any from among the ORER, FER, and PER flags in the SSR to 1 while the SCR.RIE bit is 1 leads to the generation of an ERI interrupt request. An RXI interrupt request is not generated at this time. Clearing all three flags (ORER, FER, and PER) leads to discarding of the ERI interrupt request.

Note 1. To temporarily prohibit TXI interrupts at the time of transmission of the last of the data and so on when you wish a new round of transmission to start after handling of the transmission-completed interrupt, control prohibiting and permitting of the interrupt by using the interrupt request enable bit in the interrupt controller rather than using the SCR.TIE bit. This can prevent the suppression of TXI interrupt requests in the transfer of new data.

**Table 15.16 Interrupt Sources**

Name	Interrupt Source	Interrupt Flag	DMAC Activation
ERI	Receive error	ORER, FER, or PER	Not possible
RXI	Receive data full	—	Possible
TXI	Transmit data empty	—	Possible
TEI	Transmit end	TEND	Not possible

## 15.8.2 Interrupts in Smart Card Interface Mode

Table 15.17 lists interrupt sources in smart card interface mode. A transmit end interrupt (TEI) request cannot be used in this mode.

**Table 15.17 Interrupt Sources**

Name	Interrupt Source	Interrupt Flag	DMAC Activation
ERI	Receive error or error signal detection	ORER, PER, or ERS	Not possible
RXI	Receive data full	—	Possible
TXI	Transmit data empty	TEND	Possible

Data transmission/reception using the DMAC is also possible in smart card interface mode. In transmission, when the TEND flag in SSR is set to 1, a TXI interrupt request is generated. This TXI interrupt request activates the DMAC allowing transfer of transmit data if the TXI request is specified beforehand as a source of DMAC activation. The TEND flag in SSR is automatically cleared to 0 when the DMAC transfers the data.

If an error occurs, the SCI automatically re-transmits the same data. During the retransmission, the TEND flag in SSR is kept to 0 and the DMAC is not activated. Therefore, the SCI and DMAC automatically transmit the specified number of bytes, including retransmission in the case of error occurrence. However, the ERS flag in SSR is not automatically cleared to 0 at error occurrence. Therefore, the ERS flag in SSR must be cleared by previously setting the RIE bit in SCR to 1 to enable an ERI interrupt request to be generated at error occurrence.

When transmitting/receiving data using the DMAC, be sure to make settings to enable the DMAC before making SCI settings. For DMAC settings, see section 9, Direct Memory Access Controller.

In reception, an RXI interrupt request is generated when receive data is set to RDR. This RXI interrupt request activates the DMAC allowing transfer of receive data if the RXI request is specified beforehand as a source of DMAC activation. If an error occurs, the error flag is set. Therefore, the DMAC is not activated and an ERI interrupt request is issued to the CPU instead; the error flag must be cleared.

## 15.9 Usage Notes

### 15.9.1 Setting the Module Standby Function

SCI operation can be started and stopped by setting the module standby mode. With the value after a reset, SCI operations are stopped. The registers of the modules only become accessible after release from the module standby state. For details, refer to section 55, Power-Down Modes.

### 15.9.2 Break Detection and Processing

When a framing error is detected, a break can be detected by reading the RXDn pin value directly. In a break, the input from the RXDn pin becomes all 0s, and so the FER flag in SSR is set to 1 (framing error), and the PER flag in SSR may also be set to 1 (parity error). The SCI continues the receive operation even after a break is received. Therefore, note that even if the FER flag in SSR is cleared to 0 (no framing error), it will be set to 1 again.

### 15.9.3 The Mark State and Production of Breaks

When the SCR.TE bit is 0 (prohibiting serial transmission), setting the general-purpose I/O port makes selection of the level and direction (input or output) of the TXDn pin possible. If this is done, the TXDn pin can be placed in the mark state to send a break at the time of data transmission. Until the SCR.TE bit is set to 1 (permitting serial transmission), the general-purpose I/O port is used to set the TXDn pin to output 1 and set the pin mode to a general-purpose I/O port pin, and thus place the transfer circuit in the mark state (state of having the value 1). On the other hand, to output a break at the time of data transmission, set the TXDn pin to output 0 and make the pin mode settings for a general-purpose I/O port pin. When the SCR.TE bit is set to 0, the transmission section is initialized regardless of the current state of transmission.

### 15.9.4 Receive Error Flags and Transmit Operations (Clock Synchronous Mode Only)

Transmission cannot be started when a receive error flag (ORER) in SSR is set to 1, even if data is written to TDR. Be sure to clear the receive error flags to 0 before starting transmission. Note also that the receive error flags cannot be cleared to 0 even if the RE bit in SCR is cleared to 0 (serial reception disabled).

### 15.9.5 Writing Data to TDR

Data can always be written to TDR. However, if new data is written to TDR when transmit data is remaining in TDR, the previous data in TDR is lost because it has not been transferred to TSR yet. Be sure to write transmit data to TDR in the TXI interrupt request processing routine.

### 15.9.6 Restrictions on Clock Synchronous Transmission

When the external clock source is used as a synchronization clock, update TDR by the DMAC and wait for at least five P1 $\phi$  clock cycles before allowing the transmit clock to be input. If the transmit clock is input within four clock cycles after TDR is updated, the SCI may malfunction.

### 15.9.7 Restrictions on Using DMAC

When using the DMAC to read RDR, be sure to set the receive end interrupt (RXI) as the activation source of the relevant channel.

### 15.9.8 Points to Note on Starting Transfer

On the generation of an interrupt request for the interrupt controller at the point where transfer starts, follow the procedure below to clear interrupt requests before permitting operations (by setting the SCR.TE or SCR.RE bit to 1).

- Confirm that transfer has stopped (the setting of the SCR.TE or SCR.RE bits is 0).
- Set the corresponding interrupt enable bit (SCR.TIE or SCR.RIE) to 0.
- Read out the corresponding interrupt enable bit (SCR.TIE or SCR.RIE bit) to check that it has actually become 0.
- Clear the interrupt status flag in the interrupt controller to 0.

### 15.9.9 SCI Operations during Low Power Consumption State

#### (1) Transmission

When making settings for the module standby state or in transitions to software standby, stop operations (by setting the TIE, TE, and TEIE bits in the SCR to 0) after switching the TXDn pin to the general-purpose I/O port pin function. Clearing the TE bit in SCR to 0 resets the TSR and the TEND bit in the SSR. The states of the output pins in the software standby mode depend on the settings for the power-down modes. When transitions to these states are made during transmission, the data being transmitted become indeterminate.

To transmit data in the same transmission mode after cancellation of the low power consumption state, set the TE bit in SCR to 1, read SSR, and write data to TDR sequentially to start data transmission. To transmit data with a different transmission mode, initialize the SCI first.

Figure 15.38 shows a sample flowchart for transition to software standby mode during transmission.

#### (2) Reception

Before specifying the module standby state or making a transition to software standby mode, stop the receive operations (RE = 0 in SCR). If transition is made during data reception, the data being received will be invalid.

To receive data in the same reception mode after cancellation of the low power consumption state, set the RE bit in SCR to 1, and then start reception. To receive data in a different reception mode, initialize the SCI first.



Figure 15.39 shows a sample flowchart for transition to software standby mode during reception.

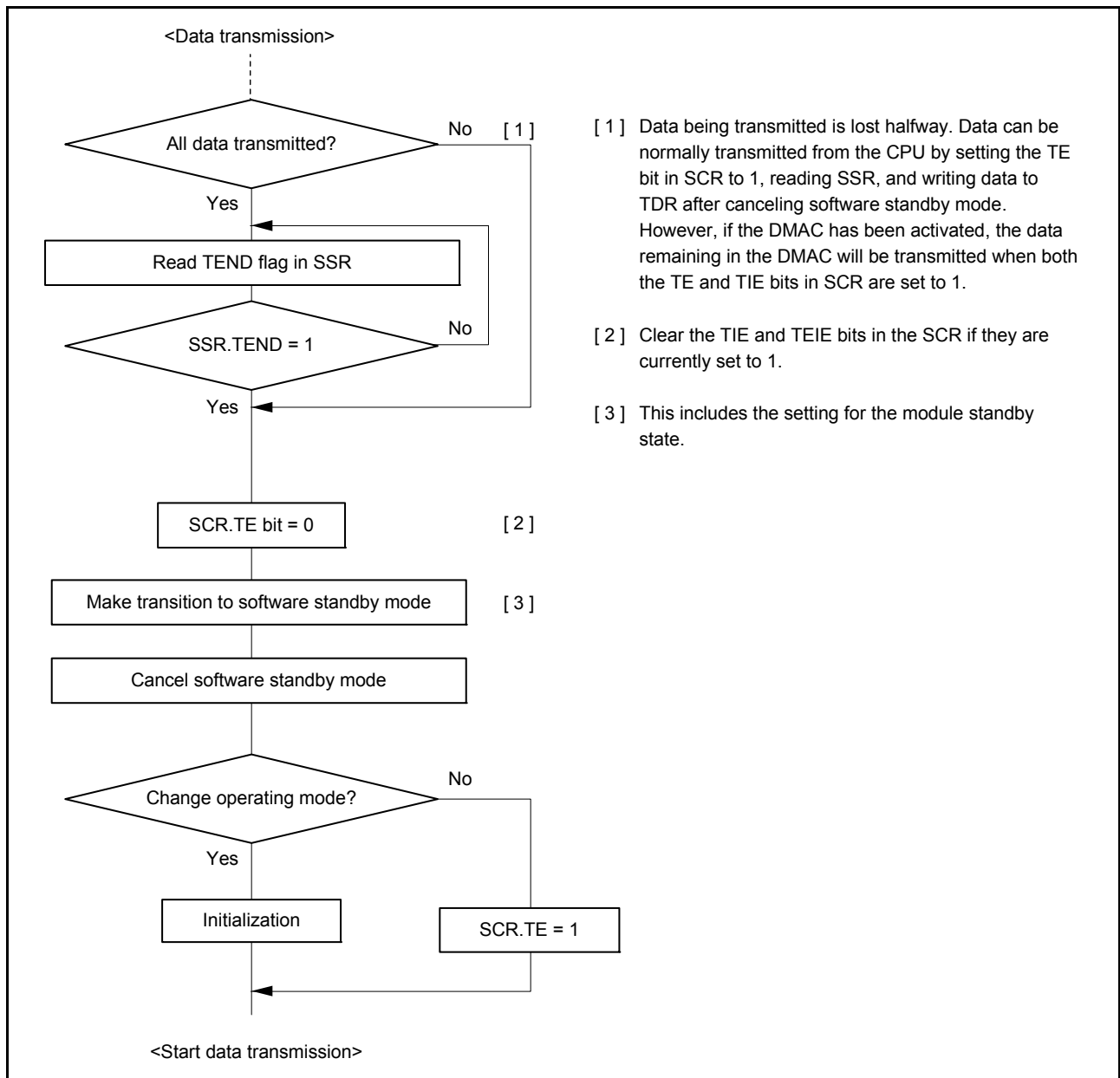


Figure 15.38 Example of Flowchart for Transition to Software Standby Mode during Transmission

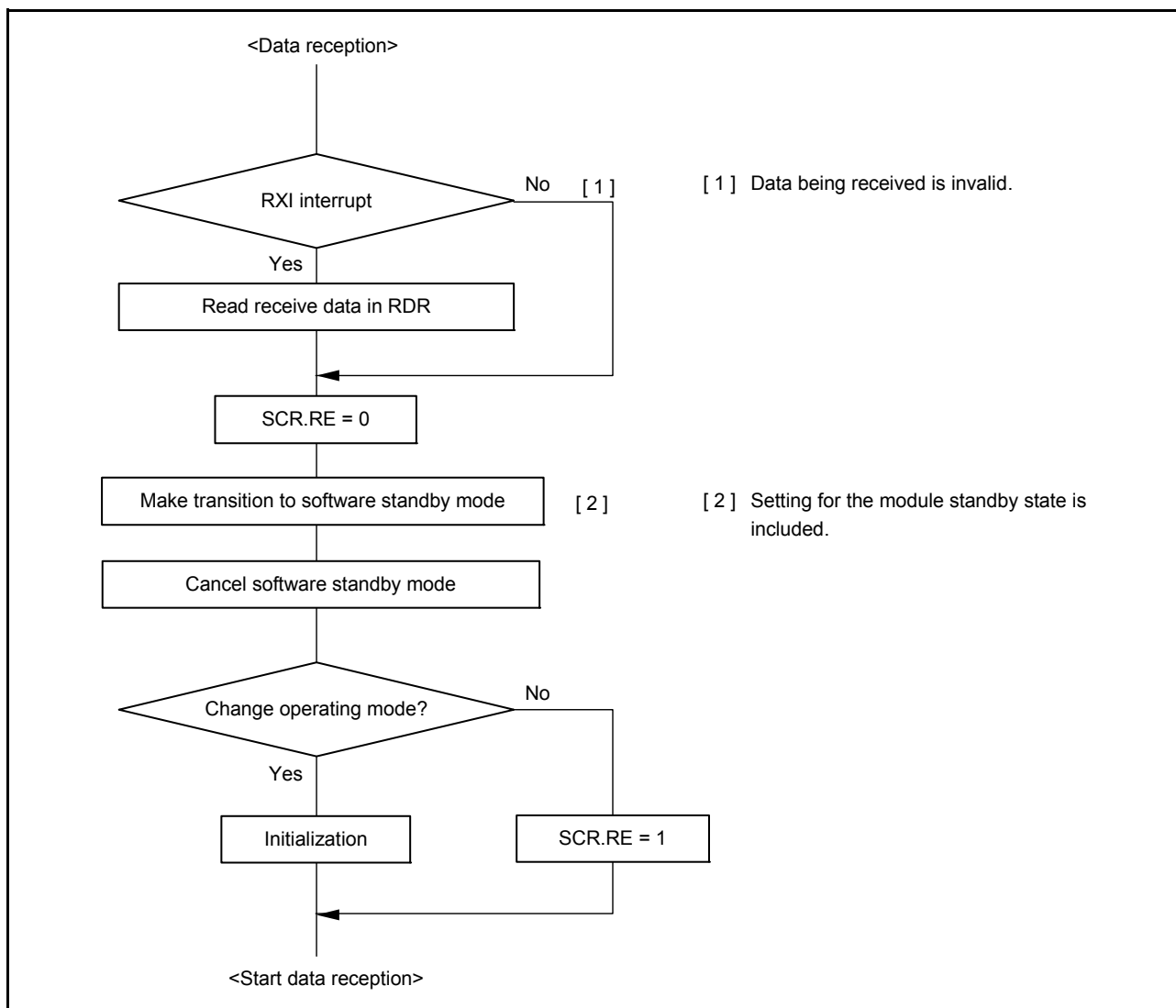


Figure 15.39 Example of Flowchart for Transition to Software Standby Mode during Reception

### 15.9.10 External Clock Input in Clock Synchronous Mode

In clock synchronous mode, the external clock SCKn must be input as follows:

High-pulse period, low-pulse period = 2 P1 $\phi$  clock cycles or more, period = 6 P1 $\phi$  clock cycles or more

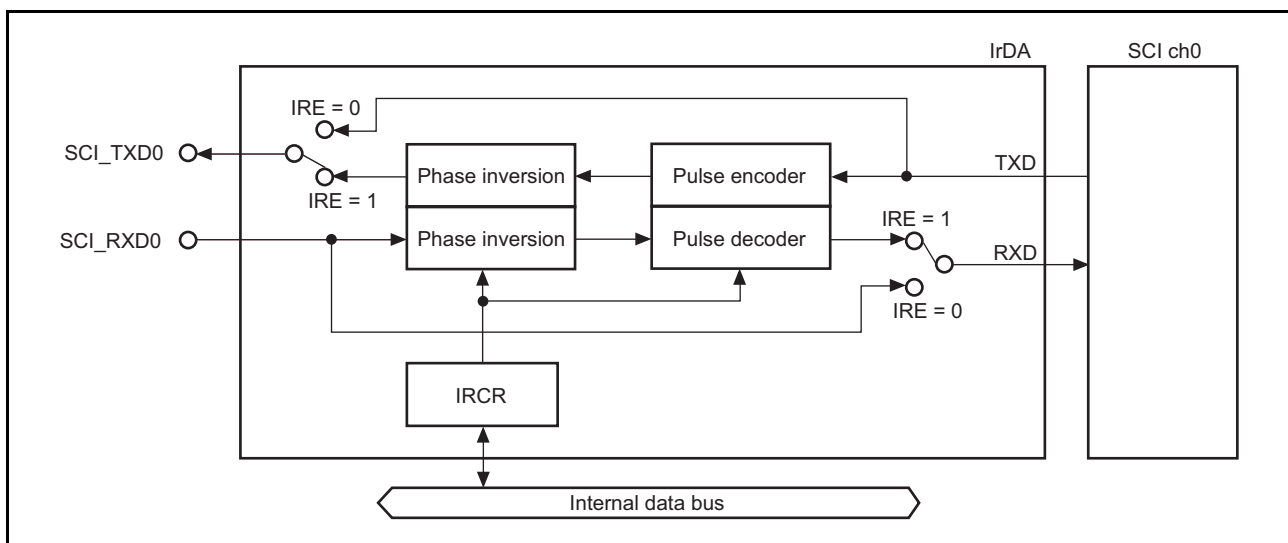
## 15.10 IrDA Communications

In combination with the on-chip IrDA module, the channel 0 serial communications interface (SCI) transmits and receives waveforms conforming with version 1.0 of the Infrared Data Association (IrDA) standard.

When the IrDA function is enabled by the IRE bit in the IRCR register, the SCI\_TXD0 and SCI\_RXD0 signals transmitted and received on channel 0 are encoded to and decoded from waveforms conforming with the standard. Connecting these signals to an infrared transceiver enables infrared transmission and reception conforming with the standard.

The standard allows starting communications at a transfer rate of 9600 bps and changing the rate as required. The IrDA module does not have the function of automatically changing the transfer rate. To change the transfer rate of IrDA communications, change the transfer rate of the SCI.

Figure 15.40 shows the block diagram. Table 15.18 shows the pin configuration of the IrDA.



**Figure 15.40** Block Diagram

**Table 15.18** Pin Configuration

Name	Pin Name	I/O	Description
IrDA transmit data pin	SCI_TXD0	O	IrDA transmit data input
IrDA receive data pin	SCI_RXD0	I	IrDA receive data input

## 15.11 IrDA Register Description

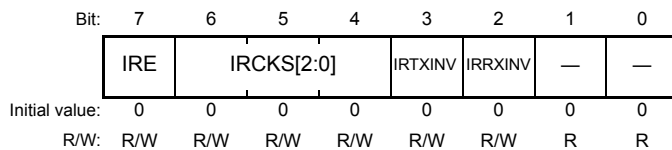
Table 15.19 shows the register configuration.

**Table 15.19 Register Configuration**

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
IrDA control register	IRCR	R/W	H'00	H'E8014000	8

### 15.11.1 IrDA Control Register (IRCR)

IRCR is the register which sets the operation of the IrDA module.



Bit	Bit Name	Initial Value	R/W	Description
7	IRE	0	R/W	IrDA Enable Sets whether the SCI_TXD0 and SCI_RXD0 pins have normal serial function or IrDA function. 0: IrDA function is disabled (The TXD signal transmitted on channel 0 is directly output to the SCI_TXD0 pin) The RXD signal received on channel 0 is directly input to the SCI_RXD0 pin) 1: IrDA function is enabled (The TXD signal transmitted on channel 0 is output to the SCI_TXD0 pin after encoded) The RXD signal received on channel 0 is input to the SCI_RXD0 pin after decoded)
6 to 4	IRCKS[2:0]	000	R/W	IrDA Clock Select Sets the pulse width when the SCI_TXD0 output pulse is encoded with the IRE bit set to 1. 000: $B \times 3/16$ (B = bit rate) 001: $P1\phi/2$ 010: $P1\phi/4$ 011: $P1\phi/8$ 100: $P1\phi/16$ 101: $P1\phi/32$ 110: $P1\phi/64$ 111: $P1\phi/128$
3	IRTXINV	0	R/W	SCI_TXD0 Data Sense Switch Sets whether or not to invert the logic levels for SCI_TXD0 output. 0: Data for transmission are directly output to SCI_TXD0. The pulse width set by the IRCKS bits is the width at the high level. 1: Data for transmission are output to SCI_TXD0 after inversion. The pulse width set by the IRCKS bits is the width at the low level.
2	IRRXINV	0	R/W	SCI_RXD0 Data Sense Switch Sets whether or not to invert the logic levels of SCI_RXD0 input. 0: SCI_RXD0 input is directly used as receive data. 1: SCI_RXD0 input is used as receive data after inversion.
1, 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

## 15.12 IrDA Operation

### 15.12.1 Flow of IrDA Setting

To set the operation of the IrDA module, follow the procedure below.

1. Make the general-purpose I/O port settings.
2. Set the IRCR register.
3. Set the relevant registers of the SCI.

### 15.12.2 Transmission

In transmission with the IrDA function enabled, serial data from the TXD pin for the SCI (UART frame data) are converted to IR frames (see Figure 15.41). When the IRTXINV bit is 0 and the value of the serial data is 0, a high-level pulse is output to the SCI\_TXD0 pin for three sixteenths of one-bit period (initial value). The width of the high-level pulse can be changed by the IRCKS[2:0] bits in the IRCR register. The IrDA standard stipulates that the high-level pulse width is at least 1.41  $\mu\text{s}$  and no greater than  $((3/16 + 2.5\%) \times \text{bit rate})$  or  $((3/16 \times \text{bit rate}) + 1.08) \mu\text{s}$ . When P1 $\phi$  is 66.67 MHz, the width of the high-level pulse can be set from 1.41  $\mu\text{s}$  to 1.92  $\mu\text{s}$ . When the value of a bit of the serial data is 1, no pulse is output.

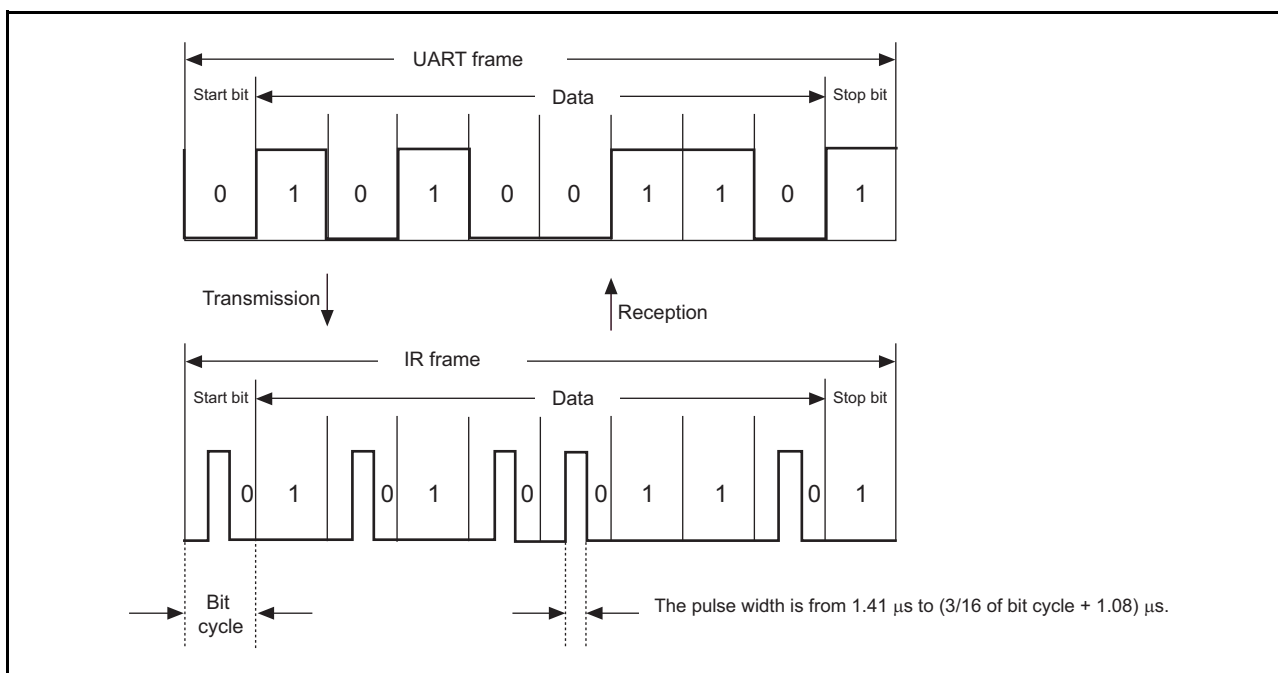


Figure 15.41 Example of IrDA Transmission and Reception

### 15.12.3 Reception

In reception with the IRDA function enabled, IR frame data from the SCI\_RXD0 pin are converted to serial data and output to the RXD pin for the SCI. When the IRRXINV bit is 0, a bit with the value 0 is output on the detection of a high-level pulse. When no pulse is received during one-bit period, a bit with the value 1 is output. Pulses shorter than the lower limit (1.41  $\mu$ s) are not recognized.

### 15.12.4 Selection of High-Level Pulse Width

Table 15.20 shows the correspondence between the applicable IRCKS[2:0] bit setting (shortest pulse width), operating frequency  $P1\phi$ , and bit rate when the pulse width is shortened below ( $\text{bit rate} \times 3/16$ ) in transmission.

**Table 15.20 IRCKS[2:0] Bit Setting**

	Bit Rate (bps) (Upper Row) / Bit Cycle $\times$ 3/16 ( $\mu$ s) (Lower Row)					
	2400	9600	19200	38400	57600	115200
$P1\phi$ (MHz)	78.13	19.53	9.77	4.88	3.26	1.63
50	111	111	111	111	111	—*1
64	111	111	111	111	111	—*2
66.67	111	111	111	111	111	—*2

Note 1. The bit rate cannot be set at the SCI.

Note 2. A pulse width shorter than ( $\text{bit rate} \times 3/16$ ) cannot be set.

## 15.13 Notes on Using the IrDA Module

### 15.13.1 Shortest Pulse Width in Reception

Pulses shorter than the lower limit (1.41  $\mu$ s) are not recognized.

### 15.13.2 Asynchronous Basic Clock for Serial Communication Interface

The IrDA module receives the basic clock with a frequency which is 16 times as high as the communication bit rate from the SCI and operates in combination with the clock. Although one-bit period of the SCI can be set to 16 or 8 clock cycles, the IrDA module supports the SCI that one-bit period is set to 16 clock cycles.

## 16. Renesas Serial Peripheral Interface

This LSI circuit includes five independent Renesas serial peripheral interfaces.

This module is capable of full-duplex synchronous serial communication.

### 16.1 Features

This module has the following features.

- SPI transfer functions
  - Use of MOSI (master out/slave in), MISO (master in/slave out), SSL (slave select), and RSPCK (SPI clock) signals allows for serial communications through SPI operation (four-wire method).
  - Capable of serial communications in master/slave mode
  - Supports mode fault error detection (only in SPI slave mode)
  - Supports overrun error detection (only in SPI slave mode)
  - Switching of the polarity of the serial transfer clock
  - Switching of the clock phase of serial transfer
- Data format
  - MSB-first/LSB-first selectable
  - Transfer bit-length is selectable as 8, 16, or 32 bits.
- Bit rate
  - RSPCK can be divided by a maximum of 4096 in master mode
  - RSPCK can be generated by dividing P1 $\phi$  by the on-chip baud rate generator.
  - An externally input clock can be used as a serial clock.
- Buffer configuration
  - 8 bytes for transmission and 32 bytes for reception
- SSL control function
  - One SSL signal for each channel
  - In master mode, outputs SSL signal.
  - In slave mode, inputs SSL signal.
  - Controllable delay from SSL output assertion to RSPCK operation (RSPCK delay)
    - Range: 1 to 8 RSPCK cycles (set in RSPCK-cycle units)
  - Controllable delay from RSPCK stoppage to SSL output negation (SSL negation delay)
    - Range: 1 to 8 RSPCK cycles (set in RSPCK-cycle units)
  - Controllable wait for next-access SSL output assertion (next-access delay)
    - Range: 1 to 8 RSPCK cycles (set in RSPCK-cycle units)
  - Function for changing SSL polarity
- Control in master transfer
  - A transfer of up to four commands can be executed sequentially in looped execution.
  - For each command, the following can be set:
    - SSL signal value, bit rate, RSPCK polarity/phase, transfer data length, LSB/MSB first, burst, RSPCK delay, SSL negation delay, and next-access delay.
  - A transfer can be initiated by writing to the transmit buffer.
  - A transfer can be initiated by clearing the SPTEF bit.
  - MOSI signal value specifiable in SSL negation
- Interrupt sources
  - Maskable interrupt sources:
    - Receive interrupt (receive buffer full)
    - Transmit interrupt (transmit buffer empty)
    - Error interrupt (mode fault, overrun)



- Others
  - Provides loop back mode
  - Provides a function for disabling (initializing) this module

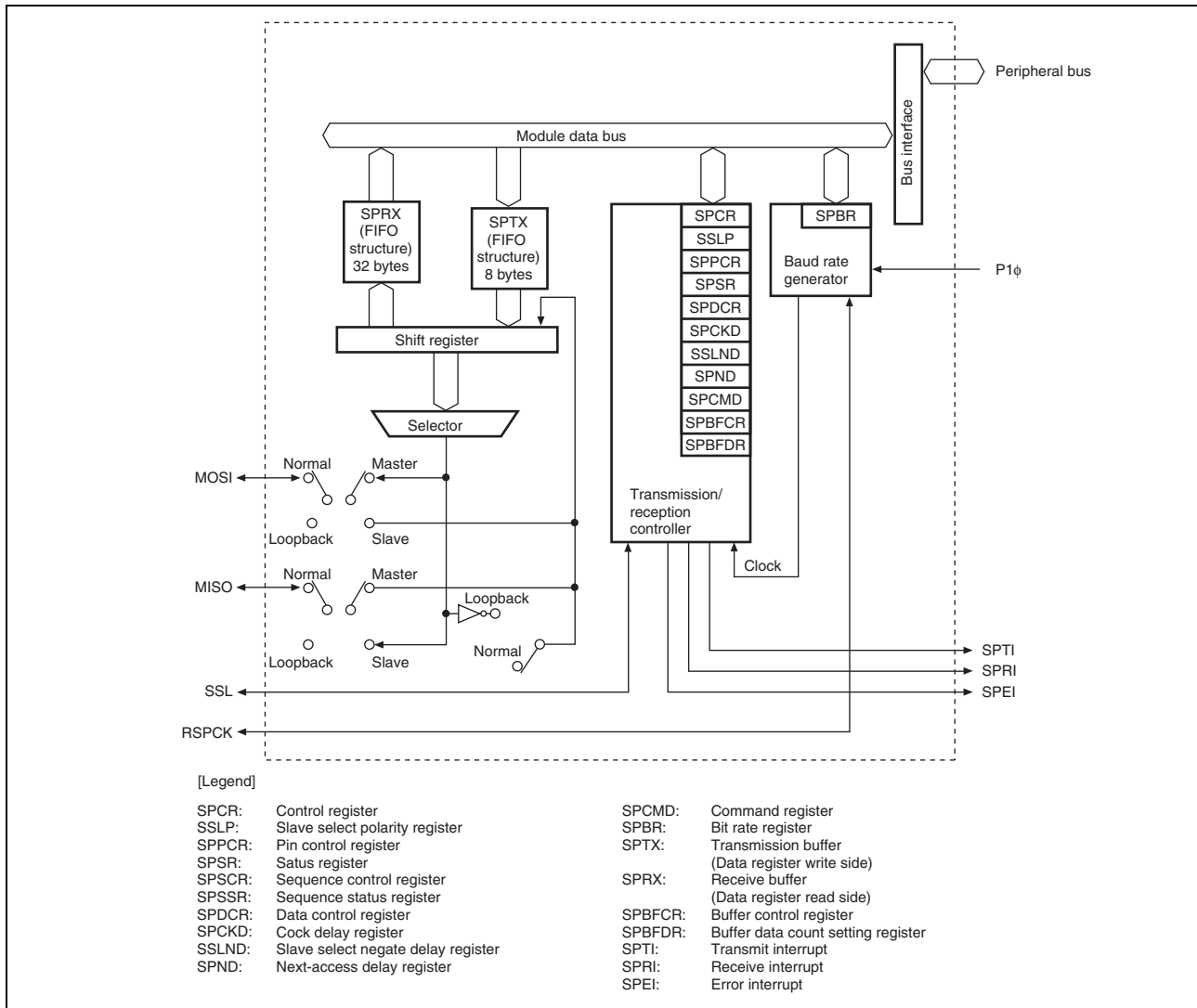


Figure 16.1 Block Diagram (for One Channel)

## 16.2 Input/Output Pins

Table 16.1 shows the pin configuration. This module automatically switches the input/output direction of the SSL pin. SSL is set as an output in master mode and as an input in slave mode. Pins RSPCK, MOSI, and MISO are automatically set as inputs or outputs according to the setting of master or slave and the level input on SSL (see section 16.4.2, Pin Control).

**Table 16.1 Pin Configuration**

Channel	Pin Name	Pin Name	I/O	Function
0	Clock pin	RSPCK0	I/O	Clock input/output
	Master transmit data pin	MOSI0	I/O	Master transmit data
	Slave transmit data pin	MISO0	I/O	Slave transmit data
	Slave select 0 pin	SSL00	I/O	Slave selection
1	Clock pin	RSPCK1	I/O	Clock input/output
	Master transmit data pin	MOSI1	I/O	Master transmit data
	Slave transmit data pin	MISO1	I/O	Slave transmit data
	Slave select 0 pin	SSL10	I/O	Slave selection
2	Clock pin	RSPCK2	I/O	Clock input/output
	Master transmit data pin	MOSI2	I/O	Master transmit data
	Slave transmit data pin	MISO2	I/O	Slave transmit data
	Slave select 0 pin	SSL20	I/O	Slave selection
3	Clock pin	RSPCK3	I/O	Clock input/output
	Master transmit data pin	MOSI3	I/O	Master transmit data
	Slave transmit data pin	MISO3	I/O	Slave transmit data
	Slave select 0 pin	SSL30	I/O	Slave selection
4	Clock pin	RSPCK4	I/O	Clock input/output
	Master transmit data pin	MOSI4	I/O	Master transmit data
	Slave transmit data pin	MISO4	I/O	Slave transmit data
	Slave select 0 pin	SSL40	I/O	Slave selection

Note: • In the description of the pins, the channel is omitted and pin names are described as RSPCK, MOSI, MISO, and SSL.

### 16.3 Register Descriptions

Table 16.2 shows the register configuration. These registers enable this module to perform the following controls: specifying master/slave modes, specifying a transfer format, and controlling the transmitter and receiver.

**Table 16.2 Register Configuration**

Channel	Register Name	Abbreviation*1	R/W	Initial Value	Address	Access Size	
0	Control register_0	SPCR_0	R/W	H'00	H'E800C800	8	
	Slave select polarity register_0	SSLP_0	R/W	H'00	H'E800C801	8	
	Pin control register_0	SPPCR_0	R/W	H'00	H'E800C802	8	
	Status register_0	SPSR_0	R/(W)*2	H'60	H'E800C803	8	
	Data register_0	SPDR_0	R/W	Undefined	H'E800C804	8, 16, 32	
	Sequence control register_0	SPSCR_0	R/W	H'00	H'E800C808	8	
	Sequence status register_0	SPSSR_0	R	H'00	H'E800C809	8	
	Bit rate register_0	SPBR_0	R/W	H'FF	H'E800C80A	8	
	Data control register_0	SPDCR_0	R/W	H'20	H'E800C80B	8	
	Clock delay register_0	SPCKD_0	R/W	H'00	H'E800C80C	8	
	Slave select negation delay register_0	SSLND_0	R/W	H'00	H'E800C80D	8	
	Next-access delay register_0	SPND_0	R/W	H'00	H'E800C80E	8	
	Command register0_0	SPCMD0_0	R/W	H'070D	H'E800C810	16	
	Command register1_0	SPCMD1_0	R/W	H'070D	H'E800C812	16	
	Command register2_0	SPCMD2_0	R/W	H'070D	H'E800C814	16	
	Command register3_0	SPCMD3_0	R/W	H'070D	H'E800C816	16	
	Buffer control register_0	SPBFCR_0	R/W	H'00	H'E800C820	8	
	Buffer data count setting register_0	SPBFDR_0	R	H'0000	H'E800C822	16	
	1	Control register_1	SPCR_1	R/W	H'00	H'E800D000	8
		Slave select polarity register_1	SSLP_1	R/W	H'00	H'E800D001	8
Pin control register_1		SPPCR_1	R/W	H'00	H'E800D002	8	
Status register_1		SPSR_1	R/(W)*2	H'60	H'E800D003	8	
Data register_1		SPDR_1	R/W	Undefined	H'E800D004	8, 16, 32	
Sequence control register_1		SPSCR_1	R/W	H'00	H'E800D008	8	
Sequence status register_1		SPSSR_1	R	H'00	H'E800D009	8	
Bit rate register_1		SPBR_1	R/W	H'FF	H'E800D00A	8	
Data control register_1		SPDCR_1	R/W	H'20	H'E800D00B	8	
Clock delay register_1		SPCKD_1	R/W	H'00	H'E800D00C	8	
Slave select negation delay register_1		SSLND_1	R/W	H'00	H'E800D00D	8	
Next-access delay register_1		SPND_1	R/W	H'00	H'E800D00E	8	
Command register0_1		SPCMD0_1	R/W	H'070D	H'E800D010	16	
Command register1_1		SPCMD1_1	R/W	H'070D	H'E800D012	16	
Command register2_1		SPCMD2_1	R/W	H'070D	H'E800D014	16	
Command register3_1		SPCMD3_1	R/W	H'070D	H'E800D016	16	
Buffer control register_1		SPBFCR_1	R/W	H'00	H'E800D020	8	
Buffer data count setting register_1		SPBFDR_1	R	H'0000	H'E800D022	16	

**Table 16.2 Register Configuration**

Channel	Register Name	Abbreviation*1	R/W	Initial Value	Address	Access Size	
2	Control register_2	SPCR_2	R/W	H'00	H'E800D800	8	
	Slave select polarity register_2	SSLP_2	R/W	H'00	H'E800D801	8	
	Pin control register_2	SPPCR_2	R/W	H'00	H'E800D802	8	
	Status register_2	SPSR_2	R/(W) *2	H'60	H'E800D803	8	
	Data register_2	SPDR_2	R/W	Undefined	H'E800D804	8, 16, 32	
	Sequence control register_2	SPSCR_2	R/W	H'00	H'E800D808	8	
	Sequence status register_2	SPSSR_2	R	H'00	H'E800D809	8	
	Bit rate register_2	SPBR_2	R/W	H'FF	H'E800D80A	8	
	Data control register_2	SPDCR_2	R/W	H'20	H'E800D80B	8	
	Clock delay register_2	SPCKD_2	R/W	H'00	H'E800D80C	8	
	Slave select negation delay register_2	SSLND_2	R/W	H'00	H'E800D80D	8	
	Next-access delay register_2	SPND_2	R/W	H'00	H'E800D80E	8	
	Command register0_2	SPCMD0_2	R/W	H'070D	H'E800D810	16	
	Command register1_2	SPCMD1_2	R/W	H'070D	H'E800D812	16	
	Command register2_2	SPCMD2_2	R/W	H'070D	H'E800D814	16	
	Command register3_2	SPCMD3_2	R/W	H'070D	H'E800D816	16	
	Buffer control register_2	SPBFCR_2	R/W	H'00	H'E800D820	8	
	Buffer data count setting register_2	SPBFDR_2	R	H'0000	H'E800D822	16	
	3	Control register_3	SPCR_3	R/W	H'00	H'E800E000	8
		Slave select polarity register_3	SSLP_3	R/W	H'00	H'E800E001	8
Pin control register_3		SPPCR_3	R/W	H'00	H'E800E002	8	
Status register_3		SPSR_3	R/(W)*2	H'60	H'E800E003	8	
Data register_3		SPDR_3	R/W	Undefined	H'E800E004	8, 16, 32	
Sequence control register_3		SPSCR_3	R/W	H'00	H'E800E008	8	
Sequence status register_3		SPSSR_3	R	H'00	H'E800E009	8	
Bit rate register_3		SPBR_3	R/W	H'FF	H'E800E00A	8	
Data control register_3		SPDCR_3	R/W	H'20	H'E800E00B	8	
Clock delay register_3		SPCKD_3	R/W	H'00	H'E800E00C	8	
Slave select negation delay register_3		SSLND_3	R/W	H'00	H'E800E00D	8	
Next-access delay register_3		SPND_3	R/W	H'00	H'E800E00E	8	
Command register0_3		SPCMD0_3	R/W	H'070D	H'E800E010	16	
Command register1_3		SPCMD1_3	R/W	H'070D	H'E800E012	16	
Command register2_3		SPCMD2_3	R/W	H'070D	H'E800E014	16	
Command register3_3		SPCMD3_3	R/W	H'070D	H'E800E016	16	
Buffer control register_3		SPBFCR_3	R/W	H'00	H'E800E020	8	
Buffer data count setting register_3		SPBFDR_3	R	H'0000	H'E800E022	16	

**Table 16.2 Register Configuration**

Channel	Register Name	Abbreviation*1	R/W	Initial Value	Address	Access Size
4	Control register_4	SPCR_4	R/W	H'00	H'E800E800	8
	Slave select polarity register_4	SSLP_4	R/W	H'00	H'E800E801	8
	Pin control register_4	SPPCR_4	R/W	H'00	H'E800E802	8
	Status register_4	SPSR_4	R/(W) *2	H'60	H'E800E803	8
	Data register_4	SPDR_4	R/W	Undefined	H'E800E804	8, 16, 32
	Sequence control register_4	SPSCR_4	R/W	H'00	H'E800E808	8
	Sequence status register_4	SPSSR_4	R	H'00	H'E800E809	8
	Bit rate register_4	SPBR_4	R/W	H'FF	H'E800E80A	8
	Data control register_4	SPDCR_4	R/W	H'20	H'E800E80B	8
	Clock delay register_4	SPCKD_4	R/W	H'00	H'E800E80C	8
	Slave select negation delay register_4	SSLND_4	R/W	H'00	H'E800E80D	8
	Next-access delay register_4	SPND_4	R/W	H'00	H'E800E80E	8
	Command register0_4	SPCMD0_4	R/W	H'070D	H'E800E810	16
	Command register1_4	SPCMD1_4	R/W	H'070D	H'E800E812	16
	Command register2_4	SPCMD2_4	R/W	H'070D	H'E800E814	16
	Command register3_4	SPCMD3_4	R/W	H'070D	H'E800E816	16
	Buffer control register_4	SPBFCR_4	R/W	H'00	H'E800E820	8
	Buffer data count setting register_4	SPBFDR_4	R	H'0000	H'E800E822	16

Note 1. In the description of the register names, the channel is omitted.

Note 2. Only 0 can be written to clear the flag.

### 16.3.1 Control Register (SPCR)

SPCR sets the operating mode. If the MSTR and MODFEN bits are changed while the function of this module is enabled by setting the SPE bit to 1, subsequent operations cannot be guaranteed.

Bit:	7	6	5	4	3	2	1	0
	SPRIE	SPE	SPTIE	SPEIE	MSTR	MOD FEN	—	—
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Function
7	SPRIE	0	R/W	Receive Interrupt Enable Enables or disables generation of receive interrupt requests (SPRI) when the number of receive data units in the receive buffer (SPRX) is equal to or greater than the specified receive buffer data triggering number and the SPRF flag in SPSR is set to 1. 0: Disables the generation of receive interrupt requests. 1: Enables the generation of receive interrupt requests.
6	SPE	0	R/W	Function Enable Setting this bit to 1 enables the module function. When the MODF bit in the status register (SPSR) is 1, the SPE bit cannot be set to 1 (see section 16.4.6, Error Detection). Setting the SPE bit to 0 disables the module function, and initializes a part of the module function (see section 16.4.7, Initialization). 0: Disables the module function. 1: Enables the module function.
5	SPTIE	0	R/W	Transmit Interrupt Enable Enables or disables generation of transmit interrupt requests (SPTI) when the number of transmit data units in the transmit buffer (SPTX) is equal to or less than the specified transmit buffer data triggering number and the SPTEF flag in SPSR is set to 1. 0: Disables the generation of transmit interrupt requests. 1: Enables the generation of transmit interrupt requests.
4	SPEIE	0	R/W	Error Interrupt Enable Enables or disables the generation of error interrupt requests when this module detects a mode fault error and sets the MODF bit in the status register (SPSR) to 1, or when this module detects an overrun error and sets the OVRF bit in SPSR to 1 (see section 16.4.6, Error Detection). 0: Disables the generation of error interrupt requests. 1: Enables the generation of error interrupt requests. Note: This bit is valid only in SPI slave mode.
3	MSTR	0	R/W	Master/Slave Mode Select Selects master/slave mode. According to MSTR bit settings, this module determines the direction of the RSPCK, MOSI, MISO, and SSL pins. 0: Slave mode 1: Master mode
2	MODFEN	0	R/W	Mode Fault Error Detection Enable Enables or disables the detection of a mode fault error (see section 16.4.6, Error Detection). 0: Disables the detection of a mode fault error. 1: Enables the detection of a mode fault error. Note: This bit is valid only in SPI slave mode. When master mode is specified with the MSTR bit, this bit should always be cleared to 0.
1, 0	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.

### 16.3.2 Slave Select Polarity Register (SSLP)

SSLP sets the polarity of the SSL signal. If the contents of SSL0P are changed while the function of this module is enabled by setting the SPE bit in the control register (SPCR) to 1, subsequent operations cannot be guaranteed.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SSL0P
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Function
7 to 1	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
0	SSL0P	0	R/W	SSL Signal Polarity Setting Sets the polarity of the SSL signal. The value of SSL0P indicates the active polarity of the SSL signal. 0: SSL signal 0-active 1: SSL signal 1-active

### 16.3.3 Pin Control Register (SPPCR)

SPPCR sets the modes of the pins. If the contents of this register are changed while the function of this module is enabled by setting the SPE bit in the control register (SPCR) to 1, subsequent operations cannot be guaranteed.

Bit:	7	6	5	4	3	2	1	0
	—	—	MOIFE	MOIFV	—	—	—	SPLP
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Function
7, 6	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
5	MOIFE	0	R/W	MOSI Idle Value Fixing Enable Fixes the MOSI output value when this module in master mode is in an SSL negation period (including the SSL retention period during a burst transfer). When MOIFE is 0, this module outputs the last output value from the previous serial transfer during the SSL negation period to the MOSI pin. (The value is undefined when CPHA is 0). When MOIFE is 1, this module outputs the fixed value set in the MOIFV bit to the MOSI pin. 0: MOSI output value equals the last output value from previous transfer. (The value is undefined when CPHA is 0). 1: MOSI output value equals the value set in the MOIFV bit.
4	MOIFV	0	R/W	MOSI Idle Fixed Value If the MOIFE bit is 1 in master mode, this module, according to MOIFV bit settings, determines the MOSI signal value during the SSL negation period (including the SSL retention period during a burst transfer). 0: MOSI Idle fixed value equals 0. 1: MOSI Idle fixed value equals 1.
3 to 1	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
0	SPLP	0	R/W	Loopback When the SPLP bit is set to 1, this module shuts off the path between the MISO pin and the shift register, and between the MOSI pin and the shift register, and connects (reverses) the input path and the output path for the shift register. 0: Normal mode 1: Loopback mode

### 16.3.4 Status Register (SPSR)

SPSR indicates the operating status.

Bit:	7	6	5	4	3	2	1	0
	SPRF	TEND	SPTEF	—	—	MODF	—	OVRF
Initial value:	0	1	1	0	0	0	0	0
R/W:	R	R	R	R	R	R/(W)*	R	R/(W)*

Note: \* Only 0 can be written to clear the flag after reading 1.

Bit	Bit Name	Initial Value	R/W	Function
7	SPRF	0	R	<p>Receive Buffer Full Flag</p> <p>Indicates that the number of receive data units in the receive buffer (SPRX) is equal to or greater than the receive buffer data triggering number specified in the buffer control register (SPBFCR).</p> <p>0: The number of receive data units in the receive buffer is less than the receive buffer data triggering number.</p> <p>1: The number of receive data units in the receive buffer is equal to or greater than the receive buffer data triggering number.</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> <li>The receive buffer data is read until the number of data units in the receive buffer becomes less than the specified receive buffer data triggering number.</li> <li>Receive buffer data reset is enabled.</li> <li>Power-on reset</li> </ul> <p>[Setting condition]</p> <ul style="list-style-type: none"> <li>The number of data units in the receive buffer is equal to or greater than the specified receive buffer data triggering number.</li> </ul>
6	TEND	1	R	<p>Transmit End</p> <p>This bit is set to 1 when transmission is completed, and this bit is 0 when transmission is not completed.</p> <p>[Clearing condition]</p> <ul style="list-style-type: none"> <li>When transmit data are transferred from the transmit register to the shift register.</li> </ul> <p>[Setting condition]</p> <ul style="list-style-type: none"> <li>When the number of data units in the transmit buffer (SPTX) is zero when a serial transfer is completed.</li> </ul> <p>Note: • This bit is valid only in SPI master mode.</p>
5	SPTEF	1	R	<p>Transmit Buffer Empty Flag</p> <p>Indicates that the number of transmit data units in the transmit buffer (SPTX) is equal to or less than the transmit buffer data triggering number specified in the buffer control register (SPBFCR).</p> <p>0: The number of transmit data units in the transmit buffer is equal to or greater than the specified transmit buffer data triggering number.</p> <p>1: The number of transmit data units in the transmit buffer is less than the specified transmit buffer data triggering number.</p> <p>[Clearing condition]</p> <ul style="list-style-type: none"> <li>When data is written to the transmit buffer until the number of transmit data units in the transmit buffer exceeds the specified transmit buffer data triggering number.</li> </ul> <p>[Setting conditions]</p> <ul style="list-style-type: none"> <li>When the number of transmit data units in the transmit buffer is less than the specified transmit buffer data triggering number.</li> <li>When transmit buffer data reset is enabled.</li> <li>Power-on reset</li> </ul>
4, 3	—	All 0	R	<p>Reserved</p> <p>The write value should always be 0. Otherwise, operation cannot be guaranteed.</p>



Bit	Bit Name	Initial Value	R/W	Function
2	MODF	0	R/(W)*1	<p>Mode Fault Error Flag</p> <p>Indicates the occurrence of a mode fault error. If the MODFEN bit is set to 1 when this module is in slave mode and the SSL pin is negated before the RSPCK cycle necessary for data transfer ends, this module detects a mode fault error. The active level of the SSL signal is determined by the SSL0P bit in the slave select polarity register (SSLP).</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> <li>• SPSR is read when the MODF bit is 1, and then 0 is written to the MODF bit.</li> <li>• Power-on reset</li> </ul> <p>0: No mode fault error occurred 1: A mode fault error occurred</p> <p>Note: This bit is valid only in SPI slave mode.</p>
1	—	0	R	<p>Reserved</p> <p>The write value should always be 0. Otherwise, operation cannot be guaranteed.</p>
0	OVRF	0	R/(W)*1	<p>Overrun Error Flag</p> <p>Indicates the occurrence of an overrun error. If a serial transfer ends when there is not enough space for receiving the specified length of data in the receive buffer (SPRX), this module detects an overrun error, and sets the OVRF bit to 1.</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> <li>• SPSR is read when the OVRF bit is 1, and then 0 is written to the OVRF bit.</li> <li>• Power-on reset</li> </ul> <p>0: No overrun error occurred 1: An overrun error occurred</p> <p>Note: This bit is valid only in SPI slave mode.</p>

Note 1. Only 0 can be written to clear the flag after reading 1.

### 16.3.5 Data Register (SPDR)

SPDR is a buffer that holds data for transmission and reception.

The transmit buffer (SPTX) and receive buffer (SPRX) are independent and are mapped to SPDR.

SPDR should be read or written to in byte, word, or longword units according to the access width specification bit (SPLW) in the data control register (SPDCR).

The bit length to be used is determined by the data length specification bits (SPB3 to SPB0) in the command register (SPCMD).

When data is written to SPDR, the data will be written to the transmit buffer from SPDR if the transmit buffer has a space equal to or more than the SPDR access width. If there is not enough space, data will not be written to the transmit buffer. Even if an attempt is made to write data to the buffer, the data is ignored.

When data is read from SPDR, receive data in the receive buffer will be read. If SPDR is read when there is no receive data in the receive buffer, the read value is undefined.

When SPDR is written to with the longword-, word-, or byte-access width, the transmit data should be written to address 0 irrespective of the access width. If data is written to the other addresses, the data is not guaranteed.

When SPDR is read with the longword-, word-, or byte-access width, the receive data should be read from address 0. If data is read from the other addresses, the data is not guaranteed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SPD31	SPD30	SPD29	SPD28	SPD27	SPD26	SPD25	SPD24	SPD23	SPD22	SPD21	SPD20	SPD19	SPD18	SPD17	SPD16
Initial value:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SPD15	SPD14	SPD13	SPD12	SPD11	SPD10	SPD9	SPD8	SPD7	SPD6	SPD5	SPD4	SPD3	SPD2	SPD1	SPD0
Initial value:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

### 16.3.6 Sequence Control Register (SPSCR)

SPSCR sets the sequence control method when this module operates in master mode. If the contents of SPSCR are changed while the MSTR and SPE bits in the control register (SPCR) are 1 with the function of this module enabled in master mode, the subsequent operation cannot be guaranteed.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SPS LN1	SPS LNO
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Function															
7 to 2	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.															
1	SPSLN1	0	R/W	Sequence Length Specification These bits specify a sequence length when this module in master mode performs sequential operations. This module in master mode changes command registers 0 to 3 (SPCMD0 to SPCMD3) to be referenced and the order in which they are referenced according to the sequence length that is set in the SPSLN1 and SPSLN0 bits. The relationship among the setting of bits SPSLN1 and SPSLN0, sequence length, and SPCMD0 to SPCMD3 referenced by this module is shown below. In slave mode, SPCMD0 is always referenced.															
0	SPSLN0	0	R/W																
				<table border="0"> <tr> <td></td> <td>Sequence Length</td> <td>Referenced SPCMD #</td> </tr> <tr> <td>00:</td> <td>1</td> <td>0 → 0 → ...</td> </tr> <tr> <td>01:</td> <td>2</td> <td>0 → 1 → 0 → ...</td> </tr> <tr> <td>10:</td> <td>3</td> <td>0 → 1 → 2 → 0 → ...</td> </tr> <tr> <td>11:</td> <td>4</td> <td>0 → 1 → 2 → 3 → 0 → ...</td> </tr> </table>		Sequence Length	Referenced SPCMD #	00:	1	0 → 0 → ...	01:	2	0 → 1 → 0 → ...	10:	3	0 → 1 → 2 → 0 → ...	11:	4	0 → 1 → 2 → 3 → 0 → ...
	Sequence Length	Referenced SPCMD #																	
00:	1	0 → 0 → ...																	
01:	2	0 → 1 → 0 → ...																	
10:	3	0 → 1 → 2 → 0 → ...																	
11:	4	0 → 1 → 2 → 3 → 0 → ...																	

### 16.3.7 Sequence Status Register (SPSSR)

SPSSR indicates the sequence control status when this module operates in master mode.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SPCP1	SPCP0
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Function
7 to 2	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
1	SPCP1	0	R	Command Pointer
0	SPCP0	0	R	During sequence control, these bits indicate one of the command registers 0 to 3 (SPCMD0 to SPCMD3) that is currently pointed to by the pointer. The relationship between the setting of SPCP1 and SPCP0 and SPCMD0 to SPCMD3 is shown below. For the sequence control, see section 16.4.8 (1) (c), Sequence Control. 00: SPCMD0 01: SPCMD1 10: SPCMD2 11: SPCMD3

### 16.3.8 Bit Rate Register (SPBR)

SPBR sets the bit rate in master mode. If the contents of SPBR are changed while the MSTR and SPE bits in the control register (SPCR) are 1 with the function of this module enabled in master mode, the subsequent operation cannot be guaranteed.

Bit:	7	6	5	4	3	2	1	0
	SPR7	SPR6	SPR5	SPR4	SPR3	SPR2	SPR1	SPR0
Initial value:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

When this module is used in slave mode, the bit rate depends on the bit rate of the input clock regardless of the settings of SPBR and BRDV.

The bit rate is determined by combinations of SPBR settings and the bit settings in the BRDV1 and BRDV0 bits in the command registers (SPCMD0 to SPCMD3). The equation for calculating the bit rate is given below. In the equation, n denotes an SPBR setting (0, 1, 2, ..., 255), and N denotes a BRDV1 and BRDV0 bit setting (0, 1, 2, 3).

$$\text{Bit rate} = \frac{f(P1\phi)}{2 \times (n + 1) \times 2^N}$$

Table 16.3 shows examples of the relationship between the SPBR register and BRDV1 and BRDV0 bit settings.

**Table 16.3 Relationship between SPBR and BRDV1 and BRDV0 Settings**

SPBR (n)	BRDV[1:0] (N)	Division Ratio	Bit Rate		
			P1 $\phi$ = 50 MHz	P1 $\phi$ = 64 MHz	P1 $\phi$ = 66.67 MHz
0	0	2	25.00 Mbps	32.00 Mbps	33.33 Mbps
1	0	4	12.50 Mbps	16.00 Mbps	16.67 Mbps
2	0	6	8.33 Mbps	10.67 Mbps	11.11 Mbps
3	0	8	6.25 Mbps	8.00 Mbps	8.33 Mbps
4	0	10	5.00 Mbps	6.40 Mbps	6.67 Mbps
5	0	12	4.17 Mbps	5.33 Mbps	5.56 Mbps
5	1	24	2.08 Mbps	2.67 Mbps	2.78 Mbps
5	2	48	1.04 Mbps	1.33 Mbps	1.39 Mbps
5	3	96	520.83 Kbps	666.67 Kbps	694.44 Kbps
255	3	4096	12.21 Kbps	15.63 Kbps	16.28 Kbps

### 16.3.9 Data Control Register (SPDCR)

SPDCR selects the width to access SPDR from longword-, word-, and byte-width, and enables or disables dummy data transmission for the master mode operation.

If the contents of SPDCR are changed while bit TEND in the status register (SPSR) indicates that transmission is not completed, the subsequent operation cannot be guaranteed.

Bit:	7	6	5	4	3	2	1	0
	TXDMY	SPLW1	SPLW0	—	—	—	—	—
Initial value:	0	0	1	0	0	0	0	0
R/W:	R/W	R/W	R/W	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Function
7	TXDMY	0	R/W	Dummy Data Transmission Enable Enables or disables dummy data transmission. When communication is performed with this bit set to 1, dummy data is transmitted from the MOSI pin and a serial communication can be performed even if there is no transmit data in the transmit buffer. Specifically, if there is no transmit data in the transmit buffer and this bit is set to 1, dummy data is transferred to the shift register. Data previously transmitted from the pin is used as dummy data. If this bit is set to 1 after the initialization and a transfer is performed, the transmitted dummy data is undefined. 0: Disables dummy data transmission. 1: Enables dummy data transmission. Note: This bit is valid only in the master mode.
6	SPLW1	0	R/W	Access Width Specification
5	SPLW0	1	R/W	Specifies the width for accessing the data register (SPDR). If the length of data transferred to SPDR does not agree with these bit settings, operation is not guaranteed. 00: Setting prohibited 01: SPDR is accessed in bytes (8 bits). 10: SPDR is accessed in words (16 bits). 11: SPDR is accessed in longwords (32 bits).
4 to 0	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.

### 16.3.10 Clock Delay Register (SPCKD)

SPCKD sets a period from the beginning of SSL signal assertion to RSPCK oscillation (RSPCK delay) when the SCKDEN bit in the command register (SPCMD) is 1. If the contents of SPCKD are changed while the MSTR and SPE bits in the control register (SPCR) are 1 with the function of this module enabled in master mode, the subsequent operation cannot be guaranteed.

When using this module in slave mode, set B'000 to SCKDL2 to SCKDL0.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	SCK DL2	SCK DL1	SCK DL0
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Function
7 to 3	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
2	SCKDL2	0	R/W	RSPCK Delay Setting
1	SCKDL1	0	R/W	These bits set an RSPCK delay value when the SCKDEN bit in SPCMD is 1. The relationship between the setting of SCKDL2 to SCKDL0 and the RSPCK delay value is shown below. 000: 1 RSPCK 001: 2 RSPCK 010: 3 RSPCK 011: 4 RSPCK 100: 5 RSPCK 101: 6 RSPCK 110: 7 RSPCK 111: 8 RSPCK
0	SCKDL0	0	R/W	

### 16.3.11 Slave Select Negation Delay Register (SSLND)

SSLND sets a period (SSL negation delay) from the transmission of a final RSPCK edge to the negation of the SSL signal during a serial transfer by this module in master mode. If the contents of SSLND are changed while the MSTR and SPE bits in the control register (SPCR) are 1 with the function of this module enabled in master mode, the subsequent operation cannot be guaranteed.

When using this module in slave mode, set B'000 to SLNDL2 to SLNDL0.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	SLN DL2	SLN DL1	SLN DL0
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Function
7 to 3	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
2	SLNDL2	0	R/W	SSL Negation Delay Setting
1	SLNDL1	0	R/W	These bits set an SSL negation delay when the SLNDEN bit in SPCMD is 1.
0	SLNDL0	0	R/W	The relationship between the setting of SLNDL2 to SLNDL0 and the SSL negation delay value is shown below. 000: 1 RSPCK 001: 2 RSPCK 010: 3 RSPCK 011: 4 RSPCK 100: 5 RSPCK 101: 6 RSPCK 110: 7 RSPCK 111: 8 RSPCK

### 16.3.12 Next-Access Delay Register (SPND)

SPND sets a non-active period (next-access delay) after termination of a serial transfer when the SPNDEN bit in the command register (SPCMD) is 1. If the contents of SPND are changed while the MSTR and SPE bits in the control register (SPCR) are 1 with the function of this module enabled in master mode, the subsequent operation cannot be guaranteed.

When using this module in slave mode, set B'000 to SPNDL2 to SPNDL0.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	SPN DL2	SPN DL1	SPN DL0
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Function
7 to 3	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
2	SPNDL2	0	R/W	Next-Access Delay Setting
1	SPNDL1	0	R/W	These bits set a next-access delay when the SPNDEN bit in SPCMD is 1.
0	SPNDL0	0	R/W	The relationship between the setting of SPNDL2 to SPNDL0 and the next-access delay value is shown below. 000: 1 RSPCK + 2 P1 $\phi$ 001: 2 RSPCK + 2 P1 $\phi$ 010: 3 RSPCK + 2 P1 $\phi$ 011: 4 RSPCK + 2 P1 $\phi$ 100: 5 RSPCK + 2 P1 $\phi$ 101: 6 RSPCK + 2 P1 $\phi$ 110: 7 RSPCK + 2 P1 $\phi$ 111: 8 RSPCK + 2 P1 $\phi$

### 16.3.13 Command Register (SPCMD)

Each channel has four command registers (SPCMD0 to SPCMD3). SPCMD0 to SPCMD3 are used to set a transfer format for master mode operation. Some of the bits in SPCMD0 are used to set a transfer mode for slave mode operation. In master mode, this module sequentially references SPCMD0 to SPCMD3 according to the settings in bits SPSLN1 and SPSLN0 in the sequence control register (SPSCR), and executes the serial transfer that is set in the referenced SPCMD. While bit TEND in the status register (SPSR) indicates that transmission is not completed, correct operation of this module cannot be guaranteed if SPCMD is changed that is referred by this module. SPCMD referenced by this module in master mode can be checked by means of bits SPCP1 and SPCP0 in the sequence status register (SPSSR). When the function of this module in slave mode is enabled, operation cannot be guaranteed if the value set in SPCMD0 is changed.

Bit:	15	14	13	12	11	10	9	8
	SCK DEN	SLN DEN	SPN DEN	LSBF	SPB3	SPB2	SPB1	SPB0
Initial value:	0	0	0	0	0	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	7	6	5	4	3	2	1	0
	SSLKP	—	—	—	BRDV1	BRDV0	CPOL	CPHA
Initial value:	0	0	0	0	1	1	0	1
R/W:	R/W	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Function
15	SCKDEN	0	R/W	<b>RSPCK Delay Setting Enable</b> Sets the period from the point this module in master mode activates the SSL signal until the RSPCK starts oscillation (RSPCK delay). If the SCKDEN bit is 0, this module sets the RSPCK delay to 1 RSPCK. If the SCKDEN bit is 1, this module starts the oscillation of RSPCK at an RSPCK delay in compliance with the clock delay register (SPCKD) settings. To use this module in slave mode, the SCKDEN bit should be set to 0. 0: An RSPCK delay of 1 RSPCK 1: An RSPCK delay equal to SPCKD settings.
14	SLNDEN	0	R/W	<b>SSL Negation Delay Setting Enable</b> Sets the period from the point this module in master mode stops RSPCK oscillation until this module sets the SSL signal inactive (SSL negation delay). If the SLNDEN bit is 0, this module sets the SSL negation delay to 1 RSPCK. If the SLNDEN bit is 1, this module negates the SSL signal at an SSL negation delay in compliance with the slave select negation delay register (SSLND) settings. To use this module in slave mode, the SLNDEN bit should be set to 0. 0: An SSL negation delay of 1 RSPCK 1: An SSL negation delay equal to SSLND settings.
13	SPNDEN	0	R/W	<b>Next-Access Delay Enable</b> Sets the period from the point this module in master mode terminates a serial transfer and sets the SSL signal inactive until this module enables the SSL signal assertion for the next access (next-access delay). If the SPNDEN bit is 0, this module sets the next-access delay to 1 RSPCK + 2P1 $\phi$ . If the SPNDEN bit is 1, this module inserts a next-access delay in compliance with the next-access delay register (SPND) settings. To use this module in slave mode, the SPNDEN bit should be set to 0. 0: A next-access delay of 1 RSPCK + 2 P1 $\phi$ 1: A next-access delay equal to SPND settings.
12	LSBF	0	R/W	<b>LSB First</b> Sets the data format in master mode or slave mode to MSB first or LSB first. 0: MSB first 1: LSB first



Bit	Bit Name	Initial Value	R/W	Function
11	SPB3	0	R/W	Data Length Setting
10	SPB2	1	R/W	These bits set a transfer data length in master mode or slave mode. 0100 to 0111: 8 bits 1111: 16 bits 0010, 0011: 32 bits Others: Setting prohibited
9	SPB1	1	R/W	
8	SPB0	1	R/W	
7	SSLKP	0	R/W	
6 to 4	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
3	BRDV1	1	R/W	Bit Rate Division Setting These bits are used to determine the bit rate. A bit rate is determined by combinations of bits BRDV1 and BRDV0 and the settings in the bit rate register (SPBR) (see section 16.3.8, Bit Rate Register (SPBR)). The settings in SPBR determine the base bit rate. The settings in bits BRDV1 and BRDV0 are used to select a bit rate which is obtained by dividing the base bit rate by 1, 2, 4, or 8. In the bits SPCMD0 to SPCMD3, different BRDV1 and BRDV0 settings can be specified. This permits the execution of serial transfers at a different bit rate for each command. 00: Select the base bit rate. 01: Select the base bit rate divided by 2. 10: Select the base bit rate divided by 4. 11: Select the base bit rate divided by 8.
2	BRDV0	1	R/W	
1	CPOL	0	R/W	RSPCK Polarity Setting Sets an RSPCK polarity in master or slave mode. When data communication is performed between the Renesas serial peripheral interface module and the other modules, the same RSPCK polarity should be set for both modules. 0: RSPCK = 0 when idle 1: RSPCK = 1 when idle
0	CPHA	1	R/W	RSPCK Phase Setting Sets an RSPCK phase in master or slave mode. When data communication is performed between the Renesas serial peripheral interface module and the other modules, the same RSPCK phase should be set for both modules. 0: Data sampling on odd edge, data variation on even edge 1: Data variation on odd edge, data sampling on even edge

### 16.3.14 Buffer Control Register (SPBFCR)

SPBFCR resets the number of data units in the transmit buffer (SPTX) or receive buffer (SPRX) and sets the number of triggering data units.

Bit:	7	6	5	4	3	2	1	0
	TXRST	RXRST	TXTRG[1:0]	—	—	RXTRG[2:0]		
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Function
7	TXRST	0	R/W	<b>Transmit Buffer Data Reset</b> Resets the transmit buffer to an empty state. Transmit data in the transmit buffer becomes invalid when this bit is set to 1. 0: Disables the reset operation*. 1: Enables the reset operation Note: The reset operation is performed after a power-on reset.
6	RXRST	0	R/W	<b>Receive Buffer Data Reset</b> Resets the receive buffer to an empty state. Receive data in the receive buffer becomes invalid when this bit is set to 1. 0: Disables the reset operation*. 1: Enables the reset operation Note: The reset operation is performed after a power-on reset.
5, 4	TXTRG[1:0]	00	R/W	<b>Transmit Buffer Data Triggering Number</b> Specifies the timing at which the transmit buffer empty state is determined, that is when the SPTEF flag in the status register is set. When the number of bytes of data in the transmit buffer (SPTX) is equal to or less than the specified triggering number, the SPTEF flag is set to 1. 00: 7 bytes (1)* 01: 6 bytes (2)* 10: 4 bytes (4)* 11: 0 bytes (8)* Note: The value in the parenthesis shows the number of available bytes in the transmit buffer (SPTX).
3	—	0	R	<b>Reserved</b> The write value should always be 0. Otherwise, operation cannot be guaranteed.
2 to 0	RXTRG[2:0]	000	R/W	<b>Receive Buffer Data Triggering Number</b> Specifies the timing at which the receive buffer full state is determined, that is when the SPRF flag in the status register is set. When the number of bytes of data in the receive buffer (SPRX) is equal to or greater than the specified triggering number, the SPRF flag is set to 1. 000: 1 byte (31)* 001: 2 bytes (30)* 010: 4 bytes (28)* 011: 8 bytes (24)* 100: 16 bytes (16)* 101: 24 bytes (8)* 110: 32 bytes (0)* 111: 5 bytes (27)* Note: * The value in the parenthesis shows the number of available bytes in the receive buffer (SPRX).

### 16.3.15 Buffer Data Count Setting Register (SPBFDR)

SPBFDR indicates the number of data units stored in the transmit buffer (SPTX) and receive buffer (SPRX). The upper eight bits indicate the number of transmit data units in SPTX and the lower eight bits indicate the number of receive data units in SPRX.

Bit:	15	14	13	12	11	10	9	8
	—	—	—	—	T[3:0]			
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
Bit:	7	6	5	4	3	2	1	0
	—	—	R[5:0]					
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Function
15 to 12	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
11 to 8	T[3:0]	0000	R	Indicates the number of bytes of data to be transmitted in SPTX. B'0000 indicates that SPTX is empty. B'1000 indicates that SPTX is full.
7, 6	—	All 0	R	Reserved The write value should always be 0. Otherwise, operation cannot be guaranteed.
5 to 0	R[5:0]	000000	R	Shows the number of bytes of received data in SPTX. B'000000 indicates that SPRX is empty. B'100000 indicates that SPRX is full.

## 16.4 Operation

In this section, the serial transfer period means a period from the beginning of driving valid data to the fetching of the final valid data.

### 16.4.1 Overview of Operations

This module is capable of serial transfers in slave mode and master mode. A particular mode of this module can be selected by using the MSTR bit in the control register (SPCR). Table 16.4 gives the relationship between the modes and SPCR settings, and a description of each mode.

**Table 16.4 Relationship between Modes and SPCR and Description of Each Mode**

Mode	Slave (SPI Operation)	Master (SPI Operation)
MSTR bit setting	0	1
MODFEN bit setting	0 or 1	0
RSPCK signal	Input	Output
MOSI signal	Input	Output
MISO signal	Output/Hi-Z	Input
SSL signal	Input	Output
SSL polarity modification function	Supported	Supported
Transfer rate	Up to $P1\phi/8$	Up to $P1\phi/2$
Clock source	RSPCK input	On-chip baud rate generator
Clock polarity	Two	Two
Clock phase	Two	Two
First transfer bit	MSB/LSB	MSB/LSB
Transfer data length	8, 16, or 32 bits	8, 16, or 32 bits
Burst transfer	Possible (CPHA = 1)	Possible (CPHA = 0,1)
RSPCK delay control	Not supported	Supported
SSL negation delay control	Not supported	Supported
Next-access delay control	Not supported	Supported
Transfer activation method	SSL input active or RSPCK oscillation	Transmit buffer is written when SPE = 1
Sequence control	Not supported	Supported
Transmit buffer empty detection	Supported	Supported
Receive buffer full detection	Supported	Supported
Overrun error detection	Supported	Not Supported
Mode fault error detection	Supported (MODFEN = 1)	Not supported

## 16.4.2 Pin Control

According to the MSTR bit in the control register (SPCR), this module can automatically switch pin directions and output modes. Table 16.5 shows the relationship between pin states and bit settings.

**Table 16.5 Relationship between Pin States and Bit Settings**

Mode	Pin	Pin State*1
Master mode (SPI operation) (MSTR = 1)	RSPCK	CMOS output
	SSL	CMOS output
	MOSI	CMOS output
	MISO	Input
Slave mode (SPI operation) (MSTR = 0)	RSPCK	Input
	SSL	Input
	MOSI	Input
	MISO*1	CMOS output/Hi-Z

Note 1. When SSL is at the non-active level or the SPE bit in SPCR is cleared to 0, the pin state is Hi-Z.

This module in master mode (SPI operation) determines MOSI signal values during the SSL negation period (including the SSL retention period during a burst transfer) according to MOIFE and MOIFV bit settings in SPPCR, as shown in Table 16.6.

**Table 16.6 MOSI Signal Value Determination during SSL Negation Period**

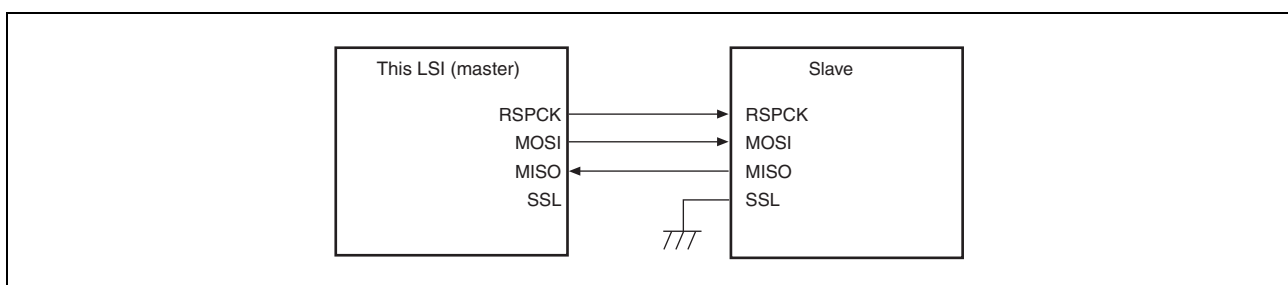
MOIFE	MOIFV	MOSI Signal Value during SSL Negation Period
0	0, 1	Last output value from previous transfer (The value is undefined when CPHA is 0)
1	0	Always 0
1	1	Always 1

## 16.4.3 System Configuration Example

### (1) Master/Slave (with This LSI Acting as Master)

Figure 16.2 shows a master/slave system configuration example when this LSI is used as a master. In master/slave configuration, the SSL output of this LSI (master) is not used. The SSL input of the slave is fixed to the low level, and the slave is always maintained in a selected state. In the transfer format corresponding to the case where the CPHA bit in the control register (SPCR) is 0, there are slave devices for which the SSL signal cannot be fixed to the active level. In situations where the SSL signal cannot be fixed, the SSL output of this LSI should be connected to the SSL input of the slave device.

This LSI (master) always drives the RSPCK and MOSI. The slave always drives the MISO.



**Figure 16.2 Master/Slave Configuration Example (This LSI = Master)**

## (2) Master/Slave (with This LSI Acting as Slave)

Figure 16.3 shows a master/slave system configuration example when this LSI is used as a slave. When this LSI is to operate as a slave, the SSL pin is used as SSL input. The master always drives the RSPCK and MOSI. This LSI (slave) always drives the MISO. When SSL is at the non-active level, the pin state is Hi-Z.

In the slave configuration in which the CPHA bit in the command register (SPCMD) is set to 1, the SSL input of this LSI (slave) is fixed to the 0 level, this LSI (slave) is always maintained in a selected state, and in this manner it is possible to execute serial transfer (Figure 16.4).

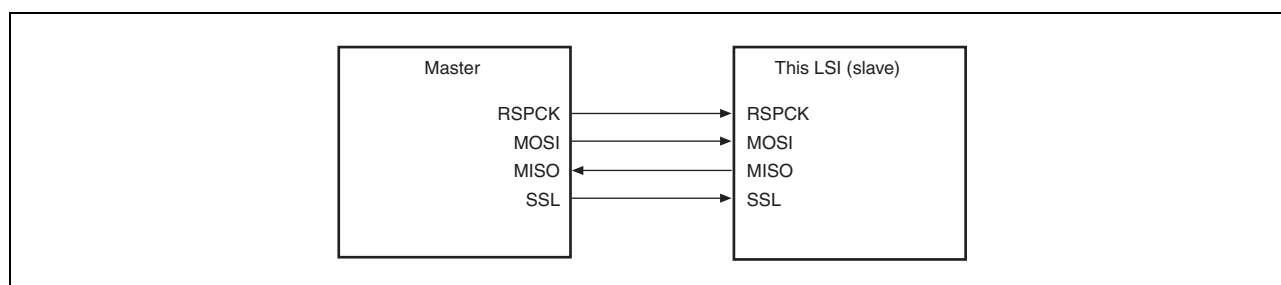


Figure 16.3 Master/Slave Configuration Example (This LSI = Slave)

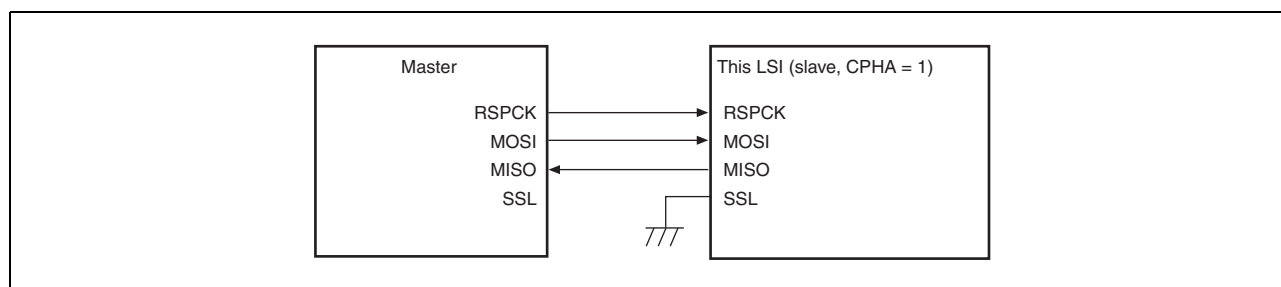


Figure 16.4 Master/Slave Configuration Example (This LSI = Slave, CPHA = 1)

### (3) Master/Multi-Slave (with This LSI Acting as Slave)

Figure 16.5 shows a master/multi-slave system configuration example when this LSI is used as a slave. In the example of Figure 16.5, the system is comprised of a master and two LSIs (slave X and slave Y).

The RSPCK and MOSI outputs of the master are connected to the RSPCK and MOSI inputs of the LSIs (slave X and slave Y). The MISO outputs of the LSIs (slave X and slave Y) are all connected to the MISO input of the master. SSLX and SSLY outputs of the master are connected to the SSL inputs of the LSIs (slave X and slave Y), respectively. The master always drives RSPCK, MOSI, SSLX, and SSLY. Of the LSIs (slave X and slave Y), the slave that receives low level input into the SSL0 input drives MISO.

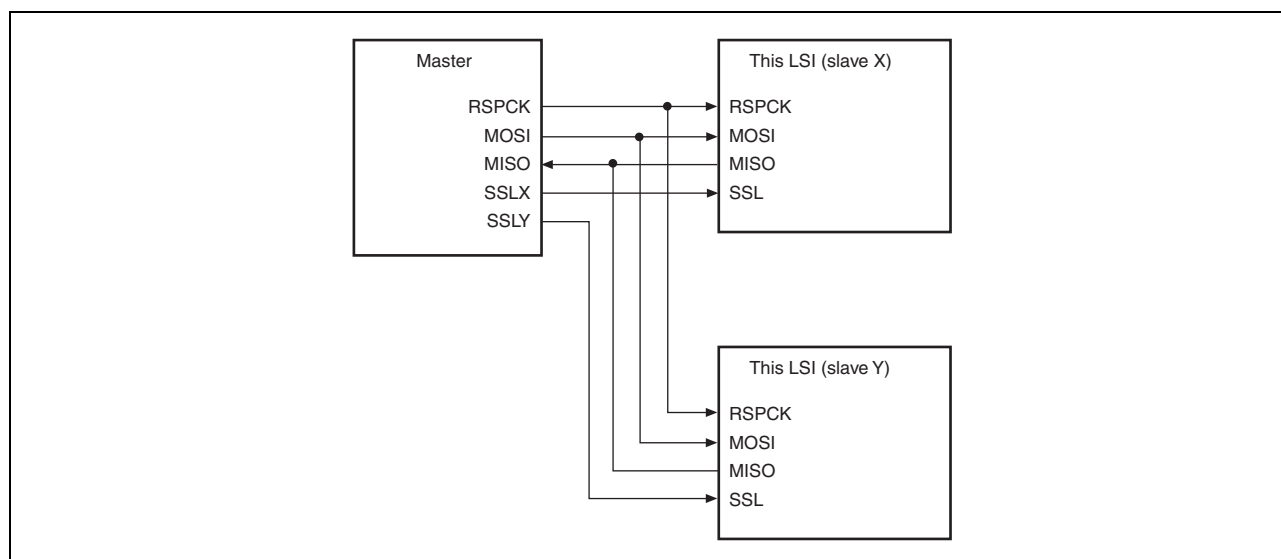


Figure 16.5 Master/Multi-Slave Configuration Example (This LSI = Slave)

### 16.4.4 Transfer Format

#### (1) CPHA = 0

Figure 16.6 shows a sample transfer format for the serial transfer of 8-bit data when the CPHA bit in the command register (SPCMD) is 0. In Figure 16.6, RSPCK (CPOL = 0) indicates the RSPCK signal waveform when the CPOL bit in SPCMD is 0; RSPCK (CPOL = 1) indicates the RSPCK signal waveform when the CPOL bit is 1. The sampling timing represents the timing at which this module fetches serial transfer data into the shift register. The input/output directions of the signals depend on the settings of this module. For details, see section 16.4.2, Pin Control.

When the CPHA bit is 0, the driving of valid data to the MOSI and MISO signals commences at an SSL signal assertion timing. The first RSPCK signal change timing that occurs after the SSL signal assertion becomes the first transfer data fetching timing. After this timing, data is sampled at every 1 RSPCK cycle. The change timing for MOSI and MISO signals is always 1/2 RSPCK cycle after the transfer data fetch timing. The settings in the CPOL bit do not affect the RSPCK signal operation timing; they only affect the signal polarity.

t1 denotes a period from an SSL signal assertion to RSPCK oscillation (RSPCK delay). t2 denotes a period from the cessation of RSPCK oscillation to an SSL signal negation (SSL negation delay). t3 denotes a period in which SSL signal assertion is suppressed for the next transfer after the end of serial transfer (next-access delay). t1, t2, and t3 are controlled by a master device running on the system. For a description of t1, t2, and t3 when this module is in master mode, see section 16.4.3 (1), Master/Slave (with This LSI Acting as Master).

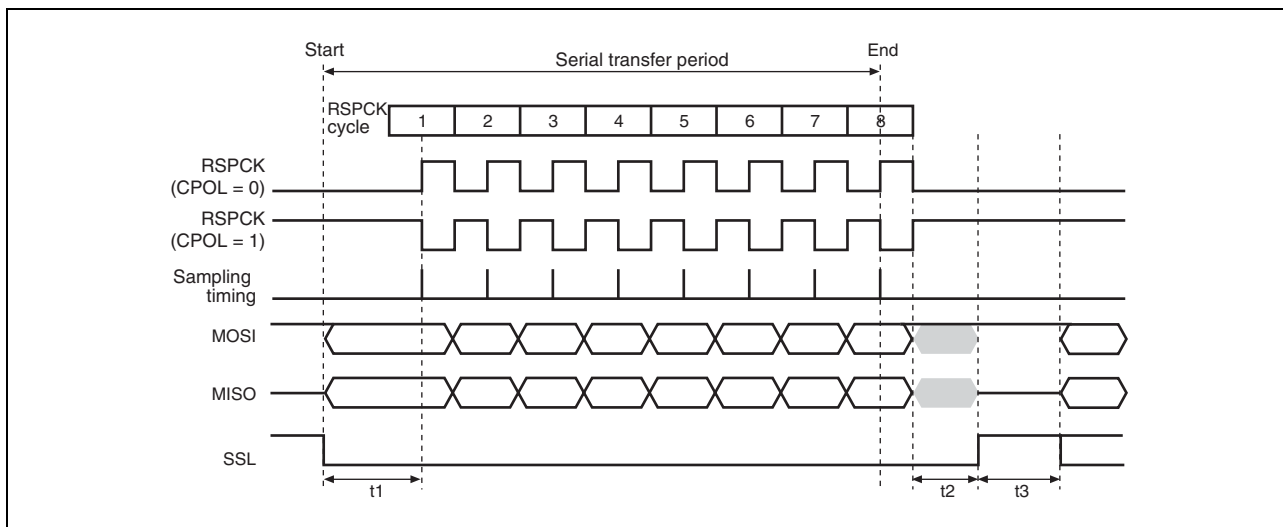


Figure 16.6 Transfer Format (CPHA = 0)



(2) CPHA = 1

Figure 16.7 shows a sample transfer format for the serial transfer of 8-bit data when the CPHA bit in the command register (SPCMD) is 1. In Figure 16.7, RSPCK (CPOL = 0) indicates the RSPCK signal waveform when the CPOL bit in SPCMD is 0; RSPCK (CPOL = 1) indicates the RSPCK signal waveform when the CPOL bit is 1. The sampling timing represents the timing at which this module fetches serial transfer data into the shift register. The input/output directions of the signals depend on the modes (master or slave). For details, see section 16.4.2, Pin Control. When the CPHA bit is 1, the driving of invalid data to the MOSI and MISO signals commences at an SSL signal assertion timing. The driving of valid data to the MOSI and MISO signals commences at the first RSPCK signal change timing that occurs after the SSL signal assertion. After this timing, data is updated at every 1 RSPCK cycle. The transfer data fetch timing is always 1/2 RSPCK cycle after the data update timing. The settings in the CPOL bit do not affect the RSPCK signal operation timing; they only affect the signal polarity.

t1, t2, and t3 are the same as those in the case of CPHA = 0. For a description of t1, t2, and t3 when this module is in master mode, see section 16.4.3 (1), Master/Slave (with This LSI Acting as Master).

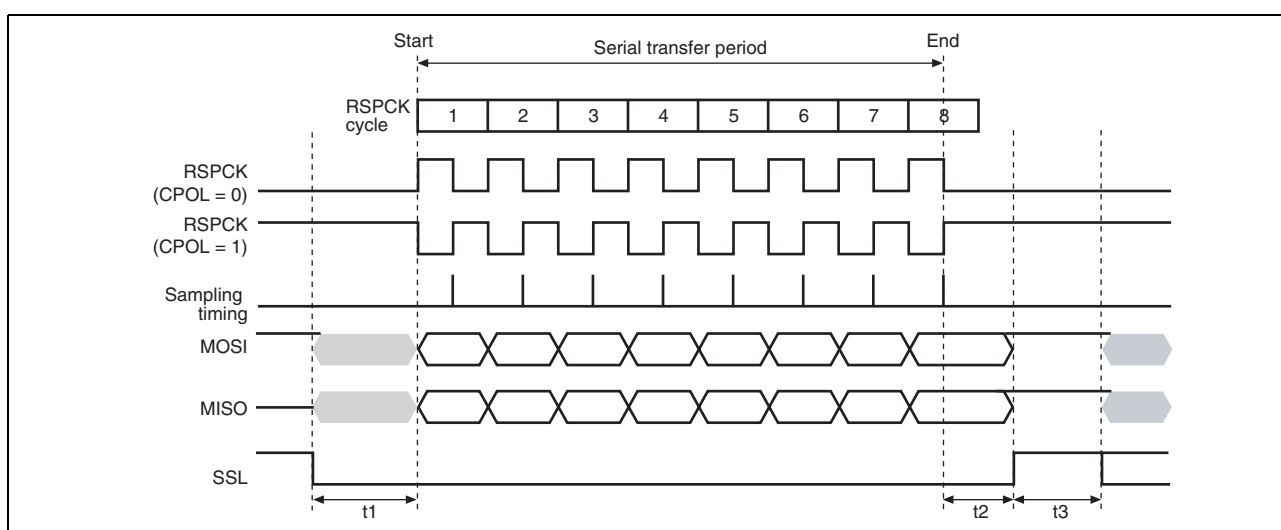


Figure 16.7 Transfer Format (CPHA = 1)

### 16.4.5 Data Format

The data format depends on the settings in the command register (SPCMD). Irrespective of MSB/LSB first, this module treats the range from the LSB of the data register (SPDR) to the assigned data length as transfer data.

#### (1) MSB First Transfer (32-Bit Data)

Figure 16.8 shows the operation of the transmit buffer (SPTX) and the shift register when this module performs a 32-bit data length MSB-first data transfer.

The CPU or direct memory access controller writes T31 to T00 to the transmit buffer of SPDR. If the shift register is empty, this module copies the data in the transmit buffer to the shift register, and fully populates the shift register. When serial transfer starts, this module outputs data from the MSB (bit 31) in the shift register, and shifts in the data from the LSB (bit 0) in the shift register. When the RSPCK cycle required for the serial transfer of 32 bits has passed, data R31 to R00 is stored in the shift register. In this state, this module copies the data from the shift register to the receive buffer, and empties the shift register. If the receive buffer does not have a space for the receive data length after the receive data has been copied from the shift register to the receive buffer, another serial transfer will not be started. In order to start another serial transfer, data for the receive data length should be read from the receive buffer to secure the necessary space in the receive buffer.

If another serial transfer is started before the CPU or direct memory access controller writes to the transmit buffer, received data R31 to R00 is shifted out from the shift register.

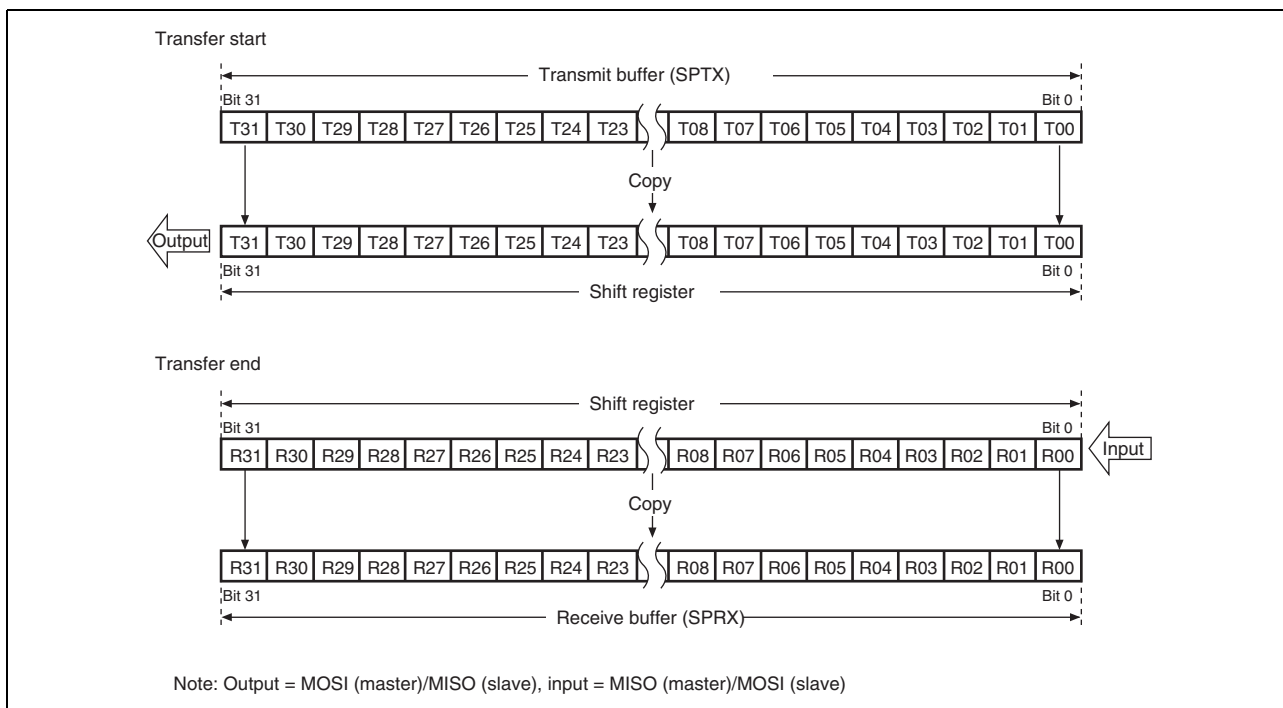


Figure 16.8 MSB First Transfer (32-Bit Data)

(2) MSB First Transfer (16-Bit Data)

Figure 16.9 shows the operation of the transmit buffer (SPTX) and the shift register when this module performs a 16-bit data length MSB-first data transfer.

The CPU or direct memory access controller writes T15 to T00 to the transmit buffer. If the shift register is empty, this module copies the data in the transmit buffer to the shift register, and fully populates the shift register. When serial transfer starts, this module outputs data from bit 15 in the shift register, and shifts in the data from the LSB (bit 0) in the shift register. When the RSPCK cycle required for the serial transfer of 16 bits has passed, received data R15 to R00 is stored in bits 15 to 0 in the shift register. After completion of the serial transfer, data that existed before the transfer is retained in bits 31 to 16 in the shift register. In this state, this module copies the data from the shift register to the receive buffer, and empties the shift register. If the receive buffer does not have a space for the receive data length after receive data has been copied from the shift register to the receive buffer, another serial transfer will not be started. In order to start another serial transfer, data for the receive data length should be read from the receive buffer to secure the necessary space in the receive buffer.

If another serial transfer is started before the CPU or direct memory access controller writes to the transmit buffer, received data R15 to R00 is shifted out from the shift register.

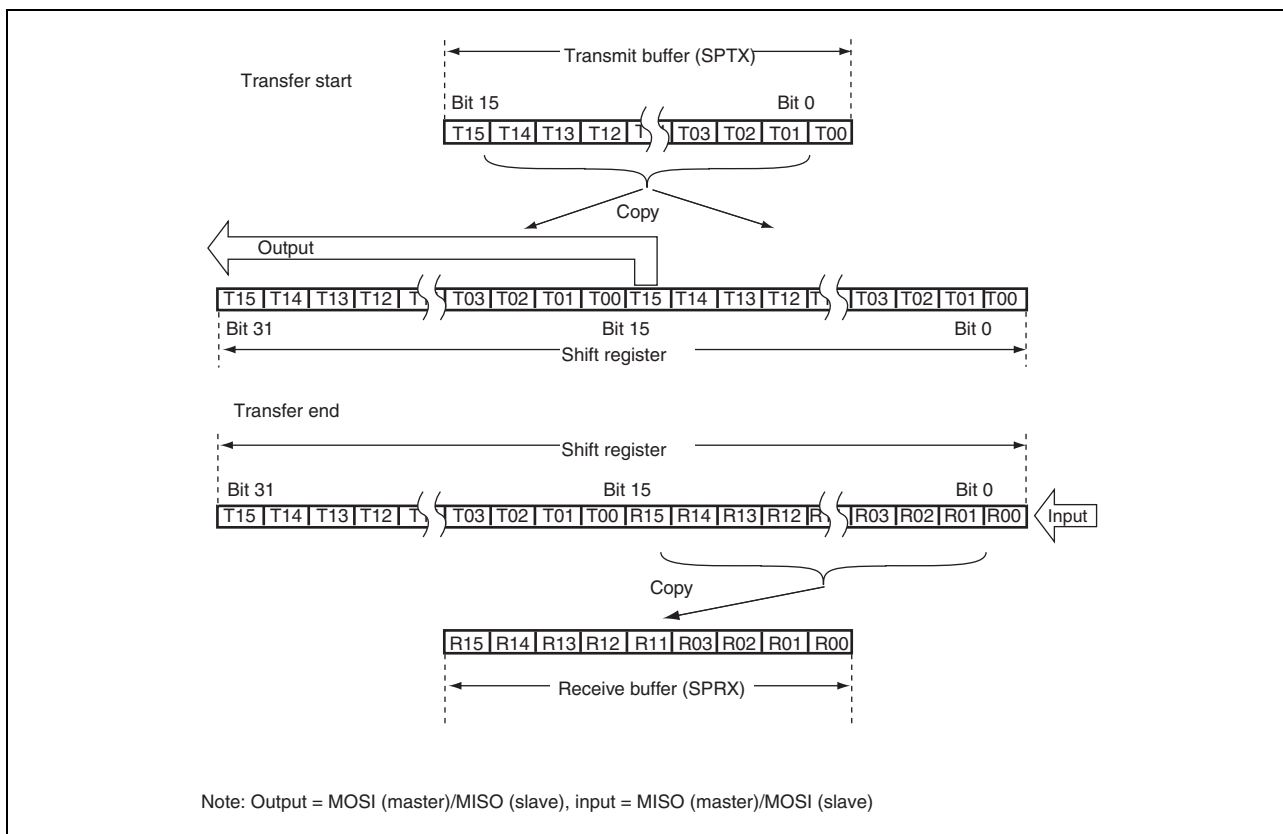


Figure 16.9 MSB First Transfer (16-Bit Data)

(3) MSB First Transfer (8-Bit Data)

Figure 16.10 shows the operation of the transmit buffer (SPTX) and the shift register when this module performs an 8-bit data length MSB-first data transfer.

The CPU or direct memory access controller writes T07 to T00 to the transmit buffer. If the shift register is empty, this module copies the data in the transmit buffer to the shift register, and fully populates the shift register. When serial transfer starts, this module outputs data from bit 7 in the shift register, and shifts in the data from the LSB (bit 0) in the shift register. When the RSPCK cycle required for the serial transfer of 8 bits has passed, received data R07 to R00 is stored in bits 7 to 0 in the shift register. After completion of the serial transfer, data that existed before the transfer is retained in bits 31 to 8 in the shift register. In this state, this module copies the data from the shift register to the receive buffer, and empties the shift register. If the receive buffer does not have a space for the receive data length after receive data has been copied from the shift register to the receive buffer, another serial transfer will not be started. In order to start another serial transfer, data for the receive data length should be read from the receive buffer to secure the necessary area in the receive buffer.

If another serial transfer is started before the CPU or direct memory access controller writes to the transmit buffer, received data R07 to R00 is shifted out from the shift register.

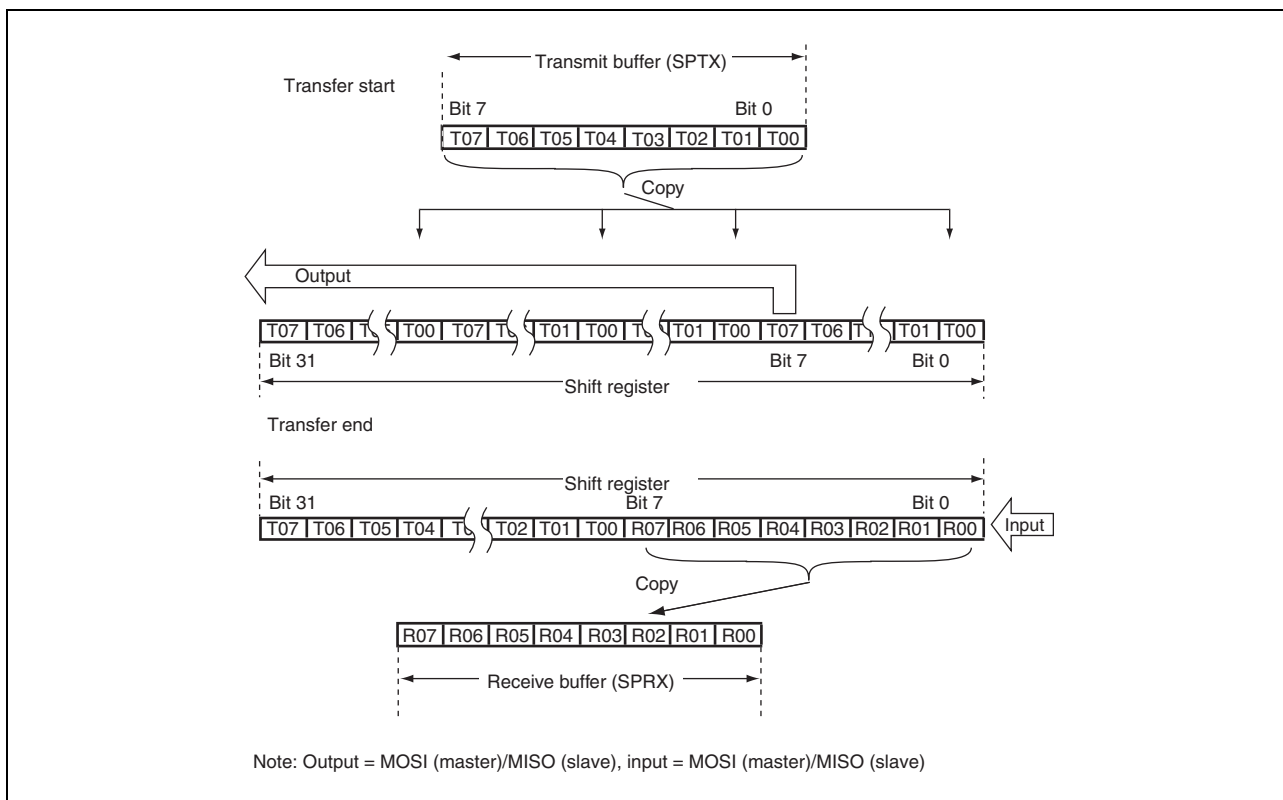


Figure 16.10 MSB First Transfer (8-Bit Data)

#### (4) LSB First Transfer (32-Bit Data)

Figure 16.11 shows the operation of the transmit buffer (SPTX) and the shift register when this module performs a 32-bit data length LSB-first data transfer.

The CPU or direct memory access controller writes T31 to T00 to the transmit buffer. If the shift register is empty, this module reverses the order of the bits of the data in the transmit buffer, copies it to the shift register, and fully populates the shift register. When serial transfer starts, this module outputs data from the MSB (bit 31) in the shift register, and shifts in the data from the LSB (bit 0) in the shift register. When the RSPCK cycle required for the serial transfer of 32 bits has passed, data R00 to R31 is stored in the shift register. In this state, this module copies the data, in which the order of the bits is reversed, from the shift register to the receive buffer, and empties the shift register. If the receive buffer does not have a space for the receive data length after receive data has been copied from the shift register to the receive buffer, another serial transfer will not be started. In order to start another serial transfer, data for the receive data length should be read from the receive buffer to secure the necessary space in the receive buffer.

If another serial transfer is started before the CPU or direct memory access controller writes to the transmit buffer of the SPDR, received data R00 to R31 is shifted out from the shift register.

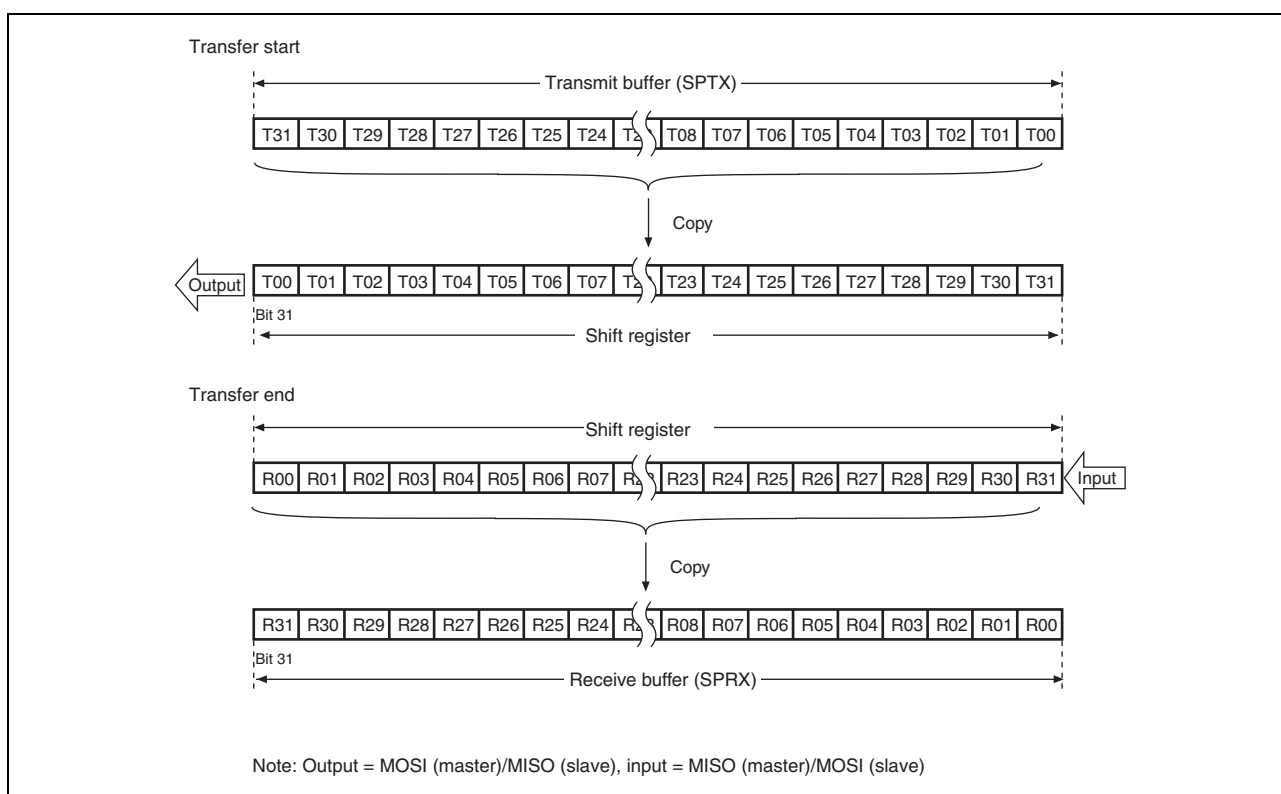


Figure 16.11 LSB First Transfer (32-Bit Data)

(5) LSB First Transfer (16-Bit Data)

Figure 16.12 shows the operation of the transmit buffer (SPTX) and the shift register when this module performs a 16-bit data length LSB-first data transfer.

The CPU or direct memory access controller writes T15 to T00 to the transmit buffer. If the shift register is empty, this module reverses the order of the bits of the data in the transmit buffer, copies it to the shift register, and fully populates the shift register. When serial transfer starts, this module outputs data from the MSB (bit 31) in the shift register, and shifts in the data from bit 16 in the shift register. When the RSPCK cycle required for the serial transfer of 16 bits has passed, received data R00 to R15 is stored in bits 31 to 16 in the shift register. After completion of the serial transfer, data that existed before the transfer is retained in bits 15 to 0 in the shift register. In this state, this module copies the data, in which the order of the bits is reversed, from the shift register to the receive buffer of SPDR, and empties the shift register. If the receive buffer does not have a space for the receive data length after receive data has been copied from the shift register to the receive buffer, another serial transfer will not be started. In order to start another serial transfer, data for the receive data length should be read from the receive buffer to secure the necessary space in the receive buffer. If another serial transfer is started before the CPU or direct memory access controller writes to the transmit buffer of SPDR, received data R00 to R15 is shifted out from the shift register.

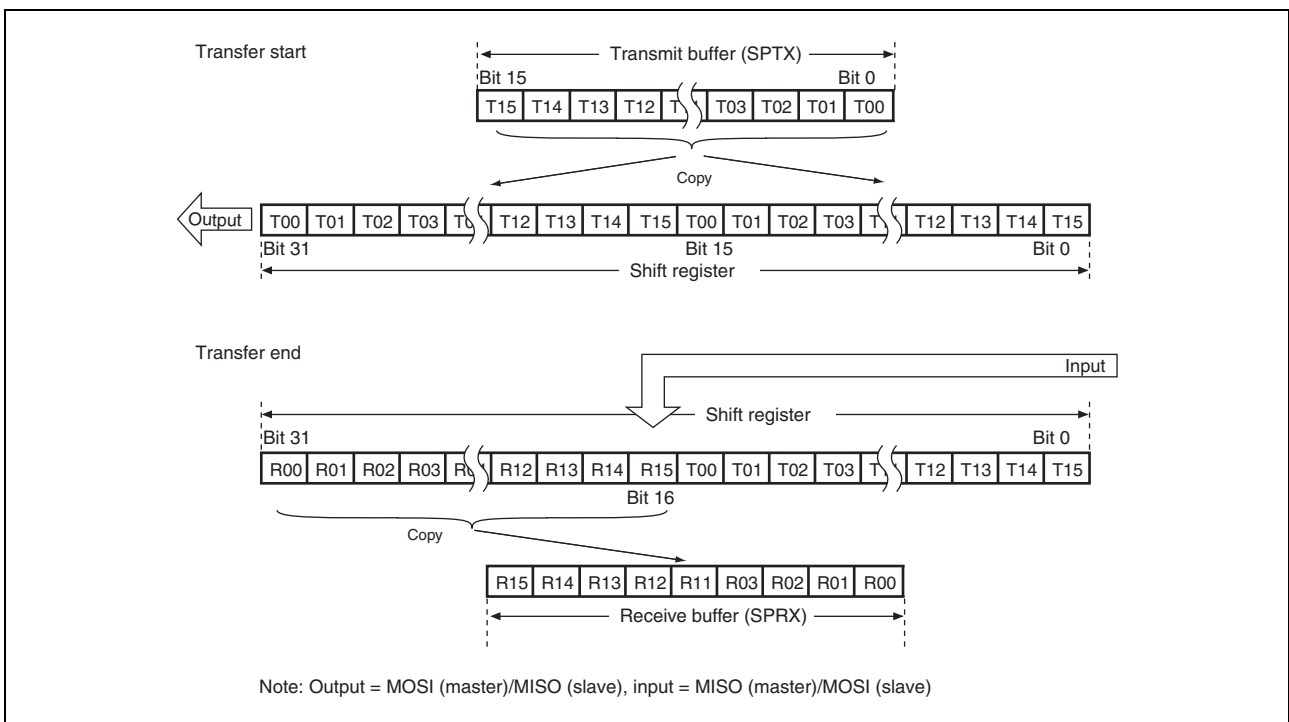


Figure 16.12 LSB First Transfer (16-Bit Data)

(6) LSB First Transfer (8-Bit Data)

Figure 16.13 shows the operation of the transmit buffer (SPTX) and the shift register when this module performs an 8-bit data length LSB-first data transfer.

The CPU or direct memory access controller writes T07 to T00 to the transmit buffer. If the shift register is empty, this module reverses the order of the bits of the data in the transmit buffer, copies it to the shift register, and fully populates the shift register. When serial transfer starts, this module outputs data from the MSB (bit 31) in the shift register, and shifts in the data from bit 24 in the shift register. When the RSPCK cycle required for the serial transfer of 8 bits has passed, received data R00 to R07 is stored in bits 31 to 24 of the shift register. After completion of the serial transfer, data that existed before the transfer is retained in bits 23 to 0 in the shift register. In this state, this module copies the data, in which the order of the bits is reversed, from the shift register to the receive buffer of SPDR, and empties the shift register. If the receive buffer does not have a space for the receive data length after the receive data has been copied from the shift register to the receive buffer, another serial transfer will not be started. In order to start another serial transfer, data for the receive data length should be read from the receive buffer to secure the necessary space in the receive buffer. If another serial transfer is started before the CPU or direct memory access controller writes to the transmit buffer of SPDR, received data R00 to R07 is shifted out from the shift register.

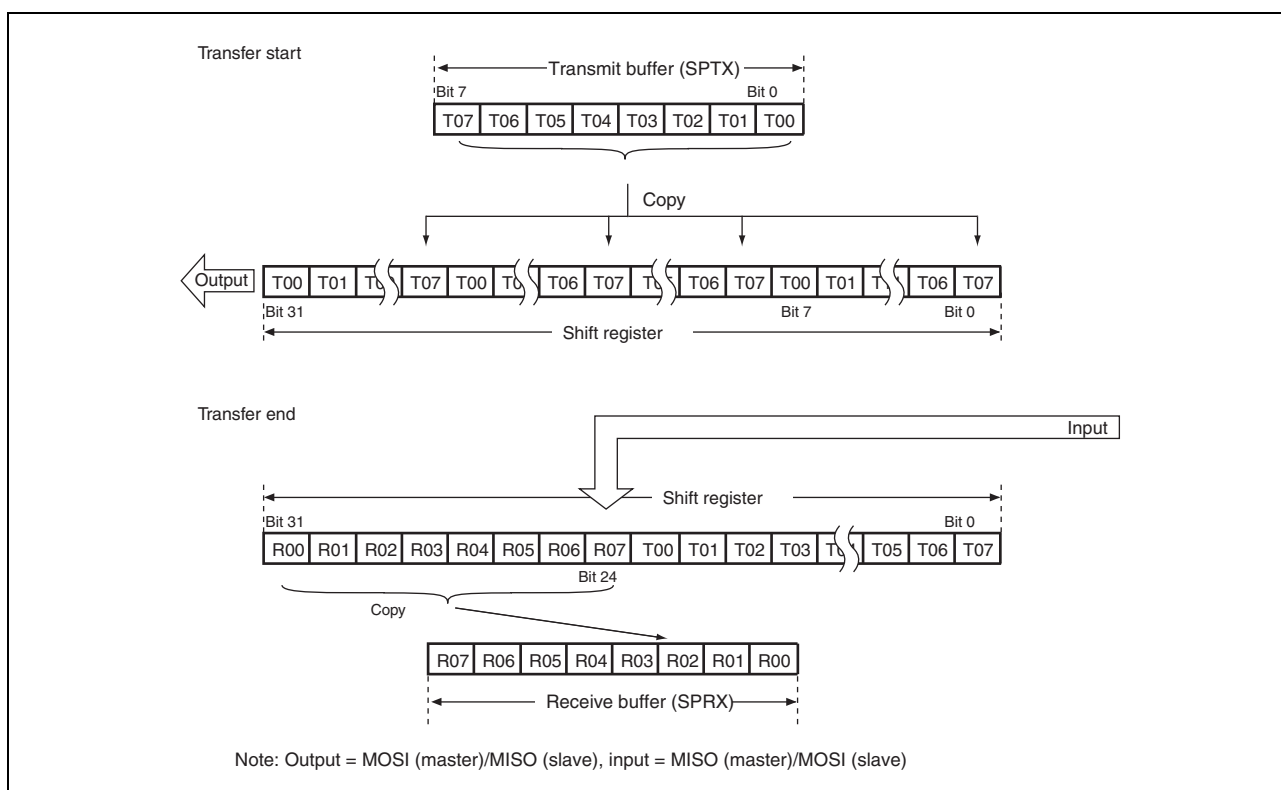


Figure 16.13 LSB First Transfer (8-Bit Data)

### 16.4.6 Error Detection

In the normal serial transfer, the data written to the transmit buffer of the data register (SPDR) is serially transmitted, and the serially received data can be read from the receive buffer of SPDR. If access is made to SPDR, depending on the status of the transmit buffer/receive buffer or the status at the beginning or end of serial transfer, in some cases non-normal transfers can be executed.

If a non-normal transfer operation occurs, this module detects the event as an overrun error or a mode fault error. Table 16.7 shows the relationship between non-normal transfer operations and the error detection function.

**Table 16.7 Relationship between Non-Normal Transfer Operations and Error Detection Function**

	Occurrence Condition	Operation	Error Detection
A	SPDR is written when the transmit buffer is full.	Missing write data.	None
B	Serial transfer is started in slave mode when transmit data is still not loaded on the shift register.	Data received in previous serial transfer is serially transmitted.	None
C	SPDR is read when the receive buffer is empty.	The output data is undefined.	None
D	Serial transfer terminates when the receive buffer is full.	Missing serial receive data.	Overrun error (only in slave mode)
E	The SSL input signal is negated during serial transfer in slave mode.	Serial transfer suspended. Missing send/receive data. Operation disabled.	Mode fault error

On operation A shown in Table 16.7, this module does not detect an error. Whether SPDR can be written to or not can be checked using the T[3:0] bits in the buffer data count setting register (SPBFDR).

Likewise, this module does not detect an error on operation B. In a serial transfer that was started before the shift register was updated, this module sends the data that was received in the previous serial transfer, and does not treat the operation indicated in B as an error. Note that the received data from the previous serial transfer is retained in the receive buffer of SPDR, thus it can be correctly read.

Similarly, this module does not detect an error on operation C. To prevent extraneous data from being read, the number of receive data units stored in the receive buffer should be read from the R[5:0] bits in the buffer data count setting register (SPBFDR).

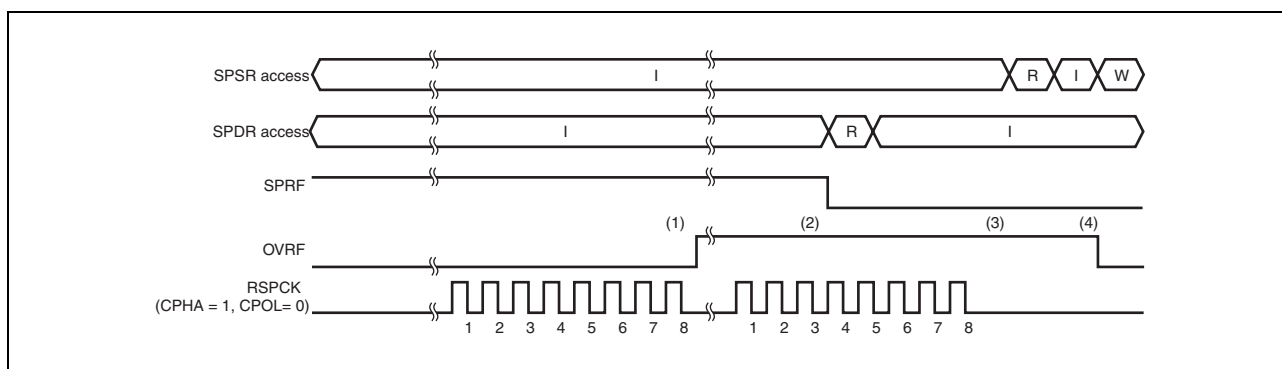
An overrun error shown in D is described in section 16.4.6 (1), Overrun Error. A mode fault error shown in E is described in section 16.4.6 (2), Mode Fault Error.



### (1) Overrun Error

If serial transfer ends when the receive buffer of the data register (SPDR) is full, this module detects an overrun error, and sets the OVRF bit in SPSR to 1. When the OVRF bit is 1, this module does not copy data from the shift register to the receive buffer so that the data prior to the occurrence of the error is retained in the receive buffer. To reset the OVRF bit in SPSR to 0, either perform a power-on reset, or write a 0 to the OVRF bit after SPSR has been read with the OVRF bit set to 1.

Figure 16.14 shows an example of operation of the SPRF and OVRF bits in SPSR. The SPSR and SPDR accesses shown in Figure 16.14 indicate the condition of accesses to SPSR and SPDR, respectively, where I denotes an idle cycle, W a write cycle, and R a read cycle. In the example of Figure 16.14, this module performs an 8-bit serial transfer in which the CPHA bit in the command register (SPCMD) is 1, and CPOL is 0. The numbers given under the RSPCK waveform represent the number of RSPCK cycles (i.e., the number of transferred bits).



**Figure 16.14 SPRF and OVRF Bit Operation Example**

The operation of the flags at the timing shown in steps (1) to (4) in the figure is described below.

1. If a serial transfer terminates when the receive buffer does not have a space for the receive data length, this module detects an overrun error, and sets the OVRF bit to 1. This module does not copy the data in the shift register to the receive buffer.
2. The OVRF bit is not cleared even when SPDR is read and thus the number of data bytes in the receive buffer becomes less than the number of the receive buffer data triggering number specified by the RXTRG bits.
3. If the serial transfer terminates in an overrun error state, this module determines that the shift register is empty; in this manner, data transfer is enabled from the transmit buffer to the shift register.
4. If 0 is written to the OVRF bit after SPSR is read with OVRF = 1, this module clears the OVRF bit.

The occurrence of an overrun can be checked either by reading SPSR or by using an error interrupt and reading SPSR. When using an error interrupt, set the SPEIE bit in the control register (SPCR) to 1. When executing a serial transfer without using an error interrupt, measures should be taken to ensure the early detection of overrun errors, such as reading SPSR immediately after SPDR is read.

The OVRF bit is cleared to 0 under the following conditions:

- After SPSR is read in a condition in which the OVRF bit is set to 1, 0 is written to the OVRF bit.
- Power-on reset

Note: • When the receive buffer has area enough to store receive data with an overrun error, this module receives receive data.

## (2) Mode Fault Error

When the MSTR bit is 0, this module operates in slave mode. This module detects a mode fault error if the SSL input signal is negated during the serial transfer period (from the time the driving of valid data is started to the time the final valid data is fetched) when the MODFEN bit is 1 in slave mode.

Upon detecting a mode fault error, this module stops driving of the output signals and clears the SPE bit in SPCR to 0. When the SPE bit is cleared to 0, the function of this module is disabled and this module stops driving external signals. For details of disabling the function of this module by clearing the SPE bit to 0, see [section 16.4.7, Initialization](#).

The occurrence of a mode fault error can be checked either by reading SPSR or by using an error interrupt and reading SPSR. When using an error interrupt, set the SPEIE bit in the control register (SPCR) to 1. To detect a mode fault error without using an error interrupt, it is necessary to poll SPSR.

When the MODF bit is 1, writing 1 to the SPE bit is ignored. To enable the function of this module after the detection of a mode fault error, the MODF bit must be set to 0. The MODF bit is cleared to 0 under the following conditions:

- After SPSR is read in a condition where the MODF bit has turned 1, 0 is written to the MODF bit.
- Power-on reset

## 16.4.7 Initialization

If 0 is written to the SPE bit in the control register (SPCR) or this module clears the SPE bit to 0 because of the detection of a mode fault error, this module disables the module function, and initializes a part of the module function. When a power-on reset is generated, this module initializes all of the module function. An explanation follows of initialization by the clearing of the SPE bit.

### (1) Initialization by Clearing SPE Bit

When the SPE bit in SPCR is cleared, this module performs the following initialization:

- Suspending any serial transfer that is being executed
- Stopping the driving of output signals (Hi-Z) in slave mode
- Initializing the internal state
- Initializing the TEND bit in SPSR

Initialization by the clearing of the SPE bit does not initialize the control bits of this module. For this reason, this module can be started in the same transfer mode as prior to the initialization if the SPE bit is re-set to 1.

## 16.4.8 SPI Operation

### (1) Multi-Master Mode Operation

This section explains the operation in multi-master mode.

#### (a) Starting Serial Transfer

A serial transfer is started when transmit data is copied from the transmit buffer to the shift register, the shift register becomes full, and the receive buffer has a space for the receive data length. If transmit data has already been written to the shift register, data is not copied from the transmit buffer to the shift register.

For details of the transfer format, see [section 16.4.4, Transfer Format](#).

#### (b) Terminating Serial Transfer

Irrespective of the CPHA bit in the command register (SPCMD), this module terminates the serial transfer after transmitting an RSPCK edge corresponding to the final sampling timing. After the serial transfer is completed, receive data is copied from the shift register to the receive buffer. If the receive buffer does not have a space for the receive data length after receive data is copied from the shift register to the receive buffer, another serial transfer will not be performed. In order to perform another serial transfer, data for the receive data length should be read from the receive buffer to secure the space for the receive data.

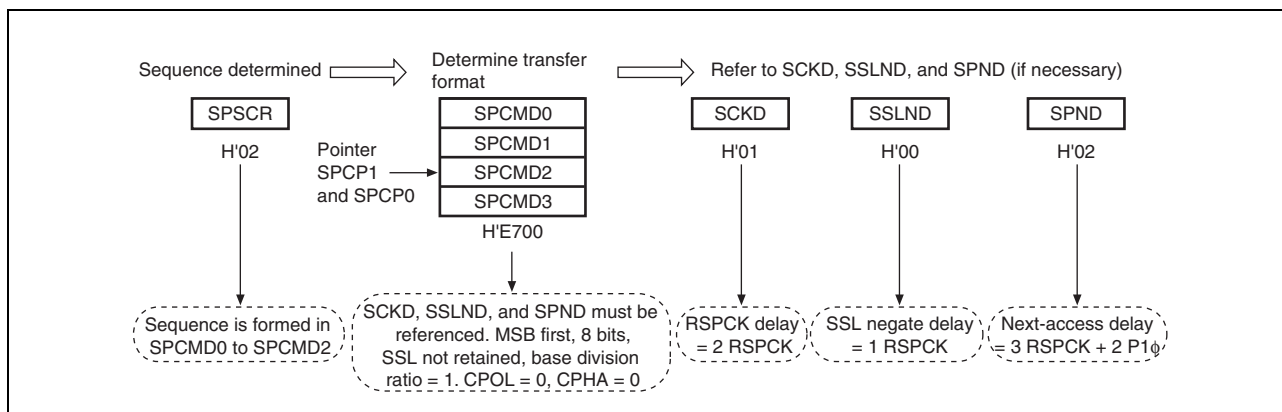
It should be noted that the final sampling timing varies depending on the bit length of transfer data. In master mode, the data length depends on the settings in bits SPB3 to SPB0 in SPCMD. For details on the transfer format, see [section 16.4.4, Transfer Format](#).

#### (c) Sequence Control

The transfer format that is employed in master mode is determined by the sequence control register (SPSCR), command registers 0 to 3 (SPCMD0 to SPCMD3), the bit rate register (SPBR), the clock delay register (SPCKD), the slave select negation delay register (SSLND), and the next-access delay register (SPND).

SPSCR is a register used to determine the sequence configuration for serial transfers that are executed by this module in master mode. The following items are set in command registers SPCMD0 to SPCMD3: SSL output signal value, MSB/LSB first, data length, some of the bit rate settings, RSPCK polarity/phase, whether SPCKD is to be referenced, whether SSLND is to be referenced, and whether SPND is to be referenced. SPBR holds some of the bit rate settings; SPCKD, a clock delay value; SSLND, an SSL negation delay; and SPND, a next-access delay value.

According to the sequence length that is assigned to SPSCR, this module makes up a sequence comprised of a part or all of SPCMD0 to SPCMD3. This module contains a pointer to the SPCMD that makes up the sequence. The value of this pointer can be checked by reading bits SPCP1 and SPCP0 in the sequence status register (SPSSR). When the SPE bit in the control register (SPCR) is set to 1 and the function of this module is enabled, this module loads the pointer to the commands in SPCMD0, and incorporates the SPCMD0 settings into the transfer format at the beginning of serial transfer. This module increments the pointer each time the next-access delay period for a data transfer ends. Upon completion of the serial transfer that corresponds to the final command comprising the sequence, this module sets the pointer in SPCMD0, and in this manner the sequence is executed repeatedly.



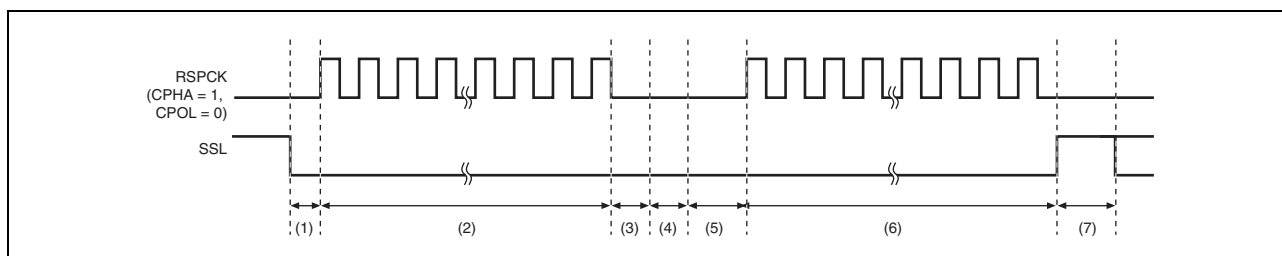
**Figure 16.15** Determination Procedure of Serial Transfer Mode in Master Mode

#### (d) Burst Transfer

If the SSLKP bit in the command register (SPCMD) that this module references during the current serial transfer is 1, this module keeps the SSL signal level during the serial transfer until the beginning of the SSL signal assertion for the next serial transfer. If the SSL signal level for the next serial transfer is the same as the SSL signal level for the current serial transfer, this module can execute continuous serial transfers while keeping the SSL signal assertion status (burst transfer).

Figure 16.16 shows an example of an SSL signal operation for the case where a burst transfer is implemented using SPCMD0 and SPCMD1 settings. The text below explains operations (1) to (7) as depicted in Figure 16.16. It should be noted that the polarity of the SSL output signal depends on the settings in the slave select polarity register (SSLP).

1. Based on SPCMD0, this module asserts the SSL signal and inserts RSPCK delays.
2. Serial transfers are executed according to SPCMD0.
3. SSL negation delays are inserted.
4. Because the SSLKP bit in SPCMD0 is 1, this module keeps the SSL signal value on SPCMD0. This period is sustained, at the shortest, for a period equal to the next-access delay of SPCMD0. If the shift register is empty after the passage of a minimum period, this period is sustained until such time as the transmit data is stored in the shift register for another transfer.
5. Based on SPCMD1, this module asserts the SSL signal and inserts RSPCK delays.
6. Serial transfers are executed according to SPCMD1.
7. Because the SSLKP bit in SPCMD1 is 0, this module negates the SSL signal. In addition, a next-access delay is inserted according to SPCMD1.



**Figure 16.16** Example of Burst Transfer Operation using SSLKP Bit

If the SSL signal settings in the SPCMD in which 1 is assigned to the SSLKP bit are different from the SSL signal output settings in the SPCMD to be used in the next transfer, this module switches the SSL signal status to SSL signal assertion ((5) in Figure 16.16) corresponding to the command for the next transfer. Notice that if such an SSL signal switching occurs, the slaves that drive the MISO signal compete, and the possibility arises of the collision of signal levels.

This module in master mode references within the module the SSL signal operation for the case where the SSLKP bit is

not used. Even when the CPHA bit in SPCMD is 0, this module can accurately start serial transfers by asserting the SSL signal for the next transfer. For this reason, burst transfers in master mode can be executed irrespective of CPHA bit settings (see section 16.4.8 (2), Slave Mode Operation).

#### (e) RSPCK Delay ( $t_1$ )

The RSPCK delay value in master mode depends on SCKDEN bit settings in the command register (SPCMD) and on clock delay register (SPCKD) settings. This module determines the SPCMD to be referenced during serial transfer by pointer control, and determines an RSPCK delay value during serial transfer by using the SCKDEN bit in the selected SPCMD and SPCKD, as shown in Table 16.8. For a definition of RSPCK delay, see section 16.4.4, Transfer Format.

**Table 16.8 Relationship among SCKDEN and SPCKD Settings and RSPCK Delay Values**

SCKDEN	SPCKD	RSPCK Delay Value
0	000 to 111	1 RSPCK
1	000	1 RSPCK
	001	2 RSPCK
	010	3 RSPCK
	011	4 RSPCK
	100	5 RSPCK
	101	6 RSPCK
	110	7 RSPCK
	111	8 RSPCK

#### (f) SSL Negation Delay ( $t_2$ )

The SSL negation delay value in master mode depends on SLNDEN bit settings in the command register (SPCMD) and on SSL negation delay register (SSLND) settings. This module determines the SPCMD to be referenced during serial transfer by pointer control, and determines an SSL negation delay value during serial transfer by using the SLNDEN bit in the selected SPCMD and SSLND, as shown in Table 16.9. For a definition of SSL negation delay, see section 16.4.4, Transfer Format.

**Table 16.9 Relationship among SLNDEN and SSLND Settings and SSL Negation Delay Values**

SLNDEN	SSLND	SSL Negation Delay Value
0	000 to 111	1 RSPCK
1	000	1 RSPCK
	001	2 RSPCK
	010	3 RSPCK
	011	4 RSPCK
	100	5 RSPCK
	101	6 RSPCK
	110	7 RSPCK
	111	8 RSPCK

**(g) Next-Access Delay ( $t_3$ )**

The next-access delay value in master mode depends on SPNDEN bit settings in the command register (SPCMD) and on next-access delay register (SPND) settings. This module determines the SPCMD to be referenced during serial transfer by pointer control, and determines a next-access delay value during serial transfer by using the SPNDEN bit in the selected SPCMD and SPND, as shown in Table 16.10. For a definition of next-access delay, see section 16.4.4, Transfer Format.

**Table 16.10 Relationship among SPNDEN and SPND Settings and Next-Access Delay Values**

SPNDEN	SPND	Next-Access Delay Value
0	000 to 111	1 RSPCK + 2 P1 $\phi$
1	000	1 RSPCK + 2 P1 $\phi$
	001	2 RSPCK + 2 P1 $\phi$
	010	3 RSPCK + 2 P1 $\phi$
	011	4 RSPCK + 2 P1 $\phi$
	100	5 RSPCK + 2 P1 $\phi$
	101	6 RSPCK + 2 P1 $\phi$
	110	7 RSPCK + 2 P1 $\phi$
	111	8 RSPCK + 2 P1 $\phi$

## (h) Initialization Flowchart

Figure 16.17 is a flowchart illustrating an example of initialization in SPI operation when this module is used in master mode. For a description of how to set up the interrupt controller, direct memory access controller, and input/output ports, see the descriptions given in the individual blocks.

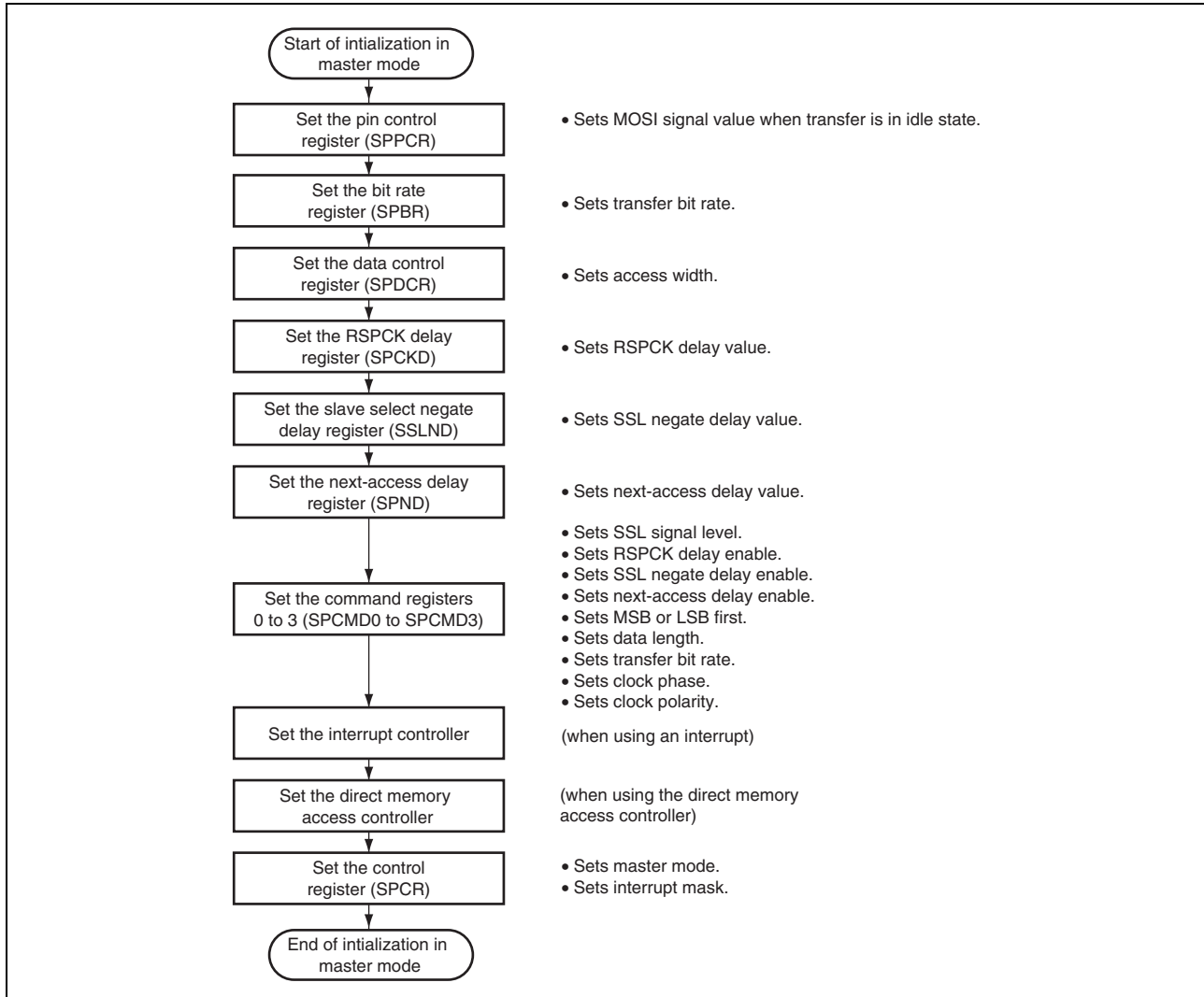


Figure 16.17 Example of Initialization Flowchart in Master Mode

(i) Transfer Operation Flowchart

Figure 16.18 is a flowchart illustrating a transfer in SPI operation when this module is used in master mode.

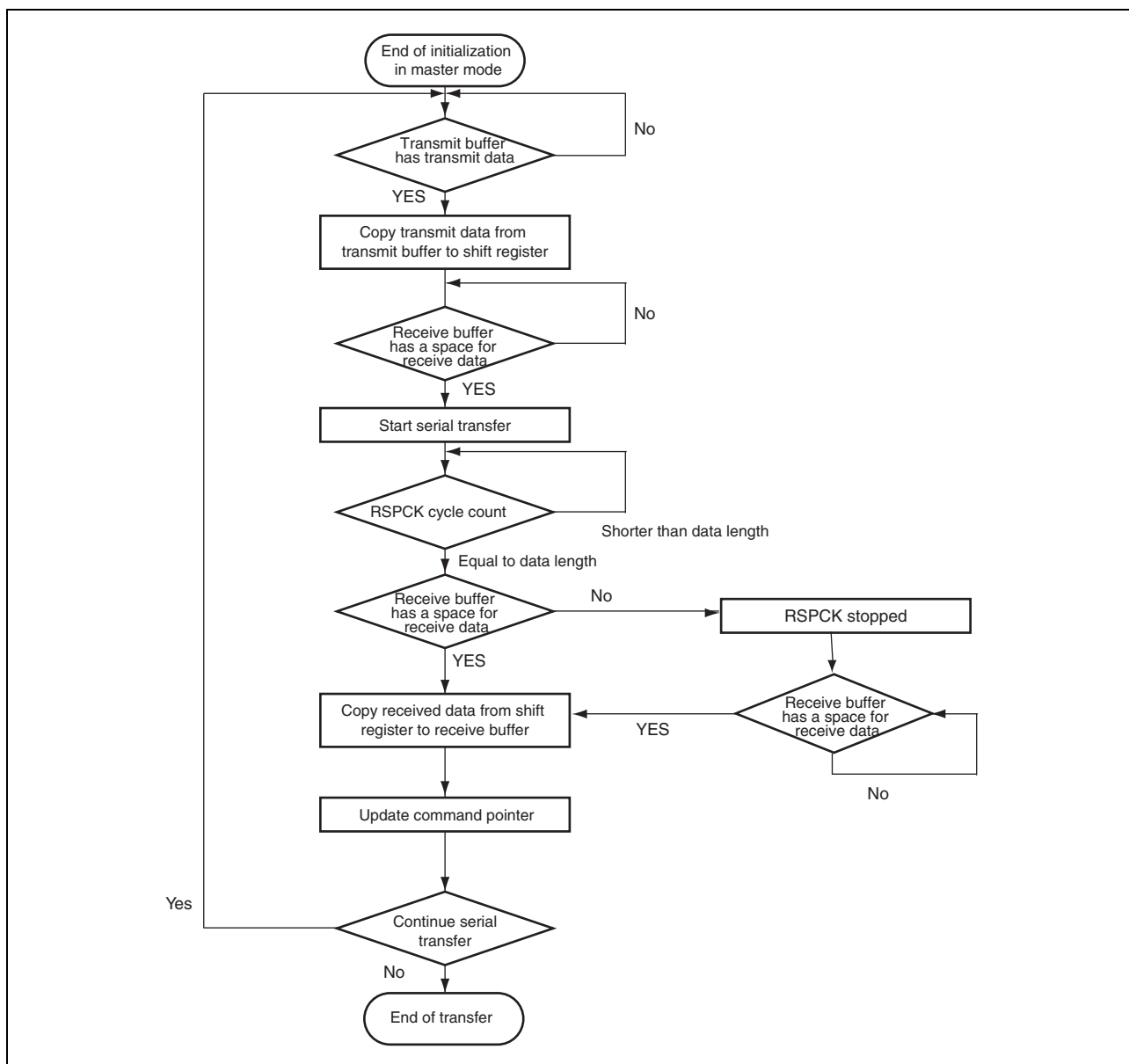


Figure 16.18 Transfer Operation Flowchart in Master Mode



## (2) Slave Mode Operation

### (a) Starting Serial Transfer

If this module detects an SSL input signal assertion when the CPHA bit in the command register 0 (SPCMD0) is 0, this module is required to start driving valid data to the MISO output signal. For this reason, when the CPHA bit is 0, the asserting of the SSL input signal triggers the start of a serial transfer.

If this module detects the first RSPCK edge in an SSL signal asserted condition when the CPHA bit is 1, this module is required to start driving valid data to the MISO output signal. For this reason, when the CPHA bit is 1, the first RSPCK edge in an SSL signal asserted condition triggers the start of a serial transfer.

When detecting the start of a serial transfer in a condition in which the shift register is empty, this module changes the status of the shift register to "full", so that data cannot be copied from the transmit buffer to the shift register when serial transfer is in progress. If the shift register was full before the serial transfer started, this module leaves the status of the shift register intact, in the full state.

Irrespective of CPHA bit settings, this module starts driving MISO output signals at the SSL signal assertion timing.

Whether the data output from this module is valid or invalid differs depending on CPHA bit settings.

For details on the transfer format, see [section 16.4.4, Transfer Format](#). The polarity of the SSL input signal depends on the setting of the SSL0P bit in the slave select polarity register (SSLP).

### (b) Terminating Serial Transfer

Irrespective of the CPHA bit in the command register 0 (SPCMD0), this module terminates the serial transfer after detecting an RSPCK edge corresponding to the final sampling timing. When the receive buffer has an enough space for receive data, this module copies received data from the shift register to the receive buffer of the data register (SPDR) upon termination of the serial transfer. Irrespective of the value of the SPRF bit, this module changes the status of the shift register to "empty" upon termination of the serial transfer. If this module detects an SSL input signal negation from the beginning of serial transfer to the end of serial transfer, a mode fault error occurs (see [section 16.4.6, Error Detection](#)).

The final sampling timing changes depending on the bit length of the transfer data. In slave mode, the data length depends on the settings in bits SPB3 to SPB0 bits in SPCMD0. The polarity of the SSL input signal depends on the setting in the SSL0P bit in the slave select polarity register (SSLP). For details on the transfer format, see [section 16.4.4, Transfer Format](#).

### (c) Notes on Slave Operations

If the CPHA bit in the command register 0 (SPCMD0) is 0, this module starts serial transfers when it detects the assertion edge for an SSL input signal. In the type of configuration shown in [Figure 16.4](#) as an example, if this module is used in single-slave mode, the SSL signal is always fixed at active state. Therefore, when the CPHA bit is set to 0, this module cannot correctly start a serial transfer. To correctly execute send/receive operation in a configuration in which the SSL input signal is fixed at active state, the CPHA bit should be set to 1. When it is necessary to set the CPHA bit to 0, the SSL input signal should not be fixed.

### (d) Burst Transfer

If the CPHA bit in the command register 0 (SPCMD0) is 1, continuous serial transfer (burst transfer) can be executed while retaining the assertion state for the SSL input signal. If the CPHA bit is 1, the period from the first RSPCK edge to the sampling timing for the reception of the final bit in an SSL signal active state corresponds to a serial transfer period. Even when the SSL input signal remains at the active level, this module can accommodate burst transfers because it can detect the start of access.

If the CPHA bit is 0, for the reason given in [section 16.4.8 \(2\) \(c\), Notes on Slave Operations](#), second and subsequent serial transfers during the burst transfer cannot be executed correctly.

## (e) Initialization Flowchart

Figure 16.19 is a flowchart illustrating an example of initialization in SPI operation when this module is used in slave mode. For a description of how to set up the interrupt controller, direct memory access controller, and input/output ports, see the descriptions given in the individual blocks.

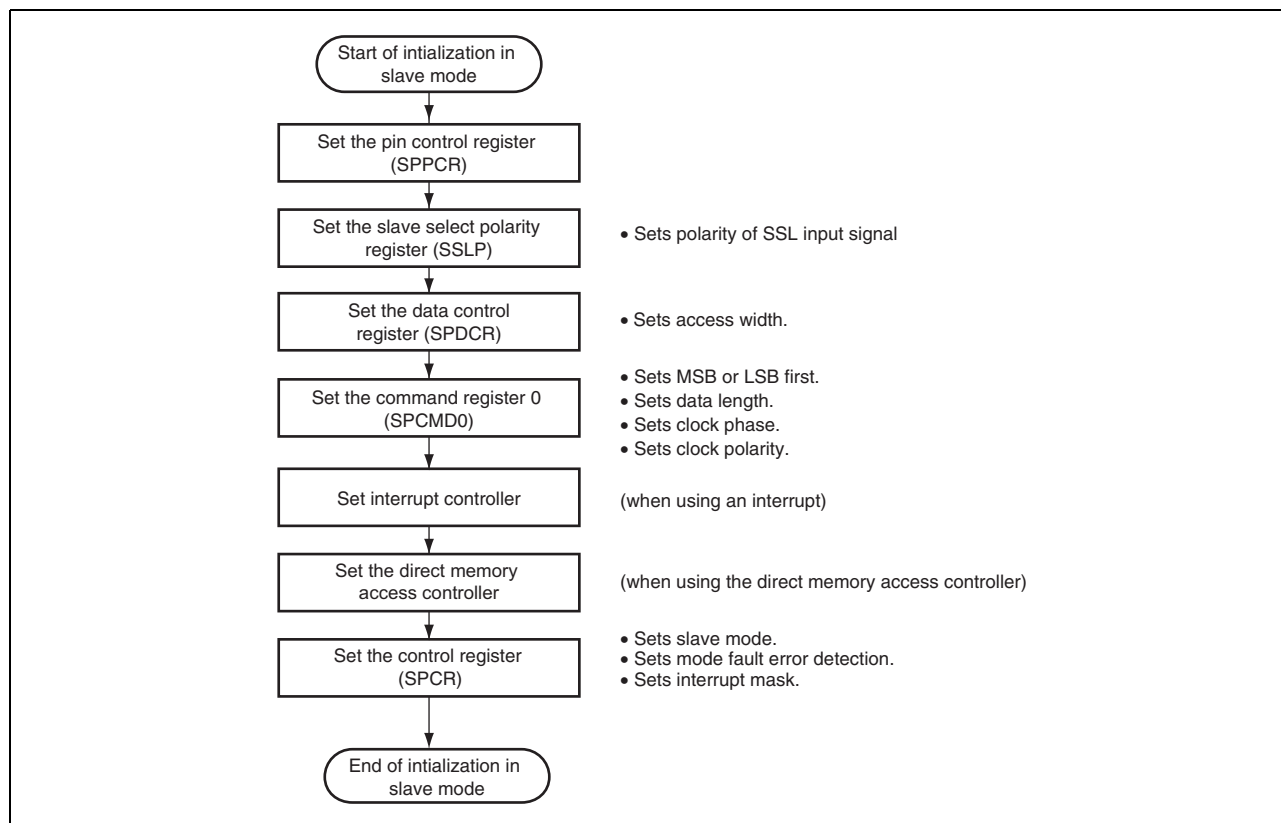


Figure 16.19 Example of Initialization Flowchart in Slave Mode

(f) Transfer Operation Flowchart (CPHA = 0)

Figure 16.20 is a flowchart illustrating a transfer in SPI operation when this module is used in slave mode with the CPHA bit in the command register 0 (SPCMD0) set to 0.

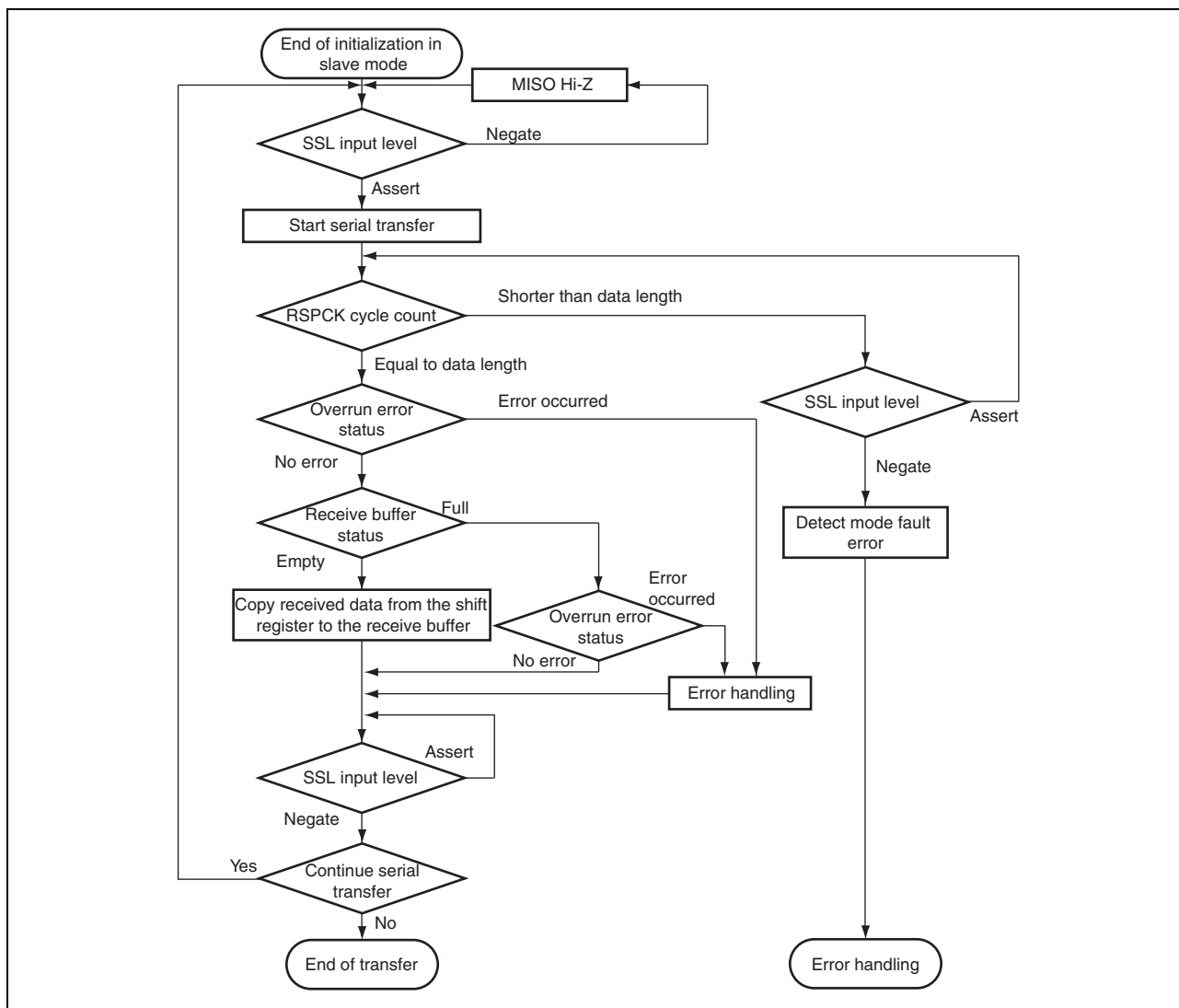


Figure 16.20 Transfer Operation Flowchart in Slave (CPHA = 0)

(g) Transfer Operation Flowchart (CPHA = 1)

Figure 16.21 is a flowchart illustrating a transfer in SPI operation when this module is used in slave mode with the CPHA bit in the command register 0 (SPCMD0) and the MODFEN bit in the control register (SPCR) set to 1, respectively. The subsequent operation is not guaranteed when the serial transfer is started with the MODFEN bit set to 0 and the SSL input level is negated with the number of RSPCK cycles shorter than the data length.

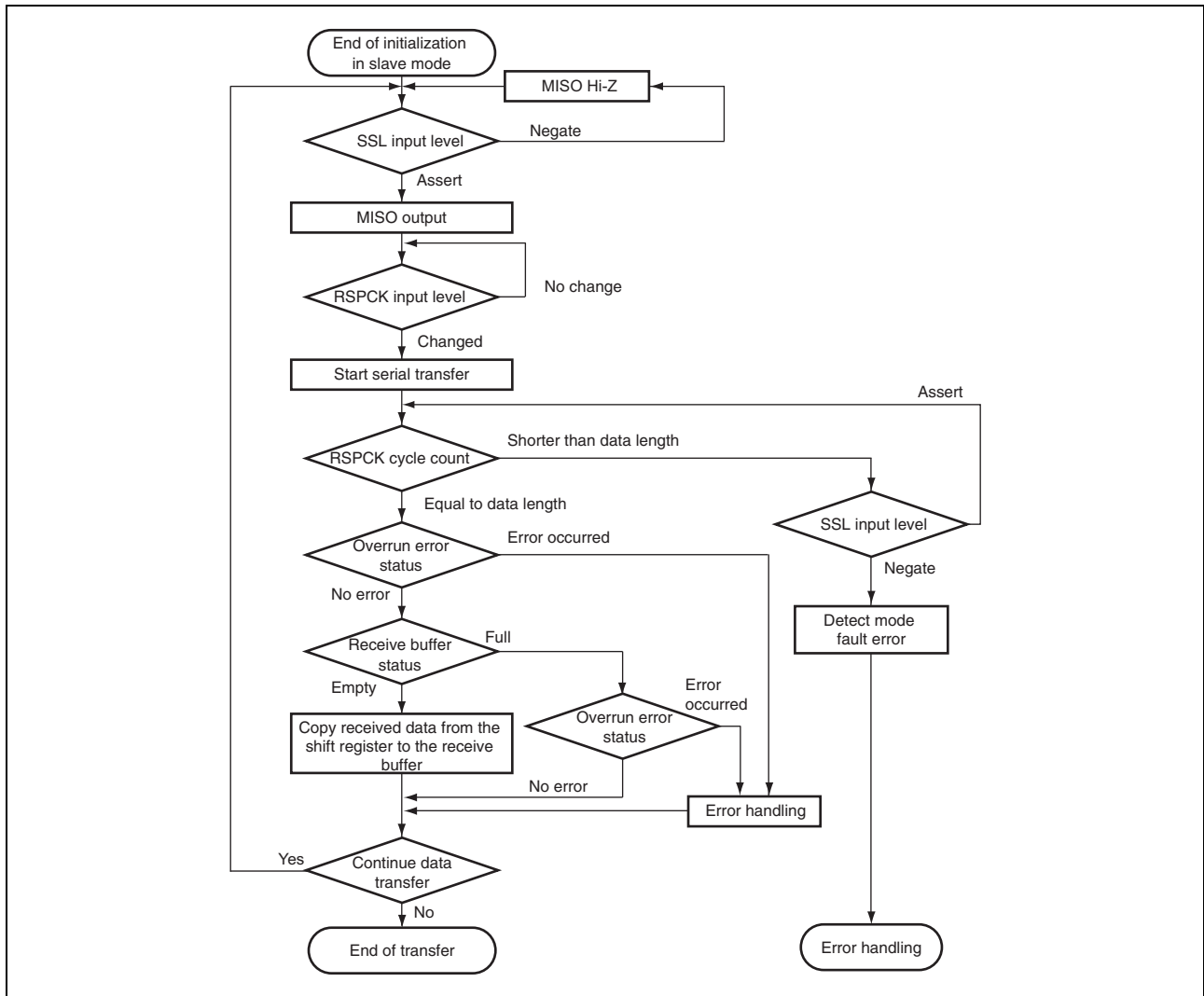


Figure 16.21 Transfer Operation Flowchart in Slave Mode (CPHA = 1)

### 16.4.9 Error Handling

Figure 16.22 and Figure 16.23 show the error handling. The following error handling is used to return from the error state after an error in master or slave mode.

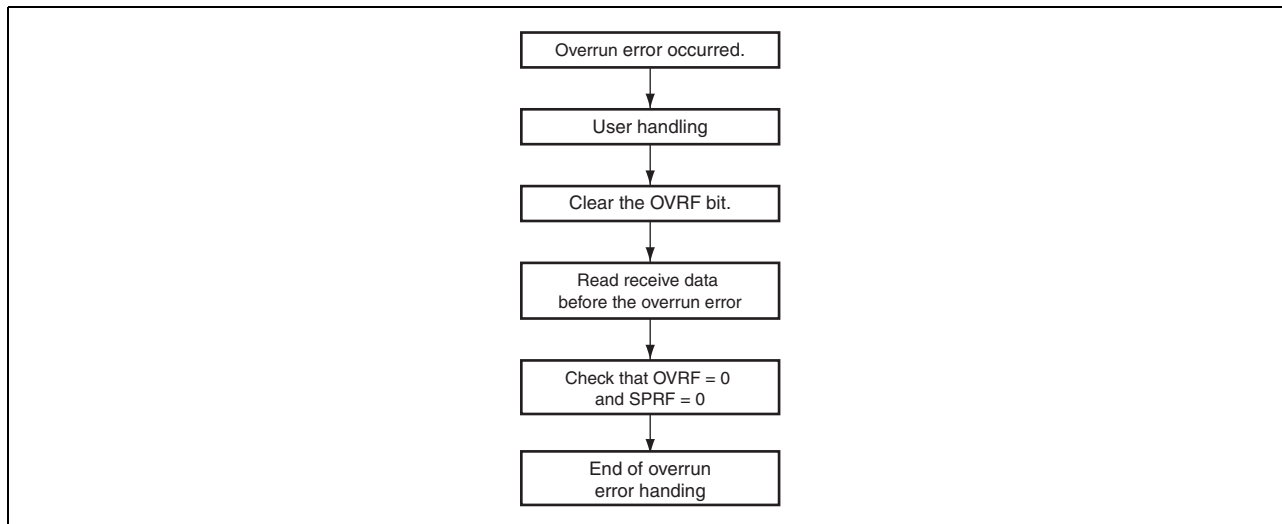


Figure 16.22 Error Handling (Overrun Error)

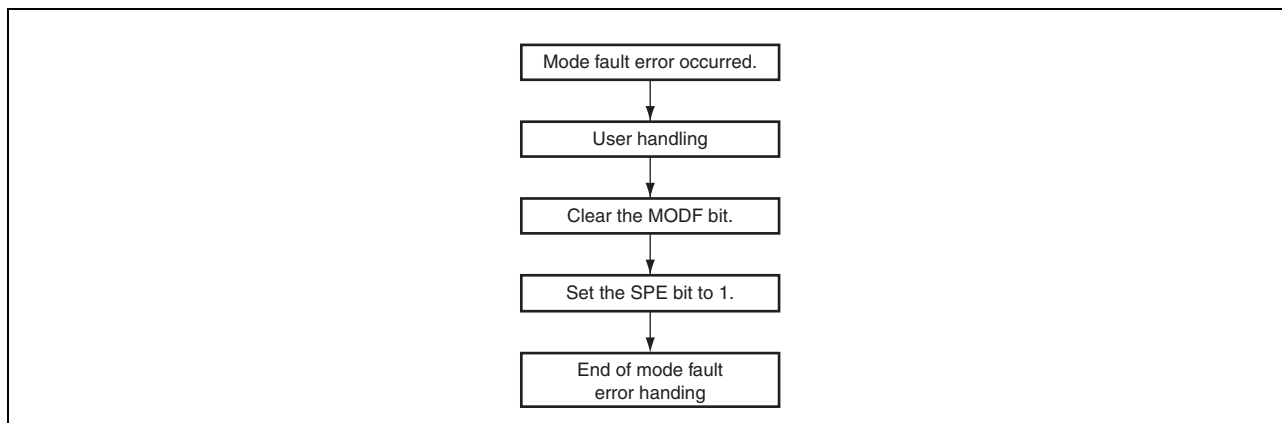
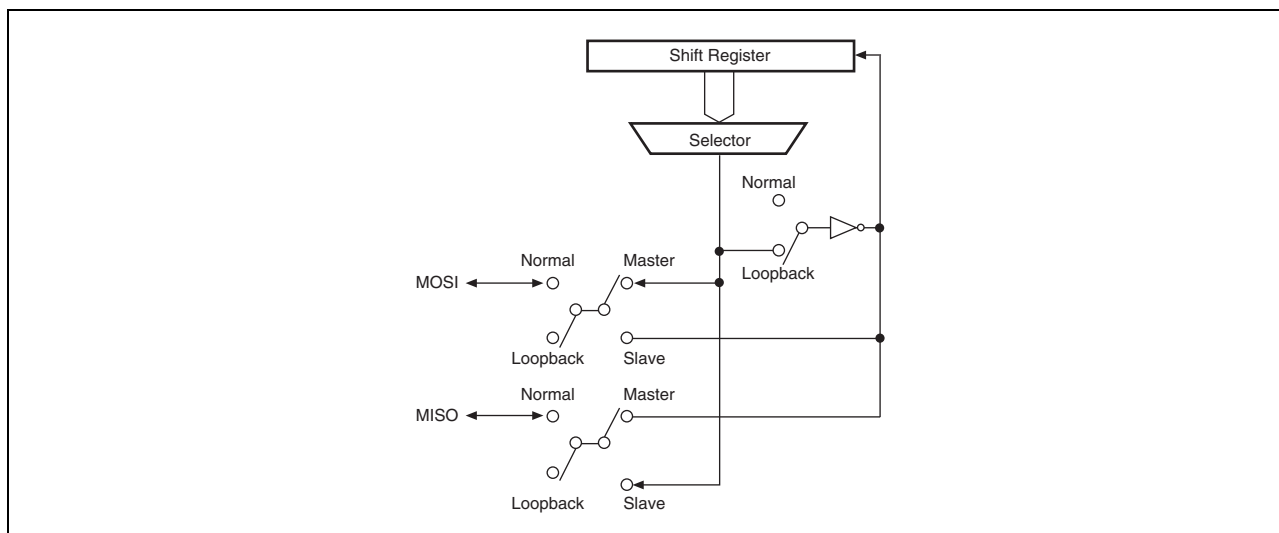


Figure 16.23 Error Handling (Mode Fault Error)

### 16.4.10 Loopback Mode

When 1 is written to the SPLP bit in the pin control register (SPPCR), this module shuts off the path between the MISO pin and the shift register, and between the MOSI pin and the shift register, and connects the input path and the output path (reversed) of the shift register. This is called loopback mode. When a serial transfer is executed in loopback mode, the transmit data becomes the received data. Figure 16.24 shows the configuration of the shift register input/output paths for the case where this module in master mode is set in loopback mode.



**Figure 16.24** Configuration of Shift Register Input/Output Paths in Loopback Mode (Master Mode)

### 16.4.11 Interrupt Sources

This module has interrupt sources of receive buffer full, transmit buffer empty, mode fault, and overrun. In addition, the direct memory access controller can be activated by the receive buffer full or transmit buffer empty interrupt for data transfer.

Table 16.11 shows the interrupt sources.

When any of the interrupt conditions in Table 16.11 is met, an interrupt is generated. The interrupt sources should be cleared with data transfer by the CPU or direct memory access controller.

**Table 16.11** Interrupt Sources

Name	Interrupt Source	Abbreviation	Interrupt Condition	Activation of Direct Memory Access Controller
SPRI	Receive buffer full	RXI	(SPRIE = 1) • (SPRF = 1)	Possible
SPTI	Transmit buffer empty	TXI	(SPTIE = 1) • (SPTEF = 1)	Possible
SPEI	Mode fault	MOI	(SPEIE = 1) • (MODF = 1)	—
	Overrun	OVI	(SPEIE = 1) • (OVRF = 1)	—

## 17. SPI Multi I/O Bus Controller

The SPI multi I/O bus controller outputs control signals to the serial flash memory connected to the SPI multi I/O bus space, thus enabling direct connection of the serial flash memory.

This LSI incorporates two independent SPI multi I/O bus controller channels.

### 17.1 Features

This module allows the connected serial flash memory to be accessed by directly reading the SPI multi I/O bus space, or using SPI operating mode to transmit and receive data.

- Serial Flash Memory Interface
  - Up to two serial flash memories per channel can be connected.
  - A data bus size of 1 bit, 2 bits, or 4 bits can be selected for one serial flash memory device.
  - Serial flash memory for DDR transfer can be directly connected.
- External Address Space Read Mode
  - A maximum of 8-Gbyte address space is supported (when two serial flash memories are connected)
  - The SPBSSL pin can be automatically controlled through access address monitoring
  - Efficient data reception due to built-in read cache (64-bit line × 16 entries)
- SPI Operating Mode
  - Desired read/write access to serial flash memory possible
- Bit rate
  - SPBCLK is generated by frequency division of B $\phi$  by internal baud rate generator
  - SPBCLK frequency division ratio can be set from 2 to 4080
- SPBSSL Pin Control
  - Delay from SPBSSL signal assertion to SPBCLK operation (clock delay) can be set
  - Range: 1 to 8 SPBCLK cycles (set in SPBCLK-cycle units)
  - Delay from SPBCLK stop to SPBSSL output negation (SPBSSL negation delay) can be set
  - Range: 1.5 to 8.5 SPBCLK cycles (set in SPBCLK-cycle units)
  - SPBSSL output assertion wait before next access (next access delay) can be set
  - Range: 1 to 8 SPBCLK cycles (set in SPBCLK-cycle units)
  - SPBSSL polarity can be changed

### 17.2 Block Diagram

Figure 17.1 shows a block diagram of this module for one channel.

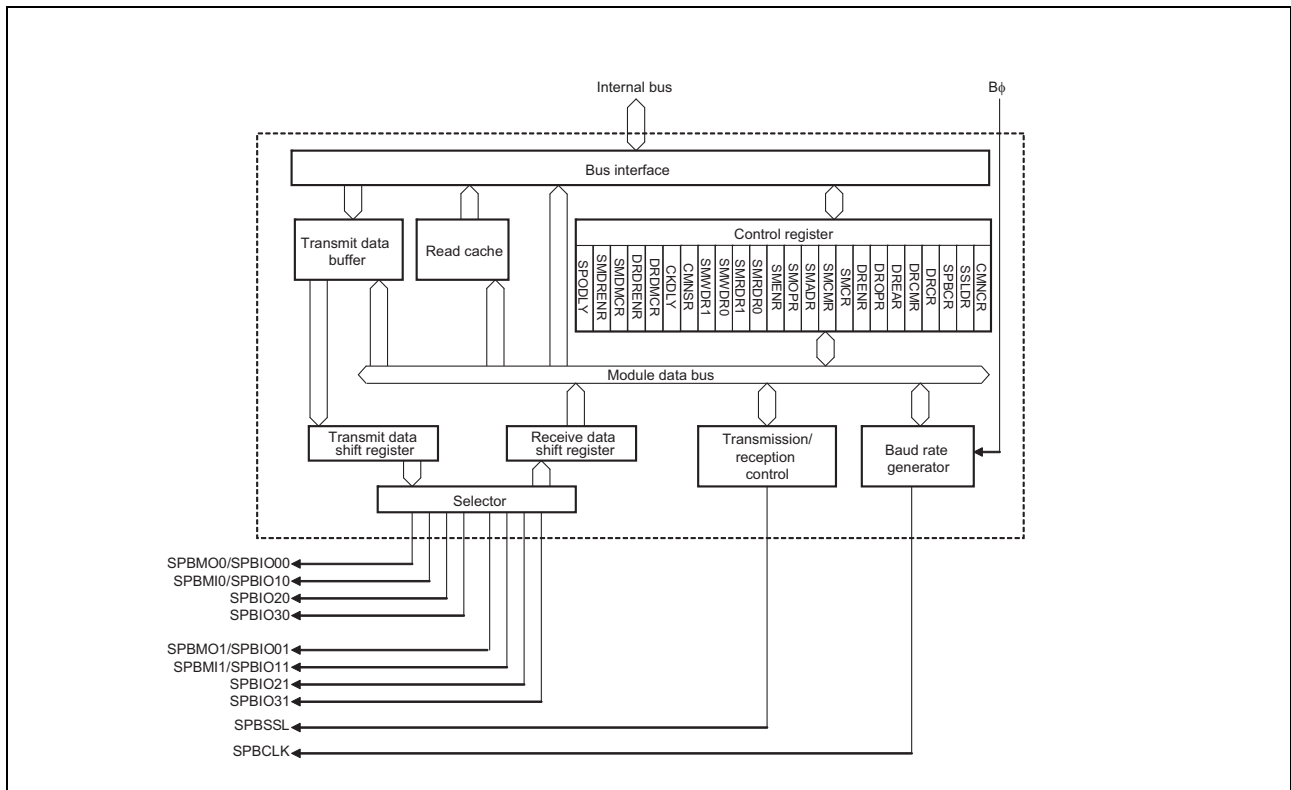


Figure 17.1 Block Diagram



### 17.3 Input/Output Pins

Table 17.1 shows the pin configuration for one channel.

**Table 17.1 Pin Configuration**

Port	Pin Name	Symbol	I/O	Function
Common	Clock pin	SPCLK_n	Output	Clock output
	Slave select pin	SPSSL_n	Output	Slave selection
0	Data 0 pin	SPMO0_n/ SPBIO00_n	I/O	Port 0 master transmit data/data 0
	Port 0 data 1 pin	SPMI0_n/ SPBIO10_n	I/O	Port 0 master input data/ data 1
	Port 0 data 2 pin	SPBIO20_n	I/O	Port 0 data 2
	Port 0 data 3 pin	SPBIO30_n	I/O	Port 0 data 3
1	Port 1 data 0 pin	SPMO1_n/ SPBIO01_n	I/O	Port 1 master transmit data/data 0
	Port 1 data 1 pin	SPMI1_n/ SPBIO11_n	I/O	Port 1 master input data/ data 1
	Port 1 data 2 pin	SPBIO21_n	I/O	Port 1 data 2
	Port 1 data 3 pin	SPBIO31_n	I/O	Port 1 data 3

Note: • n represents a channel number (0 or 1). In the text, the channel number is omitted.

## 17.4 Register Descriptions

Table 17.2 shows the register configuration.

**Table 17.2 Register Configuration**

Channel	Register Name	Abbreviation	Initial Value	R/W	Address	Access Size
0	Common control register_0	CMNCR_0	H'01AA4000	R/W	H'3FEFA000	32
	SSL delay register_0	SSLDR_0	H'00070707	R/W	H'3FEFA004	32
	Bit rate register_0	SPBCR_0	H'00000003	R/W	H'3FEFA008	32
	Data read control register_0	DRCR_0	H'00000000	R/W	H'3FEFA00C	32
	Data read command setting register_0	DRCMR_0	H'00030000	R/W	H'3FEFA010	32
	Data read extended address setting register_0	DREAR_0	H'00000000	R/W	H'3FEFA014	32
	Data read option setting register_0	DROPR_0	H'00000000	R/W	H'3FEFA018	32
	Data read enable setting register_0	DRENr_0	H'00004700	R/W	H'3FEFA01C	32
	SPI mode control register_0	SMCR_0	H'00000000	R/W	H'3FEFA020	32
	SPI mode command setting register_0	SMCMR_0	H'00000000	R/W	H'3FEFA024	32
	SPI mode address setting register_0	SMADR_0	H'00000000	R/W	H'3FEFA028	32
	SPI mode option setting register_0	SMOPR_0	H'00000000	R/W	H'3FEFA02C	32
	SPI mode enable setting register_0	SMENR_0	H'00004000	R/W	H'3FEFA030	32
	SPI mode read data register 0_0	SMRDR0_0	Undefined	R	H'3FEFA038	8, 16, 32
	SPI mode read data register 1_0	SMRDR1_0	Undefined	R	H'3FEFA03C	8, 16, 32
	SPI mode write data register 0_0	SMWDR0_0	H'00000000	R/W	H'3FEFA040	8, 16, 32
	SPI mode write data register 1_0	SMWDR1_0	H'00000000	R/W	H'3FEFA044	8, 16, 32
	Common status register_0	CMNSR_0	H'00000001	R	H'3FEFA048	32
	SPI AC input characteristics adjustment register_0	CKDLY_0	H'00000004	R/W	H'3FEFA050	32
	Data read dummy cycle setting register_0	DRDMCR_0	H'00000000	R/W	H'3FEFA058	32
	Data read DDR enable register_0	DRDRENr_0	H'00000000	R/W	H'3FEFA05C	32
	SPI mode dummy cycle setting register_0	SMDMCR_0	H'00000000	R/W	H'3FEFA060	32
	SPI mode DDR enable register_0	SMDRENr_0	H'00000000	R/W	H'3FEFA064	32
	SPI AC output characteristics adjustment register_0	SPODLY_0	H'00000000	R/W	H'3FEFA068	32

**Table 17.2 Register Configuration**

Channel	Register Name	Abbreviation	Initial Value	R/W	Address	Access Size
1	Common control register_1	CMNCR_1	H'01AA4000	R/W	H'3FEFB000	32
	SSL delay register_1	SSLDR_1	H'00070707	R/W	H'3FEFB004	32
	Bit rate register_1	SPBCR_1	H'00000003	R/W	H'3FEFB008	32
	Data read control register_1	DRCR_1	H'00000000	R/W	H'3FEFB00C	32
	Data read command setting register_1	DRCMR_1	H'00030000	R/W	H'3FEFB010	32
	Data read extended address setting register_1	DREAR_1	H'00000000	R/W	H'3FEFB014	32
	Data read option setting register_1	DROPR_1	H'00000000	R/W	H'3FEFB018	32
	Data read enable setting register_1	DRENr_1	H'00004700	R/W	H'3FEFB01C	32
	SPI mode control register_1	SMCR_1	H'00000000	R/W	H'3FEFB020	32
	SPI mode command setting register_1	SMCMR_1	H'00000000	R/W	H'3FEFB024	32
	SPI mode address setting register_1	SMADR_1	H'00000000	R/W	H'3FEFB028	32
	SPI mode option setting register_1	SMOPR_1	H'00000000	R/W	H'3FEFB02C	32
	SPI mode enable setting register_1	SMENR_1	H'00004000	R/W	H'3FEFB030	32
	SPI mode read data register 0_1	SMRDR0_1	Undefined	R	H'3FEFB038	8, 16, 32
	SPI mode read data register 1_1	SMRDR1_1	Undefined	R	H'3FEFB03C	8, 16, 32
	SPI mode write data register 0_1	SMWDR0_1	H'00000000	R/W	H'3FEFB040	8, 16, 32
	SPI mode write data register 1_1	SMWDR1_1	H'00000000	R/W	H'3FEFB044	8, 16, 32
	Common status register_1	CMNSR_1	H'00000001	R	H'3FEFB048	32
	SPI AC input characteristics adjustment register_1	CKDLY_1	H'00000004	R/W	H'3FEFB050	32
	Data read dummy cycle setting register_1	DRDMCR_1	H'00000000	R/W	H'3FEFB058	32
	Data read DDR enable register_1	DRDRENr_1	H'00000000	R/W	H'3FEFB05C	32
	SPI mode dummy cycle setting register_1	SMDMCR_1	H'00000000	R/W	H'3FEFB060	32
	SPI mode DDR enable register_1	SMDRENr_1	H'00000000	R/W	H'3FEFB064	32
	SPI AC output characteristics adjustment register_1	SPODLY_1	H'00000000	R/W	H'3FEFB068	32

Note: • In the text, the channel number is omitted.

### 17.4.1 Common Control Register (CMNCR)

CMNCR is a 32-bit register that controls the SPI multi I/O bus controller. The settings of this register are reflected both in external address space read mode and SPI operating mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MD	-	-	-	-	-	-	SFDE	MOIIIO3[1:0]	MOIIIO2[1:0]	MOIIIO1[1:0]	MOIIIO0[1:0]				
Initial value:	0	0	0	0	0	0	0	1	1	0	1	0	1	0	1	0
R/W:	R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IO3FV[1:0]	IO2FV[1:0]	-	-	IO0FV[1:0]	-	CPHAT	CPHAR	SSLP	CPOL	-	BSZ[1:0]				
Initial value:	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R/W	R/W	R/W	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	MD	0	R/W	Operating Mode Switch Switches the operating modes. 0: External address space read mode 1: SPI operating mode
30 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	SFDE	1	R/W	Data Swap Setting for Serial Flash Memory Specifies whether or not swapping of data in serial flash memory is performed. 0: Swapping is not performed. 1: Swapping is performed in 8-bit units. For details, see section 17.5.4, Data Alignment.
23, 22	MOIIIO3[1:0]	10	R/W	SPBSSL Output Idle Value Fix SPBIO30, SPBIO31 Fixes output values of SPBIO30 and SPBIO31 in SPBSSL negation period. 00: Output value 0 01: Output value 1 10: Output value is the last bit value of the previous transfer (Hi-Z, if Hi-Z was the last bit value of the previous transfer). 11: Output value Hi-Z
21, 20	MOIIIO2[1:0]	10	R/W	SPBSSL Output Idle Value Fix SPBIO20, SPBIO21 Fixes output values of SPBIO20 and SPBIO21 in SPBSSL negation period. 00: Output value 0 01: Output value 1 10: Output value is the last bit value of the previous transfer (Hi-Z, if Hi-Z was the last bit value of the previous transfer). 11: Output value Hi-Z
19, 18	MOIIIO1[1:0]	10	R/W	SPBSSL Output Idle Value Fix SPBIO10, SPBIO11 Fixes output values of SPBIO10 and SPBIO11 in SPBSSL negation period. 00: Output value 0 01: Output value 1 10: Output value is the last bit value of the previous transfer (Hi-Z, if Hi-Z was the last bit value of the previous transfer). 11: Output value Hi-Z
17, 16	MOIIIO0[1:0]	10	R/W	SPBSSL Output Idle Value Fix SPBIO00, SPBIO01 Fixes output values of SPBIO00 and SPBIO01 in SPBSSL negation period. 00: Output value 0 01: Output value 1 10: Output value is the last bit value of the previous transfer (Hi-Z, if Hi-Z was the last bit value of the previous transfer). 11: Output value Hi-Z

Bit	Bit Name	Initial Value	R/W	Description															
15, 14	IO3FV[1:0]	01	R/W	SPBIO30, SPBIO31 Fixed Value for 1-bit/2-bit Size Fixes the output value of SPBIO30 and SPBIO31 pins for 1-bit/2-bit size. 00: Output value 0 01: Output value 1 10: Output value is the last bit value of the previous transfer (Hi-Z, if Hi-Z was the last bit value of the previous transfer). 11: Output value Hi-Z															
13, 12	IO2FV[1:0]	00	R/W	SPBIO20, SPBIO21 Fixed Value for 1-bit/2-bit Size Fixes the output value of SPBIO20 and SPBIO21 pins for 1-bit/2-bit size. 00: Output value 0 01: Output value 1 10: Output value is the last bit value of the previous transfer (Hi-Z, if Hi-Z was the last bit value of the previous transfer). 11: Output value Hi-Z															
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.															
9, 8	IO0FV[1:0]	00	R/W	SPBIO00, SPBIO01 Fixed Value for 1-bit Size Input Fixes the output value of SPBIO00 and SPBIO01 pins for 1-bit size input. 00: Output value 0 01: Output value 1 10: Output value is the last bit value of the previous transfer (Hi-Z, if Hi-Z was the last bit value of the previous transfer). 11: Output value Hi-Z															
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.															
6	CPHAT	0	R/W	Output Shift Sets the SPBCLK edge of the output data. CPHAT and CPHAR should be set according to the table in the description of CPHAR. 0: Data transmission at even edge during SDR transfer Data transmission starts at even edge during DDR transfer. 1: Data transmission at odd edge during SDR transfer Data transmission starts at odd edge during DDR transfer.															
5	CPHAR	0	R/W	Input Latch Sets the SPBCLK edge of the reception data. CPHAT and CPHAR should be set according to the following table. 0: Data reception at odd edge during SDR transfer Data reception starts at odd edge during DDR transfer. 1: Data reception at even edge during SDR transfer Data reception starts at even edge during DDR transfer.  <table border="1"> <thead> <tr> <th>CPHAT</th> <th>CPHAR</th> <th></th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>Setting enabled</td> </tr> <tr> <td>0</td> <td>1</td> <td>Setting enabled during SDR transfer</td> </tr> <tr> <td>1</td> <td>0</td> <td>Setting prohibited</td> </tr> <tr> <td>1</td> <td>1</td> <td>Setting enabled</td> </tr> </tbody> </table> Note: To set DDR transfer, set both the CPHAT and CPHAR bits to 0 or 1.	CPHAT	CPHAR		0	0	Setting enabled	0	1	Setting enabled during SDR transfer	1	0	Setting prohibited	1	1	Setting enabled
CPHAT	CPHAR																		
0	0	Setting enabled																	
0	1	Setting enabled during SDR transfer																	
1	0	Setting prohibited																	
1	1	Setting enabled																	
4	SSLP	0	R/W	SPBSSL Signal Polarity Sets the polarity of SPBSSL signal. 0: Active low SPBSSL signal 1: Active high SPBSSL signal															
3	CPOL	0	R/W	SPBSSL Negation Period SPBCLK Output Direction Sets the SPBCLK output direction during SPBSSL negation period. 0: SPBCLK output is 0 during SPBSSL negation period. 1: SPBCLK output is 1 during SPBSSL negation period.															
2	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.															

Bit	Bit Name	Initial Value	R/W	Description
1, 0	BSZ[1:0]	00	R/W	Data Bus Size Specifies the number of serial flash memories to be connected. 00: 1 memory 01: 2 memories 1X: Setting prohibited Note: After changing (the value of) this bit, all the entries in the read cache must be cleared by setting the RCF bit in DRCCR to 1.

### 17.4.2 SSL Delay Register (SSLDR)

SSLDR is a 32-bit register that adjusts the timing between the SPBSSL signal and the SPBCLK signal.

The settings of this register are reflected both in external address space read mode and SPI operating mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	SPNDL[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	SLNDL[2:0]			-	-	-	-	-	SCKDL[2:0]		
Initial value:	0	0	0	0	0	1	1	1	0	0	0	0	0	1	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
18 to 16	SPNDL[2:0]	111	R/W	Next Access Delay Sets the period from transfer end to next transfer start (next access). 000: 1 SPBCLK cycle 001: 2 SPBCLK cycles 010: 3 SPBCLK cycles 011: 4 SPBCLK cycles 100: 5 SPBCLK cycles 101: 6 SPBCLK cycles 110: 7 SPBCLK cycles 111: 8 SPBCLK cycles
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 8	SLNDL[2:0]	111	R/W	SPBSSL Negation Delay Sets the period from the time the last SPBCLK edge is sent of a transfer to SPBSSL pin negation (SPBSSL negation delay). 000: 1.5 SPBCLK cycles 001: 2.5 SPBCLK cycles 010: 3.5 SPBCLK cycles 011: 4.5 SPBCLK cycles 100: 5.5 SPBCLK cycles 101: 6.5 SPBCLK cycles 110: 7.5 SPBCLK cycles 111: 8.5 SPBCLK cycles
7 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
2 to 0	SCKDL[2:0]	111	R/W	Clock Delay Sets the period from SPBSSL pin assertion to SPBCLK oscillation (clock delay). 000: 1 SPBCLK cycle 001: 2 SPBCLK cycles 010: 3 SPBCLK cycles 011: 4 SPBCLK cycles 100: 5 SPBCLK cycles 101: 6 SPBCLK cycles 110: 7 SPBCLK cycles 111: 8 SPBCLK cycles

### 17.4.3 Bit Rate Register (SPBCR)

SPBCR is a 32-bit register that sets the bit rate.

The settings of this register are reflected both in external address space read mode and SPI operating mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SPBR[7:0]								-	-	-	-	-	-	BRDV[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 8	SPBR[7:0]	All 0	R/W	Bit Rate Sets the bit rate. The bit rate is determined by a combination of these bits with the BRDV[1:0] bits. For details, see Table 17.3, Relationship between SPBR[7:0] and BRDV[1:0] Settings.
7 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	BRDV[1:0]	11	R/W	Bit Rate Frequency Division Sets the bit rate. The bit rate is determined by a combination of these bits with the SPBR[7:0] bits. The SPBR value is used to set the base bit rate. The BRDV value is used to select a division ratio of the base bit rate from among no division, 2, 4, and 8. 00: Base bit rate 01: Base bit rate divided by 2 10: Base bit rate divided by 4 11: Base bit rate divided by 8

**(1) Bit Rate**

SPBR[7:0] and BRDV[1:0] are used for setting the bit rate.

The following formula is used to calculate the bit rate when SPBR[7:0]  $\neq$  0.

$$\text{Bit rate} = B\phi / (2 \times n \times 2^N)$$

n: SPBR[7:0] setting (1, ..., 255)  
N: BRDV[1:0] setting (0 to 3)

Also the following formula is used to calculate the bit rate when SPBR[7:0] = 0.

$$\text{Bit rate} = B\phi / 2^N$$

N: BRDV[1:0] setting (0 to 3)

**Table 17.3 Relationship between SPBR[7:0] and BRDV[1:0] Settings**

SPBR[7:0] (n)	BRDV[1:0] (N)	Division Ratio	Bit Rate		
			B $\phi$ = 128 MHz	B $\phi$ = 128 MHz	B $\phi$ = 133.33 MHz
0	0	1	Setting prohibited		
1	0	2	50 Mbps	64 Mbps	66.67 Mbps
2	0	4	25 Mbps	32 Mbps	33.33 Mbps
3	0	6	16.67 Mbps	21.33 Mbps	22.22 Mbps
4	0	8	12.5 Mbps	16 Mbps	16.67 Mbps
5	0	10	10 Mbps	12.8 Mbps	13.33 Mbps
6	0	12	8.33 Mbps	10.67 Mbps	11.11 Mbps
6	1	24	4.17 Mbps	5.33 Mbps	5.56 Mbps
6	2	48	2.08 Mbps	2.67 Mbps	2.78 Mbps
6	3	96	1.04 Mbps	1.33 Mbps	1.39 Mbps
255	3	4080	324.51 Kbps	31.37 Kbps	32.68 Kbps

Note: • The bit rate should be set so that it will satisfy the AC characteristics of this module.



### 17.4.4 Data Read Control Register (DRCR)

DRCR is a 32-bit register that sets the operation in external address space read mode.

The bits except the SSLN bit should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	SSLN	-	-	-	-	RBURST[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	W	R	R	R	R	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	RCF	RBE	-	-	-	-	-	-	-	SSLE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	W	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	SSLN	0	W	SPBSSL Negation Asserted SPBSSL can be negated by writing 1 to this bit when both the RBE and SSLE bits are 1. This bit is always read as 0. Note: To start next access after SPBSSL negation using this bit, read SSLF in CMNSR = 0 to confirm that the SPBSSL has been negated.
23 to 20	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
19 to 16	RBURST [3:0]	0000	R/W	Read Data Burst Length Sets the burst length (data unit count) when reading. This bit is enabled when the RBE bit is set to 1. 0000: 1 data unit 0001: 2 continuous data units : 1110: 15 continuous data units 1111: 16 continuous data units One data unit is 64 bits long.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	RCF	0	W	Read Cache Flush When 1 is written to this bit, all the entries in the read cache are cleared. This bit is always read as 0. Note: After flushing the read cache by writing 1 to the RCF bit, read the DRCR before proceeding to read from the external address space.
8	RBE	0	R/W	Read Burst Turns burst ON or OFF when reading. 0: Data is read according to the access size. 1: Read cache is enabled, and as many data units as the burst count specified in RBURST[3:0] bits is read.
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SSLE	0	R/W	SPBSSL Negation Sets the conditions for SPBSSL negation during read burst. SPBSSL is negated for each access during normal read. 0: SPBSSL is negated after transfer of data set in burst length. 1: SPBSSL is negated when the accessed address is not continuous with the previously transferred address.

### 17.4.5 Data Read Command Setting Register (DRCMR)

DRCMR is a 32-bit register that sets the commands issued in external address space read mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	CMD[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	OCMD[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	CMD[7:0]	H'03	R/W	Command Sets the command.
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	OCMD[7:0]	H'00	R/W	Optional Command Sets the optional command.

### 17.4.6 Data Read Extended Address Setting Register (DREAR)

DREAR is a 32-bit register that sets the address when the serial flash address is output in 32-bit mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

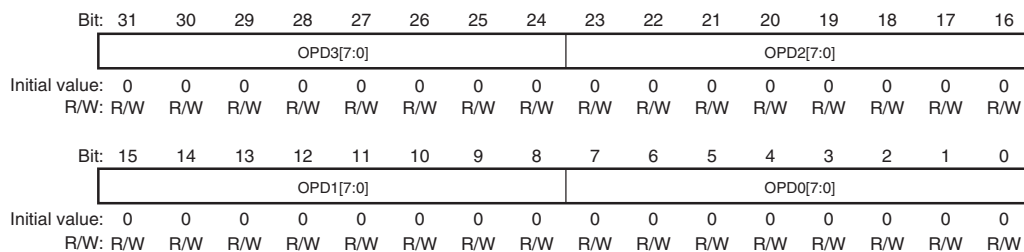
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	EAV[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	EAC[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	EAV[7:0]	H'00	R/W	32-Bit Extended Upper Address Fixed Value Sets the upper address bit values of the external address specified by the EAC[2:0] bits when the serial flash address is output in 32-bit mode. Bit 0 corresponds to the serial flash address bit [25], and bit 7 corresponds to the bit [32]. This setting is valid when the ADE[3] bit in DRENr is 1. When EAC[2:0] are 000, serial flash address [32:25] fixed values should be set to EAV[7:0]. When EAC[2:0] are 001, serial flash address [32:26] fixed values should set to EAV[7:1]. (1) When BSZ[1:0] in CMNCR = 00 (one serial flash memory connected) Serial flash addresses [31:0] are used for accessing. (2) When BSZ[1:0] in CMNCR = 01 (two serial flash memories connected) Serial flash addresses [32:1] are used for accessing.
15 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	EAC[2:0]	000	R/W	32-Bit Extended External Address Valid Range Sets the range of the external address to be used as serial flash address when the serial flash address is output in 32-bit mode. This setting is valid when the ADE[3] bit in DRENr is 1. 000: External address bits [24:0] enabled 001: External address bits [25:0] enabled Other than above: Setting prohibited

### 17.4.7 Data Read Option Setting Register (DROPR)

DROPR is a 32-bit register that sets the option data in external address space read mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	OPD3[7:0]	H'00	R/W	Option Data 3 Sets the option data 3.
23 to 16	OPD2[7:0]	H'00	R/W	Option Data 2 Sets the option data 2.
15 to 8	OPD1[7:0]	H'00	R/W	Option Data 1 Sets the option data 1.
7 to 0	OPD0[7:0]	H'00	R/W	Option Data 0 Sets the option data 0.

Note: • OPD3, OPD2, OPD1, and OPD0 are output in this order.

### 17.4.8 Data Read Enable Setting Register (DRENr)

DRENr is a 32-bit register that sets the bit size of the command, optional command, address, option data, and read data in external address space read mode and enables outputting them other than read data.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CDB[1:0]		OCDB[1:0]		-	-	ADB[1:0]		-	-	OPDB[1:0]		-	-	DRDB[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DME	CDE	-	OCDE	ADE[3:0]				OPDE[3:0]				-	-	-	-
Initial value:	0	1	0	0	0	1	1	1	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	CDB[1:0]	00	R/W	Command Bit Size Sets the command size in bit units. 00: 1 bit 01: 2 bits 10: 4 bits 11: Setting prohibited
29, 28	OCDB[1:0]	00	R/W	Optional Command Bit Size Sets the optional command size in bit units. 00: 1 bit 01: 2 bits 10: 4 bits 11: Setting prohibited
27, 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25, 24	ADB[1:0]	00	R/W	Address Bit Size Sets the address size in bit units. 00: 1 bit 01: 2 bits 10: 4 bits 11: Setting prohibited
23, 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
21, 20	OPDB[1:0]	00	R/W	Option Data Bit Size Sets the option data size in bit units. 00: 1 bit 01: 2 bits 10: 4 bits 11: Setting prohibited
19, 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17, 16	DRDB[1:0]	00	R/W	Data Read Bit Size Sets the data read size in bit units. 00: 1 bit 01: 2 bits 10: 4 bits 11: Setting prohibited
15	DME	0	R/W	Dummy Cycle Enable Enables insertion of the dummy cycle before the read data. Note: A setting is prohibited for a transfer starting with a dummy cycle. 0: Dummy cycle insertion disabled 1: Dummy cycle insertion enabled

Bit	Bit Name	Initial Value	R/W	Description
14	CDE	1	R/W	Command Enable Sets the command to be output. 0: Command output disabled 1: Command output enabled
13	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
12	OCDE	0	R/W	Optional Command Enable Sets the optional command to be output. 0: Optional command output disabled 1: Optional command output enabled
11 to 8	ADE[3:0]	0111	R/W	Address Enable Sets the address to be output. Be sure to use the following setting; otherwise, the operation is not guaranteed. (1) BSZ[1:0] in CMNCR = 00 (one serial flash memory connected) 0000: Output disabled 0111: Address[23:0] 1111: Address[31:0] Other than above: Setting prohibited (2) BSZ[1:0] in CMNCR = 01 (two serial flash memories connected) 0000: Output disabled 0111: Address[24:1] 1111: Address[32:1] Other than above: Setting prohibited
7 to 4	OPDE[3:0]	0000	R/W	Option Data Enable Sets the option data to be output. Use only the settings given below. Otherwise, the operation cannot be guaranteed. 0000: Output disabled 1000: OPD3 1100: OPD3, OPD2 1110: OPD3, OPD2, OPD1 1111: OPD3, OPD2, OPD1, OPD0 Other than above: Setting prohibited
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

### 17.4.9 SPI Mode Control Register (SMCR)

SMCR is a 32-bit register that sets the operation in SPI operating mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	SSLKP	-	-	-	-	-	SPIRE	SPIWE	SPIE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R/W	R/W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	SSLKP	0	R/W	SPBSSL Signal Level Determines the SPBSSL status after the end of transfer. 0: SPBSSL signal is negated at the end of transfer. 1: SPBSSL signal level is maintained from the end of transfer to the start of next access.
7 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	SPIRE	0	R/W	Data Read Enable Enables reading in SPI operating mode. 0: Data reading disabled 1: Data reading enabled Note: When the transfer data bit size is set to 2 bits or 4 bits with the SPIDB[1:0] bits, the SPIRE and SPIWE bits should not be set to 1 at the same time.
1	SPIWE	0	R/W	Data Write Enable Enables writing in SPI operating mode. 0: Data writing disabled 1: Data writing enabled Note: When the transfer data bit size is set to 2 bits or 4 bits with the SPIDB[1:0] bits, the SPIRE and SPIWE bits should not be set to 1 at the same time.
0	SPIE	0	W	SPI Data Transfer Enable Data is transferred by setting this bit to 1. This bit is enabled only when the TEND bit in CMNSR is set to 1. The operation cannot be guaranteed when this bit is set to 1 with the TEND bit set to 0. This bit is always read as 0. Note: When the SPBSSL pin is de-asserted, the command, optional command, address, and option data that are output enabled are output even if the SPIRE and SPIWE bits are set to 0. When the SPBSSL pin is asserted, follow the notes described in section 17.6.2, Notes on Starting Transfer from the SPBSSL Retained State in SPI Operating Mode.

### 17.4.10 SPI Mode Command Setting Register (SMCMR)

SMCMR is a 32-bit register that sets the commands issued in SPI operating mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	CMD[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	OCMD[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	CMD[7:0]	H'00	R/W	Command Sets the command.
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	OCMD[7:0]	H'00	R/W	Optional Command Sets the optional command.

### 17.4.11 SPI Mode Address Setting Register (SMADR)

SMADR is a 32-bit register that sets the addresses in SPI operating mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ADR[31:24]								ADR[23:16]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADR[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	ADR[31:24]	H'00	R/W	Address Sets the value of bits 31 to 24 when the serial flash address is output in 32-bit units. This setting is valid when ADE[3] in SMENR is 1.
23 to 0	ADR[23:0]	H'000000	R/W	Address Sets the address.



### 17.4.12 SPI Mode Option Setting Register (SMOPR)

SMOPR is a 32-bit register that sets the option data in SPI operating mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OPD3[7:0]								OPD2[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OPD1[7:0]								OPD0[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	OPD3[7:0]	H'00	R/W	Option Data 3 Sets the option data 3.
23 to 16	OPD2[7:0]	H'00	R/W	Option Data 2 Sets the option data 2.
15 to 8	OPD1[7:0]	H'00	R/W	Option Data 1 Sets the option data 1.
7 to 0	OPD0[7:0]	H'00	R/W	Option Data 0 Sets the option data 0.

Note: • OPD3, OPD2, OPD1, and OPD0 are output in this order.

### 17.4.13 SPI Mode Enable Setting Register (SMENR)

SMENR is a 32-bit register that sets the bit size of the command, optional command, address, option data, and transfer data in SPI operating mode and enables their output. SMENR also enables dummy cycle insertion. Disabling all of the command, optional command, address, option data, dummy cycle, and transfer data is prohibited. At least one of them except dummy cycle must be enabled.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit: 31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	
CDB[1:0]		OCDB[1:0]		-	-	ADB[1:0]		-	-	OPDB[1:0]		-	-	SPIDB[1:0]		
Initial value: 0		0		0	0	0		0	0	0		0	0	0		
R/W: R/W		R/W		R/W	R/W	R		R	R	R/W		R/W	R	R	R/W	R/W
Bit: 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
DME	CDE	-	OCDE	ADE[3:0]				OPDE[3:0]				SPIDE[3:0]				
Initial value: 0		1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W: R/W		R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	CDB[1:0]	00	R/W	Command Bit Size Sets the command size in bit units. 00: 1 bit 01: 2 bits 10: 4 bits 11: Setting prohibited
29, 28	OCDB[1:0]	00	R/W	Optional Command Bit Size Sets the optional command size in bit units. 00: 1 bit 01: 2 bits 10: 4 bits 11: Setting prohibited
27, 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25, 24	ADB[1:0]	00	R/W	Address Bit Size Sets the address size in bit units. 00: 1 bit 01: 2 bits 10: 4 bits 11: Setting prohibited
23, 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
21, 20	OPDB[1:0]	00	R/W	Option Data Bit Size Sets the option data size in bit units. 00: 1 bit 01: 2 bits 10: 4 bits 11: Setting prohibited
19, 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17, 16	SPIDB[1:0]	00	R/W	Transfer Data Bit Size Sets the transfer data size in bit units. 00: 1 bit 01: 2 bits 10: 4 bits 11: Setting prohibited

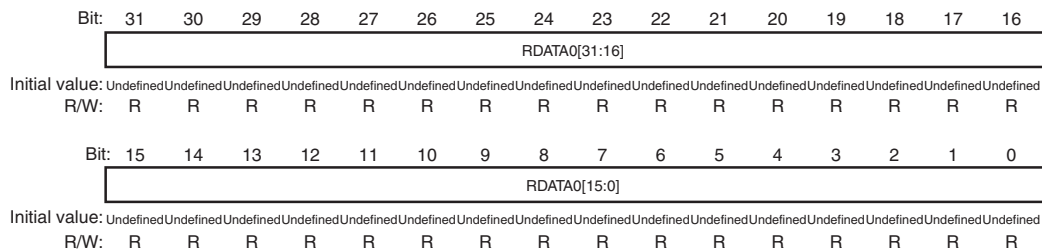
Bit	Bit Name	Initial Value	R/W	Description
15	DME	0	R/W	Dummy Cycle Enable Enables insertion of the dummy cycle before the read data. Note: Dummy cycle insertion is prohibited for write in SPI operating mode including the case in which a transfer ends with a dummy cycle. Note: A setting is prohibited for a transfer starting with a dummy cycle. 0: Dummy cycle insertion disabled 1: Dummy cycle insertion enabled
14	CDE	1	R/W	Command Enable Sets the command to be output. 0: Command output disabled 1: Command output enabled
13	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
12	OCDE	0	R/W	Optional Command Enable Sets the optional command to be output. 0: Optional command output disabled 1: Optional command output enabled
11 to 8	ADE[3:0]	0000	R/W	Address Enable Sets the address to be output. Use only the settings given below. Otherwise, the operation cannot be guaranteed. 0000: Output disabled 0100: ADR[23:16] 0110: ADR[23:8] 0111: ADR[23:0] 1111: ADR[31:0] Other than above: Setting prohibited
7 to 4	OPDE[3:0]	0000	R/W	Option Data Enable Sets the option data to be output. Use only the settings given below. Otherwise, the operation cannot be guaranteed. 0000: Output disabled 1000: OPD3 1100: OPD3, OPD2 1110: OPD3, OPD2, OPD1 1111: OPD3, OPD2, OPD1, OPD0 Other than above: Setting prohibited
3 to 0	SPIDE[3:0]	0000	R/W	Transfer Data Enable Sets valid transfer data. Valid data differs depending on the BSZ[1:0] bit setting in CMNCR. The following settings must be used. Otherwise, the operation is not guaranteed. (1) BSZ[1:0] bits in CMNCR = 00 (one serial flash memory connected) 0000: Not transferred 1000: 8 bits transferred (enables data at address 0 of the SPI mode read/write data registers 0) 1100: 16 bits transferred (enables data at addresses 0 and 1 of the SPI mode read/write data registers 0 ) 1111: 32 bits transferred (enables data at addresses 0 to 3 of the SPI mode read/write data registers 0) Other than above: Setting prohibited (2) BSZ[1:0] bits in CMNCR = 01 (two serial flash memories connected) 0000: Not transferred 1000: 16 bits transferred (enables data at addresses 0 and 1 of the SPI mode read/write data registers 0 ) 1100: 32 bits transferred (enables data at addresses 0 to 3 of the SPI mode read/write data registers 0) 1111: 64 bits transferred (enables data at addresses 0 to 3 of the SPI mode read/write data registers 0 and data at addresses 0 to 3 of the SPI mode read/write data registers 1) Other than above: Setting prohibited

### 17.4.14 SPI Mode Read Data Register 0 (SMRDR0)

SMRDR0 is a 32-bit register that stores the read data in SPI operating mode.

Access to this register should be performed in the same size as the transfer size specified in the SPIDE[3:0] bits in the SPI mode enable setting register (SMENR). Be sure to access from address 0.

The settings of this register should be read when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	RDATA0 [31:0]	Undefined	R	Read Data Holds the data read in SPI operating mode. Data bits differ depending on the BSZ[1:0] bit setting in CMNCR. BSZ[1:0] = 00: Read data[31:0]. BSZ[1:0] = 01: Read data[63:32].

Note: • The contents of this register and SMRDR1 are modified upon completion of reception in SPI operating mode. Be sure to read data when reception in SPI operating mode is completed.

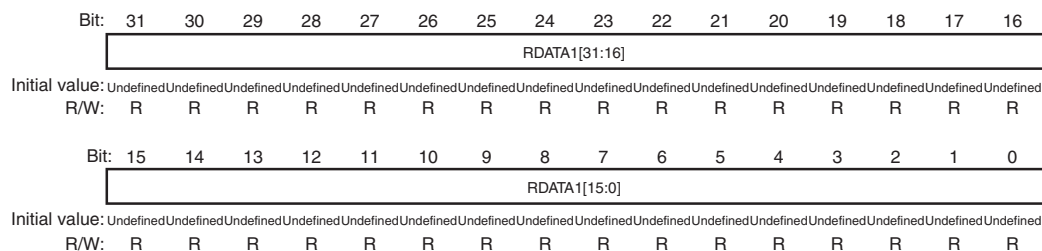
### 17.4.15 SPI Mode Read Data Register 1 (SMRDR1)

SMRDR1 is a 32-bit register that stores the read data in SPI operating mode.

This register is enabled when the BSZ[1:0] bits in CMNCR are set to 01 (two serial flash memories connected) and disabled when the BSZ[1:0] bits in CMNCR are set to 00 (one serial flash memory connected).

Access to this register should be performed in the same size as the transfer size specified in the SPIDE[3:0] bits in the SPI mode enable setting register (SMENR). Be sure to access from address 0.

The settings of this register should be read when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.



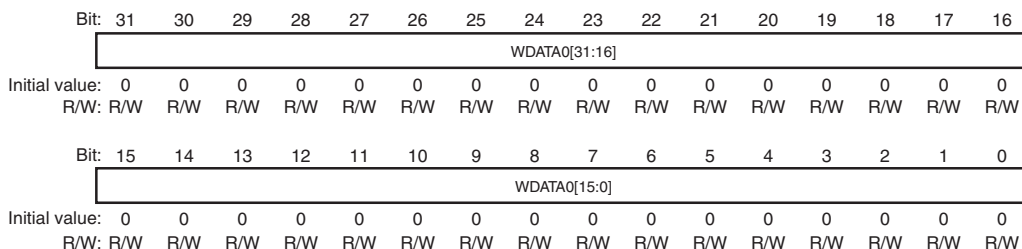
Bit	Bit Name	Initial Value	R/W	Description
31 to 0	RDATA1 [31:0]	Undefined	R	Read Data Holds the data read in SPI operating mode. Data bits differ depending on the BSZ[1:0] bit setting in CMNCR. BSZ[1:0] = 00: Bits in this register are disabled. BSZ[1:0] = 01: Read data[31:0]

### 17.4.16 SPI Mode Write Data Register 0 (SMWDR0)

SMWDR0 is a 32-bit register that sets the write data in SPI operating mode.

Access to this register should be performed in the same size as the transfer size specified in the SPIDE[3:0] bits in the SPI mode enable setting register (SMENR). Be sure to access from address 0.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	WDATA0 [31:0]	All 0	R/W	Write Data Holds the data to be written in SPI operating mode. Data bits differ depending on the BSZ[1:0] bit setting in CMNCR. BSZ[1:0] = 00: Write data[31:0]. BSZ[1:0] = 01: Write data[63:32].

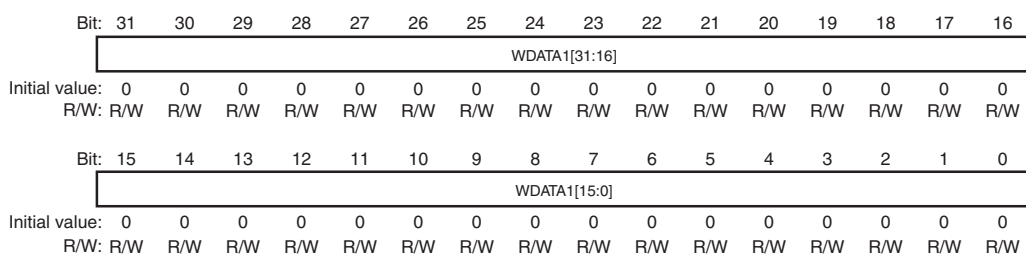
### 17.4.17 SPI Mode Write Data Register 1 (SMWDR1)

SMWDR1 is a 32-bit register that sets the write data in SPI operating mode.

This register is enabled when the BSZ[1:0] bits in CMNCR are set to 01 (two serial flash memories connected) and disabled when the BSZ[1:0] bits in CMNCR are set to 00 (one serial flash memory connected).

Access to this register should be performed in the same size as the transfer size specified in the SPIDE[3:0] bits in the SPI mode enable setting register (SMENR). Be sure to access from address 0.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	WDATA1 [31:0]	All 0	R/W	Write Data Holds the data to be written in SPI operating mode. Data bits differ depending on the BSZ[1:0] bit setting in CMNCR. BSZ[1:0] = 00: Bits in this register are disabled. BSZ[1:0] = 01: Write data[31:0].

### 17.4.18 Common Status Register (CMNSR)

CMNSR is a 32-bit register that holds flags indicating the operating state.

The settings of this register are reflected both in external address space read mode and SPI operating mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	SSLF	TEND
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	SSLF	0	R	SPBSSL Pin Monitor 0: SPBSSL pin is negated 1: SPBSSL pin is asserted
0	TEND	1	R	Transfer End Flag Indicates whether the data transfer has ended. 0: Indicates that data transfer is in progress 1: Indicates that data transfer has ended

### 17.4.19 SPI AC Input Characteristics Adjustment Register (CKDLY)

CKDLY is used to adjust the timing of the setup and hold times for data input. The timing should be adjusted to suit the AC characteristics of the serial flash memory to be connected. Settings of this register should be changed while the SSLF flag in CMNSR is 0; otherwise, the operation cannot be guaranteed. When writing, write to the register as a 32-bit unit with bits 15 to 8 set to H'A5.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GB[7:0]							-	-	-	-	CKDLY[3:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 8	GB[7:0]	All 0	R/W	Guard When writing, write to the register as a 32-bit unit with these bits set to H'A5. These bits are always read as 0.
7 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	CKDLY[3:0]	0100	R/W	Input Characteristics Adjustment Switches the relative timing of the setup and hold times for data input. The two values below are specifiable. 0100: Initial value 1000: Makes the data input setup time shorter and the data input hold time longer. Other than the above: Settings prohibited

### 17.4.20 Data Read Dummy Cycle Setting Register (DRDMCR)

DRDMCR is a 32-bit register that sets the size and number of dummy cycles to be inserted in external address space read mode.

The settings of this register are enabled when the DME bit in the data read enable setting register (DRENr) is 1.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	DMDB[1:0]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	DMCYC[2:0]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17, 16	DMDB [1:0]	00	R/W	Dummy Cycle Bit Size Sets the dummy cycle size in bit units. The setting of these bits is combined with the setting of the IO0FV, IO2FV, and IO3FV bits in the common control register (CMNCR) to determine the state of the unused pins during the dummy cycles. The state of the used pins is Hi-Z. 00: 1 bit 01: 2 bits 10: 4 bits 11: Setting prohibited
15 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	DMCYC [2:0]	000	R/W	Number of Dummy Cycles Sets the number of dummy cycles to be inserted when the DME bit in the data read enable setting register (DRENr) is 1. 000: 1 cycle 001: 2 cycles 010: 3 cycles 011: 4 cycles 100: 5 cycles 101: 6 cycles 110: 7 cycles 111: 8 cycles



### 17.4.21 Data Read DDR Enable Register (DRDRENr)

DRDRENr is a 32-bit register that sets SDR or DDR transfer of the address, option data, and read data in external address space read mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	ADDRE	-	-	-	OPDRE	-	-	-	DRDRD
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	ADDRE	0	R/W	Address DDR Enable Sets SDR or DDR transfer of the address. 0: SDR transfer 1: DDR transfer
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	OPDRE	0	R/W	Option Data DDR Enable Sets SDR or DDR transfer of the option data. 0: SDR transfer 1: DDR transfer
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	DRDRD	0	R/W	Data Read DDR Enable Sets SDR or DDR transfer of the read data. 0: SDR transfer 1: DDR transfer

### 17.4.22 SPI Mode Dummy Cycle Setting Register (SMDMCR)

SMDMCR is a 32-bit register that sets the size and number of dummy cycles to be inserted in SPI operating mode.

The settings of this register are enabled when the DME bit in the SPI mode enable setting register (SMENR) is 1.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	DMDB[1:0]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	DMCYC[2:0]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17 and 16	DMDB [1:0]	00	R/W	Dummy Cycle Bit Size Sets the dummy cycle size in bit units. The setting of these bits is combined with the setting of the IO0FV, IO2FV, and IO3FV bits in the common control register (CMNCR) to determine the state of the unused pins during the dummy cycles. The state of the used pins is Hi-Z. 00: 1 bit 01: 2 bits 10: 4 bits 11: Setting prohibited
15 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	DMCYC[2:0]	000	R/W	Number of Dummy Cycles Sets the number of dummy cycles to be inserted when the DME bit in the SPI mode enable setting register (SMENR) is 1. 000: 1 cycle 001: 2 cycles 010: 3 cycles 011: 4 cycles 100: 5 cycles 101: 6 cycles 110: 7 cycles 111: 8 cycles

### 17.4.23 SPI Mode DDR Enable Register (SMDRENR)

SMDRENR is a 32-bit register that sets SDR or DDR transfer of the address, option data, and transfer data in SPI operating mode.

The settings of this register should be changed when the TEND flag in CMNSR is 1; otherwise, the operation cannot be guaranteed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	ADDRE	-	-	-	OPDRE	-	-	-	SPIDRE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	ADDRE	0	R/W	Address DDR Enable Sets SDR or DDR transfer of the address. 0: SDR transfer 1: DDR transfer
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	OPDRE	0	R/W	Option Data DDR Enable Sets SDR or DDR transfer of the option data. 0: SDR transfer 1: DDR transfer
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SPIDRE	0	R/W	Transfer Data DDR Enable Sets SDR or DDR transfer of the transfer data. 0: SDR transfer 1: DDR transfer

### 17.4.24 SPI AC Output Characteristics Adjustment Register (SPODLY)

SPODLY is used to adjust the timing of the delay, hold, buffer on and buffer off times for data output. The timing should be adjusted to suit the AC characteristics of the serial flash memory to be connected. Settings of this register should be changed while the SSLF flag in CMNSR is 0; otherwise, the operation cannot be guaranteed. When writing, write to the register as a 32-bit unit with bits 31 to 24 set to H'A5.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GB[7:0]								-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SPODLY[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

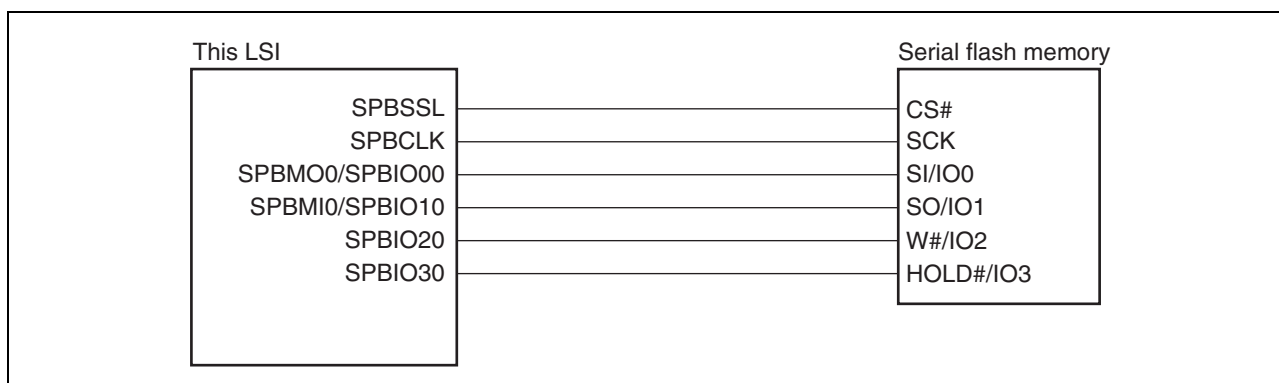
Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GB[7:0]	All 0	R/W	Guard When writing, write to the register as a 32-bit unit with these bits set to H'A5. These bits are always read as 0.
23 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	SPODLY[15:0]	All 0	R/W	Output Characteristics Adjustment Switches the timing of the delay, hold, buffer on and buffer off times for data output. The two values below are specifiable. H'0000: Initial value H'6363: The delay, hold, buffer on and buffer off times for data output are lengthened. Other than the above: Settings prohibited

## 17.5 Operation

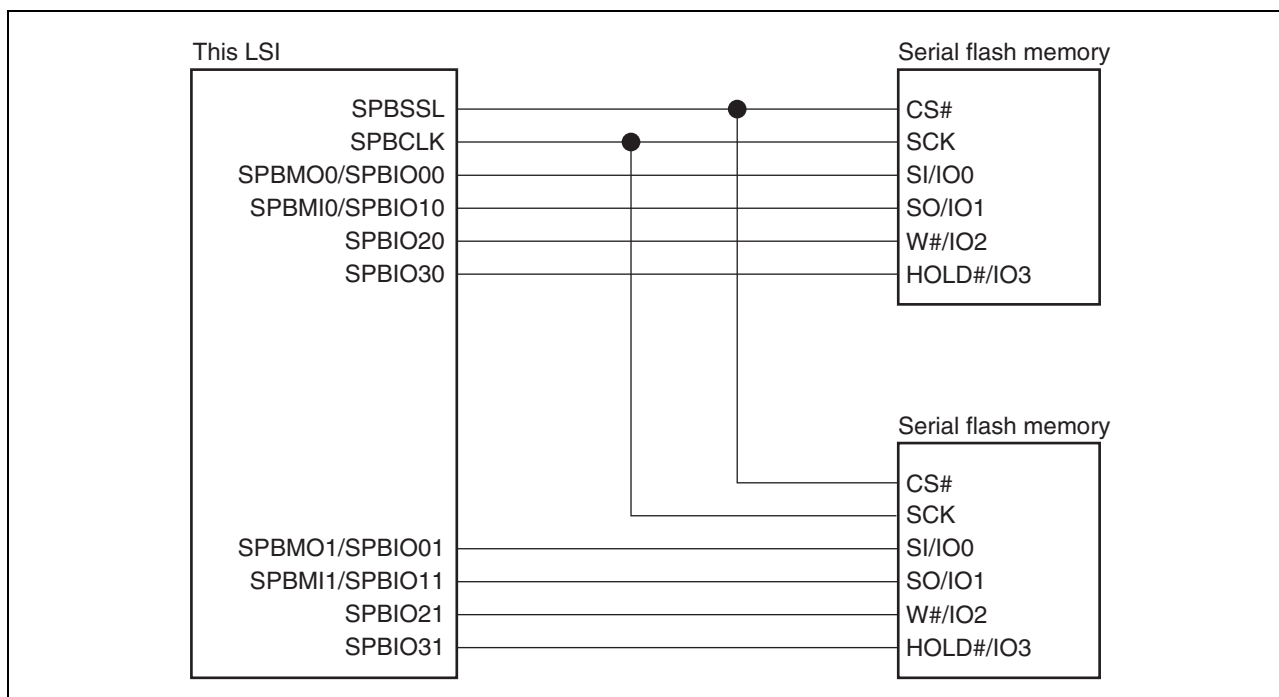
### 17.5.1 System Configuration

With this module, one or two serial flash memories can be directly connected per channel (data size of 1, 2, and 4 bits). The number of connected memories can be selected using the BSZ[1:0] bits in CMNCR.

Examples of system configuration with one serial flash memory connected and two serial flash memories connected are shown in Figure 17.2 and Figure 17.3, respectively.



**Figure 17.2** System Configuration Example with 4-Bit Data Size and One Serial Flash Memory Connected (BSZ[1:0] Bits in CMNCR = 00)



**Figure 17.3** System Configuration Example with 4-Bit Data Size and Two Serial Flash Memories Connected (BSZ[1:0] Bits in CMNCR = 01)

### 17.5.2 Address Map

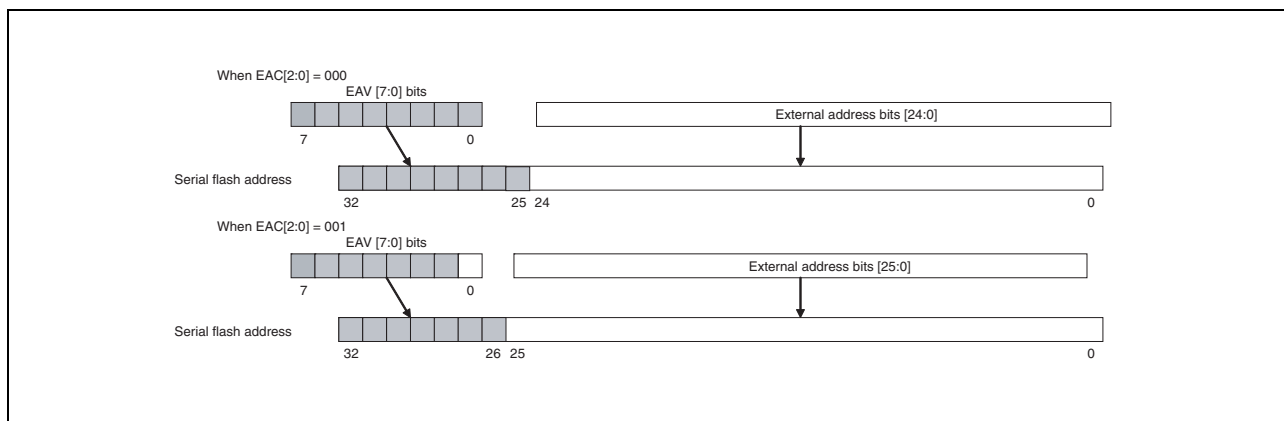
In external address space read mode, the serial flash connected is assigned in the SPI multi I/O bus space. A maximum accessible address space differs depending on the number of serial flash memories connected. In combination with DREAR, a maximum of 4 Gbytes can be accessed when one serial flash memory is connected, and a maximum of 8 Gbytes can be accessed when two memories are connected.

**Table 17.4 Address Map**

Channel	Number of Serial Flash Memories Connected	Internal Address	Max. Access Area
0	1	H'18000000 to H'1BFFFFFF H'58000000 to H'5BFFFFFF (mirror area)	4 Gbytes
	2	H'18000000 to H'1BFFFFFF H'58000000 to H'5BFFFFFF (mirror area)	8 Gbytes
1	1	H'1C000000 to H'1FFFFFFF H'5C000000 to H'5FFFFFFF (mirror area)	4 Gbytes
	2	H'1C000000 to H'1FFFFFFF H'5C000000 to H'5FFFFFFF (mirror area)	8 Gbytes

### 17.5.3 32-bit Serial Flash Addresses

Since the SPI multi I/O bus space is 64 Mbytes, only a part of the 32-bit serial flash address area can be directly accessed. Here, the fixed value set in the pertinent register is used as the upper bit value of a 32-bit address. To output serial flash addresses in 32 bits, set the ADE[3] bit in DRENr to 1, set the range of the external addresses used as the serial flash addresses to the EAC[2:0] bits in DREAR, and set the upper bit value of the 32-bit address as the fixed value to the EAV[7:0] bits in DREAR.



**Figure 17.4 32-Bit Address Setting**

Setting the ADE[3] bit in DRENr to 1 allows the serial flash address to be output using [31:0] bits. When EAC[2:0] = 000, external address bits [24:0] are valid; set the value for [32:25] bits to EAV[7:0]. When EAC[2:0] = 001, external address bits [25:0] are valid; set the value for [32:26] bits to EAV[7:1].

The address bits actually used for access depend on the number of serial flash memories connected. When one serial flash memory is connected, address bits [31:0] are used and when two memories are connected, address bits [32:1] are used.

Note: • When the capacity of the serial flash memory used is smaller than 4 Gbytes, keep the following point in mind. If an access spreads over the last address of the serial flash in burst mode (RBE bit in DRECR = 1), the access address does not agree with the internal address of the serial flash. To prevent this, software should appropriately manage the accessible address areas for the serial flash memory used according to the memory capacity.

### 17.5.4 Data Alignment

Data alignment can be set by using the SFDE bit in the common control register (CMNCR). Data alignment in data read mode and in SPI mode are shown in Figure 17.5 and Figure 17.6, respectively.

When two serial flash memories are connected, the serial flash memory connected to the pin SPBIO30-SPBIO00 has the address 2n and the serial flash memory connected to the pin SPBIO31-SPBIO01 has the address 2n + 1. The data should be accessed in word or larger units. It cannot be accessed in byte units.

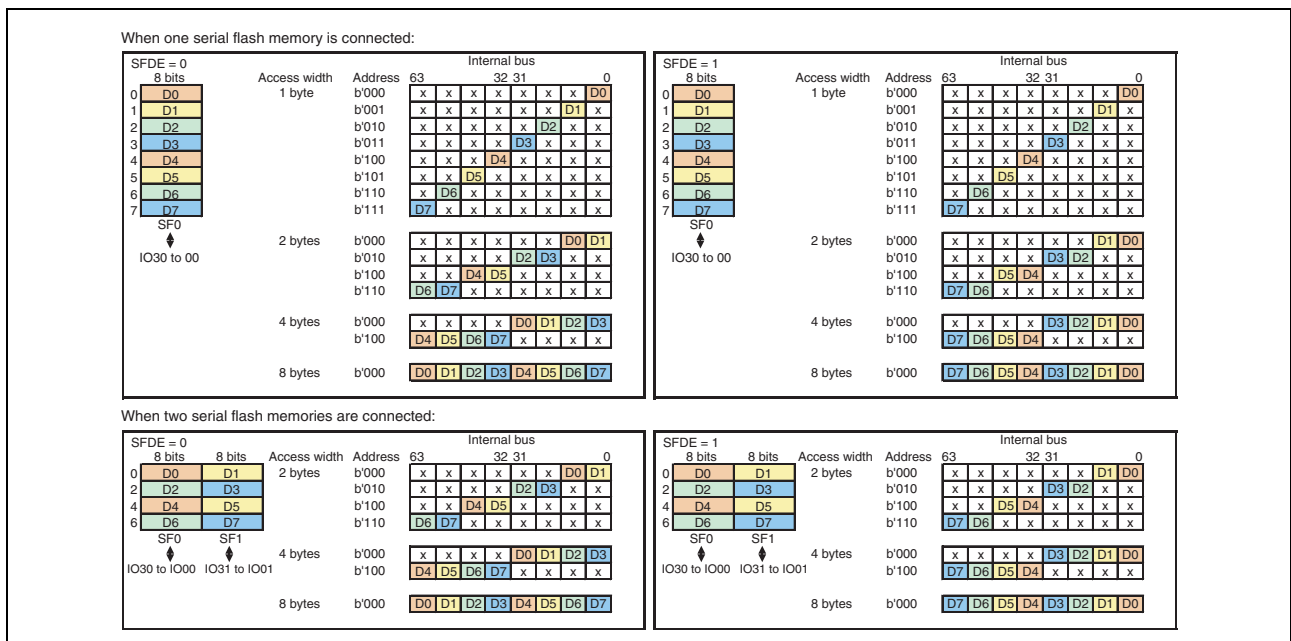


Figure 17.5 Data Alignment in Data Read Mode

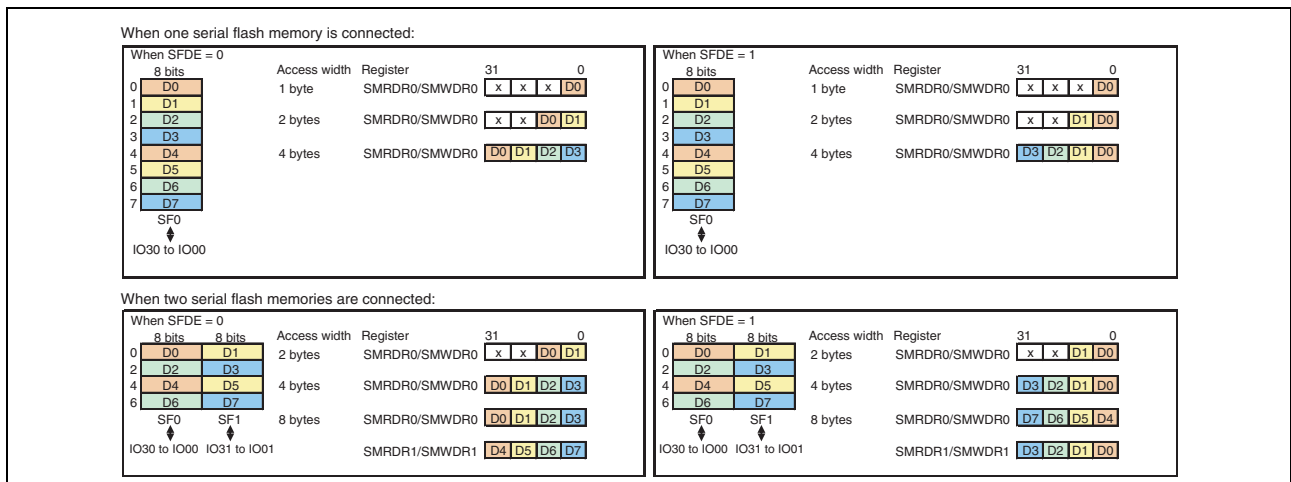


Figure 17.6 Data Alignment in SPI Mode

### 17.5.5 Operating Modes

This module has two operating modes: external address space read mode and SPI operating mode.

In external address space read mode, a read access to the SPI multi I/O bus space is converted into SPI communication and data is received. After data acquisition, data is returned to the bus master that is the issuing source. For details, see section 17.5.6, External Address Space Read Mode.

In SPI operating mode, arbitrary SPI communication is carried out using register settings. For details, see section 17.5.8, SPI Operating Mode.

### 17.5.6 External Address Space Read Mode

A read access to the SPI multi I/O bus space can be converted into SPI communication in external address space read mode. Further, the commands, optional commands, option data, and dummy cycle issued for reading can be modified using registers. For the address, option data, and read data, either SDR or DDR transfer can be selected using the appropriate register when the SPBCLK frequency division ratio is two or larger.

In external address space read mode, either normal read operation or burst read operation can be selected. The transfer format is determined based on the common control register (CMNCR), SSL delay register (SSLDR), bit rate setting register (SPBCR), data read control register (DRCCR), data read command setting register (DRCMR), data read extended address setting register (DREAR), data read option setting register (DROPR), data read enable setting register (DRENR), data read dummy cycle setting register (DRDMCR), and data read DDR enable register (DRDRENR).

#### (1) Normal Read Operation

When the RBE bit in DRCCR is set to 0, normal read operation is performed.

In the normal read operation, the data of 8 bits, 16 bits, 32 bits, and 64 bits are read for respectively a byte, a word, and a longword read access. Here, a byte access is enabled only when one serial flash memory is connected. After reading, the SPBSSL pin is negated.

The normal read operation timing is shown in Figure 17.7.

$t_1$  is the time period from SPBSSL pin assertion to SPBCLK oscillation (clock delay),  $t_2$  is the time period from transmission of the last SPBCLK edge of a transfer to SPBSSL pin negation (SPBSSL negation delay), and  $t_3$  is the time period from one transfer end to the next transfer start (next access). For details of  $t_1$ ,  $t_2$ , and  $t_3$ , see section 17.5.9, Transfer Format.

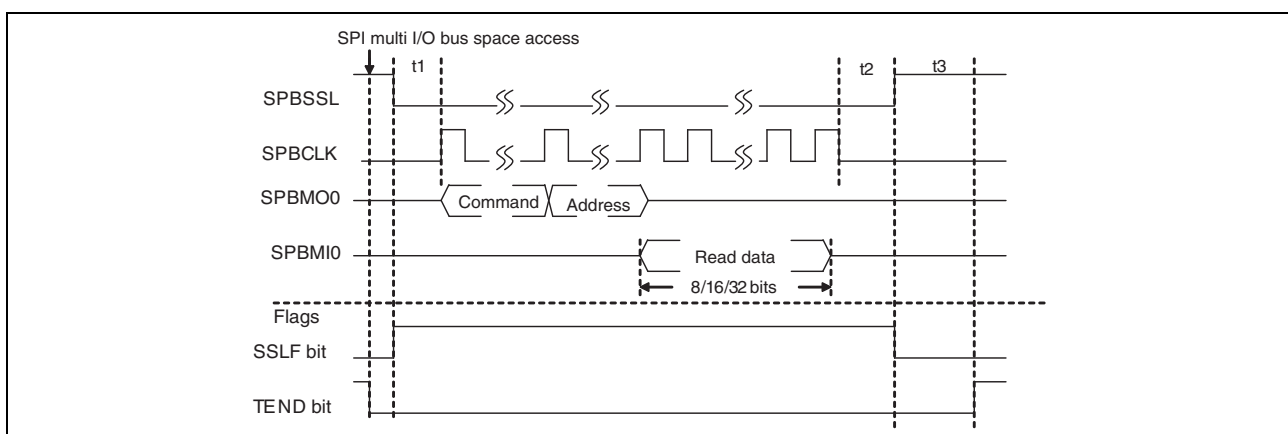


Figure 17.7 Normal Read Operation Timing



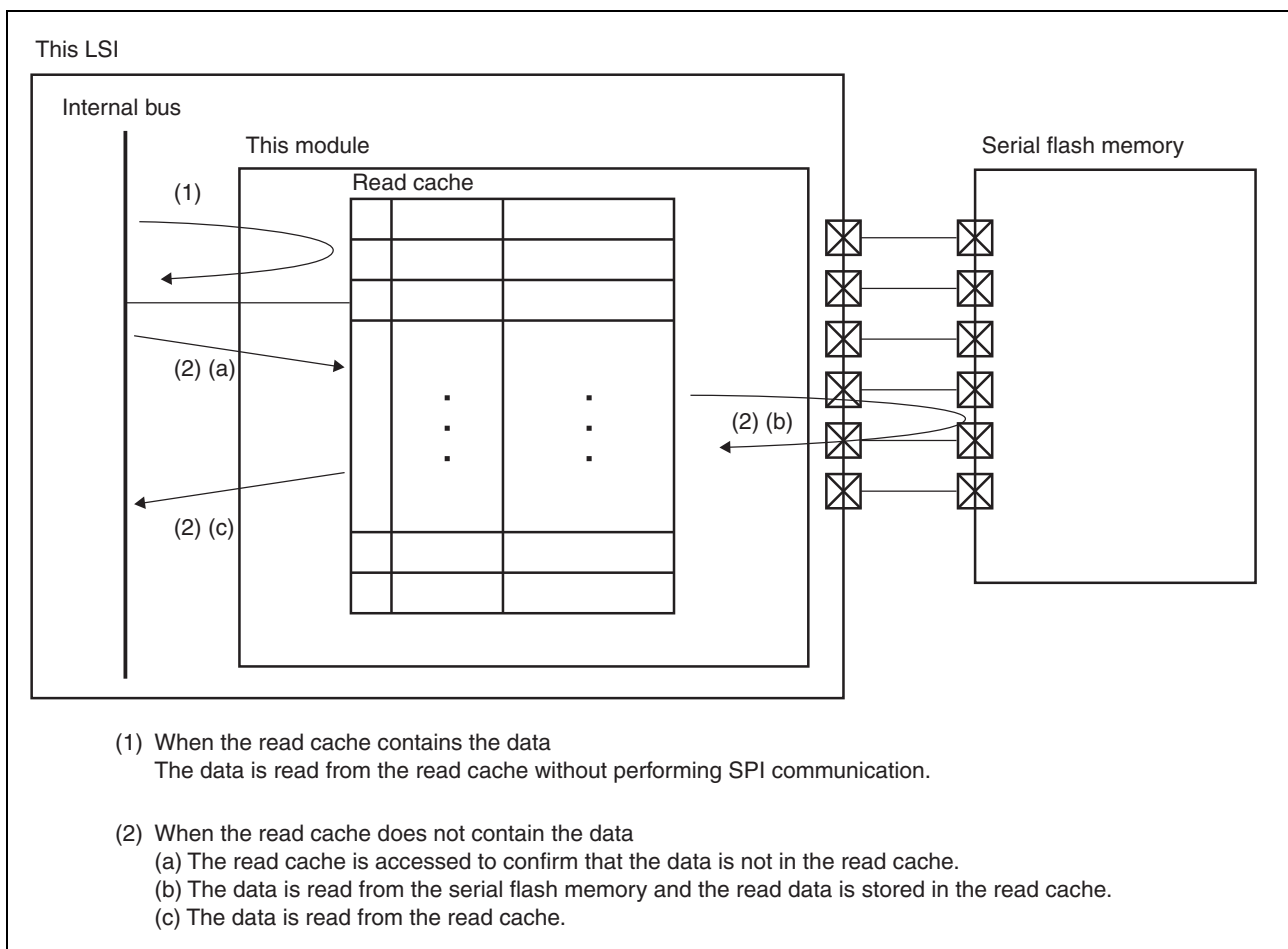
## (2) Burst Read Operation

When the RBE bit in DRCR is set to 1, burst read operation is performed.

Read cache is enabled in the burst read operation. For read cache operation, see section 17.5.7, Read Cache.

For reading bytes, words, or longwords, the read cache is first referred to for the data. When the read cache contains the data, the data is read from the read cache without accessing the serial flash memory. When the read cache does not contain the data, burst read operation is performed in the serial flash memory and the read data is stored in the read cache. The data transfer length at that time is  $64 \text{ bits} \times \text{RBURST}[3:0]$  bits and the data is always read from the 64-bit boundary. The SPBSSL pin status after data transfer can be selected by using the SSLE bit in DRCR. When the SSLE bit is set to 0, the SPBSSL pin is negated after data transfer. For an operation performed when the SSLE bit is set to 1, see (3) Burst Read Operation with Automatic SPBSSL Negation, just below.

A pattern diagram of this operation and a burst read operation timing diagram when SSLE bit is set to 0 are shown in Figure 17.8 and Figure 17.9.



**Figure 17.8 Burst Read Operation**

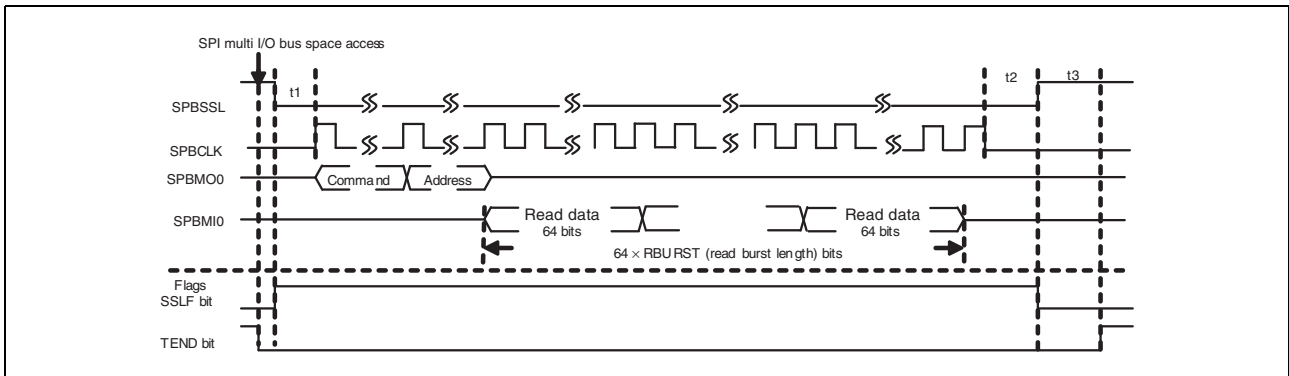


Figure 17.9 Burst Read Operation Timing (SSLE Bit = 0)

(3) Burst Read Operation with Automatic SPBSSL Negation

When SSLE bit in DRCR is set to 1, this module does not negate the SPBSSL pin after the burst read transfer. When accessing the next time, if the address is continuous with the previous read address, the burst read operation is performed without issuing the command, optional command, address, option data, or dummy cycle. If the address is not continuous with the previous read address, the SPBSSL pin is once negated and the burst read operation is performed after issuing the command, optional command, address, option data, or dummy cycle.

Burst read timing diagrams for continuous address and non-continuous address are shown in Figure 17.10 and Figure 17.11.

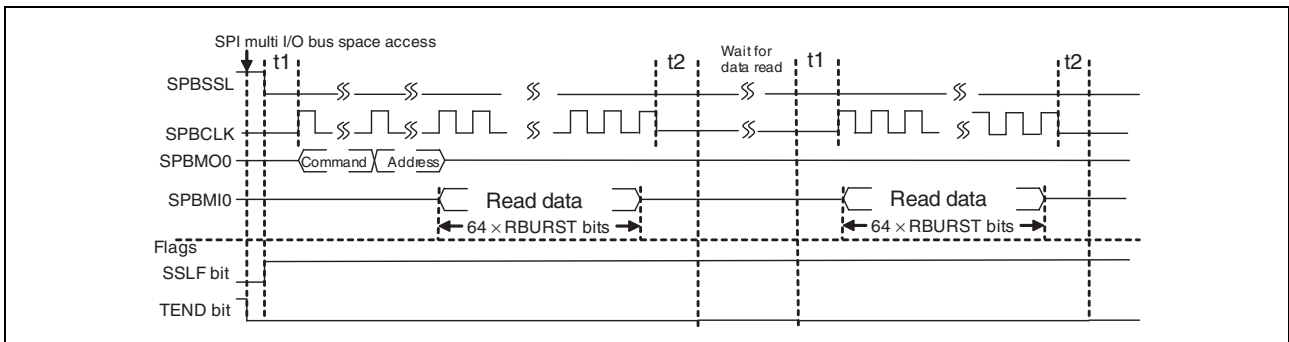


Figure 17.10 Burst Read Timing for Continuous Address (SSLE Bit = 1)

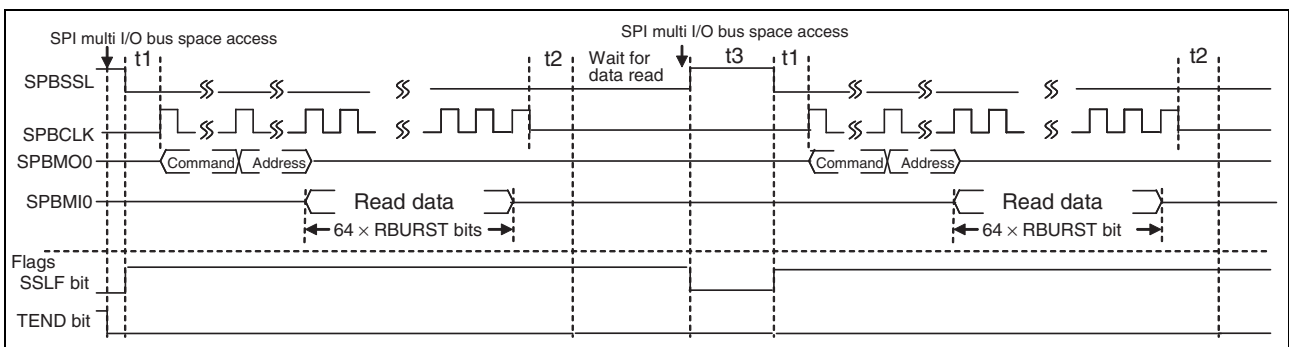
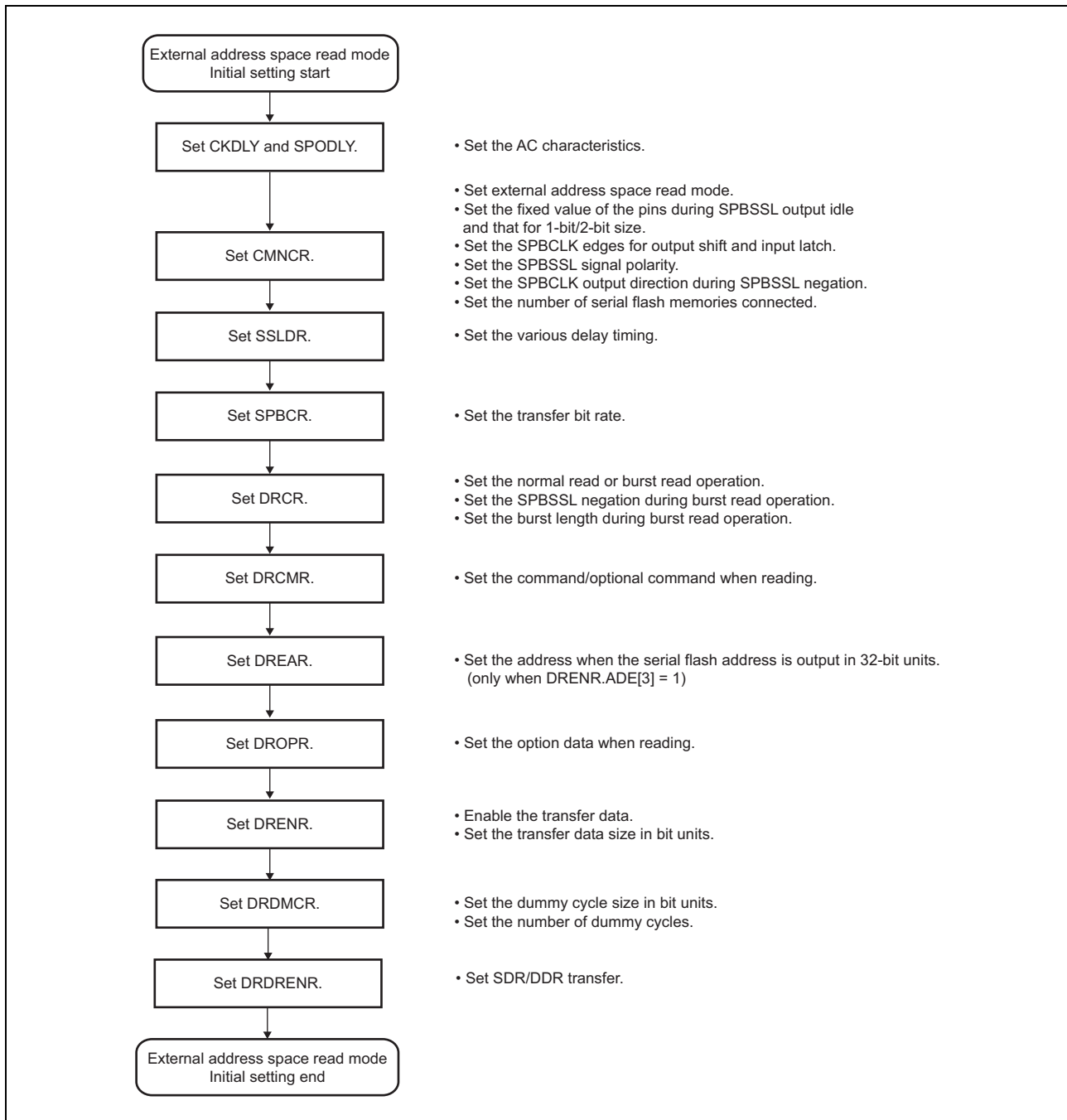


Figure 17.11 Burst Read Timing for Non-Continuous Address (SSLE Bit = 1)

For the next access after negation of the SPBSSL with the SSLN bit in DRCR with this operation, read SSLF = 0 in CMNSR to confirm that the SPBSSL has been negated.

#### (4) Initial Setting Flow

An example of an initial setting flow in external address space read mode is shown in Figure 17.12.

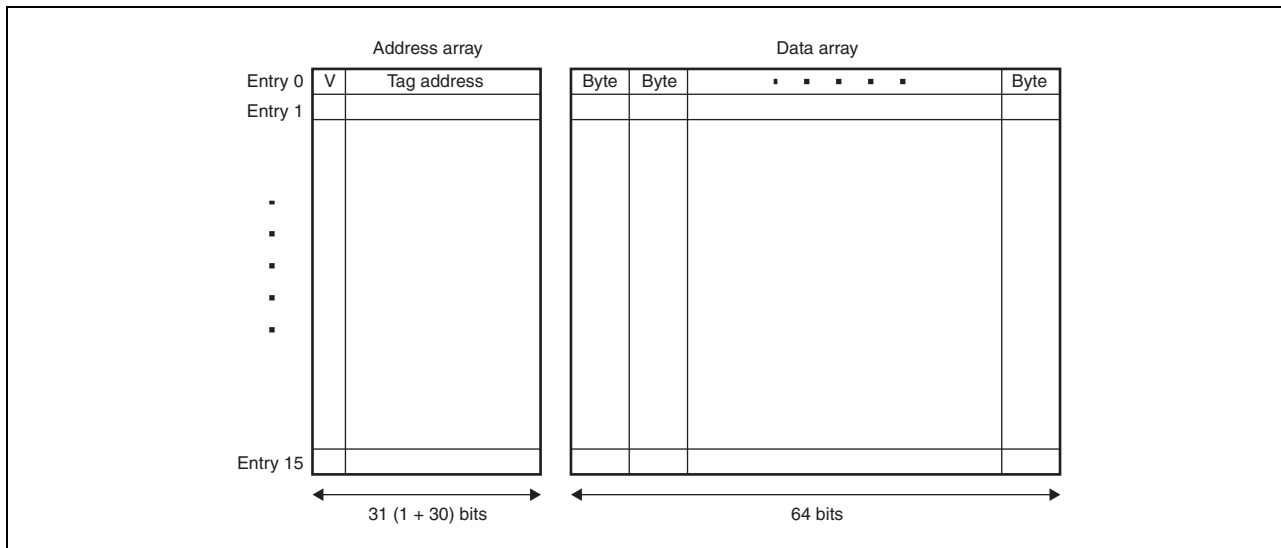


**Figure 17.12 Example of Initial Setting Flow in External Address Space Read Mode**

### 17.5.7 Read Cache

This module has a simple built-in read cache. The read cache can be used during external address space read mode and burst read operation. The read cache is configured with a line size of 64 bits and 16 entries.

Read cache configuration is shown in Figure 17.13.



**Figure 17.13 Read Cache Configuration**

#### (1) Address Array

The V bit indicates whether the entry data is valid. When the V bit is 1, the data is valid and when V bit is 0, the data is invalid.

The tag address bits hold the address used for the serial flash memory. Address bits 32 to 3 are used for the purpose. Address bits 23 to 3 are enabled when address output is 24 bits and one serial flash memory is connected; and address bits 24 to 3 are enabled when two serial flash memories are connected.

Address bits 31 to 3 are enabled when address output is 32 bits and one serial flash memory is connected; and address bits 32 to 3 are enabled when two serial flash memories are connected.

#### (2) Data Array

It retains the 64-bit read data. Registration in the read cache is performed in line units.

#### (3) Read Operation

In case of read-hit, data is read from the read cache. In case of read-miss, after the  $64 \times \text{RBURST}$  (read burst length) data is read from the serial flash memory and the read cache is updated, the data is returned to the bus master.

#### (4) Data Replacement

The write pointer is used to update data. In case of read-miss, the RBURST (read burst length) portion data is replaced starting at the entry specified by the write pointer. In other words, the data is replaced in the storage order of the data. Whether data is referred to or not will not affect the replacement order of data.

### 17.5.8 SPI Operating Mode

This module can carry out an arbitrary SPI operation by using the register settings.

The transfer format is determined based on the common control register (CMNCR), SSL delay register (SSLDR), bit rate setting register (SPBCR), SPI mode control register (SMCR), SPI mode command setting register (SMCMR), SPI mode address setting register (SMADR), SPI mode option setting register (SMOPR), and SPI mode enable setting register (SMENR), SPI mode read data register (SMRDR), SPI mode write data register (SMWDR), SPI mode dummy cycle setting register (SMDMCR), and SPI mode DDR enable register (SMDRENR). For the address, option data, and transfer data, either SDR or DDR transfer can be selected using the appropriate register when the SPBCLK frequency division ratio is two or larger.

SPI operating mode can be used for reading the status of the serial flash memory and writing to the serial flash memory. In this mode, one transfer refers to the operation from when the SPIE bit in SMCR is set to 1 to when the TEND bit is set to 1.

#### (1) Transfer Start

The transfer of data is started in the set transfer format by setting the SPIE bit in SMCR to 1. When write operation is enabled, the SPI mode write data register is transmitted to the serial flash memory. When read operation is enabled, data read from the serial flash memory is stored into the SPI mode read data register.

The SPI operation timing is shown in Figure 17.14.

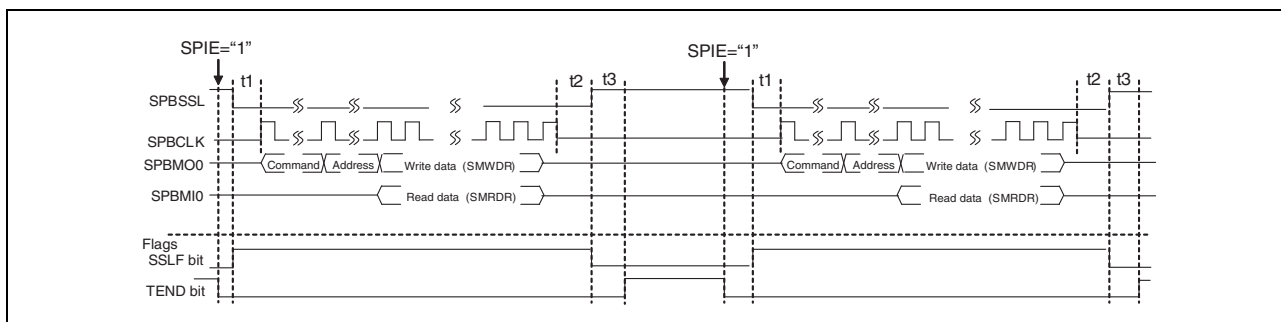


Figure 17.14 SPI Operation Timing

#### (2) Read/Write Enable

- Read operation: Data can be read by setting the SPIRE bit in SMCR to 1. The read data is stored into SMRDR.
- Write operation: Data can be written by setting the SPIWE bit in SMCR to 1. The data stored in SMWDR is output.

When the data size is set to 1 bit using the SPIDB[1:0] bits in SMENR, data can be transmitted and received by setting the SPIRE and SPIWE bits to 1. However, when the data size is set to 2 or 4 bits by using the SPIDB[1:0] bits, only one of the SPIRE and SPIWE bits should be enabled. The operation is not guaranteed if both the bits are enabled.

### (3) Retention of SPBSSL Pin Assertion

By setting the SSLKP bit in SMCR to 1, assertion of the SPBSSL pin can be continued till the next transfer. With this function, the transfer can be carried out continuously with the SPBSSL kept in the asserted state.

The data transfer timing using the SSLKP bit is shown in Figure 17.15.

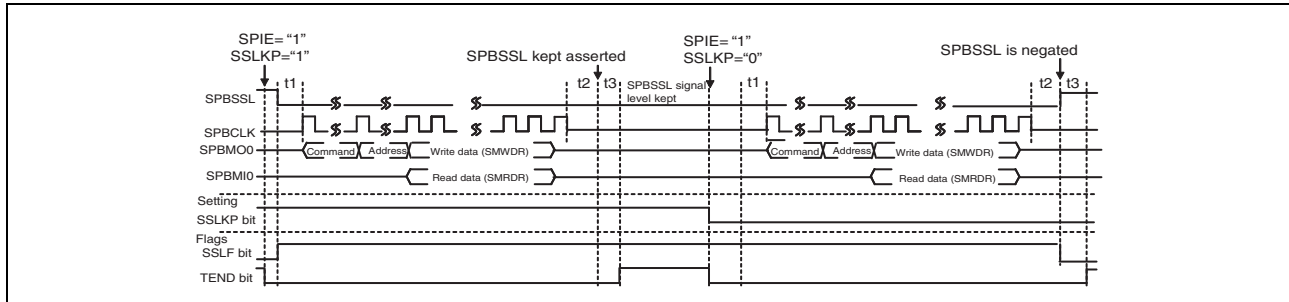


Figure 17.15 Data Transfer Timing using the SSLKP Bit

### (4) Initial Setting Flow

An example of an initial setting flow in SPI operating mode is shown in Figure 17.16.

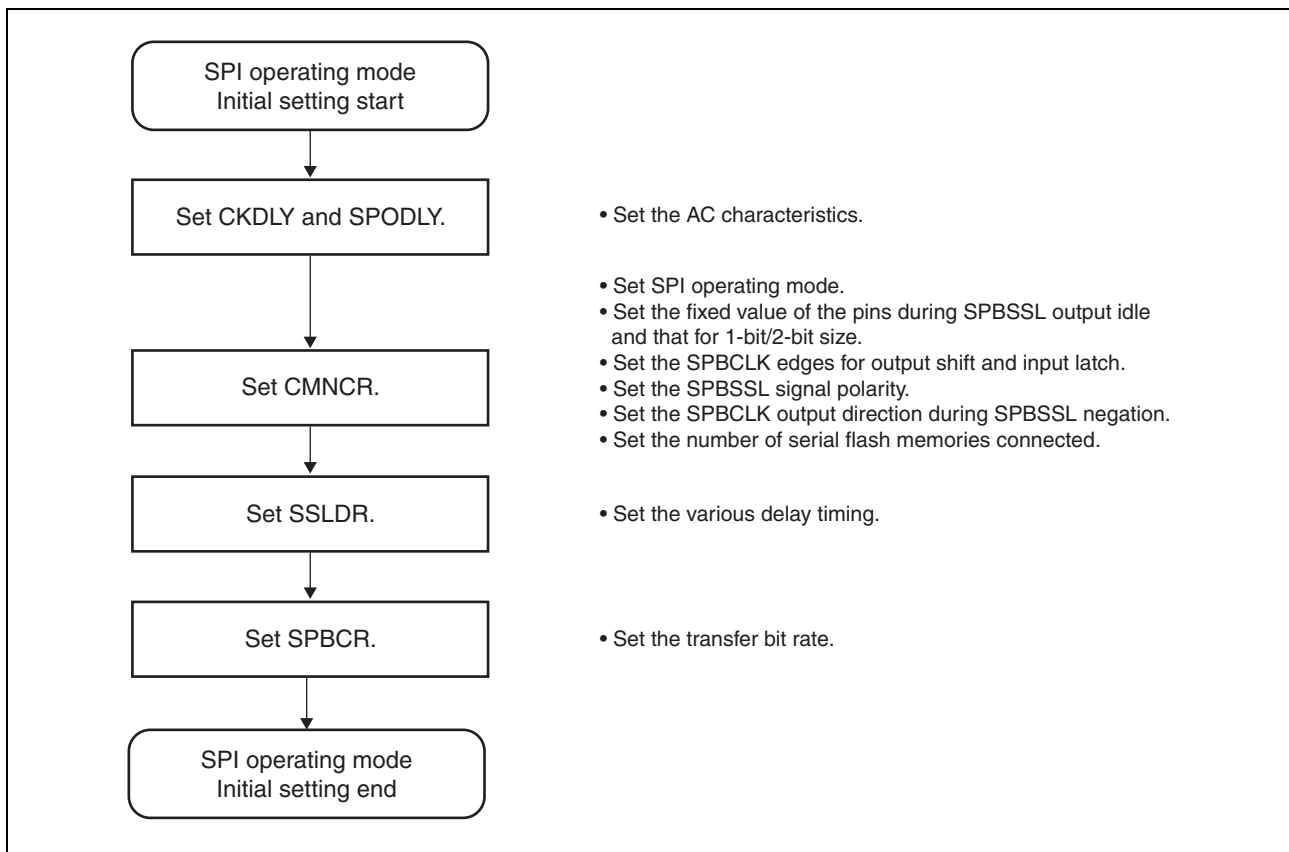


Figure 17.16 Example of Initial Setting Flow in SPI Operating Mode

## (5) Data Transfer Setting Flow

An example of a data transfer setting flow in SPI operating mode is shown in Figure 17.17.

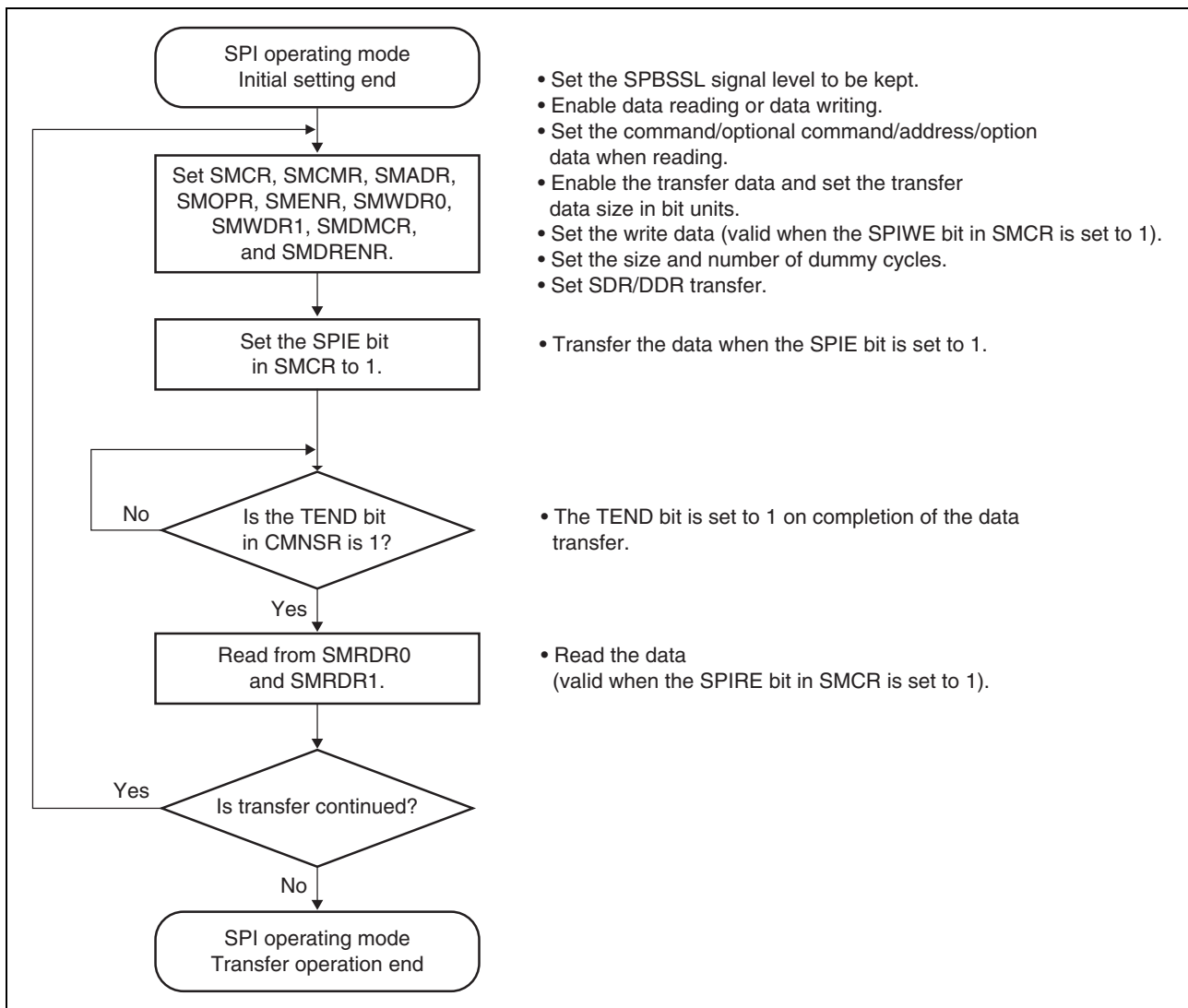


Figure 17.17 Example of a Data Transfer Setting Flow in SPI Operating Mode

### 17.5.9 Transfer Format

#### (1) SPBSSL Pin Enable Polarity Control

The enable polarity of the SPBSSL pin can be changed with the SSLP bit in CMNCR.

#### (2) SPBCLK Output

The SPBCLK output direction during SPBSSL negation can be set with the CPOL bit in CMNCR.

#### (3) Data Transmission and Reception Timing

Data transmission and reception timing is different between SDR transfer and DDR transfer.

During SDR transfer, data is transmitted and received at either the odd or even edges. The data transmission timing can be set to the odd or even edge with the CPHAT bit in CMNCR. Similarly, the data reception timing can be set to the odd or even edge with the CPHAR bit in CMNCR.

During DDR transfer, data is transmitted and received at both the odd and even edges. The first data transmission timing can be set to the odd or even edge with the CPHAT bit in CMNCR. Similarly, the first data reception timing can be set to the odd or even edge with the CPHAR bit in CMNCR.

#### (4) Delay Settings

$t_1$  is the time period from SPBSSL pin assertion to SPBCLK oscillation (clock delay). It can be set with the SCKDL[2:0] bits in SSLDR.  $t_2$  is the time period till the SPBSSL signal negation after the SPBCLK oscillation is stopped (SPBSSL negation delay). It can be set with the SLNDL[2:0] bits in SSLDR.  $t_3$  is the time period required to prevent SPBSSL signal assertion for the next transfer after the end of the previous transfer (next access delay). It can be set with the SPNDL[2:0] bits in SSLDR.

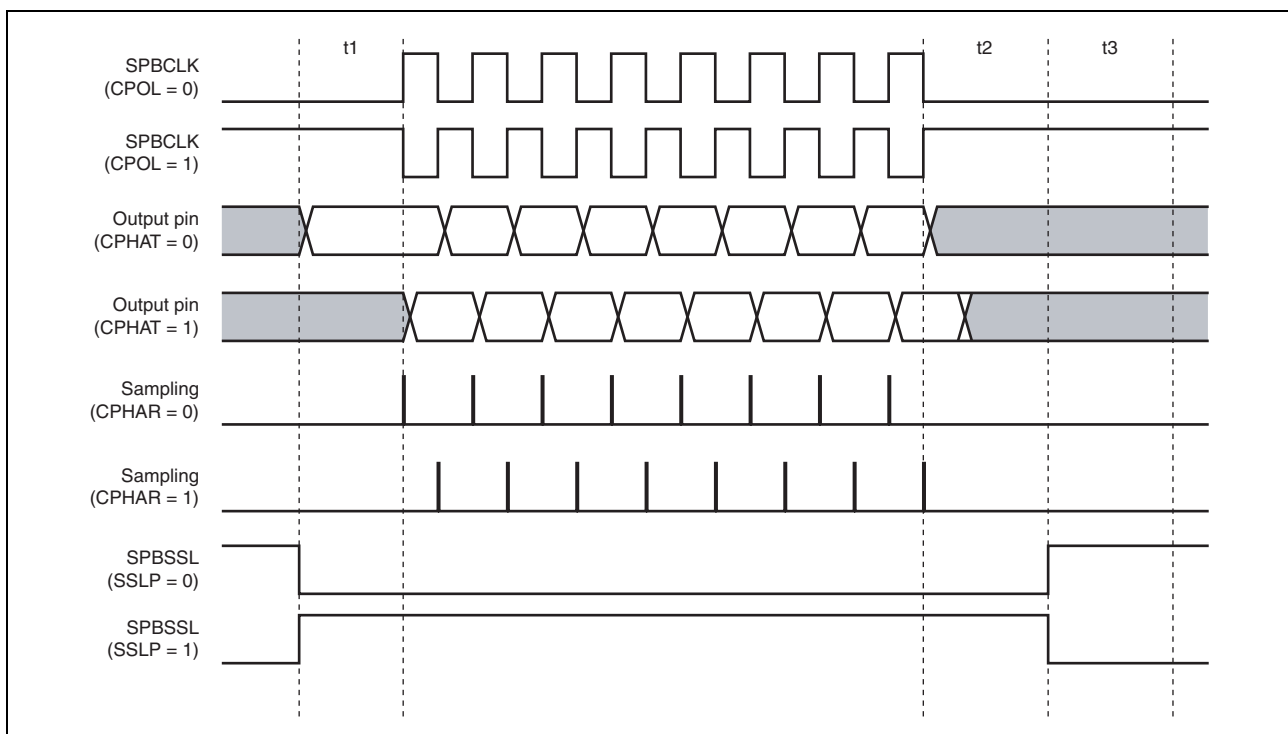
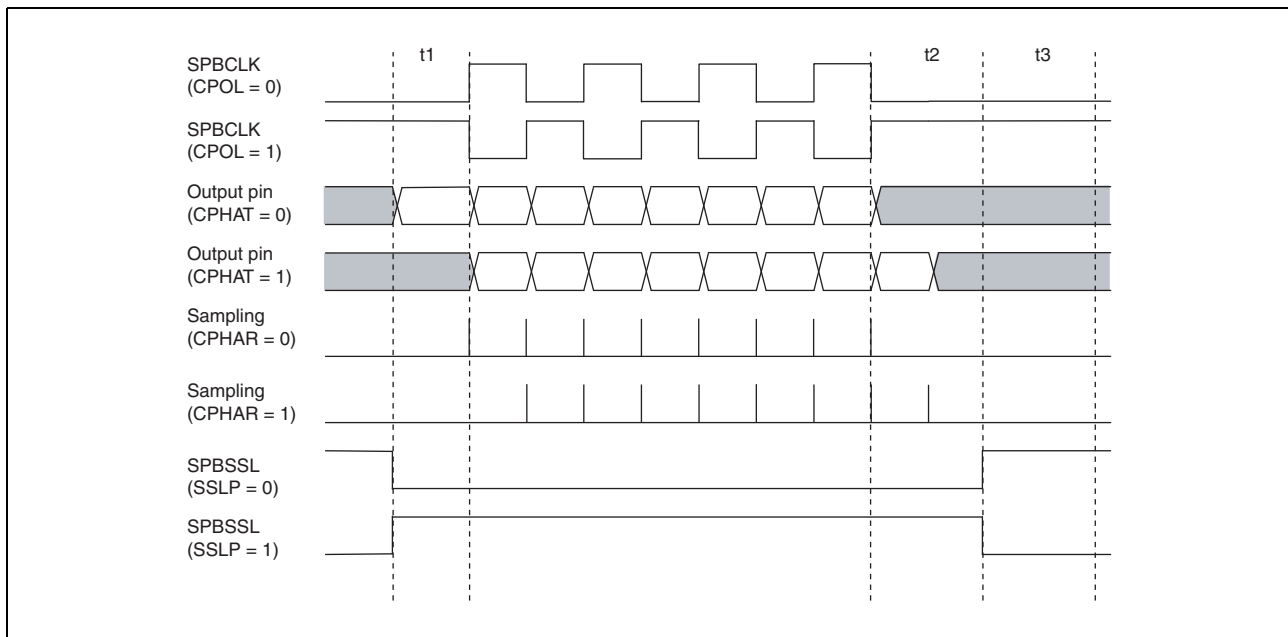


Figure 17.18 SDR Transfer Format





**Figure 17.19 DDR Transfer Format**

Note: • In DDR reception when CPHAR = 1, the sampling timing of the last bit is based on the frequency-divided clock in this module.

When CPHAT = 1 in DDR transfer, the serial flash memory cannot provide the sampling timing of the last bit; therefore, a transfer ending with a DDR transfer is not performed correctly when CPHAT = 1.

### 17.5.10 Data Format

This module can input and output data in the order of command, optional command, address, option data, dummy cycle, and data.

#### (1) Data Registers

Table 17.5 shows the input and output data.

**Table 17.5 Data Registers**

Data		External Address Space Read Operation	SPI Operation
Command (8 bits)		CMD[7:0] bits in DRCMR	CMD[7:0] bits in SMCMR
Optional command (8 bits)		OCMD[7:0] bits in DRCMR	OCMD[7:0] bits in SMCMR
Address (32/24 bits)	BSZ[1:0] = 00 (one flash memory connected)	32 bits: DREAR.EAV[6:1 to 0] bits + lower [25 to 24:0] bits of the read address. 24 bits: Lower [23:0] bits of the read address	32 bits: ADR[31:0] bits in SMADR 24 bits: ADR[23:0] bits in SMADR
	BSZ[1:0] = 01 (two flash memories connected)	32 bits: DREAR.EAV[7:1 to 0] bits + lower [25 to 24:1] bits of the read address. 24 bits: Lower [24:1] bits of the read address	
Option data (8 bits × 4)		DROPR	SMOPR
Dummy cycle (1 to 8 cycles)		DRDMC	SMCMC (only when read)
Transfer data		Normal read: 8, 16, and 32 bits Burst read: 64 × RBURST bits	Read: SMRDR0, SMRDR1 Write: SMWDR0, SMWDR1

(2) Data Enable

In external address space read mode, transfer enable or disable of the command, optional command, address, option data, and dummy cycle can be controlled with the CDE, OCDE, ADE[3:0], OPDE[3:0], and DME bits in DRENr, respectively. The size and number of dummy cycles can be controlled with the data read dummy cycle setting register (DRDMCR). When the SPBCLK frequency division ratio is two or larger, either SDR or DDR transfer can be selected for the address, option data, and read data, using the ADDRE, OPDRE, and DRDRE bits in the data read DDR enable register (DRDRENr).

Similarly, in SPI operating mode, enable or disable of the command, optional command, address, option data, dummy cycle, and transfer data can be controlled with the CDE, OCDE, ADE[3:0], OPDE[3:0], DME, and SPIDE[3:0] bits in SMENr, respectively. However, disabling all the above parameters is prohibited in SPI operating mode. At least one of them except dummy cycle must be enabled. The size and number of dummy cycles can be controlled with the SPI mode dummy cycle setting register (SMDMCR). When the SPBCLK frequency division ratio is two or larger, either SDR or DDR transfer can be selected for the address, option data, and transfer data, using the ADDRE, OPDRE, and SPIDRE bits in the SPI mode DDR enable register (SMDRENr).

For the address and option data in external address space read mode; and the address, option data, and transfer data in SPI operating mode, the enable bit setting allowed is determined according to the transfer data size. For the allowed setting combinations of the enable bits and transfer data size, refer to the description of the pertinent register.

If data is disabled, that data is skipped, and input and output of the next data is carried out. The command, optional command, address, and option data are always output. During dummy cycles, the state of the used pins is Hi-Z. In external address space read mode, data is always input; and in SPI operating mode, input and output of data is determined based on the settings of the SPIRE and SPIWE bits in SMCR.

There are some restrictions on dummy cycle insertion; refer to the description of the DME bits in DRENr and SMENr for details.

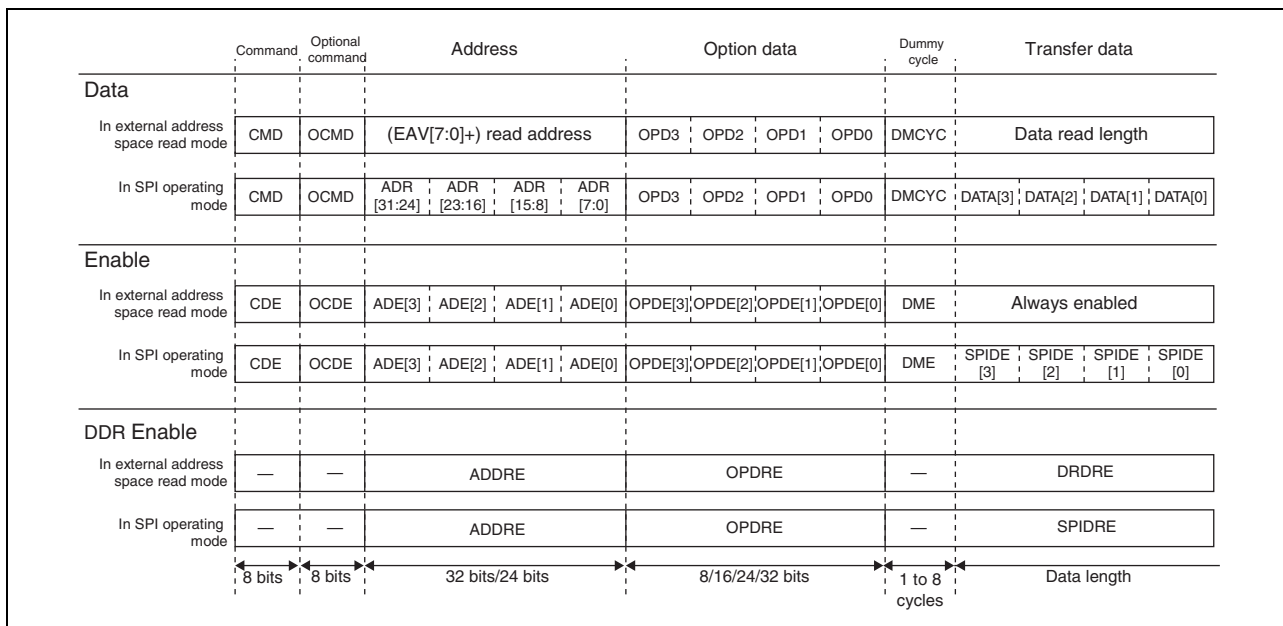


Figure 17.20 Data and Enable

(3) Bit Size

In external address space read mode, the size of the command, optional command, address, option data, and the read data in bit units is respectively controlled with the CDB[1:0], OCDB[1:0], ADB[1:0], OPDB[1:0], and DRDB[1:0] bits in DRENr. The size of the dummy cycle in bit units is also controlled with the DMDB[1:0] bits in DRDMCR.

Similarly, in SPI operating mode, the size of the command, optional command, address, option data, and read write data in bit units is controlled with the CDB[1:0], OCDB[1:0], ADB[1:0], OPDB[1:0], and SPIDB[1:0] bits in SMENr. The size of the dummy cycle in bit units is also controlled with the DMDB[1:0] bits in SMDMCR.

(a) 1-bit Size

When the size is set to 1 bit, SPBMI0 and SPBMI1 pins will be the input pins and SPBMO0 and SPBMO1 pins will be the output pins. SPBIO20, SPBIO21, SPBIO30, and SPBIO31 pins are not used.

Figure 17.21 and Figure 17.22 show the transfer format examples.

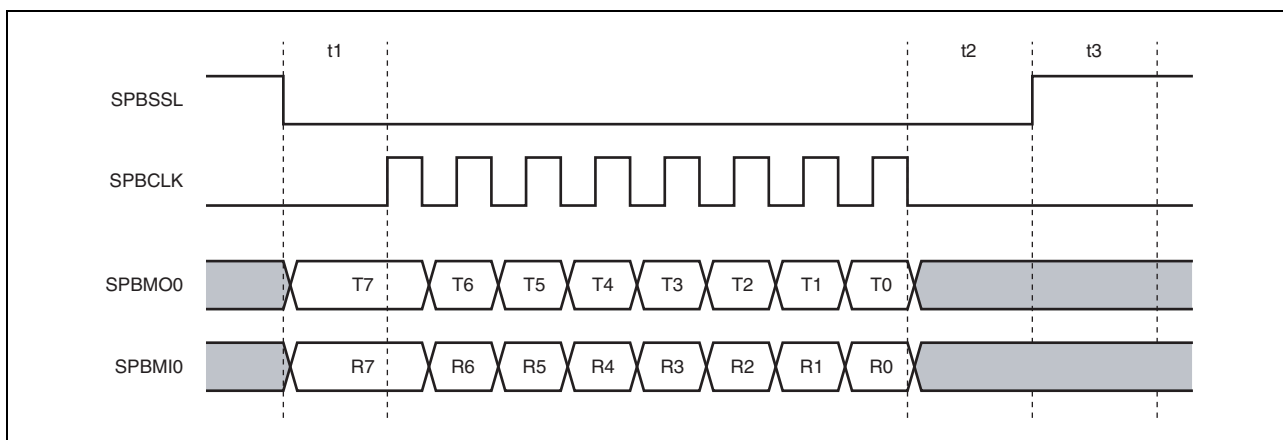


Figure 17.21 Transfer Format Example with 1-Bit Data Size and One Serial Flash Memory Connected

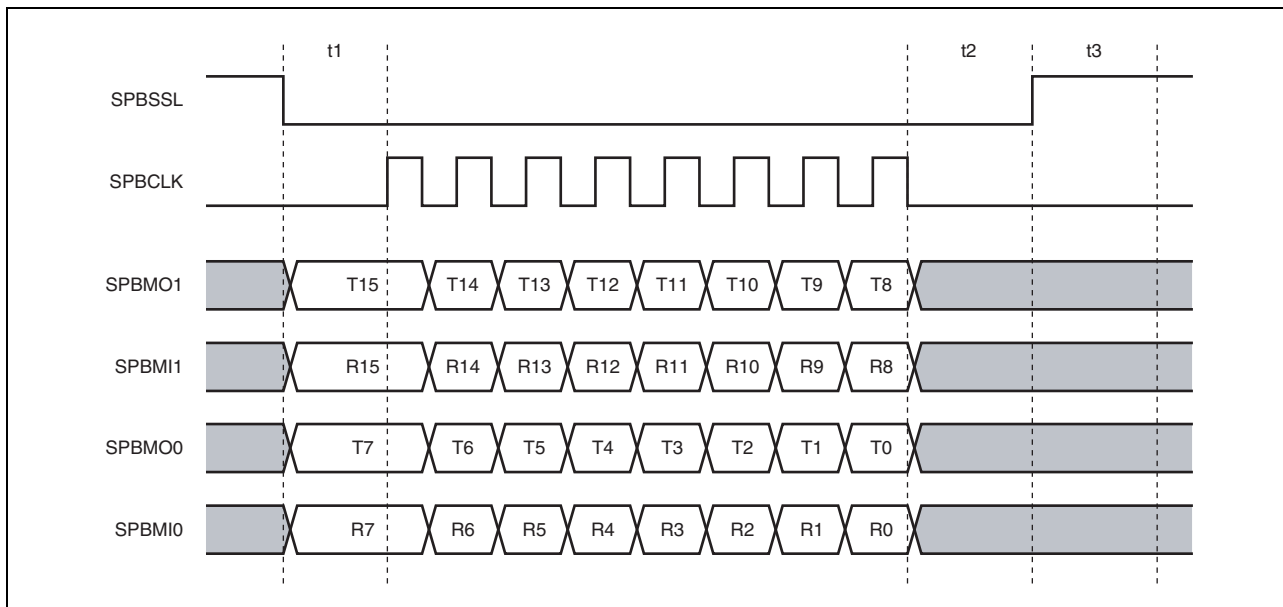


Figure 17.22 Transfer Format Example with 1-Bit Data Size and Two Serial Flash Memories Connected

(b) 2-bit Size

When the size is set to 2 bits, SPBIO0\_0, SPBIO01, SPBIO10, and SPBIO11 pins will be either the input pins or the output pins. SPBIO20, SPBIO21, SPBIO30, and SPBIO31 pins are not used.

Figure 17.23 and Figure 17.24 show the transfer format examples.

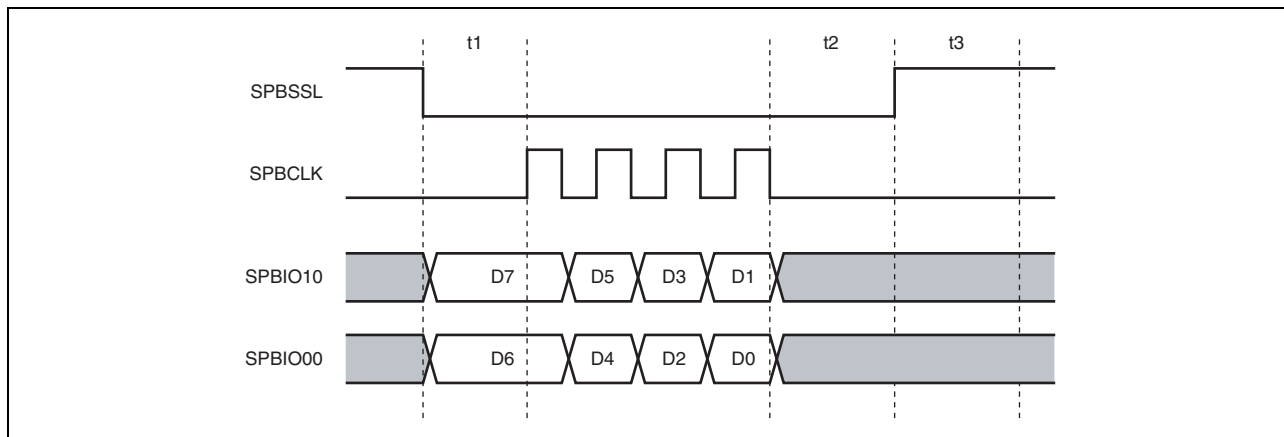


Figure 17.23 Transfer Format Example with 2-Bit Data Size and One Serial Flash Memory Connected

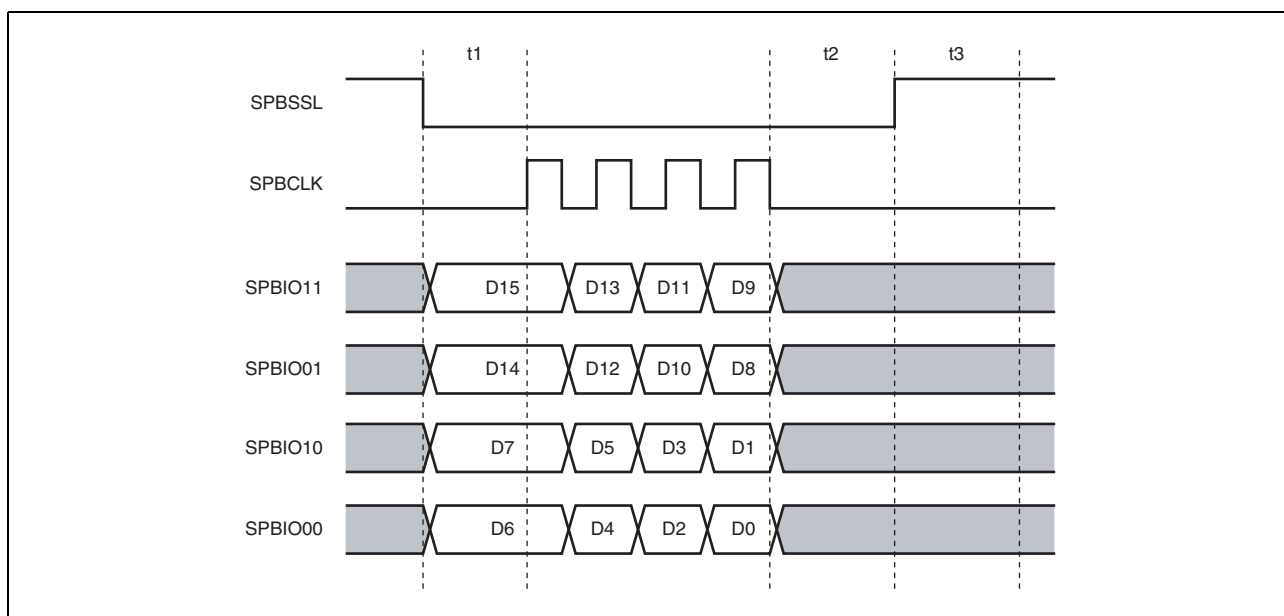


Figure 17.24 Transfer Format Example with 2-Bit Data Size and Two Serial Flash Memories Connected

(c) 4-bit Size

When the size is set to 4 bits, SPBIO00, SPBIO01, SPBIO10, SPBIO11, SPBIO20, SPBIO21, SPBIO30, and SPBIO31 pins will be either the input pins or the output pins.

Figure 17.25 and Figure 17.26 show the transfer format examples.

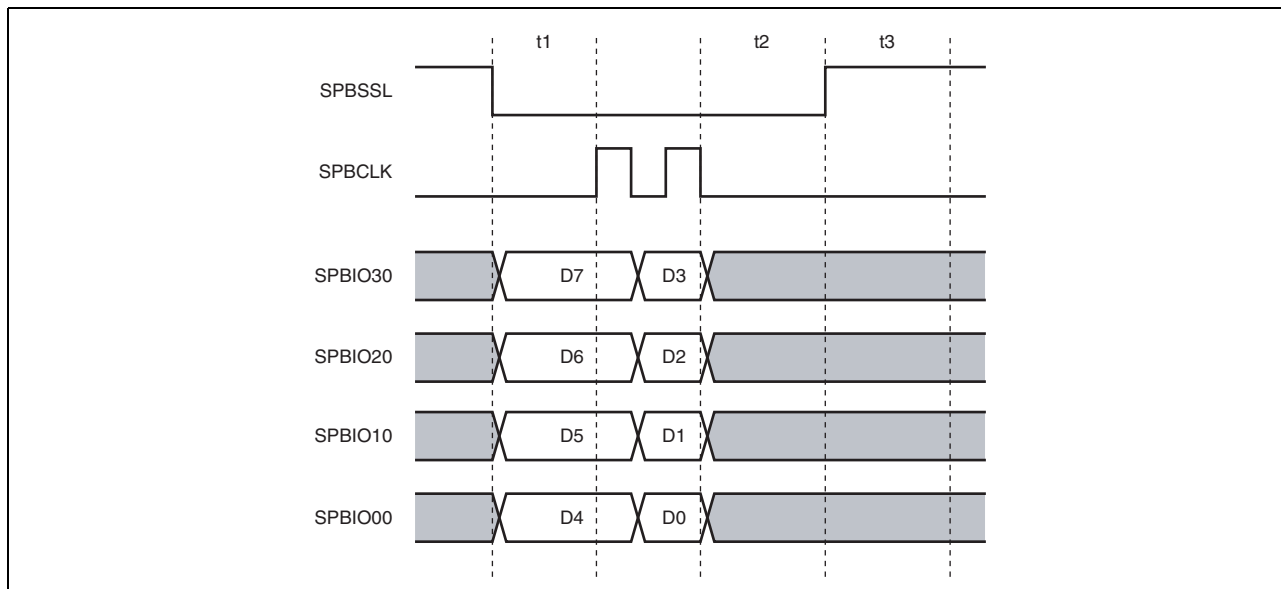


Figure 17.25 Transfer Format Example with 4-Bit Data Size and One Serial Flash Memory Connected

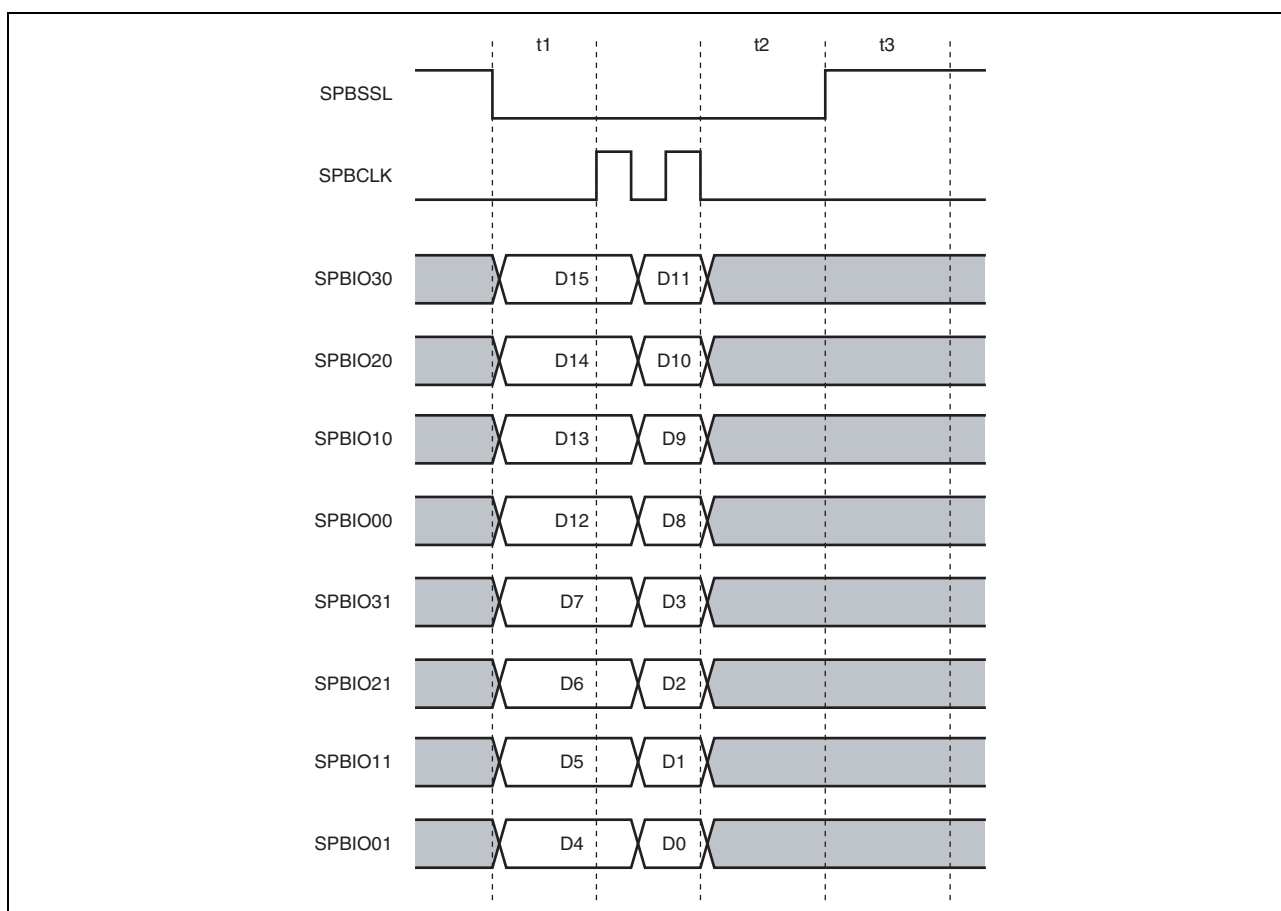


Figure 17.26 Transfer Format Example with 4-Bit Data Size and Two Serial Flash Memories Connected

### 17.5.11 Data Pin Control

With this module, the status of pins can be automatically changed based on the data size to be used and the read/write settings. The pin status during the SPBSSL negation can be set with the MOII03, MOII02, MOII01, and MOII00 bits in CMNCR. The SPBSSL and SPBCLK pins are always output pins. The status of respective pins is specified in Table 17.6 to Table 17.9.

**Table 17.6 Pin Status (1)**

Pin	SPBSSL Negation	SPBSSL Assertion		
		Command, Optional Command, Address, Option Data		
		1-bit Size	2-bit Size	4-bit Size
SPBMO0/ SPBIO00, SPBMO1/ SPBIO01	MOII00 bit value	Output	Output	Output
SPBMO10/ SPBIO10, SPBMO11/ SPBIO11	MOII01 bit value	Hi-Z	Output	Output
SPBIO20, SPBIO21	MOII02 bit value	IO2FV bit value	IO2FV bit value	Output
SPBIO30, SPBIO31	MOII03 bit value	IO3FV bit value	IO3FV bit value	Output

**Table 17.7 Pin Status (2)**

Pin	Transfer Data					
	External Address Space Read Operation			SPI Operation		
	1-bit Size	2-bit Size	4-bit Size	SPIRE Bit = 1, SPIWE Bit = 0		
				1-bit Size	2-bit Size	4-bit Size
SPBMO0/ SPBIO00, SPBMO1/ SPBIO01	IO0FV bit value	Input	Input	IO0FV bit value	Input	Input
SPBMO10/ SPBIO10, SPBMO11/ SPBIO11	Input	Input	Input	Input	Input	Input
SPBIO20, SPBIO21	MOII02 bit value	MOII02 bit value	Input	MOII02 bit value	MOII02 bit value	Input
SPBIO30, SPBIO31	MOII03 bit value	MOII03 bit value	Input	MOII03 bit value	MOII03 bit value	Input

**Table 17.8 Pin Status (3)**

Pin	Transfer Data					
	SPI Operation					
	SPIRE Bit = 0, SPIWE Bit = 1			SPIRE Bit = 1, SPIWE Bit = 1		
	1-bit Size	2-bit Size	4-bit Size	1-bit Size	2-bit Size	4-bit Size
SPBMO0/ SPBIO00, SPBMO1/ SPBIO01	Output	Output	Output	Output	Setting prohibited	Setting prohibited
SPBMO10/ SPBIO10, SPBMO11/ SPBIO11	Hi-Z	Output	Output	Input	Setting prohibited	Setting prohibited
SPBIO20, SPBIO21	MOIO2 bit value	MOIO2 bit value	Output	MOIO2 bit value	Setting prohibited	Setting prohibited
SPBIO30, SPBIO31	MOIO3 bit value	MOIO3 bit value	Output	MOIO3 bit value	Setting prohibited	Setting prohibited

**Table 17.9 Pin Status (4)**

Pin	Dummy Cycle		
	1-bit Size	2-bit Size	4-bit Size
SPBMO0/ SPBIO00, SPBMO1/ SPBIO01	IO0FV bit value	Hi-Z	Hi-Z
SPBMO10/ SPBIO10, SPBMO11/ SPBIO11	Hi-Z	Hi-Z	Hi-Z
SPBIO20, SPBIO21	IO2FV bit value	IO2FV bit value	Hi-Z
SPBIO30, SPBIO31	IO3FV bit value	IO3FV bit value	Hi-Z

### 17.5.12 SPBSSL Pin Control

Negation conditions of the SPBSSL pin are as follows.

#### (1) External Address Space Read Mode

##### (a) Normal read operation (RBE bit in DRCR = 0)

SPBSSL negated after completing the data transfer and t2 cycle.

##### (b) Burst read without automatic SPBSSL negation (RBE bit in DRCR = 1, SSLE bit in DRCR = 0)

SPBSSL negated after completing the data transfer and t2 cycle.

##### (c) Burst read with automatic SPBSSL negation (RBE bit in DRCR = 1, SSLE bit in DRCR = 1)

- SPBSSL negated after t2 cycle when the read address is not continuous with the previously read address
- SPBSSL negated after the SSLN bit in DRCR is set to 1

#### (2) SPI Operating Mode

##### (a) SPBSSL pin assertion not retained (SSLKP bit in SMCR = 0)

SPBSSL negated after completing the data transfer and t2 cycle.

##### (b) SPBSSL pin assertion retained (SSLKP bit in SMCR = 1)

SPBSSL not negated.

When to be negated, data should be transferred after setting the SSLKP bit to 0.

### 17.5.13 Flags

This module has two flag bits SSLF and TEND in CMNSR. These bits are read-only bits.

#### (1) SSLF Bit

This bit indicates the SPBSSL pin status. The status is 1 when the SPBSSL is asserted, and the status is 0 when the SPBSSL is negated.

#### (2) TEND Bit

This bit indicates whether transfer of data is in progress or the transfer of data has ended.

During t1 time period, data transfer, t2 time period, t3 time period, and waiting for read access by burst read and SPBSSL automatic negation, the TEND bit is read as 0 to indicate that the transfer of data is in progress.

When other than the above, the TEND bit is read as 1 to indicate that transfer of data has ended.

#### (3) Register Re-writing Timing

The status of the TEND bit determines the rewritable registers.

The registers which can be written to, except the SSLN bit in DRCR, should be modified when TEND = 1. Read SMRDR0 and SMRDR1 when TEND = 1. CMNSR can always be read.



## 17.6 Usage Notes

### 17.6.1 Notes on Transfer to Read Data in SPI Operating Mode

If the setting for the bit mode is for division by two or more in SPI operating mode, take note of the following points for caution when setting the SPI mode enable setting register (SMENR) to enable transfer only for reading data.

“Transfer only for reading data” indicates transfer to read data while the CDE, OCDE, ADE[3:0], and OPDE[3:0] bits in SMENR are all 0.

#### (1) Transfer to read data while the signal on the SPBSSL pin is de-asserted

Set the SMENR.SPIDE[3:0] bits to 1100 or 1111 when transfer only for reading data is to proceed.

Transfer will not proceed normally if the setting of the SMENR.SPIDE[3:0] bits is 1000.

#### (2) Transfer to read data while the signal on the SPBSSL pin is asserted

When transfer only for reading data is to proceed, set the SMENR.SPIDE[3:0] bits to 1100 or 1111, or end the immediately preceding transfer with reading data.

When the immediately preceding transfer is of a command, optional command, address, or option data, or is transfer for writing data, the subsequent transfer only for reading data will not proceed normally if the setting of the SMENR.SPIDE[3:0] bits is 1000.

### 17.6.2 Notes on Starting Transfer from the SPBSSL Retained State in SPI Operating Mode

Be sure to set the SPIWE bit in the SMCR register to 1 when the transfer of a command, optional command, address, or option data is started while the SPBSSL pin is being asserted in SPI operating mode.

## 18. I<sup>2</sup>C Bus Interface

This section gives an overall description of the I<sup>2</sup>C bus interface (RIIC).

The first section describes the features specific to this LSI, including the number of units and the register base addresses. The subsequent sections describe the RIIC's functions and registers.

### 18.1 Features

#### 18.1.1 Channels

This LSI has the following number of channels of the I<sup>2</sup>C bus interface (RIIC).

**Table 18.1 Channels of RIIC**

Product Name	RZ/A1H 256 Pins	RZ/A1H 324 Pins
Number of channels	4	
Name	RIICn (n = 0 to 3)	

**Table 18.2 Index**

Index	Description
n	Throughout this section, the individual channels of the I <sup>2</sup> C bus interface are identified by the index "n" (n = 0 to 3); for example, RIICnCR1 for the I <sup>2</sup> C bus control register 1.

#### 18.1.2 Register Base Addresses

All RIICn register addresses are given as values obtained by adding offsets to the register base address <RIICn\_base> for each channel.

The base address <RIICn\_base> of each RIICn is listed in the following table:

**Table 18.3 Register base addresses <RIICn\_base>**

Channel	Register base address <RIICn_base>
RIIC0	FCFEE00 <sub>H</sub>
RIIC1	FCFEE400 <sub>H</sub>
RIIC2	FCFEE800 <sub>H</sub>
RIIC3	FCFEEC00 <sub>H</sub>

### 18.1.3 I/O signals

The following table shows the I/O signals of the RIIC.

**Table 18.4 RIICn Pin Configuration**

Alternative Port Pin Name	Function
<b>RIIC0</b>	
Port RIIC0SCL	RIIC0 serial clock I/O pin
Port RIIC0SDA	RIIC0 serial data I/O pin
<b>RIIC1</b>	
Port RIIC1SCL	RIIC1 serial clock I/O pin
Port RIIC1SDA	RIIC1 serial data I/O pin
<b>RIIC2</b>	
Port RIIC2SCL	RIIC2 serial clock I/O pin
Port RIIC2SDA	RIIC2 serial data I/O pin
<b>RIIC3</b>	
Port RIIC3SCL	RIIC3 serial clock I/O pin
Port RIIC3SDA	RIIC3 serial data I/O pin

## 18.2 Overview

### 18.2.1 Functional Overview

#### Communications format

- I<sup>2</sup>C bus format or SMBus format
- Master mode or slave mode selectable
- Automatic securing of the various set-up times, hold times, and bus-free times for the transfer rate

#### Transfer rate

Up to 400 kbps

#### SCL clock

For master operation, the duty cycle of the SCL clock is selectable in the range from 0% to 100%.

#### Issuing and detecting conditions

Start, restart, and stop conditions are automatically generated. Start conditions (including restart conditions) and stop conditions are detectable.

#### Slave address

- Up to three slave-address settings can be made.
- Seven- and ten-bit address formats are supported (along with the use of both at once).
- General call addresses, device ID addresses, and SMBus host addresses are detectable.

#### Acknowledgement

- For transmission, the acknowledge bit is automatically loaded
  - Transfer of the next data for transmission can be automatically suspended on detection of a not-acknowledge bit.
- For reception, the acknowledge bit is automatically transmitted
  - If a wait between the eighth and ninth clock cycles has been selected, software control of the value in the acknowledge field in response to the received value is possible.

#### Wait function

- In reception, the following periods of waiting can be obtained by holding the clock signal (SCL) at the low level:
  - Waiting between the eighth and ninth clock cycles
  - Waiting between the ninth clock cycle and the first clock cycle of the next transfer (WAIT function)

#### SDA output delay function

Timing of the output of transmitted data, including the acknowledge bit, can be delayed.

### Arbitration

- For multi-master operation
  - Operation to synchronize the SCL (clock) signal in cases of conflict with the SCL signal from another master is possible.
  - When issuing the start condition would create conflict on the bus, loss of arbitration is detected by testing for non-matching between the internal signal for the SDA line and the level on the SDA line.
  - In master operation, loss of arbitration is detected by testing for non-matching between the signal on the SDA line and the internal signal for the SDA line.
- Loss of arbitration due to detection of the start condition while the bus is busy is detectable (to prevent the issuing of double start conditions).
- Loss of arbitration in transfer of a not-acknowledge bit due to the internal signal for the SDA line and the level on the SDA line not matching is detectable.
- Loss of arbitration due to non-matching of internal and line levels for data is detectable in slave transmission.

### Timeout function

The internal time-out function is capable of detecting long-interval stop of the SCL (clock signal).

### Noise removal

The interface incorporates digital noise filters for both the SCL and SDA signals, and the width for noise cancellation by the filters is adjustable by software.

### Interrupt sources

- Eight sources:
  - Transmission complete
  - Receive-data-full
  - Transmit-data-empty
  - Detection of a stop condition
  - Detection of a start condition
  - Reception of a NACK
  - Arbitration lost
  - Timeout

### Low power consumption function

Module-stop state can be set.

### 18.2.2 Block Diagram

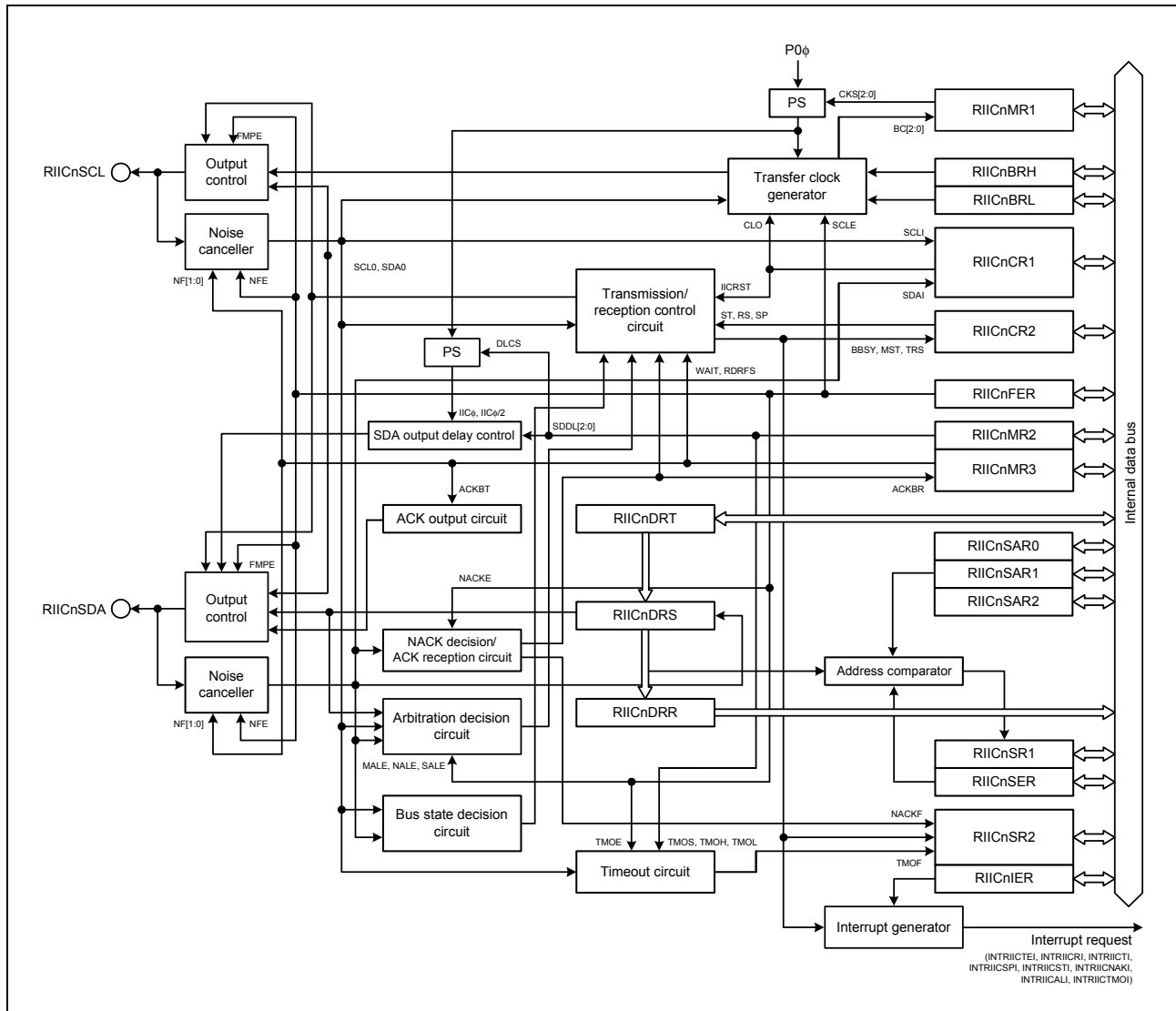
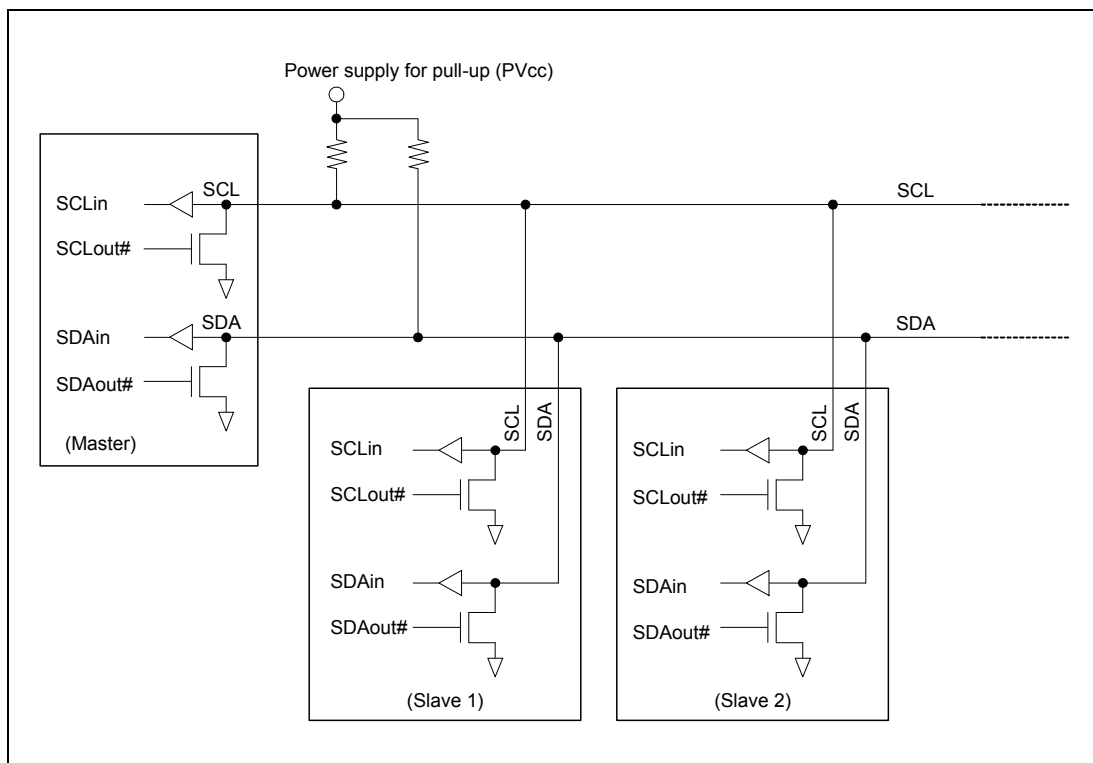


Figure 18.1 Block Diagram of RIIC



**Figure 18.2 Connections to the External Circuit by the I/O Pins (I<sup>2</sup>C Bus Configuration Example)**

SCL and SDA are open-drain output/Schmitt input pins for both master and slave operations. Since the output is open-drain, an external pull-up resistor should be added.

## 18.3 Registers

### 18.3.1 RIICnCR1 — I<sup>2</sup>C Bus Control Register 1

**Access:** RIICnCR1 is a 32-bit readable/writable register.  
RIICnCR1L and RIICnCR1H are 16-bit readable/writable registers.  
RIICnCR1LL, RIICnCR1LH, RIICnCR1HL, and RIICnCR1HH are 8-bit readable/writable registers.

**Address:** RIICnCR1: <RIICn\_base> + 0000<sub>H</sub>  
RIICnCR1L: <RIICn\_base> + 0000<sub>H</sub>, RIICnCR1H: <RIICn\_base> + 0002<sub>H</sub>  
RIICnCR1LL: <RIICn\_base> + 0000<sub>H</sub>, RIICnCR1LH: <RIICn\_base> + 0001<sub>H</sub>, RIICnCR1HL: <RIICn\_base> + 0002<sub>H</sub>,  
RIICnCR1HH: <RIICn\_base> + 0003<sub>H</sub>

**Initial Value:** 0000 001F<sub>H</sub>. This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ICE	IICRST	CLO	SOWP	SCLO	SDAO	SCLI	SDAI
Initial value	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R

**Table 18.5 RIICnCR1 register contents (1/2)**

Bit Position	Bit Name	Function
b31 to b8	—	Reserved These bits are read as 0. The write value should be 0.
b7	ICE	I <sup>2</sup> C Bus Interface Enable 0: Output to the RIICnSCL and RIICnSDA pins is disabled. (Input to the RIICnSCL and RIICnSDA pins is enabled.) 1: Enabled (RIICnSCL and RIICnSDA pins are in the driving state) (An RIIC reset or an internal reset is selected according to the combination of this bit and IICRST bit settings.)
b6	IICRST	I <sup>2</sup> C Bus Interface Internal Reset 0: Clears the RIIC reset or internal reset. 1: Initiates the RIIC reset or internal reset. (Clears the bit counter and the SCL/SDA output latch)
b5	CLO	Extra SCL Clock Cycle Output 0: Does not output an extra SCL clock cycle (default). 1: Outputs an extra SCL clock cycle. (The CLO bit is cleared automatically after one clock cycle is output.)
b4	SOWP <sup>*2</sup>	SCLO/SDAO Write Protect 0: Allows the SCLO and SDAO bits to be rewritten. (This bit is read as 1.)
b3	SCLO <sup>*1,*2</sup>	SDA Output Control <ul style="list-style-type: none"> <li>Read: <ul style="list-style-type: none"> <li>0: RIICnSCL pin output is at a low level.</li> <li>1: RIICnSCL pin is in a high-impedance state.</li> </ul> </li> <li>Write: <ul style="list-style-type: none"> <li>0: Changes the RIICnSCL pin output to a low level.</li> <li>1: Changes the RIICnSCL pin in a high-impedance state. (High level output is achieved through an external pull-up resistor.)</li> </ul> </li> </ul>



**Table 18.5 RIICnCR1 register contents (2/2)**

Bit Position	Bit Name	Function
b2	SDAO <sup>1,2</sup>	SDA Output Control <ul style="list-style-type: none"> <li>• Read:               <ul style="list-style-type: none"> <li>0: RIICnSDA pin output is at a low level.</li> <li>1: RIICnSDA pin is in a high-impedance state.</li> </ul> </li> <li>• Write:               <ul style="list-style-type: none"> <li>0: Changes the RIICnSDA pin output to a low level.</li> <li>1: Changes the RIICnSDA pin in a high-impedance state. (High level output is achieved through an external pull-up resistor.)</li> </ul> </li> </ul>
b1	SCLI	SCL Bus Input Monitor <ul style="list-style-type: none"> <li>0: RIICnSCL pin input is at a low level.</li> <li>1: RIICnSCL pin input is at a high level.</li> </ul>
b0	SDAI	SDA Bus Input Monitor <ul style="list-style-type: none"> <li>0: RIICnSDA pin input is at a low level.</li> <li>1: RIICnSDA pin input is at a high level.</li> </ul>

Note 1. Do not write to these bits during communication. Changing a value during communication may cause a transmission or reception failure or an AL error.

Note 2. To change the SDAO and SCLO bits, set the SOWP bit to 0 at the same timing to set the SDAO and SCLO bits to 0.

### CLO Bit (Extra SCL Clock Cycle Output)

This bit is used to output an extra SCL clock cycle for debugging or error processing.

Normally, set the bit to 0. Setting the bit to 1 in a normal communication state causes a communication error.

For details on this function, see **Section 18.13.2, Extra SCL Clock Cycle Output Function**.

### IICRST Bit (I<sup>2</sup>C Bus Interface Internal Reset)

This bit is used to reset the internal states of the RIIC.

Setting this bit to 1 initiates an RIIC reset or internal reset.

Whether an RIIC reset or internal reset is initiated is determined according to the combination with the ICE bit. **Table 18.6** lists the resets of the RIIC.

The RIIC reset resets all registers including the RIICnCR2.BBSY flag and internal states of the RIIC, and the internal reset resets the bit counter (RIICnMR1.BC[2:0] bits), the I<sup>2</sup>C bus shift register (RIICnICDRS), and the I<sup>2</sup>C bus status registers (RIICnSR1 and RIICnICSR2) as well as the internal states of the RIIC. For the reset conditions for each register, see **Section 18.15, Reset States**.

An internal reset initiated with the IICRST bit set to 1 during operation (with the ICE bit set to 1) resets the internal states of the RIIC without initializing the port settings and the control and setting registers of the RIIC when the bus or RIIC hangs up due to a communication error.

If the RIIC hangs up in a low level output state, resetting the internal states cancels the low level output state and releases the bus with the RIICnSCL pin and RIICnSDA pin at a high impedance.

**CAUTION**

If an internal reset is initiated using the IICRST bit for a bus hang-up occurred during communication with the master device in slave mode, the states may become different between the slave device and the master device (due to the difference in the bit counter information). For this reason, do not initiate an internal reset in slave mode, but initiate restoration processing from the master device. If an internal reset is necessary because the RIIC hangs up with the SCL line in a low level output state in slave mode, initiate an internal reset and then issue a restart condition from the master device or resume communication from the start condition issuance after issuing a stop condition. If communication is restarted by initiating a reset solely in the slave device without issuing a start condition or restart condition from the master device, synchronization will be lost because the master and slave devices operate asynchronously.

**Table 18.6 RIIC Resets**

IICRST	ICE	State	Specifications
1	0	RIIC reset	Resets all registers and internal states of the RIIC.
	1	Internal reset	Reset the RIICnMR1.BC[2:0] bits, and the RIICnSR1, RIICnSR2, RIICnDRS registers and the internal states of the RIIC.

**ICE Bit (I<sup>2</sup>C Bus Interface Enable)**

The ICE bit selects driving or non-driving of the RIICnSCL and RIICnSDA pins. Moreover, this bit can perform two types of reset in combination with the IICRST bit. For the types of reset, see **Table 18.6, RIIC Resets**.

Set the ICE bit to 1 when using RIIC. Setting the ICE bit to 1 selects driving of the RIICnSCL and RIICnSDA pins.

Set the ICE bit to 0 when RIIC is not to be used. Clearing the ICE bit 0 stops driving of the RIICnSCL and RIICnSDA pins.

### 18.3.2 RIICnCR2 — I<sup>2</sup>C Bus Control Register 2

**Access:** RIICnCR2 is a 32-bit readable/writable register.

RIICnCR2L and RIICnCR2H are 16-bit readable/writable registers.

RIICnCR2LL, RIICnCR2LH, RIICnCR2HL, and RIICnCR2HH are 8-bit readable/writable registers.

**Address:** RIICnCR2: <RIICn\_base> + 0004<sub>H</sub>

RIICnCR2L: <RIICn\_base> + 0004<sub>H</sub>, RIICnCR2H: <RIICn\_base> + 0006<sub>H</sub>

RIICnCR2LL: <RIICn\_base> + 0004<sub>H</sub>, RIICnCR2LH: <RIICn\_base> + 0005<sub>H</sub>, RIICnCR2HL: <RIICn\_base> + 0006<sub>H</sub>,

RIICnCR2HH: <RIICn\_base> + 0007<sub>H</sub>

**Initial Value:** 0000 0000<sub>H</sub> This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	BBSY	MST	TRS	—	SP	RS	ST	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R

**Table 18.7 RIICnCR2 register contents**

Bit Position	Bit Name	Function
b31 to b8	—	Reserved These bits are read as 0. The write value should be 0.
b7	BBSY	Bus Busy Detection Flag 0: The I <sup>2</sup> C bus is released (bus free state). 1: The I <sup>2</sup> C bus is occupied (bus busy state or in the bus free state).
b6	MST	Master/Slave Mode 0: Slave mode 1: Transmit mode
b5	TRS	Transmit/Receive Mode 0: Receive mode 1: Transmit mode
b4	—	Reserved These bits are read as 0. The write value should be 0.
b3	SP	Stop Condition Issuance Request 0: Does not request to issue a stop condition. 1: Requests to issue a stop condition.
b2	RS	Restart Condition Issuance Request 0: Does not request to issue a restart condition. 1: Requests to issue a restart condition.
b1	ST	Start Condition Issuance Request 0: Does not request to issue a start condition. 1: Requests to issue a start condition.
b0	—	Reserved These bits are read as 0. The write value should be 0.

#### ST Bit (Start Condition Issuance Request)

This bit is used to request transition to master mode and issuance of a start condition.

When this bit is set to 1 to request to issue a start condition, a start condition is issued when the BBSY flag is set to 0 (bus free).

For details on the start condition issuance, see **Section 18.12, Start Condition/Restart Condition/Stop Condition Issuing Function**.

[Setting condition]

When 1 is written to the ST bit

[Clearing conditions]

- When 0 is written to the ST bit
- When a start condition has been issued
- When the RIICnSR2.AL (arbitration-lost) flag is set to 1
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

#### CAUTION

Set the ST bit to 1 (start condition issuance request) when the BBSY flag is set to 0 (bus free).

Note that arbitration may be lost if the ST bit is set to 1 (start condition issuance request) when the BBSY flag is set to 1 (bus busy).

#### RS Bit (Restart Condition Issuance Request)

This bit is used to request that a restart condition be issued in master mode.

When this bit is set to 1 to request to issue a restart condition, a restart condition is issued when the BBSY flag is set to 1 (bus busy) and the MST bit is set to 1 (master mode).

For details on the restart condition issuance, see **Section 18.12, Start Condition/Restart Condition/Stop Condition Issuing Function**.

[Setting condition]

When 1 is written to the RS bit with the RIICnCR2.BBSY flag set to 1

[Clearing conditions]

- When 0 is written to the RS bit
- When a restart condition has been issued or a start condition is detected
- When the RIICnCR2.AL (arbitration-lost) flag is set to 1
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

#### CAUTIONS

1. Do not set the RS bit to 1 while issuing a stop condition.
2. If the RS bit is set to 1 (restart condition issuance request) in mode other than master mode, the restart condition is not issued in this mode but the restart condition issuance request bit remains set. If the operating mode changes to master mode with the bit not being cleared, the restart condition may be issued.

**SP Bit (Stop Condition Issuance Request)**

This bit is used to request that a stop condition be issued in master mode.

When this bit is set to 1 to request to issue a stop condition, a stop condition is issued when the BBSY flag is set to 1 (bus busy) and the MST bit is set to 1 (master mode).

For details on the stop condition issuance, see **Section 18.12, Start Condition/Restart Condition/Stop Condition Issuing Function**.

[Setting condition]

When 1 is written to the SP bit with both the RIICnCR2.BBSY flag and the RIICnCR2.MST bit set to 1

[Clearing conditions]

- When 0 is written to the SP bit
- When a stop condition has been issued or a stop condition is detected
- When the RIICnSR2.AL (arbitration-lost) flag is set to 1
- When a start condition and a restart condition are detected
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

**CAUTIONS**

1. Writing to the SP bit is not possible while the setting of the BBSY flag is 0 (bus free).
2. Do not set the SP bit to 1 while a restart condition is being issued.

**TRS Bit (Transmit/Receive Mode)**

This bit indicates transmit or receive mode.

The RIIC is in receive mode when the TRS bit is set to 0 and is in transmit mode when the bit is set to 1. Combination of this bit and the MST bit indicates the operating mode of the RIIC.

The value of the TRS bit is automatically changed to the value for transmission mode or reception mode by detection or issuing of a start condition, setting or clearing of the R/W# bit, etc.

[Setting conditions]

- When a start condition is issued normally according to the start condition issuance request (when a start condition is detected with the ST bit set to 1)
- When the R/W# bit added to the slave address is set to 0 in master mode
- When the address received in slave mode matches the address enabled in RIICnSER, with the R/W# bit set to 1

[Clearing conditions]

- When a stop condition is detected
- The RIICnSR2.AL (arbitration-lost) flag being set to 1
- In master mode, reception of a slave address to which an R/W# bit with the value 1 is appended
- In slave mode, a match between the received address and the address enabled in RIICnSER when the value of the received R/W# bit is 0 (including cases where the received address is the general call address)
- In slave mode, a restart condition is detected (a start condition is detected with RIICnCR2.BBSY

= 1 and RIICnCR2.MST = 0)

- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

#### **MST Bit (Master/Slave Mode)**

This bit indicates master or slave mode.

The RIIC is in slave mode when the MST bit is set to 0 and is in master mode when the bit is set to 1. Combination of this bit and the TRS bit indicates the operating mode of the RIIC.

The value of the MST bit is automatically changed to the value for master mode or slave mode by detection or issuing of a start condition, etc.

[Setting conditions]

- When a start condition is issued normally according to the start condition issuance request (when a start condition is detected with the ST bit set to 1)

[Clearing conditions]

- When a stop condition is detected
- When the RIICnSR2.AL (arbitration-lost) flag is set to 1
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

#### **BBSY Flag (Bus Busy Detection)**

The BBSY flag indicates whether the I<sup>2</sup>C bus is occupied (bus busy) or released (bus free).

This bit is set to 1 when the SDA line changes from high to low under the condition of SCL = high, assuming that a start condition has been issued.

When the SDA line changes from low to high under the condition of SCL = high, this bit is cleared to 0 after the bus free time (specified in RIICnBRL) start condition is not detected, assuming that a stop condition has been issued.

[Setting condition]

When a start condition is detected

[Clearing conditions]

- When the bus free time (specified in RIICnBRL) start condition is not detected after detecting a stop condition
- When 1 is written to the RIICnCR1.IICRST bit with the RIICnCR1.ICE bit set to 0 (RIIC reset)

### 18.3.3 RIICnMR1 — I<sup>2</sup>C Bus Mode Register 1

**Access:** RIICnMR1 is a 32-bit readable/writable register.

RIICnMR1L and RIICnMR1H are 16-bit readable/writable registers.

RIICnMR1LL, RIICnMR1LH, RIICnMR1HL, and RIICnMR1HH are 8-bit readable/writable registers.

**Address:** RIICnMR1: <RIICn\_base> + 0008<sub>H</sub>  
 RIICnMR1L: <RIICn\_base> + 0008<sub>H</sub>, RIICnMR1H: <RIICn\_base> + 000A<sub>H</sub>  
 RIICnMR1LL: <RIICn\_base> + 0008<sub>H</sub>, RIICnMR1LH: <RIICn\_base> + 0009<sub>H</sub>, RIICnMR1HL: <RIICn\_base> + 000A<sub>H</sub>,  
 RIICnMR1HH: <RIICn\_base> + 000B<sub>H</sub>

**Initial Value:** 0000 0008<sub>H</sub> This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	CKS[2:0]		BCWP	BC[2:0]			
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

**Table 18.8** RIICnMR1 register contents

Bit Position	Bit Name	Function
b31 to b7	—	Reserved These bits are read as 0. The write value should be 0.
b6 to b4	CKS[2:0]	Internal Reference Clock (IIC $\phi$ ) Selection b6 b4 0 0 0: IIC $\phi$ = P0 $\phi$ /1 0 0 1: IIC $\phi$ = P0 $\phi$ /2 0 1 0: IIC $\phi$ = P0 $\phi$ /4 0 1 1: IIC $\phi$ = P0 $\phi$ /8 1 0 0: IIC $\phi$ = P0 $\phi$ /16 1 0 1: IIC $\phi$ = P0 $\phi$ /32 1 1 0: IIC $\phi$ = P0 $\phi$ /64 1 1 1: IIC $\phi$ = P0 $\phi$ /128
b3	BCWP <sup>1</sup>	BC Write Protect 0: Enables a value to be written in the BC[2:0] bits. (This bit is read as 1.)
b2 to b0	BC[2:0]	Bit Counter b2 b0 0 0 0: 9 bits 0 0 1: 2 bits 0 1 0: 3 bits 0 1 1: 4 bits 1 0 0: 5 bits 1 0 1: 6 bits 1 1 0: 7 bits 1 1 1: 8 bits

Note 1. Set the BCWP bit to 0 to rewrite the BC[2:0] bits. The BC[2:0] bits must be rewritten by using the MOV instruction.

**BC[2:0] Bits (Bit Counter)**

These bits function as a counter that indicates the number of bits remaining to be transferred at the detection of a rising edge on the SCL line. Although these bits are writable and readable, it is not necessary to access these bits under normal conditions.

To write to these bits, specify the number of bits to be transferred plus one (data is transferred with an additional acknowledge bit) between transferred frames when the SCL line is at a low level.

The values of the BC[2:0] bits return to 000b at the end of a data transfer including the acknowledge bit or when a start condition including a restart condition is detected.



### 18.3.4 RIICnMR2 — I<sup>2</sup>C Bus Mode Register 2

**Access:** RIICnMR2 is a 32-bit readable/writable register.

RIICnMR2L and RIICnMR2H are 16-bit readable/writable registers.

RIICnMR2LL, RIICnMR2LH, RIICnMR2HL, and RIICnMR2HH are 8-bit readable/writable registers.

**Address:** RIICnMR2: <RIICn\_base> + 000C<sub>H</sub>

RIICnMR2L: <RIICn\_base> + 000C<sub>H</sub>, RIICnMR2H: <RIICn\_base> + 000E<sub>H</sub>

RIICnMR2LL: <RIICn\_base> + 000C<sub>H</sub>, RIICnMR2LH: <RIICn\_base> + 000D<sub>H</sub>, RIICnMR2HL: <RIICn\_base> + 000E<sub>H</sub>,

RIICnMR2HH: <RIICn\_base> + 000F<sub>H</sub>

**Initial Value:** 0000 0006<sub>H</sub> This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	DLCS	SDDL[2:0]			—	TMOH	TMOL	TMOS
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

**Table 18.9** RIICnMR2 register contents (1/2)

Bit Position	Bit Name	Function
b31 to b8	—	Reserved These bits are read as 0. The write value should be 0.
b7	DLCS	SDA Output Delay Clock Source Selection 0: The internal reference clock (IIC $\phi$ ) is selected as the clock source of the SDA output delay counter. 1: The internal reference clock divided by 2 (IIC $\phi$ /2) is selected as the clock source of the SDA output delay counter.* <sup>1</sup>
b6 to b4	SDDL[2:0]	SDA Output Delay Counter <ul style="list-style-type: none"> <li>When RIICnMR2.DLCS = 0 (IIC<math>\phi</math>) <ul style="list-style-type: none"> <li>b6 b4</li> <li>0 0 0: No output delay</li> <li>0 0 1: 1 IIC<math>\phi</math> cycle</li> <li>0 1 0: 2 IIC<math>\phi</math> cycles</li> <li>0 1 1: 3 IIC<math>\phi</math> cycles</li> <li>1 0 0: 4 IIC<math>\phi</math> cycles</li> <li>1 0 1: 5 IIC<math>\phi</math> cycles</li> <li>1 1 0: 6 IIC<math>\phi</math> cycles</li> <li>1 1 1: 7 IIC<math>\phi</math> cycles</li> </ul> </li> <li>When RIICnMR2.DLCS = 1 (IIC<math>\phi</math>/2) <ul style="list-style-type: none"> <li>b6 b4</li> <li>0 0 0: No output delay</li> <li>0 0 1: 1 or 2 IIC<math>\phi</math> cycles</li> <li>0 1 0: 3 or 4 IIC<math>\phi</math> cycles</li> <li>0 1 1: 5 or 6 IIC<math>\phi</math> cycles</li> <li>1 0 0: 7 or 8 IIC<math>\phi</math> cycles</li> </ul> </li> </ul>
b3	—	Reserved These bits are read as 0. The write value should be 0.
b2	TMOH	Timeout H Count Control 0: Count is disabled while the SCL line is at a high level. 1: Count is enabled while the SCL line is at a high level.
b1	TMOL	Timeout L Count Control 0: Count is disabled while the SCL line is at a low level. 1: Count is enabled while the SCL line is at a low level.

Table 18.9 RIICnMR2 register contents (2/2)

Bit Position	Bit Name	Function
b0	TMOS	Timeout Detection Time Selection 0: Long mode is selected. 1: Short mode is selected.

Note 1. The setting DLCS = 1 (IIC $\phi$ /2) only becomes valid when SCL is at the low level. When SCL is at the high level, the setting DLCS = 1 becomes invalid and the clock source becomes the internal reference clock (IIC $\phi$ ).

#### TMOS Bit (Timeout Detection Time Selection)

This bit is used to select long mode or short mode for the timeout detection time when the timeout function is enabled (RIICnFER.TMOE bit = 1). When this bit is set to 0, long mode is selected. When this bit is set to 1, short mode is selected. In long mode, the timeout detection internal counter functions as a 16 bit-counter. In short mode, the counter functions as a 14 bit-counter. While the SCL line is in the state that enables this counter as specified by bits TMOH and TMOL, the counter counts up in synchronization with the internal reference clock (IIC $\phi$ ) as a count source.

For details on the timeout function, see **Section 18.13.1, Timeout Function**.

#### TMOL Bit (Timeout L Count Control)

This bit is used to enable or disable the internal counter of the timeout function to count up while the SCL line is held low when the timeout function is enabled (RIICnFER.TMOE bit = 1).

#### TMOH Bit (Timeout H Count Control)

This bit is used to enable or disable the internal counter of the timeout function to count up while the SCL line is held high when the timeout function is enabled (RIICnFER.TMOE bit = 1).

#### SDDL[2:0] Bits (SDA Output Delay Setup Counter)

The SDA output can be delayed by the SDDL[2:0] setting. This counter works with the clock source selected by the DLCS bit. The setting of this function can be used for all types of SDA output, including the transmission of the acknowledge bit.

For details on this function, see **Section 18.7, Facility for Delaying SDA Output**.

#### CAUTION

Set the SDA output delay time to meet the I<sup>2</sup>C bus standard (within the data enable time/acknowledge enable time\*<sup>1</sup>) or the SMBus standard (within the data hold time: 300 [ns] or more, and SCL-clock low-level period - the data setup time: 250 [ns]). Note that, if a value outside the standard is set, communication with communication devices may malfunction or it may seemingly become a start condition or stop condition depending on the bus state.

Note 1. Data enable time/acknowledge enable time  
3,450 [ns] (0 to 100 [kbps]: standard mode [Sm])  
900 [ns] (0 to 400 [kbps]: fast mode [fm])

### 18.3.5 RIICnMR3 — I<sup>2</sup>C Bus Mode Register 3

**Access:** RIICnMR3 is a 32-bit readable/writable register.

RIICnMR3L and RIICnMR3H are 16-bit readable/writable registers.

RIICnMR3LL, RIICnMR3LH, RIICnMR3HL, and RIICnMR3HH are 8-bit readable/writable registers.

**Address:** RIICnMR3: <RIICn\_base> + 0010<sub>H</sub>

RIICnMR3L: <RIICn\_base> + 0010<sub>H</sub>, RIICnMR3H: <RIICn\_base> + 0012<sub>H</sub>

RIICnMR3LL: <RIICn\_base> + 0010<sub>H</sub>, RIICnMR3LH: <RIICn\_base> + 0011<sub>H</sub>, RIICnMR3HL: <RIICn\_base> + 0012<sub>H</sub>,

RIICnMR3HH: <RIICn\_base> + 0013<sub>H</sub>

**Initial Value:** 0000 0000<sub>H</sub> This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	SMBE	WAIT	RDRFS	ACKWP	ACKBT	ACKBR	NF[1:0]	
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	W	R/W	R	R/W	R/W

**Table 18.10** RIICnMR3 register contents

Bit Position	Bit Name	Function
b31 to b8	—	Reserved This bit is read as 0. The write value should be 0.
b7	SMBE	SMBus/I <sup>2</sup> C Bus Selection 0: I <sup>2</sup> C bus is selected. 1: SMBus is selected.
b6	WAIT <sup>*2</sup>	WAIT 0: No WAIT (The period between ninth clock cycle and first clock cycle is not held low.) 1: WAIT (The period between ninth clock cycle and first clock cycle is held low.) Low-hold is released by reading RIICnDRR.
b5	RDRFS <sup>*2</sup>	RDRF Flag Set Timing Selection 0: The RDRF flag is set at the rising edge of the ninth SCL clock cycle. (The SCL line is not held low at the falling edge of the eighth clock cycle.) 1: The RDRF flag is set at the rising edge of the eighth SCL clock cycle. (The SCL line is held low at the falling edge of the eighth clock cycle.) Low-hold is released by writing a value to the ACKBT bit.
b4	ACKWP <sup>*1</sup>	ACKBT Write Protect 0: Modification of the ACKBT bit is disabled. 1: Modification of the ACKBT bit is enabled.
b3	ACKBT <sup>*1</sup>	Transmit Acknowledge 0: A 0 is sent as the acknowledge bit (ACK transmission). 1: A 1 is sent as the acknowledge bit (NACK transmission).
b2	ACKBR	Receive Acknowledge 0: A 0 is received as the acknowledge bit (ACK reception). 1: A 1 is received as the acknowledge bit (NACK reception).
b1, b0	NF[1:0]	Noise Filter Stage Selection b1 b0 0 0: Noise of up to one IIC <sub>φ</sub> cycle is filtered out (single-stage filter). 0 1: Noise of up to two IIC <sub>φ</sub> cycles is filtered out (2-stage filter). 1 0: Noise of up to three IIC <sub>φ</sub> cycles is filtered out (3-stage filter). 1 1: Noise of up to four IIC <sub>φ</sub> cycles is filtered out (4-stage filter).

Note 1. If it is attempted to write 1 to both ACKWP and ACKBT bits, the ACKBT bit cannot be set to 1.

Note 2. The WAIT and RDRFS bits are valid only in receive mode (invalid in transmit mode).

### NF[1:0] Bits (Noise Filter Stage Selection)

These bits are used to select the width of noise that can be removed from the signals input to the SCL and SDA pins.

### ACKBR Bit (Receive Acknowledge)

This bit is used to store the acknowledge bit information received from the receive device in transmit mode.

[Setting condition]

When 1 is received as the acknowledge bit with the RIICnCR2.TRS bit set to 1

[Clearing conditions]

- When 0 is received as the acknowledge bit with the RIICnCR2.TRS bit set to 1
- When 1 is written to the RIICnCR1.IICRST bit while the RIICnCR1.ICE bit is 0 (RIIC reset)

### ACKBT Bit (Transmit Acknowledge)

This bit is used to set the bit to be sent at the acknowledge timing in receive mode.

[Setting condition]

When 1 is written to this bit with the ACKWP bit set to 1

[Clearing conditions]

- When 0 is written to this bit with the ACKWP bit set to 1
- When stop condition issuance is detected (when a stop condition is detected with the RIICnCR2.SP bit set to 1)
- When 1 is written to the RIICnCR1.IICRST bit while the RIICnCR1.ICE bit is 0 (RIIC reset)

### CAUTION

The ACKBT bit must be written to while the ACKWP bit is 1. If the ACKBT bit is written to with the ACKWP bit cleared to 0, writing to the ACKBT bit is disabled.

### ACKWP Bit (ACKBT Write Protect)

This bit is used to control the modification of the ACKBT bit.

### RDRFS Bit (RDRF Flag Set Timing Selection)

This bit is used to select the RDRF flag set timing in receive mode and also to select whether to hold the SCL line low at the falling edge of the eighth SCL clock cycle.

When the RDRFS bit is 0, the SCL line is not held low at the falling edge of the eighth SCL clock cycle, and the RDRF flag is set to 1 at the rising edge of the ninth SCL clock cycle.

When the RDRFS bit is 1, the RDRF flag is set to 1 at the rising edge of the eighth SCL clock cycle and the SCL line is held low at the falling edge of the eighth SCL clock cycle. The low-hold of the SCL line is released by writing a value to the ACKBT bit.

After data is received with this setting, the SCL line is automatically held low before the acknowledge bit is sent. This enables processing to send ACK (ACKBT = 0) or NACK (ACKBT = 1) according to receive data.

**WAIT Bit (WAIT)**

This bit is used to control whether to hold the period between the ninth SCL clock cycle and the first SCL clock cycle low until the receive data buffer (RIICnDRR) is completely read each time single-byte data is received in receive mode.

When the WAIT bit is 0, the receive operation is continued without holding the period between the ninth and the first SCL clock cycle low. When both the RDRFS and WAIT bits are 0, continuous receive operation is enabled with the double buffer.

When the WAIT bit is 1, the SCL line is held low from the falling edge of the ninth clock cycle until the RIICnDRR value is read each time single-byte data is received. This enables receive operation in byte units.

**CAUTION**

---

When the value of the WAIT bit is to be read, be sure to read the RIICnDRR beforehand.

---

**SMBE Bit (SMBus Pin Enable)**

Setting this bit to 1 enables the SMBus pin and the RIICnSER.HOAE bit.

### 18.3.6 RIICnFER — I<sup>2</sup>C Bus Function Enable Register

**Access:** RIICnFER is a 32-bit readable/writable register.

RIICnFERL and RIICnFERH are 16-bit readable/writable registers.

RIICnFERLL, RIICnFERLH, RIICnFERHL, and RIICnFERHH are 8-bit readable/writable registers.

**Address:** RIICnFER: <RIICn\_base> + 0014<sub>H</sub>  
 RIICnFERL: <RIICn\_base> + 0014<sub>H</sub>, RIICnFERH: <RIICn\_base> + 0016<sub>H</sub>  
 RIICnFERLL: <RIICn\_base> + 0014<sub>H</sub>, RIICnFERLH: <RIICn\_base> + 0015<sub>H</sub>, RIICnFERHL: <RIICn\_base> + 0016<sub>H</sub>,  
 RIICnFERHH: <RIICn\_base> + 0017<sub>H</sub>

**Initial Value:** 0000 0072<sub>H</sub> This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	SCLE	NFE	NACKE	SALE	NALE	MALE	TMOE
Initial value	0	0	0	0	0	0	0	0	0	1	1	1	0	0	1	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

**Table 18.11** RIICnFER register contents

Bit Position	Bit Name	Function
b31 to b7	—	Reserved This bit is read as 0. The write value should be 0.
b6	SCLE	SCL Synchronous Circuit Enable 0: No SCL synchronous circuit is used. 1: An SCL synchronous circuit is used.
b5	NFE	Digital Noise Filter Circuit Enable 0: No digital noise filter circuit is used. 1: A digital noise filter circuit is used.
b4	NACKE	NACK Reception Transfer Suspension Enable 0: Transfer operation is not suspended during NACK reception (transfer suspension disabled). 1: Transfer operation is suspended during NACK reception (transfer suspension enabled).
b3	SALE	Slave Arbitration-Lost Detection Enable 0: Slave arbitration-lost detection is disabled. 1: Slave arbitration-lost detection is enabled.
b2	NALE	NACK Transmission Arbitration-Lost Detection Enable 0: NACK transmission arbitration-lost detection is disabled. 1: NACK transmission arbitration-lost detection is enabled.
b1	MALE	Master Arbitration-Lost Detection Enable 0: Master arbitration-lost detection is disabled. (Disables the arbitration-lost detection function and does not clear the RIICnCR2.MST and TRS bits automatically when arbitration is lost.) 1: Master arbitration-lost detection is enabled. (Enables the arbitration-lost detection function and clears the RIICnCR2.MST and TRS bits automatically when arbitration is lost.)
b0	TMOE	Timeout Function Enable 0: The timeout function is disabled. 1: The timeout function is enabled.

**TMOE Bit (Timeout Function Enable)**

This bit is used to enable or disable the timeout function.

For details on the timeout function, see **Section 18.13.1, Timeout Function**.

**MALE Bit (Master Arbitration-Lost Detection Enable)**

This bit is used to specify whether to use the arbitration-lost detection function in master mode.

Normally, set this bit to 1.

**NALE Bit (NACK Transmission Arbitration-Lost Detection Enable)**

This bit is used to specify whether to cause arbitration to be lost when ACK is detected during transmission of NACK in receive mode (such as when slaves with the same address exist on the bus or when two or more masters select the same slave device simultaneously with different number of receive bytes).

**SALE Bit (Slave Arbitration-Lost Detection Enable)**

This bit is used to specify whether to cause arbitration to be lost when a value different from the value being transmitted is detected on the bus in slave transmit mode (such as when slaves with the same address exist on the bus or when a mismatch with the transmit data occurs due to noise).

**NACKE Bit (NACK Reception Transfer Suspension Enable)**

This bit is used to specify whether to continue or discontinue the transfer operation when NACK is received from the slave device in transmit mode. Normally, set this bit to 1.

When NACK is received with the NACKE bit set to 1, the next transfer operation is suspended.

When the NACKE bit is 0, the next transfer operation is continued regardless of the received acknowledge content.

**SCLE Bit (SCL Synchronous Circuit Enable)**

This bit is used to specify whether to synchronize the SCL clock with the SCL input clock. Normally, set this bit to 1.

When the SCLE bit is cleared to 0 (SCL synchronous circuit not used), the RIIC does not synchronize the SCL clock with the SCL input clock. In this setting, the RIIC outputs the SCL clock with the transfer rate set in RIICnBRH and RIICnBRL regardless of the SCL line state. For this reason, if the bus load of the I<sup>2</sup>C bus line is much larger than the specification value or if the SCL clock output overlaps in multiple masters, the short-cycle SCL clock that does not meet the specification may be output. When no SCL synchronous circuit is used, it also affects the issuance of a start condition, restart condition, and stop condition, and the continuous output of extra SCL clock cycles.

This bit must not be cleared to 0 except for checking the output of the transfer rate.

### 18.3.7 RIICnSER — I<sup>2</sup>C Bus Status Enable Register

**Access:** RIICnSER is a 32-bit readable/writable register.

RIICnSERL and RIICnSERH are 16-bit readable/writable registers.

RIICnSERLL, RIICnSERLH, RRIICnSERHL, and RIICnSERHH are 8-bit readable/writable registers.

**Address:** RIICnSER: <RIICn\_base> + 0018<sub>H</sub>  
 RIICnSERL: <RIICn\_base> + 0018<sub>H</sub>, RIICnSERH: <RIICn\_base> + 001A<sub>H</sub>  
 RIICnSERLL: <RIICn\_base> + 0018<sub>H</sub>, RIICnSERLH: <RIICn\_base> + 0019<sub>H</sub>, RIICnSERHL: <RIICn\_base> + 001A<sub>H</sub>,  
 RIICnSERHH: <RIICn\_base> + 001B<sub>H</sub>

**Initial Value:** 0000 0009<sub>H</sub> This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	HOAE	—	DIDE	—	GCE	SAR2E	SAR1E	SAR0E
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1
R/W	R	R	R	R	R	R	R	R	R/W	R	R/W	R	R/W	R/W	R/W	R/W

**Table 18.12** RIICnSER register contents

Bit Position	Bit Name	Function
b31 to b8	—	Reserved This bit is read as 0. The write value should be 0.
b7	HOAE	Host Address Enable 0: Host address detection is disabled. 1: Host address detection is enabled.
b6	—	Reserved This bit is read as 0. The write value should be 0.
b5	DIDE	Device-ID Address Detection Enable 0: Device-ID address detection is disabled. 1: Device-ID address detection is enabled.
b4	—	Reserved This bit is read as 0. The write value should be 0.
b3	GCE	General Call Address Enable 0: General call address detection is disabled. 1: General call address detection is enabled.
b2	SAR2E	Slave Address Register 2 Enable 0: Slave address in RIICnSAR2 is disabled. 1: Slave address in RIICnSAR2 is enabled.
b1	SAR1E	Slave Address Register 1 Enable 0: Slave address in RIICnSAR1 is disabled. 1: Slave address in RIICnSAR1 is enabled.
b0	SAR0E	Slave Address Register 0 Enable 0: Slave address in RIICnSAR0 is disabled. 1: Slave address in RIICnSAR0 is enabled.



**SARyE Bit (Slave Address Register y Enable) (y = 0 to 2)**

This bit is used to enable or disable the received slave address and the slave address set in RIICnSARy.

When this bit is set to 1, the slave address set in RIICnSARy is enabled and is compared with the received slave address.

When this bit is cleared to 0, the slave address set in RIICnSARy is disabled and is ignored even if it matches the received slave address.

**GCE Bit (General Call Address Enable)**

This bit is used to specify whether to ignore the general call address (0000 000b + 0 [W]: All 0) when it is received.

When this bit is set to 1, if the received slave address matches the general call address, the RIIC recognizes the received slave address as the general call address independently of the slave addresses set in RIICnSARy (y = 0 to 2) and performs data receive operation.

When this bit is cleared to 0, the received slave address is ignored even if it matches the general call address.

**DIDE Bit (Device-ID Address Detection Enable)**

This bit is used to specify whether to recognize and execute the Device-ID address when a device ID (1111 100b) is received in the first frame after a start condition or restart condition is detected.

When this bit is set to 1, if the received first frame matches the device ID, the RIIC recognizes that the Device-ID address has been received. When the following R/W# bit is 0 [W], the RIIC recognizes the second and the following frames as slave addresses and continues the receive operation.

When this bit is cleared to 0, the RIIC ignores the received first frame even if it matches the device ID address and recognizes the first frame as a normal slave address.

For details on the device-ID address detection, see **Section 18.9.3, Device-ID Address Detection**.

**HOAE Bit (Host Address Enable)**

This bit is used to specify whether to ignore received host address (0001 000b) when the RIICnMR3.SMBS bit is 1.

When this bit is set to 1 while the RIICnMR3.SMBS bit is 1, if the received slave address matches the host address, the RIIC recognizes the received slave address as the host address independently of the slave addresses set in RIICnSARy (y = 0 to 2) and performs the receive operation.

When the RIICnMR3.SMBS bit or the HOAE bit is cleared to 0, the received slave address is ignored even if it matches the host address.

### 18.3.8 RIICnIER — I<sup>2</sup>C Bus Interrupt Enable Register

**Access:** RIICnIER is a 32-bit readable/writable register.

RIICnIERL and RIICnIERH are 16-bit readable/writable registers.

RIICnIERLL, RIICnIERLH, RIICnIERHL, and RIICnIERHH are 8-bit readable/writable registers.

**Address:** RIICnIER: <RIICn\_base> + 001C<sub>H</sub>  
 RIICnIERL: <RIICn\_base> + 001C<sub>H</sub>, RIICnIERH: <RIICn\_base> + 001E<sub>H</sub>  
 RIICnIERLL: <RIICn\_base> + 001C<sub>H</sub>, RIICnIERLH: <RIICn\_base> + 001D<sub>H</sub>, RIICnIERHL: <RIICn\_base> + 001E<sub>H</sub>,  
 RIICnIERHH: <RIICn\_base> + 001F<sub>H</sub>

**Initial Value:** 0000 0000<sub>H</sub> This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TIE	TEIE	RIE	NAKIE	SPIE	STIE	ALIE	TMOIE
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

**Table 18.13** RIICnIER register contents

Bit Position	Bit Name	Function
b31 to b8	—	Reserved This bit is read as 0. The write value should be 0.
b7	TIE	Transmit Data Empty Interrupt Enable 0: Transmit data empty interrupt request (INTRIICTI) is disabled. 1: Transmit data empty interrupt request (INTRIICTI) is enabled.
b6	TEIE	Transmit End Interrupt Enable 0: Transmit end interrupt request (INTRIICTEI) is disabled. 1: Transmit end interrupt request (INTRIICTEI) is enabled.
b5	RIE	Receive Data Full Interrupt Enable 0: Receive data full interrupt request (INTRIICRI) is disabled. 1: Receive data full interrupt request (INTRIICRI) is enabled.
b4	NAKIE	NACK Reception Interrupt Enable 0: NACK reception interrupt request (INTRIICNAKI) is disabled. 1: NACK reception interrupt request (INTRIICNAKI) is enabled.
b3	SPIE	Stop Condition Detection Interrupt Enable 0: Stop condition detection interrupt request (INTRIICSPI) is disabled. 1: Stop condition detection interrupt request (INTRIICSPI) is enabled.
b2	STIE	Start Condition Detection Interrupt Enable 0: Start condition detection interrupt request (INTRIICSTI) is disabled. 1: Start condition detection interrupt request (INTRIICSTI) is enabled.
b1	ALIE	Arbitration-Lost Interrupt Enable 0: Arbitration-lost interrupt request (INTRIICALI) is disabled. 1: Arbitration-lost interrupt request (INTRIICALI) is enabled.
b0	TMOIE	Timeout Interrupt Enable 0: Timeout interrupt request (INTRIICTMOI) is disabled. 1: Timeout interrupt request (INTRIICTMOI) is enabled.

**TMOIE Bit (Timeout Interrupt Enable)**

This bit is used to enable or disable timeout interrupt requests (INTRIICTMOI) when the RIICnSR2.TMOF flag is set to 1. An INTRIICTMOI interrupt request is canceled by clearing the TMOF flag or the TMOIE bit to 0.

**ALIE Bit (Arbitration-Lost Interrupt Enable)**

This bit is used to enable or disable arbitration-lost interrupt requests (INTRIICALII) when the RIICnSR2.AL flag is set to 1. An INTRIICALII interrupt request is canceled by clearing the AL flag or the ALIE bit to 0.

**STIE Bit (Start Condition Detection Interrupt Enable)**

This bit is used to enable or disable start condition detection interrupt requests (INTRIICSTI) when the RIICnSR2.START flag is set to 1. An INTRIICSTI interrupt request is canceled by clearing the START flag or the STIE bit to 0.

**SPIE Bit (Stop Condition Detection Interrupt Enable)**

This bit is used to enable or disable stop condition detection interrupt requests (INTRIICSPI) when the RIICnSR2.STOP flag is set to 1. An INTRIICSPI interrupt request is canceled by clearing the STOP flag or the SPIE bit to 0.

**NAKIE Bit (NACK Reception Interrupt Enable)**

This bit is used to enable or disable NACK reception interrupt requests (INTRIICNAKI) when the RIICnSR2.NACKF flag is set to 1. An INTRIICNAKI interrupt request is canceled by clearing the NACKF flag or the NAKIE bit to 0.

**RIE Bit (Receive Data Full Interrupt Enable)**

This bit is used to enable or disable receive data full interrupt requests (INTRIICRI) when the RIICnSR2.RDRF flag in ICSR2 is set to 1. An INTRIICRI interrupt request is canceled by clearing the RDRF flag or the RIE bit to 0.

**TEIE Bit (Transmit End Interrupt Enable)**

This bit is used to enable or disable transmit end interrupts (INTRIICTEI) when the RIICnSR2.TEND flag is set to 1. An INTRIICTEI interrupt request is canceled by clearing the TEND flag or the TEIE bit to 0.

**TIE Bit (Transmit Data Empty Interrupt Enable)**

This bit is used to enable or disable transmit data empty interrupts (INTRIICTI) when the RIICnSR2.TDRE flag is set to 1.

### 18.3.9 RIICnSR1 — I<sup>2</sup>C Bus Status Register 1

**Access:** RIICnSR1 is a 32-bit readable/writable register.  
RIICnSR1L and RIICnSR1H are 16-bit readable/writable registers.  
RIICnSR1LL, RIICnSR1LH, RIICnSR1HL, and RIICnSR1HH are 8/1-bit readable/writable registers.

**Address:** RIICnSR1: <RIICn\_base> + 0020<sub>H</sub>  
RIICnSR1L: <RIICn\_base> + 0020<sub>H</sub>, RIICnSR1H: <RIICn\_base> + 0022<sub>H</sub>  
RIICnSR1LL: <RIICn\_base> + 0020<sub>H</sub>, RIICnSR1LH: <RIICn\_base> + 0021<sub>H</sub>, RIICnSR1HL: <RIICn\_base> + 0022<sub>H</sub>,  
RIICnSR1HH: <RIICn\_base> + 0023<sub>H</sub>

**Initial Value:** 0000 0000<sub>H</sub> This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	HOA	—	DID	—	GCA	AAS2	AAS1	AAS0
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R <sub>1</sub> (W)	R	R <sub>1</sub> (W)	R	R <sub>1</sub> (W)	R <sub>1</sub> (W)	R <sub>1</sub> (W)	R <sub>1</sub> (W)

Note 1. Only 0 can be written to this bit.

**Table 18.14** RIICnSR1 register contents

Bit Position	Bit Name	Function
b31 to b8	—	Reserved This bit is read as 0. The write value should be 0.
b7	HOA	Host Address Detection Flag 0: Host address is not detected. 1: Host address is detected.
b6	—	Reserved This bit is read as 0. The write value should be 0.
b5	DID	Device-ID Address Detection Flag 0: Device-ID command is not detected. 1: Device-ID command is detected.
b4	—	Reserved This bit is read as 0. The write value should be 0.
b3	GCA	General Call Address Detection Flag 0: General call address is not detected. 1: General call address is detected.
b2	AAS2	Slave Address 2 Detection Flag 0: Slave address 2 is not detected. 1: Slave address 2 is detected.
b1	AAS1	Slave Address 1 Detection Flag 0: Slave address 1 is not detected. 1: Slave address 1 is detected.
b0	AAS0	Slave Address 0 Detection Flag 0: Slave address 0 is not detected. 1: Slave address 0 is detected.

**AASy Flag (Slave Address y Detection) (y = 0 to 2)**

[Setting conditions]

&lt;For 7-bit address format: RIICnSARy.FSy = 0&gt;

When the received slave address matches the RIICnSARy.SVA[7:1] value with the RIICnSER.SARyE bit set to 1 (slave address y detection enabled)

This flag is set to 1 at the rising edge of the ninth SCL clock cycle in the frame.

&lt;For 10-bit address format: RIICnSARy.FSy = 1&gt;

When the received slave address matches a value of (11110b + RIICnSARy.SVA[9:8]) and the following address matches the RIICnSARy.SVA[7:0] value with the RIICnSER.SARyE bit set to 1 (slave address y detection enabled)

This flag is set to 1 at the rising edge of the ninth SCL clock cycle in the frame.

[Clearing conditions]

- When 0 is written to the AASy bit after reading AASy = 1
- When a stop condition is detected
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

&lt;For 7-bit address format: RIICnSARy.FSy = 0&gt;

- When the received slave address does not match the RIICnSARy.SVA[7:1] value with the RIICnSER.SARyE bit set to 1 (slave address y detection enabled)

This flag is cleared to 0 at the rising edge of the ninth SCL clock cycle in the frame.

&lt;For 10-bit address format: RIICnSARy.FSy = 1&gt;

- When the received slave address does not match a value of (11110b + RIICnSARy.SVA[9:8]) with the RIICnSER.SARyE bit set to 1 (slave address y detection enabled)

This flag is cleared to 0 at the rising edge of the ninth SCL clock cycle in the frame.

- When the received slave address matches a value of (11110b + RIICnSARy.SVA[9:8]) and the following address does not match the RIICnSARy.SVA[7:0] value with the RIICnSER.SARyE bit set to 1 (slave address y detection enabled)

This flag is cleared to 0 at the rising edge of the ninth SCL clock cycle in the frame.

**GCA Flag (General Call Address Detection)**

[Setting condition]

When the received slave address matches the general call address (0000 000b + 0 [W]) with the RIICnSER.GCE bit set to 1 (general call address detection enabled)

This flag is set to 1 at the rising edge of the ninth SCL clock cycle in the frame.

[Clearing conditions]

- When 0 is written to the GCA bit after reading GCA = 1
  - When a stop condition is detected
  - When the received slave address does not match the general call address (0000 000b + 0 [W]) with the RIICnSER.GCE bit set to 1 (general call address detection enabled)
- This flag is cleared to 0 at the rising edge of the ninth SCL clock cycle in the frame.
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

### DID Flag (Device-ID Address Detection)

[Setting condition]

When the first frame received immediately after a start condition or restart condition is detected matches a value of (device ID (1111 100b) + 0 [W]) with the RIICnSER.DIDE bit set to 1 (Device-ID address detection enabled)

This flag is set to 1 at the rising edge of the ninth SCL clock cycle in the frame.

[Clearing conditions]

- When 0 is written to the DID bit after reading DID = 1
- When a stop condition is detected
- When the first frame received immediately after a start condition or restart condition is detected does not match a value of (device ID (1111 100b)) with the RIICnSER.DIDE bit set to 1 (Device-ID address detection enabled)  
This flag is cleared to 0 at the rising edge of the ninth SCL clock cycle in the frame.
- When the first frame received immediately after a start condition or restart condition is detected matches a value of (device ID (1111 100b) + 0 [W]) and the second frame does not match any of slave addresses 0 to 2 with the RIICnSER.DIDE bit set to 1 (Device-ID address detection enabled)  
This flag is cleared to 0 at the rising edge of the ninth SCL clock cycle in the frame.
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

### HOA Flag (Host Address Detection)

[Setting condition]

When the received slave address matches the host address (0001 000b) with the RIICnSER.HOAE bit set to 1 (host address detection enabled)

This flag is set to 1 at the rising edge of the ninth SCL clock cycle in the frame.

[Clearing conditions]

- When 0 is written to the HOA bit after reading HOA = 1
- When a stop condition is detected
- When 0 is written to the RIICnMR3.SMBS bit in ICMR3 or the RIICnSER.HOAE bit
- When the received slave address does not match the host address (0001 000b) with the RIICnSER.HOAE bit set to 1 (host address detection enabled)  
This flag is cleared to 0 at the rising edge of the ninth SCL clock cycle in the frame.
- When 1 is written to the RIICnCR1.IICRST bit 1 to apply an RIIC reset or an internal reset

### 18.3.10 RIICnSR2 — I<sup>2</sup>C Bus Status Register 2

**Access:** RIICnSR2 is a 32-bit readable/writable register.

RIICnSR2L and RIICnSR2H are 16-bit readable/writable registers.

RIICnSR2LL, RIICnSR2LH, RIICnSR2HL, and RIICnSR2HH are 8/1-bit readable/writable registers.

**Address:** RIICnSR2: <RIICn\_base> + 0024<sub>H</sub>  
 RIICnSR2L: <RIICn\_base> + 0024<sub>H</sub>, RIICnSR2H: <RIICn\_base> + 0026<sub>H</sub>  
 RIICnSR2LL: <RIICn\_base> + 0024<sub>H</sub>, RIICnSR2LH: <RIICn\_base> + 0025<sub>H</sub>, RIICnSR2HL: <RIICn\_base> + 0026<sub>H</sub>,  
 RIICnSR2HH: <RIICn\_base> + 0027<sub>H</sub>

**Initial Value:** 0000 0000<sub>H</sub> This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TDRE	TEND	RDRF	NACKF	STOP	START	AL	TMOF
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R <sub>1</sub> (W)	R <sub>1</sub> (W)	R <sub>1</sub> (W)	R <sub>1</sub> (W)	R <sub>1</sub> (W)	R <sub>1</sub> (W)	R <sub>1</sub> (W)

Note 1. Only 0 can be written to this bit.

**Table 18.15** RIICnSR2 register contents

Bit Position	Bit Name	Function
b31 to b8	—	Reserved This bit is read as 0. The write value should be 0.
b7	TDRE	Transmit Data Empty Flag 0: RIICnDRT contains transmit data. 1: RIICnDRT contains no transmit data.
b6	TEND	Transmit End Flag 0: Data is being transmitted. 1: Data has been transmitted.
b5	RDRF	Receive Data Full Flag 0: RIICnDRR contains no receive data. 1: RIICnDRR contains receive data.
b4	NACKF	NACK Reception Flag 0: NACK is not received. 1: NACK is received.
b3	STOP	Stop Condition Detection Flag 0: Stop condition is not detected. 1: Stop condition is detected.
b2	START	Start Condition Detection Flag 0: Start condition is not detected. 1: Start condition is detected.
b1	AL	Arbitration-Lost Flag 0: Arbitration is not lost. 1: Arbitration is lost.
b0	TMOF	Timeout Flag 0: No timeout has occurred. 1: Timeout has occurred.

**TMOF Flag (Timeout)**

This flag is set to 1 when the RIIC recognizes timeout after the SCL line state remains unchanged for a certain period.

[Setting condition]

When the SCL line state remains unchanged for the period specified by bits RIICnMR2.TMOH, TMOL, and TMOS with the RIICnFER.TMOE bit set to 1 (timeout detection enabled) in master mode or in the slave specification state.

[Clearing conditions]

- When 0 is written to the TMOF bit after reading TMOF = 1
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

**AL Flag (Arbitration-Lost)**

This flag shows that bus mastership has been lost (loss in arbitration) due to a bus conflict or some other reason when a start condition is issued or an address and data are transmitted. The RIIC monitors the level on the SDA line during transmission and, if the level on the line does not match the value of the bit being output, sets the value of the AL bit to 1 to indicate that the bus is occupied by another device.

The RIIC can also set the flag to indicate the detection of loss of arbitration during NACK transmission in master mode or during data transmission in slave mode.

[Setting conditions]

<When master arbitration-lost detection is enabled: RIICnFER.MALE = 1>

- When the internal SDA output state does not match the SDA line level at the rising edge of SCL clock except for the ACK period during data (including slave address) transmission in master transmit mode (when the SDA line is driven low while the internal SDA output is at a high level (the SDA pin is in the high-impedance state))
- When a start condition is detected while the RIICnCR2.ST bit is 1 (start condition issuance request) or the internal SDA output state does not match the SDA line level
- When the RIICnCR2.ST bit is set to 1 (start condition issuance request) with the RIICnCR2.BBSY flag set to 1.

<When NACK arbitration-lost detection is enabled: RIICnFER.NALE = 1>

When the internal SDA output state does not match the SDA line level at the rising edge of SCL clock in the ACK period during NACK transmission in receive mode

<When slave arbitration-lost detection is enabled: RIICnFER.SALE = 1>

When the internal SDA output state does not match the SDA line level at the rising edge of SCL clock except for the ACK period during data transmission in slave transmit mode

[Clearing conditions]

- When 0 is written to the AL bit after reading AL = 1
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset



**Table 18.16 Relationship between Arbitration-Lost Generation Sources and Arbitration-Lost Enable Functions**

RIICnFER			RIICnSR2	Error	Arbitration-Lost Generation Source
MALE	NALE	SALE	AL		
1	x	x	1	Start condition issuance error	When internal SDA output state does not match SDA line level when a start condition is detected while the RIICnCR2.ST bit is 1 When RIICnCR2.ST is set to 1 with RIICnCR2.BBSY set to 1
			1	Transmit data mismatch	When transmit data (including slave address) does not match the bus state in master transmit mode
x	1	x	1	NACK transmission mismatch	When ACK is detected during transmission of NACK in master receive mode or slave receive mode
x	x	1	1	Transmit data mismatch	When transmit data does not match the bus state in slave transmit mode

x: Don't care

### START Flag (Start Condition Detection)

[Setting condition]

When a start condition (or a restart condition) is detected

[Clearing conditions]

- When 0 is written to the START bit after reading START = 1
- When a stop condition is detected
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

### STOP Flag (Stop Condition Detection)

[Setting condition]

When a stop condition is detected

[Clearing conditions]

- When 0 is written to the STOP bit after reading STOP = 1
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

### NACKF Flag (NACK Reception)

[Setting condition]

When acknowledge is not received (NACK is received) from the receive device in transmit mode with the RIICnFER.NACKE bit set to 1 (transfer suspension enabled)

[Clearing conditions]

- When 0 is written to the NACKF bit after reading NACKF = 1
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

**CAUTION**

When the NACKF flag is set to 1, the RIIC suspends data transmission/reception. Writing to RIICnDRT in transmit mode or reading from RIICnDRR in receive mode with the NACKF flag set to 1 does not enable data transmit/receive operation. To restart data transmission/reception, clear the NACKF flag to 0.

**RDRF Flag (Receive Data Full)**

[Setting conditions]

- Slave receive mode
  - When the received slave address matches and the RIICnCR2.TRS bit is cleared to 0 after a start condition (or a restart condition) is detected
  - At the rising edge of the eighth or ninth SCL clock cycle (selected by the RIICnMR3.RDRFS bit) after receive data is transferred from RIICnDRS to RIICnDRR
- Master receive mode
  - When the slave address and the data direction are transmitted and the receive mode is entered (the RIICnCR2.TRS bit is set to 1) after a start condition (or a restart condition) is issued
  - At the rising edge of the eighth or ninth SCL clock cycle (selected by the RIICnMR3.RDRFS bit) after receive data is transferred from RIICnDRS to RIICnDRR

[Clearing conditions]

- When 0 is written to the RDRF bit after reading RDRF = 1
- When data is read from RIICnDRR
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

**TEND Flag (Transmit End)**

[Setting condition]

At the rising edge of the ninth SCL clock cycle while the TDRE flag is 1

[Clearing conditions]

- When 0 is written to the TEND bit after reading TEND = 1
- When data is written to RIICnDRT
- When a stop condition is detected
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

**TDRE Flag (Transmit Data Empty)**

[Setting conditions]

- When data has been transferred from RIICnDRT to RIICnDRS and RIICnDRT becomes empty
- When the RIICnCR2.TRS bit is set to 1
  - When the RIICnCR2.MST bit is set to 1 after a start condition (or a restart condition) is detected
  - When the RIIC enters transmit mode from receive mode
- When the received slave address matches while the TRS bit is 1

[Clearing conditions]

- When data is written to RIICnDRT
- When the RIICnCR2.TRS bit is cleared to 0
  - When a stop condition is detected
  - When the RIIC enters receive mode from transmit mode
- When 1 is written to the RIICnCR1.IICRST bit to apply an RIIC reset or an internal reset

**CAUTION**

---

When the NACKF flag is set to 1 while the RIICnFER.NACKE bit is 1, the RIIC suspends data transmission/reception. Here, if the TDRE flag is 0 (next transmit data has been written), data is transferred to the RIICnDRS register and the RIICnDRT register becomes empty at the rising edge of the ninth clock cycle, but the TDRE flag is not set to 1.

---

### 18.3.11 RIICnSARy — I<sup>2</sup>C Slave Address Register y (y = 0 to 2)

**Access:** RIICnSARy is a 32-bit readable/writable register.

RIICnSARyL and RIICnSARyH are 16-bit readable/writable registers.

RIICnSARyLL, RIICnSARyLH, RIICnSARyHL, and RIICnSARyHH are 8-bit readable/writable registers.

**Address:** RIICnSAR0: <RIICn\_base> + 0028<sub>H</sub>  
 RIICnSAR0L: <RIICn\_base> + 0028<sub>H</sub>, RIICnSAR0H: <RIICn\_base> + 002A<sub>H</sub>  
 RIICnSAR0LL: <RIICn\_base> + 0028<sub>H</sub>, RIICnSAR0LH: <RIICn\_base> + 0029<sub>H</sub>, RIICnSAR0HL: <RIICn\_base> + 002A<sub>H</sub>,  
 RIICnSAR0HH: <RIICn\_base> + 002B<sub>H</sub>  
 RIICnSAR1: <RIICn\_base> + 002C<sub>H</sub>  
 RIICnSAR1L: <RIICn\_base> + 002C<sub>H</sub>, RIICnSAR1H: <RIICn\_base> + 002E<sub>H</sub>  
 RIICnSAR1LL: <RIICn\_base> + 002C<sub>H</sub>, RIICnSAR1LH: <RIICn\_base> + 002D<sub>H</sub>, RIICnSAR1HL: <RIICn\_base> + 002E<sub>H</sub>,  
 RIICnSAR1HH: <RIICn\_base> + 002F<sub>H</sub>  
 RIICnSAR2: <RIICn\_base> + 0030<sub>H</sub>  
 RIICnSAR2L: <RIICn\_base> + 0030<sub>H</sub>, RIICnSAR2H: <RIICn\_base> + 0032<sub>H</sub>  
 RIICnSAR2LL: <RIICn\_base> + 0030<sub>H</sub>, RIICnSAR2LH: <RIICn\_base> + 0031<sub>H</sub>, RIICnSAR2HL: <RIICn\_base> + 0032<sub>H</sub>,  
 RIICnSAR2HH: <RIICn\_base> + 0033<sub>H</sub>

**Initial Value:** 0000 0000<sub>H</sub> This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FSy	—	—	—	—	—	SVA[9:1]									SVA0
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

**Table 18.17 RIICnSARy register contents**

Bit Position	Bit Name	Function
b31 to b16	—	Reserved This bit is read as 0. The write value should be 0.
b15	FSy	7-Bit/10-Bit Address Format Selection 0: The 7-bit address format is selected. 1: The 10-bit address format is selected.
b14 to b10	—	Reserved This bit is read as 0. The write value should be 0.
b9 to b1	SVA[9:1]	7-Bit Address/10-Bit Address Lower Bits A slave address is set. <ul style="list-style-type: none"> <li>When the FSy bit is 0 (7-bit address format), the SVA[7:1] bits are Valid and form a 7-bit slave address.</li> <li>When the FSy bit is 1 (10-bit address format), SVA[9:1] bits form a 10-bit slave address (combined with the SVA0 bit).</li> </ul>
b0	SVA0	10-Bit Address LSB The least significant bit (LSB) of a 10-bit slave address is set. <ul style="list-style-type: none"> <li>When the FSy bit is 0 (7-bit address format), this bit is invalid.</li> <li>When the FSy bit is 1 (10-bit address format), this bit is a 10-bit slave address (combined with the SVA[9:1] bits).</li> </ul>

#### SVA0 Bit (10-Bit Address LSB)

When the 10-bit address format is selected (RIICnSARy.FSy = 1), this bit functions as the LSB of a 10-bit address and forms a 10-bit address in combination with the SVA[9:1] bits.

When the RIICnSER.SARyE bit is set to 1 (RIICnSARy enabled) and the RIICnSARy.FSy bit is 1, this bit is valid. While the RIICnSARy.FSy bit or SARyE bit is 0, the setting of this bit is ignored.

**SVA[9:1] Bits (7-Bit Address/10-Bit Address Lower Bits)**

When the 7-bit address format is selected (RIICnSARy.FSy = 0), these bits function as a 7-bit address. When the 10-bit address format is selected (RIICnSARy.FSy = 1), these bits function as a 10-bit address in combination with the SVA0 bit.

While the RIICnSER.SARyE bit is 0, the setting of these bits is ignored.

**FSy Bit (7-Bit/10-Bit Address Format Selection)**

This bit is used to select 7-bit address or 10-bit address for slave address y (in RIICnSARy).

When the RIICnSER.SARyE bit is set to 1 (RIICnSARy enabled) and the RIICnSARy.FSy bit is 0, the 7-bit address format is selected for slave address y, the RIICnSARy.SVA[7:1] setting is valid, and the settings of the SVA[9:8] bits and the RIICnSARy.SVA0 bit are ignored.

When the RIICnSER.SARyE bit is set to 1 (RIICnSARy enabled) and the RIICnSARy.FSy bit is 1, the 10-bit address format is selected for slave address y and the settings of the SVA[9:1] bits and the SVA0 bit are valid.

While the RIICnSER.SARyE bit is 0 (RIICnSARy disabled), the setting of the RIICnSARy.FSy bit is invalid.

### 18.3.12 RIICnBRL — I<sup>2</sup>C Bus Bit Rate Low-Level Register

**Access:** RIICnBRL is a 32-bit readable/writable register.  
RIICnBRL and RIICnBRLH are 16-bit readable/writable registers.  
RIICnBRLLL, RIICnBRLH, RIICnBRLHL, and RIICnBRLHH are 8-bit readable/writable registers.

**Address:** RIICnBRL: <RIICn\_base> + 0034<sub>H</sub>  
RIICnBRL: <RIICn\_base> + 0034<sub>H</sub>, RIICnBRLH: <RIICn\_base> + 0036<sub>H</sub>  
RIICnBRLLL: <RIICn\_base> + 0034<sub>H</sub>, RIICnBRLH: <RIICn\_base> + 0035<sub>H</sub>, RIICnBRLHL: <RIICn\_base> + 0036<sub>H</sub>,  
RIICnBRLHH: <RIICn\_base> + 0037<sub>H</sub>

**Initial Value:** 0000 00FF<sub>H</sub> This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	BRL[4:0]				
Initial value	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

**Table 18.18** RIICnBRL register contents

Bit Position	Bit Name	Function
b31 to b8	—	Reserved This bit is read as 0. The write value should be 0.
b7 to b5	—	Reserved This bit is read as 1. The write value should be 1.
b4 to b0	BRL[4:0]	Bit Rate Low-Level Period Low-level period of SCL clock

The RIICnBRL register is a 5-bit register that is used to set the width at low level for the SCL clock.

It also works to generate the data setup time for automatic SCL low-hold operation (see **Section 18.10, Automatically Low-Hold Function for SCL**); when the RIIC is used only in slave mode, this register needs to be set to a value longer than the data setup time\*<sup>1</sup>.

RIICnBRL counts the low-level period with the internal reference clock source (IIC $\phi$ ) specified by the RIICnMR1.CKS[2:0] bits.

Note 1. Data setup time (t<sub>SU: DAT</sub>)  
250 [ns] (0 to 100 [kbps]: standard mode (Sm))  
100 [ns] (0 to 400 [kbps]: fast mode (Fm))

### 18.3.13 RIICnBRH — I<sup>2</sup>C Bus Bit Rate High-Level Register

**Access:** RIICnBRH is a 32-bit readable/writable register.  
RIICnBRHL and RIICnBRHH are 16-bit readable/writable registers.  
RIICnBRHLL, RIICnBRHLH, RIICnBRHHL, and RIICnBRHHH are 8-bit readable/writable registers.

**Address:** RIICnBRH: <RIICn\_base> + 0038<sub>H</sub>  
RIICnBRHL: <RIICn\_base> + 0038<sub>H</sub>, RIICnBRHH: <RIICn\_base> + 003A<sub>H</sub>  
RIICnBRHLL: <RIICn\_base> + 0038<sub>H</sub>, RIICnBRHLH: <RIICn\_base> + 0039<sub>H</sub>, RIICnBRHHL: <RIICn\_base> + 003A<sub>H</sub>,  
RIICnBRHHH: <RIICn\_base> + 003B<sub>H</sub>

**Initial Value:** 0000 00FF<sub>H</sub> This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	BRH[4:0]				
Initial value	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

**Table 18.19** RIICnBRH register contents

Bit Position	Bit Name	Function
b31 to b8	—	Reserved This bit is read as 0. The write value should be 0.
b7 to b5	—	Reserved This bit is read as 1. The write value should be 1.
b4 to b0	BRH[4:0]	Bit Rate High-Level Period High-level period of SCL clock

RIICnBRH is a 5-bit register to set the high-level period of SCL clock. RIICnCBRH is valid in master mode. If the RIIC is used only in slave mode, this register need not to set the high-level period.

RIICnBRH counts the high-level period with the internal reference clock source (IIC $\phi$ ) specified by the RIICnMR1.CKS[2:0] bits in ICMR1.

The frequency and duty cycle are calculated using one of the following expressions (1) to (5) according to the register settings.

#### CAUTION

The minimum value that can be specified in RIICnBRL and RIICnBRH is determined according to the values of the SCLE and NFE bits in RIICnFER and the NF bit in RIICnMR3. For details of the minimum specifiable value, see **Table 18.20**.

- (1) When SCLE = 0  
 Frequency =  $1 / \{ [(BRH + 1) + (BRL + 1)] / IIC\phi \}$   
 Duty cycle =  $\{ (BRH + 1) / IIC\phi \} / \{ [(BRH + 1) + (BRL + 1)] / IIC\phi \}$

- (2) When SCLE = 1, NFE = 0, CKS = 000 (IIC $\phi$  = P0 $\phi$ )  
 Frequency =  $1 / \{[(BRH + 3) + (BRL + 3)] / IIC\phi\}$   
 Duty cycle =  $\{(BRH + 3) / IIC\phi\} / \{[(BRH + 3) + (BRL + 3)] / IIC\phi\}$
- (3) When SCLE = 1, NFE = 1, CKS = 000 (IIC $\phi$  = P0 $\phi$ )  
 Frequency =  $1 / \{[(BRH + 3 + nf) + (BRL + 3 + nf)] / IIC\phi\}$   
 Duty cycle =  $\{(BRH + 3 + nf) / IIC\phi\} / \{[(BRH + 3 + nf) + (BRL + 3 + nf)] / IIC\phi\}$
- (4) When SCLE = 1, NFE = 0, CKS  $\neq$  000 (IIC $\phi$  < P0 $\phi$ )  
 Frequency =  $1 / \{[(BRH + 2) + (BRL + 2)] / IIC\phi\}$   
 Duty cycle =  $\{(BRH + 2) / IIC\phi\} / \{[(BRH + 2) + (BRL + 2)] / IIC\phi\}$
- (5) When SCLE = 1, NFE = 1, CKS  $\neq$  000 (IIC $\phi$  < P0 $\phi$ )  
 Frequency =  $1 / \{[(BRH + 2 + nf) + (BRL + 2 + nf)] / IIC\phi\}$   
 Duty cycle =  $\{(BRH + 2 + nf) / IIC\phi\} / \{[(BRH + 2 + nf) + (BRL + 2 + nf)] / IIC\phi\}$

Symbols in the expressions

SCLE: RIICnFER.SCLE bit

BRH: RIICnBRH.BRH[4:0] bits

BRL: RIICnBRL.BRL[4:0] bits

CKS: RIICnMR1.CKS bits

NFE: RIICnFER.NFE bit

IIC $\phi$ : Internal reference clock selected by the CKS bits

nf: Number of digital noise filter stages specified in the RIICnMR3.NF[0.1] bits

**Table 18.20 Minimum Specifiable Value for RIICnBRL and RIICnBRH**

SCLE	NFE	nf	Minimum Pulse Width that Passes through Digital Filter	Minimum Specifiable Value for BRH and BRL	Pulse Width when Minimum Value is Specified
0	0	—	1 × IIC $\phi$	1	2 × IIC $\phi$
0	1	1	2 × IIC $\phi$	2	3 × IIC $\phi$
0	1	2	3 × IIC $\phi$	3	4 × IIC $\phi$
0	1	3	4 × IIC $\phi$	4	5 × IIC $\phi$
0	1	4	5 × IIC $\phi$	5	6 × IIC $\phi$
IIC $\phi$ cycle > P0 $\phi$ cycle (CKS $\neq$ 000)					
1	0	—	1 × IIC $\phi$	0	2 × IIC $\phi$
1	1	1	2 × IIC $\phi$	1	4 × IIC $\phi$
1	1	2	3 × IIC $\phi$	2	6 × IIC $\phi$
1	1	3	4 × IIC $\phi$	3	8 × IIC $\phi$
1	1	4	5 × IIC $\phi$	4	10 × IIC $\phi$
IIC $\phi$ cycle = P0 $\phi$ cycle (CKS = 000)					
1	0	—	2 × P0 $\phi$	0	3 × IIC $\phi$



Table 18.20 Minimum Specifiable Value for RIICnBRL and RIICnBRH

SCLE	NFE	nf	Minimum Pulse Width that Passes through Digital Filter	Minimum Specifiable Value for BRH and BRL	Pulse Width when Minimum Value is Specified
1	1	1	$3 \times P0\phi$	1	$5 \times IIC\phi$
1	1	2	$4 \times P0\phi$	2	$7 \times IIC\phi$
1	1	3	$5 \times P0\phi$	3	$9 \times IIC\phi$
1	1	4	$6 \times P0\phi$	4	$11 \times IIC\phi$

Table 18.21 and Table 18.22 list examples of RIICnBRH/RIICnBRL settings.

Table 18.21 Examples of RIICnBRH/RIICnBRL Settings for Transfer Rate (when RIICnFER.SCLE = 1 and RIICnFER.NFE = 0)

Transfer Rate [kbps]	Peripheral Clock Operating Frequency P0 $\phi$ [MHz]								
	25			30			33		
	RIICnMR1. CKS[2:0]	RIICnBRH. BRH	RIICnBRL. BRL	RIICnMR1. CKS[2:0]	RIICnBRH. BRH	RIICnBRL. BRL	RIICnMR1. CKS[2:0]	RIICnBRH. BRH	RIICnBRL. BRL
10	110b	21 (F5h)	14 (EEh)	110b	25 (F9h)	18 (F2h)	111b	12 (ECh)	10 (EAh)
50	100b	20 (F4h)	7 (E7h)	100b	20 (F4h)	13 (EDh)	100b	22 (F6h)	15 (EFh)
100	010b	30 (FEh)	29 (FDh)	011b	18 (F2h)	16 (F0h)	011b	20 (F4h)	18 (F2h)
400	001b	17 (F1h)	10 (EAh)	001b	23 (F7h)	11 (EBh)	001b	24 (F8h)	13 (EDh)

Table 18.22 Examples of RIICnBRH/RIICnBRL Settings for Transfer Rate (when RIICnFER.SCLE = 1, RIICnFER.NFE = 1, and Number of NF Stages = 4)

Transfer Rate [kbps]	Peripheral Clock Operating Frequency P0 $\phi$ [MHz]								
	25			30			33		
	RIICnMR1. CKS[2:0]	RIICnBRH. BRH	RIICnBRL. BRL	RIICnMR1. CKS[2:0]	RIICnBRH. BRH	RIICnBRL. BRL	RIICnMR1. CKS[2:0]	RIICnBRH. BRH	RIICnBRL. BRL
10	110b	19 (F3h)	8 (E8h)	110b	24 (F8h)	11 (EBh)	110b	28 (FCh)	12 (ECh)
50	011b	28 (FCh)	22 (F6h)	100b	19 (F3h)	7 (E7h)	100b	20 (F4h)	10 (EAh)
100	010b	26 (FAh)	25 (F9h)	011b	14 (EEh)	12 (ECh)	011b	16 (F0h)	14 (EEh)
400	000b	27 (FBh)	22 (F6h)	001b	18 (F2h)	8 (E8h)	001b	19 (F3h)	10 (EAh)

### 18.3.14 RIICnDRT — I<sup>2</sup>C Bus Transmit Data Register

**Access:** RIICnDRT is a 32-bit readable/writable register.  
 RIICnDRTL and RIICnDRTH are 16-bit readable/writable registers.  
 RIICnDRTLL, RIICnDRTLH, RIICnDRTHL, and RIICnDRTHH are 8-bit readable/writable registers.

**Address:** RIICnDRT: <RIICn\_base> + 003C<sub>H</sub>  
 RIICnDRTL: <RIICn\_base> + 003C<sub>H</sub>, RIICnDRTH: <RIICn\_base> + 003E<sub>H</sub>  
 RIICnDRTLL: <RIICn\_base> + 003C<sub>H</sub>, RIICnDRTLH: <RIICn\_base> + 003D<sub>H</sub>, RIICnDRTHL: <RIICn\_base> + 003E<sub>H</sub>,  
 RIICnDRTHH: <RIICn\_base> + 003F<sub>H</sub>

**Initial Value:** 0000 00FF<sub>H</sub> This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	DRT[7:0]							
Initial value	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

When RIICnDRT detects a space in the I<sup>2</sup>C bus shift register (RIICnDRS), it transfers the transmit data that has been written to RIICnDRT to RIICnDRS and starts transmitting data in transmit mode.

The double-buffer structure of RIICnDRT and RIICnDRS allows continuous transmit operation if the next transmit data has been written to RIICnDRT while the RIICnDRS data is being transmitted.

RIICnDRT can always be read and written. When writing, be sure to write 0 to bits 8 to 15. Write transmit data to RIICnDRT once when a transmit data empty interrupt (INTRIICTI) request is generated.

### 18.3.15 RIICnDRR — I<sup>2</sup>C Bus Receive Data Register

**Access:** RIICnDRR is a 32-bit readable/writable register.  
RIICnDRRL and RIICnDRRH are 16-bit readable/writable registers.  
RIICnDRRLL, RIICnDRRLH, RIICnDRRH, and RIICnDRRH are 8-bit readable/writable registers.

**Address:** RIICnDRR: <RIICn\_base> + 0040<sub>H</sub>  
RIICnDRRL: <RIICn\_base> + 0040<sub>H</sub>, RIICnDRRH: <RIICn\_base> + 0042<sub>H</sub>  
RIICnDRRLL: <RIICn\_base> + 0040<sub>H</sub>, RIICnDRRLH: <RIICn\_base> + 0041<sub>H</sub>, RIICnDRRH: <RIICn\_base> + 0042<sub>H</sub>,  
RIICnDRRH: <RIICn\_base> + 0043<sub>H</sub>

**Initial Value:** 0000 0000<sub>H</sub> This register is initialized by any reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	DRR[7:0]							
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

When 1 byte of data has been received, the received data is transferred from the I<sup>2</sup>C bus shift register (RIICnDRS) to RIICnDRR to enable the next data to be received.

The double-buffer structure of RIICnDRS and RIICnDRR allows continuous receive operation if the received data has been read from RIICnDRR while RIICnDRS is receiving data.

RIICnDRR cannot be written. Read data from RIICnDRR once when a receive data full interrupt (INTRIICRI) request is generated.

If DRR receives the next receive data before the current data is read from RIICnDRR (while the RIICnSR2.RDRF flag is 1), the RIIC automatically holds the SCL clock low one cycle before the RDRF flag is set to 1 next.

### 18.3.16 RIICnDRS — I<sup>2</sup>C Bus Shift Register

**Access:** This register is not accessible.

**Address:** —

**Initial Value:** 0000 00FF<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	DRS[7:0]							
Initial value	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

RIICnDRS is an 8-bit shift register to transmit and receive data.

During transmission, transmit data is transferred from RIICnDRT to RIICnDRS and is sent from the SDA pin. During reception, data is transferred from RIICnDRS to RIICnDRR after 1 byte of data has been received.

RIICnDRS cannot be accessed directly.

## 18.4 Interrupt Sources

The RIIC issues eight types of interrupt request: transmit end, receive data full, transmit data empty, stop condition detection, start condition detection, NACK reception, arbitration-lost, and timeout.

**Table 18.23** lists details of the several interrupt requests. The receive data full and transmit data empty sources are both capable of launching data transfer by theDMAC.

**Table 18.23** Interrupt Sources

Symbol	Interrupt Source	Interrupt Flag	DMAC Launching	Priority* <sup>1</sup>	Interrupt Condition
INTRIICTEI	Transmission complete	TEND	Not possible	↑ High        Low	TEND = 1 • TEIE = 1
INTRIICRI	Receive-data-full	RDRF	Possible		RDRF = 1 • RIE = 1
INTRIICTI	Transmit-data-empty	TDRE	Possible		TDRE = 1 • TIE = 1
INTRIICSPI	Detection of a stop condition	STOP	Not possible		STOP = 1 • SPIE = 1
INTRIICSTI	Detection of a start condition	START	Not possible		START = 1 • STIE = 1
INTRIICNAKI	Reception of a NACK	NACKF	Not possible		NACKF = 1 • NAKIE = 1
INTRIICALI	Arbitration lost	AL	Not possible		AL = 1 • ALIE = 1
INTRIICTMOI	Timeout	TMOF	Not possible		TMOF = 1 • TMOIE = 1

Note 1. When the interrupt priority register (ICDIPRn) setting is the same

Clear or mask the each flag during interrupt handling.

### CAUTIONS

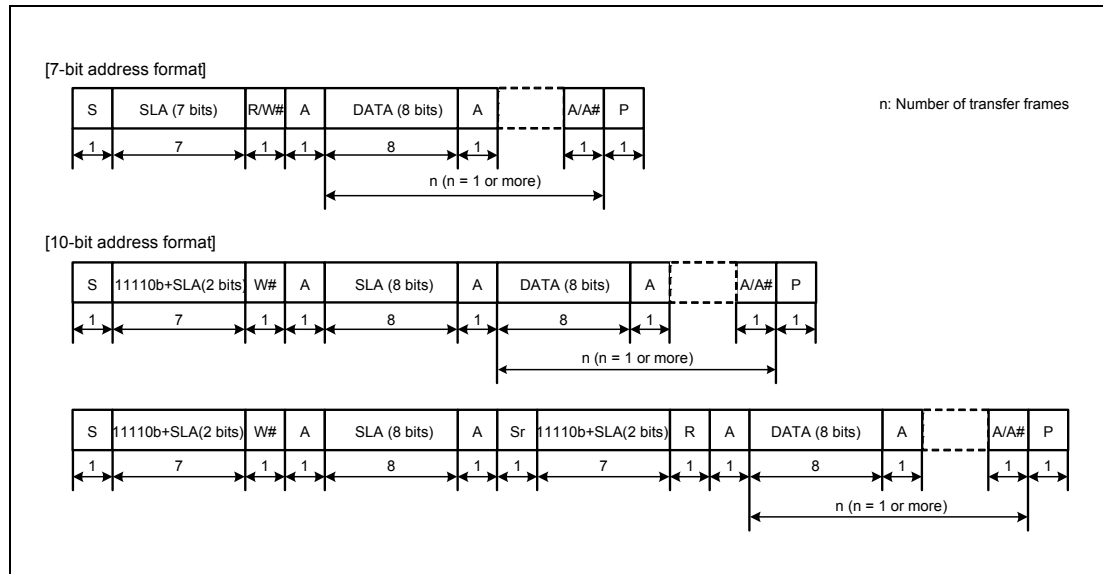
1. There is a latency (delay) between the execution of a write instruction for a peripheral module by the CPU and actual writing to the module. Thus, when an interrupt flag has been cleared or masked, read the relevant flag again to check whether clearing or masking has been completed, and then return from interrupt processing. Returning from interrupt processing without checking that writing to the module has been completed creates a possibility of repeated processing of the same interrupt.
2. Since INTRIICRI and INTRIICTI are edge-detected interrupts, they do not require clearing.
3. When using the INTRIICTEI interrupt, clear the RIICnSR2.TEND flag in the INTRIICTEI interrupt processing.
4. When using the INTRIICSPI interrupt, clear the RIICnSR2.STOP flag in the INTRIICSPI interrupt processing.
5. When using the INTRIICSTI interrupt, clear the RIICnSR2.START flag in the INTRIICSTI interrupt processing.
6. When using the INTRIICNAKI interrupt, clear the RIICnSR2.NACKF flag in the INTRIICNAKI interrupt processing.
7. When using the INTRIICALI interrupt, clear the RIICnSR2.AL flag in the INTRIICALI interrupt processing.
8. When using the INTRIICTMOI interrupt, clear the RIICnSR2.TMOF flag in the INTRIICTMOI interrupt processing.

## 18.5 Operation

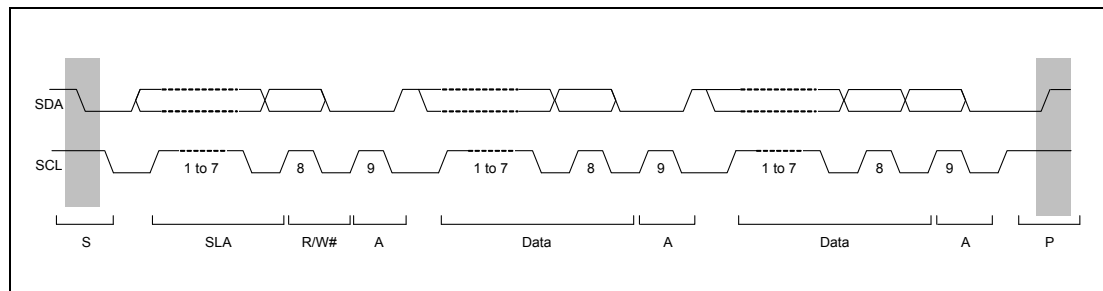
### 18.5.1 Communication Data Format

The I<sup>2</sup>C bus format consists of 8-bit data and 1-bit acknowledge. The frame following a start condition or restart condition is an address frame used to specify a slave device with which the master device communicates. The specified slave is valid until a new slave is specified or a stop condition is issued.

**Figure 18.3** shows the I<sup>2</sup>C bus format, and **Figure 18.4** shows the I<sup>2</sup>C bus timing.



**Figure 18.3** I<sup>2</sup>C Bus Format



**Figure 18.4** I<sup>2</sup>C Bus Timing (SLA = 7 Bits)

**S:** Start condition. The master device drives the SDA line low from high level while the SCL line is at a high level.

**SLA:** Slave address, by which the master device selects a slave device.

**R/W#:** Indicates the direction of data transfer: from the slave device to the master device when R/W is 1, or from the master device to the slave device when R/W is 0.

**A:** Acknowledge. The receive device drives the SDA line low. (In master transmit mode, the slave device returns acknowledge. In master receive mode, the master device returns acknowledge.)

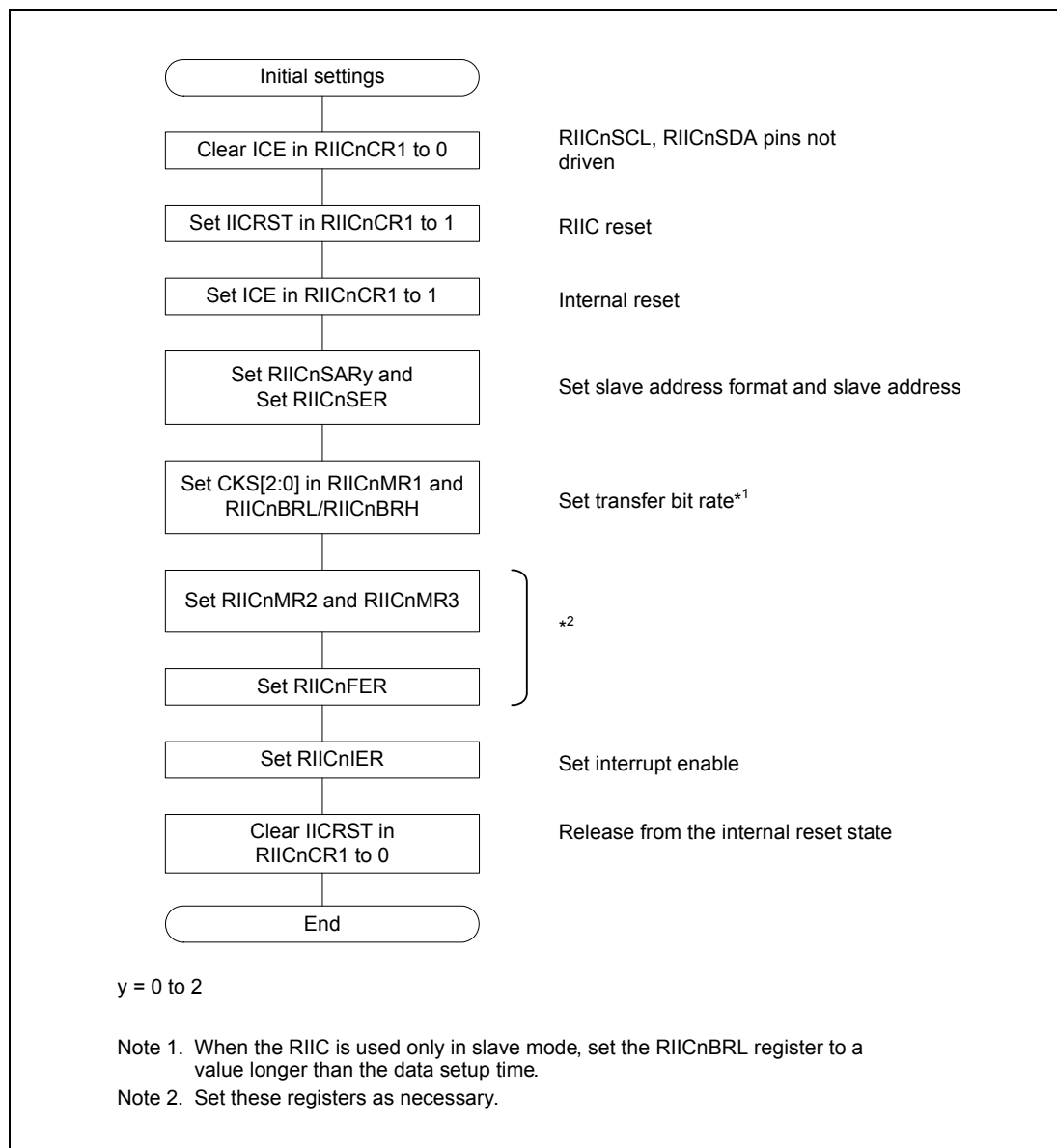
**Sr:** Restart condition. The master device drives the SDA line low from the high level after the setup time has elapsed with the SCL line at the high level.

**DATA:** Transmitted or received data

**P:** Stop condition. The master device drives the SDA line high from low level while the SCL line is at a high level.

## 18.5.2 Initial Settings

Before starting data transmission and reception, initialize the RIIC according to the procedure in **Figure 18.5**.



**Figure 18.5** Example of RIIC Initialization Flowchart

### 18.5.3 Master Transmit Operation

In master transmit operation, the RIIC outputs the SCL (clock) and transmitted data signals as the master device, and the slave device returns acknowledgements. **Figure 18.6** shows an example of usage of master transmission and **Figure 18.7** to **Figure 18.9** show the timing of operations in master transmission.

The following describes the procedure and operations for master transmission.

- (1) Set the RIICnCR1.IICRST bit 1 to 1 (RIIC reset) and then set the RIICnCR1.ICE bit to 1 (internal reset) with the RIICnCR1.ICE bit cleared to 0 (RIICnSCL and RIICnSDA pins not driven). This initializes the internal state and the various flags of RIICnSR1. After that, set registers RIICnSARy, RIICnSER, RIICnMR1, RIICnBRH, and RIICnBRL (y = 0 to 2), and set the other registers as necessary (for initial settings of the RIIC, see **Figure 18.5**). When the necessary register settings have been completed, set the RIICnCR1.IICRST bit to 0 (for release from the reset state). This step is not necessary if initialization of the RIIC has already been completed.
- (2) Read the RIICnCR2.BBSY flag to check that the bus is open, and then set the RIICnCR2.ST bit to 1 (start condition issuance request). Upon receiving the request, the RIIC issues a start condition. At the same time, the BBSY flag and the RIICnSR2.START flag are automatically set to 1 and the ST bit is automatically cleared to 0. At this time, if the start condition is detected and the internal levels for the SDA output state and the levels on the SDA line have matched while the ST bit is 1, the RIIC recognizes that issuing of the start condition as requested by the ST bit has been successfully completed, and the RIICnCR2.MST and TRS bits are automatically set to 1, placing the RIIC in master transmit mode. The RIICnSR2.TDRE flag is also automatically set to 1 in response to setting of the TRS bit to 1.
- (3) Check that the RIICnSR2.TDRE flag is 1, and then write the value for transmission (the slave address and the R/W# bit) to RIICnDRT. Once the data for transmission are written to RIICnDRT, the TDRE flag is automatically cleared to 0, the data are transferred from RIICnDRT to RIICnDRS, and the TDRE flag is again set to 1. After the byte containing the slave address and R/W# bit has been transmitted, the value of the TRS bit is automatically updated to select master transmit or master receive mode in accord with the value of the transmitted R/W# bit. If the value of the R/W# bit was 0, the RIIC continues in master transmit mode.  
 Since the RIICnSR2.NACKF flag being 1 at this time indicates that no slave device recognized the address or there was an error in communications, write 1 to the RIICnCR2.SP bit to issue a stop condition.  
 For data transmission with an address in the 10-bit format, start by writing 1111 0b, the two higher-order bits of the slave address, and W to RIICnDRT as the first address transmission. Then, as the second address transmission, write the eight lower-order bits of the slave address to RIICnDRT.
- (4) After confirming that the RIICnSR2.TDRE flag is 1, write the data for transmission to the RIICnDRT register. The RIIC automatically holds the SCL line low until the data for transmission are ready or a stop condition is issued.
- (5) After all bytes of data for transmission have been written to the RIICnDRT register, wait until the value of the RIICnSR2.TEND flag returns to 1, and then set the RIICnCR2.SP bit to 1 (stop condition issuance request). Upon receiving a stop condition issuance request, the RIIC issues the stop condition.



- (6) Upon detecting the stop condition, the RIIC automatically clears the RIICnCR2.MST and TRS bits to 00b and enters slave receive mode. Furthermore, it automatically clears the RIICnSR2.TDRE and TEND flags to 0, and sets the RIICnSR2.STOP flag in to 1.
- (7) After checking that the RIICnSR2.STOP flag is 1, clear the RIICnSR2.NACKF and STOP flags to 0 for the next transfer operation.

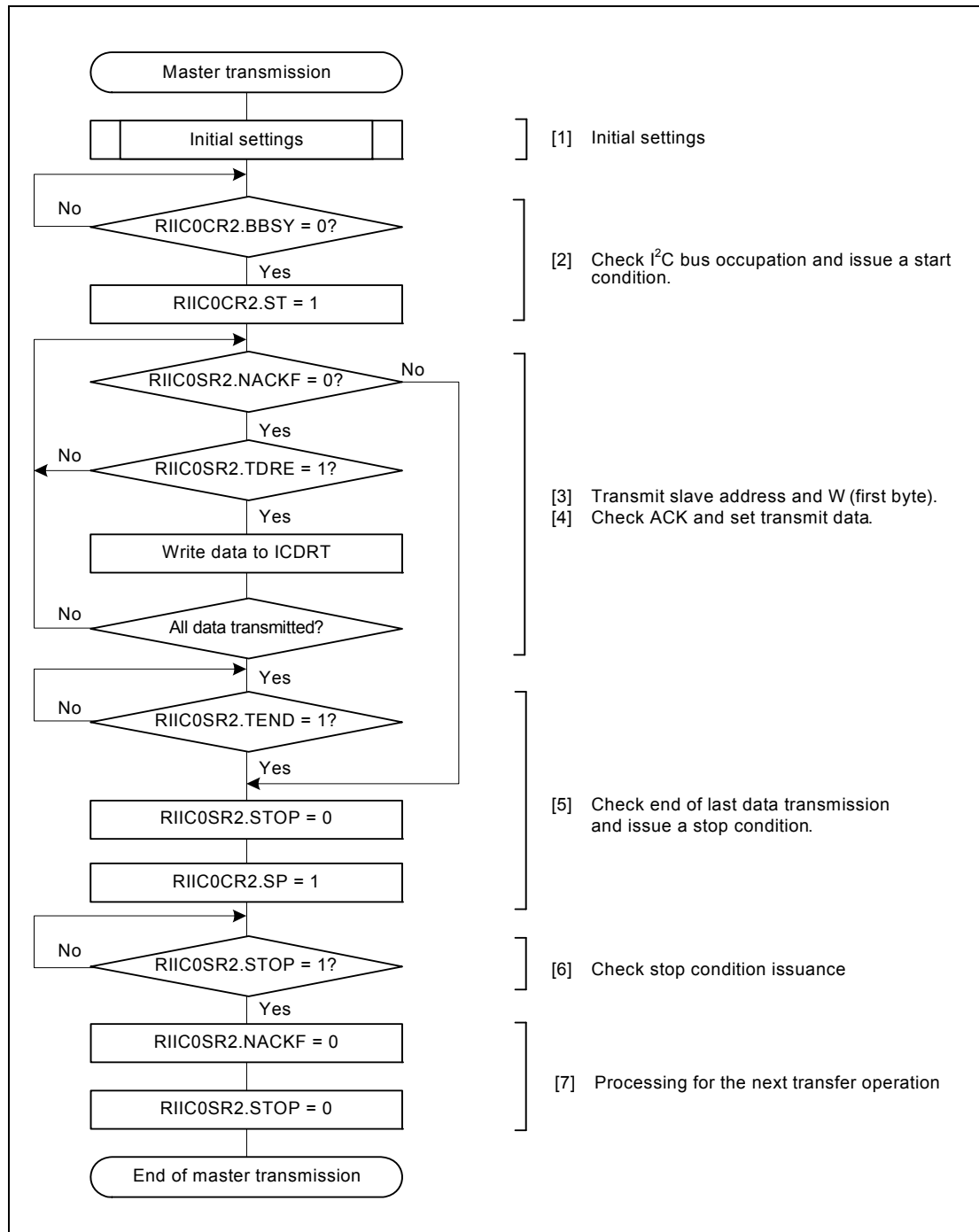


Figure 18.6 Example of Master Transmission Flowchart

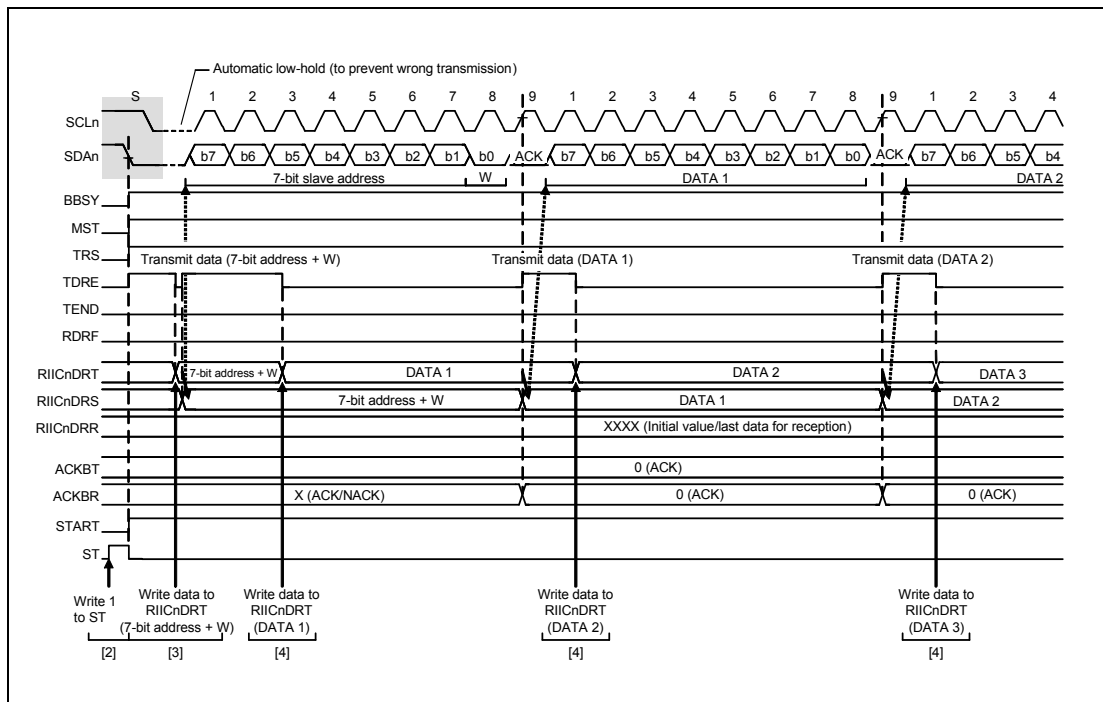


Figure 18.7 Master Transmit Operation Timing (1) (7-Bit Address Format)

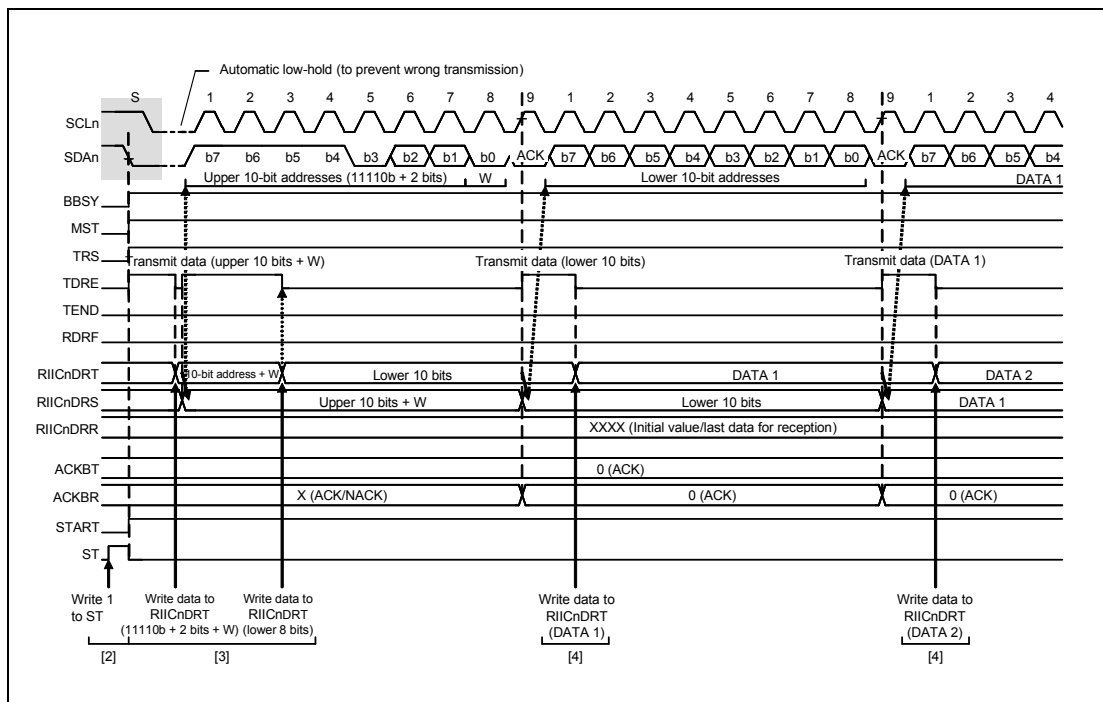


Figure 18.8 Master Transmit Operation Timing (2) (10-Bit Address Format)

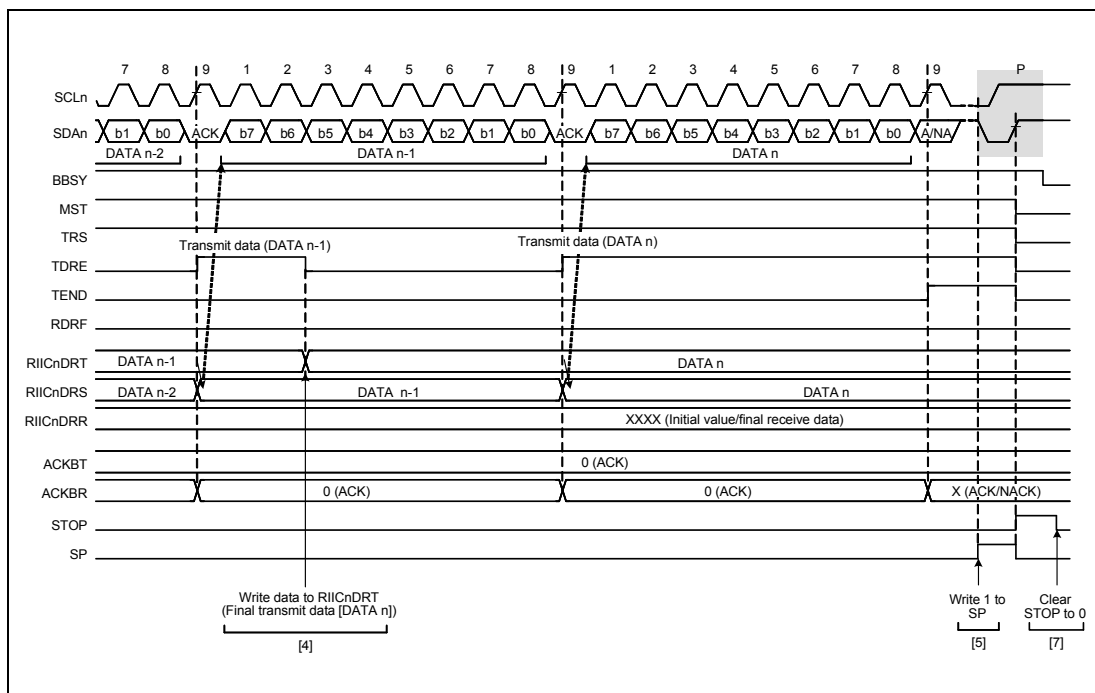


Figure 18.9 Master Transmit Operation Timing (3)

### 18.5.4 Master Receive Operation

In master receive operation, the RIIC as a master device outputs the SCL (clock) signal, receives data from the slave device, and returns acknowledgements. Since the RIIC must start by sending a slave address to the corresponding slave device, this part of the procedure is performed in master transmit mode, but the subsequent steps are in master receive mode.

**Figure 18.10** shows an example of usage of master reception and **Figure 18.11** to **Figure 18.13** show the timing of operations in master reception.

The following describes the procedure and operations for master reception.

- (1) Set the RIICnCR1.IICRST bit to 1 (RIIC reset) and then set the RIICnCR1.ICE bit to 1 (internal reset) with the RIICnCR1.ICE bit cleared to 0 (RIICnSCL and RIICnSDA pins not driven). This initializes the internal state and the various flags of RIICnSR1. After that, set registers RIICnSARy, RIICnSER, RIICnMR1, RIICnBRH, and RIICnBRL (y = 0 to 2), and set the other registers as necessary (for initial settings of the RIIC, see **Figure 18.5**). When the necessary register settings have been completed, set the RIICnCR1.IICRST bit to 0 (for release from the reset state). This step is not necessary if initialization of the RIIC has already been completed.
- (2) Read the RIICnCR2.BBSY flag to check that the bus is open, and then set the RIICnCR2.ST bit to 1 (start condition issuance request). Upon receiving the request, the RIIC issues a start condition. When the RIIC detects the start condition, the BBSY flag and the RIICnSR2.START flag are automatically set to 1 and the ST bit is automatically cleared to 0. At this time, if the start condition is detected and the levels for the SDA output and the levels on the SDA line have matched while the ST bit is 1, the RIIC recognizes that issuing of the start condition as requested by the ST bit has been successfully completed, and the RIICnCR2.MST and TRS bits are automatically set to 1, placing the RIIC in master transmit mode. The RIICnSR2.TDRE flag is also automatically set to 1 in response to setting of the TRS bit to 1.
- (3) Check that the RIICnSR2.TDRE flag is 1, and then write the value for transmission (the first byte indicates the slave address and value of the R/W# bit) to RIICnDRT. Once the data for transmission are written to RIICnDRT, the TDRE flag is automatically cleared to 0, the data are transferred from RIICnDRT to RIICnDRS, and the TDRE flag is again set to 1. Once the byte containing the slave address and R/W# bit has been transmitted, the value of the RIICnCR2.TRS bit is automatically updated to select transmit or receive mode in accord with the value of the transmitted R/W# bit. If the value of the R/W# bit was 1, the RIICnCR2.TRS bit is cleared to 0 on the rising edge of the ninth cycle of SCL (the clock signal), placing the RIIC in master receive mode. At this time, the TDRE flag is automatically cleared to 0 and the RIICnSR2.RDRF flag is automatically set to 1.

Since the RIICnSR2.NACKF flag being 1 at this time indicates that no slave device recognized the address or there was an error in communications, write 1 to the RIICnCR2.SP bit to issue a stop condition.

For master reception from a device with a 10-bit address, start by using master transmission to issue the 10-bit address, and then issue a restart condition. After that, transmitting 1111 0b, the two higher-order bits of the slave address, and the R bit places the RIIC in master receive mode.

- (4) Dummy read RIICnDRR after confirming that the RIICnSR2.RDRF flag is 1; this makes the RIIC start output of the SCL (clock) signal and start data reception.
- (5) After 1 byte of data has been received, the RIICnSR2.RDRF flag is set to 1 on the rising edge of the eighth or ninth cycle of SCL clock (the clock signal) as selected by the RIICnMR3.RDRFS bit. Reading out RIICnDRR at this time will produce the received data, and the RDRF flag is automatically cleared to 0 at the same time. Furthermore, the value of the acknowledgement field

received during the ninth cycle of SCL clock is returned as the value set in the RIICnMR3.ACKBT bit. Furthermore, if the next byte to be received is the next to last byte, set the RIICnMR3.WAIT bit to 1 (for wait insertion) before reading the RIICnDRR (containing the second byte from last). As well as enabling NACK output even in the case of delays in processing to set the RIICnMR3.ACKBT bit to 1 (NACK) in step (6), due to other interrupts, etc., this fixes the SCL line to the low level on the rising edge of the ninth clock cycle in reception of the last byte, so the state is such that issuing a stop condition is possible.

- (6) When the RIICnMR3.RDRFS bit is 0 and the slave device must be notified that it is to end transfer for data reception after transfer of the next (final) byte, set the RIICnMR3.ACKBT bit to 1 (NACK).
- (7) After reading out the byte before last from the RIICnDRR register, if the value of the RIICnSR2.RDRF flag is confirmed to be 1, write 1 to the RIICnCR2.SP bit (stop condition issuance request) and then read the last byte from RIICnDRR. When RIICnDRR is read, the RIIC is released from the wait state and issues the stop condition after low-level output in the ninth clock cycle is completed or the SCL line is released from the low-hold state.
- (8) Upon detecting the stop condition, the RIIC automatically clears the RIICnCR2.MST and TRS bits to 00b and enters slave receive mode. Furthermore, detection of the stop condition leads to setting of the RIICnSR2.STOP flag to 1.
- (9) After checking that the RIICnSR2.STOP flag is 1, clear the RIICnSR2.NACKF and STOP flags to 0 for the next transfer operation.

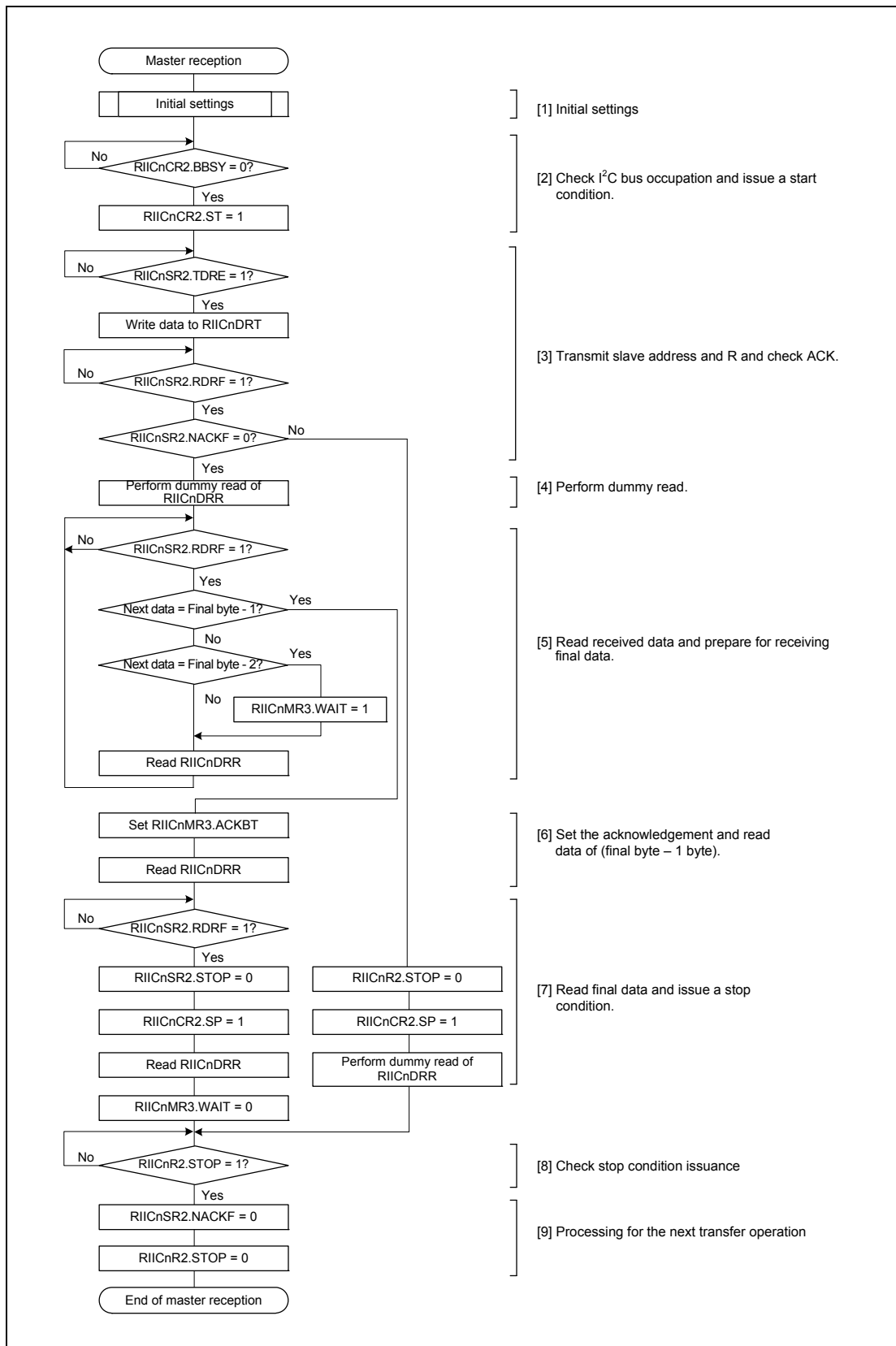


Figure 18.10 Example of Master Reception Flowchart (7-Bit Address Format)

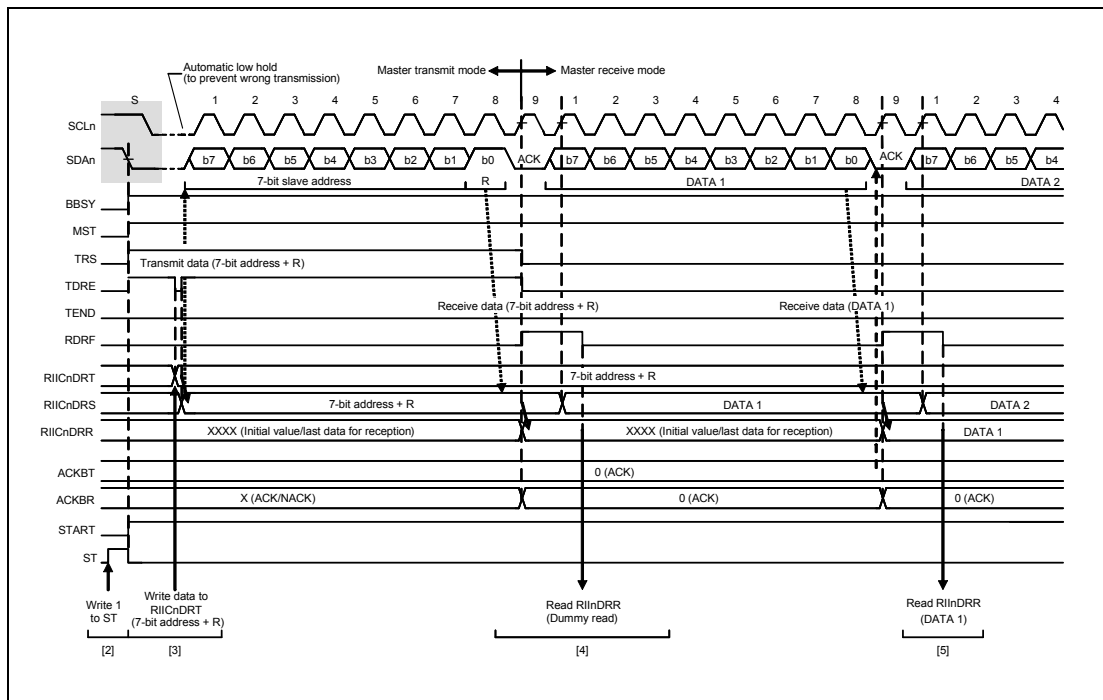


Figure 18.11 Master Receive Operation Timing (1) (7-Bit Address Format, when RDRFS = 0)

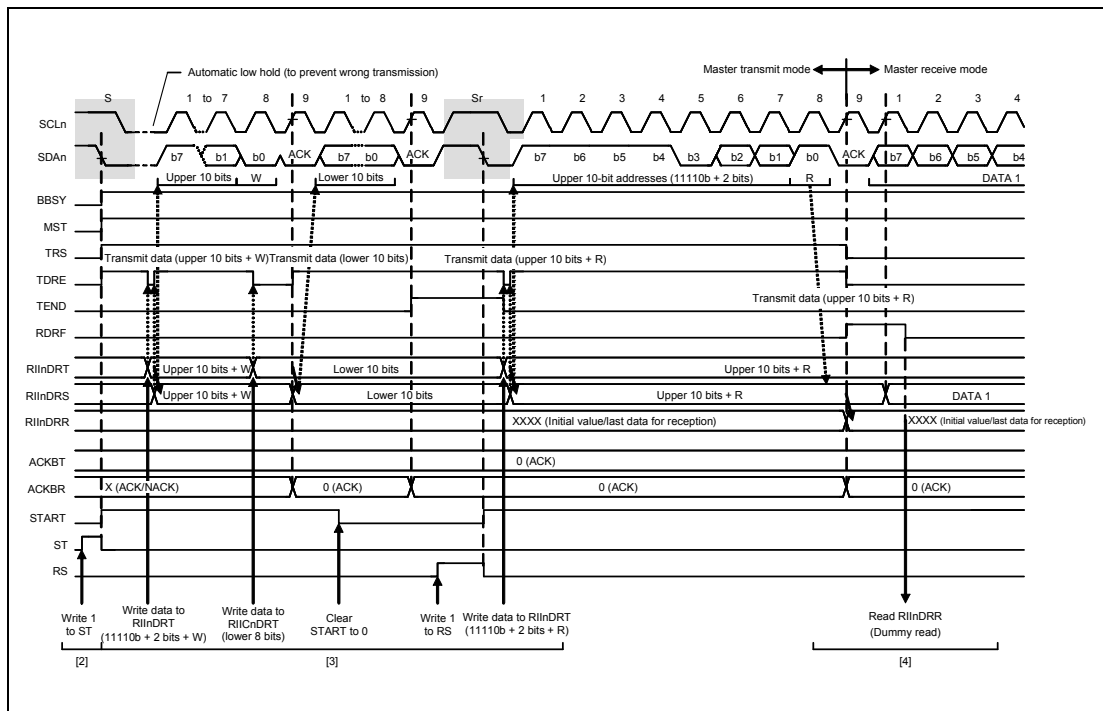


Figure 18.12 Master Receive Operation Timing (2) (10-Bit Address Format, when RDRFS = 0)

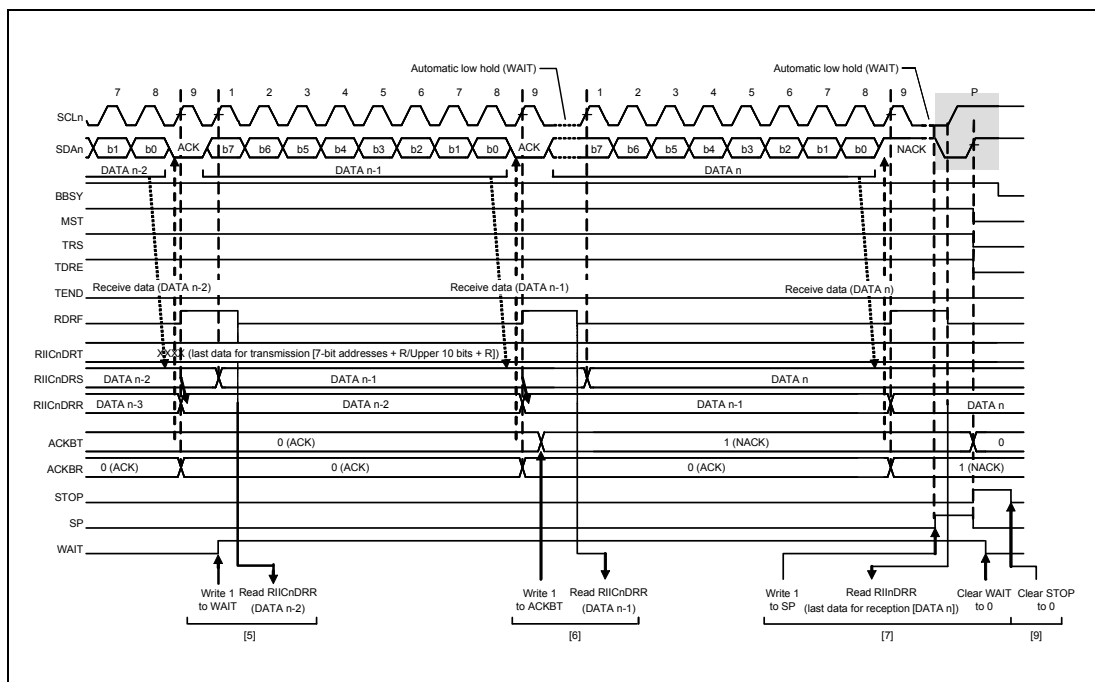


Figure 18.13 Master Receive Operation Timing (3) (when RDRFS = 0)



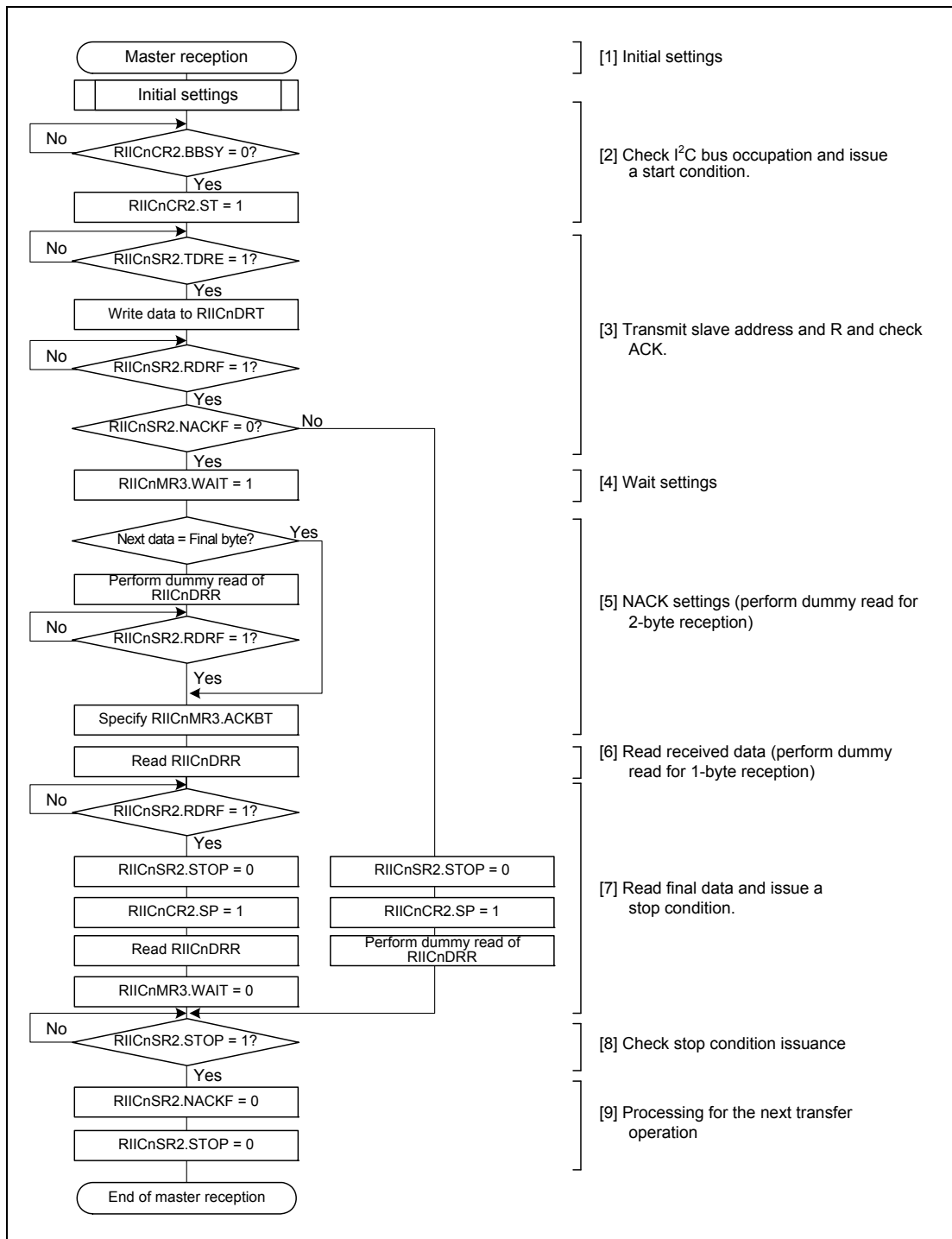


Figure 18.14 Example of 1-Byte/2-Byte Master Reception Flowchart (7-Bit Address Format)

### 18.5.5 Slave Transmit Operation

In slave transmit operation, the master device outputs the SCL (clock) signal, the RIIC transmits data as a slave device, and the master device returns acknowledgements.

**Figure 18.15** shows an example of usage of slave transmission and **Figure 18.16** and **Figure 18.17** show the timing of operations in slave transmission.

The following describes the procedure and operations for slave transmission.

- (1) Follow the procedure in **Figure 18.5** to make initial settings for the RIIC. This step is not necessary if initialization of the RIIC has already been completed. After initial settings, the RIIC will stay in the standby state until it receives a slave address that it matches.
- (2) After receiving a matching slave address, the RIIC sets one of the corresponding bits RIICnSR1.HOA, GCA, and AASy (y = 0 to 2) to 1 on the rising edge of the ninth cycle of SCL clock (the clock signal) and outputs the value set in the RIICnMR3.ACKBT bit to the acknowledge bit on the ninth cycle of SCL clock. If the value of the R/W# bit that was also received at this time is 1, the RIIC automatically places itself in slave transmit mode by setting both the RIICnCR2.TRS bit and the RIICnSR2.TDRE flag to 1.
- (3) After the RIICnSR2.TEND flag is confirmed to be 1, write the data for transmission to the RIICnDRT register. At this time, if the RIIC receives no acknowledge from the master device (receives an NACK signal) while the RIICnFER.NACKF bit is 1, the RIIC suspends transfer of the next data.
- (4) Wait until the RIICnSR2.TEND flag is set to 1 while the RIICnSR2.TDRE flag is 1, after the RIICnSR2.NACKF flag is set to 1 or the last byte for transmission is written to the RIICnDRT register. When the RIICnSR2.NACKF flag or the TEND flag is 1, the RIIC drives the SCL line low on the ninth falling edge of SCL clock.
- (5) When the RIICnSR2.NACKF flag or the RIICnSR2.TEND flag is 1, dummy read RIICnDRR to complete the processing. This releases the SCL line.
- (6) Upon detecting the stop condition, the RIIC automatically clears bits RIICnSR1.HOA, GCA, and AASy (y = 0 to 2), flags RIICnSR2.TDRE and TEND, and the RIICnCR2.TRS bit to 0, and enters slave receive mode.
- (7) After checking that the RIICnSR2.STOP flag is 1, clear the RIICnSR2.NACKF and STOP flags to 0 for the next transfer operation.

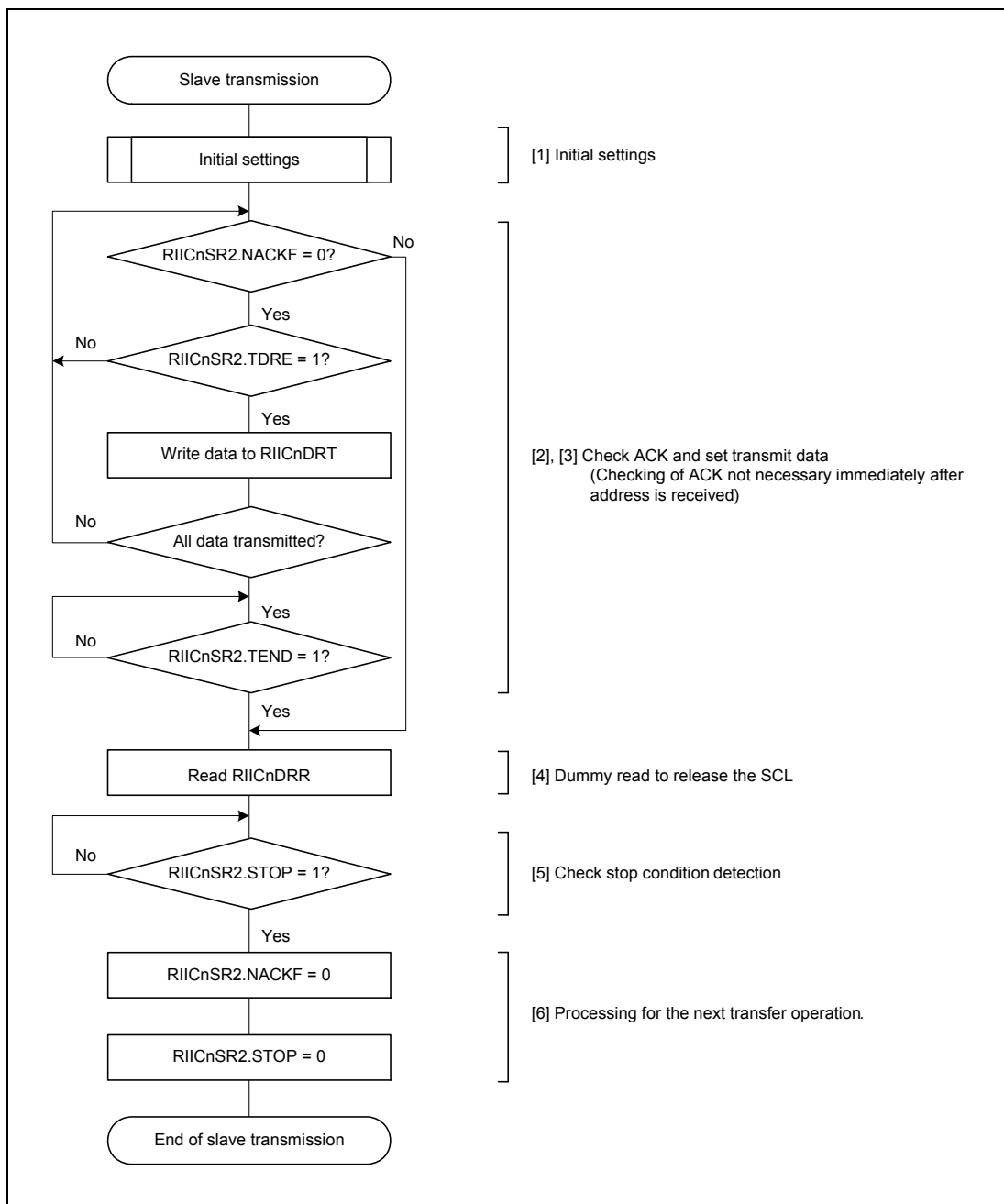


Figure 18.15 Example of Slave Transmission Flowchart

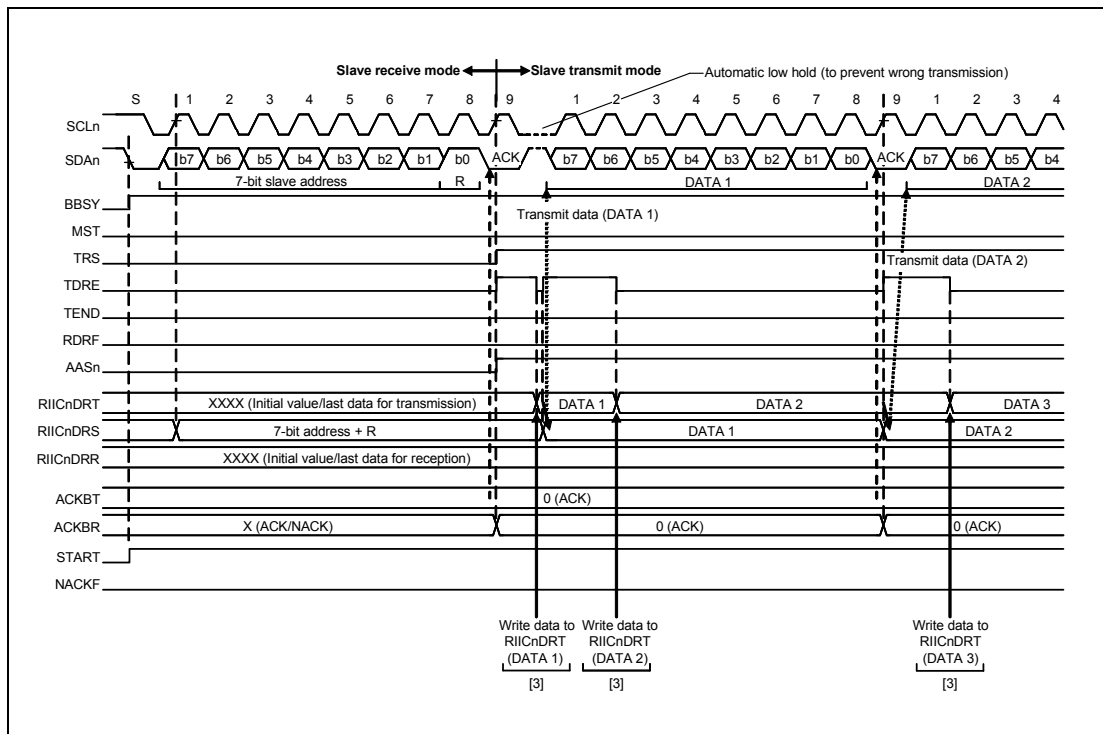


Figure 18.16 Slave Transmit Operation Timing (1) (7-Bit Address Format)

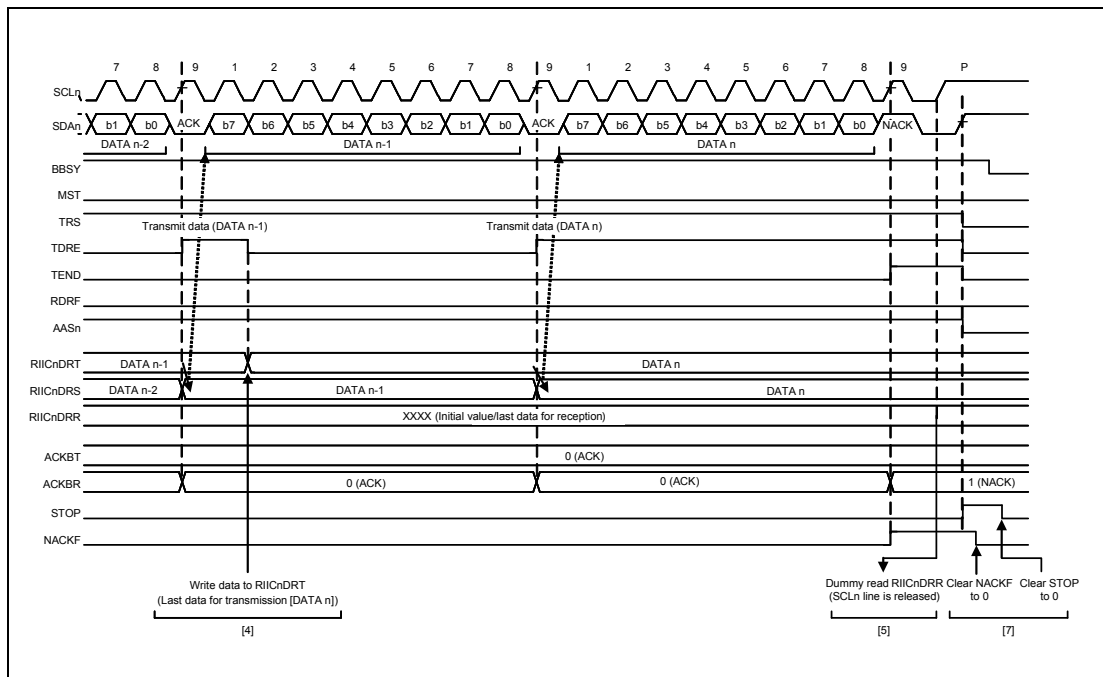


Figure 18.17 Slave Transmit Operation Timing (2)

### 18.5.6 Slave Receive Operation

In slave receive operation, the master device outputs the SCL clock and transmit data, and the RIIC returns acknowledgements as a slave device.

**Figure 18.18** shows an example of usage of slave reception and **Figure 18.19** and **Figure 18.20** show the timing of operations in slave reception.

The following describes the procedure and operations for slave reception.

- (1) Follow the procedure in **Figure 18.5** to make initial settings for the RIIC. This step is not necessary if initialization of the RIIC has already been completed. After initial settings, the RIIC will stay in the standby state until it receives a slave address that it matches.
- (2) After receiving a matching slave address, the RIIC sets one of the corresponding bits RIICnSR1.HOA, GCA, and AASy (y = 0 to 2) to 1 on the rising edge of the ninth cycle of SCL clock (the clock signal) and outputs the value set in the RIICnMR3.ACKBT bit to the acknowledge bit on the ninth cycle of SCL clock. If the value of the R/W# bit that was also received at this time is 0, the RIIC continues to place itself in slave receive mode and sets the RIICnSR2.RDRF flag to 1.
- (3) After the RIICnSR2.STOP flag is confirmed to be 0 and the RIICnSR2.RDRF flag to be 1, dummy read RIICnDRR (the dummy value consists of the slave address and R/W# bit when the 7-bit address format is selected, or the lower eight bits when the 10-bit address format is selected).
- (4) When RIICnDRR is read, the RIIC automatically clears the RIICnSR2.RDRF flag to 0. If reading of RIICnDRR is delayed and a next byte is received while the RDRF flag is still set to 1, the RIIC holds the SCL line low from one SCL cycle before the timing with which RDRF should be set. In this case, reading RIICnDRR releases the SCL line from being held at the low level. When the RIICnSR2.STOP flag is 1 and the RIICnSR2.RDRF flag is also 1, read RIICnDRR until all the data is completely received.
- (5) Upon detecting the stop condition, the RIIC automatically clears bits RIICnSR1.HOA, GCA, and AASy (y = 0 to 2) to 0.
- (6) After checking that the RIICnSR2.STOP flag is 1, clear the RIICnSR2.STOP flag to 0 for the next transfer operation.

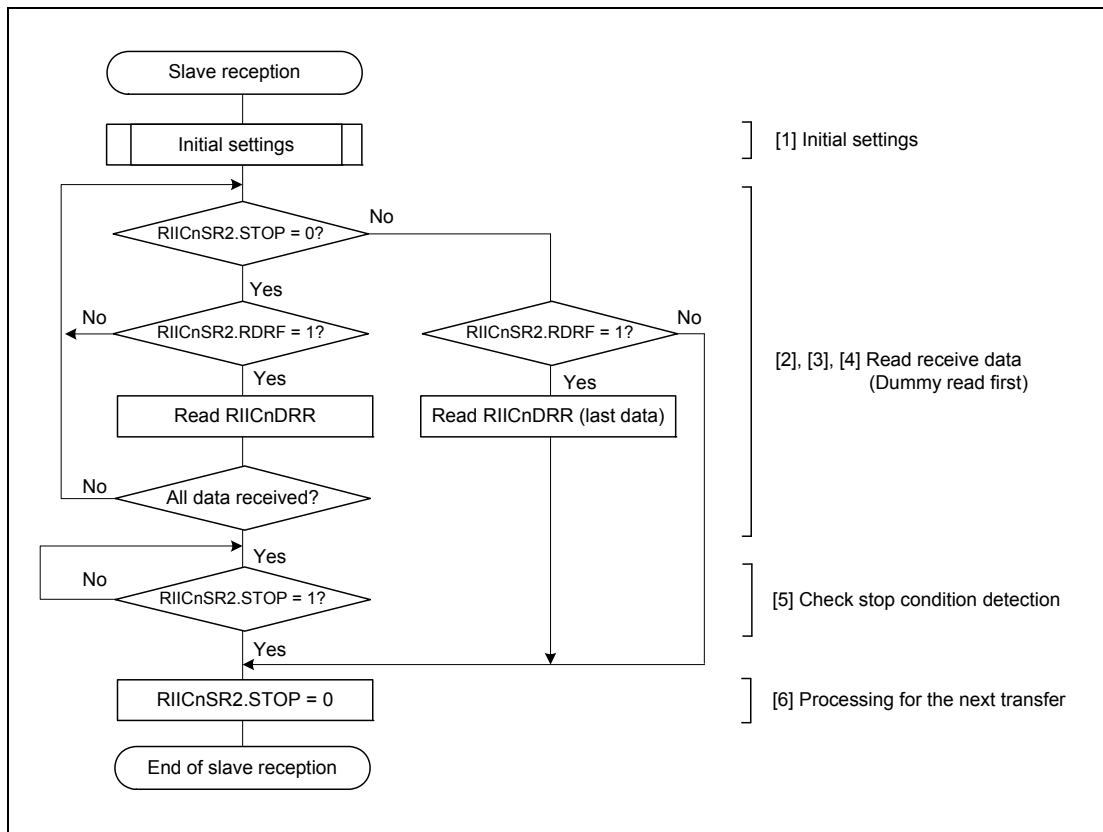


Figure 18.18 Example of Slave Reception Flowchart

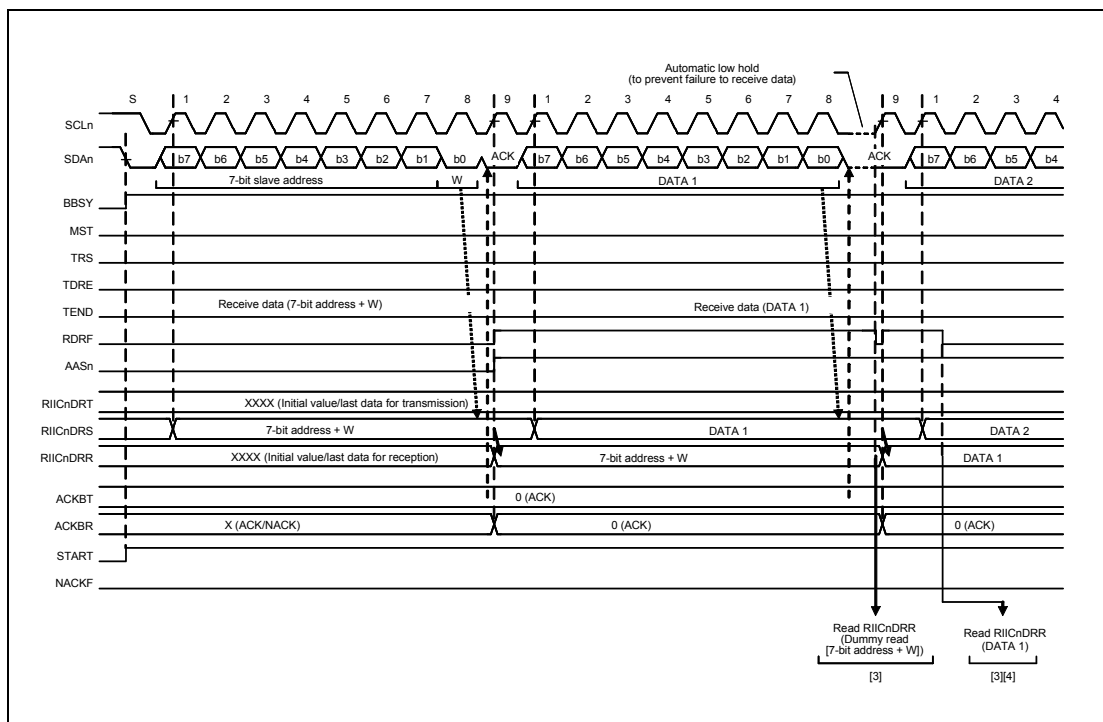


Figure 18.19 Slave Receive Operation Timing (1) (7-Bit Address Format, when RDRFS = 0)

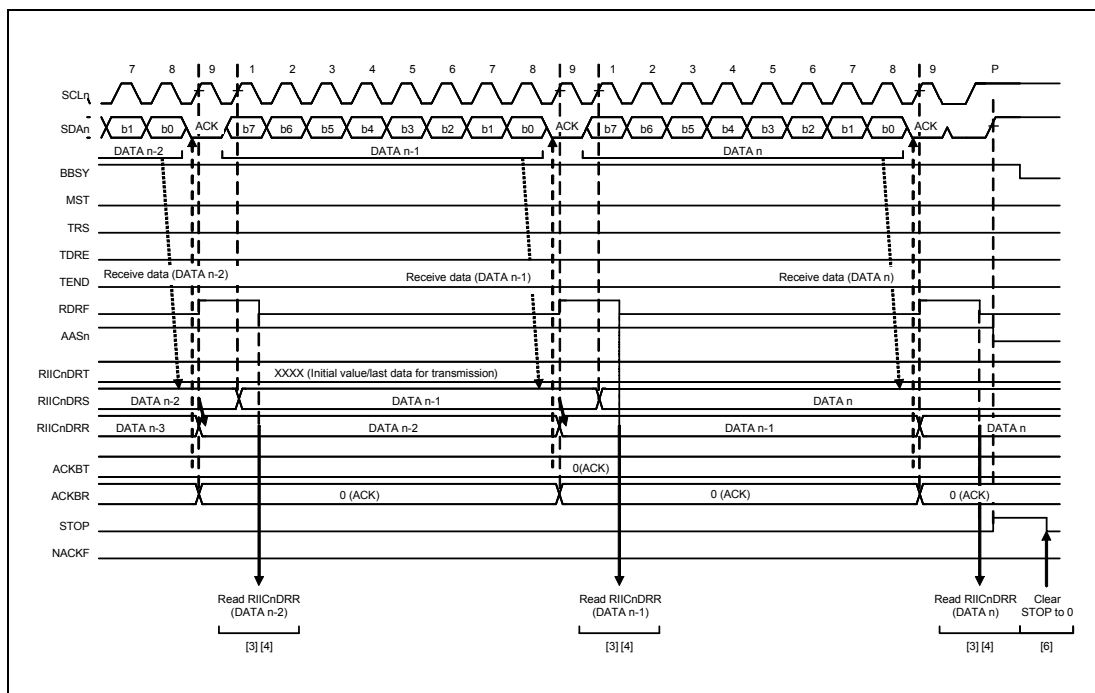


Figure 18.20 Slave Receive Operation Timing (2) (when RDRFS = 0)

## 18.6 SCL Synchronization Circuit

In generation of the SCL (clock) signal, the RIIC starts counting out the value for width at high level specified in RIICnBRH when it detects a rising edge on the SCL line and drives the SCL line low once counting of the width at high level is complete. When the RIIC detects the falling edge of the SCL line, it starts counting out the width at low level period specified in RIICnBRL, and then stops driving the SCL line (releases the line) once counting of the width at low level is complete. The SCL (clock) signal is thus generated.

If multiple master devices are connected to the I<sup>2</sup>C bus, a collision of SCL signals may arise due to contention with another master device. In such cases, the master devices have to synchronize their SCL signals. Since this synchronization of SCL signals must be bit by bit, the RIIC is equipped with a facility (the SCL synchronization circuit) to obtain bit-by-bit synchronization of the SCL clock signals by monitoring the SCL line while in master mode.

When the RIIC has detected a rising edge on the SCL line and thus started counting out the width at high level specified in RIICnBRH, and the level on the SCL line falls because an SCL signal is being generated by another master device, the RIIC stops counting when it detects the falling edge, drives the level on the SCL line low, and starts counting out the width at low level specified in RIICnBRL. When the RIIC finishes counting out the width at low level, it stops driving the SCL line to the low level (i.e. releases the line). At this time, if the width at low level of the SCL clock signal from the other master device is longer than the width at low level set in the RIIC, the width at low level of the SCL signal will be extended. Once the width at low level for the other master device has ended, the SCL signal rises because the SCL line has been released. When the RIIC finishes outputting the low-level period of the SCL clock, the SCL line is released and the SCL clock rises. That is, in cases of contention of SCL signals from more than one master, the width at high level of the SCL signal is synchronized with that of the clock having the narrower width, and the width at low level of the SCL signal is synchronized with that of the clock having the broader width. However, such synchronization of the SCL signal is only enabled when the RIICnFER.SCLE bit is set to 1.

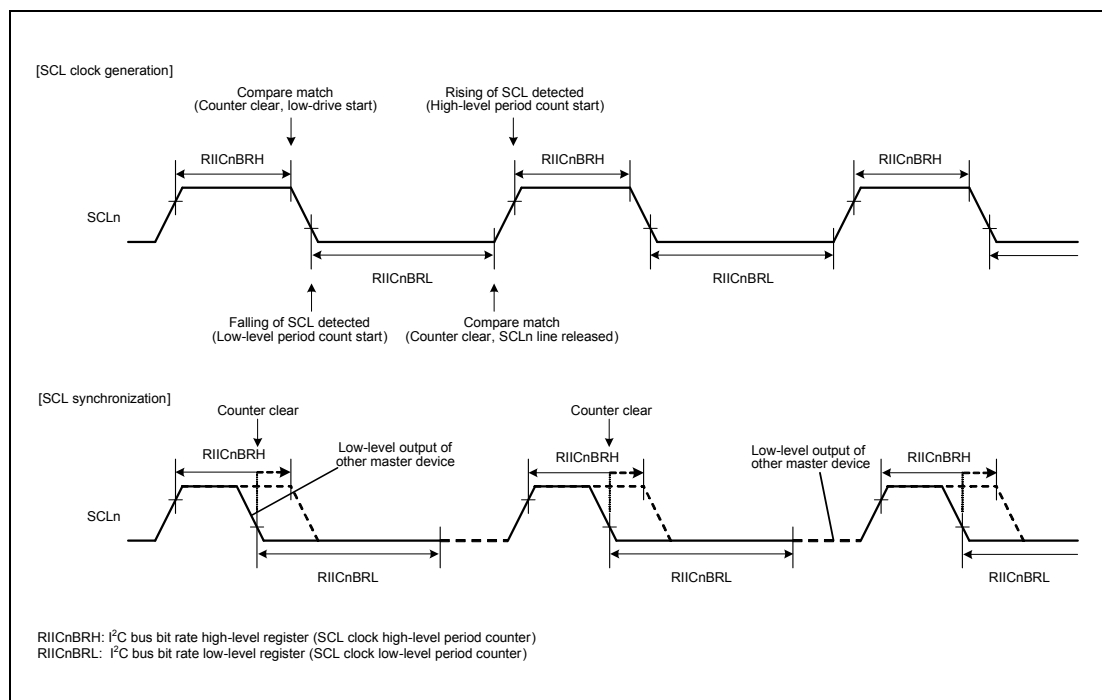


Figure 18.21 Generation and Synchronization of the SCL Signal from the RIIC



## 18.7 Facility for Delaying SDA Output

The RIIC module incorporates a facility for delaying output on the SDA line. The delay can be applied to all output (issuing of the start, restart, and stop conditions, data, and the ACK and NACK signals) on the SDA line.

With the SDA output delay facility, SDA output is delayed from detection of a falling edge of the SCL signal to ensure that the SDA signal is output within the interval over which the SCL (clock) signal is at the low level. Doing this leads to usage with the aim of preventing erroneous operation of communications devices, with the aim of satisfying the 300-ns (min.) data-hold time requirement of the SMBus specification.

The output delay facility is enabled by setting the RIICnMR2.SDDL[2:0] bits to any value other than 000b, and disabled by setting the same bits to 000b.

While the SDA output delay facility is enabled (i.e. while the SDDL[2:0] bits in IMCR2 are set to any value other than 000b), the RIICnMR2.DLCS bit selects the clock source for counting by the SDA output delay counter as the internal base clock (IIC $\phi$ ) for the RIIC module or as a clock signal derived by dividing the frequency of the internal base clock by two (IIC $\phi$ /2). The counter counts the number of cycles set in the SDDL[2:0] bits in IMCR2. After counting of the set number of cycles of delay is completed, the RIIC module places the required output (start, restart, or stop condition, data, or an ACK or NACK signal) on the SDA line.

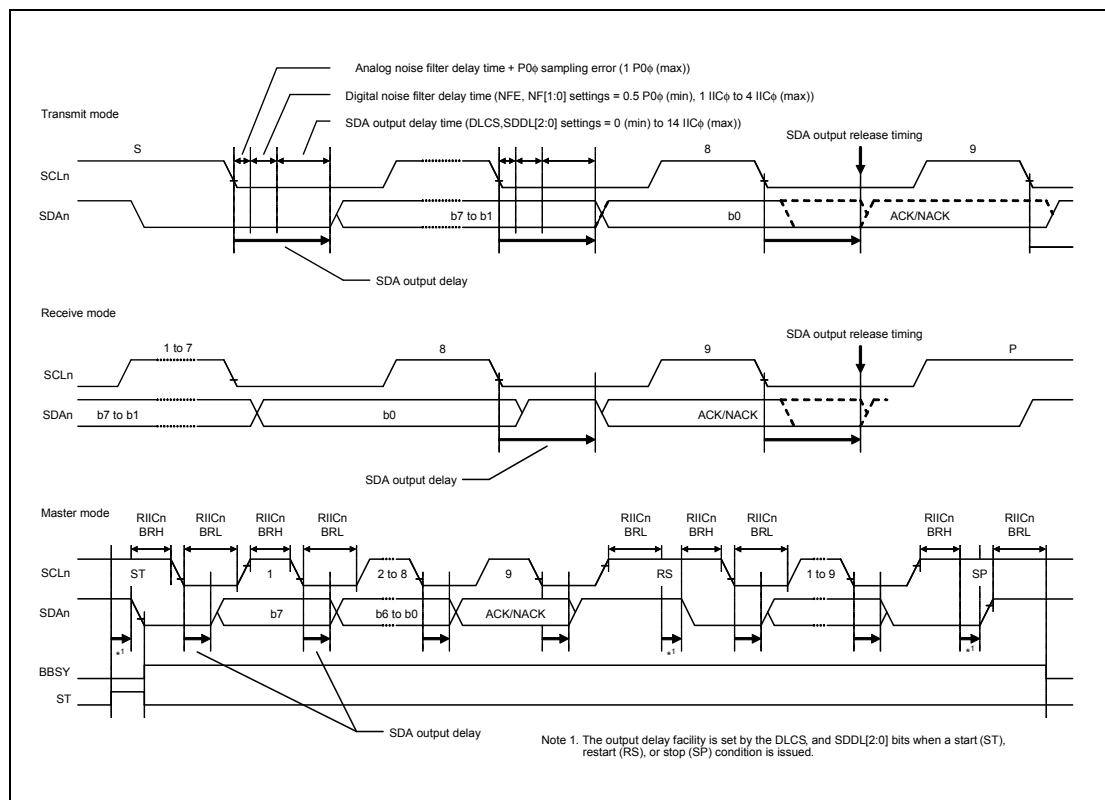


Figure 18.22 SDA Output Delay Facility

## 18.8 Digital Noise-Filter Circuits

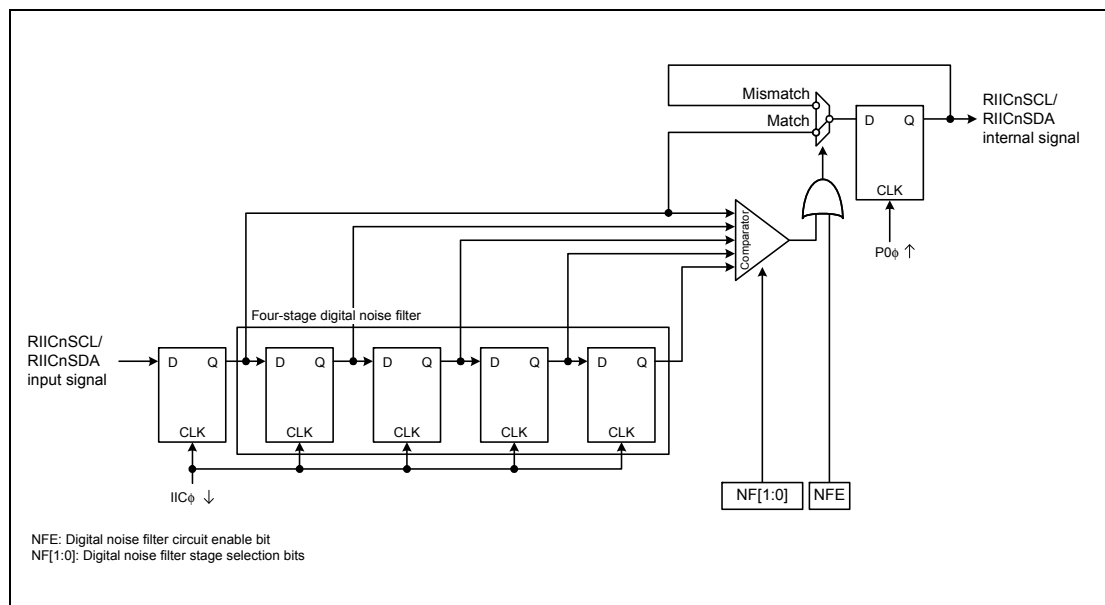
The states of the RIICnSCL and RIICnSDA pins are conveyed to the internal circuitry through a digital noise-filter circuit when the NFE bit in RIICnFER is set to 1. **Figure 18.23** is a block diagram of the digital noise-filter circuit.

The on-chip digital noise-filter circuit of the RIIC consists of four flip-flop circuit stages connected in series and a match-detection circuit.

The number of effective stages in the digital noise filter is selected by the RIICnMR3.NF[1:0] bits. The selected number of effective stages determines the noise-filtering capability as a period from one to four IIC $\phi$  cycles.

The input signal to the RIICnSCL pin (or RIICnSDA pin) is sampled on falling edges of the IIC $\phi$  signal. When the input signal level matches the output level of the number of effective flip-flop circuit stages as selected by the RIICnMR3.NF[1:0] bits, the signal level is conveyed to the subsequent stage. If the signal levels do not match, the previous value is retained.

If the ratio between P0 $\phi$  and IIC $\phi$  is small when the RIICnMR1.CKS[2:0] bits are set to 000b, note that the characteristics of the digital noise filter may lead to the elimination of needed signals as noise.



**Figure 18.23** Block Diagram of Digital Noise Filter Circuit

## 18.9 Address Match Detection

The RIIC can set three unique slave addresses in addition to the general call address and host address, and also can set 7-bit or 10-bit slave addresses.

### 18.9.1 Slave-Address Match Detection

The RIIC can set three unique slave addresses, and has a slave address detection function for each unique slave address. When the RIICnSER.SARyE bit ( $y = 0$  to 2) is set to 1, the slave addresses set in RIICnSARy ( $y = 0$  to 2) can be detected.

When the RIIC detects a match of the set slave address, the corresponding RIICnSR1.AASy flag ( $y = 0$  to 2) is set to 1 at the falling edge of the ninth SCL clock cycle, and the RIICnSR2.RDRF flag or the RIICnSR2.TDRE flag is set to 1 by the following R/W# bit. This causes a receive data full interrupt (INTRIICRI) or transmit data empty interrupt (INTRIICTI) to be generated. The AASy flag is used to identify which slave address has been specified.

Figure 18.24 to Figure 18.26 show the AASy flag set timing in three cases.

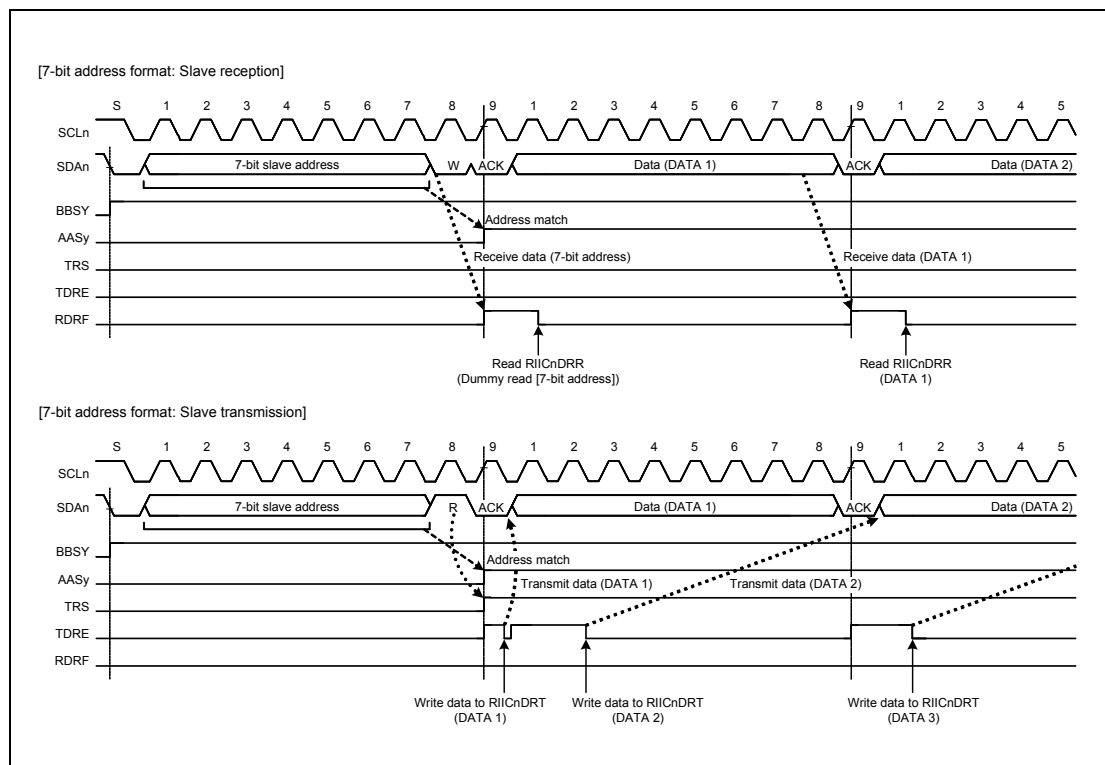


Figure 18.24 AASy Flag Set Timing with 7-Bit Address Format Selected

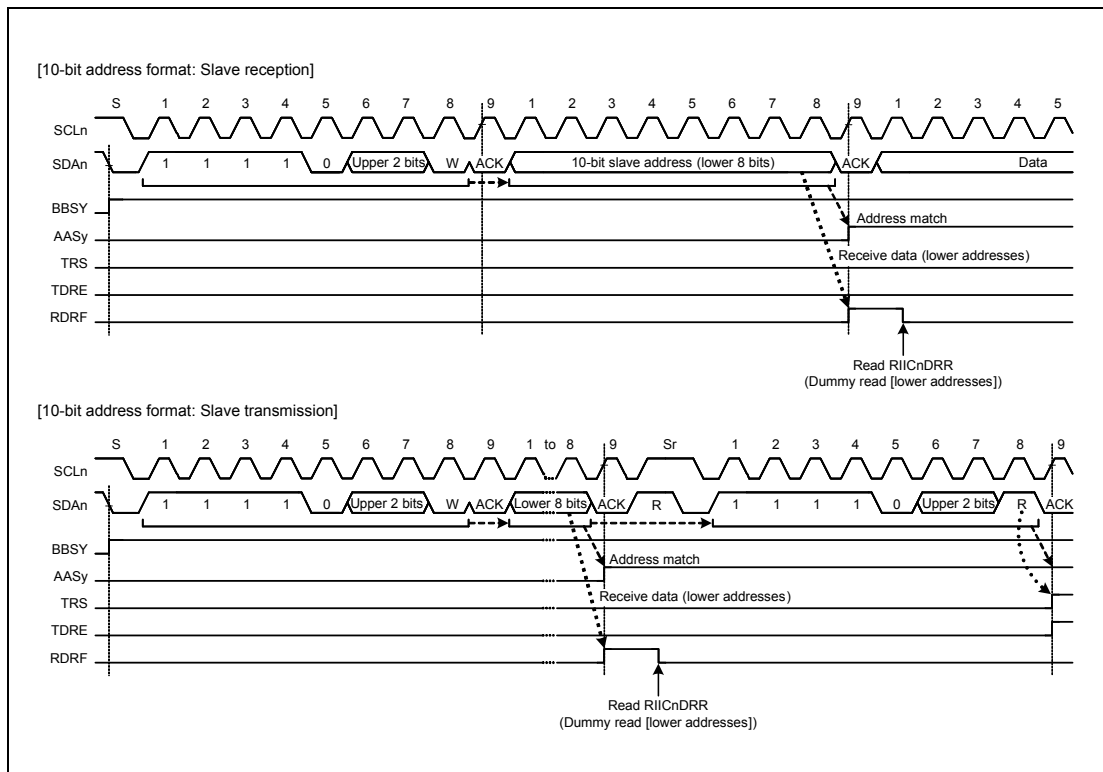


Figure 18.25 AASy Flag Set Timing with 10-Bit Address Format Selected

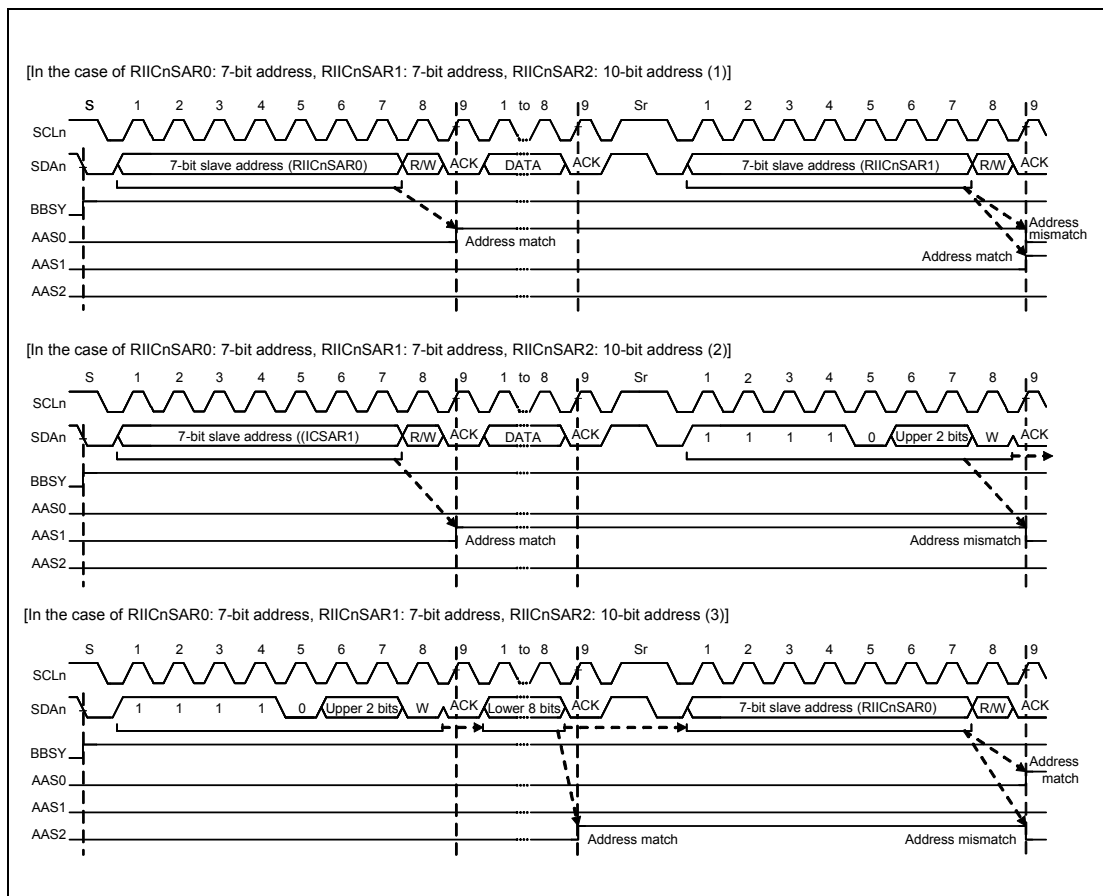


Figure 18.26 AASy Flag Set/Clear Timing with 7-Bit/10-Bit Address Formats Mixed

## 18.9.2 Detection of the General Call Address

The RIIC has a facility for detecting the general call address (0000 000b + 0 [W]). This is enabled by setting the RIICnSER.GCE bit to 1.

If the address received after a start or restart condition is issued is 0000 000b + 1[R] (start byte), the RIIC recognizes this as the address of a slave device with an “all-zero” address but not as the general call address.

When the RIIC detects the general call address, both the RIICnSR1.GCA flag and the RIICnSR2.RDRF flag are set to 1 on the falling edge of the ninth cycle of SCL clock. This leads to the generation of a receive data full interrupt (INTRIICRI). The value of the GCA flag can be confirmed to recognize that the general call address has been transmitted.

Operation after detection of the general call address is the same as normal slave receive operation.

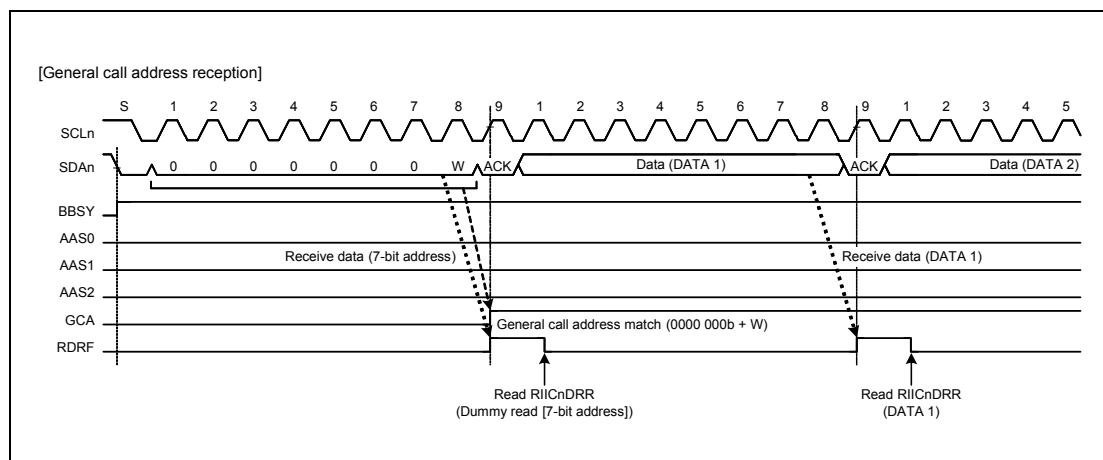


Figure 18.27 Timing of GCA Flag Setting during Reception of General Call Address

### 18.9.3 Device-ID Address Detection

The RIIC module has a facility for detecting device-ID addresses conformant with the I<sup>2</sup>C bus specification (Rev. 03). When the RIIC receives 1111 100b as the first byte after a start condition or restart condition was issued with the RIICnSER.DIDE bit set to 1, the RIIC recognizes the address as a device ID, sets the RIICnSR1.DID flag to 1 on the rising edge of the ninth SCL clock cycle when the following R/W# bit is 0, and then compares the second and subsequent bytes with its own slave address. If the address matches the value in the slave address register, the RIIC sets the corresponding RIICnSR1.AASy flag (y = 0 to 2) to 1.

After that, when the first byte received after a start or restart condition is issued matches the device ID address (1111 100b) again and the following R/W# bit is 1, the RIIC does not compare the second and subsequent bytes and sets the RIICnSR2.TDRE flag to 1.

In the device-ID address detection function, the RIIC clears the DID flag to 0 if a match with the RIIC's own slave address is not obtained or a match with the device ID address is not obtained after a match with the RIIC's own slave address and the detection of a restart condition. If the first byte after detection of a start or restart condition matches the device ID address (1111 100b) and the R/W# bit is 0, the RIIC sets the DID flag to 1 and compares the second and subsequent bytes with the RIIC's slave address. If the R/W# bit is 1, the DID flag holds the previous value and the RIIC does not compare the second and subsequent bytes. Therefore, the reception of a device-ID address can be checked by reading the DID flag after confirming that TDRE = 1.

Furthermore, prepare the device-ID fields (three bytes: 12 bits indicating the manufacturer + 9 bits identifying the part + 3 bits indicating the revision) that must be sent to the host after reception of a continuous device-ID field as normal data for transmission. For details of the information that must be included in device-ID fields, contact NXP Semiconductors.

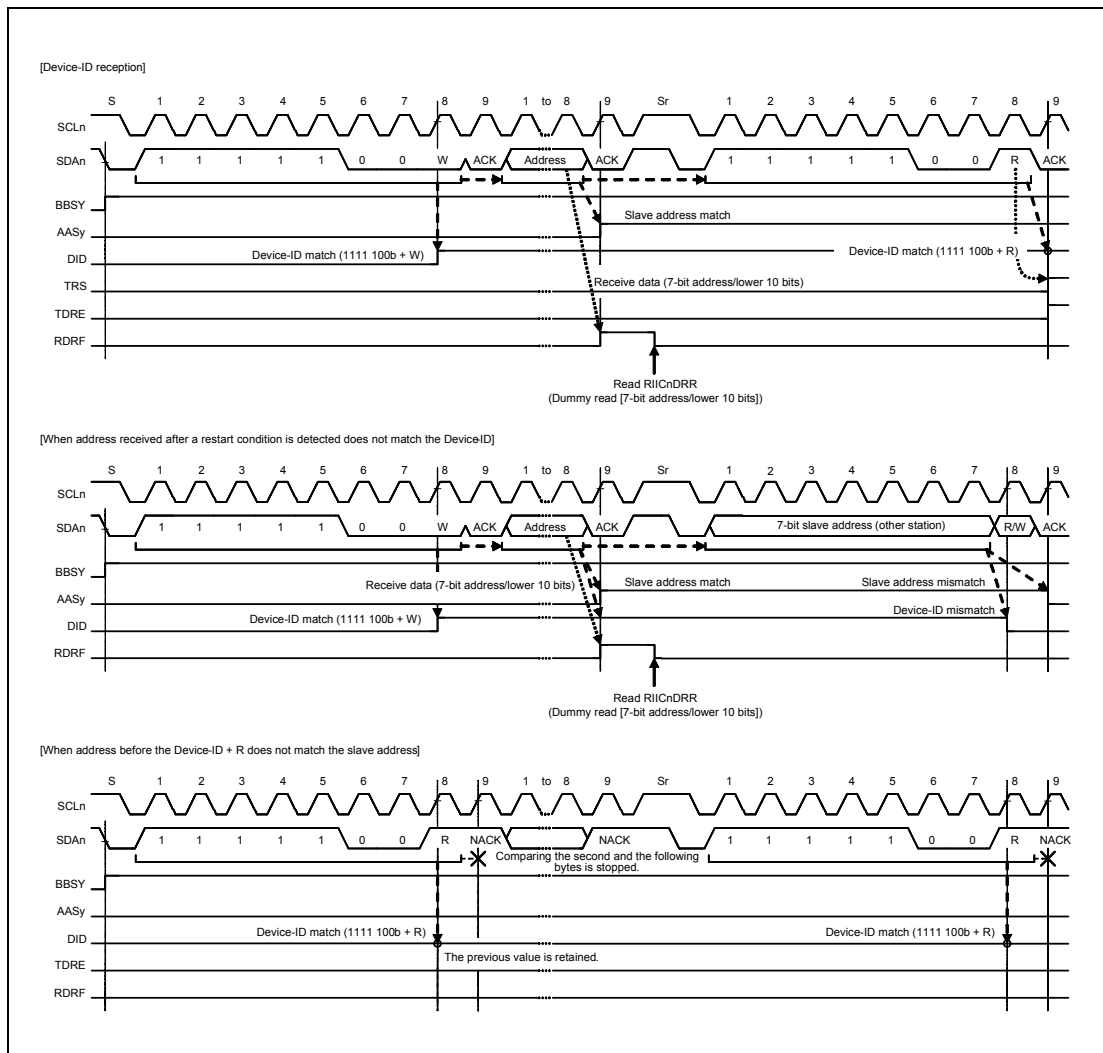


Figure 18.28 AASy/DID Flag Set/Clear Timing during Reception of Device-ID

### 18.9.4 Host Address Detection

The RIIC has a function to detect the host address while the SMBus is operating. When the RIICnSER.HOAE bit is set to 1 while the RIICnMR3.SMBS bit is 1, the RIIC can detect the host address (0001 000b) in slave receive mode (RIICnCR2.MST and TRS bits = 00b).

When the RIIC detects the host address, the RIICnSR1.HOA flag is set to 1 at the falling edge of the ninth SCL clock cycle, and at the same time, the RIICnSR2.TDRE flag is set to 1 when the R/W# bit is 0 (Wr bit). This causes a transmit data empty interrupt (INTRIICTI) to be generated. The HOA flag is used to recognize that the host address was sent from the smart battery or other devices.

If the bit following the host address (0001 000b) is an Rd bit (R/W# bit = 1), the RIIC can also detect the host address. After the host address is detected, the RIIC operates in the same manner as normal slave operation.

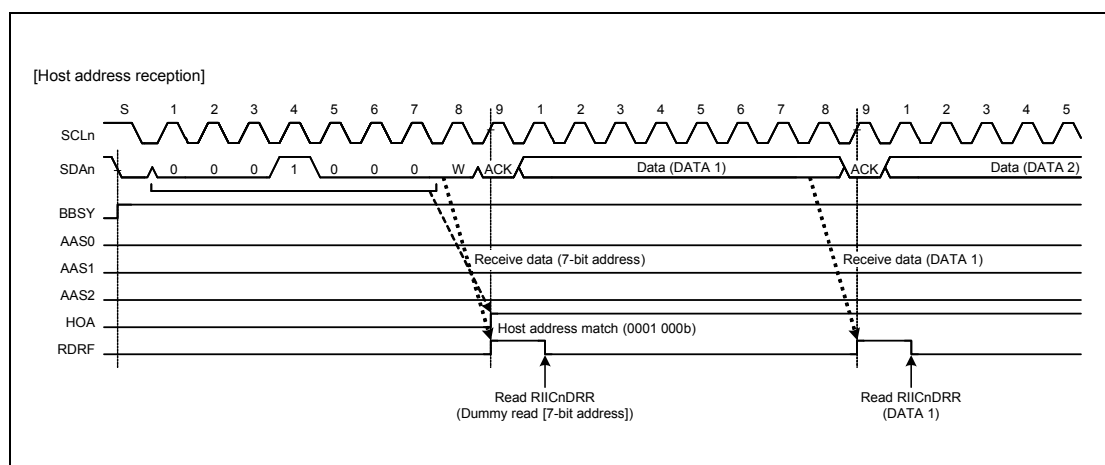


Figure 18.29 HOA Flag Set Timing during Reception of Host Address



## 18.10 Automatically Low-Hold Function for SCL

### 18.10.1 Function to Prevent Wrong Transmission of Transmit Data

If the shift register (RIICnDRS) is empty when data have not been written to the transmit data register (RIICnDRT) with the RIIC in transmission mode (RIICnCR2.TRS bit = 1), the SCL signal is automatically held at the low level over the intervals shown below. This low-hold period is extended until data for transmission have been written, which prevents the unintended transmission of erroneous data.

#### <Master transmit mode>

- Low-level interval after a start condition or restart condition is issued
- Low-level interval of one clock cycle between the ninth clock cycle of one transfer and the first clock cycle of the next

#### <Slave transmit mode>

- Low-level interval between the ninth clock cycle of one transfer and the first clock cycle of the next

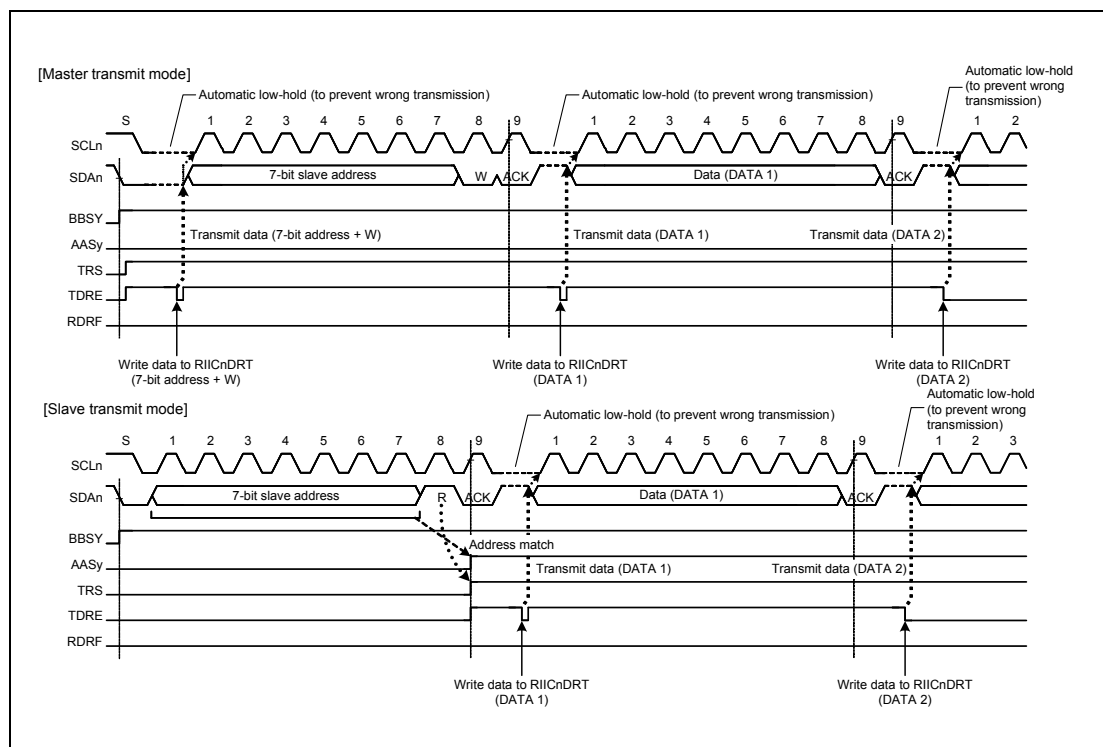


Figure 18.30 Automatic Low-Hold Operation in Transmit Mode

### 18.10.2 NACK Reception Transfer Suspension Function

The RIIC has a function to suspend transfer operation when NACK is received in transmit mode (RIICnCR2.TRIS bit = 1). This function is enabled when the RIICnFER.NACKIE bit is set to 1 (transfer suspension enabled). If the next transmit data has already been written (RIICnSR2.TDRE flag = 0) when NACK is received, next data transmission at the falling edge of the ninth SCL clock cycle is automatically suspended. This prevents the SDA line output level from being held low when the MSB of the next transmit data is 0.

If the transfer operation is suspended by this function (RIICnSR2.NACKF flag = 1), transmit operation and receive operation are discontinued. To restore transmit/receive operation, be sure to clear the NACKF flag to 0. In master transmit mode, clear the NACKF flag to 0, issue a restart or stop condition, and then issue a start condition again.

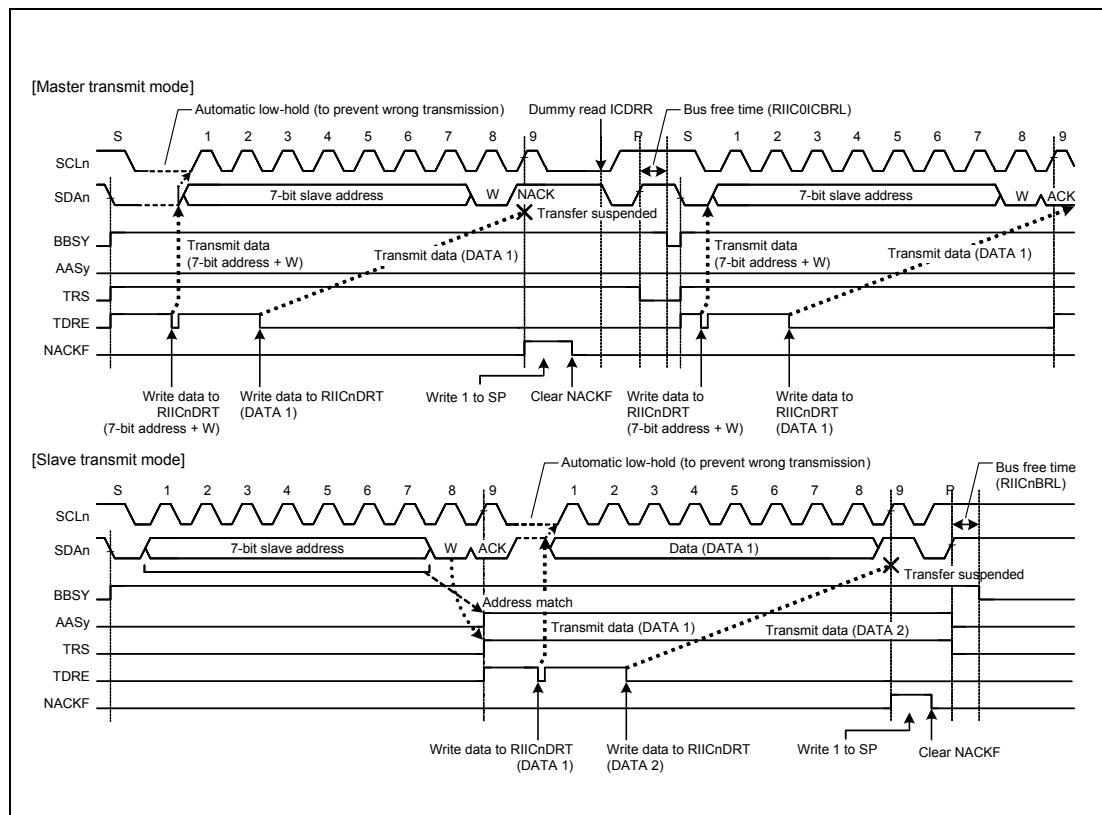


Figure 18.31 Suspension of Data Transfer when NACK is Received (NACKIE = 1)

### 18.10.3 Function to Prevent Failure to Receive Data

If response processing is delayed when receive data (RIICnDRR) read is delayed for a period of one transfer frame or more with receive data full (RIICnSR2.RDRF flag = 1) in receive mode (RIICnCR2.TRS = 0), the RIIC holds the SCL line low automatically immediately before the next data is received to prevent failure to receive data.

This function to prevent failure to receive data using the automatic low-hold function is also enabled even if the read processing of the final receive data is delayed and, in the meantime, the RIIC's own slave address is designated after a stop condition is issued. This function does not disturb other communication because the RIIC does not hold the SCL line low when a mismatch with its own slave address occurs after a stop condition is issued.

Sections in which the SCL line is held low can be selected with a combination of the RIICnMR3.WAIT and RDRFS bits.

#### (1) One-Byte Receive Operation and Automatic Low-Hold Function Using the WAIT Bit

When the RIICnMR3.WAIT bit is set to 1, the RIIC performs one-byte receive operation using the WAIT bit function.

Furthermore, when the RIICnMR3.RDRFS bit is 0, the RIIC automatically sends the RIICnMR3.ACKBT bit value for the acknowledge bit in the period from the falling edge of the eighth SCL clock cycle to the falling edge of the ninth SCL clock cycle, and automatically holds the SCL line low at the falling edge of the ninth SCL clock cycle using the WAIT bit function. This low-hold is released by reading data from RIICnDRR, which enables bitwise receive operation.

The WAIT bit function is enabled for receive frames after a match with the RIIC's own slave address (including the general call address and host address) is obtained in master receive mode or slave receive mode.

#### (2) One-Byte Receive Operation (ACK/NACK Transmission Control) and Automatic Low-Hold Function Using the RDRFS Bit

When the RIICnMR3.RDRFS bit is set to 1, the RIIC performs one-byte receive operation using the RDRFS bit function.

When the RIICnSR2.RDRFS bit is set to 1, the RDRF flag (receive data full) in RIICnSR2 is set to 1 at the rising edge of the eighth SCL clock cycle, and the SCL line is automatically held low at the falling edge of the eighth SCL clock cycle. This lowhold is released by writing a value to the RIICnMR3.ACKBT bit, but cannot be released by reading data from RIICnDRR, which enables receive operation by the ACK/NACK transmission control according to the data received in byte units.

The RDRFS bit function is enabled for receive frames after a match with the RIIC's own slave address (including the general call address and host address) is obtained in master receive mode or slave receive mode.

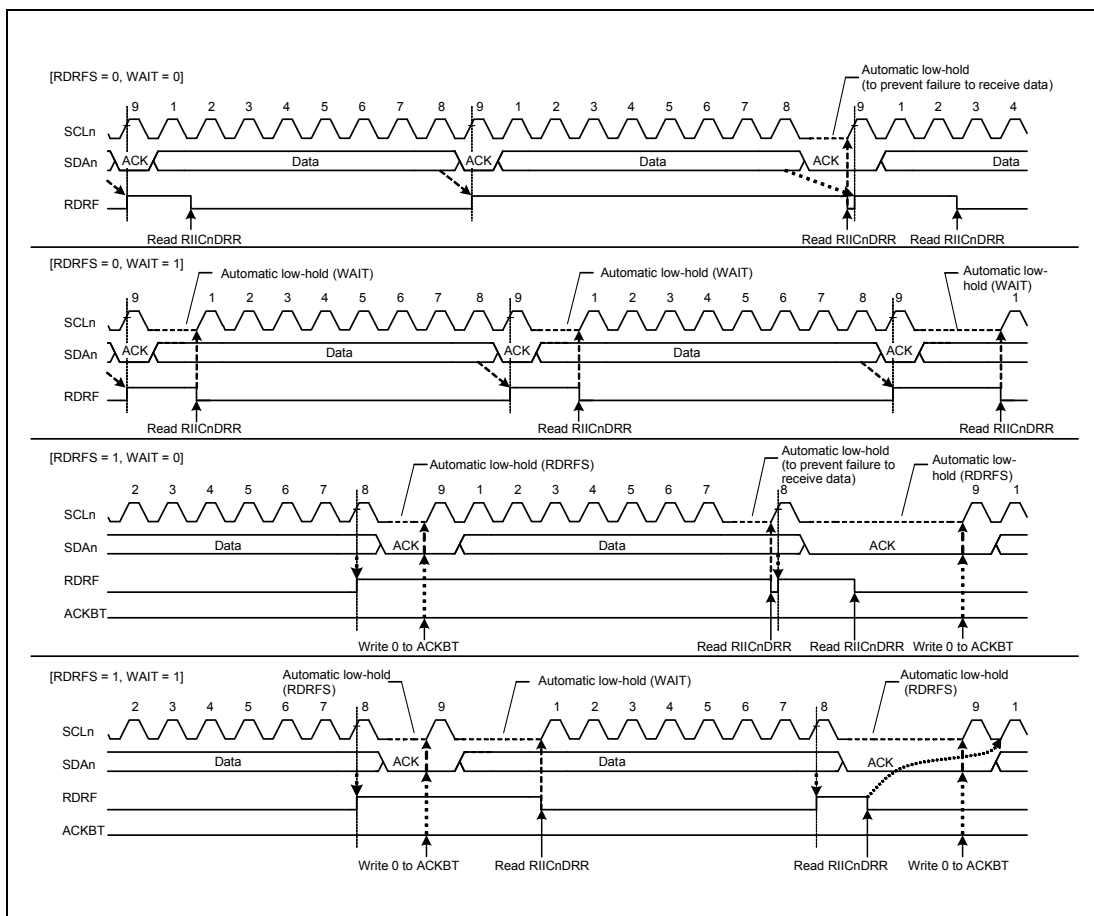


Figure 18.32 Automatic Low-Hold Operation in Receive Mode (Using RDRFS and WAIT Bits)

## 18.11 Arbitration-Lost Detection Functions

In addition to the normal arbitration-lost detection function defined by the I<sup>2</sup>C bus standard, the RIIC has functions to prevent double-issue of a start condition, to detect arbitration-lost during transmission of NACK, and to detect arbitration-lost in slave transmit mode.

### 18.11.1 Master Arbitration-Lost Detection (MALE Bit)

The RIIC drives the SDA line low to issue a start condition. However, if the SDA line has already been driven low by another master device issuing a start condition, the RIIC regards its own issuing of a start condition as an error and considers this a loss in arbitration, so priority is given to transfer by the other master device. Similarly, if a request to issue a start condition is made by setting the RIICnCR2.ST bit to 1 while the bus is busy (RIICnCR2.BBSY flag = 1), the RIIC regards this as a double-issuing-of-start-condition error and considers itself to have lost in arbitration, thus preventing a failure of transfer due to issuing of a start condition while transfer is in progress.

When a start condition is issued successfully, if the data for transmission including the address bits (i.e. the internal SDA output level) and the level on the SDA line do not match (the high output as the internal SDA output; i.e. the SDA pin is in the high-impedance state) and the low level is detected on the SDA line, the RIIC loses in arbitration.

After a loss in arbitration of mastership, the RIIC immediately enters slave receive mode. If a slave address (including the general call address) matches its own address at this time, the RIIC continues in slave operation.

A loss in arbitration of mastership is detected when the following conditions are met while the RIICnFER.MALE bit is 1 (master arbitration-lost detection enabled).

[Master arbitration-lost conditions]

- Non-matching of the internal level for output on SDA and the level on the SDA line after a start condition was issued by setting the RIICnCR2.ST bit to 1 while the RIICnCR2.BBSY flag was cleared to 0 (erroneous issuing of a start condition)
- Setting of the RIICnCR2.ST bit to 1 (start condition double-issue error) while the RIICnCR2.BBSY flag is set to 1
- When the transmit data excluding acknowledge (internal SDA output level) does not match the level on the SDA line in master transmit mode (RIICnCR2.MST and TRS bits = 11b)

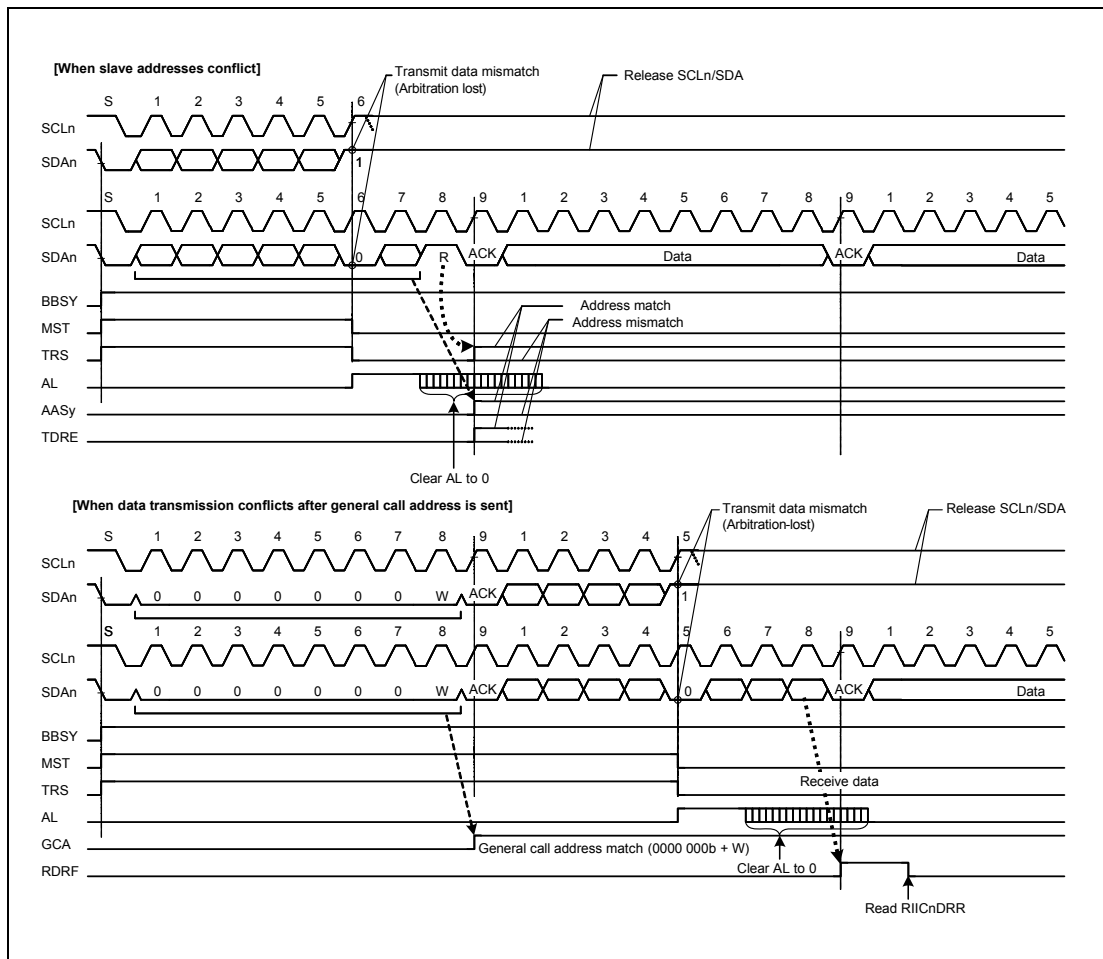


Figure 18.33 Examples of Master Arbitration-Lost Detection (MALE = 1)

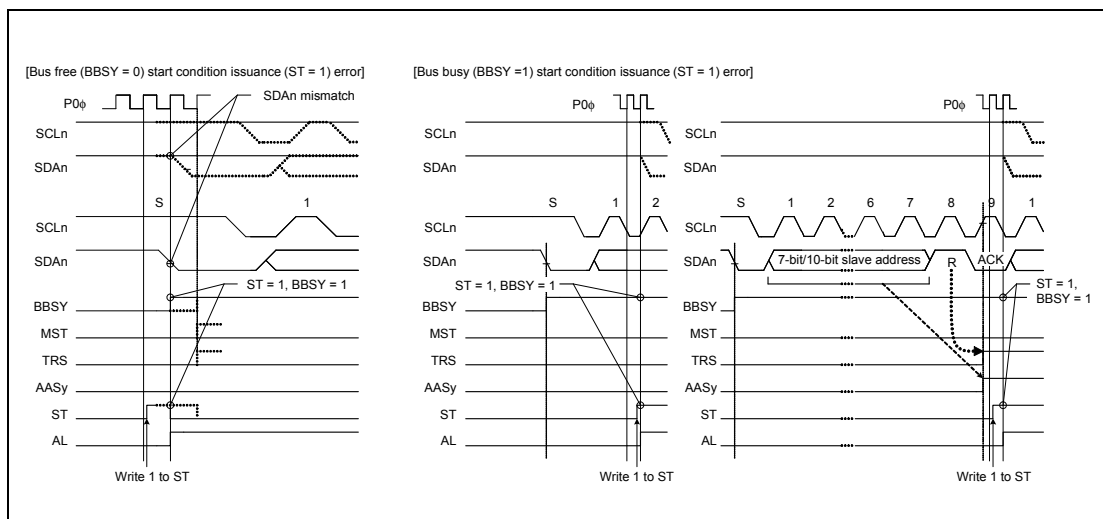
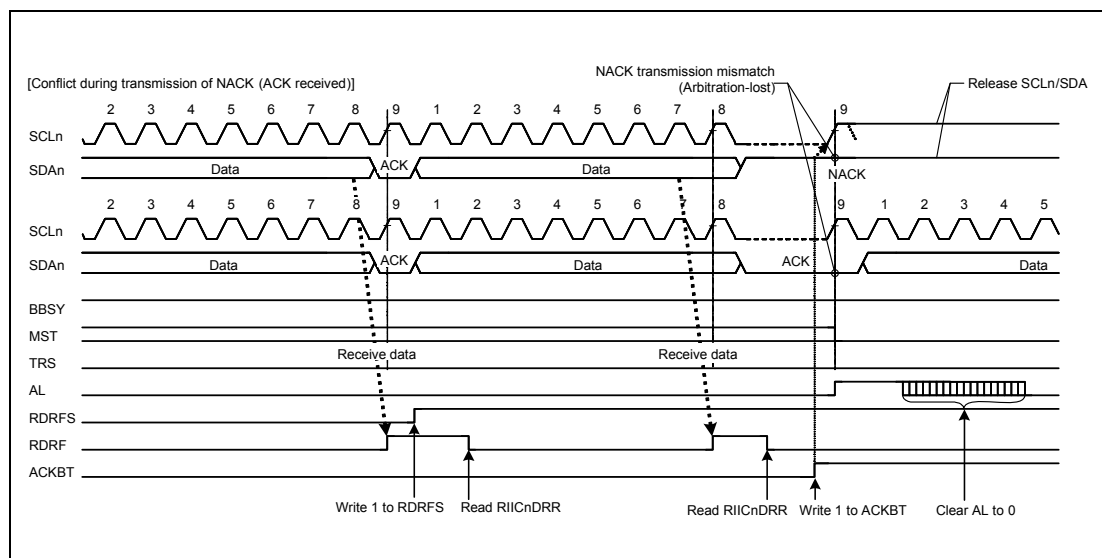


Figure 18.34 Arbitration-Lost when a Start Condition is Issued (MALE = 1)

### 18.11.2 Function to Detect Loss of Arbitration during NACK Transmission (NALE Bit)

The RIIC has a function to cause arbitration to be lost if the internal SDA output level does not match the level on the SDA line (the high output as the internal SDA output; i.e. the SDA pin is in the high-impedance state) and the low level is detected on the SDA line during transmission of NACK in receive mode. Arbitration is lost due to a conflict of NACK transmission and ACK transmission when two or more master devices receive data from the same slave device simultaneously in a multi-master system. Such conflict occurs when multiple master devices send/receive the same information through a single slave device. **Figure 18.35** shows an example of arbitration-lost detection during transmission of NACK.



**Figure 18.35 Example of Arbitration-Lost Detection during Transmission of NACK (NALE = 1)**

The following explains arbitration-lost detection using an example where two master devices (master A and master B) and a single slave device are connected through the bus. In this example, master A receives two bytes of data from the slave device, and master B receives four bytes of data from the slave device.

If master A and master B access the slave device simultaneously, because the slave address is identical, arbitration is not lost in both master A and master B during access to the slave device. Therefore, both master A and master B recognize that they have obtained the bus mastership and operate as such. Here, master A sends NACK when it has received two final bytes of data from the slave device. Meanwhile, master B sends ACK because it has not received necessary four bytes of data. At this time, the NACK transmission from master A and the ACK transmission from master B conflict. In general, if a conflict like this occurs, master A cannot detect ACK transmitted by master B and issues a stop condition.

Therefore, the issuance of the stop condition conflicts with the SCL clock output of master B, which disturbs communication.

When the RIIC receives ACK during transmission of NACK, it detects a defeat in conflict with other master devices and causes arbitration to be lost.

If arbitration is lost during transmission of NACK, the RIIC immediately cancels the slave match condition and enters slave receive mode. This prevents a stop condition from being issued, preventing a communication failure on the bus.

Similarly, in the ARP command processing of SMBus, the function to detect loss of arbitration during transmission of NACK is also available for eliminating the extra clock cycle processing (such as FFh transmission processing) necessary if the UDID (Unique Device Identifier) of assign address does not match in the Get UDID (general) processing after the Assign address command.

The RIIC detects arbitration-lost during transmission of NACK when the following condition is met with the RIICnFER.NALE bit set to 1 (arbitration-lost detection during NACK transmission enabled).

[Condition for arbitration-lost during NACK transmission]

When the internal SDA output level does not match the SDA line (ACK is received) during transmission of NACK (RIICnMR3.ACKBT bit = 1)

### 18.11.3 Slave Arbitration-Lost Detection (SALE Bit)

The RIIC has a function to cause arbitration to be lost if the data for transmission (i.e. the internal SDA output level) and the level on the SDA line do not match (the high output as the internal SDA output; i.e. the SDA pin is in the highimpedance state) and the low level is detected on the SDA line in slave transmit mode. This arbitration-lost detection function is mainly used when transmitting a UDID (Unique Device Identifier) over an SMBus.

When it loses slave arbitration, the RIIC is immediately released from the slave-matched state and enters slave receive mode. This function can detect conflicts of data during transmission of UDIDs over an SMBus and eliminates

subsequent redundant processing (processing for the transmission of FFh).

The RIIC detects slave arbitration-lost when the following condition is met with the RIICnFER.SALE bit set to 1 (slave arbitration-lost detection enabled).

[Condition for slave arbitration-lost]

When transmit data excluding acknowledge (internal SDA output level) does not match the SDA line in slave transmit mode (RIICnCR2.MST and TRS bits = 01b)

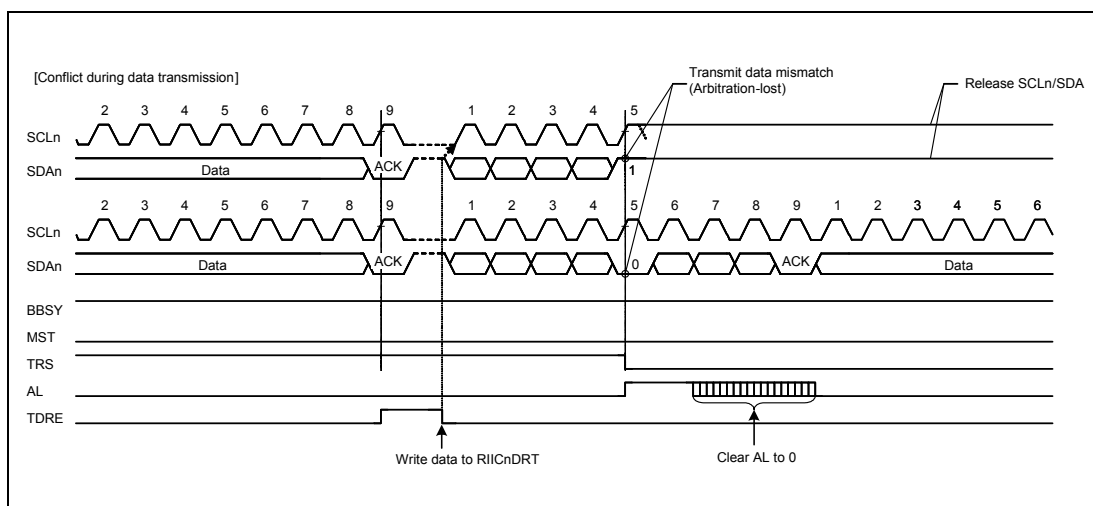


Figure 18.36 Example of Slave Arbitration-Lost Detection (SALE = 1)



## 18.12 Start Condition/Restart Condition/Stop Condition Issuing Function

### 18.12.1 Issuing a Start Condition

The RIIC issues a start condition when the RIICnCR2.ST bit is set to 1.

When the ST bit is set to 1, a start condition issuance request is made and the RIIC issues a start condition when the RIICnCR2.BBSY flag is 0 (bus free). When a start condition is issued normally, the RIIC automatically shifts to the master transmit mode.

A start condition is issued in the following sequence.

[Start condition issuance]

- Drive the SDA line low (high level to low level).
- Ensure the time set in RIICnBRH and the start condition hold time.
- Drive the SCL line low (high level to low level).
- Detect low level of the SCL line and ensure the low-level period of SCL line set in RIICnBRL.

### 18.12.2 Issuing a Restart Condition

The RIIC issues a restart condition when the RIICnCR2.RS bit is set to 1.

When the RS bit is set to 1, a restart condition issuance request is made and the RIIC issues a restart condition when the RIICnCR2.BBSY flag is 1 (bus busy) and the RIICnCR2.MST bit is 1 (master mode).

A restart condition is issued in the following sequence.

[Restart condition issuance]

- Release the SDA line.
- Ensure the low-level period of SCL line set in RIICnBRL.
- Release the SCL line (low level to high level).
- Detect a high level of the SCL line and ensure the time set in RIICnBRL and the restart condition setup time.
- Drive the SDA line low (high level to low level).
- Ensure the time set in RIICnBRH and the restart condition hold time.
- Drive the SCL line low (high level to low level).
- Detect a low level of the SCL line and ensure the low-level period of SCL line set in RIICnBRL.

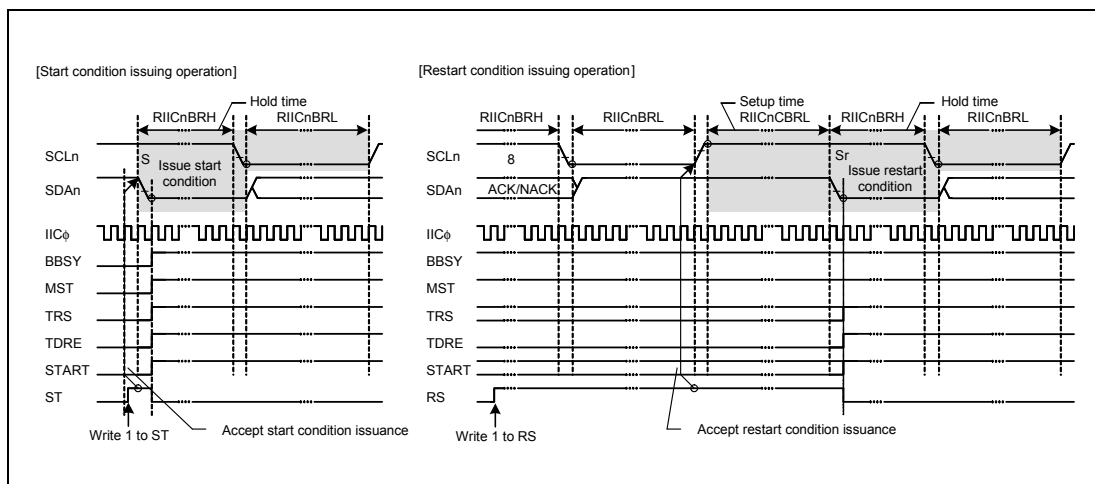


Figure 18.37 Start Condition/Restart Condition Issue Timing (ST and RS Bits)

### 18.12.3 Issuing a Stop Condition

The RIIC issues a stop condition when the RIICnCR2.SP bit is set to 1.

When the SP bit is set to 1, a stop condition issuance request is made and the RIIC issues a stop condition when the RIICnCR2.BBSY flag is 1 (bus busy) and the RIICnCR2.MST bit is 1 (master mode).

A stop condition is issued in the following sequence.

[Stop condition issuance]

- Drive the SDA line low (high level to low level).
- Ensure the low-level period of SCL line set in RIICnBRL.
- Release the SCL line (low level to high level).
- Detect a high level of the SCL line and ensure the time set in RIICnBRH and the stop condition setup time.
- Release the SDA line (low level to high level).
- Ensure the time set in RIICnBRL and the bus free time.
- Clear the BBSY flag to 0 (to release the bus mastership).

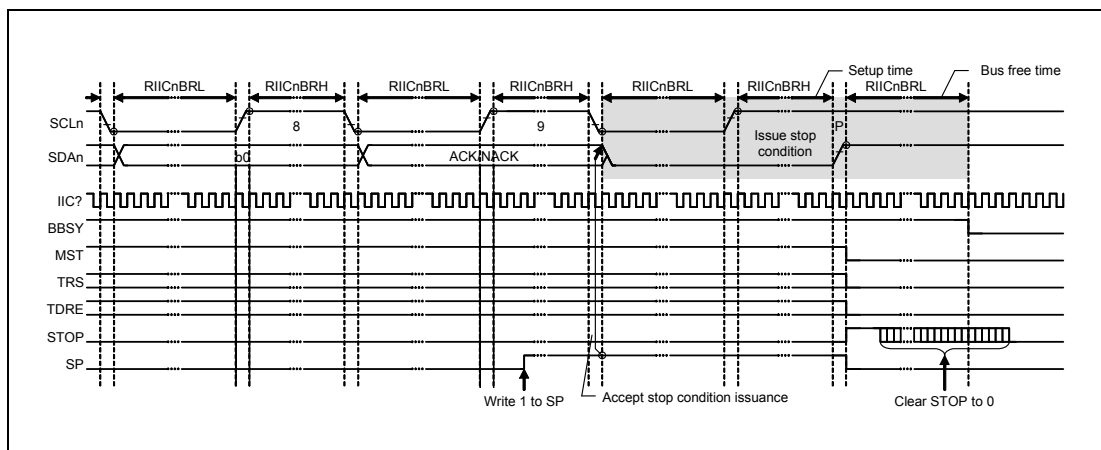


Figure 18.38 Stop Condition Issue Timing (SP Bit)

## 18.13 Bus Hanging

If the clock signals from the master and slave devices go out of synchronization due to noise or other factors, the I<sup>2</sup>C bus might hang with a fixed level on the SCL line and/or SDA line.

As measures against the bus hanging, the RIIC has a timeout function to detect hanging by monitoring the SCL line, a function for the output of an extra SCL clock cycle to release the bus from a hung state due to clock signals being out of synchronization, and the RIIC/internal reset function.

By checking the RIICnCR1.SCLO, SDAO, SCLI, and SDAI bits, it is possible to see whether the RIIC or its partner in communications is placing the low level on the SCL or SDA lines.

### 18.13.1 Timeout Function

The RIIC has the timeout function to detect an abnormality that the SCL line is held for a certain period of time. In the bus busy state, the RIIC can detect an abnormal bus state by monitoring that the SCL line is held low or high for a predetermined time.

The timeout function monitors the SCL line state and counts the low-level period or high-level period using the internal counter. The timeout function resets the internal counter each time the SCL line changes (rising or falling), but continues to count unless the SCL line changes. If the internal counter overflows due to no SCL line change, the RIIC can detect the timeout and report the bus abnormality.

This timeout function is enabled when the RIICnFER.TMOE bit is 1. It detects an abnormal bus state in which the SCLn line is held low or high in the following cases.

1. When the bus is busy (RIICnCR2.BBSY = 1) in master mode (RIICnCR2.MST = 1)
2. When the bus is busy (RIICnCR2.BBSY = 1) and the RIIC's own slave address matches (ICSR1 is not 00h) in slave mode (RIICnCR2.MST = 0)
3. While the bus is free (RIICnCR2.BBSY = 0) and the RIICnCR2.ST bit is 1

The internal counter of the timeout function works using the internal reference clock (IIC $\phi$ ) set by the RIICnMR1.CKS[2:0] bits as a count source. It functions as a 16-bit counter when long mode is selected (RIICnMR2.TMOS bit = 0) or a 14-bit counter when short mode is selected (TMOS bit = 1).

The SCL line level (low/high or both levels) during which this counter is activated can be selected by the setting of the RIICnMR2.TMOH and TMOL bits. If both TMOL and TMOH bits are cleared to 0, the internal counter does not work.

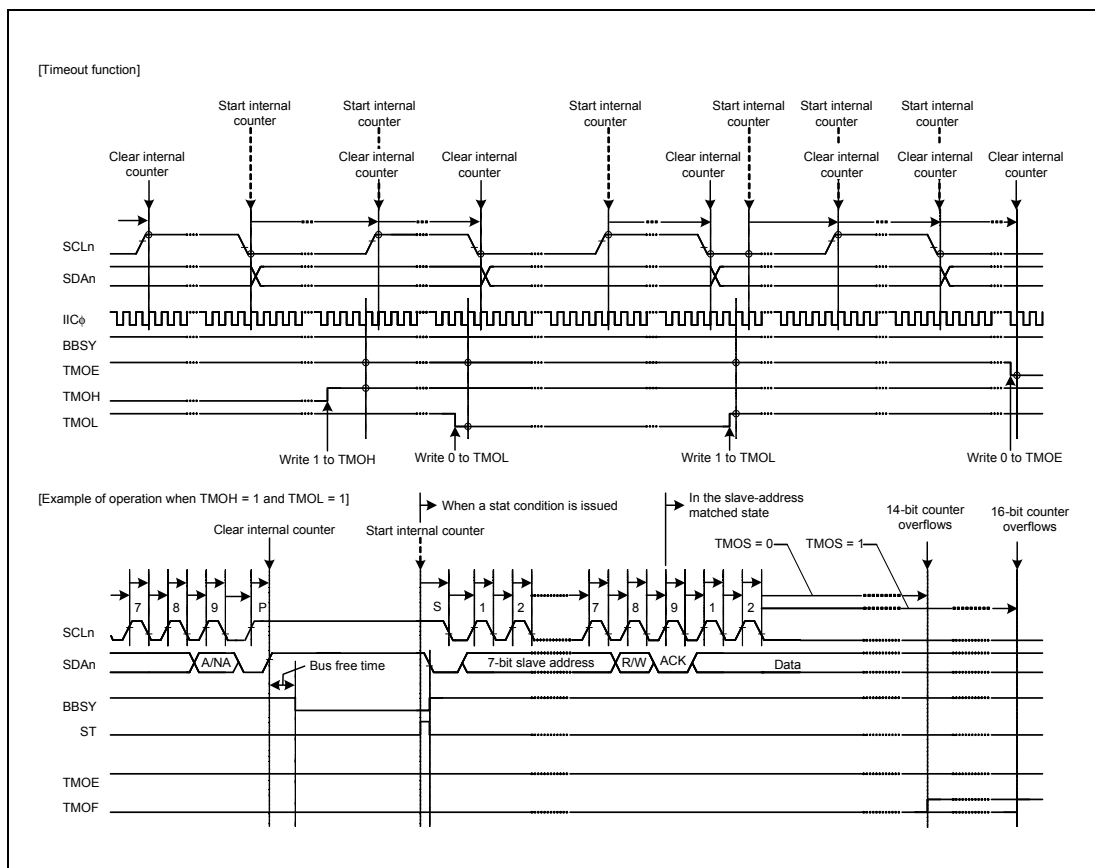


Figure 18.39 Timeout Function (TMOE, TMOS, TMOH, and TMOL Bits)

### 18.13.2 Extra SCL Clock Cycle Output Function

In master mode, the RIIC module has a facility for the output of extra SCL (clock) cycles to release the SDA line of the slave device from being held at the low level due to the master being out of synchronization with the slave device.

This function is mainly used in master mode to release the SDA line of the slave device from the state of being fixed to the low level by including extra cycles of SCL output from the RIIC with single cycles of the SCL (clock) signal as the unit in the case of a bus error where the RIIC cannot issue a stop condition because the slave device is holding the SDA line at the low level. Do not use this facility in normal situations. Using it when communications are proceeding correctly will lead to malfunctions.

When the RIICnCR1.CLO bit is set to 1 in master mode, a single cycle of the SCL clock at the frequency corresponding to the transfer rate settings (settings of the RIICnMR1.CKS[2:0] bits, and of the RIICnBRH and RIICnBRL registers) is output as an extra clock cycle. After output of this single cycle of the SCL clock, the CLO bit is automatically cleared to 0. Therefore, further extra clock cycles can be output consecutively by the software program writing 1 to the CLO bit after having read CLO = 0.

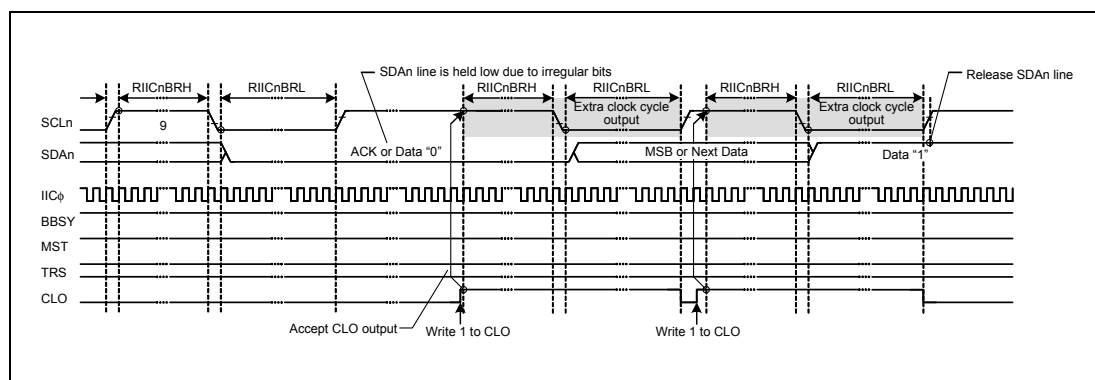
When the RIIC module is in master mode and the slave device is holding the SDA line at the low level because synchronization with the slave device has been lost due to the effects of noise, etc., the output of a stop condition is not possible. The facility for output of an extra cycle of the SCL (clock) signal can be used to output extra cycles of SCL one by one to make the slave device release the SDA line from being held at the low level, thus recovering the bus from an unusable state. Release of the SDA line by the slave device can be monitored by reading the RIICnCR1.SDAI bit. After confirming release of the SDA line by the slave device, complete communications by reissuing the stop condition.

Use this facility with the RIICnFER.MALE bit (master arbitration-lost detection disabled) cleared to 0. If the MALE bit is set to 1 (master arbitration-lost detection enabled), arbitration is lost when the value of the RIICnCR1.SDAO bit does not match the state of the SDA line, so take care on this point.

[Output conditions for using the RIICnCR1.CLO bit]

- When the bus is free (RIICnCR2.BBSY flag = 0) or in master mode (RIICnCR2.MST bit = 1 and BBSY flag = 1)
- When the communication device does not hold the SCL line low

**Figure 18.40** shows the operation timing of the extra SCL clock cycle output function (CLO bit).



**Figure 18.40** Extra SCL Clock Cycle Output Function (CLO Bit)

### 18.13.3 RIIC Reset and Internal Reset

The RIIC module incorporates a function for resetting itself. There are two types of reset. One is referred to as an RIIC reset; this initializes all registers including the RIICnCR2.BBSY flag. The other is referred to as an internal reset; this releases the RIIC from the slave-address matched state and initializes the internal counter while retaining other settings.

After issuing a reset, be sure to clear the RIICnCR1.IICRST bit to 0.

Both types of reset are effective for release from bus-hung states since both restore the output state of the SCL and SDA pins to the high impedance state.

Issuing a reset during slave operation may lead to a loss of synchronization between the master device clock and the slave device clock, so avoided this where possible. Note that monitoring of the bus state, such as for the presence of a start condition, is not possible during an RIIC reset (RIICnCR1.ICE and IICRST bits = 01b).

For a detailed description of the RIIC and internal resets, see **Section 18.15, Reset States**.

## 18.14 SMBus Operation

The RIIC is available for data communication conforming to the SMBus (Version 2.0). To perform SMBus communication, set the RIICnMR3.SMBS bit to 1. To use the transfer rate within a range of 10 kbps to 100 kbps of the SMBus standard, set the RIICnMR1.CKS[2:0] bits, RIICnCBRH, and RIICnBRL. In addition, determine the values of the RIICnICMR2.DLCS bit and the RIICnMR2.SDDL[2:0] bits to meet the data hold time specification of 300 ns or more. If the RIIC is used only as a slave device, the transfer rate setting is not necessary. When the RIIC is used only as a slave device, the transfer rate setting is not necessary, whereas the RIICnBRL needs to be set to a value longer than the data setup time (250 ns).

For the SMBus device default address (1100 001b), use one of the slave address registers L0 to L2 (RIICnSAR0, RIICnSAR1, and RIICnSAR2), and set the corresponding RIICnICSARy.FSy bit (7-bit/10-bit address format select) (y = 0 to 2) to 0 (7-bit address format).

When transmitting the UDID (Unique Device Identifier), set the RIICnFER.SALE bit to 1 to enable the slave arbitration lost detection function.

### 18.14.1 SMBus Timeout Measurement

#### (1) Measuring timeout of slave device

The following period (timeout interval:  $T_{LOW:SEXT}$ ) must be measured for slave devices in SMBus communication.

- From start condition to stop condition

To measure timeout for slave devices, measure the period from start condition detection to stop condition detection with the internal timer using a start condition detection interrupt (INTRIICSTI) and stop condition detection interrupt (INTRIICSPI) of the RIIC. The measured timeout period must be within the total clock low-level period [slave device]  $T_{LOW:SEXT}$ : 25 ms (max.) of the SMBus standard.

If the time measured with the internal timer exceeds the clock low-level detection timeout  $T_{TIMEOUT}$ : 25 ms (min.) of the SMBus standard, the slave device must release the bus by writing 1 to the RIICnCR1.IICRST bit to issue an internal reset of the RIIC. When an internal reset is issued, the RIIC stops driving the bus for the SCL pin and SDA pin and make the SCL/SDA pin outputs high impedance, which releases the bus.

#### (2) Measuring timeout of master device

The following periods (timeout interval:  $T_{LOW:MEXT}$ ) must be measured for master devices in SMBus communication.

- From start condition to acknowledge bit
- Between acknowledge bits
- From acknowledge bit to stop condition

To measure timeout for master devices, measure these periods with the internal timer using a start condition detection interrupt (INTRIICSTI), stop condition detection interrupt (INTRIICSPI), and transmit end interrupt (INTRIICTEI) or receive data full interrupt (INTRIICRI) of the RIIC. The measured timeout period must be within the total clock low-level extended period [master device]  $T_{LOW:MEXT}$ : 10 ms (max.) of the SMBus standard, and the total of all  $T_{LOW:MEXT}$  from start condition to stop condition must be within  $T_{LOW:SEXT}$ : 25 ms (max.).



For the ACK receive timing (rising edge of the ninth SMBCLK clock cycle), monitor the RIICnSR2.TEND flag in master transmit mode (master transmitter) and the RIICnSR2.RDRF flag in master receive mode (master receiver). For this reason, perform bitwise transmit operation in master transmit mode, and hold the RIICnMR3.RDRFS bit 0 until the byte just before reception of the final byte in master receive mode. While the RDRFS bit is 0, the RDRF flag is set to 1 at the rising edge of the ninth SMBCLK clock cycle.

If the period measured with the internal timer exceeds the total clock low-level extended period [master device]  $T_{LOW:MEXT}$ : 10 ms (max.) of the SMBus standard or the total of measured periods exceeds the clock low-level detection timeout  $T_{TIMEOUT}$ : 25 ms (min.) of the SMBus standard, the master device must stop the transaction by issuing a stop condition. In master transmit mode, immediately stop the transmit operation (writing data to RIICnDRT).

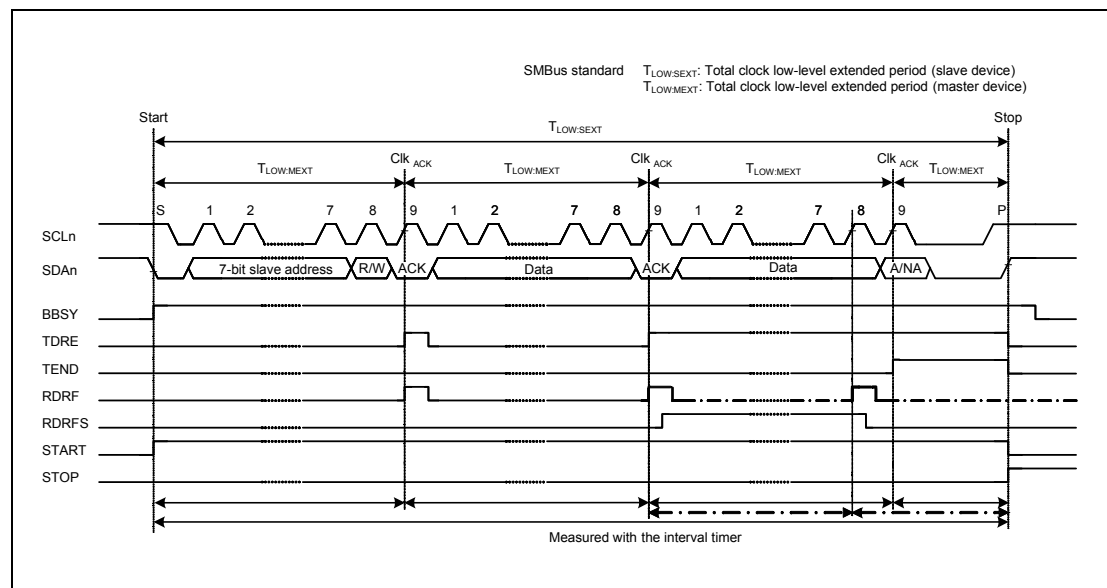


Figure 18.41 SMBus Timeout Measurement

### 18.14.2 SMBus Host Notification Protocol/Notify ARP Master

In communications over an SMBus, a slave device can temporarily act as a master device to notify the SMBus host (or ARP master) of (or request the SMBus host for) its own slave address or to request its own slave address from the SMBus host.

For this LSI to operate as an SMBus host (or ARP master), the host address (0001 000b) sent from the slave device must be detected as a slave address, so the RIIC has a function for detecting the host address. To detect the host address as a slave address, set the RIICnMR3.SMBS bit and the RIICnSER.HOAE bit to 1. Operation after the host address has been detected is the same as normal slave operation.

## 18.15 Reset States

The RIIC has chip reset, RIIC reset, and internal reset functions. **Table 18.24** lists the scope of each reset and reset conditions.

**Table 18.24** Reset Conditions

		Power-on Reset	RIIC Reset (ICE = 0, IICRST = 1)	Internal Reset (ICE = 1, IICRST = 1)	Start Condition/Restart Condition Detection	Stop Condition Detection
RIICnCR1	ICE, IICRST	At a reset	Retained	Retained	Operation (retained)	Operation (retained)
	SCLO, SDAO		At a reset	At a reset		
	Others			Retained		
RIICnCR2	BBSY	At a reset	At a reset	Operation	Operation	Operation
	ST			At a reset	At a reset	Operation (retained)
	Others					At a reset
RIICnMR1	BC[2:0]	At a reset	At a reset	At a reset	At a reset	Operation (retained)
	Others			Retained	Operation (retained)	
RIICnMR2		At a reset	At a reset	Retained	Operation (retained)	Operation (retained)
RIICnMR3		At a reset	At a reset	Retained	Operation (retained)	Operation (retained)
RIICnFER		At a reset	At a reset	Retained	Operation (retained)	Operation (retained)
RIICnSER		At a reset	At a reset	Retained	Operation (retained)	Operation (retained)
RIICnIER		At a reset	At a reset	Retained	Operation (retained)	Operation (retained)
RIICnSR1		At a reset	At a reset	At a reset	Operation (retained)	At a reset
RIICnSR2	TDRE, TEND	At a reset	At a reset	At a reset	Operation (retained)	At a reset
	START				Operation	
	STOP				Operation (retained)	Operation
	Others					Operation (retained)
RIICnSAR0, 1, 2		At a reset	At a reset	Retained	Operation (retained)	Operation (retained)
RIICnBRH, RIICnBRL		At a reset	At a reset	Retained	Operation (retained)	Operation (retained)
RIICnDRT		At a reset	At a reset	Retained	Operation (retained)	Operation (retained)
RIICnDRR		At a reset	At a reset	Retained	Operation (retained)	Operation (retained)
RIICnDRS		At a reset	At a reset	At a reset	Operation (retained)	Operation (retained)
Timeout function		At a reset	At a reset	Operation	Operation	Operation
Bus free time measurement		At a reset	At a reset	Operation	Operation	Operation

## 19. Serial Sound Interface

The serial sound interface is a module designed to send or receive audio data interface with various devices offering I<sup>2</sup>S bus compatibility. It also provides additional modes for other common formats, as well as support for multi-channel mode.

### 19.1 Features

- Number of channels: Six channels
- Operating mode: Non-compressed mode  
The non-compressed mode supports serial audio streams divided by channels.
- Serves as both a transmitter and a receiver  
Channels 0, 1, 3, and 5 support full-duplex communications.
- Capable of using serial bus format
- Asynchronous transfer takes place between the data buffer and the shift register.
- It is possible to select a dividing ratio for the clock used by the serial bus interface.
- It is possible to control data transmission or reception with DMA transfer and interrupt requests.  
A path is also provided for direct data transfer between this module and the SCUX module.
- Selects the oversampling clock input from among the following pins:
  - AUDIO\_CLK (1 to 50 MHz)
  - AUDIO\_X1, AUDIO\_X2 (when connecting a crystal resonator: 10 to 50 MHz, when used to input external clock: 1 to 50 MHz)
- Includes 8-stage FIFO buffers in transmitter and receiver
- Supports multi-channel mode (TDM mode) in which the SSIWS signal is high only for system word 1 period.
- Supports WS continue mode in which the SSIWS signal is not stopped.
- An interrupt is used to notify the CPU of a change of sampling frequency during transfer.

Figure 19.1 shows a block diagram of this module.

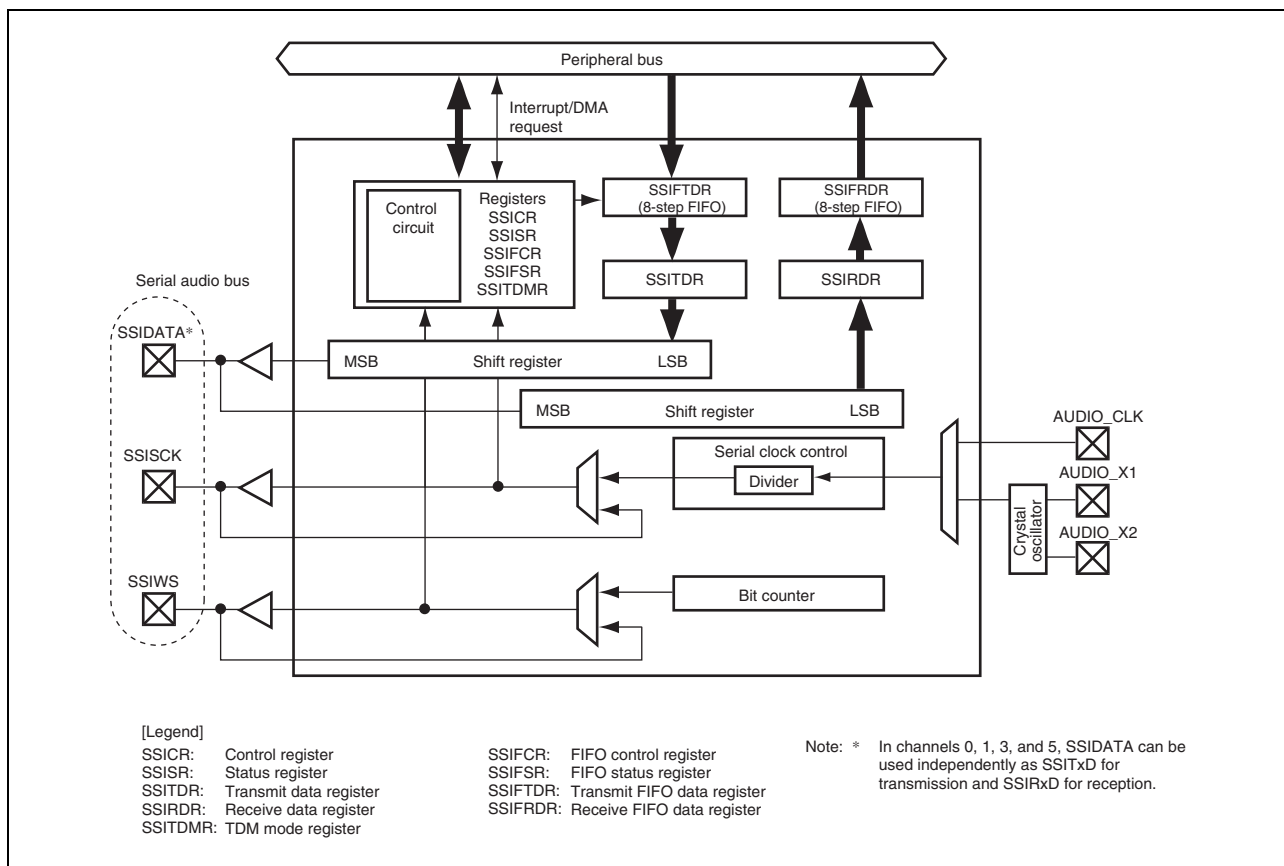


Figure 19.1 Block Diagram of Serial Sound Interface

## 19.2 Input/Output Pins

Table 19.1 shows the pin assignments relating to this module.

**Table 19.1 Pin Assignments**

Channel	Pin Name	I/O	Description
0, 1, 3, 5	SSISCK0*1*3, SSISCK1*1*3, SSISCK3*1*3, SSISCK5*1*3	I/O	Serial bit clock
	SSIWS0*1*3, SSIWS1*1*3, SSIWS3*1*3, SSIWS5*1*3	I/O	Word selection
	SSITxD0, SSITxD1, SSITxD3, SSITxD5	Output	Serial data output
	SSIRxD0*1, SSIRxD1*1, SSIRxD3*1, SSIRxD5*1	Input	Serial data input
2, 4	SSISCK2*1*3, SSISCK4*1*3	I/O	Serial bit clock
	SSIWS2*1*3, SSIWS4*1*3	I/O	Word selection
	SSIDATA2*1, SSIDATA4*1	I/O	Serial data input/output
Common	AUDIO_CLK*2	Input	External clock for audio (input oversampling clock)
	AUDIO_X1	Input	Crystal resonator/external clock for audio (input oversampling clock)
	AUDIO_X2	Output	

Note 1. In slave mode, whether or not to use the noise canceler in the input route can be selected. For details, refer to section 54.3.15, Serial Sound Interface Noise Canceler Control Register (SNCR), under section 54, Ports.

Note 2. When the SSInCKS bit (n = 0 to 5) is set to 1, the MLB\_CLK pin is used as the AUDIO\_CLK pin. For details, refer to section 48.3.70, SSI Pin Mode Register (SSIPMD\_CIM) and section 48.4.4, Pin Connection Specifications of SSIF.

Note 3. Each of SSIF1 to SSIF3 can use SSISCK0 and SSIWS0 of SSIF0 as its own SSISCK and SSIWS. Each of SSIF4 and SSIF5 can use SSISCK3 and SSIWS3 of SSIF3 as its own SSISCK and SSIWS. For details, refer to section 48.3.70, SSI Pin Mode Register (SSIPMD\_CIM) and section 48.4.4, Pin Connection Specifications of SSIF. When SSIF1 to SSIF3 uses SSIF0 as the master or SSIF4 and SSIF5 uses SSIF3 as the master, the noise canceler function cannot be used.

### 19.3 Register Description

Table 19.2 lists the register configuration. Note that explanation in the text does not refer to the channels.

**Table 19.2 Register Configuration**

Channel	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
0	Control register 0	SSICR_0	R/W	H'00000000	H'E820B000	32
	Status register 0	SSISR_0	R/W*1	H'02000013	H'E820B004	32
	FIFO control register 0	SSIFCR_0	R/W	H'00000000	H'E820B010	32
	FIFO status register 0	SSIFSR_0	R/(W)*2	H'00010000	H'E820B014	32
	Transmit FIFO data register 0	SSIFTDR_0	W	Undefined	H'E820B018	32
	Receive FIFO data register 0	SSIFRDR_0	R	Undefined	H'E820B01C	32
	TDM mode register 0	SSITDMR_0	R/W	H'00000000	H'E820B020	32
	FC control register 0	SSIFCCR_0	R/W	H'00000000	H'E820B024	32
	FC mode register 0	SSIFCMR_0	R/W	H'00000000	H'E820B028	32
	FC status register 0	SSIFCSR_0	R/(W)*3	H'00000000	H'E820B02C	32
	1	Control register 1	SSICR_1	R/W	H'00000000	H'E820B800
Status register 1		SSISR_1	R/W*1	H'02000013	H'E820B804	32
FIFO control register 1		SSIFCR_1	R/W	H'00000000	H'E820B810	32
FIFO status register 1		SSIFSR_1	R/(W)*2	H'00010000	H'E820B814	32
Transmit FIFO data register 1		SSIFTDR_1	W	Undefined	H'E820B818	32
Receive FIFO data register 1		SSIFRDR_1	R	Undefined	H'E820B81C	32
TDM mode register 1		SSITDMR_1	R/W	H'00000000	H'E820B820	32
FC control register 1		SSIFCCR_1	R/W	H'00000000	H'E820B824	32
FC mode register 1		SSIFCMR_1	R/W	H'00000000	H'E820B828	32
FC status register 1		SSIFCSR_1	R/(W)*3	H'00000000	H'E820B82C	32
2		Control register 2	SSICR_2	R/W	H'00000000	H'E820C000
	Status register 2	SSISR_2	R/W*1	H'02000013	H'E820C004	32
	FIFO control register 2	SSIFCR_2	R/W	H'00000000	H'E820C010	32
	FIFO status register 2	SSIFSR_2	R/(W)*2	H'00010000	H'E820C014	32
	Transmit FIFO data register 2	SSIFTDR_2	W	Undefined	H'E820C018	32
	Receive FIFO data register 2	SSIFRDR_2	R	Undefined	H'E820C01C	32
	TDM mode register 2	SSITDMR_2	R/W	H'00000000	H'E820C020	32
	FC control register 2	SSIFCCR_2	R/W	H'00000000	H'E820C024	32
	FC mode register 2	SSIFCMR_2	R/W	H'00000000	H'E820C028	32
	FC status register 2	SSIFCSR_2	R/(W)*3	H'00000000	H'E820C02C	32
	3	Control register 3	SSICR_3	R/W	H'00000000	H'E820C800
Status register 3		SSISR_3	R/W*1	H'02000013	H'E820C804	32
FIFO control register 3		SSIFCR_3	R/W	H'00000000	H'E820C810	32
FIFO status register 3		SSIFSR_3	R/(W)*2	H'00010000	H'E820C814	32
Transmit FIFO data register 3		SSIFTDR_3	W	Undefined	H'E820C818	32
Receive FIFO data register 3		SSIFRDR_3	R	Undefined	H'E820C81C	32
TDM mode register 3		SSITDMR_3	R/W	H'00000000	H'E820C820	32
FC control register 3		SSIFCCR_3	R/W	H'00000000	H'E820C824	32
FC mode register 3		SSIFCMR_3	R/W	H'00000000	H'E820C828	32
FC status register 3		SSIFCSR_3	R/(W)*3	H'00000000	H'E820C82C	32

**Table 19.2 Register Configuration**

Channel	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
4	Control register 4	SSICR_4	R/W	H'00000000	H'E820D000	32
	Status register 4	SSISR_4	R/W <sup>*1</sup>	H'02000013	H'E820D004	32
	FIFO control register 4	SSIFCR_4	R/W	H'00000000	H'E820D010	32
	FIFO status register 4	SSIFSR_4	R/(W) <sup>*2</sup>	H'00010000	H'E820D014	32
	Transmit FIFO data register 4	SSIFTDR_4	W	Undefined	H'E820D018	32
	Receive FIFO data register 4	SSIFRDR_4	R	Undefined	H'E820D01C	32
	TDM mode register 4	SSITDMR_4	R/W	H'00000000	H'E820D020	32
	FC control register 4	SSIFCCR_4	R/W	H'00000000	H'E820D024	32
	FC mode register 4	SSIFCMR_4	R/W	H'00000000	H'E820D028	32
	FC status register 4	SSIFCSR_4	R/(W) <sup>*3</sup>	H'00000000	H'E820D02C	32
5	Control register 5	SSICR_5	R/W	H'00000000	H'E820D800	32
	Status register 5	SSISR_5	R/W <sup>*1</sup>	H'02000013	H'E820D804	32
	FIFO control register 5	SSIFCR_5	R/W	H'00000000	H'E820D810	32
	FIFO status register 5	SSIFSR_5	R/(W) <sup>*2</sup>	H'00010000	H'E820D814	32
	Transmit FIFO data register 5	SSIFTDR_5	W	Undefined	H'E820D818	32
	Receive FIFO data register 5	SSIFRDR_5	R	Undefined	H'E820D81C	32
	TDM mode register 5	SSITDMR_5	R/W	H'00000000	H'E820D820	32
	FC control register 5	SSIFCCR_5	R/W	H'00000000	H'E820D824	32
	FC mode register 5	SSIFCMR_5	R/W	H'00000000	H'E820D828	32
	FC status register 5	SSIFCSR_5	R/(W) <sup>*3</sup>	H'00000000	H'E820D82C	32

Note 1. Although bits 29 to 26 in these registers can be read from or written to, bits other than these are read-only. For details, refer to section 19.3.2, Status Register (SSISR).

Note 2. To bits 16 and 0 in these registers, only 0 can be written to clear the flags. Other bits are read-only. For details, refer to section 19.3.6, FIFO Status Register (SSIFSR).

Note 3. To bit 24 in these registers, only 0 can be written to clear the flag. Other bits are read-only. For details, refer to section 19.3.12, FC Status Register (SSIFCSR).

### 19.3.1 Control Register (SSICR)

SSICR is a 32-bit readable/writable register that controls the IRQ, selects the polarity status, and sets operating mode.

Bit: 31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
-	CKS	TUIEN	TOIEN	RUIEN	ROIEN	IIEN	-	CHNL[1:0]	DWL[2:0]			SWL[2:0]			
Initial value: 0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W: R	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit: 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SCKD	SWSD	SCKP	SWSP	SPDP	SDTA	PDTA	DEL	CKDV[3:0]			MUEN	-	TEN	REN	
Initial value: 0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W: R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W

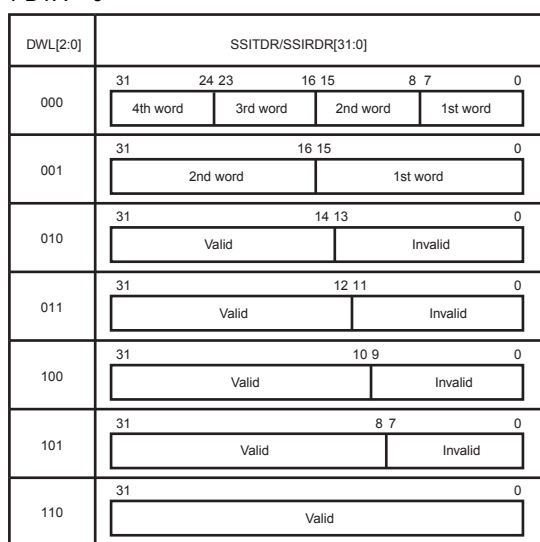
Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved The read value is undefined. The write value should always be 0.
30	CKS	0	R/W	Oversampling Clock Select Selects the clock source for oversampling. 0: AUDIO_X1 input 1: AUDIO_CLK input
29	TUIEN	0	R/W	Transmit Underflow Interrupt Enable 0: Disables an underflow interrupt. 1: Enables an underflow interrupt.
28	TOIEN	0	R/W	Transmit Overflow Interrupt Enable 0: Disables an overflow interrupt. 1: Enables an overflow interrupt.
27	RUIEN	0	R/W	Receive Underflow Interrupt Enable 0: Disables an underflow interrupt. 1: Enables an underflow interrupt.
26	ROIEN	0	R/W	Receive Overflow Interrupt Enable 0: Disables an overflow interrupt. 1: Enables an overflow interrupt.
25	IIEN	0	R/W	Idle Mode Interrupt Enable 0: Disables an idle mode interrupt. 1: Enables an idle mode interrupt.
24	—	0	R	Reserved The read value is undefined. The write value should always be 0.
23, 22	CHNL[1:0]	00	R/W	Channels [When TDM = 0] These bits show the number of channels in each system word. 00: Having one channel per system word 01: Having two channels per system word 10: Having three channels per system word 11: Having four channels per system word [When TDM = 1] These bits show the number of system words in each TDM frame. 00: Setting prohibited 01: Having four system words per TDM frame 10: Having six system words per TDM frame 11: Having eight system words per TDM frame



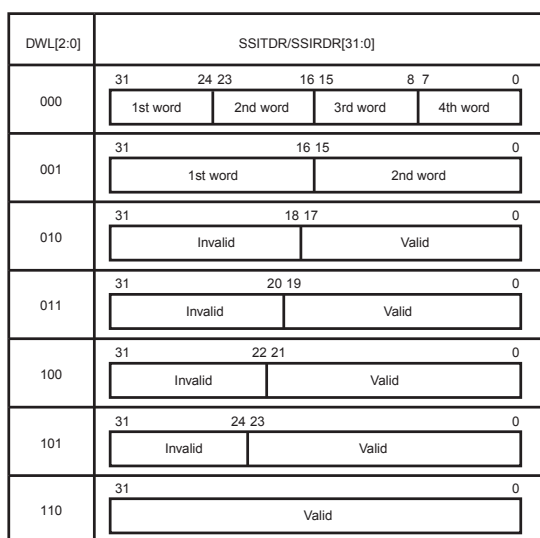
Bit	Bit Name	Initial Value	R/W	Description															
21 to 19	DWL[2:0]	000	R/W	Data Word Length These bits indicate the number of bits in a data word. 000: 8 bits 001: 16 bits 010: 18 bits 011: 20 bits 100: 22 bits 101: 24 bits 110: 32 bits 111: Setting prohibited															
18 to 16	SWL[2:0]	000	R/W	System Word Length These bits indicate the number of bits in a system word. 000: 8 bits 001: 16 bits 010: 24 bits 011: 32 bits 100: 48 bits 101: 64 bits 110: 128 bits 111: 256 bits															
15	SCKD	0	R/W	Serial Bit Clock Direction 0: Serial bit clock is input, slave mode. 1: Serial bit clock is output, master mode. Note: Only the following settings are allowed: (SCKD, SWSD) = (0, 0) or (1, 1). Other settings are prohibited.															
14	SWSD	0	R/W	Serial WS Direction 0: Serial word select is input, slave mode. 1: Serial word select is output, master mode. Note: Only the following settings are allowed: (SCKD, SWSD) = (0,0) or (1,1). Other settings are prohibited.															
13	SCKP	0	R/W	Serial Bit Clock Polarity 0: SSIWS and SSIDATA change at the SSISCK falling edge (sampled at the SCK rising edge). 1: SSIWS and SSIDATA change at the SSISCK rising edge (sampled at the SCK falling edge). <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th></th> <th>SCKP = 0</th> <th>SCKP = 1</th> </tr> </thead> <tbody> <tr> <td>SSIDATA input sampling timing at the time of reception</td> <td>SSISCK rising edge</td> <td>SSISCK falling edge</td> </tr> <tr> <td>SSIDATA output change timing at the time of transmission</td> <td>SSISCK falling edge</td> <td>SSISCK rising edge</td> </tr> <tr> <td>SSIWS input sampling timing at the time of slave mode (SWSD = 0)</td> <td>SSISCK rising edge</td> <td>SSISCK falling edge</td> </tr> <tr> <td>SSIWS output change timing at the time of master mode (SWSD = 1)</td> <td>SSISCK falling edge</td> <td>SSISCK rising edge</td> </tr> </tbody> </table>		SCKP = 0	SCKP = 1	SSIDATA input sampling timing at the time of reception	SSISCK rising edge	SSISCK falling edge	SSIDATA output change timing at the time of transmission	SSISCK falling edge	SSISCK rising edge	SSIWS input sampling timing at the time of slave mode (SWSD = 0)	SSISCK rising edge	SSISCK falling edge	SSIWS output change timing at the time of master mode (SWSD = 1)	SSISCK falling edge	SSISCK rising edge
	SCKP = 0	SCKP = 1																	
SSIDATA input sampling timing at the time of reception	SSISCK rising edge	SSISCK falling edge																	
SSIDATA output change timing at the time of transmission	SSISCK falling edge	SSISCK rising edge																	
SSIWS input sampling timing at the time of slave mode (SWSD = 0)	SSISCK rising edge	SSISCK falling edge																	
SSIWS output change timing at the time of master mode (SWSD = 1)	SSISCK falling edge	SSISCK rising edge																	
12	SWSP	0	R/W	Serial WS Polarity [When TDM = 0] 0: SSIWS is low for the 1st channel, high for the 2nd channel. 1: SSIWS is high for the 1st channel, low for the 2nd channel. [When TDM = 1] 0: SSIWS is high only for system word 1 period, low for other periods. 1: Setting prohibited															
11	SPDP	0	R/W	Serial Padding Polarity This bit is used to specify the active sense of padding bits. It is also used to specify the logical value to be output from the SSITxD pin when the MUEN bit is set to 0 with transfer disabled in WS continue mode. 0: Padding bits are low. 1: Padding bits are high.															

Bit	Bit Name	Initial Value	R/W	Description
10	SDTA	0	R/W	Serial Data Alignment 0: Transmitting and receiving in the order of serial data and padding bits 1: Transmitting and receiving in the order of padding bits and serial data

9	PDTA	0	R/W	Parallel Data Alignment When the data word length is 32 bits, this configuration field has no meaning. This bit applies to SSIRDR in receive mode and SSITDR in transmit mode. When data word length is 8 or 16 bits: 0: The lower bits of parallel data (SSITDR, SSIRDR) are transferred prior to the upper bits. 1: The upper bits of parallel data (SSITDR, SSIRDR) are transferred prior to the lower bits. When data word length is 18, 20, 22, or 24 bits: 0: Parallel data (SSITDR, SSIRDR) is left-aligned. 1: Parallel data (SSITDR, SSIRDR) is right-aligned. • PDTA = 0
---	------	---	-----	--



9	PDTA	0	R/W	• PDTA = 1
---	------	---	-----	------------



Note: This bit has no meaning in direct transfer between this module and the SCUX. Use the SWAP function of the SCUX.

8	DEL	0	R/W	Serial Data Delay 0: 1 clock cycle delay between SSIWS and SSIDATA 1: No delay between SSIWS and SSIDATA
---	-----	---	-----	--

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	CKDV[3:0]	0000	R/W	<p>Serial Oversampling Clock Division Ratio</p> <p>Sets the ratio between the oversampling clock (AUDIO<math>\phi</math>) and the serial bit clock. When the SCKD bit is 0, the setting of these bits is ignored. The serial bit clock is used in the shift register and is supplied from the SSISCK pin.</p> <p>0000: AUDIO<math>\phi</math>            0001: AUDIO<math>\phi</math>/2            0010: AUDIO<math>\phi</math>/4            0011: AUDIO<math>\phi</math>/8            0100: AUDIO<math>\phi</math>/16            0101: AUDIO<math>\phi</math>/32            0110: AUDIO<math>\phi</math>/64            0111: AUDIO<math>\phi</math>/128            1000: AUDIO<math>\phi</math>/6            1001: AUDIO<math>\phi</math>/12            1010: AUDIO<math>\phi</math>/24            1011: AUDIO<math>\phi</math>/48            1100: AUDIO<math>\phi</math>/96            1101: Setting prohibited            1110: Setting prohibited            1111: Setting prohibited</p>
3	MUEN	0	R/W	<p>Mute Enable</p> <p>This bit is used to specify muting of the signals from this module during transmission. It is also used to specify the logical value to be output from the SSITxD pin when the MUEN bit is set to 0 with transfer disabled in WS continue mode.</p> <p>0: This module is not muted.            1: This module is muted.</p> <p>Note: When this module is muted, serial data to be output is rewritten to 0 but data transmission in the module is not stopped. Write dummy data to the SSIFTDR not to generate a transmit underflow because the number of data in the transmit FIFO is decreasing.</p>
2	$\frac{3}{4}$	0	R	<p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p>
1	TEN	0	R/W	<p>Transmit Enable</p> <p>0: Disables the transmit operation.            1: Enables the transmit operation.</p> <p>Note: When transmission stop/start is controlled by the SSI control register (SSICTRL_CIM) in the SCUX, set this bit to 0.</p>
0	REN	0	R/W	<p>Receive Enable</p> <p>0: Disables the receive operation.            1: Enables the receive operation.</p> <p>Note: When reception stop/start is controlled by the SSI control register (SSICTRL_CIM) in the SCUX, set this bit to 0.</p>

### 19.3.2 Status Register (SSISR)

SSISR consists of status flags indicating the operational status of this module and bits indicating the current channel numbers and word numbers.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	TUIRQ	TOIRQ	RUIRQ	ROIRQ	IIRQ	-	-	-	-	-	-	-	-	-
Initial value:	Undefined	Undefined	0	0	0	0	1	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W:	R	R	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	TCHNO[1:0]	TSWNO	RCHNO[1:0]	RSWNO	IDST		
Initial value:	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	0	0	1	0	0	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Note: \* The bit can be read or written to. Writing 0 initializes the bit, but writing 1 is ignored.

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
29	TUIRQ	0	R/(W)*	<p>Transmit Underflow Error Interrupt Status Flag</p> <p>This status flag indicates that transmit data was supplied at a lower rate than was required.</p> <p>This bit is set to 1 regardless of the value of the TUIEN bit and can be cleared by writing 0 to this bit.</p> <p>If TUIRQ = 1 and TUIEN = 1, an SSI interrupt occurs.</p> <p>If TUIRQ = 1, SSITDR did not have data written to it before it was required for transmission. This will lead to the same data being transmitted once more and a potential corruption of multi-channel data. As a result, this module will output erroneous data.</p> <p>Note: When an underflow error occurs, the current data in the data buffer of this module is transmitted until the next data is written.</p>
28	TOIRQ	0	R/(W)*	<p>Transmit Overflow Error Interrupt Status Flag</p> <p>This status flag indicates that transmit data was supplied at a higher rate than was required.</p> <p>This bit is set to 1 regardless of the value of the TOIEN bit and can be cleared by writing 0 to this bit.</p> <p>If TOIRQ = 1 and TOIEN = 1, an SSI interrupt occurs.</p> <p>If TOIRQ = 1, SSIFDR had data written to it while the transmit FIFO is full (TDC = H'8). This will lead to the loss of data and a potential corruption of multi-channel data.</p>
27	RUIRQ	0	R/(W)*	<p>Receive Underflow Error Interrupt Status Flag</p> <p>This status flag indicates that receive data was supplied at a lower rate than was required.</p> <p>This bit is set to 1 regardless of the value of the RUIEN bit and can be cleared by writing 0 to this bit.</p> <p>If RUIRQ = 1 and RUIEN = 1, an SSI interrupt occurs.</p> <p>If RUIRQ = 1, SSIFRDR was read while the receive FIFO is empty (RDC = H'0). This can cause invalid receive data to be stored, which may lead to corruption of multi-channel data.</p>
26	ROIRQ	0	R/(W)*	<p>Receive Overflow Error Interrupt Status Flag</p> <p>This status flag indicates that receive data was supplied at a higher rate than was required.</p> <p>This bit is set to 1 regardless of the value of the ROIEN bit and can be cleared by writing 0 to this bit.</p> <p>If ROIRQ = 1 and ROIEN = 1, an SSI interrupt occurs.</p> <p>If ROIRQ = 1, SSIRDR was not read before there was new unread data written to it. This will lead to the loss of data and a potential corruption of multi-channel data.</p> <p>Note: When an overflow error occurs, the current data in the data buffer of this module is overwritten by the next incoming data from the SSI interface.</p>

Bit	Bit Name	Initial Value	R/W	Description
25	IIRQ	1	R	<p>Idle Mode Interrupt Status Flag</p> <p>This status flag indicates whether this module is in the idle state. This bit is set to 1 regardless of the value of the IEN bit to allow polling. The interrupt can be masked by clearing IEN to 0, but cannot be cleared by writing 0 to this bit.</p> <p>If IIRQ = 1 and IEN = 1, an SSI interrupt occurs.</p> <p>0: This module is not in idle state. 1: This module is in idle state.</p>
24 to 7	—	Undefined	R	<p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p>
6, 5	TCHNO [1:0]	00	R	<p>Transmit Channel Number</p> <p>These bits show the current channel number. These bits indicate the number of a channel whose data is required to be written to SSITDR. This value will change as the data is copied to the shift register, regardless of whether the data is written to SSITDR. When TDM or CONT is 1, these bits cannot be used.</p>
4	TSWNO	1	R	<p>Transmit Serial Word Number</p> <p>This status bit indicates the current word number. This bit indicates which system word is required to be written to SSITDR. This value will change as the data is copied to the shift register, regardless of whether the data is written to SSITDR. When TDM or CONT is 1, this bit cannot be used.</p>
3, 2	RCHNO [1:0]	00	R	<p>Receive Channel Number</p> <p>These bits show the current channel number. These bits indicate which channel the data in SSIRDR currently represents. This value will change as the data in SSIRDR is updated from the shift register. When TDM or CONT is 1, these bits cannot be used.</p>
1	RSWNO	1	R	<p>Receive Serial Word Number</p> <p>This status bit indicates the current word number. This bit indicates which system word the data in SSIRDR currently represents. This value will change as the data in SSIRDR is updated from the shift register, regardless of whether SSIRDR has been read. When TDM or CONT is 1, this bit cannot be used.</p>
0	IDST	1	R	<p>Idle Mode Status Flag</p> <p>This status flag indicates that the serial bus activity has stopped. This bit is cleared to 0 if the serial bus is currently active while TEN = 1 or REN = 1. This bit is automatically set to 1 if both TEN and REN are cleared to 0 and the current system word communication is completed.</p> <p>Note: If the external device stops the serial bus clock before the current system word is completed, this bit is not set.</p>

Note: \* The bit can be read or written to. Writing 0 initializes the bit, but writing 1 is ignored.

### 19.3.3 Transmit Data Register (SSITDR)

SSITDR is a 32-bit register that stores data to be transmitted. The data for transmission to be stored to SSITDR is automatically transferred from the transmit FIFO data register.

Data written to this register is transferred to the shift register upon transmission request. If the data word length is less than 32 bits, the alignment is determined by the setting of the PDTA control bit in SSICR.

The CPU cannot read or write data from/to SSITDR.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Initial value:	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W:	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial value:	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W:	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

### 19.3.4 Receive Data Register (SSIRDR)

SSIRDR is a 32-bit register that stores received data. The received data stored in SSIRDR is automatically transferred to the receive FIFO data register.

Data in this register is transferred from the shift register each time data word is received. If the data word length is less than 32 bits, the alignment is determined by the setting of the PDTA control bit in SSICR.

The CPU cannot read or write data from/to SSIRDR.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Initial value:	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W:	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial value:	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W:	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

### 19.3.5 FIFO Control Register (SSIFCR)

SSIFCR is a 32-bit readable/writable register that specifies the data trigger numbers for the transmit and receive FIFO data registers, and enables or disables FIFO data resets and interrupt requests.

SSIFCR can always be read or written by the CPU.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	TTRG[1:0]	RTRG[1:0]	TIE	RIE	TFRST	FRST		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7, 6	TTRG[1:0]	00	R/W	<p>Transmit Data Trigger Number</p> <p>When the FIFO is operating for transmission, these bits specify the number of bytes for transmission in the FIFO (trigger number for transmission) at which the TDE flag in the FIFO status register (SSIFSR) will be set. The TDE flag is set to 1 when the number of bytes for transmission in the transmit FIFO data register (SSIFTDR) has fallen to or below the trigger number corresponding to the setting as shown below.</p> <p>00: 7 (1)* 01: 6 (2)* 10: 4 (4)* 11: 2 (6)*</p> <p>Note: * The values in parentheses are the number of empty stages in SSIFTDR at which the TDE flag is set.</p>
5, 4	RTRG[1:0]	00	R/W	<p>Receive Data Trigger Number</p> <p>When the FIFO is operating for reception, these bits specify the number of received bytes in the FIFO (trigger number for reception) at which the RDF flag in the FIFO status register (SSIFSR) will be set. The RDF flag is set to 1 when the number of received bytes in the receive FIFO data register (SSIFRDR) has risen to or above the trigger number corresponding to the setting as shown below.</p> <p>00: 1 01: 2 10: 4 11: 6</p>
3	TIE	0	R/W	<p>Transmit Interrupt Enable</p> <p>This bit enables or disables generation of transmit data empty interrupt (TXI) requests in the following situation: when the FIFO is operating for transmission, the data for transmission in the transmit FIFO data register (SSIFTDR) are transferred to the transmit data register (SSITDR) and the number of data bytes in the transmit FIFO data register has become less than the set transmit trigger number, so that the TDE flag in the FIFO status register (SSIFSR) is set to 1.</p> <p>0: Transmit data empty interrupt (TXI) request is disabled. 1: Transmit data empty interrupt (TXI) request is enabled.*</p> <p>Note: * TXI can be cleared by clearing either the TDE flag (see the description of the TDE bit for details) or TIE bit to 0.</p>

Bit	Bit Name	Initial Value	R/W	Description
2	RIE	0	R/W	<p>Receive Interrupt Enable</p> <p>Enables or disables generation of receive data full interrupt (RXI) requests when the RDF flag in the FIFO status register (SSIFSR) is set to 1 while the FIFO is operating for reception.</p> <p>0: Receive data full interrupt (RXI) request is disabled. 1: Receive data full interrupt (RXI) request is enabled.*</p> <p>Note: * RXI can be cleared by clearing either the RDF flag (see the description of the RDF bit for details) or RIE bit to 0.</p>
1	TFRST	0	R/W	<p>Transmit FIFO Data Register Reset</p> <p>Invalidates the data in the transmit FIFO data register (SSIFTDR) to reset the FIFO to an empty state.</p> <p>0: Reset is disabled.* 1: Reset is enabled.</p> <p>Note: * FIFO is reset at a power-on reset.</p>
0	RFRST	0	R/W	<p>Receive FIFO Data Register Reset</p> <p>Invalidates the data in the receive FIFO data register (SSIFRDR) to reset the FIFO to an empty state.</p> <p>0: Reset is disabled.* 1: Reset is enabled.</p> <p>Note: * FIFO is reset at a power-on reset.</p>



### 19.3.6 FIFO Status Register (SSIFSR)

SSIFSR contains status flags that indicate the state of operation of the transmit and receive FIFO data registers.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	
	-	-	-	-	TDC[3:0]				-	-	-	-	-	-	-	-	TDE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/(W)*	

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	-	-	-	-	RDC[3:0]				-	-	-	-	-	-	-	-	RDF
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/(W)*	

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 24	TDC[3:0]	0000	R	Number of Data Bytes Stored in SSIFTDR TDC[3:0] = H'0 indicates no data for transmission. TDC[3:0] = H'8 indicates that 32 bytes of data for transmission is stored in SSIFTDR.
23 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	TDE	1	R/(W)*	Transmit Data Empty Indicates that, when the FIFO is operating for transmission, the data for transmission in the transmit FIFO data register (SSIFTDR) is transferred to the transmit data register (SSITDR), the number of data bytes in SSIFTDR has become less than the transmit trigger number specified by TTRG[1:0] in the FIFO control register (SSIFCR), and thus writing of transmit data to SSIFTDR has been enabled. 0: Number of data bytes for transmission in SSIFTDR is greater than the set transmit trigger number. [Clearing conditions] <ul style="list-style-type: none"> <li>• 0 is written to TDE after data of the number of bytes larger than the set transmit trigger number is written to SSIFTDR.</li> <li>• The direct memory access controller is activated by transmit data empty (TXI) interrupt, and data of the number of bytes larger than the set transmit trigger number is written to SSIFTDR.</li> </ul> 1: Number of data bytes for transmission in SSIFTDR is equal to or less than the set transmit trigger number.*1 [Setting conditions] <ul style="list-style-type: none"> <li>• Power-on reset</li> <li>• Number of transmission data bytes stored in SSIFTDR has become equal to or less than the set transmit trigger number.</li> </ul> Note: *1 Since SSIFTDR is an 8-stage FIFO register, the amount of data that can be written to it while TDE = 1 is "8 – transmit trigger number to be specified" bytes at maximum. Writing more data will be ignored. The number of data bytes in SSIFTDR is indicated in the TDC bits in SSIFSR.
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 8	RDC[3:0]	0000	R	Number of Data Bytes Stored in SSIFRDR RDC[3:0] = H'0 indicates no received data. RDC[3:0] = H'8 indicates that 32 bytes of received data is stored in SSIFRDR.
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

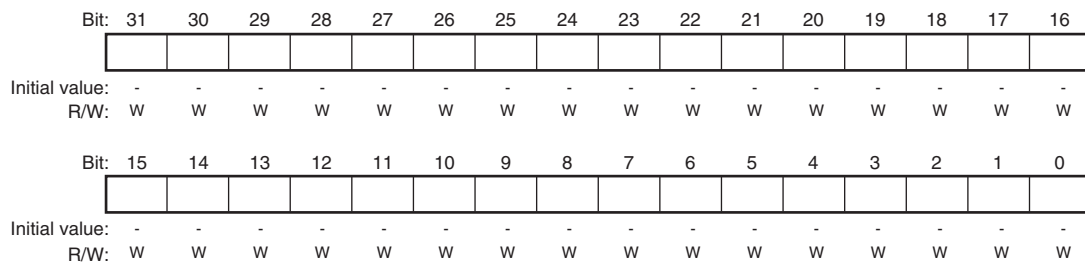
Bit	Bit Name	Initial Value	R/W	Description
0	RDF	0	R/(W)*	<p>Receive Data Full</p> <p>Indicates that, when the FIFO is operating for reception, the received data is transferred to the receive FIFO data register (SSIFRDR) and the number of data bytes in SSIFRDR has become greater than the receive trigger number specified by RTRG[1:0] in the FIFO control register (SSIFCR).</p> <p>0: Number of received data bytes in SSIFRDR is less than the set receive trigger number.</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> <li>• Power-on reset</li> <li>• 0 is written to RDF after the receive FIFO is emptied with writing 1 to RFRST.</li> <li>• 0 is written to RDF after data is read from SSIFRDR until the number of data bytes in SSIFRDR becomes less than the set receive trigger number.</li> <li>• The direct memory access controller is activated by receive data full (RXI) interrupt, and data is read from SSIFRDR until the number of data bytes in SSIFRDR becomes less than the set receive trigger number.</li> </ul> <p>1: Number of received data bytes in SSIFRDR is equal to or greater than the set receive trigger number.</p> <p>[Setting condition]</p> <ul style="list-style-type: none"> <li>• Data of the number of bytes that is equal to or greater than the set receive trigger number is stored in SSIFRDR.*1</li> </ul> <p>Note: *1 Since SSIFRDR is an 8-stage FIFO register, the amount of data that can be read from it while RDF = 1 is the set receive trigger number of bytes at maximum.</p> <p>Continuing to read data from SSIFRDR after reading all the data will result in undefined data to be read. The number of data bytes in SSIFRDR is indicated in the RDC bits in SSIFSR.</p>

Note: \* The bit can be read or written to. Writing 0 initializes the bit, but writing 1 is ignored.

### 19.3.7 Transmit FIFO Data Register (SSIFTDR)

SSIFTDR is a FIFO register consisting of eight stages of 32-bit registers for storing data to be serially transmitted. On detecting that the transmit data register (SSITDR) is empty, this module transfers the data for transmission written to SSIFTDR to SSITDR to start serial transmission, which can continue until SSIFTDR becomes empty. SSIFTDR can be written to by the CPU at any time.

Note that when SSIFTDR is full of transmit data (32 bytes), the next data cannot be written to it. If writing is attempted, it will be ignored and an overflow occurs.

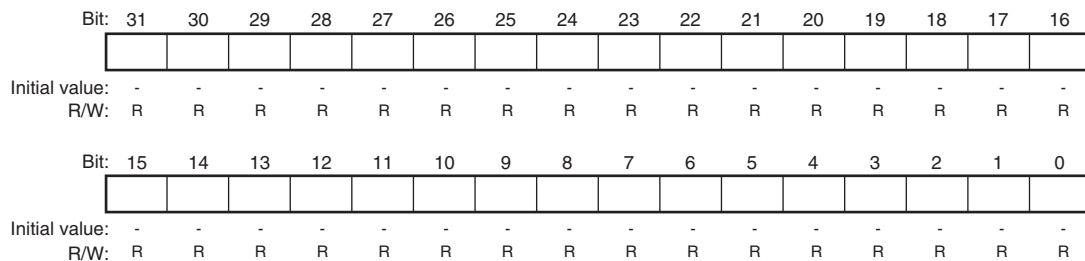


Note: \* Not writable while FIFO is receiving data.

### 19.3.8 Receive FIFO Data Register (SSIFRDR)

SSIFRDR is a FIFO register consisting of eight stages of 32-bit registers for storing serially received data. When four bytes of data have been received, this module transfers the received data in the receive data register (SSIRDR) to SSIFRDR to complete reception operation. Reception can continue until 32 bytes of data have been stored to SSIFRDR. SSIFRDR can be read but cannot be written to by the CPU. Note that when SSIFRDR is read while it does not hold received data, the value read is undefined and a reception underflow will occur.

After SSIFRDR becomes full of received data, the data received thereafter will be lost and a receive overflow occurs.



### 19.3.9 TDM Mode Register (SSITDMR)

SSITDMR is a 32-bit readable/writable register that enables or disables muting of receive data in direct transfer, TDM mode, and WS continue mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	RXD MUTE	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	CONT	-	-	-	-	-	-	-	TDM
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17	RXDMUTE	0	R/W	Receive Direct Data Mute Setting When receive data is output directly to the SCUX, the output data is forcibly muted so that there is no signal (output as 0 data). 0: The receive data is output without change. 1: 0 data is output.
16	—	0	R/W	Reserved Always write 0 to this bit.
15 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	CONT	0	R/W	WS Continue Mode 0: Disables WS continue mode. 1: Enables WS continue mode. Note: This bit can be set only in master mode (SCKD = 1 and SWSD = 1)
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	TDM	0	R/W	TDM Mode 0: Disables TDM mode. 1: Enables TDM mode.

### 19.3.10 FC Control Register (SSIFCCR)

SSIFCCR is a 32-bit readable/writable register that controls frequency change detection.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	FIEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	FCEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	FIEN	0	R/W	Frequency Change Detection Interrupt Enable 0: Disables a frequency change detection interrupt. 1: Enables a frequency change detection interrupt.
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	FCEN	0	R/W	Frequency Change Detection Enable When this bit is set to 1, counting up of cycles of peripheral clock 1 (P1 $\phi$ ) starts at the beginning of the next SSIWS cycle. On the start of each SSIWS cycle, the current counted value is moved to the VALUE bits in the FC Status Register (SSIFCSR). The counter is then cleared to 0 and counting up is resumed. When this bit is set to 0, the counter is cleared to 0 and counting up is stopped. 0: Disables frequency change detection. 1: Enables frequency change detection. Note: Set this bit to 1 after setting the desired values in SSICR, SSIFCMR, and SSITDMR.

### 19.3.11 FC Mode Register (SSIFCMR)

SSIFCMR sets the maximum and minimum allowable numbers of cycles of the peripheral clock 1 (P1 $\phi$ ) for each SSIWS cycle, when frequency change detection is enabled.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	MAXV													
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	MINV													
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
29 to 16	MAXV	0	R/W	Maximum Value Sets the maximum allowable number of cycles of the peripheral clock 1 (P1 $\phi$ ) for each SSIWS cycle, when SSIFCCR.FCEN = 1.
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13 to 0	MINV	0	R/W	Minimum Value Sets the minimum allowable number of cycles of the peripheral clock 1 (P1 $\phi$ ) for each SSIWS cycle, when SSIFCCR.FCEN = 1.

### 19.3.12 FC Status Register (SSIFCSR)

SSIFCSR consists of the frequency change detection status flag and the bits that indicate the current cycle count of the peripheral clock 1 (P1 $\phi$ ).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	FCIRQ	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/(W)*	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	VALUE													
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	FCIRQ	0	R/(W)*	Frequency Change Detection Error Interrupt Status Indicates VALUE > SSIFCMR.MAXV or 0 < VALUE < SSIFCMR.MINV when SSIFCCR.FCEN = 1. This bit is set to 1 regardless of the setting of the FIEN bit in SSIFCCR. Write 0 to clear this flag to 0. When FCIRQ = 1 and SSIFCCR.FIEN = 1, an SSI interrupt is generated.
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13 to 0	VALUE	0	R	VALUE Indicates the current cycle count of the peripheral clock 1 (P1 $\phi$ ) in an SSIWS cycle, when SSIFCCR.FCEN = 1. Each time the next SSIWS cycle starts, the value is updated. Note: When the SSISCK signal stops, the start of an SSIWS cycle cannot be detected. Consequently, the value is not updated.

Note: \* The bit can be read or written to. Writing 0 initializes the bit, but writing 1 is ignored.

## 19.4 Operation Description

### 19.4.1 Bus Format

This module can operate as a transmitter or a receiver and can be configured into many serial bus formats in either mode. The bus format can be selected from one of the twelve major modes shown in Table 19.3.

**Table 19.3 Bus Format for SSIF Module**

	TEN	REN	SCKD	SWSD	TDM	MUEN	IEN	TOIEN	TUIEN	ROIEN	RUIEN	CONT	SWSP	DEL	PDTA	SDTA	SPDP	SCKP	SWL[2:0]	DWL[2:0]	CHNL[1:0]	
Non-Compression Slave Receiver	0	1	0	0	0	Control Bits						Configuration Bits										
Non-Compression Slave Transmitter	1	0	0	0	0																	
Non-Compression Slave Transceiver*	1	1	0	0	0																	
Non-Compression Master Receiver	0	1	1	1	0																	
Non-Compression Master Transmitter	1	0	1	1	0																	
Non-Compression Master Transceiver*	1	1	1	1	0																	
TDM Slave Receiver	0	1	0	0	1	Control Bits						0	Configuration Bits									
TDM Slave Transmitter	1	0	0	0	1							0										
TDM Slave Transceiver*	1	1	0	0	1							0										
TDM Master Receiver	0	1	1	1	1							0										
TDM Master Transmitter	1	0	1	1	1							0										
TDM Master Transceiver*	1	1	1	1	1							0										

Note: \* Set the TEN and REN bits to 1 at the same time when using transceiver mode.



## 19.4.2 Non-Compressed Modes

The non-compressed modes support all serial audio streams split into channels. It supports the I<sup>2</sup>S compatible format as well as many more variants on these modes.

### (1) Slave Receiver

This mode allows the module to receive serial data from another device. The clock and word select signal used for the serial data stream is also supplied from an external device. If these signals do not conform to the format specified in the configuration fields of this module, operation is not guaranteed.

### (2) Slave Transmitter

This mode allows the module to transmit serial data to another device. The clock and word select signal used for the serial data stream is also supplied from an external device. If these signals do not conform to the format specified in the configuration fields of this module, operation is not guaranteed.

### (3) Slave Transceiver

This mode allows serial data transmission and reception between this module and another device. The clock and word select signal used for the serial data stream is also supplied from an external device. If these signals do not conform to the format specified in the configuration fields of this module, operation is not guaranteed.

### (4) Master Receiver

This mode allows the module to receive serial data from another device. The clock and word select signals are internally derived from the oversampling clock. The format of these signals is defined in the configuration fields of this module. If the incoming data from another device does not follow the configured format, operation is not guaranteed.

### (5) Master Transmitter

This mode allows the module to transmit serial data to another device. The clock and word select signals are internally derived from the oversampling clock. The format of these signals is defined in the configuration fields of this module.

### (6) Master Transceiver

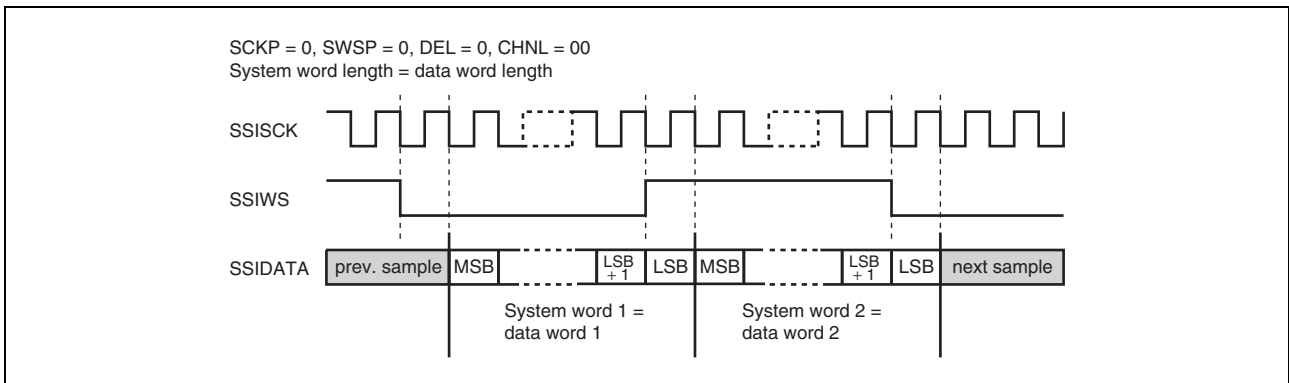
This mode allows serial data transmission and reception between this module and another device. The clock and word select signals are internally derived from the oversampling clock. The format of these signals is defined in the configuration fields of this module.

### (7) Operation Setting Related to Word Length

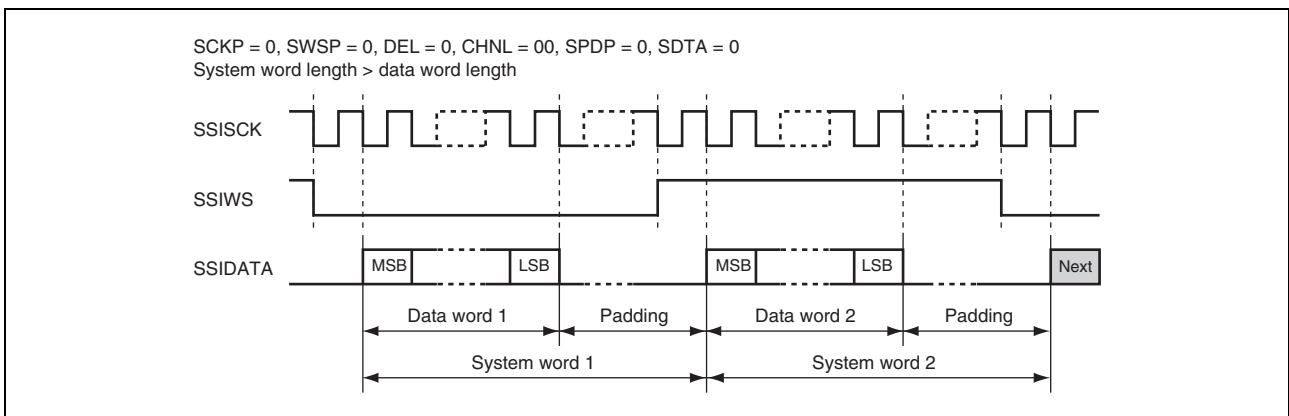
All bits related to the SSICR's word length are valid in non-compressed modes. There are many configurations this module supports, but some of the combinations are shown below for the I<sup>2</sup>S compatible format, MSB-first and left-aligned format, and MSB-first and right-aligned format.

- I<sup>2</sup>S Compatible Format

Figure 19.2 and Figure 19.3 show the I<sup>2</sup>S compatible formats without and with padding, respectively. Padding occurs when the data word length is smaller than the system word length.



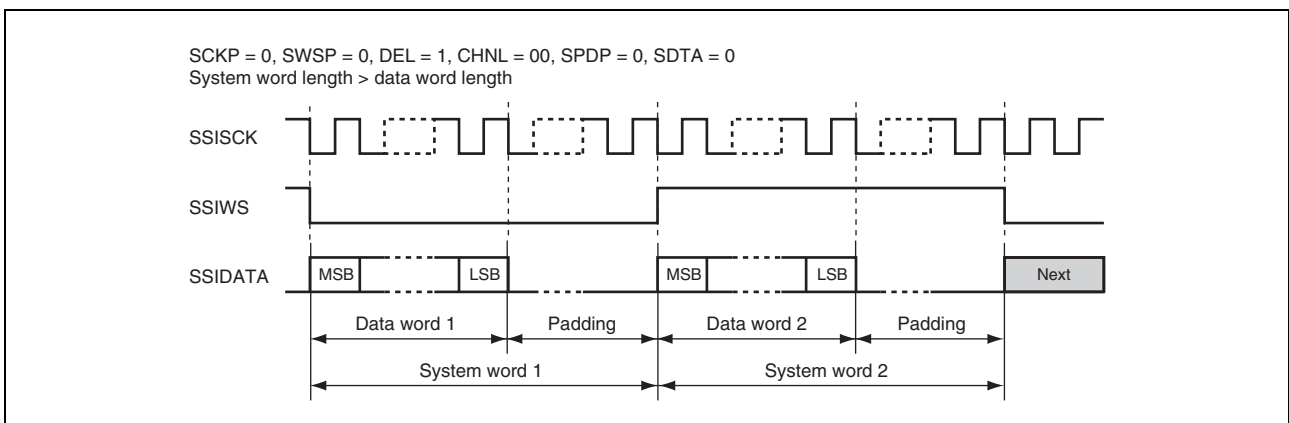
**Figure 19.2 I<sup>2</sup>S Compatible Format (without Padding)**



**Figure 19.3 I<sup>2</sup>S Compatible Format (with Padding)**

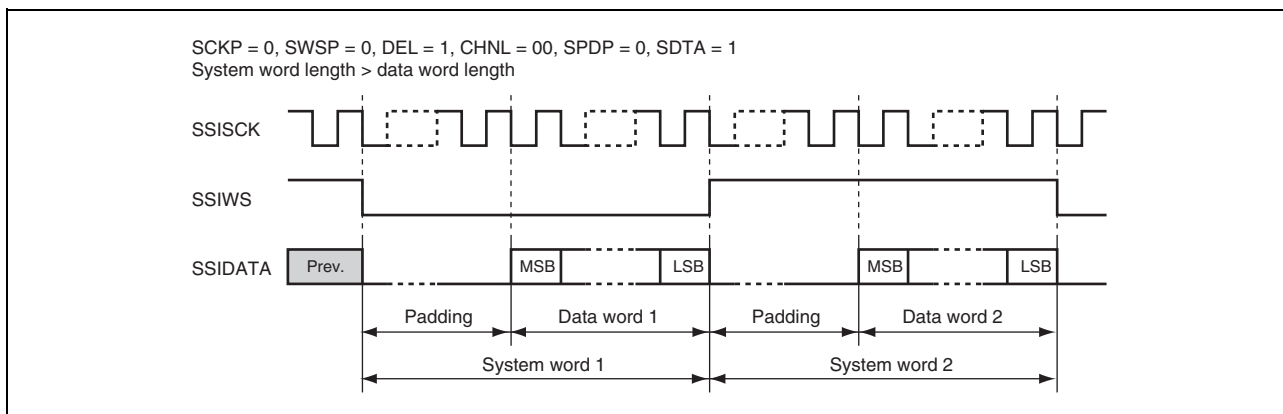
Figure 19.4 shows the MSB-first and left-aligned format and Figure 19.5 shows the MSB-first and right-aligned format.

- MSB-first and Left-aligned Format



**Figure 19.4 MSB-first and Left-aligned Format (Transmitted and Received in the Order of Serial Data and Padding Bits)**

- MSB-first and Right-aligned Format



**Figure 19.5 MSB-first and Right-aligned Format (Transmitted and Received in the Order of Padding Bits and Serial Data)**

### (8) Multi-channel Formats

Some devices extend the definition of the I<sup>2</sup>S bus specification and allow more than 2 channels to be transferred within two system words.

This module supports the transfer of 4, 6, and 8 channels by using the CHNL, SWL and DWL bits only when the system word length (SWL) is greater than or equal to the data word length (DWL) multiplied by channels (CHNL).

Table 19.4 shows the number of padding bits for each of the valid setting. If setting is not valid, "—" is indicated instead of a number.

**Table 19.4 The Number of Padding Bits for Each Valid Setting**

Padding Bits per System Word		DWL[2:0]	000	001	010	011	100	101	110	
CHNL [1:0]	Decoded Channels per System Word	SWL [2:0]	Decoded Word Length	8	16	18	20	22	24	32
00	1	000	8	0	—	—	—	—	—	—
		001	16	8	0	—	—	—	—	—
		010	24	16	8	6	4	2	0	—
		011	32	24	16	14	12	10	8	0
		100	48	40	32	30	28	26	24	16
		101	64	56	48	46	44	42	40	32
		110	128	120	112	110	108	106	104	96
		111	256	248	240	238	236	234	232	224
01	2	000	8	—	—	—	—	—	—	—
		001	16	0	—	—	—	—	—	—
		010	24	8	—	—	—	—	—	—
		011	32	16	0	—	—	—	—	—
		100	48	32	16	12	8	4	0	—
		101	64	48	32	28	24	20	16	0
		110	128	112	96	92	88	84	80	64
		111	256	240	224	220	216	212	208	192
10	3	000	8	—	—	—	—	—	—	—
		001	16	—	—	—	—	—	—	—
		010	24	0	—	—	—	—	—	—
		011	32	8	—	—	—	—	—	—
		100	48	24	0	—	—	—	—	—
		101	64	40	16	10	4	—	—	—
		110	128	104	80	74	68	62	56	32
		111	256	232	208	202	196	190	184	160
11	4	000	8	—	—	—	—	—	—	—
		001	16	—	—	—	—	—	—	—
		010	24	—	—	—	—	—	—	—
		011	32	0	—	—	—	—	—	—
		100	48	16	—	—	—	—	—	—
		101	64	32	0	—	—	—	—	—
		110	128	96	64	56	48	40	32	0
		111	256	224	192	184	176	168	160	128

When this module acts as a transmitter, each word written to SSITDR is transmitted to the serial audio bus in the order they are written. When this module acts as a receiver, each word received by the serial audio bus is read from the SSIRDR register in the order they are received.

Figure 19.6 to Figure 19.8 show how the data on 4, 6, and 8 channels are transferred to the serial audio bus. Note that there are no padding bits in the first example (Figure 19.6), the second example (Figure 19.7) is left-aligned and the third (Figure 19.8) is right-aligned. The other conditions in these examples have been selected arbitrarily.

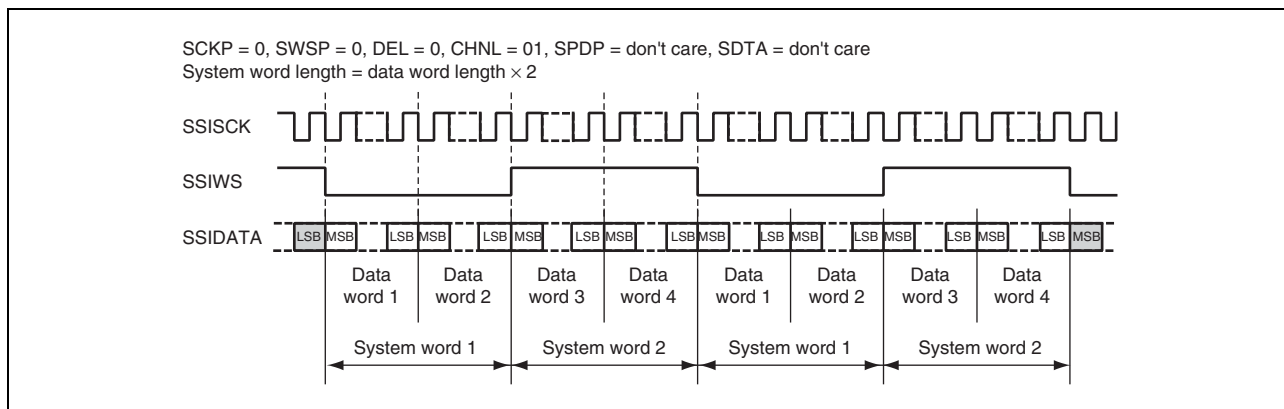


Figure 19.6 Multi-Channel Format (4 Channels Without Padding)

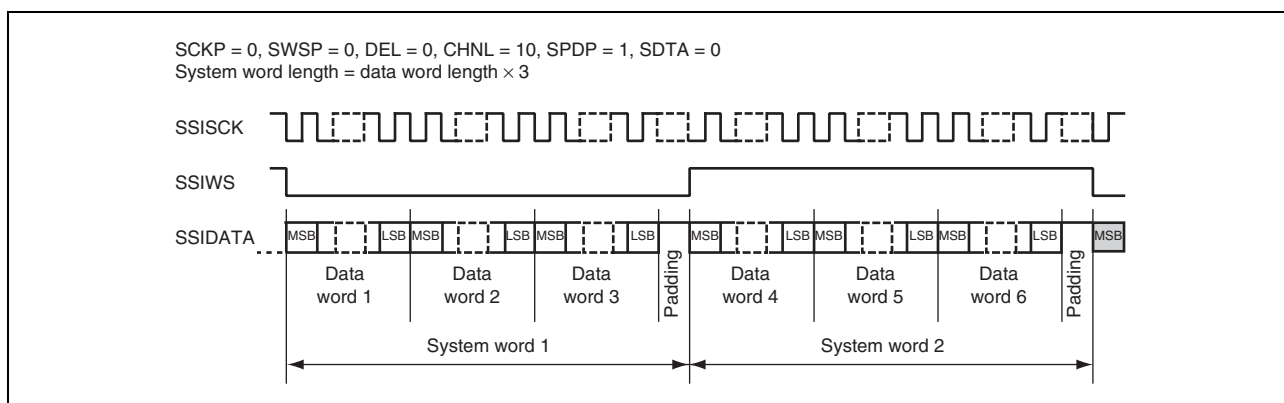


Figure 19.7 Multi-Channel Format (6 Channels with High Padding)

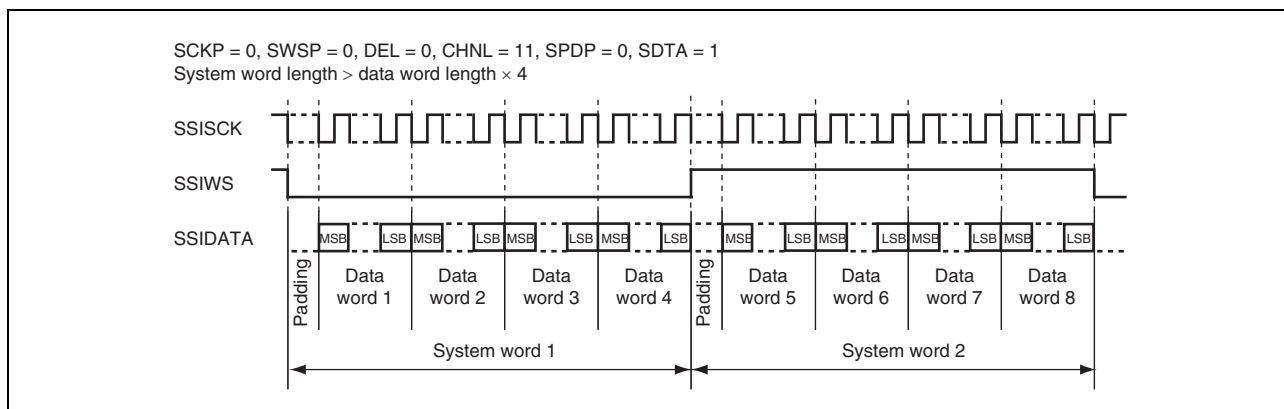


Figure 19.8 Multi-Channel Format (8 Channels; Transmitting and Receiving in the Order of Padding Bits and Serial Data ; with Padding)

(9) Operation Format Configuration Bits

Several more configuration bits in non-compressed mode are shown below. These bits are not mutually exclusive, but some combinations may not be useful.

These configuration bits are described below with reference to the basic sample format in Figure 19.9.

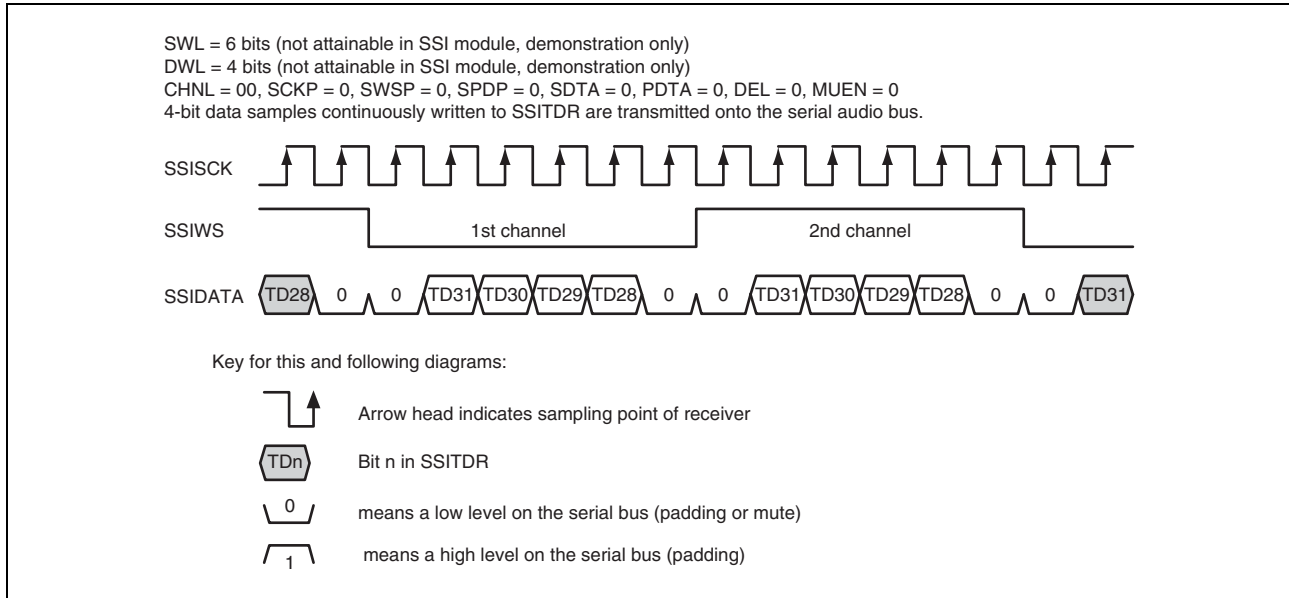


Figure 19.9 Basic Sample Format (Transmit Mode with Example System/Data Word Length)

Figure 19.9 uses a system word length of 6 bits and a data word length of 4 bits. These settings are not possible with this module but are used as an example only for clarification of the other configuration bits.

- Inverted Clock

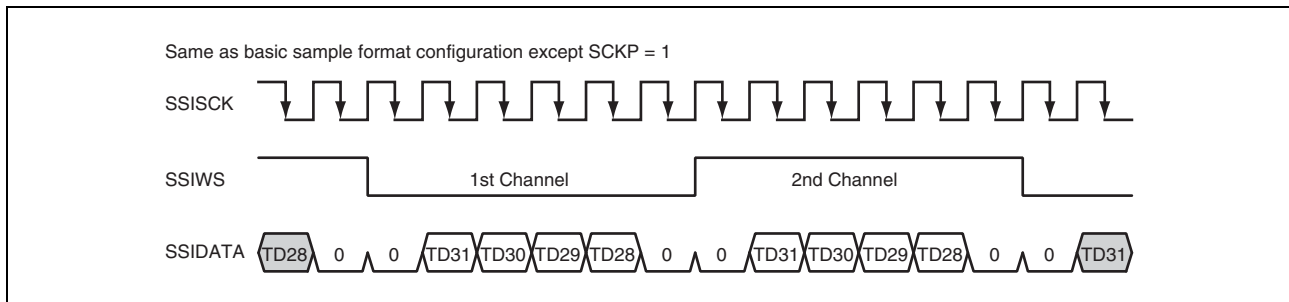


Figure 19.10 Inverted Clock

- Inverted Word Select

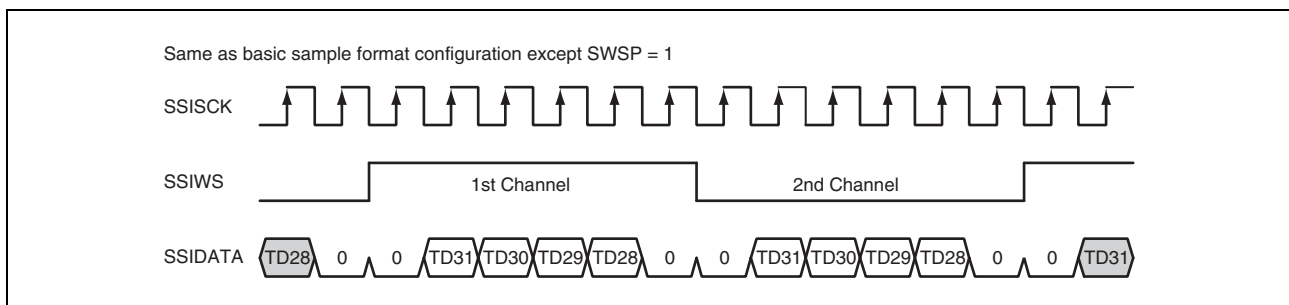


Figure 19.11 Inverted Word Select

- Inverted Padding Polarity

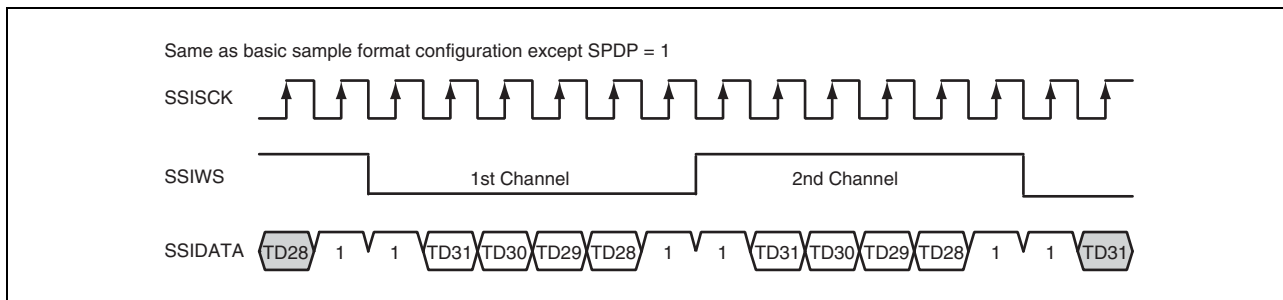


Figure 19.12 Inverted Padding Polarity

- Transmitting and Receiving in the Order of Padding Bits and Serial Data; with Delay

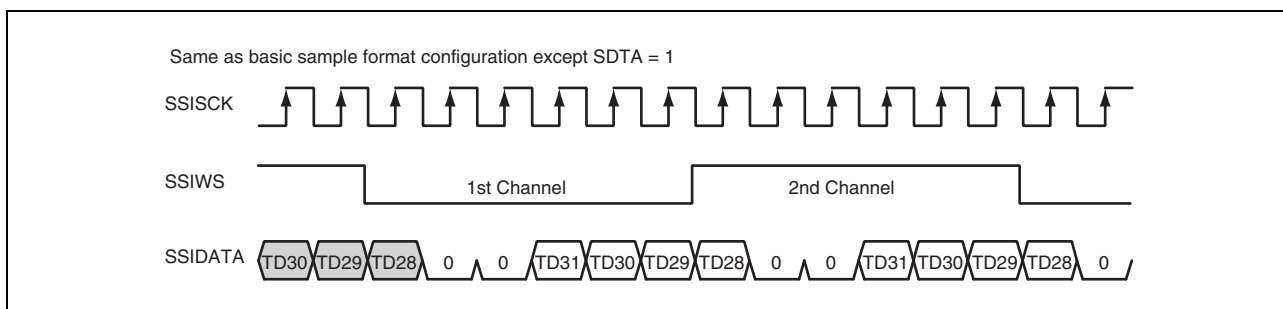


Figure 19.13 Transmitting and Receiving in the Order of Padding Bits and Serial Data; with Delay

- Transmitting and Receiving in the Order of Padding Bits and Serial Data; without Delay

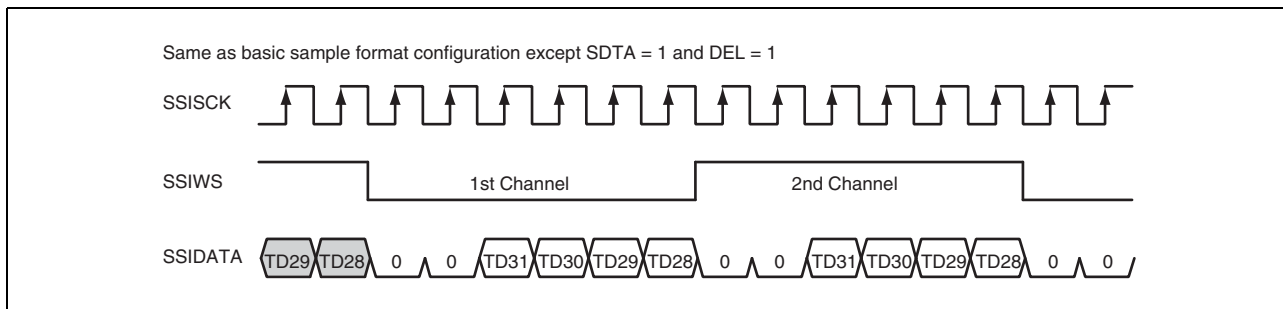


Figure 19.14 Transmitting and Receiving in the Order of Padding Bits and Serial Data; without Delay

- Transmitting and Receiving in the Order of Serial Data and Padding Bits; without Delay

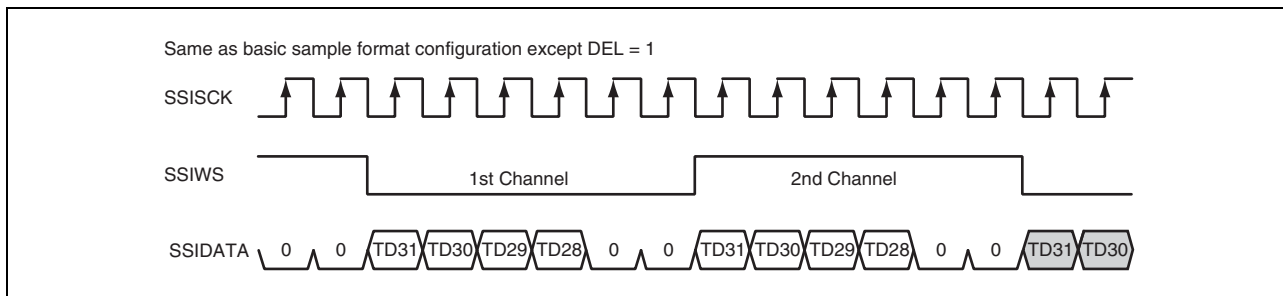
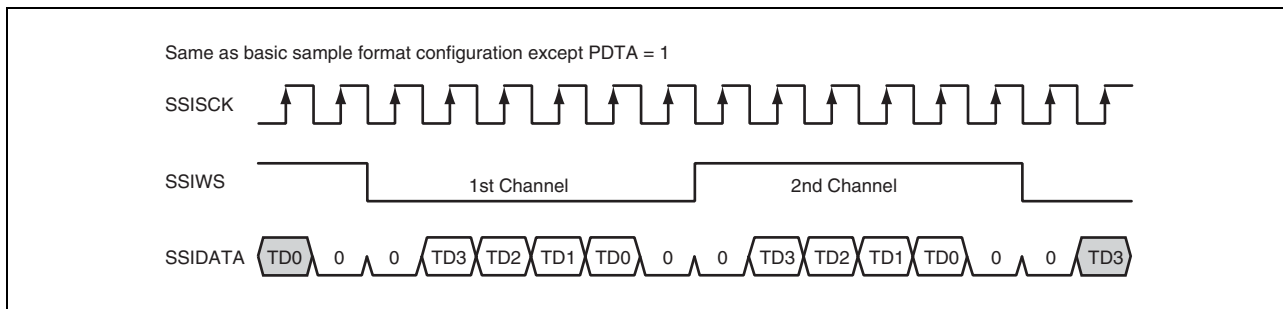


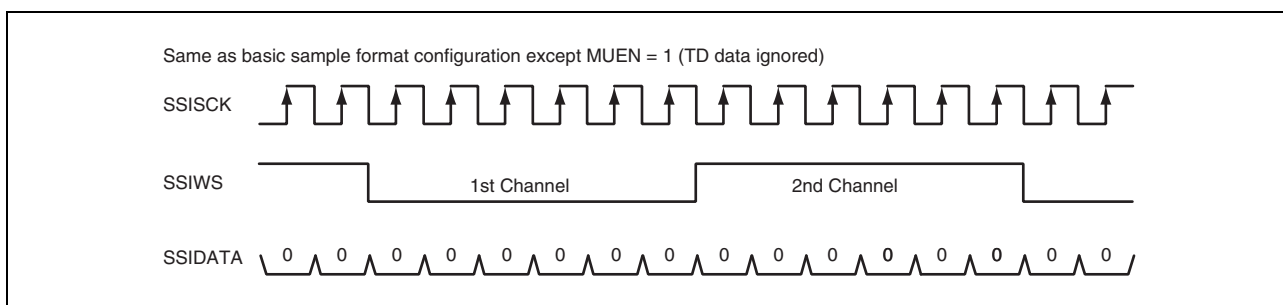
Figure 19.15 Transmitting and Receiving in the Order of Serial Data and Padding Bits; without Delay

- Parallel Right-Aligned with Delay



**Figure 19.16 Parallel Right-Aligned with Delay**

- Mute Enabled



**Figure 19.17 Mute Enabled**



### 19.4.3 TDM Mode

TDM mode is provided to enable connection to multi-channel devices for TDM. This mode can be set using the TDM bit in the TDM mode register (SSITDMR). In this mode, the SSIWS signal is high only for system word 1 period and low for the other periods. The pulse produced on the SSIWS signal is defined as SYNC pulse. Note that the SYNC pulse always has the positive polarity (high only for system word 1 period).

Figure 19.18 and Figure 19.19 show the TDM formats without and with padding, respectively.

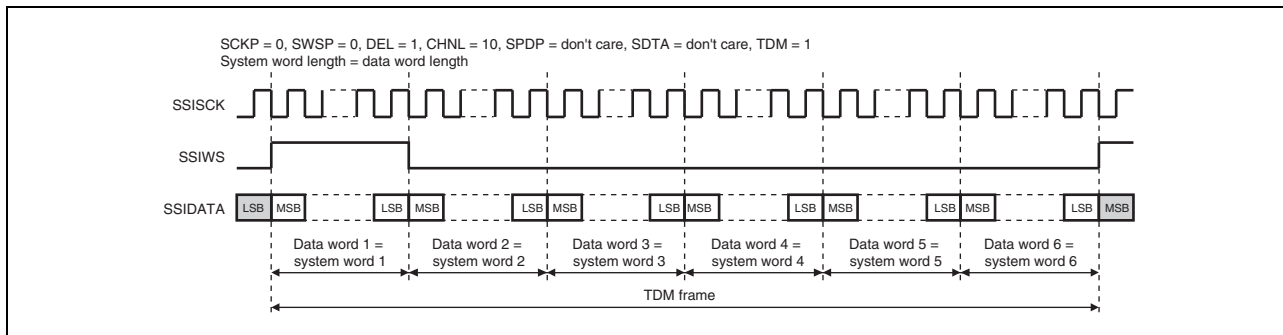


Figure 19.18 TDM Format (6 system words, no padding)

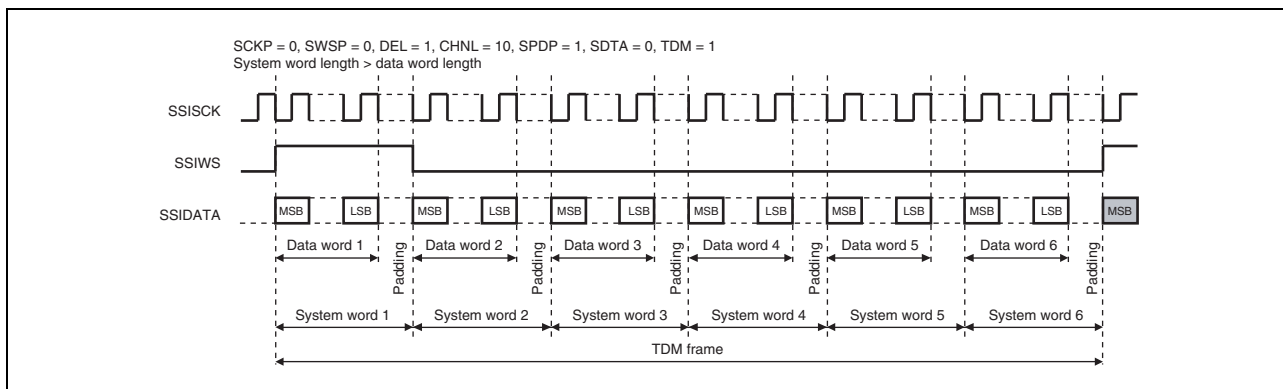


Figure 19.19 TDM Format (6 system words, with padding)

### 19.4.4 WS Continue Mode

In WS continue mode, the SSIWS signal continues to be output irrespective whether data transfer is enabled or disabled. This mode can be set using the CONT bit in the TDM mode register (SSITDMR). With this mode enabled, the SSIWS signal does not stop but continues operating even if TEN and REN bits in the control register (SSICR) are both set to 0 (transfer disabled). While transfer is disabled, the SSITxD pin outputs 0 if the MUEN bit in SSICR is set to 0, and outputs the value specified by the SPDP bit in SSICR if the MUEN bit is set to 1. With this mode disabled, the SSIWS signal stops if TEN and REN bits are both set to 0.

Figure 19.20 and Figure 19.21 show the operations with WS continue mode enabled and disabled, respectively.

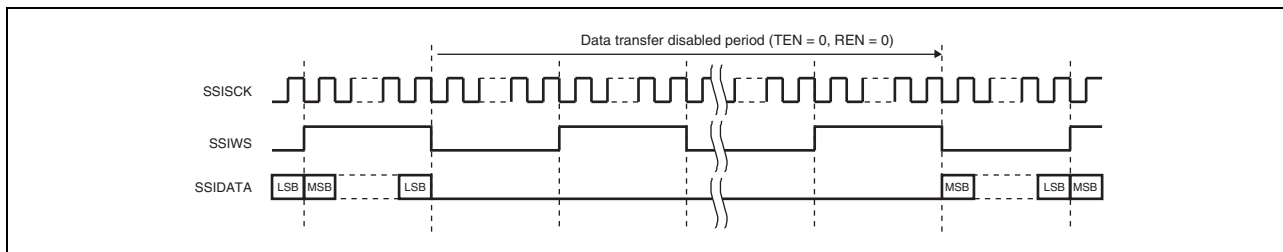


Figure 19.20 WS Continue Mode Enabled

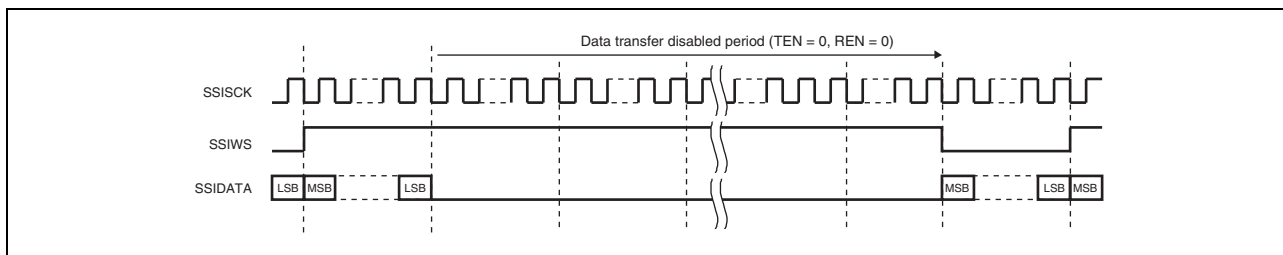


Figure 19.21 WS Continue Mode Disabled

### 19.4.5 Operation Modes

There are three modes of operation: configuration, enabled and disabled. Figure 19.22 shows how the module enters each of these modes.

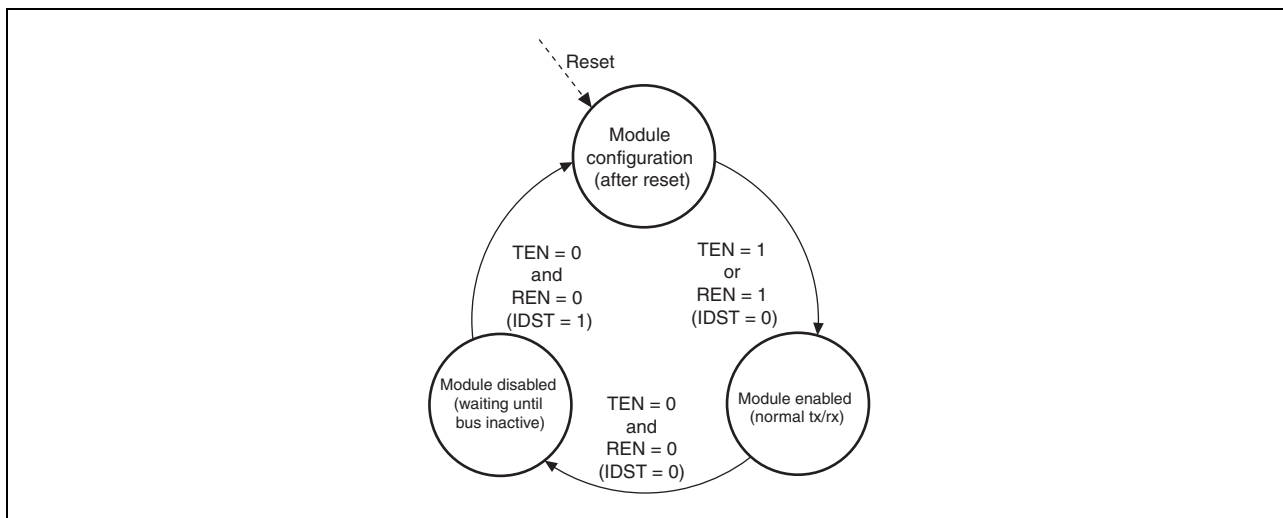


Figure 19.22 Operation Modes

### (1) Configuration Mode

This mode is entered after the module is released from reset. All required configuration fields in the control register should be defined in this mode, before this module is enabled by setting the TEN and REN bits.

Setting the TEN and REN bits causes the module to enter the module enabled mode.

### (2) Module Enabled Mode

Operation of the module in this mode is dependent on the operation mode selected. For details, refer to [section 19.4.6, Transmit Operation](#), and [section 19.4.7, Receive Operation](#), below.

## 19.4.6 Transmit Operation

Transmission can be controlled either by DMA transfer or interrupt.

DMA control is preferred to reduce the processor load. In DMA control mode, the processor will only receive interrupts if there is an underflow or overflow of data or if the DMA transfer has been completed.

The alternative method is using the interrupts that this module generates to supply data as required.

When disabling this module, the clock\* must be kept supplied to this module until the IIRQ bit indicates that the module is in the idle state.

[Figure 19.23](#) shows the transmit operation in DMA control mode, and [Figure 19.24](#) shows the transmit operation in interrupt control mode.

Note: \* Input clock from the SSISCK pin when SCKD = 0.

Oversampling clock when SCKD = 1.

## (1) Transmission Using Direct Memory Access Controller

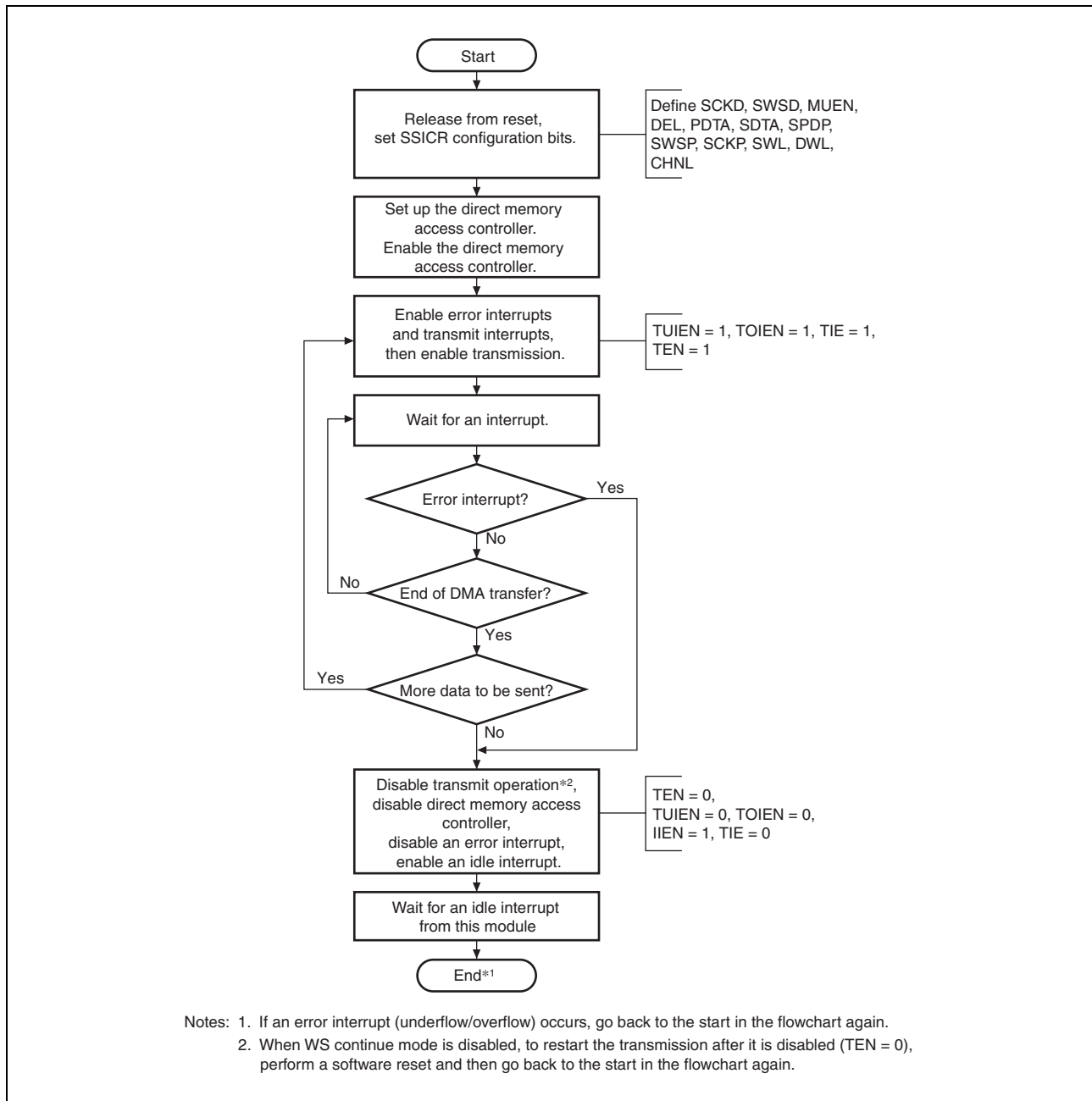


Figure 19.23 Transmission Using Direct Memory Access Controller

## (2) Transmission Using Interrupt-Driven Data Flow Control

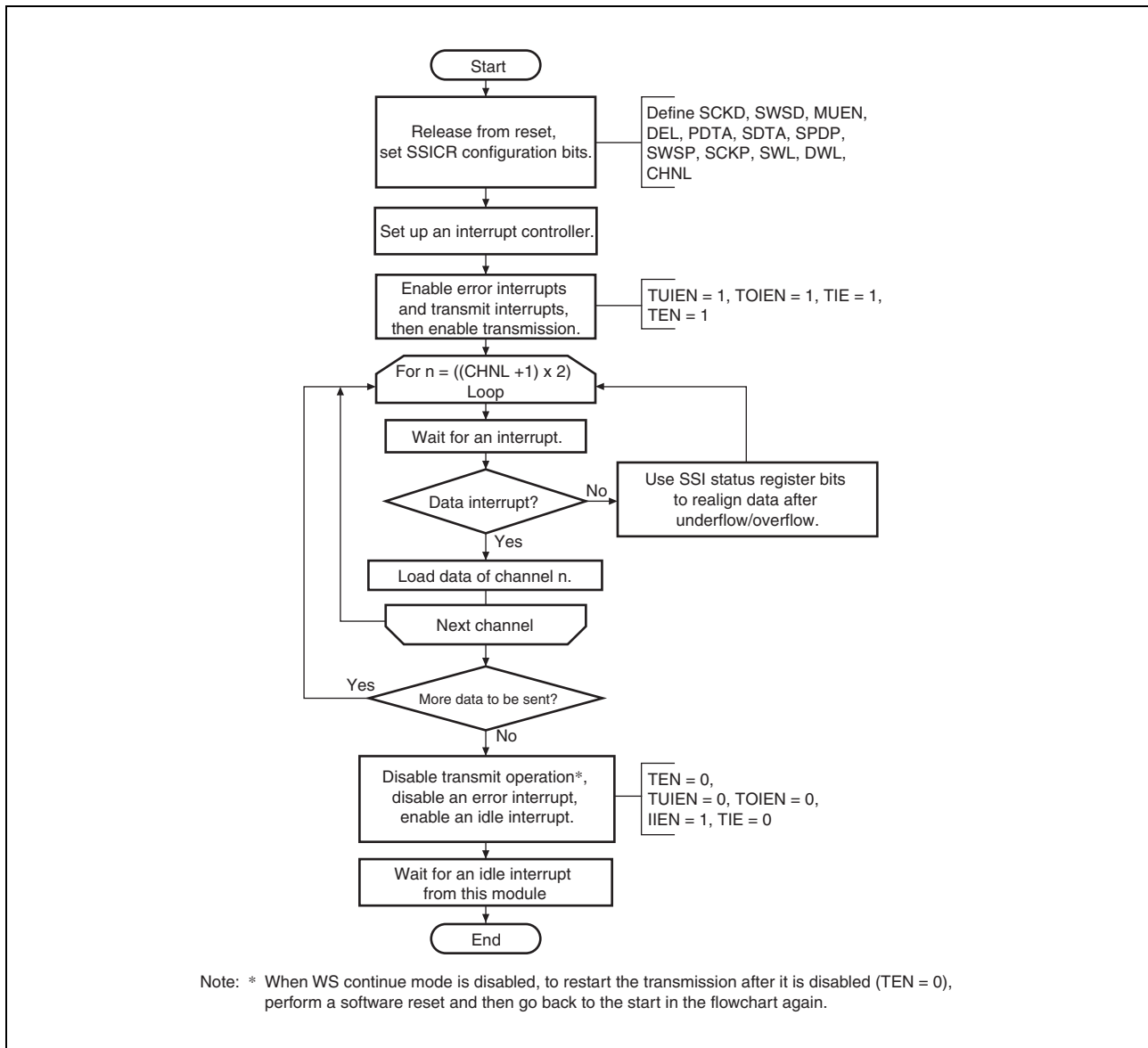


Figure 19.24 Transmission Using Interrupt-Driven Data Flow Control

### 19.4.7 Receive Operation

Like transmission, reception can be controlled either by DMA transfer or interrupt.

Figure 19.25 and Figure 19.26 show the flow of operation.

When disabling this module, the clock\* must be kept supplied to this module until the IIRQ bit indicates that the module is in the idle state.

Note: \* Input clock from the SSISCK pin when SCKD = 0.  
Oversampling clock when SCKD = 1.

#### (1) Reception Using Direct Memory Access Controller

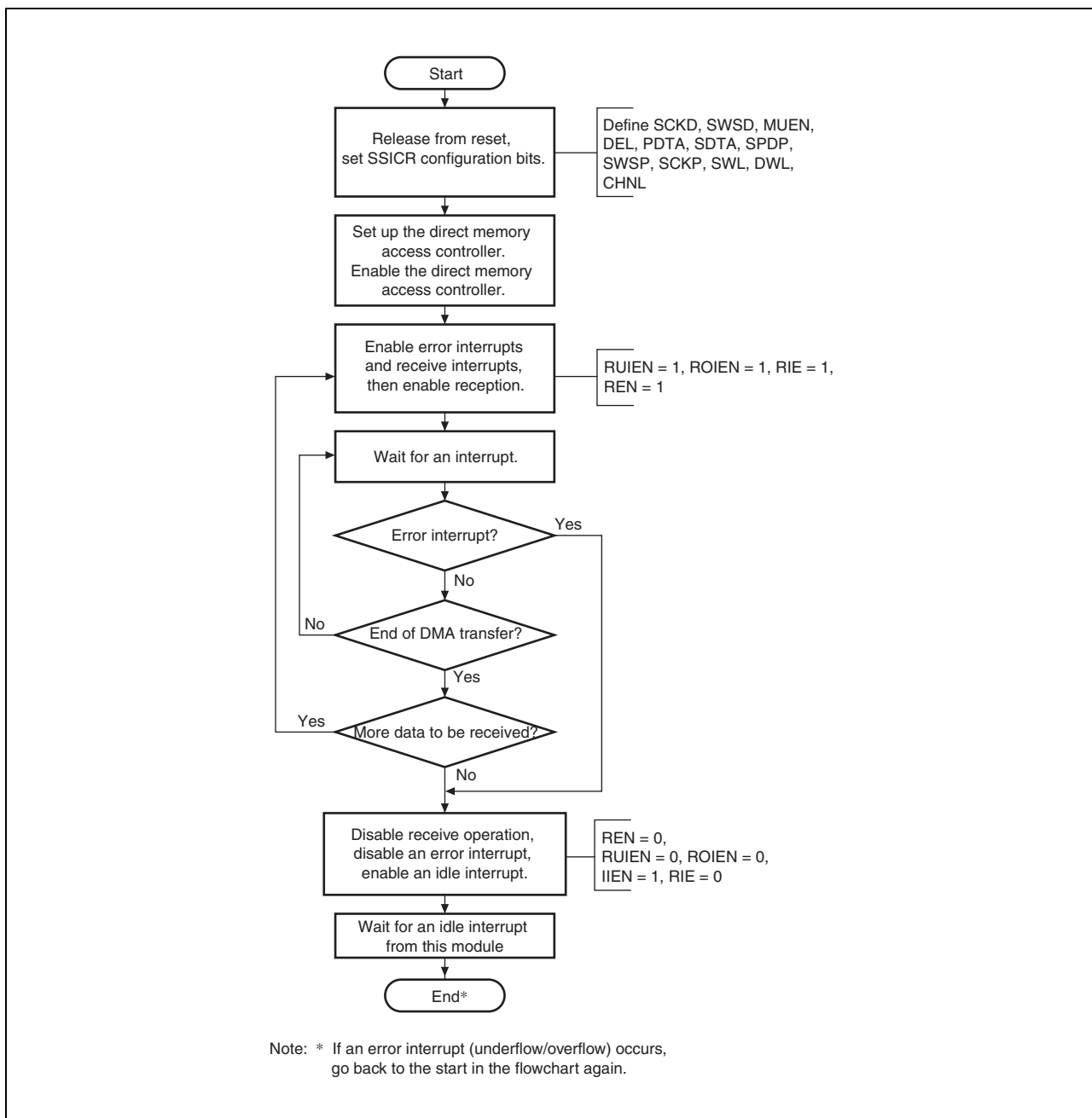
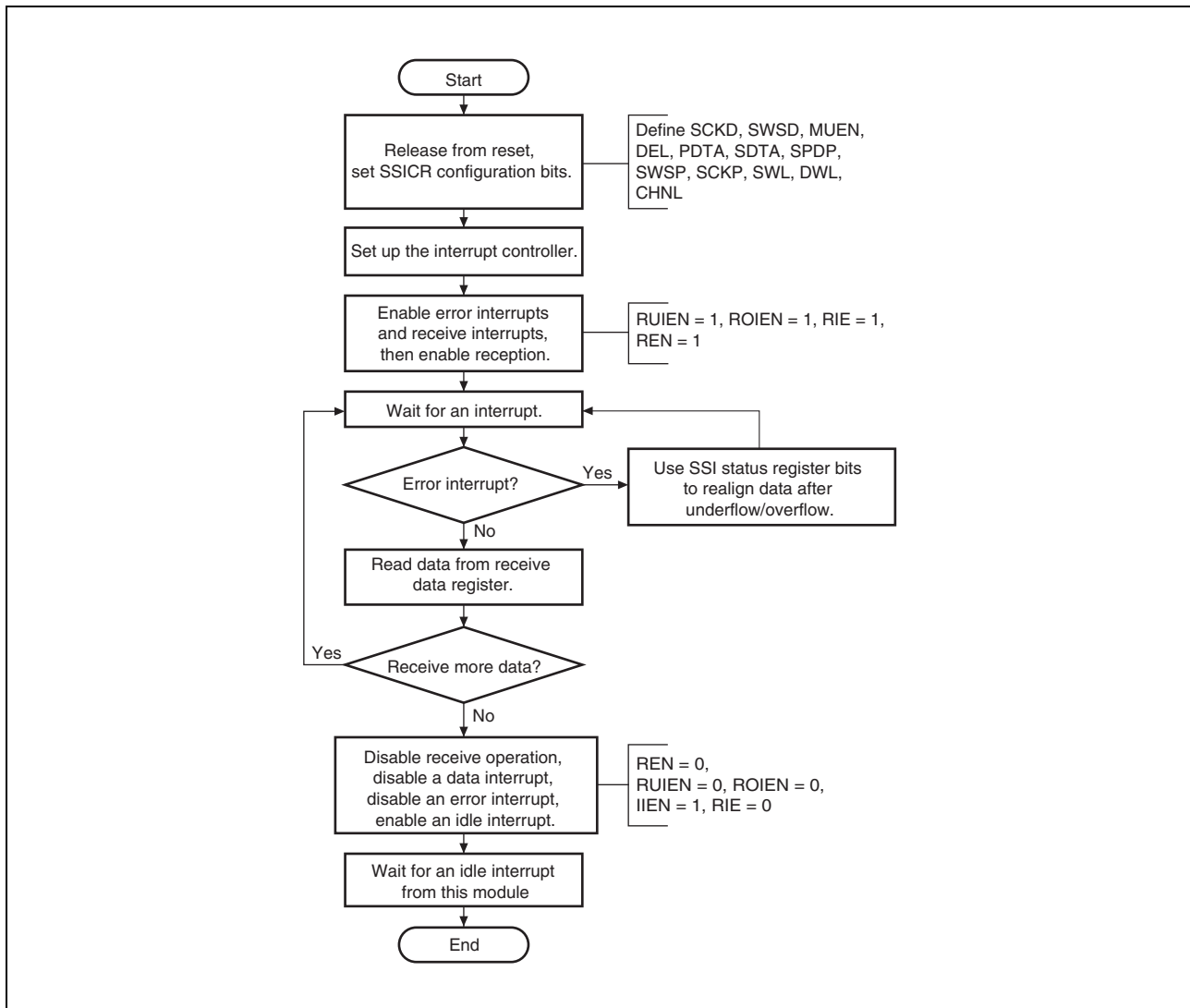


Figure 19.25 Reception Using Direct Memory Access Controller

## (2) Reception Using Interrupt-Driven Data Flow Control



**Figure 19.26 Reception Using Interrupt-Driven Data Flow Control**

When an underflow or overflow error condition has matched, this module can be recovered to the status before underflow or overflow condition match by using the TCHNO [1:0] and TSWNO bits in transmission and the RCHNO[1:0] and RSWNO bits in reception. When an underflow or overflow occurs, the host CPU can read the channel number and system word number to determine what point the serial audio stream has reached. In the transmitter case, the host CPU can skip forward through the data it wants to transmit until it finds the sample data that matches what this module is expecting to transmit next, and so resynchronize with the audio data stream. In the receiver case the host CPU can store null data to make the number of receive data items consistent until it is ready to store the sample data that this module is indicating will be received next, and so resynchronize with the audio data stream.

### 19.4.8 Serial Bit Clock Control

This function is used to control and select the clock that is used for the serial bus interface.

If the serial bit clock direction is set to input ( $SCKD = 0$ ), this module is in clock slave mode and the shift register uses the bit clock that was input to the SSISCK pin.

If the serial bit clock direction is set to output ( $SCKD = 1$ ), this module is in clock master mode, and the shift register uses the oversampling clock or a divided oversampling clock as the bit clock. The oversampling clock is divided by the ratio specified by the serial oversampling clock division ratio bits (CKDV) in SSICR for use as the bit clock by the shift register.

In either case above, the output of the SSISCK pin is the same as the bit clock.

## 19.5 Usage Notes

### 19.5.1 Limitations from Underflow or Overflow during DMA Operation

If an underflow or overflow occurs while the DMA is in operation, the module should be restarted. The transmit and receive buffers in the SSIF consists of 32-bit registers that share the L and R channels. Therefore, data to be transmitted and received at the L channel may sometimes be transmitted and received at the R channel if an underflow or overflow occurs, for example, under the following condition: the control register (SSICR) has a 32-bit setting for both data word length (DWL2 to DWL0) and system word length (SWL2 to SWL0).

If an error occurrence is confirmed with four types of error interrupts (transmit underflow, transmit overflow, receive underflow, and receive overflow) or the corresponding error status flags (the bits TUIRQ, TOIRQ, RUIRQ, and ROIRQ in SSISR), write 0 to the TEN or REN bit in SSICR to disable DMA transfer requests in this module, thus stopping the operation. (In this case, the direct memory access controller setting should also be stopped.) After this, if reception had been in progress, write 0 to the error status flag bit to clear it, set the direct memory access controller again, and restart the transfer. For transmission, issue a software reset and execute the procedure to start again.

### 19.5.2 Note on Changing Mode from Master Transceiver to Master Receiver

If a transmit underflow occurs in master transceiver mode while WS continue mode is disabled ( $SSITDMR.CONT = 0$ ) and the TEN bit in SSICR is set to 0 in order to disable transmit operation, SSIWS output is broken. In order to receive seamlessly after changing mode to master receiver mode, write dummy data to SSIFTDR to suppress transmit underflow.

### 19.5.3 Limits on TDM mode and WS Continue Mode

If TDM mode or WS continue mode setting is changed, the operation of the SSISCK and SSIWS signals immediately after switching are not guaranteed. If it affects the device to be connected, do not change the setting dynamically.

To temporarily halt and restart transmission while the WS continue mode is enabled ( $SSITDMR.CONT = 1$ ), after writing to the transmit FIFO data register (SSIFTDR) a multiple of two times, use the transmit underflow error interrupt or the corresponding error status flag (SSISR.TUIRQ) to confirm that an error has occurred, and then write 0 to the TEN bit of the SSISCR register.

Note that after the transmit underflow error, the last value written to SSIFTDR will be repeatedly sent as long as  $SSISCR.TEN = 1$ . Therefore, write a dummy value as the last data for transmission or mute the signal by writing 1 to the MUEN bit of the SSISCR register.

To restart transmission, do not apply a software reset; after writing 0 to the error status flag bit to clear it, use the idle mode status flag (SSISR.IDST) to confirm that this module is in the idle state, and then write 1 to the TEN bit of the SSISCR register.



## 20. Media Local Bus

This product supports the media local bus as an interface to connect to the MOST controller.

### 20.1 Features

- Supporting MediaLB (including OS62400 module from SMSC) to connect to the INIC from SMSC
- 3-pin interface
- A maximum of 50 Mbps of data can be transferred.

For details on the functions and registers, refer to the specifications for the MediaLB Device Interface Macro OS62420 (document number: DS62400AP5) from SMSC.

Figure 20.1 shows the block diagram.

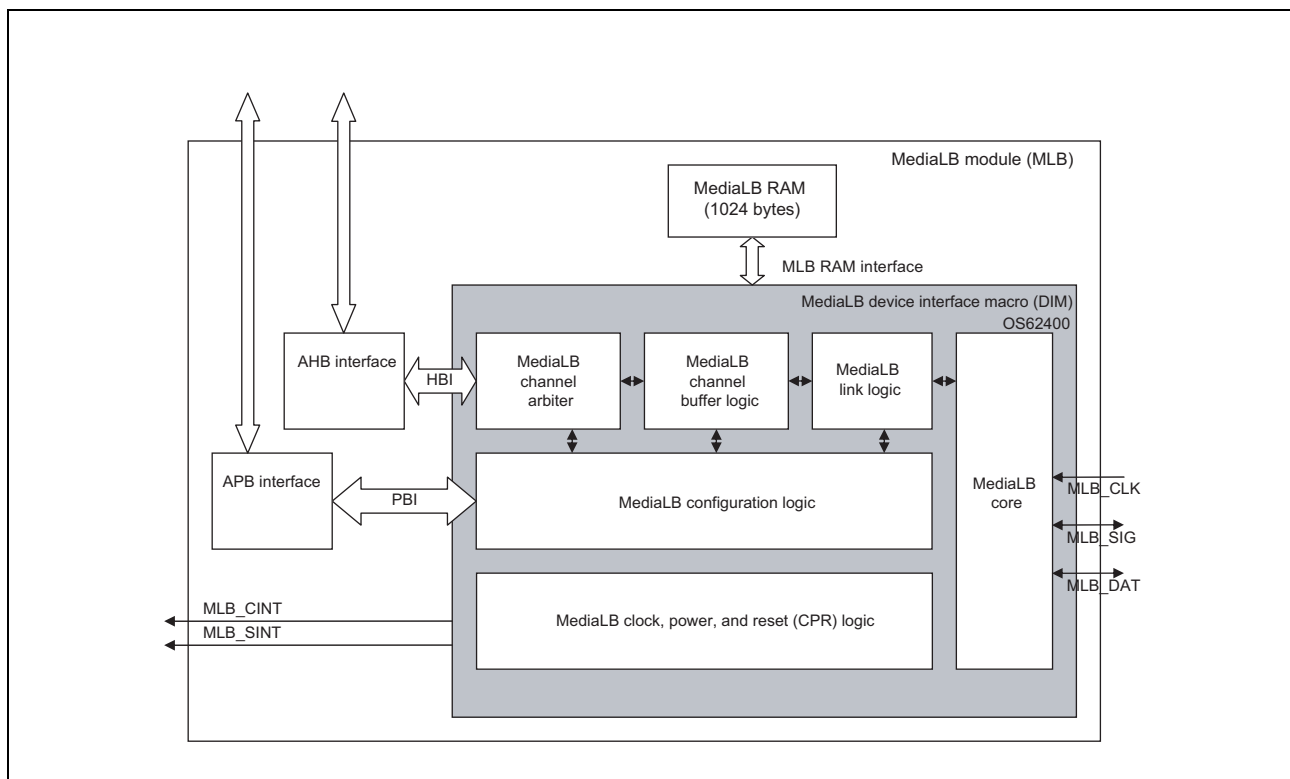


Figure 20.1 Block Diagram

## 20.2 Input/Output Pins

Table 20.1 shows the pin configuration.

**Table 20.1 Pin Configuration**

Pin Name	I/O	Description
MLB_CLK	I	MediaLB clock input
MLB_SIG	I/O	MediaLB signal information I/O
MLB_DAT	I/O	MediaLB data I/O

## 20.3 Register Description

Table 20.2 shows the register configuration.

For details on the registers, refer to the specifications for the MediaLB Device Interface Macro OS62420 (document number: DS62400AP5) from SMSC.

**Table 20.2 Register Configuration**

	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Configuration control register	Device control configuration register	DCCR	*1	*1	H'E8034000	32
	System status configuration register	SSCR	*1	*1	H'E8034004	32
	System data configuration register	SDCR	*1	*1	H'E8034008	32
	System mask configuration register	SMCR	*1	*1	H'E803400C	32
	Version control configuration register	VCCR	*1	*1	H'E803401C	32
	Synchronous base address configuration register	SBCR	*1	*1	H'E8034020	32
	Asynchronous base address configuration register	ABCR	*1	*1	H'E8034024	32
	Control base address configuration register	CBCR	*1	*1	H'E8034028	32
	Isochronous base address configuration register	IBCR	*1	*1	H'E803402C	32
	Channel interrupt configuration register	CICR	*1	*1	H'E8034030	32
Channel m*2 configuration register	Channel m entry configuration register	CECRm	*1	*1	H'E8034040 + m x 10	32
	Channel m status configuration register	CSCRm	*1	*1	H'E8034044 + m x 10	32
	Channel m current buffer configuration register	CCBCRm	*1	*1	H'E8034048 + m x 10	32
	Channel m next buffer configuration register	CNBCRm	*1	*1	H'E803404C + m x 10	32
	Local channel m buffer configuration register	LCBCRm	*1	*1	H'E8034280 + m x 4	32

Note 1. Refer to the specifications for the MediaLB Device Interface Macro OS62420.

Note 2. Each channel of the media local bus is identified by "m" (m = 0 to 30).

## 21. CAN Interface

This section gives an overall description of the CAN interface (RS-CAN).

The first section describes the features specific to this LSI, including the number of units and the register base addresses. The subsequent sections describe the RS-CAN's functions and registers.

### 21.1 Overview

#### 21.1.1 Units

This microcontroller incorporates the following number of units of the CAN interface (RS-CAN).

**Table 21.1 Units of RS-CAN**

RS-CAN	
Number of units	1
Name	RSCAN0

The RS-CAN has five channels.

**Table 21.2 Channels of RS-CAN**

RS-CAN	
Number of channels	5
Name	CAN0, CAN1, CAN2, CAN3, CAN4

**Table 21.3 Index**

Index	Meaning	Available Number of CAN Channels and Index
		5 channels
n	Throughout this section, the unit of the RS-CAN is identified by the index "n": for example, RSCANnGCFG is the global configuration register in the RS-CANn unit.	n = 0
m	Throughout this section, the individual channels in the RS-CAN units are identified by the index "m": for example, RSCAN0CmSTS is the channel m status register in the RS-CAN0 unit.	m = 0 to 4
j	The individual receive rule table registers in the RS-CAN units are identified by the index "j": for example, RSCAN0GAFLIDj is the receive rule ID register j in the RS-CAN0 unit.	j = 0 to 15
k	The individual transmit/receive FIFO in the RS-CAN units are identified by the index "k" (k = 0 to channel m × 3 + 2): for example, RSCAN0CFCK is the transmit/receive FIFO buffer configuration and control register k in the RS-CAN0 unit.	k = 0 to 14
q	The individual receive buffer in the RS-CAN units are identified by the index "q" (q = 0 to channel m × 16 + 15): for example, RSCAN0RMIDq is the receive buffer ID register q in the RS-CAN0 unit.	q = 0 to 79
p	The individual transmit buffers in the RS-CAN units are identified by the index "p" (p = 0 to channel m × 16 + 15): for example, RSCAN0TMIDp is the transmit buffer control register p in the RS-CAN0 unit.	p = 0 to 79
y	When the registers other than above are collectively explained, they are identified by the index "y" (y = 0 to 2): for example, RSCAN0RMNDy is the receive buffer new data register in the RS-CAN0 unit.	y = 0 to 2

### 21.1.2 Register addresses

All RS-CAN register addresses are given as address offsets to the individual base address <RSCAN0\_base>.

The base address <RSCAN0\_base> of each RS-CANn is listed in the following table:

**Table 21.4** Register base address <RSCAN0\_base>

RS-CAN unit	<RSCAN0_base> address
RSCAN0	E803A000h

### 21.1.3 Clock supply

The RS-CAN provides two clock inputs:

**Table 21.5** RS-CAN clock supply

Module	RS-CAN clock Clock	Connected to
RSCAN0	clk_xincan	CAN_CLK
	clk	P1 $\phi$ /2
	pclk	P1 $\phi$

The operating frequency of the RS-CAN depends on the transfer rate and the number of channels in use. **Table 21.6** shows the range of the frequency.

**Table 21.6** Range of Operating Frequency Depending on the Transfer Rate and the Number of Channels in Use in this LSI

Condition		Range of Operating Frequency		
Transfer Rate	No. of Channels in Use	pclk	clk_xincan* <sup>1</sup>	clk* <sup>1, *2</sup>
1Mbps	5ch	pclk $\geq$ 46MHz	8MHz $\leq$ clk_xincan $\leq$ pclk/2	12.5MHz $\leq$ clk $\leq$ pclk/2
	4ch	pclk $\geq$ 40MHz		
	3ch	pclk $\geq$ 32MHz		
	2ch	pclk $\geq$ 26MHz		
	1ch	pclk $\geq$ 18MHz		
500kbps	5ch	pclk $\geq$ 23MHz	4MHz $\leq$ clk_xincan $\leq$ pclk/2	12.5MHz $\leq$ clk $\leq$ pclk/2
	4ch	pclk $\geq$ 20MHz		
	3ch	pclk $\geq$ 16MHz		
	2ch	pclk $\geq$ 13MHz		
	1ch	pclk $\geq$ 8MHz		
125kbps	5ch	pclk $\geq$ 8MHz	4MHz $\leq$ clk_xincan $\leq$ pclk/2	12.5MHz $\leq$ clk $\leq$ pclk/2
	4ch			
	3ch			
	2ch			
	1ch			

Note 1. Setting the DCS bit in RSCAN0GCFG enables to select either clk\_xincan or clk. Set clocks less than or equal to pclk/2.

Note 2. Select clk\_xincan when pclk < 25 MHz.

### 21.1.4 Interrupts

The Controller Area Network (RS-CAN) can generate the interrupt requests shown in the following table.

**Table 21.7 RS-CAN interrupt requests**

Unit Interrupt Name	Outline	Interrupt ID	DMA Trigger Number
<b>RSCAN0</b>			
INTRCANGERR	CAN global error interrupt	253	—
INTRCANGRECC	CAN receive FIFO interrupt	254	—
<b>CAN0</b>			
INTRCANmERR (m = 0)	CAN0 error interrupt	256	—
INTRCANmREC (m = 0)	CAN0 transmit/receive FIFO receive completion interrupt	255	—
INTRCANmTRX (m = 0)	CAN0 transmit interrupt	257	—
<b>CAN1</b>			
INTRCANmERR (m = 1)	CAN1 error interrupt	259	—
INTRCANmREC (m = 1)	CAN1 transmit/receive FIFO receive completion interrupt	258	—
INTRCANmTRX (m = 1)	CAN1 transmit interrupt	260	—
<b>CAN2</b>			
INTRCANmERR (m = 2)	CAN2 error interrupt	262	—
INTRCANmREC (m = 2)	CAN2 transmit/receive FIFO receive completion interrupt	261	—
INTRCANmTRX (m = 2)	CAN2 transmit interrupt	263	—
<b>CAN3</b>			
INTRCANmERR (m = 3)	CAN3 error interrupt	265	—
INTRCANmREC (m = 3)	CAN3 transmit/receive FIFO receive completion interrupt	264	—
INTRCANmTRX (m = 3)	CAN3 transmit interrupt	266	—
<b>CAN4</b>			
INTRCANmERR (m = 4)	CAN4 error interrupt	268	—
INTRCANmREC (m = 4)	CAN4 transmit/receive FIFO receive completion interrupt	267	—
INTRCANmTRX (m = 4)	CAN4 transmit interrupt	269	—

### 21.1.5 I/O signals

Table 21.8 lists the I/O pins of the RS-CAN module.

Table 21.8 I/O Pins of the RS-CAN Module

Unit Signal Name	Outline	Alternative port pin signal
<b>CAN0</b>		
CANmRX (m = 0)	CAN0 receive data input	CAN0RX
CANmTX (m = 0)	CAN0 transmit data output	CAN0TX
<b>CAN1</b>		
CANmRX (m = 1)	CAN1 receive data input	CAN1RX
CANmTX (m = 1)	CAN1 transmit data output	CAN1TX
<b>CAN2</b>		
CANmRX (m = 2)	CAN2 receive data input	CAN2RX
CANmTX (m = 2)	CAN2 transmit data output	CAN2TX
<b>CAN3</b>		
CANmRX (m = 3)	CAN3 receive data input	CAN3RX
CANmTX (m = 3)	CAN3 transmit data output	CAN3TX
<b>CAN4</b>		
CANmRX (m = 4)	CAN4 receive data input	CAN4RX
CANmTX (m = 4)	CAN4 transmit data output	CAN4TX

## 21.2 Function

This LSI incorporates one unit of the CAN interface (RS-CAN) which consists of five channels (CAN0 to CAN4) of the CAN controller conforming to the ISO11898-1 specifications. Table 21.9 shows the RSCAN module specifications.

Figure 21.1 shows the RS-CAN module block diagram.

Table 21.9 RS-CAN Module Specifications (1/3)

Item	Specification
Number of channels	5
Protocol	ISO11898-1 compliant
Communication speed	<ul style="list-style-type: none"> <li>Maximum 1 Mbps</li> </ul>

$$\text{Communication speed (CANm bit time clock)} = \frac{1}{\text{CANm bit time}}$$

$$\text{CANm bit time} = \text{CANmTq} \times \text{Tq count per bit}$$

$$\text{CANmTq} = \frac{(\text{BRP}[9:0] \text{ bits in the RSCAN0CmCFG register} + 1)}{f_{\text{CAN}}}$$

m = 0 to 4

Tq: Time quantum

fCAN: Frequency of CAN clock (selected by the DCS bit in the RSCAN0GCFG register)

Table 21.9 RS-CAN Module Specifications (2/3)

Item	Specification
Buffer	<p>400 buffers in total</p> <ul style="list-style-type: none"> <li>Individual buffers: 80 buffers (16 buffers × 5 channels) Transmit buffer: 16 buffers per channel Transmit queue: Single queue per channel</li> <li>Shared buffers: 320 buffers for all channels Receive buffer: 0 to 80 buffers Receive FIFO buffer: 8 FIFO buffers (up to 128 buffers allocatable to each) Transmit/receive FIFO buffer: 3 FIFO buffers per channel (up to 128 buffers allocatable to each)</li> </ul>
Reception function	<ul style="list-style-type: none"> <li>Receives data frames and remote frames.</li> <li>Selects ID format (standard ID, extended ID, or both IDs) to be received.</li> <li>Sets interrupt enable/disable for each FIFO.</li> <li>Mirror function (CAN mode receives its own transmitted messages.)</li> <li>Timestamp function (to record message reception time as a 16-bit timer value)</li> </ul>
Reception filter function	<ul style="list-style-type: none"> <li>Selects receive messages according to 320 receive rules.</li> <li>Sets the number of receive rules (0 to 128) for each channel.</li> <li>Acceptance filter processing: Sets ID and mask for each receive rule.</li> <li>DLC filter processing: Sets DLC check value for each acceptance rule.</li> </ul>
Receive message transfer function	<ul style="list-style-type: none"> <li>Routing function to transfer receive messages to arbitrary destinations (can be transferred to up to 8 buffers) Transfer destination: Receive buffer, receive FIFO buffer, and/or transmit/receive FIFO buffer</li> <li>Label addition function Stores label information together with a message in a receive buffer and FIFO buffer.</li> </ul>
Transmission function	<ul style="list-style-type: none"> <li>Transmits data frames and remote frames.</li> <li>Selects ID format (standard ID, extended ID, or both IDs) to be transmitted.</li> <li>Sets interrupt enable/disable for each transmit buffer and transmit/receive FIFO buffer.</li> <li>Selects ID priority transmission or transmit buffer number priority transmission.</li> <li>Transmit request can be aborted (possible to confirm with a flag)</li> <li>One-shot transmission function</li> </ul>
Interval transmission function	Transmit messages at intervals (transmit mode or gateway mode of transmit/receive FIFO buffers)
Transmit queue function	Transmits all stored messages according to the ID priority.
Transmit history function	Stores the history information of transmitted messages.
Gateway function	A received message is automatically routed to a different channel.
Bus off recovery mode selection	<p>Selects the method for returning from bus off state.</p> <ul style="list-style-type: none"> <li>ISO11898-1 compliant</li> <li>Automatic entry to channel halt mode at bus-off entry</li> <li>Automatic entry to channel halt mode at bus-off end</li> <li>Transition to channel standby mode by program request</li> <li>Transition to the error-active state by program request</li> </ul>
Error status monitoring	<ul style="list-style-type: none"> <li>Monitors CAN protocol errors (stuff error, form error, ACK error, CRC error, bit error, ACK delimiter error, and bus dominant lock).</li> <li>Detects error status transitions (error warning, error passive, bus off entry, and bus off recovery)</li> <li>Reads the error counter.</li> <li>Monitors DLC errors.</li> </ul>

Table 21.9 RS-CAN Module Specifications (3/3)

Item	Specification
Interrupt source	17 sources <ul style="list-style-type: none"> <li>• Global Interrupts [2 sources: common among channels]</li> <li>Receive FIFO interrupt [1 source: common among channels]</li> <li>Global error interrupt [1 source: common among channels]</li> <li>• Channel interrupts [15 sources: 3 sources × number of channels]</li> <li>CANm transmit interrupt [1 source for each channel]               <ul style="list-style-type: none"> <li>– CANm transmit complete interrupt</li> <li>– CANm transmit abort interrupt</li> <li>– CANm transmit/receive FIFO transmit complete interrupt (in transmit mode, gateway mode)</li> <li>– CANm transmit history interrupt</li> <li>– CANm transmit queue interrupt</li> </ul> </li> <li>CANm transmit/receive FIFO receive complete interrupt (in receive mode, gateway mode) [1 source for each channel]</li> <li>CANm error interrupt [1 source for each channel] (m = 0 to 4)</li> </ul>
CAN stop mode	Reduces power consumption by stopping clock supply to the RS-CAN module.
CAN clock source	Selects the clkc or the clk_xincan. As for the range of operating frequency, refer to <b>Table 21.6</b> .
Test function	Test function for user evaluation <ul style="list-style-type: none"> <li>• Listen-only mode</li> <li>• Self-test mode 0 (external loopback)</li> <li>• Self-test mode 1 (internal loopback)</li> <li>• Inter-channel communication test</li> </ul>



### 21.2.1 Block Diagram

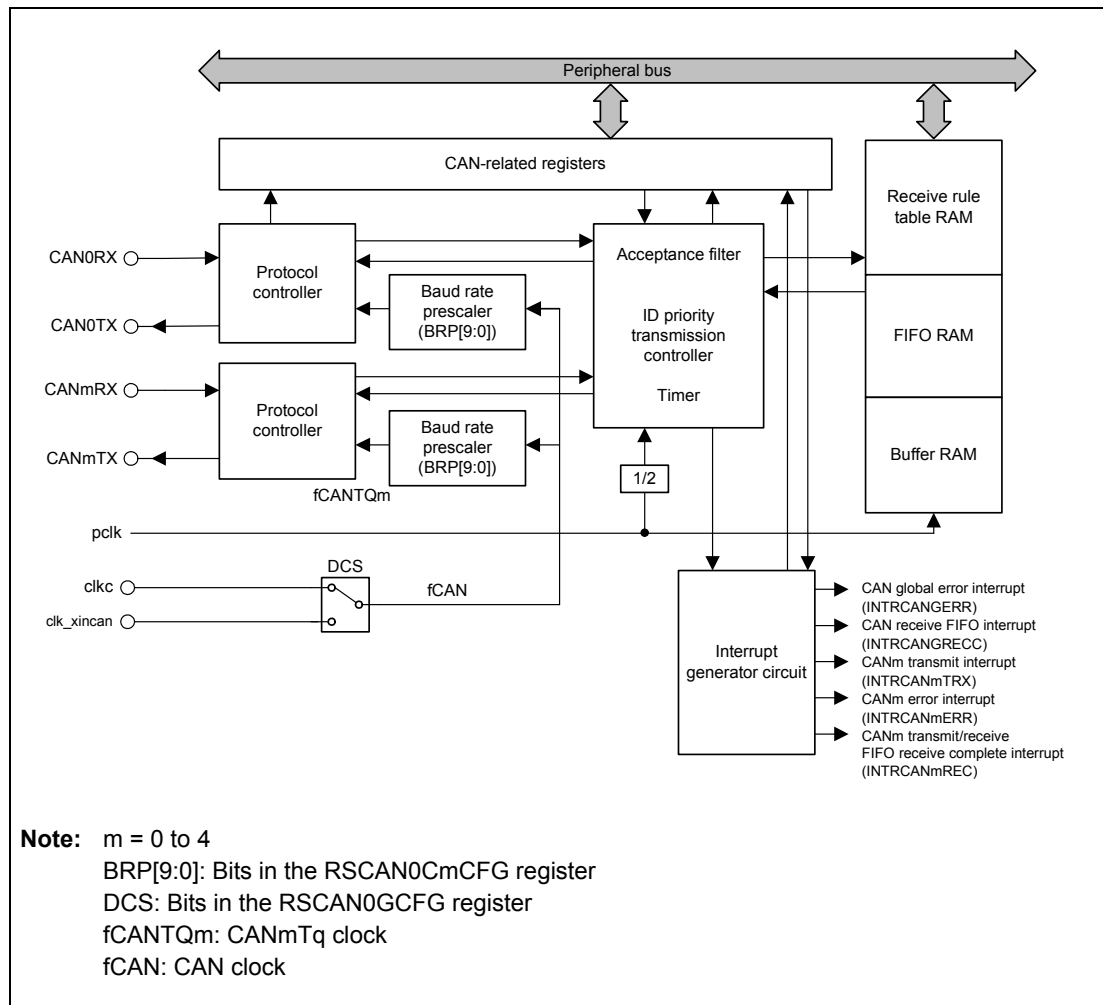


Figure 21.1 RS-CAN Module Block Diagram

## 21.3 Registers

**Table 21.10** lists the registers of the RS-CAN module.n = 0, 1.

**Table 21.10 List of RS-CAN Module Registers (1/25)**

Register Name	Symbol	After Reset	Address	Access Size
Channel 0 configuration register	RSCAN0C0CFG	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0000 <sub>H</sub>	8, 16, 32
Channel 0 control register	RSCAN0C0CTR	0000 0005 <sub>H</sub>	<RSCAN0_base> + 0004 <sub>H</sub>	8, 16, 32
Channel 0 status register	RSCAN0C0STS	0000 0005 <sub>H</sub>	<RSCAN0_base> + 0008 <sub>H</sub>	8, 16, 32
Channel 0 error flag register	RSCAN0C0ERFL	0000 0000 <sub>H</sub>	<RSCAN0_base> + 000C <sub>H</sub>	8, 16, 32
Channel 1 configuration register	RSCAN0C1CFG	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0010 <sub>H</sub>	8, 16, 32
Channel 1 control register	RSCAN0C1CTR	0000 0005 <sub>H</sub>	<RSCAN0_base> + 0014 <sub>H</sub>	8, 16, 32
Channel 1 status register	RSCAN0C1STS	0000 0005 <sub>H</sub>	<RSCAN0_base> + 0018 <sub>H</sub>	8, 16, 32
Channel 1 error flag register	RSCAN0C1ERFL	0000 0000 <sub>H</sub>	<RSCAN0_base> + 001C <sub>H</sub>	8, 16, 32
Channel 2 configuration register	RSCAN0C2CFG	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0020 <sub>H</sub>	8, 16, 32
Channel 2 control register	RSCAN0C2CTR	0000 0005 <sub>H</sub>	<RSCAN0_base> + 0024 <sub>H</sub>	8, 16, 32
Channel 2 status register	RSCAN0C2STS	0000 0005 <sub>H</sub>	<RSCAN0_base> + 0028 <sub>H</sub>	8, 16, 32
Channel 2 error flag register	RSCAN0C2ERFL	0000 0000 <sub>H</sub>	<RSCAN0_base> + 002C <sub>H</sub>	8, 16, 32
Channel 3 configuration register	RSCAN0C3CFG	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0030 <sub>H</sub>	8, 16, 32
Channel 3 control register	RSCAN0C3CTR	0000 0005 <sub>H</sub>	<RSCAN0_base> + 0034 <sub>H</sub>	8, 16, 32
Channel 3 status register	RSCAN0C3STS	0000 0005 <sub>H</sub>	<RSCAN0_base> + 0038 <sub>H</sub>	8, 16, 32
Channel 3 error flag register	RSCAN0C3ERFL	0000 0000 <sub>H</sub>	<RSCAN0_base> + 003C <sub>H</sub>	8, 16, 32
Channel 4 configuration register	RSCAN0C4CFG	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0040 <sub>H</sub>	8, 16, 32
Channel 4 control register	RSCAN0C4CTR	0000 0005 <sub>H</sub>	<RSCAN0_base> + 0044 <sub>H</sub>	8, 16, 32
Channel 4 status register	RSCAN0C4STS	0000 0005 <sub>H</sub>	<RSCAN0_base> + 0048 <sub>H</sub>	8, 16, 32
Channel 4 error flag register	RSCAN0C4ERFL	0000 0000 <sub>H</sub>	<RSCAN0_base> + 004C <sub>H</sub>	8, 16, 32
Global configuration register	RSCAN0GCFG	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0084 <sub>H</sub>	8, 16, 32
Global control register	RSCAN0GCTR	0000 0005 <sub>H</sub>	<RSCAN0_base> + 0088 <sub>H</sub>	8, 16, 32
Global status register	RSCAN0GSTS	0000 000D <sub>H</sub>	<RSCAN0_base> + 008C <sub>H</sub>	8, 16, 32
Global error flag register	RSCAN0GERFL	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0090 <sub>H</sub>	8, 16, 32
Global timestamp counter register	RSCAN0GTSC	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0094 <sub>H</sub>	16, 32
Receive rule entry control register	RSCAN0GAFLECTR	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0098 <sub>H</sub>	8, 16, 32
Receive rule configuration register 0	RSCAN0GAFLCFG0	0000 0000 <sub>H</sub>	<RSCAN0_base> + 009C <sub>H</sub>	8, 16, 32
Receive rule configuration register 1	RSCAN0GAFLCFG1	0000 0000 <sub>H</sub>	<RSCAN0_base> + 00A0 <sub>H</sub>	8, 16, 32
Receive buffer number register	RSCAN0RMNB	0000 0000 <sub>H</sub>	<RSCAN0_base> + 00A4 <sub>H</sub>	8, 16, 32
Receive buffer new data register 0	RSCAN0RMND0	0000 0000 <sub>H</sub>	<RSCAN0_base> + 00A8 <sub>H</sub>	8, 16, 32
Receive buffer new data register 1	RSCAN0RMND1	0000 0000 <sub>H</sub>	<RSCAN0_base> + 00AC <sub>H</sub>	8, 16, 32
Receive buffer new data register 2	RSCAN0RMND2	0000 0000 <sub>H</sub>	<RSCAN0_base> + 00B0 <sub>H</sub>	8, 16, 32
Receive FIFO buffer configuration and control register 0	RSCAN0RFCC0	0000 0000 <sub>H</sub>	<RSCAN0_base> + 00B8 <sub>H</sub>	8, 16, 32
Receive FIFO buffer configuration and control register 1	RSCAN0RFCC1	0000 0000 <sub>H</sub>	<RSCAN0_base> + 00BC <sub>H</sub>	8, 16, 32
Receive FIFO buffer configuration and control register 2	RSCAN0RFCC2	0000 0000 <sub>H</sub>	<RSCAN0_base> + 00C0 <sub>H</sub>	8, 16, 32
Receive FIFO buffer configuration and control register 3	RSCAN0RFCC3	0000 0000 <sub>H</sub>	<RSCAN0_base> + 00C4 <sub>H</sub>	8, 16, 32
Receive FIFO buffer configuration and control register 4	RSCAN0RFCC4	0000 0000 <sub>H</sub>	<RSCAN0_base> + 00C8 <sub>H</sub>	8, 16, 32
Receive FIFO buffer configuration and control register 5	RSCAN0RFCC5	0000 0000 <sub>H</sub>	<RSCAN0_base> + 00CC <sub>H</sub>	8, 16, 32
Receive FIFO buffer configuration and control register 6	RSCAN0RFCC6	0000 0000 <sub>H</sub>	<RSCAN0_base> + 00D0 <sub>H</sub>	8, 16, 32
Receive FIFO buffer configuration and control register 7	RSCAN0RFCC7	0000 0000 <sub>H</sub>	<RSCAN0_base> + 00D4 <sub>H</sub>	8, 16, 32
Receive FIFO buffer status register 0	RSCAN0RFSTS0	0000 0001 <sub>H</sub>	<RSCAN0_base> + 00D8 <sub>H</sub>	8, 16, 32
Receive FIFO buffer status register 1	RSCAN0RFSTS1	0000 0001 <sub>H</sub>	<RSCAN0_base> + 00DC <sub>H</sub>	8, 16, 32
Receive FIFO buffer status register 2	RSCAN0RFSTS2	0000 0001 <sub>H</sub>	<RSCAN0_base> + 00E0 <sub>H</sub>	8, 16, 32

Table 21.10 List of RS-CAN Module Registers (2/25)

Register Name	Symbol	After Reset	Address	Access Size
Receive FIFO buffer status register 3	RSCAN0RFSTS3	0000 0001 <sub>H</sub>	<RSCAN0_base> + 00E4 <sub>H</sub>	8, 16, 32
Receive FIFO buffer status register 4	RSCAN0RFSTS4	0000 0001 <sub>H</sub>	<RSCAN0_base> + 00E8 <sub>H</sub>	8, 16, 32
Receive FIFO buffer status register 5	RSCAN0RFSTS5	0000 0001 <sub>H</sub>	<RSCAN0_base> + 00EC <sub>H</sub>	8, 16, 32
Receive FIFO buffer status register 6	RSCAN0RFSTS6	0000 0001 <sub>H</sub>	<RSCAN0_base> + 00F0 <sub>H</sub>	8, 16, 32
Receive FIFO buffer status register 7	RSCAN0RFSTS7	0000 0001 <sub>H</sub>	<RSCAN0_base> + 00F4 <sub>H</sub>	8, 16, 32
Receive FIFO buffer pointer control register 0	RSCAN0RFPCTR0	—	<RSCAN0_base> + 00F8 <sub>H</sub>	8, 16, 32
Receive FIFO buffer pointer control register 1	RSCAN0RFPCTR1	—	<RSCAN0_base> + 00FC <sub>H</sub>	8, 16, 32
Receive FIFO buffer pointer control register 2	RSCAN0RFPCTR2	—	<RSCAN0_base> + 0100 <sub>H</sub>	8, 16, 32
Receive FIFO buffer pointer control register 3	RSCAN0RFPCTR3	—	<RSCAN0_base> + 0104 <sub>H</sub>	8, 16, 32
Receive FIFO buffer pointer control register 4	RSCAN0RFPCTR4	—	<RSCAN0_base> + 0108 <sub>H</sub>	8, 16, 32
Receive FIFO buffer pointer control register 5	RSCAN0RFPCTR5	—	<RSCAN0_base> + 010C <sub>H</sub>	8, 16, 32
Receive FIFO buffer pointer control register 6	RSCAN0RFPCTR6	—	<RSCAN0_base> + 0110 <sub>H</sub>	8, 16, 32
Receive FIFO buffer pointer control register 7	RSCAN0RFPCTR7	—	<RSCAN0_base> + 0114 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer configuration and control register 0	RSCAN0CFCC0	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0118 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer configuration and control register 1	RSCAN0CFCC1	0000 0000 <sub>H</sub>	<RSCAN0_base> + 011C <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer configuration and control register 2	RSCAN0CFCC2	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0120 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer configuration and control register 3	RSCAN0CFCC3	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0124 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer configuration and control register 4	RSCAN0CFCC4	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0128 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer configuration and control register 5	RSCAN0CFCC5	0000 0000 <sub>H</sub>	<RSCAN0_base> + 012C <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer configuration and control register 6	RSCAN0CFCC6	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0130 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer configuration and control register 7	RSCAN0CFCC7	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0134 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer configuration and control register 8	RSCAN0CFCC8	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0138 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer configuration and control register 9	RSCAN0CFCC9	0000 0000 <sub>H</sub>	<RSCAN0_base> + 013C <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer configuration and control register 10	RSCAN0CFCC10	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0140 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer configuration and control register 11	RSCAN0CFCC11	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0144 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer configuration and control register 12	RSCAN0CFCC12	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0148 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer configuration and control register 13	RSCAN0CFCC13	0000 0000 <sub>H</sub>	<RSCAN0_base> + 014C <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer configuration and control register 14	RSCAN0CFCC14	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0150 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer status register 0	RSCAN0CFSTS0	0000 0001 <sub>H</sub>	<RSCAN0_base> + 0178 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer status register 1	RSCAN0CFSTS1	0000 0001 <sub>H</sub>	<RSCAN0_base> + 017C <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer status register 2	RSCAN0CFSTS2	0000 0001 <sub>H</sub>	<RSCAN0_base> + 0180 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer status register 3	RSCAN0CFSTS3	0000 0001 <sub>H</sub>	<RSCAN0_base> + 0184 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer status register 4	RSCAN0CFSTS4	0000 0001 <sub>H</sub>	<RSCAN0_base> + 0188 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer status register 5	RSCAN0CFSTS5	0000 0001 <sub>H</sub>	<RSCAN0_base> + 018C <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer status register 6	RSCAN0CFSTS6	0000 0001 <sub>H</sub>	<RSCAN0_base> + 0190 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer status register 7	RSCAN0CFSTS7	0000 0001 <sub>H</sub>	<RSCAN0_base> + 0194 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer status register 8	RSCAN0CFSTS8	0000 0001 <sub>H</sub>	<RSCAN0_base> + 0198 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer status register 9	RSCAN0CFSTS9	0000 0001 <sub>H</sub>	<RSCAN0_base> + 019C <sub>H</sub>	8, 16, 32

Table 21.10 List of RS-CAN Module Registers (3/25)

Register Name	Symbol	After Reset	Address	Access Size
Transmit/receive FIFO buffer status register 10	RSCAN0CFSTS10	0000 0001 <sub>H</sub>	<RSCAN0_base> + 01A0 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer status register 11	RSCAN0CFSTS11	0000 0001 <sub>H</sub>	<RSCAN0_base> + 01A4 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer status register 12	RSCAN0CFSTS12	0000 0001 <sub>H</sub>	<RSCAN0_base> + 01A8 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer status register 13	RSCAN0CFSTS13	0000 0001 <sub>H</sub>	<RSCAN0_base> + 01AC <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer status register 14	RSCAN0CFSTS14	0000 0001 <sub>H</sub>	<RSCAN0_base> + 01B0 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer pointer control register 0	RSCAN0CFPCTR0	—	<RSCAN0_base> + 01D8 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer pointer control register 1	RSCAN0CFPCTR1	—	<RSCAN0_base> + 01DC <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer pointer control register 2	RSCAN0CFPCTR2	—	<RSCAN0_base> + 01E0 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer pointer control register 3	RSCAN0CFPCTR3	—	<RSCAN0_base> + 01E4 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer pointer control register 4	RSCAN0CFPCTR4	—	<RSCAN0_base> + 01E8 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer pointer control register 5	RSCAN0CFPCTR5	—	<RSCAN0_base> + 01EC <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer pointer control register 6	RSCAN0CFPCTR6	—	<RSCAN0_base> + 01F0 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer pointer control register 7	RSCAN0CFPCTR7	—	<RSCAN0_base> + 01F4 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer pointer control register 8	RSCAN0CFPCTR8	—	<RSCAN0_base> + 01F8 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer pointer control register 9	RSCAN0CFPCTR9	—	<RSCAN0_base> + 01FC <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer pointer control register 10	RSCAN0CFPCTR10	—	<RSCAN0_base> + 0200 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer pointer control register 11	RSCAN0CFPCTR11	—	<RSCAN0_base> + 0204 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer pointer control register 12	RSCAN0CFPCTR12	—	<RSCAN0_base> + 0208 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer pointer control register 13	RSCAN0CFPCTR13	—	<RSCAN0_base> + 020C <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer pointer control register 14	RSCAN0CFPCTR14	—	<RSCAN0_base> + 0210 <sub>H</sub>	8, 16, 32
FIFO empty status register	RSCAN0FESTS	007F FFFF <sub>H</sub>	<RSCAN0_base> + 0238 <sub>H</sub>	8, 16, 32
FIFO full status register	RSCAN0FFSTS	0000 0000 <sub>H</sub>	<RSCAN0_base> + 023C <sub>H</sub>	8, 16, 32
FIFO Msg lost status register	RSCAN0FMSTS	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0240 <sub>H</sub>	8, 16, 32
Receive FIFO buffer interrupt flag status register	RSCAN0RFISTS	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0244 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer RX interrupt flag status register	RSCAN0CFRISTS	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0248 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer TX interrupt flag status register	RSCAN0CFTISTS	0000 0000 <sub>H</sub>	<RSCAN0_base> + 024C <sub>H</sub>	8, 16, 32
Transmit buffer control register 0	RSCAN0TMC0	00 <sub>H</sub>	<RSCAN0_base> + 0250 <sub>H</sub>	8
Transmit buffer control register 1	RSCAN0TMC1	00 <sub>H</sub>	<RSCAN0_base> + 0251 <sub>H</sub>	8
Transmit buffer control register 2	RSCAN0TMC2	00 <sub>H</sub>	<RSCAN0_base> + 0252 <sub>H</sub>	8
Transmit buffer control register 3	RSCAN0TMC3	00 <sub>H</sub>	<RSCAN0_base> + 0253 <sub>H</sub>	8
Transmit buffer control register 4	RSCAN0TMC4	00 <sub>H</sub>	<RSCAN0_base> + 0254 <sub>H</sub>	8
Transmit buffer control register 5	RSCAN0TMC5	00 <sub>H</sub>	<RSCAN0_base> + 0255 <sub>H</sub>	8
Transmit buffer control register 6	RSCAN0TMC6	00 <sub>H</sub>	<RSCAN0_base> + 0256 <sub>H</sub>	8
Transmit buffer control register 7	RSCAN0TMC7	00 <sub>H</sub>	<RSCAN0_base> + 0257 <sub>H</sub>	8
Transmit buffer control register 8	RSCAN0TMC8	00 <sub>H</sub>	<RSCAN0_base> + 0258 <sub>H</sub>	8
Transmit buffer control register 9	RSCAN0TMC9	00 <sub>H</sub>	<RSCAN0_base> + 0259 <sub>H</sub>	8
Transmit buffer control register 10	RSCAN0TMC10	00 <sub>H</sub>	<RSCAN0_base> + 025A <sub>H</sub>	8
Transmit buffer control register 11	RSCAN0TMC11	00 <sub>H</sub>	<RSCAN0_base> + 025B <sub>H</sub>	8
Transmit buffer control register 12	RSCAN0TMC12	00 <sub>H</sub>	<RSCAN0_base> + 025C <sub>H</sub>	8
Transmit buffer control register 13	RSCAN0TMC13	00 <sub>H</sub>	<RSCAN0_base> + 025D <sub>H</sub>	8
Transmit buffer control register 14	RSCAN0TMC14	00 <sub>H</sub>	<RSCAN0_base> + 025E <sub>H</sub>	8
Transmit buffer control register 15	RSCAN0TMC15	00 <sub>H</sub>	<RSCAN0_base> + 025F <sub>H</sub>	8
Transmit buffer control register 16	RSCAN0TMC16	00 <sub>H</sub>	<RSCAN0_base> + 0260 <sub>H</sub>	8
Transmit buffer control register 17	RSCAN0TMC17	00 <sub>H</sub>	<RSCAN0_base> + 0261 <sub>H</sub>	8
Transmit buffer control register 18	RSCAN0TMC18	00 <sub>H</sub>	<RSCAN0_base> + 0262 <sub>H</sub>	8
Transmit buffer control register 19	RSCAN0TMC19	00 <sub>H</sub>	<RSCAN0_base> + 0263 <sub>H</sub>	8
Transmit buffer control register 20	RSCAN0TMC20	00 <sub>H</sub>	<RSCAN0_base> + 0264 <sub>H</sub>	8

Table 21.10 List of RS-CAN Module Registers (4/25)

Register Name	Symbol	After Reset	Address	Access Size
Transmit buffer control register 21	RSCAN0TMC21	00 <sub>H</sub>	<RSCAN0_base> + 0265 <sub>H</sub>	8
Transmit buffer control register 22	RSCAN0TMC22	00 <sub>H</sub>	<RSCAN0_base> + 0266 <sub>H</sub>	8
Transmit buffer control register 23	RSCAN0TMC23	00 <sub>H</sub>	<RSCAN0_base> + 0267 <sub>H</sub>	8
Transmit buffer control register 24	RSCAN0TMC24	00 <sub>H</sub>	<RSCAN0_base> + 0268 <sub>H</sub>	8
Transmit buffer control register 25	RSCAN0TMC25	00 <sub>H</sub>	<RSCAN0_base> + 0269 <sub>H</sub>	8
Transmit buffer control register 26	RSCAN0TMC26	00 <sub>H</sub>	<RSCAN0_base> + 026A <sub>H</sub>	8
Transmit buffer control register 27	RSCAN0TMC27	00 <sub>H</sub>	<RSCAN0_base> + 026B <sub>H</sub>	8
Transmit buffer control register 28	RSCAN0TMC28	00 <sub>H</sub>	<RSCAN0_base> + 026C <sub>H</sub>	8
Transmit buffer control register 29	RSCAN0TMC29	00 <sub>H</sub>	<RSCAN0_base> + 026D <sub>H</sub>	8
Transmit buffer control register 30	RSCAN0TMC30	00 <sub>H</sub>	<RSCAN0_base> + 026E <sub>H</sub>	8
Transmit buffer control register 31	RSCAN0TMC31	00 <sub>H</sub>	<RSCAN0_base> + 026F <sub>H</sub>	8
Transmit buffer control register 32	RSCAN0TMC32	00 <sub>H</sub>	<RSCAN0_base> + 0270 <sub>H</sub>	8
Transmit buffer control register 33	RSCAN0TMC33	00 <sub>H</sub>	<RSCAN0_base> + 0271 <sub>H</sub>	8
Transmit buffer control register 34	RSCAN0TMC34	00 <sub>H</sub>	<RSCAN0_base> + 0272 <sub>H</sub>	8
Transmit buffer control register 35	RSCAN0TMC35	00 <sub>H</sub>	<RSCAN0_base> + 0273 <sub>H</sub>	8
Transmit buffer control register 36	RSCAN0TMC36	00 <sub>H</sub>	<RSCAN0_base> + 0274 <sub>H</sub>	8
Transmit buffer control register 37	RSCAN0TMC37	00 <sub>H</sub>	<RSCAN0_base> + 0275 <sub>H</sub>	8
Transmit buffer control register 38	RSCAN0TMC38	00 <sub>H</sub>	<RSCAN0_base> + 0276 <sub>H</sub>	8
Transmit buffer control register 39	RSCAN0TMC39	00 <sub>H</sub>	<RSCAN0_base> + 0277 <sub>H</sub>	8
Transmit buffer control register 40	RSCAN0TMC40	00 <sub>H</sub>	<RSCAN0_base> + 0278 <sub>H</sub>	8
Transmit buffer control register 41	RSCAN0TMC41	00 <sub>H</sub>	<RSCAN0_base> + 0279 <sub>H</sub>	8
Transmit buffer control register 42	RSCAN0TMC42	00 <sub>H</sub>	<RSCAN0_base> + 027A <sub>H</sub>	8
Transmit buffer control register 43	RSCAN0TMC43	00 <sub>H</sub>	<RSCAN0_base> + 027B <sub>H</sub>	8
Transmit buffer control register 44	RSCAN0TMC44	00 <sub>H</sub>	<RSCAN0_base> + 027C <sub>H</sub>	8
Transmit buffer control register 45	RSCAN0TMC45	00 <sub>H</sub>	<RSCAN0_base> + 027D <sub>H</sub>	8
Transmit buffer control register 46	RSCAN0TMC46	00 <sub>H</sub>	<RSCAN0_base> + 027E <sub>H</sub>	8
Transmit buffer control register 47	RSCAN0TMC47	00 <sub>H</sub>	<RSCAN0_base> + 027F <sub>H</sub>	8
Transmit buffer control register 48	RSCAN0TMC48	00 <sub>H</sub>	<RSCAN0_base> + 0280 <sub>H</sub>	8
Transmit buffer control register 49	RSCAN0TMC49	00 <sub>H</sub>	<RSCAN0_base> + 0281 <sub>H</sub>	8
Transmit buffer control register 50	RSCAN0TMC50	00 <sub>H</sub>	<RSCAN0_base> + 0282 <sub>H</sub>	8
Transmit buffer control register 51	RSCAN0TMC51	00 <sub>H</sub>	<RSCAN0_base> + 0283 <sub>H</sub>	8
Transmit buffer control register 52	RSCAN0TMC52	00 <sub>H</sub>	<RSCAN0_base> + 0284 <sub>H</sub>	8
Transmit buffer control register 53	RSCAN0TMC53	00 <sub>H</sub>	<RSCAN0_base> + 0285 <sub>H</sub>	8
Transmit buffer control register 54	RSCAN0TMC54	00 <sub>H</sub>	<RSCAN0_base> + 0286 <sub>H</sub>	8
Transmit buffer control register 55	RSCAN0TMC55	00 <sub>H</sub>	<RSCAN0_base> + 0287 <sub>H</sub>	8
Transmit buffer control register 56	RSCAN0TMC56	00 <sub>H</sub>	<RSCAN0_base> + 0288 <sub>H</sub>	8
Transmit buffer control register 57	RSCAN0TMC57	00 <sub>H</sub>	<RSCAN0_base> + 0289 <sub>H</sub>	8
Transmit buffer control register 58	RSCAN0TMC58	00 <sub>H</sub>	<RSCAN0_base> + 028A <sub>H</sub>	8
Transmit buffer control register 59	RSCAN0TMC59	00 <sub>H</sub>	<RSCAN0_base> + 028B <sub>H</sub>	8
Transmit buffer control register 60	RSCAN0TMC60	00 <sub>H</sub>	<RSCAN0_base> + 028C <sub>H</sub>	8
Transmit buffer control register 61	RSCAN0TMC61	00 <sub>H</sub>	<RSCAN0_base> + 028D <sub>H</sub>	8
Transmit buffer control register 62	RSCAN0TMC62	00 <sub>H</sub>	<RSCAN0_base> + 028E <sub>H</sub>	8
Transmit buffer control register 63	RSCAN0TMC63	00 <sub>H</sub>	<RSCAN0_base> + 028F <sub>H</sub>	8
Transmit buffer control register 64	RSCAN0TMC64	00 <sub>H</sub>	<RSCAN0_base> + 0290 <sub>H</sub>	8
Transmit buffer control register 65	RSCAN0TMC65	00 <sub>H</sub>	<RSCAN0_base> + 0291 <sub>H</sub>	8
Transmit buffer control register 66	RSCAN0TMC66	00 <sub>H</sub>	<RSCAN0_base> + 0292 <sub>H</sub>	8
Transmit buffer control register 67	RSCAN0TMC67	00 <sub>H</sub>	<RSCAN0_base> + 0293 <sub>H</sub>	8

Table 21.10 List of RS-CAN Module Registers (5/25)

Register Name	Symbol	After Reset	Address	Access Size
Transmit buffer control register 68	RSCAN0TMC68	00 <sub>H</sub>	<RSCAN0_base> + 0294 <sub>H</sub>	8
Transmit buffer control register 69	RSCAN0TMC69	00 <sub>H</sub>	<RSCAN0_base> + 0295 <sub>H</sub>	8
Transmit buffer control register 70	RSCAN0TMC70	00 <sub>H</sub>	<RSCAN0_base> + 0296 <sub>H</sub>	8
Transmit buffer control register 71	RSCAN0TMC71	00 <sub>H</sub>	<RSCAN0_base> + 0297 <sub>H</sub>	8
Transmit buffer control register 72	RSCAN0TMC72	00 <sub>H</sub>	<RSCAN0_base> + 0298 <sub>H</sub>	8
Transmit buffer control register 73	RSCAN0TMC73	00 <sub>H</sub>	<RSCAN0_base> + 0299 <sub>H</sub>	8
Transmit buffer control register 74	RSCAN0TMC74	00 <sub>H</sub>	<RSCAN0_base> + 029A <sub>H</sub>	8
Transmit buffer control register 75	RSCAN0TMC75	00 <sub>H</sub>	<RSCAN0_base> + 029B <sub>H</sub>	8
Transmit buffer control register 76	RSCAN0TMC76	00 <sub>H</sub>	<RSCAN0_base> + 029C <sub>H</sub>	8
Transmit buffer control register 77	RSCAN0TMC77	00 <sub>H</sub>	<RSCAN0_base> + 029D <sub>H</sub>	8
Transmit buffer control register 78	RSCAN0TMC78	00 <sub>H</sub>	<RSCAN0_base> + 029E <sub>H</sub>	8
Transmit buffer control register 79	RSCAN0TMC79	00 <sub>H</sub>	<RSCAN0_base> + 029F <sub>H</sub>	8
Transmit buffer status register 0	RSCAN0TMSTS0	00 <sub>H</sub>	<RSCAN0_base> + 02D0 <sub>H</sub>	8
Transmit buffer status register 1	RSCAN0TMSTS1	00 <sub>H</sub>	<RSCAN0_base> + 02D1 <sub>H</sub>	8
Transmit buffer status register 2	RSCAN0TMSTS2	00 <sub>H</sub>	<RSCAN0_base> + 02D2 <sub>H</sub>	8
Transmit buffer status register 3	RSCAN0TMSTS3	00 <sub>H</sub>	<RSCAN0_base> + 02D3 <sub>H</sub>	8
Transmit buffer status register 4	RSCAN0TMSTS4	00 <sub>H</sub>	<RSCAN0_base> + 02D4 <sub>H</sub>	8
Transmit buffer status register 5	RSCAN0TMSTS5	00 <sub>H</sub>	<RSCAN0_base> + 02D5 <sub>H</sub>	8
Transmit buffer status register 6	RSCAN0TMSTS6	00 <sub>H</sub>	<RSCAN0_base> + 02D6 <sub>H</sub>	8
Transmit buffer status register 7	RSCAN0TMSTS7	00 <sub>H</sub>	<RSCAN0_base> + 02D7 <sub>H</sub>	8
Transmit buffer status register 8	RSCAN0TMSTS8	00 <sub>H</sub>	<RSCAN0_base> + 02D8 <sub>H</sub>	8
Transmit buffer status register 9	RSCAN0TMSTS9	00 <sub>H</sub>	<RSCAN0_base> + 02D9 <sub>H</sub>	8
Transmit buffer status register 10	RSCAN0TMSTS10	00 <sub>H</sub>	<RSCAN0_base> + 02DA <sub>H</sub>	8
Transmit buffer status register 11	RSCAN0TMSTS11	00 <sub>H</sub>	<RSCAN0_base> + 02DB <sub>H</sub>	8
Transmit buffer status register 12	RSCAN0TMSTS12	00 <sub>H</sub>	<RSCAN0_base> + 02DC <sub>H</sub>	8
Transmit buffer status register 13	RSCAN0TMSTS13	00 <sub>H</sub>	<RSCAN0_base> + 02DD <sub>H</sub>	8
Transmit buffer status register 14	RSCAN0TMSTS14	00 <sub>H</sub>	<RSCAN0_base> + 02DE <sub>H</sub>	8
Transmit buffer status register 15	RSCAN0TMSTS15	00 <sub>H</sub>	<RSCAN0_base> + 02DF <sub>H</sub>	8
Transmit buffer status register 16	RSCAN0TMSTS16	00 <sub>H</sub>	<RSCAN0_base> + 02E0 <sub>H</sub>	8
Transmit buffer status register 17	RSCAN0TMSTS17	00 <sub>H</sub>	<RSCAN0_base> + 02E1 <sub>H</sub>	8
Transmit buffer status register 18	RSCAN0TMSTS18	00 <sub>H</sub>	<RSCAN0_base> + 02E2 <sub>H</sub>	8
Transmit buffer status register 19	RSCAN0TMSTS19	00 <sub>H</sub>	<RSCAN0_base> + 02E3 <sub>H</sub>	8
Transmit buffer status register 20	RSCAN0TMSTS20	00 <sub>H</sub>	<RSCAN0_base> + 02E4 <sub>H</sub>	8
Transmit buffer status register 21	RSCAN0TMSTS21	00 <sub>H</sub>	<RSCAN0_base> + 02E5 <sub>H</sub>	8
Transmit buffer status register 22	RSCAN0TMSTS22	00 <sub>H</sub>	<RSCAN0_base> + 02E6 <sub>H</sub>	8
Transmit buffer status register 23	RSCAN0TMSTS23	00 <sub>H</sub>	<RSCAN0_base> + 02E7 <sub>H</sub>	8
Transmit buffer status register 24	RSCAN0TMSTS24	00 <sub>H</sub>	<RSCAN0_base> + 02E8 <sub>H</sub>	8
Transmit buffer status register 25	RSCAN0TMSTS25	00 <sub>H</sub>	<RSCAN0_base> + 02E9 <sub>H</sub>	8
Transmit buffer status register 26	RSCAN0TMSTS26	00 <sub>H</sub>	<RSCAN0_base> + 02EA <sub>H</sub>	8
Transmit buffer status register 27	RSCAN0TMSTS27	00 <sub>H</sub>	<RSCAN0_base> + 02EB <sub>H</sub>	8
Transmit buffer status register 28	RSCAN0TMSTS28	00 <sub>H</sub>	<RSCAN0_base> + 02EC <sub>H</sub>	8
Transmit buffer status register 29	RSCAN0TMSTS29	00 <sub>H</sub>	<RSCAN0_base> + 02ED <sub>H</sub>	8
Transmit buffer status register 30	RSCAN0TMSTS30	00 <sub>H</sub>	<RSCAN0_base> + 02EE <sub>H</sub>	8
Transmit buffer status register 31	RSCAN0TMSTS31	00 <sub>H</sub>	<RSCAN0_base> + 02EF <sub>H</sub>	8
Transmit buffer status register 32	RSCAN0TMSTS32	00 <sub>H</sub>	<RSCAN0_base> + 02F0 <sub>H</sub>	8
Transmit buffer status register 33	RSCAN0TMSTS33	00 <sub>H</sub>	<RSCAN0_base> + 02F1 <sub>H</sub>	8
Transmit buffer status register 34	RSCAN0TMSTS34	00 <sub>H</sub>	<RSCAN0_base> + 02F2 <sub>H</sub>	8

Table 21.10 List of RS-CAN Module Registers (6/25)

Register Name	Symbol	After Reset	Address	Access Size
Transmit buffer status register 35	RSCAN0TMSTS35	00 <sub>H</sub>	<RSCAN0_base> + 02F3 <sub>H</sub>	8
Transmit buffer status register 36	RSCAN0TMSTS36	00 <sub>H</sub>	<RSCAN0_base> + 02F4 <sub>H</sub>	8
Transmit buffer status register 37	RSCAN0TMSTS37	00 <sub>H</sub>	<RSCAN0_base> + 02F5 <sub>H</sub>	8
Transmit buffer status register 38	RSCAN0TMSTS38	00 <sub>H</sub>	<RSCAN0_base> + 02F6 <sub>H</sub>	8
Transmit buffer status register 39	RSCAN0TMSTS39	00 <sub>H</sub>	<RSCAN0_base> + 02F7 <sub>H</sub>	8
Transmit buffer status register 40	RSCAN0TMSTS40	00 <sub>H</sub>	<RSCAN0_base> + 02F8 <sub>H</sub>	8
Transmit buffer status register 41	RSCAN0TMSTS41	00 <sub>H</sub>	<RSCAN0_base> + 02F9 <sub>H</sub>	8
Transmit buffer status register 42	RSCAN0TMSTS42	00 <sub>H</sub>	<RSCAN0_base> + 02FA <sub>H</sub>	8
Transmit buffer status register 43	RSCAN0TMSTS43	00 <sub>H</sub>	<RSCAN0_base> + 02FB <sub>H</sub>	8
Transmit buffer status register 44	RSCAN0TMSTS44	00 <sub>H</sub>	<RSCAN0_base> + 02FC <sub>H</sub>	8
Transmit buffer status register 45	RSCAN0TMSTS45	00 <sub>H</sub>	<RSCAN0_base> + 02FD <sub>H</sub>	8
Transmit buffer status register 46	RSCAN0TMSTS46	00 <sub>H</sub>	<RSCAN0_base> + 02FE <sub>H</sub>	8
Transmit buffer status register 47	RSCAN0TMSTS47	00 <sub>H</sub>	<RSCAN0_base> + 02FF <sub>H</sub>	8
Transmit buffer status register 48	RSCAN0TMSTS48	00 <sub>H</sub>	<RSCAN0_base> + 0300 <sub>H</sub>	8
Transmit buffer status register 49	RSCAN0TMSTS49	00 <sub>H</sub>	<RSCAN0_base> + 0301 <sub>H</sub>	8
Transmit buffer status register 50	RSCAN0TMSTS50	00 <sub>H</sub>	<RSCAN0_base> + 0302 <sub>H</sub>	8
Transmit buffer status register 51	RSCAN0TMSTS51	00 <sub>H</sub>	<RSCAN0_base> + 0303 <sub>H</sub>	8
Transmit buffer status register 52	RSCAN0TMSTS52	00 <sub>H</sub>	<RSCAN0_base> + 0304 <sub>H</sub>	8
Transmit buffer status register 53	RSCAN0TMSTS53	00 <sub>H</sub>	<RSCAN0_base> + 0305 <sub>H</sub>	8
Transmit buffer status register 54	RSCAN0TMSTS54	00 <sub>H</sub>	<RSCAN0_base> + 0306 <sub>H</sub>	8
Transmit buffer status register 55	RSCAN0TMSTS55	00 <sub>H</sub>	<RSCAN0_base> + 0307 <sub>H</sub>	8
Transmit buffer status register 56	RSCAN0TMSTS56	00 <sub>H</sub>	<RSCAN0_base> + 0308 <sub>H</sub>	8
Transmit buffer status register 57	RSCAN0TMSTS57	00 <sub>H</sub>	<RSCAN0_base> + 0309 <sub>H</sub>	8
Transmit buffer status register 58	RSCAN0TMSTS58	00 <sub>H</sub>	<RSCAN0_base> + 030A <sub>H</sub>	8
Transmit buffer status register 59	RSCAN0TMSTS59	00 <sub>H</sub>	<RSCAN0_base> + 030B <sub>H</sub>	8
Transmit buffer status register 60	RSCAN0TMSTS60	00 <sub>H</sub>	<RSCAN0_base> + 030C <sub>H</sub>	8
Transmit buffer status register 61	RSCAN0TMSTS61	00 <sub>H</sub>	<RSCAN0_base> + 030D <sub>H</sub>	8
Transmit buffer status register 62	RSCAN0TMSTS62	00 <sub>H</sub>	<RSCAN0_base> + 030E <sub>H</sub>	8
Transmit buffer status register 63	RSCAN0TMSTS63	00 <sub>H</sub>	<RSCAN0_base> + 030F <sub>H</sub>	8
Transmit buffer status register 64	RSCAN0TMSTS64	00 <sub>H</sub>	<RSCAN0_base> + 0310 <sub>H</sub>	8
Transmit buffer status register 65	RSCAN0TMSTS65	00 <sub>H</sub>	<RSCAN0_base> + 0311 <sub>H</sub>	8
Transmit buffer status register 66	RSCAN0TMSTS66	00 <sub>H</sub>	<RSCAN0_base> + 0312 <sub>H</sub>	8
Transmit buffer status register 67	RSCAN0TMSTS67	00 <sub>H</sub>	<RSCAN0_base> + 0313 <sub>H</sub>	8
Transmit buffer status register 68	RSCAN0TMSTS68	00 <sub>H</sub>	<RSCAN0_base> + 0314 <sub>H</sub>	8
Transmit buffer status register 69	RSCAN0TMSTS69	00 <sub>H</sub>	<RSCAN0_base> + 0315 <sub>H</sub>	8
Transmit buffer status register 70	RSCAN0TMSTS70	00 <sub>H</sub>	<RSCAN0_base> + 0316 <sub>H</sub>	8
Transmit buffer status register 71	RSCAN0TMSTS71	00 <sub>H</sub>	<RSCAN0_base> + 0317 <sub>H</sub>	8
Transmit buffer status register 72	RSCAN0TMSTS72	00 <sub>H</sub>	<RSCAN0_base> + 0318 <sub>H</sub>	8
Transmit buffer status register 73	RSCAN0TMSTS73	00 <sub>H</sub>	<RSCAN0_base> + 0319 <sub>H</sub>	8
Transmit buffer status register 74	RSCAN0TMSTS74	00 <sub>H</sub>	<RSCAN0_base> + 031A <sub>H</sub>	8
Transmit buffer status register 75	RSCAN0TMSTS75	00 <sub>H</sub>	<RSCAN0_base> + 031B <sub>H</sub>	8
Transmit buffer status register 76	RSCAN0TMSTS76	00 <sub>H</sub>	<RSCAN0_base> + 031C <sub>H</sub>	8
Transmit buffer status register 77	RSCAN0TMSTS77	00 <sub>H</sub>	<RSCAN0_base> + 031D <sub>H</sub>	8
Transmit buffer status register 78	RSCAN0TMSTS78	00 <sub>H</sub>	<RSCAN0_base> + 031E <sub>H</sub>	8
Transmit buffer status register 79	RSCAN0TMSTS79	00 <sub>H</sub>	<RSCAN0_base> + 031F <sub>H</sub>	8
Transmit buffer transmit request status register 0	RSCAN0TMTRSTS0	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0350 <sub>H</sub>	8, 16, 32
Transmit buffer transmit request status register 1	RSCAN0TMTRSTS1	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0354 <sub>H</sub>	8, 16, 32

Table 21.10 List of RS-CAN Module Registers (7/25)

Register Name	Symbol	After Reset	Address	Access Size
Transmit buffer transmit request status register 2	RSCAN0TMRSTS2	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0358 <sub>H</sub>	8, 16, 32
Transmit buffer transmit abort request status register 0	RSCAN0TMTARSTS0	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0360 <sub>H</sub>	8, 16, 32
Transmit buffer transmit abort request status register 1	RSCAN0TMTARSTS1	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0364 <sub>H</sub>	8, 16, 32
Transmit buffer transmit abort request status register 2	RSCAN0TMTARSTS2	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0368 <sub>H</sub>	8, 16, 32
Transmit buffer transmit complete status register 0	RSCAN0TMCSTS0	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0370 <sub>H</sub>	8, 16, 32
Transmit buffer transmit complete status register 1	RSCAN0TMCSTS1	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0374 <sub>H</sub>	8, 16, 32
Transmit buffer transmit complete status register 2	RSCAN0TMCSTS2	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0378 <sub>H</sub>	8, 16, 32
Transmit buffer transmit abort status register 0	RSCAN0TMTASTS0	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0380 <sub>H</sub>	8, 16, 32
Transmit buffer transmit abort status register 1	RSCAN0TMTASTS1	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0384 <sub>H</sub>	8, 16, 32
Transmit buffer transmit abort status register 2	RSCAN0TMTASTS2	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0388 <sub>H</sub>	8, 16, 32
Transmit buffer interrupt enable configuration register 0	RSCAN0TMIEC0	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0390 <sub>H</sub>	8, 16, 32
Transmit buffer interrupt enable configuration register 1	RSCAN0TMIEC1	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0394 <sub>H</sub>	8, 16, 32
Transmit buffer interrupt enable configuration register 2	RSCAN0TMIEC2	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0398 <sub>H</sub>	8, 16, 32
Transmit queue configuration and control register 0	RSCAN0TXQCC0	0000 0000 <sub>H</sub>	<RSCAN0_base> + 03A0 <sub>H</sub>	8, 16, 32
Transmit queue configuration and control register 1	RSCAN0TXQCC1	0000 0000 <sub>H</sub>	<RSCAN0_base> + 03A4 <sub>H</sub>	8, 16, 32
Transmit queue configuration and control register 2	RSCAN0TXQCC2	0000 0000 <sub>H</sub>	<RSCAN0_base> + 03A8 <sub>H</sub>	8, 16, 32
Transmit queue configuration and control register 3	RSCAN0TXQCC3	0000 0000 <sub>H</sub>	<RSCAN0_base> + 03AC <sub>H</sub>	8, 16, 32
Transmit queue configuration and control register 4	RSCAN0TXQCC4	0000 0000 <sub>H</sub>	<RSCAN0_base> + 03B0 <sub>H</sub>	8, 16, 32
Transmit queue status register 0	RSCAN0TXQSTS0	0000 0001 <sub>H</sub>	<RSCAN0_base> + 03C0 <sub>H</sub>	8, 16, 32
Transmit queue status register 1	RSCAN0TXQSTS1	0000 0001 <sub>H</sub>	<RSCAN0_base> + 03C4 <sub>H</sub>	8, 16, 32
Transmit queue status register 2	RSCAN0TXQSTS2	0000 0001 <sub>H</sub>	<RSCAN0_base> + 03C8 <sub>H</sub>	8, 16, 32
Transmit queue status register 3	RSCAN0TXQSTS3	0000 0001 <sub>H</sub>	<RSCAN0_base> + 03CC <sub>H</sub>	8, 16, 32
Transmit queue status register 4	RSCAN0TXQSTS4	0000 0001 <sub>H</sub>	<RSCAN0_base> + 03D0 <sub>H</sub>	8, 16, 32
Transmit queue pointer control register 0	RSCAN0TXQPCTR0	—	<RSCAN0_base> + 03E0 <sub>H</sub>	8, 16, 32
Transmit queue pointer control register 1	RSCAN0TXQPCTR1	—	<RSCAN0_base> + 03E4 <sub>H</sub>	8, 16, 32
Transmit queue pointer control register 2	RSCAN0TXQPCTR2	—	<RSCAN0_base> + 03E8 <sub>H</sub>	8, 16, 32
Transmit queue pointer control register 3	RSCAN0TXQPCTR3	—	<RSCAN0_base> + 03EC <sub>H</sub>	8, 16, 32
Transmit queue pointer control register 4	RSCAN0TXQPCTR4	—	<RSCAN0_base> + 03F0 <sub>H</sub>	8, 16, 32
Transmit history configuration and control register 0	RSCAN0THLCC0	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0400 <sub>H</sub>	8, 16, 32
Transmit history configuration and control register 1	RSCAN0THLCC1	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0404 <sub>H</sub>	8, 16, 32
Transmit history configuration and control register 2	RSCAN0THLCC2	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0408 <sub>H</sub>	8, 16, 32
Transmit history configuration and control register 3	RSCAN0THLCC3	0000 0000 <sub>H</sub>	<RSCAN0_base> + 040C <sub>H</sub>	8, 16, 32
Transmit history configuration and control register 4	RSCAN0THLCC4	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0410 <sub>H</sub>	8, 16, 32
Transmit history status register 0	RSCAN0THLSTS0	0000 0001 <sub>H</sub>	<RSCAN0_base> + 0420 <sub>H</sub>	8, 16, 32
Transmit history status register 1	RSCAN0THLSTS1	0000 0001 <sub>H</sub>	<RSCAN0_base> + 0424 <sub>H</sub>	8, 16, 32
Transmit history status register 2	RSCAN0THLSTS2	0000 0001 <sub>H</sub>	<RSCAN0_base> + 0428 <sub>H</sub>	8, 16, 32
Transmit history status register 3	RSCAN0THLSTS3	0000 0001 <sub>H</sub>	<RSCAN0_base> + 042C <sub>H</sub>	8, 16, 32
Transmit history status register 4	RSCAN0THLSTS4	0000 0001 <sub>H</sub>	<RSCAN0_base> + 0430 <sub>H</sub>	8, 16, 32
Transmit history pointer control register 0	RSCAN0THLPCTR0	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0440 <sub>H</sub>	8, 16, 32
Transmit history pointer control register 1	RSCAN0THLPCTR1	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0444 <sub>H</sub>	8, 16, 32
Transmit history pointer control register 2	RSCAN0THLPCTR2	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0448 <sub>H</sub>	8, 16, 32
Transmit history pointer control register 3	RSCAN0THLPCTR3	0000 0000 <sub>H</sub>	<RSCAN0_base> + 044C <sub>H</sub>	8, 16, 32
Transmit history pointer control register 4	RSCAN0THLPCTR4	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0450 <sub>H</sub>	8, 16, 32
Global TX interrupt status register 0	RSCAN0GTINTSTS0	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0460 <sub>H</sub>	8, 16, 32
Global TX interrupt status register 1	RSCAN0GTINTSTS1	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0464 <sub>H</sub>	8, 16, 32
Global test configuration register	RSCAN0GTSTCFG	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0468 <sub>H</sub>	8, 16, 32
Global test control register	RSCAN0GTSTCTR	0000 0000 <sub>H</sub>	<RSCAN0_base> + 046C <sub>H</sub>	8, 16, 32



Table 21.10 List of RS-CAN Module Registers (8/25)

Register Name	Symbol	After Reset	Address	Access Size
Global lock key register	RSCAN0GLOCKK	—	<RSCAN0_base> + 047C <sub>H</sub>	16, 32
Receive rule ID register 0	RSCAN0GAFLID0	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0500 <sub>H</sub>	8, 16, 32
Receive rule mask register 0	RSCAN0GAFLM0	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0504 <sub>H</sub>	8, 16, 32
Receive rule pointer 0 register 0	RSCAN0GAFLP00	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0508 <sub>H</sub>	8, 16, 32
Receive rule pointer 1 register 0	RSCAN0GAFLP10	0000 0000 <sub>H</sub>	<RSCAN0_base> + 050C <sub>H</sub>	8, 16, 32
Receive rule ID register 1	RSCAN0GAFLID1	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0510 <sub>H</sub>	8, 16, 32
Receive rule mask register 1	RSCAN0GAFLM1	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0514 <sub>H</sub>	8, 16, 32
Receive rule pointer 0 register 1	RSCAN0GAFLP01	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0518 <sub>H</sub>	8, 16, 32
Receive rule pointer 1 register 1	RSCAN0GAFLP11	0000 0000 <sub>H</sub>	<RSCAN0_base> + 051C <sub>H</sub>	8, 16, 32
Receive rule ID register 2	RSCAN0GAFLID2	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0520 <sub>H</sub>	8, 16, 32
Receive rule mask register 2	RSCAN0GAFLM2	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0524 <sub>H</sub>	8, 16, 32
Receive rule pointer 0 register 2	RSCAN0GAFLP02	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0528 <sub>H</sub>	8, 16, 32
Receive rule pointer 1 register 2	RSCAN0GAFLP12	0000 0000 <sub>H</sub>	<RSCAN0_base> + 052C <sub>H</sub>	8, 16, 32
Receive rule ID register 3	RSCAN0GAFLID3	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0530 <sub>H</sub>	8, 16, 32
Receive rule mask register 3	RSCAN0GAFLM3	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0534 <sub>H</sub>	8, 16, 32
Receive rule pointer 0 register 3	RSCAN0GAFLP03	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0538 <sub>H</sub>	8, 16, 32
Receive rule pointer 1 register 3	RSCAN0GAFLP13	0000 0000 <sub>H</sub>	<RSCAN0_base> + 053C <sub>H</sub>	8, 16, 32
Receive rule ID register 4	RSCAN0GAFLID4	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0540 <sub>H</sub>	8, 16, 32
Receive rule mask register 4	RSCAN0GAFLM4	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0544 <sub>H</sub>	8, 16, 32
Receive rule pointer 0 register 4	RSCAN0GAFLP04	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0548 <sub>H</sub>	8, 16, 32
Receive rule pointer 1 register 4	RSCAN0GAFLP14	0000 0000 <sub>H</sub>	<RSCAN0_base> + 054C <sub>H</sub>	8, 16, 32
Receive rule ID register 5	RSCAN0GAFLID5	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0550 <sub>H</sub>	8, 16, 32
Receive rule mask register 5	RSCAN0GAFLM5	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0554 <sub>H</sub>	8, 16, 32
Receive rule pointer 0 register 5	RSCAN0GAFLP05	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0558 <sub>H</sub>	8, 16, 32
Receive rule pointer 1 register 5	RSCAN0GAFLP15	0000 0000 <sub>H</sub>	<RSCAN0_base> + 055C <sub>H</sub>	8, 16, 32
Receive rule ID register 6	RSCAN0GAFLID6	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0560 <sub>H</sub>	8, 16, 32
Receive rule mask register 6	RSCAN0GAFLM6	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0564 <sub>H</sub>	8, 16, 32
Receive rule pointer 0 register 6	RSCAN0GAFLP06	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0568 <sub>H</sub>	8, 16, 32
Receive rule pointer 1 register 6	RSCAN0GAFLP16	0000 0000 <sub>H</sub>	<RSCAN0_base> + 056C <sub>H</sub>	8, 16, 32
Receive rule ID register 7	RSCAN0GAFLID7	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0570 <sub>H</sub>	8, 16, 32
Receive rule mask register 7	RSCAN0GAFLM7	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0574 <sub>H</sub>	8, 16, 32
Receive rule pointer 0 register 7	RSCAN0GAFLP07	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0578 <sub>H</sub>	8, 16, 32
Receive rule pointer 1 register 7	RSCAN0GAFLP17	0000 0000 <sub>H</sub>	<RSCAN0_base> + 057C <sub>H</sub>	8, 16, 32
Receive rule ID register 8	RSCAN0GAFLID8	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0580 <sub>H</sub>	8, 16, 32
Receive rule mask register 8	RSCAN0GAFLM8	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0584 <sub>H</sub>	8, 16, 32
Receive rule pointer 0 register 8	RSCAN0GAFLP08	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0588 <sub>H</sub>	8, 16, 32
Receive rule pointer 1 register 8	RSCAN0GAFLP18	0000 0000 <sub>H</sub>	<RSCAN0_base> + 058C <sub>H</sub>	8, 16, 32
Receive rule ID register 9	RSCAN0GAFLID9	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0590 <sub>H</sub>	8, 16, 32
Receive rule mask register 9	RSCAN0GAFLM9	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0594 <sub>H</sub>	8, 16, 32
Receive rule pointer 0 register 9	RSCAN0GAFLP09	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0598 <sub>H</sub>	8, 16, 32
Receive rule pointer 1 register 9	RSCAN0GAFLP19	0000 0000 <sub>H</sub>	<RSCAN0_base> + 059C <sub>H</sub>	8, 16, 32
Receive rule ID register 10	RSCAN0GAFLID10	0000 0000 <sub>H</sub>	<RSCAN0_base> + 05A0 <sub>H</sub>	8, 16, 32
Receive rule mask register 10	RSCAN0GAFLM10	0000 0000 <sub>H</sub>	<RSCAN0_base> + 05A4 <sub>H</sub>	8, 16, 32
Receive rule pointer 0 register 10	RSCAN0GAFLP010	0000 0000 <sub>H</sub>	<RSCAN0_base> + 05A8 <sub>H</sub>	8, 16, 32
Receive rule pointer 1 register 10	RSCAN0GAFLP110	0000 0000 <sub>H</sub>	<RSCAN0_base> + 05AC <sub>H</sub>	8, 16, 32
Receive rule ID register 11	RSCAN0GAFLID11	0000 0000 <sub>H</sub>	<RSCAN0_base> + 05B0 <sub>H</sub>	8, 16, 32
Receive rule mask register 11	RSCAN0GAFLM11	0000 0000 <sub>H</sub>	<RSCAN0_base> + 05B4 <sub>H</sub>	8, 16, 32

Table 21.10 List of RS-CAN Module Registers (9/25)

Register Name	Symbol	After Reset	Address	Access Size
Receive rule pointer 0 register 11	RSCAN0GAFLP011	0000 0000 <sub>H</sub>	<RSCAN0_base> + 05B8 <sub>H</sub>	8, 16, 32
Receive rule pointer 1 register 11	RSCAN0GAFLP111	0000 0000 <sub>H</sub>	<RSCAN0_base> + 05BC <sub>H</sub>	8, 16, 32
Receive rule ID register 12	RSCAN0GAFLID12	0000 0000 <sub>H</sub>	<RSCAN0_base> + 05C0 <sub>H</sub>	8, 16, 32
Receive rule mask register 12	RSCAN0GAFLM12	0000 0000 <sub>H</sub>	<RSCAN0_base> + 05C4 <sub>H</sub>	8, 16, 32
Receive rule pointer 0 register 12	RSCAN0GAFLP012	0000 0000 <sub>H</sub>	<RSCAN0_base> + 05C8 <sub>H</sub>	8, 16, 32
Receive rule pointer 1 register 12	RSCAN0GAFLP112	0000 0000 <sub>H</sub>	<RSCAN0_base> + 05CC <sub>H</sub>	8, 16, 32
Receive rule ID register 13	RSCAN0GAFLID13	0000 0000 <sub>H</sub>	<RSCAN0_base> + 05D0 <sub>H</sub>	8, 16, 32
Receive rule mask register 13	RSCAN0GAFLM13	0000 0000 <sub>H</sub>	<RSCAN0_base> + 05D4 <sub>H</sub>	8, 16, 32
Receive rule pointer 0 register 13	RSCAN0GAFLP013	0000 0000 <sub>H</sub>	<RSCAN0_base> + 05D8 <sub>H</sub>	8, 16, 32
Receive rule pointer 1 register 13	RSCAN0GAFLP113	0000 0000 <sub>H</sub>	<RSCAN0_base> + 05DC <sub>H</sub>	8, 16, 32
Receive rule ID register 14	RSCAN0GAFLID14	0000 0000 <sub>H</sub>	<RSCAN0_base> + 05E0 <sub>H</sub>	8, 16, 32
Receive rule mask register 14	RSCAN0GAFLM14	0000 0000 <sub>H</sub>	<RSCAN0_base> + 05E4 <sub>H</sub>	8, 16, 32
Receive rule pointer 0 register 14	RSCAN0GAFLP014	0000 0000 <sub>H</sub>	<RSCAN0_base> + 05E8 <sub>H</sub>	8, 16, 32
Receive rule pointer 1 register 14	RSCAN0GAFLP114	0000 0000 <sub>H</sub>	<RSCAN0_base> + 05EC <sub>H</sub>	8, 16, 32
Receive rule ID register 15	RSCAN0GAFLID15	0000 0000 <sub>H</sub>	<RSCAN0_base> + 05F0 <sub>H</sub>	8, 16, 32
Receive rule mask register 15	RSCAN0GAFLM15	0000 0000 <sub>H</sub>	<RSCAN0_base> + 05F4 <sub>H</sub>	8, 16, 32
Receive rule pointer 0 register 15	RSCAN0GAFLP015	0000 0000 <sub>H</sub>	<RSCAN0_base> + 05F8 <sub>H</sub>	8, 16, 32
Receive rule pointer 1 register 15	RSCAN0GAFLP115	0000 0000 <sub>H</sub>	<RSCAN0_base> + 05FC <sub>H</sub>	8, 16, 32
Receive buffer ID register 0	RSCAN0RMID0	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0600 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 0	RSCAN0RMPTR0	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0604 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 0	RSCAN0RMDf00	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0608 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 0	RSCAN0RMDf10	0000 0000 <sub>H</sub>	<RSCAN0_base> + 060C <sub>H</sub>	8, 16, 32
Receive buffer ID register 1	RSCAN0RMID1	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0610 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 1	RSCAN0RMPTR1	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0614 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 1	RSCAN0RMDf01	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0618 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 1	RSCAN0RMDf11	0000 0000 <sub>H</sub>	<RSCAN0_base> + 061C <sub>H</sub>	8, 16, 32
Receive buffer ID register 2	RSCAN0RMID2	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0620 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 2	RSCAN0RMPTR2	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0624 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 2	RSCAN0RMDf02	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0628 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 2	RSCAN0RMDf12	0000 0000 <sub>H</sub>	<RSCAN0_base> + 062C <sub>H</sub>	8, 16, 32
Receive buffer ID register 3	RSCAN0RMID3	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0630 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 3	RSCAN0RMPTR3	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0634 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 3	RSCAN0RMDf03	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0638 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 3	RSCAN0RMDf13	0000 0000 <sub>H</sub>	<RSCAN0_base> + 063C <sub>H</sub>	8, 16, 32
Receive buffer ID register 4	RSCAN0RMID4	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0640 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 4	RSCAN0RMPTR4	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0644 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 4	RSCAN0RMDf04	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0648 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 4	RSCAN0RMDf14	0000 0000 <sub>H</sub>	<RSCAN0_base> + 064C <sub>H</sub>	8, 16, 32
Receive buffer ID register 5	RSCAN0RMID5	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0650 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 5	RSCAN0RMPTR5	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0654 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 5	RSCAN0RMDf05	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0658 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 5	RSCAN0RMDf15	0000 0000 <sub>H</sub>	<RSCAN0_base> + 065C <sub>H</sub>	8, 16, 32
Receive buffer ID register 6	RSCAN0RMID6	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0660 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 6	RSCAN0RMPTR6	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0664 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 6	RSCAN0RMDf06	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0668 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 6	RSCAN0RMDf16	0000 0000 <sub>H</sub>	<RSCAN0_base> + 066C <sub>H</sub>	8, 16, 32
Receive buffer ID register 7	RSCAN0RMID7	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0670 <sub>H</sub>	8, 16, 32

Table 21.10 List of RS-CAN Module Registers (10/25)

Register Name	Symbol	After Reset	Address	Access Size
Receive buffer pointer register 7	RSCAN0RMPTR7	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0674 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 7	RSCAN0RMDf07	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0678 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 7	RSCAN0RMDf17	0000 0000 <sub>H</sub>	<RSCAN0_base> + 067C <sub>H</sub>	8, 16, 32
Receive buffer ID register 8	RSCAN0RMID8	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0680 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 8	RSCAN0RMPTR8	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0684 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 8	RSCAN0RMDf08	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0688 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 8	RSCAN0RMDf18	0000 0000 <sub>H</sub>	<RSCAN0_base> + 068C <sub>H</sub>	8, 16, 32
Receive buffer ID register 9	RSCAN0RMID9	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0690 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 9	RSCAN0RMPTR9	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0694 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 9	RSCAN0RMDf09	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0698 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 9	RSCAN0RMDf19	0000 0000 <sub>H</sub>	<RSCAN0_base> + 069C <sub>H</sub>	8, 16, 32
Receive buffer ID register 10	RSCAN0RMID10	0000 0000 <sub>H</sub>	<RSCAN0_base> + 06A0 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 10	RSCAN0RMPTR10	0000 0000 <sub>H</sub>	<RSCAN0_base> + 06A4 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 10	RSCAN0RMDf10	0000 0000 <sub>H</sub>	<RSCAN0_base> + 06A8 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 10	RSCAN0RMDf110	0000 0000 <sub>H</sub>	<RSCAN0_base> + 06AC <sub>H</sub>	8, 16, 32
Receive buffer ID register 11	RSCAN0RMID11	0000 0000 <sub>H</sub>	<RSCAN0_base> + 06B0 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 11	RSCAN0RMPTR11	0000 0000 <sub>H</sub>	<RSCAN0_base> + 06B4 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 11	RSCAN0RMDf011	0000 0000 <sub>H</sub>	<RSCAN0_base> + 06B8 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 11	RSCAN0RMDf111	0000 0000 <sub>H</sub>	<RSCAN0_base> + 06BC <sub>H</sub>	8, 16, 32
Receive buffer ID register 12	RSCAN0RMID12	0000 0000 <sub>H</sub>	<RSCAN0_base> + 06C0 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 12	RSCAN0RMPTR12	0000 0000 <sub>H</sub>	<RSCAN0_base> + 06C4 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 12	RSCAN0RMDf012	0000 0000 <sub>H</sub>	<RSCAN0_base> + 06C8 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 12	RSCAN0RMDf112	0000 0000 <sub>H</sub>	<RSCAN0_base> + 06CC <sub>H</sub>	8, 16, 32
Receive buffer ID register 13	RSCAN0RMID13	0000 0000 <sub>H</sub>	<RSCAN0_base> + 06D0 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 13	RSCAN0RMPTR13	0000 0000 <sub>H</sub>	<RSCAN0_base> + 06D4 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 13	RSCAN0RMDf013	0000 0000 <sub>H</sub>	<RSCAN0_base> + 06D8 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 13	RSCAN0RMDf113	0000 0000 <sub>H</sub>	<RSCAN0_base> + 06DC <sub>H</sub>	8, 16, 32
Receive buffer ID register 14	RSCAN0RMID14	0000 0000 <sub>H</sub>	<RSCAN0_base> + 06E0 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 14	RSCAN0RMPTR14	0000 0000 <sub>H</sub>	<RSCAN0_base> + 06E4 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 14	RSCAN0RMDf014	0000 0000 <sub>H</sub>	<RSCAN0_base> + 06E8 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 14	RSCAN0RMDf114	0000 0000 <sub>H</sub>	<RSCAN0_base> + 06EC <sub>H</sub>	8, 16, 32
Receive buffer ID register 15	RSCAN0RMID15	0000 0000 <sub>H</sub>	<RSCAN0_base> + 06F0 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 15	RSCAN0RMPTR15	0000 0000 <sub>H</sub>	<RSCAN0_base> + 06F4 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 15	RSCAN0RMDf015	0000 0000 <sub>H</sub>	<RSCAN0_base> + 06F8 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 15	RSCAN0RMDf115	0000 0000 <sub>H</sub>	<RSCAN0_base> + 06FC <sub>H</sub>	8, 16, 32
Receive buffer ID register 16	RSCAN0RMID16	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0700 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 16	RSCAN0RMPTR16	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0704 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 16	RSCAN0RMDf016	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0708 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 16	RSCAN0RMDf116	0000 0000 <sub>H</sub>	<RSCAN0_base> + 070C <sub>H</sub>	8, 16, 32
Receive buffer ID register 17	RSCAN0RMID17	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0710 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 17	RSCAN0RMPTR17	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0714 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 17	RSCAN0RMDf017	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0718 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 17	RSCAN0RMDf117	0000 0000 <sub>H</sub>	<RSCAN0_base> + 071C <sub>H</sub>	8, 16, 32
Receive buffer ID register 18	RSCAN0RMID18	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0720 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 18	RSCAN0RMPTR18	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0724 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 18	RSCAN0RMDf018	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0728 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 18	RSCAN0RMDf118	0000 0000 <sub>H</sub>	<RSCAN0_base> + 072C <sub>H</sub>	8, 16, 32

Table 21.10 List of RS-CAN Module Registers (11/25)

Register Name	Symbol	After Reset	Address	Access Size
Receive buffer ID register 19	RSCAN0RMID19	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0730 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 19	RSCAN0RMPTR19	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0734 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 19	RSCAN0RMDf019	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0738 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 19	RSCAN0RMDf119	0000 0000 <sub>H</sub>	<RSCAN0_base> + 073C <sub>H</sub>	8, 16, 32
Receive buffer ID register 20	RSCAN0RMID20	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0740 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 20	RSCAN0RMPTR20	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0744 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 20	RSCAN0RMDf020	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0748 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 20	RSCAN0RMDf120	0000 0000 <sub>H</sub>	<RSCAN0_base> + 074C <sub>H</sub>	8, 16, 32
Receive buffer ID register 21	RSCAN0RMID21	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0750 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 21	RSCAN0RMPTR21	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0754 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 21	RSCAN0RMDf021	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0758 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 21	RSCAN0RMDf121	0000 0000 <sub>H</sub>	<RSCAN0_base> + 075C <sub>H</sub>	8, 16, 32
Receive buffer ID register 22	RSCAN0RMID22	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0760 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 22	RSCAN0RMPTR22	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0764 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 22	RSCAN0RMDf022	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0768 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 22	RSCAN0RMDf122	0000 0000 <sub>H</sub>	<RSCAN0_base> + 076C <sub>H</sub>	8, 16, 32
Receive buffer ID register 23	RSCAN0RMID23	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0770 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 23	RSCAN0RMPTR23	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0774 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 23	RSCAN0RMDf023	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0778 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 23	RSCAN0RMDf123	0000 0000 <sub>H</sub>	<RSCAN0_base> + 077C <sub>H</sub>	8, 16, 32
Receive buffer ID register 24	RSCAN0RMID24	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0780 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 24	RSCAN0RMPTR24	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0784 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 24	RSCAN0RMDf024	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0788 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 24	RSCAN0RMDf124	0000 0000 <sub>H</sub>	<RSCAN0_base> + 078C <sub>H</sub>	8, 16, 32
Receive buffer ID register 25	RSCAN0RMID25	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0790 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 25	RSCAN0RMPTR25	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0794 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 25	RSCAN0RMDf025	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0798 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 25	RSCAN0RMDf125	0000 0000 <sub>H</sub>	<RSCAN0_base> + 079C <sub>H</sub>	8, 16, 32
Receive buffer ID register 26	RSCAN0RMID26	0000 0000 <sub>H</sub>	<RSCAN0_base> + 07A0 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 26	RSCAN0RMPTR26	0000 0000 <sub>H</sub>	<RSCAN0_base> + 07A4 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 26	RSCAN0RMDf026	0000 0000 <sub>H</sub>	<RSCAN0_base> + 07A8 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 26	RSCAN0RMDf126	0000 0000 <sub>H</sub>	<RSCAN0_base> + 07AC <sub>H</sub>	8, 16, 32
Receive buffer ID register 27	RSCAN0RMID27	0000 0000 <sub>H</sub>	<RSCAN0_base> + 07B0 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 27	RSCAN0RMPTR27	0000 0000 <sub>H</sub>	<RSCAN0_base> + 07B4 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 27	RSCAN0RMDf027	0000 0000 <sub>H</sub>	<RSCAN0_base> + 07B8 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 27	RSCAN0RMDf127	0000 0000 <sub>H</sub>	<RSCAN0_base> + 07BC <sub>H</sub>	8, 16, 32
Receive buffer ID register 28	RSCAN0RMID28	0000 0000 <sub>H</sub>	<RSCAN0_base> + 07C0 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 28	RSCAN0RMPTR28	0000 0000 <sub>H</sub>	<RSCAN0_base> + 07C4 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 28	RSCAN0RMDf028	0000 0000 <sub>H</sub>	<RSCAN0_base> + 07C8 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 28	RSCAN0RMDf128	0000 0000 <sub>H</sub>	<RSCAN0_base> + 07CC <sub>H</sub>	8, 16, 32
Receive buffer ID register 29	RSCAN0RMID29	0000 0000 <sub>H</sub>	<RSCAN0_base> + 07D0 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 29	RSCAN0RMPTR29	0000 0000 <sub>H</sub>	<RSCAN0_base> + 07D4 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 29	RSCAN0RMDf029	0000 0000 <sub>H</sub>	<RSCAN0_base> + 07D8 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 29	RSCAN0RMDf129	0000 0000 <sub>H</sub>	<RSCAN0_base> + 07DC <sub>H</sub>	8, 16, 32
Receive buffer ID register 30	RSCAN0RMID30	0000 0000 <sub>H</sub>	<RSCAN0_base> + 07E0 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 30	RSCAN0RMPTR30	0000 0000 <sub>H</sub>	<RSCAN0_base> + 07E4 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 30	RSCAN0RMDf030	0000 0000 <sub>H</sub>	<RSCAN0_base> + 07E8 <sub>H</sub>	8, 16, 32

Table 21.10 List of RS-CAN Module Registers (12/25)

Register Name	Symbol	After Reset	Address	Access Size
Receive buffer data field 1 register 30	RSCAN0RMDf130	0000 0000 <sub>H</sub>	<RSCAN0_base> + 07EC <sub>H</sub>	8, 16, 32
Receive buffer ID register 31	RSCAN0RMID31	0000 0000 <sub>H</sub>	<RSCAN0_base> + 07F0 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 31	RSCAN0RMPTR31	0000 0000 <sub>H</sub>	<RSCAN0_base> + 07F4 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 31	RSCAN0RMDf031	0000 0000 <sub>H</sub>	<RSCAN0_base> + 07F8 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 31	RSCAN0RMDf131	0000 0000 <sub>H</sub>	<RSCAN0_base> + 07FC <sub>H</sub>	8, 16, 32
Receive buffer ID register 32	RSCAN0RMID32	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0800 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 32	RSCAN0RMPTR32	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0804 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 32	RSCAN0RMDf032	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0808 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 32	RSCAN0RMDf132	0000 0000 <sub>H</sub>	<RSCAN0_base> + 080C <sub>H</sub>	8, 16, 32
Receive buffer ID register 33	RSCAN0RMID33	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0810 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 33	RSCAN0RMPTR33	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0814 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 33	RSCAN0RMDf033	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0818 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 33	RSCAN0RMDf133	0000 0000 <sub>H</sub>	<RSCAN0_base> + 081C <sub>H</sub>	8, 16, 32
Receive buffer ID register 34	RSCAN0RMID34	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0820 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 34	RSCAN0RMPTR34	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0824 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 34	RSCAN0RMDf034	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0828 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 34	RSCAN0RMDf134	0000 0000 <sub>H</sub>	<RSCAN0_base> + 082C <sub>H</sub>	8, 16, 32
Receive buffer ID register 35	RSCAN0RMID35	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0830 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 35	RSCAN0RMPTR35	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0834 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 35	RSCAN0RMDf035	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0838 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 35	RSCAN0RMDf135	0000 0000 <sub>H</sub>	<RSCAN0_base> + 083C <sub>H</sub>	8, 16, 32
Receive buffer ID register 36	RSCAN0RMID36	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0840 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 36	RSCAN0RMPTR36	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0844 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 36	RSCAN0RMDf036	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0848 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 36	RSCAN0RMDf136	0000 0000 <sub>H</sub>	<RSCAN0_base> + 084C <sub>H</sub>	8, 16, 32
Receive buffer ID register 37	RSCAN0RMID37	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0850 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 37	RSCAN0RMPTR37	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0854 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 37	RSCAN0RMDf037	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0858 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 37	RSCAN0RMDf137	0000 0000 <sub>H</sub>	<RSCAN0_base> + 085C <sub>H</sub>	8, 16, 32
Receive buffer ID register 38	RSCAN0RMID38	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0860 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 38	RSCAN0RMPTR38	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0864 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 38	RSCAN0RMDf038	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0868 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 38	RSCAN0RMDf138	0000 0000 <sub>H</sub>	<RSCAN0_base> + 086C <sub>H</sub>	8, 16, 32
Receive buffer ID register 39	RSCAN0RMID39	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0870 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 39	RSCAN0RMPTR39	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0874 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 39	RSCAN0RMDf039	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0878 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 39	RSCAN0RMDf139	0000 0000 <sub>H</sub>	<RSCAN0_base> + 087C <sub>H</sub>	8, 16, 32
Receive buffer ID register 40	RSCAN0RMID40	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0880 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 40	RSCAN0RMPTR40	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0884 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 40	RSCAN0RMDf040	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0888 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 40	RSCAN0RMDf140	0000 0000 <sub>H</sub>	<RSCAN0_base> + 088C <sub>H</sub>	8, 16, 32
Receive buffer ID register 41	RSCAN0RMID41	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0890 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 41	RSCAN0RMPTR41	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0894 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 41	RSCAN0RMDf041	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0898 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 41	RSCAN0RMDf141	0000 0000 <sub>H</sub>	<RSCAN0_base> + 089C <sub>H</sub>	8, 16, 32
Receive buffer ID register 42	RSCAN0RMID42	0000 0000 <sub>H</sub>	<RSCAN0_base> + 08A0 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 42	RSCAN0RMPTR42	0000 0000 <sub>H</sub>	<RSCAN0_base> + 08A4 <sub>H</sub>	8, 16, 32

Table 21.10 List of RS-CAN Module Registers (13/25)

Register Name	Symbol	After Reset	Address	Access Size
Receive buffer data field 0 register 42	RSCAN0RMDf042	0000 0000 <sub>H</sub>	<RSCAN0_base> + 08A8 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 42	RSCAN0RMDf142	0000 0000 <sub>H</sub>	<RSCAN0_base> + 08AC <sub>H</sub>	8, 16, 32
Receive buffer ID register 43	RSCAN0RMID43	0000 0000 <sub>H</sub>	<RSCAN0_base> + 08B0 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 43	RSCAN0RMPTR43	0000 0000 <sub>H</sub>	<RSCAN0_base> + 08B4 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 43	RSCAN0RMDf043	0000 0000 <sub>H</sub>	<RSCAN0_base> + 08B8 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 43	RSCAN0RMDf143	0000 0000 <sub>H</sub>	<RSCAN0_base> + 08BC <sub>H</sub>	8, 16, 32
Receive buffer ID register 44	RSCAN0RMID44	0000 0000 <sub>H</sub>	<RSCAN0_base> + 08C0 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 44	RSCAN0RMPTR44	0000 0000 <sub>H</sub>	<RSCAN0_base> + 08C4 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 44	RSCAN0RMDf044	0000 0000 <sub>H</sub>	<RSCAN0_base> + 08C8 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 44	RSCAN0RMDf144	0000 0000 <sub>H</sub>	<RSCAN0_base> + 08CC <sub>H</sub>	8, 16, 32
Receive buffer ID register 45	RSCAN0RMID45	0000 0000 <sub>H</sub>	<RSCAN0_base> + 08D0 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 45	RSCAN0RMPTR45	0000 0000 <sub>H</sub>	<RSCAN0_base> + 08D4 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 45	RSCAN0RMDf045	0000 0000 <sub>H</sub>	<RSCAN0_base> + 08D8 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 45	RSCAN0RMDf145	0000 0000 <sub>H</sub>	<RSCAN0_base> + 08DC <sub>H</sub>	8, 16, 32
Receive buffer ID register 46	RSCAN0RMID46	0000 0000 <sub>H</sub>	<RSCAN0_base> + 08E0 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 46	RSCAN0RMPTR46	0000 0000 <sub>H</sub>	<RSCAN0_base> + 08E4 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 46	RSCAN0RMDf046	0000 0000 <sub>H</sub>	<RSCAN0_base> + 08E8 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 46	RSCAN0RMDf146	0000 0000 <sub>H</sub>	<RSCAN0_base> + 08EC <sub>H</sub>	8, 16, 32
Receive buffer ID register 47	RSCAN0RMID47	0000 0000 <sub>H</sub>	<RSCAN0_base> + 08F0 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 47	RSCAN0RMPTR47	0000 0000 <sub>H</sub>	<RSCAN0_base> + 08F4 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 47	RSCAN0RMDf047	0000 0000 <sub>H</sub>	<RSCAN0_base> + 08F8 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 47	RSCAN0RMDf147	0000 0000 <sub>H</sub>	<RSCAN0_base> + 08FC <sub>H</sub>	8, 16, 32
Receive buffer ID register 48	RSCAN0RMID48	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0900 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 48	RSCAN0RMPTR48	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0904 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 48	RSCAN0RMDf048	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0908 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 48	RSCAN0RMDf148	0000 0000 <sub>H</sub>	<RSCAN0_base> + 090C <sub>H</sub>	8, 16, 32
Receive buffer ID register 49	RSCAN0RMID49	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0910 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 49	RSCAN0RMPTR49	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0914 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 49	RSCAN0RMDf049	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0918 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 49	RSCAN0RMDf149	0000 0000 <sub>H</sub>	<RSCAN0_base> + 091C <sub>H</sub>	8, 16, 32
Receive buffer ID register 50	RSCAN0RMID50	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0920 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 50	RSCAN0RMPTR50	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0924 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 50	RSCAN0RMDf050	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0928 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 50	RSCAN0RMDf150	0000 0000 <sub>H</sub>	<RSCAN0_base> + 092C <sub>H</sub>	8, 16, 32
Receive buffer ID register 51	RSCAN0RMID51	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0930 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 51	RSCAN0RMPTR51	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0934 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 51	RSCAN0RMDf051	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0938 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 51	RSCAN0RMDf151	0000 0000 <sub>H</sub>	<RSCAN0_base> + 093C <sub>H</sub>	8, 16, 32
Receive buffer ID register 52	RSCAN0RMID52	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0940 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 52	RSCAN0RMPTR52	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0944 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 52	RSCAN0RMDf052	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0948 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 52	RSCAN0RMDf152	0000 0000 <sub>H</sub>	<RSCAN0_base> + 094C <sub>H</sub>	8, 16, 32
Receive buffer ID register 53	RSCAN0RMID53	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0950 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 53	RSCAN0RMPTR53	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0954 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 53	RSCAN0RMDf053	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0958 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 53	RSCAN0RMDf153	0000 0000 <sub>H</sub>	<RSCAN0_base> + 095C <sub>H</sub>	8, 16, 32
Receive buffer ID register 54	RSCAN0RMID54	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0960 <sub>H</sub>	8, 16, 32

Table 21.10 List of RS-CAN Module Registers (14/25)

Register Name	Symbol	After Reset	Address	Access Size
Receive buffer pointer register 54	RSCAN0RMPTR54	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0964 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 54	RSCAN0RMDf054	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0968 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 54	RSCAN0RMDf154	0000 0000 <sub>H</sub>	<RSCAN0_base> + 096C <sub>H</sub>	8, 16, 32
Receive buffer ID register 55	RSCAN0RMID55	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0970 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 55	RSCAN0RMPTR55	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0974 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 55	RSCAN0RMDf055	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0978 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 55	RSCAN0RMDf155	0000 0000 <sub>H</sub>	<RSCAN0_base> + 097C <sub>H</sub>	8, 16, 32
Receive buffer ID register 56	RSCAN0RMID56	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0980 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 56	RSCAN0RMPTR56	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0984 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 56	RSCAN0RMDf056	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0988 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 56	RSCAN0RMDf156	0000 0000 <sub>H</sub>	<RSCAN0_base> + 098C <sub>H</sub>	8, 16, 32
Receive buffer ID register 57	RSCAN0RMID57	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0990 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 57	RSCAN0RMPTR57	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0994 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 57	RSCAN0RMDf057	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0998 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 57	RSCAN0RMDf157	0000 0000 <sub>H</sub>	<RSCAN0_base> + 099C <sub>H</sub>	8, 16, 32
Receive buffer ID register 58	RSCAN0RMID58	0000 0000 <sub>H</sub>	<RSCAN0_base> + 09A0 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 58	RSCAN0RMPTR58	0000 0000 <sub>H</sub>	<RSCAN0_base> + 09A4 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 58	RSCAN0RMDf058	0000 0000 <sub>H</sub>	<RSCAN0_base> + 09A8 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 58	RSCAN0RMDf158	0000 0000 <sub>H</sub>	<RSCAN0_base> + 09AC <sub>H</sub>	8, 16, 32
Receive buffer ID register 59	RSCAN0RMID59	0000 0000 <sub>H</sub>	<RSCAN0_base> + 09B0 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 59	RSCAN0RMPTR59	0000 0000 <sub>H</sub>	<RSCAN0_base> + 09B4 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 59	RSCAN0RMDf059	0000 0000 <sub>H</sub>	<RSCAN0_base> + 09B8 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 59	RSCAN0RMDf159	0000 0000 <sub>H</sub>	<RSCAN0_base> + 09BC <sub>H</sub>	8, 16, 32
Receive buffer ID register 60	RSCAN0RMID60	0000 0000 <sub>H</sub>	<RSCAN0_base> + 09C0 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 60	RSCAN0RMPTR60	0000 0000 <sub>H</sub>	<RSCAN0_base> + 09C4 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 60	RSCAN0RMDf060	0000 0000 <sub>H</sub>	<RSCAN0_base> + 09C8 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 60	RSCAN0RMDf160	0000 0000 <sub>H</sub>	<RSCAN0_base> + 09CC <sub>H</sub>	8, 16, 32
Receive buffer ID register 61	RSCAN0RMID61	0000 0000 <sub>H</sub>	<RSCAN0_base> + 09D0 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 61	RSCAN0RMPTR61	0000 0000 <sub>H</sub>	<RSCAN0_base> + 09D4 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 61	RSCAN0RMDf061	0000 0000 <sub>H</sub>	<RSCAN0_base> + 09D8 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 61	RSCAN0RMDf161	0000 0000 <sub>H</sub>	<RSCAN0_base> + 09DC <sub>H</sub>	8, 16, 32
Receive buffer ID register 62	RSCAN0RMID62	0000 0000 <sub>H</sub>	<RSCAN0_base> + 09E0 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 62	RSCAN0RMPTR62	0000 0000 <sub>H</sub>	<RSCAN0_base> + 09E4 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 62	RSCAN0RMDf062	0000 0000 <sub>H</sub>	<RSCAN0_base> + 09E8 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 62	RSCAN0RMDf162	0000 0000 <sub>H</sub>	<RSCAN0_base> + 09EC <sub>H</sub>	8, 16, 32
Receive buffer ID register 63	RSCAN0RMID63	0000 0000 <sub>H</sub>	<RSCAN0_base> + 09F0 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 63	RSCAN0RMPTR63	0000 0000 <sub>H</sub>	<RSCAN0_base> + 09F4 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 63	RSCAN0RMDf063	0000 0000 <sub>H</sub>	<RSCAN0_base> + 09F8 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 63	RSCAN0RMDf163	0000 0000 <sub>H</sub>	<RSCAN0_base> + 09FC <sub>H</sub>	8, 16, 32
Receive buffer ID register 64	RSCAN0RMID64	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A00 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 64	RSCAN0RMPTR64	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A04 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 64	RSCAN0RMDf064	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A08 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 64	RSCAN0RMDf164	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A0C <sub>H</sub>	8, 16, 32
Receive buffer ID register 65	RSCAN0RMID65	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A10 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 65	RSCAN0RMPTR65	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A14 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 65	RSCAN0RMDf065	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A18 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 65	RSCAN0RMDf165	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A1C <sub>H</sub>	8, 16, 32

Table 21.10 List of RS-CAN Module Registers (15/25)

Register Name	Symbol	After Reset	Address	Access Size
Receive buffer ID register 66	RSCAN0RMID66	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A20 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 66	RSCAN0RMPTR66	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A24 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 66	RSCAN0RMD066	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A28 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 66	RSCAN0RMD166	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A2C <sub>H</sub>	8, 16, 32
Receive buffer ID register 67	RSCAN0RMID67	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A30 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 67	RSCAN0RMPTR67	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A34 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 67	RSCAN0RMD067	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A38 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 67	RSCAN0RMD167	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A3C <sub>H</sub>	8, 16, 32
Receive buffer ID register 68	RSCAN0RMID68	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A40 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 68	RSCAN0RMPTR68	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A44 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 68	RSCAN0RMD068	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A48 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 68	RSCAN0RMD168	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A4C <sub>H</sub>	8, 16, 32
Receive buffer ID register 69	RSCAN0RMID69	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A50 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 69	RSCAN0RMPTR69	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A54 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 69	RSCAN0RMD069	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A58 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 69	RSCAN0RMD169	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A5C <sub>H</sub>	8, 16, 32
Receive buffer ID register 70	RSCAN0RMID70	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A60 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 70	RSCAN0RMPTR70	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A64 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 70	RSCAN0RMD070	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A68 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 70	RSCAN0RMD170	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A6C <sub>H</sub>	8, 16, 32
Receive buffer ID register 71	RSCAN0RMID71	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A70 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 71	RSCAN0RMPTR71	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A74 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 71	RSCAN0RMD071	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A78 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 71	RSCAN0RMD171	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A7C <sub>H</sub>	8, 16, 32
Receive buffer ID register 72	RSCAN0RMID72	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A80 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 72	RSCAN0RMPTR72	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A84 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 72	RSCAN0RMD072	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A88 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 72	RSCAN0RMD172	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A8C <sub>H</sub>	8, 16, 32
Receive buffer ID register 73	RSCAN0RMID73	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A90 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 73	RSCAN0RMPTR73	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A94 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 73	RSCAN0RMD073	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A98 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 73	RSCAN0RMD173	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0A9C <sub>H</sub>	8, 16, 32
Receive buffer ID register 74	RSCAN0RMID74	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0AA0 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 74	RSCAN0RMPTR74	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0AA4 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 74	RSCAN0RMD074	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0AA8 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 74	RSCAN0RMD174	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0AAC <sub>H</sub>	8, 16, 32
Receive buffer ID register 75	RSCAN0RMID75	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0AB0 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 75	RSCAN0RMPTR75	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0AB4 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 75	RSCAN0RMD075	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0AB8 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 75	RSCAN0RMD175	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0ABC <sub>H</sub>	8, 16, 32
Receive buffer ID register 76	RSCAN0RMID76	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0AC0 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 76	RSCAN0RMPTR76	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0AC4 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 76	RSCAN0RMD076	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0AC8 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 76	RSCAN0RMD176	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0ACC <sub>H</sub>	8, 16, 32
Receive buffer ID register 77	RSCAN0RMID77	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0AD0 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 77	RSCAN0RMPTR77	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0AD4 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 77	RSCAN0RMD077	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0AD8 <sub>H</sub>	8, 16, 32



Table 21.10 List of RS-CAN Module Registers (16/25)

Register Name	Symbol	After Reset	Address	Access Size
Receive buffer data field 1 register 77	RSCAN0RMDf177	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0ADC <sub>H</sub>	8, 16, 32
Receive buffer ID register 78	RSCAN0RMID78	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0AE0 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 78	RSCAN0RMPTR78	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0AE4 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 78	RSCAN0RMDf078	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0AE8 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 78	RSCAN0RMDf178	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0AEC <sub>H</sub>	8, 16, 32
Receive buffer ID register 79	RSCAN0RMID79	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0AF0 <sub>H</sub>	8, 16, 32
Receive buffer pointer register 79	RSCAN0RMPTR79	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0AF4 <sub>H</sub>	8, 16, 32
Receive buffer data field 0 register 79	RSCAN0RMDf079	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0AF8 <sub>H</sub>	8, 16, 32
Receive buffer data field 1 register 79	RSCAN0RMDf179	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0AFC <sub>H</sub>	8, 16, 32
Receive FIFO buffer access ID register 0	RSCAN0RFID0	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E00 <sub>H</sub>	8, 16, 32
Receive FIFO buffer access pointer register 0	RSCAN0RFPTR0	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E04 <sub>H</sub>	8, 16, 32
Receive FIFO buffer access data field 0 register 0	RSCAN0RFDf00	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E08 <sub>H</sub>	8, 16, 32
Receive FIFO buffer access data field 1 register 0	RSCAN0RFDf10	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E0C <sub>H</sub>	8, 16, 32
Receive FIFO buffer access ID register 1	RSCAN0RFID1	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E10 <sub>H</sub>	8, 16, 32
Receive FIFO buffer access pointer register 1	RSCAN0RFPTR1	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E14 <sub>H</sub>	8, 16, 32
Receive FIFO buffer access data field 0 register 1	RSCAN0RFDf01	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E18 <sub>H</sub>	8, 16, 32
Receive FIFO buffer access data field 1 register 1	RSCAN0RFDf11	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E1C <sub>H</sub>	8, 16, 32
Receive FIFO buffer access ID register 2	RSCAN0RFID2	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E20 <sub>H</sub>	8, 16, 32
Receive FIFO buffer access pointer register 2	RSCAN0RFPTR2	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E24 <sub>H</sub>	8, 16, 32
Receive FIFO buffer access data field 0 register 2	RSCAN0RFDf02	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E28 <sub>H</sub>	8, 16, 32
Receive FIFO buffer access data field 1 register 2	RSCAN0RFDf12	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E2C <sub>H</sub>	8, 16, 32
Receive FIFO buffer access ID register 3	RSCAN0RFID3	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E30 <sub>H</sub>	8, 16, 32
Receive FIFO buffer access pointer register 3	RSCAN0RFPTR3	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E34 <sub>H</sub>	8, 16, 32
Receive FIFO buffer access data field 0 register 3	RSCAN0RFDf03	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E38 <sub>H</sub>	8, 16, 32
Receive FIFO buffer access data field 1 register 3	RSCAN0RFDf13	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E3C <sub>H</sub>	8, 16, 32
Receive FIFO buffer access ID register 4	RSCAN0RFID4	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E40 <sub>H</sub>	8, 16, 32
Receive FIFO buffer access pointer register 4	RSCAN0RFPTR4	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E44 <sub>H</sub>	8, 16, 32
Receive FIFO buffer access data field 0 register 4	RSCAN0RFDf04	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E48 <sub>H</sub>	8, 16, 32
Receive FIFO buffer access data field 1 register 4	RSCAN0RFDf14	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E4C <sub>H</sub>	8, 16, 32
Receive FIFO buffer access ID register 5	RSCAN0RFID5	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E50 <sub>H</sub>	8, 16, 32
Receive FIFO buffer access pointer register 5	RSCAN0RFPTR5	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E54 <sub>H</sub>	8, 16, 32
Receive FIFO buffer access data field 0 register 5	RSCAN0RFDf05	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E58 <sub>H</sub>	8, 16, 32
Receive FIFO buffer access data field 1 register 5	RSCAN0RFDf15	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E5C <sub>H</sub>	8, 16, 32
Receive FIFO buffer access ID register 6	RSCAN0RFID6	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E60 <sub>H</sub>	8, 16, 32
Receive FIFO buffer access pointer register 6	RSCAN0RFPTR6	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E64 <sub>H</sub>	8, 16, 32
Receive FIFO buffer access data field 0 register 6	RSCAN0RFDf06	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E68 <sub>H</sub>	8, 16, 32
Receive FIFO buffer access data field 1 register 6	RSCAN0RFDf16	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E6C <sub>H</sub>	8, 16, 32
Receive FIFO buffer access ID register 7	RSCAN0RFID7	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E70 <sub>H</sub>	8, 16, 32
Receive FIFO buffer access pointer register 7	RSCAN0RFPTR7	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E74 <sub>H</sub>	8, 16, 32
Receive FIFO buffer access data field 0 register 7	RSCAN0RFDf07	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E78 <sub>H</sub>	8, 16, 32
Receive FIFO buffer access data field 1 register 7	RSCAN0RFDf17	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E7C <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer access ID register 0	RSCAN0CFID0	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E80 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer access pointer register 0	RSCAN0CFPTR0	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E84 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer access data field 0 register 0	RSCAN0CFDF00	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E88 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer access data field 1 register 0	RSCAN0CFDF10	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E8C <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer access ID register 1	RSCAN0CFID1	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E90 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer access pointer register 1	RSCAN0CFPTR1	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0E94 <sub>H</sub>	8, 16, 32



Table 21.10 List of RS-CAN Module Registers (18/25)

Register Name	Symbol	After Reset	Address	Access Size
Transmit/receive FIFO buffer access pointer register 13	RSCAN0CFPTR13	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0F54 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer access data field 0 register 13	RSCAN0CFDF013	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0F58 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer access data field 1 register 13	RSCAN0CFDF113	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0F5C <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer access ID register 14	RSCAN0CFID14	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0F60 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer access pointer register 14	RSCAN0CFPTR14	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0F64 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer access data field 0 register 14	RSCAN0CFDF014	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0F68 <sub>H</sub>	8, 16, 32
Transmit/receive FIFO buffer access data field 1 register 14	RSCAN0CFDF114	0000 0000 <sub>H</sub>	<RSCAN0_base> + 0F6C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 0	RSCAN0TMID0	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1000 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 0	RSCAN0TMPTR0	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1004 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 0	RSCAN0TMDF00	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1008 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 0	RSCAN0TMDF10	0000 0000 <sub>H</sub>	<RSCAN0_base> + 100C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 1	RSCAN0TMID1	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1010 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 1	RSCAN0TMPTR1	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1014 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 1	RSCAN0TMDF01	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1018 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 1	RSCAN0TMDF11	0000 0000 <sub>H</sub>	<RSCAN0_base> + 101C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 2	RSCAN0TMID2	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1020 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 2	RSCAN0TMPTR2	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1024 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 2	RSCAN0TMDF02	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1028 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 2	RSCAN0TMDF12	0000 0000 <sub>H</sub>	<RSCAN0_base> + 102C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 3	RSCAN0TMID3	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1030 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 3	RSCAN0TMPTR3	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1034 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 3	RSCAN0TMDF03	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1038 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 3	RSCAN0TMDF13	0000 0000 <sub>H</sub>	<RSCAN0_base> + 103C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 4	RSCAN0TMID4	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1040 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 4	RSCAN0TMPTR4	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1044 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 4	RSCAN0TMDF04	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1048 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 4	RSCAN0TMDF14	0000 0000 <sub>H</sub>	<RSCAN0_base> + <RSCAN0_base> + 104C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 5	RSCAN0TMID5	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1050 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 5	RSCAN0TMPTR5	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1054 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 5	RSCAN0TMDF05	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1058 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 5	RSCAN0TMDF15	0000 0000 <sub>H</sub>	<RSCAN0_base> + 105C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 6	RSCAN0TMID6	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1060 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 6	RSCAN0TMPTR6	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1064 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 6	RSCAN0TMDF06	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1068 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 6	RSCAN0TMDF16	0000 0000 <sub>H</sub>	<RSCAN0_base> + 106C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 7	RSCAN0TMID7	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1070 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 7	RSCAN0TMPTR7	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1074 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 7	RSCAN0TMDF07	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1078 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 7	RSCAN0TMDF17	0000 0000 <sub>H</sub>	<RSCAN0_base> + 107C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 8	RSCAN0TMID8	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1080 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 8	RSCAN0TMPTR8	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1084 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 8	RSCAN0TMDF08	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1088 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 8	RSCAN0TMDF18	0000 0000 <sub>H</sub>	<RSCAN0_base> + 108C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 9	RSCAN0TMID9	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1090 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 9	RSCAN0TMPTR9	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1094 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 9	RSCAN0TMDF09	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1098 <sub>H</sub>	8, 16, 32

Table 21.10 List of RS-CAN Module Registers (19/25)

Register Name	Symbol	After Reset	Address	Access Size
Transmit buffer data field 1 register 9	RSCAN0TMDF19	0000 0000 <sub>H</sub>	<RSCAN0_base> + 109C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 10	RSCAN0TMID10	0000 0000 <sub>H</sub>	<RSCAN0_base> + 10A0 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 10	RSCAN0TMPTR10	0000 0000 <sub>H</sub>	<RSCAN0_base> + 10A4 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 10	RSCAN0TMDF010	0000 0000 <sub>H</sub>	<RSCAN0_base> + 10A8 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 10	RSCAN0TMDF110	0000 0000 <sub>H</sub>	<RSCAN0_base> + 10AC <sub>H</sub>	8, 16, 32
Transmit buffer ID register 11	RSCAN0TMID11	0000 0000 <sub>H</sub>	<RSCAN0_base> + 10B0 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 11	RSCAN0TMPTR11	0000 0000 <sub>H</sub>	<RSCAN0_base> + 10B4 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 11	RSCAN0TMDF011	0000 0000 <sub>H</sub>	<RSCAN0_base> + 10B8 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 11	RSCAN0TMDF111	0000 0000 <sub>H</sub>	<RSCAN0_base> + 10BC <sub>H</sub>	8, 16, 32
Transmit buffer ID register 12	RSCAN0TMID12	0000 0000 <sub>H</sub>	<RSCAN0_base> + 10C0 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 12	RSCAN0TMPTR12	0000 0000 <sub>H</sub>	<RSCAN0_base> + 10C4 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 12	RSCAN0TMDF012	0000 0000 <sub>H</sub>	<RSCAN0_base> + 10C8 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 12	RSCAN0TMDF112	0000 0000 <sub>H</sub>	<RSCAN0_base> + 10CC <sub>H</sub>	8, 16, 32
Transmit buffer ID register 13	RSCAN0TMID13	0000 0000 <sub>H</sub>	<RSCAN0_base> + 10D0 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 13	RSCAN0TMPTR13	0000 0000 <sub>H</sub>	<RSCAN0_base> + 10D4 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 13	RSCAN0TMDF013	0000 0000 <sub>H</sub>	<RSCAN0_base> + 10D8 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 13	RSCAN0TMDF113	0000 0000 <sub>H</sub>	<RSCAN0_base> + 10DC <sub>H</sub>	8, 16, 32
Transmit buffer ID register 14	RSCAN0TMID14	0000 0000 <sub>H</sub>	<RSCAN0_base> + 10E0 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 14	RSCAN0TMPTR14	0000 0000 <sub>H</sub>	<RSCAN0_base> + 10E4 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 14	RSCAN0TMDF014	0000 0000 <sub>H</sub>	<RSCAN0_base> + 10E8 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 14	RSCAN0TMDF114	0000 0000 <sub>H</sub>	<RSCAN0_base> + 10EC <sub>H</sub>	8, 16, 32
Transmit buffer ID register 15	RSCAN0TMID15	0000 0000 <sub>H</sub>	<RSCAN0_base> + 10F0 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 15	RSCAN0TMPTR15	0000 0000 <sub>H</sub>	<RSCAN0_base> + 10F4 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 15	RSCAN0TMDF015	0000 0000 <sub>H</sub>	<RSCAN0_base> + 10F8 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 15	RSCAN0TMDF115	0000 0000 <sub>H</sub>	<RSCAN0_base> + 10FC <sub>H</sub>	8, 16, 32
Transmit buffer ID register 16	RSCAN0TMID16	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1100 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 16	RSCAN0TMPTR16	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1104 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 16	RSCAN0TMDF016	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1108 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 16	RSCAN0TMDF116	0000 0000 <sub>H</sub>	<RSCAN0_base> + 110C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 17	RSCAN0TMID17	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1110 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 17	RSCAN0TMPTR17	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1114 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 17	RSCAN0TMDF017	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1118 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 17	RSCAN0TMDF117	0000 0000 <sub>H</sub>	<RSCAN0_base> + 111C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 18	RSCAN0TMID18	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1120 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 18	RSCAN0TMPTR18	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1124 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 18	RSCAN0TMDF018	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1128 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 18	RSCAN0TMDF118	0000 0000 <sub>H</sub>	<RSCAN0_base> + 112C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 19	RSCAN0TMID19	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1130 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 19	RSCAN0TMPTR19	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1134 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 19	RSCAN0TMDF019	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1138 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 19	RSCAN0TMDF119	0000 0000 <sub>H</sub>	<RSCAN0_base> + 113C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 20	RSCAN0TMID20	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1140 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 20	RSCAN0TMPTR20	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1144 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 20	RSCAN0TMDF020	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1148 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 20	RSCAN0TMDF120	0000 0000 <sub>H</sub>	<RSCAN0_base> + 114C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 21	RSCAN0TMID21	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1150 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 21	RSCAN0TMPTR21	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1154 <sub>H</sub>	8, 16, 32

Table 21.10 List of RS-CAN Module Registers (20/25)

Register Name	Symbol	After Reset	Address	Access Size
Transmit buffer data field 0 register 21	RSCAN0TMDF021	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1158 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 21	RSCAN0TMDF121	0000 0000 <sub>H</sub>	<RSCAN0_base> + 115C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 22	RSCAN0TMID22	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1160 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 22	RSCAN0TMPTR22	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1164 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 22	RSCAN0TMDF022	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1168 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 22	RSCAN0TMDF122	0000 0000 <sub>H</sub>	<RSCAN0_base> + 116C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 23	RSCAN0TMID23	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1170 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 23	RSCAN0TMPTR23	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1174 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 23	RSCAN0TMDF023	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1178 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 23	RSCAN0TMDF123	0000 0000 <sub>H</sub>	<RSCAN0_base> + 117C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 24	RSCAN0TMID24	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1180 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 24	RSCAN0TMPTR24	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1184 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 24	RSCAN0TMDF024	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1188 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 24	RSCAN0TMDF124	0000 0000 <sub>H</sub>	<RSCAN0_base> + 118C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 25	RSCAN0TMID25	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1190 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 25	RSCAN0TMPTR25	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1194 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 25	RSCAN0TMDF025	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1198 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 25	RSCAN0TMDF125	0000 0000 <sub>H</sub>	<RSCAN0_base> + 119C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 26	RSCAN0TMID26	0000 0000 <sub>H</sub>	<RSCAN0_base> + 11A0 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 26	RSCAN0TMPTR26	0000 0000 <sub>H</sub>	<RSCAN0_base> + 11A4 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 26	RSCAN0TMDF026	0000 0000 <sub>H</sub>	<RSCAN0_base> + 11A8 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 26	RSCAN0TMDF126	0000 0000 <sub>H</sub>	<RSCAN0_base> + 11AC <sub>H</sub>	8, 16, 32
Transmit buffer ID register 27	RSCAN0TMID27	0000 0000 <sub>H</sub>	<RSCAN0_base> + 11B0 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 27	RSCAN0TMPTR27	0000 0000 <sub>H</sub>	<RSCAN0_base> + 11B4 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 27	RSCAN0TMDF027	0000 0000 <sub>H</sub>	<RSCAN0_base> + 11B8 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 27	RSCAN0TMDF127	0000 0000 <sub>H</sub>	<RSCAN0_base> + 11BC <sub>H</sub>	8, 16, 32
Transmit buffer ID register 28	RSCAN0TMID28	0000 0000 <sub>H</sub>	<RSCAN0_base> + 11C0 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 28	RSCAN0TMPTR28	0000 0000 <sub>H</sub>	<RSCAN0_base> + 11C4 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 28	RSCAN0TMDF028	0000 0000 <sub>H</sub>	<RSCAN0_base> + 11C8 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 28	RSCAN0TMDF128	0000 0000 <sub>H</sub>	<RSCAN0_base> + 11CC <sub>H</sub>	8, 16, 32
Transmit buffer ID register 29	RSCAN0TMID29	0000 0000 <sub>H</sub>	<RSCAN0_base> + 11D0 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 29	RSCAN0TMPTR29	0000 0000 <sub>H</sub>	<RSCAN0_base> + 11D4 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 29	RSCAN0TMDF029	0000 0000 <sub>H</sub>	<RSCAN0_base> + 11D8 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 29	RSCAN0TMDF129	0000 0000 <sub>H</sub>	<RSCAN0_base> + 11DC <sub>H</sub>	8, 16, 32
Transmit buffer ID register 30	RSCAN0TMID30	0000 0000 <sub>H</sub>	<RSCAN0_base> + 11E0 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 30	RSCAN0TMPTR30	0000 0000 <sub>H</sub>	<RSCAN0_base> + 11E4 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 30	RSCAN0TMDF030	0000 0000 <sub>H</sub>	<RSCAN0_base> + 11E8 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 30	RSCAN0TMDF130	0000 0000 <sub>H</sub>	<RSCAN0_base> + 11EC <sub>H</sub>	8, 16, 32
Transmit buffer ID register 31	RSCAN0TMID31	0000 0000 <sub>H</sub>	<RSCAN0_base> + 11F0 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 31	RSCAN0TMPTR31	0000 0000 <sub>H</sub>	<RSCAN0_base> + 11F4 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 31	RSCAN0TMDF031	0000 0000 <sub>H</sub>	<RSCAN0_base> + 11F8 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 31	RSCAN0TMDF131	0000 0000 <sub>H</sub>	<RSCAN0_base> + 11FC <sub>H</sub>	8, 16, 32
Transmit buffer ID register 32	RSCAN0TMID32	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1200 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 32	RSCAN0TMPTR32	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1204 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 32	RSCAN0TMDF032	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1208 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 32	RSCAN0TMDF132	0000 0000 <sub>H</sub>	<RSCAN0_base> + 120C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 33	RSCAN0TMID33	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1210 <sub>H</sub>	8, 16, 32

Table 21.10 List of RS-CAN Module Registers (21/25)

Register Name	Symbol	After Reset	Address	Access Size
Transmit buffer pointer register 33	RSCAN0TMPTR33	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1214 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 33	RSCAN0TMDF033	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1218 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 33	RSCAN0TMDF133	0000 0000 <sub>H</sub>	<RSCAN0_base> + 121C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 34	RSCAN0TMID34	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1220 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 34	RSCAN0TMPTR34	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1224 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 34	RSCAN0TMDF034	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1228 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 34	RSCAN0TMDF134	0000 0000 <sub>H</sub>	<RSCAN0_base> + 122C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 35	RSCAN0TMID35	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1230 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 35	RSCAN0TMPTR35	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1234 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 35	RSCAN0TMDF035	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1238 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 35	RSCAN0TMDF135	0000 0000 <sub>H</sub>	<RSCAN0_base> + 123C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 36	RSCAN0TMID36	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1240 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 36	RSCAN0TMPTR36	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1244 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 36	RSCAN0TMDF036	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1248 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 36	RSCAN0TMDF136	0000 0000 <sub>H</sub>	<RSCAN0_base> + 124C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 37	RSCAN0TMID37	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1250 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 37	RSCAN0TMPTR37	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1254 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 37	RSCAN0TMDF037	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1258 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 37	RSCAN0TMDF137	0000 0000 <sub>H</sub>	<RSCAN0_base> + 125C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 38	RSCAN0TMID38	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1260 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 38	RSCAN0TMPTR38	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1264 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 38	RSCAN0TMDF038	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1268 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 38	RSCAN0TMDF138	0000 0000 <sub>H</sub>	<RSCAN0_base> + 126C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 39	RSCAN0TMID39	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1270 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 39	RSCAN0TMPTR39	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1274 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 39	RSCAN0TMDF039	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1278 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 39	RSCAN0TMDF139	0000 0000 <sub>H</sub>	<RSCAN0_base> + 127C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 40	RSCAN0TMID40	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1280 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 40	RSCAN0TMPTR40	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1284 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 40	RSCAN0TMDF040	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1288 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 40	RSCAN0TMDF140	0000 0000 <sub>H</sub>	<RSCAN0_base> + 128C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 41	RSCAN0TMID41	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1290 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 41	RSCAN0TMPTR41	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1294 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 41	RSCAN0TMDF041	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1298 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 41	RSCAN0TMDF141	0000 0000 <sub>H</sub>	<RSCAN0_base> + 129C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 42	RSCAN0TMID42	0000 0000 <sub>H</sub>	<RSCAN0_base> + 12A0 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 42	RSCAN0TMPTR42	0000 0000 <sub>H</sub>	<RSCAN0_base> + 12A4 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 42	RSCAN0TMDF042	0000 0000 <sub>H</sub>	<RSCAN0_base> + 12A8 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 42	RSCAN0TMDF142	0000 0000 <sub>H</sub>	<RSCAN0_base> + 12AC <sub>H</sub>	8, 16, 32
Transmit buffer ID register 43	RSCAN0TMID43	0000 0000 <sub>H</sub>	<RSCAN0_base> + 12B0 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 43	RSCAN0TMPTR43	0000 0000 <sub>H</sub>	<RSCAN0_base> + 12B4 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 43	RSCAN0TMDF043	0000 0000 <sub>H</sub>	<RSCAN0_base> + 12B8 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 43	RSCAN0TMDF143	0000 0000 <sub>H</sub>	<RSCAN0_base> + 12BC <sub>H</sub>	8, 16, 32
Transmit buffer ID register 44	RSCAN0TMID44	0000 0000 <sub>H</sub>	<RSCAN0_base> + 12C0 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 44	RSCAN0TMPTR44	0000 0000 <sub>H</sub>	<RSCAN0_base> + 12C4 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 44	RSCAN0TMDF044	0000 0000 <sub>H</sub>	<RSCAN0_base> + 12C8 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 44	RSCAN0TMDF144	0000 0000 <sub>H</sub>	<RSCAN0_base> + 12CC <sub>H</sub>	8, 16, 32

Table 21.10 List of RS-CAN Module Registers (22/25)

Register Name	Symbol	After Reset	Address	Access Size
Transmit buffer ID register 45	RSCAN0TMID45	0000 0000 <sub>H</sub>	<RSCAN0_base> + 12D0 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 45	RSCAN0TMPTR45	0000 0000 <sub>H</sub>	<RSCAN0_base> + 12D4 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 45	RSCAN0TMDF045	0000 0000 <sub>H</sub>	<RSCAN0_base> + 12D8 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 45	RSCAN0TMDF145	0000 0000 <sub>H</sub>	<RSCAN0_base> + 12DC <sub>H</sub>	8, 16, 32
Transmit buffer ID register 46	RSCAN0TMID46	0000 0000 <sub>H</sub>	<RSCAN0_base> + 12E0 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 46	RSCAN0TMPTR46	0000 0000 <sub>H</sub>	<RSCAN0_base> + 12E4 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 46	RSCAN0TMDF046	0000 0000 <sub>H</sub>	<RSCAN0_base> + 12E8 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 46	RSCAN0TMDF146	0000 0000 <sub>H</sub>	<RSCAN0_base> + 12EC <sub>H</sub>	8, 16, 32
Transmit buffer ID register 47	RSCAN0TMID47	0000 0000 <sub>H</sub>	<RSCAN0_base> + 12F0 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 47	RSCAN0TMPTR47	0000 0000 <sub>H</sub>	<RSCAN0_base> + 12F4 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 47	RSCAN0TMDF047	0000 0000 <sub>H</sub>	<RSCAN0_base> + 12F8 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 47	RSCAN0TMDF147	0000 0000 <sub>H</sub>	<RSCAN0_base> + 12FC <sub>H</sub>	8, 16, 32
Transmit buffer ID register 48	RSCAN0TMID48	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1300 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 48	RSCAN0TMPTR48	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1304 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 48	RSCAN0TMDF048	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1308 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 48	RSCAN0TMDF148	0000 0000 <sub>H</sub>	<RSCAN0_base> + 130C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 49	RSCAN0TMID49	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1310 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 49	RSCAN0TMPTR49	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1314 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 49	RSCAN0TMDF049	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1318 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 49	RSCAN0TMDF149	0000 0000 <sub>H</sub>	<RSCAN0_base> + 131C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 50	RSCAN0TMID50	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1320 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 50	RSCAN0TMPTR50	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1324 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 50	RSCAN0TMDF050	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1328 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 50	RSCAN0TMDF150	0000 0000 <sub>H</sub>	<RSCAN0_base> + 132C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 51	RSCAN0TMID51	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1330 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 51	RSCAN0TMPTR51	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1334 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 51	RSCAN0TMDF051	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1338 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 51	RSCAN0TMDF151	0000 0000 <sub>H</sub>	<RSCAN0_base> + 133C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 52	RSCAN0TMID52	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1340 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 52	RSCAN0TMPTR52	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1344 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 52	RSCAN0TMDF052	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1348 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 52	RSCAN0TMDF152	0000 0000 <sub>H</sub>	<RSCAN0_base> + 134C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 53	RSCAN0TMID53	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1350 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 53	RSCAN0TMPTR53	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1354 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 53	RSCAN0TMDF053	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1358 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 53	RSCAN0TMDF153	0000 0000 <sub>H</sub>	<RSCAN0_base> + 135C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 54	RSCAN0TMID54	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1360 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 54	RSCAN0TMPTR54	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1364 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 54	RSCAN0TMDF054	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1368 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 54	RSCAN0TMDF154	0000 0000 <sub>H</sub>	<RSCAN0_base> + 136C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 55	RSCAN0TMID55	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1370 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 55	RSCAN0TMPTR55	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1374 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 55	RSCAN0TMDF055	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1378 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 55	RSCAN0TMDF155	0000 0000 <sub>H</sub>	<RSCAN0_base> + 137C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 56	RSCAN0TMID56	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1380 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 56	RSCAN0TMPTR56	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1384 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 56	RSCAN0TMDF056	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1388 <sub>H</sub>	8, 16, 32

Table 21.10 List of RS-CAN Module Registers (23/25)

Register Name	Symbol	After Reset	Address	Access Size
Transmit buffer data field 1 register 56	RSCAN0TMDF156	0000 0000 <sub>H</sub>	<RSCAN0_base> + 138C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 57	RSCAN0TMID57	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1390 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 57	RSCAN0TMPTR57	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1394 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 57	RSCAN0TMDF057	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1398 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 57	RSCAN0TMDF157	0000 0000 <sub>H</sub>	<RSCAN0_base> + 139C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 58	RSCAN0TMID58	0000 0000 <sub>H</sub>	<RSCAN0_base> + 13A0 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 58	RSCAN0TMPTR58	0000 0000 <sub>H</sub>	<RSCAN0_base> + 13A4 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 58	RSCAN0TMDF058	0000 0000 <sub>H</sub>	<RSCAN0_base> + 13A8 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 58	RSCAN0TMDF158	0000 0000 <sub>H</sub>	<RSCAN0_base> + 13AC <sub>H</sub>	8, 16, 32
Transmit buffer ID register 59	RSCAN0TMID59	0000 0000 <sub>H</sub>	<RSCAN0_base> + 13B0 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 59	RSCAN0TMPTR59	0000 0000 <sub>H</sub>	<RSCAN0_base> + 13B4 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 59	RSCAN0TMDF059	0000 0000 <sub>H</sub>	<RSCAN0_base> + 13B8 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 59	RSCAN0TMDF159	0000 0000 <sub>H</sub>	<RSCAN0_base> + 13BC <sub>H</sub>	8, 16, 32
Transmit buffer ID register 60	RSCAN0TMID60	0000 0000 <sub>H</sub>	<RSCAN0_base> + 13C0 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 60	RSCAN0TMPTR60	0000 0000 <sub>H</sub>	<RSCAN0_base> + 13C4 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 60	RSCAN0TMDF060	0000 0000 <sub>H</sub>	<RSCAN0_base> + 13C8 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 60	RSCAN0TMDF160	0000 0000 <sub>H</sub>	<RSCAN0_base> + 13CC <sub>H</sub>	8, 16, 32
Transmit buffer ID register 61	RSCAN0TMID61	0000 0000 <sub>H</sub>	<RSCAN0_base> + 13D0 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 61	RSCAN0TMPTR61	0000 0000 <sub>H</sub>	<RSCAN0_base> + 13D4 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 61	RSCAN0TMDF061	0000 0000 <sub>H</sub>	<RSCAN0_base> + 13D8 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 1	RSCAN0TMDF161	0000 0000 <sub>H</sub>	<RSCAN0_base> + 13DC <sub>H</sub>	8, 16, 32
Transmit buffer ID register 62	RSCAN0TMID62	0000 0000 <sub>H</sub>	<RSCAN0_base> + 13E0 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 62	RSCAN0TMPTR62	0000 0000 <sub>H</sub>	<RSCAN0_base> + 13E4 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 62	RSCAN0TMDF062	0000 0000 <sub>H</sub>	<RSCAN0_base> + 13E8 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 62	RSCAN0TMDF162	0000 0000 <sub>H</sub>	<RSCAN0_base> + 13EC <sub>H</sub>	8, 16, 32
Transmit buffer ID register 63	RSCAN0TMID63	0000 0000 <sub>H</sub>	<RSCAN0_base> + 13F0 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 63	RSCAN0TMPTR63	0000 0000 <sub>H</sub>	<RSCAN0_base> + 13F4 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 63	RSCAN0TMDF063	0000 0000 <sub>H</sub>	<RSCAN0_base> + 13F8 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 63	RSCAN0TMDF163	0000 0000 <sub>H</sub>	<RSCAN0_base> + 13FC <sub>H</sub>	8, 16, 32
Transmit buffer ID register 64	RSCAN0TMID64	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1400 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 64	RSCAN0TMPTR64	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1404 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 64	RSCAN0TMDF064	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1408 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 64	RSCAN0TMDF164	0000 0000 <sub>H</sub>	<RSCAN0_base> + 140C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 65	RSCAN0TMID65	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1410 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 65	RSCAN0TMPTR65	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1414 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 65	RSCAN0TMDF065	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1418 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 65	RSCAN0TMDF165	0000 0000 <sub>H</sub>	<RSCAN0_base> + 141C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 66	RSCAN0TMID66	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1420 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 66	RSCAN0TMPTR66	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1424 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 66	RSCAN0TMDF066	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1428 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 66	RSCAN0TMDF166	0000 0000 <sub>H</sub>	<RSCAN0_base> + 142C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 67	RSCAN0TMID67	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1430 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 67	RSCAN0TMPTR67	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1434 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 67	RSCAN0TMDF067	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1438 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 67	RSCAN0TMDF167	0000 0000 <sub>H</sub>	<RSCAN0_base> + 143C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 68	RSCAN0TMID68	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1440 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 68	RSCAN0TMPTR68	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1444 <sub>H</sub>	8, 16, 32



Table 21.10 List of RS-CAN Module Registers (24/25)

Register Name	Symbol	After Reset	Address	Access Size
Transmit buffer data field 0 register 68	RSCAN0TMDF068	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1448 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 68	RSCAN0TMDF168	0000 0000 <sub>H</sub>	<RSCAN0_base> + 144C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 69	RSCAN0TMID69	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1450 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 69	RSCAN0TMPTR69	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1454 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 69	RSCAN0TMDF069	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1458 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 69	RSCAN0TMDF169	0000 0000 <sub>H</sub>	<RSCAN0_base> + 145C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 70	RSCAN0TMID70	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1460 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 70	RSCAN0TMPTR70	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1464 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 70	RSCAN0TMDF070	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1468 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 70	RSCAN0TMDF170	0000 0000 <sub>H</sub>	<RSCAN0_base> + 146C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 71	RSCAN0TMID71	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1470 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 71	RSCAN0TMPTR71	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1474 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 71	RSCAN0TMDF071	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1478 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 71	RSCAN0TMDF171	0000 0000 <sub>H</sub>	<RSCAN0_base> + 147C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 72	RSCAN0TMID72	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1480 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 72	RSCAN0TMPTR72	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1484 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 72	RSCAN0TMDF072	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1488 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 72	RSCAN0TMDF172	0000 0000 <sub>H</sub>	<RSCAN0_base> + 148C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 73	RSCAN0TMID73	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1490 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 73	RSCAN0TMPTR73	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1494 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 73	RSCAN0TMDF073	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1498 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 73	RSCAN0TMDF173	0000 0000 <sub>H</sub>	<RSCAN0_base> + 149C <sub>H</sub>	8, 16, 32
Transmit buffer ID register 74	RSCAN0TMID74	0000 0000 <sub>H</sub>	<RSCAN0_base> + 14A0 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 74	RSCAN0TMPTR74	0000 0000 <sub>H</sub>	<RSCAN0_base> + 14A4 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 74	RSCAN0TMDF074	0000 0000 <sub>H</sub>	<RSCAN0_base> + 14A8 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 74	RSCAN0TMDF174	0000 0000 <sub>H</sub>	<RSCAN0_base> + 14AC <sub>H</sub>	8, 16, 32
Transmit buffer ID register 75	RSCAN0TMID75	0000 0000 <sub>H</sub>	<RSCAN0_base> + 14B0 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 75	RSCAN0TMPTR75	0000 0000 <sub>H</sub>	<RSCAN0_base> + 14B4 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 75	RSCAN0TMDF075	0000 0000 <sub>H</sub>	<RSCAN0_base> + 14B8 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 75	RSCAN0TMDF175	0000 0000 <sub>H</sub>	<RSCAN0_base> + 14BC <sub>H</sub>	8, 16, 32
Transmit buffer ID register 76	RSCAN0TMID76	0000 0000 <sub>H</sub>	<RSCAN0_base> + 14C0 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 76	RSCAN0TMPTR76	0000 0000 <sub>H</sub>	<RSCAN0_base> + 14C4 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 76	RSCAN0TMDF076	0000 0000 <sub>H</sub>	<RSCAN0_base> + 14C8 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 76	RSCAN0TMDF176	0000 0000 <sub>H</sub>	<RSCAN0_base> + 14CC <sub>H</sub>	8, 16, 32
Transmit buffer ID register 77	RSCAN0TMID77	0000 0000 <sub>H</sub>	<RSCAN0_base> + 14D0 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 77	RSCAN0TMPTR77	0000 0000 <sub>H</sub>	<RSCAN0_base> + 14D4 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 77	RSCAN0TMDF077	0000 0000 <sub>H</sub>	<RSCAN0_base> + 14D8 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 77	RSCAN0TMDF177	0000 0000 <sub>H</sub>	<RSCAN0_base> + 14DC <sub>H</sub>	8, 16, 32
Transmit buffer ID register 78	RSCAN0TMID78	0000 0000 <sub>H</sub>	<RSCAN0_base> + 14E0 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 78	RSCAN0TMPTR78	0000 0000 <sub>H</sub>	<RSCAN0_base> + 14E4 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 78	RSCAN0TMDF078	0000 0000 <sub>H</sub>	<RSCAN0_base> + 14E8 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 78	RSCAN0TMDF178	0000 0000 <sub>H</sub>	<RSCAN0_base> + 14EC <sub>H</sub>	8, 16, 32
Transmit buffer ID register 79	RSCAN0TMID79	0000 0000 <sub>H</sub>	<RSCAN0_base> + 14F0 <sub>H</sub>	8, 16, 32
Transmit buffer pointer register 79	RSCAN0TMPTR79	0000 0000 <sub>H</sub>	<RSCAN0_base> + 14F4 <sub>H</sub>	8, 16, 32
Transmit buffer data field 0 register 79	RSCAN0TMDF079	0000 0000 <sub>H</sub>	<RSCAN0_base> + 14F8 <sub>H</sub>	8, 16, 32
Transmit buffer data field 1 register 79	RSCAN0TMDF179	0000 0000 <sub>H</sub>	<RSCAN0_base> + 14FC <sub>H</sub>	8, 16, 32
Transmit history access register 0	RSCAN0THLACC0	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1800 <sub>H</sub>	8, 16, 32

Table 21.10 List of RS-CAN Module Registers (25/25)

Register Name	Symbol	After Reset	Address	Access Size
Transmit history access register 1	RSCAN0THLACC1	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1804 <sub>H</sub>	8, 16, 32
Transmit history access register 2	RSCAN0THLACC2	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1808 <sub>H</sub>	8, 16, 32
Transmit history access register 3	RSCAN0THLACC3	0000 0000 <sub>H</sub>	<RSCAN0_base> + 180C <sub>H</sub>	8, 16, 32
Transmit history access register 4	RSCAN0THLACC4	0000 0000 <sub>H</sub>	<RSCAN0_base> + 1810 <sub>H</sub>	8, 16, 32

Table 21.11 Transmit Buffer p Allocated to Each Channel

	CANn
Transmit buffer p	Transmit buffer 16 × m + 0
	Transmit buffer 16 × m + 1
	Transmit buffer 16 × m + 2
	Transmit buffer 16 × m + 3
	Transmit buffer 16 × m + 4
	Transmit buffer 16 × m + 5
	Transmit buffer 16 × m + 6
	Transmit buffer 16 × m + 7
	Transmit buffer 16 × m + 8
	Transmit buffer 16 × m + 9
	Transmit buffer 16 × m + 10
	Transmit buffer 16 × m + 11
	Transmit buffer 16 × m + 12
	Transmit buffer 16 × m + 13
	Transmit buffer 16 × m + 14
Transmit buffer 16 × m + 15	

Table 21.12 Transmit/Receive FIFO Buffer k Allocated to Each Channel

	CANn
Transmit/receive FIFO buffer k	Transmit/receive FIFO buffer 3 × m + 0
	Transmit/receive FIFO buffer 3 × m + 1
	Transmit/receive FIFO buffer 3 × m + 2

**Table 21.13** Transmit Buffer p Linked to the Transmit/Receive FIFO Buffer by the Setting of Bits CFTML[3:0]

Setting of Bits CFTML[3:0]	Transmit Buffer p Linked to the Transmit/Receive FIFO Buffer
0000 <sub>B</sub>	Transmit buffer $16 \times m + 0$
0001 <sub>B</sub>	Transmit buffer $16 \times m + 1$
0010 <sub>B</sub>	Transmit buffer $16 \times m + 2$
0011 <sub>B</sub>	Transmit buffer $16 \times m + 3$
0100 <sub>B</sub>	Transmit buffer $16 \times m + 4$
0101 <sub>B</sub>	Transmit buffer $16 \times m + 5$
0110 <sub>B</sub>	Transmit buffer $16 \times m + 6$
0111 <sub>B</sub>	Transmit buffer $16 \times m + 7$
1000 <sub>B</sub>	Transmit buffer $16 \times m + 8$
1001 <sub>B</sub>	Transmit buffer $16 \times m + 9$
1010 <sub>B</sub>	Transmit buffer $16 \times m + 10$
1011 <sub>B</sub>	Transmit buffer $16 \times m + 11$
1100 <sub>B</sub>	Transmit buffer $16 \times m + 12$
1101 <sub>B</sub>	Transmit buffer $16 \times m + 13$
1110 <sub>B</sub>	Transmit buffer $16 \times m + 14$
1111 <sub>B</sub>	Transmit buffer $16 \times m + 15$

**Table 21.14** Transmit Buffer p Allocated to the Transmit Queue of Each Channel

Setting of Bits TXQDC[3:0]	Transmit Buffer p Allocated to the Transmit Queue
0000 <sub>B</sub>	Setting prohibited
0001 <sub>B</sub>	Setting prohibited
0010 <sub>B</sub>	Transmit buffer $16 \times m + 15$ to $16 \times m + 13$
0011 <sub>B</sub>	Transmit buffer $16 \times m + 15$ to $16 \times m + 12$
0100 <sub>B</sub>	Transmit buffer $16 \times m + 15$ to $16 \times m + 11$
0101 <sub>B</sub>	Transmit buffer $16 \times m + 15$ to $16 \times m + 10$
0110 <sub>B</sub>	Transmit buffer $16 \times m + 15$ to $16 \times m + 9$
0111 <sub>B</sub>	Transmit buffer $16 \times m + 15$ to $16 \times m + 8$
1000 <sub>B</sub>	Transmit buffer $16 \times m + 15$ to $16 \times m + 7$
1001 <sub>B</sub>	Transmit buffer $16 \times m + 15$ to $16 \times m + 6$
1010 <sub>B</sub>	Transmit buffer $16 \times m + 15$ to $16 \times m + 5$
1011 <sub>B</sub>	Transmit buffer $16 \times m + 15$ to $16 \times m + 4$
1100 <sub>B</sub>	Transmit buffer $16 \times m + 15$ to $16 \times m + 3$
1101 <sub>B</sub>	Transmit buffer $16 \times m + 15$ to $16 \times m + 2$
1110 <sub>B</sub>	Transmit buffer $16 \times m + 15$ to $16 \times m + 1$
1111 <sub>B</sub>	Transmit buffer $16 \times m + 15$ to $16 \times m + 0$

### 21.3.1 RSCAN0CmCFG — Channel Configuration Register (m = 0 to 4)

**Access:** Can be read/written in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0000<sub>H</sub> + (m \* 0010<sub>H</sub>)

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	SJW[1:0]		—	TSEG2[2:0]			TSEG1[3:0]			
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	BRP[9:0]									
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

**Table 21.15 RSCAN0CmCFG register contents**

Bit Position	Bit Name	Function
31 to 26	Reserved	These bits are always read as 0. The write value should always be 0.
25, 24	SJW[1:0]	Resynchronization Jump Width Control b25 b24 0 0: 1 Tq 0 1: 2 Tq 1 0: 3 Tq 1 1: 4 Tq
23	Reserved	This bit is always read as 0. The write value should always be 0.
22 to 20	TSEG2[2:0]	Time Segment 2 Control b22 b21 b20 0 0 0: Setting prohibited 0 0 1: 2 Tq 0 1 0: 3 Tq 0 1 1: 4 Tq 1 0 0: 5 Tq 1 0 1: 6 Tq 1 1 0: 7 Tq 1 1 1: 8 Tq
19 to 16	TSEG1[3:0]	Time Segment 1 Control b19 b18 b17 b16 0 0 0 0: Setting prohibited 0 0 0 1: Setting prohibited 0 0 1 0: Setting prohibited 0 0 1 1: 4 Tq 0 1 0 0: 5 Tq 0 1 0 1: 6 Tq 0 1 1 0: 7 Tq 0 1 1 1: 8 Tq 1 0 0 0: 9 Tq 1 0 0 1: 10 Tq 1 0 1 0: 11 Tq 1 0 1 1: 12 Tq 1 1 0 0: 13 Tq 1 1 0 1: 14 Tq 1 1 1 0: 15 Tq 1 1 1 1: 16 Tq
15 to 10	Reserved	These bits are always read as 0. The write value should always be 0.
9 to 0	BRP[9:0]	Prescaler Division Ratio Set When these bits are set to P (0 to 1023), the baud rate prescaler divides fCAN by P + 1.

Modify the RSCAN0CmCFG register in channel reset mode or channel halt mode. Set this register before requesting a transition to channel communication mode or channel wait mode. For a description of the bit timing parameters and settings, see **Section 21.10.1, Initial Settings**.

**SJW[1:0] Bits**

These bits are used to specify a Tq value for the resynchronization jump width. Allowed values are 1 Tq to 4 Tq, inclusive.

Set a value less than or equal to the value of the TSEG2 bits.

**TSEG2[2:0] Bits**

These bits are used to specify a Tq value for the length of phase buffer segment 2 (PHASE\_SEG2). Allowed values are 2 Tq to 8 Tq, inclusive.

Set a value smaller than the value of the TSEG1 bits.

**TSEG1[3:0] Bits**

These bits are used to specify a Tq value for the total length of the propagation time segment (PROP\_SEG) and phase buffer segment 1 (PHASE\_SEG1).

Allowed values are 4 Tq to 16 Tq, inclusive.

**BRP[9:0] Bits**

The CANmTq (fCANTQm) clock is calculated by dividing the CAN clock (fCAN) by the baud rate prescaler, ((BRP[9:0] + 1). One clock cycle of the CANmTq clock is 1 Time Quantum (Tq).

### 21.3.2 RSCAN0CmCTR — Channel Control Register (m = 0 to 4)

**Access:** Can be read/written in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0004<sub>H</sub> + (m \* 0010<sub>H</sub>)

**Initial value:** 0000 0005<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	CTMS[1:0]	CTME	ERRD	BOM[1:0]	—	—	—	—	—	—	TAIE
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ALIE	BLIE	OLIE	BORIE	BOEIE	EPIE	EWIE	BEIE	—	—	—	—	RTBO	CSLPR	CHMDC[1:0]	
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

**Table 21.16** RSCAN0CmCTR register contents (1/2)

Bit Position	Bit Name	Function
31 to 27	Reserved	These bits are always read as 0. The write value should always be 0.
26, 25	CTMS[1:0]	Communication Test Mode Select b <sub>26</sub> b <sub>25</sub> 0 0: Standard test mode 0 1: Listen-only mode 1 0: Self-test mode 0 (external loopback mode) 1 1: Self-test mode 1 (internal loopback mode)
24	CTME	Communication Test Mode Enable 0: Communication test mode is disabled. 1: Communication test mode is enabled.
23	ERRD	Error Display Mode Select 0: Error flags are displayed only for the first error information after bits 14 to 8 in RSCAN0CmERFL are all cleared. 1: Error flags for all error information are displayed.
22, 21	BOM[1:0]	Bus Off Recovery Mode Select b <sub>22</sub> b <sub>21</sub> 0 0: ISO11898-1 compliant 0 1: Entry to channel halt mode automatically at bus-off entry 1 0: Entry to channel halt mode automatically at bus-off end 1 1: Entry to channel halt mode (during bus-off recovery period) by program request
20 to 17	Reserved	These bits are always read as 0. The write value should always be 0.
16	TAIE	Transmit Abort Interrupt Enable 0: Transmit abort interrupt is disabled. 1: Transmit abort interrupt is enabled.
15	ALIE	Arbitration Lost Interrupt Enable 0: Arbitration lost interrupt is disabled. 1: Arbitration lost interrupt is enabled.
14	BLIE	Bus Lock Interrupt Enable 0: Bus lock interrupt is disabled. 1: Bus lock interrupt is enabled.
13	OLIE	Overload Frame Transmit Interrupt Enable 0: Overload frame transmit interrupt is disabled. 1: Overload frame transmit interrupt is enabled.
12	BORIE	Bus Off Recovery Interrupt Enable 0: Bus off recovery interrupt is disabled. 1: Bus off recovery interrupt is enabled.

Table 21.16 RSCAN0CmCTR register contents (2/2)

Bit Position	Bit Name	Function
11	BOEIE	Bus Off Entry Interrupt Enable 0: Bus off entry interrupt is disabled. 1: Bus off entry interrupt is enabled.
10	EPIE	Error Passive Interrupt Enable 0: Error passive interrupt is disabled. 1: Error passive interrupt is enabled.
9	EWIE	Error Warning Interrupt Enable 0: Error warning interrupt is disabled. 1: Error warning interrupt is enabled.
8	BEIE	Bus Error Interrupt Enable 0: Bus error interrupt is disabled. 1: Bus error interrupt is enabled.
7 to 4	Reserved	These bits are always read as 0. The write value should always be 0.
3	RTBO	Forcible Return from Bus-off When this bit is set to 1, forcible return from the bus off state is made. This bit is always read as 0.
2	CSLPR	Channel Stop Mode 0: Other than channel stop mode 1: Channel stop mode
1, 0	CHMDC[1:0]	Mode Select b1 b0 0 0: Channel communication mode 0 1: Channel reset mode 1 0: Channel halt mode 1 1: Setting prohibited

**CTMS[1:0] Bits**

These bits are used to select a communication test mode. Modify these bits in channel halt mode only. These bits are set to 0 in channel reset mode.

**CTME Bit**

Setting this bit to 1 enables communication test mode. Modify these bits in channel halt mode. This bit is set to 0 in channel reset mode.

**ERRD Bit**

This bit is used to control the display mode of bits 14 to 8 in the RSCAN0CmERFL register. When this bit is clear to 0, only the flags of the first error are set to 1. If two or more errors occur in the first error, all the flags of the detected errors are set to 1.

When this bit is set to 1, all the flags of errors that have occurred are set to 1 regardless of the error occurrence order.

Modify this bit only in channel reset mode or channel halt mode.

### **BOM[1:0] Bits**

These bits are used to select the bus off recovery mode of the RS-CAN module.

When the BOM[1:0] bits are set to 00<sub>B</sub>, return from the bus off state to the error active state is compliant with the CAN specifications. That is, the RS-CAN module reenters the CAN communication (error active state) after 11 consecutive recessive bits are detected 128 times. A bus off recovery interrupt request is generated at the time of return from the bus off state. Even if the CHMDC[1:0] bits are set to 10<sub>B</sub> (channel halt mode) before recessive bits are detected 128 times, the RS-CAN module does not transition to channel halt mode until recessive bits are detected 128 times.

When the RS-CAN module reaches the bus off state when the BOM[1:0] bits are set to 01<sub>B</sub>, the CHMDC[1:0] bits in the RSCAN0CmCTR register (m = 0 to 4) are set to 10<sub>B</sub> and the RS-CAN module transitions to channel halt mode. No bus off recovery interrupt request is generated at the time of return from the bus off state and the TEC[7:0] and REC[7:0] bits in the RSCAN0CmSTS register are cleared to H'00.

When the RS-CAN module reaches the bus off state when the BOM[1:0] bits are set to 10<sub>B</sub>, the CHMDC[1:0] bits are set to 10<sub>B</sub> and the RS-CAN module transitions to channel halt mode after return from the bus off state (11 consecutive recessive bits are detected 128 times). A bus off recovery interrupt request is generated at the time of return from the bus off state and the TEC[7:0] and REC[7:0] bits are cleared to 00<sub>H</sub>.

When the BOM[1:0] bits are set to 11<sub>B</sub> and the CHMDC[1:0] bits are set to 10<sub>B</sub> while the RS-CAN module is in the bus off state, the RS-CAN module transitions to channel halt mode. No bus off recovery interrupt request is generated at the time of return from the bus off state and the TEC[7:0] and REC[7:0] bits are cleared to 00<sub>H</sub>. However, if 11 consecutive recessive bits are detected 128 times and the RS-CAN module has recovered to the error active state from the bus off state before the CHMDC[1:0] bits are set to 10<sub>B</sub>, a bus off recovery interrupt request is generated.

If the CPU requests a transition to channel reset mode at the same time as the RS-CAN module transition to channel halt mode (at bus off entry when the BOM[1:0] bits are 01<sub>B</sub> or at bus off end when the BOM[1:0] bits are 10<sub>B</sub>), the CPU's request takes precedence. Modify the BOM bits only in channel reset mode.

### **TAIE Bit**

When transmit abort of the transmit buffer is completed with the TAIE bit set to 1, an interrupt request is generated. Modify this bit only in channel reset mode.

### **ALIE Bit**

When the ALF flag in the RSCAN0CmERFL register is set to 1 with the ALIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

### **BLIE Bit**

When the BLF flag in the RSCAN0CmERFL register is set to 1 with the BLIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

### **OLIE Bit**

When the OLF flag in the RSCAN0CmERFL register is set to 1 with the OLIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.



**BORIE Bit**

When the BORF flag in the RSCAN0CmERFL register is set to 1 with the BORIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

**BOEIE Bit**

When the BOEF flag in the RSCAN0CmERFL register is set to 1 with the BOEIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

**EPIE Bit**

When the EPF flag in the RSCAN0CmERFL register is set to 1 with the EPIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

**EWIE Bit**

When the EWF flag in the RSCAN0CmERFL register is set to 1 with the EWIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

**BEIE Bit**

When the BEF flag in the RSCAN0CmERFL register is set to 1 with the BEIE bit set to 1, an error interrupt request is generated. Modify this bit only in channel reset mode.

**RTBO Bit**

Setting this bit to 1 in the bus off state forcibly returns the state from the bus off state to the error active state. This bit is automatically cleared to 0. Setting this bit to 1 clears the TEC[7:0] and REC[7:0] bits in the RSCAN0CmSTS register to 00<sub>H</sub> and also clears the BOSTS flag in the RSCAN0CmSTS register to 0 (not in bus off state). The other registers remain unchanged. No bus off recovery interrupt request is generated upon return from the bus off state in this case. Use this bit only when the BOM[1:0] bits in the RSCAN0CmCTR register are 00<sub>B</sub> (ISO11898-1 compliant).

A delay of up to 1 CAN bit time occurs after the RTBO bit is set to 1 until the RSCAN module transitions to the error active state. Set this bit to 1 in channel communication mode.

**CSLPR Bit**

Setting this bit to 1 places the channel into channel stop mode.

Clearing this bit to 0 makes the channel exit channel stop mode.

Modify this bit from 0 to 1 only in channel reset mode.

**CHMDC[1:0] Bits**

These bits are used to select a channel mode (channel communication mode, channel reset mode, or channel halt mode). For details, see **Section 21.5.2, Channel Modes**. Setting the CSLPR bit to 1 in channel reset mode allows transition to channel stop mode. Do not set the CHMDC[1:0] bits to 11<sub>B</sub>. When the CAN module has automatically transitioned to channel halt mode based on the setting of the BOM[1:0] bits, the CHMDC[1:0] bits automatically become 10<sub>B</sub>.

### 21.3.3 RSCAN0CmSTS — Channel Status Register (m = 0 to 4)

**Access:** Can be read in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0008<sub>H</sub> + (m \* 0010<sub>H</sub>)

**Initial value:** 0000 0005<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TEC[7:0]								REC[7:0]							
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	COMST S	RECST S	TRMST S	BOSTS	EPSTS	CSLPST S	CHLTST S	CRSTS TS
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

**Table 21.17 RSCAN0CmSTS register contents**

Bit Position	Bit Name	Function
31 to 24	TEC[7:0]	The transmit error counter (TEC) can be read.
23 to 16	REC[7:0]	The receive error counter (REC) can be read.
15 to 8	Reserved	These bits are always read as 0.
7	COMSTS	Communication Status Flag 0: Communication is not ready. 1: Communication is ready.
6	RECSTS	Receive Status Flag 0: Bus idle, in transmission or bus off state 1: In reception
5	TRMSTS	Transmit Status Flag 0: Bus idle or in reception 1: In transmission or bus off state
4	BOSTS	Bus Off Status Flag 0: Not in bus off state 1: In bus off state
3	EPSTS	Error Passive Status Flag 0: Not in error passive state 1: In error passive state
2	CSLPSTS	Channel Stop Status Flag 0: Not in channel stop mode 1: In channel stop mode
1	CHLTSTS	Channel Halt Status Flag 0: Not in channel halt mode 1: In channel halt mode
0	CRSTSTS	Channel Reset Status Flag 0: Not in channel reset mode 1: In channel reset mode

#### TEC[7:0] Bits

These bits contain the transmit error counter value. For transmit error counter increment/decrement conditions, see the CAN specification (ISO11898-1).

These bits are cleared to 0 in channel reset mode.

**REC[7:0] Bits**

These bits contain the receive error counter value. For receive error counter increment/decrement conditions, see the CAN specifications (ISO11898-1).

These bits are cleared to 0 in channel reset mode.

**COMSTS Flag**

This bit indicates that communication is ready.

This flag becomes 1 when the CAN module has detected 11 consecutive recessive bits after it has transitioned from channel reset mode or channel halt mode to channel communication mode. This flag is cleared to 0 in channel reset mode or channel halt mode.

**RECSTS Flag**

This flag is set to 1 when reception has started, and is cleared to 0 when the bus has become idle or transmission has started.

**TRMSTS Flag**

This flag is set to 1 when transmission has started, and is cleared to 0 when the bus has become idle or reception has started. This flag remains 1 in the bus off state.

**BOSTS Flag**

This flag is set to 1 when the bus off state ( $TEC[7:0] > 255$ ) is entered. It is cleared to 0 when the CAN module has exited the bus off state.

**EPSTS Flag**

This flag is set to 1 when the RS-CAN module has entered the error passive state ( $(128 \leq TEC[7:0] \leq 255)$  or  $(128 \leq REC[7:0])$ ), It is cleared to 0 when the RS-CAN module has exited the error passive state or has entered channel reset mode.

**CSLPSTS Flag**

This flag is set to 1 when the CAN module has transitioned to channel stop mode, and is cleared to 0 when the CAN module has returned from channel stop mode.

**CHLTSTS Flag**

This flag is set to 1 when the CAN module has transitioned to channel halt mode, and is cleared to 0 when the CAN module has returned from channel halt mode.

**CRSTSTS Flag**

This flag is set to 1 when the CAN module has transitioned to channel reset mode, and is cleared to 0 when the CAN module has transitioned to channel communication mode or channel halt mode. This flag remains 1 when the CAN module transitions from channel reset mode to channel stop mode.

### 21.3.4 RSCAN0CmERFL — Channel Error Flag Register (m = 0 to 4)

**Access:** Can be read/written in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 000C<sub>H</sub> + (m \* 0010<sub>H</sub>)

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRCREG[14:0]															
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	ADERR	B0ERR	B1ERR	CERR	AERR	FERR	SERR	ALF	BLF	OVL	BORF	BOEF	EPF	EW	BEF
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

**Table 21.18 RSCAN0CmERFL register contents (1/2)**

Bit Position	Bit Name	Function
31	Reserved	This bit is always read as 0. The write value should always be 0.
30 to 16	CRCREG[14:0]	CRC Calculation Data A CRC value calculated based on the transmit message or receive message is indicated.
15	Reserved	This bit is always read as 0. The write value should always be 0.
14	ADERR	ACK Delimiter Error Flag 0: No ACK delimiter error is detected. 1: ACK delimiter error is detected.
13	B0ERR	Dominant Bit Error Flag 0: No dominant bit error is detected. 1: Dominant bit error is detected.
12	B1ERR	Recessive Bit Error Flag 0: No recessive bit error is detected. 1: Recessive bit error is detected.
11	CERR	CRC Error Flag 0: No CRC error is detected. 1: CRC error is detected.
10	AERR	ACK Error Flag 0: No ACK error is detected. 1: ACK error is detected.
9	FERR	Form Error Flag 0: No form error is detected. 1: Form error is detected.
8	SERR	Stuff Error Flag 0: No stuff error is detected. 1: Stuff error is detected.
7	ALF	Arbitration-lost Flag 0: No arbitration-lost is detected. 1: Arbitration-lost is detected.
6	BLF	Dominant Lock Flag 0: No dominant lock is detected. 1: Dominant lock is detected.

**Table 21.18 RSCAN0CmERFL register contents (2/2)**

Bit Position	Bit Name	Function
5	OVLf	Overload Flag 0: No overload is detected. 1: Overload is detected.
4	BORf	Bus Off Recovery Flag 0: No bus off recovery is detected. 1: Bus off recovery is detected.
3	BOEF	Bus Off Entry Flag 0: No bus off entry is detected. 1: Bus off entry is detected.
2	EPF	Error Passive Flag 0: No error passive is detected. 1: Error passive is detected.
1	EWf	Error Warning Flag 0: No error warning is detected. 1: Error warning is detected.
0	BEF	Protocol Error Flag 0: No protocol error is detected. 1: Protocol error is detected.

See the CAN specification (ISO11898-1) for a description of error occurrence conditions. To clear each flag of this register, the program must write a 0 to the corresponding bit. These flags cannot be set to 1 by the program. If any of these error occurs at the same time that the program writes 0 to the flag, the flag is still set to 1. The channel reset mode transition clears all of these flags to 0. If the ERRD bit in the RSCAN0CmCTR register is set to 0 (ie, only the flags from the first error event are displayed) and an error related to bits 14 to 8 of RSCAN0CmERFL is detected, the flag bits are only set by the error event if bits 14 to 8 were all 0 at the time the error occurred.

### **CRCREG[14:0] Bits**

When the CTME bit in the RSCAN0CmCTR register is set to 1 (communication test mode is enabled), the CRC value calculated based on the transmit or receive message can be read. When the CTME bit is set to 0 (communication test mode is disabled), these bits are always read as 0.

### **ADERR Flag**

This flag is set to 1 when a form error has been detected in the ACK delimiter during transmission.

### **B0ERR Flag**

This flag is set to 1 when a recessive bit has been detected though a dominant bit was transmitted.

### **B1ERR Flag**

This flag is set to 1 when a dominant bit has been detected though a recessive bit was transmitted.

### **CERR Flag**

This flag is set to 1 when a CRC error has been detected.

### **AERR Flag**

This flag is set to 1 when an ACK error has been detected.

**FERR Flag**

This flag is set to 1 when a form error has been detected.

**SERR Flag**

This flag is set to 1 when a stuff error has been detected.

**ALF Flag**

This flag is set to 1 when an arbitration-lost has been detected.

**BLF Flag**

This flag is set to 1 when 32 consecutive dominant bits have been detected on the CAN bus in channel communication mode. After that, dominant lock can be detected again if any of the following conditions is met.

- a recessive bit is detected after the BLF bit has been cleared from 1 to 0.
- the CAN module transitions to channel reset mode and returns to channel communication mode after the BLF bit has been cleared from 1 to 0.

**OVLf Flag**

This flag is set to 1 when the overload frame transmit condition has been detected when performing reception or transmission.

**BORF Flag**

This flag is set to 1 when 11 consecutive recessive bits have been detected 128 times and the CAN module returns from the bus off state. However, this flag is not set to 1 if the CAN module returns from the bus off state in any of the following ways before 11 consecutive recessive bits are detected 128 times.

- The CHMDC[1:0] bits in the RSCAN0CmCTR register are set to 01<sub>B</sub> (channel reset mode).
- The RTBO bit in the RSCAN0CmCTR register is set to 1 (forcible return from the bus off state is made).
- The BOM[1:0] bits in the RSCAN0CmCTR register are set to 01<sub>B</sub> (transition to channel halt mode at bus off entry).
- The CHMDC[1:0] bits in the RSCAN0CmCTR register are set to 10<sub>B</sub> (channel halt mode) before 11 consecutive recessive bits are detected 128 times with the BOM[1:0] bits set to 11<sub>B</sub> (transition to channel halt mode upon a request from the program during bus off).

**BOEF Flag**

This flag is set to 1 when the bus off state is reached (TEC[7:0] value > 255). This flag is also set to 1 if the bus off state is reached when the BOM[1:0] bits in the RSCAN0CmCTR register (m = 0 to 4) set to 01<sub>B</sub> (transition to channel halt mode at bus off entry).

**EPF Flag**

This flag becomes 1 when the error passive state is reached (REC[7:0] or TEC[7:0] value > 127).

This flag becomes 1 only when the REC[7:0] or TEC[7:0] value first exceeds 127. Therefore, if the program writes 0 to this flag while the value of REC[7:0] or TEC[7:0] remains over 127, this bit is not set to 1 until both REC [7:0] and TEC[7:0] values become 127 or less and then the REC[7:0] or TEC[7:0] value exceeds 127 again.

**EWF Flag**

This flag is set to 1 only when the REC[7:0] or TEC[7:0] value first exceeds 95. Therefore, if the program writes 0 to this flag while the value of REC[7:0] or TEC[7:0] remains over 95, this bit is not set to 1 until both REC [7:0] and TEC[7:0] values become 95 or less and then the REC[7:0] or TEC[7:0] value exceeds 95 again.

**BEF Flag**

This flag is set to 1 when any one of the ADERR, B0ERR, B1ERR, CERR, AERR, FERR, and SERR flags in the RSCAN0CmERFL register is set to 1.

### 21.3.5 RSCAN0GCFG — Global Configuration Register

**Access:** Can be read/written in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0084<sub>H</sub>

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ITRCP															
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSBTCS[2:0]			TSSS	TSP[3:0]				—	—	—	DCS	MME	DRE	DCE	TPRI
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

**Table 21.19 RSCAN0GCFG register contents (1/2)**

Bit Position	Bit Name	Function																																																																																					
31 to 16	ITRCP[15:0]	Interval Timer Prescaler Set When these bits are set to M, the pclk is divided by M. Setting 0000 <sub>H</sub> is prohibited when the interval timer is in use.																																																																																					
15 to 13	TSBTCS[2:0]	Timestamp Clock Source Select <table border="0"> <tr><td>b15</td><td>b14</td><td>b13</td><td></td></tr> <tr><td>0</td><td>0</td><td>0</td><td>Channel 0 bit time clock</td></tr> <tr><td>0</td><td>0</td><td>1</td><td>Channel 1 bit time clock</td></tr> <tr><td>0</td><td>1</td><td>0</td><td>Channel 2 bit time clock</td></tr> <tr><td>0</td><td>1</td><td>1</td><td>Channel 3 bit time clock</td></tr> <tr><td>1</td><td>0</td><td>0</td><td>Channel 4 bit time clock</td></tr> <tr><td>1</td><td>0</td><td>1</td><td>Setting prohibited</td></tr> <tr><td>1</td><td>1</td><td>0</td><td>Setting prohibited</td></tr> <tr><td>1</td><td>1</td><td>1</td><td>Setting prohibited</td></tr> </table>	b15	b14	b13		0	0	0	Channel 0 bit time clock	0	0	1	Channel 1 bit time clock	0	1	0	Channel 2 bit time clock	0	1	1	Channel 3 bit time clock	1	0	0	Channel 4 bit time clock	1	0	1	Setting prohibited	1	1	0	Setting prohibited	1	1	1	Setting prohibited																																																	
b15	b14	b13																																																																																					
0	0	0	Channel 0 bit time clock																																																																																				
0	0	1	Channel 1 bit time clock																																																																																				
0	1	0	Channel 2 bit time clock																																																																																				
0	1	1	Channel 3 bit time clock																																																																																				
1	0	0	Channel 4 bit time clock																																																																																				
1	0	1	Setting prohibited																																																																																				
1	1	0	Setting prohibited																																																																																				
1	1	1	Setting prohibited																																																																																				
12	TSSS	Timestamp Source Select 0: pclk/2*1 1: Bit time clock																																																																																					
11 to 8	TSP[3:0]	Timestamp Clock Source Division <table border="0"> <tr><td>b11</td><td>b10</td><td>b9</td><td>b8</td><td></td></tr> <tr><td>0</td><td>0</td><td>0</td><td>0</td><td>Not divided</td></tr> <tr><td>0</td><td>0</td><td>0</td><td>1</td><td>Divided by 2</td></tr> <tr><td>0</td><td>0</td><td>1</td><td>0</td><td>Divided by 4</td></tr> <tr><td>0</td><td>0</td><td>1</td><td>1</td><td>Divided by 8</td></tr> <tr><td>0</td><td>1</td><td>0</td><td>0</td><td>Divided by 16</td></tr> <tr><td>0</td><td>1</td><td>0</td><td>1</td><td>Divided by 32</td></tr> <tr><td>0</td><td>1</td><td>1</td><td>0</td><td>Divided by 64</td></tr> <tr><td>0</td><td>1</td><td>1</td><td>1</td><td>Divided by 128</td></tr> <tr><td>1</td><td>0</td><td>0</td><td>0</td><td>Divided by 256</td></tr> <tr><td>1</td><td>0</td><td>0</td><td>1</td><td>Divided by 512</td></tr> <tr><td>1</td><td>0</td><td>1</td><td>0</td><td>Divided by 1024</td></tr> <tr><td>1</td><td>0</td><td>1</td><td>1</td><td>Divided by 2048</td></tr> <tr><td>1</td><td>1</td><td>0</td><td>0</td><td>Divided by 4096</td></tr> <tr><td>1</td><td>1</td><td>0</td><td>1</td><td>Divided by 8192</td></tr> <tr><td>1</td><td>1</td><td>1</td><td>0</td><td>Divided by 16384</td></tr> <tr><td>1</td><td>1</td><td>1</td><td>1</td><td>Divided by 32768</td></tr> </table>	b11	b10	b9	b8		0	0	0	0	Not divided	0	0	0	1	Divided by 2	0	0	1	0	Divided by 4	0	0	1	1	Divided by 8	0	1	0	0	Divided by 16	0	1	0	1	Divided by 32	0	1	1	0	Divided by 64	0	1	1	1	Divided by 128	1	0	0	0	Divided by 256	1	0	0	1	Divided by 512	1	0	1	0	Divided by 1024	1	0	1	1	Divided by 2048	1	1	0	0	Divided by 4096	1	1	0	1	Divided by 8192	1	1	1	0	Divided by 16384	1	1	1	1	Divided by 32768
b11	b10	b9	b8																																																																																				
0	0	0	0	Not divided																																																																																			
0	0	0	1	Divided by 2																																																																																			
0	0	1	0	Divided by 4																																																																																			
0	0	1	1	Divided by 8																																																																																			
0	1	0	0	Divided by 16																																																																																			
0	1	0	1	Divided by 32																																																																																			
0	1	1	0	Divided by 64																																																																																			
0	1	1	1	Divided by 128																																																																																			
1	0	0	0	Divided by 256																																																																																			
1	0	0	1	Divided by 512																																																																																			
1	0	1	0	Divided by 1024																																																																																			
1	0	1	1	Divided by 2048																																																																																			
1	1	0	0	Divided by 4096																																																																																			
1	1	0	1	Divided by 8192																																																																																			
1	1	1	0	Divided by 16384																																																																																			
1	1	1	1	Divided by 32768																																																																																			
7 to 5	Reserved	These bits are always read as 0. The write value should always be 0.																																																																																					
4	DCS	CAN Clock Source Select*2 0: clk 1: clk_xincan																																																																																					



**Table 21.19 RSCAN0GCFG register contents (2/2)**

Bit Position	Bit Name	Function
3	MME	Mirror Function Enable 0: Mirror function is disabled. 1: Mirror function is enabled.
2	DRE	DLC Replacement Enable 0: DLC replacement is disabled. 1: DLC replacement is enabled.
1	DCE	DLC Check Enable 0: DLC check is disabled. 1: DLC check is enabled.
0	TPRI	Transmit Priority Select 0: ID priority 1: Transmit buffer number priority

Note 1. When specifying CKSCLK\_ICANOSC as the timestamp counter count source, set bits TSBTCS[2:0] to 000<sub>B</sub>.

Note 2. For the setting of the CAN clock frequency, see **Table 21.6, Range of Operating Frequency Depending on the Transfer Rate and the Number of Channels in Use in this LSI.**

Modify the RSCAN0GCFG register only in global reset mode.

### ITRCP[15:0] Bits

These bits are used to set a clock source division value of the interval timer for FIFO buffers. Refer to **Section 21.7.3.1, Interval Transmission Function.**

### TSBTCS[2:0] Bits

When the TSSS bit is 1, these bits are used to select the channel of the bit time clock that will be the clock source of the timestamp counter.

### TSSS Bit

This bit is used to select a clock source of the timestamp counter.

### TSP[3:0] Bits

A clock obtained by dividing the clock source selected with the TSBTCS[2:0] bits and TSSS bit according to the TSP[3:0] bits is used as the timestamp counter count source.

### DCS Bit

When this bit is set to 0, clk<sub>c</sub> is used as the clock source of the CAN clock ( $f_{CAN}$ ).

When this bit is set to 1, clk\_xincan is used as the clock source of the CAN clock ( $f_{CAN}$ ).

For the setting of the CAN clock frequency, see **Table 21.6, Range of Operating Frequency Depending on the Transfer Rate and the Number of Channels in Use in this LSI.**

### MME Bit

Setting this bit to 1 makes the mirror function available.

**DRE Bit**

When the DRE bit is set to 1, the DLC value of the receive rule is stored in the buffer instead of the DLC value of the received message after the DLC value has passed through the DLC filter. In this case, a value of 00<sub>H</sub> is stored in each data byte beyond the first n bytes, where n is the DLC value of the receive rule.

The DLC replacement function is only available when the DCE bit is set to 1 (DLC check is enabled).

**DCE Bit**

Setting this bit to 1 makes the DLC check function available. When disabling the DLC check function, set the GAFLDLC[3:0] bits in the RSCAN0GAFLP0j register to 0000<sub>B</sub> before clearing the DCE bit in the RSCAN0GCFG register to 0.

**TPRI Bit**

This bit is used to set the transmit priority.

When this bit is set to 0, ID priority is selected and the transmit priority complies with the CAN bus arbitration rule (ISO11898-1 specifications). When this bit is set to 1, transmit buffer number priority is selected and the lowest transmit buffer number of those with pending messages has the highest priority.

While the transmit queue is in use, this bit should be set to 0.

### 21.3.6 RSCAN0GCTR — Global Control Register

**Access:** Can be read/written in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0088<sub>H</sub>

**Initial value:** 0000 0005<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TSRST
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	THLEIE	MEIE	DEIE	—	—	—	—	—	GSLPR	GMDC[1:0]	
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0 1
R/W	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

**Table 21.20 RSCAN0GCTR register contents**

Bit Position	Bit Name	Function
31 to 17	Reserved	These bits are always read as 0. The write value should always be 0.
16	TSRST	Timestamp Counter Reset Setting the TSRST bit to 1 resets the timestamp counter. This bit is always read as 0.
15 to 11	Reserved	These bits are always read as 0. The write value should always be 0.
10	THLEIE	Transmit History Buffer Overflow Interrupt Enable 0: Transmit history buffer overflow interrupt is disabled. 1: Transmit history buffer overflow interrupt is enabled.
9	MEIE	FIFO Message Lost Interrupt Enable 0: FIFO message lost interrupt is disabled. 1: FIFO message lost interrupt is enabled.
8	DEIE	DLC Error Interrupt Enable 0: DLC error interrupt is disabled. 1: DLC error interrupt is enabled.
7 to 3	Reserved	These bits are always read as 0. The write value should always be 0.
2	GSLPR	Global Stop Mode 0: Other than global stop mode 1: Global stop mode
1, 0	GMDC[1:0]	Global Mode Select b1 b0 0 0: Global operating mode 0 1: Global reset mode 1 0: Global test mode 1 1: Setting prohibited

#### TSRST Bit

This bit is used to reset the timestamp counter. When this bit is set to 1, the RSCAN0GTSC register is cleared to 0000<sub>H</sub>.

#### THLEIE Bit

When the THLEIE bit is set to 1 and the THLES flag in the RSCAN0GERFL register is set to 1, an interrupt request is generated. Modify this bit only in global reset mode.

**MEIE Bit**

When the MEIE bit is set to 1 and the MES flag in the RSCAN0GERFL register is set to 1, an interrupt request is generated. Modify this bit only in global reset mode.

**DEIE Bit**

When the DEIE bit is set to 1 and the DEF flag in the RSCAN0GERFL register is set to 1, an interrupt request is generated. Modify this bit only in global reset mode.

**GSLPR Bit**

When the RSCAN module is in global reset mode, setting this bit to 1 places the RSCAN module into global stop mode.

Clearing this bit to 0 makes the RSCAN module leave from global stop mode.

Modify this bit only in global reset mode.

**GMDC[1:0] Bits**

These bits are used to select the mode of entire RS-CAN module (global operating mode, global reset mode, or global test mode). For details, see **Section 21.5.1, Global Modes**. Setting the GSLPR bit to 1 when in global reset mode places the RS-CAN module into global stop mode.

### 21.3.7 RSCAN0GSTS — Global Status Register

**Access:** Can be read in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 008C<sub>H</sub>

**Initial value:** 0000 000D<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	GRAMI NIT	GSLPS TS	GHLTST S	GRSTS TS
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

**Table 21.21 RSCAN0GSTS register contents**

Bit Position	Bit Name	Function
31 to 4	Reserved	These bits are always read as 0.
3	GRAMINIT	CAN RAM Initialisation Status Flag 0: CAN RAM initialization is completed. 1: CAN RAM initialization is ongoing.
2	GSLPSTS	Global Stop Status Flag 0: Not in global stop mode 1: In global stop mode
1	GHLTSTS	Global Test Status Flag 0: Not in global test mode 1: In global test mode
0	GRSTSTS	Global Reset Status Flag 0: Not in global reset mode 1: In global reset mode

#### GRAMINIT Flag

This flag indicates the initialization status of the CAN RAM.

This flag is set to 1 after this LSI has been reset, and is cleared to 0 when CAN RAM initialisation is completed.

#### GSLPSTS Flag

This flag is set to 1 when the CAN module has transitioned to global stop mode, and is cleared to 0 when the CAN module has returned from global stop mode.

#### GHLTSTS Flag

This flag is set to 1 when the CAN module has transitioned to global test mode, and is cleared to 0 when the CAN module has exited global test mode.

**GRSTSTS Flag**

This flag is set to 1 when the CAN module has transitioned to global reset mode, and is cleared to 0 when the CAN module has exited global reset mode. This flag remains 1 even when the CAN module has transitioned from global reset mode to global stop mode.

### 21.3.8 RSCAN0GERFL — Global Error Flag Register

**Access:** Can be read in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0090<sub>H</sub>

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	THLES	MES	DEF
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W*1

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

**Table 21.22 RSCAN0GERFL register contents**

Bit Position	Bit Name	Function
31 to 29	Reserved	The read value is undefined. The write value should always be 0.
2	THLES	Transmit History Buffer Overflow Status Flag 0: No transmit history buffer overflow has occurred. 1: A transmit history buffer overflow has occurred.
1	MES	FIFO Message Lost Status Flag 0: No FIFO message lost error has occurred. 1: A FIFO message lost error has occurred.
0	DEF	DLC Error Flag 0: No DLC error has occurred. 1: A DLC error has occurred.

All flags in the RSCAN0GERFL register are cleared to 0 in global reset mode.

#### THLES Flag

The THLES flag is set to 1 when any one of the THLELT flags in the RSCAN0THLSTSm register (m = 0 to 4) is set to 1.

This flag is cleared to 0 when the THLELT flags of all channels are set to 0.

#### MES Flag

The MES flag is set to 1 when any one of the RFMLT flags in the RSCAN0RFSTSk register (x = 0 to 7) or the CFMLT flags in the RSCAN0CFSTSk register (k = 0 to 14) is set to 1.

This flag is cleared to 0 when all RFMLT flags and CFMLT flags are set to 0.

#### DEF Flag

The DEF flag is set to 1 when an error has been detected during the DLC check. The program can clear this flag by writing 0 to this bit.

### 21.3.9 RSCAN0GTINTSTS0 — Global TX Interrupt Status Register 0

**Access:** Can be read in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0460<sub>H</sub>

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	THIF3	CFTIF3	TQIF3	TAIF3	TSIF3	—	—	—	THIF2	CFTIF2	TQIF2	TAIF2	TSIF2
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R*1	R*1	R*1	R*1	R*1	R	R	R	R*1	R*1	R*1	R*1	R*1
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	THIF1	CFTIF1	TQIF1	TAIF1	TSIF1	—	—	—	THIF0	CFTIF0	TQIF0	TAIF0	TSIF0
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R*1	R*1	R*1	R*1	R*1	R	R	R	R*1	R*1	R*1	R*1	R*1

Note 1. This bit is automatically cleared in the global reset or channel reset mode.

**Table 21.23 RSCAN0GTINTSTS0 register contents (1/2)**

Bit Position	Bit Name	Function
31 to 29	Reserved	These bits are always read as 0. The write value should always be 0.
28	THIF3	Channel 3 transmit history interrupt status flag 0: Transmit history interrupt is not requested. 1: Transmit history interrupt is requested.
27	CFTIF3	Channel 3 transmit/receive FIFO transmit/gateway mode interrupt status flag 0: Transmit/receive FIFO transmit/gateway mode interrupt is not requested. 1: Transmit/receive FIFO transmit/gateway mode interrupt is requested.
26	TQIF3	Channel 3 transmit queue interrupt status flag 0: Transmit queue interrupt is not requested. 1: Transmit queue interrupt is requested.
25	TAIF3	Channel 3 transmit buffer abort interrupt status flag 0: Transmit buffer abort interrupt is not requested. 1: Transmit buffer abort interrupt is requested.
24	TSIF3	Channel 3 transmit buffer transmit complete interrupt status flag 0: Transmit buffer transmit complete interrupt is not requested. 1: Transmit buffer transmit complete interrupt is requested.
23 to 21	Reserved	These bits are always read as 0. The write value should always be 0.
20	THIF2	Channel 2 transmit history interrupt status flag 0: Transmit history interrupt is not requested. 1: Transmit history interrupt is requested.
19	CFTIF2	Channel 2 transmit/receive FIFO transmit/gateway mode interrupt status flag 0: Transmit/receive FIFO transmit/gateway mode interrupt is not requested. 1: Transmit/receive FIFO transmit/gateway mode interrupt is requested.
18	TQIF2	Channel 2 transmit queue interrupt status flag 0: Transmit queue interrupt is not requested. 1: Transmit queue interrupt is requested.
17	TAIF2	Channel 2 transmit buffer abort interrupt status flag 0: Transmit buffer abort interrupt is not requested. 1: Transmit buffer abort interrupt is requested.
16	TSIF2	Channel 2 transmit buffer interrupt status flag 0: Transmit buffer transmit complete interrupt is not requested. 1: Transmit buffer transmit complete interrupt is requested.
15 to 13	Reserved	These bits are always read as 0. The write value should always be 0.



Table 21.23 RSCAN0GTINTSTS0 register contents (2/2)

Bit Position	Bit Name	Function
12	THIF1	Channel 1 transmit history interrupt status flag 0: Transmit history interrupt is not requested. 1: Transmit history interrupt is requested.
11	CFTIF1	Channel 1 transmit/receive FIFO transmit/gateway mode interrupt status flag 0: Transmit/receive FIFO transmit/gateway mode interrupt is not requested. 1: Transmit/receive FIFO transmit/gateway mode interrupt is requested.
10	TQIF1	Channel 1 transmit queue interrupt status flag 0: Transmit queue interrupt is not requested. 1: Transmit queue interrupt is requested.
9	TAIF1	Channel 1 transmit buffer abort interrupt status flag 0: Transmit buffer transmit abort interrupt is not requested. 1: Transmit buffer transmit abort interrupt is requested.
8	TSIF1	Channel 1 transmit buffer interrupt status flag 0: Transmit buffer transmit complete interrupt is not requested. 1: Transmit buffer transmit complete interrupt is requested.
7 to 5	Reserved	These bits are always read as 0. The write value should always be 0.
4	THIF0	Channel 0 transmit history interrupt status flag 0: Transmit history interrupt is not requested. 1: Transmit history interrupt is requested.
3	CFTIF0	Channel 0 transmit/receive FIFO transmit/gateway mode interrupt status flag 0: Transmit/receive FIFO transmit/gateway mode interrupt is not requested. 1: Transmit/receive FIFO transmit/gateway mode interrupt is requested.
2	TQIF0	Channel 0 transmit queue interrupt status flag 0: Transmit queue interrupt is not requested. 1: Transmit queue interrupt is requested.
1	TAIF0	Channel 0 transmit buffer abort interrupt status flag 0: Transmit buffer transmit abort interrupt is not requested. 1: Transmit buffer transmit abort interrupt is requested.
0	TSIF0	Channel 0 transmit buffer interrupt status flag 0: Transmit buffer transmit complete interrupt is not requested. 1: Transmit buffer transmit complete interrupt is requested.

### TSIFm Bits

The TSIFm bit is set to 1 when the TMIE bit in the RSCAN0TMIECy register is set to 1 (transmit buffer interrupt enabled) and the TMTRF[1:0] flags in the RSCAN0TMSTSp register are set to 10<sub>B</sub> (transmit completed without abort request) or 11<sub>B</sub> (transmit completed with abort request).

When the TMTRF[1:0] flags are cleared to 00<sub>B</sub> under the condition that the TSIFm bit can be set to 1, this flag is cleared to 0. In addition, clearing the TMIE bit to 0 also clears this flag to 0.

### TAIFm Bits

The TAIFm bit is set to 1 when the TAIE bit in the RSCAN0CmCTR register is 1 (transmit abort interrupt enabled) and the TMTRF[1:0] flags in the RSCAN0TMSTSp register are set to 01<sub>B</sub> (transmit abort completed).

This flag is cleared to 0 when the TMTRF[1:0] flags are cleared to 00<sub>B</sub> after the transmit abort is completed.

**TQIFm Bits**

When the TXQIE bit in the RSCAN0TXQCCm register is set to 1 (transmit queue interrupt enabled) and the TXQIF bit in the RSCAN0TXQSTSm register is set to 1 (transmit queue interrupt request), the TQIFm bit is set to 1.

When the TXQIF bit (transmit queue interrupt request) in the RSCAN0TXQSTSm register is cleared to 0, this bit is cleared to 0. This flag is also cleared to 0 when the TXQIE bit is cleared to 0.

**CFTIFm Bits**

When the CFTXIE bit in the RSCAN0CFCCk register is set to 1 (transmit/receive FIFO buffer transmit interrupt enabled) and the CFTXIF bit in the RSCAN0CFSTSk register is set to 1 (transmit/receive FIFO transmit interrupt request), the CFTIFm bit is set to 1.

When the CFTXIF bit is cleared to 0 under the condition that the CFTIFm bit can be set to 1, this bit is cleared to 0. This flag is also cleared to 0 when the CFTXIE bit is cleared to 0.

**THIFm Bits**

When the THLIE bit in the RSCAN0THLCCm register is set to 1 (transmit history interrupt enabled) and the THLIF bit in the RSCAN0THLSTSm register is set to 1 (transmit history interrupt request), the THIFm bit is set to 1.

When the THLIF bit in the RSCAN0THLSTSm register is cleared to 0, this bit is cleared to 0. This flag is also cleared to 0 when the THLIE bit is cleared to 0.

### 21.3.10 RSCAN0GTINTSTS1 — Global TX Interrupt Status Register 1

**Access:** Can be read in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0464<sub>H</sub>

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	THIF4	CFTIF4	TQIF4	TAIF4	TSIF4
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Note 1. This bit is automatically cleared in the global reset or channel reset mode.

**Table 21.24 RSCAN0GTINTSTS1 register contents**

Bit Position	Bit Name	Function
31 to 5	Reserved	These bits are always read as 0. The write value should always be 0.
4	THIF4	Channel 4 transmit history interrupt status flag 0: Transmit history interrupt is not requested. 1: Transmit history interrupt is requested.
3	CFTIF4	Channel 4 transmit/receive FIFO transmit/gateway mode interrupt status flag 0: Transmit/receive FIFO transmit/gateway mode interrupt is not requested. 1: Transmit/receive FIFO transmit/gateway mode interrupt is requested.
2	TQIF4	Channel 4 transmit queue interrupt status flag 0: Transmit queue interrupt is not requested. 1: Transmit queue interrupt is requested.
1	TAIF4	Channel 4 transmit buffer abort interrupt status flag 0: Transmit buffer abort interrupt is not requested. 1: Transmit buffer abort interrupt is requested.
0	TSIF4	Channel 4 transmit buffer interrupt status flag 0: Transmit buffer transmit complete interrupt is not requested. 1: Transmit buffer transmit complete interrupt is requested.

#### TSIFm Bits

The TSIFm bit is set to 1 when the TMIE bit in the RSCAN0TMIECy register is set to 1 (transmit buffer interrupt enabled) and the TMTRF[1:0] flags in the RSCAN0TMSTSp register are set to 10<sub>B</sub> (transmit completed without abort request) or 11<sub>B</sub> (transmit completed with abort request).

When the TMTRF[1:0] flags are cleared to 00<sub>B</sub> under the condition that the TSIFm bit can be set to 1, this flag is cleared to 0. In addition, clearing the TMIE bit to 0 also clears this flag to 0.

#### TAIFm Bits

When the TAIE bit in the RSCAN0CmCTR register is set to 1 (transmit abort interrupt enabled) and the TMTRF[1:0] flags in the RSCAN0TMSTSp register are set to 01<sub>B</sub> (transmit abort completed), the TAIFm bit is set to 1.

This flag is cleared to 0 when the TMTRF[1:0] flags are cleared to 00<sub>B</sub> after the transmit abort is completed.

**TQIFm Bits**

When the TXQIE bit in the RSCAN0TXQCCm register is set to 1 (transmit queue interrupt enabled) and the TXQIF bit in the RSCAN0TXQSTSm register is set to 1 (transmit queue interrupt request), the TQIFm bit is set to 1.

When the TXQIF bit (transmit queue interrupt request) in the RSCAN0TXQSTSm register is cleared to 0, this bit is cleared to 0. Clearing the TXQIE bit to 0 also clears this flag to 0.

**CFTIFm Bits**

When the CFTXIE bit in the RSCAN0CFCCk register is set to 1 (transmit/receive FIFO buffer transmit interrupt enabled) and the CFTXIF bit in the RSCAN0CFSTSk register is set to 1 (transmit/receive FIFO transmit interrupt request), the CFTIFm bit is set to 1.

When the CFTXIF bit is cleared to 0 under the condition that the CFTIFm bit can be set to 1, this bit is cleared to 0. Clearing the CFTXIE bit to 0 also clears this flag to 0.

**THIFm Bits**

When the THLIE bit in the RSCAN0THLCCm register is set to 1 (transmit history interrupt enabled) and the THLIF bit in the RSCAN0THLSTSm register is set to 1 (transmit history interrupt request), the THIFm bit is set to 1.

When the THLIF bit in the RSCAN0THLSTSm register is cleared to 0, this bit is cleared to 0. Clearing the THLIE bit to 0 also clears this flag to 0.

### 21.3.11 RSCAN0GTSC — Global Timestamp Counter Register

**Access:** Can be read in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0094<sub>H</sub>

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TS[15:0]															
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

**Table 21.25 RSCAN0GTSC register contents**

Bit Position	Bit Name	Function
31 to 16	Reserved	These bits are always read as 0.
15 to 0	TS[15:0]	Timestamp Value The timestamp counter value can be read. Counter Value: 0000 <sub>H</sub> to FFFF <sub>H</sub>

#### TS[15:0] Bits

When the TS[15:0] bits are read, the read value shows the timestamp counter (16-bit free-running counter) value at that time. When the SOF is detected, the TS[15:0] value is captured and later stored in the receive buffer or the FIFO buffer. The timestamp counter is initialized in global reset mode.

The timestamp counter starts and stops counting differently, depending on the count source.

- When the TSSS bit in the RSCAN0GCFG register is 0 (pclk):  
The timestamp counter starts counting when the RSCAN module has transitioned to global operating mode.  
This counter stops counting when the RSCAN module has transitioned to global stop mode or global test mode.
- When the TSSS bit is 1 (CANm bit time clock):  
The timestamp counter starts counting when the corresponding channel has transitioned to channel communication mode.  
This counter stops counting when the corresponding channel has transitioned to channel reset mode or channel halt mode.

### 21.3.12 RSCAN0GAFLECTR — Receive Rule Entry Control Register

**Access:** Can be read/written in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0098<sub>H</sub>

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	AFLDAE	—	—	—	AFLPN[4:0]				
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

**Table 21.26 RSCAN0GAFLECTR register contents**

Bit Position	Bit Name	Function
31 to 9	Reserved	These bits are always read as 0. The write value should always be 0.
8	AFLDAE	Receive Rule Table Write Enable 0: Receive rule table write is disabled. 1: Receive rule table write is enabled.
7 to 5	Reserved	These bits are always read as 0. The write value should always be 0.
4 to 0	AFLPN[4:0]	Receive Rule Table Page Number Configuration A page number can be selected from a range of page 0 (00000 <sub>B</sub> ) to page 19 (10011 <sub>B</sub> ).

#### AFLDAE Bit

Setting this bit to 0 disables the write to the receive rule table. After writes to the receive rule table are completed, set this bit to 0 to disable the write to the table. The receive rule table can be read regardless of the value of this bit.

Set the AFLDAE bit to 1 only in global reset mode.

#### AFLPN[4:0] Bits

These bits are used to set the page number of the receive rule table. Sixteen receive rules can be set per page.

Set these bits to a value within the range of 00000<sub>B</sub> to 10011<sub>B</sub>.

### 21.3.13 RSCAN0GAFLCFG0 — Receive Rule Configuration Register 0

**Access:** Can be read/written in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 009C<sub>H</sub>

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RNC0[7:0]								RNC1[7:0]							
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RNC2[7:0]								RNC3[7:0]							
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

**Table 21.27 RSCAN0GAFLCFG0 register contents**

Bit Position	Bit Name	Function
31 to 24	RNC0[7:0]	Number of Rules for Channel 0 Set the number of receive rules exclusively used for channel 0.
23 to 16	RNC1[7:0]	Number of Rules for Channel 1 Set the number of receive rules exclusively used for channel 1.
15 to 8	RNC2[7:0]	Number of Rules for Channel 2 Set the number of receive rules exclusively used for channel 2.
7 to 0	RNC3[7:0]	Number of Rules for Channel 3 Set the number of receive rules exclusively used for channel 3.

Modify the RSCAN0GAFLCFG0 register only in global reset mode.

Up to 64 x (number of channels) rules can be registered in the receive rule table as the entire unit. The number of receive rules per channel should meet the following conditions.

- The maximum number of rules per channel is 128.
- The total of the number of rules allocated to each channel is not larger than the number of rules that can be registered in the entire unit.

#### RNC0[7:0] Bits

These bits are used to set the number of rules to be registered in the channel 0 receive rule table.

Set these bits to a value within the range of 00<sub>H</sub> to 80<sub>H</sub>.

#### RNC1[7:0] Bits

These bits are used to set the number of rules to be registered in the channel 1 receive rule table.

Set these bits to a value within the range of 00<sub>H</sub> to 80<sub>H</sub>.

**RNC2[7:0] Bits**

These bits are used to set the number of rules to be registered in the channel 2 receive rule table.

Set these bits to a value within the range of 00<sub>H</sub> to 80<sub>H</sub>.

**RNC3[7:0] Bits**

These bits are used to set the number of rules to be registered in the channel 3 receive rule table.

Set these bits to a value within the range of 00<sub>H</sub> to 80<sub>H</sub>.



### 21.3.14 RSCAN0GAFLCFG1 — Receive Rule Configuration Register 1

**Access:** Can be read/written in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 00A0<sub>H</sub>

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RNC4[7:0]								—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

**Table 21.28 RSCAN0GAFLCFG1 register contents**

Bit Position	Bit Name	Function
31 to 24	RNC4[7:0]	Number of Rules for Channel 4 Set the number of receive rules exclusively used for channel 4.
23 to 0	Reserved	These bits are always read as 0. The write value should always be 0.

Modify the RSCAN0GAFLCFG1 register only in global reset mode.

Up to 64 x (number of channels) rules can be registered in the receive rule table as the entire unit. The number of receive rules per channel should meet the following conditions.

- The maximum number of rules per channel is 128.
- The total of the number of rules allocated to each channel is not larger than the number of rules that can be registered in the entire unit.

#### RNC4[7:0] Bits

These bits are used to set the number of rules to be registered in the channel 4 receive rule table.

Set these bits to a value within the range of 00<sub>H</sub> to 80<sub>H</sub>.

### 21.3.15 RSCAN0GAFLIDj — Receive Rule ID Register (j = 0 to 15)

**Access:** Can be read/written in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0500<sub>H</sub> + (j \* 0010<sub>H</sub>)

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAFLID E	GAFLR TR	GAFLLB	GAFLID[28:16]												
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAFLID[15:0]															
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

**Table 21.29 RSCAN0GAFLIDj register contents**

Bit Position	Bit Name	Function
b31	GAFLIDE	IDE Select 0: Standard ID 1: Extended ID
b30	GAFLRTR	RTR Select 0: Data frame 1: Remote frame
b29	GAFLLB	Receive Rule Target Message Select 0: When a message transmitted from another CAN node is received 1: When the own transmitted message is received
b28 to b0	GAFLID[28:0]	ID Set the ID of the receive rule. For the standard ID, set the ID in bits b10 to b0 and set bits b28 to b11 to 0.

Modify the RSCAN0GAFLIDj register when the AFLDAE bit in the RSCAN0GAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

#### GAFLIDE Bit

This bit is used to select the ID format (standard ID or extended ID) of the receive rule. This bit is compared with the IDE bit in the received message during the acceptance filter processing.

#### GAFLRTR Bit

This bit is used to select the frame format (data frame or remote frame) of the receive rule. This bit is compared with the RTR bit in the received message during the acceptance filter processing.

**GAFLLB Bit**

When this bit is set to 0, data processing using the receive rule is performed when receiving messages transmitted from another CAN node.

When this bit is set to 1 when the mirror function is used, data processing using the receive rule is performed when the CAN node is receiving its own transmitted messages.

**GAFLLD[28:0] Bits**

These bits are used to set the ID field of the receive rule. The ID value set by these bits is compared with the ID of the received message during the acceptance filter processing.

### 21.3.16 RSCAN0GAFLMj — Receive Rule Mask Register (j = 0 to 15)

**Access:** Can be read/written in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0504<sub>H</sub> + (j \* 0010<sub>H</sub>)

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAFLID EM	GAFLR TRM	—	GAFLIDM[28:16]												
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAFLIDM[15:0]															
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

**Table 21.30 RSCAN0GAFLMj register contents**

Bit Position	Bit Name	Function
31	GAFLIDEM	IDE Mask 0: The IDE bit is not compared. 1: The IDE bit is compared.
30	GAFLRTRM	RTR Mask 0: The RTR bit is not compared. 1: The RTR bit is compared
29	Reserved	This bit is always read as 0. The write value should always be 0.
28 to 0	GAFLIDM[28:0]	ID Mask 0: The corresponding ID bit is not compared. 1: The corresponding ID bit is compared.

Modify the RSCAN0GAFLMj register when the AFLDAE bit in the RSCAN0GAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

#### GAFLIDEM Bit

When this bit is set to 1, filter processing is performed only for messages of the ID format specified by the GAFLIDE bit in the RSCAN0GAFLIDj register.

When this bit is cleared to 0, the IDs of all the receive messages and the specified IDs are regarded as matched. To set the GAFLIDEM bit to 0, set the GAFLIDM[28:0] bits to all 0 at the same time.

#### GAFLRTRM Bit

This bit is used to mask the RTR bit of the receive rule.

#### GAFLIDM[28:0] Bits

These bits are used to mask the corresponding ID bit of the receive rule.

### 21.3.17 RSCAN0GAFLP0j — Receive Rule Pointer 0 Register (j = 0 to 15)

**Access:** Can be read/written in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0508<sub>H</sub> + (j \* 0010<sub>H</sub>)

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAFLDLC[3:0]				GAFLPTR[11:0]											
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAFLR VM	GAFLRMDP[6:0]						—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

**Table 21.31 RSCAN0GAFLP0j register contents**

Bit Position	Bit Name	Function
31 to 28	GAFLDLC[3:0]	Receive Rule DLC b31 b30 b29 b28 0 0 0 0: DLC check is disabled. 0 0 0 1: 1 data byte 0 0 1 0: 2 data bytes 0 0 1 1: 3 data bytes 0 1 0 0: 4 data bytes 0 1 0 1: 5 data bytes 0 1 1 0: 6 data bytes 0 1 1 1: 7 data bytes 1 X X X: 8 data bytes
27 to 16	GAFLPTR[11:0]	Receive Rule Label Set the 12-bit label information.
15	GAFLRMV	Receive Buffer Enable 0: No receive buffer is used. 1: A receive buffer is used.
14 to 8	GAFLRMDP[6:0]	Receive Buffer Number Select Set the receive buffer number to store receive messages.
7 to 0	Reserved	These bits are always read as 0. The write value should always be 0.

Modify the RSCAN0GAFLP0j register when the AFLDAE bit in the RSCAN0GAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

#### GAFLDLC[3:0] Bits

These bits are used to set the minimum data length necessary for receiving messages. If the data length of a message that is being filtered is equal to or larger than the value set by the GAFLDLC[3:0] bits, the message passes the DLC check. Setting these bits to 0000<sub>B</sub> disables the DLC check function allowing messages with any data length to pass the DLC check.

#### GAFLPTR[11:0] Bits

These bits are used to set a 12-bit label to be attached to messages that have passed through the filter. A label is attached when a message is stored in the receive buffer or the FIFO buffer.

**GAFLRMV Bit**

When this bit is set to 1, receive messages that have passed through the filter are stored in the receive buffer selected by the GAFLRMDP[6:0] bits.

**GAFLRMDP[6:0] Bits**

These bits are used to select the number of the receive buffer that stores receive messages that have passed through the filter when the GAFLRMV bit is set to 1. Set these bits to a value smaller than the value set by the NRXMB[7:0] bits in the RSCAN0RMNB register.

### 21.3.18 RSCAN0GAFLP1j — Receive Rule Pointer 1 Register (j = 0 to 15)

**Access:** Can be read/written in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 050C<sub>H</sub> + (j \* 0010<sub>H</sub>)

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	GAFLFDP[22:16]						
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAFLFDP [15:0]															
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

**Table 21.32 RSCAN0GAFLP1j register contents**

Bit Position	Bit Name	Function
31 to 23	Reserved	These bits are always read as 0. The write value should always be 0.
22 to 0	GAFLFDP [22:0]	FIFO Buffer z Select (z = 0 to 22) z = 0 to 7 0: Receive FIFO buffer z is not selected. 1: Receiver FIFO buffer z is selected. z = 8 to 22 0: Transmit/receive FIFO buffer z-8 is not selected. 1: Transmit/receive FIFO buffer z-8 is selected.

Modify the RSCAN0GAFLP1j register when the AFLDAE bit in the RSCAN0GAFLECTR register is set to 1 (receive rule table write is enabled) in global reset mode.

#### GAFLFDP [22:0]

These bits are used to specify FIFO buffers that store receive messages that have passed through the filter. Up to eight FIFO buffers are selectable. However, when the GAFLRMV bit in the RSCAN0GAFLP0j register is set to 1 (a message is stored in the receive buffer), up to seven FIFO buffers can be selected.

Only receive FIFO buffers and the transmit/receive FIFO buffer for which the CFM[1:0] bits in the RSCAN0CFCK register are set to 00<sub>B</sub> (receive mode) or 10<sub>B</sub> (gateway mode) are selectable.

### 21.3.19 RSCAN0RMNB — Receive Buffer Number Register

**Access:** Can be read/written in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 00A4<sub>H</sub>

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	NRXMB[7:0]							
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

**Table 21.33 RSCAN0RMNB register contents**

Bit Position	Bit Name	Function
31 to 8	Reserved	These bits are always read as 0. The write value should always be 0.
7 to 0	NRXMB[7:0]	Receive Buffer Number Configuration Set the number of receive buffers. Set a value of 0 to 79.

Modify the RSCAN0RMNB register only in global reset mode.

#### NRXMB[7:0] Bits

These bits are used to set the total number of receive buffers of the RS-CAN module. The maximum value is  $16 \times$  (number of channels).

Setting these bits all to 0 makes receive buffers unavailable.



### 21.3.20 RSCAN0RMNDy — Receive Buffer New Data Register y (y = 0 to 2)

**Access:** Can be read/written in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 00A8<sub>H</sub> + (y \* 0004<sub>H</sub>)

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMNSq (q = y × 32 + 31 to y × 32 + 16 (y = 0, 1))															
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMNSq (q = y × 32 + 15 to y × 32 + 0 (y = 0, 1, 2))															
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

**Table 21.34 RSCAN0RMNDy register contents**

Bit Position	Bit Name	Function
31 to 16	RMNSq	Receive Buffer Receive Complete Flag q (q = y × 32 + 31 to y × 32 + 16) 0: There is no new message in receive buffer q. 1: There is a new message in receive buffer q.
15 to 0	RMNSq	Receive Buffer Receive Complete Flag q (q = y × 32 + 15 to y × 32 + 0) 0: There is no new message in receive buffer q. 1: There is a new message in receive buffer q.

Write 0 to the RSCAN0RMNDy register in global operating mode or global test mode.

#### RMNSq Flags(q = 0 to 79)

Each RMNS flag is set to 1 when the processing for storing a message in the corresponding receive buffer starts.

To clear any of these flags, the program must write 0 to the corresponding to the flag to be cleared and 1 to all other bits, using the MOV instruction. These bits cannot be set to 0 while a message is being stored. It takes ten clock cycles of pclk to store a message.

These flags are cleared to 0 in global reset mode.

### 21.3.21 RSCAN0RMIDq — Receive Buffer ID Register (q = 0 to 79)

**Access:** Can be read in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0600<sub>H</sub> + (q \* 0010<sub>H</sub>)

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMIDE	RMRTR	—	RMID[28:16]												
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMID[15:0]															
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

**Table 21.35 RSCAN0RMIDq register contents**

Bit Position	Bit Name	Function
31	RMIDE	Receive Buffer IDE 0: Standard ID 1: Extended ID
30	RMRTR	Receive Buffer RTR 0: Data frame 1: Remote frame
29	Reserved	This bit is always read as 0.
28 to 0	RMID[28:0]	Receive Buffer ID Data These bits contain the standard ID or extended ID of the received message. Read bits b10 to b0 for standard ID. Bits b28 to b11 are read as 0.

#### GAFLIDE Bit

This bit indicates the ID format (standard ID or extended ID) of the message stored in the receive buffer.

#### RMRTR Bit

This bit indicates the frame format (data frame or remote frame) of the message stored in the receive buffer.

#### RMID[28:0] Bits

These bits contain the ID of the message stored in the receive buffer.

### 21.3.22 RSCAN0RMPTRq — Receive Buffer Pointer Register (q = 0 to 79)

**Access:** Can be read in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0604<sub>H</sub> + (q \* 0010<sub>H</sub>)

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMDLC[3:0]				RMPTR[11:0]											
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMTS[15:0]															
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

**Table 21.36 RSCAN0RMPTRq**

Bit Position	Bit Name	Function
31 to 28	RMDLC[3:0]	Receive Buffer DLC Data b31 b30 b29 b28 0 0 0 0: 0 data bytes 0 0 0 1: 1 data byte 0 0 1 0: 2 data bytes 0 0 1 1: 3 data bytes 0 1 0 0: 4 data bytes 0 1 0 1: 5 data bytes 0 1 1 0: 6 data bytes 0 1 1 1: 7 data bytes 1 X X X: 8 data bytes
27 to 16	RMPTR[11:0]	Receive Buffer Label Data Label information of the received message.
15 to 0	RMTS[15:0]	Receive Buffer Timestamp Data Timestamp value of the received message.

#### RMDLC[3:0] Bits

These bits indicate the data length of the message stored in the receive buffer.

#### RMPTR[11:0] Bits

These bits indicate the label information of the message stored in the receive buffer.

#### RMTS[15:0] Bits

These bits indicate the timestamp value of the message stored in the receive buffer.

### 21.3.23 RSCAN0RMDF0q — Receive Buffer Data Field 0 Register (q = 0 to 79)

**Access:** Can be read in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0608<sub>H</sub> + (q \* 0010<sub>H</sub>)

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMDB3[7:0]								RMDB2[7:0]							
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMDB1[7:0]								RMDB0[7:0]							
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

**Table 21.37 RSCAN0RMDF0q register contents**

Bit Position	Bit Name	Function
31 to 24	RMDB3[7:0]	Receive Buffer Data Byte 3
23 to 16	RMDB2[7:0]	Receive Buffer Data Byte 2
15 to 8	RMDB1[7:0]	Receive Buffer Data Byte 1
7 to 0	RMDB0[7:0]	Receive Buffer Data Byte 0
		Data for a message stored in the receive buffer can be read.

When the RMDLC[3:0] value in the RSCAN0RMPTRq register is smaller than 1000<sub>B</sub>, data bytes for which no data is set are read as 00<sub>H</sub>.

### 21.3.24 RSCAN0RMDF1q — Receive Buffer Data Field 1 Register (q = 0 to 79)

**Access:** Can be read in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 060C<sub>H</sub> + (q \* 0010<sub>H</sub>)

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMDB7[7:0]								RMDB6[7:0]							
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMDB5[7:0]								RMDB4[7:0]							
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

**Table 21.38 RSCAN0RMDF1q register contents**

Bit Position	Bit Name	Function
31 to 24	RMDB7[7:0]	Receive Buffer Data Byte 7
23 to 16	RMDB6[7:0]	Receive Buffer Data Byte 6
15 to 8	RMDB5[7:0]	Receive Buffer Data Byte 5
7 to 0	RMDB4[7:0]	Receive Buffer Data Byte 4
		Data for a message stored in the receive buffer can be read.

When the RMDLC[3:0] value in the RSCAN0RMPTRq register is smaller than 1000<sub>B</sub>, data bytes for which no data is set are read as 00<sub>H</sub>.

### 21.3.25 RSCAN0RFCCx — Receive FIFO Buffer Configuration and Control Register (x = 0 to 7)

**Access:** Can be read/written in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 00B8<sub>H</sub> + (x \* 0004<sub>H</sub>)

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFIGCV[2:0]			RFIM	—	RFDC[2:0]			—	—	—	—	—	—	RFIE	RFE
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W

**Table 21.39 RSCAN0RFCCx register contents**

Bit Position	Bit Name	Function
31 to 16	Reserved	These bits are always read as 0. The write value should always be 0.
15 to 13	RFIGCV[2:0]	Receive FIFO Interrupt Request Timing Select b <sup>15</sup> b <sup>14</sup> b <sup>13</sup> 0 0 0: When FIFO is 1/8 full. 0 0 1: When FIFO is 2/8 full. 0 1 0: When FIFO is 3/8 full. 0 1 1: When FIFO is 4/8 full. 1 0 0: When FIFO is 5/8 full. 1 0 1: When FIFO is 6/8 full. 1 1 0: When FIFO is 7/8 full. 1 1 1: When FIFO is full.
12	RFIM	Receive FIFO Interrupt Source Select 0: An interrupt occurs when the condition set by the RFIGCV[2:0] bits is met. 1: An interrupt occurs each time a message has been received.
11	Reserved	This bit is always read as 0. The write value should always be 0.
10 to 8	RFDC[2:0]	Receive FIFO Buffer Depth Configuration b <sup>10</sup> b <sup>9</sup> b <sup>8</sup> 0 0 0: 0 messages 0 0 1: 4 messages 0 1 0: 8 messages 0 1 1: 16 messages 1 0 0: 32 messages 1 0 1: 48 messages 1 1 0: 64 messages 1 1 1: 128 messages
7 to 2	Reserved	These bits are always read as 0. The write value should always be 0.
1	RFIE	Receive FIFO Interrupt Enable 0: Receive FIFO interrupt is disabled. 1: Receive FIFO interrupt is enabled.
0	RFE	Receive FIFO Buffer Enable 0: No receive FIFO buffer is used. 1: Receive FIFO buffers are used.

**RFIGCV[2:0] Bits**

These bits are used to specify the number of received messages for generating a receive FIFO interrupt request when the RFIM bit is set to 0 with a fraction for the total number of buffers (the setting of RFDC[2:0]).

When the RFDC[2:0] bits are set to 001<sub>B</sub> (4 messages), set the RFIGCV[2:0] bits to 001<sub>B</sub>, 011<sub>B</sub>, 101<sub>B</sub>, or 111<sub>B</sub> (fractions which are even multiples of 1/4). Modify these bits only in global reset mode.

**RFIM Bit**

This bit is used to select a FIFO interrupt source. Modify this bit only in global reset mode.

**RFDC[2:0] Bits**

These bits are used to select the number of messages that can be stored in a single receive FIFO buffer. When these bits are set to 000<sub>B</sub>, no receive FIFO buffer should be used. Modify these bits only in global reset mode.

**RFIE Bit**

Setting the RFIE bit to 1 enables receive FIFO interrupts. Modify this bit when the RFE bit set to 0 (no receive FIFO buffer is used).

**RFE Bit**

Setting the RFE bit to 1 makes receive FIFO buffers available. Clearing this bit to 0 sets the RFEMP flag in the RSCAN0RFSTSx register to 1 (the receive FIFO buffer contains no unread message (buffer empty)). Modify this bit in global operating mode or global test mode.

### 21.3.26 RSCAN0RFST<sub>x</sub> — Receive FIFO Buffer Status Register (x = 0 to 7)

**Access:** Can be read in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 00D8<sub>H</sub> + (x \* 0004<sub>H</sub>)

**Initial value:** 0000 0001<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFMC[7:0]							—	—	—	—	RFIF	RFMLT	RFFLL	RFEMP	
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W*1	R/W*1	R	R

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

**Table 21.40 RSCAN0RFST<sub>x</sub> register contents**

Bit Position	Bit Name	Function
31 to 16	Reserved	These bits are always read as 0. The write value should always be 0.
15 to 8	RFMC[7:0]	Receive FIFO Unread Message Counter The number of unread messages stored in the receive FIFO buffer is displayed.
7 to 4	Reserved	These bits are always read as 0. The write value should always be 0.
3	RFIF	Receive FIFO Interrupt Request Flag 0: No receive FIFO interrupt request is present. 1: A receive FIFO interrupt request is present.
2	RFMLT	Receive FIFO Message Lost Flag 0: No receive FIFO message is lost. 1: A receive FIFO message is lost.
1	RFFLL	Receive FIFO Buffer Full Status Flag 0: The receive FIFO buffer is not full. 1: The receive FIFO buffer is full.
0	RFEMP	Receive FIFO Buffer Empty Status Flag 0: The receive FIFO buffer contains unread message. 1: The receive FIFO buffer contains no unread message (buffer empty).

#### RFMC[7:0] Flag

This flag indicates the number of unread messages in the receive FIFO buffer. This flag becomes 00<sub>H</sub> when the RFE bit in the RSCAN0RFCC<sub>x</sub> register is set to 0.



**RFIF Flag**

This flag is set to 1 when the receive FIFO interrupt request generation conditions set by the RFIGCV[2:0] bits and the RFIM bit in the RSCAN0RFCCx register are met. This flag is cleared to 0 in global reset mode or by writing 0 to this flag. Modify this bit in global operating mode or global test mode.

**RFMLT Flag**

This flag is set to 1 when an attempt is made to store a new message while the receive FIFO buffer is full. In this case, the new message is discarded.

This flag is cleared to 0 in global reset mode or by writing 0 to this flag.

Modify this bit in global operating mode or global test mode.

**CFLL Flag**

This flag is set to 1 when the number of messages stored in the receive FIFO buffer matches the FIFO buffer depth set by the RFDC[2:0] bits in the RSCAN0RFCCx register.

If the number of messages stored in the receive FIFO buffer becomes smaller than the FIFO buffer depth set by the RFDC[2:0] bits, this flag is cleared to 0. This flag is also cleared to 0 when the RFE bit in the RSCAN0RFCCx register is set to 0 (no receive FIFO buffer is used) or in global reset mode.

**RFEMP Flag**

This flag is set to 1 when all messages in the receive FIFO buffer have been read. This flag is also set to 1 when the RFE bit in the RSCAN0RFCCx register is 0 or in global reset mode.

This flag is cleared to 0 when even a single received message has been stored in the receive FIFO buffer.

### 21.3.27 RSCAN0RFPCTR<sub>x</sub> — Receive FIFO Buffer Pointer Control Register ( $x = 0$ to 7)

**Access:** Can be written in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 00F8<sub>H</sub> + ( $x * 0004$ <sub>H</sub>)

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	RFPC[7:0]							
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

**Table 21.41 RSCAN0RFPCTR<sub>x</sub> register contents**

Bit Position	Bit Name	Function
31 to 8	Reserved	The write value should always be 0.
7 to 0	RFPC[7:0]	Receive FIFO Pointer Control When these bits are set to FF <sub>H</sub> , the read pointer moves to the next unread message in the receive FIFO buffer.

#### RFPC[7:0] Bits

When the RFPC[7:0] bits are set to FF<sub>H</sub>, the read pointer moves to the next unread message in the receive FIFO buffer. At this time, the RFMC[7:0] (receive FIFO unread message counter) value in the RSCAN0RFSTS<sub>x</sub> register is decremented. Read the RSCAN0RFID, RSCAN0RFPTR, RSCAN0RDF0, and RSCAN0RDF1 registers to read messages in the receive FIFO buffer, and then write FF<sub>H</sub> to the RFPC[7:0] bits.

Write FF<sub>H</sub> to these bits when the RFE bit in the RSCAN0RFCC<sub>m</sub> register is set to 1 (receive FIFO buffers are used) and the RFEMP flag in the RSCAN0RFSTS<sub>m</sub> register is 0 (the receive FIFO buffer contains unread messages).

### 21.3.28 RSCAN0RFIDx — Receive FIFO Buffer Access ID Register (x = 0 to 7)

**Access:** Can be read in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0E00<sub>H</sub> + (x \* 0010<sub>H</sub>)

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFIDE	RFRTR	—	RFID[28:16]												
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFID[15:0]															
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

**Table 21.42 RSCAN0RFIDx register contents**

Bit Position	Bit Name	Function
31	RFIDE	Receive FIFO Buffer IDE 0: Standard ID 1: Extended ID
30	RFRTR	Receive FIFO Buffer RTR 0: Data frame 1: Remote frame
29	Reserved	This bit is always read as 0.
28 to 0	RFID[28:0]	Receive FIFO Buffer ID Data The standard ID or extended ID of received message can be read. Read bits b10 to b0 for standard ID. Bits b28 to b11 are read as 0.

#### RFIDE Bit

This bit indicates the ID format (standard ID or extended ID) of the message stored in the receive FIFO buffer.

#### RFRTR Bit

This bit indicates the frame format (data frame or remote frame) of the message stored in the receive FIFO buffer.

#### RFID[28:0] Bits

These bits indicate the ID of the message stored in the receive FIFO buffer.

### 21.3.29 RSCAN0RFPTRx — Receive FIFO Buffer Access Pointer Register (x = 0 to 7)

**Access:** Can be read in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0E04<sub>H</sub> + (x \* 0010<sub>H</sub>)

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFDLC[3:0]				RFPTR[11:0]											
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFTS[15:0]															
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

**Table 21.43 RSCAN0RFPTRx register contents**

Bit Position	Bit Name	Function
31 to 28	RFDLC[3:0]	Receive FIFO Buffer DLC Data b31 b30 b29 b28 0 0 0 0: 0 data bytes 0 0 0 1: 1 data byte 0 0 1 0: 2 data bytes 0 0 1 1: 3 data bytes 0 1 0 0: 4 data bytes 0 1 0 1: 5 data bytes 0 1 1 0: 6 data bytes 0 1 1 1: 7 data bytes 1 X X X: 8 data bytes
27 to 16	RFPTR[11:0]	Receive FIFO Buffer Label Data Label information of the received message can be read.
15 to 0	RFTS[15:0]	Receive FIFO Buffer Timestamp Data Timestamp value of the received message can be read.

#### RFDLC[3:0] Bits

These bits contain the data length of the message stored in the receive FIFO buffer.

#### RFPTR[11:0] Bits

These bits contain the label information of the message stored in the receive FIFO buffer.

#### RFTS[15:0] Bits

These bits contain the timestamp value of the message stored in the receive FIFO buffer.

### 21.3.30 RSCAN0RFDF0x — Receive FIFO Buffer Access Data Field 0 Register (x = 0 to 7)

**Access:** Can be read in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0E08<sub>H</sub> + (x \* 0010<sub>H</sub>)

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFDB3[7:0]								RFDB2[7:0]							
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFDB1[7:0]								RFDB0[7:0]							
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

**Table 21.44 RSCAN0RFDF0x register contents**

Bit Position	Bit Name	Function
31 to 24	RFDB3[7:0]	Receive FIFO Buffer Data Byte 3
23 to 16	RFDB2[7:0]	Receive FIFO Buffer Data Byte 2
15 to 8	RFDB1[7:0]	Receive FIFO Buffer Data Byte 1
7 to 0	RFDB0[7:0]	Receive FIFO Buffer Data Byte 0
		Data for a message stored in the receive FIFO buffer can be read.

When the RFDLC[3:0] value in the RSCAN0RFPTRx register is smaller than 1000<sub>B</sub>, data bytes for which no data is set are read as 00<sub>H</sub>.

### 21.3.31 RSCAN0RFDF1x — Receive FIFO Buffer Access Data Field 1 Register (x = 0 to 7)

**Access:** Can be read in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0E0C<sub>H</sub> + (x \* 0010<sub>H</sub>)

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFDB7[7:0]								RFDB6[7:0]							
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFDB5[7:0]								RFDB4[7:0]							
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

**Table 21.45 RSCAN0RFDF1x register contents**

Bit Position	Bit Name	Function
31 to 24	RFDB7[7:0]	Receive FIFO Buffer Data Byte 7
23 to 16	RFDB6[7:0]	Receive FIFO Buffer Data Byte 6
15 to 8	RFDB5[7:0]	Receive FIFO Buffer Data Byte 5
7 to 0	RFDB4[7:0]	Data for a message stored in the receive FIFO buffer can be read.

When the RFDLC[3:0] value in the RSCAN0RFPTRx register is smaller than 1000<sub>B</sub>, data bytes for which no data is set are read as 00<sub>H</sub>.

### 21.3.32 RSCAN0CFCK — Transmit/receive FIFO buffer Configuration and Control Register k (k = 0 to 14)

**Access:** Can be read/written in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0118<sub>H</sub> + (k \* 0004<sub>H</sub>)

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFITT[7:0]							CFTML[3:0]			CFITR	CFITSS	CFM[1:0]			
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFIGCV[2:0]		CFIM	—	CFDC[2:0]		—	—	—	—	—	—	CFIXIE	CFRXIE	CFE	
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

**Table 21.46 RSCAN0CFCK register contents (1/2)**

Bit Position	Bit Name	Function
31 to 24	CFITT[7:0]	Set a message transmission interval. Set Value: 00 <sub>H</sub> to FF <sub>H</sub>
23 to 20	CFTML[3:0]	Transmit Buffer Link Configuration Set the transmit buffer number to be linked to the transmit/receive FIFO buffer.
19	CFITR	Transmit/Receive FIFO Interval Timer Resolution 0: Clock dividing pclk by ((ITRCP [15:0] bits) 1: Clock dividing pclk by ((ITRCP [15:0] bits × 10)
18	CFITSS	Transmit/Receive FIFO Interval Timer Clock Source Select 0: Interval timer clock source selected by the CFITR bit 1: Interval timer clock source is the bit time clock for the channel to which the FIFO is linked.
17, 16	CFM[1:0]	Transmit/Receive FIFO Mode Select b17 b16 0 0: Receive mode 0 1: Transmit mode 1 0: Gateway mode 1 1: Setting prohibited
15 to 13	CFIGCV[2:0]	Transmit/Receive FIFO Receive Interrupt Request Timing Select b15 b14 b13 0 0 0: When FIFO is 1/8 full. 0 0 1: When FIFO is 2/8 full. 0 1 0: When FIFO is 3/8 full. 0 1 1: When FIFO is 4/8 full. 1 0 0: When FIFO is 5/8 full. 1 0 1: When FIFO is 6/8 full. 1 1 0: When FIFO is 7/8 full. 1 1 1: When FIFO is full.

Table 21.46 RSCAN0CFCCk register contents (2/2)

Bit Position	Bit Name	Function
b12	CFIM	Transmit/Receive FIFO Interrupt Source Select 0: <ul style="list-style-type: none"> <li>Receive mode/gateway mode When the number of received messages has met the condition set by the CFIMGV[2:0] bits, a FIFO receive interrupt request is generated.</li> <li>Transmit mode/gateway mode When the buffer becomes empty upon completion of message transmission, a FIFO transmit interrupt request is generated.</li> </ul> 1: <ul style="list-style-type: none"> <li>Receive mode/gateway mode A FIFO receive interrupt request is generated each time a message has been received.</li> <li>Transmit mode/gateway mode A FIFO transmit interrupt request is generated each time a message has been transmitted.</li> </ul>
b11	Reserved	This bit is always read as 0. The write value should always be 0.
b10 to b8	CFDC[2:0]	Transmit/Receive FIFO Buffer Depth Configuration b10 b9 b8 0 0 0: 0 messages 0 0 1: 4 messages 0 1 0: 8 messages 0 1 1: 16 messages 1 0 0: 32 messages 1 0 1: 48 messages 1 1 0: 64 messages 1 1 1: 128 messages
b7 to b3	Reserved	These bits are always read as 0. The write value should always be 0.
b2	CFTXIE	Transmit/Receive FIFO Transmit Interrupt Enable 0: Transmit/receive FIFO transmit interrupt is disabled. 1: Transmit/receive FIFO transmit interrupt is enabled.
b1	CFRXIE	Transmit/Receive FIFO Receive Interrupt Enable 0: Transmit/receive FIFO receive interrupt is disabled. 1: Transmit/receive FIFO receive interrupt is enabled.
b0	CFE	Transmit/Receive FIFO Buffer Enable 0: No transmit/receive FIFO buffer is used. 1: Transmit/receive FIFO buffers are used.

### CFITT[7:0] Bits

These bits are used to set a message transmission interval when transmitting messages continuously from a transmit/receive FIFO buffer whose CFM[1:0] bits are set to 01<sub>B</sub> (transmit mode) or 10<sub>B</sub> (gateway mode).

Clear the CFE bit to 0 (no transmit/receive FIFO buffer is used) before modifying the CFITT[7:0] bits.

### CFTML[3:0] Bits

These bits are used to set the number of transmit buffer on the channel which will be linked to transmit/receive FIFO buffer k when the CFM[1:0] bits are set to 01<sub>B</sub> (transmit mode) or 10<sub>B</sub> (gateway mode). There are three transmit/receive FIFO buffers per channel, so channel number n of FIFO buffer k is calculated as  $m = k/3$  (integer division). The actual assigned transmit buffer number p linked to FIFO buffer k will be  $((16 \times m) + CFTML[3:0])$ .

Refer to **Table 21.11** and **Table 21.12**, as for the relationship between transmit/receive FIFO buffer k and transmit buffer p.



Setting the CFDC[2:0] bits to 001<sub>B</sub> or more enables the setting of the CFTML[3:0] bits.

Do not link to any transmit buffer which is already allocated to a transmit queue on the identical channel or to another transmit/receive FIFO buffer. Modify these bits only in global reset mode.

#### **CFITR Bit**

This bit is enabled when the CFITSS bit is 1.

When this bit is 0, the interval timer clock source is the pclk/2 clock divided by the value of the ITRCP[15:0] bits in the RSCAN0GCFG register.

When this bit is 1, the interval timer clock source is the pclk/2 clock divided by (the value of the ITRCP[15:0] bits in the RSCAN0GCFG register × 10).

#### **CFITSS Bit**

When this bit is 0, the clock selected by the CFITR bit is the count source of the interval timer.

When this bit is 1, the bit time clock of the channel to which the FIFO is linked is the count source of the interval timer.

#### **CFM[1:0] Bits**

These bits are used to select transmit/receive FIFO mode. Modify these bits only in global reset mode.

#### **CFIGCV[2:0] Bits**

These bits are used to specify the number of received messages for generating a transmit/receive FIFO receive interrupt request when the CFM[1:0] bits are set to 00<sub>B</sub> (receive mode) or 10<sub>B</sub> (gateway mode) and the CFIM bit is set to 0 with a fraction for the total number of buffers (the setting of CFDC[2:0]).

An interrupt request is generated when the number of stored messages reaches the specified ratio (in fraction) of the storable messages set with the CFDC[2:0] bits.

When the CFDC[2:0] bits are set to 001<sub>B</sub> (4 messages), set the CFIGCV[2:0] bits to 001<sub>B</sub>, 011<sub>B</sub>, 101<sub>B</sub>, or B'111.

Modify these bits only in global reset mode.

#### **CFIM Bit**

This bit is used to select a transmit/receive FIFO interrupt source. Modify this bit only in global reset mode.

#### **CFDC[2:0] Bits**

These bits are used to set the number of messages that can be stored in a single transmit/receive FIFO buffer. When these bits are set to 000<sub>B</sub>, do not use a transmit/receive FIFO buffer. Modify these bits only in global reset mode.

#### **CFTXIE Bit**

When this bit is set to 1 and the CFTXIF flag in the RSCAN0CFSTSk register is set to 1, a transmit/receive FIFO transmit interrupt request is generated.

Modify this bit with the CFE bit set to 0 (no transmit/receive FIFO buffer is used).

**CFRXIE Bit**

When this bit is set to 1 and the CFRXIF flag in the RSCAN0CFSTSk register is set to 1, a transmit/receive FIFO receive interrupt request is generated.

Modify this bit with the CFE bit set to 0.

**CFE Bit**

Setting this bit to 1 makes transmit/receive FIFO buffers available.

When this bit is set to 0 in transmit mode or gateway mode, if a message in the transmit/receive FIFO buffer is being transmitted or will be transmitted next, the transmit/receive FIFO buffer becomes empty after completion of transmission of that message, or upon detection of a CAN bus error, or arbitration-lost. In other cases or in receive mode, the transmit/receive FIFO buffer becomes empty immediately.

This bit is cleared to 0 when the following conditions are met.

- Receive mode: Global reset mode
- Transmit mode or gateway mode: Channel reset mode  
Modify this bit in the following mode.
- Receive mode: Global operating mode or global test mode
- Transmit mode or gateway mode: Channel communication mode or channel halt mode

### 21.3.33 RSCAN0CFSTSk — Transmit/receive FIFO buffer Status Register (k = 0 to 14)

**Access:** Can be read/written in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0178<sub>H</sub> + (k \* 0004<sub>H</sub>)

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFMC[7:0]							—	—	—	CFTXIF	CFRXIF	CFMLT	CFLL	CFEMP	
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W*1	R/W*1	R/W*1	R	R

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

**Table 21.47 RSCAN0CFSTSk register contents**

Bit Position	Bit Name	Function
31 to 16	Reserved	These bits are always read as 0. The write value should always be 0.
15 to 8	CFMC[7:0]	Transmit/Receive FIFO Message Counter The number of messages stored in the transmit/receive FIFO buffer.
7 to 5	Reserved	These bits are always read as 0. The write value should always be 0.
4	CFTXIF	Transmit/Receive FIFO Transmit Interrupt Request Flag 0: No transmit/receive FIFO transmit interrupt request is present. 1: A transmit/receive FIFO transmit interrupt request is present.
3	CFRXIF	Transmit/Receive FIFO Receive Interrupt Request Flag 0: No transmit/receive FIFO receive interrupt request is present. 1: A transmit/receive FIFO receive interrupt request is present.
2	CFMLT	Transmit/Receive FIFO Message Lost Flag 0: No transmit/receive FIFO message is lost. 1: A transmit/receive FIFO message is lost.
1	CFLL	Transmit/Receive FIFO Buffer Full Status Flag 0: The transmit/receive FIFO buffer is not full. 1: The transmit/receive FIFO buffer is full.
0	CFEMP	Transmit/Receive FIFO Buffer Empty Status Flag 0: The transmit/receive FIFO buffer contains messages. 1: The transmit/receive FIFO buffer contains no message (buffer empty).

### CFMC[7:0] Bits

The CFMC[7:0] bits indicate the following values that depend on the setting of the CFM[1:0] bits in the RSCAN0CFCCk register.

- When CFM[1:0] value is 01<sub>B</sub> (transmit mode): Number of untransmitted messages in the buffer
- When CFM[1:0] value is 00<sub>B</sub> (receive mode): Number of unread received messages in the buffer
- When CFM[1:0] value is 10<sub>B</sub> (gateway mode): Number of untransmitted received messages in the buffer

These bits are cleared to 0 when any of the following conditions is met.

- When CFM[1:0] value is 00<sub>B</sub>: In global reset mode
- When CFM[1:0] value is 01<sub>B</sub> or 10<sub>B</sub>: In channel reset mode

### CFTXIF Flag

The CFTXIF flag is set to 1 when any of the following conditions is met.

- When the CFM[1:0] bits are set to 01<sub>B</sub> or 10<sub>B</sub>, and the factor selected by the CFIM bit in the RSCAN0CFCCk register occurs

The CFTXIF flag is cleared to 0 when any of the following conditions is met.

- When 0 is written to the CFTXIF flag
- When the CFM[1:0] bits are set to 00<sub>B</sub>: In global reset mode
- When the CFM[1:0] bits are set to 01<sub>B</sub> or 10<sub>B</sub>: In channel reset mode

Write 0 to this flag in global operating mode or global test mode.

### CFRXIF Flag

The CFRXIF flag is set to 1 when any of the following conditions is met.

- When the CFM[1:0] bits are set to 00<sub>B</sub> or 10<sub>B</sub>, and the factor selected by the CFIM bit in the RSCAN0CFCCk register occurs

The CFRXIF flag is cleared to 0 when any of the following conditions is met.

- When 0 is written to the CFRXIF flag
- When the CFM[1:0] bits are set to 00<sub>B</sub>: In global reset mode
- When the CFM[1:0] bits are set to 01<sub>B</sub> or 10<sub>B</sub>: In channel reset mode

Write 0 to this flag in global operating mode or global test mode.

### CFMLT Flag

The CFMLT flag is set to 1 when any of the following conditions is met.

- When an attempt is made to store a new message while the transmit/receive FIFO buffer is full. In this case, the new message is discarded.

The CFMLT flag is cleared to 0 when any of the following conditions is met.

- When 0 is written to the CFMLT flag
- When the CFM[1:0] bits are set to 00<sub>B</sub>: In global reset mode

- When the CFM[1:0] bits are set to 01<sub>B</sub> or 10<sub>B</sub>: In channel reset mode

Write 0 to this flag in global operating mode or global test mode

### CFLL Flag

The CFLL flag is set to 1 when any of the following conditions is met.

- When the number of messages stored in the transmit/receive FIFO buffer matches the FIFO buffer depth set by the CFDC[2:0] bits in the RSCAN0CFCCk register.

The CFMLT flag is cleared to 0 when any of the following conditions is met.

- When the number of messages stored in the transmit/receive FIFO buffer becomes smaller than the FIFO buffer depth set by the CFDC[2:0] bits.
- When the CFE bit in the RSCAN0CFCCk register is 0 (no transmit/receive FIFO buffer is used):  
When not in the transmit abort
- When the CFM[1:0] bits are set to 00<sub>B</sub>: In global reset mode
- When the CFM[1:0] bits are set to 01<sub>B</sub> or 10<sub>B</sub>: In channel reset mode

### CFEMP Flag

The CFEMP flag is set to 1 when any of the following conditions is met.

- When the CFM[1:0] bits are set to 00<sub>B</sub> or 10<sub>B</sub>: All messages have been read, or in global reset mode
- When the CFM[1:0] bits are set to 01<sub>B</sub>: All messages have been transmitted, or in channel reset mode
- When the CFE bit is 0 (no transmit/receive FIFO buffer is used): Not in the transmit abort

The CFEMP flag is cleared to 0 when any of the following conditions is met.

- When the CFM[1:0] bits are set to 00<sub>B</sub>: At least one received message has been stored in the transmit/receive FIFO buffer.
- When the CFM[1:0] bits are set to 01<sub>B</sub> or 10<sub>B</sub>: A value of FFH has been written to the RSCAN0CFPCTRk register after data was written to the RSCAN0CFIDk, RSCAN0CFPTRk, RSCAN0CFDF0k, and RSCAN0CFDF1k registers.

### 21.3.34 RSCAN0CFPCTRk — Transmit/receive FIFO buffer Pointer Control Register (k = 0 to 14)

**Access:** Can be written in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 01D8<sub>H</sub> + (k \* 0004<sub>H</sub>)

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CFPC[7:0]							
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

**Table 21.48 RSCAN0CFPCTRk register contents**

Bit Position	Bit Name	Function
31 to 8	Reserved	The write value should always be 0.
7 to 0	CFPC[7:0]	Transmit/Receive FIFO Pointer Control <ul style="list-style-type: none"> <li>• Receive mode: Writing FF<sub>H</sub> to these bits moves the read pointer to the next unread message in the transmit/receive FIFO buffer.</li> <li>• Transmit mode: Writing FF<sub>H</sub> to these bits moves the write pointer to the next stage of the transmit/receive FIFO buffer.</li> <li>• Gateway mode: Setting prohibited</li> </ul>

#### CFPC[7:0] Bits

- Receive mode (CFM[1:0] value in the RSCAN0CFCCk register is 00<sub>B</sub>):  
Writing FF<sub>H</sub> to the CFPC[7:0] bits moves the read pointer to the next unread message in the transmit/receive FIFO buffer. At this time, the CFMC[7:0] value (transmit/receive FIFO message counter) in the RSCAN0CFSTSk register is decremented. Read the RSCAN0CFIDk, RSCAN0CFPTRk, RSCAN0CFDF0k, and RSCAN0CFDF1k registers to read messages from the transmit/receive FIFO buffer, and then write FF<sub>H</sub> to the CFPC[7:0] bits.  
Write FF<sub>H</sub> to these bits when the CFE bit in the RSCAN0CFCCk register is set to 1 (transmit/receive FIFO buffers are used) and the CFEMP flag in the RSCAN0CFSTSk register is cleared to 0 (the transmit/receive FIFO buffer contains messages).

- Transmit mode (CFM[1:0] value in the RSCAN0FCCK register is 01<sub>B</sub>):  
Writing FF<sub>H</sub> to the CFPC[7:0] bits stores the data written to the RSCAN0CFIDk, RSCAN0CFPTRk, RSCAN0CFDF0k, and RSCAN0CFDF1k registers in the transmit/receive FIFO buffer and moves the write pointer to the next stage of the transmit/receive FIFO buffer. At this time, the CFMC[7:0] value is incremented. Write transmit messages to the RSCAN0CFIDk, RSCAN0CFPTRk, RSCAN0CFDF0k, and RSCAN0CFDF1k registers before writing FF<sub>H</sub> to the CFPC[7:0] bits.  
Write FF<sub>H</sub> to these bits when the CFE bit in the RSCAN0FCCK register is set to 1 and the CFFLL flag in the RSCAN0CFSTSk register is cleared to 0 (the transmit/receive FIFO buffer is not full).
- Gateway mode (CFM[1:0] value in the RSCAN0FCCK register is 10<sub>B</sub>):  
Setting prohibited

### 21.3.35 RSCAN0CFIDk — Transmit/receive FIFO buffer Access ID Register (k = 0 to 14)

**Access:** Can be read/written in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0E80<sub>H</sub> + (k \* 0010<sub>H</sub>)

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFIDE	CFRTR	THLEN	CFID[28:16]												
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFID[15:0]															
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

**Table 21.49 RSCAN0CFIDk register contents**

Bit Position	Bit Name	Function
b31	CFIDE	Transmit/Receive FIFO Buffer IDE 0: Standard ID 1: Extended ID
b30	CFRTR	Transmit/Receive FIFO Buffer RTR 0: Data frame 1: Remote frame
b29	THLEN	Transmit History Data Store Enable This bit is valid only when the CFM[1:0] value is 01 <sub>B</sub> (transmit mode). 0: Transmit history data is not stored in the buffer. 1: Transmit history data is stored in the buffer.
b28 to b0	CFID[28:0]	Transmit/Receive FIFO Buffer ID Data <ul style="list-style-type: none"> <li>When CFM[1:0] value is B'01 (transmit mode): Set standard ID or extended ID. For standard ID, write an ID to bits b10 to b0 and write 0 to bits b28 to b11.</li> <li>When CFM[1:0] value is B'00 (receive mode): Standard ID or extended ID in the received message can be read. For standard ID, read bits b10 to b0. Bits b28 to b11 are read as 0.</li> </ul>

This register is writable only when the CFM[1:0] value in the RSCAN0FCCK register is 01<sub>B</sub> (transmit mode). This register is readable only when the CFM[1:0] value is 00<sub>B</sub> (receive mode). This RSCAN0CFIDk register should not be read or written when the CFM[1:0] value is 10<sub>B</sub> (gateway mode).

#### CFIDE Bit

This bit indicates the ID format (standard ID or extended ID) of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00<sub>B</sub>. When the CFM[1:0] value is 01<sub>B</sub>, these bits are used to set the ID format of the message to be transmitted from the transmit/receive FIFO buffer.



**CFRTR Bit**

This bit indicates the data format (data frame or remote frame) of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00<sub>B</sub>. When the CFM[1:0] value is 01<sub>B</sub>, this bit is used to set the data format of the message to be transmitted from the transmit/receive FIFO buffer.

**THLEN Bit**

When this bit is set to 1, the transmit history data (label information, buffer number, and buffer type) of transmit messages is stored in the transmit history buffer after transmission is completed.

This bit is enabled when the CFM[1:0] value is 01<sub>B</sub> (transmit mode).

**CFID[28:0] Bits**

These bits contain the ID of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00<sub>B</sub>.

When the CFM[1:0] value is B'01, this bit is used to set the ID of the message to be transmitted from the transmit/receive FIFO buffer.

### 21.3.36 RSCAN0CFPTRk — Transmit/receive FIFO buffer Access Pointer Register (k = 0 to 14)

**Access:** Can be read/written in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0E84<sub>H</sub> + (k \* 0010<sub>H</sub>)

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFDLC[3:0]				CFPTR[11:0]											
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFTS[15:0]															
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

**Table 21.50 RSCAN0CFPTRk register contents**

Bit Position	Bit Name	Function
b31 to b28	CFDLC[3:0]	Transmit/Receive FIFO Buffer DLC Data b31 b30 b29 b28 0 0 0 0: 0 data bytes 0 0 0 1: 1 data byte 0 0 1 0: 2 data bytes 0 0 1 1: 3 data bytes 0 1 0 0: 4 data bytes 0 1 0 1: 5 data bytes 0 1 1 0: 6 data bytes 0 1 1 1: 7 data bytes 1 X X X: 8 data bytes
b27 to b16	CFPTR[11:0]	Transmit/Receive FIFO Buffer Label Data <ul style="list-style-type: none"> <li>When CFM[1:0] value is 01<sub>B</sub> (transmit mode): Set the label information to be stored in the transmit history buffer. Only bits CFPTR[7:0] are valid.</li> <li>When CFM[1:0] value is 00<sub>B</sub> (receive mode): The label information of the received message can be read.</li> </ul>
b15 to b0	CFTS[15:0]	Transmit/Receive FIFO Buffer Timestamp Data These bits are valid only when the CFM[1:0] value is B'00 (receive mode). The timestamp value of the received message can be read.

This register is writable only when the CFM[1:0] value in the RSCAN0CFCCk register is 01<sub>B</sub> (transmit mode). This register is readable only when the CFM[1:0] value is 00<sub>B</sub> (receive mode). This register should not be read or written when the CFM[1:0] value is 10<sub>B</sub> (gateway mode).

**CFDLC[3:0] Bits**

These bits indicate the data length of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00<sub>B</sub>. When the CFM[1:0] value is 01<sub>B</sub>, these bits are used to set the data length of the message to be transmitted from the transmit/receive FIFO buffer. If the data length is set to 9 bytes or more, the actual transmit data defaults to 8 bytes.

**CFPTR[11:0] Bits**

These bits indicate the label information attached to the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00<sub>B</sub>. When the CFM[1:0] value is 01<sub>B</sub>, the CFPTR[7:0] value is stored in the transmit history buffer when message transmission has been completed.

**CFTS[15:0] Bits**

These bits indicate the timestamp value of the message stored in the transmit/receive FIFO buffer.

These bits are valid when the CFM[1:0] value is 00<sub>B</sub>.

### 21.3.37 RSCAN0CFDF0k — Transmit/receive FIFO buffer Access Data Field 0 Register (k = 0 to 14)

**Access:** Can be read/written in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0E88<sub>H</sub> + (k \* 0010<sub>H</sub>)

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFDB3[7:0]								CFDB2[7:0]							
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFDB1[7:0]								CFDB0[7:0]							
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

**Table 21.51 RSCAN0CFDF0k register contents**

Bit Position	Bit Name	Function
b31 to b24	CFDB3[7:0]	Transmit/Receive FIFO Buffer Data Byte 3
b23 to b16	CFDB2[7:0]	Transmit/Receive FIFO Buffer Data Byte 2
b15 to b8	CFDB1[7:0]	Transmit/Receive FIFO Buffer Data Byte 0
b7 to b0	CFDB0[7:0]	<ul style="list-style-type: none"> <li>When CFM[1:0] value is 01<sub>B</sub> (transmit mode): Set the transmit/receive FIFO buffer data.</li> <li>When CFM[1:0] value is 00<sub>B</sub> (receive mode): The message data stored in the transmit/receive FIFO buffer can be read.</li> </ul>

This register is writable only when the CFM[1:0] value in the RSCAN0CFCK register is 01<sub>B</sub> (transmit mode). This register is readable only when the CFM[1:0] value is 00<sub>B</sub> (receive mode). When the CFDLC[3:0] value in the RSCAN0CFPTRk register is smaller than 1000<sub>B</sub>, data bytes for which no data is set are read as 00<sub>H</sub>. This register should not be read or written when the CFM[1:0] value is 10<sub>B</sub> (gateway mode).

### 21.3.38 RSCAN0CFDF1k — Transmit/receive FIFO buffer Access Data Field 1 Register (k = 0 to 14)

**Access:** Can be read/written in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0E8C<sub>H</sub> + (k \* 0010<sub>H</sub>)

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFDB7[7:0]								CFDB6[7:0]							
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFDB5[7:0]								CFDB4[7:0]							
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

**Table 21.52 RSCAN0CFDF1k register contents**

Bit Position	Bit Name	Function
b31 to b24	CFDB7[7:0]	Transmit/Receive FIFO Buffer Data Byte 7
b23 to b16	CFDB6[7:0]	Transmit/Receive FIFO Buffer Data Byte 6
b15 to b8	CFDB5[7:0]	Transmit/Receive FIFO Buffer Data Byte 5
b7 to b0	CFDB4[7:0]	Transmit/Receive FIFO Buffer Data Byte 4 <ul style="list-style-type: none"> <li>When CFM[1:0] value is 01<sub>B</sub> (transmit mode): Set the transmit/receive FIFO buffer data.</li> <li>When CFM[1:0] value is 00<sub>B</sub> (receive mode): The message data stored in the transmit/receive FIFO buffer can be read.</li> </ul>

This register is writable only when the CFM[1:0] value in the RSCAN0CFCK register is 01<sub>B</sub> (transmit mode). This register is readable only when the CFM[1:0] value is 00<sub>B</sub> (receive mode). When the CFDLC[3:0] value in the RSCAN0CFPTRk register is smaller than 1000<sub>B</sub>, data bytes for which no data is set are read as 00<sub>H</sub>. This register should not be read or written when the CFM[1:0] value is 10<sub>B</sub> (gateway mode).

### 21.3.39 RSCAN0FESTS — FIFO Empty Status Register

**Access:** Can be read in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0238<sub>H</sub>

**Initial value:** 007F FFFF<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	CF14E MP	CF13E MP	CF12E MP	CF11EM P	CF10E MP	CF9EM P	CF8EM P
Initial value	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CF7EM P	CF6EM P	CF5EM P	CF4EM P	CF3EM P	CF2EM P	CF1EM P	CF0EM P	RF7EM P	RF6EM P	RF5EM P	RF4EM P	RF3EM P	RF2EM P	RF1EM P	RF0EM P
Initial value	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

**Table 21.53 RSCAN0FESTS register contents**

Bit Position	Bit Name	Function
31 to 23	Reserved	These bits are always read as 0.
22	CF14EMP	Transmit/Receive FIFO Buffer Empty Status Flag
21	CF13EMP	0: Transmit/receive FIFO buffer k contains a message.
20	CF12EMP	1: Transmit/receive FIFO buffer k contains no message.
19	CF11EMP	(k = 0 to 14)
18	CF10EMP	
17	CF9EMP	
16	CF8EMP	
15	CF7EMP	
14	CF6EMP	
13	CF5EMP	
12	CF4EMP	
11	CF3EMP	
10	CF2EMP	
9	CF1EMP	
8	CF0EMP	
7	RF7EMP	Receive FIFO Buffer Empty Status Flag
6	RF6EMP	0: Receive FIFO buffer x contains an unread message.
5	RF5EMP	1: Receive FIFO buffer x contains no unread message (buffer empty).
4	RF4EMP	(x = 0 to 7)
3	RF3EMP	
2	RF2EMP	
1	RF1EMP	
0	RF0EMP	

The RSCAN0FESTS register is set to 007F FFFF<sub>H</sub> in global reset mode.

**CFkEMP Flag (k = 0 to 14)**

The CFkEMP flag is set to 1 when the CFEMP flag in the RSCAN0CFSTSk register is set to 1 (the transmit/receive FIFO buffer contains no message (buffer empty)). When the CFEMP flag is cleared to 0 (the transmit/receive FIFO buffer contains messages), the CFkEMP flag is cleared to 0.

**RFxEMP Flag (x = 0 to 7)**

The RFxEMP flag is set to 1 when the RFEMP flag in the RSCAN0RFSTSt register is set to 1 (the receive FIFO buffer contains no unread message (buffer empty)). When the RFEMP flag is cleared to 0 (the receive FIFO buffer contains unread messages), the RFxEMP flag is cleared to 0.

### 21.3.40 RSCAN0FFSTS — FIFO Full Status Register

**Access:** Can be read in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 023C<sub>H</sub>

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	CF14FL L	CF13FL L	CF12FL L	CF11FL L	CF10FL L	CF9FLL	CF8FLL
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CF7FLL	CF6FLL	CF5FLL	CF4FLL	CF3FLL	CF2FLL	CF1FLL	CF0FLL	RF7FLL	RF6FLL	RF5FLL	RF4FLL	RF3FLL	RF2FLL	RF1FLL	RF0FLL
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

**Table 21.54 RSCAN0FFSTS register contents**

Bit Position	Bit Name	Function
31 to 23	Reserved	These bits are always read as 0.
22	CF14FLL	Transmit/Receive FIFO Buffer Full Status Flag 0: Transmit/receive buffer k is not full. 1: Transmit/receive buffer k is full.
21	CF13FLL	(k = 0 to 14)
20	CF12FLL	
19	CF11FLL	
18	CF10FLL	
17	CF9FLL	
16	CF8FLL	
15	CF7FLL	
14	CF6FLL	
13	CF5FLL	
12	CF4FLL	
11	CF3FLL	
10	CF2FLL	
9	CF1FLL	
8	CF0FLL	
7	RF7FLL	Receive FIFO Buffer Full Status Flag 0: Receive FIFO buffer x is not full. 1: Receive FIFO buffer x is full.
6	RF6FLL	(x = 0 to 7)
5	RF5FLL	
4	RF4FLL	
3	RF3FLL	
2	RF2FLL	
1	RF1FLL	
0	RF0FLL	

The RSCAN0FFSTS register is cleared to 0000 0000<sub>H</sub> in global reset mode.



**CFkFLL Flag (k = 0 to 14)**

The CFkFLL flag is set to 1 when the CFFLL flag in the RSCAN0CFSTSk register is set to 1 (the transmit/receive FIFO buffer is full). When the CFFLL flag is cleared to 0 (the transmit/receive FIFO buffer is not full), the CFkFLL flag is cleared to 0.

**RFxFLL Flag (x = 0 to 7)**

The RFxFLL flag is set to 1 when the RFFLL flag in the RSCAN0RFSTSc register is set to 1 (the receive FIFO buffer is full). When the RFFLL flag is cleared to 0 (the receive FIFO buffer is not full), the RFxFLL flag is cleared to 0.

### 21.3.41 RSCAN0FMSTS — FIFO Message Lost Status Register

**Access:** Can be read in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0240<sub>H</sub>

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	CF14MLT	CF13MLT	CF12MLT	CF11MLT	CF10MLT	CF9MLT	CF8MLT
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CF7MLT	CF6MLT	CF5MLT	CF4MLT	CF3MLT	CF2MLT	CF1MLT	CF0MLT	RF7MLT	RF6MLT	RF5MLT	RF4MLT	RF3MLT	RF2MLT	RF1MLT	RF0MLT
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

**Table 21.55 RSCAN0FMSTS register contents**

Bit Position	Bit Name	Function
31 to 23	Reserved	These bits are always read as 0.
22	CF14MLT	Transmit/Receive FIFO Buffer Message Lost Status Flag 0: No transmit/receive FIFO buffer k message is lost. 1: A transmit/receive FIFO buffer k message is lost. (k = 0 to 14)
21	CF13MLT	
20	CF12MLT	
19	CF11MLT	
18	CF10MLT	
17	CF9MLT	
16	CF8MLT	
15	CF7MLT	
14	CF6MLT	
13	CF5MLT	
12	CF4MLT	
11	CF3MLT	
10	CF2MLT	
9	CF1MLT	
8	CF0MLT	
7	RF7MLT	Receive FIFO Buffer Message Lost Status Flag 0: No receive FIFO buffer x message is lost. 1: A receive FIFO buffer x message is lost. (x = 0 to 7)
6	RF6MLT	
5	RF5MLT	
4	RF4MLT	
3	RF3MLT	
2	RF2MLT	
1	RF1MLT	
0	RF0MLT	

The RSCAN0FMSTS register is cleared to 0000 0000<sub>H</sub> in global reset mode.

**CFkMLT Flag (k = 0 to 14)**

The CFkMLT flag is set to 1 when the CFMLT flag in the RSCAN0CFSTS<sub>k</sub> register is set to 1 (a transmit/receive FIFO message is lost). When the CFMLT flag is cleared to 0, the CFkMLT flag is cleared to 0.

**RFxMLT Flag (x = 0 to 7)**

The RFxMLT flag is set to 1 when the RFMLT flag in the RSCAN0RFSTS<sub>x</sub> register is set to 1 (a receive FIFO message is lost). When the RFMLT flag is cleared to 0, the RFxMLT flag is cleared to 0.

### 21.3.42 RSCAN0RFISTS — Receive FIFO Buffer Interrupt Flag Status Register

**Access:** Can be read in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0244<sub>H</sub>

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	RF7IF	RF6IF	RF5IF	RF4IF	RF3IF	RF2IF	RF1IF	RF0IF
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

**Table 21.56 RSCAN0RFISTS register contents**

Bit Position	Bit Name	Function
31 to 8	Reserved	These bits are always read as 0.
7	RF7IF	Receive FIFO Buffer Interrupt Request Status Flag 0: No receive FIFO buffer x interrupt request is present. 1: A receive FIFO buffer x interrupt request is present. (x = 0 to 7)
6	RF6IF	
5	RF5IF	
4	RF4IF	
3	RF3IF	
2	RF2IF	
1	RF1IF	
0	RF0IF	

The RSCAN0RFISTS register is cleared to 0000 0000<sub>H</sub> in global reset mode.

#### RFxIF Flag (x = 0 to 7)

The RFxIF flag is set to 1 when the RFIF flag in the RSCAN0RFISTSx register is set to 1 (a receive FIFO interrupt request is present). When the RFIF flag is cleared to 0, the RFxIF flag is cleared to 0.

### 21.3.43 RSCAN0CFRISTS — Transmit/receive FIFO buffer Receive Interrupt Flag Status Register

**Access:** Can be read in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0248<sub>H</sub>

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	CF14RXIF	CF13RXIF	CF12RXIF	CF11RXIF	CF10RXIF	CF9RXIF	CF8RXIF	CF7RXIF	CF6RXIF	CF5RXIF	CF4RXIF	CF3RXIF	CF2RXIF	CF1RXIF	CF0RXIF
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

**Table 21.57 RSCAN0CFRISTS register contents**

Bit Position	Bit Name	Function
31 to 15	Reserved	These bits are always read as 0.
14	CF14RXIF	Transmit/Receive FIFO Buffer Receive Interrupt Request Status Flag
13	CF13RXIF	0: No transmit/receive FIFO buffer k receive interrupt request is present.
12	CF12RXIF	1: A transmit/receive FIFO buffer k receive interrupt request is present.
11	CF11RXIF	(k = 0 to 14)
10	CF10RXIF	
9	CF9RXIF	
8	CF8RXIF	
7	CF7RXIF	
6	CF6RXIF	
5	CF5RXIF	
4	CF4RXIF	
3	CF3RXIF	
2	CF2RXIF	
1	CF1RXIF	
0	CF0RXIF	

The RSCAN0CFRISTS register is cleared to 0000 0000<sub>H</sub> in global reset mode.

#### CFkRXIF Flag (k = 0 to 14)

The CFkRXIF flag is set to 1 when the CFRXIF flag in the RSCAN0CFSTSk register is set to 1 (a transmit/receive FIFO receive interrupt request is present). When the CFRXIF flag is cleared to 0, the CFkRXIF flag is cleared to 0.

### 21.3.44 RSCAN0CFTISTS — Transmit/receive FIFO buffer Transmit Interrupt Flag Status Register

**Access:** Can be read in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 024C<sub>H</sub>

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	CF14TXIF	CF13TXIF	CF12TXIF	CF11TXIF	CF10TXIF	CF9TXIF	CF8TXIF	CF7TXIF	CF6TXIF	CF5TXIF	CF4TXIF	CF3TXIF	CF2TXIF	CF1TXIF	CF0TXIF
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

**Table 21.58 RSCAN0CFTISTS register contents**

Bit Position	Bit Name	Function
31 to 15	Reserved	These bits are always read as 0.
14	CF14TXIF	Transmit/Receive FIFO Buffer Transmit Interrupt Request Status Flag
13	CF13TXIF	0: No transmit/receive FIFO buffer k transmit interrupt request is present.
12	CF12TXIF	1: A transmit/receive FIFO buffer k transmit interrupt request is present.
11	CF11TXIF	(k = 0 to 14)
10	CF10TXIF	
9	CF9TXIF	
8	CF8TXIF	
7	CF7TXIF	
6	CF6TXIF	
5	CF5TXIF	
4	CF4TXIF	
3	CF3TXIF	
2	CF2TXIF	
1	CF1TXIF	
0	CF0TXIF	

The RSCAN0CFTISTS register is cleared to 0000 0000<sub>H</sub> in global reset mode.

#### CFkTXIF Flag (k = 0 to 14)

The CFkTXIF flag is set to 1 when the CFTXIF flag in the RSCAN0CFSTSk register is set to 1 (a transmit/receive FIFO transmit interrupt request is present). When the CFTXIF flag is cleared to 0, the CFkTXIF flag is cleared to 0.

### 21.3.45 RSCAN0TMCp — Transmit Buffer Control Register (p = 0 to 79)

**Access:** Can be read/written in 8-bit units

**Address:** <RSCAN0\_base> + 0250<sub>H</sub> + (01<sub>H</sub> × p)

**Initial value:** 00<sub>H</sub>

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	TMOM	TMTAR	TMTR
Initial value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W* <sup>1</sup>	R/W* <sup>1</sup>

Note 1. The only effective value for writing to this bit is 1, which sets the bit. Otherwise writing to the bit results in retention of its state.

**Table 21.59 RSCAN0TMCp register contents**

Bit Position	Bit Name	Function
7 to 3	Reserved	These bits are always read as 0. The write value should always be 0.
2	TMOM	One-Shot Transmission Enable 0: One-shot transmission is disabled. 1: One-shot transmission is enabled.
1	TMTAR	Transmit Abort Request 0: Transmit abort is not requested. 1: Transmit abort is requested.
0	TMTR	Transmit Request 0: Transmission is not requested. 1: Transmission is requested.

When the RSCAN0TMCp register meets any of the following conditions, set it to 00<sub>H</sub>.

- The RSCAN0TMCp register corresponds to the transmit buffer number selected by the CFTML[3:0] bits in the RSCAN0CFCCk register (p = m × 16 + the value of FTML[3:0] bits).
- The RSCAN0TMCp register corresponds to the transmit buffer allocated to the transmit queue by the TXQDC[3:0] bits in the RSCAN0TXQCCm register (m = 0 to 4) (p = (m × 16 + the value of TXQDC[3:0] bits) to (m × 16 + 0)).

Bits in the RSCAN0TMCp register are all cleared to 0 in channel reset mode. Modify the RSCAN0TMCp register in channel communication mode or channel halt mode.

#### TMOM Bit

Setting this bit to 1 enables one-shot transmission. When transmission fails, retransmission defined in the CAN protocol is not performed.

Modify the TMOM bit when the TMTRM flag in the RSCAN0TMSTSp register is set to 0. Set the TMOM bit to 1 together with the TMTR bit.

**TMTAR Bit**

Setting this bit to 1 generates a transmit abort request for the message stored in the transmit buffer. However, a message that is being transmitted or one that will be transmitted next cannot be aborted.

When the TMTAR bit can be set to 1 when TMTR bit is 1.

The TMTAR bit is cleared to 0 when any of the following conditions is met, but cannot be cleared by the program writing 0 to the bit.

- Transmission has been completed.
- Transmit abort has been completed.
- An error or arbitration loss has been detected.

If this bit becomes 0 at the same time as the program writes 1 to this bit, this bit becomes 0.

**TMTR Bit**

Setting this bit to 1 transmits the message stored in the transmit buffer.

The TMTR bit is cleared to 0 when any of the following conditions is met, but cannot be cleared by the program writing 0 to the bit.

- Transmission has been completed.
- Transmit abort has been completed after the TMTAR bit was set to 1.
- An error or arbitration-lost has been detected with the TMOM bit set to 1.

Set the TMTR bit to 1 when the value of TMTRF[1:0] in the RSCAN0TMSTSp register is 00<sub>B</sub>.



### 21.3.46 RSCAN0TMSTSp — Transmit Buffer Status Register (p = 0 to 79)

**Access:** Can be read/written in 8-bit units

**Address:** <RSCAN0\_base> + 02D0<sub>H</sub> + (01<sub>H</sub> × p)

**Initial value:** 00<sub>H</sub>

Bit	7	6	5	4	3	2	1	0
	—	—	—	TMTARM	TMTRM	TMTRF[1:0]		TMTSTS
Initial value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R

**Table 21.60 RSCAN0TMSTSp register contents**

Bit Position	Bit Name	Function
7 to 5	Reserved	These bits are always read as 0. The write value should always be 0.
4	TMTARM	Transmit Buffer Transmit Abort Request Status Flag 0: No transmit abort request is present. 1: A transmit abort request is present.
3	TMTRM	Transmit Buffer Transmit Request Status Flag 0: No transmit request is present. 1: A transmit request is present.
2, 1	TMTRF[1:0]	Transmit Buffer Transmit Result Status Flag b2 b1 0 0: Transmission is in progress or no transmit request is present. 0 1: Transmit abort has been completed. 1 0: Transmission has been completed (without transmit abort request). 1 1: Transmission has been completed (with transmit abort request).
0	TMTSTS	Transmit Buffer Transmit Status Flag 0: Transmission is not in progress. 1: Transmission is in progress.

The RSCAN0TMSTSp register is cleared to all 0 in channel reset mode.

#### TMTARM Flag

The TMTARM flag is set to 1 when the TMTAR bit in the RSCAN0TMCp register is set to 1.

The TMTARM flag is set to 0 when the TMTAR bit in the RSCAN0TMCp register is set to 0.

#### TMTRM Flag

The TMTRM flag is set to 1 when the TMTR bit in the RSCAN0TMCp register is set to 1.

The TMTRM flag is set to 0 when the TMTR bit in the RSCAN0TMCp register is set to 0

**TMTRF[1:0] Flag**

This flag indicates the result of transmission from the transmit buffer.

00<sub>B</sub>: Transmission is in progress or no transmit request is present.

01<sub>B</sub>: Transmission from the transmit buffer was aborted.

10<sub>B</sub>: Transmission has been completed with the TMTAR bit in the RSCAN0TMCp register set to 0 (transmit abort is not requested).

11<sub>B</sub>: Transmission has been completed with the TMTAR bit in the RSCAN0TMCp register set to 1 (transmit abort is requested).

Write 00<sub>B</sub> to the TMTRF[1:0] flag in channel communication mode or channel halt mode. Do not write any value other than 00<sub>B</sub> to this flag.

**TMTSTS Flag**

This flag is set to 1 when transmission from the transmit buffer starts, and is cleared to 0 when transmission from the transmit buffer has been completed or terminated due to a bus error or arbitration lost.

### 21.3.47 RSCAN0TMTRSTSy — Transmit Buffer Transmit Request Status Register y (y = 0 to 2)

**Access:** Can be read in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0350<sub>H</sub> + (y \* 0004<sub>H</sub>)

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTRSTSp (p = y × 32 + 31 to y × 32 + 16 (y = 0, 1))															
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTRSTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0, 1, 2))															
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

**Table 21.61 RSCAN0TMTRSTSy register contents**

Bit Position	Bit Name	Function
31 to 16	TMTRSTSp	Transmit Buffer Transmit Request Status Flag p (p = y × 32 + 31 to y × 32 + 16) 0: No transmission is requested. 1: Transmission is requested.
15 to 0	TMTRSTSp	Transmit Buffer Transmit Request Status Flag p (p = y × 32 + 15 to y × 32 + 0) 0: No transmit request is present. 1: A transmit request is present.

#### TMTRSTSp Flags (p = 0 to 79)

These flags indicate the status of the TMTR bit in the TMCp register.

When the TMTR bit is set to 1 (transmission is requested), the corresponding TMTRSTSp flag is set to 1.

The corresponding TMTRSTSp flag is cleared to 0 when the TMTR bit is set to 0 (transmission is not requested) or in channel reset mode.

Table 21.62 shows the bit assignment.

Table 21.62 TMTRSTSp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
.	.	.
15	0	15
16	1	0
.	.	.
30	1	14
31	1	15
32	2	0
33	2	1
.	.	.
47	2	15
48	3	0
.	.	.
62	3	14
63	3	15
64	4	0
65	4	1
.	.	.
78	4	14
79	4	15

### 21.3.48 RSCAN0TMTARSTSy — Transmit Buffer Transmit Abort Request Status Register y (y = 0 to 2)

**Access:** Can be read in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0360<sub>H</sub> + (y \* 0004<sub>H</sub>)

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TMTARSTSp (p = y × 32 + 31 to y × 32 + 16 (y = 0, 1))																
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMTARSTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0, 1, 2))																
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

**Table 21.63 RSCAN0TMTARSTSy register contents**

Bit Position	Bit Name	Function
31 to 16	TMTARSTSp	Transmit Buffer Transmit Abort Request Status Flag p (p = y × 32 + 31 to y × 32 + 16) 0: No transmission abort is requested. 1: Transmission abort is requested.
15 to 0	TMTARSTSp	Transmit Buffer Transmit Abort Request Status Flag p (p = y × 32 + 15 to y × 32 + 0) 0: No transmit abort request is present. 1: A transmit abort request is present.

#### TMTARSTSp Flags (p = 0 to 79)

These flags indicate the status of the TMTAR bit in the RSCAN0TMCp register.

When the TMTAR bit is set to 1 (transmit abort is requested), the corresponding TMTARSTSp flag is set to 1.

The corresponding TMTARSTSp flag is cleared to 0 when the TMTAR bit is set to 0 (transmit abort is not requested) or in channel reset mode.

Table 21.64 shows the bit assignment.

Table 21.64 TMTARSTSp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
.	.	.
15	0	15
16	1	0
.	.	.
30	1	14
31	1	15
32	2	0
33	2	1
.	.	.
47	2	15
48	3	0
.	.	.
62	3	14
63	3	15
64	4	0
65	4	1
.	.	.
78	4	14
79	4	15

### 21.3.49 RSCAN0TMCSTSy — Transmit Buffer Transmit Complete Status Register y (y = 0 to 2)

**Access:** Can be read in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0370<sub>H</sub> + (y \* 0004<sub>H</sub>)

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMCSTSp (p = y × 32 + 31 to y × 32 + 16 (y = 0, 1))															
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMCSTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0, 1, 2))															
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

**Table 21.65 RSCAN0TMCSTSy register contents**

Bit Position	Bit Name	Function
31 to 16	TMCSTSp	Transmit Buffer Transmit Complete Status Flag p (p = y × 32 + 31 to y × 32 + 16) 0: Transmission is not completed 1: Transmission is completed
15 to 0	TMCSTSp	Transmit Buffer Transmit Complete Status Flag p (p = y × 32 + 15 to y × 32 + 0) 0: Transmission has not been completed. 1: Transmission has been completed.

#### TMCSTSp Flags (p = 0 to 79)

When the TMTRF[1:0] flag in the RSCAN0TMSTSp register is set to 10<sub>B</sub> (transmission has been completed (without transmit abort request)) or 11<sub>B</sub> (transmission has been completed (with transmit abort request)), the corresponding TMCSTSp flag is set to 1.

A TMCSTSp flag is cleared to 0 when the corresponding TMTRF[1:0] flag is set to 00<sub>B</sub> or in channel reset mode.

Table 21.66 shows the bit assignment.

Table 21.66 TMCSTSp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
.	.	.
15	0	15
16	1	0
.	.	.
30	1	14
31	1	15
32	2	0
33	2	1
.	.	.
47	2	15
48	3	0
.	.	.
62	3	14
63	3	15
64	4	0
65	4	1
.	.	.
78	4	14
79	4	15



### 21.3.50 RSCAN0TMTASTSy — Transmit Buffer Transmit Abort Status Register y (y = 0 to 2)

**Access:** Can be read in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0380<sub>H</sub> + (y \* 0004<sub>H</sub>)

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TMTASTSp (p = y × 32 + 31 to y × 32 + 16 (y = 0, 1))																
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TMTASTSp (p = y × 32 + 15 to y × 32 + 0 (y = 0, 1, 2))																
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

**Table 21.67 RSCAN0TMTASTSy register contents**

Bit Position	Bit Name	Function
31 to 16	TMTASTSp	Transmit Buffer Transmit Abort Status Flag p (p = y × 32 + 31 to y × 32 + 16) 0: Transmission is not aborted 1: Transmission is aborted
15 to 0	TMTASTSp	Transmit Buffer Transmit Abort Status Flag p (p = y × 32 + 15 to y × 32 + 0) 0: Transmission is not aborted. 1: Transmission is aborted.

#### TMTASTSp Flags (p = 0 to 79)

When the TMTRF[1:0] flag in the RSCAN0TMSTSp register is set to 01<sub>B</sub> (transmit abort has been completed), the corresponding TMTASTSp flag is set to 1.

A TMTASTSp flag is cleared to 0 when the corresponding TMTRF[1:0] flag is set to 00<sub>B</sub> or in channel reset mode.

Table 21.68 shows the bit assignment.

Table 21.68 TMTASTSp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
.	.	.
15	0	15
16	1	0
.	.	.
30	1	14
31	1	15
32	2	0
33	2	1
.	.	.
47	2	15
48	3	0
.	.	.
62	3	14
63	3	15
64	4	0
65	4	1
.	.	.
78	4	14
79	4	15

### 21.3.51 RSCAN0TMIECy — Transmit Buffer Interrupt Enable Configuration Register y (y = 0 to 2)

**Access:** Can be read/written in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0390<sub>H</sub> + (y \* 0004<sub>H</sub>)

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMIEp (p = y × 32 + 16 to y × 32 + 31 (y = 0, 1))															
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMIEp (p = y × 32 + 0 to y × 32 + 15 (y = 0, 1, 2))															
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

**Table 21.69 RSCAN0TMIECy register contents**

Bit Position	Bit Name	Function
31 to 16	TMIEp	Transmit Buffer Interrupt Enable Bit p (p = y × 32 + 31 to y × 32 + 16) 0: Transmit buffer interrupt is disabled 1: Transmit buffer interrupt is enabled
15 to 0	TMIEp	Transmit Buffer Interrupt Enable Bit p (p = y × 32 + 15 to y × 32 + 0) 0: Transmit buffer interrupt is disabled. 1: Transmit buffer interrupt is enabled.

#### TMIEp Bits (p = 0 to 79)

When any of these bits is set to 1 and the corresponding transmission has been completed, a transmit buffer interrupt request is generated.

Modify these bits when the TMTRM flag in the corresponding RSCAN0TMSTSp register is 0 (no transmit request is present).

Write 0 to bits corresponding to transmit buffers linked to transmit/receive FIFO buffers or transmit buffers allocated to the transmit queue.

Table 21.70 shows the bit assignment.

Table 21.70 TMIEp Bit Assignment

Bit	Channel	Transmit Buffer Number
0	0	0
1	0	1
.	.	.
15	0	15
16	1	0
.	.	.
30	1	14
31	1	15
32	2	0
33	2	1
.	.	.
47	2	15
48	3	0
.	.	.
62	3	14
63	3	15
64	4	0
65	4	1
.	.	.
78	4	14
79	4	15

### 21.3.52 RSCAN0TMIDp — Transmit Buffer ID Register (p = 0 to 79)

**Access:** Can be read/written in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 1000<sub>H</sub> + (p \* 0010<sub>H</sub>)

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMIDE	TMRTR	THLEN	TMID[28:16]												
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMID[15:0]															
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

**Table 21.71 RSCAN0TMIDp register contents**

Bit Position	Bit Name	Function
31	TMIDE	Transmit Buffer IDE 0: Standard ID 1: Extended ID
30	TMRTR	Transmit Buffer RTR 0: Data frame 1: Remote frame
29	THLEN	Transmit History Data Store Enable 0: Transmit history data is not stored in the buffer. 1: Transmit history data is stored in the buffer.
28 to 0	TMID[28:0]	Transmit Buffer ID Data Set standard ID or extended ID. For standard ID, write an ID to bits 10 to 0 and write 0 to bits 28 to 11.

Modify this register when the TMTRM bit in the corresponding RSCAN0TMSTSp register is set to 0 (no transmit request is present). If this register is linked to a transmit/receive FIFO buffer, do not write data to this register. If this register is allocated to the transmit queue, only write data to a transmit buffer p (p = m × 16 + 15) for the corresponding channel.

#### TMIDE Bit

This bit is used to set the ID format of the message to be transmitted from the transmit buffer.

#### TMRTR Bit

This bit is used to set the data format of the message to be transmitted from the transmit buffer.

**THLEN Bit**

When this bit is set to 1, the transmit history data of the transmit message (the label information and the number and type of the transmit buffer) are stored in the transmit history buffer after transmission is completed.

**TMID[28:0] Bits**

These bits are used to set the ID of the message to be transmitted from the transmit buffer.

### 21.3.53 RSCAN0TMPTRp — Transmit Buffer Pointer Register (p= 0 to 79)

**Access:** Can be read/written in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 1004<sub>H</sub> + (p \* 0010<sub>H</sub>)

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMDLC[3:0]				—	—	—	—	TMPTR[7:0]							
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

**Table 21.72 RSCAN0TMPTRp register contents**

Bit Position	Bit Name	Function
31 to 28	TMDLC[3:0]	Transmit Buffer DLC Data b31 b30 b29 b28 0 0 0 0: 0 data bytes 0 0 0 1: 1 data byte 0 0 1 0: 2 data bytes 0 0 1 1: 3 data bytes 0 1 0 0: 4 data bytes 0 1 0 1: 5 data bytes 0 1 1 0: 6 data bytes 0 1 1 1: 7 data bytes 1 X X X: 8 data bytes
27 to 24	Reserved	These bits are always read as 0. The write value should always be 0.
23 to 16	TMPTR[7:0]	Transmit Buffer Label Data Set the label information to be stored in the transmit history buffer.
15 to 0	Reserved	These bits are always read as 0. The write value should always be 0.

Modify this register when the TMTRM bit in the corresponding RSCAN0TMSTSp register is set to 0 (no transmit request is present). If this register is linked to a transmit/receive FIFO buffer, do not write to this register. If this register is allocated to the transmit queue, only write to a transmit buffer p (p = m × 16 + 15) for the corresponding channel.

#### TMDLC[3:0] Bits

These bits are used to set the data length of the message to be transmitted from the transmit buffer when the TMRTR bit in the RSCAN0TMIDp register is set to 0 (data frame). If the data length is set to 9 bytes or more, the transmit data is 8 bytes long.

When the TMRTR bit is set to 1 (remote frame), set the data length of messages to be requested.

#### TMPTR[7:0] Bits

When message transmission has been completed, the TMPTR[7:0] value is stored in the transmit history buffer.

### 21.3.54 RSCAN0TMDF0p — Transmit Buffer Data Field 0 Register (p = 0 to 79)

**Access:** Can be read/written in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 1008<sub>H</sub> + (p \* 0010<sub>H</sub>)

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMDB3[7:0]								TMDB2[7:0]							
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMDB1[7:0]								TMDB0[7:0]							
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

**Table 21.73 RSCAN0TMDF0p register contents**

Bit Position	Bit Name	Function
31 to 24	TMDB3[7:0]	Transmit Buffer Data Byte 3
23 to 16	TMDB2[7:0]	Transmit Buffer Data Byte 2
15 to 8	TMDB1[7:0]	Transmit Buffer Data Byte 0
7 to 0	TMDB0[7:0]	Set the transmit buffer data.

Modify this register when the TMTRM bit in the corresponding RSCAN0TMSTSp register is set to 0 (no transmit request is present). If this register is linked to a transmit/receive FIFO buffer, do not write to this register. If this register is allocated to the transmit queue, only write to a transmit buffer p (p = m × 16 + 15) for the corresponding channel.



### 21.3.55 RSCAN0TMDF1p — Transmit Buffer Data Field 1 Register (p = 0 to 79)

**Access:** Can be read/written in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 100C<sub>H</sub> + (p \* 0010<sub>H</sub>)

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMDB7[7:0]								TMDB6[7:0]							
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMDB5[7:0]								TMDB4[7:0]							
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

**Table 21.74 RSCAN0TMDF1p register contents**

Bit Position	Bit Name	Function
31 to 24	TMDB7[7:0]	Transmit Buffer Data Byte 7
23 to 16	TMDB6[7:0]	Transmit Buffer Data Byte 6
15 to 8	TMDB5[7:0]	Transmit Buffer Data Byte 5
7 to 0	TMDB4[7:0]	Transmit Buffer Data Byte 4 Set the transmit buffer data.

Modify this register when the TMTRM bit in the corresponding RSCAN0TMSTSp register is set to 0 (no transmit request is present). If this register is linked to a transmit/receive FIFO buffer, do not write to this register. If this register is allocated to the transmit queue, only write to a transmit buffer p (p = m × 16 + 15) for the corresponding channel.

### 21.3.56 RSCAN0TXQCCm — Transmit Queue Configuration and Control Register (m = 0 to 4)

**Access:** Can be read/written in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 03A0<sub>H</sub> + (m \* 0010<sub>H</sub>)

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	TXQIM	TXQIE	TXQDC[3:0]			—	—	—	—	—	—	—	—	TXQE
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R/W

**Table 21.75 RSCAN0TXQCCm register contents**

Bit Position	Bit Name	Function
31 to 14	Reserved	These bits are always read as 0. The write value should always be 0.
13	TXQIM	Transmit Queue Interrupt Source Select 0: When the buffer becomes empty upon completion of message transmission, a transmit queue interrupt request is generated. 1: A transmit queue interrupt request is generated each time a message has been transmitted.
12	TXQIE	Transmit Queue Interrupt Enable 0: Transmit queue interrupt is disabled. 1: Transmit queue interrupt is enabled.
11 to 8	TXQDC[3:0]	Transmit Queue Depth Configuration Setting these bits to g (g = 2 to 15) makes the (g + 1)-buffer transmit queue available. Setting these bits to 0 disables the transmit queue. Setting these bits to 1 is prohibited.
7 to 1	Reserved	These bits are always read as 0. The write value should always be 0.
0	TXQE	Transmit Queue Enable 0: The transmit queue is not used. 1: The transmit queue is used.

#### TXQIM Bit

This bit is used to select a transmit queue interrupt source. Modify this bit in channel reset mode.

#### TXQIE Bit

When the TXQIE bit is set to 1 and the source selected by the TXQIM bit occurs, an interrupt request is generated.

Set the TXQE bit to 0 before modifying the TXQIE bit.

**TXQDC[3:0] Bits**

These bits are used to specify the number of transmit buffers to be allocated to the transmit queues. Transmit buffers are allocated to transmit queues in descending order of buffer number, that is, from  $(m \times 16 + 15)$  to  $(m \times 16 + 0)$ . For examples of how buffer allocation is done, see **Figure 21.9**. Modify these bits only in channel reset mode.

**TXQE Bit**

Setting this bit to 1 makes the transmit queue available. Modify this bit in channel communication mode or channel halt mode. This bit is cleared to 0 in channel reset mode.

Before setting the TXQE bit to 1, set the TXQDC[3:0] bits to  $0010_B$  or more.

### 21.3.57 RSCAN0TXQSTSm — Transmit Queue Status Register (m = 0 to 4)

**Access:** Can be read/written in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 03C0<sub>H</sub> + (m \* 0004<sub>H</sub>)

**Initial value:** 0000 0001<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	TXQIF	TXQFLL	TXQEMP
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W*1	R	R

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

**Table 21.76 RSCAN0TXQSTSm register contents**

Bit Position	Bit Name	Function
31 to 3	Reserved	Values read from these bits are undefined. The write value should always be 0.
2	TXQIF	Transmit Queue Interrupt Request Flag 0: No transmit queue interrupt request is present. 1: A transmit queue interrupt request is present.
1	TXQFLL	Transmit Queue Full Status Flag 0: The transmit queue is not full. 1: The transmit queue is full.
0	TXQEMP	Transmit Queue Empty Status Flag 0: The transmit queue contains messages. 1: The transmit queue contains no message (transmit queue empty).

#### TXQIF Flag

The TXQIF flag is set to 1 when the event specified by the TXQIM bit in the RSCAN0TXQCCm register has occurred.

The TXQIF flag is cleared to 0 in channel reset mode or by writing 0 to this flag. This flag is not cleared to 0 by setting the TXQE bit in the RSCAN0TXQCCm register to 0 (the transmit queue is not used).

#### TXQFLL Flag

The TXQFLL flag is set to 1 when the number of messages set for the transmit queue matches the transmit queue depth set by the TXQDC[3:0] bits in the RSCAN0TXQCCm register.

This flag is cleared to 0 in any of the following cases.

- The number of messages set for the transmit queue is smaller than the transmit queue depth set by the TXQDC[3:0] bits.
- In channel reset mode

**TXQEMP Flag**

The TXQEMP flag is cleared to 0 when even a single message is pending in the transmit queue.

This flag is set to 1 in any of the following cases.

- The TXQE bit is set to 0 (the transmit queue is not used).
- The transmit queue becomes empty.
- In channel reset mode

### 21.3.58 RSCAN0TXQPCTRM — Transmit Queue Pointer Control Register (m = 0 to 4)

**Access:** Can be written in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 03E0<sub>H</sub> + (m \* 0004<sub>H</sub>)

**Initial value:** 0000 0001<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TXQPC[7:0]							
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

**Table 21.77 RSCAN0TXQPCTRM register contents**

Bit Position	Bit Name	Function
31 to 8	Reserved	The write value should always be 0.
7 to 0	TXQPC[7:0]	Transmit Queue Pointer Control Writing FF <sub>H</sub> to these bits moves the write pointer of the transmit queue to the next queue buffer. Set Value: FF <sub>H</sub>

#### TXQPC[7:0] Bits

Writing FF<sub>H</sub> to the TXQPC[7:0] bits moves the write pointer to the next transmit queue buffer and generates a transmit request of the message. Write transmit messages to the RSCAN0TMID<sub>p</sub>, RSCAN0TMPTR<sub>p</sub>, RSCAN0TMDF0<sub>p</sub>, and RSCAN0TMDF1<sub>p</sub> registers (p = 15, 31, 47, 63, and 79) before writing FF<sub>H</sub> to the TXQPC[7:0] bits.

Write FF<sub>H</sub> only when the TXQE bit in the RSCAN0TXQCC<sub>m</sub> register set to 1 (the transmit queue is used) and the TXQFLL flag in the RSCAN0TXQSTS<sub>m</sub> register set to 0 (not full).

### 21.3.59 RSCAN0THLCCm — Transmit History Configuration and Control Register (m = 0 to 4)

**Access:** Can be read/written in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0400<sub>H</sub> + (m \* 0004<sub>H</sub>)

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	THLDTE	THLIM	THLIE	—	—	—	—	—	—	—	THLE
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R	R	R/W

**Table 21.78 RSCAN0THLCCm register contents**

Bit Position	Bit Name	Function
31 to 11	Reserved	These bits are always read as 0. The write value should always be 0.
10	THLDTE	Transmit History Target Buffer Select 0: Entry from transmit/receive FIFO buffers and transmit queue 1: Entry from transmit buffers, transmit/receive FIFO buffers, and transmit queue
9	THLIM	Transmit History Interrupt Source Select 0: When 12 sets of data have been stored in the transmit history buffer 1: When a single set of transmit history data has been stored
8	THLIE	Transmit History Interrupt Enable 0: Transmit history interrupt is disabled. 1: Transmit history interrupt is enabled.
7 to 1	Reserved	These bits are always read as 0. The write value should always be 0.
0	THLE	Transmit History Buffer Enable 0: Transmit history buffer is not used. 1: Transmit history buffer is used.

#### THLDTE Bit

When this bit is set to 0, the transmit history data of messages transmitted from transmit/receive FIFO buffers and the transmit queue is stored in the transmit history buffer. When this bit is set to 1, the transmit history data of messages transmitted from transmit buffers, transmit/receive FIFO buffers, and the transmit queue is stored in the transmit history buffer.

Modify this bit only in channel reset mode.

**THLIM Bit**

This bit is used to select a transmit history interrupt source.

Modify this bit only in channel reset mode.

**THLIE Bit**

When the THLIE bit is set to 1 and the source selected by the THLIM bit has occurred, a transmit history interrupt request is generated. Modify the THLIE bit only when the THLE bit set to 0.

**THLE Bit**

Setting this bit to 1 makes the transmit history buffer available. When data transmission from the buffer selected by the THLDTE bit has been completed, the transmit history data of transmit messages is stored in the transmit history buffer.

Modify this bit in channel communication mode or channel halt mode.



### 21.3.60 RSCAN0THLSTSm — Transmit History Status Register (m = 0 to 4)

**Access:** Can be read in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0420<sub>H</sub> + (m \* 0004<sub>H</sub>)

**Initial value:** 0000 0001<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	THLMC[4:0]				—	—	—	—	THLIF	THLELT	THLFLL	THLEMP	
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W*1	R/W*1	R	R

Note 1. The only effective value for writing to this flag bit is 0, which clears the bit. Otherwise writing to the bit results in retention of its state.

**Table 21.79 RSCAN0THLSTSm register contents**

Bit Position	Bit Name	Function
31 to 13	Reserved	These bits are always read as 0. The write value should always be 0.
12 to 8	THLMC[4:0]	Transmit History Buffer Unread Data Counter These bits indicate the number of unread data sets stored in the transmit history buffer.
7 to 4	Reserved	These bits are always read as 0. The write value should always be 0.
3	THLIF	Transmit History Interrupt Request Flag 0: No transmit history interrupt request is present. 1: A transmit history interrupt request is present.
2	THLELT	Transmit History Buffer Overflow Flag 0: Transmit history buffer overflow has not occurred. 1: Transmit history buffer overflow has occurred.
1	THLFLL	Transmit history Buffer Full Status Flag 0: Transmit history buffer is not full. 1: Transmit history buffer is full.
0	THLEMP	Transmit History Buffer Empty Status Flag 0: Transmit history buffer contains unread data. 1: Transmit history buffer contains no unread data (buffer empty).

#### THLMC[4:0] Bits

These bits indicate the number of unread data sets stored in the transmit history buffer.

#### THLIF Flag

The THLIF flag is set to 1 when the interrupt source specified with the THLIM bit in the RSCAN0THLCCm register occurs.

This flag is cleared to 0 in channel reset mode or by the program writing 0 to this flag.

**THLELT Flag**

The THLELT flag is set to 1 when an attempt is made to store new transmit history data while the transmit history buffer is full. In this case, the new data is discarded. This flag becomes 0 in channel reset mode or by the program writing 0 to this flag.

**THLFLL Flag**

The THLFLL flag is set to 1 when 16 data sets have been stored in the transmit history buffer, and is cleared to 0 when the number of data sets stored in the transmit history buffer has decreased to less than 16. This bit is also cleared to 0 in channel reset mode or when the THLE bit in the RSCAN0THLCCm register is set to 0 (transmit history buffer is not used).

**THLEMP Flag**

The THLEMP flag is cleared to 0 when even a single set of transmit history data has been stored in the transmit history buffer.

This flag is set to 1 when all the data in the transmit history buffer has been read. This flag is also set to 1 in channel reset mode or when the THLE bit in the RSCAN0THLCCm register is set to 0 (transmit history buffer is not used).

### 21.3.61 RSCAN0THLACCm — Transmit History Access Register (m = 0 to 4)

**Access:** Can be read in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 1800<sub>H</sub> + (m \* 0004<sub>H</sub>)

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TID[7:0]							—	BN[3:0]			BT[2:0]				
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

**Table 21.80 RSCAN0THLACCm register contents**

Bit Position	Bit Name	Function																
31 to 16	Reserved	These bits are always read as 0.																
15 to 8	TID[7:0]	Label Data The label information of stored data can be read.																
7	Reserved	This bit is always read as 0.																
6 to 3	BN[3:0]	Buffer Number Data The buffer number of transmit source (transmit buffer, transmit/receive FIFO or transmit queue) can be read.																
2 to 0	BT[2:0]	Buffer Type Data <table border="0"> <tr> <td>b2</td><td>b1</td><td>b0</td><td></td> </tr> <tr> <td>0</td><td>0</td><td>1</td><td>: Transmit buffer</td> </tr> <tr> <td>0</td><td>1</td><td>0</td><td>: Transmit FIFO buffer</td> </tr> <tr> <td>1</td><td>0</td><td>0</td><td>: Transmit queue</td> </tr> </table>	b2	b1	b0		0	0	1	: Transmit buffer	0	1	0	: Transmit FIFO buffer	1	0	0	: Transmit queue
b2	b1	b0																
0	0	1	: Transmit buffer															
0	1	0	: Transmit FIFO buffer															
1	0	0	: Transmit queue															

#### TID[7:0] Bits

These bits indicate the label information of transmit history data stored in the transmit history buffer.

#### BN[3:0] Bits

These bits indicate the transmit history data stored in the transmit history buffer.

#### BT[2:0] Bits

These bits indicate the transmit history data stored in the transmit history buffer.

### 21.3.62 RSCAN0THLPCTRm — Transmit History Pointer Control Register (m = 0 to 4)

**Access:** Can be written in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0440<sub>H</sub> + (m \* 0004<sub>H</sub>)

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	THLPC[7:0]							
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

**Table 21.81 RSCAN0THLPCTRm register contents**

Bit Position	Bit Name	Function
31 to 8	Reserved	The write value should always be 0.
7 to 0	THLPC[7:0]	Transmit History List Pointer Control Writing FF <sub>H</sub> to these bits moves the read pointer to the next unread data in the transmit history buffer. Set Value: FF <sub>H</sub>

#### THLPC[7:0] Bits

When the THLPC[7:0] bits are set to FF<sub>H</sub>, the read pointer moves to the next data in the transmit history buffer. At this time, the THLMC[4:0] (transmit history buffer unread data counter) value in the RSCAN0THLSTSm register is decremented. Write FF<sub>H</sub> to the THLPC[7:0] bits after reading from the RSCAN0THLACCm register.

Write FF<sub>H</sub> only when the THLE bit in the RSCAN0THLCCm register is set to 1 (transmit history buffer is used) and the THLEMP flag in the RSCAN0THLSTSm register is 0.

### 21.3.63 RSCAN0GTSTCFG — Global Test Configuration Register

**Access:** Can be read/written in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 0468<sub>H</sub>

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	C4ICBCE	C3ICBCE	C2ICBCE	C1ICBCE	C0ICBCE
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

**Table 21.82 RSCAN0GTSTCFG register contents**

Bit Position	Bit Name	Function
31 to 5	Reserved	These bits are always read as 0. The write value should always be 0.
4	C4ICBCE	RSCAN4 Inter-channel Communication Test Enable 0: RSCAN4 inter-channel communication test is disabled. 1: RSCAN4 inter-channel communication test is enabled.
3	C3ICBCE	RSCAN3 Inter-channel Communication Test Enable 0: RSCAN3 inter-channel communication test is disabled. 1: RSCAN3 inter-channel communication test is enabled.
2	C2ICBCE	RSCAN2 Inter-channel Communication Test Enable 0: RSCAN2 inter-channel communication test is disabled. 1: RSCAN2 inter-channel communication test is enabled.
1	C1ICBCE	CAN1 Inter-Channel Communication Test Enable 0: CAN1 inter-channel communication test is disabled. 1: CAN1 inter-channel communication test is enabled.
0	C0ICBCE	CAN0 Inter-Channel Communication Test Enable 0: CAN0 inter-channel communication test is disabled. 1: CAN0 inter-channel communication test is enabled.

Modify the RSCAN0GTSTCFG register only in global test mode.

#### **C4ICBCE Bit**

Setting this bit to 1 enables the channel 4 inter-channel communication test.

#### **C3ICBCE Bit**

Setting this bit to 1 enables the channel 3 inter-channel communication test.

#### **C2ICBCE Bit**

Setting this bit to 1 enables the channel 2 inter-channel communication test.

#### **C1ICBCE Bit**

Setting this bit to 1 enables the channel 1 inter-channel communication test.

**C0ICBCE Bit**

Setting this bit to 1 enables the channel 0 inter-channel communication test.

### 21.3.64 RSCAN0GTSTCTR — Global Test Control Register

**Access:** Can be read/written in 8-, 16-, and 32-bit units

**Address:** <RSCAN0\_base> + 046C<sub>H</sub>

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ICBCTM E
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

**Table 21.83 RSCAN0GTSTCTR register contents**

Bit Position	Bit Name	Function
31 to 1	Reserved	These bits are always read as 0. The write value should always be 0.
0	ICBCTME	Communication Test between Channels Enable 0: Communication test between channels disabled 1: Communication test between channels enabled

#### ICBCTME Bit

When this bit is set to 1, a communication test is enabled between the channels for which the CmICBCE bit (m = 0 to 4) in the RSCAN0GTSTCFG register has been set to 1. Modify the ICBCTME bit only in global test mode.

### 21.3.65 RSCAN0GLOCKK — Global Lock Key Register

**Access:** Can be written in 16- and 32-bit units

**Address:** <RSCAN0\_base> + 047C<sub>H</sub>

**Initial value:** 0000 0000<sub>H</sub>

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LOCK[15:0]															
Initial value	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1

Note 1. Writing to these bits is effective only when the RS-CAN module is in global test mode.

**Table 21.84 RSCAN0GLOCKK register contents**

Bit Position	Bit Name	Function
31 to 16	Reserved	When read, an undefined value is returned. The write value should be 0.
15 to 0	LOCK[15:0]	Lock Key These bits are key bits to release protection of test mode.

The RSCAN0GLOCKK register releases protection of special test bits and is write-only.

#### LOCK[15:0] Bits

Writing the protection release data to the LOCK[15:0] bits in succession enables writing 1 to the RTME bit in the RSCAN0GTSTCTR register.

After the protection has been released, writing to the I/O register area (<RCAN0\_base> + 0000<sub>H</sub> to <RCAN0\_base> + 04FF<sub>H</sub>) of the CAN (except the RAM) enables the protection again.

Reading from the I/O register area of the CAN or reading from/writing to other areas does not enable the protection.



## 21.4 Interrupt Sources

The RS-CAN module has 17 interrupts that are grouped into global interrupts and channel interrupts.

Global interrupts [2 sources: common among channels]:

- Receive FIFO interrupt [1 source: common among channels]
- Global error interrupt [1 source: common among channels]

Channel interrupts [15 sources: 3 sources × number of channels]:

- CANm transmit interrupt [1 source for each channel]
  - CANm transmit complete interrupt
  - CANm transmit abort interrupt
  - CANm transmit/receive FIFO transmit complete interrupt (in transmit mode, gateway mode)
  - CANm transmit history interrupt
  - CANm transmit queue Interrupt

CANm transmit/receive FIFO receive complete interrupt (in transmit mode, gateway mode)  
[1 source for each channel]

CANm error interrupt [1 source for each channel]

(m = 0 to 4)

When an interrupt request is generated, the corresponding interrupt request flag is set to 1 (interrupt request present). In that case, when the interrupt enable bit is set to 1 (enabling interrupts), an interrupt request is output from the RS-CAN module. (Generation of interrupts also depends on the interrupt control register settings of the interrupt controller.)

Setting the interrupt request flag to 0 (no interrupt request present) or setting the interrupt enable bit to 0 (disabling interrupts) clears the current interrupt request. The current interrupt request is still output until the interrupt request flag is cleared.

**Table 21.85** lists the CAN interrupt sources. **Figure 21.2** shows the CAN global interrupt block diagram. **Figure 21.3** shows the CAN channel interrupt block diagram.

Table 21.85 List of CAN Interrupt Sources

Interrupt Source		Corresponding Interrupt Request Flag	Corresponding Interrupt Enable Bit	
Global interrupts	Receive FIFO	Receive FIFO 0	RFIF in the RSCAN0RFSTS0 register	RFIE in the RSCAN0RFCC0 register
		Receive FIFO 1	RFIF in the RSCAN0RFSTS1 register	RFIE in the RSCAN0RFCC1 register
		Receive FIFO 2	RFIF in the RSCAN0RFSTS2 register	RFIE in the RSCAN0RFCC2 register
		Receive FIFO 3	RFIF in the RSCAN0RFSTS3 register	RFIE in the RSCAN0RFCC3 register
		Receive FIFO 4	RFIF in the RSCAN0RFSTS4 register	RFIE in the RSCAN0RFCC4 register
		Receive FIFO 5	RFIF in the RSCAN0RFSTS5 register	RFIE in the RSCAN0RFCC5 register
		Receive FIFO 6	RFIF in the RSCAN0RFSTS6 register	RFIE in the RSCAN0RFCC6 register
		Receive FIFO 7	RFIF in the RSCAN0RFSTS7 register	RFIE in the RSCAN0RFCC7 register
Global error		<ul style="list-style-type: none"> <li>• DEF in the RSCAN0GERFL register</li> <li>• MES in the RSCAN0GERFL register</li> <li>• THLES in the RSCAN0GERFL register</li> </ul>	<ul style="list-style-type: none"> <li>• DEIE in the RSCAN0GCTR register</li> <li>• MEIE in the RSCAN0GCTR register</li> <li>• THLEIE in the RSCAN0GCTR register</li> </ul>	
Channel interrupts (m = 0 to 4)	CANm transmit	CANm transmit complete	TMTRF[1:0] in the RSCAN0TMSTSp register	TMIE in the RSCAN0TMIECy register
		CANm transmit abort	TMTRF[1:0] in the RSCAN0TMSTSp register	TAIE in the RSCAN0CmCTR register
		CANm transmit/receive FIFO transmit	CFTXIF in the RSCAN0CFSTSk register	CFTXIE in the RSCAN0CFCCk register
		CANm transmit queue	TXQIF in the RSCAN0TXQSTSn register	TXQIE in the RSCAN0TXQCCn register
		CANm transmit history	THLIF in the RSCAN0THLSTSn register	THLIE in the RSCAN0THLCCn register
		CANm transmit/receive FIFO receive	CFRXIF in the RSCAN0CFSTSk register	CFRXIE in the RSCAN0CFCCk register
		CANm error	<ul style="list-style-type: none"> <li>• BEF in the RSCAN0CmERFL register</li> <li>• ALF in the RSCAN0CmERFL register</li> <li>• BLF in the RSCAN0CmERFL register</li> <li>• OVLF in the RSCAN0CmERFL register</li> <li>• BORF in the RSCAN0CmERFL register</li> <li>• BOEF in the RSCAN0CmERFL register</li> <li>• EPF in the RSCAN0CmERFL register</li> <li>• EWF in the RSCAN0CmERFL register</li> </ul>	<ul style="list-style-type: none"> <li>• BEIE in the RSCAN0CmCTR register</li> <li>• ALIE in the RSCAN0CmCTR register</li> <li>• BLIE in the RSCAN0CmCTR register</li> <li>• OLIE in the RSCAN0CmCTR register</li> <li>• BORIE in the RSCAN0CmCTR register</li> <li>• BOEIE in the RSCAN0CmCTR register</li> <li>• EPIE in the RSCAN0CmCTR register</li> <li>• EWIE in the RSCAN0CmCTR register</li> </ul>

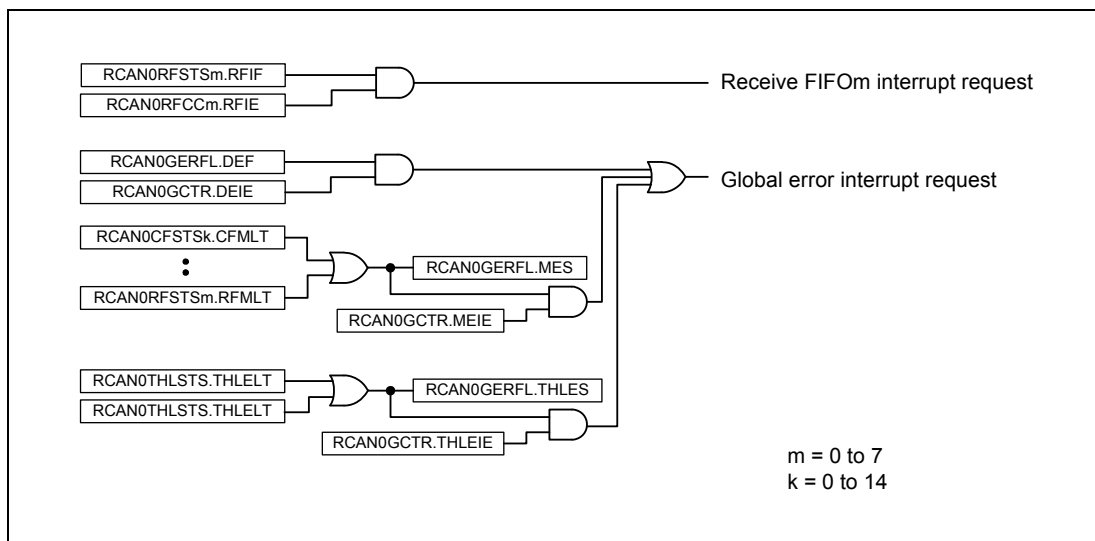


Figure 21.2 CAN Global Interrupt Block Diagram

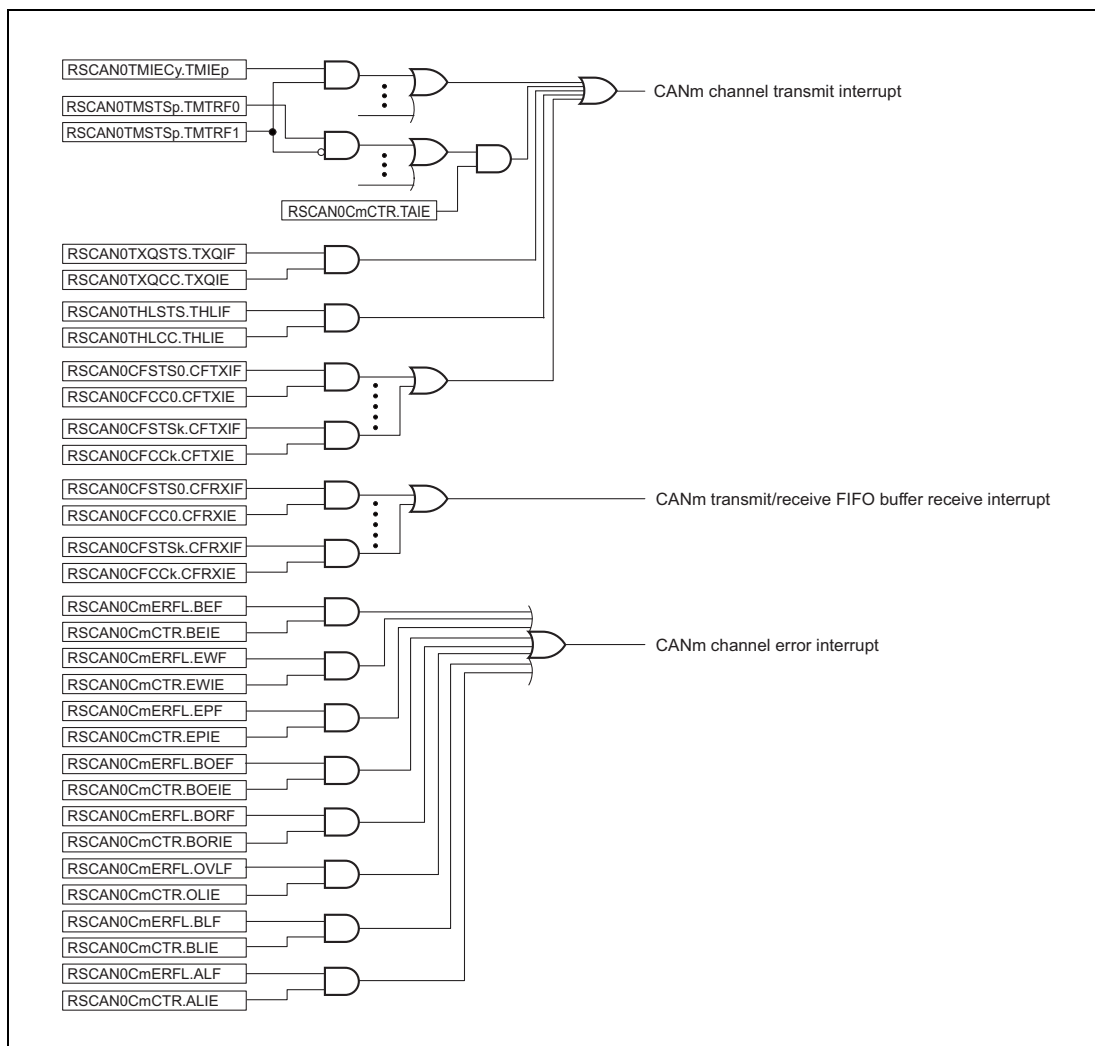


Figure 21.3 CAN Channel Interrupt Block Diagram

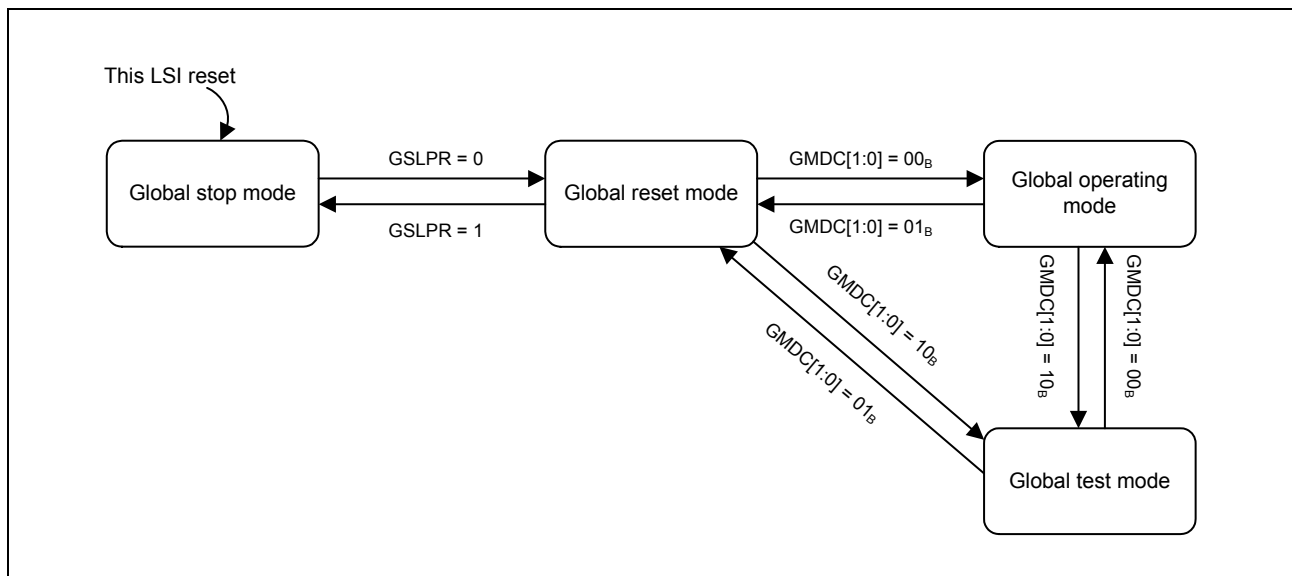
## 21.5 RSCAN Modes

The RS-CAN module has four global modes to control the entire RS-CAN module status and four channel modes to control individual channel status. Details of global modes are described in **Section 21.5.1, Global Modes**, and details of channel modes are described in **Section 21.5.2, Channel Modes**.

- Global stop mode: Stops the clocks of the entire module to achieve low power consumption.
- Global reset mode: Performs initial settings for the entire module.
- Global test mode: Performs test settings.
- Global operating mode: Makes the entire module operable.
- Channel stop mode: Stops the channel clock.
- Channel reset mode: Performs initial settings for the channels.
- Channel halt mode: Stops CAN communication and allows channel testing.
- Channel communication mode: Performs CAN communication.

### 21.5.1 Global Modes

**Figure 21.4** shows the transitions of global modes.



**Figure 21.4** Transitions of Global Modes

In some cases, global mode transitions also force channel mode transitions. **Table 21.86** shows the channel mode transitions depending on the global mode setting dictated by the GMDC[1:0] bits and the GSLPR bit.

**Table 21.86** Transitions of Channel Modes Depending on Global Mode Setting (GMDC[1:0] and GSLPR Bits)

Channel Mode before Setting	Channel Mode after Setting			
	GMDC[1:0] = 00 <sub>B</sub> GSLPR = 0 (Global Operation)	GMDC[1:0] = 10 <sub>B</sub> GSLPR = 0 (Global Test)	GMDC[1:0] = 01 <sub>B</sub> GSLPR = 0 (Global Reset)	GMDC[1:0] = 01 <sub>B</sub> GSLPR = 1 (Global Stop)
Channel communication	Channel communication	Channel halt	Channel reset	Transition prohibited
Channel halt	Channel halt	Channel halt	Channel reset	Transition prohibited
Channel reset	Channel reset	Channel reset	Channel reset	Channel stop
Channel stop	Channel stop	Channel stop	Channel stop	Channel stop

**Note:** GMDC[1:0], GSLPR: Bits in the RSCAN0GCTR register

**Table 21.87** shows the global mode transition time.

**Table 21.87** Global Mode Transition Time

Mode before Transition	Mode after Transition	Maximum Transition Time
Global stop	Global reset	Three pclk cycles
Global reset	Global stop	Three pclk cycles
Global reset	Global test	Ten pclk cycles
Global reset	Global operating	Ten pclk cycles
Global test	Global reset	Three pclk cycles
Global test	Global operating	Three pclk cycles
Global operating	Global reset	Three pclk cycles
Global operating	Global test	Two CAN frames* <sup>1</sup>

Note 1. CAN frame time of the lowest communication speed of the channels in use

### 21.5.1.1 Global Stop Mode

In global stop mode, clocks of the CAN do not run and therefore power consumption is reduced. CAN registers can be read, but writing data to them is prohibited. Register values are retained. Only the clock used by the CPU for writing to the GSLPR bit runs in this mode.

After this LSI is reset, the CAN module transitions to global stop mode. Setting the GSLPR bit in the RSCAN0GCTR register to 1 (in global stop mode) in global reset mode sets the CSLPR bit in each of the RSCAN0CmCTR register to 1 (channel stop mode). If all channels are forced to transition to channel stop mode, the CAN module transitions to global stop mode. The GSLPR bit should not be modified in global operating mode or global test mode.

### 21.5.1.2 Global Reset Mode

In global reset mode, RS-CAN module settings are performed. When the RS-CAN module transitions to global reset mode, some registers are initialized. **Table 21.90** and **Table 21.91** list the registers to be initialized.

Setting the GMDC[1:0] bits in the RSCAN0GCTR register to 01<sub>B</sub> sets the CHMDC[1:0] bits in each of the RSCAN0CmCTR registers (m = 0 to 4) to 01<sub>B</sub> (channel reset mode). If all channels are forced to transition to channel reset mode, the CAN module transitions to global reset mode. Channels that are already in channel reset mode or channel stop mode do not transition (because the CHMDC[1:0] bits have already been set to 01<sub>B</sub>).

### 21.5.1.3 Global Test Mode

In global test mode, settings for test-related registers are performed. When the CAN module transitions to global test mode, all CAN communications are disabled.

Setting the GMDC[1:0] bits in the RSCAN0GCTR register to 10<sub>B</sub> sets the CHMDC[1:0] bits in each of the RSCAN0CmCTR register to 10<sub>B</sub> (channel halt mode). If all channels are forced to transition to channel halt mode, the CAN module transitions to global test mode. Channels that are in channel stop mode, channel reset mode, or channel halt mode do not transition.

### 21.5.1.4 Global Operating Mode

The RS-CAN module operates in global operating mode.

When the GMDC[1:0] bits in the RSCAN0GCTR register are set to 00<sub>B</sub>, the RS-CAN module transitions to global operating mode.

## 21.5.2 Channel Modes

Figure 21.5 shows a channel mode state transition chart. Table 21.88 shows the channel mode transition time.

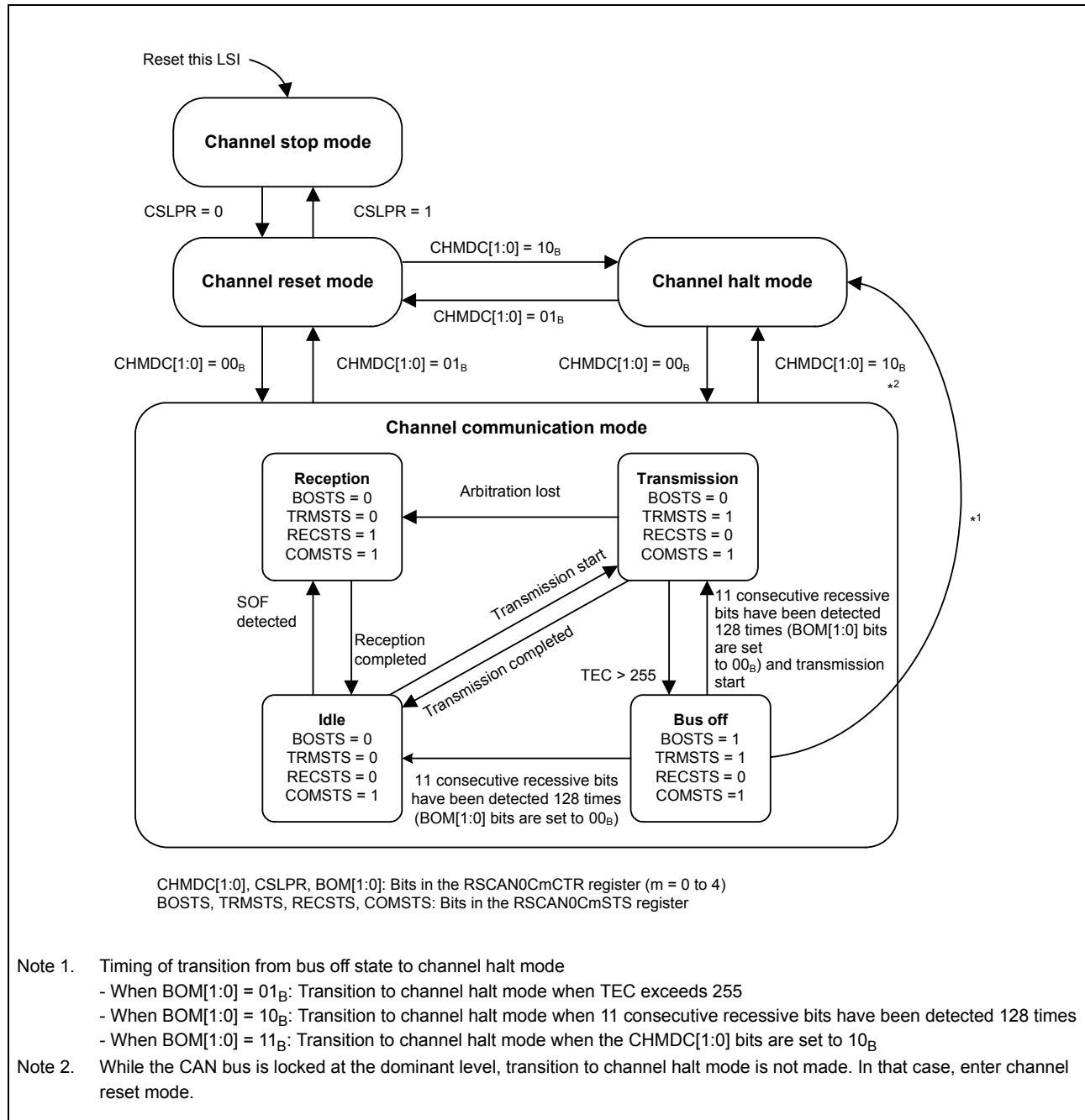


Figure 21.5 Channel Mode State Transition Chart



**Table 21.88 Channel Mode Transition Time**

Mode before Transition	Mode after Transition	Maximum Transition Time
Channel stop	Channel reset	Three pclk cycles
Channel reset	Channel stop	Three pclk cycles
Channel reset	Channel halt	Three CANm bit times
Channel reset	Channel communication	Two CANm bit times
Channel halt	Channel reset	Three pclk cycles
Channel halt	Channel communication	Three CANm bit times
Channel communication	Channel reset	Three pclk cycles
Channel communication	Channel halt	Two CANm frames

### 21.5.2.1 Channel Stop Mode

In channel stop mode, clocks are not supplied to channels and therefore power consumption is reduced. CAN registers can be read, but writing data to them is prohibited. Register values are retained.

Each channel enters channel stop mode after this LSI is reset. Channels also transition to channel stop mode when the GSLPR bit in the RSCAN0CmCTR register (m = 0 to 4) is set to 1 (channel stop mode) in channel reset mode. The GSLPR bit should not be modified in channel communication mode and channel halt mode.

### 21.5.2.2 Channel Reset Mode

In channel reset mode, channel settings are performed. When a channel transitions to channel reset mode, some channel-related registers are initialized. **Table 21.89** lists the registers to be initialized.

When the CHMDC[1:0] bits in the RSCAN0CmCTR register are set to 01<sub>B</sub> (channel reset mode) during CAN communication, communication is terminated before it is completed and the channel transitions to channel reset mode. **Table 21.89** shows the operation when the CHMDC[1:0] bits are set to 01<sub>B</sub> (channel reset mode) during CAN communication.

### 21.5.2.3 Channel Halt Mode

In channel halt mode, settings for test-related registers of channels are performed. When a channel transitions to channel halt mode, CAN communication of the channel stops.

**Table 21.89** shows operation when the CHMDC[1:0] bits are set to 10<sub>B</sub> (channel halt mode) during CAN communication.

**Table 21.89 Operation a Channel Transitions to Channel Reset Mode/Channel Halt Mode**

Mode	During Reception	During Transmission	Bus Off State
Channel reset (CHMDC[1:0] = 01 <sub>B</sub> )	Transitions to channel reset mode before reception is completed. <asterisk>* <Super_A>1	Transitions to channel reset mode before transmission is completed. <asterisk>* <Super_A>1	Transitions to channel reset mode before bus off recovery.
Channel halt<asterisk>* <Super_A>3 (CHMDC[1:0] = 10 <sub>B</sub> )	Transitions to channel halt mode after reception is completed. <asterisk>* <Super_A>2	Transitions to channel halt mode after transmission is completed.	[When BOM[1:0] = 00 <sub>B</sub> ] Transitions to channel halt mode (CHMDC[1:0] = 10 <sub>B</sub> ) only after bus off recovery. [When BOM[1:0] = 01 <sub>B</sub> ] Transitions to channel halt mode automatically when the condition for transition to bus off state is met. [When BOM[1:0] = 10 <sub>B</sub> ] Transitions to channel halt mode automatically after bus off recovery. [When BOM[1:0] = 11 <sub>B</sub> ] Transitions to channel halt mode immediately after the CHMDC[1:0] bits are set to 10 <sub>B</sub> before bus off recovery.

Note 1. To allow transition to channel reset mode after communication is completed, set the CHMDC[1:0] bits to 10<sub>B</sub> and confirm that communication has been completed and transition to channel halt mode has been made, and then set the CHMDC[1:0] bits to 01<sub>B</sub>.

Note 2. While the CAN bus is locked at the dominant level, transition to channel halt mode is not made. In that case, enter channel reset mode. The CAN bus status can be confirmed with the BLF flag of the RSCAN0CmERFL register that becomes 1 when dominant lock is detected.

Note 3. When the transition from channel reset mode to channel wait mode is to be made, set the RSCAN0CmCFG register in channel reset mode and then shift to channel wait mode.

### 21.5.2.4 Channel Communication Mode

In channel communication mode, CAN communication is performed. Each channel has the following communication states during CAN communication.

- Idle : Neither reception nor transmission is in progress.
- Reception : Receiving a message sent from another node.
- Transmission : Transmitting a message.
- Bus off : Isolated from CAN communication.

When the CHMDC[1:0] bits in the RSCAN0CmCTR register are set to 00<sub>B</sub>, the channel transitions to channel communication mode. After that, once 11 consecutive recessive bits have been detected, the COMSTS flag in the RSCAN0CmSTS register (m = 0 to 4) is set to 1 (communication is ready) and transmission and reception are enabled on the CAN network as an active node. At this time, transmission and reception of messages can be started.

### 21.5.2.5 Bus Off State

A channel transitions to the bus off state according to the transmit/receive error counter increment/decrement rules of the CAN specifications.

The conditions for returning from the bus off state are determined by the BOM[1:0] bits in the RSCAN0CmCTR register.

- When BOM[1:0] = 00<sub>B</sub>:  
Bus off recovery is compliant with the CAN specifications. After 11 consecutive recessive bits have been detected 128 times, a channel returns from the bus off state to the CAN communication ready state (error active state). At that time, the TEC[7:0] and REC[7:0] bits in the RSCAN0CmSTS register are initialized to 00<sub>H</sub> and the BORF flag in the RSCAN0CmERFL register is set to 1 (bus off recovery is detected). When the CHMDC[1:0] bits in the RSCAN0CmCTR register are set to B'10 (channel halt mode) in the bus off state, the channel transitions to channel halt mode after bus off recovery has been completed (11 consecutive recessive bits have been detected 128 times).
- When BOM[1:0] = 01<sub>B</sub>:  
When a channel transitions to the bus off state, the CHMDC[1:0] bits are set to 10<sub>B</sub> and the channel transitions to channel halt mode. At that time, the TEC[7:0] and REC[7:0] bits are initialized to 00<sub>H</sub> but the BORF flag is not set to 1.
- When BOM[1:0] = 10<sub>B</sub>:  
When a channel has transitioned to the bus off state, the CHMDC[1:0] bits are set to 10<sub>B</sub>. After bus off recovery has been completed (11 consecutive recessive bits have been detected 128 times), the channel transitions to channel halt mode. At that time, the TEC[7:0] and REC[7:0] bits are initialized to 00<sub>H</sub> and the BORF flag is set to 1.
- When BOM[1:0] = 11<sub>B</sub>:  
When the CHMDC[1:0] bits are set to 10<sub>B</sub> in the bus off state, the channel transitions to channel halt mode before bus off recovery is completed. At that time, the TEC[7:0] and REC[7:0] bits are initialized to 00<sub>H</sub> but the BORF flag is not set to 1.  
However, the BORF flag becomes 1 if a CAN module transitions to error active state (by detecting 128 times of 11 consecutive recessive bits) before CHMDC[1:0] bits are set to 10<sub>B</sub>.

If the RS-CAN module causes the channel to transition to channel halt mode simultaneously with a program write to the CHMDC[1:0] bits, the program write takes precedence. An automatic transition to channel halt mode when the BOM[1:0] bits are set to 01<sub>B</sub> or 10<sub>B</sub> is made only when the CHMDC[1:0] bits are 00<sub>B</sub> (channel communication mode). Furthermore, setting the RTBO bit in the RSCAN0CmCTR register to 1 allows a forced return from the bus off state. As soon as the RTBO bit is set to 1, the state changes to the error active state. After 11 consecutive recessive bits have been detected, the CAN module becomes ready for communication. In this case, the BORF flag is not set to 1 and the TEC[7:0] and REC[7:0] bits are initialized to 00<sub>H</sub>. Write 1 to the RTBO bit only when the BOM[1:0] value is 00<sub>B</sub>.

**Table 21.90 Registers Initialized in Global Reset Mode or Channel Reset Mode**

Register	Bit / Flag
RSCAN0CmCTR register	CTMS[1:0], CTME, CHMDC[1:0]
RSCAN0CmSTS register	CHLTSTS, EPSTS, BOSTS, TRMSTS, RECSTS, COMSTS, REC[7:0], TEC[7:0]
RSCAN0CmERFL register	CRCREG[14:0], ADERR, B0ERR, B1ERR, CERR, AERR, FERR, SERR, ALF, BLF, OVLF, BORF, BOEF, EPF, EWF, BEF
RSCAN0CFCCk register (k = 0 to 14)	When transmit/receive FIFO buffer is in transmit mode or gateway mode: CFE
RSCAN0CFSTSk register (k = 0 to 14)	When transmit/receive FIFO buffer is in transmit mode or gateway mode: CFMC[7:0], CFFLL, CFEMP, CFMLT, CFRXIF, CFTXIF
RSCAN0CFTISTS register	CFkTXIF
RSCAN0TMCP register	TMOM, TMTAR, TMTR
RSCAN0TMSTSp register (k = 0 to 79)	TMTARM, TMTRM, TMTRF[1:0], TMTSTS
RSCAN0TMTRSTSy register	TMTRSTSp (Bits of corresponding channel are initialized in channel reset mode.)
RSCAN0TMTARSTSy register	TMTARSTSp (Bits of corresponding channel are initialized in channel reset mode.)
RSCAN0TMCSTSy register	TMCSTSp (Bits of corresponding channel are initialized in channel reset mode.)
RSCAN0TMASTSy register	TMASTSp (Bits of corresponding channel are initialized in channel reset mode.)
RSCAN0TXQCCm register	TXQE
RSCAN0TXQSTSm register	TXQIF, TXQFLL, TXQEMP
RSCAN0THLCCm register	THLE
RSCAN0THLSTSm register	THLMC[4:0], THLIF, THLELT, THLFLL, THLEMP
RSCAN0GTINTSTS0 register	TSIFm, TAIFm, TQIFm, CFTIFm, THIFm (m = 0 to 3)
RSCAN0GTINTSTS1 register	TSIFm TAIFm, TQIFm, CFTIFm, THIFm (m = 4)

**Table 21.91 Registers Initialized Only in Global Reset Mode**

Register	Bit / Flag
RSCAN0GSTS register	GHLTSTS
RSCAN0GERFL register	THLES, MES, DEF
RSCAN0GTSC register	TS[15:0]
RSCAN0RMNDy register	RMNSq
RSCAN0RFCCx register	RFE
RSCAN0RFSTsx register	RFMC[7:0], RFIF, RFMLT, RFFLL, RFEMP
RSCAN0CFCCk register	When transmit/receive FIFO buffer is in receive mode: CFE
RSCAN0CFSTSk register	When transmit/receive FIFO buffer is in receive mode: CFMC[7:0], CFFLL, CFEMP, CFTXIF, CFRXIF, CFMLT
RSCAN0FESTS register	CFkEMP, RFxEMP
RSCAN0FFSTS register	CFkFLL, RFxFLL
RSCAN0FMSTS RSCAN0register	CFkMLT, RFxMLT
RSCAN0RFISTS register	RFxIF
RSCAN0CFRISTS register	CFkRXIF
RSCAN0GTSTCFG register	C0ICBCE, C1ICBCE, C2ICBCE, C3ICBCE, C4ICBCE
RSCAN0GTSTCTR register	ICBCTME

## 21.6 Reception Function

There are two reception types.

- Reception by receive buffers:  
Zero to 79 receive buffers can be shared by all channels. Since messages stored in receive buffers are overwritten at each reception, the latest receive data can always be read.
- Reception by receive FIFO buffers and transmit/receive FIFO buffers (receive mode):  
Eight receive FIFO buffers can be shared by all channels and three dedicated transmit/receive FIFO buffers are provided for each channel. Messages of up to the number of buffer stages specified with the RFDC[2:0] and CFDC[2:0] bits can be stored in FIFO buffers and can be read sequentially from the oldest.

### 21.6.1 Data Processing Using the Receive Rule Table

Data processing using the receive rule table allows dispatching of selected messages to the specified buffer. Data processing includes acceptance filter processing, DLC filter processing, routing processing, label addition processing, and mirror function processing.

Up to 128 receive rules can be registered per channel and up to (64 × number of channels) total receive rules can be registered in the entire module. (Up to 320 receive rules can be registered in this module that has five channels.) Set receive rules for each channel. Receive rules cannot be shared with other channels. If receive rules are not set, no messages can be received. **Figure 21.6** illustrates how receive rules are registered.

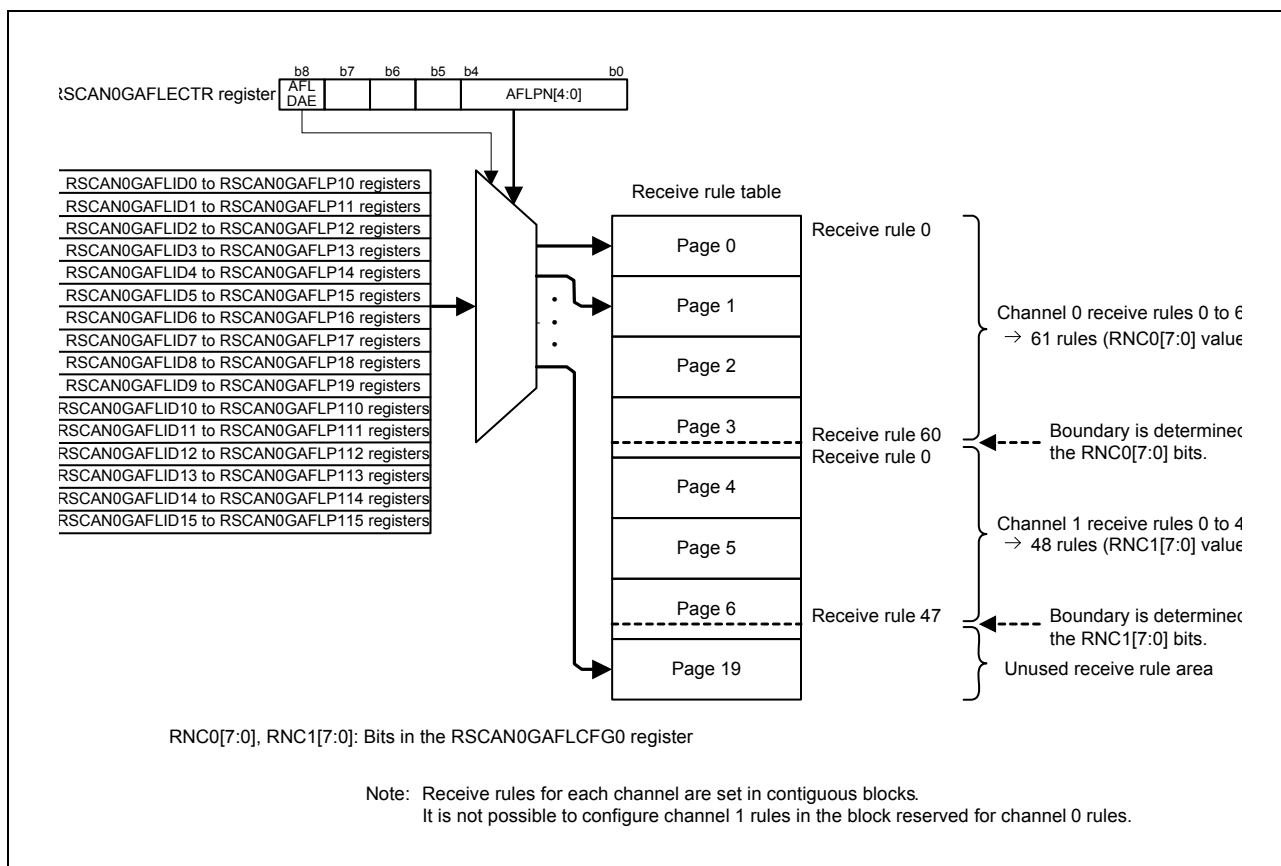


Figure 21.6 Entry of Receive Rules (for Setting Channel 0 and 1)

Each receive rule consists of 16 bytes in the RSCAN0GAFLID<sub>j</sub>, RSCAN0GAFLM<sub>j</sub>, RSCAN0GAFLP0<sub>j</sub>, and RSCAN0GAFLP1<sub>j</sub> registers ( $j = 0$  to 15). The RSCAN0GAFLID<sub>j</sub> register ( $j = 0$  to 15) is used to set GAFLID, GAFLIDE bit, GAFLRTR bit, and the mirror function, the RSCAN0GAFLM<sub>j</sub> register is used to set mask, the RSCAN0GAFLP0<sub>j</sub> register is used to set label information to be added, DLC value, and storage receive buffer, and the RSCAN0GAFLP1<sub>j</sub> register is used to set storage FIFO buffer. Up to 16 receive rules can be set per page.

### 21.6.1.1 Acceptance Filter Processing

In the acceptance filter processing, the ID data, IDE bit, and RTR bit in a received message are compared with the ID data, IDE bit, and RTR bit set in the receive rule of the corresponding channel. When all these bits match, the message passes through the acceptance filter processing. The ID data, IDE bit, and RTR bit in the received message which correspond to the bits set to 0 (bits are not compared) in the RSCAN0GAFLM<sub>j</sub> register are not compared and are regarded as matched.

Check begins with the receive rule of the minimum number for the corresponding channel. When all the bits to be compared in a received message match the bits set in the receive rule or when all the receive rules are compared without any match, filter processing stops. If there is no matching receive rule, the received message is not stored in the receive buffer or FIFO buffer.

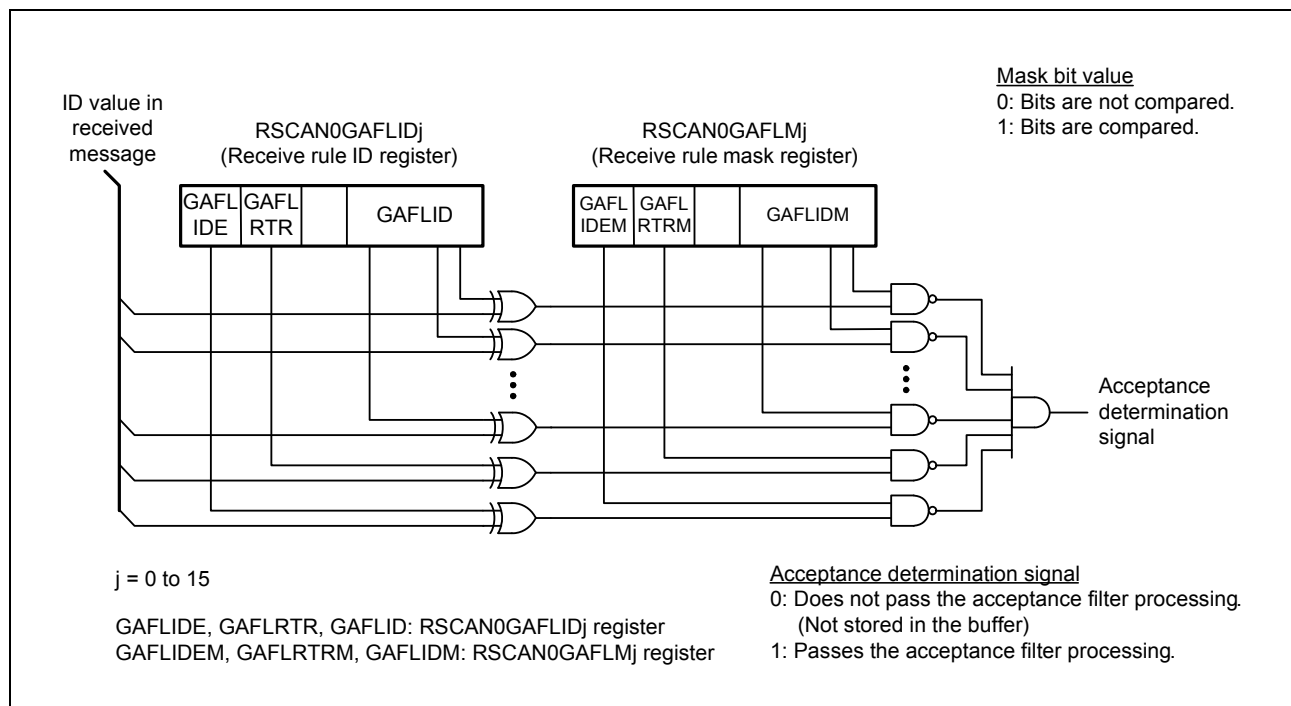


Figure 21.7 Acceptance Filter Function

### 21.6.1.2 DLC Filter Processing

When the DCE bit in the RSCAN0GCFG register is set to 1 (DLC check is enabled), DLC filter processing is added to messages that passed through the acceptance filter processing. When the DLC value in a message is equal to or larger than the DLC value set in the receive rule, the message passes through the DLC filter processing.

When a message has passed through the DLC filter processing with the DRE bit in the RSCAN0GCFG register set to 0 (DLC replacement is disabled), the DLC value in the received message is stored in the buffer. In this case, all the data bytes in the received message are stored in the buffer.

When a message has passed through the DLC filter processing with the DRE bit in the RSCAN0GCFG register set to 1 (DLC replacement is enabled), the DLC value in the receive rule is stored in the buffer instead of the DLC value in the received message. In this case, a value of 00<sub>H</sub> is stored in each data byte beyond the number of bytes which is indicated by the DLC value in the receive rule.

When the DLC value in the received message is smaller than that in the receive rule, the message does not pass through the DLC filter processing. In this case, the message is not stored in the receive buffer or the FIFO buffer and the DEF flag in the RSCAN0GERFL register is set to 1 (a DLC error is present).

### 21.6.1.3 Routing Processing

Messages that passed through the acceptance filter processing and the DLC filter processing are stored in receive buffers, receive FIFO buffers, or transmit/receive FIFO buffers (set to receive mode or gateway mode). Message storage destination is set by the GAFLRMV and GAFLRMDP[6:0] bits in the RSCAN0GAFLP0j register (j = 0 to 15) and by the RSCAN0GAFLP1j register. Messages that passed through the acceptance filter processing and the DLC filter processing can be stored in up to eight buffers.

### 21.6.1.4 Label Addition Processing

It is possible to add 12-bit label information to messages that passed through the filter processing and store them in buffers. This label information is set in the GAFLPTR[11:0] bits in the RSCAN0GAFLP0j register.

### 21.6.1.5 Mirror Function Processing

The mirror function allows the CAN node to receive its own transmitted messages. The mirror function is made available by setting the MME bit in the RSCAN0GCFG register to 1 (mirror function is enabled).

When the mirror function is in use, receive rules for which the GAFLLB bit in the RSCAN0GAFLIDj register is set to 0 are used for data processing when receiving messages transmitted from other CAN nodes. When the CAN node is receiving its own transmitted messages, receive rules for which the GAFLLB bit is set to 1 are used for data processing.

### 21.6.1.6 Timestamp

The timestamp counter is a 16-bit free-running counter used for recording message receive time. The timestamp counter value is fetched at the start-of-frame (SOF) timing of a message and is then stored in a receive buffer or a FIFO buffer together with the message ID and data. Either pclk or the CANm bit time clock (m = 0 to 4) may be selected as a timestamp counter clock source using the TSBTCS[2:0] and TSSS bits in the RSCAN0GCFG register. The timestamp counter count source is obtained by dividing the selected clock source by the TSP[3:0] value in the RSCAN0GCFG register.

When the CANm bit time clock is used as a clock source, the timestamp counter stops when the corresponding channel transitions to channel reset mode or channel halt mode. When the pclk is used as a clock source, the timestamp function is not affected by channel mode.

The timestamp counter value is reset to 0000<sub>H</sub> by setting the TSRST bit in the RSCAN0GCTR register to 1.

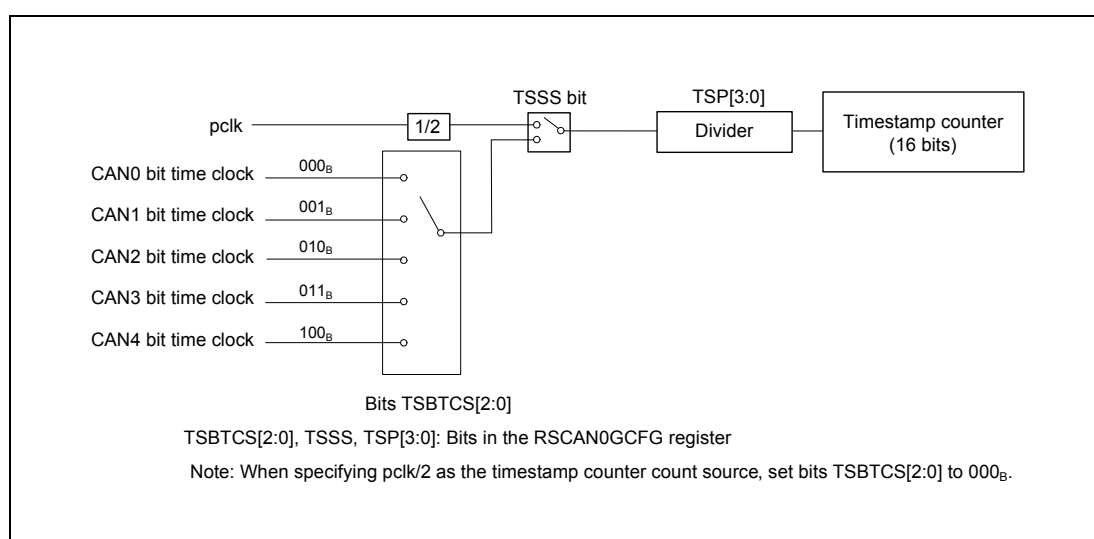


Figure 21.8 Timestamp Function Block Diagram



## 21.7 Transmission Functions

There are three types of transmission.

- Transmission using transmit buffers:  
Each channel has 16 buffers.
- Transmission using transmit/receive FIFO buffers (transmit mode):  
Each channel has three FIFO buffers. Up to 128 messages can be contained in a single FIFO buffer. Each FIFO buffer is used with a link to a transmit buffer. Only the message to be transmitted next in a FIFO buffer becomes the target of transmit priority determination. Messages are transmitted sequentially on a first-in, first-out basis.
- Transmission using transmit queues:  
Up to 16 transmit buffers per channel can be allocated to the transmit queues. Transmit buffer  $((16 \times m) + 15)$  is used as an access window of a corresponding channel. Transmit buffers are allocated to transmit queues in descending order of buffer number. All messages in transmit queues, which are targets of priority determination, are transmitted in the order of ID number.

Figure 21.9 shows the allocation of transmit queues and transmit/receive FIFO buffer link.

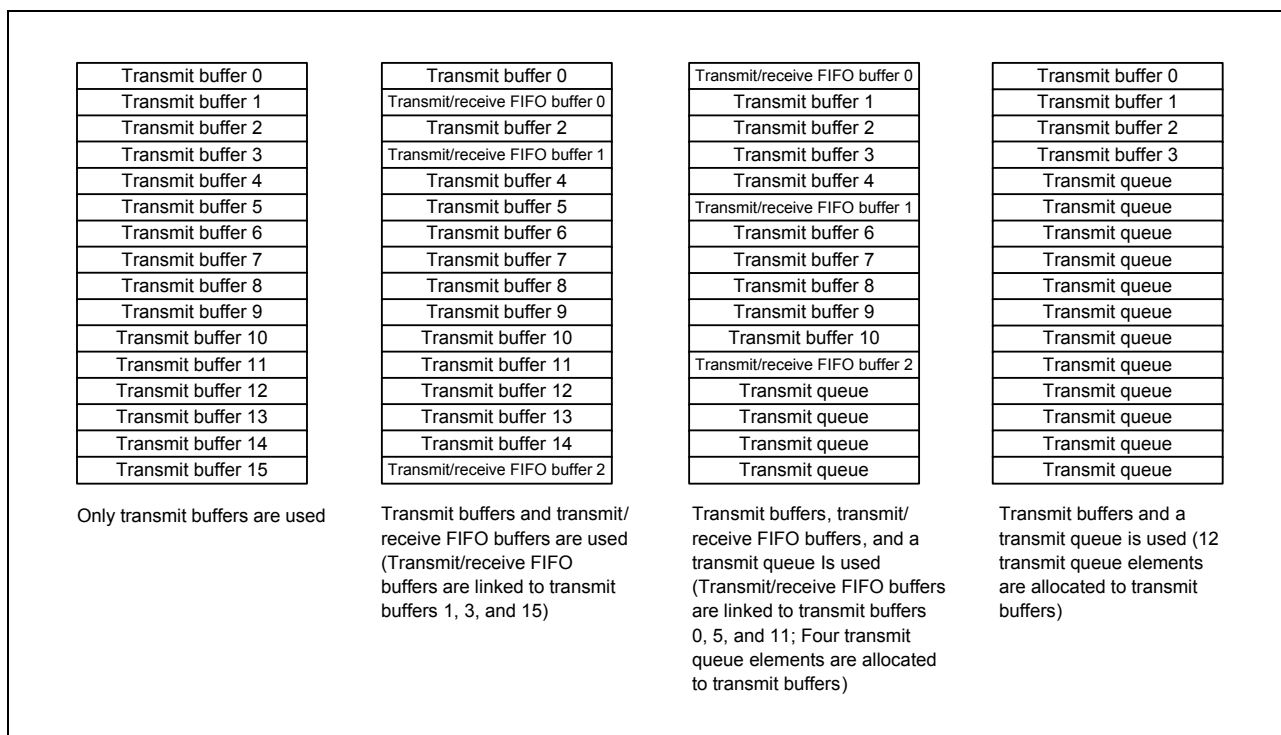


Figure 21.9 Allocation of Transmit Queues and Transmit/Receive FIFO Buffer Links

### 21.7.1 Transmit Priority Determination

If transmit requests are issued from multiple buffers or from the queue on the same channel, transmit priority is determined using one of the following methods.

The priority is determined by using one of the following methods.

- ID priority (TPRI bit = 0)
- Transmit buffer number priority (TPRI bit = 1)

All CAN channels use the setting of the TPRI bit in the RSCAN0GCFG register.

When the TPRI bit is set to 0, messages are transmitted according to the priority of stored message IDs. ID priority conforms to the CAN bus arbitration specification defined in the CAN specifications. All IDs of pending transmit messages are targets of priority determination, regardless of whether they are stored in transmit buffers, transmit/receive FIFO buffers (set to transmit mode or gateway mode), or the transmit queue. If even a single transmit queue is used, select ID priority. When transmit/receive FIFO buffers are used, the oldest message in a FIFO buffer becomes the target of priority determination. When a message is being transmitted from a transmit/receive FIFO buffer, the next message in the FIFO buffer becomes the target of priority determination. When a transmit queue is used, all messages in the transmit queue are targets of priority determination. If the same ID is set for two or more buffers, the buffer with the smaller buffer number takes precedence.

When the TPRI bit is set to 1, the message in the transmit buffer with the minimum buffer number among all buffers with a transmit request is transmitted first. When transmit/receive FIFO buffers are linked to transmit buffers, transmit priority is determined according to linked transmit buffer numbers.

When messages are retransmitted due to an arbitration-lost or an error, transmit priority determination is made again regardless of the TPRI bit.

### 21.7.2 Transmission Using Transmit Buffers

Setting the transmit request bit (TMTR bit in the RSCAN0TMCp register) in a transmit buffer to 1 (transmission is requested) allows transmission of data frames or remote frames.

The transmit result is shown by the TMTRF[1:0] flag in the corresponding RSCAN0TMSTSp register ( $p = 0$  to 79). When transmit completes successfully, the TMTRF[1:0] flag is set to  $10_B$  (transmission has been completed (without transmit abort request)) or  $11_B$  (transmission has been completed (with transmit abort request)).

#### 21.7.2.1 Transmit Abort Function

With respect to transmit buffers for which the TMTRM bit in the RSCAN0TMSTSp register is set to 1 (a transmit request is present), when the TMTAR bit in the RSCAN0TMCp register is set to 1 (transmit abort is requested), the transmit request is canceled. When transmit abort is completed, the TMTRF[1:0] flag in the RSCAN0TMSTSp register is set to  $01_B$  (transmit abort has been completed) and the transmit request is canceled (clearing the TMTRM bit to 0).

A message that is being transmitted or a message to be transmitted next according to the transmit priority determination cannot be aborted. However, when an arbitration-lost or an error occurs during transmission of a message for which the TMTAR bit is set to 1, retransmission is not performed.

### 21.7.2.2 One-Shot Transmission Function (Retransmission Disabling Function)

When the TMOM bit in the RSCAN0TMCp register is set to 1 (one-shot transmission is enabled), transmission is performed only once. Even if an arbitration-lost or an error occurs, retransmission is not performed.

The one-shot transmit result is shown by the TMTRF[1:0] flag in the corresponding RSCAN0TMSTSp register. When one-shot transmission completes successfully, the TMTRF[1:0] flag is set to 10<sub>B</sub> or 11<sub>B</sub>. When an arbitration-lost or an error occurs, the TMTRF[1:0] flag is set to 01<sub>B</sub> (transmit abort has been completed).

### 21.7.3 Transmission Using FIFO Buffers

Multiple messages can be stored in a single transmit/receive FIFO buffers, up to the number specified by the FIFO buffer depth, which is set by the CFDC[2:0] bits in the RSCAN0CFCCk register (k = 0 to 14). Messages are transmitted sequentially on a first-in, first-out basis.

Each transmit/receive FIFO buffer is linked to a transmit buffer selected by the CFTML[3:0] bits in the RSCAN0CFCCk register. When the CFE bit in the RSCAN0CFCCk register is set to 1 (transmit/receive FIFO buffers are used), transmit/receive FIFO buffers become targets of transmit priority determination. Priority of only the next transmit message is determined in the FIFO buffer.

When the CFE bit is set to 0 (no transmit/receive FIFO buffer is used), the CFEMP flag is set to 1 (the transmit/receive FIFO buffer contains no message (buffer empty)) at the timing below.

- The transmit/receive FIFO buffer becomes empty immediately if the message in it is not being transmitted or is not to be transmitted next.
- The transmit/receive FIFO buffer becomes empty after transmission completion, CAN bus error detection, or arbitration-lost in the case that a message in it is being transmitted or to be transmitted next.

When the CFE bit is cleared to 0, all messages in transmit/receive FIFO buffers are lost and messages cannot be stored in FIFO buffers. Confirm that the CFEMP flag is set to 1 before setting the CFE bit to 1 again.

### 21.7.3.1 Interval Transmission Function

A message transmission interval time can be set to space the transmission of messages from the same FIFO buffer when using a transmit/receive FIFO buffer set to transmit mode or gateway mode.

Immediately after the first message has been transmitted successfully from the FIFO buffer with the CFE bit in the RSCAN0CFCCk register set to 1, the interval timer starts counting (after EOF7 of the CAN protocol). After that, when the interval time has passed, the next message is transmitted. The interval timer stops in channel reset mode or by clearing the CFE bit to 0.

The interval time is set by the CFITT[7:0] bits in the RSCAN0CFCCk register. When the interval timer is not used, set the CFITT[7:0] bits to 00<sub>H</sub>.

Select an interval timer count source using the CFITR and CFITSS bits in the RSCAN0CFCCk register. When the CFITR and CFITSS bits are set to 00<sub>B</sub>, the count source is obtained by dividing  $plk/2$  by the value of the ITRCP[15:0] bits. When the CFITR and CFITSS bits are set to 10<sub>B</sub>, the count source is obtained by dividing  $plk/2$  by (the value of the ITRCP[15:0] bits in the RSCAN0GCFG register  $\times 10$ ). When the CFITR and CFITSS bits are set to x1<sub>B</sub>, the CAN<sub>n</sub> bit time clock is used as a count source.

The interval time is calculated by the following equations where M is the value set to ITRCP[15:0] and N is the set CFITT[7:0] value.

- When CFITR and CFITSS = 00<sub>B</sub> ( $f_{PBA}$  is the frequency of  $plk$ ):

$$\frac{1}{f_{PBA}} \times 2 \times M \times N$$

- When CFITR and CFITSS = 10<sub>B</sub>:

$$\frac{1}{f_{PBA}} \times 2 \times M \times 10 \times N$$

- When CFITR and CFITSS = x1<sub>B</sub> ( $f_{CANBIT}$  is the frequency of CAN<sub>m</sub> bit time clock):

$$\frac{1}{f_{CANBIT}} \times N$$

Figure 21.10 shows the interval timer block diagram.

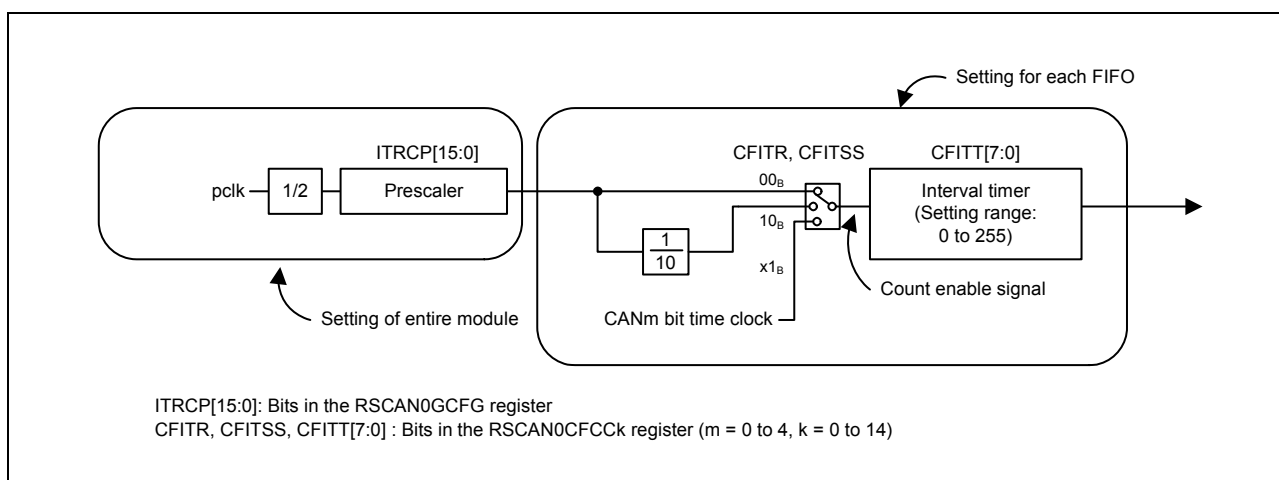


Figure 21.10 Interval Timer Block Diagram

Figure 21.11 shows the interval timer timing diagram.

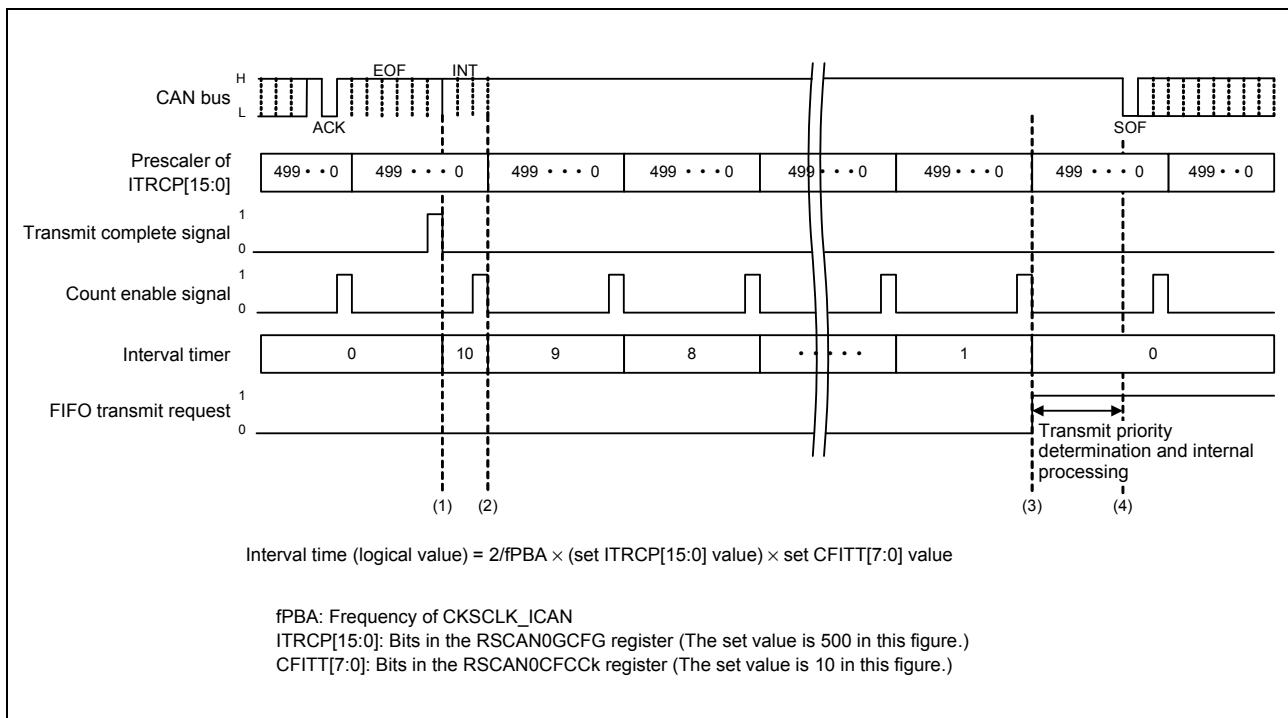


Figure 21.11 Interval Timer Timing Chart

- (1) The interval timer starts counting upon completion of transmission. Since the prescaler is not initialized at the time of transmission completion, the first interval time contains an error of up to one count of the interval timer.
- (2) The interval timer is decremented by the next count enable signal.
- (3) When the interval timer has decreased to 0, the transmit/receive FIFO buffer issues a transmit request.
- (4) The transmit/receive FIFO buffer is determined for the next transmission by the priority determination, it starts transmitting data. Transmission starts usually with a delay of three CANm bit time clock cycles or less from the issue of transmit request. If multiple internal processes (such as receive filter processing, message routing, and transmit priority determination) take place in all channels, a delay of up to 504 cycles of the clk<sub>c</sub> may be generated.

### 21.7.4 Transmission Using Transmit Queues

Three to sixteen buffers are allocated to a transmit queue for each channel, and transmit buffer  $((16 \times m) + 15)$  is used as an access window of a corresponding channel.

All messages in a transmit queue are targets of transmit priority determination and are transmitted in the ID priority order regardless of storage sequence. If two messages having the same ID are stored in a transmit queue, these messages are not always transmitted in the order of their storage in the transmit queue.

Setting the TXQE bit in the RSCAN0TXQCCm register to 0 disables transmit queues. When the TXQE bit is set to 0, the TXQEMP flag in the RSCAN0TXQSTSm register is set to 1 (the transmit queue contains no messages (transmit queue empty)) at the timing below.

- The transmit queue becomes empty immediately when no message in it is being transmitted or will be transmitted next.
- The transmit queue becomes empty after transmission completion, CAN bus error detection, or arbitration-lost when a message in it is being transmitted or will be transmitted next.

When the TXQE bit is cleared to 0, all messages in transmit queues are lost and messages cannot be stored in transmit queues. Confirm that the TXQEMP flag is set to 1 before setting the TXQE bit to 1 again.

### 21.7.5 Transmit History Function

Information about transmitted messages can be stored in the transmit history buffer. Each channel has a single transmit history buffer that can contain 16 sets of transmit history data.

A message transmit source buffer type can be selected by the THLDTE bit in the RSCAN0THLCCm register. The THLEN bit in the RSCAN0CFIDk register ( $k = 0$  to 14) determines whether transmit history data is stored for each message.

The following information on a transmitted message will be stored in the transmission history buffer after the successful completion of transmission.

Storage of the transmission history data after the successful completion of transmission may take up to 144 cycles of pclk.

- Buffer type
  - 001<sub>B</sub>: Transmit buffer
  - 010<sub>B</sub>: Transmit/receive FIFO buffer
  - 100<sub>B</sub>: Transmit queue
- Buffer number
  - Number of source transmit buffer, transmit queue, or transmit/receive FIFO buffer.
  - This number depends on buffer types. See **Table 21.92**.
- Label data
  - Label information of the transmit message

Table 21.92 Transmit History Data Buffer Numbers

Buffer type			
Buffer No.	001 <sub>B</sub>	010 <sub>B</sub>	100 <sub>B</sub>
0000 <sub>B</sub>	Transmit buffer 16 × m + 0	Buffer numbers of the transmit buffer linked to the transmit/receive FIFO buffer by the CFTML[3:0] bits in the RSCAN0CFCCk register (k = 0 to 14)	Buffer numbers of the transmit buffer allocated to the transmit queue that performed transmission
0001 <sub>B</sub>	Transmit buffer 16 × m + 1		
0010 <sub>B</sub>	Transmit buffer 16 × m + 2		
0011 <sub>B</sub>	Transmit buffer 16 × m + 3		
0100 <sub>B</sub>	Transmit buffer 16 × m + 4		
0101 <sub>B</sub>	Transmit buffer 16 × m + 5		
0110 <sub>B</sub>	Transmit buffer 16 × m + 6		
0111 <sub>B</sub>	Transmit buffer 16 × m + 7		
1000 <sub>B</sub>	Transmit buffer 16 × m + 8		
1001 <sub>B</sub>	Transmit buffer 16 × m + 9		
1010 <sub>B</sub>	Transmit buffer 16 × m + 10		
1011 <sub>B</sub>	Transmit buffer 16 × m + 11		
1100 <sub>B</sub>	Transmit buffer 16 × m + 12		
1101 <sub>B</sub>	Transmit buffer 16 × m + 13		
1110 <sub>B</sub>	Transmit buffer 16 × m + 14		
1111 <sub>B</sub>	Transmit buffer 16 × m + 15		

Label data is used to identify each message. Unique label data can be added to each message transmitted from a transmit buffer, transmit queue, or transmit/receive FIFO buffer.

Transmit history data can be read from the RSCAN0THLACCm register. If an attempt is made to store new transmit history data while the buffer is full, the buffer overflows and the new data is discarded.

## 21.8 Gateway Function

When a transmit/receive FIFO buffer is set to gateway mode, receive messages can be transmitted from an arbitrary channel without CPU intervention.

When a transmit/receive FIFO buffer for which the CFM[1:0] bits in the RSCAN0CFCCk register are set to 10<sub>B</sub> (gateway mode) is selected by the RSCAN0GAFLP1j register, messages that passed through the filter processing of the receive rule are stored in the specified transmit/receive FIFO buffer and are automatically transmitted from the buffer.

Messages stored in a transmit/receive FIFO buffer are transmitted sequentially on a first-in, first-out basis. Only the message to be transmitted next becomes the target of transmit priority determination.

Transmit/receive FIFO buffers in the gateway mode are disabled by setting the CFE bit in the RSCAN0CFCCk register to 0 and the CFEMP flag becomes 1 according to the timing below.

- The transmit/receive FIFO buffer becomes empty immediately when the oldest message in it is not being transmitted and will not to be transmitted next.
- The transmit/receive FIFO buffer becomes empty after transmission completion, CAN bus error detection, or arbitration-lost when the message in it is being transmitted or will be transmitted next.

When the CFE bit is cleared to 0, all messages in transmit/receive FIFO buffers are lost and messages can no longer be stored in transmit/receive FIFO buffers. Confirm that the CFEMP flag is set to 1 before setting the CFE bit to 1 again.



## 21.9 Test Function

The test function is classified into communication tests and global tests.

- Communication tests: Performed for each channel.
  - Standard test mode
  - Listen-only mode
  - Self-test mode 0 (external loopback mode)
  - Self-test mode 1 (internal loopback mode)
- Global tests: Performed for the entire module
  - Inter-channel communication test

### 21.9.1 Standard Test Mode

Standard test mode allows CRC test.

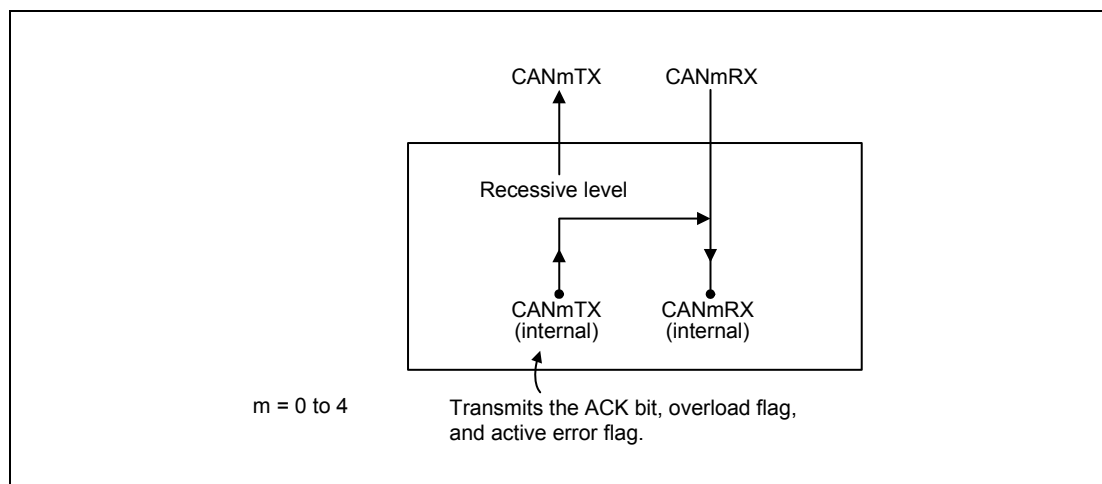
### 21.9.2 Listen-Only Mode

Listen-only mode allows reception of data frames and remote frames. Only recessive bits are transmitted on the CAN bus, and the ACK bit, overload flag, and active error flag are not transmitted.

Listen-only mode is available for detecting the communication speed.

Do not make a transmit request from any buffer or queue in listen-only mode.

**Figure 21.12** shows the connection when listen-only mode is selected.



**Figure 21.12** Connection when Listen-Only Mode is Selected

### 21.9.3 Self-Test Mode (Loopback Mode)

In self-test mode, transmitted messages are compared with the receive rule of the own channel and the messages are stored in a buffer if they have passed through the filter processing. Messages transmitted from other CAN nodes are compared only with the receive rule for which the GAFLLB bit in the RSCAN0GAFLIDj register (j = 0 to 15) is set to 0 (when a message transmitted from another CAN node is received).

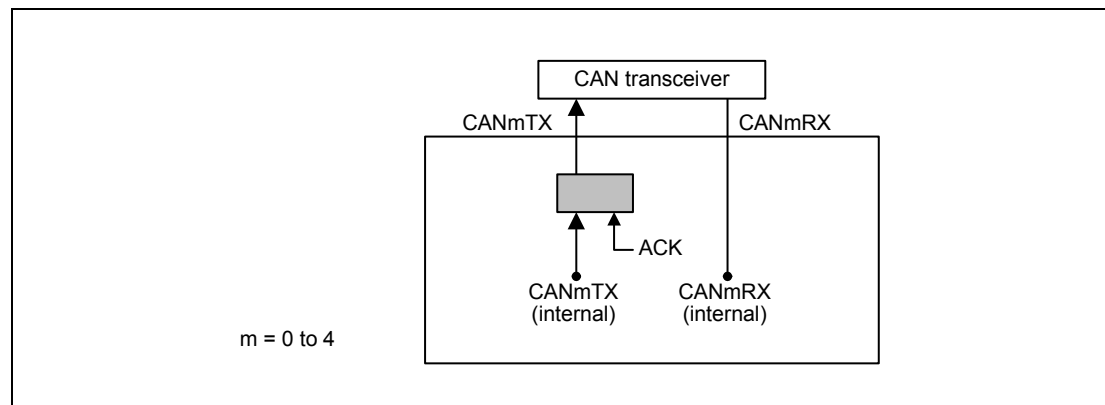
If the mirror function and self-test mode are both enabled, the self-test mode setting takes precedence.

#### 21.9.3.1 Self-Test Mode 0 (External Loopback Mode)

Self-test mode 0 is used to perform a loopback test within a channel including the CAN transceiver.

In self-test mode 0, transmitted messages are handled as messages received through the CAN transceiver and are stored in a buffer. An ACK bit is generated to receive messages transmitted from the own CAN node.

**Figure 21.13** shows the connection when self-test mode 0 is selected.



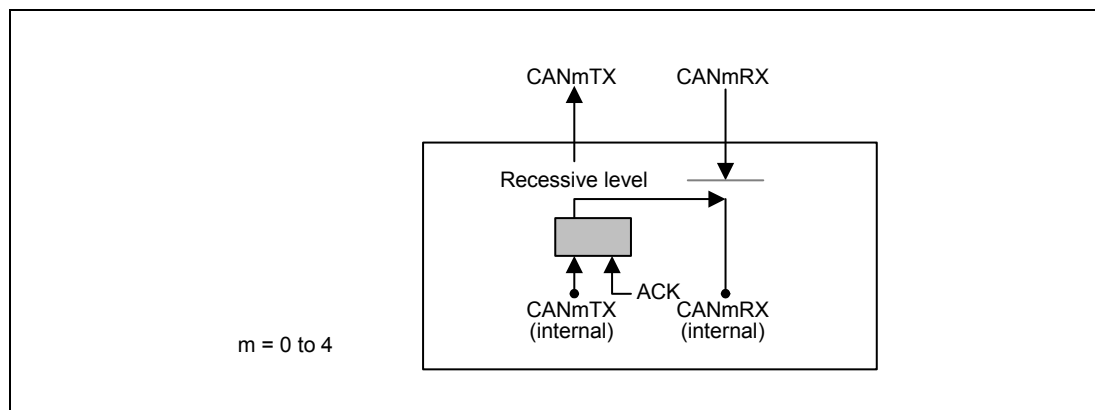
**Figure 21.13** Connection when Self-Test Mode 0 is Selected

### 21.9.3.2 Self-Test Mode 1 (Internal Loopback Mode)

In self-test mode 1, transmitted messages are handled as received messages and are stored in a buffer. An ACK bit is generated to receive messages transmitted from the own CAN node.

In self-test mode 1, internal feedback from the internal CANmTX pin ( $m = 0$  to 4) to the internal CANmRX pin is performed. The external CANmRX pin input is isolated. The external CANmTX pin outputs only recessive bits.

**Figure 21.14** shows the connection when self-test mode 1 is selected.



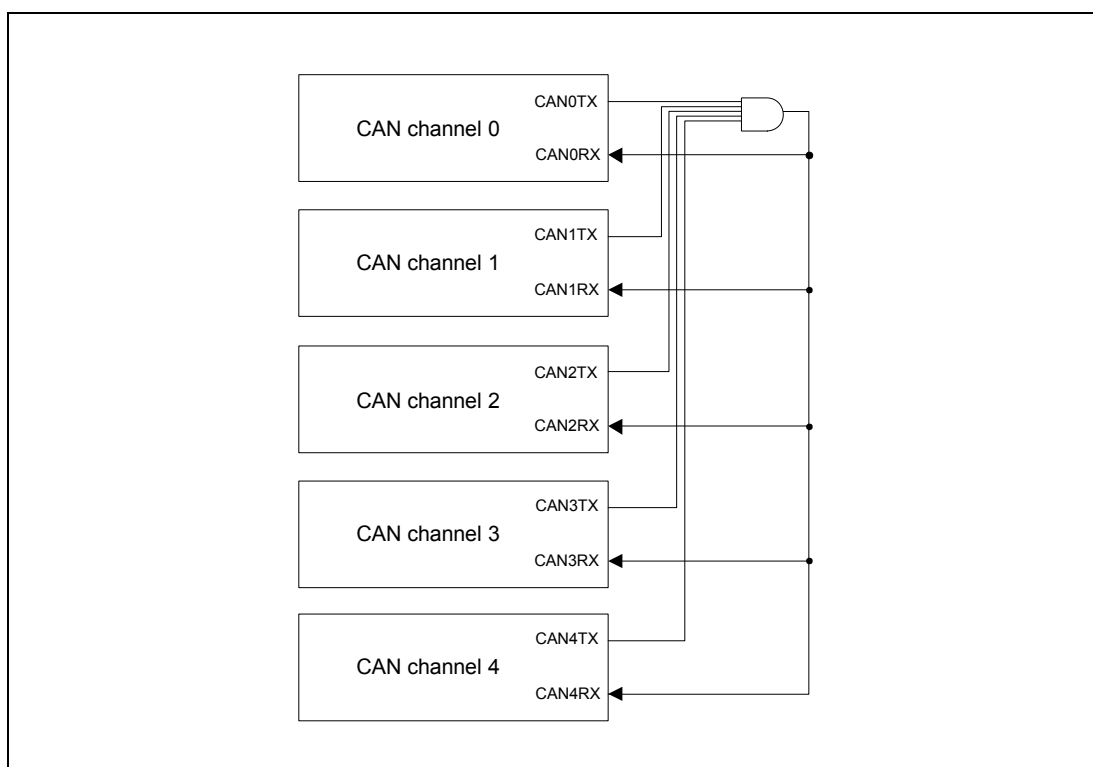
**Figure 21.14** Connection when Self-Test Mode 1 is Selected

### 21.9.4 Inter-Channel Communication Test

The inter-channel communication test function allows communication test by internally connecting CAN channels to each other. During this test, channels are isolated from the external CAN bus.

Before starting data transmission/reception in channel communication mode, make transmission/reception settings for each channel.

**Figure 21.15** shows the connection for inter-channel communication test.



**Figure 21.15** Connection for Inter-Channel Communication Test

## 21.10 RS-CAN Setting Procedure

### 21.10.1 Initial Settings

The RS-CAN module initializes the CAN RAM after this LSI is reset. The RAM initialization time is 6082 cycles of the clk. The GRAMINIT flag in the RSCAN0GSTS register is set to 1 (CAN RAM initialization is ongoing) during the RAM initialization and is cleared to 0 (CAN RAM initialization is finished) when the initialization is completed. Make CAN settings after the GRAMINIT flag is cleared to 0. **Figure 21.16** shows the CAN setting procedure after this LSI is reset.

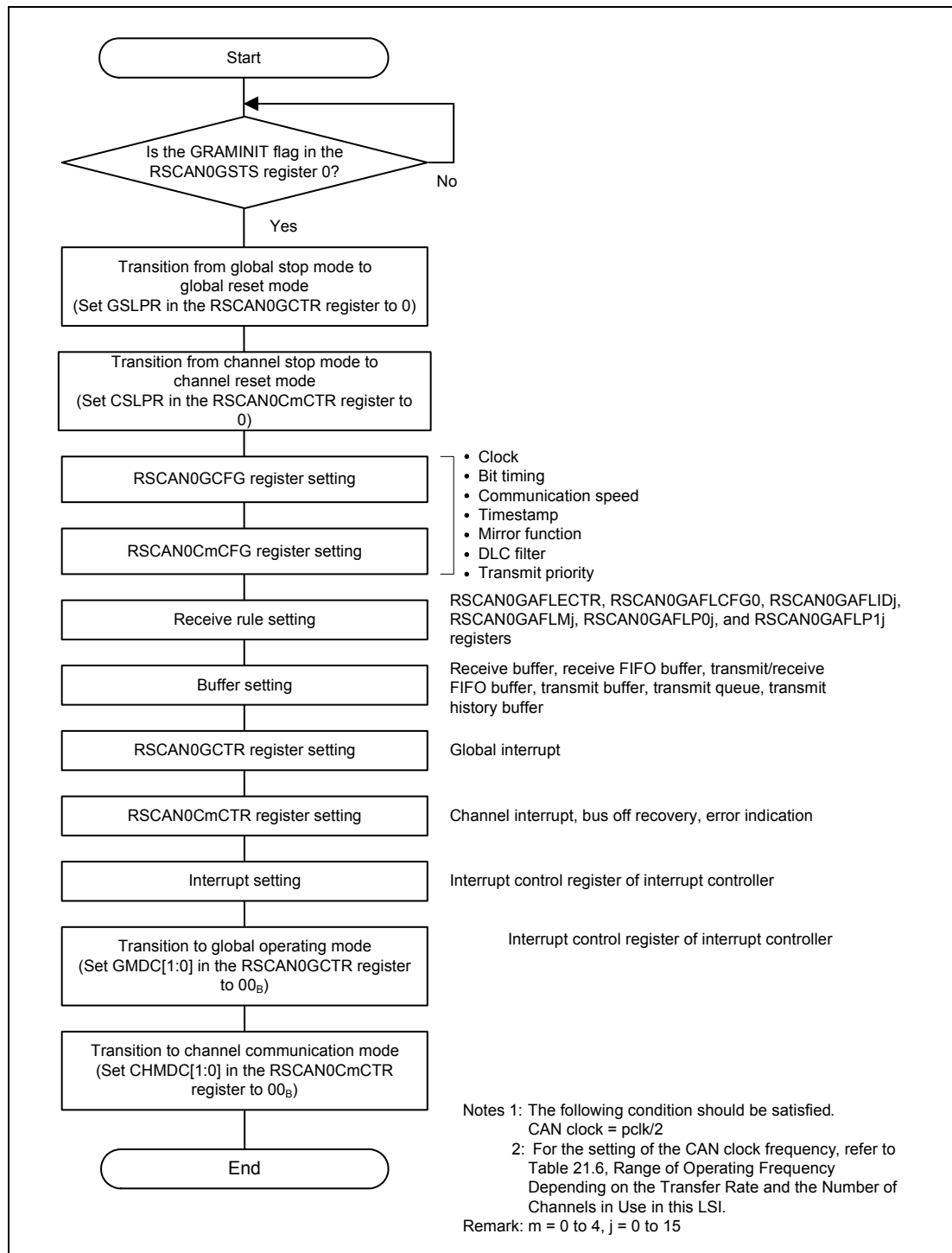


Figure 21.16 CAN Setting Procedure after the this LSI is Reset

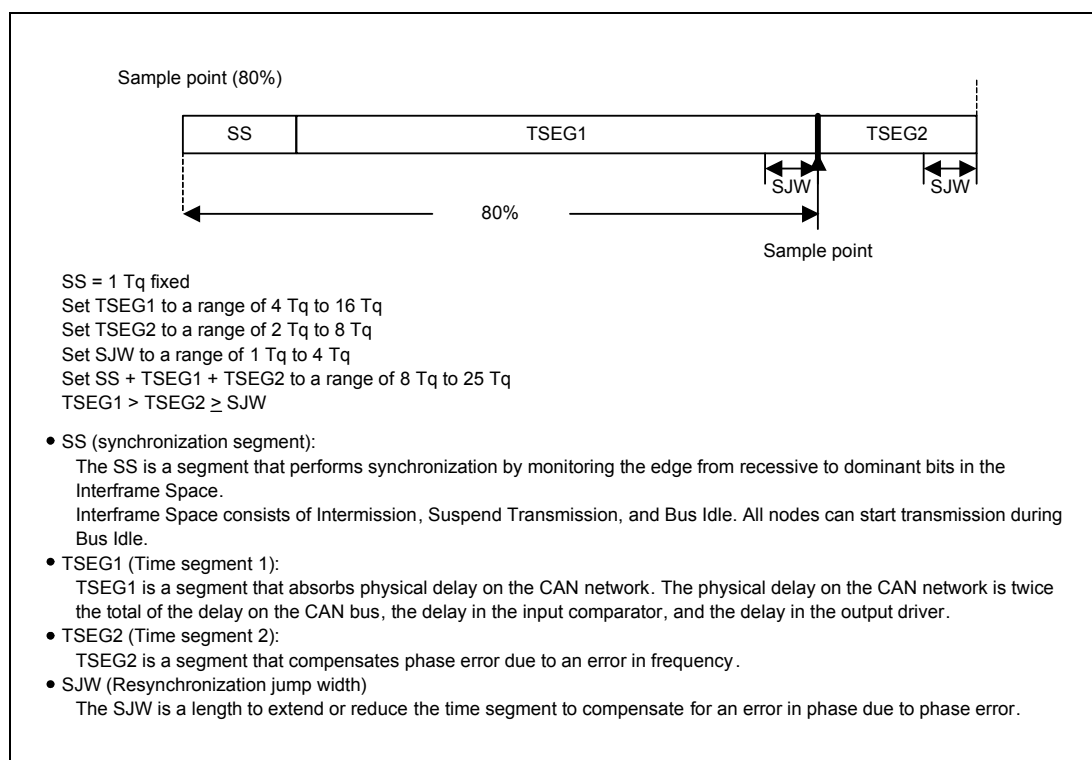
### 21.10.1.1 Clock Setting

Set the CAN clock (fCAN) as a clock source of the RS-CAN module. Select the clk\_xincan or clkc using the DCS bit in the RSCAN0GCFG register.

### 21.10.1.2 Bit Timing Setting

In the CAN protocol, one bit of a communication frame consists of three segments SS, TSEG1, and TSEG2. Two of the segments, TSEG1 and TSEG2, can be set by the RSCAN0CmCFG register for each channel. Sample point timing can be determined by setting these two segments. This timing can be adjusted in units of 1 Time Quantum (referred to as Tq hereinafter). 1 Tq is equal to one CANmTq clock cycle. The CANmTq clock is obtained by selecting the clock source with the DCS bit in the RSCAN0GCFG register and selecting the clock division ratio with the BRP[9:0] bits in the RSCAN0CmCFG register.

**Figure 21.17** shows the bit timing chart. **Table 21.93** shows an example of bit timing setting.



**Figure 21.17** Bit Timing Chart

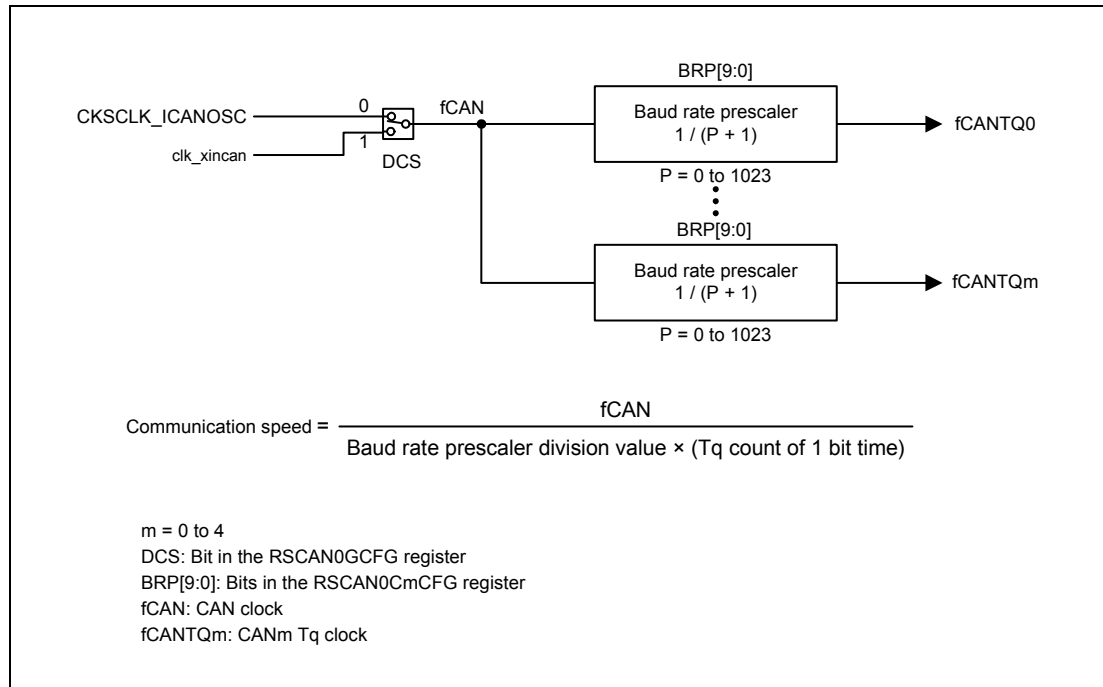
**Table 21.93** Example of Bit Timing Setting

1 Bit	Set Value (Tq)				Sample Point (%) Note: See Figure 21.17.
	SS	TSEG1	TSEG2	SJW	
8Tq	1	4	3	1	62.50
	1	5	2	1	75.00
12Tq	1	8	3	1	75.00
	1	9	2	1	83.33
16Tq	1	10	5	1	68.75
	1	11	4	1	75.00
24Tq	1	14	9	1	62.50
	1	15	8	1	66.66

### 21.10.1.3 Communication Speed Setting

Set the CAN communication speed for each channel using the fCAN, baud rate prescaler division value (BRP[9:0] bits in the RSCAN0CmCFG register), and Tq count per bit time.

**Figure 21.18** shows the CAN clock control block diagram, and **Table 21.94** shows an example of the communication speed setting.



**Figure 21.18** CAN Clock Control Block Diagram

**Table 21.94** Example of Communication Speed Setting

Communication speed	fCAN		
	32MHz	24MHz	8MHz
1Mbps	8Tq (4) 16Tq (2)	8Tq (3) 12Tq (2) 24Tq (1)	8Tq (1)
500Kbps	8Tq (8) 16Tq (4)	8Tq (6) 12Tq (4) 24Tq (2)	8Tq (2) 16Tq (1)
125Kbps	8Tq (32) 16Tq (16)	8Tq (24) 12Tq (16) 24Tq (8)	8Tq (8) 16Tq (4)

**Note:** Values in ( ) are baud rate prescaler division values.

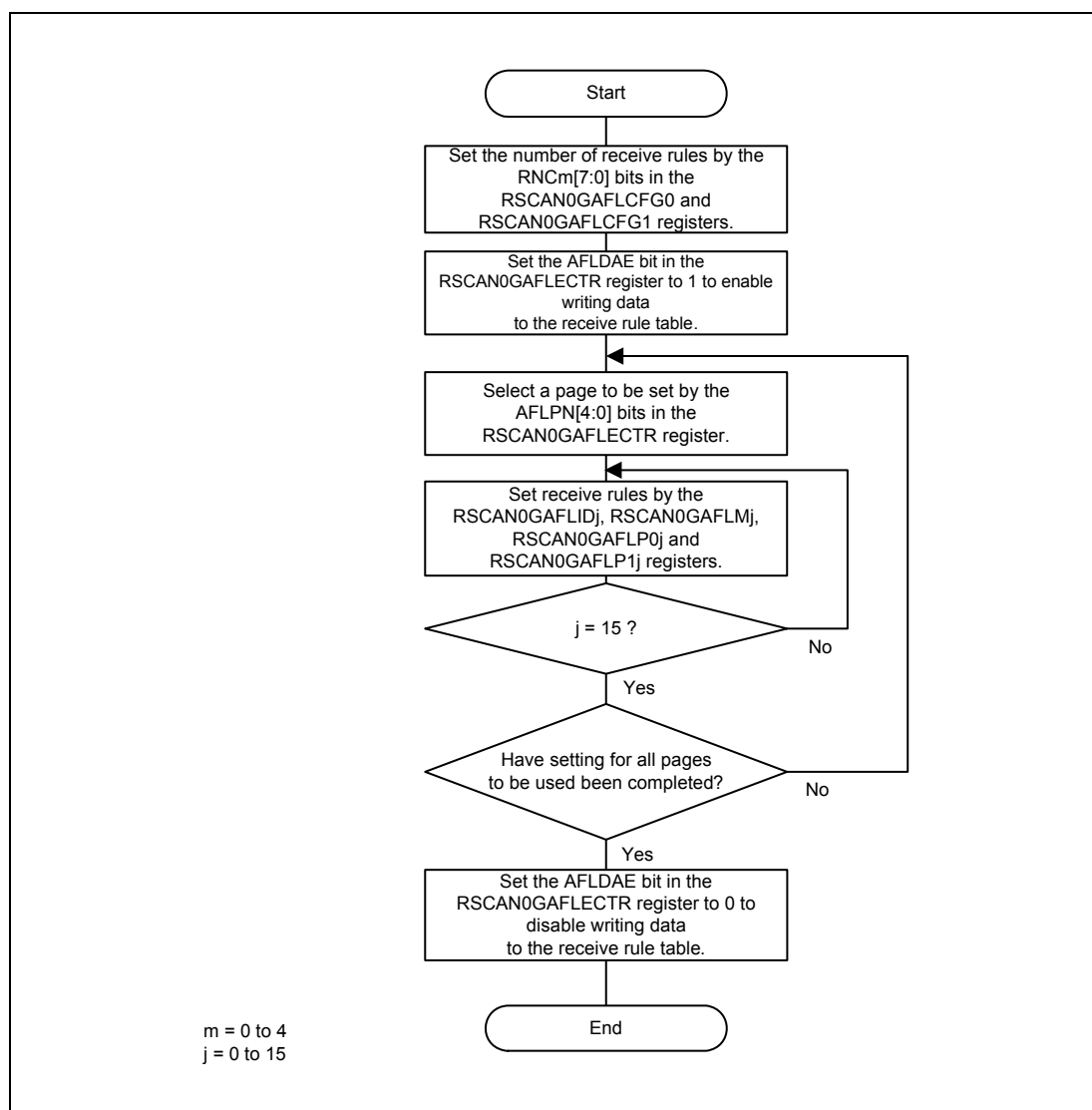
### 21.10.1.4 Receive Rule Setting

Receive rules can be set using receive rule-related registers.

Up to 16 receive rules can be registered per page. Specify pages 0 to 19 by the AFLPN[4:0] bits in the RSCAN0GAFLECTR register.

Set receive rule table write enable/disable using the AFLDAE bit.

**Figure 21.19** shows the receive rule setting procedure.



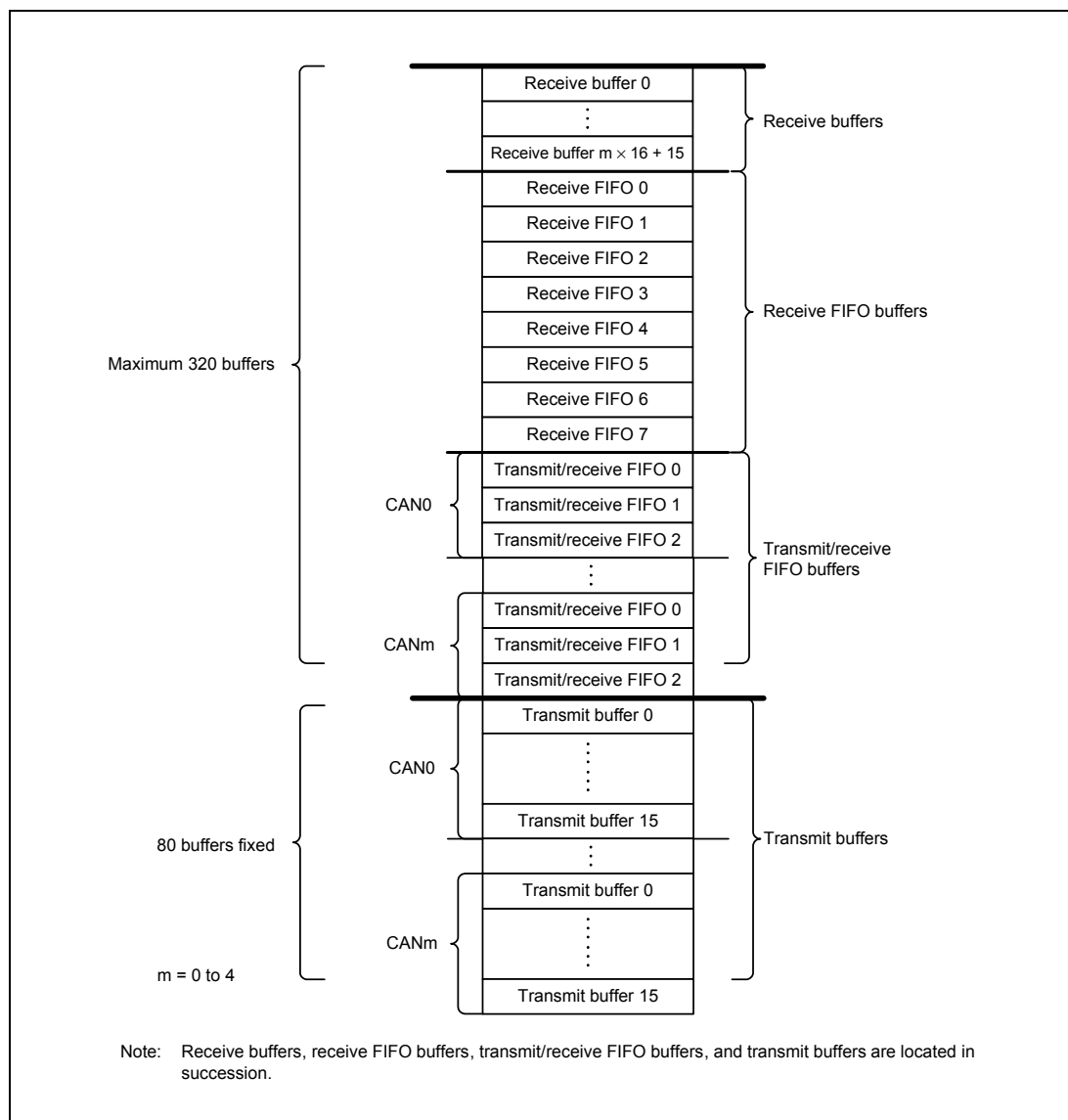
**Figure 21.19** Receive Rule Setting Procedure



### 21.10.1.5 Buffer Setting

Set sizes and interrupt sources of buffers. For transmit/receive FIFO buffers that are set to transmit mode, set transmit buffers to be linked.

**Figure 21.20** shows the buffer configuration. **Figure 21.21** shows the buffer setting procedure.



**Figure 21.20** Buffer Configuration

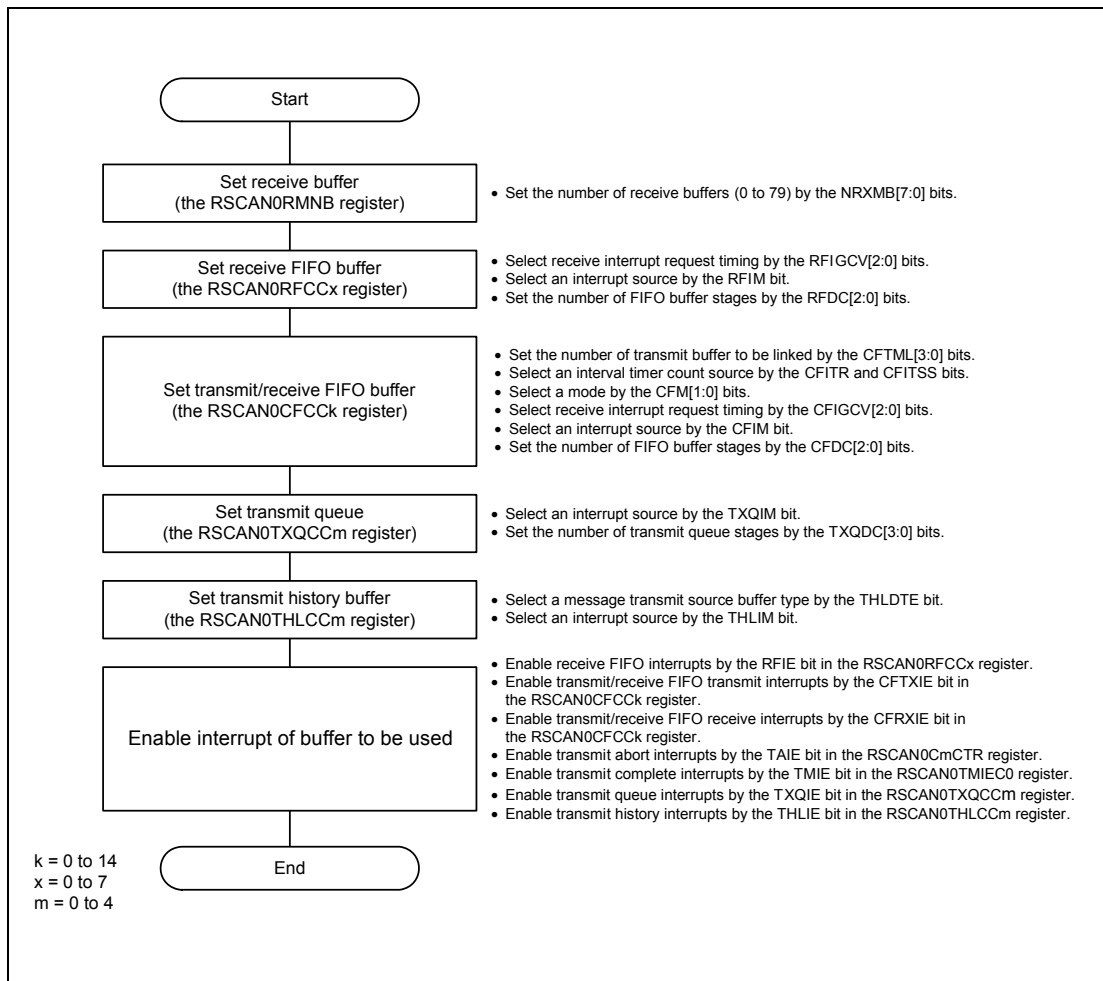


Figure 21.21 Buffer Setting Procedure

## 21.10.2 Reception Procedure

### 21.10.2.1 Receive Buffer Reading Procedure

When the processing to store received messages in a receive buffer starts, the RMNSq flag in the RSCAN0RMNDy register ( $y = 0$  to  $2$ ,  $q = 0$  to  $79$ ) is set to 1 (receive buffer  $q$  contains a new message). Messages can be read from the RSCAN0RMIDq, RSCAN0RMPTRq, RSCAN0RMDF0q, and RSCAN0RMDF1q registers. If the next message has been received before the current message is read from the receive buffer, the message is overwritten. **Figure 21.22** shows the receive buffer reading procedure.

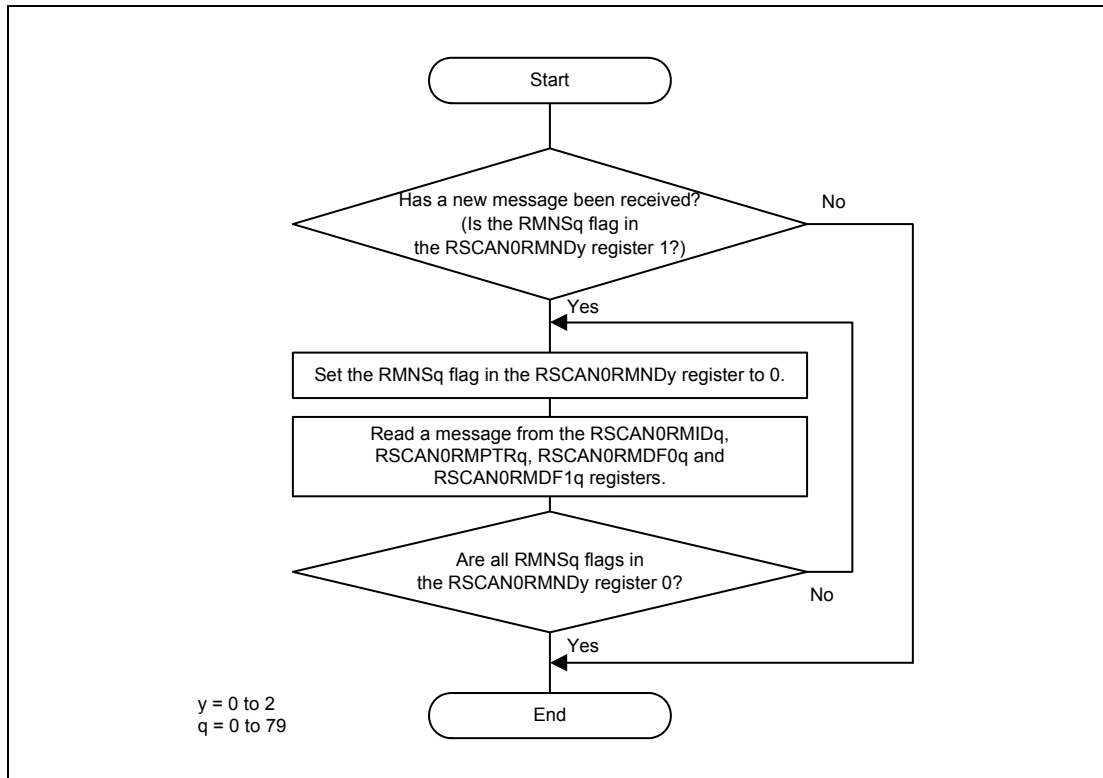
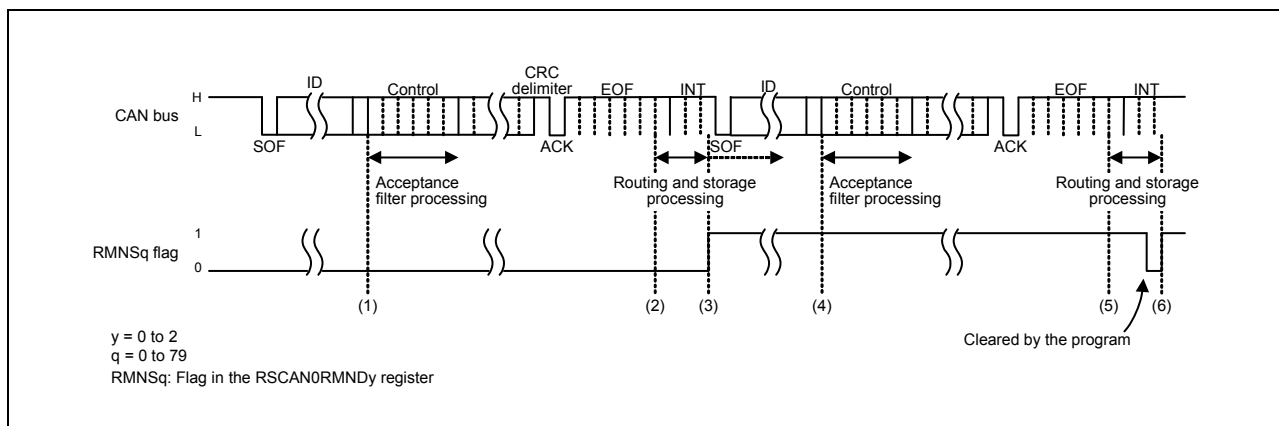


Figure 21.22 Receive Buffer Reading Procedure



**Figure 21.23 Receive Buffer Reception Timing Chart**

- (1) When the ID field in a message has been received, the acceptance filter processing starts.
- (2) When the message matches the receive rule of the corresponding channel and the message has been successfully received, the routing processing to transfer the message to the specified buffer starts. When the DCE bit in the RSCAN0GCFG register is set to 1 (DLC check is enabled), the DLC filter processing starts at this time.
- (3) When the message has passed through the DLC filter processing, the processing to store the message in the specified receive buffer starts.  
When the message storage processing starts, the RMNSq flag in the corresponding the RSCAN0RMNDy register is set to 1 (receive buffer n contains a new message). If other channels are performing filter processing or transmit priority determination processing, the routing processing and the storage processing may be delayed.
- (4) When the ID field of the next message has been received, the acceptance filter processing starts.
- (5) When the message matches the receive rule of the corresponding channel and the message has been successfully received, the routing processing to transfer the message to the specified buffer starts. When the DCE bit in the RSCAN0GCFG register is set to 1 (DLC check is enabled), the DLC filter processing starts at this time.
- (6) When the corresponding RMNSq flag is cleared to 0 (receive buffer n contains no new message), this flag is set to 1 again when the message storage processing starts. Even if the RMNSq flag remains 1, a new message is overwritten to the receive buffer. The RMNSq flag should not be cleared to 0 during storage of messages.

### 21.10.2.2 FIFO Buffer Reading Procedure

When received messages have been stored in one or more receive FIFO buffers or a transmit/receive FIFO buffer that is set to receive mode or gateway mode, the corresponding message count display counter (RFMC[7:0] bits in the RSCAN0RFSTSx register (x = 0 to 7) or CFMC[7:0] bits in the RSCAN0CFSTS<sub>k</sub> register (k = 0 to 14)) is incremented. At this time, when the RFIE bit (receive FIFO interrupt is enabled) in the RSCAN0RFCCx register or the CFRXIE bit (transmit/receive FIFO receive interrupt is enabled) in the RSCAN0CFCCk register is set to 1, an interrupt request is generated. Received messages can be read from the RSCAN0RFID<sub>x</sub>, RSCAN0RFPTR<sub>x</sub>, RSCAN0RFD<sub>0x</sub>, and RSCAN0RFD<sub>1x</sub> registers for receive FIFO buffers, or from the RSCAN0CFID<sub>k</sub>, RSCAN0CFPTR<sub>k</sub>, RSCAN0CFD<sub>0k</sub>, and RSCAN0CFD<sub>1k</sub> registers for transmit/receive FIFO buffers. Messages in FIFO buffers can be read sequentially on a first-in, first-out basis.

When the message count display counter value matches the FIFO buffer depth (a value set by the RFDC[2:0] bits in the RSCAN0RFCCx register or the CFDC[2:0] bits in the RSCAN0CFCCk register), the RFFLL or CFFLL flag is set to 1 (the receive FIFO buffer is full).

When all messages have been read out of the FIFO buffer, the RFEMP flag in the RSCAN0RFSTSx register or the CFEMP flag in the RSCAN0CFSTS<sub>k</sub> register is set to 1 (the receive FIFO buffer contains no unread message (buffer empty)).

If the RFE bit or the CFE bit is cleared to 0 (no receive FIFO buffer is used) with the interrupt request flag (RFIF flag in the RSCAN0RFSTSx register or CFRXIF flag in the RSCAN0CFSTS<sub>k</sub> register) set to 1 (a receive FIFO interrupt request is present), the interrupt request flag is not automatically cleared to 0. The program must clear the interrupt request flag to 0.

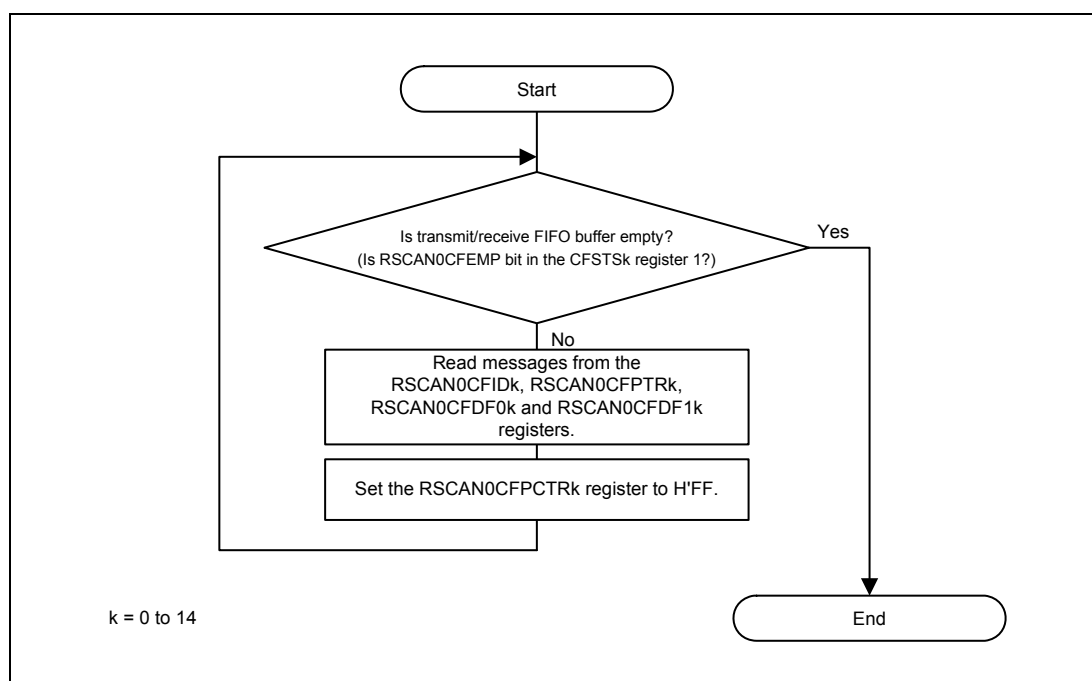


Figure 21.24 Transmit/Receive FIFO Buffer Reading Procedure

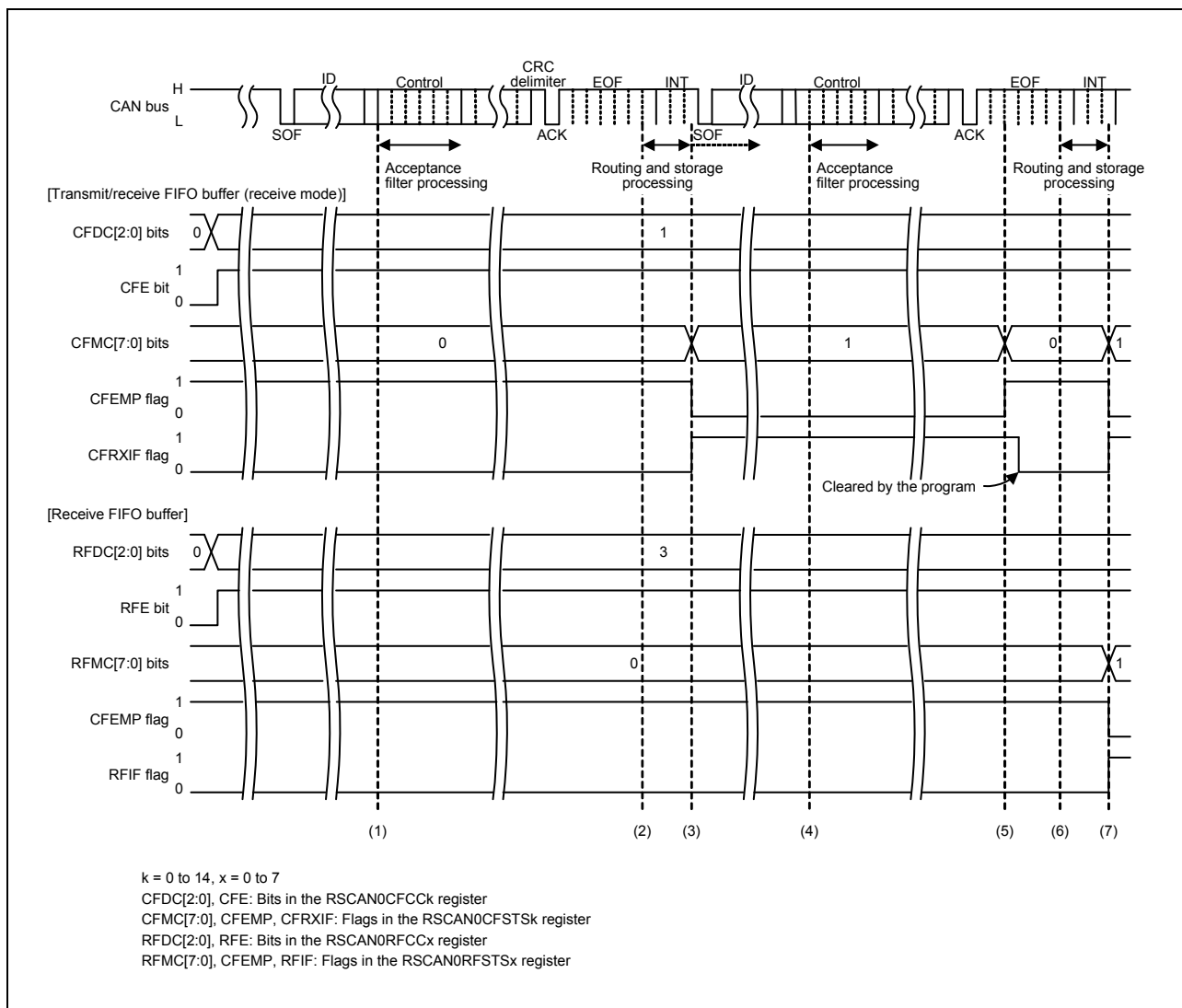


Figure 21.25 FIFO Buffer Reception Timing Chart

- (1) When the ID field in a message has been received, the acceptance filter processing starts.
- (2) When the message matches the receive rule of the corresponding channel and the message has been successfully received, the routing processing to transfer the message to the specified buffer starts. When the DCE bit in the RSCAN0GCFG register is set to 1 (DLC check is enabled), the DLC filter processing starts at this time.
- (3) When the message has passed through the DLC filter processing and the CFE value in the RSCAN0FCCK register is 1 (transmit/receive FIFO buffers are used) and the CFDC[2:0] value in the RSCAN0FCCK register is 001<sub>B</sub> or more, the message is stored in the transmit/receive FIFO buffer that is set to receive mode. The CFMC[7:0] value in the RSCAN0CFSTSk register is incremented and becomes H'01. When the CFIM bit in the RSCAN0FCCK register is set to 1 (a FIFO receive interrupt request is generated each time a message has been received), the CFRXIF flag in the RSCAN0CFSTSk register is set to 1 (a transmit/receive FIFO receive interrupt request is present). The CFRXIF flag can be reset to 0 by the program.
- (4) When the ID field of the next message has been received, the acceptance filter processing starts.
- (5) Read received messages from the RSCAN0CFIDk, RSCAN0CFPTRk, RSCAN0CFDF0k, and RSCAN0CFDF1k registers and write FF<sub>H</sub> to the RSCAN0CFPTRk register. This causes the

CFMC[7:0] bits in the RSCAN0CFSTSk register to be decremented. When CFMC[7:0] becomes 00<sub>H</sub>, the CFEMP flag in the RSCAN0CFSTSk register becomes 1 (the transmit/receive FIFO buffer contains no message (buffer empty)).

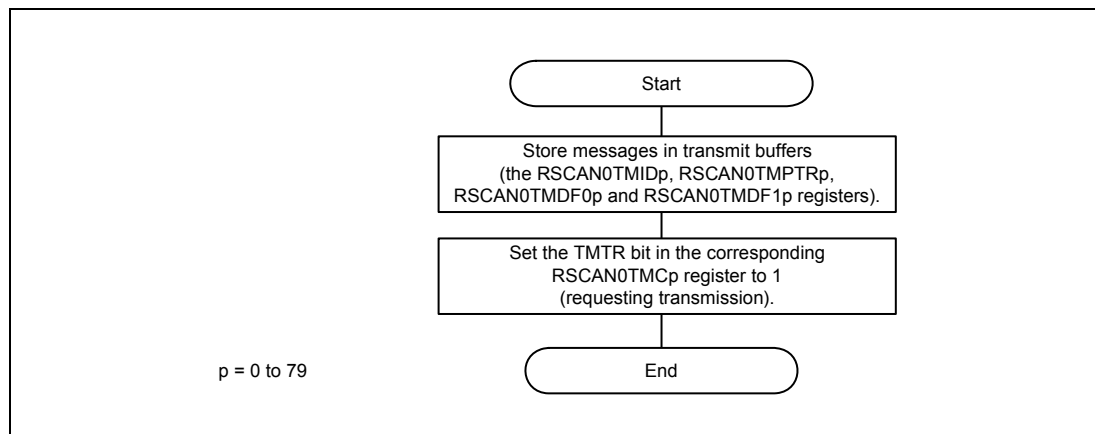
- (6) When the message matches the receive rule of the corresponding channel and the message has been successfully received, the routing processing to transfer the message to the specified buffer starts. When the DCE bit in the RSCAN0GCFG register is set to 1 (DLC check is enabled), the DLC filter processing starts at this time.
- (7) The message is stored in the transmit/receive FIFO buffer set in receive mode when the message has passed through the DLC filter process if the CFE bit is set to 1 (transmit/receive FIFO buffers are used), the RFE bit in the RSCAN0RFCCm register is set to 1, and the CFDC[2:0] bits are set to 001B or more. The CFMC[7:0] bit value is incremented by 1 to be 01H. When the CFIM bit is set to 1 (an interrupt occurs each time a message has been received), the CFRXIF flag is set to 1 (a transmit/receive FIFO receive interrupt request is present).  
The message is stored in the receive FIFO buffer if the RFE bit in the RSCAN0RFCCx register is set to 1 (receive FIFO buffers are used), and the RFDC[2:0] bits in the RSCAN0RFCCx register are set to 001B or more. The RFMC[7:0] bits in the RSCAN0RFSTsx register are set to 01H by being incremented by 1. When the RFIM bit in the RSCAN0RFCCx register is set to 1 (an interrupt occurs each time a message has been received), the RFIF flag in the RSCAN0RFSTsx register is set to 1 (a receive FIFO interrupt request is present).

## 21.10.3 Transmission Procedure

### 21.10.3.1 Procedure for Transmission from Transmit Buffers

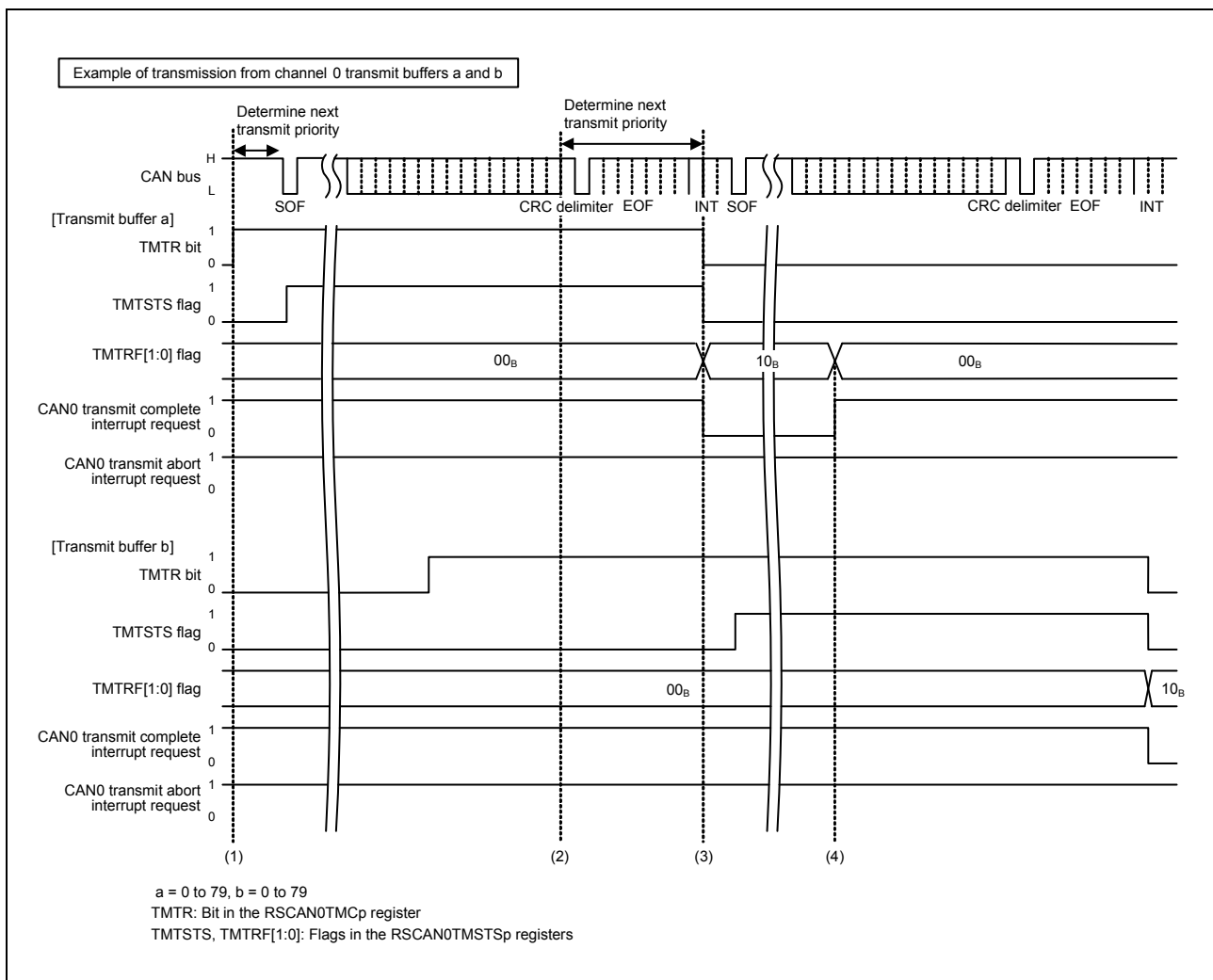
**Figure 21.26** shows the procedure for transmission from transmit buffers.

**Figure 21.27** shows a timing chart where messages are transmitted from two transmit buffers in the same channel and transmission has been successfully completed. **Figure 21.28** shows a timing chart where messages are transmitted from two transmit buffers in the same channel and transmit abort has been completed.



**Figure 21.26** Procedure for Transmission from Transmit Buffers





**Figure 21.27 Transmit Buffer Transmission Timing Chart (Transmission Completed Successfully)**

- (1) When the TMTR bit in the RSCAN0TMCa register is set to 1 while the CAN bus is idle, the transmit priority determination processing starts to determine the highest-priority transmit buffer. If transmit buffer a is determined to be the highest-priority transmit buffer, the TMTSTS flag in the corresponding the RSCAN0TMCa register is set to 1 (transmission is in progress) and the CAN channel starts transmitting data.
- (2) When a transmit request from a buffer is present, the priority determination starts with the CRC delimiter for the next transmission.
- (3) When transmission completes successfully, the TMTRF[1:0] flag in the RSCAN0TMCa register is set to 10<sub>B</sub> (transmission has been completed (without transmit abort request)) and the TMTSTS flag and the TMTR bit in the RSCAN0TMCa register are cleared to 0. When the TMIEa value in the RSCAN0TMIEC0 register is 1 (transmit buffer interrupt is enabled), a CAN0 transmit complete interrupt request is generated. To clear the interrupt request, set the TMTRF[1:0] flag to 00<sub>B</sub> (transmission is in progress or no transmit request is present).
- (4) Before starting the next transmission, set the TMTRF[1:0] flag to 00<sub>B</sub>. Write the next message to the transmit buffer, and then set the TMTR bit to 1 (transmission is requested). The TMTR bit can be set to 1 only when the TMTRF[1:0] flag value is 00<sub>B</sub>.

If an arbitration-lost has occurred after transmission is started, the TMTSTS flag is cleared to 0. The transmit priority determination is reexecuted at the beginning of the CRC delimiter to search the highest-priority transmit buffer. If an error has occurred during transmission or after arbitration loss, the priority determination processing is reexecuted during transmission of an error frame.

The timing the TMTSTS flag is set to 1 is not always the start timing of SOF. The start timing of SOF can be delayed up to the start timing of the basic ID due to the synchronization logic implemented for PLL bypass.

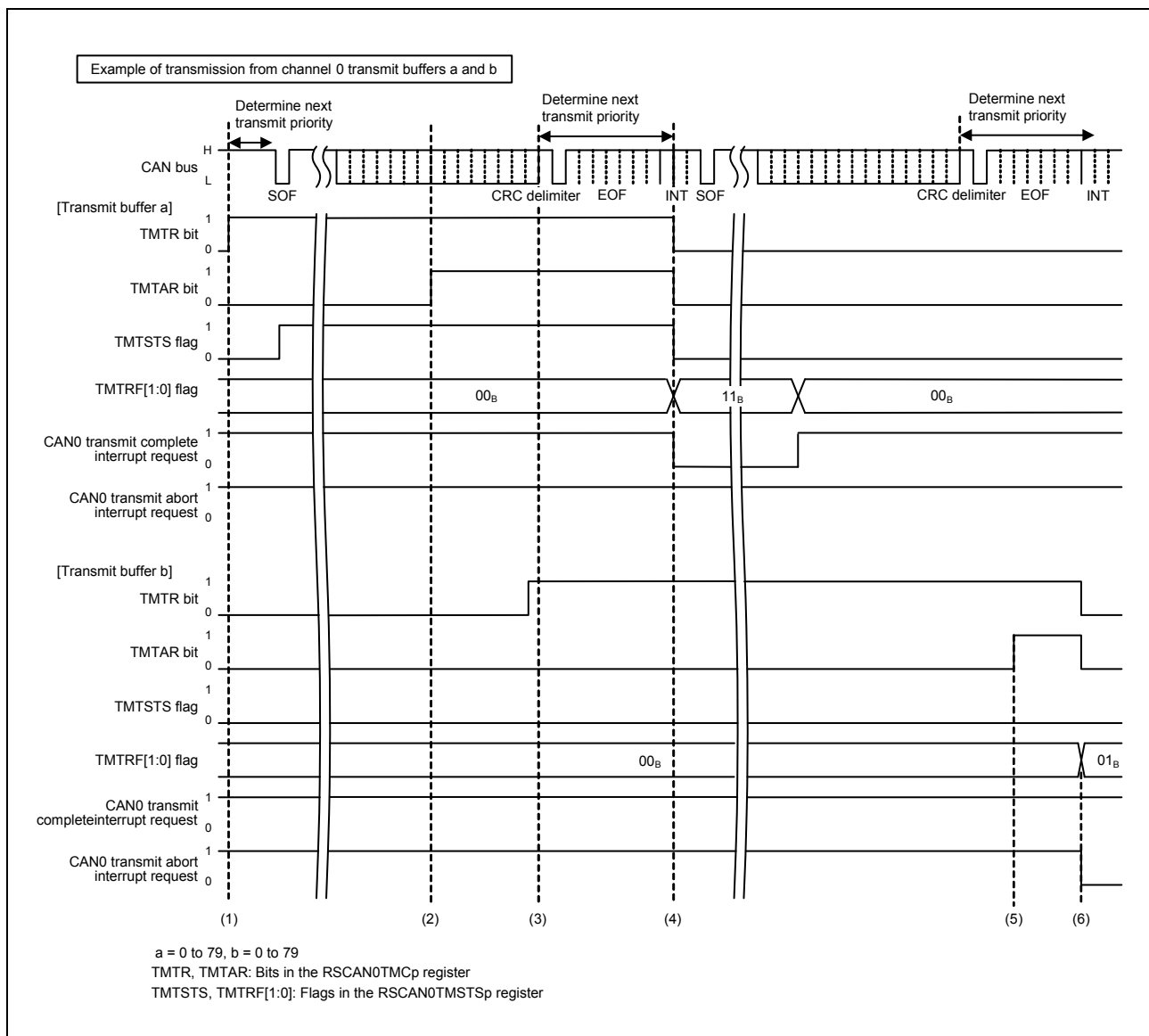


Figure 21.28 Transmit Buffer Transmission Timing Chart (Transmit Abort Completed)

- (1) When the TMTR bit in the RSCAN0TMCa register is set to 1 while the CAN bus is idle, the transmit priority determination processing starts to determine the highest-priority transmit buffer. If transmit buffer a is determined to be the highest-priority transmit buffer, the TMTSTS flag in the corresponding RSCAN0TMCa register is set to 1 (transmission is in progress) and the CAN channel starts transmitting data.
- (2) When it is determined that the transmit buffer is used for the next transmission or transmission is in progress, message transmission is not aborted unless an error or arbitration loss occurs even if the TMTAR bit is set to 1 (transmit abort is requested).

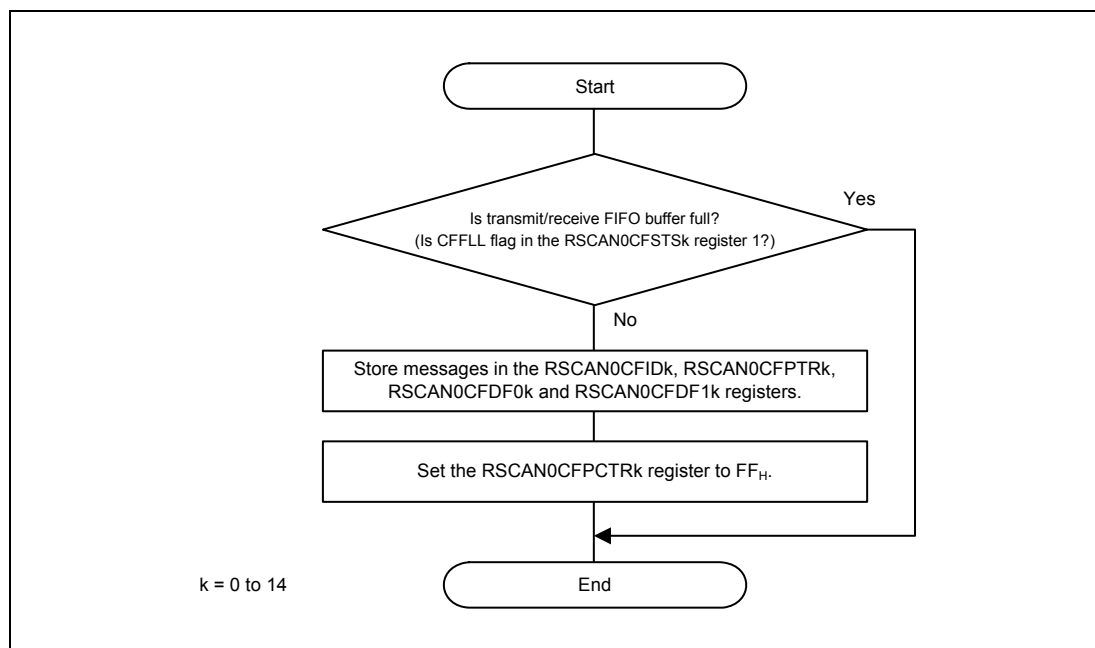
- (3) The priority determination starts with the CRC delimiter for the next transmission. In this timing chart, buffer b is not selected as the next transmit buffer.
- (4) When transmission completes successfully, the TMTRF[1:0] flag in the RSCAN0TMCa register is set to 11<sub>B</sub> (transmission has been completed (with transmit abort request)) and the TMTSTS flag and the TMTR bit in the RSCAN0TMCa register are cleared to 0. When the TMIEa value in the RSCAN0TMIEC0 register is 1 (transmit buffer interrupt is enabled), a CAN0 transmit complete interrupt request is generated. To clear the interrupt request, set the TMTRF[1:0] flag to 00<sub>B</sub> (transmission is in progress or no transmit request is present).
- (5) While another CAN node is transmitting data on the CAN bus (TMTSTS flag = 0), if the TMTAR bit is set to 1 while the corresponding channel is determining transmit priority, the TMTR bit cannot be cleared to 0.
- (6) After the internal processing time has passed, the transmission is terminated and the TMTRF[1:0] flag is set to 01<sub>B</sub>. When the transmit buffer is not transmitting data and is not selected as the next transmit buffer and priority determination is not being made, an abort request is immediately accepted and the TMTRF[1:0] flag is set to 01<sub>B</sub>. At this time, the TMTR and TMTAR bits are cleared to 0. When transmit abort is completed with the TAIE bit in the RSCAN0CmCTR register set to 1 (transmit abort interrupt is enabled), an interrupt request is generated. To clear the interrupt request, set the TMTRF[1:0] flag to 00<sub>B</sub>.

If an arbitration loss has occurred after the CAN channel started transmission, the TMTSTS bit is cleared to 0. The transmit priority determination is reexecuted at the beginning of the CRC delimiter to find the highest-priority transmit buffer. If an error has occurred during transmission or after arbitration loss, the priority determination processing is reexecuted during transmission of an error frame.

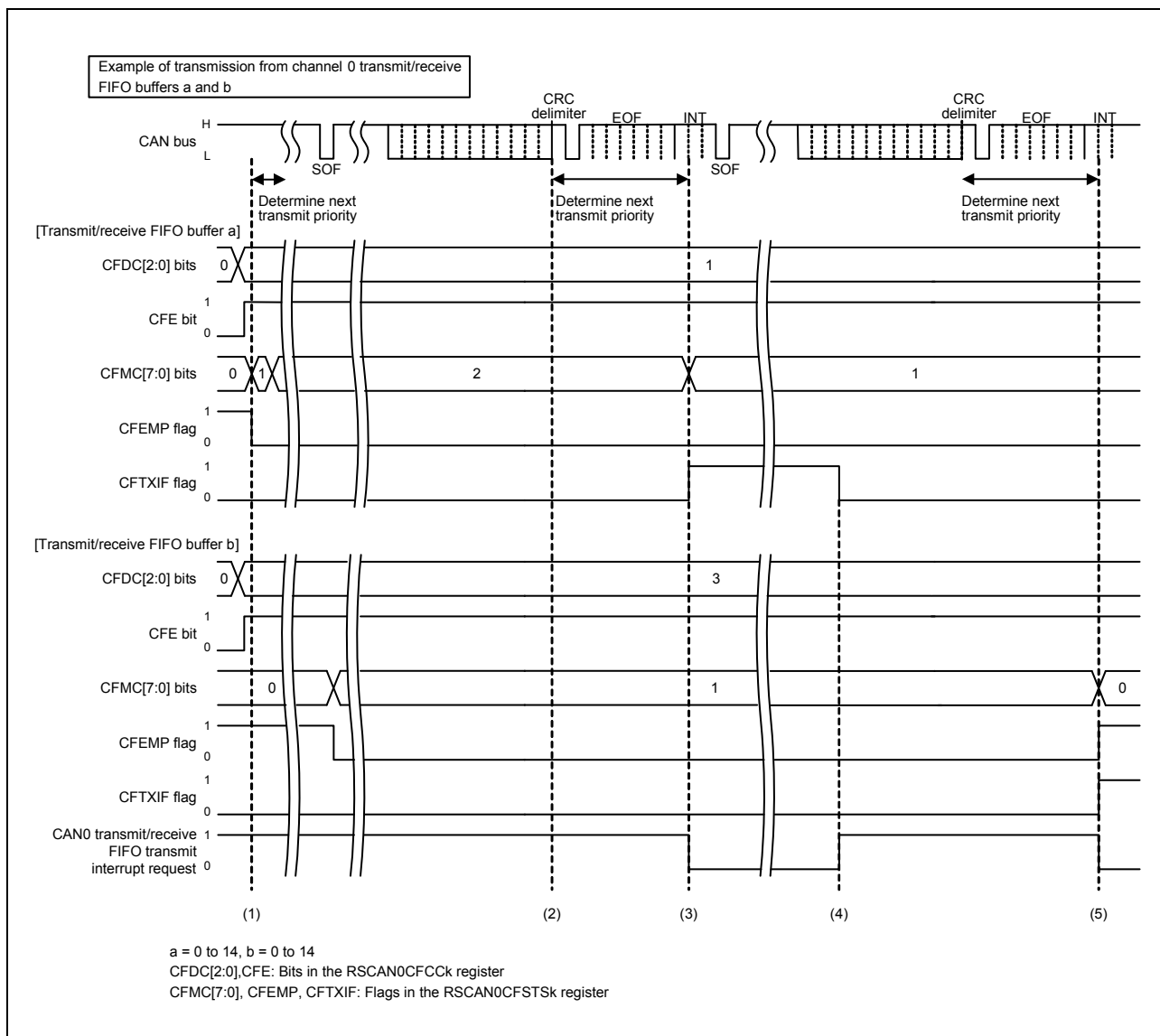
### 21.10.3.2 Procedure for Transmission from Transmit/Receive FIFO Buffers

**Figure 21.29** shows the procedure for transmission from transmit/receive FIFO buffers.

**Figure 21.30** shows a timing chart where messages are transmitted from two transmit/receive FIFO buffers in the same channel and transmission has been successfully completed. **Figure 21.31** shows a timing chart where messages are transmitted from two transmit/receive FIFO buffers in the same channel and transmit abort has been completed.



**Figure 21.29** Procedure for Transmission from Transmit/Receive FIFO Buffers

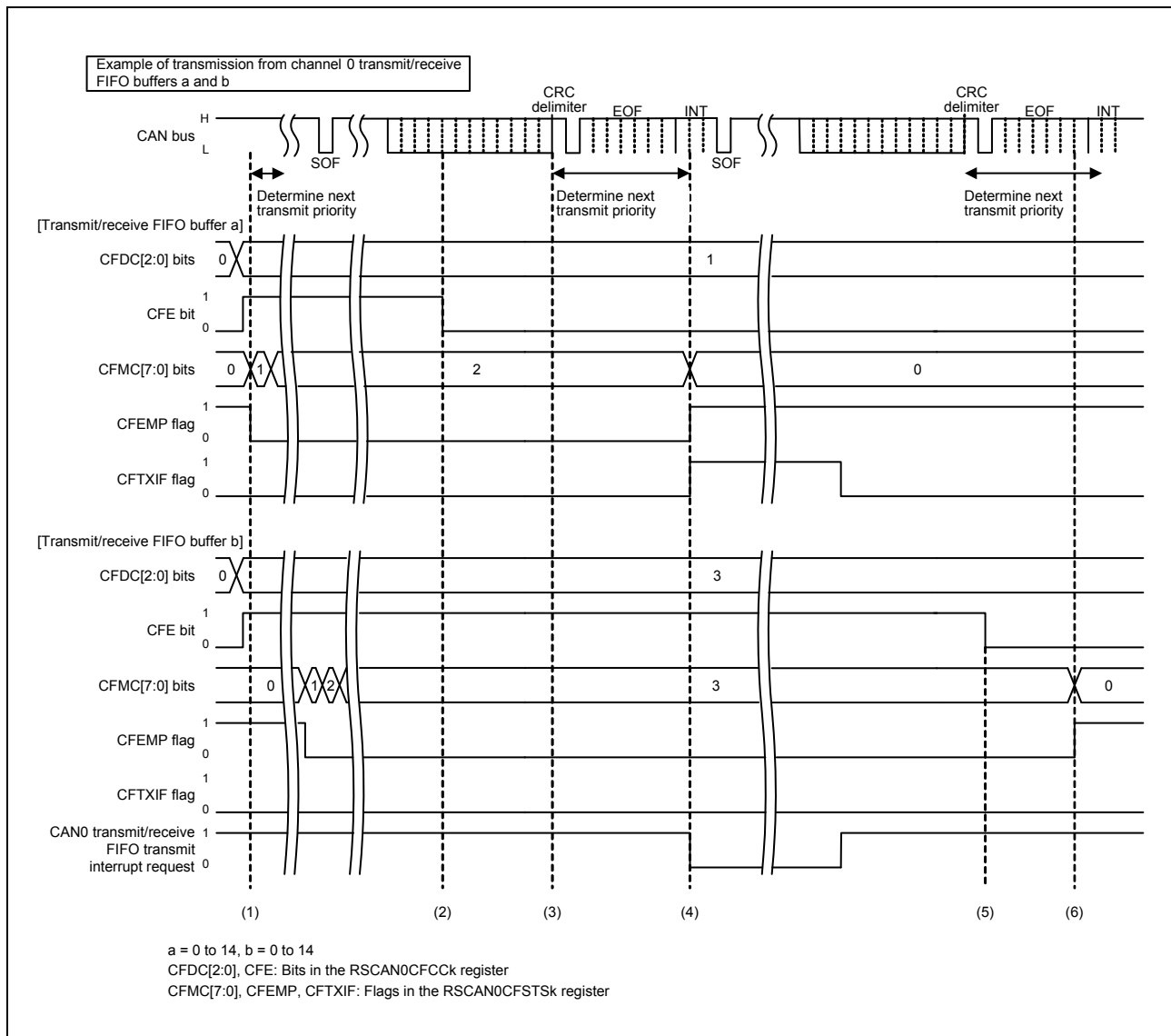


**Figure 21.30 Transmit/Receive FIFO Buffer Transmission Timing Chart (Transmission Completed Successfully)**

- (1) While the CAN bus is idle, when the CFE value in the RSCAN0CFCCa register is 1 (transmit/receive FIFO buffers are used) and the CFDC[2:0] value in the RSCAN0CFCCa register is 001<sub>B</sub> (4 messages) or more and the CFMC[7:0] value in the RSCAN0CFSTSa register is H'01 or more, the priority determination processing starts to determine the highest-priority transmit message. When the highest-priority transmit message has been determined, transmission of the message starts. In this figure, the message is transmitted from transmit/receive FIFO buffer a of channel 0.
- (2) When a transmit request from a buffer is present, the priority determination starts with the CRC delimiter for the next transmission.
- (3) When transmission completes successfully, the CFMC[7:0] value in the RSCAN0CFSTSa register is decremented. Setting the CFIM bit in the RSCAN0CFCCa register to 1 (a FIFO transmit interrupt request is generated each time a message has been transmitted) sets the CFTXIF flag in the RSCAN0CFSTSa register to 1 (a transmit/receive FIFO transmit interrupt request is present).
- (4) The program can clear the CFTXIF flag.

(5) Message transmission from transmit/receive FIFO buffer b of channel 0 has been completed and the CFMC[7:0] value in the RSCAN0CFSTSb register is decremented. The CFMC[7:0] bits are cleared to 00<sub>H</sub> and therefore the CFEMP flag in the RSCAN0CFSTS<sub>k</sub> register is set to 1 (the transmit/receive FIFO buffer contains no message (buffer empty)).

Transmission is continued until the CFEMP flag is set to 1. It is possible to continuously store transmit messages in FIFO buffers until the CFFLL flag in the RSCAN0CFSTS<sub>a</sub> and RSCAN0CFSTS<sub>b</sub> register is set to 1 (the transmit/receive FIFO buffer is full).



**Figure 21.31 Transmit/Receive FIFO Buffer Transmission Timing Chart (Transmit Abort Completed)**

(1) While the CAN bus is idle, when the CFE value in the RSCAN0CFCC<sub>a</sub> register (a = 0 to 14) is 1 (transmit/receive FIFO buffers are used) and the CFDC[2:0] value in the RSCAN0CFCC<sub>a</sub> register is 001<sub>B</sub> (4 messages) or more and the CFMC[7:0] value in the RSCAN0CFSTS<sub>a</sub> register is H'01 or more, the priority determination processing starts to determine the highest-priority transmit message. When the highest-priority transmit message has been determined, transmission of the message starts. In this figure, the message is transmitted from transmit/receive FIFO buffer a of channel 0.

- (2) When transmission is in progress or it is determined that the transmit/receive FIFO buffer is used for the next transmission, message transmission is not aborted unless an error or arbitration loss occurs even if the CFE bit is set to 0 (no transmit/receive FIFO buffer is used).
- (3) When a transmit request from a buffer is present, the priority determination starts with the CRC delimiter for the next transmission. In this figure, transmit/receive FIFO buffer b is not selected as a buffer for the next transmission.
- (4) When transmit completes successfully, the CFMC[7:0] value is cleared to 00<sub>H</sub>. Setting the CFIM bit to 1 (a FIFO transmit interrupt request is generated each time a message has been transmitted) sets the CFTXIF flag in the RSCAN0CFSTSa register to 1 (a transmit/receive FIFO transmit interrupt request is present). The program can clear the CFTXIF flag.
- (5) If another CAN node on the CAN bus is transmitting data (not from transmit/receive FIFO buffer b), transmit/receive FIFO buffers cannot be disabled immediately even if the CFE bit in the RSCAN0CFCCb register is cleared to 0 (no transmit/receive FIFO buffer is used) during transmit priority determination. (The CFEMP flag in the RSCAN0CFSTSB register is not set to 1 (the transmit/receive FIFO buffer contains no message (buffer empty)) immediately.)
- (6) After the internal processing time has passed, transmit/receive FIFO buffers are disabled and the CFMC[7:0] bits in the RSCAN0CFSTSB register are cleared to 00<sub>H</sub> and the CFEMP flag is set to 1. When the transmit/receive FIFO buffer is not transmitting data and is not selected as the next transmit buffer and priority determination is not in progress, the transmit/receive FIFO buffer is immediately disabled. (The CFMC[7:0] bits are cleared to 00<sub>H</sub> and the CFEMP flag is set to 1.)

### 21.10.3.3 Procedure for Transmission from the Transmit Queue

Figure 21.32 shows the procedure for transmission from the transmit queue.

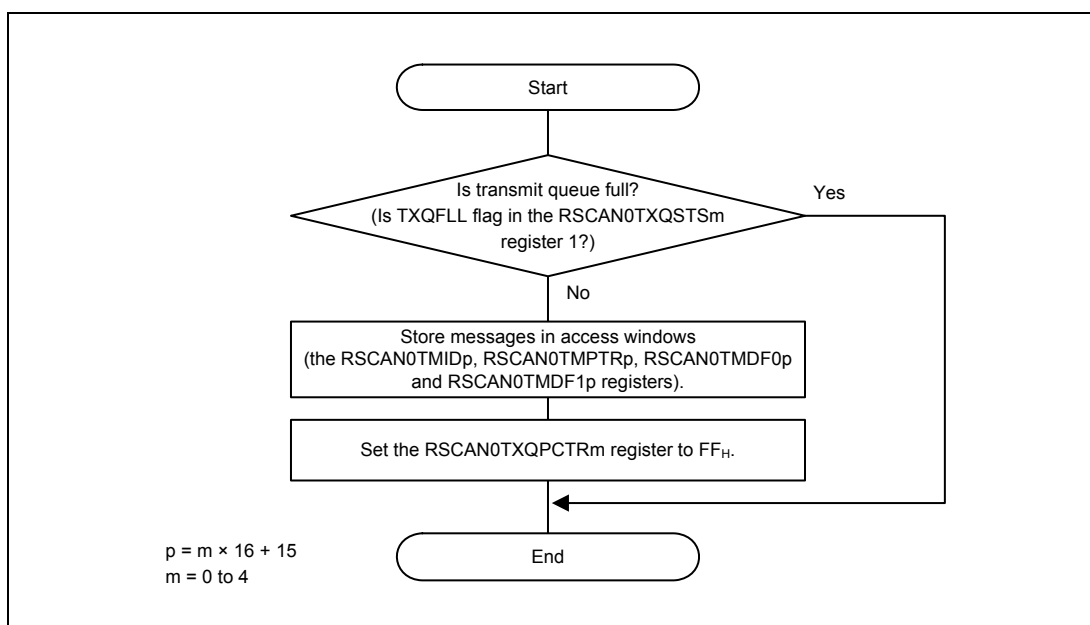


Figure 21.32 Procedure for Transmission from the Transmit Queue

### 21.10.3.4 Transmit History Buffer Reading Procedure

Transmit history data can be read from the RSCAN0THLACCm register. The next data can be accessed by writing H'FF to the corresponding RSCAN0THLPCTRm register (m = 0 to 4) after reading a set of data. **Figure 21.33** shows the transmit history buffer reading procedure.

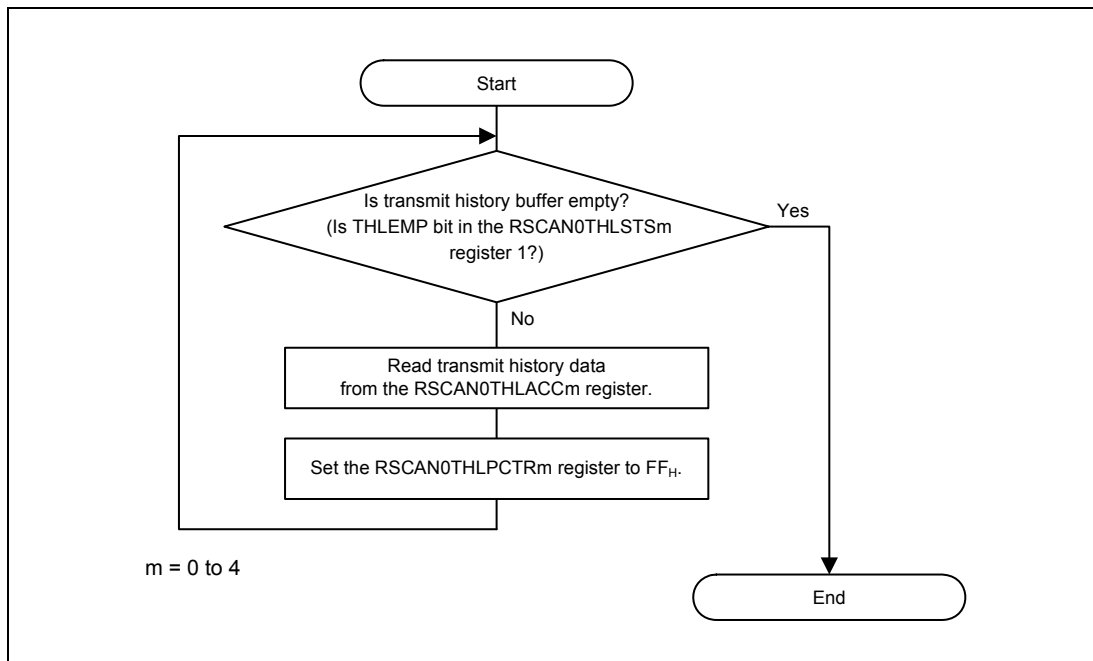


Figure 21.33 Transmit History Buffer Reading Procedure



## 21.10.4 Test Settings

### 21.10.4.1 Self-Test Mode Setting Procedure

Self-test mode allows communication test on a channel basis by enabling a CAN node to receive its own transmitted messages.

Figure 21.34 shows the self-test mode setting procedure.

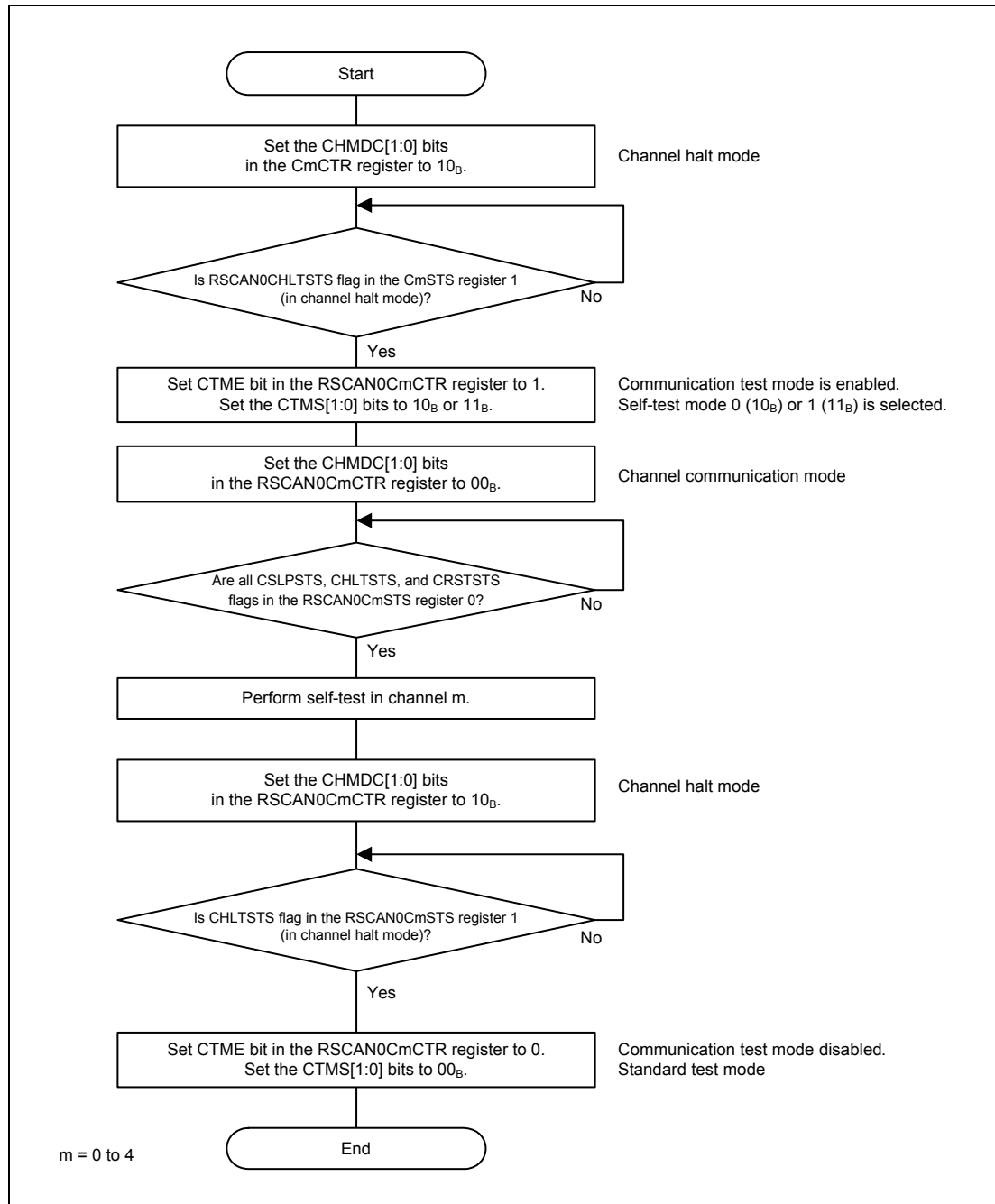


Figure 21.34 Self-Test Mode Setting Procedure

### 21.10.4.2 Inter-Channel Communication Test Setting Procedure

Communication testing can be performed by transmitting and receiving data between different channels.

Figure 21.35 shows the inter-channel communication test setting procedure.

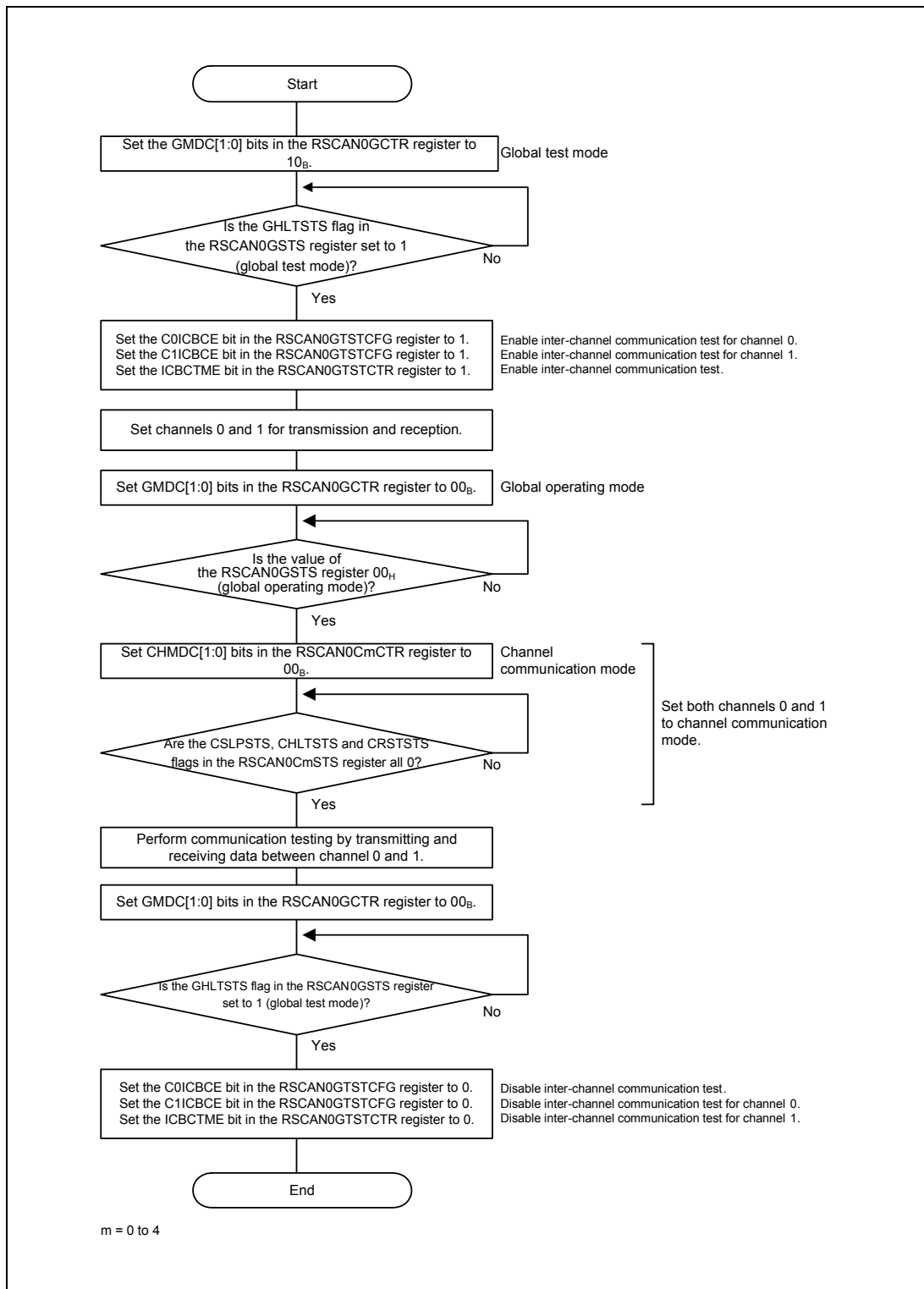


Figure 21.35 Inter-Channel Communication Test Setting Procedure(Example of Communication Test between Channel 0 and Channel 1)

## 21.11 Notes on the RS-CAN Module

- When changing a global mode, check the GSLPSTS, GHLTSTS, and GRSTSTS flags in the RSCAN0GSTS register for transitions. When changing a channel mode, check the CSLPSTS, CHLTSTS, and CRSTSTS flags in the RSCAN0CmSTS register (m = 0 to 4) for transitions.
- The acceptance filter processing checks receive rules sequentially in ascending order from the minimum rule number. If the same ID, IDE bit, or RTR bit value is set for multiple receive rules, the minimum number of receive rule is used for the acceptance filter processing. If the message does not pass through the subsequent DLC filter processing, the data processing is terminated without returning to the acceptance filter processing and the message is not stored in the buffer.
- When linking transmit buffers to transmit/receive FIFO buffers or allocating transmit buffers to transmit queues, set the control register (RSCAN0TMCp) of the corresponding transmit buffer to 00<sub>H</sub>. The status register (RSCAN0TMSTSp) of the corresponding transmit buffer should not be used. Flags in other status registers (registers RSCAN0TMTRSTS0, RSCAN0TMTARSTS0, RSCAN0TMTTCSTS0, and TMTASTS0RSCAN0), which correspond to transmit buffers linked to transmit/receive FIFO buffers or allocated to transmit queues remain unchanged. Set the enable bit in the corresponding interrupt enable register (the RSCAN0TMIEC0 register) to 0 (transmit buffer interrupt is disabled).
- Transmit buffers that are linked to transmit/receive FIFO buffers must not be allocated to transmit queues.
- Only a single transmit/receive FIFO buffer can be linked to a transmit buffer. Do not link two or more transmit/receive FIFO buffers to transmit buffers of the same number.
- When the CANm bit time clock is selected as a timestamp counter clock source, the timestamp counter stops when the corresponding channel has transitioned to channel reset mode or channel halt mode.
- In case of an attempt to store a new received message when the receive FIFO buffer and the transmit/receive FIFO buffer are full, the new message is discarded. If you wish to store a new transmit message in the transmit/receive FIFO buffer or the transmit queue, check that the transmit/receive FIFO buffer or the transmit queue is not full.

## 22. IEBus Controller

IEBus (Inter Equipment Bus) is a small-scale digital data transfer system that transfers data between units. To implement IEBus with this LSI, an external IEBus driver and receiver are necessary because they are not provided.

The internal IEBus controllers of this LSI are of negative logic.

### 22.1 IEBC Features

**Instances** This microcontroller has the following number of instances of the IEBC.

**Table 22-1** Instances of IEBC

IEBC	
Number of instances	1
Name	IEBC0

**Instances index n** Throughout this chapter, the individual instances of IEBC are identified by the index “n” (n = 0), for example, IEBCnBCR for the IEBCn bus control register.

**Register addresses** All IEBCn register addresses are given as addresses offset from the base address <IEBCn\_base>.

The base address <IEBCn\_base> of each IEBCn is listed in the following table:

**Table 22-2** Register base address <IEBCn\_base>

IEBCn	<IEBCn_base>
IEBC0	FCFEF000

**Clock supply** The following clock is supplied to IEBCn:

**Table 22-3** IEBCn clock supply

IEBCn	Clock	Connected to:
IEBC0	P0φ	Clock pulse generator

**Interrupts and DMA** IEBB can generate the following interrupt requests and DMA requests:

**Table 22-4 IEBBn interrupts and DMA requests**

Interrupt request signal	Function	Direct memory access controller activation
IEBBTD	Data interrupt request	√
IEBBTV	Vector interrupt request	√
IEBBTERR	Error interrupt request	—
IEBBTSTA	Status interrupt request	—

**I/O signals** The I/O signals of IEBB0 are listed in the following table:

**Table 22-5 IEBBn I/O signals**

Pin name	Function	Input/output
IERxD	IEBB0 reception data	Input
IETxD	IEBB0 transmission data	Output

## 22.2 Configuration

### 22.2.1 Function overview

- Features summary**
- **The data transfer system complies with the IEBus (communication mode 1/communication mode 2) protocol.**
  - **The IEBus (Inter Equipment Bus) controller is mainly intended to transfer data between automotive devices by using a two-line serial bus interface.**
  - Effective transmission speed: Approximately 17 kbps (communication mode 1), or approximately 26 kbps (communication mode 2)
  - The single mode or FIFO mode can be selected.
  - Maximum number of transferred bytes:
    - 32 bytes/frame (communication mode 1)
    - 128 bytes/frame (communication mode 2)
  - To implement IEBus, an external IEBus driver and receiver are necessary. The driver and receiver are not built in.
  - This IEBus controller uses negative logic.
  - Operation clock: 8 MHz (input P0 $\phi$  = 32 MHz to this module)
  - Interrupt request signal
    - Data interrupt (IEBBTD)
      - For transmission data write processing (single mode, FIFO mode)
      - For reception data read processing (single mode)
    - Error interrupt (IEBBTERR)
      - For error processing
    - Status interrupt (IEBBTSTA)
      - Start interrupt processing (single mode)
      - Status transmission interrupt (single mode)
      - Communication completion interrupt (single mode, FIFO mode)
      - Frame completion interrupt (single mode, FIFO mode)
    - Vector interrupt (IEBBTV)
      - Occurs at the same time as IEBBTERR or IEBBTSTA (single mode)
      - For reception data read processing (FIFO mode)
  - Pin configuration
    - IERxD: IEBus reception data input signal
    - IETxD: IEBus transmission data output signal

22.2.2 Block diagram

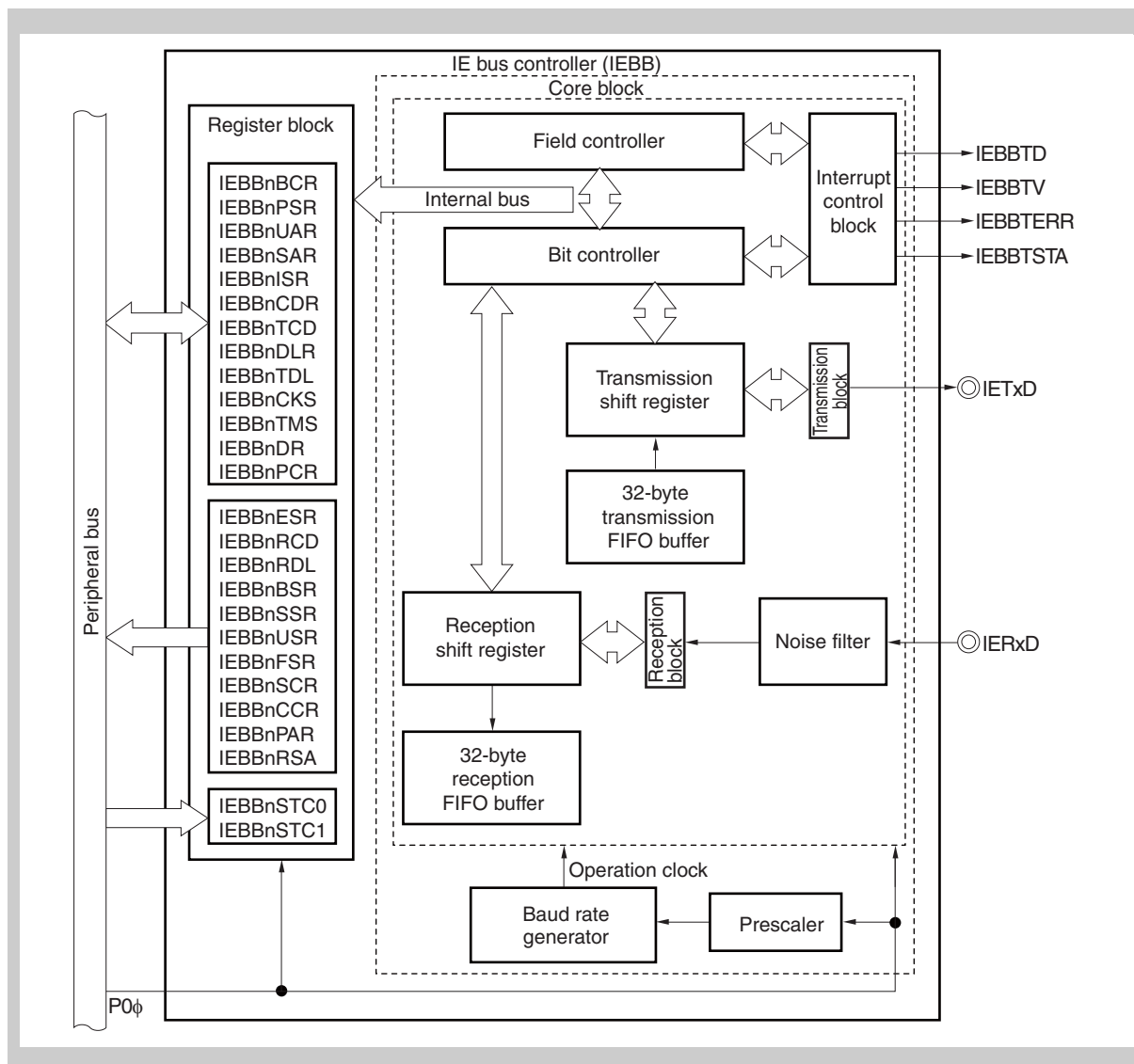


Figure 22-1 IEBn block diagram

## 22.3 Registers

### 22.3.1 IEBBn register overview

IEBBn is controlled and operated by the following registers:

Table 22-6 IEBBn registers

Register name	Symbol	Address
IEBBn bus control register	IEBBnBCR	<IEBBn_base> + 0000 <sub>H</sub>
IEBBn power save register	IEBBnPSR	<IEBBn_base> + 0004 <sub>H</sub>
IEBBn unit address register	IEBBnUAR	<IEBBn_base> + 0008 <sub>H</sub>
IEBBn slave address register	IEBBnSAR	<IEBBn_base> + 000C <sub>H</sub>
IEBBn partner address register	IEBBnPAR	<IEBBn_base> + 0010 <sub>H</sub>
IEBBn reception slave address register	IEBBnRSA	<IEBBn_base> + 0014 <sub>H</sub>
IEBBn control data register	IEBBnCDR	<IEBBn_base> + 0018 <sub>H</sub>
IEBBn transmission control data register	IEBBnTCD	<IEBBn_base> + 001C <sub>H</sub>
IEBBn reception control data register	IEBBnRCD	<IEBBn_base> + 0020 <sub>H</sub>
IEBBn message length register	IEBBnDLR	<IEBBn_base> + 0024 <sub>H</sub>
IEBBn transmission message length register	IEBBnTDL	<IEBBn_base> + 0028 <sub>H</sub>
IEBBn reception message length register	IEBBnRDL	<IEBBn_base> + 002C <sub>H</sub>
IEBBn clock selection register	IEBBnCKS	<IEBBn_base> + 0030 <sub>H</sub>
IEBBn transfer mode setting register	IEBBnTMS	<IEBBn_base> + 0034 <sub>H</sub>
IEBBn pointer clear register	IEBBnPCR	<IEBBn_base> + 0038 <sub>H</sub>
IEBBn buffer status register	IEBBnBSR	<IEBBn_base> + 003C <sub>H</sub>
IEBBn slave status register	IEBBnSSR	<IEBBn_base> + 0040 <sub>H</sub>
IEBBn unit status register	IEBBnUSR	<IEBBn_base> + 0044 <sub>H</sub>
IEBBn interrupt status register	IEBBnISR	<IEBBn_base> + 0048 <sub>H</sub>
IEBBn error status register	IEBBnESR	<IEBBn_base> + 004C <sub>H</sub>
IEBBn field status register	IEBBnFSR	<IEBBn_base> + 0050 <sub>H</sub>
IEBBn success count register	IEBBnSCR	<IEBBn_base> + 0054 <sub>H</sub>
IEBBn communication count register	IEBBnCCR	<IEBBn_base> + 0058 <sub>H</sub>
IEBBn status clear register 0	IEBBnSTC0	<IEBBn_base> + 005C <sub>H</sub>
IEBBn status clear register 1	IEBBnSTC1	<IEBBn_base> + 0060 <sub>H</sub>
IEBBn data register	IEBBnDR	<IEBBn_base> + 0064 <sub>H</sub>

<IEBBn\_base> The IEBBn base address <IEBBn\_base> is defined in Table 22-2 "Register base address <IEBBn\_base>".



### 22.3.2 IEBBn control register details

#### (1) IEBBnBCR - IEBBn bus control register

The IEBBnBCR register is used to control the operation of IEBBn.

**Access** This register can be read or written in 8-bit units.

**Address** <IEBBn\_base> + 0000H

**Initial value** 00H

The IEBBnMSRQ, IEBBnALRQ, IEBBnSTXE, and IEBBnSRXE bits are reset by writing 0 to the IEBBnPW bit.

- Cautions**
- When operation is enabled (when the IEBBnPW bit = 1), writing 1 to the IEBBnMSRQ bit is prohibited while the IEBBnMSRQ bit = 1. To write 1 to this bit, first clear it to 0.
  - Note the following when accessing the register:
    - When the IEBBnPW bit = 0, it is not possible to write to the IEBBnMSRQ, IEBBnALRQ, IEBBnSTXE, and IEBBnSRXE bits.
    - Because the IEBBnMSRQ, IEBBnALRQ, IEBBnSTXE, and IEBBnSRXE bits are reset at the same time by writing 0 to the IEBBnPW bit, even if an 8-bit write that results in the IEBBnPW bit being cleared to 0 is performed, the IEBBnMSRQ, IEBBnALRQ, IEBBnSTXE, and IEBBnSRXE bits are not written to. When an 8-bit write that results in the IEBBnPW bit being set to 1 is performed, the IEBBnMSRQ, IEBBnALRQ, IEBBnSTXE, and IEBBnSRXE bits can be written to.

Example: If 78H is written to the IEBBnBCR register and then the register is read, 00H is returned.

If F8H is written to the IEBBnBCR register and then the register is read, F8H is returned.

7	6	5	4	3	2	1	0
IEBBn PW	IEBBn MSRQ	IEBBn ALRQ	IEBBn STXE	IEBBn SRXE	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R

Table 22-7 IEBBnBCR register contents

Bit position	Bit name	Function
7	IEBBnPW	<p>Communication enable flag            0: Stop IEBBn unit operation.            1: Enable IEBBn unit operation.</p> <p><b>Caution</b> When the IEBBnPW bit is set (to 1), the IEBBnPSR, IEBBnUAR, IEBBnCKS, and IEBBnTMS registers below cannot be overwritten. Therefore, these registers must be set up before setting the IEBBnPW bit.</p>
6	IEBBnMSR Q	<p>Master request flag            0: Do not request the IEBBn unit as the master.            1: Request the IEBBn unit as the master.</p>
5	IEBBnALR Q	<p>Broadcast request flag            0: Request individual communication.            1: Request broadcast communication.</p>
4	IEBBnSTX E	<p>Slave transmission enable flag            0: Disable slave transmission.            1: Enable slave transmission.</p>
3	IEBBnSRX E	<p>Slave reception enable flag            0: Disable slave reception.            1: Enable slave reception.</p>

**(a) Communication enable flag (IEBBnPW): Bit 7**

- Set/clear condition

Set: By software (Write 1 to the IEBBnPW bit.)

Clear: By software (Write 0 to the IEBBnPW bit.)

Depending on when the IEBBnPW bit is set (to 1), the IEBBn communication participation method differs.

**Table 22-8 IEBBnPW bit setting timing and communication participation method**

Timing for setting the IEBBnPW bit (to 1)	IEBBn communication participation method
When communication is not being performed on IEBus	Communication is participated in starting at the next frame or communication is started.
When communication is being performed on IEBus, and start bit communication is being performed by another bus master	Participates in communication from that frame if the start bit is detected. If the start bit is not detected, participates in communication from the next frame.
When communication is being performed on IEBus, and post-start bit communication is being performed by another bus master	Participates in communication from the next frame.

If the IEBBnPW bit is cleared (to 0), communication is immediately stopped even if it is in progress, and the internal flags and registers are reset, with some exceptions. The registers that are not reset by the IEBBnPW bit are shown below.

When the IEBBnPW = 0, even if another unit starts communication, IEBBn does not respond.

**Table 22-9 Registers that are not reset by the IEBBnPW bit**

Registers that are not reset by the IEBBnPW bit	Remark
IEBBnPSR	Not reset
IEBBnUAR	Not reset
IEBBnSAR	Not reset
IEBBnCDR	Data written from the CPU is not reset but data received during communication is.
IEBBnTCD	Not reset
IEBBnDLR	Data written from the CPU is not reset but data received during communication is.
IEBBnTDL	Not reset
IEBBnCKS	Not reset
IEBBnTMS	Not reset
IEBBnPCR	Not reset
IEBBnSTC0	Not reset
IEBBnSTC1	Not reset
IEBBnDR	Data written from the CPU is not reset but data received during communication is.

**(b) Master request flag (IEBBnMSRQ): Bit 6**

- Set/clear condition

Set: By software

Clear:

- Single mode:

The flag is cleared (to 0) by hardware when master communication is started and when the start interrupt of the master occurs.

The flag is cleared (to 0) by hardware when a communication error interrupt occurs (when the EBBnISR.IEBBnIEBE bit = 1).

The flag is cleared (to 0) by hardware when arbitration loss occurs.

The flag is cleared (to 0) when the IEBBnPW bit is cleared.

- FIFO mode:

The flag is cleared (to 0) by hardware after master communication starts, communication is performed without arbitration loss occurring, and the parity bit of the slave address field output by the unit is transmitted.

The flag is cleared (to 0) by hardware when a communication error interrupt occurs (when the EBBnISR.IEBBnIEBE bit = 1).

The flag is cleared (to 0) by hardware if arbitration losses consecutively occur the number of times specified by the IEBBnTMS.IEBBnALC2 to IEBBnALC0 bits.

The flag is cleared (to 0) when the IEBBnPW bit is cleared.

When the IEBBnMSRQ bit is set (to 1), the IEBus controller starts communication on IEBus as the master.

If communication is in progress on IEBus (if the start bit cannot be detected while the start bit is being communicated or if communication is in progress after the start bit has been detected), however, the controller waits until the current frame ends (holds the master request pending), outputs the start bit after the frame has ended, and starts communication as the master.

- 
- Cautions**
1. Only set the IEBBnMSRQ bit after clearing the IEBBnSTXE bit to 0. After setting the IEBBnSTXE bit to 1, if arbitration loss occurs and the slave is selected, the transmission data prepared for the master might be used as slave transmission data.
  2. Reissue master requests in the single mode and FIFO mode as described below.
    - Single mode:  
When arbitration is lost, use software to reissue master requests.
    - FIFO mode:  
When arbitration is lost, use hardware to reissue master requests.  
However, if arbitration losses consecutively occur the number of times specified by the IEBBnTMS.IEBBnALC2 to IEBBnALC0 bits, the request must be reissued by using software.
-

**(c) Broadcast request flag (IEBBnALRQ): Bit 5**

- Set/clear condition

Set: By software

Clear: By software

- Cautions**
1. The IEBBnMSRQ bit is cleared (to 0) by hardware, but the IEBBnALRQ bit is not. Therefore, if the next master request is for individual communication, clear the IEBBnALRQ bit (to 0).
  2. Be sure to change the value of the IEBBnALRQ bit before setting the IEBBnMSRQ bit (to 1).

**(d) Slave transmission enable flag (IEBBnSTXE): Bit 4**

- Set/clear condition

Set: By software

Clear: By software

Slave transmission is controlled by the value of the slave transmission enable flag, but whether IEBBn performs slave transmission (whether there is an  $\overline{\text{ACK}}$  signal response for the control field) is determined by other conditions.

The  $\overline{\text{ACK}}$  signal response conditions for the control field are shown below.

**Table 22-10 Control field  $\overline{\text{ACK}}$  signal response conditions (when the received control data is 0H, 3H, 4H, 5H, 6H, or 7H)**

Communication target (IEBBnUSR. IEBBnSRQF bit) Slave specification = 1 No specification = 0	Lock status (IEBBnUSR. IEBBnLCKF bit) Lock = 1 No lock = 0	Master unit judgment (IEBBnPAR register match) Lock request unit = 1 Other = 0	Slave transmission enabled (IEBBnBCR.I EBBnSTXE bit)	Slave reception enabled (IEBBnBCR. IEBBnSRXE bit)	Received Control Data					
					0H	3H	4H	5H	6H	7H
1	0	don't care	0	don't care	A	N	N	N	A	N
			1		A	A	N	N	A	A
	1	0	don't care		A	N	A	A	N	N
		1	0		A	N	A	A	A	N
			1	A	A	A	A	A	A	
Other than the above					N					

**Note** A: Slave transmission is performed. (The  $\overline{\text{ACK}}$  signal is returned.)

N: Slave transmission is not performed. (The NACK signal is returned.)

Slave transmission is not performed if the received control data is AH, BH, EH, or FH.

**Table 22-11** Control field  $\overline{\text{ACK}}$  signal response conditions (when the received control data is AH, BH, EH, or FH)

Communication target (IEBBnUSR. IEBBnSRQF bit) Slave specification = 1 No specification = 0	Lock status (IEBBnUSR. IEBBnLCKF bit) Lock = 1 No lock = 0	Master unit judgment (IEBBnPAR register match) Lock request unit = 1 Other = 0	Slave transmission enabled (IEBBnBCR.IE BBnSTXE bit)	Slave reception enabled (IEBBnBCR.IE BBnSRXE bit)	Received control data			
					AH	BH	EH	FH
1	0	don't care	don't care	1	A			
	1	1						
Other than the above					N			

**Note** A: The  $\overline{\text{ACK}}$  signal is returned.

N: The NACK signal is returned.

- Cautions**
1. Set the IEBBnSTXE bit before the control field parity bit is received.
  2. When there is a master request, clear the IEBBnSTXE bit (to 0) before setting the IEBBnMSRQ bit (to 1). This is to avoid transmission of the data of the IEBBnDR register that tries master transmission if the controller loses arbitration after master operation and if slave transmission is requested by the master.

---

**(e) Slave reception enable flag (IEBBnSRXE): Bit 3**

- Set/clear condition
  - Set: By software
  - Clear: By software

When the IEBBnSRXE bit = 1 and the reception control data for the communicated control field addressed to the unit is AH, BH, EH, or FH (or when the lock status is specified and the master unit address of the communication matches the address for which a lock was requested), the ACK signal is returned for the control field, and a slave reception operation is performed.

When the IEBBnSRXE bit = 0 and the reception control data for the communicated control field addressed to the unit is AH, BH, EH, or FH, the NACK signal is returned for the control field, and no slave reception operation is performed.

- 
- Cautions**
1. Set the IEBBnSRXE bit before the control field parity bit is received.
  2. The IEBBnSRXE bit is used to enable or disable slave reception for both individual and broadcast communication. For individual communication, a NACK signal can be returned for the control field to end communication by clearing the IEBBnSRXE bit to 0 (thereby prohibiting slave reception), but, for broadcast communication, although communication cannot be ended by clearing this bit because no ACK/NACK signal is transmitted, no data interrupt occurs because IEBBn does not respond to the broadcast communication.
-

**(2) IEBBnPSR - IEBBn power save register**

The IEBBnPSR register is used to operate and stop the IEBBn operation clock and to control the communication mode.

**Access** This register can be read or written in 8-bit units.

**Address** <IEBBn\_base> + 0004H

**Initial value** 00H

- Cautions**
- The IEBBnPSR register can only be set up when the IEBBnBCR.IEBBnPW bit = 0. Do not set up the register when this bit = 1. If an attempt is made to set up the register when the IEBBnPW bit = 1, the value is ignored.
  - To use IEBBn, first set the IEBBnCLKE bit (to 1) and enable the operation clock.
 

To start the bus operation, specify the settings below.

    - When communication has started
      - Set up the IEBBnCKS register.
      - Set the IEBBnCLKE bit (to 1). (The operation clock operates.)
 

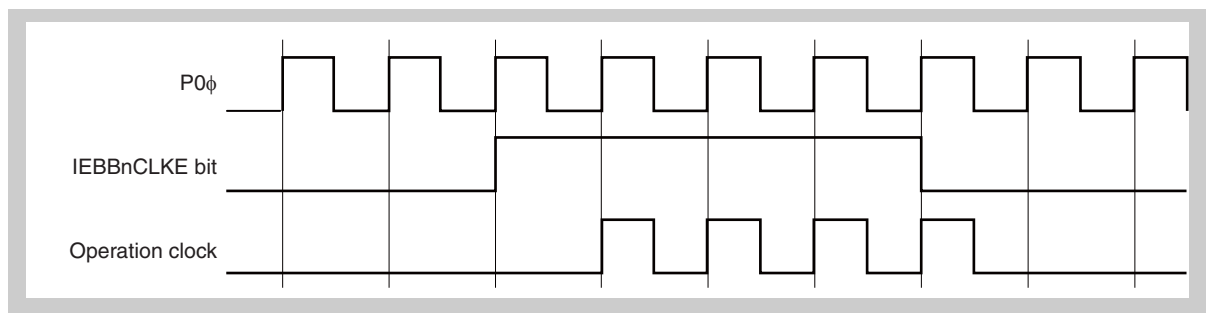
Set the IEBBnCMD bit to 0 or 1 to specify the communication mode.
      - Set up registers such as IEBBnUAR, IEBBnSAR, IEBn0TCD, IEBBnTDL, and IEBBnDR depending on the type of communication.
      - Set the IEBBnBCR.IEBBnPW bit (to 1) to start communication.
    - When communication is stopped
      - Clear the IEBBnPW bit to 0.
      - Clear the IEBBnCLKE bit (to 0). (The operation clock stops.)

7	6	5	4	3	2	1	0
IEBBn CLKE	IEBBn CMD	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R

**Table 22-12 IEBBnPSR register contents**

Bit position	Bit name	Function
7	IEBBnCLKE	<p>Operation clock enable flag</p> <p>0: Stop the operation clock. (This makes it possible to reduce the power consumed by IEBBn.) Initialize the prescaler and baud rate generator.</p> <p>1: Enable the operation clock.</p> <p>The operation clock starts operating one clock cycle after the IEBBnCLKE bit is set (to 1). (For details, see <i>Figure 22-2 “Starting and stopping the operation clock”</i>.) Similarly, the operation clock stops operating one clock cycle after the IEBBnCLKE bit is cleared (to 0). (For details, see <i>Figure 22-2 “Starting and stopping the operation clock”</i>.)</p>
6	IEBBnCMD	<p>IEBBn communication mode setting flag</p> <p>0: Specify communication mode 1. 1: Specify communication mode 2.</p>





**Figure 22-2** Starting and stopping the operation clock

**(3) IEBBnUAR - IEBBn unit address register**

The IEBBnUAR register is used to specify the unit address of the IEBus unit. This register must always be set before starting communication.

Specify the unit address (12 bits) for bits 11 to 0.

**Access** This register can be read or written in 16-bit units.

**Address** <IEBBn\_base> + 0008H

**Initial value** 0000H

- Cautions**
1. The IEBBnUAR register can only be set up when the IEBBnBCR.IEBBnPW bit = 0. Do not set up the register when this bit = 1. If an attempt is made to set up the register when the IEBBnPW bit = 1, the value is ignored.
  2. Writing to this register in 8-bit units is prohibited.

15	14	13	12	11	10	9	8
0	0	0	0				
R	R	R	R	R/W	R/W	R/W	R/W
7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

**(4) IEBBnSAR - IEBBn slave address register**

The IEBBnSAR register is used to specify the address of the communication-partner slave unit during master communication.

During a master request, the value of this register is transmitted as the slave address field data.

Specify the slave address (12 bits) for bits 11 to 0.

**Access** This register can be read or written in 16-bit units.

**Address** <IEBBn\_base> + 000CH

**Initial value** 0000H

**Cautions** 1. When the IEBBnSAR register is overwritten during communication (while the IEBBnBCR.IEBBnPW bit = 1), communication might not be correctly performed. Therefore, overwriting is prohibited from when a master request is issued until the communication or frame completion timing.

Note that overwriting is enabled at the following times:

- When the IEBBnPW bit = 0
- From when the IEBBnPW bit is set to 1 until the first master request (when the IEBBnMSRQ bit = 1)
- From the communication or frame completion timing (assuming the IEBBnPW bit = 1 and the IEBBnMSRQ bit = 0) until the next master request (when the IEBBnMSRQ bit = 1)

2. Writing to this register in 8-bit units is prohibited.

15	14	13	12	11	10	9	8
0	0	0	0				
R	R	R	R	R/W	R/W	R/W	R/W
7	6	5	4	3	2	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

**(5) IEBBnPAR - IEBBn partner address register**

The IEBBnPAR register is used to store the reception master address in the master address field.

When operating the unit is enabled (when the IEBBnBCR.IEBBnPW bit = 1), the received master address is stored using the master address field regardless of whether the unit master is operating or the slave is operating.

- Storage in the single mode

Upon completion of the parity period for the master address field, this is only performed if the parity value is normal and the unit is in the non-lock status.

- Storage in the FIFO mode

If reading the data received during the previous communication has finished (if the IEBBnBSR.IEBBnRFLF bit = 0 and the IEBBnBSR.IEBBnSRFP4 to IEBBnSRFP0 bits = 00000), upon completion of the parity period for the master address field, storage is only performed if the parity value is normal and the unit is in the non-lock status. If reading the received data has not finished, the IEBBnPAR register is not updated until it finishes.

When there is a unit lock, because the address of the unit that requested the lock (the lock master) is retained, the IEBBnPAR register is not updated.

- Lock address transmission request reception in the single mode

If a lock address transmission request is received from the master as a status transmission request, when the received control data receives the lock address (higher four bit) read request (5H), the value of the IEBBnPAR register is read by using software, and then the data in bits 15 to 8 of the IEBBnPAR register is written to the IEBBnDR register.

In addition, if a lock address (lower 8 bits) read request (4H) is received, the value of the IEBBnPAR register is read by using software, and then the data in bits 7 to 0 of the IEBBnPAR register is written to the IEBBnDR register.

- Lock address transmission request reception in the FIFO mode

If a lock address transmission request is received from the master as a status transmission request, the data in the IEBBnPAR register is automatically transmitted to the data field by using hardware.

Specify the partner address (12 bits) for bits 11 to 0.

**Access** This register is read-only, in 16-bit units.

**Address** <IEBBn\_base> + 0010H

**Initial value** 0000H

This register is reset when the IEBBnBCR.IEBBnPW bit is overwritten.

15	14	13	12	11	10	9	8
0	0	0	0				
R	R	R	R	R	R	R	R
7	6	5	4	3	2	1	0
R	R	R	R	R	R	R	R

**(6) IEBBnRSA - IEBBn reception slave address register**

The IEBBnRSA register is used to store the slave address value received using the slave address field.

When operating the unit is enabled (when the IEBBnBCR.IEBBnPW bit = 1), the received slave address is stored using the slave address field regardless of whether the unit master is operating or the slave is operating.

- Storage in the single mode

This is performed upon the completion of the slave address field parity period if the parity value is normal.

- Storage in the FIFO mode

If reading the data received during the previous communication has finished (if the IEBBnBSR.IEBBnRFLF bit = 0 and the IEBBnBSR.IEBBnSRFP4 to IEBBnSRFP0 bits = 00000), upon completion of the parity period for the slave address field, storage is only performed if the parity value is normal. Until reading the received data finishes, the IEBBnRSA register is not updated.

Specify the slave address (12 bits) for bits 11 to 0.

**Access** This register is read-only, in 16-bit units.

**Address** <IEBBn\_base> + 0014H

**Initial value** 0000H

This register is reset when the IEBBnBCR.IEBBnPW bit is overwritten.

15	14	13	12	11	10	9	8
0	0	0	0				
R	R	R	R	R	R	R	R
7	6	5	4	3	2	1	0
R	R	R	R	R	R	R	R

**(7) IEBBnCDR - IEBBn control data register**

The IEBBnCDR register is used to specify the control data transmitted using the control field.

After writing to the IEBBnCDR register, the IEBBnTCD register is written to.

After reading the IEBBnCDR register, the IEBBnRCD register value is read.

**Access** This register can be read or written in 8-bit units.

**Address** <IEBBn\_base> + 0018H

**Initial value** 00H

The read value is reset when the IEBBnBCR.IEBBnPW bit is overwritten.

---

**Caution** When issuing a master request, be sure to set up the IEBBnCDR register before starting communication (when the IEBBnBCR.IEBBnMSRQ bit = 0).

---

**Note** The IEBBnCDR register consists of a write register and a read register. Therefore, data written to this register cannot be read as is. The data received during IEBus communication can be read.

7	6	5	4	3	2	1	0
0	0	0	0	IEBBn SLCD3	IEBBn SLCD2	IEBBn SLCD1	IEBBn SLCD0
R	R	R	R	R/W	R/W	R/W	R/W

**(8) IEBBnTCD - IEBBn transmission control data register**

The IEBBnTCD register is used to specify the control data transmitted using the control field.

The value of the lower 4 bits of the value written to the IEBBnTCD register is transmitted as control data by using the control field during master transmission.

**Access** This register can be read or written in 8-bit units.

**Address** <IEBBn\_base> + 001CH

**Initial value** 00H

- Cautions**
1. When issuing a master request, be sure to set up the IEBBnTCD register before starting communication (when the IEBBnBCR.IEBBnMSRQ bit = 0).
  2. Do not specify undefined values.
  3. During broadcast transmission, specifying slave transmission control data is prohibited.

7	6	5	4	3	2	1	0
0	0	0	0	IEBBn SLTD3	IEBBn SLTD2	IEBBn SLTD1	IEBBn SLTD0
R	R	R	R	R/W	R/W	R/W	R/W

**Table 22-13 IEBBnTCD register contents**

Bit position	Bit name	Function				
3 to 0	IEBBnSLTD [3-0]	Specify the control data transmitted by using the control field.				
		IEBBn SLTD3	IEBBn SLTD2	IEBBn SLTD1	IEBBn SLTD0	Function
		0	0	0	0	Read slave status
		0	0	0	1	Undefined
		0	0	1	0	Undefined
		0	0	1	1	Data reading and locking
		0	1	0	0	Lock address reading (lower 8 bits)
		0	1	0	1	Lock address reading (higher 4 bits)
		0	1	1	0	Slave status reading and unlocking
		0	1	1	1	Read data
		1	0	0	0	Undefined
		1	0	0	1	Undefined
		1	0	1	0	Command writing and locking
		1	0	1	1	Data writing and locking
		1	1	0	0	Undefined
		1	1	0	1	Undefined
		1	1	1	0	Write command
1	1	1	1	Write data		

**(9) IEBBnRCD - IEBBn reception control data register**

The IEBBnRCD register is used to store the control data received using the control field.

The data received by using the control field is read to the lower 4 bits of the IEBBnRCD register. Data is stored in the IEBBnRCD register upon completion of the control field parity period if the parity value is normal.

- Storage in the single mode

When a status transmission request is received, the user performs each process (settings for the transmission data of the IEBBnSSR register or the IEBBnPAR register) according to the value of the lower 4 bits of the IEBBnRCD register read value.

- Storage in the FIFO mode

When a status transmission request is received, the hardware automatically performs the status transmission processing (settings for the transmission data of the IEBBnSSR register or the IEBBnPAR register).

Because it is necessary to judge whether the received data is a command or data, be sure to read the value of this register upon the completion of communication.

In the FIFO mode, if reading the data received during the previous communication has finished (if the IEBBnBSR.IEBBnRFLF bit = 0 and the IEBBnBSR.IEBBnSRFP4-SRFP0 bit = 00000), upon completion of the parity period for the control data field, storage is performed if the parity value is normal. Until reading the received data finishes, the IEBBnRCD register is not updated.

**Access** This register is read-only, in 8-bit units.

**Address** <IEBBn\_base> + 0020H

**Initial value** 00H

This register is reset when the IEBBnBCR.IEBBnPW bit is overwritten.

7	6	5	4	3	2	1	0
0	0	0	0	IEBBn SLRD3	IEBBn SLRD2	IEBBn SLRD1	IEBBn SLRD0
R	R	R	R	R	R	R	R



Table 22-14 IEBBnRCD register contents

Bit position	Bit name	Function				
3 to 0	IEBBnSLRD[3-0]	Specify the control data received by using the control field.				
		IEBBnSLRD3	IEBBnSLRD2	IEBBnSLRD1	IEBBnSLRD0	Function
		0	0	0	0	Read slave status
		0	0	0	1	Undefined
		0	0	1	0	Undefined
		0	0	1	1	Data reading and locking
		0	1	0	0	Lock address reading (lower 8 bits)
		0	1	0	1	Lock address reading (higher 4 bits)
		0	1	1	0	Slave status reading and unlocking
		0	1	1	1	Read data
		1	0	0	0	Undefined
		1	0	0	1	Undefined
		1	0	1	0	Command writing and locking
		1	0	1	1	Data writing and locking
		1	1	0	0	Undefined
		1	1	0	1	Undefined
		1	1	1	0	Write command
1	1	1	1	Write data		

**(10) IEBBnDLR - IEBBn message length register**

The IEBBnDLR register is used to specify the message length data transmitted using the message length field.

After writing to the IEBBnDLR register, the IEBBnTDL register is written to.

After reading the IEBBnDLR register, the IEBBnRDL register value is read.

**Access** This register can be read or written in 8-bit units.

**Address** <IEBBn\_base> + 0024H

**Initial value** 01H

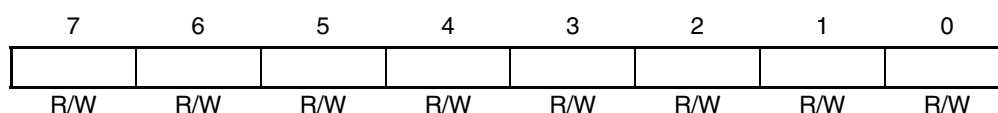
The read value is reset when the IEBBnBCR.IEBBnPW bit is overwritten.

---

**Caution** When issuing a master request, be sure to set up the IEBBnDLR register before starting communication (when the IEBBnBCR.IEBBnMSRQ bit = 0).

---

**Note** The IEBBnDLR register consists of a write register and a read register. Therefore, data written to this register cannot be read as is. The data received during IEBus communication can be read.



**(11) IEbBnTDL - IEbBn transmission message length register**

The IEbBnTDL register is used to specify the message length data transmitted using the message length field.

The value written to the IEbBnTDL register is transmitted as message length data by using the message length field if the unit is the transmission unit (master transmission, slave transmission).

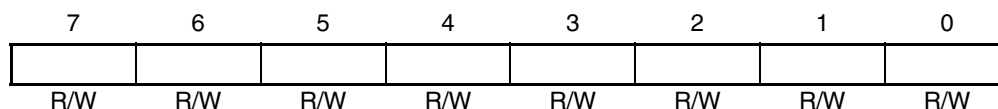
However, when a status transmission request is received, 0H is transmitted as the message length data regardless of the IEbBnTDL register setting.

**Access** This register can be read or written in 8-bit units.

**Address** <IEbBn\_base> + 0028H

**Initial value** 01H

- Cautions**
1. Be sure to set up the IEbBnTDL register before starting communication (when the IEbBnBCR.IEbBnMSRQ bit = 0).
  2. The maximum number of bytes that can be transferred per frame is determined according to the communication mode. For example, when transferring 48 bytes in communication mode 1, perform communication by dividing the data among multiple frames. In this case, when performing the second communication, use the IEbBnSCR register to check the number of data bytes transmitted during the first communication, subtract the number of bytes that were successfully transmitted from the number of bytes you want to transmit, and then specify the result for the IEbBnTDL register. Write the next data to the IEbBnDR register at the same time, and then issue a master request.



**Table 22-15 IEbBnTDL register contents**

Bit								Setting	Number of remaining communication data bytes
7	6	5	4	3	2	1	0		
0	0	0	0	0	0	0	1	01H	1 byte
0	0	0	0	0	0	1	0	02H	2 bytes
...	...	...	...	...	...	...	...	...	...
0	0	0	1	0	1	0	0	20H	32 bytes
...	...	...	...	...	...	...	...	...	...
1	1	1	1	1	1	1	1	FFH	255 bytes
0	0	0	0	0	0	0	0	00H	256 bytes

**(12) IEBBnRDL - IEBBn reception message length register**

The IEBBnRDL register is used to specify the message length data received using the message length field.

The IEBBnRDL register read value is the data received using the message length field.

- Storage in the single mode

Storage proceeds if the parity value is normal at the end of the parity period for the message length field.

- Storage in the FIFO mode

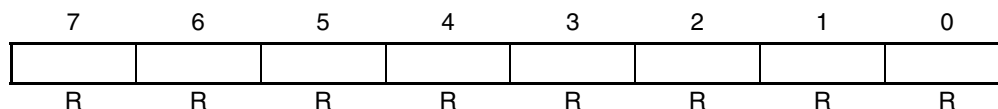
If reading the data received during the previous communication has finished (IEBBnBSR.IEBBnRFLF bit = 0 and the IEBBnBSR.IEBBnSRFP4-SRFP0 bit = 00000), upon completion of the parity period for the message length field, storage is performed if the parity value is normal. Until reading the received data finishes, the IEBBnRDL register is not updated.

**Access** This register is read-only, in 8-bit units.

**Address** <IEBBn\_base> + 002CH

**Initial value** 01H

This register is reset when the IEBBnBCR.IEBBnPW bit is overwritten.



**Table 22-16 IEBBnRDL register contents**

Bit								Setting	Number of remaining communication data bytes
7	6	5	4	3	2	1	0		
0	0	0	0	0	0	0	1	01H	1 byte
0	0	0	0	0	0	1	0	02H	2 bytes
...	...	...	...	...	...	...	...	...	...
0	0	0	1	0	1	0	0	20H	32 bytes
...	...	...	...	...	...	...	...	...	...
1	1	1	1	1	1	1	1	FFH	255 bytes
0	0	0	0	0	0	0	0	00H	256 bytes

**(13) IEBBnCKS - IEBBn clock selection register**

The IEBBnCKS register is used to control the clock selection of the IEBus controller.

**Access** This register can be read or written in 8-bit units.

**Address** <IEBBn\_base> + 0030H

**Initial value** 17H

**Caution** The IEBBnCKS register can only be set up when the IEBBnBCR.IEBBnPW bit = 0 and the IEBBnPSR.IEBBnCLKE bit = 0. Do not set up the register when this bit = 1. If an attempt is made to set up the register when the IEBBnPW bit = 1, the value is ignored.

7	6	5	4	3	2	1	0
0	0	0	IEBBn PRS	0	IEBBn BRS2	IEBBn BRS1	IEBBn BRS0
R	R	R	R/W	R	R/W	R/W	R/W

**Table 22-17 IEBBnCKS register contents**

Bit position	Bit name	Function																																				
4	IEBBnPRS	Specify the prescaler output (PRSOUT). 0: P0 $\phi$ 1: P0 $\phi$ 2  <b>Caution</b> The conditions under which the prescaler is initialized are as follows: <ul style="list-style-type: none"> <li>When the IEBBnPRS bit is overwritten</li> <li>When the IEBBnBCR.IEBBnPW bit = 0 and the IEBBnPSR.IEBBnCLKE bit = 1</li> </ul>																																				
2 to 0	IEBBnBRS [2-0]	Specify the operation clock output (MCK). <table border="1"> <thead> <tr> <th>IEBBnBRS2</th> <th>IEBBnBRS1</th> <th>IEBBnBRS0</th> <th>Operation clock output (MCK)</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>PRSOUT/1</td> </tr> <tr> <td>0</td> <td>0</td> <td>1</td> <td>PRSOUT/1</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> <td>PRSOUT/2</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> <td>PRSOUT/3</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> <td>PRSOUT/4</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> <td>PRSOUT/5</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> <td>PRSOUT/6</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> <td>PRSOUT/7</td> </tr> </tbody> </table> <b>Caution</b> The conditions under which the baud rate generator is initialized are as follows: <ul style="list-style-type: none"> <li>When the IEBBnBRS2 to IEBBnBRS0 bits are overwritten</li> <li>When the IEBBnBCR.IEBBnPW bit = 0 and the IEBBnPSR.IEBBnCLKE bit = 1</li> </ul>	IEBBnBRS2	IEBBnBRS1	IEBBnBRS0	Operation clock output (MCK)	0	0	0	PRSOUT/1	0	0	1	PRSOUT/1	0	1	0	PRSOUT/2	0	1	1	PRSOUT/3	1	0	0	PRSOUT/4	1	0	1	PRSOUT/5	1	1	0	PRSOUT/6	1	1	1	PRSOUT/7
IEBBnBRS2	IEBBnBRS1	IEBBnBRS0	Operation clock output (MCK)																																			
0	0	0	PRSOUT/1																																			
0	0	1	PRSOUT/1																																			
0	1	0	PRSOUT/2																																			
0	1	1	PRSOUT/3																																			
1	0	0	PRSOUT/4																																			
1	0	1	PRSOUT/5																																			
1	1	0	PRSOUT/6																																			
1	1	1	PRSOUT/7																																			

Table 22-18 Input clock specification example

P0 $\phi$	IEBBnPRS	IEBBnBRS2	IEBBnBRS1	IEBBnBRS0	Specified value
32 MHz	0	1	0	0	04H

---

**Caution** The IEBus controller is designed to operate at 8 MHz.

---

**(14) IEBBnTMS - IEBBn transfer mode setting register**

The IEBBnTMS register is used to control the IEBus controller communication operations.

**Access** This register can be read or written in 8-bit units.

**Address** <IEBBn\_base> + 0034H

**Initial value** 01H

**Caution** The IEBBnTMS register can only be set up when the IEBBnBCR.IEBBnPW bit = 0. Do not set up the register when this bit = 1. If an attempt is made to set up the register when the IEBBnPW bit = 1, the value is ignored.

7	6	5	4	3	2	1	0
IEBBn FMDE	IEBBn SLRI1	IEBBn SLRIO	IEBBn SLTI1	IEBBn SLTI0	IEBBn ALC2	IEBBn ALC1	IEBBn ALC0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

**Table 22-19 IEBBnTMS register contents (1/2)**

Bit position	Bit name	Function															
7	IEBBnFMDE	Specify whether to enable or disable FIFO mode operation. 0: Disable FIFO mode operation (single mode). 1: Enable FIFO mode operation.															
6, 5	IEBBnSLRI [1, 0]	Specify the IEBBTV occurrence timing during FIFO mode reception. <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>IEBBnSLRI1</th> <th>IEBBnSLRIO</th> <th>IEBBTV occurrence timing</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td>When the reception data that has not been read from the reception FIFO buffer reaches 32 bytes</td> </tr> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">1</td> <td>When the reception data that has not been read from the reception FIFO buffer reaches 24 bytes</td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td>When the reception data that has not been read from the reception FIFO buffer reaches 16 bytes</td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td>When the reception data that has not been read from the reception FIFO buffer reaches 8 bytes</td> </tr> </tbody> </table>	IEBBnSLRI1	IEBBnSLRIO	IEBBTV occurrence timing	0	0	When the reception data that has not been read from the reception FIFO buffer reaches 32 bytes	0	1	When the reception data that has not been read from the reception FIFO buffer reaches 24 bytes	1	0	When the reception data that has not been read from the reception FIFO buffer reaches 16 bytes	1	1	When the reception data that has not been read from the reception FIFO buffer reaches 8 bytes
IEBBnSLRI1	IEBBnSLRIO	IEBBTV occurrence timing															
0	0	When the reception data that has not been read from the reception FIFO buffer reaches 32 bytes															
0	1	When the reception data that has not been read from the reception FIFO buffer reaches 24 bytes															
1	0	When the reception data that has not been read from the reception FIFO buffer reaches 16 bytes															
1	1	When the reception data that has not been read from the reception FIFO buffer reaches 8 bytes															
4, 3	IEBBnSLTI [1, 0]	Specify the IEBBTD occurrence timing during FIFO mode transmission. <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>IEBBnSLTI1</th> <th>IEBBnSLTI0</th> <th>IEBBTD occurrence timing</th> </tr> </thead> <tbody> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">0</td> <td>When the transmission FIFO buffer becomes empty</td> </tr> <tr> <td style="text-align: center;">0</td> <td style="text-align: center;">1</td> <td>When the untransmitted data remaining in the transmission FIFO buffer reaches 2 bytes</td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">0</td> <td>When the untransmitted data remaining in the transmission FIFO buffer reaches 4 bytes</td> </tr> <tr> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> <td>When the untransmitted data remaining in the transmission FIFO buffer reaches 8 bytes</td> </tr> </tbody> </table>	IEBBnSLTI1	IEBBnSLTI0	IEBBTD occurrence timing	0	0	When the transmission FIFO buffer becomes empty	0	1	When the untransmitted data remaining in the transmission FIFO buffer reaches 2 bytes	1	0	When the untransmitted data remaining in the transmission FIFO buffer reaches 4 bytes	1	1	When the untransmitted data remaining in the transmission FIFO buffer reaches 8 bytes
IEBBnSLTI1	IEBBnSLTI0	IEBBTD occurrence timing															
0	0	When the transmission FIFO buffer becomes empty															
0	1	When the untransmitted data remaining in the transmission FIFO buffer reaches 2 bytes															
1	0	When the untransmitted data remaining in the transmission FIFO buffer reaches 4 bytes															
1	1	When the untransmitted data remaining in the transmission FIFO buffer reaches 8 bytes															

Table 22-19 IEBBnTMS register contents (2/2)

Bit position	Bit name	Function																
2 to 0	IEBBnALC [2-0]	<p>Specify the arbitration loss number. This is only valid in the FIFO mode. The settings of these bits are invalid in the single mode. When arbitration is lost and the counter is set to 0H, the IEBBnBCR.IEBBnMSRQ bit is not retained. The settings of the IEBBnALC2 to IEBBnALC0 bits are always retained. The settings are not changed each time arbitration is lost. The arbitration loss counter is decremented separately from the IEBBnALC2 to IEBBnALC0 bits. Overwriting the IEBBnALC2 to IEBBnALC0 bits while the arbitration loss counter is counting does not affect the counter. (The values of the bits are specified for the arbitration loss counter when the IEBBnMSRQ bit = 1.)</p> <table border="1"> <thead> <tr> <th>IEBBnALC2</th> <th>IEBBnALC1</th> <th>IEBBnALC0</th> <th>Maximum arbitration loss count</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>1</td> <td>1</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> <td>3</td> </tr> <tr> <td colspan="3">Other than the above</td> <td>Setting prohibited</td> </tr> </tbody> </table>	IEBBnALC2	IEBBnALC1	IEBBnALC0	Maximum arbitration loss count	0	0	1	1	0	1	1	3	Other than the above			Setting prohibited
IEBBnALC2	IEBBnALC1	IEBBnALC0	Maximum arbitration loss count															
0	0	1	1															
0	1	1	3															
Other than the above			Setting prohibited															



**(a) FIFO operation enable bit (IEBBnFMDE): Bit 7**

Differences between operation in the single mode and FIFO mode are shown below.

**Table 22-20 Differences between operation in the single mode and FIFO mode**

Transfer mode	Arbitration loss master request flag (IEBBnBCR.IEBBnMSRQ)	Arbitration loss error interrupt signal (IEBBTERR)	Support for slave status requests and lock address requests
Single mode	Clear	<ul style="list-style-type: none"> <li>The signal is not output.</li> <li>The IEBBnESR.IEBBnABTE bit is fixed to 0.</li> </ul>	During the slave status interrupt servicing, the value of the IEBBnSSR or IEBBnPAR register is written to the IEBBnDR register.
FIFO mode	The flag value is retained until arbitration is lost the number of times specified for the IEBBnALC2 to IEBBnALC0 bits. (The flag is cleared unless an error occurs during communication between third parties.)	<ul style="list-style-type: none"> <li>The signal is output if arbitration is lost the specified number of times.</li> <li>The IEBBnESR.IEBBnABTE bit is simultaneously set (to 1).</li> </ul>	The value of the IEBBnSSR or IEBBnPAR register is automatically sent by hardware by using the data field.

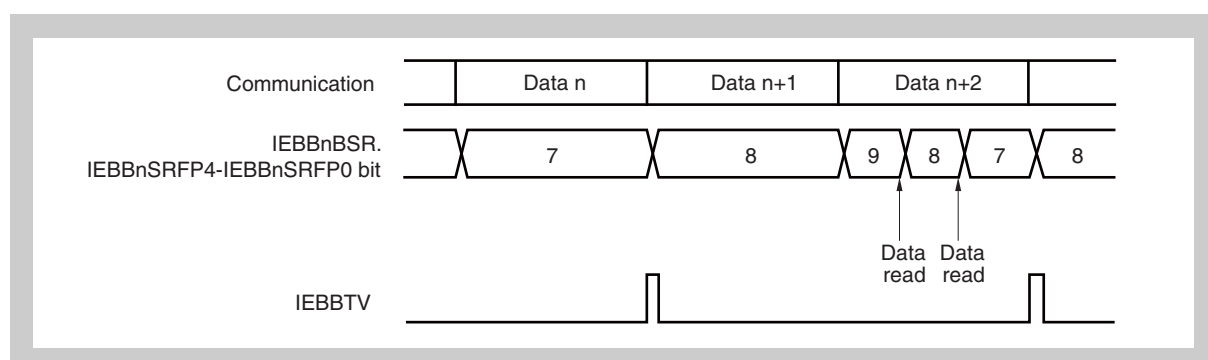
**(b) FIFO mode reception IEBBTV occurrence timing specification bits (IEBBnSLRI1, IEBBnSLRIO): Bits 6, 5**

In communication mode 1, the IEBBnSLRI1 and IEBBnSLRIO bits are cleared to 00.

In communication mode 2, because data that exceeds 32 bytes is received, data must be read during reception. The occurrence timing of the data interrupt (IEBBTV) that requests that received data be read is specified by the IEBBnSLRI1 and IEBBnSLRIO bits.

The IEBBTV interrupt occurs when received data is stored in the FIFO buffer and the number of unread bytes of data reaches the value specified for the IEBBnSLRI1 and IEBBnSLRIO bits.

For the data exceeding the number of bytes specified for the IEBBnSLRI1 and IEBBnSLRIO bits, the data is read, and no interrupt occurs even if the number of unread bytes of received data exceeds the value specified for the IEBBnSLRI1 and IEBBnSLRIO bits.

**Figure 22-3 Example of operation in communication mode 2: When the IEBBnSLRI1 and IEBBnSLRIO bits = 11**

**(c) FIFO mode transmission IEBBTB occurrence timing specification bits (IEBBnSLTI1, IEBBnSLTI0): Bits 4, 3**

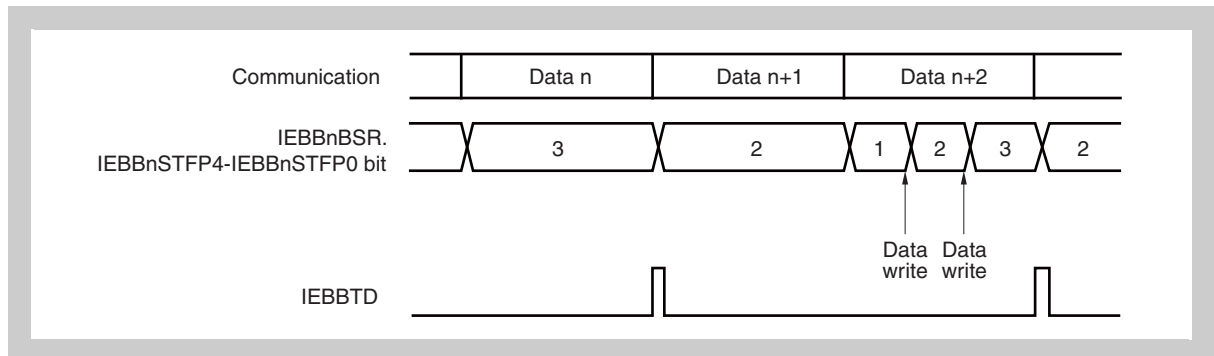
In communication mode 1, clear the IEBBnSLTI1 and IEBBnSLTI0 bits to 00.

In communication mode 2, because data that exceeds 32 bytes is transmitted, transmission data must be written during transmission. The occurrence timing of the data interrupt (IEBBTD) that requests that transmission data be written is specified by the IEBBnSLTI1 and IEBBnSLTI0 bits.

When less than 32 bytes of data are written to the FIFO buffer, IEBBTD occurs for the remaining number of bytes of data to be transmitted.

The IEBBTD interrupt occurs when transmission data is transferred from the FIFO buffer to the shift register and the number of bytes of data that have not been transmitted reaches the value specified for the IEBBnSLTI1 and IEBBnSLTI0 bits.

For data less than the number of bytes specified for the IEBBnSLTI1 and IEBBnSLTI0 bits, the data is written, and no interrupt occurs even if the number of bytes of data that have not been transmitted reaches the value specified for the IEBBnSLTI1 and IEBBnSLTI0 bits.

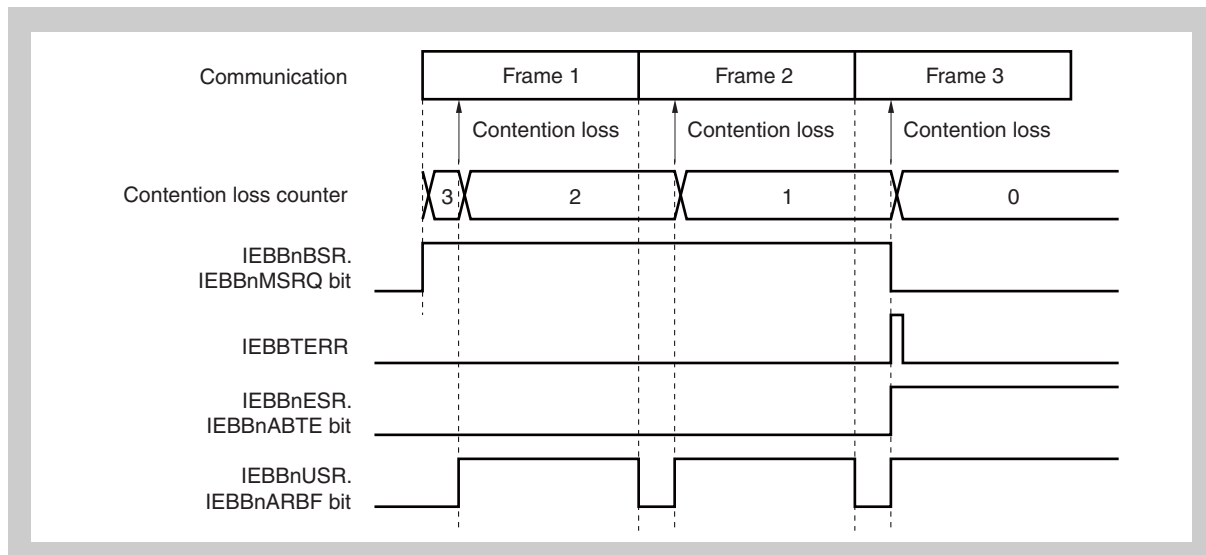


**Figure 22-4 Example of operation in communication mode 2: When the IEBBnSLTI1 and IEBBnSLTI0 bits = 01**

**(d) Arbitration loss count specification bits (IEBBnALC2 to IEBBnALC0):  
Bits 2 to 0**

This is only valid in the FIFO mode. The settings of these bits are invalid in the single mode.

When arbitration is lost and the counter is set to 0H, the IEBBnBCR.IEBBnMSRQ bit is not retained.



**Figure 22-5 Example of operation in the FIFO mode: When the IEBBnALC2 to IEBBnALC0 bits = 011**

**(15) IEBBnPCR - IEBBn pointer clear register**

The IEBBnPCR register is the trigger bit register for clearing the FIFO buffer pointer.

**Access** This register is write-only, in 8-bit units.

**Address** <IEBBn\_base> + 0038H

**Initial value** 00H

7	6	5	4	3	2	1	0
IEBBn CRPT	IEBBn CTPT	0	0	0	0	0	0
W	W	R	R	R	R	R	R

**Table 22-21 IEBBnPCR register contents**

Bit position	Bit name	Function
7	IEBBnCRPT	<p>Clear trigger bit for the store pointer and read pointer of the reception FIFO buffer</p> <p>0: No operation 1: Clear the store pointer and read pointer of the reception FIFO buffer.</p> <p>The bit value can only be changed by setting the bit (to 1). Attempting to clear the bit (to 0) does not change the bit value. When the bit is read, 0 is always returned.</p> <p><b>Caution</b> During reception or before reading received data, if the IEBBnCRPT bit is set to 1, the received data cannot be read. Except when discarding received data, only write 1 to the IEBBnCRPT bit after receiving and then reading data.</p>
6	IEBBnCTPT	<p>Clear the trigger bit for the write pointer and load pointer of the transmission FIFO buffer</p> <p>0: No operation 1: Clear the write pointer and load pointer of the transmission FIFO buffer.</p> <p>The bit value can only be changed by setting the bit (to 1). Attempting to clear the bit (to 0) does not change the bit value. When the bit is read, 0 is always returned.</p> <p><b>Cautions</b></p> <ol style="list-style-type: none"> <li>If the IEBBnCTPT bit = 1 while specifying data for the FIFO buffer or during transmission, the specified data cannot be transmitted. Except when discarding transmission data, only write 1 to the IEBBnCTPT bit after transmitting the data written to the FIFO buffer.</li> <li>To discard transmission data, set the IEBBnCTPT bit to 1 during transmission. Because the data is lost in this case, transmission is not performed and an underrun error occurs.</li> </ol>

**(16) IEBBnBSR - IEBBn buffer status register**

The IEBBnBSR register indicates the FIFO buffer status.

**Access** This register is read only, in 16-bit units.

**Address** <IEBBn\_base> + 003CH

**Initial value** 0000H

This register is reset when 0 is written to the IEBBnBCR.IEBBnPW bit.

15	14	13	12	11	10	9	8
IEBBn RFLF	IEBBn FOVR	0	IEBBn SRFP4	IEBBn SRFP3	IEBBn SRFP2	IEBBn SRFP1	IEBBn SRFP0
R	R	R	R	R	R	R	R
7	6	5	4	3	2	1	0
IEBBn TFLF	IEBBn FOVW	0	IEBBn STFP4	IEBBn STFP3	IEBBn STFP2	IEBBn STFP1	IEBBn STFP0
R	R	R	R	R	R	R	R

**Table 22-22 IEBBnBSR register contents (1/2)**

Bit position	Bit name	Function
15	IEBBnRFLF	Reception FIFO buffer full status flag 0: There are 31 bytes of data or less that have not been read in the reception FIFO buffer. 1: There are 32 bytes of data that have not been read in the reception FIFO buffer.  This flag indicates that there are 32 bytes of data that have not been read in the reception FIFO buffer and that the buffer is full.
14	IEBBnFOVR	This flag indicates whether reception FIFO buffer over-reading has occurred. 0: Reception FIFO buffer over-reading has not occurred. 1: Reception FIFO buffer over-reading has occurred.
12 to 8	IEBBnSRFP[4-0]	This flag indicates the number of data bytes that have not been read remaining in the reception FIFO buffer. The (store pointer - read pointer) value can be read. However, the following values are used when IEBBnSRFP4 to IEBBnSRFP0 = 00000. <ul style="list-style-type: none"> <li>When the IEBBnRFLF bit = 0 and the IEBBnSRFP4 to IEBBnSRFP0 bits = 00000 Number of remaining data bytes that have not been read = 0 bytes</li> <li>When the IEBBnRFLF bit = 1, and the IEBBnSRFP4 to IEBBnSRFP0 bits = 00000 Number of remaining data bytes that have not been read = 32 bytes</li> </ul>
7	IEBBnTFLF	Transmission FIFO buffer full status flag 0: There are 31 bytes of data or less that have not been transferred in the transmission FIFO buffer. 1: There are 32 bytes of data that have not been transferred in the transmission FIFO buffer.  This flag indicates that there are 32 bytes of data that have not been transferred in the transmission FIFO buffer and that the buffer is full.
6	IEBBnFOVW	Transmission FIFO buffer overwrite flag 0: No transmission FIFO buffer overwrite has occurred. 1: A transmission FIFO buffer overwrite has occurred.

Table 22-22 IEBBnBSR register contents (2/2)

Bit position	Bit name	Function
4 to 0	IEBBnSTFP[4-0]	<p>This flag indicates the number of data bytes that have not been transferred remaining in the transmission FIFO buffer. The (write pointer - load pointer) value can be read. However, the following values are used when the IEBBnSTFP4 to STFPSTFP0 bits = 00000.</p> <ul style="list-style-type: none"> <li>When the IEBBnTFLF bit = 0 and the IEBBnSTFP4 to IEBBnSTFP0 bits = 00000 Number of remaining data bytes that have not been transferred = 0 bytes</li> <li>When the IEBBnTFLF bit = 1 and the IEBBnSTFP4 to IEBBnSTFP0 bits = 00000 Number of remaining data bytes that have not been transferred = 32 bytes</li> </ul>

**(a) Reception FIFO buffer over-read indicating flag (IEBBnFOVR): Bit 14**

- Set/clear condition

Set:

- Single mode:

The IEBBnFOVR bit is not set (to 1).

- FIFO mode:

When an over-read (a read of the IEBBnDR register while the store pointer = the read pointer) occurs for the reception FIFO buffer

Clear:

- Single mode or FIFO mode:

By software (The flag is cleared when 1 is written to the IEBBnPCR.IEBBnCRPT bit.)

After reading the reception FIFO buffer, detect whether an over-read occurred by reading the IEBBnFOVR bit. (The data read during the over-read is the last bytes read during multiple operations.)

Even if an over-read is detected, because the data cannot be read again, perform software processing such as requesting retransmission. The IEBBnFOVR bit is cleared by writing 1 to the IEBBnPCR.IEBBnCRPT bit. (If an over-read occurred, the FIFO buffer pointers have already been cleared.)

**(b) Transmission FIFO buffer overwrite flag (IEBBnFOVW): Bit 6**

- Set/clear condition

Set:

- Single mode:

The IEBBnFOVW bit is not set (to 1).

- FIFO mode:

When there are 32 bytes of that have not been transmitted in the transmission FIFO buffer and a 33rd byte of data is written

Clear:

- Single mode or FIFO mode:

By software (The flag is cleared when 1 is written to the IEBBnPCR.IEBBnCTPT bit.)

After writing the required data to the transmission FIFO buffer, read the IEBBnFOVW bit before setting the master request flag. (IEBBnBCR.IEBBnMSRQ) (to 1) to detect whether an overwrite has occurred. If an overwrite is detected, clear the FIFO buffer pointers by writing 1 to the IEBBnPCR.IEBBnCTPT bit and then specify the data again.

**(17) IEBBnSSR - IEBBn slave status register**

The IEBBnSSR register indicates the communication status of the slave unit.

When a slave status transmission request interrupt is received from the master and the received control data is 0H or 6H, the IEBBnSSR register value is automatically written to the IEBBnDR register, and the slave status is transmitted.

In addition, when transmitting the slave status, because 01H is automatically transmitted as the message length, the IEBBnTDL register does not have to be set up.

Because bits 7 and 6 indicate the highest mode supported by the unit, they are fixed to 10 (which indicates communication mode 2).

**Access** This register is read-only, in 8-bit units.

**Address** <IEBBn\_base> + 0040H

**Initial value** 81H

The IEBBnSSLF and IEBBnSTLF bits are reset by writing 0 to the IEBBnBCR.IEBBnPW bit.

The IEBBnSRXF and IEBBnSTXF bits are reset when the value of the IEBBnPW bit is overwritten with a different value.

7	6	5	4	3	2	1	0
1	0	0	IEBBn SSLF	0	IEBBn STLF	IEBBn SRXF	IEBBn STXF
R	R	R	R	R	R	R	R

**Table 22-23 IEBBnSSR register contents (1/2)**

Bit position	Bit name	Function
4	IEBBnSSLF	Slave transmission status flag 0: Slave transmission is stopped. 1: Slave transmission is enabled.
2	IEBBnSTLF	Lock status flag 0: Unlocked 1: Locked
1	IEBBnSRXF	IEBBnDR register or FIFO buffer reception status flag In the single mode 0: Received data has not been stored in the IEBBnDR register. 1: Received data has been stored in the IEBBnDR register. In the FIFO mode 0: Received data has not been stored in the FIFO buffer. 1: Received data has been stored in the FIFO buffer.



Table 22-23 IEBBnSSR register contents (2/2)

Bit position	Bit name	Function
0	IEBBnSTXF	<p>IEBBnDR register or FIFO buffer transmission status flag</p> <ul style="list-style-type: none"> <li>When communication is not being performed           <ol style="list-style-type: none"> <li>The flag is always this value.</li> </ol> </li> <li>Master           <p>In the single mode (transmission)</p> <ol style="list-style-type: none"> <li>The data specified for the IEBBnDR register has been transferred to the transmission shift register, and the next transmission data has not been written to the IEBBnDR register.</li> <li>Transmission data remains in the IEBBnDR register. (This is the status up until the contents of the IEBBnDR register are transferred to the transmission shift register.)</li> </ol> <p>In the single mode (reception)</p> <ol style="list-style-type: none"> <li>The flag is always this value.</li> </ol> <p>In the FIFO mode (transmission)</p> <ol style="list-style-type: none"> <li>The number of bytes of data written to the FIFO buffer have been transferred from the FIFO buffer to the transmission shift register, and the next transfer data has not been written to the FIFO buffer.</li> <li>This is the status from when communication starts until data is transferred to the transmission shift register and the number of bytes of data written to the FIFO buffer disappear.</li> </ol> <p>In the FIFO mode (reception)</p> <ol style="list-style-type: none"> <li>The flag is always this value.</li> </ol> </li> <li>Slave           <p>In the single mode (transmission)</p> <ol style="list-style-type: none"> <li>The data specified for the IEBBnDR register has been transferred to the transmission shift register, and the next transmission data has not been written to the IEBBnDR register.</li> <li>This is the status from when communication starts until the first transmission data is transferred from the IEBBnDR register to the transmission shift register. This is also the status after writing data to the IEBBnDR register until the data is transferred to the transmission shift register.</li> </ol> <p>In the single mode (reception)</p> <ol style="list-style-type: none"> <li>The flag is always this value.</li> </ol> <p>In the FIFO mode (transmission)</p> <ol style="list-style-type: none"> <li>The number of bytes of data written to the FIFO buffer have been transferred from the FIFO buffer to the transmission shift register, and the next transfer data has not been written to the FIFO buffer.</li> <li>This is the status from when communication starts until data is transferred to the transmission shift register and the number of bytes of data written to the FIFO buffer disappear.</li> </ol> <p>In the FIFO mode (reception)</p> <ol style="list-style-type: none"> <li>The flag is always this value.</li> </ol> </li> <li>Third party           <ol style="list-style-type: none"> <li>The flag is always this value.</li> </ol> </li> </ul>

**(a) Slave transmission status flag (IEBBnSSLF): Bit 4**

The value of the slave transmission enable flag (the IEBBnBCR.IEBBnSTXE bit) is applied as is.

**(b) Lock status flag (IEBBnSTLF): Bit 2**

The value of the lock status flag (the IEBBnUSR.IEBBnLCKF bit) is applied as is.

**(c) IEBBnDR register or FIFO buffer reception status flag (IEBBnSRXF): Bit 1**

- Set/clear condition

Set:

- Single mode:

When received data is stored in the IEBBnDR register

- FIFO mode:

When received data is stored in the FIFO buffer

Clear:

- Single mode:

When the IEBBnDR register contents are read

- FIFO mode:

When all the received data stored in the FIFO buffer has been read

In the single mode, when the IEBBnSRXF bit is set (to 1), the data interrupt IEBBTD occurs.

In the single mode, when IEBBTD occurs on data reception, the IEBBnDR register must be read before the next data is received. During broadcast communication, an overrun error occurs if the IEBBnDR register is not read regardless of whether IEBBTD has occurred. For details, see the description of the IEBBnOVRE bit in 22.3.2 (20) "*IEBBnESR - IEBBn error status register*".

In the FIFO mode, even if the IEBBnSRXF bit is set (to 1), the interrupt signal IEBBTV is not generated. For details about the IEBBTV generation timing, see the description of the IEBBnSLRI1, IEBBnSLRI0 bits in 22.3.2 (14) "*IEBBnTMS - IEBBn transfer mode setting register*".

(d) **IEBBnDR register or FIFO buffer transmission status flag (IEBBnSTXF):  
Bit 0**

- Set/clear condition

Set:

- When communication finishes
- Single mode:  
When the IEBBnDR register is written to
- FIFO mode:  
When transmission data is written to the FIFO buffer

Clear:

- Single mode:  
When the contents of the IEBBnDR register are written to the transmission shift register
- FIFO mode:  
When the number of bytes of data written to the FIFO buffer are transferred from the FIFO buffer to the transmission shift register

In the single mode, when the IEBBnSTXF bit is cleared, the data interrupt IEBBTD occurs.

In the single mode, when IEBBTD occurs on data transmission, the next transmission data must be written to the IEBBnDR register.

Regardless of whether IEBBTD occurs, an underrun error occurs if IEBBnDR is not written to. For details, see the description of the IEBBnUNRE bit in 22.3.2 (20) "*IEBBnESR - IEBBn error status register*".

In the FIFO mode, even if the IEBBnSRXF bit is set (to 1), the interrupt signal IEBBTD is not generated. For details about the IEBBTD generation timing, see the description of the IEBBnSLT11, IEBBnSLT10 bits in 22.3.2 (14) "*IEBBnTMS - IEBBn transfer mode setting register*".

**(18) IEBBnUSR - IEBBn unit status register**

The IEBBnUSR register indicates the unit status.

**Access** This register is read-only, in 8-bit units.

**Address** <IEBBn\_base> + 0044H

**Initial value** 00H

This register is reset when the value of the IEBBnPW bit is overwritten with a different value.

7	6	5	4	3	2	1	0
0	IEBBn SRQF	IEBBn ARBF	IEBBn ALTF	IEBBn ACKF	IEBBn LCKF	0	0
R	R	R	R	R	R	R	R

**Table 22-24 IEBBnUSR register contents**

Bit position	Bit name	Function
6	IEBBnSRQF	Slave request flag for the unit 0: There are no slave requests. 1: There is a slave request.
5	IEBBnARBF	Arbitration result flag 0: Arbitration loss did not occur. 1: Arbitration loss occurred.
4	IEBBnALTF	Broadcast communication flag 0: Individual communication status 1: Broadcast communication status
3	IEBBnACKF	Acknowledge transmission flag 0: A NACK signal is transmitted. 1: An $\overline{\text{ACK}}$ signal is transmitted.
2	IEBBnLCKF	Lock status flag 0: Unlocked 1: Locked

**(a) Slave request flag for the unit (IEBBnSRQF): Bit 6**

- Set/clear condition

Set:

When the unit is requested as a slave (if the condition in *Table 22-25 “Slave request conditions (conditions for setting the IEBBnSRQF bit)”* is satisfied), this flag is set (to 1) by hardware when the parity bit communication of the slave address field ends<sup>a</sup>.

Clear:

This flag is cleared (to 0) by hardware when the unit is not requested as a slave (if the condition in *Table 22-25 “Slave request conditions (conditions for setting the IEBBnSRQF bit)”* is not satisfied). The timing<sup>a</sup> is the same as that for setting the flag.

- a) The bit is updated when the communication of the slave address field parity bit finishes without an error such as a parity error occurring. For example, if the slave address reception parity is incorrect, the IEBBnSRQF bit is not updated and the previous value is retained.

**Table 22-25 Slave request conditions (conditions for setting the IEBBnSRQF bit)**

Status of unit	Received master address	Communication mode	Received slave address
Not locked	don't care	Individual	IEBBnUAR match
		Broadcast	Group matching
			FFFH match
Locked	Locked master matching	Individual	IEBBnUAR match
		Broadcast	Group matching
			FFFH match

**Note** IEBBnUAR match: When the reception slave address and unit IEBBnUAR register match

Group match: When the reception-slave-address group address and unit IEBBnUAR-register group address match

FFFH match: When the reception slave address is FFFH

**Table 22-26  $\overline{\text{ACK}}$  signal response condition for the slave address field**

Status of unit	Received master address	Communication mode	Received slave address
don't care	don't care	Individual	IEBBnUAR match

**Note** IEBBnUAR match: When the reception slave address and unit IEBBnUAR register match

**Caution** If a unit other than the locked master communicates with the unit while the unit is locked, the IEBBnSRQF bit is not set but the  $\overline{\text{ACK}}$  signal is returned to the slave address field. This is because communication must be continued, even for communication of a unit other than the locked master, if the control data received using the control field is a slave status transmission request.

**(b) Arbitration result flag (IEBBnARBF): Bit 5**

- Set/clear condition

Set:

When the data output by the unit does not match the received data during the arbitration period<sup>a</sup>

Clear: After each communication frame start bit is transmitted or received

- a) In the FIFO mode, the arbitration loss flag is set (to 1) based on the same condition even while the master request flag (IEBBnBCR.IEBBnMSRQ) is retained.

Note that, if arbitration is lost in the FIFO mode, a NACK reception error is detected by returning the NACK signal after control data reception, and software processing by outputting IEBBTERR is required.

In the case of a status transmission request, the NACK signal is not returned because there is an automatic response.

The IEBBnARBF bit is set (to 1) if there is inter-unit-data contention during the arbitration period (the broadcast field and master address field period) and the unit loses arbitration.

Arbitration loss is judged to have occurred when the unit output data does not match the received data.

Because the IEBus controller uses AND logic, the unit that outputs 0 wins arbitration.

In other words, among units that output broadcast data (0) for the broadcast field, the unit that has the smallest master address wins arbitration.

---

**Caution** In the single mode, when the start interrupt occurs upon issuing a master request, use the IEBBnARBF bit to check for arbitration loss. To transfer data again in the case of arbitration loss, perform software processing to issue another master request.

In the FIFO mode, another master request is issued even if arbitration is lost the number of times specified by the IEBBnTMS.IEBBnALC0 to IEBBnALC2 bits. If the number of arbitration losses exceeds this setting, an interrupt occurs to indicate an arbitration loss error. To transfer data again, perform software processing to issue another master request.

---

**(c) Broadcast communication flag (IEBBnALTF): Bit 4**

Flag indicating whether the unit is performing broadcast communication. The contents of the flag are updated in the broadcast field of each frame.

- Set/clear condition

Set: When “broadcast” is received by the broadcast field

Clear: When individual is received by the broadcast field

---

**Caution** The broadcast flag is updated regardless of whether IEBus is the communication target.

---

In the FIFO mode, storage is performed upon header completion if reading the data received during the previous communication has finished (if the IEBBnBSR.IEBBnRFLF bit = 0 and the nBSR.IEBBnSRFP4 to IEBBnSRFP0 bits = 00000). Until reading the received data finishes, the IEBBnALTF register is not updated.

**(d) Acknowledge transmission flag (IEBBnACKF): Bit 3**

This flag indicates whether the  $\overline{\text{ACK}}$  signal was transmitted during the acknowledge bit period of the acknowledge bit field when IEBus is the receiving unit.

- Set/clear condition

Set:

When  $\overline{\text{ACK}}$  is transmitted upon the completion of the acknowledge bit period for each field

Clear:

When NACK is transmitted upon the completion of the acknowledge bit period for each field

- 
- Cautions**
1. If a communication error occurs and the unit returns to the initial status, no update is performed at the end of the acknowledge bit period for the corresponding field. For example, if the reception parity for the control field is incorrect, because IEBBn changes to the initial status (the communication standby status) after parity reception due to the parity error, a NACK signal is returned by using the control field (more accurately, no  $\overline{\text{ACK}}$  signal is returned), but this is not applied to the IEBBnACKF bit, and the previous value is retained.
  2. In the single mode, because the occurrence timing (when IEBBTSTA or IEBBTV becomes active) for the start interrupt and status transmission interrupt is when parity bit reception ends, the reading of the IEBBnUSR register by the previously described interrupt in the slave mode during interrupt handler processing might overlap with the changing of the IEBBnACKF bit.
-

---

**(e) Lock status flag (IEBBnLCKF): Bit 2**

A flag that indicates whether the unit is locked.

- Set/clear condition

Set:

When an individual communication frame ends, lock-related data (3H, 6H, AH, and BH) is received by using the control field, the communication completion flag (the IEBBnISR.IEBBnETRF bit) is cleared (to 0), and the frame completion flag (the IEBBnISR.IEBBnEFMF bit) is set (to 1).

Clear:

When an individual communication frame ends, lock-related data (3H, 6H, AH, and BH) is received by using the control field, and the communication completion flag (the IEBBnISR.IEBBnETRF bit) is set (to 1)

- 
- Cautions**
1. Locking and unlocking are performed only during individual communication, not during broadcast communication.
  2. While the master is locked, communication from a unit other than the master is not generally acknowledged. However, as an exception, if the control data (0H, 4H, or 5H) of a slave status transmission request is received, the communication is acknowledged even if it is not from the locked master. Note that, at this time, only a status request interrupt occurs, not a start interrupt or completion interrupt.
-



**(19) IEBBnISR - IEBBn interrupt status register**

The IEBBnISR status register indicates the interrupt source when an IEBBTSTA, IEBBTERR, or IEBBTV interrupt occurs.

Each time the IEBBTSTA, IEBBTERR, and IEBBTV interrupts occur, the IEBBnISR register is read, and the specified interrupt servicing is performed.

**Access** Only bit 6 can be read or written in 8-bit units.

Bits other than bit 6 are read-only, in 8-bit units.

**Address** <IEBBn\_base> + 0048H

**Initial value** 00H

The IEBBnIEBE bit is reset when 0 is written to the IEBBnBCR.IEBBnPW bit.

Bits other than IEBBnIEBE are reset when the value of the IEBBnPW bit is overwritten with a different value.

**Caution** Be sure to set bits 1, and 7 to 0.

7	6	5	4	3	2	1	0
0	IEBBn IEBE	IEBBn STRF	IEBBn STSF	IEBBn ETRF	IEBBn EFMF	0	IEBBn FOVE
R	R/W <sup>a</sup>	R	R	R	R	R	R

a) Only the IEBBnIEBE bit can be written. Note that, when writing to the IEBBnIEBE bit, the bit can only be cleared (to 0). Even if 1 is written, the IEBBnIEBE bit is not set (to 1).

**Table 22-27 IEBBnISR register contents**

Bit position	Bit name	Function
6	IEBBnIEBE	Communication error flag 0: No communication error has occurred. 1: A communication error has occurred.
5	IEBBnSTRF	Start interrupt flag 0: No start interrupt has occurred. 1: A start interrupt has occurred.
4	IEBBnSTSF	Status transmission flag (slave) 0: There is no status transmission request. 1: There is a status transmission request.
3	IEBBnETRF	Communication completion flag 0: Communication of the number of transmission bytes specified by the message length field has not finished. 1: Communication of the number of transmission bytes specified by the message length field has finished.
2	IEBBnEFMF	Frame completion flag 0: The frame (communication of the maximum number of transmission bytes <sup>a</sup> ) has not finished. 1: The frame (communication of the maximum number of transmission bytes <sup>a</sup> ) has finished.
0	IEBBnFOVE	Frame over error flag 0: No frame over error has occurred. 1: A frame over error has occurred.

a) Communication mode 1: 32 bits, communication mode 2: 128 bits

**(a) Communication error flag (IEBBnIEBE): Bit 6**

A flag that indicates a communication error has occurred.

- Set/clear condition

Set:

- Single mode:

When a timing error, parity error (except in the data field during individual reception), NACK reception error, underrun error, or overrun error (during broadcast reception) occurs

- FIFO mode:

When a timing error, parity error (except in the data field during individual reception), NACK reception error, underrun error, overrun error (during broadcast reception), or arbitration loss error occurs

Clear:

- Single mode or FIFO mode:

By software (The flag is cleared (to 0) when 0 is written to the IEBBnIEBE bit.)

When a communication error occurs, IEBBTERR IEBBTV occur in the single mode, and IEBBTERR occurs in the FIFO mode.

It is possible to determine what caused the error by reading the IEBBnESR and IEBBnISR registers.

**(b) Start interrupt flag: (IEBBnSTRF): Bit 5**

A flag that indicates the start interrupt.

- Set/clear condition

Set:

- Single mode:

The flag is set (to 1) during master unit operation, regardless of whether arbitration is won or lost.

The flag is set (to 1) during slave unit operation if there is a slave request (when the IEBBnUSR.IEBBnSRQF bit = 1) from the master (only the locked master when the unit is locked).

The flag is set upon the completion of the slave address field parity period in all cases.

- FIFO mode: The IEBBnSTRF bit is not set (to 1).

Clear:

- Single mode:

If the unit is the communication target (during communication with the master unit or slave unit), the flag is cleared (to 0) by hardware when a status transmission interrupt, communication completion interrupt, frame completion interrupt, transmission data write request interrupt, reception data read interrupt, or communication error interrupt occurs.

- FIFO mode: The IEBBnSTRF bit is normally cleared.

In the single mode, IEBBTSTA and IEBBTV occur when a start interrupt occurs. In the FIFO mode, the start interrupt, IEBBTSTA, and IEBBTV do not occur.

- 
- Cautions**
1. When a start interrupt occurs, read the IEBBnUSR register to check the slave request flag (IEBBnSRQF) and arbitration result flag (IEBBnARBF) for the unit.
  2. If the arbitration result flag (IEBBnARBF) is set (to 1) when a start interrupt occurs after the unit issues a master request, perform software processing to reissue the master request.
-

**(c) Status transmission flag (slave) (IEBBnSTSF): Bit 4**

This flag indicates that the master requested transmission of the slave status and lock address (higher 4 bits and lower 8 bits) when the controller was serving as a slave.

- Set/clear condition

Set:

- Single mode:

When the unit is not locked, there is a slave request from any master, and 0H or 6H is received by using the control field

When the unit is locked, there is a slave request from the locked master, and 0H, 4H, 5H, or 6H is received by using the control field, or when 0H, 4H, or 5H is received from a unit other than the locked master by using the control field

The flag is set upon the completion of the control field parity period in all cases.

For details, see *Table 22-28 “Conditions for setting the status transmission request flag (slave)”*.

- FIFO mode: The IEBBnSTSF bit is not set (to 1).

Clear:

- Single mode:

If the unit is the communication target (during communication with the master unit or slave unit), the flag is cleared (to 0) by hardware when a start interrupt, communication completion interrupt, frame completion interrupt, transmission data write request interrupt, reception data read interrupt, or communication error interrupt occurs.

- FIFO mode: The IEBBnSTSF bit is normally cleared.

In the single mode, IEBBTSTA and IEBBTV occur when there is a status transmission request. In the FIFO mode, no interrupt signal is generated even if there is a status transmission request.

---

**Caution** Even if the slave transmission enable flag (the IEBBnBCR.IEBBnSTXE bit) is set to the prohibited value (0), the IEBBnSTSF bit is set (to 1).

---

Table 22-28 Conditions for setting the status transmission request flag (slave)

Various statuses					Value received using the control field				
equa	lockf	eqpa	IEBBnSTXE	IEBBnSRXE	0H	3H, 7H	4H, 5H	6H	AH, BH, EH, FH
1	0	0	Any	Any	Set	Not set	Not set	Set	Not set
1	0	1	Any	Any	Set	Not set	Not set	Set	Not set
1	1	0	Any	Any	Set	Not set	Set	Not set	Not set
1	1	1	Any	Any	Set	Not set	Set	Set	Not set

**Note** equa: Unit match (during individual communication, IEBBnUAR register match)

lockf: Whether there is a lock

eqpa: Lock master match

IEBBnSTXE: Slave transmission enable flag (IEBBnBCR register bit 4)

IEBBnSRXE: Slave reception enable flag (IEBBnBCR register bit 3)

If a slave status transmission request interrupt occurs in the single mode, read the IEBBnCDR register to check the received control data contents, and then write the required slave status information to the IEBBnDR register.

The received control data and data written to the IEBBnDR register are shown below.

Table 22-29 Received control data and data written to the IEBBnDR register

Received control data	Function	Data written to the IEBBnDR register
0H, 6H	Slave status transmission	Value read from the IEBBnSSR register
4H	Transmission of the lower 8 bits of the lock address	Lower 8 bits of the IEBBnPAR register
5H	Transmission of the higher 8 bits of the lock address	Higher 8 bits of the IEBBnPAR register

**Caution** After a slave status transmission request interrupt occurs, be sure to write the appropriate data to the IEBBnDR register before the completion of the message length field.

**(d) Communication completion flag (IEBBnETRF): Bit 3**

A flag that indicates whether communication ends after the number of bytes set in the message length field have been transferred.

- Set/clear condition

Set:

- Single mode:

When the unit is the communication target (during communication with the master unit or slave unit) and the value of the IEBBnSCR register becomes 0 at the end of the data field acknowledge period

Clear:

- Single mode:

The flag is cleared (to 0) by hardware when a start interrupt, status transmission interrupt, frame completion interrupt (when a communication completion interrupt does not occur), transmission data write request interrupt, reception data read interrupt, or communication error interrupt occurs.

In the single mode, IEBBTSTA and IEBBTV occur when the communication completion flag is set (to 1).

In the FIFO mode, the IEBBTSTA interrupt occurs at the same timing as in the single mode.

In the FIFO mode, reading the IEBBnETRF bit is prohibited. If the bit is read, the returned value is undefined.

In the FIFO mode, after IEBBTSTA occurs, transmission and reception can be controlled by performing the software processing below.

- When reception ends
  1. Detecting the completion of communication based on the occurrence of the IEBBTSTA interrupt
  2. Checking whether communication or a frame has finished by using the IEBBnFSR.IEBBnRTRF bit
  3. Checking the number of received data bytes by using the IEBBnBSR.IEBBnSRFP4 to IEBBnSRFP0 bits (the number of bytes that have not been read)
  4. Reading the received data from the IEBBnDR register (the reception FIFO buffer)
- When transmission ends
  1. Detecting the completion of communication based on the occurrence of the IEBBTSTA interrupt
  2. Checking whether communication or a frame has finished by using the IEBBnFSR.IEBBnTTRF bit
  3. Checking the number of transmission data bytes by using the IEBBnBSR.IEBBnSTFP4 to IEBBnSTFP0 bits (the number of bytes that have not been transmitted)
  4. Proceeding to retransmission processing if there is data that has not been transmitted

**(e) Frame completion flag (IEBBnEFMF): Bit 2**

This flag indicates whether communication ends after the maximum number of bytes (communication mode 1: 32 bytes, communication mode 2: 128 bytes) have been transferred.

- Set/clear condition

Set:

- Single mode:

When the unit is the communication target (during communication with the master unit or slave unit) and the value of the IEBBnCCR register becomes 0 at the end of the data field acknowledge period

Clear:

- Single mode:

The flag is cleared (to 0) by hardware when a start interrupt, status transmission interrupt, communication completion interrupt (when a frame completion interrupt does not occur), transmission data write request interrupt, reception data read interrupt, or communication error interrupt occurs.

In the single mode, IEBBTSTA and IEBBTV occur when the frame completion flag is set (to 1).

- 
- Cautions**
1. In the single mode, if the IEBBnSCR and IEBBnCCR registers are both cleared to 00H at the end of the data field acknowledge period, the IEBBnETRF and IEBBnEFMF bits are set (to 1) at the same time.
  2. In the single mode, if the last data field is a NACK signal when the maximum number of bytes that can be transmitted is reached during retransmission, the IEBBnEFMF and IEBBnIEBE bits (NACK reception error) are set (to 1) at the same time.
- 

In the FIFO mode, the IEBBTSTA interrupt occurs at the same timing as in the single mode.

In the FIFO mode, reading the IEBBnEFMF bit is prohibited. If the bit is read, the returned value is undefined.

After IEBBTSTA occurs, transmission and reception can be controlled by performing the software processing below.

- When reception ends
  1. Detecting the completion of communication based on the occurrence of the IEBBTSTA interrupt
  2. Checking whether communication or a frame has finished by using the IEBBnFSR.IEBBnRTRF bit
  3. Checking the number of received data bytes by using the IEBBnBSR.IEBBnSRFP4 to IEBBnSRFP0 bits (the number of bytes that have not been read)
  4. Reading the received data from the IEBBnDR register (the reception FIFO buffer)

- When transmission ends
  1. Detecting the completion of communication based on the occurrence of the IEBBTSTA interrupt
  2. Checking whether communication or a frame has finished by using the IEBBnFSR.IEBBnTTRF bit
  3. Checking the number of transmission data bytes by using the IEBBnBSR.IEBBnSTFP4 to IEBBnSTFP0 bits (the number of bytes that have not been transmitted)
  4. Proceeding to retransmission processing if there is data that has not been transmitted

**(f) Frame over error flag (IEBBnFOVE): Bit 0**

This flag indicates that a frame over error has occurred.

- Set/clear condition

Set:

- Single mode: The IEBBnFOVE bit is not set (to 1).
- FIFO mode:

If the next broadcast communication is received as a slave unit before reading the data received during the previous communication finishes, the flag is set (to 1) after the first data field parity period finishes.

Clear:

- Single mode or FIFO mode:

By software

(The flag is cleared (to 0) when 1 is written to the IEBBnSTC1.IEBBnCLFF bit.)

In the single mode, the IEBBnFOVE bit is not set (to 1), and no interrupt requests occur.

In the FIFO mode, IEBBTERR occurs when the IEBBnFOVE bit is set (to 1).

Even if a frame over error occurs for the current frame, the data received during the previous frame is valid.



**(20) IEBBnESR - IEBBn error status register**

The IEBBnESR register is used to indicate the cause when an IEBus controller communication error interrupt occurs. Each bit of the IEBBnESR register is set (to 1) as soon as the communication error flag of the IEBBnISR register (IEBBnIEBE) is set (to 1). The cause of a communication error, if any, can be identified by checking the contents of the IEBBnESR register. (If the IEBBnISR.IEBBnIEBE bit is already set to 1, only the bits of the IEBBnESR register are set (to 1).)

Note that the bits can only be set (to 1) by hardware and are cleared by writing 1 to the IEBBnSTC0 register.

When 1 is written to the IEBBnSTC0 register, if there is contention with the hardware trying to specify (1), the hardware is prioritized.

Note that writing to the IEBBnESR register is invalid.

**Access** This register is read-only, in 8-bit units.

**Address** <IEBBn\_base> + 004CH

**Initial value** 00H

This register is reset when the value of the IEBBnBCR.IEBBnPW bit is overwritten with a different value.

**Caution** When a communication error occurs, IEBBn returns to the initial status and prepares for the next communication. Regardless of whether an error occurs, if the next communication is started without handling errors, any error flags that have been set remain set. (If the system returns to the initial status due to a timing error, and a parity error is received during the next communication, both the timing error and parity error bits of the IEBBnESR register are set to 1.) Therefore, handle any errors that occur before the next communication starts.

7	6	5	4	3	2	1	0
IEBBn TIME	IEBBn PARE	IEBBn NACE	IEBBn UNRE	IEBBn OVRE	0	IEBBn ABTE	IEBBn TRDE
R	R	R	R	R	R	R	R

Table 22-30 IEBBnESR register contents (1/2)

Bit position	Bit name	Function
7	IEBBnTIME	Timing error occurrence flag 0: No timing error has occurred. 1: A timing error has occurred.
6	IEBBnPARE	Parity error occurrence flag 0: No parity error has occurred. 1: A parity error has occurred.
5	IEBBnNACE	NACK reception error flag 0: No NACK reception error has occurred. 1: A NACK reception error has occurred.
4	IEBBnUNRE	Underrun error occurrence flag 0: No underrun error has occurred. 1: An underrun error has occurred.

Table 22-30 IEBBnESR register contents (2/2)

Bit position	Bit name	Function
3	IEBBnOVRE	Overrun error occurrence flag 0: No overrun error has occurred. 1: An overrun error has occurred.
1	IEBBnABTE	Arbitration loss error occurrence flag 0: The number of arbitration losses specified for the arbitration loss count setting bits (IEBBnTMS.IEBBnALC2 to IEBBnALC0) have not occurred. 1: The number of arbitration losses specified for the arbitration loss count setting bits (IEBBnTMS.IEBBnALC2 to IEBBnALC0) have occurred.
0	IEBBnTRDE	Inter-third-party communication error occurrence flag 0: An error occurred during communication targeting the unit. 1: An error occurred during inter-third-party communication.

**(a) Timing error occurrence flag (IEBBnTIME): Bit 7**

- Set condition

Set: This flag is set (to 1) if a timing error occurs.

A timing error occurs if the high-/low-level width of the communication bit is not the defined value.

The defined value of the high- and low-level width is set to the bit processing block and monitored by the internal timer.

**(b) Parity error occurrence flag (IEBBnPARE): Bit 6**

- Set condition

Set: This flag is set (to 1) if a parity error occurs.

When the unit is the reception unit (including while communication between others is being monitored), a parity error occurs when the parity data generated by the data received using the master address field, slave address field, control data field, or message length field does not match the received parity data.

However, when there is a data field mismatch, a NACK signal is returned and a data retransmission request is issued during individual communication, but a parity error occurs during broadcast communication.

**Note** During the above parity period, if the parity data received on the transmission side is inverted for some reason, a timing error occurs and communication ends.

**Table 22-31 Operation when the parity data does not match**

Field	Communication mode	Operation when the parity data does not match
Master address field	Individual/broadcast	A parity error occurs.
Slave address field	Individual/broadcast	A parity error occurs.
Control data field	Individual/broadcast	A parity error occurs.
Message length field	Individual/broadcast	A parity error occurs.
Data field	Individual	A NACK signal is returned to request retransmission.
	Broadcast	A parity error occurs.

**(c) NACK reception error flag (IEBBnNACE): Bit 5**

- Set condition

Set: This flag is set (to 1) if a NACK reception error occurs.

A NACK reception error occurs if a NACK signal is received during the acknowledge bit period of the slave address field, control data field, or message length field during individual communication, regardless of whether the controller is operating as the master or slave.

When a NACK signal is received for the data field, no NACK reception error generally occurs because this reception signals a data retransmission request. However, if the last data field is a NACK signal, a NACK reception error does occur.

During reception, a NACK reception error is judged to have occurred if an output NACK signal is received for the last data of the slave address field, control data field, message length field, or data field.

Note that, during broadcast communication, no NACK reception errors occur because ACK/NACK signal judgment is not performed.

No NACK reception errors occur during inter-third-party communication because only timing/parity errors are detected as errors. However, for the slave address field, NACK reception error judgment is performed because communication is participated in as a slave.

**Table 22-32 NACK reception error judgment period**

Communication mode		Slave address field	Control data field Data field	Message length field	Data field (last)
Individual communication	Master transmission	Occurs	Occurs	Occurs	Occurs
	Master reception	—	—	Occurs	Occurs
	Slave transmission	—	—	Occurs	Occurs
	Slave reception	Occurs	Occurs	Occurs	Occurs
	Inter-third-party communication	Occurs	Does not occur	Does not occur	Does not occur
Broadcast communication	All	Does not occur	Does not occur	Does not occur	Does not occur

**(d) Underrun error occurrence flag (IEBBnUNRE): Bit 4**

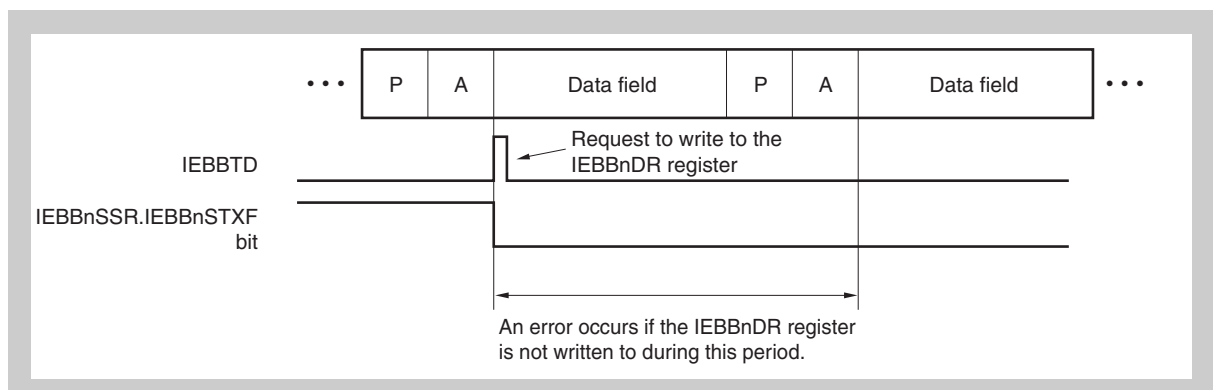
- Set condition

Set:

- Single mode:

During data transmission using the data field, an underrun error occurs if writing the next data to be transmitted to the IEBBnDR register does not finish before the end of the data-field acknowledge bit period following the occurrence of IEBBTD, and this flag is set (to 1). However, if the NACK signal is received during the acknowledge bit period, no underrun error occurs because retransmission is performed.

During inter-third-party communication, underrun errors do not occur because only timing/parity errors are detected as errors.



**Figure 22-6 Underrun error occurrence timing**

- FIFO mode:

Before data of the message length specified by the IEBBnTDL register is transmitted, if there are 0 data items to be transmitted remaining in the transmission FIFO buffer (if the IEBBnSSR.IEBBnSTXF bit = 0), an underrun error occurs if writing the next data to be transmitted to the IEBBnDR register does not finish before the end of the next data-field acknowledge bit period, and this flag is set (to 1). However, if the NACK signal is received during the acknowledge bit period, no underrun error occurs because retransmission is performed.

During inter-third-party communication, underrun errors do not occur because only timing/parity errors are detected as errors.

**(e) Overrun error occurrence flag (IEBBnOVRE): Bit 3**

- Set condition

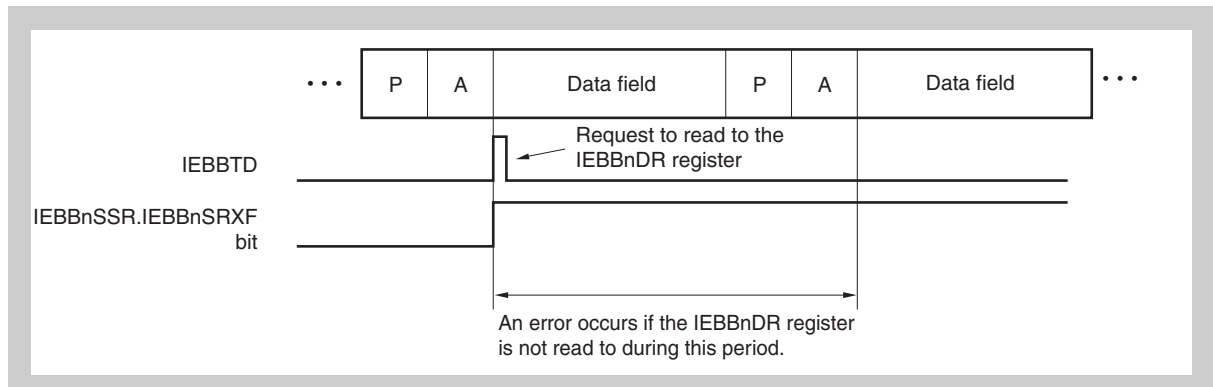
Set:

- Single mode:

When the data field is used to receive data during broadcast communication, an overrun error occurs if reading the IEBBnDR register does not finish between when IEBBTD occurs and when the parity period of the data field finishes, and this flag is set (to 1).

During individual communication, data retransmission is requested by returning a NACK signal without an error, and returning the NACK signal continues until the IEBBnDR register is read. (However, the frame ends when the maximum number of bytes that can be transferred is reached.)

During inter-third-party communication, overrun errors do not occur because only timing/parity errors are detected as errors.



**Figure 22-7** Overrun error occurrence timing

- FIFO mode:

During broadcast communication in communication mode 2, if there are 32 bytes of unread data in the reception FIFO buffer, an overrun error occurs if a 33rd byte of data is received, and this flag is set (to 1).

During individual communication, data retransmission is requested by returning a NACK signal without an error, and returning the NACK signal continues until the IEBBnDR register is read. (However, the frame ends when the maximum number of bytes that can be transferred is reached.)

If an overrun error occurs, do not read data by servicing interrupts.

If an interrupt occurs while reading data, it might no longer be possible to read the correct data.

**(f) Arbitration loss error occurrence flag (IEBBnABTE): Bit 1**

- Set condition

Set:

If arbitration losses occur the number of times specified for the IEBBnTMS.IEBBnALC2 to IEBBnALC0 bits, this flag is set (to 1) when the last arbitration loss error occurs.

For details about arbitration loss, see the description of the IEBBnARBF bit in 22.3.2 (18) "IEBBnUSR - IEBBn unit status register".

**(g) Inter-third-party communication error occurrence flag (IEBBnTRDE): Bit 0**

- Set condition

Set:

If an inter-third-party communication error occurs at the same time as a timing error or parity error that occurs during communication that is not related to the unit (inter-third-party communication), this flag is set (to 1) at the same time as the IEBBnTIME or IEBBnPARE bit.

---

**Caution** If an error occurs before the inter-third-party communication starts even when the slave address field does not match that of the unit (for example, if the NACK signal is received when the received address does not match that of the unit in the slave address field (if the IEBBnNACE bit is set (to 1))), the IEBBnTRDE bit is not set (to 1).

---

**Note** Communication between third parties may take place in the following two cases.

1. If the address received in the slave address field does not match that of the unit (during individual communication: matching with the IEBBnUAR register, during broadcast communication: matching with the group or FFFH) and communication continues after the  $\overline{\text{ACK}}$  signal has been received, the unit monitors that communication.
2. If the unit cannot respond to the received control data in the control field during broadcast communication and if communication continues, the unit monitors that communication. For example, this happens when the unit receives the control data FH from the master during broadcast communication but the slave reception enable flag of the unit is disabled (IEBBnBCR.IEBBnSRXE bit = 0). (During individual communication, the NACK signal is returned and communication ends.)

**(21) IEBBnFSR - IEBBn field status register**

The IEBBnFSR register is used to store the field status state of the IEBus controller when various interrupts (IEBBTD, IEBBTSTA, IEBBTERR, and IEBBTV) occur.

**Access** This register is read-only, in 8-bit units.

**Address** <IEBBn\_base> + 0050H

**Initial value** 00H

The IEBBnSSFS1 and IEBBnSSFS0 bits are reset when the IEBBnBCR.IEBBnPW bit is overwritten with a different value.

- Cautions**
1. If a different interrupt occurs before the IEBBnFSR register is read, the status information used at the time of the previous interrupt is overwritten with the status information used at the time of the new interrupt.
  2. If an interrupt occurs during communication between third parties (during the reception of communication between other units), the IEBBnSSFS1 and IEBBnSSFS0 bits are cleared to 00. However, because the only interrupts that occur during communication between third parties are interrupts caused by errors, an inter-third-party communication error can be judged to have occurred by reading the inter-third-party communication error occurrence flag (the IEBBnTRDE bit) of the IEBBnESR register.
  3. Even if the field status signal (an internal signal) changes, the IEBBnSSFS1 and IEBBnSSFS0 bits retain their previous values until an interrupt occurs.

7	6	5	4	3	2	1	0
IEBBn RTRF	IEBBn TTRF	0	0	0	0	IEBBn SSFS1	IEBBn SSFS0
R	R	R	R	R	R	R	R

**Table 22-33 IEBBnFSR register contents**

Bit position	Bit name	Function
7	IEBBnRTRF	Reception communication completion flag 0: Communication did not finish during reception. 1: Communication finished during reception.
6	IEBBnTTRF	Transmission communication completion flag 0: Communication did not finish during transmission. 1: Communication finished during transmission.
1, 0	IEBBnSSFS[1, 0]	For details about the IEBBnSSFS1 and IEBBnSSFS0 bits, see <i>Table 22-34 "Field status"</i> .



Table 22-34 Field status

Field status	Description		
	Master/slave	Field	Transmission/reception
Slave reception status IEBBnSSFS1 and IEBBnSSFS0 bits = 00 (IEBBnFSR register = 00H)	Slave operation	Start bit	Reception
		Master address field	
		Slave address field	
		Control data field	
		Message length field	
		Data field	
Slave transmission status IEBBnSSFS1 and IEBBnSSFS0 bits = 01 (IEBBnFSR register = 01H)	Slave operation	Message length field	Transmission
		Data field	
Master reception status IEBBnSSFS1 and IEBBnSSFS0 bits = 10 (IEBBnFSR register = 02H)	Master operation	Message length field	Reception
		Data field	
Master transmission status IEBBnSSFS1 and IEBBnSSFS0 bits = 11 (IEBBnFSR register = 03H)	Master operation	Start bit	Transmission
		Master address field	
		Slave address field	
		Control data field	
		Message length field	
		Data field	

---

**(a) Reception communication completion flag (IEBBnRTRF): Bit 7**

This flag indicates that communication equivalent to the number of bytes specified by the message length has finished during reception.

- Set/clear condition

Set:

- Single mode: The IEBBnRTRF bit is not set (to 1).
- FIFO mode:

During reception, the flag is set (to 1) when the IEBBnISR.IEBBnETRF bit is set (to 1).

Clear:

- Single mode: The IEBBnRTRF bit always has the clear status.
- FIFO mode:

During reception, the flag is cleared (to 0) when the IEBBnISR.IEBBnEFMF bit is set (to 1).

- 
- Cautions**
1. In the FIFO mode, if the set and clear conditions are both satisfied, setting the flag is prioritized.
  2. The IEBBnRTRF bit is not cleared (to 0) by writing 1 to the IEBBnPCR.IEBBnCRPT bit.
-

---

**(b) Transmission communication completion flag (IEBBnTTRF): Bit 6**

This flag indicates that communication equivalent to the number of bytes specified by the message length has finished during transmission.

- Set/clear condition

Set:

- Single mode: The IEBBnTTRF bit is not set (to 1).
- FIFO mode:

During transmission, the flag is set (to 1) when the IEBBnISR.IEBBnETRF bit is set (to 1).

Clear:

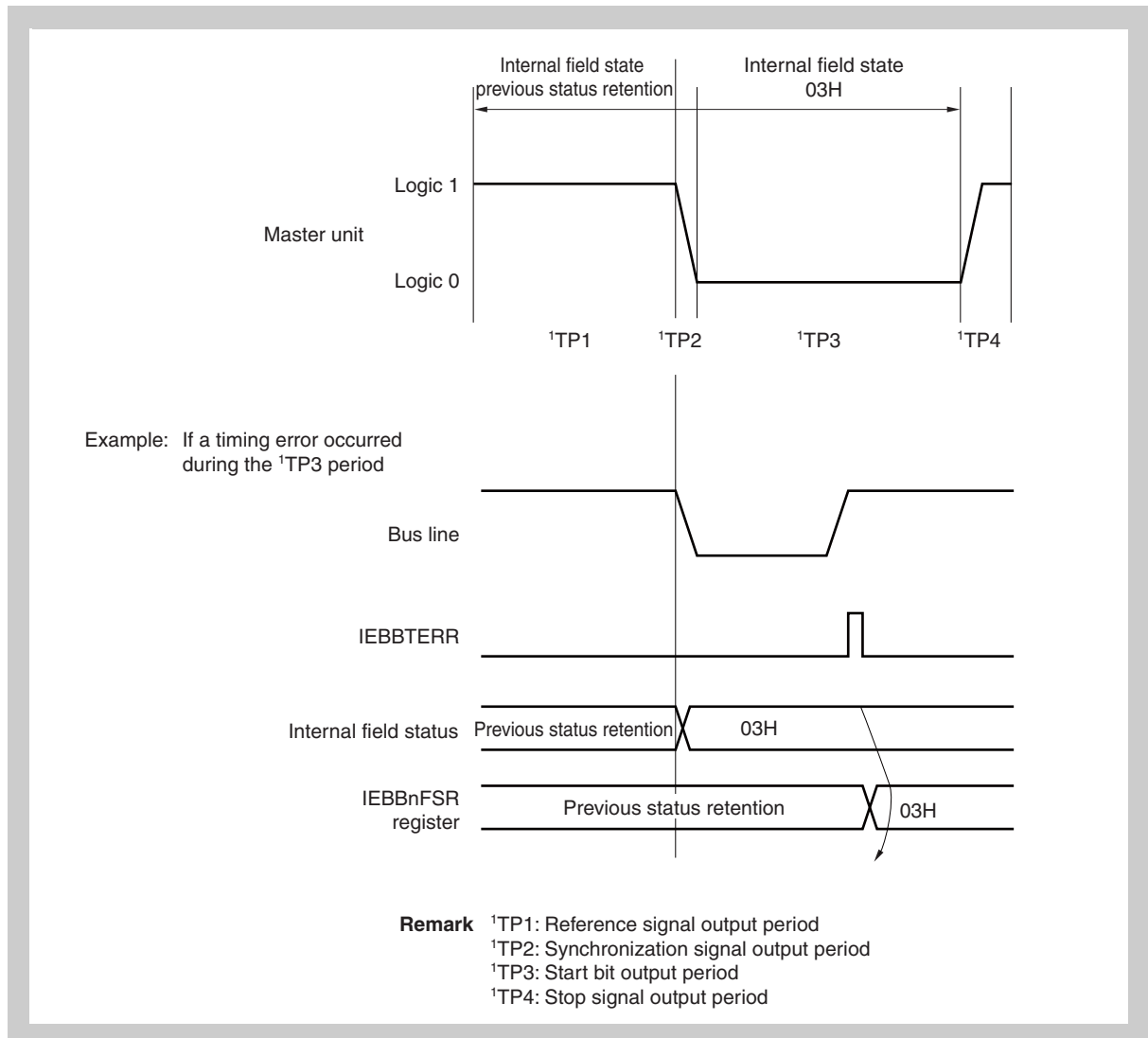
- Single mode: The IEBBnTTRF bit always has the clear status.
- FIFO mode:

During transmission, the flag is cleared (to 0) when the IEBBnISR.IEBBnEFMF bit is set (to 1).

- 
- Cautions**
1. In the FIFO mode, if the set and clear conditions are both satisfied, setting the flag is prioritized.
  2. The IEBBnTTRF bit is not cleared (to 0) by writing 1 to the IEBBnPCR.IEBBnCTPT bit.
-

**(c) Field status flags (IEBBnSSFS1 and IEBBnSSFS0): Bits 1 and 0**

These flags store the state of the IEBus controller field status when various interrupts (IEBBTD, IEBBTSTA, IEBBTERR, and IEBBTV) occur.



**Figure 22-8 Start bit field status for the master (internal signal)**

When the start bit shown in *Figure 22-8 “Start bit field status for the master (internal signal)”* is output for the master, the previous field status value is retained until <sup>1</sup>TP1. At point <sup>1</sup>TP2 and after, the field status value is 03H. If a timing error occurs at point <sup>1</sup>TP3 and IEBBTERR is output, 03H is stored in the IEBBnFSR register.

Because IEBBTERR does not occur if communication is performed normally, the field status value is not stored in the IEBBnFSR register, and the IEBBnFSR register retains the previous value at and after point <sup>1</sup>TP2.

**(22) IEBBnSCR - IEBBn success count register**

The IEBBnSCR register indicates the number of remaining communication bytes.

The value specified by the IEBBnDLR register is stored in the IEBBnSCR register after the message length field processing finishes, and the count value of the counter to be decremented according to the data field ACK signal is read.

In other words, because the number of successfully communicated bytes is subtracted from the number of data bytes to be communicated, the IEBBnSCR register indicates the remaining number of bytes to be communicated.

Note that the communication completion flag (the IEBBnISR.IEBBnETRF bit) is set (to 1) when the count value reaches 00H.

The data in the IEBBnSCR register is updated when the  $\overline{\text{ACK}}$  signal is received at the end of the message length field parity period or data field acknowledge bit period.

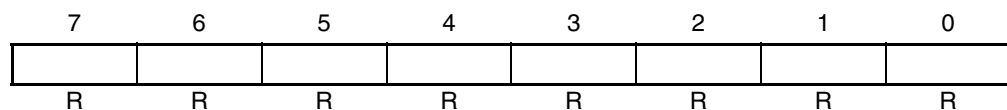
**Access** This register is read-only, in 8-bit units.

**Address** <IEBBn\_base> + 0054H

**Initial value** 01H

This register is reset when the value of the IEBBnBCR.IEBBnPW bit is overwritten with a different value.

**Caution** When 00H is read from the IEBBnSCR register, it is not possible to judge whether the remaining number of communication data bytes is 0 (indicating communication completion) or 256. Therefore, the communication completion flag (the IEBBnISR.IEBBnETRF bit) must also be used with this register to make this judgment.



**Table 22-35 IEBBnSCR register contents**

Bit								Setting	Number of remaining communication data bytes
7	6	5	4	3	2	1	0		
0	0	0	0	0	0	0	1	01H	1 byte
0	0	0	0	0	0	1	0	02H	2 bytes
...	...	...	...	...	...	...	...	...	...
0	0	0	1	0	1	0	0	20H	32 bytes
...	...	...	...	...	...	...	...	...	...
1	1	1	1	1	1	1	1	FFH	255 bytes
0	0	0	0	0	0	0	0	00H	0 bytes (communication completion) or 256 bytes

**(23) IEBBnCCR - IEBBn communication count register**

The IEBBnCCR register indicates the number of bytes remaining from the communication byte number specified by the communication mode.

This register indicates the number of transfer bytes.

The maximum number of transmitted bytes per frame defined in each mode (communication mode 1: 32 bytes, communication mode 2: 128 bytes) is preset to this register. The count value of the counter that is decremented during the acknowledge bit period of the data field regardless of the  $\overline{\text{ACK}}$ /NACK signal is read from this register. In contrast with the IEBBnSCR register, which is decremented when there is normal communication (the ACK signal), the IEBBnCCR register is decremented when one byte is communicated, regardless of the ACK/NACK signal. Note that the frame completion flag (the IEBBnISR.IEBBnEFMF bit) is set (to 1) when the counter reaches 00H.

The preset value of the maximum number of transmitted bytes per frame is 20H (32 bytes) in communication mode 1 and 80H (128 bytes) in communication mode 2.

Updating of data proceeds independently of parity at the end of the ACK interval for the data field on completion of transmission (or reception) of the start bit for the preset maximum number of bytes.

**Access** This register is read-only, in 8-bit units.

**Address** <IEBBn\_base> + 0058H

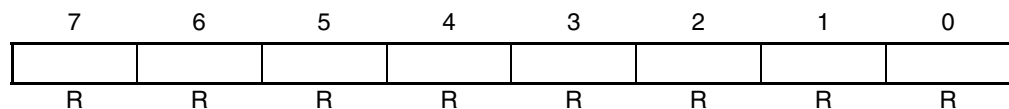
**Initial value** 20H

This register is reset when the value of the IEBBnBCR.IEBBnPW bit is overwritten with a different value.

---

**Caution** The value of the IEBBnCCR register is not updated by writing to the IEBBnPSR.IEBBnCMD bit.

---



**(24) IEBBnSTC0 - IEBBn status clear register 0**

The IEBBnSTC0 register is used to clear the IEBBnESR register.

**Access** This register is write-only, in 8-bit units.

**Address** <IEBBn\_base> + 005CH

**Initial value** 00H

7	6	5	4	3	2	1	0
IEBBn CLTM	IEBBn CLPA	IEBBn CLNC	IEBBn CLUR	IEBBn CLOV	0	IEBBn CLAB	IEBBn CLTR
W	W	W	W	W	R	W	W

**Table 22-36 IEBBnSTC0 register contents**

Bit position	Bit name	Function
7	IEBBnCLTM	This bit is used to clear the timing error flag (IEBBnESR.IEBBnTIME). 0: No operation 1: Clear the IEBBnTIME bit. Writing 1 is valid, and writing 0 does not change the internal status. When the bit is read, 0 is always returned.
6	IEBBnCLPA	This bit is used to clear the parity error flag (IEBBnESR.IEBBnPARE). 0: No operation 1: Clear the IEBBnPARE bit. Writing 1 is valid, and writing 0 does not change the internal status. When the bit is read, 0 is always returned.
5	IEBBnCLNC	This bit is used to clear the NACK reception error flag (IEBBnESR.IEBBnNACE). 0: No operation 1: Clear the IEBBnNACE bit. Writing 1 is valid, and writing 0 does not change the internal status. When the bit is read, 0 is always returned.
4	IEBBnCLUR	This bit is used to clear the underrun error flag (IEBBnESR.IEBBnUNRE). 0: No operation 1: Clear the IEBBnUNRE bit. Writing 1 is valid, and writing 0 does not change the internal status. When the bit is read, 0 is always returned.
3	IEBBnCLOV	This bit is used to clear the overrun error flag (IEBBnESR.IEBBnOVRE). 0: No operation 1: Clear the IEBBnOVRE bit. Writing 1 is valid, and writing 0 does not change the internal status. When the bit is read, 0 is always returned.
1	IEBBnCLAB	This bit is used to clear the arbitration loss error flag (IEBBnESR.IEBBnABTE). 0: No operation 1: Clear the IEBBnABTE bit. Writing 1 is valid, and writing 0 does not change the internal status. When the bit is read, 0 is always returned.
0	IEBBnCLTR	This bit is used to clear the inter-third-party communication error flag (IEBBnESR.IEBBnTRDE). 0: No operation 1: Clear the IEBBnTRDE bit. Writing 1 is valid, and writing 0 does not change the internal status. When the bit is read, 0 is always returned.

**(25) IEBBnSTC1 - IEBBn status clear register 1**

The IEBBnSTC1 register is used to clear the IEBBnISR.IEBBnFOVE bit.

**Access** This register is write-only, in 8-bit units.

**Address** <IEBBn\_base> + 0060H

**Initial value** 00H

7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	IEBBn CLFF
R	R	R	R	R	R	R	W

**Table 22-37 IEBBnSTC1 register contents**

Bit position	Bit name	Function
0	IEBBnCLFF	This bit is used to clear the frame over error flag (IEBBnISR.IEBBnFOVE). 0: No operation 1: Clear the IEBBnFOVE bit. Writing 1 is valid, and writing 0 does not change the internal status. When the bit is read, 0 is always returned.



**(26) IEBBnDR - IEBBn data register**

The IEBBnDR register is used to set up the communication data. Specify the communication data (8 bits) for bits 7 to 0.

- Notes**
1. The IEBBnDR register consists of a write register and a read register. Therefore, data written to this register cannot be read as is. The data received during IEBus communication can be read.
  2. In the FIFO mode, the data in the FIFO buffer can be transferred by continuously accessing the IEBBnDR register. (See 22.5.1 (1) "Transmission FIFO buffer" for details about using the transmission unit or 22.5.1 (2) "Reception FIFO buffer" for details about using the reception unit.)

**(a) For the transmission unit**

If the unit is the transmission unit (during master or slave transmission), the bits in the data field are transmitted as data bits starting with the highest bits when writing to the IEBBnDR register.

Specify the first byte of transmission data before starting communication (IEBBnBCR.IEBBnMSRQ bit = 0).

Even in the FIFO mode, write at least one byte of transmission data to the FIFO buffer before starting communication (IEBBnMSRQ bit = 0).

In the FIFO mode, the contents of the FIFO buffer are not reset in sync with the IEBBnBCR.IEBBnPW bit. Clearing the pointer value (to 0) eliminates the remaining data visible to the user. The stored data becomes undefined.

During transmission (master or slave transmission), if a data interrupt (IEBBTD) occurs, the next transmission data is written to the IEBBnDR register.

In the single mode, if a status transmission interrupt (IEBBTSTA or IEBBTV) occurs, the status data is written to the IEBBnDR register according to the control data.

**(b) For the reception unit**

The 1 byte of data received using the data field is read from the IEBBnDR register if the unit is the reception unit (master or slave reception). Storage is performed at the end of the data field parity period if the parity value is normal. The read value is reset by clearing the IEBBnPW bit to 0.

During reception (master or slave reception), if a data interrupt (IEBBTD) occurs, received data is read from the IEBBnDR register.

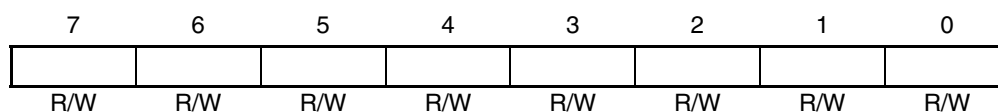
**Access** This register can be read or written in 8-bit units.

**Address** <IEBBn\_base> + 0064H

**Initial value** 00H

The read value is reset when the IEBBnBCR.IEBBnPW bit is overwritten.

- Cautions**
1. If writing to the IEBBnDR register is not in time for transmission, an underrun error occurs and communication ends.
  2. Write to the status data register IEBBnDR after a status transmission interrupt occurs and before the end of the message length field.
  3. In the single mode, if the IEBBnDR register is not read before the next reception, the communication differs in the case of individual versus broadcast communication.
    - For individual communication, a NACK signal is returned for the field, and the master is requested to transmit the same data. Received data is not stored in the IEBBnDR register. If the NACK signal is returned again before the IEBBnDR register is read and the register has still not been read when the maximum number of transferable bytes is reached, frame completion (IEBBTSTA, IEBBTV) and a NACK reception error (IEBBTERR, IEBBTV) occur at the same time.
    - During broadcast communication, an overrun error occurs and communication ends. Received data is not stored in the IEBBnDR register. The overrun error flag (IEBBnOVRE) is set (to 1).
  4. In the FIFO mode, if the next reception occurs before all the data received during the previous communication is read, the communication differs in the case of individual versus broadcast communication.
    - For individual communication, a NACK signal is returned for the data field, and the master is requested to transmit the same data. Received data is not stored in the FIFO buffer. If the NACK signal is returned again before all the data in the FIFO buffer is read and the FIFO buffer has still not been read when the maximum number of transferable bytes is reached, frame completion (IEBBTSTA) and a NACK reception error (IEBBTERR) occur at the same time.
    - During broadcast communication, an overrun error occurs and communication ends. Received data is not stored in the FIFO buffer. The frame over error flag (IEBBnISR.IEBBnFOVE) is set (to 1).
  5. In the FIFO mode and communication mode 2, if there are 32 bytes of unread data and the next reception occurs, the communication differs in the case of individual versus broadcast communication.
    - For individual communication, a NACK signal is returned for the field, and the master is requested to transmit the same data. Received data is not stored in the FIFO buffer. If the NACK signal is returned again before the IEBBnDR register is read and reading the register has still not finished when the maximum number of transferable bytes is reached, frame completion (IEBBTSTA) and a NACK reception error (IEBBTERR) occur at the same time.
    - During broadcast communication, an overrun error occurs and communication ends. Received data is not stored in the FIFO buffer. The overrun error flag (IEBBnESR.IEBBnOVRE) is set (to 1).



## 22.4 Interrupt Operations

### 22.4.1 Interrupt request signals

Various interrupts occur in response to the eight interrupt requests below. The high level width of each interrupt signal is one P0 $\phi$  clock cycle.

The interrupts that occur differ depending on whether the system is in the single or FIFO mode.

#### (1) Single mode

The causes of interrupts in the single mode are shown below.

Table 22-38 Causes of interrupts in the single mode

Symbol	IEBBTD	IEBBTV	IEBBTERR	IEBBTSTA	Interrupt cause
IEBBnIEBE		Occurs	Occurs		Communication error (When the IEBBnISR.IEBBnIEBE bit = 1) Note that IEBBnIEBE occurs when the following bits of the IEBBnESR register = 1. <ul style="list-style-type: none"> <li>• Timing error (IEBBnTIME)</li> <li>• Parity error (IEBBnPARE)</li> <li>• NACK reception error (IEBBnNACE)<sup>a</sup></li> <li>• Underrun error (IEBBnUNRE)</li> <li>• Overrun error (IEBBnOVRE)</li> </ul>
IEBBnSTRF		Occurs		Occurs	Start request (when the IEBBnISR.IEBBnSTRF bit = 1)
IEBBnSTSF		Occurs		Occurs	Status transmission request (when the IEBBnISR.IEBBnSTSF bit = 1)
IEBBnETRF		Occurs		Occurs	End of communication (When the IEBBnISR.IEBBnETRF bit = 1)
IEBBnEFMF		Occurs		Occurs	End of frame (When the IEBBnISR.IEBBnEFMF bit = 1) <sup>a</sup>
IEBBnFOVE					Frame over (When the IEBBnISR.IEBBnFOVE bit = 1)
WRREQ	Occurs				Transmission data write request (when the IEBBnSSR.IEBBnSTXF bit = 0) <sup>b</sup>
RDREQ	Occurs				Reception data read request (when the IEBBnSSR.IEBBnSRXF bit = 1) <sup>c</sup>

<sup>a)</sup> If the frame data ends with a NACK signal, IEBBTV and IEBBTSTA are triggered by setting the frame completion indicating bit IEBBnISR.IEBBnEFMF (to 1).

At this time, the IEBBTERR and IEBBTV interrupts are triggered by a NACK reception error.

Three interrupts (IEBBTV, IEBBTSTA, and IEBBTERR) occur at the same time.

- b) During master transmission:
1. IEBBTD occurs after receiving the  $\overline{\text{ACK}}$  signal, which follows message length field transmission. However, if the transfer size is one byte (the IEBBnTDL register = 01H), IEBBTD does not occur.
  2. IEBBTD occurs after receiving the  $\overline{\text{ACK}}$  signal, which follows data field transmission. However, IEBBTD does not occur before transmitting the final data, or after transmitting the final data and then receiving the  $\overline{\text{ACK}}$  signal. More specifically, if the message length is five bytes, IEBBTD does not occur after transmitting the 4th or 5th byte. In addition, when transmitting the maximum number of transferable bytes for one frame, IEBBTD does not occur after transmitting byte number (maximum number of transferable bytes – 1) or byte number (maximum number of transferable bytes).
- During slave transmission:
1. IEBBTD occurs after receiving the  $\overline{\text{ACK}}$  signal, which follows message length field transmission. However, if the transfer size is one byte or the received control bit is a status request (0H, 4H, 5H, or 6H), IEBBTD does not occur (and a status transmission interrupt occurs instead).
  2. After data field transmission, the operation is the same as for 2 under During master transmission.
- c) RDREQ occurs after receiving the parity bit by using the data field. However, if the self-transmitted parity bit differs from the received parity bit, there is a timing error and no interrupt occurs.

**(2) FIFO mode**

The causes of interrupts in the FIFO mode are shown below.

**Table 22-39 Causes of interrupts in the FIFO mode**

Symbol	IEBBTD	IEBBTV	IEBBTERR	IEBBTSTA	Interrupt cause
IEBBnIEBE			Occurs		Communication error (When the IEBBnISR.IEBBnIEBE bit = 1) Note that IEBBnIEBE occurs when the following bits of the IEBBnESR register = 1. <ul style="list-style-type: none"> <li>• Timing error (IEBBnTIME)</li> <li>• Parity error (IEBBnPARE)</li> <li>• NACK reception error (IEBBnNACE)<sup>a</sup></li> <li>• Underrun error (IEBBnUNRE)</li> <li>• Overrun error (IEBBnOVRE)<sup>b</sup></li> <li>• Arbitration loss error (IEBBnABTE)</li> </ul>
IEBBnSTRF					Start request (when the IEBBnISR.IEBBnSTRF bit = 1)
IEBBnSTSF					Status transmission request (when the IEBBnISR.IEBBnSTSF bit = 1)
IEBBnETRF		Occurs <sup>c</sup>		Occurs	End of communication (When the IEBBnISR.IEBBnETRF bit = 1)
IEBBnEFMF		Occurs <sup>c</sup>		Occurs	End of frame (When the IEBBnISR.IEBBnEFMF bit = 1) <sup>a</sup>
IEBBnFOVE			Occurs		Frame over (When the IEBBnISR.IEBBnFOVE bit = 1) <sup>b</sup>
WRREQ	Occurs <sup>d</sup>				Transmit data write request
RDREQ		Occurs <sup>e</sup>			Receive data write request Parity

a) If the frame data ends with a NACK signal, IEBBTV and IEBBTSTA are triggered by setting the frame completion indicating bit IEBBnISR.IEBBnEFMF (to 1).  
Note that the IEBBTERR interrupt is triggered by a NACK reception error.  
Three interrupts (IEBBTV, IEBBTSTA, and IEBBTERR) occur at the same time, regardless of transmission or reception.

b) If data is received during the broadcast communication for the next frame while the FIFO buffer is full due to the reception of the previous frame and the data has not been read, the IEBBnISR.IEBBnFOVE and IEBBnESR.IEBBnOVRE bits are set (to 1) at the same time.

- c) IEBBTV occurs based on the same conditions and at the same timing as IEBBTSTA. (For details, see Table 22-40 " $\overline{\text{ACK}}/\text{NACK}$  for IEBBnETRF and IEBBnEFMF".)
- For individual communication:
- Transmission-side device
- IEBBnETRF:  
This is set (to 1) after receiving the  $\overline{\text{ACK}}$  signal. If the NACK signal is received, communication does not end.
- IEBBnEFMF: This is set (to 1) after reception regardless of the  $\overline{\text{ACK}}/\text{NACK}$  signal.
  - Reception-side device

IEBBnETRF:  
This is set (to 1) after transmitting the  $\overline{\text{ACK}}$  signal. If the NACK signal is transmitted, communication does not end.

IEBBnEFMF:  
This is set (to 1) after transmission regardless of the  $\overline{\text{ACK}}/\text{NACK}$  signal.  
The NACK signal is output when there is no room in the reception FIFO buffer or when there is a data retransmission request due to a parity mismatch. Not returning an  $\overline{\text{ACK}}$  signal to the reception side as the bus status after exiting communications due to the detection of an error does not constitute a NACK signal. In this case, IEBBTSTA and IEBBTV do not occur because the system does not enter the frame completion status.

For broadcast communication:  
For broadcast communication, no  $\overline{\text{ACK}}$  signal is returned from the slave. Therefore, it is judged that the NACK signal was successfully returned regardless of master transmission or slave reception, and the IEBBnETRF or IEBBnEFMF interrupt occurs.

d) This occurs when the conditions specified by the IEBBnTMS.IEBBnSLTI1 and IEBBnSLTI0 bits are satisfied. The occurrence timing is after receiving the  $\overline{\text{ACK}}$  signal, which follows data field transmission. However, if the transfer size is one byte or the received control bit is a status request (0H, 4H, 5H, or 6H), IEBBTD does not occur.

e)

  - Transmission-side device  
RDREQ is not set under this condition.
  - Reception-side device  
RDREQ is set (to 1) when the conditions specified by the IEBBnTMS.IEBBnSLRI1 and IEBBnSLRI0 bits are satisfied (after confirming that a normal parity bit value has been received).  
If there is a parity bit mismatch, RDREQ is not set (to 1) because the conditions are not satisfied.  
On normal completion or frame completion, RDREQ interrupt sources are masked.

Table 22-40  $\overline{\text{ACK}}/\text{NACK}$  for IEBBnETRF and IEBBnEFMF

Field status	Individual communication				Broadcast communication			
	IEBBnETRF		IEBBnEFMF		IEBBnETRF		IEBBnEFMF	
	$\overline{\text{ACK}}$	NACK	$\overline{\text{ACK}}$	NACK	$\overline{\text{ACK}}$	NACK	$\overline{\text{ACK}}$	NACK
Master transmission	Occurs	Does not occur	Occurs	Occurs	–	Occurs	–	Occurs
Master reception	Occurs	Does not occur	Occurs	Occurs	–	–	–	–
Slave transmission	Occurs	Does not occur	Occurs	Occurs	–	–	–	–
Slave reception	Occurs	Does not occur	Occurs	Occurs	–	Occurs	–	Occurs

**Note** The IEBBTV usage method is shown below.

1. Generating IEBBTV before communication finishes (when the IEBBnTMS.IEBBnSLRI1 and IEBBnSLRI0 bit settings are 32 bits or less)  
This operation is not generally performed in communication mode 1. (It can be performed but is not.) The operation is performed in communication mode 2.
  - Use IEBBTV to check the number of data bytes received by the FIFO buffer, and then perform a readout operation. (Be sure to check the number because the IEBBTV interrupt servicing is assumed to be late.)
  - For IEBBTSTA (communication completion/frame completion), check the status.
  - Because the status is changed after receiving data, the interrupt priority is IEBBTV followed by IEBBTSTA.
2. If not generating IEBBTV during communication (if the IEBBnSLRI1 and IEBBnSLRI0 bit settings are 32 bytes)  
Use communication mode 1. (Communication mode 2 can also be used.)
  - Communication mode 1Mask IEBBTV (so it is not used).  
Use IEBBTSTA (communication completion/frame completion) to check the number of data bytes in the FIFO buffer, and then check the readout and status changes.
  - Communication mode 2Perform the same operations as in 1.

## 22.4.2 Interrupt judgment examples

Interrupt judgment examples for the single mode are shown below.

### (1) When using IEBBTD

IEBBn transmission/reception must be checked by issuing an IEBBTD interrupt.

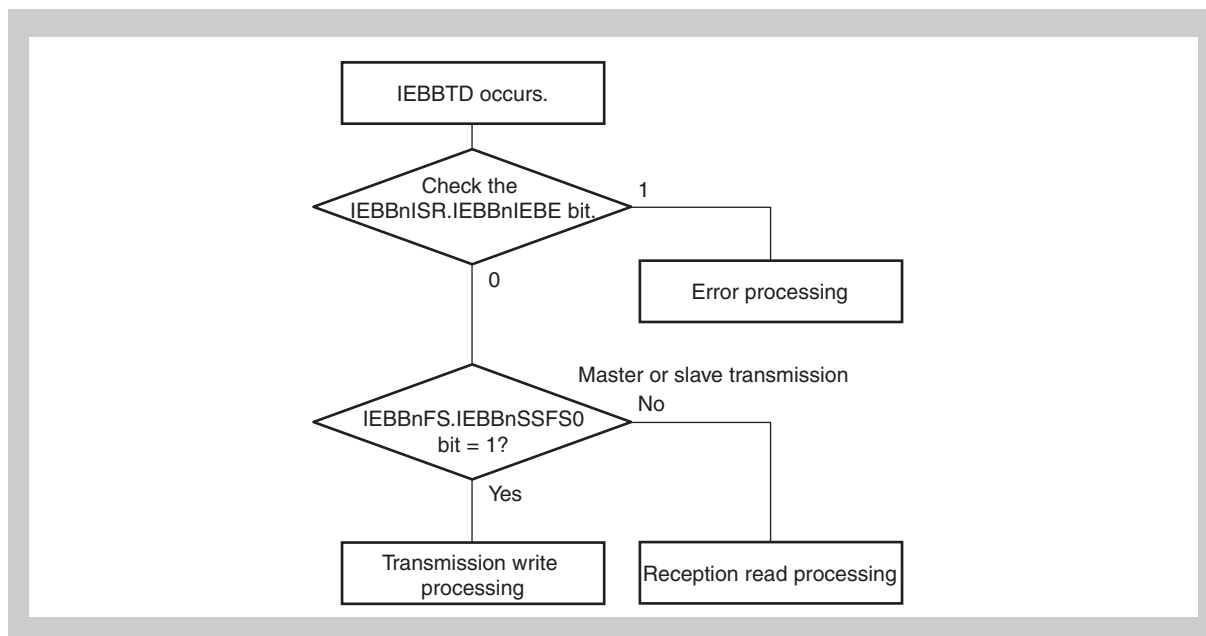


Figure 22-9 IEBBTD interrupt judgment example

**Caution** Even if IEBBTD occurs, an error might occur depending on when the interrupt is handled.

Such errors include timing errors after IEBBTD occurs. To increase data processing reliability, only handle data after using the IEBBnISR.IEBBnIEBE bit to make sure that no error has occurred.

### (2) When using IEBBTERR

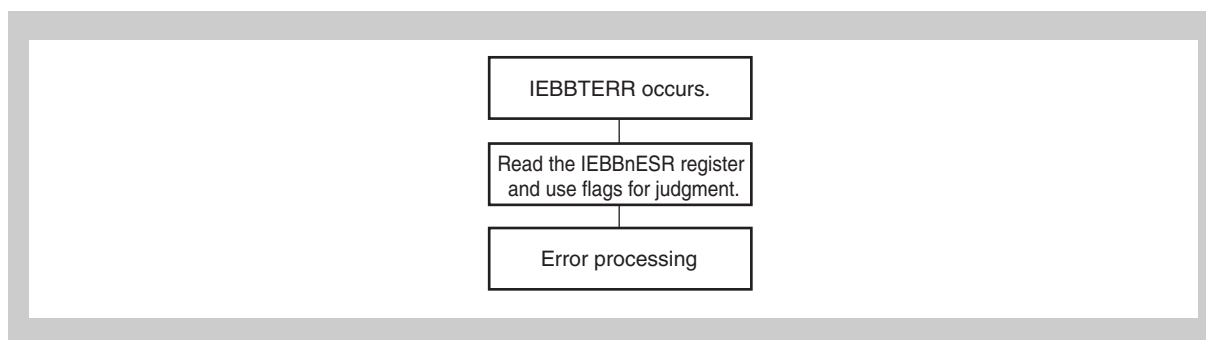


Figure 22-10 IEBBTERR interrupt judgment example



(3) When using IEBBTSTA or IEBBTV

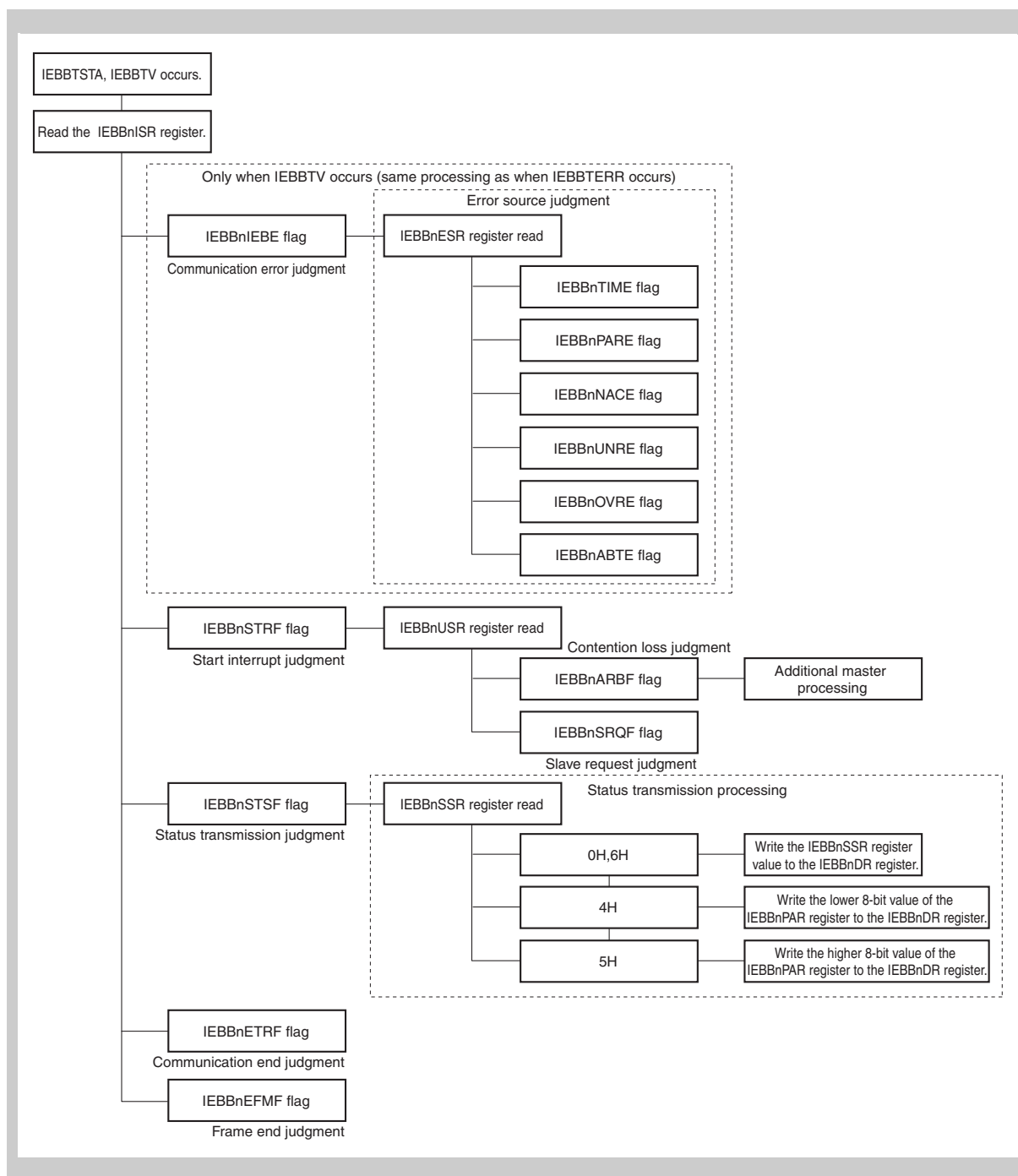


Figure 22-11 IEBBTSTA and IEBBTV interrupt judgment examples

## 22.5 Operation

### 22.5.1 FIFO

#### (1) Transmission FIFO buffer

When the IEBBnTMS.IEBBnFMDE bit = 1, data can be stored in the FIFO buffer while automatically incrementing the FIFO buffer pointer for writing by continuously writing transmission data to the IEBBnDR register. The FIFO buffer size is 8 bits x 32 stages.

When the transfer is started, the data indicated by the load pointer is transferred. After the transfer finishes, the load pointer is incremented. The initial value for the write pointer and load pointer is 0.

The IEBBnBSR.IEBBnTFLF bit is set (to 1) when there are 32 bytes of data in the FIFO buffer, and the transmission FIFO buffer overwrite flag (IEBBnBSR.IEBBnFOVW) is set (to 1) when a 33rd byte of data is written while the IEBBnTFLF bit = 1. At this time, the write data is ignored and the write pointer is not changed.

The data below can be written when one byte is transferred while the IEBBnTFLF bit = 1 and then the bit is cleared to 0.

If the write is not in time for data loading, an underrun error occurs.

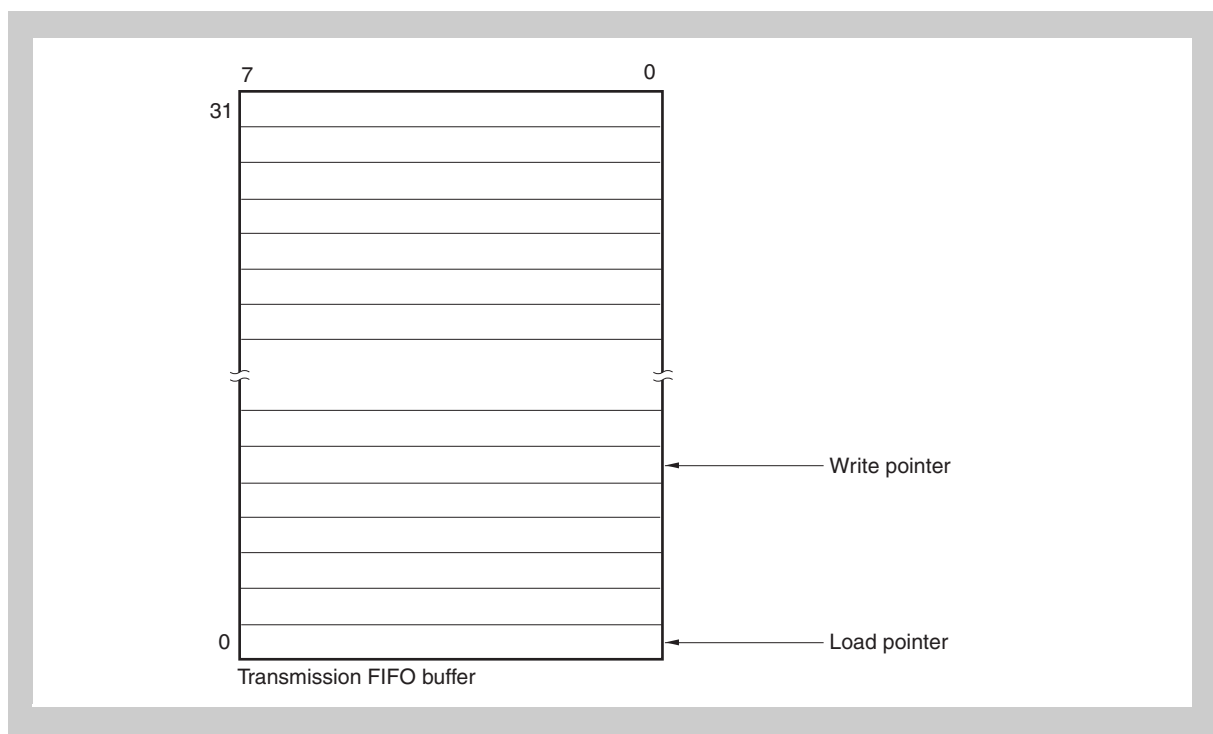


Figure 22-12 Transmission FIFO buffer

## (2) Reception FIFO buffer

When the IEBBnTMS.IEBBnFMDE bit = 1, the received data is stored at the address indicated by the storage pointer. The storage pointer is incremented after storing the data. Data is stored in the FIFO buffer at the end of the data field parity period if the parity value is normal. The FIFO buffer size is 8 bits x 32 stages.

By reading the IEBBnDR register, the data in the FIFO buffer can be read while automatically incrementing the read pointer. The initial value for the read pointer and storage pointer is 0.

If there are 32 bytes of unread data in the FIFO buffer, the operation when the next data is received is as follows.

- During individual communication: No data is stored, a NACK signal is returned, and data retransmission is requested.
- During broadcast communication: No data is stored and an overrun error occurs.

If the IEBBnDR register is read again after reading the received data that has been stored finishes (when the read pointer = the storage pointer), the read pointer is not changed, and the over-read flag (IEBBnBSR.IEBBnFOVR bit) is set (to 1).

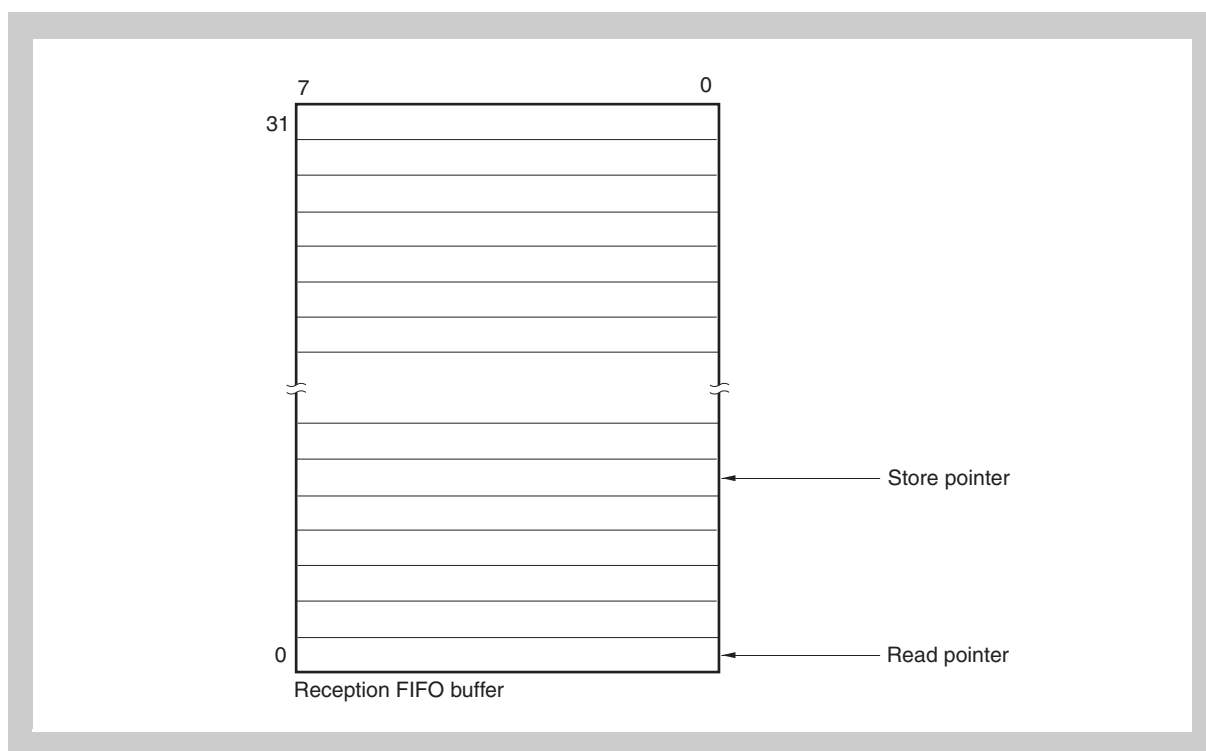


Figure 22-13 Reception FIFO buffer

## 22.5.2 Initial settings

After setting the IEBBnBCR.IEBBnPW bit to 1, set up the registers below, and then start communication processing.

Table 22-41 Initial setup

Register name	Function	Example
IEBBnPSR	Operation clock and communication mode settings	80H
IEBBnUAR	Set a unit address.	101H
IEBBnCKS	Clock Selection	15H
IEBBnTMS	Communication control	01H <sup>a</sup>

<sup>a)</sup> For use in the single mode, the initial values do not have to be changed.

### 22.5.3 Master transmission (single mode)

The unit transmits data and commands to the slave unit as the master.

Data interrupts are used to write transmission data to the IEBBnDR register for each one-byte transfer.

#### (1) Register settings

After specifying the initial settings in 22.5.2 "Initial settings", set up the registers below before starting communication.

Table 22-42 Standard initial processing

Register name	Function	Example
IEBBnSAR	Communication partner unit address	102H
IEBBnCDR or IEBBnTCD	Control data (AH, BH, EH, FH)	FH
IEBBnDLR	Message length	02H
IEBBnDR	Data (1st byte of data)	11H

Table 22-43 Communication startup processing

Register name	Function	Example
IEBBnBCR	Communication startup processing	C8H

#### (2) Interrupt occurrence timing

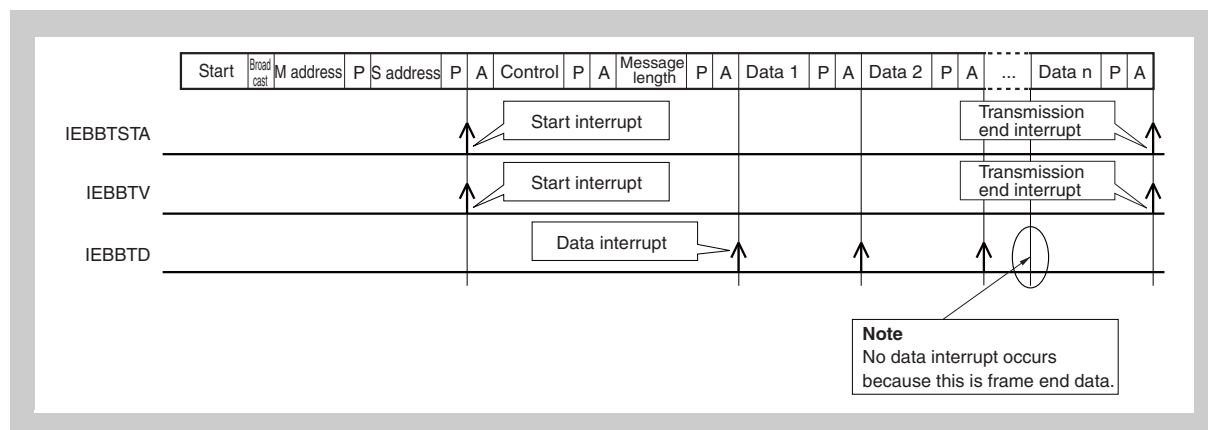
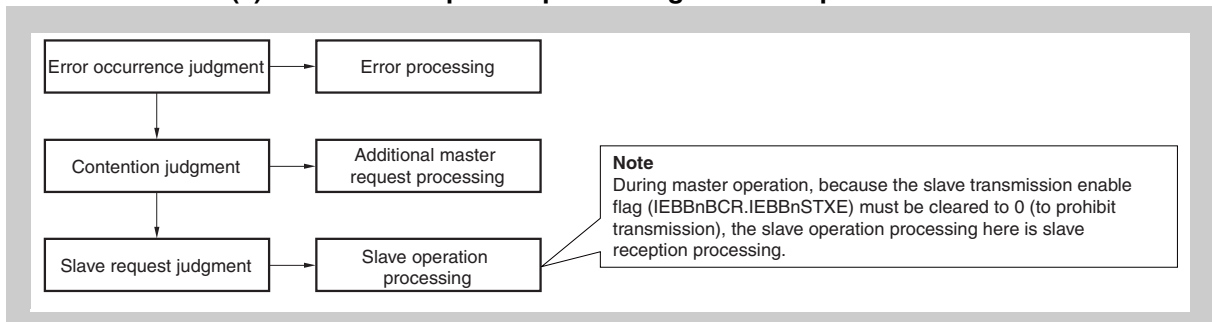


Figure 22-14 Interrupt occurrence timing

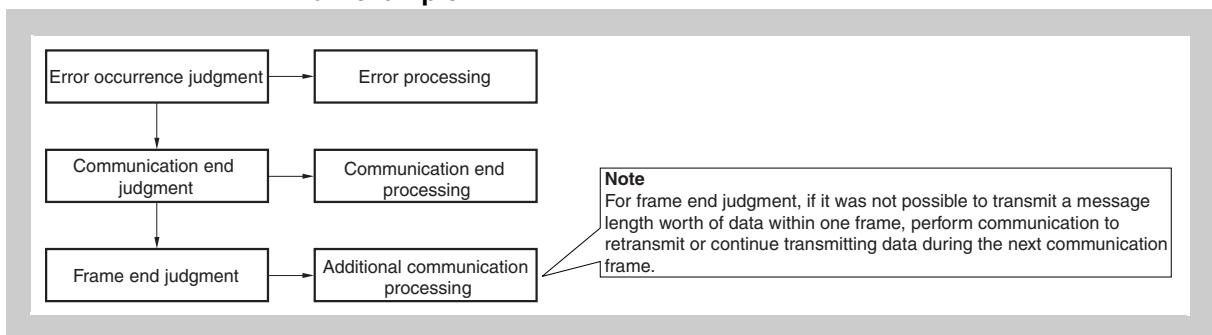
**(3) Interrupt servicing examples**

**(a) Start interrupt CPU processing flow example**



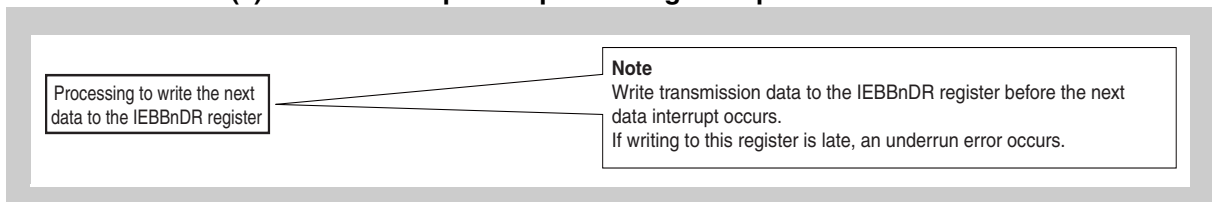
**Figure 22-15 Start interrupt CPU processing flow example**

**(b) Transmission completion interrupt (IEBBTV, IEBBTSTA) CPU processing flow example**



**Figure 22-16 Transmission completion interrupt (IEBBTV, IEBBTSTA) CPU processing flow example**

**(c) Data interrupt CPU processing example**



**Figure 22-17 Data interrupt CPU processing**

### 22.5.4 Master transmission (FIFO mode)

The unit transmits data and commands to the slave unit as the master.

Transmission data is written into the buffer in advance, and then a master request is issued.

#### (1) Register settings

After specifying the initial settings in 22.5.2 "Initial settings", set up the registers below before starting communication.

Table 22-44 Standard initial processing

Register name	Function	Example
IEBBnSAR	Communication partner unit address	102H
IEBBnCDR or IEBBnTCD	Control data (AH, BH, EH, FH)	FH
IEBBnTDL	Message length	02H
IEBBnDR	Data (all data to be transmitted)	11H, ...

Table 22-45 Communication startup processing

Register name	Function	Example
IEBBnBCR	Communication startup processing	C8H

#### (2) Interrupt occurrence timing

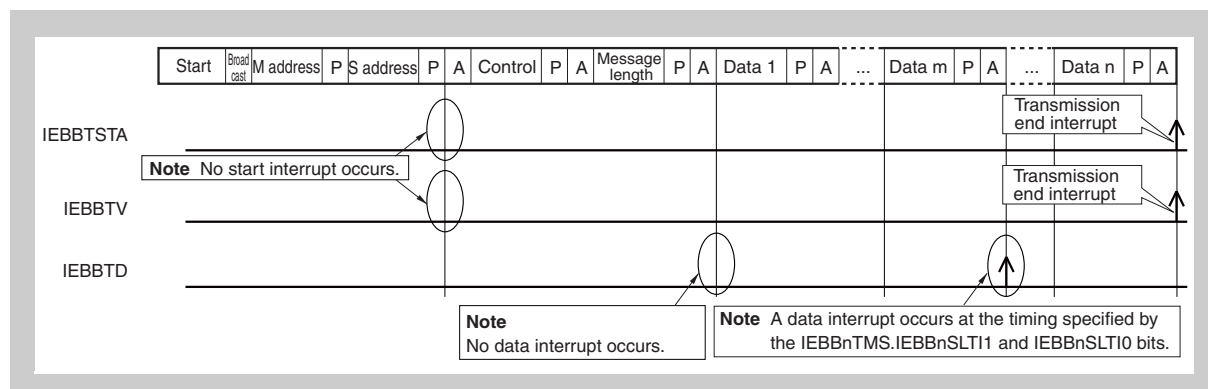
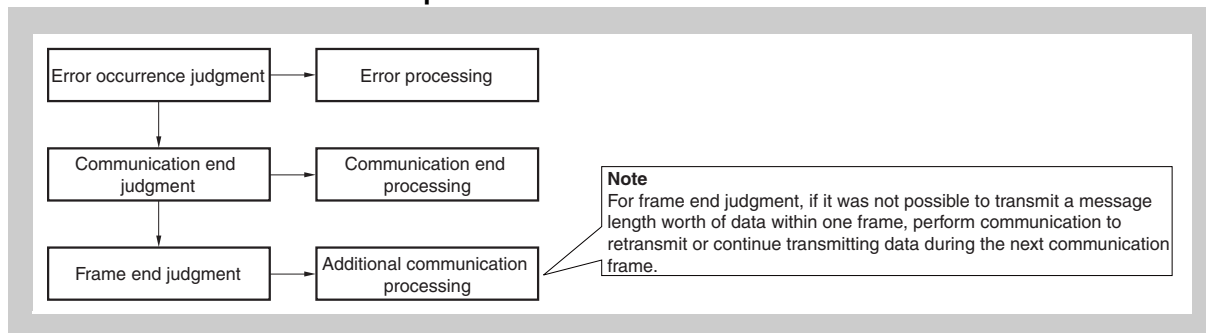


Figure 22-18 Interrupt occurrence timing

**(3) Interrupt servicing examples****(a) Transmission completion interrupt (IEBBTV, IEBBTSTA) CPU processing flow example**

**Figure 22-19** Transmission completion interrupt (IEBBTV, IEBBTSTA) CPU processing flow example



## 22.5.5 Master reception (single mode)

The unit receives data and commands from the slave unit as the master. Because the slave transfers the message length field in the case of master reception, indicate the message length of data to be transmitted to the slave, such as during other communication. Read the received data one byte at a time by using data interrupts.

### (1) Register settings

After specifying the initial settings in 22.5.2 "Initial settings", set up the registers below before starting communication.

Table 22-46 Standard initial processing

Register name	Function	Example
IEBBnSAR	Communication partner unit address	102H
IEBBnCDR or IEBBnTCD	Control data (0H, 3H, 4H, 5H, 6H, 7H)	7H

Table 22-47 Communication startup processing

Register name	Function	Example
IEBBnBCR	Communication startup processing	C8H

### (2) Interrupt occurrence timing

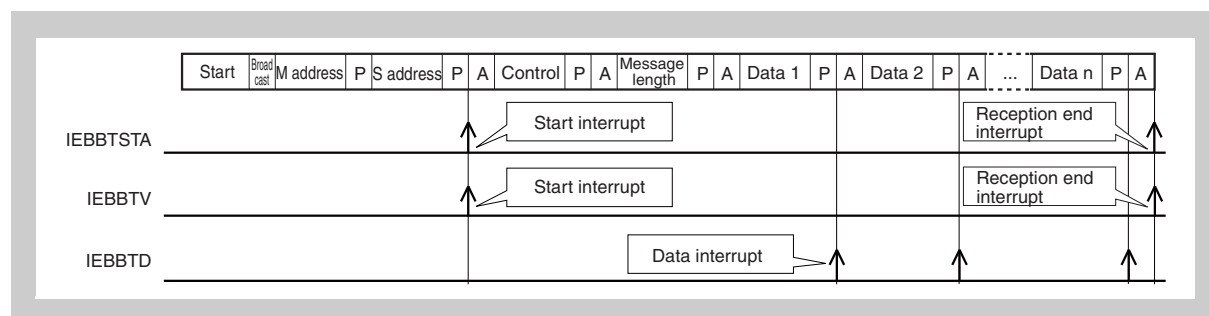


Figure 22-20 Interrupt occurrence timing

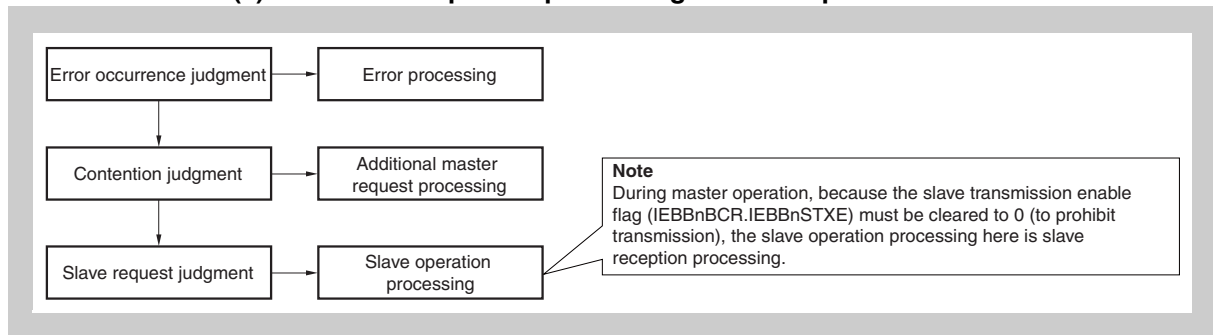
**(3) Interrupt servicing examples****(a) Start interrupt CPU processing flow example**

Figure 22-21 Start interrupt CPU processing flow example

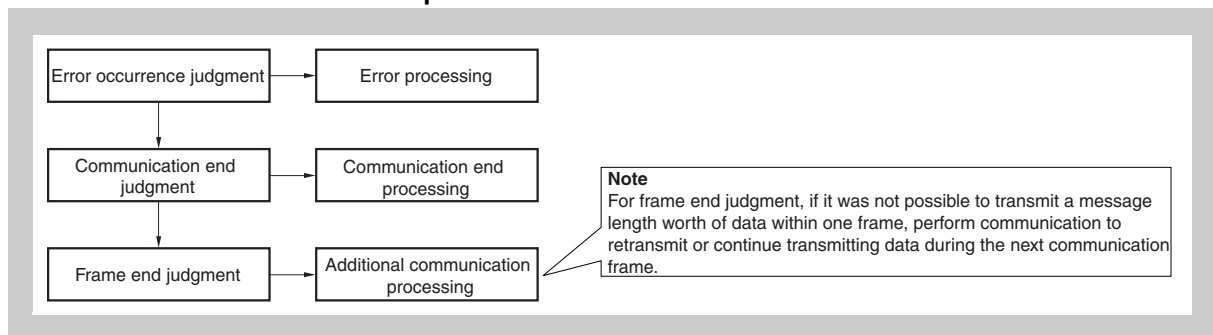
**(b) Reception completion interrupt (IEBBTV, IEBBTSTA) CPU processing flow example**

Figure 22-22 Reception completion interrupt (IEBBTV, IEBBTSTA) CPU processing flow example

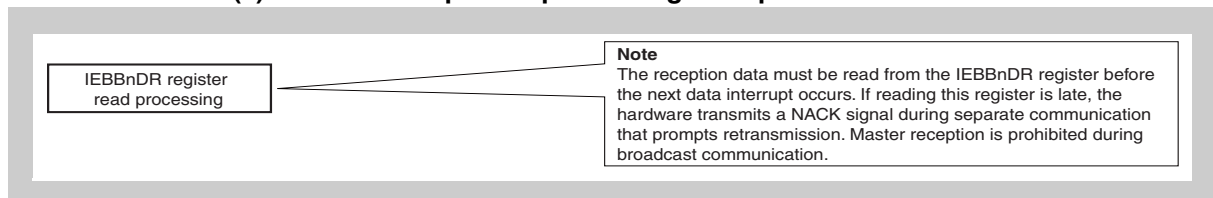
**(c) Data interrupt CPU processing example**

Figure 22-23 Data interrupt CPU processing

### 22.5.6 Master reception (FIFO mode)

The unit receives data and commands from the slave unit as the master. Because the slave transfers the message length field in the case of master reception, indicate the message length of data to be transmitted to the slave, such as during other communication.

#### (1) Register settings

After specifying the initial settings in 22.5.2 "Initial settings", set up the registers below before starting communication.

Table 22-48 Standard initial processing

Register name	Function	Example
IEBBnSAR	Communication partner unit address	102H
IEBBnCDR or IEBBnTCD	Control data (0H, 3H, 4H, 5H, 6H, 7H)	7H

Table 22-49 Communication startup processing

Register name	Function	Example
IEBBnBCR	Communication startup processing	C8H

#### (2) Interrupt occurrence timing

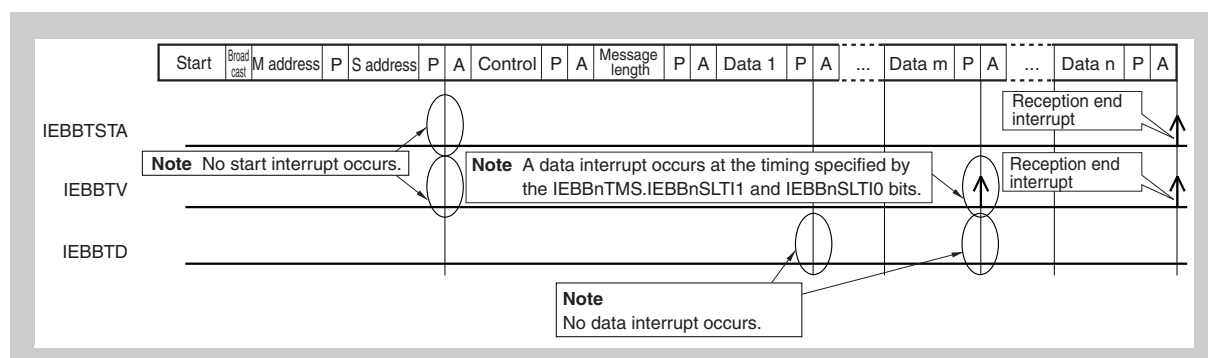
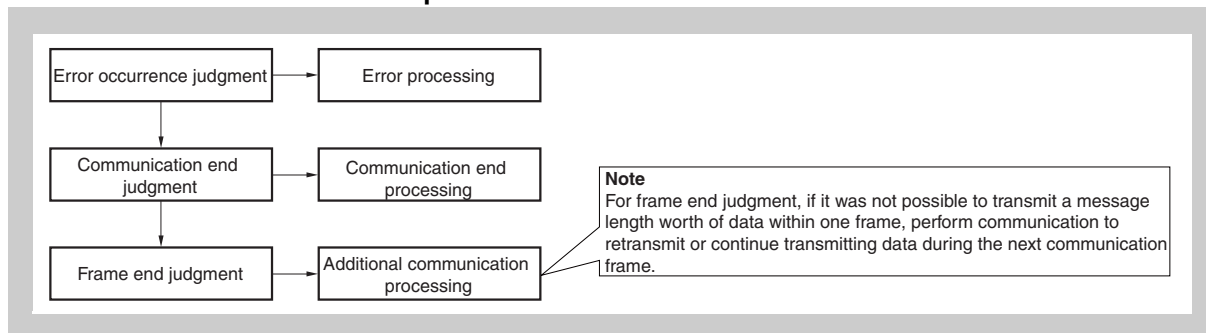


Figure 22-24 Interrupt occurrence timing

**(3) Interrupt servicing examples****(a) Reception completion interrupt (IEBBTV, IEBBTSTA) CPU processing flow example**

**Figure 22-25 Reception completion interrupt (IEBBTV, IEBBTSTA) CPU processing flow example**

### 22.5.7 Slave transmission (single mode)

The unit transfers data and commands to the master unit as a slave.

Data interrupts are used to write transmission data to the IEBBnDR register for each one-byte transfer.

#### (1) Register settings

After specifying the initial settings in 22.5.2 "Initial settings", set up the registers below before starting communication.

Table 22-50 Standard initial processing

Register name	Function	Example
IEBBnDLR	Message length (other than during slave status transmission)	02H
IEBBnDR	Data (1st byte of data)	11H

**Caution** When starting slave transmission, information such as the value to be set to the message length register (IEBBnDLR) and which data is to be returned (the value to be set to the IEBBnDR register) must be assigned in advance by the master, such as during separate communication.

Table 22-51 Communication startup processing

Register name	Function	Example
IEBBnBCR	Communication startup processing	90H

#### (2) Interrupt occurrence timing

##### (a) When the control bit 3H or 7H, which is addressed to the unit, is received

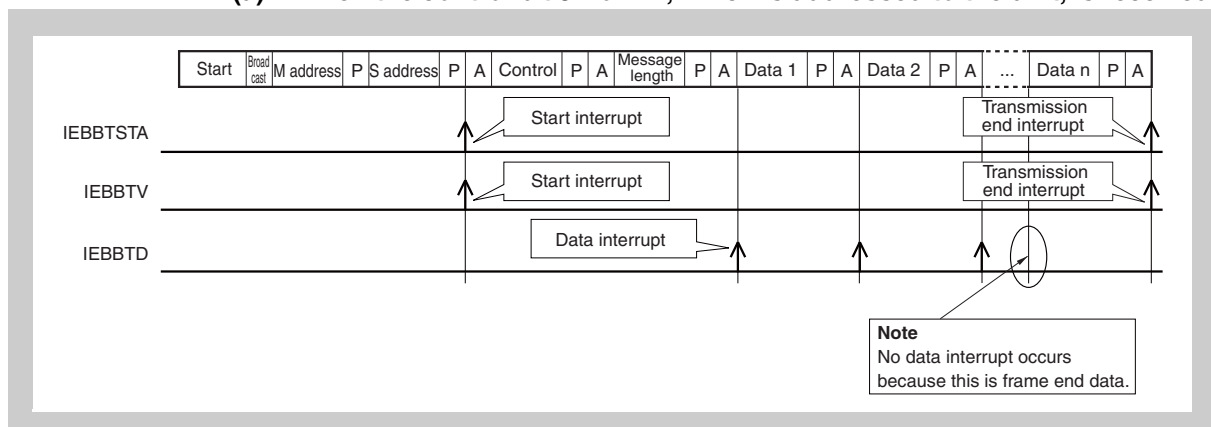
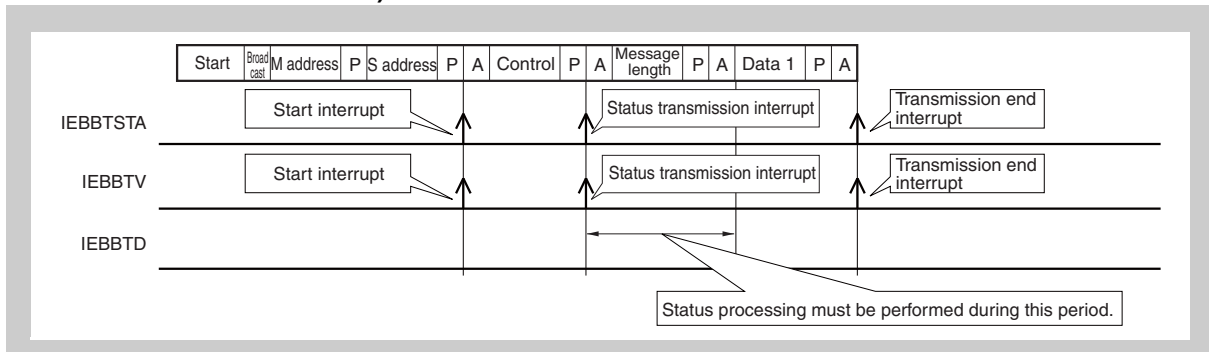


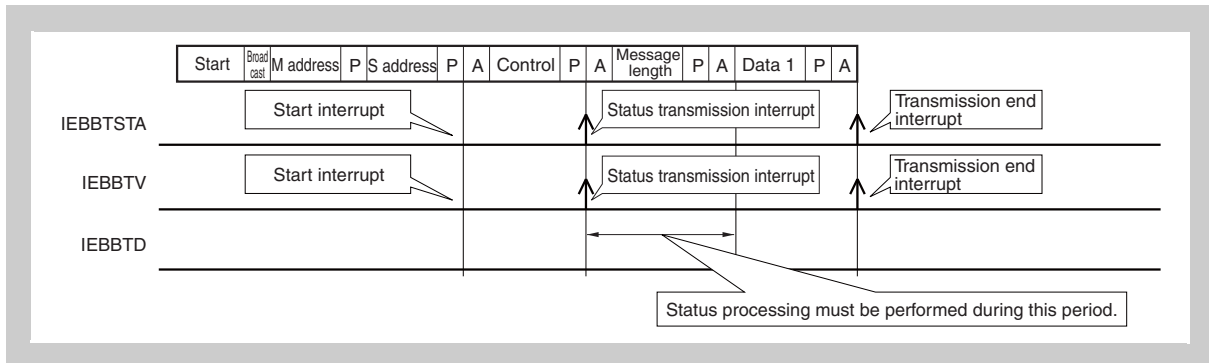
Figure 22-26 When the control bit 3H or 7H, which is addressed to the unit, is received

**(b) When the control bit 0H or 6H, which is addressed to the unit, is received (or when 4H or 5H is received from the locked master while the unit is locked)**



**Figure 22-27** When the control bit 0H or 6H, which is addressed to the unit, is received (or when 4H or 5H is received from the locked master while the unit is locked)

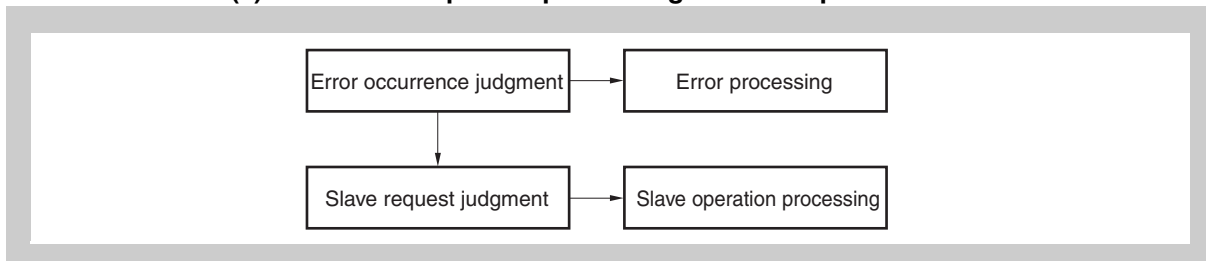
**(c) When the control bit 0H, 4H, or 5H, which is addressed to the unit, is received from a unit other than the locked master while the unit is locked**



**Figure 22-28** When the control bit 0H, 4H, or 5H, which is addressed to the unit, is received from a unit other than the locked master while the unit is locked

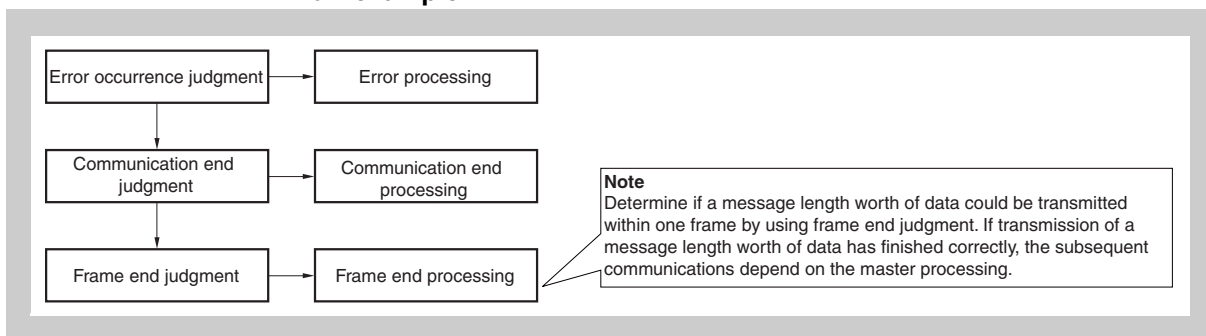
**(3) Interrupt servicing examples**

**(a) Start interrupt CPU processing flow example**



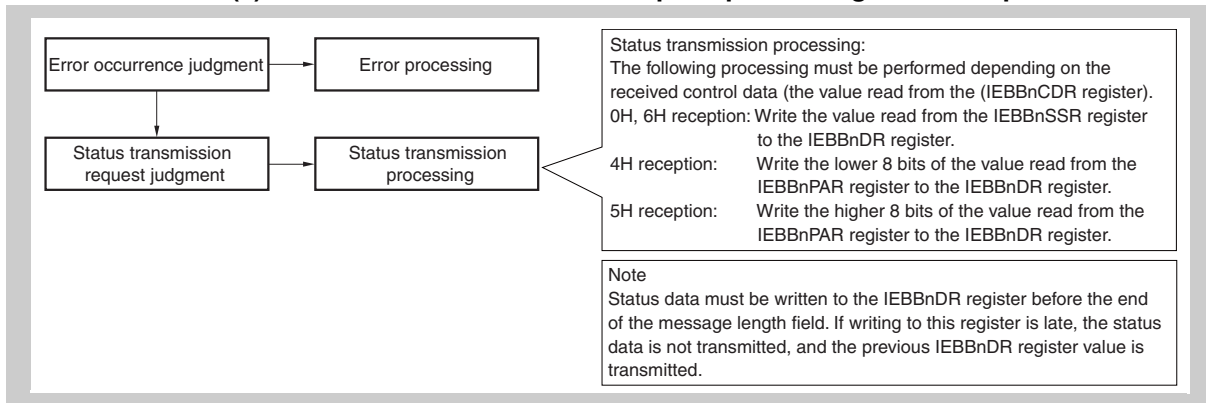
**Figure 22-29 Start interrupt CPU processing flow example**

**(b) Transmission completion interrupt (IEBBTV, IEBBTSTA) CPU processing flow example**



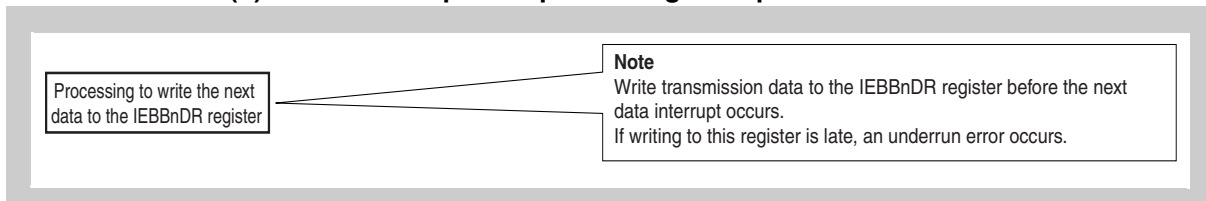
**Figure 22-30 Transmission completion interrupt (IEBBTV, IEBBTSTA) CPU processing flow example**

**(c) Save status transmission request processing flow example**



**Figure 22-31 Slave status transmission request processing flow**

**(d) Data interrupt CPU processing example**



**Figure 22-32 Data interrupt CPU processing**

### 22.5.8 Slave transmission (FIFO mode)

The unit transfers data and commands to the master unit as a slave.

Write the transmission data to the buffer in advance.

#### (1) Register settings

After specifying the initial settings in 22.5.2 "Initial settings", set up the registers below before starting communication.

Table 22-52 Standard initial processing

Register name	Function	Example
IEBBnTDL	Message length (other than during slave status transmission)	02H
IEBBnDR	Data (all data to be transmitted)	11H, ...

**Caution** When starting slave transmission, information such as the value to be set to the message length register (IEBBnTDL) and which data is to be returned (the value to be set to the IEBBnDR register) must be assigned in advance by the master, such as during separate communication.

Table 22-53 Communication startup processing

Register name	Function	Example
IEBBnBCR	Communication startup processing	90H

#### (2) Interrupt occurrence timing

##### (a) When the control bit 3H or 7H, which is addressed to the unit, is received

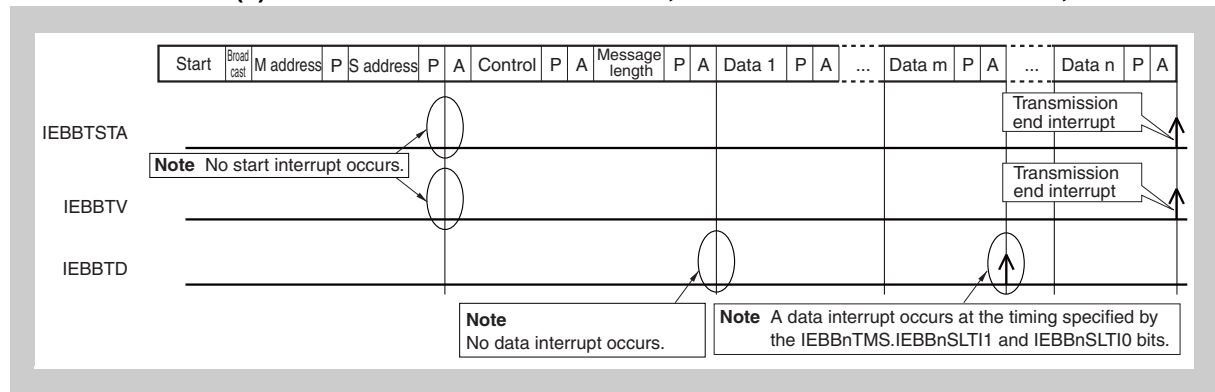
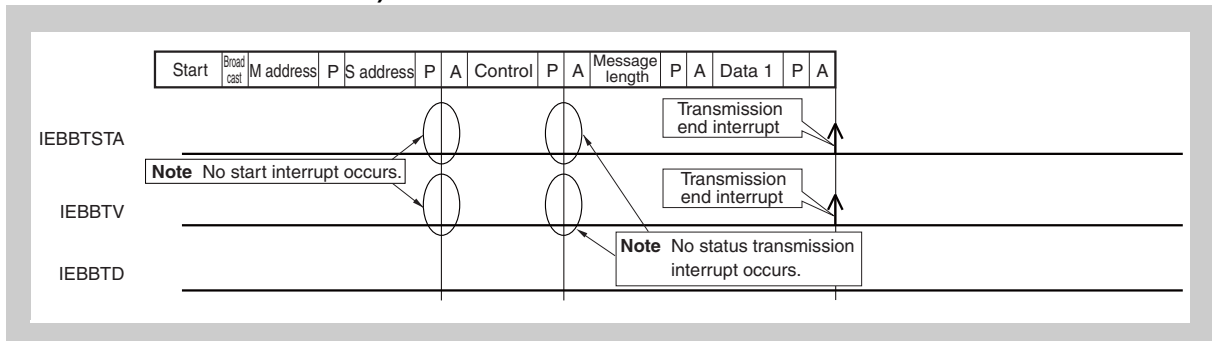


Figure 22-33 When the control bit 3H or 7H, which is addressed to the unit, is received

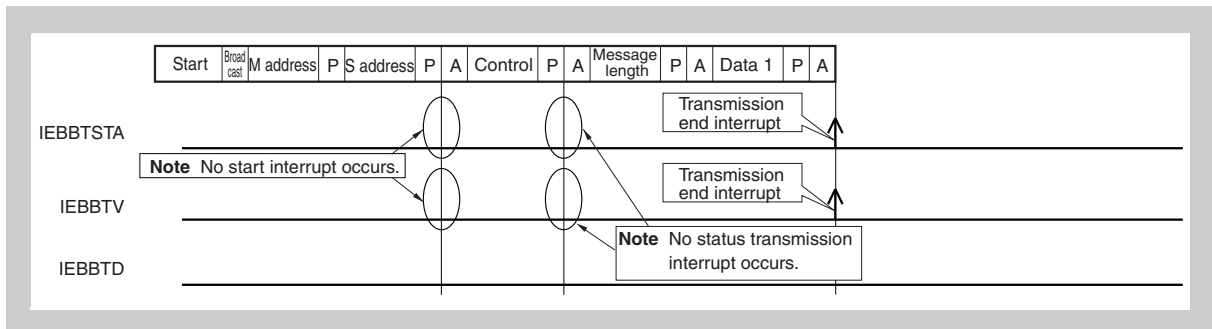


**(b) When the control bit 0H or 6H, which is addressed to the unit, is received (or when 4H or 5H is received from the locked master while the unit is locked)**

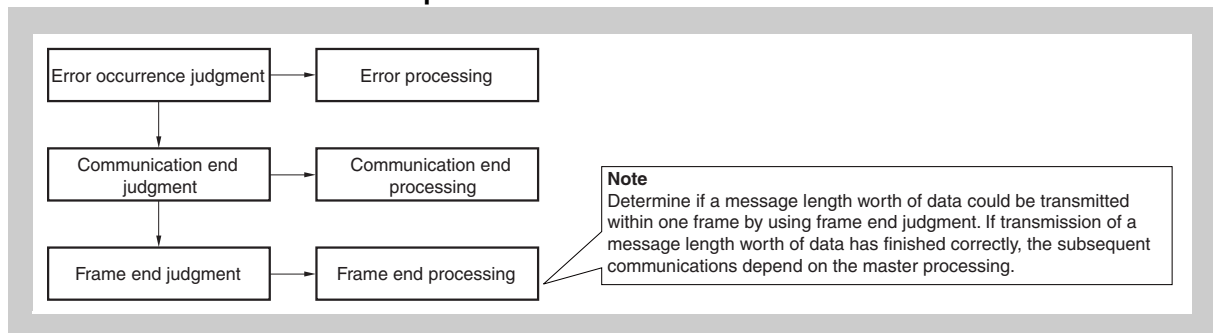
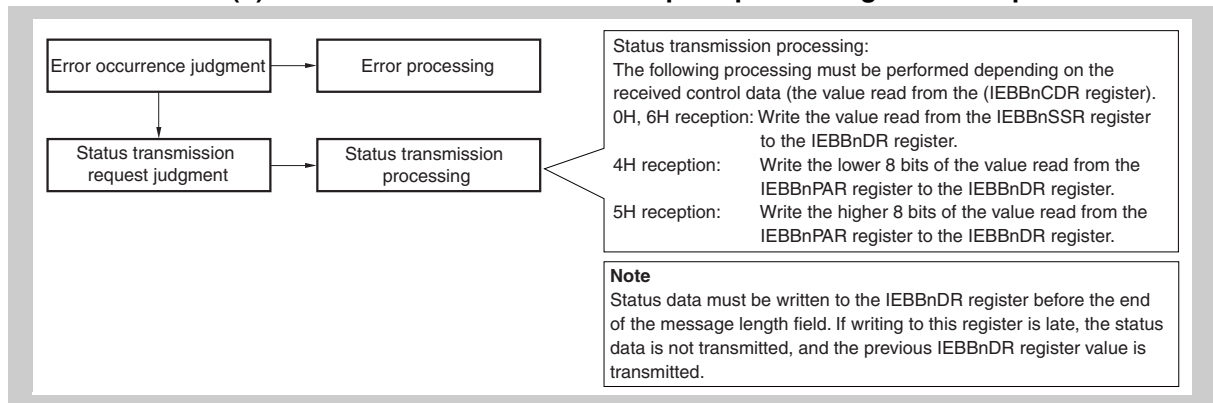


**Figure 22-34** When the control bit 0H or 6H, which is addressed to the unit, is received (or when 4H or 5H is received from the locked master while the unit is locked)

**(c) When the control bit 0H, 4H, or 5H, which is addressed to the unit, is received from a unit other than the locked master while the unit is locked**



**Figure 22-35** When the control bit 0H, 4H, or 5H, which is addressed to the unit, is received from a unit other than the locked master while the unit is locked

**(3) Interrupt servicing examples****(a) Transmission completion interrupt (IEBBTV, IEBBTSTA) CPU processing flow example****Figure 22-36** Transmission completion interrupt (IEBBTV, IEBBTSTA) CPU processing flow example**(b) Slave status transmission request processing flow example****Figure 22-37** Slave status transmission request processing flow

In the FIFO mode, if the unit loses a conflict between a unit master transmission request and a slave transmission request addressed to the unit, because the slave transmission enable flag (IEBBnBCR.IEBBnSTXE) is not set (to 1) for the unit, a NACK signal is transmitted using the control data field and communication ends. Next, specify the slave transmission data for the FIFO buffer, set the slave transmission enable flag (to 1), and prepare to receive another slave transmission request from the master.

### 22.5.9 Slave reception (single mode)

The unit receives data and commands from the master unit as a slave.  
 Read the received data one byte at a time by using data interrupts.

#### (1) Register settings

After specifying the initial settings in 22.5.2 "Initial settings", set up the registers below before starting communication.

Table 22-54 Communication startup processing

Register name	Function	Example
IEBBnBCR	Communication startup processing	88H

#### (2) Interrupt occurrence timing

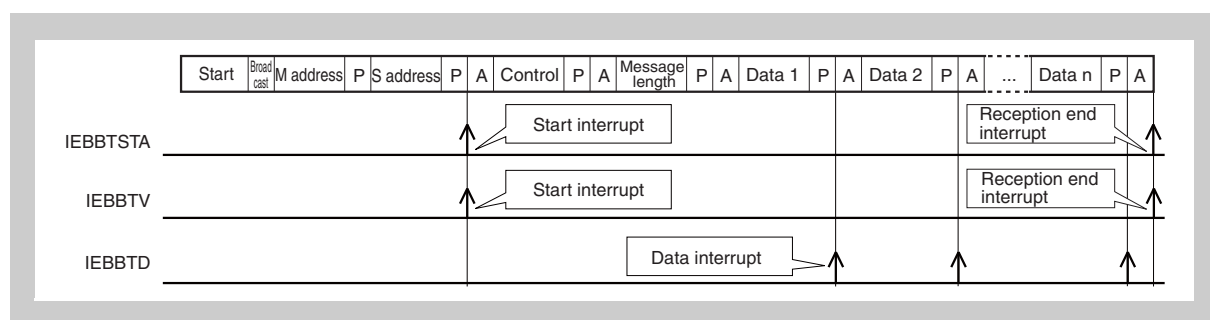
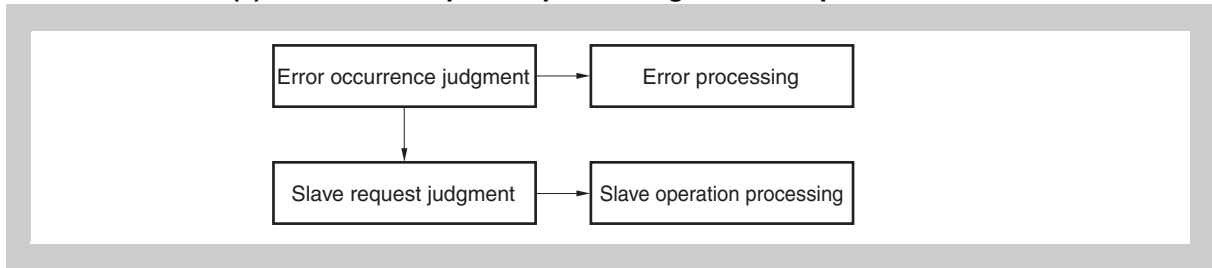


Figure 22-38 Interrupt occurrence timing

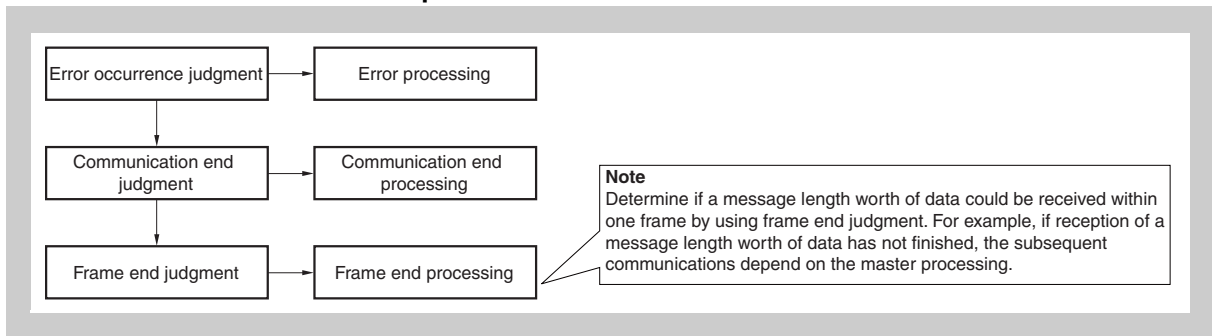
**(3) Interrupt servicing examples**

**(a) Start interrupt CPU processing flow example**



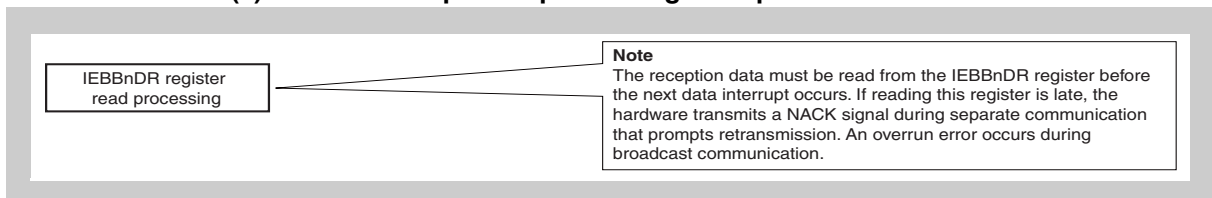
**Figure 22-39 Start interrupt CPU processing flow example**

**(b) Reception completion interrupt (IEBBTV, IEBBTSTA) CPU processing flow example**



**Figure 22-40 Reception completion interrupt (IEBBTV, IEBBTSTA) CPU processing flow example**

**(c) Data interrupt CPU processing example**



**Figure 22-41 Data interrupt CPU processing**

### 22.5.10 Slave reception (FIFO mode)

The unit receives data and commands from the master unit as a slave.

#### (1) Register settings

After specifying the initial settings in 22.5.2 "Initial settings", set up the registers below before starting communication.

Table 22-55 Communication startup processing

Register name	Function	Example
IEBBnBCR	Communication startup processing	88H

#### (2) Interrupt occurrence timing

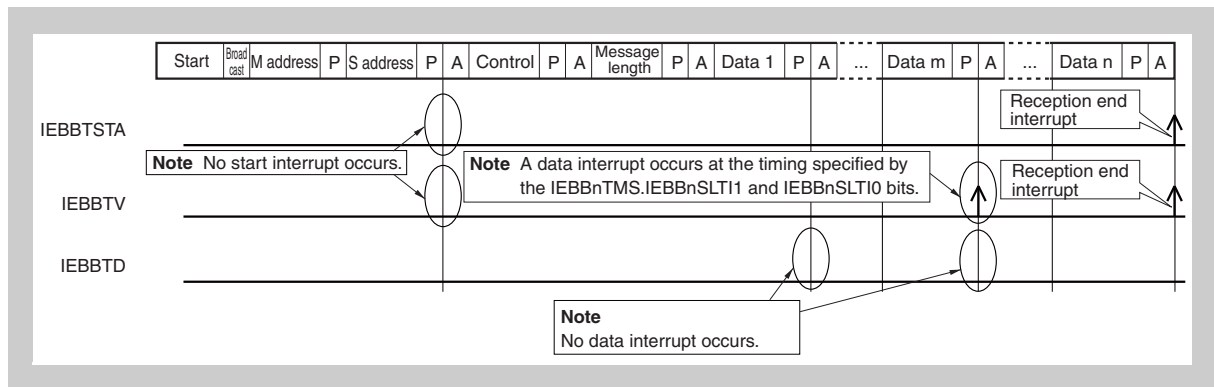


Figure 22-42 Interrupt occurrence timing

#### (3) Interrupt servicing examples

##### (a) Reception completion interrupt (IEBBTV, IEBBTSTA) CPU processing flow example

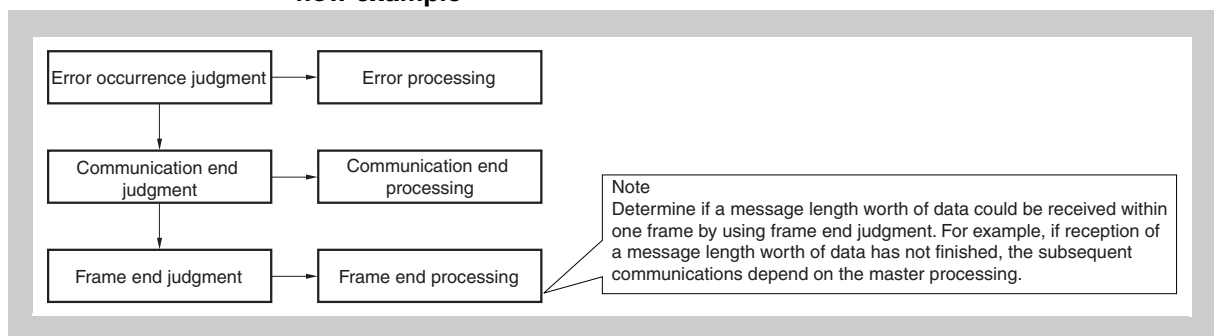


Figure 22-43 Reception completion interrupt (IEBBTV, IEBBTSTA) CPU processing flow example

## 22.6 Setup Procedures

### 22.6.1 Master transmission (single mode)

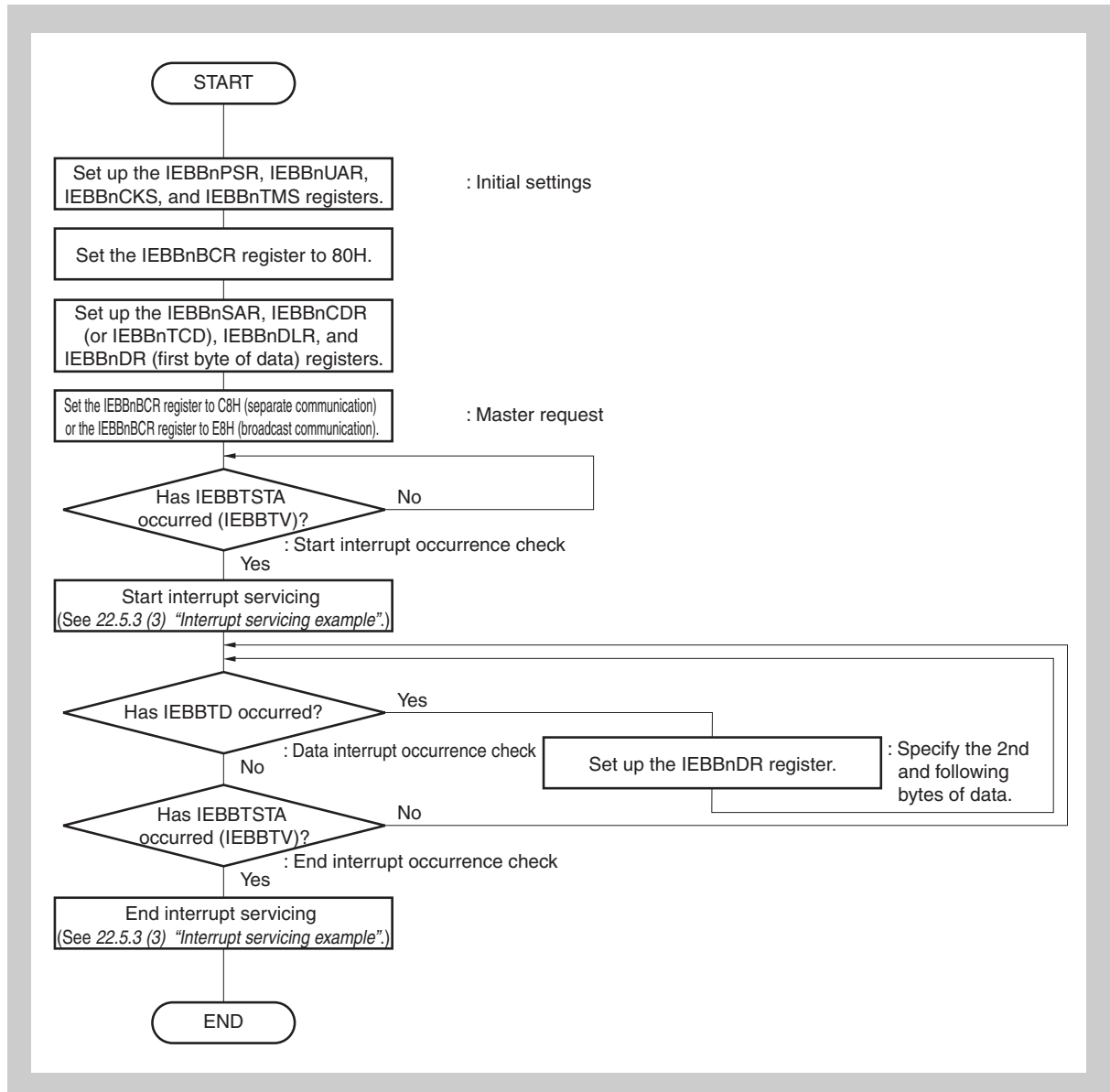


Figure 22-44 Master transmission (single mode)

### 22.6.2 Master transmission (FIFO mode)

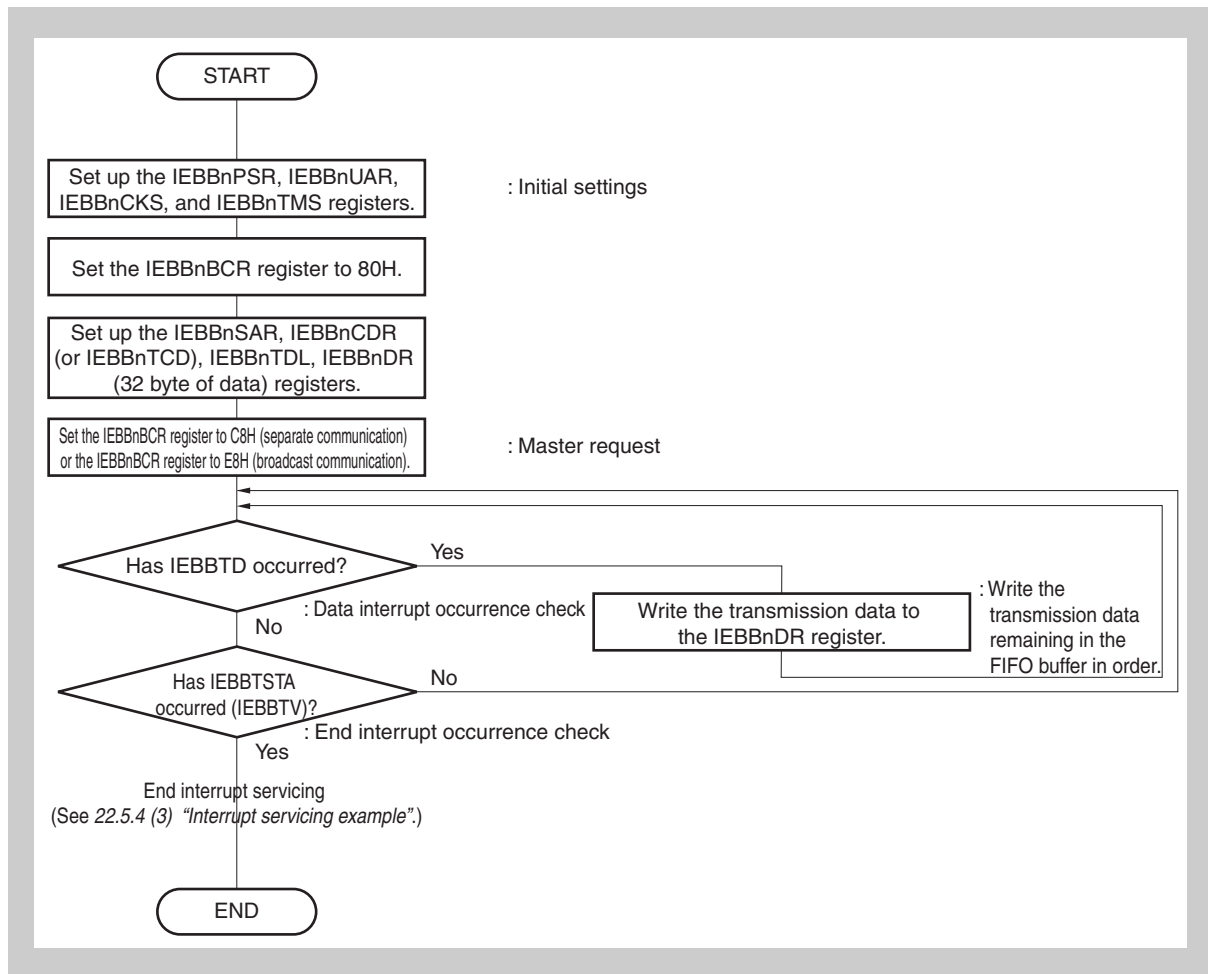


Figure 22-45 Master transmission (FIFO mode)

### 22.6.3 Master reception (single mode)

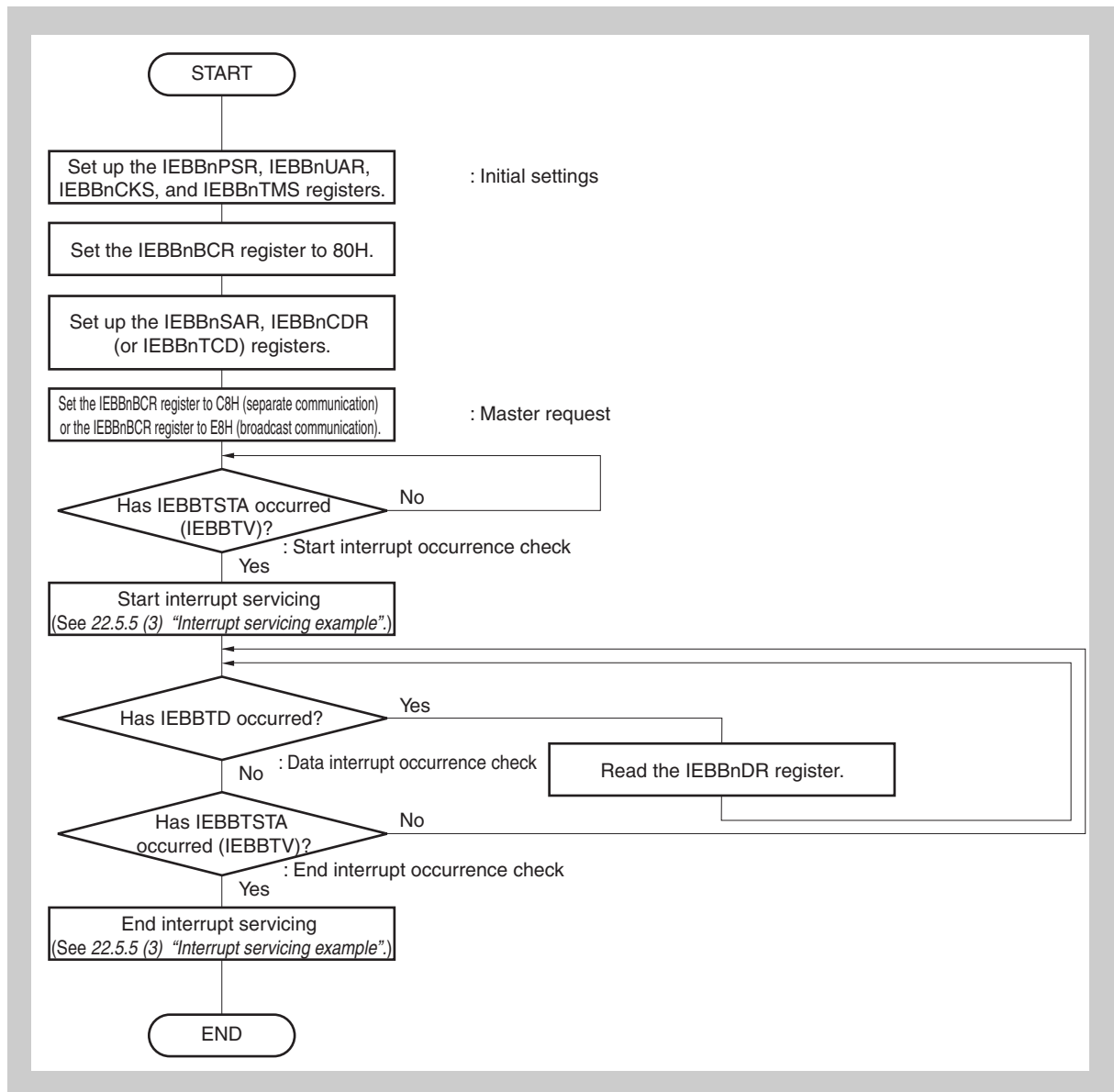


Figure 22-46 Master reception (single mode)



## 22.6.4 Master reception (FIFO mode)

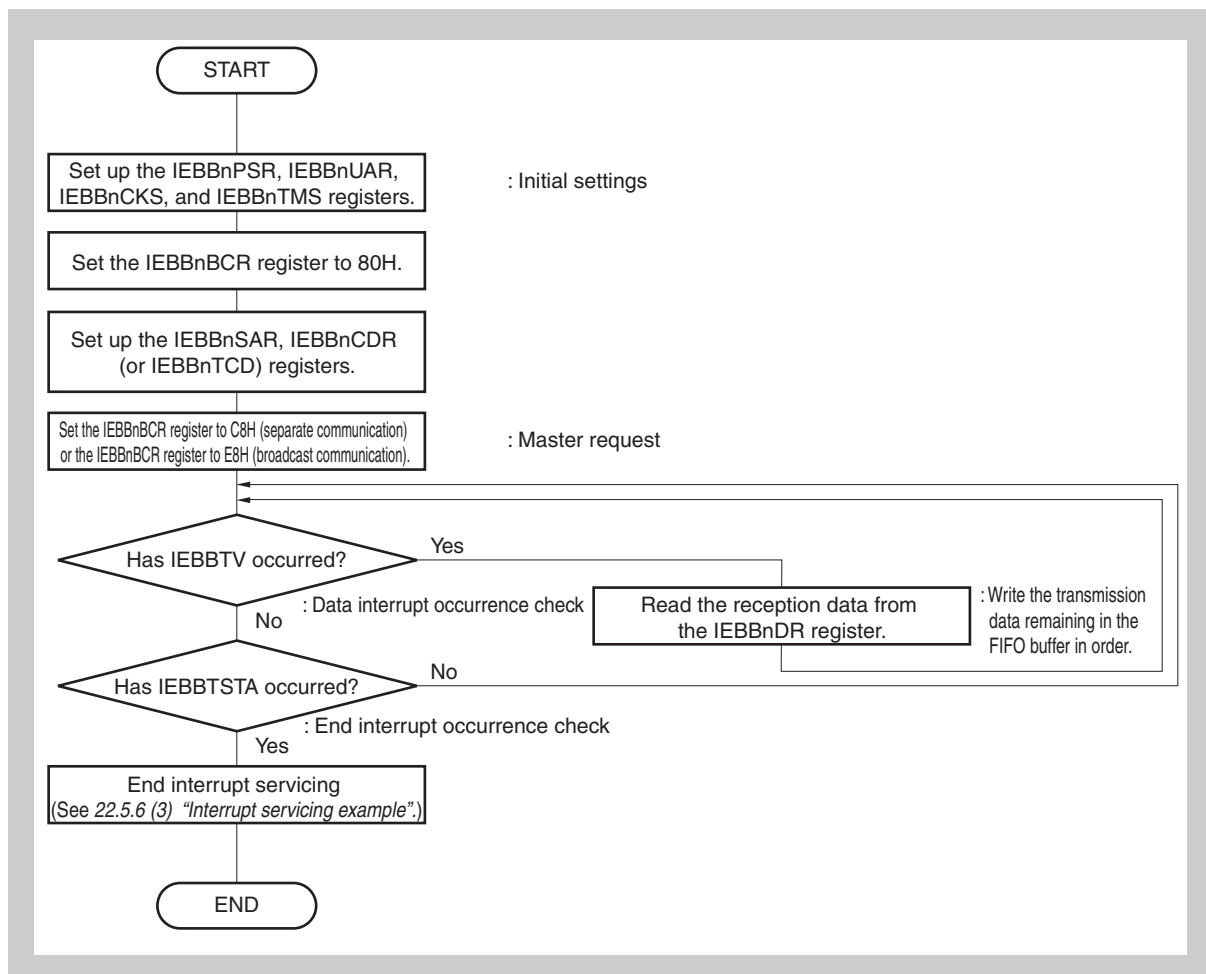


Figure 22-47 Master reception (FIFO mode)

## 22.6.5 Slave transmission (single mode)

### (1) When the control bit 3H or 7H is received

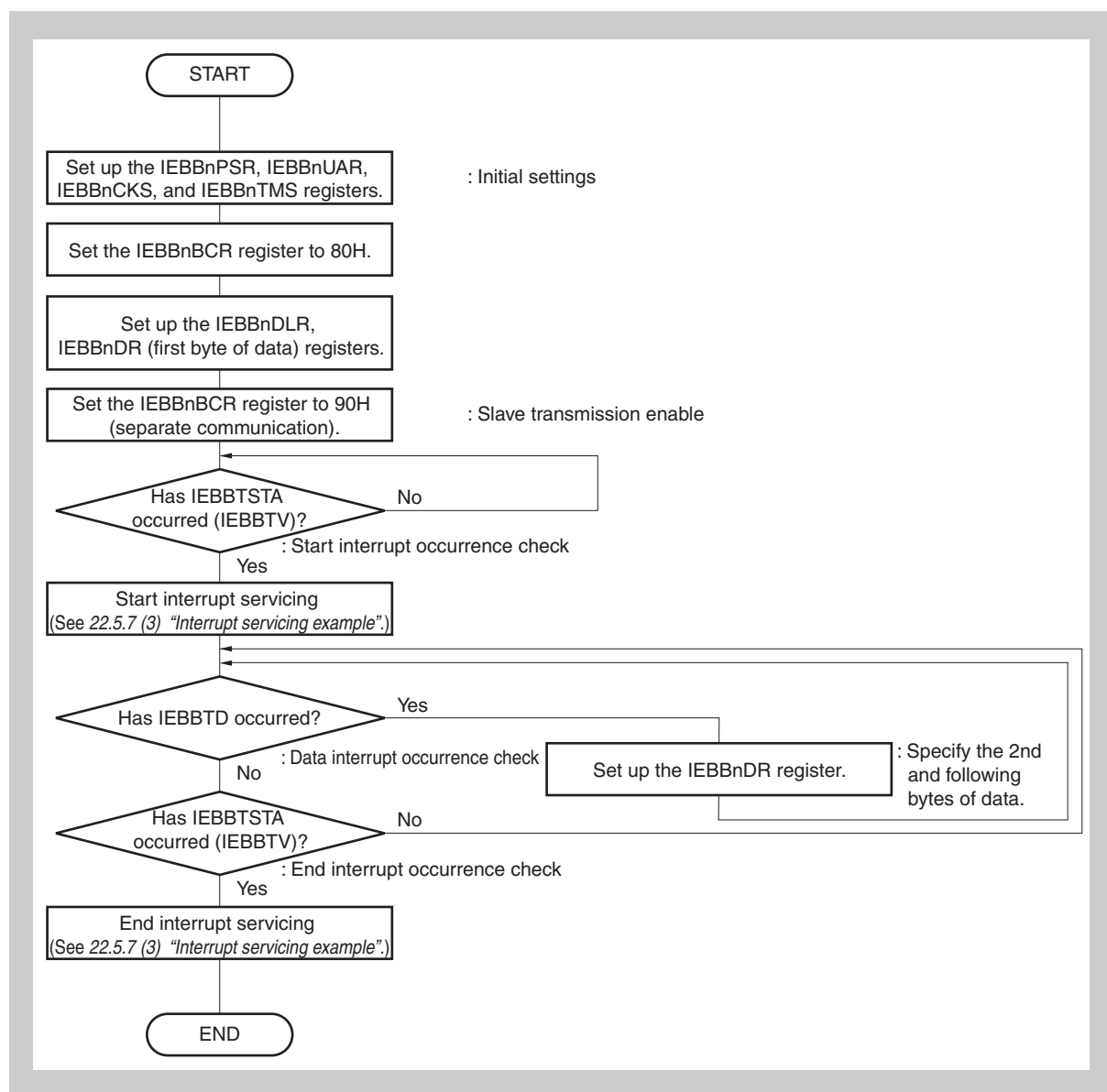


Figure 22-48 Slave transmission (single mode): When the control bit 3H or 7H is received

(2) When the control bit 0H or 6H is received (or when 4H or 5H is received from the locked master while the unit is locked)

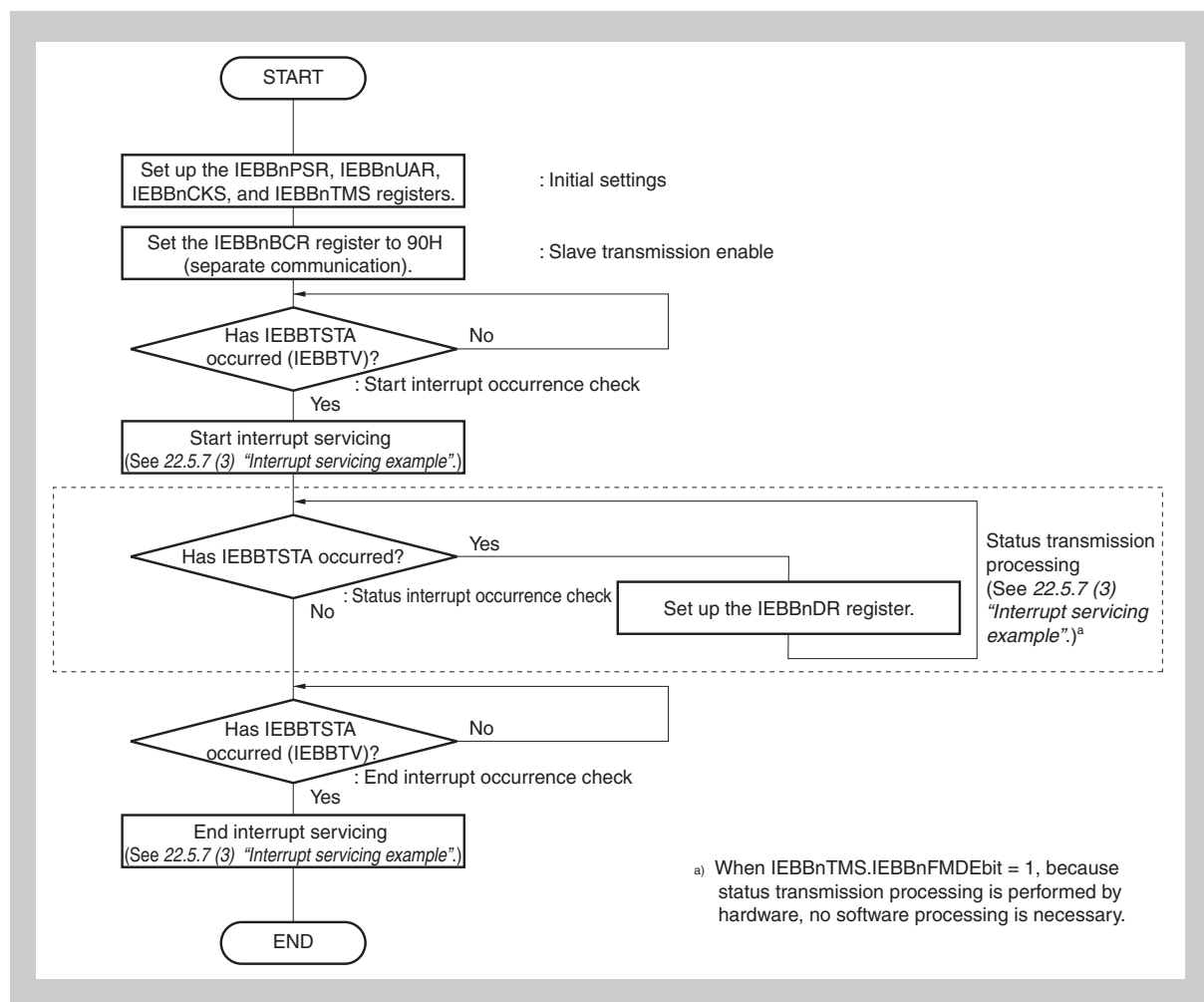
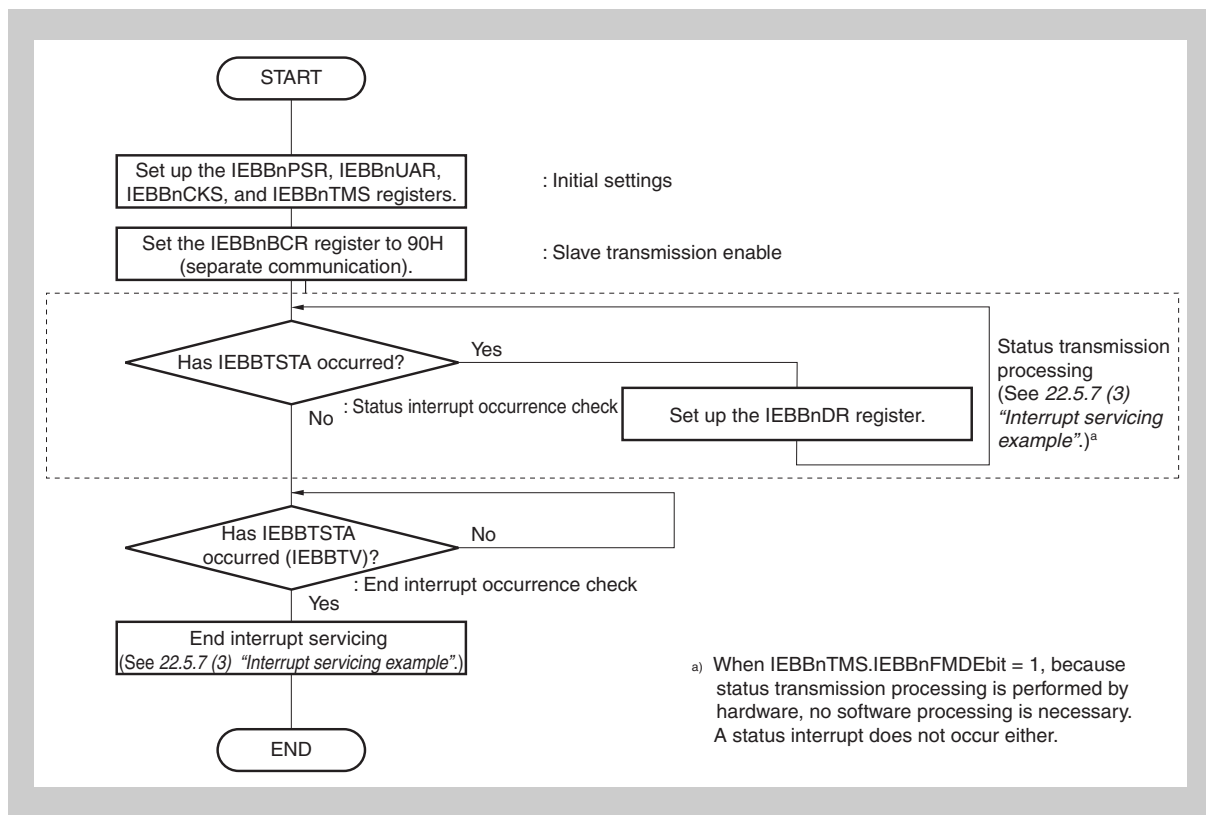


Figure 22-49 Slave transmission (single mode): When the control bit 0H or 6H is received (or when 4H or 5H is received from the locked master while the unit is locked)

**(3) When the control bit 0H, 4H, or 5H, which is addressed to the unit, is received from a unit other than the locked master while the unit is locked**



**Figure 22-50 Slave transmission (single mode): When the control bit 0H, 4H, or 5H, which is addressed to the unit, is received from a unit other than the locked master while the unit is locked**

## 22.6.6 Slave transmission (FIFO mode)

### (1) When the control bit 3H or 7H is received

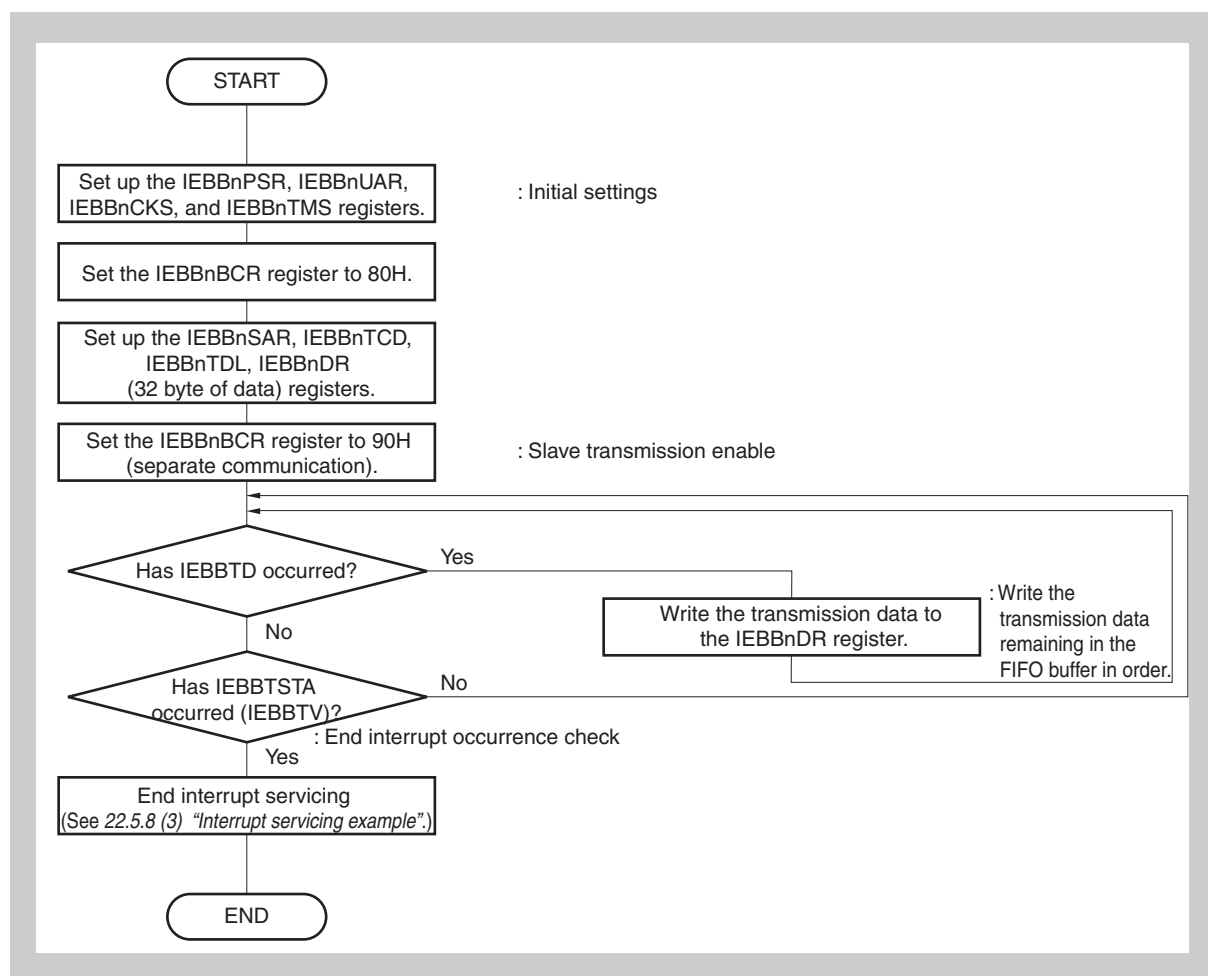
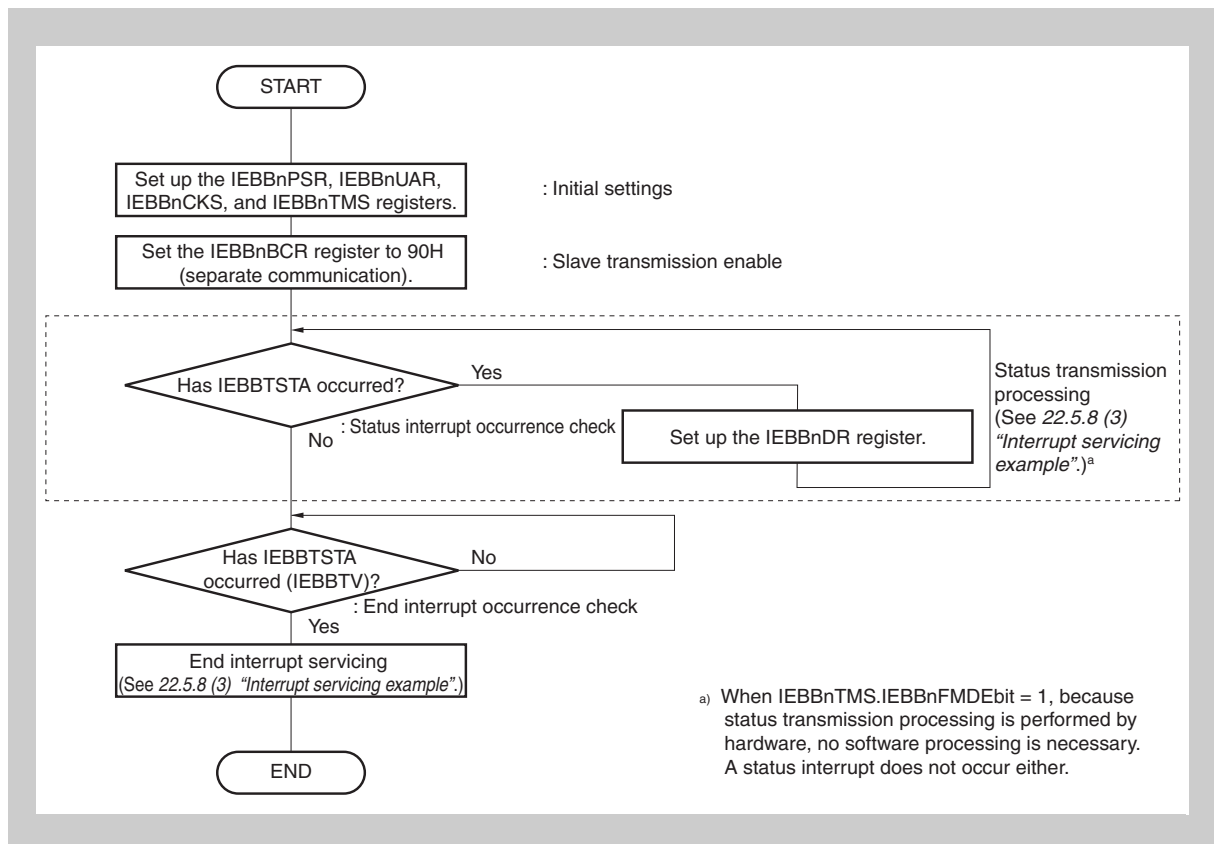


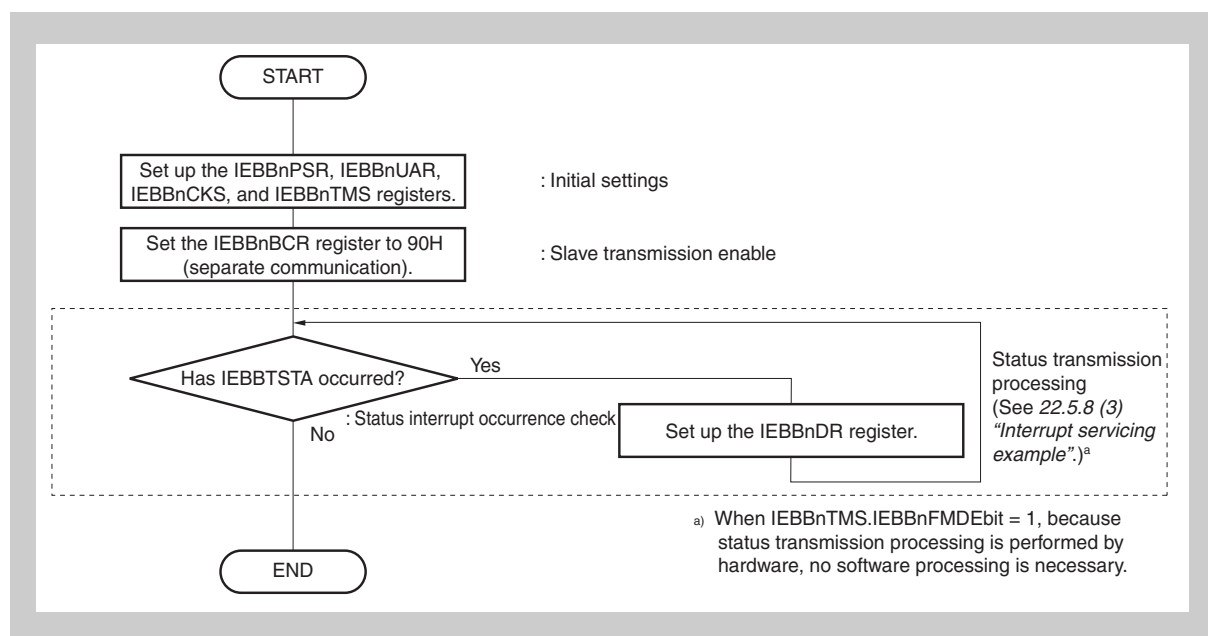
Figure 22-51 Slave transmission (FIFO mode): When the control bit 3H or 7H is received

**(2) When the control bit 0H or 6H is received (or when 4H or 5H is received from the locked master while the unit is locked)**



**Figure 22-52 Slave transmission (FIFO mode): When the control bit 0H or 6H is received (or when 4H or 5H is received from the locked master while the unit is locked)**

**(3) When the control bit 0H, 4H, or 5H, which is addressed to the unit, is received from a unit other than the locked master while the unit is locked**



**Figure 22-53 Slave transmission (FIFO mode): When the control bit 0H, 4H, or 5H, which is addressed to the unit, is received from a unit other than the locked master while the unit is locked**

### 22.6.7 Slave reception (single mode)

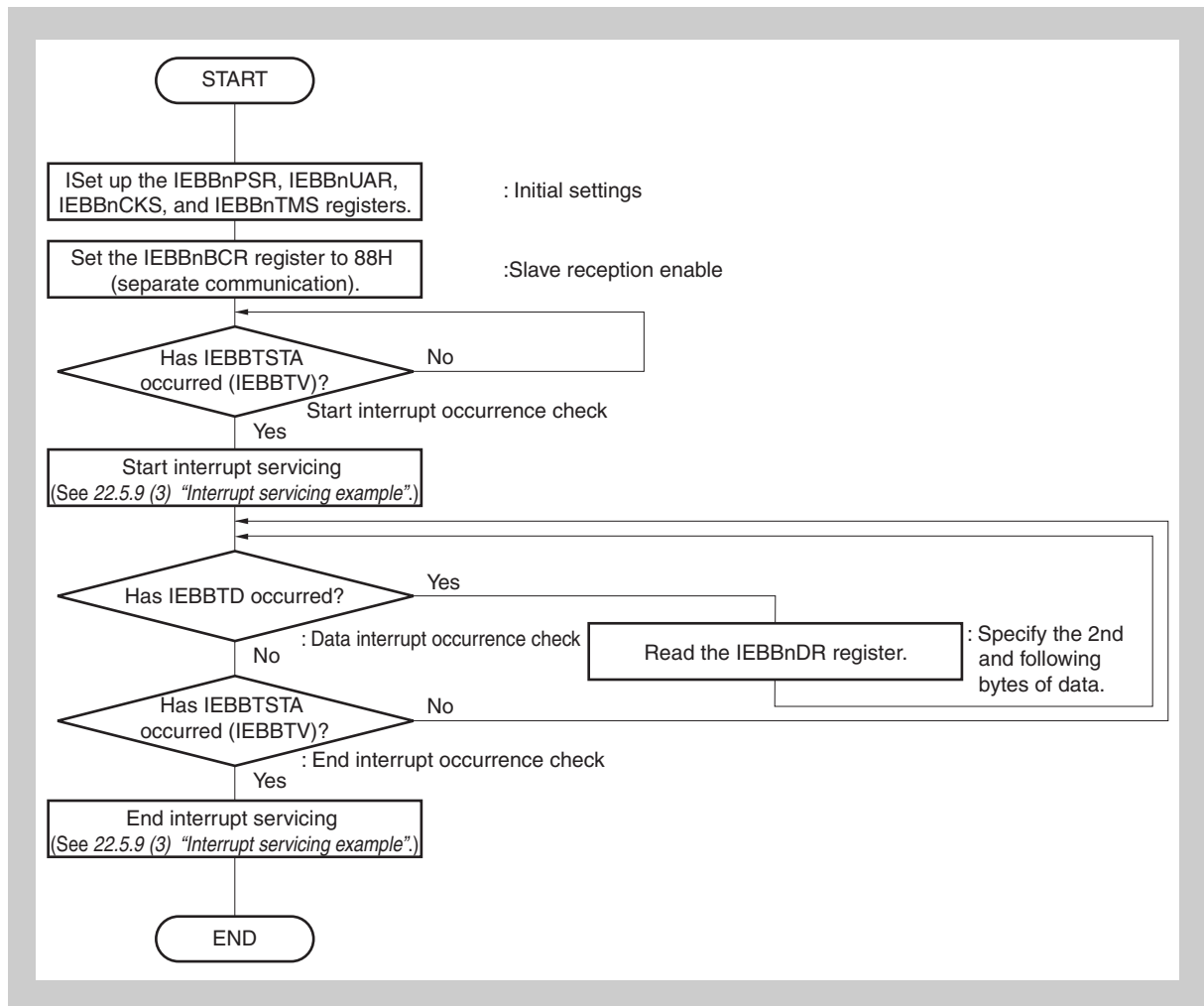


Figure 22-54 Slave reception (single mode)



### 22.6.8 Slave reception (FIFO mode)

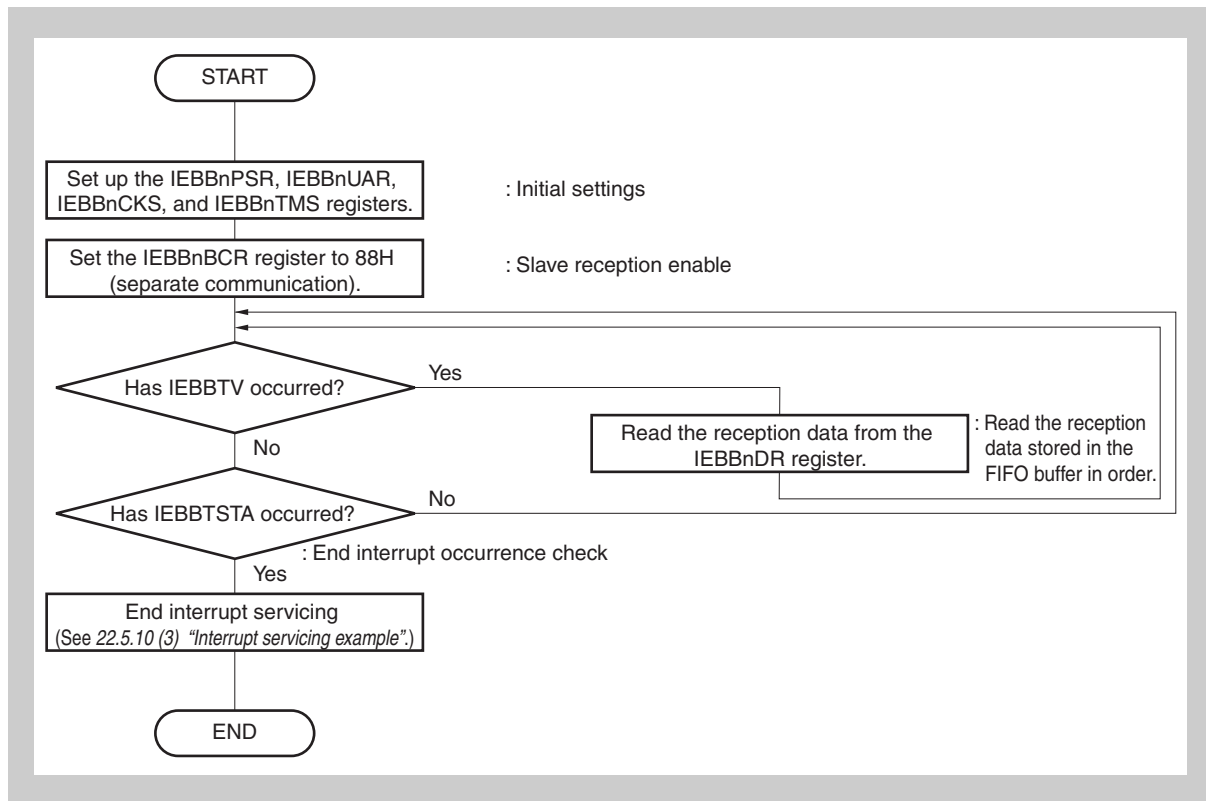


Figure 22-55 Slave reception (FIFO mode)

---

## 22.7 Functions

### 22.7.1 IEBus communication protocol

The communication protocol of the IEBus is as follows.

**(1) Multi-task mode**

All the units connected to the IEBus can transfer data to the other units.

**(2) Broadcast communication**

Communication between one unit and multiple units can be performed as follows.

- Group broadcast communication: Broadcast communication to group units
- All-unit broadcast communication: Broadcast communication to all units

**(3) Effective transmission speed**

The effective transfer rate is in communication mode 1 or communication mode 2. (The V850E2/Sx4-H does not support mode 0 for the effective transfer rate.)

- Communication mode 1: Approx. 17 kbps
- Communication mode 2: Approx. 26 kbps

---

**Caution** Different modes (communication mode 1, communication mode 2) must not be mixed on one IEBus.

---

**(4) Communication mode**

Data transfer is executed in half-duplex asynchronous communication mode.

**(5) Access control: CSMA/CD (Carrier Sense Multiple Access with Collision Detection)**

The priority of the IEBus is as follows:

1. Broadcast communication takes precedence over individual communication (communication from one unit to another).
2. The lower master address takes precedence.

**(6) Communication scale**

The communication scale of IEBus is as follows:

- Number of units: 50 maximum
- Cable length: 150 m maximum (when twisted pair cable is used)

---

**Caution** The communication scale in an actual system differs depending on the characteristics of the cables, etc., constituting the IEBus driver/receiver and IEBus.

---

### 22.7.2 Determination of bus mastership (arbitration)

An operation to occupy the bus is performed when a unit connected to the IEBus controls the other units. This operation is called arbitration.

When multiple units simultaneously start transmission, arbitration is used to grant one of the units permission to occupy the bus.

Because only one unit is granted bus mastership as a result of arbitration, the priority conditions of the bus are predetermined as follows.

---

**Caution** Bus mastership is canceled if communication is aborted.

---

#### (1) Priority by communication type

Broadcast communication (communication from one unit to multiple units) takes precedence over normal communication (communication from one unit to another).

#### (2) Priority by master address

If the communication type is the same, communication with the lower master address takes precedence.

A master address consists of 12 bits, with unit 000H having the highest priority and unit FFFH having the lowest priority.

### 22.7.3 Communication mode

The IEBus has three communication modes, each of which has a different transfer rate. This module supports communication modes 1 and 2. The transfer rate and the maximum number of transfer bytes per communication frame in communication modes 1 and 2 are shown below.

**Table 22-56 Transfer rate and maximum number of transfer bytes in each communication mode**

Communication mode	Maximum number of transfer bytes (bytes/frame)	Effective transfer rate <sup>a</sup>
1	32 bytes/frame	Approx. 17 kbps
2	128 bytes/frame	Approx. 26 kbps

<sup>a)</sup> Effective transfer rate when the maximum number of transfer bytes is transmitted

Select the communication mode for each unit connected to the IEBus before starting communication. If the communication mode of the master unit and that of the partner unit (slave unit) are not the same, communication is not correctly executed.

### 22.7.4 Communication address

For the IEBus, each unit is assigned a specific 12-bit address. This communication address consists of the following identification numbers.

- Higher 4 bits: Group number (number to identify the group to which each unit belongs)
- Lower 8 bits: Unit number (number to identify each unit in a group)

### 22.7.5 Broadcast communication

Normally, transmission or reception is performed between the master unit and its partner slave unit on a one-to-one basis. During broadcast communication, however, multiple slave units exist and the master unit executes transmission to these slave units. Because multiple slave units exist, the NACK signal is returned by the communicating slave unit as an acknowledge bit.

Whether broadcast communication or normal communication is to be executed is selected by the broadcast bit. (For details about this bit, see 22.7.6 (2) "Broadcast bit".)

Broadcast communication is classified into two types: group-unit broadcast communication and all-unit broadcast communication. Group-unit broadcast and all-unit broadcast are identified by the value of the slave address. (For the slave address, see 22.7.6 (4) "Slave address field".)

#### (1) Group-unit broadcast communication

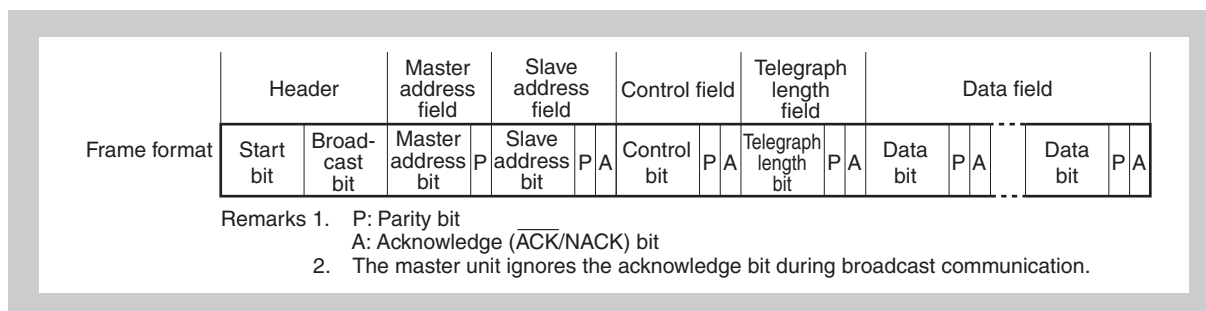
Broadcast communication is performed to the units in a group identified by the group number indicated by the higher 4 bits of the communication address.

#### (2) All-unit broadcast communication

Broadcast communication is performed to all the units, regardless of the value of the group number.

## 22.7.6 IEBus transfer format

The IEBus transfer signal format is shown in *Figure 22-56*.



**Figure 22-56 IEBus transfer signal format**

### (1) Start bit

The start bit is a signal that informs the other units of the start of a data transfer.

The unit that is to start a data transfer outputs a low-level signal (start bit) for a specific time, and then starts outputting the broadcast bit.

If another unit has already output its start bit when one unit is to output the start bit, this unit does not output the start bit and instead waits for completion of output of the start bit by the other unit. When the output of the start bit by the other unit is complete, the unit starts outputting the broadcast bit in synchronization with the completion of the start bit output by the other unit.

The units other than the one that started communication detect this start bit, and enter the reception status.

### (2) Broadcast bit

This bit indicates whether the master selects one slave (individual communication) or multiple slaves (broadcast communication) as the other party of communication.

When the broadcast bit is 0, it indicates broadcast communication. When it is 1, individual communication is indicated. Broadcast communication is classified into two types: group-unit communication and all-unit communication. These communication types are identified by the value of the slave address. (For the slave address, see 22.7.6 (4) "Slave address field".)

Because multiple slave units exist as a partner slave unit of communication in the case of broadcast communication, the NACK signal is returned as an acknowledge bit in each field subsequent to the master address field.

If multiple units start transmitting a communication frame at the same time, broadcast communication takes precedence over individual communication, and wins in arbitration.

If one unit occupies the bus as the master, the value set to the broadcast request flag (the IEBBnBCR.IEBBnALRQ bit) is output.

**(3) Master address field**

The master address field is output by the master to inform a slave of the master's address.

The configuration of the master address field is shown in *Figure 22-57*.

If multiple units start transmitting the broadcast bit at the same time, the master address field makes a judgment of arbitration.

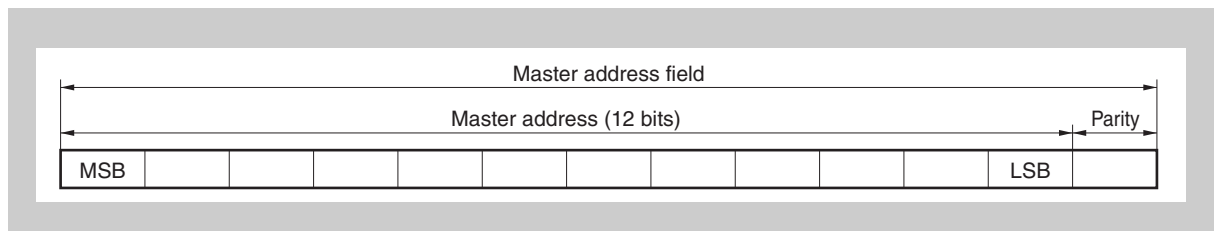
The master address field compares the data it outputs with the data on the bus each time it has output one bit. If the master address output by the master address field is found to differ from the data on the bus as a result of comparison, it is assumed that the master has lost arbitration.

As a result, the master stops transmission and enters the reception status. Because the IEBus is configured of wired AND, the unit having the smallest master address of the units participating in arbitration (arbitration masters) wins arbitration.

After a 12-bit master address has been output, only one unit remains in the transmission status as one master unit.

Next, this master unit outputs a parity bit, determines the master address of other unit, and starts outputting a slave address field.

If one unit occupies the bus as the master, the address specified by the IEbBnUAR register is output.



**Figure 22-57** Master address field

**(4) Slave address field**

The master outputs the address of the unit with which it is to communicate.

The configuration of the slave address field is shown in *Figure 22-58*.

A parity bit is output after a 12-bit slave address has been transmitted to prevent the wrong slave address from being received by mistake. Next, the master unit detects an  $\overline{\text{ACK}}$  signal from the slave unit to confirm that the slave unit exists on the bus. The master unit starts outputting the control field after detecting the  $\overline{\text{ACK}}$  signal. During broadcast communication, however, the master does not confirm the acknowledge bit and instead starts outputting the control field.

The slave unit outputs the  $\overline{\text{ACK}}$  signal if its slave address matches and if the slave detects that the parities of both the master address and slave address are even. The slave unit judges that the master address or slave address has not been correctly received and outputs the NACK signal if the parities are odd. At this time, the master unit is in the standby (monitor) status, and communication ends.

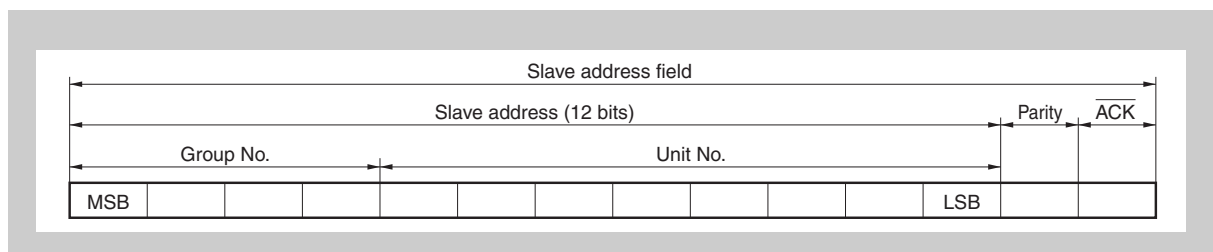
During broadcast communication, the slave address is used to identify group-unit broadcast or all-unit broadcast, as follows:

If the slave address is FFFH: All-unit broadcast communication

If the slave address is not FFFH: Group-unit broadcast communication

**Note** The group No. during group-unit broadcasting communication is the value of the higher 4 bits of the slave address.

If one unit occupies the bus as the master, the address specified by the IEbBnSAR register is output.



**Figure 22-58 Slave address field**

**(5) Control data field**

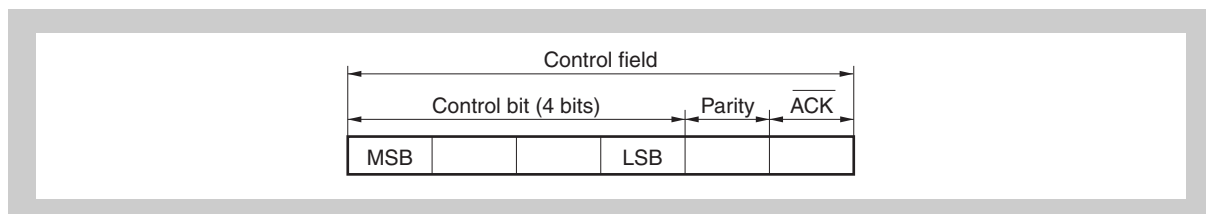
The master uses this field to output the operation it requires the slave to perform.

The configuration of the control field is shown in *Figure 22-59*.

If the parity following the control bit is even and the slave unit can execute the function required by the master unit, the slave unit outputs an ACK signal and starts outputting the message length field. If the slave unit cannot execute the function required by the master unit even if the parity is even, or if the parity is odd, the slave unit outputs the NACK signal, and returns to the standby (monitor) status.

The master unit starts outputting the message length field after detecting the ACK signal.

If the master detects the NACK signal, the master unit enters the standby status, and communication ends. During broadcast communication, however, the master unit does not confirm the acknowledge bit and starts outputting the message length field.



**Figure 22-59 Control field**

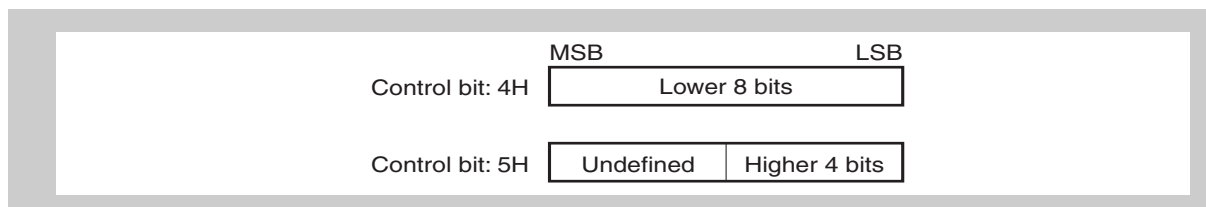
The contents of the control bits are shown below.

**Table 22-57 Control bit contents**

Bit 3 <sup>a</sup>	Bit 2	Bit 1	Bit 0	Function
0	0	0	0	Read slave status
0	0	0	1	Undefined
0	0	1	0	Undefined
0	0	1	1	Read data and lock <sup>b</sup>
0	1	0	0	Read lock address (lower 8 bits) <sup>c</sup>
0	1	0	1	Lock address reading (higher 4 bits) <sup>c</sup>
0	1	1	0	Slave status reading and unlocking <sup>b</sup>
0	1	1	1	Read data
1	0	0	0	Undefined
1	0	0	1	Undefined
1	0	1	0	Command writing and locking <sup>b</sup>
1	0	1	1	Data writing and locking <sup>b</sup>
1	1	0	0	Undefined
1	1	0	1	Undefined
1	1	1	0	Write command
1	1	1	1	Write data



- a) The message length bit of the message length field and data transfer direction of the data field change as follows depending on the value of bit 3 (MSB).  
If bit 3 is 1: Transfer from master unit to slave unit  
If bit 3 is 0: Transfer from slave unit to master unit
- b) This is a control bit that specifies locking or unlocking. (For details, see 22.7.7 (4) "Locking and unlocking".)
- c) The lock address is transferred in 1-byte (8-bit) units and is configured as follows:



If the control bit received from the master unit is not as shown in *Table 22-58*, the unit locked by the master unit rejects acknowledging the control bit, and outputs the NACK signal.

**Table 22-58 Control field for locked slave unit**

Bit 3	Bit 2	Bit 1	Bit 0	Function
0	0	0	0	Read slave status
0	1	0	0	Lock address reading (lower 8 bits)
0	1	0	1	Lock address reading (higher 4 bits)

In addition, units for which locking is not set up by the master unit reject acknowledgment and output a NACK signal when the control data shown in *Table 22-59* is acknowledged.

**Table 22-59 Control field for unlocked slave unit**

Bit 3	Bit 2	Bit 1	Bit 0	Function
0	1	0	0	Lock address reading (lower 8 bits)
0	1	0	1	Lock address reading (higher 4 bits)

If one unit occupies the bus as the master, the value set to the IEBBnTCD register is output.

Table 22-60 Control field  $\overline{\text{ACK}}$  signal response conditions (when the received control data is H, 3H, 4H, 5H, 6H, or 7H)

Communication target (IEBBnUSR.IEBBnSR QF bit) Slave specification = 1 No specification = 0	Lock status (IEBBnUSR.IEB BnLCKF bit) Lock = 1 No lock = 0	Master unit judgment (IEBBnPAR register match) Lock request unit = 1 Other = 0	Slave transmission enabled (IEBBnBCR.IEB BnSTXE bit)	Slave reception enabled (IEBBnBCR.IEB BnSRXE bit)	Received control data					
					0H	3H	4H	5H	6H	7H
1	0	don't care	0	don't care	A	N	N	N	A	N
			1		A	A	N	N	A	A
	1	0	don't care		A	N	A	A	N	N
		1	0		A	N	A	A	A	N
		1	A		A	A	A	A	A	
Other than the above					N					

**Note** A: Slave transmission is performed. (The  $\overline{\text{ACK}}$  signal is returned.)

N: Slave transmission is not performed. (The NACK signal is returned.)

**Caution** If the received control data is other than the data shown in the above table, x is unconditionally assumed. (Slave transmission is not performed (and the NACK signal is returned).)

Table 22-61 Control field  $\overline{\text{ACK}}$  signal response conditions (when the received control data is AH, BH, EH, or FH)

Communication target (IEBBnUSR.IEBBnSR QF bit) Slave specification = 1 No specification = 0	Lock status (IEBBnUSR.IEB BnLCKF bit) Lock = 1 No lock = 0	Master unit judgment (IEBBnPAR register match) Lock request unit = 1 Other = 0	Slave transmission enabled (IEBBnBCR.IEB BnSTXE bit)	Slave reception enabled (IEBBnBCR.IE BBnSRXE bit)	Received control data			
					AH	BH	EH	FH
1	0	don't care	don't care	1	A			
	1	1						
Other than the above					N			

**Note** A: Slave transmission is performed. (The  $\overline{\text{ACK}}$  signal is returned.)

N: Slave transmission is not performed. (The NACK signal is returned.)

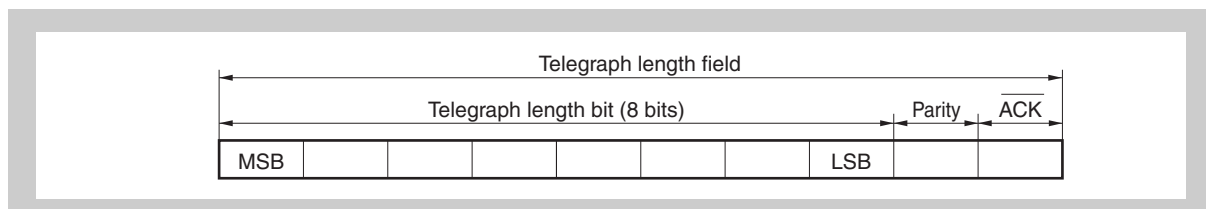
**Caution** If the received control data is other than the data shown in the above table, x is unconditionally assumed. (Slave transmission is not performed (and the NACK signal is returned).)

**(6) Message length field**

This field is output by the transmission side to inform the reception side of the number of bytes of the transmit data.

The configuration of the message length field is shown in *Figure 22-60*.

*Table 22-62* shows the relationship between the message length bit and the number of transmission data bytes.



**Figure 22-60** Message length field

**Table 22-62** Contents of the message length bit

Message length bit (hexadecimal)	Number of transmission data bytes
01H	1 byte
02H	2 bytes
...	...
FFH	255 bytes
00H	256 bytes

The operation of the message length field differs depending on whether the master transmits data (when control bit 3 is 1) or receives data (when control bit 3 is 0).

**(a) During master transmission**

The message length bit and parity bit are output by the master unit and the synchronization signals of bits are output by the master unit. When the slave unit detects that the parity is even, it outputs the  $\overline{\text{ACK}}$  signal, and starts outputting the data field. During broadcast communication, however, the slave unit outputs the NACK signal.

If the parity is odd, the slave unit judges that the message length bit has not been correctly received, outputs the NACK signal, and returns to the standby (monitor) status. At this time, the master unit also returns to the standby status, and communication ends.

**(b) Master reception**

The message length bit and parity bit are output by the slave unit and the synchronization signals of bits are output by the master unit. If the master unit detects that the parity bit is even, it outputs the  $\overline{\text{ACK}}$  signal.

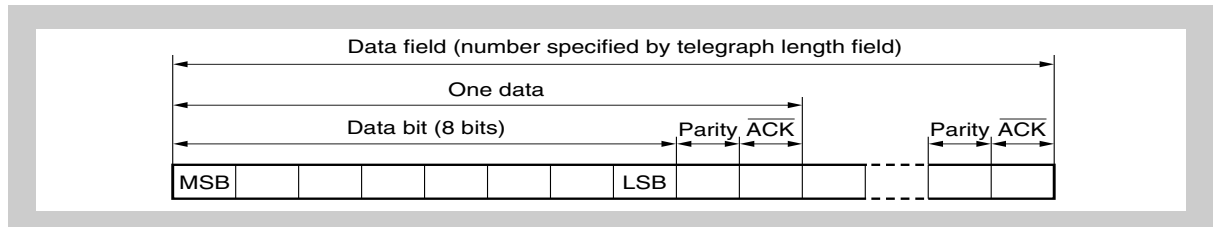
If the parity bit is odd, the master unit judges that the message length bit has not been correctly received, outputs the NACK signal, and returns to the standby status. At this time, the slave unit also returns to the standby status, and communication ends.

**(7) Data field**

This is data output by the transmission side.

The master unit transmits or receives data to or from a slave unit by using the data field.

The configuration of the data field is shown below.



**Figure 22-61 Data field**

Following the data bit, the parity bit and acknowledge bit are output by the master unit and slave unit, respectively.

Use broadcast communication only when the master unit transmits data. At this time, the acknowledge bit is ignored.

The operation differs as follows depending on whether the master transmits or receives data.

**(a) During master transmission**

When the master unit writes data to a slave unit, the master unit transmits the data bit and parity bit to the slave unit. If the parity is even and the received data is not stored in the IEBBnDR register when the slave unit has received the data bit and parity bit, the slave unit outputs an  $\overline{\text{ACK}}$  signal. If the parity is odd or the received data is stored in the IEBBnDR register, the slave unit rejects receiving the data, and outputs the NACK signal.

If the slave unit outputs the NACK signal, the master unit transmits the same data again. This operation continues until the master detects the  $\overline{\text{ACK}}$  signal from the slave unit, or the data exceeds the maximum number of transmit bytes.

If there is more data and the maximum number of transmission bytes is not exceeded when the parity is even and when the slave unit outputs the  $\overline{\text{ACK}}$  signal, the master unit transmits the next data.

During broadcast communication, the slave unit outputs the NACK signal, and the master unit transfers 1 byte of data at a time. If the parity is odd or the IEBBnDR register is storing received data after the slave unit receives the data bit and parity bit during broadcast communication, the slave unit judges that reception has not been performed correctly, and stops reception.

**(b) Master reception**

When the master unit reads data from a slave unit, the master unit outputs a sync signal corresponding to all the read bits.

The slave unit outputs the contents of the data and parity bits to the bus in response to the sync signal from the master unit.

The master unit reads the data and parity bits output by the slave unit, and checks the parity.

If the parity is odd or the IEBBnDR register is storing received data, the master unit rejects accepting the data, and outputs the NACK signal. If the maximum number of transmission bytes is within the value that can be transmitted in one communication frame, the master unit rereads the same data.

If the parity is even and the IEBBnDR register is not storing received data, the master unit accepts the data and outputs the  $\overline{\text{ACK}}$  signal. If the maximum number of transmission bytes is within the value that can be transmitted in one frame, the master unit reads the next data.

---

**Caution** During broadcast communication, do not perform master reception. If you do this, the slave unit cannot be defined and data transfers cannot be performed correctly.

---

Note that, due to the IEBBn specifications, overrun errors can occur. Therefore, even if reading the IEBBnDR register is late during individual communication and the system has reached the timing for receiving the next data (the overrun status), data can be retransmitted from the master unit by returning a NACK signal, which makes it possible to buy time for reading the IEBBnDR register. However, during broadcast communication, because no  $\overline{\text{ACK}}$  signal is output from the slave unit and the master unit ignores  $\overline{\text{ACK}}$  signals, even if reading the IEBBnDR register is late, no data is retransmitted from the master. Therefore, for IEBBn, if an overrun occurs during broadcast communication, normal reception is not possible, an overrun error occurs, and an interrupt request (for a communication error) is output.

**(8) Parity bit**

The parity bit is used to make sure that the transmission data has no error.

The parity bit is appended to each data of the master address, slave address, control, message length, and data bits.

The parity is an even parity. If the number of data bits that are '1' is odd, the parity bit is '1'. If the number of data bits that are '1' is even, the parity bit is '0'.

**(9) Acknowledge bit**

During normal communication (communication from one unit to another), an acknowledge bit is appended to the following locations to check whether the data has been correctly received.

- End of slave address field
- End of control field
- End of message length field
- End of data field

The definition of the acknowledge bit is as follows.

0: The transmission data is recognized. ( $\overline{\text{ACK}}$  signal)

1: The transmission data is not recognized. (NACK signal)

During broadcast communication, however, the contents of the acknowledge bit are ignored.

**(a) Last acknowledge bit of the slave address field**

The last acknowledge bit of the slave address field serves as a NACK signal in any of the following cases, and transmission is stopped.

- If the parity of the master address bit or slave address bit is incorrect
- If a timing error (an error in the bit format) occurs
- If a slave unit does not exist

**(b) Last acknowledge bit of the control field**

The last acknowledge bit of the control field serves as a NACK signal in any of the following cases, and transmission is stopped.

- If the parity of the control bit is incorrect
- If control bit 3 is 1 (write operation) when the slave reception enable flag (the IEBBnBCR.IEBBnSRXE bit) is not set (to 1)  
(For details, see 22.3.2 (1) "IEBBnBCR - IEBBn bus control register".)
- If control bit data is read (3H, 7H) when the slave reception enable flag (the IEBBnBCR.IEBBnSRXE bit) is not set (to 1)  
(For details, see 22.3.2 (1) "IEBBnBCR - IEBBn bus control register".)
- If a unit other than one that has set locking requests 3H, 6H, 7H, AH, BH, EH, or FH of the control bit when locking is set
- If the control bit indicates reading of lock addresses (4H, 5H) even when locking is not set
- If a timing error occurs
- If the control bit is undefined

---

**Cautions** 1. The  $\overline{\text{ACK}}$  signal is always returned when the control data of the slave status request is received, if the IEBBnSTXE bit = 0.

2. The NACK signal is returned by the acknowledge bit in the control field when the control data for data/command writing is received, even if the IEBBnSRXE bit = 0.

Slave reception can be disabled (communication stopped) by the IEBBnSRXE bit only in the case of individual communication. In the case of broadcast communication, communication is maintained and the data interrupt (IEBBTD) or completion interrupt (IEBBTSTA) is generated.

---

**(c) Last acknowledge bit of message length field**

The last acknowledge bit of the message length field serves as a NACK signal in any of the following cases, and transmission is stopped.

- If the parity of the message length bit is incorrect
- If a timing error occurs

**(d) Last acknowledge bit of the data field**

The last acknowledge bit of the data field serves as a NACK signal in any of the following cases, and transmission is stopped.

- If the parity of the data bit is incorrect<sup>a</sup>
- If a timing error occurs after the preceding acknowledge bit has been transmitted
- If the received data is stored in the IEBBnDR register and no more data can be received<sup>a</sup>

a) In this case, when the communication executed is individual communication, if the maximum number of transmit bytes is within the value that can be transmitted in one frame, the transmission side executes transmission of that data field again. For broadcast communication, the transmission side does not execute transmission again, a communication error occurs on the reception side and reception stops.

## 22.7.7 Transfer data

**(1) Slave status**

The master unit can learn why the slave unit did not return the  $\overline{\text{ACK}}$  signal by reading the slave status.

The slave status is determined according to the result of the last communication the slave unit has executed.

All the slave units can supply information on the slave status.

The configuration of the slave status is shown below.

MSB				LSB			
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Bit 0 <sup>a</sup>	Function						
0	Transmit data is not written in IEBBnDR register						
1	Transmit data is written in IEBBnDR register						
Bit 1	Function						
0	Receive data is not stored in IEBBnDR register						
1	Receive data is stored in IEBBnDR register						
Bit 2	Function						
0	Unit is not locked						
1	Unit is locked						
Bit 3	Function						
0	Fixed to 0						
Bit 4 <sup>b</sup>	Function						
0	Slave transmission is stopped						
1	Slave transmission is ready						
Bit 5	Function						
0	Fixed to 0						
Bit 7	Bit 6	Function					
0	0	Communication mode 0	Indicates the highest communication mode supported by the unit <sup>c</sup> .				
0	1	Communication mode 1					
1	0	Communication mode 2					
1	1	Not used					

a) After reset: Bit 0 is set to 1.

b) When the V850E2/Sx4-H serves as a slave unit, this bit corresponds to the status indicated by IEBBnBCR.IEBBnSTXE bit.

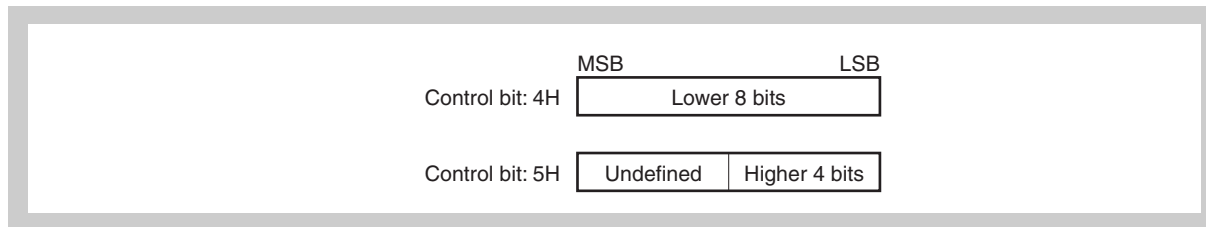
c) Bits 7 and 6 are fixed to "10" because this module can support communication modes 1 and 2.

Figure 22-62 Slave status bit configuration



**(2) Lock address**

When the lock address is read (control bit: 4H or 5H), the address (12 bits) of the master unit that has issued the lock instruction is configured in 1-byte units as shown below and read.



**Figure 22-63 Lock address configuration**

**(3) Data**

If the control bit indicates reading of data (3H or 7H), the data in the data buffer of the slave unit is read by the master unit.

If the control bit indicates writing of data (BH or FH), the data received by the slave unit is processed according to the operation rule of that slave unit.

**(4) Locking and unlocking**

The lock function is used when a message is transferred in two or more communication frames.

The unit that is locked does not receive data from units other than the one that has locked the unit (does not receive broadcast communication).

A unit is locked or unlocked as follows.

**(a) Lock setting**

If the communication frame is completed without succeeding to transmit or receive data of the number of bytes specified by the message length bit after the message length field has been transmitted or received ( $\overline{ACK} = 0$ ) by the control bit that specifies locking (3H, AH, or BH), the slave unit is locked by the master unit. At this time, the bit (bit 2) in the byte indicating the slave status is set to '1'.

**(b) Unlocked**

After transmitting or receiving data of the number of data bytes specified by the message length bit in one communication frame by the control bit that has specified locking (3H, AH, or BH), or the control bit that has specified unlocking (6H), the slave unit is unlocked by the master unit. At this time, the bit related to locking (bit 2) in the byte indicating the slave status is reset to '0'.

Locking or unlocking is not performed during broadcast communication.

Locking and unlocking conditions are shown below.

**Table 22-63 Setting conditions:**

Control data	Broadcast communication		Individual communication	
	End of communication	End of frame	End of communication	End of frame
3H, 6H <sup>a</sup>	–	–	Cannot be locked	Lock set
AH, BH	Cannot be locked	Cannot be locked	Cannot be locked	Lock set
0H, 4H, 5H, EH, FH	Cannot be locked	Cannot be locked	Cannot be locked	Cannot be locked

a) The frame end of control data 6H (slave status read/unlock) occurs when the parity in the data field is odd, and when the NACK signal from the IEBus unit is repeated with up to the maximum number of transfer bytes being output.

**Table 22-64 Unlocking conditions (while locked)**

Control data	Broadcast communication from the lock request unit		Individual communication from the lock request unit	
	End of communication	End of frame	End of communication	End of frame
3H, 6H <sup>a</sup>	–	–	Unlocked	Remains locked
AH, BH	Unlocked	Unlocked	Unlocked	Remains locked
0H, 4H, 5H, EH, FH	Remains locked	Remains locked	Remains locked	Remains locked

a) The frame end of control data 6H (slave status read/unlock) occurs when the parity in the data field is odd, and when the NACK signal from the IEBus unit is repeated with up to the maximum number of transfer bytes being output.

### 22.7.8 Bit format

The format of the bits constituting the communication frame of the IEBus is shown below.

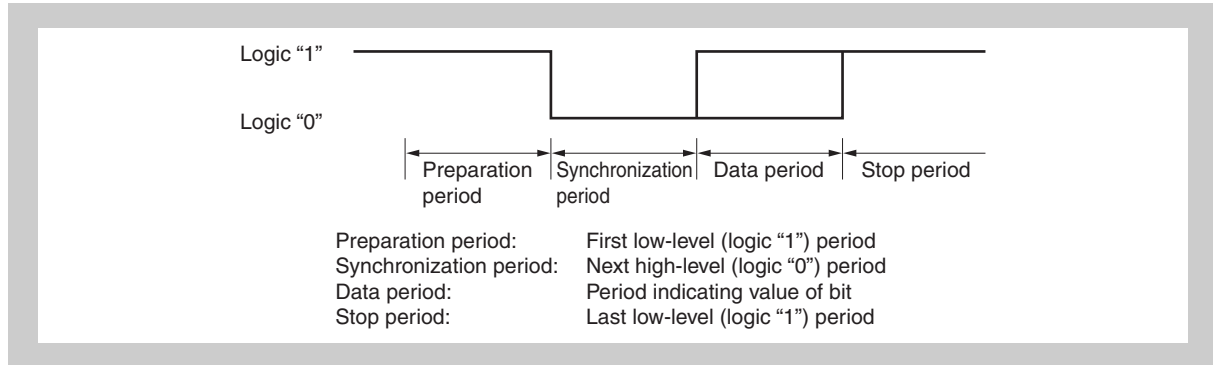


Figure 22-64 IEBus bit format

The synchronization period and data period are almost equal to each other in length.

The IEBus synchronizes each bit. The specifications on the time of the entire bit and the time related to the period allocated to that bit differ depending on the type of transmit bit, or whether the unit is the master unit or a slave unit. The master and slave units monitor whether each period (preparation period, synchronization period, data period, and stop period) is output for the specified time while they are in communication. If a period is not output for the specified time, the master and slave units report a timing error, immediately terminate communication, and enter the standby status.

## 23. Renesas SPDIF Interface

### 23.1 Overview

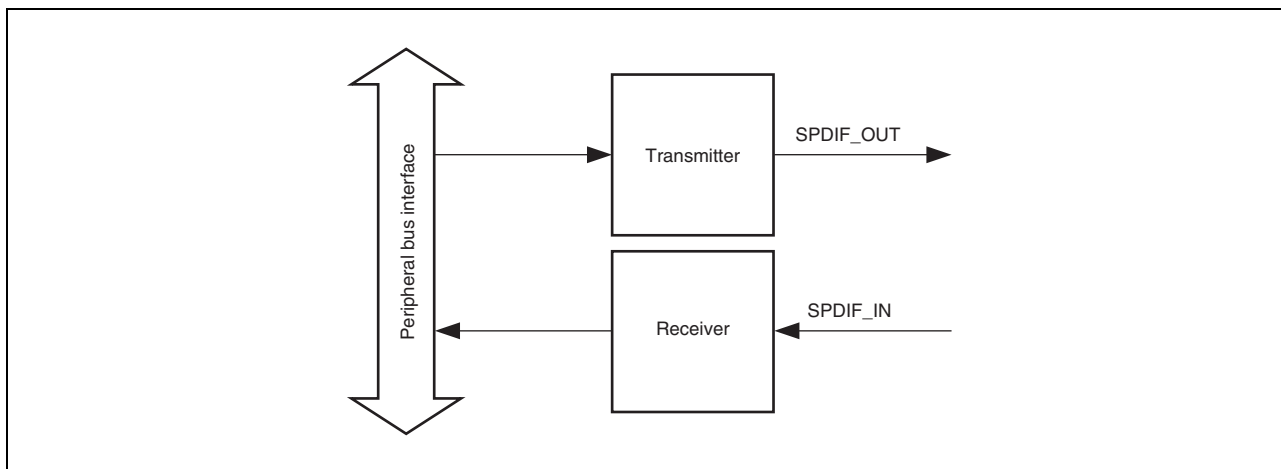


Figure 23.1 Overview Block Diagram

### 23.2 Features

- Supports the IEC 60958 standard (stereo and consumer use modes only).
- Supports sampling frequencies of 32 kHz, 44.1 kHz, and 48 kHz.
- Supports audio word sizes of 16 to 24 bits per sample.
- Biphase mark encoding.
- Double buffered data.
- Parity encoded serial data.
- Simultaneous transmit and receive
- Receiver autodetects IEC 61937 compressed mode data

### 23.3 Functional Block Diagram

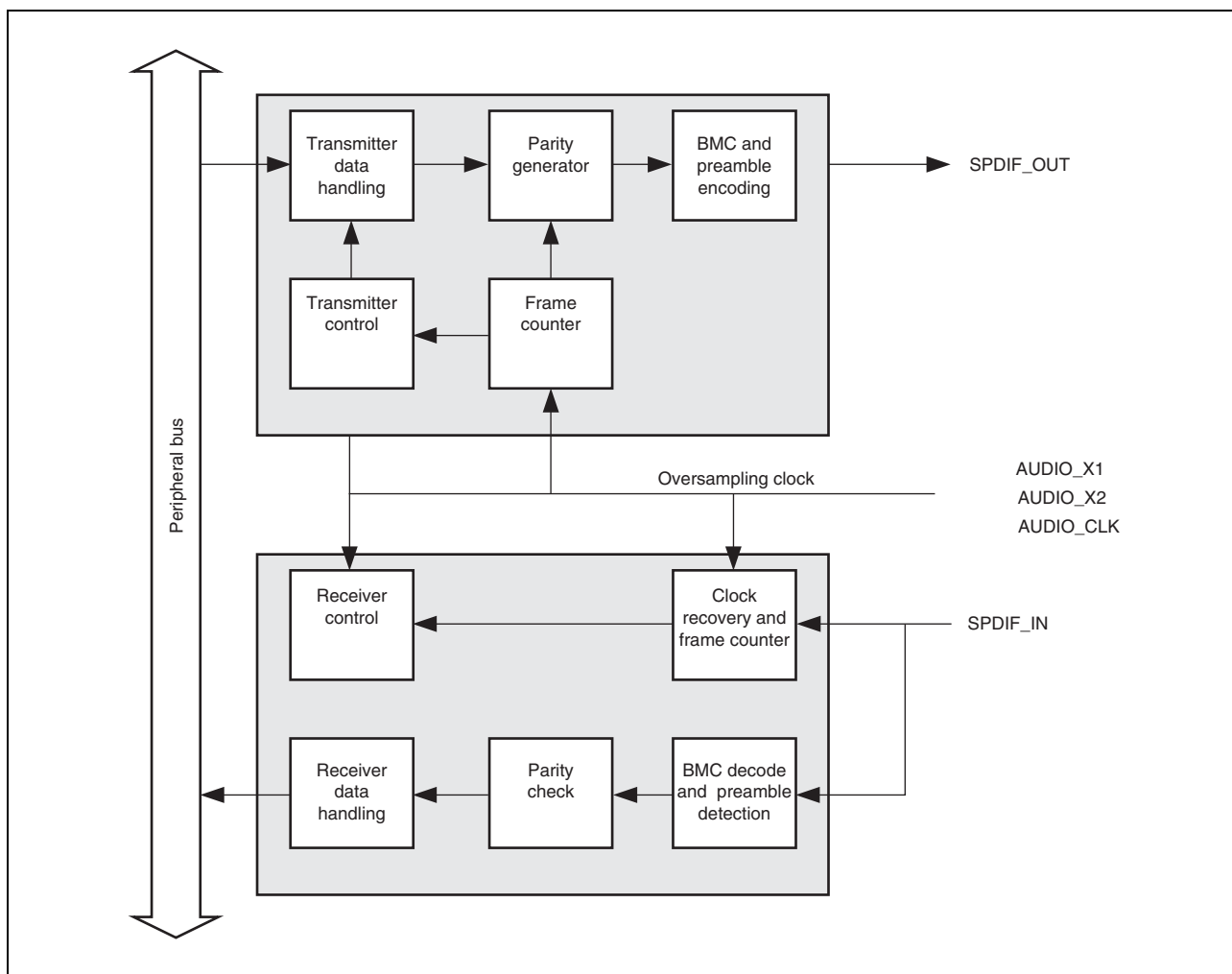


Figure 23.2 Functional Block Diagram

## 23.4 Input/Output Pins

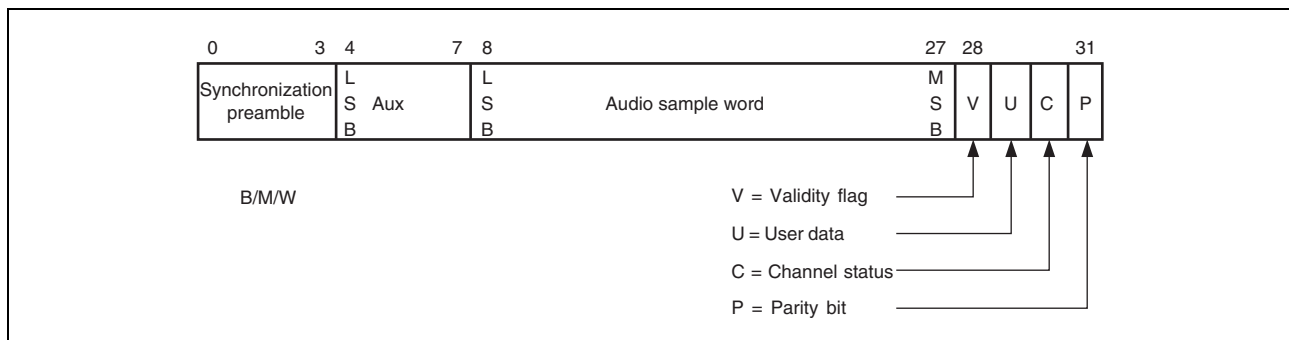
Table 23.1 shows the pin configuration.

**Table 23.1 Pin Configuration**

Channel	Pin Name	I/O	Description
0	SPDIF_OUT	Output	Transmitter biphasemark encoded SPDIF bitstream
1	SPDIF_IN	Input	Receiver biphasemark encoded SPDIF bitstream
0, 1 (Common)	AUDIO_CLK	Input	External clock for audio
	AUDIO_X1	Input	Crystal resonator/external clock for audio
	AUDIO_X2	Output	

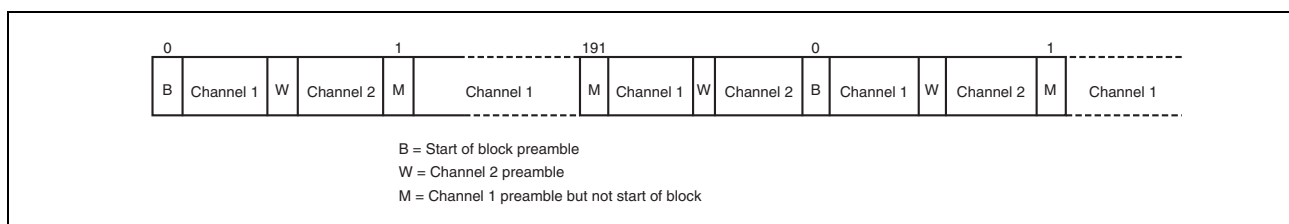
## 23.5 Renesas SPDIF (IEC60958) Frame Format

The Renesas SPDIF frame consists of two subframes (for channels 1 and 2), each of which contains a 4-bit preamble, audio data of up to 24 bits, a V flag, a user bit, a channel status bit, and an even parity bit. Figure 23.3 shows the subframe format. According to this format, the Renesas SPDIF performs biphasemark modulation (channel coding) that will make the transmission line's DC component a minimum value.



**Figure 23.3 Subframe Format**

Figure 23.4 shows the block format, which consists of 192 continuous frames. One block begins at the starting frame (preamble B) and ends at the 192nd frame (frame 191), and the preamble is used to identify all subframes. Each block has a total of 384 subframes, which are classified into three categories: subframe 0 indicating the beginning of a new block, subframe 1 (usually the channel 1), and subframe 2 (usually the channel 2). Usually, the music data sent and received by the SPDIF is continuous so that continuous blocks appear.



**Figure 23.4 Block Format**

Table 23.2 shows the binary values of the Renesas SPDIF preambles. The polarity of these preambles differs depending on the status of the preceding symbol (parity bit).

**Table 23.2 Binary Preamble Values**

Preamble	Preceding Symbol's Status = 0	Preceding Symbol's Status = 1
B	11101000	00010111
M	11100010	00011101
W	11100100	00011011

Note: As shown in Figure 23.3, the even parity bit at time slot 31 of a subframe determines the type of a preamble for one cycle of transmission. Usually, therefore, any one is selected from the set states that are sent through the Renesas SPDIF. However, IEC60958 requires decoding both types in view of connection with the preamble polarity reversed; the Renesas SPDIF has preambles decoded according to Table 23.2.

Channel status information is encoded at the rate of one bit per subframe, making the channel status information per block have a total of 192 bits for each of subframes 1 and 2. For the format of the channel status, refer to the IEC 60958 standard.

## 23.6 Register

Table 23.3 shows the register configuration.

**Table 23.3 Register Configuration**

Channel	Register Name	Abbreviation	Address	Access Size
0 (Transmit)	Transmitter channel 1 audio register	TLCA	H'FFFF D800	32
	Transmitter channel 2 audio register	TRCA	H'FFFF D804	32
	Transmitter channel 1 status register	TLCS	H'FFFF D808	32
	Transmitter channel 2 status register	TRCS	H'FFFF D80C	32
	Transmitter user data register	TUI	H'FFFF D810	32
1 (Receive)	Receiver channel 1 audio register	RLCA	H'FFFF D814	32
	Receiver channel 2 audio register	RRCA	H'FFFF D818	32
	Receiver channel 1 status register	RLCS	H'FFFF D81C	32
	Receiver channel 2 status register	RRCS	H'FFFF D820	32
	Receiver user data register	RUI	H'FFFF D824	32
0, 1 (Common)	Control register	CTRL	H'FFFF D828	32
	Status register	STAT	H'FFFF D82C	32
0, 1 (Common)	Transmitter DMA audio data register	TDAD	H'FFFF D830	32
	Receiver DMA audio data register	RDAD	H'FFFF D834	32

Note: All registers are longword registers and must be accessed as such.

A register diagram containing a 0 indicates that the write value should always be 0 (if the register is writeable) and that the read value should always be 0 (if readable).



## 23.7 Register Descriptions

### Legend:

Initial Value: Register value after reset

—: Undefined value

R/W: Readable/writable register. The write value can be read.

R: Read only register. The write value should always be 0.

R/WC0: Readable/writable register. Writing 0 initializes the bit, but writing 1 is ignored.

R/WC1: Readable/writable register. Writing 1 initializes the bit, but writing 0 is ignored.

W: Write only register. Reading is prohibited. If this bit is reserved, the write value should always be 0.

—/W: Write only, Read value undefined

### 23.7.1 Control Register (CTRL)

Bit:	31	30	29	28	27	26	25	24
	-	-	-	CKS	-	PB	RASS	
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R/W	R/W	R/W

Bit:	23	22	21	20	19	18	17	16
	TASS		RDE	TDE	NCSI	AOS	RME	TME
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8
	REIE	TEIE	UBOI	UBUI	CREI	PAEI	PREI	CSEI
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	7	6	5	4	3	2	1	0
	ABOI	ABUI	RUII	TUII	RCSI	RCBI	TCSI	TCBI
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved
28	CKS	0	R/W	Oversampling clock select Selects oversampling clock supply source. 0: AUDIO_X1 1: AUDIO CLK
27	—	0	R	Reserved
26	PB	0	R/W	Pass Back Passes transmitter SPDIF output into SPDIF receiver in SPDIF module. 0: Pass Back disabled 1: Pass Back enabled
25, 24	RASS	All 0	R/W	Receiver Audio Sample Bit Size These bits Indicate the receiver audio sample bit size (16, 20, or 24 bits), for data alignment purposes. 00: 16-bit sample 01: 20-bit sample 10: 24-bit sample 11: Reserved

Bit	Bit Name	Initial Value	R/W	Description
23, 22	TASS	All 0	R/W	Transmitter Audio Sample Bit Size These bits Indicate the transmitter audio sample bit size (16, 20, or 24 bits), for data alignment purposes. 00: 16-bit sample 01: 20-bit sample 10: 24-bit sample 11: Reserved
21	RDE	0	R/W	Receiver DMA Enable Enables DMA requests for the receiver. 0: Receiver DMA disabled 1: Receiver DMA enabled
20	TDE	0	R/W	Transmitter DMA Enable Enables the DMA requests for the transmitter. 0: Transmitter DMA disabled 1: Transmitter DMA enabled
19	NCSI	0	R/W	New Channel Status Information Set this bit to 1 when new channel status information to be corrected is in the transmitter. 0: New channel status information has not been in transmitter 1: New channel status information has been in transmitter
18	AOS	0	R/W	Audio Only Samples Clear this bit to 0 when audio channel 1 and channel 2 registers contain user information. When this bit is set to 1, all user bits are cleared to 0. 0: User information present 1: User information not present
17	RME	0	R/W	Receiver Module Enable Enables the receiver module. 0: Receiver module disabled 1: Receiver module enabled
16	TME	0	R/W	Transmitter Module Enable Enables the transmitter module. 0: Transmitter module disabled 1: Transmitter module enabled
15	REIE	0	R/W	Receiver Error Interrupt Enable Enables the receiver error interrupts. 0: Receiver error interrupt disabled 1: Receiver error interrupt enabled
14	TEIE	0	R/W	Transmitter Error Interrupt Enable Enables the transmitter error interrupts. 0: Transmitter error interrupt disabled 1: Transmitter error interrupt enabled
13	UBOI	0	R/W	User Buffer Overrun Interrupt Enable Enables the user buffer overrun interrupts. 0: User buffer overrun interrupt disabled 1: User buffer overrun interrupt enabled
12	UBUI	0	R/W	User Buffer Underrun Interrupt Enable Enables the user buffer underrun interrupts. 0: User buffer underrun interrupt disabled 1: User buffer underrun interrupt enabled
11	CREI	0	R/W	Clock Recovery Error Interrupt Enable Enables the clock recovery error interrupts. 0: Clock recovery error interrupt disabled 1: Clock recovery error interrupt enabled
10	PAEI	0	R/W	Parity Error Interrupt Enable Enables the parity check error interrupts. 0: Parity check error interrupt disabled 1: Parity check error interrupt enabled

Bit	Bit Name	Initial Value	R/W	Description
9	PREI	0	R/W	Preamble Error Interrupt Enable Enables the preamble check error interrupts. 0: Preamble error interrupt disabled 1: Preamble error interrupt enabled
8	CSEI	0	R/W	Channel Status Error Interrupt Enable Enables the channel status error interrupts. 0: Channel status error interrupt disabled 1: Channel status error interrupt enabled
7	ABOI	0	R/W	Audio Buffer Overrun Interrupt Enable Enables the receiver audio buffer overrun interrupts. 0: Audio buffer overrun interrupt disabled 1: Audio buffer overrun interrupt enabled
6	ABUI	0	R/W	Audio Buffer Underrun Interrupt Enable Enables the transmitter audio buffer underrun interrupts. 0: Audio buffer underrun interrupt disabled 1: Audio buffer underrun interrupt enabled
5	RUII	0	R/W	Receiver User Information Interrupt Enable Enables the receiver user information register full interrupts. 0: Receiver user information interrupt disabled 1: Receiver user information interrupt enabled
4	TUII	0	R/W	Transmitter User Information Interrupt Enable Enables the transmitter user information register empty interrupts. 0: Transmitter user information interrupt disabled 1: Transmitter user information interrupt enabled
3	RCSI	0	R/W	Receiver Channel Status Interrupt Enable Enables the receiver channel status register full interrupts. 0: Receiver channel status interrupt disabled 1: Receiver channel status interrupt enabled
2	RCBI	0	R/W	Receiver Channel Buffer Interrupt Enable Enables the receiver audio channel buffer full interrupts. 0: Receiver audio channel interrupt disabled 1: Receiver audio channel interrupt enabled
1	TCSI	0	R/W	Transmitter Channel Status Interrupt Enable Enables the transmitter channel status register empty interrupts. 0: Transmitter channel status interrupt disabled 1: Transmitter channel status interrupt enabled
0	TCBI	0	R/W	Transmitter Channel Buffer Interrupt Enable Enables the transmitter audio channel buffer empty interrupts. 0: Transmitter audio channel interrupt disabled 1: Transmitter audio channel interrupt enabled

## 23.7.2 Status Register (STAT)

Bit:	31	30	29	28	27	26	25	24
	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
Bit:	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	CMD
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8
	RIS	TIS	UBO	UBU	CE	PARE	PREE	CSE
Initial value:	1	1	0	0	0	0	0	0
R/W:	R	R	R/WC0	R/WC0	R/WC0	R/WC0	R/WC0	R/WC0
Bit:	7	6	5	4	3	2	1	0
	ABO	ABU	RUIR	TUIR	CSRX	CBRX	CSTX	CBTX
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/WC0	R/WC0	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved
16	CMD	0	R	Compressed Mode Data Sets if the data being received is compressed mode data (When bit 1 = 1 in the V flag and channel status). 0: Data is not in compressed mode 1: Data is in compressed mode
15	RIS	1	R	Receiver Idle State Sets if the receiver is in the idle state. 0: Receiver is not in idle state 1: Receiver in idle state
14	TIS	1	R	Transmitter Idle State Sets if the transmitter is in the idle state. 0: Transmitter is not in idle state 1: Transmitter is in idle state
13	UBO	0	R/WC0	User Buffer Overrun* Sets if the receiver user buffer overruns. This bit is cleared by writing 0 to the register. If bit REIE and bit UBOI in the control register are set this causes an interrupt. 0: User buffer has not overrun 1: User buffer has overrun
12	UBU	0	R/WC0	User Buffer Underrun* Sets if the transmitter user buffer underrun. This bit is cleared by writing 0. If bits TEIE and UBUI in the control register are set this causes an interrupt. 0: User buffer has not underrun 1: User buffer has underrun
11	CE	0	R/WC0	Clock Error* Sets when the clock recovery falls out of synchronization. This bit is cleared by writing 0. If bits REIE and CREI in the control register are set this causes an interrupt. 0: Clock recovery stable 1: Clock recovery error
10	PARE	0	R/WC0	Parity Error* Sets when the parity checker produces a fail result. This bit is cleared by writing 0. If bits REIE and PAEI in the control register are set this causes an interrupt. 0: Parity check correct 1: Parity error

Bit	Bit Name	Initial Value	R/W	Description
9	PREE	0	R/WC0	<p>Preamble Error*</p> <p>Sets when the start of word preamble fails to appear in the correct place. This bit is cleared by writing 0. If bits REIE and PREI in the control register are set this causes an interrupt.</p> <p>Note: Only set after a start of block preamble has occurred.</p> <p>0: Preamble is in the correct place 1: Preamble error</p>
8	CSE	0	R/WC0	<p>Channel Status Error*<sup>1</sup></p> <p>Sets when the channel status information is written before the 32nd frame of the current block. This bit is cleared by writing 0. If bits TEIE and CSEI in the control register are set this causes an interrupt.</p> <p>0: Channel status correct 1: Channel status error</p>
7	ABO	0	R/WC0	<p>Audio Buffer Overrun*<sup>1</sup></p> <p>Indicates that the receiver audio buffer is full in both the first and second stages and that data has been overwritten. This bit is cleared by writing 0. If bits REIE and ABOI in the control register are set then this causes an interrupt.</p> <p>0: Receiver audio buffer has not overrun 1: Receiver audio buffer has overrun</p>
6	ABU	0	R/WC0	<p>Audio Buffer Underrun*<sup>1</sup></p> <p>Indicates that the transmitter audio buffer is empty in both the first and second stages and that the last data transmission has been repeated. This bit is cleared by writing 0. If bits TEIE and ABUI in the control register are set then this causes an interrupt.</p> <p>0: Transmitter audio buffer has not underrun 1: Transmitter audio buffer has underrun</p>
5	RUIR	0	R	<p>Receiver User Information Register Status</p> <p>Indicates the status of the receiver user information register. This bit is cleared by reading from the receiver user register. If bit RUII in the control register is set then this causes an interrupt.</p> <p>0: Receiver user information register is empty 1: Receiver user information register is full</p>
4	TUIR	0	R	<p>Transmitter User Information Register Status</p> <p>Indicates the status of the transmitter user information register. This bit is cleared by writing to the transmitter user register. If bit TUII in the control register is set then this causes an interrupt.</p> <p>0: Transmitter user information register is full 1: Transmitter user information register is empty</p>
3	CSRX	0	R	<p>Channel 1 and Channel 2 Status for Receiver</p> <p>Indicates the status of the receiver channel status registers. This bit is cleared by reading from the receiver channel status registers. If bit RCSI in the control register is set this causes an interrupt.</p> <p>0: Receiver channel status registers are empty 1: Receiver channel status registers are full</p>
2	CBRX	0	R	<p>Channel 1 and Channel 2 Buffers for Receiver</p> <p>Indicates the status of the receiver audio channel registers. This bit is cleared by reading from the receiver audio channel registers. If bit RCBI in the control register is set this causes an interrupt.</p> <p>0: Receiver audio channel registers are empty 1: Receiver audio channel registers are full</p>
1	CSTX	0	R	<p>Channel 1 and Channel 2 Status for Transmitter</p> <p>Indicates the status of the transmitter channel status registers. This bit is cleared by writing to the transmitter channel status registers. If bit TCSI in the control register is set this causes an interrupt.</p> <p>0: Transmitter channel status register is full 1: Transmitter channel status register is empty</p>

Bit	Bit Name	Initial Value	R/W	Description
0	CBTX	0	R	Channel 1 and Channel 2 Buffers for Transmitter Indicates the status of the transmitter audio channel registers. This bit is cleared by writing to the transmitter audio channel registers. If bit TCBI in the control register is set this causes an interrupt. 0: Transmitter audio channel registers are full 1: Transmitter audio channel registers are empty

Note 1. When an error bit is detected during DMA transfer, DMA transfer settings must be made again. In this case, the Renesas SPDIF's module enable bit (either the RME or TME bit) and the DMA enable bit (either the RDE or TDE bit) must be disabled and the error status must be cleared before making DMA transfer settings again. Then the module enable bit should be set and DMA transfer can be started again.

### 23.7.3 Transmitter Channel 1 Audio Register (TLCA)

Bit:	31	30	29	28	27	26	25	24
	-	-	-	-	-	-	-	-
Initial value:	-	-	-	-	-	-	-	-
R/W:	W	W	W	W	W	W	W	W
Bit:	23	22	21	20	19	18	17	16
	Audio PCM Data							
Initial value:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W
Bit:	15	14	13	12	11	10	9	8
	Audio PCM Data							
Initial value:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W
Bit:	7	6	5	4	3	2	1	0
	Audio PCM Data							
Initial value:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	—	W	Reserved
23 to 0	Audio PCM Data	All 0	W	Audio PCM Data LSB aligned PCM encoded audio data.

### 23.7.4 Transmitter Channel 2 Audio Register (TRCA)

Bit:	31	30	29	28	27	26	25	24
	-	-	-	-	-	-	-	-
Initial value:	-	-	-	-	-	-	-	-
R/W:	W	W	W	W	W	W	W	W
Bit:	23	22	21	20	19	18	17	16
	Audio PCM Data							
Initial value:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W
Bit:	15	14	13	12	11	10	9	8
	Audio PCM Data							
Initial value:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W
Bit:	7	6	5	4	3	2	1	0
	Audio PCM Data							
Initial value:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	—	W	Reserved
23 to 0	Audio PCM Data	All 0	W	Audio PCM Data LSB aligned PCM encoded audio data.

### 23.7.5 Transmitter DMA Audio Data Register (TDAD)

Bit:	31	30	29	28	27	26	25	24
	-	-	-	-	-	-	-	-
Initial value:	-	-	-	-	-	-	-	-
R/W:	W	W	W	W	W	W	W	W
Bit:	23	22	21	20	19	18	17	16
	Audio PCM Data							
Initial value:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W
Bit:	15	14	13	12	11	10	9	8
	Audio PCM Data							
Initial value:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W
Bit:	7	6	5	4	3	2	1	0
	Audio PCM Data							
Initial value:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	—	W	Reserved
23 to 0	Audio PCM Data	All 0	W	Audio PCM Data LSB aligned PCM encoded audio data.

### 23.7.6 Transmitter User Data Register (TUI)

U-bit data in subframes is written in to this register. Because U-bit data is transmitted in a sequence of subframes 1 and 2, you need to update the data on a 16-frame basis. For the contents of the user bytes refer to the appropriate standard for the device in use. The user bits to be transmitted are set in sequence starting at the LSB.

Bit:	31	30	29	28	27	26	25	24
	User Byte 4							
Initial value:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W
Bit:	23	22	21	20	19	18	17	16
	User Byte 3							
Initial value:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W
Bit:	15	14	13	12	11	10	9	8
	User Byte 2							
Initial value:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W
Bit:	7	6	5	4	3	2	1	0
	User Byte 1							
Initial value:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	User Byte 4	All 0	W	U-bit information is stored here.
23 to 16	User Byte 3	All 0	W	
15 to 8	User Byte 2	All 0	W	
7 to 0	User Byte 1	All 0	W	



### 23.7.7 Transmitter Channel 1 Status Register (TLCS)

The 30-bit register stores the channel status information to be transmitted. For each channel, channel status information per frame consists of 192 bits. Because necessary data covers only the 30 bits that are set in the following register, zeros continue to be sent after the transmission of the first 30 bits.

Bit:	31	30	29	28	27	26	25	24
	-	-	CLAC[1:0]		FS[3:0]			
Initial value:	-	-	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W
Bit:	23	22	21	20	19	18	17	16
	CHNO[3:0]				SRCNO[3:0]			
Initial value:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W
Bit:	15	14	13	12	11	10	9	8
	CATCD[7:0]							
Initial value:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W
Bit:	7	6	5	4	3	2	1	0
	-	-	CTL[4:0]				-	-
Initial value:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	—	W	Reserved
29, 28	CLAC[1:0]	All 0	W	Clock Accuracy 00: Level 2 01: Level 1 10: Level 3 11: Reserved
27 to 24	FS[3:0]	All 0	W	Sample Frequency (FS) 0000: 44.1 kHz 0010: 48 kHz 0011: 32 kHz
23 to 20	CHNO[3:0]	All 0	W	Channel Number 0000: Don't care 0001: A (left channel) 0010: B (right channel) 0011: C
19 to 16	SRCNO[3:0]	All 0	W	Source Number 0000: Don't care 0001: 1 0010: 2 0011: 3
15 to 8	CATCD[7:0]	All 0	W	Category Code (Example) 00000000: 2-channel general format 00000001: 2-channel compact disc (IEC 908) 00000010: 2-channel PCM encoder/decoder 00000011: 2-channel digital audio tape recorder
7, 6	—	All 0	W	Reserved The write value should always be 0.
5 to 1	CTL[4:0]	All 0	W	Control The control bits are copied from the source (see IEC60958 standard).
0	—	0	W	Reserved The write value should always be 0.

### 23.7.8 Transmitter Channel 2 Status Register (TRCS)

The 30-bit register stores the channel status information to be transmitted. For each channel, channel status information per frame consists of 192 bits. Because necessary data covers only the 30 bits that are set in the following register, zeros continue to be sent after the transmission of the first 30 bits.

Bit:	31	30	29	28	27	26	25	24
	-	-	CLAC[1:0]		FS[3:0]			
Initial value:	-	-	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W
Bit:	23	22	21	20	19	18	17	16
	CHNO[3:0]				SRCNO[3:0]			
Initial value:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W
Bit:	15	14	13	12	11	10	9	8
	CATCD[7:0]							
Initial value:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W
Bit:	7	6	5	4	3	2	1	0
	-	-	CTL[4:0]				-	-
Initial value:	0	0	0	0	0	0	0	0
R/W:	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	—	W	Reserved
29, 28	CLAC[1:0]	All 0	W	Clock Accuracy 00: Level 2 01: Level 1 10: Level 3 11: Reserved
27 to 24	FS[3:0]	All 0	W	Sample Frequency (FS) 0000: 44.1 kHz 0010: 48 kHz 0011: 32 kHz
23 to 20	CHNO[3:0]	All 0	W	Channel Number 0000: Don't care 0001: A (left channel) 0010: B (right channel) 0011: C
19 to 16	SRCNO[3:0]	All 0	W	Source Number 0000: Don't care 0001: 1 0010: 2 0011: 3
15 to 8	CATCD[7:0]	All 0	W	Category Code (Example) 00000000: 2-channel general format 00000001: 2-channel compact disc (IEC 908) 00000010: 2-channel PCM encoder/decoder 00000011: 2-channel digital audio tape recorder
7, 6	—	All 0	W	Reserved The write value should always be 0.
5 to 1	CTL[4:0]	All 0	W	Control The control bits are copied from the source (see IEC60958 standard).
0	—	0	W	Reserved The write value should always be 0.

## 23.7.9 Receiver Channel 1 Audio Register (RLCA)

Bit:	31	30	29	28	27	26	25	24
	-	-	-	-	-	-	-	-
Initial value:	-	-	-	-	-	-	-	-
R/W:	R	R	R	R	R	R	R	R
Bit:	23	22	21	20	19	18	17	16
	Audio PCM Data							
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8
	Audio PCM Data							
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
Bit:	7	6	5	4	3	2	1	0
	Audio PCM Data							
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	—	R	Reserved
23 to 0	Audio PCM Data	All 0	R	Audio PCM Data LSB aligned PCM encoded audio data.

## 23.7.10 Receiver Channel 2 Audio Register (RRCA)

Bit:	31	30	29	28	27	26	25	24
	-	-	-	-	-	-	-	-
Initial value:	-	-	-	-	-	-	-	-
R/W:	R	R	R	R	R	R	R	R
Bit:	23	22	21	20	19	18	17	16
	Audio PCM Data							
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8
	Audio PCM Data							
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
Bit:	7	6	5	4	3	2	1	0
	Audio PCM Data							
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	—	R	Reserved
23 to 0	Audio PCM Data	All 0	R	Audio PCM Data LSB aligned PCM encoded audio data.

## 23.7.11 Receiver DMA Audio Data (RDAD)

Bit:	31	30	29	28	27	26	25	24
	-	-	-	-	-	-	-	-
Initial value:	-	-	-	-	-	-	-	-
R/W:	R	R	R	R	R	R	R	R
Bit:	23	22	21	20	19	18	17	16
	Audio PCM Data							
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8
	Audio PCM Data							
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
Bit:	7	6	5	4	3	2	1	0
	Audio PCM Data							
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	—	R	Reserved
23 to 0	Audio PCM Data	All 0	R	Audio PCM Data LSB aligned PCM encoded audio data.

### 23.7.12 Receiver User Data Register (RUI)

The register stores the U-bit data received through the Renesas SPDIF. Because U-bit data is stored in a sequence of subframes 1 and 2 starting at the LSB, you need to read the data on a 16-frame basis. For the contents of the user bytes refer to the appropriate standard for the device in use.

Bit:	31	30	29	28	27	26	25	24
	User Byte 4							
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
Bit:	23	22	21	20	19	18	17	16
	User Byte 3							
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8
	User Byte 2							
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
Bit:	7	6	5	4	3	2	1	0
	User Byte 1							
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	User Byte 4	All 0	R	U-bit information is stored here.
23 to 16	User Byte 3	All 0	R	
15 to 8	User Byte 2	All 0	R	
7 to 0	User Byte 1	All 0	R	

### 23.7.13 Receiver Channel 1 Status Register (RLCS)

The channel status is stored starting at the register's LSB in a way that subframe 1 received from the beginning of the block is stored. For the contents of the channel status register, refer to the IEC-60958 standard.

Bit:	31	30	29	28	27	26	25	24
	-	-	CLAC[1:0]				FS[3:0]	
Initial value:	-	-	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
Bit:	23	22	21	20	19	18	17	16
			CHNO[3:0]				SRCNO[3:0]	
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8
							CATCD[7:0]	
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R
Bit:	7	6	5	4	3	2	1	0
	-	-					CTL[4:0]	-
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	—	R	Reserved
29, 28	CLAC[1:0]	All 0	R	Clock Accuracy 00: Level 2 01: Level 1 10: Level 3 11: Reserved
27 to 24	FS[3:0]	All 0	R	Sample Frequency (FS) 0000: 44.1 kHz 0010: 48 kHz 0011: 32 kHz
23 to 20	CHNO[3:0]	All 0	R	Channel Number 0000: Don't care 0001: A (left channel) 0010: B (right channel) 0011: C
19 to 16	SRCNO[3:0]	All 0	R	Source Number 0000: Don't care 0001: 1 0010: 2 0011: 3
15 to 8	CATCD[7:0]	All 0	R	Category Code (Example) 00000000: 2-channel general format 00000001: 2-channel compact disc (IEC 908) 00000010: 2-channel PCM encoder/decoder 00000011: 2-channel digital audio tape recorder
7, 6	—	All 0	R	Reserved
5 to 1	CTL[4:0]	All 0	R	Control The control bits are copied from the source (see IEC60958 standard).
0	—	0	R	Reserved

### 23.7.14 Receiver Channel 2 Status Register (RRCS)

The channel status is stored starting at the register's LSB in a way that subframe 2 received from the beginning of the block is stored. For the contents of the channel status register, refer to the IEC-60958 standard.

Bit:	31	30	29	28	27	26	25	24
	-	-	CLAC[1:0]		FS[3:0]			
Initial value:	-	-	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit:	23	22	21	20	19	18	17	16
	CHNO[3:0]				SRCNO[3:0]			
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8
	CATCD[7:0]							
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit:	7	6	5	4	3	2	1	0
	-	-	CTL[4:0]				-	-
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	—	R	Reserved
29, 28	CLAC[1:0]	All 0	R	Clock Accuracy 00: Level 2 01: Level 1 10: Level 3 11: Reserved
27 to 24	FS[3:0]	All 0	R	Sample Frequency (FS) 0000: 44.1 kHz 0010: 48 kHz 0011: 32 kHz
23 to 20	CHNO[3:0]	All 0	R	Channel Number 0000: Don't care 0001: A (left channel) 0010: B (left channel) 0011: C
19 to 16	SRCNO[3:0]	All 0	R	Source Number 0000: Don't care 0001: 1 0010: 2 0011: 3
15 to 8	CATCD[7:0]	All 0	R	Category Code (Example) 00000000: 2-channel general format 00000001: 2-channel compact disc (IEC 908) 00000010: 2-channel PCM encoder/decoder 00000011: 2-channel digital audio tape recorder
7, 6	—	All 0	R	Reserved
5 to 1	CTL[4:0]	All 0	R	Control The control bits are copied from the source (see IEC60958 standard).
0	—	0	R	Reserved

## 23.8 Functional Description—Transmitter

### 23.8.1 Transmitter Module

The transmitter module transmits PCM data and auxiliary information after encoding it according to the method of biphase-mark modulation that complies with the IEC60958 standard (SPDIF).

The clock for the transmitter module is an oversampling clock supplied from the outside. This clock usually selects a value that serves as an oversample at a frequency eight times larger than the clock frequency required for biphase-mark encoding. In this case, the clock frequency required to transmit 32 time slots in a subframe is 512 times as large as the sample frequency for audio data.

Audio data and channel status information are first written into the module's channel 1 and then into channel 2.

Generally, the channel status need to be written only when the information changes. The SPDIF module requests that the channel status be written in 30 frames -- when all the current channel status data have been transmitted. You need to write somewhere between frame 31 and the beginning of the next block of 192 frames.

The audio data is stored in a double buffer arrangement. To make sure that the first stage buffer is empty, you can send an interrupt request or poll the status register. DMA transfers send channel 1 audio data on the first request and channel 2 data on the second.

The channel status information is stored in the 30-bit registers of channels 1 and 2. For each channel, the channel status information per frame consists of 192 bits. Because necessary data covers only 30 bits, zeros continue to be sent after the transmission of the first 30 bits until the block is completed.

User data forms a 32-bit double buffer arrangement. You can make sure that the first stage buffer is empty by either sending an interrupt request or polling the status register. Usually, information about the user data will become insufficient with the length of data between blocks. Transmission takes place in a sequence of channels 1 and 2. For the user data within a block, 384 bits are transmitted before the next block is continuously transmitted.

The audio data handled by the Renesas SPDIF module is a linear PCM, making it possible to set up to 24 bits. For this reason, the V flag indicating that audio data is a linear PCM remains to be 0. The V flag involves no register-based setting. An even parity is created for each 32 bits of serial output data (excluding the preamble).

Note: • When transmitter user buffer underrun occurs, the current data in the buffer data of SPDIF is transmitted until the next data is filled.

### 23.8.2 Transmitter Module Initialization

The device defaults to an idle state when it comes out of reset, or can be put into an idle state when 0 is written to the TME bit in the CTRL register. When the transmitter module is idle, it has the following settings:

- The transmitter idle status bit (TIS) is set to 1, all other status bits are cleared to 0.
- Preamble generation is invalid.
- Synchronization between channels 1 and 2 is set to 0 (0 for channel 1, 1 for channel 2).
- Both word\_count and frame\_count are set to 0.
- The output from the biphase-mark encoder is set to 0.

Channel status, user and audio data registers will retain its value prior to putting the module into idle. To exit the idle state the user must write 1 to the TME bit in the CTRL register.



### 23.8.3 Initial Settings for Transmitter Module

When the TME bit is set to 1, the TUIR and CSTX bits are set to 1. After that, if data is written in the order of 1) TUI and 2) TLCS and TRCS, a channel status error will occur. To avoid this, be sure to write data in the order of 1) TLCS and TRCS and 2) TUI.

Before writing the first audio data (write access to TLCA or TRCA by the CPU or write access to TDAD by the DMA transfer) after setting the TME bit to 1, be sure to check that the CSTX and TUIR bits are cleared by writing to TLCS, TRCS, and TUI.

### 23.8.4 Transmitter Module Data Transfer

Once the transmitter module has left the idle state, it is ready for data transfer. Data transfer timing can be achieved in three ways. Either the transfer is done by interrupts, DMA requests or by polling the status register. There is a shared interrupt line (for both transmit and receive) and a single transmitter DMA request line.

Figure 23.5 shows a data transfer with an interrupt for the transmitter.

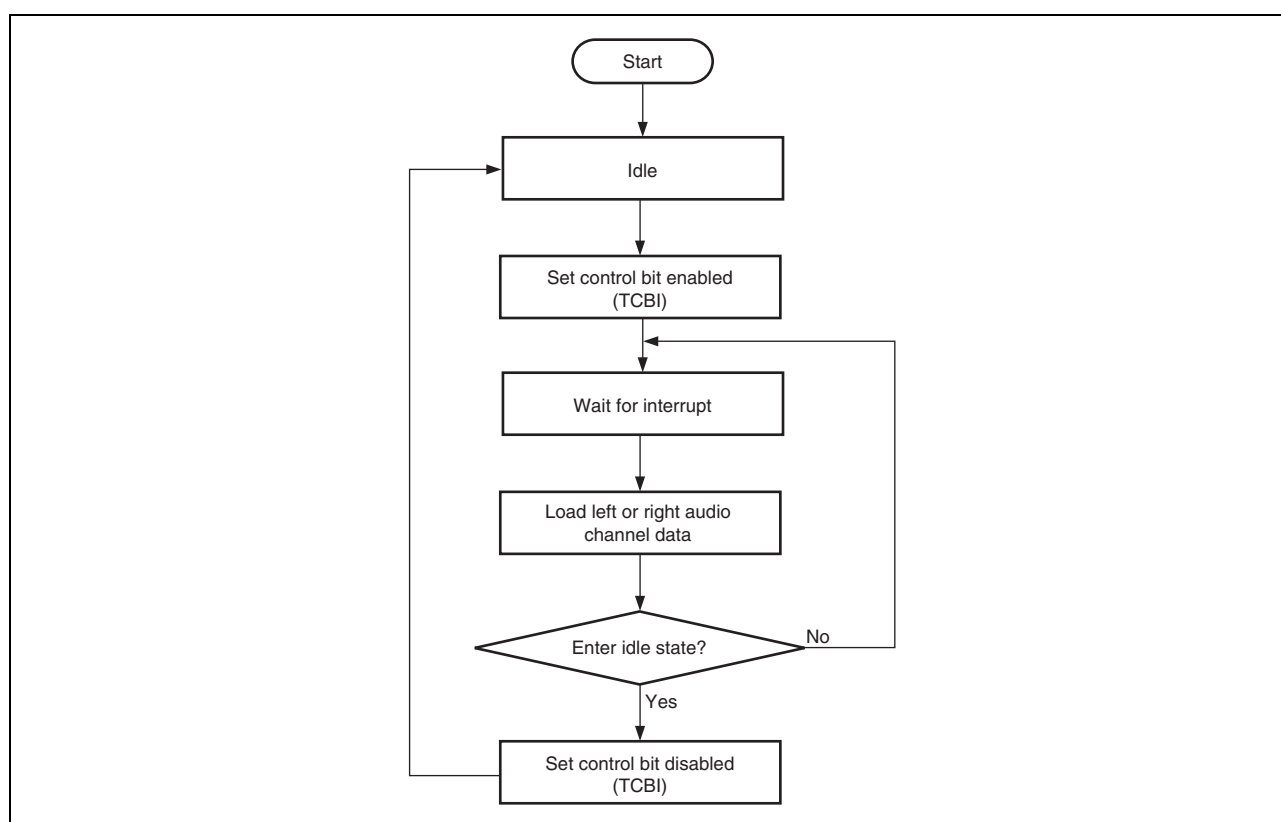
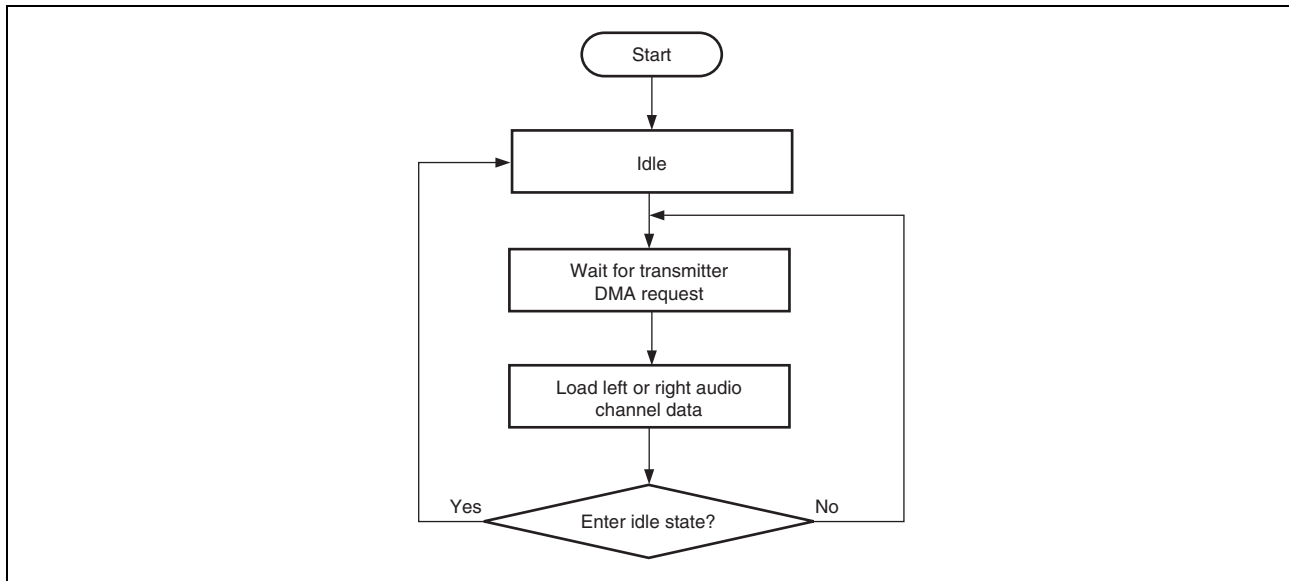


Figure 23.5 Transmitter Data Transfer Flow Diagram - Interrupt Driven

Figure 23.6 shows a data transfer with a DMA transfer for the transmitter.



**Figure 23.6 Transmitter Data Transfer Flow Diagram—DMA Request Driven**

Channel status information is required to be updated when the information has changed. Because the updating needs to be done before the transmission of the next block, the channel status to be updated should be written after 30 frames have been sent; this is indicated either by an interrupt or by polling the status bit. If channel status is written before 30 frames have been sent (while current information is being sent) then an interrupt indicates that the channel status error bit (CSE) in the status register has been set.

Note: • 30 frames contains all the valid information in a single channel status block.

## 23.9 Functional Description—Receiver

### 23.9.1 Receiver Module

The receiver module demodulates data and clock signals from the input encoded according to the IEC60958 standard. The encoded data, shown in linear PCM format, is stored into the audio data register. The register also stores the channel status and user information being received simultaneously as auxiliary information.

The main clock for the receiver module is an oversampling clock supplied from the outside. The module operates at a frequency four times as large as the oversampling clock.

Note: • The oversampling clock is the same for the transmitter and receiver.

Clock recovery is performed using a pulse width counter and averaging filters to produce a sampling pulse in the middle of each bit in the datastream. A clock error status bit indicates clock synchronization loss. Synchronization is achieved when a preamble occurs on the data stream for the first time. Continuous adjustment prevents jitter and/or clock drift from affecting clock recovery, provided that they fall within the clock recovery specifications.

Once the clock recovery is successful the biphase-mark decoder initiates its preamble detection. The decoder searches for the start of block preamble (see Table 23.2). A preamble error status bit indicates that following preambles have not appeared at the correct time, such failures are most likely caused by transmission loss or interference.

Even parity checking is performed on the decoded data. A discrepancy will result in the parity error status bit being set. The SPDIF module acquires user data and channel status information in addition to audio data. The audio is stored in a double buffer arrangement. Either an interrupt request because of a full buffer or polling of the status bit will indicate when the data is ready to be read. DMA transfers receive channel 1 audio data on the first request and channel 2 data on the second.

Channel status is stored in a 30-bit register. Channel status information is received at 1-bit per subframe. Therefore the registers will not be full until a total of 30 frames for each channel have been received. New channel status is compared with the current data to see if it has changed and is only read by the processor if it has. User data, which is also received at the same time, is stored into the register on a subframe basis, so that the reception is completed when 16 frames are reached.

Note 1. Channel status data requests do not support DMA.

Note 2. When receiver user buffer overrun occurs, the current data in the buffer data of SPDIF is overwritten by the next incoming data from SPDIF interface.

### 23.9.2 Receiver Module Initialization

The device defaults to an idle state when it comes out of reset, or can be put into an idle state by writing 0 to bit RME in the CTRL register. Whilst idle the module has the following settings:

- The receiver idle status bit is set to 1, all other status bits are cleared to 0.
- Synchronization between channels 1 and 2 is set to 0 (0 for channel 1, 1 for channel 2).
- Both Word\_count and frame\_count are set to 0.

Channel status registers, user data registers and audio data registers will retain its value prior to putting the module into idle. To exit the idle state the user must write 1 to the bit RME in the CTRL register.

### 23.9.3 Receiver Module Data Transfer

Once the module has left the idle state it is ready for data transfer. Data transfer timing can be achieved in three ways. The transfer can be done by interrupts, or by polling the status register, or by DMA. There is a shared interrupt line (transmit and receive) and a single receiver DMA request line. Data transfer for the receiver can be interrupted by error signals caused by:

1. Clock recovery failure.
2. Transmission loss or interference – indicated by a preamble error.
3. Parity check failure.

Transmission loss or interference can cause the start of subframe or start of block preamble to be misplaced or not present.

Parity check failure occurs when the parity bit is incorrect, this can be caused by any of the above.

- Clock Recovery Deviation

The receive margin for clock recovery is based on the following equation:

$$M = \left| \left( 0.5 - \frac{1}{2N} \right) - (L - 0.5) F - \frac{|D - 0.5|}{N} (1 + F) \right| \times 100\%$$

where  $M$  = receive margin

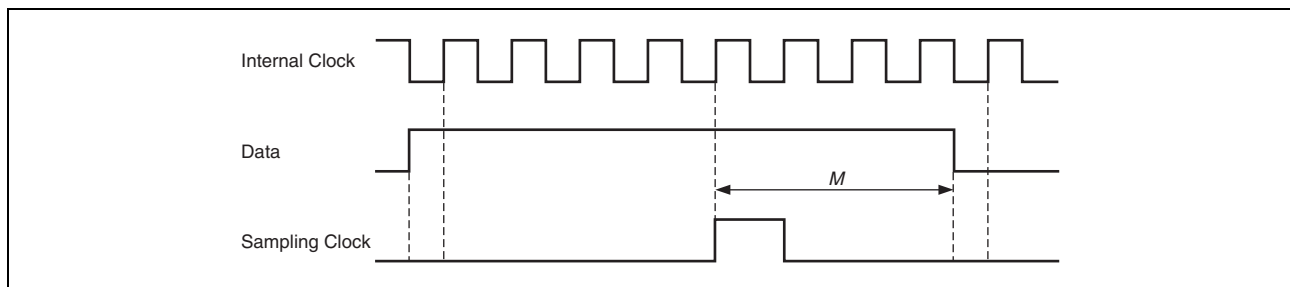
$N$  = oversampling rate

$L$  = frame length = 33

$D$  = duty cycle = 0.6

$F$  = oversampling clock deviation = Level II accuracy = 1000 in  $10e^{-6}$

Figure 23.7 indicates what the receive margin  $M$  represents



**Figure 23.7 Receive Margin**

Introducing jitter into the equation gives the following inequality.

$$j \leq \left| \left( 0.5 - \frac{1}{2N} \right) - (L - 0.5) F - \frac{|D - 0.5|}{N} (1 + F) \right| \times 100\%$$

$J$  = clock jitter

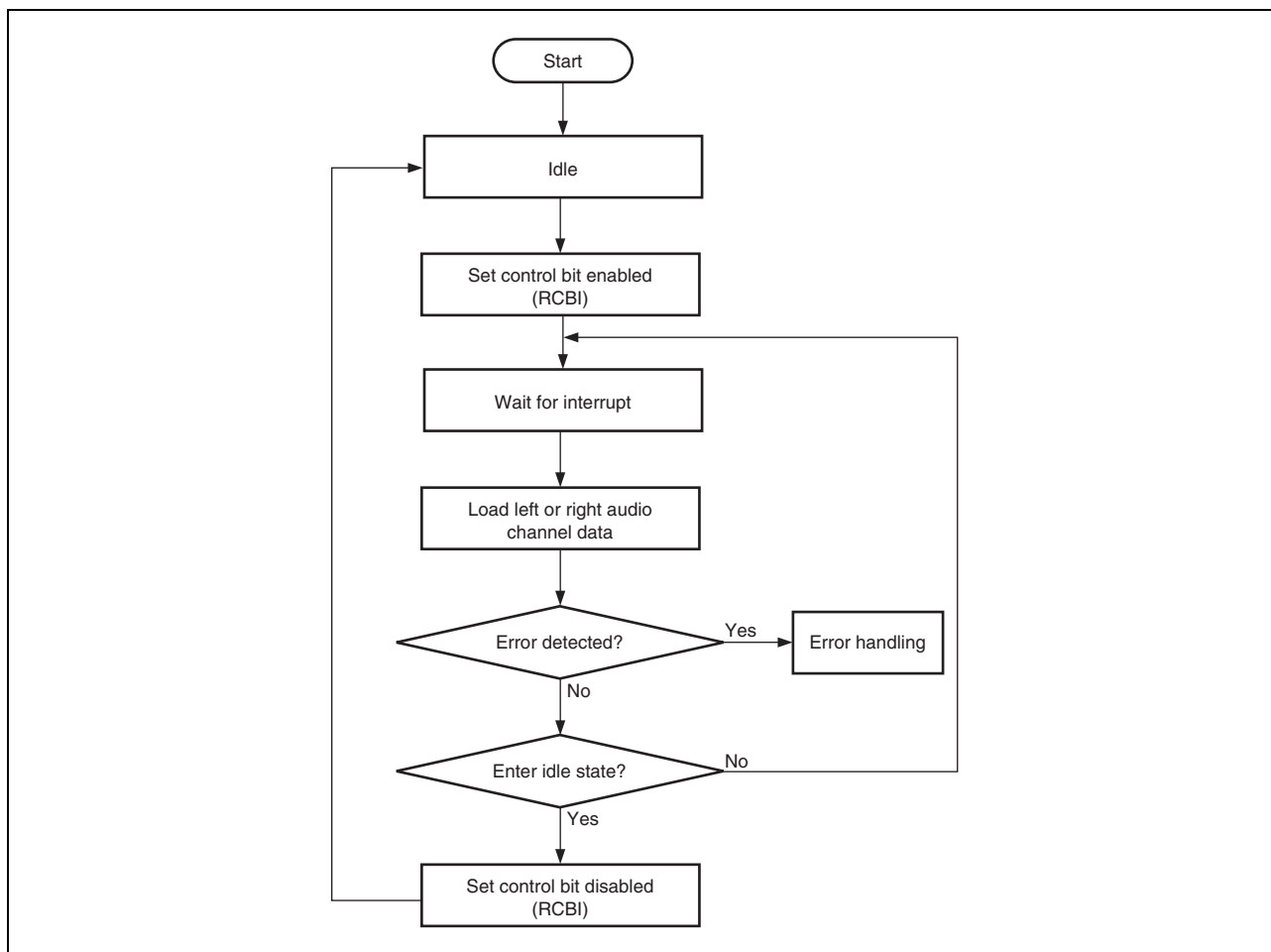
Eight times oversampling produces a receive margin = 39.25%

Four times oversampling produces a receive margin = 31.75%

Two times oversampling produces a receive margin = 16.75%

The fastest sample frequency is 48 kHz. This requires a clock speed of  $128 \times 48 \text{ kHz} = 6.144 \text{ MHz}$ . The worst case jitter in one cycle is specified at  $40 \text{ ns} = 24.5\%$  of the period. This means that an oversampling rate of 4 or more will satisfy the inequality and therefore be sufficient for clock recovery.

Figure 23.8 illustrates the receiver data transfer using interrupts.



**Figure 23.8 Receiver Data Transfer Flow Diagram - Interrupt Driven**

Interrupts to indicate that the channel status information register is full occur after frame 30 has been received and only if the information has changed. When the first four bytes have been stored an interrupt occurs.

## 23.10 Disabling the Module

### 23.10.1 Transmitter and Receiver Idle

The transmitter or receiver modules can be disabled by writing 0 to the idle bit in the control register (TME for the transmitter and RME for the receiver). The idle state can be detected by polling the idle bit in the status register (TIS and RIS).

## 23.11 Compressed Mode Data

Compressed mode data is defined in the IEC 61937 specification. This module only detects compressed mode data. This is done by checking the parity flag (V flag) and bit 1 in the channel status data. If both are one then the data is in compressed mode. This is indicated by the setting of the CMD bit in the status register.

Note: • Only the receiver detects compressed mode data since the information is not relevant to the transmitter.

## 23.12 References

IEC60958 Digital Audio Interface

IEC61937 Compressed Mode Digital Audio Interface

## 23.13 Usage Notes

### 23.13.1 Clearing TUIR

After TUI is written to, the TUIR bit is cleared only after transmission of a maximum of one frame is completed. When using a transmitter user information interrupt to write data to TUI, check that the TUIR bit is cleared before terminating the interrupt handling routine so that the interrupt is not unexpectedly accepted again.

### 23.13.2 Frequency of Clock Input for Audio

The frequency of the clock input to the AUDIO\_X1 and AUDIO\_X2 or AUDIO\_CLK must be lower than the B $\phi$  frequency.

## 24. CD-ROM Decoder

The CD-ROM decoder decodes streams of data transferred from the CD-DSP. When the medium is CD-DA\*1, the data stream is not input to the CD-ROM decoder because it consists of PCM data. In the case of CD-ROM\*2, the stream of data is input and the CD-ROM decoder performs sync code detection and maintenance, descrambling, ECC correction, and EDC checking, and outputs the resulting stream of data.

However, since the stream received by the CD-ROM decoder is assumed to consist of data from a CD-ROM transferred via the serial sound interface, the decoder does not bother with the subcodes defined in the CD-DA standard.

Note 1. Compliant with JIS S 8605 (Red Book)

Note 2. Compliant with JIS X 6281 (Yellow Book)

### 24.1 Features

- Sync-code detection and maintenance

Detects sync codes from the CD-ROM and is capable of providing sync-code maintenance (automatic interpolation of sync codes) when the sync code cannot be detected because of defects such as scratches on the disc.

Five sector-synchronization modes are supported: automatic sync maintenance mode, external sync mode, interpolated sync mode, and interpolated sync plus external sync mode.

- Descrambling

- ECC correction

P-parity-based correction, Q-parity-based correction, PQ correction, and QP correction are available.

PQ correction and QP correction can be applied repeatedly up to three times. This, however, depends on the speed of the CD. For example, three iterations are possible when the CD-ROM decoder is operating at 60 MHz with a double-speed CD drive.

Two buffers are provided due to the need for ECC correction. This allows parallel operation, where ECC correction is performed in one buffer while the data stream is being received in the other.

- EDC checking

The EDC is checked before and after ECC correction. An operating mode is available in which, if the result of pre-ECC correction EDC checking indicates no errors, ECC correction is not performed regardless of the result of syndrome calculation.

- Data buffering control

The CD-ROM decoder outputs data to the buffer area in a specific format where the sync code is at the head of the data for each sector.

### 24.1.1 Formats Supported by CD-ROM Decoder

This module supports the five formats shown in Figure 24.1.

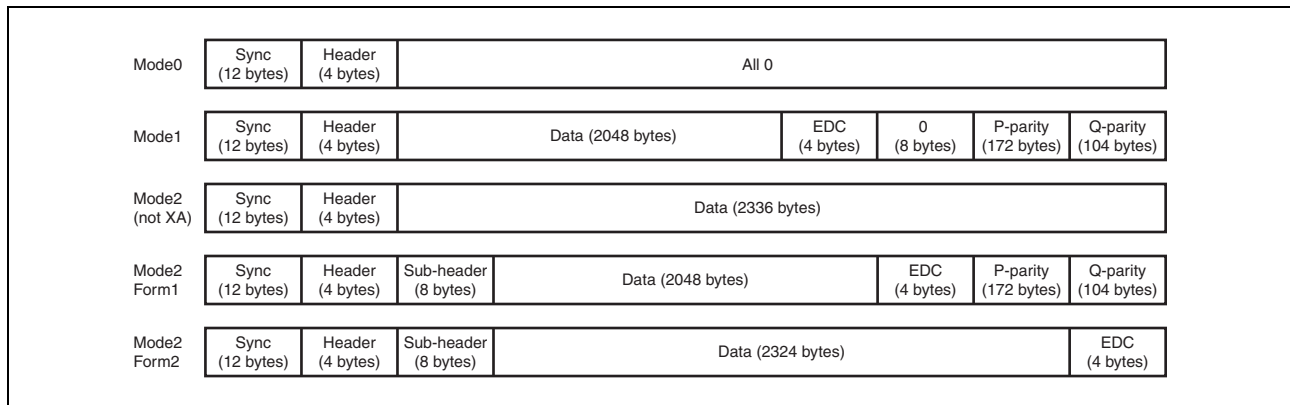


Figure 24.1 Formats Supported by CD-ROM Decoder

### 24.2 Block Diagrams

Figure 24.2 is a block diagram of the CD-ROM decoder functions of this LSI and the bus bridge for connection to the bus, that is, of the elements required to implement the CD-ROM decoder function.

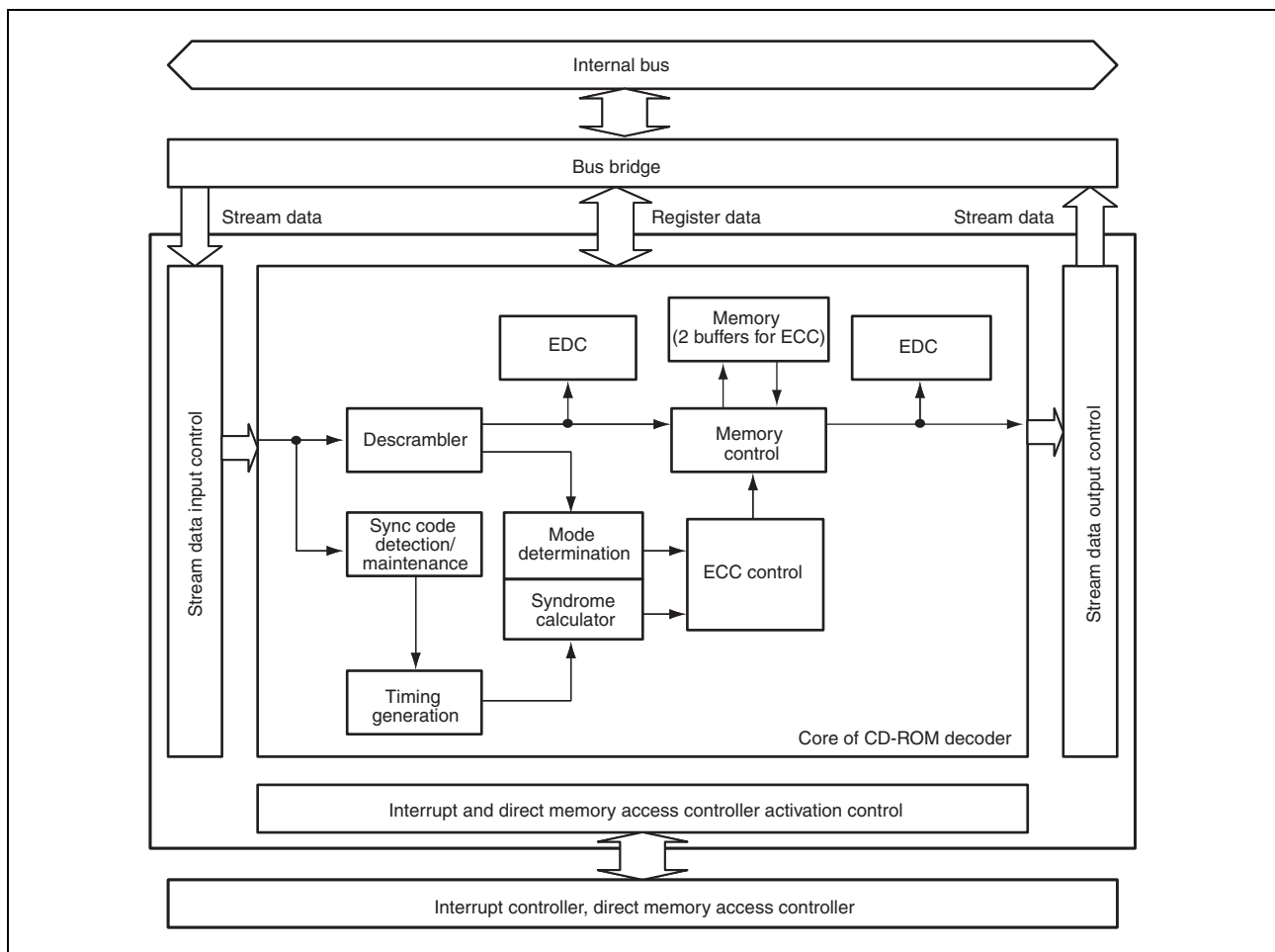


Figure 24.2 Block Diagram



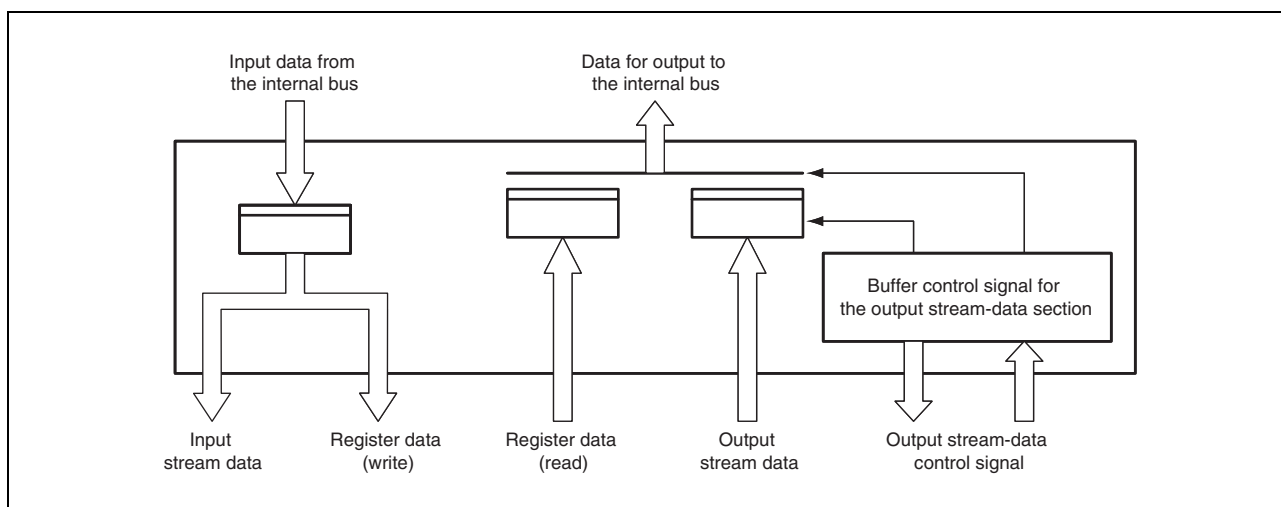
The core of the CD-ROM decoder executes a series of processing required for CD-ROM decoding, including descrambling, sync code detection, ECC correction (P- and Q-parity-based correction), and EDC checking. The core includes sufficient memory to hold two sectors of data.

Input data come from the internal bus and output data go out via the internal bus along a single line each, but the bus bridge logic sets up branches for the register access port and stream data port.

The stream data from the CD-DSP are transferred via the serial sound interface to the stream data input control block. They are then subjected to descrambling, ECC correction, and EDC checking as they pass through the CD-ROM decoder. After these processes, data from one sector are obtained. The data are subsequently transferred to the stream-data buffer via the stream-data output control block. Stream data can be transferred by either the direct memory access controller or the CPU.

Figure 24.3 is a block diagram of the bus-bridge logic.

Since the input stream is transferred over the serial sound interface, transfer is relatively slow. On the other hand, data from the output stream can be transferred at high speeds because they are already in the core of the CD-ROM decoder. Since the data for output are buffered in SDRAM or other memory, they must be transferred at high speeds in order to reduce the busy rate of the SDRAM. For this reason, the data for the output stream are read out before the CD-ROM decoder receives an output stream data read request from the internal bus. This allows the accumulation of streaming data in the registers of the bus bridge, so that the data are ready for immediate output to the internal bus upon a request from the internal bus. Accordingly, the reception of a request to read from registers other than the stream-data registers after the stream data has already been read out and stored in the register of the bus bridge is possible. To cope with this, the CD-ROM decoder is provided with separate intermediary registers for the output stream-data register and the other registers.

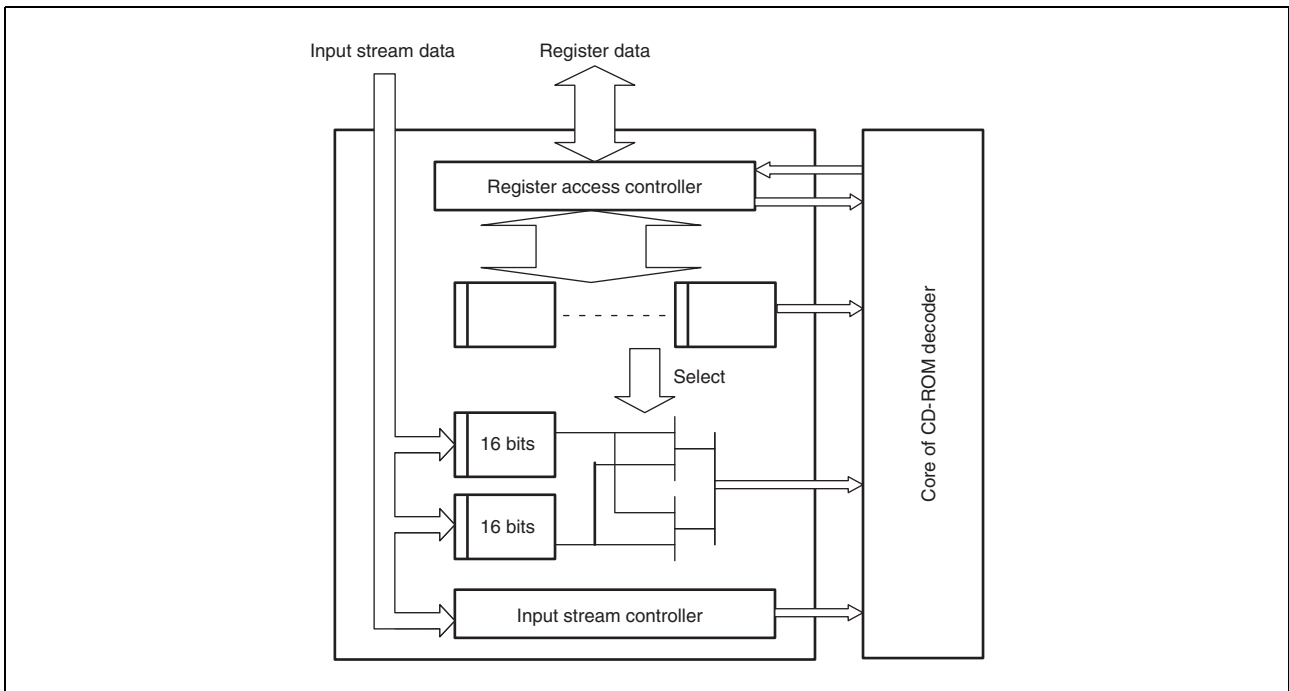


**Figure 24.3 Schematic Diagram of the Bus Bridge**

Figure 24.4 is a schematic diagram of the stream-data input control block. The stream-data input controller contains logic that controls the stream of input data and a register that is used to change the control mode of the CD-ROM decoder.

The serial sound interface mode used to transfer the stream data may affect the order (through the endian setting) or lead to padding before the data is transferred. To handle the different arrangements of data appropriately, the stream-data input control block includes a register for changing the operating mode and generates signals to control the core of the CD-ROM decoder.

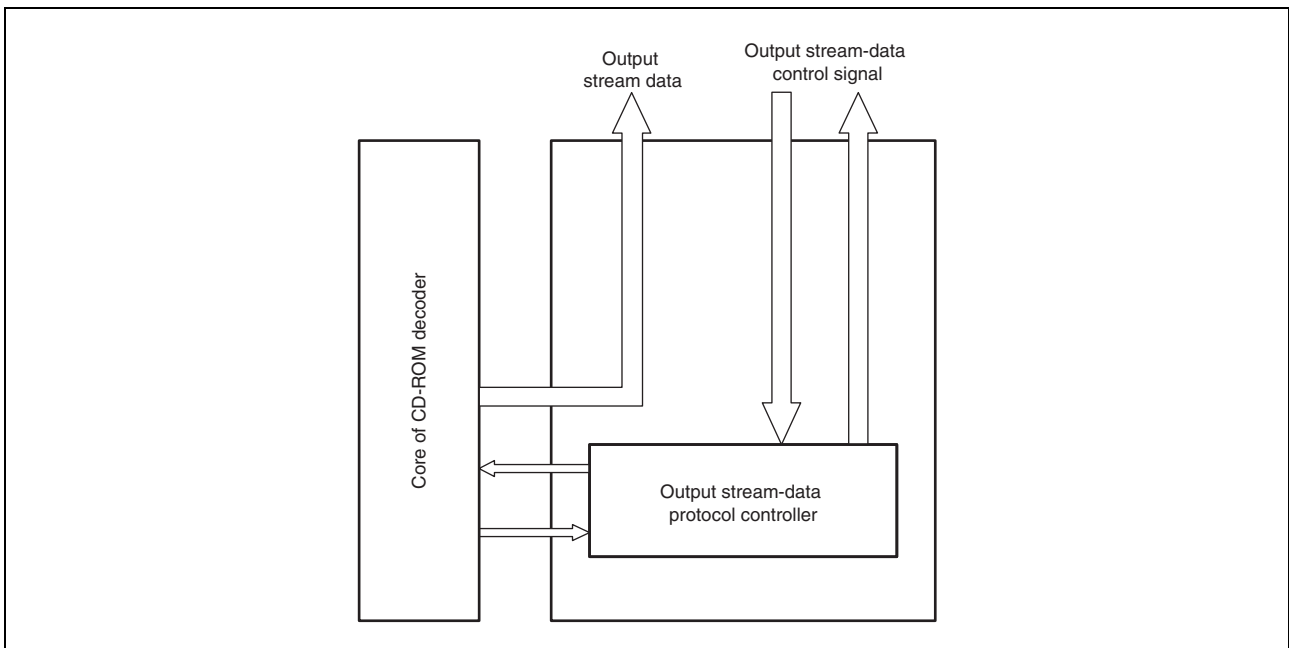
The data holding registers for the input stream consists of two 16-bit registers. The data holding registers are controlled according to the mode set in the control register. For example, controlling the order in which 16-bit data is supplied to the core of the CD-ROM decoder (sending the second 16-bytes first or vice versa). It is also possible to stop the supply of padding data to the core of the CD-ROM decoder.



**Figure 24.4 Schematic Diagram of the Stream-Data Input Control Block**

Figure 24.5 is a schematic diagram of the stream-data output control block.

On recognizing that one sector of CD-ROM data is ready in the core of the CD-ROM decoder, this block ensures that the output stream-data register in the bus bridge section is empty and then starts to acquire the data for output from the core of the CD-ROM decoder.



**Figure 24.5 Schematic Diagram of the Stream-Data Output Control Block**

This block has functions related to interrupts and direct memory access controller activation control such as suspending and masking of interrupts, turning interrupt flags off after they are read, asserting the activation signal to the direct

memory access controller, and negating the activation signal according to the detected amount of data that has been transferred.

### 24.3 Register Descriptions

This module has the following registers.

**Table 24.1 Register Configuration**

Name	Abbreviation	R/W	Initial Value	Address	Access Size
Enable control register	CROMEN	R/W	H'00	0xE8005000	8
Sync code-based synchronization control register	CROMSY0	R/W	H'89	0xE8005001	8
Decoding mode control register	CROMCTL0	R/W	H'82	0xE8005002	8
EDC/ECC check control register	CROMCTL1	R/W	H'D1	0xE8005003	8
Automatic decoding stop control register	CROMCTL3	R/W	H'00	0xE8005005	8
Decoding option setting control register	CROMCTL4	R/W	H'00	0xE8005006	8
HEAD20 to HEAD22 representation control register	CROMCTL5	R/W	H'00	0xE8005007	8
Sync code status register	CROMST0	R	H'00	0xE8005008	8
Post-ECC header error status register	CROMST1	R	H'00	0xE8005009	8
Post-ECC subheader error status register	CROMST3	R	H'00	0xE800500B	8
Header/subheader validity check status register	CROMST4	R	H'00	0xE800500C	8
Mode determination and link sector detection status register	CROMST5	R	H'00	0xE800500D	8
ECC/EDC error status register	CROMST6	R	H'00	0xE800500E	8
Buffer status register	CBUFST0	R	H'00	0xE8005014	8
Decoding stoppage source status register	CBUFST1	R	H'00	0xE8005015	8
Buffer overflow status register	CBUFST2	R	H'00	0xE8005016	8
Pre-ECC correction header: minutes data register	HEAD00	R	H'00	0xE8005018	8
Pre-ECC correction header: seconds data register	HEAD01	R	H'00	0xE8005019	8
Pre-ECC correction header: frames (1/75 second) data register	HEAD02	R	H'00	0xE800501A	8
Pre-ECC correction header: mode data register	HEAD03	R	H'00	0xE800501B	8
Pre-ECC correction subheader: file number (byte 16) data register	SHEAD00	R	H'00	0xE800501C	8
Pre-ECC correction subheader: channel number (byte 17) data register	SHEAD01	R	H'00	0xE800501D	8
Pre-ECC correction subheader: sub-mode (byte 18) data register	SHEAD02	R	H'00	0xE800501E	8
Pre-ECC correction subheader: data type (byte 19) data register	SHEAD03	R	H'00	0xE800501F	8
Pre-ECC correction subheader: file number (byte 20) data register	SHEAD04	R	H'00	0xE8005020	8
Pre-ECC correction subheader: channel number (byte 21) data register	SHEAD05	R	H'00	0xE8005021	8
Pre-ECC correction subheader: sub-mode (byte 22) data register	SHEAD06	R	H'00	0xE8005022	8
Pre-ECC correction subheader: data type (byte 23) data register	SHEAD07	R	H'00	0xE8005023	8
Post-ECC correction header: minutes data register	HEAD20	R	H'00	0xE8005024	8
Post-ECC correction header: seconds data register	HEAD21	R	H'00	0xE8005025	8
Post-ECC correction header: frames (1/75 second) data register	HEAD22	R	H'00	0xE8005026	8
Post-ECC correction header: mode data register	HEAD23	R	H'00	0xE8005027	8
Post-ECC correction subheader: file number (byte 16) data register	SHEAD20	R	H'00	0xE8005028	8
Post-ECC correction subheader: channel number (byte 17) data register	SHEAD21	R	H'00	0xE8005029	8
Post-ECC correction subheader: sub-mode (byte 18) data register	SHEAD22	R	H'00	0xE800502A	8
Post-ECC correction subheader: data type (byte 19) data register	SHEAD23	R	H'00	0xE800502B	8

**Table 24.1 Register Configuration**

Name	Abbreviation	R/W	Initial Value	Address	Access Size
Post-ECC correction subheader: file number (byte 20) data register	SHEAD24	R	H'00	0xE800502C	8
Post-ECC correction subheader: channel number (byte 21) data register	SHEAD25	R	H'00	0xE800502D	8
Post-ECC correction subheader: sub-mode (byte 22) data register	SHEAD26	R	H'00	0xE800502E	8
Post-ECC correction subheader: data type (byte 23) data register	SHEAD27	R	H'00	0xE800502F	8
Automatic buffering setting control register	CBUFCTL0	R/W	H'04	0xE8005040	8
Automatic buffering start sector setting: minutes control register	CBUFCTL1	R/W	H'00	0xE8005041	8
Automatic buffering start sector setting: seconds control register	CBUFCTL2	R/W	H'00	0xE8005042	8
Automatic buffering start sector setting: frames control register	CBUFCTL3	R/W	H'00	0xE8005043	8
ISY interrupt source mask control register	CROMST0M	R/W	H'00	0xE8005045	8
CD-ROM decoder reset control register	ROMDECRST	R/W	H'00	0xE8005100	8
CD-ROM decoder reset status register	RSTSTAT	R	H'00	0xE8005101	8
Serial sound interface data control register	SSI	R/W	H'18	0xE8005102	8
Interrupt flag register	INTHOLD	R/W	H'00	0xE8005108	8
Interrupt source mask control register	INHINT	R/W	H'00	0xE8005109	8
CD-ROM decoder stream data input register	STRMDIN0	R/W	H'0000	0xE8005200	R: 16 W: 16, 32
CD-ROM decoder stream data input register	STRMDIN2	R/W	H'0000	0xE8005202	16
CD-ROM decoder stream data output register	STRMDOUT0	R	H'0000	0xE8005204	16

### 24.3.1 Enable Control Register (CROMEN)

The enable control register (CROMEN) enables subcode processing and CD-ROM decoding, and stops CD-ROM decoding forcibly.

Bit:	7	6	5	4	3	2	1	0
	SUBC_	CROM_	CROM_	-	-	-	-	-
	EN	EN	STP					
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	SUBC_EN	0	R/W	Subcode Processing Enable This bit should be set and cleared simultaneously with CROM_EN. It is automatically cleared when decoding is automatically stopped due to an abnormal condition or when CROM_STP = 1
6	CROM_EN	0	R/W	CD-ROM Decoding Enable When this bit is set to 1, CD-ROM decoding starts after detection of a valid sync code. When the bit is cleared to 0, decoding stops on completion of the processing for the sector currently being decoded. This bit is automatically cleared to 0 when the automatic decode-stopping function works or when CROM_STP = 1.
5	CROM_STP	0	R/W	Forcible Stop of CD-ROM Decoding When this bit is set to 1, CD-ROM decoding is stopped immediately and the SUBC_EN and CROM_EN bits are automatically reset to 0. Before decoding can resume, this bit must be cleared to 0.
4 to 0	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.

### 24.3.2 Sync Code-Based Synchronization Control Register (CROMSY0)

The sync code-based synchronization control register (CROMSY0) selects the sync code maintenance function.

Bit:	7	6	5	4	3	2	1	0
	SY AUT	SY IEN	SY DEN	-	-	-	-	-
Initial value:	1	0	0	0	1	0	0	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	SY_AUT	1	R/W	Automatic CD-ROM Sync Code Maintenance Mode When this bit is set to 1, automatic sync maintenance (insertion of sync codes) is applied to obtain the CD-ROM sync codes. While this bit is set, the settings of the SY_IEN and SY_DEN bits are invalid.
6	SY_IEN	0	R/W	Internal Sync Signal Enable Enables the internal sync signal that is produced by the counter in the CD-ROM decoder. When this bit is set to 1 while SY_AUT = 0, synchronization of the CD-ROM data is in interpolated mode, i.e. driven by the internal counter.
5	SY_DEN	0	R/W	Synchronization with External Sync Code Selects constant monitoring for the sync code in the input data and bases synchronization solely on detection of the code, regardless of the value of the internal counter. The setting of this bit is valid when SY_AUT = 0.
4	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
3	—	1	R/W	Reserved This bit is always read as 1. The write value should always be 1.
2, 1	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
0	—	1	R/W	Reserved This bit is always read as 1. The write value should always be 1.

**Table 24.2 Register Settings for Sync Code Maintenance Function**

SY_AUT	SY_IEN	SY_DEN	Operating Mode
1	—	—	Automatic sync maintenance mode
0	0	1	External sync mode
0	1	0	Interpolated sync mode
0	1	1	Interpolated sync plus external sync mode
0	0	0	Setting prohibited

### 24.3.3 Decoding Mode Control Register (CROMCTL0)

The decoding mode control register (CROMCTL0) enables/disables the various functions, selects criteria for mode or form determination, and specifies the sector type. The setting of this register becomes valid at the sector-to-sector transition.

Bit:	7	6	5	4	3	2	1	0
	MD_DESC	-	MD_AUTO	MD_AUTOS1	MD_AUTOS2	MD_SEC[2:0]		
Initial value:	1	0	0	0	0	0	1	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	MD_DESC	1	R/W	Descrambling Function ON/OFF 0: Disables descrambling function 1: Enables descrambling function
6	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
5	MD_AUTO	0	R/W	Automatic Mode/Form Detection ON/OFF 0: OFF 1: ON Detectable formats are Mode 0, Mode 1, Mode 2 (non-XA), Mode 2 Form 1, and Mode 2 Form 2. If the mode and form cannot be detected, the mode and form of the previous sector is used. If the mode and form of the first sector after decoding starts is undetectable, the setting of the MD_SEC[2:0] bits is used as the initial value.
4	MD_AUTOS1	0	R/W	Criteria for Mode Determination when MD_AUTO = 1 0: Mode determination is made only when the sync code is detected 1: Mode determination is always made The setting of this bit is valid only when the MD_AUTO bit is 1. If the mode cannot be determined, the mode of the previous sector is used. When this bit is cleared to 0, mode determination is made only when the sync code is detected for the sector.
3	MD_AUTOS2	0	R/W	Criteria for Mode 2 Form Determination when MD_AUTO = 1 0: The sector is assumed to be non-XA if the two form code bytes in the subheader do not match 1: No determination of XA or non-XA for the sector. The first form byte is regarded as valid. However, the two form bytes are compared, and the result is reflected in a status bit. The setting of this bit is valid only when the MD_AUTO bit is 1.
2 to 0	MD_SEC[2:0]	010	R/W	Sector Type 000: Setting prohibited 001: Mode 0 010: Mode 1 011: Long (Mode 0, Mode 1, or Mode 2 with no EDC/ECC data) 100: Setting prohibited 101: Mode 2 Form 1 110: Mode 2 Form 2 111: Mode 2 with automatic form detection If the form cannot be determined when set to B'111, it is processed as Mode 2 not XA.

### 24.3.4 EDC/ECC Check Control Register (CROMCTL1)

The EDC/ECC check control register (CROMCTL1) controls EDC/ECC checking. The setting of this register becomes valid at the sector-to-sector transition.

Bit:	7	6	5	4	3	2	1	0
	M2F2 EDC	MD_DEC[2:0]			-	-	MD_PQREP[1:0]	
Initial value:	1	1	0	1	0	0	0	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	M2F2EDC	1	R/W	For Mode 2 Form 2, disables the EDC function for sectors where all bits of the EDC are 0. When this bit set to 1 and all bits of the EDC for a Mode 2 Form 2 sector are 0, an IERR interrupt is not generated even if the result of EDC checking is 'fail'.
6 to 4	MD_DEC[2:0]	101	R/W	EDC/ECC Checking Mode Select 000: No checking 001: EDC only 010: Q correction + EDC 011: P correction + EDC 100: QP correction + EDC 101: PQ correction + EDC 110: Setting prohibited 111: Setting prohibited
3, 2	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	MD_PQREP[1:0]	01	R/W	Number of Iterations of PQ or QP Correction Number of correction iterations when PQ- or QP- correction is specified by MD_DEC[2:0]. 00: Setting prohibited 01: One iteration 10: Two iterations 11: Three iterations



### 24.3.5 Automatic Decoding Stop Control Register (CROMCTL3)

The automatic decoding stop control register (CROMCTL3) is used to select abnormal conditions on which decoding will be automatically stopped. When decoding is stopped in response to any of the selected conditions, an IBUF interrupt is generated and the condition is indicated in the CBUFST1 register. The setting of this register becomes valid at the sector-to-sector transition.

Bit:	7	6	5	4	3	2	1	0
	STP ECC	STP EDC	-	STP MD	STP MIN	-	-	-
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	STP_ECC	0	R/W	When this bit is set to 1, decoding is stopped if an error is found to be not correctable by ECC correction.
6	STP_EDC	0	R/W	When this bit is set to 1, decoding is stopped if post-ECC correction EDC checking indicates an error.
5	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
4	STP_MD	0	R/W	When this bit is set to 1, decoding is stopped if the sector has a mode or form setting that does not match those of the immediately preceding sector.
3	STP_MIN	0	R/W	When this bit is set to 1, decoding is stopped if a non-sequential minutes, seconds, or frames (1 frame = 1/75 second) value is encountered.
2 to 0	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.

### 24.3.6 Decoding Option Setting Control Register (CROMCTL4)

The decoding option setting control register (CROMCTL4) enables/disables buffering control at link block detection, specifies the information indicated by the status register, and controls the ECC correction mode. The setting of this register becomes valid at the sector-to-sector transition.

Bit:	7	6	5	4	3	2	1	0
	-	LINK2	-	ER0SEL	NO_ECC	-	-	-
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	—	0	R/W	Reserved Either 0 or 1 can be written to this bit. When it is read, the value written will be read.
6	LINK2	0	R/W	Link Block Detection Condition 0: The block is regarded as a link block when either run-out 1 or 2 and both run-in 3 and 4 have been detected. 1: The block is regarded as a link block when two out of run-out 1 and 2 and “link” have been detected. The condition for setting of the LINK_ON bit in CROMST5 is decoding of the link sector.
5	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
4	ER0SEL	0	R/W	CD-ROM Data-Related Status Register Setting Condition 0: Information is on the sector being decoded. 1: Information is on the latest sector that has been buffered. This condition affects the information given by bits 5 to 0 in the CROMST0 register, bits 7 to 1 in the CROMST4 and CROMST5 registers, and HEAD00 to HEAD02.
3	NO_ECC	0	R/W	ECC correction mode when the result of the EDC check before ECC correction was ‘pass’ When this bit is set to 1, ECC correction is not performed if the result of pre-ECC correction EDC checking is a ‘pass’, regardless of the results of syndrome calculation.
2 to 0	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.

### 24.3.7 HEAD20 to HEAD22 Representation Control Register (CROMCTL5)

The HEAD20 to HEAD22 representation control register (CROMCTL5) specifies the representation mode for HEAD20 to HEAD22.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	MSF_LBA_SEL
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 1	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
0	MSF_LBA_SEL	0	R/W	HEAD20 to HEAD22 Representation Mode 0: Header MSF is represented in BCD (decimal) as is 1: Total sector number is represented in HEX (hexadecimal)

### 24.3.8 Sync Code Status Register (CROMST0)

The sync code status register (CROMST0) indicates various status information in sync code maintenance modes.

Bit:	7	6	5	4	3	2	1	0
	-	-	ST_SYIL	ST_SYNO	ST_BLKs	ST_BLKL	ST_SECS	ST_SECL
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 0	R	Reserved These bits are always read as 0 and cannot be modified.
5	ST_SYIL	0	R	Indicates that a sync code was detected at a position where the value in the word counter (used to measure intervals between sync codes) was not correct, but the sync code was ignored and not taken into account in synchronization. This bit is only valid in automatic sync maintenance mode and interpolated sync mode.
4	ST_SYNO	0	R	Indicates that a sync code has not been detected despite the word counter having reached the final value, and synchronization has been continued with the aid of an interpolated sync code. This bit is only valid in automatic sync maintenance mode and interpolated sync mode.
3	ST_BLKs	0	R	Indicates that a sync code was detected at a position where the value in the word counter was not correct, and the sync code was used in synchronization. This bit is only valid in automatic sync maintenance mode and external sync mode.
2	ST_BLKL	0	R	Indicates that a sync code has not been detected despite the word counter having reached the final value, and the period of the sector has been prolonged. This bit is only valid in external sync mode.
1	ST_SECS	0	R	Indicates that a sector has been processed as a short sector with the aid of interpolated sync codes. If this bit is set to 1, stop decoding immediately and retry the procedure starting from the sector prior to the sector currently being decoded.
0	ST_SECL	0	R	Indicates that a sector has been processed as a long sector with the aid of interpolated sync codes. If this bit is set to 1, stop decoding immediately and retry the procedure starting from two sectors prior to the sector currently being decoded.

### 24.3.9 Post-ECC Header Error Status Register (CROMST1)

The post-ECC header error status register (CROMST1) indicates error status in the post-ECC header.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	ER2_HEAD0	ER2_HEAD1	ER2_HEAD2	ER2_HEAD3
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	—	All 0	R	Reserved These bits are always read as 0 and cannot be modified.
3	ER2_HEAD0	0	R	Indicates an error in the minutes field of the header after ECC correction.
2	ER2_HEAD1	0	R	Indicates an error status in the seconds field of the header after ECC correction.
1	ER2_HEAD2	0	R	Indicates an error in the frames (1 frame = 1/75 second) field of the header after ECC correction.
0	ER2_HEAD3	0	R	Indicates an error in the mode field of the header after ECC correction.

### 24.3.10 Post-ECC Subheader Error Status Register (CROMST3)

The post-ECC subheader error status register (CROMST3) indicates error status in the post-ECC subheader.

Bit:	7	6	5	4	3	2	1	0
	ER2_SHEAD0	ER2_SHEAD1	ER2_SHEAD2	ER2_SHEAD3	ER2_SHEAD4	ER2_SHEAD5	ER2_SHEAD6	ER2_SHEAD7
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	ER2_SHEAD0	0	R	Indicates that the subheader (file number) still has an error after ECC correction. Indicates the error of the SHEAD20 register.
6	ER2_SHEAD1	0	R	Indicates that the subheader (channel number) still has an error after ECC correction. Indicates the error of the SHEAD21 register.
5	ER2_SHEAD2	0	R	Indicates that the subheader (sub-mode) still has an error after ECC correction. Indicates the error of the SHEAD22 register.
4	ER2_SHEAD3	0	R	Indicates that the subheader (data type) still has an error after ECC correction. Indicates the error of the SHEAD23 register.
3	ER2_SHEAD4	0	R	Indicates that the subheader (file number) still has an error after ECC correction. Indicates the error of the SHEAD24 register.
2	ER2_SHEAD5	0	R	Indicates that the subheader (channel number) still has an error after ECC correction. Indicates the error of the SHEAD25 register.
1	ER2_SHEAD6	0	R	Indicates that the subheader (sub-mode) still has an error after ECC correction. Indicates the error of the SHEAD26 register.
0	ER2_SHEAD7	0	R	Indicates that the subheader (data type) still has an error after ECC correction. Indicates the error of the SHEAD27 register.

### 24.3.11 Header/Subheader Validity Check Status Register (CROMST4)

The header/subheader validity check status register (CROMST4) indicates errors relating to the automatic mode determination or form determination for Mode 2.

Bit:	7	6	5	4	3	2	1	0
	NG_MD	NG_MDCMP1	NG_MDCMP2	NG_MDCMP3	NG_MDCMP4	NG_MDDEF	NG_MDTIM1	NG_MDTIM2
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	NG_MD	0	R	Indicates that the sector mode could not be determined according to the automatic mode determination criteria.
6	NG_MDCMP1	0	R	Indicates a mismatch between the file number bytes (bytes 16 and 20) during the form determination for Mode 2.
5	NG_MDCMP2	0	R	Indicates a mismatch between the channel number bytes (bytes 17 and 21) during the form determination for Mode 2.
4	NG_MDCMP3	0	R	Indicates a mismatch between the sub-mode bytes (bytes 18 and 22) during the form determination for Mode 2.
3	NG_MDCMP4	0	R	Indicates a mismatch between the data-type bytes (bytes 19 and 23) during the form determination for Mode 2.
2	NG_MDDEF	0	R	Indicates that the mode and form differ from those of the previous sector.
1	NG_MDTIM1	0	R	Indicates that the minutes, seconds, or frames (1 frame = 1/75 second) value is out of sequence. In the continuity check for the next and subsequent sectors, the updated values will be used.
0	NG_MDTIM2	0	R	Indicates that the minutes, seconds, or frames (1 frame = 1/75 second) value was not a BCD value. Specifically, this bit means that any half-byte was beyond the range for BCD (i.e. was A to F), HEAD01 was greater than H'59, or HEAD02 was greater than H'74. In the continuity check for the next and subsequent sectors, interpolated values will be used.

### 24.3.12 Mode Determination and Link Sector Detection Status Register (CROMST5)

The mode determination and link sector detection status register (CROMST5) indicates the result of automatic mode determination and link block detection.

Bit:	7	6	5	4	3	2	1	0
	ST_AMD[2:0]			ST_MDX	LINK_ON	LINK_DET	LINK_SDET	LINK_OUT1
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	ST_AMD[2:0]	000	R	<p>Result of Automatic Mode Determination</p> <p>These bits indicate the result of mode determination when the automatic mode determination function is used.</p> <p>000: Automatic mode determination function is not used</p> <p>001: Mode 0</p> <p>010: Mode 1</p> <p>011: —</p> <p>100: Mode 2 not XA</p> <p>101: Mode 2 Form 1</p> <p>110: Mode 2 Form 2</p> <p>111: —</p>
4	ST_MDX	0	R	<p>Indicates that, when the mode has been manually set rather than automatically determined, the mode setting disagrees with the mode as recognized by the logic. In this case, the manually set value takes priority.</p>
3	LINK_ON	0	R	<p>This bit is set to 1 when a link block was recognized in link block determination.</p> <p>For the criteria for link block determination, refer to the LINK2 bit in the CROMCTL4 register.</p>
2	LINK_DET	0	R	<p>Indicates that a link block (run-out 1 to run-in 4) was detected.</p> <p>Since detection is based on the data before ECC correction, LINK_DET may also be set to 1 if data erroneously happens to contain the same code as a link block.</p>
1	LINK_SDET	0	R	<p>Indicates that a link block was detected within seven sectors after the start of decoding.</p>
0	LINK_OUT1	0	R	<p>Indicates that the sector after ECC correction has been identified as a run-out 1 sector.</p> <p>This bit is only valid when an IERR interrupt is not generated (i.e. when ECC correction was successful).</p>

### 24.3.13 ECC/EDC Error Status Register (CROMST6)

The ECC/EDC error status register (CROMST6) indicates ECC processing error or EDC check error before/after ECC correction.

Bit:	7	6	5	4	3	2	1	0
	ST_ERR	-	ST_ECCABT	ST_ECCNG	ST_ECCP	ST_ECCQ	ST_EDC1	ST_EDC2
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	ST_ERR	0	R	Indicates that the decoded block after ECC correction contains any error (even in a single byte).
6	—	0	R	Reserved This bit is always read as 0 and cannot be modified.
5	ST_ECCABT	0	R	Indicates that ECC processing was discontinued. This bit is set to 1 when a transition from sector to sector occurs while ECC correction is in progress. This does not indicate a problem for ECC correction if the BUF_NG bit in the CBUFST2 register is 0 at the same time. Whether or not this is so depends on the timing of the sector transition.
4	ST_ECCNG	0	R	Indicates that error correction was not possible. This bit is also set to 1 on detection of a short sector.
3	ST_ECCP	0	R	Indicates that P-parity errors were not corrected in ECC correction. This bit is only valid when synchronization is normal (the sector is neither short nor long). This bit is set to 1 when the result of syndrome calculation for P parity is other than all 0s.
2	ST_ECCQ	0	R	Indicates that Q-parity errors were not corrected in ECC correction. This bit is only valid when synchronization is normal (the sector is neither short nor long). This bit is set to 1 when the result of syndrome calculation for Q parity is other than all 0s.
1	ST_EDC1	0	R	Indicates that the result of the EDC check before ECC correction was 'fail'. This bit is also set to 1 if a short sector is encountered while EDC is enabled.
0	ST_EDC2	0	R	Indicates that the result of the EDC check after ECC correction was 'fail'.

### 24.3.14 Buffer Status Register (CBUFST0)

The buffer status register (CBUFST0) indicates that the system is searching for the first sector to be buffered, or that buffering is in progress.

Bit:	7	6	5	4	3	2	1	0
	BUF_REF	BUF_ACT	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	BUF_REF	0	R	Indicates that the search for the first sector to be buffered is in progress. This bit is only valid when the automatic buffering function is used (CBUF_AUT = 1).
6	BUF_ACT	0	R	Indicates that buffering is in progress.
5 to 0	—	All 0	R	Reserved These bits are always read as 0 and cannot be modified.

### 24.3.15 Decoding Stoppage Source Status Register (CBUFST1)

The decoding stoppage source status register (CBUFST1) indicates that decoding/buffering has been stopped due to some errors.

A bit in this register can only be set when the corresponding bit in the CROMCTL3 register is set to 1.

Bit:	7	6	5	4	3	2	1	0
	BUF_ECC	BUF_EDC	-	BUF_MD	BUF_MIN	-	-	-
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	BUF_ECC	0	R	Indicates that decoding and buffering have been stopped because of an error that is not correctable by using the ECC.
6	BUF_EDC	0	R	Indicates that decoding and buffering have been stopped because the post-ECC correction EDC check indicated an error.
5	—	0	R	Reserved This bit is always read as 0 and cannot be modified.
4	BUF_MD	0	R	Indicates that decoding and buffering have been stopped because the current sector is in a mode or form differing from that of the previous sector.
3	BUF_MIN	0	R	Indicates that decoding and buffering have been stopped because a non-sequential minutes, seconds, or frames (1 frame = 1/75 second) value has been encountered.
2 to 0	—	All 0	R	Reserved These bits are always read as 0 and cannot be modified.

### 24.3.16 Buffer Overflow Status Register (CBUFST2)

The buffer overflow status register (CBUFST2) indicates that a sector-to-sector transition occurred before data transfer to the buffer is completed.

Bit:	7	6	5	4	3	2	1	0
	BUF_NG	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	BUF_NG	0	R	Indicates that a sector-to-sector transition has occurred before the data transfer to the buffer is completed. This bit is set to 1 when the data of a third sector are input while data for the output stream from the CD-ROM decoder remains unread. No interrupt is generated. Once this bit has been set to 1, its value will not recover unless it is reset by the LOGICRST bit in the ROMDECRST register.
6 to 0	—	All 0	R	Reserved These bits are always read as 0 and cannot be modified.



### 24.3.17 Pre-ECC Correction Header: Minutes Data Register (HEAD00)

The pre-ECC correction header: minutes data register (HEAD00) indicates the minutes value in the header before ECC correction.

Bit:	7	6	5	4	3	2	1	0
	HEAD00[7:0]							
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	HEAD00[7:0]	All 0	R	Minutes Value in Header before ECC Correction

### 24.3.18 Pre-ECC Correction Header: Seconds Data Register (HEAD01)

The pre-ECC correction header: seconds data register (HEAD01) indicates the seconds value in the header before ECC correction.

Bit:	7	6	5	4	3	2	1	0
	HEAD01[7:0]							
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	HEAD01[7:0]	All 0	R	Seconds Value in Header before ECC Correction

### 24.3.19 Pre-ECC Correction Header: Frames (1/75 Second) Data Register (HEAD02)

The pre-ECC correction header: frames (1/75 second) data register (HEAD02) indicates the frames value (1 frame = 1/75 second) in the header before ECC correction.

Bit:	7	6	5	4	3	2	1	0
	HEAD02[7:0]							
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	HEAD02[7:0]	All 0	R	Frames Value (1 frame = 1/75 second) in Header before ECC Correction

### 24.3.20 Pre-ECC Correction Header: Mode Data Register (HEAD03)

The pre-ECC correction header: mode data register (HEAD03) indicates the mode value in the header before ECC correction.

Bit:	7	6	5	4	3	2	1	0
	HEAD03[7:0]							
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	HEAD03[7:0]	All 0	R	Mode Value in Header before ECC Correction

### 24.3.21 Pre-ECC Correction Subheader: File Number (Byte 16) Data Register (SHEAD00)

The pre-ECC correction subheader: file number (byte 16) data register (SHEAD00) indicates the file number value in the subheader before ECC correction (byte 16).

Bit:	7	6	5	4	3	2	1	0
	SHEAD00[7:0]							
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	SHEAD00[7:0]	All 0	R	Indicates file number value in the subheader before ECC correction (byte 16). For sectors not in Mode 2, this register contains the byte of data at the corresponding position.

### 24.3.22 Pre-ECC Correction Subheader: Channel Number (Byte 17) Data Register (SHEAD01)

The pre-ECC correction subheader: channel number (byte 17) data register (SHEAD01) indicates the channel number value in the subheader before ECC correction (byte 17).

Bit:	7	6	5	4	3	2	1	0
	SHEAD01[7:0]							
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	SHEAD01[7:0]	All 0	R	Indicates channel number value in the subheader before ECC correction (byte 17). For sectors not in Mode 2, this register contains the byte of data at the corresponding position.

### 24.3.23 Pre-ECC Correction Subheader: Sub-Mode (Byte 18) Data Register (SHEAD02)

The pre-ECC correction subheader: sub-mode (byte 18) data register (SHEAD02) indicates the sub-mode value in the subheader before ECC correction (byte 18).

Bit:	7	6	5	4	3	2	1	0
	SHEAD02[7:0]							
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	SHEAD02[7:0]	All 0	R	Indicates sub-mode value in the subheader before ECC correction (byte 18). For sectors not in Mode 2, this register contains the byte of data at the corresponding position.

### 24.3.24 Pre-ECC Correction Subheader: Data Type (Byte 19) Data Register (SHEAD03)

The pre-ECC correction subheader: data type (byte 19) data register (SHEAD03) indicates the data type value in the subheader before ECC correction (byte 19).

Bit:	7	6	5	4	3	2	1	0
SHEAD03[7:0]								
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	SHEAD03[7:0]	All 0	R	Indicates data type value in the subheader before ECC correction (byte 19). For sectors not in Mode 2, this register contains the byte of data at the corresponding position.

### 24.3.25 Pre-ECC Correction Subheader: File Number (Byte 20) Data Register (SHEAD04)

The pre-ECC correction subheader: file number (byte 20) data register (SHEAD04) indicates the file number value in the subheader before ECC correction (byte 20).

Bit:	7	6	5	4	3	2	1	0
SHEAD04[7:0]								
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	SHEAD04[7:0]	All 0	R	Indicates file number value in the subheader before ECC correction (byte 20). For sectors not in Mode 2, this register contains the byte of data at the corresponding position.

### 24.3.26 Pre-ECC Correction Subheader: Channel Number (Byte 21) Data Register (SHEAD05)

The pre-ECC correction subheader: channel number (byte 21) data register (SHEAD05) indicates the channel number value in the subheader before ECC correction (byte 21).

Bit:	7	6	5	4	3	2	1	0
SHEAD05[7:0]								
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	SHEAD05[7:0]	All 0	R	Indicates channel number value in the subheader before ECC correction (byte 21). For sectors not in Mode 2, this register contains the byte of data at the corresponding position.

### 24.3.27 Pre-ECC Correction Subheader: Sub-Mode (Byte 22) Data Register (SHEAD06)

The pre-ECC correction subheader: sub-mode (byte 22) data register (SHEAD06) indicates the sub-mode value in the subheader before ECC correction (byte 22).

Bit:	7	6	5	4	3	2	1	0
	SHEAD06[7:0]							
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	SHEAD06[7:0]	All 0	R	Indicates sub-mode value in the subheader before ECC correction (Byte 22). For sectors not in Mode 2, this register contains the byte of data at the corresponding position.

### 24.3.28 Pre-ECC Correction Subheader: Data Type (Byte 23) Data Register (SHEAD07)

The pre-ECC correction subheader: data type (byte 23) data register (SHEAD07) indicates the data type value in the subheader before ECC correction (byte 23).

Bit:	7	6	5	4	3	2	1	0
	SHEAD07[7:0]							
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	SHEAD07[7:0]	All 0	R	Indicates data type value in the subheader before ECC correction (Byte 23). For sectors not in Mode 2, this register contains the byte of data at the corresponding position.

### 24.3.29 Post-ECC Correction Header: Minutes Data Register (HEAD20)

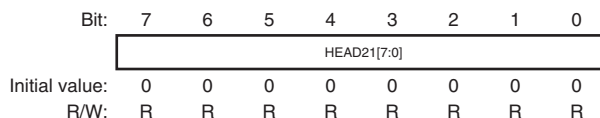
The post-ECC correction header: minutes data register (HEAD20) indicates the minutes value in the header after ECC correction.

Bit:	7	6	5	4	3	2	1	0
	HEAD20[7:0]							
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	HEAD20[7:0]	All 0	R	Indicates minutes value in the header after ECC correction. When MSF_LBA_SEL = 1, this register indicates the first byte (1/3) of the total number of sectors calculated from M, S, and F.

### 24.3.30 Post-ECC Correction Header: Seconds Data Register (HEAD21)

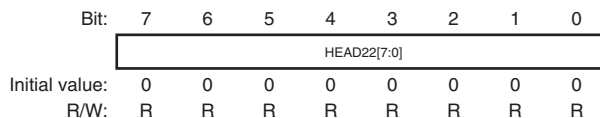
The post-ECC correction header: seconds data register (HEAD21) indicates the seconds value in the header after ECC correction.



Bit	Bit Name	Initial Value	R/W	Description
7 to 0	HEAD21[7:0]	All 0	R	Indicates seconds value in the header after ECC correction. When MSF_LBA_SEL = 1, this register indicates the second byte (2/3) of the total number of sectors calculated from M, S, and F.

### 24.3.31 Post-ECC Correction Header: Frames (1/75 Second) Data Register (HEAD22)

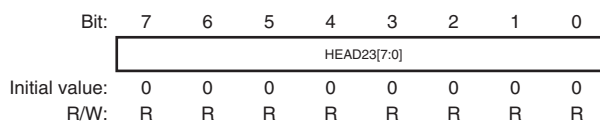
The post-ECC correction header: frames (1/75 second) data register (HEAD22) indicates the frames value (1 frame = 1/75 seconds) in the header after ECC correction.



Bit	Bit Name	Initial Value	R/W	Description
7 to 0	HEAD22[7:0]	All 0	R	Indicates frames value in the header after ECC correction. When MSF_LBA_SEL = 1, this register indicates the third byte (3/3) of the total number of sectors calculated from M, S, and F.

### 24.3.32 Post-ECC Correction Header: Mode Data Register (HEAD23)

The post-ECC correction header: mode data register (HEAD23) indicates the mode value in the header after ECC correction.



Bit	Bit Name	Initial Value	R/W	Description
7 to 0	HEAD23[7:0]	All 0	R	Indicates mode value in the header after ECC correction.

### 24.3.33 Post-ECC Correction Subheader: File Number (Byte 16) Data Register (SHEAD20)

The post-ECC correction subheader: file number (byte 16) data register (SHEAD20) indicates the file number value in the subheader after ECC correction (byte 16).

Bit:	7	6	5	4	3	2	1	0
	SHEAD20[7:0]							
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	SHEAD20[7:0]	All 0	R	Indicates file number value in the subheader after ECC correction (byte 16).

### 24.3.34 Post-ECC Correction Subheader: Channel Number (Byte 17) Data Register (SHEAD21)

The post-ECC correction subheader: channel number (byte 17) data register (SHEAD21) indicates the channel number value in the subheader after ECC correction (byte 17).

Bit:	7	6	5	4	3	2	1	0
	SHEAD21[7:0]							
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	SHEAD21[7:0]	All 0	R	Indicates channel number value in the subheader after ECC correction (byte 17).

### 24.3.35 Post-ECC Correction Subheader: Sub-Mode (Byte 18) Data Register (SHEAD22)

The post-ECC correction subheader: sub-mode (byte 18) data register (SHEAD22) indicates the sub-mode value in the subheader after ECC correction (byte 18).

Bit:	7	6	5	4	3	2	1	0
	SHEAD22[7:0]							
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	SHEAD22[7:0]	All 0	R	Indicates sub-mode value in the subheader after ECC correction (byte 18).

### 24.3.36 Post-ECC Correction Subheader: Data Type (Byte 19) Data Register (SHEAD23)

The post-ECC correction subheader: data type (byte 19) data register (SHEAD23) indicates the data type value in the subheader after ECC correction (byte 19).

Bit:	7	6	5	4	3	2	1	0
SHEAD23[7:0]								
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	SHEAD23[7:0]	All 0	R	Indicates data type value in the subheader after ECC correction (byte 19).

### 24.3.37 Post-ECC Correction Subheader: File Number (Byte 20) Data Register (SHEAD24)

The post-ECC correction subheader: file number (byte 20) data register (SHEAD24) indicates the file number value in the subheader after ECC correction (byte 20).

Bit:	7	6	5	4	3	2	1	0
SHEAD24[7:0]								
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	SHEAD24[7:0]	All 0	R	Indicates file number value in the subheader after ECC correction (byte 20).

### 24.3.38 Post-ECC Correction Subheader: Channel Number (Byte 21) Data Register (SHEAD25)

The post-ECC correction subheader: channel number (byte 21) data register (SHEAD25) indicates the channel number value in the subheader after ECC correction (byte 21).

Bit:	7	6	5	4	3	2	1	0
SHEAD25[7:0]								
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	SHEAD25[7:0]	All 0	R	Indicates channel number value in the subheader after ECC correction (byte 21).

### 24.3.39 Post-ECC Correction Subheader: Sub-Mode (Byte 22) Data Register (SHEAD26)

The post-ECC correction subheader: sub-mode (byte 22) data register (SHEAD26) indicates the sub-mode value in the subheader after ECC correction (byte 22).

Bit:	7	6	5	4	3	2	1	0
	SHEAD26[7:0]							
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	SHEAD26[7:0]	All 0	R	Indicates sub-mode value in the subheader after ECC correction (byte 22).

### 24.3.40 Post-ECC Correction Subheader: Data Type (Byte 23) Data Register (SHEAD27)

The post-ECC correction subheader: data type (byte 23) data register (SHEAD27) indicates the data type value in the subheader after ECC correction (byte 23).

Bit:	7	6	5	4	3	2	1	0
	SHEAD27[7:0]							
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	SHEAD27[7:0]	All 0	R	Indicates data type value in the subheader after ECC correction (byte 23).



## 24.3.41 Automatic Buffering Setting Control Register (CBUFCTL0)

Bit:	7	6	5	4	3	2	1	0
	CBUF_ AUT	CBUF_ EN	-	CBUF_MD[1:0]		CBUF_ TS	CBUF_ Q	-
Initial value:	0	0	0	0	0	1	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	CBUF_AUT	0	R/W	Automatic Buffering Function ON/OFF When this bit is to be set or cleared while CROM_EN = 1, CBUF_EN should also be set or cleared simultaneously. Otherwise, the validity of the status indications in CBUFST0, CBUFST1 and CBUFST2 cannot be guaranteed. 0: Automatic buffering is OFF. 1: Automatic buffering is ON.
6	CBUF_EN	0	R/W	Buffering to Buffer RAM Enable This bit turns on/off buffering in both automatic and manual buffering modes. In manual buffering mode, set this bit after generation of the ISEC interrupt. This bit is automatically reset when automatic buffering stops. 0: Buffering is OFF. 1: Buffering is ON.
5	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.
4, 3	CBUF_MD[1:0]	00	R/W	Start-sector detection mode when the automatic buffering function is in use 00: The header values for the previous and current sectors must be in sequence. 01: The header value detected in the current sector must be in sequence with the interpolated value. 10: A current sector with any header value is OK. 11: Start-sector detection is based on the interpolated value even if the current sector is not detected.
2	CBUF_TS	1	R/W	CBUFCTL1 to CBUFCTL3 Setting Mode 0: CBUFCTL1 to CBUFCTL3: BCD (in decimal) 1: Total number of sectors (in hexadecimal)
1	CBUF_Q	0	R/W	Q-channel code buffering data specification in the case of a CRC error in the Q-channel code 0: The values for the last sector for which the CRC returned a correct result are buffered. 1: The erroneous data is buffered as is. Note: Since subcodes are not input with this LSI, always set this bit to 1.
0	—	0	R/W	Reserved This bit is always read as 0. The write value should always be 0.

### 24.3.42 Automatic Buffering Start Sector Setting: Minutes Control Register (CBUFCTL1)

The automatic buffering start sector setting: minutes control register (CBUFCTL1) indicates the minutes value in the header for the first sector to be buffered.

Bit:	7	6	5	4	3	2	1	0
	M2F2 EDC	MD_DEC[2:0]			-	-	MD_POREP[1:0]	
Initial value:	1	1	0	1	0	0	0	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	BS_MIN[7:0]	All 0	R/W	Indicates setting of the minutes value in the header for the first sector to be buffered.

### 24.3.43 Automatic Buffering Start Sector Setting: Seconds Control Register (CBUFCTL2)

The automatic buffering start sector setting: seconds control register (CBUFCTL2) indicates the seconds value in the header for the first sector to be buffered.

Bit:	7	6	5	4	3	2	1	0
	BS_SEC[7:0]							
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	BS_SEC[7:0]	All 0	R/W	Indicates setting of the seconds value in the header for the first sector to be buffered.

### 24.3.44 Automatic Buffering Start Sector Setting: Frames Control Register (CBUFCTL3)

The automatic buffering start sector setting: frames control register (CBUFCTL3) indicates the frames (1 frame = 1/75 second) value in the header for the first sector to be buffered.

Bit:	7	6	5	4	3	2	1	0
	BS_FRM[7:0]							
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	BS_FRM[7:0]	All 0	R/W	Indicates setting of the frames (1 frame = 1/75 second) value in the header for the first sector to be buffered.

### 24.3.45 ISY Interrupt Source Mask Control Register (CROMST0M)

The ISY interrupt source mask control register (CROMST0M) masks the ISY interrupt sources specified by the bits in the sync code status register (CROMST0).

Bit:	7	6	5	4	3	2	1	0
	-	-	ST_SYILM	ST_SYNOM	ST_BLKSM	ST_BLKLM	ST_SECSM	ST_SECLM
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.
5	ST_SYILM	0	R/W	ISY interrupt ST_SYIL (bit 5 in the CROMST0 register) source mask
4	ST_SYNOM	0	R/W	ISY interrupt ST_SYNO (bit 4 in the CROMST0 register) source mask
3	ST_BLKSM	0	R/W	ISY interrupt ST_BLK (bit 3 in the CROMST0 register) source mask
2	ST_BLKLM	0	R/W	ISY interrupt ST_BLKL (bit 2 in the CROMST0 register) source mask
1	ST_SECSM	0	R/W	ISY interrupt ST_SECS (bit 1 in the CROMST0 register) source mask
0	ST_SECLM	0	R/W	ISY interrupt ST_SECL (bit 0 in the CROMST0 register) source mask

### 24.3.46 CD-ROM Decoder Reset Control Register (ROMDECRST)

The CD-ROM decoder reset control register (ROMDECRST) resets the random logic of the CD-ROM decoder and clears the RAM in the CD-ROM decoder.

Bit:	7	6	5	4	3	2	1	0
	LOGICRST	RAMRST	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	LOGICRST	0	R/W	CD-ROM Decoder Random Logic Reset Signal A reset signal is output while this bit is set to 1.
6	RAMRST	0	R/W	CD-ROM Decoder RAM Clearing Signal Refer to the RAMCLRST bit in the RSTSTAT register to confirm that RAM clearing is complete.
5 to 0	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.

Note: • Before setting LOGICRST to 1, make sure that the RAMRST bit is cleared to 0 and then write B'10000000 to this register.

### 24.3.47 CD-ROM Decoder Reset Status Register (RSTSTAT)

The CD-ROM decoder reset status register (RSTSTAT) indicates that the RAM in the CD-ROM decoder has been cleared.

Bit:	7	6	5	4	3	2	1	0
	RAM CLRST	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	RAMCLRST	0	R	This bit is set to 1 on completion of RAM clearing after the RAMRST bit in ROMDECRST is set to 1. The bit is cleared by writing a 0 to the RAMRST bit.
6 to 0	—	All 0	R	Reserved These bits are always read as 0 and cannot be modified.

### 24.3.48 Serial Sound Interface Data Control Register (SSI)

The serial sound interface data control register (SSI) provides various settings related to the data stream. For the operation corresponding to the setting of this register, refer to section 24.4.1, Endian Conversion for Data in the Input Stream.

Bit:	7	6	5	4	3	2	1	0
	BYTEND	BITEND	BUFEND0[1:0]		BUFEND1[1:0]		-	-
Initial value:	0	0	0	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	BYTEND	0	R/W	Specifies the endian of input data from the serial sound interface. When this bit is set to 1, byte 0 and byte 1 in STRMDIN0 are swapped. This is the same for STRMDIN2.
6	BITEND	0	R/W	Specifies treatment of the bit order of the input data from the serial sound interface. When this bit is set to 1, the bits within each byte are rearranged to place them in reverse order, bit 0 → bit 7 to bit 7 → bit 0.
5, 4	BUFEND0[1:0]	01	R/W	<p>These bits select whether to change the order of 16-bit units of data transferred from the serial sound interface or suppress the stream data. In the serial sound interface, either “padding mode” or “non-padding mode” is selectable. In non-padding mode, each 32 bits of data transferred from the serial sound interface are transferred as CD-ROM data. Since the CD-ROM decoder has two 16-bit input data registers, the order of the 16-bit data can be swapped within the 32 bits. On the other hand, in padding mode each 32 bits of data transferred from the serial sound interface includes padding. Since the padding has no meaning, it should be kept out of the input stream to the decoder. This suppression can be specified by the setting of this register.</p> <p>The CD-ROM decoder handles data as a stream of 16-bit data, and this register controls which 16-bit portion of each 32 bits of data transferred from the serial sound interface should be input first.</p> <p>00: The 16 bits of stream data that would otherwise be processed first is discarded.</p> <p>01: The higher-order 16 bits of each 32 bits of data received from the serial sound interface are placed first in the stream to the decoder.</p> <p>10: The lower-order 16 bits of each 32 bits of data received from the serial sound interface are placed first in the stream to the decoder.</p> <p>11: Setting prohibited</p>
3, 2	BUFEND1[1:0]	10	R/W	<p>These bits select whether to change the order of 16-bit units of data transferred from the serial sound interface or suppress the stream data. In the serial sound interface, either “padding mode” or “non-padding mode” is selectable. In non-padding mode, each 32 bits of data transferred from the serial sound interface are transferred as CD-ROM data. Since the CD-ROM decoder has two 16-bit input data registers, the order of the 16-bit data can be swapped within the 32 bits. On the other hand, in padding mode each 32 bits of data transferred from the serial sound interface includes padding. Since the padding has no meaning, it should be kept out of the input stream to the decoder. This suppression can be specified by the setting of this register.</p> <p>The CD-ROM decoder handles data as a stream of 16-bit data, and this register controls which 16-bit portion of each 32 bits of data transferred from the serial sound interface should be input second.</p> <p>00: The 16 bits of stream data that would otherwise be processed second is discarded.</p> <p>01: The higher-order 16 bits of each 32 bits of data received from the serial sound interface are placed second in the stream to the decoder.</p> <p>10: The higher-order 16 bits of each 32 bits of data received from the serial sound interface are placed second in the stream to the decoder.</p> <p>11: Setting prohibited</p>
1, 0	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.

### 24.3.49 Interrupt Flag Register (INTHOLD)

The interrupt flag register (INTHOLD) consists of various interrupt flags.

Bit:	7	6	5	4	3	2	1	0
	ISEC	ITARG	ISY	IERR	IBUF	IREADY	-	-
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	ISEC	0	R/W	ISEC Interrupt Flag Writing 0 to this bit is only possible after 1 has been read from it.
6	ITARG	0	R/W	ITARG Interrupt Flag Writing 0 to this bit is only possible after 1 has been read from it.
5	ISY	0	R/W	ISY Interrupt Flag Writing 0 to this bit is only possible after 1 has been read from it.
4	IERR	0	R/W	IERR Interrupt Flag Writing 0 to this bit is only possible after 1 has been read from it.
3	IBUF	0	R/W	IBUF Interrupt Flag Writing 0 to this bit is only possible after 1 has been read from it.
2	IREADY	0	R/W	IREADY Interrupt Flag Writing 0 to this bit is only possible after 1 has been read from it.
1, 0	—	All 0	R/W	Reserved These bits are always read as 0. The write value should always be 0.

### 24.3.50 Interrupt Source Mask Control Register (INHINT)

The interrupt source mask control register (INHINT) controls masking of various interrupt requests in the CD-ROM decoder.

Bit:	7	6	5	4	3	2	1	0
	INH ISEC	INH ITARG	INH ISY	INH IERR	INH IBUF	INH IREADY	PREINH REQDM	PREINH IREADY
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	INHISEC	0	R/W	ISEC Interrupt Mask When set to 1, inhibits ISEC interrupt requests.
6	INHITARG	0	R/W	ITARG Interrupt Mask When set to 1, inhibits ITARG interrupt requests.
5	INHISY	0	R/W	ISY Interrupt Mask When set to 1, inhibits ISY interrupt requests.
4	INHIERR	0	R/W	IERR Interrupt Mask When set to 1, inhibits IERR interrupt requests.
3	INHIBUF	0	R/W	IBUF Interrupt Mask When set to 1, inhibits IBUF interrupt requests.
2	INHIREADY	0	R/W	IREADY Interrupt Mask When set to 1, inhibits IREADY interrupt requests.
1	PREINHREQDM	0	R/W	Inhibits setting of the DMA-transfer-request interrupt source flag for the output data stream. When this bit is set to 1, the DMA-transfer-request interrupt source is not retained.
0	PREINHIREADY	0	R/W	Inhibits setting of the IREADY interrupt flag. When this bit is set to 1, the IREADY interrupt source is not retained.

### 24.3.51 CD-ROM Decoder Stream Data Input Register (STRMDIN0)

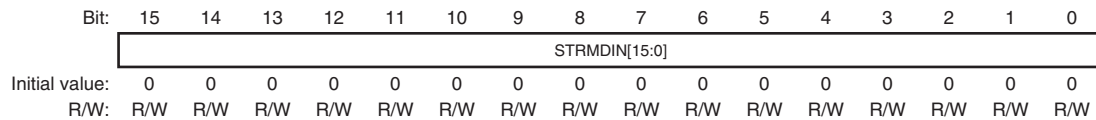
The CD-ROM decoder stream data input register (STRMDIN0) holds the higher 2 bytes (from MSB) of the 4 bytes of data that is to be input to the CD-ROM decoder.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	STRMDIN[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	STRMDIN[31:16]	All 0	R/W	Indicates the higher 2 bytes (from MSB) of the 4-bytes of data that is to be input to the CD-ROM decoder. The CD-ROM decoder has a 4-byte wide data window as a data input register to handle the data input to this register as a stream data. The amount of data for one sector is 2352 bytes.

### 24.3.52 CD-ROM Decoder Stream Data Input Register (STRMDIN2)

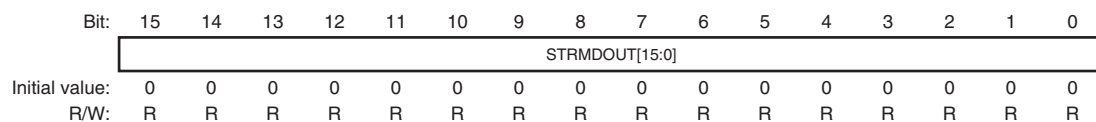
The CD-ROM decoder stream data input register (STRMDIN2) holds the lower 2 bytes (from LSB) of the 4 bytes of data that is to be input to the CD-ROM decoder.



Bit	Bit Name	Initial Value	R/W	Description
15 to 0	STRMDIN[15:0]	All 0	R/W	Indicates the lower 2 bytes (from LSB) of the 4-bytes of data that is to be input to the CD-ROM decoder. The CD-ROM decoder has a 4-byte wide data window as a data input register to handle the data input to this register as a stream data. The amount of data for one sector is 2352 bytes.

### 24.3.53 CD-ROM Decoder Stream Data Output Register (STRMDOUT0)

The CD-ROM decoder stream data output register (STRMDOUT0) holds 2 bytes that is to be output from the CD-ROM decoder.



Bit	Bit Name	Initial Value	R/W	Description
15 to 0	STRMDOUT [15:0]	H'0000	R	Indicates 2 bytes that are to be output from the CD-ROM decoder. The CD-ROM decoder has a 2-byte wide data window or set of registers for the output of decoded data. Every time the relevant register is accessed, further data of access size are output sequentially in the output format that is separately defined. The amount of data for one sector is 2768 bytes. Always read 2768 bytes.



## 24.4 Operation

### 24.4.1 Endian Conversion for Data in the Input Stream

Stream data must be input to the core of the CD-ROM decoder in order according to the CD-ROM data format specifications. In some systems, however, the order of the data from the serial sound interface may have to be changed or the data will have been padded before transfer. To cope with this, the stream data input control section is capable of swapping the order of the data and preventing the input of padding data to the core of the CD-ROM decoder. These functions are controlled through the serial sound interface data control register (SSI).

Figure 24.6 shows a case where the upper and lower 16 bits of the data, consisting of padding data plus the first 2 bytes of sync code, that is, H'000000FF, are swapped (H'00FF0000) and input to the CD-ROM decoder as the stream data.

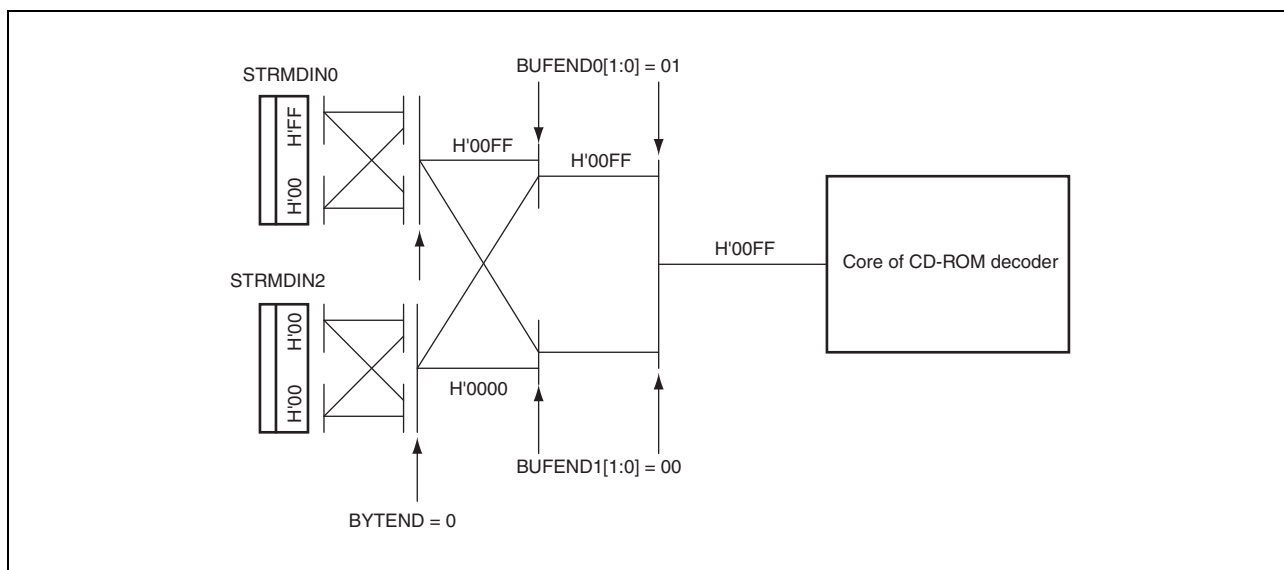


Figure 24.6 Example of Padded Stream Data Control by the SSI Register

Figure 24.7 shows a case of input stream data that has no padding (H'12345678). The upper and lower 16 bits of data are swapped (H'56781234) for input to the CD-ROM decoder.

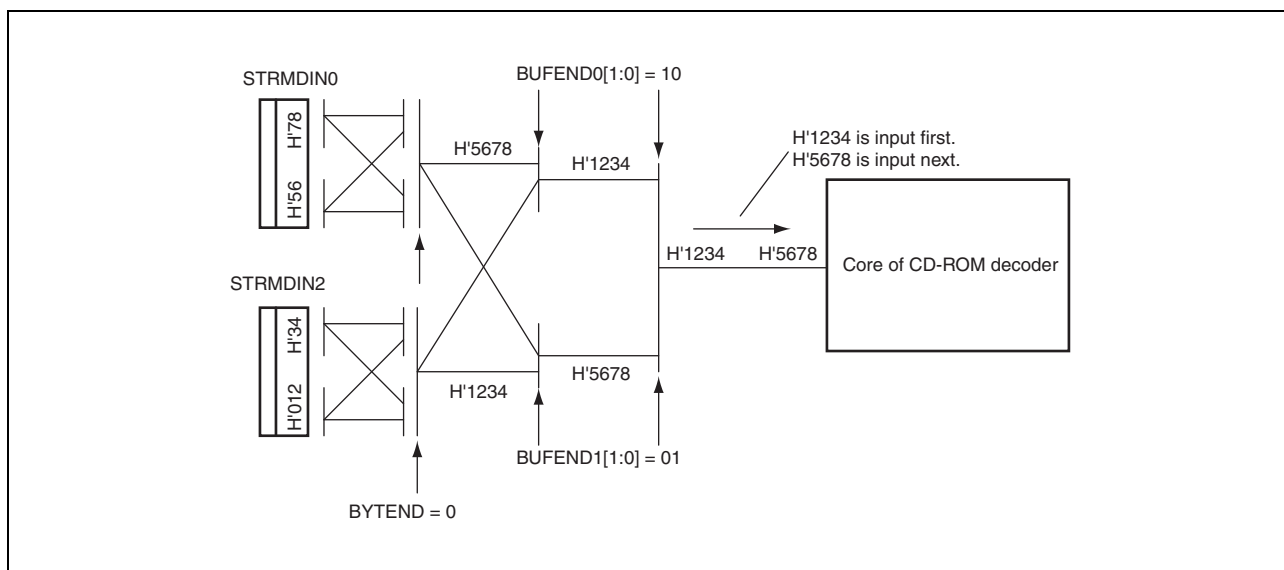


Figure 24.7 Example of Non-Padded Stream Data Control by the SSI Register

### 24.4.2 Sync Code Maintenance Function

Each sector of CD-ROM data consists of 2352 bytes starting with H'00FFFFFFFFFFFFFFFFF00 (sync code). However, a scratch on the disc or some other factor might lead to erroneous recognition of the sync code sequence at the wrong time. Conversely, a sync code might not be detected at a point where it should be detected. As a solution to these problems, the CD-ROM decoder of this LSI has a sync-code maintenance function, which operates to ignore sync codes detected at abnormal times and maintain the appearance of the sync code at the expected times when it is not actually detected on the disc.

The operating modes of the sync-code maintenance function are listed below. For details on the settings, refer to section 24.3.2, Sync Code-Based Synchronization Control Register (CROMSY0), and Table 24.2.

- Automatic sync maintenance mode
- External sync mode
- Interpolated sync mode
- Interpolated sync plus external sync mode

#### (1) Automatic Sync Maintenance Mode

In automatic sync maintenance mode, the sync code is ignored if detected within the one-sector (2352-byte) period. Furthermore, if a sync code is not detected at the point where a next sector should start, sync code maintenance is applied. If synchronization timing has changed, re-synchronization is performed at the point where a sync code is detected within 2352 bytes after the change.

Therefore, this mode is effective in rejecting abnormal sync patterns and following changes in synchronization timing. Note, however, that this mode cannot achieve synchronization with the first sector after a change to the synchronization timing.

Figure 24.8 shows operation in the case of normal sync-code detection, Figure 24.9 shows a case where a sync code is detected before a current one-sector period has elapsed, and Figure 24.10 shows the case where the actual sync code is only detected some time after a full one-sector period has elapsed.

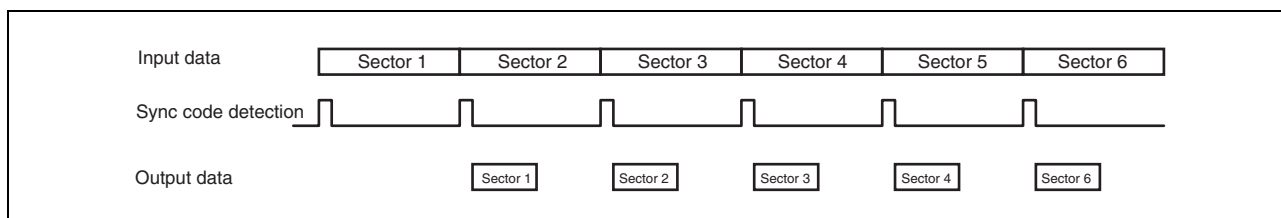


Figure 24.8 Operation in Automatic Sync Maintenance Mode (Normal Timing)

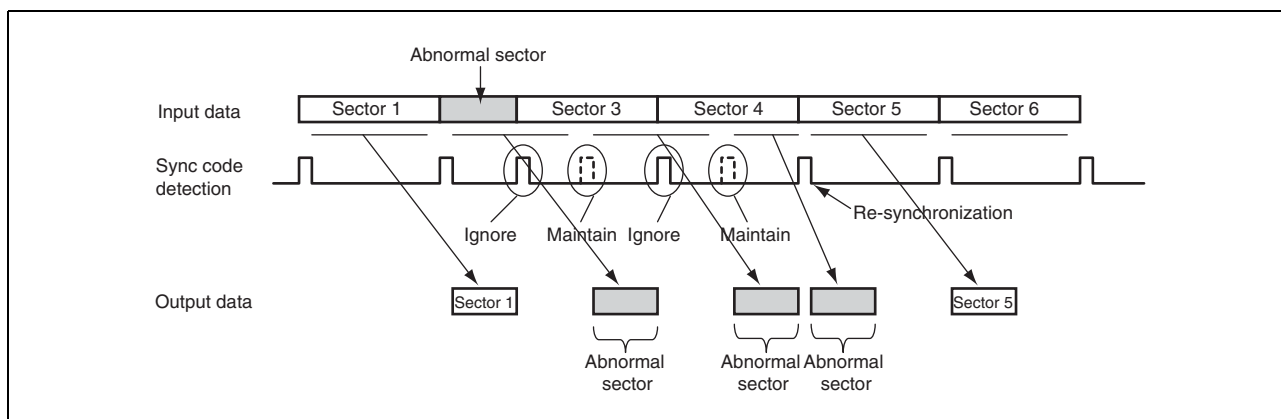
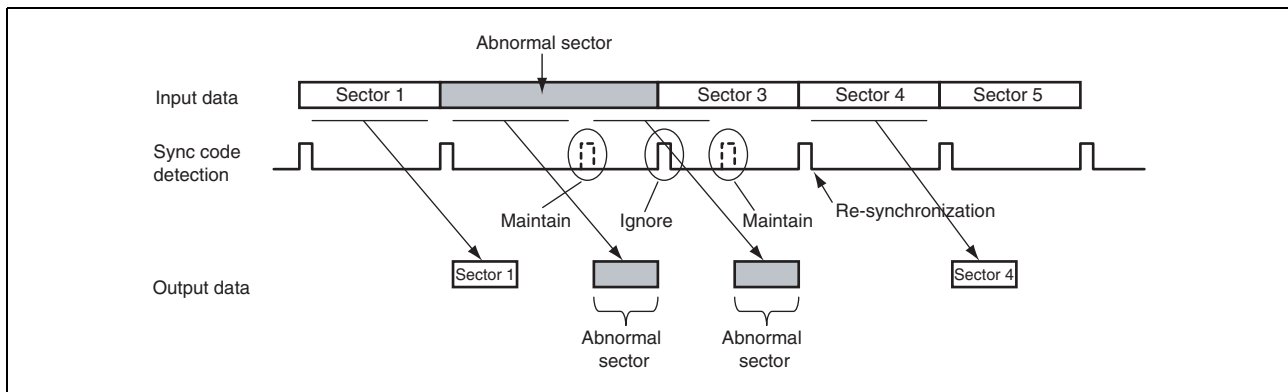


Figure 24.9 Operation in Automatic Sync Maintenance Mode (When an Abnormally Short Sector is Encountered)



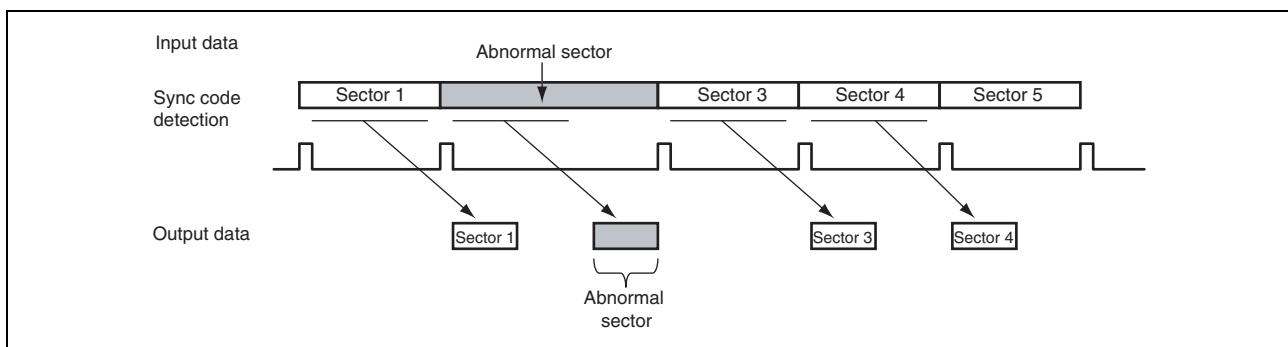
**Figure 24.10 Operation in Automatic Sync Maintenance Mode  
(When an Abnormally Long Sector is Encountered)**

### (2) External Sync Mode

In external sync mode, synchronization is always based on the sync codes in the incoming data. Even if the next sync code is not detected at the 2352nd byte, decoding does not proceed until the next sync code is detected.

Accordingly, this mode is effective in that it strictly follows the external synchronization timing. Note, however, that decoding will not be performed normally when the sync-code pattern is input with abnormal timing.

Figure 24.11 shows the operation in external sync mode.



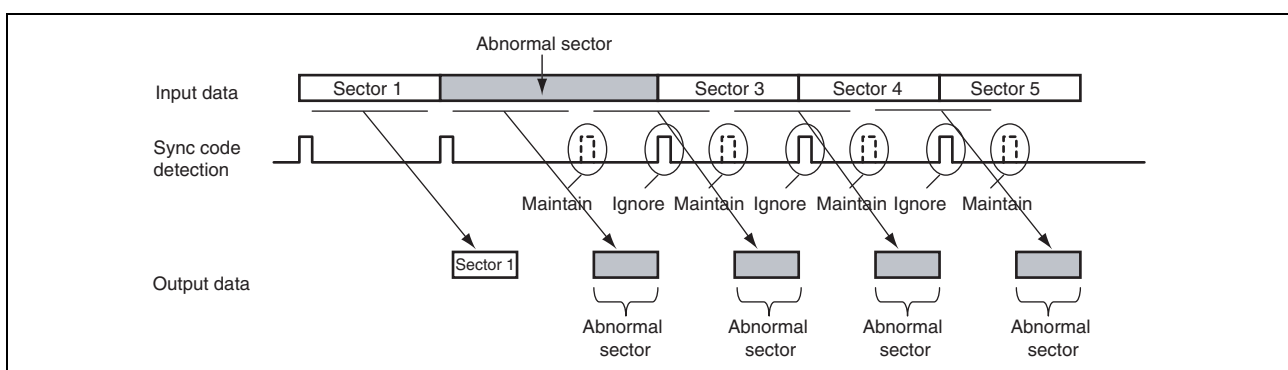
**Figure 24.11 Operation in External Sync Mode**

### (3) Interpolated Sync Mode

In interpolated sync mode, synchronization is always driven by the internal counter after a sync code pattern has been detected at the start of decoding. Accordingly, this mode is effective when the sync patterns have been damaged.

However, decoding becomes incorrect after a change to the synchronization timing, since the change in timing is not followed.

Figure 24.12 shows the operation in interpolated sync mode.



**Figure 24.12 Operation in Interpolated Sync Mode**

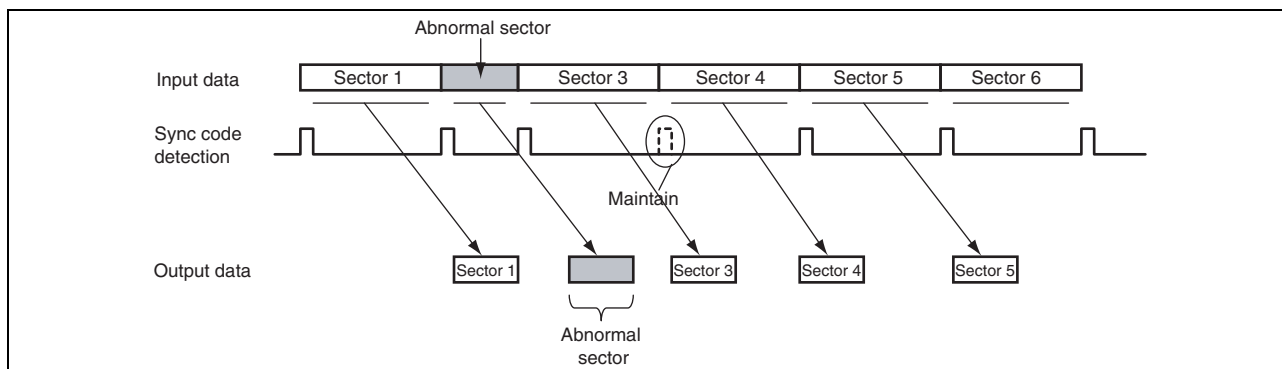
#### (4) Interpolated Sync Plus External Sync Mode

In interpolated sync plus external sync mode, synchronization is based on the detected sync code patterns as long as they are present, and if a sync pattern is not detected at the 2352nd byte, the sync code maintenance is applied.

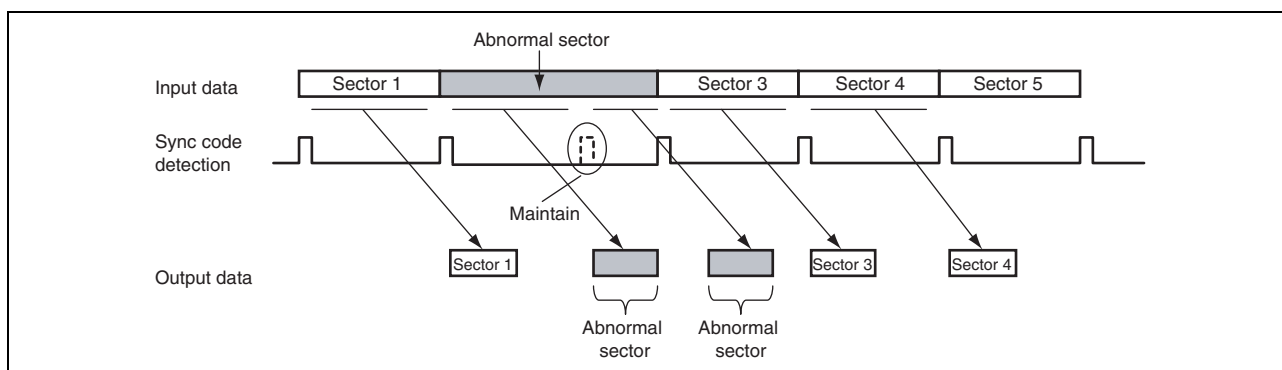
Synchronization in this mode is more quickly responsive to changes in synchronization timing than synchronization in the automatic sync maintenance mode.

However, decoding still becomes incorrect when a sync pattern is input with abnormal timing.

Figure 24.13 and Figure 24.14 show the operation in interpolated sync plus external sync mode in the cases of abnormally short and long sectors, respectively.



**Figure 24.13** Operation in Interpolated Sync Plus External Sync Mode (When an Abnormally Short Sector is Encountered)



**Figure 24.14** Operation in Interpolated Sync Plus External Sync Mode (When an Abnormally Long Sector is Encountered)

### 24.4.3 Error Correction

The CD-ROM decoder handles data in the formats containing information relevant to error correction, including the EDC, P parity, and Q parity. The CD-ROM decoder includes the following functions for use in error correction.

- Syndrome calculation
- ECC correction
- EDC checking

#### (1) Syndrome Calculation

After the data of a sector in Mode 1 or Form 1 of Mode 2 has been input, the ECC is used in correction if any error is detected (the result of syndrome calculation is non-zero). After correction, the results of syndrome operation for the corrected data are output to bits ST\_ECCP (P parity) and ST\_ECCQ (Q parity) in the CROMST6 register, respectively.

## (2) ECC correction and EDC Checking

For CD-ROM format data that contains EDC, P-parity, and Q-parity fields, the CD-ROM decoder performs EDC checking and ECC correction. Supported correction modes are P correction, Q correction, PQ correction (P correction followed by Q correction), and QP correction (Q correction followed by P correction). In PQ and QP correction modes, up to three iterations of correction are possible (the number of iterations is limited by the playback speed).

The EDC check is performed twice, before and after correction.

The mode of ECC correction and EDC checking is specified by bits MD\_DEC[2:0] in the CROMCTL1 register. When the PQ or QP correction mode is selected, the number of iterations of correction is specified by bits MD\_PQREP[1:0] in the CROMCTL1 register.

When the automatic mode/form detection function is in use, the sector mode determines whether or not ECC correction and EDC checking can be performed. For sectors in Mode 0 and Mode 2 (non-XA), which include neither parity bits nor EDC, ECC correction and EDC checking are not performed. For sectors in Form 2 of Mode 2, ECC correction is not performed.

### (a) ECC Correction

When ECC correction is in use and an error in a sector is identified as non-correctable, the CD-ROM decoder generates an IERR interrupt and sets the ST\_ECCNG bit of the CROMST6 register to 1. The CD-ROM detector also sets this bit to 1 on detecting a short sector.

While the NO\_ECC bit of the CROMCTL4 register is set to 1, a 'pass' result in pre-ECC correction EDC checking makes the CD-ROM decoder skip ECC correction, regardless of the results of the syndrome operation.

### (b) EDC Checking

When EDC checking is in use, checking is in line with the specified or detected sector mode and form, depending on whether or not automatic sector mode and form detection is selected.

The results of EDC checking before and after correction are reflected in the ST\_EDC1 and ST\_EDC2 bits of the CROMST6 register, respectively. If EDC checking after ECC correction indicates that an error remains, an IERR interrupt is generated.

## 24.4.4 Automatic Decoding Stop Function

Decoding can be stopped automatically in response to an error during the decoding of CD-ROM data.

The possible conditions for automatically stopping the decoding process are listed below. The applicable conditions are specified in the CROMCTL3 register.

- An error is found to be not correctable by ECC correction.
- Post-ECC correction EDC checking indicates that an error remains.
- A change of the sector mode or form
- A non-sequential MSF (minutes, seconds, frames (1/75 second)) value

When automatic stopping is set up and any of the above conditions is encountered in a certain sector, the decoding is stopped after the results of decoding for that sector have been output.

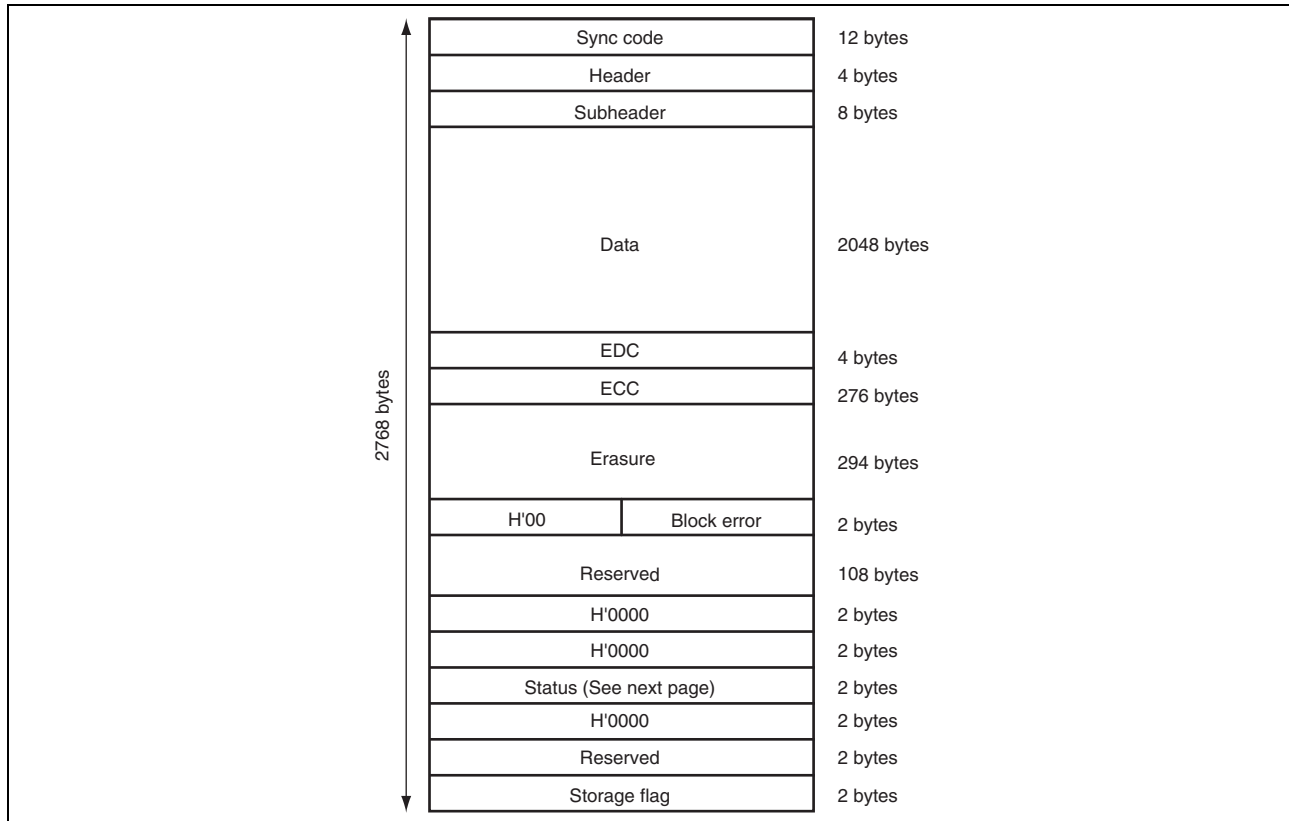
After decoding has been stopped in response to a condition specified in the CROMCTL3 register, the condition can be identified by reading the CBUFST1 register.

The CD-ROM decoder has buffer space for two sectors. If input of the data stream continues and the output stream of data is not read, the CD-ROM decoder stops at the point where the data of a third sector starts to be input. At this time, the BUF\_NG bit in the CBUFST2 register is set to 1, but no interrupt is generated. Once the BUF\_NG bit in the CBUFST2 register has been set to 1, recovery can only be accomplished by using the LOGICRST bit in the ROMDECRST register to reset the CD-ROM decoder function. When the LOGICRST bit in the ROMDECRST register is set to 1, a reset signal is output and any registers in which settings have been made are cleared to their initial values.

### 24.4.5 Buffering Format

Figure 24.15 shows the format of the output data stream produced by CD-ROM decoding.

A 2-byte-wide window register STRMDOUT0 is provided for the output. When this window register is accessed after decoding of a CD-ROM sector has finished, the bytes of data are output in order from the sync code.



**Figure 24.15 Output Data Stream Format**

The meanings of bits in the two-byte status field shown in Figure 24.15 are given below. The values of the non-assigned bits are undefined.

Status															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PERR	QERR	EDCE	—	—	—	—	—	SD	SY		FM[2:0]		HD	—	—

[Legend]

PERR: Indicates that a P-parity error remains.

QERR: Indicates that a Q-parity error remains.

EDCE: Indicates that a remaining error was detected in post-ECC correction EDC checking.

SD: Indicates that a short sector was encountered.

SY: Indicates that a sync code was interpolated.

FM: Indicates the data format.

001: Mode 0

010: Mode 1

011: Long (format with no EDC and ECC)

100: Mode 2 (non-XA)

101: Mode 2 Form 1

110: Mode 2 Form 2

HD: Header continuity (minutes, seconds, and frames (1 frame = 1/75 second) are non-sequential)

The value of the storage flag field in Figure 24.15 is incremented every time the data for one sector are output. The value starts at H'0000 and wraps back around to H'0000 after it reaches H'FFFF. Note that the upper byte and lower byte in the storage flag are swapped.

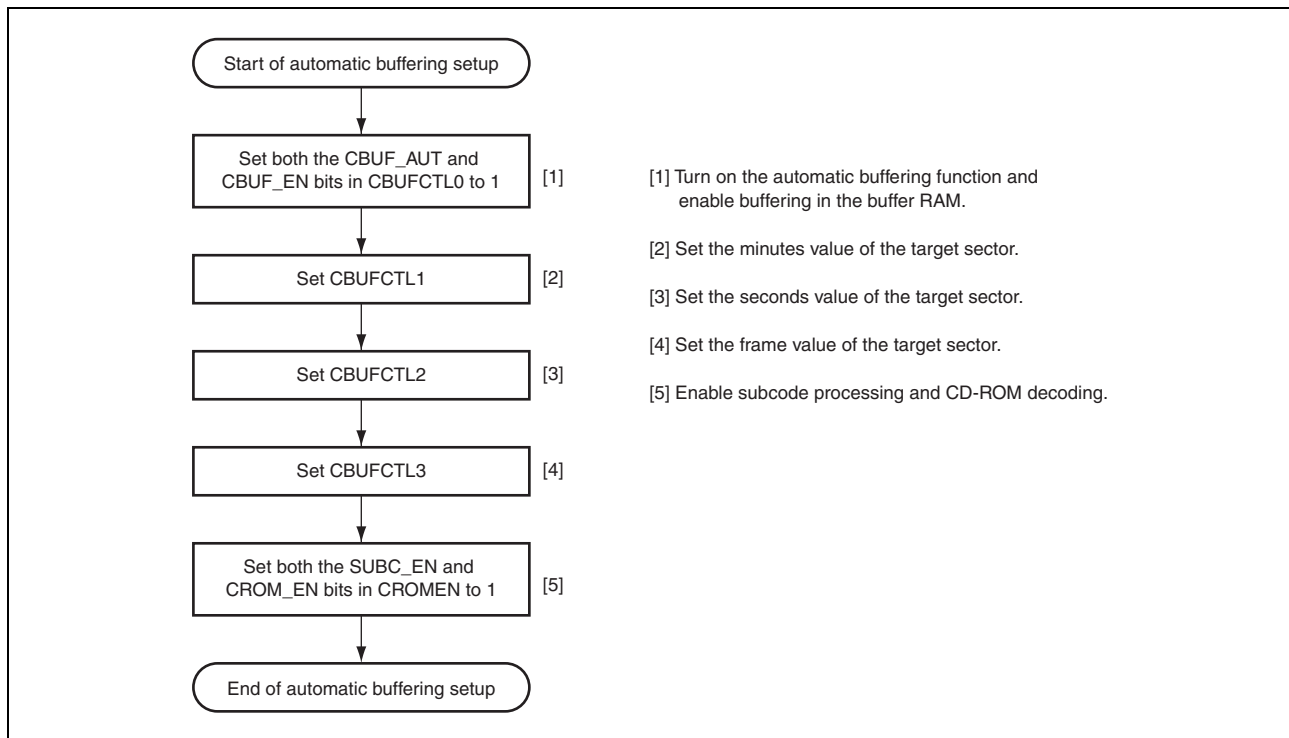
### 24.4.6 Target-Sector Buffering Function

In the CD-ROM decoder, the sector for output can be designated in two ways: automatic buffering, where the CD-ROM decoder itself detects the presence of target sector that has been set beforehand, and manual buffering, where the target sector for output is designated by software and the software also recognizes the sectors buffered in the CD-ROM decoder.

The following describes the procedures for setting the registers in the CD-ROM decoder to set up automatic or manual buffering.

#### (1) Setting Up Automatic Buffering

Figure 24.16 shows an example of setting up the automatic buffering. Set the relevant CD-ROM decoder registers and start input of the data stream; the CD-ROM decoder then detects the target sector and starts the output of the stream data.



**Figure 24.16** Example of Setting Up Automatic Buffering

## (2) Setting Up Manual Buffering

Figure 24.17 shows an example of setting up manual buffering. Each time an ISEC interrupt is generated, the software checks whether or not the sector is the target sector and starts buffering when the target sector is found.

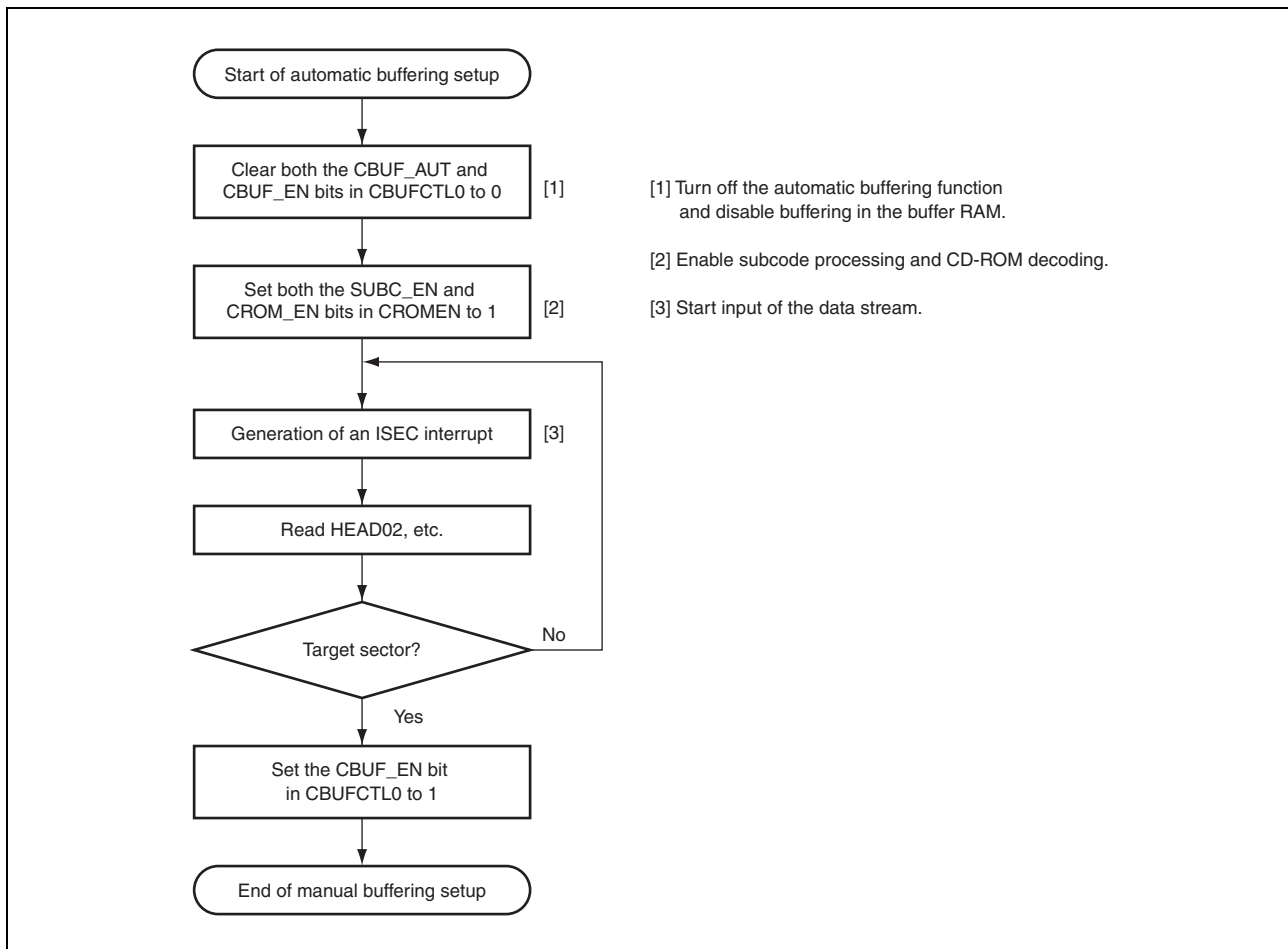


Figure 24.17 Example of Setting Up Manual Buffering



## 24.5 Interrupt Sources

### 24.5.1 Interrupt and DMA Transfer Request Signals

Table 24.3 lists the interrupt signals and DMA transfer request signal generated by the CD-ROM decoder, along with the meanings and the modules to which the signals are connected.

**Table 24.3 Interrupt and DMA Transfer Request Signals**

Name	Condition	Connected To
ISEC	Transition from sector to sector	Interrupt controller
ITARG	Access to a CD-ROM sector that is not the expected target sector	Interrupt controller
ISY	A sync code from the CD-ROM with abnormal timing	Interrupt controller
IERR	An error that was not correctable by ECC correction or an error indicated by EDC checking after ECC correction	Interrupt controller
IBUF	State changes in data transfer to the buffer	Interrupt controller
IREADY	Request for data transfer to the buffer for CD-ROM	Interrupt controller
DMA transfer request	Request for data transfer to the buffer for CD-ROM	Direct memory access controller

#### (1) ISEC Interrupt

This interrupt is generated when the sync code indicates a transition from sector to sector.

#### (2) ITARG Interrupt

This interrupt is generated when the stream data transferred from the CD-DSP is not the data of the target sector. The CD-ROM decoder checks the time data in the subcode. In correct operation, data transfer is expected to start slightly before the target sector. An ITARG interrupt is generated in the following cases.

- When data of a sector preceding the target sector by quite a few sectors have been transferred
- When data of a sector that comes after the target sector have been transferred

For the generation of this interrupt, ITARG is detected from the subcode. However, this interrupt has no meaning in this LSI because CD-ROM data are transferred from the serial sound interface.

#### (3) ISY Interrupt

This interrupt can be generated in the following cases.

- When a sync code was detected at a position where the value in the word counter (counter for checking sync code intervals) was not correct and the sync code was ignored
- When a sync code has not been detected although the word counter has reached the final value and a sync code has been interpolated (for sync maintenance)
- When a sync code was detected at a position where the value in the word counter (counter for checking sync code intervals) was not correct and the sync code was used in resynchronization
- When a sync code has not been detected although the word counter has reached the final value, so the period taken up by the sector has been prolonged
- When the sector has been processed as a short sector with the aid of interpolated sync codes
- When the sector has been processed as a long sector with the aid of interpolated sync codes

#### (4) IERR Interrupt

This interrupt is generated in the following cases.

- When ECC correction was incapable of correcting an error
- When ECC correction was OK but the subsequent EDC check indicated an error

#### (5) IBUF Interrupt

This interrupt is generated when the following transitions occur.

- Data transfer to the buffer → Data transfer complete (searching for data for the next transfer)
- Data for transfer to the buffer are being searched for → Data transfer started

#### (6) IREADY Interrupt

This interrupt is generated when decoding of data for one sector is completed. This interrupt should be used to start the CPU buffering stream data for output to SDRAM.

#### (7) DMA Transfer Request

The source of direct memory access controller activation is the same as that of IREADY. An interrupt request is generated when output stream data for one sector becomes ready, and after the 2768 bytes of data shown in figure 24.15 have been transferred, the request signal is negated. This is because a certain amount of time is required before the output data for the next sector is ready, so the transfer request from the direct memory access controller should be turned off between transfers.

### 24.5.2 Timing of Status Registers Updates

The status information registers of the CD-ROM decoder are updated on each ISEC interrupt. The sector for which information is reflected in the status registers is selected by the ER0SEL bit of the CROMCTL4 register.

## 24.6 Usage Notes

### 24.6.1 Stopping and Resuming Buffering Alone during Decoding

When the data of the output stream are not being read out but operation of the CD-ROM decoder has continued until the buffers are full, the BUF\_NG bit in the CBUFST2 register is set to 1; after that, the CD-ROM decoder becomes incapable of operation.

To stop buffering alone, clear the CBUF\_EN bit in the CBUFCTL0 register to 0. If the automatic buffering function is in use, clear the CBUF\_AUT bit in the CBUFCTL0 register to 0 at the same time. In this case, the sectors currently in the buffers must be read out.

To resume automatic buffering, set the CBUF\_AUT and CBUF\_EN bits in the CBUFCTL0 register to 1 at the same time.

### 24.6.2 Setting Sync Code Status Register (CROMST0)

1. When the ST\_SECS bit in the CROMST0 register becomes 1, stop decoding immediately and retry from one sector before the sector that was being decoded.
2. When the ST\_SECL bit in the CROMST0 register becomes 1, stop decoding immediately and retry from two sectors before the sector that was being decoded.

### 24.6.3 Link Blocks

The CD-ROM decoder uses the header information before ECC correction to detect link blocks. Accordingly, an input data stream that contains an error may be erroneously detected as a link block. To prevent this, the following measures should be implemented in software.

- During buffering (BUF\_ACT = 1 in the CBUFST0 register), check the LINK\_OUT1 bit in the CROMST5 register on each ISEC interrupt. If it is set to 1, check to see if an IERR interrupt has also occurred; if an IERR interrupt has not occurred, save the MFS values from the HEAD20 to HEAD23 registers. If an IERR interrupt has occurred, do not save the MSF values.
- Perform the following processing for seven sectors (indicated by ISEC being generated seven times) after finding that the LINK\_OUT1 bit has been set to 1.
- In either of cases 1 and 2 below,
  1. LINK\_ON = 1 (in the CROMST5 register) is confirmed at each ISEC interrupt, and LINK\_ON = 1 is detected again within the subsequent two-sector period
  2. LINK\_ON = 1 was not detected at any ISEC interrupt

forcibly stop decoding, set the CROMSY0 register to place the decoder in external sync mode, and retry decoding by specifying the MSF value stored as described above + 7 (immediately after a link block) as the MSF value for the target sector.

The start sector address will be the address where RUN\_OUT is stored + 7.

### 24.6.4 Stopping and Resuming CD-DSP Operation

When stopping and then resuming the stream data input to the CD-ROM decoder, if the input data stream does not stop immediately before a sync code and is then resumed, the CD-ROM decoder may recognize the data as incorrect. This happens because the system holds the data up to the point where input was stopped and the data that is input from the point of resumption at the same time. Take care on this point when stopping and then resuming input.

### 24.6.5 Note on Clearing the IREADY Flag

To clear the IREADY flag to 0 in interrupt processing etc., be sure to read one sector of data (2768 bytes) beforehand. If

the IREADY flag is cleared to 0 before reading of one sector of data is complete, decoding of the subsequent sectors will not be possible. For recovery from this situation, write 1 to the LOGICRST bit in the CD-ROM decoder reset control register (ROMDECRST), and then clear the bit to 0.

#### 24.6.6 Note on Stream Data Transfer (1)

If reading of the output data stream is slower than writing of the input data stream, the buffer of the CD-ROM decoder will overflow. This causes the CD-ROM decoder to abnormally stop.

When DMA transfer is in use, ensure that reading of the output data stream is faster than writing of the input data stream by making settings as listed below.

- Set a larger transfer size for reading of the output data stream than for writing of the input data stream.
- Give reading of the output data stream higher priority than writing of the input data stream.
- Set a smaller interval count for reading of the output data stream than for writing of the input data stream.

When the CPU handles transfer, ensure that reading of the output data stream is faster than writing of the input data stream by taking similar measures to those in the case of DMA transfer.

#### 24.6.7 Note on Stream Data Transfer (2)

When reading the stream data, be sure to use either the direct memory access controller or the CPU. If both the direct memory access controller and the CPU are used for reading, the stream data may not be recognized as being in the CD-ROM format.

## 25. LIN Interface

This section contains a generic description of the LIN Interface (RLIN3). The first part of this section describes all RZ/A1H specific properties, such as the register base addresses, etc. The remainder of the section describes the functions and registers of RLIN3.

### 25.1 RZ/A1H LIN Interface Features

#### 25.1.1 Channels

This LSI has following number of channels of the LIN Interface.

**Table 25.1 Channels of RLIN3n**

LIN Interface	RZ/A1H
Channels	2
Name	RLIN30 and RLIN31

#### Index n

Throughout this section, the channel of the LIN Interface (RLIN3) is identified by the index “n” (n = 0 or 1), for example, RLIN3nLMD for the LIN mode register.

#### 25.1.2 Register addresses

All LIN interface register addresses are given as address offsets to the individual base address <RLIN3\_base>. The register base address <RLIN3\_base> of RLIN3n is listed in **Table 25.2**.

**Table 25.2 Register Base Addresses**

RLIN3n	Register Base Address <RLIN3n_base>
RLIN3n	FCFE 9000 <sub>H</sub>

#### 25.1.3 Clock supply

The following clock is provided for the LIN interface.

**Table 25.3 LIN3n Clock Supply**

RLIN3n	Clock	Connected to
RLIN3n	LIN Communication Clock Source	P0φ

### 25.1.4 Interrupts and DMA

The LIN Interface can generate the following interrupt and DMA requests:

**Table 25.4** RLIN3n Interrupt and DMA Requests

RLIN3n signal	Function
<b>RLIN30</b>	
LIN0_INT_M	LIN0 interrupt
LIN0_INT_T	LIN0 transmission interrupt
LIN0_INT_R	LIN0 successful reception interrupt
LIN0_INT_S	LIN0 status error interrupt
<b>RLIN31</b>	
LIN1_INT_M	LIN1 interrupt
LIN1_INT_T	LIN1 transmission interrupt
LIN1_INT_R	LIN1 successful reception interrupt
LIN1_INT_S	LIN1 status error interrupt

### 25.1.5 I/O signals

The I/O signals of the LIN Interface are used for various purposes, as listed in **Table 25.5**.

**Table 25.5** RLIN3n I/O Signals

Name	Function
<b>RLIN30</b>	
Port RLIN30RX	RLIN30 receive data input
Port RLIN30TX	RLIN30 transmit data output
<b>RLIN31</b>	
Port RLIN31RX	RLIN31 receive data input
Port RLIN31TX	RLIN31 transmit data output

## 25.2 Function

The LIN Interface is a hardware LIN communication controller that supports LIN Specification Package Revision 1.3, 2.0, and 2.1, and automatically performs frame communication and error determination.

The LIN master mode is only available.

### LIN master

- LIN reset mode
- LIN mode (LIN master mode)
  - LIN wake-up mode
  - LIN operation mode
- LIN self-test mode

**Table 25.6** gives the LIN Interface specifications and **Figure 25.1** shows a block diagram of the LIN Interface.

**Table 25.6 LIN Interface Specifications**

Item	Specifications		
	Channel count	2	
LIN communication function	Protocol	LIN Specification Package Revision 1.3, 2.0, and 2.1	
	Variable frame structure	Master	<ul style="list-style-type: none"> <li>Break (low) transmission width: 13 to 28 Tbits</li> <li>Break delimiter transmission width: 1 to 4 Tbits</li> <li>Inter-byte space (header): 0 to 7 Tbits (space between Sync field and ID field)*<sup>1</sup></li> <li>Response space: 0 to 7 Tbits*<sup>1</sup></li> <li>Inter-byte space: 0 to 3 Tbits (space between data bytes in response area)</li> <li>Wake-up: 1 to 16 Tbits</li> </ul>
		Checksum	<ul style="list-style-type: none"> <li>Automatic operation for both transmission and reception</li> <li>Classic or enhanced selectable (for each frame)</li> </ul>
		Response field data byte count	Variable from 0 to 8 bytes Multi-byte (9 or more bytes) response transmission and reception also possible
		Frame communication modes	Master <ul style="list-style-type: none"> <li>Mode in which header transmission and response transmission/reception is started with a single transmission start request</li> <li>Mode in which header transmission and response transmission are started with separate transmission start requests (frame separate mode)</li> </ul>
		Wake-up transmission and reception	LIN wake-up mode provided <ul style="list-style-type: none"> <li>Wake-up transmission (1 to 16 Tbits)</li> <li>Wake-up reception</li> </ul> Low-level width of input signals measured
	Status	Master <ul style="list-style-type: none"> <li>Successful frame/wake-up transmission</li> <li>Successful header transmission</li> <li>Successful frame/wake-up reception*<sup>2</sup></li> <li>Successful data 1 reception</li> <li>Error detection</li> <li>Operation mode (LIN reset mode, LIN wake-up mode, LIN operation mode, LIN self-test mode)</li> </ul>	
	Error status	Master <ul style="list-style-type: none"> <li>Bit error</li> <li>Checksum error</li> <li>Frame timeout error/response timeout error</li> <li>Physical bus error</li> <li>Framing error</li> <li>Response preparation error</li> </ul>	
	Baud rate selection	Baud rate conforming to the LIN specifications generated using baud rate generator	
	Test mode	Self-test mode for user evaluation	
Interrupt function	Master	<ul style="list-style-type: none"> <li>Successful header/frame/wake-up transmission</li> <li>Successful frame/wake-up reception*<sup>2</sup></li> <li>Error detection</li> </ul>	

Note 1. Since the same register is used for setting, the inter-byte space (header) = response space.

Note 2. For wake-up reception, the low level width of the input signal is indicated.



### 25.2.1 Block Diagram

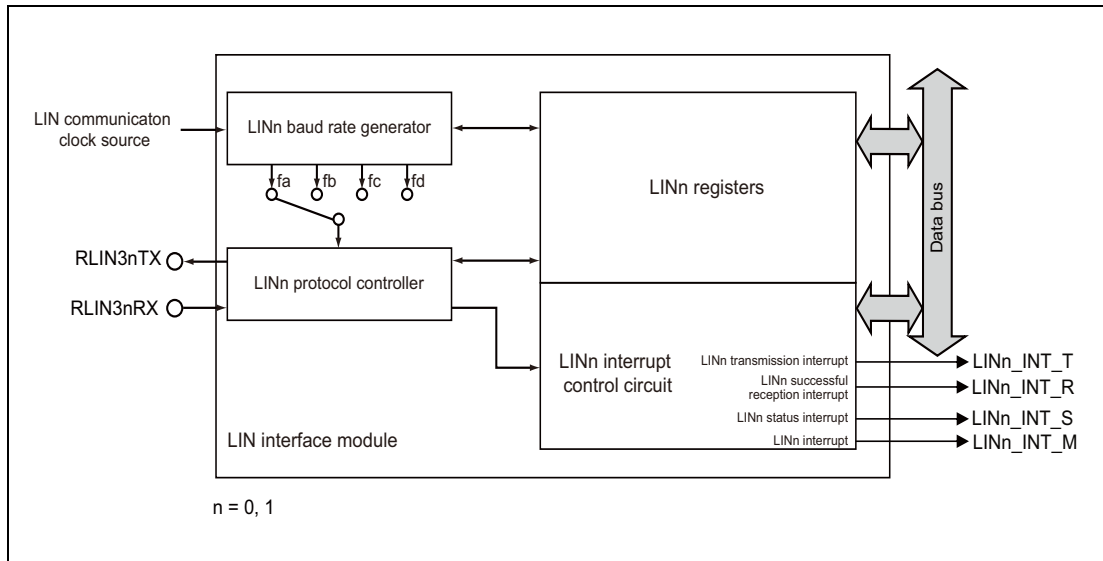


Figure 25.1 LIN Interface Block Diagram

### 25.2.2 Description of Blocks

- RLIN3nTX, RLIN3nRX: LIN Interface I/O pins
- LINn baud rate generator: Generates the LIN Interface communication clock signal.
- LINn registers: LIN Interface registers
- LINn interrupt controller: Controls interrupt requests generated by the LIN Interface

## 25.3 Registers

**Table 25.7** lists the LIN Interface registers.

**Table 25.7 List of LIN Interface Registers**

Register Name	Symbol
LIN wake-up baud rate selector register	RLN3nLWBR
LIN baud rate prescaler 0 register	RLN3nLBRP0
LIN baud rate prescaler 1 register	RLN3nLBRP1
LIN self-test control register	RLN3nLSTC
LIN mode register	RLN3nLMD
LIN break field configuration register	RLN3nLBFC
LIN space configuration register	RLN3nLSC
LIN wake-up configuration register	RLN3nLWUP
LIN interrupt enable register	RLN3nLIE
LIN error detection enable register	RLN3nLEDE
LIN control register	RLN3nLCUC
LIN transmission control register	RLN3nLTRC
LIN mode status register	RLN3nLMST
LIN status register	RLN3nLST
LIN error status register	RLN3nLEST
LIN data field configuration register	RLN3nLDFC
LIN ID buffer register	RLN3nLIDB
LIN checksum buffer register	RLN3nLCBR
LIN data buffer 1 register	RLN3nLDBR1
LIN data buffer 2 register	RLN3nLDBR2
LIN data buffer 3 register	RLN3nLDBR3
LIN data buffer 4 register	RLN3nLDBR4
LIN data buffer 5 register	RLN3nLDBR5
LIN data buffer 6 register	RLN3nLDBR6
LIN data buffer 7 register	RLN3nLDBR7
LIN data buffer 8 register	RLN3nLDBR8

**Note:** When writing to a register not used, write 00<sub>H</sub>.

## 25.3.1 LIN Master Related Registers

### 25.3.1.1 RLN3nLWBR — LIN Wake-up Baud Rate Select Register

**Access:** This register can be read/written in 8-bit units.

**Address:** <RLIN3\_base> + 800<sub>H</sub> × n + 01<sub>H</sub> (n = 0, 1)

**Initial value:** 00<sub>H</sub>

Bit	7	6	5	4	3	2	1	0
	NSPB[3:0]				LPRS[2:0]			LWBR0
Initial value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

**Table 25.8** RLN3nLWBR register contents

Bit Position	Bit Name	Function
b7 to b4	NSPB[3:0]	Bit Sampling Count Select b7 b4 0 0 0 0: 16 sampling 1 1 1 1: 16 sampling Settings other than the above are prohibited.
b3 to b1	LPRS[2:0]	Prescaler Clock Select b3 b1 0 0 0: 1/1 0 0 1: 1/2 Other than above: Setting prohibited
b0	LWBR0	Wake-up Baud Rate Select 0: When LIN1.3 is used 1: When LIN 2.0 or 2.1 is used

Set the RLN3nLWBR register when the OMM0 bit in the RLN3nLMST register is 0B (in LIN reset mode).

#### NSPB[3:0] bits (bit sampling count select bits)

These bits select the number of sampling in one Tbit (reciprocal of the bit rate).

In LIN master mode, set these bits to 0000B or 1111B (16 sampling).

#### LPRS[2:0] bits (prescaler clock select bits)

These bits select the frequency division ratio for the prescaler.

#### LWBR0 bit (wake-up baud rate select bit)

When LIN Specification Package Revision 1.3 is used, set the LWBR0 bit in the RLN3nLWBR register to 0. This allows the 2.5-Tbit or longer low-level width of the input signal to be measured. When LIN Specification Package Revision 2.0, 2.1 is used, set the LWBR0 bit to 1. Setting the LWBR0 bit to 1 selects  $f_a$  as the LIN system clock ( $f_{LIN}$ ) during LIN wake-up mode regardless of the setting of the RLN3nLMD.LCKS bit (the LCKS bit is not changed). This allows the 2.5-Tbit or longer low-level width of the input signal to be measured.

Setting the baud rate to 19200 bps while  $f_a$  is selected allows the 130  $\mu$ s or longer low-level width of the input signal to be detected during LIN wake-up mode regardless of the setting of the RLN3nLMD.LCKS bit.

### 25.3.1.2 RLN3nLBRP0 — LIN Baud Rate Prescaler 0 Register

**Access:** This register can be read/written in 8-bit units.

**Address:** <RLIN3\_base> + 800<sub>H</sub> × n + 02<sub>H</sub> (n = 0, 1)

**Initial value:** 00<sub>H</sub>

Bit	7	6	5	4	3	2	1	0
	LBRP0[7:0]							
Initial value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

**Table 25.9** RLN3nLBRP0 register contents

Bit Position	Bit Name	Function
b7 to b0	LBRP0[7:0]	Assuming that the value set in this register is N (0 to 255), the baud rate prescaler divides the frequency of the prescaler clock divided by the the LPRS bits by N + 1. Setting Range: 00 <sub>H</sub> to FF <sub>H</sub>

Set the RLN3nLBRP0 register when the OMM0 bit in the RLN3nLMST register is 0B (in LIN reset mode).

The value set in this register is used to control the frequency of baud rate clock sources fa, fb, and fc.

Assuming that the value set in this register is N, baud rate prescaler 0 divides the frequency of the clock that is selected by the LPRS bits by N + 1.

### 25.3.1.3 RLN3nLBRP1 — LIN Baud Rate Prescaler 1 Register

**Access:** This register can be read/written in 8-bit units.

**Address:** <RLIN3\_base> + 800<sub>H</sub> × n + 03<sub>H</sub> (n = 0, 1)

**Initial value:** 00<sub>H</sub>

Bit	7	6	5	4	3	2	1	0
	LBRP1[7:0]							
Initial value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

**Table 25.10** RLN3nLBRP1 register contents

Bit Position	Bit Name	Function
b7 to b0	LBRP1[7:0]	Assuming that the value set in this register is M (5 to 255), the baud rate prescaler 1 divides the frequency of the prescaler clock divided by the the LPRS bits by M + 1. Setting Range: 00 <sub>H</sub> to FF <sub>H</sub>

Set the RLN3nLBRP1 register when the OMM0 bit in the RLN2uunLcMST register is 0B (in LIN reset mode).

The value set in this register is used to control the frequency of baud rate clock source fd.

Assuming that the value set in this register is M, baud rate prescaler 1 divides the frequency of the clock that is selected by the LPRS bits (prescaler clock select bits) by M + 1.

### 25.3.1.4 RLN3nLSTC — LIN Self-Test Control Register

**Access:** This register can be read/written in 8-bit units.

**Address:** <RLIN3\_base> + 800<sub>H</sub> × n + 04<sub>H</sub> (n = 0, 1)

**Initial value:** 00<sub>H</sub>

Bit	7	6	5	4	3	2	1	0
								LSTM
Initial value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

**Table 25.11** RLN3nLSTC register contents

Bit Position	Bit Name	Function
b7 to b1	Reserved	When read, an initial value is returned. When written, write an initial value.
b0	LSTM	LIN Self-Test Mode 0: The module is not in LIN self-test mode 1: The module is in LIN self-test mode.

The RLN3nLSTC register cancels protection of LIN self-test mode.

Set the RLN3nLSTC register when the OMM0 bit in the RLN3nLMST register is 0B (in LIN reset mode).

Writing A7<sub>H</sub>, 58<sub>H</sub>, and 01<sub>H</sub> successively to the RLN3nLSTC register places the module into LIN self-test mode.

When successive writing is completed thus placing LIN self-test mode to be entered, the LSTM bit is set to 1.

Do not write any other value during successive writing.

For making transition to LIN self-test mode, refer to **Section 25.8, LIN Self-Test Mode**.

#### LSTM bit (LIN self test mode bit)

When transition to LIN self-test mode is completed, the LSTM bit is set to 1.

For leaving LIN self-test mode, refer to **Section 25.8, LIN Self-Test Mode**.

Writing 1 to this bit does not affect the value of the RLN3nLSTC register if it is not a part of successive writing of A7<sub>H</sub>, 58<sub>H</sub>, and 01<sub>H</sub>.

### 25.3.1.5 RLN3nLMD — LIN Mode Register

**Access:** This register can be read/written in 8-bit units.

**Address:** <RLIN3\_base> + 800<sub>H</sub> × n + 08<sub>H</sub> (n = 0, 1)

**Initial value:** 00<sub>H</sub>

Bit	7	6	5	4	3	2	1	0
	—	—	LRDNFS	LIOS	LCKS[1:0]		LMD[1:0]	
Initial value	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

**Table 25.12** RLN3nLMD register contents

Bit Position	Bit Name	Function
b7, b6	Reserved	When read, an initial value is returned. When written, write an initial value.
b5	LRDNFS	LIN Reception Data Noise Filtering Disable 0: The noise filter is enabled. 1: The noise filter is disabled.
b4	LIOS	LIN Interrupt Output Select 0: LINn interrupt is used. 1: LINn transmission interrupt, LINn successful reception interrupt, and LINn status error interrupt are used.
b3, b2	LCKS[1:0]	LIN System Clock Select b3 b2 0 0: fa (Clock generated by baud rate prescaler 0) 0 1: fb (1/2 clock generated by baud rate prescaler 0) 1 0: fc (1/8 clock generated by baud rate prescaler 0) 1 1: fd (1/2 clock generated by baud rate prescaler 1)
b1, b0	LMD[1:0]	LIN Mode Select b1 b0 0 0: LIN master mode

Set the RLN3nLMD register when the OMM0 bit in the RLN3nLMST register is 0B (in LIN reset mode).

#### LRDNFS bit (LIN reception data noise filtering disable bit)

The LRDNFS bit enables or disables the noise filter when receiving data.

With 0 set, the noise filter is enabled when receiving data.

With 1 set, the noise filter is disabled when receiving data.

#### LIOS bit (LIN interrupt output select bit)

The LIOS bit selects the number of interrupt outputs from the LIN Interface.

With 0 set, the LINn interrupt is generated from the LIN Interface.

With 1 set, the LINn transmission interrupt, LINn successful reception interrupt, and LINn status interrupt are generated from the LIN Interface.

For each interrupt source, refer to **Section 25.4, Interrupt Sources**.

#### LCKS[1:0] bits (LIN system clock select bits)

The LCKS bits select the clock to be input to the protocol controller.

With 00<sub>B</sub> set, the protocol controller is provided with fa (clock generated by baud rate prescaler 0).

With  $01_B$  set, the protocol controller is provided with  $f_b$  (1/2 clock generated by baud rate prescaler 0).

With  $10_B$  set, the protocol controller is provided with  $f_c$  (1/8 clock generated by baud rate prescaler 0).

With  $11_B$  set, the protocol controller is provided with  $f_d$  (1/2 clock generated by baud rate prescaler 1).

#### **LMD[1:0] bits (LIN mode select bits)**

The LMD bits select the LIN Interface mode.

To use the LIN Interface as an LIN master, set these bits to  $00_B$ .

### 25.3.1.6 RLN3nLBFC — LIN Break Field Configuration Register

**Access:** This register can be read/written in 8-bit units.

**Address:** <RLIN3\_base> + 800<sub>H</sub> × n + 09<sub>H</sub> (n = 0, 1)

**Initial value:** 00<sub>H</sub>

Bit	7	6	5	4	3	2	1	0
	—	—	BDT[1:0]		BLT[3:0]			
Initial value	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

**Table 25.13** RLN3nLBFC register contents

Bit Position	Bit Name	Function
b7, b6	Reserved	When read, an initial value is returned. When written, write an initial value.
b5, b4	BDT[1:0]	Break Delimiter (High level) width select bit b5 b4 0 0: 1 Tbit 0 1: 2 Tbits 1 1: 3 Tbits 1 1: 4 Tbits
b3 to b0	BLT[3:0]	Transmit Break (Low level) width select bit b3 b0 0 0 0 0: 13 Tbits 0 0 0 1: 14 Tbits 0 0 1 0: 15 Tbits : 1 1 1 0: 27 Tbits 1 1 1 1: 28 Tbits

Set the RLN3nLBFC register when the OMM0 bit in the RLN3nLMST register is 0B (in LIN reset mode).

Some combinations of the set values result in the length of a frame exceeding the timeout time. Set the appropriate values in this register.

#### BDT[1:0] bit (Transmission Break Delimiter high level width setting bit)

This bit is used to set the break high **level** width of transmission frame header.

1 Tbit to 4 Tbits can be set.

#### BLT[3:0] bit (Transmission Break Low level width setting bit)

This BLT bits set the break low **level** width of transmission frame header.

13 Tbits to 28 Tbits can be set.



### 25.3.1.7 RLN3nLSC — LIN Space Configuration Register

**Access:** This register can be read/written in 8-bit units.

**Address:** <RLIN3\_base> + 800<sub>H</sub> × n + 0A<sub>H</sub> (n = 0, 1)

**Initial value:** 00<sub>H</sub>

Bit	7	6	5	4	3	2	1	0
	—	—	IBS[1:0]		—	IBHS[2:0]		
Initial value	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R	R/W	R/W	R/W

**Table 25.14** RLN3nLSC register contents

Bit Position	Bit Name	Function
b7, b6	Reserved	When read, an initial value is returned. When written, write an initial value.
b5, b4	IBS[1:0]	Inter-Byte Space Select b5 b4 0 0: 0 Tbit 0 1: 1 Tbit 1 1: 2 Tbits 1 1: 3 Tbits
b3	Reserved	When read, an initial value is returned. When written, write an initial value.
b2 to b0	IBHS[2:0]	Inter-Byte Space (Header)/Response Space Select b2 b0 0 0 0: 0 Tbit 0 0 1: 1 Tbit 0 1 0: 2 Tbits 0 1 1: 3 Tbits 1 0 0: 4 Tbits 1 0 1: 5 Tbits 1 1 0: 6 Tbits 1 1 1: 7 Tbits

Set the RLN3nLSC register when the OMM0 bit in the RLN3nLMST register is 0B (in LIN reset mode).

Some combinations of the set values result in the length of a frame or a response exceeding the timeout time. Set the appropriate values in this register.

#### IBS[1:0] bits (inter-byte space select bits)

The IBS bits set the width of the inter-byte space of the transmission frame response field.

0 Tbit to 3 Tbits can be set.

These bits are enabled only during response transmission; these are disabled during response reception.

#### IBHS[2:0] bits (inter-byte space (header)/response space select bits)

The IBHS bits set the width of the inter-byte space (header) of the transmission frame header field and the response space.

0 Tbit to 7 Tbits can be set.

The response space setting is enabled only during response transmission; setting is disabled during response reception.

The inter-byte space (header) is equal to the response space.

### 25.3.1.8 RLN3nLWUP — LIN Wake-up Configuration Register

**Access:** This register can be read/written in 8-bit units.

**Address:** <RLIN3\_base> + 800<sub>H</sub> × n + 0B<sub>H</sub> (n = 0, 1)

**Initial value:** 00<sub>H</sub>

Bit	7	6	5	4	3	2	1	0
	WUTL[3:0]				—	—	—	—
Initial value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R

**Table 25.15** RLN3nLWUP register contents

Bit Position	Bit Name	Function
b7 to b4	WUTL[3:0]	Wake-up Transmission Low level Width Select b7 b4 0 0 0 0: 1 Tbit 0 0 0 1: 2 Tbits 0 0 1 0: 3 Tbits 0 0 1 1: 4 Tbits : 1 1 0 0: 13 Tbits 1 1 0 1: 14 Tbits 1 1 1 0: 15 Tbits 1 1 1 1: 16 Tbits
b3 to b0	Reserved	When read, an initial value is returned. When written, write an initial value.

Set the RLN3nLWUP register when the OMM0 bit in the RLN3nLMST register is 0B (in LIN reset mode).

#### WUTL[3:0] bits (wake-up transmission low level width select bits)

The WUTL bits set the low level width of the wake-up signal transmission.

1 Tbit to 16 Tbits can be set.

While 1 is set in the LWBR0 bit in the RLN3nLWBR register (LIN 2.0 or 2.1 is used), fa is selected as the LIN system clock (fLIN) regardless of the setting of the RLN3nLMD.LCKS bit (the LCKS bit is not changed).

### 25.3.1.9 RLN3nLIE — LIN Interrupt Enable Register

**Access:** This register can be read/written in 8-bit units.

**Address:** <RLIN3\_base> + 800<sub>H</sub> × n + 0C<sub>H</sub> (n = 0, 1)

**Initial value:** 00<sub>H</sub>

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	SHIE	ERRIE	FRCIE	FTCIE
Initial value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

**Table 25.16** RLN3nLIE register contents

Bit Position	Bit Name	Function
b7 to b4	Reserved	When read, an initial value is returned. When written, write an initial value.
b3	SHIE	Successful Header Transmission Interrupt Enable 0: Disables successful header transmission interrupt. 1: Enables successful header transmission interrupt.
b2	ERRIE	Error Detection Interrupt Enable 0: Disables error detection interrupt. 1: Enables error detection interrupt.
b1	FRCIE	Successful Frame/Wake-up Reception Interrupt Enable 0: Disables successful frame/wake-up reception interrupt. 1: Enables successful frame/wake-up reception interrupt.
b0	FTCIE	Successful Frame/Wake-up Transmission Interrupt Enable 0: Disables successful frame/wake-up transmission interrupt. 1: Enables successful frame/wake-up transmission interrupt.

Set the RLN3nLIE register when the OMM0 bit in the RLN3nLMST register is 0B (in LIN reset mode).

#### SHIE bit (successful header transmission interrupt enable bit)

The SHIE bit enables or disables interrupt generation upon successful transmission of a header.

With 0 set, the interrupt is not generated when the HTRC flag in the RLN3nLST register is set to 1.

With 1 set, the interrupt is generated when the HTRC flag in the RLN3nLST register is set to 1.

#### ERRIE bit (error detection interrupt enable bit)

The ERRIE bit enables or disables interrupt generation upon detection of an error.

With 0 set, the interrupt is not generated when the ERR flag in the RLN3nLST register is set to 1.

With 1 set, the interrupt is generated when the ERR flag in the RLN3nLST register is set to 1.

Interrupt sources can be the bit error, physical bus error, frame/response timeout error, framing error, checksum error, and response preparation error.

Detection of the bit error, physical bus error, frame/response timeout error, and framing error can be enabled or disabled using the RLN3nLEDE register.

#### FRCIE bit (successful frame/wake-up reception interrupt enable bit)

The FRCIE bit enables or disables interrupt generation upon successful reception of a frame or a wake-up signal (counting of low level width of the input signal).

With 0 set, the interrupt is not generated when the FRC flag in the RLN3nLST register is set to 1.

With 1 set, the interrupt is generated when the FRC flag in the RLN3nLST register is set to 1.

**FTCIE bit (successful frame/wake-up transmission interrupt enable bit)**

The FTCIE bit enables or disables interrupt generation upon successful transmission of a frame or a wake-up signal.

With 0 set, the interrupt is not generated when the FTC flag in the RLN3nLST register is set to 1.

With 1 set, the interrupt is generated when the FTC flag in the RLN3nLST register is set to 1.

### 25.3.1.10 RLN3nLEDE —LIN Error Detection Enable Register

**Access:** This register can be read/written in 8-bit units.

**Address:** <RLIN3\_base> + 800<sub>H</sub> × n + 0D<sub>H</sub> (n = 0, 1)

**Initial value:** 00<sub>H</sub>

Bit	7	6	5	4	3	2	1	0
	LTES	—	—	—	FERE	FTERE	PBERE	BERE
Initial value	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R/W	R/W	R/W	R/W

**Table 25.17** RLN3nLEDE register contents

Bit Position	Bit Name	Function
b7	LTES	Timeout Error Select 0: Frame timeout error 1: Response timeout error
b6 to b4	Reserved	When read, an initial value is returned. When written, write an initial value.
b3	FERE	Framing Error Detection Enable 0: Disables framing error detection. 1: Enables framing error detection.
b2	FTERE	Timeout Error Detection Enable 0: Disables frame/response timeout error detection. 1: Enables frame/response timeout error detection.
b1	PBERE	Physical Bus Error Detection Enable 0: Disables physical bus error detection. 1: Enables physical bus error detection.
b0	BERE	Bit Error Detection Enable 0: Disables bit error detection. 1: Enables bit error detection.

Set the RLN3nLEDE register when the OMM0 bit in the RLN3nLMST register is 0B (in LIN reset mode).

#### LTES bit (timeout error select bit)

The LTES bit selects the specific timeout function to be used.

With 0 set, the timeout function applies to frame timeout.

With 1 set, the timeout function applies to response timeout.

For details of the timeout error, refer to **Section 25.7.6, Error Status**.

#### FERE bit (framing error detection enable bit)

The FERE bit enables or disables detection of the framing error.

With 0 set, the framing error is not detected.

With 1 set, the framing error is detected.

When this bit is set to 1, the detection result is indicated in the FER flag in the RLN3nLEST register.

For details of the framing error, refer to **Section 25.7.6, Error Status**.

**FTERE bit (timeout error detection enable bit)**

The FTERE bit enables or disables detection of the frame timeout error or the response timeout error.

With 0 set, the frame timeout error or response timeout error is not detected.

With 1 set, the frame timeout error or response timeout error is detected.

When this bit is set to 1, the detection result is indicated in the FTER flag in the RLN3nLEST register.

With the LTES bit, either the frame timeout error or response timeout error can be selected.

Do not use the timeout error if response data of 9 bytes or more is to be transmitted or received.

For details of the timeout error, refer to **Section 25.7.6, Error Status**.

**PBERE bit (physical bus error detection enable bit)**

The PBERE bit enables or disables detection of the physical bus error.

With 0 set, the physical bus error is not detected.

With 1 set, the physical bus error is detected.

When this bit is set to 1, the detection result is indicated in the PBER flag in the RLN3nLEST register.

For details of the physical bus error, refer to **Section 25.7.6, Error Status**.

**BERE bit (bit error detection enable bit)**

The BERE bit enables or disables detection of the bit error.

With 0 set, the bit error is not detected.

With 1 set, the bit error is detected.

When this bit is set to 1, the detection result is indicated in the BER flag in the RLN3nLEST register.

For details of the bit error, refer to **Section 25.7.6, Error Status**.

### 25.3.1.11 RLN3nLCUC — LIN Control Register

**Access:** This register can be read/written in 8-bit units.

**Address:** <RLIN3\_base> + 800<sub>H</sub> × n + 0E<sub>H</sub> (n = 0, 1)

**Initial value:** 00<sub>H</sub>

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	OM1	OM0
Initial value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

**Table 25.18** RLN3nLCUC register contents

Bit Position	Bit Name	Function
b7 to b2	Reserved	When read, an initial value is returned. When written, write an initial value.
b1	OM1	LIN Mode Select 0: LIN wake-up mode is caused. 1: LIN operation mode is caused.
b0	OM0	LIN Reset 0: LIN reset mode is caused. 1: LIN reset mode is canceled.

Set the RLN3nLCUC register to 01<sub>H</sub> to cause a transition to LIN wake-up mode after canceling LIN reset mode, and set the register to 03<sub>H</sub> to cause a transition to LIN operation mode.

In LIN self-test mode, set the RLN3nLCUC register to 03h after a transition to LIN self-test mode is completed.

After a value is written to this register, confirm that the value written is actually indicated in the RLN3nLMST register before writing another value.

#### OM1 bit (LIN mode select bit)

The OM1 bit selects the specific LIN operation mode (either LIN wake-up mode or LIN operation mode) after canceling LIN reset mode.

With 0 set, LIN wake-up mode is caused.

With 1 set, LIN operation mode is caused.

This bit is valid only when the OMM0 bit in the RLN3nLMST register is 1.

Writing a value to this bit is disabled while the FTS bit in the RLN3nLTRC register is 1.

#### OM0 bit (LIN reset bit)

The OM0 bit selects either causing a transition to LIN reset mode or canceling LIN reset mode.

With 0 set, LIN reset mode is caused.

With 1 set, LIN reset mode is canceled.

### 25.3.1.12 RLN3nLTRC — LIN Transmission Control Register

**Access:** This register can be read/written in 8-bit units.

**Address:** <RLIN3\_base> + 800<sub>H</sub> × n + 10<sub>H</sub> (n = 0, 1)

**Initial value:** 00<sub>H</sub>

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RTS	FTS
Initial value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

**Table 25.19** RLN3nLTRC register contents

Bit Position	Bit Name	Function
b7 to b2	Reserved	When read, an initial value is returned. When written, write an initial value.
b1	RTS	Response Transmission/Reception Start 0: Response transmission/reception is stopped in frame separate mode. 1: Response transmission/reception is started in frame separate mode.
b0	FTS	Frame Transmission/wake-up Transmission /Reception Start 0: Frame Transmission/wake-up transmission /reception is stopped. 1: Frame Transmission/wake-up transmission reception is started.

#### RTS bit (response transmission/reception start bit)

Set the RTS bit to 1 in frame separate mode after header transmission is completed (HTRC flag in the RLN3nLST register is 1) and response transmission data is ready. Operations cannot be guaranteed if this bit is set to 1 before header transmission is completed. Once set, this bit is automatically cleared to 0 upon completion of frame communication or transition to LIN reset mode.

Only 1 can be written to this bit; 0 cannot be written. Write 0 to the bit to be cleared and write 1 to other bits.

Writing a value to this bit is disabled in LIN reset mode (when the OMM0 bit is 0).

Writing a value to this bit is disabled when the FTS bit is 0 (frame transmission or wake-up transmission/reception is halted).

When response data of 9 bytes or more is to be transmitted or received, set this bit to 1 each time a data group (variable from 0 to 8 bytes) is transmitted or received. Once set, this bit is automatically cleared to 0 upon completion of data group transmission/reception or transition to LIN reset mode.

#### FTS bit (frame transmission/wake-up transmission/reception start bit)

Set the FTS bit to 1 to start frame/wake-up transmission.

Also set this bit to 1 to allow wake-up reception (counting of the low level width of the input signal).

Only 1 can be written to this bit; 0 cannot be written. Write 0 to the bit to be cleared and write 1 to other bits.

Writing a value to this bit is disabled in LIN reset mode (when the OMM0 bit is 0). This bit is set to 0 upon completion of frame or wake-up communication and transition to LIN reset mode.



### 25.3.1.13 RLN3nLMST — LIN Mode Status Register

**Access:** This register can be read/written in 8-bit units.

**Address:** <RLIN3\_base> + 800<sub>H</sub> × n + 11<sub>H</sub> (n = 0, 1)

**Initial value:** 00<sub>H</sub>

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	OMM1	OMM0
Initial value	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

**Table 25.20** RLN3nLMST register contents

Bit Position	Bit Name	Function
b7 to b2	Reserved	When read, an initial value is returned. When written, write an initial value.
b1	OMM1	LIN Mode Status Monitor 0: The module is in LIN wake-up mode. 1: The module is in LIN operation mode.
b0	OMM0	LIN Reset Status Monitor 0: The module is in LIN reset mode. 1: The module is not in LIN reset mode.

#### OMM1 bit (LIN mode status monitor)

The OMM1 bit indicate the current operating mode.

#### OMM0 bit (LIN reset status monitor)

The OMM0 bit indicates the current operating mode.

### 25.3.1.14 RLN3nLST — LIN Status Register

**Access:** This register can be read/written in 8-bit units.

**Address:** <RLIN3\_base> + 800<sub>H</sub> × n + 12<sub>H</sub> (n = 0, 1)

**Initial value:** 00<sub>H</sub>

Bit	7	6	5	4	3	2	1	0
	HTRC	D1RC	—	—	ERR	—	FRC	FTC
Initial value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R/W	R/W

**Table 25.21** RLN3nLST register contents

Bit Position	Bit Name	Function
b7	HTRC	Successful Header Transmission Flag 0: Header transmission has not been completed. 1: Header transmission has been completed.
b6	D1RC	Successful Data 1 Reception Flag These bits are always read as 0. The write value should always be 0.
b5, b4	Reserved	When read, an initial value is returned. When written, write an initial value.
b3	ERR	Error Detection Flag 0: No error has been detected. 1: Error has been detected.
b2	Reserved	When read, an initial value is returned. When written, write an initial value.
b1	FRC	Successful Frame/Wake-up Reception Flag 0: Frame or wake-up reception has not been completed. 1: Frame or wake-up reception has been completed.
b0	FTC	Successful Frame/Wake-up Transmission Flag 0: Frame or wake-up transmission has not been completed. 1: Frame or wake-up transmission has been completed.

The RLN3nLST register is automatically cleared to 00<sub>H</sub> upon transition to LIN reset mode and start of the next communication.

In LIN reset mode, the register retains 00<sub>H</sub>. Write 0 to the bit to be cleared and write 1 to other bits.

#### HTRC flag (successful header transmission flag)

Only 0 can be written to the HTRC flag; when 1 is written, the bit retains the value that has been retained before 1 is written.

The HTRC flag is set to 1 upon completion of header transmission. Here, an interrupt is generated if the SHIE bit in the RLN3nLIE register is 1 (interrupt is enabled). To clear the bit to 0 before the next communication, write 0 to the bit in LIN operation mode.

#### D1RC flag (successful data 1 reception flag)

Only 0 can be written to the D1RC flag; when 1 is written, the bit retains the value that has been retained before 1 is written.

The D1RC flag is set to 1 upon completion of data 1 reception. Here, an interrupt is not generated. To clear the bit to 0 before the next communication, write 0 to the bit in LIN operation mode.

When response data of 9 bytes or more is to be received, this bit is set to 1 each time data 1 of a data group (variable from 0 to 8 bytes) is received. Write 0 before starting reception of the next data group.

**ERR flag (error detection flag)**

The ERR flag is set to 1 upon detection of an error. Here, an interrupt is generated if the ERRIE bit in the RLN3nLIE register is 1 (interrupt is enabled). To clear the bit to 0 before the next communication, write 0 to the RPER, CSER, FER, FTER, PBER, and BER flags in the RLN3nLEST register in LIN operation mode or LIN wake-up mode. This clears the ERR flag to 0.

**FRC flag (successful frame/wake-up reception flag)**

Only 0 can be written to the FRC flag; when 1 is written, the bit retains the value that has been retained before 1 is written.

The FRC flag is set to 1 upon completion of frame or wake-up reception. Here, an interrupt is generated if the FRCIE bit in the RLN3nLIE register is 1 (interrupt is enabled). To clear the bit to 0 before the next communication, write 0 to the bit in LIN operation mode or LIN wake-up mode.

When response data of 9 bytes or more is to be received, this bit is set to 1 each time a data group (variable from 0 to 8 bytes) is received. Write 0 before starting reception of the next data group.

**FTC flag (successful frame/wake-up transmission flag)**

Only 0 can be written to the FTC flag; when 1 is written, the bit retains the value that has been retained before 1 is written.

The FTC flag is set to 1 upon completion of frame or wake-up transmission. Here, an interrupt is generated if the FTCIE bit in the RLN3nLIE register is 1 (interrupt is enabled). To clear the bit to 0 before the next communication, write 0 to the bit in LIN operation mode or LIN wake-up mode.

When response data of 9 bytes or more is to be transmitted, this bit is set to 1 each time a data group (variable from 0 to 8 bytes) is transmitted. Write 0 before starting transmission of the next data group.

### 25.3.1.15 RLN3nLEST — LIN Error Status Register

**Access:** This register can be read/written in 8-bit units.

**Address:** <RLIN3\_base> + 800<sub>H</sub> × n + 13<sub>H</sub> (n = 0, 1)

**Initial value:** 00H

Bit	7	6	5	4	3	2	1	0
	RPER	—	CSER	—	FER	FTER	PBER	BER
Initial value	0	0	0	0	0	0	0	0
R/W	R/W	R	R/W	R	R/W	R/W	R/W	R/W

**Table 25.22 RLN3nLEST register contents**

Bit Position	Bit Name	Function
b7	RPER	Response Preparation Error Flag 0: Response preparation error has not been detected. 1: Response preparation error has been detected.
b6	Reserved	When read, an initial value is returned. When written, write an initial value.
b5	CSER	Checksum Error Flag 0: Checksum error has not been detected. 1: checksum error has been detected.
b4	Reserved	When read, an initial value is returned. When written, write an initial value.
b3	FER	Framing Error Flag 0: Framing error has not been detected. 1: Framing error has been detected.
b2	FTER	Timeout Error Flag 0: Frame/response timeout error has not been detected. 1: Frame/response timeout error has been detected.
b1	PBER	Physical Bus Error Flag 0: Physical bus error has not been detected. 1: Physical bus error has been detected.
b0	BER	Bit Error Flag 0: Bit error has not been detected. 1: Bit error has been detected.

The RLN3nLEST register is automatically cleared to 00<sub>H</sub> upon transition to LIN reset mode and start of the next communication.

In LIN reset mode, the register retains 00<sub>H</sub>.

When the FTS bit in the RLN3nLTRC register is 1 (frame transmission or wake-up transmission/reception is started), do not write a value to this register. Write 0 to the bit to be cleared and write 1 to other bits.

#### **RPER flag (response preparation error flag)**

Only 0 can be written to the RPER flag; when 1 is written, the bit retains the value that has been retained before 1 is written.

The RPER flag is set to 1 upon response preparation error detection. To clear the bit to 0 before the next communication, write 0 to the bit in LIN operation mode.

#### **CSER flag (checksum error flag)**

Only 0 can be written to the CSER flag; when 1 is written, the bit retains the value that has been retained before 1 is written.

The CSER flag is set to 1 upon checksum error detection. To clear the bit to 0 before the next communication, write 0 to the bit in LIN operation mode.

**FER flag (framing error flag)**

Only 0 can be written to the FER flag; when 1 is written, the bit retains the value that has been retained before 1 is written.

The FER flag is set to 1 upon framing error detection. To clear the bit to 0 before the next communication, write 0 to the bit in LIN operation mode.

**FTER flag (timeout error flag)**

Only 0 can be written to the FTER flag; when 1 is written, the bit retains the value that has been retained before 1 is written.

The FTER flag is set to 1 upon frame timeout error or response timeout error detection. To clear the bit to 0 before the next communication, write 0 to the bit in LIN operation mode.

**PBER flag (physical bus error flag)**

Only 0 can be written to the PBER flag; when 1 is written, the bit retains the value that has been retained before 1 is written.

The PBER flag is set to 1 upon physical bus error detection. To clear the bit to 0 before the next communication, write 0 to the bit in LIN operation mode or LIN wake-up mode.

**BER flag (bit error flag)**

Only 0 can be written to the BER flag; when 1 is written, the bit retains the value that has been retained before 1 is written.

The BER flag is set to 1 upon bit error detection. To clear the bit to 0 before the next communication, write 0 to the bit in LIN operation mode or LIN wake-up mode.

### 25.3.1.16 RLN3nLDFC — LIN Data Field Configuration Register

**Access:** This register can be read/written in 8-bit units.

**Address:** <RLIN3\_base> + 800<sub>H</sub> × n + 14<sub>H</sub> (n = 0, 1)

**Initial value:** 00<sub>H</sub>

Bit	7	6	5	4	3	2	1	0
	LSS	FSM	CSM	RFT	RFDL[3:0]			
Initial value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

**Table 25.23** RLN3nLDFC register contents

Bit Position	Bit Name	Function
b7	LSS	Transmission/Reception Continuation Select 0: The data group to be transmitted/received next is the last one. 1: The data group to be transmitted/received next is not the last one. (Checksum is not included.)
b6	FSM	Frame Separate Mode Select 0: Frame separate mode is not set. 1: Frame separate mode is set.
b5	CSM	Checksum Select 0: Classic checksum mode 1: Enhanced checksum mode
b4	RFT	Response Field Communication Direction Select 0: Reception 1: Transmission
b3 to b0	RFDL[3:0]	Response Field Length Select b3 b0 0 0 0 0: 0 byte (+ checksum) 0 0 0 1: 1 byte (+ checksum) 0 0 1 0: 2 bytes (+ checksum) : 0 1 1 1: 7 bytes (+ checksum) 1 0 0 0: 8 bytes (+ checksum) Settings other than the above are prohibited.

#### LSS bit (transmission/reception continuation select bit)

The LSS bit sets the data group when response data of 9 bytes or more is to be transmitted or received. With 0 set, data and checksum are transmitted or received because the next data group to be transmitted or received is the last one.

With 1 set, only data is transmitted or received, and the checksum is not included because the next data group to be transmitted or received is not the last one.

Set the LSS bit only when the FSM bit is 1 (frame separate mode) and response data of 9 bytes or more is to be transmitted or received.

#### FSM bit (frame separate mode select bit)

The FSM bit sets the response communication mode.

With 0 set, frame separate mode is not selected. In this case, after header transmission is started (the FTS bit in the RLN3nLTRC register is 1), response is transmitted/received without the RTS bit in the RLN3nLTRC register being set.

With 1 set, frame separate mode is selected. In this case, after header transmission is started (the FTS bit in the RLN3nLTRC register is 1), response is transmitted when the following conditions are satisfied.

- Header transmission is successfully completed (the HTRC bit in the RLN3nLST register is 1).
- Response transmission is requested (the RTS bit in the RLN3nLTRC register is 1).

For response reception (the RFT bit is 0), set the FSM bit to 0.

When causing a transition to LIN self-test mode, set this bit to 0 before transition.

For details of frame separate mode, refer to **Section 25.7.3.1, (1) Frame Separate Mode**.

Set this bit when the FTS bit in the RLN3nLTRC register is 0 (frame transmission or wake-up transmission/reception is halted).

When response data of 9 bytes or more is to be transmitted or received, set the FSM bit to 1.

#### **CSM bit (checksum select bit)**

The CSM bit sets the checksum mode.

With 0 set, classic checksum mode is selected.

With 1 set, enhanced checksum mode is selected.

When the timeout error is used (the FTERE bit in the RLN3nLEDE register is 1), the specific timeout time depends on the setting of this bit. For details of the bit error, refer to **Section 25.7.6, Error Status**.

Set this bit when the FTS bit in the RLN3nLTRC register is 0 (frame transmission or wake-up transmission/reception is halted).

When response data of 9 bytes or more is to be transmitted or received, do not change the CSM bit setting after the first data group through the last data group.

During communication of response data of 9 bytes or more, only the last data group (the LSS bit is 0) includes the checksum, and no other groups (the LSS bit is 1) include the checksum.

#### **RFT bit (response field communication direction select bit)**

The RFT bits set the direction of the response field communication.

With 0 set, reception is performed in the response field. In LIN wake-up mode, wake-up reception is performed (low level width of the input signal is counted).

With 1 set, transmission is performed in the response field. In LIN wake-up mode, wake-up transmission is performed.

Set this bit when the FTS bit in the RLN3nLTRC register is 0 (frame transmission or wake-up transmission/reception is halted).

When response data of 9 bytes or more is to be transmitted or received, do not change the RFT bit setting after the first data group through the last data group.

#### **RFDL[3:0] bits (response field length select bits)**

The RFDL bits set the length of the response field data.

The data length can be 0 to 8 bytes excluding the checksum size.

To transmit response data with the FSM bit set to 0 (not frame separate mode), set the RFDL bits before header transmission (the FTS bit in the RLN3nLTRC register is 0).

To transmit response data with the FSM bit set to 1 (frame separate mode), set the RFDL bits before response transmission (the FTS bit in the RLN3nLTRC register is 0).

To receive response data, set the RFDL bits before header transmission (the FTS bit in the RLN3nLTRC register is 0).

When response data of 9 bytes or more is to be transmitted or received, set the RFDL bits before data group transmission/reception (RTS bit in the RLN3nLTRC register is 0).

During communication of response data of 9 bytes or more, only the last data group (the LSS bit is 0) includes the checksum, and no other groups (the LSS bit is 1) include the checksum.



### 25.3.1.17 RLN3nLIDB — LIN ID Buffer Register

**Access:** This register can be read/written in 8-bit units.

**Address:** <RLIN3\_base> + 800<sub>H</sub> × n + 15<sub>H</sub> (n = 0, 1)

**Initial value:** 00<sub>H</sub>

Bit	7	6	5	4	3	2	1	0
	IDP[1:0]		ID[5:0]					
Initial value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

**Table 25.24 RLN3nLIDB register contents**

Bit Position	Bit Name	Function
b7	IDP1	Parity Setting (P1) Sets the parity bit (P1) to be transmitted in the ID field.
b6	IDP0	Parity Setting (P0) Sets the parity bit (P0) to be transmitted in the ID field.
b5 to b0	ID[5:0]	ID Setting Sets the 6-bit ID value to be transmitted in the ID field.

Set the RLN3nLIDB register when the FTS bit in the RLN3nLTRC register is 0 (frame transmission or wake-up transmission/reception is halted).

#### In LIN self-test mode, this register operates as follows:

- When RFT bit is 1 (transmission):  
The reversed value of the value transmitted can be read from the register. The value to be transmitted can be written to the register before communication.
- When RFT bit is 0 (reception):  
The reversed value of the value received can be read from the register. The value to be received can be written to the register before communication.

#### IDP[1:0] bits (parity setting bits)

The IDP bits set the parity bits (P0 and P1) to be transmitted in the ID field of the LIN frame (IDP0 for P0 and IDP1 for P1). Since parity is not automatically calculated, set the calculation result. Note that if the erroneous result is set, it is transmitted as is.

#### ID[5:0] bits (ID setting bits)

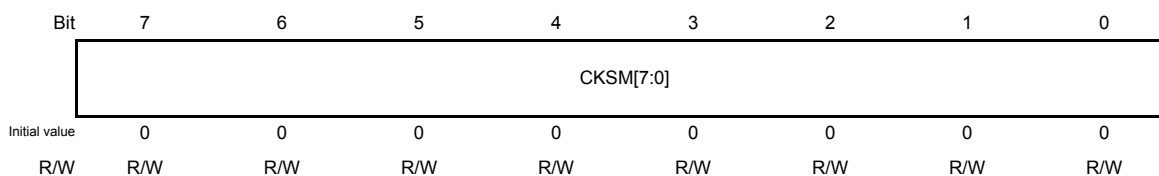
The ID bit sets the 6-bit ID value to be transmitted in the ID field of the LIN frame.

### 25.3.1.18 RLN3nLCBR — LIN Checksum Buffer Register

**Access:** This register can be read/written in 8-bit units.

**Address:** <RLIN3\_base> + 800<sub>H</sub> × n + 16<sub>H</sub> (n = 0, 1)

**Initial value:** 00<sub>H</sub>



**Table 25.25** RLN3nLCBR register contents

Bit Position	Bit Name	Function
b7 to b0	CKSM[7:0]	Holds the checksum value transmitted or received.

In LIN mode, this register operates as follows:

- When the RFT bit in the RLN3nLDFC register is 1 (transmission):  
The value transmitted can be read from the register. Read the value after transmission is completed.  
Writing to this register is invalid.
- When the RFT bit in the RLN3nLDFC register is 0 (reception):  
The value received can be read from the register. Read the value after reception is completed.  
Writing to this register is invalid.

In LIN self-test mode, this register operates as follows:

- When the RFT bit in the RLN3nLDFC register is 1 (transmission):  
The reversed value of the value transmitted can be read from the register. Read the value after transmission is completed.  
Writing to this register is invalid.
- When the RFT bit in the RLN3nLDFC register is 0 (reception):  
The reversed value of the value received can be read from the register. Read the value after reception is completed.  
The value to be received can be written to the register before communication.

Set the RLN3nLCBR register when the FTS bit in the RLN3nLTRC register is 0 (frame transmission or wake-up transmission/reception is halted).

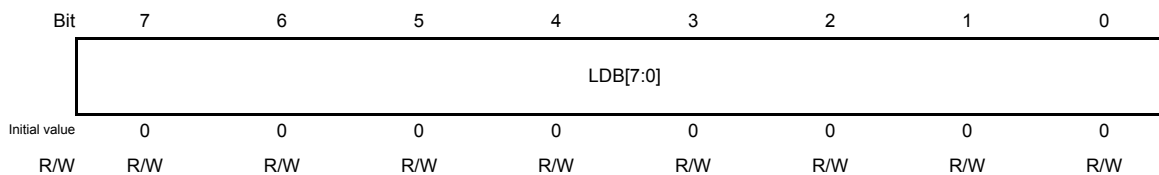
When response data of 9 bytes or more is to be transmitted or received, the checksum is appended only to the last data group; this register is not updated for the other data groups.

### 25.3.1.19 RLN3nLDBRm — LIN Data Buffer m Register (m = 1 to 8)

**Access:** This register can be read/written in 8-bit units.

**Address:** RLN3nLDBR1: <RLIN3\_base> + 800<sub>H</sub> × n + 18<sub>H</sub> (n = 0, 1)  
 RLN3nLDBR2: <RLIN3\_base> + 800<sub>H</sub> × n + 19<sub>H</sub> (n = 0, 1)  
 RLN3nLDBR3: <RLIN3\_base> + 800<sub>H</sub> × n + 1A<sub>H</sub> (n = 0, 1)  
 RLN3nLDBR4: <RLIN3\_base> + 800<sub>H</sub> × n + 1B<sub>H</sub> (n = 0, 1)  
 RLN3nLDBR5: <RLIN3\_base> + 800<sub>H</sub> × n + 1C<sub>H</sub> (n = 0, 1)  
 RLN3nLDBR6: <RLIN3\_base> + 800<sub>H</sub> × n + 1D<sub>H</sub> (n = 0, 1)  
 RLN3nLDBR7: <RLIN3\_base> + 800<sub>H</sub> × n + 1E<sub>H</sub> (n = 0, 1)  
 RLN3nLDBR8: <RLIN3\_base> + 800<sub>H</sub> × n + 1F<sub>H</sub> (n = 0, 1)

**Initial value:** 00<sub>H</sub>



**Table 25.26** RLN3nLDBRm (m = 1 to 8) register contents

Bit Position	Bit Name	Function
b7 to b0	LDB[7:0]	Sets the data to be transmitted or allows the received data to be read. Setting Range: 00 <sub>H</sub> to FF <sub>H</sub>

#### For response transmission:

The LDBRn registers set the data to be transmitted in the response field.

Use these registers with the following settings.

- RFT in RLN3nLDFC register is 1 (transmission)
- FSM in RLN3nLDFC register is 0 (not frame separate mode)
- FTS bit in RLN3nLTRC register is 0 (frame transmission or wake-up transmission/reception is halted)

or

- RFT in RLN3nLDFC register is 1 (transmission)
- FSM in RLN3nLDFC register is 1 (frame separate mode)
- RTS in RLN3nLTRC register is 0 (response transmission/reception is halted)

#### For response reception:

The LDBRn registers hold the data received in the response field.

The received data is overwritten. If an error is detected, the data prior to reception interruption is stored in the register.

Do not read these registers when the FTS bit is 1 (frame transmission or wake-up transmission/reception is started)

**For transmission of response data of 9 bytes or more:**

Use the LDBRn registers with the following settings.

- RFT in RLN3nLDFC register is 1 (transmission)
- FSM in RLN3nLDFC register is 1 (frame separate mode)
- RTS in RLN3nLTRC register is 0 (response transmission/reception is halted)

**For reception of response data of 9 bytes or more:**

Do not read these registers when the RTS bit is 1 (response transmission/reception is started).

**In LIN self-test mode, these registers operate as follows:**

- When the RFT bit in the RLN3nLDFC register is 1 (transmission):  
The reversed value of the value transmitted can be read from the registers. The value to be transmitted can be written to the registers before communication.
- When the RFT bit in the RLN3nLDFC register is 0 (reception):  
The reversed value of the value received can be read from the register. The value to be received can be written to the registers before communication.

## 25.4 Interrupt Sources

The LIN Interface generates four types of interrupt requests.

- LINn transmission interrupt
- LINn successful reception interrupt
- LINn status interrupt
- LINn interrupt

Setting the LIOS bit in the RLN3nLMD register to 0 allows to perform logical OR operation on all of the interrupt sources, outputting the interrupt request from the LINn interrupt.

Setting the LIOS bit in the RLN3nLMD register to 1 allows to output the LINn transmission interrupt, LINn successful reception interrupt, or LINn status interrupt depending on the interrupt request.

**Table 25.27** lists the sources for each interrupt.

**Table 25.27** Interrupt Sources

		LIOS bit in RLN3nLMD register is 0		LIOS bit in RLN3nLMD register is 1	
		LINn Interrupt	LINn Transmission Interrupt	LINn Successful Reception Interrupt	LINn Status Interrupt
LIN mode	LIN master mode	<ul style="list-style-type: none"> <li>• Successful frame transmission</li> <li>• Successful frame reception</li> <li>• Successful wake-up transmission</li> <li>• Successful wake-up reception</li> <li>• Successful header transmission</li> <li>• Bit error</li> <li>• Physical bus error</li> <li>• Frame/response timeout error</li> <li>• Framing error</li> <li>• Checksum error</li> <li>• Response preparation error</li> </ul>	<ul style="list-style-type: none"> <li>• Successful frame transmission</li> <li>• Successful wake-up transmission</li> <li>• Successful header transmission</li> </ul>	<ul style="list-style-type: none"> <li>• Successful wake-up reception</li> <li>• Successful wake-up reception</li> </ul>	<ul style="list-style-type: none"> <li>• Bit error</li> <li>• Physical bus error</li> <li>• Frame/response timeout error</li> <li>• Framing error</li> <li>• Checksum error</li> <li>• Response preparation error</li> </ul>

Each interrupt request is output when the corresponding bit in the RLN3nLIE register is 1 (interrupt is enabled) and the corresponding flag in the RLN3nLST register is 1.

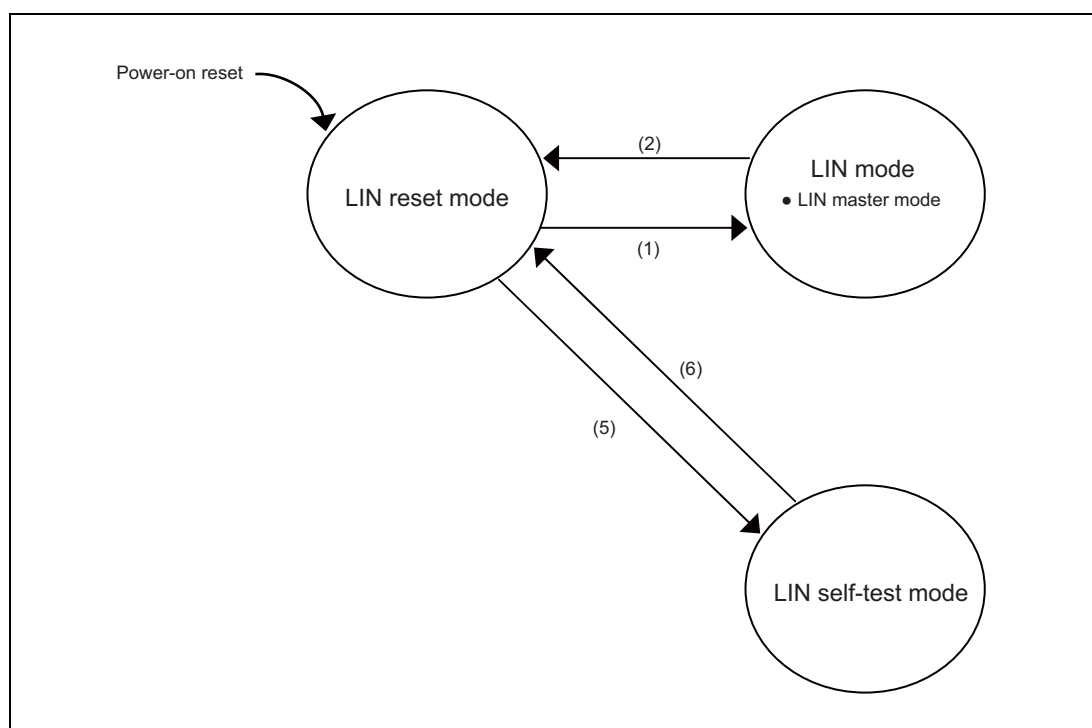
## 25.5 Modes

The LIN Interface provides the following three modes, depending upon the specific function to be performed:

- LIN reset mode
- LIN mode (LIN master mode)
- LIN self-test mode

The supply of clocks to the LIN Interface is stopped in LIN reset mode, which reduces power consumption.

**Figure 25.2** shows mode transitions. **Table 25.28** describes mode transition conditions. **Table 25.29** lists operations available in each mode.



**Figure 25.2** Mode Transitions

**Table 25.28** Transition Condition of Each Mode

Mode transition	Transition condition
1 LIN reset mode → LIN mode (LIN master mode)	LMD bit in RLN3nLMD register = 00 <sub>B</sub> and OM1 and OM0 bits in RLN3nLCUC register = 01 <sub>B</sub> or 11 <sub>B</sub>
2 LIN mode → LIN reset mode	OM0 bit in RLN3nLCUC register = 0 <sub>B</sub>
5 LIN reset mode → LIN self-test mode	See <b>Section 25.8, LIN Self-Test Mode.</b>
6 LIN self-test mode → LIN reset mode	See <b>Section 25.8, LIN Self-Test Mode.</b>

Table 25.29 Operations Available in Each Mode

LIN mode	
LIN master mode	LIN self-test mode
Header transmission	Self test
Response transmission	
Response reception	
Wake-up transmission	
Wake-up reception	
Error detection	

Whether a transition has been caused to LIN reset mode or LIN mode can be verified by reading the LMD bits in the RLN3nLMD register or the OMM0 bit in the RLN3nLMST register.

For a description of the LIN self-test mode, see **Section 25.8, LIN Self-Test Mode**.

## 25.6 LIN Reset Mode

Setting the OM0 bit in the RLN3nLCUC register to 0b (LIN reset mode) causes a transition to LIN reset mode. The change to LIN reset mode can be verified by determining that the OMM0 bit in the RLN3nLMST register has been set to 0<sub>B</sub> (LIN reset mode). In this mode, the LIN communication function is halted.

From LIN reset mode, transitions to LIN mode and LIN self-test mode can be made.

When the mode changes to LIN reset mode, the following registers are initialized to their reset values, and as long as LIN reset mode is in effect, they retain their initial values.

- RLN3nLTRC register
- RLN3nLST register
- RLN3nLEST register

The following registers retain their previous values even when a transition to LIN reset mode is made:

- RLN3nLWBR register
- RLN3nLBRP0 register
- RLN3nLBRP1 register
- RLN3nLMD register
- RLN3nLBFC register
- RLN3nLSC register
- RLN3nLWUP register
- RLN3nLIE register
- RLN3nLEDE register
- RLN3nLDFC register
- RLN3nLIDB register
- RLN3nLCBR register
- RLN3nLDBRm register (m = 1 to 8)



## 25.7 LIN Mode

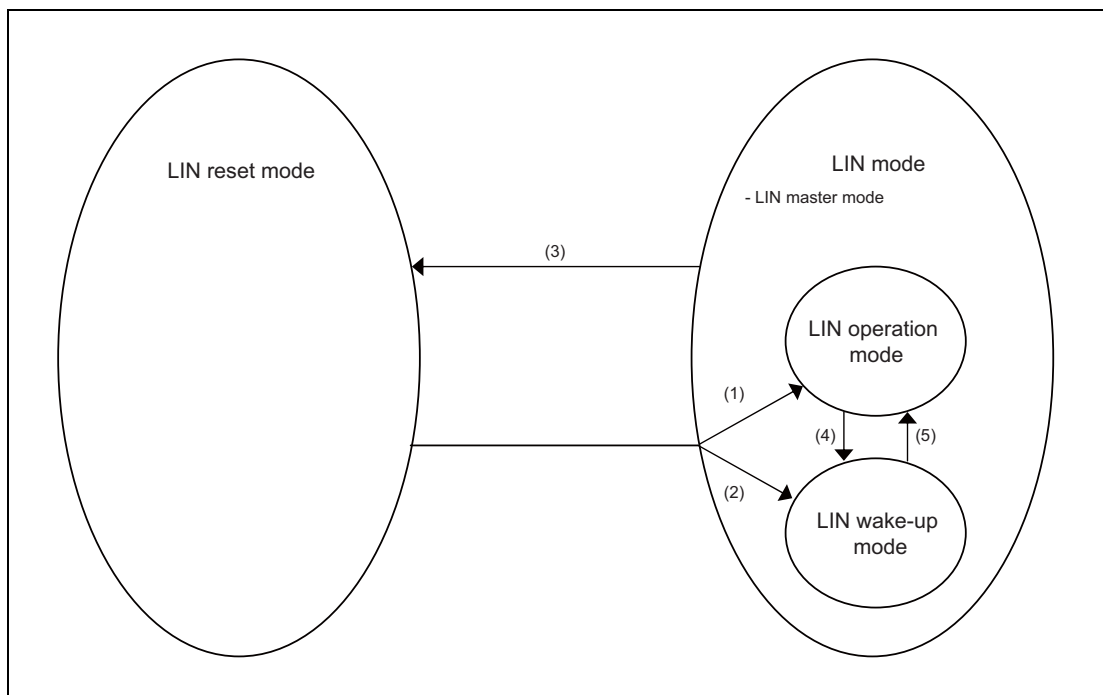
LIN mode can operate in the LIN master mode.

In LIN master mode, the following operations can be performed: header transmission, response transmission, response reception, wake-up transmission, wake-up reception, and error detection. In LIN reset mode, setting the LMD bits in the RLN3nLMD register to  $00_B$  (LIN master mode) and the OM1 and OM0 bits in the RLN3nLCUC register to either  $01_B$  or  $11_B$  sets LIN master mode, turning the OMM1 and OMM0 bits in the RLN3nLMST register to either  $01_B$  to  $11_B$ .

The LIN mode provides the following two operation modes:

- LIN operation mode
- LIN wake-up mode

**Figure 25.3** shows the transition of operation modes. **Table 25.30** describes the transition conditions of operation modes.



**Figure 25.3** Transition of Operation Modes

Table 25.30 Transition Conditions of Operation Modes

Operation mode transition		Transition condition
(1) LIN reset mode	→ LIN mode - LIN operation mode	LMD bit in RLN3nLMD register = 00 <sub>B</sub> and OM1 and OM0 bits in RLN3nLCUC register = 11 <sub>B</sub>
(2) LIN reset mode	→ LIN mode - LIN wake-up mode	LMD bit in RLN3nLMD register = 00 <sub>B</sub> and OM1 and OM0 bits in RLN3nLCUC register = 01 <sub>B</sub>
(3) LIN mode -LIN operation mode -LIN wake-up mode	→ LIN reset mode	OM0 bit in LCUC register = 0 <sub>B</sub>
(4) LIN mode *1 -LIN operation mode	→ LIN mode - LIN wake-up mode	OM1 and OM0 bits in RLN3nLCUC register = 01 <sub>B</sub>
(5) LIN mode *1 -LIN wake-up mode	→ LIN mode - LIN operation mode	OM1 and OM0 bits in RLN3nLCUC register = 11 <sub>B</sub>

Note 1. Transition between LIN operation mode and LIN wake-up mode cannot be made when communication is going on (when the FTS bit in the RLN3nLTRC register is 1).

### (1) LIN Operation Mode

In LIN operation mode, frame processing (header transmission, header reception, response transmission, response reception, and error detection) can be performed.

During a transition from LIN reset mode to LIN mode, setting the OM1 and OM0 bits in the RLN3nLCUC register to 11<sub>B</sub> changes the mode to LIN operation mode, changing the OMM1 and OMM0 bits in the RLN3nLMST register to 11<sub>b</sub>. Communication settings should be performed after the RLN3nLMST register has become 11<sub>B</sub>.

### (2) LIN Wake-up Mode

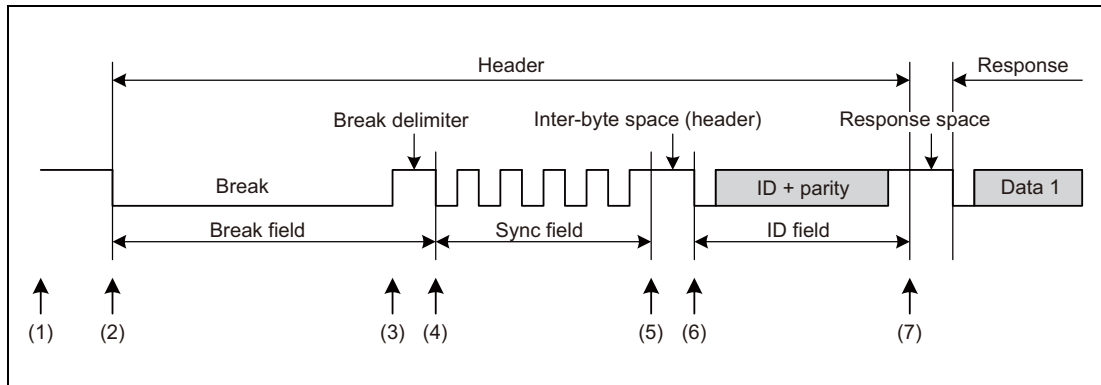
In LIN wake-up mode, wake-up signal processing (wake-up transmission, wake-up reception, and error detection) can be performed.

During a transition from LIN reset mode to LIN mode, setting the OM1 and OM0 bits in the RLN3nLCUC register to 01<sub>B</sub> changes the mode to LIN wake-up mode, changing the OMM1 and OMM0 bits in the RLN3nLMST register to 01<sub>B</sub>. Communication settings should be performed after the RLN3nLMST register has become 01<sub>B</sub>.

## 25.7.1 LIN Master Mode

### 25.7.1.1 Header Transmission

**Figure 25.4** shows the operation of the LIN Interface (LIN master mode) in header transmission. **Table 25.31** provides processing in header transmission.



**Figure 25.4** Operation in Header Transmission

**Table 25.31** Processing in Header Transmission

Software processing	LIN Interface processing
(1) <ul style="list-style-type: none"> <li>• Sets a baud rate</li> <li>• Sets noise filter ON/OFF</li> <li>• Enables interrupt</li> <li>• Enables error detection</li> <li>• Sets frame configuration parameters</li> <li>• Changes the LIN Interface to the LIN master mode: LIN operation mode</li> <li>• Sets information on the frame to be transmitted (ID, parity, data length, response direction, Checksum method, and transmission data)</li> </ul>	Waits for the setting of the FTS bit in the RLN3nLTRC register by software (idle)
(2) Sets the FTS bit in the RLN3nLTRC register to 1 (frame transmission or wake-up transmission/reception started)	Transmits a break.
(3) Waits for an interrupt request	Transmits a break delimiter.
(4)	Transmits a sync field (55h).
(5)	Transmits an inter-byte space (header).
(6)	Transmits an ID field.
(7)	Sets a successful header transmission flag.

**Note:** For information about error detection, refer to **Section 25.7.6, Error Status**.

### 25.7.1.2 Response Transmission

Figure 25.5 shows the operation of the LIN Interface (LIN master mode) in response transmission.

Table 25.32 provides processing in response transmission.

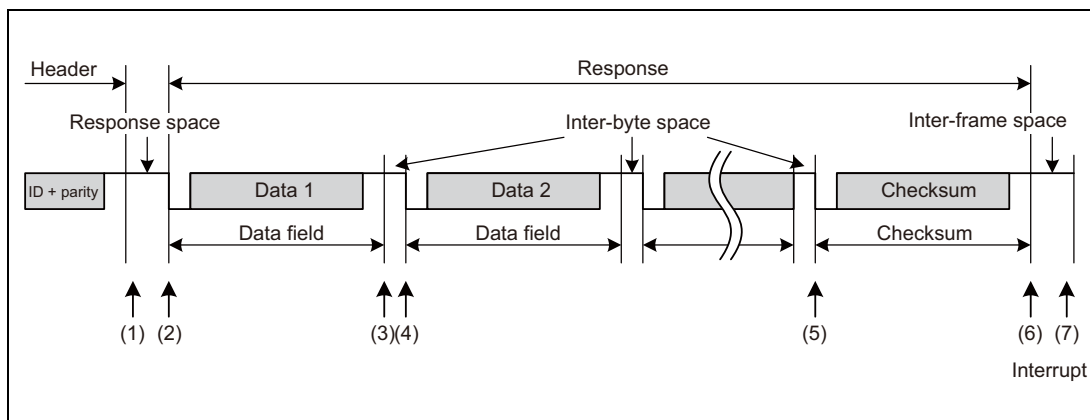


Figure 25.5 Operation in Response Transmission

Table 25.32 Processing in Response Transmission

Software processing	LIN Interface processing
(1) (When in frame separate mode) <ul style="list-style-type: none"> <li>Sets the RTS bit in the RLN3nLTRC register to 1 (response transmission/reception started)</li> </ul> (When not in frame separate mode) <ul style="list-style-type: none"> <li>Waits for an interrupt request</li> </ul>	(When in frame separate mode) <ul style="list-style-type: none"> <li>Waits for the setting of the RTS bit in the RLN3nLTRC register to 1 by software.</li> <li>When the bit is set to 1, sends a response space.</li> </ul> (When not in frame separate mode) <ul style="list-style-type: none"> <li>Sends a response space.</li> </ul>
(2) Waits for an interrupt request	Transmits the data 1.
(3)	Transmits an inter-byte space.
(4)	<ul style="list-style-type: none"> <li>Transmits the data 2.</li> <li>Transmits an inter-byte space</li> <li>Transmits the data 3.</li> <li>Transmits an inter-byte space</li> </ul> (Repeats the transmission of inter-byte spaces as many times as the data length specified in bits RFDL[3:0] in the RFC register, and stops the transmission when the BER flag in the RLN3nLEST register is 1 (bit error detected). If an error occurs, does not perform the Checksum transmission in item (5)). : :
(5)	Transmits the checksum.
(6)	<ul style="list-style-type: none"> <li>Sets a successful frame/wake-up transmission flag.</li> <li>Sets the FTS bit in the RLN3nLTRC register to 0 (frame transmission or wake-up transmission/reception stopped) (When in frame separate mode), and the RTS bit in the RLN3nLTRC register to 0 (response transmission/reception stopped).</li> </ul>
(7) <ul style="list-style-type: none"> <li>Processing after communication</li> <li>Checks the RLN3nLST register, and clears flags.</li> </ul>	Idle

**Note:** For information about error detection, refer to Section 25.7.6, Error Status.

### 25.7.1.3 Response Reception

Figure 25.6 shows the operation of the LIN Interface (LIN master mode) on response reception.

Table 25.33 provides processing in response reception.

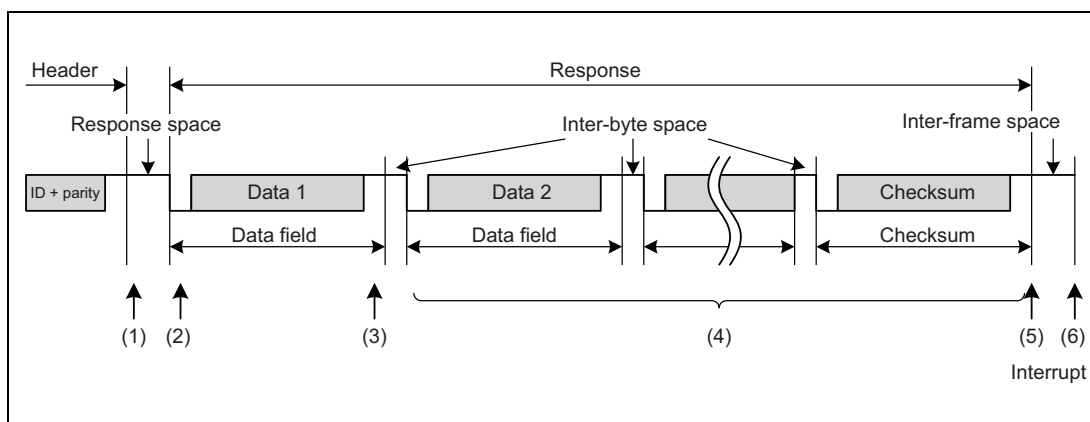


Figure 25.6 Operation in Response Reception

Table 25.33 Processing in Response Reception

Software processing	LIN Interface processing
(1) Waits for an interrupt request (no processing).	Waits for detection of a start bit.
(2) Waits for an interrupt request.	Receives the data 1 when the start bit is detected.
(3)	Sets the successful data 1 reception flag.
(4)	<ul style="list-style-type: none"> <li>Receives the data 2 when the start bit is detected.</li> <li>Receives the data 3 when the start bit is detected.</li> </ul> Repeats the transmission of inter-byte spaces as many times as the data length specified in bits RFDL[3:0] in the RLN3nLDFC register, and stops the transmission when any bit in the RLN3nLEST register is 1 (bit error detected). If an error occurs, the checksum determination in item (5) is not performed). : : <ul style="list-style-type: none"> <li>Receives the checksum when the start bit is detected.</li> </ul>
(5)	<ul style="list-style-type: none"> <li>Determines the checksum.</li> <li>Sets the successful frame/wake-up reception flag.</li> <li>Sets the RTS bit in the RLN3nLTRC register to 0 (response transmission/reception stopped).</li> </ul>
(6) <ul style="list-style-type: none"> <li>Processing after communication</li> <li>Reads the received data.</li> <li>Checks the RLN3nLST register, and clears flags.</li> </ul>	Idle

**Note:** For information about error detection, refer to Section 25.7.6, Error Status.

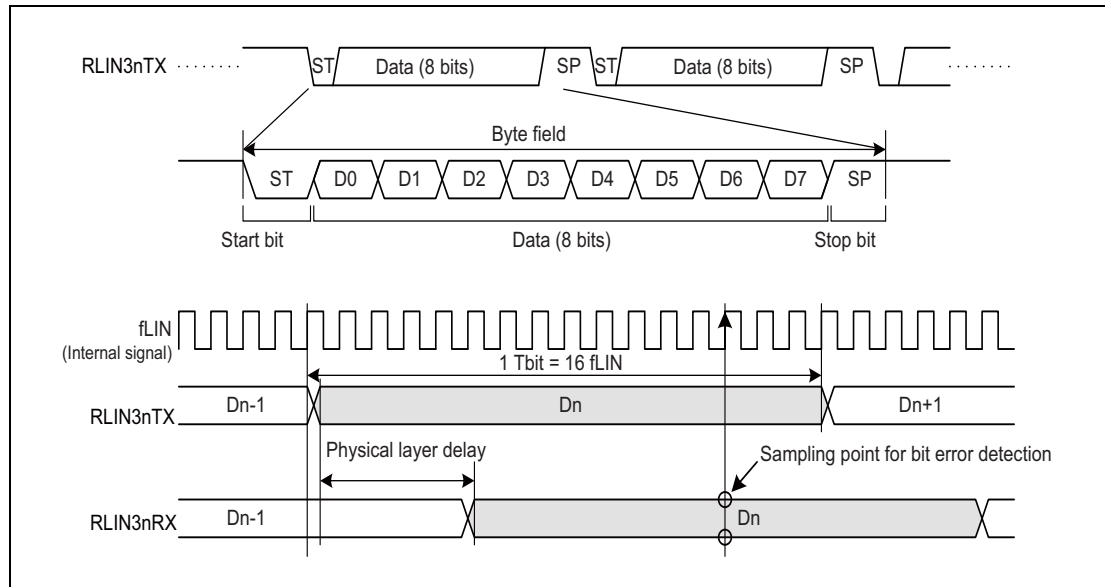
## 25.7.2 Data Transmission/Reception

### 25.7.2.1 Data Transmission

One bit of data is transmitted per 1 Tbit.

The data that is transmitted returns to the reception data input pin via the LIN transceiver. The received data and the transmitted data is compared bit by bit, and the results are stored in the BER flag in the RLIN3nLEST register (see **Section 25.7.6, Error Status**).

**Figure 25.7** shows an example of data transmission timing.



**Figure 25.7** Example of Data Transmission Timing (LIN Master Mode)

### 25.7.2.2 Data Reception

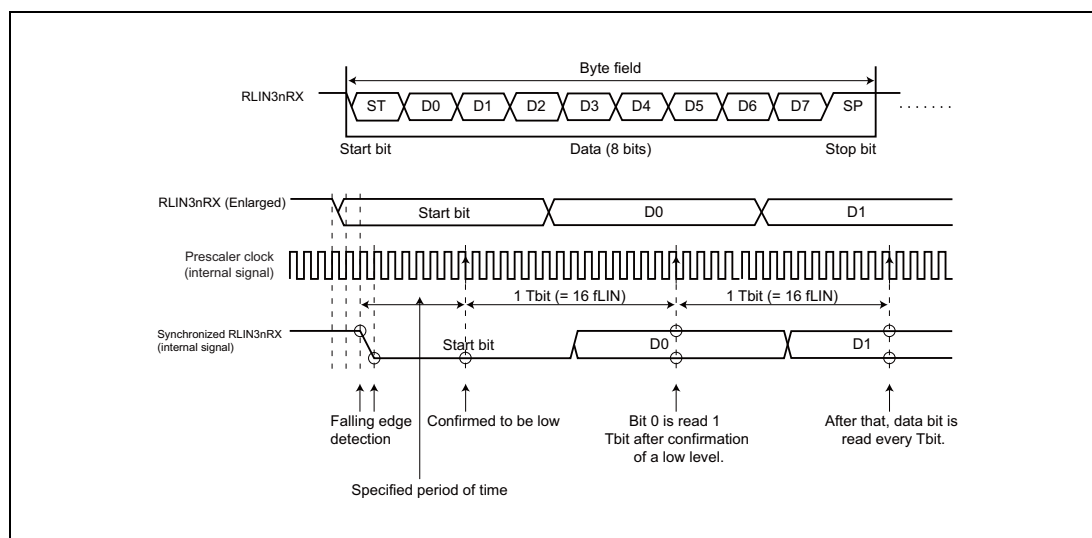
Data reception is performed by using the synchronized RLIN3nRX signal (an internal signal) that is the input from the RLIN3nRX pin synchronized with prescaler clock.

The byte field is synchronized at the falling edge of the start bit for the synchronized RLIN3nRX signal. After the falling edge is detected, sampling is performed again a specified period of time later, and the falling edge is recognized as a start bit if the synchronized RLIN3nRX signal is low level. The falling edge is not recognized as a start bit if the RLIN3nRX signal after the clearing of the resetting is low-level-fixed or if a high level is detected on re-sampling.

After the start bit is detected, the system samples 1 bit per Tbit.

The LIN Interface has a noise filter function with respect to reception data. If the LRDNFS bit in the RLIN3nLMD register is 0, the LIN Interface uses a noise filter, and for a sampling value the value determined by a 3-sampling majority rule on prescaler clocks is used. If the LRDNFS bit in the RLIN3nLMD register is 1, the LIN Interface does not use a noise filter, and for a sampling value the value of the synchronized RLIN3nRX value at the sampling position is used as is.

**Figure 25.8** shows an example of data reception timing.



**Figure 25.8** Example of Data Reception Timing

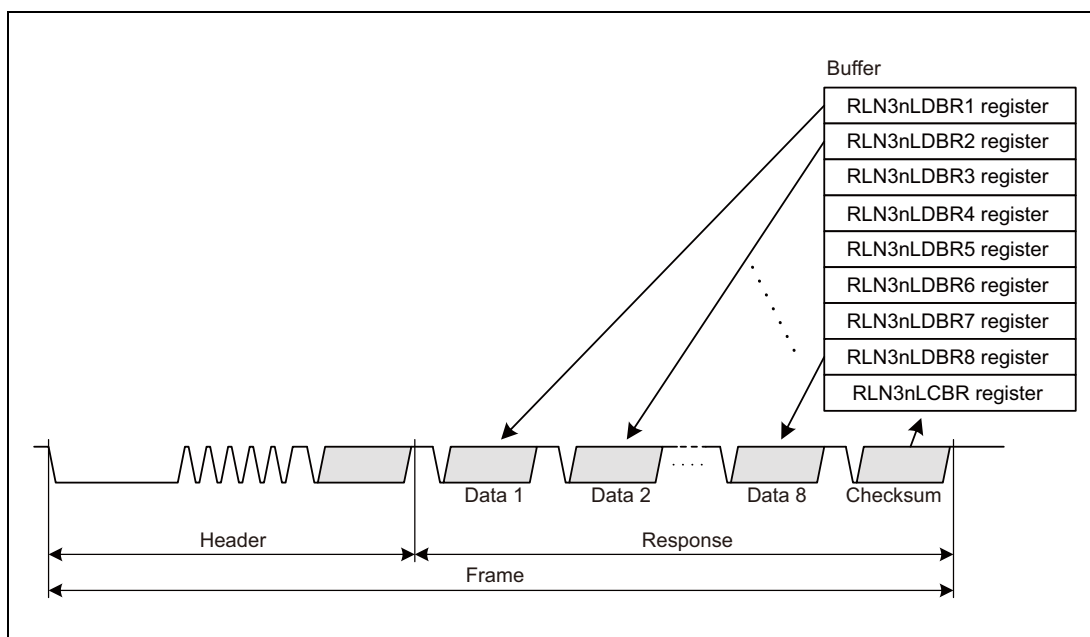
### 25.7.3 Transmission/Reception Data Buffering

This section explains the buffer processing that takes place when the LIN Interface sends or receives data continuously.

#### 25.7.3.1 Transmission of LIN frames

For an 8-byte transmission, the contents stored in registers RLN3nLDBR1 to RLN3nLDBR8 are sequentially transmitted to data areas 1 to 8 of the LIN frame. In the case of a 4-bytes transmission, the contents stored in registers RLN3nLDBR1 to RLN3nLDBR4 are transmitted to data areas 1 to 4 of the LIN frame, but the contents of registers RLN3nLDBR5 to RLN3nLDBR8 are not transmitted. The transmitted checksum data is stored in the RLN3nLCBR register.

**Figure 25.9** depicts the LIN transmission processing and the required buffer.



**Figure 25.9** LIN Transmission Processing and Required Buffer

#### (1) Frame Separate Mode

Setting the FSM bit in the RLN3nLDFC register to 1 turns on the frame separate mode.

In frame separate mode, a header and a response are transmitted when prompted by separate transmission start requests.

When the transmission of a header is finished, the HTRC flag in the RLN3nLST register turns 1 (successful header transmission).

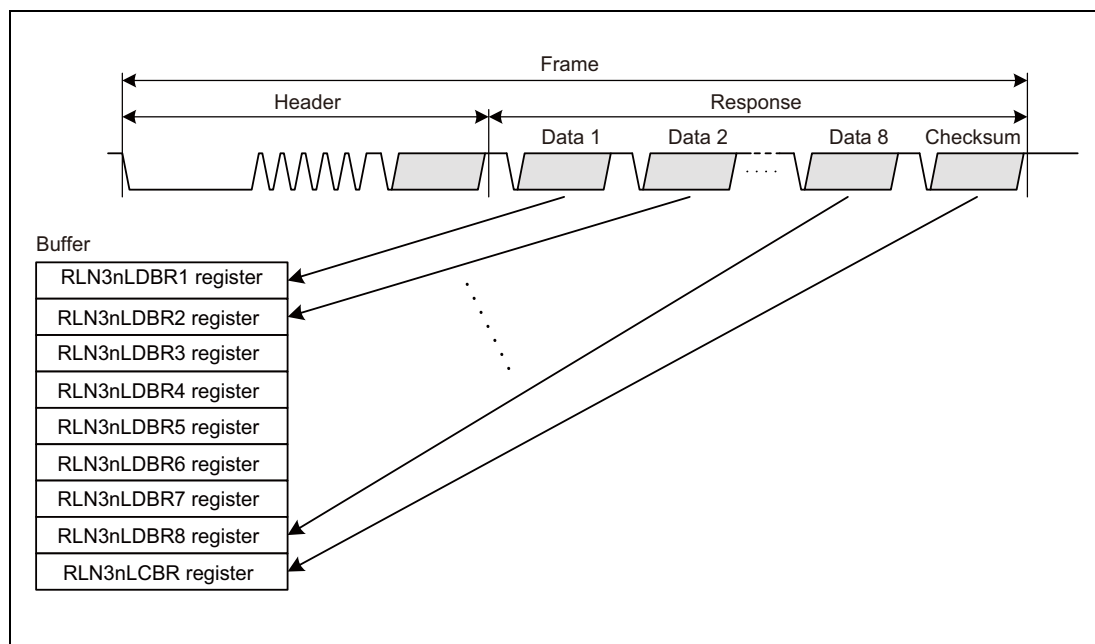
Use frame separate mode when sending or receiving response data of 9 bytes or greater in LIN master mode.



### 25.7.3.2 Reception of LIN Frames

For an 8-byte reception, the contents of data areas 1 to 8 of the LIN frame is stored in registers RLN3nRLN3nLDBR1 to RLN3nLDBR8, respectively, upon receipt of a stop bit. In the case of a 4-byte reception, the contents of data areas 1 to 4 of the LIN frame are stored in registers RLN3nLDBR1 to RLN3nLDBR4, respectively; however, no data is stored in registers RLN3nLDBR5 to RLN3nLDBR8. Also, the received checksum data is stored in the RLN3nLCBR register.

**Figure 25.10** depicts the LIN reception processing and the required buffer.



**Figure 25.10** LIN Reception Processing and Required Buffer

#### (1) Reception of Data 1

When the reception of the first byte of data is finished, the D1RC flag in the RLN3nLST register turns 1 (successful data 1 reception).

### 25.7.3.3 Multi-Byte Response Transmission/Reception Function

Normally in LIN communications, a response is 9 bytes or less (including a checksum field); however, responses in 10 bytes or greater can also be sent and received.

In such a case, the bit error, framing error, response preparation error detection, and auto checksum functions are enabled.

If the data length is greater than 8 bytes, the LSS bit in RLN3nLDFC register should be set to 1 (indicating that the next data group to be sent or received is not the final data group) in the first data group (variable in 0 to 8 bytes) before sending or receiving the data group. After the transmission or reception, the user should determine whether the next data group is the final data group. If it is the final data group, the LSS bit should be set to 0 (indicating that the next data group to be sent or received is the final data group, and a checksum should be appended to the final data group).

By changing the RFDL bit in RLN3nLDFC register settings when the RTS bit in RLN3nLTRC register is 0, the user can change the data length for each data group.

When performing multi-byte response transmission/reception in LIN master mode, set the FSM bit in RLN3nLDFC register in the RLN3nLDFC register to 1 (frame separate mode).

## 25.7.4 Wake-up Transmission/Reception

The wake-up transmission/reception can be used in LIN wake-up mode.

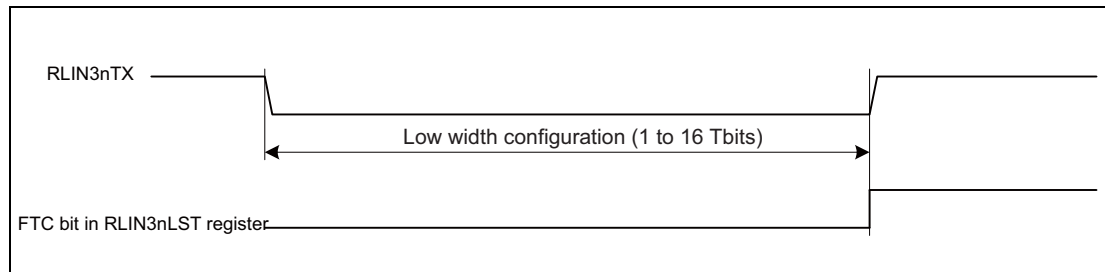
### 25.7.4.1 Wake-up Transmission

In LIN wake-up mode, setting the RCDS bit in the RLIN3nLDFC register to 1 (transmission) and the FTS bit in the RLIN3nLTRC register to 1 (header reception or wake-up transmission/reception started) causes a wake-up signal to be output from the output pin. The low level width of the wake-up signal should be set using the WUTL[3:0] bits in the RLIN3nLWUP register.

If a wake-up low is output without any bit error, the FTC flag in the RLIN3nLST register turns 1 (successful response or wake-up transmission); when the FTCIE bit in the RLIN3nLIE register is 1 (successful response/wakeup transmission interrupt enabled), an interrupt request is generated.

If a bit error is detected, wake-up transmission is canceled and the BER flag in the RLIN3nLEST register is set to 1 (bit error detection).

**Figure 25.11** shows the wake-up transmission timing.



**Figure 25.11** Wake-up Transmission Timing

### 25.7.4.2 Wake-up Reception

The detection of a wake-up involves the use of an input signal low level width count function.

The input signal low level width count function measures the low width of the input signal to the RLIN3nTX pin, using the same sampling point as data reception. The function can measure the input signal low level width of 2.5 Tbits or greater.

When using this function, in LIN wake-up mode set the RCDS bit in the RLN3nLDFC register to 0 (reception), and the FTS bit in the RLN3nLTRC register to 1 (header reception or wake-up transmission/reception a started).

When the low level width to be measured is reached, the FRC flag in the RLN3nLST register turns 1 (successful response/wake-up reception). If the FRCIE bit in the RLN3nLIE register is 1 (successful response or wake-up reception interrupt enabled), an interrupt request is generated.

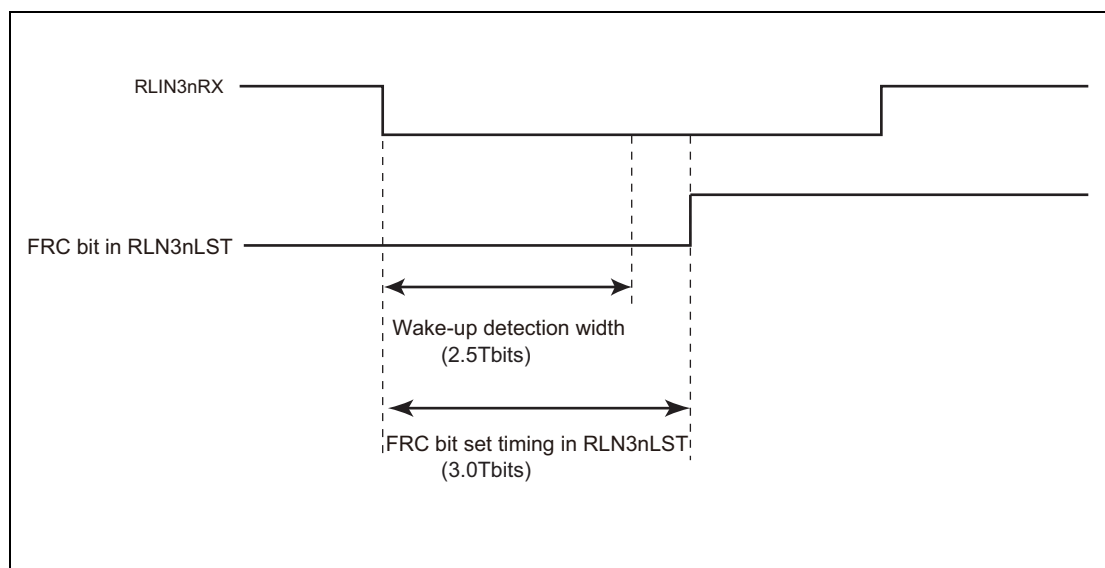


Figure 25.12 Input Signal Low level Count Function

## 25.7.5 Status

During LIN mode operation, the LIN Interface can detect seven types of statuses.

The four statuses, successful frame/wake-up transmission, successful frame/wake-up reception, error detection, successful header transmission/header reception, can generate interrupt requests.

**Table 25.34** shows the types of statuses available in LIN master mode.

**Table 25.34** Types of Statuses in LIN Master Mode

Status	Status set condition	Status clear condition	Operation mode capable of status detection	Corresponding bit	Interrupt
Reset	After the OM0 bit in the RLN3nLCUC register is set to LIN reset mode (OM0 = 0), if the LIN Interface enters LIN reset mode.	After the OM0 bit in the RLN3nLCUC register is set to 0 (LIN reset mode is canceled), if LIN reset mode is canceled.	All modes	OMM0 bit in RLN3nLMST register	—
Operation mode	After the OM1 bit in the RLN3nLCUC register is set to LIN operation mode, if actually the LIN Interface enters LIN operation mode.	After the OM1 bit in the RLN3nLCUC register is set to LIN wake-up mode, if actually the LIN Interface enters LIN wake-up mode.	<ul style="list-style-type: none"> <li>• LIN operation mode</li> <li>• LIN wake-up mode</li> </ul>	OMM1 bit in RLN3nLMST register	—
Frame/wake-up transmission end	When a frame (header transmission + response transmission), a wake-up signal, or a data group is transmitted successfully.	<ul style="list-style-type: none"> <li>• When another communication is started</li> <li>• When cleared by software</li> <li>• After transition to LIN reset mode</li> </ul>	<ul style="list-style-type: none"> <li>• LIN operation mode</li> <li>• LIN wake-up mode</li> </ul>	FTC flag in RLN3nLST register	√
Frame/wake-up reception end	When a frame (header transmission + response reception), a wake-up signal, or a data group is received successfully.	<ul style="list-style-type: none"> <li>• When another communication is started</li> <li>• When cleared by software</li> <li>• After transition to LIN reset mode</li> </ul>	<ul style="list-style-type: none"> <li>• LIN operation mode</li> <li>• LIN wake-up mode</li> </ul>	FRC flag in RLN3nLST register	√
Error detection	If any of the PRER flag, CSER flag, FER flag, FTER flag, PBER flag, and BER flags in the RLN3nLEST register turns 1 (error detected).	<ul style="list-style-type: none"> <li>• When another communication is started</li> <li>• When cleared by software</li> <li>• After transition to LIN reset mode</li> </ul>	<ul style="list-style-type: none"> <li>• LIN operation mode</li> <li>• LIN wake-up mode</li> </ul>	ERR flag in RLN3nLST register	√
Data 1 reception end	The RFT bit in the RLN3nLDFC register is 0 (reception) and the first byte of the response field is received.*2	<ul style="list-style-type: none"> <li>• When another communication is started</li> <li>• When cleared by software</li> <li>• After transition to LIN reset mode</li> </ul>	LIN operation mode	D1RC flag in RLN3nLST register	—
Header transmission end	When a header field is transmitted successfully.	<ul style="list-style-type: none"> <li>• When another communication is started</li> <li>• When cleared by software</li> <li>• After transition to LIN reset mode</li> </ul>	LIN operation mode	HTRC flag in RLN3nLST register	√

Note 1. In LIN operation mode, the ERR flag in the RLN3nLST register is cleared to 0 by writing 0 to the PRER flag, CSER flag, FER flag, FER flag, FTER flag, PBER flag or BER flags in the RLN3nLEST register.

Note 2. Not detected when the RFDL [3:0] bits in the RLN3nLDFC register are 0000<sub>B</sub> (0-byte + checksum).

## 25.7.6 Error Status

### 25.7.6.1 LIN Master Mode

#### (1) Types of Error Statuses

The LIN Interface can detect six types of error statuses in LIN master mode. The condition of these error statuses can be checked by means of the corresponding bits in the RLN3nLEST register.

All error statuses represent interrupt events.

**Table 25.35** shows the types of error statuses.

**Table 25.35** Types of Error Statuses in LIN Master Mode

Status	Error detection condition	Operation mode capable of error detection	Communication	Enable/disable detection	Corresponding bit
Bit error	The transmitted data and the data on the LIN bus monitored by the receive pin do not match *1	<ul style="list-style-type: none"> <li>LIN operation mode</li> <li>LIN wake-up mode</li> </ul>	Cancel	√	BER flag in RLN3nLEST register
Physical bus error	<ul style="list-style-type: none"> <li>LIN bus is detected to be high level when sending a break</li> <li>LIN bus is detected to be low level when sending a break delimiter</li> <li>LIN bus is detected to be high level when sending a wake-up</li> </ul>	<ul style="list-style-type: none"> <li>LIN operation mode</li> <li>LIN wake-up mode</li> </ul>	Cancel	√	PBER flag in RLN3nLEST register
Timeout error	A frame or response transmission/reception does not terminate within a given time*2	LIN operation mode	Cancel	√	FTER flag in RLN3nLEST register
Framing error	In response field reception, a stop bit of each data byte is low level	LIN operation mode	Cancel	√	FER flag in RLN3nLEST register
Checksum error	In response field reception, the result of checksum test gives an error	LIN operation mode	—	×	CSER flag in RLN3nLEST register
Response preparation error	<p>One of the following conditions occurs in frame separate mode during a multi-byte response reception:</p> <ul style="list-style-type: none"> <li>The first reception data byte is received after completion of header transmission but before a response transmission/reception request is set</li> <li>The first reception data byte is received after the completion of previous data group reception before a transmission/reception request for another data group is set</li> </ul>	LIN operation mode	Cancel	×	RPER flag in RLN3nLEST register

Note 1. If a bit error is detected, the process is canceled after a stop bit is sent. If a bit error is detected in a non-data area, such as an inter-byte space, the transmission is canceled immediately after that area. If a bit error is detected during the transmission of a wake-up, the transmission of the wake-up is canceled after the error-causing bit is sent.

Note 2. The timeout time depends on the response field data length (the RFDL [3:0] bits in the RLN3nLDFC register) and the checksum selection (the CSM bit in the RLN3nLDFC register), and this can be calculated according to the following formula:

[Frame timeout]

On classic selection (when the CSM bit in RLN3nLDFC is 0): Timeout time = 49 + (number of data bytes + 1) × 14 [Tbit]

On enhanced selection (when the CSM bit in RLN3nLDFC is 1): Timeout time = 48 + (number of data bytes + 1) × 14 [Tbit]

The aforementioned timeout time is a time greater than the TFRAME\_MAX of LIN Specification Package

Revision 1.3 on classic selection, or the TFRAME\_MAX of LIN Specification Package Revision 2.0 or 2.1 on enhanced selection.

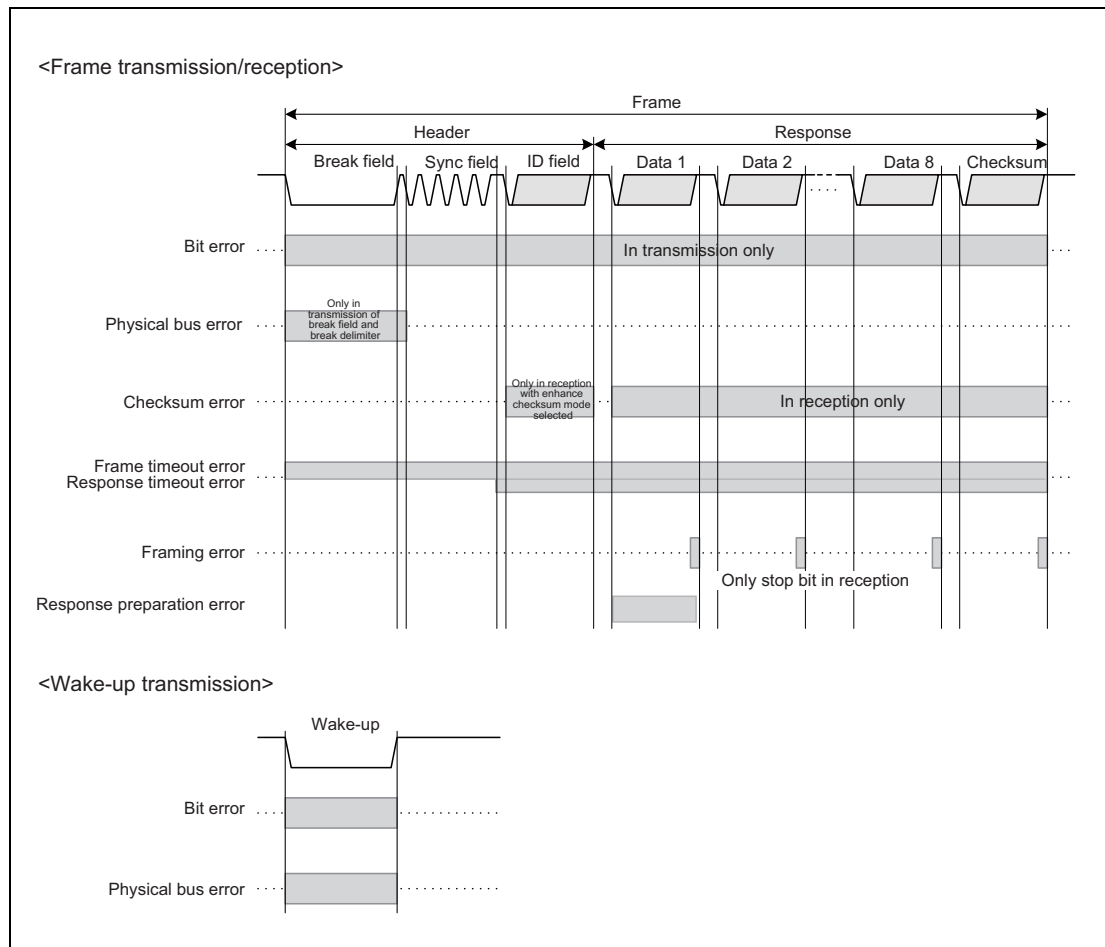
[Response timeout]

$$\text{Timeout time} = (\text{number of data bytes} + 1) \times 14 \text{ [Tbit]}$$

The error status is cleared when the next communication is started, by software, or at a transition to LIN reset mode.

**(2) Target Time Area for LIN Error Detection**

**Figure 25.13** shows the time domain in which the LIN Interface in master mode performs monitoring for error detection.



**Figure 25.13 Target Time Area for LIN Error Detection (LIN Master Mode)**

## 25.8 LIN Self-Test Mode

When the LIN interface enters the LIN self-test mode, RLIN3nTX and RLIN3nRX are disconnected from external pins and RLIN3nTX and RLIN3nRX are connected in the LIN interface. Therefore, the frame transmitted from RLIN3nTX is looped back to RLIN3nRX.

The LIN self-test mode can perform tests exclusively in LIN mode.

The self-test can be performed in the following two types.

- LIN master self-test mode (transmission): Header transmission and response transmission
- LIN master self-test mode (reception): Header transmission and response reception

In LIN self-test mode, the operate is at the fastest baud rate, regardless of the setting of the baud rate generator,

Regardless of the setting of the baud rate related registers, the baud rate operates at the LIN communication clock source/16 [bps]. (The NSPB bits in the RLIN3nLWBR register should be set to 0000<sub>B</sub> or 1111<sub>B</sub>.)

In addition, in LIN self-mode, the following functions are not supported.

- LIN wake-up mode
- Frame separate mode
- Multi-byte response transmission/reception
- Frame/response timeout error



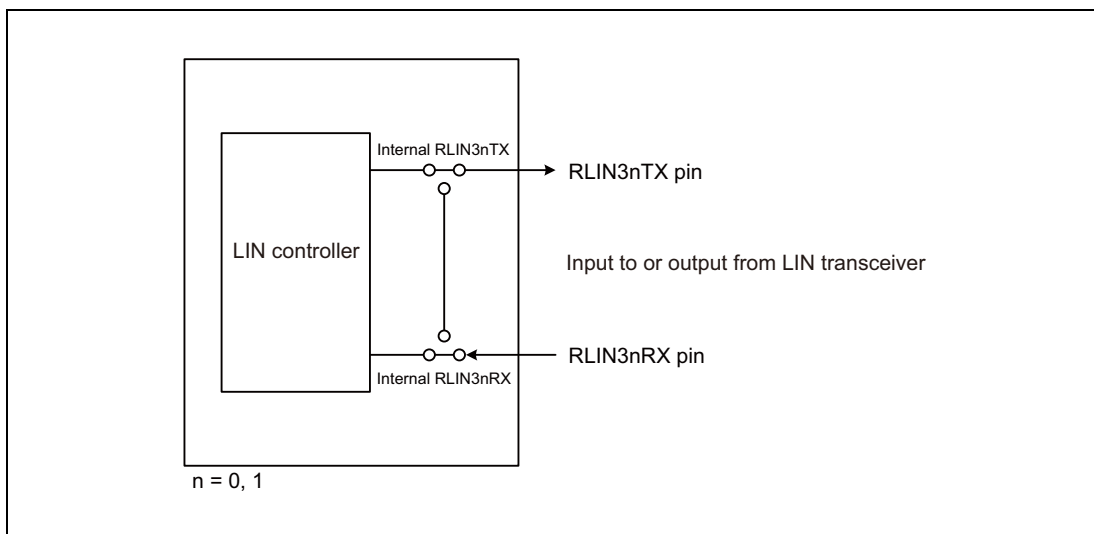


Figure 25.14 Connection in LIN Reset Mode and LIN Mode

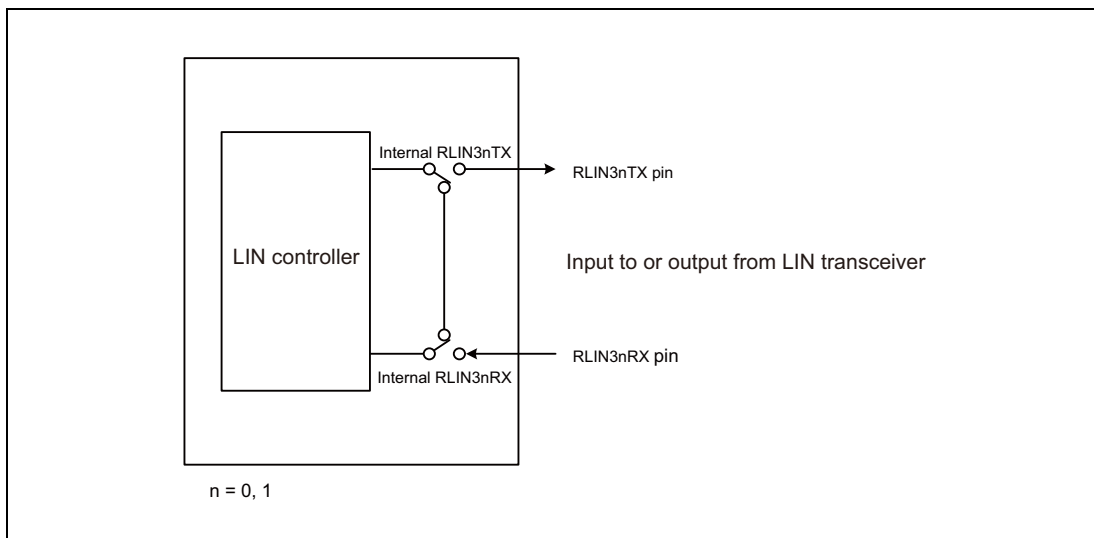


Figure 25.15 Connection in LIN Self-Test Mode

### 25.8.1 Change to LIN Self-Test Mode

Writing to the RLN3nLSTC register makes a transition to the LIN self-test mode.

When the LSTM bit in the RLN3nLSTC register is set to 1, the shift to the LIN self-test mode is checked.

When changing to LIN self-test mode, be sure to execute a specific sequence. In that sequence, information must be written three times consecutively to the LIN self-test control register, as follows:

- Change to LIN reset mode  
Set the OM0 bit in the RLN3nLCUC register to 0 (LIN reset mode).  
Read the OMM0 bit in the RLN3nLMST register; verify that it is 0 (LIN reset mode).
- Select a LIN mode  
LMD bits in RLN3nLMD = 00<sub>B</sub> (LIN master mode)
- 1st write: RLN3nLSTC register = 1010 0111 (A7<sub>H</sub>)
- 2nd write: RLN3nLSTC register = 0101 1000 (58<sub>H</sub>)
- 3rd write: RLN3nLSTC register = 0000 0001 (01<sub>H</sub>)
- Verify the transition to LIN self-test mode  
Read the LSTM bit in the RLN3nLSTC register; verify that it is 1 (LIN self-test mode).

If the key of the first write (A7<sub>H</sub>) is written twice by mistake, the transition to LIN self-test mode is canceled. The above sequence should be retried from the step of first write. In addition, if a write to another LIN-related register is performed during transition to LIN self-test mode (three consecutive write operations to the RLN3nLSTC register), the transition is also canceled.

## 25.8.2 Transmission in LIN Master Self-Test Mode

To execute a self-test on LIN master transmission, perform the procedure below:

- Set the baud rate, noise filter, and interrupt output related registers.  
 RLN3nLWBR register = 0000xxxxB\*<sup>1</sup>  
 RLN3nLBRP0 register = xxxxxxxxB\*<sup>1</sup>  
 RLN3nLBRP1 register = xxxxxxxxB\*<sup>1</sup>  
 RLN3nLMD register = 00xxxx00B\*<sup>1</sup>
- Set the interrupt enable and error enable related registers.  
 RLN3nLIE register = 0000xxxxB  
 RLN3nLEDE register = x000x0xxB
- Set the break field and space related registers.  
 RLN3nLBFC register = 00xxxxxB  
 RLN3nLSC register = 00xx0xxxB
- Cancel the LIN reset mode.  
 Write 11B to the OM1 and OM0 bits in the RLN3nLCUC register, and check that the OMM1 and OMM0 bits in the RLN3nLMST register are 11B.
- Set the transmit frame related registers.  
 RLN3nLDFC register = 00x1xxxxB  
 RLN3nLIDB register = xxxxxxxxB  
 RLN3nLDRB1 to RLN3nLDRB8 registers = xxxxxxxxB
- Header transmission → response transmission started  
 Set the FTS bit in the RLN3nLTRC register to 1 (frame transmission or wake-up transmission/reception started).  
 The LIN master self mode (transmission) is executed. In this mode, interrupt are generated, and status and error status are also updated. The checksum is automatically calculated by the LIN interface.
- When the transmission is completed, the reversed value of the looped-back frame data is stored in the RLN3nLIDB, RLN3nLDBRm (m = 1 to 8), and RLN3nLCBR registers (the data is reversed before being stored because the transmitted value should be compared with the looped-back value). Then, the FTS bit in the RLN3nLTRC register is cleared.
- If the transmission fails to complete due to an error, the applicable error flag is set and the FTS bit in the RLN3nLTRC register is cleared.

**Note:** x: Don't care

**Note 1.** The following register settings are not reflected to the operation of the LIN self-test mode. The LPRS bit in the RLN3nLWBR register, the RLN3nLBRP0 register, the RLN3nLBRP1 register, and the LCKS bit in the RLN3nLMD register. Therefore, those settings are not necessary.

**Note 2.** When the successful header transmission interrupt and the successful frame transmission interrupt are used in the same interrupt processing, if the software processing of the successful header transmission interrupt is not completed before the generation of the successful frame transmission interrupt, the SHIE bit in the RLN3nLIE register should not be set to 1 (successful header transmission interrupt enabled). The time required from the set of the successful header transmission flag to the set of the successful frame/wake-up transmission flag is calculated by the following formula.

$$10 \times (\text{number of data bytes} + 1) [\text{Tbit}]$$

$$1 \text{ Tbit} = \text{LIN communication clock source} \times 16$$

### 25.8.3 Reception in LIN Master Self-Test Mode

To execute a self-test on LIN master reception, perform the procedure below:

- Set the baud rate, noise filter, and interrupt output related registers.  
 RLN3nLWBR register = 0000xxxxB\*<sup>1</sup>  
 RLN3nLBRP0 register = xxxxxxxxB\*<sup>1</sup>  
 RLN3nLBRP1 register = xxxxxxxxB\*<sup>1</sup>  
 RLN3nLMD register = 00xxxx00B\*<sup>1</sup>
- Set the interrupt enable and error enable related registers.  
 RLN3nLIE register = 0000xxxxB  
 RLN3nLEDE register = x000x0xxB
- Set the break field and space related registers.  
 RLN3nLBFC register = 00xxxxxB  
 RLN3nLSC register = 00xx0xxxB\*<sup>1</sup>
- Cancel the LIN reset mode.  
 Write 11B to the OM1 and OM0 bits in the RLN3nLCUC register, and check that the OMM1 and OMM0 bits in the RLN3nLMST register are 11B.
- Set the reception frame related registers.  
 RLN3nLDFC register = 00x0xxxxB\*<sup>3</sup>  
 RLN3nLIDB register = xxxxxxxxB  
 RLN3nLDBR1 to RLN3nLDBR8 registers = xxxxxxxxB  
 RLN3nLCBR register = xxxxxxxxB  
 Since the checksum is not automatically calculated, set the calculation value. If an incorrect checksum is set at this time, the checksum error can be tested.
- Header transmission → response reception started  
 Set the FTS bit in the RLN3nLTRC register to 1 (frame transmission or wake-up transmission/reception started).  
 The LIN master self mode (reception) is executed. In this mode, interrupt are generated, and status and error status are also updated.
- When the reception is completed, the reversed value of the looped-back frame data is stored in the RLN3nLIDB, RLN3nLDBRm (m = 1 to 8), and RLN3nLCBR registers (the data is reversed before being stored because the set value should be compared with the looped-back value). Then, the FTS bit in the RLN3nLTRC register is cleared.
- If the reception fails to complete due to an error, the applicable error flag is set and the FTS bit in the RLN3nLTRC register is cleared.

**Note:** x: Don't care

**Note 1.** The following register settings are not reflected to the operation of the LIN self-test mode. The LPRS bit in the RLN3nLWBR register, the RLN3nLBRP0 register, the RLN3nLBRP1 register, the LCKS bit in the RLN3nLMD register, and the IBS bit in the RLN3nLSC register. Therefore, those settings are not necessary.

**Note 2.** When the successful header transmission interrupt and the successful frame reception interrupt are used in the same interrupt processing, if the software processing of the

successful header transmission interrupt is not completed before the generation of the successful frame reception interrupt, the SHIE bit in the RLN3nLIE register should not be set to 1 (successful header transmission interrupt enabled).

The time required from the set of the successful header transmission flag to the set of the successful frame/wake-up reception flag is calculated by the following formula.

$$10 \times (\text{number of data bytes} + 1) [\text{Tbit}]$$
$$1 \text{ Tbit} = \text{LIN communication clock source} \times 16$$

**Note 3.** When the reception is in self-test mode, be sure to set the response field length (RFDL bits) to at least 1 byte.

#### 25.8.4 Terminating LIN Self-Test Mode

To terminate LIN self-test mode, perform the procedure below:

- Write 0 (LIN reset mode) to the OM0 bit in the RLN3nLCUC register.  
If the OMM1 and OMM0 bits in the RLN3nLMST register are not 11<sub>B</sub>, write 11<sub>B</sub> to the OM1 and OM0 bits in the RLN3nLCUC register. After confirming that the OMM1 and OMM0 bits in the RLN3nLMST register have turned 11<sub>B</sub>, change to LIN reset mode.
- Verify the cancelation of LIN self-test mode.  
Read the LSTM bit in the RLN3nLSTC register; confirm that it is not 0 (not in LIN self-test)
- Verify the transition to LIN reset mode.  
Read the OMM0 bit in the RLN3nLMST register; verify that it is 0 (LIN reset mode).

## 25.9 Baud Rate Generator

The prescaler clock is obtained by frequency-dividing the LIN communication clock source by the prescaler, and the LIN system clock ( $f_{LIN}$ ) is obtained by frequency-dividing the prescaler clock by the baud rate generator. The clock obtained by frequency-dividing the LIN system clock ( $f_{LIN}$ ) by the number of samples is the bit rate. The reciprocal of this bit rate is called the bit time (Tbit).

The LIN Interface has two kinds of baud rate generators. The baud rate generators switch over according to the mode used.

### 25.9.1 LIN Master Mode (x.7.1)

Figure 25.16 shows a block diagram of baud rate generation in LIN master mode.

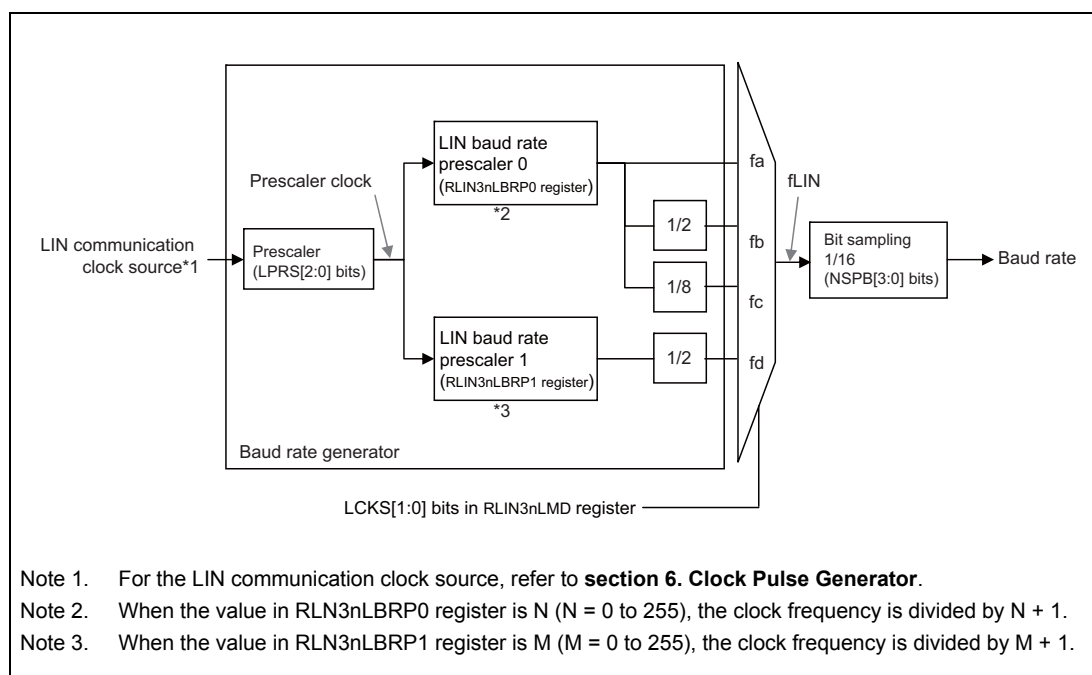


Figure 25.16 Block Diagram of Baud Rate Generation in LIN Master Mode

By setting the RLN3nLBRP0 register so that  $f_a$  is 307200 Hz ( $= 19200 \times 16$ ), the resulting bit rates are  $f_a = 19200 \times 16$ ,  $f_b = 9600 \times 16$  and  $f_c = 2400 \times 16$ . These bit rates are frequency-divided by 16 in the bit timing generator, enabling bit rates of 19200 bps, 9600 bps and 2400 bps to be generated. Also, by setting the RLN3nLBRP1 register so that  $f_d$  is 166672 Hz ( $= 10417 \times 16$ ), the resulting bit rate is  $f_d = 10417 \times 16$ . This bit rate is frequency-divided by 16 in the bit timing generator, enabling 10417 bps to be generated.

Table 25.36 shows examples of baud rate (19200, 10417, 9600, and 2400 bps) generation for each LIN communication clock source frequency, and also the corresponding errors.

**Table 25.36 Examples of Baud Rate (19200, 10417, 9600, and 2400 bps) Generation in LIN Master Mode**

Clock frequency	Prescaler	Baud rate generator 0 (N + 1) division	Baud rate generator 1 (M + 1) division	System clock	Baud rate	Error
33 MHz	1/1	107	—	fa	19.2308 kbps	0.16%
		—	99	fd	10.4167 kbps	0.00%
		107	—	fb	9.6154 kbps	0.16%
		107	—	fc	2.4038 kbps	0.16%
25 MHz	1/1	81	—	fa	19.2308 kbps	0.16%
		—	75	fd	10.4167 kbps	0.00%
		81	—	fb	9.6154 kbps	0.16%
		81	—	fc	2.4038 kbps	0.16%

## 25.10 Noise Filter

The LIN Interface has a noise filter for reducing erroneous receiving of data due to noise. By setting the LRDNFS bit in the RLIN3nLMD register to 0 (to use the noise filter), the noise filter is activated. The synchronized RLIN3nRX is sampled by the prescaler clock. The sampled synchronous RLIN3nRX determines the value of each bit of the received data.

**Figure 25.17** shows the configuration of the noise filter, **Figure 25.18** an example of a noise filter circuit, and **Figure 25.19** the determination of the received data when the noise filter is used.

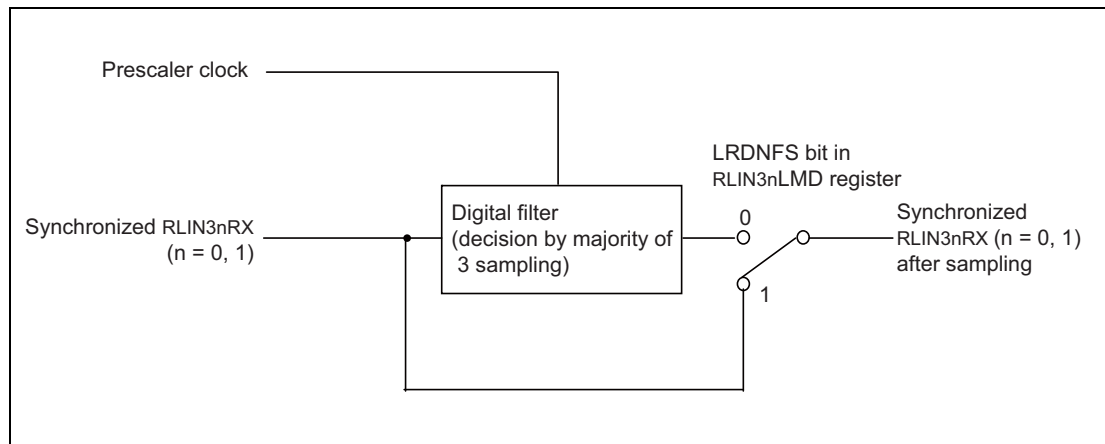


Figure 25.17 Configuration of Noise Filter

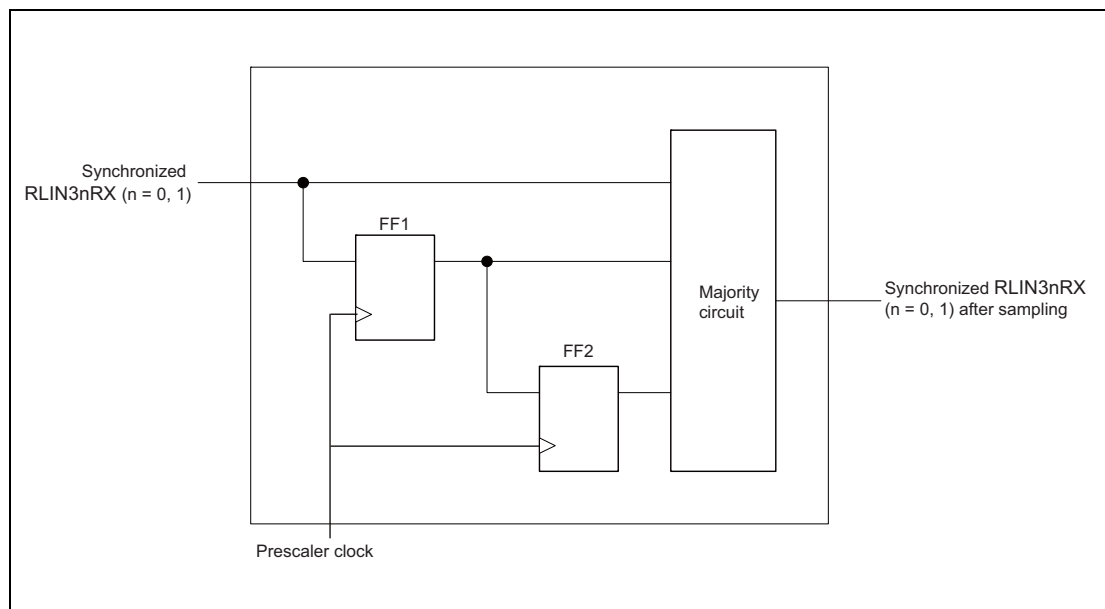


Figure 25.18 Example of Noise Filter Circuit



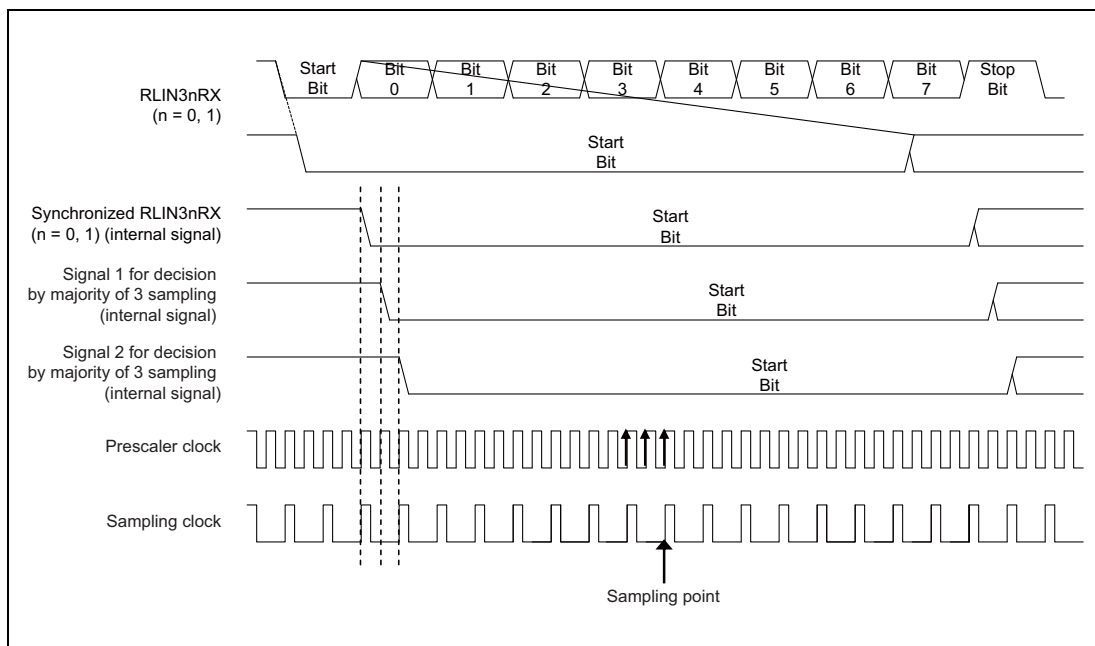


Figure 25.19 Determination of Received Data when Noise Filter is Used

## 26. Ethernet Controller

This LSI has an on-chip Ethernet controller (ETHER) conforming to the Ethernet or the IEEE802.3 MAC (Media Access Control) layer standard. Connecting a physical-layer LSI (PHY-LSI) complying with this standard enables the ETHER to perform transmission and reception of Ethernet/IEEE802.3 frames. The Ethernet controller in this LSI has one MAC layer interface port, which can be made to perform transmission and reception independently.

The ETHER can transfer the transmitted or received Ethernet frame data to and from the transmit/receive buffer in the memory at high speed using a dedicated direct memory access controller (E-DMAC).

### 26.1 Features

- MAC (Media Access Control) function
  - Constructs/deconstructs data frames (frame format conforming to IEEE802.3, 2000 Edition)
  - Supports transfer at 10 and 100 Mbps
  - Supports full-duplex mode
  - One channel (ETHER0)
  - Flow control conforming to IEEE802.3x
  - Supports one PHY interface conforming to IEEE802.3
    - MII (Media Independent Interface)
  - Upward protocol support (checksum) function
- E-DMAC (Direct Memory Access Controller for Ethernet controller) function
  - Data transfer between ETHER and external/internal memory
  - One channel
  - 32-byte burst transfer
  - Supports single-frame/single-descriptor operation and single-frame/multi-descriptor (multi-buffer) operation
  - Transfer data width: 32 bits
  - Transmit/receive FIFO (for transmission: 2 Kbytes, for reception: 4 Kbytes)
  - Function for calculating the intelligent checksum value

Figure 26.1 shows the configuration of the ETHER.

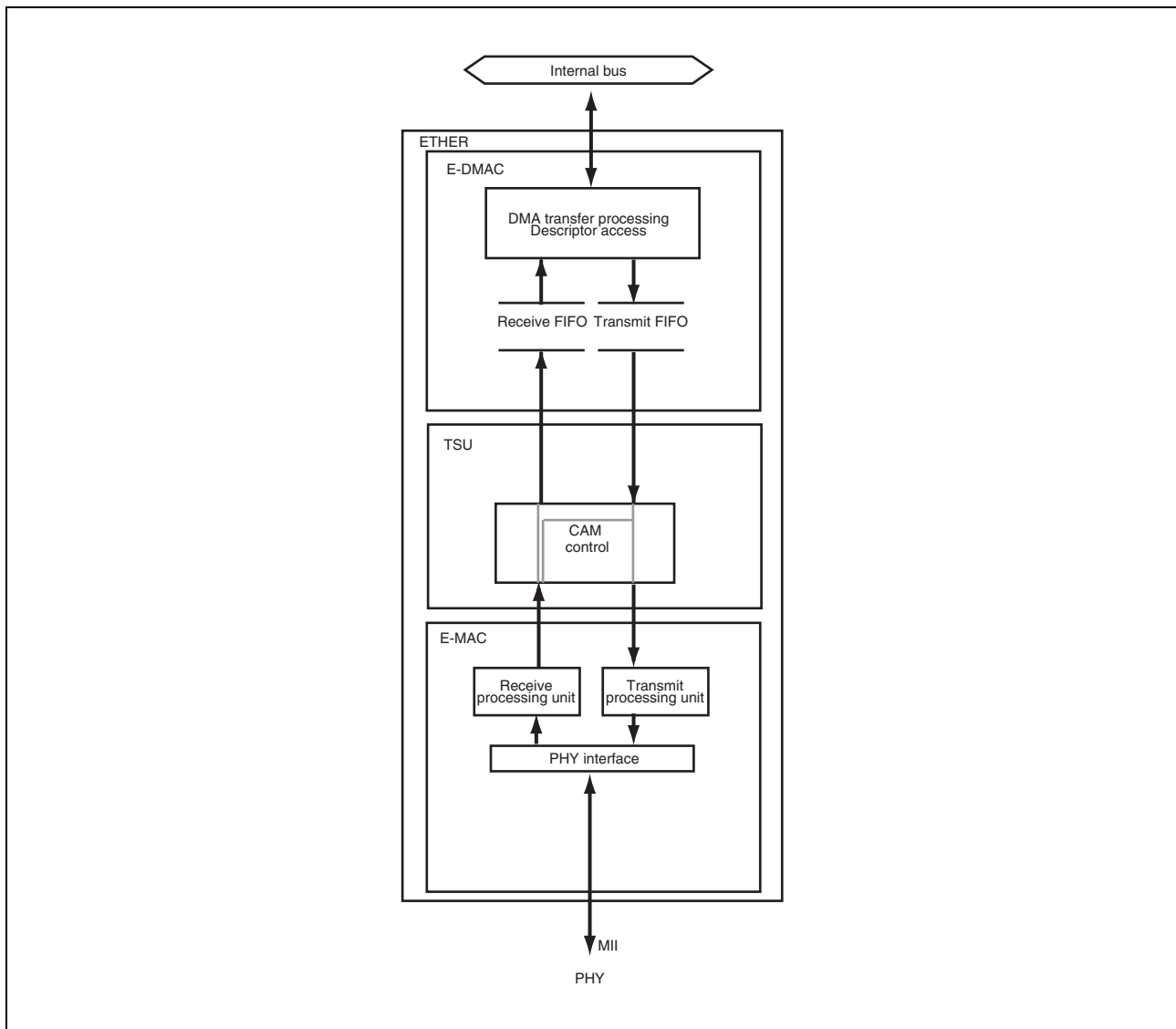


Figure 26.1 Configuration of ETHER

## 26.2 Input/Output Pins

Table 26.1 lists the pin configuration of the ETHER.

**Table 26.1 Pin Configuration**

Name	Abbreviation	I/O	Function
Transmit clock	ET_TXCLK*1	Input	ET_TXEN, ET_TXD[3:0] timing reference signal
Transmit enable	ET_TXEN*1	Output	Indicates that transmit data is ready on ET_TXD[3:0]
MII transmit data	ET_TXD[3:0]*1	Output	MII transmit data
Collision detection	ET_COL*1	Input	Collision detection signal
Transmit error	ET_TXER*1	Output	Not asserted in the ETHER
Receive clock	ET_RXCLK*1	Input	ET_RXDV, ET_RXD[3:0], ET_RXER timing reference signal
Receive data valid	ET_RXDV*1	Input	Indicates that valid receive data is on ET_RXD[3:0]
MII receive data	ET_RXD[3:0]*1	Input	MII receive data
Receive error	ET_RXER*1	Input	Identifies error state occurred during data reception
Carrier detection	ET_CRD*1	Input	Carrier detection signal
Management data clock	ET_MDC*1	Output	Reference clock signal for information transfer via ET_MDIO
Management data I/O	ET_MDIO*1	I/O	Bidirectional signal for exchange of management information between STA and PHY

Note 1. MII signal conforming to IEEE802.3u

## 26.3 Register Descriptions

Table 26.2 shows the configuration of registers of the ETHER.

**Table 26.2 Register Configuration**

Name	Abbreviation	R/W	Address	Access Size
Software reset register	ARSTR	R/W	H'E820 4800	32
E-MAC mode register	ECMR0	R/W	H'E820 3500	32
E-MAC status register	ECSR0	R/W	H'E820 3510	32
E-MAC interrupt permission register	ECSIPR0	R/W	H'E820 3518	32
PHY interface register	PIR0	R/W	H'E820 3520	32
MAC address high register	MAHR0	R/W	H'E820 35C0	32
MAC address low register	MALR0	R/W	H'E820 35C8	32
Receive frame length register	RFLR0	R/W	H'E820 3508	32
CRC error frame receive counter register	CEFCR0	R/W	H'E820 3740	32
Frame receive error counter register	FRECR0	R/W	H'E820 3748	32
Too-short frame receive counter register	TSFRCR0	R/W	H'E820 3750	32
Too-long frame receive counter register	TLFRCR0	R/W	H'E820 3758	32
Residual-bit frame receive counter register	RFCR0	R/W	H'E820 3760	32
Multicast address frame receive counter register	MAFCR0	R/W	H'E820 3778	32
Automatic PAUSE frame register	APR0	R/W	H'E820 3554	32
Manual PAUSE frame register	MPR0	R/W	H'E820 3558	32
Automatic PAUSE frame retransmit count register	TPAUSER0	R/W	H'E820 3564	32
PAUSE frame transmit counter register	PFTCR0	R	H'E820 355C	32
PAUSE frame receive counter register	PFRCR0	R	H'E820 3560	32
TSU counter reset register	TSU_CTRST	R/W	H'E820 4804	32
VLANtag set register	TSU_VTAG0	R/W	H'E820 4858	32
CAM entry table busy register	TSU_ADSBSY	R	H'E820 4860	32
CAM entry table enable register	TSU_TEN	R/W	H'E820 4864	32
CAM entry table 0H register	TSU_ADRH0	R/W	H'E820 4900	32
CAM entry table 1H register	TSU_ADRH1	R/W	H'E820 4908	32
CAM entry table 2H register	TSU_ADRH2	R/W	H'E820 4910	32
CAM entry table 3H register	TSU_ADRH3	R/W	H'E820 4918	32
CAM entry table 4H register	TSU_ADRH4	R/W	H'E820 4920	32
CAM entry table 5H register	TSU_ADRH5	R/W	H'E820 4928	32
CAM entry table 6H register	TSU_ADRH6	R/W	H'E820 4930	32
CAM entry table 7H register	TSU_ADRH7	R/W	H'E820 4938	32
CAM entry table 8H register	TSU_ADRH8	R/W	H'E820 4940	32
CAM entry table 9H register	TSU_ADRH9	R/W	H'E820 4948	32
CAM entry table 10H register	TSU_ADRH10	R/W	H'E820 4950	32
CAM entry table 11H register	TSU_ADRH11	R/W	H'E820 4958	32
CAM entry table 12H register	TSU_ADRH12	R/W	H'E820 4960	32
CAM entry table 13H register	TSU_ADRH13	R/W	H'E820 4968	32
CAM entry table 14H register	TSU_ADRH14	R/W	H'E820 4970	32
CAM entry table 15H register	TSU_ADRH15	R/W	H'E820 4978	32
CAM entry table 16H register	TSU_ADRH16	R/W	H'E820 4980	32
CAM entry table 17H register	TSU_ADRH17	R/W	H'E820 4988	32

**Table 26.2 Register Configuration**

Name	Abbreviation	R/W	Address	Access Size
CAM entry table 18H register	TSU_ADRH18	R/W	H'E820 4990	32
CAM entry table 19H register	TSU_ADRH19	R/W	H'E820 4998	32
CAM entry table 20H register	TSU_ADRH20	R/W	H'E820 49A0	32
CAM entry table 21H register	TSU_ADRH21	R/W	H'E820 49A8	32
CAM entry table 22H register	TSU_ADRH22	R/W	H'E820 49B0	32
CAM entry table 23H register	TSU_ADRH23	R/W	H'E820 49B8	32
CAM entry table 24H register	TSU_ADRH24	R/W	H'E820 49C0	32
CAM entry table 25H register	TSU_ADRH25	R/W	H'E820 49C8	32
CAM entry table 26H register	TSU_ADRH26	R/W	H'E820 49D0	32
CAM entry table 27H register	TSU_ADRH27	R/W	H'E820 49D8	32
CAM entry table 28H register	TSU_ADRH28	R/W	H'E820 49E0	32
CAM entry table 29H register	TSU_ADRH29	R/W	H'E820 49E8	32
CAM entry table 30H register	TSU_ADRH30	R/W	H'E820 49F0	32
CAM entry table 31H register	TSU_ADRH31	R/W	H'E820 49F8	32
CAM entry table 0L register	TSU_ADRL0	R/W	H'E820 4904	32
CAM entry table 1L register	TSU_ADRL1	R/W	H'E820 490C	32
CAM entry table 2L register	TSU_ADRL2	R/W	H'E820 4914	32
CAM entry table 3L register	TSU_ADRL3	R/W	H'E820 491C	32
CAM entry table 4L register	TSU_ADRL4	R/W	H'E820 4924	32
CAM entry table 5L register	TSU_ADRL5	R/W	H'E820 492C	32
CAM entry table 6L register	TSU_ADRL6	R/W	H'E820 4934	32
CAM entry table 7L register	TSU_ADRL7	R/W	H'E820 493C	32
CAM entry table 8L register	TSU_ADRL8	R/W	H'E820 4944	32
CAM entry table 9L register	TSU_ADRL9	R/W	H'E820 494C	32
CAM entry table 10L register	TSU_ADRL10	R/W	H'E820 4954	32
CAM entry table 11L register	TSU_ADRL11	R/W	H'E820 495C	32
CAM entry table 12L register	TSU_ADRL12	R/W	H'E820 4964	32
CAM entry table 13L register	TSU_ADRL13	R/W	H'E820 496C	32
CAM entry table 14L register	TSU_ADRL14	R/W	H'E820 4974	32
CAM entry table 15L register	TSU_ADRL15	R/W	H'E820 497C	32
CAM entry table 16L register	TSU_ADRL16	R/W	H'E820 4984	32
CAM entry table 17L register	TSU_ADRL17	R/W	H'E820 498C	32
CAM entry table 18L register	TSU_ADRL18	R/W	H'E820 4994	32
CAM entry table 19L register	TSU_ADRL19	R/W	H'E820 499C	32
CAM entry table 20L register	TSU_ADRL20	R/W	H'E820 49A4	32
CAM entry table 21L register	TSU_ADRL21	R/W	H'E820 49AC	32
CAM entry table 22L register	TSU_ADRL22	R/W	H'E820 49B4	32
CAM entry table 23L register	TSU_ADRL23	R/W	H'E820 49BC	32
CAM entry table 24L register	TSU_ADRL24	R/W	H'E820 49C4	32
CAM entry table 25L register	TSU_ADRL25	R/W	H'E820 49CC	32
CAM entry table 26L register	TSU_ADRL26	R/W	H'E820 49D4	32
CAM entry table 27L register	TSU_ADRL27	R/W	H'E820 49DC	32
CAM entry table 28L register	TSU_ADRL28	R/W	H'E820 49E4	32
CAM entry table 29L register	TSU_ADRL29	R/W	H'E820 49EC	32
CAM entry table 30L register	TSU_ADRL30	R/W	H'E820 49F4	32

**Table 26.2 Register Configuration**

Name	Abbreviation	R/W	Address	Access Size
CAM entry table 31L register	TSU_ADRL31	R/W	H'E820 49FC	32
Transmit frame counter register (normal transmission only)	TXNLCR0	R	H'E820 4880	32
Transmit frame counter register (normal and erroneous transmission)	TXALCR0	R	H'E820 4884	32
Receive frame counter register (normal reception only)	RXNLCR0	R	H'E820 4888	32
Receive frame counter register (normal and erroneous reception)	RXALCR0	R	H'E820 488C	32
E-DMAC start register	EDSR0	W	H'E820 3000	32
E-DMAC mode register	EDMR0	R/W	H'E820 3400	32
E-DMAC transmit request register	EDTRR0	R/W	H'E820 3408	32
E-DMAC receive request register	EDRRR0	R/W	H'E820 3410	32
E-MAC/E-DMAC status register	EESR0	R/W	H'E820 3428	32
E-MAC/E-DMAC status interrupt permission register	EESIPR0	R/W	H'E820 3430	32
Transmit descriptor list start address register	TDLAR0	R/W	H'E820 3010	32
Transmit descriptor fetch address register	TDFAR0	R/W	H'E820 3014	32
Transmit descriptor finished address register	TDFXR0	R/W	H'E820 3018	32
Transmit descriptor final flag register	TDFFR0	R/W	H'E820 301C	32
Receive descriptor list start address register	RDLAR0	R/W	H'E820 3030	32
Receive descriptor fetch address register	RDFAR0	R/W	H'E820 3034	32
Receive descriptor finished address register	RDFXR0	R/W	H'E820 3038	32
Receive descriptor final flag register	RDFFR0	R/W	H'E820 303C	32
Transmit/receive status copy enable register	TRSCER0	R/W	H'E820 3438	32
Receive missed-frame counter register	RMFCR0	R/W	H'E820 3440	32
Transmit FIFO threshold register	TFTR0	R/W	H'E820 3448	32
FIFO depth register	FDR0	R/W	H'E820 3450	32
Receiving method control register	RMCR0	R/W	H'E820 3458	32
Receive data padding insert register	RPADIR0	R/W	H'E820 3460	32
Overflow alert FIFO threshold register	FCFTR0	R/W	H'E820 3468	32
Intelligent checksum mode register	CSMR	R/W	H'E820 34E4	32
Intelligent checksum skipped bytes monitor register	CSSBM	R	H'E820 34E8	32
Intelligent checksum monitor register	CSSMR	R	H'E820 34EC	32

### 26.3.1 Software Reset Register (ARSTR)

ARSTR resets all blocks (E-MAC, TSU, and E-DMAC) in the ETHER. By writing 1 to the ARST bit in this register, a software reset is issued to all blocks of the ETHER (for 256 cycles of internal bus clock B $\phi$ ). The ARST bit is always read as 0. While a software reset is issued, register access to all blocks of the ETHER is prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ARST
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	ARST	0	R/W	Software Reset When 1 is written to this bit, a software reset is issued to all blocks of the ETHER (for 256 cycles of internal bus clock B $\phi$ ). Writing 0 does not affect this bit. This bit is always read as 0. While a software reset is issued, register access to all blocks of the ETHER is prohibited. The following registers are not initialized by a software reset. TSU_ADRH0 to TSU_ADRH31, TSU_ADRL0 to TSU_ADRL31, TXNLCR0, TXALCR0, RXNLCR0, RXALCR0



### 26.3.2 E-MAC Mode Register (ECMR)

ECMR is a 32-bit readable/writable register that specifies the operating mode of the ETHER. The settings in this register are normally made in the initialization process following a reset.

The operating mode setting must not be changed while the transmitting and receiving functions are enabled. To switch the operating mode, return the E-MAC and E-DMAC to their initial states by means of the SWRT and SWRR bits in EDMR before making settings again.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	TRCCM	—	—	RCSC	—	DPAD	RZPF	ZPF	PFR	RXF	TXF
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R	R	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	MCT	—	—	—	—	—	—	RE	TE	—	—	—	DM	PRM
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26	TRCCM	0	R/W	Counter Clear Mode Sets the method for clearing the counter register. Refer to the description of each register. 0: Cleared to 0 when the relevant register is written 1: Cleared to 0 when the relevant register is read
25, 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23	RCSC	0	R/W	Checksum Calculation Specifies whether to perform automatic calculation (hardware calculation) of the checksum of the receive frame data unit. 0: Checksum is not automatically calculated 1: Checksum is automatically calculated Note that the checksum calculation of a frame with a VLAN tag is not supported. For details, see section 26.6.1, Checksum Calculation of Ethernet Frames.
22	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
21	DPAD	0	R/W	Data Padding 0: Padding is inserted to data less than 60 bytes so it is transmitted as 60-byte data 1: Padding is not inserted to data less than 60 bytes and it is transmitted without changes
20	RZPF	0	R/W	PAUSE Frame Reception with TIME = 0 0: Reception of a PAUSE frame whose TIME parameter value is 0 is disabled 1: Reception of a PAUSE frame whose TIME parameter value is 0 is enabled
19	ZPF	0	R/W	PAUSE Frame Usage with TIME = 0 Enable 0: Control of a PAUSE frame whose TIME parameter value is 0 is disabled. The next frame is not transmitted until the time specified by the Timer value has elapsed. If a PAUSE frame whose time specified by the Timer value is 0 is received, that PAUSE frame is discarded. 1: Control of a PAUSE frame whose TIME parameter value is 0 is enabled. When the data size in the receive FIFO becomes smaller than the FCFT setting before the time specified by the Timer value elapses, an automatic PAUSE frame with a Timer value of 0 is transmitted. On receiving a PAUSE frame with a Timer value of 0, the transmission wait state is canceled.

Bit	Bit Name	Initial Value	R/W	Description
18	PFR	0	R/W	PAUSE Frame Receive Mode 0: PAUSE frame is not transferred to E-DMAC 1: PAUSE frame is transferred to E-DMAC
17	RXF	0	R/W	Operating Mode for Receiving Port Flow Control 0: PAUSE frame detection is disabled 1: Flow control for the receiving port is enabled
16	TXF	0	R/W	Operating Mode for Transmitting Port Flow Control 0: Flow control for the transmitting port is disabled (Automatic PAUSE frame is not transmitted) 1: Flow control for the transmitting port is enabled (Automatic PAUSE frame is transmitted as required)
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13	MCT	0	R/W	Multicast Address Frame Receive Mode 0: Frames other than the multicast address set by the CAM entry table 0 to 31 (H/L) registers are received. However, if the on-chip CAM entry table reference is disabled, all multicast address frames are received. 1: Only the multicast address set by the CAM entry table 0 to 31 (H/L) registers is received.
12 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6	RE	0	R/W	Reception Enable If a switch is made from receiving function enabled (RE = 1) to disabled (RE = 0) while a frame is being received, the receiving function will be enabled until reception of the corresponding frame is completed. 0: Receiving function is disabled 1: Receiving function is enabled
5	TE	0	R/W	Transmission Enable If a switch is made from transmitting function enabled (TE = 1) to disabled (TE = 0) while a frame is being transmitted, the transmitting function will be enabled until transmission of the corresponding frame is completed. 0: Transmitting function is disabled 1: Transmitting function is enabled
4 to 2	—	0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	DM	0	R/W	Full-duplex Transfer Enable 0: Full-duplex transfer function is disabled. 1: Full-duplex transfer function is enabled.
0	PRM	0	R/W	Promiscuous Mode Setting this bit enables all Ethernet frames to be received. All Ethernet frames means all receivable frames, irrespective of differences or enabled/disabled status (destination address, broadcast address, multicast bit, etc.). 0: ETHER performs normal operation 1: ETHER performs promiscuous mode operation

Note: • All bits, except for TE and RE, should be changed while the transmitting function is disabled (TE = 0) and the receiving function is disabled (RE = 0).

### 26.3.3 E-MAC Status Register (ECSR)

ECSR is a 32-bit readable/writable register that indicates the status in the E-MAC. This status can be notified to the CPU by interrupts. When 1 is written to the PFROI and ICD bits, the corresponding flags can be cleared. Writing 0 does not affect the flag. For bits that generate interrupts, the interrupt can be enabled or disabled by the corresponding bit in ECSIPR.

The interrupts generated due to this status register are indicated in the ECI bit in EESR of the E-DMAC.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	PFROI	—	—	—	ICD
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	PFROI	0	R/W	PAUSE Frame Retransmit Retry Over Indicates whether the retransmit count for retransmitting a PAUSE frame when flow control is enabled has exceeded the retransmit upper-limit set in the automatic PAUSE frame retransmit count register (TPAUSER). 0: PAUSE frame retransmit count has not exceeded the upper limit 1: PAUSE frame retransmit count has exceeded the upper limit
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	ICD	0	R/W	Illegal Carrier Detection Indicates that the PHY-LSI has detected an illegal carrier on the line. If a change in the signal input from the PHY-LSI occurs in a period shorter than the software recognition period, the correct information may not be obtained. Refer to the timing specification for the PHY-LSI used. 0: PHY-LSI has not detected an illegal carrier on the line 1: PHY-LSI has detected an illegal carrier on the line

### 26.3.4 E-MAC Interrupt Permission Register (ECSIPR)

ECSIPR is a 32-bit readable/writable register that enables or disables the interrupt sources indicated by ECSR. Each bit can disable or enable interrupts corresponding to the bits in ECSR.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	PFR IP	PHY IP	LCHN GIP	MPDIP	ICDIP
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	PFR IP	0	R/W	PAUSE Frame Retransmit Interrupt Enable 0: Interrupt notification by the PFR IP bit is disabled 1: Interrupt notification by the PFR IP bit is enabled
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	ICDIP	0	R/W	Illegal Carrier Detect Interrupt Enable 0: Interrupt notification by the ICD bit is disabled 1: Interrupt notification by the ICD bit is enabled

### 26.3.5 PHY Interface Register (PIR)

PIR is a 32-bit readable/writable register that provides a means of accessing the PHY-LSI internal registers via the MII.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	MDI	MDO	MMD	MDC
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	—	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	MDI	Undefined	R	MII Management Data-In Indicates the level of the ET_MDIO pin.
2	MDO	0	R/W	MII Management Data-Out Outputs the value set in this bit from the ET_MDIO pin when the MMD bit is 1.
1	MMD	0	R/W	MII Management Mode Specifies the data read/write direction with respect to the MII. 0: Read direction is specified 1: Write direction is specified
0	MDC	0	R/W	MII Management Data Clock Outputs the value set in this bit from the MDC pin and supplies the MII with the management data clock. For the method of accessing the MII registers, see section 26.5.2, Accessing MII Registers.

### 26.3.6 MAC Address High Register (MAHR)

MAHR is a 32-bit readable/writable register that specifies the upper 32 bits of the 48-bit MAC address. The settings in this register are normally made in the initialization process after a reset. The MAC address setting must not be changed while the transmitting and receiving functions are enabled. Return the E-MAC and E-DMAC to their initial states by means of the SWRT and SWRR bits in EDMR before making settings again.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MA[47:32]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MA[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	MA[47:16]	All 0	R/W	MAC Address Bits 47 to 16 These bits are used to set the upper 32 bits of the MAC address. If the MAC address is 01-23-45-67-89-AB (hexadecimal), set H'01234567 in this register.

### 26.3.7 MAC Address Low Register (MALR)

MALR is a 32-bit readable/writable register that specifies the lower 16 bits of the 48-bit MAC address. The settings in this register are normally made in the initialization process after a reset. The MAC address setting must not be changed while the transmitting and receiving functions are enabled. Return the E-MAC and E-DMAC to their initial states by means of the SWRT and SWRR bits in EDMR before making settings again.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MA[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	MA[15:0]	All 0	R/W	MAC Address Bits 15 to 0 These bits are used to set the lower 16 bits of the MAC address. If the MAC address is 01-23-45-67-89-AB (hexadecimal), set H'000089AB in this register.

### 26.3.8 Receive Frame Length Register (RFLR)

RFLR is a 32-bit readable/writable register that specifies the maximum frame length (in bytes) that can be received by this LSI. The settings in this register must not be changed while the receiving function is enabled.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RFL[17:16]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

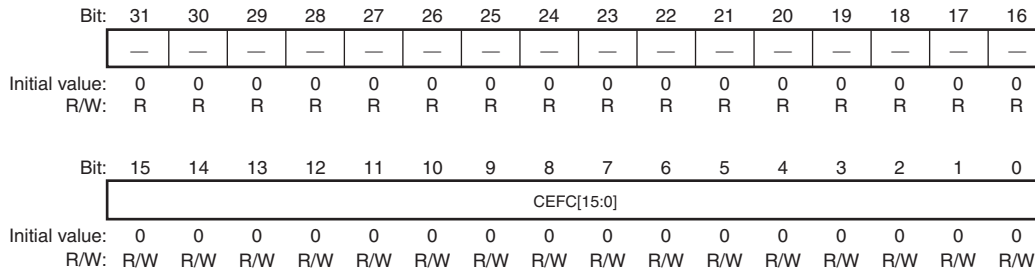
  

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFL[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17 to 0	RFL[17:0]	All 0	R/W	Receive Frame Length The frame data described here refers to all fields from the destination address up to the CRC data. Frame contents from the destination address up to the data are actually transferred to memory. CRC data is not included in the transfer. When data that exceeds the specified value is received, the part of data that exceeds the specified value is discarded. H'00000 to H'005EE: 1,518 bytes H'005EF: 1,519 bytes H'005F0: 1,520 bytes : : H'007FF: 2,047 bytes H'00800: 2,048 bytes : : H'01000: 4,096 bytes : : H'10000: 65,536 bytes : : H'20000 to H'3FFFF: 131,072 bytes

### 26.3.9 CRC Error Frame Receive Counter Register (CEFECR)

CEFECR is a 16-bit counter that indicates the number of times a frame with a CRC error was received. When the value in this register reaches H'0000FFFF, count-up is halted. This register is cleared to 0 when it is read with the TRCCM bit in ECCR set to 1. When the TRCCM bit in ECCR is 0, writing to this register will clear it regardless of the value written.

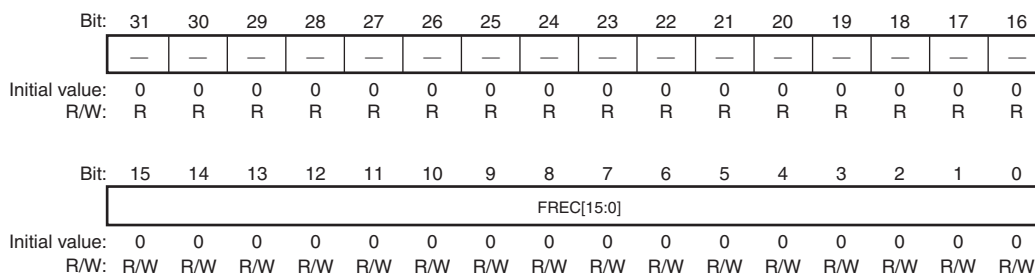


Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	CEFECR[15:0]	All 0	R/W	CRC Error Frame Count These bits indicate the number of CRC error frames received.

Note: • When count-up and count-clearing of this register value occur simultaneously, count-clearing is performed.

### 26.3.10 Frame Receive Error Counter Register (FRECCR)

FRECCR is a 16-bit counter that indicates the number of frames for which a receive error was generated by the RXER pin input from the PHY-LSI. FRECCR is incremented each time the RXER pin becomes active. When the value in this register reaches H'0000FFFF, count-up is halted. This register is cleared to 0 when it is read with the TRCCM bit in ECCR set to 1. When the TRCCM bit in ECCR is 0, writing to this register will clear it regardless of the value written.



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	FRECCR[15:0]	All 0	R/W	Frame Receive Error Count These bits indicate the number of errors during frame reception.

Note: • When count-up and count-clearing of this register value occur simultaneously, count-clearing is performed.



### 26.3.11 Too-Short Frame Receive Counter Register (TSFRCR)

TSFRCR is a 16-bit counter that indicates the number of frames received with a length fewer than 64 bytes. When the value in this register reaches H'0000FFFF, count-up is halted. This register is cleared to 0 when it is read with the TRCCM bit in ECMR set to 1. When the TRCCM bit in ECMR is 0, writing to this register will clear it regardless of the value written.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSFC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	TSFC[15:0]	All 0	R/W	Too-Short Frame Receive Count These bits indicate the number of frames received with a length of less than 64 bytes.

Note: • When count-up and count-clearing of this register value occur simultaneously, count-clearing is performed.

### 26.3.12 Too-Long Frame Receive Counter Register (TLFRCR)

TLFRCR is a 16-bit counter that indicates the number of frames received with a length exceeding the value specified by the receive frame length register (RFLR). When the value in this register reaches H'0000FFFF, count-up is halted. This register is not incremented when a frame containing residual bits is received. In this case, the reception of the frame is indicated in the residual-bit frame receive counter register (RFRCR). This register is cleared to 0 when it is read with the TRCCM bit in ECMR set to 1. When the TRCCM bit in ECMR is 0, writing to this register will clear it regardless of the value written.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TLFC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	TLFC[15:0]	All 0	R/W	Too-Long Frame Receive Count These bits indicate the number of frames received with a length exceeding the value in RFLR.

Note: • When count-up and count-clearing of this register value occur simultaneously, count-clearing is performed.

### 26.3.13 Residual-Bit Frame Receive Counter Register (RFCR)

RFCR is a 16-bit counter that indicates the number of frames received containing residual bits (less than an 8-bit unit). When the value in this register reaches H'0000FFFF, count-up is halted. This register is cleared to 0 when it is read with the TRCCM bit in ECMR set to 1. When the TRCCM bit in ECMR is 0, writing to this register will clear it regardless of the value written.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	RFC[15:0]	All 0	R/W	Residual-Bit Frame Receive Count These bits indicate the number of frames received containing residual bits.

Note: • When count-up and count-clearing of this register value occur simultaneously, count-clearing is performed.

### 26.3.14 Multicast Address Frame Receive Counter Register (MAFCR)

MAFCR is a 16-bit counter that indicates the number of frames received with a specified multicast address. When the value in this register reaches H'0000FFFF, count-up is halted. This register is cleared to 0 when it is read with the TRCCM bit in ECMR set to 1. When the TRCCM bit in ECMR is 0, writing to this register will clear it regardless of the value written.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MAFC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	MAFC[15:0]	All 0	R/W	Multicast Address Frame Count These bits indicate the number of multicast frames received.

Note: • When count-up and count-clearing of this register value occur simultaneously, count-clearing is performed.

### 26.3.15 Automatic PAUSE Frame Register (APR)

APR is used to set the TIME parameter value of an automatic PAUSE frame. When an automatic PAUSE frame is transmitted, the value set in this register is used as the TIME parameter of the PAUSE frame.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AP[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	AP[15:0]	All 0	R/W	Automatic PAUSE These bits set the TIME parameter value of an automatic PAUSE frame. One bit is equivalent to 512 bit-time. When flow control is enabled in transmission (PAUSE frame transmission) (TXF bit in ECMR = 1), set a value other than H'0000 in these bits. H'0000: — H'0001: 512 × 1 bit-time H'0002: 512 × 2 bit-time : : H'FFFF: 512 × 65,535 bit-time Note: The bit-time becomes as follows according to the transfer speed. 100 Mbps: 1 bit-time = 10 ns 10 Mbps: 1 bit-time = 100 ns

### 26.3.16 Manual PAUSE Frame Register (MPR)

MPR is used to set the TIME parameter value of a manual PAUSE frame. When a manual PAUSE frame is transmitted, the value set in this register is used as the TIME parameter of the PAUSE frame.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MP[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	MP[15:0]	All 0	R/W	Manual PAUSE These bits set the TIME parameter value of a manual PAUSE frame. One bit is equivalent to 512 bit-time. H'0000: — H'0001: 512 × 1 bit-time H'0002: 512 × 2 bit-time : : H'FFFF: 512 × 65,535 bit-time Note: The bit-time becomes as follows according to the transfer speed. 100 Mbps: 1 bit-time = 10 ns 10 Mbps: 1 bit-time = 100 ns

### 26.3.17 Automatic PAUSE Frame Retransmit Count Register (TPAUSER)

TPAUSER is used to set the upper limit for the number of times to retransmit an automatic PAUSE frame. The settings in this register must not be changed while the transmitting function is enabled.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TPAUSE[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	TPAUSE[15:0]	All 0	R/W	Upper Limit for Automatic PAUSE Frame Retransmission H'0000: Retransmit count is unlimited H'0001: Retransmit count is 1 : : H'FFFF: Retransmit count is 65,535

### 26.3.18 PAUSE Frame Transmit Counter Register (PFTCR)

PFTCR is a 16-bit counter that indicates the number of times a PAUSE frame is transmitted. This register is cleared to 0 when it is read.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PFTXC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	PFTXC[15:0]	All 0	R	PAUSE Frame Transmit Count These bits indicate the total number of automatic PAUSE frames and manual PAUSE frames transmitted.

Note: • When count-up and count-clearing of this register value occur simultaneously, count-clearing is performed.

### 26.3.19 PAUSE Frame Receive Counter Register (PFRCR)

PFRCR is a 16-bit counter that indicates the number of times a PAUSE frame is received. This register is cleared to 0 when it is read.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PFRXC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	PFRXC[15:0]	All 0	R	PAUSE Frame Receive Count These bits indicate the number of PAUSE frames received when flow control is enabled in reception (RXF bit in ECMR = 1).

Note: • When count-up and count-clearing of this register value occur simultaneously, count-clearing is performed.

### 26.3.20 TSU Counter Reset Register (TSU\_CTRST)

TSU\_CTRST clears the transmit, receive, and relay frame counters to 0.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	CTRST	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	CTRST	0	R/W	TSU Counter Reset When 1 is written to this bit, the values of registers TXNLCR0, TXALCR0, RXNLCR0, and RXALCR0 are cleared to 0. Writing 0 does not affect this bit. This bit is always read as 0.
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

### 26.3.21 VLANtag Set Register (TSU\_VTAG0)

TSU\_VTAG0 enables or disables the frame receive/discard evaluation function based on the VLAN number, and also sets the VLAN number.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	VTAG0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	VID0[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	VTAG0	0	R/W	Port 0 VLANtag Evaluation Function 0: Disables receive/discard evaluation for frames based on the VLAN number 1: Enables receive/discard evaluation for frames based on the VLAN number
30 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 0	VID0[11:0]	All 0	R/W	V-LAN ID Setting (VID) These bits set the VLAN number received by receive frames.

### 26.3.22 CAM Entry Table Busy Register (TSU\_ADSBSY)

When CAM entry table registers (TSU\_ADRH0 to TSU\_ADRH31 and TSU\_ADRL0 to TSU\_ADRL31) are set by register writing, the ADSBSY bit in this register is set to 1 (when the process of reflecting the contents of the CAM entry table registers in the CAM controller is completed inside the TSU, the ADSBSY bit is automatically restored to 0).

Access to TSU\_ADRH0 to TSU\_ADRH31 and TSU\_ADRL0 to TSU\_ADRL31 is prohibited, while the ADSBSY bit in this register is set to 1. This register is a read-only status register, which must not be written to.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ADS BSY
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	ADSBSY	0	R	CAM Entry Table Setting Busy When TSU_ADRH0 to TSU_ADRH31 and TSU_ADRL0 to TSU_ADRL31 are set by register writing, this bit is set to 1. When the process of reflecting the contents of the CAM entry table registers in the CAM controller is completed inside the TSU, this bit is automatically restored to 0. Access to TSU_ADRH0 to TSU_ADRH31 and TSU_ADRL0 to TSU_ADRL31 is prohibited, while this bit is set to 1. Writing to this register is also prohibited.



### 26.3.23 CAM Entry Table Enable Register (TSU\_TEN)

TSU\_TEN enables or disables the settings of TSU\_ADRH0 to TSU\_ADRH31 and TSU\_ADRL0 to TSU\_ADRL31.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TEN0	TEN1	TEN2	TEN3	TEN4	TEN5	TEN6	TEN7	TEN8	TEN9	TEN10	TEN11	TEN12	TEN13	TEN14	TEN15
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

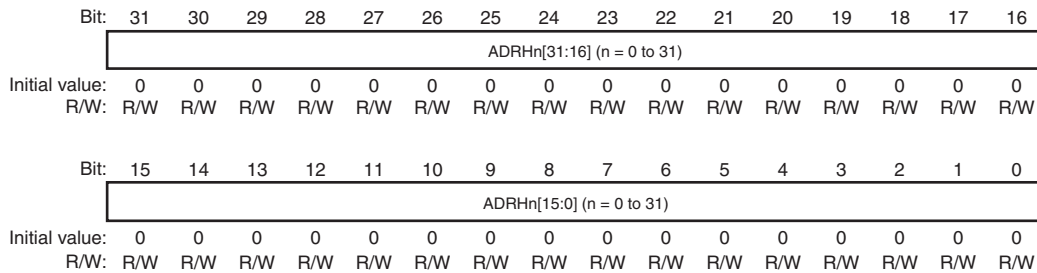
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TEN16	TEN17	TEN18	TEN19	TEN20	TEN21	TEN22	TEN23	TEN24	TEN25	TEN26	TEN27	TEN28	TEN29	TEN30	TEN31
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	TEN0	0	R/W	CAM Entry Table 0 (TSU_ADRH0 and TSU_ADRL0) Setting 0: Disabled 1: Enabled
30	TEN1	0	R/W	CAM Entry Table 1 (TSU_ADRH1 and TSU_ADRL1) Setting 0: Disabled 1: Enabled
29	TEN2	0	R/W	CAM Entry Table 2 (TSU_ADRH2 and TSU_ADRL2) Setting 0: Disabled 1: Enabled
28	TEN3	0	R/W	CAM Entry Table 3 (TSU_ADRH3 and TSU_ADRL3) Setting 0: Disabled 1: Enabled
27	TEN4	0	R/W	CAM Entry Table 4 (TSU_ADRH4 and TSU_ADRL4) Setting 0: Disabled 1: Enabled
26	TEN5	0	R/W	CAM Entry Table 5 (TSU_ADRH5 and TSU_ADRL5) Setting 0: Disabled 1: Enabled
25	TEN6	0	R/W	CAM Entry Table 6 (TSU_ADRH6 and TSU_ADRL6) Setting 0: Disabled 1: Enabled
24	TEN7	0	R/W	CAM Entry Table 7 (TSU_ADRH7 and TSU_ADRL7) Setting 0: Disabled 1: Enabled
23	TEN8	0	R/W	CAM Entry Table 8 (TSU_ADRH8 and TSU_ADRL8) Setting 0: Disabled 1: Enabled
22	TEN9	0	R/W	CAM Entry Table 9 (TSU_ADRH9 and TSU_ADRL9) Setting 0: Disabled 1: Enabled
21	TEN10	0	R/W	CAM Entry Table 10 (TSU_ADRH10 and TSU_ADRL10) Setting 0: Disabled 1: Enabled
20	TEN11	0	R/W	CAM Entry Table 11 (TSU_ADRH11 and TSU_ADRL11) Setting 0: Disabled 1: Enabled
19	TEN12	0	R/W	CAM Entry Table 12 (TSU_ADRH12 and TSU_ADRL12) Setting 0: Disabled 1: Enabled
18	TEN13	0	R/W	CAM Entry Table 13 (TSU_ADRH13 and TSU_ADRL13) Setting 0: Disabled 1: Enabled

Bit	Bit Name	Initial Value	R/W	Description
17	TEN14	0	R/W	CAM Entry Table 14 (TSU_ADRH14 and TSU_ADRL14) Setting 0: Disabled 1: Enabled
16	TEN15	0	R/W	CAM Entry Table 15 (TSU_ADRH15 and TSU_ADRL15) Setting 0: Disabled 1: Enabled
15	TEN16	0	R/W	CAM Entry Table 16 (TSU_ADRH16 and TSU_ADRL16) Setting 0: Disabled 1: Enabled
14	TEN17	0	R/W	CAM Entry Table 17 (TSU_ADRH17 and TSU_ADRL17) Setting 0: Disabled 1: Enabled
13	TEN18	0	R/W	CAM Entry Table 18 (TSU_ADRH18 and TSU_ADRL18) Setting 0: Disabled 1: Enabled
12	TEN19	0	R/W	CAM Entry Table 19 (TSU_ADRH19 and TSU_ADRL19) Setting 0: Disabled 1: Enabled
11	TEN20	0	R/W	CAM Entry Table 20 (TSU_ADRH20 and TSU_ADRL20) Setting 0: Disabled 1: Enabled
10	TEN21	0	R/W	CAM Entry Table 21 (TSU_ADRH21 and TSU_ADRL21) Setting 0: Disabled 1: Enabled
9	TEN22	0	R/W	CAM Entry Table 22 (TSU_ADRH22 and TSU_ADRL22) Setting 0: Disabled 1: Enabled
8	TEN23	0	R/W	CAM Entry Table 23 (TSU_ADRH23 and TSU_ADRL23) Setting 0: Disabled 1: Enabled
7	TEN24	0	R/W	CAM Entry Table 24 (TSU_ADRH24 and TSU_ADRL24) Setting 0: Disabled 1: Enabled
6	TEN25	0	R/W	CAM Entry Table 25 (TSU_ADRH25 and TSU_ADRL25) Setting 0: Disabled 1: Enabled
5	TEN26	0	R/W	CAM Entry Table 26 (TSU_ADRH26 and TSU_ADRL26) Setting 0: Disabled 1: Enabled
4	TEN27	0	R/W	CAM Entry Table 27 (TSU_ADRH27 and TSU_ADRL27) Setting 0: Disabled 1: Enabled
3	TEN28	0	R/W	CAM Entry Table 28 (TSU_ADRH28 and TSU_ADRL28) Setting 0: Disabled 1: Enabled
2	TEN29	0	R/W	CAM Entry Table 29 (TSU_ADRH29 and TSU_ADRL29) Setting 0: Disabled 1: Enabled
1	TEN30	0	R/W	CAM Entry Table 30 (TSU_ADRH30 and TSU_ADRL30) Setting 0: Disabled 1: Enabled
0	TEN31	0	R/W	CAM Entry Table 31 (TSU_ADRH31 and TSU_ADRL31) Setting 0: Disabled 1: Enabled

### 26.3.24 CAM Entry Table 0H to 31H Registers (TSU\_ADRH0 to TSU\_ADRH31)

TSU\_ADRH0 to TSU\_ADRH31 are entry tables referred to by the CAM in reception and relay. Each of these registers sets the upper 32 bits of the 48-bit MAC address. Maximum 32 entries of MAC addresses can be registered.



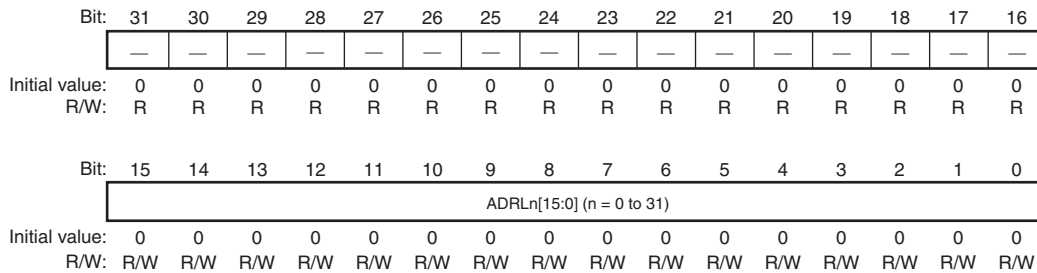
Bit	Bit Name	Initial Value	R/W	Description
31 to 0	ADRHn[31:0] (n: 0 to 31)	All 0	R/W	MAC Address Bits These bits set the upper 32 bits of the MAC address. When the MAC address is 01-23-45-67-89-AB (displayed in hexadecimal), set H'01234567 in this register.

Note: • Set the CAM entry tables following the procedure below.

1. Check that the ADSBSY bit in TSU\_ADSBSY is cleared to 0.
2. Set the upper 32 bits of the MAC addresses by TSU\_ADRH0 to TSU\_ADRH31.
3. Set the lower 16 bits of the MAC addresses by TSU\_ADRL0 to TSU\_ADRL31.

### 26.3.25 CAM Entry Table 0L to 31L Registers (TSU\_ADRL0 to TSU\_ADRL31)

TSU\_ADRL0 to TSU\_ADRL31 are entry tables referred to by the CAM in reception and relay. Each of these registers sets the lower 16 bits of the 48-bit MAC address. Maximum 32 entries of MAC addresses can be registered.



Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	ADRLn[15:0] (n: 0 to 31)	All 0	R/W	MAC Address Bits These bits set the lower 16 bits of the MAC address. When the MAC address is 01-23-45-67-89-AB (displayed in hexadecimal), set H'000089AB in this register.

Note: • Set the CAM entry tables following the procedure below.

1. Check that the ADSBSY bit in TSU\_ADSBSY is cleared to 0.
2. Set the upper 32 bits of the MAC addresses by TSU\_ADRH0 to TSU\_ADRH31.
3. Set the lower 16 bits of the MAC addresses by TSU\_ADRL0 to TSU\_ADRL31.

### 26.3.26 Transmit Frame Counter Register (Normal Transmission Only) (TXNLCR0)

TXNLCR0 is a 32-bit counter indicating the number of frames successfully transmitted in the E-MAC. When the value in this register reaches H'FFFFFFFF, count-up is halted. The counter is cleared to 0 by reading from this register. This register cannot be written to.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	NTC0[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NTC0[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	NTC0[31:0]	All 0	R	Transmit Frame Counter Bits These bits indicate the number of frames successfully transmitted.

Note: • When count-up and count-clearing of this register value occur simultaneously, count-clearing is performed.

### 26.3.27 Transmit Frame Counter Register (Normal and Erroneous Transmission) (TXALCR0)

TXALCR0 is a 32-bit counter indicating the number of frames transmitted in the E-MAC, including the number of frames erroneously transmitted. When the value in this register reaches H'FFFFFFFF, count-up is halted. The counter is cleared to 0 by reading from this register. This register cannot be written to.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TC0[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TC0[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	TC0[31:0]	All 0	R	Transmit Frame Counter Bits These bits indicate the number of frames successfully transmitted and erroneously transmitted.

Note: • When count-up and count-clearing of this register value occur simultaneously, count-clearing is performed.

### 26.3.28 Receive Frame Counter Register (Normal Reception Only) (RXNLCR0)

RXNLCR0 is a 32-bit counter indicating the number of frames successfully received in the E-MAC. When the value in this register reaches H'FFFFFFFF, count-up is halted. The counter is cleared to 0 by reading from this register. This register cannot be written to.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	NRC0[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NRC0[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	NRC0[31:0]	All 0	R	Receive Frame Counter Bits These bits indicate the number of frames successfully received.

Note: • When count-up and count-clearing of this register value occur simultaneously, count-clearing is performed.

### 26.3.29 Receive Frame Counter Register (Normal and Erroneous Reception) (RXALCR0)

RXALCR0 is a 32-bit counter indicating the number of frames received in the E-MAC, including the number of frames erroneously received. When the value in this register reaches H'FFFFFFFF, count-up is halted. The counter is cleared to 0 by reading from this register. This register cannot be written to.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RC0[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RC0[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	RC0[31:0]	All 0	R	Receive Frame Counter Bits These bits indicate the number of frames successfully received and erroneously received.

Note: • When count-up and count-clearing of this register value occur simultaneously, count-clearing is performed.

### 26.3.30 E-DMAC Start Register (EDSR)

EDSR specifies activation of the transmitting unit and receiving unit of the E-DMAC. This register can only be written to, and the read values are invalid.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ENT	ENR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	ENT	0	W	E-DMAC Transmitting Unit Start 0: Stops the E-DMAC transmitting unit 1: Starts the E-DMAC transmitting unit
0	ENR	0	W	E-DMAC Receiving Unit Start 0: Stops the E-DMAC receiving unit 1: Starts the E-DMAC receiving unit

### 26.3.31 E-DMAC Mode Register (EDMR)

EDMR is a 32-bit readable/writable register that specifies E-DMAC resetting and the transmit/receive descriptor length. This register is to be set before the transmitting or receiving function is enabled (before the TR bit in EDTRR or the RR bit in EDRRR is set to 1). However, the SWRR and SWRT bits can be written to even after the transmitting or receiving function is enabled. If a software reset is executed with this register during data transmission, abnormal data may be transmitted on the line. Execute a software reset with this register before specifying the transmit/receive descriptor length or modifying the settings of TDLAR, RDLAR, and so forth, the setting of ECMR (E-MAC mode register), and the settings of registers related to the E-DMAC and E-MAC operation.

To execute a software reset with this register, 1 must be written to both the SWRT and SWRR bits simultaneously. Writing 1 to the SWRT and SWRR bits initializes the E-MAC registers and E-DMAC registers, except for TDLAR, RDLAR, and RMFCR of the E-DMAC. The TSU registers (registers whose names are prefixed with TSU\_) are not initialized. The SWRT and SWRR bits in EDMR0 initializes the registers related to the E-DMAC and E-MAC. Note that during the period a software reset is issued (for 64 cycles of the internal bus clock Bck), accesses to all Ethernet-related registers are prohibited.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	DE	DL[1:0]	—	—	—	SWRT	SWRR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6	DE	0	R/W	Transmit/Receive Frame Endian Sets the endian mode for DMA transfer of frame data between the transmit/receive FIFO and transmit/receive buffer. 0: Big endian (longword access) 1: Little endian (longword access)
5, 4	DL[1:0]	00	R/W	Transmit/Receive Descriptor Length These bits specify the descriptor length. (See section 26.4.1, Descriptors and Descriptor List.) 00: 16 bytes 01: 32 bytes 10: 64 bytes 11: Setting prohibited
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	SWRT	0	R/W	Software Reset of Transmit FIFO Controller [Writing] 0: Disabled 1: Software reset started [Reading] 0: Software reset not executed (or completed) 1: Software reset being executed
0	SWRR	0	R/W	Software Reset of Receive FIFO Controller [Writing] 0: Disabled 1: Software reset started [Reading] 0: Software reset not executed (or completed) 1: Software reset being executed

### 26.3.32 E-DMAC Transmit Request Register (EDTRR)

EDTRR is a 32-bit readable/writable register that issues transmit directives to the E-DMAC. After writing 11 to bits TR[1:0] in this register, the E-DMAC reads the transmit descriptor at the address specified by TDLAR. If the TACT bit of this transmit descriptor is set to 1 (valid), transmit DMA transfer by the E-DMAC starts. When DMA transfer based on the first transmit descriptor is completed, the E-DMAC reads the next transmit descriptor. If the TACT bit of that transmit descriptor is set to 1 (valid), the E-DMAC continues transmit DMA operation. If the TACT bit of a transmit descriptor is cleared to 0 (invalid), the E-DMAC clears bits TR[1:0] and stops transmit DMA operation.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TR[1:0]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	TR[1:0]	00	R/W	Transmit Request 00, 01, 10: Transmission-halted state If 00, 01, or 10 is written to these bits, the E-DMAC stops DMA transfer of the currently processed transmit descriptor, reads the next transmit descriptor, and then clears these bits. (Write-back is completed for the valid transmit descriptors that have been detected up till then.) The E-DMAC clears these bits when transmit descriptor empty occurs, or transmission of a transmit descriptor has completed. (Write-back is completed for the valid transmit descriptors that have been detected up till then.) 11: Transmit DMA operation by E-DMAC After writing 11 to these bits, the E-DMAC starts reading a transmit descriptor.



### 26.3.33 E-DMAC Receive Request Register (EDRRR)

EDRRR is a 32-bit readable/writable register that issues receive directives to the E-DMAC. After writing 1 to the RR bit in this register, the E-DMAC reads the receive descriptor at the address specified by RDLAR. If the RACT bit of this receive descriptor is set to 1 (valid), and the receive FIFO holds a receive frame, the E-DMAC starts receive DMA transfer. When DMA transfer based on the first receive descriptor is completed, the E-DMAC reads the next receive descriptor. If the RACT bit of that receive descriptor is set to 1 (valid), the E-DMAC continues receive DMA operation. However, if the receive FIFO holds no receive data, the E-DMAC places receive DMA operation in the standby state. If the RACT bit of the receive descriptor is cleared to 0 (invalid), the E-DMAC clears the RR bit and stops receive DMAC operation.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

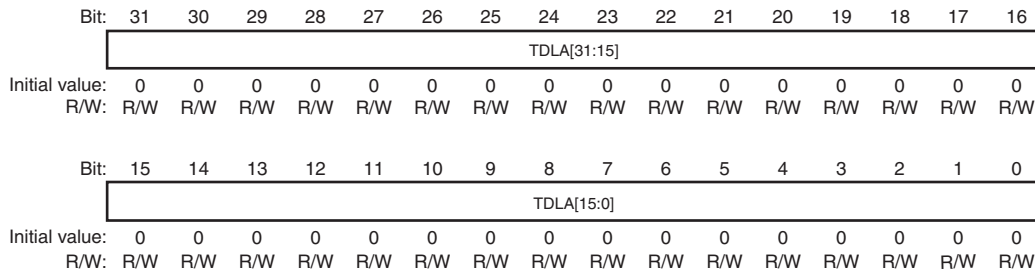
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	RR	0	R/W	Receive Request 0: Receiving function is disabled*1 If 0 is written to this bit, the E-DMAC stops receive operation after DMA transfer of one frame has completed and then clears this bit. The E-DMAC clears this bit when receive descriptor empty occurs. 1: Receive descriptor is read, and the E-DMAC is ready to receive.

Note 1. If the receiving function is disabled during frame reception, write-back is not performed successfully to the receive descriptor. Following pointers to read a receive descriptor become abnormal and the E-DMAC cannot operate successfully. In this case, to make E-DMAC reception enabled again, execute a software reset by the SWRT and SWRR bits in EDMR0. To disable the E-DMAC receiving function without executing a software reset, specify the RE bit in ECMR0. Next, after the E-DMAC has completed the reception and write-back to the receive descriptor has been confirmed, disable the receiving function using this register.

### 26.3.34 Transmit Descriptor List Start Address Register (TDLAR)

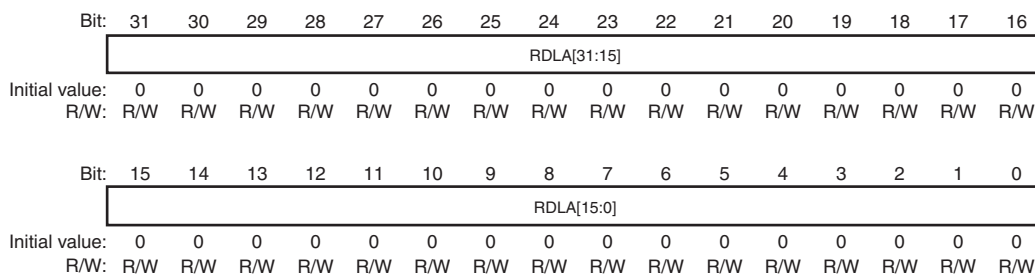
TDLAR is a 32-bit readable/writable register that specifies the start address of the transmit descriptor list. Descriptors have a boundary configuration in accordance with the descriptor length indicated by the DL bits in EDMR. This register must not be modified during transmission. Modifications to this register should only be made in the transmission-halted state specified by bits TR[1:0] (= 00) in the E-DMAC transmit request register (EDTRR).



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	TDLA[31:0]	All 0	R/W	Transmit Descriptor Start Address The lower bits are set according to the specified descriptor length. 16-byte boundary: TDLA[3:0] = 0000 32-byte boundary: TDLA[4:0] = 00000 64-byte boundary: TDLA[5:0] = 000000

### 26.3.35 Receive Descriptor List Start Address Register (RDLAR)

RDLAR is a 32-bit readable/writable register that specifies the start address of the receive descriptor list. Descriptors have a boundary configuration in accordance with the descriptor length indicated by the DL bits in EDMR. This register must not be modified during reception. Modifications to this register should only be made while reception is disabled by the RR bit (= 0) in the E-DMAC receive request register (EDRRR).



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	RDLA[31:0]	All 0	R/W	Receive Descriptor Start Address The lower bits are set according to the specified descriptor length. 16-byte boundary: RDLA[3:0] = 0000 32-byte boundary: RDLA[4:0] = 00000 64-byte boundary: RDLA[5:0] = 000000

### 26.3.36 E-MAC/E-DMAC Status Register (EESR)

EESR is a 32-bit readable/writable register that shows communications status information on the E-DMAC in combination with the E-MAC. The information in this register is reported in the form of interrupt sources. Individual bits are cleared by writing 1 (however, bit 22 (ECI) is a read-only bit that is not cleared by writing 1) and are not affected by writing 0. Each interrupt source can also be masked by means of the corresponding bit in the E-MAC/E-DMAC status interrupt permission register (EESIPR).

The interrupt generated by this status register is ETHERI. For interrupt priorities, see section 7.4, Interrupt Sources.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TWB[1:0]	TC[1]	TUC	ROC	TABT	RABT	RFCOF	—	ECI	TC[0]	TDE	TFUF	FR	RDE	RFOF	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	RMAF	—	—	RRF	RTLF	RTSF	PRE	CERF
Initial value:	0	0	0	0	0	Undefined	Undefined	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	TWB[1:0]	00	R/W	<p>Write-Back Complete</p> <p>Indicates that write-back from the E-DMAC to the corresponding descriptor after frame transmission has completed. This operation is enabled only when the TWBI bit in the transmit descriptor that includes the end of the transmit frame is set to 1.</p> <p>00: Write-back has not completed, or no transmission directive 11: Write-back has completed Others: Setting disabled</p>
29	TC[1]	0	R/W	<p>Frame Transmission Complete</p> <p>Indicates, in combination with the TC[0] bit, that all the data specified by the transmit descriptor has been transmitted from the E-MAC. This bit is set to 1 on assuming the completion of transmission. This is when transmission of one frame is completed and the transmit descriptor valid bit (TACT) of the next transmit descriptor not being set in single-frame/single-descriptor operation or when the last data of a frame has been transmitted and the transmit descriptor valid bit (TACT) of the next descriptor not being set in multi-buffer frame processing based on single-frame/multi-descriptor operation. After frame transmission has completed, the E-DMAC writes the transmission status back to the relevant descriptor.</p> <p>TC[1:0] 00: Transmission has not completed, or no transmission directive 11: Transmission has completed Others: Setting disabled</p>
28	TUC	0	R/W	<p>Transmit Underflow Frame Write-Back Complete</p> <p>0: Write-back has not completed for the frame causing transmit underflow 1: Write-back has completed for the frame causing transmit underflow</p>
27	ROC	0	R/W	<p>Receive Overflow Frame Write-Back Complete</p> <p>0: Write-back has not completed for the frame causing receive overflow 1: Write-back has completed for the frame causing receive overflow</p>
26	TABT	0	R/W	<p>Transmit Abort Detect</p> <p>Indicates that the E-MAC aborts transmitting a frame because of failures during frame transmission.</p> <p>0: Frame transmission has not been aborted or no transmission directive 1: Frame transmission has been aborted</p>
25	RABT	0	R/W	<p>Receive Abort Detect</p> <p>Indicates that the E-MAC aborts receiving a frame because of failures during frame reception.</p> <p>0: Frame reception has not been aborted or no reception directive 1: Frame reception has been aborted</p>

Bit	Bit Name	Initial Value	R/W	Description
24	RFCOF	0	R/W	Receive Frame Counter Overflow Indicates that the frame counter in the receive FIFO has overflowed. 0: Receive frame counter has not overflowed 1: Receive frame counter has overflowed
23	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
22	ECI	0	R	E-MAC Status Register Source This bit is a read-only bit. When the source of an ECSR interrupt is cleared, this bit is also cleared. 0: E-MAC status interrupt source has not been detected 1: E-MAC status interrupt source has been detected
21	TC[0]	0	R/W	Frame Transmission Complete Indicates, in combination with the TC[1] bit, that all the data specified by the transmit descriptor has been transmitted from the E-MAC. For details, see the description of the TC[1] bit.
20	TDE	0	R/W	Transmit Descriptor Empty Indicates that the transmit descriptor valid bit (TACT) of a transmit descriptor read by the E-DMAC is not set if the previous descriptor does not represent the end of a frame in multi-buffer frame processing based on single-frame/multi-descriptor operation. As a result, an incomplete frame may be sent. 0: Transmit descriptor active bit TACT = 1 detected 1: Transmit descriptor active bit TACT = 0 detected When transmit descriptor empty (TDE = 1) occurs, execute a software reset and initiate transmission. In this case, transmission starts from the address that is stored in the transmit descriptor list start address register (TDLAR).
19	TFUF	0	R/W	Transmit FIFO Underflow Indicates that an underflow has occurred in the transmit FIFO during frame transmission. Incomplete data is sent onto the line. 0: Underflow has not occurred 1: Underflow has occurred
18	FR	0	R/W	Frame Reception Indicates that a frame has been received and the receive descriptor has been updated. This bit is set to 1 each time a frame is received. 0: Frame has not been received 1: Frame has been received
17	RDE	0	R/W	Receive Descriptor Empty Indicates that the RACT bit of a receive descriptor read by the E-DMAC for receive DMA operation is cleared to 0 (invalid). When receive descriptor empty (RDE = 1) occurs, reception can be resumed by setting the RACT bit (cleared to 0) of the receive descriptor to 1 and then writing 1 to the RR bit in EDRRR. 0: Receive descriptor active bit RACT = 1 detected 1: Receive descriptor active bit RACT = 0 detected
16	RFOF	0	R/W	Receive FIFO Overflow Indicates that the receive FIFO has overflowed during frame reception. 0: Overflow has not occurred 1: Overflow has occurred
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10, 9	—	Undefined	R	Reserved The write value should always be 0.
8	—	0	R	Reserved The write value should always be 0.
7	RMAF	0	R/W	Receive Multicast Address Frame 0: Multicast address frame has not been received 1: Multicast address frame has been received
6, 5	—	All 0	R	Reserved The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
4	RRF	0	R/W	Receive Residual-Bit Frame 0: Residual-bit frame has not been received 1: Residual-bit frame has been received
3	RTLFL	0	R/W	Receive Too-Long Frame Indicates that a frame whose byte size exceeds the upper limit for the receive frame length set by RFLR has been received. 0: Too-long frame has not been received 1: Too-long frame has been received
2	RTSFL	0	R/W	Receive Too-Short Frame Indicates that a frame of fewer than 64 bytes has been received. 0: Too-short frame has not been received 1: Too-short frame has been received
1	PRE	0	R/W	PHY-LSI Receive Error 0: PHY-LSI receive error has not been detected 1: PHY-LSI receive error has been detected
0	CERF	0	R/W	CRC Error on Received Frame 0: CRC error has not been detected 1: CRC error has been detected

### 26.3.37 E-MAC/E-DMAC Status Interrupt Permission Register (EESIPR)

EESIPR is a 32-bit readable/writable register that enables interrupts corresponding to individual bits in the E-MAC/E-DMAC status register (EESR). An interrupt is enabled by writing 1 to the corresponding bit.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TWB1 IP	TWB0 IP	TC1 IP	TUC IP	ROC IP	TABT IP	RABT IP	RFCOF IP	—	ECI IP	TC0 IP	TDE IP	TFUF IP	FR IP	RDE IP	RFOF IP
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	RMAF IP	—	—	RRF IP	RTLF IP	RTSF IP	PRE IP	CERF IP	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	TWB1IP	0	R/W	Write-Back Complete Interrupt Enable 0: Write-back complete interrupt is disabled 1: Write-back complete interrupt is enabled
30	TWB0IP	0	R/W	Write-Back Complete Interrupt Enable 0: Write-back complete interrupt is disabled 1: Write-back complete interrupt is enabled
29	TC1IP	0	R/W	Frame Transmission Complete Interrupt Enable 0: Frame transmission complete interrupt is disabled 1: Frame transmission complete interrupt is enabled
28	TUCIP	0	R/W	Transmit Underflow Frame Write-Back Complete Interrupt Enable 0: Transmit underflow frame write-back complete interrupt is disabled 1: Transmit underflow frame write-back complete interrupt is enabled
27	ROCIP	0	R/W	Receive Overflow Frame Write-Back Complete Interrupt Enable 0: Receive overflow frame write-back complete interrupt is disabled 1: Receive overflow frame write-back complete interrupt is enabled
26	TABTIP	0	R/W	Transmit Abort Detect Interrupt Enable 0: Transmit abort detect interrupt is disabled 1: Transmit abort detect interrupt is enabled
25	RABTIP	0	R/W	Receive Abort Detect Interrupt Enable 0: Receive abort detect interrupt is disabled 1: Receive abort detect interrupt is enabled
24	RFCOFIP	0	R/W	Receive Frame Counter Overflow Interrupt Enable 0: Receive frame counter overflow interrupt is disabled 1: Receive frame counter overflow interrupt is enabled
23	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
22	ECIIP	0	R/W	E-MAC Status Register Source Interrupt Enable 0: E-MAC status interrupt is disabled 1: E-MAC status interrupt is enabled
21	TC0IP	0	R/W	Frame Transmission Complete Interrupt Enable 0: Frame transmission complete interrupt is disabled 1: Frame transmission complete interrupt is enabled
20	TDEIP	0	R/W	Transmit Descriptor Empty Interrupt Enable 0: Transmit descriptor empty interrupt is disabled 1: Transmit descriptor empty interrupt is enabled
19	TFUFIP	0	R/W	Transmit FIFO Underflow Interrupt Enable 0: Underflow interrupt is disabled 1: Underflow interrupt is enabled
18	FRIP	0	R/W	Frame Reception Interrupt Enable 0: Frame reception interrupt is disabled 1: Frame reception interrupt is enabled

Bit	Bit Name	Initial Value	R/W	Description
17	RDEIP	0	R/W	Receive Descriptor Empty Interrupt Enable 0: Receive descriptor empty interrupt is disabled 1: Receive descriptor empty interrupt is enabled
16	RFOFIP	0	R/W	Receive FIFO Overflow Interrupt Enable 0: Overflow interrupt is disabled 1: Overflow interrupt is enabled
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	RMAFIP	0	R/W	Receive Multicast Address Frame Interrupt Enable 0: Receive multicast address frame interrupt is disabled 1: Receive multicast address frame interrupt is enabled
6, 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	RRFIP	0	R/W	Receive Residual-Bit Frame Interrupt Enable 0: Receive residual-bit frame interrupt is disabled 1: Receive residual-bit frame interrupt is enabled
3	RTLFI	0	R/W	Receive Too-Long Frame Interrupt Enable 0: Receive too-long frame interrupt is disabled 1: Receive too-long frame interrupt is enabled
2	RTSFIP	0	R/W	Receive Too-Short Frame Interrupt Enable 0: Receive too-short frame interrupt is disabled 1: Receive too-short frame interrupt is enabled
1	PREIP	0	R/W	PHY-LSI Receive Error Interrupt Enable 0: PHY-LSI receive error interrupt is disabled 1: PHY-LSI receive error interrupt is enabled
0	CERFIP	0	R/W	CRC Error on Received Frame Interrupt Enable 0: CRC error interrupt is disabled 1: CRC error interrupt is enabled

### 26.3.38 Transmit/Receive Status Copy Enable Register (TRSCER)

TRSCER specifies whether the information for the transmit and receive state reported by bits 26, 25, and 10 to 0 in the E-MAC/E-DMAC status register (EESR) is to be reflected in the TFE or RFE bit of the corresponding descriptor. The bits in this register correspond to bits 26, 25, and 10 to 0 in EESR. When a bit is cleared to 0, the transmit status (bits 26 and 10 to 8 in EESR) is reflected in the TFE bit of the transmit descriptor, and the receive status (bits 25 and 7 to 0 in EESR) is reflected in the RFE bit of the receive descriptor. In this case, the state of a status bit set to 1 is reflected as the TFE or RFE bit set to 1. When a bit is set to 1, the occurrence of the corresponding source is not reflected in the descriptor. After this LSI is reset, all bits are cleared to 0.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TABT CE	RABT CE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	RMAF CE	—	—	RRF CE	RTLF CE	RTSF CE	PRE CE	CERF CE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17	TABTCE	0	R/W	TABT Bit Copy Directive 0: Reflects the TABT bit status in the TFE bit of the transmit descriptor 1: Occurrence of the corresponding source is not reflected in the TFE bit of the transmit descriptor
16	RABTCE	0	R/W	RABT Bit Copy Directive 0: Reflects the RABT bit status in the RFE bit of the receive descriptor 1: Occurrence of the corresponding source is not reflected in the RFE bit of the receive descriptor
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	RMAFCE	0	R/W	RMAF Bit Copy Directive 0: Reflects the RMAF bit status in the RFE bit of the receive descriptor 1: Occurrence of the corresponding source is not reflected in the RFE bit of the receive descriptor
6, 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	RRFCE	0	R/W	RRF Bit Copy Directive 0: Reflects the RRF bit status in the RFE bit of the receive descriptor 1: Occurrence of the corresponding source is not reflected in the RFE bit of the receive descriptor
3	RTLFCE	0	R/W	RTLF Bit Copy Directive 0: Reflects the RTLF bit status in the RFE bit of the receive descriptor 1: Occurrence of the corresponding source is not reflected in the RFE bit of the receive descriptor
2	RTSFCE	0	R/W	RTSF Bit Copy Directive 0: Reflects the RTSF bit status in the RFE bit of the receive descriptor 1: Occurrence of the corresponding source is not reflected in the RFE bit of the receive descriptor
1	PRECE	0	R/W	PRE Bit Copy Directive 0: Reflects the PRE bit status in the RFE bit of the receive descriptor 1: Occurrence of the corresponding source is not reflected in the RFE bit of the receive descriptor



Bit	Bit Name	Initial Value	R/W	Description
0	CERFCE	0	R/W	CERF Bit Copy Directive 0: Reflects the CERF bit status in the RFE bit of the receive descriptor 1: Occurrence of the corresponding source is not reflected in the RFE bit of the receive descriptor

### 26.3.39 Receive Missed-Frame Counter Register (RMFCR)

RMFCR is a 16-bit counter that indicates the number of frames that could not be saved in the receive buffer and so were discarded during reception. When the receive FIFO overflows, the receive frames in the FIFO are discarded. The number of frames discarded at this time is counted. When the value in this register reaches H'0000FFFF, count-up is halted. Clear the counter by writing H'00000000 in this register. Note that a value other than H'00000000 must not be written to this register.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MFC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	MFC[15:0]	All 0	R/W	Missed-Frame Counter These bits indicate the number of frames that are discarded and not transferred to the receive buffer during reception.

### 26.3.40 Transmit FIFO Threshold Register (TFTR)

TFTR is a 32-bit readable/writable register that specifies the transmit FIFO threshold at which the first transmission is started. The actual threshold is 4 times the set value. The E-MAC starts transmission when the amount of data in the transmit FIFO exceeds the number of bytes specified by this register, when the transmit FIFO is full, or when one frame of data write is performed. This register must not be written to during transmission (bits TR[1:0] in EDTRR = 11).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	TFT[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	TFT[10:0]	All 0	R/W	Transmit FIFO Threshold A value in 32-byte units and smaller than the FIFO size specified by FDR must be set as the transmit FIFO threshold. H'000: Store and forward modes H'008: 32 bytes H'010: 64 bytes H'018: 128 bytes : : H'07F: 508 bytes H'080: 512 bytes : : H'0FF: 1,020 bytes H'100: 1,024 bytes : : H'1FF: 2,044 bytes H'200: 2,048 bytes

Note: • When starting transmission before one frame of data write has completed, take care no underflow occurs.

### 26.3.41 FIFO Depth Register (FDR)

FDR is a 32-bit readable/writable register that specifies the sizes of the transmit and receive FIFOs.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	TFD[2:0]			—	—	—	RFD[4:0]				
Initial value:	0	0	0	0	0	1	1	1	0	0	0	1	1	1	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 8	TFD[2:0]	All 1	R/W	Transmit FIFO Size Specifies 256 bytes to 2 Kbytes in 256-byte units as the size of the transmit FIFO whose maximum size is 2 Kbytes. The setting must not be changed after transmission/reception has started. H'00: 256 bytes H'01: 512 bytes : : H'07: 2048 bytes
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4 to 0	RFD[4:0]	All 1	R/W	Receive FIFO Size Specifies 256 bytes to 4 Kbytes in 256-byte units as the size of the receive FIFO whose maximum size is 4 Kbytes. The setting must not be changed after transmission/reception has started. H'00: 256 bytes H'01: 512 bytes : : H'0F: 4096 bytes

### 26.3.42 Receiving Method Control Register (RMCR)

RMCR is a 32-bit readable/writable register that specifies the control method for the RE bit in ECMR while a frame is received. This register must be set during the receiving-halted state.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RNC
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	RNC	0	R/W	Receive Enable Control Sets whether to continue frame reception. 0: Upon completion of reception of one frame, the E-DMAC writes the receive status to the descriptor and clears the RR bit in EDRRR to 0. 1: Upon completion of reception of one frame, the E-DMAC writes (writes back) the receive status to the descriptor. In addition, the E-DMAC reads the next descriptor and prepares for reception of the next frame.

### 26.3.43 Receive Descriptor Fetch Address Register (RDFAR)

RDFAR stores the descriptor start address that is required when the E-DMAC fetches descriptor information from the receive descriptor. Which receive descriptor information is used for processing by the E-DMAC can be recognized by monitoring addresses displayed in this register. The address from which the E-DMAC is actually fetching a descriptor may be different from the value read from this register. In the initial setting, set the address of the receive descriptor at which receive processing is to be started.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RDFAR[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RDFAR[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	RDFAR[31:0]	All 0	R/W	Receive Descriptor Fetch Address Writing to these bits during the reception is prohibited.

### 26.3.44 Receive Descriptor Finished Address Register (RDFXR)

RDFXR stores the start address of the receive descriptor for which the E-DMAC has just completed the write-back processing. Up to which receive descriptor has been processed by the E-DMAC can be recognized by monitoring addresses displayed in this register. In the initial setting, set the address of the descriptor immediately before the descriptor that is pointed to by the address in RDFAR.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RDFX[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RDFX[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	RDFX[31:0]	All 0	R/W	Receive Descriptor Finished Address Writing to these bits during the reception is prohibited.

### 26.3.45 Receive Descriptor Final Flag Register (RDFFR)

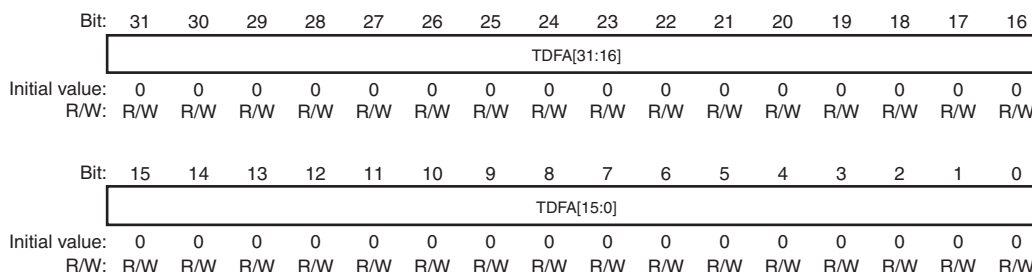
RDFFR indicates whether the receive descriptor for which the E-DMAC has just completed the write-back processing and whose start address is stored in RDFXR is at the end of the receive descriptor queue (descriptor list).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RDLF
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	RDLF	0	R/W	Receive Descriptor Queue Last Flag Indicates whether the receive descriptor for which the E-DMAC has just completed the write-back processing and whose start address is stored in RDFXR is at the end of the receive descriptor queue (descriptor list). 0: Not the last descriptor in the receive descriptor queue 1: Last descriptor in the receive descriptor queue

### 26.3.46 Transmit Descriptor Fetch Address Register (TDFAR)

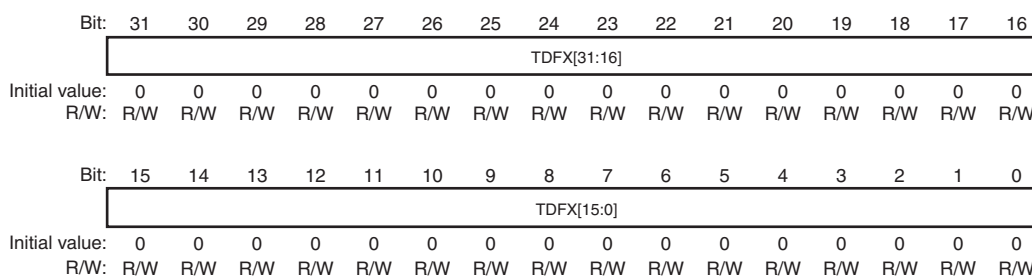
TDFAR stores the descriptor start address that is required when the E-DMAC fetches descriptor information from the transmit descriptor. Which transmit descriptor information is used for processing by the E-DMAC can be recognized by monitoring addresses displayed in this register. The address from which the E-DMAC is actually fetching a descriptor may be different from the value read from this register. In the initial setting, set the address of the transmit descriptor at which transmit processing is to be started.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	TDFAR[31:0]	All 0	R/W	Transmit Descriptor Fetch Address Writing to these bits during transmission is prohibited.

### 26.3.47 Transmit Descriptor Finished Address Register (TDFXR)

TDFXR stores the start address of the transmit descriptor for which the E-DMAC has just completed the write-back processing. Up to which transmit descriptor has been processed by the E-DMAC can be recognized by monitoring addresses displayed in this register. In the initial setting, set the address of the transmit descriptor immediately before the descriptor that is pointed to by the address in TDFAR.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	TDFXR[31:0]	All 0	R/W	Transmit Descriptor Finished Address Writing to these bits during transmission is prohibited.

### 26.3.48 Transmit Descriptor Final Flag Register (TDFFR)

TDFFR indicates whether the transmit descriptor for which the E-DMAC has just completed the write-back processing and whose start address is stored in TDFXR is at the end of the transmit descriptor queue (descriptor list).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TDLF
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	TDLF	0	R/W	Transmit Descriptor Queue Last Flag Indicates whether the transmit descriptor for which the E-DMAC has just completed the write-back processing and whose start address is stored in TDFXR is at the end of the transmit descriptor queue (descriptor list). 0: Not the last descriptor in the transmit descriptor queue 1: Last descriptor in the transmit descriptor queue

### 26.3.49 Overflow Alert FIFO Threshold Register (FCFTR)

FCFTR is a 32-bit readable/writable register that sets the flow control of the E-MAC. The threshold can be set by the size of the receive FIFO data (bits RFD[7:0]) and the number of receive frames (bits RFF[4:0]).

If the same receive FIFO size as set by the FIFO depth register (FDR) is set when flow control is turned on according to the RFD setting condition, flow control is turned on with (FIFO data size – 64) bytes. For instance, when the RFD bits in FDR is 7 and the RFD bits in this register is 7, flow control is turned on when (2,048 – 64) bytes of data is stored in the receive FIFO. The value set in the RFD bits in this register should be equal to or less than that set in the RFD bits in FDR.

Flow control is turned on when either of the setting conditions of bits RFF[4:0] and bits RFD[7:0] is satisfied. Flow control is turned off when neither of the conditions is satisfied (release).

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	RFF[4:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	RFD[7:0]							
Initial value:	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20 to 16	RFF[4:0]	H'17	R/W	Receive FIFO Overflow Alert Signal Output Threshold H'00: When one receive frame has been stored in the receive FIFO H'01: When two receive frames have been stored in the receive FIFO : : H'16: When 23 receive frames have been stored in the receive FIFO H'17: When 24 receive frames have been stored in the receive FIFO
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	RFD[7:0]	H'FF	R/W	Receive FIFO Overflow Alert Signal Output Threshold H'00: When (256 – 32) bytes of data is stored in the receive FIFO H'01: When (512 – 32) bytes of data is stored in the receive FIFO : : H'06: When (1,792 – 32) bytes of data is stored in the receive FIFO H'07: When (2,048 – 64) bytes of data is stored in the receive FIFO



### 26.3.50 Receive Data Padding Insert Register (RPADIR)

RPADIR is a 32-bit readable/writable register that inserts padding in receive data. To change the settings of this register, execute a software reset by means of the SWRT and SWRR bits in the E-DMAC mode register (EDMR) before making settings again.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	PADS[4:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PADR[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20 to 16	PADS[4:0]	H'00	R/W	Padding Size H'00: No padding insertion H'01: 1-byte insertion : : H'1F: 31-byte insertion
15 to 0	PADR[15:0]	H'0000	R/W	Padding Slot H'0000: Inserts specified size of padding at the first byte H'0001: Inserts specified size of padding at the second byte : : H'FFFF: Inserts specified size of padding at the 64K byte

### 26.3.51 Intelligent Checksum Mode Register (CSMR)

CSMR is a readable 32-bit register that specifies the intelligent checksum operation mode. This register must be set while reception is halted.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CSEBL	CSMD	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	SB[5:0]					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	CSELB	1	R	Intelligent Checksum Calculation Operation Setting 0: The result of checksum calculation is not written back to the receive descriptor. 1: The result of checksum calculation is written back to the receive descriptor.
30	CSMD	1	R/W	Intelligent Checksum Calculation Mode Setting 0: After having skipped the number of bytes specified in SB[5:0], counting from the beginning of the MAC-layer packet, the checksum is calculated for all subsequent data 1: MAC- or IP-layer packets are detected and checksums are calculated for upper-layer protocol packets such as TCP or UDP.
29 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5 to 0	SB[5:0]*1	011010	R/W	Intelligent Checksum Calculation Skip Bytes These bits specify the number of bytes to be skipped for checksum calculation, counting from the beginning of the data received in the E-DMAC. When padding bytes are to be added, specify the checksum start position to cover the amount or extent of padding. H'00: 0 bytes (meaning checksum calculation is performed from the beginning of the packet.) H'02: 2 bytes : : H'1A: 26 bytes H'3E: 62 bytes

Note 1. These bits should only be set when CSEL is 1 and CSMD is 0; otherwise, set to H'00.

### 26.3.52 Intelligent Checksum Skipped Bytes Monitor Register (CSSBM)

CSSBM is a 32-bit readable register that holds the number of bytes that have been skipped in received packets being handled by the E-DMAC. The number of skipped bytes can be monitored through this register. The amount of data received in E-DMAC may not match the number of skipped bytes.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	SBM[5:0]					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5 to 0	SBM[5:0]	000000	R	Number of Skipped Bytes These bits are read-only. Writing is prohibited. These bits are initialized when the beginning of a packet for reception is detected.

Note: • This register is only valid when CSEL is 1 and CSMD is 0.

### 26.3.53 Intelligent Checksum Monitor Register (CSSMR)

CSSMR is a 32-bit register that holds the checksum value of received packets being handled by the E-DMAC. The checksum value can be monitored through this register. The amount of data received in E-DMAC may not match the number of skipped bytes.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CS[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5 to 0	CS[15:0]	000000	R	Intelligent Checksum Value These bits are read-only. Writing is prohibited. These bits are initialized when the beginning of a packet for reception is detected.

Note: • This register is only valid when CSEL = 1 and CSMD = 0.

## 26.4 Operation

The ETHER consists of the following three function units:

- DMA transfer controller (E-DMAC): DMA transfer between the transmit/receive buffer in the memory and the transmit/receive FIFO
- MAC controller (E-MAC): Transmission/reception processing between the transmit/receive FIFO and the MII
- Transfer Switching Unit (TSU): CAM processing

Using its direct memory access (DMA) function, the E-DMAC performs DMA transfer of frame data between a user-specified Ethernet frame transmission/reception data storage destination (accessible memory space: transmit buffer/receive buffer) and the transmit/receive FIFO in the E-DMAC. The user cannot read and write data from and to the transmit/receive FIFO directly via the CPU.

To enable the E-DMAC to perform DMA transfer, information (data) including a transmit/receive data storage address and so forth, referred to as a descriptor, is required. The E-DMAC reads transmit data from the transmit buffer or writes receive data to the receive buffer according to the descriptor information. By arranging multiple descriptors as a descriptor row (list) (to be placed in a readable/writable memory space), multiple Ethernet frames can be transmitted or received continuously.

The E-MAC constructs an Ethernet frame using the data written to the transmit FIFO and transmits the frame to the MII. It also performs a CRC check of an Ethernet frame received from the MII and deconstructs the frame to write to the receive FIFO. The E-MAC supports the MII format for interface to the PHI-LSI connected externally to this LSI. The TSU, which is placed between the E-DMAC and E-MAC, references the CAM entry table to select one of the following tasks according to the Ethernet frame destination address (DA) input to the E-MAC.

- Receives data and writes to the receive FIFO.
- Discards data.

Figure 26.2 shows the frame data path and an overview of each setting.

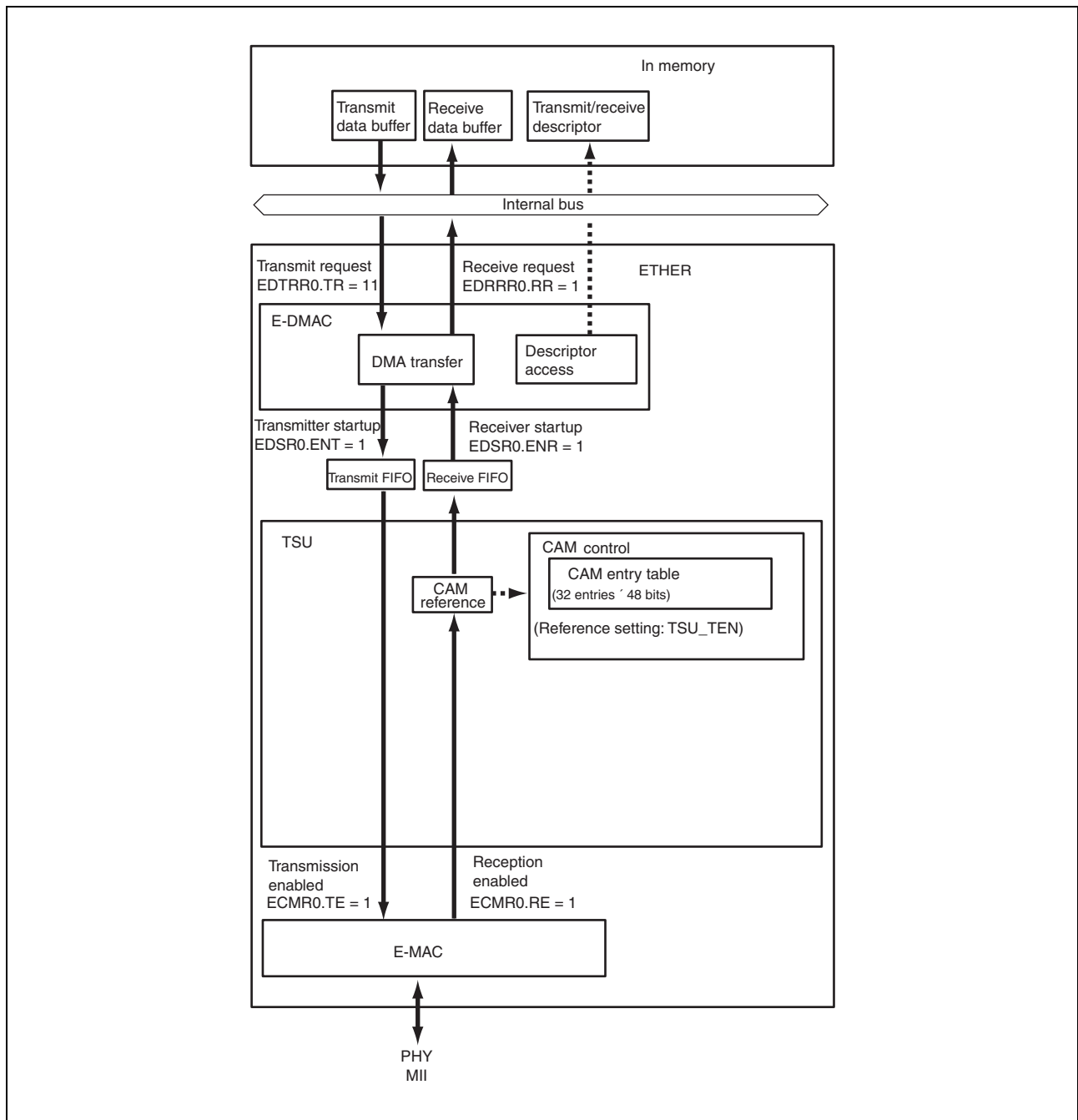


Figure 26.2 ETHER Data Path and Various Settings

### 26.4.1 Descriptors and Descriptor List

The E-DMAC performs DMA transfer according to the information (data), referred to as a descriptor, written in memory space. There are two types of descriptors: transmit descriptors and receive descriptors. Before a DMA transfer, DMA transfer information including a transmit/receive frame data storage address must be set by software.

The E-DMAC automatically starts reading a transmit/receive descriptor when the TR bits in EDTRR are set to 11 or the RR bit in EDRRR is set to 1, and performs DMA transfer of frame data between the transmit/receive buffer and transmit/receive FIFO according to the information stored in the descriptor. After completion of Ethernet frame transmission/reception, the E-DMAC disables the descriptor valid/invalid bit and reflects the result of transmission/reception in the status bits.

Descriptors are placed in a readable/writable memory space. The address of the start descriptor (descriptor to be read first by the E-DMAC) is set in TDLAR/RDLAR. When multiple descriptors are prepared as a descriptor row (descriptor list), the descriptors are placed in continuous addresses (memory) according to the descriptor length set in the DL0 and DL1 bits in EDMR.

#### (1) Transmit Descriptor

Figure 26.3 shows the configuration of a transmit descriptor and the relationship with a transmit buffer.

The data of a transmit descriptor consists of TD0, TD1, TD2, and padding data in groups of 32 bits from top to end. The length of padding data is determined according to the descriptor length specified by the DL0 and DL1 bits in EDMR. TD0 indicates whether the transmit descriptor is valid or invalid, and information about the descriptor configuration and status. TD1 indicates the length of data in a transmit buffer to be transferred (TDL) as specified by the descriptor. TD2 indicates the start address of a transmit buffer that holds data to be transferred (TBA).

Depending on the descriptor specification, one transmit descriptor can specify all transmit data of one frame (single-frame/single-buffer) or multiple descriptors can specify the transmit data of one frame (single-frame/multi-buffer). As an example of single-frame/multi-buffer operation, the data portion that is used in a fixed manner in each Ethernet frame transmission can be referenced by multiple descriptors. For example, multiple descriptors can share the destination address and transmit source address in an Ethernet frame, and the remaining data can be stored in each separate buffer.

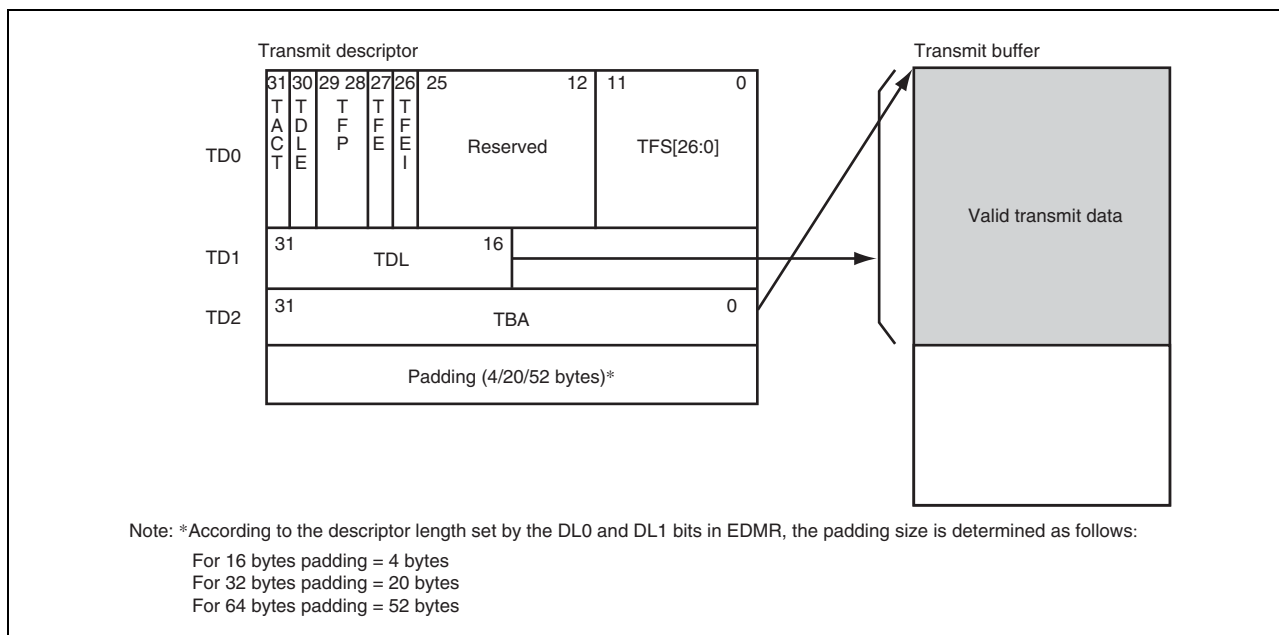


Figure 26.3 Relationship between Transmit Descriptor and Transmit Buffer

## (a) Transmit Descriptor 0 (TD0)

Before the TR bits in EDTRR are set to 11, the user sets whether the bits of the descriptor are valid or invalid bit and sets other descriptor configuration. After Ethernet frame transmission, the E-DMAC disables the valid/invalid bits of the descriptor and writes status information. This operation is referred to as write-back.

When using TD0, the user should write desired values to bits 31 to 28 and 26 according to the descriptor configuration. Bits 27 and 25 to 0 should be cleared to 0.

Bit	Bit Name	Initial Value	R/W	Description
31	TACT	0	R/W	<p>Transmit Descriptor Valid/Invalid Indicates whether the corresponding descriptor is valid or invalid. To make this bit valid, store transmit data in a transmit buffer (user-specified transmit data storage destination) beforehand, then write 1 to this bit. The E-DMAC clears this bit to 0 after data transfer.</p> <p>0: Indicates that this transmit descriptor is invalid Indicates the initial setting state, the state after 0 is written, or (in case the user writes 1 to this bit) that this bit is cleared to 0 because the E-DMAC data transfer processing is completed. If this state is recognized when the E-DMAC reads a descriptor, the E-DMAC clears the TR bit in EDTRR to 0, and halts transfer operation related to transmission by the E-DMAC.</p> <p>1: Indicates that this transmit descriptor is valid After the user writes 1 to this bit, this bit indicates that data is not transferred yet or data is being transferred. When there is a descriptor row (descriptor list) consisting of multiple continuous descriptors, the E-DMAC can continue operation when this bit of the next descriptor is valid.</p>
30	TDLE	0	R/W	<p>Transmit Descriptor List End Indicates whether the corresponding descriptor is the last descriptor of the descriptor row (descriptor list).</p> <p>0: Not last descriptor After transfer of the corresponding descriptor, the E-DMAC reads the next one in the list of continuous descriptors.</p> <p>1: Last descriptor After transfer of the corresponding descriptor, the E-DMAC reads the descriptor placed at the address indicated by TDLAR.</p>
29, 28	TFP[1:0]	00	R/W	<p>Transmit Frame Position These bits indicate whether information of this descriptor represents information about the start, middle, or end of the transmit frame.</p> <p>00: The information of the descriptor represents information about the middle of the frame.</p> <p>01: The information of the descriptor represents information about the end of the frame.</p> <p>10: The information of the descriptor represents information about the start of the frame.</p> <p>11: The information of the descriptor represents all information about the frame (single-frame/single-descriptor (single-buffer)).</p> <p>Reference When one frame is divided for use, the method of specifying this bit for a descriptor row according to the number of divisions is described below.</p> <ul style="list-style-type: none"> <li>For single-frame/single-descriptor operation First descriptor: TFP[1:0] = 11</li> <li>For single-frame/two-descriptor operation First descriptor: TFP[1:0] = 10 Second descriptor: TFP[1:0] = 01</li> <li>For single-frame/three-descriptor operation First descriptor: TFP[1:0] = 10 Second descriptor: TFP[1:0] = 00 Third descriptor: TFP[1:0] = 01</li> </ul> <p>When the number of divisions is large, a descriptor row is configured by adding intermediate descriptors with TFP[1:0] = 00.</p>

Bit	Bit Name	Initial Value	R/W	Description
27	TFE	0	R/W	Transmit Frame Error Occurrence Indicates that an error occurred in the transmit frame. 0: The TFS11 to TFS0 bits are all 0 1: One of the TFS11 to TFS0 bits is 1 The TFS8 to TFS0 bits can be masked for each factor by using TRSCER. The TFS11 to TFS9 bits cannot be masked. This bit is set by the E-DMAC write-back operation.
26	TWBI	0	R/W	Write-Back Completion Interrupt Notification 0: Does not notify of a write-back completion interrupt 1: After a write-back operation to this descriptor is complete, this bit sets the TWB1 and TWB0 bits in EESR to 11 and notifies the CPU of a write-back completion interrupt. This bit is valid only for the descriptor including the end of transmit frame (TFP = 01 or 11). This bit is cleared to 0 by the E-DMAC write-back operation.
25 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 0	TFS [11:0]	All 0	R/W	Transmit Frame Status These bits indicate the status of the corresponding frame. A bit below, which is set by the E-DMAC write-back operation, indicates the occurrence of the corresponding event when set to 1. <ul style="list-style-type: none"> <li>TFS[11:10]: Reserved (The write value should always be 0.)</li> <li>TFS[9]: Transmit FIFO underflow (Corresponding to the TUC bit in EESR)</li> <li>TFS[8]: Detection of transmission abort (Corresponding to the TABT bit in EESR)</li> <li>TFS[7:0]: Reserved (The write value should always be 0.)</li> </ul>

### (b) Transmit Descriptor 1 (TD1)

TD1 indicates the data length of the transmit buffer used by the corresponding descriptor.

The user should set TD1 before the start of a read by the E-DMAC.

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	TDL [15:0]	All 0	R/W	Transmit Buffer Data Length (in bytes) These bits set the data length of the corresponding transmit buffer in bytes. The specifiable data lengths are from a minimum of 1 (H'0001) byte to a maximum of 64 K – 32 (H'FFE0) bytes.
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

### (c) Transmit Descriptor 2 (TD2)

TD2 indicates the start address of the corresponding 32-bit width transmit buffer. An address value should be specified in a 16-byte boundary.

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	TBA [31:0]	All 0	R/W	Transmit Buffer Start Address These bits set the start address of the corresponding transmit buffer in a 16-byte boundary.

If descriptors are set below, the E-DMAC does not return to normal operation until a system reset is performed.

- TFP (transmit frame position) is not logically correct  
Example: The TFP bits are set to 11 in a descriptor (descriptor A) and the TFP bits are set to 01 in the next descriptor (descriptor B). This specification means that there is no descriptor indicating the start of the transmit



frame specified by descriptor B.

- TBL (transmit buffer length) is set to 0

When one transmit frame is divided into three parts or more with transmit descriptors, the E-DMAC performs the following write-back operation:

- A write-back operation is performed for a transmit descriptor including information for the start of the transmit frame (TFP = 10 or 11) and for a transmit descriptor including information for the end of the frame (TFP = 01 or 11).
- A write-back operation is not performed for a transmit descriptor for the middle of the frame (TFP = 00).

However, TFE (transmit frame error occurrence) or TFS (transmit frame status) is written only to a transmit descriptor including information for the end of the frame (TFP = 01 or 11) by a write-back operation.

Before changing a transmit descriptor with the software, make sure that a write-back operation has been performed (TACT = 0) for the transmit descriptor including information for the end of the frame (TFP = 01 or 11) to avoid overwriting (re-setting) an unprocessed transmit descriptor.

## (2) Receive Descriptor

Figure 26.4 shows the relationship between a receive descriptor and receive buffer.

The data of a receive descriptor consists of RD0, RD1, RD2, and padding data in groups of 32 bits from top to end. The length of padding data is determined according to the descriptor length specified by the DL0 and DL1 bits in EDMR. RD0 indicates whether the receive descriptor is valid or invalid, and information about descriptor configuration and status. RD1 indicates the length of data that can be received in the receive buffer specified by the descriptor (RBL) and the length of the received frame data (RDL). RD2 indicates the start address of the receive buffer for storing receive data (RBA).

Depending on the descriptor specification, one receive descriptor can specify the storing of all receive data of one frame in a receive buffer (single-frame/single-buffer). All receive frames can be stored in a single buffer if RBL of each descriptor is set to more than 1514 bytes (maximum Ethernet frame length).

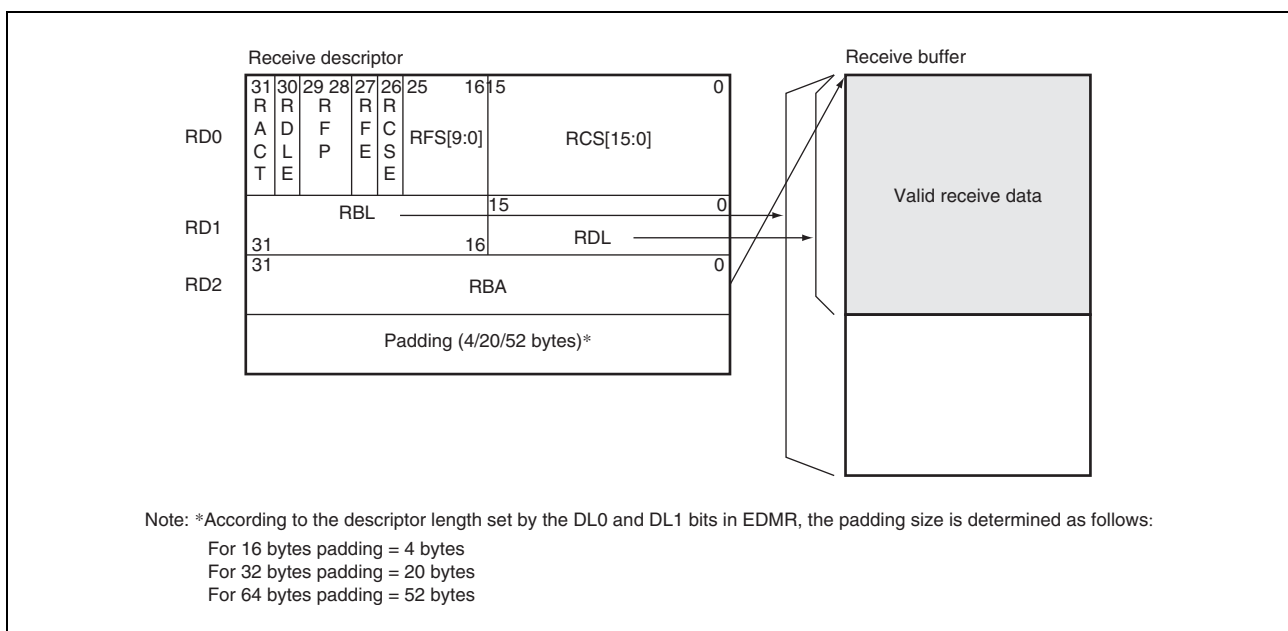


Figure 26.4 Relationship between Receive Descriptor and Receive Buffer

## (a) Receive Descriptor 0 (RD0)

The user sets whether the bits of the descriptor are valid or invalid and whether the descriptor represents the end of the descriptor list in RD0 before the RR bit in EDRRR is set to 1 and the start of a read by the E-DMAC. After receive DMA transfer of an Ethernet frame by the E-DMAC, the E-DMAC disables the valid/invalid bits of the descriptor and writes status information. This operation is referred to as write-back.

When using RD0, the user should write desired values to bits 31 and 30 according to the descriptor configuration. Bits 29 to 0 should be cleared to 0.

Bit	Bit Name	Initial Value	R/W	Description
31	RACT	0	R/W	<p>Receive Descriptor Valid/Invalid</p> <p>Indicates whether this descriptor is valid or invalid. To make this bit valid, prepare a receive buffer (user-specified receive data storage destination) beforehand, then write 1 to this bit. The E-DMAC clears this bit to 0 after data transfer.</p> <p>0: Indicates that this receive descriptor is invalid</p> <p>Indicates the initial setting state, the state after 0 is written to, or (in case the user writes 1 to this bit) that this bit is cleared to 0 because the E-DMAC data transfer processing is completed</p> <p>If this state is recognized when the E-DMAC reads a descriptor, the E-DMAC clears the RR bit in EDRRR to 0, and halts transfer operation related to reception by the E-DMAC</p> <p>1: Indicates that this receive descriptor is valid</p> <p>Indicates that data is not transferred yet after the user writes 1 to this bit, or that data is being transferred</p> <p>When there is a descriptor row (descriptor list) consisting of multiple continuous descriptors, the E-DMAC can continue operation when this bit of the next descriptor is valid</p>
30	RDLE	0	R/W	<p>Receive Descriptor List End</p> <p>Indicates whether this descriptor is the last descriptor of the descriptor row (descriptor list).</p> <p>0: Not last descriptor</p> <p>After transfer of this descriptor, the E-DMAC reads the next one in the list of continuous descriptors</p> <p>1: Last descriptor</p> <p>After transfer of this descriptor, the E-DMAC reads the descriptor placed at the address indicated by RDLAR</p>
29, 28	RFP[1:0]	00	R/W	<p>Receive Frame Position 1, 0</p> <p>The E-DMAC indicates by write-back operation whether information of the corresponding descriptor represents information about the start, middle, or end of the receive frame.</p> <p>00: The information of the descriptor represents information about the middle of the frame</p> <p>01: The information of the descriptor represents information about the end of the frame</p> <p>10: The information of the descriptor represents information about the start of the frame</p> <p>11: The information of the descriptor represents all information about the frame (single-frame/single-descriptor (single-buffer))</p> <p>Reference</p> <p>The relationship between a frame after reception of one frame and a descriptor is described below.</p> <p>When the receive buffer data length is less than the receive frame length, these bits are set to a value other than 11 after reception. When they are set so, reception processing should be started after a software reset.</p>
27	RFE	0	R/W	<p>Receive Frame Error Occurrence</p> <p>Indicates that an error occurred in the receive frame.</p> <p>0: RFS11 to RFS0 are all 0</p> <p>1: One of RFS11 to RFS0 is 0</p> <p>Each of RFS8 to RFS0 can be masked by using TRSCER. RFS11 to RFS9 cannot be masked.</p> <p>This bit is set by the E-DMAC write-back operation.</p>

Bit	Bit Name	Initial Value	R/W	Description
26	RCSE	0	R/W	Receive Packet Checksum Value Evaluation by Intelligent Checksum When CSEBL = 1 and CSMD = 1, the value of this bit is set as shown in Table 26.3, according to the receive packet and receive data. The information of this bit is invalid when operation is performed with a setting other than above.
25 to 16	RFS[9:0]	All 0	R/W	Receive Frame Status These bits indicate the error status during frame reception. RFS9: Receive FIFO overflow (corresponding to the RFOF bit in EESR) RFS8: Reserved (write value should be 0) RFS7: Multicast address frame received (corresponding to the RMAF bit in EESR) RFS6: CAM entry unregistered frame received (corresponding to the RUAF bit in EESR) RFS5: Reserved (write value should be 0) RFS4: Residual-bit frame receive error (corresponding to the RRF bit in EESR) RFS3: Long frame receive error (corresponding to the RTLf bit in EESR) RFS2: Short frame receive error (corresponding to the RTSF bit in EESR) RFS1: PHY-LSI receive error (corresponding to the PRE bit in EESR) RFS0: CRC error on receive frame (corresponding to the CERF bit in EESR)
15 to 0	RCS[15:0]	All 0	R/W	Receive Packet Checksum Value in Intelligent Checksum

**Table 26.3 RCSE State Determined by Receive Packet Type and Receive Data**

IP Version	Frame Type Option and Extension Header	Normal Data		Abnormal Data	
		RCS[15:0]	RCSE	RCS[15:0]	RCSE
IPv4	None	H'FFFF H'0000	0	Undefined	1
	Fragment	Undefined	Undefined	Undefined	Undefined
	Option	H'FFFF H'0000	0	Undefined	1
IPv6	None	H'FFFF H'0000	0	Undefined	1
	Hop-by-hop	H'FFFF H'0000	0	Undefined	1
	Routing	H'FFFF H'0000	0	Undefined	1
	Destination options	H'FFFF H'0000	0	Undefined	1
	AH	H'FFFF H'0000	0	Undefined	1
	Fragment	Undefined	Undefined	Undefined	Undefined
	ESP	H'0000	1	H'0000	1
	MobileIPv6	H'0000	1	H'0000	1
	Others	H'0000	1	H'0000	1
Other than IPv4 or IPv6		H'0000	0	H'0000	0

**(b) Receive Descriptor 1 (RD1)**

In RD1, the user specifies the data length of a receive buffer usable by the corresponding descriptor. After reception of a frame, RD1 indicates the length of a frame received by the E-DMAC.

The user should set RD1 before the start of a read by the E-DMAC.

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	RBL [15:0]	All 0	R/W	Receive Buffer Data Length (in bytes, to be specified with a 32-byte boundary) These bits set the length of data that can be received by the corresponding receive buffer with an integral multiple of 32 bytes. The specifiable data lengths are from a minimum of 32 (H'0020) bytes to a maximum of 64 K – 32 (H'FFE0) bytes.
15 to 0	RDL [15:0]	All 0	R	Receive Data Length These bits indicate the data length of a receive frame stored in the receive buffer. Receive data transferred to the receive buffer does not include CRC data (4 bytes) placed at the end of a frame. As a receive frame length, the number of bytes (valid data bytes) not including CRC data are reported. In single-frame/multi-buffer (descriptor) operation, only the receive data length of the last descriptor is valid. The receive data length of an intermediate descriptor has no meaning. The maximum frame length that can be received is: When padding function is invalid: 64 Kbytes between 1 byte (H'FFFF) When padding function is valid: 64 Kbytes between 32 bytes (H'FFE0)

**(c) Receive Descriptor 2 (RD2)**

RD2 indicates the start address of the corresponding receive buffer. Set the start address of a receive buffer with a 32-byte boundary.

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	RBA [31:0]	All 0	R/W	Receive Buffer Start Address These bits set the start address of the corresponding receive buffer with a 32-byte boundary.

The E-DMAC performs DMA transfer for a receive frame from the address specified by RBA (receive buffer address) to the receive buffer in 32-byte units. RBL (receive buffer length) must be set to be an integral multiple of 32 bytes.

If data to be transferred is less than 32 bytes, invalid data will be written to.

[Example]

When the receive frame length is 170 bytes and the required receive buffer capacity is 192 bytes (32 bytes × 6), the sixth DMA-transfer causes invalid data to be written to the receive buffer (In the 32-byte DMA data, the former 10 bytes are valid and the latter 22 bytes are invalid).

Padding of the value 0 can be inserted into only one position in the receive frame by setting RPADIR. The padding size can be selected from 1 byte to 31 bytes in byte units. When padding is inserted into a receive frame, a receive buffer area equal to the total of "receive frame length and padding size" is required. RPADIR setting is valid for all receive frames. RFE (receive frame error occurrence), PV (padding insertion), RFS (receive frame status) and RFS (receive frame status) are only set in the receive descriptor including information for the end of the frame (TFP = 01 or 11) by a write-back operation.

Before re-setting a receive descriptor with the software, completion of a write-back operation for the receive descriptor (RACT = 0) must be confirmed to avoid rewriting to (and re-setting) an unprocessed receive descriptor.

(3) Descriptor and Transmit/Receive Buffer

(a) Transmission

Each transmit descriptor specifies one transmit buffer. The E-DMAC transfers a transmit frame stored in a transmit buffer specified by a transmit descriptor to the transmit FIFO. Multiple transmit frames stored in transmit buffers specified by multiple descriptors can be connected into one transmit frame and transferred to the transmit FIFO.

Figure 26.5 shows the relationship between the transmit descriptors and transmit buffers.

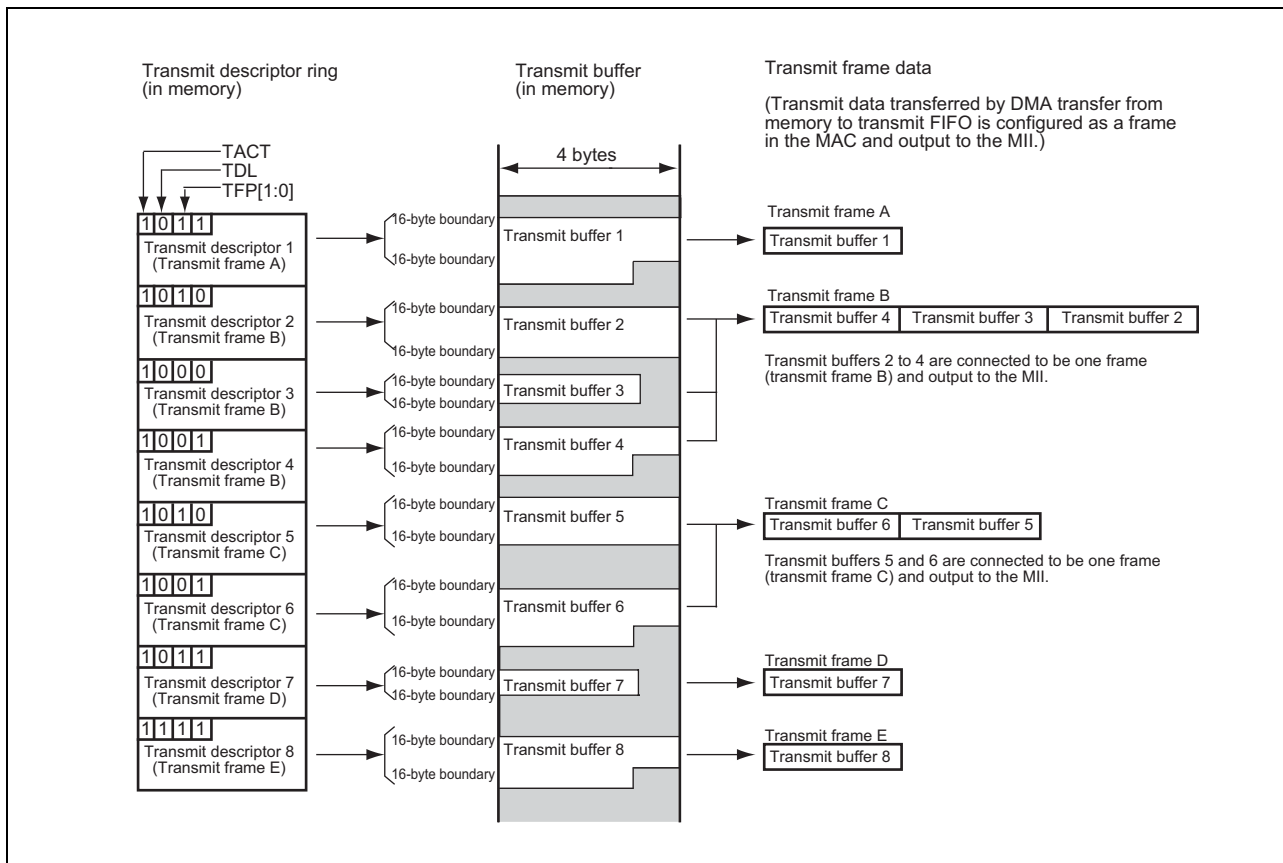


Figure 26.5 Relationship between Transmit Descriptor and Transmit Buffer

(b) Reception

Each receive descriptor specifies one receive buffer. The E-DMAC receives a receive frame from the receive FIFO and stores it in a receive buffer specified by a receive descriptor. If the receive frame size exceeds the receive buffer size, the remaining data of the receive frame can be stored in a different receive buffer specified by a different receive descriptor. Thus, one receive frame can be stored in multiple receive buffers.

Figure 26.6 shows the relationship between the receive descriptors and receive buffers.

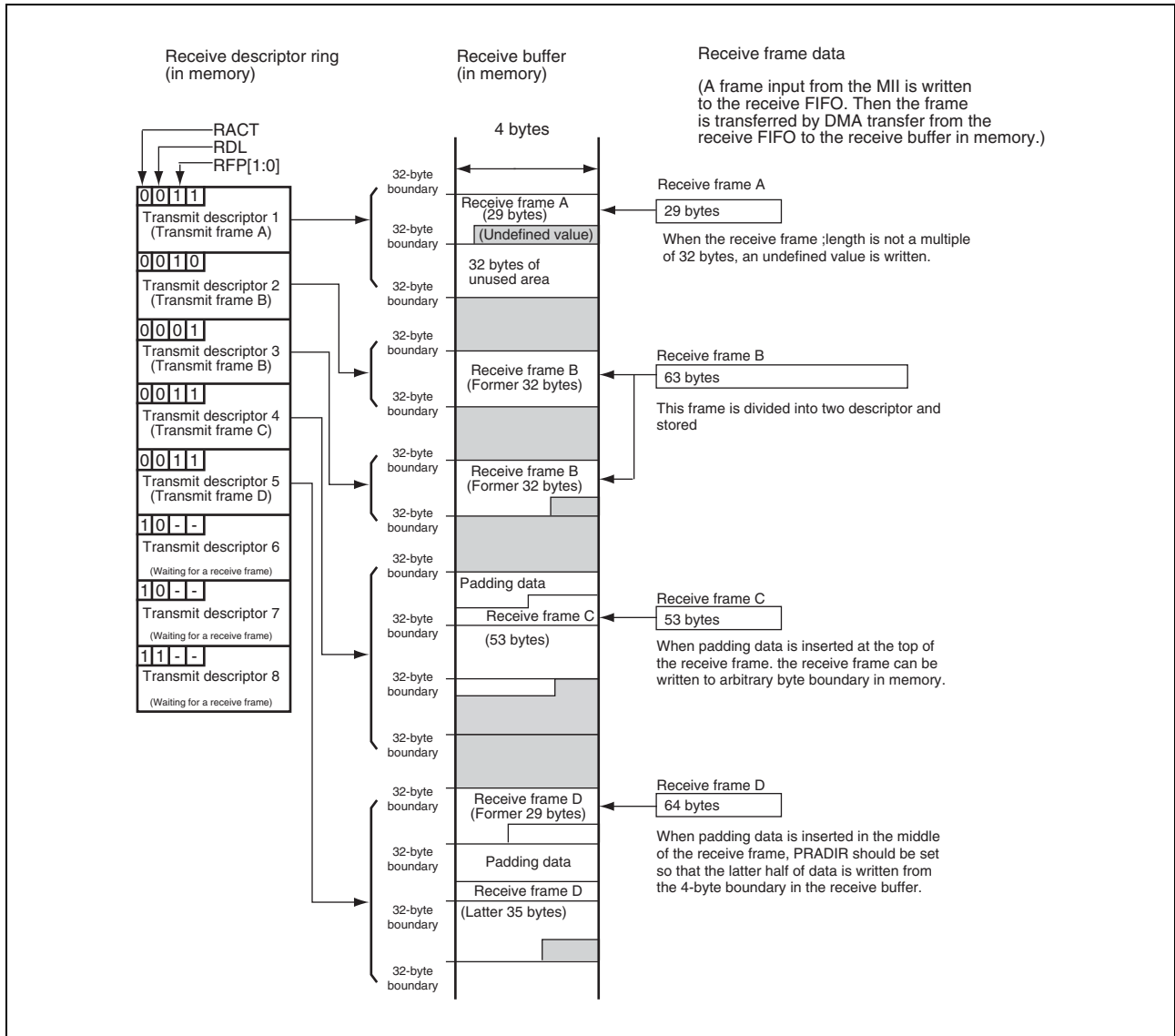


Figure 26.6 Relationship between Receive Descriptor and Receive Buffer

#### (4) Descriptor Pointer

The E-DMAC controls the transmit and receive descriptor addresses in memory and the processing priority by using the following registers.

1. Registers related to a transmit descriptor
  - TDLAR: Address of the start descriptor in a list of transmit descriptors.
  - TDFAR: Address of the transmit descriptor to be processed
  - TDFXR: Address of the transmit descriptor that finished processing (set by a write-back operation) last
  - TDFFR (DL bit): Indicates whether the TDLE value of the transmit descriptor specified by TDFXR is 1 or not.
2. Registers related to receive descriptor:
  - RDLAR: Address of the start descriptor in a list of receive descriptors.
  - RDFAR: Address of the receive descriptor to be processed
  - RDFXR: Address of the receive descriptor that finished processing (set by a write-back operation) last
  - RDFFR (DL bit): Indicates whether the RDLE value of the receive descriptor specified by RDFXR is 1 or not.

Transmit descriptors and receive descriptors have a ring structure. When the TDLE (RDLE) value of the processed transmit (receive) descriptor is 0, the next descriptor will be processed. The next descriptor is the transmit (receive) descriptor at the address obtained by adding the processed transmit (receive) descriptor address to the descriptor length specified by the DL bits in EDMR. When the TDLE (RDLE) value of the processed transmit (receive) descriptor is 1, the transmit descriptor indicated by TDLAR (RDLAR) will be processed next. **Figure 26.7** shows the relationship between the transmit/receive descriptor ring and read pointer.

The transmit descriptor list must be large enough to point to five or more transmit frames. If four or less transmit frames are pointed to in a list, E-DMAC operation is not guaranteed. Accordingly, do not set that all the transmit descriptors in a ring are used by four or less descriptors. The receive descriptor list does not have this restriction. For example, one receive frame can use all receive descriptors in a list.

In the initial setting, the start address of a descriptor list must be set to TDLAR (RDLAR) and TDFAR (RDFAR), and the end descriptor address of the descriptor list to TDFXR (RDFXR) by the software.

The E-DMAC updates TDFAR (RDFAR), TDFXR (RDFXR), and the DL bit in TDFFR (DL bit in RDFFR) each time a descriptor is processed.

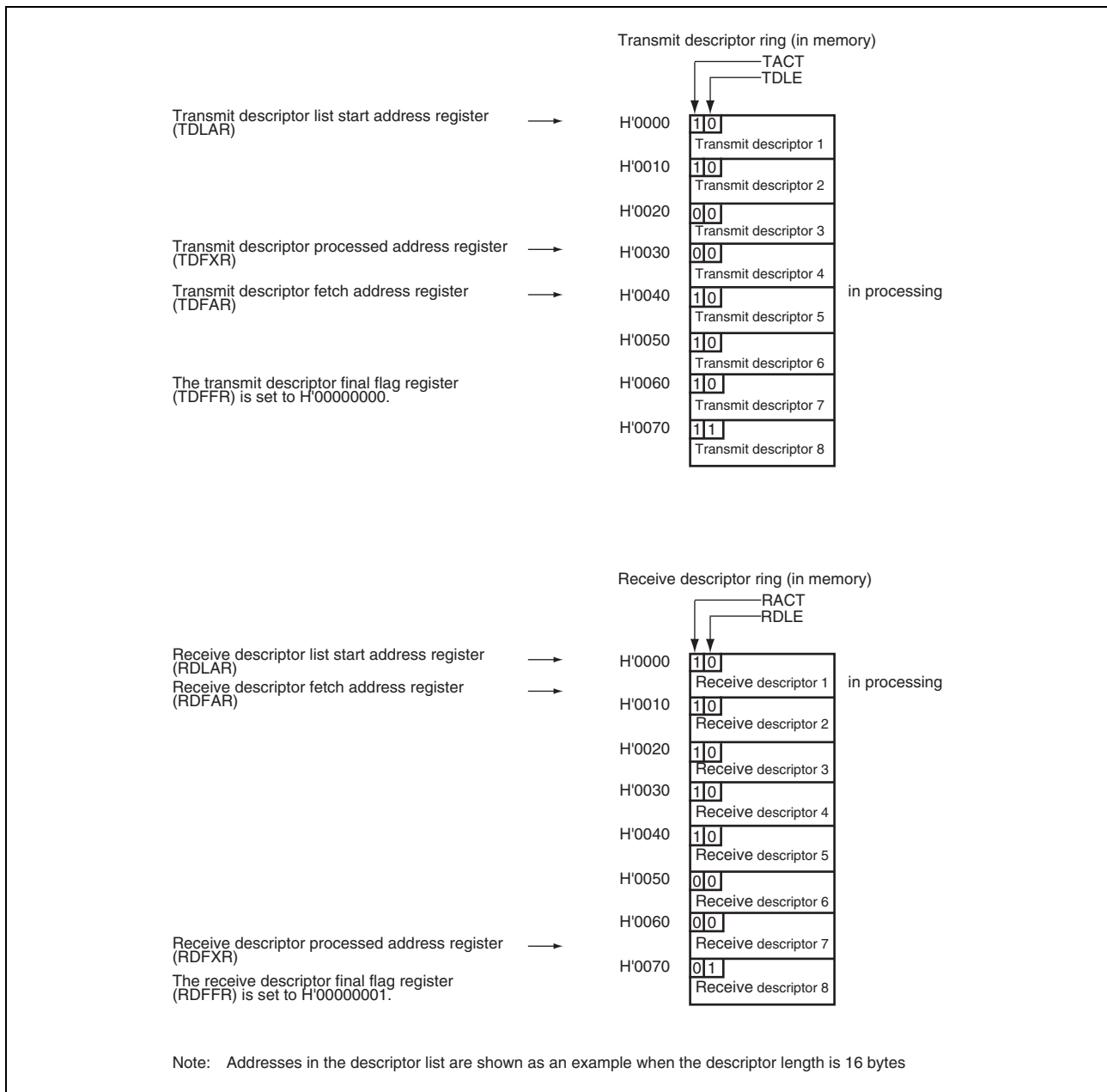


Figure 26.7 Relationship between Transmit/Receive Descriptor and Descriptor Pointing Registers



## 26.4.2 Transmission

### (1) Transmission Procedure and Processing Flow

When 11 is written to the TR bits in EDTRR with the TE bit in ECMR set to 1 and there is empty space of 32 bytes or more in the transmit FIFO, the E-DMAC reads the descriptor following the previously used descriptor from the transmit descriptor list (or the descriptor indicated by TDLAR at the initial startup).

If the TACT bit of the read descriptor is set to 1 (valid), the E-DMAC sequentially reads transmit frame data from the transmit buffer start address specified by TD2 and transfers the data to the transmit FIFO. The E-DMAC configures a transmit frame and starts transmission to the MII. After DMA transfer of data equivalent to the buffer length specified in the descriptor, the following processing is carried out according to the TFP value.

- TFP = 10 (start of a frame)  
Descriptor write-back (writing 0 to the TACT bit) is performed after completion of DMA transfer.
- TFP = 01 or 11 (end of a frame)  
Descriptor write-back (writing 0 to the TACT bit and writing status) is performed after completion of frame transmission.
- TFP = 00 (frame continued)  
Descriptor write-back is not performed. The TACT bit retains the value 1.

As long as the TACT bit of a read descriptor is set to 1 (valid), the reading of E-DMAC descriptors and the transmission of frames continue.

When a descriptor with the TACT bit cleared to 0 (invalid) is read, the E-DMAC performs the following processing and completes transmit processing.

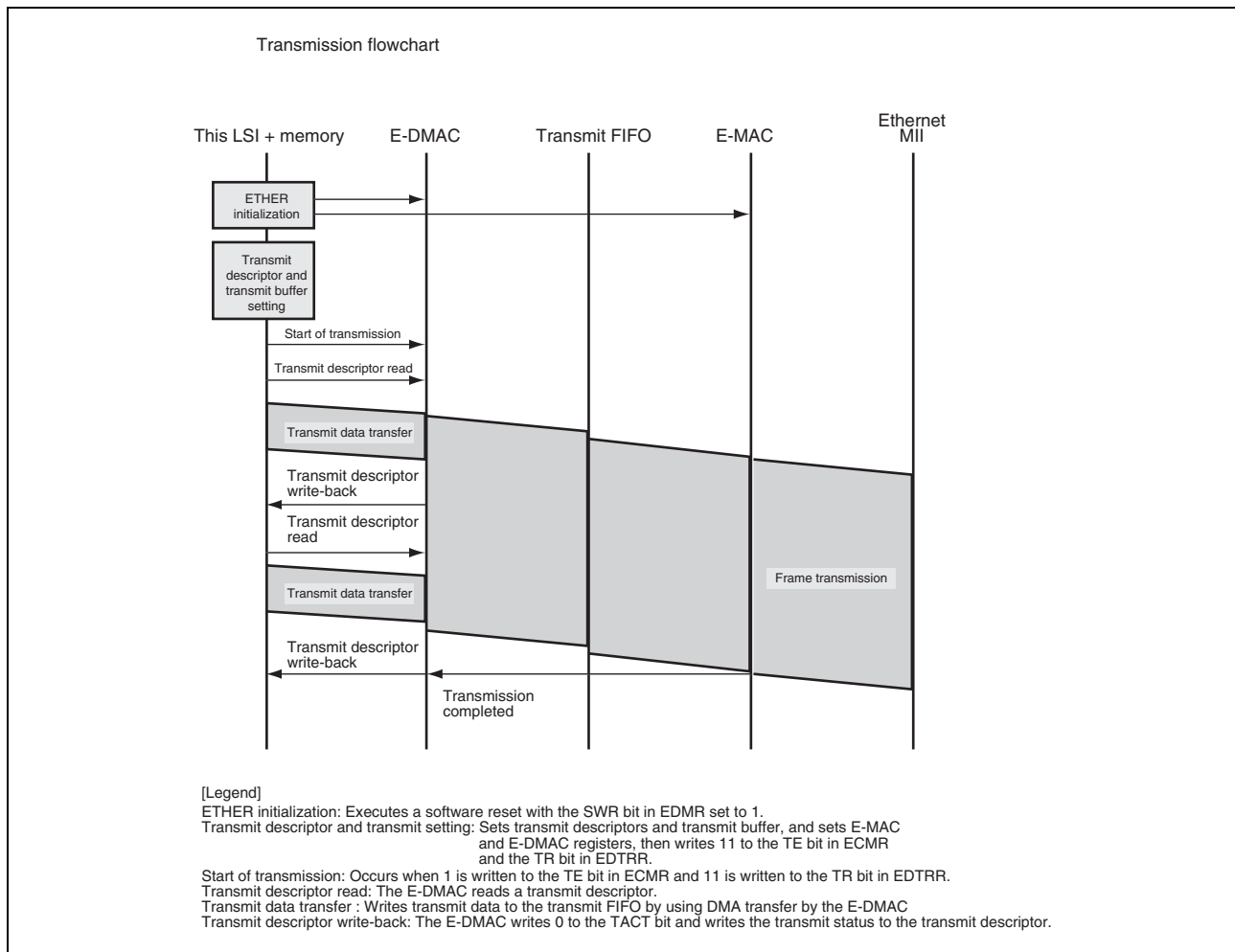
- Clears the TR bits in EDTRR to 00.
- Writes the TC bits in EESR to 11 and generates an interrupt to the CPU.

The E-DMAC can store up to four frames of data in the transmit FIFO.

When the following conditions are satisfied, the E-MAC transmit processing section reads transmit data from the transmit FIFO to configure a frame and transmits the frame to the MII

- The amount of data in the transmit FIFO exceeds the number of bytes specified by TFTR.
- One or more frame of data is stored in the transmit FIFO.
- The transmit FIFO has no space (full of transmit wait data for the MII).

Figure 26.8 shows an example of transmission flow.



**Figure 26.8 Sample Transmission Flowchart (Single-Frame/Two-Description)**

Figure 26.9 shows the status change of the E-MAC transmitter.

1. When the TE bit in EDCMR is set, the transmitter enters the transmit idle state.
2. The preamble is sent as soon as a transmit request is issued by the E-DMAC.
3. The transmitter sends the SFD, data, and CRC sequentially. At the end of transmission, the transmit E-DMAC generates a transmission complete interrupt (TC).
4. After waiting for the frame interval time, the transmitter enters the idle state, and if there is more transmit data, continues transmitting.

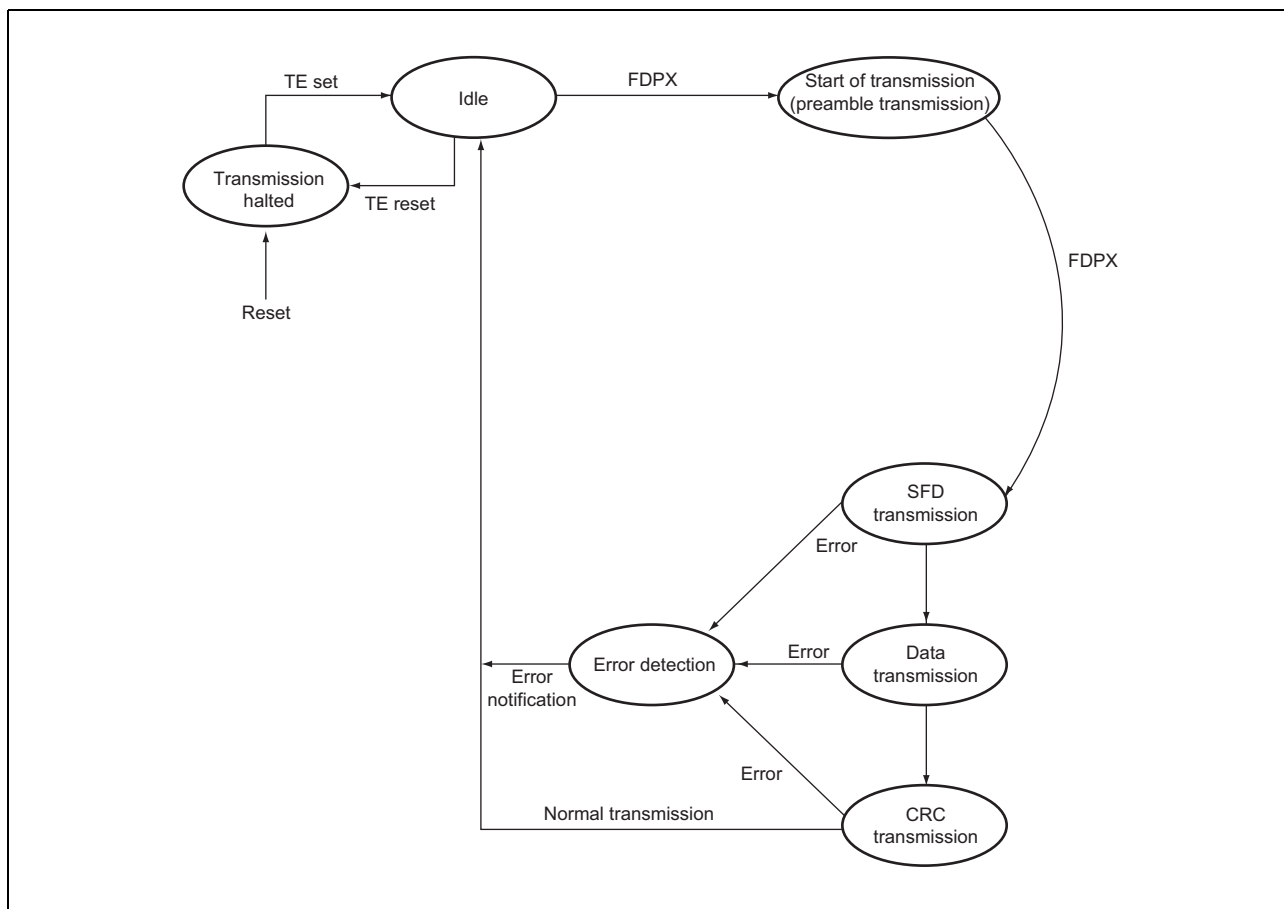


Figure 26.9 E-MAC Transmitter State Transitions

## (2) Transmission Error Processing

### (a) Transmission Abort

If a transmission error is detected during frame transmission from the transmit FIFO to the MII, transmission of the frame data is aborted. At this time, if DMA transfer of the appropriate frame from the transmit buffer to the transmit FIFO has not been completed, the DMA transfer is also aborted.

Following a write-back operation to the transmit descriptor related to the transmit frame aborted by a transmission error, 1 is written to the TABT bit in EESR and an interrupt is issued to the CPU. The subsequent transmit descriptors will be processed normally.

### (b) Transmit FIFO Underflow

If the transmit FIFO is empty (transmit FIFO underflow) during frame transmission from the transmit FIFO to the MII, the E-MAC forcibly aborts transmission of the frame to the MII. At this time, the frame that the E-MAC receives from the E-DMAC is cut off halfway. Then, the E-MAC performs the following operation:

- Writes the TFUF bit in EESR to 1 and generates an interrupt to the CPU.
- Performs a write-back operation to the transmit descriptor corresponding to the transmit frame.
- Following the write-back operation, writes the TUC bit in EESR and generates an interrupt to the CPU.

The subsequent transmit descriptors operate normally.

The E-MAC waits to start frame transmission from the transmit FIFO to the MII until the data that was stored in the transmit FIFO exceeds the number of the bytes specified by TFTR. Through the effective use of TFTR, the transmit FIFO underflow counts can be controlled.

### (c) Transmit Descriptor Empty

When the TFP bits of the descriptor previously processed are set to 00 or 10 and the TACT bit of the read transmit descriptor is set to 0 (invalid), a transmit descriptor empty state is determined and 1 is written to the TDE bit in EESR, and then an interrupt is issued to the CPU.

When a transmit descriptor state is empty, start transmission processing after a software reset.

### 26.4.3 Reception

#### (1) Reception Procedure and Processing Flow

The E-MAC receiver separates the frame from the MII into preamble, SFD, data and CRC, and transfers the fields from DA (destination address) to the data to the receive FIFO. Up to 24 frames can be written in the receive FIFO. Figure 26.10 shows the status change of the E-MAC receiver.

1. When the RE bit in EDCMR is set to 1, the receiver enters the receive idle state.
2. When an SFD (start frame delimiter) is detected after a receive packet preamble, the receiver starts receive processing. A frame with an invalid pattern is discarded.
3. In normal mode, if the destination of the frame address is this LSI, the receiver starts data reception when broadcast or multicast transmission is specified. In promiscuous mode, data reception starts regardless of the frame type.
4. Following data reception from the MII, the receiver carries out a CRC check. The result is indicated as a status bit in the descriptor after the frame data has been written to the receive FIFO. Reports an error status in the case of an abnormality.

After one frame has been received, if the RE bit in EDCMR is set to 1, the receiver prepares to receive the next frame.

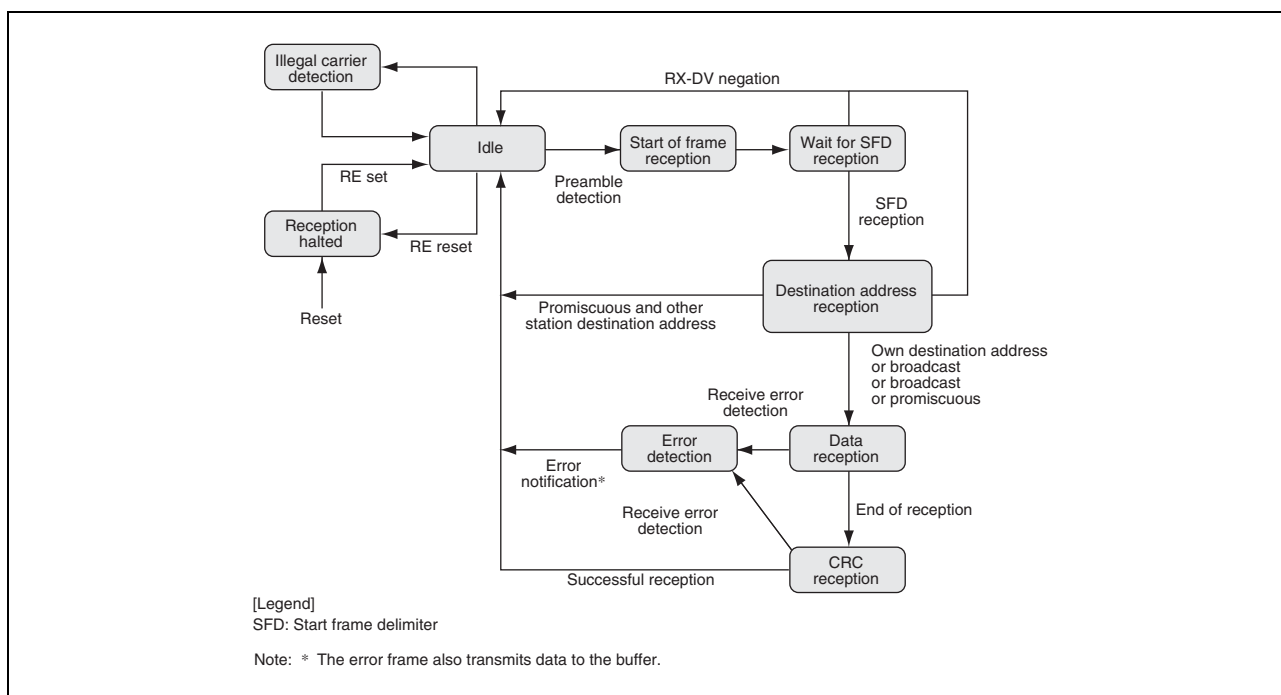


Figure 26.10 E-MAC Receiver State Transitions

CAM evaluation can be referenced during frame processing in reception (for details on the CAM function, refer to section 26.4.4, CAM Function).

When 1 is written to the RR bit in EDCRR while the RE bit in EDCMR is set to 1, the E-DMAC reads the descriptor following the previously used descriptor from the receive descriptor list (or the descriptor indicated by RDLAR at the initial startup) then enters the receive wait state. If 32 bytes or more of data or the last byte of the receive frame is stored in the receive FIFO, the E-DMAC transfers receive FIFO data to the receive buffer specified by RD2 according to the receive descriptor with the RACT bit set to 1 (valid).

If the data length of a received frame is longer than the buffer length specified by RD1, the E-DMAC performs a write-back operation to the descriptor (set RFP to 10 or 00) when the buffer is full, then reads the next descriptor. The E-DMAC then continues to transfer data to the receive buffer specified by the new RD2.

When the following conditions are satisfied, a write-back operation is performed for the descriptor (RFP = 11 or 01), 11 is written to the FR bits in EESR, and an interrupt is issued to the CPU.

- The receive buffer has been full during DMA transfer.
- DMA transfer to the receive buffer of the last byte of the receive frame has been completed.

After the reception processing of the frame, the next descriptor reading standby state begins. At this time, if 32 bytes or more of data or the last byte of the receive frame is stored in the receive FIFO, the next receive descriptor process is performed continuously.

When the TACT bit of the read receive descriptor is 0 (invalid), the receive descriptor empty state is determined and the RDE bit in EESR is written to 1, and then an interrupt is issued to the CPU.

To receive frames continuously, set the RNC bit in RMCR to 1. The initial value is 0.

Figure 26.11 shows an example of reception flow.

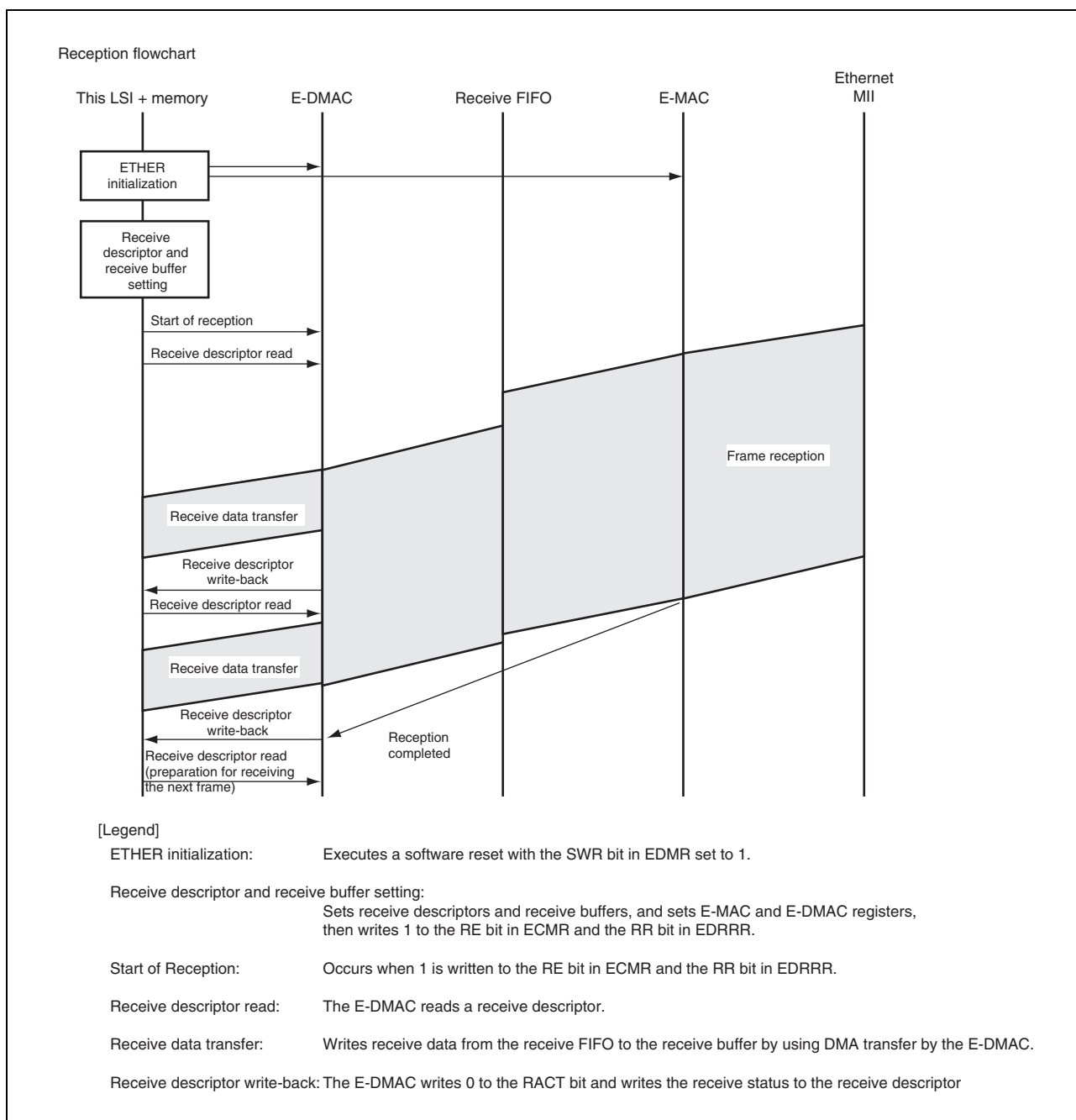


Figure 26.11 Sample Reception Flowchart (Single-Frame/Two-Descriptor)

## (2) Reception Error Processing

### (a) Reception Error

When a reception error occurs, the FR and RABT bits in EESR are set to 1 and an interrupt is issued to the CPU after a write-back operation for the receive descriptor related to the reception error frame.

If a reception error occurs when the length of the frame received from the MII is less than 32 bytes, DMA transfer to the receive buffer for the frame is not performed. At this time, the receive frame is discarded in the E-DMAC (flush function). However, if padding is inserted in the receive frame by RPADIR, the flush function is performed when the frame length including the padding bytes is less than 32 bytes.

### (b) Receive FIFO Overflow

In any of the following cases, the E-MAC cannot receive frames from the MII because it has no space to store receive frames, and all the receive frames that have been transferred to the E-MAC will be discarded in the E-MAC (receive FIFO overflow).

- Receive FIFO is full of data waiting for DMA transfer (the receive FIFO has no space).
- The number of receive frames waiting for DMA transfer is 24 in total (the receive frame information managing area has no empty space; up to 24 frames can be managed).

If an overflow occurs due to the former case, the RFE bit in EESR is set to 1 and an interrupt is generated to the CPU. If an overflow occurs due to the latter case, the RFCOF bit in EESR is set to 1 and an interrupt is generated to the CPU. Each time a receive frame is discarded due to an overflow, RMFCR is incremented. However, RMFCR is not incremented for a receive frame that is cut off due to insufficient receive FIFO space. If a receive frame is cut off due to insufficient receive FIFO space (the frame is partially stored in the receive FIFO), the E-DMAC performs the following operation:

- Performs DMA transfers for the cut-off frame stored in the receive FIFO to the receive buffer.
- After the DMA transfer, performs a write-back operation on the receive descriptor.
- After the write-back operation, sets the ROC bit in EESR to 1 and generates an interrupt to the CPU.

When the receive FIFO is full of data waiting for DMA transfer, frame reception from the MII can be resumed if DMA transfer is performed from the receive FIFO to the receive buffer and 32 bytes or more of empty space is generated in the receive FIFO. When the number of receive frames waiting for DMA transfer is 24 in total, frame reception from the MII can be resumed if one or more frame has been DMA transferred from the receive FIFO to the receive buffer. For restarting frame reception from the MII, when the E-DMAC resumes frame reception from the MII, it only accepts from the start of the frame.

### (c) Flow Control

When the amount of receive data or the number of receive frames in the receive FIFO leads to one of the following conditions, the E-DMAC notifies the E-MAC to control E-MAC writing to the receive FIFO.

- When the space used in the receive FIFO exceeds the data amount specified by FCFTR
- When the number of receive frames in the receive FIFO exceeds the value specified by FCFTR

The threshold of the receive data amount can be set in a range from 256 to 65536 bytes in 256-byte units.

The threshold of receive frames can be set in a range from 1 to 24 frames (by the frame) in frame units.

#### (d) Receive Descriptor Empty

When the RACT bit of the read descriptor is 0 (invalid), the receive descriptor empty state is determined and DMA transfer is stopped. Then the following operation is performed.

- Writes the RR bit in EDRRR to 0
- Sets the RDE bit in EESR to 1 and generates an interrupt to the CPU.

To resume the DMA transfer to the receive buffer, the interrupt source needs to be cleared by software, the receive descriptor needs to be re-set and the RR bit in EDRRR should be set to 1.

Even if receive descriptor is empty, frame reception from the MII to the receive FIFO is continued if there is empty space left in the receive FIFO and receive frame information management area. Therefore, even if a receive descriptor empty state is determined, the DMA transfer can be performed without discarding the frames received from the MII if DMA transfer to the receive buffer can be resumed before an overflow occurs.

### 26.4.4 CAM Function

Frames input to the E-MAC are grouped into the following four types; unicast for this LSI, broadcast, multicast, and unicast to other destinations. The MAC addresses of unicast for this LSI and broadcast are fixed, and determined only by register settings. Consequently, only multicast and unicast to other destinations determine whether to receive or not by using the CAM (unicast frames whose destination MAC addresses match this LSI are called unicast frames to this LSI, and those that do not are called unicast frames to other destinations).

Furthermore, the evaluation of reception of unicast to other destinations and multicast frames by using CAM are performed by referencing the registered MAC addresses of the CAM entry table in the TSU. By using this function, receive FIFO overflow can be prevented caused by accumulation of frame data not required for reception, and CPU processing for determining reception can be reduced.

The on-chip CAM has entry tables which can register the MAC address of 32 entries, the details of which can be set by TSU\_ADRH0 to TSU\_ADRH31 and TSU\_ADRL0 to TSU\_ADRL31. The setting to enable/disable referencing of the on-chip CAM entry table is performed by the CAM entry table enable setting register which sets whether to perform CAM evaluation or not. When on-chip CAM entry table referencing is enabled, the destination address in the frame and MAC address registered in the CAM entry table are compared, and it is determined whether to transfer the frames input to the E-MAC to E-DMAC (have E-DMAC receive the frames) or discard the frames. Table 26.4 shows the processing method of frames (receive or discard) in reception from E-MAC to E-DMAC.

**Table 26.4 Receive Frame Processing**

CAM Entry Table Referencing Results	Types of Frame	Normal Mode		Promiscuous Mode	
		MCT = 0	MCT = 1	MCT = 0	MCT = 1
CAM hit (when addresses match)	Frame to this LSI	Discarded		Discarded	
	Broadcast frame	Discarded		Discarded	
	Multicast frame	Discarded	Received	Discarded	Received
	Frames having destinations other than this LSI	Received		Discarded	
CAM mishit (when addresses do not match)	Frames to this LSI	Received		Received	
	Broadcast frame	Received		Received	
	Multicast frame	Received	Discarded	Received	Discarded
	Frames having destinations other than this LSI	Discarded		Received	

[Legend]

MCT (Bit 13 in ECMR): Multicast receive mode (0: Receive when CAM mishit/1: Receive when CAM hit)



### 26.4.5 Transmit Processing of Multi-Buffer Frame (Single-Frame/Multi-Descriptor)

#### (1) Multi-Buffer Frame Transmit Processing

If an error occurs during multi-buffer frame transmission, the processing shown in Figure 26.12 is carried out by the E-DMAC.

In the figure where the transmit descriptor is shown as inactive (TACT bit = 0), buffer data has already been transmitted successfully, and where the transmit descriptor is shown as active (TACT bit = 1), buffer data has not been transmitted. If a frame transmit error occurs in the first descriptor part where the transmit descriptor is active (TACT bit = 1), transmission is halted, and the TACT bit is cleared to 0, immediately. The next descriptor is then read, and the position within the transmit frame is determined on the basis of bits TFP1 and TFP0 (continuing [B'00] or end [B'01]). In the case of a continuing descriptor, the TACT bit is cleared to 0, and the next descriptor is read immediately. If the descriptor is the final descriptor, not only is the TACT bit cleared to 0, but write-back is also performed to the TFE and TFS bits at the same time. Data in the buffer is not transmitted between the occurrence of an error and write-back to the final descriptor. If error interrupts are enabled in EESIPR, an interrupt is generated immediately after the final descriptor write-back.

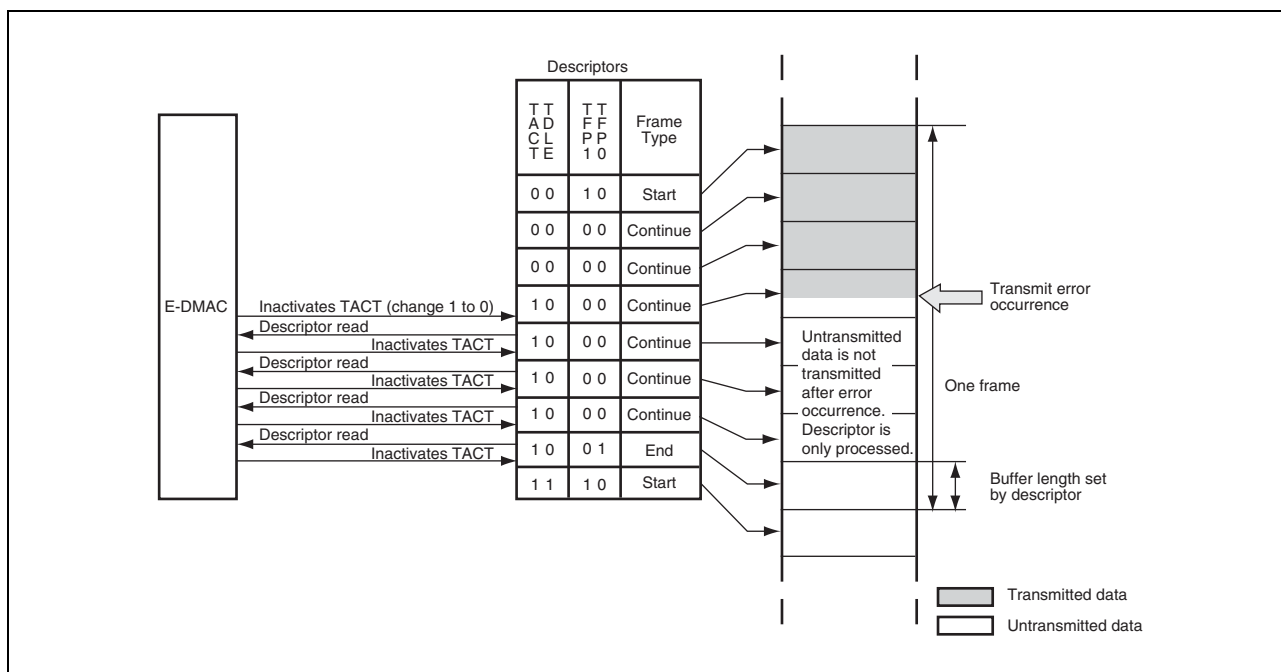


Figure 26.12 E-DMAC Operation after Transmit Error

### 26.4.6 Padding Insertion in Receive Data

In the E-DMAC, one to three bytes of padding can be inserted in any byte position of receive data to improve software handling capability. By using this function, for instance, inserting 2-byte padding after the MAC header (14 bytes) of Ethernet frame enables data following the MAC header to set in 4-byte boundary.

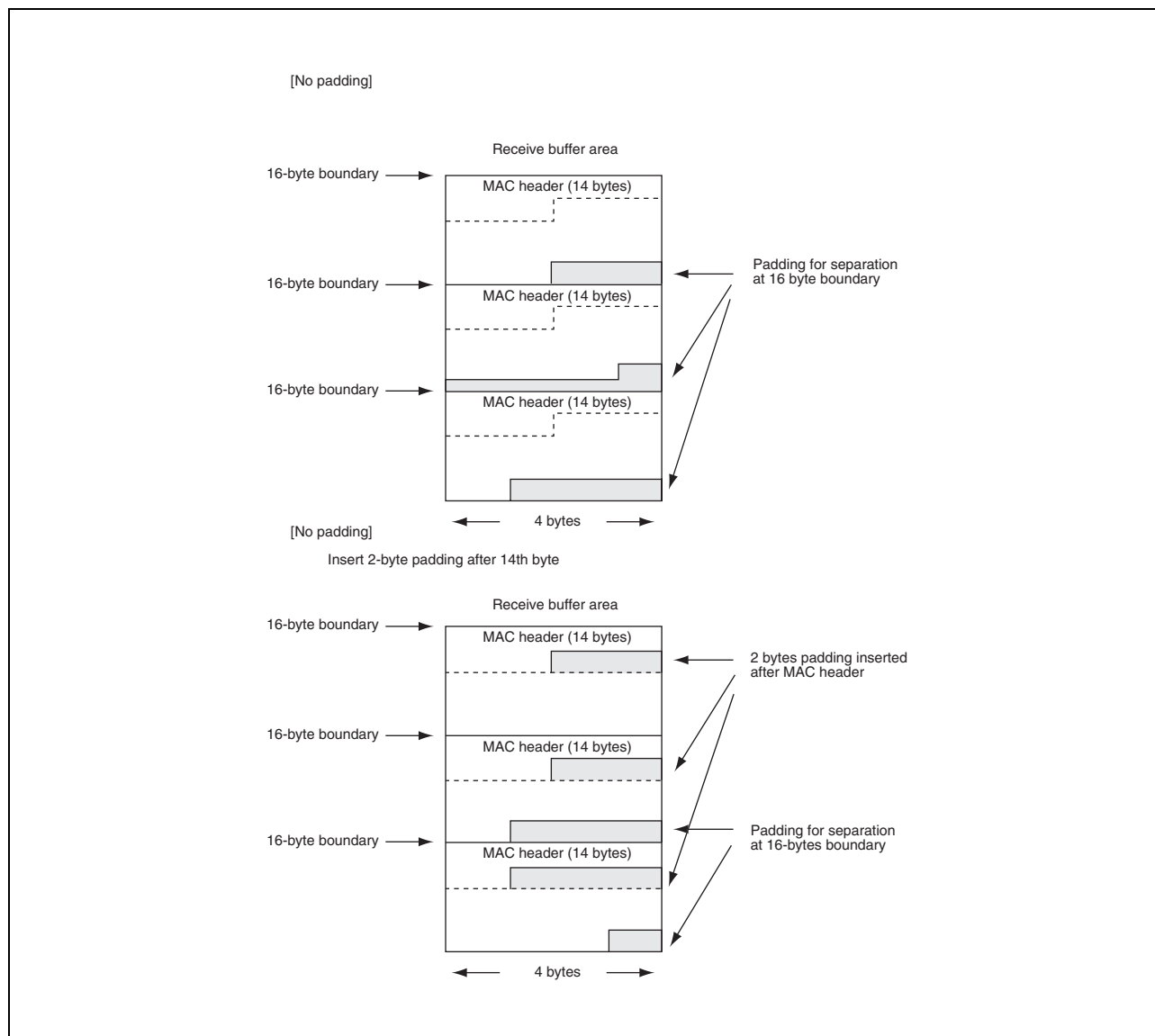


Figure 26.13 Padding Insertion in Receive Data

## 26.4.7 Interrupt Processing

### (1) Interrupt Sources

The ETHER issues one type of interrupt to the CPU: receive/transmit interrupts (ETHERI).

ETHERI interrupts are generated in correspondence with the transmit/receive operation. When an interrupt source is generated, it is set in EESR0 and an interrupt is issued to the CPU. For some interrupt sources, the EESR0 setting and an interrupt to the CPU are performed after a write-back operation to a descriptor is completed, not immediately after the interrupt source is detected. Interrupt sources other than the E-MAC status register source (ECI bit) are cleared by writing a 1 to the corresponding source bit. The E-MAC status register source (ECI bit) is cleared by writing a 1 to the corresponding source bit in ECSR. Interrupt source bits retain the values until they are cleared. ETHERI interrupt source is allowed to issue interrupts by setting the corresponding bit in EESIPR0. Each E-MAC state register source (ECI bit) is allowed to issue an interrupt by setting the corresponding bit in ECSIPR. In the initial value, interrupts are disabled. Table 26.5 shows these three interrupts, interrupt sources, interrupt status registers and bits set at interrupt occurrence and interrupt generation timing.

**Table 26.5 List of ETHER Interrupts**

Interrupt	Interrupt Source	Register and Bit	Interrupt Generation Timing
Transmit/ receive interrupt (ETHERI)	Write-back completed	EESR0.TWB	After write-back
	Transmit underflow frame write-back completed	EESR0.TUC	After write-back
	Receive underflow frame write-back completed	EESR0.ROC	After write-back
	Transmission abort detection	EESR0.TABT	After write-back
	Reception abort detection	EESR0.RABT	After write-back
	Receive frame counter overflow	EESR0.RFCOF	When the interrupt source is detected
	E-MAC status register source	EESR0.ECI	When the interrupt source is detected
	Frame transmission completed	EESR0.TUC	After write-back
	Transmit descriptor empty	EESR0.TDE	When the interrupt source is detected
	Transmit FIFO underflow	EESR0.TFUF	When the interrupt source is detected
	Frame reception	EESR0.FR	After write-back
	Receive descriptor empty	EESR0.RDE	When the interrupt source is detected
	Receive FIFO overflow	EESR0.RFOF	When the interrupt source is detected
Transmit/ receive interrupt (ETHERI)	Receive Multicast Address Frame	EESR0.RMAF	After write-back
	Receive Residual-Bit Frame	EESR0.RRF	After write-back
	Receive Too-Long Frame	EESR0.RTLF	After write-back
	Receive Too-Short Frame	EESR0.RTSF	After write-back
	PHY-LSI Receive Error	EESR0.PRE	After write-back
	CRC Error on Received Frame	EESR0.CERF	After write-back

### 26.4.8 Activation Procedure

The ETHER should be activated by the following procedure:

#### (1) Reset

1. Perform a power-on reset.
2. Set the ET\_TXCLK and ET\_RXCLK pins and supply the clock signal (for details on pin function settings, see section 54, Ports).
3. Start the E-DMAC transmitter and receiver (activation of descriptor engine).  
—Set ENT to 1 and ENR to 1 in EDSR.
4. Perform a software reset.  
—Set SWRR to 1 and SWRT to 1 in EDMR simultaneously.
5. Initialize the descriptor entry table.
6. Confirm cancellation of the software reset.  
—Check that the SWRR and SWRT bits in EDMR are cleared to 0.

#### (2) Pin Settings

See section 54, Ports.

#### (3) Registration of Descriptor Ring

The address of a descriptor ring configured in memory is registered in the descriptor entry table.

1. Transmit descriptor setting  
—Set TDLAR.  
—Set TDFAR.  
—Set TDFXR.  
—Set TDFFR. When the descriptor indicated by TDFXR is the last descriptor in the descriptor list, set H'00000001.
2. Receive descriptor setting  
—Set RDLAR.  
—Set RDFAR.  
—Set RDFXR.  
—Set RDFFR. When the descriptor indicated by RDFXR is the last descriptor in the descriptor list, set H'00000001.

#### (4) Register Settings

The following registers should be set as necessary.

1. E-DMAC related registers  
—Set EDMR: Operating mode, etc.  
—Set EESIPR: Interrupt masks  
—Set TRSCER: Error masks  
—Set TFTR: Transmit FIFO threshold  
—Set FDR: External FIFO size  
—Set RMCR: Reset method for reception activation  
—Set RPADIR: Padding insertion into receive data  
—Set FCFTR: Receive BSY output threshold
2. E-MAC related registers  
—Set ECMR setting: Transmission/reception specifications  
—Set ECSIPR setting: Interrupt masks  
—Set MAHR: MAC address  
—Set MALR: MAC address

- Set RFLR: Maximum receive frame length
- Set APR: TIME parameter value of an automatic pause frame
- Set MPR: TIME parameter value of a manual PAUSE frame
- Set TPAUSER: Upper limit of automatic PAUSE frame retransmission

## (5) Activation

1. Start the E-DMAC transmission/reception function
  - Set the TR bits in EDTRR to 11.
  - Set the RR bit in EDRRR to 1.
2. Start the E-MAC transmission/reception function
  - Set the TE and RE bits in ECMR to 1.

## 26.4.9 Flow Control

The ETHER supports flow control functions conforming to IEEE802.3x for full-duplex operation. The flow control can be applied to both receive and transmit operations. When transmitting PAUSE frames, flow control can be performed by the following two procedures:

### (1) Automatic PAUSE Frame Transmission

For receive frames, PAUSE frames are automatically transmitted when the number of data written to the receive FIFO reaches the value set in FCFTR. The TIME parameter included in the PAUSE frame is set by APR. The automatic PAUSE frame transmission is repeated until the number of data in the receive FIFO becomes less than the value set in FCFTR as the receive data is read from the FIFO. Using TPAUSER, the upper limit of retransmission counts of the PAUSE frames can also be set in the range from 1 to 65535. In this case, PAUSE frame transmission is repeated until the number of receive FIFO data becomes less than the FCFTR value, or the number of transmits reaches the value set by TPAUSER. The transmission counter is cleared to 0 when the next PAUSE frame is transmitted after the number of data in the receive FIFO becomes less than the FCFTR value.

The automatic PAUSE frame transmission is enabled when the TXF bit in ECMR is 1.

### (2) Manual PAUSE Frame Transmission

PAUSE frames are transmitted by directives from the software. When writing the Timer value to MPR, manual PAUSE frame transmission is started. With this method, PAUSE frame transmission is carried out only once.

### (3) PAUSE Frame Reception

The next frame is not transmitted until the time indicated by the Timer value elapses after receiving a PAUSE frame. However, the transmission of the current frame is continued. A received PAUSE frame is valid only when the RXF bit in ECMR is set to 1. The number of times of PAUSE frame receptions is counted.

### (4) 0-Time PAUSE Frame Control

Flow control is performed using a PAUSE frame with the TIME parameter value set to 0. The PAUSE frame with the TIME parameter set to 0 can be enabled or disabled by the ZPF bit in ECMR.

- When PAUSE frame control with the TIME parameter value set to 0 is enabled
 

A PAUSE frame with the TIME parameter value set to 0 is transmitted when the number of data in the receive FIFO is less than the FCFTR value before the time indicated by the TIME parameter value has not elapsed. When a PAUSE frame with the time indicated by the TIME parameter value set to 0 is received, the transmit standby state is canceled.
- When PAUSE frame control with the TIME parameter value set to 0 is disabled
 

A PAUSE frame with the TIME parameter value set to 0 is not transmitted. When a PAUSE frame with the TIME parameter value set to 0 is received, the PAUSE frame is discarded.

## 26.4.10 Intelligent Checksum Calculation Function

This function accelerates checksum calculation on received packets, and provides the following two modes.

- MAC/IP packet analyzing intelligent checksum calculation mode
- All-data intelligent checksum calculation mode with bytes to be skipped specified

### (1) MAC/IP Packet Analyzing Intelligent Checksum Calculation Mode (CSEBL = 1 and CSMD = 1 in CSMR)

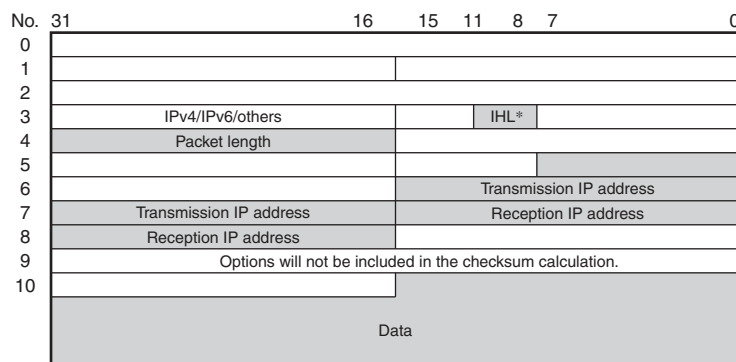
In this mode, the checksum of received packets indicated in the table is calculated. However, if a MAC packet payload includes padding data in fields other than those for the IP packet itself because there is too little data for a full packet, it is not included in the checksum.

IPver	Items
IPv4	Option present
	Option not present
	Fragment* <sup>1</sup>
IPv6	Extension header not present
	Hop-by-hop options extension header length
	Routing extension header length
	Fragment extension header length* <sup>1</sup>
	Destination options header length
	AH extension header length
	ESP extension header length* <sup>2</sup>
	Extension header length for mobile IPv6* <sup>2</sup>

Note 1. This packet is to be checksummed, however, the RCS[15:0] bits and RCSE bit in RD0 are to be undefined even if the data is successfully received.

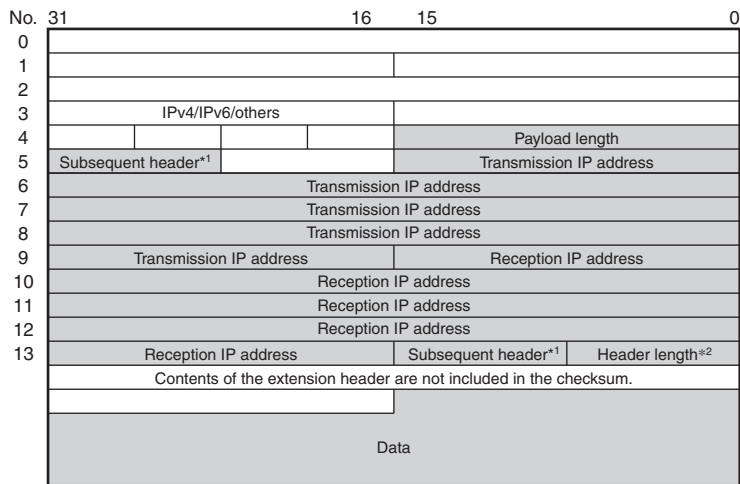
Note 2. The RD0.RCSE bit is set to 1 without calculating the value of the RD0.RCS[15:0] bits.

The shaded regions of the following figure indicate the parts of an IPv4 packet which are used to obtain the checksum.



Note: \* After conversion to octet units, this is subtracted in the checksum calculation.  
In the calculation: {8'h00, protocol no.[7:0]}

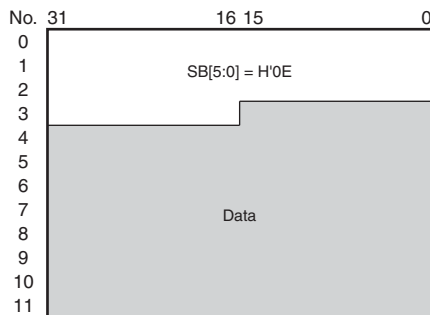
The shaded regions of the following figure indicate the parts of an IPv6 packet which are used to obtain the checksum.



Notes: \*1 Only included in the checksum when the header is for the TCP or UDP. Calculation coverage is extended to {8'h00, protocol No.[7:0]} when the checksum is taken.  
 \*2 After conversion to octet units, this is subtracted in the checksum calculation.

(2) All-Data Intelligent Checksum Calculation Mode with Bytes to be Skipped Specified (CSELB = 1 and CSMD = 0 in CSMR)

After having skipped the number of bytes specified in the SB[5:0] bits in CSMR, counting from the beginning of the packet, the checksum is calculated for all subsequent data (e.g. 14 bytes may be skipped).



## 26.5 Connection to PHY-LSI

### 26.5.1 MII Frame Transmission/Reception Timing

Each MII frame transmission/reception timing is shown in Figure 26.14 to Figure 26.17.

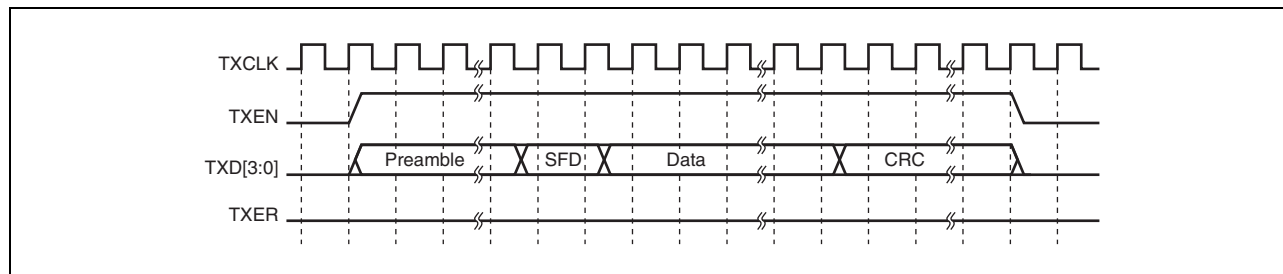


Figure 26.14 MII Frame Transmit Timing (Normal Transmission)

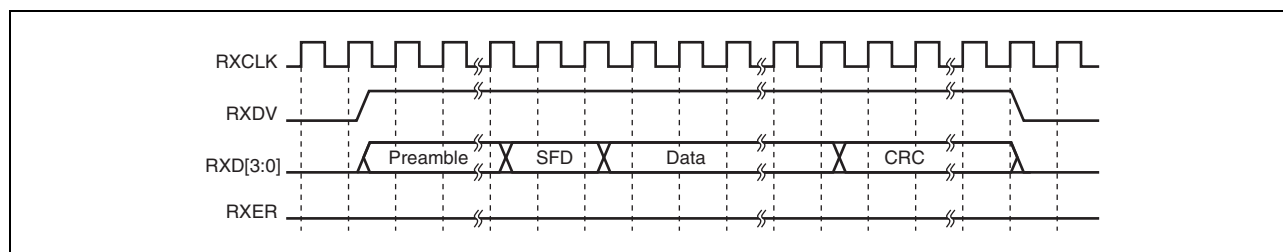


Figure 26.15 MII Frame Receive Timing (Normal Reception)

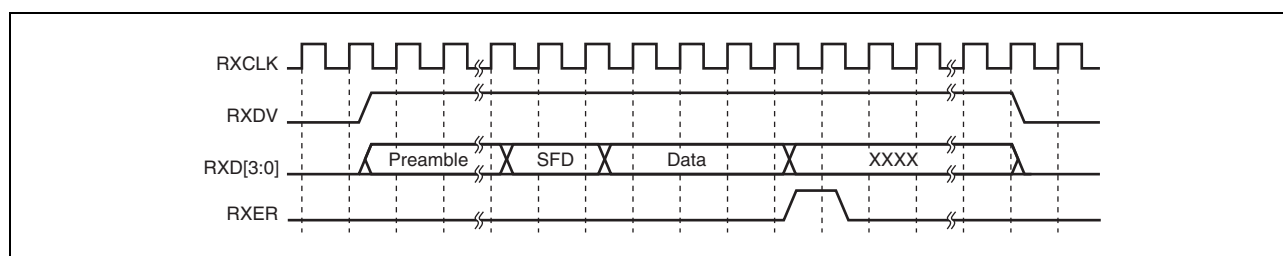


Figure 26.16 MII Frame Receive Timing (Reception Error (1))

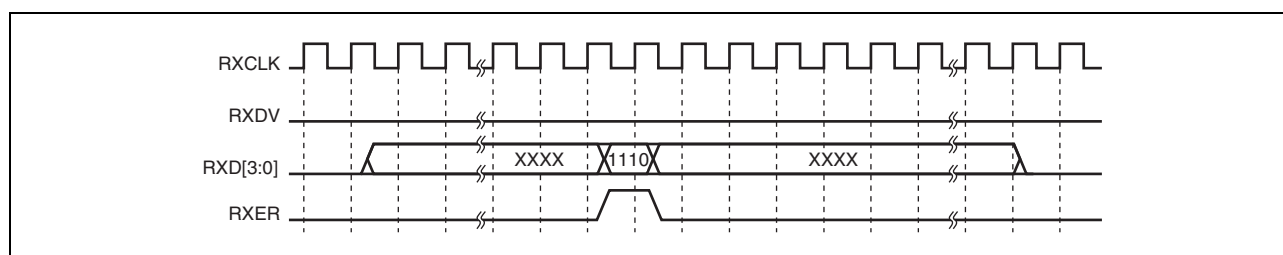


Figure 26.17 MII Frame Receive Timing (Reception Error (2))



## 26.5.2 Accessing MII Registers

MII registers in the PHY-LSI are accessed via PIR in this LSI. PIR is used as a serial interface conforming to the MII frame format specified in IEEE802.3u.

### (1) MII Management Frame Format

Figure 26.18 shows the format of an MII management frame. To access an MII register, a management frame is implemented by the program in accordance with the procedures shown in MII Register Access Procedure.

Access Type	MII Management Frame							
Item	PRE	ST	OP	PHYAD	REGAD	TA	DATA	IDLE
Number of bits	32	2	2	5	5	2	16	
Read	1..1	01	10	00001	RRRRR	Z0	D..D	
Write	1..1	01	01	00001	RRRRR	10	D..D	X

[Legend]

PRE: 32 consecutive 1s  
 ST: Write of 01 indicating start of frame  
 OP: Write of code indicating access type  
 PHYAD: Write of 0001 if the PHY-LSI address is 1 (sequential write starting with the MSB).  
 This bit changes depending on the PHY-LSI address.  
 REGAD: Write of 000q if the register address is 1 (sequential write starting with the MSB).  
 This bit changes depending on the PHY-LSI register address.  
 TA: Time for switching data transmission source on MII interface  
 (a) Write: 10 written  
 (b) Read: Bus release (notation: Z0) performed  
 DATA: 16-bit data. Sequential write or read from MSB  
 (a) Write: 16-bit data write  
 (b) Read: 16-bit data read  
 IDLE: Wait time until next MII management format input  
 (a) Write: Independent bus release (notation: X) performed  
 (d) Read: Bus already released in TA: control unnecessary

**Figure 26.18 MII Management Frame Format**

(2) MII Register Access Procedure

The program accesses MII registers via PIR. Access is implemented by a combination of 1-bit-unit data write, 1-bit-unit data read, bus release, and independent bus release. Figure 26.19 (1) to Figure 26.19 (4) show the MII register access timing. The timing will differ depending on the PHY-LSI type.

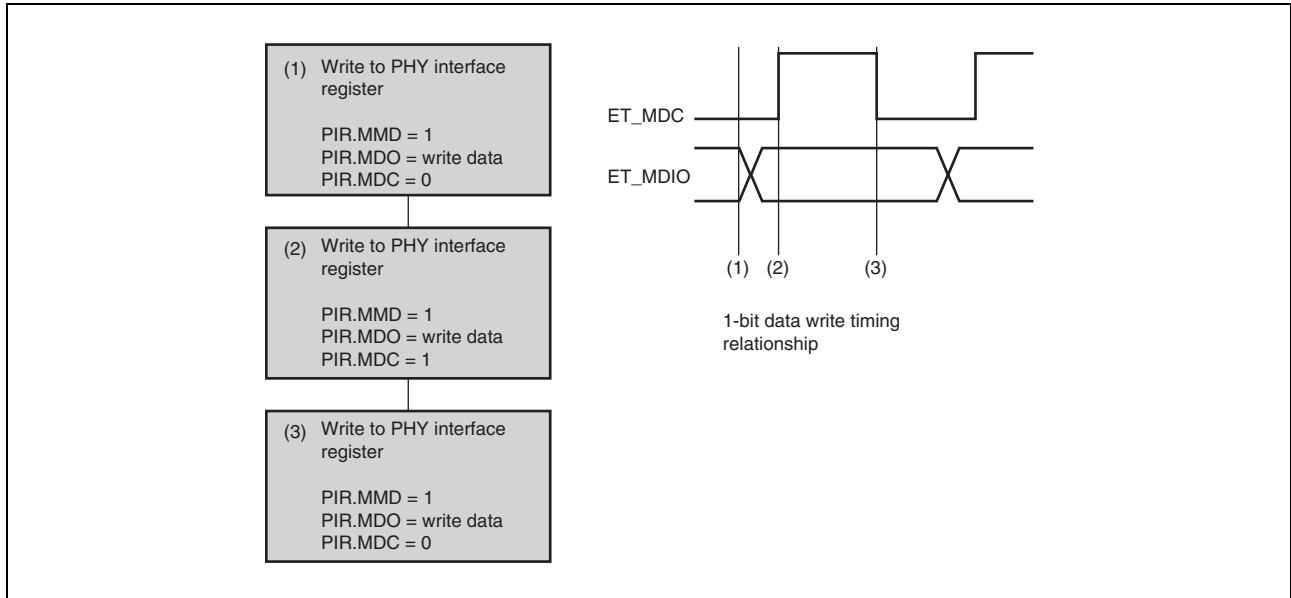


Figure 26.19 (1) 1-Bit Data Write Flowchart

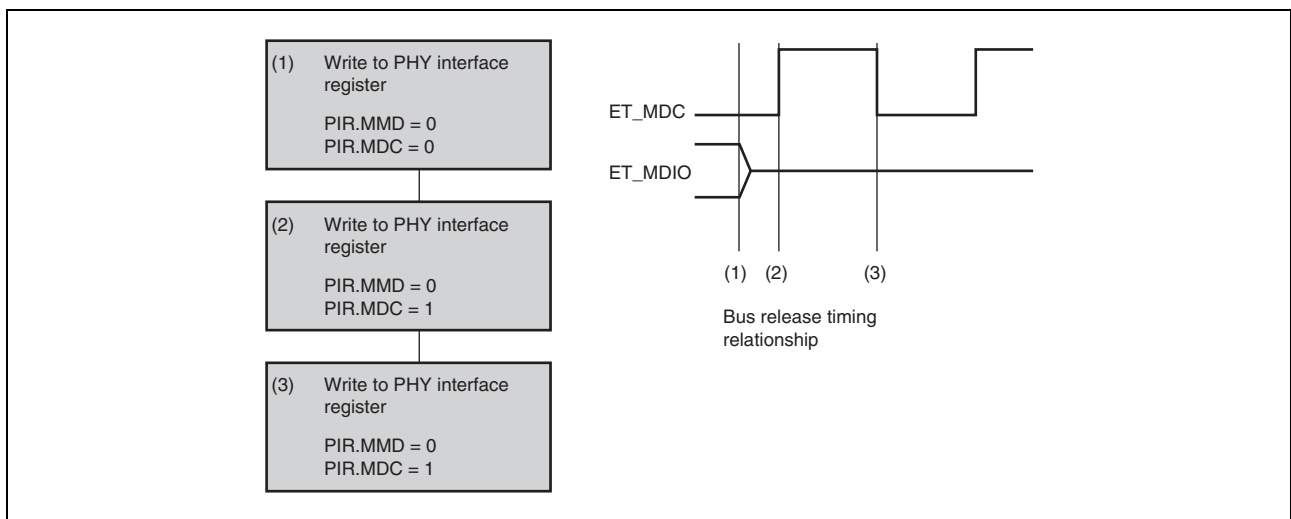


Figure 26.19 (2) Bus Release Flowchart (TA in Read in Figure 26.18)

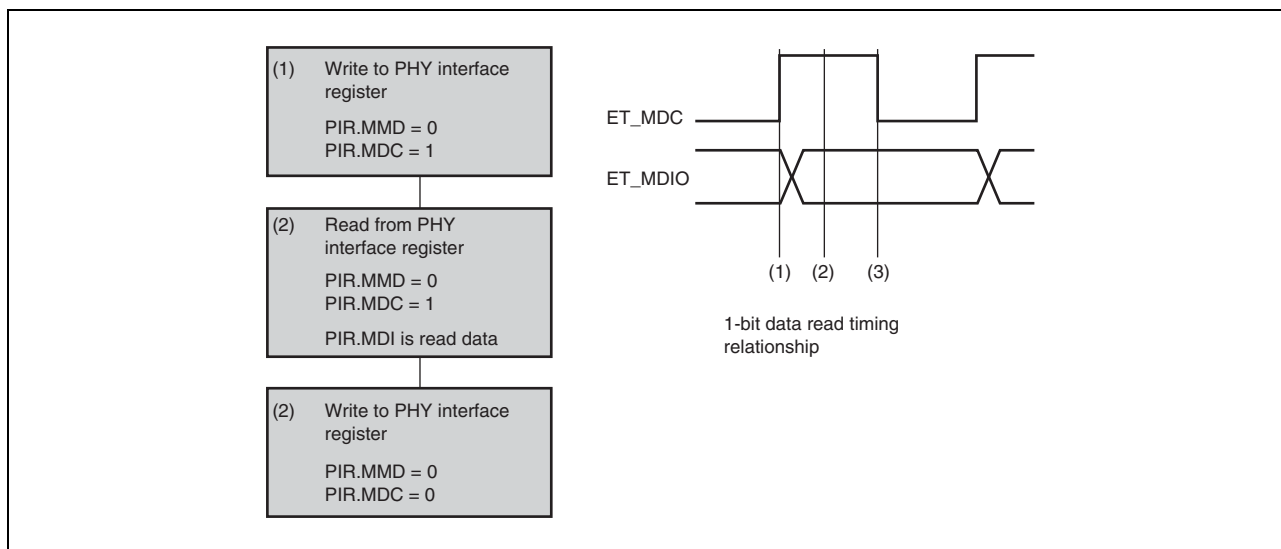


Figure 26.19 (3) 1-Bit Data Read Flowchart

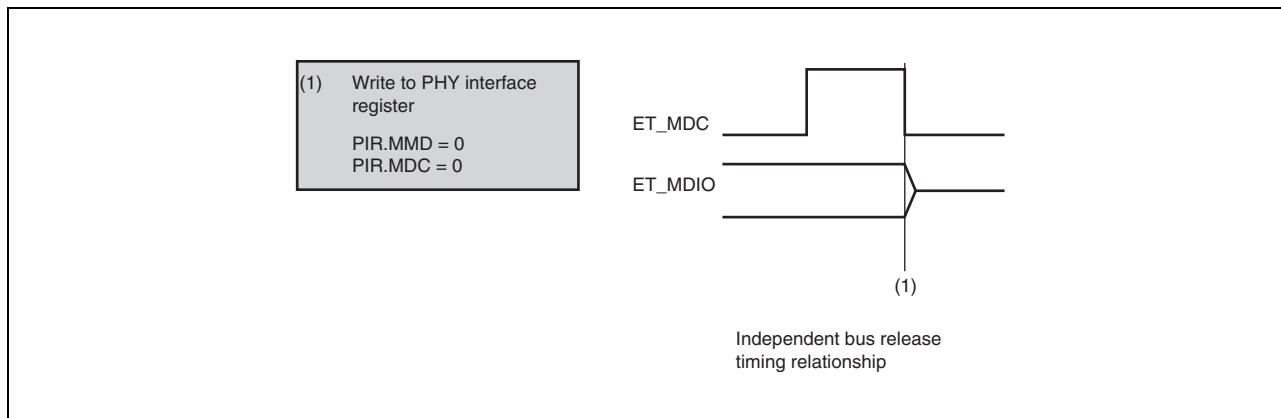


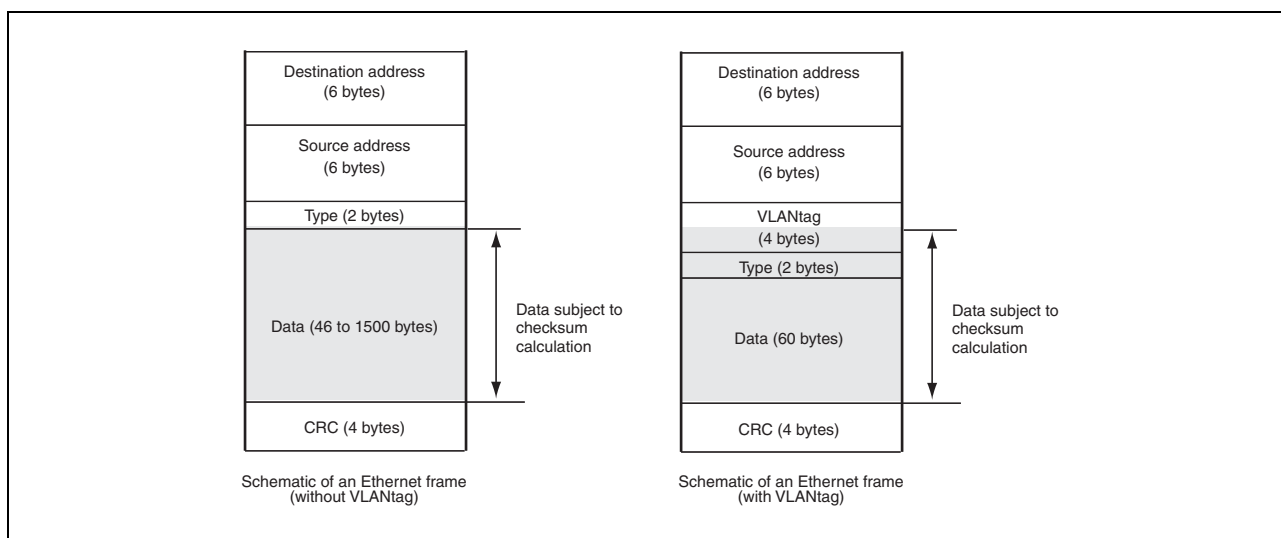
Figure 26.19 (4) Independent Bus Release Flowchart (IDLE in Write in Figure 26.18)

## 26.6 Usage Notes

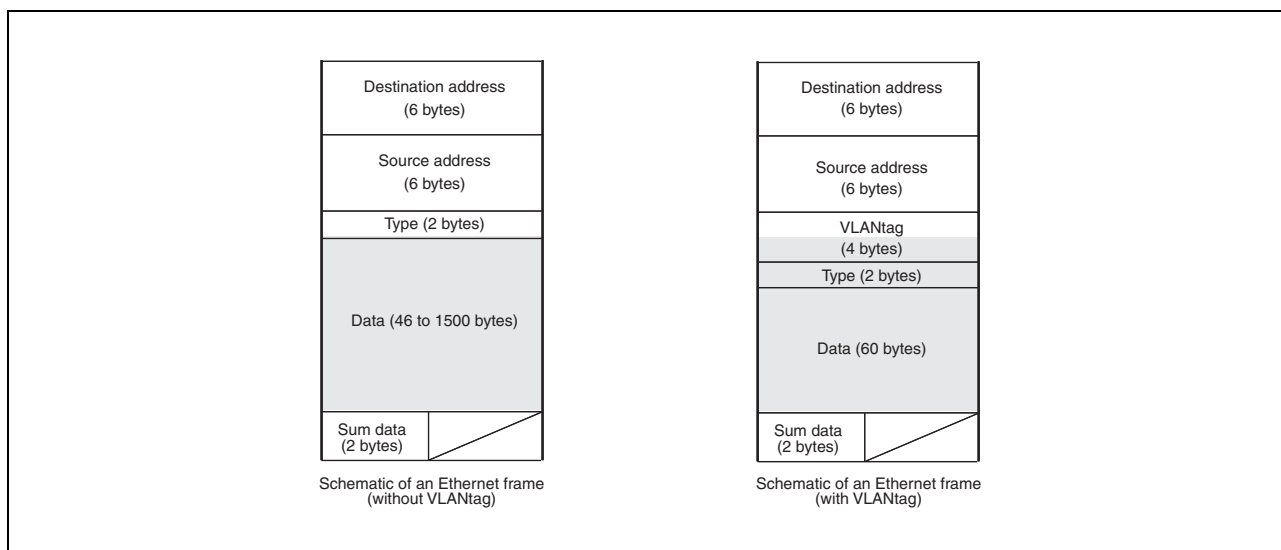
### 26.6.1 Checksum Calculation of Ethernet Frames

This LSI is capable of calculating the checksum data of the received frames. Only the data fields of the Ethernet frames are subject to checksum calculation. Specifically, a data field follows the length/type field and is followed by the CRC field. Figure 26.20 shows schematics indicating which parts of the Ethernet frames are calculated. Calculation involves 16-bit addition only; it does not involve bit inversion. Note that when the checksum data is valid, the CRC data (4 bytes) is not transferred as a receive frame, and the checksum data (sum data) is added automatically. Figure 26.21 shows schematics of Ethernet frames to which the checksum data has been added.

Note: • Also for the frames with VLANtag inserted, the 15th byte from the top and the following bytes before the CRC field are subject to calculation.



**Figure 26.20 Data Subject to Checksum Calculation**



**Figure 26.21 Data after Checksum Data Addition**

### 26.6.2 Notes on Using the Intelligent Checksum Function

Checksum calculation using the intelligent checksum function is not affected by padding insertion specified by the receive data padding insert register (RPADIR). This is because checksum calculation is performed when transferring receive data from E-MAC to E-DMAC, while padding of receive data is performed when transferring receive data from E-DMAC to the receive buffer in memory.

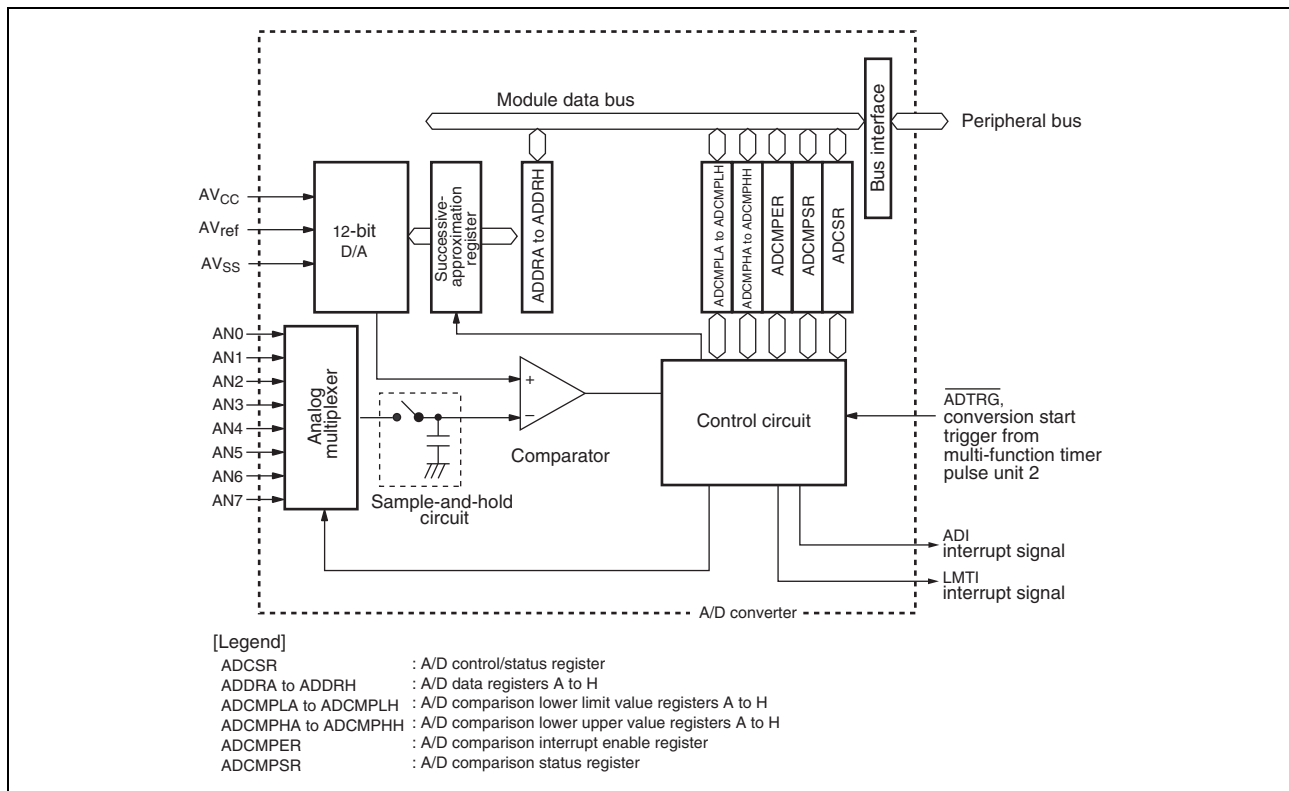
## 27. A/D Converter

This LSI includes a 12-bit successive-approximation A/D converter allowing selection of up to eight analog input channels.

### 27.1 Features

- Resolution: 12 bits
- Input channels: Eight channels
- Minimum conversion time: 5.0  $\mu$ s per channel
- Absolute accuracy:  $\pm 11$  LSB
- Operating modes: Three
  - Single mode: A/D conversion on one channel
  - Multi mode: A/D conversion on one to four channels or on one to eight channels
  - Scan mode: Continuous A/D conversion on one to four channels or on one to eight channels
- Data registers: Eight  
Conversion results are held in a 16-bit data register for each channel
- Sample-and-hold function
- A/D conversion start methods: Three methods
  - Software
  - Conversion start trigger from the multi-function timer pulse unit 2
  - External trigger signal
- Interrupt source: Two sources  
An A/D conversion end interrupt (ADI) request can be generated on completion of A/D conversion.  
An over-limit interrupt (LMTI) request can be generated when a conversion result exceeds the upper or lower limit value for each channel.\*1  
Note 1. This is only available when 10-bit precision is in use.
- Module standby mode can be set

Figure 27.1 shows a block diagram of the A/D converter.



**Figure 27.1** Block Diagram of A/D Converter

## 27.2 Input/Output Pins

Table 27.1 shows the A/D converter pins.

**Table 27.1 Pin Configuration**

Pin Name	Symbol	I/O	Function
Analog power supply pin	AVcc	Input	Analog power supply pin
Analog ground pin	AVss	Input	Analog ground pin and A/D conversion reference ground
Analog reference voltage pin	AVref	Input	A/D converter reference voltage pin
Analog input pin 0	AN0	Input	Analog input
Analog input pin 1	AN1	Input	
Analog input pin 2	AN2	Input	
Analog input pin 3	AN3	Input	
Analog input pin 4	AN4	Input	
Analog input pin 5	AN5	Input	
Analog input pin 6	AN6	Input	
Analog input pin 7	AN7	Input	
A/D external trigger input pin	ADTRG	Input	External trigger input to start A/D conversion



## 27.3 Register Descriptions

The A/D converter has the following registers.

**Table 27.2 Register Configuration**

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
A/D data register A	ADDRA	R	H'0000	H'E8005800	16
A/D data register B	ADDRB	R	H'0000	H'E8005802	16
A/D data register C	ADDRC	R	H'0000	H'E8005804	16
A/D data register D	ADDRD	R	H'0000	H'E8005806	16
A/D data register E	ADDRE	R	H'0000	H'E8005808	16
A/D data register F	ADDRF	R	H'0000	H'E800580A	16
A/D data register G	ADDRG	R	H'0000	H'E800580C	16
A/D data register H	ADDRH	R	H'0000	H'E800580E	16
A/D comparison upper limit value register A	ADCMPHA	R/W	H'0000	H'E8005820	16
A/D comparison lower limit value register A	ADCMPLA	R/W	H'0000	H'E8005822	16
A/D comparison upper limit value register B	ADCMPHB	R/W	H'0000	H'E8005824	16
A/D comparison lower limit value register B	ADCMPLB	R/W	H'0000	H'E8005826	16
A/D comparison upper limit value register C	ADCMPHC	R/W	H'0000	H'E8005828	16
A/D comparison lower limit value register C	ADCMPLC	R/W	H'0000	H'E800582A	16
A/D comparison upper limit value register D	ADCMPHD	R/W	H'0000	H'E800582C	16
A/D comparison lower limit value register D	ADCMPLD	R/W	H'0000	H'E800582E	16
A/D comparison upper limit value register E	ADCMPHE	R/W	H'0000	H'E8005830	16
A/D comparison lower limit value register E	ADCMPLE	R/W	H'0000	H'E8005832	16
A/D comparison upper limit value register F	ADCMPHF	R/W	H'0000	H'E8005834	16
A/D comparison lower limit value register F	ADCMPLF	R/W	H'0000	H'E8005836	16
A/D comparison upper limit value register G	ADCMPHG	R/W	H'0000	H'E8005838	16
A/D comparison lower limit value register G	ADCMPLG	R/W	H'0000	H'E800583A	16
A/D comparison upper limit value register H	ADCMPHH	R/W	H'0000	H'E800583C	16
A/D comparison lower limit value register H	ADCMPLH	R/W	H'0000	H'E800583E	16
A/D control/status register	ADCSR	R/W	H'0000	H'E8005860	16
A/D comparison interrupt enable register	ADCMPER	R/W	H'0000	H'E8005862	16
A/D comparison status register	ADCMPSR	R/W	H'0000	H'E8005864	16

### 27.3.1 A/D Data Registers A to H (ADDRA to ADDRH)

The eight A/D data registers, ADDRA to ADDRH, are 16-bit read-only registers that store the results of A/D conversion. An A/D conversion produces 12-bit data, which is transferred for storage into bits 15 to 4 of the ADDR corresponding to the selected channel. Bits 3 to 0 of ADDR are reserved bits that are always read as 0.

Access to ADDR in 8-bit units is prohibited. ADDR must always be accessed in 16-bit units.

Table 27.3 indicates the correspondence between analog input channels and ADDR.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 4		All 0	R	Bit Data (12bits)
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

**Table 27.3 Correspondence between Analog Input Channels and ADDR**

Analog Input Channel	A/D Data Register where Conversion Result is Stored
AN0	ADDRA
AN1	ADDRB
AN2	ADDRC
AN3	ADDRD
AN4	ADDRE
AN5	ADDRF
AN6	ADDRG
AN7	ADDRH

### 27.3.2 A/D Comparison Upper Limit Value Registers A to H (ADCMPHA to ADCMPHH)

ADCMPH is a 16-bit readable/writable register that holds the upper limit value to be compared with an A/D conversion result. Eight registers, ADCMPHA to ADCMPHH, are provided.

On completion of A/D conversion on each channel, the A/D conversion result is compared with the data stored in the corresponding upper limit value register. When the result is greater than the upper limit value, the corresponding bit in the A/D comparison status register (ADCMPSTR) is set. Comparison with the upper limit value is performed only when the corresponding bit in the A/D comparison interrupt enable register (ADCMPER) is set.

Access to ADCMPH in 8-bit units is prohibited. ADCMPH must always be accessed in 16-bit units.

Table 27.4 indicates the correspondence between the ADDR registers and the ADCMPH/ADCMPL registers.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
											-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 6		All 0	R/W	Comparison Upper Limit Value (10 bits)
5 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

### 27.3.3 A/D Comparison Lower Limit Value Registers A to H (ADCMPLA to ADCMPLH)

ADCMPL is a 16-bit readable/writable register that holds the lower limit value to be compared with an A/D conversion result. Eight registers, ADCMPLA to ADCMPLH, are provided.

On completion of A/D conversion on each channel, the A/D conversion result is compared with the data stored in the corresponding lower limit value register. When the result is smaller than the lower limit value, the corresponding bit in the A/D comparison status register (ADCMPSR) is set. Comparison with the lower limit value is performed only when the corresponding bit in the A/D comparison interrupt enable register (ADCMPIER) is set.

Access to ADCMPL in 8-bit units is prohibited. ADCMPL must always be accessed in 16-bit units.

Table 27.4 indicates the correspondence between the ADDR registers and the ADCMPH/ADCMPL registers.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 6		All 0	R/W	Comparison Lower Limit Value (10 bits)
5 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

**Table 27.4 Correspondence between ADDR and ADCMPH/ADCMPL Registers**

ADDR where Conversion Result is Stored	ADCMPH/ADCMPL (Target for Comparison)
ADDRA	ADCMPHA/ADCMPLA
ADDRB	ADCMPHB/ADCMPLB
ADDRC	ADCMPHC/ADCMPLC
ADDRD	ADCMPHD/ADCMPLD
ADDRE	ADCMPHE/ADCMPLE
ADDRF	ADCMPHF/ADCMPLF
ADDRG	ADCMPHG/ADCMPLG
ADDRH	ADCMPHH/ADCMPLH

### 27.3.4 A/D Control/Status Register (ADCSR)

ADCSR is a 16-bit readable/writable register that selects the operating mode, controls the A/D converter, and enables or disables starting of A/D conversion by external trigger input.

Bit: 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADF		ADIE	ADST	TRGS[3:0]				CKS[2:0]		MDS[2:0]			CH[2:0]		
Initial value: 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0															
R/W:R(W)*1		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: \*1 Only 0 can be written to clear the flag after 1 is read.

Bit	Bit Name	Initial Value	R/W	Description
15	ADF	0	R(W)*1	A/D End Flag Status flag indicating the end of A/D conversion. [Clearing conditions] <ul style="list-style-type: none"> <li>Cleared by reading ADF while ADF = 1, then writing 0 to ADF</li> <li>Cleared when the direct memory access controller is activated by ADI interrupt and ADDR is read</li> </ul> [Setting conditions] <ul style="list-style-type: none"> <li>A/D conversion ends in single mode</li> <li>A/D conversion ends for all the selected channels in multi mode</li> <li>A/D conversion ends for all the selected channels in scan mode</li> </ul>
14	ADIE	0	R/W	A/D Interrupt Enable Enables or disables the interrupt (ADI) requested at the end of A/D conversion. Set the ADIE bit while A/D conversion is not being performed. 0: A/D conversion end interrupt (ADI) request is disabled 1: A/D conversion end interrupt (ADI) request is enabled
13	ADST	0	R/W*2	A/D Start Starts or stops A/D conversion. This bit remains set to 1 during A/D conversion. 0: A/D conversion is stopped 1: Single mode: A/D conversion starts. This bit is automatically cleared to 0 when A/D conversion ends on the selected channel. Multi mode: A/D conversion starts. This bit is automatically cleared to 0 when A/D conversion is completed cycling through the selected channels. Scan mode: A/D conversion starts. A/D conversion is continuously performed until this bit is cleared to 0 by software, by a power-on reset as well as by a transition to deep standby mode.
12 to 9	TRGS[3:0]	0000	R/W	Timer Trigger Select These bits enable or disable starting of A/D conversion by a trigger signal. 0000: Start of A/D conversion by external trigger input is disabled 0001: A/D conversion is started by conversion trigger TRGAN from the multi-function timer pulse unit 2 0010: A/D conversion is started by conversion trigger TRG0N from the multi-function timer pulse unit 2 0011: A/D conversion is started by conversion trigger TRG4AN from the multi-function timer pulse unit 2 0100: A/D conversion is started by conversion trigger TRG4BN from the multi-function timer pulse unit 2 1001: A/D conversion is started by ADTRG Other than above: Setting prohibited
8 to 6	CKS[2:0]	000	R/W	Clock Select These bits select the A/D conversion time.*3 Set the A/D conversion time while A/D conversion is halted (ADST = 0). 000: Conversion time = 256 t <sub>cyc</sub> *4 (maximum) 001: Conversion time = 298 t <sub>cyc</sub> *4 (maximum) 010: Conversion time = 340 t <sub>cyc</sub> *4 (maximum) 011: Conversion time = 382 t <sub>cyc</sub> *4 (maximum) 100, 101, 110, 111: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description																											
5 to 3	MDS[2:0]	000	R/W	Multi-scan Mode These bits select the operating mode for A/D conversion. 0xx: Single mode 100: Multi mode: A/D conversion on 1 to 4 channels 101: Multi mode: A/D conversion on 1 to 8 channels 110: Scan mode: A/D conversion on 1 to 4 channels 111: Scan mode: A/D conversion on 1 to 8 channels																											
2 to 0	CH[2:0]	000	R/W	Channel Select These bits and the MDS bits in ADCSR select the analog input channels.																											
				<table border="1"> <thead> <tr> <th>MDS = 0xx</th> <th>MDS = 100 or MDS = 110</th> <th>MDS = 101 or MDS = 111</th> </tr> </thead> <tbody> <tr> <td>000: AN0</td> <td>000: AN0</td> <td>000: AN0</td> </tr> <tr> <td>001: AN1</td> <td>001: AN0, AN1</td> <td>001: AN0, AN1</td> </tr> <tr> <td>010: AN2</td> <td>010: AN0 to AN2</td> <td>010: AN0 to AN2</td> </tr> <tr> <td>011: AN3</td> <td>011: AN0 to AN3</td> <td>011: AN0 to AN3</td> </tr> <tr> <td>100: AN4</td> <td>100: AN4</td> <td>100: AN0 to AN4</td> </tr> <tr> <td>101: AN5</td> <td>101: AN4, AN5</td> <td>101: AN0 to AN5</td> </tr> <tr> <td>110: AN6</td> <td>110: AN4 to AN6</td> <td>110: AN0 to AN6</td> </tr> <tr> <td>111: AN7</td> <td>111: AN4 to AN7</td> <td>111: AN0 to AN7</td> </tr> </tbody> </table>	MDS = 0xx	MDS = 100 or MDS = 110	MDS = 101 or MDS = 111	000: AN0	000: AN0	000: AN0	001: AN1	001: AN0, AN1	001: AN0, AN1	010: AN2	010: AN0 to AN2	010: AN0 to AN2	011: AN3	011: AN0 to AN3	011: AN0 to AN3	100: AN4	100: AN4	100: AN0 to AN4	101: AN5	101: AN4, AN5	101: AN0 to AN5	110: AN6	110: AN4 to AN6	110: AN0 to AN6	111: AN7	111: AN4 to AN7	111: AN0 to AN7
MDS = 0xx	MDS = 100 or MDS = 110	MDS = 101 or MDS = 111																													
000: AN0	000: AN0	000: AN0																													
001: AN1	001: AN0, AN1	001: AN0, AN1																													
010: AN2	010: AN0 to AN2	010: AN0 to AN2																													
011: AN3	011: AN0 to AN3	011: AN0 to AN3																													
100: AN4	100: AN4	100: AN0 to AN4																													
101: AN5	101: AN4, AN5	101: AN0 to AN5																													
110: AN6	110: AN4 to AN6	110: AN0 to AN6																													
111: AN7	111: AN4 to AN7	111: AN0 to AN7																													

## [Legend]

x: Don't care

Note 1. Only 0 can be written to clear the flag after 1 is read.

Note 2. Clear the ADST to stop A/D conversion before transition to software standby mode or module standby mode.

Note 3. Set the A/D conversion time to minimum or more values to meet the absolute accuracy of the A/D converter characteristics.

Note 4.  $t_{cyc}$  indicates a cycle time of the peripheral clock 1 (P1 $\phi$ ).

### 27.3.5 A/D Comparison Interrupt Enable Register (ADCMPER)

ADCMPER is a 16-bit readable/writable register that enables or disables comparison between the A/D data register (ADDR) that holds the A/D conversion result on each channel and the A/D comparison upper/lower limit value register (ADCMPLH/ADCMPL), and enables or disables interrupt request.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HLMENH	HLMENG	HLMENF	HLMENE	HLMEND	HLMENC	HLMENB	HLMENA	LLMENH	LLMENG	LLMENF	LLMENE	LLMEND	LLMENC	LLMENB	LLMENA
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	HLMENH to HLMENA	All 0	R/W	<p>Comparison with Upper Limit Value and Interrupt Enable</p> <p>Enables or disables comparison between an A/D conversion result and the upper limit value as well as an LMTI interrupt request in the case of the result being greater than the upper limit value.</p> <p>0: Disables comparison between an A/D conversion result and the upper limit value as well as an over-limit interrupt (LMTI) request.</p> <p>1: Enables comparison between an A/D conversion result and the upper limit value as well as an over-limit interrupt (LMTI) request.</p>
7 to 0	LLMENH to LLMENA	All 0	R/W	<p>Comparison with Lower Limit Value and Interrupt Enable</p> <p>Enables or disables comparison between an A/D conversion result and the lower limit value as well as an LMTI interrupt request in the case of the result being smaller than the lower limit value.</p> <p>0: Disables comparison between an A/D conversion result and the lower limit value as well as an over-limit interrupt (LMTI) request.</p> <p>1: Enables comparison between an A/D conversion result and the lower limit value as well as an over-limit interrupt (LMTI) request.</p>

### 27.3.6 A/D Comparison Status Register (ADCMPSR)

ADCMPSR is a 16-bit read-only register that indicates the results of comparison between the A/D data register (ADDR) that holds an A/D conversion result on each channel and the A/D comparison upper/lower limit value register (ADCMPL/ADCMPL).

The bits in this register are valid only when the comparison and an LMTI interrupt are enabled in the A/D comparison interrupt enable register (ADCMPIER). When they are disabled, the bits are always in the cleared state.

When an A/D conversion result exceeds the upper or lower limit value, the corresponding bit is set to 1.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HOVRH	HOVRG	HOVRF	HOVRE	HOVRD	HOVRC	HOVRB	HOVRA	LUDRH	LUDRG	LUDRF	LUDRE	LUDRD	LUDRC	LUDRB	LUDRA
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	HOVRH to HOVRA	All 0	R	Upper Limit Over Flag Status flag that indicates that an A/D conversion result on each channel has exceeded the upper limit value in ADCMPH. [Clearing condition] <ul style="list-style-type: none"> <li>Cleared by writing 0 to the HLMENn bit for the corresponding channel in ADCMPIER register</li> </ul> [Setting condition] <ul style="list-style-type: none"> <li>The HLMEN bit for the corresponding channel in ADCMPIER is 1, and an A/D conversion result for the corresponding channel that is greater than the upper limit value in ADCMPH is stored in the ADDR register</li> </ul>
7 to 0	LUDRH to LUDRA	All 0	R	Lower Limit Under Flag Status flag that indicates that an A/D conversion result on each channel has exceeded the lower limit value in ADCMPL. [Clearing condition] <ul style="list-style-type: none"> <li>Cleared by writing 0 to the LLMENn bit for the corresponding channel in ADCMPIER register</li> </ul> [Setting condition] The LLMEN bit for the corresponding channel in ADCMPIER is 1, and an A/D conversion result for the corresponding channel that is smaller than the lower limit value in ADCMPL is stored in the ADDR register



## 27.4 Operation

The A/D converter uses the successive-approximation method, and the resolution is 12 bits. It has three operating modes: single mode, multi mode, and scan mode. Switching the operating mode or analog input channels must be done while the ADST bit in ADCSR is 0 to prevent incorrect operation. The ADST bit can be set at the same time as the operating mode or analog input channels are changed.

### 27.4.1 Single Mode

Single mode should be selected when only A/D conversion on one channel is required.

In single mode, A/D conversion is performed once for the specified one analog input channel, as follows:

1. A/D conversion for the selected channel starts when the ADST bit in ADCSR is set to 1 by software, the multi-function timer pulse unit 2, or external trigger input.
2. When A/D conversion is completed, the A/D conversion result is transferred to the A/D data register corresponding to the channel.
3. After A/D conversion has completed, the ADF bit in ADCSR is set to 1. If the ADIE bit is set to 1 at this time, an ADI interrupt request is generated.
4. The ADST bit that remains 1 during A/D conversion is automatically cleared to 0 when A/D conversion is completed, and the A/D converter becomes idle.

When the operating mode or analog input channel selection must be changed during A/D conversion, to prevent incorrect operation, first clear the ADST bit to 0 to halt A/D conversion. After making the necessary changes, set the ADST bit to 1 to start A/D conversion again. The ADST bit can be set at the same time as the mode or channel selection is switched.

Typical operations when a single channel (AN1) is selected in single mode (MDS[2] = 0) are described next. Figure 27.2 shows a timing diagram for this example (the bits which are set in this example belong to ADCSR).

1. Single mode is selected, input channel AN1 is selected (CH[2:0] = 001), the A/D interrupt is enabled (ADIE = 1), and A/D conversion is started (ADST = 1).
2. When A/D conversion is completed, the A/D conversion result is transferred into ADDR0. At the same time the ADF flag is set to 1, the ADST bit is cleared to 0, and the A/D converter becomes idle.
3. Since ADF = 1 and ADIE = 1, an ADI interrupt is requested.
4. The A/D interrupt handling routine starts.
5. The routine reads ADF = 1, and then writes 0 to the ADF flag.
6. The routine reads and processes the A/D conversion result (ADDR0).
7. Execution of the A/D interrupts handling routine ends. Then, when the ADST bit is set to 1, A/D conversion starts and steps 2 to 7 are executed.

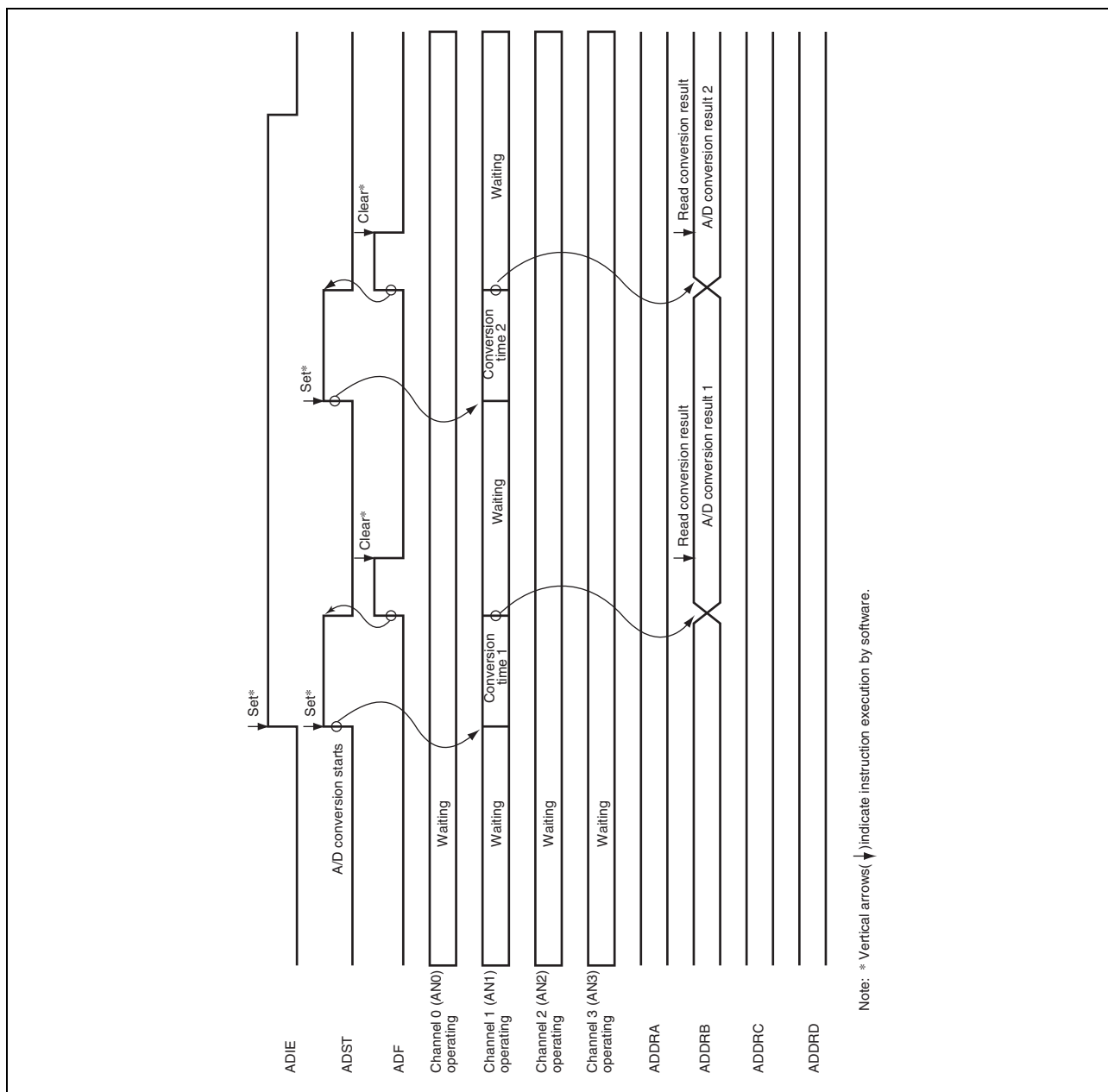


Figure 27.2 Example of A/D Converter Operation (Single Mode, One Channel (AN1) Selected)

### 27.4.2 Multi Mode

Multi mode should be selected when performing A/D conversion once on one or more channels.

In multi mode, A/D conversion is performed once for a maximum of eight specified analog input channels, as follows:

1. A/D conversion starts from the analog input channel with the lowest number (e.g. AN0, AN1, ..., AN3) when the ADST bit in ADCSR is set to 1 by software, the multi-function timer pulse unit 2, or external trigger input.
2. When A/D conversion is completed on each channel, the A/D conversion result is sequentially transferred to the A/D data register corresponding to that channel.
3. After A/D conversion on all selected channels has completed, the ADF bit in ADCSR is set to 1. If the ADIE bit is set to 1 at this time, an ADI interrupt request is generated.
4. The ADST bit that remains 1 during A/D conversion is automatically cleared to 0 when A/D conversion is completed, and the A/D converter becomes idle. If the ADST bit is cleared to 0 during A/D conversion, A/D conversion is halted and the A/D converter becomes idle. The ADF bit is cleared by reading ADF while ADF = 1, then writing 0 to the ADF bit.

A/D conversion is to be performed once on all the specified channels. The conversion results are transferred for storage into the A/D data registers (ADDR) corresponding to the channels.

When the operating mode or analog input channel selection must be changed during A/D conversion, to prevent incorrect operation, first clear the ADST bit to 0 to halt A/D conversion. After making the necessary changes, set the ADST bit to 1.

A/D conversion will start again from the first channel in the group. The ADST bit can be set at the same time as the mode or channel selection is changed.

Typical operations when three channels (AN0 to AN2) are selected in multi mode are described next. Figure 27.3 shows a timing diagram for this example.

1. Multi mode is selected (MDS[2] = 1, MDS[1] = 0), analog input channels AN0 to AN2 are selected (CH[2:0] = 010), and A/D conversion is started (ADST = 1).
2. A/D conversion of the first channel (AN0) starts. When A/D conversion is completed, the A/D conversion result is transferred into ADDR0.
3. Next, the second channel (AN1) is selected automatically and A/D conversion starts.
4. Conversion proceeds in the same way through the third channel (AN2).
5. When conversion of all selected channels (AN0 to AN2) is completed, the ADF flag is set to 1 and the ADST bit is cleared to 0.
6. If the ADIE bit is set to 1 at this time, an ADI interrupt is requested.

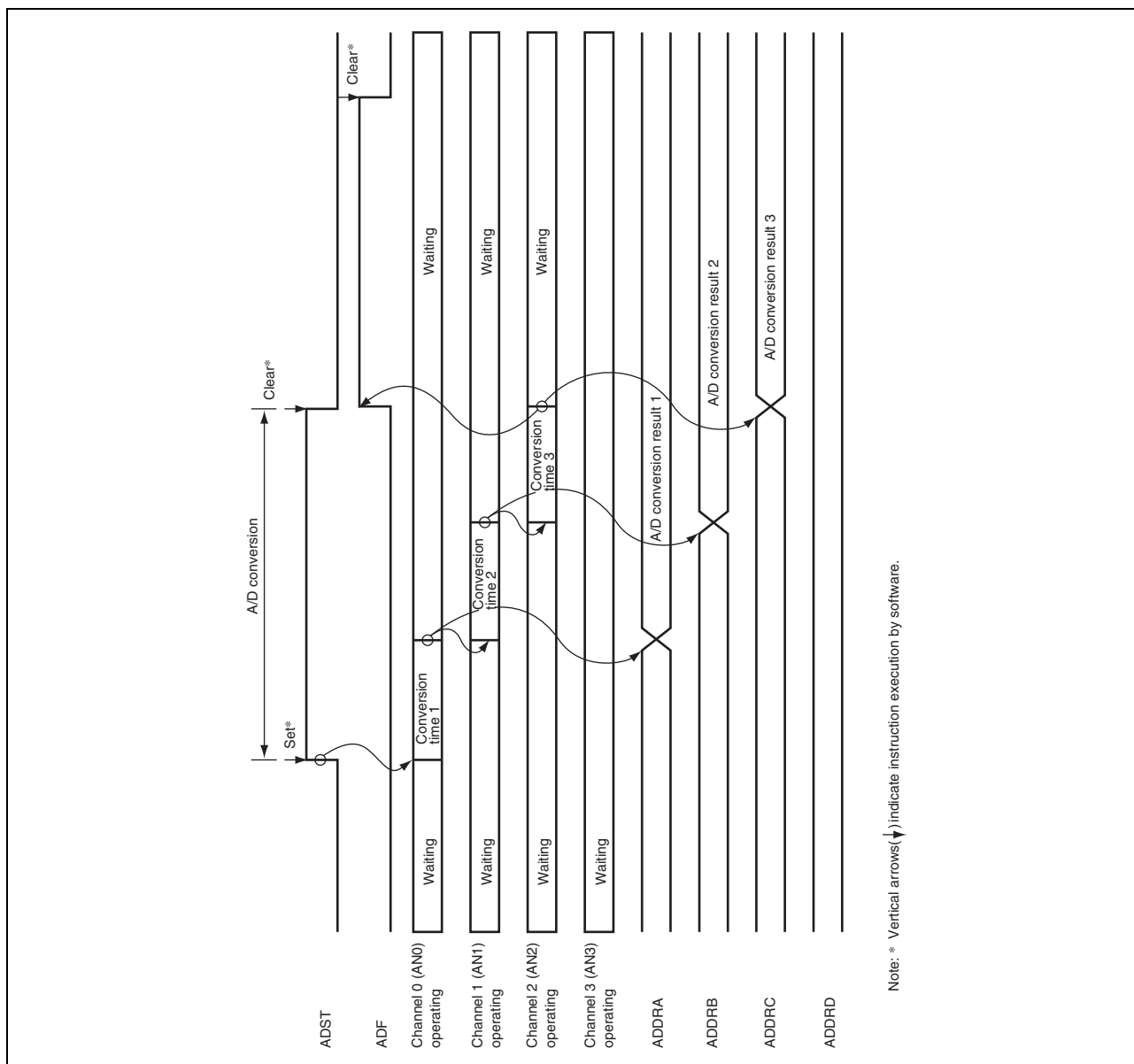


Figure 27.3 Example of A/D Converter Operation (Multi Mode, Three Channels (AN0 to AN2) Selected)

### 27.4.3 Scan Mode

Scan mode is useful for monitoring analog inputs in a group of one or more channels at all times. In scan mode, A/D conversion is performed sequentially for a maximum of eight specified analog input channels, as follows:

1. A/D conversion starts from the analog input channel with the lowest number (e.g. AN0, AN1, ..., AN3) when the ADST bit in ADCSR is set to 1 by software, the multi-function timer pulse unit 2, or external trigger input.
2. When A/D conversion is completed on each channel, the A/D conversion result is sequentially transferred to the A/D data register corresponding to that channel.
3. After A/D conversion on all selected channels has completed, the ADF bit in ADCSR is set to 1. If the ADIE bit is set to 1 at this time, an ADI interrupt request is generated. The A/D converter starts A/D conversion again from the channel with the lowest number.
4. The ADST bit is not cleared automatically, so steps 2. and 3. are repeated as long as the ADST bit remains set to 1. When the ADST bit is cleared to 0, A/D conversion halts and the A/D converter becomes idle. The ADF bit is cleared by reading ADF while ADF = 1, then writing 0 to the ADF bit.

When the operating mode or analog input channel selection must be changed during A/D conversion, to prevent incorrect operation, first clear the ADST bit to 0 to halt A/D conversion. After making the necessary changes, set the ADST bit to 1. A/D conversion will start again from the first channel in the group. The ADST bit can be set at the same time as the mode or channel selection is changed.

Typical operations when three channels (AN0 to AN2) are selected in scan mode are described as follows. Figure 27.4 shows a timing diagram for this example.

1. Scan mode is selected (MDS[2] = 1, MDS[1] = 1), analog input channels AN0 to AN2 are selected (CH[2:0] = 010), and A/D conversion is started (ADST = 1).
2. A/D conversion of the first channel (AN0) starts. When A/D conversion is completed, the A/D conversion result is transferred into ADDRA.
3. Next, the second channel (AN1) is selected automatically and A/D conversion starts.
4. Conversion proceeds in the same way through the third channel (AN2).
5. When conversion of all the selected channels (AN0 to AN2) is completed, the ADF flag is set to 1 and conversion of the first channel (AN0) starts again. If the ADIE bit is set to 1 at this time, an ADI interrupt is requested after A/D conversion of the third channel.
6. The ADST bit is not cleared automatically, so steps 2. to 4. are repeated as long as the ADST bit remains set to 1. When steps 2. to 4. are repeated, the ADF flag is kept to 1. When the ADST bit is cleared to 0, A/D conversion stops. The ADF bit is cleared by reading ADF while ADF = 1, then writing 0 to the ADF bit.

If both the ADF flag and ADIE bit are set to 1 while steps 2. to 4. are repeated, an ADI interrupt is requested at all times. To generate an interrupt on completing conversion of the third channel, clear the ADF bit to 0 after an interrupt is requested.

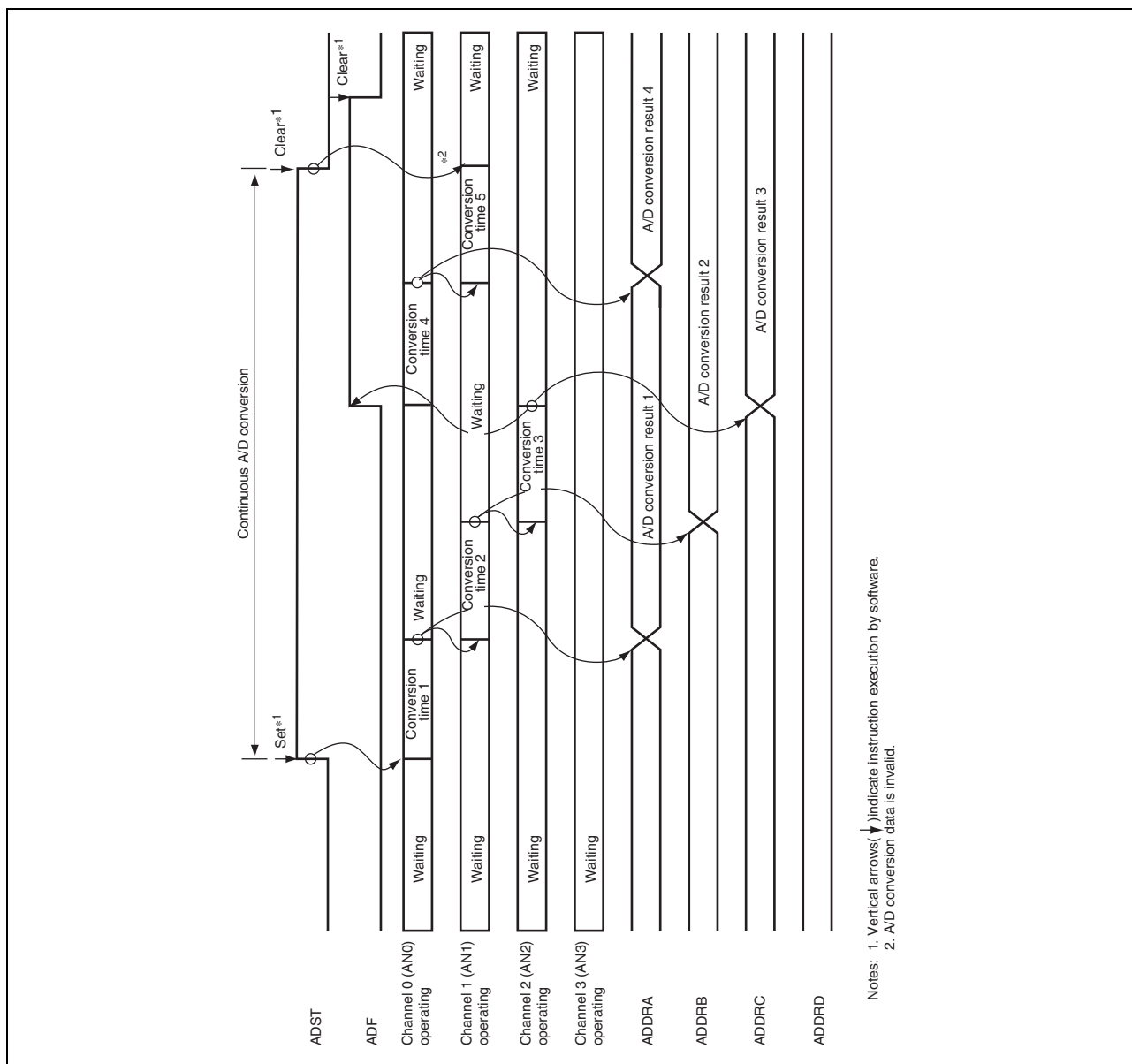


Figure 27.4 Example of A/D Converter Operation (Scan Mode, Three Channels (AN0 to AN2) Selected)

### 27.4.4 A/D Converter Activation by External Trigger or Multi-Function Timer Pulse Unit 2

The A/D converter can be independently activated by an external trigger or an A/D conversion request from the multi-function timer pulse unit 2. To activate the A/D converter by an external trigger or the multi-function timer pulse unit 2, set the A/D trigger enable bits (TRGS[3:0]). When an external trigger or an A/D conversion request from the multi-function timer pulse unit 2 is generated with this bit setting, the ADST bit is set to 1 to start A/D conversion. The channel combination is determined by bits CH2 to CH0 in ADCSR. The timing from setting of the ADST bit until the start of A/D conversion is the same as when 1 is written to the ADST bit by software.

### 27.4.5 Input Sampling and A/D Conversion Time

The A/D converter has a built-in sample-and-hold circuit. The A/D converter samples the analog input at the A/D conversion start delay time ( $t_D$ ) after the ADST bit in ADCSR is set to 1, then starts conversion. Figure 27.5 shows the A/D conversion timing. Table 27.4 indicates the A/D conversion time.

As indicated in Figure 27.5, the A/D conversion time ( $t_{CONV}$ ) includes  $t_D$  and the input sampling time ( $t_{SPL}$ ). The length of  $t_D$  varies depending on the timing of the write access to ADCSR. The total conversion time therefore varies within the ranges indicated in Table 27.4.

In multi mode and scan mode, the values given in Table 27.4 apply to the first conversion. In the second and subsequent conversions, time is the values given in Table 27.5.

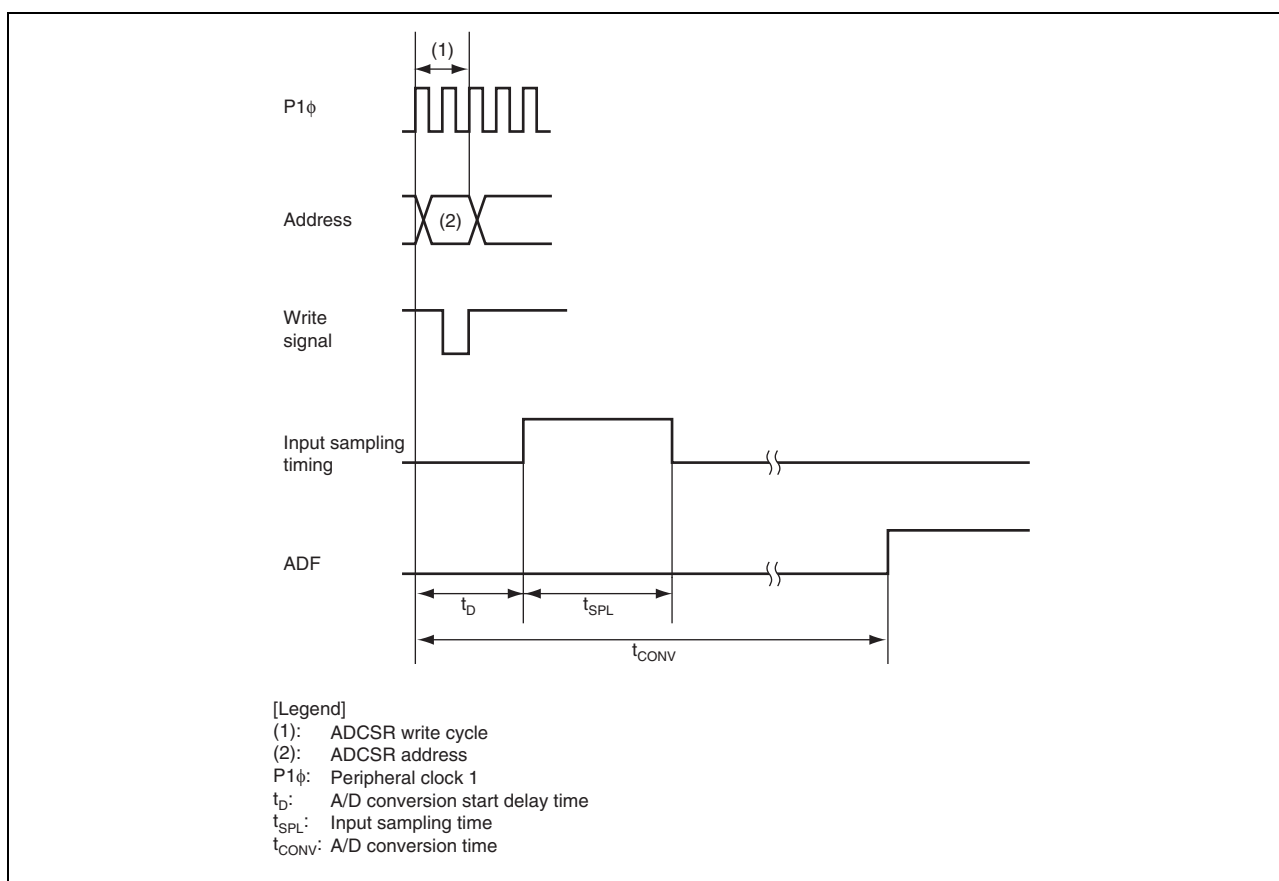


Figure 27.5 A/D Conversion Timing

**Table 27.5 A/D Conversion Time (Single Mode)**

Item	Symbol	CKS1 = 0						CKS1 = 1					
		CKS0 = 0			CKS0 = 1			CKS0 = 0			CKS0 = 1		
		Min.	Typ.	Max.	Min.	Typ.	Max.	Min.	Typ.	Max.	Min.	Typ.	Max.
A/D conversion start delay time	$t_D$	9	—	14	10	—	16	11	—	18	12	—	20
Input sampling time	$t_{SPL}$	—	78	—	—	91	—	—	104	—	—	117	—
A/D conversion time	$t_{CONV}$	251	—	256	292	—	298	333	—	340	374	—	382

Note: Values in the table are represented in terms of  $t_{cyc}$ .  $t_{cyc}$  indicates a cycle time of the peripheral clock 1 ( $P1\phi$ ).

**Table 27.6 A/D Conversion Time (Multi Mode and Scan Mode)**

CKS1	CKS0	Conversion Time ( $t_{cyc}$ )
0	0	240 (constant)
	1	280 (constant)
1	0	320 (constant)
	1	360 (constant)

Note: Values in the table are represented in terms of  $t_{cyc}$ .  $t_{cyc}$  indicates a cycle time of the peripheral clock 1 ( $P1\phi$ ).



### 27.4.6 External Trigger Input Timing

A/D conversion can also be externally triggered. When the TRGS[3:0] bits in ADCSR are set to B'1001, an external trigger is input to the  $\overline{\text{ADTRG}}$  pin. The ADST bit in ADCSR is set to 1 at the falling edge of the ADTRG pin, thus starting A/D conversion. Other operations, regardless of the operating mode, are the same as when the ADST bit has been set to 1 by software. Figure 27.6 shows the timing.

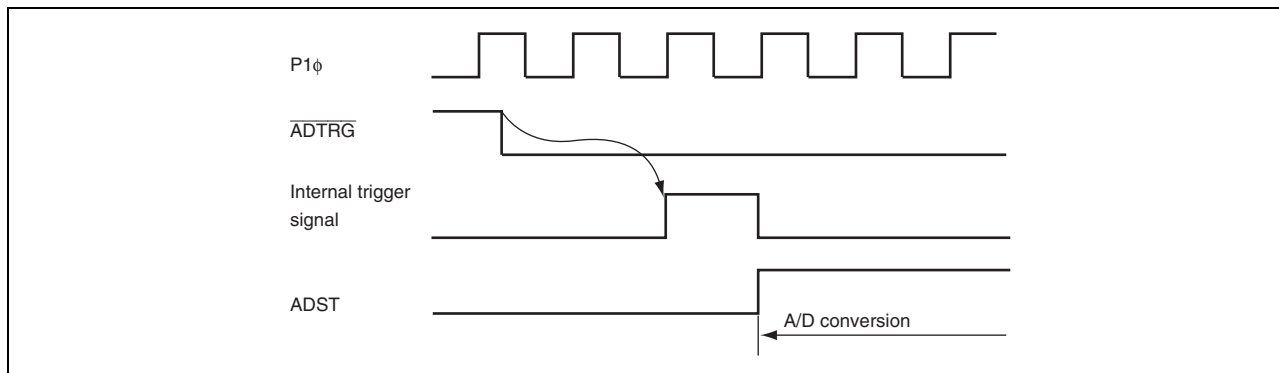


Figure 27.6 External Trigger Input Timing

## 27.5 Interrupt Sources and DMA Transfer Request

The A/D converter generates an A/D conversion end interrupt (ADI) and an over-limit interrupt (LMTI). The direct memory access controller can be activated by an ADI interrupt request depending on the setting of the direct memory access controller.

Table 27.6 shows the relationship between interrupt sources and the DMA transfer request.

**Table 27.7 Relation between Interrupt Sources and DMA Transfer Request**

Name	Interrupt Source	Interrupt Flag	Direct Memory Access Controller Activation
ADI	A/D conversion end	ADF in ADCSR	Possible
LMTI	Conversion result exceeding the limit value	HOVRA to HOVRH and LUDRA to LUDRH in ADCMPSR	Impossible

### (a) ADI Interrupt

The A/D converter generates an A/D conversion end interrupt (ADI) at the end of A/D conversion. An ADI interrupt request is generated if the ADIE bit is set to 1 when the ADF bit in ADCSR is set to 1 on completion of A/D conversion. Note that the direct memory access controller can be activated by an ADI interrupt depending on the setting of the direct memory access controller. In this case, an interrupt is not issued to the CPU. If the setting to activate the direct memory access controller has not been made, an interrupt request is sent to the CPU. Having the converted data read by the direct memory access controller in response to an ADI interrupt enables continuous conversion to be achieved without imposing a load on software.

In single mode, set the direct memory access controller so that DMA transfer initiated by an ADI interrupt is performed only once. In the case of A/D conversion on multiple channels in scan mode or multi mode, setting the DMA transfer count to one causes DMA transfer to finish after transferring only one channel of data. To make the direct memory access controller transfer all conversion data, set the ADDR where A/D conversion data is stored as the transfer source address, and the number of converted channels as the transfer count.

When the direct memory access controller is activated by ADI, the ADF bit in ADCSR is automatically cleared to 0 when data is transferred by the direct memory access controller.

### (b) LMTI Interrupt

An A/D conversion result is compared with the upper or lower limit value on each channel, and if the result exceeds the limit, the over-limit interrupt (LMTI) is generated. While the HLMEN bit in ADCMPER is 1, a conversion result is compared with the upper limit in ADCMPH the instant the conversion value is determined. When the value exceeds the upper limit, the corresponding bit in ADCMPSR is set to 1, which causes an LMTI interrupt to be generated. In the same way, while the LLMEN bit in ADCMPER is 1, a conversion result is compared with the lower limit. If it exceeds the limit, an LMTI interrupt request is generated.

The ADCMPSR register can be referenced in the LMTI interrupt handling to identify the channel causing over-limit value and to know whether the result has exceeded the upper limit value or the lower one.

## 27.6 Definitions of A/D Conversion Accuracy

The A/D converter compares an analog value input from an analog input channel with its analog reference value and converts it to 10-bit digital data. The absolute accuracy of this A/D conversion is the deviation between the input analog value and the output digital value. It includes the following errors:

- Offset error
- Full-scale error
- Quantization error
- Nonlinearity error

These four error quantities are explained below with reference to Figure 27.7. In the figure, the 12-bit A/D converter is illustrated as the 3-bit A/D converter for explanation. Offset error is the deviation between actual and ideal A/D conversion characteristics when the digital output value changes from the minimum (zero voltage) B'000000000000 (000 in the figure) to B'000000000001 (001 in the figure) (Figure 27.7, item (1)). Full-scale error is the deviation between actual and ideal A/D conversion characteristics when the digital output value changes from B'111111111110 (110 in the figure) to the maximum B'111111111111 (111 in the figure) (Figure 27.7, item (2)). Quantization error is the intrinsic error of the A/D converter and is expressed as 1/2 LSB (Figure 27.7, item (3)). Nonlinearity error is the deviation between actual and ideal A/D conversion characteristics between zero voltage and full-scale voltage (Figure 27.7, item (4)). Note that it does not include offset, full-scale, or quantization error.

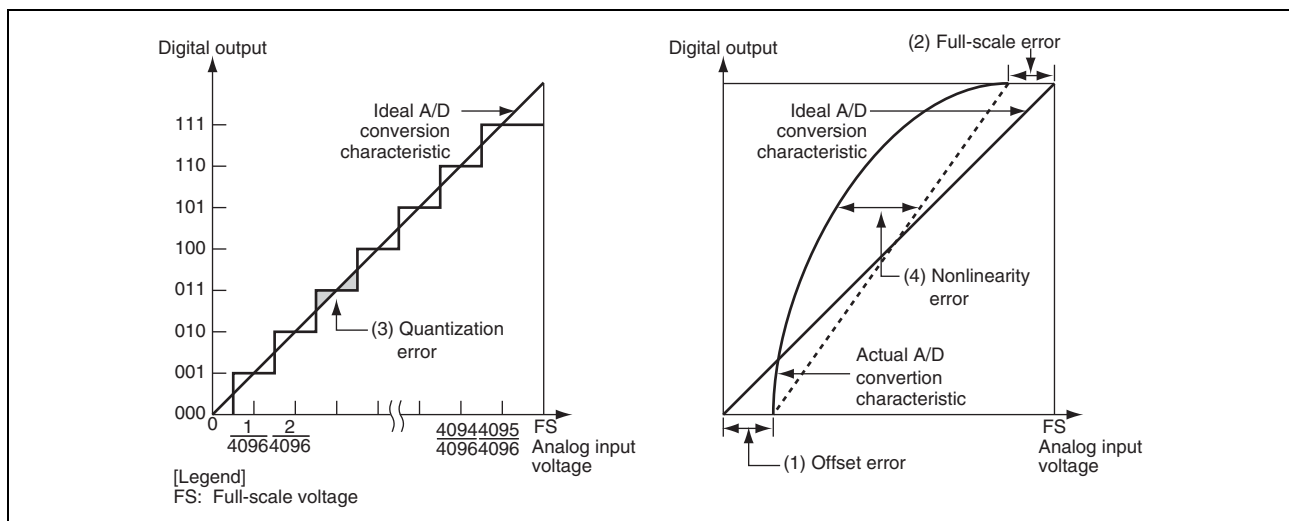


Figure 27.7 Definitions of A/D Conversion Accuracy

## 27.7 Usage Notes

When using the A/D converter, note the following points.

### 27.7.1 Module Standby Mode Setting

Operation of the A/D converter can be disabled or enabled using the standby control register. The initial setting is for operation of the A/D converter to be halted. Register access is enabled by clearing module standby mode. For details, see section 55, Power-Down Modes. The wait time (20  $\mu$ s) is required after release from the module standby state. After the wait time has elapsed, perform conversion by the A/D converter.

### 27.7.2 Setting Analog Input Voltage

Using the LSI outside the following voltage ranges may impair the LSI reliability.

1. Analog input range

During A/D conversion, voltages on the analog input pins ANn should not go beyond the following range:  $AV_{ss} \leq AN_n \leq AV_{cc}$  (n = 0 to 7).

2. AVcc and AVss input voltages

Input voltages AVcc and AVss should be  $PV_{cc} - 0.3 \text{ V} \leq AV_{cc} \leq PV_{cc}$  and  $AV_{ss} = V_{ss}$ . Do not leave the AVcc and AVss pins open even when the A/D converter is not in use and in software standby mode. When not in use, connect AVcc to the power supply (PVcc) and AVss to the ground (Vss).

3. Setting range of AVref input voltage

Set the reference voltage range of the AVref pin as  $3.0 \text{ V} \leq AV_{ref} \leq AV_{cc}$ .

### 27.7.3 Notes on Board Design

In board design, digital circuitry and analog circuitry should be as mutually isolated as possible, and layout in which digital circuit signal lines and analog circuit signal lines cross or are in close proximity should be avoided as far as possible. Failure to do so may result in incorrect operation of the analog circuitry due to inductance, adversely affecting A/D conversion values.

Digital circuitry must be isolated from the analog input signals (AN0 to AN7), analog reference voltage (AVref), and analog power supply (AVcc) by the analog ground (AVss). Also, the analog ground (AVss) should be connected at one point to a stable digital ground (Vss) on the board.

### 27.7.4 Processing of Analog Input Pins

To prevent damage from voltage surges at the analog input pins (AN0 to AN7), connect an input protection circuit like the one shown in Figure 27.8. The circuit shown also includes a CR filter to suppress noise. This circuit is shown as an example; the circuit constants should be selected according to actual application conditions.

Figure 27.9 shows an equivalent circuit diagram of the analog input ports and Table 27.8 lists the analog input pin specifications.

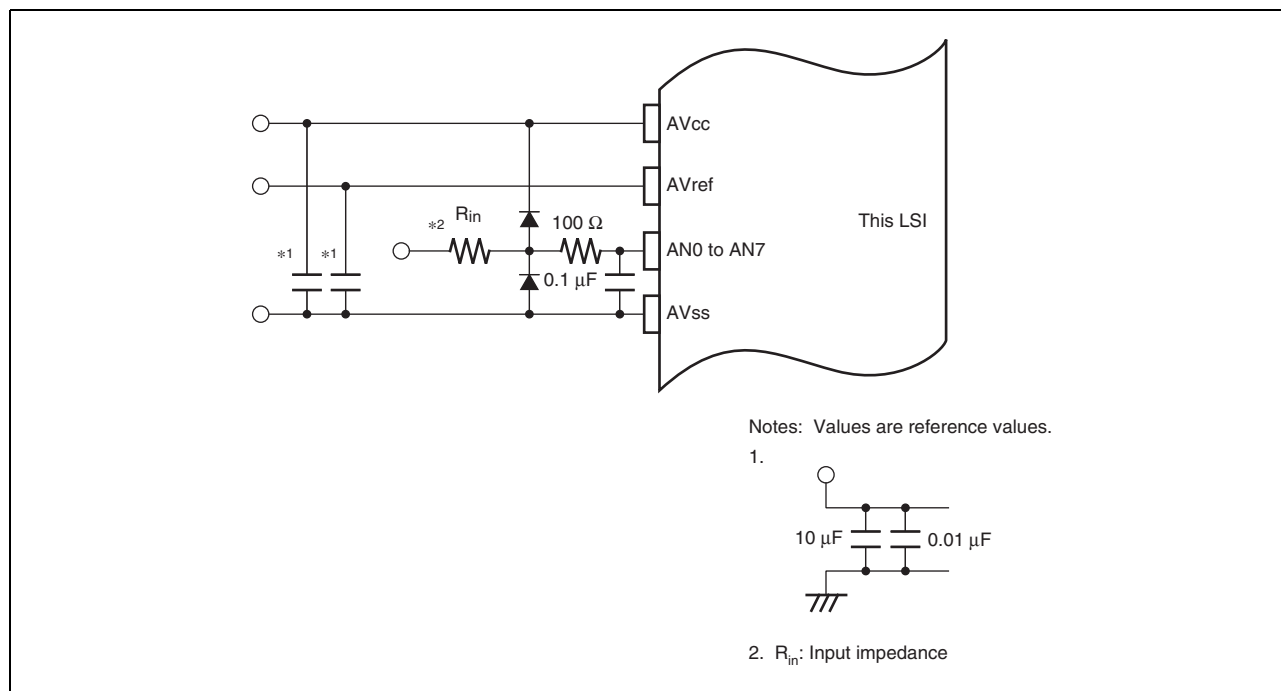


Figure 27.8 Example of Analog Input Protection Circuit

### 27.7.5 Permissible Signal Source Impedance

This LSI's analog input is designed such that conversion precision is guaranteed for an input signal for which the signal source impedance is 3 k $\Omega$  or less. This specification is provided to enable the A/D converter's sample-and-hold circuit input capacitance to be charged within the sampling time; if the sensor output impedance exceeds 3 k $\Omega$ , charging may be insufficient and it may not be possible to guarantee A/D conversion precision. However, for A/D conversion in single mode with a large capacitance provided externally for A/D conversion in single mode, the input load will essentially comprise only the internal input resistance of 1 k $\Omega$ , and the signal source impedance is ignored. However, as a low-pass filter effect is obtained in this case, it may not be possible to follow an analog signal with a large differential coefficient (e.g., 5 mV/ $\mu$ s or greater) (see Figure 27.9). When converting a high-speed analog signal or performing A/D conversion in scan mode, a low-impedance buffer should be inserted.

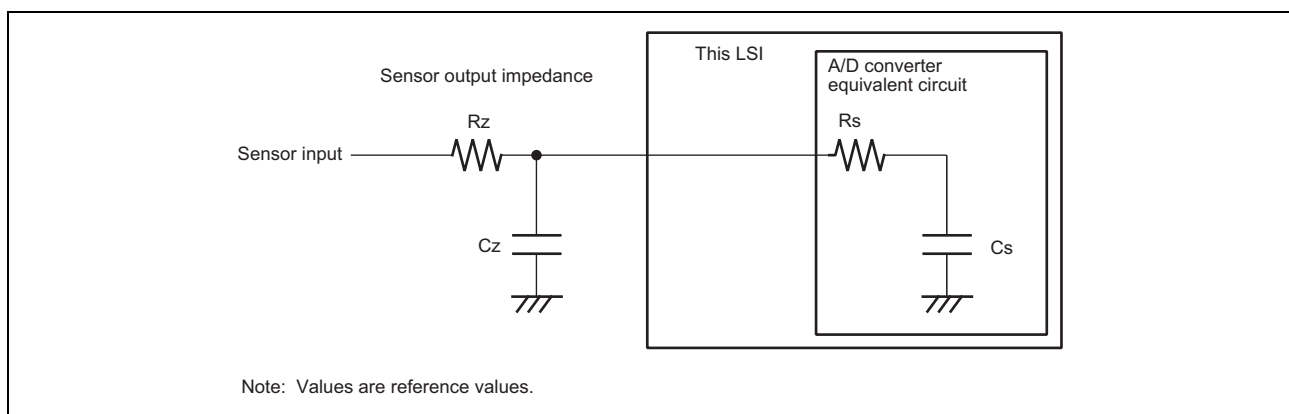


Figure 27.9 Example of Analog Input Circuit

Table 27.8 Analog Input Pin Ratings

Item	Symbol	Min.	Max.	Unit
Permissible Signal Source Impedance	Rz	—	3	k $\Omega$
Low-Pass Filter	Cz	—	0.1	$\mu$ F
Equivalent Circuit of A/D Converter	Rs	—	1	k $\Omega$
	Cs	—	20	pF

### 27.7.6 Influences on Absolute Precision

Adding capacitance results in coupling with GND, and therefore noise in GND may adversely affect absolute precision. Be sure to connect AVSS, etc. to an electrically stable GND. Care is also required to insure that filter circuits do not communicate with digital signals on the mounting board (i.e., acting as antennas).

### 27.7.7 Usage Note on Port Pins

The analog input pins (AN0 to AN7) are multiplexed with general-purpose I/O port pin functions (pins P1\_8 to P1\_15) and the latter can be used as digital inputs. Do not use these general-purpose I/O port pins as digital inputs while the A/D converter is in use with 12-bit precision.

## 28. NAND Flash Memory Controller

The NAND flash memory controller provides interfaces for an external NAND-type flash memory.

### 28.1 Features

#### (1) NAND-Type Flash Memory Interface

- Interface directly connectable to NAND-type flash memory
- Read or write in byte units
- Supports large-block (2048 + 64 bytes) flash memory\*
- Supports addresses for 2 Gbits and more by extension to 5-byte addresses

#### (2) Access Mode

- Command access mode: Performs an access by specifying in a register a command to be issued from this module to flash memory, address, and data size to be input or output.

#### (3) Data Error

- When a program error or erase error occurs, the error is reflected on the error source flags. Interrupts for each source can be specified.

#### (4) Data Transfer FIFO and Data Register

- The 224-byte data FIFO register (FLDTFIFO) is incorporated for data transfer of flash memory.

#### (5) DMA Transfer

#### (6) Access Time

- The operating clock (FCLK) on the pins for the NAND-type flash memory is generated by dividing the peripheral clock 0 (P0φ). The division ratio can be specified by the QTSEL bit in the common control register (FLCMNCR).
- Before changing the clock pulse generator configuration, this module must be placed in a module stop state.
- In NAND-type flash memory, the FRE and FWE pins operate at the frequency of FCLK. The operating frequency must be specified within the maximum operating frequency of memory to be connected.

Figure 28.1 shows a block diagram.

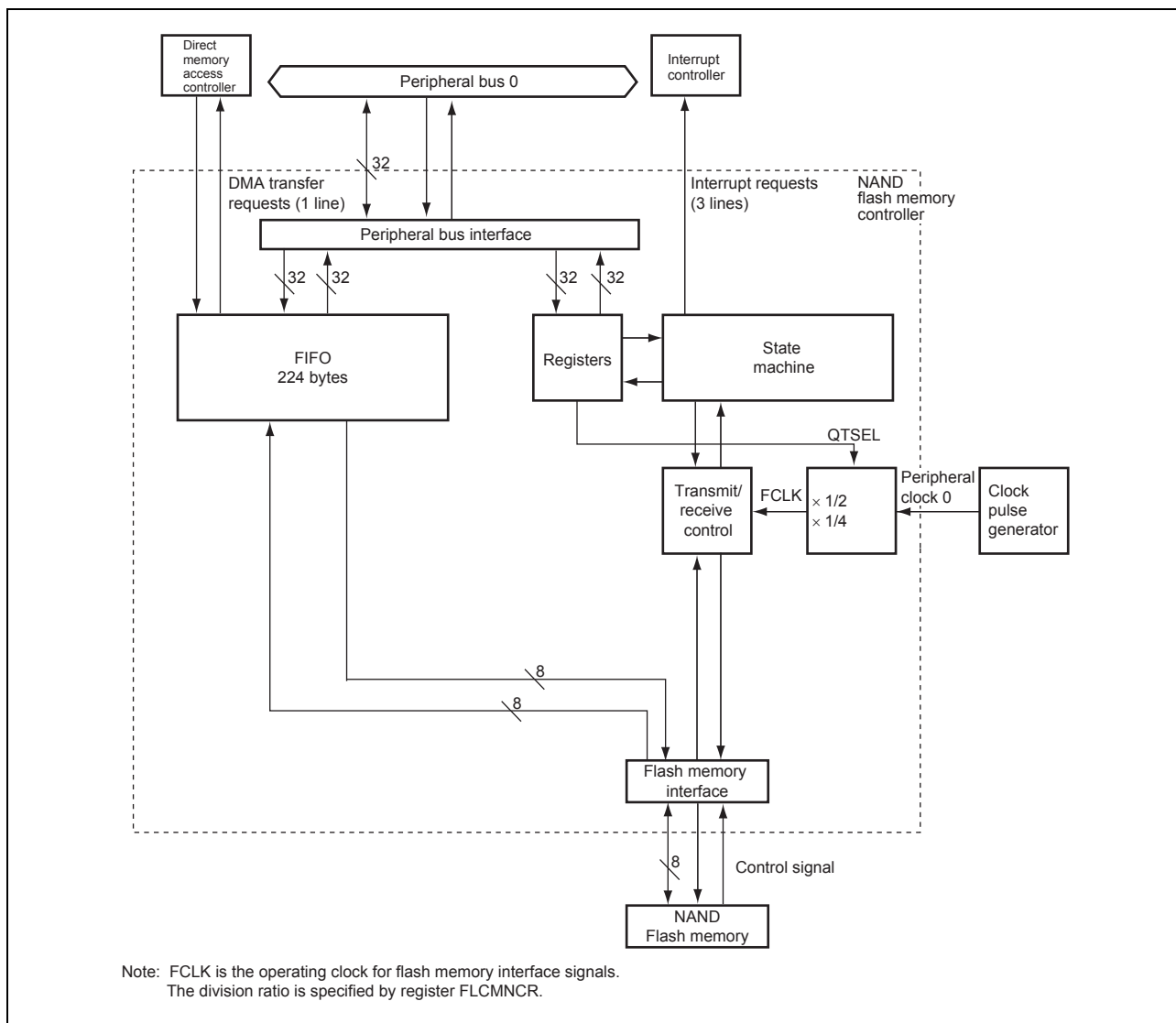


Figure 28.1 Block Diagram



## 28.2 Input/Output Pins

The pin configuration is listed in Table 28.1.

**Table 28.1 Pin Configuration**

Pin Name	I/O	Corresponding Flash Memory Pin	Function
		NAND Type	
$\overline{\text{FCE}}$	Output	$\overline{\text{CE}}$	Flash Memory Chip Enable Enables flash memory connected to this LSI.
NAF7 to NAF0	I/O	I/O7 to I/O0	Flash Memory Data I/O pins for command, address, and data.
$\overline{\text{FCLE}}$	Output	$\overline{\text{CLE}}$	Flash Memory Command Latch Enable Asserted when a command is output.
$\overline{\text{FALE}}$	Output	$\overline{\text{ALE}}$	Flash Memory Address Latch Enable Asserted when an address is output and negated when data is input or output.
$\overline{\text{FRE}}$	Output	$\overline{\text{RE}}$	Flash Memory Read Enable Reads data at the falling edge of $\overline{\text{RE}}$ .
$\overline{\text{FWE}}$	Output	$\overline{\text{WE}}$	Flash Memory Write Enable Flash memory latches a command, address, and data at the rising edge of $\overline{\text{WE}}$ .
FRB	Input	$\overline{\text{R/B}}$	Flash Memory Ready/Busy Indicates ready state at high level; indicates busy state at low level.
—*	—	$\overline{\text{WP}}$	Write Protect/Reset When this pin goes low, erroneous erasure or programming at power on or off can be prevented.
—*	—	$\overline{\text{SE}}$	Spare Area Enable Enables access to the spare area. This pin must be fixed at low in sector access mode.

Note: \* Not supported in this LSI.

### 28.3 Register Descriptions

Table 28.2 shows the register configuration.

**Table 28.2 Register Configuration**

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Common control register	FLCMNCR	R/W	H'00100001	H'FCFF4000	32
Command control register	FLCMDCR	R/W	H'00000000	H'FCFF4004	32
Command code register	FLCMCDR	R/W	H'00000000	H'FCFF4008	32
Address register	FLADR	R/W	H'00000000	H'FCFF400C	32
Address register 2	FLADR2	R/W	H'00000000	H'FCFF403C	32
Data register	FLDATAR	R/W	H'00000000	H'FCFF4010	32
Data counter register	FLDTCNTR	R/W	H'00000000	H'FCFF4014	32
Interrupt DMA control register	FLINTDMACR	R/W	H'00000000	H'FCFF4018	32
Ready busy timeout setting register	FLBSYTMR	R/W	H'00000000	H'FCFF401C	32
Ready busy timeout counter	FLBSYCNT	R	H'00000000	H'FCFF4020	32
Data FIFO register	FLDTFIFO	R/W	H'xxxxxxxx	H'FCFF4050	32
Transfer control register	FLTRCR	R/W	H'00	H'FCFF402C	8

### 28.3.1 Common Control Register (FLCMNCR)

FLCMNCR is a 32-bit readable/writable register that specifies access mode and other items.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	SNAND	QT SEL	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	ACM[1:0]		NAND WF	-	-	-	-	-	CE	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
18	SNAND	0	R/W	Large-Capacity NAND Flash Memory Select This bit is used to specify 1-Gbit or larger NAND flash memory with the page configuration of 2048 + 64 bytes. 0: When flash memory with the page configuration of 512 + 16 bytes is used. 1: When NAND flash memory with the page configuration of 2048 + 64 is used.
17	QTSEL	0	R/W	Flash Clock Division Ratio Select Selects the division ratio of clock FCLK in the flash memory. 0: Divides a clock (P0 $\phi$ ) provided from the clock pulse generator by two and uses it as FCLK. 1: Divides a clock (P0 $\phi$ ) provided from the clock pulse generator by four and uses it as FCLK.
16 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11, 10	ACM[1:0]	00	R/W	Access Mode Specification 1 and 0 Specify access mode. 00: Command access mode 01: Setting prohibited 10: Setting prohibited 11: Setting prohibited
9	NANDWF	0	R/W	NAND Wait Insertion Operation 0: Performs address or data input/output in one FCLK cycle 1: Performs address or data input/output in two FCLK cycles
8 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	CE	0	R/W	Chip Enable 0: Disables the chip (Outputs high level to the FCE pin) 1: Enables the chip (Outputs low level to the FCE pin)
2, 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.

### 28.3.2 Command Control Register (FLCMDCR)

FLCMDCR is a 32-bit readable/writable register that issues a command in command access mode, specifies address issue, and specifies source or destination of data transfer.

Bit: 31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADR CNT2	-	-	-	-	-	CDS RC	DOSR	-	-	SEL RW	DOA DR	ADRCNT[1:0]	DOC MD2	DOC MD1	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W
Bit: 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	ADRCNT2	0	R	Address Issue Byte Count Specification 2 Specifies the number of bytes for the address data to be issued in address stage. This bit is used together with ADRCNT[1:0]. 0: Issues the address of byte count, specified by ADRCNT[1:0]. 1: Issues 5-byte address. ADRCNT[1:0] should be set to 00.
30 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25	CDSRC	0	R/W	Data Buffer Specification Specifies the data buffer to be read from or written to in the data stage in command access mode. 0: Specifies FLDATAR as the data buffer. 1: Specifies FLDTFIFO as the data buffer.
24	DOSR	0	R/W	Status Read Check Specifies whether or not the status read is performed after the second command has been issued in command access mode. 0: Performs no status read 1: Performs status read
23, 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
21	SELRW	0	R/W	Data Read/Write Specification Specifies the direction of read or write in data stage. 0: Read 1: Write
20	DOADR	0	R/W	Address Stage Execution Specification Specifies whether or not the address stage is executed in command access mode. 0: Performs no address stage 1: Performs address stage
19, 18	ADRCNT[1:0]	00	R/W	Address Issue Byte Count Specification [1:0] Specify the number of bytes for the address data to be issued in address stage. 00: Issue 1-byte address 01: Issue 2-byte address 10: Issue 3-byte address 11: Issue 4-byte address
17	DOCMD2	0	R/W	Second Command Stage Execution Specification Specifies whether or not the second command stage is executed in command access mode. 0: Does not execute the second command stage 1: Executes the second command stage
16	DOCMD1	0	R/W	First Command Stage Execution Specification Specifies whether or not the first command stage is executed in command access mode. 0: Does not execute the first command stage 1: Executes the first command stage

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

### 28.3.3 Command Code Register (FLCMCDR)

FLCMCDR is a 32-bit readable/writable register that specifies a command code to be issued in command access.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

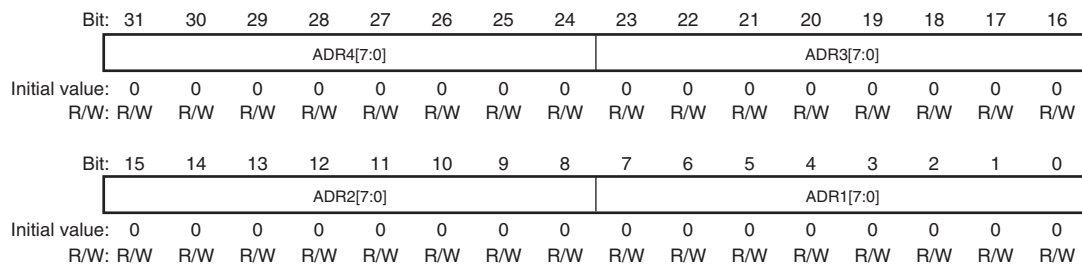
  

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMD2[7:0]								CMD1[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 8	CMD2[7:0]	H'00	R/W	Second Command Data Specify a command code to be issued in the second command stage.
7 to 0	CMD1[7:0]	H'00	R/W	First Command Data Specify a command code to be issued in the first command stage.

### 28.3.4 Address Register (FLADR)

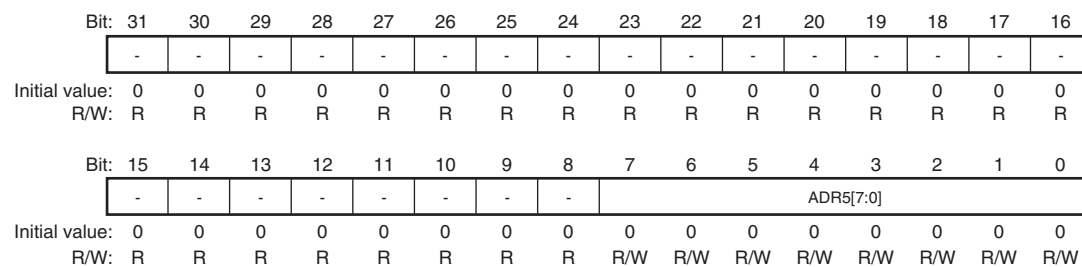
FLADR is a 32-bit readable/writable register that specifies the value to be output as an address. The address of the size specified by ADRCNT[1:0] in the command control register is output sequentially from ADR1 in byte units.



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	ADR4[7:0]	H'00	R/W	Fourth Address Data Specify 4th data to be output to flash memory as an address.
23 to 16	ADR3[7:0]	H'00	R/W	Third Address Data Specify 3rd data to be output to flash memory as an address.
15 to 8	ADR2[7:0]	H'00	R/W	Second Address Data Specify 2nd data to be output to flash memory as an address.
7 to 0	ADR1[7:0]	H'00	R/W	First Address Data Specify 1st data to be output to flash memory as an address.

### 28.3.5 Address Register 2 (FLADR2)

FLADR2 is a 32-bit readable/writable register, and is valid when the ADRCNT2 bit in FLCMDCR is set to 1. FLADR2 specifies the value to be output as an address in command access mode.



Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	ADR5[7:0]	H'00	R/W	Fifth Address Data Specify the 5th data to be output to flash memory as an address.

### 28.3.6 Data Counter Register (FLDTCNTR)

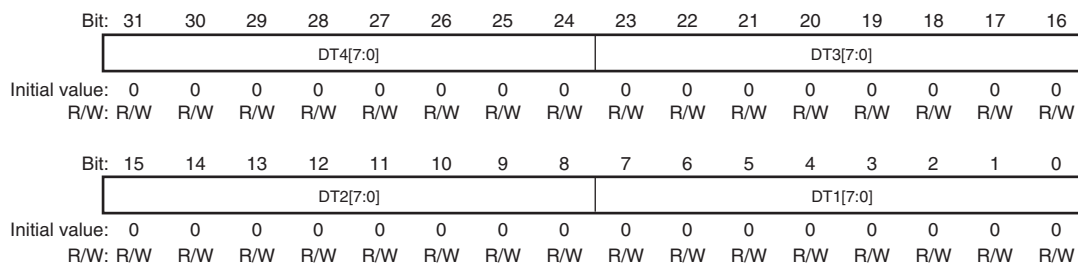
FLDTCNTR is a 32-bit readable/writable register that specifies the number of bytes to be read or written in command access mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	DTFLW[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	DTCNT[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	DTFLW[7:0]	H'00	R	FLDTFIFO Access Count Specify the number of longwords in FLDTFIFO to be read or written. These bit values are used when the CPU reads from or writes to FLDTFIFO. In FLDTFIFO read, these bits specify the number of longwords of the data that can be read from FLDTFIFO. In FLDTFIFO write, these bits specify the number of longwords of unoccupied area that can be written in FLDTFIFO.
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 0	DTCNT[11:0]	H'000	R/W	Data Count Specification Specify the number of bytes of data to be read or written in command access mode. (Up to 2048 + 64 bytes for writing to and 128 bytes for reading from flash memory can be specified.)

### 28.3.7 Data Register (FLDATAR)

FLDATAR is a 32-bit readable/writable register. It stores input/output data used when 0 is written to the CDSRC bit in FLCMDCR in command access mode. FLDATAR cannot be used for reading or writing of five or more bytes of contiguous data.



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	DT4[7:0]	H'00	R/W	<b>Fourth Data</b> Specify the 4th data to be input or output via the NAF7 to NAF0 pins. In write: Specify write data In read: Store read data
23 to 16	DT3[7:0]	H'00	R/W	<b>Third Data</b> Specify the 3rd data to be input or output via the NAF7 to NAF0 pins. In write: Specify write data In read: Store read data
15 to 8	DT2[7:0]	H'00	R/W	<b>Second Data</b> Specify the 2nd data to be input or output via the NAF7 to NAF0 pins. In write: Specify write data In read: Store read data
7 to 0	DT1[7:0]	H'00	R/W	<b>First Data</b> Specify the 1st data to be input or output via the NAF7 to NAF0 pins. In write: Specify write data In read: Store read data



### 28.3.8 Interrupt DMA Control Register (FLINTDMACR)

FLINTDMACR is a 32-bit readable/writable register that enables or disables DMA transfer requests or interrupts. A transfer request from this module to the direct memory access controller is issued after each access mode has been started.

Bits 9 to 5 are the flag bits that indicate various errors occurred in flash memory access and whether there is a transfer request from the FIFO. Only 0 can be written to these bits. To clear a flag, write 0 to the target flag bit and 1 to the other flag bits.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	FIFOTRG [1:0]	-	AC0 CLR	-	DREQ0 EN	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R/W	R	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	ST ERB	BTO ERB	-	TRR EQFO	STER INTE	RBER INTE	TE INTE	-	TR INTE0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/(W)* R/(W)*	R	R	R/(W)*	R/W	R/W	R/W	R	R/W

Note: \* Only 0 can be written to these bits.

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
21, 20	FIFOTRG[1:0]	00	R/W	FIFO Trigger Setting Specify the condition (as a number of bytes) for generation of FLDTFIFO transfer requests. <ul style="list-style-type: none"> <li>In flash-memory read Issue an interrupt to the CPU or issue a DMA transfer request when FLDTFIFO stores the following number of bytes or more: <ul style="list-style-type: none"> <li>00: 4</li> <li>01: 16</li> <li>10: 128</li> <li>11: 128</li> </ul> </li> <li>In flash-memory programming Issue an interrupt to the CPU or issue a DMA transfer request when FLDTFIFO has the following empty area of bytes or more: <ul style="list-style-type: none"> <li>00: 4</li> <li>01: 16</li> <li>10: 128</li> <li>11: 128</li> </ul> </li> </ul> Note: For DMA transfer from/to FLDTFIFO, setting 10 and 11 are prohibited.
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
18	AC0CLR	0	R/W	FLDTFIFO Clear Clears FLDTFIFO. 0: Retains the FLDTFIFO value. In flash-memory access, this bit should be cleared to 0. 1: Clears FLDTFIFO. After FLDTFIFO has been cleared, this bit should be cleared to 0. Note: When FLDTFIFO is to be used in reading data from flash memory, be sure to clear FLDTFIFO before the reading starts.
17	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
16	DREQ0EN	0	R/W	FLDTFIFODMA Request Enable Enables or disables the DMA transfer request issued from FLDTFIFO. 0: Disables the DMA transfer request issued from the FLDTFIFO 1: Enables the DMA transfer request issued from the FLDTFIFO
15 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
8	STERB	0	R/(W)*	<p>Status Error</p> <p>Indicates the result of status read. This bit is set to 1 if the specific bit in the bits STAT[7:0] in FLBSYCNTR is set to 1 in status read.</p> <p>This bit is a flag. 1 cannot be written to this bit. Only 0 can be written to clear the flag.</p> <p>0: Indicates that no status error occurs (the specific bit in the bits STAT[7:0] in FLBSYCNTR is 0.)</p> <p>1: Indicates that a status error occurs</p> <p>For details on the specific bit in STAT7 to STAT0 bits, see section 28.4.4, Status Read.</p>
7	BTOERB	0	R/(W)*	<p>R/B Timeout Error</p> <p>This bit is set to 1 if an R/B timeout error occurs (the bits RBTIMCNT[19:0] in FLBSYCNTR are decremented to 0).</p> <p>This bit is a flag. 1 cannot be written to this bit. Only 0 can be written to clear the flag.</p> <p>0: Indicates that no R/B timeout error occurs</p> <p>1: Indicates that an R/B timeout error occurs</p>
6	—	0	R	<p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p>
5	TRREQF0	0	R/(W)*	<p>FLDTFIFO Transfer Request Flag</p> <p>Indicates that a transfer request is issued from FLDTFIFO.</p> <p>This bit is a flag. 1 cannot be written to this bit. Only 0 can be written to clear the flag.</p> <p>0: Indicates that no transfer request is issued from FLDTFIFO</p> <p>1: Indicates that a transfer request is issued from FLDTFIFO</p>
4	STERINTE	0	R/W	<p>Interrupt Enable at Status Error</p> <p>Enables or disables an interrupt request to the CPU when a status error has occurred.</p> <p>0: Disables the interrupt request to the CPU by a status error</p> <p>1: Enables the interrupt request to the CPU by a status error</p>
3	RBERINTE	0	R/W	<p>Interrupt Enable at R/B Timeout Error</p> <p>Enables or disables an interrupt request to the CPU when a timeout error has occurred.</p> <p>0: Disables the interrupt request to the CPU by an R/B timeout error</p> <p>1: Enables the interrupt request to the CPU by an R/B timeout error</p>
2	TEINTE	0	R/W	<p>Transfer End Interrupt Enable</p> <p>Enables or disables an interrupt request to the CPU when a transfer has been ended (TREND bit in FLTRCR).</p> <p>0: Disables the transfer end interrupt request to the CPU</p> <p>1: Enables the transfer end interrupt request to the CPU</p>
1	—	0	R	<p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p>
0	TRINTE0	0	R/W	<p>FLDTFIFO Transfer Request Enable to CPU</p> <p>Enables or disables an interrupt request to the CPU by a transfer request issued from FLDTFIFO.</p> <p>0: Disables an interrupt request to the CPU by a transfer request from FLDTFIFO</p> <p>1: Enables an interrupt request to the CPU by a transfer request from FLDTFIFO</p> <p>When the DMA transfer is enabled, this bit should be cleared to 0.</p>

Note: \* Only 0 can be written to these bits.

### 28.3.9 Ready Busy Timeout Setting Register (FLBSYTMR)

FLBSYTMR is a 32-bit readable/writable register that specifies the timeout time when the FRB pin is busy.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	RBTMOUT[19:16]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RBTMOUT[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
19 to 0	RBTMOUT [19:0]	H'00000	R/W	Ready Busy Timeout Specify timeout time (the number of P0φ clocks) in busy state. When these bits are set to 0, timeout is not generated.

### 28.3.10 Ready Busy Timeout Counter (FLBSYCNT)

FLBSYCNT is a 32-bit read-only register.

The status of flash memory obtained by the status read is stored in the bits STAT[7:0].

The timeout time set in the bits RBTMOUT[19:0] in FLBSYTMR is copied to the bits RBTMCNT[19:0] and counting down is started when the FRB pin is placed in a busy state. When values in the RBTMCNT[19:0] become 0, 1 is set to the BTOERB bit in FLINTDMACR, thus notifying that a timeout error has occurred. In this case, an FLSTE interrupt request can be issued if an interrupt is enabled by the RBERINTE bit in FLINTDMACR.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	STAT[7:0]							-	-	-	-	RBTMCNT[19:16]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RBTMCNT[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

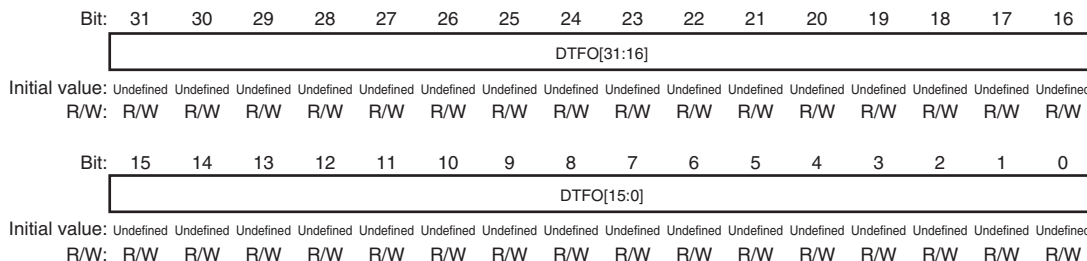
Bit	Bit Name	Initial Value	R/W	Description
31 to 24	STAT[7:0]	H'00	R	Indicate the flash memory status obtained by the status read.
23 to 20	—	All 0	R	Reserved These bits are always read as 0.
19 to 0	RBTMCNT [19:0]	H'00000	R	Ready Busy Timeout Counter When the FRB pin is placed in a busy state, the values of the bits RBTMOUT[19:0] in FLBSYTMR are copied to these bits. These bits are counted down while the FRB pin is busy. A timeout error occurs when these bits are decremented to 0.

### 28.3.11 Data FIFO Register (FLDTFIFO)

FLDTFIFO is used to read or write the data FIFO area.

In DMA transfer, this register must be specified as the destination or source.

Note that the direction of read or write specified by the SELRW bit in FLCMDCR must match that specified in this register. When changing the read/write direction, FLDTFIFO should be cleared by setting the AC0CLR bit in FLINTDMACR before use.



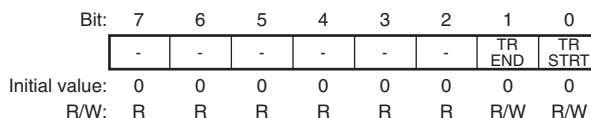
Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DTFO[31:0]	H'xxxxxxxx	R/W	Data FIFO Area Read/Write Data In write: Data in this register is written to the data FIFO area. In read: Data read from the data FIFO area is stored in this register.

### 28.3.12 Transfer Control Register (FLTRCR)

Setting the TRSTRT bit to 1 initiates access to flash memory. Access completion can be checked by the TREND bit.

During the transfer (from when the TRSTRT bit is set to 1 until the TREND bit is set to 1), the processing should not be forcibly ended (by setting the TRSTRT bit to 0).

When reading from flash memory, TREND is set when reading from flash memory have been finished. However, if there is any read data remaining in the FIFO, the processing should not be forcibly ended until all data has been read from the FIFO.



Bit	Bit Name	Initial Value	R/W	Description
7 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	TREND	0	R/W	Processing End Flag Bit Indicates that the processing performed in the specified access mode has been completed. The write value should always be 0.
0	TRSTRT	0	R/W	Transfer Start By setting this bit from 0 to 1 when the TREND bit is 0, processing in the access mode specified by the access mode specification bits ACM[1:0] is initiated. 0: Stops transfer 1: Starts transfer

## 28.4 Operation

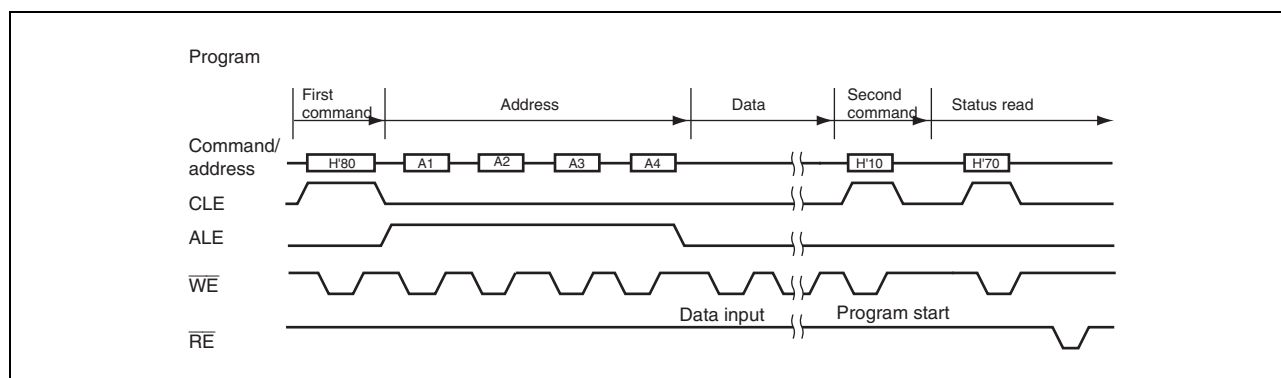
### 28.4.1 Access Sequence

This module performs accesses in several independent stages.

For example, NAND-type flash memory programming consists of the following five stages.

- First command issue stage (program setup command)
- Address issue stage (program address)
- Data stage (output)
- Second command issue stage (program start command)
- Status read stage

NAND-type flash memory programming access is achieved by executing these five stages sequentially. An access to flash memory is completed at the end of the final stage (status read stage).



**Figure 28.2 Programming Operation for NAND-Type Flash Memory and Stages**

For details on NAND-type flash memory read operation, see section 28.4.3, Command Access Mode.

### 28.4.2 Register Setting Procedure

Figure 28.3 shows the register setting flow required for accessing the flash memory.

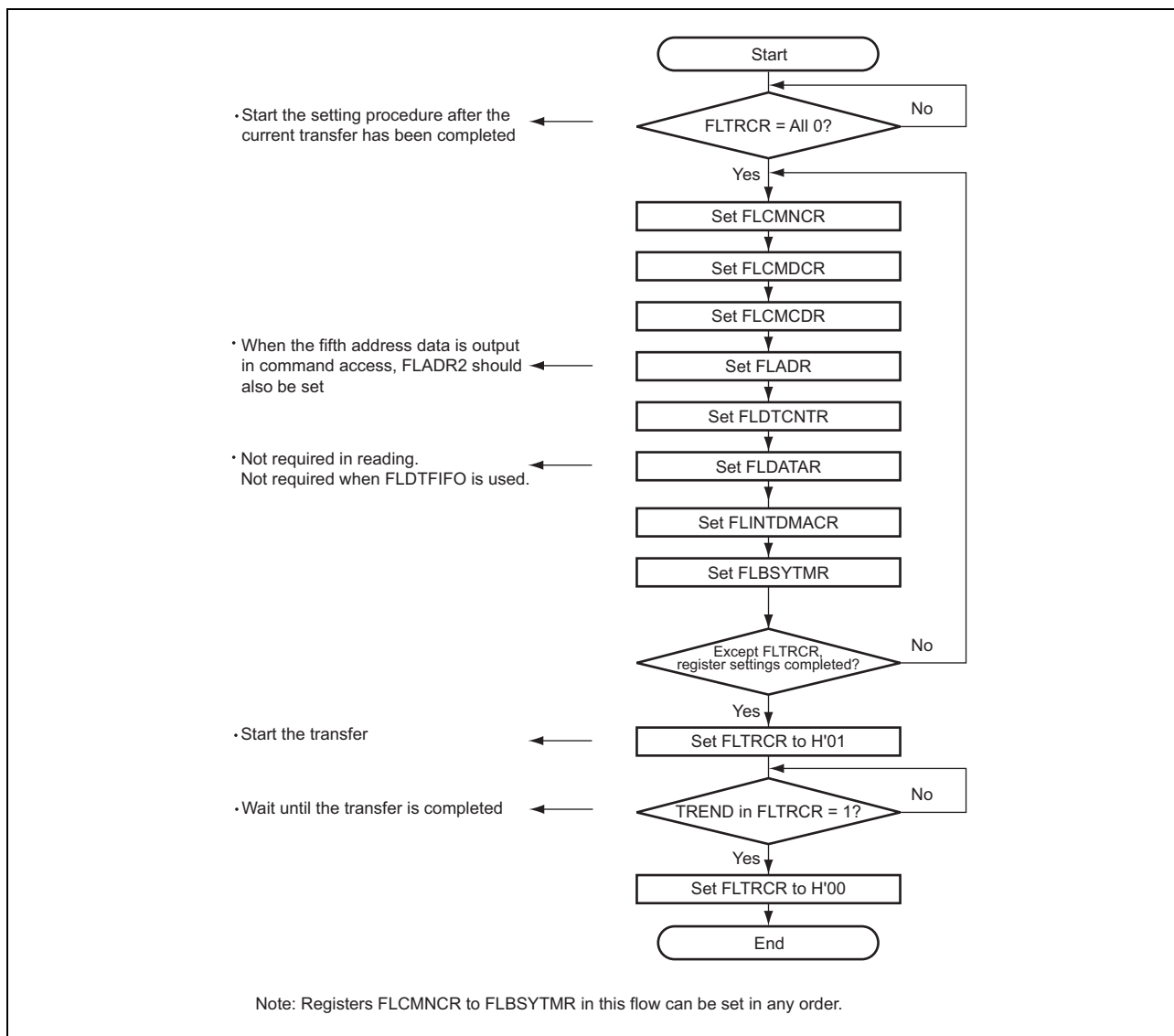


Figure 28.3 Register Setting Flow

### 28.4.3 Command Access Mode

Command access mode accesses flash memory by specifying a command to be issued to flash memory, address, data, read/write direction, and number of times to the registers. In this mode, I/O data can be transferred by the DMA via FLDTFIFO.

#### (1) NAND-Type Flash Memory Access

Figure 28.4 shows an example of read operation for NAND-type flash memory. In this example, the first command is specified as H'00, address data length is specified as 3 bytes, and the number of read bytes is specified as 8 bytes in the data counter.

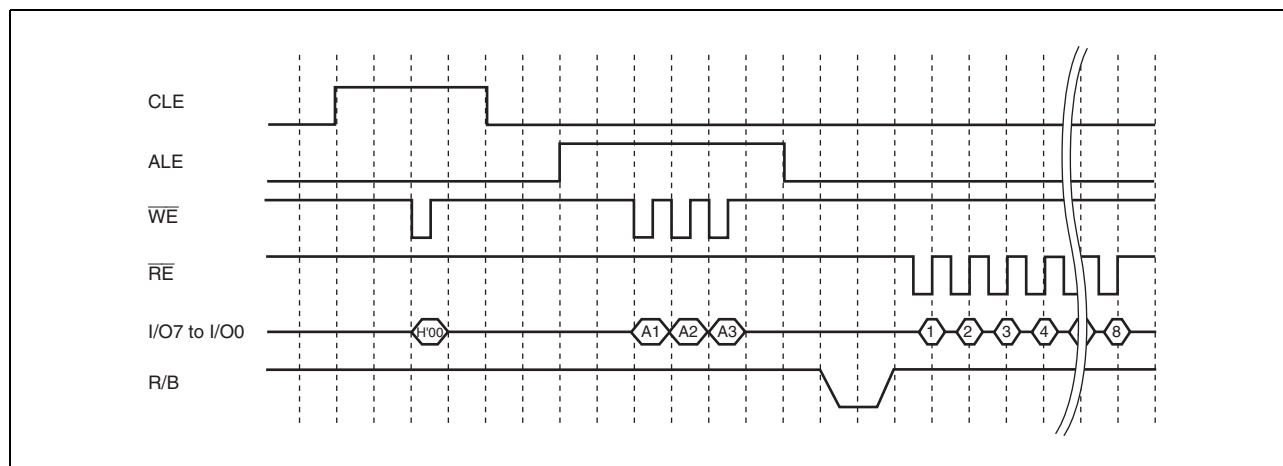


Figure 28.4 Read Operation Timing for NAND-Type Flash Memory

Figure 28.5 and Figure 28.6 show examples of programming operation for NAND-type flash memory.

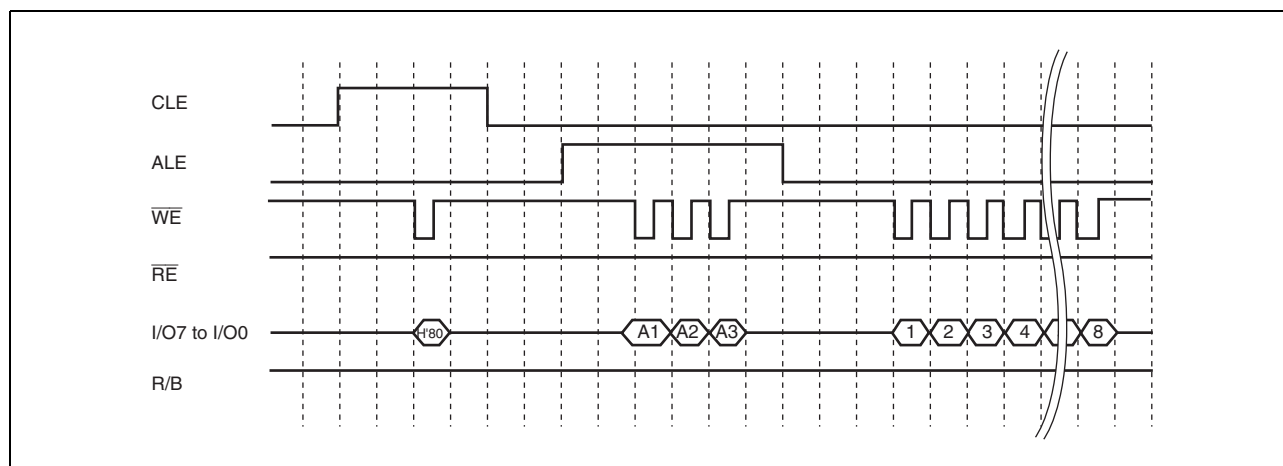
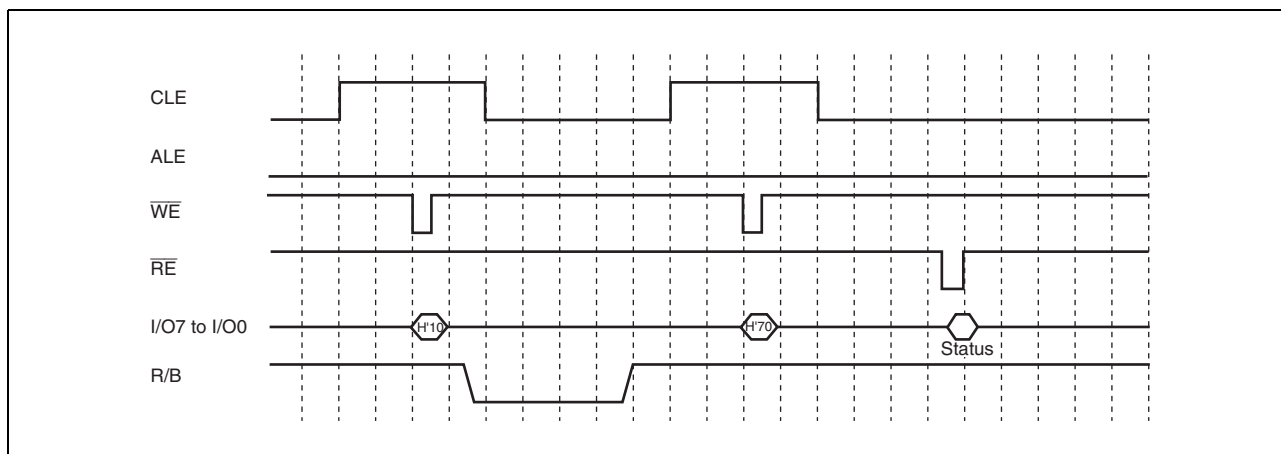


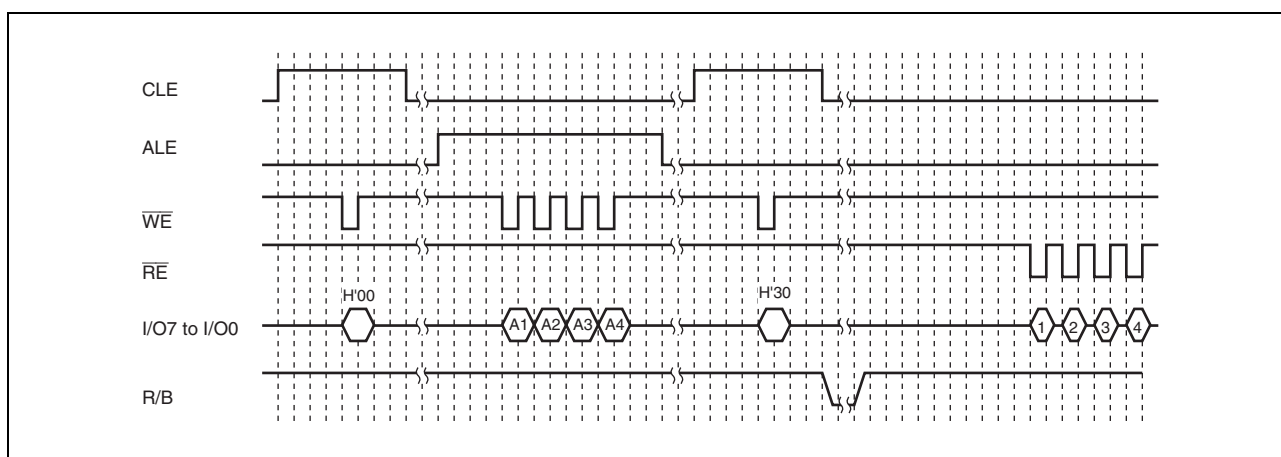
Figure 28.5 Programming Operation Timing for NAND-Type Flash Memory (1)



**Figure 28.6 Programming Operation Timing for NAND-Type Flash Memory (2)**

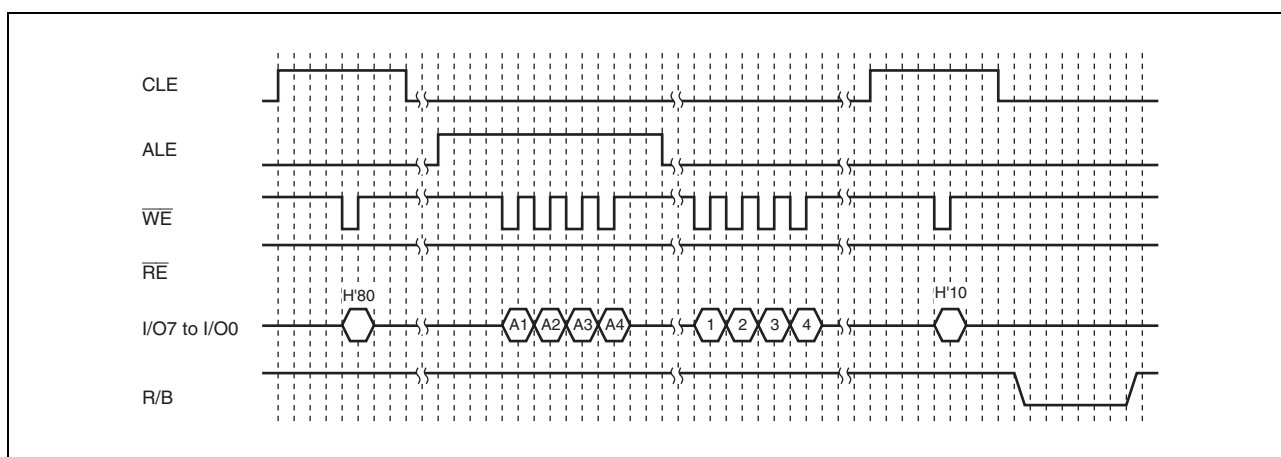
**(2) NAND-Type Flash Memory (2048 + 64 Bytes) Access**

Figure 28.7 shows an example of read operation for NAND-type flash memory (2048 + 64 bytes). In this example, the first command is specified as H'00, the second command is specified as H'30, and address data length is specified as 4 bytes. The number of read bytes is specified as 4 bytes in the data counter.



**Figure 28.7 Read Operation Timing for NAND-Type Flash Memory**

Figure 28.8 and Figure 28.9 show examples of programming operation for NAND-type flash memory (2048 + 64 bytes).



**Figure 28.8 Programming Operation Timing for NAND-Type Flash Memory (1)**



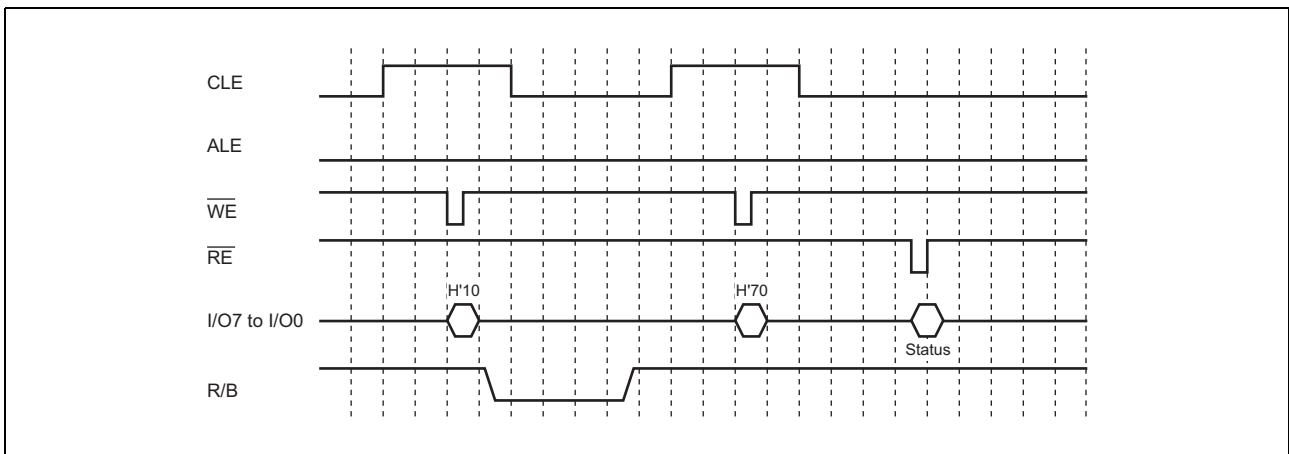


Figure 28.9 Programming Operation Timing for NAND-Type Flash Memory (2)

### 28.4.4 Status Read

This module can read the status register of a NAND-type flash memory. The data in the status register is input through the I/O7 to I/O0 pins and stored in the bits STAT[7:0] in FLBSYCNT, which can be read by the CPU. If a program error or erase error is detected when the status register value is stored in the bits STAT[7:0] in FLBSYCNT, the STERB bit in FLINTDMACR is set to 1 and generates an interrupt to the CPU if the STERINTE bit in FLINTDMACR is enabled. If a status error occurs during continuous sector access, the TREND bit in FLTRCR is set to 1 and the procedure stops.

#### (1) Status Read of NAND-Type Flash Memory

The status register of NAND-type flash memory can be read by inputting command H'70 to NAND-type flash memory. If programming is executed in command access mode or sector access mode while the DOSR bit in FLCMDCR is set to 1, this module automatically inputs command H'70 to NAND-type flash memory and reads the status register. When the status register of NAND-type flash memory is read, the I/O7 to I/O0 pins indicate the following information as described in Table 28.3.

**Table 28.3 Status Read of NAND-Type Flash Memory**

I/O	Status (definition)	Description
I/O7	Program protection	0: Cannot be programmed 1: Can be programmed
I/O6	Ready/busy	0: Busy state 1: Ready state
I/O5 to I/O1	Reserved	—
I/O0	Program/erase	0: Pass 1: Fail

## 28.5 Interrupt Processing

This module has four interrupt sources: Status error, ready/busy timeout error, transfer end, and FIFO0 transfer request. Each of the interrupt sources has its corresponding interrupt flag and the interrupt can be requested independently to the CPU if the interrupt is enabled by the interrupt enable bit. Note that the status error and ready/busy timeout error use the common FLSTE interrupt to the CPU.

**Table 28.4 Interrupt Requests**

Interrupt Source	Interrupt Flag	Enable Bit	Description	Priority
FLSTE interrupt	STERB	STERINTE	Status error	High ↑
	BTOERB	RBERINTE	Ready/busy timeout error	
FLTEND interrupt	TREND	TEINTE	Transfer end	↓ Low
FLTRQ0 interrupt	TRREQF0	TRINTE0	FIFO0 transfer request	

## 28.6 DMA Transfer Settings

For details on settings of the direct memory access controller, see section 9, Direct Memory Access Controller.

## 28.7 Usage Notes

### 28.7.1 Usage Note for the SNAND Bit

When the SNAND bit in FLCMNCR is 1, both the first and second commands are executed regardless of the settings of the DOCMD1 and DOCMD2 bits in FLCMDCR.

When no command or only the first command is issued, 0 should be written to the SNAND bit.

## 29. USB2.0 Host/Function Module

This LSI includes a two-channel USB 2.0 host/function module. This module is a USB controller which provides capabilities as a USB host controller and USB function controller.

This module supports high-speed transfer, full-speed transfer, and low-speed transfer defined by USB (Universal Serial Bus) Specification 2.0, when used as the host controller, and supports high-speed transfer and full-speed transfer defined by USB (Universal Serial Bus) Specification 2.0, when used as the function controller.

This module supports all of the transfer types defined by the USB Specification. This module has an 8-Kbyte buffer memory for data transfer, providing a maximum of 16 pipes. Any endpoint numbers can be assigned to PIPE1 to PIPE15, based on the peripheral devices or user system for communication.

### 29.1 Features

#### (1) Host Controller and Function Controller Supporting USB High-Speed Operation

- The USB host controller and USB function controller are incorporated.
- The USB host controller and USB function controller can be switched by register settings.
- On-chip USB transceiver

#### (2) All Types of USB Transfers Supported

All types of USB transfers including isochronous transfer are supported.

- Control transfer
- Bulk transfer
- Interrupt transfer (high bandwidth transfers not supported)
- Isochronous transfer (high bandwidth transfers not supported)

#### (3) Internal Bus Interfaces

- Two DMA interfaces available for each channel

#### (4) Pipe Configuration

- Up to 8 Kbytes of buffer memory for USB communications are supported for each channel
- Up to sixteen pipes can be selected per channel (including the default control pipe)
- Programmable pipe configuration
- Endpoint numbers can be assigned flexibly to PIPE1 to PIPE15.
- Transfer conditions that can be set for each pipe:
  - PIPE0: Control transfer, 256-byte fixed single buffer
  - PIPE1 and PIPE2: Bulk transfer or isochronous transfer can be selected, continuous transfer mode, programmable buffer size (up to 2 Kbytes: double buffer can be specified)
  - PIPE3 to PIPE5: Bulk transfer, continuous transfer mode, programmable buffer size (up to 2 Kbytes: double buffer can be specified)
  - PIPE6 to PIPE8: Interrupt transfer, 64-byte fixed single buffer
  - PIPE9: Bulk transfer (only when the function controller mode is selected)  
Interrupt transfer (only when the host controller mode is selected), programmable buffer size (up to 2 Kbytes: double buffer can be specified only when bulk transfer has been selected)
  - PIPE10: Bulk transfer or interrupt transfer can be selected (only when the function controller mode is selected), programmable buffer size (up to 2 Kbytes: double buffer can be specified only when bulk transfer has been selected)

PIPE11 to PIPE15: Bulk transfer (only when the function controller mode is selected), programmable buffer size (up to 2 Kbytes: double buffer can be specified)

#### (5) Features of the USB Host Controller

- High-speed transfer (480 Mbps), full-speed transfer (12 Mbps), and low-speed transfer (1.5 Mbps) are supported.
- Communications with multiple peripheral devices connected via a single HUB
- Automatic response to the reset handshake
- Automatic scheduling for SOF and packet transmissions
- Programmable intervals for isochronous and interrupt transfers

#### (6) Features of the USB Function Controller

- High-speed transfer (480 Mbps) and full-speed transfer (12 Mbps) are supported.
- Automatic recognition of high-speed operation or full-speed operation based on automatic response to the reset handshake
- Control transfer stage monitoring function
- Device state monitoring function
- Auto response function for SET\_ADDRESS request
- NAK response interrupt function (NRDY)
- SOF interpolation function

#### (7) Other Features

- Transfer ending function using transaction count
- BRDY interrupt event notification timing change function (BFRE)
- Function that automatically clears the buffer memory after the data for the pipe specified at the Dx FIFO port has been read (DCLRM)
- NAK setting function for response PID generated by end of transfer (SHTNAK)

## 29.2 Input/Output Pins

Table 29.1 shows the pin configuration.

**Table 29.1 Pin Configuration**

Channel	Name	Abbreviation	I/O	Function
0	USB D+ data	DP0	Input/ Output	The D+ input/output pin of the on-chip USB transceiver Connect this pin to the D+ pin of the USB bus.
	USB D- data	DM0	Input/ Output	The D- input/output pin of the on-chip USB transceiver Connect this pin to the D- pin of the USB bus.
	VBUS input	VBUS0	Input	A pin for monitoring connection of the USB cable Connect this pin to the Vbus pin of the USB bus so that it can detect connection and disconnection of the Vbus pin. When it is not connected to the Vbus pin of the USB, It should be fixed to 5 V. Also supply 5 V to this pin when the host controller mode is selected. Note: This module is not capable of supplying Vbus power to connected peripheral devices.
1	USB D+ data	DP1	Input/ Output	The D+ input/output pin of the on-chip USB transceiver Connect this pin to the D+ pin of the USB bus.
	USB D- data	DM1	Input/ Output	The D- input/output pin of the on-chip USB transceiver Connect this pin to the D- pin of the USB bus.
	VBUS input	VBUS1	Input	A pin for monitoring connection of the USB cable Connect this pin to the Vbus pin of the USB bus so that it can detect connection and disconnection of the Vbus pin. When it is not connected to the Vbus pin of the USB, It should be fixed to 5 V. Also supply 5 V to this pin when the host controller mode is selected. Note: This module is not capable of supplying Vbus power to connected peripheral devices.
Common	Reference input	REFRIN	Input	A pin for connecting the reference resistor Connect this pin to the USBAPVss pin through a resistor with a value of $5.6\text{ k}\Omega \pm 1\%$ .
	USB crystal oscillator/external clock	USB_X1	Input	Connect this pin to the crystal oscillator for USB. An external clock can also be input to the USB_X1 pin.
		USB_X2	Output	
	Transceiver digital core power*	USBDVcc	Input	Power supply for the module's digital core
	Transceiver digital core ground*	USBDVss	Input	Ground for the module's digital core
	Transceiver digital pin power source*	USBDPVcc	Input	Power supply for pins
	Transceiver digital pin ground*	USBDPVss	Input	Ground for pins
	Transceiver analog pin power source	USBAPVcc	Input	Power supply for pins
	Transceiver analog pin ground*	USBAPVss	Input	Ground for pins
	Transceiver analog core power	USBAVcc	Input	Power supply for the module's analog core
Transceiver analog core ground*	USBAVss	Input	Ground for the module's analog core	
Power source for the UTMI module*	USBVcc	Input	Power supply for operation at 480 MHz	
Ground for the UTMI module*	USBVss	Input	Power supply for operation at 480 MHz	

Note: \* These pins are not available in the BGA package products.

## 29.3 Register Descriptions

Table 29.2 shows the register configuration of this module.

**Table 29.2 Register Configuration**

Channel	Register Name	Abbreviation	R/W	Address	Access Size
0	System configuration control register_0	SYSCFG0_0	R/W	H'E801 0000	16
	CPU bus wait setting register_0	BUSWAIT_0	R/W	H'E801 0002	16
	System configuration status register_0	SYSSTS0_0	R	H'E801 0004	16
	Device state control register 0_0	DVSTCTR0_0	R/W	H'E801 0008	16
	Test mode register_0	TESTMODE_0	R/W	H'E801 000C	16
	DMA0-FIFO bus configuration register_0	D0FBCFG_0	R/W	H'E801 0010	16
	DMA1-FIFO bus configuration register_0	D1FBCFG_0	R/W	H'E801 0012	16
	CFIFO port register_0	CFIFO_0	R/W	H'E801 0014	8, 16, 32
	D0FIFO port register_0	D0FIFO_0	R/W	H'E801 0018	8, 16, 32
	D1FIFO port register_0	D1FIFO_0	R/W	H'E801 001C	8, 16, 32
	CFIFO port select register_0	CFIFOSEL_0	R/W	H'E801 0020	16
	CFIFO port control register_0	CFIFOCTR_0	R/W	H'E801 0022	16
	D0FIFO port select register_0	D0FIFOSEL_0	R/W	H'E801 0028	16
	D0FIFO port control register_0	D0FIFOCTR_0	R/W	H'E801 002A	16
	D1FIFO port select register_0	D1FIFOSEL_0	R/W	H'E801 002C	16
	D1FIFO port control register_0	D1FIFOCTR_0	R/W	H'E801 002E	16
	Interrupt enable register 0_0	INTENB0_0	R/W	H'E801 0030	16
	Interrupt enable register 1_0	INTENB1_0	R/W	H'E801 0032	16
	BRDY interrupt enable register_0	BRDYENB_0	R/W	H'E801 0036	16
	NRDY interrupt enable register_0	NRDYENB_0	R/W	H'E801 0038	16
	BEMP interrupt enable register_0	BEMPENB_0	R/W	H'E801 003A	16
	SOF output configuration register_0	SOFCFG_0	R/W	H'E801 003C	16
	Interrupt status register 0_0	INTSTS0_0	R/W	H'E801 0040	16
	Interrupt status register 1_0	INTSTS1_0	R/W	H'E801 0042	16
	BRDY interrupt status register_0	BRDYSTS_0	R/W	H'E801 0046	16
	NRDY interrupt status register_0	NRDYSTS_0	R/W	H'E801 0048	16
	BEMP interrupt status register_0	BEMPSTS_0	R/W	H'E801 004A	16
	Frame number register_0	FRMNUM_0	R/W	H'E801 004C	16
	μFrame number register_0	UFRMNUM_0	R	H'E801 004E	16
	USB address register_0	USBADDR_0	R	H'E801 0050	16
	USB request type register_0	USBREQ_0	R/W	H'E801 0054	16
	USB request value register_0	USBVAL_0	R/W	H'E801 0056	16
	USB request index register_0	USBINDX_0	R/W	H'E801 0058	16
	USB request length register_0	USBLENG_0	R/W	H'E801 005A	16
	DCP configuration register_0	DCPCFG_0	R/W	H'E801 005C	16
	DCP maximum packet size register_0	DCPMAXP_0	R/W	H'E801 005E	16



Table 29.2 Register Configuration

Channel	Register Name	Abbreviation	R/W	Address	Access Size
0	DCP control register_0	DCPCTR_0	R/W	H'E801 0060	16
	Pipe window select register_0	PIPESEL_0	R/W	H'E801 0064	16
	Pipe configuration register_0	PIPECFG_0	R/W	H'E801 0068	16
	Pipe buffer setting register_0	PIPEBUF_0	R/W	H'E801 006A	16
	Pipe maximum packet size register_0	PIPEMAXP_0	R/W	H'E801 006C	16
	Pipe timing control register_0	PIPEPERI_0	R/W	H'E801 006E	16
	Pipe 1 control register_0	PIPE1CTR_0	R/W	H'E801 0070	16
	Pipe 2 control register_0	PIPE2CTR_0	R/W	H'E801 0072	16
	Pipe 3 control register_0	PIPE3CTR_0	R/W	H'E801 0074	16
	Pipe 4 control register_0	PIPE4CTR_0	R/W	H'E801 0076	16
	Pipe 5 control register_0	PIPE5CTR_0	R/W	H'E801 0078	16
	Pipe 6 control register_0	PIPE6CTR_0	R/W	H'E801 007A	16
	Pipe 7 control register_0	PIPE7CTR_0	R/W	H'E801 007C	16
	Pipe 8 control register_0	PIPE8CTR_0	R/W	H'E801 007E	16
	Pipe 9 control register_0	PIPE9CTR_0	R/W	H'E801 0080	16
	Pipe A control register_0	PIPEACTR_0	R/W	H'E801 0082	16
	Pipe B control register_0	PIPEBCTR_0	R/W	H'E801 0084	16
	Pipe C control register_0	PIPECCTR_0	R/W	H'E801 0086	16
	Pipe D control register_0	PIPEDCTR_0	R/W	H'E801 0088	16
	Pipe E control register_0	PIPEECTR_0	R/W	H'E801 008A	16
	Pipe F control register_0	PIPEFCTR_0	R/W	H'E801 008C	16
	Pipe 1 transaction counter enable register_0	PIPE1TRE_0	R/W	H'E801 0090	16
	Pipe 1 transaction counter register_0	PIPE1TRN_0	R/W	H'E801 0092	16
	Pipe 2 transaction counter enable register_0	PIPE2TRE_0	R/W	H'E801 0094	16
	Pipe 2 transaction counter register_0	PIPE2TRN_0	R/W	H'E801 0096	16
	Pipe 3 transaction counter enable register_0	PIPE3TRE_0	R/W	H'E801 0098	16
	Pipe 3 transaction counter register_0	PIPE3TRN_0	R/W	H'E801 009A	16
	Pipe 4 transaction counter enable register_0	PIPE4TRE_0	R/W	H'E801 009C	16
	Pipe 4 transaction counter register_0	PIPE4TRN_0	R/W	H'E801 009E	16
	Pipe 5 transaction counter enable register_0	PIPE5TRE_0	R/W	H'E801 00A0	16
	Pipe 5 transaction counter register_0	PIPE5TRN_0	R/W	H'E801 00A2	16
	Pipe B transaction counter enable register_0	PIPEBTRE_0	R/W	H'E801 00A4	16
	Pipe B transaction counter register_0	PIPEBTRN_0	R/W	H'E801 00A6	16
	Pipe C transaction counter enable register_0	PIPECTRE_0	R/W	H'E801 00A8	16
	Pipe C transaction counter register_0	PIPECTRN_0	R/W	H'E801 00AA	16
	Pipe D transaction counter enable register_0	PIPEDTRE_0	R/W	H'E801 00AC	16
	Pipe D transaction counter register_0	PIPEDTRN_0	R/W	H'E801 00AE	16
	Pipe E transaction counter enable register_0	PIPEETRE_0	R/W	H'E801 00B0	16
	Pipe E transaction counter register_0	PIPEETR_0	R/W	H'E801 00B2	16
	Pipe F transaction counter enable register_0	PIPEFTRE_0	R/W	H'E801 00B4	16
	Pipe F transaction counter register_0	PIPEFTRN_0	R/W	H'E801 00B6	16
	Pipe 9 transaction counter enable register_0	PIPE9TRE_0	R/W	H'E801 00B8	16
	Pipe 9 transaction counter register_0	PIPE9TRN_0	R/W	H'E801 00BA	16

**Table 29.2 Register Configuration**

Channel	Register Name	Abbreviation	R/W	Address	Access Size
0	Pipe A transaction counter enable register_0	PIPEATRE_0	R/W	H'E801 00BC	16
	Pipe A transaction counter register_0	PIPEATR_N_0	R/W	H'E801 00BE	16
	Device address 0 configuration register_0	DEVADD0_0	R/W	H'E801 00D0	16
	Device address 1 configuration register_0	DEVADD1_0	R/W	H'E801 00D2	16
	Device address 2 configuration register_0	DEVADD2_0	R/W	H'E801 00D4	16
	Device address 3 configuration register_0	DEVADD3_0	R/W	H'E801 00D6	16
	Device address 4 configuration register_0	DEVADD4_0	R/W	H'E801 00D8	16
	Device address 5 configuration register_0	DEVADD5_0	R/W	H'E801 00DA	16
	Device address 6 configuration register_0	DEVADD6_0	R/W	H'E801 00DC	16
	Device address 7 configuration register_0	DEVADD7_0	R/W	H'E801 00DE	16
	Device address 8 configuration register_0	DEVADD8_0	R/W	H'E801 00E0	16
	Device address 9 configuration register_0	DEVADD9_0	R/W	H'E801 00E2	16
	Device address A configuration register_0	DEVADDA_0	R/W	H'E801 00E4	16
	Suspend mode register_0	SUSPMODE_0	R/W	H'E801 0102	16
	D0FIFO continuous transfer port register 0_0	D0FIFOB0_0	R/W	H'E801 0160	32
	D0FIFO continuous transfer port register 1_0	D0FIFOB1_0	R/W	H'E801 0164	32
	D0FIFO continuous transfer port register 2_0	D0FIFOB2_0	R/W	H'E801 0168	32
	D0FIFO continuous transfer port register 3_0	D0FIFOB3_0	R/W	H'E801 016C	32
	D0FIFO continuous transfer port register 4_0	D0FIFOB4_0	R/W	H'E801 0170	32
	D0FIFO continuous transfer port register 5_0	D0FIFOB5_0	R/W	H'E801 0174	32
	D0FIFO continuous transfer port register 6_0	D0FIFOB6_0	R/W	H'E801 0178	32
	D0FIFO continuous transfer port register 7_0	D0FIFOB7_0	R/W	H'E801 017C	32
	D1FIFO continuous transfer port register 0_0	D1FIFOB0_0	R/W	H'E801 0180	32
	D1FIFO continuous transfer port register 1_0	D1FIFOB1_0	R/W	H'E801 0184	32
	D1FIFO continuous transfer port register 2_0	D1FIFOB2_0	R/W	H'E801 0188	32
	D1FIFO continuous transfer port register 3_0	D1FIFOB3_0	R/W	H'E801 018C	32
	D1FIFO continuous transfer port register 4_0	D1FIFOB4_0	R/W	H'E801 0190	32
	D1FIFO continuous transfer port register 5_0	D1FIFOB5_0	R/W	H'E801 0194	32
	D1FIFO continuous transfer port register 6_0	D1FIFOB6_0	R/W	H'E801 0198	32
	D1FIFO continuous transfer port register 7_0	D1FIFOB7_0	R/W	H'E801 019C	32

**Table 29.2 Register Configuration**

Channel	Register Name	Abbreviation	R/W	Address	Access Size
1	System configuration control register_1	SYSCFG0_1	R/W	H'E820 7000	16
	CPU bus wait setting register_1	BUSWAIT_1	R/W	H'E820 7002	16
	System configuration status register_1	SYSSTS0_1	R	H'E820 7004	16
	Device state control register 0_1	DVSTCTR0_1	R/W	H'E820 7008	16
	Test mode register_1	TESTMODE_1	R/W	H'E820 700C	16
	DMA0-FIFO bus configuration register_1	D0FBCFG_1	R/W	H'E820 7010	16
	DMA1-FIFO bus configuration register_1	D1FBCFG_1	R/W	H'E820 7012	16
	CFIFO port register_1	CFIFO_1	R/W	H'E820 7014	8, 16, 32
	D0FIFO port register_1	D0FIFO_1	R/W	H'E820 7018	8, 16, 32
	D1FIFO port register_1	D1FIFO_1	R/W	H'E820 701C	8, 16, 32
	CFIFO port select register_1	CFIFOSEL_1	R/W	H'E820 7020	16
	CFIFO port control register_1	CFIFOCTR_1	R/W	H'E820 7022	16
	D0FIFO port select register_1	D0FIFOSEL_1	R/W	H'E820 7028	16
	D0FIFO port control register_1	D0FIFOCTR_1	R/W	H'E820 702A	16
	D1FIFO port select register_1	D1FIFOSEL_1	R/W	H'E820 702C	16
	D1FIFO port control register_1	D1FIFOCTR_1	R/W	H'E820 702E	16
	Interrupt enable register 0_1	INTENB0_1	R/W	H'E820 7030	16
	Interrupt enable register 1_1	INTENB1_1	R/W	H'E820 7032	16
	BRDY interrupt enable register_1	BRDYENB_1	R/W	H'E820 7036	16
	NRDY interrupt enable register_1	NRDYENB_1	R/W	H'E820 7038	16
	BEMP interrupt enable register_1	BEMPENB_1	R/W	H'E820 703A	16
	SOF output configuration register_1	SOFCFG_1	R/W	H'E820 703C	16
	Interrupt status register 0_1	INTSTS0_1	R/W	H'E820 7040	16
	Interrupt status register 1_1	INTSTS1_1	R/W	H'E820 7042	16
	BRDY interrupt status register_1	BRDYSTS_1	R/W	H'E820 7046	16
	NRDY interrupt status register_1	NRDYSTS_1	R/W	H'E820 7048	16
	BEMP interrupt status register_1	BEMPSTS_1	R/W	H'E820 704A	16
	Frame number register_1	FRMNUM_1	R/W	H'E820 704C	16
	μFrame number register_1	UFRMNUM_1	R	H'E820 704E	16

**Table 29.2 Register Configuration**

Channel	Register Name	Abbreviation	R/W	Address	Access Size
1	USB address register_1	USBADDR_1	R	H'E820 7050	16
	USB request type register_1	USBREQ_1	R/W	H'E820 7054	16
	USB request value register_1	USBVAL_1	R/W	H'E820 7056	16
	USB request index register_1	USBINDX_1	R/W	H'E820 7058	16
	USB request length register_1	USBLENG_1	R/W	H'E820 705A	16
	DCP configuration register_1	DCPCFG_1	R/W	H'E820 705C	16
	DCP maximum packet size register_1	DCPMAXP_1	R/W	H'E820 705E	16
	DCP control register_1	DCPCTR_1	R/W	H'E820 7060	16
	Pipe window select register_1	PIPESEL_1	R/W	H'E820 7064	16
	Pipe configuration register_1	PIPECFG_1	R/W	H'E820 7068	16
	Pipe buffer setting register_1	PIPEBUF_1	R/W	H'E820 706A	16
	Pipe maximum packet size register_1	PIPEMAXP_1	R/W	H'E820 706C	16
	Pipe timing control register_1	PIPEPERI_1	R/W	H'E820 706E	16
	Pipe 1 control register_1	PIPE1CTR_1	R/W	H'E820 7070	16
	Pipe 2 control register_1	PIPE2CTR_1	R/W	H'E820 7072	16
	Pipe 3 control register_1	PIPE3CTR_1	R/W	H'E820 7074	16
	Pipe 4 control register_1	PIPE4CTR_1	R/W	H'E820 7076	16
	Pipe 5 control register_1	PIPE5CTR_1	R/W	H'E820 7078	16
	Pipe 6 control register_1	PIPE6CTR_1	R/W	H'E820 707A	16
	Pipe 7 control register_1	PIPE7CTR_1	R/W	H'E820 707C	16
	Pipe 8 control register_1	PIPE8CTR_1	R/W	H'E820 707E	16
	Pipe 9 control register_1	PIPE9CTR_1	R/W	H'E820 7080	16
	Pipe A control register_1	PIPEACTR_1	R/W	H'E820 7082	16
	Pipe B control register_1	PIPEBCTR_1	R/W	H'E820 7084	16
	Pipe C control register_1	PIPECCTR_1	R/W	H'E820 7086	16
	Pipe D control register_1	PIPEDCTR_1	R/W	H'E820 7088	16
	Pipe E control register_1	PIPEECTR_1	R/W	H'E820 708A	16
	Pipe F control register_1	PIPEFCTR_1	R/W	H'E820 708C	16
	Pipe 1 transaction counter enable register_1	PIPE1TRE_1	R/W	H'E820 7090	16
	Pipe 1 transaction counter register_1	PIPE1TRN_1	R/W	H'E820 7092	16
	Pipe 2 transaction counter enable register_1	PIPE2TRE_1	R/W	H'E820 7094	16
	Pipe 2 transaction counter register_1	PIPE2TRN_1	R/W	H'E820 7096	16
	Pipe 3 transaction counter enable register_1	PIPE3TRE_1	R/W	H'E820 7098	16
	Pipe 3 transaction counter register_1	PIPE3TRN_1	R/W	H'E820 709A	16
	Pipe 4 transaction counter enable register_1	PIPE4TRE_1	R/W	H'E820 709C	16
	Pipe 4 transaction counter register_1	PIPE4TRN_1	R/W	H'E820 709E	16
	Pipe 5 transaction counter enable register_1	PIPE5TRE_1	R/W	H'E820 70A0	16
	Pipe 5 transaction counter register_1	PIPE5TRN_1	R/W	H'E820 70A2	16
	Pipe B transaction counter enable register_1	PIPEBTRE_1	R/W	H'E820 70A4	16
	Pipe B transaction counter register_1	PIPEBTRN_1	R/W	H'E820 70A6	16
	Pipe C transaction counter enable register_1	PIPECTRE_1	R/W	H'E820 70A8	16
	Pipe C transaction counter register_1	PIPECTRN_1	R/W	H'E820 70AA	16
	Pipe D transaction counter enable register_1	PIPEDTRE_1	R/W	H'E820 70AC	16

Table 29.2 Register Configuration

Channel	Register Name	Abbreviation	R/W	Address	Access Size
1	Pipe D transaction counter register_1	PIPEDTRN_1	R/W	H'E820 70AE	16
	Pipe E transaction counter enable register_1	PIPEETRE_1	R/W	H'E820 70B0	16
	Pipe E transaction counter register_1	PIPEETRN_1	R/W	H'E820 70B2	16
	Pipe F transaction counter enable register_1	PIPEFTRE_1	R/W	H'E820 70B4	16
	Pipe F transaction counter register_1	PIPEFTRN_1	R/W	H'E820 70B6	16
	Pipe 9 transaction counter enable register_1	PIPE9TRE_1	R/W	H'E820 70B8	16
	Pipe 9 transaction counter register_1	PIPE9TRN_1	R/W	H'E820 70BA	16
	Pipe A transaction counter enable register_1	PIPEATRE_1	R/W	H'E820 70BC	16
	Pipe A transaction counter register_1	PIPEATR_1	R/W	H'E820 70BE	16
	Device address 0 configuration register_1	DEVADD0_1	R/W	H'E820 70D0	16
	Device address 1 configuration register_1	DEVADD1_1	R/W	H'E820 70D2	16
	Device address 2 configuration register_1	DEVADD2_1	R/W	H'E820 70D4	16
	Device address 3 configuration register_1	DEVADD3_1	R/W	H'E820 70D6	16
	Device address 4 configuration register_1	DEVADD4_1	R/W	H'E820 70D8	16
	Device address 5 configuration register_1	DEVADD5_1	R/W	H'E820 70DA	16
	Device address 6 configuration register_1	DEVADD6_1	R/W	H'E820 70DC	16
	Device address 7 configuration register_1	DEVADD7_1	R/W	H'E820 70DE	16
	Device address 8 configuration register_1	DEVADD8_1	R/W	H'E820 70E0	16
	Device address 9 configuration register_1	DEVADD9_1	R/W	H'E820 70E2	16
	Device address A configuration register_1	DEVADDA_1	R/W	H'E820 70E4	16
	Suspend mode register_1	SUSPMODE_1	R/W	H'E820 7102	16
	D0FIFO continuous transfer port register 0_1	D0FIFOB0_1	R/W	H'E820 7160	32
	D0FIFO continuous transfer port register 1_1	D0FIFOB1_1	R/W	H'E820 7164	32
	D0FIFO continuous transfer port register 2_1	D0FIFOB2_1	R/W	H'E820 7168	32
	D0FIFO continuous transfer port register 3_1	D0FIFOB3_1	R/W	H'E820 716C	32
	D0FIFO continuous transfer port register 4_1	D0FIFOB4_1	R/W	H'E820 7170	32
	D0FIFO continuous transfer port register 5_1	D0FIFOB5_1	R/W	H'E820 7174	32
	D0FIFO continuous transfer port register 6_1	D0FIFOB6_1	R/W	H'E820 7178	32
	D0FIFO continuous transfer port register 7_1	D0FIFOB7_1	R/W	H'E820 717C	32
	D1FIFO continuous transfer port register 0_1	D1FIFOB0_1	R/W	H'E820 7180	32
	D1FIFO continuous transfer port register 1_1	D1FIFOB1_1	R/W	H'E820 7184	32
	D1FIFO continuous transfer port register 2_1	D1FIFOB2_1	R/W	H'E820 7188	32
D1FIFO continuous transfer port register 3_1	D1FIFOB3_1	R/W	H'E820 718C	32	
D1FIFO continuous transfer port register 4_1	D1FIFOB4_1	R/W	H'E820 7190	32	
D1FIFO continuous transfer port register 5_1	D1FIFOB5_1	R/W	H'E820 7194	32	
D1FIFO continuous transfer port register 6_1	D1FIFOB6_1	R/W	H'E820 7198	32	
D1FIFO continuous transfer port register 7_1	D1FIFOB7_1	R/W	H'E820 719C	32	

### 29.3.1 System Configuration Control Register (SYSCFG0)

SYSCFG is a register that enables high-speed operation, selects the host controller mode or function controller mode, controls the DP and DM pins, and enables operation of this module.

This register is initialized by a power-on reset.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	HSE	DCFM	DRPD	DPRPU	—	UCK SEL	UPLLE	USBE
Initial value:	—	—	—	—	—	—	—	0	0	0	1	0	—	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 9	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
8	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
7	HSE	0	R/W	High-Speed Operation Enable 0: High-speed operation is disabled When the function controller mode is selected: Only full-speed operation is enabled. When the host controller mode is selected: Full-speed or low-speed operation is enabled. 1: High-speed operation is enabled (communication speed is detected by this module) (1) When the host controller mode is selected When HSE = 0, the USB port performs low-speed or full-speed operation. Set HSE to 0 when connection of a low-speed peripheral device to the USB port has been detected. When HSE = 1, this module executes the reset handshake protocol, and automatically allows the USB port to perform high-speed or full-speed operation according to the protocol execution result. This bit should be modified after detecting device connection (after detecting the ATTCH interrupt) and before executing a USB bus reset (before setting USBRST to 1). (2) When the function controller mode is selected When HSE = 0, this module performs full-speed operation. When HSE = 1, this module executes the reset handshake protocol, and automatically performs high-speed or full-speed operation according to the protocol execution result. This bit should be modified while DPRPU is 0.
6	DCFM	0	R/W	Controller Mode Select Selects the host controller mode or function controller mode. 0: Function controller mode is selected. 1: Host controller mode is selected. This bit should be modified while DPRPU and DRPD are 0.
5	DRPD	1	R/W	D+/D- Line Resistor Control Enables or disables pulling down D+ and D- lines when the host controller mode is selected. 0: Pulling down the lines is disabled. 1: Pulling down the lines is enabled. This bit should be set to 1 if the host controller mode is selected, and should be set to 0 if the function controller mode is selected.

Bit	Bit Name	Initial Value	R/W	Description
4	DPRPU	0	R/W	D+ Line Resistor Control Enables or disables pulling up D+ line when the function controller mode is selected. 0: Pulling up the line is disabled. 1: Pulling up the line is enabled. Setting this bit to 1 when the function controller mode is selected allows this module to pull up the D+ line, thus notifying the USB host of connection. Modifying this bit from 1 to 0 allows this module to cancel pulling up the D+ line, thus notifying the USB host of disconnection. This bit should be set to 1 if the function controller mode is selected, and should be set to 0 if the host controller mode is selected.
3	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
2	UCKSEL*1	0	R/W	Input Clock Selection Selects the clock to be supplied to this module from 48-MHz USB_X1 or 12-MHz EXTAL. 0: The 48-MHz USB_X1 clock is selected. 1: The 12-MHz EXTAL clock is selected. This bit should be modified while SUSPMODE.SUSPM for channel 0 and channel 1 is 0. Set this bit to 0 when this module is to enter software standby or USB standby mode. Note that this bit is available only in channel 0 (SYSCFG0_0). Make the required setting for SYSCFG0_0 to use channel 1.
1	UPLLE*1	0	R/W	USB Internal PLL Operation Enable Enables or disables operation of the internal PLL of the USB block. 0: Disables operation of the internal PLL. 1: Enables operation of the internal PLL. This bit should be modified while SUSPMODE.SUSPM for channel 0 and channel 1 is 0. Set this bit to 0 when this module is to enter software standby or USB standby mode. Note that this bit is available only in channel 0 (SYSCFG0_0). Make the required setting for SYSCFG0_0 to use channel 1.
0	USBE	0	R/W	USB Module Operation Enable Enables or disables operation of this module. 0: USB module operation is disabled. 1: USB module operation is enabled. Modifying this bit from 1 to 0 initializes some register bits as listed in Table 29.3 and Table 29.4. This bit should be modified while SUSPMODE.SUSPM is 1. When the host controller mode is selected, this bit should be set to 1 after setting DPRD to 1, eliminating LNST bit chattering, and checking that the USB bus has been settled.

Note: • Writing to this register is possible even while the clock supply to this module is stopped (SUSPM = 0).

Note 1. UCKSEL and UPLLE are available only in channel 0 (SYSCFG0\_0).

**Table 29.3 Register Bits Initialized by Writing USBE = 0 (when Function Controller Mode is Selected)**

Register Name	Bit Name	Remarks
SYSSTS0	LNST	The value is retained when the host controller mode is selected.
DVSTCTR0	RHST	
INTSTS0	DVSQ	The value is retained when the host controller mode is selected.
USBADDR	USBADDR	The value is retained when the host controller mode is selected.
USEREQ	BRequest, bmRequestType	The values are retained when the host controller mode is selected.
USBVAL	wValue	The value is retained when the host controller mode is selected.
USBINDX	wIndex	The value is retained when the host controller mode is selected.
USBLENG	wLength	The value is retained when the host controller mode is selected.

**Table 29.4 Register Bits Initialized by Writing USBE = 0 (when Host Controller Mode is Selected)**

Register Name	Bit Name	Remarks
DVSTCTR	RHST	
FRMNUM	FRNM	The value is retained when the function controller mode is selected.
UFRMNUM	UFRNM	The value is retained when the function controller mode is selected.

### 29.3.2 CPU Bus Wait Setting Register (BUSWAIT)

BUSWAIT is a register that specifies the number of wait cycles to be inserted during an access from the CPU to this module.

This register can be modified even when the SUSPM bit in SUSPMODE is 0.

This register is initialized by a power-on reset.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	BWAIT[5:0]					
Initial value:	—	—	0	0	1	1	1	1	—	—	0	0	1	1	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15, 14	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
13, 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 8	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 0.
7, 6	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
5 to 0	BWAIT[5:0]	001111	R/W	<p>CPU Bus Wait</p> <p>Specifies the number of wait cycles to be inserted during an access to this module.</p> <p>000000: 0 wait cycles (2 access cycles)</p> <p>:</p> <p>000010: 2 wait cycles (4 access cycles)</p> <p>:</p> <p>000100: 4 wait cycles (6 access cycles)</p> <p>:</p> <p>001111: 15 wait cycles (17 access cycles) (initial value)</p> <p>:</p> <p>111111: 63 wait cycles (65 access cycles)</p> <p>There is the following constraint imposed on the cycle period required to access SYSSTS0 and the subsequent registers of this module:</p> <p>Wait constraint: The cycle period required to consecutively access registers of this controller must be at least 67 ns.</p> <p>To satisfy this constraint, it is necessary to exercise wait control according to the frequency of peripheral clock 1 (P1<math>\phi</math>). The initial value is 17 clock cycles. Select an appropriate value.</p> <p>This setting is also applied to the accesses to the FIFO port registers. The maximum access speeds for the FIFO ports are as follows:</p> <p>MBW = 10 (32-bit width): up to 60 MBytes/sec</p> <p>MBW = 01 (16-bit width): up to 30 MBytes/sec</p> <p>MBW = 00 (8-bit width): up to 15 MBytes/sec</p>



### 29.3.3 System Configuration Status Register (SYSSTS0)

SYSSTS is a register that monitors the line status (D+ and D- lines) of the USB data bus.

This register is initialized by a power-on reset or a USB bus reset.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	HTACT	SOFEA	—	—	—	—	LNST[1:0]
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 7	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
6	HTACT	Undefined	R	USB Host Sequencer Status Monitor Indicates the status of the host sequencer. 0: Stopped 1: Running This bit is set to 0 when the host sequencer in this controller is completely stopped. Make sure that the HTACT bit is set to 0 to put this controller in the USB suspended state by setting the UACT bit to 0 and stop the clock by setting the SUSPM bit to 0 in the host-mode communication state.
5	SOFEA	Undefined	R	SOF Active Monitor when the Host Controller Mode Is Selected Indicates the SOF output status. 0: SOF output stopped 1: SOF output in progress This bit allows checking if the last SOF has been output after setting the UACT bit to 0 when putting this controller in the USB suspended state with the host controller mode selected. To set the SUSPM bit to 0 (to stop the clock) after stopping this controller by setting the USBE bit to 0 in the host-mode communication state, make sure that both the HTACT bit and the SOFEA bit are set to 0.
4 to 2	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
1, 0	LNST[1:0]	Undefined	R	USB Data Line Status Monitor Indicates the status of the USB data bus lines (D+ and D-) as shown in Table 29.5. These bits should be read after setting DPRPU to 1 to notify connection when the function controller mode is selected; whereas after setting DRPD to 1 to enable pulling down the lines when the host controller mode is selected.

**Table 29.5 USB Data Bus Line States**

LNST[1]	LNST[0]	During Low-Speed Operation (Only when Host Controller Mode is Selected)	During Full-Speed Operation	During High-Speed Operation	During Chirp Operation
0	0	SE0	SE0	Squelch	Squelch
0	1	K state	J state	UnSquelch	Chirp J
1	0	J state	K state	Invalid	Chirp K
1	1	SE1	SE1	Invalid	Invalid

[Legend]

Chirp: The reset handshake protocol (RHSP) is being executed in high-speed operation enabled state (the HSE bit in SYSCFG is set to 1).

Squelch: SE0 or idle state

UnSquelch: High-speed J state or high-speed K state

Chirp J: Chirp J state

Chirp K: Chirp K state

### 29.3.4 Device State Control Register 0 (DVSTCTR0)

DVSTCTR is a register that controls and confirms the state of the USB data bus.

This register is initialized by a power-on reset. After a USB bus reset, only the WKUP bit is initialized.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	WKUP	RWUPE	USBRST	RESUME	UACT	—	RHST[2:0]		
Initial value:	—	—	—	—	0	0	0	0	0	0	0	0	—	0	0	0
R/W:	R	R	R	R	R	R	R	R/W*1	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	WKUP	0	R/W*1	<p>Wakeup Output Enables or disables outputting the remote wakeup signal (resume signal) to the USB bus when the function controller mode is selected. 0: Remote wakeup signal is not output. 1: Remote wakeup signal is output.</p> <p>This module controls the output time of a remote wakeup signal. When this bit is set to 1, this module clears this bit to 0 after outputting the 10-ms K state.</p> <p>According to the USB Specification, the USB bus idle state must be kept for 5 ms or longer before a remote wakeup signal is output. If this module writes 1 to this bit right after detection of suspended state, the K state will be output after 2 ms.</p> <p>Do not write 1 to this bit unless the device state is in the suspended state (the DVSQ bits in the INTSTS0 register are set to 1xx) and the USB host enables the remote wakeup signal. When this bit is set to 1, the internal clock must not be stopped even in the suspended state (write 1 to this bit while SUSPMODE.SUSPM is 1).</p> <p>This bit should be set to 0 if the host controller mode is selected.</p>
7	RWUPE	0	R/W	<p>Remote Wakeup Detection Enable Enables or disables the downstream port peripheral device to use the remote wakeup function (resume signal output) when the host controller mode is selected. 0: Downstream port remote wakeup is disabled. 1: Downstream port remote wakeup is enabled.</p> <p>With this bit set to 1, on detecting the resume signal (K-state for 2.5 μs) to the downstream port when the remote wakeup signal has been detected, this module performs the resume process (drives the port to the K-state). With this bit set to 0, this module ignores the detected remote wakeup signal (K-state) from the peripheral device connected to the downstream port.</p> <p>While this bit is 1, the internal clock should not be stopped even in the suspended state (SUSPMODE.SUSPM should be set to 1). Also note that the USB bus should not be reset from the suspended state (USBRST should not be set to 1); it is prohibited by USB Specification 2.0.</p> <p>This bit should be set to 0 if the function controller mode is selected.</p>

Bit	Bit Name	Initial Value	R/W	Description
6	USBRST	0	R/W	<p><b>Bus Reset Output</b> Controls the USB bus reset signal output when the host controller mode is selected.</p> <p>0: USB bus reset signal is not output. 1: USB bus reset signal is output.</p> <p>When the host controller mode is selected, setting this bit to 1 allows this module to drive the USB port to SE0 to reset the USB bus. Here, this module performs the reset handshake protocol if the HSE bit is 1. This module continues outputting SE0 while USBRST is 1 (until 0 is written to USBRST). Ensure the period over which USBRST being set to 1 (= USB bus reset period) conforms to the USB Specification 2.0. Even if 1 is written to this bit during communication (UACT = 1) or during the resume process (RESUME = 1), this module does not start the USB bus reset process until both UACT and RESUME become 0. Write 1 to the UACT bit simultaneously with the end of the USB bus reset process (writing 0 to USBRST). This bit should be set to 0 if the function controller mode is selected.</p>
5	RESUME	0	R/W	<p><b>Resume Output</b> Controls the resume signal output when the host controller mode is selected.</p> <p>0: Resume signal is not output. 1: Resume signal is output.</p> <p>Setting this bit to 1 allows this module to drive the port to the K-state and output the resume signal. The controller sets this bit to 1 when detecting a remote wakeup signal while the RWUPE bit is set to 1 and the controller is in the USB suspended state. This module continues outputting K-state while RESUME is 1 (until 0 is written to RESUME). Ensure the period over which RESUME being set to 1 (= resume period) conforms to the USB Specification 2.0. This bit should only be set to 1 in the suspended state. Write 1 to the UACT bit simultaneously with the end of the resume process (writing 0 to RESUME). This bit should be set to 0 if the function controller mode is selected.</p>
4	UACT	0	R/W	<p><b>USB Bus Enable</b> Enables operation of the USB bus (controls the SOF or <math>\mu</math>SOF packet transmission to the USB bus) when the host controller mode is selected.</p> <p>0: Downstream port is disabled (SOF/<math>\mu</math>SOF transmission is disabled). 1: Downstream port is enabled (SOF/<math>\mu</math>SOF transmission is enabled).</p> <p>With this bit set to 1, this module puts the USB port to the USB-bus enabled state and performs SOF output and data transmission and reception. This module starts outputting SOF/<math>\mu</math>SOF within 1 (<math>\mu</math>) frame after 1 is written to UACT. With this bit set to 0, this module enters the idle state after outputting SOF/<math>\mu</math>SOF. This module sets this bit to 0 on any of the following conditions.</p> <ul style="list-style-type: none"> <li>• A DTCH interrupt is detected during communication (while UACT = 1).</li> <li>• An EOFERR interrupt is detected during communication (while UACT = 1).</li> </ul> <p>Writing 1 to this bit should be done at the end of the USB bus reset process (writing 0 to USBRST) or at the end of the resume process from the suspended state (writing 0 to RESUME). This bit should be set to 0 if the function controller mode is selected.</p>
3	—	Undefined	R	<p><b>Reserved</b> The read value is undefined. The write value should always be 0.</p>

Bit	Bit Name	Initial Value	R/W	Description
2 to 0	RHST[2:0]	000	R	<p>Reset Handshake</p> <p>Indicates the status of the reset handshake.</p> <p>(1) When the host controller mode is selected</p> <p>000: Communication speed not determined (powered state or no connection)</p> <p>1xx: Reset handshake in progress</p> <p>001: Low-speed connection</p> <p>010: Full-speed connection</p> <p>011: High-speed connection</p> <p>These bits indicate 100 after 1 is written to USBRST.</p> <p>If HSE has been set to 1, these bits indicate 111 as soon as this module detects Chirp-K from the peripheral device.</p> <p>This module fixes the value of the RHST bits when 0 is written to USBRST and this module completes SE0 driving.</p> <p>When 1xxx is written to UTST (parameters for the host test have been set), these bits indicate 011.</p> <p>(2) When the function controller mode is selected</p> <p>000: Communication speed not determined (powered state or no connection)</p> <p>100: Reset handshake in progress</p> <p>010: Full-speed connection</p> <p>011: High-speed connection</p> <p>If HSE has been set to 1, these bits indicate 100 as soon as this module detects the USB bus reset. Then, these bits indicate 011 as soon as this module outputs Chirp-K and detects Chirp-JK from the USB host three times. If the connection speed is not fixed to high speed within 2.5 ms after Chirp-K output, these bits indicate 010.</p> <p>If HSE has been set to 0, these bits indicate 010 as soon as this module detects the USB bus reset.</p> <p>A DVST interrupt is generated as soon as this module detects the USB bus reset and then the value of the RHST bits is fixed to 010 or 011.</p>

Note 1. Only 1 can be written.

### 29.3.5 Test Mode Register (TESTMODE)

TESTMODE is a register that controls the USB test signal output during high-speed operation.

This register is initialized by a power-on reset.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	UTST[3:0]			
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 4	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
3 to 0	UTST[3:0]	0000	R/W	<p>Test Mode</p> <p>This module outputs the USB test signals during the high-speed operation, when these bits are written appropriate value.</p> <p>(1) When the host controller mode is selected</p> <p>These bits can be set after writing 1 to DRPD. This module outputs waveforms when both DPRD and UACT have been set to 1. This module also performs high-speed termination for the USB port by writing the appropriate value to these bits.</p> <ul style="list-style-type: none"> <li>• Procedure for setting the UTST bits <ol style="list-style-type: none"> <li>1. Power-on reset.</li> <li>2. Start the clock supply.</li> <li>3. Set SUSPM to 1.</li> <li>4. Set DCFM and DPRD to 1 (setting HSE to 1 is not required).</li> <li>5. Set USBE to 1.</li> <li>6. Set the UTST bits to the appropriate value according to the test specifications.</li> <li>7. Set the UACT bit to 1.</li> </ol> </li> <li>• Procedure for modifying the UTST bits <ol style="list-style-type: none"> <li>1. (In the state after executing step 7 above) Set UACT and USBE to 0.</li> <li>2. Set USBE to 1.</li> <li>3. Set the UTST bits to the appropriate value according to the test specifications.</li> <li>4. Set the UACT bit to 1.</li> </ol> </li> </ul> <p>When these bits are set to Test_SE0_NAK (1011), this module does not output the SOF packet even when 1 has been set to UACT for the port.</p> <p>When these bits are set to Test_Force_Enable (1101), this module outputs the SOF packet when 1 has been set to UACT. In this test mode, this module does not perform hardware control consequent to detection of high-speed disconnection (detection of the DTCH interrupt).</p> <p>When setting the UTST bits, the PID bits for all the pipes should be set to NAK.</p> <p>To return to normal USB communication after a test mode has been set, a power-on reset should be applied.</p> <p>(2) When the function controller mode is selected</p> <p>The appropriate value should be set to these bits according to the SetFeature request from the USB host during high-speed communication.</p> <p>This module does not make a transition to the suspended state while these bits are 0001 to 0100. Perform a power-on reset to carry out normal USB communication after configuring the test mode.</p>

**Table 29.6 Test Mode Operation**

Test Mode	UTST Bit Setting	
	When Function Controller Mode is Selected	When Host Controller Mode is Selected
Normal operation	0000	0000
Test_J	0001	1001
Test_K	0010	1010
Test_SE0_NAK	0011	1011
Test_Packet	0100	1100
Test_Force_Enable	—	1101
Reserved	0101 to 0111	1110 to 1111

### 29.3.6 DMA<sub>n</sub>-FIFO Bus Configuration Registers (D0FBCFG, D1FBCFG)

D0FBCFG is a register that controls bus access to the DMA0-FIFO and D1FBCFG is a register that controls bus access to the DMA1-FIFO. Note that the setting of this register is invalid when the DMA0-FIFO bus or DMA1-FIFO bus is connected to the local bus.

These registers are initialized by a power-on reset.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	DFACC[1:0]	—	—	—	—	—	—	—	—	TENDE	—	—	—	—
Initial value:	—	—	0	0	—	—	—	—	—	—	—	0	—	—	—	—
R/W:	R	R	R/W	R/W	R	R	R	R	R	R	R	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15, 14	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
13, 12	DFACC[1:0]	00	R/W	<p>DMA<sub>x</sub> FIFO Access Mode Specifies the access mode of the FIFO port.</p> <p>00: Cycle-stealing mode (initial value) 01: 16-byte continuous access mode 10: 32-byte continuous access mode 11: Setting prohibited</p> <p>These bits specify a DMA transfer mode.</p> <p>(a) In cycle-stealing mode, use the Dx<sub>x</sub>FIFO port to access the FIFO buffer. (b) In 16-byte/32-byte continuous access mode, use the Dx<sub>x</sub>FIFO continuous transfer port to access the FIFO buffer. The MBW bit in Dx<sub>x</sub>FIFOSEL can be set to 10 (32-bit width) only.</p> <p>Be sure to follow the procedure below when setting the DFACC bits for 16- or 32-byte continuous access.</p> <p>(1) Set the DREQE bit in the DnFIFOSEL register to 0. (2) Set the DFACC bits to 01 (16 bytes) or 10 (32 bytes). (3) In the DnFIFOSEL register, set the CURPIPE bits to 0000 (specifying no pipe) and the MBW bits to 10 (32-bit width) at the same time. After that, read the CURPIPE bits to confirm that they have been updated to the written value (0000). (4) Use the CPU to dummy-read the DnFIFO port register (DnFIFO) with 32-bit width (the value read data can be discarded). After that, specify the target pipe by using the CURPIPE bits in the DnFIFOSEL register and read the CURPIPE bits to confirm that they have been updated to written value. (5) Set the DREQE bit in the DnFIFOSEL to 1.</p> <p>Note that the above procedure is not required if these bits are to be set for the cycle-stealing mode (not following the procedure has no effect).</p>

Bit	Bit Name	Initial Value	R/W	Description
11 to 5	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
4	TENDE	0	R/W	DMA Transfer End Sampling Enable Controls acceptance of DMA transfer end signal. 0: DMA transfer end signal is not sampled. 1: DMA transfer end signal is sampled. Controls acceptance of DMA transfer end signal output from the direct memory access controller on completion of a DMA transfer. This module can control input of the DMA transfer end signal to end the DMA transfer of data to the FIFO. Set this bit to 0 when the DFACC bits are set to 01 or 10.
3 to 0	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.

### 29.3.7 FIFO Port Registers (CFIFO, D0FIFO, D1FIFO)

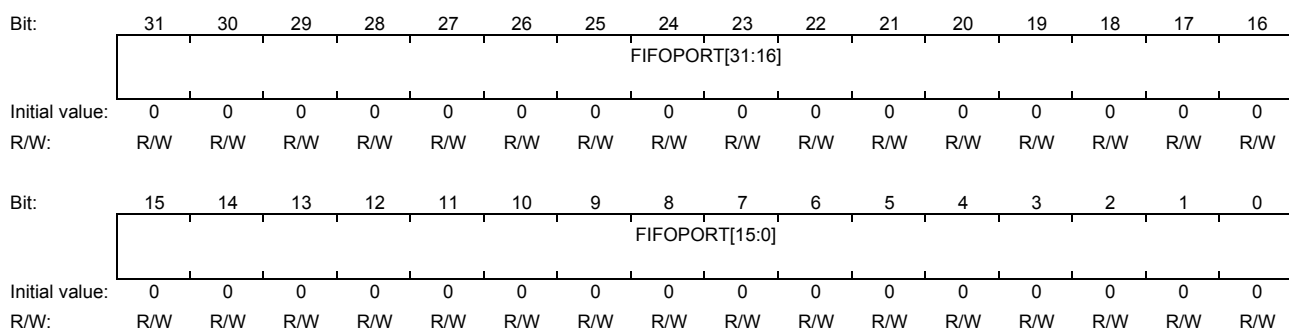
CFIFO, D0FIFO, and D1FIFO are port registers that are used to read data from the FIFO buffer memory and write data to the FIFO buffer memory.

There are three FIFO ports: the CFIFO, D0FIFO and D1FIFO ports. There are also DxFIFO continuous transfer ports for continuous transfer. Each FIFO port is configured of a port register (CFIFO, D0FIFO, or D1FIFO) that handles reading of data from the FIFO buffer memory and writing of data to the FIFO buffer memory, a select register (CFIFOSEL, D0FIFOSEL, or D1FIFOSEL) that is used to select the pipe assigned to the FIFO port, and a control register (CFIFOCTR, D0FIFOCTR, or D1FIFOCTR).

Each FIFO port has the following characteristics.

- Access to the FIFO buffer for the DCP should be performed through the CFIFO port.
- Access to the FIFO buffer by DMA transfer should be performed through the D0FIFO or D1FIFO port when the DFACC bits are set to 00 (cycle-stealing mode).
- Access to the FIFO buffer by DMA transfer should be performed through D0FIFO or D1FIFO continuous transfer ports when the DFACC bits are set to 01 (16-byte continuous access mode) or 10 (32-byte continuous access mode).
- The D1FIFO or D0FIFO port can be accessed by the CPU.
- When using functions specific to the FIFO port, the pipe number (selected pipe) specified by the CURPIPE bits cannot be changed (when the DMA transfer function is used, etc.).
- Registers configuring a FIFO port do not affect other FIFO ports.
- The same pipe should not be assigned to two or more FIFO ports.
- There are two FIFO buffer states: the access right is on the CPU side and it is on the SIE side. When the FIFO buffer access right is on the SIE side, the FIFO buffer cannot be accessed from the CPU.

These registers are initialized by a power-on reset.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	FIFOPORT [31:0]	All 0	R/W	<p>FIFO Port</p> <p>Accessing these bits allow reading the received data from the FIFO buffer or writing the transmit data to the FIFO buffer.</p> <p>These bits can be accessed only while the FRDY bit in each FIFO port control register (CFIFOCTR, D0FIFOCTR, or D1FIFOCTR) is 1 (or this module is asserting the UCL_Dx_DREQ signal).</p> <p>The valid bits in this register depend on the settings of the MBW bits (access bit width setting) and BIGEND bit (endian setting) as shown in Table 29.7 to Table 29.9.</p>

**Table 29.7 Endian Operation in 32-Bit Access**

BIGEND Bit	Bits 31 to 24	Bits 23 to 16	Bits 15 to 8	Bits 7 to 0
0	N+3 address	N+2 address	N+1 address	N+0 address
1	N+0 address	N+1 address	N+2 address	N+3 address

**Table 29.8 Endian Operation in 16-Bit Access**

BIGEND Bit	Bits 31 to 24	Bits 23 to 16	Bits 15 to 8	Bits 7 to 0
0	N+1 address	N+0 address	Writing: invalid, reading: prohibited*1	
1	Writing: invalid, reading: prohibited*1		N+0 address	N+1 address

Note 1. Reading data from the invalid bits in a word or byte unit is prohibited.

**Table 29.9 Endian Operation in 8-Bit Access**

BIGEND Bit	Bits 31 to 24	Bits 23 to 16	Bits 15 to 8	Bits 7 to 0
0	N+0 address	Writing: invalid, reading: prohibited*1		
1	Writing: invalid, reading: prohibited*1			N+0 address

Note 1. Reading data from the invalid bits in a word or byte unit is prohibited.



### 29.3.8 FIFO Port Select Registers (CFIFOSEL, D0FIFOSEL, D1FIFOSEL)

CFIFOSEL, D0FIFOSEL and D1FIFOSEL are registers that assign the pipe to the FIFO port, and control access to the corresponding port.

The same pipe should not be specified by the CURPIPE bits in CFIFOSEL, D0FIFOSEL and D1FIFOSEL. When the CURPIPE bits in D0FIFOSEL and D1FIFOSEL are cleared to B'0000, no pipe is selected.

The pipe number should not be changed while the DMA transfer is enabled.

These registers are initialized by a power-on reset.

#### (1) CFIFOSEL

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RCNT	REW	—	—	MBW[1:0]		—	BIG END	—	—	ISEL	—	CURPIPE[3:0]			
Initial value:	0	0	—	—	0	0	—	0	—	—	0	—	0	0	0	0
R/W:	R/W	R/W*1	R	R	R/W	R/W	R	R/W	R	R	R/W	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	RCNT	0	R/W	<p>Read Count Mode</p> <p>Specifies the read mode for the value in the DTLN bits in CFIFOCTR.</p> <p>0: The DTLN bits are cleared when all of the receive data has been read from the CFIFO.</p> <p>(In double buffer mode, the DTLN bit value is cleared when all the data has been read from a single plane.)</p> <p>1: The DTLN bits are decremented when the receive data is read from the CFIFO.</p> <p>When this bit is cleared to 0, this module clears the DTLN bits in CFIFOCTR to 0 when all of the receive data has been read from the FIFO buffer that is assigned to the pipe specified in the CURPIPE bits (called the specified pipe) (in double buffer mode, the timing to clear is when finished reading data from one FIFO buffer plane).</p> <p>When this bit is set to 1, this module decrements the DTLN bits in CFIFOCTR every time receive data is read from the FIFO buffer that is assigned to the specified pipe.</p>
14	REW	0	R/W*1	<p>Buffer Pointer Rewind</p> <p>Specifies whether or not to rewind the buffer pointer.</p> <p>0: The buffer pointer is not rewind.</p> <p>1: The buffer pointer is rewind.</p> <p>When the selected pipe is in the receiving direction, setting this bit to 1 while the FIFO buffer is being read allows re-reading the FIFO buffer from the first data (in double buffer mode, re-reading the currently-read one FIFO buffer plane from the first data is allowed).</p> <p>Do not set REW to 1 simultaneously with modifying the CURPIPE bits. Before setting REW to 1, be sure to check that FRDY is 1.</p> <p>To re-write to the FIFO buffer again from the first data for the pipe in the transmitting direction, use the BCLR bit.</p>
13, 12	—	Undefined	R	<p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p>

Bit	Bit Name	Initial Value	R/W	Description
11, 10	MBW[1:0]	00	R/W	<p>CFIFO Port Access Bit Width</p> <p>Specifies the bit width for accessing the CFIFO port.</p> <p>00: 8-bit width 01: 16-bit width 10: 32-bit width 11: Setting prohibited</p> <p>When the pipe specified by the CURPIPE bits is in the receiving direction, once reading data is started after setting these bits, these bits should not be modified until all the data has been read. When the specified pipe is in the receiving direction, set the CURPIPE and MBW bits simultaneously. When the specified pipe is in the transmitting direction, the bit width cannot be changed from the 8-bit width to the 16-/32-bit width or from the 16-bit width to the 32-bit width while data is being written to the buffer memory.</p>
9	—	Undefined	R	<p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p>
8	BIGEND	0	R/W	<p>CFIFO Port Endian Control</p> <p>Specifies the byte endian for the CFIFO port. For details, refer to the description of the FIFO port bits in section 29.3.7, FIFO Port Registers (CFIFO, D0FIFO, D1FIFO).</p> <p>0: Little endian 1: Big endian</p>
7, 6	—	Undefined	R	<p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p>
5	ISEL	0	R/W	<p>CFIFO Port Access Direction when DCP is Selected</p> <p>Specifies the direction of FIFO port access when the DCP is selected through the CURPIPE bits.</p> <p>0: Reading from the buffer memory is selected 1: Writing to the buffer memory is selected</p> <p>After writing to this bit with the DCP being a selected pipe, read this bit to check that the written value agrees with the read value before proceeding to the next process.</p> <p>When this bit is modified during access to the FIFO buffer, the access results up to that point can be retained, and after the bit is restored to the previous value, access can be continued.</p> <p>Set this bit and the CURPIPE bits simultaneously.</p>
4	—	Undefined	R	<p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p>
3 to 0	CURPIPE[3:0]	0000	R/W	<p>CFIFO Port Access Pipe Specification</p> <p>Specifies the pipe number using which data is read or written through the CFIFO port.</p> <p>0000: DCP 0001: PIPE 1 0010: PIPE 2 : 1110: PIPE 14 1111: PIPE 15 Others: Setting prohibited</p> <p>After writing to these bits, read these bits to check that the written value agrees with the read value before proceeding to the next process.</p> <p>Do not set the same pipe number to the CURPIPE bits in CFIFOSEL, D0FIFOSEL, and D1FIFOSEL.</p> <p>Even if the setting of these bits is modified during access to the FIFO buffer, the state of the FIFO buffer is retained, with continued access proceeding after these bits are re-set to the value before the modification.</p>

Note 1. Only 0 can be read and 1 can be written to.

## (2) D0FIFOSEL, D1FIFOSEL

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RCNT	REW	DCLRM	DREQE	MBW[1:0]		—	BIG END	—	—	—	—	CURPIPE[3:0]			
Initial value:	0	0	0	0	0	0	—	0	—	—	—	—	0	0	0	0
R/W:	R/W	R/W*	R/W	R/W	R/W	R/W	R	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	RCNT	0	R/W	<p>Read Count Mode</p> <p>Specifies the read mode for the value in the DTLN bits in DnFIFOCTR.</p> <p>0: The DTLN bits are cleared when all of the receive data has been read from the DnFIFO. (In double buffer mode, the DTLN bit value is cleared when all the data has been read from a single plane.)</p> <p>1: The DTLN bits are decremented when the receive data is read from the DnFIFO.</p> <p>When this bit is cleared to 0, this module clears the DTLN bits in DnFIFOCTR to 0 when all of the receive data has been read from the FIFO buffer that is assigned to the pipe specified in the CURPIPE bits (called the specified pipe) (in double buffer mode, the timing to clear is when finished reading data from one FIFO buffer plane).</p> <p>When this bit is set to 1, this module decrements the DTLN bits in DnFIFOCTR every time receive data is read from the FIFO buffer that is assigned to the specified pipe.</p>
14	REW	0	R/W*	<p>Buffer Pointer Rewind</p> <p>Specifies whether or not to rewind the buffer pointer.</p> <p>0: The buffer pointer is not rewind.</p> <p>1: The buffer pointer is rewind.</p> <p>When the specified pipe is in the receiving direction, setting this bit to 1 while the FIFO buffer is being read allows re-reading the FIFO buffer from the first data (in double buffer mode, re-reading the currently-read FIFO buffer plane from the first data is allowed). After 1 is written to this bit by software, this module returns this bit to 0.</p> <p>Do not set REW to 1 simultaneously with modifying the CURPIPE bits. Before setting REW to 1, be sure to check that FRDY is 1.</p> <p>To re-write to the FIFO buffer again from the first data for the pipe in the transmitting direction, use the BCLR bit.</p>
13	DCLRM	0	R/W	<p>Auto Buffer Memory Clear Mode Accessed after Specified Pipe Data is Read</p> <p>Enables or disables the buffer memory to be cleared automatically after data has been read out using the selected pipe.</p> <p>0: Auto buffer clear mode is disabled.</p> <p>1: Auto buffer clear mode is enabled.</p> <p>With this bit set to 1, this module sets BCLR to 1 for the FIFO buffer of the selected pipe on receiving a zero-length packet while the FIFO buffer assigned to the selected pipe is empty, or on receiving a short packet and reading the data while BFRE is 1.</p> <p>When using this module with the BRDYM bit set to 1, set this bit to 0.</p>
12	DREQE	0	R/W	<p>UCL_Dx_DREQ Signal Output Enable</p> <p>Enables or disables the UCL_Dx_DREQ signal output.</p> <p>0: Output is disabled.</p> <p>1: Output is enabled.</p> <p>To enable the UCL_Dx_DREQ signal output, set this bit to 1 after setting the CURPIPE bits. When modifying the CURPIPE bits, first set this bit to 0.</p>

Bit	Bit Name	Initial Value	R/W	Description
11, 10	MBW[1:0]	00	R/W	<p>FIFO Port Access Bit Width</p> <p>Specifies the bit width for accessing the DnFIFO port.</p> <p>00: 8-bit width 01: 16-bit width 10: 32-bit width 11: Setting prohibited</p> <p>Set 10 when the DFACC bits are set to 01 or 10.</p> <p>Once reading from the buffer memory has started, the bit width for accessing the FIFO port cannot be changed until all data has been read.</p> <p>When the specified pipe is in the receiving direction, set the CURPIPE and MBW bits simultaneously. Refer to section 29.4.4, FIFO Buffer Memory.</p> <p>When the specified pipe is in the transmitting direction, the bit width cannot be changed from the 8-bit width to the 16-/32-bit width or from the 16-bit width to the 32-bit width while data is being written to the buffer memory.</p> <p>Set these bits to 10 (32-bit width) when the DFACC bits are set to 01 (16-byte continuous access mode) or 10 (32-byte continuous access mode).</p>
9	—	Undefined	R	<p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p>
8	BIGEND	0	R/W	<p>FIFO Port Endian Control</p> <p>Specifies the byte endian for the DnFIFO port. For details, refer to the description of the FIFO port bits in section 29.3.7, FIFO Port Registers (CFIFO, D0FIFO, D1FIFO).</p> <p>0: Little endian 1: Big endian</p>
7 to 4	—	Undefined	R	<p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p>
3 to 0	CURPIPE[3:0]	0000	R/W	<p>FIFO Port Access Pipe Specification</p> <p>Specify a desired pipe number for which data is read or written through the D0FIFO or D1FIFO port.</p> <p>0000: No pipe specified 0001: PIPE 1 0010: PIPE 2 : 1110: PIPE 14 1111: PIPE 15 Others: Setting prohibited</p> <p>After writing to these bits, read these bits to check that the written value agrees with the read value before proceeding to the next process.</p> <p>Do not set the same pipe number to the CURPIPE bits in CFIFOSEL, D0FIFOSEL, and D1FIFOSEL.</p> <p>Even if the setting of these bits is modified during access to the FIFO buffer, the state of the FIFO buffer is retained, with continued access proceeding after these bits are re-set to the value before the modification.</p>

### 29.3.9 FIFO Port Control Registers (CFIFOCTR, D0FIFOCTR, D1FIFOCTR)

CFIFOCTR, D0FIFOCTR and D1FIFOCTR are registers that determine whether or not writing to the buffer memory has been finished, the buffer accessed from the CPU has been cleared, and the FIFO port is accessible. CFIFOCTR, D0FIFOCTR, and D1FIFOCTR are used for the corresponding FIFO ports.

These registers are initialized by a power-on reset.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BVAL	BCLR	FRDY	—							DTLN[11:0]					
Initial value:	0	0	0	—	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*2	R/W*1	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	BVAL	0	R/W*2	<p><b>Buffer Memory Valid Flag</b></p> <p>Set this bit to 1 when writing has completed in the CPU-side FIFO buffer for the pipe specified in CURPIPE (called the selected pipe).</p> <p>0: Invalid 1: Writing ended</p> <p>When the selected pipe is in the transmitting direction, set this bit to 1 in the following cases. Then, this module switches the FIFO buffer from the CPU side to the SIE side, enabling transmission.</p> <ul style="list-style-type: none"> <li>To transmit a short packet, set this bit to 1 after data has been written.</li> <li>To transmit a zero-length packet, set this bit to 1 before writing data to the FIFO buffer.</li> <li>Set this bit to 1 after the number of data bytes has been written for the pipe in continuous transfer mode, where the number is a natural integral multiple of the maximum packet size and less than the buffer size.</li> </ul> <p>When the data of the maximum packet size has been written for the pipe in non-continuous transfer mode, this module sets this bit to 1 and switches the FIFO buffer from the CPU side to the SIE side, enabling transmission.</p> <p>Writing 1 to this bit should be done while FRDY indicates 1.</p> <p>When checking the FRDY bit after setting this bit, allow an interval of at least 80 ns before referencing FRDY.</p> <p>When the selected pipe is in the receiving direction, do not write 1 to this bit.</p>
14	BCLR	0	R/W*1	<p><b>CPU Buffer Clear</b></p> <p>This bit should be set to 1 to clear the FIFO buffer on the CPU side for the selected pipe.</p> <p>0: Invalid 1: Clears the buffer memory on the CPU side.</p> <p>When double buffer mode is set for the FIFO buffer assigned to the selected pipe, this module clears only one plane of the FIFO buffer even when both planes are read-enabled.</p> <p>When the selected pipe is the DCP, setting BCLR to 1 allows this module to clear the FIFO buffer regardless of whether the FIFO buffer is on the CPU side or SIE side. When clearing the buffer on the SIE side, set the PID bits for the DCP to NAK before setting BCLR to 1.</p> <p>When the selected pipe is not the DCP, writing 1 to this bit should be done while FRDY indicates 1. When checking the FRDY bit after setting this bit, allow an interval of at least 80 ns before referencing FRDY.</p>

Bit	Bit Name	Initial Value	R/W	Description
13	FRDY	0	R	<p>FIFO Port Ready</p> <p>Indicates whether the FIFO port can be accessed.</p> <p>0: FIFO port access is disabled.</p> <p>1: FIFO port access is enabled.</p> <p>In the following cases, this module sets FRDY to 1 but data cannot be read via the FIFO port because there is no data to be read. In these cases, set BCLR to 1 to clear the FIFO buffer, and enable transmission and reception of the next data.</p> <ul style="list-style-type: none"> <li>• A zero-length packet is received when the FIFO buffer assigned to the selected pipe is empty.</li> <li>• A short packet is received and the data is completely read while BFRE is 1.</li> </ul>
12	—	Undefined	R	<p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p>
11 to 0	DTLN[11:0]	H'000	R	<p>Receive Data Length</p> <p>Indicates the length of the receive data.</p> <p>While the FIFO buffer is being read, these bits indicate the different values depending on the RCNT bit value as described below.</p> <ul style="list-style-type: none"> <li>• When RCNT = 0: <ul style="list-style-type: none"> <li>The length of received data is set in these bits, and the value is retained until all received data has been read from a single FIFO buffer plane.</li> <li>While BFRE is 1, these bits retain the length of the receive data until BCLR is set to 1 even after all the data has been read.</li> </ul> </li> <li>• When RCNT = 1: <ul style="list-style-type: none"> <li>This module decrements the value indicated by these bits each time data is read from the FIFO buffer. (The value is decremented by one when MBW is 00, by two when MBW is 01, and by four when MBW is 10.)</li> </ul> </li> </ul> <p>This module sets these bits to 0 when all the data has been read from one FIFO buffer plane. However, in double buffer mode, if data has been received in one FIFO buffer plane before all the data has been read from the other plane, this module sets these bits to indicate the length of the receive data in the latter plane when all the data has been read from the former plane.</p> <p>Note: When reading these bits during FIFO buffer reading while RCNT = 1, note that these bits are updated within ten bus cycles after a read cycle for the FIFO port.</p>

Note 1. Only 0 can be read and 1 can be written to.

Note 2. Only 1 can be written to.

### 29.3.10 Interrupt Enable Register 0 (INTENB0)

INTENB0 is a register that enables various interrupts. On detecting the interrupt corresponding to the bit in this register which has been set to 1, this module generates the USB interrupt.

This module sets 1 to each status bit in INTSTS0 when a detection condition of the corresponding interrupt source has been satisfied regardless of the set value in INTENB0 (regardless of whether the interrupt output is enabled or disabled). While the status bit in INTSTS0 corresponding to the interrupt source indicates 1, this module generates the USB interrupt when the corresponding interrupt enable bit in INTENB0 is changed from 0 to 1.

This register is initialized by a power-on reset.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VBSE	RSME	SOFE	DVSE	CTRE	BEMPE	NRDYE	BRDYE	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	—	—	—	—	—	—	—	—
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	VBSE	0	R/W	VBUS Interrupt Enable Enables or disables the USB interrupt request when the VBINT interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled
14	RSME	0	R/W	Resume Interrupt Enable* <sup>1</sup> Enables or disables the USB interrupt request when the RESM interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled
13	SOFE	0	R/W	Frame Number Update Interrupt Enable Enables or disables the USB interrupt request when the SOFR interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled
12	DVSE	0	R/W	Device State Transition Interrupt Enable* <sup>1</sup> Enables or disables the USB interrupt request when the DVST interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled
11	CTRE	0	R/W	Control Transfer Stage Transition Interrupt Enable* <sup>1</sup> Enables or disables the USB interrupt request when the CTRT interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled
10	BEMPE	0	R/W	Buffer Empty Interrupt Enable Enables or disables the USB interrupt request when the BEMP interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled
9	NRDYE	0	R/W	Buffer Not Ready Response Interrupt Enable Enables or disables the USB interrupt request when the NRDY interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled
8	BRDYE	0	R/W	Buffer Ready Interrupt Enable Enables or disables the USB interrupt request when the BRDY interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.

Note 1. The RSME, DVSE, and CTRE bits can be set to 1 only when the function controller mode is selected; do not set these bits to 1 to enable the corresponding interrupt output when the host controller mode is selected.

### 29.3.11 Interrupt Enable Register 1 (INTENB1)

INTENB1 is a register that enables various interrupts when the host controller mode is selected.

On detecting the interrupt corresponding to the bit in this register which has been set to 1, this module generates the USB interrupt.

This module sets 1 to each status bit in INTSTS1 when a detection condition of the corresponding interrupt source has been satisfied regardless of the set value in INTENB1 (regardless of whether the interrupt output is enabled or disabled).

While the status bit in INTSTS1 corresponding to the interrupt source indicates 1, this module generates the USB interrupt when the corresponding interrupt enable bit in INTENB1 is changed from 0 to 1.

When the function controller mode is selected, the interrupts should not be enabled.

This register is initialized by a power-on reset.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	BCHGE	—	DTCHE	ATTCH E	—	—	—	—	EOFER RE	SIGNE	SACKE	—	—	—	—
Initial value:	0	0	—	0	0	—	0	0	—	0	0	0	—	—	—	0
R/W:	R	R/W	R	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	BCHGE	0	R/W	USB Bus Change Interrupt Enable Enables or disables the USB interrupt request when the BCHG interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled
13	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
12	DTCHE	0	R/W	Disconnection Detection Interrupt Enable Enables or disables the USB interrupt request when the DTCH interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled
11	ATTCH E	0	R/W	Connection Detection Interrupt Enable Enables or disables the USB interrupt request when the ATTCH interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled
10	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
9, 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.



Bit	Bit Name	Initial Value	R/W	Description
6	EOFERRE	0	R/W	EOF Error Detection Interrupt Enable Enables or disables the USB interrupt request when the EOFERR interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled
5	SIGNE	0	R/W	Setup Transaction Error Interrupt Enable Enables or disables the USB interrupt request when the SIGN interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled
4	SACKE	0	R/W	Setup Transaction Normal Response Interrupt Enable Enables or disables the USB interrupt request when the SACK interrupt is detected. 0: Interrupt output disabled 1: Interrupt output enabled
3 to 1	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
0	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Note 1. The INTENB1 register bits can be set to 1 only when the host controller mode is selected; do not set these bits to 1 to enable the corresponding interrupt output when the function controller mode is selected.

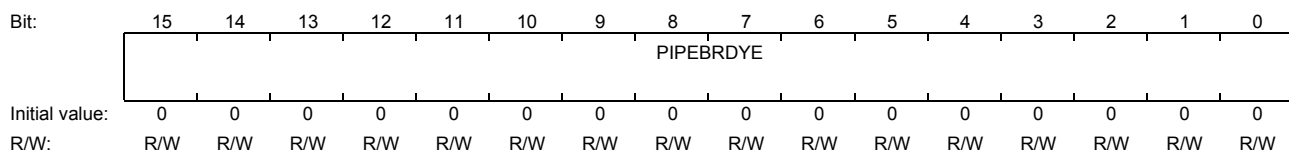
### 29.3.12 BRDY Interrupt Enable Register (BRDYENB)

BRDYENB is a register that enables or disables the BRDY bit in INTSTS0 to be set to 1 when the BRDY interrupt is detected for each pipe.

On detecting the BRDY interrupt for the pipe corresponding to the bit in this register which has been set to 1, this module sets 1 to the corresponding PIPEBRDY bit in BRDYSTS and the BRDY bit in INTSTS0, and generates the BRDY interrupt.

While at least one PIPEBRDY bit in BRDYSTS indicates 1, this module generates the BRDY interrupt when the corresponding interrupt enable bit in BRDYENB is changed from 0 to 1.

This register is initialized by a power-on reset.



Bit	Bit Name	Initial Value	R/W	Description
15 to 0	PIPEBRDYE	H'0000	R/W	BRDY Interrupt Enable for each Pipe 0: Interrupt output disabled 1: Interrupt output enabled

Note 1. The bit number corresponds to the pipe number.

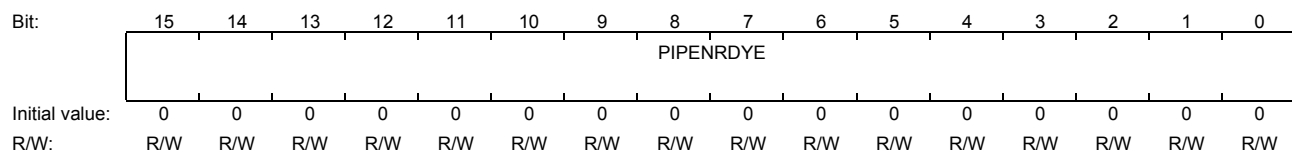
### 29.3.13 NRDY Interrupt Enable Register (NRDYENB)

NRDYENB is a register that enables or disables the NRDY bit in INTSTS0 to be set to 1 when the NRDY interrupt is detected for each pipe.

On detecting the NRDY interrupt for the pipe corresponding to the bit in this register which has been set to 1, this module sets 1 to the corresponding PIPENRDY bit in NRDYSTS and the NRDY bit in INTSTS0, and generates the NRDY interrupt.

While at least one PIPENRDY bit in NRDYSTS indicates 1, this module generates the NRDY interrupt when the corresponding interrupt enable bit in NRDYENB is changed from 0 to 1.

This register is initialized by a power-on reset.



Bit	Bit Name	Initial Value	R/W	Description
15 to 0	PIPENRDYE	H'0000	R/W	NRDY Interrupt Enable for each Pipe 0: Interrupt output disabled 1: Interrupt output enabled

Note 1. The bit number corresponds to the pipe number.

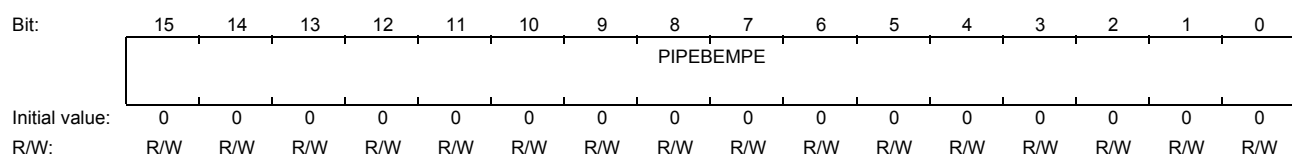
### 29.3.14 BEMP Interrupt Enable Register (BEMPENB)

BEMPENB is a register that enables or disables the BEMP bit in INTSTS0 to be set to 1 when the BEMP interrupt is detected for each pipe.

On detecting the BEMP interrupt for the pipe corresponding to the bit in this register which has been set to 1, this module sets 1 to the corresponding PIPEBEMP bit in BEMPSTS and the BEMP bit in INTSTS0, and generates the BEMP interrupt.

While at least one PIPEBEMP bit in BEMPSTS indicates 1, this module generates the BEMP interrupt when the corresponding interrupt enable bit in BEMPENB is changed from 0 to 1.

This register is initialized by a power-on reset.



Bit	Bit Name	Initial Value	R/W	Description
15 to 0	PIPEBEMPE	H'0000	R/W	BEMP Interrupt Enable for each Pipe 0: Interrupt output disabled 1: Interrupt output enabled

Note 1. The bit number corresponds to the pipe number.

### 29.3.15 SOF Output Configuration Register (SOFCFG)

SOFCFG is a register that specifies the transaction-enabled time and BRDY interrupt status clear timing. This register is initialized by a power-on reset.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TRNEN SEL	—	BRDY M	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	—	0	—	0	0	0	0	0	—	—
R/W:	R	R	R	R	R	R	R	R/W	R	R/W	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 9	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
8	TRNENSEL	0	R/W	Transaction-Enabled Time Select Selects the transaction-enabled time either for full-speed or low-speed communication, where is the time in which this module issues tokens in a frame. 0: For non-low-speed communication 1: For low-speed communication This bit is valid only when the host controller mode is selected. Even when the host controller mode is selected, the setting of this bit has no effect on the transaction-enabled time during high-speed communication. This bit should be set to 0 when the function controller mode is selected.
7	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
6	BRDYM	0	R/W	BRDY Interrupt Status Clear Timing for Each Pipe Specifies the timing for clearing the BRDY interrupt status for each pipe. 0: Clears the status by writing 0 to this bit. 1: This module automatically clears the status when data has been read from the FIFO buffer or data has been written to the FIFO buffer.
5 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.

### 29.3.16 Interrupt Status Register 0 (INTSTS0)

INTSTS0 is a register that indicates the status of the various interrupts detected.

This register is initialized by a power-on reset. By a USB bus reset, the DVST and DVSQ[2:0] bits are initialized.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VBINT	RESM	SOFR	DVST	CTRT	BEMP	NRDY	BRDY	VBSTS	DVSQ[2:0]			VALID	CTSQ[2:0]		
Initial value:	0	0	0	0/1 <sup>*1</sup>	0	0	0	0	0/1 <sup>*3</sup>	0 <sup>*2</sup>	0 <sup>*2</sup>	0/1 <sup>*2</sup>	0	0	0	0
R/W:	R/W <sup>*7</sup>	R/W <sup>*7</sup>	R/W <sup>*7</sup>	R/W <sup>*7</sup>	R/W <sup>*7</sup>	R	R	R	R	R	R	R	R/W <sup>*7</sup>	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	VBINT	0	R/W <sup>*7</sup>	VBUS Interrupt Status <sup>*4*5</sup> 0: VBUS interrupts not generated 1: VBUS interrupts generated This module sets this bit to 1 when a change in the value input to the VBUS pin (high to low or low to high) is detected. The module indicates the VBUS pin input value in the VBSTS bit. When a VBINT interrupt is generated, the VBSTS bit is read several times to remove the chattering effect until the same value is read repeatedly from the bit.
14	RESM	0	R/W <sup>*7</sup>	Resume Interrupt Status <sup>*4*5*6</sup> 0: Resume interrupts not generated 1: Resume interrupts generated When the function controller mode is selected, this module sets this bit to 1 on detecting the falling edge of the signal on the DP pin in the suspended state (DVSQ = 1XX). When the host controller mode is selected, the read value is invalid.
13	SOFR	0	R/W <sup>*7</sup>	Frame Number Refresh Interrupt Status <sup>*4</sup> 0: SOF interrupts not generated 1: SOF interrupts generated (1) When the host controller mode is selected This module sets this bit to 1 on updating the frame number when the UACT bit is set to 1. (This interrupt is detected every 1 ms.) (2) When the function controller mode is selected This module sets this bit to 1 on updating the frame number. (This interrupt is detected every 1 ms.) This module can detect an SOFR interrupt through the internal interpolation function even when a damaged SOF packet is received from the USB host.
12	DVST	0/1 <sup>*1</sup>	R/W <sup>*7</sup>	Device State Transition Interrupt Status <sup>*4*6</sup> 0: Device state transition interrupts not generated 1: Device state transition interrupts generated When the function controller mode is selected, this module updates the DVSQ value and sets this bit to 1 on detecting a change in the device state. When this interrupt is generated, clear the status before this module detects the next device state transition. When the host controller mode is selected, the read value is invalid.
11	CTRT	0	R/W <sup>*7</sup>	Control Transfer Stage Transition Interrupt Status <sup>*4*6</sup> 0: Control transfer stage transition interrupts not generated 1: Control transfer stage transition interrupts generated When the function controller mode is selected, this module updates the CTSQ value and sets this bit to 1 on detecting a change in the control transfer stage. When this interrupt is generated, clear the status before this module detects the next control transfer stage transition. When the host controller mode is selected, the read value is invalid.

Bit	Bit Name	Initial Value	R/W	Description
10	BEMP	0	R	<p>Buffer Empty Interrupt Status</p> <p>0: BEMP interrupts not generated 1: BEMP interrupts generated</p> <p>This module sets this bit to 1 when at least one PIPEBEMP bit in BEMPSTS is set to 1 among the PIPEBEMP bits corresponding to the PIPEBEMPE bits in BEMPENB to which 1 has been set (when this module detects the BEMP interrupt status in at least one pipe among the pipes for which the BEMP interrupt notification is enabled). For the conditions for PIPEBEMP status assertion, refer to section 29.4.2 (4) BEMP Interrupt.</p> <p>This module clears this bit to 0 when 0 is written to all the PIPEBEMP bits corresponding to the PIPEBEMPE bits to which 1 has been set. This bit cannot be cleared to 0 even if 0 is written to this bit.</p>
9	NRDY	0	R	<p>Buffer Not Ready Interrupt Status</p> <p>0: NRDY interrupts not generated 1: NRDY interrupts generated</p> <p>This module sets this bit to 1 when at least one PIPENRDY bit in NRDYSTS is set to 1 among the PIPENRDY bits corresponding to the PIPENRDYE bits in NRDYENB to which 1 has been set (when this module detects the NRDY interrupt status in at least one pipe among the pipes for which the NRDY interrupt notification is enabled). For the conditions for PIPENRDY status assertion, refer to section 29.4.2 (3) NRDY Interrupt.</p> <p>This module clears this bit to 0 when 0 is written to all the PIPENRDY bits corresponding to the PIPENRDYE bits to which 1 has been set. This bit cannot be cleared to 0 even if 0 is written to this bit.</p>
8	BRDY	0	R	<p>Buffer Ready Interrupt Status</p> <p>0: BRDY interrupts not generated 1: BRDY interrupts generated</p> <p>This module sets this bit to 1 when at least one PIPEBRDY bit in BRDYSTS is set to 1 among the PIPEBRDY bits corresponding to the PIPEBRDYE bits in BRDYENB to which 1 has been set (when this module detects the BRDY interrupt status in at least one pipe among the pipes for which the BRDY interrupt notification is enabled). For the conditions for PIPEBRDY status assertion, refer to section 29.4.2 (2) BRDY Interrupt.</p> <p>This module clears this bit to 0 when 0 is written to all the PIPEBRDY bits corresponding to the PIPEBRDYE bits to which 1 has been set. This bit cannot be cleared to 0 even if 0 is written to this bit.</p>
7	VBSTS	0/1 <sup>3</sup>	R	<p>VBUS Input Status</p> <p>0: The VBUS pin is low level. 1: The VBUS pin is high level.</p>
6 to 4	DVSQL[2:0]	000/001 <sup>2</sup>	R	<p>Device State</p> <p>000: Powered state 001: Default state 010: Address state 011: Configured state 1xx: Suspended state</p> <p>When the host controller mode is selected, the read value is invalid.</p>
3	VALID	0	R/W <sup>7</sup>	<p>USB Request Reception</p> <p>0: Not detected 1: Setup packet reception</p> <p>When the host controller mode is selected, the read value is invalid.</p>
2 to 0	CTSQQ[2:0]	000	R	<p>Control Transfer Stage</p> <p>000: Idle or setup stage 001: Control read data stage 010: Control read status stage 011: Control write data stage 100: Control write status stage 101: Control write (no data) status stage 110: Control transfer sequence error 111: Setting prohibited</p> <p>When the host controller mode is selected, the read value is invalid.</p>

- Note 1. This bit is initialized to B'0 by a power-on reset and B'1 by a USB bus reset.
- Note 2. These bits are initialized to B'000 by a power-on reset and B'001 by a USB bus reset.
- Note 3. This bit is 1 when the level of the VBUS pin input is high and 0 when low.
- Note 4. To clear the VBINT, RESM, SOFR, DVST, or CTRT bit, write 0 only to the bits to be cleared; write 1 to the other bits. Do not write 0 to the status bits indicating 0.
- Note 5. A change in the status indicated by the VBINT and RESM bits can be detected by this module even while the clock supply is stopped, and the interrupts are output when the corresponding interrupt enable bits are enabled. Clearing the status should be done after enabling the clock supply.
- Note 6. A change in the status of the RESM, DVST, and CTRT bits occurs only when the function controller mode is selected; disable the corresponding interrupt enable bits (set to 0) when the host controller mode is selected.
- Note 7. Only 0 can be written to.

### 29.3.17 Interrupt Status Register 1 (INTSTS1)

INTSTS1 is a register that is used to confirm interrupt status.

The various interrupts indicated by the bits in this register should be enabled only when the host controller mode is selected.

This register is initialized by a power-on reset.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	BCHG	—	DTCH	ATTCH	—	—	—	—	EOFER R	SIGN	SACK	—	—	—	—
Initial value:	0	0	—	0	0	—	0	0	—	0	0	0	—	—	—	0
R/W:	R	R/W*	R	R/W*	R/W*	R	R	R	R	R/W*	R/W*	R/W*	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	BCHG	0	R/W*	USB Bus Change Interrupt Status Indicates the status of the USB bus change interrupt. 0: BCHG interrupts not generated 1: BCHG interrupts generated This module detects the BCHG interrupt when a change in the full-speed or low-speed signal level occurs on the USB port (a change from J-state, K-state, or SE0 to J-state, K-state, or SE0), and sets this bit to 1. Here, if the corresponding interrupt enable bit is set to 1, this module generates the interrupt. This module sets the LNST bits in SYSSTS to indicate the current input state of the USB port. When the BCHG interrupt is generated, the LNST bits are read several times to remove the chattering effect until the same value is read repeatedly from the bits. A change in the USB bus state can be detected even while the internal clock supply is stopped. When the function controller mode is selected, the read value is invalid.
13	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
12	DTCH	0	R/W*	USB Disconnection Detection Interrupt Status Indicates the status of the USB disconnection detection interrupt when the host controller mode is selected. 0: DTCH interrupts not generated 1: DTCH interrupts generated This module detects the DTCH interrupt on detecting USB bus disconnection, and sets this bit to 1. Here, if the corresponding interrupt enable bit is set to 1, this module generates the interrupt. This module detects bus disconnection based on USB Specification 2.0. After detecting the DTCH interrupt, this module controls hardware as described below (irrespective of the set value of the corresponding interrupt enable bit). Software should terminate all the pipes in which communications are currently carried out for the USB port and make a transition to the wait state for bus connection to the USB port (wait state for ATTCH interrupt generation). (1) Modifies the UACT bit to 0. (2) Puts the port into the idle state. When the function controller mode is selected, the read value is invalid.

Bit	Bit Name	Initial Value	R/W	Description
11	ATTCH	0	R/W*	<p><b>ATTCH Interrupt Status</b> Indicates the status of the ATTCH interrupt when the host controller mode is selected.</p> <p>0: ATTCH interrupts not generated 1: ATTCH interrupts generated</p> <p>This module detects the ATTCH interrupt on detecting J-state or K-state of the full-speed or low-speed level signal for 2.5 <math>\mu</math>s, and sets this bit to 1. Here, if the corresponding interrupt enable bit is set to 1, this module generates the interrupt.</p> <p>Specifically, this module detects the ATTCH interrupt on any of the following conditions.</p> <ul style="list-style-type: none"> <li>• K-state, SE0, or SE1 changes to J-state, and J-state continues 2.5 <math>\mu</math>s.</li> <li>• J-state, SE0, or SE1 changes to K-state, and K-state continues 2.5 <math>\mu</math>s.</li> </ul> <p>When the function controller mode is selected, the read value is invalid.</p>
10	—	Undefined	R	<p><b>Reserved</b> The read value is undefined. The write value should always be 0.</p>
9, 8	—	All 0	R	<p><b>Reserved</b> These bits are always read as 0. The write value should always be 0.</p>
7	—	Undefined	R	<p><b>Reserved</b> The read value is undefined. The write value should always be 0.</p>
6	EOFERR	0	R/W*	<p><b>EOF Error Detection Interrupt Status</b> Indicates the status of the EOFERR interrupt when the host controller mode is selected.</p> <p>0: EOFERR interrupt not generated 1: EOFERR interrupt generated</p> <p>This module detects the EOFERR interrupt on detecting that communication is not completed at the EOF2 timing prescribed by USB Specification 2.0, and sets this bit to 1. Here, if the corresponding interrupt enable bit is set to 1, this module generates the EOFERR interrupt.</p> <p>After detecting the EOFERR interrupt, this module controls hardware as described below (irrespective of the set value of the corresponding interrupt enable bit). Software should terminate all the pipes in which communications are currently carried for the USB port and perform re-enumeration of the USB port.</p> <p>(1) Modifies the UACT bit to 0. (2) Puts the port into the idle state.</p> <p>When the function controller mode is selected, the read value is invalid.</p>
5	SIGN	0	R/W*	<p><b>Setup Transaction Error Interrupt Status</b> Indicates the status of the setup transaction error interrupt when the host controller mode is selected.</p> <p>0: SIGN interrupts not generated 1: SIGN interrupts generated</p> <p>This module detects the SIGN interrupt when ACK response is not returned from the peripheral device three consecutive times during the setup transactions issued by this module, and sets this bit to 1. Here, if the corresponding interrupt enable bit is set to 1, this module generates the SIGN interrupt.</p> <p>Specifically, this module detects the SIGN interrupt when any of the following response conditions occur for three consecutive setup transactions.</p> <p>Timeout is detected when the peripheral device has returned no response.</p> <ul style="list-style-type: none"> <li>• A damaged ACK packet is received.</li> <li>• A handshake other than ACK (NAK, NYET, or STALL) is received.</li> </ul> <p>When the function controller mode is selected, the read value is invalid.</p>



Bit	Bit Name	Initial Value	R/W	Description
4	SACK	0	R/W*	Setup Transaction Normal Response Interrupt Status Indicates the status of the setup transaction normal response interrupt when the host controller mode is selected. 0: SACK interrupts not generated 1: SACK interrupts generated This module detects the SACK interrupt when ACK response is returned from the peripheral device during the setup transactions issued by this module, and sets this bit to 1. Here, if the corresponding interrupt enable bit is set to 1, this module generates the SACK interrupt. When the function controller mode is selected, the read value is invalid.
3 to 1	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
0	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Note 1. To clear the status indicated by the bits in this register, write 0 only to the bits to be cleared; write 1 to the other bits. Do not write 0 to the status bits indicating 0.

Note 2. A change in the status indicated by the BCHG bit can be detected by this module even while the clock supply is stopped, and the interrupt is output when the corresponding interrupt enable bit is enabled. Clearing the status should be done after enabling the clock supply.

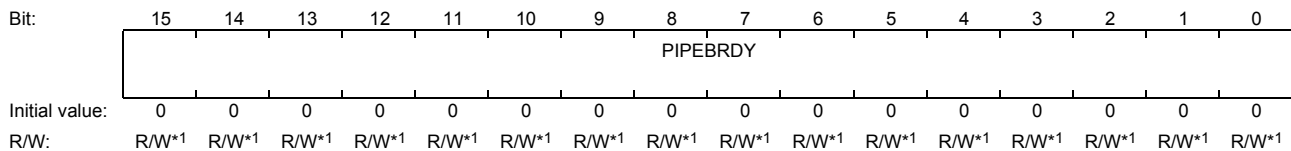
No interrupts other than BCHG can be detected while the clock supply is stopped.

\* Only 0 can be written to.

### 29.3.18 BRDY Interrupt Status Register (BRDYSTS)

BRDYSTS is a register that indicates the BRDY interrupt status for each pipe.

This register is initialized by a power-on reset.



Bit	Bit Name	Initial Value	R/W	Description
15 to 0	PIPEBRDY	H'0000	R/W*1	BRDY Interrupt Status for each Pipe*2 0: Interrupts not generated 1: Interrupts generated

Note 1. The bit number corresponds to the pipe number.

Note 2. To clear the status indicated by the bits in this register when BRDYM is 0, write 0 only to the bits to be cleared; write 1 to the other bits. Do not write 0 to the status bits indicating 0.

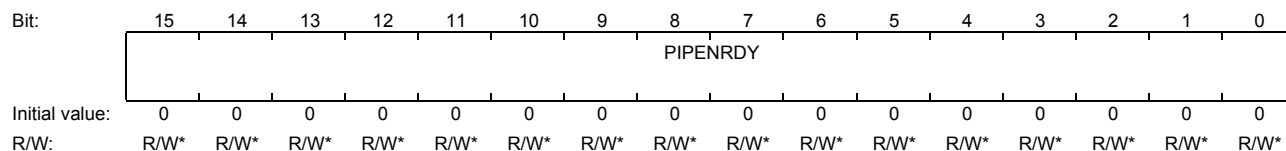
\* 1. Only 0 can be written to.

\* 2. When BRDYM is 0, clearing this bit should be done before accessing the FIFO.

### 29.3.19 NRDY Interrupt Status Register (NRDYSTS)

NRDYSTS is a register that indicates the NRDY interrupt status for each pipe.

This register is initialized by a power-on reset.



Bit	Bit Name	Initial Value	R/W	Description
15 to 10	PIPENRDY	H'0000	R/W*	NRDY Interrupt Status for each Pipe 0: Interrupts not generated 1: Interrupts generated

Note 1. The bit number corresponds to the pipe number.

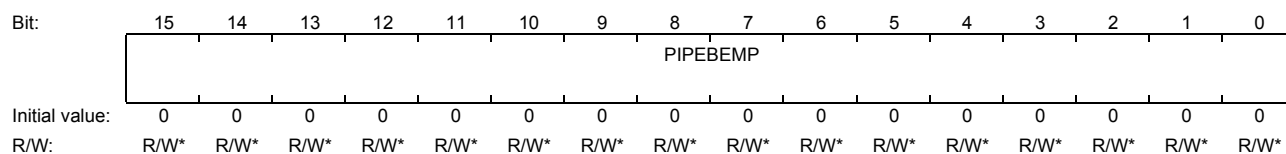
Note 2. To clear the status indicated by the bits in this register, write 0 only to the bits to be cleared; write 1 to the other bits. Do not write 0 to the status bits indicating 0.

\* Only 0 can be written to.

### 29.3.20 BEMP Interrupt Status Register (BEMPSTS)

BEMPSTS is a register that indicates the BEMP interrupt status for each pipe.

This register is initialized by a power-on reset.



Bit	Bit Name	Initial Value	R/W	Description
15 to 0	PIPEBEMP	H'0000	R/W*	BEMP Interrupt Status for each Pipe 0: Interrupts not generated 1: Interrupts generated

Note 1. The bit number corresponds to the pipe number.

Note 2. To clear the status indicated by the bits in this register, write 0 only to the bits to be cleared; write 1 to the other bits. Do not write 0 to the status bits indicating 0.

\* Only 0 can be written to.

### 29.3.21 Frame Number Register (FRMNUM)

FRMNUM is a register that determines the source of isochronous error notification and indicates the frame number. This register is initialized by a power-on reset.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OVRN	CRCE	—	—	—											
Initial value:	0	0	—	—	—	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W*1	R/W*1	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	OVRN	0	R/W*1	<p>Overrun/Underrun Detection Status</p> <p>Indicates whether an overrun/underrun error has been detected in the pipe during isochronous transfer. On detecting either error, this module simultaneously generates the internal NRDY interrupt request. For details, refer to section 29.4.2, Interrupt Functions.</p> <p>0: No error 1: An error occurred</p> <p>This bit can be cleared to 0 by writing 0 to the bit. Here, 1 should be written to the other bits in this register.</p> <p>(1) When the host controller mode is selected</p> <p>This module sets this bit to 1 on any of the following conditions.</p> <ul style="list-style-type: none"> <li>For the isochronous transfer pipe in the transmitting direction, the time to issue an OUT token comes before all the transmit data has been written to the FIFO buffer.</li> <li>For the isochronous transfer pipe in the receiving direction, the time to issue an IN token comes when no FIFO buffer planes are empty.</li> </ul> <p>(2) When the function controller mode is selected</p> <p>This module sets this bit to 1 on any of the following conditions.</p> <ul style="list-style-type: none"> <li>For the isochronous transfer pipe in the transmitting direction, the IN token is received before all the transmit data has been written to the FIFO buffer.</li> <li>For the isochronous transfer pipe in the receiving direction, the OUT token is received when no FIFO buffer planes are empty.</li> </ul> <p>Note: This bit should be used for debugging. When designing a system, control the timing so that neither overrun nor underrun occurs.</p>
14	CRCE	0	R/W*1	<p>Receive Data Error</p> <p>Indicates whether a CRC error or bit stuffing error has been detected in the pipe during isochronous transfer. On detecting either error, this module simultaneously generates the internal NRDY interrupt request. For details, refer to section 29.4.2, Interrupt Functions.</p> <p>0: No error 1: An error occurred</p> <p>This bit can be cleared to 0 by writing 0 to the bit. Here, 1 should be written to the other bits in this register.</p>
13 to 11	—	Undefined	R	<p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p>
10 to 0	FRNM[10:0]	H'000	R	<p>Frame Number</p> <p>This module sets these bits to indicate the latest frame number, which is updated every time an SOF packet is issued or received (every 1 ms). When reading these bits, repeat reading until the same value is read twice.</p>

Note 1. Only 0 can be written to.

### 29.3.22 $\mu$ Frame Number Register (UFRMNUM)

UFRMNUM is a register that indicates the  $\mu$ frame number.

This register is initialized by a power-on reset.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	UFRNM[2:0]		
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 3	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
2 to 0	UFRNM[2:0]	000	R	$\mu$ Frame The $\mu$ frame number can be confirmed. This module sets these bits to indicate the $\mu$ frame number during high-speed operation. During operation other than high-speed operation, this module sets these bits to B'000. When reading these bits, repeat reading until the same value is read twice.

### 29.3.23 USB Address Register (USBADDR)

USBADDR is a register that indicates the USB address. This register is valid only when the function controller mode is selected. When the host controller mode is selected, peripheral device addresses should be set using the DEVSEL bits in PIPEMAXP.

This register is initialized by a power-on reset or a USB bus reset.

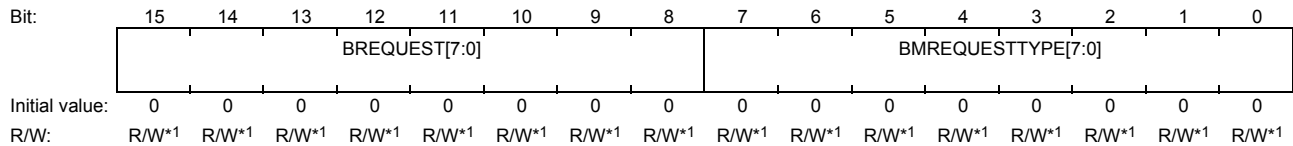
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	USBADDR[6:0]						
Initial value:	—	—	—	—	—	—	—	—	—	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 7	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
6 to 0	USBADDR[6:0]	H'00	R	USB Address When the function controller mode is selected, these bits indicate the USB address assigned by the host when the SET_ADDRESS request is successfully processed. When the function controller mode is selected, these bits indicate H'00 upon detection of a USB bus reset signal. When the host controller mode is selected, the read value is invalid.

### 29.3.24 USB Request Type Register (USBREQ)

USBREQ is a register that stores setup requests for control transfers.

When the function controller mode is selected, the values of bRequest and bmRequestType that have been received are stored. When the host controller mode is selected, the values of bRequest and bmRequestType to be transmitted are set. This register is initialized by a power-on reset or a USB bus reset.



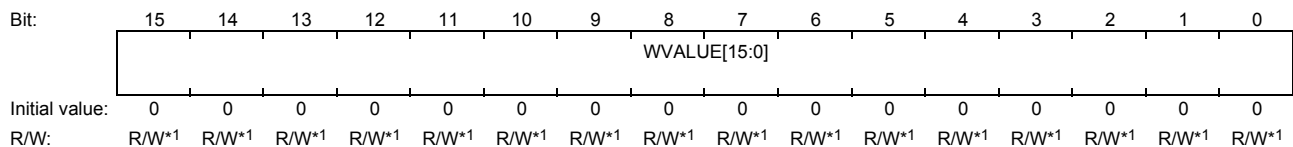
Bit	Bit Name	Initial Value	R/W	Description
15 to 8	BREQUEST[7:0]	H'00	R/W*1	<b>Request</b> These bits store the USB request bRequest value. (1) When the host controller mode is selected The USB request data value for the setup transaction to be transmitted should be set in these bits. After SUREQ has been set to 1, do not modify these bits until SUREQ is read as 0. (2) When the function controller mode is selected Indicates the USB request data value received during the setup transaction. Writing to these bits is invalid.
7 to 0	BMREQUEST TYPE[7:0]	H'00	R/W*1	<b>Request Type</b> These bits store the USB request bmRequestType value. (1) When the host controller mode is selected The USB request type value for the setup transaction to be transmitted should be set in these bits. After SUREQ has been set to 1, do not modify these bits until SUREQ is read as 0. (2) When the function controller mode is selected Indicates the USB request type value received during the setup transaction. Writing to these bits is invalid.

Note 1. When the function controller mode is selected, these bits can only be read, and writing to these bits is invalid. When the host controller mode is selected, these bits can be read and written to.

### 29.3.25 USB Request Value Register (USBVAL)

USBVAL is a register that stores setup requests for control transfers. When the function controller mode is selected, the value of wValue that has been received is stored. When the host controller mode is selected, the value of wValue to be transmitted is set.

This register is initialized by a power-on reset or a USB bus reset.



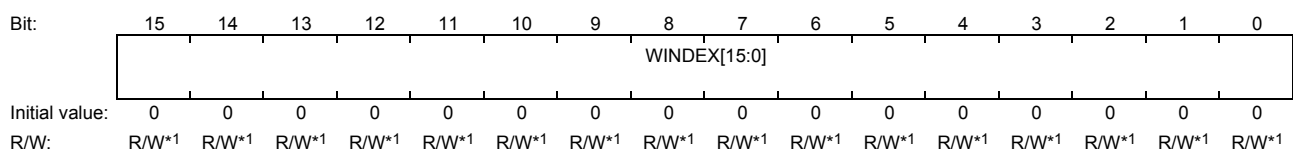
Bit	Bit Name	Initial Value	R/W	Description
15 to 0	WVALUE[15:0]	H'0000	R/W*1	Value These bits store the USB request wValue value. (1) When the host controller mode is selected The USB request wValue value for the setup transaction to be transmitted should be set in these bits. After SUREQ has been set to 1, do not modify these bits until SUREQ is read as 0. (2) When the function controller mode is selected Indicates the USB request wValue value received during the setup transaction. Writing to these bits is invalid.

Note 1. When the function controller mode is selected, these bits can only be read, and writing to these bits is invalid. When the host controller mode is selected, these bits can be read and written to.

### 29.3.26 USB Request Index Register (USBINDX)

USBINDX is a register that stores setup requests for control transfers. When the function controller mode is selected, the value of wIndex that has been received is stored. When the host controller mode is selected, the value of wIndex to be transmitted is set.

This register is initialized by a power-on reset or a USB bus reset.



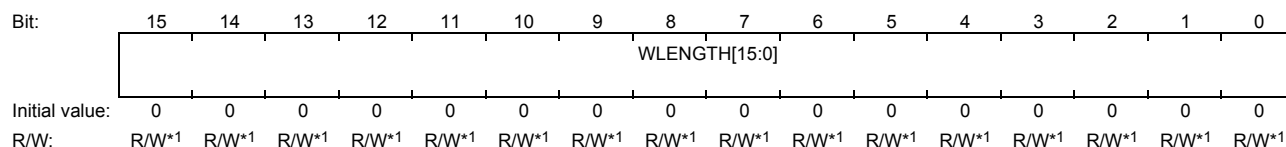
Bit	Bit Name	Initial Value	R/W	Description
15 to 0	WINDEX[15:0]	H'0000	R/W*1	Index These bits store the USB request wIndex value. (1) When the host controller mode is selected The USB request wIndex value for the setup transaction to be transmitted should be set in these bits. After SUREQ has been set to 1, do not modify these bits until SUREQ is read as 0. (2) When the function controller mode is selected Indicates the USB request wIndex value received during the setup transaction. Writing to these bits is invalid.

Note 1. When the function controller mode is selected, these bits can only be read, and writing to these bits is invalid. When the host controller mode is selected, these bits can be read and written to.

### 29.3.27 USB Request Length Register (USBLENG)

USBLENG is a register that stores setup requests for control transfers. When the function controller mode is selected, the value of wLength that has been received is stored. When the host controller mode is selected, the value of wLength to be transmitted is set.

This register is initialized by a power-on reset or a USB bus reset.



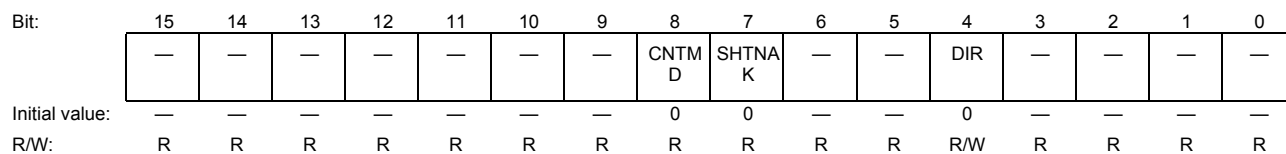
Bit	Bit Name	Initial Value	R/W	Description
15 to 0	WLENGTH[15:0]	H'0000	R/W*1	Length These bits store the USB request wLength value. (1) When the host controller mode is selected The USB request wLength value for the setup transaction to be transmitted should be set in these bits. After SUREQ has been set to 1, do not modify these bits until SUREQ is read as 0. (2) When the function controller mode is selected Indicates the USB request wLength value received during the setup transaction. Writing to these bits is invalid.

Note 1. When the function controller mode is selected, these bits can only be read, and writing to these bits is invalid. When the host controller mode is selected, these bits can be read and written to.

### 29.3.28 DCP Configuration Register (DCPCFG)

DCPCFG is a register that specifies the data transfer direction for the default control pipe (DCP).

This register is initialized by a power-on reset.



Bit	Bit Name	Initial Value	R/W	Description
15 to 9	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
8	CNTMD	0	R/W	<p>Continuous Transfer Mode</p> <p>Specifies whether to use the default control pipe in continuous transfer mode.</p> <p>0: Non-continuous transfer mode 1: Continuous transfer mode</p> <p>This module determines whether transmitting to/receiving from the FIFO buffer allocated for the DCP has completed or not using this bit setting, as shown in Table 29.10 .</p> <p>Modify this bit while CSSTS is 0, PID is NAK, and the pipe is not selected by the CURPIPE bits.</p> <p>To modify this bit after completing USB communication using the DCP, write 1 to BCLR to clear the FIFO buffer assigned to the DCP while the CSSTS, PID, and CURPIPE bits are in the above-described state.</p> <p>Before modifying these bits after modifying the PID bits for the DCP from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, the PBUSY bit does not have to be checked.</p>
7	SHTNAK	0	R/W	<p>Pipe Disabled at End of Transfer</p> <p>Specifies whether to modify PID to NAK upon the end of transfer when the default control pipe is in the receiving direction.</p> <p>0: Pipe continued at the end of transfer 1: Pipe disabled at the end of transfer</p> <p>When this bit is set to 1, this module modifies the PID bits corresponding to the DCP to NAK on determining the end of the transfer. This module determines that the transfer has ended when a short packet (including a zero-length packet) is successfully received.</p> <p>Modify this bit while CSSTS is 0 and PID is NAK.</p> <p>Before modifying this bit after modifying the PID bits for the DCP from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, the PBUSY bit does not have to be checked.</p> <p>This bit should be cleared to 0 for the DCP in the transmitting direction.</p>
6, 5	—	Undefined	R	<p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p>
4	DIR	0	R/W	<p>Transfer Direction</p> <p>When the host controller mode is selected, this bit sets the transfer direction of data stage and status stage for control transfers.</p> <p>0: Data receiving direction 1: Data transmitting direction</p> <p>When the function controller mode is selected, this bit should be cleared to 0.</p>
3 to 0	—	Undefined	R	<p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p>



**Table 29.10 Relationship between Transfer Mode Settings by CNTMD Bit and Timings at which Reading Data or Transmitting Data from FIFO Buffer is Enabled**

Continuous or Non-Continuous Transfer Mode	When Reading Data or Transmitting Data is Enabled
Non-continuous transfer (CNTMD = 0)	<p>In the receiving direction (DIR = 0), reading data from the FIFO buffer is enabled in the following case.</p> <p>This module receives a single packet.</p> <hr/> <p>In the transmitting direction (DIR = 1), transmitting data from the FIFO buffer is enabled in either of the following cases.</p> <p>(1) Data of the maximum packet size is written to the FIFO buffer.</p> <p>(2) Data of the short packet size (including 0-byte data) is written to the FIFO buffer and then 1 is written to the BVAL bit</p>
Continuous transfer (CNTMD = 1)	<p>In the receiving direction (DIR = 0), reading data from the FIFO buffer is enabled in any of the following cases.</p> <p>(1) The number of the data bytes received in the FIFO buffer assigned to the DCP becomes the same as the number of assigned data bytes (fixed to 256 bytes)</p> <p>(2) This module receives a short packet other than a zero-length packet</p> <p>(3) This module receives a zero-length packet when data is already stored in the FIFO buffer assigned to the DCP.</p> <hr/> <p>In the transmitting direction (DIR = 1), transmitting data from the FIFO buffer is enabled in either of the following cases.</p> <p>(1) The number of the data bytes written to the FIFO buffer becomes the same as the number of data bytes in a single FIFO buffer plane assigned to the DCP.</p> <p>(2) A number of data bytes less than the size of a single FIFO buffer (including 0-byte data) assigned to the DCP being written to the FIFO buffer and then 1 being written to BVAL.</p>

### 29.3.29 DCP Maximum Packet Size Register (DCPMAXP)

DCPMAXP is a register that specifies the maximum packet size for the DCP.

This register is initialized by a power-on reset.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	DEVSEL[3:0]				—	—	—	—	—	MXPS[6:0]							
Initial value:	0	0	0	0	—	—	—	—	—	1	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	DEVSEL[3:0]	0000	R/W	<p><b>Device Select</b></p> <p>When the host controller mode is selected, these bits specify the communication target peripheral device address for control transfers.</p> <p>0000: Address 0000 0001: Address 0001 : : 1001: Address 1001 1010: Address 1010 Others: Setting prohibited</p> <p>These bits should be set after setting the DEVADDn register corresponding to the value to be set in these bits. For example, before setting DEVSEL to 0010, the DEVADD2 register should be set first.</p> <p>These bits should be set while CSSTS is 0, PID is NAK, and SUREQ is 0. Before modifying these bits after modifying the PID bits for the DCP from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, the PBUSY bit does not have to be checked.</p> <p>When the function controller mode is selected, these bits should be set to B'0000.</p>
11 to 7	—	Undefined	R	<p><b>Reserved</b></p> <p>The read value is undefined. The write value should always be 0.</p>
6 to 0	MXPS[6:0]	H'40	R/W	<p><b>Maximum Packet Size</b></p> <p>Specifies the maximum data payload (maximum packet size) for the DCP. These bits are initialized to H'40 (64 bytes). These bits should be set to the value based on the USB Specification. These bits should be set while CSSTS is 0, PID is NAK, and the pipe is not selected by the CURPIPE bits. Before modifying these bits after modifying the PID bits for the DCP from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, the PBUSY bit does not have to be checked. While MXPS is 0, do not write to the FIFO buffer or do not set PID to BUF.</p>

### 29.3.30 DCP Control Register (DCPCTR)

DCPCTR is a register that is used to confirm the buffer memory status, change and confirm the data PID sequence bit, and set the response PID for the DCP.

This register is initialized by a power-on reset. The CCPL and PID[1:0] bits are initialized by a USB bus reset.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BSTS	SURE Q	CSCLR	CSSTS	SURE QCLR	—	—	SQCLR	SQSET	SQMO N	PBUSY	PINGE	—	CCPL	PID[1:0]	
Initial value:	0	0	0	0	0	—	—	0	0	1	0	0	—	0	0	0
R/W:	R	R/W <sup>2</sup>	R/W <sup>*1</sup>	R	R/W <sup>*1</sup>	R	R	R/W <sup>*1</sup>	R/W <sup>*1</sup>	R	R	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	BSTS	0	R	<p>Buffer Status</p> <p>Indicates whether DCP FIFO buffer access is enabled or disabled.</p> <p>0: Buffer access is disabled.</p> <p>1: Buffer access is enabled.</p> <p>The meaning of the BSTS bit depends on the ISEL bit setting as follows.</p> <ul style="list-style-type: none"> <li>• When ISEL = 0, BSTS indicates whether the received data can be read from the buffer.</li> <li>• When ISEL = 1, BSTS indicates whether the data to be transmitted can be written to the buffer.</li> </ul>
14	SUREQ	0	R/W <sup>2</sup>	<p>Setup Token Transmission</p> <p>Transmits the setup packet by setting this bit to 1 when the host controller mode is selected.</p> <p>0: Writing invalid</p> <p>1: Transmits the setup packet.</p> <p>After completing the setup transaction process, this module generates either the SACK or SIGN interrupt and clears this bit to 0.</p> <p>This module also clears this bit to 0 when the SUREQCLR bit is set to 1.</p> <p>Before setting this bit to 1, set the DEVSEL bits, USBREQ register, USBVAL register, USBINDX register, and USBLENG register appropriately to transmit the desired USB request in the setup transaction.</p> <p>Before setting this bit to 1, check that the PID bits for the DCP are set to NAK. After setting this bit to 1, do not modify the DEVSEL bits, USBREQ register, USBVAL register, USBINDX register, or USBLENG register while the setup transaction is in progress (SUREQ = 1).</p> <p>Write 1 to this bit only when transmitting the setup token; for the other purposes, write 0.</p> <p>When the function controller mode is selected, be sure to write 0 to this bit.</p>
13	CSCLR	0	R/W <sup>*1</sup>	<p>C-SPLIT Status Clear for Split Transaction</p> <p>When the host controller mode is selected, setting this bit to 1 clears the CSSTS bit to 0 for the transfer using the split transaction. In this case, the next DCP transfer restarts with the S-SPLIT.</p> <p>0: Writing invalid</p> <p>1: Clears the CSSTS bit to 0.</p> <p>When this bit is set to 1, this module clears the CSSTS bit to 0.</p> <p>For the transfer using the split transaction, to restart the next transfer with the S-SPLIT forcibly, set this bit to 1. However, for the normal split transaction, this module automatically clears the CSSTS bit to 0 upon completion of the C-SPLIT; therefore, clearing the CSSTS bit is not necessary.</p> <p>Controlling the CSSTS bit through this bit must be done while UACT is 0 and thus communication is halted or while no transfer is being performed with bus disconnection detected.</p> <p>Setting this bit to 1 while CSSTS is 0 has no effect.</p> <p>When the function controller mode is selected, be sure to write 0 to this bit.</p>

Bit	Bit Name	Initial Value	R/W	Description
12	CSSTS	0	R	<p>COMPLETE SPLIT (C-SPLIT) Status of Split Transaction Indicates the C-SPLIT status of the split transaction when the host controller mode is selected. 0: START-SPLIT (S-SPLIT) transaction being processed or the device not using the split transaction being processed 1: C-SPLIT transaction being processed This module sets this bit to 1 upon start of the C-SPLIT and clears this bit to 0 upon detection of C-SPLIT completion. When the function controller mode is selected, the read value is invalid.</p>
11	SUREQCLR	0	R/W <sup>*1</sup>	<p>SUREQ Bit Clear When the host controller mode is selected, setting this bit to 1 clears the SUREQ bit to 0. 0: Writing invalid 1: Clears the SUREQ bit to 0. This bit always indicates 0. Set this bit to 1 when communication has stopped with SUREQ being 1 during the setup transaction. However, for normal setup transactions, this module automatically clears the SUREQ bit to 0 upon completion of the transaction; therefore, clearing the SUREQ bit is not necessary. Controlling the SUREQ bit through this bit must be done while UACT is 0 and thus communication is halted or while no transfer is being performed with bus disconnection detected. When the function controller mode is selected, be sure to write 0 to this bit.</p>
10, 9	—	Undefined	R	<p>Reserved The read value is undefined. The write value should always be 0.</p>
8	SQCLR	0	R/W <sup>*1</sup>	<p>Toggle Bit Clear Specifies DATA0 as the expected value of the sequence toggle bit for the next transaction during the DCP transfer. 0: Writing invalid 1: Specifies DATA0. This bit always indicates 0. Do not set the SQCLR and SQSET bits to 1 simultaneously. Set this bit to 1 while CSSTS is 0, PID is NAK, and the pipe is not selected by the CURPIPE bits. Before setting this bit to 1 after modifying the PID bits for the DCP from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, the PBUSY bit does not have to be checked.</p>
7	SQSET	0	R/W <sup>*1</sup>	<p>Toggle Bit Set Specifies DATA1 as the expected value of the sequence toggle bit for the next transaction during the DCP transfer. 0: Writing invalid 1: Specifies DATA1. Do not set the SQCLR and SQSET bits to 1 simultaneously. Set this bit to 1 while CSSTS is 0 and PID is NAK. Before setting this bit to 1 after modifying the PID bits for the DCP from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, the PBUSY bit does not have to be checked.</p>
6	SQMON	1	R	<p>Sequence Toggle Bit Monitor Indicates the expected value of the sequence toggle bit for the next transaction during the DCP transfer. 0: DATA0 1: DATA1 This module allows this bit to toggle upon normal completion of the transaction. However, this bit is not allowed to toggle when a DATA-PID disagreement occurs during the transfer in the receiving direction. When the function controller mode is selected, this module sets this bit to 1 (specifies DATA1 as the expected value) upon normal reception of the setup packet. When the function controller mode is selected, this module does not reference to this bit during the IN/OUT transaction of the status stage, and does not allow this bit to toggle upon normal completion.</p>

Bit	Bit Name	Initial Value	R/W	Description
5	PBUSY	0	R	<p>Pipe Busy</p> <p>Indicates whether or not the actual communication state of the DCP has entered the NAK state when the PID bits for the DCP are changed from BUF to NAK.</p> <p>0: Has not finished the transition to NAK 1: Has finished the transition to NAK</p> <p>This module modifies this bit from 0 to 1 upon start of the USB transaction for the DCP, and modifies the bit from 1 to 0 upon completion of one transaction.</p> <p>Reading this bit after PID has been set to NAK allows checking whether modification of the pipe settings is possible.</p> <p>For details, refer to section 29.4.3 (1) Pipe Control Register Switching Procedures.</p>
4	PINGE	0	R/W	<p>PING Token Issue Enable</p> <p>When the host controller mode is selected, setting this bit to 1 allows this module to issue the PING token during transfers in the transmitting direction and start a transfer in the transmitting direction with the PING transaction.</p> <p>0: Disables issuing PING token. 1: Enables normal PING operation.</p> <p>When having detected the ACK handshake during PING transactions, this module performs the OUT transaction as the next transaction.</p> <p>When having detected the NAK handshake or NYET handshake during OUT transactions, this module performs the PING transaction as the next transaction.</p> <p>When the host controller mode is selected, setting this bit to 0 prevents this module from issuing the PING token during transfers in the transmitting direction and only allows this module to perform OUT transactions for the transfers in the transmitting direction.</p> <p>This bit should be modified while CSSTS is 0 and PID is NAK.</p> <p>Before setting this bit to 1 after modifying the PID bits for the DCP from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, the PBUSY bit does not have to be checked.</p> <p>When the function controller mode is selected, be sure to write 0 to this bit.</p>
3	—	Undefined	R	<p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p>
2	CCPL	0	R/W	<p>Control Transfer End Enable</p> <p>When the function controller mode is selected, setting this bit to 1 enables the status stage of the control transfer to be completed.</p> <p>0: Completion of control transfer is disabled. 1: Completion of control transfer is enabled.</p> <p>In function controller mode, when this bit is set to 1 while the corresponding PID bits are set to BUF, this module completes the control transfer status stage.</p> <p>Specifically, during control read transfer, this module transmits the ACK handshake in response to the OUT transaction from the USB host, and outputs the zero-length packet in response to the IN transaction from the USB host during control write or no-data control transfer. However, on detecting the SET_ADDRESS request, this module operates in auto response mode from the setup stage up to the status stage completion irrespective of the setting of this bit.</p> <p>This module modifies this bit from 1 to 0 on receiving the new setup packet.</p> <p>1 cannot be written to this bit while VALID is 1.</p> <p>When the host controller mode is selected, be sure to write 0 to this bit.</p>

Bit	Bit Name	Initial Value	R/W	Description
1, 0	PID[1:0]	00	R/W	<p>Response PID Controls the response type of this module during control transfer.</p> <p>00: NAK response 01: BUF response (depending on the buffer state) 10: STALL response 11: STALL response</p> <p>(1) When the host controller mode is selected Modify the setting of these bits from NAK to BUF using the following procedure.</p> <ul style="list-style-type: none"> <li>When the transmitting direction is set Write all the transmit data to the FIFO buffer while UACT is 1 and PID is NAK, and then set PID to BUF. After PID has been set to BUF, this module executes the OUT transaction (or PING transaction).</li> <li>When the receiving direction is set Check that the FIFO buffer is empty (or empty the buffer) while UACT is 1 and PID is NAK, and then set PID to BUF. After PID has been set to BUF, this module executes the IN transaction.</li> </ul> <p>This module modifies the setting of these bits as follows.</p> <ul style="list-style-type: none"> <li>This module sets PID to STALL (11) on receiving the data of the size exceeding the maximum packet size when PID has been set to BUF.</li> <li>This module sets PID to NAK on detecting a reception error such as a CRC error three consecutive times.</li> <li>This module also sets PID to STALL (11) on receiving the STALL handshake.</li> </ul> <p>Even if the PID bits are modified to NAK after this module has issued S-SPLIT of the split transaction for the selected pipe (while CSSTS indicates 1), this module continues the transaction until C-SPLIT completes. This module sets PID to NAK upon completion of C-SPLIT.</p> <p>(2) When the function controller mode is selected This module modifies the setting of these bits as follows.</p> <ul style="list-style-type: none"> <li>This module modifies PID to NAK on receiving the setup packet. Here, this module sets VALID to 1. The setting of PID cannot be modified until VALID is set to 0.</li> <li>This module sets PID to STALL (11) on receiving the data of the size exceeding the maximum packet size when PID has been set to BUF.</li> <li>This module sets PID to STALL (1x) on detecting the control transfer sequence error.</li> <li>This module sets PID to NAK on detecting the USB bus reset.</li> </ul> <p>This module does not reference to the setting of the PID bits while the SET_ADDRESS request is processed (auto processing).</p>

Note 1. Only 0 can be read and 1 can be written to.

Note 2. Only 1 can be written to.

### 29.3.31 Pipe Window Select Register (PIPESEL)

PIPE1 to PIPE9 should be set using PIPESEL, PIPECFG, PIPEBUF, PIPEMAXP, PIPEPERI, PIPEnCTR, PIPEnTRE, and PIPEnTRN.

After selecting the pipe using PIPESEL, functions of the pipe should be set using PIPECFG, PIPEBUF, PIPEMAXP, and PIPEPERI. PIPEnCTR, PIPEnTRE, and PIPEnTRN can be set regardless of the pipe selection in PIPESEL.

This register is initialized by a power-on reset.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	PIPESEL[3:0]			
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 4	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
3 to 0	PIPESEL[3:0]	0000	R/W	Pipe Window Select When a value between 0001 and 1111 is set in these bits, the information and settings for the corresponding pipe can be read from the PIPECFG, PIPEBUF, PIPEMAXP, and PIPEPERI registers. 0000: No pipe selected 0001: PIPE1 0010: PIPE2 0011: PIPE3 0100: PIPE4 0101: PIPE5 0110: PIPE6 0111: PIPE7 1000: PIPE8 1001: PIPE9 1010: PIPE10 1011: PIPE11 1100: PIPE12 1101: PIPE13 1110: PIPE14 1111: PIPE15 Others: Setting prohibited When 0000 is set in these bits, 0 is read from all of the bits in the PIPECFG, PIPEBUF, PIPEMAXP, and PIPEPERI registers. Writing to the bits in these registers is invalid.

### 29.3.32 Pipe Configuration Register (PIPECFG)

PIPECFG is a register that specifies the transfer type, buffer memory access direction, and endpoint numbers for PIPE1 to PIPE15. It also selects continuous or non-continuous transfer mode, single or double buffer mode, and whether to continue or disable pipe operation at the end of transfer.

This register is initialized by a power-on reset.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TYPE[1:0]		—	—	—	BFRE	DBLB	CNTM D	SHT NAK	—	—	DIR	EPNUM[3:0]			
Initial value:	0	0	—	—	—	0	0	0	0	—	—	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15, 14	TYPE[1:0]	00	R/W	<p>Transfer Type</p> <p>Specifies the transfer type for the pipe selected by the PIPESEL bits (selected pipe).</p> <ul style="list-style-type: none"> <li>• PIPE1 and PIPE2 <ul style="list-style-type: none"> <li>00: Pipe not used</li> <li>01: Bulk transfer</li> <li>10: Setting prohibited</li> <li>11: Isochronous transfer</li> </ul> </li> <li>• PIPE3 to PIPE5 <ul style="list-style-type: none"> <li>00: Pipe not used</li> <li>01: Bulk transfer</li> <li>10: Setting prohibited</li> <li>11: Setting prohibited</li> </ul> </li> <li>• PIPE6 to PIPE9 <ul style="list-style-type: none"> <li>00: Pipe not used</li> <li>01: Bulk transfer (when in function controller mode)</li> <li>10: Interrupt transfer (when in host controller mode)</li> <li>11: Setting prohibited</li> </ul> </li> <li>• PIPE10 (available only in function controller mode) <ul style="list-style-type: none"> <li>01: Bulk transfer</li> <li>10: Interrupt transfer</li> </ul> </li> <li>• PIPE11 to PIPE15 (available only in function controller mode) <ul style="list-style-type: none"> <li>01: Bulk transfer</li> </ul> </li> </ul> <p>Before setting PID to BUF for the selected pipe (before starting USB communication using the selected pipe), be sure to set these bits to the value other than 00.</p> <p>Modify these bits while the PID bits for the selected pipe are set to NAK. Before modifying these bits after modifying the PID bits for the selected pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, the PBUSY bit does not have to be checked.</p>
13 to 11	—	Undefined	R	<p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p>



Bit	Bit Name	Initial Value	R/W	Description
10	BFRE	0	R/W	<p>BRDY Interrupt Operation Specification</p> <p>Specifies the BRDY interrupt generation timing from this module to the CPU with respect to the selected pipe.</p> <p>0: BRDY interrupt upon transmitting or receiving data 1: BRDY interrupt upon completion of reading data</p> <p>This bit is valid when PIPE1 to PIPE5 and PIPE9 to PIPE15 are selected. When this bit is set to 1 and the selected pipe is in the receiving direction, this module detects the transfer completion and generates the BRDY interrupt on having read the pertinent packet.</p> <p>When the BRDY interrupt is generated with the above conditions, 1 needs to be written to BCLR. The FIFO buffer assigned to the selected pipe is not enabled for reception until 1 is written to BCLR.</p> <p>When this bit is set to 1 and the selected pipe is in the transmitting direction, this module does not generate the BRDY interrupt.</p> <p>For details, refer to section 29.4.2 (2) BRDY Interrupt.</p> <p>Modify this bit while CSSTS is 0, PID is NAK, and the pipe is not selected by the CURPIPE bits.</p> <p>To modify this bit after completing USB communication using the selected pipe, write 1 and then 0 to ACLRM continuously to clear the FIFO buffer assigned to the selected pipe while the CSSTS, PID, and CURPIPE bits are in the above-described state.</p> <p>Before modifying this bit after modifying the PID bits for the selected pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, the PBUSY bit does not have to be checked.</p>
9	DBLB	0	R/W	<p>Double Buffer Mode</p> <p>Selects either single or double buffer mode for the FIFO buffer used by the selected pipe.</p> <p>0: Single buffer 1: Double buffer</p> <p>This bit is valid when PIPE1 to PIPE5 and PIPE9 to PIPE15 are selected. For PIPE9 and PIPE10, this bit is only valid when the transfer type is set to bulk transfer.</p> <p>When this bit is set to 1, this module assigns two planes of the FIFO buffer size specified by the BUFSIZE bits in PIPEBUF to the selected pipe.</p> <p>Specifically, the following expression determines the FIFO buffer size assigned to the selected pipe by this module.</p> $(BUFSIZE + 1) \times 64 \times (DBLB + 1) \text{ [bytes]}$ <p>Modify this bit while CSSTS is 0, PID is NAK, and the pipe is not selected by the CURPIPE bits.</p> <p>To modify this bit after completing USB communication using the selected pipe, write 1 and then 0 to ACLRM continuously to clear the FIFO buffer assigned to the selected pipe while the CSSTS, PID, and CURPIPE bits are in the above-described state.</p> <p>Before modifying this bit after modifying the PID bits for the selected pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, the PBUSY bit does not have to be checked.</p>

Bit	Bit Name	Initial Value	R/W	Description
8	CNTMD	0	R/W	<p>Continuous Transfer Mode</p> <p>Specifies whether to use the selected pipe in continuous transfer mode.</p> <p>0: Non-continuous transfer mode 1: Continuous transfer mode</p> <p>This bit is valid when PIPE1 to PIPE5 and PIPE9 to PIPE15 are selected by the PIPESEL bits and bulk transfer is selected (TYPE = 01). This module determines whether transmitting to/receiving from the FIFO buffer allocated for the selected pipe has completed or not using this bit setting, as shown in Table 29.11.</p> <p>Modify this bit while CSSTS is 0, PID is NAK, and the pipe is not selected by the CURPIPE bits.</p> <p>To modify this bit after completing USB communication using the selected pipe, write 1 and then 0 to ACLRM continuously to clear the FIFO buffer assigned to the selected pipe while the CSSTS, PID, and CURPIPE bits are in the above-described state.</p> <p>Before modifying this bit after modifying the PID bits for the selected pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, the PBUSY bit does not have to be checked.</p>
7	SHTNAK	0	R/W	<p>Pipe Disabled at End of Transfer</p> <p>Specifies whether to modify PID to NAK upon the end of transfer when the selected pipe is in the receiving direction.</p> <p>0: Pipe continued at the end of transfer 1: Pipe disabled at the end of transfer</p> <p>This bit is valid when the selected pipe is PIPE1 to PIPE5 and PIPE9 to PIPE15 in the receiving direction.</p> <p>When this bit is set to 1 for the selected pipe in the receiving direction, this module modifies the PID bits corresponding to the selected pipe to NAK on determining the end of the transfer. This module determines that the transfer has ended on any of the following conditions.</p> <ul style="list-style-type: none"> <li>• A short packet (including a zero-length packet) is successfully received.</li> <li>• The transaction counter is used and the number of packets specified by the counter is successfully received.</li> </ul> <p>Modify this bit while CSSTS is 0 and PID is NAK.</p> <p>Before modifying this bit after modifying the PID bits for the selected pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, the PBUSY bit does not have to be checked.</p> <p>This bit should be cleared to 0 for the pipe in the transmitting direction.</p>
6	—	Undefined	R	<p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p>
4	DIR	0	R/W	<p>Transfer Direction</p> <p>Specifies the transfer direction for the selected pipe.</p> <p>0: Receiving direction 1: Transmitting direction</p> <p>When this bit is set to 0, this module uses the selected pipe in the receiving direction, and when this bit is set to 1, this module uses the selected pipe in the transmitting direction.</p> <p>Modify this bit while CSSTS is 0, PID is NAK, and the pipe is not selected by the CURPIPE bits.</p> <p>To modify this bit after completing USB communication using the selected pipe, write 1 and then 0 to ACLRM continuously to clear the FIFO buffer assigned to the selected pipe while the CSSTS, PID, and CURPIPE bits are in the above-described state.</p> <p>Before modifying this bit after modifying the PID bits for the selected pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, the PBUSY bit does not have to be checked.</p>

Bit	Bit Name	Initial Value	R/W	Description
3 to 0	EPNUM[3:0]	0000	R/W	<p>Endpoint Number</p> <p>These bits specify the endpoint number for the selected pipe. Setting 0000 means unused pipe. Modify these bits while CSSTS is 0, PID is NAK, and the pipe is not selected by the CURPIPE bits. Before modifying these bits after modifying the PID bits for the selected pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, the PBUSY bit does not have to be checked. Do not make the settings such that the combination of the set values in the DIR and EPNUM bits should be the same for two or more pipes (EPNUM = 0000 can be set for all the pipes).</p>

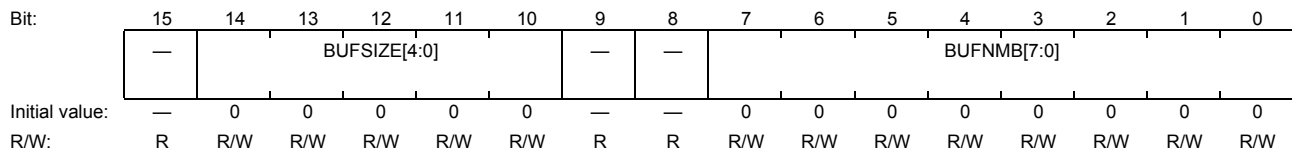
**Table 29.11 Relationship between Transfer Mode Settings by CNTMD Bit and Timings at which Reading Data or Transmitting Data from FIFO Buffer is Enabled**

Continuous or Non-Continuous Transfer Mode	When Reading Data or Transmitting Data is Enabled
Non-continuous transfer (CNTMD = 0)	<p>In the receiving direction (DIR = 0), reading data from the FIFO buffer is enabled in the following case. This module receives a single packet.</p> <hr/> <p>In the transmitting direction (DIR = 1), transmitting data from the FIFO buffer is enabled in either of the following cases.</p> <ol style="list-style-type: none"> <li>(1) Data of the maximum packet size is written to the FIFO buffer.</li> <li>(2) Data of the short packet size (including 0-byte data) is written to the FIFO buffer and then 1 is written to BVAL.</li> </ol>
Continuous transfer (CNTMD = 1)	<p>In the receiving direction (DIR = 0), reading data from the FIFO buffer is enabled in any of the following cases.</p> <ol style="list-style-type: none"> <li>(1) The number of the data bytes received in the FIFO buffer assigned to the selected pipe becomes the same as the number of assigned data bytes ((BUFSIZE + 1) × 64).</li> <li>(2) This module receives a short packet other than a zero-length packet.</li> <li>(3) This module receives a zero-length packet when data is already stored in the FIFO buffer assigned to the selected pipe.</li> <li>(4) This module receives the number of packets equal to the transaction counter value specified for the selected pipe.</li> </ol> <hr/> <p>In the transmitting direction (DIR = 1), transmitting data from the FIFO buffer is enabled in any of the following cases.</p> <ol style="list-style-type: none"> <li>(1) The number of the data bytes written to the FIFO buffer becomes the same as the number of data bytes in a single FIFO buffer plane assigned to the selected pipe.</li> <li>(2) A number of data bytes less than the size of a single FIFO buffer (including 0 bytes of data) assigned to the selected pipe being written to the FIFO buffer and then 1 being written to BVAL</li> <li>(3) After setting the DMA transfer end sampling enable bit (TENDE) to 1, a number of data bytes less than the size of a single FIFO buffer (including 0 bytes of data) assigned to the selected pipe being written to the FIFO buffer by DMA transfer and the DMA transfer end signal is received when the last byte is written.</li> </ol>

### 29.3.33 Pipe Buffer Setting Register (PIPEBUF)

PIPEBUF is a register that specifies the buffer size and buffer number for PIPE1 to PIPE9.

This register is initialized by a power-on reset.



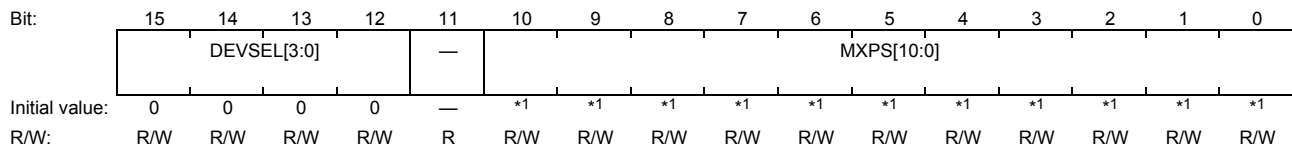
Bit	Bit Name	Initial Value	R/W	Description
15	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
14 to 10	BUFSIZE[4:0]	H'00	R/W	<p><b>Buffer Size</b> Specifies the size of the buffer for the pipe selected by the PIPESEL bits (selected pipe) in terms of blocks, where one block comprises 64 bytes. 00000 (H'00): 64 bytes 00001 (H'01): 128 bytes : : 11111 (H'1F): 2 Kbytes</p> <p>When the DBLB bit is set to 1, this module assigns two planes of the FIFO buffer size specified by the BUFSIZE bits to the selected pipe. Specifically, the following expression determines the FIFO buffer size assigned to the selected pipe by this module. (BUFSIZE + 1) × 64 × (DBLB + 1) [bytes] The valid value for these bits depends on the selected pipe.</p> <ul style="list-style-type: none"> <li>• PIPE1 to PIPE5 and PIPE9 to PIPE15: Any value from H'00 to H'1F is valid.</li> <li>• PIPE6 to PIPE8: H'00 should be set.</li> </ul> <p>When used with CNTMD = 1, set an integral multiple of the maximum packet size to the BUFSIZE bits. Modify these bits while CSSTS is 0, PID is NAK, and the pipe is not selected by the CURPIPE bits. Before modifying these bits after modifying the PID bits for the selected pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, the PBUSY bit does not have to be checked.</p>
9, 8	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	BUFNMB[7:0]	H'00	R/W	<p>Buffer Number</p> <p>The first block number in the FIFO buffer to be allocated for the selected pipe should be set in these bits. The FIFO buffer blocks allocated for the selected pipe by this module are determined as follows:            Block number: BUFNMB to block number of <math>(BUFNMB + (BUFSIZE + 1) \times (DBLB + 1) - 1)</math></p> <p>These bits should be set to a value from H'04 to H'7F.            BUFNMB = H'00 is used exclusively for DCP.            BUFNMB = H'04 is used exclusively for PIPE6.                When PIPE6 is not used, H'04 can be used for other pipes.                When PIPE6 is selected, writing to these bits is invalid and H'04 is automatically assigned by this module.            BUFNMB = H'05 is used exclusively for PIPE7.                When PIPE7 is not used, H'05 can be used for other pipes.                When PIPE7 is selected, writing to these bits is invalid and H'05 is automatically assigned by this module.            BUFNMB = H'06 is used exclusively for PIPE8.                When PIPE8 is not used, H'06 can be used for other pipes.                When PIPE8 is selected, writing to these bits is invalid and H'06 is automatically assigned by this module.            BUFNMB = H'07 is used exclusively for PIPE9.                When PIPE9 is not used, H'07 can be used for other pipes.                When PIPE9 is selected, writing to these bits is valid and H'07 is assigned as the initial value by this module.</p> <p>Modify these bits while CSSTS is 0, PID is NAK, and the pipe is not selected by the CURPIPE bits.            Before modifying these bits after modifying the PID bits for the selected pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, the PBUSY bit does not have to be checked.</p>

### 29.3.34 Pipe Maximum Packet Size Register (PIPEMAXP)

PIPEMAXP is a register that specifies the maximum packet size for PIPE1 to PIPE15.

This register is initialized by a power-on reset.



Bit	Bit Name	Initial Value	R/W	Description
15 to 12	DEVSEL[3:0]	0000	R/W	<p>Device Select</p> <p>When the host controller mode is selected, these bits specify the USB address of the communication target peripheral device.</p> <p>0000: Address 0000 0001: Address 0001 0010: Address 0010 : : 1010: Address 1010 Others: Setting prohibited</p> <p>These bits should be set after setting the address to the DEVADDn (n = 0 to A) register corresponding to the value to be set in these bits. For example, before setting DEVSEL to 0010, the DEVADD2 address should be set first.</p> <p>Before modifying these bits after modifying the PID bits for the selected pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, the PBUSY bit does not have to be checked.</p> <p>When the function controller mode is selected, these bits should be set to B'0000.</p>
11	—	Undefined	R	<p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p>
10 to 0	MXPS[10:0]	*1	R/W	<p>Maximum Packet Size</p> <p>Specifies the maximum data payload (maximum packet size) for the selected pipe. The valid value for these bits depends on the pipe as follows.</p> <p>PIPE1, PIPE2: 1 byte (H'001) to 1,024 bytes (H'400) PIPE3 to PIPE5: 8 bytes (H'008), 16 bytes (H'010), 32 bytes (H'020), 64 bytes (H'040), and 512 bytes (H'200) (Bits [2:0] are not available.) PIPE6 to PIPE8: 1 byte (H'001) to 64 bytes (H'040) PIPE9 (when in host controller mode): 1 byte (H'001) to 64 bytes (H'040) PIPE9 (when in function controller mode): 8 bytes (H'008), 16 bytes (H'010), 32 bytes (H'020), 64 bytes (H'040), 512 bytes (H'200) (Bits [2:0] are not available.) PIPE10 to PIPE15: 8 bytes (H'008), 16 bytes (H'010), 32 bytes (H'020), 64 bytes (H'040), 512 bytes (H'200) (Bits [2:0] are not available.)</p> <p>These bits should be set to the appropriate value for each transfer type based on the USB Specification.</p> <p>For split transactions using the isochronous pipe, these bits should be set to 188 bytes or less.</p> <p>Modify these bits while CSSTS is 0, PID is NAK, and the pipe is not selected by the CURPIPE bits.</p> <p>Before modifying these bits after modifying the PID bits for the selected pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, the PBUSY bit does not have to be checked.</p> <p>While MXPS is 0, do not write to the FIFO buffer or set PID to BUF.</p>

Note 1. The initial value of MXPS is H'000 when no pipe is selected with the PIPESEL bits in PIPESEL and H'040 when a pipe is selected with the PIPESEL bits in PIPESEL.

### 29.3.35 Pipe Timing Control Register (PIPEPERI)

PIPEPERI is a register that selects whether the buffer is flushed or not when an interval error occurred during isochronous IN transfer, and sets the interval error detection interval for PIPE1 to PIPE15.

This register is initialized by a power-on reset.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	IFIS	—	—	—	—	—	—	—	—	—	IITV[2:0]		
Initial value:	—	—	—	0	—	—	—	—	—	—	—	—	—	0	0	0
R/W:	R	R	R	R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
12	IFIS	0	R/W	<p><b>Isochronous IN Buffer Flush</b> Specifies whether to flush the buffer when the pipe selected by the PIPESEL bits is used for isochronous IN transfers.</p> <p>0: The buffer is not flushed. 1: The buffer is flushed.</p> <p>When the function controller mode is selected and the selected pipe is for isochronous IN transfers, this module automatically clears the FIFO buffer when this module fails to receive the IN token from the USB host within the interval set by the IITV bits in terms of (<math>\mu</math>) frames.</p> <p>In double buffer mode (DBLB = 1), this module only clears the data in the plane used earlier.</p> <p>This module clears the FIFO buffer on receiving the SOF packet immediately after the (<math>\mu</math>) frame in which this module has expected to receive the IN token. Even if the SOF packet is corrupted, this module also clears the FIFO buffer at the right timing to receive the SOF packet by using the internal interpolation.</p> <p>When the host controller mode is selected, set this bit to 0. When the selected pipe is not for the isochronous transfer, set this bit to 0.</p>
11 to 3	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
2 to 0	IITV[2:0]	000	R/W	<p><b>Interval Error Detection Interval</b> Specifies the interval error detection timing for the selected pipe in terms of frames, which is expressed as n-th power of 2 (n is the value to be set). As described later, the detailed functions are different in host controller mode and in function controller mode.</p> <p>Modify these bits while CSSTS is 0, PID is NAK, and the pipe is not selected by the CURPIPE bits.</p> <p>Before modifying these bits after modifying the PID bits for the selected pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, the PBUSY bit does not have to be checked.</p> <p>Before modifying these bits after USB communication has been completed with these bits set to a certain value, set PID to NAK and then set ACLRM to 1 to initialize the interval timer.</p> <p>The IITV bits are invalid for PIPE3 to PIPE5 and PIPE10 to PIPE15; set these bits to 000 for these pipes.</p>

### 29.3.36 PIPEn Control Registers (PIPEnCTR) (n = 1 to F)

PIPEnCTR is a register that is used to confirm the buffer memory status for the corresponding pipe, change and confirm the data PID sequence bit, determine whether auto response mode is set, determine whether auto buffer clear mode is set, and set a response PID for PIPE1 to PIPE15. This register can be set regardless of the pipe selection in PIPESEL. These registers are initialized by a power-on reset. The PID[1:0] bits are initialized by a USB bus reset.

#### (1) PIPEnCTR (n = 1 to 5, 9, A to F)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BSTS	INB UFM	CSCLR	CSSTS	—	ATREP M	ACLRM	SQCLR	SQSET	SQ MON	PBUSY	—	—	—	PID[1:0]	
Initial value:	0	0	0	0	—	0	0	0	0	0	0	—	—	—	0	0
R/W:	R	R	R/W <sup>2</sup>	R	R	R/W	R/W	R/W <sup>1</sup>	R/W <sup>1</sup>	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	BSTS	0	R	<p><b>Buffer Status</b> Indicates whether or not the FIFO buffer allocated for the pertinent pipe can be accessed by the CPU. 0: Buffer access by the CPU is disabled. 1: Buffer access by the CPU is enabled. The meaning of this bit depends on the settings of the DIR, BFRE, and DCLRM bits as shown in Table 29.12.</p>
14	INBUFM	0	R	<p><b>Transmission Buffer Monitor</b> Indicates the pertinent FIFO buffer status when the pertinent pipe is in the transmitting direction. 0: There is no transmissible data in the buffer memory. 1: There is transmissible data in the buffer memory. When the pertinent pipe is in the transmitting direction (DIR = 1), this module sets this bit to 1 when writing data to at least one FIFO buffer plane is completed. This module sets this bit to 0 when this module completes transmitting the data from the FIFO buffer plane to which all the data has been written. In double buffer mode (DBLB = 1), this module sets this bit to 0 after this module has completed transmitting the data from both FIFO buffer planes but before it has completed writing data to a single FIFO buffer plane. This bit indicates the same value as the BSTS bit when the pertinent pipe is in the receiving direction (DIR = 0).</p>
13	CSCLR	0	R/W <sup>2</sup>	<p><b>C-SPLIT Status Clear Bit</b> When the host controller mode is selected, setting this bit to 1 allows this module to clear the CSSTS bit to 0. 0: Writing invalid 1: Clears the CSSTS bit to 0. For the transfer using the split transaction, to restart the next transfer with the S-SPLIT forcibly, set this bit to 1. However, for the normal split transaction, this module automatically clears the CSSTS bit to 0 upon completion of the C-SPLIT; therefore, clearing the CSSTS bit is not necessary. Controlling the CSSTS bit through this bit must be done while UACT is 0 and thus communication is halted or while no transfer is being performed with bus disconnection detected. Setting this bit to 1 while CSSTS is 0 has no effect. When the function controller mode is selected, be sure to write 0 to this bit.</p>



Bit	Bit Name	Initial Value	R/W	Description
12	CSSTS	0	R	<p>CSSTS Status Bit</p> <p>Indicates the C-SPLIT status of the split transaction when the host controller mode is selected.</p> <p>0: START-SPLIT (S-SPLIT) transaction being processed or the transfer not using the split transaction in progress</p> <p>1: C-SPLIT transaction being processed</p> <p>This module sets this bit to 1 upon start of the C-SPLIT and clears this bit to 0 upon detection of C-SPLIT completion.</p> <p>Indicates the valid value only when the host controller mode is selected.</p>
11	—	Undefined	R	<p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p>
10	ATREPM	0	R/W	<p>Auto Response Mode</p> <p>Enables or disables auto response mode for the pertinent pipe.</p> <p>0: Auto response disabled</p> <p>1: Auto response enabled</p> <p>When the function controller mode is selected and the pertinent pipe is for bulk transfer, this bit can be set to 1.</p> <p>When this bit is set to 1, this module responds to the token from the USB host as described below.</p> <p>(1) When the pertinent pipe is for bulk IN transfer (TYPE = 01 and DIR = 1)</p> <p>When ATREPM = 1 and PID = BUF, this module transmits a zero-length packet in response to the IN token.</p> <p>This module updates (toggles) the sequence toggle bit (DATA-PID) each time this module receives the ACK from the USB host (in a single transaction, IN token is received, zero-length packet is transmitted, and then ACK is received.).</p> <p>In this case, this module does not generate the BRDY or BEMP interrupt.</p> <p>(2) When the pertinent pipe is for bulk OUT transfer (TYPE = 01 and DIR = 0)</p> <p>When ATREPM = 1 and PID = BUF, this module returns NAK in response to the OUT token (or PING token) and generates the NRDY interrupt.</p> <p>Modify this bit while CSSTS is 0 and PID is NAK. Before modifying this bit after modifying the PID bits for the corresponding pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, the PBUSY bit does not have to be checked.</p> <p>For USB communication in auto response mode, set this bit to 1 while the FIFO buffer is empty. Do not write to the FIFO buffer during USB communication in auto response mode.</p> <p>When the pertinent pipe is for isochronous transfer, be sure to set this bit to 0.</p> <p>When the host controller mode is selected, set this bit to 0.</p>
9	ACLRM	0	R/W	<p>Auto Buffer Clear Mode</p> <p>Enables or disables automatic buffer clear mode for the pertinent pipe.</p> <p>0: Disabled</p> <p>1: Enabled (all buffers are initialized)</p> <p>To delete the contents in the FIFO buffer assigned to the pertinent pipe completely, write 1 and then 0 to this bit continuously.</p> <p>Table 29.13 shows the contents cleared by writing 1 and 0 to this bit continuously and the cases in which clearing the contents is necessary.</p> <p>Modify this bit while CSSTS is 0, PID is NAK, and the pertinent pipe is not selected by the CURPIPE bits.</p> <p>Before modifying this bit after modifying the PID bits for the corresponding pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, the PBUSY bit does not have to be checked.</p>

Bit	Bit Name	Initial Value	R/W	Description
8	SQCLR	0	R/W <sup>*1</sup>	<p>Toggle Bit Clear</p> <p>This bit should be set to 1 to clear the expected value (to set DATA0 as the expected value) of the sequence toggle bit for the next transaction of the pertinent pipe.</p> <p>0: Writing invalid 1: Specifies DATA0.</p> <p>Setting this bit to 1 allows this module to set DATA0 as the expected value of the sequence toggle bit of the pertinent pipe. This bit always indicates 0.</p> <p>When the host controller mode is selected, setting this bit to 1 for the pipe for bulk OUT transfer, this module starts the next transfer of the pertinent pipe with the PING token.</p> <p>Set the SQCLR bit to 1 while CSSTS is 0 and PID is NAK.</p> <p>Before modifying this bit after modifying the PID bits for the corresponding pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, the PBUSY bit does not have to be checked.</p>
7	SQSET	0	R/W <sup>*1</sup>	<p>Toggle Bit Set</p> <p>This bit should be set to 1 to set DATA1 as the expected value of the sequence toggle bit for the next transaction of the pertinent pipe.</p> <p>0: Writing invalid 1: Specifies DATA1.</p> <p>Setting this bit to 1 allows this module to set DATA1 as the expected value of the sequence toggle bit of the pertinent pipe. This module always sets this bit to 0.</p> <p>Set the SQSET bit to 1 while CSSTS is 0 and PID is NAK.</p> <p>Before modifying this bit after modifying the PID bits for the corresponding pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, the PBUSY bit does not have to be checked.</p>
6	SQMON	0	R	<p>Toggle Bit Confirmation</p> <p>Indicates the expected value of the sequence toggle bit for the next transaction of the pertinent pipe.</p> <p>0: DATA0 1: DATA1</p> <p>When the pertinent pipe is not for the isochronous transfer, this bit is toggled upon normal completion of the transaction. However, this bit is not toggled when a DATA-PID disagreement occurs during the receiving transfer.</p>
5	PBUSY	0	R	<p>Pipe Busy</p> <p>This bit indicates whether or not the pertinent pipe is being currently used for the transaction.</p> <p>0: The pertinent pipe is not being currently used for the transaction. 1: The pertinent pipe is being currently used for the transaction.</p> <p>This module modifies this bit from 0 to 1 upon start of the USB transaction for the pertinent pipe, and modifies the bit from 1 to 0 upon completion of one transaction.</p> <p>Reading this bit after PID has been set to NAK allows checking that modification of the pipe settings is possible.</p>
4 to 2	—	Undefined	R	<p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p>

Bit	Bit Name	Initial Value	R/W	Description
1, 0	PID[1:0]	00	R/W	<p>Response PID</p> <p>Specifies the response type for the next transaction of the pertinent pipe.</p> <p>00: NAK response 01: BUF response (depending on the buffer state) 10: STALL response 11: STALL response</p> <p>The default setting of these bits is NAK. Modify the setting to BUF to use the pertinent pipe for USB transfer. Table 29.14 and Table 29.15 show the basic operation (operation when there are no errors in the transmitted and received packets) of this module depending on the PID bit setting. Some registers require these bits to be set to NAK before their settings can be changed by software. Set these bits to NAK by software when changing such kind of registers. To confirm which registers apply to this, reference the descriptions of bits. After modifying the setting of these bits from BUF to NAK during USB communication using the pertinent pipe, check that PBUSY is 0 to see if USB communication using the pertinent pipe has actually entered the NAK state. However, if the PID bits have been modified to NAK by this module, the PBUSY bit does not have to be checked.</p> <p>Even if the PID bits are modified to NAK after S-SPLIT of the split transaction has been issued for the pertinent pipe (while CSSTS indicates 1), this module continues the transaction until C-SPLIT completes.</p> <ul style="list-style-type: none"> <li>• This module modifies the setting of these bits as follows. <ul style="list-style-type: none"> <li>This module sets PID to NAK on recognizing the completion of the transfer when the pertinent pipe is in the receiving direction and the SHTNAK bit for the selected pipe has been set to 1.</li> </ul> </li> <li>• This module sets PID to STALL (11) on receiving the data packet with the payload exceeding the maximum packet size of the pertinent pipe.</li> <li>• This module sets PID to NAK on detecting a USB bus reset when the function controller mode is selected.</li> <li>• This module sets PID to NAK on detecting a reception error such as a CRC error three consecutive times when the host controller mode is selected.</li> <li>• This module sets PID to STALL (11) on receiving the STALL handshake when the host controller mode is selected.</li> </ul> <p>To specify each response type, set these bits as follows.</p> <ul style="list-style-type: none"> <li>• To make a transition from NAK (00) to STALL, set 10.</li> <li>• To make a transition from BUF (01) to STALL, set 11.</li> <li>• To make a transition from STALL (11) to NAK, set 10 and then 00.</li> <li>• To make a transition from STALL to BUF, set 00 (NAK) and then 01 (BUF).</li> </ul>

Note 1. Only 0 can be read and 1 can be written to.

Note 2. Only 1 can be written to.

**Table 29.12 Meaning of BSTS Bit**

DIR Bit	BFRE Bit	DCLRM Bit	Meaning of BSTS Bit
0	0	0	1: The received data can be read from the FIFO buffer. 0: The received data has been completely read from the FIFO buffer.
		1	Setting prohibited
	1	0	1: The received data can be read from the FIFO buffer. 0: The BCLR bit has been set to 1 after the received data has been completely read from the FIFO buffer.
		1	1: The received data can be read from the FIFO buffer. 0: The received data has been completely read from the FIFO buffer.
1	0	0	1: The transmit data can be written to the FIFO buffer. 0: The transmit data has been completely written to the FIFO buffer.
		1	Setting prohibited
	1	0	Setting prohibited
		1	Setting prohibited

**Table 29.13 Contents Cleared by This Module by Setting ACLRM = 1**

No.	Contents Cleared by ACLRM Bit Manipulation	Cases in which Clearing the Contents is Necessary
1	All the contents in the FIFO buffer assigned to the pertinent pipe (all the information in two FIFO buffer planes in double buffer mode)	
2	The interval count value when the pertinent pipe is for isochronous transfer	When the interval count value is to be reset
3	Values of the internal flags related to the BFRE bit	When the BFRE setting is modified
4	FIFO buffer toggle control	When the DBLB setting is modified
5	Values of the internal flags related to the transaction count	When the transaction count function is forcibly terminated

**Table 29.14 Operation of This Module Depending on PID Setting (when Host Controller Mode is Selected)**

PID	Transfer Type (TYPE Bits)	Transfer Direction (DIR Bit)	Operation of This Module
00 (NAK)	Operation does not depend on the setting.	Operation does not depend on the setting.	Does not issue tokens.
01 (BUF)	Bulk (TYPE = 01) or interrupt (TYPE = 10)	Operation does not depend on the setting.	Issues tokens while UACT is 1 and the FIFO buffer corresponding to the pertinent pipe is ready for transmission and reception. Does not issue tokens while UACT is 0 or the FIFO buffer corresponding to the pertinent pipe is not ready for transmission or reception.
	Isochronous (TYPE = 11)	Operation does not depend on the setting.	Issues tokens irrespective of the status of the FIFO buffer corresponding to the pertinent pipe when the UACT bit is set to 1. Does not issue tokens when the UACT bit is set to 0.
10 (STALL) or 11 (STALL)	Operation does not depend on the setting.	Operation does not depend on the setting.	Does not issue tokens.

**Table 29.15 Operation of This Module Depending on PID Setting (when Function Controller Mode is Selected)**

PID	Transfer Type (TYPE Bits)	Transfer Direction (DIR Bit)	Operation of This Module
00 (NAK)	Bulk (TYPE = 01) or interrupt (TYPE = 10)	Operation does not depend on the setting.	Returns NAK in response to the token from the USB host.
	Isochronous (TYPE = 11)	Operation does not depend on the setting.	Returns nothing in response to the token from the USB host.
01 (BUF)	Bulk (TYPE = 01)	Receiving direction (DIR = 0)	Receives data and returns ACK or NYET in response to the OUT token from the USB host if the FIFO buffer corresponding to the pertinent pipe is ready for reception. Returns NAK if not ready. Returns ACK in response to the PING token from the USB host if the FIFO buffer corresponding to the pertinent pipe is ready for reception. Returns NAK if not ready.
		Receiving direction (DIR = 0)	Receives data and returns ACK in response to the OUT token from the USB host if the FIFO buffer corresponding to the pertinent pipe is ready for reception. Returns NAK if not ready.
	Bulk (TYPE = 01) or interrupt (TYPE = 10)	Transmitting direction (DIR = 1)	Transmits data in response to the token from the USB host if the corresponding FIFO buffer is ready for transmission. Returns NAK if not ready.
	Isochronous (TYPE = 11)	Receiving direction (DIR = 0)	Receives data in response to the OUT token from the USB host if the FIFO buffer corresponding to the pertinent pipe is ready for reception. Discards data if not ready.
		Transmitting direction (DIR = 1)	Transmits data in response to the token from the USB host if the corresponding FIFO buffer is ready for transmission. Transmits the zero-length packet if not ready.
10 (STALL) or 11 (STALL)	Bulk (TYPE = 01) or interrupt (TYPE = 10)	Operation does not depend on the setting.	Returns STALL in response to the token from the USB host.
	Isochronous (TYPE = 11)	Operation does not depend on the setting.	Returns nothing in response to the token from the USB host.

## (2) PIPEnCTR (n = 6 to 8)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BSTS	—	CSCLR	CSSTS	—	—	ACLRM	SQCLR	SQSET	SQ MON	PBUSY	—	—	—	PID[1:0]	
Initial value:	0	—	0	0	—	—	0	0	0	0	0	—	—	—	0	0
R/W:	R	R	R/W*1	R/W	R	R	R/W	R/W*1	R/W*1	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	BSTS	0	R	<p>Buffer Status</p> <p>Indicates whether or not the FIFO buffer allocated for the pertinent pipe can be accessed by the CPU.</p> <p>0: Buffer access is disabled.</p> <p>1: Buffer access is enabled.</p> <p>The meaning of this bit depends on the settings of the DIR, BFRE, and DCLRM bits as shown in Table 29.12.</p>
14	—	Undefined	R	<p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p>
13	CSCLR	0	R/W*1	<p>C-SPLIT Status Clear Bit</p> <p>Setting this bit to 1 allows this module to clear the CSSTS bit of the pertinent pipe to 0.</p> <p>0: Writing invalid</p> <p>1: Clears the CSSTS bit to 0.</p> <p>For the transfer using the split transaction, to restart the next transfer with the S-SPLIT forcibly, set this bit to 1. However, for the normal split transaction, this module automatically clears the CSSTS bit to 0 upon completion of the C-SPLIT; therefore, clearing the CSSTS bit is not necessary.</p> <p>Controlling the CSSTS bit through this bit must be done while UACT is 0 thus communication is halted or while no transfer is being performed with bus disconnection detected.</p> <p>Setting this bit to 1 while CSSTS is 0 has no effect.</p> <p>When the function controller mode is selected, be sure to write 0 to this bit.</p>
12	CSSTS	0	R/W	<p>CSSTS Status Bit</p> <p>Indicates the C-SPLIT status of the split transaction when the host controller mode is selected.</p> <p>0: START-SPLIT (S-SPLIT) transaction being processed or the transfer not using the split transaction in progress</p> <p>1: C-SPLIT transaction being processed</p> <p>This module sets this bit to 1 upon start of the C-SPLIT and clears this bit to 0 upon detection of C-SPLIT completion.</p> <p>Indicates the valid value only when the host controller mode is selected.</p>
11, 10	—	Undefined	R	<p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p>
9	ACLRM	0	R/W	<p>Auto Buffer Clear Mode</p> <p>Enables or disables automatic buffer clear mode for the pertinent pipe.</p> <p>0: Disabled</p> <p>1: Enabled (all buffers are initialized)</p> <p>To delete the contents in the FIFO buffer assigned to the pertinent pipe completely, write 1 and then 0 to this bit continuously.</p> <p>Table 29.16 shows the contents cleared by writing 1 and 0 to this bit continuously and the cases in which clearing the contents is necessary.</p> <p>Modify this bit while CSSTS is 0, PID is NAK, and the pertinent pipe is not selected by the CURPIPE bits.</p> <p>Before modifying this bit after modifying the PID bits for the corresponding pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, the PBUSY bit does not have to be checked.</p>

Bit	Bit Name	Initial Value	R/W	Description
8	SQCLR	0	R/W*1	<p>Toggle Bit Clear</p> <p>This bit should be set to 1 to clear the expected value (to set DATA0 as the expected value) of the sequence toggle bit for the next transaction of the pertinent pipe.</p> <p>0: Writing invalid 1: Specifies DATA0.</p> <p>Setting this bit to 1 allows this module to set DATA0 as the expected value of the sequence toggle bit of the pertinent pipe. This bit always indicates 0.</p> <p>When the host controller mode is selected, setting this bit to 1 for the pipe for bulk OUT transfer, this module starts the next transfer of the pertinent pipe with the PING token.</p> <p>Set the SQCLR bit to 1 while CSSTS is 0 and PID is NAK.</p> <p>Before modifying this bit after modifying the PID bits for the corresponding pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, the PBUSY bit does not have to be checked.</p>
7	SQSET	0	R/W*1	<p>Toggle Bit Set</p> <p>This bit should be set to 1 to set DATA1 as the expected value of the sequence toggle bit for the next transaction of the pertinent pipe.</p> <p>0: Writing invalid 1: Specifies DATA1.</p> <p>Setting this bit to 1 allows this module to set DATA1 as the expected value of the sequence toggle bit of the pertinent pipe. This module always sets this bit to 0.</p> <p>Set the SQSET bit to 1 while CSSTS is 0 and PID is NAK.</p> <p>Before modifying this bit after modifying the PID bits for the corresponding pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, the PBUSY bit does not have to be checked.</p>
6	SQMON	0	R	<p>Toggle Bit Confirmation</p> <p>Indicates the expected value of the sequence toggle bit for the next transaction of the pertinent pipe.</p> <p>0: DATA0 1: DATA1</p> <p>When the pertinent pipe is not for the isochronous transfer, this bit is toggled upon normal completion of the transaction. However, this bit is not toggled when a DATA-PID disagreement occurs during the receiving transfer.</p>
5	PBUSY	0	R	<p>Pipe Busy</p> <p>This bit indicates whether or not the pertinent pipe is being currently used for the transaction.</p> <p>0: The pertinent pipe is not being currently used for the transaction. 1: The pertinent pipe is being currently used for the transaction.</p> <p>This module modifies this bit from 0 to 1 upon start of the USB transaction for the pertinent pipe, and modifies the bit from 1 to 0 upon completion of one transaction.</p> <p>Reading this bit after PID has been set to NAK allows checking that modification of the pipe settings is possible.</p>
4 to 2	—	Undefined	R	<p>Reserved</p> <p>The read value is undefined. The write value should always be 0.</p>

Bit	Bit Name	Initial Value	R/W	Description
1, 0	PID[1:0]	00	R/W	<p>Response PID</p> <p>Specifies the response type for the next transaction of the pertinent pipe.</p> <p>00: NAK response 01: BUF response (depending on the buffer state) 10: STALL response 11: STALL response</p> <p>The default setting of these bits is NAK. Modify the setting to BUF to use the pertinent pipe for USB transfer. Table 29.14 and Table 29.15 show the basic operation (operation when there are no errors in the transmitted and received packets) of this module depending on the PID bit setting. Some registers require these bits to be set to NAK before their settings can be changed by software. Set these bits to NAK by software when changing such kind of registers. To confirm which registers apply to this, reference the descriptions of bits. After modifying the setting of these bits from BUF to NAK during USB communication using the pertinent pipe, check that PBUSY is 0 to see if USB communication using the pertinent pipe has actually entered the NAK state. However, if the PID bits have been modified to NAK by this module, the PBUSY bit does not have to be checked.</p> <p>Even if the PID bits are modified to NAK after S-SPLIT of the split transaction has been issued for the pertinent pipe (while CSSTS indicates 1), this module continues the transaction until C-SPLIT completes.</p> <p>This module modifies the setting of these bits as follows.</p> <ul style="list-style-type: none"> <li>• This module sets PID to NAK on recognizing the completion of the transfer when the pertinent pipe is in the receiving direction and the SHTNAK bit for the selected pipe has been set to 1.</li> <li>• This module sets PID to STALL (11) on receiving the data packet with the payload exceeding the maximum packet size of the pertinent pipe.</li> <li>• This module sets PID to NAK on detecting a USB bus reset when the function controller mode is selected.</li> <li>• This module sets PID to NAK on detecting a reception error such as a CRC error three consecutive times when the host controller mode is selected.</li> <li>• This module sets PID to STALL (11) on receiving the STALL handshake when the host controller mode is selected.</li> </ul> <p>To specify each response type, set these bits as follows.</p> <ul style="list-style-type: none"> <li>• To make a transition from NAK (00) to STALL, set 10.</li> <li>• To make a transition from BUF (01) to STALL, set 11.</li> <li>• To make a transition from STALL (11) to NAK, set 10 and then 00.</li> <li>• To make a transition from STALL to BUF, set 00 (NAK) and then 01 (BUF).</li> </ul>

Note 1. Only 0 can be read and 1 can be written to.

**Table 29.16 Contents Cleared by This Module by Setting ACLRM = 1**

No.	Contents Cleared by ACLRM Bit Manipulation	Cases in which Clearing the Contents is Necessary
1	All the contents in the FIFO buffer assigned to the selected pipe	
2	When the host controller mode is selected, the interval count value when the selected pipe is for interrupt transfer	When the interval count value is to be reset
3	Values of the internal flags related to the BFRE bit	When the BFRE setting is modified
4	Values of the internal flags related to the transaction count	When the transaction count function is forcibly terminated



### 29.3.37 PIPE<sub>n</sub> Transaction Counter Enable Registers (PIPE<sub>n</sub>TRE) (n = 1 to 5, 9, and A to F)

PIPE<sub>n</sub>TRE is a register that enables or disables the transaction counter corresponding to PIPE1 to PIPE5, and clears the transaction counter.

These registers are initialized by a power-on reset.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TRENB	TRCLR	—	—	—	—	—	—	—	—
Initial value:	—	—	—	—	—	—	0	0	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R/W	R/W*	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
9	TRENB	0	R/W	Transaction Counter Enable Enables or disables the transaction counter. 0: The transaction counter is disabled. 1: The transaction counter is enabled. For the pipe in the receiving direction, setting this bit to 1 after setting the total number of the packets to be received in the TRNCNT bits allows this module to control hardware as described below on having received the number of packets equal to the set value in the TRNCNT bits. <ul style="list-style-type: none"> <li>In continuous transmission/reception mode (CNTMD = 1), this module switches the FIFO buffer to the CPU side even if the FIFO buffer is not full on completion of reception.</li> <li>While SHTNAK is 1, this module modifies the PID bits to NAK for the corresponding pipe on having received the number of packets equal to the set value in the TRNCNT bits.</li> <li>While BFRE is 1, this module asserts the BRDY interrupt on having received the number of packets equal to the set value in the TRNCNT bits and then reading out the last received data.</li> </ul> For the pipe in the transmitting direction, set this bit to 0. When the transaction counter is not used, set this bit to 0. When the transaction counter is used, set the TRNCNT bits before setting this bit to 1. Set this bit to 1 before receiving the first packet to be counted by the transaction counter.
8	TRCLR	0	R/W*	Transaction Counter Clear When this bit is set to 1, this module clears the current counter value of the transaction counter corresponding to the pertinent pipe and then sets this bit to 0. 0: Invalid 1: The current counter value is cleared.
7 to 0	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.

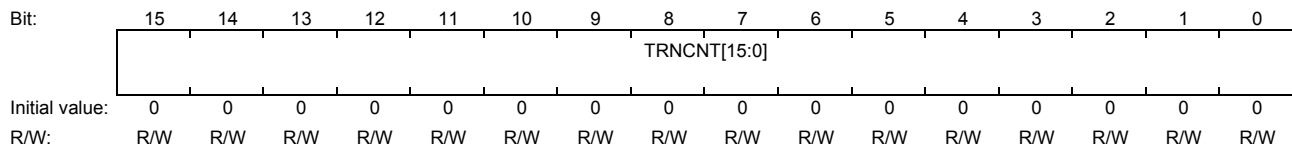
Note 1. Modify each bit in this register while CSSTS is 0 and PID is NAK. Before modifying each bit after modifying the PID bits for the corresponding pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, the PBUSY bit does not have to be checked.

\* Only 0 can be read and 1 can be written to.

### 29.3.38 PIPE<sub>n</sub> Transaction Counter Registers (PIPE<sub>n</sub>TRN) (n = 1 to 5, 9, and A to F)

PIPE<sub>n</sub>TRN is a transaction counter corresponding to PIPE1 to PIPE5, 9, and A to F.

These registers are initialized by a power-on reset, but retain the set value by a USB bus reset.



Bit	Bit Name	Initial Value	R/W	Description
15 to 0	TRNCNT[15:0]	H'0000	R/W	<p>Transaction Counter</p> <p>When written to: Specifies the total number of packets to be received by the pertinent pipe (number of transactions).</p> <p>When read from: Indicates the specified number of transactions if TRENB is 0. Indicates the number of currently counted transaction if TRENB is 1.</p> <p>This module increments the value of these bits by one when all of the following conditions are satisfied on receiving the packet.</p> <ul style="list-style-type: none"> <li>• TRENB is 1.</li> <li>• (TRNCNT setting <math>\neq</math> current counter value + 1) on receiving the packet.</li> <li>• The payload of the received packet agrees with the set value in the MXPS bits.</li> </ul> <p>This module clears the value of these bits to 0 when any of the following conditions are satisfied.</p> <ul style="list-style-type: none"> <li>• All the following conditions are satisfied. <ul style="list-style-type: none"> <li>TRENB is 1.</li> <li>(TRNCNT setting = current counter value + 1) on receiving the packet.</li> <li>The payload of the received packet agrees with the set value in the MXPS bits.</li> </ul> </li> <li>• All the following conditions are satisfied. <ul style="list-style-type: none"> <li>TRENB is 1.</li> <li>This module has received a short packet.</li> </ul> </li> <li>• The following condition is satisfied. <ul style="list-style-type: none"> <li>TRCLR is 1.</li> </ul> </li> </ul> <p>For the pipe in the transmitting direction, set these bits to 0. When the transaction counter is not used, set these bits to 0. Modify these bits while CSSTS is 0, PID is NAK, and TRENB is 0. Before modifying these bits after modifying the PID bits for the corresponding pipe from BUF to NAK, check that CSSTS and PBUSY are 0. However, if the PID bits have been modified to NAK by this module, the PBUSY bit does not have to be checked. To modify the value of these bits, set TRCLR to 1 before setting TRENB to 1.</p>

### 29.3.39 Device Address n Configuration Registers (DEVADDn) (n = 0 to 9, and A)

DEVADDn is a register that specifies the USB address and port number of the hub to which the communication target peripheral device is connected and also specifies the USB transfer speed of the communication target peripheral device. These registers are initialized by a power-on reset.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	UPPHUB[3:0]				HUBPORT[2:0]			USBSPD[1:0]		—	—	—	—	—	—
Initial value:	—	0	0	0	0	0	0	0	0	0	—	—	—	—	—	—
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
14 to 11	UPPHUB[3:0]	0000	R/W	Address of Hub to which Communication Target is Connected Specifies the USB address of the hub to which the communication target peripheral device is connected. 0000: The peripheral device is directly connected to the port of this module. 0001 to 1010: USB address of the hub 1011 to 1111: Setting prohibited
10 to 8	HUBPORT[2:0]	000	R/W	Port Number of Hub to which Communication Target is Connected Specifies the port number of the hub to which the communication target peripheral device is connected. 000: The peripheral device is directly connected to the port of this module. 001 to 111: Port number of the hub
7, 6	USBSPD[1:0]	00	R/W	Transfer Speed of the Communication Target Device Specifies the USB transfer speed of the communication target peripheral device. 00: DEVADDn register is not used. 01: Low speed 10: Full speed 11: High speed
5 to 0	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.

Note 1. When the host controller mode is selected, the bits in this register should be set before starting communication using each pipe.

(1) When the host controller mode is selected, this module refers to the settings of the UPPHUB bits and HUBPORT bits to generate packets for split transactions.

(2) When the host controller mode is selected, this module refers to the setting of the USBSPD bits to generate packets.

Note 2. The bits in this register should be modified while no valid pipes are using the settings of this register. Valid pipes refer to the ones satisfying both of conditions (1) and (2) below.

(1) This register is selected by the DEVSEL bits as the communication target.

(2) The PID bits are set to BUF for the pertinent pipe or the pertinent pipe is the DCP with SUREQ being 1.

Note 3. When the function controller mode is selected, set all the bits in this register to 0.

### 29.3.40 UTMI Suspend Mode Register (SUSPMODE)

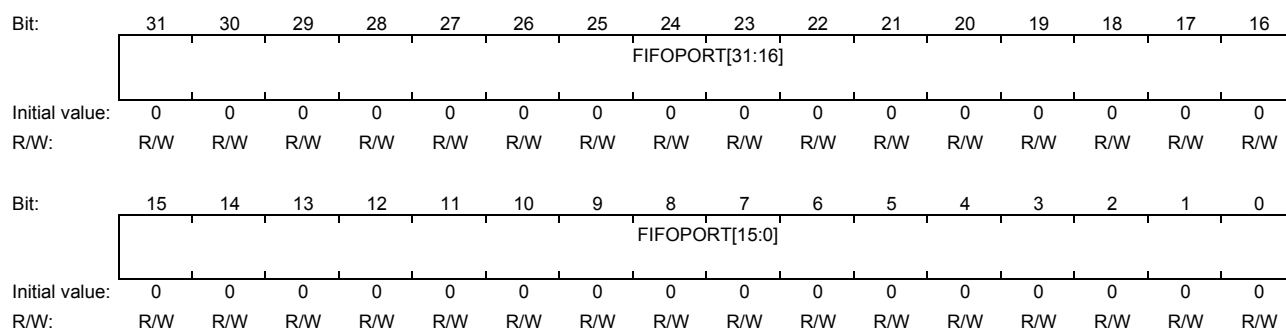
SUSPMODE is a register that specifies the SuspendM signal to be sent to the UTMI.

This register is initialized by a power-on reset.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	SUSP M	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	—	0	—	0	—	—	—	0	—	—	—	—	0	—	0	0
R/W:	R	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
14	SUSPM	0	R/W	SuspendM Control Enables or disables the clock to be supplied to this module. 0: The clock supplied to this module is disabled. 1: The clock supplied to this module is enabled. This module controls clock output by using the SuspendM signal. The clock to the LINK is stopped, when the SuspendM signal is low. Writing to the registers of this module is impossible when the SUSPM bit is set to 0 (the clock to this module is stopped). Reading the registers is possible. Note that writing to the following registers is possible even when the SUSPM bit is set to 0. - SYSCFG0 - BUSWAIT - INTENB1* - SUSPMODE Note: * Writing to bit 0 in INTENB1 is only possible when the SUSPM bit is set to 0.  When setting this bit to 1, wait for at least 1 ms after setting the UPLLE bit to 1. Set this bit to 0 when this module is to enter software standby or USB standby mode.
13	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
12	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
11 to 9	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
8	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
7 to 4	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	—	Undefined	R	Reserved The read value is undefined. The write value should always be 0.
1, 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

29.3.41 FIFO Continuous Transfer Port Registers (D0FIFOBn, D1FIFOBn) (n = 0 to 7)



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	FIFOPORT [31:0]	H'0000 0000	R/W	FIFO Port Accessing these bits allows reading the received data from the FIFO buffer or writing the transmit data to the FIFO buffer. When the DFACC bits are set to 01 (16-byte continuous access mode) or 10 (32-byte continuous access mode), use the Dx FIFO continuous transfer port registers to access the Dx FIFO buffer.

## 29.4 Operation

### 29.4.1 System Control and Oscillation Control

This section describes the register operations that are necessary to the initial settings of this module, and the registers necessary for power consumption control.

#### (1) Resets

Table 29.17 lists the types of resets for this module. For the initialized states of the registers following the reset operations, see section 29.3, Register Descriptions.

**Table 29.17 Types of Reset**

Name	Operation
Power-on reset	Low level input from the $\overline{\text{RES}}$ pin
USB bus reset	Automatically detected by this module from the D+ and D- lines when the function controller mode is selected

#### (2) Controller Mode Selection

This module can select the host controller mode or function controller mode using the DCFM bit in SYSCFG0. Changing the DCFM bit should be done in the initial settings immediately after a power-on reset or in the D+ pull-up disabled (DPRPU = 0) and D+ /D- pull-down disabled (DRPD = 0) state.

#### (3) USB Data Bus Resistor Control

This module controls switching between a pull-up resistor for the D+ signal and a pull-down resistor for the D+ and D- signals for the Renesas USB 2.0 PHY port. The DPRPU and DRPD bits of the SYSCFG0 register are used to make the pull-up and pull-down resistor settings.

When the function controller mode is selected, set the DPRPU bit of the SYSCFG0 register to 1, which pulls up the D+ signal, after connection to a USB host is recognized.

When disconnection from the USB host is recognized, set the DPRPU and DCFM bits according to the procedure below.

1. Set the DPRPU bit to 0.
2. Wait for at least 1  $\mu\text{s}$  (1000 ns).
3. Set the DCFM bit to 1.
4. Wait for at least 200 ns.
5. Set the DCFM bit to 0.

This module incorporates the terminating resistor for the D+ and D- signals during high-speed operation and the output resistor for the signals during full-speed operation. This module automatically switches the resistor after connection with the host controller or peripheral device upon the detection of a reset handshake, suspend, and resume event.

When the function controller mode is selected and the DPRPU bit in SYSCFG0 is cleared to 0 during communication with the host controller, the pull-up resistor (or the terminating resistor) of the USB data line is disabled, making it possible to notify the USB host of the device disconnection.

#### (4) Selecting the Input Clock

This module can select the signal on USB\_X1 or EXTAL as an input clock. The UCKSEL bit of the SYSCFG0 register for channel 0 is used to select the input clock. The UCKSEL bit should be set while supply of the clock signal to the USB module is stopped (SUSPM = 0 for channel 0 and channel 1).

### (5) Setting the Clock Supply for the USB Module

Set the UCKSEL bit of the SYSCFG0 register to the target input clock and then set clock supply by following the appropriate procedure below.

Example 1: When the clock supply is enabled by the initial setting immediately after a power-on reset,

1. set the UPLLE bit to 1,
2. wait for 1 ms, and
3. set the SUSPM bit to 1.

Example 2: When the clock supply is stopped in the suspended state,

1. set the SUSPM bit to 0 and
2. set the UPLLE bit to 0.

Example 3: When the clock supply is enabled after recovery from the suspended state,

1. set the UPLLE bit to 1,
2. wait for 1 ms, and
3. set the SUSPM bit to 1.

Note: • During high-speed operation when the function controller mode is selected, set the SUSPM bit to 1 within 2.5 ms of this module resuming operation from the suspended state due to a USB reset.

## 29.4.2 Interrupt Functions

### (1) Overview of Interrupt Functions

Table 29.18 lists the interrupt generation conditions for this module.

**Table 29.18 Interrupt Generation Conditions**

Bit	Interrupt Name	Cause of Interrupt	Mode That Generates the Interrupt	Related Status
VBINT	VBUS interrupt	<ul style="list-style-type: none"> <li>When a change in the state of the VBUS input pin has been detected (low to high or high to low)</li> </ul>	Host, function	VBSTS
RESM	Resume interrupt	<ul style="list-style-type: none"> <li>When a change in the state of the USB bus has been detected in the suspended state (J-state to K-state or J-state to SE0)</li> </ul>	Function	—
SOFR	Frame number update interrupt	When the host controller mode is selected: <ul style="list-style-type: none"> <li>When an SOF packet with a different frame number has been transmitted</li> </ul> When the function controller mode is selected: <ul style="list-style-type: none"> <li>If SOFRM = 0, reception of an SOF packet with a different frame number</li> <li>If SOFRM = 1, reception of an SOF packet with the microframe number 0 was not possible because it was corrupted.</li> </ul>	Host, function	—
DVST	Device state transition interrupt	<ul style="list-style-type: none"> <li>When a device state transition is detected</li> <li>A USB bus reset detected</li> <li>The suspended state detected</li> <li>SET_ADDRESS request received</li> <li>SET_CONFIGURATION request received</li> </ul>	Function	DVSTQ
CTRT	Control transfer stage transition interrupt	<ul style="list-style-type: none"> <li>When a stage transition is detected in control transfer</li> <li>Setup stage completed</li> <li>Control write transfer status stage transition</li> <li>Control read transfer status stage transition</li> <li>Control transfer completed</li> <li>A control transfer sequence error occurred</li> </ul>	Function	CTSQ
BEMP	Buffer empty interrupt	<ul style="list-style-type: none"> <li>When transmission of all of the data in the buffer memory has been completed and the buffer has become empty</li> <li>When an excessive maximum packet size error has been detected</li> </ul>	Host, function	PIPEBEMP
NRDY	Buffer not ready interrupt	When the host controller mode is selected: <ul style="list-style-type: none"> <li>When STALL is received from the peripheral side for the issued token</li> <li>When a response cannot be received correctly from the peripheral side for the issued token (No response is returned three consecutive times or a packet reception error occurred three consecutive times.)</li> <li>When an overrun/underrun occurred during isochronous transfer</li> </ul> When the function controller mode is selected: <ul style="list-style-type: none"> <li>Reception of a token while PID = BUF and the buffer memory is not ready for transmission or reception</li> <li>When a CRC error or a bit stuffing error occurred during data reception in isochronous transfer</li> <li>When an interval error occurred during data reception in isochronous transfer</li> </ul>	Host, function	PIPENRDY
BRDY	Buffer ready interrupt	<ul style="list-style-type: none"> <li>When the buffer is ready (reading or writing is enabled)</li> </ul>	Host, function	PIPEBRDY
BCHG	Bus change interrupt	<ul style="list-style-type: none"> <li>When a change of USB bus state is detected</li> </ul>	Host	—
DTCH	Device disconnection detection	<ul style="list-style-type: none"> <li>When disconnection of a peripheral device connected to this LSI's USB port is detected.</li> </ul>	Host	—
ATTCH	Device connection detection	<ul style="list-style-type: none"> <li>When J-state or K-state is detected on the USB port for 2.5 <math>\mu</math>s. Used for checking whether a peripheral device is connected.</li> </ul>	Host	—



**Table 29.18 Interrupt Generation Conditions**

<b>Bit</b>	<b>Interrupt Name</b>	<b>Cause of Interrupt</b>	<b>Mode That Generates the Interrupt</b>	<b>Related Status</b>
EOFERR	EOF error detection	<ul style="list-style-type: none"><li>When EOF error of a peripheral device is detected</li></ul>	Host	—
SACK	Normal setup operation	<ul style="list-style-type: none"><li>When the normal response (ACK) for the setup transaction is received</li></ul>	Host	—
SIGN	Setup error	<ul style="list-style-type: none"><li>When a setup transaction error (no response or ACK packet corruption) is detected three consecutive times.</li></ul>	Host	—

Figure 29.1 shows a diagram relating to interrupts of this module.

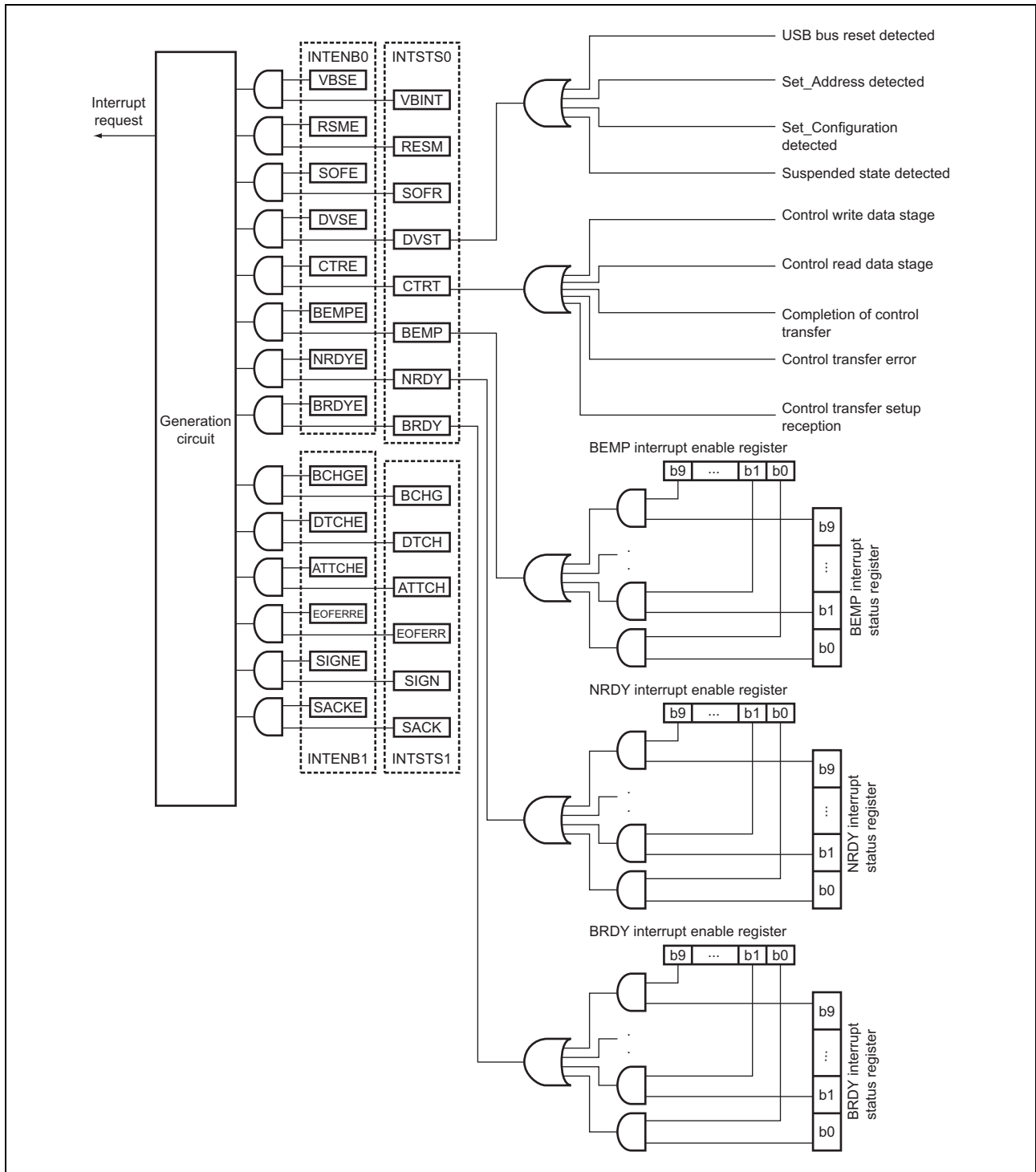


Figure 29.1 Items Relating to Interrupts

## (2) BRDY Interrupt

The BRDY interrupt is generated when either of the host controller mode or function controller mode is selected. When each pipe satisfies the following conditions, this module sets 1 to a corresponding bit in BRDYSTS. Under this condition, if the PIPEBRDYE bit in BRDYENB that corresponds to the pipe is set to 1 and the BRDYE bit in INTENB0 is set to 1, this module sets the BRDY bit in INTSTS0 to 1, allowing the BRDY interrupt to be generated.

The conditions for generating and clearing the BRDY interrupt depend on the settings of the BRDYM bit and BFRE bit for the pertinent pipe as described below.

### (a) When BRDYM Bit is 0 and BFRE Bit is 0

With these settings, the BRDY interrupt indicates that the FIFO port is accessible.

On any of the following conditions, this module generates the internal BRDY interrupt request trigger and sets 1 to the PIPEBRDY bit corresponding to the pertinent pipe.

#### 1. For the pipe in the transmitting direction

- When the DIR bit is changed from 0 to 1.

- When packet transmission is completed using the pertinent pipe when write-access from the CPU to the FIFO buffer for the pertinent pipe is disabled (when the BSTS bit is read as 0).

In continuous transmission/reception mode, the request trigger is generated on completion of transmitting data of one plane of the FIFO buffer.

- When one FIFO buffer is empty on completion of writing data to the other FIFO buffer in double buffer mode.

The request trigger is not generated until completion of writing data to the currently-written FIFO buffer plane even if transmission to the other FIFO buffer is completed.

- When the hardware flushes the buffer of the pipe for isochronous transfers.

- When 1 is written to the ACLRM bit, which causes the FIFO buffer to make transition from the write-disabled to write-enabled state.

The request trigger is not generated for the DCP (that is, during data transmission for control transfers).

#### 2. For the pipe in the receiving direction

- When packet reception is completed successfully thus enabling the FIFO buffer to be read when read-access from the CPU to the FIFO buffer for the pertinent pipe is disabled (when the BSTS bit is read as 0).

The request trigger is not generated for the transaction in which DATA-PID disagreement occurs.

In continuous transmission/reception mode, the request trigger is not generated when the data is of the specified maximum packet size and the buffer has available space.

When a short packet is received, the request trigger is generated even if the FIFO buffer has available space.

- When the transaction counter is used, the request trigger is generated on receiving the specified number of packets.

In this case, the request trigger is generated even if the FIFO buffer has available space.

When one FIFO buffer is read-enabled on completion of reading data from the other FIFO buffer in double buffer mode.

The request trigger is not generated until completion of reading data from the currently-read FIFO buffer plane even if reception by the other FIFO buffer is completed.

When the function controller mode is selected, the BRDY interrupt is not generated in the status stage of control transfers.

The PIPEBRDY interrupt status of the pertinent pipe can be cleared to 0 by writing 0 to the corresponding PIPEBRDY interrupt status bit in the BRDYSTS register. In this case, 1s should be written to the PIPEBRDY interrupt status bits for the other pipes.

Be sure to clear the BRDY status before accessing the FIFO buffer.

**(b) When BRDYM Bit is 0 and BFRE Bit is 1**

With these settings, this module generates the BRDY interrupt on completion of reading all the data for a single transfer using the pipe in the receiving direction, and sets 1 to the PIPEBRDY bit corresponding to the pertinent pipe.

On any of the following conditions, this module determines that the last data for a single transfer has been received.

- When a short packet including a zero-length packet is received.
- When the transaction counter register (TRNCNT bits) is used and the number of packets specified by the TRNCNT bits is completely received.

When the pertinent data is completely read out after any of the above determination conditions has been satisfied, this module determines that all the data for a single transfer has been completely read out.

When a zero-length packet is received when the FIFO buffer is empty, this module determines that all the data for a single transfer has been completely read out upon the FRDY and DTLN bits of the FIFO port control register being set to 1 and 0, respectively. In this case, to start the next transfer, write 1 to the BCLR bit in the corresponding FIFOCTR register.

With these settings, this module does not detect the BRDY interrupt for the pipe in the transmitting direction.

The PIPEBRDY interrupt status of the pertinent pipe can be cleared to 0 by writing 0 to the corresponding PIPEBRDY interrupt status bit. In this case, 1s should be written to the PIPEBRDY interrupt status bits for the other pipes.

In this mode, the BFRE bit setting should not be modified until all the data for a single transfer has been processed.

When it is necessary to modify the BFRE bit before completion of processing, all the FIFO buffers for the pertinent pipe should be cleared using the ACLRM bit.

**(c) When the BRDYM bit is 1 and the BFRE bit is 0**

With these settings, the PIPEBRDY values are linked to the BSTS bit settings for each pipe. In other words, the BRDY interrupt status bits (PIPEBRDY) are set to 1 or 0 by this module depending on the FIFO buffer status.

**1. For the pipe in the transmitting direction**

The BRDY interrupt status bits are set to 1 when the FIFO buffer is write-enabled and are set to 0 when write-disabled.

However, the BRDY interrupt is not generated if the DCP in the transmitting direction is write-enabled.

**2. For the pipe in the receiving direction**

The BRDY interrupt status bits are set to 1 when the FIFO buffer is read-enabled and are set to 0 when all the data have been read (read-disabled).

When a zero-length packet is received when the FIFO buffer is empty, the pertinent bit is set to 1 and the BRDY interrupt continues to be effective until BCLR = 1 is written.

With this setting, the PIPEBRDY bit cannot be cleared to 0. When BRDYM is set to 1, all of the BFRE bits (for all pipes) should be cleared to 0.

### (3) NRDY Interrupt

On generating the internal NRDY interrupt request for the pipe whose PID bits are set to BUF, this module sets the corresponding PIPENRDY bit in NRDYSTS to 1. If the corresponding bit in NRDYENB is set to 1, this module sets the NRDY bit in INTSTS0 to 1, allowing the NRDY interrupt to be generated.

The following describes the conditions on which this module generates the internal NRDY interrupt request for a given pipe.

However, the internal NRDY interrupt request is not generated during setup transaction execution when the host controller mode is selected. During setup transactions when the host controller mode is selected, the SACK or SIGN interrupt is detected.

The internal NRDY interrupt request is not generated during status stage execution of the control transfer when the function controller mode is selected.

#### (a) If Host Controller Mode is Selected when Connection is Used in which No Split Transactions Occur

##### 1. For the pipe in the transmitting direction

On any of the following conditions, this module detects the NRDY interrupt.

- For the pipe for isochronous transfers, when the time to issue an OUT token comes in a state in which there is no data to be transmitted in the FIFO buffer.

In this case, this module transmits a zero-length packet following the OUT token, setting the corresponding PIPENRDY bit and the OVRN bit to 1.

- During communications other than setup transactions using the pipe for the transfers other than isochronous transfers, when any combination of the following two cases occur three consecutive times: 1) no response is returned from the peripheral device (when timeout is detected before detection of the handshake packet from the peripheral device) and 2) an error is detected in the packet from the peripheral device.

In this case, this module sets the corresponding PIPENRDY bit to 1 and modifies the setting of the PID bits of the corresponding pipe to NAK.

- During communications other than setup transactions, when the STALL handshake is received from the peripheral device (including the STALL handshake in response to PING in addition to the STALL handshake in response to OUT).

In this case, this module sets the corresponding PIPENRDY bit to 1 and modifies the setting of the PID bits of the corresponding pipe to STALL (11).

##### 2. For the pipe in the receiving direction

- For the pipe for isochronous transfers, when the time to issue an IN token comes in a state in which there is no space available in the FIFO buffer.

In this case, this module discards the received data for the IN token, setting the PIPENRDY bit of the corresponding pipe and the OVRN bit to 1.

When a packet error is detected in the received data for the IN token, this module also sets the CRCE bit to 1.

- For the pipe for the transfers other than isochronous transfers, when any combination of the following two cases occur three consecutive times: 1) no response is returned from the peripheral device for the IN token issued by this module (when timeout is detected before detection of the DATA packet from the peripheral device) and 2) an error is detected in the packet from the peripheral device.

In this case, this module sets the corresponding PIPENRDY bit to 1 and modifies the setting of the PID bits of the corresponding pipe to NAK.

- For the pipe for isochronous transfers, when no response is returned from the peripheral device for the IN token (when timeout is detected before detection of the DATA packet from the peripheral device) or an error is detected in the packet from the peripheral device.

In this case, this module sets the corresponding PIPENRDY bit to 1. (The setting of the PID bits of the corresponding pipe is not modified.)

- For the pipe for isochronous transfers, when a CRC error or a bit stuffing error is detected in the received data packet.

In this case, this module sets the corresponding PIPENRDY bit and CRCE bit to 1.

- When the STALL handshake is received.

In this case, this module sets the corresponding PIPENRDY bit to 1 and modifies the setting of the PID bits of the corresponding pipe to STALL.

## (b) If Host Controller Mode is Selected when Connection is Used in which Split Transactions Occur

### 1. For the pipe in the transmitting direction

- For the pipe for isochronous transfers, when the time to issue an OUT token comes in a state in which there is no data to be transmitted in the FIFO buffer.

In this case, this module transmits a zero-length packet following the OUT token, setting the corresponding PIPENRDY bit and the OVRN bit to 1 at the issuance of the start-split transaction (S-SPLIT).

- For the pipe for the transfers other than isochronous transfers, when any combination of the following two cases occur three consecutive times: 1) no response is returned from the HUB for the S-SPLIT or complete-split transaction (C-SPLIT) (when timeout is detected before detection of the handshake packet from the HUB) and 2) an error is detected in the packet from the HUB.

In this case, this module sets the PIPENRDY bit of the corresponding pipe to 1 and modifies the setting of the PID bits of the corresponding pipe to NAK.

If the NRDY interrupt is detected when the C-SPLIT is issued, this module clears the CSSTS bit to 0.

- When the STALL handshake is received in response to the C-SPLIT.

In this case, this module sets the corresponding PIPENRDY bit to 1, modifies the setting of the PID bits of the corresponding pipe to STALL (11) and clears the CSSTS bit to 0.

This interrupt is not detected for setup transactions.

- For the pipe for interrupt transfers when the NYET is received in response to the C-SPLIT and the microframe number = 4.

In this case, this module sets the corresponding PIPENRDY bit to 1 and clears the CSSTS bit to 0 (does not modify the setting of the PID bits for the corresponding pipe).

### 2. For the pipe in the receiving direction

- For the pipe for isochronous transfers, when the time to issue an IN token comes in a state in which there is no space available in the FIFO buffer.

In this case, this module discards the received data for the IN token, setting the corresponding PIPENRDY bit and the OVRN bit to 1 at the issuance of the S-SPLIT.

- For the pipes for bulk transfers, or the transfers other than setup transactions with the DCP, when any combination of the following two cases occur three consecutive times: 1) no response is returned from the HUB for the IN token issued by this module at the issuance of S-SPLIT or C-SPLIT (when timeout is detected before detection of the DATA packet from the HUB) and 2) an error is detected in the packet from the HUB.

In this case, this module sets the corresponding PIPENRDY bit to 1 and modifies the setting of the PID bits of the corresponding pipe to NAK. When the condition is generated during the C-SPLIT transaction, this module clears the CSSTS bit to 0.

- During the C-SPLIT transaction for the pipe for isochronous transfers or interrupt transfers, when any combination of the following two cases occur three consecutive times: 1) no response is returned from the HUB for the IN token issued by this module (when timeout is detected before detection of the DATA packet from the HUB) and 2) an error is detected in the packet from the HUB.

On generating this condition for the pipe for interrupt transfers, this module sets the corresponding PIPENRDY bit to 1, modifies the setting of the PID bits of the corresponding pipe to NAK and clears the CSSTS bit to 0.

On generating this condition for the pipe for isochronous transfers, this module sets the corresponding PIPENRDY

bit to 1 and CRCE bit to 1, and clears the CSSTS bit to 0 (does not modify the setting of the PID bits for the corresponding pipe).

- During the C-SPLIT transaction, when the STALL handshake is received for the pipe for the transfers other than isochronous transfers.

In this case, this module sets the corresponding PIPENRDY bit to 1, modifies the setting of the PID bits of the corresponding pipe to STALL (11) and clears the CSSTS bit to 0.

- During the C-SPLIT transaction, when the NYET handshake is received for the pipe for the isochronous transfers or interrupt transfers and the microframe number = 4.

In this case, this module sets the corresponding PIPENRDY bit for the pipe to 1, sets the CRCE bit to 1, and clears the CSSTS bit to 0 (does not modify the setting of the PID bits for the corresponding pipe).

### (c) When Function Controller Mode is Selected

#### 1. For the pipe in the transmitting direction

- On receiving an IN token when the PID bits are set to 01 (BUF) for the pertinent pipe and there is no data to be transmitted in the FIFO buffer.

In this case, this module generates an NRDY interrupt request at the reception of the IN token, setting the PIPENRDY bit to 1. For the pipe for the isochronous transfers in which an interrupt is generated, this module transmits a zero-length packet, setting the OVRN bit to 1.

#### 2. For the pipe in the receiving direction

- On receiving an OUT token when the PID bits are set to 01 (BUF) for the pertinent pipe and there is no space available in the FIFO buffer.

For the pipe for the isochronous transfers in which an interrupt is generated, this module generates an NRDY interrupt request at the reception of the OUT token, setting the PIPENRDY bit to 1 and OVRN bit to 1.

For the pipe for the transfers other than isochronous transfers in which an interrupt is generated, this module generates an NRDY interrupt request when a NAK handshake is transferred after the data following the OUT token was received, setting the PIPENRDY bit to 1.

However, during re-transmission (due to DATA-PID disagreement), the NRDY interrupt request is not generated. In addition, if an error occurs in the DATA packet, the NRDY interrupt request is not generated.

- On receiving a PING token when the PID bits are set to 01 (BUF) for the pertinent pipe and there is no space available in the FIFO buffer.

In this case, this module generates an NRDY interrupt request at the reception of the PING token, setting the PIPENRDY bit to 1.

- For the pipe for isochronous transfers, when the PID bits are set to 01 (BUF) for the pertinent pipe and a token is not received normally within an interval frame.

In this case, this module generates an NRDY interrupt request at the reception of an SOF, and sets the PIPENRDY bit to 1.

#### (4) BEMP Interrupt

On detecting the BEMP interrupt for the pipe whose PID bits are set to BUF, this module sets the corresponding PIPEBEMP bit in BEMPSTS to 1. If the corresponding bit in BEMPENB is set to 1, this module sets the BEMP bit in INTSTS0 to 1, allowing the USB interrupt to be generated.

The following describes the conditions on which this module generates the internal BEMP interrupt request.

1. For the pipe in the transmitting direction, when the FIFO buffer of the corresponding pipe is empty on completion of transmission (including zero-length packet transmission). In single buffer mode, the internal BEMP interrupt request is generated simultaneously with the BRDY interrupt for the pipe other than DCP. However, the internal BEMP interrupt request is not generated on any of the following conditions.

- When writing data to the FIFO buffer on the CPU side is started on completion of transmitting data of one plane in double buffer mode.
- When the buffer is cleared (emptied) by setting the ACLRM or BCLR bit to 1.
- When IN transfer (zero-length packet transmission) is performed during the control transfer status stage in function controller mode.

2. For the pipe in the receiving direction

When the successfully-received data packet size exceeds the specified maximum packet size. In this case, this module generates the BEMP interrupt request, setting the corresponding PIPEBEMP bit to 1, and discards the received data and modifies the setting of the PID bits of the corresponding pipe to STALL (11).

Here, this module returns no response when used as the host controller, and returns STALL response when used as the function controller.

However, the internal BEMP interrupt request is not generated on any of the following conditions.

- When a CRC error or bit stuffing error is detected in the received data.
- When a setup transaction is being performed.

Writing 0 to the PIPEBEMP bit clears the status; writing 1 to the PIPEBEMP bit has no effect.



(5) Device State Transition Interrupt (Function Controller Mode)

Figure 29.2 shows a diagram of how this module handles the device state transitions. This module monitors device states and generates device state transition interrupts. However, recovery from the suspended state (resume signal detection) is detected by means of the resume interrupt. The device state transition interrupts can be enabled or disabled individually by using INTENB0. The device state after a transition can be confirmed by using the DVSQ bits in INTSTS0.

When making a transition to the default state, the device state transition interrupt is generated after the reset handshake protocol has been completed.

Device state can be monitored only when the function controller mode is selected. Also, the device state transition interrupts can be generated only when the function controller mode is selected.

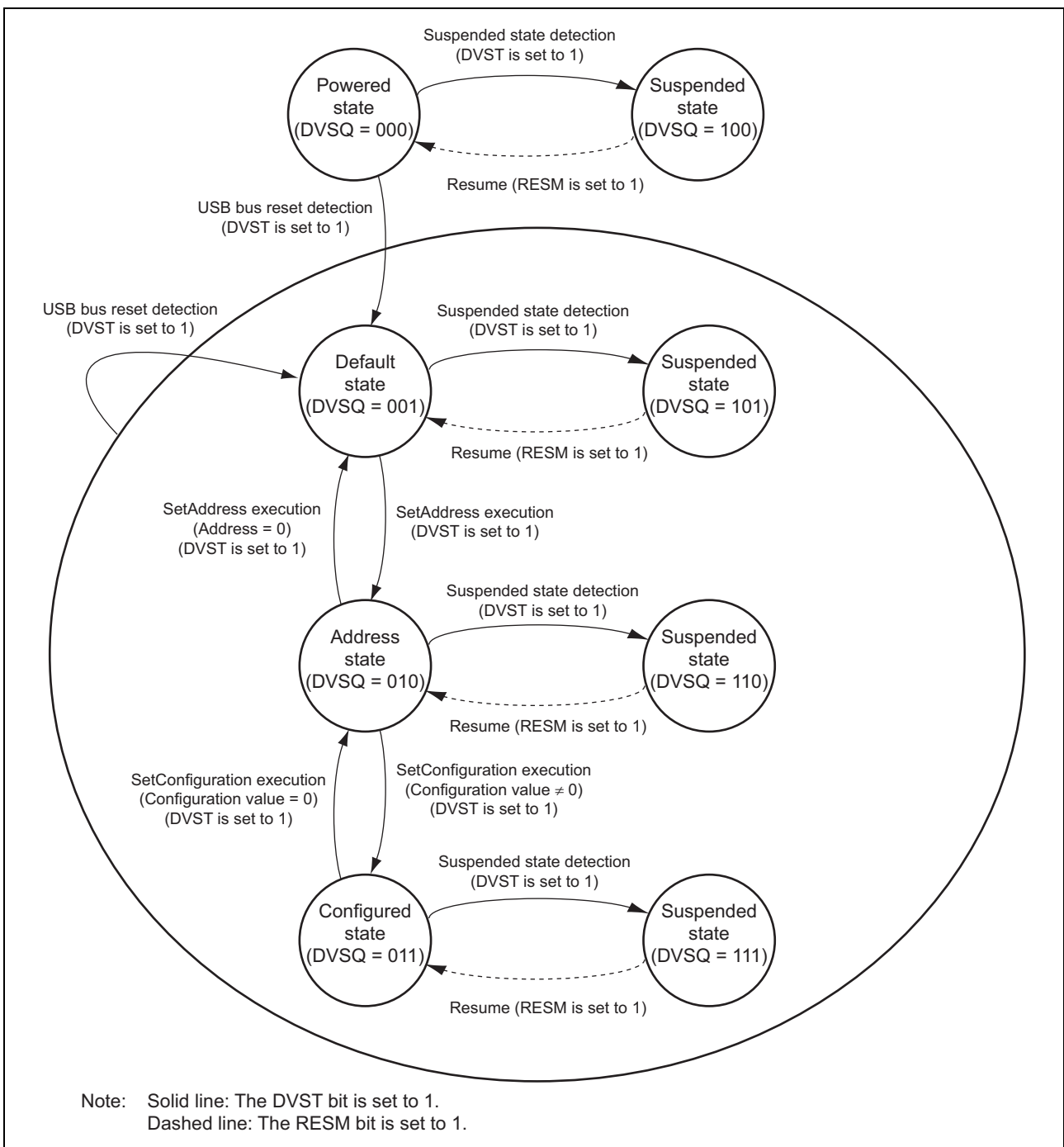


Figure 29.2 Device State Transitions

## (6) Control Transfer Stage Transition Interrupt (Function Controller Mode)

Figure 29.3 shows a diagram of how this module handles the control transfer stage transition. This module monitors the control transfer sequence and generates control transfer stage transition interrupts. Control transfer stage transition interrupts can be enabled or disabled individually using INTENB0. The control transfer stage after a transition can be confirmed using the CTSQ bits in INTSTS0.

The control transfer stage transition interrupts are generated only when the function controller mode is selected.

The control transfer sequence errors are described below. If an error occurs, the PID bits in DCPCTR are set to B'1x (STALL).

1. During control read transfers
  - At the IN token of the data stage, an OUT or PING token is received when there have been no data transfers at all.
  - An IN token is received at the status stage
  - A packet is received at the status stage for which the data packet is DATAPID = DATA0
2. During control write transfers
  - At the OUT token of the data stage, an IN token is received when there have been no ACK response at all
  - A packet is received at the data stage for which the first data packet is DATAPID = DATA0
  - At the status stage, an OUT or PING token is received
3. During control write no-data transfers
  - At the status stage, an OUT or PING token is received

At the control write transfer data stage, if the number of receive data exceeds the wLength value of the USB request, it cannot be recognized as a control transfer sequence error. At the control read transfer status stage, packets other than zero-length packets are received by an ACK response and the transfer ends normally.

When a CTRT interrupt occurs in response to a sequence error (SERR = 1), the CTSQ = 110 value is retained until CTRT = 0 is written from the system (the interrupt status is cleared). Therefore, while CTSQ = 110 is being retained, the CTRT interrupt that indicates completion of the setup stage will not be generated even if a further USB request is received (this module retains the indication of completion of the setup stage, and after the interrupt status flag has been cleared, a CTRT interrupt is generated).

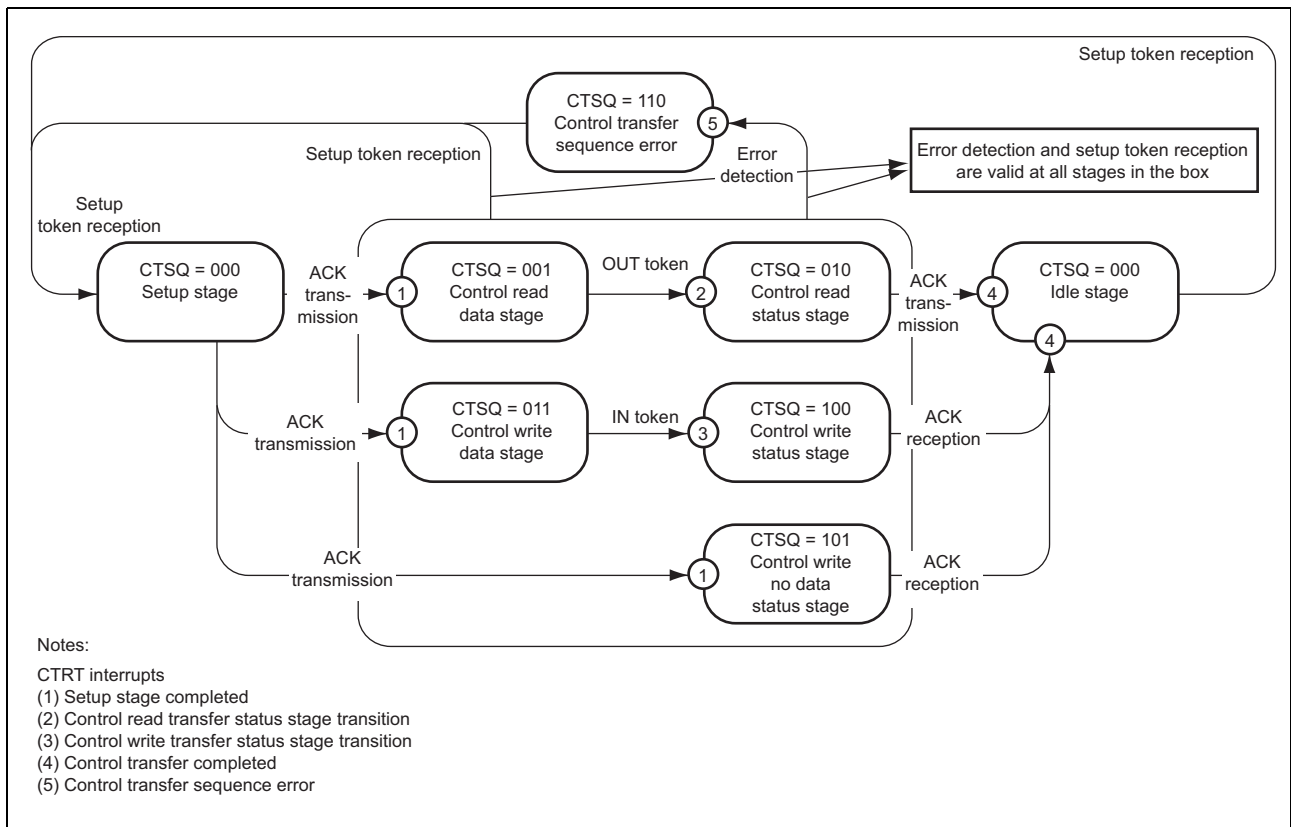


Figure 29.3 Control Transfer Stage Transitions

### 29.4.3 Pipe Control

Table 29.19 lists the pipe setting items of this module. With USB data transfer, data transmission has to be carried out using the logic pipe called the endpoint. This module has 16 pipes that are used for data transfer. Settings should be entered for each of the pipes in conjunction with the specifications of the system.

**Table 29.19 Pipe Setting Items**

Register Name	Bit Name	Setting Contents	Remarks
DCPCFG PIPECFG	TYPE	Specifies the transfer type	PIPE1 to PIPE15: Can be set
	BFRE	Selects the BRDY interrupt mode	PIPE1 to PIPE5, PIPE11 to PIPE15: Can be set
	DBLB	Selects a double buffer	PIPE1 to PIPE5, PIPE11 to PIPE15: Can be set PIPE9 and PIPE10: Can be set (only when bulk transfer has been selected).
	CNTMD	Selects continuous transfer or non-continuous transfer	DCP: Can be set. PIPE1 and PIPE2, PIPE9 and PIPE10: Can be set (only when bulk transfer has been selected). PIPE3 to PIPE5, PIPE11 to PIPE15: Can be set
	DIR	Selects transfer direction	IN or OUT can be set
	EPNUM	Endpoint number	PIPE1 to PIPE15: Can be set A value other than 0000 should be set when the pipe is used.
	SHTNAK	Selects disabled state for pipe when transfer ends	DCP: Can be set. PIPE1 and PIPE2, PIPE9 and PIPE10: Can be set (only when bulk transfer has been selected) PIPE3 to PIPE5, PIPE11 to PIPE15: Can be set
PIPEBUF	BUFSIZE	Buffer memory size	DCP: Cannot be set (fixed at 256 bytes) PIPE1 to PIPE5, PIPE9 to PIPE15: Can be set (a maximum of 2 Kbytes can be specified) PIPE6 to PIPE8: Cannot be set (fixed at 64 bytes)
	BUFNMB	Buffer memory number	DCP: Cannot be set (areas fixed at H'0 to H'3) PIPE1 to PIPE5, PIPE9 to PIPE15: Can be set (can be specified in areas H'7 to H'7F) PIPE6 to PIPE8: Cannot be set (areas fixed at H'4 to H'6)
DCPMAXP PIPEMAXP	DEVSEL	Selects a device	Referenced only when the host controller mode is selected.
	MXPS	Maximum packet size	Compliant with the USB standard.
PIPEPERI	IFIS	Buffer flush	PIPE1 and PIPE2: Can be set (only when isochronous transfer has been selected) PIPE3 to PIPE15: Cannot be set
	IITV	Interval counter	PIPE1 and PIPE2: Can be set (only when isochronous transfer has been selected) PIPE3 to PIPE5: Cannot be set PIPE6 to PIPE9: Can be set (only when the host controller mode has been selected) PIPE10 to PIPE15: Can be set

**Table 29.19 Pipe Setting Items**

Register Name	Bit Name	Setting Contents	Remarks
DCPCTR PIPEnCTR	BSTS	Buffer status	For the DCP, receive buffer status and transmit buffer status are switched with the ISEL bit.
	INBUFM	IN buffer monitor	Mounted for only PIPE3 to PIPE5 and PIPE9 to PIPE15.
	SUREQ	Setup request	Can be set only for the DCP. Can be controlled only when the host controller mode has been selected.
	SUREQCLR	SUREQ clear	Can be set only for the DCP. Can be controlled only when the host controller mode has been selected.
	CSCLR	CSSTS clear	Can be controlled only when the host controller mode has been selected.
	CSSTS	SPLIT status indication	Can be referenced only when the host controller mode has been selected.
	ATREPM	Auto response mode	PIPE1 to PIPE5, PIPE9 to PIPE15: Can be set Can be set only when the function controller mode has been selected.
	ACLRM	Auto buffer clear	PIPE1 to PIPE15: Can be set
	SQCLR	Sequence clear	Clears the data toggle bit
	SQSET	Sequence set	Sets the data toggle bit
	SQMON	Sequence monitor	Monitors the data toggle bit
	PBUSY	Pipe busy confirmation	
	PID	Response PID	
PIPEnTRE	TRENB	Transaction counter enable	PIPE1 to PIPE5, PIPE9 to PIPE15: Can be set
	TRCLR	Current transaction counter clear	PIPE1 to PIPE5, PIPE9 to PIPE15: Can be set
PIPEnTRN	TRNCNT	Transaction counter	PIPE1 to PIPE5, PIPE9 to PIPE15: Can be set

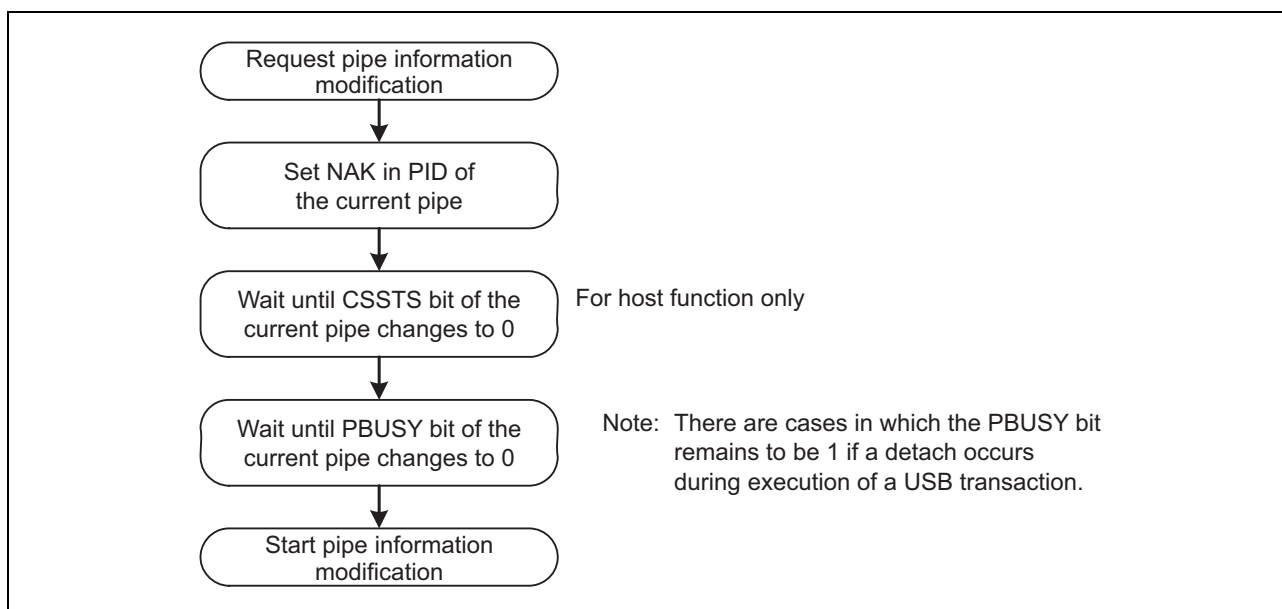
### (1) Pipe Control Register Switching Procedures

The following bits in the pipe control registers are only modifiable when USB communication is disabled (PID = NAK): Figure 29.4 shows the procedure for modifying the pipe control registers from the USB communication enabled (PID = BUF) state.

[Bits that should not be set in the USB communication enabled (PID = BUF) state]

- All bits in DCPCFG and DCPMAXP
- The SQCLR and SQSET bits in DCPCTR
- All bits in PIPECFG, PIPEBUF, PIPEMAXP, and PIPEPERI
- The ATREPM, ACLRM, SQCLR, and SQSET bits in PIPExCTR
- All bits in PIPExTRE and PIPExTRN
- All bits in DEVADDx

In addition to the settings described for the CSCLR bit and all bits in DEVADDx, the settings for each bit described in section 29.3, Register Descriptions must also be complied with.



**Figure 29.4 Procedure for Modifying Pipe Information from USB Communication Enabled (PID = BUF) State**

The following bits in the pipe control registers are only modifiable when the pertinent pipe has not been specified by the CURPIPE bits in CFIFOSEL, D0FIFOSEL, and D1FIFOSEL.

[Bits that should not be set when the pertinent pipe is specified by the CURPIPE bits in FIFO port select registers]

- All bits in DCPCFG and DCPMAXP
- All bits in PIPECFG, PIPEBUF, PIPEMAXP, and PIPEPERI
- The ACLRM bit in PIPExCTR

In order to modify pipe information, the CURPIPE bits should be set to the pipes other than the pipe to be modified. For the DCP, the buffer should be cleared using BCLR after the pipe information is modified.

## (2) Maximum Packet Size Setting

The MXPS bits in DCPMAXP and PIPEMAXP are used to specify the maximum packet size for each pipe. DCP, PIPE1 to PIPE5, and PIPE11 to PIPE15 can be set to any of the maximum packet sizes defined by the USB Specification. For PIPE6 to PIPE10, 64 bytes are the upper limit of the maximum packet size. The maximum packet size should be set before beginning the transfer (PID = BUF).

- DCP: 64 should be set when using high-speed operation.
- DCP: Select and set 8, 16, 32, or 64 when using full-speed operation.
- PIPE1 to PIPE5: 512 should be set when using high-speed bulk transfer.
- PIPE1 to PIPE5: Select and set 8, 16, 32, or 64 when using full-speed bulk transfer.
- PIPE1 and PIPE2: Set a value between 1 and 1024 when using high-speed isochronous transfer.
- PIPE1 and PIPE2: Set a value between 1 and 1023 when using full-speed isochronous transfer.  
For details, see [section 29.4.9, Isochronous Transfers \(PIPE1 and PIPE2\)](#).
- PIPE6 to PIPE8: Set a value between 1 and 64.
- PIPE9: 64 should be set when using interrupt transfer (only when the host controller mode has been selected).  
512 should be set when using high-speed bulk transfer (only when the function controller mode has been selected).  
Select and set 8, 16, 32, or 64 when using full-speed bulk transfer (only when the function controller mode has been selected).
- PIPE10 to PIPE15: 512 should be set when using high-speed bulk transfer (only when the function controller mode has been selected).
- PIPE10 to PIPE15: Select and set 8, 16, 32, or 64 when using full-speed bulk transfer (only when the function controller mode has been selected).

The high bandwidth transfers used with interrupt transfers and isochronous transfers are not supported.

### (3) Response PID

The PID bits in DCPCTR and PIPEnCTR are used to set the response PID for each pipe.

The following shows this module operation with various response PID settings:

1. Response PID settings when the host controller mode is selected

The response PID is used to specify the execution of transactions.

- NAK setting: Using pipes is disabled. No transaction is executed.

- BUF setting: Transactions are executed based on the status of the buffer memory.

For OUT direction: If there are transmit data in the buffer memory, an OUT token is issued.

For IN direction: If there is an area to receive data in the buffer memory, an IN token is issued.

- STALL setting: Using pipes is disabled. No transaction is executed.

Note: • Setup transactions for the DCP are set with the SUREQ bit.

2. Response PID settings when the function controller mode is selected

The response PID is used to specify the response to transactions from the host.

- NAK setting: The NAK response is always returned in response to the generated transaction.

- BUF setting: Responses are made to transactions based on the status of the buffer memory.

- STALL setting: The STALL response is always returned in response to the generated transaction.

Note: • For setup transactions, an ACK response is always returned, regardless of the PID setting, and the USB request is stored in registers USBREQ, USBVAL, USBINDEX, and USBLENG.

This module may carry out writing to the PID bits, depending on the results of the transaction. Writing to the PID bits by this module is carried out in the following cases.

1. When the host controller mode has been selected and the response PID is set by this module

- NAK setting: In the following cases, PID = NAK is set and issuing of tokens is automatically stopped:

- For transfer that is not isochronous, any combination of the following two items occurring three consecutive times in response to transmitted tokens

1) no response being returned or 2) a reception error such as a CRC error or a bit stuffing error

- For isochronous transfer, a reception error such as a CRC error or a bit stuffing error occurring three consecutive times in response to transmitted tokens

- Reception of a short packet at the stage of control read transfer data when the setting of the SHTNAK bit in PIPECFG is 1

- If a short packet is received when the SHTNAK bit in PIPECFG has been set to 1 for bulk transfer.

- If counting by the transaction counter ends while the SHTNAK bit in PIPECFG is set to 1 during bulk transfer.

- BUF setting: There is no BUF writing by this module.

- STALL setting: In the following cases, PID = STALL is set and issuing of tokens is automatically stopped:

When STALL is received in response to the transmitted token.

When the size of the receive data packet exceeds the maximum packet size.

2. When the function controller mode has been selected and the response PID is set by this module

- NAK setting: When the setup token is received normally (DCP only).

If counting by the transaction counter ends or a short packet is received while the SHTNAK bit in PIPECFG is set to 1 during bulk transfer.

- BUF setting: There is no BUF writing by this module.

- STALL setting: When a maximum packet size exceeded error has been detected in the receive data packet.

When a control transfer sequence error has been detected.



#### (4) Data PID Sequence Bit

This module automatically toggles the sequence bit in the data PID when data is transferred normally in the control transfer data stage, bulk transfer and interrupt transfer. The sequence bit of the data PID that is to be transmitted for the next transaction can be confirmed with the SQMON bit in DCPCTR or PIPEnCTR. When data is transmitted, the sequence bit switches at the timing at which the ACK handshake is received. When data is received, the sequence bit switches at the timing at which the ACK handshake is transmitted. The SQCLR and SQSET bits in DCPCTR or PIPEnCTR can be used to change the data PID sequence bit.

When the function controller mode has been selected and control transfer is used, this module automatically sets the sequence bit when a stage transition is made. The data PID sequence bit becomes DATA1 when the setup stage ends and this module does not reference the sequence bit and responds with PID = DATA1 in the status stage. Therefore, settings are not required. However, when the host controller mode has been selected and control transfer is used, the sequence bit should be set at the stage transition.

For the ClearFeature request transmission or reception, the data PID sequence bit should be set, regardless of whether the host controller mode or function controller mode is selected.

With pipes for which isochronous transfer has been set, sequence bit operation cannot be carried out using the SQSET bit.

### 29.4.4 FIFO Buffer Memory

This section describes the operation of the FIFO buffers incorporated in this module. Unless specifically specified, the buffer operation is the same regardless of whether the host controller mode or function controller mode is selected.

#### (1) FIFO Buffer Memory Allocation

Figure 29.5 shows an example of a FIFO buffer memory map for this module. The FIFO buffer memory is an area shared by the CPU and this module. In the FIFO buffer memory status, there are times when the access right to the buffer memory is allocated to the user system (CPU side), and times when it is allocated to this module (SIE side). Independent FIFO buffer memory areas should be set for each pipe. Each memory area can be set using the first block number and the number of blocks (specified using the BUFNMB and BUFSIZE bits in PIPEBUF), where one block comprises 64 bytes.

When continuous transfer mode has been selected using the CNTMD bit in PIPECFG, the BUFSIZE bits should be set so that the buffer memory size should be an integral multiple of the maximum packet size. When double buffer mode has been selected using the DBLB bit in PIPECFG, two planes of the memory area specified using the BUFSIZE bits in PIPEBUF can be assigned to a single pipe.

Moreover, three FIFO ports are used for access to the FIFO buffer memory (reading and writing data). A pipe is assigned to the FIFO port by specifying the pipe number using the CURPIPE bits in CFIFOSEL/DnFIFOSEL.

The FIFO buffer status of each pipe can be confirmed using the BSTS bit in DCPCTR or the BSTS and INBUFM bits in PIPECTR. Also, the access right of the FIFO port can be confirmed using the FRDY bit in CFIFOCTR or DnFIFOCTR.

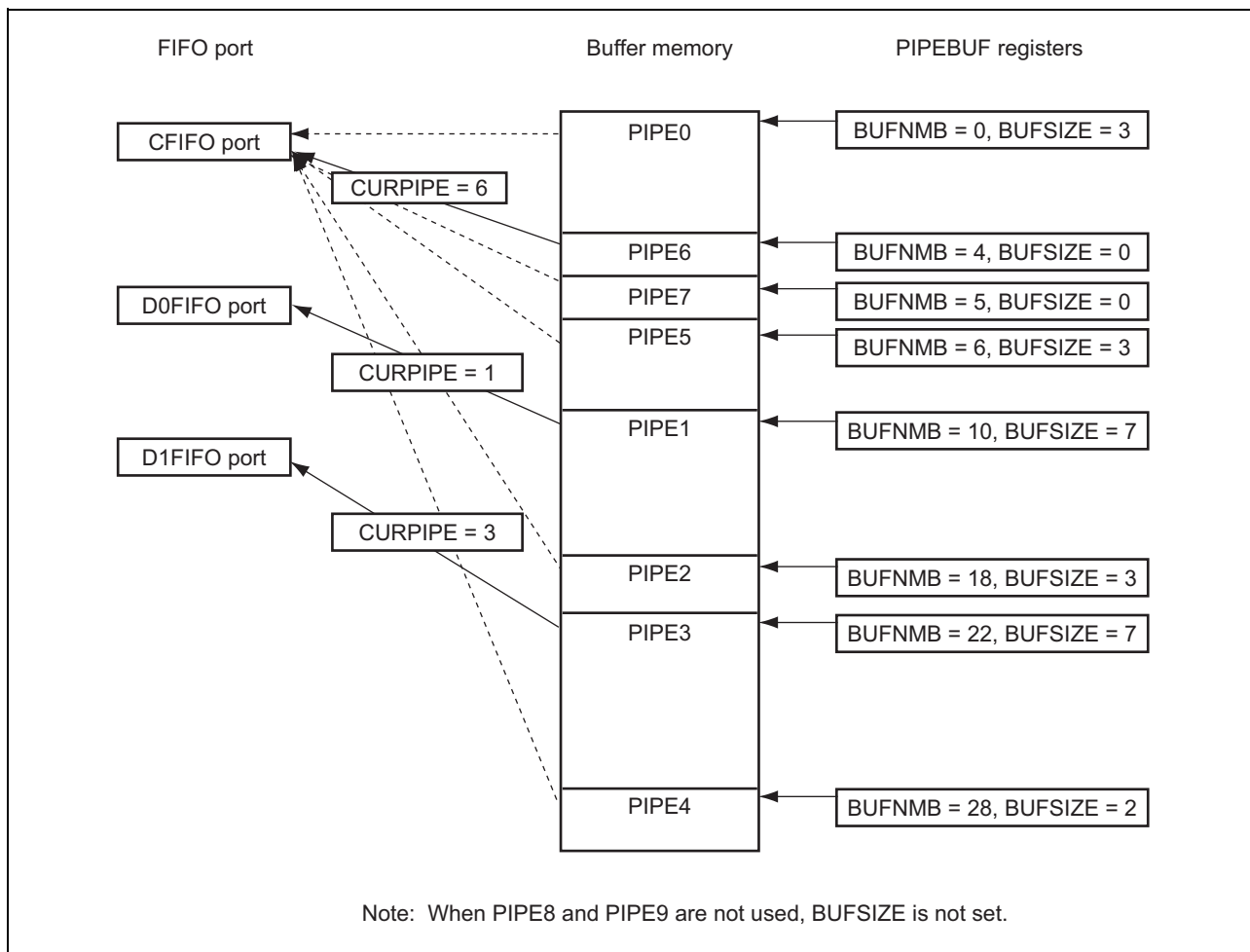


Figure 29.5 Example of a FIFO Buffer Memory Map

## (a) FIFO Buffer Clearing

Table 29.20 shows the clearing of the FIFO buffer memory by this module. The FIFO buffer memory can be cleared using the BCLR, DCLRM, and ACLRM bits.

**Table 29.20 List of FIFO Buffer Clearing Methods**

Bit Name	BCLR	DCLRM	ACLRM
Register	CFIFOCTR DnFIFOCTR	DnFIFOSEL	PIPEnCTR
Function	Clears the FIFO buffer memory on the CPU side	In this mode, after the data of the specified pipe has been read, the FIFO buffer memory is cleared automatically.	This is the auto buffer clear mode, in which all of the received packets are discarded.
Clearing method	Cleared by writing 1	1: Mode valid 0: Mode invalid	1: Mode valid 0: Mode invalid

### 29.4.5 FIFO Port Functions

This section describes the FIFO port functions. Table 29.21 shows the settings for the FIFO port functions of this module. In write access, writing data until the buffer is full (or the maximum packet size for non-continuous transfers) automatically enables sending of the data to the USB bus. To enable sending of data before the buffer is full (or before the maximum packet size for non-continuous transfers), the BVAL bit in CFIFOCTR/DnFIFOCTR must be set to end the writing (TEND signal for DMA transfers). Also, to send a zero-length packet, the BCLR bit in the same register must be used to clear the buffer and then the BVAL bit is set in order to end the writing.

In read access, reception of new packets is automatically enabled if all of the data has been read. Data cannot be read when a zero-length packet has been received (DTLN = 0), so the BCLR bit in the same register must be used to clear the buffer. The length of the data being received can be confirmed using the DTLN bit in CFIFOCTR/DnFIFOCTR.

**Table 29.21 FIFO Port Function Settings**

Register Name	Bit Name	Function	Note
CFIFOSEL/ DnFIFOSEL	RCNT	Selects DTLN read mode	
	REW	Buffer memory rewind (re-read, rewrite)	
	DCLRM	Automatically clears data received for a specified pipe after the data has been read	For DnFIFO only
	DREQE	Asserts the DREQ signal	For DnFIFO only
	MBW	FIFO port access bit width	
	BIGEND	Selects FIFO port endian	
	ISEL	FIFO port access direction	For DCP only
	CURPIPE	Selects the current pipe	
CFIFOCTR/ DnFIFOCTR	BVAL	Ends writing to the buffer memory	
	BCLR	Clears the buffer memory on the CPU side	
	FRDY	Monitors whether the FIFO port is ready for access	
	DTLN	Checks the length of received data	

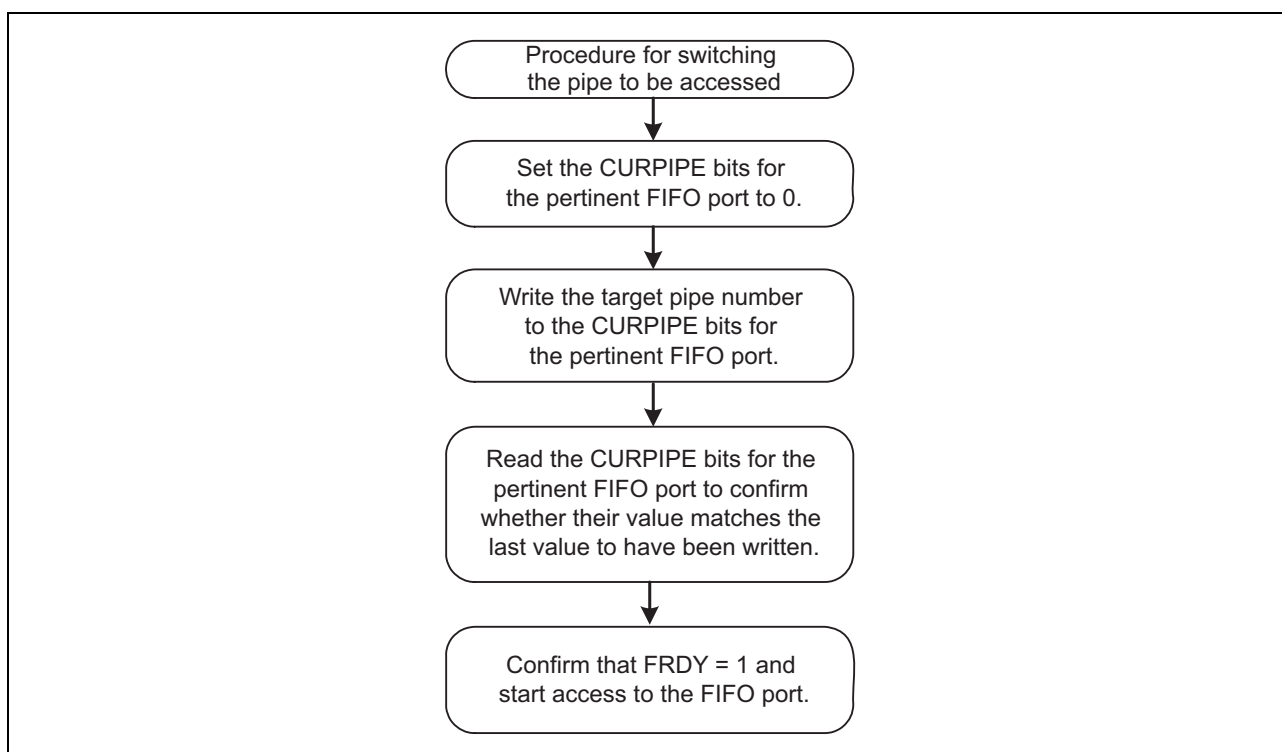
#### (a) FIFO Port Selection

Table 29.22 shows the pipes that can be selected with each FIFO port. The pipe to be accessed is selected using the CURPIPE bits in CFIFOSEL/DnFIFOSEL. After the pipe is selected, whether the CURPIPE value for the pipe which was written last can be correctly read should be checked. (If the previous pipe number is read, it indicates that the pipe switching is being executed by this module.) Then, the FIFO port can be accessed after FRDY = 1 is checked. Figure 29.6 shows the procedure for pipe switching during access to the FIFO port.

Also, the bus width to be accessed should be selected using the MBW bit. The buffer memory access direction conforms to the DIR bit in PIPEnCFG. The ISEL bit determines this only for the DCP.

**Table 29.22 FIFO Port Access Categorized by Pipe**

Pipe	Access Method	Port that can be Used
DCP	CPU access	CFIFO port register
PIPE1 to PIPE15	CPU access	CFIFO port register
	DMA access	D0FIFO/D1FIFO port register



**Figure 29.6 Procedure for Pipe Switching during Access to the FIFO Port**

**(b) DnFIFO Auto Clear Mode (D0FIFO/D1FIFO Port Reading Direction)**

If 1 is set for the DCLRM bit in DnFIFOSEL, the module automatically clears the buffer memory of the selected pipe when reading of the data from the buffer memory has been completed.

Table 29.23 shows the packet reception and buffer memory clearing processing for each of the various settings. As shown in Table 29.23, the buffer clear conditions depend on the value set to the BFRE bit. Using the DCLRM bit eliminates the need for the buffer to be cleared even if a situation occurs that necessitates clearing of the buffer. This makes it possible to carry out DMA transfers without involving software.

This function can be set only in the buffer memory reading direction.

**Table 29.23 Packet Reception and Buffer Memory Clearing Processing**

Buffer Status when Packet is Received	Register Setting			
	DCLRM = 0		DCLRM = 1	
	BFRE = 0	BFRE = 1	BFRE = 0	BFRE = 1
Buffer full	Doesn't need to be cleared	Doesn't need to be cleared	Doesn't need to be cleared	Doesn't need to be cleared
Zero-length packet reception	Needs to be cleared	Needs to be cleared	Doesn't need to be cleared	Doesn't need to be cleared
Normal short packet reception	Doesn't need to be cleared	Needs to be cleared	Doesn't need to be cleared	Doesn't need to be cleared
Transaction count ended	Doesn't need to be cleared	Needs to be cleared	Doesn't need to be cleared	Doesn't need to be cleared

### (c) Timing Selection Function for BRDY Interrupts

The BFRE bit of the PIPECFG register can be used to select non-generation of the BRDY interrupt when a data packet with the maximum packet size has been received.

For DMA transfer, this function enables the generation of an interrupt only when the last of the data have been received. Reception of the last of the data indicates either that a short packet was received or completion of counting by the transaction counter. If BFRE = 1, a BRDY interrupt is generated after the received data have been read. The length of the data packet that was last to have been received before the BRDY interrupt was generated can be confirmed by reading the DTLN bits of the DnFIFOCTR register.

Table 29.24 shows the times at which a BRDY interrupt is generated by this module.

**Table 29.24 Times BRDY Interrupts are Generated**

Buffer Status when Packet is Received	Register Setting	
	BFRE = 0	BFRE = 1
Buffer full (normal packet reception)	When a packet is received	No interrupt is generated.
Zero-length packet reception	When a packet is received	When a packet is received
Normal short packet reception	When a packet is received	When received data have been read from buffer memory
Transaction count ended	When a packet is received	When received data have been read from buffer memory

The BFRE bit function is only valid for the reading of buffer memory. In the case of writing, the value of the BFRE bit should always be 0.

## 29.4.6 Control Transfers (DCP)

Data transfers of the data stage of control transfers are done using the default control pipe (DCP). The DCP buffer memory is a 64-byte single buffer, and is a fixed area that is shared for both control reading and control writing. The buffer memory can be accessed through the CFIFO port.

### (1) Control Transfers when the Host Controller Mode is Selected

#### (a) Setup Stage

USBREQ, USBVAL, USBINDX, and USBLENG are the registers that are used to transmit a USB request for setup transactions. Writing setup packet data to the registers and writing 1 to the SUREQ bit in DCPCTR transmits the specified data for setup transactions. Upon completion of transactions, the SUREQ bit is cleared to 0 by this module. The above USB request registers should not be modified while SUREQ = 1. The device address for setup transactions is specified using the DEVSEL bits in DCPMAXP.

When the data for setup transactions has been sent, a SIGN or SACK interrupt request is generated according to the response received from the peripheral device (SIGN or SACK bit in INTSTS1), by means of which the result of the setup transactions can be confirmed.

A data packet of DATA0 (USB request) is transmitted as the data packet for the setup transactions regardless of the setting of the SQMON bit in DCPCTR.

#### (b) Data Stage

Data transfers are done using the DCP buffer memory.

The access direction of the DCP buffer memory should be specified using the ISEL bit in CFIFOSEL. The transfer direction should be specified using the DIR bit in DCPCFG.

For the first data packet of the data stage, the data PID must be transferred as DATA1. Accordingly, transaction should be done by setting the data PID to DATA1 by using the SQSET bit in DCPCTR, and setting the response PID to BUF by using the PID bits in DCPCTR. Completion of data transfer is detected using the BRDY and BEMP interrupts.

The data in multiple packets can be transferred in continuous transfer mode. For continuous transfer in the receiving direction, however, note that a BRDY interrupt is only generated when the buffer memory is full or a short packet is received (when the number of bytes of data is an integer multiple of the maximum packet size and no greater than 256 bytes).

For control write transfers, when the number of data bytes to be sent is the integral multiple of the maximum packet size, a zero-length packet must be sent at the end.

#### (c) Status Stage

Zero-length packet data transfers are done in the direction opposite to that in the data stage. As with the data stage, data transfers are done using the DCP buffer memory. Transactions are done in the same manner as the data stage.

For the data packets of the status stage, the data PID must be transferred as DATA1. The data PID should be set to DATA1 using the SQSET bit in DCPCTR.

For reception of a zero-length packet, the received data length must be confirmed using the DTLN bits in CFIFOCTR after the BRDY interrupt is generated, and the buffer memory must then be cleared using the BCLR bit.

## (2) Control Transfers when the Function Controller Mode is Selected

### (a) Setup Stage

This module always sends an ACK response in response to a setup packet that is normal with respect to this module. The operation of this module in the setup stage is noted below.

(i) When a new setup packet is received, this module sets the following registers:

- Set the VALID bit in INTSTS0 to 1.
- Set the PID bits in DCPCTR to NAK.
- Set the CCPL bit in DCPCTR to 0.

(ii) When a data packet is received right after the setup packet, the USB request parameters are stored in USBREQ, USBVAL, USBINDX, and USBLENG.

Response processing with respect to the control transfer should always be carried out after first setting VALID = 0. In the VALID = 1 state, PID = BUF cannot be set, and the data stage cannot be terminated.

Using the function of the VALID bit, this module is able to interrupt the processing of a request currently being processed if a new USB request is received during a control transfer, and can send a response in response to the newest request.

Also, this module automatically judges the direction bit (bit 8 of the bmRequestType) and the request data length (wLength) of the USB request that was received, and then distinguishes between control read transfers, control write transfers, and control write no-data transfers, and monitors the stage transition. For a wrong sequence, the sequence error of the control transfer stage transition interrupt is generated. For information on the stage control of this module, see Figure 29.3.

### (b) Data Stage

Data transfers corresponding to USB requests that have been received should be done using the DCP. Before accessing the DCP buffer memory, the access direction should be specified using the ISEL bit in CFIFOSEL.

Transaction should be done by setting the response PID to BUF by using the PID bits in DCPCTR.

Completion of data transfer is detected using the BRDY interrupt for control write transfers and BEMP interrupt for control read transfers, respectively.

With control write transfers during high-speed operation, the NYET handshake response is carried out based on the state of the buffer memory.

### (c) Status Stage

Control transfers are terminated by setting the CCPL bit to 1 with the PID bits in DCPCTR set to BUF.

After the above settings have been entered, this module automatically executes the status stage in accordance with the data transfer direction determined at the setup stage. The specific procedure is as follows.

- For control read transfers  
This module receives a zero-length packet from the USB host controller and transmits an ACK response.
- For control write transfers and no-data control transfers  
This module transmits a zero-length packet and receives an ACK response from the USB host controller.



#### (d) Control Transfer Auto Response Function

This module automatically responds to a normal SET\_ADDRESS request. If any of the following errors occur in the SET\_ADDRESS request, a response is necessary.

- bmRequestType ≠ H'00
- wIndex ≠ H'00
- wLength ≠ H'00
- wValue > H'7F
- DVSQ = B'011 (Configured)

For all requests other than the SET\_ADDRESS request, corresponding responses are required.

### 29.4.7 Bulk Transfers (PIPE1 to PIPE5, PIPE9 to PIPE15)

The buffer memory specifications for bulk transfers (single/double buffer setting, or continuous/non-continuous transfer mode setting) can be selected. The maximum size that can be set for the buffer memory is 2 Kbytes. The buffer memory state is controlled by this module, with a response sent automatically for a PING packet/NYET handshake.

#### (1) PING Packet Control when the Host Controller Mode is Selected

This module automatically sends a PING packet in the OUT direction at a specific timing.

On receiving an ACK handshake in the initial state in which PING packet sending mode is set, this module sends an OUT packet as noted below. Reception of an NAK or NYET handshake returns this module to PING packet sending mode. This control also applies to the control transfers in the data stage and status stage.

<Starts sending an OUT data>

1. Sets OUT data sending mode.
2. Sends a PING packet.
3. Receives an ACK handshake.
4. Sends an OUT data packet.
5. Receives an ACK handshake.  
(Repeats steps 4 and 5.)
6. Sends an OUT data packet.
7. Receives an NAK/NYET handshake.
8. Sends a PING packet.

This module is returned to PING packet sending mode by a power-on reset, receiving an NYET/NAK handshake, clearing the sequence toggle bit (SQCLR), and setting the buffer clear bit (ACLRM) in PIPEnCTR.

#### (2) NYET Handshake Control when the Function Controller Mode is Selected

Table 29.25 lists responses to tokens received in bulk or control transfer. The NYET response of this module is made when there is only enough space in the buffer memory for one packet when an OUT token for such transfer is received. When a short packet is received, an ACK response will be produced instead of an NYET response even if the above condition holds.

**Table 29.25 NYET Handshake Responses**

Value Set for PID Bits in DPCPTR	Buffer Memory State	Received Token	Response	Note
NAK/STALL	—	SETUP	ACK	—
	—	IN/OUT/PING	NAK/STALL	—
BUF	—	SETUP	ACK	—
	RCV-BRDY	OUT/PING	ACK	If an OUT token is received, a data packet is received.*1
	RCV-BRDY	OUT	NYET	A data packet is received.*2
	RCV-BRDY	OUT (Short)	ACK	A data packet is received.*2
	RCV-BRDY	PING	ACK	*2
	RCV-NRDY	OUT/PING	NAK	
	TRN-BRDY	IN	DATA0/DATA1	A data packet is transmitted.
	TRN-NRDY	IN	NAK	

## [Legend]

RCV-BRDY\*1:When an OUT/PING token is received, there is space in the buffer memory for two or more packets.

RCV-BRDY\*2:When an OUT token is received, there is only enough space in the buffer memory for one packet.

RCV-NRDY:When a PING token is received, there is no space in the buffer memory.

TRN-BRDY:When an IN token is received, there is data to be transmitted in the buffer memory.

TRN-NRDY:When an IN token is received, there is no data to be transmitted in the buffer memory.

### 29.4.8 Interrupt Transfers (PIPE6 to PIPE9, PIPE10)

When the function controller mode is selected, this module carries out interrupt transfers in accordance with the timing controlled by the host controller. For interrupt transfers, PING packets are ignored (no responses are sent), and the ACK, NAK, and STALL responses are carried out without an NYET handshake response being made.

When the host controller mode is selected, this module can set the timing of issuing a token using the interval counter. At this time, this module issues an OUT token even in the OUT direction, without issuing a PING token.

This module does not support high bandwidth transfers of interrupt transfers.

#### (1) Interval Counter during Interrupt Transfers when the Host Controller Mode is Selected

##### (a) Operation Outline

For interrupt transfers, intervals between transactions are set in the IITV bits in PIPEPERI. This module issues an interrupt transfer token based on the specified intervals.

##### (b) Counter Initialization

This module initializes the interval counter under the following conditions.

- Power-on reset:  
The IITV bits are initialized.
- Buffer memory initialization using the ACLRM bit:  
The IITV bits are not initialized but the count value is. Setting the ACLRM bit to 0 starts counting from the value set in the IITV bits.

Note that the interval counter is not initialized in the following case.

- USB bus reset, USB suspended:  
The IITV bits are not initialized. Setting 1 to the UACT bit starts counting from the value before entering the USB bus reset state or USB suspended state.

##### (c) Operation when Transmission/Reception is Impossible at Token Issuance Timing

This module cannot issue tokens even at token issuance timing in the following cases. In such a case, this module attempts transactions at the subsequent interval.

- When the PID is set to NAK or STALL.
- When the buffer memory is full at the token sending timing in the receiving (IN) direction.
- When there is no data to be sent in the buffer memory at the token sending timing in the sending (OUT) direction.

### 29.4.9 Isochronous Transfers (PIPE1 and PIPE2)

This module has the following functions pertaining to isochronous transfers.

- Notification of isochronous transfer error information
- Interval counter (specified by the IITV bits)
- Isochronous IN transfer data setup control (IDLTY function)
- Isochronous IN transfer buffer flush function (specified by the IFIS bit)
- SOF pulse output function

This module does not support the high bandwidth transfers of isochronous transfers.

#### (1) Error Detection with Isochronous Transfers

This module has a function for detecting the error information in isochronous transfers noted below. Table 29.26 and Table 29.27 show the priority in which errors are confirmed and the interrupts that are generated.

1. PID errors
  - If the PID of the packet being received is illegal
2. CRC errors and bit stuffing errors
  - If an error occurs in the CRC of the packet being received, or the bit stuffing is illegal
3. Exceeded maximum packet size
  - The data size of the received packet exceeded the specified maximum packet size.
4. Overrun and underrun errors
  - When host controller mode is selected
    - When using isochronous IN transfers (reception), the IN token was transmitted but there was not enough space in the buffer memory.
    - When using isochronous OUT transfers (transmission), the OUT token was transmitted, but the data was not in the buffer memory.
  - When function controller mode is selected
    - When using isochronous IN transfers (transmission), the IN token was received but the data was not in the buffer memory.
    - When using isochronous OUT transfers (reception), the OUT token was received, but there was not enough space in the buffer memory.
5. Interval errors

When function controller mode is selected, interval errors occur in following cases.

- During an isochronous IN transfer, the IN token could not be received during the interval frame.
- During an isochronous OUT transfer, the OUT token could not be received during the interval frame.

**Table 29.26 Error Detection when a Token is Received**

Detection Priority Order	Error	Generated Interrupt and Status
1	PID errors	No interrupts are generated in both cases when the host controller mode is selected and the function controller mode is selected (ignored as a corrupted packet).
2	CRC error and bit stuffing errors	No interrupts are generated in both cases when the host controller mode is selected and the function controller mode is selected (ignored as a corrupted packet).
3	Overflow and underrun errors	An NRDY interrupt is generated and the OVRN bit is set in both host controller mode and function controller mode. When the function controller mode is selected, a zero-length packet is transmitted in response to IN token. However, no data packet is received in response to OUT token.
4	Interval errors	An NRDY interrupt is generated when the function controller mode is selected. It is not generated in the host controller mode.

**Table 29.27 Error Detection when a Data Packet is Received**

Detection Priority Order	Error	Generated Interrupt and Status
1	PID errors	No interrupts are generated (ignored as a corrupted packet)
2	CRC error and bit stuffing errors	An NRDY interrupt is generated and the CRCE bit is set in both host controller mode and function controller mode.
3	Maximum packet size exceeded error	A BEMP interrupt is generated to set the PID bits to STALL in both cases when the host controller mode is selected and the function controller mode is selected.

## (2) DATA-PID

This module does not support high bandwidth transfers. When the function controller mode is selected, this module operates as follows in response to the received PID.

1. IN direction
  - DATA0: Sent as data packet PID
  - DATA1: Not sent
  - DATA2: Not sent
  - mData: Not sent
2. OUT direction (when using full-speed operation)
  - DATA0: Received normally as data packet PID
  - DATA1: Received normally as data packet PID
  - DATA2: Packets are ignored
  - mData: Packets are ignored
3. OUT direction (when using high-speed operation)
  - DATA0: Received normally as data packet PID
  - DATA1: Received normally as data packet PID
  - DATA2: Received normally as data packet PID
  - mData: Received normally as data packet PID

### (3) Interval Counter

#### (a) Operation Outline

The isochronous interval can be set using the IITV bits in PIPEPERI. The interval counter enables the functions shown in Table 29.28 when the function controller mode is selected. When the host controller mode is selected, this module generates the token issuance timing. When the host controller mode is selected, the interval counter operation is the same as the interrupt transfer operation. Refer to section 29.4.8 (1) Interval Counter during Interrupt Transfers when the Host Controller Mode is Selected.

**Table 29.28 Functions of the Interval Counter when the Function Controller Mode is Selected**

Transfer Direction	Function	Conditions for Detection
IN	Transmission buffer flush function	When an IN token cannot be normally received in the interval frame during an isochronous IN transfer
OUT	Notifies that a token not being received	When an OUT token cannot be normally received in the interval frame during an isochronous OUT transfer

The interval count is carried out when an SOF is received or for interpolated SOFs, so the isochronism can be maintained even if an SOF is damaged. The frame interval that can be set is the  $2^{IITV}$  frames or  $2^{IITV}$   $\mu$  frames.

#### (b) Interval Counter Initialization when the Function Controller Mode is Selected

This module initializes the interval counter under the following conditions.

- Power-on reset  
The IITV bits are initialized.
- Buffer memory clearing using the ACLRM bit  
The IITV bits are not initialized but the count value is.
- USB bus reset

After the interval counter has been initialized, the counter is started under the following condition 1 or 2 when a packet has been transferred normally.

1. An SOF is received following transmission of data in response to an IN token, in the PID = BUF state.
2. An SOF is received after data following an OUT token is received in the PID = BUF state.

The interval counter is not initialized under the conditions below.

1. When the PID bits are set to NAK or STALL  
The interval timer does not stop. This module attempts the transactions at the subsequent interval.
2. The USB bus is reset or USB operations are suspended  
The IITV bits are not initialized. When the SOF has been received, the counter is restarted from the value prior to the reception of the SOF.

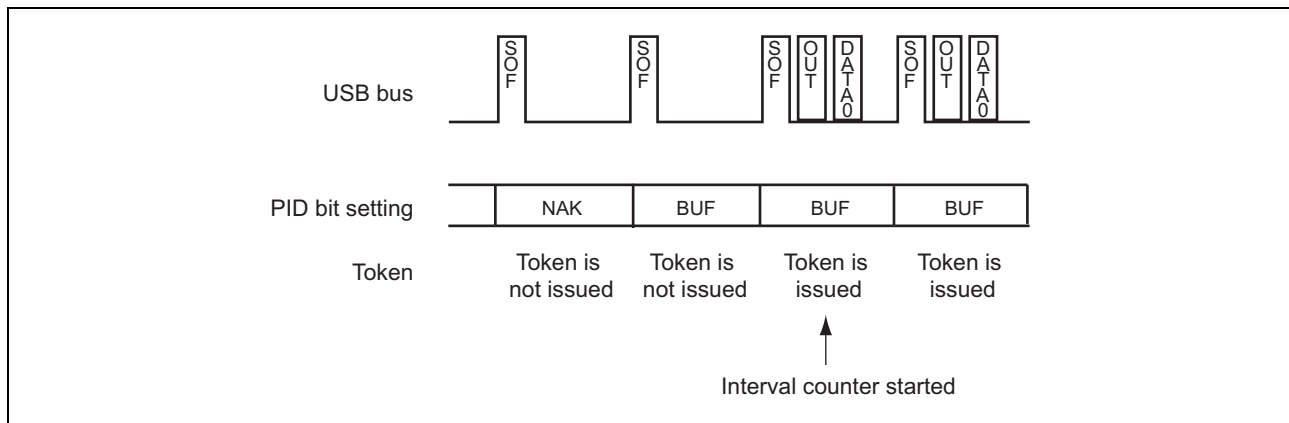
#### (c) Interval Counting and Transfer Control when the Host Controller Mode is Selected

The IITV bits can be set when the selected pipe is for isochronous or interrupt transfers. This module controls the interval between token issuance operations based on the IITV bit setting. Specifically, this module issues a token for the selected pipe once every  $2^{IITV}$  ( $\mu$ ) frames.

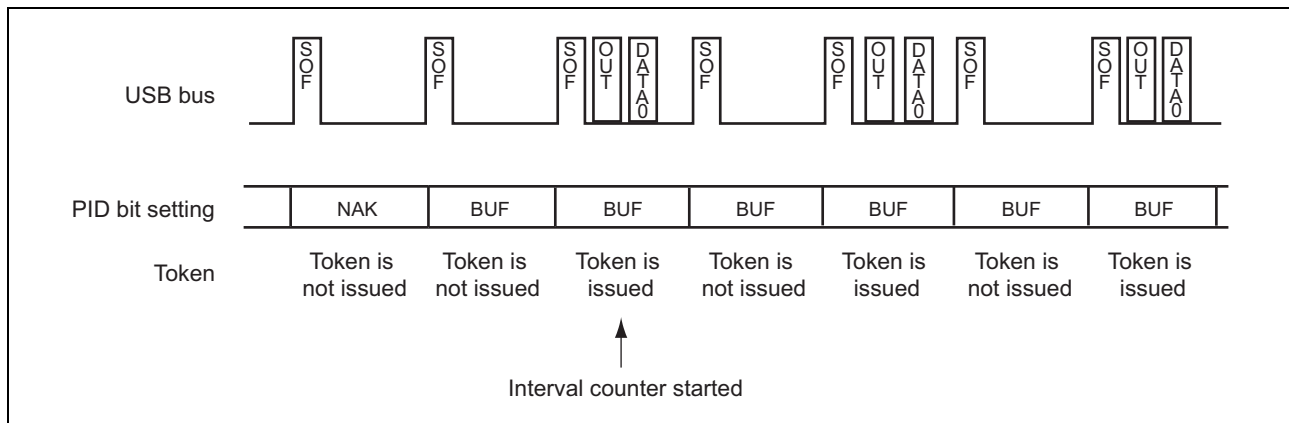
This module counts the interval every 1-ms frame for the pipes used for communications with the full-speed or low-speed peripheral devices connected to a high-speed HUB.

This module starts counting the token issuance interval at the ( $\mu$ ) frame following the ( $\mu$ ) frame in which the PID bits are

set to BUF.



**Figure 29.7** Token Issuance when IITV = 0



**Figure 29.8** Token Issuance when IITV = 1

When the selected pipe is for isochronous transfers, this module carries out the operation below in addition to controlling token issuance interval. This module issues a token even when the NRDY interrupt generation condition is satisfied.

1. When the selected pipe is for isochronous IN transfers

This module generates the NRDY interrupt when this module issues the IN token but does not receive a packet successfully from a peripheral device (no response or packet error).

This module sets the OVRN bit to 1 generating the NRDY interrupt when the time to issue an IN token comes in a state in which this module cannot receive data because the FIFO buffer is full (because reading data from the FIFO buffer is slow).

2. When the selected pipe is for isochronous OUT transfers

This module sets the OVRN bit to 1 generating the NRDY interrupt and transmitting a zero-length packet when the time to issue an OUT token comes in a state in which there is no data to be transmitted in the FIFO buffer (because writing data to the FIFO buffer is slow).

The token issuance interval is reset by a power-on reset or when the ACLRM bit is set to 1.

(d) Interval Counting and Transfer Control when the Function Controller Mode is Selected

The IITV bits can be set when the selected pipe is for isochronous transfers.

1. When the selected pipe is for isochronous OUT transfers

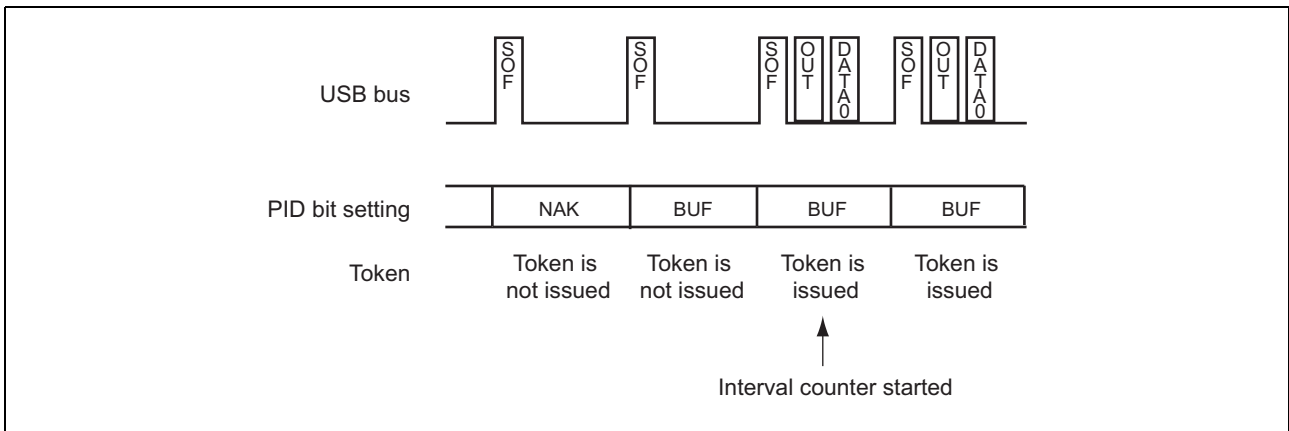
This module generates the NRDY interrupt when it fails to receive a data packet within the interval set for ( $\mu$ ) frames by the IITV bits.

This module generates the NRDY interrupt when this module fails to receive a data packet because of a CRC error or other errors contained in the packet, or because of the FIFO buffer being full (because reading data from the FIFO buffer is slow).

This module generates the NRDY interrupt on receiving an SOF packet. Even if the SOF packet is corrupted, the internal interpolation is used and allows the interrupt to be generated at the timing to receive the SOF packet. However, when the IITV bits are set to the value other than 0, this module generates the NRDY interrupt on receiving an SOF packet for every interval after starting interval counting operation. When the PID bits are set to NAK after starting the interval timer, this module does not generate the NRDY interrupt on receiving an SOF packet.

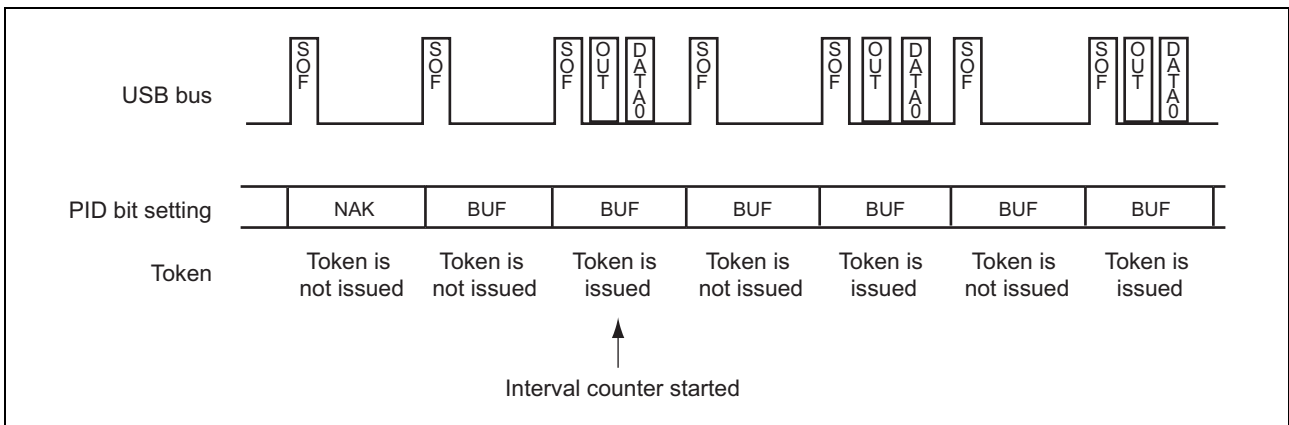
The interval counting starts at the different timing depending on the IITV bit setting as follows.

- When IITV = 0: The interval counting starts at the (μ) frame following the (μ) frame in which the PID bits for the selected pipe are set to BUF.



**Figure 29.9 Relationship between (μ) Frames and Expected Token Reception when IITV = 0**

- When IITV ≠ 0: The interval counting starts on completion of successful reception of the first data packet after the PID bits for the selected pipe have been modified to BUF.



**Figure 29.10 Relationship between (μ) Frames and Expected Token Reception when IITV = 1**

2. When the selected pipe is for isochronous IN transfers

The IFIS bit should be 1 for this use. When IFIS = 0, this module transmits a data packet in response to the received IN token irrespective of the IITV bit setting.

When IFIS = 1, this module clears the FIFO buffer when this module fails to receive an IN token within the interval set for (μ) frames by the IITV bits in a state in which there is data to be transmitted in the FIFO buffer.

This module also clears the FIFO buffer when this module fails to receive an IN token successfully because of a bus error such as a CRC error contained in the token.



This module clears the FIFO buffer on receiving an SOF packet. Even if the SOF packet is corrupted, the internal interpolation is used and allows the FIFO buffer to be cleared at the timing to receive the SOF packet.

The interval counting starts at the different timing depending on the IITV bit setting (similar to the timing during OUT transfers).

The clearing conditions for the interval counter are any of the following in function controller mode.

- When a power-on-reset is applied to this module (the value set in the IITV bits is also cleared to 0).
- When the ACLRM bit is set to 1.
- When this module detects a USB bus reset.

#### (4) Setup of Data to be Transmitted using Isochronous Transfer when the Function Controller Mode is Selected

With isochronous data transmission using this module in function controller mode, after data has been written to the buffer memory, a data packet can be sent with the next frame in which an SOF packet is detected. This function is called the isochronous transfer transmission data setup function. This function enables identification of the frame that has started being transmitted.

When a double buffer is in use as the buffer memory, transmission from only one of the two buffers will be possible even after the writing of data to both buffers has been completed; the given buffer memory will be that to which the writing of data was completed first. For this reason, even if multiple IN tokens are received in a single frame, the only buffer memory that can be sent is one packet's worth of data.

When an IN token is received, if the buffer memory is in the transmission enabled state, this module transmits the data as a normal response. If the buffer memory is not in the transmission enabled state, however, a zero-length packet is sent and an underrun error occurs.

Figure 29.11 shows an example of transmission using the isochronous transfer transmission data setup function with this module, when IITV = 0 (every frame) has been set.

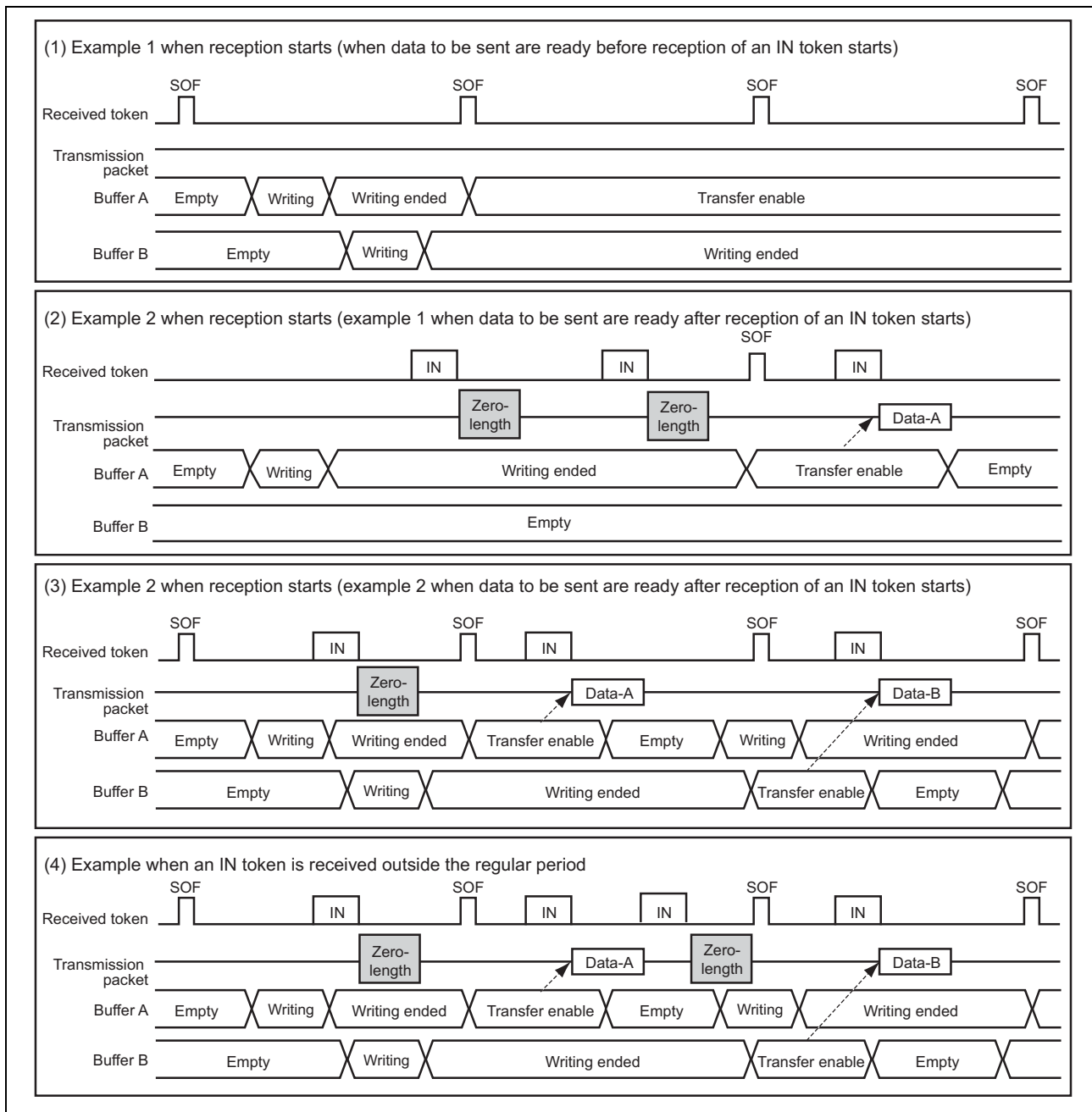


Figure 29.11 Example of Data Setup Function Operation

(5) Isochronous Transfer Transmission Buffer Flush when the Function Controller Mode is Selected

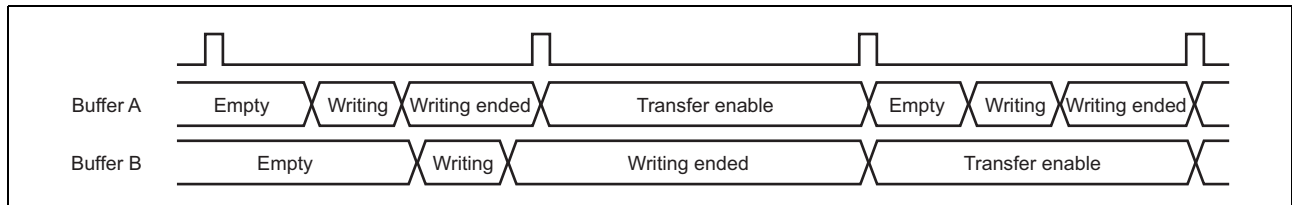
When the function controller mode is selected and an SOF packet or a  $\mu$ SOF packet of the next frame is received without receiving an IN token in the interval frame during isochronous data transmission, this module operates as if a corrupted IN token was received, and clears the buffer for which transmission is enabled, putting that buffer in the writing enabled state.

If a double buffer is being used and writing to both buffers has been completed, data are considered to have been sent from the buffer memory that was cleared in the same interval frame, and transmission is enabled for the buffer memory that is not discarded with SOF or  $\mu$ SOF packets reception.

The timing at which the buffer flush function is activated varies depending on the value set for the IITV bits.

- If IITV = 0  
The buffer flush operation starts from the next frame after the pipe becomes valid.
- In any cases other than IITV = 0  
The buffer flush operation is carried out subsequent to the first normal transaction.

Figure 29.12 shows an example of the buffer flush function of this module. When an unanticipated token is received prior to the interval frame, this module sends the written data or a zero-length packet as an underrun error according to the data setup state.



**Figure 29.12 Example of Buffer Flush Function Operation**

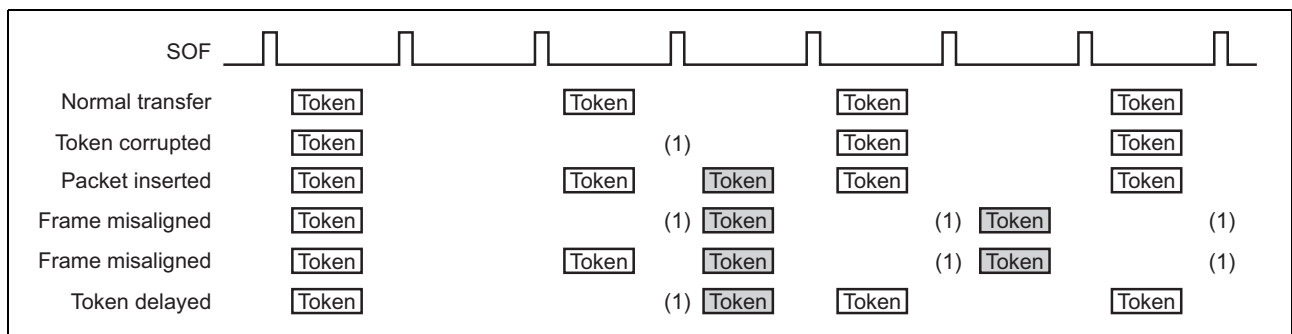
Figure 29.13 shows an example of this module generating an interval error. There are five types of interval errors, as shown below. The interval error is generated at the timing indicated by (1) in the figure, and the buffer flush function is activated.

If an interval error occurs during an IN transfer, the buffer flush function is activated; and if it occurs during an OUT transfer, an NRDY interrupt is generated.

The OVRN bit should be used to distinguish between NRDY interrupts such as received packet errors and overrun errors.

In response to tokens that are shaded in the figure, responses occur based on the buffer memory status.

1. IN direction
  - If the buffer is in the transmission enabled state, the data is transferred as a normal response.
  - If the buffer is in the transmission disabled state, a zero-length packet is sent and an underrun error occurs.
2. OUT direction
  - If the buffer is in the reception enabled state, the data is received as a normal response.
  - If the buffer is in the reception disabled state, the data is discarded and an overrun error occurs.



**Figure 29.13 Example of an Interval Error being Generated when IITV = 1**

### 29.4.10 SOF Interpolation Function

When the function controller mode is selected and if an SOF packet could not be received at intervals of 1 ms (when using full-speed operation) or 125  $\mu$ s (when using high-speed operation) because of corruption or missing, this module interpolates the SOF. The SOF interpolation operation begins when both the USBE bit in SYSCFG and the SUSPM bit in SUSPMODE have been set to 1 and an SOF packet is received. The interpolation function is initialized under the following conditions.

- Power-on reset
- USB bus reset
- Suspended state detected

Also, the SOF interpolation operates under the following specifications.

- Frame interval (125  $\mu$ s or 1 ms) conforms to the results of the reset handshake protocol.
- The interpolation function is not activated until an SOF packet is received.
- After the first SOF packet is received, either 125  $\mu$ s or 1 ms is counted with an internal clock of 48 MHz, and interpolation is carried out.
- After the second and subsequent SOF packets are received, interpolation is carried out at the previous reception interval.
- Interpolation is not carried out in the suspended state or while a USB bus reset is being received. (With suspended transitions in high-speed operation, interpolation continues for 3 ms after the last packet is received.)

This module supports the following functions based on the SOF reception. These functions also operate normally with SOF interpolation, if the SOF packet was missing.

- Refreshing of the frame number and micro-frame number
- SOFR interrupt and  $\mu$ SOF lock
- SOF pulse output
- Isochronous transfer interval count

If an SOF packet is missing when full-speed operation is being used, the FRNM bits in FRMNUM are not refreshed. If a  $\mu$ SOF packet is missing during high-speed operation, the UFRNM bits in UFRMNUM are refreshed. However, if a  $\mu$ SOF packet for UFRNM = 000 is missing, the FRNM bits are not refreshed. In this case, the FRNM bits are not refreshed even if successive  $\mu$ SOF packets other than UFRNM = 000 are received normally.

## 30. Digital Video Decoder

### 30.1 Features

This LSI incorporates two channels of digital video decoders. This module consists of an A/D converter for video signal input, a sync separator circuit, a burst controlled oscillator (BCO), a 2D Y/C separator circuit, chroma decoding circuit, a digital clamp circuit, and an output gain adjustment circuit. Table 30.1 shows the digital video decoder functions.

**Table 30.1 Digital Video Decoder Functions**

Item	Function
Input signal	Video signal <ul style="list-style-type: none"> <li>• Composite video signal (CVBS)</li> </ul>
Functional outline	<ul style="list-style-type: none"> <li>• A/D converter for video signal input               <ul style="list-style-type: none"> <li>— VIN1 and VIN2 pin input selection</li> <li>— Low-pass filter (LPF)</li> <li>— Sync tip clamp</li> <li>— Programmable gain amplifier (PGA) (0 to 6.021 dB)</li> <li>— 10-bit precision pipelined A/D converter</li> </ul> </li> <li>• Sync separation               <ul style="list-style-type: none"> <li>— Noise reduction LPS, auto level control sync slicer, horizontal auto frequency control (AFC), vertical count-down, interlace detection, auto gain control (AGC)/peak limiter control</li> </ul> </li> <li>• Burst controlled oscillator (BCO)               <ul style="list-style-type: none"> <li>— Color sub-carrier reproduction, color system detection (For details, see Table 30.3.)</li> </ul> </li> <li>• Y/C separation (For details, see Table 30.2.)               <ul style="list-style-type: none"> <li>— Supporting NTSC 2D, PAL 2D, SECAM 1D</li> </ul> </li> <li>• Chroma decoding               <ul style="list-style-type: none"> <li>— Supporting NTSC, PAL SECAM</li> <li>— Color killer, auto color control (ACC), TINT correction, R-Y axis correction</li> </ul> </li> <li>• Digital clamp               <ul style="list-style-type: none"> <li>— Pedestal clamp (Y), center clamp (Cb/Cr), noise detection</li> </ul> </li> <li>• Output gain adjustment               <ul style="list-style-type: none"> <li>— Contrast adjustment: 0 to approx. two times</li> <li>— Color adjustment (Cb/Cr independent): 0 to approx. two times</li> </ul> </li> </ul>

**Table 30.2 Supported Y/C Separation Operation**

Color System	Y/C Separation Operation
NTSC-3.58	Two dimensional
NTSC-4.43	Two dimensional
PAL-M	Two dimensional
PAL-N	Two dimensional
PAL-4.43	Two dimensional
SECAM	One dimensional

**Table 30.3 Color System Detection**

COLORSYS[1:0]	FSCMODE	FVMODE	Detection Result
0: NTSC	0: 3.58 MHz	Don't care	NTSC-M
0: NTSC	1: 4.43 MHz	Don't care	NTSC-4.43
1: PAL	0: 3.58 MHz	0: 50 Hz	PAL-N
1: PAL	0: 3.58 MHz	1: 60 Hz	PAL-M
1: PAL	1: 4.43 MHz	0: 50 Hz	PAL-B, H, I, G, D
1: PAL	1: 4.43 MHz	1: 60 Hz	PAL-60
2: SECAM	—	—	SECAM
3: Unknown	—	—	Cannot be detected

### 30.2 Block Diagram

Figure 30.1 shows a block diagram of this module.

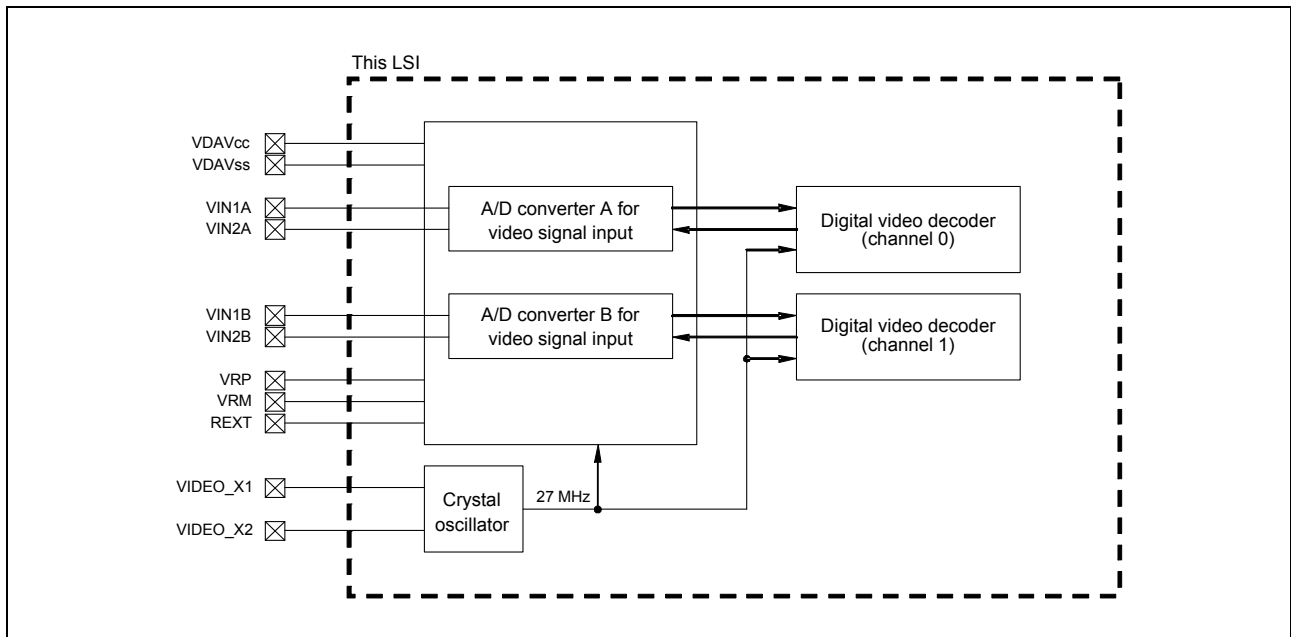


Figure 30.1 Block Diagram of This Module

Figure 30.2 shows a detailed block diagram of each channel.

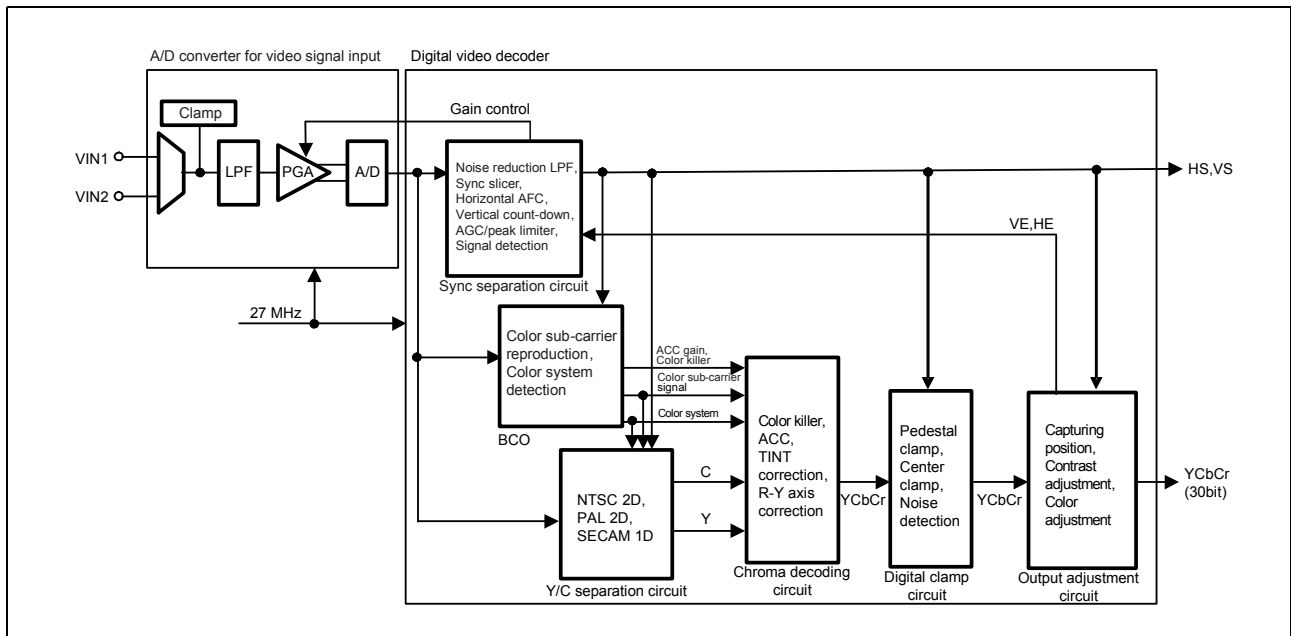


Figure 30.2 Detailed Block Diagram of Each Channel

### 30.3 Input/Output Pins

Table 30.4 shows the pin configuration.

**Table 30.4 Pin Configuration**

Category	Name	Pin Symbol	I/O	Description
Signal	Composite video signal input	VIN1A	Input	Composite video signal (CVBS) input pin 1 (channel 0)
		VIN2A	Input	Composite video signal (CVBS) input pin 2 (channel 0)
		VIN1B	Input	Composite video signal (CVBS) input pin 1 (channel 1)
		VIN2B	Input	Composite video signal (CVBS) input pin 2 (channel 1)
Clock	Crystal oscillator/ external clock	VIDEO_X1	Input	Connect to a crystal resonator for the digital video decoder. The VIDEO_X1 pin can also be used for external clock input.
		VIDEO_X2	Output	
Reference voltage	TOP reference voltage	VRP	Output	TOP reference voltage pin for the A/D converter for video signal input Connect to the VDAVss via a 0.1- $\mu$ F capacitor.
	BOTTOM reference voltage	VRM	Output	BOTTOM reference voltage pin for the A/D converter for video signal input Connect to the VDAVss via a 0.1- $\mu$ F capacitor.
	Reference voltage	REXT	Output	Reference voltage pin for the A/D converter for video signal input Connect to the VDAVss via a 22-k $\Omega$ $\pm$ 1% resistor.
Power supply	Analog power supply	VDAVcc	Input	Power supply pin for the A/D converter for video signal input
	Analog ground	VDAVss	Input	Ground pin for the A/D converter for video signal input

## 30.4 Register Descriptions

Table 30.5 shows the register configuration. The channel can be identified by the number 0 or 1 appended to the register name (e.g. ADCCR1\_0 for channel 0).

**Table 30.5 Register Configuration**

Channel	Register Name	Abbr.	R/W	Address	Access Size
0	ADC control register 1_0	ADCCR1_0	R/W	H'FCFFB808	16
	Timing generation control register 1_0	TGCR1_0	R/W	H'FCFFB80E	16
	Timing generation control register 2_0	TGCR2_0	R/W	H'FCFFB810	16
	Timing generation control register 3_0	TGCR3_0	R/W	H'FCFFB812	16
	Sync separation control register 1_0	SYNSCR1_0	R/W	H'FCFFB81A	16
	Sync separation control register 2_0	SYNSCR2_0	R/W	H'FCFFB81C	16
	Sync separation control register 3_0	SYNSCR3_0	R/W	H'FCFFB81E	16
	Sync separation control register 4_0	SYNSCR4_0	R/W	H'FCFFB820	16
	Sync separation control register 5_0	SYNSCR5_0	R/W	H'FCFFB822	16
	Horizontal AFC control register 1_0	HAFCCR1_0	R/W	H'FCFFB824	16
	Horizontal AFC control register 2_0	HAFCCR2_0	R/W	H'FCFFB826	16
	Horizontal AFC control register 3_0	HAFCCR3_0	R/W	H'FCFFB828	16
	Vertical countdown control register 1_0	VCDWCR1_0	R/W	H'FCFFB82A	16
	Digital clamp control register 1_0	DCPCR1_0	R/W	H'FCFFB830	16
	Digital clamp control register 2_0	DCPCR2_0	R/W	H'FCFFB832	16
	Digital clamp control register 3_0	DCPCR3_0	R/W	H'FCFFB834	16
	Digital clamp control register 4_0	DCPCR4_0	R/W	H'FCFFB836	16
	Digital clamp control register 5_0	DCPCR5_0	R/W	H'FCFFB838	16
	Digital clamp control register 6_0	DCPCR6_0	R/W	H'FCFFB83A	16
	Digital clamp control register 7_0	DCPCR7_0	R/W	H'FCFFB83C	16
	Digital clamp control register 8_0	DCPCR8_0	R/W	H'FCFFB83E	16
	Noise detection control register_0	NSDCR_0	R/W	H'FCFFB840	16
	Burst lock/chroma decoding control register_0	BTLCR_0	R/W	H'FCFFB842	16
	Burst gate pulse control register_0	BTGPCR_0	R/W	H'FCFFB844	16
	ACC control register 1_0	ACCCR1_0	R/W	H'FCFFB846	16
	ACC control register 2_0	ACCCR2_0	R/W	H'FCFFB848	16
	ACC control register 3_0	ACCCR3_0	R/W	H'FCFFB84A	16
	TINT control register_0	TINTCR_0	R/W	H'FCFFB84C	16
	Y/C delay/chroma decoding control register_0	YDCR_0	R/W	H'FCFFB84E	16
	AGC control register 1_0	AGCCR1_0	R/W	H'FCFFB850	16
	AGC control register 2_0	AGCCR2_0	R/W	H'FCFFB852	16
	Peak limiter control register_0	PKLIMITCR_0	R/W	H'FCFFB854	16
	Over-range control register 1_0	RGORCR1_0	R/W	H'FCFFB856	16
	Over-range control register 2_0	RGORCR2_0	R/W	H'FCFFB858	16
	Over-range control register 3_0	RGORCR3_0	R/W	H'FCFFB85A	16
	Over-range control register 4_0	RGORCR4_0	R/W	H'FCFFB85C	16
	Over-range control register 5_0	RGORCR5_0	R/W	H'FCFFB85E	16
	Over-range control register 6_0	RGORCR6_0	R/W	H'FCFFB860	16
	Over-range control register 7_0	RGORCR7_0	R/W	H'FCFFB862	16
	Feedback control register for horizontal AFC phase comparator_0	AFCPFCR_0	R/W	H'FCFFB87C	16



**Table 30.5 Register Configuration**

Channel	Register Name	Abbr.	R/W	Address	Access Size
0	Register update enable register_0	RUPDCR_0	R/W	H'FCFFB87E	16
	Sync separation status/vertical cycle read register_0	VSYNCSR_0	R	H'FCFFB880	16
	Horizontal cycle read register_0	HSYNCSR_0	R	H'FCFFB882	16
	Digital clamp read register 1_0	DCPSR1_0	R	H'FCFFB884	16
	Digital clamp read register 2_0	DCPSR2_0	R	H'FCFFB886	16
	Noise detection read register_0	NSDSR_0	R	H'FCFFB88C	16
	Chroma decoding read register 1_0	CROMASR1_0	R	H'FCFFB88E	16
	Chroma decoding read register 2_0	CROMASR2_0	R	H'FCFFB890	16
	Sync separation read register_0	SYNCSR_0	R	H'FCFFB892	16
	AGC control read register 1_0	AGCSR1_0	R	H'FCFFB894	16
	AGC control read register 2_0	AGCSR2_0	R	H'FCFFB896	16
	Y/C separation control register 3_0	YCSCR3_0	R/W	H'FCFFB904	16
	Y/C separation control register 4_0	YCSCR4_0	R/W	H'FCFFB906	16
	Y/C separation control register 5_0	YCSCR5_0	R/W	H'FCFFB908	16
	Y/C separation control register 6_0	YCSCR6_0	R/W	H'FCFFB90A	16
	Y/C separation control register 7_0	YCSCR7_0	R/W	H'FCFFB90C	16
	Y/C separation control register 8_0	YCSCR8_0	R/W	H'FCFFB90E	16
	Y/C separation control register 9_0	YCSCR9_0	R/W	H'FCFFB910	16
	Y/C separation control register 11_0	YCSCR11_0	R/W	H'FCFFB914	16
	Y/C separation control register 12_0	YCSCR12_0	R/W	H'FCFFB916	16
	Digital clamp control register 9_0	DCPCR9_0	R/W	H'FCFFB980	16
	Chroma filter TAP coefficient (WA_F0) register for Y/C separation_0	YCTWA_F0_0	R/W	H'FCFFB992	16
	Chroma filter TAP coefficient (WA_F1) register for Y/C separation_0	YCTWA_F1_0	R/W	H'FCFFB994	16
	Chroma filter TAP coefficient (WA_F2) register for Y/C separation_0	YCTWA_F2_0	R/W	H'FCFFB996	16
	Chroma filter TAP coefficient (WA_F3) register for Y/C separation_0	YCTWA_F3_0	R/W	H'FCFFB998	16
	Chroma filter TAP coefficient (WA_F4) register for Y/C separation_0	YCTWA_F4_0	R/W	H'FCFFB99A	16
	Chroma filter TAP coefficient (WA_F5) register for Y/C separation_0	YCTWA_F5_0	R/W	H'FCFFB99C	16
	Chroma filter TAP coefficient (WA_F6) register for Y/C separation_0	YCTWA_F6_0	R/W	H'FCFFB99E	16
	Chroma filter TAP coefficient (WA_F7) register for Y/C separation_0	YCTWA_F7_0	R/W	H'FCFFB9A0	16
	Chroma filter TAP coefficient (WA_F8) register for Y/C separation_0	YCTWA_F8_0	R/W	H'FCFFB9A2	16
	Chroma filter TAP coefficient (WB_F0) register for Y/C separation_0	YCTWB_F0_0	R/W	H'FCFFB9A4	16
	Chroma filter TAP coefficient (WB_F1) register for Y/C separation_0	YCTWB_F1_0	R/W	H'FCFFB9A6	16
	Chroma filter TAP coefficient (WB_F2) register for Y/C separation_0	YCTWB_F2_0	R/W	H'FCFFB9A8	16
	Chroma filter TAP coefficient (WB_F3) register for Y/C separation_0	YCTWB_F3_0	R/W	H'FCFFB9AA	16
	Chroma filter TAP coefficient (WB_F4) register for Y/C separation_0	YCTWB_F4_0	R/W	H'FCFFB9AC	16

**Table 30.5 Register Configuration**

Channel	Register Name	Abbr.	R/W	Address	Access Size
0	Chroma filter TAP coefficient (WB_F5) register for Y/C separation_0	YCTWB_F5_0	R/W	H'FCFFB9AE	16
	Chroma filter TAP coefficient (WB_F6) register for Y/C separation_0	YCTWB_F6_0	R/W	H'FCFFB9B0	16
	Chroma filter TAP coefficient (WB_F7) register for Y/C separation_0	YCTWB_F7_0	R/W	H'FCFFB9B2	16
	Chroma filter TAP coefficient (WB_F8) register for Y/C separation_0	YCTWB_F8_0	R/W	H'FCFFB9B4	16
	Chroma filter TAP coefficient (NA_F0) register for Y/C separation_0	YCTNA_F0_0	R/W	H'FCFFB9B6	16
	Chroma filter TAP coefficient (NA_F1) register for Y/C separation_0	YCTNA_F1_0	R/W	H'FCFFB9B8	16
	Chroma filter TAP coefficient (NA_F2) register for Y/C separation_0	YCTNA_F2_0	R/W	H'FCFFB9BA	16
	Chroma filter TAP coefficient (NA_F3) register for Y/C separation_0	YCTNA_F3_0	R/W	H'FCFFB9BC	16
	Chroma filter TAP coefficient (NA_F4) register for Y/C separation_0	YCTNA_F4_0	R/W	H'FCFFB9BE	16
	Chroma filter TAP coefficient (NA_F5) register for Y/C separation_0	YCTNA_F5_0	R/W	H'FCFFB9C0	16
	Chroma filter TAP coefficient (NA_F6) register for Y/C separation_0	YCTNA_F6_0	R/W	H'FCFFB9C2	16
	Chroma filter TAP coefficient (NA_F7) register for Y/C separation_0	YCTNA_F7_0	R/W	H'FCFFB9C4	16
	Chroma filter TAP coefficient (NA_F8) register for Y/C separation_0	YCTNA_F8_0	R/W	H'FCFFB9C6	16
	Chroma filter TAP coefficient (NB_F0) register for Y/C separation_0	YCTNB_F0_0	R/W	H'FCFFB9C8	16
	Chroma filter TAP coefficient (NB_F1) register for Y/C separation_0	YCTNB_F1_0	R/W	H'FCFFB9CA	16
	Chroma filter TAP coefficient (NB_F2) register for Y/C separation_0	YCTNB_F2_0	R/W	H'FCFFB9CC	16
	Chroma filter TAP coefficient (NB_F3) register for Y/C separation_0	YCTNB_F3_0	R/W	H'FCFFB9CE	16
	Chroma filter TAP coefficient (NB_F4) register for Y/C separation_0	YCTNB_F4_0	R/W	H'FCFFB9D0	16
	Chroma filter TAP coefficient (NB_F5) register for Y/C separation_0	YCTNB_F5_0	R/W	H'FCFFB9D2	16
	Chroma filter TAP coefficient (NB_F6) register for Y/C separation_0	YCTNB_F6_0	R/W	H'FCFFB9D4	16
	Chroma filter TAP coefficient (NB_F7) register for Y/C separation_0	YCTNB_F7_0	R/W	H'FCFFB9D6	16
	Chroma filter TAP coefficient (NB_F8) register for Y/C separation_0	YCTNB_F8_0	R/W	H'FCFFB9D8	16
	Luminance (Y) signal gain control register_0	YGAINCR_0	R/W	H'FCFFBA00	16
	Color difference (Cb) signal gain control register_0	CBGAINCR_0	R/W	H'FCFFBA02	16
	Color difference (Cr) signal gain control register_0	CRGAINCR_0	R/W	H'FCFFBA04	16
	PGA register update_0	PGA_UPDATE_0	R/W	H'FCFFBA80	16
	PGA control register_0	PGACR_0	R/W	H'FCFFBA82	16
	ADC control register_2_0	ADCCR2_0	R/W	H'FCFFBA84	16

**Table 30.5 Register Configuration**

Channel	Register Name	Abbr.	R/W	Address	Access Size
1	ADC control register 1_1	ADCCR1_1	R/W	H'FCFFA008	16
	Timing generation control register 1_1	TGCR1_1	R/W	H'FCFFA00E	16
	Timing generation control register 2_1	TGCR2_1	R/W	H'FCFFA010	16
	Timing generation control register 3_1	TGCR3_1	R/W	H'FCFFA012	16
	Sync separation control register 1_1	SYNSCR1_1	R/W	H'FCFFA01A	16
	Sync separation control register 2_1	SYNSCR2_1	R/W	H'FCFFA01C	16
	Sync separation control register 3_1	SYNSCR3_1	R/W	H'FCFFA01E	16
	Sync separation control register 4_1	SYNSCR4_1	R/W	H'FCFFA020	16
	Sync separation control register 5_1	SYNSCR5_1	R/W	H'FCFFA022	16
	Horizontal AFC control register 1_1	HAFCCR1_1	R/W	H'FCFFA024	16
	Horizontal AFC control register 2_1	HAFCCR2_1	R/W	H'FCFFA026	16
	Horizontal AFC control register 3_1	HAFCCR3_1	R/W	H'FCFFA028	16
	Vertical countdown control register 1_1	VCDWCR1_1	R/W	H'FCFFA02A	16
	Digital clamp control register 1_1	DCPCR1_1	R/W	H'FCFFA030	16
	Digital clamp control register 2_1	DCPCR2_1	R/W	H'FCFFA032	16
	Digital clamp control register 3_1	DCPCR3_1	R/W	H'FCFFA034	16
	Digital clamp control register 4_1	DCPCR4_1	R/W	H'FCFFA036	16
	Digital clamp control register 5_1	DCPCR5_1	R/W	H'FCFFA038	16
	Digital clamp control register 6_1	DCPCR6_1	R/W	H'FCFFA03A	16
	Digital clamp control register 7_1	DCPCR7_1	R/W	H'FCFFA03C	16
	Digital clamp control register 8_1	DCPCR8_1	R/W	H'FCFFA03E	16
	Noise detection control register_1	NSDCR_1	R/W	H'FCFFA040	16
	Burst lock/chroma decoding control register_1	BTLCR_1	R/W	H'FCFFA042	16
	Burst gate pulse control register_1	BTGPCR_1	R/W	H'FCFFA044	16
	ACC control register 1_1	ACCCR1_1	R/W	H'FCFFA046	16
	ACC control register 2_1	ACCCR2_1	R/W	H'FCFFA048	16
	ACC control register 3_1	ACCCR3_1	R/W	H'FCFFA04A	16
	TINT control register_1	TINTCR_1	R/W	H'FCFFA04C	16
	Y/C delay/chroma decoding control register_1	YDCR_1	R/W	H'FCFFA04E	16
	AGC control register 1_1	AGCCR1_1	R/W	H'FCFFA050	16
	AGC control register 2_1	AGCCR2_1	R/W	H'FCFFA052	16
	Peak limiter control register_1	PKLIMITCR_1	R/W	H'FCFFA054	16
	Over-range control register 1_1	RGORCR1_1	R/W	H'FCFFA056	16
	Over-range control register 2_1	RGORCR2_1	R/W	H'FCFFA058	16
	Over-range control register 3_1	RGORCR3_1	R/W	H'FCFFA05A	16
	Over-range control register 4_1	RGORCR4_1	R/W	H'FCFFA05C	16
	Over-range control register 5_1	RGORCR5_1	R/W	H'FCFFA05E	16
	Over-range control register 6_1	RGORCR6_1	R/W	H'FCFFA060	16
	Over-range control register 7_1	RGORCR7_1	R/W	H'FCFFA062	16
	Feedback control register for horizontal AFC phase comparator_1	AFCPFCR_1	R/W	H'FCFFA07C	16
	Register update enable register_1	RUPDCR_1	R/W	H'FCFFA07E	16
	Sync separation status/vertical cycle read register_1	VSYNCSR_1	R	H'FCFFA080	16
	Horizontal cycle read register_1	HSYNCSR_1	R	H'FCFFA082	16
	Digital clamp read register 1_1	DCPSR1_1	R	H'FCFFA084	16
	Digital clamp read register 2_1	DCPSR2_1	R	H'FCFFA086	16

**Table 30.5 Register Configuration**

Channel	Register Name	Abbr.	R/W	Address	Access Size
1	Noise detection read register_1	NSDSR_1	R	H'FCFFA08C	16
	Chroma decoding read register 1_1	CROMASR1_1	R	H'FCFFA08E	16
	Chroma decoding read register 2_1	CROMASR2_1	R	H'FCFFA090	16
	Sync separation read register_1	SYNCSSR_1	R	H'FCFFA092	16
	AGC control read register 1_1	AGCCSR1_1	R	H'FCFFA094	16
	AGC control read register 2_1	AGCCSR2_1	R	H'FCFFA096	16
	Y/C separation control register 3_1	YCSCR3_1	R/W	H'FCFFA104	16
	Y/C separation control register 4_1	YCSCR4_1	R/W	H'FCFFA106	16
	Y/C separation control register 5_1	YCSCR5_1	R/W	H'FCFFA108	16
	Y/C separation control register 6_1	YCSCR6_1	R/W	H'FCFFA10A	16
	Y/C separation control register 7_1	YCSCR7_1	R/W	H'FCFFA10C	16
	Y/C separation control register 8_1	YCSCR8_1	R/W	H'FCFFA10E	16
	Y/C separation control register 9_1	YCSCR9_1	R/W	H'FCFFA110	16
	Y/C separation control register 11_1	YCSCR11_1	R/W	H'FCFFA114	16
	Y/C separation control register 12_1	YCSCR12_1	R/W	H'FCFFA116	16
	Digital clamp control register 9_1	DCPCR9_1	R/W	H'FCFFA180	16
	Chroma filter TAP coefficient (WA_F0) register for Y/C separation_1	YCTWA_F0_1	R/W	H'FCFFA192	16
	Chroma filter TAP coefficient (WA_F1) register for Y/C separation_1	YCTWA_F1_1	R/W	H'FCFFA194	16
	Chroma filter TAP coefficient (WA_F2) register for Y/C separation_1	YCTWA_F2_1	R/W	H'FCFFA196	16
	Chroma filter TAP coefficient (WA_F3) register for Y/C separation_1	YCTWA_F3_1	R/W	H'FCFFA198	16
	Chroma filter TAP coefficient (WA_F4) register for Y/C separation_1	YCTWA_F4_1	R/W	H'FCFFA19A	16
	Chroma filter TAP coefficient (WA_F5) register for Y/C separation_1	YCTWA_F5_1	R/W	H'FCFFA19C	16
	Chroma filter TAP coefficient (WA_F6) register for Y/C separation_1	YCTWA_F6_1	R/W	H'FCFFA19E	16
	Chroma filter TAP coefficient (WA_F7) register for Y/C separation_1	YCTWA_F7_1	R/W	H'FCFFA1A0	16
	Chroma filter TAP coefficient (WA_F8) register for Y/C separation_1	YCTWA_F8_1	R/W	H'FCFFA1A2	16
	Chroma filter TAP coefficient (WB_F0) register for Y/C separation_1	YCTWB_F0_1	R/W	H'FCFFA1A4	16
	Chroma filter TAP coefficient (WB_F1) register for Y/C separation_1	YCTWB_F1_1	R/W	H'FCFFA1A6	16
	Chroma filter TAP coefficient (WB_F2) register for Y/C separation_1	YCTWB_F2_1	R/W	H'FCFFA1A8	16
	Chroma filter TAP coefficient (WB_F3) register for Y/C separation_1	YCTWB_F3_1	R/W	H'FCFFA1AA	16
	Chroma filter TAP coefficient (WB_F4) register for Y/C separation_1	YCTWB_F4_1	R/W	H'FCFFA1AC	16
	Chroma filter TAP coefficient (WB_F5) register for Y/C separation_1	YCTWB_F5_1	R/W	H'FCFFA1AE	16
	Chroma filter TAP coefficient (WB_F6) register for Y/C separation_1	YCTWB_F6_1	R/W	H'FCFFA1B0	16
	Chroma filter TAP coefficient (WB_F7) register for Y/C separation_1	YCTWB_F7_1	R/W	H'FCFFA1B2	16

**Table 30.5 Register Configuration**

Channel	Register Name	Abbr.	R/W	Address	Access Size
1	Chroma filter TAP coefficient (WB_F8) register for Y/C separation_1	YCTWB_F8_1	R/W	H'FCFFA1B4	16
	Chroma filter TAP coefficient (NA_F0) register for Y/C separation_1	YCTNA_F0_1	R/W	H'FCFFA1B6	16
	Chroma filter TAP coefficient (NA_F1) register for Y/C separation_1	YCTNA_F1_1	R/W	H'FCFFA1B8	16
	Chroma filter TAP coefficient (NA_F2) register for Y/C separation_1	YCTNA_F2_1	R/W	H'FCFFA1BA	16
	Chroma filter TAP coefficient (NA_F3) register for Y/C separation_1	YCTNA_F3_1	R/W	H'FCFFA1BC	16
	Chroma filter TAP coefficient (NA_F4) register for Y/C separation_1	YCTNA_F4_1	R/W	H'FCFFA1BE	16
	Chroma filter TAP coefficient (NA_F5) register for Y/C separation_1	YCTNA_F5_1	R/W	H'FCFFA1C0	16
	Chroma filter TAP coefficient (NA_F6) register for Y/C separation_1	YCTNA_F6_1	R/W	H'FCFFA1C2	16
	Chroma filter TAP coefficient (NA_F7) register for Y/C separation_1	YCTNA_F7_1	R/W	H'FCFFA1C4	16
	Chroma filter TAP coefficient (NA_F8) register for Y/C separation_1	YCTNA_F8_1	R/W	H'FCFFA1C6	16
	Chroma filter TAP coefficient (NB_F0) register for Y/C separation_1	YCTNB_F0_1	R/W	H'FCFFA1C8	16
	Chroma filter TAP coefficient (NB_F1) register for Y/C separation_1	YCTNB_F1_1	R/W	H'FCFFA1CA	16
	Chroma filter TAP coefficient (NB_F2) register for Y/C separation_1	YCTNB_F2_1	R/W	H'FCFFA1CC	16
	Chroma filter TAP coefficient (NB_F3) register for Y/C separation_1	YCTNB_F3_1	R/W	H'FCFFA1CE	16
	Chroma filter TAP coefficient (NB_F4) register for Y/C separation_1	YCTNB_F4_1	R/W	H'FCFFA1D0	16
	Chroma filter TAP coefficient (NB_F5) register for Y/C separation_1	YCTNB_F5_1	R/W	H'FCFFA1D2	16
	Chroma filter TAP coefficient (NB_F6) register for Y/C separation_1	YCTNB_F6_1	R/W	H'FCFFA1D4	16
	Chroma filter TAP coefficient (NB_F7) register for Y/C separation_1	YCTNB_F7_1	R/W	H'FCFFA1D6	16
	Chroma filter TAP coefficient (NB_F8) register for Y/C separation_1	YCTNB_F8_1	R/W	H'FCFFA1D8	16
	Luminance (Y) signal gain control register_1	YGAINCR_1	R/W	H'FCFFA200	16
	Color difference (Cb) signal gain control register_1	CBGAINCR_1	R/W	H'FCFFA202	16
	Color difference (Cr) signal gain control register_1	CRGAINCR_1	R/W	H'FCFFA204	16
	PGA register update_1	PGA_UPDATE_1	R/W	H'FCFFA280	16
	PGA control register_1	PGACR_1	R/W	H'FCFFA282	16
	ADC control register 2_1	ADCCR2_1	R/W	H'FCFFA284	16

### 30.4.1 ADC Control Register 1 (ADCCR1)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	AGC MODE	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	AGCMODE	0	R/W	A/D Converter AGC ON/OFF Control 0: AGC OFF 1: AGC ON
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

#### (1) AGC Control

The AGCMODE bit controls the AGC ON/OFF. When AGCMODE is 1, the AGC operation is performed by detecting the sync amplitude and video peak amplitude and controlling the PGA gain of the ADC.

When PGACR.PGA\_GAIN\_SEL is 1, the PGA gain can be directly controlled with the PGACR.PGA\_GAIN value. At this time, the AGCMODE setting is invalid.

Setting AGCMODE to 0 and PGACR.PGA\_GAIN\_SEL to 0 simultaneously is prohibited.

### 30.4.2 Timing Generation Control Register 1 (TGCR1)

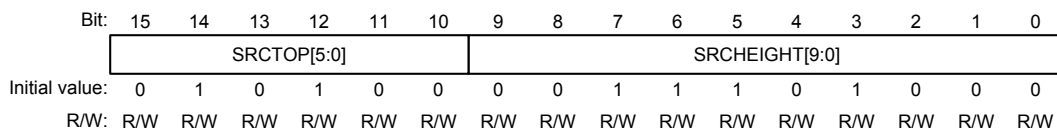
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
	—	—	—	—	—	—	—	SRCLEFT[8:0]									—	—
Initial value:	0	0	0	0	0	0	0	1	0	0	1	1	1	1	0	0		
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W		

Bit	Bit Name	Initial Value	R/W	Description
15 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8 to 0	SRCLEFT[8:0]	H'13C	R/W	Left End of Input Video Signal Capturing Area Set the position from the horizontal sync reference in 27-MHz clock cycle units.

#### (1) Timing Generation (Horizontal Start Position) Control

SRCLEFT sets the start position of the horizontal enable signal of the output video signal from the horizontal sync reference in 27-MHz clock cycle units.

### 30.4.3 Timing Generation Control Register 2 (TGCR2)



Bit	Bit Name	Initial Value	R/W	Description
15 to 10	SRCTOP[5:0]	H'14	R/W	Top End of Input Video Signal Capturing Area Set the position from the vertical sync reference in one-line units.
9 to 0	SRCHEIGHT [9:0]	H'0E8	R/W	Height of Input Video Signal Capturing Area Set the vertical active period in one-line units.

Note: • All the bits in this register are updated when the vertical sync signal is asserted with the NEWSETTING bit in RUPDCR being 1.

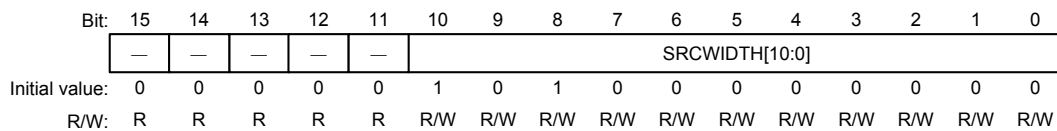
#### (1) Timing Generation (Vertical Start Position) Control

SRCTOP sets the start position of the vertical enable signal of the output video signal from the vertical sync reference in one-line units.

#### (2) Timing Generation (Vertical Width) Control

SRCHEIGHT sets the height of the vertical enable signal of the output video signal in one-line units.

### 30.4.4 Timing Generation Control Register 3 (TGCR3)



Bit	Bit Name	Initial Value	R/W	Description
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SRCWIDTH [10:0]	H'500	R/W	Width of Input Video Signal Capturing Area Set the horizontal active period in 27-MHz clock cycle units.

Note: • All the bits in this register are updated when the vertical sync signal is asserted with the NEWSETTING bit in RUPDCR being 1.

#### (1) Timing Generation (Horizontal Width) Control

SRCWIDTH sets the width of horizontal enable signal of the output video signal in 27-MHz clock cycle units.

Figure 30.3 to Figure 30.7 show the timings generated with the NTSC (59.94 Hz) and PAL/SECAM (50.00 Hz) formats.

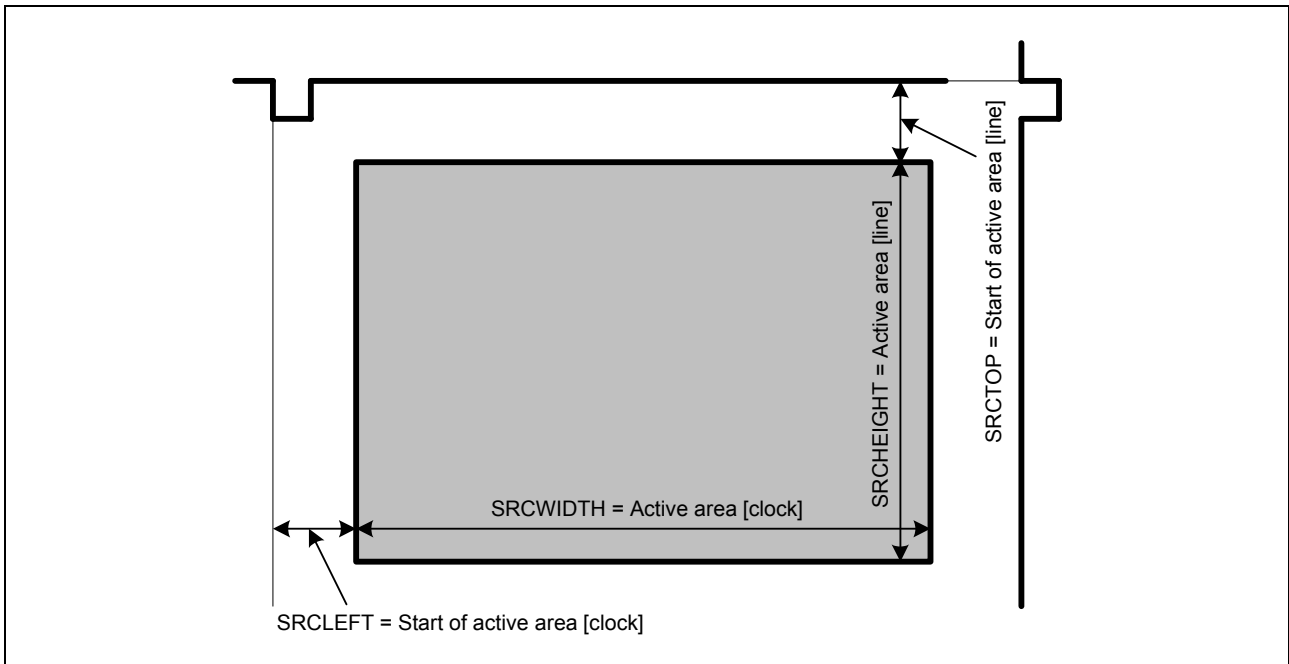


Figure 30.3 Active Image Area Setting

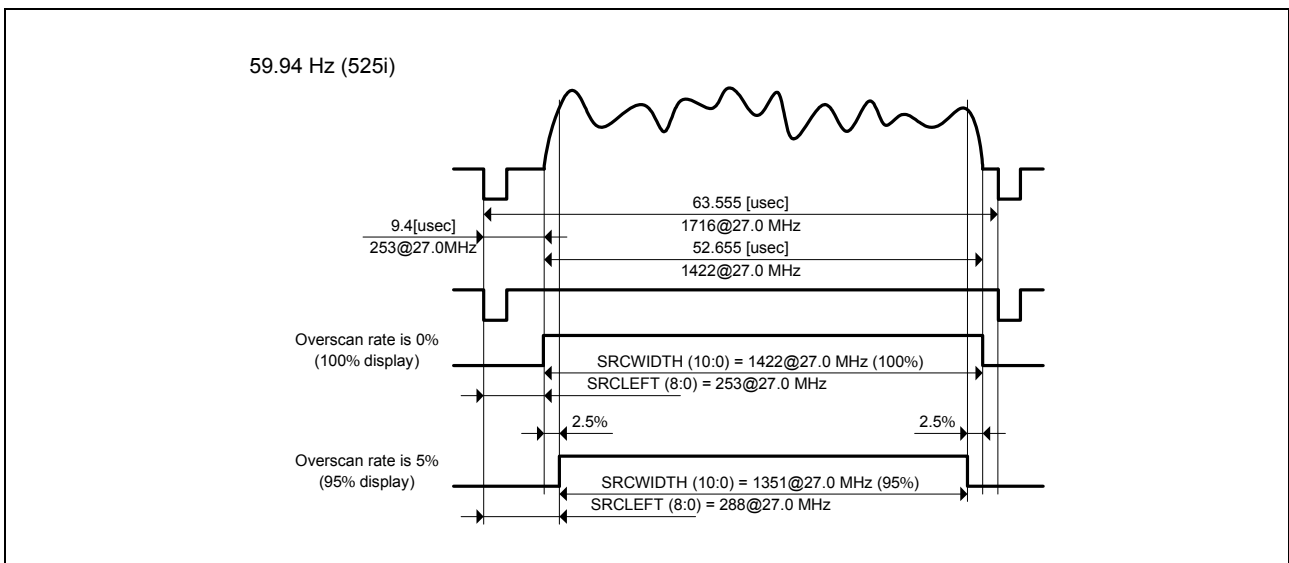


Figure 30.4 Example of Horizontal Active Image Period (59.94 Hz (525i))



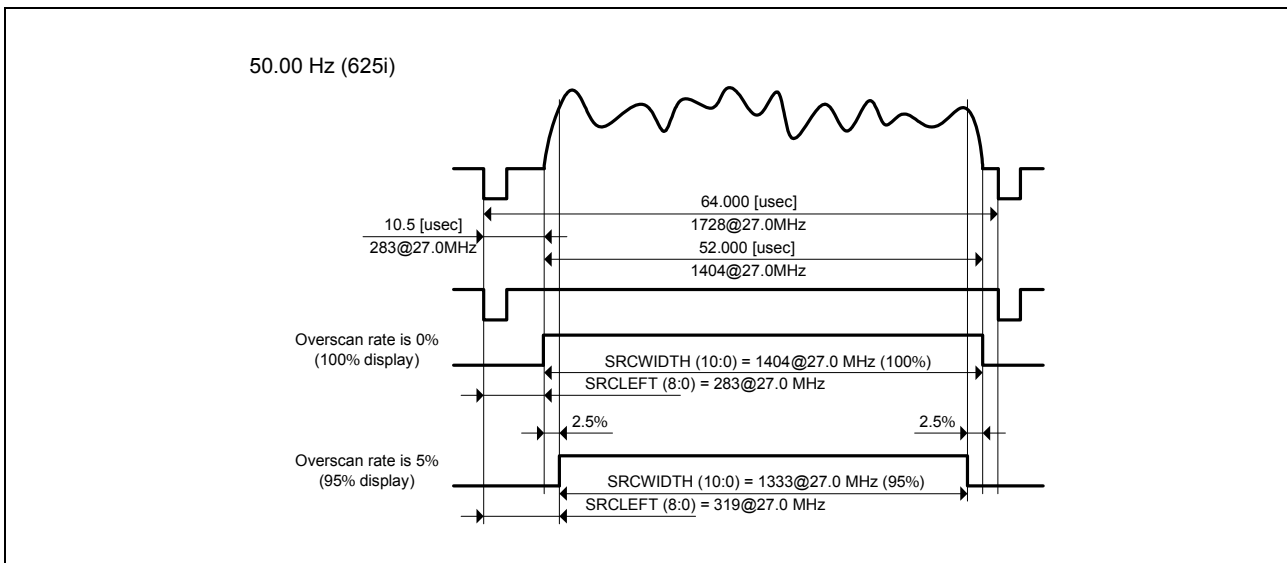


Figure 30.5 Example of Horizontal Active Image Period (50.00 Hz (625i))

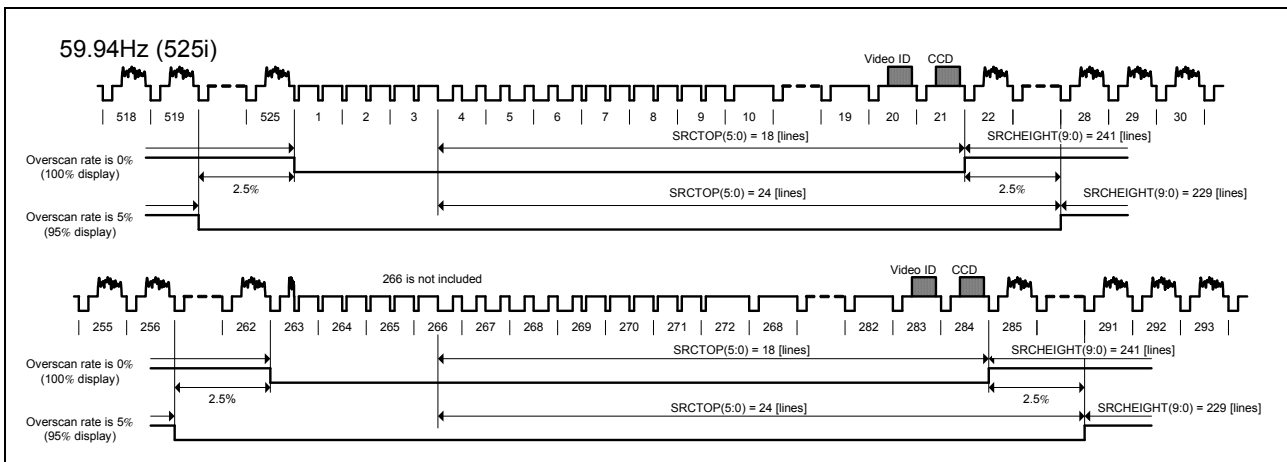


Figure 30.6 Example of Vertical Active Image Period (59.94 Hz (525i))

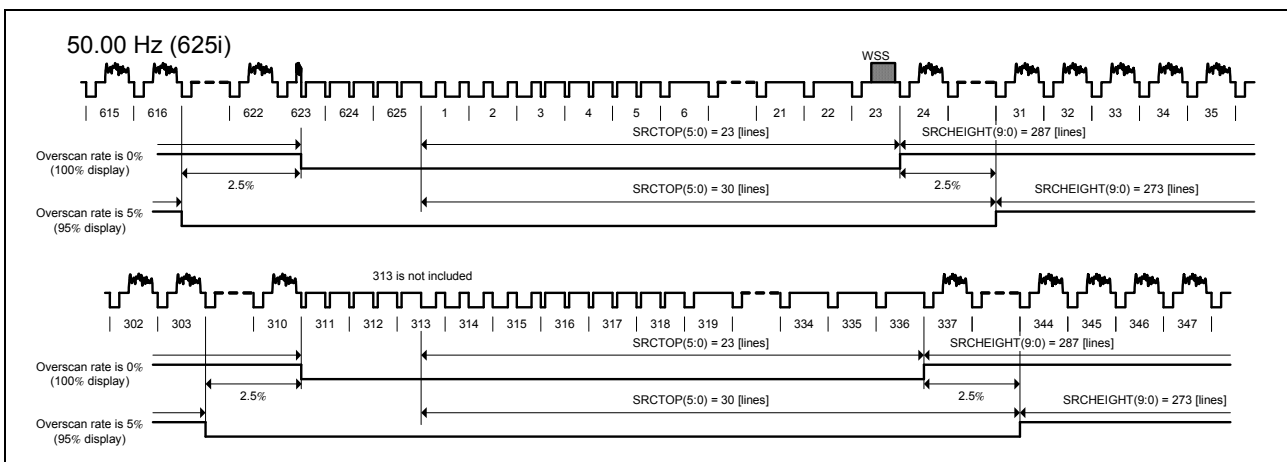


Figure 30.7 Example of Vertical Active Image Period (50.00 Hz (625i))

The active period width should not be larger than necessary.

The settings of TGCR1 to TGCR3 such as valid period setting for the peak limiter are applied only to this module. To set the display size of the input video, SCL0\_DS2 and SCL0\_DS3 of the video display controller 5 scaler should be used.

## 30.4.5 Sync Separation Control Register 1 (SYNSCR1)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LPFVSYNC[2:0]			LPFHSYNC[2:0]			—	—	VELOCITYSHIFT_H[3:0]			SLICER MODE_H[1:0]		SLICER MODE_V[1:0]		
Initial value:	0	1	1	0	1	1	0	0	0	0	0	0	1	0	1	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

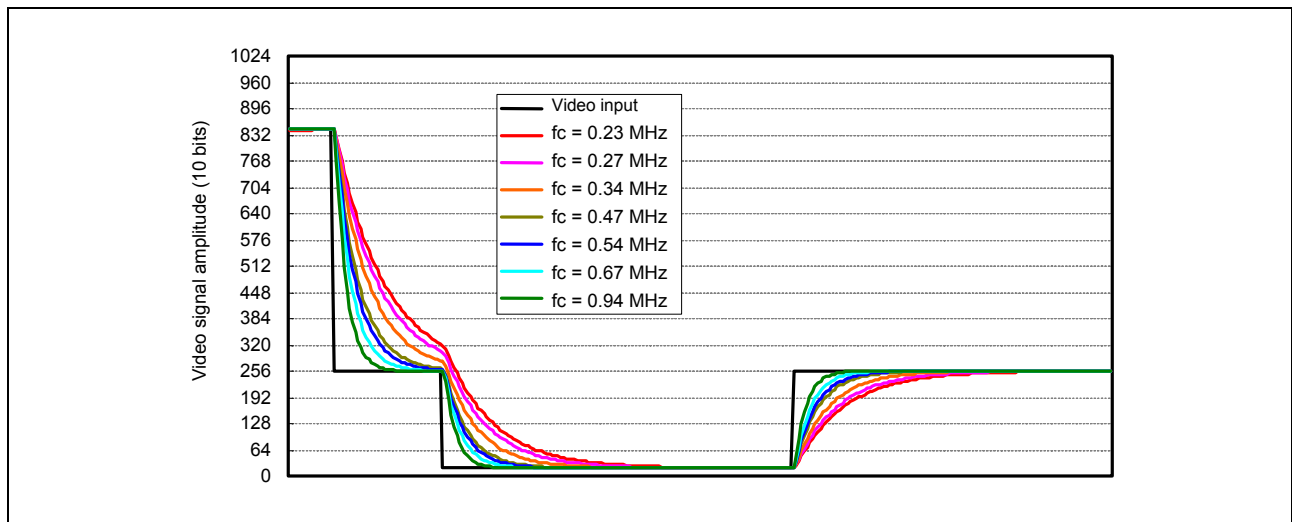
Bit	Bit Name	Initial Value	R/W	Description
15 to 13	LPFVSYNC [2:0]	011	R/W	Low-Pass Filter Cutoff Frequency before Vertical Sync Separation 0: None 1: 0.94 MHz 2: 0.67 MHz 3: 0.54 MHz 4: 0.47 MHz 5: 0.34 MHz 6: 0.27 MHz 7: 0.23 MHz
12 to 10	LPFHSYNC [2:0]	011	R/W	Low-Pass Filter Cutoff Frequency before Horizontal Sync Separation 0: None 1: 2.15 MHz 2: 1.88 MHz 3: 1.34 MHz 4: 1.07 MHz 5: 0.94 MHz 6: 0.67 MHz 7: 0.54 MHz
9, 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 4	VELOCITY SHIFT_H[3:0]	0000	R/W	Reference Level Operation Speed Control for Composite Sync Separation (for horizontal sync signal) 0: ×1 1: ×2 2: ×4 3: ×8 4: ×16 5: ×32 6: ×64 7: ×128 Others: ×256 Standard speed (×1) ⇔ High speed (×256)
3, 2	SLICERMODE _H[1:0]	10	R/W	Auto-Slice Level Setting for Composite Sync Separation Circuit (for horizontal sync signal) 0: Manual setting by CSYNCSLICE_H 1: 25% of sync depth (Auto) 2: 50% of sync depth (Auto) 3: 75% of sync depth (Auto)
1, 0	SLICERMODE _V[1:0]	10	R/W	Auto-Slice Level Setting for Composite Sync Separation Circuit (for vertical sync signal) 0: Manual setting by CSYNCSLICE_V 1: 25% of sync depth (Auto) 2: 50% of sync depth (Auto) 3: 75% of sync depth (Auto)

## (1) Low-Pass Filter Control before Vertical Sync Separation

LPFVSYNC sets the noise reduction low-pass filter for the input video signal fed to a sync separator in order to avoid sync separation error caused by noise. A low-pass filter cutoff frequency should be set not to deteriorate (i.e. to enable to detect) the composite sync signal components.

**Table 30.6 Low-Pass Filter Cutoff Frequency before Vertical Sync Separation**

For Vertical Sync Separation		
LPFVSYNC[2:0]	t	fc (MHz)
1	0.109375	0.939647766
2	0.078125	0.671176976
3	0.0625	0.536941581
4	0.0546875	0.469823883
5	0.0390625	0.335588488
6	0.003125	0.26847079
7	0.0273438	0.234911942



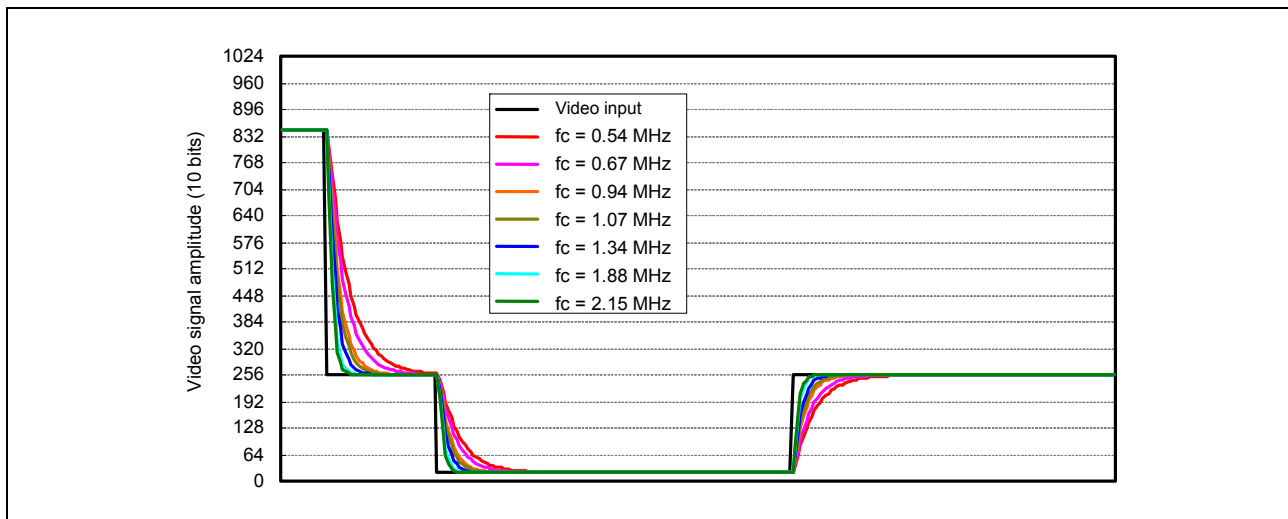
**Figure 30.8 Low-Pass Filter Output Waveform near Vertical Sync Signal during 100% White Signal Input (Vertical: Pattern Diagram)**

## (2) Low-Pass Filter Control before Horizontal Sync Separation

LPFHSYNC sets the noise reduction low-pass filter for the input video signal fed to a sync separator in order to avoid sync separation error caused by noise. A low-pass filter cutoff frequency should be set not to deteriorate (i.e. to enable to detect) the composite sync signal components.

**Table 30.7 Low-Pass Filter Cutoff Frequency before Horizontal Sync Separation**

For Horizontal Sync Separation		
LPFHSYNC[2:0]	t	fc (MHz)
1	0.25	2.147766323
2	0.21875	1.879295533
3	0.15625	1.342353952
4	0.125	1.073883162
5	0.10938	0.939647766
6	0.07813	0.671176976
7	0.0625	0.536941581



**Figure 30.9 Low-Pass Filter Output Waveform near Horizontal Sync Signal during 100% White Signal Input (Horizontal: Pattern Diagram)**

## (3) Reference Level Operation Speed Control for Sync Separation

VELOCITYSHIFT\_H controls the speed for automatically determining the slice level.

If sync skew is caused by sync sag, it can be improved by raising the determination speed using VELOCITYSHIFT\_H. The standard setting is 0.

## (4) Horizontal Sync Slicer Control

SLICERMODE\_H controls composite sync signal separation from the video signals.

The slice level for composite sync signal separation can be set either manually or automatically. When automatic setting is used, the level is automatically set using the sync signal amplitude detection result, which is described later. The sync slicer can be controlled separately for horizontal and vertical sync signals.

(5) Vertical Sync Slicer Control

SLICERMODE\_V controls composite sync signal separation from the video signals.

The slice level for composite sync signal separation can be set either manually or automatically. When automatic setting is used, the level is automatically set using the sync signal amplitude detection result, which is described later. The sync slicer can be controlled separately for horizontal and vertical sync signals.

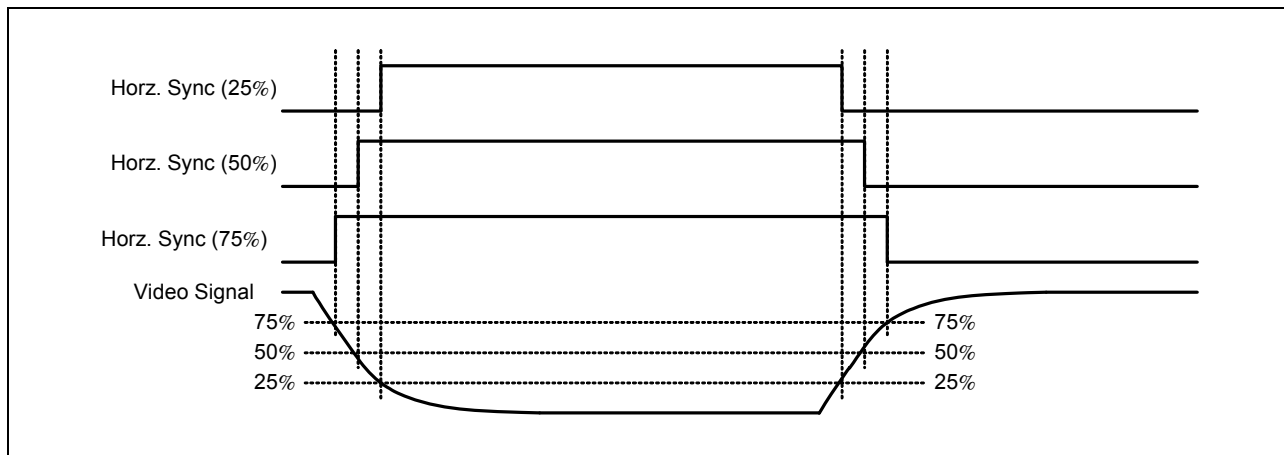


Figure 30.10 Auto Slice Level Setting

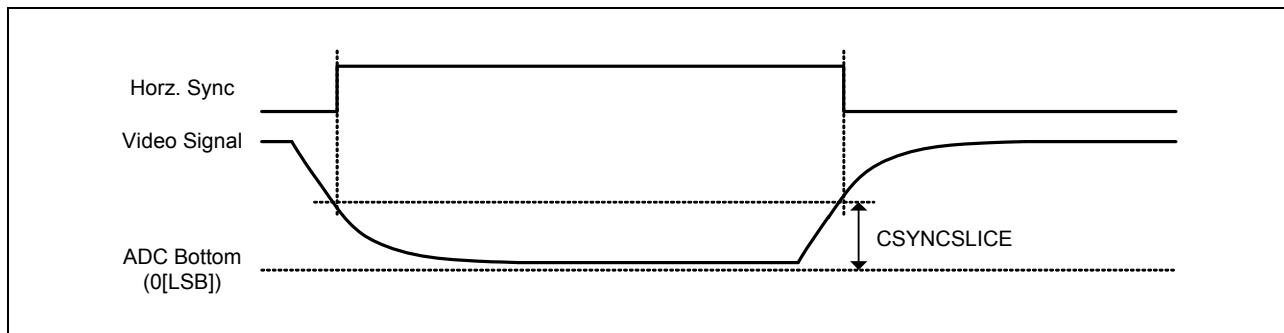


Figure 30.11 Manual Slice Level Setting

## 30.4.6 Sync Separation Control Register 2 (SYNSCR2)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	SYNCMAXDUTY_H[5:0]					SYNCMINDUTY_H[5:0]						
Initial value:	0	0	0	0	0	0	1	1	1	1	0	0	1	0	1	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 6	SYNCMAX DUTY_H[5:0]	001111	R/W	Max Ratio of Horizontal Cycle to Horizontal Sync Signal Pulse Width Valid when auto slice level setting is active (SLICERMODE_H ≠ 0).
5 to 0	SYNCMIN DUTY_H[5:0]	001010	R/W	Min Ratio of Horizontal Cycle to Horizontal Sync Signal Pulse Width Valid when auto slice level setting is active (SLICERMODE_H ≠ 0).

## (1) Sync Amplitude Detection Control for Horizontal Sync Separation

SYNCMAXDUTY\_H and SYNCMINDUTY\_H control the sync signal amplitude detection of composite sync signal included in the video signal.

Table 30.8 Auto Slice Level Register Settings for Composite Sync Separation

	Horizontal Period (μsec)	Horizontal Sync Width (μsec)	Video Active Period (μsec)	Horizontal Blanking Interval (μsec)	SYNCMAXDUTY_H [5:0] Recommended Value	SYNCMINDUTY_H [5:0] Recommended Value
525i/59.94 Hz	63.56	4.70	52.66	10.90	15	10
625i/50 Hz	64.00	4.70	52.00	12.00	15	10

### 30.4.7 Sync Separation Control Register 3 (SYNSCR3)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	SSCLIPSEL[3:0]				CSYNCSLICE_H[9:0]									
Initial value:	0	0	1	1	1	1	0	0	1	0	0	1	0	0	1	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13 to 10	SSCLIPSEL [3:0]	1111	R/W	Clipping Level Clip the video signal supplied to the vertical/horizontal sync separation low-pass filter Bit value = Clipping level (amplitude 50% to no clipping) 0: 512    1: 546 2: 580    3: 614 4: 648    5: 682 6: 716    7: 750 8: 785    9: 819 10: 853    11: 887 12: 921    13: 955 14: 989    15: 1023
9 to 0	CSYNCSLICE_ H[9:0]	0010010010	R/W	Slice Level for Composite Sync Signal Separation (for horizontal sync signal) Valid when manual slice level setting is active (SLICERMODE_H = 0). Setting range: 0 to 1023

#### (1) Video Signal Clipping Setting for Sync Separation

For input video signals supplied to the sync separator circuit, the level to clip the high tone component of the video signal is specified to reduce amplitude-dependency of the video signal. The video clipping level should be set not to deteriorate (i.e. to enable to detect) the composite sync signal components.

#### (2) Slice Level Setting for Horizontal Sync Separation

CSYNCSLICE\_H sets the slice level for sync separation. This bit is valid only when SLICERMODE\_H = 0.

## 30.4.8 Sync Separation Control Register 4 (SYNSCR4)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	SYNCMAXDUTY_V[5:0]					SYNCMINDUTY_V[5:0]						
Initial value:	0	0	0	0	0	0	1	1	1	1	0	0	1	0	1	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 6	SYNCMAX DUTY_V[5:0]	001111	R/W	Max Ratio of Horizontal Cycle to Vertical Sync Signal Pulse Width Valid when auto slice level setting is active (SLICERMODE_V ≠ 0).
5 to 0	SYNCMIN DUTY_V[5:0]	001010	R/W	Min Ratio of Horizontal Cycle to Horizontal Sync Signal Pulse Width Valid when auto slice level setting is active (SLICERMODE_V ≠ 0).

## (1) Sync Amplitude Detection Control for Vertical Sync Separation

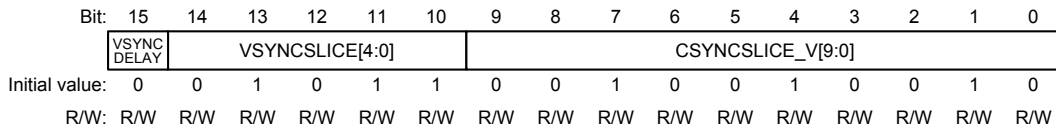
SYNCMAXDUTY\_V and SYNCMINDUTY\_V control the sync signal amplitude detection of composite sync signal included in the video signal.

**Table 30.9 Auto Slice Level Register Settings for Composite Sync Separation**

	Horizontal Period (μsec)	Horizontal Sync Width (μsec)	Video Active Period (μsec)	Horizontal Blanking Interval (μsec)	SYNCMINDUTY_V [5:0] Recommended Value	SYNCMAXDUTY_V [5:0] Recommended Value
525i/59.94 Hz	63.56	4.70	52.66	10.90	15	9
625i/50 Hz	64.00	4.70	52.00	12.00	15	9



### 30.4.9 Sync Separation Control Register 5 (SYNSCR5)



Bit	Bit Name	Initial Value	R/W	Description
15	VSYNCDELAY	0	R/W	Delays the separated vertical sync signal for 1/4 horizontal cycle. 0: Disable 1/4fH delay 1: Enable 1/4fH delay Note: • Stability of the field determination result may be improved by changing VSYNCDELAY.
14 to 10	VSYNC SLICE [4:0]	01011	R/W	Threshold for Vertical Sync Separation The greater the value, the wider pulse width is needed.
9 to 0	CSYNC SLICE_V [9:0]	0010010010	R/W	Slice Level for Composite Sync Signal Separation (for vertical sync signal) Valid when manual slice level setting is active (SLICERMODE_V = 0). Setting range: 0 to 1023

#### (1) Vertical Sync Separation Control

VSYNCDELAY controls the phases of vertical sync signal and horizontal sync signal. When VSYNCDELAY is 1, the stability of the field determination result may be improved by delaying the vertical sync signal for 1/4 fH.

#### (2) Vertical Sync Separation Control

VSYNC SLICE controls the threshold for separating vertical sync signal from composite sync signal. The value will be set depending on the serration pulse width of each video signal format. Table 30.10 shows the recommended set values.

**Table 30.10 Recommended Threshold and Serration Pulse Width (for reference)**

Serration Pulse Width [ $\mu$ sec]	VSYNC SLICE[4:0]	
525i/59.94Hz	27.08	15
625i/50Hz	27.30	15

#### (3) Slice Level Setting for Vertical Sync Separation

CSYNC SLICE\_V sets the slice level for sync separation. This bit is valid only when SLICERMODE\_V = 0.

### 30.4.10 Horizontal AFC Control Register 1 (HAFCCR1)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HAFCGAIN[3:0]				—	HAFCFREERUN	HAFCTYP[9:0]									
Initial value:	0	1	1	0	0	0	1	0	1	0	1	1	0	1	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	HAFCGAIN[3:0]	0110	R/W	Horizontal AFC Loop Gain 0 to 5: A smaller value needs a longer time for synchronization. 6: Standard value 7 to 15: A larger value needs a shorter time for synchronization.
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	HAFCFREERUN	0	R/W	Horizontal AFC Free-Run Oscillation Mode ON/OFF 0: OFF 1: ON
9 to 0	HAFCTYP[9:0]	1010110100	R/W	Horizontal AFC Center Oscillation Frequency Set an offset from 1024th clock pulse in 27-MHz clock cycle units.

#### (1) Horizontal AFC Loop Gain Control

HAFCGAIN sets the loop gain (response speed) of the horizontal AFC. The larger the value is, the faster the response speed is. However, setting a larger value will result in higher susceptibility to noise. The typical value is 6.

#### (2) Horizontal AFC Free-Run Control

HAFCFREERUN controls the horizontal AFC free-run operation. When HAFCFREERUN is 1, the horizontal AFC operates independently of the inputs and performs free-run operation. HAFCFREERUN should usually be set to 0.

#### (3) Horizontal AFC Lock Range (Horizontal) Control

HAFCMIN, HAFCTYP, and HAFCMAX set the horizontal AFC center oscillation frequency and lock range. The horizontal AFC function is controlled to stabilize the horizontal sync signal when the signals are deteriorated by trick playback of VCR or a weak electric field.

HAFCMIN[9:0]	Min oscillation frequency of horizontal AFC
HAFCTYP[9:0]	Center oscillation frequency of horizontal AFC
HAFCMAX[9:0]	Max oscillation frequency of horizontal AFC

Horizontal AFC lock range can be indicated by the following formula.

$$\text{HAFCMIN} < \text{HAFCTYP} < \text{HAFCMAX} \dots (1)$$

where

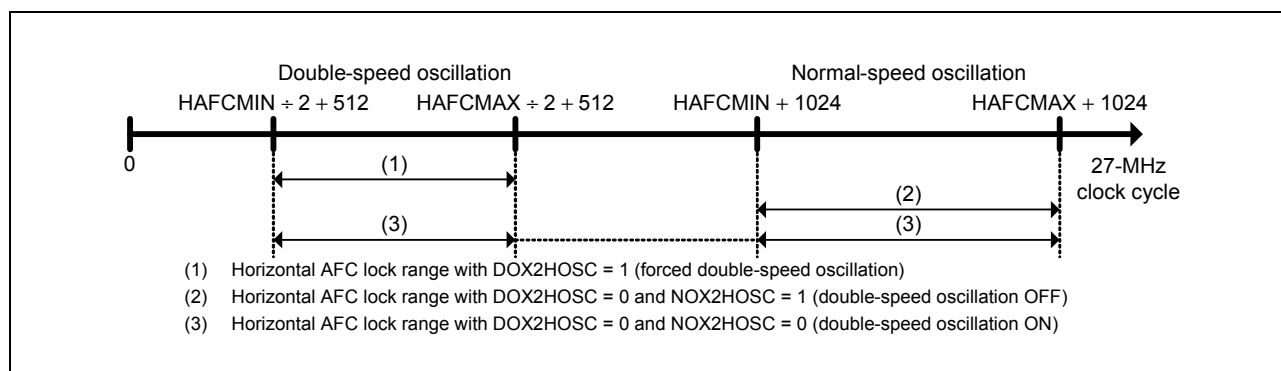
HAFCMIN	= HAFCTYP – allowable deviation
HAFCTYP	= $N \times M - 1024$
HAFCMAX	= HAFCTYP + allowable deviation
M:	Number of clock pulses per horizontal cycle (27-MHz sampling)
N:	Double speed setting 2 (double speed): $M < 1024$ 1 (normal speed): $M \geq 1024$

The horizontal AFC is locked if the formula indicated by (1) is satisfied.

When the horizontal AFC is locked, FHLOCK in VSYNCSR is set to 1. Otherwise, FHLOCK is 0.

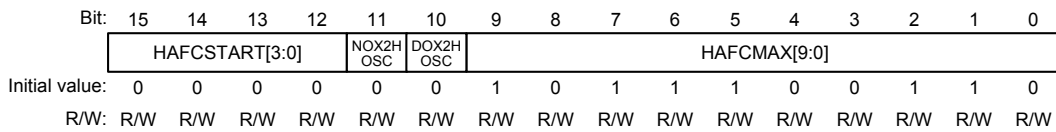
**Table 30.11 Horizontal AFC Lock Range Setting**

Signal Format	fH Horizontal Cycle	M fH@ 27.0MHz	N-Times Speed Setting	HAFCMAX [9:0]	HAFCTYP [9:0]	HAFCMIN [9:0]	Deviation	Unit
525i	63.56 [μsec]	1716 [clk]	1	771	692	618	79	[clk]
				15.034	15.734	16.434	-0.700	0.700
625i	64.00 [μsec]	1728 [clk]	1	785	704	629	81	[clk]
				14.925	15.625	16.325	-0.700	0.700



**Figure 30.12 Horizontal AFC Lock Range (Horizontal)**

### 30.4.11 Horizontal AFC Control Register 2 (HAFCCR2)



Bit	Bit Name	Initial Value	R/W	Description
15 to 12	H AFCSTART [3:0]	0000	R/W	Start Line of Horizontal AFC Normal Operation (=VBI process end line) Start the phase comparison at the Nth line after the vertical sync signal.
11	NOX2HOSC	0	R/W	Disable of Horizontal AFC Double Speed Detection 0: Auto control 1: Double speed oscillation disabled
10	DOX2HOSC	0	R/W	Horizontal AFC Forced Double-Speed Oscillation 0: Auto control 1: Forced double-speed oscillation
9 to 0	H AFCMAX[9:0]	1011100110	R/W	Maximum Oscillation Frequency of Horizontal AFC Set an offset from 1024th clock pulse in 27-MHz clock cycle units.

#### (1) Horizontal AFC Lock Range (Vertical) Control

H AFCSTART and H AFCEND specify the horizontal AFC operation range. The horizontal AFC operation should be normally stopped from several lines before the vertical sync signal to the vertical sync signal to avoid a malfunction occurring in the VCR head switch part.

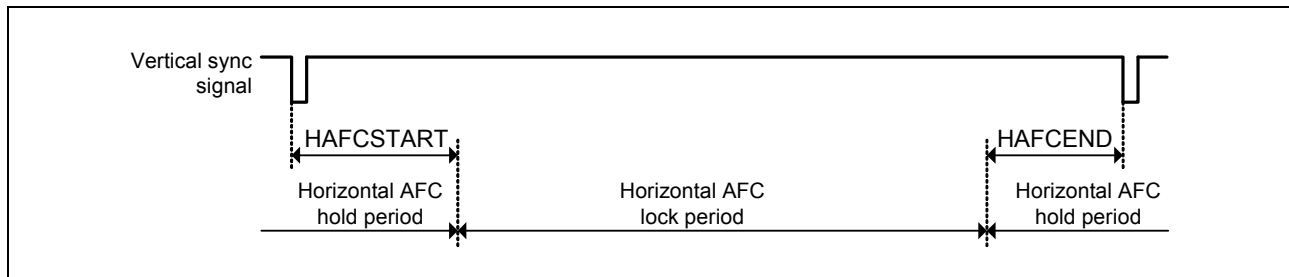
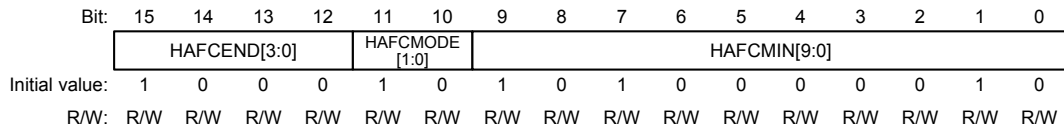


Figure 30.13 Horizontal AFC Lock Range (Vertical)

#### (2) Horizontal AFC Double-Speed Control

NOX2HOSC and DOX2HOSC control the horizontal AFC double speed detection. In the NTSC, PAL and SECAM formats, DOX2HOSC should always be 0.

### 30.4.12 Horizontal AFC Control Register 3 (HAFCCR3)



Bit	Bit Name	Initial Value	R/W	Description
15 to 12	HAFCCEND[3:0]	1000	R/W	End line of Horizontal AFC Normal Operation (=VBI process start line) Stop the phase comparison at the Nth line before the vertical sync signal.
11, 10	HAFCCMODE [1:0]	10	R/W	Horizontal AFC VBI Period Operating Mode [1] Loop gain control for low S/N 0: Loop gain is fixed. 1: Loop gain is automatically controlled. [0] Horizontal AFC control during VBI period 0: Phase comparison is stopped during VBI period. 1: Loop gain is reduced during VBI period.
9 to 0	HAFCCMIN[9:0]	1010000010	R/W	Min Oscillation Frequency of Horizontal AFC Set an offset from 1024th clock pulse in 27-MHz clock cycle units.

#### (1) Horizontal AFC Operation Control during VBI Period

The malfunction caused by noise can be avoided by setting HAFCCMODE[1] to 1 to reduce the loop gain for low S/N (VSYNC.SR.ISNOISY = 1). The recommended value for HAFCCMODE[1] is 1.

HAFCCMODE[0] controls the horizontal AFC operation during the VBI period. The recommended value for HAFCCMODE[0] is 0.

## 30.4.13 Vertical Countdown Control Register 1 (VCDWCR1)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VCDFREERUN	NOVCD50	NOVCD60	VCDDEFAULT[1:0]	VCDWINDOW[5:0]					VCDOFFSET[4:0]						
Initial value:	0	0	0	0	0	0	1	0	1	0	0	0	1	0	1	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	VCDFREERUN	0	R/W	Vertical Countdown Free-Run Oscillation Mode ON/OFF 0: OFF 1: ON
14	NOVCD50	0	R/W	Vertical Countdown 50-Hz Oscillation Mode OFF 0: 50-Hz oscillation ON 1: 50-Hz oscillation OFF
13	NOVCD60	0	R/W	Vertical Countdown 60-Hz (59.94-Hz) Oscillation Mode OFF 0: 60-Hz oscillation ON 1: 60-Hz oscillation OFF
12, 11	VCDDEFAULT[1:0]	00	R/W	Vertical Countdown Center Oscillation Frequency 0: Auto-detection 1: 50.00 Hz 2: 59.94 Hz 3: 60.00 Hz
10 to 5	VCDWINDOW[5:0]	010100	R/W	Vertical Countdown Sync Area Set a value in 0.1-ms units.
4 to 0	VCDOFFSET[4:0]	01010	R/W	Vertical Countdown Minimum Oscillation Frequency Set the shift from the center frequency in 0.1-ms units.

## (1) Vertical Countdown Free-Run Operation Control

VCDFREERUN controls the vertical countdown free-run operation. When VCDFREERUN is 1, the vertical countdown free-run operation is performed independently of the inputs. VCDFREERUN should usually be set to 0.

## (2) Vertical Countdown 50-Hz Oscillation Control

NOVD50 controls 50-Hz oscillation. When NOVCD50 is 1, the vertical countdown operation is not locked to 50 Hz.

## (3) Vertical Countdown 60-Hz Oscillation Control

NOVD60 controls 60-Hz oscillation. When NOVCD60 is 1, the vertical countdown operation is not locked to 60 Hz.

## (4) Vertical Countdown Center Oscillation Frequency Control

VCDDEFAULT sets the center frequency for the vertical countdown.

**Table 30.12 Vertical Countdown Operating Modes**

VCDDEFAULT[1:0]	Operating Mode
0	Auto-detection
1	50.00 Hz
2	59.94 Hz
3	60.00 Hz

## (5) Vertical Countdown Lock Range Control

VCDWINDOW and VCDOFFSET control the vertical countdown lock range. Figure 30.14 shows the bit settings and lock ranges.

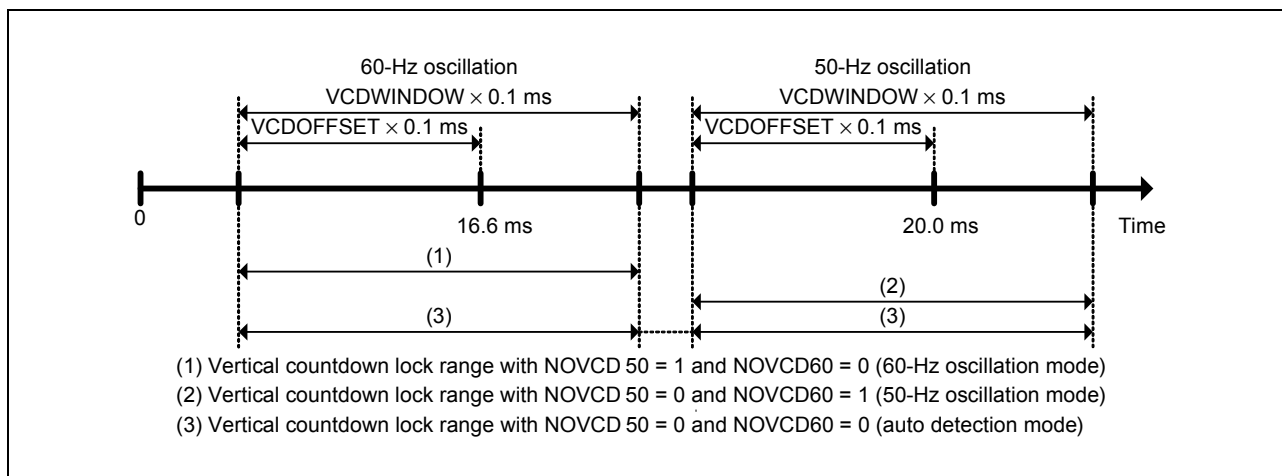


Figure 30.14 Vertical Countdown Lock Ranges

## 30.4.14 Digital Clamp Control Register 1 (DCPCR1)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCPMODE_Y	—	—	—	DCPCHECK	—	BLANKLEVEL_Y[9:0]									
Initial value:	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	DCPMODE_Y	1	R/W	Clamp Level Setting Mode (Y signal) 0: Manual clamp level setting 1: Auto clamp level setting
14 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11	DCPCHECK	0	R/W	Digital Clamp Pulse Position Check The offset given by BLANKLEVEL is added to the clamp position.
10	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
9 to 0	BLANKLEVEL_Y[9:0]	0000000000	R/W	Clamp Offset Level (Y signal) Set the subtraction value. Set a value in 1-LSB units. 2s complement

## (1) Y-Signal Clamp Operation Control

DCPMODE\_Y controls the clamp level of Y signal.

When DCPMODE\_Y = 0, the value set by BLANKLEVEL\_Y is subtracted from the video signal.

$$Y \text{ signal output} = Y \text{ signal input} - \text{BLANKLEVEL\_Y}$$

When DCPMODE\_Y is 1, the video signal level at the digital clamp pulse position (pedestal level) and BLANKLEVEL\_Y are added together and the resulting value is subtracted from the video signal.

$$Y \text{ signal output} = Y \text{ signal input} - (\text{detected value} + \text{BLANKLEVEL\_Y})$$

## (2) Digital Clamp Pulse Position Check Control

DCPCHECK allows the digital clamp pulse position to be displayed and checked on the screen. The following shows the steps to check the position.

1. Set DCPCHECK (digital clamp pulse position check bit) to 1.
2. Set SRCLEFT (left end of input video signal capturing area bit) and RES\_HS[10:0] in SCL0\_DS3 of the video display controller 5 scaler to 0.
3. Set clamp offset level of the signal to be monitored to Min value (−512 for BLANKLEVEL\_Y, (−32 for BLANKLEVEL\_Cb/Cr).
4. Adjust the pulse position and width using DCPPOS\_Y (or DCPPOS\_C) and DCPWIDTH.



## 30.4.15 Digital Clamp Control Register 2 (DCPCR2)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCPMODE_C	—	—	—	BLANKLEVEL_CB[5:0]					BLANKLEVEL_CR[5:0]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	DCPMODE_C	0	R/W	Clamp Level Setting Mode (Cb/Cr signal) 0: Manual clamp level setting 1: Auto clamp level setting
14 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 6	BLANKLEVEL_CB[5:0]	000000	R/W	Clamp Offset Level (Cb signal) Set the subtraction value. Set a value in 1-LSB units. 2s complement
5 to 0	BLANKLEVEL_CR[5:0]	000000	R/W	Clamp Offset Level (Cr signal) Set the subtraction value. Set a value in 1-LSB units. 2s complement

## (1) Cb/Cr-Signal Clamp Operation Control

DCPMODE\_C controls the clamp level of Cb/Cr-signal.

When DCPMODE\_C = 0, the value set by BLANKLEVEL\_CB/BLANKLEVEL\_CR is subtracted from the video signal.

Cb signal output = Cb signal input – BLANKLEVEL\_CB

Cr signal output = Cr signal input – BLANKLEVEL\_CR

When DCPMODE\_C = 1, sum of the video signal level (center level) in the digital clamp pulse position and BLANKLEVEL\_CB/CR is subtracted from the video signal.

Cb signal output = Cb signal input – (detected value + BLANKLEVEL\_CB)

Cr signal output = Cr signal input – (detected value + BLANKLEVEL\_CR)

### 30.4.16 Digital Clamp Control Register 3 (DCPCR3)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	DCPRESPONSE[2:0]			—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14 to 12	DCP RESPONSE [2:0]	101	R/W	Digital Clamp Response Speed The larger the value is, the faster the response speed is. However, that will result in higher susceptibility to noise.
11 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

#### (1) Digital Clamp Response Speed

DCPRESPONSE sets the digital clamp response speed.

Though the larger value makes the response faster, that will result in higher susceptibility to noise.

DCPRESPONSE is used in common to Y, Cb, and Cr signals.

### 30.4.17 Digital Clamp Control Register 4 (DCPCR4)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCPSTART[5:0]						—	—	—	—	—	—	—	—	—	—
Initial value:	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	DCPSTART [5:0]	010000	R/W	Digital Clamp Start Line (in 1-line units) Start clamp pulses at the Nth line after the vertical sync signal.
9 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

## 30.4.18 Digital Clamp Control Register 5 (DCPCR5)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCPEND[5:0]						—	—	—	—	—	—	—	—	—	—
Initial value:	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	DCPEND[5:0]	010000	R/W	Digital Clamp End Line (in 1-line units) Stop clamp pulses at the Nth line before the vertical sync signal.
9 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

## (1) Digital Clamp Pulse Control (Vertical)

DCPSTART and DCPEND control the digital clamp pulses in the vertical direction. Figure 30.15 shows the bit settings and digital clamp timing.

DCPSTART and DCPEND are used in common to Y, Cb, and Cr signals.

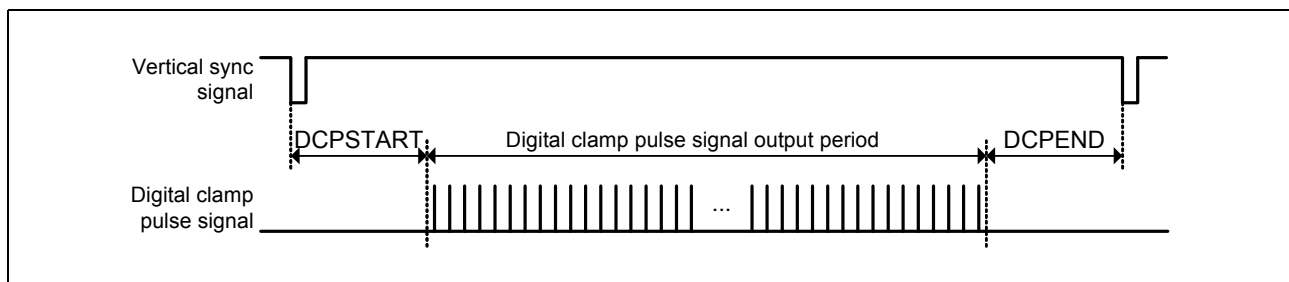


Figure 30.15 Digital Clamp Timing (Vertical)

## 30.4.19 Digital Clamp Control Register 6 (DCPCR6)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	DCPWIDTH[6:0]						—	—	—	—	—	—	—	—	—
Initial value:	0	0	1	1	0	1	1	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14 to 8	DCPWIDTH [6:0]	0110110	R/W	Digital Clamp Pulse Width Setting range: 0 to 127 Set a value in 27-MHz clock cycle units.
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

## 30.4.20 Digital Clamp Control Register 7 (DCPCR7)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCPPOS_Y[7:0]								—	—	—	—	—	—	—	—
Initial value:	1	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	DCPPOS_Y [7:0]	10100010	R/W	Digital Clamp Pulse Horizontal Start Position (Y signal) Setting range: 0 to 255 Set a value in 27-MHz clock cycle units.
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

## 30.4.21 Digital Clamp Control Register 8 (DCPCR8)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCPPOS_C[7:0]								—	—	—	—	—	—	—	—
Initial value:	0	0	0	1	1	0	1	1	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	DCPPOS_C [7:0]	00011011	R/W	Digital Clamp Pulse Horizontal Start Position (Cb/Cr signal) Setting range: 0 to 255 Set a value in 27-MHz clock cycle units.
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

## (1) Digital Clamp Pulse Control (Horizontal)

DCPWIDTH, DCPPOS\_Y, and DCPPOS\_C control the digital clamp pulses in the horizontal direction. Figure 30.16 shows the bit settings and digital clamp timing.

DCPPOS\_Y is used for Y signal and DCPPOS\_C is for Cb/Cr signal. DCPWIDTH is used in common to Y, Cb, and Cr signals.

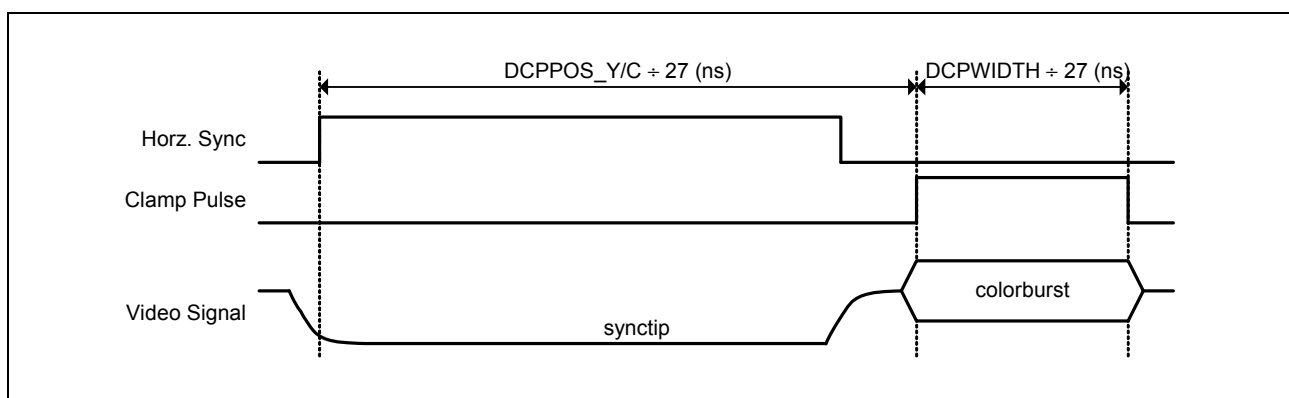


Figure 30.16 Digital Clamp Timing (Horizontal)

### 30.4.22 Noise Detection Control Register (NSDCR)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	ACFINPUT[1:0]		—	—	—	ACFLAGTIME[4:0]				—	—	ACFFILTER[1:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13, 12	ACFINPUT[1:0]	00	R/W	Video Signal for Autocorrelation Function 0: Y signal 1: Cb signal 2, 3: Cr signal
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8 to 4	ACFLAGTIME [4:0]	00000	R/W	Delay Time for Autocorrelation Function Calculation 0 to 31 clock pulses @ 27-MHz clock The NSDSR.AFCSTRENGTH value almost corresponds to noise power when the delay time is set to 0.
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	ACFFILTER [1:0]	00	R/W	Smoothing Parameter of Autocorrelation Function Data The smaller the ACFFILTER value is, the longer the time period is taken. The time period varies from 1 field to several seconds.

#### (1) Input for Noise Detection

ACFINPUT controls inputs for noise detection.

**Table 30.13 Input Selection for Noise Detection**

ACFINPUT	Input Signal
0	Y signal
1	Cb signal
2, 3	Cr signal

#### (2) Autocorrelation Function Control for Noise Detection

ACFLAGTIME controls autocorrelation function for noise detection.

The NSDSR.AFCSTRENGTH value almost corresponds to noise power when the delay time is set to 0.

#### (3) Smoothing Filter Control for Noise Detection

ACFFILTER controls smoothing function for noise detection inputs.

The smaller the ACFFILTER value is, the larger the field integration is (the longer time period is taken for noise detection).

## 30.4.23 Burst Lock/Chroma Decoding Control Register (BTLCR)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LOCKRANGE [1:0]	LOOPGAIN[1:0]	LOCKLIMIT[1:0]	BCOFR EERUN	—	DEFAULTSYS [1:0]	NONTS C358	NONTS C443	NOPAL M	NOPAL N	NOPAL 443	NOSEC AM				
Initial value:	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15, 14	LOCKRANGE [1:0]	01	R/W	Burst Lock PLL Lock Range 0: ±400 Hz 1: ±800 Hz 2: ±1200 Hz 3: ±1600 Hz
13, 12	LOOPGAIN [1:0]	01	R/W	Burst Lock PLL Loop Gain The larger value makes the response faster, but the noise is more easily picked up.
11, 10	LOCKLIMIT [1:0]	10	R/W	Level for Burst Lock PLL to Re-Search Free-Run Frequency The larger value more easily unlocks the PLL to start re-search.
9	BCOFR EERUN	0	R/W	Burst Lock PLL Free-Run Oscillation Mode ON/OFF 0: OFF 1: ON
8	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
7, 6	DEFAULTSYS[ 1:0]	00	R/W	Default Color System 0: NTSC 1: PAL 2: SECAM 3: Not specified
5	NONTSC358	0	R/W	NTSC-M Detection Control 0:NTSC-M detection ON 1:NTSC-M detection OFF
4	NONTSC443	0	R/W	NTSC-4.43 Detection Control 0:NTSC-4.43 detection ON 1:NTSC-4.43 detection OFF
3	NOPALM	0	R/W	PAL-M Detection Control 0:PAL-M detection ON 1:PAL-M detection OFF
2	NOPALN	0	R/W	PAL-N Detection Control 0:PAL-N detection ON 1:PAL-N detection OFF
1	NOPAL443	0	R/W	PAL-B, G, H, I, D Detection Control 0: PAL-B, G, H, I, D detection ON 1: PAL-B, G, H, I, D detection OFF
0	NOSECAM	00	R/W	SECAM Detection Control 0: SECAM detection ON 1: SECAM detection OFF

**(1) Lock Range of Burst Lock PLL**

LOCKRANGE controls the lock range of the burst lock PLL.

**Table 30.14 Lock Range of Burst Lock PLL**

LOCKRANGE	Lock Range of Burst Lock PLL
0	0: $\pm 400$ Hz
1	1: $\pm 800$ Hz
2	2: $\pm 1200$ Hz
3	3: $\pm 1600$ Hz

**(2) Burst Lock PLL Loop Gain Control**

LOOPGAIN controls the loop gain of the burst lock PLL.

The larger value makes the response faster, but the noise is more easily picked up.

**(3) Burst Lock PLL Lock Limit Control**

LOCKLIMIT controls the lock limit of the burst lock PLL.

The larger the LOCKLIMIT value is, the more easily the PLL free-run frequency is unlocked to start re-search.

**(4) Burst Lock PLL Free-Run Operation Control**

BCOFREERUN controls the free-run operation of burst lock PLL.

When BCOFREERUN is 1, the burst lock PLL performs free-run operation independently of the inputs. BCOFREERUN should usually be set to 0.

**(5) Color System for No Signal State during Chroma Decoding**

DEFAULTSYS sets the color system for the no signal state during chroma decoding.

**Table 30.15 Color System at No Signal**

DEFAULTSYS	Color System for No Signal State
0	NTSC
1	PAL
2	SECAM
3	Not specified

### (6) Color System Detection Control

NONTSC358, NONTSC443, NOPALM, NOPALN, NOPAL443, and NOSECAM control the color system detection. Color system detection can be fully automatic, manual, or semi-automatic (detecting the specified color system only). If the detection result does not apply to any color system type, a color system selected by DEFAULTSYS is used for the operation.

Color system detection can be controlled (ON or OFF) individually for each type. By enabling one particular color system to be detected, the color system can be fixed. Table 30.16 shows the color system detection control methods.

**Table 30.16 Color System Detection Control**

	NOSECAM	NOPAL443	NOPALN	NOPALM	NONTSC443	NONTSC358
Auto	0	0	0	0	0	0
NTSC-3.58(M)	1	1	1	1	1	0
NTSC-4.43	1	1	1	1	0	1
PAL-M	1	1	1	0	1	1
PAL-N	1	1	0	1	1	1
PAL-4.43	1	0	1	1	1	1
SECAM	0	1	1	1	1	1

In auto mode, the color system detection result is stored into the register. Table 30.17 and Table 30.18 show color system detection and detection result setting.

**Table 30.17 Color System Detection Result (1)**

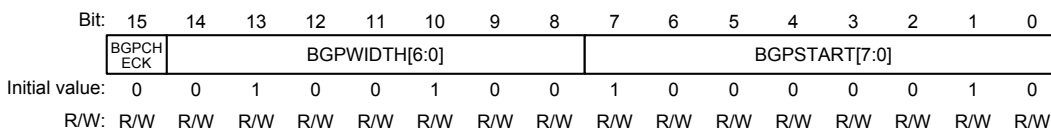
COLORSYS[1:0]	FSCMODE	FVMODE	Detection Result
0: NTSC	0: 3.58 MHz	Don't care	NTSC-M
0: NTSC	1: 4.43 MHz	Don't care	NTSC-4.43
1: PAL	0: 3.58 MHz	0: 50 Hz	PAL-N
1: PAL	0: 3.58 MHz	1: 60 Hz	PAL-M
1: PAL	1: 4.43 MHz	0: 50 Hz	PAL-B, H, I, G, D
1: PAL	1: 4.43 MHz	1: 60 Hz	PAL-60
2: SECAM	—	—	SECAM
3: Unknown	—	—	Undetectable

**Table 30.18 Color System Detection Result (2)**

	ISNTSC	ISPAL	ISSECAM
Undetectable	0	0	0
NTSC	1	0	0
PAL	0	1	0
SECAM	0	0	1



### 30.4.24 Burst Gate Pulse Control Register (BTGPCR)



Bit	Bit Name	Initial Value	R/W	Description
15	BGPCHECK	0	R/W	Burst Gate Pulse Position Check Displays the front and end edges of burst gate pulses by white lines.
14 to 8	BGPWIDTH [6:0]	0100100	R/W	Burst Gate Pulse Width (Recommended value: 88) Specify the offset from the 64th clock pulse width in 27-MHz clock cycle units.
7 to 0	BGPSTART [7:0]	10000010	R/W	Burst Gate Pulse Start Position (Recommended value: 50) Specify the position from the horizontal sync signal reference in 27-MHz clock cycle units.

#### (1) Burst Gate Pulse Control

BGPWIDTH and BGPSTART control the burst gate pulse timing.

The burst gate pulse position is specified to extract color burst from the video signal which is considered as a reference signal for the burst lock PLL. Usually, the burst gate pulse should be set so that it should start at the latter part of the horizontal sync signal and include the reference position in order to respond to an insert position shift of color burst caused by VCR.

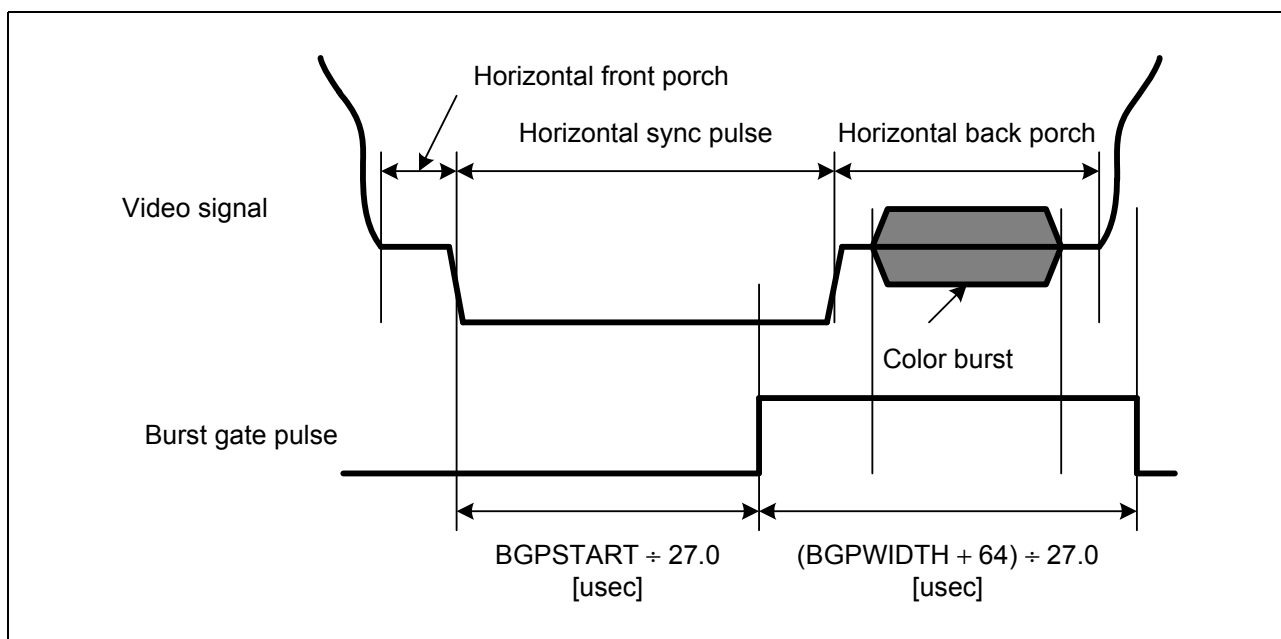


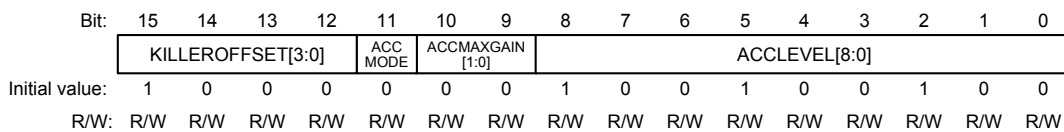
Figure 30.17 Burst Gate Pulse Generation Timing

#### (2) Burst Gate Pulse Position Check

BGPCHECK controls the burst gate pulse position check on screen. The following shows the steps to check the position:

- Burst gate position check: Set BGPCHECK to 1.
- Input video signal capturing left end setting: Set SRCLEFT to 0 and RES\_HS[10:0] in SCL0\_DS3 of the video display controller 5 scaler to 0.
- Adjust the pulse position and width with BGPSTART and BGPWIDTH.

### 30.4.25 ACC Control Register 1 (ACCCR1)



Bit	Bit Name	Initial Value	R/W	Description
15 to 12	KILLER OFFSET[3:0]	1000	R/W	The levels of these bits and KILLERLEVEL are added together to be the level to turn off the color killer. This level corresponds to the Peak-to-Peak amplitude of the color burst signal.
11	ACCMODE	0	R/W	ACC Operating Mode 0: Auto gain 1: Manual gain
10, 9	ACCMAXGAIN [1:0]	00	R/W	Maximum ACC Gain Valid when ACCMODE = 0 (auto gain setting). 0: 6 times 1: 8 times 2: 12 times 3: 16 times
8 to 0	ACCLEVEL [8:0]	100100100	R/W	ACC Reference Color Burst Amplitude (Recommended value: 236) Valid when ACCMODE = 0 (auto gain setting). Set Peak-to-Peak amplitude in 1-LSB units.

#### (1) Color Killer Offset Control

KILLEROFFSET sets the hysteresis to make the color killer OFF.

If the KILLEROFFSET value is too large, the color killer cannot be turned off as long as the burst amplitude is not large enough. If the KILLEROFFSET value is too small, the color killer is turned on and off repeatedly due to noise.

The standard value is between 4 and 10.

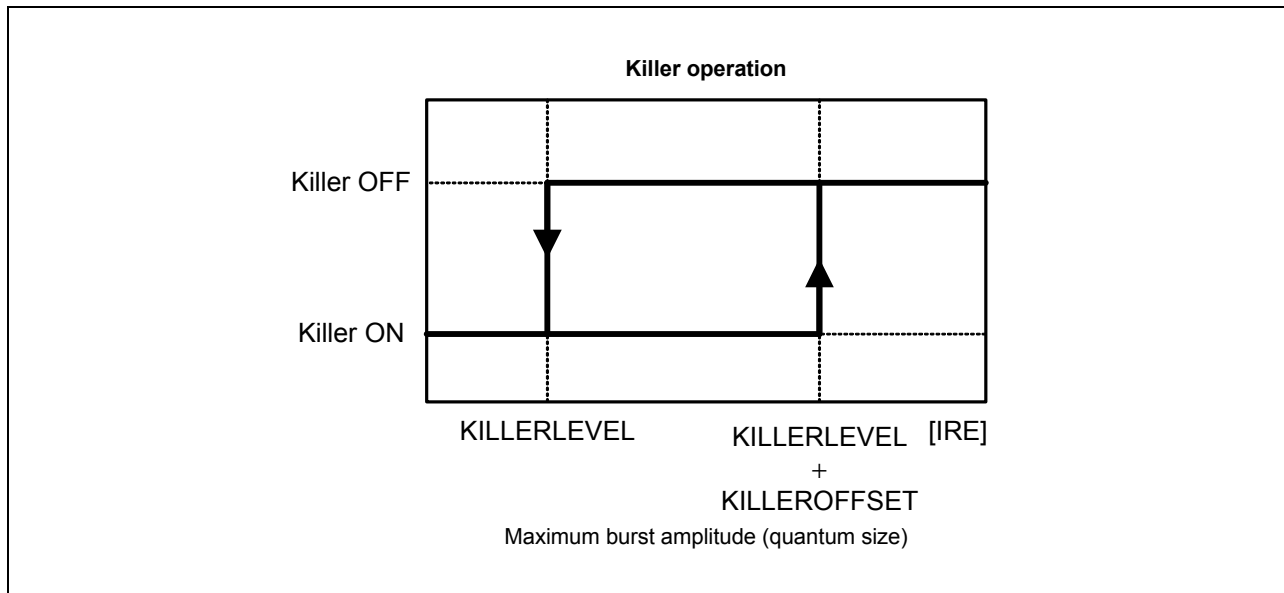


Figure 30.18 Color Killer Operation

## (2) ACC Operation Control

ACCMODE controls the ACC operation.

**Table 30.19 ACC Operating Modes**

ACCMODE	Color Gain Adjustment
0	Auto gain
1	Manual gain

## (3) Maximum ACC Gain Control

ACCMAXGAIN controls the maximum ACC gain. ACCMAXGAIN is valid only when ACCMODE = 0.

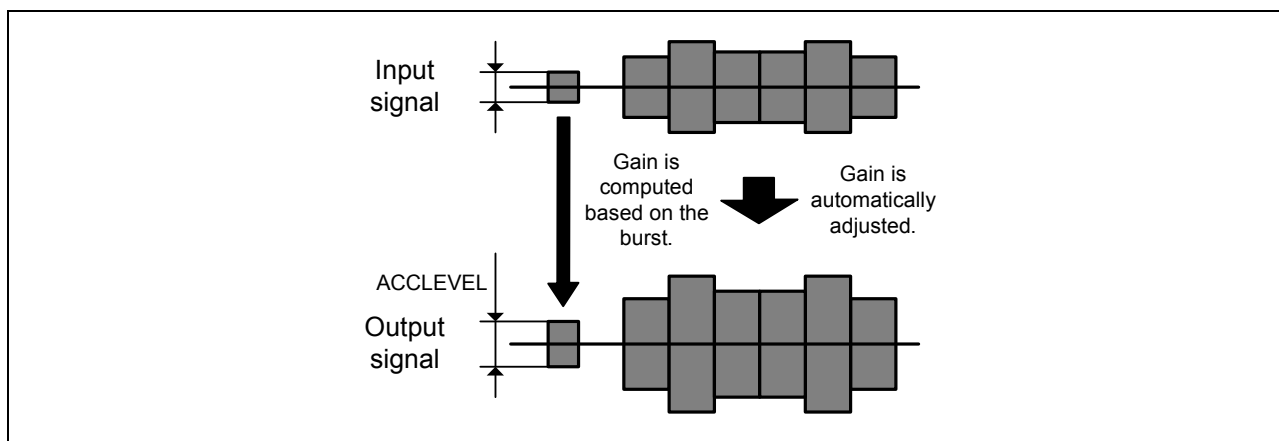
**Table 30.20 Maximum ACC Gain**

ACCMAXGAIN	Maximum Color Gain
0	6 times
1	8 times
2	12 times
3	16 times

## (4) ACC Level Control

ACCLEVEL sets the burst amplitude of the chroma signal after gain correction. ACCLEVEL is valid only when ACCMODE = 1.

The ACC adjusts the gain so that the input chroma signal burst amplitude should be the same level as the ACCLEVEL value.



**Figure 30.19 ACC Level Setting**

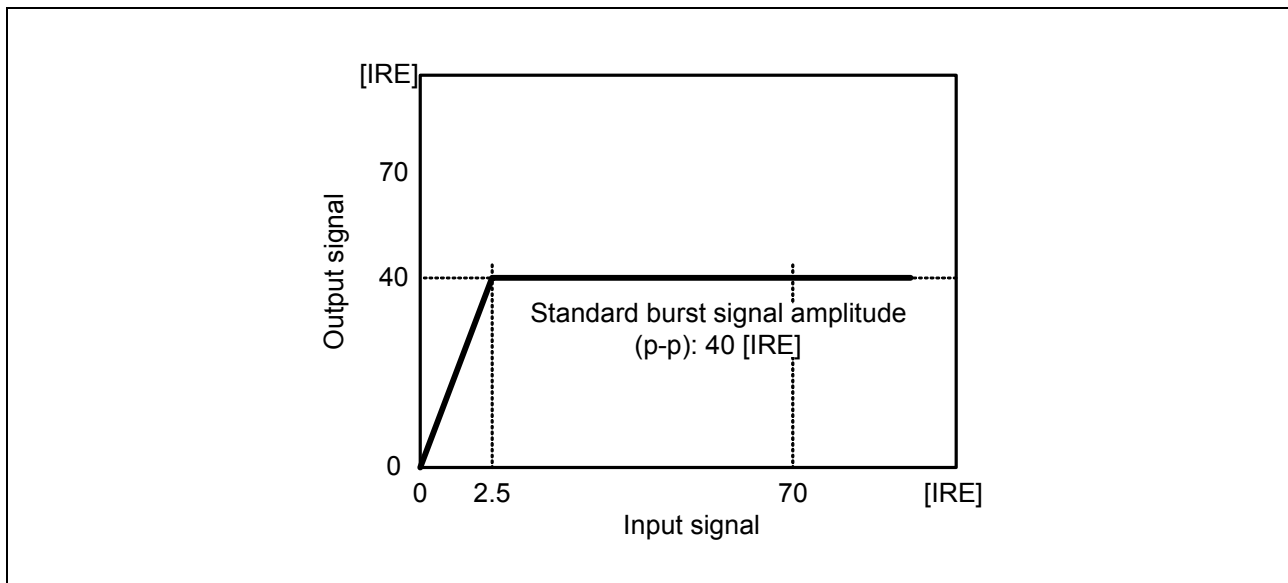


Figure 30.20 ACC Input/Output Characteristics

Table 30.21 ACC Characteristics

Input Burst Signal Level	Output Burst Signal Level
More than 24.1 [dB]	Reference amplitude (variable) ± acceptable error (variable)
24.1[dB] or less	Decreased in proportion to input level

### 30.4.26 ACC Control Register 2 (ACCCR2)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	CHROMASUBGAIN[1:0]		CHROMAMAINGAIN[8:0]								
Initial value:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10, 9	CHROMA SUBGAIN[1:0]	00	R/W	Chroma Manual Gain (sub) Valid when ACCMODE = 1 (manual gain setting) 0: 1 time 1: 2 times 2: 4 times 3: 8 times
8 to 0	CHROMA MAINGAIN[8:0]	10000000	R/W	Chroma Manual Gain (main) Valid when ACCMODE = 1 (manual gain setting) The value 256 corresponds to 1 time.

#### (1) Chroma Gain Adjustment (Manual) Control

CHROMASUBGAIN and CHROMAMAINGAIN control the chroma gain. CHROMASUBGAIN and CHROMAMAINGAIN are valid only when ACCMODE is 1.

$$C \text{ signal output} = C \text{ signal input} \times (\text{CHROMASUBGAIN} + \text{CHROMAMAINGAIN}/256)$$

### 30.4.27 ACC Control Register 3 (ACCCR3)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ACCRESPONSE [1:0]		ACCPRECIS[5:0]					KILLER MODE	KILLERLEVEL[5:0]					—		
Initial value:	0	1	0	1	0	1	0	0	0	0	0	1	0	0	1	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
15, 14	ACC RESPONSE [1:0]	01	R/W	ACC Response Speed The larger value makes the response faster, but the noise is more easily picked up. stment by 1 LSB of 10-bit accuracy.
13 to 8	ACCPRECIS [5:0]	010100	R/W	ACC Gain Adjustment Accuracy Set the acceptable error level of color burst signal amplitude after ACC adju
7	KILLERMODE	0	R/W	Forced Color Killer Mode ON/OFF 0: Auto-detection 1: Killer mode is forcedly ON.
6 to 1	KILLERLEVEL [5:0]	001001	R/W	Color Killer Operation Start Point Set the half value of Peak-to-Peak amplitude by 1 LSB of 10-bit accuracy.
0	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

#### (1) ACC Response Speed Control

ACCRESPONSE controls the ACC response speed.

The larger value makes the response faster, and the smaller value makes it slower. However, the large value causes the noise to be more easily picked up.

#### (2) ACC Acceptable Error Range Control

ACCPRECIS controls the acceptable error range of the output burst signal amplitude based on ACCLEVEL (target value).

If ACCLEVEL = 236 and ACCPRECIS = 20, the ACC gain is fixed within the following range:

$$(236 - 20) < \text{Output signal burst signal amplitude} < (236 + 20)$$

#### (3) Killer Operating Mode Control

KILLERMODE controls the killer operating mode.

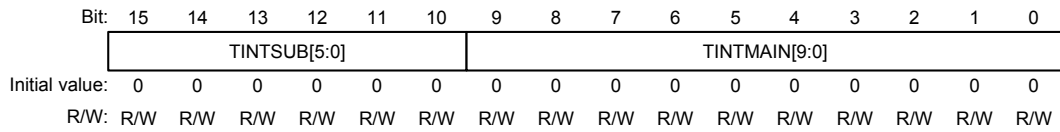
When KILLERMODE = 1, the killer is forcedly turned ON.

#### (4) Killer Level Control

KILLERLEVEL controls the level to make the killer ON.

For details, see section 30.4.25 (1), Color Killer Offset Control.

### 30.4.28 TINT Control Register (TINTCR)



Bit	Bit Name	Initial Value	R/W	Description
15 to 10	TINTSUB[5:0]	000000	R/W	Fine Adjustment of R-Y Demodulation Axis (only valid for NTSC/PAL) Set a value by 360/1024 degrees. 2s complement
9 to 0	TINTMAIN[9:0]	0000000000	R/W	Hue Adjustment Level (only valid for NTSC/PAL) Set a value by 360/1024 degrees. 2s complement

#### (1) R-Y Axis Correction Control

TINTSUB controls the phase of R-Y axis by  $\pm 11.25$  degrees.

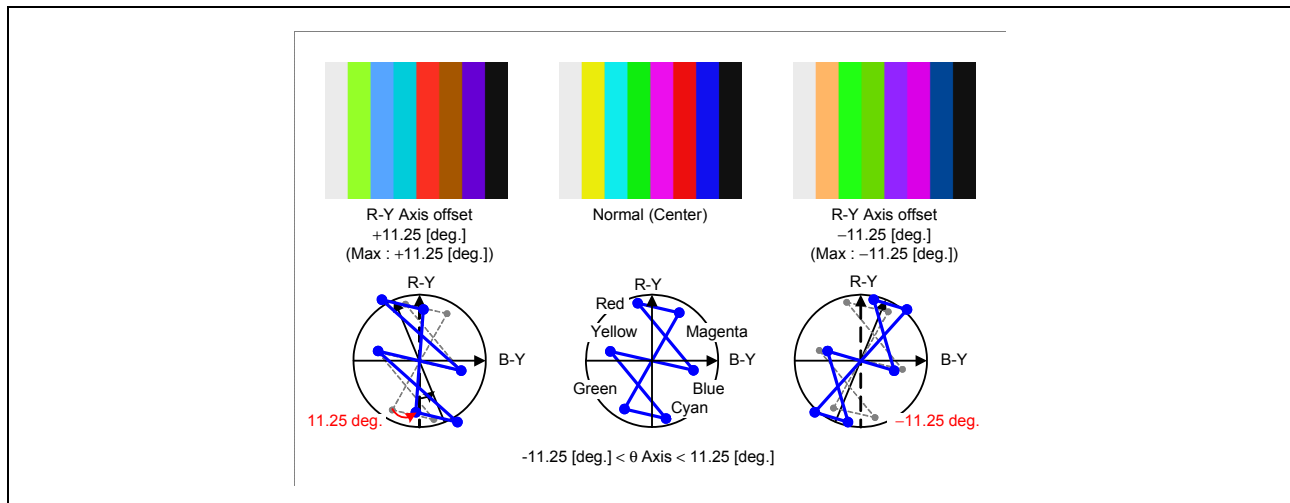


Figure 30.21 Example of R-Y Axis Correction

#### (2) Hue Adjustment (TINT) Correction Control

TINTMAIN controls the phase of demodulation axis by 0 to 360 degrees.

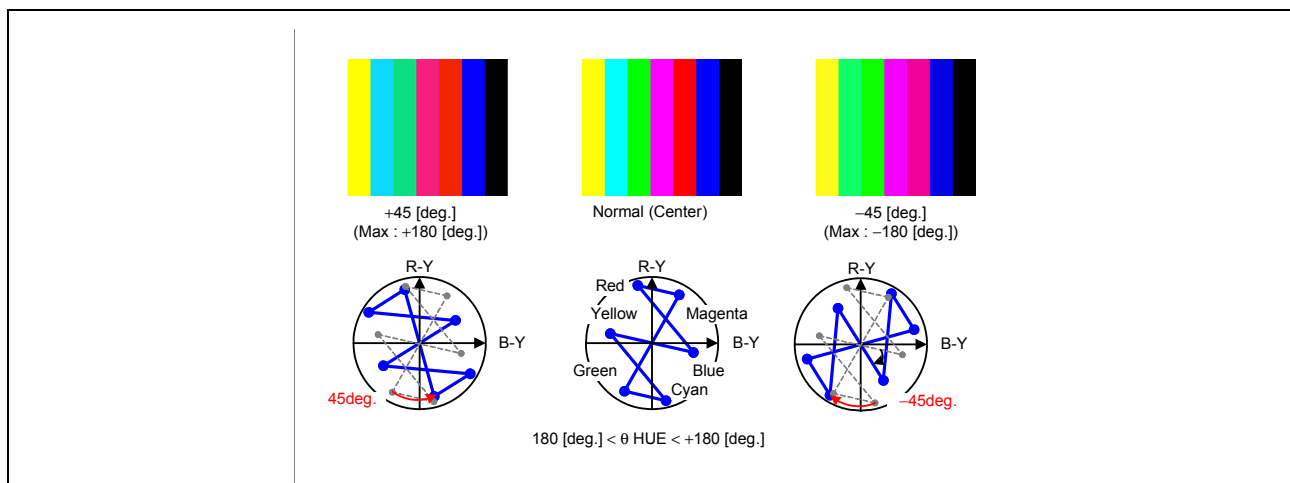


Figure 30.22 Example of Hue Adjustment (TINT) Correction

## 30.4.29 Y/C Delay/Chroma Decoding Control Register (YCDCR)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	LUMADELAY[4:0]				—	CHROMALPF	DEMOMODE [1:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8 to 4	LUMADELAY [4:0]	00000	R/W	Luminance Signal Delay Adjustment –16 to +15 clock pulses Set a value by the 2s complement.
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	CHROMALPF	0	R/W	LPF for Demodulated Chroma 0: Not used 1: Used
1, 0	DEMOMODE [1:0]	10	R/W	Averaging Processing for Pre-Demodulated Line 0: No processing 1: Setting prohibited 2: For PAL 3: Setting prohibited

## (1) Y/C Delay Adjustment Control

LUMADELAY controls the Y/C delay.

Table 30.22 Y/C Delay Adjustment

LUMADELAY	Operation
31	Advances Y signal by 1 [clk]
:	:
16	Advances Y signal by 16 [clk]
0	No delay
1	Delays Y signal by 1 [clk]
:	:
15	Delays Y signal by 15 [clk]

## (2) Frequency Band Limiting after Demodulation

CHROMALPF turns on or off the frequency band limiting filter after demodulation.

Table 30.23 Frequency Band Limiting after Demodulation

CHROMALPF	Operation
0	Frequency band-limiting filter OFF
1	Frequency band-limiting filter ON

### (3) Chroma Decoding Operation Control

DEMODMODE controls operating modes of chroma demodulation.

**Table 30.24 Chroma Decoding Operation Modes**

DEMODMODE	Operation
0	One-line demodulation
2	Two-line demodulation for PAL only
1 and 3	Setting prohibited

### 30.4.30 AGC Control Register 1 (AGCCR1)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	DORED UCE	NORED UCE	AGCRESPONSE[2:0]			AGCLEVEL[8:0]								
Initial value:	0	0	0	0	1	0	1	0	1	1	1	0	1	1	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13	DUREDUCE	0	R/W	Manual Control of Sync Signal Amplitude Detection during VBI Period 0: Sets sync amplitude to AGC standard value. 1: Sets AGC gain to 3/4 times the normal gain value.
12	NOREDUCE	0	R/W	Control of Sync Signal Amplitude Detection during VBI Period 0: Detects sync amplitude. 1: Does not detect sync amplitude.
11 to 9	AGC RESPONSE [2:0]	101	R/W	AGC Response Speed The larger register value makes the response faster. However, the larger value causes the noise to be more easily picked up.
8 to 0	AGCLEVEL [8:0]	011101100	R/W	Sync Signal Reference Amplitude Setting range: 0 to 511 10-bit unsigned value

#### (1) Sync Signal Amplitude Detection during VBI Period

DOREDUCE and NOREDUCE control detection of the AGC sync signal amplitude fluctuation during VBI period.

**Table 30.25 Sync Signal Amplitude Detection Operation during VBI Period**

DUREDUCE	Sync Signal Amplitude Detection Operation during VBI Period
0	Sets sync amplitude to AGC standard
1	Sets AGC gain to 3/4 times the normal gain value

**Table 30.26 Sync Signal Amplitude Detection during VBI Period**

NOREDUCE	Sync Signal Amplitude Detection during VBI Period
0	Detects sync amplitude.
1	Does not detect sync amplitude.



(2) AGC Response Speed Control

AGCRESPONSE controls the AGC response speed.

The larger register value makes the response faster, and smaller value makes it slower. (The large value causes the noise to be more easily picked up.)

The recommended value is 5 to avoid a malfunction caused by trick playback of VCR (fast-forward play/rewind play) or a weak electric field.

(3) AGC Level Control

AGCLEVEL controls the AGC target level.

When NTSC signals are quantized by a 10-bit A/D converter, sync signal amplitude for full range of the A/D converter can be provided by:

$$1023[\text{LSB}] \times (40[\text{IRE}] \div 173[\text{IRE}]) = 236.53179[\text{LSB}]$$

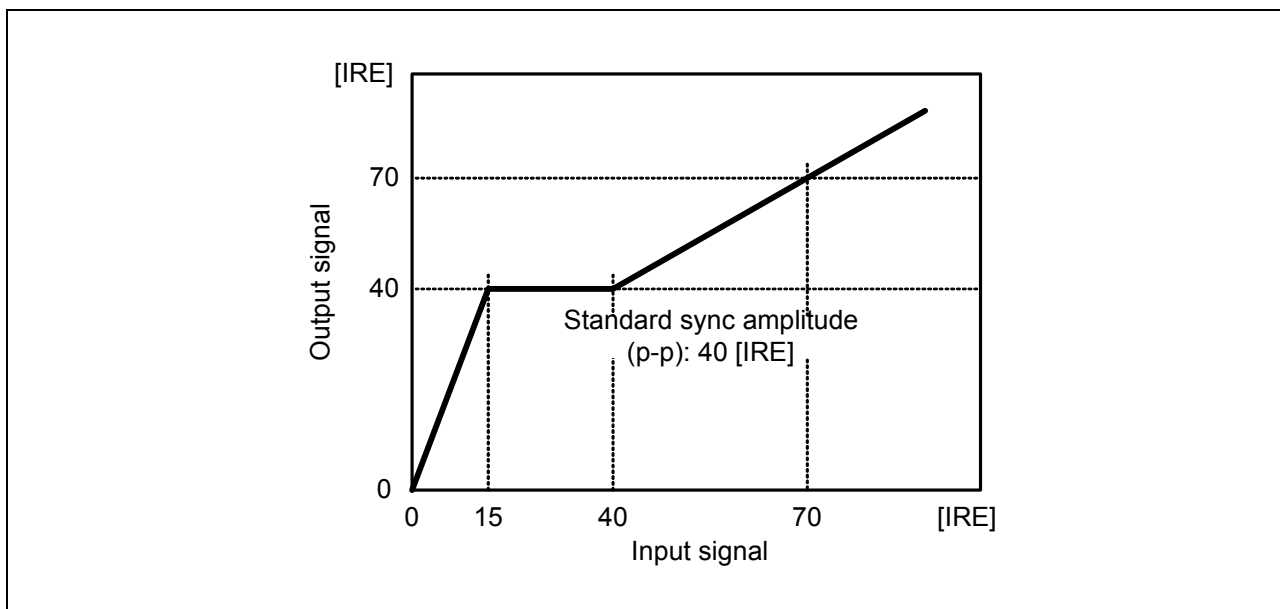
Table 30.27 shows the ideal AGC level for each input signal format.

**Table 30.27 AGC Level Setting Values (Ideal Values)**

Input Signal Format	AGCLEVEL[8:0]
NTSC	236
PAL/SECAM	248

**Table 30.28 AGC Characteristics**

Input Sync Signal Level	Output Sync Signal Level
0 or more [dB]	Increased in proportion to input level
-8.52 to 0 [dB]	Reference amplitude (variable) ± acceptable error (variable)
-8.52 or less [dB]	Decreased in proportion to input level



**Figure 30.23 AGC Characteristics (Sync Signal Amplitude Reference)**

## 30.4.31 AGC Control Register 2 (AGCCR2)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	—	—	AGCPRECIS[5:0]					—	—	—	—	—	—	—	—	—	—
Initial value:	1	1	0	0	1	0	1	0	0	1	0	0	0	0	0	0	
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	

Bit	Bit Name	Initial Value	R/W	Description
15, 14	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
13 to 8	AGCPRECIS [5:0]	001010	R/W	AGC Gain Adjustment Accuracy Set acceptable error level for sync pulse amplitude after AGC adjustment by 1 LSB of 10-bit accuracy.
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
5 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

## (1) AGC Acceptable Error Range Control

AGCPRECIS controls the acceptable error range of the output sync signal amplitude based on AGCLEVEL (target value).

If AGCLEVEL = 236 and AGCPRECIS = 10, AGC gain is fixed within the following range:

$$(236 - 10) < \text{Output sync signal amplitude} < (236 + 10)$$

In this case, the PGA gain can fall within a 2-step range. For video image with stabilized sync signal amplitude, AGCPRECIS = 4 is recommended, which enables the PGA gain to fall within a 1-step range, when the electric field is strong. For the above setting, the recommended value is 10 to avoid hunting in a weak electric field.

### 30.4.32 Peak Limiter Control Register (PKLIMITCR)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PEAKLEVEL[1:0]		PEAKATTACK [1:0]		PEAKRELEASE [1:0]		PEAKRATIO [1:0]		MAXPEAKSAMPLES[7:0]							
Initial value:	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15, 14	PEAKLEVEL [1:0]	00	R/W	Peak Luminance Value Limited by Peak Limiter (video signal level) 0: Limiter OFF 1: 1008 LSB 2: 992 LSB 3: 960 LSB Peak limiter is not operated if AGC is OFF irrespective of PEAKLEVEL value.
13, 12	PEAKATTACK [1:0]	10	R/W	Response Speed with Peak Limiter Gain Decreased The larger value makes the response faster.
11, 10	PEAKRELEASE [1:0]	00	R/W	Response Speed with Peak Limiter Gain Increased The larger value makes the response faster.
9, 8	PEAKRATIO [1:0]	00	R/W	Maximum Compression Rate of Peak Limiter 0: Compressed up to 50% 1: Compressed up to 25% 2: Compressed up to 12.5% 3: Compressed up to 0%
7 to 0	MAXPEAK SAMPLES[7:0]	00000000	R/W	Allowable Number of Overflowing Pixels Set a value by 1024 pixels. Exceeding this value will start peak limiter operation.

#### (1) Peak Limiter Level Control

PEAKLEVEL controls the peak luminance limited by the peak limiter.

If the number of pixels counted exceeds the value set in PEAKLEVEL and there exist pixels more than the value set in MAXPEAKSAMPLES, the peak limiter function is activated to reduce the gain.

**Table 30.29 Peak Limiter Level Control**

PEAKLEVEL	Output Sync Signal Level
0	Peak limiter OFF
1	Peak limiter is activated at 1008 LSB
2	Peak limiter is activated at 992 LSB
3	Peak limiter is activated at 960 LSB

#### (2) Peak Limiter Response Speed Control

PEAKATTACK controls the response speed when the peak limiter gain is reduced.

The larger value makes the response faster.

#### (3) Peak Limiter Response Speed Control

PEAKRELEASE controls the response speed when the peak limiter gain is increased.

The larger value makes the response faster.

#### (4) Peak Limiter Gain Down Control

PEAKRATIO sets the maximum compression rate of the peak limiter.

Specifically, PEAKRATIO controls the amount of gain reduction (compression ratio) using the peak limiter function.

**Table 30.30 Peak Limiter Gain Down Control**

PEAKRATIO	Output Sync Signal Level
0	Compressed up to 50%
1	Compressed up to 25%
2	Compressed up to 12.5%
3	Compressed up to 0%

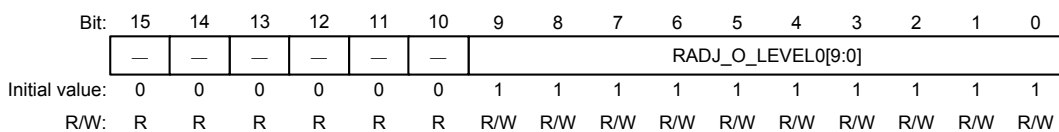
#### (5) Peak Limiter Determination Control

MAXPEAKSAMPLES controls the number of overflowing pixels allowed.

If the number of pixels counted during the vertical active period exceeds the value set in PEAKLEVEL and there exist pixels more than the value set in MAXPEAKSAMPLES, the peak limiter function is activated to reduce the gain.

The maximum allowable value is obtained by  $\text{MAXPEAKSAMPLES} \times 1024$ .

### 30.4.33 Over-Range Control Register 1 (RGORCR1)



Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	RADJ_O_LEVEL0[9:0]	111111111	R/W	A/D Over-Threshold Level (Between levels 0 and 1) Level 0 (normal) to level 3 (completely over the range) are available.

#### (1) A/D Over-Threshold Level (Between Levels 0 and 1) Control

RADJ\_O\_LEVEL0 controls the A/D over-threshold level (between levels 0 and 1).

Figure 30.24 shows the register values and threshold levels.

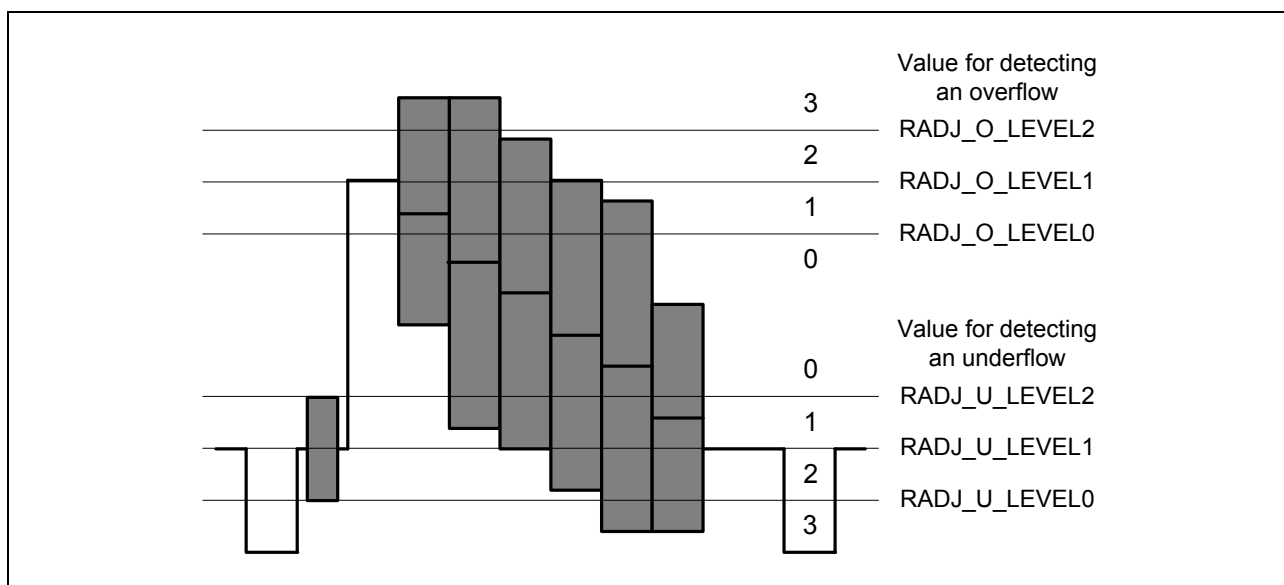


Figure 30.24 Over-Range Determination Areas

### 30.4.34 Over-Range Control Register 2 (RGORCR2)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RADJ_U_LEVEL0[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	RADJ_U_LEVEL0[9:0]	0000000000	R/W	A/D Under-Threshold Level (Between levels 2 and 3) Level 0 (normal) to level 3 (completely under the range) are available.

#### (1) A/D Under-Threshold Level (Between Levels 2 and 3) Control

RADJ\_U\_LEVEL0 controls the A/D under-threshold level (between levels 2 and 3).

For the register values and threshold levels, see section 30.4.33, Over-Range Control Register 1 (RGORCR1).

### 30.4.35 Over-Range Control Register 3 (RGORCR3)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RADJ_O_LEVEL1[9:0]									
Initial value:	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	RADJ_O_LEVEL1[9:0]	1111111111	R/W	A/D Over-Threshold Level (Between levels 1 and 2) Level 0 (normal) to level 3 (completely over the range) are available.

#### (1) A/D Over-Threshold Level (Between Levels 1 and 2) Control

RADJ\_O\_LEVEL1 controls the A/D over-threshold level (between levels 1 and 2).

For the register values and threshold levels, see section 30.4.33, Over-Range Control Register 1 (RGORCR1).

### 30.4.36 Over-Range Control Register 4 (RGORCR4)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RADJ_U_LEVEL1[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	RADJ_U_LEVEL1[9:0]	0000000000	R/W	A/D Under-Threshold Level (Between levels 1 and 2) Level 0 (normal) to level 3 (completely under the range) are available.

#### (1) A/D Under-Threshold Level (Between Levels 1 and 2) Control

RADJ\_U\_LEVEL1 controls the A/D under-threshold level (between levels 1 and 2).

For the register values and threshold levels, see section 30.4.33, Over-Range Control Register 1 (RGORCR1).

### 30.4.37 Over-Range Control Register 5 (RGORCR5)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RADJ_O_LEVEL2[9:0]									
Initial value:	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	RADJ_O_LEVEL2[9:0]	1111111111	R/W	A/D Over-Threshold Level (Between levels 2 and 3) Level 0 (normal) to level 3 (completely over the range) are available.

#### (1) A/D Over-Threshold Level (Between Levels 2 and 3) Control

RADJ\_O\_LEVEL2 controls the A/D over-threshold level (between levels 2 and 3).

For the register values and threshold levels, see section 30.4.33, Over-Range Control Register 1 (RGORCR1).

## 30.4.38 Over-Range Control Register 6 (RGORCR6)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RADJ_U_LEVEL2[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	RADJ_U_LEVEL2[9:0]	000000000	R/W	A/D Under-Threshold Level (Between levels 0 and 1) Level 0 (normal) to level 3 (completely under the range) are available.

## (1) A/D Under-Threshold Level (Between Levels 0 and 1) Control

RADJ\_U\_LEVEL2 controls the A/D under-threshold level (between levels 0 and 1).

For the register values and threshold levels, see section 30.4.33, Over-Range Control Register 1 (RGORCR1).



## 30.4.39 Over-Range Control Register 7 (RGORCR7)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	TEST_MONI[2:0]			RADJ_MIX_K_FIX[2:0]			—	—	—	—	—	—	UCMP_SW	DCMP_SW	HWIDE_SW
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14 to 12	TEST_MONI [2:0]	000	R/W	Test Mode 0 to 3: Normal operation 4: Level 0 part is output as black. 5: Level 1 part is output as black. 6: Level 2 part is output as black. 7: Level 3 part is output as black.
11 to 9	RADJ_MIX_K_ FIX[2:0]	000	R/W	Forced Range Over/Under Mode 0 to 3: Auto detection 4: Fixed to Level 0 (normal state) 5: Fixed to level 1 (almost normal) 6: Fixed to level 2 (almost over the range) 7: Fixed to level 3 (completely over the range)
8 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	UCMP_SW	0	R/W	Over-Range Detection Enable 0: Disables over-range detection 1: Enables over-range detection
1	DCMP_SW	0	R/W	Under-Range Detection Enable 0: Disables under-range detection 1: Enables under-range detection
0	HWIDE_SW	1	R/W	Horizontal Enlargement of Over/Under-Range Level 0: Does not provide horizontal enlargement 1: Provides horizontal enlargement

## (1) Over-Range Test Control

TEST\_MONI controls the over-range test.

## (2) Forced Over/Under-Range Mode Control

RADJ\_MIX\_K\_FIX controls the forced over-/under-range detection.

## (3) Over-Range Detection Control

UCMP\_SW enables the over-range detection.

## (4) Under-Range Detection Control

DCMP\_SW enables the under-range detection.

## (5) Horizontal Enlargement at Over/Under-Range Level

HWIDE\_SW controls the horizontal enlargement of the over/under-range level.

### 30.4.40 Feedback Control Register for Horizontal AFC Phase Comparator (AFCPF CR)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	PHDET_FIX	—	PHDET_DIV[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	PHDET_FIX	0	R/W	Forcible or LOWGAIN Control 0: LOWGAIN determination result used 1: Forcibly controlled (adjusted with PHDET_DIV)
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2 to 0	PHDET_DIV [2:0]	101	R/W	Phase Comparator Feedback Adjust for Low Sync Signal Lock Stability 0: 1/1 1: 1/2 2: 1/4 3: 1/8 4: 1/16 5: 1/32 6 and 7: Setting prohibited

#### (1) Phase Comparator Feedback Adjust

PHDET\_DEV adjusts the feedback amount as the phase comparison result when the lock stability is low. The greater the denominator is, the slower the reaction speed to the signal is.

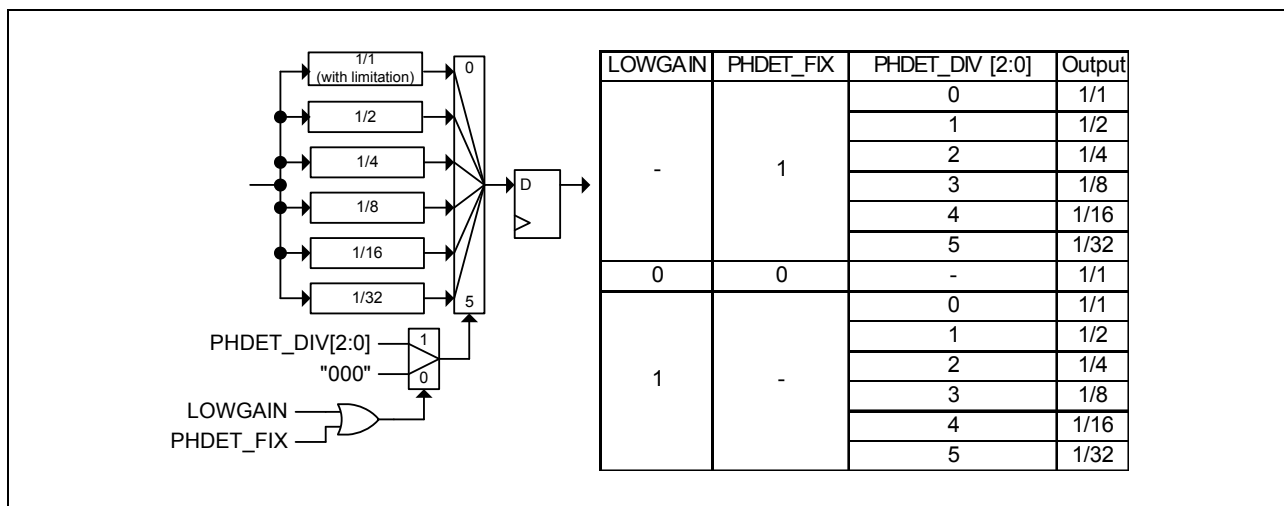


Figure 30.25 Phase Comparator Feedback Adjust

## 30.4.41 Register Update Enable Register (RUPDCR)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NEWSETTING	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	NEWSETTING	0	R/W	V Update Enable for TGCR1 to TGCR3 1: Enables update. 0: Disables update.
14 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

## (1) V Update Enable for TGCR1 to TGCR3

NEWSETTING enables/disables TGCR1 to TGCR3 to execute V update.

## 30.4.42 Sync Separation Status/Vertical Cycle Read Register (VSYNC\_CSR)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FHCOU NT[0]	FHLOCK	ISNOISY	FHMODE	NOSIGN AL	FVLOCK	FVMOD E	INTERL ACED	FVCOUNT[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	FHCOUNT[0]	0	R	Horizontal AFC Oscillation Cycle (bit 0) Set a value by 1/64 of 27-MHz clock.
14	FHLOCK	0	R	Horizontal AFC Lock Detection Result 0: Unlocked 1: Locked
13	ISNOISY	0	R	Detection Result of Low S/N Signal by Sync Separation 0: Not low S/N signal 1: Low S/N signal
12	FHMODE	0	R	Speed Detection Result 0: Normal speed (525i/625i, etc.) 1: Multiplied speed (525p/625p, etc.)
11	NOSIGNAL	0	R	No-Signal Detection Result 0: Vertical sync signal detected 1: No vertical sync signal detected
10	FVLOCK	0	R	Vertical Countdown Lock Detection Result 0: Unlocked 1: Locked
9	FVMODE	0	R	Vertical Countdown Oscillation Mode 0: 50Hz 1: 60Hz
8	INTERLACED	0	R	Interlace Detection Result 0: Progressive 1: Interlace
7 to 0	FVCOUNT[7:0]	00000000	R	Vertical Cycle Measurement Result (in 0.1-ms units)

## (1) Horizontal AFC Oscillation Cycle Read

FHCOUNT indicates bit 0 of the horizontal AFC oscillation cycle.

## (2) Horizontal AFC Lock Detection Result Read

FHLOCK indicates the horizontal AFC lock detection result.

## (3) Sync Separation Low S/N Signal Detection Result Read

ISNOISY indicates the detection result of low S/N signal by sync separation.

## (4) Speed Detection Result Read

FHMODE indicates the speed detection result.

## (5) No-Signal Detection Result Read

NOSIGNAL indicates the no-signal detection result.

## (6) Vertical Countdown Lock Detection Result Read

FVLOCK indicates the vertical countdown lock detection result.

## (7) Interlace Detection Result Read

INTERFACED indicates the interlace detection result.

**(8) Vertical Cycle Measurement Result Read**

FVCOUNT indicates the vertical cycle measurement result.

**30.4.43 Horizontal Cycle Read Register (HSYNCSR)**

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FHCCOUNT[16:1]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	FHCCOUNT [16:1]	H'0000	R	Horizontal AFC Oscillation Cycle (bit 16 to bit 1) Set a value by 1/64 of 27-MHz clock.

**(1) Horizontal AFC Oscillation Cycle Read**

FHCCOUNT indicates the upper bits of the horizontal AFC oscillation cycle.

**30.4.44 Digital Clamp Read Register 1 (DCPSR1)**

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CLAMPLEVEL_CB[5:0]						CLAMPLEVEL_Y[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	CLAMPLEVEL_ CB[5:0]	000000	R	Digital Clamp Subtraction Value (Cb signal) Offset from the reference black level Set a value in 1-LSB units. 2s complement
9 to 0	CLAMPLEVEL_ Y[9:0]	000000000	R	Digital Clamp Subtraction Value (Y signal) Offset from the reference black level Set a value in 1-LSB units. 2s complement

**(1) Reading Digital Clamp Subtraction Value of Cb Signal**

CLAMPLEVEL\_CB indicates the digital clamp subtraction value of Cb signal.

**(2) Reading Digital Clamp Subtraction Value of Y in Composite Signal**

CLAMPLEVEL\_Y indicates the digital clamp subtraction value of Y signal.

### 30.4.45 Digital Clamp Read Register 2 (DCPSR2)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CLAMPLEVEL_CR[5:0]										—	—	—	—	—	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	CLAMPLEVEL_CR[5:0]	000000	R	Digital Clamp Subtraction Value (Cr signal) Offset from the reference black level Set a value in 1-LSB units. 2s complement
9 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

#### (1) Reading Digital Clamp Subtraction Value of Cr Signal

CLAMPLEVEL\_CR indicates the digital clamp subtraction value of Cr signal.

### 30.4.46 Noise Detection Read Register (NSDSR)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ACFSTRENGTH[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 0	ACF STRENGTH [15:0]	H'0000	R	Noise Autocorrelation Strength at Digital Clamp Pulse Position (normally in the pedestal position) When ACFLAGTIME = 0, ACFSTRENGTH almost corresponds to the noise power in the pedestal position. Square root and logarithm of detection result almost correspond to noise amplitude and S/N (relative value), respectively.

#### (1) Reading Noise Autocorrelation Strength at Digital Clamp Pulse Position

ACFSTRENGTH indicates the noise correlation strength at the digital clamp pulse position (normal pedestal position).

## 30.4.47 Chroma Decoding Read Register 1 (CROMASR1)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	COLORSYS[1:0]		FSC MODE	FSC LOCK	NO BURST	ACCSUBGAIN [1:0]		ACCMaingain[8:0]								
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15, 14	COLORSYS [1:0]	00	R	Color System Detection Result 0: NTSC 1: PAL 2: SECAM 3: Undetectable
13	FSCMODE	0	R	Color Sub-Carrier Frequency Detection Result 0: 3.58 MHz 1: 4.43 MHz
12	FSCLOCK	0	R	Burst Lock PLL Lock State Detection Result 0: Unlocked 1: Locked
11	NOBURST	0	R	Color Burst Detection Result 0: Color burst present 1: No color burst present
10, 9	ACCSUBGAIN [1:0]	00	R	Current ACC Gain Value (Sub) 0: 1 time 1: 2 times 2: 4 times 3: 8 times
8 to 0	ACCMaingain [8:0]	00000000	R	Current ACC gain value (Main) The value 256 corresponds to 1 time.

## (1) Color System Detection Result Read

COLORSYS indicates the color system detection result.

## (2) Color Sub-Carrier Frequency Detection Result Read

FSCMODE indicates the color sub-carrier frequency detection result.

## (3) Burst Lock PLL Lock State Detection Result Read

FSCLOCK indicates the lock state detection result of the burst lock PLL.

## (4) Color Burst Detection Result Read

NOBURST indicates the color burst detection result.

## (5) Current ACC Gain (Sub) Value Read

ACCSUBGAIN indicates the current ACC gain (sub) value.

## (6) Current ACC Gain (Main) Value Read

ACCMaingain indicates the current ACC gain (main) value.

## 30.4.48 Chroma Decode Read Register 2 (CROMASR2)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	ISSE CAM	ISPAL	ISNTSC	—	—	LOCKLEVEL[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	ISSECAM	0	R	SECAM Detection Result 0: Not SECAM signal 1: SECAM signal
11	ISPAL	0	R	PAL Detection Result 0: Not PAL signal 1: PAL signal
10	ISNTSC	0	R	NTSC Detection Result 0: Not NTSC signal 1: NTSC signal
9, 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	LOCKLEVEL [7:0]	00000000	R	Low S/N Signal Detection Result by Burst Lock PLL The larger value corresponds to a higher S/N.

## (1) SECAM Signal Detection Result Read

ISSECAM indicates the SECAM signal detection result.

## (2) PAL Signal Detection Result Read

ISPAL indicates the PAL signal detection result.

## (3) NTSC Signal Detection Result Read

ISNTSC indicates the NTSC signal detection result.

## (4) Read of Low S/N Signal Detection Result by Burst Lock PLL

LOCKLEVEL indicates the low S/N signal detection result by the burst lock PLL.



### 30.4.49 Sync Separation Read Register (SYNCCSSR)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	—	—	—	ISREDUCED	—	—	SYNCDDEPTH[9:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	ISREDUCED	0	R	Sync Amplitude Detection Result during VBI Period 0: Amplitude is larger than that in image active period. 1: Amplitude is equal to that in image active period.
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	SYNCDDEPTH [9:0]	000000000	R	Sync Pulse Amplitude Detection Result

#### (1) Reading Sync Amplitude Detection Result during VBI Period

ISREDUCED indicates the sync amplitude detection result during VBI period.

#### (2) Reading Sync Pulse Level Amplitude Detection Result

SYNCDDEPTH indicates the sync pulse amplitude detection result.

### 30.4.50 AGC Control Read Register 1 (AGCCSR1)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HIGHSAMPLES[7:0]							PEAKSAMPLES[7:0]								
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 8	HIGH SAMPLES[7:0]	00000000	R	Number of Pixels Which Have Larger Luminance Value Than Peak Luminance Limited by Peak Limiter Indicated by 1024 pixels
7 to 0	PEAK SAMPLES[7:0]	00000000	R	Number of Overflowing Pixels Indicated by 1024 pixels

#### (1) Reading Number of Pixels Which Have Larger Luminance Value Than Peak Luminance Limited by Peak Limiter

HIGHSAMPLES indicates the number of pixels which have larger luminance value than the peak luminance limited by the peak limiter.

#### (2) Number of Overflowing Pixels

PEAKSAMPLES indicates the number of overflowing pixels.

## 30.4.51 AGC Control Read Register 2 (AGCCSR2)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	AGCCON VERGE	AGCGAIN[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	AGC CONVERGE	0	R	AGC Convergence Detection Result 0: Not converged 1: Converged
7 to 0	AGCGAIN[7:0]	01000000	R	Current AGC Gain Value The value 64 corresponds to ×1.

## (1) Reading AGC Convergence Detection Result

AGCCONVERGE indicates the AGC convergence detection result.

## (2) Reading Current AGC Gain

AGCGAIN indicates the current AGC gain.

## 30.4.52 Y/C Separation Control Register 3 (YCSCR3)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	K15[3:0]				K13[5:0]					K11[5:0]						
Initial value:	0	0	1	0	0	0	1	0	0	0	0	0	0	1	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	K15[3:0]	0010	R/W	Two-Dimensional Y/C Separation Filter Select Coefficient As the value becomes larger, the horizontal BPF is applied to the narrower range.
11 to 6	K13[5:0]	001000	R/W	Two-Dimensional Y/C Separation Filter Select Coefficient As the value becomes larger, the horizontal BPF is applied to the narrower range.
5 to 0	K11[5:0]	000100	R/W	Two-Dimensional Y/C Separation Filter Select Coefficient As the value becomes larger, the horizontal BPF is applied to the narrower range.

## (1) Two-Dimensional Y/C Separation Filter Select Coefficient Control

K15 controls the two-dimensional Y/C separation filter select coefficient.

For details, refer to section 30.5.5 (3), Horizontal and Vertical Correlation Detection Block.

## (2) Two-Dimensional Y/C Separation Filter Select Coefficient Control

K13 controls the two-dimensional Y/C separation filter select coefficient.

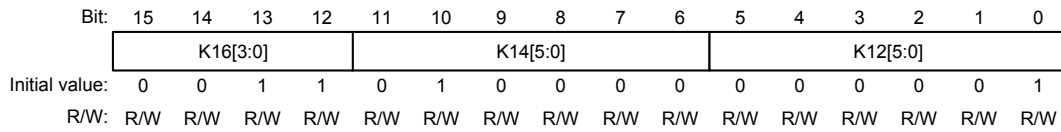
For details, refer to section 30.5.5 (3), Horizontal and Vertical Correlation Detection Block.

## (3) Two-Dimensional Y/C Separation Filter Select Coefficient Control

K11 controls the two-dimensional Y/C separation filter select coefficient.

For details, refer to section 30.5.5 (3), Horizontal and Vertical Correlation Detection Block.

### 30.4.53 Y/C Separation Control Register 4 (YCSCR4)



Bit	Bit Name	Initial Value	R/W	Description
15 to 12	K16[3:0]	0011	R/W	Two-Dimensional Y/C Separation Filter Select Coefficient As the value becomes larger, the horizontal BPF is applied to the narrower range.
11 to 6	K14[5:0]	010000	R/W	Two-Dimensional Y/C Separation Filter Select Coefficient As the value becomes larger, the horizontal BPF is applied to the narrower range.
5 to 0	K12[5:0]	000001	R/W	Two-Dimensional Y/C Separation Filter Select Coefficient As the value becomes larger, the horizontal BPF is applied to the narrower range.

#### (1) Two-Dimensional Y/C Separation Filter Select Coefficient Control

K16 controls the two-dimensional Y/C separation filter select coefficient.

For details, refer to section 30.5.5 (3), Horizontal and Vertical Correlation Detection Block.

#### (2) Two-Dimensional Y/C Separation Filter Select Coefficient Control

K14 controls the two-dimensional Y/C separation filter select coefficient.

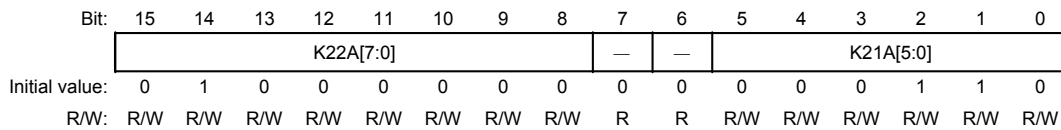
For details, refer to section 30.5.5 (3), Horizontal and Vertical Correlation Detection Block.

#### (3) Two-Dimensional Y/C Separation Filter Select Coefficient Control

K12 controls the two-dimensional Y/C separation filter select coefficient.

For details, refer to section 30.5.5 (3), Horizontal and Vertical Correlation Detection Block.

### 30.4.54 Y/C Separation Control Register 5 (YCSCR5)



Bit	Bit Name	Initial Value	R/W	Description
15 to 8	K22A[7:0]	01000000	R/W	Two-Dimensional Y/C Separation Filter Select Coefficient As the value becomes larger, the vertical BPF is applied to the narrower range.
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5 to 0	K21A[5:0]	000110	R/W	Two-Dimensional Y/C Separation Filter Select Coefficient As the value becomes larger, the vertical BPF is applied to the narrower range.

#### (1) Two-Dimensional Y/C Separation Filter Select Coefficient Control

K22A controls the two-dimensional Y/C separation filter select coefficient.

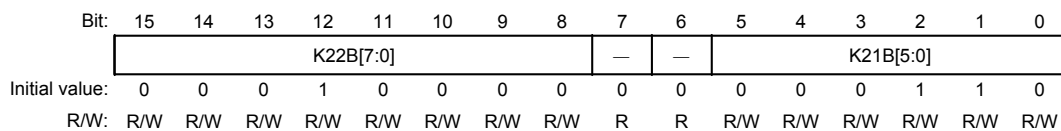
For details, refer to section 30.5.5 (3), Horizontal and Vertical Correlation Detection Block.

#### (2) Two-Dimensional Y/C Separation Filter Select Coefficient Control

K21A controls the two-dimensional Y/C separation filter select coefficient.

For details, refer to section 30.5.5 (3), Horizontal and Vertical Correlation Detection Block.

### 30.4.55 Y/C Separation Control Register 6 (YCSCR6)



Bit	Bit Name	Initial Value	R/W	Description
15 to 8	K22B[7:0]	00010000	R/W	Two-Dimensional Y/C Separation Filter Select Coefficient As the value becomes larger, the vertical BPF is applied to the narrower range.
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5 to 0	K21B[5:0]	000110	R/W	Two-Dimensional Y/C Separation Filter Select Coefficient As the value becomes larger, the vertical BPF is applied to the narrower range.

#### (1) Two-Dimensional Y/C Separation Filter Select Coefficient Control

K22B controls the two-dimensional Y/C separation filter select coefficient.

For details, refer to section 30.5.5 (3), Horizontal and Vertical Correlation Detection Block.

#### (2) Two-Dimensional Y/C Separation Filter Select Coefficient Control

K21B controls the two-dimensional Y/C separation filter select coefficient.

For details, refer to section 30.5.5 (3), Horizontal and Vertical Correlation Detection Block.

## 30.4.56 Y/C Separation Control Register 7 (YCSCR7)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	K23B[3:0]				K23A[3:0]				—	—	—	K24[4:0]				
Initial value:	0	1	1	0	0	0	1	1	0	0	1	0	0	1	0	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	K23B[3:0]	0110	R/W	Two-Dimensional Y/C Separation Filter Select Coefficient As the value becomes larger, the vertical BPF is applied to the narrower range.
11 to 8	K23A[3:0]	0011	R/W	Two-Dimensional Y/C Separation Filter Select Coefficient As the value becomes larger, the vertical BPF is applied to the narrower range.
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
4 to 0	K24[4:0]	00101	R/W	Two-Dimensional Y/C Separation Filter Select Coefficient As the value becomes larger, the vertical BPF is applied to the wider range.

## (1) Two-Dimensional Y/C Separation Filter Select Coefficient Control

K23B controls the two-dimensional Y/C separation filter select coefficient.

For details, refer to section 30.5.5 (3), Horizontal and Vertical Correlation Detection Block.

## (2) Two-Dimensional Y/C Separation Filter Select Coefficient Control

K23A controls the two-dimensional Y/C separation filter select coefficient.

For details, refer to section 30.5.5 (3), Horizontal and Vertical Correlation Detection Block.

## (3) Two-Dimensional Y/C Separation Filter Select Coefficient Control

K24 controls the two-dimensional Y/C separation filter select coefficient.

For details, refer to section 30.5.5 (3), Horizontal and Vertical Correlation Detection Block.

### 30.4.57 Y/C Separation Control Register 8 (YCSCR8)

In two-dimensional Y/C separation, horizontal BPF, vertical BPF, and horizontal/vertical BPF are adaptively switched. For the horizontal BPF and horizontal/vertical BPF, horizontal properties can be selected.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	HBPF_NARROW	HVBPF_NARROW	HBPF1_9TAP_ON	HVBPF1_9TAP_ON	HFIL_TAP_SEL	—	—	—	—	—	—	—	—	—	—	—
Initial value:	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	HBPF_NARROW	1	R/W	Latter-Stage Horizontal BPF Select 0: Bypass 1: 17 TAP
14	HVBPF_NARROW	1	R/W	Latter-Stage Horizontal/Vertical BPF Select 0: Bypass 1: 17 TAP
13	HBPF1_9TAP_ON	0	R/W	Former-Stage Horizontal BPF Select 0: 17 TAP 1: 9 TAP
12	HVBPF1_9TAP_ON	0	R/W	Former-Stage Horizontal/Vertical BPF Select 0: 17 TAP 1: 9 TAP
11	HFIL_TAP_SEL	0	R/W	Horizontal Filter and Horizontal/Vertical Filter Bandwidth Switch Signal 0: 17 TAP 1: 9 TAP
10 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

#### (1) Horizontal BPF Select Control

HBPF\_NARROW selects the latter-stage horizontal BPF.

For details, refer to section 30.5.5 (2), Horizontal and Vertical Filter Block.

#### (2) Horizontal/Vertical BPF Select Control

HVBPF\_NARROW selects the latter-stage horizontal/vertical BPF.

For details, refer to section 30.5.5 (2), Horizontal and Vertical Filter Block.

#### (3) Horizontal BPF (Broadband) Select Control

HBPF1\_9TAP\_ON selects the former-stage horizontal BPF.

For details, refer to section 30.5.5 (2), Horizontal and Vertical Filter Block.

#### (4) Horizontal/Vertical BPF (Broadband) Select Control

HVBPF1\_9TAP\_ON selects the former-stage horizontal/vertical BPF.

For details, refer to section 30.5.5 (2), Horizontal and Vertical Filter Block.

#### (5) Horizontal BPF Bandwidth Switch Control

HFIL\_TAP\_SEL switches the horizontal BPF bandwidths used for mixing.

For details, refer to section 30.5.5 (5), Horizontal and Vertical Signal Mixing Block.

## 30.4.58 Y/C Separation Control Register 9 (YCSCR9)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DET2_ON	—	—	—	HSEL_MIX_Y[3:0]				VSEL_MIX_Y[3:0]				HVSEL_MIX_Y[3:0]			
Initial value:	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	DET2_ON	1	R/W	Two-Dimensional Filter Mixing Select 0: Signals are not mixed after passing the correlation detection filter. 1: Signals are mixed after passing the correlation detection filter.
14 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 8	HSEL_MIX_Y [3:0]	0000	R/W	Mixing Ratio of Signal after Passing Horizontal Filter to Signal after Passing Former-Stage Horizontal Filter 0: Horizontal filter 100.0% 1: Horizontal filter 87.5% to former-stage horizontal filter 12.5% 2: Horizontal filter 75.0% to former-stage horizontal filter 25.0% 3: Horizontal filter 62.5% to former-stage horizontal filter 37.5% 4: Horizontal filter 50.0% to former-stage horizontal filter 50.0% 5: Horizontal filter 37.5% to former-stage horizontal filter 62.5% 6: Horizontal filter 25.0% to former-stage horizontal filter 75.0% 7: Horizontal filter 12.5% to former-stage horizontal filter 87.5% 8: Former-stage horizontal filter 100% 9 to 15: Setting prohibited
7 to 4	VSEL_MIX_Y [3:0]	0000	R/W	Mixing Ratio of Signal after Passing Vertical Filter to Signal after Passing Former-Stage Horizontal/Vertical Filter 0: Vertical filter 100.0% 1: Vertical filter 87.5% to former-stage horizontal/vertical filter 12.5% 2: Vertical filter 75.0% to former-stage horizontal/vertical filter 25.0% 3: Vertical filter 62.5% to former-stage horizontal/vertical filter 37.5% 4: Vertical filter 50.0% to former-stage horizontal/vertical filter 50.0% 5: Vertical filter 37.5% to former-stage horizontal/vertical filter 62.5% 6: Vertical filter 25.0% to former-stage horizontal/vertical filter 75.0% 7: Vertical filter 12.5% to former-stage horizontal/vertical filter 87.5% 8: Former-stage horizontal/vertical filter 100% 9 to 15: Setting prohibited
3 to 0	HVSEL_MIX_Y [3:0]	0000	R/W	Mixing Ratio of Signal after Passing Horizontal/Vertical Filter to Signal after Passing Former-Stage Horizontal/Vertical Filter 0: Horizontal/vertical filter 100.0% 1: Horizontal/vertical filter 87.5% to former-stage horizontal/vertical filter 12.5% 2: Horizontal/vertical filter 75.0% to former-stage horizontal/vertical filter 25.0% 3: Horizontal/vertical filter 62.5% to former-stage horizontal/vertical filter 37.5% 4: Horizontal/vertical filter 50.0% to former-stage horizontal/vertical filter 50.0% 5: Horizontal/vertical filter 37.5% to former-stage horizontal/vertical filter 62.5% 6: Horizontal/vertical filter 25.0% to former-stage horizontal/vertical filter 75.0% 7: Horizontal/vertical filter 12.5% to former-stage horizontal/vertical filter 87.5% 8: Former-stage horizontal/vertical filter 100% 9 to 15: Setting prohibited

## (1) Two-Dimensional Filter Mixing Select Control

DET2\_ON selects two-dimensional filter mixing.

For details, refer to section 30.5.5 (5), Horizontal and Vertical Signal Mixing Block.



### (2) Control of Mixing Ratio of Signal after Passing Horizontal Filter to Signal after Passing Former-Stage Horizontal Filter

HSEL\_MIX\_Y controls the mixing ratio of the signal after passing a horizontal filter to the signal after passing the former-stage horizontal filter.

For details, refer to section 30.5.5 (5), Horizontal and Vertical Signal Mixing Block.

### (3) Control of Mixing Ratio of Signal after Passing Vertical Filter to Signal after Passing Former-Stage Horizontal/Vertical Filter

VSEL\_MIX\_Y controls the mixing ratio of the signal after passing a vertical filter to the signal after passing the former-stage horizontal/vertical filter.

For details, refer to section 30.5.5 (5), Horizontal and Vertical Signal Mixing Block.

### (4) Control of Mixing Ratio of Signal after Passing Horizontal/Vertical Filter to Signal after Passing Former-Stage Horizontal/Vertical Filter

HVSEL\_MIX\_Y controls the mixing ratio of the signal after passing a horizontal/vertical filter to the signal after passing the former-stage horizontal/vertical filter.

For details, refer to section 30.5.5 (5), Horizontal and Vertical Signal Mixing Block.

## 30.4.59 Y/C Separation Control Register 11 (YCSCR11)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	V_Y_LEVEL[8:0]								
Initial value:	1	1	0	1	1	0	0	0	0	0	0	0	0	0	1	1
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15, 14	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
13	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
12, 11	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
10, 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8 to 0	V_Y_LEVEL [8:0]	00000011	R/W	Vertical Luminance Detection Level for Correlation Detection Filter The luminance is detected when lower than the set value.

#### (1) Vertical Luminance Detection Level for Correlation Detection Filter

V\_Y\_LEVEL[8:0] select the vertical luminance detection level for correlation detection filter.

Be sure to set 0 to all the bits in this field when this module is in use.

### 30.4.60 Y/C Separation Control Register 12 (YCSCR12)

During two-dimensional Y/C separation, the horizontal bandwidth can be further narrowed using the cascade horizontal BPF after horizontal BPF, vertical BPF, and horizontal/vertical BPF are switched.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DET2_MIX_C[3:0]				DET2_MIX_Y[3:0]				—	—	—	—	FIL2_MODE_2D [1:0]	—	FIL2_NARROW_2D	
Initial value:	0	0	0	0	0	1	1	0	0	0	0	0	0	1	0	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 12	DET2_MIX_C [3:0]	0000	R/W	Mixing Ratio of C Signal after Passing Horizontal/Vertical Adaptive Filter to Signal after Passing Correlation Detection Filter (set 0 when DET2_ON = 0) 0: Horizontal/vertical adaptive filter 100.0% 1: Horizontal/vertical adaptive filter 87.5% to correlation detection filter 12.5% 2: Horizontal/vertical adaptive filter 75.0% to correlation detection filter 25.0% 3: Horizontal/vertical adaptive filter 62.5% to correlation detection filter 37.5% 4: Horizontal/vertical adaptive filter 50.0% to correlation detection filter 50.0% 5: Horizontal/vertical adaptive filter 37.5% to correlation detection filter 62.5% 6: Horizontal/vertical adaptive filter 25.0% to correlation detection filter 75.0% 7: Horizontal/vertical adaptive filter 12.5% to correlation detection filter 87.5% 8: Correlation detection filter 100% 9 to 15: Setting prohibited
11 to 8	DET2_MIX_Y [3:0]	0110	R/W	Mixing Ratio of C Signal for Y Generation after Passing Horizontal/Vertical Adaptive Filter to Signal after Passing Correlation Detection Filter (set 0 when DET2_ON = 0) 0: Horizontal/vertical adaptive filter 100.0% 1: Horizontal/vertical adaptive filter 87.5% to correlation detection filter 12.5% 2: Horizontal/vertical adaptive filter 75.0% to correlation detection filter 25.0% 3: Horizontal/vertical adaptive filter 62.5% to correlation detection filter 37.5% 4: Horizontal/vertical adaptive filter 50.0% to correlation detection filter 50.0% 5: Horizontal/vertical adaptive filter 37.5% to correlation detection filter 62.5% 6: Horizontal/vertical adaptive filter 25.0% to correlation detection filter 75.0% 7: Horizontal/vertical adaptive filter 12.5% to correlation detection filter 87.5% 8: Correlation detection filter 100% 9 to 15: Setting prohibited
7 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3, 2	FIL2_MODE_2D [1:0]	01	R/W	Two-Dimensional Cascade/TAKE-OFF Filter Mode Select 0: Bypass 1: Cascade filter 2: TAKE-OFF filter 3: Setting prohibited
1	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
0	FIL2_NARROW_2D	1	R/W	Two-Dimensional Cascade Filter Select 0: Bypass 1: 17 TAP

#### (1) Mixing Ratio of C Signal after Passing Horizontal/Vertical Adaptive Filter to Signal after Passing Correlation Detection Filter

DET2\_MIX\_C controls the mixing ratio of the chroma signal (after adaptation) to the signal after passing the correlation detection filter.

For details, refer to section 30.5.5 (6), Correlation Detection Value Mixing Block.

## (2) Mixing Ratio of C Signal for Y Generation after Passing Horizontal/Vertical Adaptive Filter to Signal after Passing Correlation Detection Filter

For details, refer to section 30.5.5 (6), Correlation Detection Value Mixing Block.

## (3) Two-Dimensional Cascade/TAKE-OFF Filter Mode Select

FIL2\_MODE\_2D selects the two-dimensional cascade/TAKE-OFF filter mode.

For details, refer to section 30.5.5 (8), Cascade Filter Block.

## (4) Two-Dimensional Cascade Filter Select

FIL2\_NARROW\_2D selects the two-dimensional cascade filter.

For details, refer to section 30.5.5 (8), Cascade Filter Block.

### 30.4.61 Digital Clamp Control Register 9 (DCPCR9)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	CLP_HOLD_ON_Y	CLP_HOLD_ON_CB	CLP_HOLD_ON_CR	—	—	—	—	—	—	—	—	—	—
Initial value:	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
12	CLP_HOLD_ON_Y	1	R/W	Clamp Data Hold Processing ON/OFF (Y) 0: Hold processing ON 1: Hold processing OFF
11	CLP_HOLD_ON_CB	1	R/W	Clamp Data Hold Processing ON/OFF (Cb) 0: Hold processing ON 1: Hold processing OFF
10	CLP_HOLD_ON_CR	1	R/W	Clamp Data Hold Processing ON/OFF (Cr) 0: Hold processing ON 1: Hold processing OFF
9 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

#### (1) Clamp Data Hold Processing ON/OFF Control (Y)

CLP\_HOLD\_ON\_Y selects ON/OFF of hold processing for Y signal clamp data.

Be sure to set 0 to this bit when this module is in use.

#### (2) Clamp Data Hold Processing ON/OFF Control (Cb)

CLP\_HOLD\_ON\_CB selects ON/OFF of hold processing for Cb signal clamp data.

Be sure to set 0 to this bit when this module is in use.

#### (3) Clamp Data Hold Processing ON/OFF Control (Cr)

CLP\_HOLD\_ON\_CR selects ON/OFF of hold processing for Cr signal clamp data.

Be sure to set 0 to this bit when this module is in use.

### 30.4.62 Chroma Filter TAP Coefficient (WA\_F0 to WA\_F8) Registers for Y/C Separation (YCTWA\_F0 to YCTWA\_F8)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	FIL2_2D_WA_F0 to FIL2_2D_WA_F8[12:0]												
Initial value:	0	0	0	*	*	*	*	*	*	*	*	*	*	*	*	*
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 0	FIL2_2D_WA_F0 to FIL2_2D_WA_F8[12:0]	*	R/W	Two-Dimensional Cascade Broadband (3.58/4.43/SECAM-DR)/TAKE-OFF Filter TAP Coefficients 0 to 8 [12]: Sign [11:0]: Absolute value * Initial values: FIL2_2D_WA_F0: H'0018 FIL2_2D_WA_F1: H'002C FIL2_2D_WA_F2: H'0014 FIL2_2D_WA_F3: H'1034 FIL2_2D_WA_F4: H'1080 FIL2_2D_WA_F5: H'1080 FIL2_2D_WA_F6: H'100C FIL2_2D_WA_F7: H'0084 FIL2_2D_WA_F8: H'00C8

#### (1) Two-Dimensional Cascade Broadband (3.58/4.43/SECAM-DR)/TAKE-OFF Filter TAP Coefficients 0 to 8 Control

FIL2\_2D\_WA\_F0 to FIL2\_2D\_WA\_F8[12:0] control two-dimensional cascade broadband (3.58/4.43/SECAM-DR)/TAKE-OFF filter TAP coefficients 0 to 8.

The transfer function is defined as follows:

$$H(z) = \{F0(z^{-8} + z^{+8}) + F1(z^{-7} + z^{+7}) + F2(z^{-6} + z^{+6}) + F3(z^{-5} + z^{+5}) + F4(z^{-4} + z^{+4}) + F5(z^{-3} + z^{+3}) + F6(z^{-2} + z^{+2}) + F7(z^{-1} + z^{+1}) + F8(z^0)\} / 1024$$

The coefficient value is represented using the MSB for a sign and the other bits for the effective value in the absolute value.

**Table 30.31 TAP Coefficient Settings**

Most Significant Bit	Other Than MSB	Setting Value
0	0 to 4095	+0 to +4095
1	0 to 4095	-0 to -4095

For the recommended setting value for each filter, see section 30.5.5 (8), Cascade Filter Block.

### 30.4.63 Chroma Filter TAP Coefficient (WB\_F0 to WB\_F8) Registers for Y/C Separation (YCTWB\_F0 to YCTWB\_F8)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	FIL2_2D_WB_F0 to FIL2_2D_WB_F8[12:0]												
Initial value:	0	0	0	*	*	*	*	*	*	*	*	*	*	*	*	*
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 0	FIL2_2D_WB_F0 to FIL2_2D_WB_F8[12:0]	*	R/W	Two-Dimensional Cascade Broadband (SECAM-DB) Filter TAP Coefficients 0 to 8 [12]: Sign [11:0]: Absolute value * Initial values: FIL2_2D_WB_F0: H'100C FIL2_2D_WB_F1: H'0028 FIL2_2D_WB_F2: H'003C FIL2_2D_WB_F3: H'000C FIL2_2D_WB_F4: H'1068 FIL2_2D_WB_F5: H'109C FIL2_2D_WB_F6: H'1040 FIL2_2D_WB_F7: H'0078 FIL2_2D_WB_F8: H'00D0

(1) Two-Dimensional Cascade Broadband (SECAM-DB) Filter TAP Coefficients 0 to 8 Control  
FIL2\_2D\_WB\_F0 to FIL2\_2D\_WB\_F8[12:0] control two-dimensional cascade broadband (SECAM-DB) filter TAP coefficients 0 to 8.

The transfer function is defined as follows:

$$H(z) = \{F0(z^{-8} + z^{+8}) + F1(z^{-7} + z^{+7}) + F2(z^{-6} + z^{+6}) + F3(z^{-5} + z^{+5}) + F4(z^{-4} + z^{+4}) + F5(z^{-3} + z^{+3}) + F6(z^{-2} + z^{+2}) + F7(z^{-1} + z^{+1}) + F8(z^0)\} / 1024$$

The coefficient value is represented using the MSB for a sign and the other bits for the effective value in the absolute value.

**Table 30.32 TAP Coefficient Settings**

Most Significant Bit	Other Than MSB	Setting Value
0	0 to 4095	+0 to +4095
1	0 to 4095	-0 to -4095

For the recommended setting value for each filter, see section 30.5.5 (8), Cascade Filter Block.

### 30.4.64 Chroma Filter TAP Coefficient (NA\_F0 to NA\_F8) Registers for Y/C Separation (YCTNA\_F0 to YCTNA\_F8)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	FIL2_2D_NA_F0 to FIL2_2D_NA_F8[12:0]												
Initial value:	0	0	0	*	*	*	*	*	*	*	*	*	*	*	*	*
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 0	FIL2_2D_NA_F0 to FIL2_2D_NA_F8[12:0]	*	R/W	Two-Dimensional Cascade Narrowband (3.58/4.43/SECAM-DR) Filter TAP Coefficients 0 to 8 [12]: Sign [11:0]: Absolute value * Initial values: FIL2_2D_NA_F0: H'0018 FIL2_2D_NA_F1: H'002C FIL2_2D_NA_F2: H'0014 FIL2_2D_NA_F3: H'1034 FIL2_2D_NA_F4: H'1080 FIL2_2D_NA_F5: H'1080 FIL2_2D_NA_F6: H'100C FIL2_2D_NA_F7: H'0084 FIL2_2D_NA_F8: H'00C8

#### (1) Two-Dimensional Cascade Narrowband (3.58/4.43/SECAM-DR) Filter TAP Coefficients 0 to 8 Control

FIL2\_2D\_NA\_F0 to FIL2\_2D\_NA\_F8[12:0] control two-dimensional cascade narrowband (3.58/4.43/SECAM-DR) filter TAP coefficients 0 to 8.

The transfer function is defined as follows:

$$H(z) = \{F0(z^{-8} + z^{+8}) + F1(z^{-7} + z^{+7}) + F2(z^{-6} + z^{+6}) + F3(z^{-5} + z^{+5}) + F4(z^{-4} + z^{+4}) + F5(z^{-3} + z^{+3}) + F6(z^{-2} + z^{+2}) + F7(z^{-1} + z^{+1}) + F8(z^0)\} / 1024$$

The coefficient value is represented using the MSB for a sign and the other bits for the effective value in the absolute value.

**Table 30.33 TAP Coefficient Settings**

Most Significant Bit	Other Than MSB	Setting Value
0	0 to 4095	+0 to +4095
1	0 to 4095	-0 to -4095

For the recommended setting value for each filter, see section 30.5.5 (8), Cascade Filter Block.

### 30.4.65 Chroma Filter TAP Coefficient (NB\_F0 to NB\_F8) Registers for Y/C Separation (YCTNB\_F0 to YCTNB\_F8)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	FIL2_2D_NB_F0 to FIL2_2D_NB_F8[12:0]												
Initial value:	0	0	0	*	*	*	*	*	*	*	*	*	*	*	*	*
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 0	FIL2_2D_NB_F0 to FIL2_2D_NB_F8[12:0]	*	R/W	Two-Dimensional Cascade Narrowband (SECAM-DB) Filter TAP Coefficients 0 to 8 [12]: Sign [11:0]: Absolute value * Initial values: FIL2_2D_NB_F0: H'1438 FIL2_2D_NB_F1: H'0AF0 FIL2_2D_NB_F2: H'1CEC FIL2_2D_NB_F3: H'065C FIL2_2D_NB_F4: H'05A4 FIL2_2D_NB_F5: H'1CEC FIL2_2D_NB_F6: H'085C FIL2_2D_NB_F7: H'0178 FIL2_2D_NB_F8: H'1568

(1) Two-Dimensional Cascade Narrowband (SECAM-DB) Filter TAP Coefficients 0 to 8 Control  
FIL2\_2D\_NB\_F0 to FIL2\_2D\_NB\_F8[12:0] control two-dimensional cascade narrowband (SECAM-DB) filter TAP coefficients 0 to 8.

The transfer function is defined as follows:

$$H(z) = \{F0(z^{-8} + z^{+8}) + F1(z^{-7} + z^{+7}) + F2(z^{-6} + z^{+6}) + F3(z^{-5} + z^{+5}) + F4(z^{-4} + z^{+4}) + F5(z^{-3} + z^{+3}) + F6(z^{-2} + z^{+2}) + F7(z^{-1} + z^{+1}) + F8(z^0)\} / 1024$$

The coefficient value is represented using the MSB for a sign and the other bits for the effective value in the absolute value.

**Table 30.34 TAP Coefficient Settings**

Most Significant Bit	Other Than MSB	Setting Value
0	0 to 4095	+0 to +4095
1	0 to 4095	-0 to -4095

For the recommended setting value for each filter, see section 30.5.5 (8), Cascade Filter Block.

### 30.4.66 Luminance (Y) Signal Gain Control Register (YGAINCR)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	Y_GAIN2[9:0]									
Initial value:	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	Y_GAIN2[9:0]	1000000000	R/W	Y Signal Gain Coefficient (0 = 0 times, 512 = 1.0 times, 1023 ≈ 2.0 times)

#### (1) Y Signal Output Gain Control

Y\_GAIN2 controls the Y signal output gain.

Y signal output = Y signal after decoding × (Y\_GAIN2/512)

### 30.4.67 Color Difference (Cb) Signal Gain Control Register (CBGAINCR)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CB_GAIN2[9:0]									
Initial value:	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	CB_GAIN2[9:0]	1000000000	R/W	Cb Signal Gain Coefficient (0 = 0 times, 512 = 1.0 times, 1023 ≈ 2.0 times)

#### (1) Cb Signal Output Gain Control

CB\_GAIN2 controls the Cb signal output gain.

Cb signal output = Cb signal after decoding × (CB\_GAIN2/512)



### 30.4.68 Color Difference (Cr) Signal Gain Control Register (CRGAINCR)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CR_GAIN2[9:0]									
Initial value:	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	CR_GAIN2[9:0]	1000000000	R/W	Cr Signal Gain Coefficient (0 = 0 times, 512 = 1.0 times, 1023 ≈ 2.0 times)

#### (1) Cr Signal Output Gain Control

CR\_GAIN2 controls the Cr signal output gain.

Cr signal output = Cr signal after decoding × (CR\_GAIN2/512)

### 30.4.69 PGA Register Update (PGA\_UPDATE)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PGA_VEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	PGA_VEN	1	R/W	PGACR Register V Update Enable 1: Enable 0: Disable

#### (1) PGACR Register V Update Enable

PGA\_VEN enables or disables V update of PGACR register.

### 30.4.70 PGA Control Register (PGACR)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	PGA_GAIN_SEL	PGA_GAIN[5:0]					—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	1	0	1	0	1	0	0	0	0	1	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	PGA_GAIN_SEL	0	R/W	PGA Switch 0: Automatic (AGC) 1: Manual (Refer to the PGA_GAIN description below.)
13 to 8	PGA_GAIN[5:0]	010101	R/W	PGA Gain 0 (0.8 Vpp) to 63 (1.6 Vpp)
7 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
2 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note: • All the bits in this register are updated when the vertical sync signal is asserted with PGA\_VEN in PGA\_UPDATE being 1.

#### (1) PGA Switch

When PGA\_GAIN\_SEL is 0 and ADCCR1.AGCMODE is 1, the AGC-controlled value is reflected on the PGA gain. When PGA\_GAIN\_SEL is 1, the PGA\_GAIN value is directly reflected on the PGA gain. In this case, the ADCCR1.AGCMODE setting is invalid.

Setting ADCCR1.AGCMODE to 0 and PGA\_GAIN\_SEL to 0 simultaneously is prohibited.

#### (2) PGA Gain

When PGA\_GAIN\_SEL is 1, the PGA\_GAIN value is reflected on the PGA gain. One of 64 levels of gain values can be set for the PGA of this LSI.

### 30.4.71 ADC Control Register 2 (ADCCR2)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ADC_VINSEL
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	ADC_VINSEL	0	R/W	Input Pin Control 0: VIN1 input 1: VIN2 input

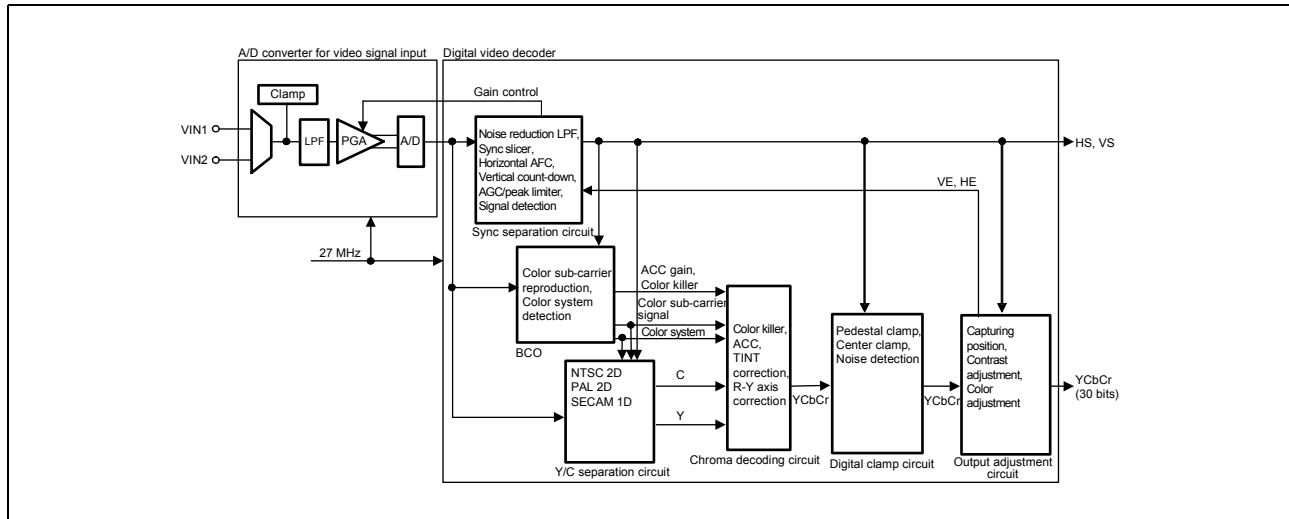
#### (1) Input Pin Control

ADC\_VINSEL selects the pin for inputting composite video signals.

## 30.5 Operation

### 30.5.1 Overview

This module decodes composite video signals (CVBS) and separates them into horizontal/vertical sync signals, luminance signals (Y), and color difference signals (Cb/Cr). Supported color systems are NTSC, PAL, and SECAM. This module consists of an A/D converter for video signal input, sync separator circuit, burst controlled oscillator (BCO), a Y/C separator circuit, a chroma decoding circuit, a digital clamp circuit, and an output gain adjustment circuit. Figure 30.26 shows an overall block diagram.



**Figure 30.26 Overall Block Diagram**

#### (1) A/D Converter for Video Signal Input

The A/D converter processes the composite video signal (CVBS) using the sync tip clamp block, the low-pass filter (LPF), and the programmable gain amplifier (PGA) and then A/D-converts the signal. The composite video signals from either VIN1 or VIN2 pin are selected.

#### (2) Sync Separator Circuit

The sync separator circuit extracts the horizontal and vertical sync signals from the composite video signal. This circuit also detects the amplitude of the sync signals and automatically adjusts the PGA gain (Automatic Gain Control = AGC).

#### (3) Burst Controlled Oscillator (BCO)

The BCO extracts the color burst signal from the composite video signal and reproduces the color sub-carrier signal required for color demodulation. The BCO also acquires phase and frequency information from the color burst signal and detects the color system used.

#### (4) Y/C Separator Circuit

The Y/C separator circuit separates the composite video signal of the NTSC, PAL, or SECAM format into the Y and C signals. Two-dimensional adaptive separation is used for NTSC and PAL and one-dimensional separation for SECAM.

#### (5) Chroma Decoding Circuit

The chroma decoding circuit demodulates the C signal extracted by the Y/C separator circuit into the Cb/Cr signal. This circuit has the automatic color control function (ACC), in which the amplitude of the color burst signal is detected to adjust the color gain automatically and also has the TINT/R-Y axis correction function, in which hue is adjusted at demodulation.

### (6) Digital Clamp Circuit

The digital clamp circuit provides pedestal clamp for the Y signals and center clamp for the Cb/Cr signals at any position. This circuit also detects the amount of noise using the autocorrelation function.

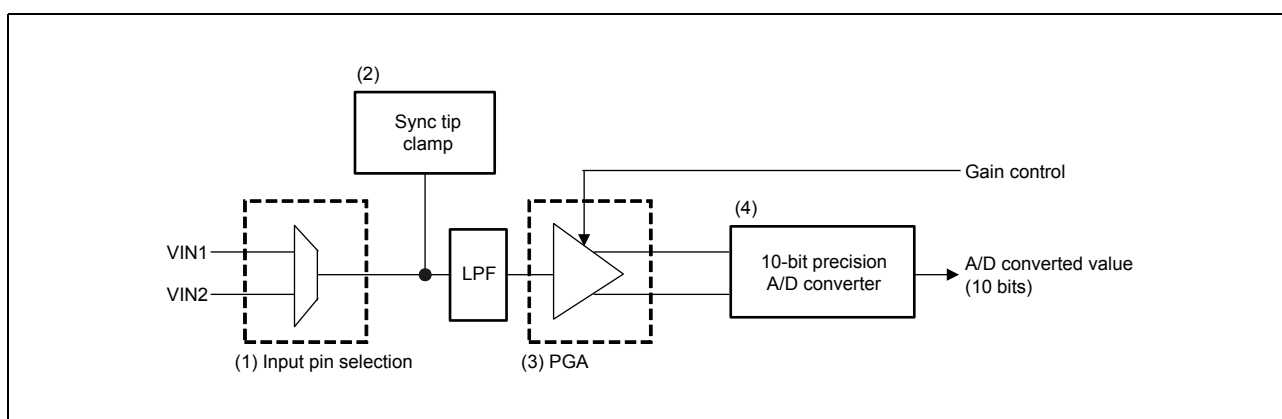
### (7) Output Adjustment Circuit

The output adjustment circuit sets the capturing position and adjusts the contrast and color.

## 30.5.2 A/D Converter for Video Signal Input

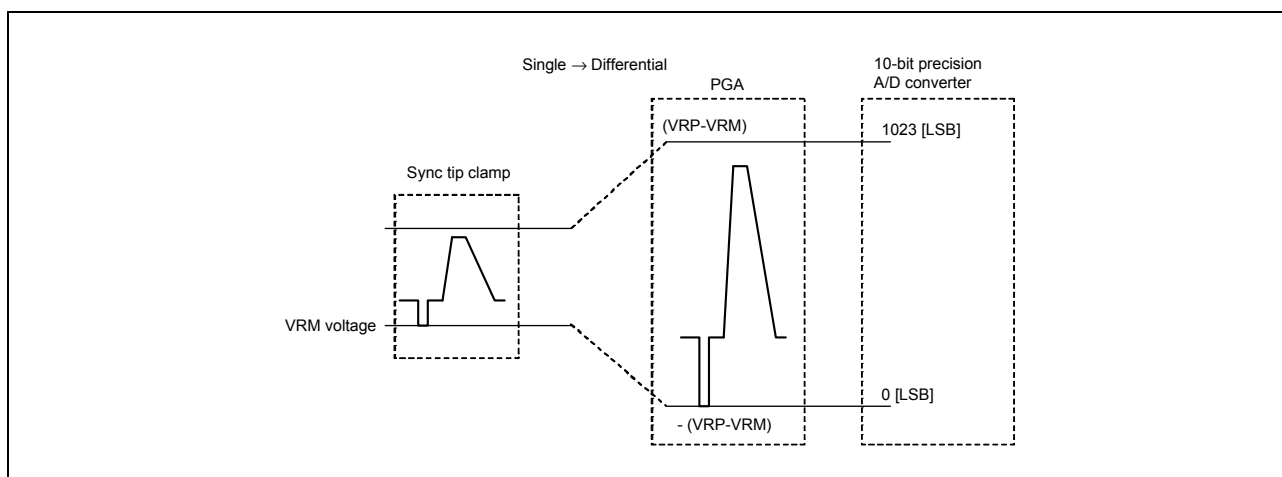
The A/D converter processes the composite video signal (CVBS) using the sync tip clamp block, the low-pass filter (LPF), and the programmable gain amplifier (PGA) and then A/D-converts the signal.

Figure 30.27 shows the block diagram of the A/D converter for video signal input.



**Figure 30.27** Block Diagram of A/D Converter for Video Signal Input

Figure 30.28 shows the waveforms when a video signal is A/D-converted.



**Figure 30.28** A/D Conversion Image

### (1) Input Pin Selection Block

The input pin selection block selects either the VIN1 or VIN2 pin for inputting the signal according to the ADCCR2.ADC\_VINSEL setting.

### (2) Sync Tip Clamp Block

The sync tip clamp block clamps the sync tip level to about 0.5 V.

### (3) Programmable Gain Amplifier (PGA)

The PGA adjusts the gain so that the input video signal voltage ( $\approx 0.8 V_{pp}$  to  $1.6 V_{pp}$ ) should be the level to be input to the A/D converter ( $\approx 2.0 V_{pp}$ ). One of 64 levels of gain values can be set. One level corresponds to gain of 0.1 dB (typ.). The minimum gain is 0 dB (typ.) and maximum gain is 6.02 dB (typ.).

Table 30.35 shows the PGA gain setting and gain values.

**Table 30.35 PGA Gain Setting and Gain Values (dB)**

PGA Gain Setting	Input Range (Vpp)	Gain Value (dB)
0	1.600	0
1	1.587	0.069
2	1.575	0.138
3	1.562	0.209
4	1.549	0.280
5	1.537	0.351
6	1.524	0.423
7	1.511	0.496
8	1.498	0.569
9	1.486	0.643
:	:	:
59	0.851	5.48
60	0.838	5.61
61	0.825	5.74
62	0.813	5.88
63	0.800	6.02

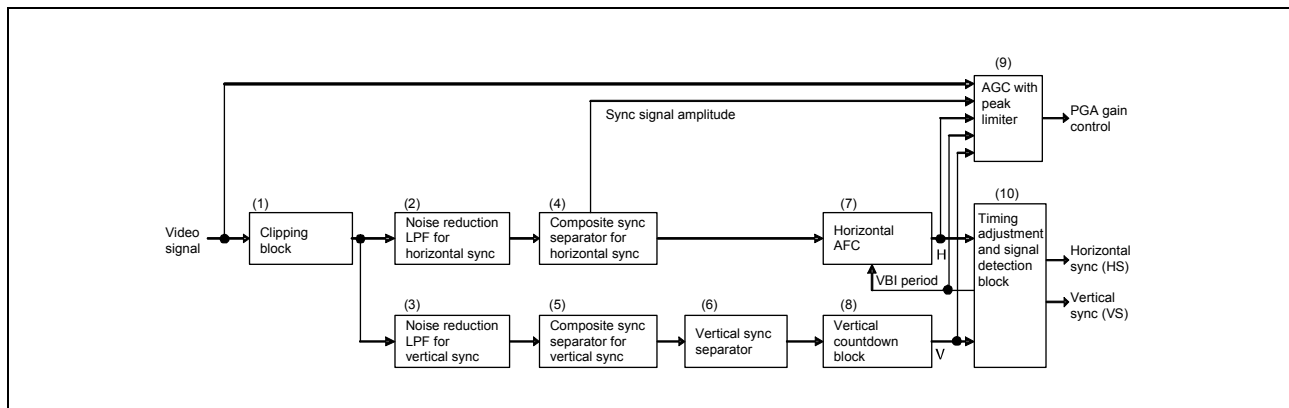
The PGA gain can be set using PGACR.PGA\_GAIN[5:0] with PGACR.PGA\_GAIN\_SEL = 1. When the AGC function is on (ADCCR1.AGCMODE = 1), the gain is automatically set.

### (4) 10-Bit Precision A/D Converter

The 10-bit precision A/D converter receives the gain-adjusted video signal from the PGA and A/D-converts the signal. The converter has 10-bit resolution and performs sampling at 27 MHz, which is the frequency of the clock signal input via VIDEO\_X1 or VIDEO\_X2.

### 30.5.3 Sync Separator Circuit

The sync separator circuit extracts the horizontal and vertical sync signals from the composite video signal. This circuit also detects the amplitude of the sync signals and automatically adjusts the PGA gain (Automatic Gain Control = AGC). Figure 30.29 shows the block diagram of the sync separator circuit.



**Figure 30.29** Block Diagram of Sync Separator Circuit

#### (1) Clipping Block

The clipping block clips the high tone component of the video signal to reduce amplitude-dependency of the video signal. The specific clipping level can be set using `SYNSCR3.SSCLIPSEL[3:0]`. The clipping level should be within the range so that the composite sync signal component should not be deteriorated (should be detectable).

#### (2) Noise Reduction Low-Pass Filter (LPF) for Horizontal Sync

The noise reduction LPF reduces the superimposed noise on the video signals before composite sync signal separation. The LPFs can be separately set for the horizontal sync and vertical sync. The cutoff frequency of the horizontal sync LPF can be set using `SYNSCR1.LPFHSYNC[2:0]`. The cutoff frequency should be within the range so that the composite sync signal component should not be deteriorated (should be detectable).

#### (3) Noise Reduction Low-Pass Filter (LPF) for Vertical Sync

The noise reduction LPF reduces the superimposed noise on the video signals before composite sync signal separation. The LPFs can be separately set for the horizontal sync and vertical sync. The cutoff frequency of the vertical sync LPF can be set using `SYNSCR1.LPFVSYNC[2:0]`. The cutoff frequency should be within the range so that the composite sync signal component should not be deteriorated (should be detectable).

#### (4) Composite Sync Separator for Horizontal Sync

The composite sync separator separates the composite sync signal from the video signal according to the slice level. The composite sync slice levels can be separately set for the horizontal and vertical sync signals. The slice level can be set either automatically or manually according to the `SYNSCR1.SLICERMODE_H[1:0]` setting. When automatic setting is used, the slice level is automatically set according to the `SYNSCR1.SLICERMODE_H[1:0]`, `SYNSCR2.SYNCMAXDUTY_H[5:0]`, and `SYNSCR2.SYNCMINDUTY_H[5:0]` setting. The slice level detection speed can be set using `SYNSCR1.VELOCITYSHIFT_H[3:0]`. When manual setting is used, the slice level is determined by `SYNSCR3.CSYNCSLICE_H[9:0]`.

#### (5) Composite Sync Separator for Vertical Sync

The composite sync separator separates the composite sync signal from the video signal according to the slice level. The composite sync slice levels can be separately set for the horizontal and vertical sync signals. The slice level can be set either automatically or manually according to the `SYNSCR1.SLICERMODE_V[1:0]` setting.

When automatic setting is used, the slice level is automatically set according to the SYNSCR1.SLICERMODE\_V[1:0], SYNSCR4.SYNCMAXDUTY\_V[5:0], and SYNSCR4.SYNCMINDUTY\_V[5:0] setting.

When manual setting is used, the slice level is determined by SYNSCR5.CSYNCSLICE\_V[9:0].

### (6) Vertical Sync Separator

The vertical sync separator extracts the vertical sync signal from the composite video signal separated using (5) above. The threshold value for separating the vertical sync signal can be set using SYNSCR5.VSYNCSLICE[4:0]. The value should be set depending on the serration pulse width of each video signal format.

### (7) Horizontal Automatic Frequency Control (AFC) Block

The horizontal AFC, which is a digital PLL, extracts the horizontal sync signal from the composite video signal separated using (4) above. The AFC removes the pseudo horizontal sync signal and interpolates the incomplete horizontal sync signal to generate a stable horizontal sync signal.

The center frequency and lock range of the horizontal AFC can be set using HAFCCR1.HAFCTYP[9:0], HAFCCR2.HAFCMAX[9:0], and HAFCCR3.HAFCMIN[9:0]. If the horizontal AFC is locked, VSYNCSR.FHLOCK becomes 1; and if the horizontal AFC is unlocked, VSYNCSR.FHLOCK becomes 0. The horizontal AFC oscillation cycle can be checked by reading HSYNCSR.FHCOUNT[16:1] and VSYNCSR.FHCOUNT[0].

The loop gain (response speed) of the horizontal AFC can be set using HAFCCR1.HAFCCGAIN[3:0]. As the speed is increased, the lockup time becomes shorter. However, the horizontal oscillation frequency will be more susceptible to noise. With HAFCCR3.HAFCMODE[1], AFCPFCR.PHDET\_FIX, and AFCPFCR.PHDET\_DIV[2:0], the loop gain can be reduced when S/N is low to prevent malfunction attributed to noise. Whether S/N is low or not can be checked with VSYNCSR.ISNOISY.

The loop gain during the vertical blanking period (VBI) can be set using HAFCCR2.HAFCSTART[3:0], HAFCCR3.HAFCEND[3:0], and HAFCCR3.HAFCMODE[0]. This is usually used to avoid malfunction in the VCR head switch.

### (8) Vertical Countdown Block

The vertical countdown block removes the pseudo vertical sync signal from the vertical sync signal separated using (6) above and interpolates the incomplete vertical sync signal to generate a stable vertical sync signal.

The oscillation cycle of the vertical countdown block can be set using VCDWCR1.VCDDEFAULT[1:0]. When set to 0, the input vertical sync signal is detected and the oscillation cycle is automatically set appropriately. The detection result of the input vertical sync signal is indicated by VSYNCSR.FVMODE. When set to 1, 50.00-Hz oscillation mode is set. Here, it is recommended to set VCDWCR1.NOVCD60 to 1 (60-Hz oscillation off) to avoid unexpected malfunction.

When set to 2 or 3, 59.94-/60.00-Hz oscillation mode is set. Here, it is recommended to set VCDWCR1.NOVCD50 to 1 (50-Hz oscillation off) to avoid unexpected malfunction.

The lock range of the vertical countdown block can be set using VCDWCR1.VCDWINDOW[5:0] and VCDWCR1.VCDOFFSET[4:0]. If the vertical countdown block is locked, VSYNCSR.FVLOCK becomes 1; and if the vertical countdown block is unlocked, VSYNCSR.FVLOCK becomes 0.

The cycle of the input vertical sync signal can be checked by reading VSYNCSR.FVCOUNT[7:0]. When the vertical sync signal input cannot be detected, VSYNCSR.NOSIGNAL is set to 1.

### (9) Automatic Gain Control (AGC) Block with Peak Limiter

The AGC block detects the amplitude of the sync signal and automatically controls the PGA gain to the target value. The AGC function is activated with ADCCR1.AGCMODE = 1.

- Gain Control according to Sync Amplitude

The target sync signal amplitude can be set using AGCCR1.AGCLEVEL[8:0] and AGCCR2.AGCPRECIS[5:0].

For example, when NTSC signals are quantized by the 10-bit A/D converter, the sync signal amplitude for the full range of the A/D converter can be provided by:

$$1023[\text{LSB}] \times (40[\text{IRE}] \div 173[\text{IRE}]) = 236.53179[\text{LSB}]$$

Therefore, 236[LSB] should be set to AGCCR1.AGCLEVEL[8:0]. The gain is fixed when it falls within the

following range.

Target value (= AGCCR1.AGCLEVEL[8:0])  $\pm$  AGCCR2.AGCPRECIS[5:0]

Whether the gain is fixed or not can be checked by reading AGCCSR2.AGCCONVERGE. The detected sync signal amplitude can be checked using SYNCSSR.SYNCDEPTH[9:0]. The AGC response speed can be set using AGCCR1.AGCRESPONSE[2:0]. As the speed is increased, the input signal is tracked more quickly; however, it will result in higher susceptibility to noise.

The currently set gain value can be checked using AGCCSR2.AGCGAIN[7:0]. The actual PGA gain setting can be roughly calculated as follows:

$$1.203008 \times (\text{AGCCSR2.AGCGAIN}[7:0] - 48)$$

For example, when AGCCSR2.AGCGAIN[7:0] is 64 (corresponding to  $\times 1$ ),

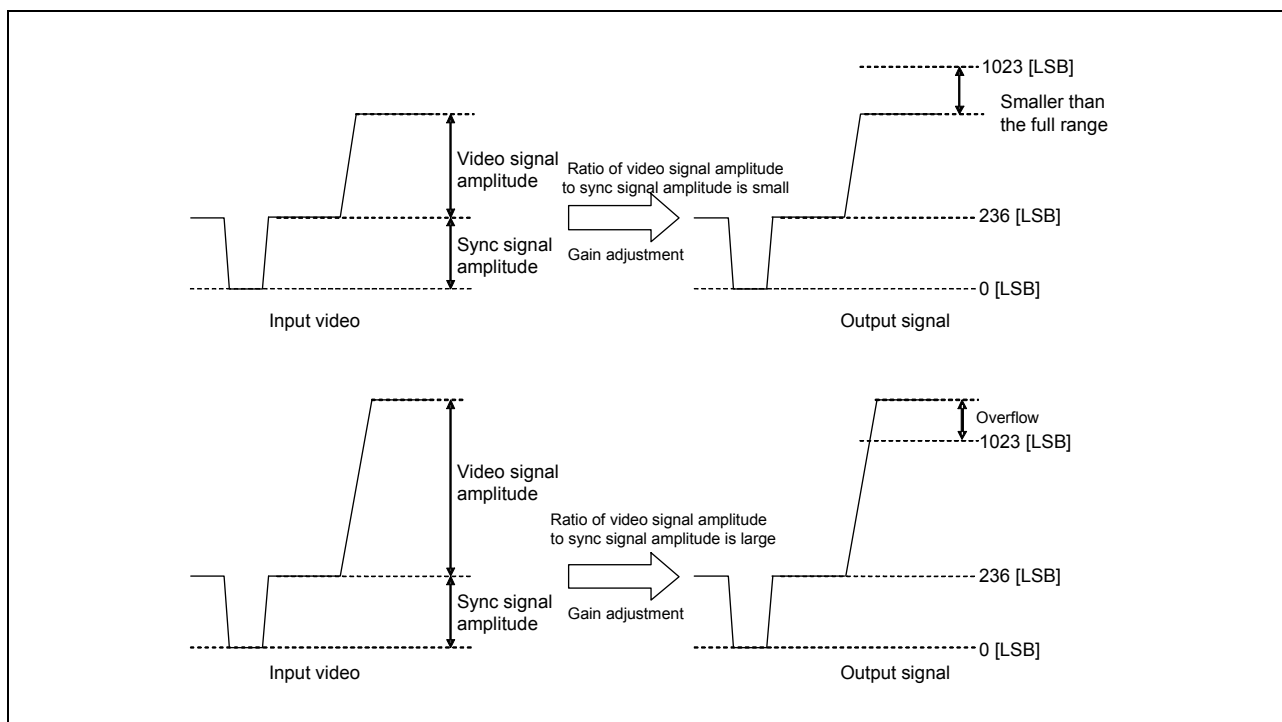
$$\text{PGA gain setting} = 1.203008 \times (64 - 48) \approx 19.24$$

As a result, the PGA gain setting is 19 or 20. One of 0 to 63 can be set as the PGA gain value.

The gain during vertical blanking period (VBI) can be set using AGCCR1.DOREDUCE and AGCCR1.NOREDUCE. The sync amplitude detection result during VBI can be read from SYNCSSR.ISREDUCED.

- Peak Limiter

The peak limiter works when the ratio of the video signal amplitude to the sync signal amplitude is inappropriate. If the ratio is smaller than expected, the PGA gain becomes smaller, and the video signal after gain adjustment becomes smaller than the full range of the A/D converter. Contrarily, if the ratio is larger than expected, the PGA gain becomes larger, and the video signal after gain adjustment becomes larger than the full range of the A/D converter.



**Figure 30.30** Cases in which Ratio of Video Signal Amplitude to Sync Signal Amplitude is Inappropriate

To deal with this problem, the peak limiter adjusts the PGA gain based on the sampled video signal peak value. The peak value used to control the gain can be set using PKLIMITCR.PEAKLEVEL[1:0]. When the peak value of the sampled video signal is smaller than the value set using PKLIMITCR.PEAKLEVEL[1:0], the gain is increased.

Contrarily, when the peak value of the sampled video signal is larger than the value set using PKLIMITCR.PEAKLEVEL[1:0] exceeding the maximum allowable value set using



PKLIMITCR.MAXPEAKSAMPLES[7:0], the gain is decreased.

The gain increase/decrease response speed and maximum compression ratio can be set using

PKLIMITCR.PEAKATTACK[1:0], PKLIMITCR.PEAKRELEASE[1:0], and PKLIMITCR.PEAKRATIO[1:0].

The number of pixels with the peak value larger than the value set using PKLIMITCR.PEAKLEVEL[1:0] can be checked with AGCCSR1.HIGHSAMPLES[7:0]; and the number of overflowing pixels (exceeding 1023[LSB]) can be checked with AGCCSR1.PEAKSAMPLES[7:0].

- Manual Setting

Manual setting of the PGA gain can be enabled by setting PGACR.PGA\_GAIN\_SEL to 1. Here, the value set using PGACR.PGA\_GAIN[5:0] is actually set as the PGA gain. When PGACR.PGA\_GAIN\_SEL is 1,

ADCCR1.AGCMODE setting is invalid. Setting PGACR.PGA\_GAIN\_SEL to 0 (automatic setting) and

ADCCR1.AGCMODE to 0 (AGC off) simultaneously is prohibited.

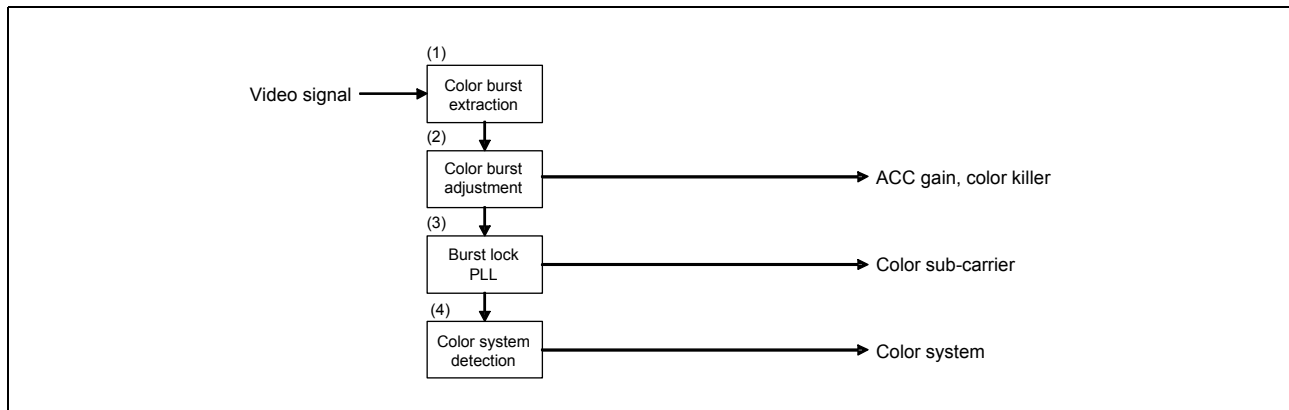
### (10) Timing Adjustment and Signal Detection Block

The timing adjustment and signal detection block adjusts the output timing of the horizontal and vertical sync signals generated using (7) and (8) above. This block also detects the field; whether the interlaced or progressive system is used can be checked with VSYNCSR.INTERLACED. If the field detection function is unstable, setting SYNSCR5.VSYNCDELAY to 1 may improve the function.

### 30.5.4 Burst Controlled Oscillator (BCO)

The BCO extracts the color burst signal from the composite video signal and reproduces the color sub-carrier signal required for color demodulation. It also acquires the phase and frequency information from the color burst signal and detects the color system used.

Figure 30.31 shows the block diagram of the BCO.



**Figure 30.31 Block Diagram of Burst Controlled Oscillator**

#### (1) Color Burst Extraction Block

The color burst extraction block extracts the color burst signal. The position of the color burst signal to be extracted can be adjusted using BTGPCR.BGPWIDTH[6:0] and BTGPCR.BGPSTART[7:0]. The extraction result of the color burst signal can be checked by reading CROMASR1.NOBURST.

#### (2) Color Burst Adjustment Block

The color burst adjustment block adjusts the amplitude of the extracted color burst signal. For details, refer to section 30.5.6 (1), Automatic Color Control (ACC) Block. This block also outputs the signal to turn on or off the color killer according to the amplitude of the input color burst signal. For details, refer to section 30.5.6 (2), Color Killer.

#### (3) Burst Lock PLL

The burst lock PLL is a digital PLL which reproduces the color sub-carrier signal from the adjusted color burst signal. The lock range of the burst lock PLL can be set using BTLCR.LOCKRANGE[1:0]. If the burst lock PLL is locked, CROMASR1.FSCLOCK becomes 1; and if unlocked, CROMASR1.FSCLOCK becomes 0.

The loop gain of the burst lock PLL can be set using BTLCR.LOOPGAIN[1:0] and BTLCR.LOCKLIMIT[1:0]. As the response speed is increased and the frequency search is started earlier, the lockup time becomes shorter. However, the PLL becomes unstable and unlocked more easily due to noise.

The S/N of the color burst signal can be checked using CROMASR2.LOCKLEVEL[7:0].

#### (4) Color System Detection Block

The color system detection block detects the color system of the input video signal based on the oscillation frequency of the burst lock PLL and phase information of the color burst signal. The color system can be detected using BTLCR.NONTSC358, BTLCR.NONTSC443, BTLCR.NOPALM, BTLCR.NOPALN, BTLCR.NOPAL443, and BTLCR.NOSECAM. Color system detection can be set to fully automatic, manual, or semi-automatic (detecting the specified color system only). If the detection result does not apply to any color system type, the color system selected with BTLCR.DEFAULTSYS[1:0] is assumed.

When an NTSC, PAL, or SECAM signal is detected, 1 is read from CROMASR2.ISNTSC, CROMASR2.ISPAL, or CROMASR2.ISSEAM, respectively. The currently used color system can be checked by reading CROMASR1.COLORSYS[1:0].

The color sub-carrier frequency can be checked by reading CROMASR1.FSCMODE.

### 30.5.5 Y/C Separator Circuit

The Y/C separator circuit separates the composite video signal of the NTSC, PAL, or SECAM format into the Y and C signals. Two-dimensional adaptive separation is used for NTSC and PAL and one-dimensional separation for SECAM. Figure 30.32 shows a block diagram of the Y/C separator circuit.

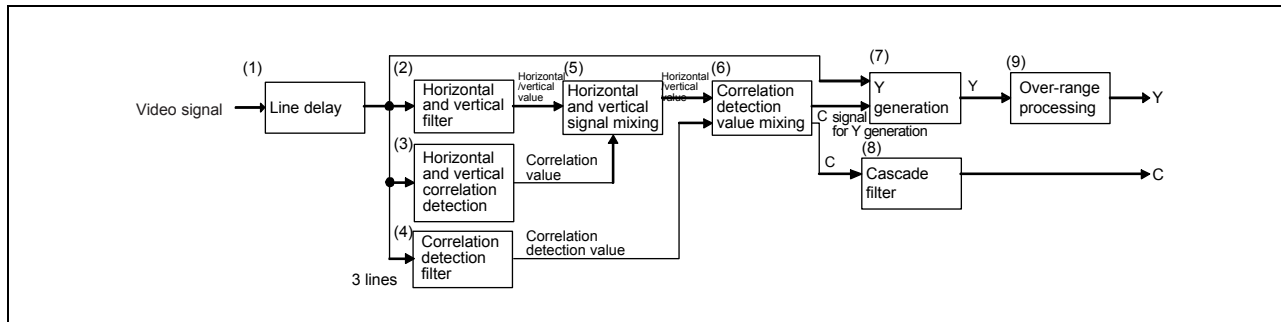


Figure 30.32 Block Diagram of Y/C Separator Circuit

Table 30.36 shows the operation of the Y/C separator circuit for each color format.

Table 30.36 Y/C Separation Operation for Each Color Format

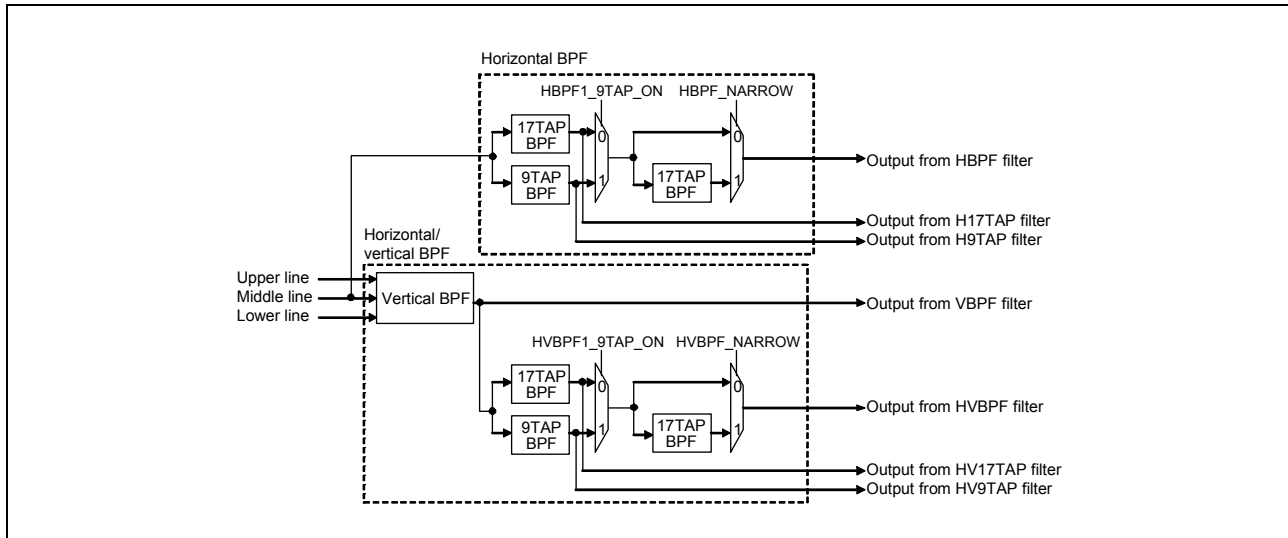
Color Format	YC Separation Operation
NTSC-3.58	Two dimensional
NTSC-4.43	Two dimensional
PAL-M	Two dimensional
PAL-N	Two dimensional
PAL-4.43	Two dimensional
SECAM	One dimensional

#### (1) Line Delay Block

In two-dimensional Y/C separation, three lines of data is required (directly adjacent three lines for NTSC and adjacent three lines on every second line for PAL). This block delays video signals to hold three lines of data.

## (2) Horizontal and Vertical Filter Block

In two-dimensional adaptive Y/C separation, the horizontal band pass filter (BPF), vertical band pass filter (BPF), and horizontal/vertical band pass filter (BPF) are adaptively switched according to the correlation between the horizontally-/vertically-adjacent pixels. This block processes the input signals using the horizontal BPF, vertical BPF, or horizontal/vertical BPF. In one-dimensional Y/C separation, only the horizontal BPF is used. Figure 30.33 shows the filter configuration.



**Figure 30.33** Horizontal and Vertical Filter Configuration

The horizontal BPF is composed of two stages. Either the 9-TAP or 17-TAP BPF is selected at the former stage. When YCSCR8.HBPF1\_9TAP\_ON/HVBPF1\_9TAP\_ON is 0, the 17-TAP BPF is selected; and when 1, the 9-TAP BPF is selected. Either bypass operation or the 17-TAP BPF is selected at the latter stage. When YCSCR8.HBPF\_NARROW/HVBPF\_NARROW is 0, bypass operation is selected; and when 1, the 17-TAP BPF is selected.

### (3) Horizontal and Vertical Correlation Detection Block

The horizontal and vertical correlation detection block detects the correlation value between the horizontal pixels, vertical pixels, and horizontal/vertical pixels. The value obtained by mixing the detected correlation value and the two-dimensional Y/C separation filter select coefficient is used for selecting the appropriate horizontal, vertical, or horizontal/vertical filter.

Table 30.37 shows the two-dimensional Y/C separation filter select coefficients.

**Table 30.37 Two-Dimensional Y/C Separation Filter Select Coefficients**

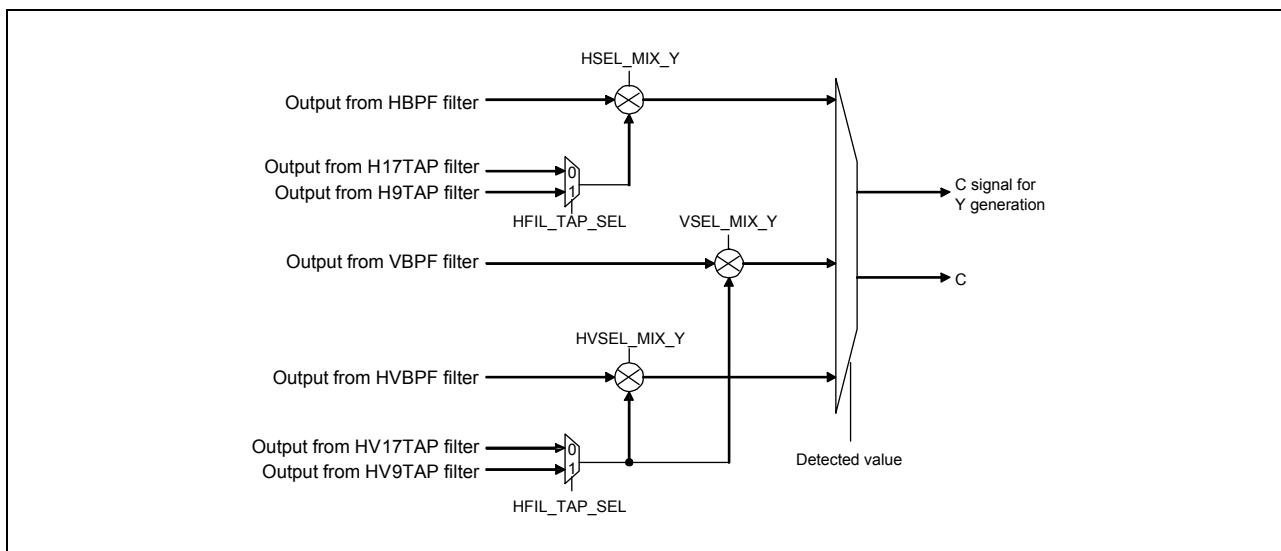
Category	Bit Name	Description	Bit Correlation
Vertical Y/C separation select coefficients	YCSCR5.K21A[5:0]	As the value becomes larger, the vertical BPF is applied to the narrower range.	There is correlation between these bits. When horizontal dot crawl is conspicuous, make the bit field value smaller (make the K24 value larger). However, when the value is too small (K24 is too large), vertical dot crawl is produced.
	YCSCR5.K22A[7:0]	As the value becomes larger, the vertical BPF is applied to the narrower range.	
	YCSCR7.K23A[3:0]	As the value becomes larger, the vertical BPF is applied to the narrower range.	
	YCSCR7.K24[4:0]	As the value becomes larger, the vertical BPF is applied to the wider range.	
	YCSCR6.K21B[5:0]	As the value becomes larger, the vertical BPF is applied to the narrower range.	There is correlation between these bits. When horizontal dot crawl is conspicuous, make the bit field value smaller. However, when the value is too small, vertical dot crawl is produced.
	YCSCR6.K22B[7:0]	As the value becomes larger, the vertical BPF is applied to the narrower range.	
	YCSCR7.K23B[3:0]	As the value becomes larger, the vertical BPF is applied to the narrower range.	
Horizontal Y/C separation select coefficients	YCSCR3.K11[5:0]	As the value becomes larger, the horizontal BPF is applied to the narrower range.	There is correlation between these bits. When vertical dot crawl is conspicuous, make the bit field value smaller. However, when the value is too small, horizontal dot crawl is produced.
	YCSCR3.K13[5:0]	As the value becomes larger, the horizontal BPF is applied to the narrower range.	
	YCSCR3.K15[3:0]	As the value becomes larger, the horizontal BPF is applied to the narrower range.	
	YCSCR4.K12[5:0]	As the value becomes larger, the horizontal BPF is applied to the narrower range.	There is correlation between these bits. When vertical dot crawl is conspicuous, make the bit field value smaller. However, when the value is too small, horizontal dot crawl is produced.
	YCSCR4.K14[5:0]	As the value becomes larger, the horizontal BPF is applied to the narrower range.	
	YCSCR4.K16[3:0]	As the value becomes larger, the horizontal BPF is applied to the narrower range.	

### (4) Correlation Detection Filter Block

The correlation detection filter block, specific to this module, attaches greater importance to the correlation between lines to reduce dot crawl, especially at the intersection of a cross. By mixing the signals after correlation detection filter block, dot crawl can be reduced when dot crawl is not fully removed by the horizontal and vertical filter block.

### (5) Horizontal and Vertical Signal Mixing Block

The horizontal and vertical signal mixing block mixes the signals output from the horizontal, vertical, and horizontal/vertical filter blocks with the signals output from the horizontal and horizontal/vertical filters of the former stage. After that, the appropriate signal is selected from among the signals output from the horizontal, vertical, and horizontal/vertical filters according to the correlation value obtained using (3). Figure 30.34 shows the configuration of the horizontal and vertical signal mixing block.



**Figure 30.34 Configuration of Horizontal and Vertical Signal Mixing Block**

The signal to be mixed, which is output from either the horizontal or horizontal/vertical filter of the former stage, can be selected with YCSCR8.HFIL\_TAP\_SEL. When YCSCR8.HFIL\_TAP\_SEL is 0, the signal output from the 17-TAP filter is selected; and when 1, the signal output from the 9-TAP filter is selected.

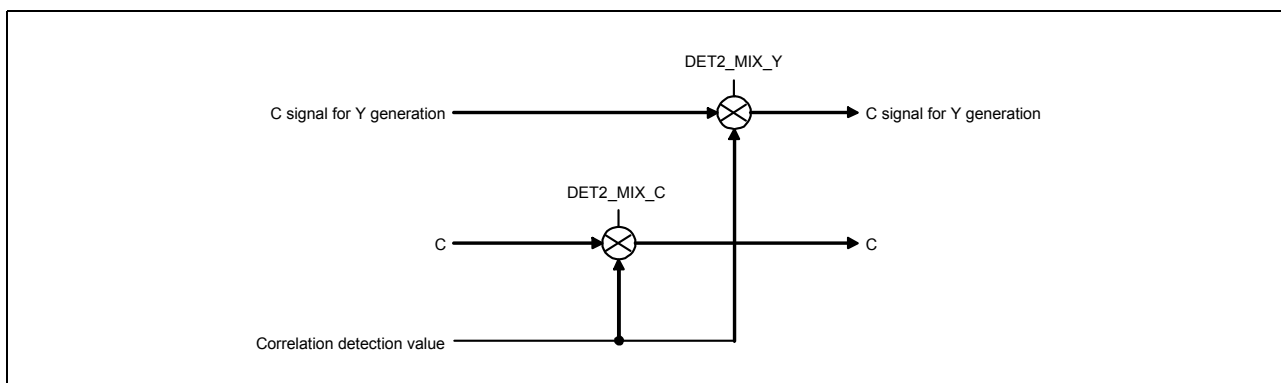
This block mixes the signals output from the horizontal filter with the signals output from the above described horizontal filter of the former stage. The mixing ratio can be set with YCSCR9.HSEL\_MIX\_Y[3:0]. Similarly, this block mixes the signals output from the vertical or horizontal/vertical filter with the signals output from the above described horizontal/vertical filter of the former stage. The mixing ratio can be set with YCSCR9.VSEL\_MIX\_Y[3:0] and YCSCR9.HVSEL\_MIX\_Y[3:0].

This block selects the appropriate signal from among the signals output from the horizontal, vertical, and horizontal/vertical filters according to the correlation value obtained using (3).

**(6) Correlation Detection Value Mixing Block**

The correlation detection value mixing block mixes the C signal for Y generation and the C signal generated using (5) with the signals after correlation detection filter block (4).

Figure 30.35 shows the configuration of the correlation detection value mixing block.



**Figure 30.35 Configuration of Correlation Detection Value Mixing Block**

When YCSCR9.DET2\_ON is 1, this block mixes the signal after the correlation detection filter block. The mixing ratio of the C signal for Y generation to the signal after the correlation detection filter block can be set with YCSCR12.DET2\_MIX\_Y[3:0]. Similarly, the mixing ratio of the C signal to the signal after the correlation detection filter block can be set with YCSCR12.DET2\_MIX\_C[3:0].

When YCSCR9.DET2\_ON is 0, this block outputs the signal after the horizontal and vertical filter block without mixing.

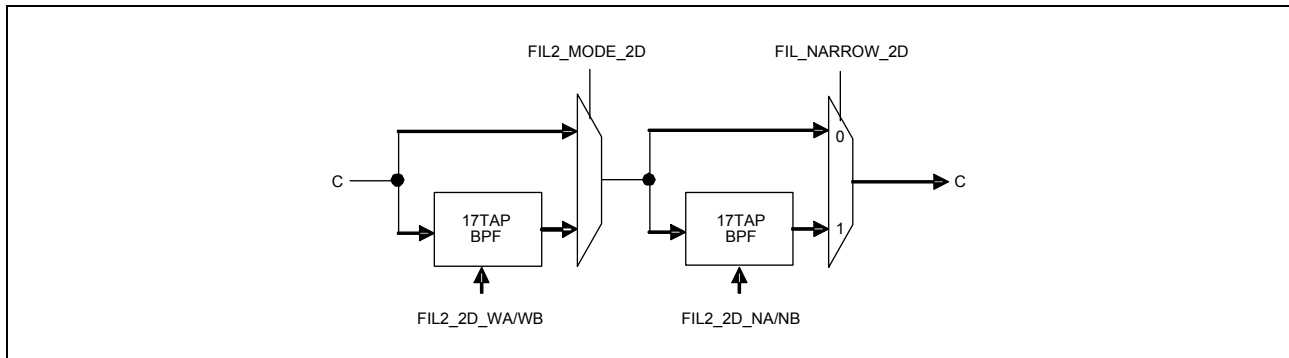
### (7) Y Generation Block

The Y generation block generates the Y signal by subtracting the C signal for Y generation from the video signal.

### (8) Cascade Filter Block

The cascade filter block allows the C signal to pass through the cascade filter or TAKE-OFF filter to further narrow the bandwidth.

Figure 30.36 shows the configuration of the cascade filter block.



**Figure 30.36 Configuration of Cascade Filter Block**

The cascade filter block is composed of two stages. Either bypass operation or 17-TAP filter is selected at the former stage. Bypass operation, cascade, or TAKE-OFF filter can be selected at the former stage with YCSCR12.FIL2\_MODE\_2D[1:0].

Similarly, either bypass operation or 17-TAP filter can be selected at the latter stage with YCSCR12.FIL2\_NARROW\_2D.

Both of the former- and latter-stage filters are universal and can be set with YCTWA\_F0 to YCTWA\_F8, YCTWB\_F0 to YCTWB\_F8, YCTNA\_F0 to YCTNA\_F8, and YCTNB\_F0 to YCTNB\_F8.

Table 30.38 to Table 30.40 show the recommended setting for each filter.

**Table 30.38 Recommended Settings for Two-Dimensional Y/C Filters (NTSC)**

Bit Name	NTSC					Bit Name	NTSC				
	Bypass Operation	Cascade Filter		TAKE-OFF Filter			Bypass Operation	Cascade Filter		TAKE-OFF Filter	
		1 Stage	2 Stages	Broad-band	Narrow-band			1 Stage	2 Stages	Broad-band	Narrow-band
FIL2_MODE_2D	0	1	2	FIL2_MODE_2D	0	1	2				
FIL2_NARROW_2D	—	0	1	—	—	FIL2_NARROW_2D	—	0	1	—	—
FIL2_2D_WA_F0	—	24	24	0	0	FIL2_2D_NA_F0	—	—	24	—	—
FIL2_2D_WA_F1	—	44	44	0	-48	FIL2_2D_NA_F1	—	—	44	—	—
FIL2_2D_WA_F2	—	20	20	0	-20	FIL2_2D_NA_F2	—	—	20	—	—
FIL2_2D_WA_F3	—	-52	-52	-28	160	FIL2_2D_NA_F3	—	—	-52	—	—
FIL2_2D_WA_F4	—	-128	-128	96	232	FIL2_2D_NA_F4	—	—	-128	—	—
FIL2_2D_WA_F5	—	-128	-128	228	-116	FIL2_2D_NA_F5	—	—	-128	—	—
FIL2_2D_WA_F6	—	-12	-12	-916	-900	FIL2_2D_NA_F6	—	—	-12	—	—
FIL2_2D_WA_F7	—	132	132	-204	-4	FIL2_2D_NA_F7	—	—	132	—	—
FIL2_2D_WA_F8	—	200	200	1648	1392	FIL2_2D_NA_F8	—	—	200	—	—
FIL2_2D_WB_F0	—	—	—	—	—	FIL2_2D_NB_F0	—	—	—	—	—
FIL2_2D_WB_F1	—	—	—	—	—	FIL2_2D_NB_F1	—	—	—	—	—
FIL2_2D_WB_F2	—	—	—	—	—	FIL2_2D_NB_F2	—	—	—	—	—
FIL2_2D_WB_F3	—	—	—	—	—	FIL2_2D_NB_F3	—	—	—	—	—
FIL2_2D_WB_F4	—	—	—	—	—	FIL2_2D_NB_F4	—	—	—	—	—
FIL2_2D_WB_F5	—	—	—	—	—	FIL2_2D_NB_F5	—	—	—	—	—
FIL2_2D_WB_F6	—	—	—	—	—	FIL2_2D_NB_F6	—	—	—	—	—
FIL2_2D_WB_F7	—	—	—	—	—	FIL2_2D_NB_F7	—	—	—	—	—
FIL2_2D_WB_F8	—	—	—	—	—	FIL2_2D_NB_F8	—	—	—	—	—

**Table 30.39 Recommended Settings for Two-Dimensional Y/C Filters (PAL)**

Bit Name	PAL					Bit Name	PAL				
	Bypass Operation	Cascade Filter		TAKE-OFF Filter			Bypass Operation	Cascade Filter		TAKE-OFF Filter	
		1 Stage	2 Stages	Broad-band	Narrow-band			1 Stage	2 Stages	Broad-band	Narrow-band
FIL2_MODE_2D	0	1	2	FIL2_MODE_2D	0	1	2				
FIL2_NARROW_2D	—	0	1	—	—	FIL2_NARROW_2D	—	0	1	—	—
FIL2_2D_WA_F0	—	-20	-20	0	0	FIL2_2D_NA_F0	—	—	-20	—	—
FIL2_2D_WA_F1	—	24	24	0	0	FIL2_2D_NA_F1	—	—	24	—	—
FIL2_2D_WA_F2	—	64	64	0	-23	FIL2_2D_NA_F2	—	—	64	—	—
FIL2_2D_WA_F3	—	40	40	16	-46	FIL2_2D_NA_F3	—	—	40	—	—
FIL2_2D_WA_F4	—	-76	-76	59	145	FIL2_2D_NA_F4	—	—	-76	—	—
FIL2_2D_WA_F5	—	-164	-164	85	409	FIL2_2D_NA_F5	—	—	-164	—	—
FIL2_2D_WA_F6	—	-84	-84	-498	-918	FIL2_2D_NA_F6	—	—	-84	—	—
FIL2_2D_WA_F7	—	108	108	-101	-363	FIL2_2D_NA_F7	—	—	108	—	—
FIL2_2D_WA_F8	—	216	216	878	1592	FIL2_2D_NA_F8	—	—	216	—	—
FIL2_2D_WB_F0	—	—	—	—	—	FIL2_2D_NB_F0	—	—	—	—	—
FIL2_2D_WB_F1	—	—	—	—	—	FIL2_2D_NB_F1	—	—	—	—	—
FIL2_2D_WB_F2	—	—	—	—	—	FIL2_2D_NB_F2	—	—	—	—	—
FIL2_2D_WB_F3	—	—	—	—	—	FIL2_2D_NB_F3	—	—	—	—	—
FIL2_2D_WB_F4	—	—	—	—	—	FIL2_2D_NB_F4	—	—	—	—	—
FIL2_2D_WB_F5	—	—	—	—	—	FIL2_2D_NB_F5	—	—	—	—	—
FIL2_2D_WB_F6	—	—	—	—	—	FIL2_2D_NB_F6	—	—	—	—	—
FIL2_2D_WB_F7	—	—	—	—	—	FIL2_2D_NB_F7	—	—	—	—	—
FIL2_2D_WB_F8	—	—	—	—	—	FIL2_2D_NB_F8	—	—	—	—	—



**Table 30.40 Recommended Settings for Two-Dimensional Y/C Filters (SECAM)**

Bit Name	SECAM				Bit Name	SECAM			
	Bypass Operation	Cascade Filter		TAKE-OFF Filter		Bypass Operation	Cascade Filter		TAKE-OFF Filter
		1 Stage	2 Stages				1 Stage	2 Stages	
FIL2_MODE_2D	0	1	2		FIL2_MODE_2D	0	1	2	
FIL2_NARROW_2D	—	0	1	—	FIL2_NARROW_2D	—	0	1	
FIL2_2D_WA_F0	—	-20	-20	0	FIL2_2D_NA_F0	—	—	-1008	
FIL2_2D_WA_F1	—	24	24	-12	FIL2_2D_NA_F1	—	—	1976	
FIL2_2D_WA_F2	—	64	64	-18	FIL2_2D_NA_F2	—	—	-2024	
FIL2_2D_WA_F3	—	40	40	38	FIL2_2D_NA_F3	—	—	444	
FIL2_2D_WA_F4	—	-76	-76	100	FIL2_2D_NA_F4	—	—	1868	
FIL2_2D_WA_F5	—	-164	-164	88	FIL2_2D_NA_F5	—	—	-2864	
FIL2_2D_WA_F6	—	-84	-84	-508	FIL2_2D_NA_F6	—	—	1352	
FIL2_2D_WA_F7	—	108	108	-114	FIL2_2D_NA_F7	—	—	1376	
FIL2_2D_WA_F8	—	216	216	852	FIL2_2D_NA_F8	—	—	-2240	
FIL2_2D_WB_F0	—	-12	-12	—	FIL2_2D_NB_F0	—	—	-1080	
FIL2_2D_WB_F1	—	40	40	—	FIL2_2D_NB_F1	—	—	2800	
FIL2_2D_WB_F2	—	60	60	—	FIL2_2D_NB_F2	—	—	-3308	
FIL2_2D_WB_F3	—	12	12	—	FIL2_2D_NB_F3	—	—	1628	
FIL2_2D_WB_F4	—	-104	-104	—	FIL2_2D_NB_F4	—	—	1444	
FIL2_2D_WB_F5	—	-156	-156	—	FIL2_2D_NB_F5	—	—	-3308	
FIL2_2D_WB_F6	—	-64	-64	—	FIL2_2D_NB_F6	—	—	2140	
FIL2_2D_WB_F7	—	120	120	—	FIL2_2D_NB_F7	—	—	376	
FIL2_2D_WB_F8	—	208	208	—	FIL2_2D_NB_F8	—	—	-1384	

### (9) Over-Range Control Block

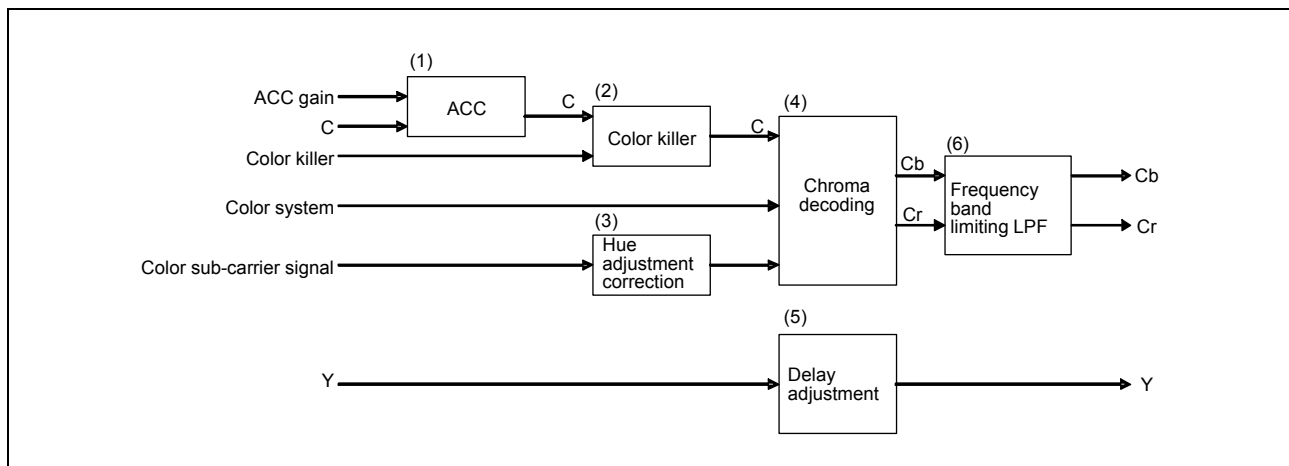
If overflow or underflow occurs at the top or bottom of the color amplitude of video signals, Y/C separation may not be correctly performed thus causing vertical lines to appear as dot crawl. To reduce this phenomenon, the over-range control block automatically inserts the low-pass filter for Y signals (cuts off the frequency components of the vertical lines). Setting RGORCR7.UCMP\_SW to 1 enables over-range control, and setting RGORCR7.DCMP\_SW to 1 enables under-range control. One of four over-range levels can be set with RGORCR1.RADJ\_O\_LEVEL0[9:0], RGORCR3.RADJ\_O\_LEVEL1[9:0], and RGORCR5.RADJ\_O\_LEVEL2[9:0]. Similarly, one of four under-range levels can be set with RGORCR2.RADJ\_U\_LEVEL0[9:0], RGORCR4.RADJ\_U\_LEVEL1[9:0], and RGORCR6.RADJ\_U\_LEVEL2[9:0]. The filter to be inserted is appropriately selected according to the over-range and under-range levels.

Setting HWIDE\_SW to 1 enables detection of the maximum (minimum) level of five pixels in the horizontal direction in addition to the currently processed pixel to detect over-range or under-range occurrence.

### 30.5.6 Chroma Decoding Circuit

The chroma decoding circuit demodulates the C signal extracted by the Y/C separator circuit into the Cb/Cr signal. This circuit has the automatic color control function (ACC), in which the amplitude of the color burst signal is detected to adjust the color gain automatically and also has the TINT/R-Y axis correction function, in which hue is adjusted at demodulation.

Figure 30.37 shows the block diagram of the chroma decoding circuit.



**Figure 30.37** Block Diagram of Chroma Decoding Circuit

#### (1) Automatic Color Control (ACC) Block

The ACC block detects the amplitude of the color burst signal and automatically controls the C signal gain so that the amplitude should be controlled to the target value. The ACC function is activated with `ACCCR1.ACCMODE = 0`. The target amplitude of the color burst signal can be set with `ACCCR1.ACCLEVEL[8:0]`. The gain is fixed when the amplitude falls within `ACCCR1.ACCLEVEL[8:0] ± ACCCR3.ACCPRECIS[5:0]`. The maximum ACC gain can be controlled with `ACCCR1.ACCMAXGAIN[1:0]`. The currently set gain value can be checked by reading `CROMASR1.ACCMAINGAIN[8:0]` (main) and `CROMASR1.ACCSUBGAIN[1:0]` (sub).

The C signal gain can also be set manually by setting `ACCCR1.ACCMODE` to 1. The specific gain value can be set with `ACCCR2.CHROMAMAINGAIN[8:0]` (main) and `ACCCR2.CHROMASUBGAIN[1:0]` (sub).

#### (2) Color Killer

The color killer deletes color information when the color burst signal amplitude is small in a weak electric field. The color killer is turned on or off based on the hysteresis; specifically, it is turned on when the amplitude reaches the value set with `ACCCR3.KILLERLEVEL[5:0]` and turned off when the amplitude reaches the value determined by `ACCCR3.KILLERLEVEL[5:0] + ACCCR1.KILLEROFFSET[3:0]`.

The color killer can also be turned on forcibly by so setting `ACCCR3.KILLERMODE`.

#### (3) Hue Adjustment Correction Block

The hue adjustment correction block adjusts the color sub-carrier signal phase to adjust the Cb/Cr hue after chroma decoding. This function can be used only for the NTSC and PAL systems. The phase of the demodulation axis is controlled with `TINTCR.TINTMAIN[9:0]` and the phase of the R-Y axis is controlled with `TINTCR.TINTSUB[5:0]`.

#### (4) Chroma Decoding Block

The chroma decoding block demodulates the Cb/Cr signal from the C signal. Line averaging can be carried out before demodulation according to `YCDCR.DEMODMODE[1:0]` setting. `YCDCR.DEMODMODE[1:0]` should usually be set to 2 (two-line demodulation for PAL only; one-line demodulation for NTSC).

### (5) Delay Adjustment Block

The delay adjustment block delays the Y signal to adjust the Y/C signal delay. The Y signal can be delayed by -16 to 15 clock pulses with YCDCR.LUMADELAY[4:0].

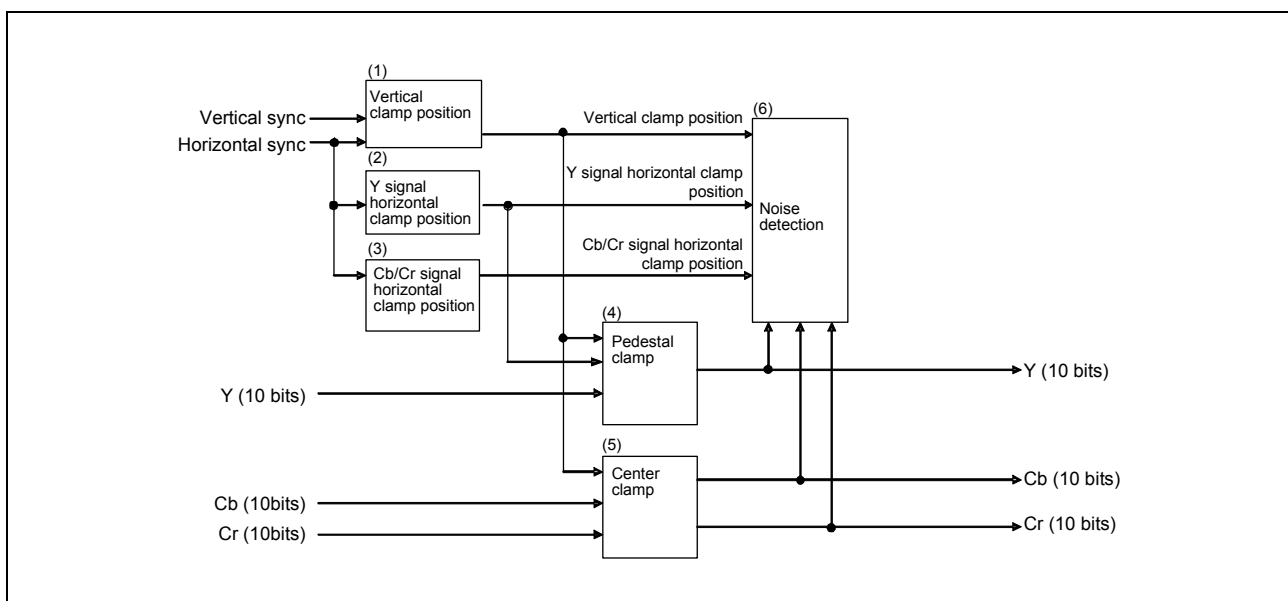
### (6) Frequency Band Limiting LPF

The frequency band limiting LPF limits the frequency band of the Cb/Cr signal after chroma decoding. This LPF is turned on or off according to YCDCR.CHROMALPF setting.

## 30.5.7 Digital Clamp Circuit

The digital clamp circuit provides pedestal clamp for the Y signals and center clamp for the Cb/Cr signals at any position. This circuit also detects the amount of noise using the autocorrelation function.

Figure 30.38 shows the block diagram of the digital clamp circuit.



**Figure 30.38** Block Diagram of Digital Clamp Circuit

### (1) Vertical Clamp Position Control Block

The vertical clamp position can be set with DCPCR4.DCPSTART[5:0] and DCPCR5.DCPEND[5:0]. The setting is used in common to Y, Cb, and Cr signals.

### (2) Y Signal Horizontal Clamp Position Control Block

The horizontal clamp start position of the Y signal can be set with DCPCR7.DCPPOS\_Y[7:0]. The horizontal clamp width can be set with DCPCR6.DCPWIDTH[6:0].

The horizontal clamp width setting is used in common to Y, Cb, and Cr signals.

### (3) Cb/Cr Signal Horizontal Clamp Position Control Block

The horizontal clamp start position of the Cb/Cr signal can be set with DCPCR8.DCPPOS\_C[7:0]. The horizontal clamp width can be set with DCPCR6.DCPWIDTH[6:0].

The horizontal clamp width setting is used in common to Y, Cb, and Cr signals.

#### (4) Pedestal Clamp Control Block

The pedestal clamp control block stabilizes the Y signal pedestal level.

When DCPCR1.DCPMODE\_Y is 0, the value set with DCPCR1.BLANKLEVEL\_Y[9:0] is subtracted from the Y signal, which is expressed as:

$$Y \text{ signal output} = Y \text{ signal input} - \text{DCPCR1.BLANKLEVEL\_Y}[9:0]$$

When DCPCR1.DCPMODE\_Y is 1, the Y signal level detected at the set clamp position and DCPCR1.BLANKLEVEL\_Y[9:0] are added together and the resulting value is subtracted from the Y signal, which is expressed as:

$$Y \text{ signal output} = Y \text{ signal input} - (\text{detected value} + \text{DCPCR1.BLANKLEVEL\_Y}[9:0])$$

The detected value can be read from DCPSR1.CLAMPLEVEL\_Y[9:0].

The clamp response speed can be set with DCPCR3.DCPRESPONSE[2:0]. The setting is used in common to Y, Cb, and Cr signals.

#### (5) Center Clamp Control Block

The center clamp control block stabilizes the Cb/Cr signal center level.

When DCPCR2.DCPMODE\_C is 0, the value set with DCPCR2.BLANKLEVEL\_CB/DCPCR2.BLANKLEVEL\_CR[5:0] is subtracted from the Cb/Cr signal, which is expressed as:

$$Cb \text{ signal output} = Cb \text{ signal input} - \text{DCPCR2.BLANKLEVEL\_CB}[5:0]$$

$$Cr \text{ signal output} = Cr \text{ signal input} - \text{DCPCR2.BLANKLEVEL\_CR}[5:0]$$

When DCPCR2.DCPMODE\_C is 1, the Cb/Cr signal level detected at the set clamp position and DCPCR2.BLANKLEVEL\_CB/DCPCR2.BLANKLEVEL\_CR[5:0] are added together and the resulting value is subtracted from the Cb/Cr signal, which is expressed as:

$$Cb \text{ signal output} = Cb \text{ signal input} - (\text{detected value} + \text{DCPCR2.BLANKLEVEL\_CB}[5:0])$$

$$Cr \text{ signal output} = Cr \text{ signal input} - (\text{detected value} + \text{DCPCR2.BLANKLEVEL\_CR}[5:0])$$

The detected value can be read from DCPSR1.CLAMPLEVEL\_CB[5:0] and DCPSR2.CLAMPLEVEL\_CR[5:0].

The clamp response speed can be set with DCPCR3.DCPRESPONSE[2:0]. The setting is used in common to Y, Cb, and Cr signals.

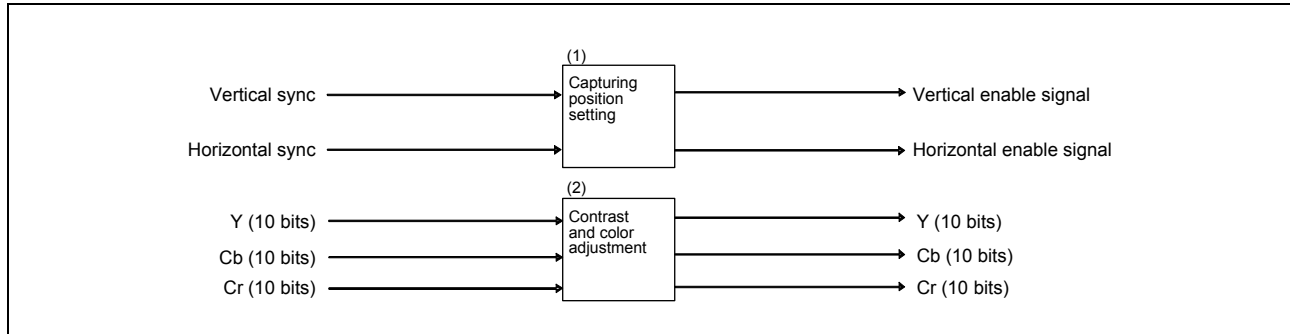
#### (6) Noise Detection Block

Using the autocorrelation function, the noise amount at the set clamp position can be detected. With NSDCR.ACFINPUT[1:0], either Y, Cb, or Cr signal can be selected for which to calculate the autocorrelation function. The delay time for autocorrelation function calculation can be set with NSDCR.ACFLAGTIME[4:0] and accumulated field amount of autocorrelation function can be set with NSDCR.ACFFILTER[1:0]. The autocorrelation function (correlation coefficient) can be read from NSDSR.ACFSTRENGTH[15:0].

### 30.5.8 Output Control Circuit

The output control circuit sets the signal capturing position and adjusts the contrast and color.

Figure 30.39 shows the block diagram of the output control circuit



**Figure 30.39** Block Diagram of Output Control Circuit

#### (1) Capturing Position Setting Block

The capturing position setting block sets the position to capture the input video signals. The position can be set with TGCR1.SRCLEFT[8:0], TGCR2.SRCTOP[5:0], TGCR2.SRCHEIGHT[9:0], and TGCR3.SRCWIDTH[10:0]. These settings are applied only to this module. To set the display size of the input video signals, the vertical capture size register (SCL0\_DS2) and horizontal capture size register (SCL0\_DS3) of the scaler of video display controller 5 should be used.

#### (2) Contrast and Color Adjustment Block

The contrast and color adjustment block adjusts the gain of the output Y/Cb/Cr signals. The contrast (Y signal gain) can be adjusted with YGAINCR.Y\_GAIN2[9:0] and the color (Cb/Cr signal gain) can be adjusted with CBGAINCR.CB\_GAIN2[9:0] and CRGAINCR.CR\_GAIN2[9:0].

## 30.6 Recommended Setting

Table 30.41 and Table 30.42 show the recommended setting for this module.

**Table 30.41 Recommended Setting Common to Various Color Formats**

Register	Bit	Initial Value (Decimal)	Recommended Value (Decimal)	Remarks
ADCCR1	AGCMODE	0	1	AGC on
SYNSCR1	LPFVSYNC	3	3	
	LPFHSYNC	3	5	
	VELOCITYSHIFT_H	0	2	
	SLICERMODE_H	2	2	Automatic slicing
	SLICERMODE_V	2	2	Automatic slicing
SYNSCR2	SYNCMAXDUTY_H	15	15	
	SYNCMINDUTY_H	10	10	
SYNSCR3	SSCLIPSEL	15	15	
	CSYNCSLICE_H	146	146	
SYNSCR4	SYNCMAXDUTY_V	15	15	
	SYNCMINDUTY_V	10	9	
SYNSCR5	VSYNCDelay	0	0	
	VSYNCSLICE	11	10	
	CSYNCSLICE_V	146	146	
HAFCCR1	HAFCGAIN	6	12	
	HAFCFREERUN	0	0	
HAFCCR2	HAFSTART	0	0	
	NOX2HOSC	0	1	
	DOX2HOSC	0	0	
HAFCCR3	HAFEND	8	8	
	HAFMODE	2	2	Comparison disabled during VBI period
VCDWCR1	VCDFREERUN	0	0	
DCPCR1	DCPMODE_Y	1	1	Automatic clamp setting
	DCPCHECK	0	0	
	BLANKLEVEL_Y	0	-40 (984)	
DCPCR2	DCPMODE_C	0	0	
	BLANKLEVEL_CB	0	0	
	BLANKLEVEL_CR	0	0	
DCPCR3	DCPRESPONSE	5	0	
DCPCR4	DCPSTART	16	16	
DCPCR5	DCPEND	16	16	
DCPCR6	DCPWIDTH	54	27	
DCPCR7	DCPPOS_Y	162	162	
DCPCR8	DCPPOS_C	27	54	
NSDCR	ACFINPUT	0	0	
	ACFLAGTIME	0	0	
	ACFFILTER	0	3	

**Table 30.41 Recommended Setting Common to Various Color Formats**

Register	Bit	Initial Value (Decimal)	Recommended Value (Decimal)	Remarks
BTLCR	LOCKRANGE	1	1	
	LOOPGAIN	1	3	
	LOCKLIMIT	2	1	
	BCOFREERUN	0	0	
BTGPCR	BGPCHECK	0	0	
	BGPWIDTH	36	54	
	BGPSTART	130	110	
ACCCR1	KILLEROFFSET	8	5	
	ACCMODE	0	0	ACC on
	ACCMAXGAIN	0	0	
ACCCR2	CHROMASUBGAIN	0	0	
	CHROMAMAINGAIN	256	210	
ACCCR3	ACCRESPONSE	1	1	
	ACCPRECIS	20	8	
	KILLERMODE	0	0	
	KILLERLEVEL	9	4	
TINTCR	TINTSUB	0	0	
	TINTMAIN	0	0	
YCDCR	LUMADELAY	0	0	
	CHROMALPF	0	0	
	DEMODMODE	2	2	
AGCCR1	DOREDUCE	0	0	
	NOREDUCED	0	0	
	AGCRESPONSE	5	4	
AGCCR2	AGCPRECIS	10	10	
PKLIMITCR	PEAKLEVEL	0	2	Peak limiter on
	PEAKATTACK	2	2	
	PEAKRELEASE	0	3	
	PEAKRATIO	0	0	
	MAXPEAKSAMPLES	0	20	
RGORCR1	RADJ_O_LEVEL0	1023	992	
RGORCR2	RADJ_U_LEVEL0	0	64	
RGORCR3	RADJ_O_LEVEL1	1023	960	
RGORCR4	RADJ_U_LEVEL1	0	48	
RGORCR5	RADJ_O_LEVEL2	1023	928	
RGORCR6	RADJ_U_LEVEL2	0	32	
RGORCR7	TEST_MONI	0	0	
	RADJ_MIX_K_FIX	0	0	
	UCMP_SW	0	1	Over-range detection enabled
	DCMP_SW	0	1	Under-range detection enabled
	HWIDE_SW	1	1	
AFCPFCR	PHDET_FIX	0	0	
	PHDET_DIV	5	5	
RUPDCR	NEWSETTING	0	1	

**Table 30.41 Recommended Setting Common to Various Color Formats**

Register	Bit	Initial Value (Decimal)	Recommended Value (Decimal)	Remarks
YCSCR8	HBBPF_NARROW	1	0	
	HVBPF_NARROW	1	0	
	HBBPF1_9TAP_ON	0	0	
	HVBPF1_9TAP_ON	0	0	
	HFIL_TAP_SEL	0	0	
YCSCR11	V_Y_LEVEL	3	0	
DCPCR9	CLP_HOLD_ON_Y	1	0	
	CLP_HOLD_ON_CB	1	0	
	CLP_HOLD_ON_CR	1	0	
YCTWA_F0 to YCTWA_F8	FIL2_2D_WA_F0 to FIL2_2D_WA_F8	Refer to 30.5.5 (8), Cascade Filter Block.		
YCTWB_F0 to YCTWB_F8	FIL2_2D_WB_F0 to FIL2_2D_WB_F8	Refer to 30.5.5 (8), Cascade Filter Block.		
YCTNA_F0 to YCTNA_F8	FIL2_2D_NA_F0 to FIL2_2D_NA_F8	Refer to 30.5.5 (8), Cascade Filter Block.		
YCTNB_F0 to YCTNB_F8	FIL2_2D_NB_F0 to FIL2_2D_NB_F8	Refer to 30.5.5 (8), Cascade Filter Block.		
YGAINCR	Y_GAIN2	512	816	
CBGAINCR	CB_GAIN2	512	663	
CRGAINCR	CR_GAIN2	512	663	
PGA_UPDATE	PGA_VEN	1	1	
PGACR	PGA_GAIN_SEL	0	0	
	PGA_GAIN	0	0	
ADCCR2	ADC_VINSEL	0	0	



**Table 30.42 Recommended Setting for Each Color Format**

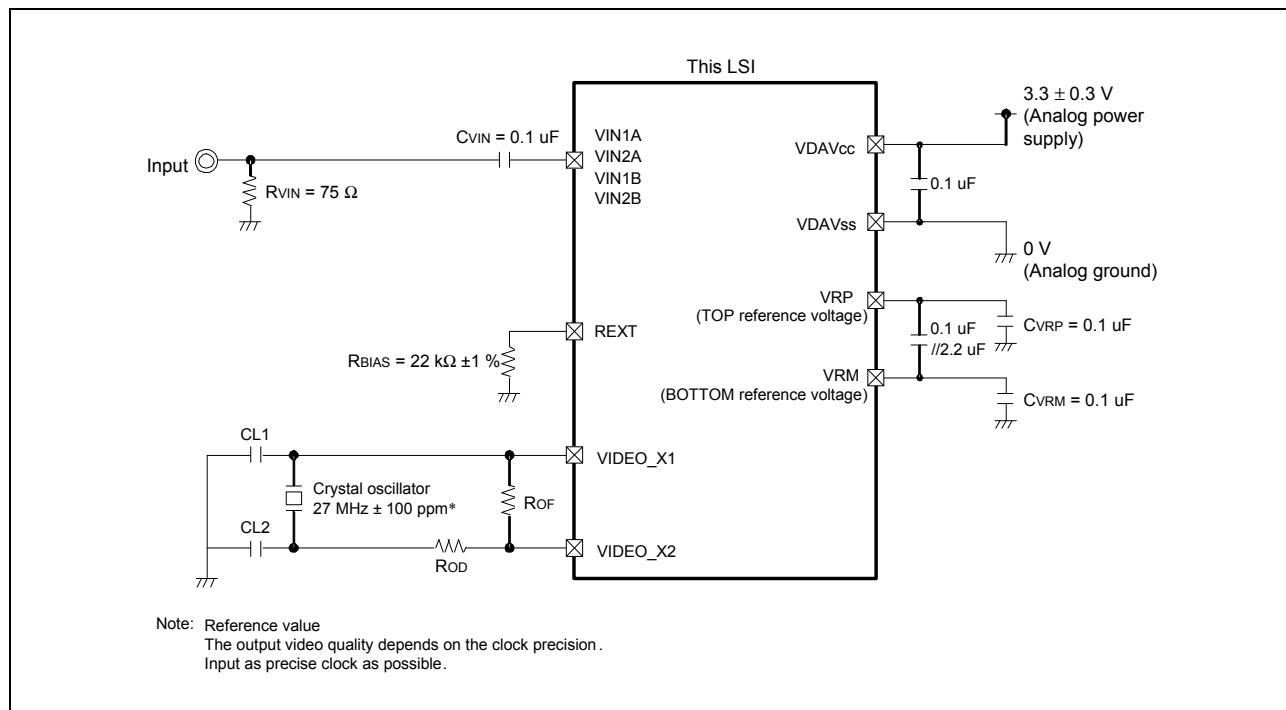
Register	Bit	NTSC- 3.58	NTSC- 4.43	PAL-4.43	PAL-M	PAL-N	SECAM	NTSC- 443 (60 Hz)	PAL-60
Capturing position setting									
TGCR1	SRCLEFT	256	256	256	256	256	256	256	256
TGCR2	SRCTOP	16	19	19	16	19	19	16	16
	SRCHEIGHT	241	288	288	241	288	288	241	241
TGCR3	SRCWIDTH	1428	1412	1412	1428	1412	1412	1428	1428
Horizontal AFC setting									
HAFCCR1	HAFCTYP	692	704	704	692	704	704	692	692
HAFCCR2	HAFCMAX	792	785	785	792	785	785	792	792
HAFCCR3	HAFCMIN	592	630	630	592	630	630	592	592
Vertical countdown setting									
VCDWCR1	NOVCD50	1	0	0	1	0	0	1	1
	NOVCD60	0	1	1	0	1	1	0	0
	VCDDEFAULT	2	1	1	2	1	1	2	2
	VCDWINDOW	30	30	30	30	30	30	30	30
	VCDOFFSET	15	15	15	15	15	15	15	15
BCO setting									
BTLCR	DEFAULTSYS	0	0	1	1	1	2	0	1
	NONTSC358	0	1	1	1	1	1	1	1
	NONTSC443	1	0	1	1	1	1	0	1
	NOPALM	1	1	1	0	1	1	1	1
	NOPALN	1	1	1	1	0	1	1	1
	NOPAL443	1	1	0	1	1	1	1	0
	NOSECAM	1	1	1	1	1	0	1	1
ACC level setting									
ACCCR1	ACCLEVEL	220	220	220	230	230	220	220	230
AGC level setting									
AGCCR1	AGCLEVEL	230	230	242	242	242	242	230	242
Y/C separation setting									
YCSCR3	K15	2	2	2	2	2	2	2	2
	K13	8	8	8	8	8	8	8	8
	K11	4	4	3	3	3	4	4	3
YCSCR4	K16	3	3	4	4	4	3	3	4
	K14	16	16	63	63	63	16	16	63
	K12	8	8	2	2	2	1	8	2
YCSCR5	K22A	32	32	32	32	32	32	32	32
	K21A	6	6	10	10	10	10	6	10
YCSCR6	K22B	8	8	15	15	15	15	8	15
	K21B	6	6	10	10	10	6	6	10
YCSCR7	K23B	6	6	3	3	3	3	6	3
	K23A	3	3	3	3	3	3	3	3
	K24	5	5	8	8	8	8	5	8

**Table 30.42 Recommended Setting for Each Color Format**

Register	Bit	NTSC- 3.58	NTSC- 4.43	PAL-4.43	PAL-M	PAL-N	SECAM	NTSC- 443 (60 Hz)	PAL-60
YCSCR9	DET2_ON	1	1	0	0	0	1	1	0
	HSEL_MIX_Y	6	6	0	0	0	6	6	0
	VSEL_MIX_Y	6	6	0	0	0	6	6	0
	HVSEL_MIX_Y	0	0	0	0	0	0	0	0
YCSCR12	DET2_MIX_C	0	0	0	0	0	0	0	0
	DET2_MIX_Y	2	2	0	0	0	0	2	0
	FIL2_MODE_2D	1	1	0	0	0	1	1	0
	FIL2_NARROW_2D	1	1	1	1	1	1	1	1

### 30.7 Connection Example

Figure 30.40 shows a pin connection example of this module.



**Figure 30.40 Pin Connection Example**

## 31. Video Display Controller 5 (1): Overview

### 31.1 Features

The video display controller 5 consists of the following seven blocks. For the image synthesis, two planes of video image + two graphics planes, one plane of video image + three graphics planes, or four graphics planes can be selected.

1. Input controller: Input video image selection, sync signal adjustment, horizontal noise reduction, and brightness adjustment, gain adjustment, and YCbCr ↔ GBR conversion using a color matrix
2. Scaler: Scale up, scale down, and rotation of input video images using the frame buffer, and repeated recording of the specified number of fields in the frame buffer
3. Image quality improver: Black stretch, LTI/sharpness, and YCbCr ↔ GBR conversion using a color matrix
4. Image synthesizer: Synthesis of two planes of video image + two graphics planes, one plane of video image + three graphics planes, or four graphics planes
5. Output image generator: Writing and reading of image data to and from the frame buffer after the image synthesis
6. Output controller: Brightness/contrast adjustment, gamma correction, dither processing, output format conversion, control signal output for TFT-LCD panel
7. System controller: Interrupt control, panel clock control, CLUT table select signal status flag output

The functions of video display controller 5 are listed in Table 31.1.

**Table 31.1 Features of Video Display Controller 5**

Item	Function
Operating frequency	Video input clock: 27/54 MHz (for video image), 87 MHz or less (for RGB/YCbCr video image) Panel clock: 87 MHz or less (depends on the panel specifications)
Input video image specification	<ul style="list-style-type: none"> <li>• 8-bit input conforming to ITU-R BT.656 standard (27 MHz, interlace signal)</li> <li>• 8-bit input conforming to ITU-R BT.656 extended standard (27 MHz, progressive signal) *1</li> <li>• 8-bit input conforming to ITU-R BT.601 extended standard (27 MHz, interlace signal) *1</li> <li>• 8-bit input conforming to ITU-R BT.601 extended standard (54 MHz, progressive signal) *1</li> <li>• 16-bit input conforming to ITU-R BT.601 extended standard (13.5 MHz, interlace signal) *1</li> <li>• Digital pin input: YCbCr422, YCbCr444, RGB888, RGB666, and RGB565 video image</li> <li>• Digital pin input size: Maximum input video image size to be set *2: 1440 pixels × 1024 lines (horizontal × vertical)</li> </ul> <p>Notes:1. The ITU-R BT.656 and 601 standards do not include the description regarding the progressive signal. The ITU-R BT.601 standard does not include the description regarding the connection interface.</p> <p>2. Depends on the AC characteristics of the connected device.</p> <ul style="list-style-type: none"> <li>• Examples of input video image size: WXGA (1280 × 768), XGA (1024 × 768), SVGA (800 × 600), WVGA (800 × 480), VGA (640 × 480), WQVGA (480 × 240), QVGA in landscape (320 × 240), QVGA in portrait (240 × 320)</li> </ul>
Video image recording function	<ul style="list-style-type: none"> <li>• Storing the video image in the YCbCr422/YCbCr444/RGB565/RGB888 format at a rate of 1/1, 1/2, 1/4, or 1/8 field.</li> <li>• Maximum video image size to be stored: ×1 size of input video image</li> </ul>
Video image quality adjustment function	Contrast adjustment, brightness adjustment, horizontal noise reduction, black stretch, LTI/sharpness
Video image scaling processing	Vertical: ×1/8 to ×8, linear/hold interpolation Horizontal: ×1/8 to ×8, linear/hold interpolation IP conversion can be performed by adjusting the initial phase.
Video image rotation function	<ul style="list-style-type: none"> <li>• 0/90/180/270 degree rotation in the YCbCr422/RGB565 format</li> <li>• Horizontal mirroring in the YCbCr422/YCbCr444/RGB565/RGB888 format</li> </ul>

**Table 31.1 Features of Video Display Controller 5**

Item	Function
Graphics	<ul style="list-style-type: none"> <li>• Number of graphic planes: Four planes (graphics 0, graphics 1, graphics 2, and graphics 3)</li> <li>• Supported pixel formats: <ul style="list-style-type: none"> <li>- RGB565 progressive format (<math>\alpha</math>: none, R: 5 bits, G: 6 bits, B: 5 bits; 16 bits in total)</li> <li>- RGB888 progressive format (<math>\alpha</math>: none, R: 8 bits, G: 8 bits, B: 8 bits; 24 bits in total)</li> <li>- <math>\alpha</math> RGB1555 progressive format (<math>\alpha</math>: 1 bit, R: 5 bits, G: 5 bits, B: 5 bits; 16 bits in total)</li> <li>- <math>\alpha</math> RGB4444 progressive format (<math>\alpha</math>: 4 bits, R: 4 bits, G: 4 bits, B: 4 bits; 16 bits in total)</li> <li>- <math>\alpha</math> RGB8888 progressive format (<math>\alpha</math>: 8 bits, R: 8 bits, G: 8 bits, B: 8 bits; 32 bits in total)</li> <li>- RGB<math>\alpha</math>5551 progressive format (R: 5 bits, G: 5 bits, B: 5 bits, <math>\alpha</math>: 1 bit; 16 bits in total)</li> <li>- RGB<math>\alpha</math>8888 progressive format (R: 8 bits, G: 8 bits, B: 8 bits, <math>\alpha</math>: 8 bits; 32 bits in total)</li> <li>- CLUT8 progressive format (CLUT: 8 bits)</li> <li>- CLUT4 progressive format (CLUT: 4 bits)</li> <li>- CLUT1 progressive format (CLUT: 1 bits)</li> <li>- YCbCr422 progressive format (Y: 8 bits, Cb/Cr: 8 bits; 16 bits in total) (only for graphics 0 and 1)</li> <li>- YCbCr444 progressive format (Y: 8 bits, Cb/Cr: 8 bits; 16 bits in total) (only for graphics 0 and 1)</li> </ul> </li> <li>• Maximum image size to be read: 1440 pixels <math>\times</math> 1440 lines (horizontal <math>\times</math> vertical)</li> </ul>
Graphics function	<p>Alpha blending in rectangular area: Mixes images according to transparency rate <math>\alpha</math> in the specified area (fade-in and fade-out functions are available.)</p> <p>Chroma-key: Mixes images using the specified RGB color and CLUT value according to transparency rate <math>\alpha</math>.</p> <p>Alpha blending in one pixel units: Mixes images according to transparency rate <math>\alpha</math> when the target graphics image is in the <math>\alpha</math>RGB1555, <math>\alpha</math>RGB4444, <math>\alpha</math>RGB8888, RGB<math>\alpha</math>5551, RGB<math>\alpha</math>8888, or CLUT8/4/1 format.</p> <hr/> <p>For each dot, the priority among the <math>\alpha</math> values of the above functions is as follows: Alpha blending in rectangular area &gt; Chroma-key &gt; Alpha blending in one pixel units</p>
Output video image size	<p>Maximum output video image size to be set*: 1999 pixels <math>\times</math> 2035 lines (horizontal <math>\times</math> vertical)</p> <p>Note: * Depends on the AC characteristics of the display panel.</p> <p>Examples of output video image size:</p> <ul style="list-style-type: none"> <li>• WXGA (1280 <math>\times</math> 768), XGA (1024 <math>\times</math> 768)</li> <li>• SVGA (800 <math>\times</math> 600), WVGA (800 <math>\times</math> 480),</li> <li>• VGA (640 <math>\times</math> 480), WQVGA (480 <math>\times</math> 240),</li> <li>• QVGA size in landscape (320 <math>\times</math> 240)</li> <li>• QVGA size in portrait (240 <math>\times</math> 320)</li> </ul>
Output video image format	<ul style="list-style-type: none"> <li>• RGB888 progressive video output (24-bit parallel output)</li> <li>• RGB666 progressive video output (18-bit parallel output)</li> <li>• RGB565 progressive video output (16-bit parallel output)</li> <li>• RGB888 progressive video output (8-bit serial output)</li> </ul>
Panel output adjustment	Panel brightness/contrast adjustment, RGB gamma correction, dither processing, output format conversion
Sync signal output	Control signal output for the TFT-LCD panel
Interrupt output	<ul style="list-style-type: none"> <li>• Vsync signal for video image input/output</li> <li>• Line interrupt output (can be output on a desired line.)</li> <li>• Erroneous Vsync cycle detection signal for video input</li> <li>• Field write completion signal</li> <li>• Overflow/underflow signal for the internal buffer</li> </ul>

### 31.2 Block Diagram

Figure 31.1 and Figure 31.2 show the entire block diagram of this module. This LSI incorporates two video display controllers (channels 0 and 1), to each of which the video signal is input from the corresponding video decoder. Each controller can also fetch output from the input control block of the other controller. For details, see the description of each block.

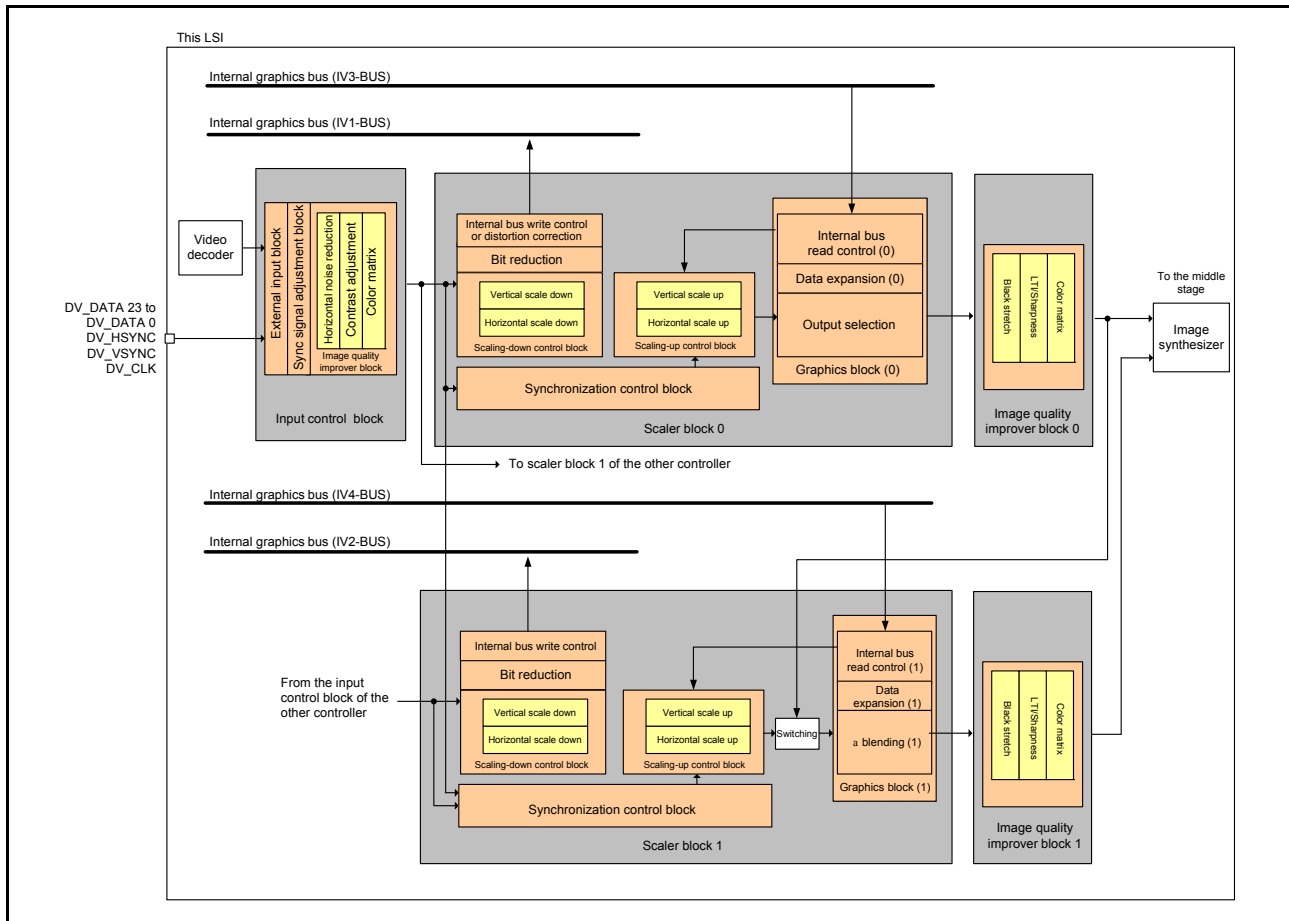


Figure 31.1 Video Display Controller 5 Former Stage Block Diagram

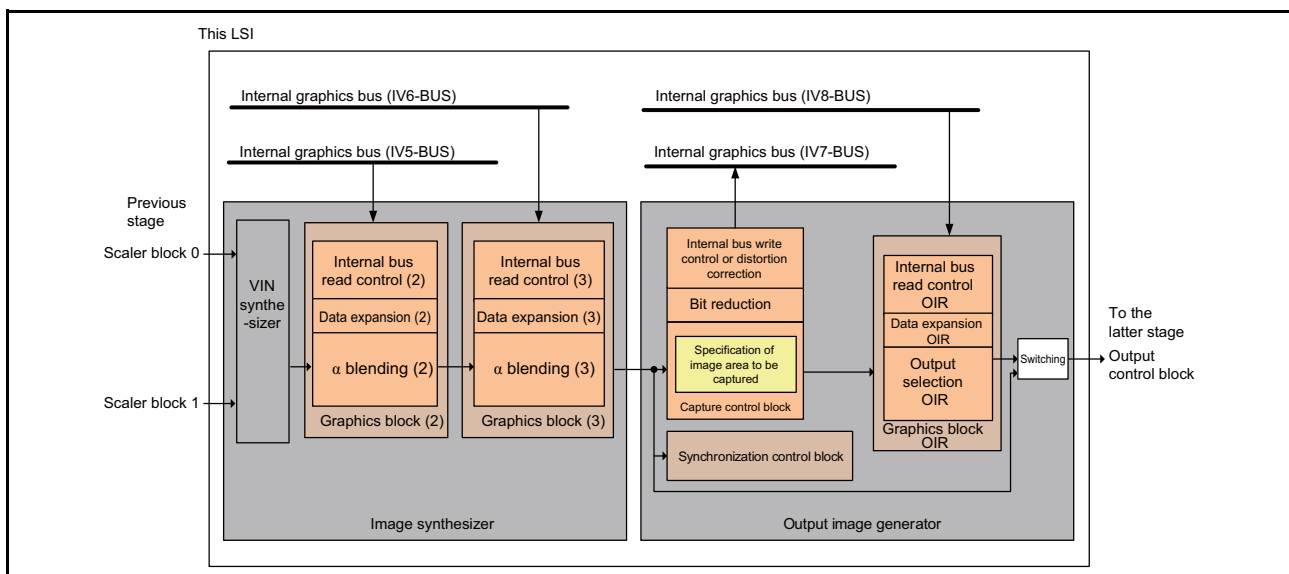


Figure 31.2 Video Display Controller 5 Middle Stage Block Diagram

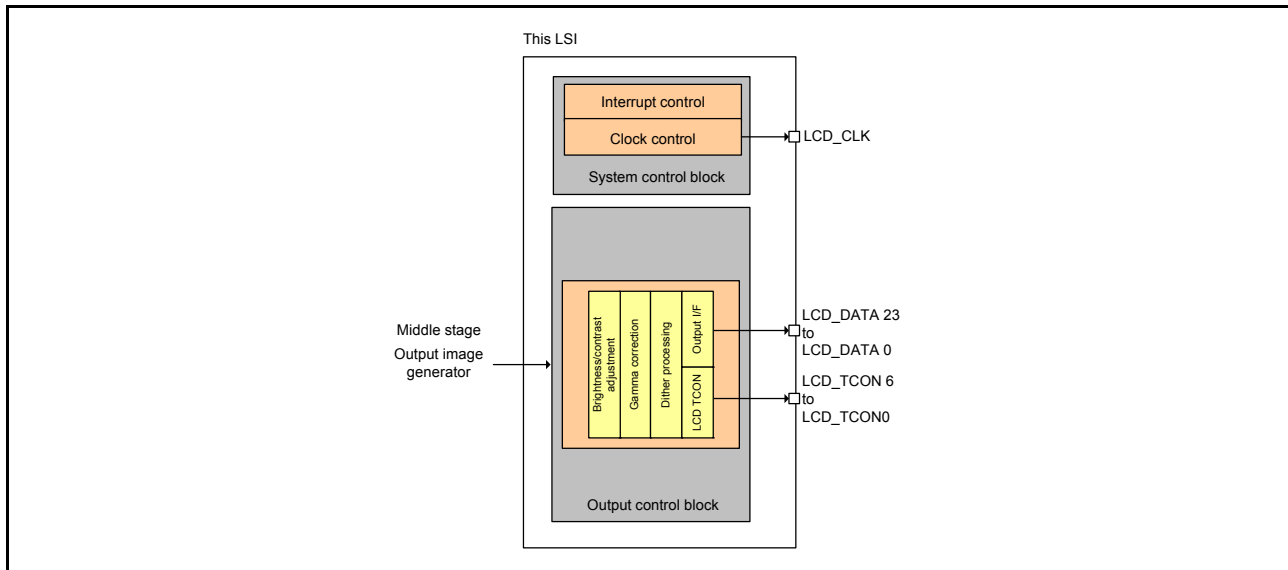


Figure 31.3 Video Display Controller 5 Latter Stage Block Diagram

### 31.3 Input/Output Pins

Table 31.2 and Table 31.3 show the pin configuration.

Table 31.2 Input/Output Pins (Channel 0)

Symbol	I/O	Pin Name	Function
DV0_CLK	Input	External input clock 0	External input 0 clock pin
DV0_VSYNC	Input	External input Vsync 0	External input 0 Vsync signal pin
DV0_HSYNC	Input	External input Hsync 0	External input 0 Hsync signal pin
DV0_DATA 23 to DV0_DATA 0	Input	External input video image data 0	External input 0 video image data pin
LCD0_CLK	Output	Panel clock 0	Panel output 0 clock pin
LCD0_DATA 23 to LCD0_DATA 0	Output	Video image data 0 for panel	Panel output 0 video image data pin
LCD0_TCON 6 to LCD0_TCON 0	Output	Control signal 0 for panel	Panel output 0 timing control pin
LCD0_EXTCLK	Input	Panel clock source 0	Panel clock source 0 input pin

Table 31.3 Input/Output Pins (Channel 1)

Symbol	I/O	Pin Name	Function
DV1_CLK	Input	External input clock 1	External input 1 clock pin
DV1_VSYNC	Input	External input Vsync 1	External input 1 Vsync signal pin
DV1_HSYNC	Input	External input Hsync 1	External input 1 Hsync signal pin
DV1_DATA 23 to DV1_DATA 0	Input	External input video image data 1	External input 1 video image data pin
LCD1_CLK	Output	Panel clock 1	Panel output 1 clock pin
LCD1_DATA 23 to LCD1_DATA 0	Output	Video image data 1 for panel	Panel output 1 video image data pin
LCD1_TCON 6 to LCD1_TCON 0	Output	Control signal 1 for panel	Panel output 1 timing control pin
LCD1_EXTCLK	Input	Panel clock source 1	Panel clock source 1 input pin

## 31.4 Clocks

There are two clocks to be mainly used by the video display controller 5: the video image clock and pixel clock.

The video image clock is used while the video image is processed in the input controller, passed to the scale-down control block in the scaler, and then written to the buffer (internal bus write control). When the INP\_SEL bit is 0 (video decoder output selected) in INP\_SEL\_CNT of the input controller, the VIDEO\_X1 clock (27 MHz) is used as the video image clock. When INP\_SEL is 1 (external input pin selected), the DV\_CLK clock is used as the video image clock.

The pixel clock is used in graphics read-out processing by the scaler (internal bus read controller) through output controller processing. When the parallel RGB output is selected in the output controller, the frequencies of the pixel clock and panel clock (LCD\_CLK) are the same. The panel clock can be selected from the video clock, LCD\_EXTCLK0, LCD\_EXTCLK1, peripheral bus clock 1 (P1 $\phi$ ), and output clock from the LVDS PLL\* with SYSCNT\_PANEL\_CLK.PANEL\_ICKSEL[1:0] and SYSCNT\_PANEL\_CLK.PANEL\_OCKSEL[1:0] of the system controller. When the serial RGB output (3/4 speed mode) is selected in the output controller, the pixel clock frequency is 1/3 or 1/4 the panel clock (LCD\_CLK) frequency.

Note: \* For details on the LVDS PLL, see section 40, LVDS Output Interface.

## 31.5 Hsync and Vsync Signals

Hsync and Vsync signals are generated by the synchronization control block of the scaler and output image generator. Since the Hsync and Vsync signals are used as the reference signals for the LCDTCON, which generates various panel driving timings, they are also the reference signals for the control signals (LCD\_TCON6 to LCD\_TCON0 pins) passed to the panel.

The output Hsync signal always operates at a free-running frequency. On the other hand, the output Vsync signal is selected from the external input and free-running Vsync signals.



### 31.5.1 External Input Vsync

#### (1) Operation Outline

In this mode, the output Vsync signal is generated according to an external input Vsync signal. The output Hsync signal is free running even in this mode. Figure 31.4 shows the timing of external input Vsync signal.

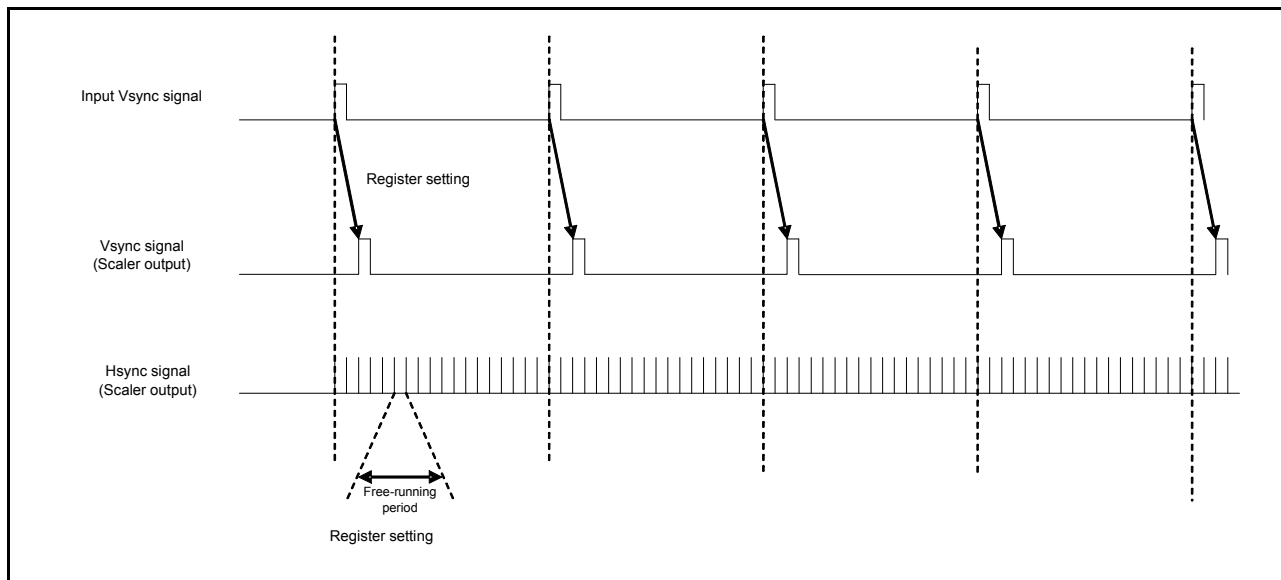


Figure 31.4 External Input Vsync Timing

(2) Notes

When Vsync is externally input, generation of the output Vsync signal is based on the external Vsync signal. That is, the output Vsync signal follows the input Vsync signal, so if an unstable Vsync signal is input, the output Vsync signal will also be unstable.

Since the output Hsync signal is generated according to the frequency generated by a free-running clock and the Vsync signal is generated from the video input as a base, the signals will not be in synchronization. This module adjusts the timing between these signals by adjusting the output Vsync signal so that it stays in time with the output Hsync signal.

Therefore, even if the input Vsync signal is stable, the timing of the output Vsync signal may be increased or decreased by up to one line to stay in synchronization with the timing of the output Hsync signal.

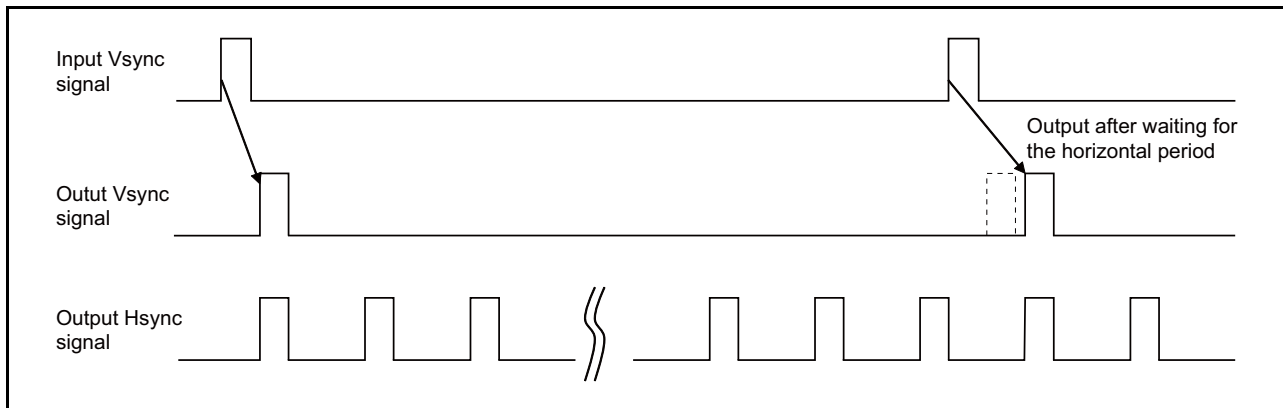


Figure 31.5 Detailed Timing Chart for Generation of the Output Vsync Signal

31.5.2 Free-Running Vsync

(1) Operation Outline

In this mode, the Vsync signal is generated according to the pixel clock (free running). The output Hsync signal is also free running. Figure 31.6 shows the timing.

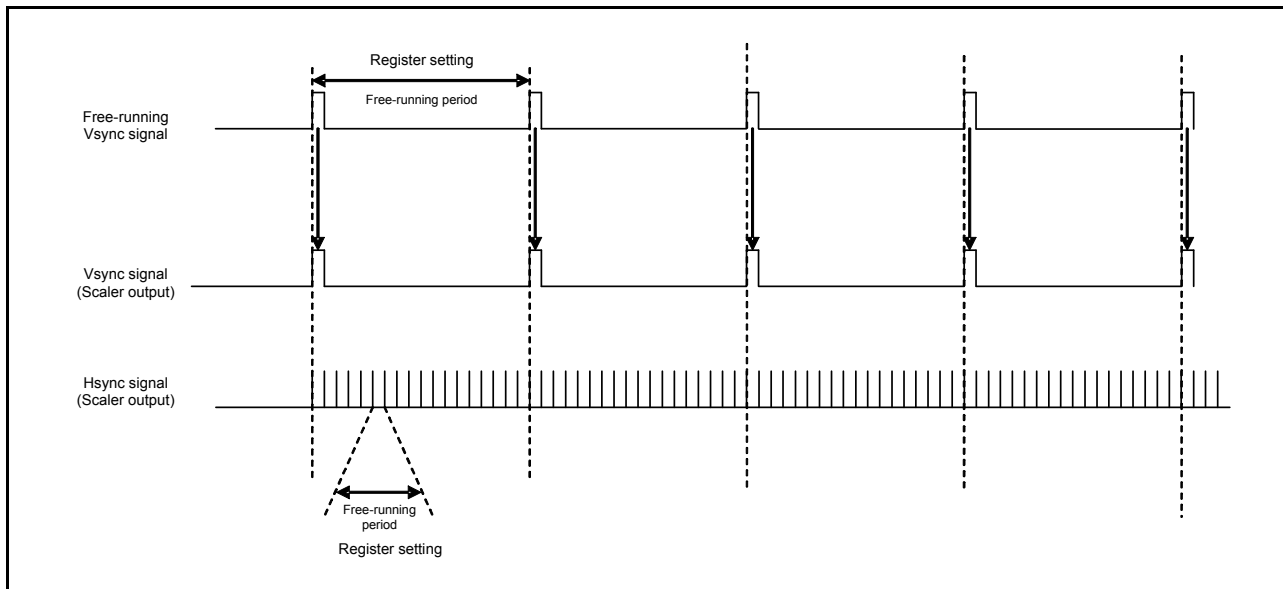


Figure 31.6 Free Running Vsync Timing

(2) Pointer Buffers

In free-running vertical synchronization mode, output of the input video image to a panel may lead to flicker in the output video image. This occurs when the input and output vertical sync signals are not in synchronization. To prevent this, use the pointer buffers to adjust the timing between the input and output video images in frame units. If the input Vsync signal is faster than the output Vsync signal, frames from the input video image are skipped in the output image for display. On the other hand, when the input Vsync signal is slower than the output Vsync signal, frames from the input video image are repeated. However, when the difference in timing between the input and output Vsync signals is too large, the pointer buffers cannot deal with the difference, so flicker may occur. As more buffers are used, the pointer buffers can deal with larger differences in frequency.

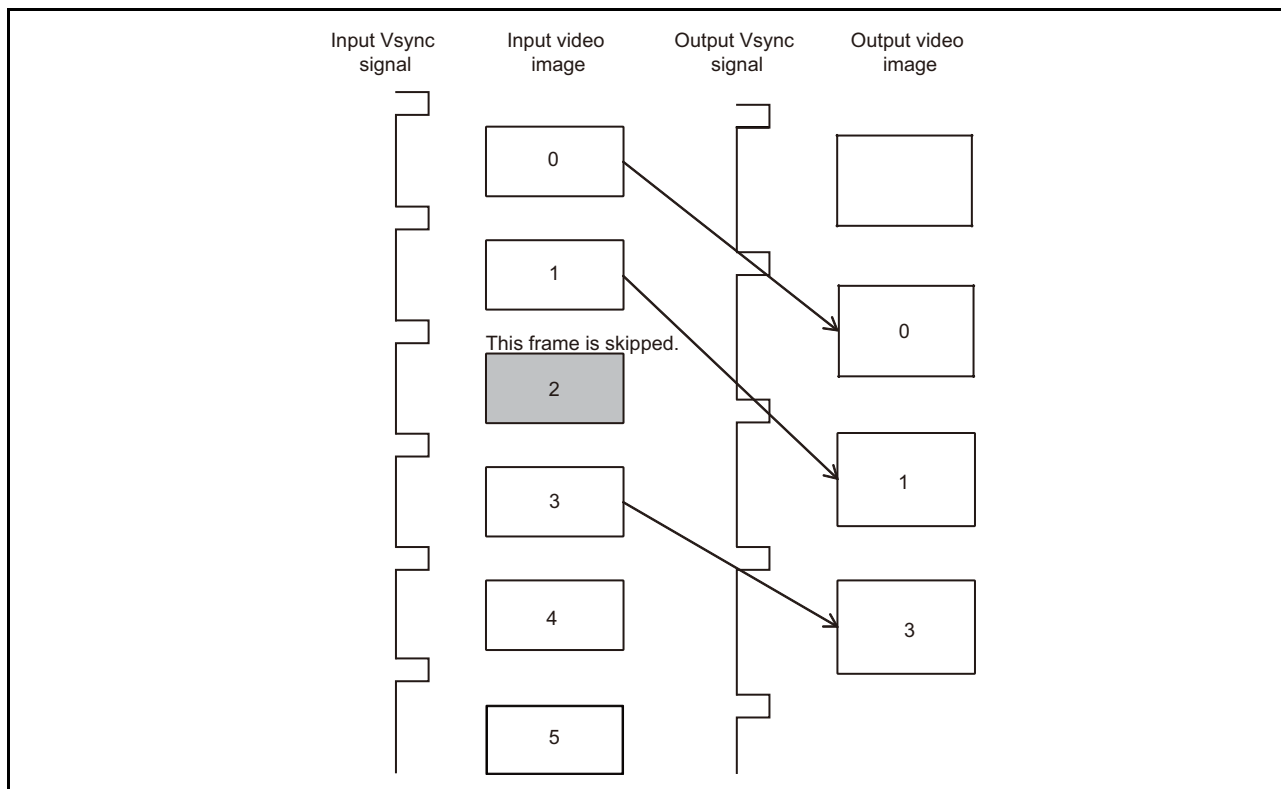
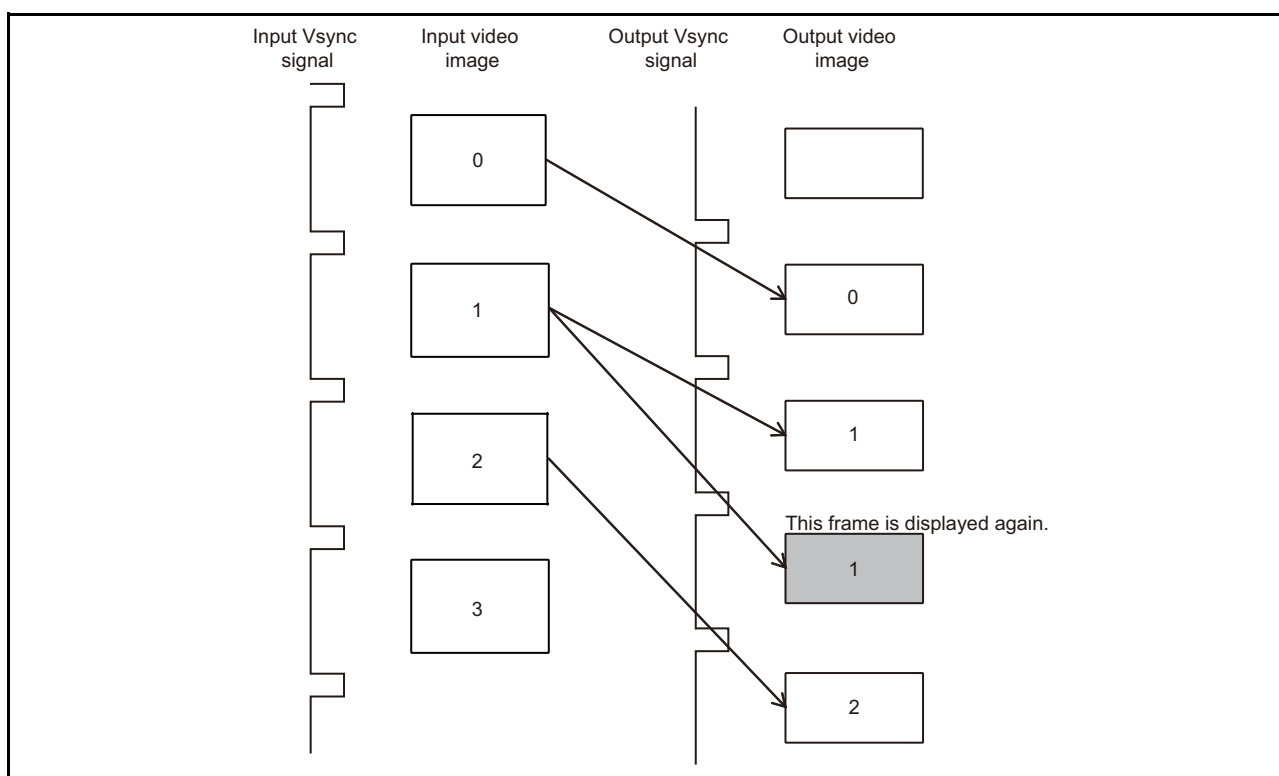


Figure 31.7 Timing when the Input Vsync Signal is Faster than the Output Vsync Signal



**Figure 31.8** Timing when the Input Vsync Signal is Slower than the Output Vsync Signal

### 31.5.3 Blending Two Input Video Images

If images input from scalers 0 and 1 are to be combined, the output synchronization signals from the two scalers must be the same.

When the synchronization signals from scaler 0 are to serve as the standard, use the synchronization signals generated by scaler 0 as the output synchronization signals for scaler 1.

In this case, the timing of the vertical synchronization signal for the video input to scaler 1 will not be synchronized with that of the output vertical synchronization signal, which may lead to flickering of the output video.

Accordingly, use a pointer buffer to apply frame-buffer control in the same way as when scaler 1 is used with a free-running vertical synchronization signal.

Likewise, use a pointer buffer to apply frame-buffer control for scaler 0 when using the synchronization signals generated by scaler 1 as the standard.

**Table 31.4** Vsync Signal When Blending Two Input Video Images

Standard output Sync signal	Scaler 0	Scaler 1
Scaler 0	Select from external or free-running Vsync signal	Sync signal generated by scaler 0 is selected and a pointer buffer handles frame buffer control.
Scaler 1	Sync signal generated by scaler 1 is selected and a pointer buffer handles frame buffer control.	Select from external or free-running Vsync signal

### 31.5.4 Usage Note on Changing Vsync Signal Selections

When the Vsync signal selection is changed, the output Vsync signal is discontinuous, resulting in disordered panel display. In this case, perform the mute processing according to the panel specification as necessary and change the Vsync signal selection.

## 32. Video Display Controller 5 (2): Input Controller

### 32.1 Input Controller Functions

#### 32.1.1 Overview of Functions

The input controller selects either video decoder output signals or signals supplied via the external input pins, and subjects the signals to synchronization adjustment, horizontal noise reduction, contrast correction (dynamic range compression), and brightness adjustment, gain adjustment, and YCbCr ↔ GBR conversion using a color matrix.

For contrast correction, refer to section 39, Dynamic Range Compression.

The functional block diagram of the input controller is shown below.

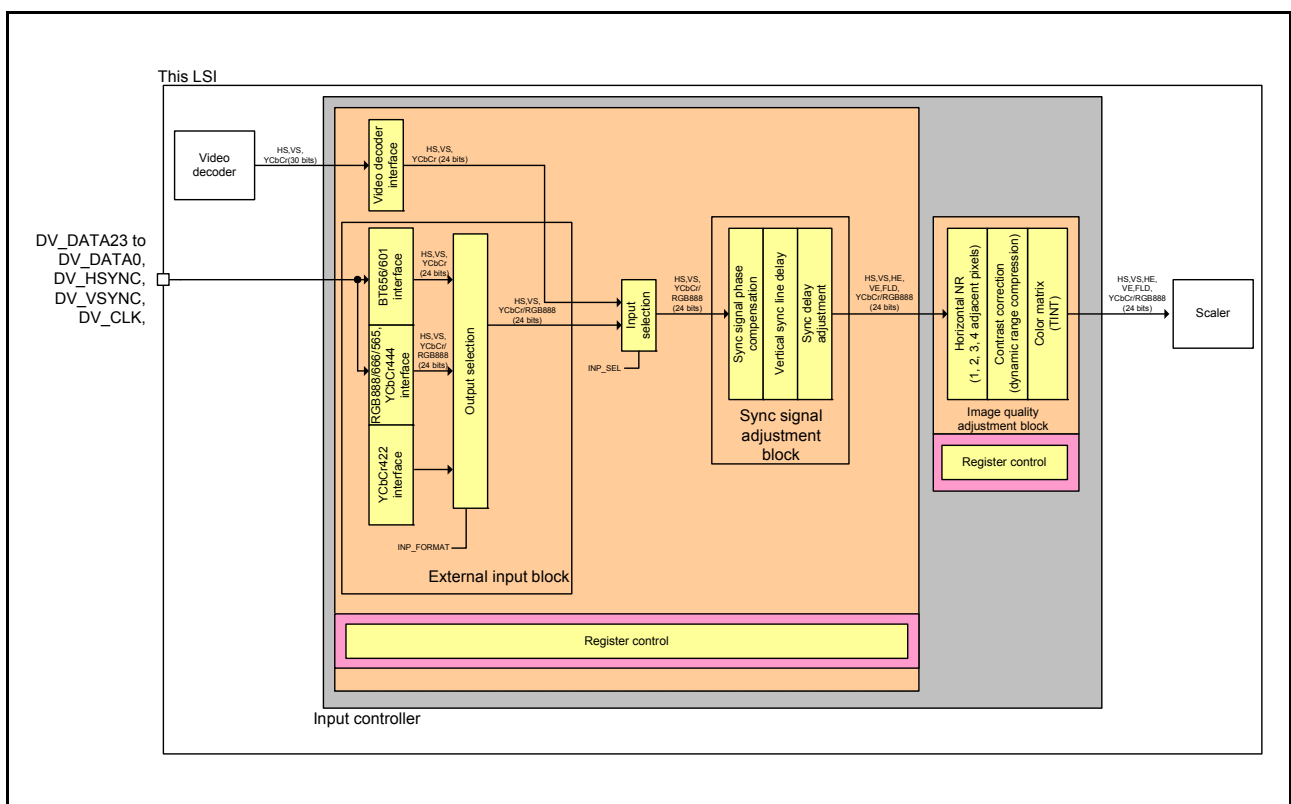


Figure 32.1 Functional Block Diagram of Input Controller

### 32.1.2 Updating Registers of External Signal Input Block and Sync Signal Adjustment Block

The control registers of the external input block and sync signal adjustment block are updated by setting the relevant update control bit to 1.

For the control registers other than the IMGCNT\_DRC\_REG register of the image quality adjustment block, the update timing is controlled using the Vsync signal.

After 1 is set to the bits in the update control register, the contents of the relevant registers are actually modified at the rising edge of the Vsync signal, when the update control register is automatically cleared to 0.

**Table 32.1 Register Update Control**

Register Name	Bit Name	Initial Value	Description
INP_UPDATE	INP_EXT_UPDATE	0	External Input Block Register Update 0: Registers are not updated. 1: Registers are updated.
INP_UPDATE	INP_IMG_UPDATE	0	Sync Signal Adjustment Block Register Update 0: Registers are not updated. 1: Registers are updated.
IMGCNT_UPDATE	IMGCNT_VEN	0	Image Quality Adjustment Block Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync signal.

### 32.1.3 Selecting Input Signals

The input controller selects either video decoder output signals or signals supplied via the external input pins.

**Table 32.2 Input Signal Selection**

Register Name	Bit Name	Initial Value	Description
INP_SEL_CNT	INP_SEL	0	Input Select 0: Video decoder output signals 1: Signals supplied via the external input pins

### 32.1.4 Controlling Externally Input Video Signals

The externally input video image signals in the YCbCr444, RGB888, RGB666, RGB565, BT656 (extended), BT601 (extended), YCbCr422 (the 16-bit data format of the BT601 standard) formats can be handled.

The BT656 signals can be used for the 525-line and 59.94-Hz (27.0-MHz) or the 625-line and 50.00-Hz (27.0-MHz) interlace signals and for the 525-line and 59.94-Hz (54.0-MHz) or the 625-line and 50.00-Hz (54.0-MHz) BT656-extended progressive signals.

The BT601 signals can be used for the 8-bit data line 525-line and 59.94 Hz (27.0-MHz) or the 625-line and 50.00-Hz (27.0-MHz) interlace signals and for the 525-line and 59.94 Hz (54.0-MHz) or the 625-line and 50.00-Hz (54.0-MHz) extended progressive signals.

The YCbCr422 signals can be used for the 16-bit data line 525-line and 59.94-Hz (13.5-MHz) or the 625-line and 50.00-Hz (13.5-MHz) BT601 interlace signals.

The YCbCr422 signals can be used for the 16-bit data line 525-line and 59.94-Hz (13.5-MHz) or the 625-line and 50.00-Hz (13.5-MHz) BT601-extended interlace signals.

The above signals can be selected by the INP\_FORMAT[2:0] bits. Bit endian change and B/R signal swap are controlled by setting the INP\_ENDIAN\_ON and INP\_SWAP\_ON bits.

**Table 32.3 Externally Input Video Signal Control**

Register Name	Bit Name	Initial Value	Description
INP_SEL_CNT	INP_FORMAT[2:0]	000	External Input Format Select 0: YCbCr444, RGB888 1: RGB666 2: RGB565 3: BT656 4: BT601 5: YCbCr422 6, 7: Setting prohibited
INP_EXT_SYNC_CNT	INP_ENDIAN_ON	0	External Input Bit Endian Change On/Off Control 0: Off 1: On
INP_EXT_SYNC_CNT	INP_SWAP_ON	0	External Input B/R Signal Swap On/Off Control 0: Off 1: On

### 32.1.5 Selecting Clock Edge for Externally Input Signals

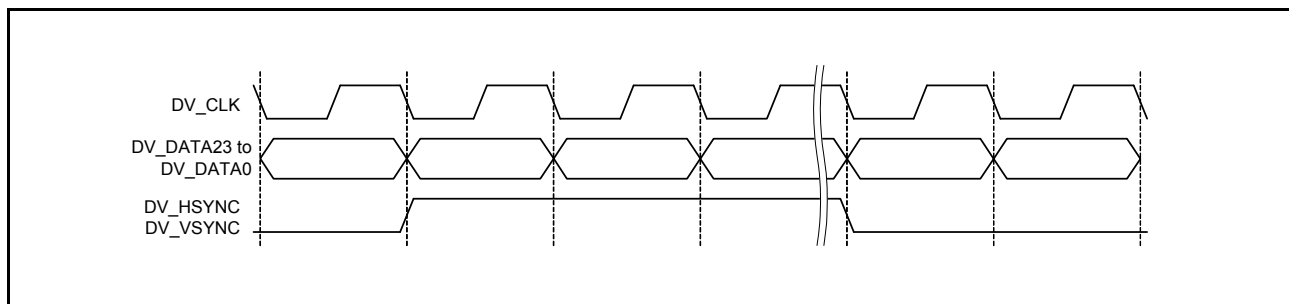
The clock edge for receiving the video image signals, Vsync signals, and Hsync signals is individually selected with the INP\_PXD\_EDGE, INP\_VS\_EDGE, INP\_HS\_EDGE bits.

**Table 32.4 Externally Input Clock Edge Selection**

Register Name	Bit Name	Initial Value	Description
INP_SEL_CNT	INP_PXD_EDGE	0	Clock Edge Select for Capturing External Input Video Image Signals DV_DATA23 to DV_DATA0 0: Rising edge 1: Falling edge
INP_SEL_CNT	INP_VS_EDGE	0	Clock Edge Select for Capturing External Input Vsync Signal DV_VSYNC 0: Rising edge 1: Falling edge
INP_SEL_CNT	INP_HS_EDGE	0	Clock Edge Select for Capturing External Input Hsync Signal DV_HSYNC 0: Rising edge 1: Falling edge

Figure 32.2 shows the typical input timing of externally input signals.

The input signals can be received at the rising edge of the clock signal DV\_CLK when the INP\_PXD\_EDGE, INP\_VS\_EDGE, and INP\_ES\_EDGE bits are 0.



**Figure 32.2 Typical Input Timing of Externally Input Signals (Clock Phase)**

### 32.1.6 Externally Input Sync Signal Inversion Control

Inversion of polarity of the Vsync and Hsync signals can be controlled by the INP\_VS\_INV and INP\_HS\_INV bits.

**Table 32.5 Sync Signal Inversion Control**

Register Name	Bit Name	Initial Value	Description
INP_EXT_SYNC_CNT	INP_VS_INV	0	External Input Vsync Signal DV_VSYNC Inversion Control 0: Not inverted (positive polarity) 1: Inverted (negative polarity)
INP_EXT_SYNC_CNT	INP_HS_INV	0	External Input Hsync Signal DV_HSYNC Inversion Control 0: Not inverted (positive polarity) 1: Inverted (negative polarity)



### 32.1.7 Bit Allocation of Externally Input Video Image Signals

Allocation of the externally input video image signal pins DV\_DATA to the signals in each format is described below.

#### (1) YCbCr444/RGB888 Input

When the external input is of YCbCr444/RGB888 format, the video image signal pins DV\_DATA are allocated to the internal signals Y/GOUT, Cb/BOUT, Cr/ROUT, as shown in Table 32.6.

**Table 32.6 Bit Allocation of DV\_DATA Pin Inputs when the External Input is of YCbCr444/RGB888**

<b>INP_FORMAT[2:0]</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>
<b>INP_ENDIAN_ON</b>	<b>0</b>	<b>0</b>	<b>1</b>	<b>1</b>
<b>INP_SWAP_ON</b>	<b>0</b>	<b>1</b>	<b>0</b>	<b>1</b>
DV_DATA23	Cr/ROUT[7]	Cb/BOUT[7]	Cr/ROUT[0]	Cb/BOUT[0]
DV_DATA22	Cr/ROUT[6]	Cb/BOUT[6]	Cr/ROUT[1]	Cb/BOUT[1]
DV_DATA21	Cr/ROUT[5]	Cb/BOUT[5]	Cr/ROUT[2]	Cb/BOUT[2]
DV_DATA20	Cr/ROUT[4]	Cb/BOUT[4]	Cr/ROUT[3]	Cb/BOUT[3]
DV_DATA19	Cr/ROUT[3]	Cb/BOUT[3]	Cr/ROUT[4]	Cb/BOUT[4]
DV_DATA18	Cr/ROUT[2]	Cb/BOUT[2]	Cr/ROUT[5]	Cb/BOUT[5]
DV_DATA17	Cr/ROUT[1]	Cb/BOUT[1]	Cr/ROUT[6]	Cb/BOUT[6]
DV_DATA16	Cr/ROUT[0]	Cb/BOUT[0]	Cr/ROUT[7]	Cb/BOUT[7]
DV_DATA15	Y/GOUT[7]	Y/GOUT[7]	Y/GOUT[0]	Y/GOUT[0]
DV_DATA14	Y/GOUT[6]	Y/GOUT[6]	Y/GOUT[1]	Y/GOUT[1]
DV_DATA13	Y/GOUT[5]	Y/GOUT[5]	Y/GOUT[2]	Y/GOUT[2]
DV_DATA12	Y/GOUT[4]	Y/GOUT[4]	Y/GOUT[3]	Y/GOUT[3]
DV_DATA11	Y/GOUT[3]	Y/GOUT[3]	Y/GOUT[4]	Y/GOUT[4]
DV_DATA10	Y/GOUT[2]	Y/GOUT[2]	Y/GOUT[5]	Y/GOUT[5]
DV_DATA9	Y/GOUT[1]	Y/GOUT[1]	Y/GOUT[6]	Y/GOUT[6]
DV_DATA8	Y/GOUT[0]	Y/GOUT[0]	Y/GOUT[7]	Y/GOUT[7]
DV_DATA7	Cb/BOUT[7]	Cr/ROUT[7]	Cb/BOUT[0]	Cr/ROUT[0]
DV_DATA6	Cb/BOUT[6]	Cr/ROUT[6]	Cb/BOUT[1]	Cr/ROUT[1]
DV_DATA5	Cb/BOUT[5]	Cr/ROUT[5]	Cb/BOUT[2]	Cr/ROUT[2]
DV_DATA4	Cb/BOUT[4]	Cr/ROUT[4]	Cb/BOUT[3]	Cr/ROUT[3]
DV_DATA3	Cb/BOUT[3]	Cr/ROUT[3]	Cb/BOUT[4]	Cr/ROUT[4]
DV_DATA2	Cb/BOUT[2]	Cr/ROUT[2]	Cb/BOUT[5]	Cr/ROUT[5]
DV_DATA1	Cb/BOUT[1]	Cr/ROUT[1]	Cb/BOUT[6]	Cr/ROUT[6]
DV_DATA0	Cb/BOUT[0]	Cr/ROUT[0]	Cb/BOUT[7]	Cr/ROUT[7]

## (2) RGB666 Input

When the external input is of RGB666 format, the video image signal pins DV\_DATA are allocated to the internal signals GOUT, BOUT, ROUT as shown in Table 32.7.

The internal signals GOUT, BOUT, ROUT to which the video image signal pins DV\_DATA are allocated are output as a 24-bit video image from the RGB666 interface with the following formulae.

$$G[7:0] = GOUT[7:2] \times 255 \div 63$$

$$B[7:0] = BOUT[7:2] \times 255 \div 63$$

$$R[7:0] = ROUT[7:2] \times 255 \div 63$$

**Table 32.7 Bit Allocation of DV\_DATA Pin Inputs When the External Input is of RGB666**

<b>INP_FORMAT[2:0]</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>1</b>
<b>INP_ENDIAN_ON</b>	<b>0</b>	<b>0</b>	<b>1</b>	<b>1</b>
<b>INP_SWAP_ON</b>	<b>0</b>	<b>1</b>	<b>0</b>	<b>1</b>
DV_DATA17	ROUT[7]	BOUT[7]	ROUT[2]	BOUT[2]
DV_DATA16	ROUT[6]	BOUT[6]	ROUT[3]	BOUT[3]
DV_DATA15	ROUT[5]	BOUT[5]	ROUT[4]	BOUT[4]
DV_DATA14	ROUT[4]	BOUT[4]	ROUT[5]	BOUT[5]
DV_DATA13	ROUT[3]	BOUT[3]	ROUT[6]	BOUT[6]
DV_DATA12	ROUT[2]	BOUT[2]	ROUT[7]	BOUT[7]
DV_DATA11	GOUT[7]	GOUT[7]	GOUT[2]	GOUT[2]
DV_DATA10	GOUT[6]	GOUT[6]	GOUT[3]	GOUT[3]
DV_DATA9	GOUT[5]	GOUT[5]	GOUT[4]	GOUT[4]
DV_DATA8	GOUT[4]	GOUT[4]	GOUT[5]	GOUT[5]
DV_DATA7	GOUT[3]	GOUT[3]	GOUT[6]	GOUT[6]
DV_DATA6	GOUT[2]	GOUT[2]	GOUT[7]	GOUT[7]
DV_DATA5	BOUT[7]	ROUT[7]	BOUT[2]	ROUT[2]
DV_DATA4	BOUT[6]	ROUT[6]	BOUT[3]	ROUT[3]
DV_DATA3	BOUT[5]	ROUT[5]	BOUT[4]	ROUT[4]
DV_DATA2	BOUT[4]	ROUT[4]	BOUT[5]	ROUT[5]
DV_DATA1	BOUT[3]	ROUT[3]	BOUT[6]	ROUT[6]
DV_DATA0	BOUT[2]	ROUT[2]	BOUT[7]	ROUT[7]

### (3) RGB565 Input

When the external input is of RGB565 format, the video image signal pins DV\_DATA are allocated to the internal signals GOUT, BOUT, ROUT as shown in Table 32.8.

The internal signals GOUT, BOUT, ROUT to which the video image signal pins DV\_DATA are allocated are output as a 24-bit video image from the RGB565 interface with the following formulae.

$$G[7:0] = GOUT[7:2] \times 255 \div 63$$

$$B[7:0] = BOUT[7:3] \times 255 \div 31$$

$$R[7:0] = ROUT[7:3] \times 255 \div 31$$

**Table 32.8 Bit Allocation of DV\_DATA Pin Inputs When the External Input is of RGB565**

<b>INP_FORMAT[2:0]</b>	<b>2</b>	<b>2</b>	<b>2</b>	<b>2</b>
<b>INP_ENDIAN_ON</b>	<b>0</b>	<b>0</b>	<b>1</b>	<b>1</b>
<b>INP_SWAP_ON</b>	<b>0</b>	<b>1</b>	<b>0</b>	<b>1</b>
DV_DATA15	ROUT[7]	BOUT[7]	ROUT[3]	BOUT[3]
DV_DATA14	ROUT[6]	BOUT[6]	ROUT[4]	BOUT[4]
DV_DATA13	ROUT[5]	BOUT[5]	ROUT[5]	BOUT[5]
DV_DATA12	ROUT[4]	BOUT[4]	ROUT[6]	BOUT[6]
DV_DATA11	ROUT[3]	BOUT[3]	ROUT[7]	BOUT[7]
DV_DATA10	GOUT[7]	GOUT[7]	GOUT[2]	GOUT[2]
DV_DATA9	GOUT[6]	GOUT[6]	GOUT[3]	GOUT[3]
DV_DATA8	GOUT[5]	GOUT[5]	GOUT[4]	GOUT[4]
DV_DATA7	GOUT[4]	GOUT[4]	GOUT[5]	GOUT[5]
DV_DATA6	GOUT[3]	GOUT[3]	GOUT[6]	GOUT[6]
DV_DATA5	GOUT[2]	GOUT[2]	GOUT[7]	GOUT[7]
DV_DATA4	BOUT[7]	ROUT[7]	BOUT[3]	ROUT[3]
DV_DATA3	BOUT[6]	ROUT[6]	BOUT[4]	ROUT[4]
DV_DATA2	BOUT[5]	ROUT[5]	BOUT[5]	ROUT[5]
DV_DATA1	BOUT[4]	ROUT[4]	BOUT[6]	ROUT[6]
DV_DATA0	BOUT[3]	ROUT[3]	BOUT[7]	ROUT[7]

#### (4) BT656/BT601 Input

When the external input is of BT656 or BT601 format, the video image signal pins DV\_DATA are allocated to the internal signal BTOUT, as shown in Table 32.9.

The internal signal BTOUT to which the video image signal pins DV\_DATA are allocated is expanded to the YCbCr signal.

For expansion to the YCbCr signal, see section 32.1.12, BT656/BT601/YCbCr422 Format Setting.

**Table 32.9 Bit Allocation of DV\_DATA Pin Inputs When the External Input is of BT656 or BT601**

INP_FORMAT[2:0]	3 to 4	3 to 4
INP_ENDIAN_ON	0	1
INP_SWAP_ON	0	0
DV_DATA7	BTOUT[7]	BTOUT[0]
DV_DATA6	BTOUT[6]	BTOUT[1]
DV_DATA5	BTOUT[5]	BTOUT[2]
DV_DATA4	BTOUT[4]	BTOUT[3]
DV_DATA3	BTOUT[3]	BTOUT[4]
DV_DATA2	BTOUT[2]	BTOUT[5]
DV_DATA1	BTOUT[1]	BTOUT[6]
DV_DATA0	BTOUT[0]	BTOUT[7]

#### (5) YCbCr422 Input

When the external input is of YCbCr422 format, the video image signal pins DV\_DATA are allocated to the internal signals Y and Cb/Cr, as shown in Table 32.10.

**Table 32.10 Bit Allocation of DV\_DATA Pin Inputs When the External Input is of YCbCr422**

INP_FORMAT[2:0]	5	5	5	5
INP_ENDIAN_ON	0	0	1	1
INP_SWAP_ON	0	1	0	1
DV_DATA15	Y[7]	Cb/Cr[7]	Y[0]	Cb/Cr[0]
DV_DATA14	Y[6]	Cb/Cr[6]	Y[1]	Cb/Cr[1]
DV_DATA13	Y[5]	Cb/Cr[5]	Y[2]	Cb/Cr[2]
DV_DATA12	Y[4]	Cb/Cr[4]	Y[3]	Cb/Cr[3]
DV_DATA11	Y[3]	Cb/Cr[3]	Y[4]	Cb/Cr[4]
DV_DATA10	Y[2]	Cb/Cr[2]	Y[5]	Cb/Cr[5]
DV_DATA9	Y[1]	Cb/Cr[1]	Y[6]	Cb/Cr[6]
DV_DATA8	Y[0]	Cb/Cr[0]	Y[7]	Cb/Cr[7]
DV_DATA7	Cb/Cr[7]	Y[7]	Cb/Cr[0]	Y[0]
DV_DATA6	Cb/Cr[6]	Y[6]	Cb/Cr[1]	Y[1]
DV_DATA5	Cb/Cr[5]	Y[5]	Cb/Cr[2]	Y[2]
DV_DATA4	Cb/Cr[4]	Y[4]	Cb/Cr[3]	Y[3]
DV_DATA3	Cb/Cr[3]	Y[3]	Cb/Cr[4]	Y[4]
DV_DATA2	Cb/Cr[2]	Y[2]	Cb/Cr[5]	Y[5]
DV_DATA1	Cb/Cr[1]	Y[1]	Cb/Cr[6]	Y[6]
DV_DATA0	Cb/Cr[0]	Y[0]	Cb/Cr[7]	Y[7]

### 32.1.8 Typical Signal Timing of BT601 Format

Figure 32.3 and Figure 32.4 show the horizontal timings and Figure 32.5 and Figure 32.6 show the vertical timings of the BT601 format.

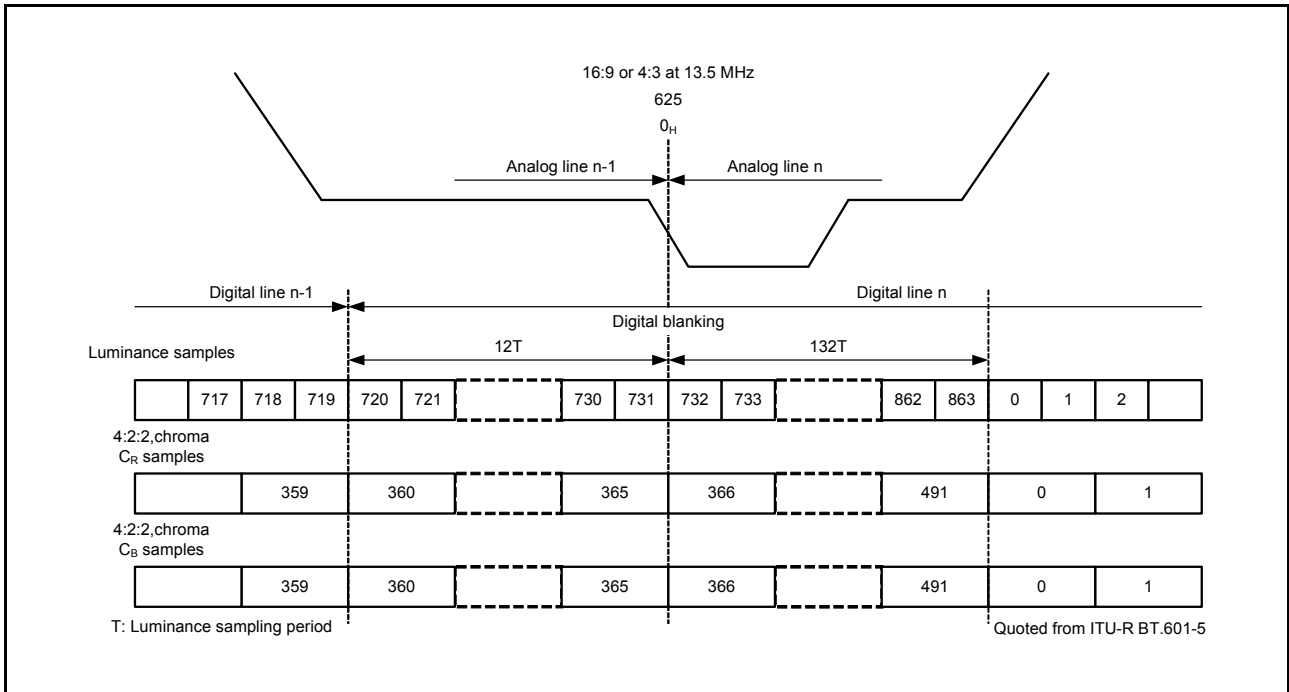


Figure 32.3 BT601 Horizontal Timing (625 Lines/50.00 Hz)

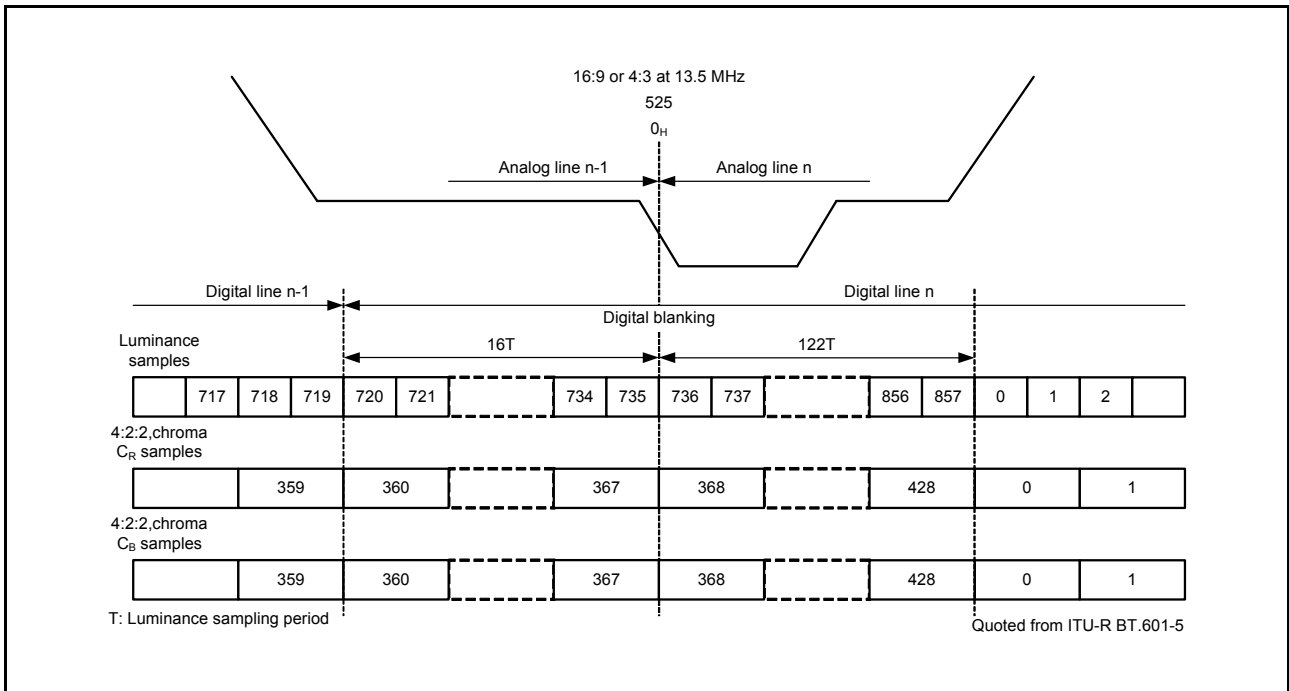


Figure 32.4 BT601 Horizontal Timing (525 Lines/59.94 Hz)

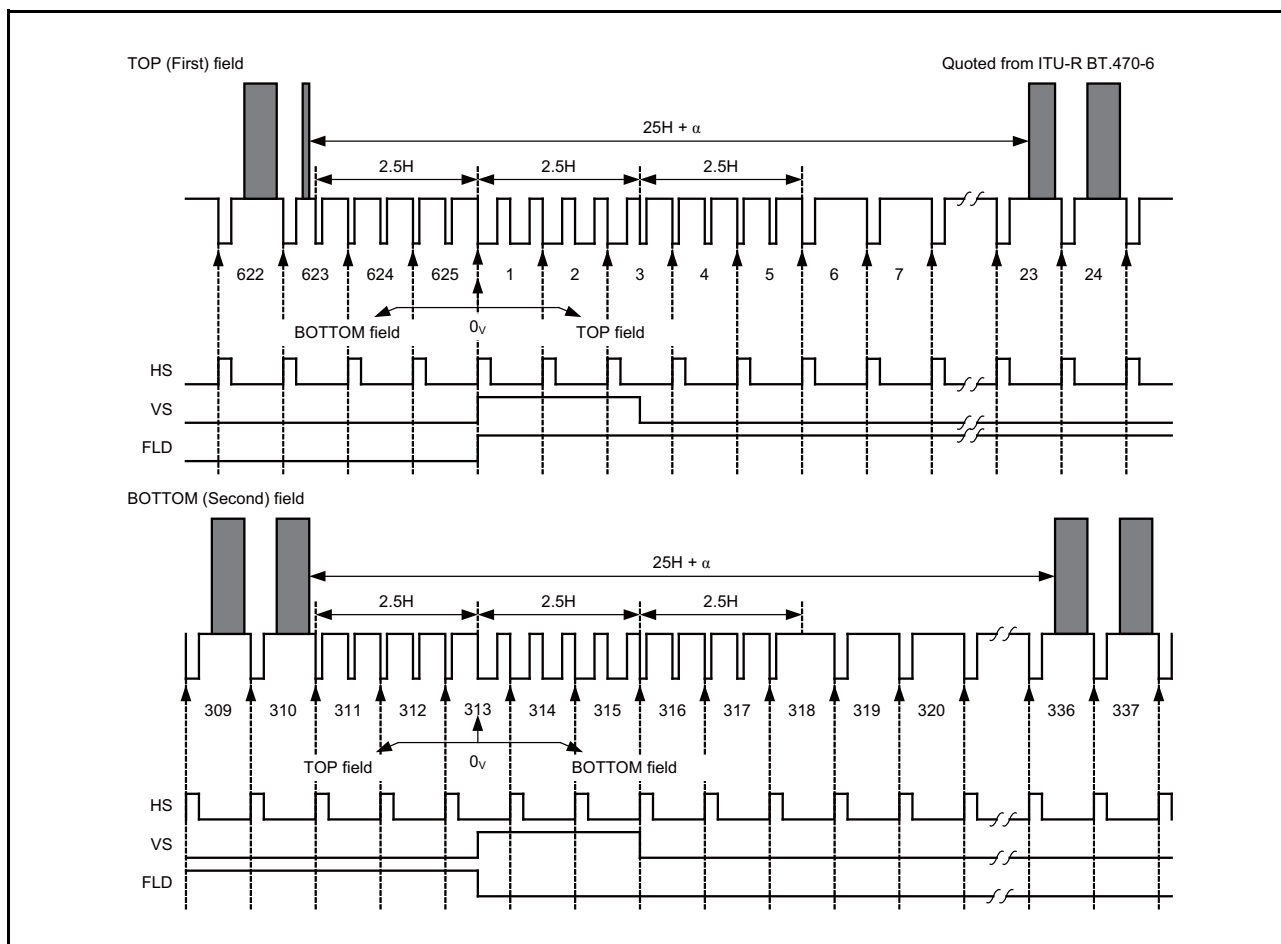


Figure 32.5 BT601 Vertical Timing (625 Lines/50.00 Hz)

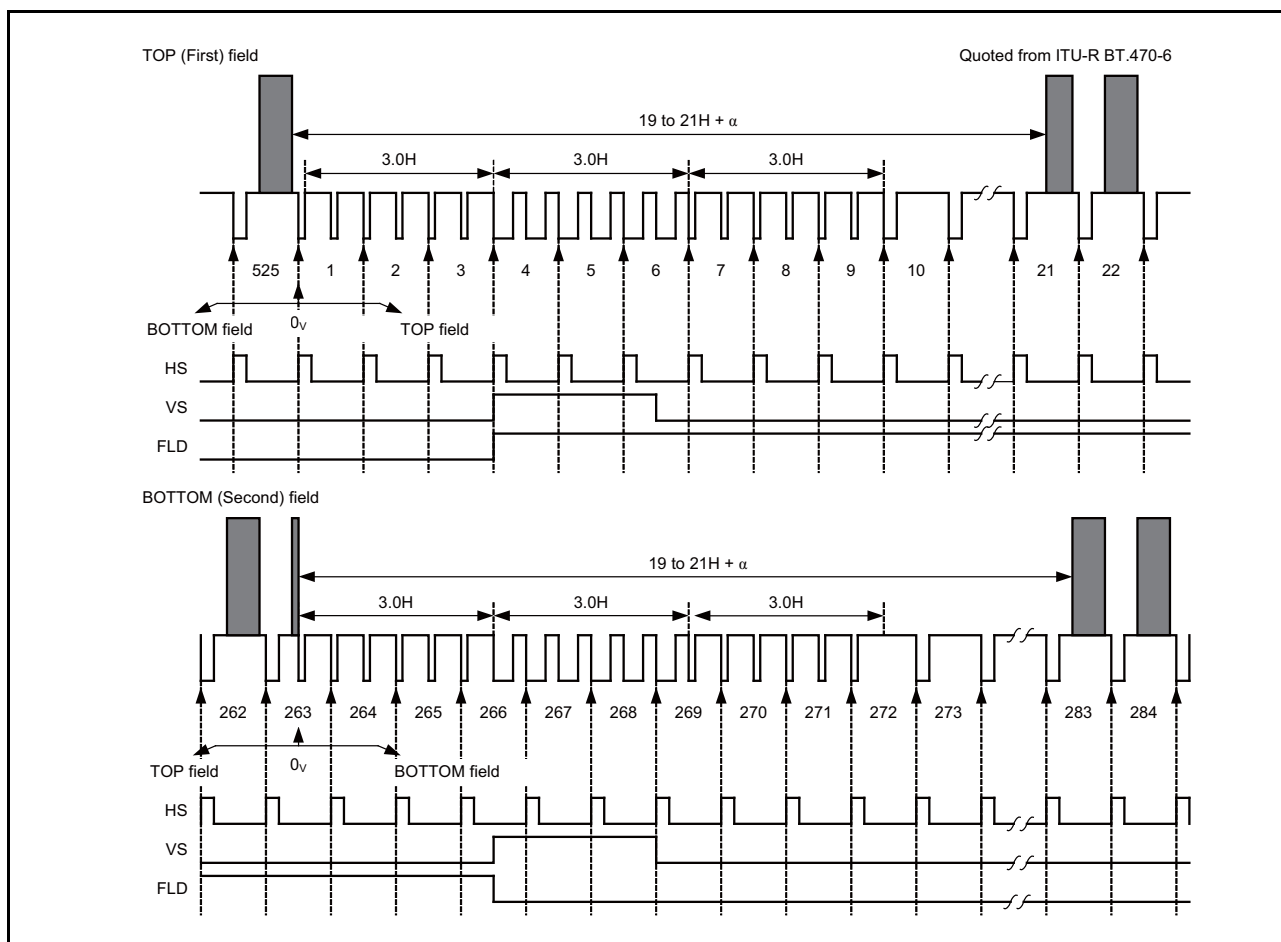


Figure 32.6 BT601 Vertical Timing (525 Lines/59.94 Hz)

### 32.1.9 Typical Signal Timing of BT656 Format

Figure 32.7 and Figure 32.8 show the horizontal timings of the BT656 format.

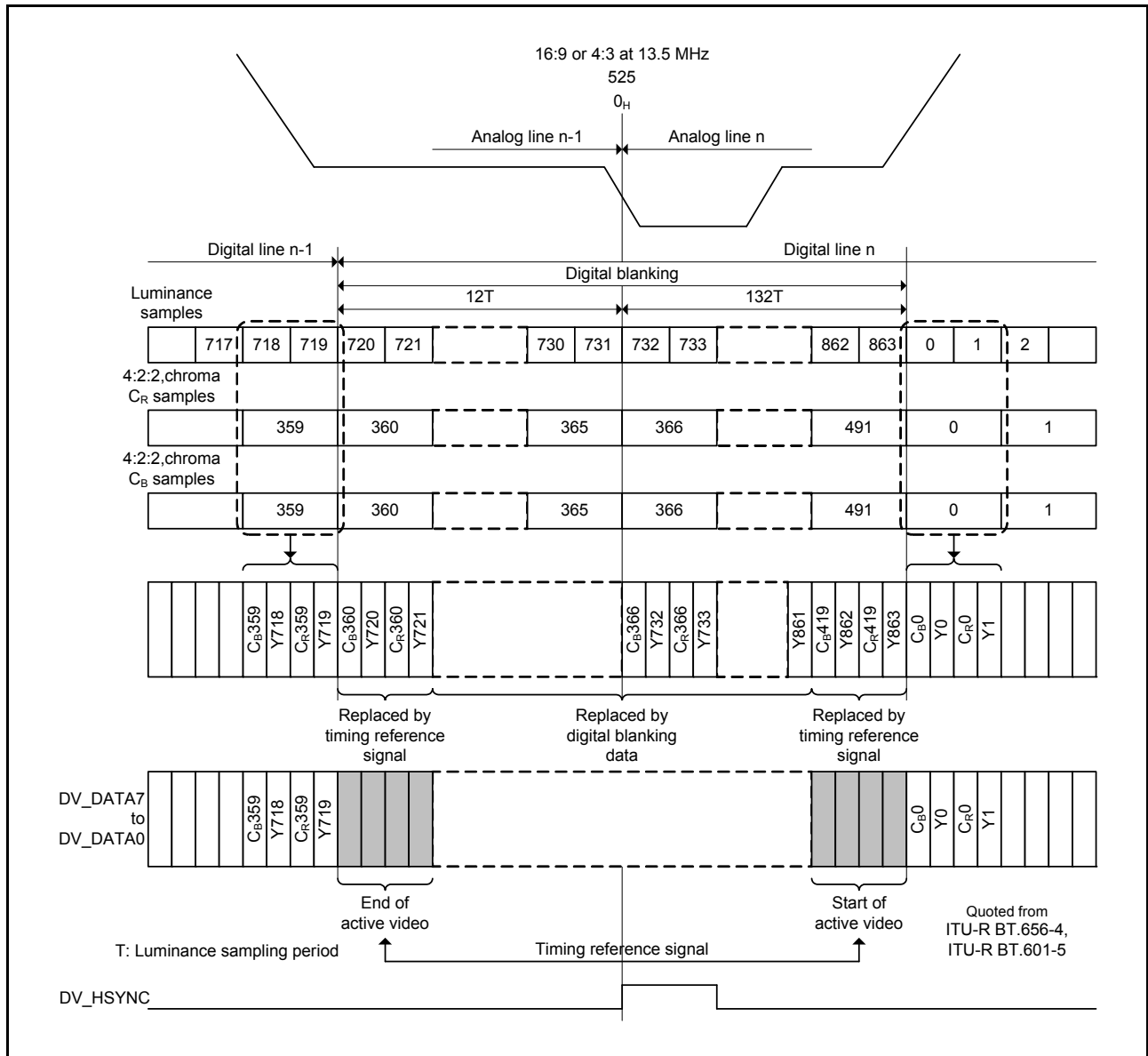


Figure 32.7 BT656 Horizontal Timing (625 Lines/50.00 Hz)



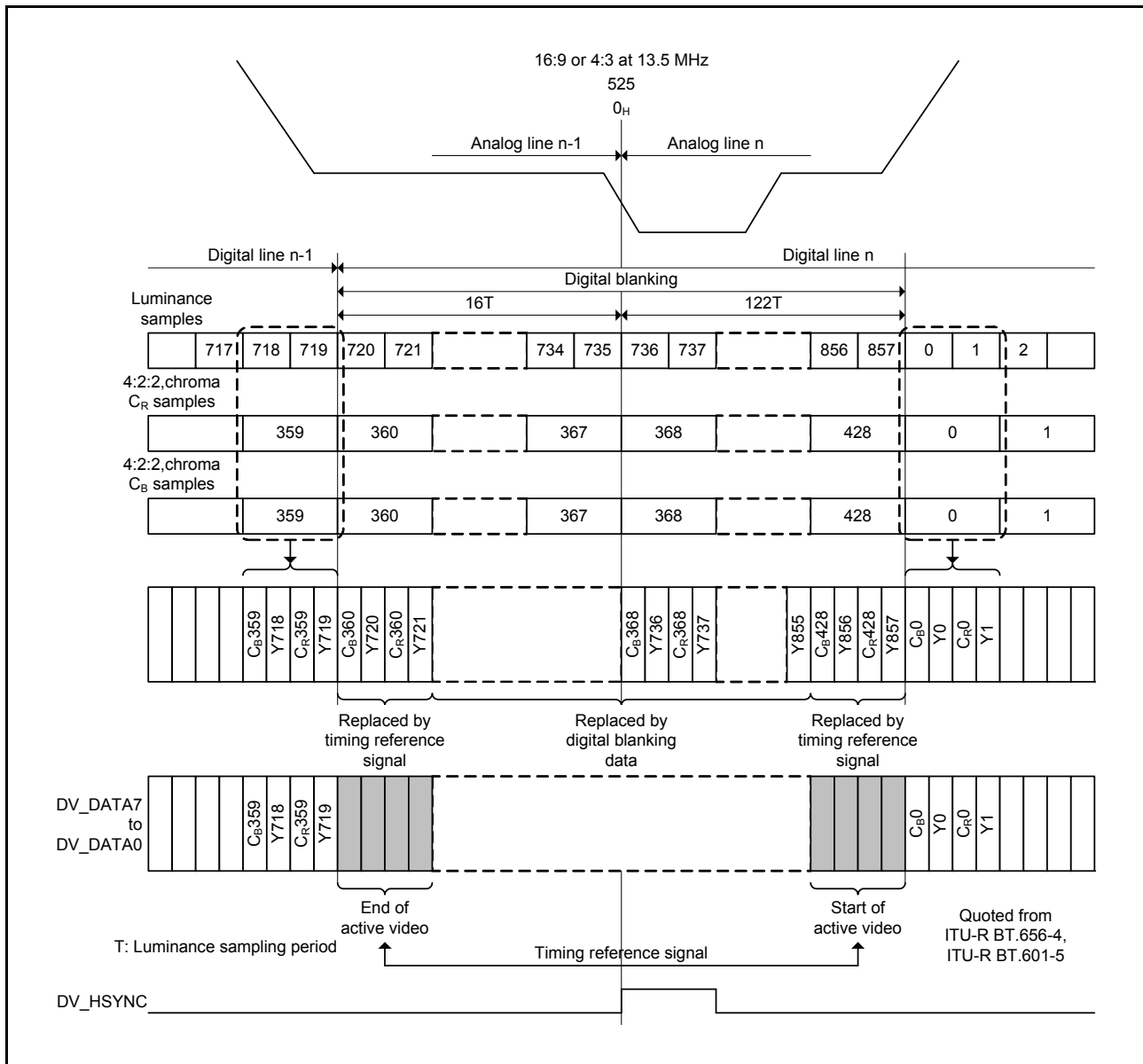


Figure 32.8 BT656 Horizontal Timing (525 Lines/59.94 Hz)

### 32.1.10 SAV/EAV Code in BT656 Format

Table 32.11 shows the timing of inserting the SAV/EAV code in the BT656 format. Bit information is shown in Table 32.12 and Table 32.13. This product does not refer to the parity bits P3 to P0 shown in Table 32.13.

**Table 32.11 SAV/EAV Code Insertion Timing (Line)**

		625	525
<b>V-digital field blanking</b>			
Field 1	Start (V = 1)	Line 624	Line 1
	Finish (V = 0)	Line 23	Line 20
Field 2	Start (V = 1)	Line 311	Line 264
	Finish (V = 0)	Line 336	Line 283
<b>V-digital field blanking</b>			
Field 1	F = 0	Line 1	Line 4
Field 2	F = 1	Line 313	Line 266

**Table 32.12 SAV/EAV Code Bit Information (1)**

Data Bit Number	1st Word (FF)	2nd Word (00)	3rd Word (00)	4th Word (XY)
7 (MSB)	1	0	0	1
6	1	0	0	F
5	1	0	0	V
4	1	0	0	H
3	1	0	0	P3
2	1	0	0	P2
1	1	0	0	P1
0	1	0	0	P0

[Legend]

F = 0 during field 1

F = 1 during field 2

V = 0 elsewhere

V = 1 during field blanking

H = 0 is SAV

H = 1 is EAV

**Table 32.13 SAV/EAV Code Bit Information (2)**

F	V	H	P3	P2	P1	P0
0	0	0	0	0	0	0
0	0	1	1	1	0	1
0	1	0	1	0	1	1
0	1	1	0	1	1	0
1	0	0	0	1	1	1
1	0	1	1	0	1	0
1	1	0	1	1	0	0
1	1	1	0	0	0	1

Figure 32.9 and Figure 32.10 show the SAV/EAV code tables.

		One Horizontal Period																	
		EAV				H blank	SAV				Valid area								
		1	2	3	4		285	286	287	288	289	290	291	292	...	1725	1726	1727	1728
Field1 (top)	1	FF	00	00	B6		FF	00	00	AB	Digital Blanking Data								
	:	FF	00	00	B6		FF	00	00	AB									
	22	FF	00	00	B6		FF	00	00	AB									
	23	FF	00	00	9D		FF	00	00	80									Cb0
	:	FF	00	00	9D	FF	00	00	80	:	Valid pixel data area								
	:	FF	00	00	9D	FF	00	00	80	:									
	:	FF	00	00	9D	FF	00	00	80	:									
	:	FF	00	00	9D	FF	00	00	80	:									
	310	FF	00	00	9D	FF	00	00	80	Cb0	Y0	Cr0	Y1	...	Cb718	Y718	Cr718	Y719	
	311	FF	00	00	B6		FF	00	00	AB	Digital Blanking Data								
312	FF	00	00	B6	FF		00	00	AB										
Field2 (bottom)	313	FF	00	00	F1		FF	00	00	EC	Digital Blanking Data								
	:	FF	00	00	F1		FF	00	00	EC									
	335	FF	00	00	F1		FF	00	00	EC									
	336	FF	00	00	DA		FF	00	00	C7	Cb0	Y0	Cr0	Y1	...	Cb718	Y718	Cr718	Y719
	:	FF	00	00	DA		FF	00	00	C7	:	Valid pixel data area							
	:	FF	00	00	DA		FF	00	00	C7	:								
	:	FF	00	00	DA		FF	00	00	C7	:								
	:	FF	00	00	DA		FF	00	00	C7	:								
	623	FF	00	00	DA		FF	00	00	C7	Cb0	Y0	Cr0	Y1	...	Cb718	Y718	Cr718	Y719
	624	FF	00	00	F1		FF	00	00	EC	Digital Blanking Data								
625	FF	00	00	F1	FF		00	00	EC										

**Figure 32.9 SAV/EAV Code in BT656 Format (625 Lines/50.00 Hz)**

		One Horizontal Period																			
		EAV				H blank	SAV				Valid area										
		1	2	3	4		273	274	275	276	277	278	279	280	...	1713	1714	1715	1716		
Field2	1	FF	00	00	F1		FF	00	00	EC	Digital Blanking Data										
	2	FF	00	00	F1		FF	00	00	EC											
	3	FF	00	00	F1		FF	00	00	EC											
	4	FF	00	00	B6		FF	00	00	AB	Digital Blanking Data										
	:	FF	00	00	B6		FF	00	00	AB											
	19	FF	00	00	B6		FF	00	00	AB											
Field1 (top)	20	FF	00	00	9D		FF	00	00	80	Cb0	Y0	Cr0	Y1	...	Cb718	Y718	Cr718	Y719		
	:	FF	00	00	9D		FF	00	00	80	:										:
	:	FF	00	00	9D		FF	00	00	80	:	Valid pixel data area									:
	:	FF	00	00	9D		FF	00	00	80	:										:
	:	FF	00	00	9D		FF	00	00	80	:										:
	:	FF	00	00	9D		FF	00	00	80	:										:
	263	FF	00	00	9D		FF	00	00	80	Cb0	Y0	Cr0	Y1	...	Cb718	Y718	Cr718	Y719		
	264	FF	00	00	B6		FF	00	00	AB	Digital Blanking Data										
265	FF	00	00	B6	FF	00	00	AB													
	266	FF	00	00	F1		FF	00	00	EC	Digital Blanking Data										
	:	FF	00	00	F1		FF	00	00	EC											
	282	FF	00	00	F1		FF	00	00	EC											
Field2 (bottom)	283	FF	00	00	DA		FF	00	00	C7	Cb0	Y0	Cr0	Y1	...	Cb718	Y718	Cr718	Y719		
	:	FF	00	00	DA		FF	00	00	C7	:										:
	:	FF	00	00	DA		FF	00	00	C7	:	Valid pixel data area									:
	:	FF	00	00	DA		FF	00	00	C7	:										:
	:	FF	00	00	DA		FF	00	00	C7	:										:
	:	FF	00	00	DA		FF	00	00	C7	:										:
525	FF	00	00	DA	FF	00	00	C7	Cb0	Y0	Cr0	Y1	...	Cb718	Y718	Cr718	Y719				

Figure 32.10 SAV/EAV Code in BT656 Format (525 Lines/59.94 Hz)

### 32.1.11 BT656 Progressive Format

This product can be connected with devices which output data in the BT656 progressive format. Because the standard for the BT656 format does not include description of output in the progressive format, there is no guarantee that this product is connected with devices which output data in the progressive format. The following description shows how to generate a vertical/horizontal synchronization signal by decoding the SAV/EAV code input via the BT656 interface of this module. Confirm the connection with devices which output data in the BT656 progressive format in accordance with this section.

#### (1) SAV/EAV Code

The SAV/EAV code consists of four words. When the first word is set to FF and the second and third words are set to 00, timing signals are generated by decoding the value of the fourth word (XY).

For bit information, see Table 32.12 in section 32.1.10, SAV/EAV Code in BT656 Format. This product does not refer to the parity bits (P3 to P0).

#### (2) Vertical/Horizontal Synchronization Signal

Based on the SAV/EAV code, the vertical/horizontal synchronization signal is generated.

##### (a) Vertical Synchronization Signal

The vertical synchronization signal is output when the value of the V bit is changed from 0 to 1 in the BT656 format. The timing of the output varies with the setting of INP\_EXT\_SYNC\_CNT.INP\_F525\_625 setting and the value of the F bit in the BT656 format.

Table 32.14 lists the timing.

**Table 32.14 Timing of Delay for Output of Vertical Synchronization Signal**

INP_EXT_SYNC_CNT. INP_F525_625	F Bit in BT656 Format	Output Timing	Remark
0: 525 lines	0 (Field 1)	2.5 lines after setting of V bit to 1 is detected	525 lines, vertical synchronization signal for field 2
	1 (Field 2)	3 lines after setting of V bit to 1 is detected	525 lines, vertical synchronization signal for field 1
1: 625 lines	0 (Field 1)	2.5 lines after setting of V bit	625 lines, vertical synchronization signal for field 2
	1 (Field 2)	3 lines after setting of V bit to 1 is detected	625 lines, vertical synchronization signal for field 1

##### (b) Horizontal Synchronization Signal

Based on the setting of the INP\_EXT\_SYNC\_CNT.INP\_H\_EDGE\_SEL bit, the horizontal synchronization signal is output.

(c) Timing Example of 525-Line Interface Input in BT656 Format

Figure 32.11 and Figure 32.12 show examples of the timing of vertical/horizontal synchronization signal extracted from 525-line interlaced input in the BT656 format.

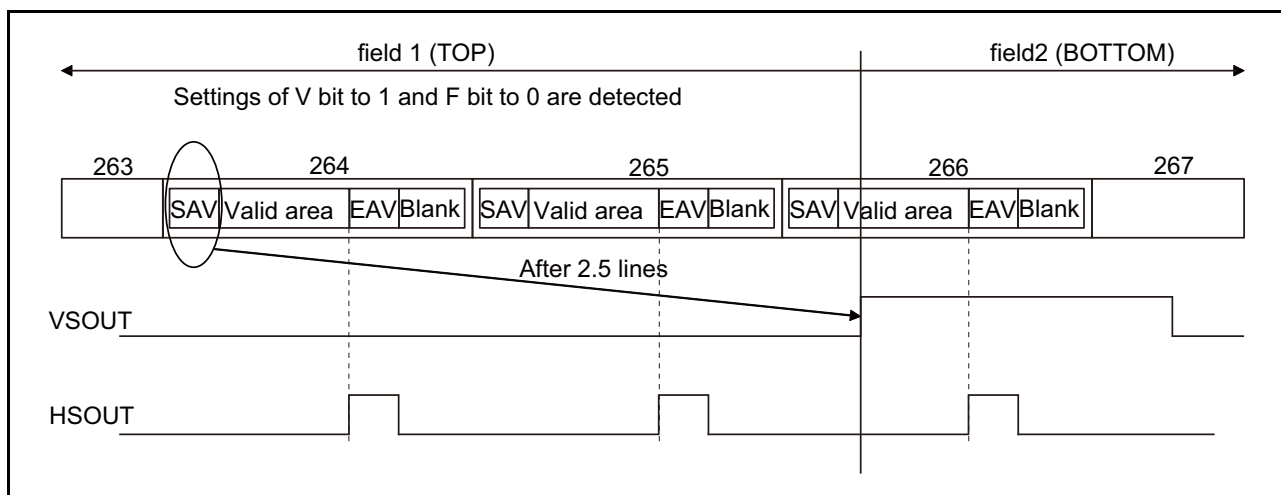


Figure 32.11 Timing of Vertical/Horizontal Synchronization Signal from 525-Line Interlaced Input in BT656 Format (Top to Bottom)

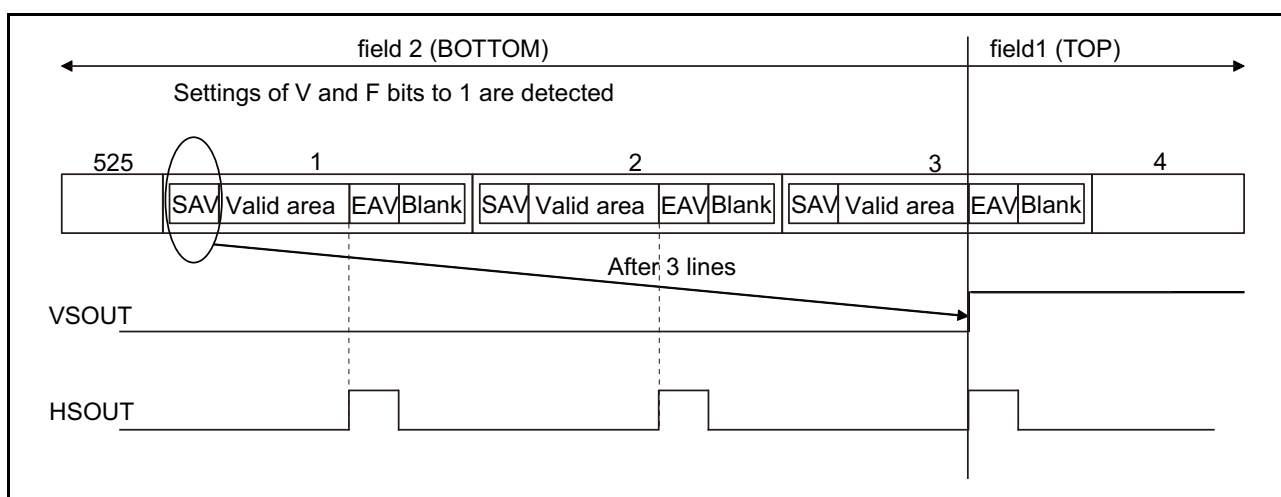


Figure 32.12 Timing of Vertical/Horizontal Synchronization Signal from 525-Line Interlaced Input in BT656 Format (Bottom to Top)

(3) Example of Timing for Progressive Input in BT656 Format

Figure 32.13 shows an example of the SAV/EAV code in 525-line progressive input in the BT656 format. Figure 32.14 shows the vertical/horizontal synchronization signal extracted from 525-line progressive input in the BT656 format. The field is detected as field 1 in this example, because the value of the F bit is set to 0 when that of the V bit is changed from 0 to 1. The field is regarded as the bottom field. The vertical synchronization signal is output 2.5 lines after the detection of the SAV code.

		One Horizontal Period																					
		EAV				H blank				SAV				Valid area									
		1	2	3	4	273	274	275	276	277	278	279	280	...	1713	1714	1715	1716					
Field1 (top)	1	FF	00	00	BX	H blank	FF	00	00	AX	Digital Blanking Data												
	:	FF	00	00	BX		FF	00	00	AX													
	19	FF	00	00	BX		FF	00	00	AX													
	20	FF	00	00	9X		H blank	FF	00	00													8X
	:	FF	00	00	9X	FF		00	00	8X	:	:											
	:	FF	00	00	9X	FF		00	00	8X	:	:											
	:	FF	00	00	9X	FF		00	00	8X	:	Valid pixel data area								:			
	:	FF	00	00	9X	FF		00	00	8X	:	:											
	:	FF	00	00	9X	FF		00	00	8X	:	:											
	504	FF	00	00	9X	FF		00	00	8X	Cb0	Y0	Cr0	Y1	...	Cb718	Y718	Cr718	Y719				
	505	FF	00	00	BX	H blank		FF	00	00	AX	Digital Blanking Data											
	:	FF	00	00	BX			FF	00	00	AX												
	525	FF	00	00	BX			FF	00	00	AX												

X: Not referred

Figure 32.13 SAV/EAV Code in BT656 Progressive Format (525 Lines, 59.94 Hz)

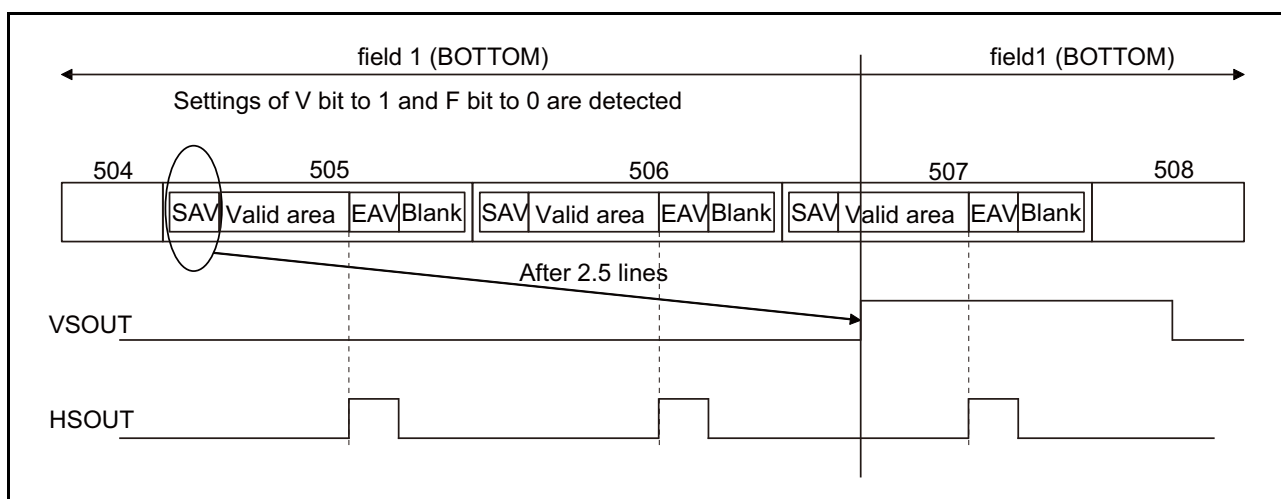


Figure 32.14 Vertical/Horizontal Synchronization Signal in BT656 Format (525 Lines, Progressive)

### 32.1.12 BT656/BT601/YCbCr422 Format Setting

The BT656 format can be used for the 525-line and 59.94-Hz or the 625-line and 50.00-Hz interlace signal format and progressive signal format (extended).

The YCbCr422 format can be used for the 525-line and 59.94-Hz or the 625-line and 50.00-Hz interlace signal format in the 16-bit data-bus format of the BT601 standard.

The Vsync signal timing for the 525-line BT656 format and 625-line BT656 format are different.

The operating mode is set by the INP\_F525\_625 bit.

**Table 32.15 Operating Mode Setting for BT656 Format**

Register Name	Bit Name	Initial Value	Description
INP_EXT_SYNC_CNT	INP_F525_625	0	Number of Lines for BT656 Input of External Input System 0: 525 lines 1: 625 lines

When the interlace signals are to be input in BT656/BT601/YCbCr422 format, half of 2fH phase timings of the Vsync signal and the Hsync signal are set with the INP\_FH50[9:0] bits.

The INP\_FH50[9:0] bits are also used for the vertical synchronous phase adjustment block. Therefore, for bit description, see Table 32.20.

When the external input is of BT656 format, the reference point of the Hsync signal is set with the INP\_H\_EDGE\_SEL bit.

**Table 32.16 Hsync Signal Reference Selection for BT656 Format**

Register Name	Bit Name	Initial Value	Description
INP_EXT_SYNC_CNT	INP_H_EDGE_SEL	0	Hsync Signal Reference Select for BT656 Format of External Input System 0: EAV 1: SAV

When the external input is of BT656/BT601 format, the internal signal BTOUT[7:0], which is input from the DV\_DATA pins and allocated, is expanded to the 24-bit YCbCr signal.

Expansion timing with respect to the Hsync signal reference is set with the INP\_H\_POS[1:0] bits.



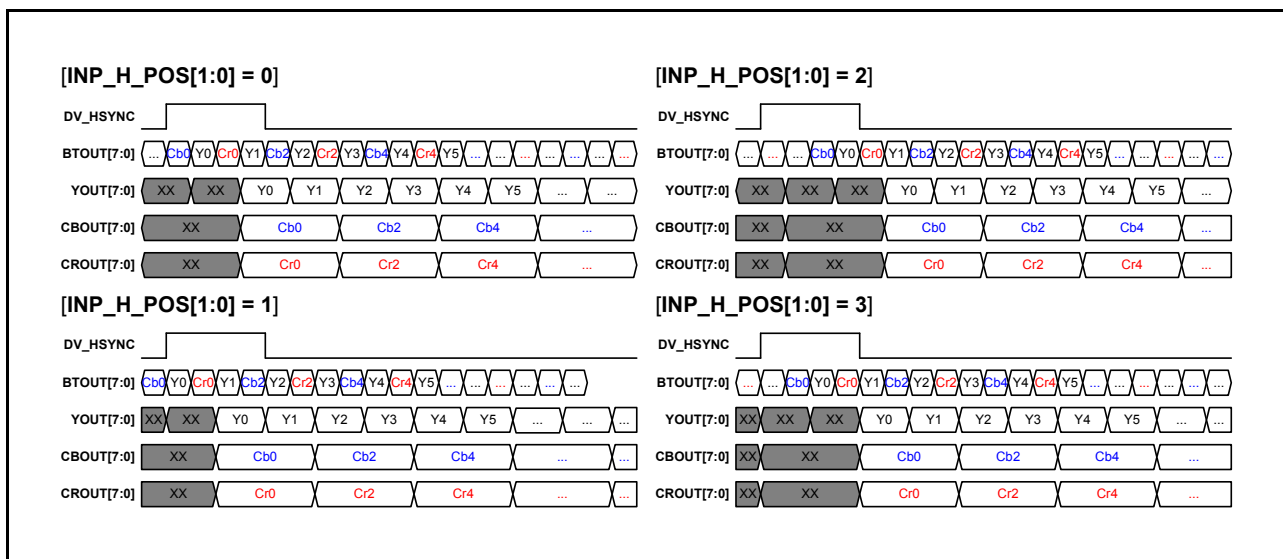


Figure 32.15 YCbCr Data Expansion for BT656/BT601 Input

Table 32.17 Data String Start Timing Selection for BT656/BT601 Input

Register Name	Bit Name	Initial Value	Description
INP_EXT_SYNC_CNT	INP_H_POS[1:0]	0	Y/Cb/Y/Cr Data String Start Timing with respect to Hsync Reference 0: Cb/Y/Cr/Y 1: Y/Cr/Y/Cb 2: Cr/Y/Cb/Y 3: Y/Cb/Y/Cr

When the external input is in YCbCr422 format, the input from the DV\_DATA pins is allocated to the internal Y[7:0] and CbCr[7:0] signals, and the CbCr[7:0] are expanded to a 16-bit signal.

Expansion timing with respect to the Hsync signal reference is set with the INP\_H\_POS[1:0] bits.

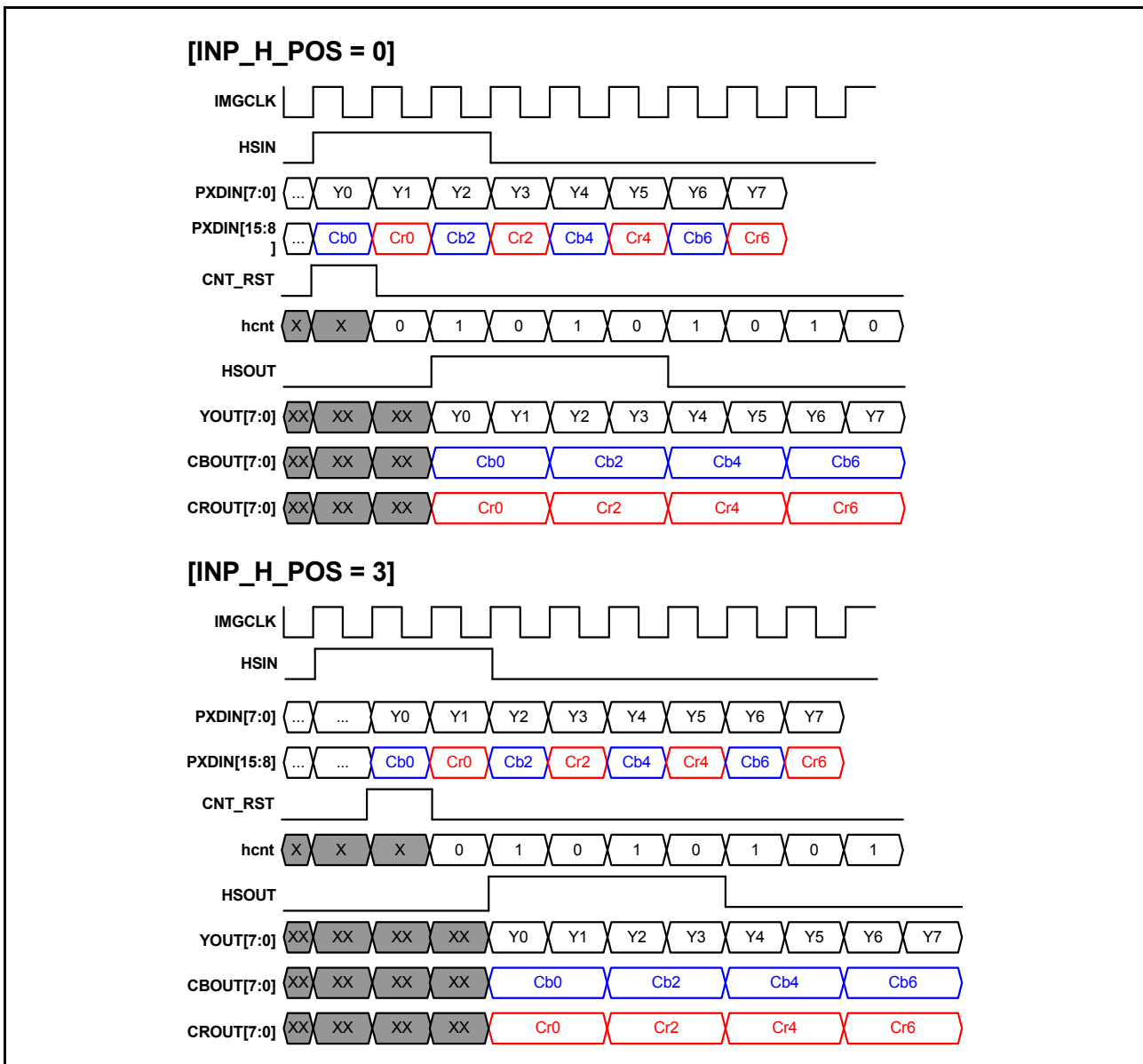


Figure 32.16 YCbCr Data Expansion for YCbCr422 Input

Table 32.18 Data String Start Timing Selection for YCbCr422 Input

Register Name	Bit Name	Initial Value	Description
INP_EXT_SYNC_CNT	INP_H_POS[1:0]	0	Cb/Cr Data String Start Timing with respect to Hsync Reference 0: Cb/Cr 3: Cr/Cb 1, 2: Setting prohibited

### 32.1.13 YCbCr444/RBG888/666/565 Input Timing

The YCbCr444/RGB888/666/565 format can be used for the progressive YCbCr/RGB signal.

The sync signal width (H\_SYNC, V\_SYNC), sync signal polarity (H\_POL, V\_POL), valid period start position (H\_BP, V\_BP), valid period end position (H\_FP, V\_FP), and valid period video width (H\_ACTIVE, V\_ACTIVE) are shown in Table 32.19.

**Table 32.19 YCbCr/RGB Signal Reception Timing**

Item	Description
External input clock	Maximum external input clock frequency: 87.00 MHz
Vsync signal width (V_SYNC)	Minimum Vsync signal width: 1 CLK
Vsync signal polarity (V_POL)	Positive or negative polarity is selected by the relevant registers.
Vertical valid period start position (V_BP)	From Vsync reference to the head of the video image: 5 lines or more
Vertical valid period video width (V_ACTIVE)	Maximum vertical valid period: 1024 lines
Vertical valid period end position (V_FP)	From the end of the video image to the Vsync reference: 4 lines or more*1
Hsync signal width (H_SYNC)	Minimum Hsync signal width: 1 CLK
Hsync signal polarity (H_POL)	Positive or negative polarity is selected by the relevant registers.
Horizontal valid period start position (H_BP)	From Hsync reference to the head of the video image: 16 CLK or more
Horizontal valid period video width (H_ACTIVE)	Maximum horizontal valid period: 1440 pixels
Horizontal valid period end position (H_FP)	From the end of the video image to the Hsync reference: 16 CLK or more*2
Number of vertical lines (V_BP+V_ACTIVE+V_FP)	Between vertical synchronization signals: at least 2047 lines
Number of horizontal pixels (H_BP+H_ACTIVE+H_FP)	Between horizontal synchronization signals: at least 2047 CLK

Note 1. When V\_FP is below 4 lines, the setting of INP\_DLY\_ADJ.INP\_VS\_DLY\_L[2:0] should be adjusted so that V\_FP is at least 4 lines.

Note 2. When H\_FP is below 16 CLK, the settings of INP\_DLY\_ADJ.INP\_VS\_DLY[7:0], INP\_HS\_DLY[7:0], and INP\_FLD\_DLY[7:0] should be adjusted so that H\_FP is at least 16 CLK.

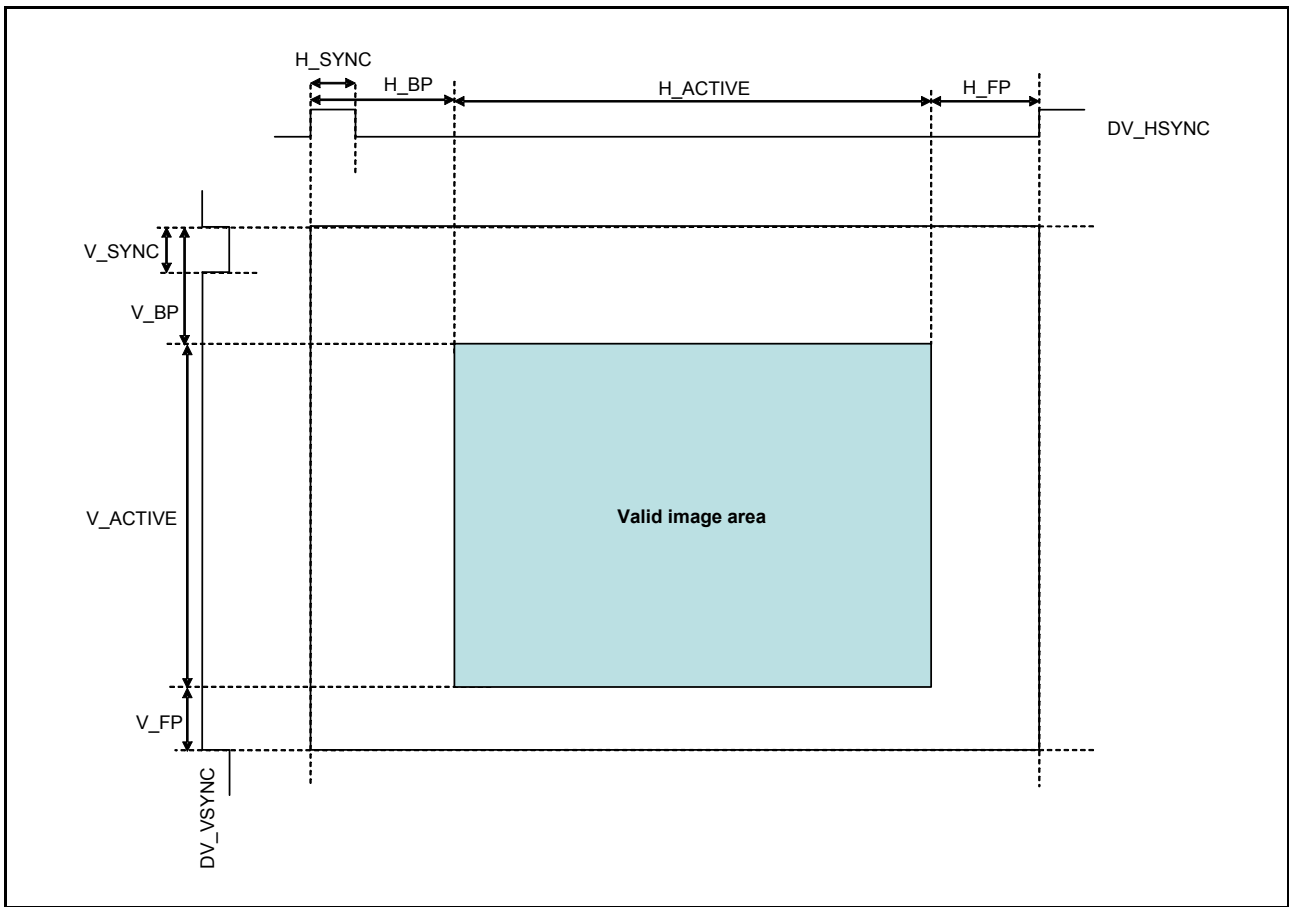
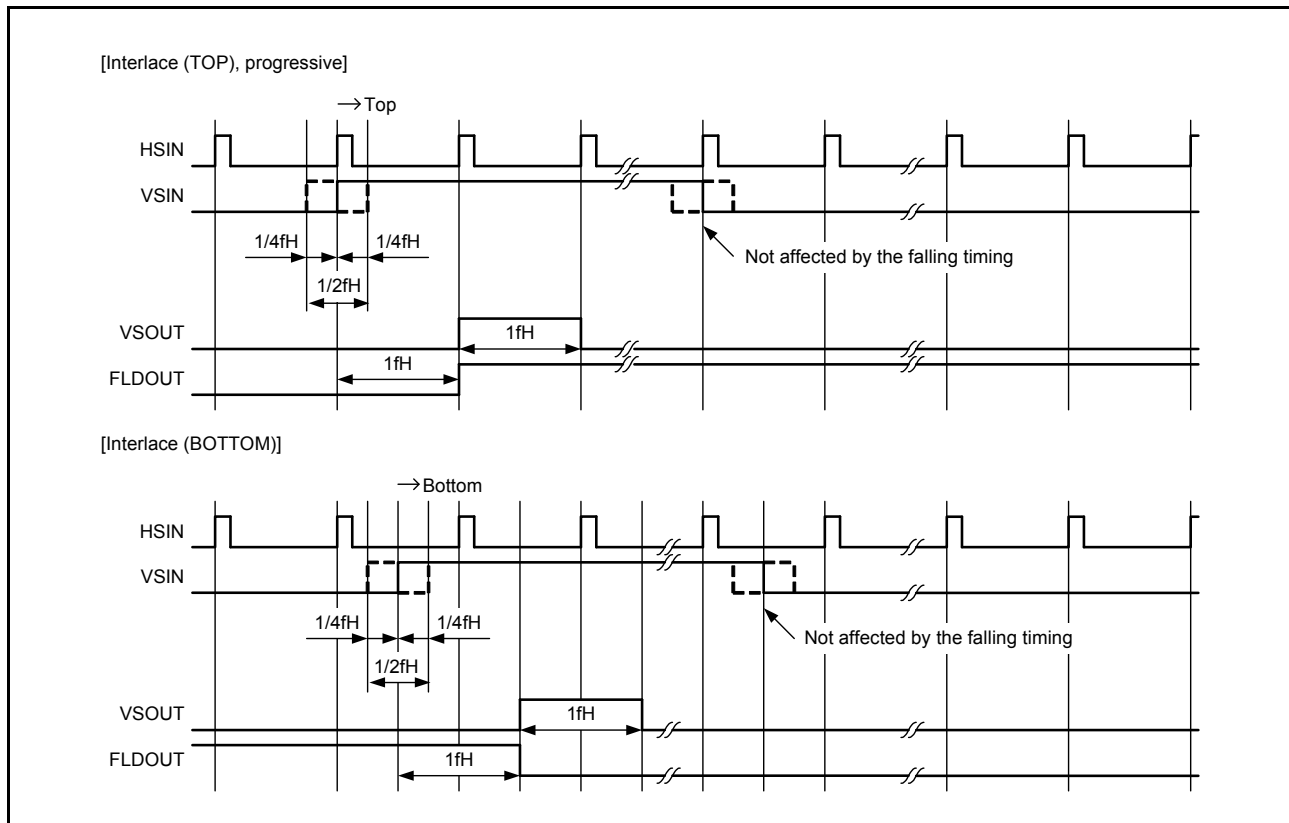


Figure 32.17 YCbCr/RGB Signal Reception Timing

### 32.1.14 Field Differentiation and Vsync Signal Phase Adjustment

The phase of the input Vsync signal and Hsync signal is detected and the field of the interlace signal is determined. When the reference point of the Vsync signal is detected within  $\pm 0.5$  horizontal period with respect to the Hsync signal, it is determined as the interlace top field. When the reference point of the Vsync signal is detected outside  $\pm 0.5$  horizontal period with respect to the Hsync signal, it is determined as the interlace bottom field.



**Figure 32.18 Vsync Signal Phase Adjustment**

The timings of  $1/2fH$  Vsync signal phase and  $1/4fH$  Vsync signal phase are set with `INP_FH50[9:0]` and `INP_FH25[9:0]`, respectively.

**Table 32.20 Vsync Signal Phase Timing Setting**

Register Name	Bit Name	Initial Value	Description
<code>INP_VSYNC_PH_ADJ</code>	<code>INP_FH50[9:0]</code>	858	Vsync Signal $1/2fH$ Phase Timing Should be $1/2$ the horizontal cycle.
<code>INP_VSYNC_PH_ADJ</code>	<code>INP_FH25[9:0]</code>	429	Vsync Signal $1/4fH$ Phase Timing Should be $1/4$ the horizontal cycle.

### 32.1.15 Vsync Signal Delay Adjustment in Line Units

The Vsync signal line delay adjust block can delay the Vsync signal and the field differentiation signal in line units.

When a video signal with a short vertical front porch is input, the vertical front porch is adjusted.

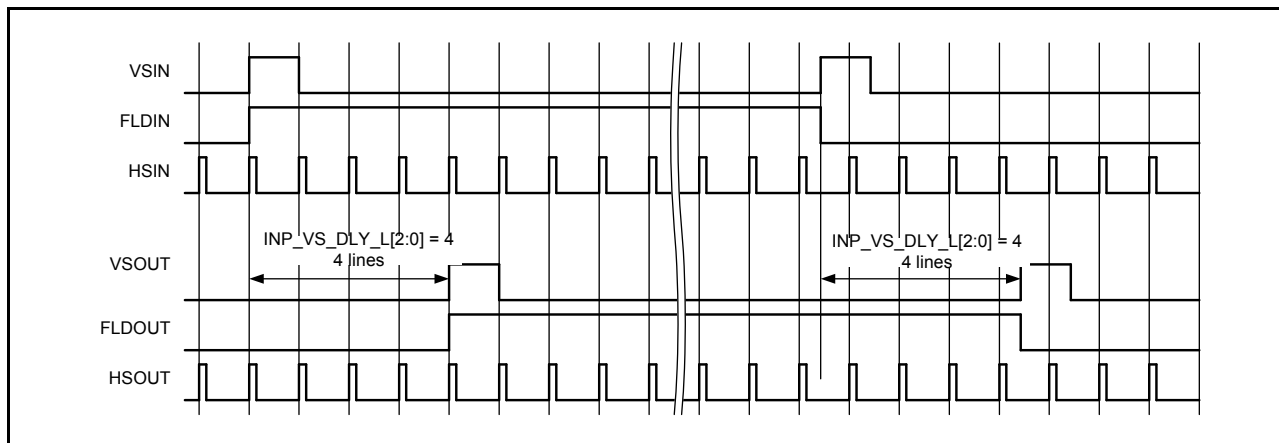


Figure 32.19 Timing of Vsync Signal Delay in Line Units

Table 32.21 Adjustment of Vsync Signal Delay in Line Units

Register Name	Bit Name	Initial Value	Description
INP_DLY_ADJ	INP_VS_DLY_L[2:0]	0	Number of Lines for Delaying Vsync Signal and Field Differentiation Signal Delay amount: 0 to 7 (lines)

### 32.1.16 Sync Signal Delay Adjustment

Delay can be adjusted independently for the Vsync signal, Hsync signal, and field differentiation signal in the units of clock.

Lacking margin of the horizontal front porch is adjusted according to the input synchronization disturbance.

Table 32.22 Sync Signal Delay Adjustment

Register Name	Bit Name	Initial Value	Description
INP_DLY_ADJ	INP_VS_DLY[7:0]	0	Vsync Signal Delay Amount Delay amount: 0 to 254 (clock cycles)
INP_DLY_ADJ	INP_HS_DLY[7:0]	0	Hsync Signal Delay Amount Delay amount: 0 to 254 (clock cycles)
INP_DLY_ADJ	INP_FLD_DLY[7:0]	0	Field Differentiation Signal Delay Amount Delay amount: 0 to 254 (clock cycles)

### 32.1.17 Horizontal Noise Reduction

Noise can be reduced according to horizontal pixel reference.

Noise reduction is controlled through noise component frequency band (TAP), noise level (threshold), and noise reduction intensity (gain).

#### (1) Frequency Band (TAP) Setting for Noise Component

The noise frequency band can be selected independently from the following four types by using the NR1D\_Y\_TAP[1:0], NR1D\_CB\_TAP[1:0], and NR1D\_CR\_TAP[1:0] bits.

When the number of adjacent pixels is one (noise reduction NR1D\_Y/CB/CR\_TAP is 0):

$$BPF_{(1)} = \frac{1}{4}(-1 \times Z_{(-1)}, 2 \times Z_{(0)}, -1 \times Z_{(+1)})$$

When the number of adjacent pixels is two (noise reduction NR1D\_Y/CB/CR\_TAP is 1):

$$BPF_{(2)} = \frac{1}{4}(-1 \times Z_{(-2)}, 2 \times Z_{(0)}, -1 \times Z_{(+2)})$$

When the number of adjacent pixels is three (noise reduction NR1D\_Y/CB/CR\_TAP is 2):

$$BPF_{(3)} = \frac{1}{4}(-1 \times Z_{(-3)}, 2 \times Z_{(0)}, -1 \times Z_{(+3)})$$

When the number of adjacent pixels is four (noise reduction NR1D\_Y/CB/CR\_TAP is 3):

$$BPF_{(4)} = \frac{1}{4}(-1 \times Z_{(-4)}, 2 \times Z_{(0)}, -1 \times Z_{(+4)})$$

Note:  $Z_{(0)}$  indicates the target pixel for noise reduction and  $Z_{(N)}$  indicates the pixel that is n pixels off from  $Z_{(0)}$  in the horizontal direction.

#### (2) Setting Noise Level (Threshold)

The absolute value of the detected noise amount (BPF output value) is compared with the values of the NR1D\_Y\_TH[6:0], NR1D\_CB\_TH[6:0], and NR1D\_CR\_TH[6:0] bits. When the detected noise amount is greater than NR1D\_Y/CB/CR\_TH, the absolute value of the detected noise amount is considered as NR1D\_Y/CB/CR\_TH (fixed value).

$$\text{ABS}(BPF_{(n)}) \leq \text{absolute value of detected noise amount when } \text{ABS}(BPF_{(n)}) \leq \text{NR1D\_Y/CB/CR\_TH: NOISE\_ABS} \\ = \text{ABS}(BPF_{(n)})$$

$$\text{ABS}(BPF_{(n)}) > \text{absolute value of detected noise amount when } \text{ABS}(BPF_{(n)}) > \text{NR1D\_Y/CB/CR\_TH: NOISE\_ABS} \\ = \text{NR1D\_Y/CB/CR\_TH}$$

#### (3) Setting Noise Reduction Intensity (Gain)

The absolute value of the detected noise amount is multiplied by the value of gain specified by the NR1D\_Y\_GAIN[1:0], NR1D\_CB\_GAIN[1:0], and NR1D\_CR\_GAIN[1:0] bits, and the feedback is calculated for the original signal.

$$\text{Computation when the amount of detected noise } (BPF_{(n)}) \text{ is negative } (-): \text{DOUT} = \text{DIN} + \text{NOISE\_ABS} \div \\ 2^{(\text{NR1D\_Y/CB/CR\_GAIN}+1)}$$

Computation when the amount of detected noise ( $BPF_{(n)}$  is positive (+):  $DOUT = DIN - NOISE\_ABS \div 2^{(NR1D\_Y/CB/CR\_GAIN+1)}$

**Table 32.23 Horizontal Noise Reduction**

Register Name	Bit Name	Initial Value	Description
IMGCNT_NR_CNT0	NR1D_MD	1	Horizontal Noise Reduction Operating Mode 0: R/G/B mode 1: Y/Cb/Cr mode
IMGCNT_NR_CNT0	NR1D_ON	0	Noise Reduction On/Off Control 0: Noise Reduction Off 1: Noise Reduction On
IMGCNT_NR_CNT0	NR1D_Y_TAP[1:0]	0	Y/G Signal TAP Select 0: Adjacent pixel 1: 2 adjacent pixels 2: 3 adjacent pixels 3: 4 adjacent pixels
IMGCNT_NR_CNT0	NR1D_Y_TH[6:0]	8	Maximum Value (Absolute Value) of Y/G Signal Coring Coring is implemented when detected noise amount value $\leq$ NR1D_Y_TH. Unsigned: 0 to 127 [LSB]
IMGCNT_NR_CNT0	NR1D_Y_GAIN[1:0]	3	Noise Reduction Gain Adjustment of Y/G Signal 0: 1/2 1: 1/4 2: 1/8 3: 1/16
IMGCNT_NR_CNT1	NR1D_CB_TAP[1:0]	0	Cb/B Signal TAP Select 0: Adjacent pixel 1: 2 adjacent pixels 2: 3 adjacent pixels 3: 4 adjacent pixels
IMGCNT_NR_CNT1	NR1D_CB_TH[6:0]	8	Maximum Value (Absolute Value) of Cb/B Signal Coring Coring is implemented when detected noise amount value $\leq$ NR1D_C_TH. Unsigned: 0 to 127 [LSB]
IMGCNT_NR_CNT1	NR1D_CB_GAIN[1:0]	3	Noise Reduction Gain Adjustment of Cb/B Signal 0: 1/2 1: 1/4 2: 1/8 3: 1/16
IMGCNT_NR_CNT1	NR1D_CR_TAP[1:0]	0	Cr/R Signal TAP Select 0: Adjacent pixel 1: 2 adjacent pixels 2: 3 adjacent pixels 3: 4 adjacent pixels
IMGCNT_NR_CNT1	NR1D_CR_TH[6:0]	8	Maximum Value (Absolute Value) of Cr/R Signal Coring Coring is implemented when detected noise amount value $\leq$ NR1D_C_TH. Unsigned: 0 to 127 [LSB]
IMGCNT_NR_CNT1	NR1D_CR_GAIN[1:0]	3	Noise Reduction Gain Adjustment of Cr/R Signal 0: 1/2 1: 1/4 2: 1/8 3: 1/16



### 32.1.18 Color Matrix

By using a color matrix, input signal offsets and nine-axis gain can be adjusted. This enables brightness adjustment, gain adjustment, and YCbCr and GBR mutual conversion.

#### (1) GBR to GBR Conversion

$$YGIN\_A = YGIN + IMGCNT\_MTX\_YG - 128$$

$$CBBIN\_A = CBBIN + IMGCNT\_MTX\_B - 128$$

$$CRRIN\_A = CRRIN + IMGCNT\_MTX\_R - 128$$

$$YGOUT = (IMGCNT\_MTX\_GG \times YGIN\_A + IMGCNT\_MTX\_GB \times CBBIN\_A + IMGCNT\_MTX\_GR \times CRRIN\_A) \div 256$$

$$CBBOUT = (IMGCNT\_MTX\_BG \times YGIN\_A + IMGCNT\_MTX\_BB \times CBBIN\_A + IMGCNT\_MTX\_BR \times CRRIN\_A) \div 256$$

$$CRROUT = (IMGCNT\_MTX\_RG \times YGIN\_A + IMGCNT\_MTX\_RB \times CBBIN\_A + IMGCNT\_MTX\_RR \times CRRIN\_A) \div 256$$

#### (2) GBR to YCbCr Conversion

$$YGIN\_A = YGIN + IMGCNT\_MTX\_YG - 128$$

$$CBBIN\_A = CBBIN + IMGCNT\_MTX\_B - 128$$

$$CRRIN\_A = CRRIN + IMGCNT\_MTX\_R - 128$$

$$YGOUT = (IMGCNT\_MTX\_GG \times YGIN\_A + IMGCNT\_MTX\_GB \times CBBIN\_A + IMGCNT\_MTX\_GR \times CRRIN\_A) \div 256$$

$$CBBOUT = (IMGCNT\_MTX\_BG \times YGIN\_A + IMGCNT\_MTX\_BB \times CBBIN\_A + IMGCNT\_MTX\_BR \times CRRIN\_A) \div 256 + 128$$

$$CRROUT = (IMGCNT\_MTX\_RG \times YGIN\_A + IMGCNT\_MTX\_RB \times CBBIN\_A + IMGCNT\_MTX\_RR \times CRRIN\_A) \div 256 + 128$$

**Table 32.24 Matrix Coefficient (Typical Value) for SMPTE 293M**

	YGIN		CBBIN		CRRIN	
	Coefficient	Set Value	Coefficient	Set Value	Coefficient	Set Value
YGOUT	0.587	IMGCNT_MTX_GG = 150	0.114	IMGCNT_MTX_GB = 29	0.299	IMGCNT_MTX_GR = 77
CBBOUT	-0.331	IMGCNT_MTX_BG = 1963	0.500	IMGCNT_MTX_BB = 128	-0.169	IMGCNT_MTX_BR = 2005
CRROUT	-0.419	IMGCNT_MTX_RG = 1941	-0.081	IMGCNT_MTX_RB = 2027	0.500	IMGCNT_MTX_RR = 128

## (3) YCbCr to GBR Conversion

$$YGIN\_A = YGIN + IMGCNT\_MTX\_YG - 128$$

$$CBBIN\_A = CBBIN - 128$$

$$CRRIN\_A = CRRIN - 128$$

$$YGOUT = (IMGCNT\_MTX\_GG \times YGIN\_A + IMGCNT\_MTX\_GB \times CBBIN\_A + IMGCNT\_MTX\_GR \times CRRIN\_A) \div 256$$

$$CBBOUT = (IMGCNT\_MTX\_BG \times YGIN\_A + IMGCNT\_MTX\_BB \times CBBIN\_A + IMGCNT\_MTX\_BR \times CRRIN\_A) \div 256$$

$$CRROUT = (IMGCNT\_MTX\_RG \times YGIN\_A + IMGCNT\_MTX\_RB \times CBBIN\_A + IMGCNT\_MTX\_RR \times CRRIN\_A) \div 256$$

**Table 32.25 Matrix Coefficient (Typical Value) for SMPTE 293M**

	YGIN		CBBIN		CRRIN	
	Coefficient	Set Value	Coefficient	Set Value	Coefficient	Set Value
YGOUT	1.000	IMGCNT_ MTX_GG = 256	-0.344	IMGCNT_MTX_GB = 1960	-0.714	IMGCNT_MTX_GR = 1865
CBBOUT	1.000	IMGCNT_ MTX_BG = 256	1.772	IMGCNT_MTX_BB = 454	0.000	IMGCNT_MTX_BR = 0
CRROUT	1.000	IMGCNT_ MTX_RG = 256	0.000	IMGCNT_MTX_RB = 0	1.402	IMGCNT_MTX_RR = 359

## (4) YCbCr to YCbCr Conversion

$$YGIN\_A = YGIN + IMGCNT\_MTX\_YG - 128$$

$$CBBIN\_A = CBBIN - 128$$

$$CRRIN\_A = CRRIN - 128$$

$$YGOUT = (IMGCNT\_MTX\_GG \times YGIN\_A + IMGCNT\_MTX\_GB \times CBBIN\_A + IMGCNT\_MTX\_GR \times CRRIN\_A) \div 256$$

$$CBBOUT = (IMGCNT\_MTX\_BG \times YGIN\_A + IMGCNT\_MTX\_BB \times CBBIN\_A + IMGCNT\_MTX\_BR \times CRRIN\_A) \div 256 + 128$$

$$CRROUT = (IMGCNT\_MTX\_RG \times YGIN\_A + IMGCNT\_MTX\_RB \times CBBIN\_A + IMGCNT\_MTX\_RR \times CRRIN\_A) \div 256 + 128$$

Table 32.26 YCbCr to GBR Conversion

Register Name	Bit Name	Initial Value	Description
IMGCNT_MTX_MODE	IMGCNT_MTX_MD [1:0]	3	Operating Mode 0: GBR → GBR 1: GBR → YCbCr 2: YCbCr → GBR 3: YCbCr → YCbCr
IMGCNT_MTX_YG_ADJ0	IMGCNT_MTX_YG [7:0]	128	Offset (DC) Adjustment of Y/G Signal Unsigned (0 (-128) to 128 (0) to 255 (+127) [LSB], 512 [LSB])
IMGCNT_MTX_CBB_ADJ0	IMGCNT_MTX_B [7:0]	128	Offset (DC) Adjustment of B Signal Unsigned (0 (-128) to 128 (0) to 255 (+127) [LSB])
IMGCNT_MTX_CRR_ADJ0	IMGCNT_MTX_R [7:0]	128	Offset (DC) Adjustment of R Signal Unsigned (0 (-128) to 128 (0) to 255 (+127) [LSB])
IMGCNT_MTX_YG_ADJ0	IMGCNT_MTX_GG [10:0]	256	Y/G Signal Gain Adjustment for Y/G Signal Output Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
IMGCNT_MTX_YG_ADJ1	IMGCNT_MTX_GB [10:0]	0	Cb/B Signal Gain Adjustment for Y/G Signal Output Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
IMGCNT_MTX_YG_ADJ1	IMGCNT_MTX_GR [10:0]	0	Cr/R Signal Gain Adjustment for Y/G Signal Output Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
IMGCNT_MTX_CBB_ADJ0	IMGCNT_MTX_BG [10:0]	0	Y/G Signal Gain Adjustment for Cb/B Signal Output Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
IMGCNT_MTX_CBB_ADJ1	IMGCNT_MTX_BB [10:0]	256	Cb/B Signal Gain Adjustment for Cb/B Signal Output Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
IMGCNT_MTX_CBB_ADJ1	IMGCNT_MTX_BR [10:0]	0	Cr/R Signal Gain Adjustment for Cb/B Signal Output Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
IMGCNT_MTX_CRR_ADJ0	IMGCNT_MTX_RG [10:0]	0	Y/G Signal Gain Adjustment for Cr/R Signal Output Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
IMGCNT_MTX_CRR_ADJ1	IMGCNT_MTX_RB [10:0]	0	Cb/B Signal Gain Adjustment for Cr/R Signal Output Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
IMGCNT_MTX_CRR_ADJ1	IMGCNT_MTX_RR [10:0]	256	Cr/R Signal Gain Adjustment for Cr/R Signal Output Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

## 32.2 Register Descriptions

Table 32.27 to Table 32.30 show register Configuration.

- Symbols used in Register Description:

Initial value: Register value after a power-on reset

—: Undefined value

R/W: Readable/writable. The written value can be read.

R/WC0: Readable/writable. Writing 0 initializes the bit. Writing 1 is ignored.

R/WC1: Readable/writable. Writing 1 initializes the bit. Writing 0 is ignored.

R: Read-only. The write value should always be 0.

—/W: Write-only. The read value is undefined.

**Table 32.27 Register Configuration of Input Controller (Channel 0)**

Name	Abbreviation	R/W	Initial Value	Address	Access Size
External input block register update control register	INP_UPDATE	R/WC1	H'0000 0000	H'FCFF 7400	32
Input select control register	INP_SEL_CNT	R/W	H'0000 0000	H'FCFF 7404	32
External input sync signal control register	INP_EXT_SYNC_CNT	R/W	H'0000 0000	H'FCFF 7408	32
Vsync signal phase adjustment register	INP_VSYNC_PH_ADJ	R/W	H'035A 01AD	H'FCFF 740C	32
Sync signal delay adjustment register	INP_DLY_ADJ	R/W	H'0000 0000	H'FCFF 7410	32

**Table 32.28 Register Configuration of Image Quality Adjustment Block (Channel 0)**

Name	Abbreviation	R/W	Initial Value	Address	Access Size
Image quality adjustment block register update control register	IMGCNT_UPDATE	R/WC1	H'0000 0000	H'FCFF 7480	32
NR control register 0	IMGCNT_NR_CNT0	R/W	H'0010 0803	H'FCFF 7484	32
NR control register 1	IMGCNT_NR_CNT1	R/W	H'0803 0803	H'FCFF 7488	32
Image quality adjustment block matrix mode register	IMGCNT_MTX_MODE	R/W	H'0000 0003	H'FCFF 74A0	32
Image quality adjustment block matrix YG adjustment register 0	IMGCNT_MTX_YG_ADJ0	R/W	H'0080 0100	H'FCFF 74A4	32
Image quality adjustment block matrix YG adjustment register 1	IMGCNT_MTX_YG_ADJ1	R/W	H'0000 0000	H'FCFF 74A8	32
Image quality adjustment block matrix CBB adjustment register 0	IMGCNT_MTX_CBB_ADJ0	R/W	H'0080 0000	H'FCFF 74AC	32
Image quality adjustment block matrix CBB adjustment register 1	IMGCNT_MTX_CBB_ADJ1	R/W	H'0100 0000	H'FCFF 74B0	32
Image quality adjustment block matrix CRR adjustment register 0	IMGCNT_MTX_CRR_ADJ0	R/W	H'0080 0000	H'FCFF 74B4	32
Image quality adjustment block matrix CRR adjustment register 1	IMGCNT_MTX_CRR_ADJ1	R/W	H'0000 0100	H'FCFF 74B8	32
Dynamic range compression register	IMGCNT_DRC_REG	R/W	H'0000 0000	H'FCFF 74C0	32

**Table 32.29 Register Configuration of Input Controller (Channel 1)**

Name	Abbreviation	R/W	Initial Value	Address	Access Size
External input block register update control register	INP_UPDATE	R/WC1	H'0000 0000	H'FCFF 9400	32
Input select control register	INP_SEL_CNT	R/W	H'0000 0000	H'FCFF 9404	32
External input sync signal control register	INP_EXT_SYNC_CNT	R/W	H'0000 0000	H'FCFF 9408	32
Vsync signal phase adjustment register	INP_VSYNC_PH_ADJ	R/W	H'035A 01AD	H'FCFF 940C	32
Sync signal delay adjustment register	INP_DLY_ADJ	R/W	H'0000 0000	H'FCFF 9410	32

**Table 32.30 Register Configuration of Image Quality Adjustment Block (Channel 1)**

Name	Abbreviation	R/W	Initial Value	Address	Access Size
Image quality adjustment block register update control register	IMGCNT_UPDATE	R/WC1	H'0000 0000	H'FCFF 9480	32
NR control register 0	IMGCNT_NR_CNT0	R/W	H'0010 0803	H'FCFF 9484	32
NR control register 1	IMGCNT_NR_CNT1	R/W	H'0803 0803	H'FCFF 9488	32
Image quality adjustment block matrix mode register	IMGCNT_MTX_MODE	R/W	H'0000 0003	H'FCFF 94A0	32
Image quality adjustment block matrix YG adjustment register 0	IMGCNT_MTX_YG_ADJ0	R/W	H'0080 0100	H'FCFF 94A4	32
Image quality adjustment block matrix YG adjustment register 1	IMGCNT_MTX_YG_ADJ1	R/W	H'0000 0000	H'FCFF 94A8	32
Image quality adjustment block matrix CBB adjustment register 0	IMGCNT_MTX_CBB_ADJ0	R/W	H'0080 0000	H'FCFF 94AC	32
Image quality adjustment block matrix CBB adjustment register 1	IMGCNT_MTX_CBB_ADJ1	R/W	H'0100 0000	H'FCFF 94B0	32
Image quality adjustment block matrix CRR adjustment register 0	IMGCNT_MTX_CRR_ADJ0	R/W	H'0080 0000	H'FCFF 94B4	32
Image quality adjustment block matrix CRR adjustment register 1	IMGCNT_MTX_CRR_ADJ1	R/W	H'0000 0100	H'FCFF 94B8	32
Dynamic range compression register	IMGCNT_DRC_REG	R/W	H'0000 0000	H'FCFF 94C0	32

## 32.2.1 External Input Block Register Update Control Register (INP\_UPDATE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	INP_EXT_UPDATE	—	—	—	INP_IMG_UPDATE
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/WC1	R	R	R	R/WC1

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	INP_EXT_UPDATE	0	R/WC1	External Input Block Register Update 0: Registers are not updated. 1: Registers are updated.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	INP_IMG_UPDATE	0	R/WC1	Sync Signal Adjustment Block Register Update 0: Registers are not updated. 1: Registers are updated.

## 32.2.2 Input Select Control Register (INP\_SEL\_CNT)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	INP_SEL	—	—	—	—
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	INP_FORMAT[2:0]	—	—	—	—	INP_PXD_EDGE	—	—	—	INP_VS_EDGE	—	—	—	—	INP_HS_EDGE
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	INP_SEL	0	R/W	Input Select 0: Video decoder output signals 1: Signals supplied via the external input pins
19 to 15	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
14 to 12	INP_FORMAT[2:0]	0	R/W	External Input Format Select 0: YcbCr444, RGB888 1: RGB666 2: RGB565 3: BT656 4: BT601 5: YCbCr422 6, 7: Setting prohibited
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	INP_PXD_EDGE	0	R/W	Clock Edge Select for Capturing External Input Video Image Signals DV_DATA23 to DV_DATA0 0: Rising edge 1: Falling edge
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	INP_VS_EDGE	0	R/W	Clock Edge Select for Capturing External Input Vsync Signals DV_VSYNC 0: Rising edge 1: Falling edge
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	INP_HS_EDGE	0	R/W	Clock Edge Select for Capturing External Input Hsync Signals DV_HSYNC 0: Rising edge 1: Falling edge

Note: INP\_FORMAT, INP\_PXD\_EDGE, INP\_VS\_EDGE, and INP\_HS\_EDGE are updated when the INP\_EXT\_UPDATE bit in INP\_UPDATE is 1. INP\_SEL is updated when set.

## 32.2.3 External Input Sync Signal Control Register (INP\_EXT\_SYNC\_CNT)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	INP_ENDIAN_ON	—	—	—	INP_SWAP_ON	—	—	—	INP_VS_INV	—	—	—	INP_HS_INV
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	INP_H_EDGE_SEL	—	—	—	INP_F525_625	—	—	INP_H_POS[1:0]	
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	INP_ENDIAN_ON	0	R/W	External Input Bit Endian Change On/Off Control 0: Off 1: On
27 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	INP_SWAP_ON	0	R/W	External Input B/R Signal Swap On/Off Control 0: Off 1: On
23 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	INP_VS_INV	0	R/W	External Input Vsync Signal DV_VSYNC Inversion Control 0: Not inverted (positive polarity) 1: Inverted (negative polarity)
19 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	INP_HS_INV	0	R/W	External Input Hsync Signal DV_HSYNC Inversion Control 0: Not inverted (positive polarity) 1: Inverted (negative polarity)
15 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	INP_H_EDGE_SEL	0	R/W	Reference Select for External Input BT656 Hsync Signal 0: EAV 1: SAV
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	INP_F525_625	0	R/W	Number of Lines for BT656 External Input 0: 525 lines 1: 625 lines
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	INP_H_POS[1:0]	0	R/W	Y/Cb/Y/Cr Data String Start Timing to Hsync Reference for BT656/601 or YCbCr422 External Input 0: Cb/Y/Cr/Y(BT656/601), Cb/Cr (YCbCr422) 1: Y/Cr/Y/Cb(BT656/601), setting prohibited (YCbCr422) 2: Cr/Y/Cb/Y(BT656/601), setting prohibited (YCbCr422) 3: Y/Cb/Y/Cr(BT656/601), Cr/Cb (YCbCr422)

Note: This register is updated when the INP\_EXT\_UPDATE bit in INP\_UPDATE is 1.



## 32.2.4 Vsync Signal Phase Adjustment Register (INP\_VSYNC\_PH\_ADJ)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	INP_FH50[9:0]									
Initial Value:	0	0	0	0	0	0	1	1	0	1	0	1	1	0	1	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	INP_FH25[9:0]									
Initial Value:	0	0	0	0	0	0	0	1	1	0	1	0	1	1	0	1
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25 to 16	INP_FH50 [9:0]	858	R/W	Vsync Signal 1/2fH Phase Timing 1/2 clock cycle of the horizontal cycle should be set.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	INP_FH25 [9:0]	429	R/W	Vsync Signal 1/4fH Phase Timing 1/4 clock cycle of the horizontal cycle should be set.

Note: The INP\_FH50[9:0] bits are updated when the INP\_EXT\_UPDATE and INP\_IMG\_UPDATE bits in INP\_UPDATE are 1. The INP\_FH25[9:0] bits are updated when the INP\_IMG\_UPDATE bit is 1.

## 32.2.5 Sync Signal Delay Adjustment Register (INP\_DLY\_ADJ)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	INP_VS_DLY_L[2:0]	INP_FLD_DLY[7:0]									
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	INP_VS_DLY[7:0]							INP_HS_DLY[7:0]								
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 24	INP_VS_ DLY_L[2:0]	0	R/W	Number of lines for Delaying Vsync signal and Field Differentiation Signal Delay amount: 0 to 7 (lines)
23 to 16	INP_FLD_ DLY[7:0]	0	R/W	Field Differentiation Signal Delay Amount Delay amount: 0 to 254 (clock cycles)
15 to 8	INP_VS_ DLY[7:0]	0	R/W	Vsync Signal Delay Amount Delay amount: 0 to 254 (clock cycles)
7 to 0	INP_HS_ DLY[7:0]	0	R/W	Hsync Signal Delay Amount Delay amount: 0 to 254 (clock cycles)

Note: This register is updated when the INP\_IMG\_UPDATE bit in INP\_UPDATE is 1.

### 32.2.6 Image Quality Adjustment Block Register Update Control Register (IMGCNT\_UPDATE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	IMGCNT_UPDATE
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/WC1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	IMGCNT_UPDATE	0	R/WC1	Image Quality Adjustment Block Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync signal.

## 32.2.7 NR Control Register 0 (IMGCNT\_NR\_CNT0)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	NR1D_MD	—	—	—	NR1D_ON
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	NR1D_Y_TH[6:0]						—	—	NR1D_Y_TAP[1:0]		—	—	NR1D_Y_GAIN[1:0]		
Initial Value:	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	NR1D_MD	1	R/W	Horizontal Noise Reduction Operating Mode 0: G/B/R mode 1: Y/Cb/Cr mode
19 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	NR1D_ON	0	R/W	Noise Reduction On/Off Control 0: Noise reduction Off 1: Noise reduction On
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14 to 8	NR1D_Y_TH[6:0]	8	R/W	Maximum Value (Absolute Value) of Y/G Signal Coring Coring is implemented when detected noise amount value ≤ NR1D_Y_TH. Unsigned: 0 to 127 [LSB]
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5, 4	NR1D_Y_TAP[1:0]	0	R/W	Y/G Signal TAP Select 0: Adjacent pixel 1: 2 adjacent pixels 2: 3 adjacent pixels 3: 4 adjacent pixels
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	NR1D_Y_GAIN[1:0]	3	R/W	Noise Reduction Gain Adjustment of Y/G Signal 0: 1/2 1: 1/4 2: 1/8 3: 1/16

Note: This register is updated when the IMGCNT\_VEN bit in IMGCNT\_UPDATE is 1.

## 32.2.8 NR Control Register 1 (IMGCNT\_NR\_CNT1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	NR1D_CB_TH[6:0]						—	—	NR1D_CB_TAP[1:0]	—	—	NR1D_CB_GAIN[1:0]			
Initial Value:	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	NR1D_CR_TH[6:0]						—	—	NR1D_CR_TAP[1:0]	—	—	NR1D_CR_GAIN[1:0]			
Initial Value:	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30 to 24	NR1D_CB_TH[6:0]	8	R/W	Maximum Value (Absolute Value) of Cb/B Signal Coring Coring is implemented when detected noise amount value ≤ NR1D_CB_TH. Unsigned: 0 to 127 [LSB]
23, 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
21, 20	NR1D_CB_TAP[1:0]	0	R/W	Cb/B Signal TAP Select 0: Adjacent pixel 1: 2 adjacent pixels 2: 3 adjacent pixels 3: 4 adjacent pixels
19, 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17, 16	NR1D_CB_GAIN[1:0]	3	R/W	Noise Reduction Gain Adjustment of Cb/B Signal 0: 1/2 1: 1/4 2: 1/8 3: 1/16
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14 to 8	NR1D_CR_TH[6:0]	8	R/W	Maximum Value (Absolute Value) of Cr/R Signal Coring Coring is implemented when detected noise amount value ≤ NR1D_CR_TH. Unsigned: 0 to 127 [LSB]
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5, 4	NR1D_CR_TAP[1:0]	0	R/W	Cr/R Signal TAP Select 0: Adjacent pixel 1: 2 adjacent pixels 2: 3 adjacent pixels 3: 4 adjacent pixels
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	NR1D_CR_GAIN[1:0]	3	R/W	Noise Reduction Gain Adjustment of Cr/R Signal 0: 1/2 1: 1/4 2: 1/8 3: 1/16

Note: This register is updated when the IMGCNT\_VEN bit in IMGCNT\_UPDATE is 1.

### 32.2.9 Image Quality Adjustment Block Matrix Mode Register (IMGCNT\_MTX\_MODE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	IMGCNT_MTX_MD[1:0]
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	IMGCNT_MTX_MD [1:0]	3	R/W	Operating Mode 0: GBR → GBR 1: GBR → YCbCr 2: YCbCr → GBR 3: YCbCr → YCbCr

Note: This register is updated when the IMGCNT\_VEN bit in IMGCNT\_UPDATE is 1.

### 32.2.10 Image Quality Adjustment Block Matrix YG Adjustment Register 0 (IMGCNT\_MTX\_YG\_ADJ0)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	IMGCNT_MTX_YG[7:0]							
Initial Value:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	IMGCNT_MTX_GG[10:0]										
Initial Value:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	IMGCNT_MTX_YG[7:0]	128	R/W	Offset (DC) Adjustment of Y/G Signal Unsigned (0 (-128) to 128 (0) to 255 (+127) [LSB])
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	IMGCNT_MTX_GG[10:0]	256	R/W	Y/G Signal Gain Adjustment for Y/G Signal Output Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

Note: This register is updated when the IMGCNT\_VEN bit in IMGCNT\_UPDATE is 1.

### 32.2.11 Image Quality Adjustment Block Matrix YG Adjustment Register 1 (IMGCNT\_MTX\_YG\_ADJ1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	IMGCNT_MTX_GB[10:0]										
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	IMGCNT_MTX_GR[10:0]										
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	IMGCNT_MTX_GB [10:0]	0	R/W	Cb/B Signal Gain Adjustment for Y/G Signal Output Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	IMGCNT_MTX_GR [10:0]	0	R/W	Cr/R Signal Gain Adjustment for Y/G Signal Output Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

Note: This register is updated when the IMGCNT\_VEN bit in IMGCNT\_UPDATE is 1.

### 32.2.12 Image Quality Adjustment Block Matrix CBB Adjustment Register 0 (IMGCNT\_MTX\_CBB\_ADJ0)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	IMGCNT_MTX_B[7:0]							
Initial Value:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	IMGCNT_MTX_BG[10:0]										
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	IMGCNT_MTX_B[7:0]	128	R/W	Offset (DC) Adjustment of Cb/B Signal Unsigned (0 (-128) to 128 (0) to 255 (+127) [LSB])
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	IMGCNT_MTX_BG[10:0]	0	R/W	Y/G Signal Gain Adjustment for Cb/B Signal Output Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

Note: This register is updated when the IMGCNT\_VEN bit in IMGCNT\_UPDATE is 1.

### 32.2.13 Image Quality Adjustment Block Matrix CBB Adjustment Register 1 (IMGCNT\_MTX\_CBB\_ADJ1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	IMGCNT_MTX_BB[10:0]										
Initial Value:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	IMGCNT_MTX_BR[10:0]										
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	IMGCNT_MTX_BB [10:0]	256	R/W	Cb/B Signal Gain Adjustment for Cb/B Signal Output Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	IMGCNT_MTX_BR [10:0]	0	R/W	Cr/R Signal Gain Adjustment for Cb/B Signal Output Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

Note: This register is updated when the IMGCNT\_VEN bit in IMGCNT\_UPDATE is 1.



### 32.2.14 Image Quality Adjustment Block Matrix CRR Adjustment Register 0 (IMGCNT\_MTX\_CRR\_ADJ0)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	IMGCNT_MTX_R[7:0]							
Initial Value:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	IMGCNT_MTX_RG[10:0]										
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	IMGCNT_MTX_R[7:0]	128	R/W	Offset (DC) Adjustment of Cr/R Signal Unsigned (0 (-128) to 128 (0) to 255 (+127) [LSB])
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	IMGCNT_MTX_RG[10:0]	0	R/W	Y/G Signal Gain Adjustment for Cr/R Signal Output Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

Note: This register is updated when the IMGCNT\_VEN bit in IMGCNT\_UPDATE is 1.

### 32.2.15 Image Quality Adjustment Block Matrix CRR Adjustment Register 1 (IMGCNT\_MTX\_CRR\_ADJ1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	IMGCNT_MTX_RB[10:0]										
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	IMGCNT_MTX_RR[10:0]										
Initial Value:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	IMGCNT_MTX_RB[10:0]	0	R/W	Cb/B Signal Gain Adjustment for Cr/R Signal Output Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	IMGCNT_MTX_RR[10:0]	256	R/W	Cr/R Signal Gain Adjustment for Cr/R Signal Output Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

Note: This register is updated when the IMGCNT\_VEN bit in IMGCNT\_UPDATE is 1.

### 32.2.16 Dynamic Range Compression Register (IMGCNT\_DRC\_REG)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DRC_EN
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	DRC_EN	0	R/W	Use of Contrast Correction (Dynamic Range Compression) 0: Does not use contrast correction 1: Uses contrast correction

## 32.3 Usage Methods

### 32.3.1 Input Format Adjustment Method

Setting examples of each input format are shown below.

**Table 32.31 Video Decoder (NTSC) Input Setting Example**

Register Name	Bit Name	Description	Setting Value
INP_SEL_CNT	INP_SEL	Selects the input signal.	0
INP_SEL_CNT	INP_FORMAT[2:0]	Selects the externally input format.	Control not necessary
INP_SEL_CNT	INP_PXD_EDGE	Selects the clock edge for capturing the externally input video signals.	Control not necessary
INP_SEL_CNT	INP_VS_EDGE	Selects the clock edge for capturing the externally input Vsync signals.	Control not necessary
INP_SEL_CNT	INP_HS_EDGE	Selects the clock edge for capturing the externally input Hsync signals.	Control not necessary
INP_EXT_SYNC_CNT	INP_ENDIAN_ON	Changes the bit endian of the external input.	Control not necessary
INP_EXT_SYNC_CNT	INP_SWAP_ON	Enables or disables the B/R signal swap of the external input.	Control not necessary
INP_EXT_SYNC_CNT	INP_HS_INV	Enables or disables the Hsync signal inversion of the external input.	Control not necessary
INP_EXT_SYNC_CNT	INP_VS_INV	Enables or disables the Hsync signal inversion of the external input.	Control not necessary
INP_EXT_SYNC_CNT	INP_H_EDGE_SEL	Selects the Hsync reference for BT656 input.	Control not necessary
INP_EXT_SYNC_CNT	INP_F525_625	Sets the number of lines for BT656 input.	Control not necessary
INP_EXT_SYNC_CNT	INP_H_POS[1:0]	Sets the data start timing with respect to the Hsync in the BT656/601 format.	Control not necessary
INP_VSYNC_PH_ADJ	INP_FH50[9:0]	Sets the 1/2fH phase in clock cycle units.	858
INP_VSYNC_PH_ADJ	INP_FH25[9:0]	Sets the 1/4fH phase in clock cycle units.	429
INP_DLY_ADJ	INP_VS_DLY_L[2:0]	Sets the number of lines for delaying the Vsync signal and field differentiation signal.	0
INP_DLY_ADJ	INP_VS_DLY[7:0]	Sets the amount of delay of the Vsync signal in clock cycle units.	0
INP_DLY_ADJ	INP_HS_DLY[7:0]	Sets the amount of delay of the Hsync signal in clock cycle units.	0
INP_DLY_ADJ	INP_FLD_DLY[7:0]	Sets the amount of delay of the field differentiation signal in clock units.	0

Note: Some registers require, after they are set, that the INP\_EXT\_UPDATE and INP\_IMG\_UPDATE bits in INP\_UPDATE should be set to 1.

**Table 32.32 External Input (BT656, 525i) Setting Example**

Register Name	Bit Name	Description	Setting Value
INP_SEL_CNT	INP_SEL	Selects the input signal.	1
INP_SEL_CNT	INP_FORMAT[2:0]	Selects the externally input format.	3
INP_SEL_CNT	INP_PXD_EDGE	Selects the clock edge for capturing the externally input video signals.	0
INP_SEL_CNT	INP_VS_EDGE	Selects the clock edge for capturing the externally input Vsync signals.	0
INP_SEL_CNT	INP_HS_EDGE	Selects the clock edge for capturing the externally input Hsync signals.	0
INP_EXT_SYNC_CNT	INP_ENDIAN_ON	Changes the bit endian of the external input.	0
INP_EXT_SYNC_CNT	INP_SWAP_ON	Enables or disables the B/R signal swap of the external input.	0
INP_EXT_SYNC_CNT	INP_HS_INV	Enables or disables the Hsync signal inversion of the external input.	1
INP_EXT_SYNC_CNT	INP_VS_INV	Enables or disables the Hsync signal inversion of the external input.	1
INP_EXT_SYNC_CNT	INP_H_EDGE_SEL	Selects the Hsync reference for BT656 input.	0
INP_EXT_SYNC_CNT	INP_F525_625	Sets the number of lines for BT656 input.	0
INP_EXT_SYNC_CNT	INP_H_POS[1:0]	Sets the data start timing with respect to the Hsync in the BT656/601 format.	0
INP_VSYNC_PH_ADJ	INP_FH50[9:0]	Sets the 1/2fH phase in clock cycle units.	858
INP_VSYNC_PH_ADJ	INP_FH25[9:0]	Sets the 1/4fH phase in clock cycle units.	429
INP_DLY_ADJ	INP_VS_DLY_L[2:0]	Sets the number of lines for delaying the Vsync signal and field differentiation signal.	0
INP_DLY_ADJ	INP_VS_DLY[7:0]	Sets the amount of delay of the Vsync signal in clock cycle units.	0
INP_DLY_ADJ	INP_HS_DLY[7:0]	Sets the amount of delay of the Hsync signal in clock cycle units.	0
INP_DLY_ADJ	INP_FLD_DLY[7:0]	Sets the amount of delay of the field differentiation signal in clock units.	0

Note: Some registers require, after they are set, that the INP\_EXT\_UPDATE and INP\_IMG\_UPDATE bits in IMP\_UPDATE should be set to 1.

**Table 32.33 External Input (BT601, 525i) Setting Example**

Register Name	Bit Name	Description	Setting Value
INP_SEL_CNT	INP_SEL	Selects the input signal.	1
INP_SEL_CNT	INP_FORMAT[2:0]	Selects the externally input format.	4
INP_SEL_CNT	INP_PXD_EDGE	Selects the clock edge for capturing the externally input video signals.	0
INP_SEL_CNT	INP_VS_EDGE	Selects the clock edge for capturing the externally input Vsync signals.	0
INP_SEL_CNT	INP_HS_EDGE	Selects the clock edge for capturing the externally input Hsync signals.	0
INP_EXT_SYNC_CNT	INP_ENDIAN_ON	Changes the bit endian of the external input.	0
INP_EXT_SYNC_CNT	INP_SWAP_ON	Enables or disables the B/R signal swap of the external input.	0
INP_EXT_SYNC_CNT	INP_HS_INV	Enables or disables the Hsync signal inversion of the external input.	1
INP_EXT_SYNC_CNT	INP_VS_INV	Enables or disables the Hsync signal inversion of the external input.	1
INP_EXT_SYNC_CNT	INP_H_EDGE_SEL	Selects the Hsync reference for BT656 input.	0
INP_EXT_SYNC_CNT	INP_F525_625	Sets the number of lines for BT656 input.	0
INP_EXT_SYNC_CNT	INP_H_POS[1:0]	Sets the data start timing with respect to the Hsync in the BT656/601 format.	0
INP_VSYNC_PH_ADJ	INP_FH50[9:0]	Sets the 1/2fH phase in clock cycle units.	858
INP_VSYNC_PH_ADJ	INP_FH25[9:0]	Sets the 1/4fH phase in clock cycle units.	429
INP_DLY_ADJ	INP_VS_DLY_L[2:0]	Sets the number of lines for delaying the Vsync signal and field differentiation signal.	0
INP_DLY_ADJ	INP_VS_DLY[7:0]	Sets the amount of delay of the Vsync signal in clock cycle units.	0
INP_DLY_ADJ	INP_HS_DLY[7:0]	Sets the amount of delay of the Hsync signal in clock cycle units.	0
INP_DLY_ADJ	INP_FLD_DLY[7:0]	Sets the amount of delay of the field differentiation signal in clock units.	0

Note: Some registers require, after they are set, that the INP\_EXT\_UPDATE and INP\_IMG\_UPDATE bits in INT\_UPDATE should be set to 1.

### 32.3.2 Usage Method of Conversion Color Matrix

Typical data conversion setting examples are shown below.

**Table 32.34 Conversion Color Matrix**

Register Name	Bit Name	GBR to GBR	GBR to YCbCr	YCbCr to GBR	YCbCr to YCbCr
IMGCNT_MTX_MODE	IMGCNT_MTX_MD[1:0]	0	1	2	3
IMGCNT_MTX_YG_ADJ0	IMGCNT_MTX_YG[7:0]	128	128	128	128
IMGCNT_MTX_YG_ADJ0	IMGCNT_MTX_GG[10:0]	256	150	256	256
IMGCNT_MTX_YG_ADJ1	IMGCNT_MTX_GB[10:0]	0	29	1960	0
IMGCNT_MTX_YG_ADJ1	IMGCNT_MTX_GR[10:0]	0	77	1865	0
IMGCNT_MTX_CBB_ADJ0	IMGCNT_MTX_B[7:0]	128	128	128	128
IMGCNT_MTX_CBB_ADJ0	IMGCNT_MTX_BG[10:0]	0	1963	256	0
IMGCNT_MTX_CBB_ADJ1	IMGCNT_MTX_BB[10:0]	256	128	454	256
IMGCNT_MTX_CBB_ADJ1	IMGCNT_MTX_BR[10:0]	0	2005	0	0
IMGCNT_MTX_CRR_ADJ0	IMGCNT_MTX_R[7:0]	128	128	128	128
IMGCNT_MTX_CRR_ADJ0	IMGCNT_MTX_RG[10:0]	0	1941	256	0
IMGCNT_MTX_CRR_ADJ1	IMGCNT_MTX_RB[10:0]	0	2027	0	0
IMGCNT_MTX_CRR_ADJ1	IMGCNT_MTX_RR[10:0]	256	128	359	256

Note: The registers require, after they are set, that the IMGCNT\_VEN bit in IMGCNT\_UPDATE should be set to 1.

### 33. Video Display Controller 5 (3): Scaler

#### 33.1 Scaler

##### 33.1.1 Overview of Functions

The scaler subjects the YCbCr and RGB signals output from the input controller, to sync signal generation; and reduction, enlargement, and rotation of the images.

The scaler also records video image in the frame buffer.

Video display controller 5 has two scalers (scalers 0 and 1) in each channel. For image blending, either cascading the scalers or using the VIN synthesizer can be selected. When the scalers are cascaded for image blending, the image from scaler 0 is placed in the lower layer and that from scaler 1 is placed in the upper layer.

Scaler 0 receives a signal from the input controller in the channel where scaler 0 belongs, and scaler 1 receives a signal from the input controller in the other channel.

For the VIN synthesizer, refer to section 35, Video Display Controller 5 (5): Image Synthesizer.

In scalers 0 and 1, either enlargement process or graphics 0 or 1 process can be used at a time.

The functional block diagrams of scalers 0 and 1 are shown below.

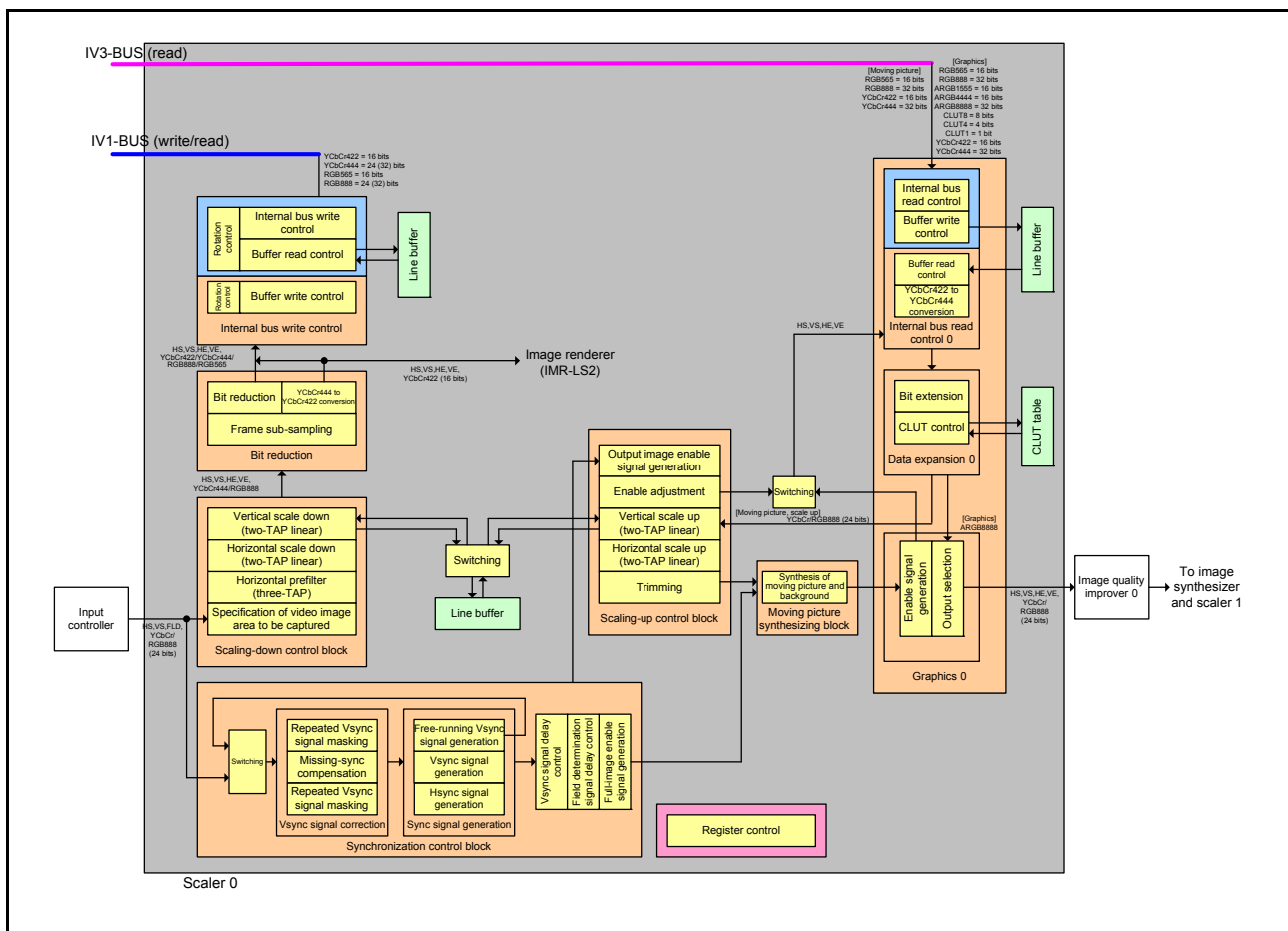


Figure 33.1 Functional Block Diagram of Scaler 0

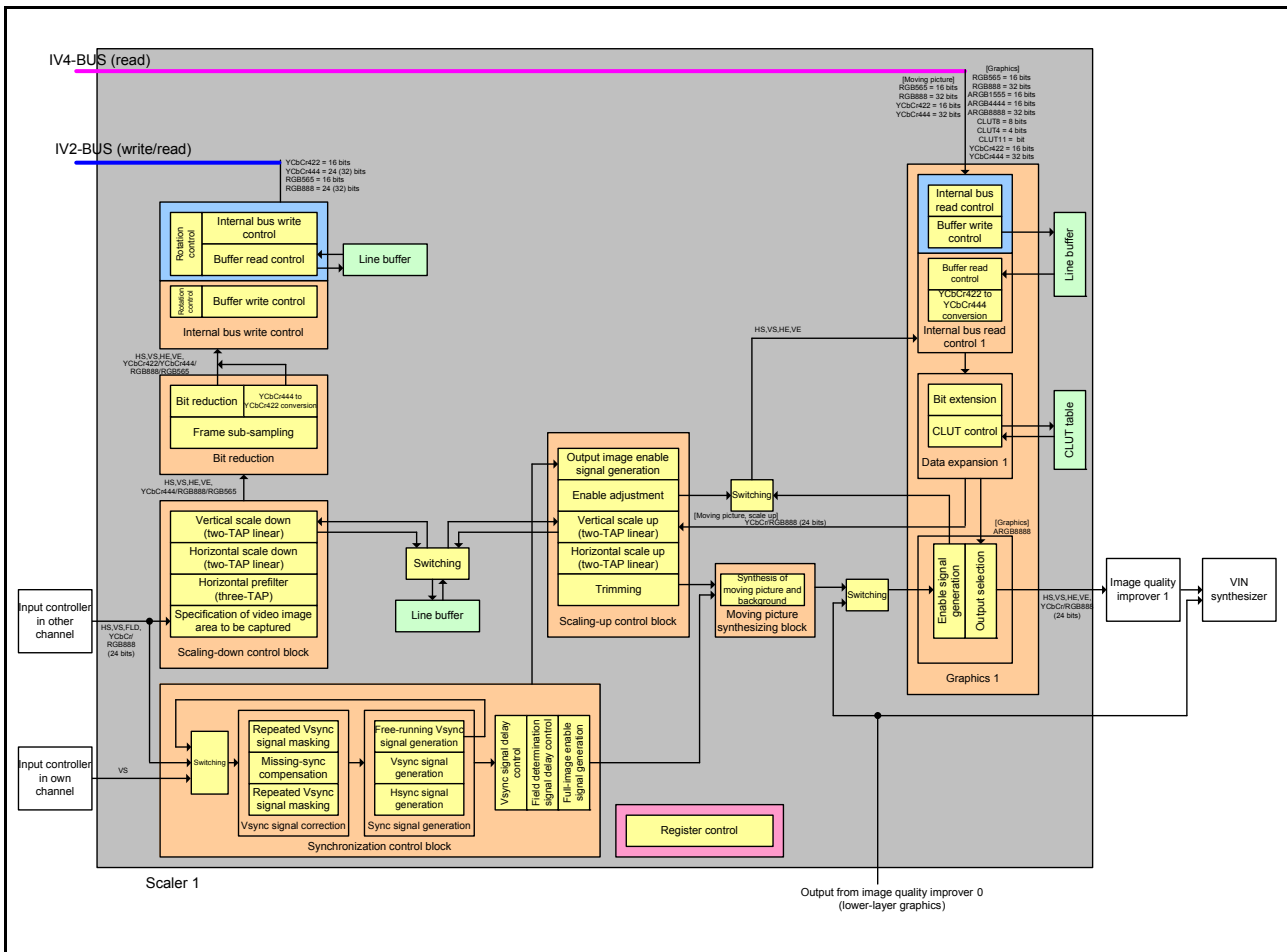


Figure 33.2 Functional Block Diagram of Scaler 1

The video display controller has two scalers and each scaler has a graphics block. The registers and bits in the scalers are named SC0\_XXXX or SC1\_XXXX and those in the graphics blocks are named GR0\_XXXX or GR1\_XXXX, but in this section, they are collectively called SC\_XXXX or GR\_XXXX.

### 33.1.2 Register Control

#### (1) Updating Registers

The Vsync signal is used to control the update timing of all the registers of the scaling and graphics blocks except some registers of the sync control block and some of the other blocks.

After 1 is set to the bits in the update control register, the contents of the relevant registers are modified at the rising edge of the Vsync signal. The update control register is automatically cleared to 0 after the modification.



**Table 33.1 Register Update Control**

Register Name	Bit Name	Initial Value	Description
SC_SCL0_UPDATE	SC_SCL0_UPDATE	0	SYNC Control Register Update 0: Registers are not updated. 1: Registers are updated.
SC_SCL0_UPDATE	SC_SCL0_VEN_D	0	Scaling-Up Control and Frame Buffer Read Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
SC_SCL0_UPDATE	SC_SCL0_VEN_C	0	Scaling-Down Control and Frame Buffer Read Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
SC_SCL0_UPDATE	SC_SCL0_VEN_B	0	Synchronization Control and Scaling-up Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
SC_SCL0_UPDATE	SC_SCL0_VEN_A	0	Scaling-Down Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
SC_SCL1_UPDATE	SC_SCL1_UPDATE_B	0	Frame Buffer Write Control Register Update 0: Registers are not updated. 1: Registers are updated.
SC_SCL1_UPDATE	SC_SCL1_UPDATE_A	0	Frame Buffer Write Control Register Update 0: Registers are not updated. 1: Registers are updated.
SC_SCL1_UPDATE	SC_SCL1_VEN_B	0	Frame Buffer Write Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
SC_SCL1_UPDATE	SC_SCL1_VEN_A	0	Frame Buffer Write Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
GR_UPDATE	GR_UPDATE	0	Frame Buffer Read Control Register Update 0: Registers are not updated. 1: Registers are updated.
GR_UPDATE	GR_P_VEN	0	Graphics Display Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
GR_UPDATE	GR_IBUS_VEN	0	Frame Buffer Read Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.

The registers controlled by SC\_SCL0\_VEN\_A, SC\_SCL0\_VEN\_C, SC\_SCL1\_VEN\_A, and SC\_SCL1\_VEN\_B are modified at the rising edge of the input Vsync signal.

The registers controlled by SC\_SCL0\_VEN\_B, SC\_SCL0\_VEN\_D, GR\_P\_VEN, and GR\_IBUS\_VEN are modified at the rising edge of the output Vsync signal.

### 33.1.3 Synchronization Control

#### (1) Selecting Vsync Signal

The Vsync signal to be output from the scaler can be selected.

When an external input signal is to be displayed, an external input Vsync signal should be selected to be output.

When an external input signal is not provided, a free-running Vsync signal should be selected to be output.

**Table 33.2 Vsync Signal Selection Control**

Register Name	Bit Name	Initial Value	Description
SC_SCL0_FRC3	SC_RES_VS_SEL	1	Vsync Signal Output Select 0: External input Vsync signal 1: Internally generated free-running Vsync signal

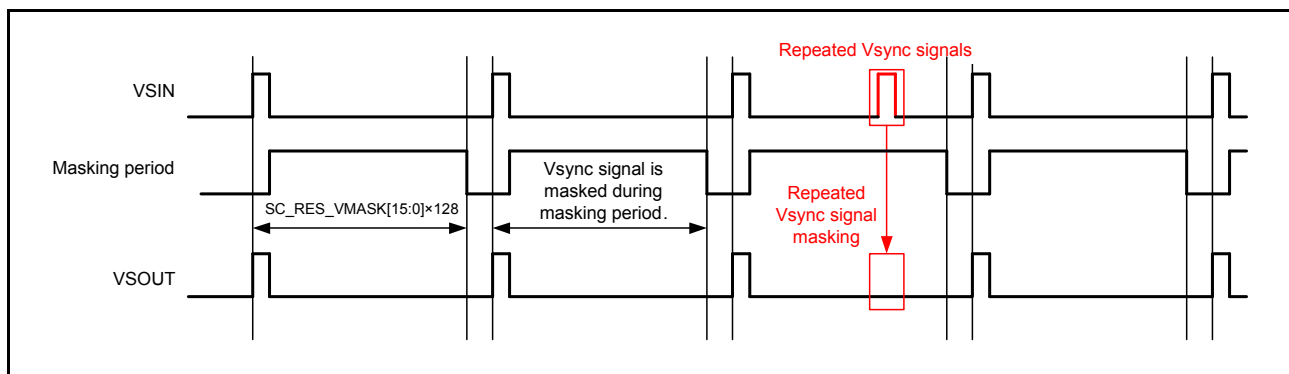
#### (2) Masking Repeated Vsync Signals

It is possible to prevent receiving the Vsync signal with a period shorter than the standard period. This is achieved by setting the start timing to receive the next Vsync signal after receiving an input Vsync signal.

The Vsync signal reception masking period is set with the SC\_RES\_VMASK[15:0] bits.

$$\text{Masking period [usec]} = \text{SC\_RES\_VMASK} \times 128 \div \text{pixel clock frequency [MHz]}$$

This function is enabled or disabled by the SC\_RES\_VMASK\_ON bit.



**Figure 33.3 Timing for Masking Repeated Vsync Signals**

**Table 33.3 Repeated Vsync Signal Mask Control**

Register Name	Bit Name	Initial Value	Description
SC_SCL0_FRC1	SC_RES_VMASK_ON	1	Repeated Vsync Signal Masking Control 0: Repeated Vsync signal masking control is disabled. 1: Repeated Vsync signal masking control is enabled.
SC_SCL0_FRC1	SC_RES_VMASK[15:0]	2800	Repeated Vsync Signal Masking Period Sets the repeated Vsync signal masking period beginning at a Vsync signal in terms of 128 pixel-clock periods. Masking period [usec] = SC_RES_VMASK × 128 ÷ pixel clock frequency [MHz]

### (3) Compensating for Missing Vsync Signals

It is possible to prevent output of the Vsync signal with a period longer than the standard period. This is achieved by setting the wait time after reception of an input Vsync signal until reception of the next Vsync signal.

If no Vsync signals are received during the wait time, an internally generated sync signal is inserted.

The wait time can be set using the SC\_RES\_VLACK[15:0] bits.

$$\text{Wait time [usec]} = \text{SC\_RES\_VLACK} \times 128 \div \text{pixel clock frequency [MHz]}$$

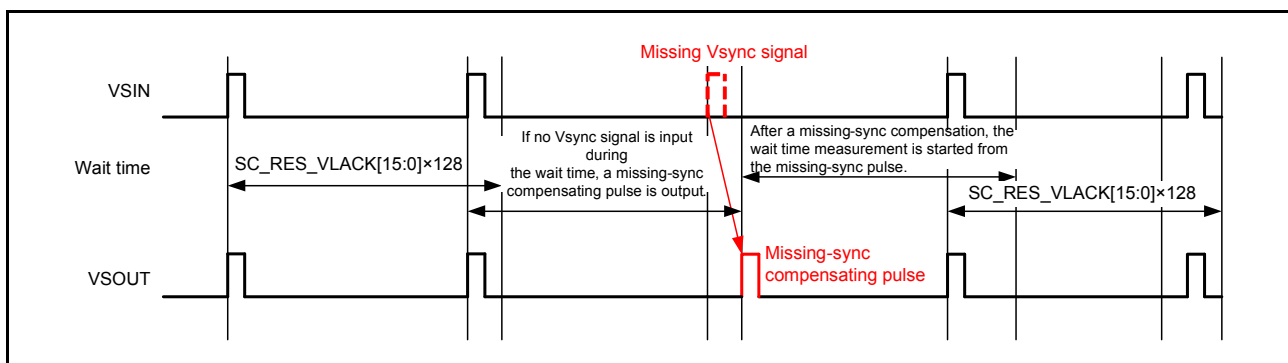
This function is enabled or disabled by the SC\_RES\_VLACK\_ON bit.

If no Vsync signals are input during the Vsync signal reception time, the SC\_RES\_QVLACK bit is set to the high level.

If Vsync signals are continuously detected four or more times during the Vsync signal reception time, the SC\_RES\_QVLOCK bit is set to the high level.

The SC\_RES\_QVLOCK bit is valid even when both the SC\_RES\_VMASK\_ON and SC\_RES\_VLACK\_ON bits are set to turn off the corresponding functions.

Note that, however, the SC\_RES\_VMASK and SC\_RES\_VLACK bits must be set correctly.



**Figure 33.4 Compensation of Missing Vsync Signals**

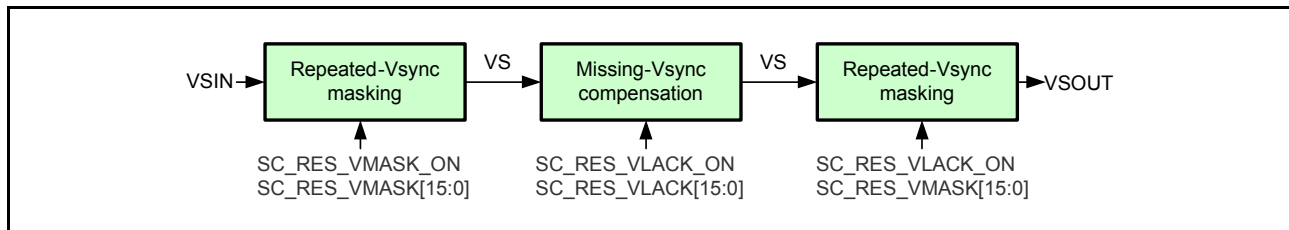
**Table 33.4 Missing Vsync Compensation Control**

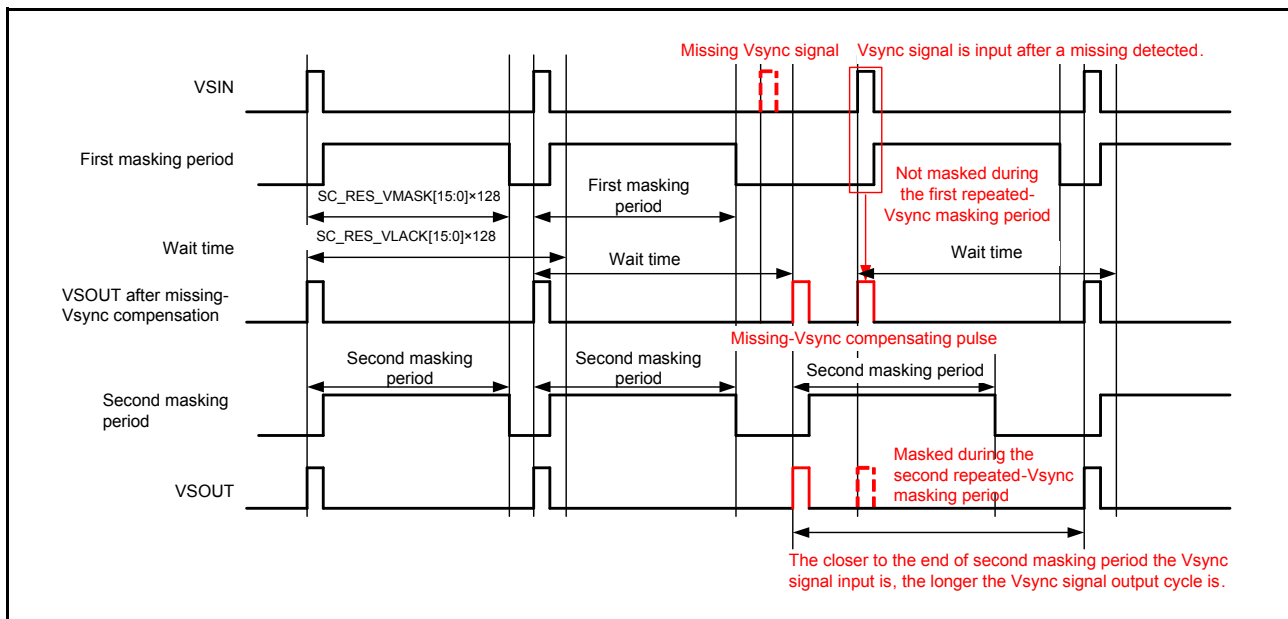
Register Name	Bit Name	Initial Value	Description
SC_SCL0_FRC2	SC_RES_VLACK_ON	1	Missing Vsync Signal Compensation 0: Compensation of missing Vsync signals is disabled. 1: Compensation of missing Vsync signals is enabled.
SC_SCL0_FRC2	SC_RES_VLACK[15:0]	3600	Missing-Sync Compensating Pulse Output Wait Time Sets the wait time before outputting a missing-sync compensating pulse after a Vsync signal. Wait time [usec] = SC_RES_VLACK × 128 ÷ pixel clock frequency [MHz]
SC_SCL0_FRC9	SC_RES_QVLACK	—	Missing Vsync Signal Detection Flag 1: Missing Vsync signal input has been detected. 0: No missing Vsync signal input has been detected.
SC_SCL0_FRC9	SC_RES_QVLOCK	—	Locked Vsync Signal Detection Flag 1: No repeated or missing Vsync signal input has been detected for four or more vertical periods. 0: Repeated or missing Vsync signal input has been detected.

For the Vsync signal, repeated-signal masking is first carried out and then missing-signal compensation is carried out, followed by another repeated-signal masking.

Repetition masking is inserted after missing-Vsync compensation to prevent output of the Vsync signal even in cases such as the input of a Vsync signal immediately after the input of a pulse to compensate for a missing Vsync signal.

On/off control of the missing-Vsync compensation also applies to the second repeated-Vsync masking; and masking period setting of the first repeated-Vsync masking also applies to the second repeated-Vsync masking.

**Figure 33.5 Repeated-Vsync Masking and Missing-Vsync Compensation**



**Figure 33.6** Timing for Masking Repeated Vsync Signals and Missing Vsync Signal Compensation

#### (4) Free-Running Period

Free-running Vsync and Hsync periods can be set.

$$\text{Hsync period [usec]} = (\text{SC\_RES\_FH} + 1) \div \text{pixel clock frequency [MHz]}$$

$$\text{Vsync period [usec]} = \text{horizontal period [usec]} \times (\text{SC\_RES\_FV} + 1)$$

**Table 33.5** Free-Running Period Control

Register Name	Bit Name	Initial Value	Description
SC_SCL0_FRC4	SC_RES_FV[10:0]	524	Free-Running Vsync Period Setting Free-running Vsync period = (SC_RES_FV + 1) × horizontal period [usec]
SC_SCL0_FRC4	SC_RES_FH[10:0]	799	Hsync Period Setting Hsync period [usec] = (SC_RES_FH + 1) ÷ pixel clock frequency [MHz]

When selecting an external input Vsync signal, set the SC\_RES\_VS\_SEL bit to 0. At this time, the internally generated free-running Vsync signal is not output.

In the meantime, the Hsync signal is always generated according to the free-running signal setting and output from the scaler.

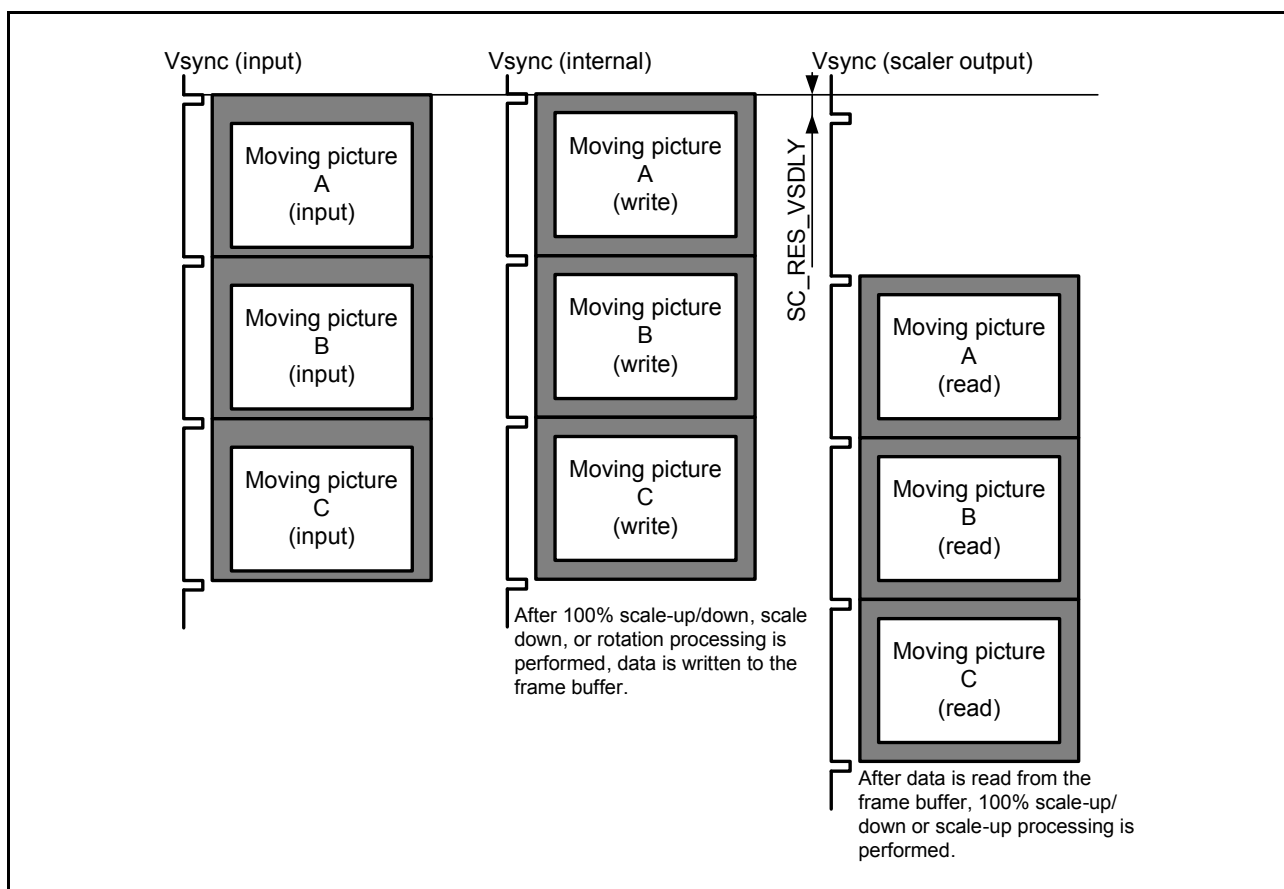
(5) Vsync Signal Delay Control

Delay of Vsync signal output from the scaler can be controlled.

The delay is used to adjust the frame buffer read timing.

**Table 33.6 Vsync Output Delay Control**

Register Name	Bit Name	Initial Value	Description
SC_SCL0_FRC5	SC_RES_VSDLY[7:0]	1	Vsync Signal Delay Control Adjusts the Vsync signal delay in the output Hsync period units. Vsync signal delay [usec]: SC_RES_VSDLY × output Hsync period [usec]



**Figure 33.7 Vsync Signal Phases (Two Frame-Buffer Planes Used)**

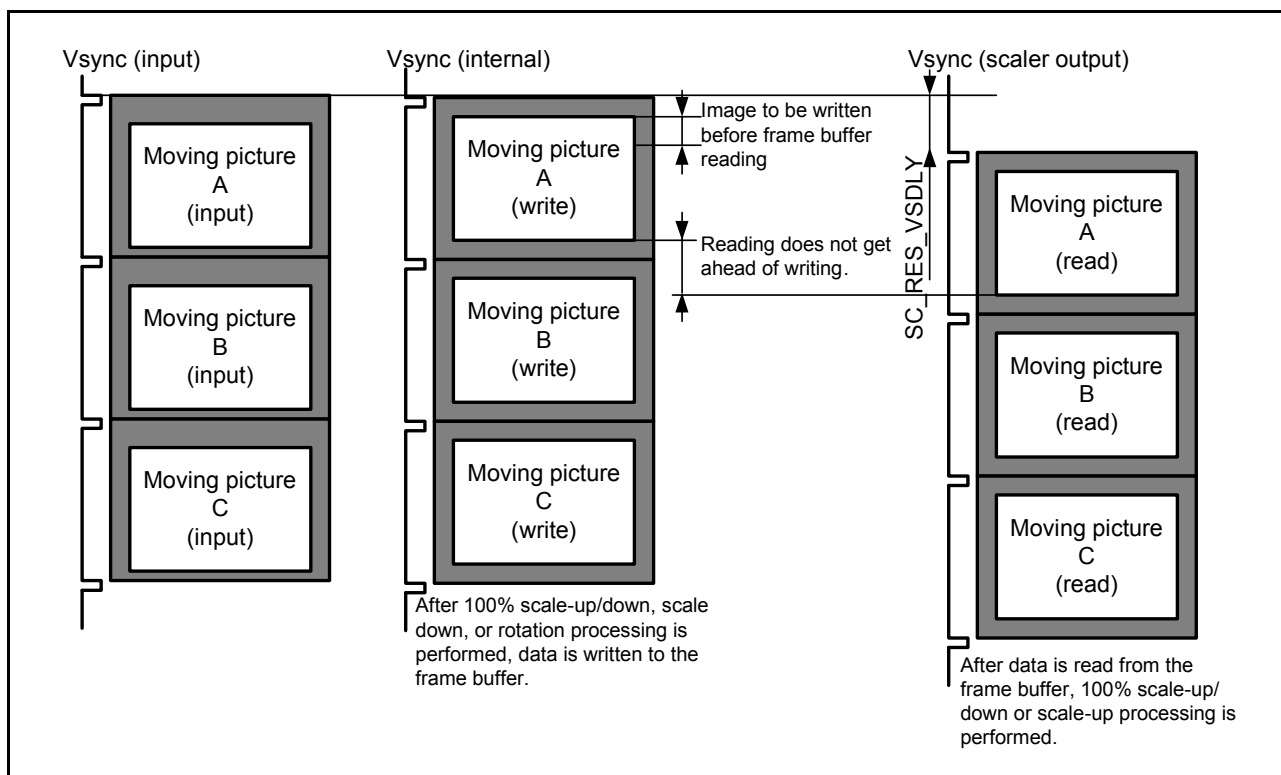


Figure 33.8 Vsync Signal Phases (One Frame-Buffer Plane Used)

### 33.1.4 Setting Angle of View

#### (1) Setting Image Area to be Captured

The image area to be captured can be set for reduction or enlargement.

The area is defined by specifying its start position and width based on the input Hsync and Vsync signals.

Table 33.7 Control of Image Area to be Captured

Register Name	Bit Name	Initial Value	Description
SC_SCL0_DS2	SC_RES_VS[10:0]	18	Vertical Position Setting for Video Signal Capturing (VSYNC + (V backporch - 1) lines) Note: The set value should be four or more (lines). SC_RES_VS + SC_RES_VW should be equal to or less than 2039 (lines).
SC_SCL0_DS2	SC_RES_VW[10:0]	240	Vertical Width of Video Signal to be Captured (lines) Note: SC_RES_VS + SC_RES_VW should be equal to or less than 2039 (lines).
SC_SCL0_DS3	SC_RES_HS[10:0]	244	Horizontal Position Setting for Video Signal Capturing (HSYNC + H backporch video-image clock cycles) Note: The set value should be 16 or more (clock cycles). SC_RES_HS + SC_RES_HW should be equal to or less than 2015 (clock cycles).
SC_SCL0_DS3	SC_RES_HW[10:0]	1440	Horizontal Width of Video Signal to be Captured (video-image clock cycles) Note: SC_RES_HS + SC_RES_HW should be equal to or less than 2015 (clock cycles).

## (2) Generating a Full-Screen Enable Signal

The valid period of the full screen to be output from the scaler can be set.

The valid period is defined by specifying its start position and width based on the Hsync and Vsync signals output from the scaler.

**Table 33.8 Full-Screen Enable Control**

Register Name	Bit Name	Initial Value	Description
SC_SCL0_FRC6	SC_RES_F_VS[10:0]	35	Vertical Enable Signal Start Position for Full Screen. (VSYNC + V backporch lines) Note: The set value should be four or more (lines). SC_RES_F_VS + SC_RES_F_VW should be equal to or less than 2039 (lines).
SC_SCL0_FRC6	SC_RES_F_VW[10:0]	480	Vertical Enable Signal Width for Full Screen (lines) Note: SC_RES_F_VS + SC_RES_F_VW should be equal to or less than 2039 (lines).
SC_SCL0_FRC7	SC_RES_F_HS[10:0]	144	Horizontal Enable Signal Start Position for Full Screen. (HSYNC + H backporch pixel-clock cycles) Note: The set value should be 16 or more (clock cycles). SC_RES_F_HS + SC_RES_F_HW should be equal to or less than 2015 (clock cycles).
SC_SCL0_FRC7	SC_RES_F_HW[10:0]	640	Horizontal Enable Signal Width for Full Screen (pixel-clock cycles) Note: SC_RES_F_HS + SC_RES_F_HW should be equal to or less than 2015 (clock cycles).



### (3) Generating an Image Output Enable Signal

The valid period of the image to be output can be set.

The valid period is defined by specifying its start position and width based on the Hsync and Vsync signals output from the scaler.

**Table 33.9 Image Output Enable Control**

Register Name	Bit Name	Initial Value	Description
SC_SCL0_US2	SC_RES_P_VS[10:0]	35	Vertical Enable Signal Start Position for Output Image. (VSYNC + V backporch lines) Note: The set value should be four or more (lines). SC_RES_P_VS + SC_RES_P_VW should be equal to or less than 2039 (lines).
SC_SCL0_US2	SC_RES_P_VW[10:0]	480	Vertical Enable Signal Width for Output Image (lines) Note: SC_RES_P_VS + SC_RES_P_VW should be equal to or less than 2039 (lines).
SC_SCL0_US3	SC_RES_P_HS[10:0]	144	Horizontal Enable Signal Start Position for Output Image. (HSYNC + H backporch pixel-clock cycles) Note: The set value should be 16 or more (clock cycles). SC_RES_P_HS + SC_RES_P_HW should be equal to or less than 2015 (clock cycles).
SC_SCL0_US3	SC_RES_P_HW[10:0]	640	Horizontal Enable Signal Width for Output Image (pixel-clock cycles) Note: SC_RES_P_HS + SC_RES_P_HW should be equal to or less than 2015 (clock cycles).

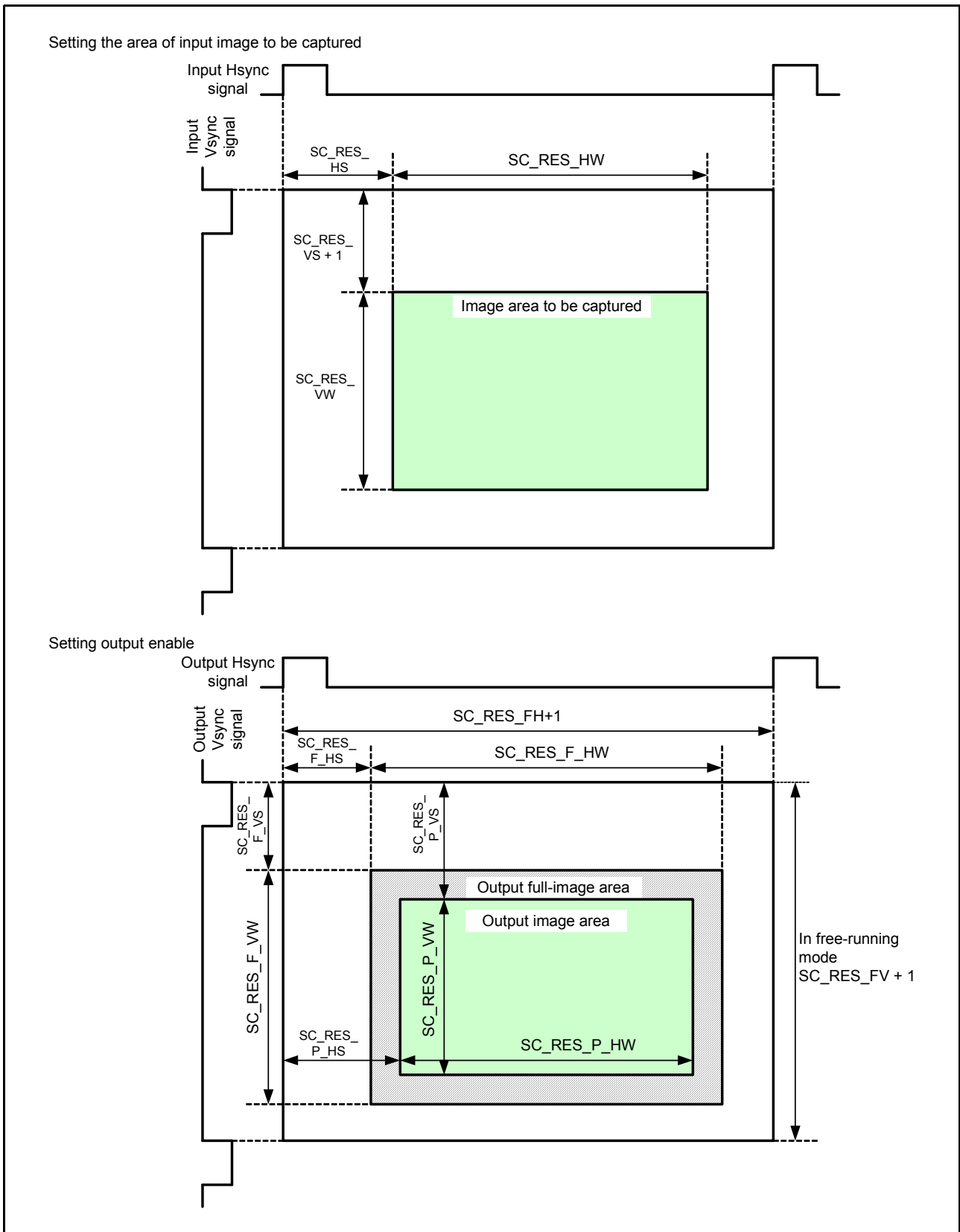


Figure 33.9 Enable Settings

### 33.1.5 Scaling Settings

#### (1) Scaling Processing Block

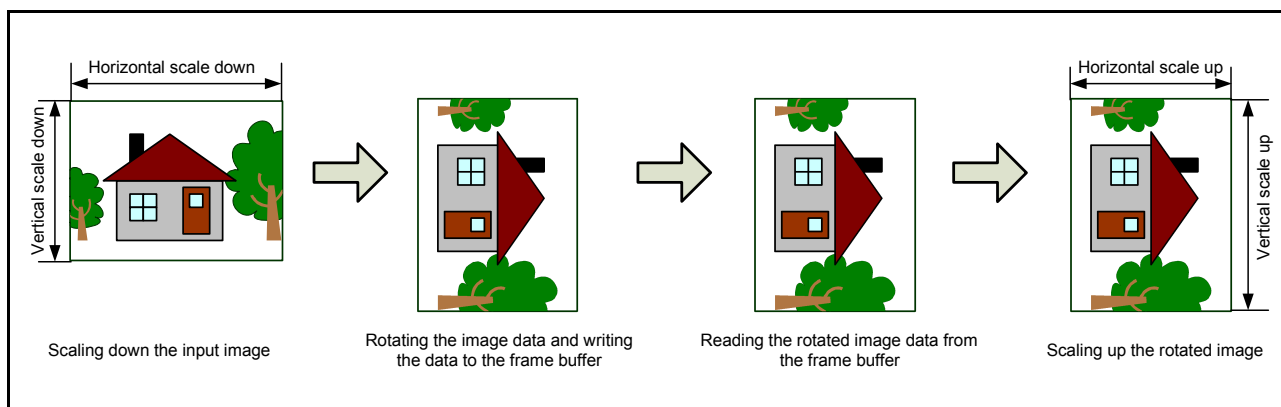
The scaling-down control block scales down the input image from the input controller.

When rotation is required, the scaling-down control block first scales down the image and then rotates it before writing it into the frame buffer.

The scaling-up control block reads the rotated image from the frame buffer and scales it up.

**Table 33.10 Rotation and Scaling Process**

Rotation	Horizontal Scaling	Vertical Scaling	Scaling-Down Control Block	Scaling-Up Control Block
Normal	Horizontal scale down	Vertical scale down	Horizontal scale down/ vertical scale down	Horizontal 100% scale up/ vertical 100% scale up
	Horizontal scale down	Vertical scale up	Horizontal scale down/ vertical 100% scale up	Horizontal 100% scale up/ vertical scale up
	Horizontal scale up	Vertical scale down	Horizontal 100% scale up/ vertical scale down	Horizontal scale up/ vertical 100% scale up
	Horizontal scale up	Vertical scale up	Horizontal 100% scale up/ vertical 100% scale up	Horizontal scale up/ vertical scale up
Horizontal mirroring	Horizontal scale down	Vertical scale down	Horizontal scale down/ vertical scale down	Horizontal 100% scale up/ vertical 100% scale up
	Horizontal scale down	Vertical scale up	Horizontal scale down/ vertical 100% scale up	Horizontal 100% scale up/ vertical scale-up
	Horizontal scale up	Vertical scale down	Horizontal 100% scale up/ vertical scale down	Horizontal scale up/ vertical 100% scale up
	Horizontal scale up	Vertical scale up	Horizontal 100% scale up/ vertical 100% scale up	Horizontal scale up/ vertical scale up
90° rotation	(Horizontal input → vertical output) scale down	(Vertical input → horizontal output) scale down	Horizontal scale down/ vertical scale down	Horizontal 100% scale up/ vertical 100% scale up
	(Horizontal input → vertical output) scale down	(Vertical input → horizontal output) scale up	Horizontal scale down/ vertical 100% scale up	Horizontal scale up/ vertical 100% scale up
	(Horizontal input → vertical output) scale up	(Vertical input → horizontal output) scale up	Horizontal 100% scale up/ vertical 100% scale up	Horizontal scale up/ vertical scale up
180° rotation	Horizontal scale down	Vertical scale down	Horizontal scale down/ vertical scale down	Horizontal 100% scale up/ vertical 100% scale up
	Horizontal scale down	Vertical scale up	Horizontal scale down/ vertical 100% scale up	Horizontal 100% scale up/ vertical scale up
	Horizontal scale up	Vertical scale down	Horizontal 100% scale up/ vertical scale down	Horizontal scale up/ vertical 100% scale up
	Horizontal scale up	Vertical scale up	Horizontal 100% scale up/ vertical 100% scale up	Horizontal scale up/ vertical scale up
270° rotation	(Horizontal input→vertical output) scale down	(Vertical input→horizontal output) scale down	Horizontal scale down/ vertical scale down	Horizontal 100% scale up/ vertical 100% scale up
	(Horizontal input → vertical output) scale down	(Vertical input → horizontal output) scale up	Horizontal scale down/ vertical 100% scale up	Horizontal scale up/ vertical 100% scale up
	(Horizontal input → vertical output) scale up	(Vertical input → horizontal output) scale up	Horizontal 100% scale up/ vertical 100% scale up	Horizontal scale up/ vertical scale up

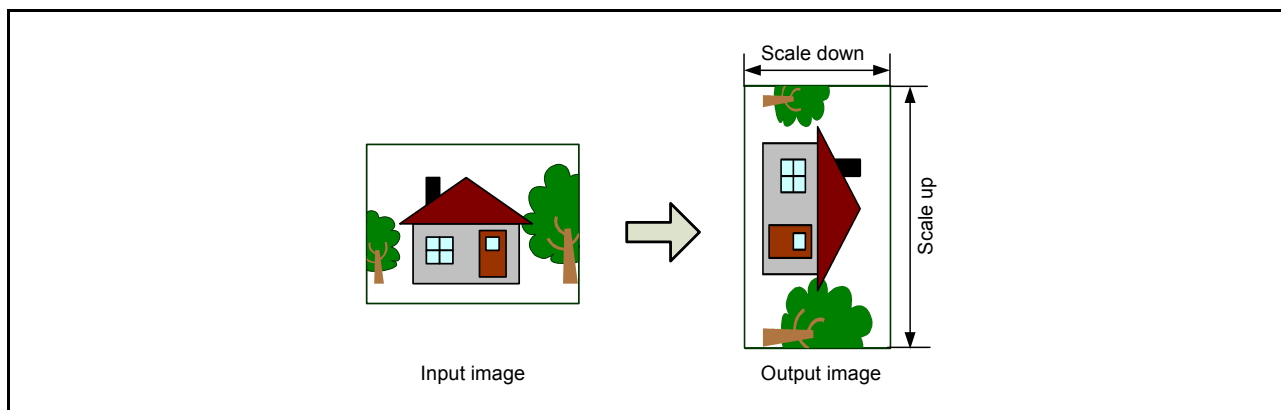


**Figure 33.10** Rotation and Scaling Process

It is impossible to use vertical reduction by the scaling-down control block and vertical enlargement by the scaling-up control block simultaneously because they are mutually exclusive. Thus, the following scaling processes cannot be performed with 90° rotation or 270° rotation.

**Table 33.11** Impossible Scaling Process

Rotation	Horizontal Scaling	Vertical Scaling	Scaling-Down Control Block	Scaling-Up Control Block
90° rotation 270° rotation	(Horizontal input → vertical output) scale up	(Vertical input → horizontal output) scale down	Horizontal 100% scale up/ vertical scale down	Horizontal 100% scale up/ vertical scale up



**Figure 33.11** Impossible Scaling Process

### 33.1.6 Horizontal Prefilter

The horizontal prefilter can be turned on or off for brightness (Y) and RGB signals to suppress the frequency band of the signals during horizontal size reduction. The input format depends on the SC\_RES\_MD[1:0] bit setting in the writing mode register (SC\_SCL1\_WR1).

When the horizontal reduction ratio is high and there is too much folding frequency component to ignore, the horizontal prefilter should be turned on.

**Table 33.12 Horizontal Prefilter Settings**

Input Format	SC_RES_PFIL_SEL	Operation
YCbCr input	1	Turns on the horizontal prefilter for Y signal and turns off the horizontal prefilter for Cb/Cr signal.
	0	Turns off the horizontal prefilter.
RGB input	1	Turns on the horizontal prefilter for RGB signal.
	0	Turns off the horizontal prefilter.

**Table 33.13 Horizontal Prefilter Control**

Register Name	Bit Name	Initial Value	Description
SC_SCL0_DS4	SC_RES_PFIL_SEL	0	Prefilter Mode Select for Brightness Signals 0: The prefilter is turned off. 1: The prefilter is turned on. (1/4 + 1/2 + 1/4)

### 33.1.7 Horizontal Scale-Down

The number of horizontally arranged pixels can be decreased at a desired ratio in the range of 1/1 to 1/8 using pixel conversion.

For the scaling filter, either hold or linear interpolation mode can be selected.

#### (1) One-TAP Hold Interpolation

When the interpolation position is between input pixels  $X_n$  and  $X_{n+1}$ , the  $X_{interpo}$  interpolation value is defined as follows.

$$X_{interpo} = X_n$$

#### (2) Two-TAP Linear Interpolation

When the interpolation position is between input pixels  $X_n$  and  $X_{n+1}$ , the  $X_{interpo}$  interpolation value is defined as follows based on the interpolation position "phase".

$$X_{interpo} = (X_n \times (4096 - \text{phase}) + X_{n+1} \times \text{phase}) / 4096$$

#### (3) Calculation of Horizontal Scale Down Ratio

The value to be set to the horizontal scale-down ratio SC\_RES\_DS\_H\_RATIO can be obtained using the following equation based on the number of input pixels SC\_RES\_HW and number of output pixels SC\_RES\_OUT\_HW, where the decimals are rounded off.

$$\text{SC\_RES\_DS\_H\_RATIO} = \text{round}(\text{SC\_RES\_HW} \div \text{SC\_RES\_OUT\_HW} \times 4096)$$

Note that, for 100% horizontal scale-up, the SC\_RES\_HW and SC\_RES\_OUT\_HW values should be identical and the SC\_RES\_DS\_H\_RATIO bits should be set to 4096.

#### (4) Handling for Lack of Last-Input Pixel

Interpolation is carried out between the second-last-input and last-input pixels to produce the last-output pixel at the right end of a screen. The interpolation position of the last-output pixel may be close to the second-last-input pixel depending on the horizontal scale-down ratio; in this case, it may appear that the last-input pixel is lacking.

The undesirable influence by lack of last-input pixel can be decreased by appropriately adjusting the horizontal scale-down ratio using the following equations.

Pre-adjustment horizontal scale-down ratio  $RATIO\_org$  should be calculated first to find adjustment value  $\sigma$ , and then scale-down ratio  $SC\_RES\_DS\_H\_RATIO$  should be determined.

$$RATIO\_org = \text{round} (SC\_RES\_HW \div SC\_RES\_OUT\_HW \times 4096)$$

$$\sigma = (RATIO\_org \times (SC\_RES\_OUT\_HW - 1) - (SC\_RES\_HW - 1) \times 4096) \div (SC\_RES\_OUT\_HW - 1)$$

$$SC\_RES\_DS\_H\_RATIO = \text{roundup} (RATIO\_org - \sigma)$$

**Table 33.14 Horizontal Scale Down Control**

Register Name	Bit Name	Initial Value	Description
SC_SCL0_DS1	SC_RES_DS_H_ON	1	Horizontal Scale Down On/Off 0: Off 1: On
SC_SCL0_DS7	SC_RES_OUT_HW[10:0]	640	Number of Valid Horizontal Pixels Output by Scaling-down Control Block (Video-image clock cycles)
SC_SCL0_DS4	SC_RES_DS_H_INTERPOTYP	1	Horizontal Interpolation Mode Select 0: Hold interpolation 1: Linear interpolation
SC_SCL0_DS4	SC_RES_DS_H_RATIO[15:0]	9224	Horizontal Scale Down Ratio ([15:12]: Integer part, [11:0]: Decimal part) $\text{round}(SC\_RES\_HW \div SC\_RES\_OUT\_HW \times 4096)$ $SC\_RES\_DS\_H\_RATIO < 4096$ : Setting prohibited $SC\_RES\_DS\_H\_RATIO = 4096$ : 100% scale up $SC\_RES\_DS\_H\_RATIO > 4096$ : Scale down

Note: The  $SC\_RES\_OUT\_HW$  value should be aligned in 4-pixel units and equal to or smaller than the  $SC\_RES\_HW$  value.

### 33.1.8 Vertical Scale-Down

The number of lines can be decreased in the vertical direction at a desired ratio in the range of 1/1 to 1/8 using pixel conversion.

For the scaling filter, either hold or linear interpolation mode can be selected.

#### (1) One-TAP Hold Interpolation

When the interpolation position is between input lines  $X_n$  and  $X_{n+1}$ , the  $X_{interpo}$  interpolation value is defined as follows.

$$X_{interpo} = X_n$$

#### (2) Two-TAP Linear Interpolation

When the interpolation position is between input lines  $X_n$  and  $X_{n+1}$ , the  $X_{interpo}$  interpolation value is defined as follows based on the interpolation position "phase".

$$X_{interpo} = (X_n \times (4096 - \text{phase}) + X_{n+1} \times \text{phase}) / 4096$$

#### (3) Calculation of Vertical Scale Down Ratio

The value to be set to the vertical scale-down ratio SC\_RES\_V\_RATIO can be obtained using the following equation based on the number of input lines SC\_RES\_VW and number of output lines SC\_RES\_OUT\_VW, where the decimals are rounded off.

$$\text{SC\_RES\_V\_RATIO} = \text{round} (\text{SC\_RES\_VW} \div \text{SC\_RES\_OUT\_VW} \times 4096)$$

Note that the SC\_RES\_VW and SC\_RES\_OUT\_VW values should be identical for vertical enlargement or 100% vertical enlargement.

For 100% vertical enlargement, reduction is carried out assuming SC\_RES\_V\_RATIO as 4096.

#### (4) Handling for Lack of Last-Input Line

Interpolation is carried out between the second-last-input and last-input lines to produce the last-output line at the lower end of a screen. The interpolation position of the last-output line may be close to the second-last-input line depending on the vertical scale-down ratio; in this case, it may appear that the last-input line is lacking.

The undesirable influence by the lack of last-input line can be decreased by appropriately adjusting the vertical scale-down ratio using the following equations.

Pre-adjustment vertical scale-down ratio  $\text{RATIO\_org}$  should be calculated first to find adjustment value  $\sigma$ , and then scale-down ratio SC\_RES\_V\_RATIO should be determined.

$$\text{RATIO\_org} = \text{round} (\text{SC\_RES\_VW} \div \text{SC\_RES\_OUT\_VW} \times 4096)$$

$$\sigma = (\text{RATIO\_org} \times (\text{SC\_RES\_OUT\_VW} - 1) - (\text{SC\_RES\_VW} - 1) \times 4096) \div (\text{SC\_RES\_OUT\_VW} - 1)$$

$$\text{SC\_RES\_V\_RATIO} = \text{round} (\text{RATIO\_org} - \sigma)$$

**Table 33.15 Vertical Scale Down Control**

Register Name	Bit Name	Initial Value	Description
SC_SCL0_DS1	SC_RES_DS_V_ON	1	Vertical Scale Down On/Off 0: Off 1: On
SC_SCL0_DS7	SC_RES_OUT_VW [10:0]	240	Number of Valid Lines in Vertical Direction Output by Scaling-Down Control Block (lines) This bit setting is used for the number of lines to be written to the frame buffer. When SC_SCL1_WR1.SC_RES_LOOP is 0 (frame write mode), these bits specify the number of lines for one frame. When SC_SCL1_WR1.SC_RES_LOOP is 1 (line write mode), these bits specify the number of lines for writing in a ring configuration.
SC_SCL0_DS5	SC_RES_V_INTERPOTYP	1	Vertical Interpolation Mode Select 0: Hold interpolation 1: Linear interpolation
SC_SCL0_DS6	SC_RES_V_RATIO [15:0]	2044	Vertical Scale UP/Down Ratio ([15:12]: Integer part, [11:0]: Decimal part) For scale down: $\text{round}(\text{SC\_RES\_VW} + \text{SC\_RES\_OUT\_VW} \times 4096)$ For scale up: $\text{round}(\text{SC\_RES\_IN\_VW} + \text{SC\_RES\_P\_VW} \times 4096)$ SC_RES_V_RATIO < 4096: Scale up SC_RES_V_RATIO = 4096: 100% scale up SC_RES_V_RATIO > 4096: Scale down

Note: SC\_RES\_V\_RATIO and SC\_RES\_V\_INTERPOTYP are both shared by vertical reduction and vertical enlargement.

It is impossible to use vertical reduction and vertical enlargement simultaneously because they are mutually exclusive.

The SC\_RES\_OUT\_VW value should be aligned in 4-line units and equal to or smaller than the SC\_RES\_VW value.



### 33.1.9 Horizontal Scale Up

The number of horizontally arranged pixels can be increased at a desired ratio in the range of 1/1 to 8/1 using pixel conversion.

For the scaling filter, either hold or linear interpolation mode can be selected.

#### (1) One-TAP Hold Interpolation

When the interpolation position is between input pixels  $X_n$  and  $X_{n+1}$ , the  $X_{interpo}$  interpolation value is defined as follows.

$$X_{interpo} = X_n$$

#### (2) Two-TAP Linear Interpolation

When the interpolation position is between input pixels  $X_n$  and  $X_{n+1}$ , the  $X_{interpo}$  interpolation value is defined as follows based on the interpolation position "phase".

$$X_{interpo} = (X_n \times (4096 - \text{phase}) + X_{n+1} \times \text{phase}) / 4096$$

#### (3) Calculation of Horizontal Scale Up Ratio

The value to be set to the horizontal scale-up ratio SC\_RES\_US\_H\_RATIO can be obtained using the following equation based on the number of input pixels SC\_RES\_IN\_HW and number of output pixels SC\_RES\_P\_HW, where the decimals are rounded off.

$$\text{SC\_RES\_US\_H\_RATIO} = \text{round} (\text{SC\_RES\_IN\_HW} \div \text{SC\_RES\_P\_HW} \times 4096)$$

Note that, for 100% horizontal scale-up, the SC\_RES\_IN\_HW and SC\_RES\_P\_HW values should be identical and the SC\_RES\_US\_H\_RATIO bits should be set to 4096.

#### (4) Folding Handling

Since interpolation is carried out between the last-input pixel and second-last-input folding pixel to produce the last-output pixel at the right end of a screen, folding may undesirably stand out depending on the horizontal scale up ratio.

The undesirable influence by folding pixels can be decreased by appropriately adjusting the horizontal scale-up ratio using the following equations.

Pre-adjustment horizontal scale-up ratio  $\text{RATIO\_org}$  should be calculated first to find adjustment value  $\sigma$ , and then scale-up ratio SC\_RES\_US\_H\_RATIO should be determined.

$$\text{RATIO\_org} = \text{round} (\text{SC\_RES\_IN\_HW} \div \text{SC\_RES\_P\_HW} \times 4096)$$

$$\sigma = (\text{RATIO\_org} \times (\text{SC\_RES\_P\_HW} - 1) - (\text{SC\_RES\_IN\_HW} - 1) \times 4096) \div (\text{SC\_RES\_P\_HW} - 1)$$

$$\text{SC\_RES\_US\_H\_RATIO} = \text{round} (\text{RATIO\_org} - \sigma)$$

**Table 33.16 Horizontal Scale Up Control**

Register Name	Bit Name	Initial Value	Description
SC_SCL0_US1	SC_RES_US_H_ON	1	Horizontal Scale Up On/Off 0: Off 1: On
SC_SCL0_US4	SC_RES_IN_HW[10:0]	640	Number of Valid Horizontal Pixels Input to Scaling-down Control Block (Pixel-clock cycles)
SC_SCL0_US6	SC_RES_US_H_INTERPOTYP	1	Horizontal Interpolation Mode Select 0: Hold interpolation 1: Linear interpolation
SC_SCL0_US5	SC_RES_US_H_RATIO [15:0]	9224	Horizontal Scale Up Ratio ([15:12]: Integer part, [11:0]: Decimal part) $\text{round}(\text{SC\_RES\_IN\_HW} \div \text{SC\_RES\_P\_HW} \times 4096)$ SC_RES_US_H_RATIO < 4096: Scale up SC_RES_US_H_RATIO = 4096: 100% scale-up SC_RES_US_H_RATIO > 4096: Setting prohibited

### 33.1.10 Vertical Scale-Up

The number of lines can be increased in the vertical direction at a desired ratio in the range of 1/1 to 8/1 using pixel conversion.

For the scaling filter, either hold or linear interpolation mode can be selected.

#### (1) One-TAP Hold Interpolation

When the interpolation position is between input lines  $X_n$  and  $X_{n+1}$ , the  $X_{interpo}$  interpolation value is defined as follows.

$$X_{interpo} = X_n$$

#### (2) Two-TAP Linear Interpolation

When the interpolation position is between input lines  $X_n$  and  $X_{n+1}$ , the  $X_{interpo}$  interpolation value is defined as follows based on the interpolation position "phase".

$$X_{interpo} = (X_n \times (4096 - \text{phase}) + X_{n+1} \times \text{phase}) / 4096$$

#### (3) Calculation of Vertical Scale Up Ratio

The value to be set to the vertical scale-up ratio SC\_RES\_V\_RATIO can be obtained using the following equation based on the number of input lines SC\_RES\_IN\_VW and number of output lines SC\_RES\_P\_VW, where the decimals are rounded off.

$$\text{SC\_RES\_V\_RATIO} = \text{round}(\text{SC\_RES\_IN\_VW} \div \text{SC\_RES\_P\_VW} \times 4096)$$

Note that, for 100% vertical enlargement or vertical reduction, the SC\_RES\_IN\_VW and SC\_RES\_P\_VW values should be identical.

#### (4) Folding Handling

The last line to be output at the bottom of the screen is produced by interpolation between the last line and line for folding (second-last line to be input). According to the vertical scale-up rate, this may cause folding to stand out.

The undesirable influence by folding lines can be decreased by appropriately adjusting the vertical scale-up ratio using the following equations.

Pre-adjustment vertical scale-up ratio  $RATIO\_org$  should be calculated first to find adjustment value  $\sigma$ , and then scale-up ratio  $SC\_RES\_V\_RATIO$  should be determined.

$$RATIO\_org = \text{round} (SC\_RES\_IN\_VW \div SC\_RES\_P\_VW \times 4096)$$

$$\sigma = (RATIO\_org \times (SC\_RES\_P\_VW - 1) - (SC\_RES\_IN\_VW - 1) \times 4096) \div (SC\_RES\_P\_VW - 1)$$

$$SC\_RES\_V\_RATIO = \text{round} (RATIO\_org - \sigma)$$

**Table 33.17 Vertical Scale Up Control**

Register Name	Bit Name	Initial Value	Description
SC_SCL0_US1	SC_RES_US_V_ON	1	Vertical Scale Up On/Off 0: Off 1: On
SC_SCL0_US4	SC_RES_IN_VW[10:0]	240	Number of Valid Lines in Vertical Direction Input to Scaling-down Control Block (Lines)
SC_SCL0_DS5	SC_RES_V_INTERPOTYP	1	Vertical Interpolation Mode Select 0: Hold interpolation 1: Linear interpolation
SC_SCL0_DS6	SC_RES_V_RATIO[15:0]	2044	Vertical Scale Up Ratio ([15:12]: Integer part, [11:0]: Decimal part) For scale down: $\text{round}(SC\_RES\_VW \div SC\_RES\_OUT\_VW \times 4096)$ For scale up: $\text{round}(SC\_RES\_IN\_VW \div SC\_RES\_P\_VW \times 4096)$ $SC\_RES\_V\_RATIO < 4096$ : Scale up $SC\_RES\_V\_RATIO = 4096$ : 100% scale up $SC\_RES\_V\_RATIO > 4096$ : Scale down

Note:  $SC\_RES\_V\_RATIO$  and  $SC\_RES\_V\_INTERPOTYP$  are both shared by vertical reduction and vertical enlargement.

It is impossible to use vertical reduction and vertical enlargement simultaneously because they are mutually exclusive.

### 33.1.11 IP Conversion

#### (1) Initial Phase Control

When interlaced signals are input, line flickering caused by the line offset between the top and bottom fields can be decreased before being displayed by independently adjusting the initial scaling phases of the fields.

For various operations, appropriate settings should be made referring to the relevant registers as listed in Table 33.18.

**Table 33.18 Initial Scaling Phase Settings (Standard Values) for IP Conversion**

Rotation	Horizontal Scaling	Vertical Scaling	Reference Bit (Setting)
Normal	Horizontal scale down	Vertical scale down	SC_RES_TOP_INIPHASE = 2048
	Horizontal scale down	Vertical scale up	SC_RES_TOP_INIPHASE = 2048
	Horizontal scale up	Vertical scale down	SC_RES_TOP_INIPHASE = 2048
	Horizontal scale up	Vertical scale up	SC_RES_TOP_INIPHASE = 2048
Horizontal mirroring	Horizontal scale down	Vertical scale down	SC_RES_TOP_INIPHASE = 2048
	Horizontal scale down	Vertical scale up	SC_RES_TOP_INIPHASE = 2048
	Horizontal scale up	Vertical scale down	SC_RES_TOP_INIPHASE = 2048
	Horizontal scale up	Vertical scale up	SC_RES_TOP_INIPHASE = 2048
90° rotation	(Horizontal input → vertical output) scale down	(Vertical input → horizontal output) scale down	SC_RES_TOP_INIPHASE = 2048
	(Horizontal input → vertical output) scale down	(Vertical input → horizontal output) scale up	SC_RES_TOP_INIPHASE = 2048
	(Horizontal input → vertical output) scale up	(Vertical input → horizontal output) scale up	SC_RES_US_HB_INIPHASE = 2048
180° rotation	Horizontal scale down	Vertical scale down	SC_RES_TOP_INIPHASE = 2048
	Horizontal scale down	Vertical scale up	SC_RES_BTM_INIPHASE = 2048
	Horizontal scale up	Vertical scale down	SC_RES_TOP_INIPHASE = 2048
	Horizontal scale up	Vertical scale up	SC_RES_BTM_INIPHASE = 2048
270° rotation	(Horizontal input → vertical output) scale down	(Vertical input → horizontal output) scale down	SC_RES_TOP_INIPHASE = 2048
	(Horizontal input → vertical output) scale down	(Vertical input → horizontal output) scale up	SC_RES_TOP_INIPHASE = 2048
	(Horizontal input → vertical output) scale up	(Vertical input → horizontal output) scale up	SC_RES_US_HT_INIPHASE = 2048

Note: Set 0 to the initial phase control registers where the specific value is not shown in the table.  
Set 0 to all the initial phase control registers when progressive signals are input.

**Table 33.19 Initial Scaling Phase Control**

Register Name	Bit Name	Initial Value	Description
SC_SCL0_DS5	SC_RES_BTM_INIPHASE [11:0]	0	Vertical Interpolation Start Phase for Bottom Field 0 to 4095 (0 to approx. 1.0)
SC_SCL0_DS5	SC_RES_TOP_INIPHASE [11:0]	2048	Vertical Interpolation Start Phase for Top Field 0 to 4095 (0 to approx. 1.0)
SC_SCL0_US6	SC_RES_US_HB_INIPHASE [11:0]	0	Horizontal Interpolation Start Phase for Bottom Field 0 to 4095 (0 to approx. 1.0)
SC_SCL0_US6	SC_RES_US_HT_INIPHASE [11:0]	0	Horizontal Interpolation Start Phase for Top Field 0 to 4095 (0 to approx. 1.0)

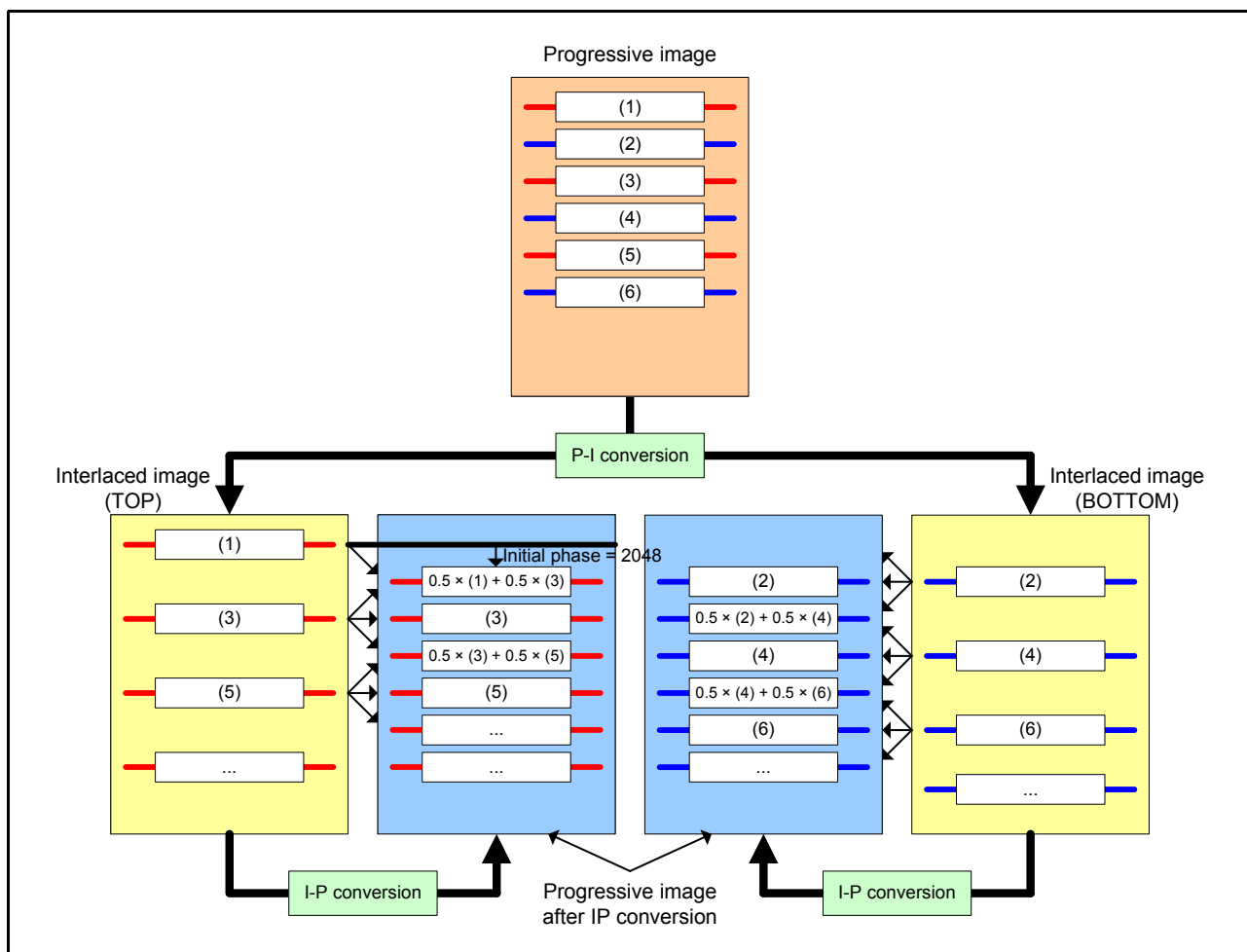


Figure 33.12 IP Conversion Processing Schematic Diagram

## (2) Field Determination Signal Control

When interlaced signals are input, the field determination signal can be controlled, which is output to the scaling-up control block during vertical scaling.

When progressive signals are input or vertical scaling is carried out by the scaling-down control block, the field determination signal output to the scaling-up control block is fixed to the specific level, and thus either 0 or 1 can be set to the SC\_RES\_FLD\_DLY\_SEL bit.

Table 33.20 Settings for Field Determination Signal Control

Input Signal	Rotation	Vertical Processing	Frame Buffer	SC_RES_FLD_DLY_SEL
Progressive	—	—	—	—
Interlace	Normal Horizontal mirroring 180° rotation	Vertical scale down	—	—
		Vertical scale up	One plane or less Two planes or more	0 1
	90° rotation 270° rotation	(Horizontal input → vertical output) scale down	—	—
		(Horizontal input → vertical output) scale up	Two planes or more	1

**Table 33.21 Field Determination Signal Control**

Register Name	Bit Name	Initial Value	Description
SC_SCL0_FRC5	SC_RES_FLD_DLY_SEL	1	Field Determination Signal Delay Control 0: No delay 1: Delay of one vertical cycle

### 33.1.12 Control of Interrupt on Specified Image Line before Scaling-down, and Reading of Current Image Line before Scaling-down

When the location of the image line input to the scaling-down control block matches the SC\_SCL1\_LINE setting, an interrupt processing is done. In addition, the current location of the line input to the scaling-down control block can be read from a register.

**Table 33.22 Control of Interrupt on Specified Image Line before Scaling-down, and Reading of Current Image line before Scaling-down**

Register Name	Bit Name	Initial Value	Description
SC_SCL0_INT	SC_RES_LINE[10:0]	All 0	Setting of Interrupt on Image Line Input to Scaling-down Control Block When the location of the image line input to the scaling-down control block matches the SC_SCL0_LINE setting, an interrupt signal is output.
SC_SCL0_MON0	SC_RES_LIN_STAT[10:0]	All 0	Current Location of Image Line Input to Scaling-down Control Block

### 33.1.13 Trimming

The upper, lower, right, and left parts of a post-scaling image can be trimmed off as specified by the SC\_RES\_V CUT and SC\_RES\_H CUT bits before being output.

The frame lines of the post-scaling image can also be displayed by setting the SC\_RES\_DISP\_ON bit to 1.

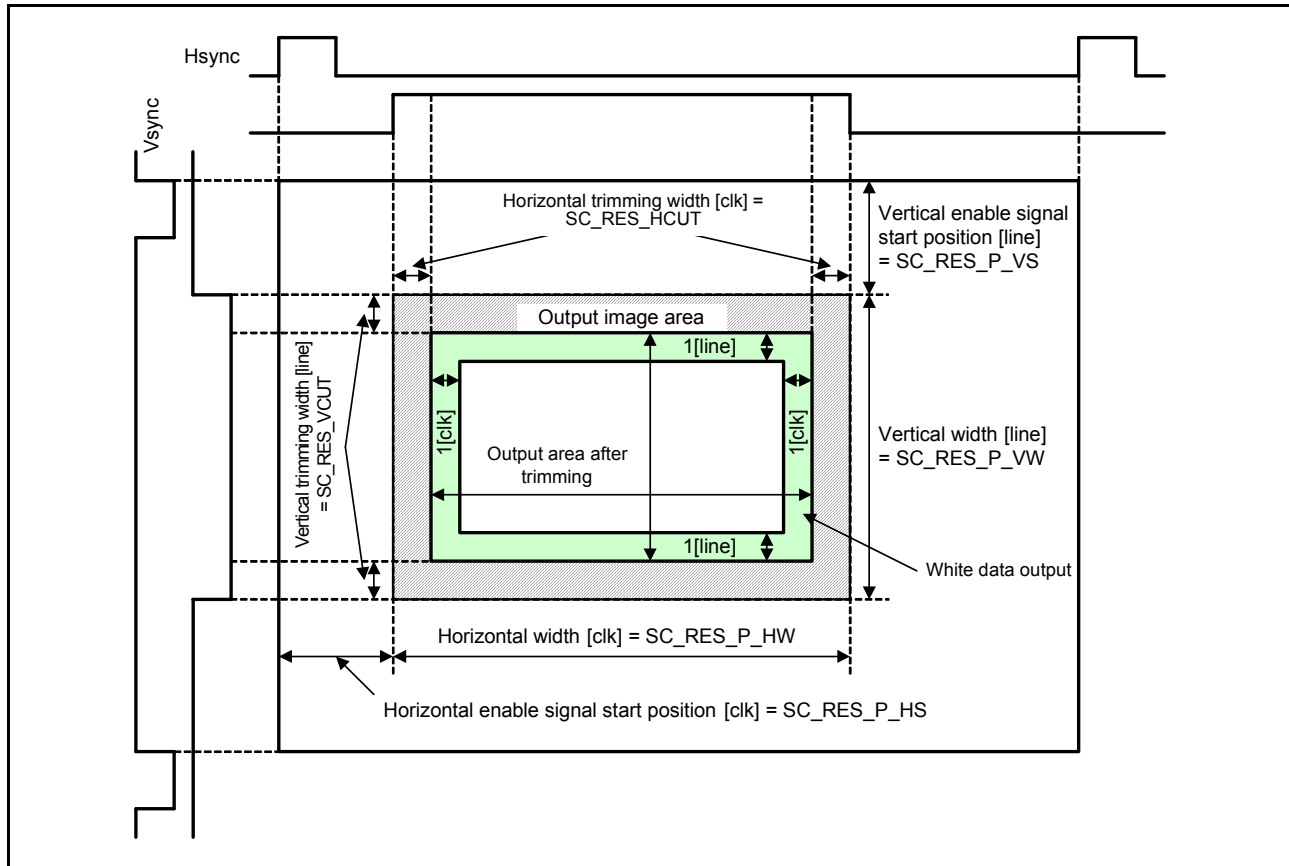


Figure 33.13 Area Relationship for Trimming (Frame Lines Displayed)

Table 33.23 Trimming Control

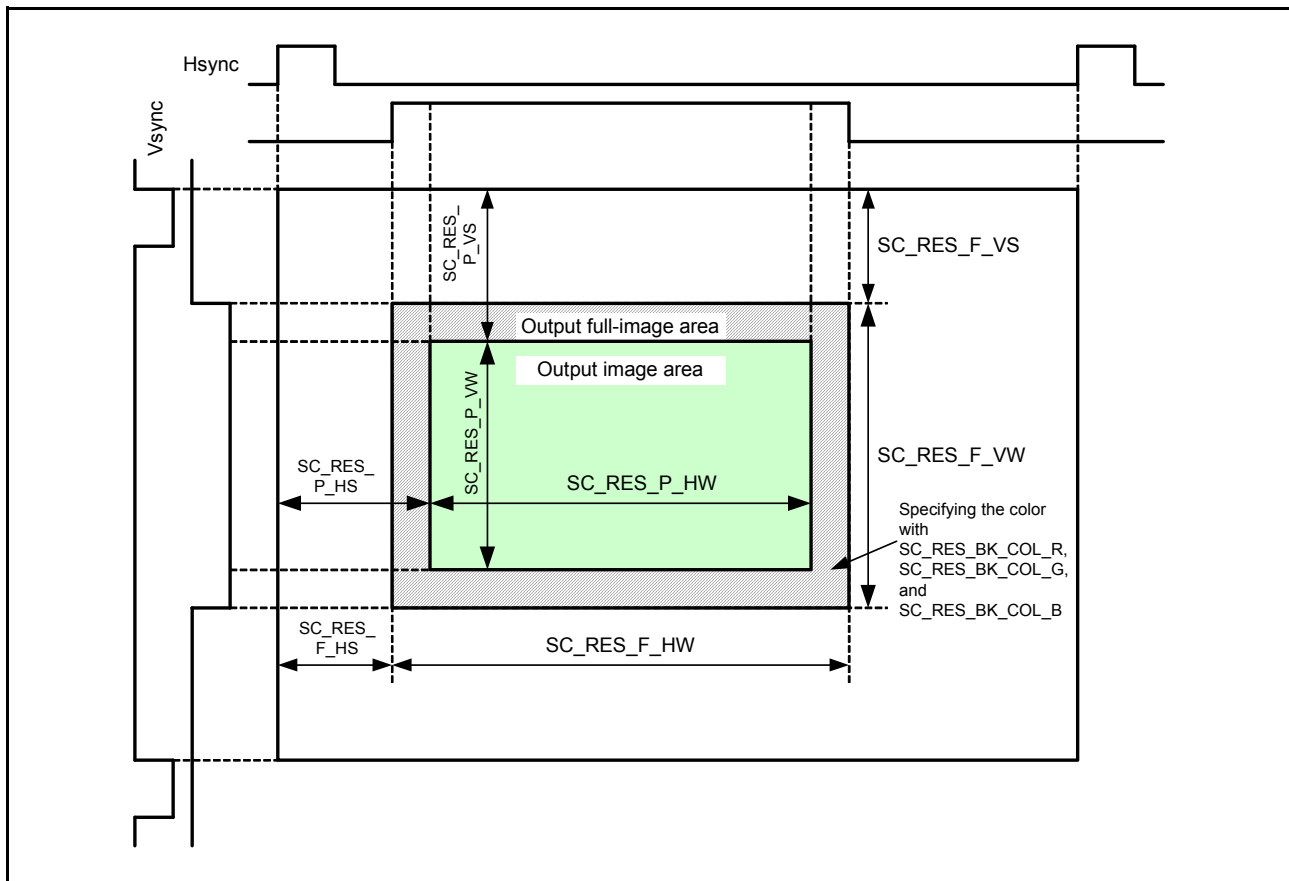
Register Name	Bit Name	Initial Value	Description
SC_SCL0_US7	SC_RES_H CUT[7:0]	0	Horizontal Amount of Cut-off Post-Scaling Image (Right and Left Parts) Sets the number of pixel-clock cycles.
SC_SCL0_US7	SC_RES_V CUT[7:0]	0	Vertical Amount of Cut-off Post-Scaling Image (Upper and Lower Parts) Sets the number of lines.
SC_SCL0_US8	SC_RES_DISP_ON	0	Post-Scaling Image Frame Display On/Off 0: Frame display on 1: Frame display off

### 33.1.14 Screen Synthesis

During the valid full-screen period, the image area can be overlaid before being output. If the image area to be output is smaller than a full-screen, the background color specified by the SC\_RES\_BK\_COL\_R, SC\_RES\_BK\_COL\_G, and SC\_RES\_BK\_COL\_B bits are displayed to fill the background.

**Table 33.24 Screen Synthesis Control**

Register Name	Bit Name	Initial Value	Description
SC_SCL0_OVR1	SC_RES_BK_COL_R [7:0]	128	Background Color Setting R/Cr Signal R: 8 bits; unsigned (0 to 255 [LSB]) Cr: 8 bits; 128 offset binary; unsigned (0 to 255 [LSB])
SC_SCL0_OVR1	SC_RES_BK_COL_B [7:0]	128	Background Color Setting B/Cb Signal B: 8 bits; unsigned (0 to 255 [LSB]) Cb: 8 bits; 128 offset binary; unsigned (0 to 255 [LSB])
SC_SCL0_OVR1	SC_RES_BK_COL_G [7:0]	0	Background Color Setting G/Y Signal G/Y: 8 bits; unsigned (0 to 255 [LSB])



**Figure 33.14 Area Relationship with Output Image Size Smaller than a Full Screen**



### 33.1.15 Selecting Format for Writing Video Image Signals to Frame Buffer

A format can be selected for writing video image signals to the frame buffer.

Although 24-bit YCbCr signals or 24-bit RGB signals are input to the scaling control block, they are converted into 16-bit YCbCr422 signals, 16-bit RGB565 signals, 32-bit YCbCr444 signals, or 32-bit RGB888 signals before being written to the frame buffer.

As bit reduction processing of RGB565, rounding off or  $2 \times 2$  pattern dither can be selected with the SC\_RES\_DTH\_ON bit. For details on pattern dither, see section 37.1.7, Dither Process in section 37, Video Display Controller 5 (7): Output Controller.

Input YCbCr signals are converted into YCbCr422 signals and output to the image renderer. Note that only scaler 0 can output signals to the image renderer.

For distortion correction, refer to section 41, Image Renderer (IMR-LS2).

**Table 33.25 Frame Buffer Writing Mode Setting**

RES_BITDEC_ON	SC_RES_MD[1:0]	Writing Mode
0	3	YCbCr444 (normal, horizontal mirroring)
0	2	RGB888 (normal, horizontal mirroring)
1	1	RGB565 (normal, horizontal mirroring, rotation)
0	0	YCbCr422 (normal, horizontal mirroring, rotation)
1	0	YCbCr422 (distortion correction)*

Note: \* Only scaler 0 can output signals to the image renderer.

**Table 33.26 Video Signal Writing Format Selection Control**

Register Name	Bit Name	Initial Value	Description
SC_SCL1_WR1	SC_RES_MD[1:0]	0	Frame Buffer Video-Signal Writing Format 0: YCbCr422 (16 bits) 1: RGB565 (16 bits) 2: RGB888 (24 (32) bits) 3: YCbCr444 (24 (32) bits)
SC_SCL1_WR6	SC_RES_BITDEC_ON	0	Bit Reduction On/Off 0: Off 1: On
SC_SCL1_WR6	SC_RES_DTH_ON	0	Dither Correction On/Off 0: Off (rounded off) 1: On ( $2 \times 2$ pattern dither)

### 33.1.16 Horizontal Mirroring and Rotation

Horizontal mirroring and rotation can be carried out for scaled-down images before being written to the frame buffer.

Table 33.27 and Table 33.28 show the relationship between various writing modes for image processing and video signals.

**Table 33.27 Relationship between Writing Modes and Video Signals**

RES_DS_WR_MD[2:0]	Writing Modes	YCbCr444	YCbCr422	RGB565	RGB888
0	Normal writing	Enabled	Enabled	Enabled	Enabled
1	Horizontal mirroring	Enabled	Enabled	Enabled	Enabled
2	90° rotation	Disabled	Enabled	Enabled	Disabled
3	180° rotation	Disabled	Enabled	Enabled	Disabled
4	270° rotation	Disabled	Enabled	Enabled	Disabled
5 to 7	Setting prohibited	—	—	—	—

**Table 33.28 Horizontal Mirroring and Rotation Control**

Register Name	Bit Name	Initial Value	Description
SC_SCL1_WR1	SC_RES_DS_WR_MD [2:0]	0	Frame Buffer Writing Mode for Image Processing 0: Normal 1: Horizontal mirroring 2: 90° rotation 3: 180° rotation 4: 270° rotation 5 to 7: Setting prohibited

### 33.1.17 Writing to Frame Buffer

#### (1) Frame Buffer Transfer Mode

Either 32-byte or 128-byte transfer mode can be selected for accessing the frame buffer in which video image data and graphics data are stored.

**Table 33.29 Frame Buffer Transfer Mode**

Register Name	Bit Name	Initial Value	Description
SC_SCL1_WR1	SC_RES_BST_MD	0	Transfer Burst Length for Frame Buffer Writing 0: 32-byte transfer (4 bursts) 1: 128-byte transfer (16 bursts)

#### (2) Frame Buffer Write Control

Frame buffer writing is enabled or disabled.

**Table 33.30 Frame Buffer Writing Control**

Register Name	Bit Name	Initial Value	Description
SC_SCL1_WR5	SC_RES_WENB	0	Frame Buffer Write Enable After making the setting to enable writing, writing starts from the second frame. 0: Frame buffer writing is disabled. 1: Frame buffer writing is enabled.

### (3) Frame Buffer Writing Rate Selection

A frame buffer writing rate can be selected from among 1/1, 1/2, 1/4, and 1/8 the vertical frequency of the input signal.

When 1/2, 1/4, or 1/8 is selected, either the top or bottom field can be selected for writing.

**Table 33.31 Frame Buffer Write Control**

Register Name	Bit Name	Initial Value	Description
SC_SCL1_WR5	SC_RES_FS_RATE [1:0]	0	Writing Rate Sets the frame buffer writing rate to the vertical frequency of the input signal. 0: 1/1 an input signal (The SC_RES_FLD_SEL setting is invalid.) 1: 1/2 an input signal 2: 1/4 an input signal 3: 1/8 an input signal
SC_SCL1_WR5	SC_RES_FLD_SEL	0	Write Field Select 0: Top field 1: Bottom field
SC_SCL1_WR5	SC_RES_INTER	1	Field Operating Mode Select 0: Progressive 1: Interlace

### (4) Frame Buffer Write Addresses

Frame buffer addresses are specified using the base address, line offset address, frame offset address, data size of a line, and the number of lines in a frame. When an interlaced video image is input, the top and bottom field data can be separately stored in the frame buffer.

The SC\_RES\_BASE[31:0], SC\_RES\_LN\_OFF[14:0], and SC\_RES\_FLM\_OFF[22:0] bits should be set in 32-byte units (the lower five bits should be fixed to 0).

For 128-byte transfer, bits [6:5] in the address control registers should be fixed to 0 since addresses should be specified in 128-byte units.

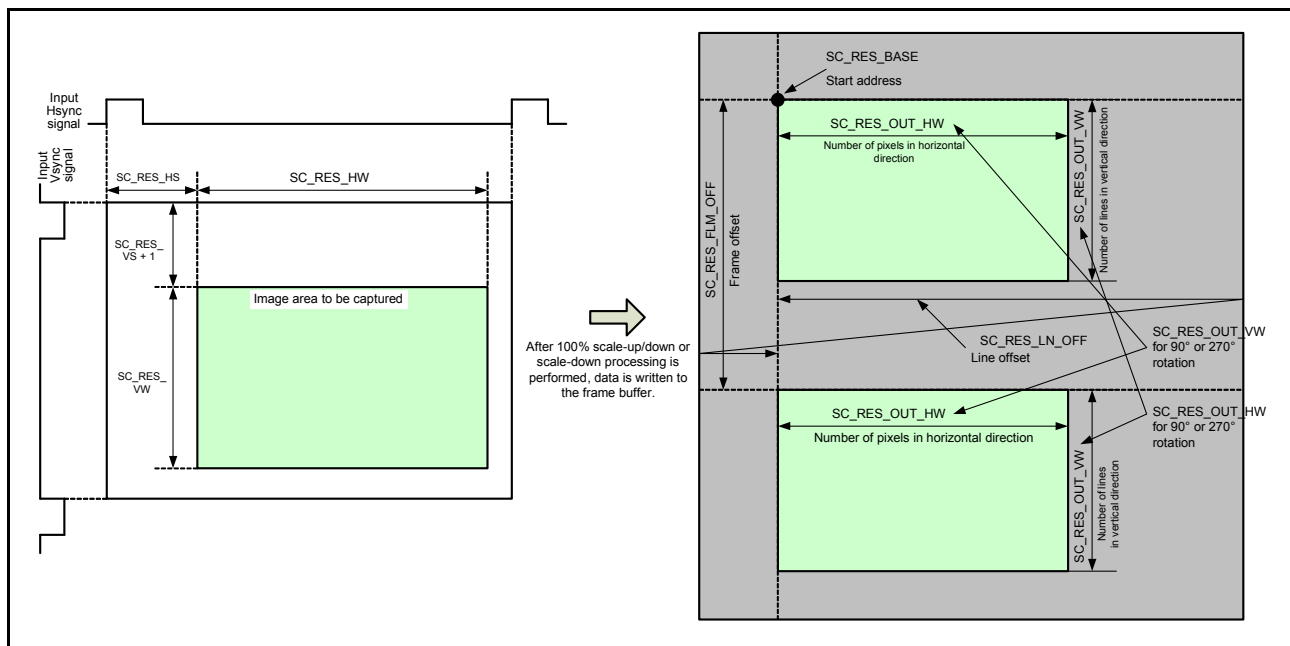
For the data size of a line and the number of lines in a frame, the relevant register values set for the scaling-down control block are used.

**Table 33.32 Frame Buffer Write Address Control**

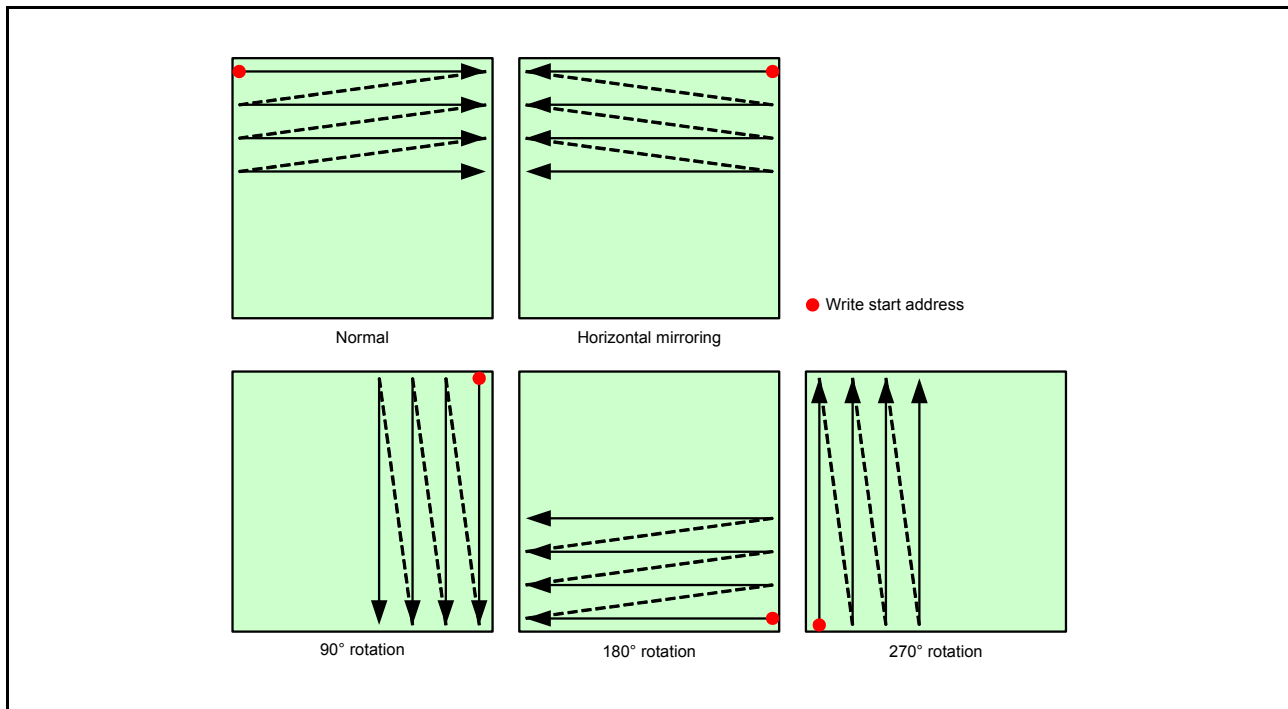
Register Name	Bit Name	Initial Value	Description
SC_SCL1_WR1	SC_RES_TB_ADD_MOD	0	Top and Bottom Data Write Address Specification Method 0: A write address is specified in common for top and bottom data. 1: Separate write addresses are specified for top and bottom data.
SC_SCL1_WR2	SC_RES_BASE [31:0]	0	Frame Buffer Base Address Sets the start address of the frame buffer to store the frame data for the top field when SC_RES_TB_ADD_MOD = 1 or that for the top and bottom fields when SC_RES_TB_ADD_MOD = 0. For 32-byte transfer: The lower five bits should be fixed to 0_0000. For 128-byte transfer: The lower seven bits should be fixed to 000_0000.
SC_SCL1_WR8	SC_RES_BASE_B [31:0]	0	Frame Buffer Base Address for Bottom Sets the start address of the frame buffer to store the frame data for the bottom field when SC_RES_TB_ADD_MOD = 1. For 32-byte transfer: The lower five bits should be fixed to 0_0000. For 128-byte transfer: The lower seven bits should be fixed to 000_0000.
SC_SCL1_WR3	SC_RES_LN_OFF [14:0]	2048	Frame Buffer Line Offset Address Sets the line offset address for calculating the line start address for the top field when SC_RES_TB_ADD_MOD = 1 or that for the top and bottom fields when SC_RES_TB_ADD_MOD = 0. Line 0: SC_RES_BASE Line 1: SC_RES_BASE + SC_RES_LN_OFF × 1 : Line n: SC_RES_BASE + SC_RES_LN_OFF × n For 32-byte transfer: The lower five bits should be fixed to 0_0000. For 128-byte transfer: The lower seven bits should be fixed to 000_0000.
SC_SCL1_WR9	SC_RES_LN_OFF_B [14:0]	2048	Frame Buffer Line Offset Address for Bottom Sets the line offset address for calculating the line start address for the bottom field when SC_RES_TB_ADD_MOD = 1. Line 0: SC_RES_BASE_B Line 1: SC_RES_BASE_B + SC_RES_LN_OFF_B × 1 : Line n: SC_RES_BASE_B + SC_RES_LN_OFF_B × n For 32-byte transfer: The lower five bits should be fixed to 0_0000. For 128-byte transfer: The lower seven bits should be fixed to 000_0000.
SC_SCL1_WR4	SC_RES_FLM_OFF [22:0]	524288	Frame Buffer Frame Offset Address Sets the frame offset address for calculating the start address of each frame for the top field when SC_RES_TB_ADD_MOD = 1 or that for the top and bottom fields when SC_RES_TB_ADD_MOD = 0. Buffer 0: SC_RES_BASE Buffer 1: SC_RES_BASE + SC_RES_FLM_OFF × 1 : Buffer n: SC_RES_BASE + SC_RES_FLM_OFF × n For 32-byte transfer: The lower five bits should be fixed to 0_0000. For 128-byte transfer: The lower seven bits should be fixed to 000_0000.

**Table 33.32 Frame Buffer Write Address Control**

Register Name	Bit Name	Initial Value	Description
SC_SCL1_WR10	SC_RES_FLM_OFF_B [22:0]	524288	<p>Frame Buffer Frame Offset Address for Bottom</p> <p>Sets the frame offset address for calculating the start address of each frame for the bottom field when SC_RES_TB_ADD_MOD = 1.</p> <p>Buffer 0: SC_RES_BASE_B                      Buffer 1: SC_RES_BASE_B + SC_RES_FLM_OFF_B × 1                      :                      Buffer n: SC_RES_BASE_B + SC_RES_FLM_OFF_B × n</p> <p>For 32-byte transfer: The lower five bits should be fixed to 0_0000.                      For 128-byte transfer: The lower seven bits should be fixed to 000_0000.</p>



**Figure 33.15 Data Arrangement in Frame Buffer**



**Figure 33.16 Data Arrangement in Frame Buffer in Various Writing Modes**

### (5) Frame Buffer Management

The scaling control block can handle multiple frames as the frame buffer.

Data is written to the buffer in cyclic mode according to the number of frames specified by the SC\_RES\_FLM\_NUM bits.

For rotation, the SC\_RES\_FLM\_NUM bits should be set to two or more frames.

To use the frame buffer as the ring buffer in line mode, the SC\_RES\_FLM\_NUM bits should be set to 0 (1 frame) and the SC\_RES\_LOOP bit to 1.

**Table 33.33 Frame Buffer Write Control**

Register Name	Bit Name	Initial Value	Description
SC_SCL1_WR3	SC_RES_FLM_NUM[9:0]	1	Number of Frames of Buffer to be Written to Sets the number of frames for the top field when SC_RES_TB_ADD_MOD = 1 or that for the top and bottom fields when SC_RES_TB_ADD_MOD = 0 Number of frames defined by SC_RES_FLM_NUM + 1 are used.
SC_SCL1_WR9	SC_RES_FLM_NUM_B[9:0]	1	Number of Frames of Buffer to be Written to for Bottom Field when SC_RES_TB_ADD_MOD = 1 Number of frames defined by SC_RES_FLM_NUM_B + 1 are used.
SC_SCL1_WR1	SC_RES_LOOP	0	Frame Buffer Write Mode Select 0: Frame mode 1: Line mode (read as ring buffer)
SC_SCL1_WR7	SC_RES_FLM_CNT[9:0]	—	Frame Number Before Frame Being Accessed Frame number before the frame being accessed in the top field when SC_RES_TB_ADD_MOD = 1 or that in the top or bottom field when SC_RES_TB_ADD_MOD = 0.
SC_SCL1_WR11	SC_RES_FLM_CNT_B[9:0]	—	Frame Number Before Frame Being Accessed in Bottom Field Frame number before the frame being accessed in the bottom field when SC_RES_TB_ADD_MOD = 1.

#### (6) Buffer Overflow Handling

If writing to the frame buffer cannot be completed due to bus-traffic related problems, an overflow interrupt can be output to the interrupt controller.

**Table 33.34 Buffer Overflow Detection**

Register Name	Bit Name	Initial Value	Description
SC_SCL1_WR7	SC_RES_OVERFLOW	—	Line Buffer Overflow Detect 1: Line buffer has overflowed. 0: Line buffer has not overflowed.

#### (7) Frame Buffer Write End Flag

When writing one frame of data to the frame buffer is completed, a frame buffer write end interrupt can be output to the interrupt controller.

### 33.1.18 Selecting a Scaling-up Process or Graphics 0 or 1 Process

Scaling-up process and graphics 0 or 1 process are mutually exclusive and thus frame buffer cannot be read out simultaneously for the processes.

When displaying input video image signals or displaying enlarged graphics, data is read from the frame buffer via the scaling-up control block.

However, graphics can be enlarged and displayed by the scaling-up control block only when the RGB565, RGB888, YCbCr422, or YCbCr444 format is used.

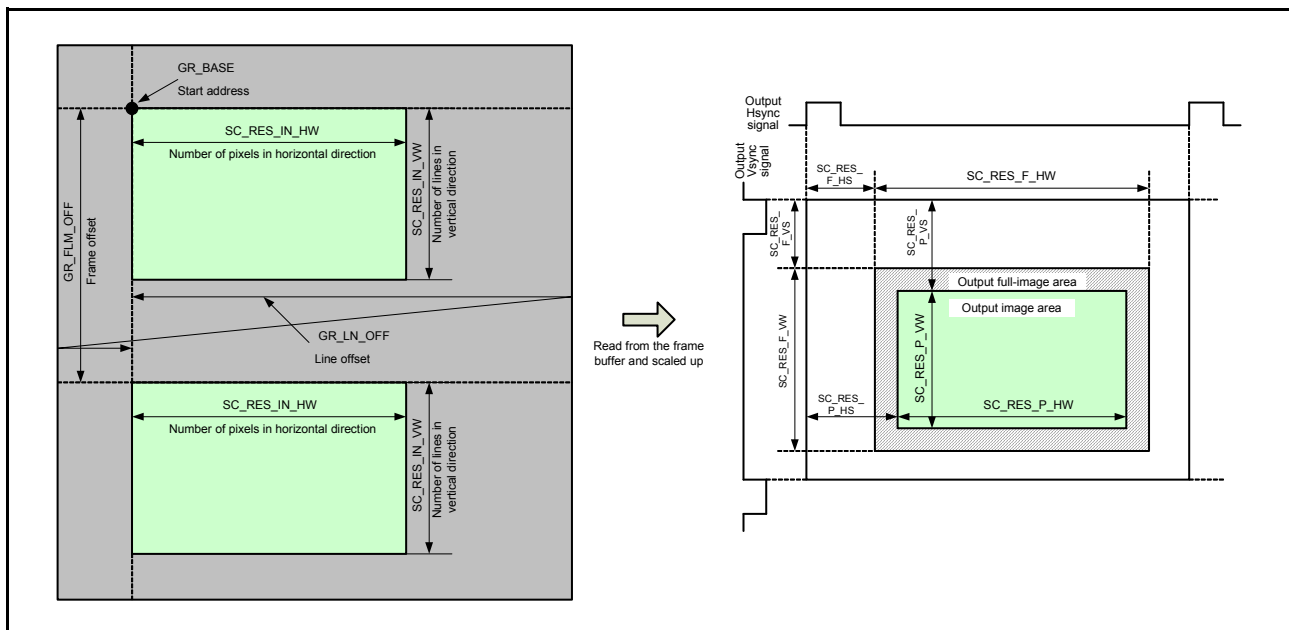
When displaying graphics without enlargement, the data is read from the frame buffer via the graphics 0 or 1 processing block.

With the SC\_RES\_IBUS\_SYNC\_SEL bit, sync signals for reading out the frame buffer and read size setting bits are selected.

**Table 33.35 Selection of Scaling-Up Process and Graphics 0 or 1 Process**

Type of Output Scaling Display	SC_RES_IBUS_SYNC_SEL	Sync Signals for Frame Buffer Read	Frame Buffer Read Size Setting Bits	Display Enabling Bits
Input video signal display Enlarged graphics display	0	Output from scaling-up control block	SC_RES_IN_VW SC_RES_IN_HW	SC_RES_P_VS SC_RES_P_VW SC_RES_P_HS SC_RES_P_HW
Graphics display	1	Output from graphics 0 or 1 processing block	GR_FLM_LNUM* GR_HW*	GR_GRC_VS GR_GRC_VW GR_GRC_HS GR_GRC_HW

Note: \* The value set to the register + 1 is the actual read size.



**Figure 33.17 Area Setting for Input Video Image Signal Display and Enlarged Graphics Display**



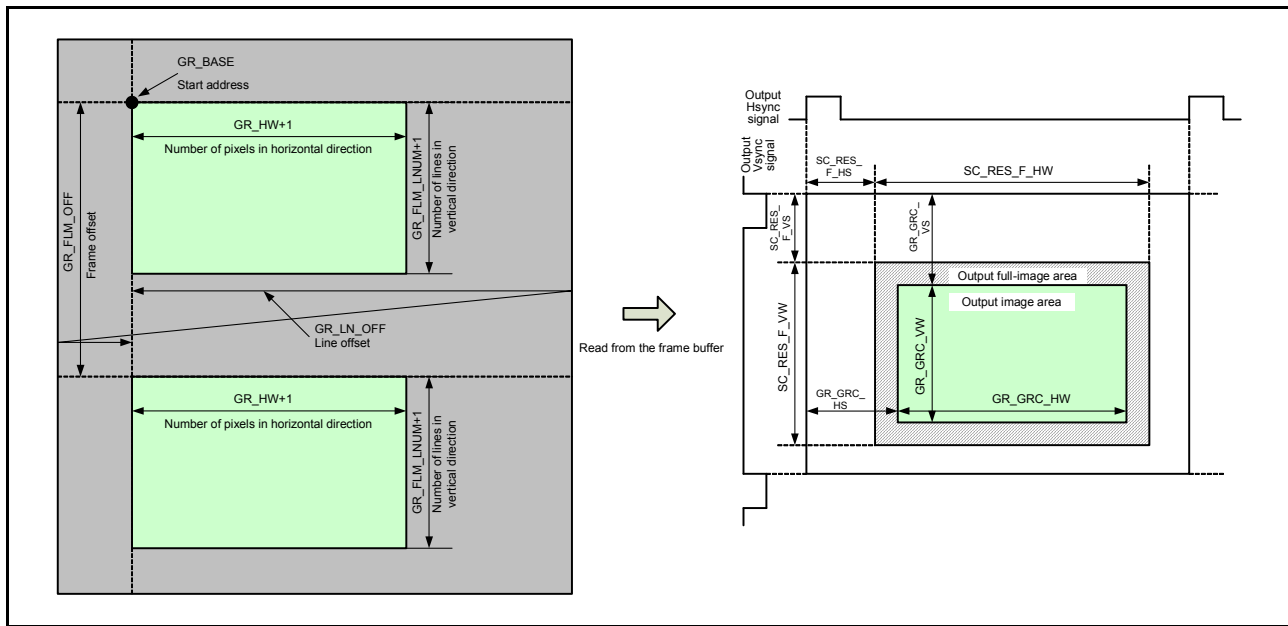


Figure 33.18 Area Setting for Graphics Display

Table 33.36 Scaling-Up Process or Graphics 0 or 1 Process Selection

Register Name	Bit Name	Initial Value	Description
SC_SCL0_US8	SC_RES_IBUS_SYNC_SEL	0	Sync Signal Select for Frame Buffer Read Block 0: Sync signals from the scaling-up control block 1: Sync signals from the graphics processing block

The GR\_DISP\_SEL bits are used to select a display by the scaling-up control block (video image display or enlarged graphics display) or graphics display.

For details on the graphics processing, refer to the section 35, Video Display Controller 5 (5): Image Synthesizer.

### 33.1.19 Selecting Field for Frame Buffer Reading

For the next frame buffer to be read, the top or bottom field can be selected. This field selection is used in the scaling-up control block.

Table 33.37 Field Specification for Frame Buffer Reading

Register Name	Bit Name	Initial Value	Description
GR_FLM1	GR_FLD_SEL	0	Enables or disables top or bottom field selection for the next frame buffer to be read. 0: Field selection is disabled. 1: Top or bottom field can be selected.
GR_FLM3	GR_FLD_NXT	0	Selects the top or bottom field for the next frame buffer. 0: Bottom 1: Top

### 33.1.20 Pointer Buffer and Frame Buffer Reading Processing

#### (1) Pointer Buffers

The pointer buffers can be used to control the frame buffer for the input video image. They are mainly used to prevent flicker in the output video image, which occurs when the input and output vertical sync signals are asynchronous.

Four pointer buffers are provided and each pointer buffer has a start address register that shows the start location of the frame buffer and a field information register that shows the current field is the top or bottom field. The four pointer buffers are arranged in a ring structure and a write pointer is provided to indicate the pointer buffer corresponding to the location currently being written to. The location pointed to by the write pointer is being written to, and the corresponding pointer buffer value is undetermined. The value in the pointer buffer corresponding to the location being written to and the value in the write pointer are automatically updated when frame data writing is completed.

When the frame buffer address setting signal is linked with the pointer buffer ( $GR\_FLM\_SEL = 3$ ), frame buffer reading can be controlled by using the read pointer that indicates the pointer buffer corresponding to the location being read; the start address and field information of the next frame buffer to be read should be read from the pointer buffer and they should be set in the frame buffer base address and field information. The read pointer value is automatically updated at the rising edge of the vertical sync signal on the reading side.

#### (2) Write Pointer Control

The write pointer is incremented by one every time frame data writing is completed.

#### (3) Read Pointer Control

The read pointer is updated at the rising edge of the vertical sync signal on the reading side according to the difference between the read and write pointer values as follows.

(A) When  $(\text{write pointer value}) - (\text{read pointer value}) \leq 1$

The read pointer value is not updated (the same frame is displayed continuously).

(B) When  $(\text{write pointer value}) - (\text{read pointer value}) = 2$

The read pointer is incremented by one with the next updating timing.

(C) When  $(\text{write pointer value}) - (\text{read pointer value}) \geq 3$

The read pointer is incremented by two with the next updating timing.

(One frame is skipped.)

#### (4) Frame Buffer Read Control

(A) When  $SC\_RES\_WENB = 0$

Frame data is not written to the frame buffer, and the frame buffer is not read.

(B) When frame data writing is terminated with  $SC\_RES\_WENB = 1$

As the pointer buffer value is determined, the frame buffer is read.

**Table 33.38 Frame Buffer Control**

Register Name	Bit Name	Initial Value	Description
SC_SCL1_PBUF0	SC_BUF0_ADD	0	Start address of the write buffer pointed to by pointer buffer 0
SC_SCL1_PBUF1	SC_BUF1_ADD	0	Start address of the write buffer pointed to by pointer buffer 1
SC_SCL1_PBUF2	SC_BUF2_ADD	0	Start address of the write buffer pointed to by pointer buffer 2
SC_SCL1_PBUF3	SC_BUF3_ADD	0	Start address of the write buffer pointed to by pointer buffer 3
SC_SCL1_PBUF_FLD	SC_FLD_INF0	0	Top or bottom field information pointed to by pointer buffer 0 0: Bottom 1: Top
SC_SCL1_PBUF_FLD	SC_FLD_INF1	0	Top or bottom field information pointed to by pointer buffer 1 0: Bottom 1: Top
SC_SCL1_PBUF_FLD	SC_FLD_INF2	0	Top or bottom field information pointed to by pointer buffer 2 0: Bottom 1: Top
SC_SCL1_PBUF_FLD	SC_FLD_INF3	0	Top or bottom field information pointed to by pointer buffer 3 0: Bottom 1: Top
SC_SCL1_PBUF_CNT	SC_PBUF_RST	0	Reset Control for the Pointer Buffer 0: Pointer buffer is not reset. 1: Pointer buffer is reset.
SC_SCL1_MON1	SC_PBUF_NUM	0	Write pointer indicating the pointer buffer number corresponding to the location currently being written to.
SC_SCL1_WR5	SC_RES_WENB	0	Frame Buffer Write Enable After making the setting to enable writing, writing starts from the second frame. 0: Writing is disabled. 1: Writing is enabled.
GR_FLM1	GR_FLM_SEL	0	Frame Buffer Address Setting Signal Selection 0: Links to scaling-down process. (This setting is prohibited when separate write addresses are specified for the top and bottom fields; that is, SC0_RES_TB_ADD_MOD = 1 in SC_SCL0_FRC3.) 1: Selects GR0_FLM_NUM. 2: Links to distortion correction. 3: Links to pointer buffer.
GR_FLM2	GR_BASE	0	Frame Buffer Base Address Sets the start address of the frame buffer where frame data is to be stored. GR_BASE[4:3] and GR_BASE[6:3] are referred to during 32-byte burst transfer and 128-byte burst transfer, respectively, to skip the start line data. The lower three bits should be fixed to 000.

For other frame buffer read operation and graphics processing, refer to section 35, Video Display Controller 5 (5): Image Synthesizer.

### 33.1.21 Cascaded Connection

To display one video image plane + three graphics planes or to display four graphics planes, scaler 0 and graphics block 1 in scaler 1 are cascaded.

When cascaded connection is selected, the scaling-up processing in scaler 1 is not available.

**Table 33.39 Cascaded Connection**

Register Name	Bit Name	Initial Value	Description
GR1_AB1	GR1_CUS_CON_ON	0	Cascaded Connection Enable/Disable 0: Cascaded connection is disabled. 1: Cascaded connection is enabled.

### 33.1.22 Blending Two Input Video Images

To display two video image planes + two graphics planes, the video images input from scalers 0 and 1 are blended in the VIN synthesizer. To blend two input video images, the vertical and horizontal sync signal output and the full-screen enable signal output from scalers 0 and 1 should be synchronized. When the vertical sync signal output is synchronized with the vertical sync signal generated in the synchronization control block in scaler 0, the vertical sync signal input and output become asynchronous in scaler 1. This timing difference may cause flicker in the output video image; to avoid this, use the pointer buffers described before to control the frame buffer in scaler 1.

Likewise, when synchronizing with the output vertical sync signal generated in the synchronization control block in scaler 1, use the pointer buffers to control the frame buffer in scaler 0.

**Table 33.40 Blending Two Input Video Images**

Register Name	Bit Name	Initial Value	Description
SC_SCL0_FRC3	SC_RES_VS_IN_SEL	0	Horizontal and Vertical Sync Signal Output and Full-Screen Enable Signal Select 0: Horizontal and vertical sync signal output and full-screen enable signal from the current scaler 1: Horizontal and vertical sync signal output and full-screen enable signal from the other scaler

## 33.2 Register Descriptions

Table 33.41 and Table 33.42 show the register configuration.

- Symbols used in Register Description:

Initial value: Register value after a reset

—: Undefined value

R/W: Readable/writable. The written value can be read.

R/WC0: Readable/writable. Writing 0 initializes the bit. Writing 1 is ignored.

R/WC1: Readable/writable. Writing 1 initializes the bit. Writing 0 is ignored.

R: Read-only. The write value should always be 0.

—/W: Write-only. The read value is undefined.

**Table 33.41 Register Configuration of the Scaler (Channel 0)**

Name	Abbreviation	R/W	Initial Value	Address	Access Size
SCL0 register update control register (SC0)	SC0_SCL0_UPDATE	R/WC1	H'0000 0000	H'FCFF 7500	32
Mask control register (SC0)	SC0_SCL0_FRC1	R/W	H'0AF0 0001	H'FCFF 7504	32
Missing Vsync compensation control register (SC0)	SC0_SCL0_FRC2	R/W	H'0E10 0001	H'FCFF 7508	32
Output sync select register (SC0)	SC0_SCL0_FRC3	R/W	H'0000 0001	H'FCFF 750C	32
Free-running period control register (SC0)	SC0_SCL0_FRC4	R/W	H'020C 031F	H'FCFF 7510	32
Output delay control register (SC0)	SC0_SCL0_FRC5	R/W	H'0000 0101	H'FCFF 7514	32
Full-screen vertical size register (SC0)	SC0_SCL0_FRC6	R/W	H'0023 01E0	H'FCFF 7518	32
Full-screen horizontal size register (SC0)	SC0_SCL0_FRC7	R/W	H'0090 0280	H'FCFF 751C	32
Vsync detection register (SC0)	SC0_SCL0_FRC9	R	H'0000 0000	H'FCFF 7524	32
Status monitor 0 register (SC0)	SC0_SCL0_MON0	R	H'0000	H'FCFF 7528	16
Interrupt control register (SC0)	SC0_SCL0_INT	R/W	H'0000	H'FCFF 752A	16
Scaling-down control register (SC0)	SC0_SCL0_DS1	R/W	H'0000 0011	H'FCFF 752C	32
Vertical capture size register (SC0)	SC0_SCL0_DS2	R/W	H'0012 00F0	H'FCFF 7530	32
Horizontal capture size register (SC0)	SC0_SCL0_DS3	R/W	H'00F4 05A0	H'FCFF 7534	32
Horizontal scale down register (SC0)	SC0_SCL0_DS4	R/W	H'1000 2408	H'FCFF 7538	32
Initial vertical phase register (SC0)	SC0_SCL0_DS5	R/W	H'1800 0000	H'FCFF 753C	32
Vertical scaling register (SC0)	SC0_SCL0_DS6	R/W	H'0000 07FC	H'FCFF 7540	32
Scaling-down control block output size register (SC0)	SC0_SCL0_DS7	R/W	H'00F0 0280	H'FCFF 7544	32
Scaling-up control register (SC0)	SC0_SCL0_US1	R/W	H'0000 0011	H'FCFF 7548	32
Output image vertical size register (SC0)	SC0_SCL0_US2	R/W	H'0023 01E0	H'FCFF 754C	32
Output image horizontal size register (SC0)	SC0_SCL0_US3	R/W	H'0090 0280	H'FCFF 7550	32
Scaling-up control block input size register (SC0)	SC0_SCL0_US4	R/W	H'00F0 0280	H'FCFF 7554	32
Horizontal scale up register (SC0)	SC0_SCL0_US5	R/W	H'0000 2408	H'FCFF 7558	32
Horizontal scale up initial phase register (SC0)	SC0_SCL0_US6	R/W	H'1000 0000	H'FCFF 755C	32
Trimming register (SC0)	SC0_SCL0_US7	R/W	H'0000 0000	H'FCFF 7560	32
Frame buffer read select register (SC0)	SC0_SCL0_US8	R/W	H'0000 0000	H'FCFF 7564	32
Background color register (SC0)	SC0_SCL0_OVR1	R/W	H'0080 0080	H'FCFF 756C	32

**Table 33.41 Register Configuration of the Scaler (Channel 0)**

Name	Abbreviation	R/W	Initial Value	Address	Access Size
SCL1 register update control register (SC0)	SC0_SCL1_UPDATE	R/WC1	H'0000 0000	H'FCFF 7580	32
Writing mode register (SC0)	SC0_SCL1_WR1	R/W	H'0000 0000	H'FCFF 7588	32
Write address register 1T (SC0)	SC0_SCL1_WR2	R/W	H'0000 0000	H'FCFF 758C	32
Write address register 2T (SC0)	SC0_SCL1_WR3	R/W	H'0800 0001	H'FCFF 7590	32
Write address register 3T (SC0)	SC0_SCL1_WR4	R/W	H'0008 0000	H'FCFF 7594	32
Frame sub-sampling register (SC0)	SC0_SCL1_WR5	R/W	H'0000 1000	H'FCFF 759C	32
Bit reduction register (SC0)	SC0_SCL1_WR6	R/W	H'0000 0000	H'FCFF 75A0	32
Write detection register (SC0)	SC0_SCL1_WR7	R	H'0000 0000	H'FCFF 75A4	32
Write address register 1B (SC0)	SC0_SCL1_WR8	R/W	H'0000 0000	H'FCFF 75A8	32
Write address register 2B (SC0)	SC0_SCL1_WR9	R/W	H'0800 0001	H'FCFF 75AC	32
Write address register 3B (SC0)	SC0_SCL1_WR10	R/W	H'0008 0000	H'FCFF 75B0	32
Write detection register B (SC0)	SC0_SCL1_WR11	R	H'0000 0000	H'FCFF 75B4	32
Status monitor 1 register (SC0)	SC0_SCL1_MON1	R	H'0000 0000	H'FCFF 75B8	32
Pointer buffer 0 register (SC0)	SC0_SCL1_PBUF0	R	H'0000 0000	H'FCFF 75BC	32
Pointer buffer 1 register (SC0)	SC0_SCL1_PBUF1	R	H'0000 0000	H'FCFF 75C0	32
Pointer buffer 2 register (SC0)	SC0_SCL1_PBUF2	R	H'0000 0000	H'FCFF 75C4	32
Pointer buffer 3 register (SC0)	SC0_SCL1_PBUF3	R	H'0000 0000	H'FCFF 75C8	32
Pointer buffer and field information register (SC0)	SC0_SCL1_PBUF_FLD	R	H'0000 0000	H'FCFF 75CC	32
Pointer buffer control register (SC0)	SC0_SCL1_PBUF_CNT	R/W	H'0000 0000	H'FCFF 75D0	32
Graphics 0 register update control register	GR0_UPDATE	R/WC1	H'0000 0000	H'FCFF 7600	32
Frame buffer read control register (graphics 0)	GR0_FLM_RD	R/W	H'0000 0000	H'FCFF 7604	32
Frame buffer control register 1 (graphics 0)	GR0_FLM1	R/W	H'0000 0000	H'FCFF 7608	32
Frame buffer control register 2 (graphics 0)	GR0_FLM2	R/W	H'0000 0000	H'FCFF 760C	32
Frame buffer control register 3 (graphics 0)	GR0_FLM3	R/W	H'0800 0001	H'FCFF 7610	32
Frame buffer control register 4 (graphics 0)	GR0_FLM4	R/W	H'0008 0000	H'FCFF 7614	32
Frame buffer control register 5 (graphics 0)	GR0_FLM5	R/W	H'0000 03FF	H'FCFF 7618	32
Frame buffer control register 6 (graphics 0)	GR0_FLM6	R/W	H'8000 0000	H'FCFF 761C	32
Alpha blending control register 1 (graphics 0)	GR0_AB1	R/W	H'0000 0000	H'FCFF 7620	32
Alpha blending control register 2 (graphics 0)	GR0_AB2	R/W	H'0000 0000	H'FCFF 7624	32
Alpha blending control register 3 (graphics 0)	GR0_AB3	R/W	H'0000 0000	H'FCFF 7628	32
Alpha blending control register 7 (graphics 0)	GR0_AB7	R/W	H'00FF 0000	H'FCFF 7638	32
Alpha blending control register 8 (graphics 0)	GR0_AB8	R/W	H'0000 0000	H'FCFF 763C	32
Alpha blending control register 9 (graphics 0)	GR0_AB9	R/W	H'0000 0000	H'FCFF 7640	32

**Table 33.41 Register Configuration of the Scaler (Channel 0)**

Name	Abbreviation	R/W	Initial Value	Address	Access Size
Alpha blending control register 10 (graphics 0)	GR0_AB10	R/W	H'0000 0000	H'FCFF 7644	32
Alpha blending control register 11 (graphics 0)	GR0_AB11	R/W	H'0000 0000	H'FCFF 7648	32
Background color control register (graphics 0)	GR0_BASE	R/W	H'0000 8080	H'FCFF 764C	32
CLUT table control register (graphics 0)	GR0_CLUT	R/W	H'0000 0000	H'FCFF 7650	32
SCL0 register update control register (SC1)	SC1_SCL0_UPDATE	R/WC1	H'0000 0000	H'FCFF 7C00	32
Mask control register (SC1)	SC1_SCL0_FRC1	R/W	H'0AF0 0001	H'FCFF 7C04	32
Missing Vsync compensation control register (SC1)	SC1_SCL0_FRC2	R/W	H'0E10 0001	H'FCFF 7C08	32
Output sync select register (SC1)	SC1_SCL0_FRC3	R/W	H'0000 0001	H'FCFF 7C0C	32
Free-running period control register (SC1)	SC1_SCL0_FRC4	R/W	H'020C 031F	H'FCFF 7C10	32
Output delay control register (SC1)	SC1_SCL0_FRC5	R/W	H'0000 0101	H'FCFF 7C14	32
Full-screen vertical size register (SC1)	SC1_SCL0_FRC6	R/W	H'0023 01E0	H'FCFF 7C18	32
Full-screen horizontal size register (SC1)	SC1_SCL0_FRC7	R/W	H'0090 0280	H'FCFF 7C1C	32
Vsync detection register (SC1)	SC1_SCL0_FRC9	R	H'0000 0000	H'FCFF 7C24	32
Status monitor 0 register (SC1)	SC1_SCL0_MON0	R	H'0000	H'FCFF 7C28	16
Interrupt control register (SC1)	SC1_SCL0_INT	R/W	H'0000	H'FCFF 7C2A	16
Scaling-down control register (SC1)	SC1_SCL0_DS1	R/W	H'0000 0011	H'FCFF 7C2C	32
Vertical capture size register (SC1)	SC1_SCL0_DS2	R/W	H'0012 00F0	H'FCFF 7C30	32
Horizontal capture size register (SC1)	SC1_SCL0_DS3	R/W	H'00F4 05A0	H'FCFF 7C34	32
Horizontal scale down register (SC1)	SC1_SCL0_DS4	R/W	H'1000 2408	H'FCFF 7C38	32
Initial vertical phase register (SC1)	SC1_SCL0_DS5	R/W	H'1800 0000	H'FCFF 7C3C	32
Vertical scaling register (SC1)	SC1_SCL0_DS6	R/W	H'0000 07FC	H'FCFF 7C40	32
Scaling-down control block output size register (SC1)	SC1_SCL0_DS7	R/W	H'00F0 0280	H'FCFF 7C44	32
Scaling-up control register (SC1)	SC1_SCL0_US1	R/W	H'0000 0011	H'FCFF 7C48	32
Output image vertical size register (SC1)	SC1_SCL0_US2	R/W	H'0023 01E0	H'FCFF 7C4C	32
Output image horizontal size register (SC1)	SC1_SCL0_US3	R/W	H'0090 0280	H'FCFF 7C50	32
Scaling-up control block input size register (SC1)	SC1_SCL0_US4	R/W	H'00F0 0280	H'FCFF 7C54	32
Horizontal scale up register (SC1)	SC1_SCL0_US5	R/W	H'0000 2408	H'FCFF 7C58	32
Horizontal scale up initial phase register (SC1)	SC1_SCL0_US6	R/W	H'1000 0000	H'FCFF 7C5C	32
Trimming register (SC1)	SC1_SCL0_US7	R/W	H'0000 0000	H'FCFF 7C60	32
Frame buffer read select register (SC1)	SC1_SCL0_US8	R/W	H'0000 0000	H'FCFF 7C64	32
Background color register (SC1)	SC1_SCL0_OVR1	R/W	H'0080 0080	H'FCFF 7C6C	32
SCL1 register update control register (SC1)	SC1_SCL1_UPDATE	R/WC1	H'0000 0000	H'FCFF 7C80	32
Writing mode register (SC1)	SC1_SCL1_WR1	R/W	H'0000 0000	H'FCFF 7C88	32
Write address register 1T (SC1)	SC1_SCL1_WR2	R/W	H'0000 0000	H'FCFF 7C8C	32
Write address register 2T (SC1)	SC1_SCL1_WR3	R/W	H'0800 0001	H'FCFF 7C90	32
Write address register 3T (SC1)	SC1_SCL1_WR4	R/W	H'0008 0000	H'FCFF 7C94	32

**Table 33.41 Register Configuration of the Scaler (Channel 0)**

Name	Abbreviation	R/W	Initial Value	Address	Access Size
Frame sub-sampling register (SC1)	SC1_SCL1_WR5	R/W	H'0000 1000	H'FCFF 7C9C	32
Bit reduction register (SC1)	SC1_SCL1_WR6	R/W	H'0000 0000	H'FCFF 7CA0	32
Write detection register (SC1)	SC1_SCL1_WR7	R	H'0000 0000	H'FCFF 7CA4	32
Write address register 1B (SC1)	SC1_SCL1_WR8	R/W	H'0000 0000	H'FCFF 7CA8	32
Write address register 2B (SC1)	SC1_SCL1_WR9	R/W	H'0800 0001	H'FCFF 7CAC	32
Write address register 3B (SC1)	SC1_SCL1_WR10	R/W	H'0008 0000	H'FCFF 7CB0	32
Write detection register B (SC1)	SC1_SCL1_WR11	R	H'0000 0000	H'FCFF 7CB4	32
Status monitor 1 register (SC1)	SC1_SCL1_MON1	R	H'0000 0000	H'FCFF 7CB8	32
Pointer buffer 0 register (SC1)	SC1_SCL1_PBUF0	R	H'0000 0000	H'FCFF 7CBC	32
Pointer buffer 1 register (SC1)	SC1_SCL1_PBUF1	R	H'0000 0000	H'FCFF 7CC0	32
Pointer buffer 2 register (SC1)	SC1_SCL1_PBUF2	R	H'0000 0000	H'FCFF 7CC4	32
Pointer buffer 3 register (SC1)	SC1_SCL1_PBUF3	R	H'0000 0000	H'FCFF 7CC8	32
Pointer buffer and field information register (SC1)	SC1_SCL1_PBUF_FLD	R	H'0000 0000	H'FCFF 7CCC	32
Pointer buffer control register (SC1)	SC1_SCL1_PBUF_CNT	R/W	H'0000 0000	H'FCFF 7CD0	32
Graphics 1 register update control register	GR1_UPDATE	R/WC1	H'0000 0000	H'FCFF 7D00	32
Frame buffer read control register (graphics 1)	GR1_FLM_RD	R/W	H'0000 0000	H'FCFF 7D04	32
Frame buffer control register 1 (graphics 1)	GR1_FLM1	R/W	H'0000 0000	H'FCFF 7D08	32
Frame buffer control register 2 (graphics 1)	GR1_FLM2	R/W	H'0000 0000	H'FCFF 7D0C	32
Frame buffer control register 3 (graphics 1)	GR1_FLM3	R/W	H'0800 0001	H'FCFF 7D10	32
Frame buffer control register 4 (graphics 1)	GR1_FLM4	R/W	H'0008 0000	H'FCFF 7D14	32
Frame buffer control register 5 (graphics 1)	GR1_FLM5	R/W	H'0000 03FF	H'FCFF 7D18	32
Frame buffer control register 6 (graphics 1)	GR1_FLM6	R/W	H'8000 0000	H'FCFF 7D1C	32
Alpha blending control register 1 (graphics 1)	GR1_AB1	R/W	H'0000 0000	H'FCFF 7D20	32
Alpha blending control register 2 (graphics 1)	GR1_AB2	R/W	H'0000 0000	H'FCFF 7D24	32
Alpha blending control register 3 (graphics 1)	GR1_AB3	R/W	H'0000 0000	H'FCFF 7D28	32
Alpha blending control register 4 (graphics 1)	GR1_AB4	R/W	H'0000 0000	H'FCFF 7D2C	32
Alpha blending control register 5 (graphics 1)	GR1_AB5	R/W	H'0000 0000	H'FCFF 7D30	32
Alpha blending control register 6 (graphics 1)	GR1_AB6	R/W	H'0000 0000	H'FCFF 7D34	32
Alpha blending control register 7 (graphics 1)	GR1_AB7	R/W	H'00FF 0000	H'FCFF 7D38	32
Alpha blending control register 8 (graphics 1)	GR1_AB8	R/W	H'0000 0000	H'FCFF 7D3C	32
Alpha blending control register 9 (graphics 1)	GR1_AB9	R/W	H'0000 0000	H'FCFF 7D40	32
Alpha blending control register 10 (graphics 1)	GR1_AB10	R/W	H'0000 0000	H'FCFF 7D44	32



**Table 33.41 Register Configuration of the Scaler (Channel 0)**

Name	Abbreviation	R/W	Initial Value	Address	Access Size
Alpha blending control register 11 (graphics 1)	GR1_AB11	R/W	H'0000 0000	H'FCFF 7D48	32
Background color control register (graphics 1)	GR1_BASE	R/W	H'0000 8080	H'FCFF 7D4C	32
CLUT table control register (graphics 1)	GR1_CLUT	R/W	H'0000 0000	H'FCFF 7D50	32
Status monitor register (graphics 1)	GR1_MON	R	H'0000 0000	H'FCFF 7D54	32

**Table 33.42 Register Configuration of the Scaler (Channel 1)**

Name	Abbreviation	R/W	Initial Value	Address	Access Size
SCL0 register update control register (SC0)	SC0_SCL0_UPDATE	R/WC1	H'0000 0000	H'FCFF9500	32
Mask control register (SC0)	SC0_SCL0_FRC1	R/W	H'0AF0 0001	H'FCFF9504	32
Missing Vsync compensation control register (SC0)	SC0_SCL0_FRC2	R/W	H'0E10 0001	H'FCFF9508	32
Output sync select register (SC0)	SC0_SCL0_FRC3	R/W	H'0000 0001	H'FCFF950C	32
Free-running period control register (SC0)	SC0_SCL0_FRC4	R/W	H'020C 031F	H'FCFF9510	32
Output delay control register (SC0)	SC0_SCL0_FRC5	R/W	H'0000 0101	H'FCFF9514	32
Full-screen vertical size register (SC0)	SC0_SCL0_FRC6	R/W	H'0023 01E0	H'FCFF9518	32
Full-screen horizontal size register (SC0)	SC0_SCL0_FRC7	R/W	H'0090 0280	H'FCFF951C	32
Vsync detection register (SC0)	SC0_SCL0_FRC9	R	H'0000 0000	H'FCFF9524	32
Status monitor 0 register (SC0)	SC0_SCL0_MON0	R	H'0000	H'FCFF 9528	16
Interrupt control register (SC0)	SC0_SCL0_INT	R/W	H'0000	H'FCFF952A	16
Scaling-down control register (SC0)	SC0_SCL0_DS1	R/W	H'0000 0011	H'FCFF952C	32
Vertical capture size register (SC0)	SC0_SCL0_DS2	R/W	H'0012 00F0	H'FCFF9530	32
Horizontal capture size register (SC0)	SC0_SCL0_DS3	R/W	H'00F4 05A0	H'FCFF9534	32
Horizontal scale down register (SC0)	SC0_SCL0_DS4	R/W	H'1000 2408	H'FCFF9538	32
Initial vertical phase register (SC0)	SC0_SCL0_DS5	R/W	H'1800 0000	H'FCFF953C	32
Vertical scaling register (SC0)	SC0_SCL0_DS6	R/W	H'0000 07FC	H'FCFF9540	32
Scaling-down control block output size register (SC0)	SC0_SCL0_DS7	R/W	H'00F0 0280	H'FCFF9544	32
Scaling-up control register (SC0)	SC0_SCL0_US1	R/W	H'0000 0011	H'FCFF9548	32
Output image vertical size register (SC0)	SC0_SCL0_US2	R/W	H'0023 01E0	H'FCFF954C	32
Output image horizontal size register (SC0)	SC0_SCL0_US3	R/W	H'0090 0280	H'FCFF9550	32
Scaling-up control block input size register (SC0)	SC0_SCL0_US4	R/W	H'00F0 0280	H'FCFF9554	32
Horizontal scale up register (SC0)	SC0_SCL0_US5	R/W	H'0000 2408	H'FCFF9558	32
Horizontal scale up initial phase register (SC0)	SC0_SCL0_US6	R/W	H'1000 0000	H'FCFF955C	32
Trimming register (SC0)	SC0_SCL0_US7	R/W	H'0000 0000	H'FCFF9560	32
Frame buffer read select register (SC0)	SC0_SCL0_US8	R/W	H'0000 0000	H'FCFF9564	32
Background color register (SC0)	SC0_SCL0_OVR1	R/W	H'0080 0080	H'FCFF956C	32

**Table 33.42 Register Configuration of the Scaler (Channel 1)**

Name	Abbreviation	R/W	Initial Value	Address	Access Size
SCL1 register update control register (SC0)	SC0_SCL1_UPDATE	R/WC1	H'0000 0000	H'FCFF9580	32
Writing mode register (SC0)	SC0_SCL1_WR1	R/W	H'0000 0000	H'FCFF9588	32
Write address register 1T (SC0)	SC0_SCL1_WR2	R/W	H'0000 0000	H'FCFF958C	32
Write address register 2T (SC0)	SC0_SCL1_WR3	R/W	H'0800 0001	H'FCFF9590	32
Write address register 3T (SC0)	SC0_SCL1_WR4	R/W	H'0008 0000	H'FCFF9594	32
Frame sub-sampling register (SC0)	SC0_SCL1_WR5	R/W	H'0000 1000	H'FCFF959C	32
Bit reduction register (SC0)	SC0_SCL1_WR6	R/W	H'0000 0000	H'FCFF95A0	32
Write detection register (SC0)	SC0_SCL1_WR7	R	H'0000 0000	H'FCFF95A4	32
Write address register 1B (SC0)	SC0_SCL1_WR8	R/W	H'0000 0000	H'FCFF95A8	32
Write address register 2B (SC0)	SC0_SCL1_WR9	R/W	H'0800 0001	H'FCFF95AC	32
Write address register 3B (SC0)	SC0_SCL1_WR10	R/W	H'0008 0000	H'FCFF95B0	32
Write detection register B (SC0)	SC0_SCL1_WR11	R	H'0000 0000	H'FCFF95B4	32
Status monitor 1 register (SC0)	SC0_SCL1_MON1	R	H'0000 0000	H'FCFF95B8	32
Pointer buffer 0 register (SC0)	SC0_SCL1_PBUF0	R	H'0000 0000	H'FCFF95BC	32
Pointer buffer 1 register (SC0)	SC0_SCL1_PBUF1	R	H'0000 0000	H'FCFF95C0	32
Pointer buffer 2 register (SC0)	SC0_SCL1_PBUF2	R	H'0000 0000	H'FCFF95C4	32
Pointer buffer 3 register (SC0)	SC0_SCL1_PBUF3	R	H'0000 0000	H'FCFF95C8	32
Pointer buffer and field information register (SC0)	SC0_SCL1_PBUF_FLD	R	H'0000 0000	H'FCFF95CC	32
Pointer buffer control register (SC0)	SC0_SCL1_PBUF_CNT	R/W	H'0000 0000	H'FCFF95D0	32
Graphics 0 register update control register	GR0_UPDATE	R/WC1	H'0000 0000	H'FCFF9600	32
Frame buffer read control register (graphics 0)	GR0_FLM_RD	R/W	H'0000 0000	H'FCFF9604	32
Frame buffer control register 1 (graphics 0)	GR0_FLM1	R/W	H'0000 0000	H'FCFF9608	32
Frame buffer control register 2 (graphics 0)	GR0_FLM2	R/W	H'0000 0000	H'FCFF960C	32
Frame buffer control register 3 (graphics 0)	GR0_FLM3	R/W	H'0800 0001	H'FCFF9610	32
Frame buffer control register 4 (graphics 0)	GR0_FLM4	R/W	H'0008 0000	H'FCFF9614	32
Frame buffer control register 5 (graphics 0)	GR0_FLM5	R/W	H'0000 03FF	H'FCFF9618	32
Frame buffer control register 6 (graphics 0)	GR0_FLM6	R/W	H'8000 0000	H'FCFF961C	32
Alpha blending control register 1 (graphics 0)	GR0_AB1	R/W	H'0000 0000	H'FCFF9620	32
Alpha blending control register 2 (graphics 0)	GR0_AB2	R/W	H'0000 0000	H'FCFF9624	32
Alpha blending control register 3 (graphics 0)	GR0_AB3	R/W	H'0000 0000	H'FCFF9628	32
Alpha blending control register 7 (graphics 0)	GR0_AB7	R/W	H'00FF 0000	H'FCFF9638	32
Alpha blending control register 8 (graphics 0)	GR0_AB8	R/W	H'0000 0000	H'FCFF963C	32
Alpha blending control register 9 (graphics 0)	GR0_AB9	R/W	H'0000 0000	H'FCFF9640	32

**Table 33.42 Register Configuration of the Scaler (Channel 1)**

Name	Abbreviation	R/W	Initial Value	Address	Access Size
Alpha blending control register 10 (graphics 0)	GR0_AB10	R/W	H'0000 0000	H'FCFF9644	32
Alpha blending control register 11 (graphics 0)	GR0_AB11	R/W	H'0000 0000	H'FCFF9648	32
Background color control register (graphics 0)	GR0_BASE	R/W	H'0000 8080	H'FCFF964C	32
CLUT table control register (graphics 0)	GR0_CLUT	R/W	H'0000 0000	H'FCFF9650	32
SCL0 register update control register (SC1)	SC1_SCL0_UPDATE	R/WC1	H'0000 0000	H'FCFF9C00	32
Mask control register (SC1)	SC1_SCL0_FRC1	R/W	H'0AF0 0001	H'FCFF9C04	32
Missing Vsync compensation control register (SC1)	SC1_SCL0_FRC2	R/W	H'0E10 0001	H'FCFF9C08	32
Output sync select register (SC1)	SC1_SCL0_FRC3	R/W	H'0000 0001	H'FCFF9C0C	32
Free-running period control register (SC1)	SC1_SCL0_FRC4	R/W	H'020C 031F	H'FCFF9C10	32
Output delay control register (SC1)	SC1_SCL0_FRC5	R/W	H'0000 0101	H'FCFF9C14	32
Full-screen vertical size register (SC1)	SC1_SCL0_FRC6	R/W	H'0023 01E0	H'FCFF9C18	32
Full-screen horizontal size register (SC1)	SC1_SCL0_FRC7	R/W	H'0090 0280	H'FCFF9C1C	32
Vsync detection register (SC1)	SC1_SCL0_FRC9	R	H'0000 0000	H'FCFF9C24	32
Status monitor 0 register (SC1)	SC1_SCL0_MON0	R	H'0000	H'FCFF9C28	16
Interrupt control register (SC1)	SC1_SCL0_INT	R/W	H'0000	H'FCFF9C2A	16
Scaling-down control register (SC1)	SC1_SCL0_DS1	R/W	H'0000 0011	H'FCFF9C2C	32
Vertical capture size register (SC1)	SC1_SCL0_DS2	R/W	H'0012 00F0	H'FCFF9C30	32
Horizontal capture size register (SC1)	SC1_SCL0_DS3	R/W	H'00F4 05A0	H'FCFF9C34	32
Horizontal scale down register (SC1)	SC1_SCL0_DS4	R/W	H'1000 2408	H'FCFF9C38	32
Initial vertical phase register (SC1)	SC1_SCL0_DS5	R/W	H'1800 0000	H'FCFF9C3C	32
Vertical scaling register (SC1)	SC1_SCL0_DS6	R/W	H'0000 07FC	H'FCFF9C40	32
Scaling-down control block output size register (SC1)	SC1_SCL0_DS7	R/W	H'00F0 0280	H'FCFF9C44	32
Scaling-up control register (SC1)	SC1_SCL0_US1	R/W	H'0000 0011	H'FCFF9C48	32
Output image vertical size register (SC1)	SC1_SCL0_US2	R/W	H'0023 01E0	H'FCFF9C4C	32
Output image horizontal size register (SC1)	SC1_SCL0_US3	R/W	H'0090 0280	H'FCFF9C50	32
Scaling-up control block input size register (SC1)	SC1_SCL0_US4	R/W	H'00F0 0280	H'FCFF9C54	32
Horizontal scale up register (SC1)	SC1_SCL0_US5	R/W	H'0000 2408	H'FCFF9C58	32
Horizontal scale up initial phase register (SC1)	SC1_SCL0_US6	R/W	H'1000 0000	H'FCFF9C5C	32
Trimming register (SC1)	SC1_SCL0_US7	R/W	H'0000 0000	H'FCFF9C60	32
Frame buffer read select register (SC1)	SC1_SCL0_US8	R/W	H'0000 0000	H'FCFF9C64	32
Background color register (SC1)	SC1_SCL0_OVR1	R/W	H'0080 0080	H'FCFF9C6C	32
SCL1 register update control register (SC1)	SC1_SCL1_UPDATE	R/WC1	H'0000 0000	H'FCFF9C80	32
Writing mode register (SC1)	SC1_SCL1_WR1	R/W	H'0000 0000	H'FCFF9C88	32
Write address register 1T (SC1)	SC1_SCL1_WR2	R/W	H'0000 0000	H'FCFF9C8C	32
Write address register 2T (SC1)	SC1_SCL1_WR3	R/W	H'0800 0001	H'FCFF9C90	32
Write address register 3T (SC1)	SC1_SCL1_WR4	R/W	H'0008 0000	H'FCFF9C94	32

**Table 33.42 Register Configuration of the Scaler (Channel 1)**

Name	Abbreviation	R/W	Initial Value	Address	Access Size
Frame sub-sampling register (SC1)	SC1_SCL1_WR5	R/W	H'0000 1000	H'FCFF9C9C	32
Bit reduction register (SC1)	SC1_SCL1_WR6	R/W	H'0000 0000	H'FCFF9CA0	32
Write detection register (SC1)	SC1_SCL1_WR7	R	H'0000 0000	H'FCFF9CA4	32
Write address register 1B (SC1)	SC1_SCL1_WR8	R/W	H'0000 0000	H'FCFF9CA8	32
Write address register 2B (SC1)	SC1_SCL1_WR9	R/W	H'0800 0001	H'FCFF9CAC	32
Write address register 3B (SC1)	SC1_SCL1_WR10	R/W	H'0008 0000	H'FCFF9CB0	32
Write detection register B (SC1)	SC1_SCL1_WR11	R	H'0000 0000	H'FCFF9CB4	32
Status monitor 1 register (SC1)	SC1_SCL1_MON1	R	H'0000 0000	H'FCFF9CB8	32
Pointer buffer 0 register (SC1)	SC1_SCL1_PBUF0	R	H'0000 0000	H'FCFF9CBC	32
Pointer buffer 1 register (SC1)	SC1_SCL1_PBUF1	R	H'0000 0000	H'FCFF9CC0	32
Pointer buffer 2 register (SC1)	SC1_SCL1_PBUF2	R	H'0000 0000	H'FCFF9CC4	32
Pointer buffer 3 register (SC1)	SC1_SCL1_PBUF3	R	H'0000 0000	H'FCFF9CC8	32
Pointer buffer and field information register (SC1)	SC1_SCL1_PBUF_FLD	R	H'0000 0000	H'FCFF9CCC	32
Pointer buffer control register (SC1)	SC1_SCL1_PBUF_CNT	R/W	H'0000 0000	H'FCFF9CD0	32
Graphics 1 register update control register	GR1_UPDATE	R/WC1	H'0000 0000	H'FCFF9D00	32
Frame buffer read control register (graphics 1)	GR1_FLM_RD	R/W	H'0000 0000	H'FCFF9D04	32
Frame buffer control register 1 (graphics 1)	GR1_FLM1	R/W	H'0000 0000	H'FCFF9D08	32
Frame buffer control register 2 (graphics 1)	GR1_FLM2	R/W	H'0000 0000	H'FCFF9D0C	32
Frame buffer control register 3 (graphics 1)	GR1_FLM3	R/W	H'0800 0001	H'FCFF9D10	32
Frame buffer control register 4 (graphics 1)	GR1_FLM4	R/W	H'0008 0000	H'FCFF9D14	32
Frame buffer control register 5 (graphics 1)	GR1_FLM5	R/W	H'0000 03FF	H'FCFF9D18	32
Frame buffer control register 6 (graphics 1)	GR1_FLM6	R/W	H'8000 0000	H'FCFF9D1C	32
Alpha blending control register 1 (graphics 1)	GR1_AB1	R/W	H'0000 0000	H'FCFF9D20	32
Alpha blending control register 2 (graphics 1)	GR1_AB2	R/W	H'0000 0000	H'FCFF9D24	32
Alpha blending control register 3 (graphics 1)	GR1_AB3	R/W	H'0000 0000	H'FCFF9D28	32
Alpha blending control register 4 (graphics 1)	GR1_AB4	R/W	H'0000 0000	H'FCFF9D2C	32
Alpha blending control register 5 (graphics 1)	GR1_AB5	R/W	H'0000 0000	H'FCFF9D30	32
Alpha blending control register 6 (graphics 1)	GR1_AB6	R/W	H'0000 0000	H'FCFF9D34	32
Alpha blending control register 7 (graphics 1)	GR1_AB7	R/W	H'00FF 0000	H'FCFF9D38	32
Alpha blending control register 8 (graphics 1)	GR1_AB8	R/W	H'0000 0000	H'FCFF9D3C	32
Alpha blending control register 9 (graphics 1)	GR1_AB9	R/W	H'0000 0000	H'FCFF9D40	32
Alpha blending control register 10 (graphics 1)	GR1_AB10	R/W	H'0000 0000	H'FCFF9D44	32

**Table 33.42 Register Configuration of the Scaler (Channel 1)**

Name	Abbreviation	R/W	Initial Value	Address	Access Size
Alpha blending control register 11 (graphics 1)	GR1_AB11	R/W	H'0000 0000	H'FCFF9D48	32
Background color control register (graphics 1)	GR1_BASE	R/W	H'0000 8080	H'FCFF9D4C	32
CLUT table control register (graphics 1)	GR1_CLUT	R/W	H'0000 0000	H'FCFF9D50	32
Status monitor register (graphics 1)	GR1_MON	R	H'0000 0000	H'FCFF9D54	32

### 33.2.1 SCL0 Register Update Control Register (SC0\_SCL0\_UPDATE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	SC0_SCL0_VEN_D	SC0_SCL0_VEN_C	—	—	—	SC0_SCL0_UPDATE	—	—	—	SC0_SCL0_VEN_B	—	—	—	SC0_SCL0_VEN_A
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/WC1	R/WC1	R	R	R	R/WC1	R	R	R	R/WC1	R	R	R	R/WC1

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13	SC0_SCL0_VEN_D	0	R/WC1	Scaling-Up Control and Frame Buffer Read Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
12	SC0_SCL0_VEN_C	0	R/WC1	Scaling-Down Control and Frame Buffer Read Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	SC0_SCL0_UPDATE	0	R/WC1	SYNC Control Register Update 0: Registers are not updated. 1: Registers are updated.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	SC0_SCL0_VEN_B	0	R/WC1	Synchronization Control and Scaling-up Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SC0_SCL0_VEN_A	0	R/WC1	Scaling-Down Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.

### 33.2.2 Mask Control Register (SC0\_SCL0\_FRC1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SC0_RES_VMASK[15:0]															
Initial value:	0	0	0	0	1	0	1	0	1	1	1	1	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SC0_RES_VMASK_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	SC0_RES_VMASK [15:0]	2800	R/W	Repeated Vsync Signal Masking Period Sets the repeated Vsync signal masking period beginning at a Vsync signal in terms of 128 pixel-clock periods. Masking period [usec] = SC0_RES_VMASK × 128 ÷ pixel clock frequency [MHz]
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SC0_RES_VMASK_ON	1	R/W	Repeated Vsync Signal Masking Control 0: Repeated Vsync signal masking control is disabled. 1: Repeated Vsync signal masking control is enabled.

Note: This register is updated when the SC0\_SCL0\_UPDATE bit in the SC0\_SCL0 register update control register (SC0\_SCL0\_UPDATE) is 1.

### 33.2.3 Missing Vsync Compensation Control Register (SC0\_SCL0\_FRC2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SC0_RES_VLACK[15:0]															
Initial value:	0	0	0	0	1	1	1	0	0	0	0	1	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SC0_RES_VLACK_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	SC0_RES_VLACK [15:0]	3600	R/W	Missing-Sync Compensating Pulse Output Wait Time Sets the wait time before outputting a missing-sync compensating pulse after a Vsync signal. Wait time [usec] = SC0_RES_VLACK × 128 ÷ pixel clock frequency [MHz]
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SC0_RES_VLACK_ON	1	R/W	Missing Vsync Signal Compensation 0: Compensation of missing Vsync signals is disabled. 1: Compensation of missing Vsync signals is enabled.

Note: This register is updated when the SC0\_SCL0\_UPDATE bit in the SC0\_SCL0 register update control register (SC0\_SCL0\_UPDATE) is 1.

## 33.2.4 Output Sync Select Register (SC0\_SCL0\_FRC3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SC0_RES_	—	—	—	—	—	—	—	SC0_RES_
								VS_IN_								VS_SEL
								SEL								
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	SC0_RES_ VS_IN_SEL	0	R/W	Horizontal and Vertical Sync Signal Output and Full-Screen Enable Signal Select Be sure to clear this bit to 0 when cascaded connection is enabled (GR1_AB1.GR1_CUS_CON_ON = 1). 0: Horizontal and vertical sync signal output and full-screen enable signal from scaler 0 1: Horizontal and vertical sync signal output and full-screen enable signal from scaler 1
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SC0_RES_ VS_SEL	1	R/W	Vsync Signal Output Select 0: Externally input Vsync signal 1: Internally generated free-running Vsync signal

Note: This register is updated when the SC0\_SCL0\_UPDATE bit in the SC0\_SCL0 register update control register (SC0\_SCL0\_UPDATE) is 1.

## 33.2.5 Free-Running Period Control Register (SC0\_SCL0\_FRC4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	SC0_RES_FV[10:0]										
Initial value:	0	0	0	0	0	0	1	0	0	0	0	0	1	1	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SC0_RES_FH[10:0]										
Initial value:	0	0	0	0	0	0	1	1	0	0	0	1	1	1	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	SC0_RES_FV [10:0]	524	R/W	Free-Running Vsync Period Setting Free-running Vsync period = (SC0_RES_FV + 1) × horizontal period [usec]
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SC0_RES_FH [10:0]	799	R/W	Hsync Period Setting Hsync period [usec] = (SC0_RES_FH + 1) ÷ pixel clock frequency [MHz]

Note: This register is updated when the SC0\_SCL0\_UPDATE bit in the SC0\_SCL0 register update control register (SC0\_SCL0\_UPDATE) is 1.

## 33.2.6 Output Delay Control Register (SC0\_SCL0\_FRC5)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SC0_RES_FLD_DLY_SEL	SC0_RES_VSDLY[7:0]							
Initial value:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	SC0_RES_FLD_DLY_SEL	1	R/W	Field Determination Signal Delay Control 0: No delay 1: Delay of one vertical cycle
7 to 0	SC0_RES_VSDLY[7:0]	1	R/W	Vsync Signal Delay Control Adjusts the Vsync signal delay in the output Hsync period units. Vsync signal delay [usec]: SC0_RES_VSDLY × output Hsync period [usec]

Note: This register is updated when the SC0\_SCL0\_VEN\_B bit in the SC0\_SCL0 register update control register (SC0\_SCL0\_UPDATE) is 1.



## 33.2.7 Full-Screen Vertical Size Register (SC0\_SCL0\_FRC6)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	SC0_RES_F_VS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SC0_RES_F_VW[10:0]										
Initial value:	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	SC0_RES_F_VS[10:0]	35	R/W	Vertical Enable Signal Start Position for Full Screen. (VSYNC + V backporch lines) Note: The set value should be four or more (lines). SC0_RES_F_VS + SC0_RES_F_VW should be equal to or less than 2039 (lines).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SC0_RES_F_VW[10:0]	480	R/W	Vertical Enable Signal Width for Full Screen (lines) Note: SC0_RES_F_VS + SC0_RES_F_VW should be equal to or less than 2039 (lines).

Note: This register is updated when the SC0\_SCL0\_VEN\_B bit in the SC0\_SCL0 register update control register (SC0\_SCL0\_UPDATE) is 1.

## 33.2.8 Full-Screen Horizontal Size Register (SC0\_SCL0\_FRC7)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	SC0_RES_F_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SC0_RES_F_HW[10:0]										
Initial value:	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	SC0_RES_F_HS[10:0]	144	R/W	Horizontal Enable Signal Start Position for Full Screen. (HSYNC+H backporch pixel-clock cycles) Note: The set value should be 16 or more (clock cycles). SC0_RES_F_HS + SC0_RES_F_HW should be equal to or less than 2015 (clock cycles).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SC0_RES_F_HW[10:0]	640	R/W	Horizontal Enable Signal Width for Full Screen (pixel-clock cycles) Note: SC0_RES_F_HS + SC0_RES_F_HW should be equal to or less than 2015 (clock cycles).

Note: This register is updated when the SC0\_SCL0\_VEN\_B bit in the SC0\_SCL0 register update control register (SC0\_SCL0\_UPDATE) is 1.

## 33.2.9 Vsync Detection Register (SC0\_SCL0\_FRC9)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	SC0_RES_QVLOCK	—	—	—	SC0_RES_QVLACK
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	SC0_RES_QVLOCK	0	R	Locked Vsync Signal Detection Flag 1: No repeated or missing Vsync signal input has been detected for four or more vertical periods. 0: Repeated or missing Vsync signal input has been detected.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SC0_RES_QVLACK	0	R	Missing Vsync Signal Detection Flag 1: Missing Vsync signal input has been detected. 0: No missing Vsync signal input has been detected.

## 33.2.10 Status Monitor 0 Register (SC0\_SCL0\_MON0)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SC0_RES_LIN_STAT[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SC0_RES_LIN_STAT [10:0]	All 0	R	Current location of the image line input to the scaling-down control block.

### 33.2.11 Interrupt Control Register (SC0\_SCL0\_INT)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SC0_RES_LINE[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SC0_RES_LINE[10:0]	All 0	R/W	Setting of Interrupt on Image Line Input to Scaling-down Control Block When the location of the image line input to the scaling-down control block matches the SC0_RES_LINE setting, an interrupt signal is output.

Note: This register is updated when the SC0\_SCL0\_VEN\_A bit in the SC0\_SCL0 register update control register (SC0\_SCL0\_UPDATE) is 1.

### 33.2.12 Scaling-Down Control Register (SC0\_SCL0\_DS1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	SC0_RES_DS_V_ON	—	—	—	SC0_RES_DS_H_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	SC0_RES_DS_V_ON	1	R/W	Vertical Scale Down On/Off 0: Off 1: On
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SC0_RES_DS_H_ON	1	R/W	Horizontal Scale Down On/Off 0: Off 1: On

Note: This register is updated when the SC0\_SCL0\_VEN\_A bit in the SC0\_SCL0 register update control register (SC0\_SCL0\_UPDATE) is 1.

## 33.2.13 Vertical Capture Size Register (SC0\_SCL0\_DS2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	SC0_RES_VS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SC0_RES_VW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	SC0_RES_VS [10:0]	18	R/W	Vertical Position Setting for Video Signal Capturing (VSYNC + (V backporch - 1) lines) Note: The set value should be four or more (lines). SC0_RES_VS + SC0_RES_VW should be equal to or less than 2039 (lines).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SC0_RES_VW [10:0]	240	R/W	Vertical Width of Video Signal to be Captured (Lines) Note: SC0_RES_VS + SC0_RES_VW should be equal to or less than 2039 (lines).

Note: This register is updated when the SC0\_SCL0\_VEN\_A bit in the SC0\_SCL0 register update control register (SC0\_SCL0\_UPDATE) is 1.

## 33.2.14 Horizontal Capture Size Register (SC0\_SCL0\_DS3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	SC0_RES_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	1	1	1	1	0	1	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SC0_RES_HW[10:0]										
Initial value:	0	0	0	0	0	1	0	1	1	0	1	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	SC0_RES_HS [10:0]	244	R/W	Horizontal Position Setting for Video Signal Capturing (HSYNC + H backporch video-image clock cycles) Note: The set value should be 16 or more (clock cycles). SC0_RES_HS + SC0_RES_HW should be equal to or less than 2015 (clock cycles).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SC0_RES_HW [10:0]	1440	R/W	Horizontal Width of Video Signal to be Captured (Video-image clock cycles) Note: SC0_RES_HS + SC0_RES_HW should be equal to or less than 2015 (clock cycles).

Note: This register is updated when the SC0\_SCL0\_VEN\_A bit in the SC0\_SCL0 register update control register (SC0\_SCL0\_UPDATE) is 1.

## 33.2.15 Horizontal Scale Down Register (SC0\_SCL0\_DS4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	SC0_RES_PFIL_SEL	SC0_RES_DS_H_INTERPOTYP	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC0_RES_DS_H_RATIO[15:0]															
Initial value:	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
29	SC0_RES_PFIL_SEL	0	R/W	Prefilter Mode Select for Brightness Signals 0: The prefilter is turned off. 1: The prefilter is turned on. (1/4 + 1/2 + 1/4)
28	SC0_RES_DS_H_INTERPOTYP	1	R/W	Horizontal Interpolation Mode Select 0: Hold interpolation 1: Linear interpolation
27 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	SC0_RES_DS_H_RATIO[15:0]	9224	R/W	Horizontal Scale Down Ratio ([15:12]: Integer part, [11:0]: Decimal part) $\text{round}(\text{SC0\_RES\_HW} \div \text{SC0\_RES\_OUT\_HW} \times 4096)$ SC0_RES_DS_H_RATIO < 4096: Setting prohibited SC0_RES_DS_H_RATIO = 4096: 100% scale up SC0_RES_DS_H_RATIO > 4096: Scale down

Note: This register is updated when the SC0\_SCL0\_VEN\_A bit in the SC0\_SCL0 register update control register (SC0\_SCL0\_UPDATE) is 1.

## 33.2.16 Initial Vertical Phase Register (SC0\_SCL0\_DS5)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	SC0_RES_V_INTERPOTYP	SC0_RES_TOP_INIPHASE[11:0]											
Initial value:	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	SC0_RES_BTM_INIPHASE[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	SC0_RES_V_INTERPOTYP	1	R/W	Vertical Interpolation Mode Select 0: Hold interpolation 1: Linear interpolation
27 to 16	SC0_RES_TOP_INIPHASE [11:0]	2048	R/W	Vertical Interpolation Start Phase for Top Field 0 to 4095 (0 to approx. 1.0)
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 0	SC0_RES_BTM_INIPHASE [11:0]	0	R/W	Vertical Interpolation Start Phase for Bottom Field 0 to 4095 (0 to approx. 1.0)

Note: This register is updated when the SC0\_SCL0\_VEN\_A and SC0\_SCL0\_VEN\_B bits in the SC0\_SCL0 register update control register (SC0\_SCL0\_UPDATE) are 1.



## 33.2.17 Vertical Scaling Register (SC0\_SCL0\_DS6)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC0_RES_V_RATIO[15:0]															
Initial value:	0	0	0	0	0	1	1	1	1	1	1	1	1	1	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	SC0_RES_V_RATIO [15:0]	2044	R/W	Vertical Scale Up/Down Ratio ([15:12]: Integer part, [11:0]: Decimal part) For scale down: $\text{round}(\text{SC0\_RES\_VW} \div \text{SC0\_RES\_OUT\_VW} \times 4096)$ For scale up: $\text{round}(\text{SC0\_RES\_IN\_VW} \div \text{SC0\_RES\_P\_VW} \times 4096)$ SC0_RES_V_RATIO < 4096: Scale up SC0_RES_V_RATIO = 4096: 100% scale up SC0_RES_V_RATIO > 4096: Scale down

Note: These bits updated when the SC0\_SCL0\_VEN\_A and SC0\_SCL0\_VEN\_B bits in the SC0\_SCL0 register update control register (SC0\_SCL0\_UPDATE) are 1. Accordingly, even a scaled-up graphics display requires both an input Vsync signal and output Vsync signal.

## 33.2.18 Scaling-Down Control Block Output Size Register (SC0\_SCL0\_DS7)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	SC0_RES_OUT_VW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SC0_RES_OUT_HW[10:0]										
Initial value:	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	SC0_RES_OUT_VW [10:0]	240	R/W	Number of Valid Lines in Vertical Direction Output by Scaling-down Control Block (lines) This bit setting is used for the number of lines to be written to the frame buffer. When SC0_SCL1_WR1.SC0_RES_LOOP is 0 (frame write mode), specify the number of lines for one frame. When SC0_SCL1_WR1.SC0_RES_LOOP is 1 (line write mode), specify the number of lines for repeated write. Note: The SC0_RES_OUT_VW value should be aligned in 4-line units and equal to or smaller than the SC0_RES_VW value.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SC0_RES_OUT_HW [10:0]	640	R/W	Number of Valid Horizontal Pixels Output by Scaling-Down Control Block (video-image clock cycles) Note: The SC0_RES_OUT_HW value should be aligned in 4-pixel units and equal to or smaller than the SC0_RES_HW value.

Note: This register is updated when the SC0\_SCL0\_VEN\_A and SC0\_SCL0\_VEN\_C bits in the SC0\_SCL0 register update control register (SC0\_SCL0\_UPDATE) are 1.

## 33.2.19 Scaling-Up Control Register (SC0\_SCL0\_US1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	SC0 RES_US_ V_ON	—	—	—	SC0 RES_US_ H_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	SC0_RES_US_ V_ON	1	R/W	Vertical Scale Up On/Off 0: Off 1: On
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SC0_RES_US_ H_ON	1	R/W	Horizontal Scale Up On/Off 0: Off 1: On

Note: This register is updated when the SC0\_SCL0\_VEN\_B bit in the SC0\_SCL0 register update control register (SC0\_SCL0\_UPDATE) is 1.

## 33.2.20 Output Image Vertical Size Register (SC0\_SCL0\_US2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	SC0_RES_P_VS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SC0_RES_P_VW[10:0]										
Initial value:	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	SC0_RES_P_VS[10:0]	35	R/W	Vertical Enable Signal Start Position for Output Image (VSYNC + V backporch lines) Note: The set value should be four or more (lines). SC0_RES_P_VS + SC0_RES_P_VW should be equal to or less than 2039 (lines).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SC0_RES_P_VW[10:0]	480	R/W	Vertical Enable Signal Width for Output Image (lines) Note: SC0_RES_P_VS + SC0_RES_P_VW should be equal to or less than 2039 (lines).

Note: This register is updated when the SC0\_SCL0\_VEN\_B bit in the SC0\_SCL0 register update control register (SC0\_SCL0\_UPDATE) is 1.

## 33.2.21 Output Image Horizontal Size Register (SC0\_SCL0\_US3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	SC0_RES_P_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SC0_RES_P_HW[10:0]										
Initial value:	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	SC0_RES_P_HS[10:0]	144	R/W	Horizontal Enable Signal Start Position for Output Image (HSYNC+H backporch pixel-clock cycles) Note: The set value should be 16 or more (clock cycles). SC0_RES_P_HS + SC0_RES_P_HW should be equal to or less than 2015 (clock cycles).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SC0_RES_P_HW[10:0]	640	R/W	Horizontal Enable Signal Width for Output Image (pixel-clock cycles) Note: SC0_RES_P_HS + SC0_RES_P_HW should be equal to or less than 2015 (clock cycles).

Note: This register is updated when the SC0\_SCL0\_VEN\_B bit in the SC0\_SCL0 register update control register (SC0\_SCL0\_UPDATE) is 1.

## 33.2.22 Scaling-Up Control Block Input Size Register (SC0\_SCL0\_US4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	SC0_RES_IN_VW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SC0_RES_IN_HW[10:0]										
Initial value:	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	SC0_RES_IN_VW[10:0]	240	R/W	Number of Valid Lines in Vertical Direction Input to Scaling-down Control Block (lines)
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SC0_RES_IN_HW[10:0]	640	R/W	Number of Valid Horizontal Pixels Input to Scaling-down Control Block (pixel-clock cycles)

Note: This register is updated when the SC0\_SCL0\_VEN\_B and SC0\_SCL0\_VEN\_D bits in the SC0\_SCL0 register update control register (SC0\_SCL0\_UPDATE) are 1.

## 33.2.23 Horizontal Scale Up Register (SC0\_SCL0\_US5)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC0_RES_US_H_RATIO[15:0]															
Initial value:	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	SC0_RES_US_H_RATIO[15:0]	9224	R/W	Horizontal Scale Up Ratio ([15:12]: Integer part, [11:0]: Decimal part) round (SC0_RES_IN_HW ÷ SC0_RES_P_HW × 4096) SC0_RES_US_H_RATIO < 4096: Scale up SC0_RES_US_H_RATIO = 4096: 100% scale up SC0_RES_US_H_RATIO > 4096: Setting prohibited

Note: This register is updated when the SC0\_SCL0\_VEN\_B bit in the SC0\_SCL0 register update control register (SC0\_SCL0\_UPDATE) is 1.

## 33.2.24 Horizontal Scale Up Initial Phase Register (SC0\_SCL0\_US6)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	SC0_RES_US_H_INTERPOTYP	SC0_RES_US_HT_INIPHASE[11:0]											
Initial value:	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	SC0_RES_US_HB_INIPHASE[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	SC0_RES_US_H_INTERPOTYP	1	R/W	Horizontal Interpolation Mode Select 0: Hold interpolation 1: Linear interpolation
27 to 16	SC0_RES_US_HT_INIPHASE[11:0]	0	R/W	Horizontal Interpolation Start Phase for Top Field 0 to 4095 (0 to approx. 1.0)
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 0	SC0_RES_US_HB_INIPHASE[11:0]	0	R/W	Horizontal Interpolation Start Phase for Bottom Field 0 to 4095 (0 to approx. 1.0)

Note: This register is updated when the SC0\_SCL0\_VEN\_B bit in the SC0\_SCL0 register update control register (SC0\_SCL0\_UPDATE) is 1.

## 33.2.25 Trimming Register (SC0\_SCL0\_US7)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC0_RES_HCUT[7:0]							SC0_RES_V CUT[7:0]								
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 8	SC0_RES_HCUT[7:0]	0	R/W	Horizontal Amount of Cut-off Post-Scaling Image (Right and Left Parts) Sets the number of pixel-clock cycles.
7 to 0	SC0_RES_V CUT[7:0]	0	R/W	Vertical Amount of Cut-off Post-Scaling Image (Upper and Lower Parts) Sets the number of lines.

Note: This register is updated when the SC0\_SCL0\_VEN\_B bit in the SC0\_SCL0 register update control register (SC0\_SCL0\_UPDATE) is 1.

## 33.2.26 Frame Buffer Read Select Register (SC0\_SCL0\_US8)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	SC0_RES_IBUS_SYNC_SEL	—	—	—	SC0_RES_DISP_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	SC0_RES_IBUS_SYNC_SEL	0	R/W	Sync Signal Select for Frame Buffer Read Block 0: Sync signals from the scaling-up control block 1: Sync signals from the graphics processing block
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SC0_RES_DISP_ON	0	R/W	Post-Scaling Image Frame Display On/Off 0: Frame display on 1: Frame display off

Note: SC0\_RES\_IBUS\_SYNC\_SEL is updated when the SC0\_SCL0\_VEN\_B and SC0\_SCL0\_VEN\_D bits in the SC0\_SCL0 register update control register (SC0\_SCL0\_UPDATE) are 1.

SC0\_RES\_DISP\_ON is updated when the SC0\_SCL0\_VEN\_B bit in the SC0\_SCL0 register update control register (SC0\_SCL0\_UPDATE) is 1.

## 33.2.27 Background Color Register (SC0\_SCL0\_OVR1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	SC0_RES_BK_COL_R[7:0]							
Initial value:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC0_RES_BK_COL_G[7:0]								SC0_RES_BK_COL_B[7:0]							
Initial value:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	SC0_RES_BK_CLO_R[7:0]	128	R/W	Background Color Setting R/Cr Signal R: 8 bits; unsigned (0 to 255 [LSB]) Cr: 8 bits; 128 offset binary; unsigned (0 to 255 [LSB])
15 to 8	SC0_RES_BK_COL_G[7:0]	0	R/W	Background Color Setting G/Y Signal G/Y: 8 bits; unsigned (0 to 255 [LSB])



Bit	Bit Name	Initial Value	R/W	Description
7 to 0	SC0_RES_BK_COL_B[7:0]	128	R/W	Background Color Setting B/Cb Signal B: 8 bits; unsigned (0 to 255 [LSB]) Cb: 8 bits; 128 offset binary; unsigned (0 to 255 [LSB])

Note: This register is updated when the SC0\_SCL0\_VEN\_B bit in the SC0\_SCL0 register update control register (SC0\_SCL0\_UPDATE) is 1.

### 33.2.28 SCL1 Register Update Control Register (SC0\_SCL1\_UPDATE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	SC0_SCL1_UPDATE_B	—	—	—	SC0_SCL1_UPDATE_A
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/WC1	R	R	R	R/WC1

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	SC0_SCL1_VEN_B	—	—	—	SC0_SCL1_VEN_A
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/WC1	R	R	R	R/WC1

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	SC0_SCL1_UPDATE_B	0	R/WC1	Frame Buffer Write Control Register Update 0: Registers are not updated. 1: Registers are updated.
19 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	SC0_SCL1_UPDATE_A	0	R/WC1	Frame Buffer Write Control Register Update 0: Registers are not updated. 1: Registers are updated.
15 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	SC0_SCL1_VEN_B	0	R/WC1	Frame Buffer Write Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SC0_SCL1_VEN_A	0	R/WC1	Frame Buffer Write Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.

## 33.2.29 Writing Mode Register (SC0\_SCL1\_WR1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	SC0_RES_WRSWA [2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	SC0_RES_TB_ADD_MOD	SC0_RES_DS_WR_MD[2:0]			SC0_RES_MD[1:0]	SC0_RES_LOOP	SC0_RES_BST_MD	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
18 to 16	SC0_RES_WRSWA[2:0]	All 0	R/W	8-Bit, 16-Bit, or 32-Bit Swap Setting These bits control swapping in frame buffer writing as follows. Bit 0 0: Swapped in 8-bit units. 1: Not swapped in 8-bit units. Bit 1 0: Swapped in 16-bit units. 1: Not swapped in 16-bit units. Bit 2 0: Swapped in 32-bit units. 1: Not swapped in 32-bit units. According to the setting of these bits, data is swapped as follows. Each number in parentheses ((1) to (8)) indicates 8-bit data. 000: (1) (2) (3) (4) (5) (6) (7) (8) [Not swapped] 001: (2) (1) (4) (3) (6) (5) (8) (7) [Swapped in 8-bit units] 010: (3) (4) (1) (2) (7) (8) (5) (6) [Swapped in 16-bit units] 011: (4) (3) (2) (1) (8) (7) (6) (5) [Swapped in 16-bit units + 8-bit units] 100: (5) (6) (7) (8) (1) (2) (3) (4) [Swapped in 32-bit units] 101: (6) (5) (8) (7) (2) (1) (4) (3) [Swapped in 32-bit units + 8-bit units] 110: (7) (8) (5) (6) (3) (4) (1) (2) [Swapped in 32-bit units + 16-bit units] 111: (8) (7) (6) (5) (4) (3) (2) (1) [Swapped in 32-bit units + 16-bit units + 8-bit units] Note: When YCbCr422 or RGB565 is selected as a frame buffer video-signal writing format, these bits should be set to 000 [Not swapped].
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	SC0_RES_TB_ADD_MOD	0	R/W	Top and Bottom Data Write Address Specification Method 0: A write address is specified in common for top and bottom data. 1: Separate write addresses are specified for top and bottom data.
6 to 4	SC0_RES_DS_WR_MD[2:0]	0	R/W	Frame Buffer Writing Mode for Image Processing 0: Normal 1: Horizontal mirroring 2: 90° rotation 3: 180° rotation 4: 270° rotation 5 to 7: Setting prohibited

Bit	Bit Name	Initial Value	R/W	Description
3, 2	SC0_RES_MD [1:0]	0	R/W	Frame Buffer Video-Signal Writing Format 0: YCbCr422 (16 bits) 1: RGB565 (16 bits) 2: RGB888 (24 (32) bits) 3: YCbCr444 (24 (32) bits)
1	SC0_RES_LOOP	0	R/W	Frame Buffer Write Mode Select 0: Frame mode 1: Line mode (read as ring buffer)
0	SC0_RES_BST_MD	0	R/W	Transfer Burst Length for Frame Buffer Writing 0: 32-byte transfer (4 bursts) 1: 128-byte transfer (16 bursts)

Note: SC0\_RES\_LOOP and SC0\_RES\_BST\_MD are updated when the SC0\_SCL1\_VEN\_B bit in the SC0\_SCL1 register update control register (SC0\_SCL1\_UPDATE) is 1.

SC0\_RES\_TB\_ADD\_MOD, SC0\_RES\_DS\_WR\_MD, and SC0\_RES\_MD are updated when the SC0\_SCL1\_VEN\_A and SC0\_SCL1\_VEN\_B bits in the SC0\_SCL1 register update control register (SC0\_SCL1\_UPDATE) are 1.

SC0\_RES\_WRSWA is updated when the SC0\_SCL1\_UPDATE\_A bit in the SC0\_SCL1 register update control register (SC0\_SCL1\_UPDATE) is 1.

### 33.2.30 Write Address Register 1T (SC0\_SCL1\_WR2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SC0_RES_BASE[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC0_RES_BASE[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SC0_RES_BASE [31:0]	0	R/W	Frame Buffer Base Address Sets the start address of the frame buffer to store the frame data for the top field when SC0_SCL1_WR1.SC0_RES_TB_ADD_MOD = 1 or that for the top and bottom fields when SC0_SCL1_WR1.SC0_RES_TB_ADD_MOD = 0. For 32-byte transfer: The lower five bits should be fixed to 0 0000. For 128-byte transfer: The lower seven bits should be fixed to 000 0000.

Note: This register is updated when the SC0\_SCL1\_VEN\_B bit in the SC0\_SCL1 register update control register (SC0\_SCL1\_UPDATE) is 1.

## 33.2.31 Write Address Register 2T (SC0\_SCL1\_WR3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	SC0_RES_LN_OFF[14:0]														
Initial value:	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SC0_RES_FLM_NUM[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30 to 16	SC0_RES_LN_OFF [14:0]	2048	R/W	Frame Buffer Line Offset Address Sets the line offset address for calculating the line start address for the top field when SC0_SCL1_WR1.SC0_RES_TB_ADD_MOD = 1 or that for the top and bottom fields when SC0_SCL1_WR1.SC0_RES_TB_ADD_MOD = 0. Line 0: SC0_RES_BASE Line 1: SC0_RES_BASE + SC0_RES_LN_OFF × 1 : Line n: SC0_RES_BASE + SC0_RES_LN_OFF × n For 32-byte transfer: The lower five bits should be fixed to 0 0000. For 128-byte transfer: The lower seven bits should be fixed to 000 0000.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	SC0_RES_FLM_NUM [9:0]	1	R/W	Number of Frames of Buffer to be Written to Sets the number of frames for the top field when SC0_SCL1_WR1.SC0_RES_TB_ADD_MOD = 1 or that for the top and bottom fields when SC0_SCL1_WR1.SC0_RES_TB_ADD_MOD = 0 Number of frames defined by SC0_RES_FLM_NUM + 1 are used.

Note: This register is updated when the SC0\_SCL1\_VEN\_B bit in the SC0\_SCL1 register update control register (SC0\_SCL1\_UPDATE) is 1.

## 33.2.32 Write Address Register 3T (SC0\_SCL1\_WR4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	SC0_RES_FLM_OFF[22:16]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC0_RES_FLM_OFF[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
22 to 0	SC0_RES_FLM_OFF[22:0]	524288	R/W	Frame Buffer Frame Offset Address Sets the frame offset address for calculating the start address of each frame for the top field when SC0_SCL1_WR1.SC0_RES_TB_ADD_MOD = 1 or that for the top and bottom fields when SC0_SCL1_WR1.SC0_RES_TB_ADD_MOD = 0. Buffer 0: SC0_RES_BASE Buffer 1: SC0_RES_BASE + SC0_RES_FLM_OFF × 1 : Buffer n: SC0_RES_BASE + SC0_RES_FLM_OFF × n For 32-byte transfer: The lower five bits should be fixed to 0 0000. For 128-byte transfer: The lower seven bits should be fixed to 000 0000.

Note: This register is updated when the SC0\_SCL1\_VEN\_B bit in the SC0\_SCL1 register update control register (SC0\_SCL1\_UPDATE) is 1.

## 33.2.33 Frame Sub-Sampling Register (SC0\_SCL1\_WR5)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	SC0_RES_INTER	—	—	SC0_RES_FS_RATE[1:0]	—	—	—	SC0_RES_FLD_SEL	—	—	—	SC0_RES_WENB	
Initial value:	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R/W	R/W	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	SC0_RES_INTER	1	R/W	Field Operating Mode Select 0: Progressive 1: Interlace
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	SC0_RES_FS_RATE[1:0]	0	R/W	Writing Rate Sets the frame buffer writing rate to the vertical frequency of the input signal. 0: 1/1 an input signal (The SC0_RES_FLD_SEL setting is invalid.) 1: 1/2 an input signal 2: 1/4 an input signal 3: 1/8 an input signal
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	SC0_RES_FLD_SEL	0	R/W	Write Field Select 0: Top field 1: Bottom field
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SC0_RES_WENB	0	R/W	Frame Buffer Write Enable After making the setting to enable writing, writing starts from the second frame. 0: Frame buffer writing is disabled. 1: Frame buffer writing is enabled.

Note: SC0\_RES\_INTER, SC0\_RES\_FS\_RATE[1:0], and SC0\_RES\_FLD\_SEL are updated when the SC0\_SCL1\_VEN\_A bit in the SC0\_SCL1 register update control register (SC0\_SCL1\_UPDATE) is 1. SC0\_RES\_WENB is updated when the SC0\_SCL1\_VEN\_A and SC0\_SCL1\_VEN\_B bits in the SC0\_SCL1 register update control register (SC0\_SCL1\_UPDATE) are 1.

## 33.2.34 Bit Reduction Register (SC0\_SCL1\_WR6)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	SC0_RES_DTH_ON	—	—	—	SC0_RES_BITDEC_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	SC0_RES_DTH_ON	0	R/W	Dither Correction On/Off 0: Off (rounded off) 1: On (2 × 2 dither pattern)
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SC0_RES_BITDEC_ON	0	R/W	Bit Reduction On/Off 0: Off 1: On

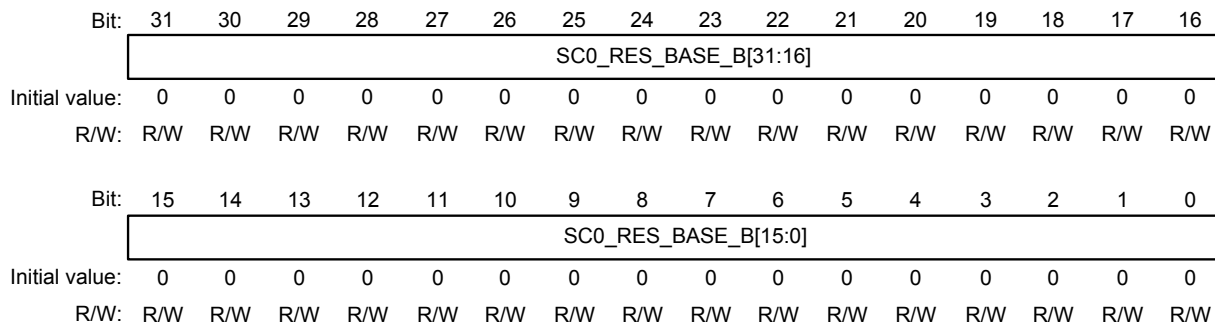
Note: This register is updated when the SC0\_SCL1\_VEN\_A bit in the SC0\_SCL1 register update control register (SC0\_SCL1\_UPDATE) is 1.

## 33.2.35 Write Detection Register (SC0\_SCL1\_WR7)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SC0_RES_OVERFLOW	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	—	—	—	—	—	—	SC0_RES_FLM_CNT[9:0]									—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	SC0_RES_OVERFLOW	0	R	Line Buffer Overflow Detect 1: Line buffer has overflowed. 0: Line buffer has not overflowed.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	SC0_RES_FLM_CNT [9:0]	0	R	Frame Number Before Frame Being Accessed Frame number before the frame being accessed in the top field when SC0_SCL1_WR1.SC0_RES_TB_ADD_MOD = 1 or that in the top or bottom field when SC0_SCL1_WR1.SC0_RES_TB_ADD_MOD = 0.

33.2.36 Write Address Register 1B (SC0\_SCL1\_WR8)



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SC0_RES_BASE_B [31:0]	0	R/W	Frame Buffer Base Address for Bottom Sets the start address of the frame buffer to store the frame data for the bottom field when SC0_SCL1_WR1.SC0_RES_TB_ADD_MOD = 1. For 32-byte transfer: The lower five bits should be fixed to 0_0000. For 128-byte transfer: The lower seven bits should be fixed to 000_0000.

Note: This register is updated when the SC0\_SCL1\_VEN\_B bit in the SC0\_SCL1 register update control register (SC0\_SCL1\_UPDATE) is 1.



## 33.2.37 Write Address Register 2B (SC0\_SCL1\_WR9)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	SC0_RES_LN_OFF_B[14:0]														
Initial value:	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SC0_RES_FLM_NUM_B[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30 to 16	SC0_RES_LN_OFF_B [14:0]	2048	R/W	Frame Buffer Line Offset Address for Bottom Sets the line offset address for calculating the line start address for the bottom field when SC0_SCL1_WR1.SC0_RES_TB_ADD_MOD = 1. Line 0: SC0_RES_BASE_B Line 1: SC0_RES_BASE_B + SC0_RES_LN_OFF_B × 1 : Line n: SC0_RES_BASE + SC0_RES_LN_OFF_B × n For 32-byte transfer: The lower five bits should be fixed to 0_0000. For 128-byte transfer: The lower seven bits should be fixed to 000_0000.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	SC0_RES_FLM_NUM_B [9:0]	1	R/W	Number of Frames of Buffer to be Written to for Bottom Field Number of frames defined by SC0_RES_FLM_NUM_B + 1 are used when SC0_SCL1_WR1.SC0_RES_TB_ADD_MOD = 1.

Note: This register is updated when the SC0\_SCL1\_VEN\_B bit in the SC0\_SCL1 register update control register (SC0\_SCL1\_UPDATE) is 1.

## 33.2.38 Write Address Register 3B (SC0\_SCL1\_WR10)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	SC0_RES_FLM_OFF_B[22:16]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC0_RES_FLM_OFF_B[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
22 to 0	SC0_RES_FLM_OFF_B[22:0]	524288	R/W	Frame Buffer Frame Offset Address for Bottom Sets the frame offset address for calculating the start address of each frame for the bottom field when SC0_SCL1_WR1.SC0_RES_TB_ADD_MOD = 1. Buffer 0: SC0_RES_BASE_B Buffer 1: SC0_RES_BASE_B + SC0_RES_FLM_OFF_B × 1 : Buffer n: SC0_RES_BASE_B + SC0_RES_FLM_OFF_B × n For 32-byte transfer: The lower five bits should be fixed to 0_0000. For 128-byte transfer: The lower seven bits should be fixed to 000_0000.

Note: This register is updated when the SC0\_SCL1\_VEN\_B bit in the SC0\_SCL1 register update control register (SC0\_SCL1\_UPDATE) is 1.

## 33.2.39 Write Detection Register B (SC0\_SCL1\_WR11)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SC0_RES_FLM_CNT_B[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	SC0_RES_FLM_CNT_B[9:0]	0	R	Frame Number Before Frame Being Accessed in Bottom Field Frame number before the frame being accessed in the bottom field when SC0_SCL1_WR1.SC0_RES_TB_ADD_MOD = 1.

## 33.2.40 Status Monitor 1 Register (SC0\_SCL1\_MON1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SC0_PBUF_NUM [1:0]	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	SC0_PBUF_NUM[1:0]	All 0	R	Write pointer indicating the pointer buffer number corresponding to the location currently being written to.
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

## 33.2.41 Pointer Buffer 0 Register (SC0\_SCL1\_PBUF0)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SC0_PBUF0_ADD[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC0_PBUF0_ADD[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SC0_PBUF0_ADD[31:0]	All 0	R	Start address of the write buffer pointed to by pointer buffer 0.

## 33.2.42 Pointer Buffer 1 Register (SC0\_SCL1\_PBUF1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SC0_PBUF1_ADD[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC0_PBUF1_ADD[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SC0_PBUF1_ADD[31:0]	All 0	R	Start address of the write buffer pointed to by pointer buffer 1.

## 33.2.43 Pointer Buffer 2 Register (SC0\_SCL1\_PBUF2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SC0_PBUF2_ADD[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC0_PBUF2_ADD[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SC0_PBUF2_ADD[31:0]	All 0	R	Start address of the write buffer pointed to by pointer buffer 2.

## 33.2.44 Pointer Buffer 3 Register (SC0\_SCL1\_PBUF3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SC0_PBUF3_ADD[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC0_PBUF3_ADD[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SC0_PBUF3_ADD[31:0]	All 0	R	Start address of the write buffer pointed to by pointer buffer 3.

## 33.2.45 Pointer Buffer and Field Information Register (SC0\_SCL1\_PBUF\_FLD)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	SC0_FLD_INF3	—	—	—	—	—	—	—	SC0_FLD_INF2
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SC0_FLD_INF1	—	—	—	—	—	—	—	SC0_FLD_INF0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	SC0_FLD_INF3	0	R	Top or bottom field information of the frame buffer pointed to by pointer buffer 3. 0: Bottom 1: Top
23 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	SC0_FLD_INF2	0	R	Top or bottom field information of the frame buffer pointed to by pointer buffer 2. 0: Bottom 1: Top
15 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	SC0_FLD_INF1	0	R	Top or bottom field information of the frame buffer pointed to by pointer buffer 1. 0: Bottom 1: Top
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
0	SC0_FLD_INF0	0	R	Top or bottom field information of the frame buffer pointed to by pointer buffer 0. 0: Bottom 1: Top

### 33.2.46 Pointer Buffer Control Register (SC0\_SCL1\_PBUF\_CNT)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SC0_PBUF_RST
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	SC0_PBUF_RST	0	R/W	Reset Control for Pointer Buffer 0: Pointer buffer is not reset. 1: Pointer buffer is reset.
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note: This register is updated when the SC0\_SCL1\_UPDATE\_B bit in the SC0\_SCL1 register update control register (SC0\_SCL1\_UPDATE) is 1.

## 33.2.47 Graphics 0 Register Update Control Register (GR0\_UPDATE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	GR0_UPDATE	—	—	—	GR0_P_VEN	—	—	—	GR0_IBUS_VEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/WC1	R	R	R	R/WC1	R	R	R	R/WC1

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	GR0_UPDATE	0	R/WC1	Frame Buffer Read Control Register Update 0: Registers are not updated. 1: Registers are updated.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	GR0_P_VEN	0	R/WC1	Graphics Display Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR0_IBUS_VEN	0	R/WC1	Frame Buffer Read Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.

## 33.2.48 Frame Buffer Read Control Register (Graphics 0) (GR0\_FLM\_RD)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR0_R_ENB
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR0_R_ENB	0	R/W	Frame Buffer Read Enable 0: Frame buffer reading is disabled. 1: Frame buffer reading is enabled.

Note: This register is updated when the GR0\_IBUS\_VEN bit in the graphics 0 register update control register (GR0\_UPDATE) is 1.

## 33.2.49 Frame Buffer Control Register 1 (Graphics 0) (GR0\_FLM1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR0_FLD_SEL	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR0_LN_OFF_DIR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	GR0_FLM_SEL[1:0]	—	—	—	GR0_IMR_FLM_INV	—	—	—	—	GR0_BST_MD
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	GR0_FLD_SEL	0	R/W	Enables or disables top or bottom field selection for the next frame buffer to be read. 0: Field selection is disabled. 1: Top or bottom field can be selected.
30 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	GR0_LN_OFF_DIR	0	R/W	Selects the line offset address direction of the frame buffer. 0: Increments the address by the line offset address. 1: Decrements the address by the line offset address.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	GR0_FLM_SEL [1:0]	0	R/W	Selects a frame buffer address setting signal. 0: Links to scaling-down process. (This setting is prohibited when separate write addresses are specified for the top and bottom fields; that is, SC0_RES_TB_ADD_MOD = 1 in SC0_SCL1_WR1.) 1: Selects GR0_FLM_NUM. 2: Links to distortion correction. 3: Links to pointer buffer.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	GR0_IMR_FLM_INV	0	R/W	Sets the frame buffer number for distortion correction. 0: Does not replace the numbers of the frames to be read. 1: Replaces the numbers of the frames to be read.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR0_BST_MD	0	R/W	Frame Buffer Burst Transfer Mode 0: 32-byte transfer 1: 128-byte transfer

Note: GR0\_FLD\_SEL is updated when the GR0\_P\_VEN bit in the graphics 0 register update control register (GR0\_UPDATE) is 1.  
GR0\_LN\_OFF\_DIR and GR0\_IMR\_FLM\_INV are updated when the GR0\_IBUS\_VEN bit in the graphics 0 register update control register (GR0\_UPDATE) is 1.  
GR0\_FLM\_SEL is updated when the GR0\_P\_VEN and GR0\_IBUS\_VEN bits in the graphics 0 register update control register (GR0\_UPDATE) are 1.  
GR0\_BST\_MD is updated when the GR0\_IBUS\_VEN and GR0\_P\_VEN bits in the graphics 0 register update control register (GR0\_UPDATE) are 1.



## 33.2.50 Frame Buffer Control Register 2 (Graphics 0) (GR0\_FLM2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR0_BASE[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR0_BASE[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	GR0_BASE [31:0]	0	R/W	Frame Buffer Base Address Sets the start address of the frame buffer where frame data is to be stored. GR0_BASE[4:3] and GR0_BASE[6:3] are referred to during 32-byte burst transfer and 128-byte burst transfer, respectively, to skip the start line data. The lower three bits should be fixed to 000.

Note: This register is updated when the GR0\_IBUS\_VEN and GR0\_P\_VEN bits in the graphics 0 register update control register (GR0\_UPDATE) are 1.

## 33.2.51 Frame Buffer Control Register 3 (Graphics 0) (GR0\_FLM3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR0_FLD_NXT	GR0_LN_OFF[14:0]														
Initial value:	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—						GR0_FLM_NUM[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	GR0_FLD_NXT	0	R/W	Top or Bottom Field Selection for Next Frame Buffer 0: Bottom 1: Top
30 to 16	GR0_LN_OFF[14:0]	2048	R/W	Frame Buffer Line Offset Address Sets the line offset address for calculating the start address of each line. Line 0: GR0_BASE Line 1: GR0_BASE + GR0_LN_OFF × 1 : Line n: GR0_BASE + GR0_LN_OFF × n For 32-byte transfer: The lower five bits should be fixed to 0 0000. For 128-byte transfer: The lower seven bits should be fixed to 000 0000.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	GR0_FLM_NUM[9:0]	1	R/W	Frame Number of Frame Buffer Manually set the frame number when GR0_FLM_SEL = 1.

Note: GR0\_FLD\_NXT is updated when the GR0\_P\_VEN bit in the graphics 0 register update control register (GR0\_UPDATE) is 1.

GR0\_LN\_OFF[14:0] and GR0\_FLM\_NUM[9:0] are updated when the GR0\_IBUS\_VEN bit in the graphics 0 register update control register (GR0\_UPDATE) is 1.

## 33.2.52 Frame Buffer Control Register 4 (Graphics 0) (GR0\_FLM4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	GR0_FLM_OFF[22:16]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR0_FLM_OFF[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
22 to 0	GR0_FLM_OFF[22:0]	524288	R/W	Frame Buffer Frame Offset Address Specifies the frame offset address used for calculating the start address of each frame buffer when more than one buffer is used. Buffer 0: GR0_BASE Buffer 1: GR0_BASE + GR0_FLM_OFF × 1 : Buffer n: GR0_BASE + GR0_FLM_OFF × n For 32-byte transfer: The lower five bits should be fixed to 0 0000. For 128-byte transfer: The lower seven bits should be fixed to 000 0000.

Note: This register is updated when the GR0\_IBUS\_VEN bit in the graphics 0 register update control register (GR0\_UPDATE) is 1.

## 33.2.53 Frame Buffer Control Register 5 (Graphics 0) (GR0\_FLM5)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR0_FLM_LNUM[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR0_FLM_LOOP[10:0]										
Initial value:	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR0_FLM_LNUM[10:0]	0	R/W	Sets number of lines in a frame Number of lines is (GR0_FLM_LNUM + 1).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR0_FLM_LOOP[10:0]	1023	R/W	Number of lines when reading the addresses repeatedly by returning to the start address after reaching the end address. (GR0_FLM_LOOP + 1) lines are read.

Note: This register is updated when the GR0\_IBUS\_VEN bit in the graphics 0 register update control register (GR0\_UPDATE) is 1.

## 33.2.54 Frame Buffer Control Register 6 (Graphics 0) (GR0\_FLM6)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR0_FORMAT[3:0]				—	GR0_HW[10:0]										
Initial value:	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR0_YCC_SWAP[2:0]			GR0_RDSWA[2:0]			—	GR0_CW444_MD	—	—	GR0_STA_POS[5:0]					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	GR0_FORMAT[3:0]	8	R/W	Sets the format of the frame buffer read signal. 0: RGB565 1: RGB888 2: $\alpha$ RGB1555 3: $\alpha$ RGB4444 4: $\alpha$ RGB8888 5: CLUT8 6: CLUT4 7: CLUT1 8: YCbCr422 9: YCbCr444 10: RGB $\alpha$ 5551 11: RGB $\alpha$ 8888 12 to 15: Setting prohibited
27	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
26 to 16	GR0_HW[10:0]	0	R/W	Sets the width of the horizontal valid period. The width is (GR0_HW + 1) pixels. Note: The set value should be equal to or more than two.
15 to 13	GR0_YCC_SWAP[2:0]	0	R/W	Controls swapping of data read from buffer in the YCbCr422 format. 0: Cb/Y0/Cr/Y1 1: Y0/Cb/Y1/Cr 2: Cr/Y0/Cb/Y1 3: Y0/Cr/Y1/Cb 4: Y1/Cr/Y0/Cb 5: Cr/Y1/Cb/Y0 6: Y1/Cb/Y0/Cr 7: Cb/Y1/Cr/Y0

Bit	Bit Name	Initial Value	R/W	Description
12 to 10	GR0_RDSWA [2:0]	0	R/W	<p>8-Bit, 16-Bit, or 32-Bit Swap Setting</p> <p>These bits control swapping in frame buffer reading as follows.</p> <p>Bit 0 0: Swapped in 8-bit units. 1: Not swapped in 8-bit units.</p> <p>Bit 1 0: Swapped in 16-bit units. 1: Not swapped in 16-bit units.</p> <p>Bit 2 0: Swapped in 32-bit units. 1: Not swapped in 32-bit units.</p> <p>According to the setting of these bits, data is swapped as follows. Each number in parentheses ((1) to (8)) indicates 8-bit data.</p> <p>000: (1) (2) (3) (4) (5) (6) (7) (8) [Not swapped]</p> <p>001: (2) (1) (4) (3) (6) (5) (8) (7) [Swapped in 8-bit units]</p> <p>010: (3) (4) (1) (2) (7) (8) (5) (6) [Swapped in 16-bit units]</p> <p>011: (4) (3) (2) (1) (8) (7) (6) (5) [Swapped in 16-bit units + 8-bit units]</p> <p>100: (5) (6) (7) (8) (1) (2) (3) (4) [Swapped in 32-bit units]</p> <p>101: (6) (5) (8) (7) (2) (1) (4) (3) [Swapped in 32-bit units + 8-bit units]</p> <p>110: (7) (8) (5) (6) (3) (4) (1) (2) [Swapped in 32-bit units + 16-bit units]</p> <p>111: (8) (7) (6) (5) (4) (3) (2) (1) [Swapped in 32-bit units + 16-bit units + 8-bit units]</p>
9	—	0	R	<p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p>
8	GR0_CNV444_MD	0	R/W	<p>Sets the interpolation mode for YCbCr422 to YCbCr444 conversion.</p> <p>0: Hold interpolation 1: Average interpolation</p>
7, 6	—	All 0	R	<p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p>
5 to 0	GR0_STA_POS[5:0]	0	R/W	<p>Sets the amount of data to be skipped through.</p> <p>Specifically data amount equal to the amount indicated by GR0_STA_POS is skipped from the start of the line.</p>

Note: GR0\_YCC\_SWAP, GR0\_CNV444, and GR0\_STA\_POS are updated when GR0\_P\_VEN bit in the graphics 0 register update control register (GR0\_UPDATE) is 1.

GR0\_RDSWA is updated when the GR0\_UPDATE bit in the graphics 0 register update control register (GR0\_UPDATE) is 1.

GR0\_FORMAT and GR0\_HW are updated when GR0\_IBUS\_VEN and GR0\_P\_VEN bits in the graphics 0 register update control register (GR0\_UPDATE) are 1.

## 33.2.55 Alpha Blending Control Register 1 (Graphics 0) (GR0\_AB1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	GR0_GRC_DISP_ON	—	—	GR0_DISP_SEL[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	GR0_GRC_DISP_ON	0	R/W	Turns on/off frame-line display of the graphics image area. 0: Frame-line display off 1: Frame-line display on
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	GR0_DISP_SEL[1:0]	0	R/W	Selects the graphics display mode. 0: Background color display (GR0_BASE) 1: Lower-layer graphics display When displaying video image or enlarged graphics, select this setting. 2: Current graphics display When displaying graphics, select this setting. 3: Blended display of lower-layer graphics and current graphics* Note: * Select this setting whenever chroma-key processing is to proceed. Since only current graphics are to be displayed by chroma-key processing, set the $\alpha$ values for both pixels to be subject to chroma-keying and pixels not to be subject to chroma-keying to 255.

Note: This register is updated when GR0\_P\_VEN bit in the graphics 0 register update control register (GR0\_UPDATE) is 1.

## 33.2.56 Alpha Blending Control Register 2 (Graphics 0) (GR0\_AB2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR0_GRC_VS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR0_GRC_VW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR0_GRC_VS[10:0]	0	R/W	Vertical Start Position of Graphics Image Area. Note: The set value should be four or more (lines). GR0_GRC_VS + GR0_GRC_VW should be equal to or less than 2039 (lines).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR0_GRC_VW[10:0]	0	R/W	Vertical Width of Graphics Image Area.

Note: This register is updated when the GR0\_P\_VEN bit in the graphics 0 register update control register (GR0\_UPDATE) is 1.

## 33.2.57 Alpha Blending Control Register 3 (Graphics 0) (GR0\_AB3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR0_GRC_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR0_GRC_HW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR0_GRC_HS[10:0]	0	R/W	Horizontal Start Position of Graphics Image Area. Note: The set value should be 16 or more (clock cycles). GR0_GRC_HS + GR0_GRC_HW should be equal to or less than 2015 (clock cycles).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.



Bit	Bit Name	Initial Value	R/W	Description
10 to 0	GR0_GRC_HW[10:0]	0	R/W	Horizontal Width of Graphics Image Area. Note: For displaying an image with 1- or 2-pixel horizontal width, set GR0_HW to 2 and GR0_GRC_HW to 1 (1 pixel) or 2 (2 pixels).

Note: This register is updated when the GR0\_P\_VEN bit in the graphics 0 register update control register (GR0\_UPDATE) is 1.

### 33.2.58 Alpha Blending Control Register 7 (Graphics 0) (GR0\_AB7)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR0 CK_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR0_CK_ON	0	R/W	CLUT-Index/RGB-Index Chroma-Key Processing On/Off 0: Off 1: On

Note: This register is updated when the GR0\_P\_VEN bit in the graphics 0 register update control register (GR0\_UPDATE) is 1.

## 33.2.59 Alpha Blending Control Register 8 (Graphics 0) (GR0\_AB8)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR0_CK_KCLUT[7:0]								GR0_CK_KG[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR0_CK_KB[7:0]								GR0_CK_KR[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GR0_CK_KCLUT[7:0]	0	R/W	CLUT Signal for CLUT-Index Chroma-Key Processing CLUT: Unsigned 8 bits (0 to 255 [LSB])
23 to 16	GR0_CK_KG[7:0]	0	R/W	G Signal for RGB-Index Chroma-Key Processing G: Unsigned 8 bits (0 to 255 [LSB])
15 to 8	GR0_CK_KB[7:0]	0	R/W	B Signal for RGB-Index Chroma-Key Processing B: Unsigned 8 bits (0 to 255 [LSB])
7 to 0	GR0_CK_KR[7:0]	0	R/W	R Signal for RGB-Index Chroma-Key Processing R: Unsigned 8 bits (0 to 255 [LSB])

Note: This register is updated when the GR0\_P\_VEN bit in the graphics 0 register update control register (GR0\_UPDATE) is 1.

## 33.2.60 Alpha Blending Control Register 9 (Graphics 0) (GR0\_AB9)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR0_CK_A[7:0]								GR0_CK_G[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR0_CK_B[7:0]								GR0_CK_R[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GR0_CK_A[7:0]	0	R/W	Replaced Alpha Signal after RGB-Index Chroma-Key Processing $\alpha$ : Unsigned 8 bits (0 to 255 [LSB]) Note: These bits should always be set to 255 to display the current graphics only.
23 to 16	GR0_CK_G[7:0]	0	R/W	Replaced G Signal after RGB-Index Chroma-Key Processing G: Unsigned 8 bits (0 to 255 [LSB])
15 to 8	GR0_CK_B[7:0]	0	R/W	Replaced B Signal after RGB-Index Chroma-Key Processing B: Unsigned 8 bits (0 to 255 [LSB])
7 to 0	GR0_CK_R[7:0]	0	R/W	Replaced R Signal after RGB-Index Chroma-Key Processing R: Unsigned 8 bits (0 to 255 [LSB])

Note: This register is updated when the GR0\_P\_VEN bit in the graphics 0 register update control register (GR0\_UPDATE) is 1.

## 33.2.61 Alpha Blending Control Register 10 (Graphics 0) (GR0\_AB10)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR0_A0[7:0]								GR0_G0[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR0_B0[7:0]								GR0_R0[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GR0_A0 [7:0]	0	R/W	CLUT1 $\alpha$ 0 Signal Replaced with $\alpha$ signal when in the CLUT1 format and CLUT1 = 0. Replaced with $\alpha$ signal when in the $\alpha$ RGB1555/RGB $\alpha$ 5551 format and $\alpha$ = 0. Note: These bits should always be set to 255 to display the current graphics only.
23 to 16	GR0_G0 [7:0]	0	R/W	CLUT1 G0 Signal Replaced with G signal when in the CLUT1 format and CLUT1 = 0.
15 to 8	GR0_B0 [7:0]	0	R/W	CLUT1 B0 Signal Replaced with B signal when in the CLUT1 format and CLUT1 = 0.
7 to 0	GR0_R0 [7:0]	0	R/W	CLUT1 R0 Signal Replaced with R signal when in the CLUT1 format and CLUT1 = 0.

Note: This register is updated when the GR0\_P\_VEN bit in the graphics 0 register update control register (GR0\_UPDATE) is 1.

## 33.2.62 Alpha Blending Control Register 11 (Graphics 0) (GR0\_AB11)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR0_A1[7:0]								GR0_G1[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR0_B1[7:0]								GR0_R1[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GR0_A1 [7:0]	0	R/W	CLUT1 $\alpha$ 1 Signal Replaced with $\alpha$ signal when in the CLUT1 format and CLUT1 = 1. Replaced with $\alpha$ signal when in the $\alpha$ RGB1555/RGB $\alpha$ 5551 format and $\alpha$ = 1. Note: These bits should always be set to 255 to display the current graphics only.
23 to 16	GR0_G1 [7:0]	0	R/W	CLUT1 G1 Signal Replaced with G signal when in the CLUT1 format and CLUT1 = 1.
15 to 8	GR0_B1 [7:0]	0	R/W	CLUT1 B1 Signal Replaced with B signal when in the CLUT1 format and CLUT1 = 1.
7 to 0	GR0_R1 [7:0]	0	R/W	CLUT1 R1 Signal Replaced with R signal when in the CLUT1 format and CLUT1 = 1.

Note: This register is updated when the GR0\_P\_VEN bit in the graphics 0 register update control register (GR0\_UPDATE) is 1.

## 33.2.63 Background Color Control Register (Graphics 0) (GR0\_BASE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	GR0_BASE_G[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR0_BASE_B[7:0]								GR0_BASE_R[7:0]							
Initial value:	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	GR0_BASE_ G[7:0]	0	R/W	Background Color G/Y Signal G/Y: Unsigned 8 bits (0 to 255 [LSB])
15 to 8	GR0_BASE_ B[7:0]	128	R/W	Background Color B/Cb Signal B: Unsigned 8 bits (0 to 255 [LSB]) Cb: 8 bits; 128 offset binary; unsigned (0 to 255 [LSB])

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	GR0_BASE_R[7:0]	128	R/W	Background Color R/Cr Signal R: Unsigned 8 bits (0 to 255 [LSB]) Cr: 8 bits; 128 offset binary; unsigned (0 to 255 [LSB])

Note: This register is updated when the GR0\_P\_VEN bit in the graphics 0 register update control register (GR0\_UPDATE) is 1.

### 33.2.64 CLUT Table Control Register (Graphics 0) (GR0\_CLUT)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR0_CLT_SEL
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	GR0_CLT_SEL	0	R/W	CLUT Table Select Signal 0: Selects CLUT table 0. Referring to the CLUT table 0 value to expand to αRGB8888 The CPU side can read-access or write-access to the CLUT table 1. 1: Selects CLUT table 1. Referring to the CLUT table 1 value to expand to αRGB8888 The CPU side can read-access or write-access to the CLUT table 0.
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note: This register is updated when the GR0\_P\_VEN bit in the graphics 0 register update control register (GR0\_UPDATE) is 1.

## 33.2.65 SCL0 Register Update Control Register (SC1\_SCL0\_UPDATE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	SC1_SCL0_VEN_D	SC1_SCL0_VEN_C	—	—	—	SC1_SCL0_UPDATE	—	—	—	SC1_SCL0_VEN_B	—	—	—	SC1_SCL0_VEN_A
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/WC1	R/WC1	R	R	R	R/WC1	R	R	R	R/WC1	R	R	R	R/WC1

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13	SC1_SCL0_VEN_D	0	R/WC1	Scaling-Up Control and Frame Buffer Read Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
12	SC1_SCL0_VEN_C	0	R/WC1	Scaling-Down Control and Frame Buffer Read Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	SC1_SCL0_UPDATE	0	R/WC1	SYNC Control Register Update 0: Registers are not updated. 1: Registers are updated.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	SC1_SCL0_VEN_B	0	R/WC1	Synchronization Control and Scaling-up Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SC1_SCL0_VEN_A	0	R/WC1	Scaling-Down Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.

## 33.2.66 Mask Control Register (SC1\_SCL0\_FRC1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SC1_RES_VMASK[15:0]															
Initial value:	0	0	0	0	1	0	1	0	1	1	1	1	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SC1_RES_VMASK_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	SC1_RES_VMASK [15:0]	2800	R/W	Repeated Vsync Signal Masking Period Sets the repeated Vsync signal masking period beginning at a Vsync signal in terms of 128 pixel-clock periods. Masking period [usec] = SC1_RES_VMASK × 128 ÷ pixel clock frequency [MHz]
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SC1_RES_VMASK_ON	1	R/W	Repeated Vsync Signal Masking Control 0: Repeated Vsync signal masking control is disabled. 1: Repeated Vsync signal masking control is enabled.

Note: This register is updated when the SC1\_SCL0\_UPDATE bit in the SC1\_SCL0 register update control register (SC1\_SCL0\_UPDATE) is 1.

## 33.2.67 Missing Vsync Compensation Control Register (SC1\_SCL0\_FRC2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SC1_RES_VLACK[15:0]															
Initial value:	0	0	0	0	1	1	1	0	0	0	0	1	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SC1_RES_VLACK_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	SC1_RES_VLACK [15:0]	3600	R/W	Missing-Sync Compensating Pulse Output Wait Time Sets the wait time before outputting a missing-sync compensating pulse after a Vsync signal. Wait time [usec] = SC1_RES_VLACK × 128 ÷ pixel clock frequency [MHz]
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SC1_RES_VLACK_ON	1	R/W	Missing Vsync Signal Compensation 0: Compensation of missing Vsync signals is disabled. 1: Compensation of missing Vsync signals is enabled.

Note: This register is updated when the SC1\_SCL0\_UPDATE bit in the SC1\_SCL0 register update control register (SC1\_SCL0\_UPDATE) is 1.

## 33.2.68 Output Sync Select Register (SC1\_SCL0\_FRC3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SC1_RES_	—	—	—	—	—	—	—	SC1_RES_
								VS_IN_								VS_SEL
								SEL								
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	SC1_RES_ VS_IN_SEL	0	R/W	Horizontal and Vertical Sync Signal Output and Full-Screen Enable Signal Select This setting is ignored when cascaded connection is enabled (GR1_AB1.GR1_CUS_CON_ON = 1). 0: Horizontal and vertical sync signal output and full-screen enable signal from scaler 1 1: Horizontal and vertical sync signal output and full-screen enable signal from scaler 0
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SC1_RES_ VS_SEL	1	R/W	Vsync Signal Output Select 0: Externally input Vsync signal 1: Internally generated free-running Vsync signal

Note: This register is updated when the SC1\_SCL0\_UPDATE bit in the SC1\_SCL0 register update control register (SC1\_SCL0\_UPDATE) is 1.



## 33.2.69 Free-Running Period Control Register (SC1\_SCL0\_FRC4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	SC1_RES_FV[10:0]										
Initial value:	0	0	0	0	0	0	1	0	0	0	0	0	1	1	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SC1_RES_FH[10:0]										
Initial value:	0	0	0	0	0	0	1	1	0	0	0	1	1	1	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	SC1_RES_FV[10:0]	524	R/W	Free-Running Vsync Period Setting Free-running Vsync period = (SC1_RES_FV + 1) × horizontal period [usec]
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SC1_RES_FH[10:0]	799	R/W	Hsync Period Setting Hsync period [usec] = (SC1_RES_FH + 1) ÷ pixel clock frequency [MHz]

Note: This register is updated when the SC1\_SCL0\_UPDATE bit in the SC1\_SCL0 register update control register (SC1\_SCL0\_UPDATE) is 1.

## 33.2.70 Output Delay Control Register (SC1\_SCL0\_FRC5)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SC1_RES_FLD_DLY_SEL	SC1_RES_VSDLY[7:0]							
Initial value:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	SC1_RES_FLD_DLY_SEL	1	R/W	Field Determination Signal Delay Control 0: No delay 1: Delay of one vertical cycle
7 to 0	SC1_RES_VSDLY[7:0]	1	R/W	Vsync Signal Delay Control Adjusts the Vsync signal delay in the output Hsync period units. Vsync signal delay [usec]: SC1_RES_VSDLY × output Hsync period [usec]

Note: This register is updated when the SC1\_SCL0\_VEN\_B bit in the SC1\_SCL0 register update control register (SC1\_SCL0\_UPDATE) is 1.

## 33.2.71 Full-Screen Vertical Size Register (SC1\_SCL0\_FRC6)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	SC1_RES_F_VS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SC1_RES_F_VW[10:0]										
Initial value:	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	SC1_RES_F_VS[10:0]	35	R/W	Vertical Enable Signal Start Position for Full Screen. (VSYNC + V backporch lines) Note: The set value should be four or more (lines). SC1_RES_F_VS + SC1_RES_F_VW should be equal to or less than 2039 (lines).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SC1_RES_F_VW[10:0]	480	R/W	Vertical Enable Signal Width for Full Screen (lines) Note: SC1_RES_F_VS + SC1_RES_F_VW should be equal to or less than 2039 (lines).

Note: This register is updated when the SC1\_SCL0\_VEN\_B bit in the SC1\_SCL0 register update control register (SC1\_SCL0\_UPDATE) is 1.

## 33.2.72 Full-Screen Horizontal Size Register (SC1\_SCL0\_FRC7)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	SC1_RES_F_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SC1_RES_F_HW[10:0]										
Initial value:	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	SC1_RES_F_HS[10:0]	144	R/W	Horizontal Enable Signal Start Position for Full Screen. (HSYNC+H backporch pixel-clock cycles) Note: The set value should be 16 or more (clock cycles). SC1_RES_F_HS + SC1_RES_F_HW should be equal to or less than 2015 (clock cycles).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
10 to 0	SC1_RES_F_HW[10:0]	640	R/W	Horizontal Enable Signal Width for Full Screen (pixel-clock cycles) Note: SC1_RES_F_HS + SC1_RES_F_HW should be equal to or less than 2015 (clock cycles).

Note: This register is updated when the SC1\_SCL0\_VEN\_B bit in the SC1\_SCL0 register update control register (SC1\_SCL0\_UPDATE) is 1.

### 33.2.73 Vsync Detection Register (SC1\_SCL0\_FRC9)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	SC1_RES_QVLOCK	—	—	—	SC1_RES_QVLACK
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	SC1_RES_QVLOCK	0	R	Locked Vsync Signal Detection Flag 1: No repeated or missing Vsync signal input has been detected for four or more vertical periods. 0: Repeated or missing Vsync signal input has been detected.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SC1_RES_QVLACK	0	R	Missing Vsync Signal Detection Flag 1: Missing Vsync signal input has been detected. 0: No missing Vsync signal input has been detected.

### 33.2.74 Status Monitor 0 Register (SC1\_SCL0\_MON0)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SC1_RES_LIN_STAT[10:0]										
Initial value:	0	0	0	0	0	0										
R/W:	R	R	R	R	R	R										

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SC1_RES_LIN_STAT [10:0]	All 0	R	Current location of the image line input to the scaling-down control block.

## 33.2.75 Interrupt Control Register (SC1\_SCL0\_INT)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SC1_RES_LINE[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SC1_RES_LINE[10:0]	All 0	R/W	Setting of Interrupt on Image Line Input to Scaling-down Control Block When the location of the image line input to the scaling-down control block matches the SC1_RES_LINE setting, an interrupt signal is output.

Note: This register is updated when the SC1\_SCL0\_VEN\_A bit in the SC1\_SCL0 register update control register (SC1\_SCL0\_UPDATE) is 1.

## 33.2.76 Scaling-Down Control Register (SC1\_SCL0\_DS1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	SC1_RES_DS_V_ON	—	—	—	SC1_RES_DS_H_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	SC1_RES_DS_V_ON	1	R/W	Vertical Scale Down On/Off 0: Off 1: On
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SC1_RES_DS_H_ON	1	R/W	Horizontal Scale Down On/Off 0: Off 1: On

Note: This register is updated when the SC1\_SCL0\_VEN\_A bit in the SC1\_SCL0 register update control register (SC1\_SCL0\_UPDATE) is 1.

## 33.2.77 Vertical Capture Size Register (SC1\_SCL0\_DS2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	SC1_RES_VS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SC1_RES_VW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	SC1_RES_VS[10:0]	18	R/W	Vertical Position Setting for Video Signal Capturing (VSYNC + V backporch lines) Note: The set value should be four or more (lines). SC1_RES_VS + SC1_RES_VW should be equal to or less than 2039 (lines).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SC1_RES_VW[10:0]	240	R/W	Vertical Width of Video Signal to be Captured (Lines) Note: SC1_RES_VS + SC1_RES_VW should be equal to or less than 2039 (lines).

Note: This register is updated when the SC1\_SCL0\_VEN\_A bit in the SC1\_SCL0 register update control register (SC1\_SCL0\_UPDATE) is 1.

## 33.2.78 Horizontal Capture Size Register (SC1\_SCL0\_DS3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	SC1_RES_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	1	1	1	1	0	1	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SC1_RES_HW[10:0]										
Initial value:	0	0	0	0	0	1	0	1	1	0	1	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	SC1_RES_HS[10:0]	244	R/W	Horizontal Position Setting for Video Signal Capturing (HSYNC + H backporch video-image clock cycles) Note: The set value should be 16 or more (clock cycles). SC1_RES_HS + SC1_RES_HW should be equal to or less than 2015 (clock cycles).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SC1_RES_HW[10:0]	1440	R/W	Horizontal Width of Video Signal to be Captured (Video-image clock cycles) Note: SC1_RES_HS + SC1_RES_HW should be equal to or less than 2015 (clock cycles).

Note: This register is updated when the SC1\_SCL0\_VEN\_A bit in the SC1\_SCL0 register update control register (SC1\_SCL0\_UPDATE) is 1.

## 33.2.79 Horizontal Scale Down Register (SC1\_SCL0\_DS4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	SC1_RES_PFIL_SEL	SC1_RES_DS_H_INTERPOTYP	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC1_RES_DS_H_RATIO[15:0]															
Initial value:	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
29	SC1_RES_PFIL_SEL	0	R/W	Prefilter Mode Select for Brightness Signals 0: The prefilter is turned off. 1: The prefilter is turned on. (1/4 + 1/2 + 1/4)
28	SC1_RES_DS_H_INTERPOTYP	1	R/W	Horizontal Interpolation Mode Select 0: Hold interpolation 1: Linear interpolation
27 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	SC1_RES_DS_H_RATIO[15:0]	9224	R/W	Horizontal Scale Down Ratio ([15:12]: Integer part, [11:0]: Decimal part) round(SC1_RES_HW ÷ SC1_RES_OUT_HW × 4096) SC1_RES_DS_H_RATIO < 4096: Setting prohibited SC1_RES_DS_H_RATIO = 4096: 100% scale up SC1_RES_DS_H_RATIO > 4096: Scale down

Note: This register is updated when the SC1\_SCL0\_VEN\_A bit in the SC1\_SCL0 register update control register (SC1\_SCL0\_UPDATE) is 1.

## 33.2.80 Initial Vertical Phase Register (SC1\_SCL0\_DS5)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	SC1_RES_V_INTERPOTYP	SC1_RES_TOP_INIPHASE[11:0]											
Initial value:	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	SC1_RES_BTM_INIPHASE[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	SC1_RES_V_INTERPOTYP	1	R/W	Vertical Interpolation Mode Select 0: Hold interpolation 1: Linear interpolation

Bit	Bit Name	Initial Value	R/W	Description
27 to 16	SC1_RES_TOP_INIPHASE [11:0]	2048	R/W	Vertical Interpolation Start Phase for Top Field 0 to 4095 (0 to approx. 1.0)
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 0	SC1_RES_BTM_INIPHASE [11:0]	0	R/W	Vertical Interpolation Start Phase for Bottom Field 0 to 4095 (0 to approx. 1.0)

Note: This register is updated when the SC1\_SCL0\_VEN\_A and SC1\_SCL0\_VEN\_B bits in the SC1\_SCL0 register update control register (SC1\_SCL0\_UPDATE) are 1.

### 33.2.81 Vertical Scaling Register (SC1\_SCL0\_DS6)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC1_RES_V_RATIO[15:0]															
Initial value:	0	0	0	0	0	1	1	1	1	1	1	1	1	1	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	SC1_RES_V_RATIO [15:0]	2044	R/W	Vertical Scale Up/Down Ratio ([15:12]: Integer part, [11:0]: Decimal part) For scale down: $\text{round}(\text{SC1\_RES\_VW} \div \text{SC1\_RES\_OUT\_VW} \times 4096)$ For scale up: $\text{round}(\text{SC1\_RES\_IN\_VW} \div \text{SC1\_RES\_P\_VW} \times 4096)$ SC1_RES_V_RATIO < 4096: Scale up SC1_RES_V_RATIO = 4096: 100% scale up SC1_RES_V_RATIO > 4096: Scale down

Note: These bits updated when the SC1\_SCL0\_VEN\_A and SC1\_SCL0\_VEN\_B bits in the SC1\_SCL0 register update control register (SC1\_SCL0\_UPDATE) are 1. Accordingly, even a scaled-up graphics display requires both an input Vsync signal and output Vsync signal.



## 33.2.82 Scaling-Down Control Block Output Size Register (SC1\_SCL0\_DS7)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	SC1_RES_OUT_VW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SC1_RES_OUT_HW[10:0]										
Initial value:	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	SC1_RES_OUT_VW [10:0]	240	R/W	Number of Valid Lines in Vertical Direction Output by Scaling-down Control Block (lines) This bit setting is used for the number of lines to be written to the frame buffer. When SC1_SCL1_WR1.SC1_RES_LOOP is 0 (frame write mode), specify the number of lines for one frame. When SC1_SCL1_WR1.SC1_RES_LOOP is 1 (line write mode), specify the number of lines for repeated write. Note: The SC1_RES_OUT_VW value should be aligned in 4-line units and equal to or smaller than the SC1_RES_VW value.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SC1_RES_OUT_HW [10:0]	640	R/W	Number of Valid Horizontal Pixels Output by Scaling-Down Control Block (video-image clock cycles) Note: The SC1_RES_OUT_HW value should be aligned in 4-pixel units and equal to or smaller than the SC1_RES_HW value.

Note: This register is updated when the SC1\_SCL0\_VEN\_A and SC1\_SCL0\_VEN\_C bits in the SC1\_SCL0 register update control register (SC1\_SCL0\_UPDATE) are 1.

## 33.2.83 Scaling-Up Control Register (SC1\_SCL0\_US1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	SC1_RES_US_V_ON	—	—	—	SC1_RES_US_H_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
4	SC1_RES_US_V_ON	1	R/W	Vertical Scale Up On/Off 0: Off 1: On
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SC1_RES_US_H_ON	1	R/W	Horizontal Scale Up On/Off 0: Off 1: On

Note: This register is updated when the SC1\_SCL0\_VEN\_B bit in the SC1\_SCL0 register update control register (SC1\_SCL0\_UPDATE) is 1.

### 33.2.84 Output Image Vertical Size Register (SC1\_SCL0\_US2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	SC1_RES_P_VS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SC1_RES_P_VW[10:0]										
Initial value:	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	SC1_RES_P_VS[10:0]	35	R/W	Vertical Enable Signal Start Position for Output Image (VSYNC + V backporch lines) Note: The set value should be four or more (lines). SC1_RES_P_VS + SC1_RES_P_VW should be equal to or less than 2039 (lines).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SC1_RES_P_VW[10:0]	480	R/W	Vertical Enable Signal Width for Output Image (lines) Note: SC1_RES_P_VS + SC1_RES_P_VW should be equal to or less than 2039 (lines).

Note: This register is updated when the SC1\_SCL0\_VEN\_B bit in the SC1\_SCL0 register update control register (SC1\_SCL0\_UPDATE) is 1.

## 33.2.85 Output Image Horizontal Size Register (SC1\_SCL0\_US3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	SC1_RES_P_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SC1_RES_P_HW[10:0]										
Initial value:	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	SC1_RES_P_HS[10:0]	144	R/W	Horizontal Enable Signal Start Position for Output Image (HSYNC+H backporch pixel-clock cycles) Note: The set value should be 16 or more (clock cycles). SC1_RES_P_HS + SC1_RES_P_HW should be equal to or less than 2015 (clock cycles).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SC1_RES_P_HW[10:0]	640	R/W	Horizontal Enable Signal Width for Output Image (pixel-clock cycles) Note: SC1_RES_P_HS + SC1_RES_P_HW should be equal to or less than 2015 (clock cycles).

Note: This register is updated when the SC1\_SCL0\_VEN\_B bit in the SC1\_SCL0 register update control register (SC1\_SCL0\_UPDATE) is 1.

## 33.2.86 Scaling-Up Control Block Input Size Register (SC1\_SCL0\_US4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	SC1_RES_IN_VW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SC1_RES_IN_HW[10:0]										
Initial value:	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	SC1_RES_IN_VW[10:0]	240	R/W	Number of Valid Lines in Vertical Direction Input to Scaling-down Control Block (lines)
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	SC1_RES_IN_HW[10:0]	640	R/W	Number of Valid Horizontal Pixels Input to Scaling-down Control Block (pixel-clock cycles)

Note: This register is updated when the SC1\_SCL0\_VEN\_B and SC1\_SCL0\_VEN\_D bits in the SC1\_SCL0 register update control register (SC1\_SCL0\_UPDATE) are 1.

### 33.2.87 Horizontal Scale Up Register (SC1\_SCL0\_US5)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC1_RES_US_H_RATIO[15:0]															
Initial value:	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	SC1_RES_US_H_RATIO [15:0]	9224	R/W	Horizontal Scale Up Ratio ([15:12]: Integer part, [11:0]: Decimal part) round (SC1_RES_IN_HW ÷ SC1_RES_P_HW × 4096) SC1_RES_US_H_RATIO < 4096: Scale up SC1_RES_US_H_RATIO = 4096: 100% scale up SC1_RES_US_H_RATIO > 4096: Setting prohibited

Note: This register is updated when the SC1\_SCL0\_VEN\_B bit in the SC1\_SCL0 register update control register (SC1\_SCL0\_UPDATE) is 1.

## 33.2.88 Horizontal Scale Up Initial Phase Register (SC1\_SCL0\_US6)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	SC1_RES_US_H_INTERPOTYP	SC1_RES_US_HT_INIPHASE[11:0]											
Initial value:	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	SC1_RES_US_HB_INIPHASE[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	SC1_RES_US_H_INTERPOTYP	1	R/W	Horizontal Interpolation Mode Select 0: Hold interpolation 1: Linear interpolation
27 to 16	SC1_RES_US_HT_INIPHASE [11:0]	0	R/W	Horizontal Interpolation Start Phase for Top Field 0 to 4095 (0 to approx. 1.0)
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 0	SC1_RES_US_HB_INIPHASE [11:0]	0	R/W	Horizontal Interpolation Start Phase for Bottom Field 0 to 4095 (0 to approx. 1.0)

Note: This register is updated when the SC1\_SCL0\_VEN\_B bit in the SC1\_SCL0 register update control register (SC1\_SCL0\_UPDATE) is 1.

## 33.2.89 Trimming Register (SC1\_SCL0\_US7)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC1_RES_HCUT[7:0]							SC1_RES_VCUT[7:0]								
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 8	SC1_RES_HCUT[7:0]	0	R/W	Horizontal Amount of Cut-off Post-Scaling Image (Right and Left Parts) Sets the number of pixel-clock cycles.
7 to 0	SC1_RES_VCUT[7:0]	0	R/W	Vertical Amount of Cut-off Post-Scaling Image (Upper and Lower Parts) Sets the number of lines.

Note: This register is updated when the SC1\_SCL0\_VEN\_B bit in the SC1\_SCL0 register update control register (SC1\_SCL0\_UPDATE) is 1.

### 33.2.90 Frame Buffer Read Select Register (SC1\_SCL0\_US8)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	SC1_RES_IBUS_SYNC_SEL	—	—	—	SC1_RES_DISP_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	SC1_RES_IBUS_SYNC_SEL	0	R/W	Sync Signal Select for Frame Buffer Read Block 0: Sync signals from the scaling-up control block 1: Sync signals from the graphics processing block
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SC1_RES_DISP_ON	0	R/W	Post-Scaling Image Frame Display On/Off 0: Frame display on 1: Frame display off

Note: SC1\_RES\_IBUS\_SYNC\_SEL is updated when the SC1\_SCL0\_VEN\_B and SC1\_SCL0\_VEN\_D bits in the SC1\_SCL0 register update control register (SC1\_SCL0\_UPDATE) are 1. SC1\_RES\_DISP\_ON is updated when the SC1\_SCL0\_VEN\_B bit in the SC1\_SCL0 register update control register (SC1\_SCL0\_UPDATE) is 1.

## 33.2.91 Background Color Register (SC1\_SCL0\_OVR1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	SC1_RES_BK_COL_R[7:0]							
Initial value:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC1_RES_BK_COL_G[7:0]								SC1_RES_BK_COL_B[7:0]							
Initial value:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	SC1_RES_BK_COL_R[7:0]	128	R/W	Background Color Setting R/Cr Signal R: 8 bits; unsigned (0 to 255 [LSB]) Cr: 8 bits; 128 offset binary; unsigned (0 to 255 [LSB])
15 to 8	SC1_RES_BK_COL_G[7:0]	0	R/W	Background Color Setting G/Y Signal G/Y: 8 bits; unsigned (0 to 255 [LSB])
7 to 0	SC1_RES_BK_COL_B[7:0]	128	R/W	Background Color Setting B/Cb Signal B: 8 bits; unsigned (0 to 255 [LSB]) Cb: 8 bits; 128 offset binary; unsigned (0 to 255 [LSB])

Note: This register is updated when the SC1\_SCL0\_VEN\_B bit in the SC1\_SCL0 register update control register (SC1\_SCL0\_UPDATE) is 1.

## 33.2.92 SCL1 Register Update Control Register (SC1\_SCL1\_UPDATE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	SC1_SCL1_UPDATE_B	—	—	—	SC1_SCL1_UPDATE_A
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/WC1	R	R	R	R/WC1
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	SC1_SCL1_VEN_B	—	—	—	SC1_SCL1_VEN_A
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/WC1	R	R	R	R/WC1

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	SC1_SCL1_UPDATE_B	0	R/WC1	Frame Buffer Write Control Register Update 0: Registers are not updated. 1: Registers are updated.
19 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	SC1_SCL1_UPDATE_A	0	R/WC1	Frame Buffer Write Control Register Update 0: Registers are not updated. 1: Registers are updated.
15 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	SC1_SCL1_VEN_B	0	R/WC1	Frame Buffer Write Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SC1_SCL1_VEN_A	0	R/WC1	Frame Buffer Write Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.

## 33.2.93 Writing Mode Register (SC1\_SCL1\_WR1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SC1_RES_WRSWA [2:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	—	—	—	—	—	—	—	—	—	SC1_RES_TB_ADD_MOD	SC1_RES_DS_WR_MD[2:0]	SC1_RES_MD[1:0]	SC1_RES_LOOP	SC1_RES_BST_MD	—	—	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.



Bit	Bit Name	Initial Value	R/W	Description
18 to 16	SC1_RES_WRSWA[2:0]	All 0	R/W	<p>8-Bit, 16-Bit, or 32-Bit Swap Setting</p> <p>These bits control swapping in frame buffer writing as follows.</p> <p>Bit 0</p> <p>0: Swapped in 8-bit units. 1: Not swapped in 8-bit units.</p> <p>Bit 1</p> <p>0: Swapped in 16-bit units. 1: Not swapped in 16-bit units.</p> <p>Bit 2</p> <p>0: Swapped in 32-bit units. 1: Not swapped in 32-bit units.</p> <p>According to the setting of these bits, data is swapped as follows. Each number in parentheses ((1) to (8)) indicates 8-bit data.</p> <p>000: (1) (2) (3) (4) (5) (6) (7) (8) [Not swapped]</p> <p>001: (2) (1) (4) (3) (6) (5) (8) (7) [Swapped in 8-bit units]</p> <p>010: (3) (4) (1) (2) (7) (8) (5) (6) [Swapped in 16-bit units]</p> <p>011: (4) (3) (2) (1) (8) (7) (6) (5) [Swapped in 16-bit units + 8-bit units]</p> <p>100: (5) (6) (7) (8) (1) (2) (3) (4) [Swapped in 32-bit units]</p> <p>101: (6) (5) (8) (7) (2) (1) (4) (3) [Swapped in 32-bit units + 8-bit units]</p> <p>110: (7) (8) (5) (6) (3) (4) (1) (2) [Swapped in 32-bit units + 16-bit units]</p> <p>111: (8) (7) (6) (5) (4) (3) (2) (1) [Swapped in 32-bit units + 16-bit units + 8-bit units]</p> <p>Note: When YCbCr422 or RGB565 is selected as a frame buffer video-signal writing format, these bits should be set to 000 [Not swapped].</p>
15 to 8	—	All 0	R	<p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p>
7	SC1_RES_TB_ADD_MOD	0	R/W	<p>Top and Bottom Data Write Address Specification Method</p> <p>0: A write address is specified in common for top and bottom data. 1: Separate write addresses are specified for top and bottom data.</p>
6 to 4	SC1_RES_DS_WR_MD [2:0]	0	R/W	<p>Frame Buffer Writing Mode for Image Processing</p> <p>0: Normal 1: Horizontal mirroring 2: 90° rotation 3: 180° rotation 4: 270° rotation 5 to 7: Setting prohibited</p>
3, 2	SC1_RES_MD [1:0]	0	R/W	<p>Frame Buffer Video-Signal Writing Format</p> <p>0: YCbCr422 (16 bits) 1: RGB565 (16 bits) 2: RGB888 (24 (32) bits) 3: YCbCr444 (24 (32) bits)</p>
1	SC1_RES_LOOP	0	R/W	<p>Frame Buffer Write Mode Select</p> <p>0: Frame mode 1: Line mode (read as ring buffer)</p>
0	SC1_RES_BST_MD	0	R/W	<p>Transfer Burst Length for Frame Buffer Writing</p> <p>0: 32-byte transfer (4 bursts) 1: 128-byte transfer (16 bursts)</p>

Note: SC1\_RES\_LOOP and SC1\_RES\_BST\_MD are updated when the SC1\_SCL1\_VEN\_B bit in the SC1\_SCL1 register update control register (SC1\_SCL1\_UPDATE) is 1.

SC1\_RES\_TB\_ADD\_MOD, SC1\_RES\_DS\_WR\_MD, and SC1\_RES\_MD are updated when the SC1\_SCL1\_VEN\_A and SC1\_SCL1\_VEN\_B bits in the SC1\_SCL1 register update control register (SC1\_SCL1\_UPDATE) are 1.

SC1\_RES\_WRSWA is updated when the SC1\_SCL1\_UPDATE\_A bit in the SC1\_SCL1 register update control register (SC1\_SCL1\_UPDATE) is 1.

## 33.2.94 Write Address Register 1T (SC1\_SCL1\_WR2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SC1_RES_BASE[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC1_RES_BASE[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SC1_RES_BASE [31:0]	0	R/W	Frame Buffer Base Address Sets the start address of the frame buffer to store the frame data for the top field when SC1_SCL1_WR1.SC1_RES_TB_ADD_MOD = 1 or that for the top and bottom fields when SC1_SCL1_WR1.SC1_RES_TB_ADD_MOD = 0. For 32-byte transfer: The lower five bits should be fixed to 0 0000. For 128-byte transfer: The lower seven bits should be fixed to 000 0000.

Note: This register is updated when the SC1\_SCL1\_VEN\_B bit in the SC1\_SCL1 register update control register (SC1\_SCL1\_UPDATE) is 1.

## 33.2.95 Write Address Register 2T (SC1\_SCL1\_WR3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	SC1_RES_LN_OFF[14:0]														
Initial value:	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SC1_RES_FLM_NUM[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30 to 16	SC1_RES_LN_OFF [14:0]	2048	R/W	Frame Buffer Line Offset Address Sets the line offset address for calculating the line start address for the top field when SC1_SCL1_WR1.SC1_RES_TB_ADD_MOD = 1 or that for the top and bottom fields when SC1_SCL1_WR1.SC1_RES_TB_ADD_MOD = 0. Line 0: SC1_RES_BASE Line 1: SC1_RES_BASE + SC1_RES_LN_OFF × 1 : Line n: SC1_RES_BASE + SC1_RES_LN_OFF × n For 32-byte transfer: The lower five bits should be fixed to 0 0000. For 128-byte transfer: The lower seven bits should be fixed to 000 0000.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	SC1_RES_FLM_NUM [9:0]	1	R/W	Number of Frames of Buffer to be Written to Sets the number of frames for the top field when SC1_SCL1_WR1.SC1_RES_TB_ADD_MOD = 1 or that for the top and bottom fields when SC1_SCL1_WR1.SC1_RES_TB_ADD_MOD = 0 Number of frames defined by SC1_RES_FLM_NUM + 1 are used.

Note: This register is updated when the SC1\_SCL1\_VEN\_B bit in the SC1\_SCL1 register update control register (SC1\_SCL1\_UPDATE) is 1.

## 33.2.96 Write Address Register 3T (SC1\_SCL1\_WR4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	SC1_RES_FLM_OFF[22:16]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC1_RES_FLM_OFF[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
22 to 0	SC1_RES_FLM_OFF [22:0]	524288	R/W	Frame Buffer Frame Offset Address Sets the frame offset address for calculating the start address of each frame for the top field when SC1_SCL1_WR1.SC1_RES_TB_ADD_MOD = 1 or that for the top and bottom fields when SC1_SCL1_WR1.SC1_RES_TB_ADD_MOD = 0. Buffer 0: SC1_RES_BASE Buffer 1: SC1_RES_BASE + SC1_RES_FLM_OFF × 1 : Buffer n: SC1_RES_BASE + SC1_RES_FLM_OFF × n For 32-byte transfer: The lower five bits should be fixed to 0 0000. For 128-byte transfer: The lower seven bits should be fixed to 000 0000.

Note: This register is updated when the SC1\_SCL1\_VEN\_B bit in the SC1\_SCL1 register update control register (SC1\_SCL1\_UPDATE) is 1.

## 33.2.97 Frame Sub-Sampling Register (SC1\_SCL1\_WR5)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	SC1_RES_INTER	—	—	SC1_RES_FS_RATE[1:0]	—	—	—	SC1_RES_FLD_SEL	—	—	—	—	SC1_RES_WENB
Initial value:	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R/W	R/W	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	SC1_RES_INTER	1	R/W	Field Operating Mode Select 0: Progressive 1: Interlace
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	SC1_RES_FS_RATE[1:0]	0	R/W	Writing Rate Sets the frame buffer writing rate to the vertical frequency of the input signal. 0: 1/1 an input signal (The SC1_RES_FLD_SEL setting is invalid.) 1: 1/2 an input signal 2: 1/4 an input signal 3: 1/8 an input signal
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	SC1_RES_FLD_SEL	0	R/W	Write Field Select 0: Top field 1: Bottom field
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SC1_RES_WENB	0	R/W	Frame Buffer Write Enable After making the setting to enable writing, writing starts from the second frame. 0: Frame buffer writing is disabled. 1: Frame buffer writing is enabled.

Note: SC1\_RES\_INTER, SC1\_RES\_FS\_RATE[1:0], and SC1\_RES\_FLD\_SEL are updated when the SC1\_SCL1\_VEN\_A bit in the SC1\_SCL1 register update control register (SC1\_SCL1\_UPDATE) is 1. SC1\_RES\_WENB is updated when the SC1\_SCL1\_VEN\_A and SC1\_SCL1\_VEN\_B bits in the SC1\_SCL1 register update control register (SC1\_SCL1\_UPDATE) are 1.

## 33.2.98 Bit Reduction Register (SC1\_SCL1\_WR6)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	SC1_RES_DTH_ON	—	—	—	SC1_RES_BITDEC_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	SC1_RES_DTH_ON	0	R/W	Dither Correction On/Off 0: Off (rounded off) 1: On (2 × 2 dither pattern)
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SC1_RES_BITDEC_ON	0	R/W	Bit Reduction On/Off 0: Off 1: On

Note: This register is updated when the SC1\_SCL1\_VEN\_A bit in the SC1\_SCL1 register update control register (SC1\_SCL1\_UPDATE) is 1.

## 33.2.99 Write Detection Register (SC1\_SCL1\_WR7)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SC1_RES_OVERFLOW	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	—	—	—	—	—	—	SC1_RES_FLM_CNT[9:0]									—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	SC1_RES_OVERFLOW	0	R	Line Buffer Overflow Detect 1: Line buffer has overflowed. 0: Line buffer has not overflowed.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	SC1_RES_FLM_CNT[9:0]	0	R	Frame Number Before Frame Being Accessed Frame number before the frame being accessed in the top field when SC1_SCL1_WR1.SC1_RES_TB_ADD_MOD = 1 or that in the top or bottom field when SC1_SCL1_WR1.SC1_RES_TB_ADD_MOD = 0.

## 33.2.100 Write Address Register 1B (SC1\_SCL1\_WR8)

	Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
		SC1_RES_BASE_B[31:16]															
Initial value:		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
	Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
		SC1_RES_BASE_B[15:0]															
Initial value:		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SC1_RES_BASE_B[31:0]	0	R/W	Frame Buffer Base Address for Bottom Sets the start address of the frame buffer to store the frame data for the bottom field when SC1_SCL1_WR1.SC1_RES_TB_ADD_MOD = 1. For 32-byte transfer: The lower five bits should be fixed to 0_0000. For 128-byte transfer: The lower seven bits should be fixed to 000_0000.

Note: This register is updated when the SC1\_SCL1\_VEN\_B bit in the SC1\_SCL1 register update control register (SC1\_SCL1\_UPDATE) is 1.



## 33.2.101 Write Address Register 2B (SC1\_SCL1\_WR9)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	SC1_RES_LN_OFF_B[14:0]														
Initial value:	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SC1_RES_FLM_NUM_B[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30 to 16	SC1_RES_LN_OFF_B [14:0]	2048	R/W	Frame Buffer Line Offset Address for Bottom Sets the line offset address for calculating the line start address for the bottom field when SC1_SCL1_WR1.SC1_RES_TB_ADD_MOD = 1. Line 0: SC1_RES_BASE_B Line 1: SC1_RES_BASE_B + SC1_RES_LN_OFF_B × 1 : Line n: SC1_RES_BASE + SC1_RES_LN_OFF_B × n For 32-byte transfer: The lower five bits should be fixed to 0_0000. For 128-byte transfer: The lower seven bits should be fixed to 000_0000.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	SC1_RES_FLM_NUM_B [9:0]	1	R/W	Number of Frames of Buffer to be Written to for Bottom Field Number of frames defined by SC1_RES_FLM_NUM_B + 1 are used when SC1_SCL1_WR1.SC1_RES_TB_ADD_MOD = 1.

Note: This register is updated when the SC1\_SCL1\_VEN\_B bit in the SC1\_SCL1 register update control register (SC1\_SCL1\_UPDATE) is 1.

## 33.2.102 Write Address Register 3B (SC1\_SCL1\_WR10)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	SC1_RES_FLM_OFF_B[22:16]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC1_RES_FLM_OFF_B[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
22 to 0	SC1_RES_FLM_OFF_B[22:0]	524288	R/W	Frame Buffer Frame Offset Address for Bottom Sets the frame offset address for calculating the start address of each frame for the bottom field when SC1_SCL1_WR1.SC1_RES_TB_ADD_MOD = 1. Buffer 0: SC1_RES_BASE_B Buffer 1: SC1_RES_BASE_B + SC1_RES_FLM_OFF_B × 1 : Buffer n: SC1_RES_BASE_B + SC1_RES_FLM_OFF_B × n For 32-byte transfer: The lower five bits should be fixed to 0_0000. For 128-byte transfer: The lower seven bits should be fixed to 000_0000.

Note: This register is updated when the SC1\_SCL1\_VEN\_B bit in the SC1\_SCL1 register update control register (SC1\_SCL1\_UPDATE) is 1.

## 33.2.103 Write Detection Register B (SC1\_SCL1\_WR11)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	—	—	—	—	—	—	SC1_RES_FLM_CNT_B[9:0]									—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	SC1_RES_FLM_CNT_B[9:0]	0	R	Frame number before the frame being accessed in Bottom Field Frame number before the frame being accessed in the bottom field when SC1_SCL1_WR11.SC1_RES_TB_ADD_MOD = 1.

## 33.2.104 Status Monitor 1 Register (SC1\_SCL1\_MON1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SC1_PBUF_NUM[1:0]		—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	SC1_PBUF_NUM[1:0]	All 0	R	Write pointer indicating the pointer buffer number corresponding to the location currently being written to.
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

## 33.2.105 Pointer Buffer 0 Register (SC1\_SCL1\_PBUF0)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SC1_PBUF0_ADD[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC1_PBUF0_ADD[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SC1_PBUF0_ADD[31:0]	All 0	R	Start address of the write buffer pointed to by pointer buffer 0.

## 33.2.106 Pointer Buffer 1 Register (SC1\_SCL1\_PBUF1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SC1_PBUF1_ADD[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC1_PBUF1_ADD[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SC1_PBUF1_ADD[31:0]	All 0	R	Start address of the write buffer pointed to by pointer buffer 1.

## 33.2.107 Pointer Buffer 2 Register (SC1\_SCL1\_PBUF2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SC1_PBUF2_ADD[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC1_PBUF2_ADD[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SC1_PBUF2_ADD[31:0]	All 0	R	Start address of the write buffer pointed to by pointer buffer 2.

## 33.2.108 Pointer Buffer 3 Register (SC1\_SCL1\_PBUF3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SC1_PBUF3_ADD[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC1_PBUF3_ADD[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	SC1_PBUF3_ADD[31:0]	All 0	R	Start address of the write buffer pointed to by pointer buffer 3.

## 33.2.109 Pointer Buffer and Field Information Register (SC1\_SCL1\_PBUF\_FLD)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	SC1_FLD_INF3	—	—	—	—	—	—	—	SC1_FLD_INF2
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SC1_FLD_INF1	—	—	—	—	—	—	—	SC1_FLD_INF0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	SC1_FLD_INF3	0	R	Top or bottom field information of the frame buffer pointed to by pointer buffer 3. 0: Bottom 1: Top
23 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	SC1_FLD_INF2	0	R	Top or bottom field information of the frame buffer pointed to by pointer buffer 2. 0: Bottom 1: Top
15 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	SC1_FLD_INF1	0	R	Top or bottom field information of the frame buffer pointed to by pointer buffer 1. 0: Bottom 1: Top
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SC1_FLD_INF0	0	R	Top or bottom field information of the frame buffer pointed to by pointer buffer 0. 0: Bottom 1: Top

## 33.2.110 Pointer Buffer Control Register (SC1\_SCL1\_PBUF\_CNT)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SC1_PBUF_RST
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	SC1_PBUF_RST	0	R/W	Reset Control for Pointer Buffer 0: Pointer buffer is not reset. 1: Pointer buffer is reset.
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note: This register is updated when the SC1\_SCL1\_UPDATE\_B bit in the SC1\_SCL1 register update control register (SC1\_SCL1\_UPDATE) is 1.

## 33.2.111 Graphics 1 Register Update Control Register (GR1\_UPDATE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	GR1_UPDATE	—	—	—	GR1_P_VEN	—	—	—	—	GR1_IBUS_VEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/WC1	R	R	R	R/WC1	R	R	R	R	R/WC1

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	GR1_UPDATE	0	R/WC1	Frame Buffer Read Control Register Update 0: Registers are not updated. 1: Registers are updated.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	GR1_P_VEN	0	R/WC1	Graphics Display Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
0	GR1_IBUS_VEN	0	R/WC1	Frame Buffer Read Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.

### 33.2.112 Frame Buffer Read Control Register (Graphics 1) (GR1\_FLM\_RD)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR1_R_ENB
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR1_R_ENB	0	R/W	Frame Buffer Read Enable 0: Frame buffer reading is disabled. 1: Frame buffer reading is enabled.

Note: This register is updated when the GR1\_IBUS\_VEN bit in the graphics 1 register update control register (GR1\_UPDATE) is 1.



## 33.2.113 Frame Buffer Control Register 1 (Graphics 1) (GR1\_FLM1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR1_FLD_SEL	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR1_LN_OFF_DIR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	GR1_FLM_SEL[1:0]	—	—	—	—	—	—	—	—	GR1_BST_MD
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	GR1_FLD_SEL	0	R/W	Enables or disables top or bottom field selection for the next frame buffer to be read. 0: Field selection is disabled. 1: Top or bottom field can be selected.
30 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	GR1_LN_OFF_DIR	0	R/W	Selects the line offset address direction of the frame buffer. 0: Increments the address by the line offset address. 1: Decrements the address by the line offset address.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	GR1_FLM_SEL [1:0]	0	R/W	Selects a frame buffer address setting signal. 0: Links to scaling-down process. (This setting is prohibited when separate write addresses are specified for the top and bottom fields; that is, SC1_RES_TB_ADD_MOD = 1 in SC1_SCL1_WR1.) 1: Selects GR1_FLM_NUM. 2: Setting prohibited. 3: Links to pointer buffer.
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR1_BST_MD	0	R/W	Frame Buffer Burst Transfer Mode 0: 32-byte transfer 1: 128-byte transfer

Note: GR1\_FLD\_SEL is updated when the GR1\_P\_VEN bit in the graphics 1 register update control register (GR1\_UPDATE) is 1.

GR1\_LN\_OFF\_DIR and GR1\_IMR\_FLM\_INV are updated when the GR1\_IBUS\_VEN bit in the graphics 1 register update control register (GR1\_UPDATE) is 1.

GR1\_FLM\_SEL is updated when the GR1\_P\_VEN and GR1\_IBUS\_VEN bits in the graphics 1 register update control register (GR1\_UPDATE) are 1.

GR1\_BST\_MD is updated when the GR1\_IBUS\_VEN and GR1\_P\_VEN bits in the graphics 1 register update control register (GR1\_UPDATE) are 1.

## 33.2.114 Frame Buffer Control Register 2 (Graphics 1) (GR1\_FLM2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR1_BASE[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR1_BASE[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	GR1_BASE [31:0]	0	R/W	Frame Buffer Base Address Sets the start address of the frame buffer where frame data is to be stored. GR1_BASE[4:3] and GR1_BASE[6:3] are referred to during 32-byte burst transfer and 128-byte burst transfer, respectively, to skip the start line data. The lower three bits should be fixed to 000.

Note: This register is updated when the GR1\_IBUS\_VEN and GR1\_P\_VEN bits in the graphics 1 register update control register (GR1\_UPDATE) are 1.

## 33.2.115 Frame Buffer Control Register 3 (Graphics 1) (GR1\_FLM3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR1_FLD_NXT	GR1_LN_OFF[14:0]														
Initial value:	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	GR1_FLM_NUM[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	GR1_FLD_NXT	0	R/W	Top or Bottom Field Selection for Next Frame Buffer 0: Bottom 1: Top
30 to 16	GR1_LN_OFF[14:0]	2048	R/W	Frame Buffer Line Offset Address Sets the line offset address for calculating the start address of each line. Line 0: GR1_BASE Line 1: GR1_BASE + GR1_LN_OFF × 1 : Line n: GR1_BASE + GR1_LN_OFF × n For 32-byte transfer: The lower five bits should be fixed to 0 0000. For 128-byte transfer: The lower seven bits should be fixed to 000 0000.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
9 to 0	GR1_FLM_NUM[9:0]	1	R/W	Frame Number of Frame Buffer Manually set the frame number when GR1_FLM_SEL = 1.

Note: GR1\_FLD\_NEXT is updated when the GR1\_P\_VEN bit in the graphics 1 register update control register (GR1\_UPDATE) is 1.

GR1\_LN\_OFF[14:0] and GR1\_FLM\_NUM[9:0] are updated when the GR1\_IBUS\_VEN bit in the graphics 1 register update control register (GR1\_UPDATE) is 1.

### 33.2.116 Frame Buffer Control Register 4 (Graphics 1) (GR1\_FLM4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	GR1_FLM_OFF[22:16]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR1_FLM_OFF[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
22 to 0	GR1_FLM_OFF[22:0]	524288	R/W	Frame Buffer Frame Offset Address Specifies the frame offset address used for calculating the start address of each frame buffer when more than one buffer is used. Buffer 0: GR1_BASE Buffer 1: GR1_BASE + GR1_FLM_OFF × 1 : Buffer n: GR1_BASE + GR1_FLM_OFF × n For 32-byte transfer: The lower five bits should be fixed to 0 0000. For 128-byte transfer: The lower seven bits should be fixed to 000 0000.

Note: This register is updated when the GR1\_IBUS\_VEN bit in the graphics 1 register update control register (GR1\_UPDATE) is 1.

## 33.2.117 Frame Buffer Control Register 5 (Graphics 1) (GR1\_FLM5)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR1_FLM_LNUM[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR1_FLM_LOOP[10:0]										
Initial value:	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR1_FLM_LNUM[10:0]	0	R/W	Sets number of lines in a frame Number of lines is (GR1_FLM_LNUM + 1).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR1_FLM_LOOP[10:0]	1023	R/W	Number of lines when reading the addresses repeatedly by returning to the start address after reaching the end address. (GR1_FLM_LOOP + 1) lines are read.

Note: This register is updated when the GR1\_IBUS\_VEN bit in the graphics 1 register update control register (GR1\_UPDATE) is 1.

## 33.2.118 Frame Buffer Control Register 6 (Graphics 1) (GR1\_FLM6)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR1_FORMAT[3:0]				—	GR1_HW[10:0]										
Initial value:	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR1_YCC_SWAP[2:0]			GR1_RDSWA[2:0]			—	GR1_CW444_MD	—	—	GR1_STA_POS[5:0]					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	GR1_FORMAT[3:0]	8	R/W	Sets the format of the frame buffer read signal. 0: RGB565 1: RGB888 2: $\alpha$ RGB1555 3: $\alpha$ RGB4444 4: $\alpha$ RGB8888 5: CLUT8 6: CLUT4 7: CLUT1 8: YCbCr422 9: YCbCr444 10: RGB $\alpha$ 5551 11: RGB $\alpha$ 8888 12 to 15: Setting prohibited
27	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
26 to 16	GR1_HW[10:0]	0	R/W	Sets the width of the horizontal valid period. The width is (GR1_HW + 1) pixels. Note: The set value should be equal to or more than two.
15 to 13	GR1_YCC_SWAP[2:0]	0	R/W	Controls swapping of data read from buffer in the YCbCr422 format. 0: Cb/Y0/Cr/Y1 1: Y0/Cb/Y1/Cr 2: Cr/Y0/Cb/Y1 3: Y0/Cr/Y1/Cb 4: Y1/Cr/Y0/Cb 5: Cr/Y1/Cb/Y0 6: Y1/Cb/Y0/Cr 7: Cb/Y1/Cr/Y0

Bit	Bit Name	Initial Value	R/W	Description
12 to 10	GR1_RDSWA [2:0]	0	R/W	<p>8-Bit, 16-Bit, or 32-Bit Swap Setting</p> <p>These bits control swapping in frame buffer reading as follows.</p> <p>Bit 0 0: Swapped in 8-bit units. 1: Not swapped in 8-bit units.</p> <p>Bit 1 0: Swapped in 16-bit units. 1: Not swapped in 16-bit units.</p> <p>Bit 2 0: Swapped in 32-bit units. 1: Not swapped in 32-bit units.</p> <p>According to the setting of these bits, data is swapped as follows. Each number in parentheses ((1) to (8)) indicates 8-bit data.</p> <p>000: (1) (2) (3) (4) (5) (6) (7) (8) [Not swapped]</p> <p>001: (2) (1) (4) (3) (6) (5) (8) (7) [Swapped in 8-bit units]</p> <p>010: (3) (4) (1) (2) (7) (8) (5) (6) [Swapped in 16-bit units]</p> <p>011: (4) (3) (2) (1) (8) (7) (6) (5) [Swapped in 16-bit units + 8-bit units]</p> <p>100: (5) (6) (7) (8) (1) (2) (3) (4) [Swapped in 32-bit units]</p> <p>101: (6) (5) (8) (7) (2) (1) (4) (3) [Swapped in 32-bit units + 8-bit units]</p> <p>110: (7) (8) (5) (6) (3) (4) (1) (2) [Swapped in 32-bit units + 16-bit units]</p> <p>111: (8) (7) (6) (5) (4) (3) (2) (1) [Swapped in 32-bit units + 16-bit units + 8-bit units]</p>
9	—	0	R	<p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p>
8	GR1_CNV444_MD	0	R/W	<p>Sets the interpolation mode for YCbCr422 to YCbCr444 conversion.</p> <p>0: Hold interpolation 1: Average interpolation</p>
7, 6	—	All 0	R	<p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p>
5 to 0	GR1_STA_POS[5:0]	0	R/W	<p>Sets the amount of data to be skipped through.</p> <p>Specifically data amount equal to the amount indicated by GR1_STA_POS is skipped from the start of the line.</p>

Note: GR1\_YCC\_SWAP, GR1\_CNV444, and GR1\_STA\_POS are updated when GR1\_P\_VEN bit in the graphics 1 register update control register (GR1\_UPDATE) is 1.

GR1\_RDSWA is updated when the GR1\_UPDATE bit in the graphics 1 register update control register (GR1\_UPDATE) is 1.

GR1\_FORMAT and GR1\_HW are updated when GR1\_IBUS\_VEN and GR1\_P\_VEN bits in the graphics 1 register update control register (GR1\_UPDATE) are 1.

## 33.2.119 Alpha Blending Control Register 1 (Graphics 1) (GR1\_AB1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	GR1_CUS_CON_ON	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR1_ARC_MUL	GR1_ACALC_MD	—	GR1_ARC_ON	—	—	—	GR1_ARC_DISP_ON	—	—	—	GR1_GRC_DISP_ON	—	—	GR1_DISP_SEL[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	GR1_CUS_CON_ON	0	R/W	Enables or disables cascaded connection. 0: Cascaded connection is disabled. 1: Cascaded connection is enabled.
27 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15	GR1_ARC_MUL	0	R/W	Enables or disables multiplication with the current $\alpha$ for alpha blending in rectangular area. 0: Disabled 1: Enabled
14	GR1_ACALC_MD	0	R/W	Enables or disables pre-multiplication for alpha blending in pixel units. 0: Disabled 1: Enabled
13	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
12	GR1_ARC_ON	0	R/W	Enables or disables alpha blending in rectangular area. 0: Disabled 1: Enabled
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	GR1_ARC_DISP_ON	0	R/W	Turns on/off frame-line display of the image area for alpha blending in rectangular area. 0: Frame-line display is turned off. 1: Frame-line display is turned on.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	GR1_GRC_DISP_ON	0	R/W	Turns on/off frame-line display of the graphics image area. 0: Frame-line display off 1: Frame-line display on
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1, 0	GR1_DISP_SEL[1:0]	0	R/W	<p>Selects the graphics display mode.</p> <p>0: Background color display (GR1_BASE)</p> <p>1: Lower-layer graphics display When displaying video image or enlarged graphics, select this setting.</p> <p>2: Current graphics display When displaying graphics, select this setting.</p> <p>3: Blended display of lower-layer graphics and current graphics*</p> <p>Note: * When cascaded connection is disabled (GR1_CUS_CON_ON = 0), select this setting whenever chroma-key processing is to proceed. Since only current graphics are to be displayed by chroma-key processing, set the <math>\alpha</math> values for both pixels to be subject to chroma-keying and pixels not to be subject to chroma-keying to 255.</p>

Note: GR1\_CUS\_CON\_ON is updated when GR1\_UPDATE bit in the graphics 1 register update control register (GR1\_UPDATE) is 1. The other bits are updated when GR1\_P\_VEN bit in the graphics 1 register update control register (GR1\_UPDATE) is 1.

### 33.2.120 Alpha Blending Control Register 2 (Graphics 1) (GR1\_AB2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR1_GRC_VS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR1_GRC_VW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR1_GRC_VS[10:0]	0	R/W	Vertical Start Position of Graphics Image Area. Note: The set value should be four or more (lines). GR1_GRC_VS + GR1_GRC_VW should be equal to or less than 2039 (lines).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR1_GRC_VW[10:0]	0	R/W	Vertical Width of Graphics Image Area.

Note: This register is updated when the GR1\_P\_VEN bit in the graphics 1 register update control register (GR1\_UPDATE) is 1.



## 33.2.121 Alpha Blending Control Register 3 (Graphics 1) (GR1\_AB3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR1_GRC_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR1_GRC_HW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR1_GRC_HS[10:0]	0	R/W	Horizontal Start Position of Graphics Image Area. Note: The set value should be 16 or more (clock cycles). GR1_GRC_HS + GR1_GRC_HW should be equal to or less than 2015 (clock cycles).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR1_GRC_HW[10:0]	0	R/W	Horizontal Width of Graphics Image Area. Note: For displaying an image with 1- or 2-pixel horizontal width, set GR1_HW to 2 and GR1_GRC_HW to 1 (1 pixel) or 2 (2 pixels).

Note: This register is updated when the GR1\_P\_VEN bit in the graphics 1 register update control register (GR1\_UPDATE) is 1.

## 33.2.122 Alpha Blending Control Register 4 (Graphics 1) (GR1\_AB4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR1_ARC_VS[10:0]										
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR1_ARC_VW[10:0]										
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR1_ARC_VS[10:0]	0	R/W	Sets the vertical start position of the valid image area for alpha blending in a rectangular area.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR1_ARC_VW[10:0]	0	R/W	Sets the vertical width of the valid image area for alpha blending in a rectangular area.

Note: This register is updated when GR1\_P\_VEN bit in the graphics 1 register update control register (GR1\_UPDATE) is 1.

## 33.2.123 Alpha Blending Control Register 5 (Graphics 1) (GR1\_AB5)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR1_ARC_HS[10:0]										
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR1_ARC_HW[10:0]										
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR1_ARC_HS[10:0]	0	R/W	Sets the horizontal start position of the valid image area for alpha blending in a rectangular area.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR1_ARC_HW[10:0]	0	R/W	Sets the horizontal width of the valid image area for alpha blending in a rectangular area.

Note: This register is updated when GR1\_P\_VEN bit in the graphics 1 register update control register (GR1\_UPDATE) is 1.

## 33.2.124 Alpha Blending Control Register 6 (Graphics 1) (GR1\_AB6)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	GR1_ARC_MODE	GR1_ARC_COEF[7:0]							
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	GR1_ARC_RATE[7:0]							
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	GR1_ARC_MODE	0	R/W	Alpha Blending Mode in Rectangular Area 0: Addition 1: Subtraction
23 to 16	GR1_ARC_COEF[7:0]	0	R/W	Alpha Coefficient for Alpha Blending in Rectangular Area (0 to 255) [7:0]: Variation (absolute value)
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	GR1_ARC_RATE[7:0]	0	R/W	Frame Rate for Alpha Blending in Rectangular Area

Note: This register is updated when GR1\_P\_VEN bit in the graphics 1 register update control register (GR1\_UPDATE) is 1.

33.2.125 Alpha Blending Control Register 7 (Graphics 1) (GR1\_AB7)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	GR1_ARC_DEF[7:0]							
Initial value:	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR1_CK_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	GR1_ARC_DEF[7:0]	255	R/W	Initial Alpha Value for Alpha Blending in Rectangular Area
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR1_CK_ON	0	R/W	CLUT-Index/RGB-Index Chroma-Key Processing On/Off 0: Off 1: On

Note: This register is updated when the GR1\_P\_VEN bit in the graphics 1 register update control register (GR1\_UPDATE) is 1.

## 33.2.126 Alpha Blending Control Register 8 (Graphics 1) (GR1\_AB8)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR1_CK_KCLUT[7:0]								GR1_CK_KG[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR1_CK_KB[7:0]								GR1_CK_KR[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GR1_CK_KCLUT[7:0]	0	R/W	CLUT Signal for CLUT-Index Chroma-Key Processing CLUT: Unsigned 8 bits (0 to 255 [LSB])
23 to 16	GR1_CK_KG[7:0]	0	R/W	G Signal for RGB-Index Chroma-Key Processing G: Unsigned 8 bits (0 to 255 [LSB])
15 to 8	GR1_CK_KB[7:0]	0	R/W	B Signal for RGB-Index Chroma-Key Processing B: Unsigned 8 bits (0 to 255 [LSB])
7 to 0	GR1_CK_KR[7:0]	0	R/W	R Signal for RGB-Index Chroma-Key Processing R: Unsigned 8 bits (0 to 255 [LSB])

Note: This register is updated when the GR1\_P\_VEN bit in the graphics 1 register update control register (GR1\_UPDATE) is 1.

## 33.2.127 Alpha Blending Control Register 9 (Graphics 1) (GR1\_AB9)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR1_CK_A[7:0]								GR1_CK_G[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR1_CK_B[7:0]								GR1_CK_R[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GR1_CK_A [7:0]	0	R/W	Replaced Alpha Signal after RGB-Index Chroma-Key Processing $\alpha$ : Unsigned 8 bits (0 to 255 [LSB]) Note: These bits should always be set to 255 to display the current graphics only when cascaded connection is disabled.
23 to 16	GR1_CK_G [7:0]	0	R/W	Replaced G Signal after RGB-Index Chroma-Key Processing G: Unsigned 8 bits (0 to 255 [LSB])
15 to 8	GR1_CK_B [7:0]	0	R/W	Replaced B Signal after RGB-Index Chroma-Key Processing B: Unsigned 8 bits (0 to 255 [LSB])
7 to 0	GR1_CK_R [7:0]	0	R/W	Replaced R Signal after RGB-Index Chroma-Key Processing R: Unsigned 8 bits (0 to 255 [LSB])

Note: This register is updated when the GR1\_P\_VEN bit in the graphics 1 register update control register (GR1\_UPDATE) is 1.

## 33.2.128 Alpha Blending Control Register 10 (Graphics 1) (GR1\_AB10)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR1_A0[7:0]								GR1_G0[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR1_B0[7:0]								GR1_R0[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GR1_A0 [7:0]	0	R/W	CLUT1 $\alpha$ 0 Signal Replaced with $\alpha$ signal when in the CLUT1 format and CLUT1 = 0. Replaced with $\alpha$ signal when in the $\alpha$ RGB1555/RGB $\alpha$ 5551 format and $\alpha$ = 0. Note: These bits should always be set to 255 to display the current graphics only when cascaded connection is disabled.
23 to 16	GR1_G0 [7:0]	0	R/W	CLUT1 G0 Signal Replaced with G signal when in the CLUT1 format and CLUT1 = 0.
15 to 8	GR1_B0 [7:0]	0	R/W	CLUT1 B0 Signal Replaced with B signal when in the CLUT1 format and CLUT1 = 0.
7 to 0	GR1_R0 [7:0]	0	R/W	CLUT1 R0 Signal Replaced with R signal when in the CLUT1 format and CLUT1 = 0.

Note: This register is updated when the GR1\_P\_VEN bit in the graphics 1 register update control register (GR1\_UPDATE) is 1.

## 33.2.129 Alpha Blending Control Register 11 (Graphics 1) (GR1\_AB11)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR1_A1[7:0]								GR1_G1[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR1_B1[7:0]								GR1_R1[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GR1_A1 [7:0]	0	R/W	CLUT1 $\alpha$ 1 Signal Replaced with $\alpha$ signal when in the CLUT1 format and CLUT1 = 1. Replaced with $\alpha$ signal when in the $\alpha$ RGB1555/RGB $\alpha$ 5551 format and $\alpha$ = 1. Note: These bits should always be set to 255 to display the current graphics only when cascaded connection is disabled.
23 to 16	GR1_G1 [7:0]	0	R/W	CLUT1 G1 Signal Replaced with G signal when in the CLUT1 format and CLUT1 = 1.
15 to 8	GR1_B1 [7:0]	0	R/W	CLUT1 B1 Signal Replaced with B signal when in the CLUT1 format and CLUT1 = 1.
7 to 0	GR1_R1 [7:0]	0	R/W	CLUT1 R1 Signal Replaced with R signal when in the CLUT1 format and CLUT1 = 1.

Note: This register is updated when the GR1\_P\_VEN bit in the graphics 1 register update control register (GR1\_UPDATE) is 1.

## 33.2.130 Background Color Control Register (Graphics 1) (GR1\_BASE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	GR1_BASE_G[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR1_BASE_B[7:0]								GR1_BASE_R[7:0]							
Initial value:	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	GR1_BASE_ G[7:0]	0	R/W	Background Color G/Y Signal G/Y: Unsigned 8 bits (0 to 255 [LSB])
15 to 8	GR1_BASE_ B[7:0]	128	R/W	Background Color B/Cb Signal B: Unsigned 8 bits (0 to 255 [LSB]) Cb: 8 bits; 128 offset binary; unsigned (0 to 255 [LSB])

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	GR1_BASE_R[7:0]	128	R/W	Background Color R/Cr Signal R: Unsigned 8 bits (0 to 255 [LSB]) Cr: 8 bits; 128 offset binary; unsigned (0 to 255 [LSB])

Note: This register is updated when the GR1\_P\_VEN bit in the graphics 1 register update control register (GR1\_UPDATE) is 1.

### 33.2.131 CLUT Table Control Register (Graphics 1) (GR1\_CLUT)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR1_CLT_SEL
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	GR1_CLT_SEL	0	R/W	CLUT Table Select Signal 0: Selects CLUT table 0. Referring to the CLUT table 0 value to expand to αRGB8888 The CPU side can read-access or write-access to the CLUT table 1. 1: Selects CLUT table 1. Referring to the CLUT table 1 value to expand to αRGB8888 The CPU side can read-access or write-access to the CLUT table 0.
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note: This register is updated when the GR1\_P\_VEN bit in the graphics 1 register update control register (GR1\_UPDATE) is 1.

33.2.132 Status Monitor Register (Graphics 1) (GR1\_MON)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR1_ARC_ST
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR1_ARC_ST	0	R	Status Flag for Alpha Blending in Rectangular Area 0: Addition or subtraction has been completed. ( $\alpha$ value is 0 or 255) 1: Addition or subtraction is in progress.



### 33.3 Usage Method

#### 33.3.1 Scaling Setting Example for 525i Video Input and VGA-Size (640 x 480) Video Output

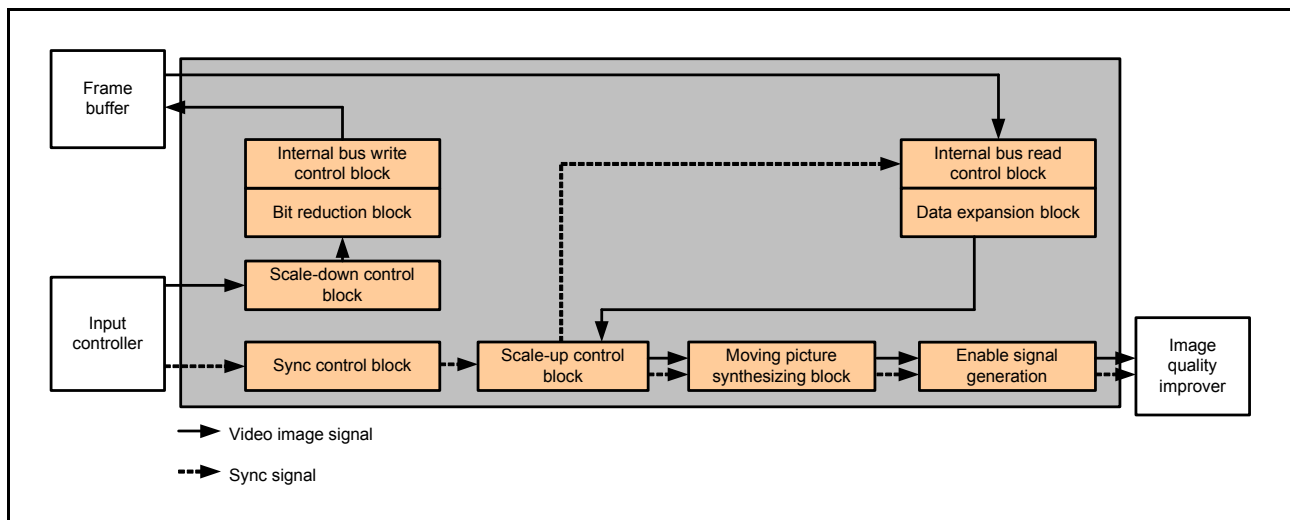
##### (1) Angles of View for Input and Output

This section describes an example of setting the signals of the input and output angles of view shown in Table 33.43.

Here, the over-scan rate is assumed to be 100%.

**Table 33.43 Input and Output Angles for 525i Video Input and VGA-Size (640 x 480) Video Output**

Input Signal	Output Signal	Signal Format	Rotation	Buffer Planes	Scaling Filter
1440 × 240	640 × 480	YCbCr	Normal	Two planes	2-tap linear



**Figure 33.19 Signal Paths for Displaying Input Video Image**

##### (2) Horizontal Scaling (Horizontal Scale Down, Scaling Filter: 2-Tap Linear)

The scaling rate for folding can be calculated as shown below.

$$\text{RATIO\_org} = \text{round}(1440 \div 640 \times 4096) = 9216$$

$$\sigma = (9216 \times (640 - 1) - (1440 - 1) \times 4096) \div (640 - 1) = -8.01$$

$$\text{Horizontal scaling ratio} = \text{roundup}(9216 - (-8.01)) = 9225$$

##### (3) Vertical Scaling (Vertical Scale Up, Scaling Filter: 2-Tap Linear)

The scaling rate for folding can be calculated as shown below.

$$\text{RATIO\_org} = \text{round}(240 \div 480 \times 4096) = 2048$$

$$\sigma = (2048 \times (480 - 1) - (240 - 1) \times 4096) \div (480 - 1) = 4.27$$

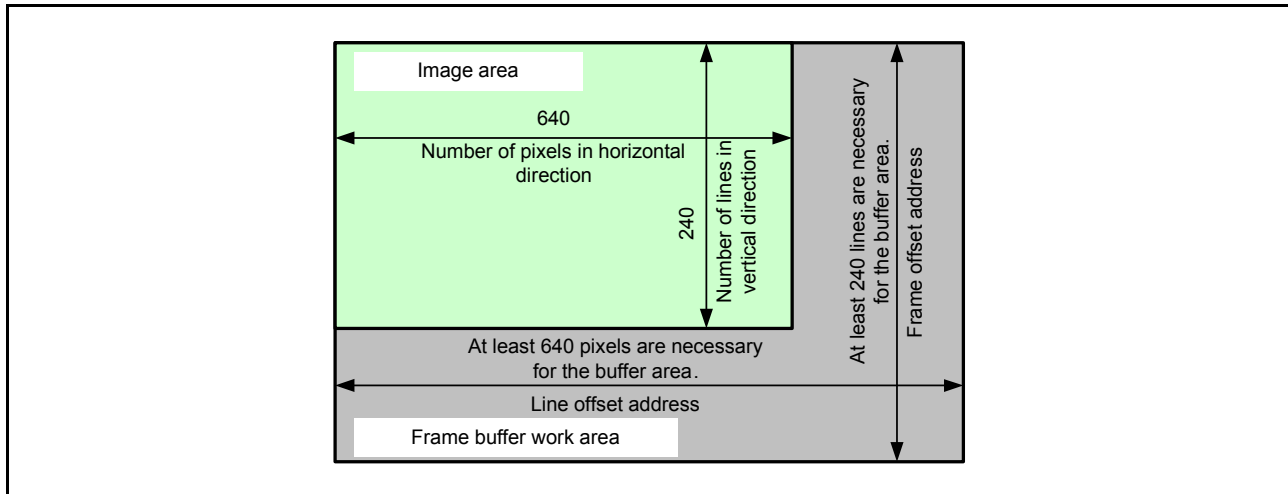
$$\text{Vertical scaling ratio} = \text{round}(2048 - (4.07)) = 2044$$

#### (4) Setting Frame Buffer Access Area

Since video data is written to the frame buffer after scaled down, the write size is  $640 \times 240$  pixels.

The frame buffer area required is 640 pixels or more for line offset and line offset  $\times$  240 pixels or more for frame offset.

Here, the frame buffer work area is assumed to be  $1024 \times 256$  pixels.



**Figure 33.20** Frame Buffer Access Area Setting

Since the frame buffer is accessed in 64-bit units, YCbCr422 (16 bits) is accessed in 4-pixel units.

The line offset address values to be set are:

$$SC\_RES\_LN\_OFF[14:0] = 1024 \times 2 = 2048$$

$$GR\_LN\_OFF[14:0] = 1024 \times 2 = 2048$$

The frame offset address values to be set are:

$$SC\_RES\_FLM\_OFF[22:0] = SC\_RES\_LN\_OFF[14:0] \times 256 = 524288$$

$$GR\_FLM\_OFF[22:0] = GR\_LN\_OFF[14:0] \times 256 = 524288$$

## (5) Register Setting Example

**Table 33.44 Register Setting Example for 525i Video Input and VGA-Size Video Output**

Register Name	Bit Name	Settings	Remarks
<b>Synchronization Control</b>			
SC_SCL0_FRC3	SC_RES_VS_SEL	0	External Vsync selected
SC_SCL0_FRC4	SC_RES_FH[10:0]	799	Horizontal period width of output signal (period width = set value + 1)
<b>Size of Angle of View</b>			
SC_SCL0_DS2	SC_RES_VS[10:0]	15	Vertical capture start position of input signal
SC_SCL0_DS2	SC_RES_VW[10:0]	240	Vertical capture width of input signal
SC_SCL0_DS3	SC_RES_HS[10:0]	244	Horizontal capture start position of input signal
SC_SCL0_DS3	SC_RES_HW[10:0]	1440	Horizontal capture width of input signal
SC_SCL0_FRC6	SC_RES_F_VS[10:0]	35	Vertical valid start position of full screen
SC_SCL0_FRC6	SC_RES_F_VW[10:0]	480	Vertical valid width of full screen
SC_SCL0_FRC7	SC_RES_F_HS[10:0]	144	Horizontal valid start position of full screen
SC_SCL0_FRC7	SC_RES_F_HW[10:0]	640	Horizontal valid width of full screen
SC_SCL0_US2	SC_RES_P_VS[10:0]	35	Vertical valid start position of output image
SC_SCL0_US2	SC_RES_P_VW[10:0]	480	Vertical valid width of output image
SC_SCL0_US3	SC_RES_P_HS[10:0]	144	Horizontal valid start position of output image
SC_SCL0_US3	SC_RES_P_HW[10:0]	640	Horizontal valid width of output image
<b>Scaling Setting</b>			
SC_SCL0_DS4	SC_RES_DS_H_RATIO [15:0]	9224	Horizontal scaling-down because SC_RES_DS_H_RATIO is equal to or larger than 4096
SC_SCL0_DS1	SC_RES_DS_H_ON	1	Horizontal scaling-down on
SC_SCL0_US1	SC_RES_US_H_ON	0	Horizontal scaling-up off
SC_SCL0_US5	SC_RES_US_H_RATIO [15:0]	4096	Horizontal scaling-up off because SC_RES_US_H_RATIO is equal to or larger than 4096
SC_SCL0_DS1	SC_RES_DS_V_ON	0	Vertical scaling-down off
SC_SCL0_US1	SC_RES_US_V_ON	1	Vertical scaling-up on
SC_SCL0_DS6	SC_RES_V_RATIO[15:0]	2044	Vertical scaling-up because SC_RES_V_RATIO is smaller than 4096
SC_SCL0_DS7	SC_RES_OUT_VW[10:0]	240	Vertical valid input width because the vertical scaling-down function is off
SC_SCL0_DS7	SC_RES_OUT_HW[10:0]	640	Horizontal image size after horizontal scaling-down
SC_SCL0_US4	SC_RES_IN_VW[10:0]	240	Vertical width of frame buffer read
SC_SCL0_US4	SC_RES_IN_HW[10:0]	640	Horizontal width of frame buffer read
<b>IP Conversion Setting</b>			
SC_SCL0_DS5	SC_RES_TOP_INIPHASE[11:0]	2048	Top field adjusted by 0.5-line phase
SC_SCL0_DS5	SC_RES_BTM_INIPHASE[11:0]	0	No phase adjustment for bottom field
SC_SCL0_FRC5	SC_RES_FLD_DLY_SEL	1	IP conversion with two planes of frame buffer used for vertical scaling-up
<b>Frame Buffer Write Setting</b>			
SC_SCL1_WR1	SC_RES_DS_WR_MD [2:0]	0	Normal write mode for rotation control
SC_SCL1_WR1	SC_RES_MD[1:0]	0	Frame buffer write format YCbCr422 (16 bits)
SC_SCL1_WR2	SC_RES_BASE[31:0]	0	Frame buffer write start address (0 in setting example)
SC_SCL1_WR3	SC_RES_LN_OFF[14:0]	2048	Frame buffer write line offset

**Table 33.44 Register Setting Example for 525i Video Input and VGA-Size Video Output**

Register Name	Bit Name	Settings	Remarks
SC_SCL1_WR3	SC_RES_FLM_NUM[9:0]	1	Two planes of frame buffer used
SC_SCL1_WR4	SC_RES_FLN_OFF[22:0]	524288	Frame buffer write frame offset
SC_SCL1_WR5	SC_RES_WENB	1	Frame buffer write enabled
<b>Frame Buffer Read Setting</b>			
GR_FLM1	GR_FLM_SEL[1:0]	0	Frame number for frame buffer write output
GR_FLM2	GR_BASE[31:0]	0	Conforming to frame buffer write setting
GR_FLM3	GR_LN_OFF[14:0]	2048	Conforming to frame buffer write setting
GR_FLM4	GR_FLM_OFF[22:0]	524288	Conforming to frame buffer write setting
GR_FLM6	GR_FORMAT[3:0]	8	Frame buffer read format YCbCr422
GR_FLM_RD	GR_R_ENB	1	Frame buffer read enabled
GR_FLM6	GR_CNV444_MD	1	Mean value interpolation in YCbCr422 → YCbCr444 conversion
<b>Scaling-up Selection</b>			
SC_SCL0_US8	SC_RES_IBUS_SYNC_SEL	0	Scaled-up video signal displayed
GR_AB1	GR_DISP_SEL[1:0]	1	Scaling display selected

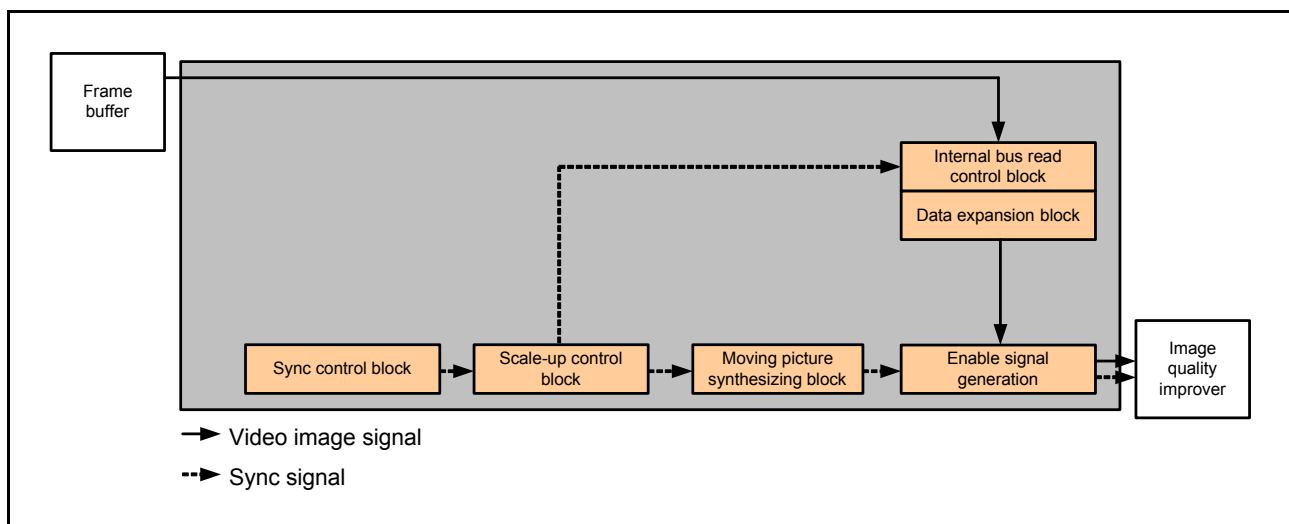
### 33.3.2 Scaling Setting Example for Graphics Display

#### (1) Angle of View for Graphics Display

This section describes an example of setting the signals of the input and output angles of view shown in Table 33.45.

**Table 33.45 Input and Output Angles of View for Graphics Display**

Graphics Size	Output Signal	Graphics Signal Format
640 × 480	640 × 480	RGB888

**Figure 33.21 Signal Paths for Graphics Display**

## (2) Setting Frame Buffer Access Area

In the frame buffer in which graphics data is stored, graphics data needs to be expanded in the area of  $640 \times 480$  pixels or larger.

Here, the frame buffer area in which graphics data is expanded is assumed to be  $640 \times 480$  pixels.

Since the frame buffer is accessed in 64-bit units, RGB888 (32 bits) is accessed in 2-pixel units.

The line offset address values to be set are:

$$\text{GR\_LN\_OFF}[14:0] = 640 \times 4 = 2560$$

The frame offset address values to be set are:

$$\text{GR\_FLM\_OFF}[22:0] = \text{GR\_LN\_OFF}[14:0] \times 480 = 1228800$$

## (3) Register Setting Example

**Table 33.46 Register Setting Example for Graphics Display**

Register Name	Bit Name	Settings	Remarks
<b>Synchronization Control</b>			
SC_SCL0_FRC3	SC_RES_VS_SEL	1	Free-running Vsync selected (when the appropriate input signal is available, an external sync can also be selected)
SC_SCL0_FRC4	SC_RES_FV[10:0]	524	Vertical period width of output signal (period width = set value + 1)
SC_SCL0_FRC4	SC_RES_FH[10:0]	799	Horizontal period width of output signal (period width = set value + 1)
<b>Size of Angle of View</b>			
SC_SCL0_FRC6	SC_RES_F_VS[10:0]	35	Vertical valid start position of full screen
SC_SCL0_FRC6	SC_RES_F_VW[10:0]	480	Vertical valid width of full screen
SC_SCL0_FRC7	SC_RES_F_HS[10:0]	144	Horizontal valid start position of full screen
SC_SCL0_FRC7	SC_RES_F_HW[10:0]	640	Horizontal valid width of full screen
GR_AB2	GR_GRC_VS[10:0]	35	Vertical valid start position of graphics output
GR_AB2	GR_GRC_VW[10:0]	480	Vertical valid width of graphics output
GR_AB3	GR_GRC_HS[10:0]	144	Horizontal valid start position of graphics output
GR_AB3	GR_GRC_HW[10:0]	640	Horizontal valid width of graphics output
<b>Frame Buffer Read Setting</b>			
GR_FLM1	GR_FLM_SEL[1:0]	1	Frame number setting with register
GR_FLM3	GR_FLM_NUM[9:0]	0	Frame number of frame buffer (0 in setting example)
GR_FLM5	GR_FLM_LNUM[9:0]	479	Number of graphics lines (number of lines = set value + 1)
GR_FLM6	GR_HW[9:0]	639	Horizontal valid width of graphics (valid width = set value + 1)
GR_FLM2	GR_BASE[31:0]	0	Conforming to graphics expansion setting (0 in setting example)
GR_FLM3	GR_LN_OFF[14:0]	2560	Conforming to graphics expansion setting
<b>Frame Buffer Read Setting</b>			
GR_FLM4	GR_FLM_OFF[22:0]	1228800	Conforming to graphics expansion setting
GR_FLM6	GR_FORMAT[3:0]	1	Frame buffer read format RGB888
GR_FLM_RD	GR_R_ENB	1	Frame buffer read enabled

**Table 33.46 Register Setting Example for Graphics Display**

Register Name	Bit Name	Settings	Remarks
Scaling-up Selection			
SC_SCL0_US8	SC_RES_IBUS_SYNC_SEL	1	Graphics output displayed
GR_AB1	GR_DISP_SEL[1:0]	2	Graphics display selected

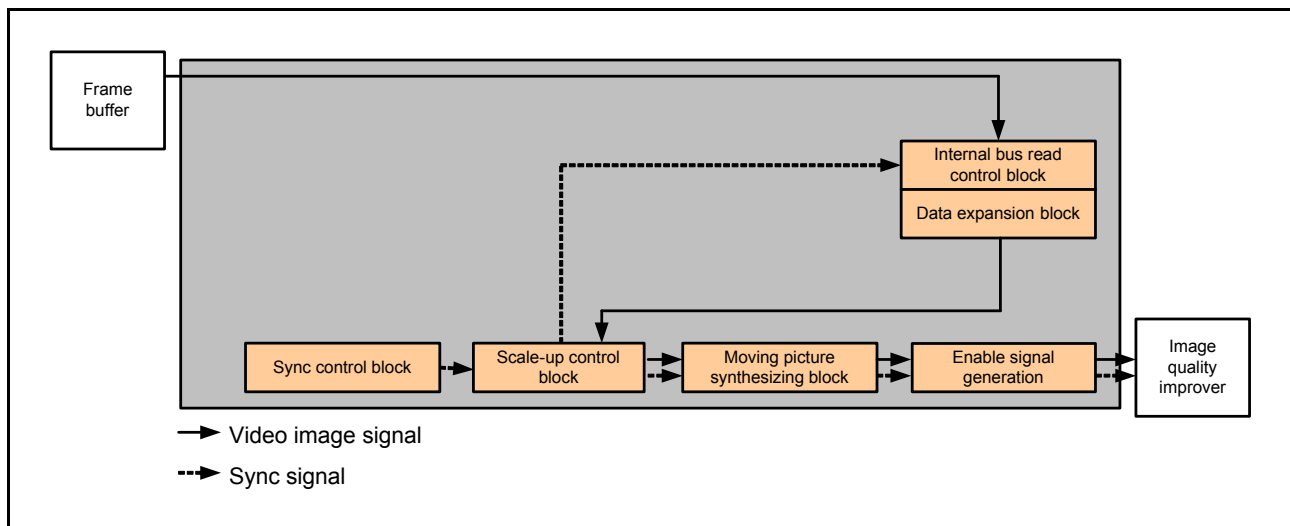
### 33.3.3 Scaling Setting Example for Scaled-up Graphics Display

#### (1) Angles of View for Input and Output

This section describes an example of setting the signals of the input and output angles of view shown in Table 33.47.

**Table 33.47 Input and Output Angles of View for Scaled-up Graphics Display**

Graphics Size	Output Signal	Graphics Signal Format
640 × 480	800 × 600	RGB565

**Figure 33.22 Signal Paths for Scaled-up Graphics Display**

#### (2) Horizontal Scaling (Horizontal Scale Up, Scaling Filter: 2-Tap Linear)

The scaling rate for folding can be calculated as shown below.

$$\text{RATIO\_org} = \text{round} (640 \div 800 \times 4096) = 3277$$

$$\sigma = (3277 \times (800 - 1) - (640 - 1) \times 4096) \div (800 - 1) = 1.23$$

$$\text{Horizontal scaling ratio} = \text{round} (3277 - (1.23)) = 3276$$

### (3) Vertical Scaling (Vertical Scale Up, Scaling Filter: 2-Tap Linear)

The scaling rate for folding can be calculated as shown below.

$$\text{RATIO\_org} = \text{round} (480 \div 600 \times 4096) = 3277$$

$$\sigma = (3277 \times (600 - 1) - (480 - 1) \times 4096) \div (600 - 1) = 1.57$$

$$\text{Vertical scaling ratio} = \text{round} (3277 - (1.57)) = 3275$$

### (4) Setting Frame Buffer Access Area

In the frame buffer in which graphics data is stored, graphics data needs to be expanded in the area of  $640 \times 480$  pixels or larger.

Here, the frame buffer area in which graphics data is expanded is assumed to be  $640 \times 480$  pixels.

Since the frame buffer is accessed in 64-bit units, RGB565 (16 bits) is accessed in 4-pixel units.

The line offset address values to be set are:

$$\text{GR\_LN\_OFF}[14:0] = 640 \times 2 = 1280$$

The frame offset address values to be set are:

$$\text{GR\_FLM\_OFF}[22:0] = \text{GR\_LN\_OFF}[14:0] \times 480 = 614400$$

## (5) Register Setting Example

**Table 33.48 Register Setting Example for Scaled-up Graphics Display**

Register Name	Bit Name	Settings	Remarks
<b>Synchronization Control</b>			
SC_SCL0_FRC3	SC_RES_VS_SEL	1	Free-running Vsync selected (when the appropriate input signal is available, an external sync can also be selected)
SC_SCL0_FRC4	SC_RES_FV[10:0]	668	Vertical period width of output signal (period width = set value + 1)
SC_SCL0_FRC4	SC_RES_FH[10:0]	1040	Horizontal period width of output signal (period width = set value + 1)
<b>Size of Angle of View</b>			
SC_SCL0_FRC6	SC_RES_F_VS[10:0]	27	Vertical valid start position of full screen
SC_SCL0_FRC6	SC_RES_F_VW[10:0]	600	Vertical valid width of full screen
SC_SCL0_FRC7	SC_RES_F_HS[10:0]	216	Horizontal valid start position of full screen
SC_SCL0_FRC7	SC_RES_F_HW[10:0]	800	Horizontal valid width of full screen
SC_SCL0_US2	SC_RES_P_VS[10:0]	27	Vertical valid start position of image output
SC_SCL0_US2	SC_RES_P_VW[10:0]	600	Vertical valid width of image output
SC_SCL0_US3	SC_RES_P_HS[10:0]	216	Horizontal valid start position of image output
SC_SCL0_US3	SC_RES_P_HW[10:0]	800	Horizontal valid width of image output
<b>Scaling Setting</b>			
SC_SCL0_US5	SC_RES_US_H_RATIO[15:0]	3276	Horizontal scaling-up because SC_RES_US_H_RATIO is smaller than 4096
SC_SCL0_DS6	SC_RES_V_RATIO[15:0]	3275	Vertical scaling-up because SC_RES_V_RATIO is smaller than 4096
SC_SCL0_US1	SC_RES_US_H_ON	1	Horizontal scaling-up on
SC_SCL0_US1	SC_RES_US_V_ON	1	Vertical scaling-up on
SC_SCL0_US4	SC_RES_IN_VW[10:0]	480	Vertical width of frame buffer read
SC_SCL0_US4	SC_RES_IN_HW[10:0]	640	Horizontal width of frame buffer read
<b>Frame Buffer Read Setting</b>			
GR_FLM1	GR_FLM_SEL[1:0]	1	Frame number setting with register
GR_FLM3	GR_FLM_NUM[9:0]	0	Frame number of frame buffer (0 in setting example)
GR_FLM2	GR_BASE[31:0]	0	Conforming to graphics expansion setting (0 in setting example)
GR_FLM3	GR_LN_OFF[14:0]	1280	Conforming to graphics expansion setting
GR_FLM4	GR_FLM_OFF[22:0]	614400	Conforming to graphics expansion setting
GR_FLM6	GR_FORMAT[3:0]	0	Frame buffer read format RGB565
GR_FLM_RD	GR_R_ENB	1	Frame buffer read enabled
<b>Scaling-up Selection</b>			
SC_SCL0_US8	SC_RES_IBUS_SYNC_SEL	0	Scaled-up video signal displayed
GR_AB1	GR_DISP_SEL[1:0]	1	Scaling display selected



## 34. Video Display Controller 5 (4): Image Quality Improver

### 34.1 Image Quality Improver

#### 34.1.1 Overview of Functions

The image quality improver subjects scaled YCbCr signals to black stretching, LTI/sharpness processing, and GBR conversion by using a color matrix.

The image quality improver does not act on RGB signals.

Figure 34.1 is a functional block diagram of the image quality improver. Image quality improvers 0 and 1 are connected to scalers 0 and 1, respectively.

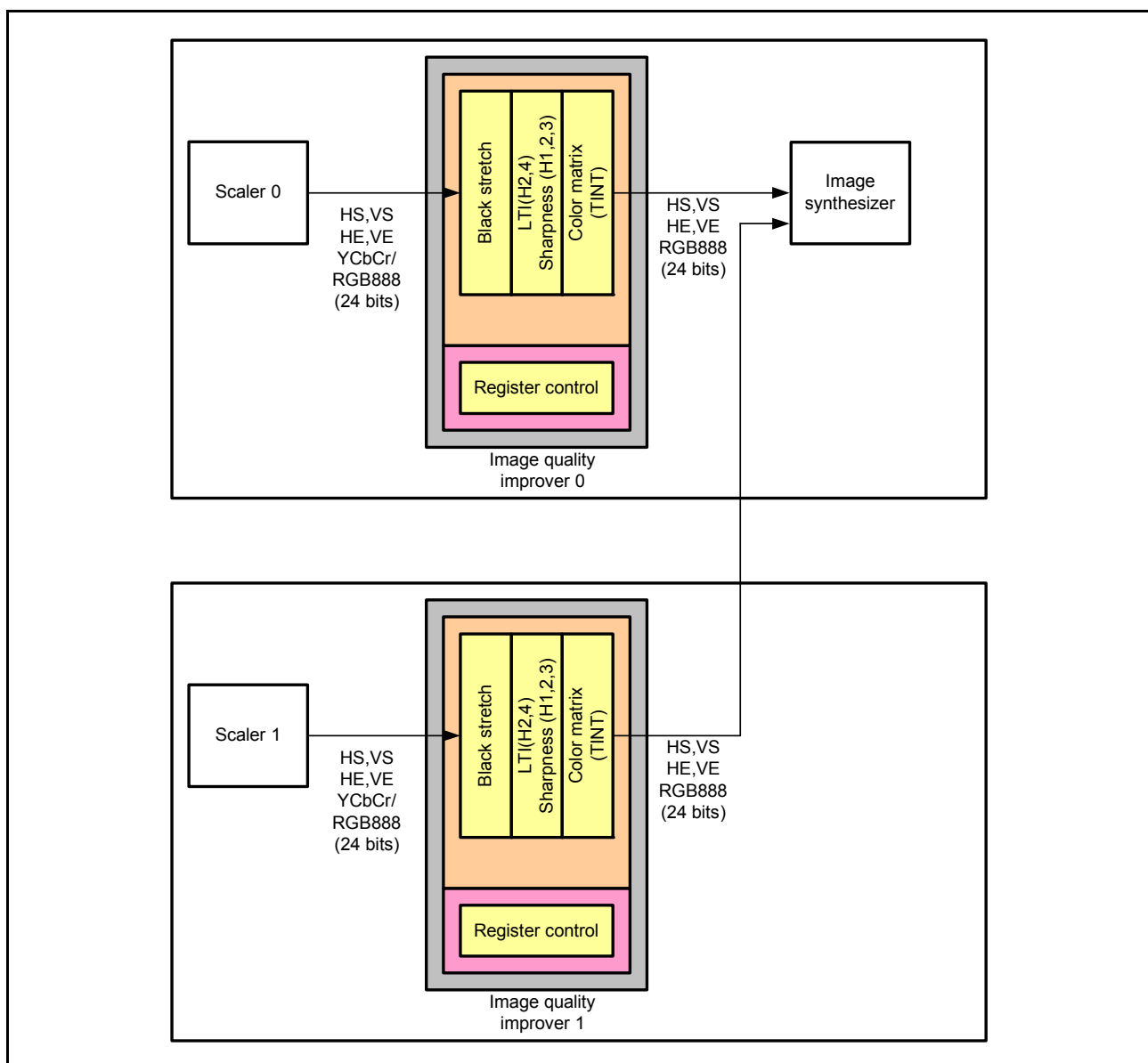


Figure 34.1 Functional Block Diagram of Image Quality Improver

### 34.1.2 Register Update Control

The control register for image quality improver controls the update timing entirely by vertical synchronous signals.

The vertical synchronous signal launched after the update control register is set to 1 is reflected in various registers, following which the update control register is automatically cleared to 0.

Note that there are two image quality improvers and registers for each improver can be identified by the number in the register name like ADJ0\_xxxx and ADJ1\_xxxx, respectively. In the sections except for Register Description, however, the number is omitted like ADJ\_xxxx for convenience sake.

**Table 34.1 Register Update Control**

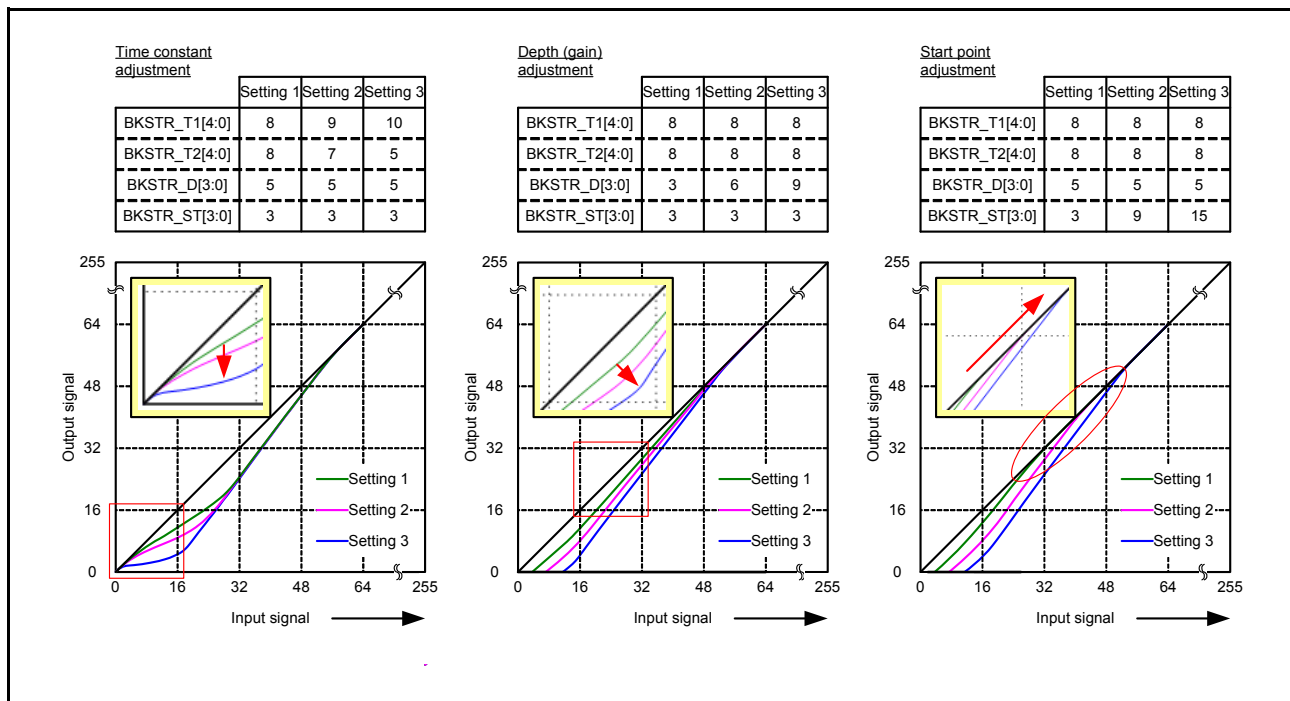
Register Name	Bit Name	Initial Value	Description
ADJ_UPDATE	ADJ_VEN	0	Image Quality Improver Register Update 0: Register is not updated. 1: Register is updated by launch of vertical synchronous signal.

### 34.1.3 Black Stretch

Black stretch refers to the black stretch correction of the Y signal of the input video signal of YCbCr format.

Correction of the Y signal is done by adjusting the time constant, depth (gain), and start point.

Figure 34.2 is a drawing illustrating black stretch correction.



**Figure 34.2 Black Stretch Correction (With Sample Settings)**

**Table 34.2 Black Stretch Control**

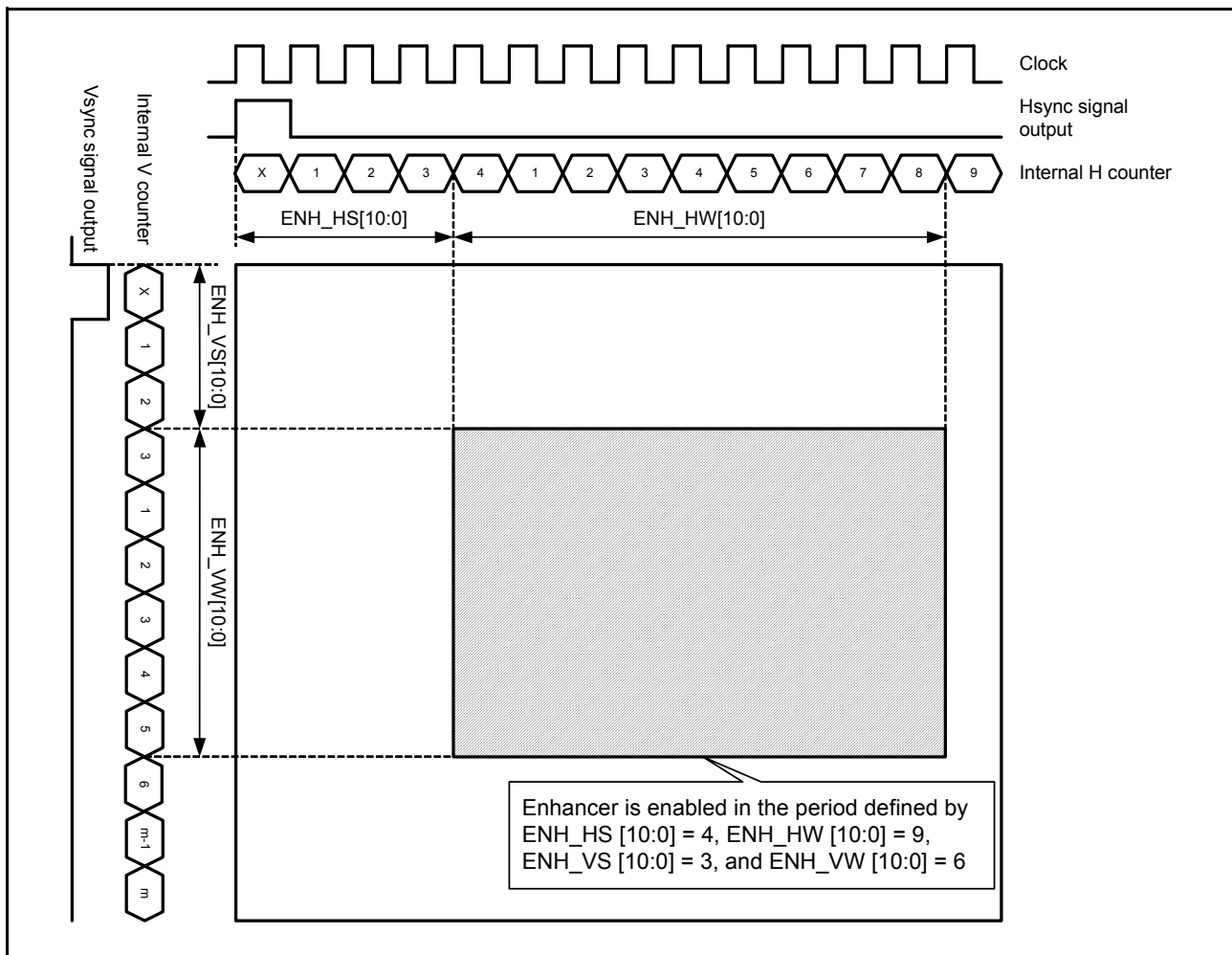
Register Name	Bit Name	Initial Value	Description
ADJ_BKSTR_SET	BKSTR_ON	0	Black Stretch On/Off Control 0: Black Stretch Off 1: Black Stretch On
ADJ_BKSTR_SET	BKSTR_ST[3:0]	0	Black Stretch Start Point 0 (low) to 15 (high)
ADJ_BKSTR_SET	BKSTR_T1[4:0]	0	Black Stretch Time Constant (T1) 0 (small) to 31 (large)
ADJ_BKSTR_SET	BKSTR_T2[4:0]	0	Black Stretch Time Constant (T2) 0 (small) to 30 (large), 31: Setting prohibited
ADJ_BKSTR_SET	BKSTR_D[3:0]	0	Black Stretch Depth 0 (shallow) to 15 (deep)

### 34.1.4 Enhancer

The enhancer subjects the scaled Y signal input to transient improvement (LTI) and sharpness processing in the horizontal direction.

#### (1) Enhancer Area Specification

The operating area of the enhancer is specified with reference to the rising edges of the Hsync signal and Vsync signal. ENH\_HS should be set to four or greater clocks, and ENH\_VS should be set to two or greater lines. Figure 34.3 shows enhancer area setting.



**Figure 34.3** Period when Enhancer is Enabled

Setting ENH\_DISP\_ON to 1 displays the enhancer-enabled area with frame lines.

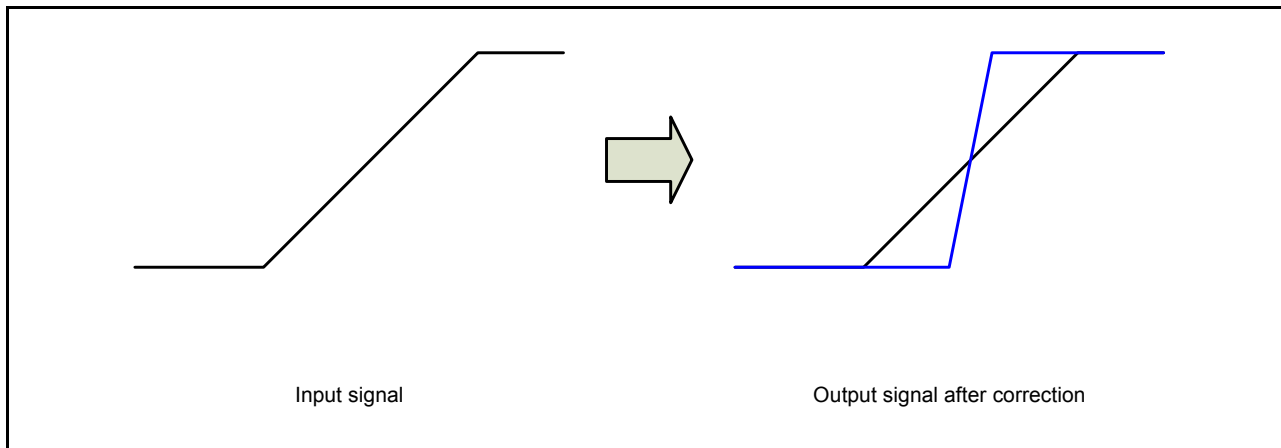
**Table 34.3** Enhancer Area Control

Register Name	Bit Name	Initial Value	Description
ADJ_ENH_TIM1	ENH_MD	1	Operating Mode 0: RGB mode 1: YCbCr mode
ADJ_ENH_TIM2	ENH_VS[10:0]	0	Start Position of Vertical Valid Image Area in Enhancer-Enabled Area Note: Set to 2 or greater lines.
ADJ_ENH_TIM2	ENH_VW[10:0]	0	Width of Vertical Valid Image Area in Enhancer-Enabled Area
ADJ_ENH_TIM3	ENH_HS[10:0]	0	Start Position of Horizontal Valid Image Area in Enhancer-Enabled Area Note: Set to 4 or greater clocks.
ADJ_ENH_TIM3	ENH_HW[10:0]	0	Width of Horizontal Valid Image Area in Enhancer-Enabled Area
ADJ_ENH_TIM1	ENH_DISP_ON	0	Frame Line Display in Enhancer-Enabled Area 0: Off 1: On

## (2) LTI (Luminance Transient Improvement)

The enhancer subjects the Y signal input to transient improvement in the horizontal direction.

Transient improvement of the blanking signal is turned off.



**Figure 34.4 LTI Correction**

After edge detection of the image, the LTI can be independently controlled in the two horizontal bands.

In LTI, the median filter is inserted after edge detection of the image.

In LTI (H4), the reference pixels of the median filter can be selected.

However, under normal operations, half the tap data (second adjacent pixel) at edge detection is used as reference.

**Table 34.4 Reference Pixel Table for LTI**

LTI Band	Reference Pixel for Edge Detection	LPF Application	Median Filter Reference Pixels
Horizontal LTI (H2)	Second adjacent pixel used as reference	LPF not applied or LPF (1,2,1)	Adjacent pixel used as reference
Horizontal LTI (H4)	Fourth adjacent pixel used as reference	LPF (1,2,1)	Adjacent pixel or second adjacent pixel used as reference

In LTI, the detection result can be subjected to a coring process.

The core value set in the register is subtracted from the edge detection result, and LTI correction is performed on the coring output after subtraction.

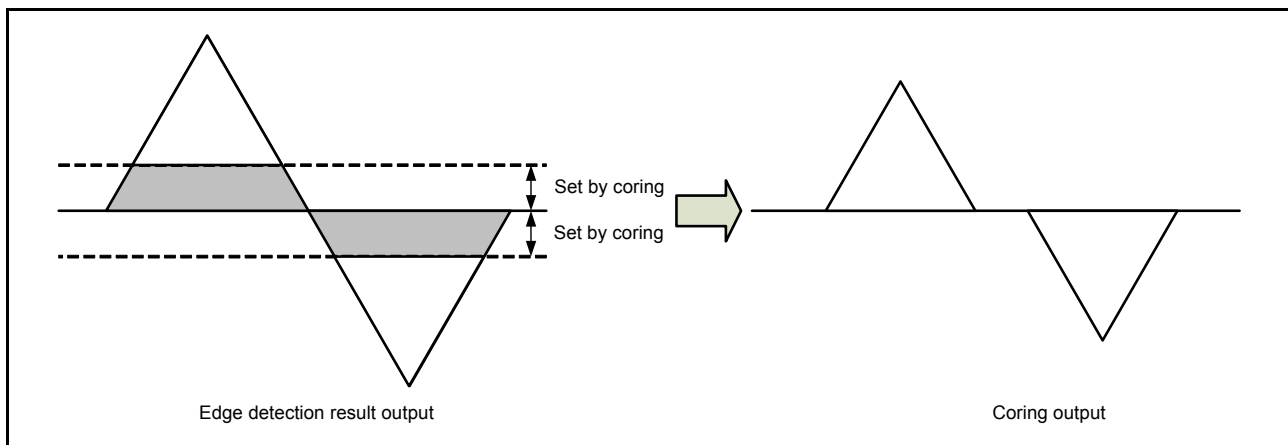


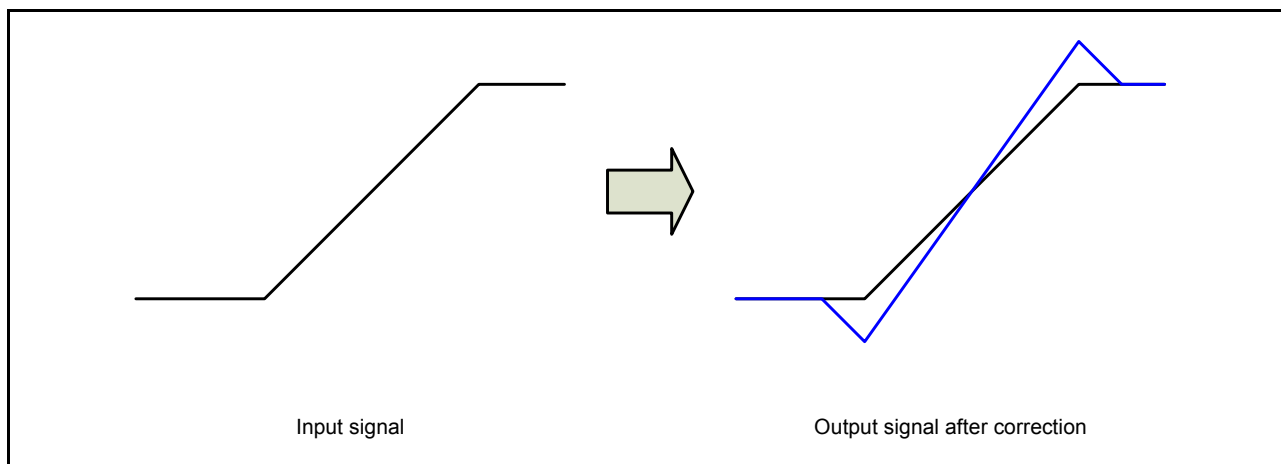
Figure 34.5 LTI Coring

Table 34.5 LTI Control

Register Name	Bit Name	Initial Value	Description
ADJ_ENH_LTI1	LTI_H_ON	0	LTI On/Off Control 0: LTI off 1: LTI on
ADJ_ENH_LTI1	LTI_H2_INC_ZERO [7:0]	10	Median Filter LTI Correction Threshold LTI correction is disabled when: $ \text{right TAP value} - \text{center TAP value}  < \text{LTI1\_H2\_INC\_ZERO}$ or $ \text{left TAP value} - \text{center TAP value}  < \text{LTI1\_H2\_INC\_ZERO}$ .
ADJ_ENH_LTI1	LTI_H2_LPF_SEL	0	LPF Selection for Folding Prevention Before H2 Edge Detection 0: LPF not selected 1: LPF selected
ADJ_ENH_LTI1	LTI_H2_GAIN[7:0]	0	LTI Edge Amplitude Value Gain 0 (0 times) to 64 (+1 times) to 255 (+ approx. 4 times)
ADJ_ENH_LTI1	LTI_H2_CORE[7:0]	0	LTI Coring (Maximum core value of 255) Amplitude smaller than or equal to the value of LTI_H2_CORE is cored from the edge amplitude value. (A core value setting of 128 remains unchanged.)
ADJ_ENH_LTI2	LTI_H4_INC_ZERO[7:0]	10	Median Filter LTI Correction Threshold LTI correction is disabled when: $ \text{right TAP value} - \text{center TAP value}  < \text{LTI1\_H4\_INC\_ZERO}$ or $ \text{left TAP value} - \text{center TAP value}  < \text{LTI1\_H4\_INC\_ZERO}$ .
ADJ_ENH_LTI2	LTI_H4_MEDIAN_TAP_SEL	0	Median Filter Reference Pixel Select 0: Second adjacent pixel selected as reference 1: Adjacent pixel selected as reference
ADJ_ENH_LTI2	LTI_H4_GAIN[7:0]	0	LTI Edge Amplitude Value Gain 0 (0 times) to 64 (+1 times) to 255 (+ approx. 4 times)
ADJ_ENH_LTI2	LTI_H4_CORE[7:0]	0	LTI Coring (Maximum core value of 255) Amplitude less than or equal to the value of LTI_H4_CORE is cored from the edge amplitude value. (A core value setting of 128 remains unchanged.)

### (3) Sharpness Process

The enhancer performs edge enhancement on the Y signal input by adding overshoot and undershoot to the original signal. Edge enhancement of the blanking signal is turned off.



**Figure 34.6 Sharpness Correction**

After edge detection of the image, the sharpness can be independently controlled in the three horizontal bands.

In horizontal sharpness, a 3-tap low-pass filter (LPF) is inserted before edge detection to prevent folding. The LPF can be turned on or off by register setting.

**Table 34.6 Reference Pixel Table for Sharpness**

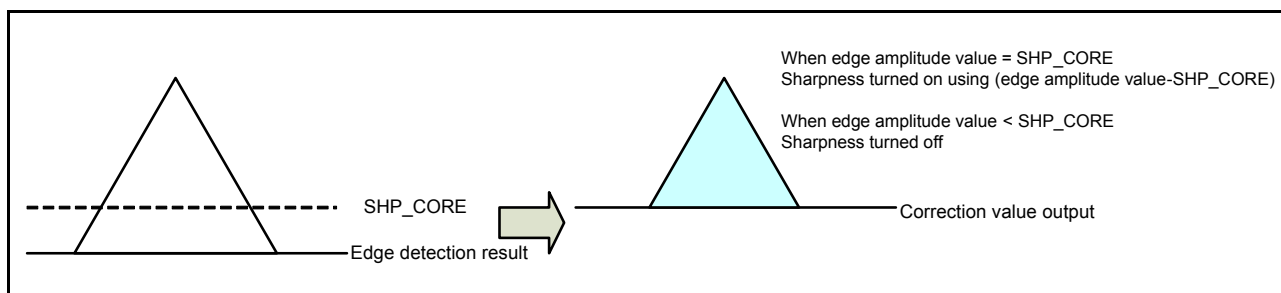
Sharpness Band	Reference Pixel for Edge Detection	LPF Application
Horizontal sharpness (H1)	Adjacent pixel used as reference	LPF not applied
Horizontal sharpness (H2)	Second adjacent pixel used as reference	LPF not applied or LPF (1,2,1)
Horizontal sharpness (H3)	Third adjacent pixel used as reference	LPF (1,2,1)

The edge amplitude of the edge to be enhanced is adjusted according to the value of SHP\_CORE.

Edge enhancement is accomplished when the edge detection result of the image is greater than the value of SHP\_CORE.

In edge enhancement, a correction value is output by multiplying (edge amplitude value - SHP\_CORE) by sharpness gain.

Sharpness is turned off when the edge detection result of the image is smaller than the value of SHP\_CORE.



**Figure 34.7 Sharpness Characteristics**

**Table 34.7 Sharpness Control**

Register Name	Bit Name	Initial Value	Description
ADJ_ENH_SHP1	SHP_H_ON	0	Sharpness On/Off Control 0: Horizontal sharpness off 1: Horizontal sharpness on
ADJ_ENH_SHP3	SHP_H2_LPF_SEL	0	LPF Selection for Folding Prevention Before H2 Edge Detection 0: LPF not selected 1: LPF selected
ADJ_ENH_SHP2	SHP_H1_CLIP_O[7:0]	0	Sharpness Correction Value Clipping (on the Overshoot Side) Correction value clipped according to SHP_H1_CLIP_O
ADJ_ENH_SHP2	SHP_H1_CLIP_U[7:0]	0	Sharpness Correction Value Clipping (on the Undershoot Side) Correction value clipped according to SHP_H1_CLIP_U
ADJ_ENH_SHP2	SHP_H1_GAIN_O[7:0]	0	Sharpness Edge Amplitude Value Gain (on the Overshoot Side) 0 (0 times) to 64 (+1 times) to 255 (+ approx. 4 times) Sharpness correction value = SHP_H1_GAIN_O × (edge amplitude value – SHP_H1_CORE)
ADJ_ENH_SHP2	SHP_H1_GAIN_U[7:0]	0	Sharpness Edge Amplitude Value Gain (on the Undershoot Side) 0 (0 times) to 64 (+1 times) to 255 (+ approx. 4 times) Sharpness correction value = SHP_H1_GAIN_U × (edge amplitude value – SHP_H1_CORE)
ADJ_ENH_SHP1	SHP_H1_CORE[6:0]	0	Active Sharpness Range Edge amplitude value ≥ SHP_H1_CORE: Sharpness processing on Edge amplitude value < SHP_H1_CORE: Sharpness processing off Sharpness processing is always on when the edge detection value is 128 or greater.
ADJ_ENH_SHP4	SHP_H2_CLIP_O[7:0]	0	Sharpness Correction Value Clipping (on the Overshoot Side) Correction value clipped according to SHP_H2_CLIP_O
ADJ_ENH_SHP4	SHP_H2_CLIP_U[7:0]	0	Sharpness Correction Value Clipping (on the Undershoot Side) Correction value clipped according to SHP_H2_CLIP_U
ADJ_ENH_SHP4	SHP_H2_GAIN_O[7:0]	0	Sharpness Edge Amplitude Value Gain (on the Overshoot Side) 0 (0 times) to 64 (+1 times) to 255 (+ approx. 4 times) Sharpness correction value = SHP_H2_GAIN_O × (edge amplitude value – SHP_H2_CORE)
ADJ_ENH_SHP4	SHP_H2_GAIN_U[7:0]	0	Sharpness Edge Amplitude Value Gain (on the Undershoot Side) 0 (0 times) to 64 (+1 times) to 255 (+ approx. 4 times) Sharpness correction value = SHP_H2_GAIN_U × (edge amplitude value – SHP_H2_CORE)
ADJ_ENH_SHP3	SHP_H2_CORE[6:0]	0	Active Sharpness Range Edge amplitude value ≥ SHP_H2_CORE: Sharpness processing on Edge amplitude value < SHP_H2_CORE: Sharpness processing off Sharpness processing is always on when the edge detection value is 128 or greater.
ADJ_ENH_SHP6	SHP_H3_CLIP_O[7:0]	0	Sharpness Correction Value Clipping (on the Overshoot Side) Correction value clipped according to SHP_H3_CLIP_O
ADJ_ENH_SHP6	SHP_H3_CLIP_U[7:0]	0	Sharpness Correction Value Clipping (on the Undershoot Side) Correction value clipped according to SHP_H3_CLIP_U
ADJ_ENH_SHP6	SHP_H3_GAIN_O[7:0]	0	Sharpness Edge Amplitude Value Gain (on the Overshoot Side) 0 (0 times) to 64 (+1 times) to 255 (+ approx. 4 times) Sharpness correction value = SHP_H3_GAIN_O × (edge amplitude value – SHP_H3_CORE)
ADJ_ENH_SHP6	SHP_H3_GAIN_U[7:0]	0	Sharpness Edge Amplitude Value Gain (on the Undershoot Side) 0 (0 times) to 64 (+1 times) to 255 (+ approx. 4 times) Sharpness correction value = SHP_H3_GAIN_U × (edge amplitude value – SHP_H3_CORE)
ADJ_ENH_SHP5	SHP_H3_CORE[6:0]	0	Active Sharpness Range Edge amplitude value ≥ SHP_H3_CORE: Sharpness processing on Edge amplitude value < SHP_H3_CORE: Sharpness processing off Sharpness processing is always on when the edge detection value is 128 or greater.



### 34.1.5 Color Matrix

Color matrix is performed by adjusting the offset of each input signal and nine-axis gain. This allows YCbCr to GBR conversion.

#### (1) GBR to GBR Conversion

$$YGIN\_A = YGIN + ADJ\_MTX\_YG - 128$$

$$CBBIN\_A = CBBIN + ADJ\_MTX\_B - 128$$

$$CRRIN\_A = CRRIN + ADJ\_MTX\_R - 128$$

$$YGOUT = (ADJ\_MTX\_GG \times YGIN\_A + ADJ\_MTX\_GB \times CBBIN\_A + ADJ\_MTX\_GR \times CRRIN\_A) \div 256$$

$$CBBOUT = (ADJ\_MTX\_BG \times YGIN\_A + ADJ\_MTX\_BB \times CBBIN\_A + ADJ\_MTX\_BR \times CRRIN\_A) \div 256$$

$$CRROUT = (ADJ\_MTX\_RG \times YGIN\_A + ADJ\_MTX\_RB \times CBBIN\_A + ADJ\_MTX\_RR \times CRRIN\_A) \div 256$$

#### (2) YCbCr to GBR Conversion

$$YGIN\_A = YGIN + ADJ\_MTX\_YG - 128$$

$$CBBIN\_A = CBBIN - 128$$

$$CRRIN\_A = CRRIN - 128$$

$$YGOUT = (ADJ\_MTX\_GG \times YGIN\_A + ADJ\_MTX\_GB \times CBBIN\_A + ADJ\_MTX\_GR \times CRRIN\_A) \div 256$$

$$CBBOUT = (ADJ\_MTX\_BG \times YGIN\_A + ADJ\_MTX\_BB \times CBBIN\_A + ADJ\_MTX\_BR \times CRRIN\_A) \div 256$$

$$CRROUT = (ADJ\_MTX\_RG \times YGIN\_A + ADJ\_MTX\_RB \times CBBIN\_A + ADJ\_MTX\_RR \times CRRIN\_A) \div 256$$

**Table 34.8 Matrix Coefficients (Standard Values) of SMPTE 293M**

	YGIN		CBBIN		CRRIN	
	Coefficient	Bit Setting	Coefficient	Bit Setting	Coefficient	Bit Setting
YGOUT	1.000	ADJ_MTX_GG = 256	-0.344	ADJ_MTX_GB = 1960	-0.714	ADJ_MTX_GR = 1865
CBBOUT	1.000	ADJ_MTX_BG = 256	1.772	ADJ_MTX_BB = 454	0.000	ADJ_MTX_BR = 0
CRROUT	1.000	ADJ_MTX_RG = 256	0.000	ADJ_MTX_RB = 0	1.402	ADJ_MTX_RR = 359

**Table 34.9 Color Matrix Control**

Register Name	Bit Name	Initial Value	Description
ADJ_MTX_MODE	ADJ_MTX_MD[1:0]	2	Operating Mode 0: GBR → GBR 1: Setting prohibited 2: YCbCr → GBR 3: Setting prohibited
ADJ_MTX_YG_ADJ0	ADJ_MTX_YG[7:0]	128	Y/G Signal Offset (DC) Adjustment Unsigned (0 (-128) to 128(0) to 255 (+127) [LSB])
ADJ_MTX_CBB_ADJ0	ADJ_MTX_B[7:0]	128	B Signal Offset (DC) Adjustment Unsigned (0 (-128) to 128 (0) to 255 (+127) [LSB])
ADJ_MTX_CRR_ADJ0	ADJ_MTX_R[7:0]	128	R Signal Offset (DC) Adjustment Unsigned (0 (-128) to 128 (0) to 255 (+127) [LSB])
ADJ_MTX_YG_ADJ0	ADJ_MTX_GG[10:0]	256	Gain Adjustment of Y/G Signal of G Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
ADJ_MTX_YG_ADJ1	ADJ_MTX_GB[10:0]	1960	Gain Adjustment of Cb/B Signal of G Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
ADJ_MTX_YG_ADJ1	ADJ_MTX_GR[10:0]	1865	Gain Adjustment of Cr/R Signal of G Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
ADJ_MTX_CBB_ADJ9	ADJ_MTX_BG[10:0]	256	Gain Adjustment of Y/G Signal of B Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
ADJ_MTX_CBB_ADJ1	ADJ_MTX_BB[10:0]	454	Gain adjustment of Cb/B signal of B Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
ADJ_MTX_CBB_ADJ1	ADJ_MTX_BR[10:0]	0	Gain Adjustment of Cr/R Signal of B Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
ADJ_MTX_CRR_ADJ0	ADJ_MTX_RG[10:0]	256	Gain Adjustment of Y/G Signal of R Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
ADJ_MTX_CRR_ADJ1	ADJ_MTX_RB[10:0]	0	Gain Adjustment of Cb/B Signal of R Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
ADJ_MTX_CRR_ADJ1	ADJ_MTX_RR[10:0]	359	Gain Adjustment of Cr/R Signal of R Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

## 34.2 Register Description

Table 34.10 and Table 34.11 show the register configurations.

[Symbols used in Register Description]

Initial value	: Register value after a reset
—	: Undefined value
R/W	: Readable/writable. The written value can be read.
R/WC0	: Read and write. Bit is initialized if 0 is written, and ignored if 1 is written.
R/WC1	: Read and write. Bit is initialized if 1 is written, and ignored if 0 is written.
R	: Read-only. The write value should always be 0.
—/W	: Write-only. Read value is undefined.

**Table 34.10 Image Quality Improver Register Configuration (Channel 0)**

Name	Abbreviation	R/W	Initial Value	Address	Access Size
Register update control register in image quality improver (image quality improver 0)	ADJ0_UPDATE	R/WC1	H'0000 0000	H'FCFF 7680	32
Black stretch register (image quality improver 0)	ADJ0_BKSTR_SET	R/W	H'0000 0000	H'FCFF 7684	32
Enhancer timing adjustment register 1 (image quality improver 0)	ADJ0_ENH_TIM1	R/W	H'0000 0010	H'FCFF 7688	32
Enhancer timing adjustment register 2 (image quality improver 0)	ADJ0_ENH_TIM2	R/W	H'0023 01E0	H'FCFF 768C	32
Enhancer timing adjustment register 3 (image quality improver 0)	ADJ0_ENH_TIM3	R/W	H'0091 0280	H'FCFF 7690	32
Enhancer sharpness register 1 (image quality improver 0)	ADJ0_ENH_SHP1	R/W	H'0000 0000	H'FCFF 7694	32
Enhancer sharpness register 2 (image quality improver 0)	ADJ0_ENH_SHP2	R/W	H'0000 0000	H'FCFF 7698	32
Enhancer sharpness register 3 (image quality improver 0)	ADJ0_ENH_SHP3	R/W	H'0000 0000	H'FCFF 769C	32
Enhancer sharpness register 4 (image quality improver 0)	ADJ0_ENH_SHP4	R/W	H'0000 0000	H'FCFF 76A0	32
Enhancer sharpness register 5 (image quality improver 0)	ADJ0_ENH_SHP5	R/W	H'0000 0000	H'FCFF 76A4	32
Enhancer sharpness register 6 (image quality improver 0)	ADJ0_ENH_SHP6	R/W	H'0000 0000	H'FCFF 76A8	32
Enhancer LTI register 1 (image quality improver 0)	ADJ0_ENH_LTI1	R/W	H'000A 0000	H'FCFF 76AC	32
Enhancer LTI register 2 (image quality improver 0)	ADJ0_ENH_LTI2	R/W	H'000A 0000	H'FCFF 76B0	32
Matrix mode register in image quality improver (image quality improver 0)	ADJ0_MTX_MODE	R/W	H'0000 0002	H'FCFF 76B4	32
Matrix YG control register 0 in image quality improver (image quality improver 0)	ADJ0_MTX_YG_ADJ0	R/W	H'0080 0100	H'FCFF 76B8	32
Matrix YG control register 1 in image quality improver (image quality improver 0)	ADJ0_MTX_YG_ADJ1	R/W	H'07A8 0749	H'FCFF 76BC	32
Matrix CBB control register 0 in image quality improver (image quality improver 0)	ADJ0_MTX_CBB_ADJ0	R/W	H'0080 0100	H'FCFF 76C0	32
Matrix CBB control register 1 in image quality improver (image quality improver 0)	ADJ0_MTX_CBB_ADJ1	R/W	H'01C6 0000	H'FCFF 76C4	32

**Table 34.10 Image Quality Improver Register Configuration (Channel 0)**

Name	Abbreviation	R/W	Initial Value	Address	Access Size
Matrix CRR control register 0 in image quality improver (image quality improver 0)	ADJ0_MTX_CRR_ADJ0	R/W	H'0080 0100	H'FCFF 76C8	32
Matrix CRR control register 1 in image quality improver (image quality improver 0)	ADJ0_MTX_CRR_ADJ1	R/W	H'0000 0167	H'FCFF 76CC	32
Register update control register in image quality improver (image quality improver 1)	ADJ1_UPDATE	R/WC1	H'0000 0000	H'FCFF 7D80	32
Black stretch register (image quality improver 1)	ADJ1_BKSTR_SET	R/W	H'0000 0000	H'FCFF 7D84	32
Enhancer timing adjustment register 1 (image quality improver 1)	ADJ1_ENH_TIM1	R/W	H'0000 0010	H'FCFF 7D88	32
Enhancer timing adjustment register 2 (image quality improver 1)	ADJ1_ENH_TIM2	R/W	H'0023 01E0	H'FCFF 7D8C	32
Enhancer timing adjustment register 3 (image quality improver 1)	ADJ1_ENH_TIM3	R/W	H'0091 0280	H'FCFF 7D90	32
Enhancer sharpness register 1 (image quality improver 1)	ADJ1_ENH_SHP1	R/W	H'0000 0000	H'FCFF 7D94	32
Enhancer sharpness register 2 (image quality improver 1)	ADJ1_ENH_SHP2	R/W	H'0000 0000	H'FCFF 7D98	32
Enhancer sharpness register 3 (image quality improver 1)	ADJ1_ENH_SHP3	R/W	H'0000 0000	H'FCFF 7D9C	32
Enhancer sharpness register 4 (image quality improver 1)	ADJ1_ENH_SHP4	R/W	H'0000 0000	H'FCFF 7DA0	32
Enhancer sharpness register 5 (image quality improver 1)	ADJ1_ENH_SHP5	R/W	H'0000 0000	H'FCFF 7DA4	32
Enhancer sharpness register 6 (image quality improver 1)	ADJ1_ENH_SHP6	R/W	H'0000 0000	H'FCFF 7DA8	32
Enhancer LTI register 1 (image quality improver 1)	ADJ1_ENH_LTI1	R/W	H'000A 0000	H'FCFF 7DAC	32
Enhancer LTI register 2 (image quality improver 1)	ADJ1_ENH_LTI2	R/W	H'000A 0000	H'FCFF 7DB0	32
Matrix mode register in image quality improver (image quality improver 1)	ADJ1_MTX_MODE	R/W	H'0000 0002	H'FCFF 7DB4	32
Matrix YG control register 0 in image quality improver (image quality improver 1)	ADJ1_MTX_YG_ADJ0	R/W	H'0080 0100	H'FCFF 7DB8	32
Matrix YG control register 1 in image quality improver (image quality improver 1)	ADJ1_MTX_YG_ADJ1	R/W	H'07A8 0749	H'FCFF 7DBC	32
Matrix CBB control register 0 in image quality improver (image quality improver 1)	ADJ1_MTX_CBB_ADJ0	R/W	H'0080 0100	H'FCFF 7DC0	32
Matrix CBB control register 1 in image quality improver (image quality improver 1)	ADJ1_MTX_CBB_ADJ1	R/W	H'01C6 0000	H'FCFF 7DC4	32
Matrix CRR control register 0 in image quality improver (image quality improver 1)	ADJ1_MTX_CRR_ADJ0	R/W	H'0080 0100	H'FCFF 7DC8	32
Matrix CRR control register 1 in image quality improver (image quality improver 1)	ADJ1_MTX_CRR_ADJ1	R/W	H'0000 0167	H'FCFF 7DCC	32

**Table 34.11 Image Quality Improver Register Configuration (Channel 1)**

Name	Abbreviation	R/W	Initial Value	Address	Access Size
Register update control register in image quality improver (image quality improver 0)	ADJ0_UPDATE	R/WC1	H'0000 0000	H'FCFF9680	32
Black stretch register (image quality improver 0)	ADJ0_BKSTR_SET	R/W	H'0000 0000	H'FCFF9684	32
Enhancer timing adjustment register 1 (image quality improver 0)	ADJ0_ENH_TIM1	R/W	H'0000 0010	H'FCFF9688	32
Enhancer timing adjustment register 2 (image quality improver 0)	ADJ0_ENH_TIM2	R/W	H'0023 01E0	H'FCFF968C	32
Enhancer timing adjustment register 3 (image quality improver 0)	ADJ0_ENH_TIM3	R/W	H'0091 0280	H'FCFF9690	32
Enhancer sharpness register 1 (image quality improver 0)	ADJ0_ENH_SHP1	R/W	H'0000 0000	H'FCFF9694	32
Enhancer sharpness register 2 (image quality improver 0)	ADJ0_ENH_SHP2	R/W	H'0000 0000	H'FCFF9698	32
Enhancer sharpness register 3 (image quality improver 0)	ADJ0_ENH_SHP3	R/W	H'0000 0000	H'FCFF969C	32
Enhancer sharpness register 4 (image quality improver 0)	ADJ0_ENH_SHP4	R/W	H'0000 0000	H'FCFF96A0	32
Enhancer sharpness register 5 (image quality improver 0)	ADJ0_ENH_SHP5	R/W	H'0000 0000	H'FCFF96A4	32
Enhancer sharpness register 6 (image quality improver 0)	ADJ0_ENH_SHP6	R/W	H'0000 0000	H'FCFF96A8	32
Enhancer LTI register 1 (image quality improver 0)	ADJ0_ENH_LTI1	R/W	H'000A 0000	H'FCFF96AC	32
Enhancer LTI register 2 (image quality improver 0)	ADJ0_ENH_LTI2	R/W	H'000A 0000	H'FCFF96B0	32
Matrix mode register in image quality improver (image quality improver 0)	ADJ0_MTX_MODE	R/W	H'0000 0002	H'FCFF96B4	32
Matrix YG control register 0 in image quality improver (image quality improver 0)	ADJ0_MTX_YG_ADJ0	R/W	H'0080 0100	H'FCFF96B8	32
Matrix YG control register 1 in image quality improver (image quality improver 0)	ADJ0_MTX_YG_ADJ1	R/W	H'07A8 0749	H'FCFF96BC	32
Matrix CBB control register 0 in image quality improver (image quality improver 0)	ADJ0_MTX_CBB_ADJ0	R/W	H'0080 0100	H'FCFF96C0	32
Matrix CBB control register 1 in image quality improver (image quality improver 0)	ADJ0_MTX_CBB_ADJ1	R/W	H'01C6 0000	H'FCFF96C4	32
Matrix CRR control register 0 in image quality improver (image quality improver 0)	ADJ0_MTX_CRR_ADJ0	R/W	H'0080 0100	H'FCFF96C8	32
Matrix CRR control register 1 in image quality improver (image quality improver 0)	ADJ0_MTX_CRR_ADJ1	R/W	H'0000 0167	H'FCFF96CC	32
Register update control register in image quality improver (image quality improver 1)	ADJ1_UPDATE	R/WC1	H'0000 0000	H'FCFF9D80	32
Black stretch register (image quality improver 1)	ADJ1_BKSTR_SET	R/W	H'0000 0000	H'FCFF9D84	32
Enhancer timing adjustment register 1 (image quality improver 1)	ADJ1_ENH_TIM1	R/W	H'0000 0010	H'FCFF9D88	32
Enhancer timing adjustment register 2 (image quality improver 1)	ADJ1_ENH_TIM2	R/W	H'0023 01E0	H'FCFF9D8C	32
Enhancer timing adjustment register 3 (image quality improver 1)	ADJ1_ENH_TIM3	R/W	H'0091 0280	H'FCFF9D90	32
Enhancer sharpness register 1 (image quality improver 1)	ADJ1_ENH_SHP1	R/W	H'0000 0000	H'FCFF9D94	32

**Table 34.11 Image Quality Improver Register Configuration (Channel 1)**

Name	Abbreviation	R/W	Initial Value	Address	Access Size
Enhancer sharpness register 2 (image quality improver 1)	ADJ1_ENH_SHP2	R/W	H'0000 0000	H'FCFF9D98	32
Enhancer sharpness register 3 (image quality improver 1)	ADJ1_ENH_SHP3	R/W	H'0000 0000	H'FCFF9D9C	32
Enhancer sharpness register 4 (image quality improver 1)	ADJ1_ENH_SHP4	R/W	H'0000 0000	H'FCFF9DA0	32
Enhancer sharpness register 5 (image quality improver 1)	ADJ1_ENH_SHP5	R/W	H'0000 0000	H'FCFF9DA4	32
Enhancer sharpness register 6 (image quality improver 1)	ADJ1_ENH_SHP6	R/W	H'0000 0000	H'FCFF9DA8	32
Enhancer LTI register 1 (image quality improver 1)	ADJ1_ENH_LTI1	R/W	H'000A 0000	H'FCFF9DAC	32
Enhancer LTI register 2 (image quality improver 1)	ADJ1_ENH_LTI2	R/W	H'000A 0000	H'FCFF9DB0	32
Matrix mode register in image quality improver (image quality improver 1)	ADJ1_MTX_MODE	R/W	H'0000 0002	H'FCFF9DB4	32
Matrix YG control register 0 in image quality improver (image quality improver 1)	ADJ1_MTX_YG_ADJ0	R/W	H'0080 0100	H'FCFF9DB8	32
Matrix YG control register 1 in image quality improver (image quality improver 1)	ADJ1_MTX_YG_ADJ1	R/W	H'07A8 0749	H'FCFF9DBC	32
Matrix CBB control register 0 in image quality improver (image quality improver 1)	ADJ1_MTX_CBB_ADJ0	R/W	H'0080 0100	H'FCFF9DC0	32
Matrix CBB control register 1 in image quality improver (image quality improver 1)	ADJ1_MTX_CBB_ADJ1	R/W	H'01C6 0000	H'FCFF9DC4	32
Matrix CRR control register 0 in image quality improver (image quality improver 1)	ADJ1_MTX_CRR_ADJ0	R/W	H'0080 0100	H'FCFF9DC8	32
Matrix CRR control register 1 in image quality improver (image quality improver 1)	ADJ1_MTX_CRR_ADJ1	R/W	H'0000 0167	H'FCFF9DCC	32

### 34.2.1 Register Update Control Register in Image Quality Improver (ADJ0\_UPDATE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ADJ0_VEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	RWC1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	ADJ0_VEN	0	R/WC1	Image Quality Improver Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.

## 34.2.2 Black Stretch Register (ADJ0\_BKSTR\_SET)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	BKSTR_ON	BKSTR_ST[3:0]			BKSTR_D[3:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	BKSTR_T1[4:0]				—	—	—	BKSTR_T2[4:0]					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	BKSTR_ON	0	R/W	Black Stretch On/Off Control 0: Black stretch off 1: Black stretch on
23 to 20	BKSTR_ST [3:0]	0	R/W	Black Stretch Start Point Setting values: 0 (low) to 15 (high)
19 to 16	BKSTR_D [3:0]	0	R/W	Depth of Black Stretch Setting Values: 0 (shallow) to 15 (deep)
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 8	BKSTR_T1 [4:0]	0	R/W	Black Stretch Time Constant (T1) Setting Values: 0 (small) to 31 (large)
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4 to 0	BKSTR_T2 [4:0]	0	R/W	Black Stretch Time Constant (T2) Setting Values: 0 (small) to 31 (large)

Note: This register is updated when ADJ0\_VEN in ADJ0\_UPDATE is 1.

## 34.2.3 Enhancer Timing Adjustment Register 1 (ADJ0\_ENH\_TIM1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	ENH_MD	—	—	—	ENH_DISP_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	ENH_MD	1	R/W	Operating Mode 0: RGB mode 1: YCbCr mode
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	ENH_DISP_ON	0	R/W	Frame Line Display On/Off of Enhancer-Enabled Area 0: Display off 1: Display on

Note: This register is updated when ADJ0\_VEN in ADJ0\_UPDATE is 1.

## 34.2.4 Enhancer Timing Adjustment Register 2 (ADJ0\_ENH\_TIM2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	ENH_VS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ENH_VW[10:0]										
Initial value:	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	ENH_VS[10:0]	35	R/W	Start Position of Vertical Valid Image Area in Enhancer-Enabled Area Note: Set to 2 or greater lines.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	ENH_VW[10:0]	480	R/W	Width of Vertical Valid Image Area in Enhancer-Enabled Area

Note: This register is updated when ADJ0\_VEN in ADJ0\_UPDATE is 1.



## 34.2.5 Enhancer Timing Adjustment Register 3 (ADJ0\_ENH\_TIM3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	ENH_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ENH_HW[10:0]										
Initial value:	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	ENH_HS[10:0]	145	R/W	Start Position of Horizontal Valid Image Area in Enhancer-Enabled Area Note: Set to 4 or greater clocks.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	ENH_HW[10:0]	640	R/W	Width of Horizontal Valid Image Area in Enhancer-Enabled Area

Note: This register is updated when ADJ0\_VEN in ADJ0\_UPDATE is 1.

## 34.2.6 Enhancer Sharpness Register 1 (ADJ0\_ENH\_SHP1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SHP_H_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	SHP_H1_CORE[6:0]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	SHP_H_ON	0	R/W	Sharpness On/Off Control 0: Horizontal sharpness off 1: Horizontal sharpness on
15 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6 to 0	SHP_H1_CORE[6:0]	0	R/W	Active Sharpness Range Edge amplitude value $\geq$ SHP_H1_CORE: Sharpness processing on Edge amplitude value $<$ SHP_H1_CORE: Sharpness processing off Sharpness processing is always on when the edge detection value is 128 or greater.

Note: This register is updated when ADJ0\_VEN in ADJ0\_UPDATE is 1.

## 34.2.7 Enhancer Sharpness Register 2 (ADJ0\_ENH\_SHP2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SHP_H1_CLIP_O[7:0]								SHP_H1_CLIP_U[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SHP_H1_GAIN_O[7:0]								SHP_H1_GAIN_U[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	SHP_H1_CLIP_O [7:0]	0	R/W	Sharpness Correction Value Clipping (on the Overshoot Side) Correction value clipped according to SHP_H1_CLIP_O
23 to 16	SHP_H1_CLIP_U [7:0]	0	R/W	Sharpness Correction Value Clipping (on the Undershoot Side) Correction value clipped according to SHP_H1_CLIP_U
15 to 8	SHP_H1_GAIN_O [7:0]	0	R/W	Sharpness Edge Amplitude Value Gain (on the Overshoot Side) 0 (0 times) to 64 (+1 times) to 255 (+approx. 4 times) Sharpness correction value = SHP_H1_GAIN_O × (edge amplitude value – SHP_H1_CORE)
7 to 0	SHP_H1_GAIN_U [7:0]	0	R/W	Sharpness Edge Amplitude Value Gain (on the Undershoot Side) 0 (0 times) to 64 (+1 times) to 255 (+approx. 4 times) Sharpness correction value = SHP_H1_GAIN_U × (Edge amplitude value – SHP_H1_CORE)

Note: This register is updated when ADJ0\_VEN in ADJ0\_UPDATE is 1.

## 34.2.8 Enhancer Sharpness Register 3 (ADJ0\_ENH\_SHP3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SHP_H2_LPF_SEL
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	SHP_H2_CORE[6:0]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	SHP_H2_LPF_SEL	0	R/W	LPF Selection for Folding Prevention before H2 Edge Detection 0: LPF not selected 1: LPF selected
15 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6 to 0	SHP_H2_CORE[6:0]	0	R/W	Active Sharpness Range Edge amplitude value $\geq$ SHP_H2_CORE: Sharpness processing on Edge amplitude value $<$ SHP_H2_CORE: Sharpness processing off Sharpness processing is always on when the edge detection value is 128 or greater.

Note: This register is updated when ADJ0\_VEN in ADJ0\_UPDATE is 1.

## 34.2.9 Enhancer Sharpness Register 4 (ADJ0\_ENH\_SHP4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SHP_H2_CLIP_O[7:0]								SHP_H2_CLIP_U[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SHP_H2_GAIN_O[7:0]								SHP_H2_GAIN_U[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	SHP_H2_CLIP_O[7:0]	0	R/W	Sharpness Correction Value Clipping (on the Overshoot Side) Correction value clipped according to SHP_H2_CLIP_O
23 to 16	SHP_H2_CLIP_U[7:0]	0	R/W	Sharpness Correction Value Clipping (on the Undershoot Side) Correction value clipped according to SHP_H2_CLIP_U
15 to 8	SHP_H2_GAIN_O[7:0]	0	R/W	Sharpness Edge Amplitude Value Gain (on the Overshoot Side) 0 (0 times) to 64 (+1 times) to 255 (+ approx. 4 times) Sharpness correction value = SHP_H2_GAIN_O × (edge amplitude value – SHP_H2_CORE)
7 to 0	SHP_H2_GAIN_U[7:0]	0	R/W	Sharpness Edge Amplitude Value Gain (on the Undershoot Side) 0 (0 times) to 64 (+1 times) to 255 (+ approx. 4 times) Sharpness correction value = SHP_H2_GAIN_U × (edge amplitude value – SHP_H2_CORE)

Note: This register is updated when ADJ0\_VEN in ADJ0\_UPDATE is 1.

## 34.2.10 Enhancer Sharpness Register 5 (ADJ0\_ENH\_SHP5)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	SHP_H3_CORE[6:0]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6 to 0	SHP_H3_CORE[6:0]	0	R/W	Active Sharpness Range Edge amplitude value ≥ SHP_H3_CORE: Sharpness processing on Edge amplitude value < SHP_H3_CORE: Sharpness processing off Sharpness processing is always on when the edge detection value is 128 or greater.

Note: This register is updated when ADJ0\_VEN in ADJ0\_UPDATE is 1.

## 34.2.11 Enhancer Sharpness Register 6 (ADJ0\_ENH\_SHP6)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SHP_H3_CLIP_O[7:0]								SHP_H3_CLIP_U[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SHP_H3_GAIN_O[7:0]								SHP_H3_GAIN_U[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	SHP_H3_CLIP_O[7:0]	0	R/W	Sharpness Correction Value Clipping (on the Overshoot Side) Correction value clipped according to SHP_H3_CLIP_O
23 to 16	SHP_H3_CLIP_U[7:0]	0	R/W	Sharpness Correction Value Clipping (on the Undershoot Side) Correction value clipped according to SHP_H3_CLIP_U
15 to 8	SHP_H3_GAIN_O[7:0]	0	R/W	Sharpness Edge Amplitude Value Gain (on the Overshoot Side) 0 (0 times) to 64 (+1 times) to 255 (+ approx. 4 times) Sharpness correction value = SHP_H3_GAIN_O × (Edge amplitude value – SHP_H3_CORE)
7 to 0	SHP_H3_GAIN_U[7:0]	0	R/W	Sharpness Edge Amplitude Value Gain (on the Undershoot Side) 0 (0 times) to 64 (+1 times) to 255 (+ approx. 4 times) Sharpness correction value = SHP_H3_GAIN_U × (Edge amplitude value – SHP_H3_CORE)

Note: This register is updated when ADJ0\_VEN in ADJ0\_UPDATE is 1.

## 34.2.12 Enhancer LTI Register 1 (ADJ0\_ENH\_LTI1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LTI_H_ON	—	—	—	—	—	—	LTI_H2_LPF_SEL	LTI_H2_INC_ZERO[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0
R/W:	R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LTI_H2_GAIN[7:0]								LTI_H2_CORE[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	LTI_H_ON	0	R/W	LTI On/Off Control 0: LTI off 1: LTI on
30 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	LTI_H2_LPF_SEL	0	R/W	LPF Selection for Folding Prevention before H2 Edge Detection 0: LPF not selected 1: LPF selected
23 to 16	LTI_H2_INC_ZERO[7:0]	10	R/W	Median Filter LTI Correction Threshold LTI correction is disabled when $  \text{right TAP value} - \text{center TAP value}   < \text{LTI\_H2\_INC\_ZERO}$ or $  \text{left TAP value} - \text{center TAP value}   < \text{LTI\_H2\_INC\_ZERO}$
15 to 8	LTI_H2_GAIN[7:0]	0	R/W	LTI Edge Amplitude Value Gain 0 (0 times) to 64 (+ 1 times) to 255 (+ approx. 4 times)
7 to 0	LTI_H2_CORE[7:0]	0	R/W	LTI Coring (Maximum Core value of 255) Amplitude less than or equal to the value of LTI_H2_CORE is cored from the edge amplitude value. (A core value setting of 128 remains unchanged)

Note: This register is updated when ADJ0\_VEN in ADJ0\_UPDATE is 1.

## 34.2.13 Enhancer LTI Register 2 (ADJ0\_ENH\_LTI2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	
	—	—	—	—	—	—	—	LTI_H4_MEDIAN_TAP_SEL	LTI_H4_INC_ZERO[7:0]								
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	LTI_H4_GAIN[7:0]								LTI_H4_CORE[7:0]								
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	LTI_H4_MEDIAN_TAP_SEL	0	R/W	Median Filter Reference Pixel Select 0: Second adjacent pixel selected as reference 1: Adjacent pixel selected as reference
23 to 16	LTI_H4_INC_ZERO[7:0]	10	R/W	Median Filter LTI Correction Threshold LTI correction is disabled when   right TAP value – center TAP value   < LTI_H4_INC_ZERO or   left TAP value – center TAP value   < LTI_H4_INC_ZERO
15 to 8	LTI_H4_GAIN[7:0]	0	R/W	LTI Edge Amplitude Value Gain 0 (0 times) to 64 (+ 1 times) to 255 (+ approx. 4 times)
7 to 0	LTI_H4_CORE[7:0]	0	R/W	LTI Coring (Maximum Core value of 255) Amplitude less than or equal to the value of LTI_H4_CORE is cored from the edge amplitude value (A core value setting of 128 remains unchanged)

Note: This register is updated when ADJ0\_VEN in ADJ0\_UPDATE is 1.

## 34.2.14 Matrix Mode Register in Image Quality Improver (ADJ0\_MTX\_MODE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ADJ0_MTX_MD[1:0]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	ADJ0_MTX_MD[1:0]	2	R/W	Operating Mode 0: GBR → GBR 1: Setting prohibited 2: YCbCr → GBR 3: Setting prohibited

Note: This register is updated when ADJ0\_VEN in ADJ0\_UPDATE is 1.

## 34.2.15 Matrix YG Control Register 0 in Image Quality Improver (ADJ0\_MTX\_YG\_ADJ0)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	ADJ0_MTX_YG[7:0]							
Initial value:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ADJ0_MTX_GG[10:0]										
Initial value:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	ADJ0_MTX_YG[7:0]	128	R/W	Y/G Signal Offset (DC) Adjustment Unsigned (0 (-128) to 128 (0) to 255 (+127) [LSB])
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	ADJ0_MTX_GG[10:0]	256	R/W	Gain Adjustment of Y/G Signal of G Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

Note: This register is updated when ADJ0\_VEN in ADJ0\_UPDATE is 1.



### 34.2.16 Matrix YG Control Register 1 in Image Quality Improver (ADJ0\_MTX\_YG\_ADJ1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	ADJ0_MTX_GB[10:0]										
Initial value:	0	0	0	0	0	1	1	1	1	0	1	0	1	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ADJ0_MTX_GR[10:0]										
Initial value:	0	0	0	0	0	1	1	1	0	1	0	0	1	0	0	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	ADJ0_MTX_GB [10:0]	1960	R/W	Gain Adjustment of Cb/B Signal of G Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	ADJ0_MTX_GR [10:0]	1865	R/W	Gain Adjustment of Cr/R Signal of G Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

Note: This register is updated when ADJ0\_VEN in ADJ0\_UPDATE is 1.

### 34.2.17 Matrix CBB Control Register 0 in Image Quality Improver (ADJ0\_MTX\_CBB\_ADJ0)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	ADJ0_MTX_B[7:0]							
Initial value:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ADJ0_MTX_BG[10:0]										
Initial value:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	ADJ0_MTX_B [7:0]	128	R/W	B Signal Offset (DC) Adjustment Unsigned (0 (-128) to 128 (0) to 255 (+127) [LSB])
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	ADJ0_MTX_BG [10:0]	256	R/W	Gain Adjustment of Y/G Signal of B Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

Note: This register is updated when ADJ0\_VEN in ADJ0\_UPDATE is 1.

### 34.2.18 Matrix CBB Control Register 1 in Image Quality Improver (ADJ0\_MTX\_CBB\_ADJ1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	ADJ0_MTX_BB[10:0]										
Initial value:	0	0	0	0	0	0	0	1	1	1	0	0	0	1	1	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ADJ0_MTX_BR[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	ADJ0_MTX_BB [10:0]	454	R/W	Gain Adjustment of Cb/B Signal of B Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	ADJ0_MTX_BR [10:0]	0	R/W	Gain Adjustment of Cr/R Signal of B Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

Note: This register is updated when ADJ0\_VEN in ADJ0\_UPDATE is 1.

### 34.2.19 Matrix CRR Control Register 0 in Image Quality Improver (ADJ0\_MTX\_CRR\_ADJ0)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	ADJ0_MTX_R[7:0]							
Initial value:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ADJ0_MTX_RG[10:0]										
Initial value:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	ADJ0_MTX_R [7:0]	128	R/W	R Signal Offset (DC) Adjustment Unsigned (0 (-128) to 128 (0) to 255 (+127) [LSB])
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	ADJ0_MTX_RG [10:0]	256	R/W	Gain Adjustment of Y/G Signal of R Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

Note: This register is updated when ADJ0\_VEN in ADJ0\_UPDATE is 1.

### 34.2.20 Matrix CRR Control Register 1 in Image Quality Improver (ADJ0\_MTX\_CRR\_ADJ1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	ADJ0_MTX_RB[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ADJ0_MTX_RR[10:0]										
Initial value:	0	0	0	0	0	0	0	1	0	1	1	0	0	1	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	ADJ0_MTX_RB [10:0]	0	R/W	Gain Adjustment of Cb/B Signal of R Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	ADJ0_MTX_RR [10:0]	359	R/W	Gain Adjustment of Cr/R Signal of R Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

Note: This register is updated when ADJ0\_VEN in ADJ0\_UPDATE is 1.

### 34.2.21 Register Update Control Register in Image Quality Improver (ADJ1\_UPDATE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ADJ1_VEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/WC1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	ADJ1_VEN	0	R/WC1	Image Quality Improver Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.

## 34.2.22 Black Stretch Register (ADJ1\_BKSTR\_SET)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	BKSTR_ON	BKSTR_ST[3:0]			BKSTR_D[3:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	BKSTR_T1[4:0]				—	—	—	BKSTR_T2[4:0]					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	BKSTR_ON	0	R/W	Black Stretch On/Off Control 0: Black stretch off 1: Black stretch on
23 to 20	BKSTR_ST [3:0]	0	R/W	Black Stretch Start Point Setting values: 0 (low) to 15 (high)
19 to 16	BKSTR_D [3:0]	0	R/W	Depth of Black Stretch Setting Values: 0 (shallow) to 15 (deep)
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 8	BKSTR_T1 [4:0]	0	R/W	Black Stretch Time Constant (T1) Setting Values: 0 (small) to 31 (large)
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4 to 0	BKSTR_T2 [4:0]	0	R/W	Black Stretch Time Constant (T2) Setting Values: 0 (small) to 31 (large)

Note: This register is updated when ADJ1\_VEN in ADJ1\_UPDATE is 1.

## 34.2.23 Enhancer Timing Adjustment Register 1 (ADJ1\_ENH\_TIM1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	ENH_MD	—	—	—	ENH_DISP_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	ENH_MD	1	R/W	Operating Mode 0: RGB mode 1: YCbCr mode
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	ENH_DISP_ON	0	R/W	Frame Line Display On/Off of Enhancer-Enabled Area 0: Display off 1: Display on

Note: This register is updated when ADJ1\_VEN in ADJ1\_UPDATE is 1.

## 34.2.24 Enhancer Timing Adjustment Register 2 (ADJ1\_ENH\_TIM2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	ENH_VS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ENH_VW[10:0]										
Initial value:	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	ENH_VS[10:0]	35	R/W	Start Position of Vertical Valid Image Area in Enhancer-Enabled Area Note: Set to 2 or greater lines.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	ENH_VW[10:0]	480	R/W	Width of Vertical Valid Image Area in Enhancer-Enabled Area

Note: This register is updated when ADJ1\_VEN in ADJ1\_UPDATE is 1.

## 34.2.25 Enhancer Timing Adjustment Register 3 (ADJ1\_ENH\_TIM3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	ENH_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ENH_HW[10:0]										
Initial value:	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	ENH_HS[10:0]	145	R/W	Start Position of Horizontal Valid Image Area in Enhancer-Enabled Area Note: Set to 4 or greater clocks.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	ENH_HW[10:0]	640	R/W	Width of Horizontal Valid Image Area in Enhancer-Enabled Area

Note: This register is updated when ADJ1\_VEN in ADJ1\_UPDATE is 1.



## 34.2.26 Enhancer Sharpness Register 1 (ADJ1\_ENH\_SHP1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SHP_H_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	SHP_H1_CORE[6:0]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	SHP_H_ON	0	R/W	Sharpness On/Off Control 0: Horizontal sharpness off 1: Horizontal sharpness on
15 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6 to 0	SHP_H1_CORE[6:0]	0	R/W	Active Sharpness Range Edge amplitude value $\geq$ SHP_H1_CORE: Sharpness processing on Edge amplitude value $<$ SHP_H1_CORE: Sharpness processing off Sharpness processing is always on when the edge detection value is 128 or greater.

Note: This register is updated when ADJ1\_VEN in ADJ1\_UPDATE is 1.

## 34.2.27 Enhancer Sharpness Register 2 (ADJ1\_ENH\_SHP2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SHP_H1_CLIP_O[7:0]								SHP_H1_CLIP_U[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SHP_H1_GAIN_O[7:0]								SHP_H1_GAIN_U[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	SHP_H1_CLIP_O[7:0]	0	R/W	Sharpness Correction Value Clipping (on the Overshoot Side) Correction value clipped according to SHP_H1_CLIP_O
23 to 16	SHP_H1_CLIP_U[7:0]	0	R/W	Sharpness Correction Value Clipping (on the Undershoot Side) Correction value clipped according to SHP_H1_CLIP_U
15 to 8	SHP_H1_GAIN_O[7:0]	0	R/W	Sharpness Edge Amplitude Value Gain (on the Overshoot Side) 0 (0 times) to 64 (+1 times) to 255 (+approx. 4 times) Sharpness correction value = SHP_H1_GAIN_O × (edge amplitude value – SHP_H1_CORE)
7 to 0	SHP_H1_GAIN_U[7:0]	0	R/W	Sharpness Edge Amplitude Value Gain (on the Undershoot Side) 0 (0 times) to 64 (+1 times) to 255 (+approx. 4 times) Sharpness correction value = SHP_H1_GAIN_U × (Edge amplitude value – SHP_H1_CORE)

Note: This register is updated when ADJ1\_VEN in ADJ1\_UPDATE is 1.

34.2.28 Enhancer Sharpness Register 3 (ADJ1\_ENH\_SHP3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SHP_H2_LPF_SEL
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	SHP_H2_CORE[6:0]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	SHP_H2_LPF_SEL	0	R/W	LPF Selection for Folding Prevention before H2 Edge Detection 0: LPF not selected 1: LPF selected
15 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6 to 0	SHP_H2_CORE[6:0]	0	R/W	Active Sharpness Range Edge amplitude value ≥ SHP_H2_CORE: Sharpness processing on Edge amplitude value < SHP_H2_CORE: Sharpness processing off Sharpness processing is always on when the edge detection value is 128 or greater.

Note: This register is updated when ADJ1\_VEN in ADJ1\_UPDATE is 1.

## 34.2.29 Enhancer Sharpness Register 4 (ADJ1\_ENH\_SHP4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SHP_H2_CLIP_O[7:0]								SHP_H2_CLIP_U[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SHP_H2_GAIN_O[7:0]								SHP_H2_GAIN_U[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	SHP_H2_CLIP_O[7:0]	0	R/W	Sharpness Correction Value Clipping (on the Overshoot Side) Correction value clipped according to SHP_H2_CLIP_O
23 to 16	SHP_H2_CLIP_U[7:0]	0	R/W	Sharpness Correction Value Clipping (on the Undershoot Side) Correction value clipped according to SHP_H2_CLIP_U
15 to 8	SHP_H2_GAIN_O[7:0]	0	R/W	Sharpness Edge Amplitude Value Gain (on the Overshoot Side) 0 (0 times) to 64 (+1 times) to 255 (+ approx. 4 times) Sharpness correction value = SHP_H2_GAIN_O × (edge amplitude value – SHP_H2_CORE)
7 to 0	SHP_H2_GAIN_U[7:0]	0	R/W	Sharpness Edge Amplitude Value Gain (on the Undershoot Side) 0 (0 times) to 64 (+1 times) to 255 (+ approx. 4 times) Sharpness correction value = SHP_H2_GAIN_U × (edge amplitude value – SHP_H2_CORE)

Note: This register is updated when ADJ1\_VEN in ADJ1\_UPDATE is 1.

## 34.2.30 Enhancer Sharpness Register 5 (ADJ1\_ENH\_SHP5)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	SHP_H3_CORE[6:0]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6 to 0	SHP_H3_CORE[6:0]	0	R/W	Active Sharpness Range Edge amplitude value ≥ SHP_H3_CORE: Sharpness processing on Edge amplitude value < SHP_H3_CORE: Sharpness processing off Sharpness processing is always on when the edge detection value is 128 or greater.

Note: This register is updated when ADJ1\_VEN in ADJ1\_UPDATE is 1.

## 34.2.31 Enhancer Sharpness Register 6 (ADJ1\_ENH\_SHP6)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SHP_H3_CLIP_O[7:0]								SHP_H3_CLIP_U[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SHP_H3_GAIN_O[7:0]								SHP_H3_GAIN_U[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	SHP_H3_CLIP_O[7:0]	0	R/W	Sharpness Correction Value Clipping (on the Overshoot Side) Correction value clipped according to SHP_H3_CLIP_O
23 to 16	SHP_H3_CLIP_U[7:0]	0	R/W	Sharpness Correction Value Clipping (on the Undershoot Side) Correction value clipped according to SHP_H3_CLIP_U
15 to 8	SHP_H3_GAIN_O[7:0]	0	R/W	Sharpness Edge Amplitude Value Gain (on the Overshoot Side) 0 (0 times) to 64 (+1 times) to 255 (+ approx. 4 times) Sharpness correction value = SHP_H3_GAIN_O × (Edge amplitude value – SHP_H3_CORE)
7 to 0	SHP_H3_GAIN_U[7:0]	0	R/W	Sharpness Edge Amplitude Value Gain (on the Undershoot Side) 0 (0 times) to 64 (+1 times) to 255 (+ approx. 4 times) Sharpness correction value = SHP_H3_GAIN_U × (Edge amplitude value – SHP_H3_CORE)

Note: This register is updated when ADJ1\_VEN in ADJ1\_UPDATE is 1.

## 34.2.32 Enhancer LTI Register 1 (ADJ1\_ENH\_LTI1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	LTI_H_ON	—	—	—	—	—	—	LTI_H2_LPF_SEL	LTI_H2_INC_ZERO[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0
R/W:	R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LTI_H2_GAIN[7:0]								LTI_H2_CORE[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	LTI_H_ON	0	R/W	LTI On/Off Control 0: LTI off 1: LTI on
30 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	LTI_H2_LPF_SEL	0	R/W	LPF Selection for Folding Prevention before H2 Edge Detection 0: LPF not selected 1: LPF selected
23 to 16	LTI_H2_INC_ZERO[7:0]	10	R/W	Median Filter LTI Correction Threshold LTI correction is disabled when   right TAP value – center TAP value   < LTI_H2_INC_ZERO or   left TAP value – center TAP value   < LTI_H2_INC_ZERO
15 to 8	LTI_H2_GAIN[7:0]	0	R/W	LTI Edge Amplitude Value Gain 0 (0 times) to 64 (+ 1 times) to 255 (+ approx. 4 times)
7 to 0	LTI_H2_CORE[7:0]	0	R/W	LTI Coring (Maximum Core value of 255) Amplitude less than or equal to the value of LTI_H2_CORE is cored from the edge amplitude value. (A core value setting of 128 remains unchanged)

Note: This register is updated when ADJ1\_VEN in ADJ1\_UPDATE is 1.

## 34.2.33 Enhancer LTI Register 2 (ADJ1\_ENH\_LTI2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	LTI_H4_MEDIAN_TAP_SEL	LTI_H4_INC_ZERO[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LTI_H4_GAIN[7:0]								LTI_H4_CORE[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	LTI_H4_MEDIAN_TAP_SEL	0	R/W	Median Filter Reference Pixel Select 0: Second adjacent pixel selected as reference 1: Adjacent pixel selected as reference
23 to 16	LTI_H4_INC_ZERO[7:0]	10	R/W	Median Filter LTI Correction Threshold LTI correction is disabled when   right TAP value – center TAP value   < LTI_H4_INC_ZERO or   left TAP value – center TAP value   < LTI_H4_INC_ZERO
15 to 8	LTI_H4_GAIN[7:0]	0	R/W	LTI Edge Amplitude Value Gain 0 (0 times) to 64 (+ 1 times) to 255 (+ approx. 4 times)
7 to 0	LTI_H4_CORE[7:0]	0	R/W	LTI Coring (Maximum Core value of 255) Amplitude less than or equal to the value of LTI_H4_CORE is cored from the edge amplitude value (A core value setting of 128 remains unchanged)

Note: This register is updated when ADJ1\_VEN in ADJ1\_UPDATE is 1.

## 34.2.34 Matrix Mode Register in Image Quality Improver (ADJ1\_MTX\_MODE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ADJ1_MTX_MD[1:0]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	ADJ1_MTX_MD[1:0]	2	R/W	Operating Mode 0: GBR → GBR 1: Setting prohibited 2: YCbCr → GBR 3: Setting prohibited

Note: This register is updated when ADJ1\_VEN in ADJ1\_UPDATE is 1.

## 34.2.35 Matrix YG Control Register 0 in Image Quality Improver (ADJ1\_MTX\_YG\_ADJ0)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	ADJ1_MTX_YG[7:0]							
Initial value:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ADJ1_MTX_GG[10:0]										
Initial value:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	ADJ1_MTX_YG[7:0]	128	R/W	Y/G Signal Offset (DC) Adjustment Unsigned (0 (-128) to 128 (0) to 255 (+127) [LSB])
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	ADJ1_MTX_GG[10:0]	256	R/W	Gain Adjustment of Y/G Signal of G Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

Note: This register is updated when ADJ1\_VEN in ADJ1\_UPDATE is 1.



### 34.2.36 Matrix YG Control Register 1 in Image Quality Improver (ADJ1\_MTX\_YG\_ADJ1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	ADJ1_MTX_GB[10:0]										
Initial value:	0	0	0	0	0	1	1	1	1	0	1	0	1	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ADJ1_MTX_GR[10:0]										
Initial value:	0	0	0	0	0	1	1	1	0	1	0	0	1	0	0	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	ADJ1_MTX_GB [10:0]	1960	R/W	Gain Adjustment of Cb/B Signal of G Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	ADJ1_MTX_GR [10:0]	1865	R/W	Gain Adjustment of Cr/R Signal of G Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

Note: This register is updated when ADJ1\_VEN in ADJ1\_UPDATE is 1.

### 34.2.37 Matrix CBB Control Register 0 in Image Quality Improver (ADJ1\_MTX\_CBB\_ADJ0)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	ADJ1_MTX_B[7:0]							
Initial value:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ADJ1_MTX_BG[10:0]										
Initial value:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	ADJ1_MTX_B [7:0]	128	R/W	B Signal Offset (DC) Adjustment Unsigned (0 (-128) to 128 (0) to 255 (+127) [LSB])
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	ADJ1_MTX_BG [10:0]	256	R/W	Gain Adjustment of Y/G Signal of B Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

Note: This register is updated when ADJ1\_VEN in ADJ1\_UPDATE is 1.

### 34.2.38 Matrix CBB Control Register 1 in Image Quality Improver (ADJ1\_MTX\_CBB\_ADJ1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	ADJ1_MTX_BB[10:0]										
Initial value:	0	0	0	0	0	0	0	1	1	1	0	0	0	1	1	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ADJ1_MTX_BR[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	ADJ1_MTX_BB [10:0]	454	R/W	Gain Adjustment of Cb/B Signal of B Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	ADJ1_MTX_BR [10:0]	0	R/W	Gain Adjustment of Cr/R Signal of B Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

Note: This register is updated when ADJ1\_VEN in ADJ1\_UPDATE is 1.

### 34.2.39 Matrix CRR Control Register 0 in Image Quality Improver (ADJ1\_MTX\_CRR\_ADJ0)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	ADJ1_MTX_R[7:0]							
Initial value:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ADJ1_MTX_RG[10:0]										
Initial value:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	ADJ1_MTX_R [7:0]	128	R/W	R Signal Offset (DC) Adjustment Unsigned (0 (-128) to 128 (0) to 255 (+127) [LSB])
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	ADJ1_MTX_RG [10:0]	256	R/W	Gain Adjustment of Y/G Signal of R Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

Note: This register is updated when ADJ1\_VEN in ADJ1\_UPDATE is 1.

### 34.2.40 Matrix CRR Control Register 1 in Image Quality Improver (ADJ1\_MTX\_CRR\_ADJ1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	
	—	—	—	—	—	ADJ1_MTX_RB[10:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	—	—	—	—	—	ADJ1_MTX_RR[10:0]											
Initial value:	0	0	0	0	0	0	0	1	0	1	1	0	0	1	1	1	
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	ADJ1_MTX_RB [10:0]	0	R/W	Gain Adjustment of Cb/B Signal of R Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	ADJ1_MTX_RR [10:0]	359	R/W	Gain Adjustment of Cr/R Signal of R Signal Output Signed (complement of 2) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

Note: This register is updated when ADJ1\_VEN in ADJ1\_UPDATE is 1.

### 34.3 Usage Method

#### 34.3.1 Black Stretch Usage Method

The degree of black stretch can be adjusted by setting the depth (BKSTR\_D[3:0]) and the start point (BKSTR\_ST[3:0]) of the black stretch. The variation in the black stretch time axis can be adjusted by setting the time constant (BKSTR\_T1[4:0] and BKSTR\_T2[4:0]). By setting the time constant, changes that occur abruptly due to swapping of the scene can be controlled.

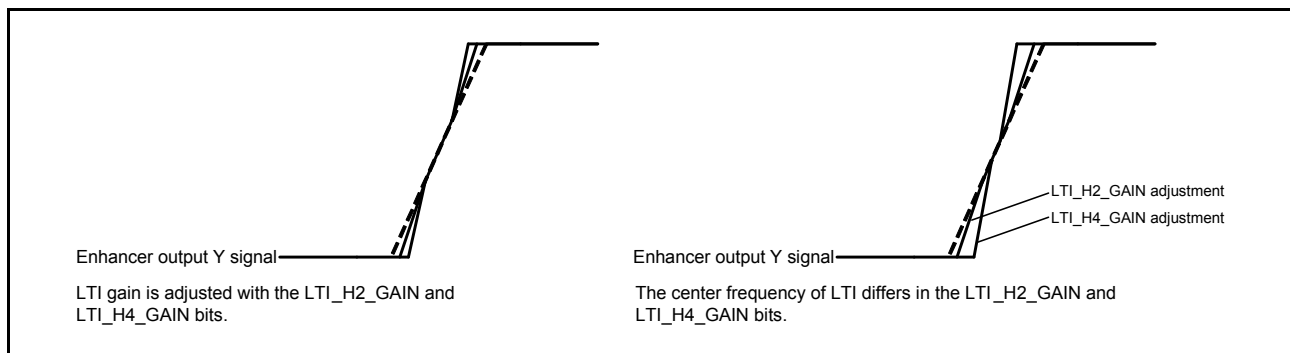
**Table 34.12 Black Stretch Setting Register**

Register Name	Bit Name	Set Value
ADJ_BKSTR_SET	BKSTR_ON	When black stretch is on: 1
ADJ_BKSTR_SET	BKSTR_D[3:0]	Set the depth of black stretch. The depth increases as the value becomes larger.
ADJ_BKSTR_SET	BKSTR_ST[3:0]	Set the start point of black stretch. The stretching area becomes larger as the value becomes larger.
ADJ_BKSTR_SET	BKSTR_T1[4:0]	Set the time constant of black stretch in the positive direction. The changes are more delayed as the value becomes larger.
ADJ_BKSTR_SET	BKSTR_T2[4:0]	Set the time constant of black stretch in the negative direction. The changes are more delayed as the value becomes larger.

Note: ADJ\_VEN in ADJ\_UPDATE should be set to 1 after setting the registers.

#### 34.3.2 LTI Processing of Enhancer

Figure 34.8 shows an example of LTI adjustment.



**Figure 34.8 Example of LTI Adjustment**

### 34.3.3 Sharpness Processing of Enhancer

Figure 34.9 shows an example of sharpness adjustment.

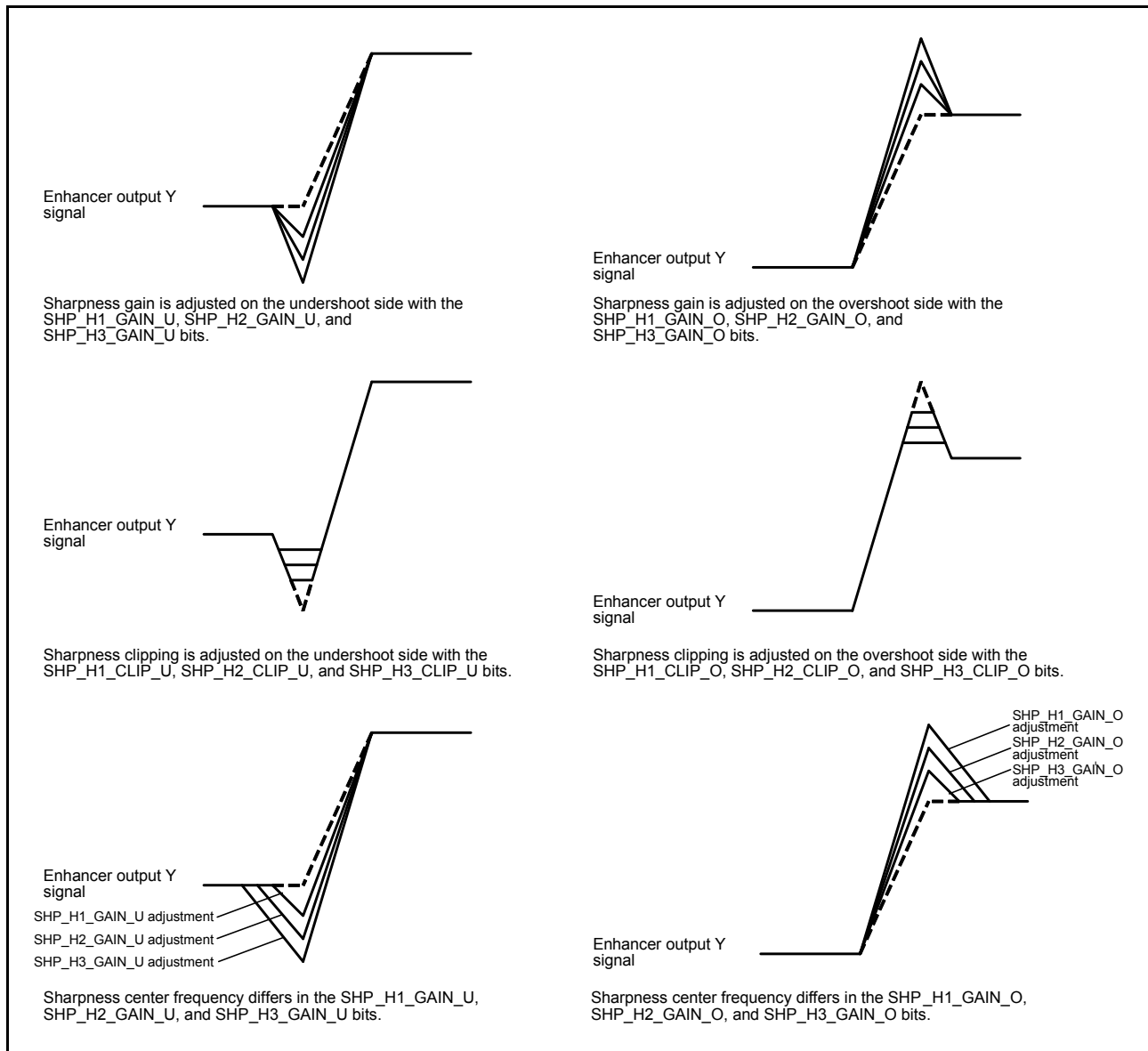


Figure 34.9 Example of Sharpness Adjustment

### 34.3.4 Setting Method for Color Matrix Data Conversion

GBR signals are assumed to be input to the circuit subsequent to the image quality improver; therefore, the output from the color matrix circuit should be in the GBR format.

Table 34.13 shows an example of GBR conversion setting.

**Table 34.13 Recommended Setting Values for Matrix Conversion**

Register Name	Bit Name	GBR to GBR Conversion	YCBCR to GBR Conversion
		Recommended Values	Recommended Values
ADJ_MTX_MODE	ADJ_MTX_MD[1:0]	0	2
ADJ_MTX_YG_ADJ0	ADJ_MTX_YG[7:0]	128	128
ADJ_MTX_CBB_ADJ0	ADJ_MTX_B[7:0]	128	128
ADJ_MTX_CRR_ADJ0	ADJ_MTX_R[7:0]	128	128
ADJ_MTX_YG_ADJ0	ADJ_MTX_GG[10:0]	256	256
ADJ_MTX_YG_ADJ1	ADJ_MTX_GB[10:0]	0	1960
ADJ_MTX_YG_ADJ1	ADJ_MTX_GR[10:0]	0	1865
ADJ_MTX_CBB_ADJ0	ADJ_MTX_BG[10:0]	0	256
ADJ_MTX_CBB_ADJ1	ADJ_MTX_BB[10:0]	256	454
ADJ_MTX_CBB_ADJ1	ADJ_MTX_BR[10:0]	0	0
ADJ_MTX_CRR_ADJ0	ADJ_MTX_RG[10:0]	0	256
ADJ_MTX_CRR_ADJ1	ADJ_MTX_RB[10:0]	0	0
ADJ_MTX_CRR_ADJ1	ADJ_MTX_RR[10:0]	256	359

Note: ADJ\_VEN in ADJ\_UPDATE should be set to 1 after setting the registers.



## 35. Video Display Controller 5 (5): Image Synthesizer

### 35.1 Image Synthesizer

#### 35.1.1 Overview of Functions

The image synthesizer reads graphics data from the frame buffer and displays the synthesized image on the screen.

Two video planes + two graphics planes, one video plane + three graphics planes, or four graphics planes can be selected for synthesis.

RGB565, RGB888,  $\alpha$ RGB1555,  $\alpha$ RGB4444,  $\alpha$ RGB8888, RGB $\alpha$ 5551, RGB $\alpha$ 8888, CLUT8, CLUT4, CLUT1, YCbCr422 (for the graphics 0 and 1 processes), and YCbCr444 (for the graphics 0 and 1 processes) formats can be used for graphics data, and RGB565, RGB888, YCbCr422, and YCbCr444 formats for video data.

On each of the graphics planes, background color, lower-layer graphics, current graphics, or blended image (for the graphics 1, 2, and 3 processes and VIN synthesizer) of lower-layer graphics and current graphics can be displayed.

In the VIN synthesizer, graphics data is not read and images (for the graphics 0 and 1 processes) are blended.

The functional block diagram of the image synthesizer is shown below.

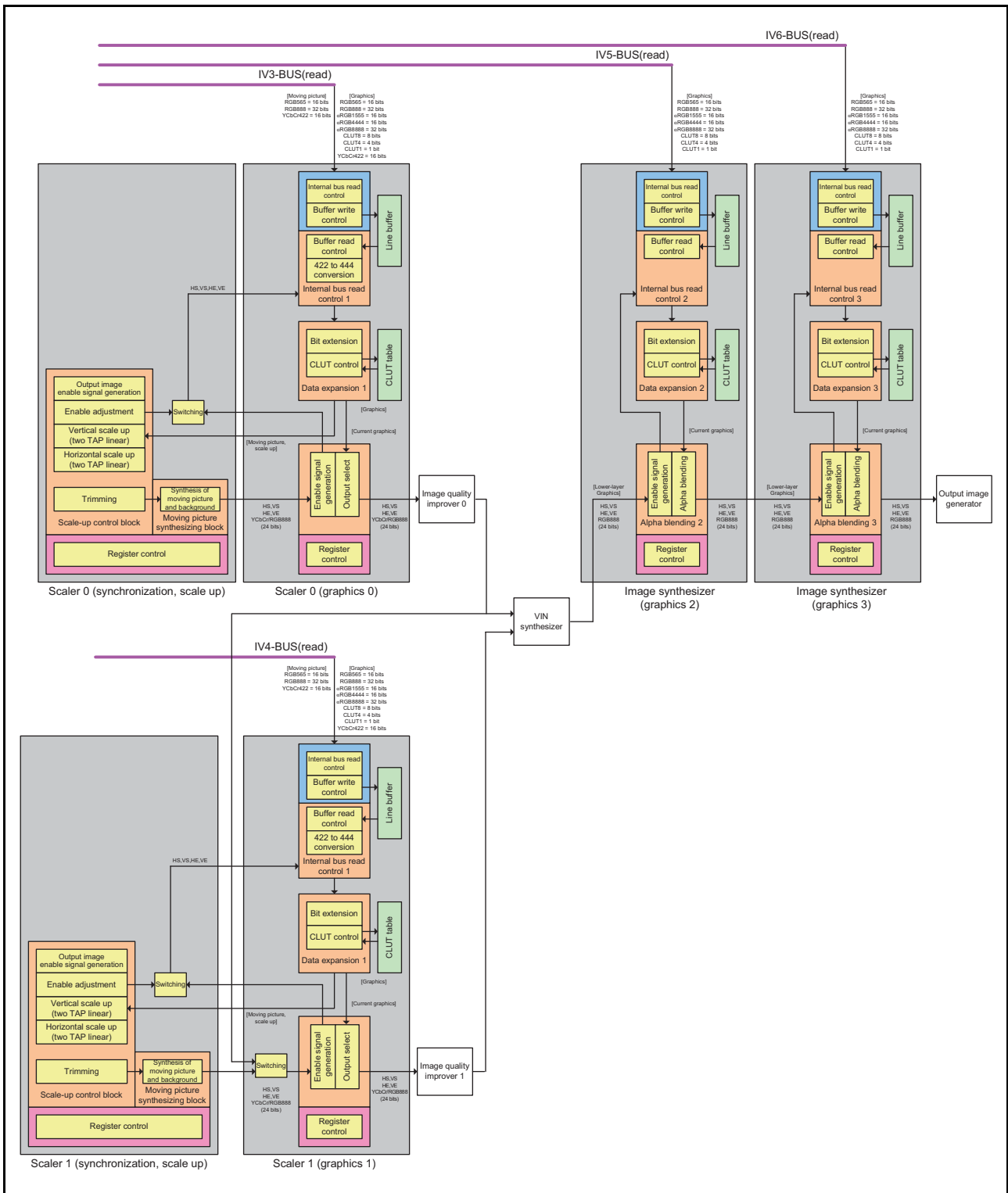


Figure 35.1 Functional Block Diagram of Image Synthesizer

### 35.1.2 Graphics Data Read Control

Graphics data read can be controlled for the five processes: the graphics 0 process in the scaler 0, the graphics 1 process in the scaler 1, the graphics 2 and 3 processes in the image synthesizer, and the graphics OIR process in the output image generator.

The register bits of each process can be identified by the number in the register name like GR0\_xxxx, GR1\_xxxx, GR2\_xxxx, GR3\_xxxx, and GR\_OIR\_xxxx, respectively. In the sections except for Register Descriptions, however, the number is omitted like GR\_xxxx for convenience sake.

In the VIN synthesizer, graphics data read is not controlled and the blending register GR\_VIN\_xxxx is present. In this manual, the name is omitted like GR\_xxxx for convenience sake. The synthesizer does not have the read control register (GR\_FLM).

#### (1) Updating Registers

The Vsync signal is used to control the update timing of the registers for graphics display and frame buffer read control, except for some of the registers.

After 1 is set to the bits in the update control register, the contents of the relevant registers are actually modified at the rising edge of the Vsync signal, when the update control register is automatically cleared to 0.

**Table 35.1 Register Update Control**

Register Name	Bit Name	Initial Value	Description
GR_UPDATE	GR_UPDATE	0	Frame Buffer Read Control Register Update 0: Registers are not updated. 1: Registers are updated.
GR_UPDATE	GR_P_VEN	0	Graphics Display Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
GR_UPDATE	GR_IBUS_VEN	0	Frame Buffer Read Register Update* 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.

Note: \* This bit is not supported for the VIN synthesizer.

#### (2) Frame Buffer Burst Transfer Mode

Either 32-byte or 128-byte transfer mode can be selected for accessing the frame buffer in which video data and graphics data are stored.

**Table 35.2 Frame Buffer Burst Transfer Mode**

Register Name	Bit Name	Initial Value	Description
GR_FLM1	GR_BST_MD	0	Frame Buffer Burst Transfer Mode 0: 32-byte transfer 1: 128- byte transfer

### (3) Frame Buffer Control Mode

More than one frame of data is read from the frame buffer.

For graphics data, set the GR\_FLM\_SEL[1:0] bits to 1, and set the specific display frame number with the GR\_FLM\_NUM[9:0] bits. For video data, select a mode with the GR\_FLM\_SEL[1:0] bits depending on the writing process used; the quantity of the frames used for video data is set in the writing process block.

**Table 35.3 Frame Buffer Control Mode**

Register Name	Bit Name	Initial Value	Description
GR_FLM1	GR_FLM_SEL[1:0]	0	Frame Buffer Address Setting Signal Select 0: Control linked to scaling-down process, or frame 0 selected. *1 1: Register GR_FLM_NUM selected. 2: Control linked to distortion correction, or frame 0 selected. *2 3: Control linked to pointer buffer, or setting prohibited. *3
GR_FLM3	GR_FLM_NUM[9:0]	0	Frame Number of Frame Buffer Manually set the frame number when GR_FLM_SEL = 1.

Notes: 1. For the graphics 0, 1, and OIR processes, frame buffer control links to the scaling-down process. For the graphics 2 and 3 processes, frame 0 is selected.  
2. For the graphics 0 and OIR processes, frame buffer control links to distortion correction. For the graphics 1, 2 and 3 processes, frame 0 is selected.  
3. For the graphics 0, 1, and OIR processes, frame buffer control links to the pointer buffer. For the graphics 2 and 3 processes, setting is prohibited.

### (4) Frame Buffer Read Control

The following bit enables or disables read access to the frame buffer.

**Table 35.4 Frame Buffer Read Control**

Register Name	Bit Name	Initial Value	Description
GR_FLM_RD	GR_R_ENB	0	Frame Buffer Read Enable 0: Disables read access to the frame buffer. 1: Enables read access to the frame buffer.

### (5) Distortion Correction Frame Buffer Control

Two frames (frames 0 and 1) are used for distortion correction, and the frame numbers to be read by the image renderer are set.

The frame numbers to be read (frames 0 and 1) can be switched by setting the GR\_IMR\_FLM\_INV bit.

This bit is enabled only when the GR\_FLM\_SEL bits are set to 2.

**Table 35.5 Distortion Correction Frame Buffer Control**

Register Name	Bit Name	Initial Value	Description
GR1_FLM1	GR1_IMR_FLM_INV	0	Sets the frame buffer number for distortion correction.* 0: Does not switch the frame numbers to be read. 1: Switches the frame numbers to be read.

Note: \* This function is supported for the graphics 0 and OIR processes only.

## (6) Frame Buffer Size

The following bits set the size of the frame buffer to be read.

The numbers of horizontal pixels and of lines in the vertical direction are set with the GR\_HW[10:0] and GR\_FLM\_LNUM[10:0] bits, respectively.

**Table 35.6 Frame Buffer Size**

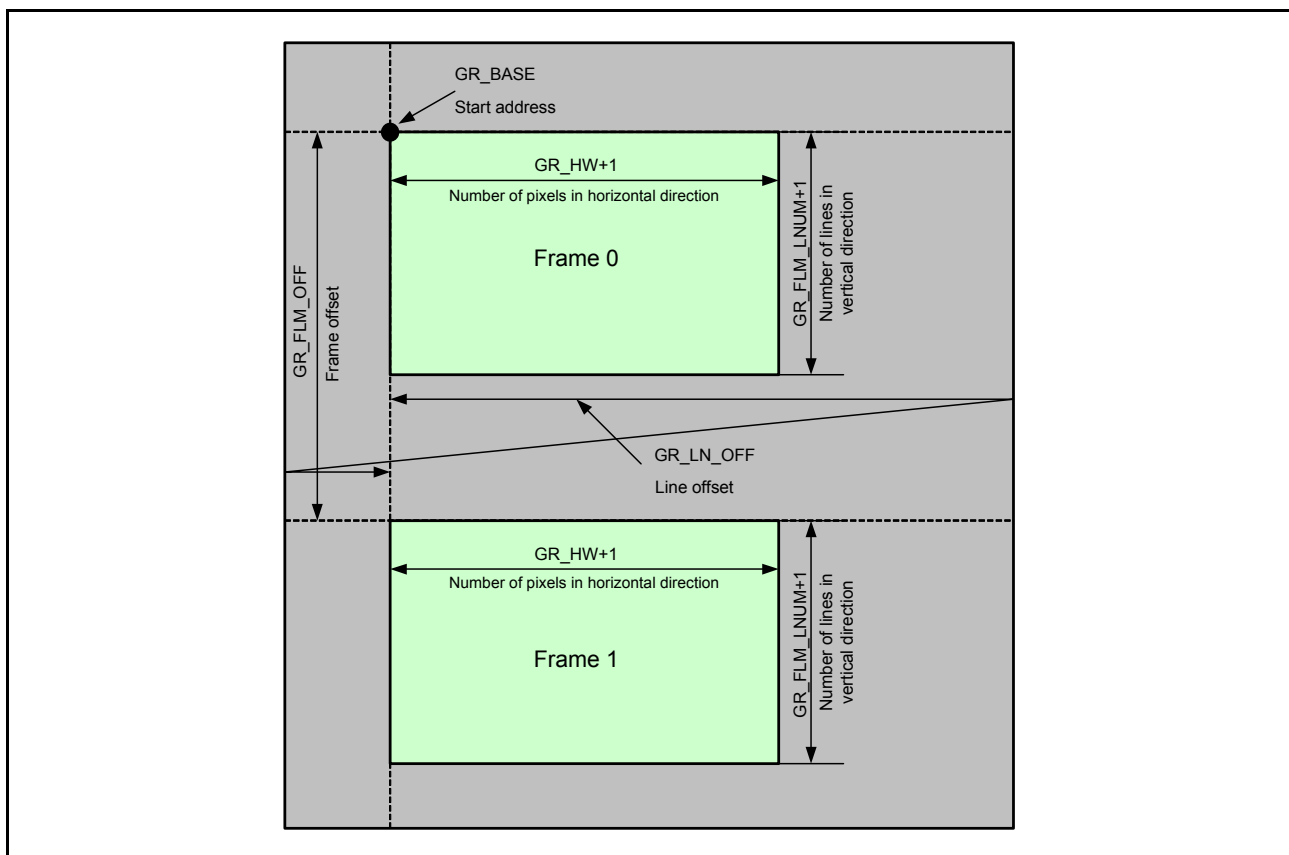
Register Name	Bit Name	Initial Value	Description
GR_FLM6	GR_HW[10:0]	0	Sets the width of the horizontal valid period. The width is (GR_HW + 1) pixels. Note: Set to 2 or greater.
GR_FLM5	GR_FLM_LNUM[10:0]	0	Sets the number of lines in a frame The number of lines is (GR_FLM_LNUM + 1).

## (7) Calculating Addresses in Frame Buffer

The data area in the frame buffer is defined using the addresses specified by GR\_BASE[31:0], GR\_LN\_OFF[14:0], and GR\_FLM\_OFF[22:0] bits and the display frame number.

The GR\_LN\_OFF[14:0] and GR\_FLM\_OFF[22:0] bits should be set in units of 32/128 bytes (the lower 5/7 bits should be fixed to 0).

The GR\_BASE[31:0] bits should be set in units of 64 bits to set the display data start position (the lower three bits should be fixed).



**Figure 35.2 Data Arrangement in Frame Buffer**

**Table 35.7 Calculation of Addresses in Frame Buffer**

Register Name	Bit Name	Initial Value	Description
GR_FLM2	GR_BASE[31:0]	0	<p>Frame Buffer Base Address</p> <p>Sets the start address of the frame buffer where frame data is to be stored.</p> <p>GR_BASE[4:3] and GR_BASE[6:3] are referred to during 32-byte burst transfer and 128-byte burst transfer, respectively, to skip the start line data.</p> <p>The lower 3 bits should be set to 000.</p>
GR_FLM3	GR_LN_OFF[14:0]	0	<p>Frame Buffer Line Offset Address</p> <p>Sets the line offset address for calculating the start address of each line.</p> <p>Line 0: GR_BASE</p> <p>Line 1: GR_BASE + GR_LN_OFF × 1</p> <p>⋮</p> <p>Line n: GR_BASE + GR_LN_OFF × n</p> <p>For 32-byte transfer, the lower 5 bits should be fixed to 0_0000.</p> <p>For 128-byte transfer, the lower 7 bits should be fixed to 000_0000.</p>
GR_FLM4	GR_FLM_OFF[22:0]	0	<p>Frame Buffer Frame Offset Address (lower)</p> <p>Specifies the frame offset address used for calculating the start address of each frame buffer when more than one buffer is used.</p> <p>Buffer 0: GR_BASE</p> <p>Buffer 1: GR_BASE + GR_FLM_OFF × 1</p> <p>⋮</p> <p>Buffer n: GR_BASE + GR_FLM_OFF × n</p> <p>For 32-byte transfer, the lower 5 bits should be fixed to 0_0000.</p> <p>For 128-byte transfer, the lower 7 bits should be fixed to 000_0000.</p>

### (8) Setting Frame Buffer Size Smaller than One Frame

Frame buffer size can be set in one-line units.

When the number of lines set with the GR\_FLM\_LOOP[10:0] bits is smaller than the value of the GR\_FLM\_LNUM[10:0] bits, data is again read from the start address of the frame buffer after the number of lines set with the (GR\_FLM\_LOOP[10:0] + 1) bits have been read.

**Table 35.8 Setting of Frame Buffer Size Smaller than One Frame**

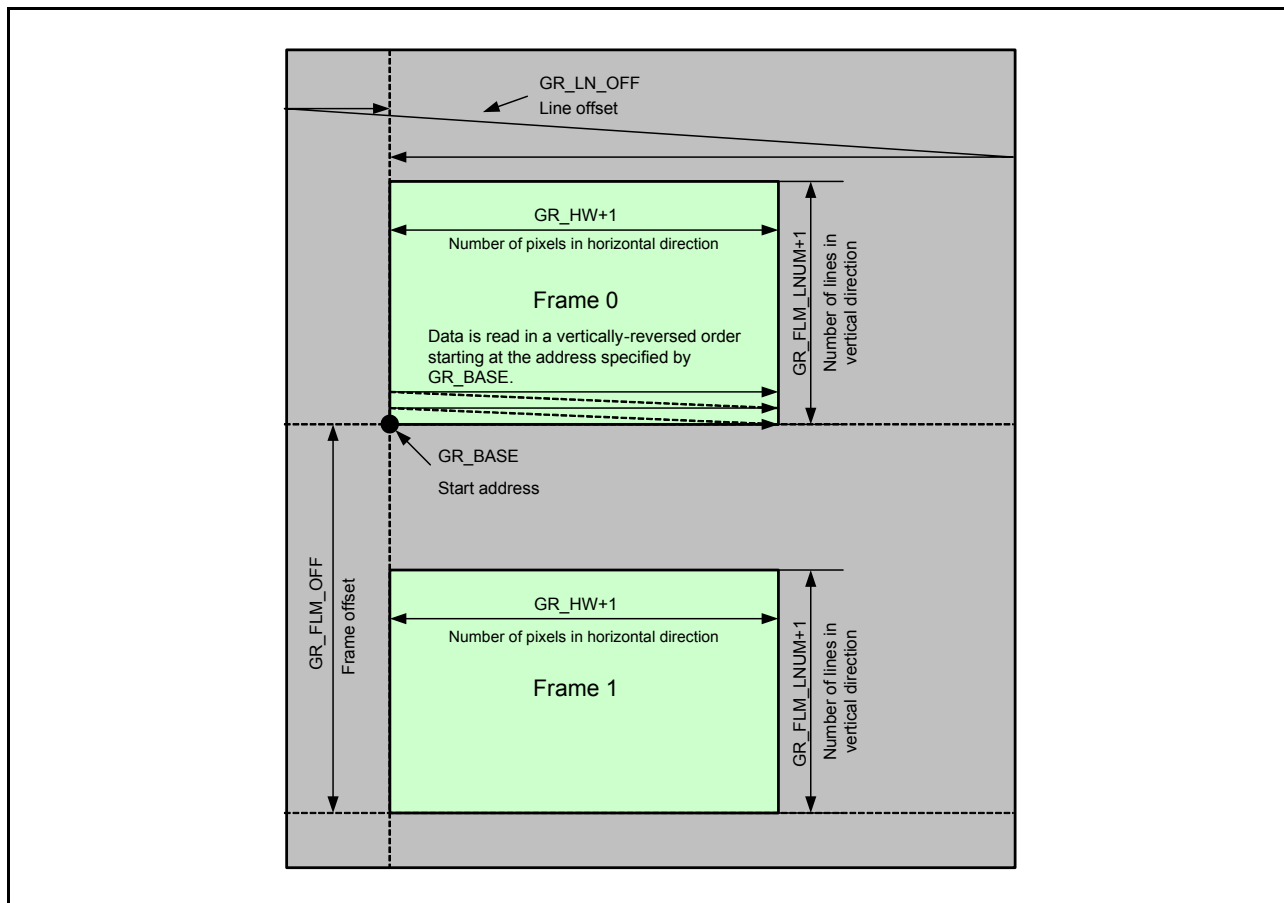
Register Name	Bit Name	Initial Value	Description
GR_FLM5	GR_FLM_LOOP[10:0]	1023	<p>Number of lines when reading the addresses repeatedly by returning to the start address after reaching the end address.</p> <p>The number of lines is (GR_FLM_LOOP + 1).</p>

(9) Line Offset Control for Frame Buffer

The following bit sets the line offset address direction of the frame buffer.

**Table 35.9 Line Offset Address Direction Control for Frame Buffer**

Register Name	Bit Name	Initial Value	Description
GR_FLM1	GR_LN_OFF_DIR	0	Selects the line offset address direction of the frame buffer. 0: Increments the address by the line offset address. 1: Decrements the address by the line offset address.



**Figure 35.3 Data Arrangement with Line Offset and Decrement Control**

(10) Selecting Format of Frame Buffer Read Signal

Signal formats RGB565, RGB888,  $\alpha$ RGB1555,  $\alpha$ RGB4444,  $\alpha$ RGB8888, RGB $\alpha$ 5551, RGB $\alpha$ 8888, CLUT8, CLUT4 and CLUT1 are supported for the graphics 0, 1, 2, 3, and OIR processes. The YCbCr422 and YCbCr444 formats are also supported for the graphics 0 and 1 processes.

The GR\_FORMAT[3:0] bits select a signal format.

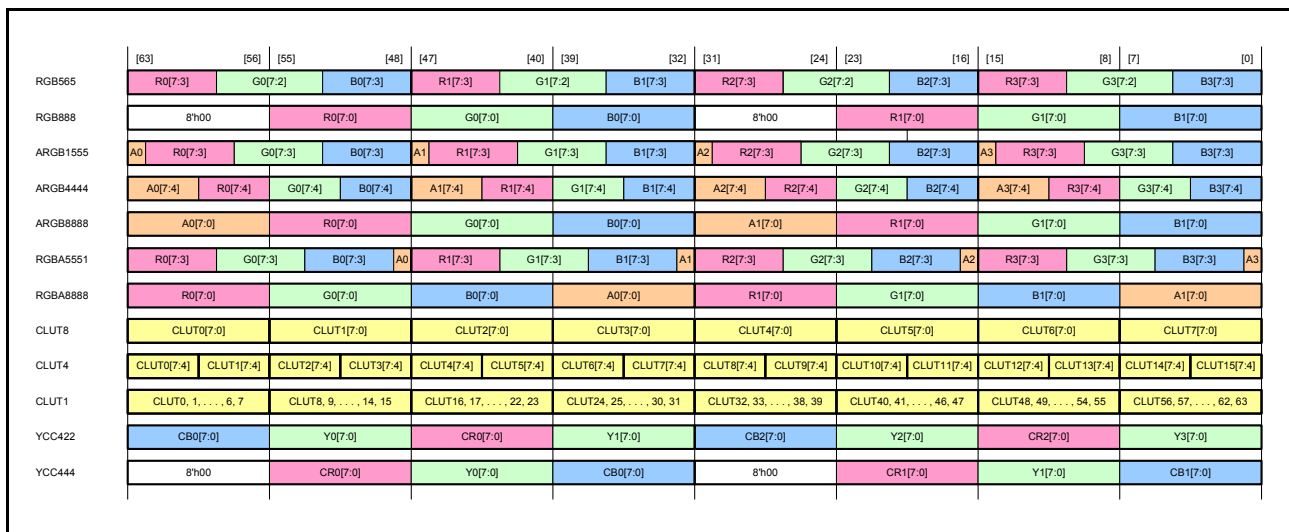
**Table 35.10 Format Selection for Frame Buffer Read Signal**

Register Name	Bit Name	Initial Value	Description
GR_FLM6	GR_FORMAT[3:0]	0	Sets the format of the frame buffer read signal. 0: RGB565 1: RGB888 2: $\alpha$ RGB1555 3: $\alpha$ RGB4444 4: $\alpha$ RGB8888 5: CLUT8 6: CLUT4 7: CLUT1 8: YCbCr422 or setting prohibited * 9: YCbCr444 or setting prohibited * 10: RGB $\alpha$ 5551 11: RGB $\alpha$ 8888 12 to 15: Setting prohibited

Note: \* Setting this value selects YCbCr422 and YCbCr444 for the graphics 0 and 1 processes, and is prohibited for the graphics 2, 3, and OIR processes.

(11) Endian Control

In the frame buffer, data is handled in 64-bit units, and endian of the data to be read can be controlled by setting the GR\_RDSWA[2:0] bits. Bit 0 of these bits indicates whether 8-bit data is swapped. Bit 1 indicates whether 16-bit data is swapped. Bit 2 indicates whether 32-bit data is swapped. In the YCbCr422 format, data can be arranged with the GR\_YCC\_SWAP[2:0] bits.



**Figure 35.4 Data Arrangement with Endian Control Disabled (GR\_RDSWA = 000)**



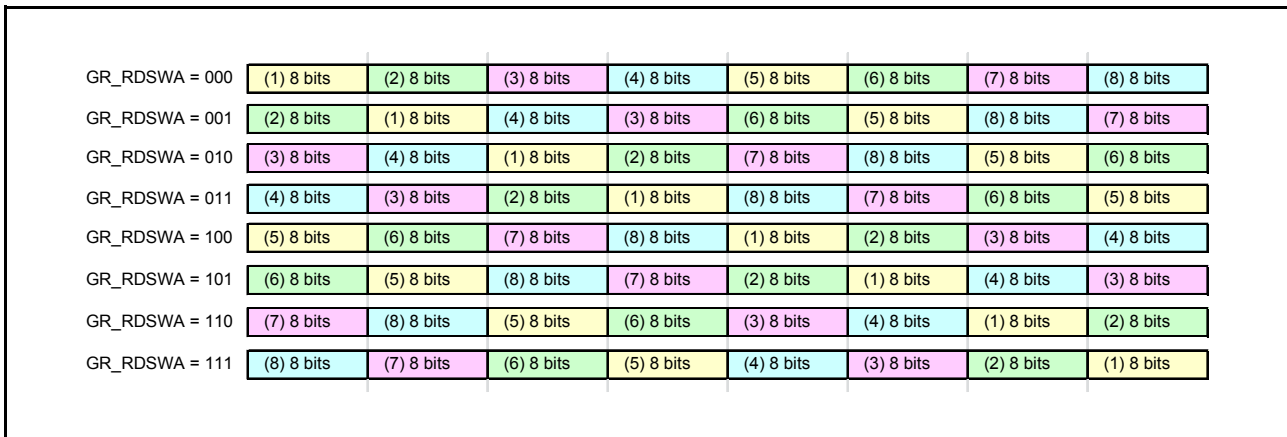


Figure 35.5 Data Arrangement with Endian Control Enabled



Figure 35.6 YCbCr422 Data Arrangement with Swapping Enabled

Table 35.11 Endian Control

Register Name	Bit Name	Initial Value	Description
GR_FLM6	GR_RDSWA[2:0]	0	<p>Sets 8-, 16-, and 32-bit swap.</p> <p>These three bits specify the method for swapping the bits of frame buffer read data as follows.</p> <p>Bit 0</p> <p>0: 8 bits are not swapped.</p> <p>1: 8 bits are swapped.</p> <p>Bit 1</p> <p>0: 16 bits are not swapped.</p> <p>1: 16 bits are swapped.</p> <p>Bit 2</p> <p>0: 32 bits are not swapped.</p> <p>1: 32 bits are swapped.</p> <p>When eight bits are put together, they are swapped as follows. Each of (1) to (8) indicates eight-bit data.</p> <p>000: (1) (2) (3) (4) (5) (6) (7) (8) [No swap]</p> <p>001: (2) (1) (4) (3) (6) (5) (8) (7) [8-bit swap]</p> <p>010: (3) (4) (1) (2) (7) (8) (5) (6) [16-bit swap]</p> <p>011: (4) (3) (2) (1) (8) (7) (6) (5) [16-bit swap + 8-bit swap]</p> <p>100: (5) (6) (7) (8) (1) (2) (3) (4) [32-bit swap]</p> <p>101: (6) (5) (8) (7) (2) (1) (4) (3) [32-bit swap + 8-bit swap]</p> <p>110: (7) (8) (5) (6) (3) (4) (1) (2) [32-bit swap + 16-bit swap]</p> <p>111: (8) (7) (6) (5) (4) (3) (2) (1) [32-bit swap + 16-bit swap + 8-bit swap]</p>

**Table 35.11 Endian Control**

Register Name	Bit Name	Initial Value	Description
GR_FLM6	GR_YCC_SWAP [2:0]	0	Controls swapping of data read from buffer in the YCbCr422 format. * 0: Cb/Y0/Cr/Y1 1: Y0/Cb/Y1/Cr 2: Cr/Y0/Cb/Y1 3: Y0/Cr/Y1/Cb 4: Y1/Cr/Y0/Cb 5: Cr/Y1/Cb/Y0 6: Y1/Cb/Y0/Cr 7: Cb/Y1/Cr/Y0

Note: \* These bits are supported for the graphics 0 and 1 processes only.

### (12) Display Start Pixel Setting for Read Data

When a horizontal offset is applied to display the image data in the frame buffer, the display start pixel is set with the GR\_BASE[31:0] and GR\_STA\_POS[5:0] bits. Calculation of the values for the GR\_BASE[31:0] and GR\_STA\_POS[5:0] bits depends on the signal format. The display start pixel can be calculated with the formulas in the table below, where H\_OFF is a horizontal offset from the display start pixel.

**Table 35.12 Calculation of Display Start Position for Various Signal Formats**

Signal Format of Video/Graphics	Number of Bits per Pixel	Calculation Formula *1
RGB888 αRGB8888, RGBα8888 YCbCr422*2 YCbCr444*3	32	GR_BASE[31:3] = t (H_OFF ÷ 2) GR_STA_POS[5:0] = mod (H_OFF ÷ 2)
RGB565 αRGB1555, RGBα5551 αRGB4444	16	GR_BASE[31:3] = int (H_OFF ÷ 4) GR_STA_POS[5:0] = mod (H_OFF ÷ 4)
CLUT8	8	GR_BASE[31:3] = int (H_OFF ÷ 8) GR_STA_POS[5:0] = mod (H_OFF ÷ 8)
CLUT4	4	GR_BASE[31:3] = int (H_OFF ÷ 16) GR_STA_POS[5:0] = mod (H_OFF ÷ 16)
CLUT1	1	GR_BASE[31:3] = int (H_OFF ÷ 64) GR_STA_POS[5:0] = mod (H_OFF ÷ 64)

- Notes:
1. The functions int() and mod() output a quotient and a remainder, respectively.
  2. The YCbCr422 format is not supported for the graphics 2, 3, and OIR processes. In the YCbCr422 format, 32 bits are used for two pixels (Cb, Y0, Cr, and Y1 components). Therefore, the start position is controlled in units of 32 bits.
  3. The YCbCr444 format is not supported for the graphics 2, 3, and OIR processes.

**Table 35.13 Setting of Display Start Pixel of Read Data**

Register Name	Bit Name	Initial Value	Description
GR_FLM6	GR_STA_POS[5:0]	0	Sets the amount of data to be skipped through. Specifically data amount equal to the amount indicated by GR_STA_POS is skipped from the start of the line.
GR_FLM2	GR_BASE[31:0]	0	Frame Buffer Base Address Sets the start address of the frame buffer where frame data is to be stored. GR_BASE[4:3] and GR_BASE[6:3] are referred to during 32-byte burst transfer and 128-byte burst transfer, respectively, to skip the start line data. The lower 3 bits should be fixed to 000.

**(13) YCbCr422 to YCbCr444 Conversion**

Data format for the graphics 0 and 1 processes are converted from YCbCr422 to YCbCr444.

This function is not supported for the graphics 2, 3, and OIR processes.

**Table 35.14 YCbCr422 to YCbCr444 Conversion**

Register Name	Bit Name	Initial Value	Description
GR1_FLM6	GR1_CNV444_MD	0	Sets the interpolation mode for YCbCr422 to YCbCr444 conversion. * 0: Hold interpolation 1: Average interpolation

Note: \* This register is not provided for the graphics 2, 3, and OIR processes, for which the YCbCr422 format is not supported.

#### (14) Bit Extension

When the value of the GR\_FORMAT[3:0] bits is 0 to 3, the RGB565, RGB888,  $\alpha$ RGB1555, and  $\alpha$ RGB4444 formats are converted to the  $\alpha$ RGB8888 format. When the value of the GR\_FORMAT[3:0] bits is 10, the RGB $\alpha$ 5551 format is converted to the RGB $\alpha$ 8888 format. The RGB $\alpha$ 5551 to RGB $\alpha$ 8888 format conversion is omitted because it differs from the  $\alpha$ RGB5551 to  $\alpha$ RGB8888 format conversion only in the position of  $\alpha$ .

- RGB565 to  $\alpha$ RGB8888 Format Conversion

After conversion,  $\alpha[7:0]$  is fixed to 255.

After conversion,  $R[7:0] = R[4:0] \times 263 \div 32$  (round off to an integer), approximation of  $\#R[4:0] \times 255 \div 31$

After conversion  $G[7:0] = G[5:0] \times 259 \div 64$  (round off to an integer), approximation of  $\#G[5:0] \times 255 \div 63$

After conversion,  $B[7:0] = B[4:0] \times 263 \div 32$  (round off to an integer), approximation of  $\#B[4:0] \times 255 \div 31$

- RGB888 to  $\alpha$ RGB8888 Format Conversion

After conversion,  $\alpha[7:0]$  is fixed to 255.

- $\alpha$ RGB1555 to  $\alpha$ RGB8888 Format Conversion

After conversion,  $\alpha[7:0]$  is GR\_A1 when  $\alpha$  input is 1, and GR\_A0 when 0.

After conversion,  $R[7:0] = R[4:0] \times 263 \div 32$  (round off to an integer), approximation of  $\#R[4:0] \times 255 \div 31$

After conversion,  $G[7:0] = G[4:0] \times 263 \div 32$  (round off to an integer), approximation of  $\#G[4:0] \times 255 \div 31$

After conversion,  $B[7:0] = B[4:0] \times 263 \div 32$  (round off to an integer), approximation of  $\#B[4:0] \times 255 \div 31$

- $\alpha$ RGB4444 to  $\alpha$ RGB8888 Format Conversion

After conversion,  $\alpha[7:0] = \alpha[3:0] \times 17$

After conversion,  $R[7:0] = R[3:0] \times 17$

After conversion,  $G[7:0] = G[3:0] \times 17$

After conversion,  $B[7:0] = B[3:0] \times 17$

#### (15) Buffer Underflow Processing

When data read from the frame buffer cannot be completed due to bus-traffic related problems, an underflow interrupt signal is output.

### 35.1.3 Setting Graphics Display Area

The graphics display area is set with the GR\_GRC\_HS[10:0], GR\_GRC\_HW[10:0], GR\_GRC\_VS[10:0], and GR\_GRC\_VW[10:0] bits based on the rising edges of the Hsync and Vsync signals.

Figure 35.7 shows the graphics display area.

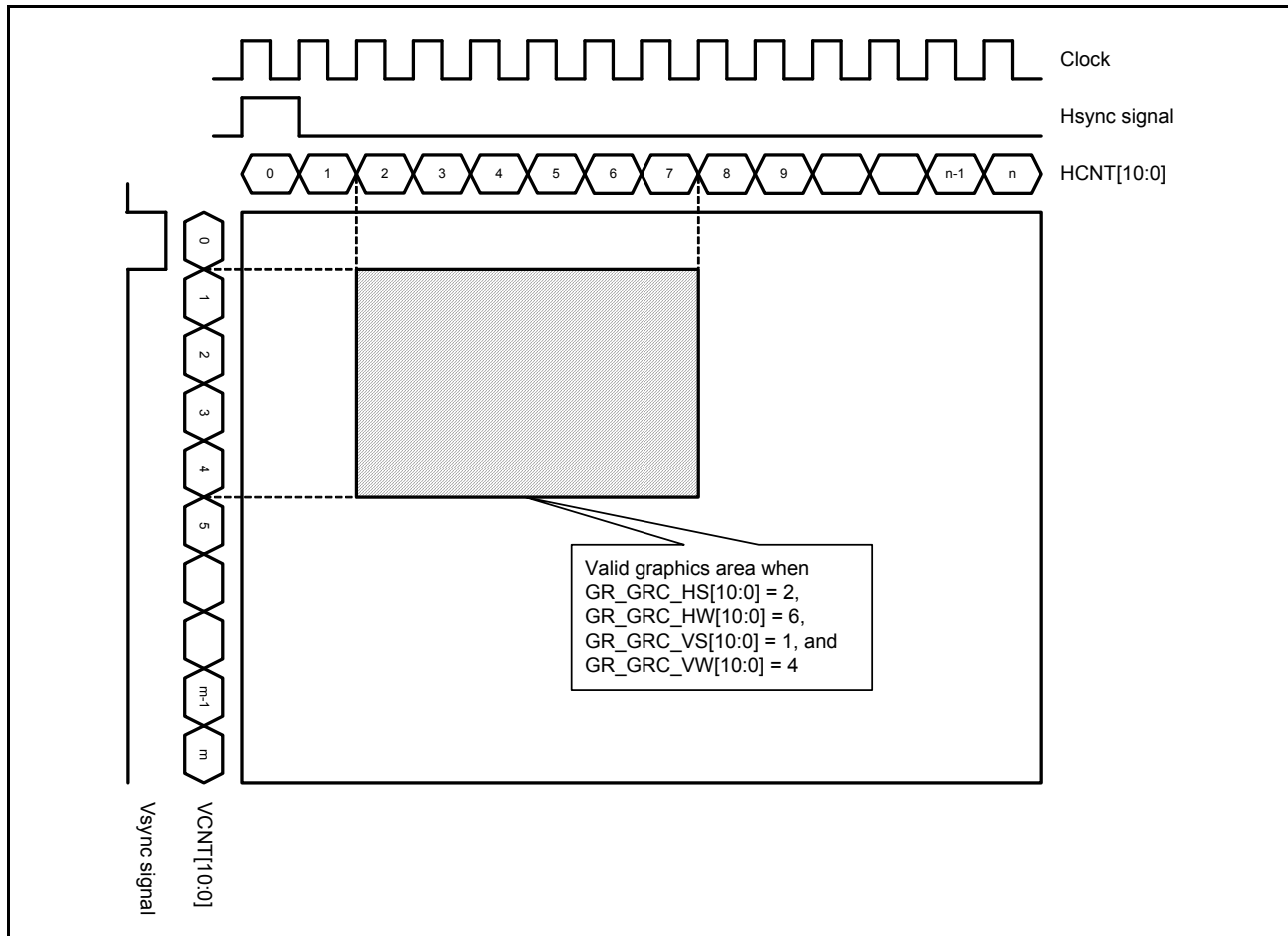


Figure 35.7 Graphics Display Area

The frame line of the graphics area can be displayed by setting the GR\_GRC\_DISP\_ON bit to 1.

Table 35.15 Graphics Image Area Setting

Register Name	Bit Name	Initial Value	Description
GR_AB3	GR_GRC_HS[10:0]	0	Sets the horizontal start position of the graphics image area. Note: Set to 16 or greater clocks and the result of GR_GRC_HS + GR_GRC_HW should be smaller than or equal to 2015 clocks.
GR_AB3	GR_GRC_HW[10:0]	0	Sets the horizontal width of the graphics image area. Note: For displaying an image with 1- or 2-pixel horizontal width, set GR_GRC_HS to 2 and GR_GRC_HW to 1 (1-pixel) or 2 (2-pixel).
GR_AB2	GR_GRC_VS[10:0]	0	Sets the vertical start position of the graphics image area. Note: Set to 4 or greater lines and the result of GR_GRC_VS + GR_GRC_VW should be smaller than or equal to 2039 lines.
GR_AB2	GR_GRC_VW[10:0]	0	Sets the vertical width of the graphics image area.
GR_AB1	GR_GRC_DISP_ON	0	Turns on/off frame-line display of the graphics image area. 0: Frame-line display off 1: Frame-line display on

### 35.1.4 Interrupt Generation at Specified Line

An interrupt signal can be generated at the line specified with the GR\_LINE[10:0] bits.

**Table 35.16 Interrupt Generation at Specified Line**

Register Name	Bit Name	Initial Value	Description
GR_CLUT_INT	GR_LINE[10:0]	0	Line Interrupt Set * When the number of lines matches the value of the GR_LINE bits, an interrupt signal is output. This function is supported for the graphics 3 and OIR processes only. This function supported for the graphics 3 process is enabled even when the graphics 3 process is not used. This function supported for the graphics OIR process is enabled only when the output image generator is enabled.

Note: \* This function is supported for the graphics 3 and OIR processes only; these bits are not supported for the graphics 0, 1, and 2 processes.

### 35.1.5 Formats of Frame Buffer Read Signals and Corresponding Alpha Blending Types

Setting the GR\_FORMAT[3:0] bits selects the format of the signal read from the frame buffer.

Table 35.17 shows the signal formats and the corresponding alpha blending types. The priority of the alpha value is: alpha blending in rectangular area > chroma-key processing > alpha blending in pixel units.

**Table 35.17 Formats of Frame Buffer Read Signal and Corresponding Alpha Blending Types**

GR_FORMAT[3:0]	Signal Format	Alpha Blending in Rectangular Area	RGB-Index Chroma-Key Processing	CLUT-Index Chroma-Key Processing	Alpha Blending in Pixel Units
0	RGB565	Supported	Supported *1	Not supported	Not supported *2
1	RGB888	Supported	Supported	Not supported	Not supported *2
2	$\alpha$ RGB1555	Supported	Supported *1*3	Not supported	Supported *3
3	$\alpha$ RGB4444	Supported	Supported *1	Not supported	Supported
4	$\alpha$ RGB8888	Supported	Supported	Not supported	Supported
5	CLUT8	Supported	Not supported	Supported	Supported
6	CLUT4	Supported	Not supported	Supported	Supported
7	CLUT1	Supported *4	Not supported	Supported *4	Supported *4
8	YCbCr422	Not supported *5	Not supported *5	Not supported *5	Not supported *5
9	YCbCr444	Not supported *5	Not supported *5	Not supported *5	Not supported *5
10	RGB $\alpha$ 5551	Supported	Supported *1*3	Not supported	Supported *3
11	RGB $\alpha$ 8888	Supported	Supported	Not supported	Supported

- Notes:
- When each color component of the RGB signal read from the frame buffer is not 8 bits, it is converted to 8 bits by calculation in RGB-index chroma-key processing. (See section 35.1.2 (14) Bit Extension.)
  - Since  $\alpha$  value is 255, the current graphics is always displayed.
  - $\alpha$  value for data read from the frame buffer is specified with one bit. This one-bit signal selects one of the two registers, each of which holds an 8-bit  $\alpha$  value.
  - CLUT value for the frame buffer signal is specified with one bit. This one-bit signal selects one of the two registers, each of which holds the  $\alpha$ , G, B, and R values (8 bits for each value). The CLUT table is not referenced.
  - YCbCr422 and YCbCr444 are supported for the graphics 0 and 1 processes, but any type of blending and chroma-key processing cannot be used.

### 35.1.6 Display Selection

The GR\_DISP\_SEL[1:0] bits select the graphics to be displayed from the background color, the lower-layer graphics, the current graphics, and the blended image of the lower-layer graphics and the current graphics. For blending, alpha blending in a rectangular area, multiplication with current alpha at alpha blending in a rectangular area, RGB-index chroma-key processing, CLUT-index chroma-key processing, alpha blending in one-pixel units, or premultiplication at alpha blending in one-pixel units can be selected. Only alpha blending in a rectangular area can be selected in the VIN synthesizer.

Table 35.18 shows the settings for various display types.

**Table 35.18 Settings for Various Display Types**

GR_DISP_SEL [1:0]	GR_ARC_ON	GR_CK_ON	GR_ARC_MUL	GR_ACALC_MD	Processing for Graphics Area	Processing for the Area outside the Graphics Area
0	—	—	—	—	Background color	Background color
1	—	—	—	—	Lower-layer graphics	Lower-layer graphics
2	—	—	—	—	Current graphics	Background color
3	1	—	0	0	Alpha blending in a rectangular area*1	Lower-layer graphics
3	1	—	0	1	Setting prohibited	
3	1	—	1	0	Multiplication with current alpha at alpha blending in a rectangular area*2	Lower-layer graphics
3	1	—	1	1	Multiplication with current alpha at alpha blending in a rectangular area with alpha premultiplied*2	Lower-layer graphics
3	0	1	—	—	RGB-index or CLUT-index chroma-key processing*3	Lower-layer graphics
3	0	0	—	0	Alpha blending in one-pixel units*2	Lower-layer graphics
3	0	0	—	1	Premultiplication at alpha blending in one-pixel units*2	Lower-layer graphics

- Notes:
1. The alpha blending in a rectangular area is not supported for the graphics 0 process in the scaler and the graphics OIR process. When this processing is selected in the VIN synthesizer, cascaded connection cannot be selected in the scaler.
  2. The multiplication with current alpha at alpha blending in a rectangular area and alpha blending function in one-pixel units are supported for the graphics 1, 2, and 3 processes only.
  3. The RGB-index or CLUT-index chroma-key processing is not supported for the VIN synthesizer.

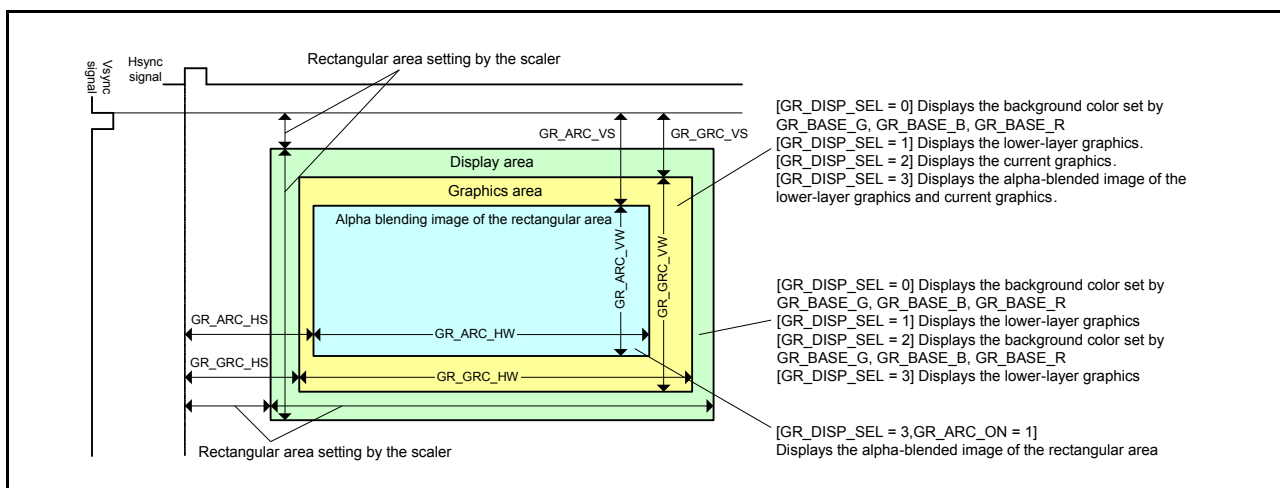


Figure 35.8 Graphic Display Types

Figure 35.9 shows the graphics planes displayed when the GR\_DISP\_SEL bits are set to 3.

For correspondence between the lower-layer graphics and the current graphics, see Figure 35.1.

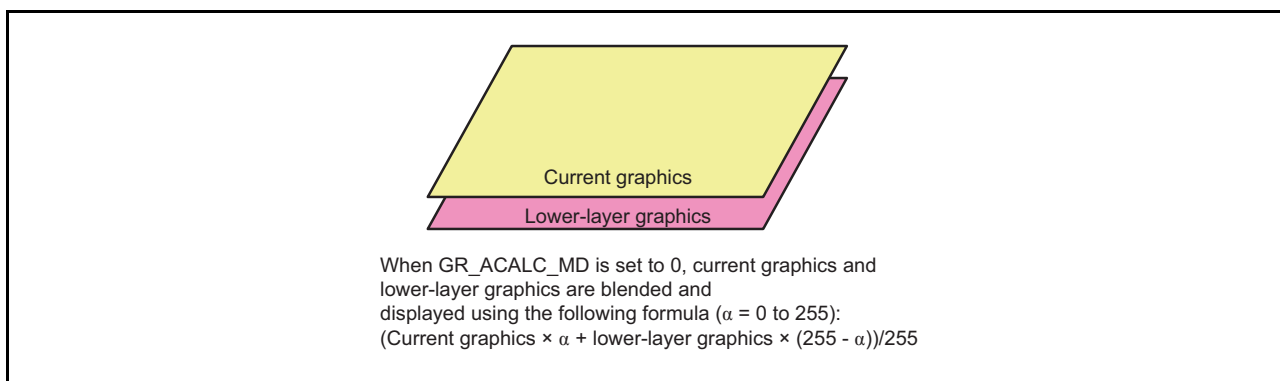


Figure 35.9 Graphics Planes with GR\_DISP\_SEL Set to 3

Table 35.19 Alpha Blending Setting

Register Name	Bit Name	Initial Value	Description
GR_AB1	GR_DISP_SEL [1:0]	0	Selects the graphics display mode. 0:Background color display 1:Lower-layer graphics display 2:Current graphics display 3:Blended display of lower-layer graphics and current graphics*1
GR_AB1	GR_ARC_ON	0	Turns on/off alpha blending in a rectangular area.*2 0: Off 1: On
GR_AB1	GR_ARC_MUL	0	Turns on/off multiplication processing with current alpha at alpha blending in a rectangular area *3 0: Off 1: On
GR_AB1	GR_ACALC_MD	0	Turns on/off premultiplication processing at alpha blending in one-pixel units *3 0: Off 1: On
GR_AB7	GR_CK_ON	0	Turns on/off CLUT-index/RGB-index chroma-key processing. *4 0: Off 1: On



- Notes:
1. The graphics 0 and OIR processes support the chroma-key processing only. When performing chroma-key processing, set the  $\alpha$  value for converting the pixels to be subjected to chroma-key processing, and the  $\alpha$  value of the pixels not to be subjected to the chroma-key processing to 255 to display the current graphics only. The VIN synthesizer supports display processing with alpha blending in a rectangular area only.
  2. This function is supported only for the graphics 1, 2, and 3 processes and the VIN synthesizer. This bit is not provided for the graphics 0 and OIR processes.
  3. This function is supported only for the graphics 1, 2, and 3 processes. This bit is not provided for the graphics 0 and OIR processes and the VIN synthesizer.
  4. This bit is not provided for the VIN synthesizer.

### 35.1.7 Background Color Display Processing

The color set with the GR\_BASE\_G[7:0], GR\_BASE\_B[7:0], and GR\_BASE\_R[7:0] bits is displayed.

G output = GR\_BASE\_G

B output = GR\_BASE\_B

R output = GR\_BASE\_R

**Table 35.20 Background Color Setting**

Register Name	Bit Name	Initial Value	Description
GR_BASE	GR_BASE_G[7:0]	0	Background color G signal G: 8 bits; unsigned (0 to 255 [LSB])
GR_BASE	GR_BASE_B[7:0]	0	Background color B signal B: 8 bits; unsigned (0 to 255 [LSB])
GR_BASE	GR_BASE_R[7:0]	0	Background color R signal R: 8 bits; unsigned (0 to 255 [LSB])

### 35.1.8 Lower-Layer Graphics Display Processing

The lower-layer graphics are displayed as follows:

G output = G input of lower-layer graphics

B output = B input of lower-layer graphics

R output = R input of lower-layer graphics

### 35.1.9 Current Graphics Display Processing

The current graphics are displayed as follows:

G output = G input of current graphics

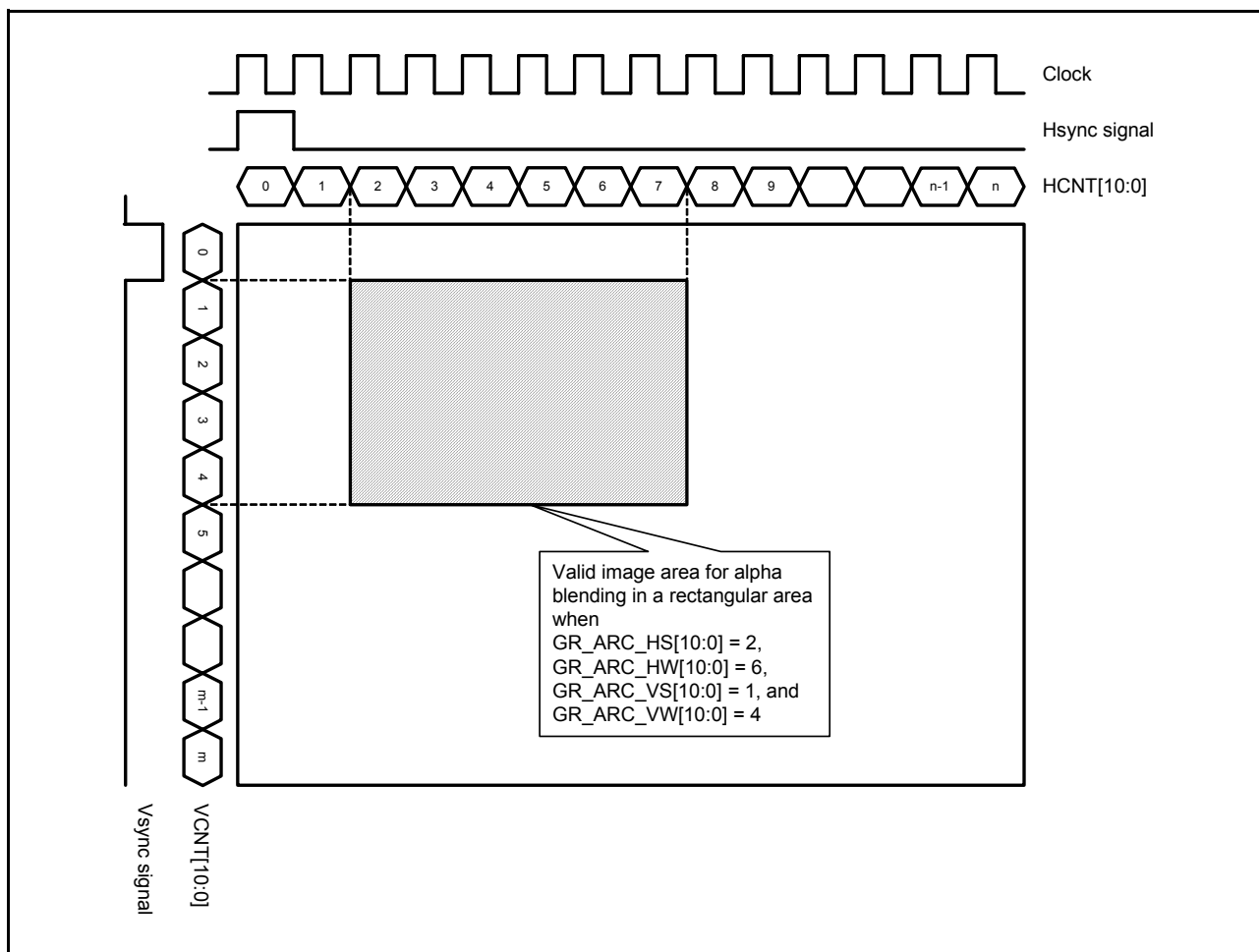
B output = B input of current graphics

R output = R input of current graphics

### 35.1.10 Display with Alpha Blending in a Rectangular Area

The rectangular area subjected to alpha blending is set with the GR\_ARC\_HS[10:0], GR\_ARC\_HW[10:0], GR\_ARC\_VS[10:0], and GR\_ARC\_VW[10:0] bits based on the rising edges of the Hsync and Vsync signals. This function is not supported for the graphics 0 and OIR processes.

Figure 35.10 shows the rectangular area setting for alpha blending.



**Figure 35.10** Rectangular Area Setting for Alpha Blending

The frame line of graphics area can be displayed by setting the GR\_ARC\_DISP\_ON bit to 1.

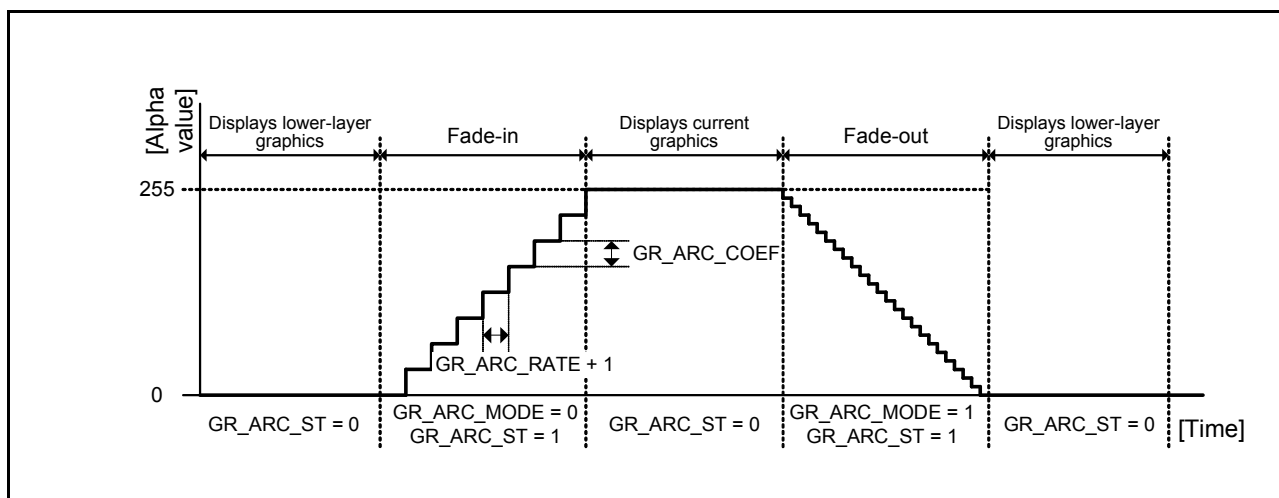
**Table 35.21 Setting of Rectangular Area for Alpha Blending**

Register Name	Bit Name	Initial Value	Description
GR_AB5	GR_ARC_HS[10:0]	0	Sets the horizontal start position of the valid image area for alpha blending in a rectangular area.
GR_AB5	GR_ARC_HW[10:0]	0	Sets the horizontal width of the valid image area for alpha blending in a rectangular area.
GR_AB4	GR_ARC_VS[10:0]	0	Sets the vertical start position of the valid image area for alpha blending in a rectangular area.
GR_AB4	GR_ARC_VW[10:0]	0	Sets the vertical width of the valid image area for alpha blending in a rectangular area.
GR_AB1	GR_ARC_DISP_ON	0	Turns on/off frame-line display of the valid image area for alpha blending in a rectangular area. 0: Frame-line display off 1: Frame-line display on

In alpha blending in a rectangular area, the current graphics are faded in or out by setting the fade-in or fade-out coefficients with the GR\_ARC\_DEF[7:0], GR\_ARC\_MODE, GR\_ARC\_COEF[7:0], and GR\_ARC\_RATE[7:0] bits.

First, the value of the GR\_ARC\_DEF[7:0] bits is assigned to the  $\alpha$  value.

Then, each time the Vsync signal rises for the number of times set with the GR\_ARC\_RATE[7:0] bits + 1, the value of the GR\_ARC\_COEF[7:0] bit is added to or subtracted from the  $\alpha$  value.

**Figure 35.11 Fade In and Fade Out****Table 35.22 Setting for Alpha Blending in a Rectangular Area**

Register Name	Bit Name	Initial Value	Description
GR_AB7	GR_ARC_DEF[7:0]	0	Sets the initial alpha value for alpha blending in a rectangular area.
GR_AB6	GR_ARC_MODE	0	Alpha Blending Mode in Rectangular Area 0: Addition 1: Subtraction
GR_AB6	GR_ARC_COEF[7:0]	0	Sets the alpha coefficient for alpha blending in a rectangular area. (0 to 255) [7:0]: Variation (absolute value)
GR_AB6	GR_ARC_RATE[7:0]	0	Sets the value obtained by subtracting 1 from the frame rate for alpha blending in a rectangular area.
GR_MON	GR_ARC_ST	—	Status Flag for Alpha Blending in Rectangular Area 0: Addition or subtraction has been completed. ( $\alpha$ value is 0 or 255) 1: Addition or subtraction is in progress.

The values specified with the following expressions are used in the alpha blending calculation described in section 35.1.14, Alpha Blending Calculation.

$\alpha$  value = Fade-in/out coefficient

G value = G input of the current graphics

B value = B input of the current graphics

R value = R input of the current graphics

### 35.1.11 RGB-Index Chroma-Key Processing

The pixels that satisfy all the expressions below are subjected to RGB-index chroma-key processing.

G input of the current graphics = GR\_CK\_KG

B input of the current graphics = GR\_CK\_KB

R input of the current graphics = GR\_CK\_KR

In RGB-index chroma-key processing, the values specified with the following expressions are used in the alpha blending calculation described in section 35.1.14, Alpha Blending Calculation. This function is not supported in the VIN synthesizer.

$\alpha$  value = GR\_CK\_A

G value = GR\_CK\_G

B value = GR\_CK\_B

R value = GR\_CK\_R

For the pixels that are not subjected to RGB-index chroma-key processing, the values specified with the following expressions are used in the alpha blending calculation described in section 35.1.14, Alpha Blending Calculation.

$\alpha$  value =  $\alpha$  input of the current graphics

G value = G input of the current graphics

B value = B input of the current graphics

R value = R input of the current graphics

**Table 35.23 Setting for RGB-Index Chroma-Key Processing**

Register Name	Bit Name	Initial Value	Description
GR_AB8	GR_CK_KG[7:0]	0	G Signal for RGB-Index Chroma-Key Processing G: 8 bits; unsigned (0 to 255 [LSB])
GR_AB8	GR_CK_KB[7:0]	0	B Signal for RGB-Index Chroma-Key Processing B: 8 bits; unsigned (0 to 255 [LSB])
GR_AB8	GR_CK_KR[7:0]	0	R Signal for RGB-Index Chroma-Key Processing R: 8 bits; unsigned (0 to 255 [LSB])
GR_AB9	GR_CK_A[7:0]	0	Replaced Alpha Signal after RGB-Index Chroma-Key Processing* $\alpha$ : 8 bits; unsigned (0 to 255 [LSB])
GR_AB9	GR_CK_G[7:0]	0	Replaced G Signal after RGB-Index Chroma-Key Processing G: 8 bits; unsigned (0 to 255 [LSB])
GR_AB9	GR_CK_B[7:0]	0	Replaced B Signal after RGB-Index Chroma-Key Processing B: 8 bits; unsigned (0 to 255 [LSB])
GR_AB9	GR_CK_R[7:0]	0	Replaced R Signal after RGB-Index Chroma-Key Processing R: 8 bits; unsigned (0 to 255 [LSB])

Note: \* To use this function for the graphics 0 and OIR processes, the alpha value should be set to 255.

### 35.1.12 CLUT-Index Chroma-Key Processing

The pixels that satisfy the expression below are subjected to CLUT-index chroma-key processing.

$$\text{CLUT input of the current graphics} = \text{GR\_CK\_KCLUT}$$

In CLUT-index chroma-key processing, the values specified with the following expressions are used in the alpha blending calculation described in section 35.1.14, Alpha Blending Calculation. This function is not supported in the VIN synthesizer.

$$\alpha \text{ value} = \text{GR\_CK\_A}$$

$$\text{G value} = \text{GR\_CK\_G}$$

$$\text{B value} = \text{GR\_CK\_B}$$

$$\text{R value} = \text{GR\_CK\_R}$$

For the pixels that are not subjected to CLUT-index chroma-key processing, the values specified with the following expressions are used in the alpha blending calculation described in section 35.1.14, Alpha Blending Calculation.

$$\alpha \text{ value} = \alpha \text{ input of the current graphics}$$

$$\text{G value} = \text{G input of the current graphics}$$

$$\text{B value} = \text{B input of the current graphics}$$

$$\text{R value} = \text{R input of the current graphics}$$

**Table 35.24 Setting for CLUT-Index Chroma-Key Processing**

Register Name	Bit Name	Initial Value	Description
GR_AB8	GR_CK_ KCLUT[7:0]	0	CLUT Signal for CLUT-Index Chroma-Key Processing CLUT: 8 bits; unsigned (0 to 255 [LSB])
GR_AB10	GR_A0[7:0]	0	CLUT1 $\alpha$ 0 Signal* Replaced with $\alpha$ signal when in the CLUT1 format and CLUT1 = 0. Replaced with $\alpha$ signal when in the $\alpha$ RGB1555 format and $\alpha = 0$ .
GR_AB10	GR_G0[7:0]	0	CLUT1 G0 Signal Replaced with G signal when in the CLUT1 format and CLUT1 = 0.
GR_AB10	GR_B0[7:0]	0	CLUT1 B0 Signal Replaced with B signal when in the CLUT1 format and CLUT1 = 0.
GR_AB10	GR_R0[7:0]	0	CLUT1 R0 Signal Replaced with R signal when in the CLUT1 format and CLUT1 = 0.
GR_AB11	GR_A1[7:0]	0	CLUT1 $\alpha$ 1 Signal* Replaced with $\alpha$ signal when in the CLUT1 format and CLUT1 = 1. Replaced with $\alpha$ signal when in the $\alpha$ RGB1555 format and $\alpha = 1$ .
GR_AB11	GR_G1[7:0]	0	CLUT1 G1 Signal Replaced with G signal when in the CLUT1 format and CLUT1 = 1.
GR_AB11	GR_B1[7:0]	0	CLUT1 B1 Signal Replaced with B signal when in the CLUT1 format and CLUT1 = 1.
GR_AB11	GR_R1[7:0]	0	CLUT1 R1 Signal Replaced with R signal when in the CLUT1 format and CLUT1 = 1.

Note: \* To use this function for the graphics 0 and OIR processes, the alpha value should be set to 255.

### 35.1.13 Display with Alpha Blending in One-Pixel Units

In the alpha blending in one-pixel units, the values specified with the following expressions are used in the alpha blending calculation described in section 35.1.14, Alpha Blending Calculation. This function is not supported in the graphics 0 and OIR processes and the VIN synthesizer.

$\alpha$  value =  $\alpha$  input of the current graphics

G value = G input of the current graphics

B value = B input of the current graphics

R value = R input of the current graphics

### 35.1.14 Alpha Blending Calculation

Alpha blending of two input signals is performed using the  $\alpha$  value as described below.

[GR\_ACALC\_MD = 0]

$$\text{G output} = (\text{G value} \times \alpha \text{ value} + \text{G input of the lower-layer graphics} \times (255 - \alpha \text{ value})) \div 255$$

$$\text{B output} = (\text{B value} \times \alpha \text{ value} + \text{B input of the lower-layer graphics} \times (255 - \alpha \text{ value})) \div 255$$

$$\text{R output} = (\text{R value} \times \alpha \text{ value} + \text{R input of the lower-layer graphics} \times (255 - \alpha \text{ value})) \div 255$$

[GR\_ACALC\_MD = 1 (premultiplication)]

$$\text{G output} = (\text{G value} + \text{G input of the lower-layer graphics} \times (255 - \alpha \text{ value})) \div 255$$

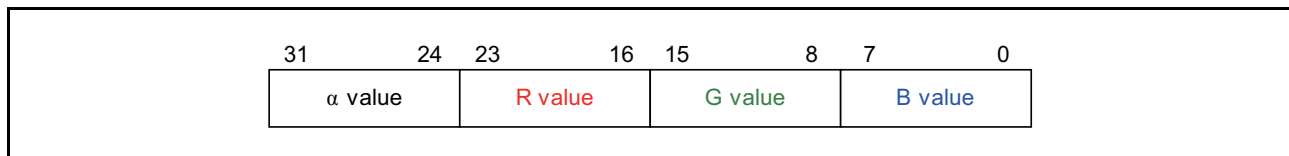
$$\text{B output} = (\text{B value} + \text{B input of the lower-layer graphics} \times (255 - \alpha \text{ value})) \div 255$$

$$\text{R output} = (\text{R value} + \text{R input of the lower-layer graphics} \times (255 - \alpha \text{ value})) \div 255$$

### 35.1.15 CLUT Table

When the signal format is CLUT8 or CLUT4, the format is converted to  $\alpha$ RGB8888 based on the CLUT table. When the format is CLUT1, it is converted to  $\alpha$ RGB8888 based on the register value.

Figure 35.12 shows data arrangement in the CLUT table.



**Figure 35.12 Data Arrangement in CLUT Table**

The CLUT tables are arranged in the following addresses (channel 0). For the arrangement in channel 1, see the register configuration described in section 35.2, Register Descriptions.

- Graphics 0 CLUT table: H'FCFF6000 to H'FCFF63FF  
(For CLUT4, addresses H'FCFF6000 to H'FCFF603F are valid.)
- Graphics 1 CLUT table: H'FCFF6400 to H'FCFF67FF  
(For CLUT4, addresses H'FCFF6400 to H'FCFF643F are valid.)
- Graphics 2 CLUT table: H'FCFF6800 to H'FCFF6BFF  
(For CLUT4, addresses H'FCFF6800 to H'FCFF683F are valid.)
- Graphics 3 CLUT table: H'FCFF6C00 to H'FCFF6FFF  
(For CLUT4, addresses H'FCFF6C00 to H'FCFF6C3F are valid.)
- Graphics OIR CLUT table: H'FCFF7000 to H'FCFF73FF  
(For CLUT4, addresses H'FCFF7000 to H'FCFF703F are valid.)

Two CLUT tables (CLUT table 0, CLUT table 1) on the different planes are allocated to the same address and one of the tables is selected with the GR\_CLT\_SEL bit. This allows rewriting one CLUT table when this module refers to the other

CLUT table.

**Table 35.25 CLUT Table Selection**

Register Name	Bit Name	Initial Value	Description
GR_CLUT	GR_CLT_SEL	0	CLUT Table Select Signal 0: Selects CLUT table 0. The format is converted to $\alpha$ RGB8888 based on the CLUT table 0. CLUT table 1 can be read from or written to by the CPU. 1: Selects CLUT table 1. The format is converted to $\alpha$ RGB8888 based on the CLUT table 1. CLUT table 0 can be read from or written to by the CPU.

### 35.1.16 Multiplication Processing with Current Alpha at Alpha Blending in Rectangular Area

In multiplication processing with current alpha at alpha blending in a rectangular area, the values specified with the following expressions are used in the alpha blending calculation described in section 35.1.14, Alpha Blending Calculation.

[GR\_ACALC\_MD = 0]

$\alpha$  value = Fade-in/out coefficient

G value = G input of the current graphics

B value = B input of the current graphics

R value = R input of the current graphics

[GR\_ACALC\_MD = 1 (premultiplication)]

$\alpha$  value = Fade-in/out coefficient  $\times$   $\alpha$  input of current graphics

G value = G input of the current graphics

B value = B input of the current graphics

R value = R input of the current graphics

### 35.1.17 Selection of Lower-Layer/Current Graphics in VIN Synthesizer

Graphics 0 and 1 are allocated to the lower-layer/current graphics in the VIN synthesizer, respectively.

**Table 35.26 Selection of Lower-Layer Plane in Scaler**

Register Name	Bit Name	Initial Value	Description
GR_VIN_AB1	GR_VIN_SCL_ UND_SEL	0	Specifies lower-layer plane in the scaler. 0: Selects graphics 0 as lower-layer graphics and graphics 1 as current graphics. 1: Selects graphics 1 as lower-layer graphics and graphics 0 as current graphics.



## 35.2 Register Descriptions

Table 35.27 to Table 35.34 show the register configuration.

- Symbols used in Register Description:

Initial value: Register value after a reset

—: Undefined value

R/W: Readable/writable. The written value can be read.

R/WC0: Readable/writable. Writing 0 initializes the bit. Writing 1 is ignored.

R/WC1: Readable/writable. Writing 1 initializes the bit. Writing 0 is ignored.

R: Read-only. The write value should always be 0.

—/W: Write-only. The read value is undefined.

Table 35.27 shows the register configuration for the graphics 2 process of channel 0.

Table 35.28 shows the register configuration for the graphics 3 process of channel 0.

Table 35.29 shows the CLUT table configuration of channel 0.

Table 35.30 shows the register configuration for the VIN synthesizer of channel 0.

Table 35.31 shows the register configuration for the graphics 2 process of channel 1.

Table 35.32 shows the register configuration for the graphics 3 process of channel 1.

Table 35.33 shows the CLUT table configuration of channel 1.

Table 35.34 shows the register configuration for the VIN synthesizer of channel 1.

The register configuration for the graphics 0 and 1 processes is described in section 33, Video Display Controller 5 (3): Scaler.

**Table 35.27 Register Configuration of the Image Synthesizer (Graphics 2 Process) (Channel 0)**

Name	Abbreviation	R/W	Initial Value	Address	Access Size
Graphics 2 register update control register	GR2_UPDATE	R/WC1	H'0000 0000	H'FCFF 7700	32
Frame buffer read control register (Graphics 2)	GR2_FLM_RD	R/W	H'0000 0000	H'FCFF 7704	32
Frame buffer control register 1 (Graphics 2)	GR2_FLM1	R/W	H'0000 0000	H'FCFF 7708	32
Frame buffer control register 2 (Graphics 2)	GR2_FLM2	R/W	H'0000 0000	H'FCFF 770C	32
Frame buffer control register 3 (Graphics 2)	GR2_FLM3	R/W	H'0000 0000	H'FCFF 7710	32
Frame buffer control register 4 (Graphics 2)	GR2_FLM4	R/W	H'0000 0000	H'FCFF 7714	32
Frame buffer control register 5 (Graphics 2)	GR2_FLM5	R/W	H'0000 03FF	H'FCFF 7718	32
Frame buffer control register 6 (Graphics 2)	GR2_FLM6	R/W	H'0000 0000	H'FCFF 771C	32
Alpha blending control register 1 (Graphics 2)	GR2_AB1	R/W	H'0000 0000	H'FCFF 7720	32
Alpha blending control register 2 (Graphics 2)	GR2_AB2	R/W	H'0000 0000	H'FCFF 7724	32
Alpha blending control register 3 (Graphics 2)	GR2_AB3	R/W	H'0000 0000	H'FCFF 7728	32
Alpha blending control register 4 (Graphics 2)	GR2_AB4	R/W	H'0000 0000	H'FCFF 772C	32
Alpha blending control register 5 (Graphics 2)	GR2_AB5	R/W	H'0000 0000	H'FCFF 7730	32
Alpha blending control register 6 (Graphics 2)	GR2_AB6	R/W	H'0000 0000	H'FCFF 7734	32
Alpha blending control register 7 (Graphics 2)	GR2_AB7	R/W	H'00FF 0000	H'FCFF 7738	32
Alpha blending control register 8 (Graphics 2)	GR2_AB8	R/W	H'0000 0000	H'FCFF 773C	32
Alpha blending control register 9 (Graphics 2)	GR2_AB9	R/W	H'0000 0000	H'FCFF 7740	32
Alpha blending control register 10 (Graphics 2)	GR2_AB10	R/W	H'0000 0000	H'FCFF 7744	32
Alpha blending control register 11 (Graphics 2)	GR2_AB11	R/W	H'0000 0000	H'FCFF 7748	32
Background color control register (Graphics 2)	GR2_BASE	R/W	H'0000 0000	H'FCFF 774C	32
CLUT table control register (Graphics 2)	GR2_CLUT	R/W	H'0000 0000	H'FCFF 7750	32
Status monitor register (Graphics 2)	GR2_MON	R	H'0000 0000	H'FCFF 7754	32

**Table 35.28 Register Configuration of the Image Synthesizer (Graphics 3 Process) (Channel 0)**

Name	Abbreviation	R/W	Initial Value	Address	Access Size
Graphics 3 register update control register	GR3_UPDATE	R/WC1	H'0000 0000	H'FCFF 7780	32
Frame buffer read control register (Graphics 3)	GR3_FLM_RD	R/W	H'0000 0000	H'FCFF 7784	32
Frame buffer control register 1 (Graphics 3)	GR3_FLM1	R/W	H'0000 0000	H'FCFF 7788	32
Frame buffer control register 2 (Graphics 3)	GR3_FLM2	R/W	H'0000 0000	H'FCFF 778C	32
Frame buffer control register 3 (Graphics 3)	GR3_FLM3	R/W	H'0000 0000	H'FCFF 7790	32
Frame buffer control register 4 (Graphics 3)	GR3_FLM4	R/W	H'0000 0000	H'FCFF 7794	32
Frame buffer control register 5 (Graphics 3)	GR3_FLM5	R/W	H'0000 03FF	H'FCFF 7798	32
Frame buffer control register 6 (Graphics 3)	GR3_FLM6	R/W	H'0000 0000	H'FCFF 779C	32
Alpha blending control register 1 (Graphics 3)	GR3_AB1	R/W	H'0000 0000	H'FCFF 77A0	32
Alpha blending control register 2 (Graphics 3)	GR3_AB2	R/W	H'0000 0000	H'FCFF 77A4	32
Alpha blending control register 3 (Graphics 3)	GR3_AB3	R/W	H'0000 0000	H'FCFF 77A8	32
Alpha blending control register 4 (Graphics 3)	GR3_AB4	R/W	H'0000 0000	H'FCFF 77AC	32
Alpha blending control register 5 (Graphics 3)	GR3_AB5	R/W	H'0000 0000	H'FCFF 77B0	32
Alpha blending control register 6 (Graphics 3)	GR3_AB6	R/W	H'0000 0000	H'FCFF 77B4	32
Alpha blending control register 7 (Graphics 3)	GR3_AB7	R/W	H'00FF 0000	H'FCFF 77B8	32
Alpha blending control register 8 (Graphics 3)	GR3_AB8	R/W	H'0000 0000	H'FCFF 77BC	32
Alpha blending control register 9 (Graphics 3)	GR3_AB9	R/W	H'0000 0000	H'FCFF 77C0	32
Alpha blending control register 10 (Graphics 3)	GR3_AB10	R/W	H'0000 0000	H'FCFF 77C4	32
Alpha blending control register 11 (Graphics 3)	GR3_AB11	R/W	H'0000 0000	H'FCFF 77C8	32
Background color control register (Graphics 3)	GR3_BASE	R/W	H'0000 0000	H'FCFF 77CC	32
CLUT table and interrupt control register (Graphics 3)	GR3_CLUT_INT	R/W	H'0000 0000	H'FCFF 77D0	32
Status monitor register (Graphics 3)	GR3_MON	R	H'0000 0000	H'FCFF 77D4	32

**Table 35.29 CLUT Table Configuration (Channel 0)**

Name	Abbreviation	R/W	Initial Value	Address	Access Size
Graphics 0 CLUT table	GR0_CLUT	R/W	—	H'FCFF 6000 to H'FCFF 63FF	32
Graphics 1 CLUT table	GR1_CLUT	R/W	—	H'FCFF 6400 to H'FCFF 67FF	32
Graphics 2 CLUT table	GR2_CLUT	R/W	—	H'FCFF 6800 to H'FCFF 6BFF	32
Graphics 3 CLUT table	GR3_CLUT	R/W	—	H'FCFF 6C00 to H'FCFF 6FFF	32
Graphics OIR CLUT table	GR_OIR_CLUT	R/W	—	H'FCFF 7000 to H'FCFF 73FF	32

**Table 35.30 Register Configuration of the VIN Synthesizer (Channel 0)**

Name	Abbreviation	R/W	Initial Value	Address	Access Size
VIN synthesizer register update control register	GR_VIN_UPDATE	R/WC1	H'0000 0000	H'FCFF 7E00	32
Alpha blending control register 1 (VIN synthesizer)	GR_VIN_AB1	R/W	H'0000 0000	H'FCFF 7E20	32
Alpha blending control register 2 (VIN synthesizer)	GR_VIN_AB2	R/W	H'0000 0000	H'FCFF 7E24	32
Alpha blending control register 3 (VIN synthesizer)	GR_VIN_AB3	R/W	H'0000 0000	H'FCFF 7E28	32
Alpha blending control register 4 (VIN synthesizer)	GR_VIN_AB4	R/W	H'0000 0000	H'FCFF 7E2C	32
Alpha blending control register 5 (VIN synthesizer)	GR_VIN_AB5	R/W	H'0000 0000	H'FCFF 7E30	32
Alpha blending control register 6 (VIN synthesizer)	GR_VIN_AB6	R/W	H'0000 0000	H'FCFF 7E34	32
Alpha blending control register 7 (VIN synthesizer)	GR_VIN_AB7	R/W	H'00FF 0000	H'FCFF 7E38	32
Background color control register (VIN synthesizer)	GR_VIN_BASE	R/W	H'0000 0000	H'FCFF 7E4C	32
Status monitor register (VIN synthesizer)	GR_VIN_MON	R	H'0000 0000	H'FCFF 7E54	32

**Table 35.31 Register Configuration of the Image Synthesizer (Graphics 2 Process) (Channel 1)**

Name	Abbreviation	R/W	Initial Value	Address	Access Size
Graphics 2 register update control register	GR2_UPDATE	R/WC1	H'0000 0000	H'FCFF 9700	32
Frame buffer read control register (Graphics 2)	GR2_FLM_RD	R/W	H'0000 0000	H'FCFF 9704	32
Frame buffer control register 1 (Graphics 2)	GR2_FLM1	R/W	H'0000 0000	H'FCFF 9708	32
Frame buffer control register 2 (Graphics 2)	GR2_FLM2	R/W	H'0000 0000	H'FCFF 970C	32
Frame buffer control register 3 (Graphics 2)	GR2_FLM3	R/W	H'0000 0000	H'FCFF 9710	32
Frame buffer control register 4 (Graphics 2)	GR2_FLM4	R/W	H'0000 0000	H'FCFF 9714	32
Frame buffer control register 5 (Graphics 2)	GR2_FLM5	R/W	H'0000 03FF	H'FCFF 9718	32
Frame buffer control register 6 (Graphics 2)	GR2_FLM6	R/W	H'0000 0000	H'FCFF 971C	32
Alpha blending control register 1 (Graphics 2)	GR2_AB1	R/W	H'0000 0000	H'FCFF 9720	32
Alpha blending control register 2 (Graphics 2)	GR2_AB2	R/W	H'0000 0000	H'FCFF 9724	32
Alpha blending control register 3 (Graphics 2)	GR2_AB3	R/W	H'0000 0000	H'FCFF 9728	32
Alpha blending control register 4 (Graphics 2)	GR2_AB4	R/W	H'0000 0000	H'FCFF 972C	32
Alpha blending control register 5 (Graphics 2)	GR2_AB5	R/W	H'0000 0000	H'FCFF 9730	32
Alpha blending control register 6 (Graphics 2)	GR2_AB6	R/W	H'0000 0000	H'FCFF 9734	32
Alpha blending control register 7 (Graphics 2)	GR2_AB7	R/W	H'00FF 0000	H'FCFF 9738	32
Alpha blending control register 8 (Graphics 2)	GR2_AB8	R/W	H'0000 0000	H'FCFF 973C	32
Alpha blending control register 9 (Graphics 2)	GR2_AB9	R/W	H'0000 0000	H'FCFF 9740	32
Alpha blending control register 10 (Graphics 2)	GR2_AB10	R/W	H'0000 0000	H'FCFF9744	32
Alpha blending control register 11 (Graphics 2)	GR2_AB11	R/W	H'0000 0000	H'FCFF9748	32
Background color control register (Graphics 2)	GR2_BASE	R/W	H'0000 0000	H'FCFF974C	32
CLUT table control register (Graphics 2)	GR2_CLUT	R/W	H'0000 0000	H'FCFF9750	32
Status monitor register (Graphics 2)	GR2_MON	R	H'0000 0000	H'FCFF9754	32

**Table 35.32 Register Configuration of the Image Synthesizer (Graphics 3 Process) (Channel 1)**

Name	Abbreviation	R/W	Initial Value	Address	Access Size
Graphics 3 register update control register	GR3_UPDATE	R/WC1	H'0000 0000	H'FCFF 9780	32
Frame buffer read control register (Graphics 3)	GR3_FLM_RD	R/W	H'0000 0000	H'FCFF 9784	32
Frame buffer control register 1 (Graphics 3)	GR3_FLM1	R/W	H'0000 0000	H'FCFF 9788	32
Frame buffer control register 2 (Graphics 3)	GR3_FLM2	R/W	H'0000 0000	H'FCFF 978C	32
Frame buffer control register 3 (Graphics 3)	GR3_FLM3	R/W	H'0000 0000	H'FCFF 9790	32
Frame buffer control register 4 (Graphics 3)	GR3_FLM4	R/W	H'0000 0000	H'FCFF 9794	32
Frame buffer control register 5 (Graphics 3)	GR3_FLM5	R/W	H'0000 03FF	H'FCFF 9798	32
Frame buffer control register 6 (Graphics 3)	GR3_FLM6	R/W	H'0000 0000	H'FCFF 979C	32
Alpha blending control register 1 (Graphics 3)	GR3_AB1	R/W	H'0000 0000	H'FCFF 97A0	32
Alpha blending control register 2 (Graphics 3)	GR3_AB2	R/W	H'0000 0000	H'FCFF 97A4	32
Alpha blending control register 3 (Graphics 3)	GR3_AB3	R/W	H'0000 0000	H'FCFF 97A8	32
Alpha blending control register 4 (Graphics 3)	GR3_AB4	R/W	H'0000 0000	H'FCFF 97AC	32
Alpha blending control register 5 (Graphics 3)	GR3_AB5	R/W	H'0000 0000	H'FCFF 97B0	32
Alpha blending control register 6 (Graphics 3)	GR3_AB6	R/W	H'0000 0000	H'FCFF 97B4	32
Alpha blending control register 7 (Graphics 3)	GR3_AB7	R/W	H'00FF 0000	H'FCFF 97B8	32
Alpha blending control register 8 (Graphics 3)	GR3_AB8	R/W	H'0000 0000	H'FCFF 97BC	32
Alpha blending control register 9 (Graphics 3)	GR3_AB9	R/W	H'0000 0000	H'FCFF 97C0	32
Alpha blending control register 10 (Graphics 3)	GR3_AB10	R/W	H'0000 0000	H'FCFF 97C4	32
Alpha blending control register 11 (Graphics 3)	GR3_AB11	R/W	H'0000 0000	H'FCFF 97C8	32
Background color control register (Graphics 3)	GR3_BASE	R/W	H'0000 0000	H'FCFF 97CC	32
CLUT table and interrupt control register (Graphics 3)	GR3_CLUT_INT	R/W	H'0000 0000	H'FCFF 97D0	32
Status monitor register (Graphics 3)	GR3_MON	R	H'0000 0000	H'FCFF 97D4	32

**Table 35.33 CLUT Table Configuration (Channel 1)**

Name	Abbreviation	R/W	Initial Value	Address	Access Size
Graphics 0 CLUT table	GR0_CLUT	R/W	—	H'FCFF 8000 to H'FCFF 83FF	32
Graphics 1 CLUT table	GR1_CLUT	R/W	—	H'FCFF 8400 to H'FCFF 87FF	32
Graphics 2 CLUT table	GR2_CLUT	R/W	—	H'FCFF 8800 to H'FCFF 8BFF	32
Graphics 3 CLUT table	GR3_CLUT	R/W	—	H'FCFF 8C00 to H'FCFF 8FFF	32
Graphics OIR CLUT table	GR_OIR_CLUT	R/W	—	H'FCFF 9000 to H'FCFF 93FF	32

**Table 35.34 Register Configuration of the VIN Synthesizer (Channel 1)**

Name	Abbreviation	R/W	Initial Value	Address	Access Size
VIN synthesizer register update control register	GR_VIN_UPDATE	R/WC1	H'0000 0000	H'FCFF 9E00	32
Alpha blending control register 1 (VIN synthesizer)	GR_VIN_AB1	R/W	H'0000 0000	H'FCFF 9E20	32
Alpha blending control register 2 (VIN synthesizer)	GR_VIN_AB2	R/W	H'0000 0000	H'FCFF 9E24	32
Alpha blending control register 3 (VIN synthesizer)	GR_VIN_AB3	R/W	H'0000 0000	H'FCFF 9E28	32
Alpha blending control register 4 (VIN synthesizer)	GR_VIN_AB4	R/W	H'0000 0000	H'FCFF 9E2C	32
Alpha blending control register 5 (VIN synthesizer)	GR_VIN_AB5	R/W	H'0000 0000	H'FCFF 9E30	32
Alpha blending control register 6 (VIN synthesizer)	GR_VIN_AB6	R/W	H'0000 0000	H'FCFF 9E34	32
Alpha blending control register 7 (VIN synthesizer)	GR_VIN_AB7	R/W	H'00FF 0000	H'FCFF 9E38	32
Background color control register (VIN synthesizer)	GR_VIN_BASE	R/W	H'0000 0000	H'FCFF 9E4C	32
Status monitor register (VIN synthesizer)	GR_VIN_MON	R	H'0000 0000	H'FCFF 9E54	32

## 35.2.1 Graphics 2 Register Update Control Register (GR2\_UPDATE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	GR2_UPDATE	—	—	—	GR2_P_VEN	—	—	—	GR2_IBUS_VEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/WC1	R	R	R	R/WC1	R	R	R	R/WC1

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	GR2_UPDATE	0	R/WC1	Frame Buffer Read Control Register Update 0: Registers are not updated. 1: Registers are updated.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	GR2_P_VEN	0	R/WC1	Graphics Display Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR2_IBUS_VEN	0	R/WC1	Frame Buffer Read Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.

## 35.2.2 Frame Buffer Read Control Register (Graphics 2) (GR2\_FLM\_RD)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR2_R_ENB
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR2_R_ENB	0	R/W	Frame Buffer Read Enable 0: Frame buffer reading is disabled. 1: Frame buffer reading is enabled.

Note: This register is updated when GR2\_IBUS\_VEN in GR2\_UPDATE is 1.



## 35.2.3 Frame Buffer Control Register 1 (Graphics 2) (GR2\_FLM1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR2_LN_OFF_DIR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	GR2_FLM_SEL[1:0]		—	—	—	—	—	—	—	GR2_BST_MD
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	GR2_LN_OFF_DIR	0	R/W	Selects the line offset address direction of the frame buffer. 0: Increments the address by the line offset address. 1: Decrements the address by the line offset address.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	GR2_FLM_SEL [1:0]	0	R/W	Selects a frame buffer address setting signal. 0: Selects frame 0. 1: Selects register GR2_FLM_NUM. 2: Selects frame 0. 3: Setting prohibited
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR2_BST_MD	0	R/W	Frame Buffer Burst Transfer Mode 0: 32-byte transfer 1: 128-byte transfer

Note: GR2\_LN\_OFF\_DIR and GR2\_FLM\_SEL are updated when GR2\_IBUS\_VEN in GR2\_UPDATE is 1.  
GR2\_BST\_MD is updated when GR2\_IBUS\_VEN and GR2\_P\_VEN in GR2\_UPDATE are 1.

## 35.2.4 Frame Buffer Control Register 2 (Graphics 2) (GR2\_FLM2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR2_BASE[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR2_BASE[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	GR2_BASE [31:0]	0	R/W	Frame Buffer Base Address (upper) Sets the start address of the frame buffer where frame data is to be stored. GR_BASE[4:3] and GR_BASE[6:3] are referred to during 32-byte burst transfer and 128-byte burst transfer, respectively, to skip the start line data. The lower 3 bits should be fixed to 000.

Note: This register is updated when GR2\_IBUS\_VEN and GR2\_P\_VEN in GR2\_UPDATE are 1.

## 35.2.5 Frame Buffer Control Register 3 (Graphics 2) (GR2\_FLM3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	GR2_LN_OFF[14:0]														
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	GR2_FLM_NUM[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30 to 16	GR2_LN_OFF[14:0]	0	R/W	Frame Buffer Line Offset Address Sets the line offset address for calculating the start address of each line. Line 0: GR2_BASE Line 1: GR2_BASE + GR2_LN_OFF × 1 : Line n: GR2_BASE + GR2_LN_OFF × n For 32 byte transfer, the lower 5 bits should be fixed to 0_0000. For 128 byte transfer, the lower 7 bits should be fixed to 000_0000.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	GR2_FLM_NUM[9:0]	0	R/W	Frame Number of Frame Buffer Manually set the frame number when GR2_FLM_SEL = 1.

Note: This register is updated when GR2\_IBUS\_VEN in GR2\_UPDATE is 1.

## 35.2.6 Frame Buffer Control Register 4 (Graphics 2) (GR2\_FLM4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	GR2_FLM_OFF[22:16]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR2_FLM_OFF[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
22 to 0	GR2_FLM_OFF[22:0]	0	R/W	Frame Buffer Frame Offset Address (upper) Specifies the frame offset address used for calculating the start address of each frame buffer when more than one buffer is used. Buffer 0: GR2_BASE Buffer 1: GR2_BASE + GR2_FLM_OFF × 1 : Buffer n: GR2_BASE + GR2_FLM_OFF × n For 32 byte transfer, the lower 5 bits should be fixed to 0_0000. For 128 byte transfer, the lower 7 bits should be fixed to 000_0000.

Note: This register is updated when GR2\_IBUS\_VEN in GR2\_UPDATE is 1.

## 35.2.7 Frame Buffer Control Register 5 (Graphics 2) (GR2\_FLM5)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR2_FLM_LNUM[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR2_FLM_LOOP[10:0]										
Initial value:	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR2_FLM_LNUM[10:0]	0	R/W	Sets number of lines in a frame The number of lines is (GR2_FLM_LNUM + 1).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR2_FLM_LOOP[10:0]	1023	R/W	Number of lines when reading the addresses repeatedly by returning to the start address after reaching the end address. The number of lines is (GR2_FLM_LOOP + 1).

Note: This register is updated when GR2\_IBUS\_VEN in GR2\_UPDATE is 1.

## 35.2.8 Frame Buffer Control Register 6 (Graphics 2) (GR2\_FLM6)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR2_FORMAT[3:0]				—	GR2_HW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	GR2_RDSWA[2:0]		—	—	—	—	GR2_STA_POS[5:0]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	GR2_FORMAT [3:0]	0	R/W	Sets the format of the frame buffer read signal. 0: RGB565 1: RGB888 2: $\alpha$ RGB1555 3: $\alpha$ RGB4444 4: $\alpha$ RGB8888 5: CLUT8 6: CLUT4 7: CLUT1 8: Setting prohibited 9: Setting prohibited 10: RGB $\alpha$ 5551 11: RGB $\alpha$ 8888 12 to 15: Setting prohibited
27	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
26 to 16	GR2_HW [10:0]	0	R/W	Sets the width of the horizontal valid period. The width is (GR2_HW + 1) pixels. Note: Set to 2 or greater.
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 10	GR2_RDSWA [2:0]	0	R/W	Sets 8-, 16-, and 32-bit swap. These three bits specify the method for swapping the bits of frame buffer read data as follows. Bit 0 0: 8 bits are not swapped. 1: 8 bits are swapped. Bit 1 0: 16 bits are not swapped. 1: 16 bits are swapped. Bit 2 0: 32 bits are not swapped. 1: 32 bits are swapped. When eight bits are put together, they are swapped as follows. Each of (1) to (8) indicates eight-bit data. 000: (1) (2) (3) (4) (5) (6) (7) (8) [No swap] 001: (2) (1) (4) (3) (6) (5) (8) (7) [8-bit swap] 010: (3) (4) (1) (2) (7) (8) (5) (6) [16-bit swap] 011: (4) (3) (2) (1) (8) (7) (6) (5) [16-bit swap + 8-bit swap] 100: (5) (6) (7) (8) (1) (2) (3) (4) [32-bit swap] 101: (6) (5) (8) (7) (2) (1) (4) (3) [32-bit swap + 8-bit swap] 110: (7) (8) (5) (6) (3) (4) (1) (2) [32-bit swap + 16-bit swap] 111: (8) (7) (6) (5) (4) (3) (2) (1) [32-bit swap + 16-bit swap + 8-bit swap]
9 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
5 to 0	GR2_STA_POS [5:0]	0	R/W	Sets the amount of data to be skipped through. Specifically data amount equal to the amount indicated by GR2_STA_POS is skipped from the start of the line.

Note: GR2\_STA\_POS is updated when GR2\_P\_VEN in GR2\_UPDATE is 1. GR2\_RDSWA is updated when GR2\_UPDATE in GR2\_UPDATE is 1. GR2\_FORMAT and GR2\_HW are updated when GR2\_IBUS\_VEN and GR2\_P\_VEN in GR2\_UPDATE are 1.

### 35.2.9 Alpha Blending Control Register 1 (Graphics 2) (GR2\_AB1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR2_ARC_MUL	GR2_ACALC_MD	—	GR2_ARC_ON	—	—	—	GR2_ARC_DISP_ON	—	—	—	GR2_GRC_DISP_ON	—	—	GR2_DISP_SEL[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15	GR2_ARC_MUL	0	R/W	Turns on/off multiplication processing with current alpha at alpha blending in a rectangular area. 0: Off 1: On
14	GR2_ACALC_MD	0	R/W	Turns on/off premultiplication processing at alpha blending in one-pixel units. 0: Off 1: On
13	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
12	GR2_ARC_ON	0	R/W	Turns on/off alpha blending in a rectangular area. 0: Off 1: On
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	GR2_ARC_DISP_ON	0	R/W	Turns on/off frame-line display of the image area for alpha blending in a rectangular area. 0: Frame-line display off 1: Frame-line display on
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	GR2_GRC_DISP_ON	0	R/W	Turns on/off frame-line display of the graphics image area. 0: Frame-line display off 1: Frame-line display on
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1, 0	GR2_DISP_SEL[1:0]	0	R/W	Selects the graphics display mode. 0: Background color display 1: Lower-layer graphics display 2: Current graphics display 3: Blended display of lower-layer graphics and current graphics

Note: This register is updated when GR2\_P\_VEN in GR2\_UPDATE is 1.

### 35.2.10 Alpha Blending Control Register 2 (Graphics 2) (GR2\_AB2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR2_GRC_VS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR2_GRC_VW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR2_GRC_VS[10:0]	0	R/W	Sets the vertical start position of the graphics image area. Note: Set to 4 or greater lines and the result of GR2_GRC_VS + GR2_GRC_VW should be smaller than or equal to 2039 lines.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR2_GRC_VW[10:0]	0	R/W	Sets the vertical width of the graphics image area.

Note: This register is updated when GR2\_P\_VEN in GR2\_UPDATE is 1.

## 35.2.11 Alpha Blending Control Register 3 (Graphics 2) (GR2\_AB3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR2_GRC_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR2_GRC_HW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR2_GRC_HS[10:0]	0	R/W	Sets the horizontal start position of the graphics image area. Note: Set to 16 or greater clocks and the result of GR2_GRC_HS + GR2_GRC_HW should be smaller than or equal to 2015 clocks.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR2_GRC_HW[10:0]	0	R/W	Sets the horizontal width of the graphics image area. Note: For displaying an image with 1- or 2-pixel horizontal width, set GR2_HW to 2 and GR2_GRC_HW to 1 (1-pixel) or 2 (2-pixel).

Note: This register is updated when GR2\_P\_VEN in GR2\_UPDATE is 1.

## 35.2.12 Alpha Blending Control Register 4 (Graphics 2) (GR2\_AB4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR2_ARC_VS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR2_ARC_VW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR2_ARC_VS[10:0]	0	R/W	Sets the vertical start position of the valid image area for alpha blending in a rectangular area.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR2_ARC_VW[10:0]	0	R/W	Sets the vertical width of the valid image area for alpha blending in a rectangular area.

Note: This register is updated when GR2\_P\_VEN in GR2\_UPDATE is 1.

## 35.2.13 Alpha Blending Control Register 5 (Graphics 2) (GR2\_AB5)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR2_ARC_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR2_ARC_HW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR2_ARC_HS[10:0]	0	R/W	Sets the horizontal start position of the valid image area for alpha blending in a rectangular area.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR2_ARC_HW[10:0]	0	R/W	Sets the horizontal width of the valid image area for alpha blending in a rectangular area.

Note: This register is updated when GR2\_P\_VEN in GR2\_UPDATE is 1.

## 35.2.14 Alpha Blending Control Register 6 (Graphics 2) (GR2\_AB6)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	GR2_ARC_MODE	GR2_ARC_COEF[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	GR2_ARC_RATE[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	GR2_ARC_MODE	0	R/W	Alpha Blending Mode in Rectangular Area 0: Addition 1: Subtraction
23 to 16	GR2_ARC_COEF[7:0]	0	R/W	Sets the alpha coefficient for alpha blending in a rectangular area. (0 to 255) [7:0]: Variation (absolute value)
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	GR2_ARC_RATE[7:0]	0	R/W	Sets the value obtained by subtracting 1 from the frame rate for alpha blending in a rectangular area.

Note: This bit is updated when GR2\_P\_VEN in GR2\_UPDATE is 1.



## 35.2.15 Alpha Blending Control Register 7 (Graphics 2) (GR2\_AB7)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	GR2_ARC_DEF[7:0]							
Initial value:	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR2_CK_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	GR2_ARC_DEF[7:0]	255	R/W	Sets the initial alpha value for alpha blending in a rectangular area.
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR2_CK_ON	0	R/W	Turns on/off CLUT-index/RGB-index chroma-key processing. 0: Off 1: On

Note: This register is updated when GR2\_P\_VEN in GR2\_UPDATE is 1.

## 35.2.16 Alpha Blending Control Register 8 (Graphics 2) (GR2\_AB8)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR2_CK_KCLUT[7:0]								GR2_CK_KG[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR2_CK_KB[7:0]								GR2_CK_KR[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GR2_CK_KCLUT[7:0]	0	R/W	CLUT Signal for CLUT-Index Chroma-Key Processing CLUT: Unsigned 8 bits (0 to 255 [LSB])
23 to 16	GR2_CK_KG[7:0]	0	R/W	G Signal for RGB-Index Chroma-Key Processing G: Unsigned 8 bits (0 to 255 [LSB])
15 to 8	GR2_CK_KB[7:0]	0	R/W	B Signal for RGB-Index Chroma-Key Processing B: Unsigned 8 bits (0 to 255 [LSB])
7 to 0	GR2_CK_KR[7:0]	0	R/W	R Signal for RGB-Index Chroma-Key Processing R: Unsigned 8 bits (0 to 255 [LSB])

Note: This register is updated when GR2\_P\_VEN in GR2\_UPDATE is 1.

## 35.2.17 Alpha Blending Control Register 9 (Graphics 2) (GR2\_AB9)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR2_CK_A[7:0]								GR2_CK_G[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR2_CK_B[7:0]								GR2_CK_R[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GR2_CK_A [7:0]	0	R/W	Replaced Alpha Signal after RGB/CLUT-Index Chroma-Key Processing $\alpha$ : Unsigned 8 bits (0 to 255 [LSB])
23 to 16	GR2_CK_G [7:0]	0	R/W	Replaced G Signal after RGB/CLUT-Index Chroma-Key Processing G: Unsigned 8 bits (0 to 255 [LSB])
15 to 8	GR2_CK_B [7:0]	0	R/W	Replaced B Signal after RGB/CLUT-Index Chroma-Key Processing B: Unsigned 8 bits (0 to 255 [LSB])
7 to 0	GR2_CK_R [7:0]	0	R/W	Replaced R Signal after RGB/CLUT-Index Chroma-Key Processing R: Unsigned 8 bits (0 to 255 [LSB])

Note: This register is updated when GR2\_P\_VEN in GR2\_UPDATE is 1.

## 35.2.18 Alpha Blending Control Register 10 (Graphics 2) (GR2\_AB10)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR2_A0[7:0]								GR2_G0[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR2_B0[7:0]								GR2_R0[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GR2_A0 [7:0]	0	R/W	CLUT1 $\alpha$ 0 Signal Replaced with $\alpha$ signal when in the CLUT1 format and CLUT1 = 0. Replaced with $\alpha$ signal when in the $\alpha$ RGB1555 or RGB $\alpha$ 5551 format and $\alpha$ = 0.
23 to 16	GR2_G0 [7:0]	0	R/W	CLUT1 G0 Signal Replaced with G signal when in the CLUT1 format and CLUT1 = 0.
15 to 8	GR2_B0 [7:0]	0	R/W	CLUT1 B0 Signal Replaced with B signal when in the CLUT1 format and CLUT1 = 0.
7 to 0	GR2_R0 [7:0]	0	R/W	CLUT1 R0 Signal Replaced with R signal when in the CLUT1 format and CLUT1 = 0.

Note: This register is updated when GR2\_P\_VEN in GR2\_UPDATE is 1.

## 35.2.19 Alpha Blending Control Register 11 (Graphics 2) (GR2\_AB11)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR2_A1[7:0]								GR2_G1[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR2_B1[7:0]								GR2_R1[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GR2_A1 [7:0]	0	R/W	CLUT1 $\alpha$ 1 Signal Replaced with $\alpha$ signal when in the CLUT1 format and CLUT1 = 1. Replaced with $\alpha$ signal when in the $\alpha$ RGB1555 or RGB $\alpha$ 5551 format and $\alpha$ = 1.
23 to 16	GR2_G1 [7:0]	0	R/W	CLUT1 G1 Signal Replaced with G signal when in the CLUT1 format and CLUT1 = 1.
15 to 8	GR2_B1 [7:0]	0	R/W	CLUT1 B1 Signal Replaced with B signal when in the CLUT1 format and CLUT1 = 1.
7 to 0	GR2_R1 [7:0]	0	R/W	CLUT1 R1 Signal Replaced with R signal when in the CLUT1 format and CLUT1 = 1.

Note: This register is updated when GR2\_P\_VEN in GR2\_UPDATE is 1.

## 35.2.20 Background Color Control Register (Graphics 2) (GR2\_BASE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	GR2_BASE_G[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR2_BASE_B[7:0]								GR2_BASE_R[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	GR2_BASE_G [7:0]	0	R/W	Background Color G Signal G: Unsigned 8 bits (0 to 255 [LSB])
15 to 8	GR2_BASE_B [7:0]	0	R/W	Background Color B Signal B: Unsigned 8 bits (0 to 255 [LSB])
7 to 0	GR2_BASE_R [7:0]	0	R/W	Background Color R Signal R: Unsigned 8 bits (0 to 255 [LSB])

Note: This register is updated when GR2\_P\_VEN in GR2\_UPDATE is 1.

## 35.2.21 CLUT Table Control Register (Graphics 2) (GR2\_CLUT)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR2_CLT_SEL
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	GR2_CLT_SEL	0	R/W	CLUT Table Select Signal 0: Selects CLUT table 0. The format is converted to $\alpha$ RGB8888 based on the CLUT table 0. CLUT table 1 can be read from or written to by the CPU. 1: Selects CLUT table 1. The format is converted to $\alpha$ RGB8888 based on the CLUT table 1. CLUT table 0 can be read from or written to by the CPU.
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note: This register is updated when GR2\_P\_VEN in GR2\_UPDATE is 1.

## 35.2.22 Status Monitor Register (Graphics 2) (GR2\_MON)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR2_ARC_ST
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR2_ARC_ST	0	R	Status Flag for Alpha Blending in Rectangular Area 0: Addition or subtraction has been completed. ( $\alpha$ value is 0 or 255) 1: Addition or subtraction is in progress.

## 35.2.23 Graphics 3 Register Update Control Register (GR3\_UPDATE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	GR3_UPDATE	—	—	—	GR3_P_VEN	—	—	—	GR3_IBUS_VEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/WC1	R	R	R	R/WC1	R	R	R	R/WC1

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	GR3_UPDATE	0	R/WC1	Frame Buffer Read Control Register Update 0: Registers are not updated. 1: Registers are updated.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	GR3_P_VEN	0	R/WC1	Graphics Display Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR3_IBUS_VEN	0	R/WC1	Frame Buffer Read Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.

## 35.2.24 Frame Buffer Read Control Register (Graphics 3) (GR3\_FLM\_RD)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR3_R_ENB
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR3_R_ENB	0	R/W	Frame Buffer Read Enable 0: Frame buffer reading is disabled. 1: Frame buffer reading is enabled.

Note: This register is updated when GR3\_IBUS\_VEN in GR3\_UPDATE is 1.

## 35.2.25 Frame Buffer Control Register 1 (Graphics 3) (GR3\_FLM1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR3_LN_OFF_DIR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	GR3_FLM_SEL[1:0]		—	—	—	—	—	—	—	GR3_BST_MD
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	GR3_LN_OFF_DIR	0	R/W	Selects the line offset address direction of the frame buffer. 0: Increments the address by the line offset address. 1: Decrements the address by the line offset address.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	GR3_FLM_SEL[1:0]	0	R/W	Selects a frame buffer address setting signal. 0: Selects frame 0. 1: Selects register GR3_FLM_NUM. 2: Selects frame 0. 3: Setting prohibited
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR3_BST_MD	0	R/W	Frame Buffer Burst Transfer Mode 0: 32-byte transfer 1: 128- byte transfer

Note: GR3\_LN\_OFF\_DIR and GR3\_FLM\_SEL are updated when GR3\_IBUS\_VEN in GR3\_UPDATE is 1.  
GR3\_BST\_MD is updated when GR3\_IBUS\_VEN and GR3\_P\_VEN in GR3\_UPDATE are 1.

## 35.2.26 Frame Buffer Control Register 2 (Graphics 3) (GR3\_FLM2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR3_BASE[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR3_BASE[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	GR3_BASE [31:0]	0	R/W	Frame Buffer Base Address (upper) Sets the start address of the frame buffer where frame data is to be stored. GR_BASE[4:3] and GR_BASE[6:3] are referred to during 32-byte burst transfer and 128-byte burst transfer, respectively, to skip the start line data. The lower 3 bits should be fixed to 000.

Note: This register is updated when GR3\_IBUS\_VEN and GR3\_P\_VEN in GR3\_UPDATE are 1.

## 35.2.27 Frame Buffer Control Register 3 (Graphics 3) (GR3\_FLM3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	GR3_LN_OFF[14:0]														
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	GR3_FLM_NUM[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30 to 16	GR3_LN_ OFF[14:0]	0	R/W	Frame Buffer Line Offset Address Sets the line offset address for calculating the start address of each line. Line 0: GR3_BASE Line 1: GR3_BASE + GR3_LN_OFF × 1 : Line n: GR3_BASE + GR3_LN_OFF × n For 32 byte transfer, the lower 5 bits should be fixed to 0_0000. For 128 byte transfer, the lower 7 bits should be fixed to 000_0000.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	GR3_FLM_ NUM[9:0]	0	R/W	Frame Number of Frame Buffer Manually set the frame number when GR3_FLM_SEL = 1.

Note: This register is updated when GR3\_IBUS\_VEN in GR3\_UPDATE is 1.

## 35.2.28 Frame Buffer Control Register 4 (Graphics 3) (GR3\_FLM4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	GR3_FLM_OFF[22:16]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR3_FLM_OFF[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
22 to 0	GR3_FLM_OFF[22:0]	0	R/W	Frame Buffer Frame Offset Address (upper) Specifies the frame offset address used for calculating the start address of each frame buffer when more than one buffer is used. Buffer 0: GR3_BASE Buffer 1: GR3_BASE + GR3_FLM_OFF × 1 : Buffer n: GR3_BASE + GR3_FLM_OFF × n For 32 byte transfer, the lower 5 bits should be fixed to 0_0000. For 128 byte transfer, the lower 7 bits should be fixed to 000_0000.

Note: This register is updated when GR3\_IBUS\_VEN in GR3\_UPDATE is 1.

## 35.2.29 Frame Buffer Control Register 5 (Graphics 3) (GR3\_FLM5)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR3_FLM_LNUM[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR3_FLM_LOOP[10:0]										
Initial value:	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR3_FLM_LNUM[10:0]	0	R/W	Sets number of lines in a frame The number of lines is (GR3_FLM_LNUM + 1).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR3_FLM_LOOP[10:0]	1023	R/W	Number of lines when reading the addresses repeatedly by returning to the start address after reaching the end address. The number of lines is (GR3_FLM_LOOP + 1).

Note: This register is updated when GR3\_IBUS\_VEN in GR3\_UPDATE is 1.



## 35.2.30 Frame Buffer Control Register 6 (Graphics 3) (GR3\_FLM6)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR3_FORMAT[3:0]				—	GR3_HW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	GR3_RDSWA[2:0]		—	—	—	—	GR3_STA_POS[5:0]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	GR3_FORMAT[3:0]	0	R/W	Sets the format of the frame buffer read signal. 0: RGB565 1: RGB888 2: $\alpha$ RGB1555 3: $\alpha$ RGB4444 4: $\alpha$ RGB8888 5: CLUT8 6: CLUT4 7: CLUT1 8: Setting prohibited 9: Setting prohibited 10: RGB $\alpha$ 5551 11: RGB $\alpha$ 8888 12 to 15: Setting prohibited
27	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
26 to 16	GR3_HW[10:0]	0	R/W	Sets the width of the horizontal valid period. The width is (GR3_HW + 1) pixels. Note: Set to 2 or greater.
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 10	GR3_RDSWA[2:0]	0	R/W	Sets 8-, 16-, and 32-bit swap. These three bits specify the method for swapping the bits of frame buffer read data as follows. Bit 0 0: 8 bits are not swapped. 1: 8 bits are swapped. Bit 1 0: 16 bits are not swapped. 1: 16 bits are swapped. Bit 2 0: 32 bits are not swapped. 1: 32 bits are swapped. When eight bits are put together, they are swapped as follows. Each of (1) to (8) indicates eight-bit data. 000: (1) (2) (3) (4) (5) (6) (7) (8) [No swap] 001: (2) (1) (4) (3) (6) (5) (8) (7) [8-bit swap] 010: (3) (4) (1) (2) (7) (8) (5) (6) [16-bit swap] 011: (4) (3) (2) (1) (8) (7) (6) (5) [16-bit swap + 8-bit swap] 100: (5) (6) (7) (8) (1) (2) (3) (4) [32-bit swap] 101: (6) (5) (8) (7) (2) (1) (4) (3) [32-bit swap + 8-bit swap] 110: (7) (8) (5) (6) (3) (4) (1) (2) [32-bit swap + 16-bit swap] 111: (8) (7) (6) (5) (4) (3) (2) (1) [32-bit swap + 16-bit swap + 8-bit swap]
9 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
5 to 0	GR3_STA_POS[5:0]	0	R/W	Sets the amount of data to be skipped through. Specifically data amount equal to the amount indicated by GR3_STA_POS is skipped from the start of the line.

Note: GR3\_STA\_POS is updated when GR3\_P\_VEN in GR3\_UPDATE is 1. GR3\_RDSWA is updated when GR3\_UPDATE in GR3\_UPDATE is 1. GR3\_FORMAT and GR3\_HW are updated when GR3\_IBUS\_VEN and GR3\_P\_VEN in GR3\_UPDATE are 1.

### 35.2.31 Alpha Blending Control Register 1 (Graphics 3) (GR3\_AB1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15	GR3_ARC_MUL	0	R/W	Turns on/off multiplication processing with current alpha at alpha blending in a rectangular area. 0: Off 1: On
14	GR3_ACALC_MD	0	R/W	Turns on/off premultiplication processing at alpha blending in one-pixel units. 0: Off 1: On
13	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
12	GR3_ARC_ON	0	R/W	Turns on/off alpha blending in a rectangular area. 0: Off 1: On
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	GR3_ARC_DISP_ON	0	R/W	Turns on/off frame-line display of the image area for alpha blending in a rectangular area. 0: Frame-line display off 1: Frame-line display on
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	GR3_GRC_DISP_ON	0	R/W	Turns on/off frame-line display of the graphics image area. 0: Frame-line display off 1: Frame-line display on
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	GR3_DISP_SEL[1:0]	0	R/W	Selects the graphics display mode. 0: Background color display 1: Lower-layer graphics display 2: Current graphics display 3: Blended display of lower-layer graphics and current graphics

Note: This register is updated when GR3\_P\_VEN in GR3\_UPDATE is 1.

## 35.2.32 Alpha Blending Control Register 2 (Graphics 3) (GR3\_AB2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR3_GRC_VS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR3_GRC_VW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR3_GRC_VS[10:0]	0	R/W	Sets the vertical start position of the graphics image area. Note: Set to 4 or greater lines and the result of GR3_GRC_VS + GR3_GRC_VW should be smaller than or equal to 2039 lines.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR3_GRC_VW[10:0]	0	R/W	Sets the vertical width of the graphics image area.

Note: This register is updated when GR3\_P\_VEN in GR3\_UPDATE is 1.

## 35.2.33 Alpha Blending Control Register 3 (Graphics 3) (GR3\_AB3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR3_GRC_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR3_GRC_HW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR3_GRC_HS[10:0]	0	R/W	Sets the horizontal start position of the graphics image area. Note: Set to 16 or greater clocks and the result of GR3_GRC_HS + GR3_GRC_HW should be smaller than or equal to 2015 clocks.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR3_GRC_HW[10:0]	0	R/W	Sets the horizontal width of the graphics image area. Note: For displaying an image with 1- or 2-pixel horizontal width, set GR3_HW to 2 and GR3_GRC_HW to 1 (1-pixel) or 2 (2-pixel).

Note: All the bits assigned to this address are updated when GR3\_P\_VEN in GR3\_UPDATE is 1.

## 35.2.34 Alpha Blending Control Register 4 (Graphics 3) (GR3\_AB4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR3_ARC_VS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR3_ARC_VW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR3_ARC_VS[10:0]	0	R/W	Sets the vertical start position of the valid image area for alpha blending in a rectangular area.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR3_ARC_VW[10:0]	0	R/W	Sets the vertical width of the valid image area for alpha blending in a rectangular area.

Note: This register is updated when GR3\_P\_VEN in GR3\_UPDATE is 1.

## 35.2.35 Alpha Blending Control Register 5 (Graphics 3) (GR3\_AB5)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR3_ARC_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR3_ARC_HW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR3_ARC_HS[10:0]	0	R/W	Sets the horizontal start position of the valid image area for alpha blending in a rectangular area.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR3_ARC_HW[10:0]	0	R/W	Sets the horizontal width of the valid image area for alpha blending in a rectangular area.

Note: This register is updated when GR3\_P\_VEN in GR3\_UPDATE is 1.

## 35.2.36 Alpha Blending Control Register 6 (Graphics 3) (GR3\_AB6)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	GR3_ARC_MODE	GR3_ARC_COEF[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	GR3_ARC_RATE[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	GR3_ARC_MODE	0	R/W	Alpha Blending Mode in Rectangular Area 0: Addition 1: Subtraction
23 to 16	GR3_ARC_COEF[7:0]	0	R/W	Sets the alpha coefficient for alpha blending in a rectangular area. (0 to 255) [7:0]: Variation (absolute value)
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	GR3_ARC_RATE[7:0]	0	R/W	Sets the value obtained by subtracting 1 from the frame rate for alpha blending in a rectangular area.

Note: This register is updated when GR3\_P\_VEN in GR3\_UPDATE is 1.

## 35.2.37 Alpha Blending Control Register 7 (Graphics 3) (GR3\_AB7)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	GR3_ARC_DEF[7:0]							
Initial value:	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR3_CK_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	GR3_ARC_DEF[7:0]	255	R/W	Sets the initial alpha value for alpha blending in a rectangular area.
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR3_CK_ON	0	R/W	Turns on/off CLUT-index/RGB-index chroma-key processing. 0: Off 1: On

Note: This register is updated when GR3\_P\_VEN in GR3\_UPDATE is 1.

## 35.2.38 Alpha Blending Control Register 8 (Graphics 3) (GR3\_AB8)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR3_CK_KCLUT[7:0]								GR3_CK_KG[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR3_CK_KB[7:0]								GR3_CK_KR[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GR3_CK_KCLUT[7:0]	0	R/W	CLUT Signal for CLUT-Index Chroma-Key Processing CLUT: Unsigned 8 bits (0 to 255 [LSB])
23 to 16	GR3_CK_KG[7:0]	0	R/W	G Signal for RGB-Index Chroma-Key Processing G: Unsigned 8 bits (0 to 255 [LSB])
15 to 8	GR3_CK_KB[7:0]	0	R/W	B Signal for RGB-Index Chroma-Key Processing B: Unsigned 8 bits (0 to 255 [LSB])
7 to 0	GR3_CK_KR[7:0]	0	R/W	R Signal for RGB-Index Chroma-Key Processing R: Unsigned 8 bits (0 to 255 [LSB])

Note: This register is updated when GR3\_P\_VEN in GR3\_UPDATE is 1.

## 35.2.39 Alpha Blending Control Register 9 (Graphics 3) (GR3\_AB9)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR3_CK_A[7:0]								GR3_CK_G[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR3_CK_B[7:0]								GR3_CK_R[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GR3_CK_A [7:0]	0	R/W	Replaced Alpha Signal after RGB/CLUT-Index Chroma-Key Processing $\alpha$ : Unsigned 8 bits (0 to 255 [LSB])
23 to 16	GR3_CK_G [7:0]	0	R/W	Replaced G Signal after RGB/CLUT-Index Chroma-Key Processing G: Unsigned 8 bits (0 to 255 [LSB])
15 to 8	GR3_CK_B [7:0]	0	R/W	Replaced B Signal after RGB/CLUT-Index Chroma-Key Processing B: Unsigned 8 bits (0 to 255 [LSB])
7 to 0	GR3_CK_R [7:0]	0	R/W	Replaced R Signal after RGB/CLUT-Index Chroma-Key Processing R: Unsigned 8 bits (0 to 255 [LSB])

Note: This register is updated when GR3\_P\_VEN in GR3\_UPDATE is 1.

## 35.2.40 Alpha Blending Control Register 10 (Graphics 3) (GR3\_AB10)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR3_A0[7:0]								GR3_G0[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR3_B0[7:0]								GR3_R0[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GR3_A0 [7:0]	0	R/W	CLUT1 $\alpha$ 0 Signal Replaced with $\alpha$ signal when in the CLUT1 format and CLUT1 = 0. Replaced with $\alpha$ signal when in the $\alpha$ RGB1555 or RGB $\alpha$ 5551 format and $\alpha$ = 0.
23 to 16	GR3_G0 [7:0]	0	R/W	CLUT1 G0 Signal Replaced with G signal when in the CLUT1 format and CLUT1 = 0.
15 to 8	GR3_B0 [7:0]	0	R/W	CLUT1 B0 Signal Replaced with B signal when in the CLUT1 format and CLUT1 = 0.
7 to 0	GR3_R0 [7:0]	0	R/W	CLUT1 R0 Signal Replaced with R signal when in the CLUT1 format and CLUT1 = 0.

Note: This register is updated when GR3\_P\_VEN in GR3\_UPDATE is 1.

## 35.2.41 Alpha Blending Control Register 11 (Graphics 3) (GR3\_AB11)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR3_A1[7:0]								GR3_G1[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR3_B1[7:0]								GR3_R1[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GR3_A1 [7:0]	0	R/W	CLUT1 $\alpha$ 1 Signal Replaced with $\alpha$ signal when in the CLUT1 format and CLUT1 = 1. Replaced with $\alpha$ signal when in the $\alpha$ RGB1555 or RGB $\alpha$ 5551 format and $\alpha$ = 1.
23 to 16	GR3_G1 [7:0]	0	R/W	CLUT1 G1 Signal Replaced with G signal when in the CLUT1 format and CLUT1 = 1.
15 to 8	GR3_B1 [7:0]	0	R/W	CLUT1 B1 Signal Replaced with B signal when in the CLUT1 format and CLUT1 = 1.
7 to 0	GR3_R1 [7:0]	0	R/W	CLUT1 R1 Signal Replaced with R signal when in the CLUT1 format and CLUT1 = 1.

Note: This register is updated when GR3\_P\_VEN in GR3\_UPDATE is 1.

## 35.2.42 Background Color Control Register (Graphics 3) (GR3\_BASE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	GR3_BASE_G[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR3_BASE_B[7:0]								GR3_BASE_R[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	GR3_BASE_G [7:0]	0	R/W	Background Color G Signal G: Unsigned 8 bits (0 to 255 [LSB])
15 to 8	GR3_BASE_B [7:0]	0	R/W	Background Color B Signal B: Unsigned 8 bits (0 to 255 [LSB])
7 to 0	GR3_BASE_R [7:0]	0	R/W	Background Color R Signal R: Unsigned 8 bits (0 to 255 [LSB])

Note: This register is updated when GR3\_P\_VEN in GR3\_UPDATE is 1.

## 35.2.43 CLUT Table and Interrupt Control Register (Graphics 3) (GR3\_CLUT\_INT)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR3_CLT_SEL
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR3_LINE[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	GR3_CLT_SEL	0	R/W	CLUT Table Select Signal 0: Selects CLUT table 0. The format is converted to $\alpha$ RGB8888 based on the CLUT table 0. CLUT table 1 can be read from or written to by the CPU. 1: Selects CLUT table 1. The format is converted to $\alpha$ RGB8888 based on the CLUT table 1. CLUT table 0 can be read from or written to by the CPU.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.



Bit	Bit Name	Initial Value	R/W	Description
10 to 0	GR3_LINE [10:0]	0	R/W	Line Interrupt Set When number of lines matches the value of the GR3_LINE bits, an interrupt signal is output. This function is enabled even when the graphics 3 process is not used.

Note: This register is updated when GR3\_P\_VEN in GR3\_UPDATE is 1.

### 35.2.44 Status Monitor Register (Graphics 3) (GR3\_MON)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR3_LIN_STAT[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR3_ARC_ST
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR3_LIN_STAT[10:0]	0	R	Line Position of Image being Currently Read
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR3_ARC_ST	0	R	Status Flag for Alpha Blending in Rectangular Area 0: Addition or subtraction has been completed. ( $\alpha$ value is 0 or 255) 1: Addition or subtraction is in progress.

## 35.2.45 VIN Synthesizer Register Update Control Register (GR\_VIN\_UPDATE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	GR_VIN_UPDATE	—	—	—	GR_VIN_P_VEN	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/WC1	R	R	R	R/WC1	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	GR_VIN_UPDATE	0	R/WC1	Graphics Display Register Update 0: Registers are not updated. 1: Registers are updated.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	GR_VIN_P_VEN	0	R/WC1	Graphics Display Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

## 35.2.46 Alpha Blending Control Register 1 (VIN Synthesizer) (GR\_VIN\_AB1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	GR_VIN_ARC_ON	—	—	—	GR_VIN_ARC_DISP_ON	—	—	—	GR_VIN_GRC_DISP_ON	—	GR_VIN_SCL_UND_SEL	GR_VIN_DISP_SEL[1:0]	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	GR_VIN_ARC_ON	0	R/W	Turns on/off alpha blending in a rectangular area. 0: Off 1: On
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	GR_VIN_ARC_DISP_ON	0	R/W	Turns on/off frame-line display of the image area for alpha blending in a rectangular area. 0: Frame-line display off 1: Frame-line display on
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	GR_VIN_GRC_DISP_ON	0	R/W	Turns on/off frame-line display of the graphics image area. 0: Frame-line display off 1: Frame-line display on
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2	GR_VIN_SCL_UND_SEL	0	R/W	Selection of Lower-Layer Plane in Scaler 0: Selects graphics 0 as lower-layer graphics and graphics 1 as current graphics 1: Selects graphics 1 as lower-layer graphics and graphics 0 as current graphics
1, 0	GR_VIN_DISP_SEL [1:0]	0	R/W	Selects the graphics display mode. 0: Background color display 1: Lower-layer graphics display 2: Current graphics display 3: Blended display of lower-layer graphics and current graphics, or setting prohibited*

Notes: GR\_VIN\_SCL\_UND\_SEL is updated when GR\_VIN\_UPDATE in GR\_VIN\_UPDATE is 1. The other bits of this register are updated when GR\_VIN\_P\_VEN in GR\_VIN\_UPDATE is 1.

\* This setting is prohibited when the graphics block 0 in scaler 0 and graphics block 1 in scaler 1 are cascaded (GR1\_AB1.GR1\_CUS\_CON\_ON = 1).

## 35.2.47 Alpha Blending Control Register 2 (VIN Synthesizer) (GR\_VIN\_AB2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR_VIN_GRC_VS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR_VIN_GRC_VW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR_VIN_GRC_VS [10:0]	0	R/W	Sets the vertical start position of the graphics image area. Note: Set to 4 or greater lines and the result of GR_VIN_GRC_VS + GR_VIN_GRC_VW should be smaller than or equal to 2039 lines.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR_VIN_GRC_VW [10:0]	0	R/W	Sets the vertical width of the graphics image area.

Note: This register is updated when GR\_VIN\_P\_VEN in GR\_VIN\_UPDATE is 1.

## 35.2.48 Alpha Blending Control Register 3 (VIN Synthesizer) (GR\_VIN\_AB3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR_VIN_GRC_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR_VIN_GRC_HW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR_VIN_GRC_HS [10:0]	0	R/W	Sets the horizontal start position of the graphics image area. Note: Set to 16 or greater clocks and the result of GR_VIN_GRC_HS + GR_VIN_GRC_HW should be smaller than or equal to 2015 clocks.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR_VIN_GRC_HW [10:0]	0	R/W	Sets the horizontal width of the graphics image area. Note: For displaying an image with 1- or 2-pixel horizontal width, set GR_VIN_HW to 2 and GR_VIN_GRC_HW to 1 (1-pixel) or 2 (2-pixel).

Note: This register is updated when GR\_VIN\_P\_VEN in GR\_VIN\_UPDATE is 1.

## 35.2.49 Alpha Blending Control Register 4 (VIN Synthesizer) (GR\_VIN\_AB4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR_VIN_ARC_VS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR_VIN_ARC_VW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR_VIN_ARC_VS [10:0]	0	R/W	Sets the vertical start position of the valid image area for alpha blending in a rectangular area.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR_VIN_ARC_VW [10:0]	0	R/W	Sets the vertical width of the valid image area for alpha blending in a rectangular area.

Note: This register is updated when GR\_VIN\_P\_VEN in GR\_VIN\_UPDATE is 1.

## 35.2.50 Alpha Blending Control Register 5 (VIN Synthesizer) (GR\_VIN\_AB5)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR_VIN_ARC_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR_VIN_ARC_HW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR_VIN_ARC_HS [10:0]	0	R/W	Sets the horizontal start position of the valid image area for alpha blending in a rectangular area.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR_VIN_ARC_HW [10:0]	0	R/W	Sets the horizontal width of the valid image area for alpha blending in a rectangular area.

Note: This register is updated when GR\_VIN\_P\_VEN in GR\_VIN\_UPDATE is 1.

## 35.2.51 Alpha Blending Control Register 6 (VIN Synthesizer) (GR\_VIN\_AB6)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	GR_VIN_ARC_MODE	GR_VIN_ARC_COEF[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0							
R/W:	R	R	R	R	R	R	R	R/W	R/W							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	GR_VIN_ARC_RATE[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0							
R/W:	R	R	R	R	R	R	R	R	R/W							

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	GR_VIN_ARC_MODE	0	R/W	Alpha Blending Mode in Rectangular Area 0: Addition 1: Subtraction
23 to 16	GR_VIN_ARC_COEF [7:0]	0	R/W	Sets the alpha coefficient for alpha blending in a rectangular area. (0 to 255) [7:0]: Variation (absolute value)
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	GR_VIN_ARC_RATE [7:0]	0	R/W	Sets the value obtained by subtracting 1 from the frame rate for alpha blending in a rectangular area.

Note: This register is updated when GR\_VIN\_P\_VEN in GR\_VIN\_UPDATE is 1.

## 35.2.52 Alpha Blending Control Register 7 (VIN Synthesizer) (GR\_VIN\_AB7)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	GR_VIN_ARC_DEF[7:0]							
Initial value:	0	0	0	0	0	0	0	0	1							
R/W:	R	R	R	R	R	R	R	R	R/W							
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0							
R/W:	R	R	R	R	R	R	R	R	R							

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	GR_VIN_ARC_DEF [7:0]	255	R/W	Sets the initial alpha value for alpha blending in a rectangular area.
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note: This register is updated when GR\_VIN\_P\_VEN in GR\_VIN\_UPDATE is 1.

## 35.2.53 Background Color Control Register (VIN Synthesizer) (GR\_VIN\_BASE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	GR_VIN_BASE_G[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR_VIN_BASE_B[7:0]								GR_VIN_BASE_R[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	GR_VIN_BASE_G [7:0]	0	R/W	Background Color G Signal G: Unsigned 8 bits (0 to 255 [LSB])
15 to 8	GR_VIN_BASE_B [7:0]	0	R/W	Background Color B Signal B: Unsigned 8 bits (0 to 255 [LSB])
7 to 0	GR_VIN_BASE_R [7:0]	0	R/W	Background Color R Signal R: Unsigned 8 bits (0 to 255 [LSB])

Note: This register is updated when GR\_VIN\_P\_VEN in GR\_VIN\_UPDATE is 1.

## 35.2.54 Status Monitor Register (VIN Synthesizer) (GR\_VIN\_MON)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR_VIN_ARC_ST
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR_VIN_ARC_ST	0	R	Status Flag for Alpha Blending in Rectangular Area 0: Addition or subtraction has been completed. ( $\alpha$ value is 0 or 255) 1: Addition or subtraction is in progress.

### 35.3 Usage Method

#### 35.3.1 Mute Image

The initial values of the GR0\_DISP\_SEL[1:0], GR1\_DISP\_SEL[1:0], GR2\_DISP\_SEL[1:0], GR3\_DISP\_SEL[1:0], GR\_VIN\_DISP\_SEL[1:0], and GR\_OIR\_DISP\_SEL[1:0] bits are all 0. Accordingly, in the initial setting, a background color is displayed both inside and outside the graphics area for the graphics 0, 1, 2, 3, and OIR processes and the VIN synthesizer. Since the default background color is black, the black mute image is displayed in the initial state.

#### 35.3.2 Alpha Blending in Rectangular Area

The alpha coefficient and the frame rate can be changed during fade in and fade out by modifying the GR\_ARC\_MODE, GR\_ARC\_COEF[7:0] and GR\_ARC\_RATE[7:0] bits, respectively.



## 36. Video Display Controller 5 (6): Output Image Generator

### 36.1 Output Image Generation Functions

#### 36.1.1 Overview of Functions

The output image generator can be used to store the RGB output from the image synthesizer in the frame buffer. In addition to storing the RGB data, it can also read the graphics data and display them. In cooperation with the image renderer for display (channel 0 only), it can also distort the output image according to a display panel.

When the display functions described above (reading and displaying of graphics data and displaying of an image after distortion) are not in use, the signals from the image synthesizer can be output directly to the output control block, bypassing the output image generator.

The output image generator has the same circuit configuration as the scaler. However, note the following differences from the scaler.

- (1) The scaling-up/-down function is not available.
- (2) The rotation control function is not available.
- (3) The pointer buffer function is not available.
- (4) The field control function is not available because the inputs are always progressive signals.
- (5) The I/O signals are always RGB signals.

The functional block diagram of the output image generator is shown below.

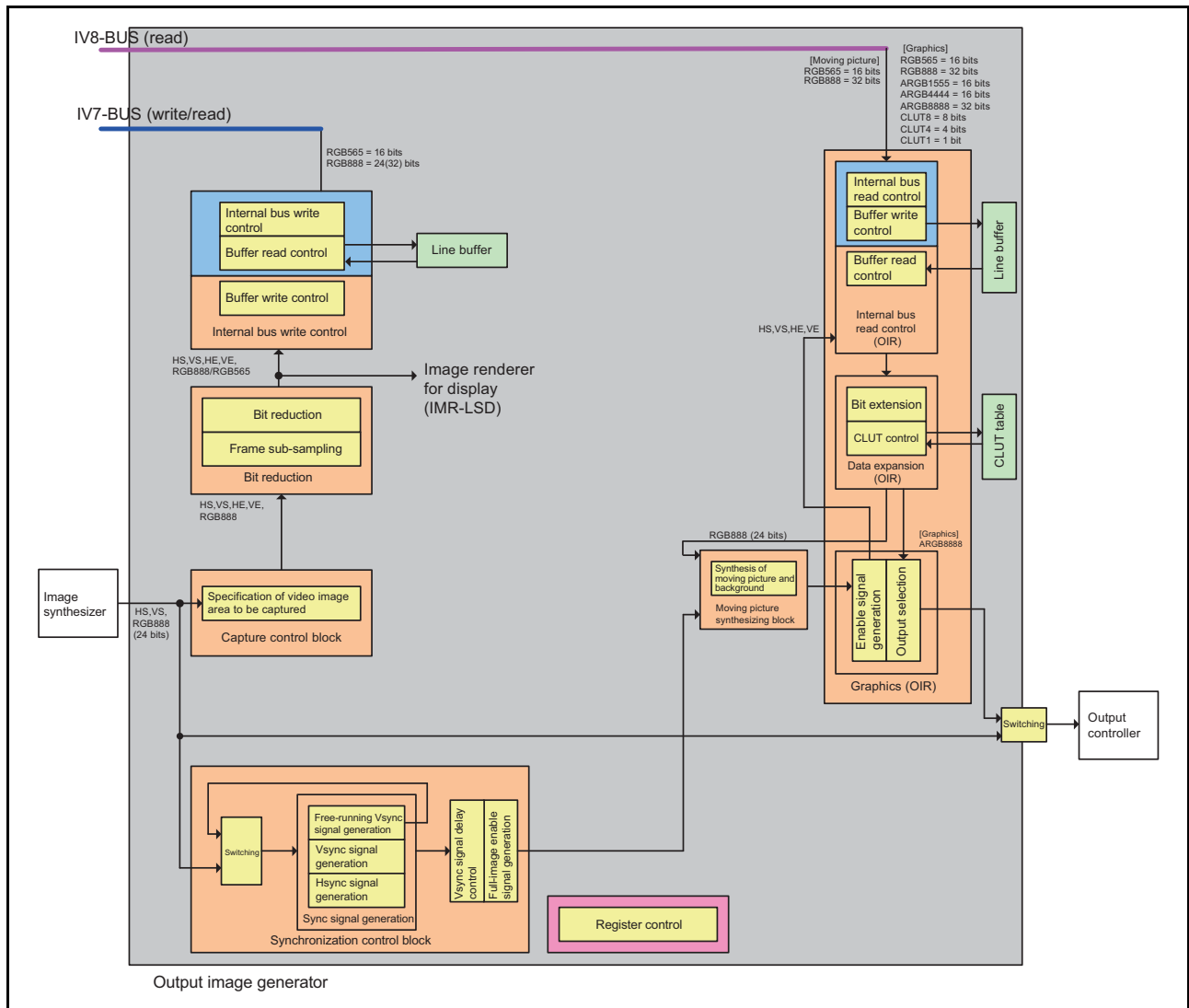


Figure 36.1 Functional Block Diagram of Output Image Generator

## 36.1.2 Register Control

### (1) Updating Registers

The Vsync signal is used to control the update timing of all the registers of the output image generator and graphics block (OIR) except some registers of the sync control block.

After 1 is set to the bits in the update control register, the contents of the relevant registers are modified at the rising edge of the Vsync signal. The update control register is automatically cleared to 0 after the modification.

**Table 36.1 Register Update Control**

Register Name	Bit Name	Initial Value	Description
OIR_SCL0_UPDATE	OIR_SCL0_UPDATE	0	SYNC Control Register Update 0: Registers are not updated. 1: Registers are updated.
OIR_SCL0_UPDATE	OIR_SCL0_VEN_D	0	Frame Buffer Read Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
OIR_SCL0_UPDATE	OIR_SCL0_VEN_C	0	Frame Buffer Read Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
OIR_SCL0_UPDATE	OIR_SCL0_VEN_B	0	Synchronization Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
OIR_SCL0_UPDATE	OIR_SCL0_VEN_A	0	Scaling-Down Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
OIR_SCL1_UPDATE	OIR_SCL1_UPDATE_A	0	Frame Buffer Write Control Register Update 0: Registers are not updated. 1: Registers are updated.
OIR_SCL1_UPDATE	OIR_SCL1_VEN_B	0	Frame Buffer Write Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
OIR_SCL1_UPDATE	OIR_SCL1_VEN_A	0	Frame Buffer Write Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
GR_OIR_VEN	GR_OIR_P_VEN	0	Graphics Display Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
GR_OIR_VEN	GR_OIR_IBUS_VEN	0	Frame Buffer Read Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.

The registers controlled by OIR\_SCL0\_VEN\_A, OIR\_SCL0\_VEN\_C, OIR\_SCL1\_VEN\_A, and OIR\_SCL1\_VEN\_B are modified at the rising edge of the input Vsync signal.

The registers controlled by OIR\_SCL0\_VEN\_B, OIR\_SCL0\_VEN\_D, GR\_OIR\_P\_VEN, and GR\_OIR\_IBUS\_VEN are modified at the rising edge of the output Vsync signal.

### 36.1.3 Enabling or Disabling Output Image Generator

The output signal from the image synthesizer can be selected.

**Table 36.2 Output Image Generator Enable Control**

Register Name	Bit Name	Initial Value	Description
OIR_SCL0_FRC3	OIR_RES_EN	0	Enabling or Disabling the Output Image Generator 0: The input data from the image synthesizer is directly output. 1: Signals from the graphics block (OIR) are output.

### 36.1.4 Synchronization Control

#### (1) Selecting Vsync Signal

The Vsync signal to be output from the output image generator can be selected.

**Table 36.3 Vsync Signal Selection Control**

Register Name	Bit Name	Initial Value	Description
OIR_SCL0_FRC3	OIR_RES_VS_SEL	1	Vsync Signal Output Select 0: Vsync signal input from the image synthesizer 1: Internally generated free-running Vsync signal

#### (2) Masking Repeated Vsync Signals

Take measures against the repeated Vsync signals by using the Vsync control block in the scaler. Accordingly, repeated Vsync signal masking control must be disabled in the output image generator when it is in use.

**Table 36.4 Repeated Vsync Signal Mask Control**

Register Name	Bit Name	Initial Value	Description
OIR_SCL0_FRC1	OIR_RES_VMASK_ON	1	Repeated Vsync Signal Masking Control This bit should always be set to 0 when the output image generator is in use. 0: Repeated Vsync signal masking control is disabled. 1: Repeated Vsync signal masking control is enabled.

#### (3) Compensating for Missing Vsync Signals

Take measures against the missed Vsync signals by using the Vsync control block in the scaler. Accordingly, compensation of missing Vsync signals must be disabled in the output image generator when it is in use.

**Table 36.5 Missing Vsync Compensation Control**

Register Name	Bit Name	Initial Value	Description
OIR_SCL0_FRC2	OIR_RES_VLACK_ON	1	Missing Vsync Signal Compensation This bit should always be set to 0 when the output image generator is in use. 0: Compensation of missing Vsync signals is disabled. 1: Compensation of missing Vsync signals is enabled.

#### (4) Free-Running Period

Free-running Vsync and Hsync periods can be set.

$$\text{Hsync period [usec]} = (\text{RES\_FH} + 1) \div \text{pixel clock frequency [MHz]}$$

$$\text{Vsync period [usec]} = \text{horizontal period [usec]} \times (\text{RES\_FV} + 1)$$

**Table 36.6 Free-Running Period Control**

Register Name	Bit Name	Initial Value	Description
OIR_SCL0_FRC4	OIR_RES_FV [10:0]	524	Free-Running Vsync Period Setting Free-running Vsync period = (OIR_RES_FV + 1) × horizontal period [usec]
OIR_SCL0_FRC4	OIR_RES_FH [10:0]	799	Hsync Period Setting Hsync period [usec] = (OIR_RES_FH + 1) ÷ pixel clock frequency [MHz]

When selecting a Vsync signal input from the image synthesizer, set the OIR\_RES\_VS\_SEL bit to 0. At this time, the internally generated free-running Vsync signal is not output.

In the meantime, the Hsync signal is always generated according to the free-running signal setting and output from the output image generator.

#### (5) Vsync Signal Delay Control

Delay of Vsync signal output from the output image generator can be controlled.

The delay is used to adjust the frame buffer read timing.

**Table 36.7 Vsync Output Delay Control**

Register Name	Bit Name	Initial Value	Description
OIR_SCL0_FRC5	OIR_RES_VSDLY [7:0]	1	Vsync Signal Delay Control Adjusts the Vsync signal delay in the output Hsync period units. Vsync signal delay [usec]: OIR_RES_VSDLY × output Hsync period [usec]

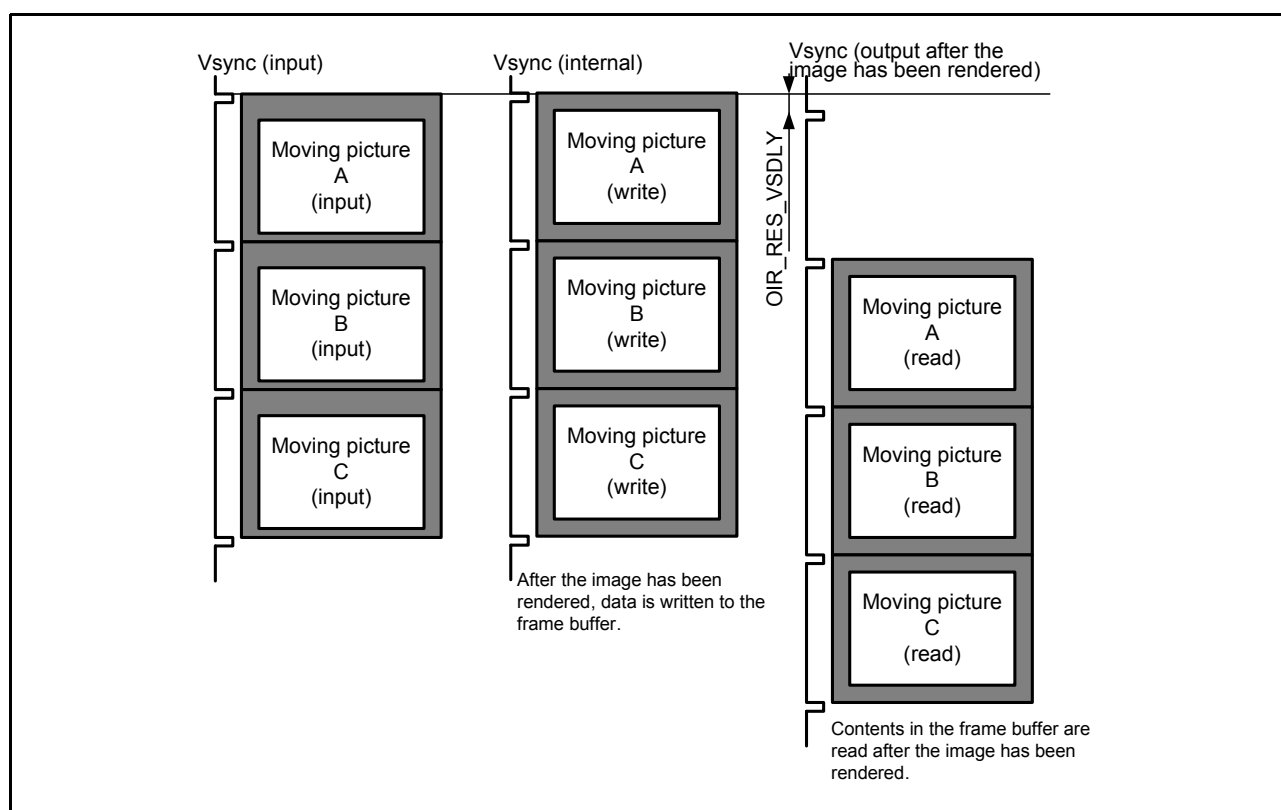


Figure 36.2 Vsync Signal Phases (Two Frame-Buffer Planes Used)

### 36.1.5 Setting Angle of View

#### (1) Setting Image Area to be Captured

The area to be captured for the input image can be set.

The area is defined by specifying its start position and width based on the input Hsync and Vsync signals.

Table 36.8 Control of Image Area to be Captured

Register Name	Bit Name	Initial Value	Description
OIR_SCL0_DS2	OIR_RES_VS [10:0]	18	Vertical Position Setting for Video Signal Capturing (VSYNC + V backporch lines) Note: The set value should be four or more (lines). OIR_RES_VS + OIR_RES_VW should be equal to or less than 2039 (lines).
OIR_SCL0_DS2	OIR_RES_VW [10:0]	240	Vertical Width of Video Signal to be Captured (lines) Note: OIR_RES_VS + OIR_RES_VW should be equal to or less than 2039 (lines).
OIR_SCL0_DS3	OIR_RES_HS [10:0]	244	Horizontal Position Setting for Video Signal Capturing (HSYNC + H backporch video-image clock cycles) Note: The set value should be 16 or more (clock cycles). OIR_RES_HS + OIR_RES_HW should be equal to or less than 2015 (clock cycles).
OIR_SCL0_DS3	OIR_RES_HW [10:0]	1440	Horizontal Width of Video Signal to be Captured (video-image clock cycles) Note: OIR_RES_HS + OIR_RES_HW should be equal to or less than 2015 (clock cycles).

## (2) Generating a Full-Screen Enable Signal

The valid period of the full screen to be output from the output image generator can be set.

The valid period is defined by specifying its start position and width based on the Hsync and Vsync signals output from the output image generator.

**Table 36.9 Full-Screen Enable Control**

Register Name	Bit Name	Initial Value	Description
OIR_SCL0_FRC6	OIR_RES_F_VS [10:0]	35	Vertical Enable Signal Start Position for Full Screen. (VSYNC + V backporch lines)  Note: The set value should be four or more (lines). OIR_RES_F_VS + OIR_RES_F_VW should be equal to or less than 2039 (lines).
OIR_SCL0_FRC6	OIR_RES_F_VW [10:0]	480	Vertical Enable Signal Width for Full Screen (lines)  Note: OIR_RES_F_VS + OIR_RES_F_VW should be equal to or less than 2039 (lines).
OIR_SCL0_FRC7	OIR_RES_F_HS [10:0]	144	Horizontal Enable Signal Start Position for Full Screen. (HSYNC + H backporch pixel-clock cycles)  Note: The set value should be 16 or more (clock cycles). OIR_RES_F_HS + OIR_RES_F_HW should be equal to or less than 2015 (clock cycles).
OIR_SCL0_FRC7	OIR_RES_F_HW [10:0]	640	Horizontal Enable Signal Width for Full Screen (pixel-clock cycles)  Note: OIR_RES_F_HS + OIR_RES_F_HW should be equal to or less than 2015 (clock cycles).

## (3) Generating an Image Output Enable Signal

The valid period of the image to be output can be set.

The valid period is defined by specifying its start position and width based on the Hsync and Vsync signals output from the output image generator.

**Table 36.10 Image Output Enable Control**

Register Name	Bit Name	Initial Value	Description
OIR_SCL0_US2	OIR_RES_P_VS [10:0]	35	Vertical Enable Signal Start Position for Output Image. (VSYNC + V backporch lines)  Note: The set value should be four or more (lines). OIR_RES_P_VS + OIR_RES_P_VW should be equal to or less than 2039 (lines).
OIR_SCL0_US2	OIR_RES_P_VW [10:0]	480	Vertical Enable Signal Width for Output Image (lines)  Note: OIR_RES_P_VS + OIR_RES_P_VW should be equal to or less than 2039 (lines).
OIR_SCL0_US3	OIR_RES_P_HS [10:0]	144	Horizontal Enable Signal Start Position for Output Image. (HSYNC + H backporch pixel-clock cycles)  Note: The set value should be 16 or more (clock cycles). OIR_RES_P_HS + OIR_RES_P_HW should be equal to or less than 2015 (clock cycles).
OIR_SCL0_US3	OIR_RES_P_HW [10:0]	640	Horizontal Enable Signal Width for Output Image (pixel-clock cycles)  Note: OIR_RES_P_HS + OIR_RES_P_HW should be equal to or less than 2015 (clock cycles).

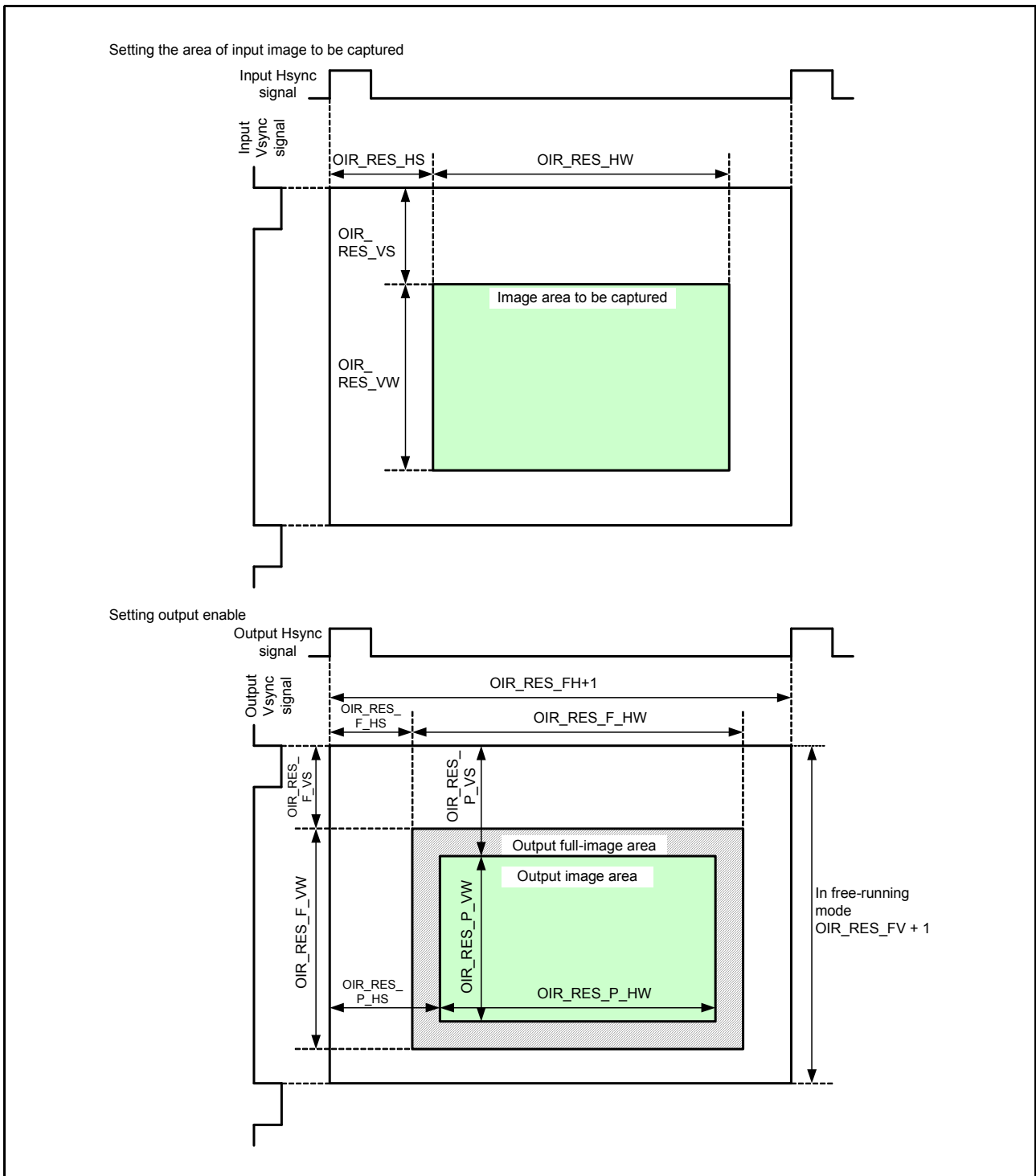


Figure 36.3 Enable Settings



### 36.1.6 Scaling Settings

Scaling up and down is not available in the output image generator. Accordingly, scaling up and down control must be off when the output image generator is in use.

**Table 36.11 Scaling-Up/-Down Control**

Register Name	Bit Name	Initial Value	Description
OIR_SCL0_DS1	OIR_RES_DS_H_ON	1	Horizontal Scale Down On/Off This bit should always be set to 0 when the output image generator is in use. 0: Off 1: On
OIR_SCL0_DS1	OIR_RES_DS_V_ON	1	Vertical Scale Down On/Off This bit should always be set to 0 when the output image generator is in use. 0: Off 1: On
OIR_SCL0_US1	OIR_RES_US_H_ON	1	Horizontal Scale Up On/Off This bit should always be set to 0 when the output image generator is in use. 0: Off 1: On
OIR_SCL0_US1	OIR_RES_US_V_ON	1	Vertical Scale Up On/Off This bit should always be set to 0 when the output image generator is in use. 0: Off 1: On

### 36.1.7 Screen Synthesis

During the valid full-screen period, the image area can be overlaid before being output. If the image area to be output is smaller than a full-screen, the background color specified by the OIR\_RES\_BK\_COL\_R, OIR\_RES\_BK\_COL\_G, and OIR\_RES\_BK\_COL\_B bits are displayed to fill the background.

**Table 36.12 Screen Synthesis Control**

Register Name	Bit Name	Initial Value	Description
OIR_SCL0_OVR1	OIR_RES_BK_COL_R [7:0]	128	Background Color Setting R/Cr Signal R: 8 bits; unsigned (0 to 255 [LSB]) Cr: 8 bits; 128 offset binary; unsigned (0 to 255 [LSB])
OIR_SCL0_OVR1	OIR_RES_BK_COL_B [7:0]	128	Background Color Setting B/Cb Signal B: 8 bits; unsigned (0 to 255 [LSB]) Cb: 8 bits; 128 offset binary; unsigned (0 to 255 [LSB])
OIR_SCL0_OVR1	OIR_RES_BK_COL_G [7:0]	0	Background Color Setting G/Y Signal G/Y: 8 bits; unsigned (0 to 255 [LSB])

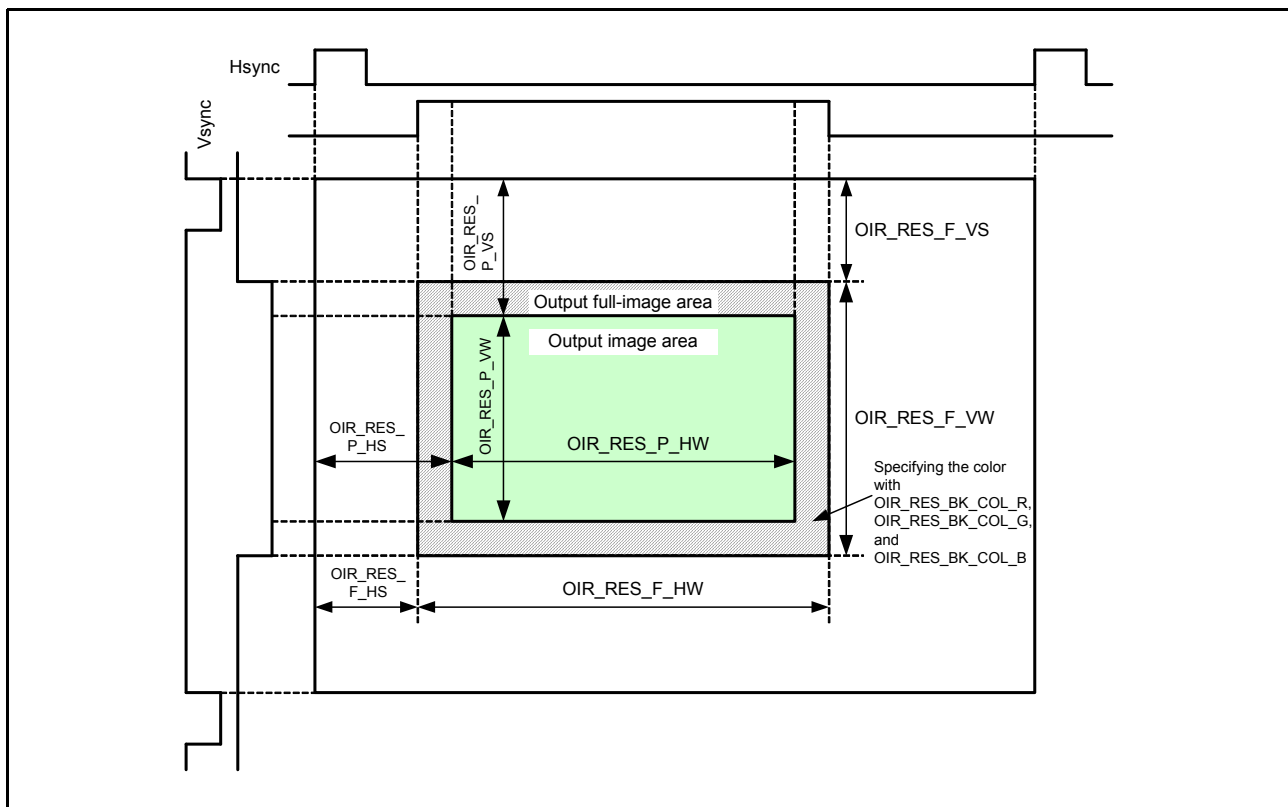


Figure 36.4 Area Relationship with Output Image Size Smaller than a Full Screen

### 36.1.8 Selecting Format for Writing Video Image Signals to Frame Buffer

A format can be selected for writing video image signals to the frame buffer.

Although 24-bit RGB signals are input to the output image generator, they are converted into 16-bit RGB565 signals or 32-bit RGB888 signals before being written to the frame buffer.

As bit reduction processing of RGB565, rounding off or  $2 \times 2$  pattern dither can be selected with the OIR\_RES\_DTH\_ON bit. For details on pattern dither, see Figure 37.1.7, Dither Process in section 37, Video Display Controller 5 (7): Output Controller.

RGB888 signals are output to the image renderer for display (this function is only available in channel 0).

For distortion correction for display, refer to section 42, Image Renderer for Display (IMR-LSD).

Table 36.13 Frame Buffer Writing Mode Setting

OIR_RES_BITDEC_ON	OIR_RES_MD[1:0]	Writing Mode
0	2	RGB888 (normal, distortion correction)
1	1	RGB565 (normal)

**Table 36.14 Video Signal Writing Format Selection Control**

Register Name	Bit Name	Initial Value	Description
OIR_SCL1_WR1	OIR_RES_MD[1:0]	0	Frame Buffer Video-Signal Writing Format 0: Setting prohibited 1: RGB565 (16 bits) 2: RGB888 (24 (32) bits) 3: Setting prohibited
OIR_SCL1_WR6	OIR_RES_BITDEC_ON	0	Bit Reduction On/Off 0: Off 1: On
OIR_SCL1_WR6	OIR_RES_DTH_ON	0	Dither Correction On/Off 0: Off (rounded off) 1: On (2 × 2 pattern dither)

### 36.1.9 Writing to Frame Buffer

#### (1) Frame Buffer Transfer Mode

Either 32-byte or 128-byte transfer mode can be selected for accessing the frame buffer in which video image data and graphics data are stored.

**Table 36.15 Frame Buffer Transfer Mode**

Register Name	Bit Name	Initial Value	Description
OIR_SCL1_WR1	OIR_RES_BST_MD	0	Transfer Burst Length for Frame Buffer Writing 0: 32-byte transfer (4 bursts) 1: 128-byte transfer (16 bursts)

#### (2) Frame Buffer Write Control

Frame buffer writing is enabled or disabled.

**Table 36.16 Frame Buffer Writing Control**

Register Name	Bit Name	Initial Value	Description
OIR_SCL1_WR5	OIR_RES_WENB	0	Frame Buffer Write Enable After making the setting to enable writing, writing starts from the second frame. 0: Frame buffer writing is disabled. 1: Frame buffer writing is enabled.

#### (3) Frame Buffer Writing Rate Selection

A frame buffer writing rate can be selected from among 1/1, 1/2, 1/4, and 1/8 the vertical frequency of the input signal.

When 1/2, 1/4, or 1/8 is selected, either the top or bottom field can be selected for writing.

The field operating mode should always be progressive because the progressive signals are always input to the output image generator.

**Table 36.17 Frame Buffer Write Control**

Register Name	Bit Name	Initial Value	Description
OIR_SCL1_WR5	OIR_RES_FS_RATE[1:0]	0	Writing Rate Sets the frame buffer writing rate to the vertical frequency of the input signal. 0: 1/1 an input signal (The OIR_RES_FLD_SEL setting is invalid.) 1: 1/2 an input signal 2: 1/4 an input signal 3: 1/8 an input signal
OIR_SCL1_WR5	OIR_RES_INTER	1	Field Operating Mode Select This bit should always be set to 0 when the output image generator is in use. 0: Progressive 1: Interlace

#### (4) Frame Buffer Write Addresses

Frame buffer addresses are specified using the base address, line offset address, frame offset address, data size of a line, and the number of lines in a frame.

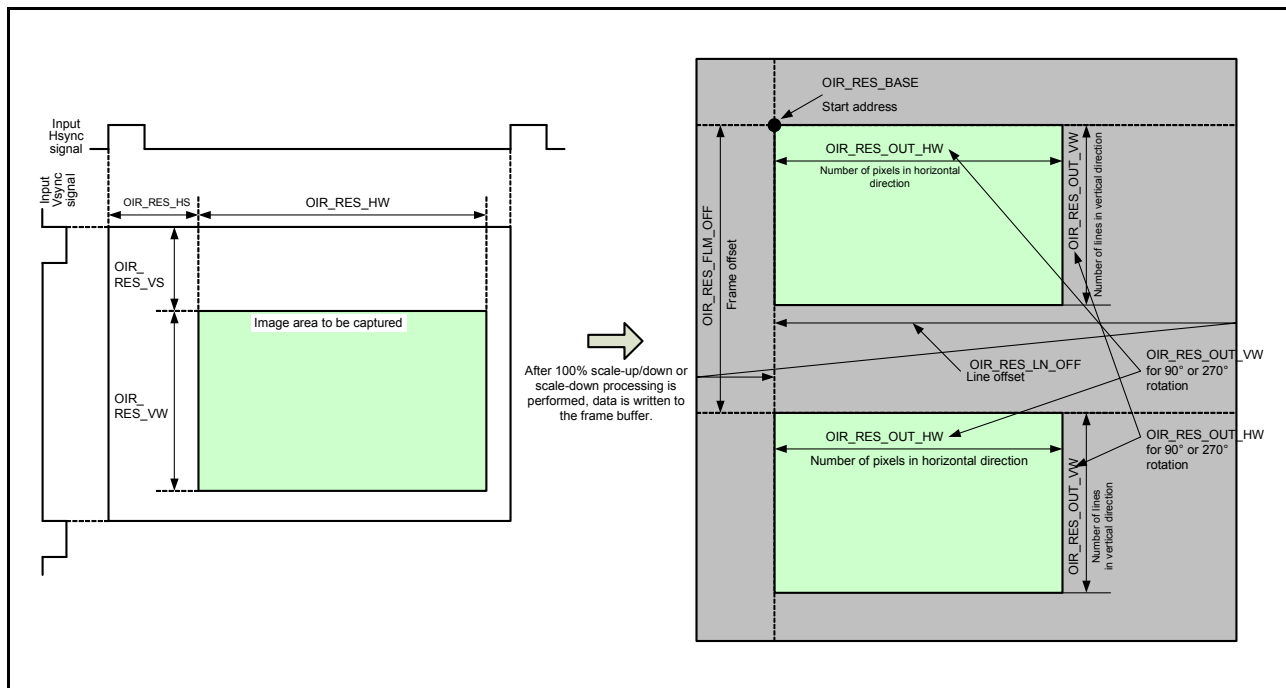
The OIR\_RES\_BASE[31:0], OIR\_RES\_LN\_OFF[14:0], and OIR\_RES\_FLM\_OFF[22:0] bits should be set in 32-byte units (the lower five bits should be fixed to 0).

For 128-byte transfer, bits [6:5] in the address control registers should be fixed to 0 since addresses should be specified in 128-byte units.

For the data size of a line and the number of lines in a frame, the relevant register values set for the scaling-down control block are used.

**Table 36.18 Frame Buffer Write Address Control**

Register Name	Bit Name	Initial Value	Description
OIR_SCL1_WR2	OIR_RES_BASE[31:0]	0	<p>Frame Buffer Base Address</p> <p>Sets the start address of the frame buffer where frame data is to be stored.</p> <p>For 32-byte transfer: The lower five bits should be fixed to 0_0000.</p> <p>For 128-byte transfer: The lower seven bits should be fixed to 000_0000.</p>
OIR_SCL1_WR3	OIR_RES_LN_OFF [14:0]	2048	<p>Frame Buffer Line Offset Address</p> <p>Sets the line offset address for calculating the start address of each line.</p> <p>Line 0: OIR_RES_BASE</p> <p>Line 1: OIR_RES_BASE + OIR_RES_LN_OFF × 1</p> <p>⋮</p> <p>Line n: OIR_RES_BASE + OIR_RES_LN_OFF × n</p> <p>For 32-byte transfer: The lower five bits should be fixed to 0_0000.</p> <p>For 128-byte transfer: The lower seven bits should be fixed to 000_0000.</p>
OIR_SCL1_WR4	OIR_RES_FLM_OFF [22:0]	524288	<p>Frame Buffer Frame Offset Address</p> <p>Sets the frame offset address for calculating the start address of each frame.</p> <p>Buffer 0: OIR_RES_BASE</p> <p>Buffer 1: OIR_RES_BASE + OIR_RES_FLM_OFF × 1</p> <p>⋮</p> <p>Buffer n: OIR_RES_BASE + OIR_RES_FLM_OFF × n</p> <p>For 32-byte transfer: The lower five bits should be fixed to 0_0000.</p> <p>For 128-byte transfer: The lower seven bits should be fixed to 000_0000.</p>



**Figure 36.5 Data Arrangement in Frame Buffer**

### (5) Frame Buffer Management

The output image generator can handle multiple frames as the frame buffer.

Data is written to the buffer in cyclic mode according to the number of frames specified by the OIR\_RES\_FLM\_NUM bits.

**Table 36.19 Frame Buffer Write Control**

Register Name	Bit Name	Initial Value	Description
OIR_SCL1_WR3	OIR_RES_FLM_NUM[9:0]	1	Number of Frames of Buffer to be Written to Number of frames defined by OIR_RES_FLM_NUM + 1 are used. For video recording: the number of frames to be stored – 1
OIR_SCL1_WR7	OIR_RES_FLM_CNT[9:0]	—	Frame Number Before Frame Being Accessed

### (6) Buffer Overflow Handling

If writing to the frame buffer cannot be completed due to bus-traffic related problems, an overflow interrupt can be output to the interrupt controller.

**Table 36.20 Buffer Overflow Detection**

Register Name	Bit Name	Initial Value	Description
OIR_SCL1_WR7	OIR_RES_OVERFLOW	—	Line Buffer Overflow Detect 1: Line buffer has overflowed. 0: Line buffer has not overflowed.

### (7) Frame Buffer Write End Flag

When writing one frame of data to the frame buffer is completed, a frame buffer write end interrupt can be output to the interrupt controller.

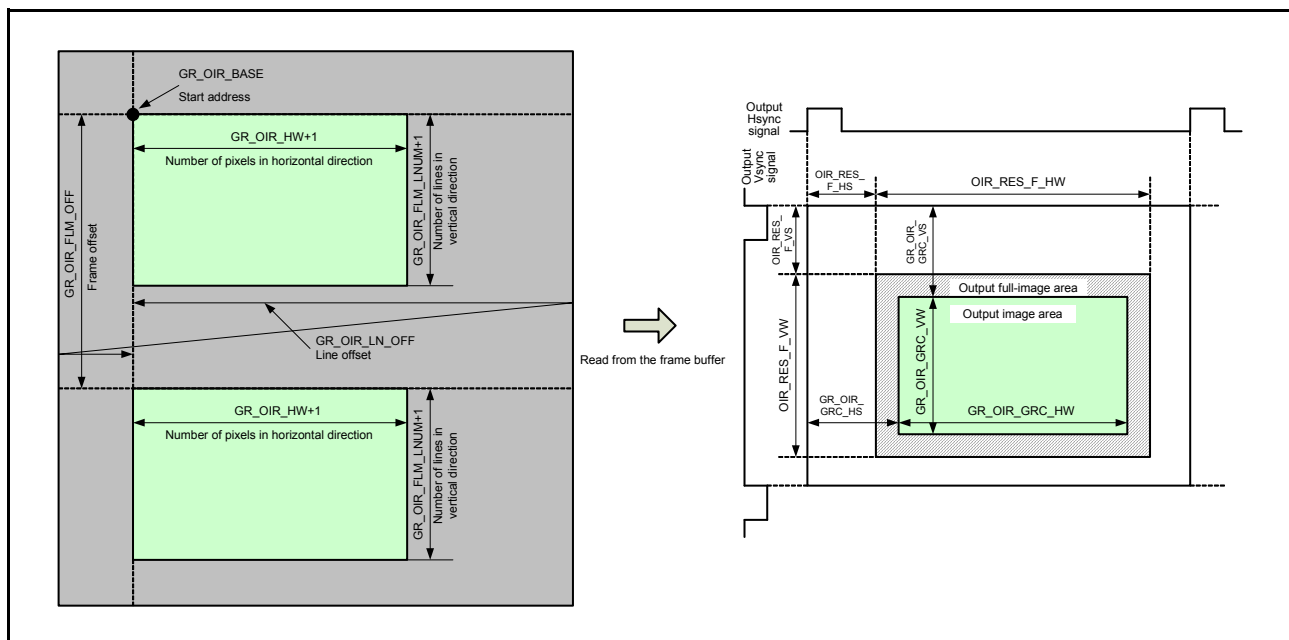
### 36.1.10 Selecting an Input Video Image Signal Processing or Graphics (OIR) Processing

Scale-up processing is not possible in the generator. Accordingly, set the OIR\_RES\_IBUS\_SYNC\_SEL bit to 1 to select graphics display.

**Table 36.21 Selection of Output Image Control Block or Graphics (OIR) Processing Block**

Type of Output Image Generation Display	OIR_RES_IBUS_SYNC_SEL	Sync Signals for Frame Buffer Read	Frame Buffer Read Size Setting Bits	Display Enabling Bits
Graphics display	1	Output from graphics (OIR) processing block	GR_OIR_FLM_LNUM* GR_OIR_HW*	GR_OIR_GRC_VS GR_OIR_GRC_VW GR_OIR_GRC_HS GR_OIR_GRC_HW

Note: \* The value set to the register + 1 is the actual read size.



**Figure 36.6 Area Setting for Graphics Display**

**Table 36.22 Output Image Control Block or Graphics (OIR) Processing Block Selection Control**

Register Name	Bit Name	Initial Value	Description
OIR_SCL0_US8	OIR_RES_IBUS_SYNC_SEL	0	Sync Signal Select for Frame Buffer Read Block This bit should always be set to 1 when the output image generator is in use. 0: Sync signals from the output image control block 1: Sync signals from the graphics processing block

For details on the graphics processing, refer to the section 35, Video Display Controller 5 (5): Image Synthesizer.

## 36.2 Register Descriptions

Table 36.23 and Table 36.24 show the register configuration.

- Symbols used in Register Description:

Initial value: Register value after a reset

—: Undefined value

R/W: Readable/writable. The written value can be read.

R/WC0: Readable/writable. Writing 0 initializes the bit. Writing 1 is ignored.

R/WC1: Readable/writable. Writing 1 initializes the bit. Writing 0 is ignored.

R: Read-only. The write value should always be 0.

—/W: Write-only. The read value is undefined.

**Table 36.23 Register Configuration of the Output Image Generator (Ch0)**

Name	Abbreviation	R/W	Initial Value	Address	Access Size
SCL0 register update control register (OIR)	OIR_SCL0_UPDATE	R/WC1	H'0000 0000	H'FCFF 7E80	32
Mask control register (OIR)	OIR_SCL0_FRC1	R/W	H'0AF0 0001	H'FCFF 7E84	32
Missing Vsync compensation control register (OIR)	OIR_SCL0_FRC2	R/W	H'0E10 0001	H'FCFF 7E88	32
Output sync select register (OIR)	OIR_SCL0_FRC3	R/W	H'0000 0001	H'FCFF 7E8C	32
Free-running period control register (OIR)	OIR_SCL0_FRC4	R/W	H'020C 031F	H'FCFF 7E90	32
Output delay control register (OIR)	OIR_SCL0_FRC5	R/W	H'0000 0101	H'FCFF 7E94	32
Full-screen vertical size register (OIR)	OIR_SCL0_FRC6	R/W	H'0023 01E0	H'FCFF 7E98	32
Full-screen horizontal size register (OIR)	OIR_SCL0_FRC7	R/W	H'0090 0280	H'FCFF 7E9C	32
Scaling-down control register (OIR)	OIR_SCL0_DS1	R/W	H'0000 0011	H'FCFF 7EAC	32
Vertical capture size register (OIR)	OIR_SCL0_DS2	R/W	H'0012 00F0	H'FCFF 7EB0	32
Horizontal capture size register (OIR)	OIR_SCL0_DS3	R/W	H'00F4 05A0	H'FCFF 7EB4	32
Capture control block output size register (OIR)	OIR_SCL0_DS7	R/W	H'00F0 0280	H'FCFF 7EC4	32
Scaling-up control register (OIR)	OIR_SCL0_US1	R/W	H'0000 0011	H'FCFF 7EC8	32
Output image vertical size register (OIR)	OIR_SCL0_US2	R/W	H'0023 01E0	H'FCFF 7ECC	32
Output image horizontal size register (OIR)	OIR_SCL0_US3	R/W	H'0090 0280	H'FCFF 7ED0	32
Frame buffer read select register (OIR)	OIR_SCL0_US8	R/W	H'0000 0000	H'FCFF 7EE4	32
Background color register (OIR)	OIR_SCL0_OVR1	R/W	H'0080 0080	H'FCFF 7EEC	32
SCL1 register update control register (OIR)	OIR_SCL1_UPDATE	R/WC1	H'0000 0000	H'FCFF 7F00	32
Writing mode register (OIR)	OIR_SCL1_WR1	R/W	H'0000 0000	H'FCFF 7F08	32
Write address register 1 (OIR)	OIR_SCL1_WR2	R/W	H'0000 0000	H'FCFF 7F0C	32
Write address register 2 (OIR)	OIR_SCL1_WR3	R/W	H'0800 0001	H'FCFF 7F10	32
Write address register 3 (OIR)	OIR_SCL1_WR4	R/W	H'0008 0000	H'FCFF 7F14	32



**Table 36.23 Register Configuration of the Output Image Generator (Ch0)**

Name	Abbreviation	R/W	Initial Value	Address	Access Size
Frame sub-sampling register (OIR)	OIR_SCL1_WR5	R/W	H'0000 1000	H'FCFF 7F1C	32
Bit reduction register (OIR)	OIR_SCL1_WR6	R/W	H'0000 0000	H'FCFF 7F20	32
Write detection register (OIR)	OIR_SCL1_WR7	R	H'0000 0000	H'FCFF 7F24	32
Graphics (OIR) register update control register	GR_OIR_UPDATE	R/WC1	H'0000 0000	H'FCFF 7F80	32
Frame buffer read control register (Graphics (OIR))	GR_OIR_FLM_RD	R/W	H'0000 0000	H'FCFF 7F84	32
Frame buffer control register 1 (Graphics (OIR))	GR_OIR_FLM1	R/W	H'0000 0000	H'FCFF 7F88	32
Frame buffer control register 2 (Graphics (OIR))	GR_OIR_FLM2	R/W	H'0000 0000	H'FCFF 7F8C	32
Frame buffer control register 3 (Graphics (OIR))	GR_OIR_FLM3	R/W	H'0800 0001	H'FCFF 7F90	32
Frame buffer control register 4 (Graphics (OIR))	GR_OIR_FLM4	R/W	H'0008 0000	H'FCFF 7F94	32
Frame buffer control register 5 (Graphics (OIR))	GR_OIR_FLM5	R/W	H'0000 03FF	H'FCFF 7F98	32
Frame buffer control register 6 (Graphics (OIR))	GR_OIR_FLM6	R/W	H'8000 0000	H'FCFF 7F9C	32
Alpha blending control register 1 (Graphics (OIR))	GR_OIR_AB1	R/W	H'0000 0000	H'FCFF 7FA0	32
Alpha blending control register 2 (Graphics (OIR))	GR_OIR_AB2	R/W	H'0000 0000	H'FCFF 7FA4	32
Alpha blending control register 3 (Graphics (OIR))	GR_OIR_AB3	R/W	H'0000 0000	H'FCFF 7FA8	32
Alpha blending control register 7 (Graphics (OIR))	GR_OIR_AB7	R/W	H'00FF 0000	H'FCFF 7FB8	32
Alpha blending control register 8 (Graphics (OIR))	GR_OIR_AB8	R/W	H'0000 0000	H'FCFF 7FBC	32
Alpha blending control register 9 (Graphics (OIR))	GR_OIR_AB9	R/W	H'0000 0000	H'FCFF 7FC0	32
Alpha blending control register 10 (Graphics (OIR))	GR_OIR_AB10	R/W	H'0000 0000	H'FCFF 7FC4	32
Alpha blending control register 11 (Graphics (OIR))	GR_OIR_AB11	R/W	H'0000 0000	H'FCFF 7FC8	32
Background color control register (Graphics (OIR))	GR_OIR_BASE	R/W	H'0000 8080	H'FCFF 7FCC	32
CLUT table control register (Graphics (OIR))	GR_OIR_CLUT	R/W	H'0000 0000	H'FCFF 7FD0	32
Status monitor register (Graphics (OIR))	GR_OIR_MON	R	H'0000 0000	H'FCFF 7FD4	32

**Table 36.24 Register Configuration of the Output Image Generator (Ch1)**

Name	Abbreviation	R/W	Initial Value	Address	Access Size
SCL0 register update control register (OIR)	OIR_SCL0_UPDATE	R/WC1	H'0000 0000	H'FCFF 9E80	32
Mask control register (OIR)	OIR_SCL0_FRC1	R/W	H'0AF0 0001	H'FCFF 9E84	32
Missing Vsync compensation control register (OIR)	OIR_SCL0_FRC2	R/W	H'0E10 0001	H'FCFF 9E88	32
Output sync select register (OIR)	OIR_SCL0_FRC3	R/W	H'0000 0001	H'FCFF 9E8C	32
Free-running period control register (OIR)	OIR_SCL0_FRC4	R/W	H'020C 031F	H'FCFF 9E90	32
Output delay control register (OIR)	OIR_SCL0_FRC5	R/W	H'0000 0101	H'FCFF 9E94	32
Full-screen vertical size register (OIR)	OIR_SCL0_FRC6	R/W	H'0023 01E0	H'FCFF 9E98	32
Full-screen horizontal size register (OIR)	OIR_SCL0_FRC7	R/W	H'0090 0280	H'FCFF 9E9C	32
Scaling-down control register (OIR)	OIR_SCL0_DS1	R/W	H'0000 0011	H'FCFF 9EAC	32
Vertical capture size register (OIR)	OIR_SCL0_DS2	R/W	H'0012 00F0	H'FCFF 9EB0	32
Horizontal capture size register (OIR)	OIR_SCL0_DS3	R/W	H'00F4 05A0	H'FCFF 9EB4	32
Capture control block output size register (OIR)	OIR_SCL0_DS7	R/W	H'00F0 0280	H'FCFF 9EC4	32
Scaling-up control register (OIR)	OIR_SCL0_US1	R/W	H'0000 0011	H'FCFF 9EC8	32
Output image vertical size register (OIR)	OIR_SCL0_US2	R/W	H'0023 01E0	H'FCFF 9ECC	32
Output image horizontal size register (OIR)	OIR_SCL0_US3	R/W	H'0090 0280	H'FCFF 9ED0	32
Frame buffer read select register (OIR)	OIR_SCL0_US8	R/W	H'0000 0000	H'FCFF 9EE4	32
Background color register (OIR)	OIR_SCL0_OVR1	R/W	H'0080 0080	H'FCFF 9EEC	32
SCL1 register update control register (OIR)	OIR_SCL1_UPDATE	R/WC1	H'0000 0000	H'FCFF 9F00	32
Writing mode register (OIR)	OIR_SCL1_WR1	R/W	H'0000 0000	H'FCFF 9F08	32
Write address register 1 (OIR)	OIR_SCL1_WR2	R/W	H'0000 0000	H'FCFF 9F0C	32
Write address register 2 (OIR)	OIR_SCL1_WR3	R/W	H'0800 0001	H'FCFF 9F10	32
Write address register 3 (OIR)	OIR_SCL1_WR4	R/W	H'0008 0000	H'FCFF 9F14	32
Frame sub-sampling register (OIR)	OIR_SCL1_WR5	R/W	H'0000 1000	H'FCFF 9F1C	32
Bit reduction register (OIR)	OIR_SCL1_WR6	R/W	H'0000 0000	H'FCFF 9F20	32
Write detection register (OIR)	OIR_SCL1_WR7	R	H'0000 0000	H'FCFF 9F24	32
Graphics (OIR) register update control register	GR_OIR_UPDATE	R/WC1	H'0000 0000	H'FCFF 9F80	32
Frame buffer read control register (Graphics (OIR))	GR_OIR_FLM_RD	R/W	H'0000 0000	H'FCFF 9F84	32
Frame buffer control register 1 (Graphics (OIR))	GR_OIR_FLM1	R/W	H'0000 0000	H'FCFF 9F88	32
Frame buffer control register 2 (Graphics (OIR))	GR_OIR_FLM2	R/W	H'0000 0000	H'FCFF 9F8C	32
Frame buffer control register 3 (Graphics (OIR))	GR_OIR_FLM3	R/W	H'0800 0001	H'FCFF 9F90	32

**Table 36.24 Register Configuration of the Output Image Generator (Ch1)**

Name	Abbreviation	R/W	Initial Value	Address	Access Size
Frame buffer control register 4 (Graphics (OIR))	GR_OIR_FLM4	R/W	H'0008 0000	H'FCFF 9F94	32
Frame buffer control register 5 (Graphics (OIR))	GR_OIR_FLM5	R/W	H'0000 03FF	H'FCFF 9F98	32
Frame buffer control register 6 (Graphics (OIR))	GR_OIR_FLM6	R/W	H'8000 0000	H'FCFF 9F9C	32
Alpha blending control register 1 (Graphics (OIR))	GR_OIR_AB1	R/W	H'0000 0000	H'FCFF 9FA0	32
Alpha blending control register 2 (Graphics (OIR))	GR_OIR_AB2	R/W	H'0000 0000	H'FCFF 9FA4	32
Alpha blending control register 3 (Graphics (OIR))	GR_OIR_AB3	R/W	H'0000 0000	H'FCFF 9FA8	32
Alpha blending control register 7 (Graphics (OIR))	GR_OIR_AB7	R/W	H'00FF 0000	H'FCFF 9FB8	32
Alpha blending control register 8 (Graphics (OIR))	GR_OIR_AB8	R/W	H'0000 0000	H'FCFF 9FBC	32
Alpha blending control register 9 (Graphics (OIR))	GR_OIR_AB9	R/W	H'0000 0000	H'FCFF 9FC0	32
Alpha blending control register 10 (Graphics (OIR))	GR_OIR_AB10	R/W	H'0000 0000	H'FCFF 9FC4	32
Alpha blending control register 11 (Graphics (OIR))	GR_OIR_AB11	R/W	H'0000 0000	H'FCFF 9FC8	32
Background color control register (Graphics (OIR))	GR_OIR_BASE	R/W	H'0000 8080	H'FCFF 9FCC	32
CLUT table control register (Graphics (OIR))	GR_OIR_CLUT	R/W	H'0000 0000	H'FCFF 9FD0	32
Status monitor register (Graphics (OIR))	GR_OIR_MON	R	H'0000 0000	H'FCFF 9FD4	32

## 36.2.1 SCL0 Register Update Control Register (OIR\_SCL0\_UPDATE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	OIR_SCL0_VEN_D	OIR_SCL0_VEN_C	—	—	—	OIR_SCL0_UPDATE	—	—	—	OIR_SCL0_VEN_B	—	—	—	OIR_SCL0_VEN_A
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/WC1	R/WC1	R	R	R	R/WC1	R	R	R	R/WC1	R	R	R	R/WC1

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13	OIR_SCL0_VEN_D	0	R/WC1	Frame Buffer Read Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
12	OIR_SCL0_VEN_C	0	R/WC1	Frame Buffer Read Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	OIR_SCL0_UPDATE	0	R/WC1	SYNC Control Register Update 0: Registers are not updated. 1: Registers are updated.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	OIR_SCL0_VEN_B	0	R/WC1	Synchronization Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	OIR_SCL0_VEN_A	0	R/WC1	Capture Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.

### 36.2.2 Mask Control Register (OIR\_SCL0\_FRC1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OIR_RES_VMASK[15:0]															
Initial value:	0	0	0	0	1	0	1	0	1	1	1	1	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	OIR_RES_VMASK_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	OIR_RES_VMASK[15:0]	2800	R/W	Repeated Vsync Signal Masking Period This setting is ignored.
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	OIR_RES_VMASK_ON	1	R/W	Repeated Vsync Signal Masking Control This bit should always be set to 0 when the output image generator is in use. 0: Repeated Vsync signal masking control is disabled. 1: Repeated Vsync signal masking control is enabled.

Note: This register is updated when the OIR\_SCL0\_UPDATE bit in the OIR\_SCL0 register update control register (OIR\_SCL0\_UPDATE) is 1.

### 36.2.3 Missing Vsync Compensation Control Register (OIR\_SCL0\_FRC2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OIR_RES_VLACK[15:0]															
Initial value:	0	0	0	0	1	1	1	0	0	0	0	1	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	OIR_RES_VLACK_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	OIR_RES_VLACK[15:0]	3600	R/W	Missing-Sync Compensating Pulse Output Wait Time This setting is ignored.
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	OIR_RES_VLACK_ON	1	R/W	This bit should always be set to 0 when the output image generator is in use. 0: Compensation of missing Vsync signals is disabled. 1: Compensation of missing Vsync signals is enabled.

Note: This register is updated when the OIR\_SCL0\_UPDATE bit in the OIR\_SCL0 register update control register (OIR\_SCL0\_UPDATE) is 1.

## 36.2.4 Output Sync Select Register (OIR\_SCL0\_FRC3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	OIR_RES_EN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	OIR_RES_VS_SEL
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	OIR_RES_EN	0	R/W	Enabling or Disabling the Output Image Generator 0: The input signal from the image synthesizer is directly output. 1: Signals from the graphics block (OIR) are output.
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	OIR_RES_VS_SEL	1	R/W	Vsync Signal Output Select 0: Vsync signal input from the image synthesizer 1: Internally generated free-running Vsync signal

Note: This register is updated when the OIR\_SCL0\_UPDATE bit in the OIR\_SCL0 register update control register (OIR\_SCL0\_UPDATE) is 1.

## 36.2.5 Free-Running Period Control Register (OIR\_SCL0\_FRC4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	OIR_RES_FV[10:0]										
Initial value:	0	0	0	0	0	0	1	0	0	0	0	0	1	1	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	OIR_RES_FH[10:0]										
Initial value:	0	0	0	0	0	0	1	1	0	0	0	1	1	1	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	OIR_RES_FV[10:0]	524	R/W	Free-Running Vsync Period Setting Free-running Vsync period = (OIR_RES_FV + 1) × horizontal period [usec]
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	OIR_RES_FH[10:0]	799	R/W	Hsync Period Setting Hsync period [usec] = (OIR_RES_FH + 1) ÷ pixel clock frequency [MHz]

Note: This register is updated when the OIR\_SCL0\_UPDATE bit in the OIR\_SCL0 register update control register (OIR\_SCL0\_UPDATE) is 1.

36.2.6 Output Delay Control Register (OIR\_SCL0\_FRC5)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	OIR_RES_VSDLY[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	OIR_RES_VSDLY[7:0]	1	R/W	Vsync Signal Delay Control Adjusts the Vsync signal delay in the output Hsync period units. Vsync signal delay [usec]: OIR_RES_VSDLY × output Hsync period [usec]

Note: This register is updated when the OIR\_SCL0\_VEN\_B bit in the OIR\_SCL0 register update control register (OIR\_SCL0\_UPDATE) is 1.

## 36.2.7 Full-Screen Vertical Size Register (OIR\_SCL0\_FRC6)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	OIR_RES_F_VS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	OIR_RES_F_VW[10:0]										
Initial value:	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	OIR_RES_F_VS[10:0]	35	R/W	Vertical Enable Signal Start Position for Full Screen. (VSYNC + V backporch lines) Note: The set value should be four or more (lines). OIR_RES_F_VS + OIR_RES_F_VW should be equal to or less than 2039 (lines).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	OIR_RES_F_VW[10:0]	480	R/W	Vertical Enable Signal Width for Full Screen (lines) Note: OIR_RES_F_VS + OIR_RES_F_VW should be equal to or less than 2039 (lines).

Note: This register is updated when the OIR\_SCL0\_VEN\_B bit in the OIR\_SCL0 register update control register (OIR\_SCL0\_UPDATE) is 1.



## 36.2.8 Full-Screen Horizontal Size Register (OIR\_SCL0\_FRC7)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	OIR_RES_F_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	OIR_RES_F_HW[10:0]										
Initial value:	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	OIR_RES_F_HS[10:0]	144	R/W	Horizontal Enable Signal Start Position for Full Screen. (HSYNC+H backporch pixel-clock cycles) Note: The set value should be 16 or more (clock cycles). OIR_RES_F_HS + OIR_RES_F_HW should be equal to or less than 2015 (clock cycles).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	OIR_RES_F_HW[10:0]	640	R/W	Horizontal Enable Signal Width for Full Screen (pixel-clock cycles) Note: OIR_RES_F_HS + OIR_RES_F_HW should be equal to or less than 2015 (clock cycles).

Note: This register is updated when the OIR\_SCL0\_VEN\_B bit in the OIR\_SCL0 register update control register (OIR\_SCL0\_UPDATE) is 1.

## 36.2.9 Scaling-Down Control Register (OIR\_SCL0\_DS1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	OIR_RES_DS_V_ON	—	—	—	OIR_RES_DS_H_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	OIR_RES_DS_V_ON	1	R/W	Vertical Scale Down On/Off This bit should always be set to 0 when the output image generator is in use. 0: Off 1: On
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	OIR_RES_DS_H_ON	1	R/W	Horizontal Scale Down On/Off This bit should always be set to 0 when the output image generator is in use. 0: Off 1: On

Note: This register is updated when the OIR\_SCL0\_VEN\_A bit in the OIR\_SCL0 register update control register (OIR\_SCL0\_UPDATE) is 1.

## 36.2.10 Vertical Capture Size Register (OIR\_SCL0\_DS2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	OIR_RES_VS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	OIR_RES_VW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	OIR_RES_VS [10:0]	18	R/W	Vertical Position Setting for Video Signal Capturing (VSYNC + V backporch lines) Note: The set value should be four or more (lines). OIR_RES_VS + OIR_RES_VW should be equal to or less than 2039 (lines).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	OIR_RES_VW [10:0]	240	R/W	Vertical Width of Video Signal to be Captured (Lines) Note: OIR_RES_VS + OIR_RES_VW should be equal to or less than 2039 (lines).

Note: This register is updated when the OIR\_SCL0\_VEN\_A bit in the OIR\_SCL0 register update control register (OIR\_SCL0\_UPDATE) is 1.

## 36.2.11 Horizontal Capture Size Register (OIR\_SCL0\_DS3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	OIR_RES_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	1	1	1	1	0	1	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	OIR_RES_HW[10:0]										
Initial value:	0	0	0	0	0	1	0	1	1	0	1	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	OIR_RES_HS [10:0]	244	R/W	Horizontal Position Setting for Video Signal Capturing (HSYNC + H backporch video-image clock cycles) Note: The set value should be 16 or more (clock cycles). OIR_RES_HS + OIR_RES_HW should be equal to or less than 2015 (clock cycles).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	OIR_RES_HW [10:0]	1440	R/W	Horizontal Width of Video Signal to be Captured (Video-image clock cycles) Note: OIR_RES_HS + OIR_RES_HW should be equal to or less than 2015 (clock cycles).

Note: This register is updated when the OIR\_SCL0\_VEN\_A bit in the OIR\_SCL0 register update control register (OIR\_SCL0\_UPDATE) is 1.

## 36.2.12 Scaling-Down Control Block Output Size Register (OIR\_SCL0\_DS7)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	OIR_RES_OUT_VW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	OIR_RES_OUT_HW[10:0]										
Initial value:	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	OIR_RES_OUT_VW [10:0]	240	R/W	Number of Valid Lines in Vertical Direction Output by Scaling-down Control Block (lines) This bit setting is used for the number of lines to be written to the frame buffer. When OIR_SCL1_WR1.GR_OIR_FLM_LOOP is 0 (frame write mode), specify the number of lines for one frame. When GR_OIR_FLM5.GR_OIR_FLM_LOOP is 1 (line write mode), specify the number of lines for repeated write. Note: The OIR_RES_OUT_VW value should be aligned in 4-line units and equal to or smaller than the OIR_RES_VW value.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	OIR_RES_OUT_HW [10:0]	640	R/W	Number of Valid Horizontal Pixels Output by Scaling-Down Control Block (video-image clock cycles) Note: The OIR_RES_OUT_HW value should be aligned in 4-pixel units and equal to or smaller than the OIR_RES_HW value.

Note: This register is updated when the OIR\_SCL0\_VEN\_A or OIR\_SCL0\_VEN\_C bit in the OIR\_SCL0 register update control register (OIR\_SCL0\_UPDATE) is 1.

## 36.2.13 Scaling-Up Control Register (OIR\_SCL0\_US1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	OIR_RES_US_V_ON	—	—	—	OIR_RES_US_H_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	OIR_RES_US_V_ON	1	R/W	Vertical Scale Up On/Off This bit should always be set to 0 when the output image generator is in use. 0: Off 1: On
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	OIR_RES_US_H_ON	1	R/W	Horizontal Scale Up On/Off This bit should always be set to 0 when the output image generator is in use. 0: Off 1: On

Note: This register is updated when the OIR\_SCL0\_VEN\_B bit in the OIR\_SCL0 register update control register (OIR\_SCL0\_UPDATE) is 1.

## 36.2.14 Output Image Vertical Size Register (OIR\_SCL0\_US2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	OIR_RES_P_VS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	OIR_RES_P_VW[10:0]										
Initial value:	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	OIR_RES_P_VS[10:0]	35	R/W	Vertical Enable Signal Start Position for Output Image (VSYNC + V backporch lines) Note: The set value should be four or more (lines). OIR_RES_P_VS + OIR_RES_P_VW should be equal to or less than 2039 (lines).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	OIR_RES_P_VW[10:0]	480	R/W	Vertical Enable Signal Width for Output Image (lines) Note: OIR_RES_P_VS + OIR_RES_P_VW should be equal to or less than 2039 (lines).

Note: This register is updated when the OIR\_SCL0\_VEN\_B bit in the OIR\_SCL0 register update control register (OIR\_SCL0\_UPDATE) is 1.

## 36.2.15 Output Image Horizontal Size Register (OIR\_SCL0\_US3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	OIR_RES_P_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	OIR_RES_P_HW[10:0]										
Initial value:	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	OIR_RES_P_HS[10:0]	144	R/W	Horizontal Enable Signal Start Position for Output Image (HSYNC+H backporch pixel-clock cycles) Note: The set value should be 16 or more (clock cycles). OIR_RES_P_HS + OIR_RES_P_HW should be equal to or less than 2015 (clock cycles).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	OIR_RES_P_HW[10:0]	640	R/W	Horizontal Enable Signal Width for Output Image (pixel-clock cycles) Note: OIR_RES_P_HS + OIR_RES_P_HW should be equal to or less than 2015 (clock cycles).

Note: This register is updated when the OIR\_SCL0\_VEN\_B bit in the OIR\_SCL0 register update control register (OIR\_SCL0\_UPDATE) is 1.



## 36.2.16 Frame Buffer Read Select Register (OIR\_SCL0\_US8)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	OIR_RES_	—	—	—	—
												IBUS_				
												SYNC_				
												SEL				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	OIR_RES_	0	R/W	Sync Signal Select for Frame Buffer Read Block This bit should always be set to 1 when the output image generator is in use. 0: Sync signals from the output image control block 1: Sync signals from the graphics processing block
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note: OIR\_RES\_IBUS\_SYNC\_SEL is updated when the OIR\_SCL0\_VEN\_D bit in the OIR\_SCL0 register update control register (OIR\_SCL0\_UPDATE) is 1.

## 36.2.17 Background Color Register (OIR\_SCL0\_OVR1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	OIR_RES_BK_COL_R[7:0]							
Initial value:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OIR_RES_BK_COL_G[7:0]								OIR_RES_BK_COL_B[7:0]							
Initial value:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	OIR_RES_BK_CLO_R [7:0]	128	R/W	Background Color Setting R/Cr Signal R: 8 bits; unsigned (0 to 255 [LSB]) Cr: 8 bits; 128 offset binary; unsigned (0 to 255 [LSB])
15 to 8	OIR_RES_BK_COL_G [7:0]	0	R/W	Background Color Setting G/Y Signal G/Y: 8 bits; unsigned (0 to 255 [LSB])
7 to 0	OIR_RES_BK_COL_B [7:0]	128	R/W	Background Color Setting B/Cb Signal B: 8 bits; unsigned (0 to 255 [LSB]) Cb: 8 bits; 128 offset binary; unsigned (0 to 255 [LSB])

Note: This register is updated when the OIR\_SCL0\_VEN\_B bit in the OIR\_SCL0 register update control register (OIR\_SCL0\_UPDATE) is 1.

## 36.2.18 SCL1 Register Update Control Register (OIR\_SCL1\_UPDATE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	OIR_SCL1_UPDATE_A
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	OIR_SCL1_VEN_B	—	—	—	OIR_SCL1_VEN_A
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/WC1	R	R	R	R/WC1

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	OIR_SCL1_UPDATE_A	0	R/W	Frame Buffer Write Control Register Update 0: Registers are not updated. 1: Registers are updated.
15 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	OIR_SCL1_VEN_B	0	R/WC1	Frame Buffer Write Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	OIR_SCL1_VEN_A	0	R/WC1	Frame Buffer Write Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.

## 36.2.19 Writing Mode Register (OIR\_SCL1\_WR1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	OIR_RES_WRSWA[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	OIR_RES_MD[1:0]		—	OIR_RES_BST_MD
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 19	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
18 to 16	OIR_RES_WRSWA[2:0]	All 0	R/W	8-Bit, 16-Bit, or 32-Bit Swap Setting These bits control swapping in frame buffer writing as follows. Bit 0 0: Swapped in 8-bit units. 1: Not swapped in 8-bit units. Bit 1 0: Swapped in 16-bit units. 1: Not swapped in 16-bit units. Bit 2 0: Swapped in 32-bit units. 1: Not swapped in 32-bit units. According to the setting of these bits, data is swapped as follows. Each number in parentheses ((1) to (8)) indicates 8-bit data. 000: (1) (2) (3) (4) (5) (6) (7) (8) [Not swapped] 001: (2) (1) (4) (3) (6) (5) (8) (7) [Swapped in 8-bit units] 010: (3) (4) (1) (2) (7) (8) (5) (6) [Swapped in 16-bit units] 011: (4) (3) (2) (1) (8) (7) (6) (5) [Swapped in 16-bit units + 8-bit units] 100: (5) (6) (7) (8) (1) (2) (3) (4) [Swapped in 32-bit units] 101: (6) (5) (8) (7) (2) (1) (4) (3) [Swapped in 32-bit units + 8-bit units] 110: (7) (8) (5) (6) (3) (4) (1) (2) [Swapped in 32-bit units + 16-bit units] 111: (8) (7) (6) (5) (4) (3) (2) (1) [Swapped in 32-bit units + 16-bit units + 8-bit units]
15 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3, 2	OIR_RES_MD [1:0]	0	R/W	Frame Buffer Video-Signal Writing Format 0: Setting prohibited 1: RGB565 (16 bits) 2: RGB888 (24 (32) bits) 3: Setting prohibited
1	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
0	OIR_RES_BST_MD	0	R/W	Transfer Burst Length for Frame Buffer Writing 0: 32-byte transfer (4 bursts) 1: 128-byte transfer (16 bursts)

Note: OIR\_RES\_BST\_MD is updated when the OIR\_SCL1\_VEN\_B bit in the OIR\_SCL1 register update control register (OIR\_SCL1\_UPDATE) is 1.

OIR\_RES\_DS\_WR\_MD and OIR\_RES\_MD are updated when the OIR\_SCL1\_VEN\_A or OIR\_VEN\_B bit in the OIR\_SCL1 register update control register (OIR\_SCL1\_UPDATE) is 1.

OIR\_RES\_WRSWA is updated when the OIR\_SCL1\_UPDATE\_A bit in the OIR\_SCL1 register update control register (OIR\_SCL1\_UPDATE) is 1.

### 36.2.20 Write Address Register 1 (OIR\_SCL1\_WR2)

	Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	
		OIR_RES_BASE[31:16]																
Initial value:		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
	Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
		OIR_RES_BASE[15:0]																
Initial value:		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	OIR_RES_BASE[31:0]	0	R/W	Frame Buffer Base Address Sets the start address of the frame buffer where frame data is to be stored. For 32-byte transfer: The lower five bits should be fixed to 0 0000. For 128-byte transfer: The lower seven bits should be fixed to 000 0000.

Note: This register is updated when the OIR\_SCL1\_VEN\_B bit in the OIR\_SCL1 register update control register (OIR\_SCL1\_UPDATE) is 1.

## 36.2.21 Write Address Register 2 (OIR\_SCL1\_WR3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	OIR_RES_LN_OFF[14:0]														
Initial value:	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	OIR_RES_FLM_NUM[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30 to 16	OIR_RES_LN_OFF [14:0]	2048	R/W	Frame Buffer Line Offset Address Sets the line offset address for calculating the start address of each line. Line 0: OIR_RES_BASE Line 1: OIR_RES_BASE + OIR_RES_LN_OFF × 1 : Line n: OIR_RES_BASE + OIR_RES_LN_OFF × n For 32-byte transfer: The lower five bits should be fixed to 0 0000. For 128-byte transfer: The lower seven bits should be fixed to 000 0000.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	OIR_RES_FLM_NUM [9:0]	1	R/W	Number of Frames of Buffer to be Written to Number of frames defined by OIR_RES_FLM_NUM + 1 are used. For video recording: the number of frames to be stored - 1

Note: This register is updated when the OIR\_SCL1\_VEN\_B bit in the OIR\_SCL1 register update control register (OIR\_SCL1\_UPDATE) is 1.

## 36.2.22 Write Address Register 3 (OIR\_SCL1\_WR4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	OIR_RES_FLM_OFF[22:16]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OIR_RES_FLM_OFF[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
22 to 0	OIR_RES_FLM_OFF [22:0]	524288	R/W	Frame Buffer Frame Offset Address Sets the frame offset address for calculating the start address of each frame. Buffer 0: OIR_RES_BASE Buffer 1: OIR_RES_BASE + OIR_RES_FLM_OFF × 1 : Buffer n: OIR_RES_BASE + OIR_RES_FLM_OFF × n For 32-byte transfer: The lower five bits should be fixed to 0 0000. For 128-byte transfer: The lower seven bits should be fixed to 000 0000.

Note: This register is updated when the OIR\_SCL1\_VEN\_B bit in the OIR\_SCL1 register update control register (OIR\_SCL1\_UPDATE) is 1.

## 36.2.23 Frame Sub-Sampling Register (OIR\_SCL1\_WR5)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	OIR_RES_INTER	—	—	OIR_RES_FS_RATE[1:0]	—	—	—	—	—	—	—	—	OIR_RES_WENB
Initial value:	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R/W	R/W	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	OIR_RES_INTER	1	R/W	Field Operating Mode Select This bit should always be set to 0 when the output image generator is in use. 0: Progressive 1: Interlace
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	OIR_RES_FS_RATE [1:0]	0	R/W	Writing Rate Sets the frame buffer writing rate to the vertical frequency of the input signal. 0: 1/1 an input signal (The OIR_RES_FLD_SEL setting is invalid.) 1: 1/2 an input signal 2: 1/4 an input signal 3: 1/8 an input signal
7 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	OIR_RES_WENB	0	R/W	Frame Buffer Write Enable After making the setting to enable writing, writing starts from the second frame. 0: Frame buffer writing is disabled. 1: Frame buffer writing is enabled.

Note: This register is updated when the OIR\_SCL1\_VEN\_A bit in the OIR\_SCL1 register update control register (OIR\_SCL1\_UPDATE) is 1.



## 36.2.24 Bit Reduction Register (OIR\_SCL1\_WR6)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	OIR_RES_DTH_ON	—	—	—	OIR_RES_BITDEC_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	OIR_RES_DTH_ON	0	R/W	Dither Correction On/Off 0: Off (rounded off) 1: On (2 × 2 dither pattern)
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	OIR_RES_BITDEC_ON	0	R/W	Bit Reduction On/Off 0: Off 1: On

Note: This register is updated when the OIR\_SCL1\_VEN\_A bit in the OIR\_SCL1 register update control register (OIR\_SCL1\_UPDATE) is 1.

## 36.2.25 Write Detection Register (OIR\_SCL1\_WR7)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	OIR_RES_OVERFLOW	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	—	—	—	—	—	—	OIR_RES_FLM_CNT[9:0]									—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	OIR_RES_OVERFLOW	0	R	Line Buffer Overflow Detect 1: Line buffer has overflowed. 0: Line buffer has not overflowed.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	OIR_RES_FLM_CNT[9:0]	0	R	Frame Number Before Frame Being Accessed

## 36.2.26 Graphics (OIR) Register Update Control Register (GR\_OIR\_UPDATE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	GR_OIR_UPDATE	—	—	—	GR_OIR_P_VEN	—	—	—	GR_OIR_IBUS_VEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R/WC1	R	R	R	R/WC1

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	GR_OIR_UPDATE	0	R/W	Frame Buffer Read Control Register Update 0: Registers are not updated. 1: Registers are updated.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	GR_OIR_P_VEN	0	R/WC1	Graphics Display Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR_OIR_IBUS_VEN	0	R/WC1	Frame Buffer Read Control Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.

## 36.2.27 Frame Buffer Read Control Register (Graphics (OIR)) (GR\_OIR\_FLM\_RD)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR_OIR_R_ENB
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR_OIR_R_ENB	0	R/W	Frame Buffer Read Enable 0: Frame buffer reading is disabled. 1: Frame buffer reading is enabled.

Note: This register is updated when the GR\_OIR\_IBUS\_VEN bit in the graphics (OIR) register update control register (GR\_OIR\_UPDATE) is 1.

## 36.2.28 Frame Buffer Control Register 1 (Graphics (OIR)) (GR\_OIR\_FLM1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR_OIR_LN_OFF_DIR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	GR_OIR_FLM_SEL [1:0]	—	—	—	—	GR_OIR_IMR_FLM_INV	—	—	—	GR_OIR_BST_MD
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	GR_OIR_LN_OFF_DIR	0	R/W	Selects the line offset address direction of the frame buffer. 0: Increments the address by the line offset address. 1: Decrements the address by the line offset address.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	GR_OIR_FLM_SEL [1:0]	0	R/W	Selects a frame buffer address setting signal. 0: Setting prohibited 1: Selects GR_OIR_FLM_NUM. 2: Links to distortion correction.* 3: Setting prohibited Note: * It is only available in channel 0. This setting is prohibited in channel 1.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	GR_OIR_IMR_FLM_INV	0	R/W	Sets the frame buffer number for distortion correction for display. 0: Does not replace the numbers of the frames to be read. 1: Replaces the numbers of the frames to be read.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR_OIR_BST_MD	0	R/W	Frame Buffer Burst Transfer Mode 0: 32-byte transfer 1: 128-byte transfer

Note: GR\_OIR\_LN\_OFF\_DIR, GR\_OIR\_FLM\_SEL, and GR\_OIR\_IMR\_FLM\_INV are updated when the GR\_OIR\_IBUS\_VEN bit in the graphics (OIR) register update control register (GR\_OIR\_UPDATE) is 1. GR\_OIR\_BST\_MD is updated when the GR\_OIR\_IBUS\_VEN and GR\_OIR\_P\_VEN bits in the graphics (OIR) register update control register (GR\_OIR\_UPDATE) are 1.

## 36.2.29 Frame Buffer Control Register 2 (Graphics (OIR)) (GR\_OIR\_FLM2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR_OIR_BASE[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR_OIR_BASE[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	GR_OIR_BASE[31:0]	0	R/W	Frame Buffer Base Address Sets the start address of the frame buffer where frame data is to be stored. GR_BASE[4:3] and GR_BASE[6:3] are referred to during 32-byte burst transfer and 128-byte burst transfer, respectively, to skip the start line data. The lower three bits should be fixed to 000.

Note: This register is updated when the GR\_OIR\_IBUS\_VEN or GR\_OIR\_P\_VEN bit in the graphics (OIR) register update control register (GR\_OIR\_UPDATE) is 1.

## 36.2.30 Frame Buffer Control Register 3 (Graphics (OIR)) (GR\_OIR\_FLM3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	GR_OIR_LN_OFF[14:0]														
Initial value:	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	GR_OIR_FLM_NUM[9:0]									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30 to 16	GR_OIR_LN_OFF [14:0]	2048	R/W	Frame Buffer Line Offset Address Sets the line offset address for calculating the start address of each line. Line 0: GR_OIR_BASE Line 1: GR_OIR_BASE + GR_OIR_LN_OFF × 1 : Line n: GR_OIR_BASE + GR_OIR_LN_OFF × n For 32-byte transfer: The lower five bits should be fixed to 0 0000. For 128-byte transfer: The lower seven bits should be fixed to 000 0000.
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	GR_OIR_FLM_NUM [9:0]	1	R/W	Frame Number of Frame Buffer Manually set the frame number when GR_OIR_FLM_SEL = 1.

Note: This register is updated when the GR\_OIR\_IBUS\_VEN bit in the graphics (OIR) register update control register (GR\_OIR\_UPDATE) is 1.

## 36.2.31 Frame Buffer Control Register 4 (Graphics (OIR)) (GR\_OIR\_FLM4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	GR_OIR_FLM_OFF[22:16]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR_OIR_FLM_OFF[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
22 to 0	GR_OIR_FLM_OFF[22:0]	524288	R/W	Frame Buffer Frame Offset Address Specifies the frame offset address used for calculating the start address of each frame buffer when more than one buffer is used. Buffer 0: GR_OIR_BASE Buffer 1: GR_OIR_BASE + GR_OIR_FLM_OFF × 1 : Buffer n: GR_OIR_BASE + GR_OIR_FLM_OFF × n For 32-byte transfer: The lower five bits should be fixed to 0 0000. For 128-byte transfer: The lower seven bits should be fixed to 000 0000.

Note: This register is updated when the GR\_OIR\_IBUS\_VEN bit in the graphics (OIR) register update control register (GR\_OIR\_UPDATE) is 1.

## 36.2.32 Frame Buffer Control Register 5 (Graphics (OIR)) (GR\_OIR\_FLM5)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR_OIR_FLM_LNUM[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR_OIR_FLM_LOOP[10:0]										
Initial value:	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR_OIR_FLM_LNUM[10:0]	0	R/W	Sets number of lines in a frame Number of lines is (GR_OIR_FLM_LNUM + 1).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR_OIR_FLM_LOOP[10:0]	1023	R/W	Number of lines when reading the addresses repeatedly by returning to the start address after reaching the end address. (GR_OIR_FLM_LOOP + 1) lines are read.

Note: This register is updated when the GR\_OIR\_IBUS\_VEN bit in the graphics (OIR) register update control register (GR\_OIR\_UPDATE) is 1.

## 36.2.33 Frame Buffer Control Register 6 (Graphics (OIR)) (GR\_OIR\_FLM6)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR_OIR_FORMAT[3:0]				—	GR_OIR_HW[10:0]										
Initial value:	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	GR0_RDSWA[2:0]			—	—	—	—	GR_OIR_STA_POS[5:0]					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	GR_OIR_FORMAT[3:0]	8	R/W	Sets the format of the frame buffer read signal. 0: RGB565 1: RGB888 2: $\alpha$ RGB1555 3: $\alpha$ RGB4444 4: $\alpha$ RGB8888 5: CLUT8 6: CLUT4 7: CLUT1 8: Setting prohibited 9: Setting prohibited 10: RGB $\alpha$ 5551 11: RGB $\alpha$ 8888 12 to 15: Setting prohibited
27	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
26 to 16	GR_OIR_HW[10:0]	0	R/W	Sets the width of the horizontal valid period. The width is (GR_OIR_HW + 1) pixels. Note: The set value should be equal to or more than two.
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.



Bit	Bit Name	Initial Value	R/W	Description
12 to 10	GR_OIR_RDSWA [2:0]	0	R/W	<p>8-Bit, 16-Bit, or 32-Bit Swap Setting These bits control swapping in frame buffer writing as follows.</p> <p>Bit 0 0: Swapped in 8-bit units. 1: Not swapped in 8-bit units.</p> <p>Bit 1 0: Swapped in 16-bit units. 1: Not swapped in 16-bit units.</p> <p>Bit 2 0: Swapped in 32-bit units. 1: Not swapped in 32-bit units.</p> <p>According to the setting of these bits, data is swapped as follows. Each number in parentheses ((1) to (8)) indicates 8-bit data.</p> <p>000: (1) (2) (3) (4) (5) (6) (7) (8) [Not swapped]</p> <p>001: (2) (1) (4) (3) (6) (5) (8) (7) [Swapped in 8-bit units]</p> <p>010: (3) (4) (1) (2) (7) (8) (5) (6) [Swapped in 16-bit units]</p> <p>011: (4) (3) (2) (1) (8) (7) (6) (5) [Swapped in 16-bit units + 8-bit units]</p> <p>100: (5) (6) (7) (8) (1) (2) (3) (4) [Swapped in 32-bit units]</p> <p>101: (6) (5) (8) (7) (2) (1) (4) (3) [Swapped in 32-bit units + 8-bit units]</p> <p>110: (7) (8) (5) (6) (3) (4) (1) (2) [Swapped in 32-bit units + 16-bit units]</p> <p>111: (8) (7) (6) (5) (4) (3) (2) (1) [Swapped in 32-bit units + 16-bit units + 8-bit units]</p>
9 to 6	—	All 0	R	<p>Reserved These bits are always read as 0. The write value should always be 0.</p>
5 to 0	GR_OIR_STA_POS [5:0]	0	R/W	<p>Sets the amount of data to be skipped through. Specifically data amount equal to the amount indicated by GR_OIR_STA_POS is skipped from the start of the line.</p>

Note: GR\_OIR\_STA\_POS is updated when GR\_OIR\_P\_VEN bit in the graphics (OIR) register update control register (GR\_OIR\_UPDATE) is 1.

GR\_OIR\_RDSWA is updated when GR\_OIR\_UPDATE bit in the graphics (OIR) register update control register (GR\_OIR\_UPDATE) is 1.

GR\_OIR\_FORMAT and GR\_OIR\_HW are updated when GR\_OIR\_IBUS\_VEN or GR\_OIR\_P\_VEN bit in the graphics (OIR) register update control register (GR\_OIR\_UPDATE) is 1.

## 36.2.34 Alpha Blending Control Register 1 (Graphics (OIR)) (GR\_OIR\_AB1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	GR_OIR_GRC_DISP_ON	—	—	GR_OIR_DISP_SEL [1:0]	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	GR_OIR_GRC_DISP_ON	0	R/W	Turns on/off frame-line display of the graphics image area. 0: Frame-line display off 1: Frame-line display on
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	GR_OIR_DISP_SEL [1:0]	0	R/W	Selects the graphics display mode. 0: Background color display (GR_OIR_BASE) 1: Setting prohibited 2: Current graphics display When displaying graphics, select this setting. 3: Blended display of lower-layer graphics and current graphics* Note: * Select this setting whenever chroma-key processing is to proceed. Since only current graphics are to be displayed by chroma-key processing, set the $\alpha$ values for both pixels to be subject to chroma-keying and pixels not to be subject to chroma-keying to 255.

Note: This register is updated when GR\_OIR\_P\_VEN bit in the graphics (OIR) register update control register (GR\_OIR\_UPDATE) is 1.

## 36.2.35 Alpha Blending Control Register 2 (Graphics (OIR)) (GR\_OIR\_AB2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR_OIR_GRC_VS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR_OIR_GRC_VW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR_OIR_GRC_VS[10:0]	0	R/W	Vertical Start Position of Graphics Image Area. Note: The set value should be four or more (lines). GR_OIR_GRC_VS + GR_OIR_GRC_VW should be equal to or less than 2039 (lines).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR_OIR_GRC_VW[10:0]	0	R/W	Vertical Width of Graphics Image Area.

Note: This register is updated when the GR\_OIR\_P\_VEN bit in the graphics (OIR) register update control register (GR\_OIR\_UPDATE) is 1.

## 36.2.36 Alpha Blending Control Register 3 (Graphics (OIR)) (GR\_OIR\_AB3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GR_OIR_GRC_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR_OIR_GRC_HW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	GR_OIR_GRC_HS [10:0]	0	R/W	Horizontal Start Position of Graphics Image Area. Note: The set value should be 16 or more (clock cycles). GR_OIR_GRC_HS + GR_OIR_GRC_HW should be equal to or less than 2015 (clock cycles).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR_OIR_GRC_HW [10:0]	0	R/W	Horizontal Width of Graphics Image Area. Note: For displaying an image with 1- or 2-pixel horizontal width, set GR_OIR_HW to 2 and GR_OIR_GRC_HW to 1 (1 pixel) or 2 (2 pixels).

Note: This register is updated when the GR\_OIR\_P\_VEN bit in the graphics (OIR) register update control register (GR\_OIR\_UPDATE) is 1.

## 36.2.37 Alpha Blending Control Register 7 (Graphics (OIR)) (GR\_OIR\_AB7)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR_OIR_CK_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR_OIR_CK_ON	0	R/W	CLUT-Index/RGB-Index Chroma-Key Processing On/Off 0: Off 1: On

Note: This register is updated when the GR\_OIR\_P\_VEN bit in the graphics (OIR) register update control register (GR\_OIR\_UPDATE) is 1.

## 36.2.38 Alpha Blending Control Register 8 (Graphics (OIR)) (GR\_OIR\_AB8)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR_OIR_CK_KCLUT[7:0]								GR_OIR_CK_KG[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR_OIR_CK_KB[7:0]								GR_OIR_CK_KR[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GR_OIR_CK_KCLUT[7:0]	0	R/W	CLUT Signal for CLUT-Index Chroma-Key Processing CLUT: Unsigned 8 bits (0 to 255 [LSB])
23 to 16	GR_OIR_CK_KG[7:0]	0	R/W	G Signal for RGB-Index Chroma-Key Processing G: Unsigned 8 bits (0 to 255 [LSB])
15 to 8	GR_OIR_CK_KB[7:0]	0	R/W	B Signal for RGB-Index Chroma-Key Processing B: Unsigned 8 bits (0 to 255 [LSB])
7 to 0	GR_OIR_CK_KR[7:0]	0	R/W	R Signal for RGB-Index Chroma-Key Processing R: Unsigned 8 bits (0 to 255 [LSB])

Note: This register is updated when the GR\_OIR\_P\_VEN bit in the graphics (OIR) register update control register (GR\_OIR\_UPDATE) is 1.

## 36.2.39 Alpha Blending Control Register 9 (Graphics (OIR)) (GR\_OIR\_AB9)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR_OIR_CK_A[7:0]								GR_OIR_CK_G[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR_OIR_CK_B[7:0]								GR_OIR_CK_R[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GR_OIR_CK_A[7:0]	0	R/W	Replaced Alpha Signal after RGB-Index Chroma-Key Processing $\alpha$ : Unsigned 8 bits (0 to 255 [LSB]) Note: These bits should always be set to 255 to display the current graphics only.
23 to 16	GR_OIR_CK_G[7:0]	0	R/W	Replaced G Signal after RGB-Index Chroma-Key Processing G: Unsigned 8 bits (0 to 255 [LSB])
15 to 8	GR_OIR_CK_B[7:0]	0	R/W	Replaced B Signal after RGB-Index Chroma-Key Processing B: Unsigned 8 bits (0 to 255 [LSB])
7 to 0	GR_OIR_CK_R[7:0]	0	R/W	Replaced R Signal after RGB-Index Chroma-Key Processing R: Unsigned 8 bits (0 to 255 [LSB])

Note: This register is updated when the GR\_OIR\_P\_VEN bit in the graphics (OIR) register update control register (GR\_OIR\_UPDATE) is 1.

## 36.2.40 Alpha Blending Control Register 10 (Graphics (OIR)) (GR\_OIR\_AB10)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR_OIR_A0[7:0]								GR_OIR_G0[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR_OIR_B0[7:0]								GR_OIR_R0[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GR_OIR_A0[7:0]	0	R/W	CLUT1 $\alpha$ 0 Signal Replaced with $\alpha$ signal when in the CLUT1 format and CLUT1 = 0. Replaced with $\alpha$ signal when in the $\alpha$ RGB1555/RGB $\alpha$ 5551 format and $\alpha$ = 0.  Note: These bits should always be set to 255 to display the current graphics only.
23 to 16	GR_OIR_G0[7:0]	0	R/W	CLUT1 G0 Signal Replaced with G signal when in the CLUT1 format and CLUT1 = 0.
15 to 8	GR_OIR_B0[7:0]	0	R/W	CLUT1 B0 Signal Replaced with B signal when in the CLUT1 format and CLUT1 = 0.
7 to 0	GR_OIR_R0[7:0]	0	R/W	CLUT1 R0 Signal Replaced with R signal when in the CLUT1 format and CLUT1 = 0.

Note: This register is updated when the GR\_OIR\_P\_VEN bit in the graphics (OIR) register update control register (GR\_OIR\_UPDATE) is 1.

## 36.2.41 Alpha Blending Control Register 11 (Graphics (OIR)) (GR\_OIR\_AB11)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GR_OIR_A1[7:0]								GR_OIR_G1[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR_OIR_B1[7:0]								GR_OIR_R1[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GR_OIR_A1[7:0]	0	R/W	CLUT1 $\alpha$ 1 Signal Replaced with $\alpha$ signal when in the CLUT1 format and CLUT1 = 1. Replaced with $\alpha$ signal when in the $\alpha$ RGB1555/RGB $\alpha$ 5551 format and $\alpha$ = 1.  Note: These bits should always be set to 255 to display the current graphics only.
23 to 16	GR_OIR_G1[7:0]	0	R/W	CLUT1 G1 Signal Replaced with G signal when in the CLUT1 format and CLUT1 = 1.
15 to 8	GR_OIR_B1[7:0]	0	R/W	CLUT1 B1 Signal Replaced with B signal when in the CLUT1 format and CLUT1 = 1.
7 to 0	GR_OIR_R1[7:0]	0	R/W	CLUT1 R1 Signal Replaced with R signal when in the CLUT1 format and CLUT1 = 1.

Note: This register is updated when the GR\_OIR\_P\_VEN bit in the graphics (OIR) register update control register (GR\_OIR\_UPDATE) is 1.



## 36.2.42 Background Color Control Register (Graphics (OIR)) (GR\_OIR\_BASE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	GR_OIR_BASE_G[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GR_OIR_BASE_B[7:0]								GR_OIR_BASE_R[7:0]							
Initial value:	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	GR_OIR_BASE_G [7:0]	0	R/W	Background Color G Signal G: Unsigned 8 bits (0 to 255 [LSB])
15 to 8	GR_OIR_BASE_B [7:0]	128	R/W	Background Color B Signal B: Unsigned 8 bits (0 to 255 [LSB])
7 to 0	GR_OIR_BASE_R [7:0]	128	R/W	Background Color R Signal R: Unsigned 8 bits (0 to 255 [LSB])

Note: This register is updated when the GR\_OIR\_P\_VEN bit in the graphics (OIR) register update control register (GR\_OIR\_UPDATE) is 1.

## 36.2.43 CLUT Table Control Register (Graphics (OIR)) (GR\_OIR\_CLUT)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GR_OIR_ CLT_SEL
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GR_OIR_LINE[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	GR_OIR_ CLT_SEL	0	R/W	CLUT Table Select Signal 0: Selects CLUT table 0. Referring to the CLUT table 0 value to expand to $\alpha$ RGB8888 The CPU side can read-access or write-access to the CLUT table 1. 1: Selects CLUT table 1. Referring to the CLUT table 1 value to expand to $\alpha$ RGB8888 The CPU side can read-access or write-access to the CLUT table 0.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	GR_OIR_ LINE[10:0]	0	R/W	Line interrupt Setting Outputs an interrupt signal when the number of lines matches with the value of GR_OIR_LINE. This function is enabled only when the output image generator is enabled (OIR_SCL0_FRC3.OIR_RES_EN = 1).

Note: This register is updated when the GR\_OIR\_P\_VEN bit in the graphics (OIR) register update control register (GR\_OIR\_UPDATE) is 1.

36.2.44 Status Monitor Register (GR\_OIR\_MON)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	OIR_LIN_STAT[10:0]										
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial Value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	OIR_LIN_STAT[10:0]	All 0	R	Line Position of Image Currently Being Read
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

## 37. Video Display Controller 5 (7): Output Controller

### 37.1 Output Controller

#### 37.1.1 Overview of Functions

The output controller subjects RGB signals output from the output image generator to brightness adjustment, contrast adjustment, gamma correction of individual RGB, dither process, and output format conversion. The output controller also generates various timing signals for LCD panel drive.

Output from the LVDS is also possible. For the LVDS output, see section 40, LVDS Output Interface.

Figure 37.1 shows the function block diagram of the output controller.

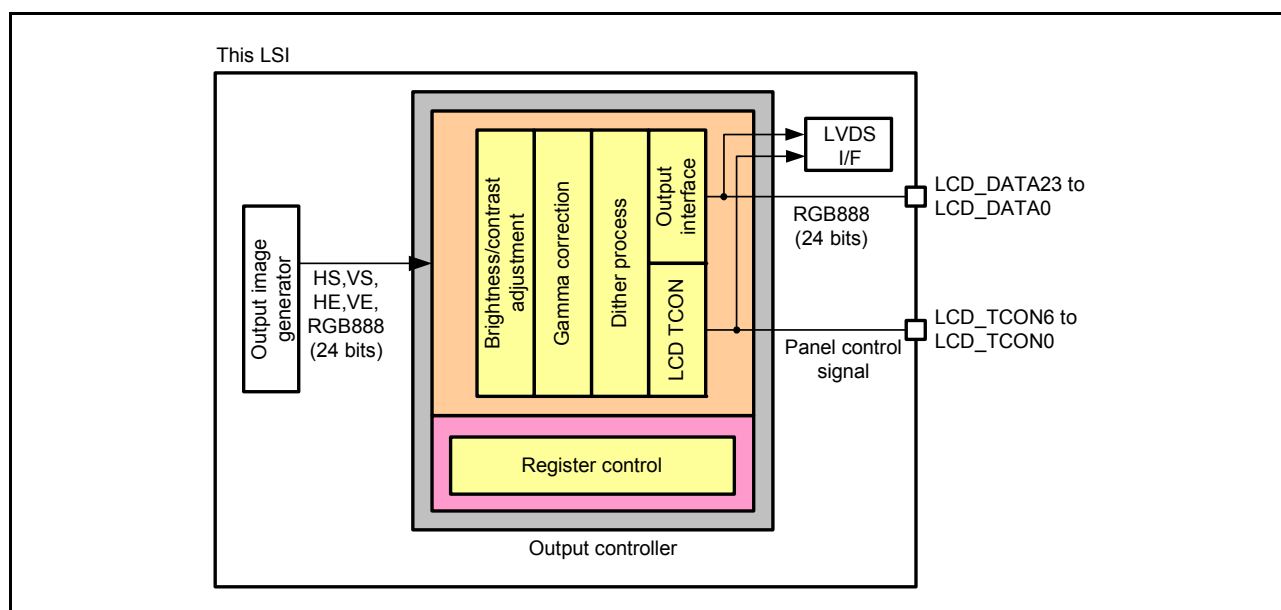


Figure 37.1 Functional Block Diagram of Output Controller

### 37.1.2 Register Update Control

The Vsync signal is used to control the update timing of all the registers of the output controller.

After 1 is set to the bits in the update control register, the contents of the relevant registers are actually modified at the rising edge of the Vsync signal, when the update control register is automatically cleared to 0.

**Table 37.1 Register Update Control**

Register Name	Bit Name	Initial Value	Description
OUT_UPDATE	OUTCNT_VEN	0	Brightness/Contrast Control, Dither Process, Output Interface Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
GAM_G_UPDATE	GAM_G_VEN	0	Gamma Correction (G) Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
GAM_B_UPDATE	GAM_B_VEN	0	Gamma Correction (B) Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
GAM_R_UPDATE	GAM_R_VEN	0	Gamma Correction (R) Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.
TCON_UPDATE	TCON_VEN	0	LCD TCON Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.

### 37.1.3 Route Selection

The processing sequence of the brightness/contrast control and gamma correction control can be swapped according to the settings of the register.

**Table 37.2 Route Selection**

Register Name	Bit Name	Initial Value	Description
OUT_CLK_PHASE	OUTCNT_ FRONT_GAM	0	Correction Circuit Sequence Control 0: Brightness → contrast → gamma correction 1: Gamma correction → brightness → contrast

### 37.1.4 Panel Brightness Adjustment

Brightness (DC) adjustment is individually performed for RGB signals from output image generator.

(BRT\_R/G/BOUT after brightness adjustment has many bits to prevent overflow or underflow. The overflow or underflow process is performed at contrast calculation.)

#### (1) Calculation formulas for brightness (DC) adjustment

$$\text{BRT\_GOUT} = \text{GIN} + \text{PBRT\_G} - 512$$

$$\text{BRT\_BOUT} = \text{BIN} + \text{PBRT\_B} - 512$$

$$\text{BRT\_ROUT} = \text{RIN} + \text{PBRT\_R} - 512$$

**Table 37.3 Brightness (DC) Adjustment**

Register Name	Bit Name	Initial Value	Description
OUT_BRIGHT1	PBRT_G[9:0]	512	Brightness (DC) Adjustment of G Signal Unsigned (0 (-512) to 512 (0) to 1023 (+511) [LSB], 512 [LSB] with offset)
OUT_BRIGHT2	PBRT_B[9:0]	512	Brightness (DC) Adjustment of B Signal Unsigned (0 (-512) to 512 (0) to 1023 (+511) [LSB], 512 [LSB] with offset)
OUT_BRIGHT2	PBRT_R[9:0]	512	Brightness (DC) Adjustment of R Signal Unsigned (0 (-512) to 512 (0) to 1023 (+511) [LSB], 512 [LSB] with offset)

### 37.1.5 Contrast Adjustment

Contrast is calculated for RGB signals obtained after brightness calculation.

(If an overflow or underflow occurs, contrast is clipped to the maximum or minimum value.)

#### (1) Calculation formulas for contrast (gain) adjustment

$$\text{GOUT} = \text{BRT\_GOUT} \times \text{CONT\_G}/128$$

$$\text{BOUT} = \text{BRT\_BOUT} \times \text{CONT\_B}/128$$

$$\text{ROUT} = \text{BRT\_ROUT} \times \text{CONT\_R}/128$$

**Table 37.4 Contrast (Gain) Adjustment**

Register Name	Bit Name	Initial Value	Description
OUT_CONTRAST	CONT_G [7:0]	128	Contrast (Gain) Adjustment of G Signal 0/128 to 255/128 (approx.2 times)
OUT_CONTRAST	CONT_B [7:0]	128	Contrast (Gain) Adjustment of B Signal 0/128 to 255/128 (approx.2 times)
OUT_CONTRAST	CONT_R [7:0]	128	Contrast (Gain) Adjustment of R Signal 0/128 to 255/128 (approx.2 times)

### 37.1.6 Gamma Correction

Gamma correction is carried out by dividing an input signal having 256 gradation levels into 32 and controlling the gain of each area. Gain coefficient of each area can be set as 0 to approx. 2.0 [times]

#### (1) Gamma correction formula for each area

$$DOUT = ((DIN - TH_{(n)}) \times GAIN_{(n)} + OFFSET_{(n)})/256$$

DIN: Input signal (8-bit)

DOUT: Output signal (10-bit)

TH(n): Threshold (8-bit)

OFFSET(n): Offset value (19-bit)

GAIN(n): Gain coefficient (11-bit)

#### (2) Offset calculation formulas for each area

$$OFFSET_{(n)} = OFFSET_{(n-1)} + DEF\_O_{(n)} \text{ (When } n = 0, \text{ } OFFSET_{(0)} = 0.)$$

$$DEF\_O_{(n)} = (TH_{(n)} - TH_{(n-1)}) \times GAIN_{(n-1)} \text{ (When } n = 0, \text{ } OFFSET_{(0)} = 0.)$$

OFFSET(n): Offset value of current area (19-bit)

OFFSET(n-1): Offset value of previous area (19-bit)

DEF\_O(n): Difference in offset value of Current and previous area (19-bit)

TH(n): Threshold of current area (8-bit)

TH(n-1): Threshold of previous area (8-bit)

GAIN(n-1): Gain coefficient of previous area (11-bit)

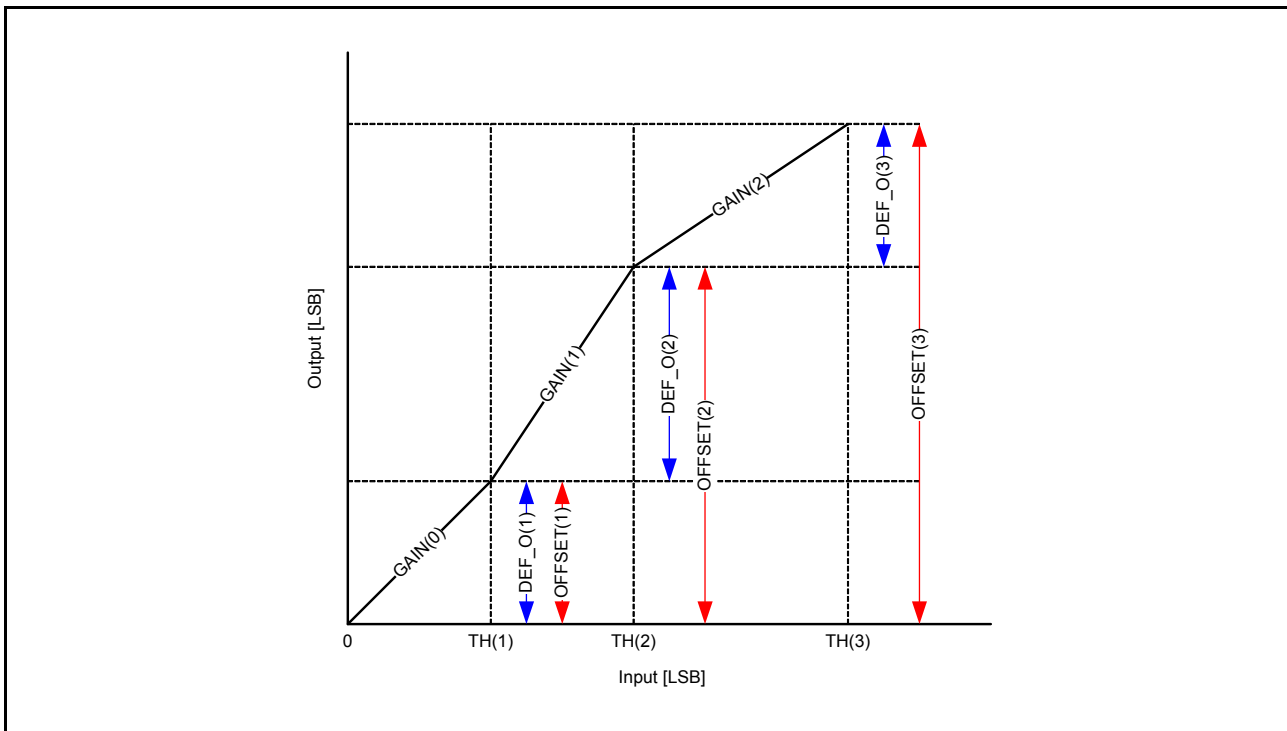


Figure 37.2 Corresponding Chart of Offset Calculation Formulas

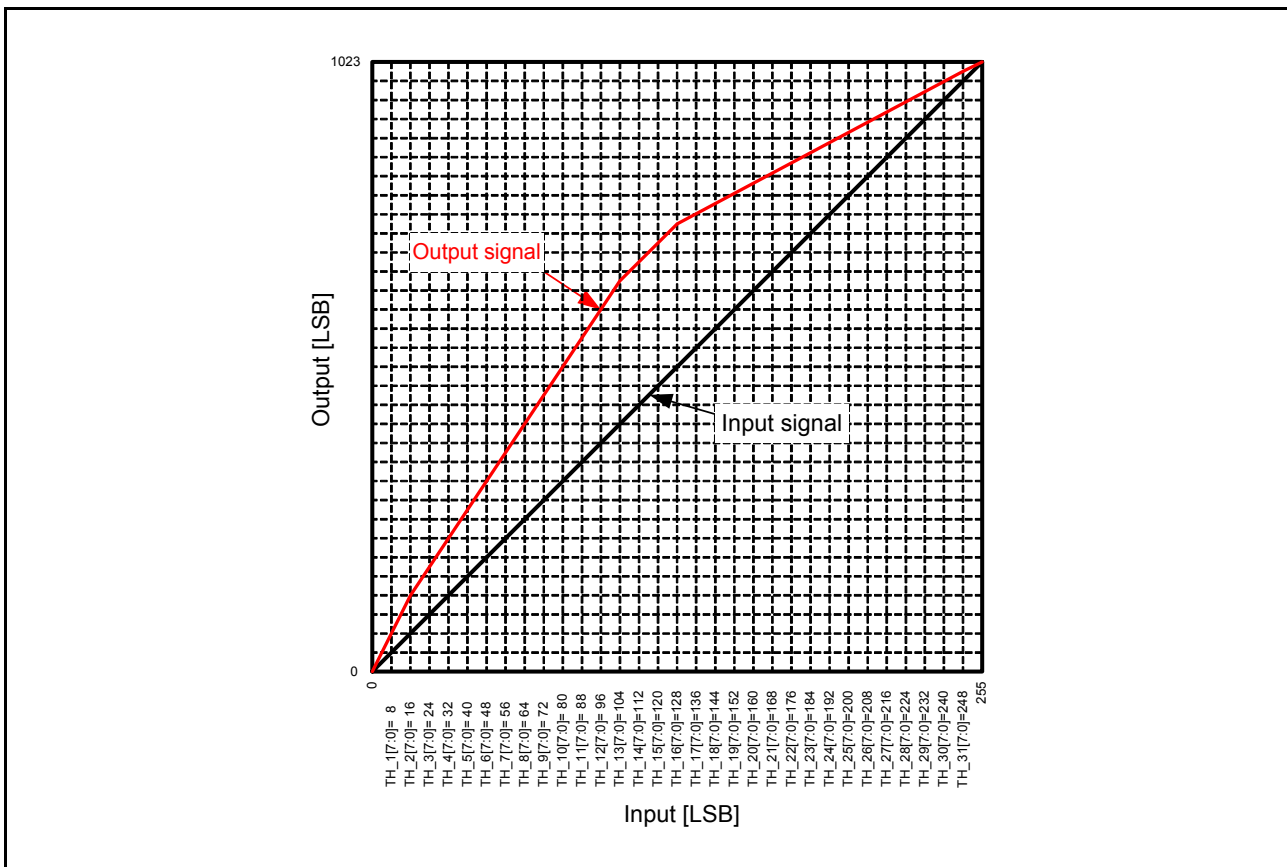


Figure 37.3 Example of Input-Output Characteristics of Gamma Correction



**Table 37.5 Gamma Correction**

Register Name	Bit Name	Initial Value	Description
GAM_SW	GAM_ON	0	Gamma Correction On/Off Control 0: Off 1: On
GAM_G_AREA1 to GAM_G_AREA8	GAM_G_TH_01 to GAM_G_TH_31 [7:0]	*	Start Threshold of Area 1 to 31 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area*1 ≤ Threshold of current area < Threshold of next area*2 *1: GAM_G_TH_01 is 0 *2: GAM_G_TH_31 is 255  *Initial Value GAM_G_TH_01:8, GAM_G_TH_02:16, GAM_G_TH_03:24, GAM_G_TH_04:32, GAM_G_TH_05:40, GAM_G_TH_06:48, GAM_G_TH_07:56, GAM_G_TH_08:64, GAM_G_TH_09:72, GAM_G_TH_10:80 GAM_G_TH_11:88, GAM_G_TH_12:96, GAM_G_TH_13:104, GAM_G_TH_14:112, GAM_G_TH_15:120, GAM_G_TH_16:128, GAM_G_TH_17:136, GAM_G_TH_18:144, GAM_G_TH_19:152, GAM_G_TH_20:160, GAM_G_TH_21:168, GAM_G_TH_22:176, GAM_G_TH_23:184, GAM_G_TH_24:192, GAM_G_TH_25:200, GAM_G_TH_26:208, GAM_G_TH_27:216, GAM_G_TH_28:224, GAM_G_TH_29:232, GAM_G_TH_30:240, GAM_G_TH_31:248
GAM_G_LUT1 to GAM_G_LUT16	GAM_G_GAIN_00 to GAM_G_GAIN_31 [10:0]	1024	Gain Adjustment of Area 0 to 31 of G Signal Unsigned (0 to 2047 [LSB], 1024 [LSB] = 1.0 [times])
GAM_B_AREA1 to GAM_B_AREA8	GAM_B_TH_01 to GAM_B_TH_31 [7:0]	*	Start Threshold of Area 1 to 31 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area*1 ≤ Threshold of current area < Threshold of next area*2 *1: GAM_B_TH_01 is 0 *2: GAM_B_TH_31 is 255  *Initial Value GAM_B_TH_01:8, GAM_B_TH_02:16, GAM_B_TH_03:24, GAM_B_TH_04:32, GAM_B_TH_05:40, GAM_B_TH_06:48, GAM_B_TH_07:56, GAM_B_TH_08:64, GAM_B_TH_09:72, GAM_B_TH_10:80 GAM_B_TH_11:88, GAM_B_TH_12:96, GAM_B_TH_13:104, GAM_B_TH_14:112, GAM_B_TH_15:120, GAM_B_TH_16:128, GAM_B_TH_17:136, GAM_B_TH_18:144, GAM_B_TH_19:152, GAM_B_TH_20:160, GAM_B_TH_21:168, GAM_B_TH_22:176, GAM_B_TH_23:184, GAM_B_TH_24:192, GAM_B_TH_25:200, GAM_B_TH_26:208, GAM_B_TH_27:216, GAM_B_TH_28:224, GAM_B_TH_29:232, GAM_B_TH_30:240, GAM_B_TH_31:248
GAM_B_LUT1 to GAM_B_LUT16	GAM_B_GAIN_00 to GAM_B_GAIN_31 [10:0]	1024	Gain Adjustment of Area 0 to 31 of B Signal Unsigned (0 to 2047 [LSB], 1024 [LSB] = 1.0 [times])

**Table 37.5 Gamma Correction**

Register Name	Bit Name	Initial Value	Description
GAM_R_AREA1 to GAM_R_AREA8	GAM_R_TH_01 to GAM_R_TH_31 [7:0]	*	Start Threshold of Area 1 to 31 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area* <sup>1</sup> ≤ Threshold of current area < Threshold of next area* <sup>2</sup> *1: GAM_R_TH_01 is 0 *2: GAM_R_TH_31 is 255  *Initial Value GAM_R_TH_01:8, GAM_R_TH_02:16, GAM_R_TH_03:24, GAM_R_TH_04:32, GAM_R_TH_05:40, GAM_R_TH_06:48, GAM_R_TH_07:56, GAM_R_TH_08:64, GAM_R_TH_09:72, GAM_R_TH_10:80 GAM_R_TH_11:88, GAM_R_TH_12:96, GAM_R_TH_13:104, GAM_R_TH_14:112, GAM_R_TH_15:120, GAM_R_TH_16:128, GAM_R_TH_17:136, GAM_R_TH_18:144, GAM_R_TH_19:152, GAM_R_TH_20:160, GAM_R_TH_21:168, GAM_R_TH_22:176, GAM_R_TH_23:184, GAM_R_TH_24:192, GAM_R_TH_25:200, GAM_R_TH_26:208, GAM_R_TH_27:216, GAM_R_TH_28:224, GAM_R_TH_29:232, GAM_R_TH_30:240, GAM_R_TH_31:248
GAM_R_LUT1 to GAM_R_LUT16	GAM_R_GAIN_00 to GAM_R_GAIN_31[10:0]	1024	Gain Adjustment of Area 0 to 31 of R Signal Unsigned (0 to 2047 [LSB], 1024 [LSB] = 1.0 [times])

### 37.1.7 Dither Process

Dither process is carried out by adjusting brightness/contrast or reducing 10-bit RGB signals output from the gamma correction block to 8-bit, 6-bit, or 5-bit RGB signals. The operation mode of dither process can be selected from truncate mode, round-off mode,  $2 \times 2$  pattern dither mode and random pattern dither mode.

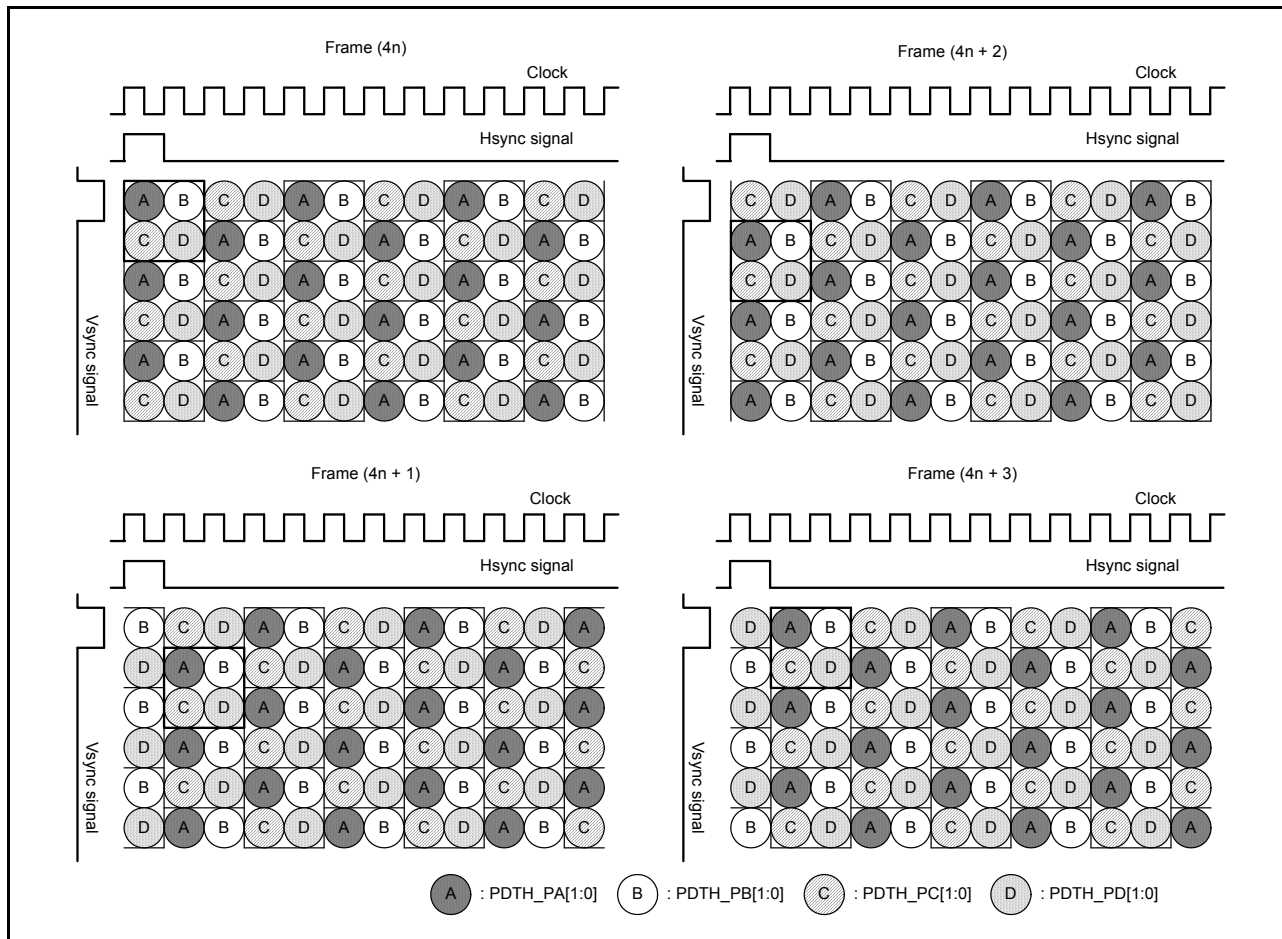


Figure 37.4 Operation Specification of  $2 \times 2$  Pattern Dither

The conversion equations are as follows.

[Truncate mode]

(a) 10 bits to 8 bits

$$\text{Output RGB data}[7:0] = \text{Input RGB data}[9:0] \div 4 \text{ (truncate the number below the decimal point)}$$

(b) 10 bits to 6 bits

$$\text{Output RGB data}[7:2] = \text{Input RGB data}[9:0] \div 16 \text{ (truncate the number below the decimal point)}$$

(c) 10 bits to 5 bits

$$\text{Output RGB data}[7:3] = \text{Input RGB data}[9:0] \div 32 \text{ (truncate the number below the decimal point)}$$

[Round-off mode]

(a) 10 bits to 8 bits

Output RGB data[7:0] = Input RGB data[9:0] ÷ 4 (round off to an integer)

(b) 10 bits to 6 bits

Output RGB data[7:2] = Input RGB data[9:0] ÷ 16 (round off to an integer)

(c) 10 bits to 5 bits

Output RGB data[7:3] = Input RGB data[9:0] ÷ 32 (round off to an integer)

[2 × 2 pattern dither mode, random pattern dither mode]

(a) 10 bits to 8 bits

Output RGB data[7:0] = Input RGB data[9:0] ÷ 4 + pattern value at the first decimal place (truncate the number below the decimal point after addition)

(b) 10 bits to 6 bits

Output RGB data[7:2] = Input RGB data[9:0] ÷ 16 + pattern value at the first decimal place (truncate the number below the decimal point after addition)

(c) 10 bits to 5 bits

Output RGB data[7:3] = Input RGB data[9:0] ÷ 32 + pattern value at the first decimal place (truncate the number below the decimal point after addition)

**Table 37.6 Panel Dither Correction**

Register Name	Bit Name	Initial Value	Description
OUT_PDTHA	PDTH_SEL[1:0]	0	Panel Dither Operation Mode 0: Truncate 1: Round-off 2: 2 × 2 pattern dither 3: Random pattern dither
OUT_PDTHA	PDTH_FORMAT[1:0]	0	Panel Dither Output Format Select 0: RGB888 1: RGB666 2: RGB565 3: Setting prohibited
OUT_PDTHA	PDTH_PA[1:0]	3	Pattern Value (A) of 2 × 2 Pattern Dither Unsigned (0 to 3 [LSB])
OUT_PDTHA	PDTH_PB[1:0]	0	Pattern Value (B) of 2 × 2 Pattern Dither Unsigned (0 to 3 [LSB])
OUT_PDTHA	PDTH_PC[1:0]	2	Pattern Value (C) of 2 × 2 Pattern Dither Unsigned (0 to 3 [LSB])
OUT_PDTHA	PDTH_PD[1:0]	1	Pattern Value (D) of 2 × 2 Pattern Dither Unsigned (0 to 3 [LSB])

### 37.1.8 Output Format Conversion

In output format conversion, the RGB signal after dither process is converted to LCD output signal having any of the following formats, namely, parallel RGB888, parallel RGB666, parallel RGB565, and serial RGB.

Further, converted data can be allocated to LCD output pins as selected.

#### (1) Bit Allocation of LCD Signals for RGB888 Output

Table 37.7 shows the RGB signal input allocated to the LCD signal output for RGB888 output.

R/G/BIN[7:0] are the RGB internal signals after dither process.

**Table 37.7 Bit Allocation of RGB Signal Input for RGB888 Output**

<b>OUT_FORMAT</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>
<b>OUT_ENDIAN_ON</b>	<b>0</b>	<b>0</b>	<b>1</b>	<b>1</b>
<b>OUT_SWAP_ON</b>	<b>0</b>	<b>1</b>	<b>0</b>	<b>1</b>
LCD_DATA23	RIN[7]	BIN[7]	RIN[0]	BIN[0]
LCD_DATA22	RIN[6]	BIN[6]	RIN[1]	BIN[1]
LCD_DATA21	RIN[5]	BIN[5]	RIN[2]	BIN[2]
LCD_DATA20	RIN[4]	BIN[4]	RIN[3]	BIN[3]
LCD_DATA19	RIN[3]	BIN[3]	RIN[4]	BIN[4]
LCD_DATA18	RIN[2]	BIN[2]	RIN[5]	BIN[5]
LCD_DATA17	RIN[1]	BIN[1]	RIN[6]	BIN[6]
LCD_DATA16	RIN[0]	BIN[0]	RIN[7]	BIN[7]
LCD_DATA15	GIN[7]	GIN[7]	GIN[0]	GIN[0]
LCD_DATA14	GIN[6]	GIN[6]	GIN[1]	GIN[1]
LCD_DATA13	GIN[5]	GIN[5]	GIN[2]	GIN[2]
LCD_DATA12	GIN[4]	GIN[4]	GIN[3]	GIN[3]
LCD_DATA11	GIN[3]	GIN[3]	GIN[4]	GIN[4]
LCD_DATA10	GIN[2]	GIN[2]	GIN[5]	GIN[5]
LCD_DATA9	GIN[1]	GIN[1]	GIN[6]	GIN[6]
LCD_DATA8	GIN[0]	GIN[0]	GIN[7]	GIN[7]
LCD_DATA7	BIN[7]	RIN[7]	BIN[0]	RIN[0]
LCD_DATA6	BIN[6]	RIN[6]	BIN[1]	RIN[1]
LCD_DATA5	BIN[5]	RIN[5]	BIN[2]	RIN[2]
LCD_DATA4	BIN[4]	RIN[4]	BIN[3]	RIN[3]
LCD_DATA3	BIN[3]	RIN[3]	BIN[4]	RIN[4]
LCD_DATA2	BIN[2]	RIN[2]	BIN[5]	RIN[5]
LCD_DATA1	BIN[1]	RIN[1]	BIN[6]	RIN[6]
LCD_DATA0	BIN[0]	RIN[0]	BIN[7]	RIN[7]

## (2) Bit Allocation of LCD Signal for RGB666 Output

Table 37.8 shows the RGB signal input allocated to the LCD signal output for RGB666 output.

R/G/BIN[7:0] are the RGB internal signals after dither process.

**Table 37.8 Bit Allocation of RGB Signal Input for RGB666 Output**

<b>OUT_FORMAT</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>1</b>
<b>OUT_ENDIAN_ON</b>	<b>0</b>	<b>0</b>	<b>1</b>	<b>1</b>
<b>OUT_SWAP_ON</b>	<b>0</b>	<b>1</b>	<b>0</b>	<b>1</b>
LCD_DATA23	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0
LCD_DATA22	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0
LCD_DATA21	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0
LCD_DATA20	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0
LCD_DATA19	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0
LCD_DATA18	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0
LCD_DATA17	RIN[7]	BIN[7]	RIN[2]	BIN[2]
LCD_DATA16	RIN[6]	BIN[6]	RIN[3]	BIN[3]
LCD_DATA15	RIN[5]	BIN[5]	RIN[4]	BIN[4]
LCD_DATA14	RIN[4]	BIN[4]	RIN[5]	BIN[5]
LCD_DATA13	RIN[3]	BIN[3]	RIN[6]	BIN[6]
LCD_DATA12	RIN[2]	BIN[2]	RIN[7]	BIN[7]
LCD_DATA11	GIN[7]	GIN[7]	GIN[2]	GIN[2]
LCD_DATA10	GIN[6]	GIN[6]	GIN[3]	GIN[3]
LCD_DATA9	GIN[5]	GIN[5]	GIN[4]	GIN[4]
LCD_DATA8	GIN[4]	GIN[4]	GIN[5]	GIN[5]
LCD_DATA7	GIN[3]	GIN[3]	GIN[6]	GIN[6]
LCD_DATA6	GIN[2]	GIN[2]	GIN[7]	GIN[7]
LCD_DATA5	BIN[7]	RIN[7]	BIN[2]	RIN[2]
LCD_DATA4	BIN[6]	RIN[6]	BIN[3]	RIN[3]
LCD_DATA3	BIN[5]	RIN[5]	BIN[4]	RIN[4]
LCD_DATA2	BIN[4]	RIN[4]	BIN[5]	RIN[5]
LCD_DATA1	BIN[3]	RIN[3]	BIN[6]	RIN[6]
LCD_DATA0	BIN[2]	RIN[2]	BIN[7]	RIN[7]

## (3) Bit Allocation of LCD Signal for RGB565 Output

Table 37.9 shows the RGB signal input allocated to the LCD signal output for RGB565 output.

R/G/BIN[7:0] are the RGB internal signals after dither process.

**Table 37.9 Bit Allocation of RGB Signal Input for RGB565 Output**

<b>OUT_FORMAT</b>	<b>2</b>	<b>2</b>	<b>2</b>	<b>2</b>
<b>OUT_ENDIAN_ON</b>	<b>0</b>	<b>0</b>	<b>1</b>	<b>1</b>
<b>OUT_SWAP_ON</b>	<b>0</b>	<b>1</b>	<b>0</b>	<b>1</b>
LCD_DATA23	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0
LCD_DATA22	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0
LCD_DATA21	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0
LCD_DATA20	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0
LCD_DATA19	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0
LCD_DATA18	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0
LCD_DATA17	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0
LCD_DATA16	Fixed to 0	Fixed to 0	Fixed to 0	Fixed to 0
LCD_DATA15	RIN[7]	BIN[7]	RIN[3]	BIN[3]
LCD_DATA14	RIN[6]	BIN[6]	RIN[4]	BIN[4]
LCD_DATA13	RIN[5]	BIN[5]	RIN[5]	BIN[5]
LCD_DATA12	RIN[4]	BIN[4]	RIN[6]	BIN[6]
LCD_DATA11	RIN[3]	BIN[3]	RIN[7]	BIN[7]
LCD_DATA10	GIN[7]	GIN[7]	GIN[2]	GIN[2]
LCD_DATA9	GIN[6]	GIN[6]	GIN[3]	GIN[3]
LCD_DATA8	GIN[5]	GIN[5]	GIN[4]	GIN[4]
LCD_DATA7	GIN[4]	GIN[4]	GIN[5]	GIN[5]
LCD_DATA6	GIN[3]	GIN[3]	GIN[6]	GIN[6]
LCD_DATA5	GIN[2]	GIN[2]	GIN[7]	GIN[7]
LCD_DATA4	BIN[7]	RIN[7]	BIN[3]	RIN[3]
LCD_DATA3	BIN[6]	RIN[6]	BIN[4]	RIN[4]
LCD_DATA2	BIN[5]	RIN[5]	BIN[5]	RIN[5]
LCD_DATA1	BIN[4]	RIN[4]	BIN[6]	RIN[6]
LCD_DATA0	BIN[3]	RIN[3]	BIN[7]	RIN[7]

#### (4) Bit Allocation of LCD Signal for Serial RGB Output

For serial RGB output, RGB signal input shown Table 37.10 is allocated to rgb internal signals and the signals are converted from parallel to serial format and output as LCD signals. R/G/BIN[7:0] are the RGB internal signals after dither process.

The internal signals r[7:0], g[7:0], and b[7:0] are serially output to LCD\_DATA7 to LCD\_DATA0.

**Table 37.10 Bit Allocation of RGB Signal Input for Serial RGB Output**

OUT_FORMAT	3	3	3	3
OUT_ENDIAN_ON	0	0	1	1
OUT_SWAP_ON	0	1	0	1
r[7]	RIN[7]	BIN[7]	RIN[0]	BIN[0]
r[6]	RIN[6]	BIN[6]	RIN[1]	BIN[1]
r[5]	RIN[5]	BIN[5]	RIN[2]	BIN[2]
r[4]	RIN[4]	BIN[4]	RIN[3]	BIN[3]
r[3]	RIN[3]	BIN[3]	RIN[4]	BIN[4]
r[2]	RIN[2]	BIN[2]	RIN[5]	BIN[5]
r[1]	RIN[1]	BIN[1]	RIN[6]	BIN[6]
r[0]	RIN[0]	BIN[0]	RIN[7]	BIN[7]
g[7]	GIN[7]	GIN[7]	GIN[0]	GIN[0]
g[6]	GIN[6]	GIN[6]	GIN[1]	GIN[1]
g[5]	GIN[5]	GIN[5]	GIN[2]	GIN[2]
g[4]	GIN[4]	GIN[4]	GIN[3]	GIN[3]
g[3]	GIN[3]	GIN[3]	GIN[4]	GIN[4]
g[2]	GIN[2]	GIN[2]	GIN[5]	GIN[5]
g[1]	GIN[1]	GIN[1]	GIN[6]	GIN[6]
g[0]	GIN[0]	GIN[0]	GIN[7]	GIN[7]
b[7]	BIN[7]	RIN[7]	BIN[0]	RIN[0]
b[6]	BIN[6]	RIN[6]	BIN[1]	RIN[1]
b[5]	BIN[5]	RIN[5]	BIN[2]	RIN[2]
b[4]	BIN[4]	RIN[4]	BIN[3]	RIN[3]
b[3]	BIN[3]	RIN[3]	BIN[4]	RIN[4]
b[2]	BIN[2]	RIN[2]	BIN[5]	RIN[5]
b[1]	BIN[1]	RIN[1]	BIN[6]	RIN[6]
b[0]	BIN[0]	RIN[0]	BIN[7]	RIN[7]



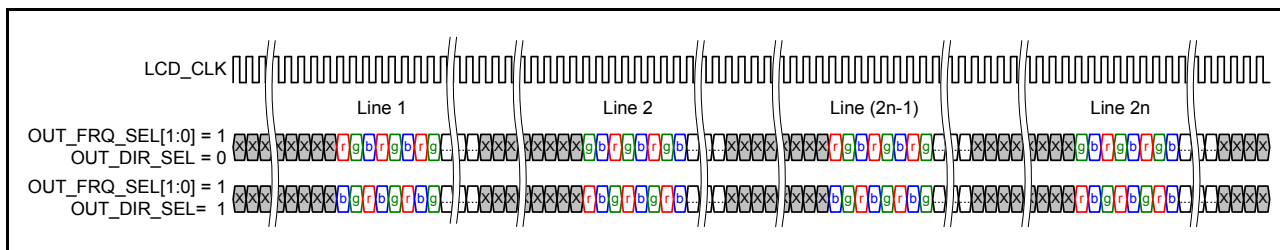
(5) Parallel to Serial Conversion

As shown in Table 37.11, four types of parallel to serial conversions are possible by controlling clock speed mode and selecting the scan direction ('n' in the table are natural numbers).

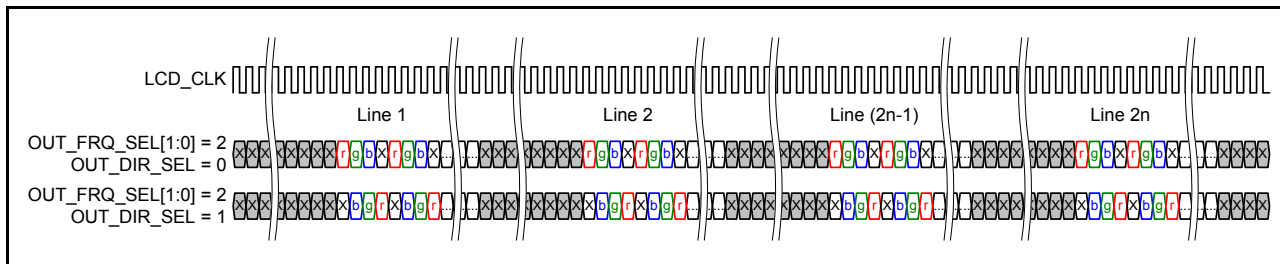
**Table 37.11 Specifications of Serial RGB Output**

OUT_FRQ_SEL	1	1	2	2
OUT_DIR_SEL	0	1	0	1
Line (2n-1)	Repeated (r → g → b)	Repeated (b → g → r)	Repeated (r → g → b → X)	Repeated (X → b → g → r)
Line 2n	Repeated (g → b → r)	Repeated (r → b → g)	Repeated (r → g → b → X)	Repeated (X → b → g → r)

Figure 37.5 and Figure 37.6 show the timing of parallel to serial conversion in triple speed and quadruple speed modes, respectively.



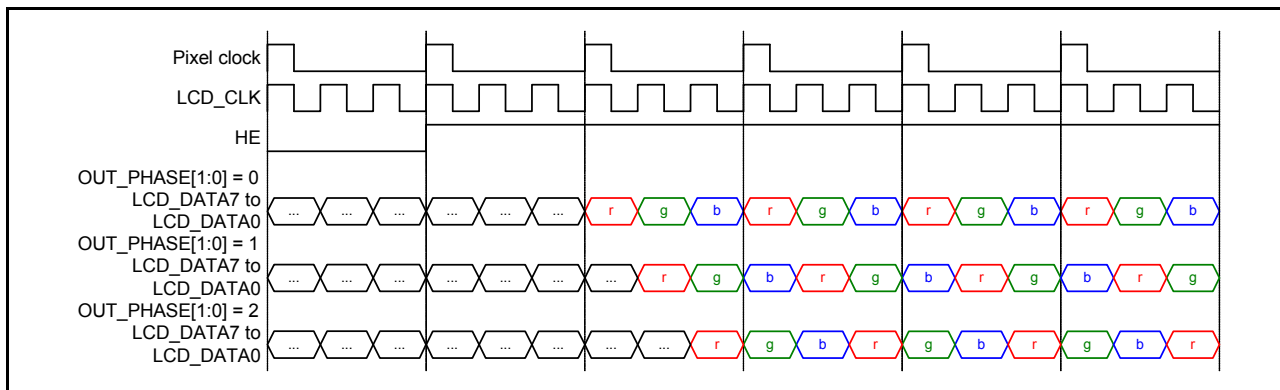
**Figure 37.5 Timing of Parallel to Serial Conversion in Triple Speed Mode**



**Figure 37.6 Timing of Parallel to Serial Conversion in Quadruple Speed Mode**

During serial output, the phase timing with the HE signal can be adjusted by OUT\_PHASE[0:1].

Figure 37.7 shows the timing of the clock phases of the serial RGB output (triple speed mode).



**Figure 37.7 Timing of Clock Phases of Serial RGB Output (Triple Speed Mode)**

Figure 37.8 shows the timing of the clock phases of the serial RGB output (quadruple speed mode).

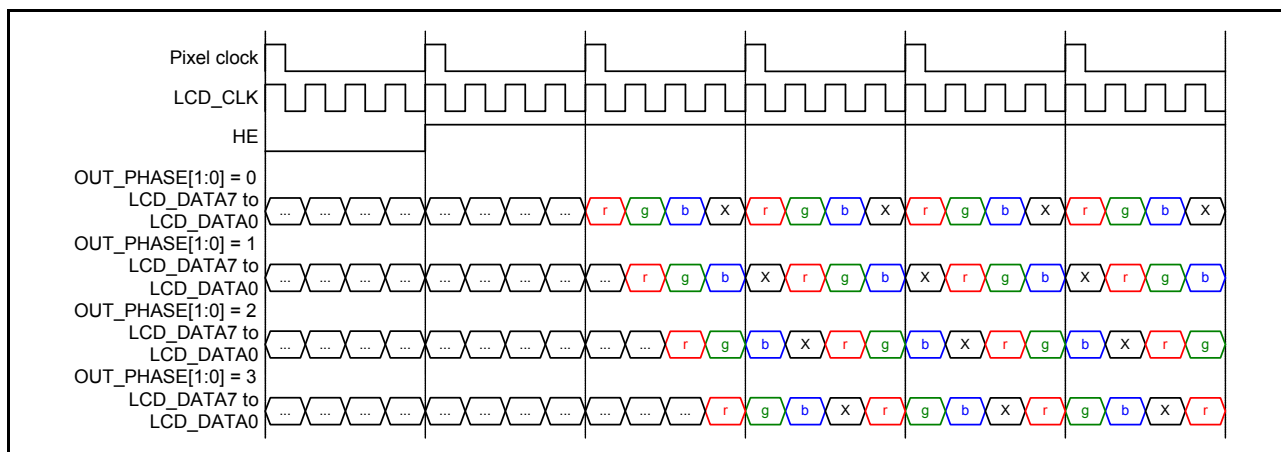


Figure 37.8 Timing of Clock Phases of Serial RGB Output (Quadruple Speed Mode)

Table 37.12 Output Format Conversion

Register Name	Bit Name	Initial Value	Description
OUT_SET	OUT_FORMAT[1:0]	0	Output Format Select 0: RGB888 1: RGB666 2: RGB565 3: Serial RGB
OUT_SET	OUT_ENDIAN_ON	0	Bit Endian Change On/Off Control 0: Off 1: On
OUT_SET	OUT_SWAP_ON	0	B/R Signal Swap On/Off Control 0: Off 1: On
OUT_SET	OUT_FRQ_SEL[1:0]	0	Clock Frequency Control 0: 100% speed — (parallel RGB) 1: Triple speed — (serial RGB) 2: Quadruple speed — (serial RGB) 3: Setting prohibited
OUT_SET	OUT_DIR_SEL	0	Scan Direction Select 0: Forward scan 1: Reverse scan
OUT_SET	OUT_PHASE[1:0]	0	Clock Phase Adjustment for Serial RGB Output Triple speed mode 0: 0 (clk) 1: 1 (clk) 2: 2 (clk) 3: Setting prohibited Quadruple speed mode 0: 0 (clk) 1: 1 (clk) 2: 2 (clk) 3: 3 (clk)

### 37.1.9 LCD TCON

The LCD TCON generates various timing signals for driving the LCD panel.

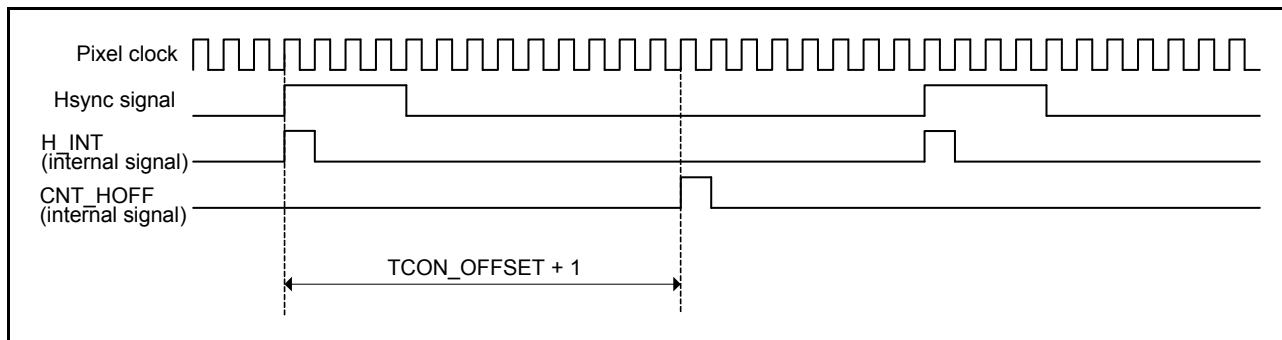
Specifically, the timing include two vertical panel driver signals, five horizontal panel driver signals, and one composite signal of the vertical and horizontal panel driver signals. Table 37.13 lists the timing signals that are generated by LCD TCON

**Table 37.13 Signals Generated by LCD TCON**

Signal Name	Type	Description
STVA/VS	Vertical	<ul style="list-style-type: none"> <li>Gate start signal The pulse width, pulse position, and pulse polarity of the signal can be controlled.</li> <li>Vsync signal The width, position, and polarity of the sync signal can be controlled.</li> </ul>
STVB/VE	Vertical	<ul style="list-style-type: none"> <li>Gate start signal The pulse width, pulse position, and pulse polarity of the signal can be controlled.</li> <li>Vertical enable signal The width, position, and polarity of the sync signal can be controlled.</li> </ul>
STH/SP/HS	Horizontal	<ul style="list-style-type: none"> <li>Source start signal The pulse width, pulse position, and pulse polarity of the signal can be controlled.</li> <li>Hsync signal The width, position, and polarity of the sync signal can be controlled.</li> </ul>
STB/LP/HE	Horizontal	<ul style="list-style-type: none"> <li>Source strobe signal The pulse width, pulse position, and pulse polarity of the signal can be controlled.</li> <li>Horizontal enable signal The width, position, and polarity of the enable signal can be controlled.</li> </ul>
CPV/GCK	Horizontal	<ul style="list-style-type: none"> <li>Gate clock signal The pulse width, pulse position, and pulse polarity of the signal can be controlled.</li> </ul>
POLA	Horizontal	<ul style="list-style-type: none"> <li>VCOM voltage polarity control signal The polarity inversion position, and polarity inversion operation (1 × 1, 1 × 2, 2 × 2) can be controlled.</li> </ul>
POLB	Horizontal	<ul style="list-style-type: none"> <li>VCOM voltage polarity control signal The polarity inversion position, and polarity inversion operation (1 × 1, 1 × 2, 2 × 2) can be controlled.</li> </ul>
DE	Horizontal/Vertical	<ul style="list-style-type: none"> <li>Data enable signal The width, position, and polarity of the enable signal can be controlled.</li> </ul>

#### (1) Horizontal Reference Offset Control

The horizontal reference offset control enables generation of a reference signal with a clock delay equivalent to the value of TCON\_OFFSET[10:0] from the rising edge of the Hsync signal. If a signal that spans across the Hsync signal needs to be generated, such a signal is generated with reference to the offset reference signal.



**Figure 37.9 Generation of Offset Horizontal Reference (H\_OFF) Signal**

**Table 37.14 Horizontal Reference Signal Selection**

Signal Name	Bit Name	Initial Value	Description
TCON_TIM	TCON_OFFSET[10:0]	0	Offset Hsync Signal Timing Sets the clock cycle count from the rising edge of the Hsync signal.
TCON_TIM_STH2	TCON_STH_HS_SEL	0	STH Signal Operating Reference Select 0: Hsync signal reference 1: Offset Hsync signal reference
TCON_TIM_STB2	TCON_STB_HS_SEL	0	STB Signal Operating Reference Select 0: Hsync signal reference 1: Offset Hsync signal reference
TCON_TIM_CPV2	TCON_CPV_HS_SEL	0	CPV Signal Operating Reference Select 0: Hsync signal reference 1: Offset Hsync signal reference
TCON_TIM_POLA2	TCON_POLA_HS_SEL	0	POLA Signal Operating Reference Select 0: Hsync signal reference 1: Offset Hsync signal reference
TCON_TIM_POLB2	TCON_POLB_HS_SEL	0	POLB Signal Operating Reference Select 0: Hsync signal reference 1: Offset Hsync signal reference

Note: When generating the POLA and POLB signals in reverse mode, the bits TCON\_POLA\_HS\_SEL and TCON\_POLB\_HS\_SEL should be set to 0.

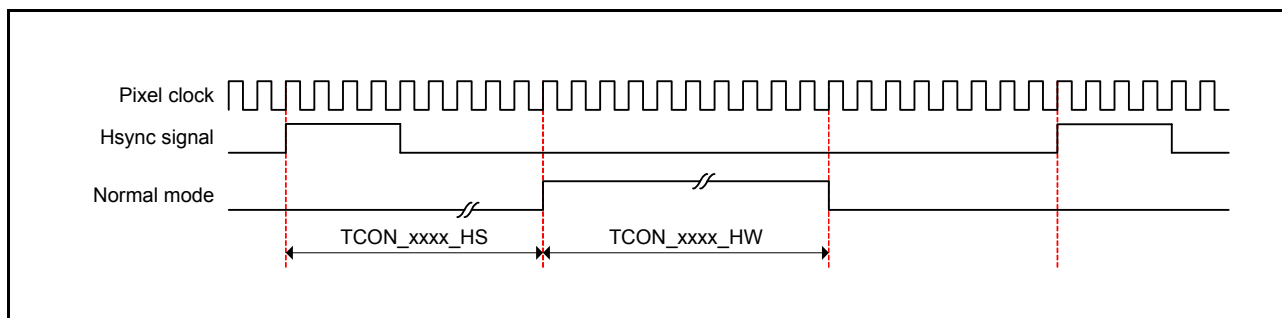
## (2) Horizontal Panel Driver Signal Generation (A)

Horizontal synchronous panel driver signal generation (A) involves generation of a timing signal that changes twice in a horizontal period according to the values of TCON\_xxxx\_HS[10:0] and TCON\_xxxx\_HW[10:0] bits, which set the first changing timing and the second changing timing, respectively.

The internal counter performs the following operations.

1. Resets the counter value at the rising edge of the Hsync signal as the reference.
2. Increments the counter value at the rising edge of the panel clock.

A fixed output value of 0 can be obtained by setting 0 in TCON\_xxxx\_HW[10:0], which set the second changing timing.



**Figure 37.10 Horizontal Panel Driver Signal (in Normal Mode)**

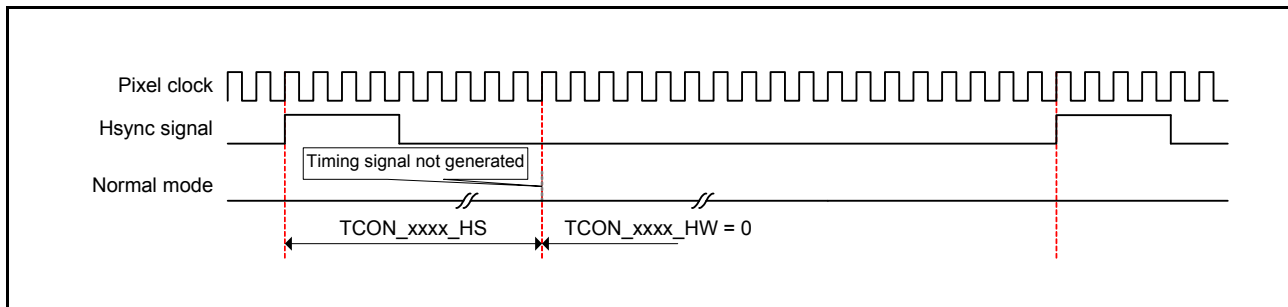


Figure 37.11 Horizontal Panel Driver Signal (in Normal Mode and When  $TCON\_xxxx\_HW\_ = 0$ )

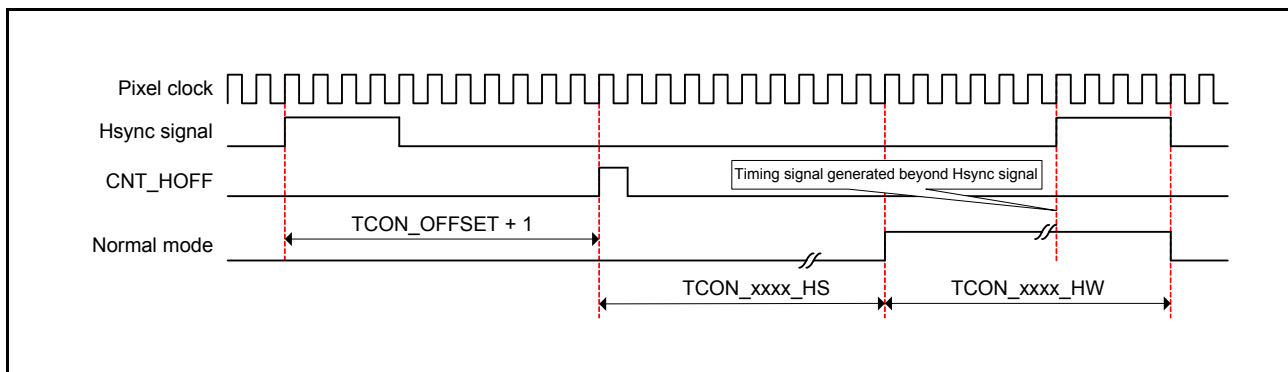


Figure 37.12 Horizontal Panel Driver Signal (in Normal Mode and When Offset Horizontal Reference is Used)

Table 37.15 Settings for Horizontal Panel Driver Signal Generation (A)

Register Name	Bit Name	Initial Value	Description
TCON_TIM_STH1	TCON_STH_HS[10:0]	0	STH Signal Pulse Start Position (First Changing Timing) Starts pulse output after the time specified by the value of TCON_STH_HS from the rising edge of the Hsync signal (clock cycles)
TCON_TIM_STH1	TCON_STH_HW[10:0]	96	STH Pulse Width (Second Changing Timing) Outputs a pulse of the duration of the value of TCON_STH_HW (clock cycles)
TCON_TIM_STB1	TCON_STB_HS[10:0]	144	STB Signal Pulse Start Position (First Changing Timing) Starts pulse output after the time specified by the value of TCON_STB_HS from the rising edge of the Hsync signal (clock cycles)
TCON_TIM_STB1	TCON_STB_HW[10:0]	640	STB Pulse Width (Second Changing Timing) Outputs a pulse of the duration of the value of TCON_STB_HW (clock cycles)
TCON_TIM_CPV1	TCON_CPV_HS[10:0]	0	CPV Signal Pulse Start Position (First Changing Timing) Starts pulse output after the time specified by the value of TCON_CPV_HS from the rising edge of the Hsync signal (clock cycles)
TCON_TIM_CPV1	TCON_CPV_HW[10:0]	0	CPV Pulse Width (Second Changing Timing) Outputs a pulse of the duration of the value of TCON_CPV_HW (clock cycles)

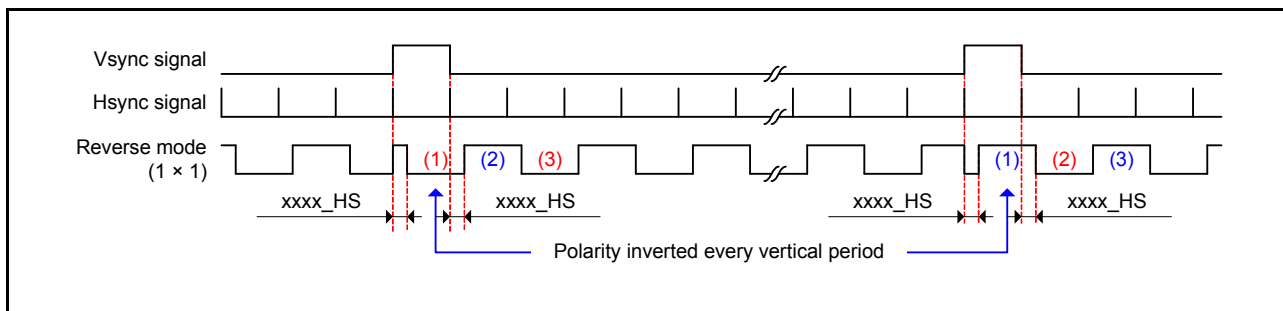
(3) Horizontal Panel Driver Signal Generation (B)

In addition to the normal mode operation described in (2), reverse mode operation, that is, horizontal panel driver signal generation (B) is provided. In reverse mode, operation starts at the rising edge of the Vsync signal as the reference and a signal is generated such that its polarity is inverted every horizontal period in the timing set by the TCON\_xxxx\_HS[10:0] bits, which set the first changing timing.

In reverse mode, regardless of whether the number of lines in the vertical direction is odd or even, the polarity of the signals generated is inverted every horizontal period. The following three reverse modes are selectable for polarity inversion operation.

**Table 37.16 Horizontal Panel Driver Signal Generation Modes**

Register Name	Bit Name	Initial Value	Description
TCON_TIM_POLA2	TCON_POLA_MD [1:0]	1	POLA Signal Generation Mode Select 0: Normal mode Generates the signal that changes twice a horizontal period. 1: 1 × 1 reverse mode Generates the signal whose polarity is inverted every horizontal period. 2: 1 × 2 reverse mode Generates the signal whose polarity is inverted in the first horizontal period and is subsequently inverted every two horizontal periods. 3: 2 × 2 reverse mode Generates the signal whose polarity is inverted every two horizontal periods.
TCON_TIM_POLB2	TCON_POLB_MD [1:0]	1	POLB Signal Generation Mode Select 0: Normal mode Generates the signal that changes twice a horizontal period. 1: 1 × 1 reverse mode Generates the signal whose polarity is inverted every horizontal period. 2: 1 × 2 reverse mode Generates the signal whose polarity is inverted in the first horizontal period and is subsequently inverted every two horizontal periods. 3: 2 × 2 reverse mode Generates the signal whose polarity is inverted every two horizontal periods.



**Figure 37.13 Horizontal Panel Driver Signal (in 1 × 1 Reverse Mode)**

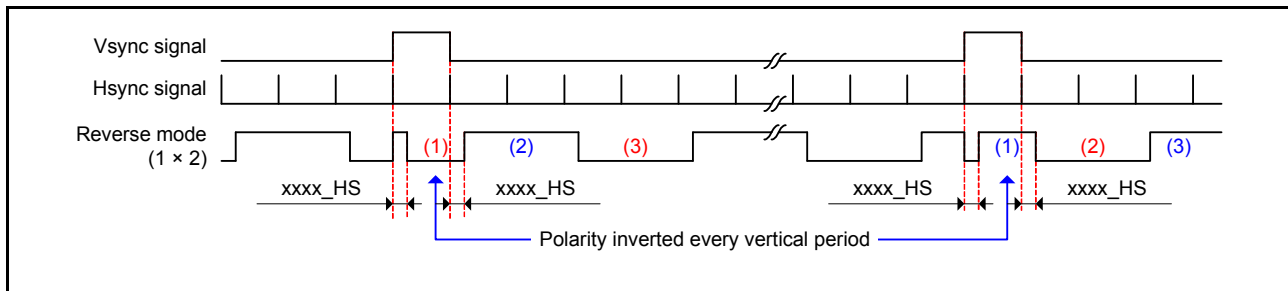


Figure 37.14 Horizontal Panel Driver Signal (in 1 × 2 Reverse Mode)

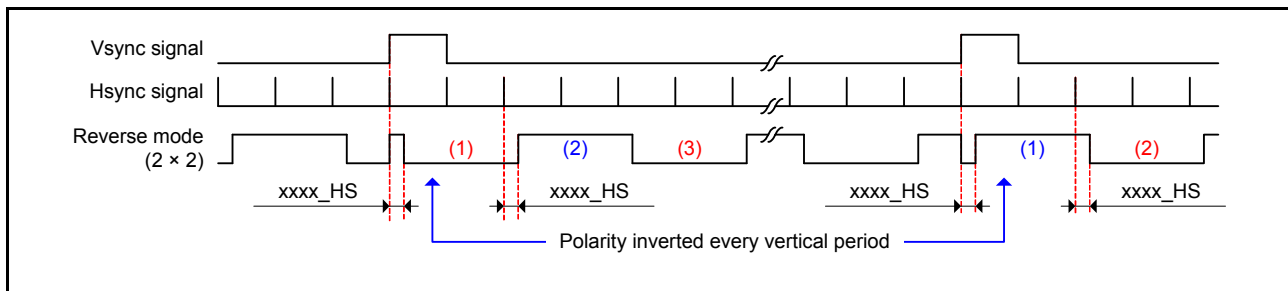


Figure 37.15 Horizontal Panel Driver Signal (in 2 × 2 Reverse Mode)

Table 37.17 Settings of Horizontal Panel Driver Signal Generation (B)

Register Name	Bit Name	Initial Value	Description
TCON_TIM_POLA1	TCON_POLA_HS [10:0]	0	POLA Signal Pulse Start Position (First Changing Timing) Starts pulse output after the time specified by the value of TCON_POLA_HS + 1 from the rising edge of the Hsync signal (clock cycles) Note: When 1 × 1, 1 × 2, or 2 × 2 reverse mode is selected, these bits should be set to 1 or greater.
TCON_TIM_POLA1	TCON_POLA_HW [10:0]	0	POLA Pulse Width (Second Changing Timing) Outputs a pulse of the duration of the value of TCON_POLA_HW (clock cycles)
TCON_TIM_POLB1	TCON_POLB_HS [10:0]	0	POLBA Signal Pulse Start Position (First Changing Timing) Starts pulse output after the time specified by the value of TCON_POLB_HS + 1 from the rising edge of the Hsync signal (clock cycles) Note: When 1 × 1, 1 × 2, or 2 × 2 reverse mode is selected, these bits should be set to 1 or greater.
TCON_TIM_POLB1	TCON_POLB_HW [10:0]	0	POLB Pulse Width (Second Changing Timing) Outputs a pulse of the duration of the value of TCON_POLB_HW (clock cycles)

#### (4) Vertical Panel Driver Signal Generation

The vertical synchronous panel driver signal generation involves the following operations.

1. Initialization at the rising edge of the Vsync signal
2. Generation of a timing signal that changes twice in a vertical period according to the values of the internal counter, and TCON\_xxxx\_VS[10:0] and TCON\_xxxx\_VW[10:0] bits, which set the first changing timing and the second changing timing, respectively.

The internal counter increments the counter value in the following two cases.

1. At the rising edge of the Hsync signal
2. At the point reached after a clock delay specified by the value of TCON\_HALF[10:0] from the rising edge of the Hsync signal (normally, 1/2fH is set).

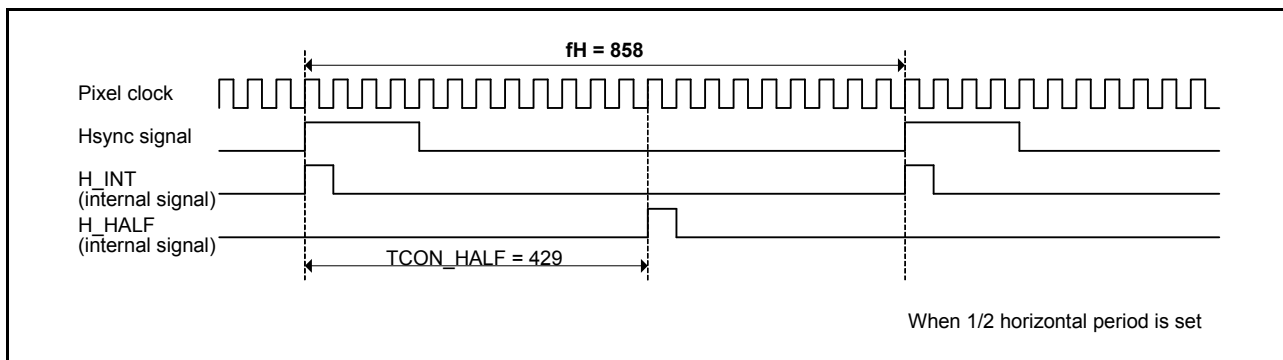


Figure 37.16 1/2 Pulse (H\_HALF) Signal Generation

Table 37.18 Settings of 1/2 Pulse (H\_HALF) Signal Generation

Register Name	Bit Name	Initial Value	Description
TCON_TIM	TCON_HALF[10:0]	400	1/2fH Timing Specifies the clock count from the rising edge of the Hsync signal as the counting timing of horizontal counter

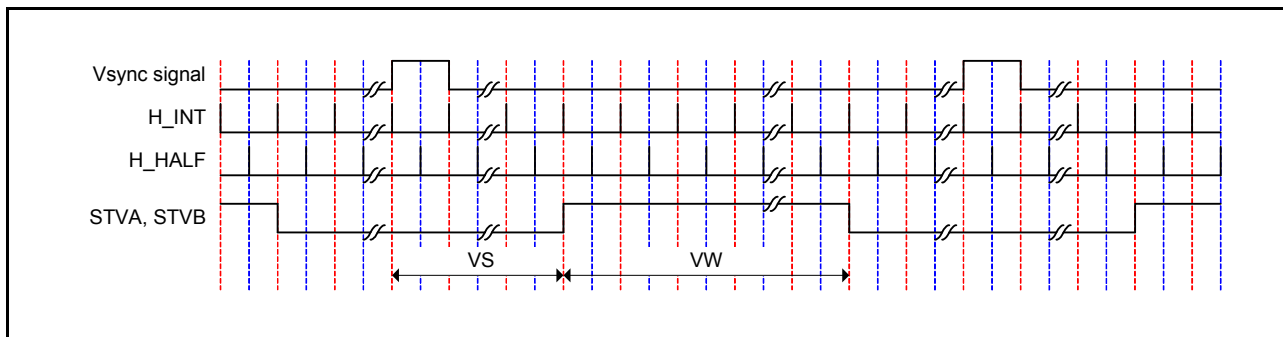


Figure 37.17 Vertical Panel Driver Signal (H\_INT Reference Operation)



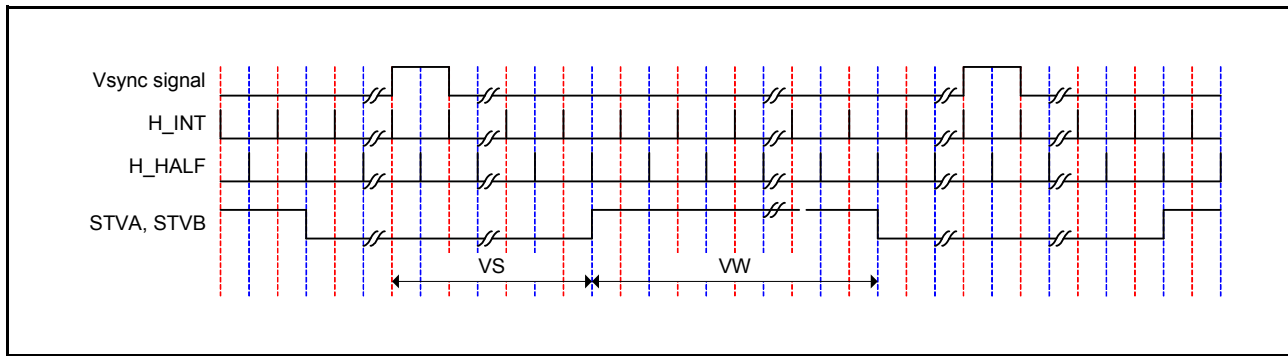


Figure 37.18 Vertical Panel Driver Signal (H\_HALF Reference Operation)

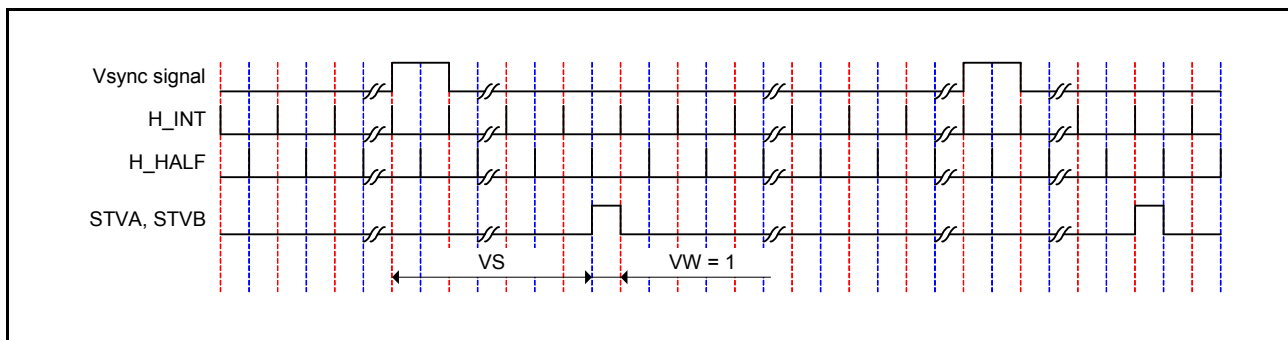


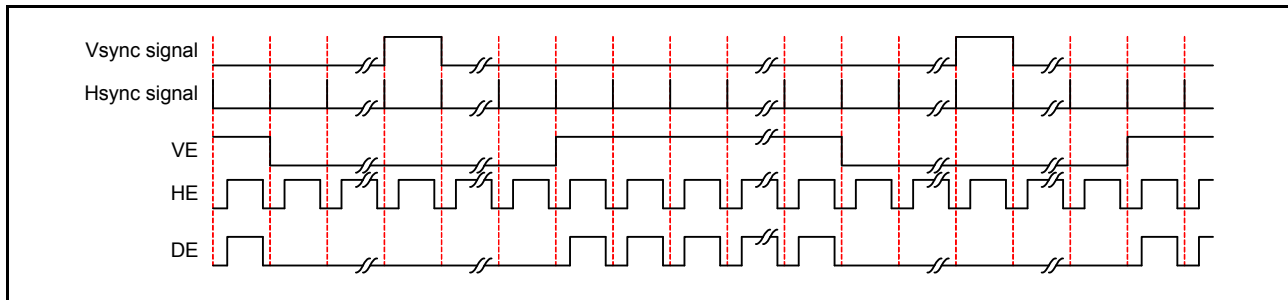
Figure 37.19 Vertical Panel Driver Signal (H\_INT and H\_HALF Reference Operation)

Table 37.19 Vertical Panel Driver Signal Generation

Register Name	Bit Name	Initial Value	Description
TCON_TIM_STVA1	TCON_STVA_VS[10:0]	0	STVA Signal Pulse Start Position (First Changing Timing) Starts pulse output after the time specified by the value of TCON_STVA_HS from the rising edge of the Vsync signal (1/2fH cycles)
TCON_TIM_STVA1	TCON_STVA_VW[10:0]	4	STVA Pulse Width (Second Changing Timing) Outputs a pulse of the duration of the value of TCON_STVA_HW (1/2fH cycles)
TCON_TIM_STVB1	TCON_STVB_VS[10:0]	70	STVB Signal Pulse Start Position (First Changing Timing) Starts pulse output after the time specified by the value of TCON_STVB_HS from the rising edge of the Vsync signal (1/2fH cycles)
TCON_TIM_STVB1	TCON_STVB_VW[10:0]	960	STVB Pulse Width (Second Changing Timing) Outputs a pulse of the duration of the value of TCON_STVB_HW (1/2fH cycles)

### (5) DE Timing Signal Generation

DE timing signal generation involves generation of data enable signal (DE) that indicates the valid period of the video signal by synthesizing the horizontal panel driver (HE) signal and the vertical panel driver (VE) signal (AND).



**Figure 37.20 Data Enable Signal Generation**

### (6) Polarity Inversion

Polarity inversion enables inversion of polarity of each signal generated by the signal generating circuit.

**Table 37.20 Panel Driver Signal Polarity Inversion Control**

Register Name	Bit Name	Initial Value	Description
TCON_TIM_STVA2	TCON_STVA_INV	1	Polarity Inversion Control of STVA Signal 0: Not inverted 1: Inverted
TCON_TIM_STVB2	TCON_STVB_INV	0	Polarity Inversion Control of STVB Signal 0: Not inverted 1: Inverted
TCON_TIM_STH2	TCON_STH_INV	1	Polarity Inversion Control of STH Signal 0: Not inverted 1: Inverted
TCON_TIM_STB2	TCON_STB_INV	0	Polarity Inversion Control of STB Signal 0: Not inverted 1: Inverted
TCON_TIM_CPV2	TCON_CPV_INV	0	Polarity Inversion Control of CPV Signal 0: Not inverted 1: Inverted
TCON_TIM_POLA2	TCON_POLA_INV	0	Polarity Inversion Control of POLA Signal 0: Not inverted 1: Inverted
TCON_TIM_POLB2	TCON_POLB_INV	0	Polarity Inversion Control of POLB Signal 0: Not inverted 1: Inverted
TCON_TIM_DE	TCON_DE_INV	0	Polarity Inversion Control of DE Signal 0: Not inverted 1: Inverted

## (7) Output Selection

An output pin is selected for every signal subjected to polarity inversion control.

**Table 37.21 Panel Driver Signal Output Selection**

Register Name	Bit Name	Initial Value	Description
TCON_TIM_STVA2	TCON_STVA_SEL [2:0]	0	Output Signal Select for LCD_TCON0 Pin 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE
TCON_TIM_STVB2	TCON_STVB_SEL [2:0]	1	Output Signal Select for LCD_TCON1 Pin 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE
TCON_TIM_STH2	TCON_STH_SEL [2:0]	2	Output Signal Select for LCD_TCON2 Pin 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE
TCON_TIM_STB2	TCON_STB_SEL [2:0]	7	Output Signal Select for LCD_TCON3 Pin 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE
TCON_TIM_CPV2	TCON_CPV_SEL [2:0]	4	Output Signal Select for LCD_TCON4 Pin 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE
TCON_TIM_POLA2	TCON_POLA_SEL [2:0]	5	Output Signal Select for LCD_TCON5 Pin 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE

**Table 37.21 Panel Driver Signal Output Selection**

Register Name	Bit Name	Initial Value	Description
TCON_TIM_POLB2	TCON_POLB_SEL [2:0]	6	Output Signal Select for LCD_TCON6 Pin 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE

**(8) Output Phase Selection**

The output phase can be individually selected for the video output signal and the various timing output signals based on the LCD\_CLK (panel clock).

**Table 37.22 Panel Output Signal Phase Selection**

Register Name	Bit Name	Initial Value	Description
OUT_CLK_PHASE	OUTCNT_LCD_EDGE	0	Output Phase Control of LCD_DATA23 to LCD_DATA0 Pin 0: Output at the rising edge of LCD_CLK pin 1: Output at the falling edge of LCD_CLK pin
OUT_CLK_PHASE	OUTCNT_STVA_EDGE	0	Output Phase Control of LCD_TCON0 Pin 0: Output at the rising edge of LCD_CLK pin 1: Output at the falling edge of LCD_CLK pin
OUT_CLK_PHASE	OUTCNT_STVB_EDGE	0	Output Phase Control of LCD_TCON1 Pin 0: Output at the rising edge of LCD_CLK pin 1: Output at the falling edge of LCD_CLK pin
OUT_CLK_PHASE	OUTCNT_STH_EDGE	0	Output Phase Control of LCD_TCON2 Pin 0: Output at the rising edge of LCD_CLK pin 1: Output at the falling edge of LCD_CLK pin
OUT_CLK_PHASE	OUTCNT_STB_EDGE	0	Output Phase Control of LCD_TCON3 Pin 0: Output at the rising edge of LCD_CLK pin 1: Output at the falling edge of LCD_CLK pin
OUT_CLK_PHASE	OUTCNT_CPV_EDGE	0	Output Phase Control of LCD_TCON4 Pin 0: Output at the rising edge of LCD_CLK pin 1: Output at the falling edge of LCD_CLK pin
OUT_CLK_PHASE	OUTCNT_POLA_EDGE	0	Output Phase Control of LCD_TCON5 Pin 0: Output at the rising edge of LCD_CLK pin 1: Output at the falling edge of LCD_CLK pin
OUT_CLK_PHASE	OUTCNT_POLB_EDGE	0	Output Phase Control of LCD_TCON6 Pin 0: Output at the rising edge of LCD_CLK pin 1: Output at the falling edge of LCD_CLK pin

## 37.2 Register Descriptions

Table 37.23 to Table 37.28 shows the register configuration.

- Symbols used in Register Description:

Initial value: Register value after a reset

—: Undefined value

R/W: Readable/writable. The written value can be read.

R/WC0: Readable/writable. Writing 0 initializes the bit. Writing 1 is ignored.

R/WC1: Readable/writable. Writing 1 initializes the bit. Writing 0 is ignored.

R: Read-only. The write value should always be 0.

—/W: Write-only. The read value is undefined.

**Table 37.23 Gamma Correction Block Register Configuration (Channel 0)**

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Register update control register G in gamma correction block	GAM_G_UPDATE	R/WC1	H'0000 0000	H'FCFF 7800	32
Function switch register in gamma correction block	GAM_SW	R/W	H'0000 0000	H'FCFF 7804	32
Table setting register G1 in gamma correction block	GAM_G_LUT1	R/W	H'0400 0400	H'FCFF 7808	32
Table setting register G2 in gamma correction block	GAM_G_LUT2	R/W	H'0400 0400	H'FCFF 780C	32
Table setting register G3 in gamma correction block	GAM_G_LUT3	R/W	H'0400 0400	H'FCFF 7810	32
Table setting register G4 in gamma correction block	GAM_G_LUT4	R/W	H'0400 0400	H'FCFF 7814	32
Table setting register G5 in gamma correction block	GAM_G_LUT5	R/W	H'0400 0400	H'FCFF 7818	32
Table setting register G6 in gamma correction block	GAM_G_LUT6	R/W	H'0400 0400	H'FCFF 781C	32
Table setting register G7 in gamma correction block	GAM_G_LUT7	R/W	H'0400 0400	H'FCFF 7820	32
Table setting register G8 in gamma correction block	GAM_G_LUT8	R/W	H'0400 0400	H'FCFF 7824	32
Table setting register G9 in gamma correction block	GAM_G_LUT9	R/W	H'0400 0400	H'FCFF 7828	32
Table setting register G10 in gamma correction block	GAM_G_LUT10	R/W	H'0400 0400	H'FCFF 782C	32
Table setting register G11 in gamma correction block	GAM_G_LUT11	R/W	H'0400 0400	H'FCFF 7830	32
Table setting register G12 in gamma correction block	GAM_G_LUT12	R/W	H'0400 0400	H'FCFF 7834	32
Table setting register G13 in gamma correction block	GAM_G_LUT13	R/W	H'0400 0400	H'FCFF 7838	32
Table setting register G14 in gamma correction block	GAM_G_LUT14	R/W	H'0400 0400	H'FCFF 783C	32
Table setting register G15 in gamma correction block	GAM_G_LUT15	R/W	H'0400 0400	H'FCFF 7840	32
Table setting register G16 in gamma correction block	GAM_G_LUT16	R/W	H'0400 0400	H'FCFF 7844	32
Area setting register G1 in gamma correction block	GAM_G_AREA1	R/W	H'0008 1018	H'FCFF 7848	32

**Table 37.23 Gamma Correction Block Register Configuration (Channel 0)**

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Area setting register G2 in gamma correction block	GAM_G_AREA2	R/W	H'2028 3038	H'FCFF 784C	32
Area setting register G3 in gamma correction block	GAM_G_AREA3	R/W	H'4048 5058	H'FCFF 7850	32
Area setting register G4 in gamma correction block	GAM_G_AREA4	R/W	H'6068 7078	H'FCFF 7854	32
Area setting register G5 in gamma correction block	GAM_G_AREA5	R/W	H'8088 9098	H'FCFF 7858	32
Area setting register G6 in gamma correction block	GAM_G_AREA6	R/W	H'A0A8 B0B8	H'FCFF 785C	32
Area setting register G7 in gamma correction block	GAM_G_AREA7	R/W	H'C0C8 D0D8	H'FCFF 7860	32
Area setting register G8 in gamma correction block	GAM_G_AREA8	R/W	H'E0E8 F0F8	H'FCFF 7864	32
Register update control register B in gamma correction block	GAM_B_UPDATE	R/WC1	H'0000 0000	H'FCFF 7880	32
Table setting register B1 in gamma correction block	GAM_B_LUT1	R/W	H'0400 0400	H'FCFF 7888	32
Table setting register B2 in gamma correction block	GAM_B_LUT2	R/W	H'0400 0400	H'FCFF 788C	32
Table setting register B3 in gamma correction block	GAM_B_LUT3	R/W	H'0400 0400	H'FCFF 7890	32
Table setting register B4 in gamma correction block	GAM_B_LUT4	R/W	H'0400 0400	H'FCFF 7894	32
Table setting register B5 in gamma correction block	GAM_B_LUT5	R/W	H'0400 0400	H'FCFF 7898	32
Table setting register B6 in gamma correction block	GAM_B_LUT6	R/W	H'0400 0400	H'FCFF 789C	32
Table setting register B7 in gamma correction block	GAM_B_LUT7	R/W	H'0400 0400	H'FCFF 78A0	32
Table setting register B8 in gamma correction block	GAM_B_LUT8	R/W	H'0400 0400	H'FCFF 78A4	32
Table setting register B9 in gamma correction block	GAM_B_LUT9	R/W	H'0400 0400	H'FCFF 78A8	32
Table setting register B10 in gamma correction block	GAM_B_LUT10	R/W	H'0400 0400	H'FCFF 78AC	32
Table setting register B11 in gamma correction block	GAM_B_LUT11	R/W	H'0400 0400	H'FCFF 78B0	32
Table setting register B12 in gamma correction block	GAM_B_LUT12	R/W	H'0400 0400	H'FCFF 78B4	32
Table setting register B13 in gamma correction block	GAM_B_LUT13	R/W	H'0400 0400	H'FCFF 78B8	32
Table setting register B14 in gamma correction block	GAM_B_LUT14	R/W	H'0400 0400	H'FCFF 78BC	32
Table setting register B15 in gamma correction block	GAM_B_LUT15	R/W	H'0400 0400	H'FCFF 78C0	32
Table setting register B16 in gamma correction block	GAM_B_LUT16	R/W	H'0400 0400	H'FCFF 78C4	32
Area setting register B1 in gamma correction block	GAM_B_AREA1	R/W	H'0008 1018	H'FCFF 78C8	32
Area setting register B2 in gamma correction block	GAM_B_AREA2	R/W	H'2028 3038	H'FCFF 78CC	32

**Table 37.23 Gamma Correction Block Register Configuration (Channel 0)**

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Area setting register B3 in gamma correction block	GAM_B_AREA3	R/W	H'4048 5058	H'FCFF 78D0	32
Area setting register B4 in gamma correction block	GAM_B_AREA4	R/W	H'6068 7078	H'FCFF 78D4	32
Area setting register B5 in gamma correction block	GAM_B_AREA5	R/W	H'8088 9098	H'FCFF 78D8	32
Area setting register B6 in gamma correction block	GAM_B_AREA6	R/W	H'A0A8 B0B8	H'FCFF 78DC	32
Area setting register B7 in gamma correction block	GAM_B_AREA7	R/W	H'C0C8 D0D8	H'FCFF 78E0	32
Area setting register B8 in gamma correction block	GAM_B_AREA8	R/W	H'E0E8 F0F8	H'FCFF 78E4	32
Register update control register R in gamma correction block	GAM_R_UPDATE	R/WC1	H'0000 0000	H'FCFF 7900	32
Table setting register R1 in gamma correction block	GAM_R_LUT1	R/W	H'0400 0400	H'FCFF 7908	32
Table setting register R2 in gamma correction block	GAM_R_LUT2	R/W	H'0400 0400	H'FCFF 790C	32
Table setting register R3 in gamma correction block	GAM_R_LUT3	R/W	H'0400 0400	H'FCFF 7910	32
Table setting register R4 in gamma correction block	GAM_R_LUT4	R/W	H'0400 0400	H'FCFF 7914	32
Table setting register R5 in gamma correction block	GAM_R_LUT5	R/W	H'0400 0400	H'FCFF 7918	32
Table setting register R6 in gamma correction block	GAM_R_LUT6	R/W	H'0400 0400	H'FCFF 791C	32
Table setting register R7 in gamma correction block	GAM_R_LUT7	R/W	H'0400 0400	H'FCFF 7920	32
Table setting register R8 in gamma correction block	GAM_R_LUT8	R/W	H'0400 0400	H'FCFF 7924	32
Table setting register R9 in gamma correction block	GAM_R_LUT9	R/W	H'0400 0400	H'FCFF 7928	32
Table setting register R10 in gamma correction block	GAM_R_LUT10	R/W	H'0400 0400	H'FCFF 792C	32
Table setting register R11 in gamma correction block	GAM_R_LUT11	R/W	H'0400 0400	H'FCFF 7930	32
Table setting register R12 in gamma correction block	GAM_R_LUT12	R/W	H'0400 0400	H'FCFF 7934	32
Table setting register R13 in gamma correction block	GAM_R_LUT13	R/W	H'0400 0400	H'FCFF 7938	32
Table setting register R14 in gamma correction block	GAM_R_LUT14	R/W	H'0400 0400	H'FCFF 793C	32
Table setting register R15 in gamma correction block	GAM_R_LUT15	R/W	H'0400 0400	H'FCFF 7940	32
Table setting register R16 in gamma correction block	GAM_R_LUT16	R/W	H'0400 0400	H'FCFF 7944	32
Area setting register R1 in gamma correction block	GAM_R_AREA1	R/W	H'0008 1018	H'FCFF 7948	32
Area setting register R2 in gamma correction block	GAM_R_AREA2	R/W	H'2028 3038	H'FCFF 794C	32
Area setting register R3 in gamma correction block	GAM_R_AREA3	R/W	H'4048 5058	H'FCFF 7950	32

**Table 37.23 Gamma Correction Block Register Configuration (Channel 0)**

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Area setting register R4 in gamma correction block	GAM_R_AREA4	R/W	H'6068 7078	H'FCFF 7954	32
Area setting register R5 in gamma correction block	GAM_R_AREA5	R/W	H'8088 9098	H'FCFF 7958	32
Area setting register R6 in gamma correction block	GAM_R_AREA6	R/W	H'A0A8 B0B8	H'FCFF 795C	32
Area setting register R7 in gamma correction block	GAM_R_AREA7	R/W	H'C0C8 D0D8	H'FCFF 7960	32
Area setting register R8 in gamma correction block	GAM_R_AREA8	R/W	H'E0E8 F0F8	H'FCFF 7964	32

**Table 37.24 TCON Block Register Configuration (Channel 0)**

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
TCON register update control register	TCON_UPDATE	R/WC1	H'0000 0000	H'FCFF 7980	32
TCON reference timing setting register	TCON_TIM	R/W	H'0190 0000	H'FCFF 7984	32
TCON vertical timing setting register A1	TCON_TIM_STVA1	R/W	H'0000 0004	H'FCFF 7988	32
TCON vertical timing setting register A2	TCON_TIM_STVA2	R/W	H'0000 0010	H'FCFF 798C	32
TCON vertical timing setting register B1	TCON_TIM_STVB1	R/W	H'0046 03C0	H'FCFF 7990	32
TCON vertical timing setting register B2	TCON_TIM_STVB2	R/W	H'0000 0001	H'FCFF 7994	32
TCON horizontal timing setting register STH1	TCON_TIM_STH1	R/W	H'0000 0060	H'FCFF 7998	32
TCON horizontal timing setting register STH2	TCON_TIM_STH2	R/W	H'0000 0012	H'FCFF 799C	32
TCON horizontal timing setting register STB1	TCON_TIM_STB1	R/W	H'0090 0280	H'FCFF 79A0	32
TCON horizontal timing setting register STB2	TCON_TIM_STB2	R/W	H'0000 0007	H'FCFF 79A4	32
TCON horizontal timing setting register CPV1	TCON_TIM_CPV1	R/W	H'0000 0000	H'FCFF 79A8	32
TCON horizontal timing setting register CPV2	TCON_TIM_CPV2	R/W	H'0000 0004	H'FCFF 79AC	32
TCON horizontal timing setting register POLA1	TCON_TIM_POLA1	R/W	H'0000 0000	H'FCFF 79B0	32
TCON horizontal timing setting register POLA2	TCON_TIM_POLA2	R/W	H'0000 1005	H'FCFF 79B4	32
TCON horizontal timing setting register POLB1	TCON_TIM_POLB1	R/W	H'0000 0000	H'FCFF 79B8	32
TCON horizontal timing setting register POLB2	TCON_TIM_POLB2	R/W	H'0000 1006	H'FCFF 79BC	32
TCON data enable polarity setting register	TCON_TIM_DE	R/W	H'0000 0000	H'FCFF 79C0	32



**Table 37.25 Output Controller Register Configuration (Channel 0)**

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Register update control register in output controller	OUT_UPDATE	R/WC1	H'0000 0000	H'FCFF 7A00	32
Output interface register	OUT_SET	R/W	H'001F 0000	H'FCFF 7A04	32
Brightness (DC) correction register 1	OUT_BRIGHT1	R/W	H'0000 0200	H'FCFF 7A08	32
Brightness (DC) correction register 2	OUT_BRIGHT2	R/W	H'0200 0200	H'FCFF 7A0C	32
Contrast (gain) correction register	OUT_CONTRAST	R/W	H'0080 8080	H'FCFF 7A10	32
Panel dither register	OUT_PDTHA	R/W	H'0000 3021	H'FCFF 7A14	32
Output phase control register	OUT_CLK_PHASE	R/W	H'0000 0000	H'FCFF 7A24	32

**Table 37.26 Gamma Correction Block Register Configuration (Channel 1)**

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Register update control register G in gamma correction block	GAM_G_UPDATE	R/WC1	H'0000 0000	H'FCFF9800	32
Function switch register in gamma correction block	GAM_SW	R/W	H'0000 0000	H'FCFF9804	32
Table setting register G1 in gamma correction block	GAM_G_LUT1	R/W	H'0400 0400	H'FCFF9808	32
Table setting register G2 in gamma correction block	GAM_G_LUT2	R/W	H'0400 0400	H'FCFF980C	32
Table setting register G3 in gamma correction block	GAM_G_LUT3	R/W	H'0400 0400	H'FCFF9810	32
Table setting register G4 in gamma correction block	GAM_G_LUT4	R/W	H'0400 0400	H'FCFF9814	32
Table setting register G5 in gamma correction block	GAM_G_LUT5	R/W	H'0400 0400	H'FCFF9818	32
Table setting register G6 in gamma correction block	GAM_G_LUT6	R/W	H'0400 0400	H'FCFF981C	32
Table setting register G7 in gamma correction block	GAM_G_LUT7	R/W	H'0400 0400	H'FCFF9820	32
Table setting register G8 in gamma correction block	GAM_G_LUT8	R/W	H'0400 0400	H'FCFF9824	32
Table setting register G9 in gamma correction block	GAM_G_LUT9	R/W	H'0400 0400	H'FCFF9828	32
Table setting register G10 in gamma correction block	GAM_G_LUT10	R/W	H'0400 0400	H'FCFF982C	32
Table setting register G11 in gamma correction block	GAM_G_LUT11	R/W	H'0400 0400	H'FCFF9830	32
Table setting register G12 in gamma correction block	GAM_G_LUT12	R/W	H'0400 0400	H'FCFF9834	32
Table setting register G13 in gamma correction block	GAM_G_LUT13	R/W	H'0400 0400	H'FCFF9838	32
Table setting register G14 in gamma correction block	GAM_G_LUT14	R/W	H'0400 0400	H'FCFF983C	32
Table setting register G15 in gamma correction block	GAM_G_LUT15	R/W	H'0400 0400	H'FCFF9840	32

**Table 37.26 Gamma Correction Block Register Configuration (Channel 1)**

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Table setting register G16 in gamma correction block	GAM_G_LUT16	R/W	H'0400 0400	H'FCFF9844	32
Area setting register G1 in gamma correction block	GAM_G_AREA1	R/W	H'0008 1018	H'FCFF9848	32
Area setting register G2 in gamma correction block	GAM_G_AREA2	R/W	H'2028 3038	H'FCFF984C	32
Area setting register G3 in gamma correction block	GAM_G_AREA3	R/W	H'4048 5058	H'FCFF9850	32
Area setting register G4 in gamma correction block	GAM_G_AREA4	R/W	H'6068 7078	H'FCFF9854	32
Area setting register G5 in gamma correction block	GAM_G_AREA5	R/W	H'8088 9098	H'FCFF9858	32
Area setting register G6 in gamma correction block	GAM_G_AREA6	R/W	H'A0A8 B0B8	H'FCFF985C	32
Area setting register G7 in gamma correction block	GAM_G_AREA7	R/W	H'C0C8 D0D8	H'FCFF9860	32
Area setting register G8 in gamma correction block	GAM_G_AREA8	R/W	H'E0E8 F0F8	H'FCFF9864	32
Register update control register B in gamma correction block	GAM_B_UPDATE	R/WC1	H'0000 0000	H'FCFF9880	32
Table setting register B1 in gamma correction block	GAM_B_LUT1	R/W	H'0400 0400	H'FCFF9888	32
Table setting register B2 in gamma correction block	GAM_B_LUT2	R/W	H'0400 0400	H'FCFF988C	32
Table setting register B3 in gamma correction block	GAM_B_LUT3	R/W	H'0400 0400	H'FCFF9890	32
Table setting register B4 in gamma correction block	GAM_B_LUT4	R/W	H'0400 0400	H'FCFF9894	32
Table setting register B5 in gamma correction block	GAM_B_LUT5	R/W	H'0400 0400	H'FCFF9898	32
Table setting register B6 in gamma correction block	GAM_B_LUT6	R/W	H'0400 0400	H'FCFF989C	32
Table setting register B7 in gamma correction block	GAM_B_LUT7	R/W	H'0400 0400	H'FCFF98A0	32
Table setting register B8 in gamma correction block	GAM_B_LUT8	R/W	H'0400 0400	H'FCFF98A4	32
Table setting register B9 in gamma correction block	GAM_B_LUT9	R/W	H'0400 0400	H'FCFF98A8	32
Table setting register B10 in gamma correction block	GAM_B_LUT10	R/W	H'0400 0400	H'FCFF98AC	32
Table setting register B11 in gamma correction block	GAM_B_LUT11	R/W	H'0400 0400	H'FCFF98B0	32
Table setting register B12 in gamma correction block	GAM_B_LUT12	R/W	H'0400 0400	H'FCFF98B4	32
Table setting register B13 in gamma correction block	GAM_B_LUT13	R/W	H'0400 0400	H'FCFF98B8	32
Table setting register B14 in gamma correction block	GAM_B_LUT14	R/W	H'0400 0400	H'FCFF98BC	32
Table setting register B15 in gamma correction block	GAM_B_LUT15	R/W	H'0400 0400	H'FCFF98C0	32
Table setting register B16 in gamma correction block	GAM_B_LUT16	R/W	H'0400 0400	H'FCFF98C4	32

**Table 37.26 Gamma Correction Block Register Configuration (Channel 1)**

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Area setting register B1 in gamma correction block	GAM_B_AREA1	R/W	H'0008 1018	H'FCFF98C8	32
Area setting register B2 in gamma correction block	GAM_B_AREA2	R/W	H'2028 3038	H'FCFF98CC	32
Area setting register B3 in gamma correction block	GAM_B_AREA3	R/W	H'4048 5058	H'FCFF98D0	32
Area setting register B4 in gamma correction block	GAM_B_AREA4	R/W	H'6068 7078	H'FCFF98D4	32
Area setting register B5 in gamma correction block	GAM_B_AREA5	R/W	H'8088 9098	H'FCFF98D8	32
Area setting register B6 in gamma correction block	GAM_B_AREA6	R/W	H'A0A8 B0B8	H'FCFF98DC	32
Area setting register B7 in gamma correction block	GAM_B_AREA7	R/W	H'C0C8 D0D8	H'FCFF98E0	32
Area setting register B8 in gamma correction block	GAM_B_AREA8	R/W	H'E0E8 F0F8	H'FCFF98E4	32
Register update control register R in gamma correction block	GAM_R_UPDATE	R/WC1	H'0000 0000	H'FCFF9900	32
Table setting register R1 in gamma correction block	GAM_R_LUT1	R/W	H'0400 0400	H'FCFF9908	32
Table setting register R2 in gamma correction block	GAM_R_LUT2	R/W	H'0400 0400	H'FCFF990C	32
Table setting register R3 in gamma correction block	GAM_R_LUT3	R/W	H'0400 0400	H'FCFF9910	32
Table setting register R4 in gamma correction block	GAM_R_LUT4	R/W	H'0400 0400	H'FCFF9914	32
Table setting register R5 in gamma correction block	GAM_R_LUT5	R/W	H'0400 0400	H'FCFF9918	32
Table setting register R6 in gamma correction block	GAM_R_LUT6	R/W	H'0400 0400	H'FCFF991C	32
Table setting register R7 in gamma correction block	GAM_R_LUT7	R/W	H'0400 0400	H'FCFF9920	32
Table setting register R8 in gamma correction block	GAM_R_LUT8	R/W	H'0400 0400	H'FCFF9924	32
Table setting register R9 in gamma correction block	GAM_R_LUT9	R/W	H'0400 0400	H'FCFF9928	32
Table setting register R10 in gamma correction block	GAM_R_LUT10	R/W	H'0400 0400	H'FCFF992C	32
Table setting register R11 in gamma correction block	GAM_R_LUT11	R/W	H'0400 0400	H'FCFF9930	32
Table setting register R12 in gamma correction block	GAM_R_LUT12	R/W	H'0400 0400	H'FCFF9934	32
Table setting register R13 in gamma correction block	GAM_R_LUT13	R/W	H'0400 0400	H'FCFF9938	32
Table setting register R14 in gamma correction block	GAM_R_LUT14	R/W	H'0400 0400	H'FCFF993C	32
Table setting register R15 in gamma correction block	GAM_R_LUT15	R/W	H'0400 0400	H'FCFF9940	32
Table setting register R16 in gamma correction block	GAM_R_LUT16	R/W	H'0400 0400	H'FCFF9944	32
Area setting register R1 in gamma correction block	GAM_R_AREA1	R/W	H'0008 1018	H'FCFF9948	32

**Table 37.26 Gamma Correction Block Register Configuration (Channel 1)**

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Area setting register R2 in gamma correction block	GAM_R_AREA2	R/W	H'2028 3038	H'FCFF994C	32
Area setting register R3 in gamma correction block	GAM_R_AREA3	R/W	H'4048 5058	H'FCFF9950	32
Area setting register R4 in gamma correction block	GAM_R_AREA4	R/W	H'6068 7078	H'FCFF9954	32
Area setting register R5 in gamma correction block	GAM_R_AREA5	R/W	H'8088 9098	H'FCFF9958	32
Area setting register R6 in gamma correction block	GAM_R_AREA6	R/W	H'A0A8 B0B8	H'FCFF995C	32
Area setting register R7 in gamma correction block	GAM_R_AREA7	R/W	H'C0C8 D0D8	H'FCFF9960	32
Area setting register R8 in gamma correction block	GAM_R_AREA8	R/W	H'E0E8 F0F8	H'FCFF9964	32

**Table 37.27 TCON Block Register Configuration (Channel 1)**

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
TCON register update control register	TCON_UPDATE	R/WC1	H'0000 0000	H'FCFF9980	32
TCON reference timing setting register	TCON_TIM	R/W	H'0190 0000	H'FCFF9984	32
TCON vertical timing setting register A1	TCON_TIM_STVA1	R/W	H'0000 0004	H'FCFF9988	32
TCON vertical timing setting register A2	TCON_TIM_STVA2	R/W	H'0000 0010	H'FCFF998C	32
TCON vertical timing setting register B1	TCON_TIM_STVB1	R/W	H'0046 03C0	H'FCFF9990	32
TCON vertical timing setting register B2	TCON_TIM_STVB2	R/W	H'0000 0001	H'FCFF9994	32
TCON horizontal timing setting register STH1	TCON_TIM_STH1	R/W	H'0000 0060	H'FCFF9998	32
TCON horizontal timing setting register STH2	TCON_TIM_STH2	R/W	H'0000 0012	H'FCFF999C	32
TCON horizontal timing setting register STB1	TCON_TIM_STB1	R/W	H'0090 0280	H'FCFF99A0	32
TCON horizontal timing setting register STB2	TCON_TIM_STB2	R/W	H'0000 0007	H'FCFF99A4	32
TCON horizontal timing setting register CPV1	TCON_TIM_CPV1	R/W	H'0000 0000	H'FCFF99A8	32
TCON horizontal timing setting register CPV2	TCON_TIM_CPV2	R/W	H'0000 0004	H'FCFF99AC	32
TCON horizontal timing setting register POLA1	TCON_TIM_POLA1	R/W	H'0000 0000	H'FCFF99B0	32
TCON horizontal timing setting register POLA2	TCON_TIM_POLA2	R/W	H'0000 1005	H'FCFF99B4	32
TCON horizontal timing setting register POLB1	TCON_TIM_POLB1	R/W	H'0000 0000	H'FCFF99B8	32
TCON horizontal timing setting register POLB2	TCON_TIM_POLB2	R/W	H'0000 1006	H'FCFF99BC	32
TCON data enable polarity setting register	TCON_TIM_DE	R/W	H'0000 0000	H'FCFF99C0	32

**Table 37.28 Output Controller Register Configuration (Channel 1)**

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Register update control register in output controller	OUT_UPDATE	R/WC1	H'0000 0000	H'FCFF9A00	32
Output interface register	OUT_SET	R/W	H'001F 0000	H'FCFF9A04	32
Brightness (DC) correction register 1	OUT_BRIGHT1	R/W	H'0000 0200	H'FCFF9A08	32
Brightness (DC) correction register 2	OUT_BRIGHT2	R/W	H'0200 0200	H'FCFF9A0C	32
Contrast (gain) correction register	OUT_CONTRAST	R/W	H'0080 8080	H'FCFF9A10	32
Panel dither register	OUT_PDTHA	R/W	H'0000 3021	H'FCFF9A14	32
Output phase control register	OUT_CLK_PHASE	R/W	H'0000 0000	H'FCFF9A24	32

### 37.2.1 Register Update Control Register G in Gamma Correction Block (GAM\_G\_UPDATE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GAM_G_VEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/WC1

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GAM_G_VEN	0	R/WC1	Gamma Correction (G) Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.

### 37.2.2 Function Switch Register in Gamma Correction Block (GAM\_SW)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GAM_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GAM_ON	0	R/W	Gamma Correction On/Off Control 0: Off 1: On

Note: This register is updated when GAM\_G\_VEN in GAM\_G\_UPDATE is 1.

### 37.2.3 Table Setting Register G1 to G16 in Gamma Correction Block (GAM\_G\_LUT1 to GAM\_G\_LUT16)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GAM_G_GAIN_xx[10:0]										
Initial value:	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GAM_G_GAIN_yy[10:0]										
Initial value:	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	*	1024	R/W	GAM_G_LUT1: Gain Adjustment of Area 0 of G Signal GAM_G_LUT2: Gain Adjustment of Area 2 of G Signal GAM_G_LUT3: Gain Adjustment of Area 4 of G Signal GAM_G_LUT4: Gain Adjustment of Area 6 of G Signal GAM_G_LUT5: Gain Adjustment of Area 8 of G Signal GAM_G_LUT6: Gain Adjustment of Area 10 of G Signal GAM_G_LUT7: Gain Adjustment of Area 12 of G Signal GAM_G_LUT8: Gain Adjustment of Area 14 of G Signal GAM_G_LUT9: Gain Adjustment of Area 16 of G Signal GAM_G_LUT10: Gain Adjustment of Area 18 of G Signal GAM_G_LUT11: Gain Adjustment of Area 20 of G Signal GAM_G_LUT12: Gain Adjustment of Area 22 of G Signal GAM_G_LUT13: Gain Adjustment of Area 24 of G Signal GAM_G_LUT14: Gain Adjustment of Area 26 of G Signal GAM_G_LUT15: Gain Adjustment of Area 28 of G Signal GAM_G_LUT16: Gain Adjustment of Area 30 of G Signal Unsigned (0 to 2047 [LSB], 1024 [LSB] = 1.0 [times])
26 to 16	*	1024	R/W	*: Bit Name GAM_G_LUT1: GAM_G_GAIN_00[10:0] GAM_G_LUT2: GAM_G_GAIN_02[10:0] GAM_G_LUT3: GAM_G_GAIN_04[10:0] GAM_G_LUT4: GAM_G_GAIN_06[10:0] GAM_G_LUT5: GAM_G_GAIN_08[10:0] GAM_G_LUT6: GAM_G_GAIN_10[10:0] GAM_G_LUT7: GAM_G_GAIN_12[10:0] GAM_G_LUT8: GAM_G_GAIN_14[10:0] GAM_G_LUT9: GAM_G_GAIN_16[10:0] GAM_G_LUT10: GAM_G_GAIN_18[10:0] GAM_G_LUT11: GAM_G_GAIN_20[10:0] GAM_G_LUT12: GAM_G_GAIN_22[10:0] GAM_G_LUT13: GAM_G_GAIN_24[10:0] GAM_G_LUT14: GAM_G_GAIN_26[10:0] GAM_G_LUT15: GAM_G_GAIN_28[10:0] GAM_G_LUT16: GAM_G_GAIN_30[10:0]
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
10 to 0	*	1024	R/W	GAM_G_LUT1: Gain Adjustment of Area 1 of G Signal GAM_G_LUT2: Gain Adjustment of Area 3 of G Signal GAM_G_LUT3: Gain Adjustment of Area 5 of G Signal GAM_G_LUT4: Gain Adjustment of Area 7 of G Signal GAM_G_LUT5: Gain Adjustment of Area 9 of G Signal GAM_G_LUT6: Gain Adjustment of Area 11 of G Signal GAM_G_LUT7: Gain Adjustment of Area 13 of G Signal GAM_G_LUT8: Gain Adjustment of Area 15 of G Signal GAM_G_LUT9: Gain Adjustment of Area 17 of G Signal GAM_G_LUT10: Gain Adjustment of Area 19 of G Signal
10 to 0	*	1024	R/W	GAM_G_LUT11: Gain Adjustment of Area 21 of G Signal GAM_G_LUT12: Gain Adjustment of Area 23 of G Signal GAM_G_LUT13: Gain Adjustment of Area 25 of G Signal GAM_G_LUT14: Gain Adjustment of Area 27 of G Signal GAM_G_LUT15: Gain Adjustment of Area 29 of G Signal GAM_G_LUT16: Gain Adjustment of Area 31 of G Signal Unsigned (0 to 2047 [LSB], 1024 [LSB] = 1.0 [times]) *: Bit Name GAM_G_LUT1: GAM_G_GAIN_01[10:0] GAM_G_LUT2: GAM_G_GAIN_03[10:0] GAM_G_LUT3: GAM_G_GAIN_05[10:0] GAM_G_LUT4: GAM_G_GAIN_07[10:0] GAM_G_LUT5: GAM_G_GAIN_09[10:0] GAM_G_LUT6: GAM_G_GAIN_11[10:0] GAM_G_LUT7: GAM_G_GAIN_13[10:0] GAM_G_LUT8: GAM_G_GAIN_15[10:0] GAM_G_LUT9: GAM_G_GAIN_17[10:0] GAM_G_LUT10: GAM_G_GAIN_19[10:0] GAM_G_LUT11: GAM_G_GAIN_21[10:0] GAM_G_LUT12: GAM_G_GAIN_23[10:0] GAM_G_LUT13: GAM_G_GAIN_25[10:0] GAM_G_LUT14: GAM_G_GAIN_27[10:0] GAM_G_LUT15: GAM_G_GAIN_29[10:0] GAM_G_LUT16: GAM_G_GAIN_31[10:0]

Note: This register is updated when GAM\_G\_VEN in GAM\_G\_UPDATE is 1.



## 37.2.4 Area Setting Register G1 in Gamma Correction Block (GAM\_G\_AREA1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	GAM_G_TH_01[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_G_TH_02[7:0]							GAM_G_TH_03[7:0]								
Initial value:	0	0	0	1	0	0	0	0	0	0	0	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	GAM_G_TH_01[7:0]	8	R/W	Start Threshold of Area 1 of G Signal Unsigned (0 to 255 [LSB]) $0 \leq \text{Threshold of current area} < \text{Threshold of next area}$
15 to 8	GAM_G_TH_02[7:0]	16	R/W	Start Threshold of Area 2 of G Signal Unsigned (0 to 255 [LSB]) $\text{Threshold of previous area} \leq \text{Threshold of current area} < \text{Threshold of next area}$
7 to 0	GAM_G_TH_03[7:0]	24	R/W	Start Threshold of Area 3 of G Signal Unsigned (0 to 255 [LSB]) $\text{Threshold of previous area} \leq \text{Threshold of current area} < \text{Threshold of next area d}$

Note: This register is updated when GAM\_G\_VEN in GAM\_G\_UPDATE is 1.

## 37.2.5 Area Setting Register G2 in Gamma Correction Block (GAM\_G\_AREA2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_G_TH_04[7:0]								GAM_G_TH_05[7:0]							
Initial value:	0	0	1	0	0	0	0	0	0	0	1	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_G_TH_06[7:0]								GAM_G_TH_07[7:0]							
Initial value:	0	0	1	1	0	0	0	0	0	0	1	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_G_TH_04[7:0]	32	R/W	Start Threshold of Area 4 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
23 to 16	GAM_G_TH_05[7:0]	40	R/W	Start Threshold of Area 5 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
15 to 8	GAM_G_TH_06[7:0]	48	R/W	Start Threshold of Area 6 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
7 to 0	GAM_G_TH_07[7:0]	56	R/W	Start Threshold of Area 7 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area d

Note: This register is updated when GAM\_G\_VEN in GAM\_G\_UPDATE is 1.

## 37.2.6 Area Setting Register G3 in Gamma Correction Block (GAM\_G\_AREA3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_G_TH_08[7:0]								GAM_G_TH_09[7:0]							
Initial value:	0	1	0	0	0	0	0	0	0	1	0	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_G_TH_10[7:0]								GAM_G_TH_11[7:0]							
Initial value:	0	1	0	1	0	0	0	0	0	1	0	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_G_TH_08[7:0]	64	R/W	Start Threshold of Area 8 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
23 to 16	GAM_G_TH_09[7:0]	72	R/W	Start Threshold of Area 9 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
15 to 8	GAM_G_TH_10[7:0]	80	R/W	Start Threshold of Area 10 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
7 to 0	GAM_G_TH_11[7:0]	88	R/W	Start Threshold of Area 11 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area

Note: This register is updated when GAM\_G\_VEN in GAM\_G\_UPDATE is 1.

## 37.2.7 Area Setting Register G4 in Gamma Correction Block (GAM\_G\_AREA4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_G_TH_12[7:0]								GAM_G_TH_13[7:0]							
Initial value:	0	1	1	0	0	0	0	0	0	1	1	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_G_TH_14[7:0]								GAM_G_TH_15[7:0]							
Initial value:	0	1	1	1	0	0	0	0	0	1	1	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_G_TH_12[7:0]	96	R/W	Start Threshold of Area 12 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
23 to 16	GAM_G_TH_13[7:0]	104	R/W	Start Threshold of Area 13 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
15 to 8	GAM_G_TH_14[7:0]	112	R/W	Start Threshold of Area 14 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
7 to 0	GAM_G_TH_15[7:0]	120	R/W	Start Threshold of Area 15 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area

Note: This register is updated when GAM\_G\_VEN in GAM\_G\_UPDATE is 1.

### 37.2.8 Area Setting Register G5 in Gamma Correction Block (GAM\_G\_AREA5)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_G_TH_16[7:0]								GAM_G_TH_17[7:0]							
Initial value:	1	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_G_TH_18[7:0]								GAM_G_TH_19[7:0]							
Initial value:	1	0	0	1	0	0	0	0	1	0	0	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_G_TH_16 [7:0]	128	R/W	Start Threshold of Area 16 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
23 to 16	GAM_G_TH_17 [7:0]	136	R/W	Start Threshold of Area 17 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
15 to 8	GAM_G_TH_18 [7:0]	144	R/W	Start Threshold of Area 18 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
7 to 0	GAM_G_TH_19 [7:0]	152	R/W	Start Threshold of Area 19 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area

Note: This register is updated when GAM\_G\_VEN in GAM\_G\_UPDATE is 1.

### 37.2.9 Area Setting Register G6 in Gamma Correction Block (GAM\_G\_AREA6)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_G_TH_20[7:0]								GAM_G_TH_21[7:0]							
Initial value:	1	0	1	0	0	0	0	0	1	0	1	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_G_TH_22[7:0]								GAM_G_TH_23[7:0]							
Initial value:	1	0	1	1	0	0	0	0	1	0	1	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_G_TH_20 [7:0]	160	R/W	Start Threshold of Area 20 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
23 to 16	GAM_G_TH_21 [7:0]	168	R/W	Start Threshold of Area 21 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
15 to 8	GAM_G_TH_22 [7:0]	176	R/W	Start Threshold of Area 22 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
7 to 0	GAM_G_TH_23 [7:0]	184	R/W	Start Threshold of Area 23 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area

Note: This register is updated when GAM\_G\_VEN in GAM\_G\_UPDATE is 1.

## 37.2.10 Area Setting Register G7 in Gamma Correction Block (GAM\_G\_AREA7)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_G_TH_24[7:0]								GAM_G_TH_25[7:0]							
Initial value:	1	1	0	0	0	0	0	0	1	1	0	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_G_TH_26[7:0]								GAM_G_TH_27[7:0]							
Initial value:	1	1	0	1	0	0	0	0	1	1	0	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_G_TH_24 [7:0]	192	R/W	Start Threshold of Area 24 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
23 to 16	GAM_G_TH_25 [7:0]	200	R/W	Start Threshold of Area 25 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
15 to 8	GAM_G_TH_26 [7:0]	208	R/W	Start Threshold of Area 26 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
7 to 0	GAM_G_TH_27 [7:0]	216	R/W	Start Threshold of Area 27 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area

Note: This register is updated when GAM\_G\_VEN in GAM\_G\_UPDATE is 1.

## 37.2.11 Area Setting Register G8 in Gamma Correction Block (GAM\_G\_AREA8)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_G_TH_28[7:0]								GAM_G_TH_29[7:0]							
Initial value:	1	1	1	0	0	0	0	0	1	1	1	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_G_TH_30[7:0]								GAM_G_TH_31[7:0]							
Initial value:	1	1	1	1	0	0	0	0	1	1	1	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_G_TH_28 [7:0]	224	R/W	Start Threshold of Area 28 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
23 to 16	GAM_G_TH_29 [7:0]	232	R/W	Start Threshold of Area 29 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
15 to 8	GAM_G_TH_30 [7:0]	240	R/W	Start Threshold of Area 30 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
7 to 0	GAM_G_TH_31 [7:0]	248	R/W	Start Threshold of Area 31 of G Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area

Note: This register is updated when GAM\_G\_VEN in GAM\_G\_UPDATE is 1.



37.2.12 Register Update Control Register B in Gamma Correction Block (GAM\_B\_UPDATE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GAM_B_VEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/WC1

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GAM_B_VEN	0	R/WC1	Gamma Correction (B) Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.

### 37.2.13 Table Setting Register B1 to B16 in Gamma Correction Block (GAM\_B\_LUT1 to GAM\_B\_LUT16)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GAM_B_GAIN_xx[10:0]										
Initial value:	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GAM_B_GAIN_yy[10:0]										
Initial value:	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	*	1024	R/W	GAM_B_LUT1: Gain Adjustment of Area 0 of B Signal GAM_B_LUT2: Gain Adjustment of Area 2 of B Signal GAM_B_LUT3: Gain Adjustment of Area 4 of B Signal GAM_B_LUT4: Gain Adjustment of Area 6 of B Signal GAM_B_LUT5: Gain Adjustment of Area 8 of B Signal GAM_B_LUT6: Gain Adjustment of Area 10 of B Signal GAM_B_LUT7: Gain Adjustment of Area 12 of B Signal GAM_B_LUT8: Gain Adjustment of Area 14 of B Signal GAM_B_LUT9: Gain Adjustment of Area 16 of B Signal GAM_B_LUT10: Gain Adjustment of Area 18 of B Signal GAM_B_LUT11: Gain Adjustment of Area 20 of B Signal GAM_B_LUT12: Gain Adjustment of Area 22 of B Signal GAM_B_LUT13: Gain Adjustment of Area 24 of B Signal GAM_B_LUT14: Gain Adjustment of Area 26 of B Signal GAM_B_LUT15: Gain Adjustment of Area 28 of B Signal GAM_B_LUT16: Gain Adjustment of Area 30 of B Signal Unsigned (0 to 2047 [LSB], 1024 [LSB] = 1.0 [times])
26 to 16	*	1024	R/W	*: Bit Name GAM_B_LUT1: GAM_B_GAIN_00[10:0] GAM_B_LUT2: GAM_B_GAIN_02[10:0] GAM_B_LUT3: GAM_B_GAIN_04[10:0] GAM_B_LUT4: GAM_B_GAIN_06[10:0] GAM_B_LUT5: GAM_B_GAIN_08[10:0] GAM_B_LUT6: GAM_B_GAIN_10[10:0] GAM_B_LUT7: GAM_B_GAIN_12[10:0] GAM_B_LUT8: GAM_B_GAIN_14[10:0] GAM_B_LUT9: GAM_B_GAIN_16[10:0] GAM_B_LUT10: GAM_B_GAIN_18[10:0] GAM_B_LUT11: GAM_B_GAIN_20[10:0] GAM_B_LUT12: GAM_B_GAIN_22[10:0] GAM_B_LUT13: GAM_B_GAIN_24[10:0] GAM_B_LUT14: GAM_B_GAIN_26[10:0] GAM_B_LUT15: GAM_B_GAIN_28[10:0] GAM_B_LUT16: GAM_B_GAIN_30[10:0]
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
10 to 0	*	1024	R/W	GAM_B_LUT1: Gain Adjustment of Area 1 of B Signal GAM_B_LUT2: Gain Adjustment of Area 3 of B Signal GAM_B_LUT3: Gain Adjustment of Area 5 of B Signal GAM_B_LUT4: Gain Adjustment of Area 7 of B Signal GAM_B_LUT5: Gain Adjustment of Area 9 of B Signal GAM_B_LUT6: Gain Adjustment of Area 11 of B Signal GAM_B_LUT7: Gain Adjustment of Area 13 of B Signal GAM_B_LUT8: Gain Adjustment of Area 15 of B Signal GAM_B_LUT9: Gain Adjustment of Area 17 of B Signal GAM_B_LUT10: Gain Adjustment of Area 19 of B Signal GAM_B_LUT11: Gain Adjustment of Area 21 of B Signal GAM_B_LUT12: Gain Adjustment of Area 23 of B Signal GAM_B_LUT13: Gain Adjustment of Area 25 of B Signal GAM_B_LUT14: Gain Adjustment of Area 27 of B Signal GAM_B_LUT15: Gain Adjustment of Area 29 of B Signal GAM_B_LUT16: Gain Adjustment of Area 31 of B Signal Unsigned (0 to 2047 [LSB], 1024 [LSB] = 1.0 [times]) *: Bit Name GAM_B_LUT1: GAM_B_GAIN_01[10:0] GAM_B_LUT2: GAM_B_GAIN_03[10:0] GAM_B_LUT3: GAM_B_GAIN_05[10:0] GAM_B_LUT4: GAM_B_GAIN_07[10:0] GAM_B_LUT5: GAM_B_GAIN_09[10:0] GAM_B_LUT6: GAM_B_GAIN_11[10:0] GAM_B_LUT7: GAM_B_GAIN_13[10:0] GAM_B_LUT8: GAM_B_GAIN_15[10:0] GAM_B_LUT9: GAM_B_GAIN_17[10:0] GAM_B_LUT10: GAM_B_GAIN_19[10:0] GAM_B_LUT11: GAM_B_GAIN_21[10:0] GAM_B_LUT12: GAM_B_GAIN_23[10:0] GAM_B_LUT13: GAM_B_GAIN_25[10:0] GAM_B_LUT14: GAM_B_GAIN_27[10:0] GAM_B_LUT15: GAM_B_GAIN_29[10:0] GAM_B_LUT16: GAM_B_GAIN_31[10:0]

Note: This register is updated when GAM\_B\_VEN in GAM\_B\_UPDATE is 1.

## 37.2.14 Area Setting Register B1 in Gamma Correction Block (GAM\_B\_AREA1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	GAM_B_TH_01[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_B_TH_02[7:0]								GAM_B_TH_03[7:0]							
Initial value:	0	0	0	1	0	0	0	0	0	0	0	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	GAM_B_TH_01[7:0]	8	R/W	Start Threshold of Area 1 of B Signal Unsigned (0 to 255 [LSB]) $0 \leq \text{Threshold of current area} < \text{Threshold of next area}$
15 to 8	GAM_B_TH_02[7:0]	16	R/W	Start Threshold of Area 2 of B Signal Unsigned (0 to 255 [LSB]) $\text{Threshold of previous area} \leq \text{Threshold of current area} < \text{Threshold of next area}$
7 to 0	GAM_B_TH_03[7:0]	24	R/W	Start Threshold of Area 3 of B Signal Unsigned (0 to 255 [LSB]) $\text{Threshold of previous area} \leq \text{Threshold of current area} < \text{Threshold of next area d}$

Note: This register is updated when GAM\_B\_VEN in GAM\_B\_UPDATE is 1.

## 37.2.15 Area Setting Register B2 in Gamma Correction Block (GAM\_B\_AREA2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_B_TH_04[7:0]								GAM_B_TH_05[7:0]							
Initial value:	0	0	1	0	0	0	0	0	0	0	1	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_B_TH_06[7:0]								GAM_B_TH_07[7:0]							
Initial value:	0	0	1	1	0	0	0	0	0	0	1	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_B_TH_04[7:0]	32	R/W	Start Threshold of Area 4 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
23 to 16	GAM_B_TH_05[7:0]	40	R/W	Start Threshold of Area 5 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
15 to 8	GAM_B_TH_06[7:0]	48	R/W	Start Threshold of Area 6 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
7 to 0	GAM_B_TH_07[7:0]	56	R/W	Start Threshold of Area 7 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area

Note: This register is updated when GAM\_B\_VEN in GAM\_B\_UPDATE is 1.

## 37.2.16 Area Setting Register B3 in Gamma Correction Block (GAM\_B\_AREA3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_B_TH_08[7:0]								GAM_B_TH_09[7:0]							
Initial value:	0	1	0	0	0	0	0	0	0	1	0	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_B_TH_10[7:0]								GAM_B_TH_11[7:0]							
Initial value:	0	1	0	1	0	0	0	0	0	1	0	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_B_TH_08[7:0]	64	R/W	Start Threshold of Area 8 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
23 to 16	GAM_B_TH_09[7:0]	72	R/W	Start Threshold of Area 9 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
15 to 8	GAM_B_TH_10[7:0]	80	R/W	Start Threshold of Area 10 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
7 to 0	GAM_B_TH_11[7:0]	88	R/W	Start Threshold of Area 11 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area

Note: This register is updated when GAM\_B\_VEN in GAM\_B\_UPDATE is 1.

## 37.2.17 Area Setting Register B4 in Gamma Correction Block (GAM\_B\_AREA4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_B_TH_12[7:0]								GAM_B_TH_13[7:0]							
Initial value:	0	1	1	0	0	0	0	0	0	1	1	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_B_TH_14[7:0]								GAM_B_TH_15[7:0]							
Initial value:	0	1	1	1	0	0	0	0	0	1	1	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_B_TH_12[7:0]	96	R/W	Start Threshold of Area 12 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
23 to 16	GAM_B_TH_13[7:0]	104	R/W	Start Threshold of Area 13 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
15 to 8	GAM_B_TH_14[7:0]	112	R/W	Start Threshold of Area 14 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
7 to 0	GAM_B_TH_15[7:0]	120	R/W	Start Threshold of Area 15 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area

Note: This register is updated when GAM\_B\_VEN in GAM\_B\_UPDATE is 1.

## 37.2.18 Area Setting Register B5 in Gamma Correction Block (GAM\_B\_AREA5)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_B_TH_16[7:0]								GAM_B_TH_17[7:0]							
Initial value:	1	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_B_TH_18[7:0]								GAM_B_TH_19[7:0]							
Initial value:	1	0	0	1	0	0	0	0	1	0	0	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_B_TH_16[7:0]	128	R/W	Start Threshold of Area 16 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
23 to 16	GAM_B_TH_17[7:0]	136	R/W	Start Threshold of Area 17 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
15 to 8	GAM_B_TH_18[7:0]	144	R/W	Start Threshold of Area 18 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
7 to 0	GAM_B_TH_19[7:0]	152	R/W	Start Threshold of Area 19 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area

Note: This register is updated when GAM\_B\_VEN in GAM\_B\_UPDATE is 1.



## 37.2.19 Area Setting Register B6 in Gamma Correction Block (GAM\_B\_AREA6)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_B_TH_20[7:0]								GAM_B_TH_21[7:0]							
Initial value:	1	0	1	0	0	0	0	0	1	0	1	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_B_TH_22[7:0]								GAM_B_TH_23[7:0]							
Initial value:	1	0	1	1	0	0	0	0	1	0	1	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_B_TH_20[7:0]	160	R/W	Start Threshold of Area 20 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
23 to 16	GAM_B_TH_21[7:0]	168	R/W	Start Threshold of Area 21 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
15 to 8	GAM_B_TH_22[7:0]	176	R/W	Start Threshold of Area 22 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
7 to 0	GAM_B_TH_23[7:0]	184	R/W	Start Threshold of Area 23 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area

Note: This register is updated when GAM\_B\_VEN in GAM\_B\_UPDATE is 1.

## 37.2.20 Area Setting Register B7 in Gamma Correction Block (GAM\_B\_AREA7)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_B_TH_24[7:0]								GAM_B_TH_25[7:0]							
Initial value:	1	1	0	0	0	0	0	0	1	1	0	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_B_TH_26[7:0]								GAM_B_TH_27[7:0]							
Initial value:	1	1	0	1	0	0	0	0	1	1	0	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_B_TH_24[7:0]	192	R/W	Start Threshold of Area 24 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
23 to 16	GAM_B_TH_25[7:0]	200	R/W	Start Threshold of Area 25 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
15 to 8	GAM_B_TH_26[7:0]	208	R/W	Start Threshold of Area 26 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
7 to 0	GAM_B_TH_27[7:0]	216	R/W	Start Threshold of Area 27 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area

Note: This register is updated when GAM\_B\_VEN in GAM\_B\_UPDATE is 1.

## 37.2.21 Area Setting Register B8 in Gamma Correction Block (GAM\_B\_AREA8)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_B_TH_28[7:0]								GAM_B_TH_29[7:0]							
Initial value:	1	1	1	0	0	0	0	0	1	1	1	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_B_TH_30[7:0]								GAM_B_TH_31[7:0]							
Initial value:	1	1	1	1	0	0	0	0	1	1	1	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_B_TH_28[7:0]	224	R/W	Start Threshold of Area 28 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
23 to 16	GAM_B_TH_29[7:0]	232	R/W	Start Threshold of Area 29 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
15 to 8	GAM_B_TH_30[7:0]	240	R/W	Start Threshold of Area 30 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
7 to 0	GAM_B_TH_31[7:0]	248	R/W	Start Threshold of Area 31 of B Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area

Note: This register is updated when GAM\_B\_VEN in GAM\_B\_UPDATE is 1.

### 37.2.22 Register Update Control Register R in Gamma Correction Block (GAM\_R\_UPDATE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GAM_R_VEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/WC1

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GAM_R_VEN	0	R/WC1	Gamma Correction (R) Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.

### 37.2.23 Table Setting Register R1 to R16 in Gamma Correction Block (GAM\_R\_LUT1 to GAM\_R\_LUT16)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	GAM_G_GAIN_xx[10:0]										
Initial value:	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	GAM_G_GAIN_yy[10:0]										
Initial value:	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	*	1024	R/W	GAM_R_LUT1: Gain Adjustment of Area 0 of R Signal GAM_R_LUT2: Gain Adjustment of Area 2 of R Signal GAM_R_LUT3: Gain Adjustment of Area 4 of R Signal GAM_R_LUT4: Gain Adjustment of Area 6 of R Signal GAM_R_LUT5: Gain Adjustment of Area 8 of R Signal GAM_R_LUT6: Gain Adjustment of Area 10 of R Signal GAM_R_LUT7: Gain Adjustment of Area 12 of R Signal GAM_R_LUT8: Gain Adjustment of Area 14 of R Signal GAM_R_LUT9: Gain Adjustment of Area 16 of R Signal GAM_R_LUT10: Gain Adjustment of Area 18 of R Signal GAM_R_LUT11: Gain Adjustment of Area 20 of R Signal GAM_R_LUT12: Gain Adjustment of Area 22 of R Signal GAM_R_LUT13: Gain Adjustment of Area 24 of R Signal GAM_R_LUT14: Gain Adjustment of Area 26 of R Signal GAM_R_LUT15: Gain Adjustment of Area 28 of R Signal GAM_R_LUT16: Gain Adjustment of Area 30 of R Signal Unsigned (0 to 2047 [LSB], 1024 [LSB] = 1.0 [times]) *: Bit Name GAM_R_LUT1: GAM_R_GAIN_00[10:0] GAM_R_LUT2: GAM_R_GAIN_02[10:0] GAM_R_LUT3: GAM_R_GAIN_04[10:0] GAM_R_LUT4: GAM_R_GAIN_06[10:0] GAM_R_LUT5: GAM_R_GAIN_08[10:0] GAM_R_LUT6: GAM_R_GAIN_10[10:0] GAM_R_LUT7: GAM_R_GAIN_12[10:0] GAM_R_LUT8: GAM_R_GAIN_14[10:0] GAM_R_LUT9: GAM_R_GAIN_16[10:0] GAM_R_LUT10: GAM_R_GAIN_18[10:0] GAM_R_LUT11: GAM_R_GAIN_20[10:0] GAM_R_LUT12: GAM_R_GAIN_22[10:0] GAM_R_LUT13: GAM_R_GAIN_24[10:0] GAM_R_LUT14: GAM_R_GAIN_26[10:0] GAM_R_LUT15: GAM_R_GAIN_28[10:0] GAM_R_LUT16: GAM_R_GAIN_30[10:0]
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
10 to 0	*	1024	R/W	GAM_R_LUT1: Gain Adjustment of Area 1 of R Signal GAM_R_LUT2: Gain Adjustment of Area 3 of R Signal GAM_R_LUT3: Gain Adjustment of Area 5 of R Signal GAM_R_LUT4: Gain Adjustment of Area 7 of R Signal GAM_R_LUT5: Gain Adjustment of Area 9 of R Signal GAM_R_LUT6: Gain Adjustment of Area 11 of R Signal GAM_R_LUT7: Gain Adjustment of Area 13 of R Signal GAM_R_LUT8: Gain Adjustment of Area 15 of R Signal GAM_R_LUT9: Gain Adjustment of Area 17 of R Signal GAM_R_LUT10: Gain Adjustment of Area 19 of R Signal GAM_R_LUT11: Gain Adjustment of Area 21 of R Signal GAM_R_LUT12: Gain Adjustment of Area 23 of R Signal GAM_R_LUT13: Gain Adjustment of Area 25 of R Signal GAM_R_LUT14: Gain Adjustment of Area 27 of R Signal GAM_R_LUT15: Gain Adjustment of Area 29 of R Signal GAM_R_LUT16: Gain Adjustment of Area 31 of R Signal Unsigned (0 to 2047 [LSB], 1024 [LSB] = 1.0 [times]) *: Bit Name GAM_R_LUT1: GAM_R_GAIN_01[10:0] GAM_R_LUT2: GAM_R_GAIN_03[10:0] GAM_R_LUT3: GAM_R_GAIN_05[10:0] GAM_R_LUT4: GAM_R_GAIN_07[10:0] GAM_R_LUT5: GAM_R_GAIN_09[10:0] GAM_R_LUT6: GAM_R_GAIN_11[10:0] GAM_R_LUT7: GAM_R_GAIN_13[10:0] GAM_R_LUT8: GAM_R_GAIN_15[10:0] GAM_R_LUT9: GAM_R_GAIN_17[10:0] GAM_R_LUT10: GAM_R_GAIN_19[10:0] GAM_R_LUT11: GAM_R_GAIN_21[10:0] GAM_R_LUT12: GAM_R_GAIN_23[10:0] GAM_R_LUT13: GAM_R_GAIN_25[10:0] GAM_R_LUT14: GAM_R_GAIN_27[10:0] GAM_R_LUT15: GAM_R_GAIN_29[10:0] GAM_R_LUT16: GAM_R_GAIN_31[10:0]

Note: This register is updated when GAM\_R\_VEN in GAM\_R\_UPDATE is 1.

## 37.2.24 Area Setting Register R1 in Gamma Correction Block (GAM\_R\_AREA1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	GAM_R_TH_01[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_R_TH_02[7:0]								GAM_R_TH_03[7:0]							
Initial value:	0	0	0	1	0	0	0	0	0	0	0	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	GAM_R_TH_01[7:0]	8	R/W	Start Threshold of Area 1 of R Signal Unsigned (0 to 255 [LSB]) $0 \leq \text{Threshold of current area} < \text{Threshold of next area}$
15 to 8	GAM_R_TH_02[7:0]	16	R/W	Start Threshold of Area 2 of R Signal Unsigned (0 to 255 [LSB]) $\text{Threshold of previous area} \leq \text{Threshold of current area} < \text{Threshold of next area}$
7 to 0	GAM_R_TH_03[7:0]	24	R/W	Start Threshold of Area 3 of R Signal Unsigned (0 to 255 [LSB]) $\text{Threshold of previous area} \leq \text{Threshold of current area} < \text{Threshold of next area d}$

Note: This register is updated when GAM\_R\_VEN in GAM\_R\_UPDATE is 1.

## 37.2.25 Area Setting Register R2 in Gamma Correction Block (GAM\_R\_AREA2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_R_TH_04[7:0]								GAM_R_TH_05[7:0]							
Initial value:	0	0	1	0	0	0	0	0	0	0	1	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_R_TH_06[7:0]								GAM_R_TH_07[7:0]							
Initial value:	0	0	1	1	0	0	0	0	0	0	1	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_R_TH_04[7:0]	32	R/W	Start Threshold of Area 4 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
23 to 16	GAM_R_TH_05[7:0]	40	R/W	Start Threshold of Area 5 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
15 to 8	GAM_R_TH_06[7:0]	48	R/W	Start Threshold of Area 6 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
7 to 0	GAM_R_TH_07[7:0]	56	R/W	Start Threshold of Area 7 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area d

Note: This register is updated when GAM\_R\_VEN in GAM\_R\_UPDATE is 1.



## 37.2.26 Area Setting Register R3 in Gamma Correction Block (GAM\_R\_AREA3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_R_TH_08[7:0]								GAM_R_TH_09[7:0]							
Initial value:	0	1	0	0	0	0	0	0	0	1	0	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_R_TH_10[7:0]								GAM_R_TH_11[7:0]							
Initial value:	0	1	0	1	0	0	0	0	0	1	0	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_R_TH_08[7:0]	64	R/W	Start Threshold of Area 8 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
23 to 16	GAM_R_TH_09[7:0]	72	R/W	Start Threshold of Area 9 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
15 to 8	GAM_R_TH_10[7:0]	80	R/W	Start Threshold of Area 10 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
7 to 0	GAM_R_TH_11[7:0]	88	R/W	Start Threshold of Area 11 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area

Note: This register is updated when GAM\_R\_VEN in GAM\_R\_UPDATE is 1.

## 37.2.27 Area Setting Register R4 in Gamma Correction Block (GAM\_R\_AREA4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_R_TH_12[7:0]								GAM_R_TH_13[7:0]							
Initial value:	0	1	1	0	0	0	0	0	0	1	1	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_R_TH_14[7:0]								GAM_R_TH_15[7:0]							
Initial value:	0	1	1	1	0	0	0	0	0	1	1	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_R_TH_12[7:0]	96	R/W	Start Threshold of Area 12 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
23 to 16	GAM_R_TH_13[7:0]	104	R/W	Start Threshold of Area 13 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
15 to 8	GAM_R_TH_14[7:0]	112	R/W	Start Threshold of Area 14 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
7 to 0	GAM_R_TH_15[7:0]	120	R/W	Start Threshold of Area 15 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area

Note: This register is updated when GAM\_R\_VEN in GAM\_R\_UPDATE is 1.

## 37.2.28 Area Setting Register R5 in Gamma Correction Block (GAM\_R\_AREA5)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_R_TH_16[7:0]								GAM_R_TH_17[7:0]							
Initial value:	1	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_R_TH_18[7:0]								GAM_R_TH_19[7:0]							
Initial value:	1	0	0	1	0	0	0	0	1	0	0	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_R_TH_16[7:0]	128	R/W	Start Threshold of Area 16 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
23 to 16	GAM_R_TH_17[7:0]	136	R/W	Start Threshold of Area 17 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
15 to 8	GAM_R_TH_18[7:0]	144	R/W	Start Threshold of Area 18 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
7 to 0	GAM_R_TH_19[7:0]	152	R/W	Start Threshold of Area 19 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area

Note: This register is updated when GAM\_R\_VEN in GAM\_R\_UPDATE is 1.

## 37.2.29 Area Setting Register R6 in Gamma Correction Block (GAM\_R\_AREA6)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_R_TH_20[7:0]								GAM_R_TH_21[7:0]							
Initial value:	1	0	1	0	0	0	0	0	1	0	1	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_R_TH_22[7:0]								GAM_R_TH_23[7:0]							
Initial value:	1	0	1	1	0	0	0	0	1	0	1	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_R_TH_20[7:0]	160	R/W	Start Threshold of Area 20 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
23 to 16	GAM_R_TH_21[7:0]	168	R/W	Start Threshold of Area 21 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
15 to 8	GAM_R_TH_22[7:0]	176	R/W	Start Threshold of Area 22 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
7 to 0	GAM_R_TH_23[7:0]	184	R/W	Start Threshold of Area 23 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area

Note: This register is updated when GAM\_R\_VEN in GAM\_R\_UPDATE is 1.

## 37.2.30 Area Setting Register R7 in Gamma Correction Block (GAM\_R\_AREA7)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_R_TH_24[7:0]								GAM_R_TH_25[7:0]							
Initial value:	1	1	0	0	0	0	0	0	1	1	0	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_R_TH_26[7:0]								GAM_R_TH_27[7:0]							
Initial value:	1	1	0	1	0	0	0	0	1	1	0	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_R_TH_24[7:0]	192	R/W	Start Threshold of Area 24 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
23 to 16	GAM_R_TH_25[7:0]	200	R/W	Start Threshold of Area 25 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
15 to 8	GAM_R_TH_26[7:0]	208	R/W	Start Threshold of Area 26 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
7 to 0	GAM_R_TH_27[7:0]	216	R/W	Start Threshold of Area 27 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area

Note: This register is updated when GAM\_R\_VEN in GAM\_R\_UPDATE is 1.

## 37.2.31 Area Setting Register R8 in Gamma Correction Block (GAM\_R\_AREA8)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAM_R_TH_28[7:0]								GAM_R_TH_29[7:0]							
Initial value:	1	1	1	0	0	0	0	0	1	1	1	0	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAM_R_TH_30[7:0]								GAM_R_TH_31[7:0]							
Initial value:	1	1	1	1	0	0	0	0	1	1	1	1	1	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	GAM_R_TH_28[7:0]	224	R/W	Start Threshold of Area 28 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
23 to 16	GAM_R_TH_29[7:0]	232	R/W	Start Threshold of Area 29 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
15 to 8	GAM_R_TH_30[7:0]	240	R/W	Start Threshold of Area 30 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area
7 to 0	GAM_R_TH_31[7:0]	248	R/W	Start Threshold of Area 31 of R Signal Unsigned (0 to 255 [LSB]) Threshold of previous area $\leq$ Threshold of current area < Threshold of next area

Note: This register is updated when GAM\_R\_VEN in GAM\_R\_UPDATE is 1.

## 37.2.32 TCON Register Update Control Register (TCON\_UPDATE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TCON_VEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/WC1

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	TCON_VEN	0	R/WC1	LCD TCON Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.

## 37.2.33 TCON Reference Timing Setting Register (TCON\_TIM)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	TCON_HALF[10:0]										
Initial value:	0	0	0	0	0	0	0	1	1	0	0	1	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	TCON_OFFSET[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	TCON_HALF[10:0]	400	R/W	1/2fH Timing Specifies the clock count from the rising edge of the Hsync signal as the counting timing of horizontal counter.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	TCON_OFFSET[10:0]	0	R/W	Offset Hsync Signal Timing Sets the clock cycle count from the rising edge of the Hsync signal.

Note: This register is updated when TCON\_VEN in TCON\_UPDATE is 1.

## 37.2.34 TCON Vertical Timing Setting Register A1 (TCON\_TIM\_STVA1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	
	—	—	—	—	—	TCON_STVA_VS[10:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	—	—	—	—	—	TCON_STVA_VW[10:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	TCON_STVA_VS [10:0]	0	R/W	STVA Signal Pulse Start Position (First Changing Timing) Starts pulse output after the time specified by the value of TCON_STVA_VS from the rising edge of the Vsync signal (1/2fH cycles).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	TCON_STVA_VW [10:0]	4	R/W	STVA Pulse Width (Second Changing Timing) Outputs a pulse of the duration of the value of TCON_STVA_VW (1/2fH cycles).

Note: This register is updated when TCON\_VEN in TCON\_UPDATE is 1.



## 37.2.35 TCON Vertical Timing Setting Register A2 (TCON\_TIM\_STVA2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	TCON_STVA_INV	—	TCON_STVA_SEL[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	TCON_STVA_INV	1	R/W	Polarity Inversion Control of STVA Signal 0: Not inverted 1: Inverted
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2 to 0	TCON_STVA_SEL [2:0]	0	R/W	Output Signal Select for LCD_TCON0 Pin 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE

Note: This register is updated when TCON\_VEN in TCON\_UPDATE is 1.

## 37.2.36 TCON Vertical Timing Setting Register B1 (TCON\_TIM\_STVB1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	
	—	—	—	—	—	TCON_STVB_VS[10:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	1	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	—	—	—	—	—	TCON_STVB_VW[10:0]											
Initial value:	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	TCON_STVB_VS [10:0]	70	R/W	STVB Signal Pulse Start Position (First Changing Timing) Starts pulse output after the time specified by the value of TCON_STVB_VS from the rising edge of the Vsync signal (1/2fH cycles).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	TCON_STVB_VW [10:0]	960	R/W	STVB Pulse Width (Second Changing Timing) Outputs a pulse of the duration of the value of TCON_STVB_VW (1/2fH cycles).

Note: This register is updated when TCON\_VEN in TCON\_UPDATE is 1.

## 37.2.37 TCON Vertical Timing Setting Register B2 (TCON\_TIM\_STVB2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	TCON_STVB_INV	—	TCON_STVB_SEL[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	TCON_STVB_INV	0	R/W	Polarity Inversion Control of STVB Signal 0: Not inverted 1: Inverted
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2 to 0	TCON_STVB_SEL [2:0]	1	R/W	Output Signal Select for LCD_TCON1 Pin 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE

Note: This register is updated when TCON\_VEN in TCON\_UPDATE is 1.

## 37.2.38 TCON Horizontal Timing Setting Register STH1 (TCON\_TIM\_STH1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	TCON_STH_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	TCON_STH_HW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	TCON_STH_HS [10:0]	0	R/W	STH Signal Pulse Start Position (First Changing Timing) Starts pulse output after the time specified by the value of TCON_STH_HS + 1 from the rising edge of the Hsync signal (clock cycles).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	TCON_STH_HW [10:0]	96	R/W	STH Pulse Width (Second Changing Timing) Outputs a pulse of the duration of the value of TCON_STH_HW (clock cycles).

Note: This register is updated when TCON\_VEN in TCON\_UPDATE is 1.

## 37.2.39 TCON Horizontal Timing Setting Register STH2 (TCON\_TIM\_STH2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TCON_STH_HS_SEL	—	—	—	TCON_STH_INV	—	TCON_STH_SEL[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	TCON_STH_HS_SEL	0	R/W	STH Signal Operating Reference Select 0: Hsync signal reference 1: Offset Hsync signal reference
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	TCON_STH_INV	1	R/W	Polarity Inversion Control of STH Signal 0: Not inverted 1: Inverted
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2 to 0	TCON_STH_SEL [2:0]	2	R/W	Output Signal Select for LCD_TCON2 Pin 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE

Note: This register is updated when TCON\_VEN in TCON\_UPDATE is 1.

## 37.2.40 TCON Horizontal Timing Setting Register STB1 (TCON\_TIM\_STB1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	TCON_STB_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	TCON_STB_HW[10:0]										
Initial value:	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	TCON_STB_HS [10:0]	144	R/W	STB Signal Pulse Start Position (First Changing Timing) Starts pulse output after the time specified by the value of TCON_STB_HS + 1 from the rising edge of the Hsync signal (clock cycles).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	TCON_STB_HW [10:0]	640	R/W	STB Pulse Width (Second Changing Timing) Outputs a pulse of the duration of the value of TCON_STB_HW (clock cycles).

Note: This register is updated when TCON\_VEN in TCON\_UPDATE is 1.

## 37.2.41 TCON Horizontal Timing Setting Register STB2 (TCON\_TIM\_STB2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TCON_STB_HS_SEL	—	—	—	TCON_STB_INV	—	TCON_STB_SEL[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	TCON_STB_HS_SEL	0	R/W	STB Signal Operating Reference Select 0: Hsync signal reference 1: Offset Hsync signal reference
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	TCON_STB_INV	0	R/W	Polarity Inversion Control of STB Signal 0: Not inverted 1: Inverted
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2 to 0	TCON_STB_SEL [2:0]	7	R/W	Output Signal Select for LCD_TCON3 Pin 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE

Note: This register is updated when TCON\_VEN in TCON\_UPDATE is 1.

## 37.2.42 TCON Horizontal Timing Setting Register CPV1 (TCON\_TIM\_CPV1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	TCON_CPV_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	TCON_CPV_HW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	TCON_CPV_HS [10:0]	0	R/W	CPV Signal Pulse Start Position (First Changing Timing) Starts pulse output after the time specified by the value of TCON_CPV_HS + 1 from the rising edge of the Hsync signal (clock cycles).
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	TCON_CPV_HW [10:0]	0	R/W	CPV Pulse Width (Second Changing Timing) Outputs a pulse of the duration of the value of TCON_CPV_HW (clock cycles).

Note: This register is updated when TCON\_VEN in TCON\_UPDATE is 1.



## 37.2.43 TCON Horizontal Timing Setting Register CPV2 (TCON\_TIM\_CPV2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TCON CPV_HS_ SEL	—	—	—	TCON CPV_INV	—	TCON_CPV_SEL[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	TCON_CPV_HS_SEL	0	R/W	CPV Signal Operating Reference Select 0: Hsync signal reference 1: Offset Hsync signal reference
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	TCON_CPV_INV	0	R/W	Polarity Inversion Control of CPV Signal 0: Not inverted 1: Inverted
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2 to 0	TCON_CPV_SEL [2:0]	4	R/W	Output Signal Select for LCD_TCON4 Pin 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE

Note: This register is updated when TCON\_VEN in TCON\_UPDATE is 1.

## 37.2.44 TCON Horizontal Timing Setting Register POLA1 (TCON\_TIM\_POLA1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	TCON_POLA_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	TCON_POLA_HW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	TCON_POLA_HS [10:0]	0	R/W	POLA Signal Pulse Start Position (First Changing Timing) Starts pulse output after the time specified by the value of TCON_POLA_HS from the rising edge of the Hsync signal (clock cycles). Note: When 1 × 1, 1 × 2, or 2 × 2 reverse mode is selected, these bits should be set to 1 or greater.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	TCON_POLA_HW [10:0]	0	R/W	POLA Pulse Width (Second Changing Timing) Outputs a pulse of the duration of the value of TCON_POLA_HW (clock cycles).

Note: This register is updated when TCON\_VEN in TCON\_UPDATE is 1.

## 37.2.45 TCON Horizontal Timing Setting Register POLA2 (TCON\_TIM\_POLA2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	TCON_POLA_MD[1:0]	—	—	—	TCON_POLA_HS_SEL	—	—	—	TCON_POLA_INV	—	TCON_POLA_SEL[2:0]			
Initial value:	0	0	0	1	0	0	0	0	0	0	0	0	0	1	0	1
R/W:	R	R	R/W	R/W	R	R	R	R/W	R	R	R	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13, 12	TCON_POLA_MD [1:0]	1	R/W	POLA Signal Generation Mode Select 0: Normal mode Generates the signal that changes twice a horizontal period. 1: 1 × 1 reverse mode Generates the signal whose polarity is inverted every horizontal period. 2: 1 × 2 reverse mode Generates the signal whose polarity is inverted in the first horizontal period and is subsequently inverted every two horizontal periods. 3: 2 × 2 reverse mode Generates the signal whose polarity is inverted every two horizontal periods.
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	TCON_POLA_HS_SEL	0	R/W	POLA Signal Operating Reference Select 0: Hsync signal reference 1: Offset Hsync signal reference
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	TCON_POLA_INV	0	R/W	Polarity Inversion Control of POLA Signal 0: Not inverted 1: Inverted
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2 to 0	TCON_POLA_SEL [2:0]	5	R/W	Output Signal Select for LCD_TCON5 pin 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE

Note: This register is updated when TCON\_VEN in TCON\_UPDATE is 1.

## 37.2.46 TCON Horizontal Timing Setting Register POLB1 (TCON\_TIM\_POLB1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	TCON_POLB_HS[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	TCON_POLB_HW[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	TCON_POLB_HS [10:0]	0	R/W	POLB Signal Pulse Start Position (First Changing Timing) Starts pulse output after the time specified by the value of TCON_POLB_HS from the rising edge of the Hsync signal (clock cycles). Note: When 1 × 1, 1 × 2, or 2 × 2 reverse mode is selected, these bits should be set to 1 or greater.
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	TCON_POLB_HW [10:0]	0	R/W	POLB Pulse Width (Second Changing Timing) Outputs a pulse of the duration of the value of TCON_POLB_HW (clock cycles).

Note: This register is updated when TCON\_VEN in TCON\_UPDATE is 1.

## 37.2.47 TCON Horizontal Timing Setting Register POLB2 (TCON\_TIM\_POLB2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	TCON_POLB_MD[1:0]	—	—	—	TCON_POLB_HS_SEL	—	—	—	TCON_POLB_INV	—	TCON_POLB_SEL[2:0]	—	—	—
Initial value:	0	0	0	1	0	0	0	0	0	0	0	0	0	1	1	0
R/W:	R	R	R/W	R/W	R	R	R	R/W	R	R	R	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13, 12	TCON_POLB_MD [1:0]	1	R/W	POLB Signal Generation Mode Select 0: Normal mode Generates the signal that changes twice a horizontal period. 1: 1 × 1 reverse mode Generates the signal whose polarity is inverted every horizontal period. 2: 1 × 2 reverse mode Generates the signal whose polarity is inverted in the first horizontal period and is subsequently inverted every two horizontal periods. 3: 2 × 2 reverse mode Generates the signal whose polarity is inverted every two horizontal periods.
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	TCON_POLB_HS_SEL	0	R/W	POLB Signal Operating Reference Select 0: Hsync signal reference 1: Offset Hsync signal reference
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	TCON_POLB_INV	0	R/W	Polarity Inversion Control of POLB Signal 0: Not inverted 1: Inverted
3	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
2 to 0	TCON_POLB_SEL [2:0]	6	R/W	Output Signal Select for LCD_TCON6 pin 0: STVA/VS 1: STVB/VE 2: STH/SP/HS 3: STB/LP/HE 4: CPV/GCK 5: POLA 6: POLB 7: DE

Note: This register is updated when TCON\_VEN in TCON\_UPDATE is 1.

## 37.2.48 TCON Data Enable Polarity Setting Register (TCON\_TIM\_DE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TCON_DE_INV
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	TCON_DE_INV	0	R/W	Polarity Inversion Control of DE Signal 0: Not inverted 1: Inverted

Note: This register is updated when TCON\_VEN in TCON\_UPDATE is 1.

## 37.2.49 Register Update Control Register in Output Controller (OUT\_UPDATE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	OUTCNT_VEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/WC1

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	OUTCNT_VEN	0	R/WC1	Brightness/Contrast, Dither Process, Output Interface Register Update 0: Registers are not updated. 1: Registers are updated at the rising edge of the Vsync.

## 37.2.50 Output Interface Register (OUT\_SET)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	OUT_ENDI- AN_ON	—	—	—	OUT_SWAF- ON	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	OUT- FORMAT[1:0]	—	—	OUT_FRQ- SEL[1:0]	—	—	—	—	—	OUT- DIR- SEL	—	—	—	OUT_PHASE[1:0]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	OUT_ENDI- AN_ON	0	R/W	Bit Endian Change On/Off Control 0: Off 1: On
27 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	OUT_SWAP- ON	0	R/W	B/R Signal Swap On/Off Control 0: Off 1: On
23 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20 to 16	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13, 12	OUT_FORMAT[1:0]	0	R/W	Output Format Select 0: RGB888 1: RGB666 2: RGB565 3: Serial RGB
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	OUT_FRQ- SEL [1:0]	0	R/W	Clock Frequency Control 0: 100% speed — (parallel RGB) 1: Triple speed — (serial RGB) 2: Quadruple speed — (serial RGB) 3: Setting prohibited
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	OUT_DIR- SEL	0	R/W	Scan Direction Select 0: Forward scan 1: Reverse scan
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1, 0	OUT_PHASE[1:0]	0	R/W	Clock Phase Adjustment During Serial RGB Output Triple speed mode 0: 0 (clk) 1: 1 (clk) 2: 2 (clk) 3: Setting prohibited Quadruple speed mode 0: 0 (clk) 1: 1 (clk) 2: 2 (clk) 3: 3 (clk)

Note: This register is updated when OUTCNT\_VEN in OUT\_UPDATE is 1.

### 37.2.51 Brightness (DC) Correction Register 1 (OUT\_BRIGHT1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	—	—	—	—	—	—	PBRT_G[9:0]										
Initial value:	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	PBRT_G[9:0]	512	R/W	Brightness (DC) Adjustment of G Signal Unsigned (0 (-512) to 512 (0) to 1023 (+511) [LSB], 512 [LSB] with offset)

Note: This register is updated when OUTCNT\_VEN in OUT\_UPDATE is 1.



## 37.2.52 Brightness (DC) Correction Register 2 (OUT\_BRIGTH2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	PBRT_B[9:0]									
Initial value:	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PBRT_R[9:0]									
Initial value:	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25 to 16	PBRT_B[9:0]	512	R/W	Brightness (DC) Adjustment of B Signal Unsigned (0 (-512) to 512 (0) to 1023 (+511) [LSB], 512 [LSB] with offset)
15 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	PBRT_R[9:0]	512	R/W	Brightness (DC) Adjustment of R Signal Unsigned (0 (-512) to 512 (0) to 1023 (+511) [LSB], 512 [LSB] with offset)

Note: This register is updated when OUTCNT\_VEN in OUT\_UPDATE is 1.

## 37.2.53 Contrast (Gain) Correction Register (OUT\_CONTRAST)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	CONT_G[7:0]							
Initial value:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CONT_B[7:0]							CONT_R[7:0]								
Initial value:	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	CONT_G[7:0]	128	R/W	Contrast (Gain) Adjustment of G Signal 0/128 to 255/128 (approx.2 times)
15 to 8	CONT_B[7:0]	128	R/W	Contrast (Gain) Adjustment of B Signal 0/128 to 255/128 (approx.2 times)
7 to 0	CONT_R[7:0]	128	R/W	Contrast (Gain) Adjustment of R Signal 0/128 to 255/128 (approx.2 times)

Note: This register is updated when OUTCNT\_VEN in OUT\_UPDATE is 1.

## 37.2.54 Panel Dither Register (OUT\_PDTHA)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	PDTH_SEL[1:0]	—	—	PDTH_FORMAT[1:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	PDTH_PA[1:0]	—	—	PDTH_PB[1:0]	—	—	PDTH_PC[1:0]	—	—	PDTH_PD[1:0]				
Initial value:	0	0	1	1	0	0	0	0	0	0	1	0	0	0	0	1
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
21, 20	PDTH_SEL[1:0]	0	R/W	Panel Dither Operation Mode 0: Truncate 1: Round-off 2: 2 × 2 pattern dither 3: Random pattern dither
19, 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17, 16	PDTH_FORMAT [1:0]	0	R/W	Panel Dither Output Format Select 0: RGB888 1: RGB666 2: RGB565 3: Setting prohibited
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13, 12	PDTH_PA[1:0]	3	R/W	Pattern Value (A) of 2 × 2 Pattern Dither Unsigned (0 to 3 [LSB])
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	PDTH_PB[1:0]	0	R/W	Pattern Value (B) of 2 × 2 Pattern Dither Unsigned (0 to 3 [LSB])
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5, 4	PDTH_PC[1:0]	2	R/W	Pattern Value (C) of 2 × 2 Pattern Dither Unsigned (0 to 3 [LSB])
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	PDTH_PD[1:0]	1	R/W	Pattern Value (D) of 2 × 2 Pattern Dither Unsigned (0 to 3 [LSB])

Note: This register is updated when OUTCNT\_VEN in OUT\_UPDATE is 1.

## 37.2.55 Output Phase Control Register (OUT\_CLK\_PHASE)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	OUTCNT_	—	—	—	OUTCNT_	—	OUTCNT_	OUTCNT_	OUTCNT_	OUTCNT_	OUTCNT_	OUTCNT_	OUTCNT_
				FRONT_				LCD_		STVA_	STVB_	STH_	STB_	CPV_	POLA_	POLB_
				GAM				EDGE		EDGE	EDGE	EDGE	EDGE	EDGE	EDGE	EDGE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	OUTCNT_	0	R/W	Correction Circuit Sequence Control 0: Brightness → contrast → gamma correction 1: Gamma correction → brightness → contrast
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	OUTCNT_	0	R/W	Output Phase Control of LCD_DATA23 to LCD_DATA0 Pin 0: Output at the rising edge of LCD_CLK pin 1: Output at the falling edge of LCD_CLK pin
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	OUTCNT_	0	R/W	Output Phase Control of LCD_TCON0 Pin 0: Output at the rising edge of LCD_CLK pin 1: Output at the falling edge of LCD_CLK pin
5	OUTCNT_	0	R/W	Output Phase Control of LCD_TCON1 Pin 0: Output at the rising edge of LCD_CLK pin 1: Output at the falling edge of LCD_CLK pin
4	OUTCNT_	0	R/W	Output Phase Control of LCD_TCON2 Pin 0: Output at the rising edge of LCD_CLK pin 1: Output at the falling edge of LCD_CLK pin
3	OUTCNT_	0	R/W	Output Phase Control of LCD_TCON3 Pin 0: Output at the rising edge of LCD_CLK pin 1: Output at the falling edge of LCD_CLK pin
2	OUTCNT_	0	R/W	Output Phase Control of LCD_TCON4 Pin 0: Output at the rising edge of LCD_CLK pin 1: Output at the falling edge of LCD_CLK pin
1	OUTCNT_	0	R/W	Output Phase Control of LCD_TCON5 Pin 0: Output at the rising edge of LCD_CLK pin 1: Output at the falling edge of LCD_CLK pin
0	OUTCNT_	0	R/W	Output Phase Control of LCD_TCON6 Pin 0: Output at the rising edge of LCD_CLK pin 1: Output at the falling edge of LCD_CLK pin

Note: This register is updated when OUTCNT\_VEN in OUT\_UPDATE is 1.

### 37.3 Usage Methods

#### 37.3.1 Gamma Correction Adjustment Method

The characteristics of G, B, R of each panel to be connected should be measured and gamma correction should be made to suit the panel.

Since the gamma correction adjustment depends on to the characteristics of the panel, there are no recommended setting values.

#### 37.3.2 Dither Usage Method

Dither is used when pseudo contour is appeared on the display screen.

**Table 37.29 Dither Settings**

Bit Name	Setting Value
PDTH_FORMAT[1:0]	Selects the format. For RGB888: 0 For RGB666: 1 For RGB565: 2
PDTH_SEL[1:0]	When 2 × 2 pattern dither is to be used: 2
PDTH_PA[1:0]	Normally 3 (initial value)
PDTH_PB[1:0]	Normally 0 (initial value)
PDTH_PC[1:0]	Normally 2 (initial value)
PDTH_PD[1:0]	Normally 1 (initial value)

### 37.3.3 Output Format Adjustment Method

The setting example of the typical output format is shown in Table 37.30 and Table 37.31.

It is necessary to carry out the setting of synchronization system signals of each output format similarly as the setting of output after scaling.

**Table 37.30 Setting Example of Synchronizing Signal**

Register Name	Bit Name	VGA	SVGA	Description
TCON_TIM	TCON_HALF[10:0]	400	528	Sets the half value of 1H time period in clock units.
<b>Vsync signal</b>				
TCON_TIM_STVA1	TCON_STVA_VS[10:0]	0	0	Sets the pulse generation start position from the rising edge of the internal Vsync signal. A 1/2H time period is set as 1.
TCON_TIM_STVA1	TCON_STVA_VW[10:0]	4	8	Sets the changing point from the above pulse generation start position. A 1/2H time period is set as 1.
TCON_TIM_STVA2	TCON_STVA_INV	0	0	Sets the output polarity of the above pulse. For inverted output: 1
TCON_TIM_STVA2	TCON_STVA_SEL[2:0]	0	0	For STVA output selection: 0
<b>Vertical enable signal</b>				
TCON_TIM_STVB1	TCON_STVB_VS[10:0]	68	44	Sets the pulse generation start position from the rising edge of the internal Vsync signal. A 1/2H time period is set as 1.
TCON_TIM_STVB1	TCON_STVB_VW[10:0]	960	1200	Sets the changing point from the above pulse generation start position. A 1/2H time period is set as 1.
TCON_TIM_STVB2	TCON_STVB_INV	0	0	Sets the output polarity of the above pulse. For inverted output: 1
TCON_TIM_STVB2	TCON_STVB_SEL[2:0]	1	1	For STVB output selection: 1
<b>Register Name</b>				
<b>Bit Name</b>				
<b>VGA</b>				
<b>SVGA</b>				
<b>Description</b>				
<b>Hsync signal</b>				
TCON_TIM_STH1	TCON_STH_HS[10:0]	0	0	Sets the pulse generation start position from the rising edge of the internal Hsync signal.
TCON_TIM_STH1	TCON_STH_HW[10:0]	96	128	Sets the changing point from the above pulse generation start position.
TCON_TIM_STH2	TCON_STH_INV	0	0	Sets the output polarity of the above pulse. For inverted output: 1
TCON_TIM_STH2	TCON_STH_SEL[2:0]	2	2	For STH output selection: 2
<b>Horizontal enable signal</b>				
TCON_TIM_STB1	TCON_STB_HS[10:0]	128	192	Sets the pulse generation start position from the rising edge of the internal Hsync signal.
TCON_TIM_STB1	TCON_STB_HW[10:0]	640	800	Sets the changing point from the above pulse generation start position.
TCON_TIM_STB2	TCON_STB_INV	0	0	Sets the output polarity of the above pulse. For inverted output: 1
TCON_TIM_STB2	TCON_STB_SEL[2:0]	3	3	For STB output selection: 3

**Table 37.31 Setting Example of Data System**

Register Name	Bit Name	RGB888	RGB ()	Description
OUT_SET	OUT_ENDIAN_ON	0	0	When bit endian is changed: 1
OUT_SET	OUT_SWAP_ON	0	0	When B/R is to be swapped: 1
OUT_SET	OUT_PIXEL_INV_ON	0	0	When the function of reducing number of simultaneous changes is to be used: 1
OUT_SET	OUT_SUM_MOVE[4:0]	31	31	When OUT_PIXEL_INV_ON = 1, this register becomes valid. Sets the simultaneous changing threshold.
OUT_SET	OUT_FORMAT[1:0]	0	3	Sets the output format. For RGB888: 0 For RGB666: 1 For RGB565: 2 For serial RGB: 3
OUT_SET	OUT_FRQ_SEL[1:0]	0	1	Sets the output clock. For RGB888, RGB666, RGB565: 0 For triple speed serial RGB output: 1 For quadruple serial RGB output: 2
OUT_SET	OUT_DIR_SEL	0	0	When data arrangement of the serial RGB output is to be reversed: 1
OUT_SET	OUT_PHASE[1:0]	0	0	Sets when output phase of the serial RGB is shifted. Not delayed Delayed by one clock cycle: 1 Delayed by two clock cycles: 2 Delayed by three clock cycles: 3 (applicable only for quadruple speed mode)

## 38. Video Display Controller 5 (8):System Controller

### 38.1 System Controller

#### 38.1.1 Overview of Functions

The system controller provides interrupt control, panel clock control, CLUT table read select signal status flag output functions.

#### 38.1.2 Interrupt Control

Twenty three interrupt signals are output from the scaler and image synthesizer. The system controller controls whether to output these interrupt signals.

One is written to the corresponding INT\_STA\* bit when an interrupt signal is to be accepted. After 1 has been written to the bit, however, its value is still read out as 0 until the interrupt signal is accepted. Once the interrupt signal has been accepted, the INT\_STA\* bit is read out as 1.

The bit for an accepted interrupt signal is cleared by writing 0 to the INT\_STA\* bit. If further interrupt signals are to be accepted after the INT\_STA\* bit has been cleared, 1 is again written to the bit.

**Table 38.1 Interrupt Signals**

Request Source Name	Bit Name	Function
S0_VI_VSYNC	INT_STA 0	Vsync signal input to scaler 0
S0_LO_VSYNC	INT_STA 1	Vsync signal output from scaler 0
S0_VSYNCERR	INT_STA 2	Missing Vsync signal for scaler 0
GR3_VLINE	INT_STA 3	Specified line signal for panel output in graphics 3
S0_VFIELD	INT_STA 4	Field end signal for recording function in scaler 0
IV1_VBUFERR	INT_STA 5	Frame buffer write overflow signal for scaler 0
IV3_VBUFERR	INT_STA 6	Frame buffer read underflow signal for graphics 0
IV5_VBUFERR	INT_STA 7	Frame buffer read underflow signal for graphics 2
IV6_VBUFERR	INT_STA 8	Frame buffer read underflow signal for graphics 3
S0_WLINE	INT_STA 9	Write specification line signal input to scaling-down control block in scaler 0
S1_VI_VSYNC	INT_STA 10	Vsync signal input to scaler 1
S1_LO_VSYNC	INT_STA 11	Vsync signal output from scaler 1
S1_VSYNCERR	INT_STA 12	Missing Vsync signal for scaler 1
S1_VFIELD	INT_STA 13	Field end signal for recording function in scaler 1
IV2_VBUFERR	INT_STA 14	Frame buffer write overflow signal for scaler 1
IV4_VBUFERR	INT_STA 15	Frame buffer read underflow signal for graphics 1
S1_WLINE	INT_STA 16	Write specification line signal input to scaling-down control block in scaler 1
OIR_VI_VSYNC	INT_STA 17	Vsync signal input to output image generator
OIR_LO_VSYNC	INT_STA 18	Vsync signal output from output image generator
OIR_VLINE	INT_STA 19	Specified line signal for panel output in output image generator
OIR_VFIELD	INT_STA 20	Field end signal for recording function in output image generator
IV7_VBUFERR	INT_STA 21	Frame buffer write overflow signal for output image generator
IV8_VBUFERR	INT_STA 22	Frame buffer read underflow signal for graphics (OIR)

**Table 38.2** Interrupt Clear/Hold Settings

Register Name	Bit Name	Initial Value	Description
SYSCNT_INT1	INT_STA0	0	S0_VI_VSYNC Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
SYSCNT_INT1	INT_STA1	0	S0_LO_VSYNC Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
SYSCNT_INT1	INT_STA2	0	S0_VSYNCERR Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
SYSCNT_INT1	INT_STA3	0	GR3_VLINE Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
SYSCNT_INT1	INT_STA4	0	S0_VFIELD Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
SYSCNT_INT1	INT_STA5	0	IV1_VBUFERR Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
SYSCNT_INT1	INT_STA6	0	IV3_VBUFERR Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
SYSCNT_INT1	INT_STA7	0	IV5_VBUFERR Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
SYSCNT_INT2	INT_STA8	0	IV6_VBUFERR Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
SYSCNT_INT2	INT_STA9	0	S0_WLINE Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
SYSCNT_INT2	INT_STA10	0	S1_VI_VSYNC Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
SYSCNT_INT2	INT_STA11	0	S1_LO_VSYNC Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.



**Table 38.2** Interrupt Clear/Hold Settings

Register Name	Bit Name	Initial Value	Description
SYSCNT_INT2	INT_STA12	0	S1_VSYNCERR Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
SYSCNT_INT2	INT_STA13	0	S1_VFIELD Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
SYSCNT_INT2	INT_STA14	0	IV2_VBUFERR Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
SYSCNT_INT2	INT_STA15	0	IV4_VBUFERR Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
SYSCNT_INT3	INT_STA16	0	S1_WLINE Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
SYSCNT_INT3	INT_STA17	0	OIR_VI_VSYNC Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
SYSCNT_INT3	INT_STA18	0	OIR_LO_VSYNC Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
SYSCNT_INT3	INT_STA19	0	OIR_VLINE Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
SYSCNT_INT3	INT_STA20	0	OIR_VFIELD Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
SYSCNT_INT3	INT_STA21	0	IV7_VBUFERR Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
SYSCNT_INT3	INT_STA22	0	IV8_VBUFERR Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.

**Table 38.3 Interrupt Output On/Off Settings**

Register Name	Bit Name	Initial Value	Description
SYSCNT_INT4	INT_OUT0_ON	0	S0_VI_VSYNC Interrupt Output On/Off 0: Off 1: On
SYSCNT_INT4	INT_OUT1_ON	0	S0_LO_VSYNC Interrupt Output On/Off 0: Off 1: On
SYSCNT_INT4	INT_OUT2_ON	0	S0_VSYNCERR Interrupt Output On/Off 0: Off 1: On
SYSCNT_INT4	INT_OUT3_ON	0	GR3_VLINE Interrupt Output On/Off 0: Off 1: On
SYSCNT_INT4	INT_OUT4_ON	0	S0_VFIELD Interrupt Output On/Off 0: Off 1: On
SYSCNT_INT4	INT_OUT5_ON	0	IV1_VBUFERR Interrupt Output On/Off 0: Off 1: On
SYSCNT_INT4	INT_OUT6_ON	0	IV3_VBUFERR Interrupt Output On/Off 0: Off 1: On
SYSCNT_INT4	INT_OUT7_ON	0	IV5_VBUFERR Interrupt Output On/Off 0: Off 1: On
SYSCNT_INT5	INT_OUT8_ON	0	IV6_VBUFERR Interrupt Output On/Off 0: Off 1: On
SYSCNT_INT5	INT_OUT9_ON	0	S0_WLINE Interrupt Output On/Off 0: Off 1: On
SYSCNT_INT5	INT_OUT10_ON	0	S1_VI_VSYNC Interrupt Output On/Off 0: Off 1: On
SYSCNT_INT5	INT_OUT11_ON	0	S1_LO_VSYNC Interrupt Output On/Off 0: Off 1: On
SYSCNT_INT5	INT_OUT12_ON	0	S1_VSYNCERR Interrupt Output On/Off 0: Off 1: On
SYSCNT_INT5	INT_OUT13_ON	0	S1_VFIELD Interrupt Output On/Off 0: Off 1: On
SYSCNT_INT5	INT_OUT14_ON	0	IV2_VBUFERR Interrupt Output On/Off 0: Off 1: On
SYSCNT_INT5	INT_OUT15_ON	0	IV4_VBUFERR Interrupt Output On/Off 0: Off 1: On
SYSCNT_INT6	INT_OUT16_ON	0	S1_WLINE Interrupt Output On/Off 0: Off 1: On
SYSCNT_INT6	INT_OUT17_ON	0	OIR_VI_VSYNC Interrupt Output On/Off 0: Off 1: On

**Table 38.3 Interrupt Output On/Off Settings**

Register Name	Bit Name	Initial Value	Description
SYSCNT_INT6	INT_OUT18_ON	0	OIR_LO_VSYNC Interrupt Output On/Off 0: Off 1: On
SYSCNT_INT6	INT_OUT19_ON	0	OIR_VLINE Interrupt Output On/Off 0: Off 1: On
SYSCNT_INT6	INT_OUT20_ON	0	OIR_VFIELD Interrupt Output On/Off 0: Off 1: On
SYSCNT_INT6	INT_OUT21_ON	0	IV7_VBUFERR Interrupt Output On/Off 0: Off 1: On
SYSCNT_INT6	INT_OUT22_ON	0	IV8_VBUFERR Interrupt Output On/Off 0: Off 1: On

### 38.1.3 Panel Clock Control

The video image clock, external clock, or peripheral clock 1 is selectable as the source of the panel clock for supply to this module. The module is also equipped with a divider for scaling the frequency by factors from 1/1 to 1/32. The bits listed in Table 38.4 control the panel clock.

**Table 38.4 Panel Clock Control**

Register Name	Bit Name	Initial Value	Description
SYSCNT_PANEL_CLK	PANEL_IKSEL [1:0]	0	Divided Clock Source Select 0: Video image clock (this is VIDEO_X1 if INP_SEL = 0 and DV_CLK if INP_SEL = 1) 1: External clock (LCD0_EXTCLK) 2: External clock (LCD1_EXTCLK) 3: Peripheral clock 1 (P1φ)
SYSCNT_PANEL_CLK	PANEL_OCKSEL[1:0]	0	Panel Clock Output Select 0: Divided clock generated from clock source selected by PANEL_CLK[1:0] 1: LVDS PLL clock 2: LVDS PLL clock divided by 7 3: Setting prohibited
SYSCNT_PANEL_CLK	PANEL_ICKEN	0	Panel Clock Operation Enable 0: Disables operation of the panel clock operation block. 1: Enables operation of the panel clock operation block. Note: Set this bit to 0 before changing the value of the PANEL_IKSEL, PANEL_OCKSEL, or PANEL_DCDR bits.*
SYSCNT_PANEL_CLK	PANEL_DCDR [5:0]	1	Clock Frequency Division Ratio Refer to Table 38.5 for details on the settings. Note: Settings other than those in Table 38.5 are prohibited.

Note: \* Set this bit to 0 before changing the value of the LVDS\_CLK\_EN bit in LVDS\_CLKSELR.

**Table 38.5 I/O Clock Frequency and Divisors**

DCDR[5:0]	Clock Ratio	I/O Clock Frequency (MHz)		
		27.00	54.00	66.67
000001	1/1	27.00	54.00	66.67
000010	1/2	13.50	27.00	33.33
000011	1/3	9.00	18.00	22.22
000100	1/4	6.75	13.50	16.67
000101	1/5	5.40	10.80	13.33
000110	1/6	4.50	9.00	11.11
000111	1/7	3.86	7.71	9.52
001000	1/8	3.38	6.75	8.33
001001	1/9	3.00	6.00	7.41
001100	1/12	2.25	4.50	5.56
010000	1/16	1.69	3.38	4.17
011000	1/24	1.13	2.25	2.78
100000	1/32	0.84	1.69	2.08

### 38.1.4 CLUT Table Read Select Signal Status Flag

The CLUT read select signal status can be read using the flags shown in Table 38.6.

**Table 38.6 CLUT Table Read Select Signal Status Flags**

Register Name	Bit Name	Initial Value	Description
SYSCNT_CLUT	GR0_CLT_SEL_ST	—	Graphics 0 CLUT Table Read Select Signal Status Flag 0: CLUT table 0 is read out. 1: CLUT table 1 is read out.
SYSCNT_CLUT	GR1_CLT_SEL_ST	—	Graphics 1 CLUT Table Read Select Signal Status Flag 0: CLUT table 0 is read out. 1: CLUT table 1 is read out.
SYSCNT_CLUT	GR2_CLT_SEL_ST	—	Graphics 2 CLUT Table Read Select Signal Status Flag 0: CLUT table 0 is read out. 1: CLUT table 1 is read out.
SYSCNT_CLUT	GR3_CLT_SEL_ST	—	Graphics 3 CLUT Table Read Select Signal Status Flag 0: CLUT table 0 is read out. 1: CLUT table 1 is read out.
SYSCNT_CLUT	GR_OIR_CLT_SEL_ST	—	Graphics OIR CLUT Table Read Select Signal Status Flag 0: CLUT table 0 is read out. 1: CLUT table 1 is read out.

## 38.2 Register Descriptions

The following register sets are allocated in the SH register map area.

- Symbols used in Register Descriptions

Initial value: Register value after a reset

—: Undefined value

R/W: Readable/writable. The written value can be read.

R/WC0: Readable/writable. Writing 0 initializes the bit. Writing 1 is ignored.

R/WC1: Readable/writable. Writing 1 initializes the bit. Writing 0 is ignored.

R: Read-only. The write value should always be 0.

—/W: Write-only. The read value is undefined.

**Table 38.7 Register Configuration of System Controller (CH0)**

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Interrupt control register 1	SYSCNT_INT1	R/W	H'0000 0000	H'FCFF 7A80	32
Interrupt control register 2	SYSCNT_INT2	R/W	H'0000 0000	H'FCFF 7A84	32
Interrupt control register 3	SYSCNT_INT3	R/W	H'0000 0000	H'FCFF 7A88	32
Interrupt control register 4	SYSCNT_INT4	R/W	H'0000 0000	H'FCFF 7A8C	32
Interrupt control register 5	SYSCNT_INT5	R/W	H'0000 0000	H'FCFF 7A90	32
Interrupt control register 6	SYSCNT_INT6	R/W	H'0000 0000	H'FCFF 7A94	32
Panel clock control register	SYSCNT_PANEL_CLK	R/W	H'0001	H'FCFF 7A98	16
CLUT table read select signal status register	SYSCNT_CLUT	R	H'0000	H'FCFF 7A9A	16

**Table 38.8 Register Configuration of System Controller (CH1)**

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Interrupt control register 1	SYSCNT_INT1	R/W	H'0000 0000	H'FCFF 9A80	32
Interrupt control register 2	SYSCNT_INT2	R/W	H'0000 0000	H'FCFF 9A84	32
Interrupt control register 3	SYSCNT_INT3	R/W	H'0000 0000	H'FCFF 9A88	32
Interrupt control register 4	SYSCNT_INT4	R/W	H'0000 0000	H'FCFF 9A8C	32
Interrupt control register 5	SYSCNT_INT5	R/W	H'0000 0000	H'FCFF 9A90	32
Interrupt control register 6	SYSCNT_INT6	R/W	H'0000 0000	H'FCFF 9A94	32
Panel clock control register	SYSCNT_PANEL_CLK	R/W	H'0001	H'FCFF 9A98	16
CLUT table read select signal status register	SYSCNT_CLUT	R	H'0000	H'FCFF 9A9A	16

## 38.2.1 Interrupt Control Register 1 (SYSCNT\_INT1)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	INT_STA7	—	—	—	INT_STA6	—	—	—	INT_STA5	—	—	—	INT_STA4
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	INT_STA3	—	—	—	INT_STA2	—	—	—	INT_STA1	—	—	—	INT_STA0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	INT_STA7	0	R/W	IV5_VBUFERR Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
27 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	INT_STA6	0	R/W	IV3_VBUFERR Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
23 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	INT_STA5	0	R/W	IV1_VBUFERR Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
19 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	INT_STA4	0	R/W	S0_VFIELD Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	INT_STA3	0	R/W	GR3_VLINE Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	INT_STA2	0	R/W	S0_VSYNCERR Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	INT_STA1	0	R/W	S0_LO_VSYNC Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	INT_STA0	0	R/W	S0_VI_VSYNC Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.

## 38.2.2 Interrupt Control Register 2 (SYSCNT\_INT2)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	INT_STA15	—	—	—	INT_STA14	—	—	—	INT_STA13	—	—	—	INT_STA12
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	INT_STA11	—	—	—	INT_STA10	—	—	—	INT_STA9	—	—	—	INT_STA8
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	INT_STA15	0	R/W	IV4_VBUFERR Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
27 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	INT_STA14	0	R/W	IV2_VBUFERR Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
23 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	INT_STA13	0	R/W	S1_VFIELD Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
19 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	INT_STA12	0	R/W	S1_VSYNCERR Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	INT_STA11	0	R/W	S1_LO_VSYNC Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	INT_STA10	0	R/W	S1_VL_VSYNC Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.



Bit	Bit Name	Initial Value	R/W	Description
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	INT_STA9	0	R/W	S0_WLINE Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	INT_STA8	0	R/W	IV6_VBUFERR Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.

## 38.2.3 Interrupt Control Register 3 (SYSCNT\_INT3)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	INT_STA22	—	—	—	INT_STA21	—	—	—	INT_STA20
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	INT_STA19	—	—	—	INT_STA18	—	—	—	INT_STA17	—	—	—	INT_STA16
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	INT_STA22	0	R/W	IV8_VBUFERR Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
23 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	INT_STA21	0	R/W	IV7_VBUFERR Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
19 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	INT_STA20	0	R/W	OIR_VFIELD Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	INT_STA19	0	R/W	OIR_VLINE Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	INT_STA18	0	R/W	OIR_LO_VSYNC Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	INT_STA17	0	R/W	OIR_VI_VSYNC Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

---

Bit	Bit Name	Initial Value	R/W	Description
0	INT_STA16	0	R/W	S1_WLINE Interrupt Clear/Hold 0(W): Clears the interrupt status. 1(W): Starts interrupt acceptance. 0(R): No interrupt has occurred. 1(R): An interrupt has occurred.

---

## 38.2.4 Interrupt Control Register 4 (SYSCNT\_INT4)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	INT_OUT7_ON	—	—	—	INT_OUT6_ON	—	—	—	INT_OUT5_ON	—	—	—	INT_OUT4_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	INT_OUT3_ON	—	—	—	INT_OUT2_ON	—	—	—	INT_OUT1_ON	—	—	—	INT_OUT0_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	INT_OUT7_ON	0	R/W	IV5_VBUFERR Interrupt Output On/Off 0: Off 1: On
27 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	INT_OUT6_ON	0	R/W	IV3_VBUFERR Interrupt Output On/Off 0: Off 1: On
23 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	INT_OUT5_ON	0	R/W	IV1_VBUFERR Interrupt Output On/Off 0: Off 1: On
19 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	INT_OUT4_ON	0	R/W	S0_VFIEELD Interrupt Output On/Off 0: Off 1: On
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	INT_OUT3_ON	0	R/W	GR3_VLINE Interrupt Output On/Off 0: Off 1: On
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	INT_OUT2_ON	0	R/W	S0_VSYNCERR Interrupt Output On/Off 0: Off 1: On
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	INT_OUT1_ON	0	R/W	S0_LO_VSYNC Interrupt Output On/Off 0: Off 1: On
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	INT_OUT0_ON	0	R/W	S0_VI_VSYNC Interrupt Output On/Off 0: Off 1: On

## 38.2.5 Interrupt Control Register 5 (SYSCNT\_INT5)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	INT_OUT15_ON	—	—	—	INT_OUT14_ON	—	—	—	INT_OUT13_ON	—	—	—	INT_OUT12_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	INT_OUT11_ON	—	—	—	INT_OUT10_ON	—	—	—	INT_OUT9_ON	—	—	—	INT_OUT8_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	INT_OUT15_ON	0	R/W	IV4_VBUFERR Interrupt Output On/Off 0: Off 1: On
27 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	INT_OUT14_ON	0	R/W	IV2_VBUFERR Interrupt Output On/Off 0: Off 1: On
23 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	INT_OUT13_ON	0	R/W	S1_VFIELD Interrupt Output On/Off 0: Off 1: On
19 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	INT_OUT12_ON	0	R/W	S1_VSYNCERR Interrupt Output On/Off 0: Off 1: On
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	INT_OUT11_ON	0	R/W	S1_LO_VSYNC Interrupt Output On/Off 0: Off 1: On
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	INT_OUT10_ON	0	R/W	S1_VI_VSYNC Interrupt Output On/Off 0: Off 1: On
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	INT_OUT9_ON	0	R/W	S0_WLINE Interrupt Output On/Off 0: Off 1: On
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	INT_OUT8_ON	0	R/W	IV6_VBUFERR Interrupt Output On/Off 0: Off 1: On

## 38.2.6 Interrupt Control Register 6 (SYSCNT\_INT6)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	INT_OUT22_ON	—	—	—	INT_OUT21_ON	—	—	—	INT_OUT20_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	INT_OUT19_ON	—	—	—	INT_OUT18_ON	—	—	—	INT_OUT17_ON	—	—	—	INT_OUT16_ON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	INT_OUT22_ON	0	R/W	IV8_VBUFERR Interrupt Output On/Off 0: Off 1: On
23 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	INT_OUT21_ON	0	R/W	IV7_VBUFERR Interrupt Output On/Off 0: Off 1: On
19 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	INT_OUT20_ON	0	R/W	OIR_VFIELD Interrupt Output On/Off 0: Off 1: On
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	INT_OUT19_ON	0	R/W	OIR_VLINE Interrupt Output On/Off 0: Off 1: On
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	INT_OUT18_ON	0	R/W	OIR_LO_VSYNC Interrupt Output On/Off 0: Off 1: On
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	INT_OUT17_ON	0	R/W	OIR_VI_VSYNC Interrupt Output On/Off 0: Off 1: On
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	INT_OUT16_ON	0	R/W	S1_WLINE Interrupt Output On/Off 0: Off 1: On

## 38.2.7 Panel Clock Control Register (SYSCNT\_PANEL\_CLK)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	PANEL_ICKSEL [1:0]	PANEL_OCKSEL [1:0]	—	PANEL_ICKEN	—	—	PANEL_DCDR[5:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R/W	R/W	R/W	R/W	R	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13, 12	PANEL_ICKSEL [1:0]	All 0	R/W	Divided Clock Source Select 0: Video image clock (this is VIDEO_X1 if INP_SEL = 0 and DV_CLK if INP_SEL = 1) 1: External clock (LCD0_EXTCLK) 2: External clock (LCD1_EXTCLK) 3: Peripheral clock 1 (P1φ)
11, 10	PANEL_OCKSEL [1:0]	All 0	R/W	Panel Clock Output Select 0: Divided clock generated from clock source selected by PANEL_CLK[1:0] 1: LVDS PLL clock 2: LVDS PLL clock divided by 7 3: Setting prohibited
9	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
8	PANEL_ICKEN	0	R/W	Panel Clock Operation Enable 0: Disables operation of the panel clock operation block. 1: Enables operation of the panel clock operation block. Note: Set this bit to 0 before changing the value of the PANEL_ICKSEL, PANEL_OCKSEL, or PANEL_DCDR bits.*
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5 to 0	PANEL_DCDR[5:0]	1	R/W	Clock Frequency Division Ratio Refer to Table 38.5 for details on the settings. Note: Settings other than those in Table 38.5 are prohibited.

Note: \* Set this bit to 0 before changing the value of the LVDS\_CLK\_EN bit in LVDS\_CLKSELR.

## 38.2.8 CLUT Table Read Select Signal Status Register (SYSCNT\_CLUT)

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	GR_OIR_CLT_SEL_ST	—	GR3_CLT_SEL_ST	—	—	—	GR2_CLT_SEL_ST	—	—	—	GR1_CLT_SEL_ST	—	—	—	GR0_CLT_SEL_ST
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	GR_OIR_CLT_SEL_ST	—	R	Graphics OIR CLUT Table Read Select Signal Status Flag 0: CLUT table 0 is read out. 1: CLUT table 1 is read out.
13	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
12	GR3_CLT_SEL_ST	—	R	Graphics 3 CLUT Table Read Select Signal Status Flag 0: CLUT table 0 is read out. 1: CLUT table 1 is read out.
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	GR2_CLT_SEL_ST	—	R	Graphics 2 CLUT Table Read Select Signal Status Flag 0: CLUT table 0 is read out. 1: CLUT table 1 is read out.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	GR1_CLT_SEL_ST	—	R	Graphics 1 CLUT Table Read Select Signal Status Flag 0: CLUT table 0 is read out. 1: CLUT table 1 is read out.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	GR0_CLT_SEL_ST	—	R	Graphics 0 CLUT Table Read Select Signal Status Flag 0: CLUT table 0 is read out. 1: CLUT table 1 is read out.



## 39. Dynamic Range Compression

For details, contact your local sales representatives.

## 40. LVDS Output Interface

This LSI has a module for providing a low voltage differential signaling (LVDS) output interface. This module converts the RGB signals output from video display controller 5 to the LVDS format that has six bits each for R, G, and B signals, and outputs the converted signals. It has a dedicated phase-locked loop (PLL) circuit, which can generate a clock with a desired frequency by dividing or multiplying the input clock frequency.

The clock output from this PLL can be used as the panel clock for video display controller 5 even when LVDS output is not used. This PLL is called LVDS PLL in this section.

### 40.1 Features

- Four pairs of differential output conforming to the TIA/EIA-644 standard (three pairs for data and one pair for the clock)
- When LVDS output is not used, the LVDS pins can be used as CMOS input/output pins.
- LVDS PLL for generating a clock with a desired frequency
- LVDS PLL power-down function

### 40.2 Block Diagram

Figure 40.1 shows a block diagram of the LVDS output interface configuration.

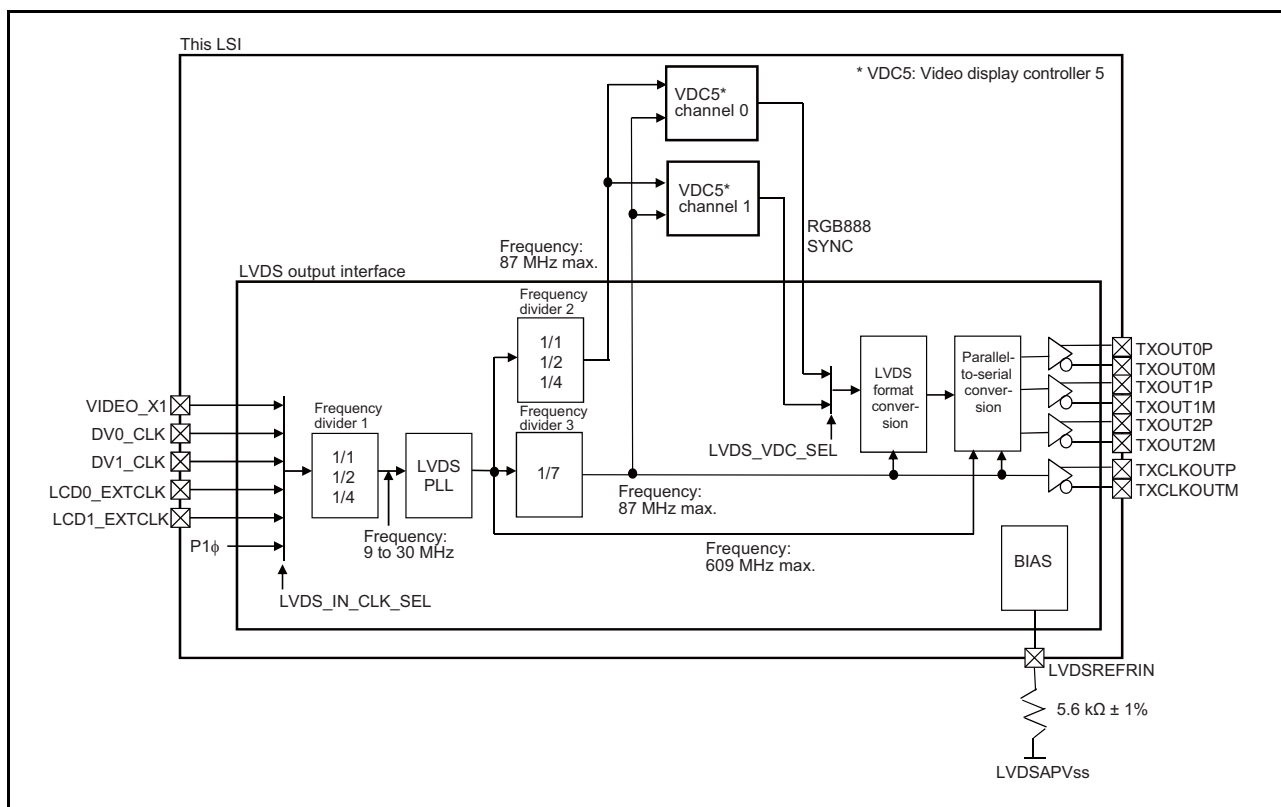
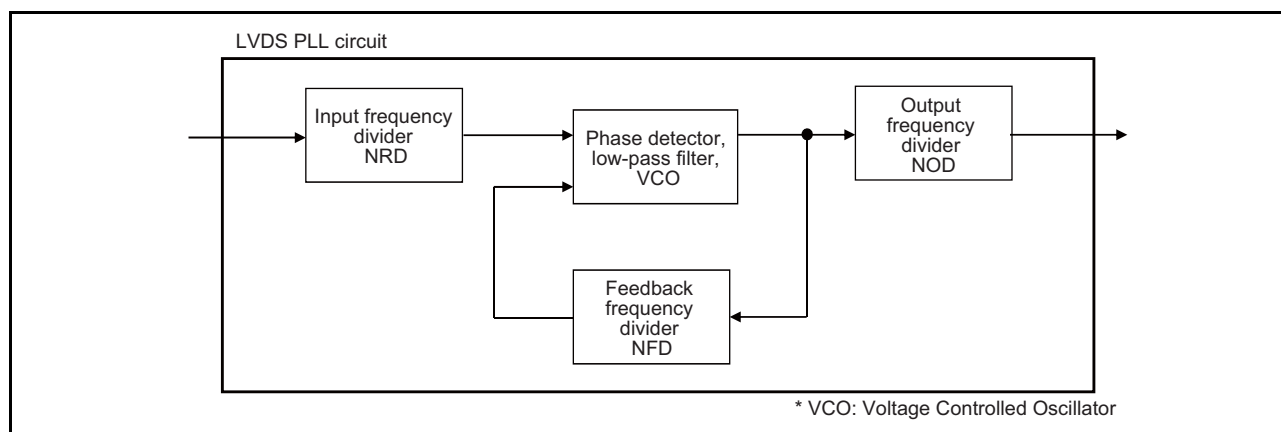


Figure 40.1 Block Diagram of LVDS Output Interface Configuration

Figure 40.2 shows a block diagram of the LVDS PLL circuit.



**Figure 40.2** Block Diagram of LVDS PLL Circuit

### 40.3 Input/Output Pins

Table 40.1 shows the pin configuration.

**Table 40.1** Pin Configuration

Pin Name	Symbol	I/O	Function
LVDS analog power supply pin	LVDSAPVcc	Input	Power supply pin for LVDS output
LVDS analog ground pin*	LVDSAPVss	Input	Ground pin for LVDS output
LVDS PLL power supply pin	LVDSPLLVcc	Input	Power supply pin for LVDS PLL
LVDS resistor connection pin	LVDSREFRIN	Input	Reference resistor connection pin Connect to LVDSAPVss via a resistor with a value of 5.6 kΩ ± 1%.
LVDS data output pin 0P	TXOUT0P	Output	LVDS data output pins
LVDS data output pin 0M	TXOUT0M	Output	
LVDS data output pin 1P	TXOUT1P	Output	LVDS data output pins
LVDS data output pin 1M	TXOUT1M	Output	
LVDS data output pin 2P	TXOUT2P	Output	LVDS data output pins
LVDS data output pin 2M	TXOUT2M	Output	
LVDS clock output pin CP	TXCLKOUTP	Output	LVDS clock output pins
LVDS clock output pin CM	TXCLKOUTM	Output	

Note: \* This pin is not present on products in the BGA package.

## 40.4 Register Descriptions

Table 40.2 shows the register configuration.

**Table 40.2 Register Configuration**

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
LVDS register update control register	LVDS_UPDATE	R/W	H'00000000	H'FCFF7A30	32
LVDS format conversion register L	LVDSFCL	R/W	H'00000321	H'FCFF7A34	32
LVDS clock select register	LCLKSELR	R/W	H'00000000	H'FCFF7A50	32
LVDSPLL setting register	LPLLSETR	R/W	H'00000001	H'FCFF7A54	32
LVDS PHY characteristics switching register	LPHYACC	R/W	H'00000000	H'FCFF7A5C	32

### 40.4.1 LVDS Register Update Control Register (LVDS\_UPDATE)

LVDS\_UPDATE controls the timing for updating LVDS registers.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LVDS_UPDATE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	LVDS_UPDATE	0	R/W	LVDS Register Update 0: Registers are not updated. 1: Registers are updated.
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

### 40.4.2 LVDS Format Conversion Register L (LVDSFCL)

LVDSFCL is used to make necessary settings for conversion of the RGB signals output from video display controller 5 to the LVDS format.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	SYNC MODE	—	—	—	—	SYNC_POL[1:0]		—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R	R/W	R/W	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	LVDS_SEL2[3:0]			LVDS_SEL1[3:0]			LVDS_SEL0[3:0]			—	—	
Initial value:	0	0	0	0	0	0	1	1	0	0	1	0	0	0	0	1
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 29	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
28	SYNC_MODE	0	R/W	SYNC Mode Switch Switches the SYNC mode. 0: LVDS_TCON0 and LVDS_TCON2 are transmitted through LVDS signals. 1: Fixed values are transmitted as LVDS_TCON0 and LVDS_TCON2 through LVDS signals. The fixed values are specified in the SYNC_POL bits.
27 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23, 22	SYNC_POL [1:0]	00	R/W	LVDS_TCON0 and LVDS_TCON2 Value The values specified in these bits are output as the LVDS_TCON0 and LVDS_TCON2 values through LVDS signals when the SYNC_MODE bit is set to 1. SYNC_POL[1]: Specifies the LVDS_TCON0 output value. SYNC_POL[0]: Specifies the LVDS_TCON2 output value.
21 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 8	LVDS_SEL2 [3:0]	0011	R/W	TXOUT2P/M Data Format Setting These bits specify the format of the data to be output through the TXOUT2P and TXOUT2M pins. Be sure to set these bits to 3 in this LSI.
7 to 4	LVDS_SEL1 [3:0]	0010	R/W	TXOUT1P/M Data Format Setting These bits specify the format of the data to be output through the TXOUT1P and TXOUT1M pins. Be sure to set these bits to 2 in this LSI.
3 to 0	LVDS_SEL0 [3:0]	0001	R/W	TXOUT0P/M Data Format Setting These bits specify the format of the data to be output through the TXOUT0P and TXOUT0M pins. Be sure to set these bits to 1 in this LSI.

Note: This register is updated by setting the LVDS\_UPDATE bit in the LVDS register update control register (LVDS\_UPDATE) to 1.

### 40.4.3 LVDS Clock Select Register (LCLKSELR)

LCLKSELR specifies the input clock, the frequency dividing values for frequency dividers 1 and 2, and the channel used in video display controller 5.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	
	—	—	—	—	—	LVDS_IN_CLK_SEL [2:0]	—	—	—	—	—	—	—	—	—	LVDS_IDIV_SET [1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
R/W:	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W	
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	LVDSPLL_TST[5:0]					—	—	—	—	—	—	LVDS_CLK_EN	—	—	—	LVDS_VDC_SEL	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R	R	R	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 24	LVDS_IN_CLK_SEL [2:0]	All 0	R/W	These bits select the clock input to frequency divider 1. 0: VIDEO_X1 1: DV0_CLK 2: DV1_CLK 3: LCD0_EXTCLK 4: LCD1_EXTCLK 5: P1φ 6 and 7: Setting prohibited
23 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17, 16	LVDS_IDIV_SET[1:0]	All 0	R/W	These bits specify the frequency dividing value (NIDIV) for frequency divider 1. NIDIV = 1, 2, or 4 0: NIDIV = 1 1: NIDIV = 2 2: NIDIV = 4 3: Setting prohibited
15 to 10	LVDSPLL_TST[5:0]	All 0	R/W	Internal Parameter Setting for LVDS PLL These bits specify the internal parameters for the LVDS PLL (the resistance of the LPF circuit and the current through the charge pump circuit). Note: Be sure to set these bits to 010000 in this LSI.
9, 8	LVDS_ODIV_SET[1:0]	All 0	R/W	These bits specify the frequency dividing value (NODIV) for frequency divider 2. NODIV = 1, 2, or 4 0: NODIV = 1 1: NODIV = 2 2: NODIV = 4 3: Setting prohibited
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	LVDS_CLK_EN	0	R/W	Enables the LVDS PLL output. 0: Output from the PLL is disabled. 1: Output from the PLL is enabled. Note: Be sure to clear this bit to 0 before making settings in any other register described in this section. In addition, before modifying this bit, be sure to clear the PANEL_IKEN bits in SYSCNT_PANEL_CLK in video display controller 5 to 0.
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	LVDS_VDC_SEL	0	R/W	Selects a channel in video display controller 5 whose data is to be output through the LVDS interface. 0: Channel 0 is selected. 1: Channel 1 is selected.

Bit	Bit Name	Initial Value	R/W	Description
0	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

#### 40.4.4 LVDS PLL Setting Register (LPLLSETR)

LPLLSETR specifies the frequency dividing value and controls power-down in the LVDS PLL.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	LVDSPLL_FD[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	LVDSPLL_RD[4:0]				—	—	LVDSPLL_OD[1:0]		—	—	—	LVDSPLL_PD	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	LVDSPLL_FD[10:0]	All 0	R/W	These bits specify the frequency dividing value (NFD) for the feedback frequency in the LVDS PLL. NFD = 24 to 2047*2 24 to 2047*2: NFD = LVDSPLL_FD value 0 to 23: Setting prohibited
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 8	LVDSPLL_RD[4:0]	All 0	R/W	These bits specify the frequency dividing value (NRD) for the input frequency in the LVDS PLL. NRD = 1 to 32 0 to 31: NRD = LVDSPLL_RD value + 1
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5, 4	LVDSPLL_OD[1:0]	All 0	R/W	These bits specify the frequency dividing value (NOD) for the output frequency in the LVDS PLL. NOD = 1, 2, 4, or 8 0 to 3: NOD = 2 ^ LVDSPLL_OD value
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	LVDSPLL_PD	1	R/W	Controls power-down for the LVDS PLL. 0: Normal operation 1: Power-down state

- Notes:
- When selecting the bypass output, set LVDSPLL\_OD to 0.
  - The following values are also prohibited: 28 to 31, 37 to 39, 46, 47, and 55.

### 40.4.5 LVDS PHY Characteristics Switching Register (LPHYACC)

LPHYACC specifies the characteristics of the LVDS output buffers.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SKEWC [1]	SKEWC [0]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	SKEWC [1:0]	00	R/W	Skew Adjustment of TXCLKOUTP and TXCLKOUTM Pins These bits adjust the skew between the signals on the TXCLKOUTP and TXCLKOUTM pins. Note: For LVDS output, be sure to set these bits to 01.



## 40.5 Operation

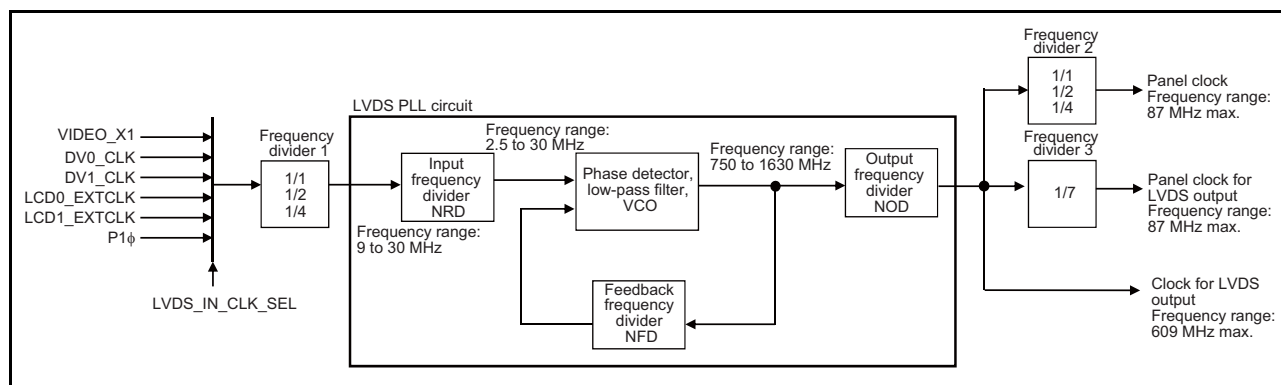
This module receives the RGB888 signals output from video display controller 5 and the SYNC signal, converts the video signals to the LVDS format, and outputs them. The precision of the RGB signals is reduced to six bits each. The operating clock is generated in the LVDS PLL.

### 40.5.1 LVDS PLL Settings

The input clock select circuit, frequency dividers 1 to 3, and LVDS PLL circuit are used to generate operating clocks. For input and output of each circuit, allowable frequency ranges are determined; be sure to make appropriate settings to generate input and output frequencies within the ranges.

Before modifying various setting registers, check that the panel clock operation in video display controller 5 is disabled (SYSCNT\_PANEL\_CLK.PANEL\_IKEN = 0) and the LVDS PLL output is disabled (LPLLSETR.LVDS\_CLK\_EN = 0).

Figure 40.3 shows a block diagram of the LVDS PLL and its peripheral circuits.



**Figure 40.3** Block Diagram of LVDS PLL and Peripheral Circuits

#### (1) Clock Input to LVDS PLL

The clock input to the LVDS PLL can be selected from among VIDEO\_X1, DV0\_CLK, DV1\_CLK, LCD0\_EXTCLK, LCD1\_EXTCLK, and P1φ clocks through the LVDS\_IN\_CLK\_SEL bit setting in LVDS\_CLKSELR.

The frequency of this input clock should be within the range from 9 to 30 MHz. When the input clock frequency is higher than the upper limit of this range, use frequency divider 1 placed before the LVDS PLL and reduce the frequency so that it is within the allowed range.

## (2) LVDS PLL Frequency Setting

The LVDS PLL has three internal frequency dividers: input frequency divider NRD, feedback frequency divider NFD, and output frequency divider NOD.

The frequency of the clock output from the LVDS PLL is calculated by the following equations. Note that the reference frequency, VCO output frequency, LVDS PLL input frequency, and LVDS PLL output frequency should be within the respective ranges shown below the equations.

$$\begin{aligned} \text{FREF} &= \text{FIN} / \text{NRD} && \text{(Reference frequency)} \\ \text{FVCO} &= \text{FIN} \times \text{NFD} / \text{NRD} && \text{(VCO output frequency )} \\ \text{FOUT} &= \text{FIN} \times \text{NFD} / (\text{NRD} \times \text{NOD}) && \text{(LVDS PLL output frequency )} \end{aligned}$$

Note:

FREF: Reference frequency. Frequency range = 2.5 MHz to 30 MHz

FVCO: VCO output frequency. Frequency range = 750 MHz to 1630 MHz

FIN: LVDS PLL input frequency. Frequency range = 9 MHz to 30 MHz

FOUT: LVDS PLL output frequency. Frequency range = 609 MHz max.

## (3) LVDS PLL Frequency Setting Timing

Before making settings for each frequency divider in the PLL circuit described in (2), place the PLL in the power-down state (PD = 1) and wait for at least 0.5  $\mu\text{s}$ . After making frequency divider settings, wait for a retention period of at least 0.5  $\mu\text{s}$  and release the power-down state (PD = 0).

Figure 40.4 shows this timing.

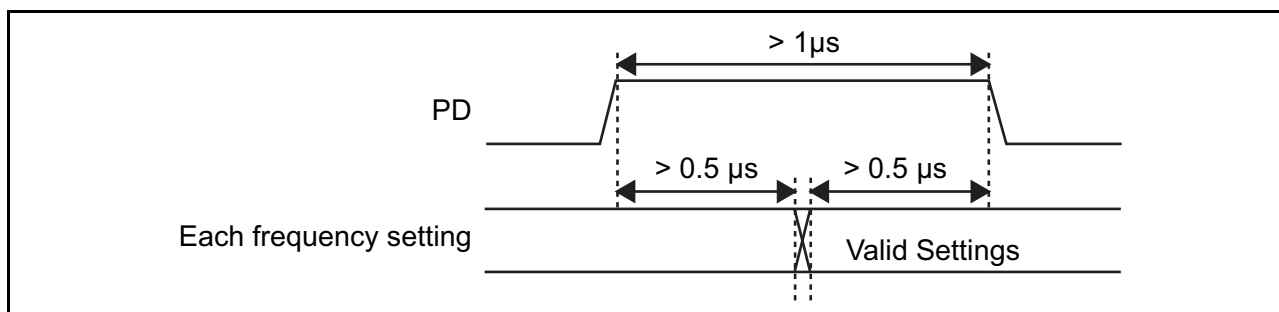


Figure 40.4 PLL Setting Timing

## (4) LVDS PLL Output Enable Timing

Output of the clock from the LVDS PLL can be enabled or disabled through the LVDS\_CLK\_EN bit setting in LCLKSELR.

To allow time for stabilization of oscillation by the PLL, wait for at least 200  $\mu\text{s}$  after release from the power-down state before enabling the LVDS PLL output. The LVDS\_CLK\_EN bit in LCLKSELR should be modified only while the panel clock operation in video display controller 5 is disabled (SYSCNT\_PANEL\_CLK.PANEL\_IKEN = 0).

(5) LVDS PLL Output Clock Frequency Setting

When using the clock output from the LVDS PLL as the clock for the LVDS output interface, select the clock from frequency divider 3 (divided by 7) as the panel clock for video display controller 5.

When using the clock output from the LVDS PLL as the clock for digital RGB output, select the clock from frequency divider 2 as the panel clock for video display controller 5. In addition, make appropriate settings so that the clock frequency after frequency division is 87 MHz or lower. This limitation on the frequency is not applied when the clock is used for the LVDS output interface.

The panel clock can be selected through the PANEL\_OCKSEL bits in SYSCNT\_PANEL\_CLK in video display controller 5.

40.5.2 LVDS Output Format

This module converts and then outputs the LCD data output from video display controller 5 and the timing signals generated by the LCD TCON.

(1) LVDS Output Format

(a) LCD data

After converted into the LVDS format, LCD data is output from LCD\_DATA23 to LCD\_DATA18, LCD\_DATA15 to LCD\_DATA10, and LCD\_DATA7 to LCD\_DATA2 pins.

(b) Timing signals generated by LCD TCON

After converted into the LVDS format, the timing signals are output from the LCD\_TCON0, LCD\_TCON2, and LCD\_TCON3 pins.

Figure 40.5 shows the bit alignment in the LVDS output format.

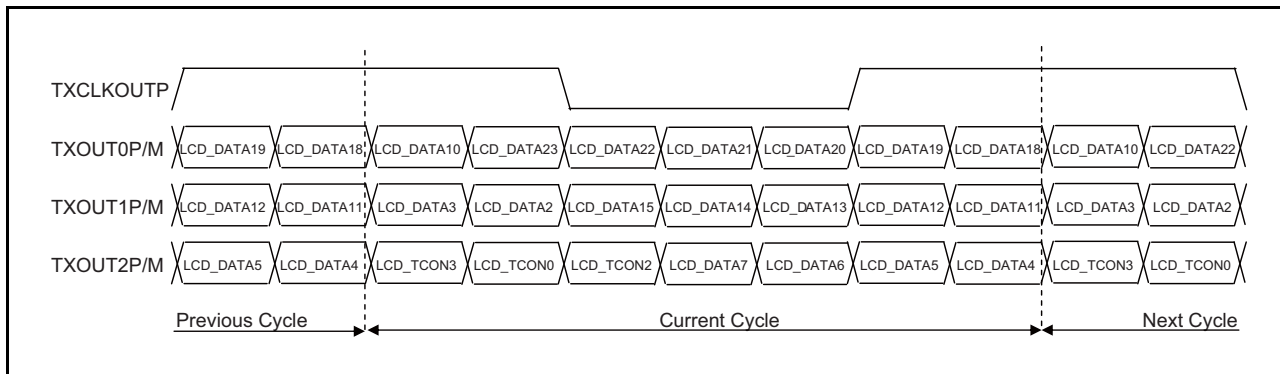


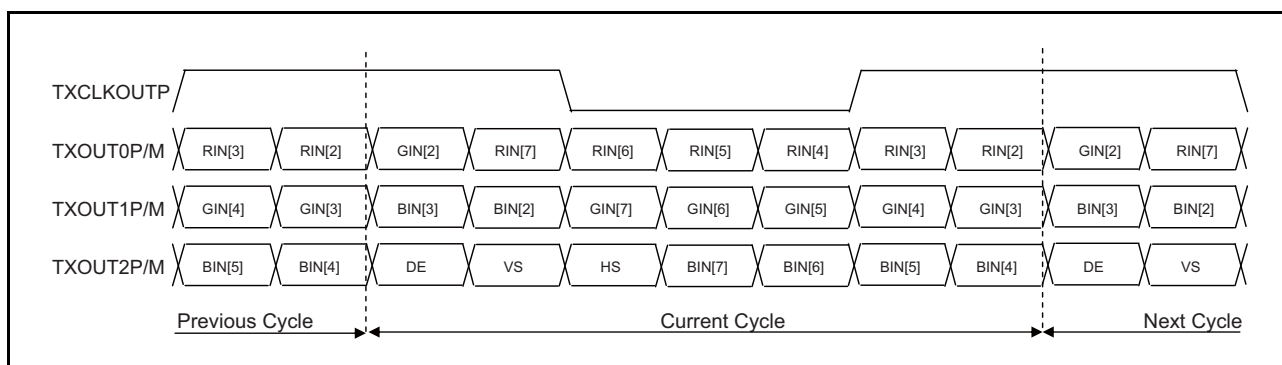
Figure 40.5 Data Map of LVDS Output Format

## (2) Setting Example

The following is the setting example of the LVDS output format. The register the setting is applied is register in video display controller 5. For the details of the register, see section 37, Video Display Controller 5 (7): Output Controller.

**Table 40.3 Setting Example of LVDS Output Format**

Register	Bit	Value	Remark
OUT_SET	OUT_ENDIAN_ON	0	Bit endian change is off.
	OUT_SWAP_ON	0	B/R signal swap is off.
	OUT_FORMAT[1:0]	0	RGB888 is selected as an output format (the upper six-bit data is output in the LVDS format).
TCON_TIM_STVA2	TCON_TIM_STVA2	0	The VS signal is selected for the LCD_TCON0 pin.
TCON_TIM_STVA1	TCON_STVA_VS[10:0]	Arbitrary	The VS signal timing is set.
	TCON_STVA_VW[10:0]	Arbitrary	The VS signal timing is set.
TCON_TIM_STH2	TCON_STH_SEL[2:0]	2	The HS signal is selected for the LCD_TCON2 pin.
TCON_TIM_STH1	TCON_STH_HS[10:0]	Arbitrary	The HS signal timing is set.
	TCON_STH_VS[10:0]	Arbitrary	The HS signal timing is set.
TCON_TIM_STB2	TCON_STB_SEL[2:0]	7	The DE signal is selected for the LCD_TCON7 pin.
TCON_TIM_STVB1	TCON_STVB_VS[10:0]	Arbitrary	The DE signal timing is set.
	TCON_STVB_VW[10:0]	Arbitrary	The DE signal timing is set.
TCON_TIM_STB1	TCON_STB_HS[10:0]	Arbitrary	The DE signal timing is set.
	TCON_STB_HW[10:0]	Arbitrary	The DE signal timing is set.



**Figure 40.6 Data Map of Setting Example of LVDS Output Format**

### 40.5.3 Procedures for Register Settings

#### (1) Initial Settings after Power-on Reset

The following shows an example of initial settings after a power-on reset.

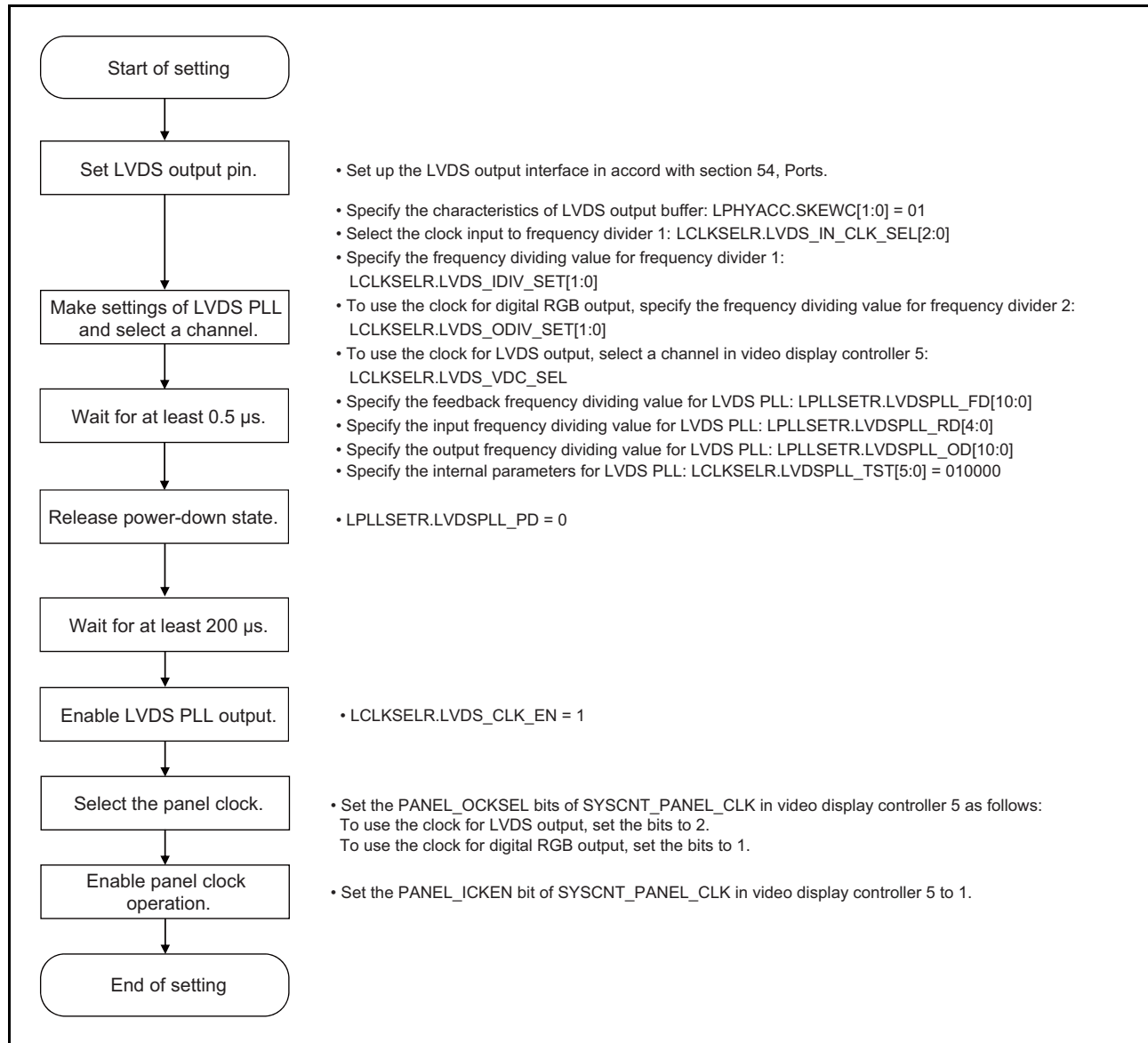


Figure 40.7 Example of Initial Settings after Power-On Reset

## (2) Frequency Modification Settings

The following shows an example of modifying frequencies.

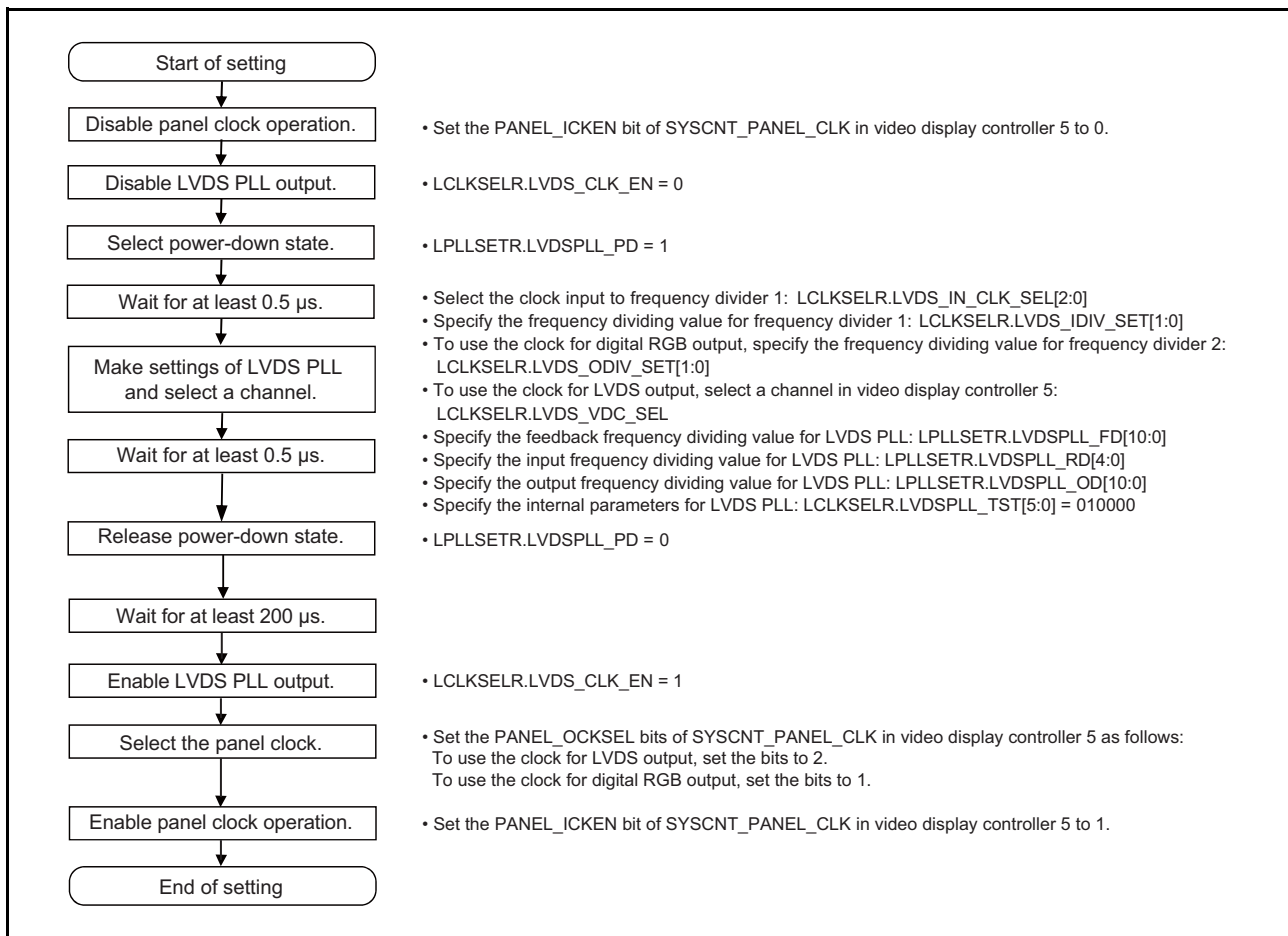


Figure 40.8 Example of Modifying Frequencies for LVDS Output

## 40.6 Notes

### 40.6.1 LVDS Output Pin Settings

The LVDS output buffers enter a low power consumption state when pins are set up other than for an LVDS output interface in accord with section 54, Ports. Although the LVDS output buffers return from the low power consumption state after pins are set to operate as an LVDS output interface, this requires up to 0.5  $\mu$ s. Accordingly, at least 0.5  $\mu$ s are required before actual LVDS output starts after pins have been set to operate as an LVDS output interface.

## 41. Image Renderer (IMR-LS2)

The contents of this section are available upon non-disclosure agreement.  
For details, contact your local sales representatives.

## 42. Image Renderer for Display (IMR-LSD)

The contents of this section are available upon non-disclosure agreement.  
For details, contact your local sales representatives.



## 43. Display Out Comparison Unit

The display out comparison unit (DISCOM) checks whether the data output from the graphics display module\* agrees with the expected graphics data. This checking is accomplished by comparing the CRC code of the data output from the graphics display module with the pre-calculated CRC code of the expected graphics data. Since this LSI incorporates two channels of graphics display modules, it also incorporates two channels of the DISCOM units corresponding to them.

Note: \*This indicates video display controller 5 in this LSI.

### 43.1 Features

This module has the following features.

- Comparison of Graphics Planes of Graphics Display Module  
One of the graphics planes of the graphics display module can be selected and its CRC code can be compared with the expected CRC code.
- Comparison of Data after  $\alpha$  Blending  
The CRC code of the graphics data obtained after  $\alpha$  blending in the graphics display module can be compared with the expected CRC code.
- Rectangular Area Specification  
The rectangular area can be specified based on the graphics data output from the graphics display module (graphics plane or graphics data obtained after  $\alpha$  blending) and its CRC code can be compared with the expected CRC code.
- Pixel Format  
A pixel format for 32 bits/pixel or 16 bits/pixel can be selected.  
ARGB8888/RGB888 are available for 32 bits/pixel, and only RGB565 is available for 16 bits/pixel.

### 43.2 Block Diagram

An overall block diagram of this module is shown in Figure 43.1.

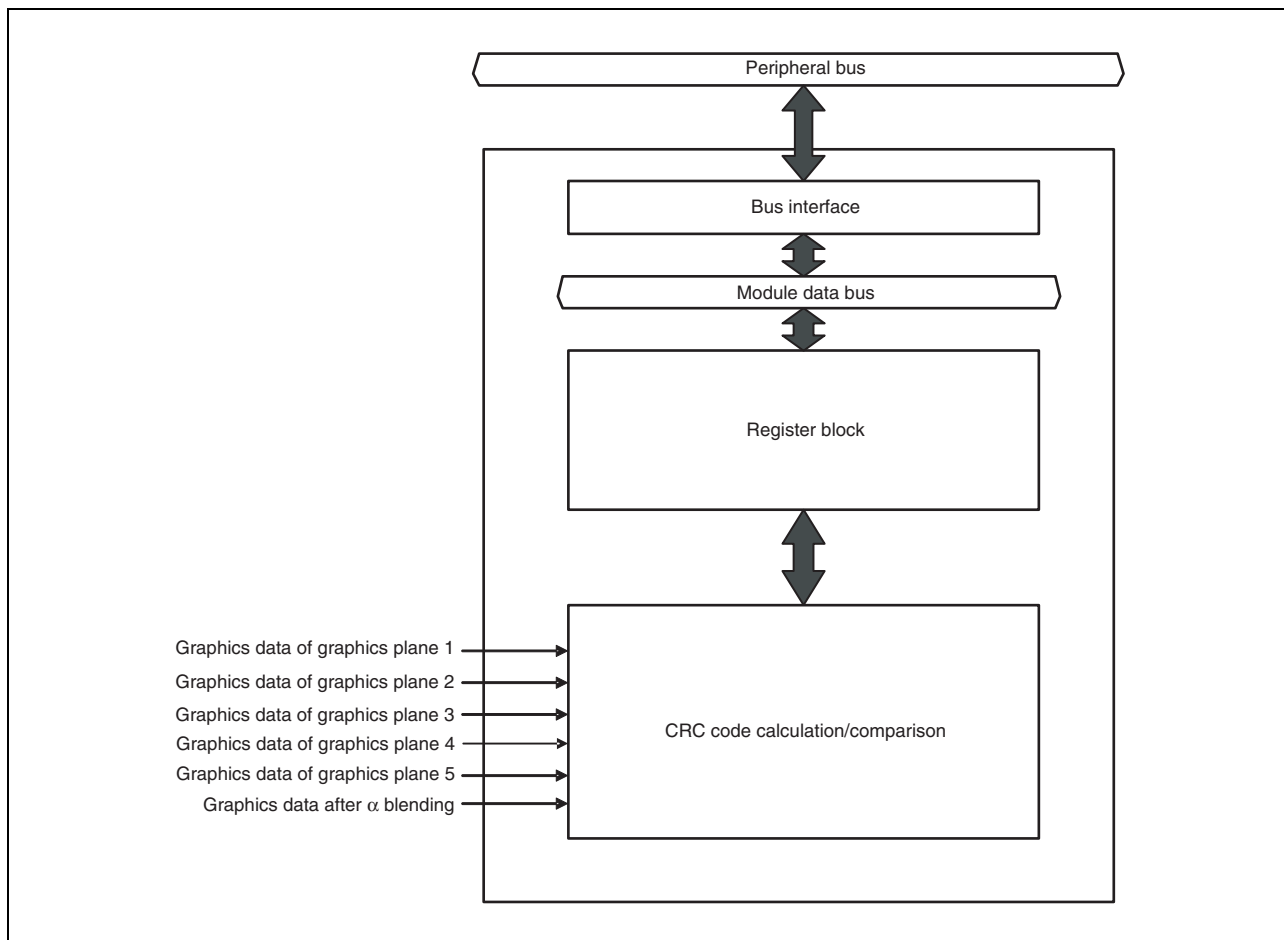


Figure 43.1 Block Diagram

### 43.3 Register Descriptions

Table 43.1 and Table 43.2 show the register configuration.

**Table 43.1 Register Configuration (channel 0)**

Register Name	Abbreviation	R/W	Address	Access Size
Control register	DOCMCR	R/W	H'FCFFA800	32
Status register	DOCMSTR	R	H'FCFFA804	32
Status clear register	DOCMCLSTR	R/W	H'FCFFA808	32
Interrupt enable register	DOCMIENR	R/W	H'FCFFA80C	32
Operation parameter setting register	DOCMPMR	R/W	H'FCFFA814	32
Expected CRC code register	DOCMECR	R/W	H'FCFFA818	32
Calculated CRC code value register	DOCMCCR	R	H'FCFFA81C	32
Horizontal start position setting register	DOCMSPXR	R/W	H'FCFFA820	32
Vertical start position setting register	DOCMSPYR	R/W	H'FCFFA824	32
Horizontal size setting register	DOCMSZXR	R/W	H'FCFFA828	32
Vertical size setting register	DOCMSZYR	R/W	H'FCFFA82C	32
CRC code initialization register	DOCMCRCIR	R/W	H'FCFFA830	32

**Table 43.2 Register Configuration (channel 1)**

Register Name	Abbreviation	R/W	Address	Access Size
Control register	DOCMCR	R/W	H'FCFFB000	32
Status register	DOCMSTR	R	H'FCFFB004	32
Status clear register	DOCMCLSTR	R/W	H'FCFFB008	32
Interrupt enable register	DOCMIENR	R/W	H'FCFFB00C	32
Operation parameter setting register	DOCMPMR	R/W	H'FCFFB014	32
Expected CRC code register	DOCMECR	R/W	H'FCFFB018	32
Calculated CRC code value register	DOCMCCR	R	H'FCFFB01C	32
Horizontal start position setting register	DOCMSPXR	R/W	H'FCFFB020	32
Vertical start position setting register	DOCMSPYR	R/W	H'FCFFB024	32
Horizontal size setting register	DOCMSZXR	R/W	H'FCFFB028	32
Vertical size setting register	DOCMSZYR	R/W	H'FCFFB02C	32
CRC code initialization register	DOCMCRCIR	R/W	H'FCFFB030	32

### 43.3.1 Control Register (DOCMCR)

DOCMCR turns CRC code comparison on or off.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CMPRU
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CMPR
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	CMPRU	0	R	Display Out Comparison Update Value Reflects the internal update of the CMPR bit. It should be checked that this bit is 0 before updating the registers other than the CMPR bit in DOCMCR, DOCMCLSTR, and DOCMIENR. For details, see section 43.4.8, Register Update Timing.
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	CMPR	0	R/W	Display Out Comparison Execution Executes display out comparison. This bit is loaded inside when the start of the valid period of the graphics data is detected. For details, see section 43.4.8, Register Update Timing. 0: Stops display out comparison. 1: Executes display out comparison.

### 43.3.2 Status Register (DOCMSTR)

DOCMSTR returns the comparison result of the CRC code. The result is reflected in this register when the end of the valid period of the graphics data is detected.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CMPST
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved Values read from these bits are undefined. The write value should always be 0.
29 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	CMPST	0	R	Display Out Comparison Status 0: Compared CRC codes match. 1: Compared CRC codes do not match.

### 43.3.3 Status Clear Register (DOCMCLSTR)

Writing 1 to the CMPCLST bit causes clearing of the CMPST bit in DOCMSTR to 0. However, clearing of the CMPST bit in DOCMSTR after 1 is written to the CMPCLST bit takes a fixed amount of time. Confirm that the CMPST bit in DOCMSTR is actually cleared after writing 1 to the CMPCLST bit.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CMP CLST
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	CMPCLST	0	R/W	Display Out Comparison Status Clear Setting this bit to 1 clears the CMPST bit in DOCMSTR to 0. This bit is always read as 0.

### 43.3.4 Interrupt Enable Register (DOCMIENR)

DOCMIENR enables interrupt of the corresponding status bits in DOCMSTR.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CMPIEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	CMPIEN	0	R/W	Display Out Comparison Mismatch Detection Interrupt Enable Enables/disables the display out comparison mismatch detection interrupt (CMPI) when the CMPST bit in DOCMSTR is set to 1. 0: Disables the display out comparison mismatch detection interrupt (CMPI). 1: Enables the display out comparison mismatch detection interrupt (CMPI).

### 43.3.5 Operation Parameter Setting Register (DOCMPMR)

DOCMPMR selects the graphics data and sets the pixel format.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CMPBT
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMPDFA[7:0]							CMPDAUF	-	-	-	CMPSELP[3:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	CMPBT	0	R/W	Pixel Format Data Width Specifies the data width of the pixel format. 0: 32 bits/pixel (ARGB8888/RGB888 format) 1: 16 bits/pixel (RGB565 format)
15 to 8	CMPDFA[7:0]	H'00	R/W	Display Out Comparison Default $\alpha$ Value Set the default $\alpha$ value. These bits are enabled when the CMPDAUF bit is 1.
7	CMPDAUF	0	R/W	Display Out Comparison Default $\alpha$ Value Use Enables the use of default $\alpha$ value. 0: Disables use of default $\alpha$ value. 1: Enables use of default $\alpha$ value. Note: This bit is enabled only if RGB888 format is selected. For ARGB8888/RGB565 format, this bit should always be set to 0. When this bit is set to 0 when RGB888/ARGB8888/RGB565 pixel format is selected, the data output from the graphics display module is used as the $\alpha$ value.
6 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	CMPSELP[3:0]	0000	R/W	Display Out Comparison Selection Plane Select the graphics data for CRC code comparison. 0: Selects no data. 1 to 5: Select graphics data of graphics plane 1 to graphics plane 5. 9: Selects graphics data after $\alpha$ blending. Other than above: Setting prohibited

### 43.3.6 Expected CRC Code Register (DOCMECR CR)

DOCMECR CR specifies the CRC code of the expected graphics data.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CMPECRC[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMPECRC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CMPECRC [31:0]	H'00000000	R/W	Expected Display Out Comparison CRC Code The expected CRC code value for the rectangular area of the selected graphics data

### 43.3.7 Calculated CRC Code Value Register (DOCMCCRCR)

The CRC code calculation result of the selected graphics plane or rectangular area can be read from this register. The calculation result is reflected in this register when the end of the valid period of the graphics data is detected.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CMPCCRC[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMPCCRC[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CMPCCRC [31:0]	H'00000000	R	Calculated Display Out Comparison CRC Code Value The calculated CRC code value for the rectangular area of the selected graphics data

### 43.3.8 Horizontal Start Position Setting Register (DOCMSPXR)

DOCMSPXR specifies the horizontal start position of the rectangular area for which the CRC code is calculated.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	CMPSPX[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	CMPSPX[10:0]	H'000	R/W	Display Out Comparison Horizontal Start Position Specify the horizontal start position of the rectangular area for which the CRC code is calculated. The set value should be smaller than or equal to the horizontal size of the graphics data.

### 43.3.9 Vertical Start Position Setting Register (DOCMSPYR)

DOCMSPYR specifies the vertical start position of the rectangular area for which the CRC code is calculated.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	CMPSPY[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	CMPSPY[10:0]	H'000	R/W	Display Out Comparison Vertical Start Position Specify the vertical start position of the rectangular area for which the CRC code is calculated. The set value should be smaller than or equal to the vertical size of the graphics data.

### 43.3.10 Horizontal Size Setting Register (DOCMSZXR)

DOCMSZXR specifies the horizontal size of the rectangular area for which the CRC code is calculated.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	CMPSZX[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	CMPSZX [10:0]	H'000	R/W	Display Out Comparison Horizontal Size Specify the horizontal size of the rectangular area for which the CRC code is calculated. The value should be set as follows: Horizontal size of the graphics data ≥ Horizontal start position (CMPSPX) + Horizontal size (CMPSZX).



### 43.3.11 Vertical Size Setting Register (DOCMSZYR)

DOCMSZYR specifies the vertical size of the rectangular area for which the CRC code is calculated.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	CMPSZY[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	CMPSZY [10:0]	H'000	R/W	Display Out Comparison Vertical Size Specify the vertical size of the rectangular area for which the CRC code is calculated. The value should be set as follows: Vertical size of the graphics data ≥ Vertical start position (CMPSPY) + Vertical size (CMPSZY).

### 43.3.12 CRC Code Initialization Register (DOCMCRCIR)

DOCMCRCIR is used to specify the initial value of the CRC code.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRCINI[31:16]															
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CRCINI[15:0]															
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CRCINI[31:0]	H'FFFFFFF	R/W	Display Out CRC Comparison Initial Value These bits specify the initial value of the CRC for the rectangular area of the selected graphics data.

## 43.4 Operation

### 43.4.1 Overview of Operations

This module is capable of calculating the CRC code of the arbitrary rectangular area of graphics data. By comparing the CRC code with the pre-calculated expected CRC code value, this module can detect whether the display output is obtained as expected.

Main features of this module are as follows.

- Graphics data can be selected from graphics plane 1 to graphics plane 5
- Graphics data after  $\alpha$  blending can be selected
- Arbitrary rectangular area of the selected graphics data can be specified
- Pixel format can be selected from ARGB8888 and RGB888 for 32 bits/pixel or RGB565 for 16 bits/pixel
- Interrupt is generated when the compared CRC codes do not match.

### 43.4.2 System Configuration

This module is configured as shown in Figure 43.2.

The CRC code is calculated after receiving the graphics data output from the graphics display module. The calculated CRC code is then compared with the pre-calculated expected CRC code value. The graphics data can be selected from graphics plane 1 to graphics plane 5, and the data after  $\alpha$  blending. Note that the graphics planes 1 to 4 correspond respectively to graphics (0) to (3) in the image synthesizer of video display controller 5, and the graphics plane 5 corresponds to graphics (OIR) of the output video image generation block.

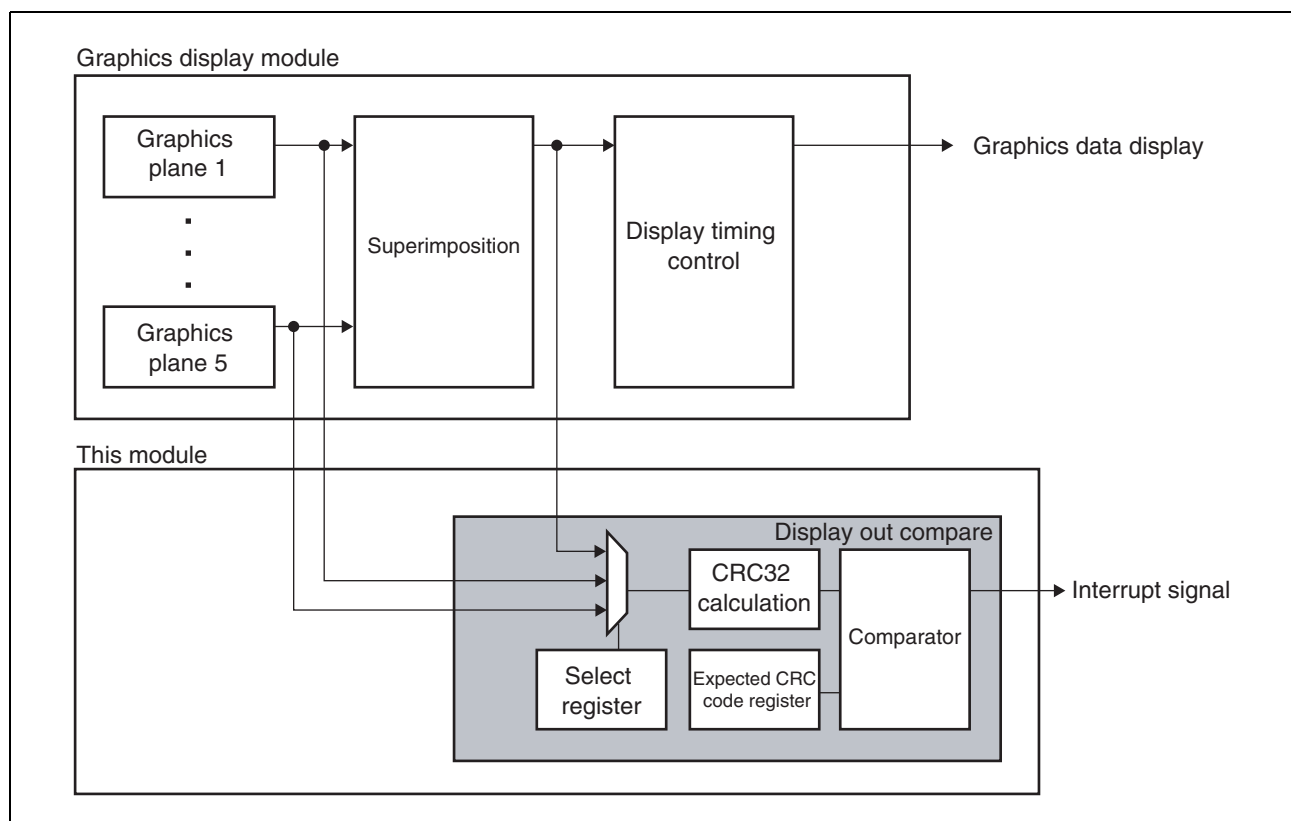


Figure 43.2 System Configuration

### 43.4.3 CRC Calculation Method

The display out comparison unit generates a 32-bit CRC code by using the following CRC polynomial (IEEE802.3).

$$x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$$

CRC is sequentially calculated beginning with LSB in pixel units.

In other words, for 32 bits/pixel, it is calculated in units of 32 bits and for 16 bits/pixel, it is calculated in units of 16 bits. During CRC calculation of data, pixel data is input when the graphics data is output (top left to bottom right).

### 43.4.4 Graphics Data Selection for CRC Code Generation

The graphics data for which the CRC code is calculated can be selected from the graphics plane 1 to graphics plane 5 or the graphics data after  $\alpha$  blending by setting the CMPSELP[3:0] bits in DOCMPMR.

### 43.4.5 Pixel Format

#### (1) Pixel Format Specification

DOCMPMR specifies the pixel format. The pixel formats are given in Table 43.3.

**Table 43.3 Pixel Format**

Bits in DOCMPMR		CMPBT	CMPDFA[7:0]	CMPDAUF
32 bits/pixel	ARGB8888	0	—	—
	RGB888		Arbitrary	Arbitrary
16 bits/pixel	RGB565	1	—	—

#### (2) Data Arrangement for Available Pixel Formats

Data arrangement for each pixel format is given below.

- ARGB8888 (32 bits/pixel)

b31	b24 b23	b16 b15	b8 b7	b0
$\alpha$ 8 bits	Red 8 bits	Green 8 bits	Blue 8 bits	

- RGB888 (32 bits/pixel)

b31	b24 b23	b16 b15	b8 b7	b0
*	Red 8 bits	Green 8 bits	Blue 8 bits	

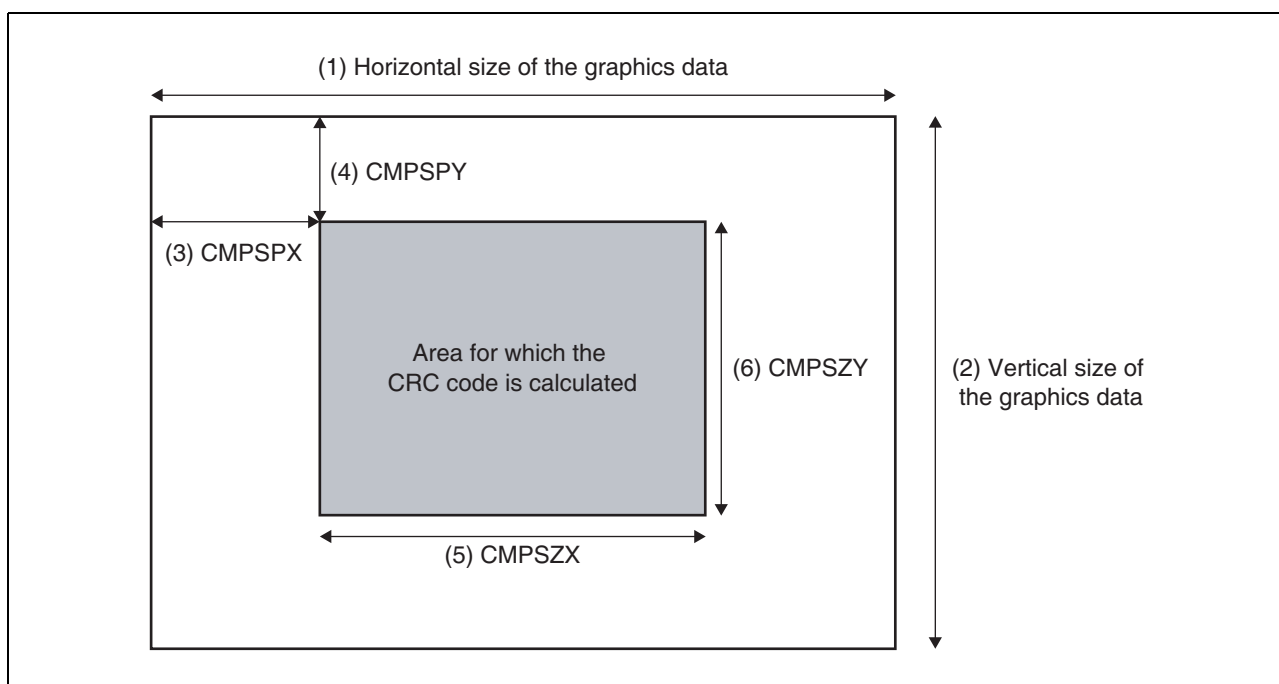
Note: \* When CMPDAUF = 0,  $\alpha$  value is the value output from the graphics display module.  
When CMPDAUF = 1,  $\alpha$  value is the value specified by CMPDFA[7:0].

- RGB565 (16 bits/pixel)

b15	b11 b10	b5 b4	b0
Red 5 bits	Green 6 bits	Blue 5 bits	

### 43.4.6 Rectangular Area Settings

Based on the selected graphics data, the start position and size of the rectangular area for which the CRC code is calculated can be set with the registers. Figure 43.3 shows such a rectangular area and Table 43.4 shows the register settings for the area.



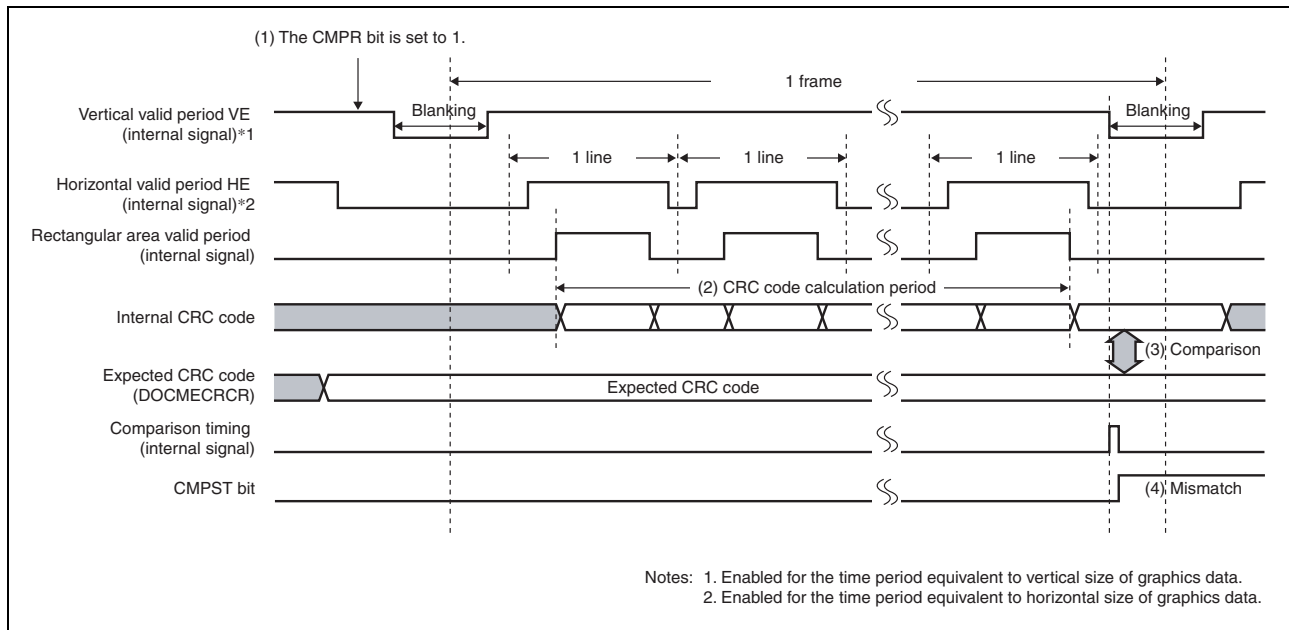
**Figure 43.3 Rectangular Area for which the CRC Code is Calculated**

**Table 43.4 Register Settings for the Rectangular Area for which the CRC Code is Calculated**

No.	Symbol in the Figure	Register Used for Setting	Description
(1)	Horizontal size of the graphics data		Horizontal size of the graphics data. Set the size using the graphics display module.
(2)	Vertical size of the graphics data		Vertical size of the graphics data. Set the size using the graphics display module.
(3)	CMPSPX (horizontal start position)	DOCMSPXR	Set the horizontal distance from the upper left origin of the graphics data to the rectangular area for which the CRC code is calculated in pixel units.
(4)	CMPSPY (vertical start position)	DOCMSPYR	Set the vertical distance from the upper left origin of the graphics data to the rectangular area for which the CRC code is calculated in line units.
(5)	CMPSZX (horizontal size)	DOCMSZXR	Set the horizontal size of the rectangular area for which the CRC code is calculated in pixel units. The value should be set as follows: Horizontal size of the graphics data $\geq$ CMPSPX + CMPSZX.
(6)	CMPSZY (vertical size)	DOCMSZYR	Set the vertical size of the rectangular area for which the CRC code is calculated in line units. The value should be set as follows: Vertical size of the graphics data $\geq$ CMPSPY + CMPSZY.

### 43.4.7 CRC Calculation Time Period and Comparison Timing

Figure 43.4 shows the CRC calculation time period and timing of comparing the calculated result with the expected value.



**Figure 43.4 CRC Calculation Time Period and Timing of Comparing the Calculated Result with the Expected Value**

#### [Operation]

- (1) The operation starts at the next frame after the CMPR bit in DOCMCR is set to 1. For the register update timing, see section 43.4.8, Register Update Timing.
- (2) CRC code is calculated in the set rectangular area.
- (3) The CRC code calculation result is compared with the expected CRC code value (DOCMECRCR) at the end of the valid period of the graphics data.
- (4) If the compared CRC codes do not match, the CMPST bit in DOCMSTR is set.

### 43.4.8 Register Update Timing

#### (1) Timing when Register Values are Loaded Inside

All the register bits except the CMPR bit in DOCMCR are loaded inside immediately after the registers are written to. Thus, if the register is updated with the CMPRU bit in DOCMCR as 1, an unexpected result is likely to be obtained in the CRC code calculation. The registers that affect the CRC code calculation (i.e., registers other than DOCMCLSTR and DOCMIENR) should be updated after confirming that the CMPRU bit in DOCMCR is 0.

The CMPR bit in DOCMCR is loaded inside upon detection of the start of the valid period of the graphics data. Thus, even if the register is rewritten to in the middle of a frame, the CRC code calculation of the frame does not get affected.

#### (2) Timing when Internal State is Reflected in Registers

The internal state is reflected in DOCMSTR and DOCMCCRCR at the end of the valid period of the graphics data. The internal state of the CMPR bit is reflected in the CMPRU bit in DOCMCR at the start of the valid period of the graphics data.

Figure 43.5 shows the register update timing.

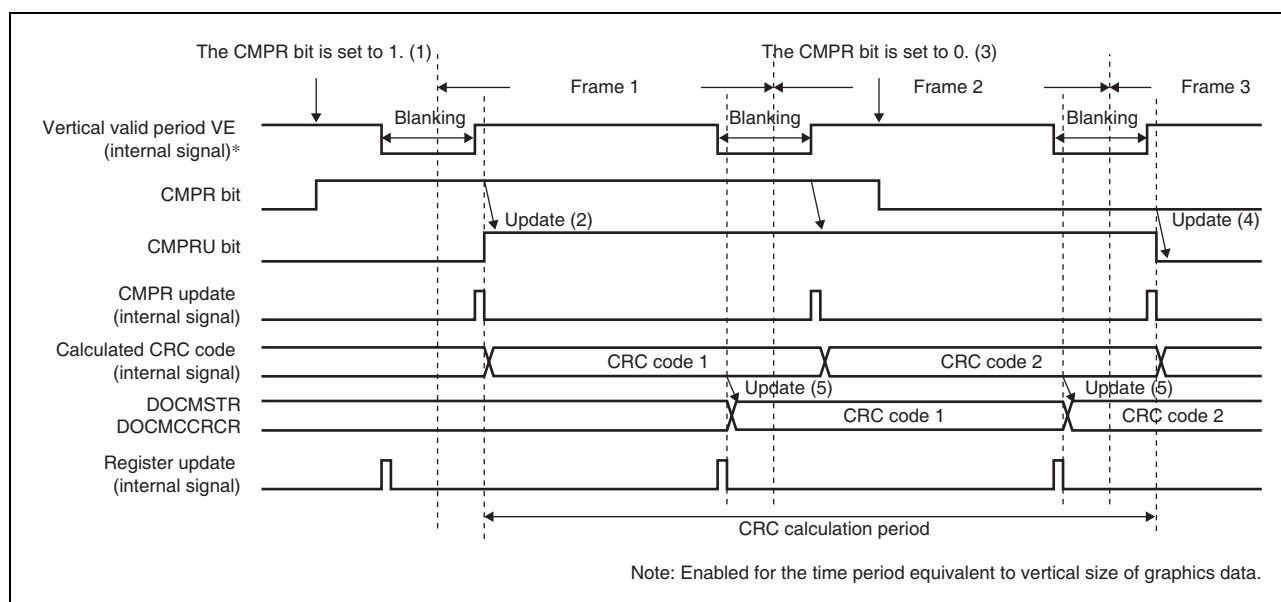


Figure 43.5 Register Update Timing

#### [Operation]

- (1) The CMPR bit is set to 1. This bit is not immediately loaded inside.
- (2) CRC calculation is carried out after the CMPR bit value is loaded inside upon detection of the start of the valid period of the graphics data.
- (3) To suspend the CRC comparison, the CMPR bit is set to 0. Similarly to (1), this bit is not immediately loaded inside.
- (4) The CRC calculation is suspended after the CMPR bit value is loaded inside upon detection of the start of the valid period of the graphics data.
- (5) The internal state is reflected in the register at the end of the valid period of the graphics data.

### 43.4.9 Operation Flow

#### (1) Procedure for Starting Display Out Comparison

Figure 43.6 shows a sample procedure for starting the display out comparison.

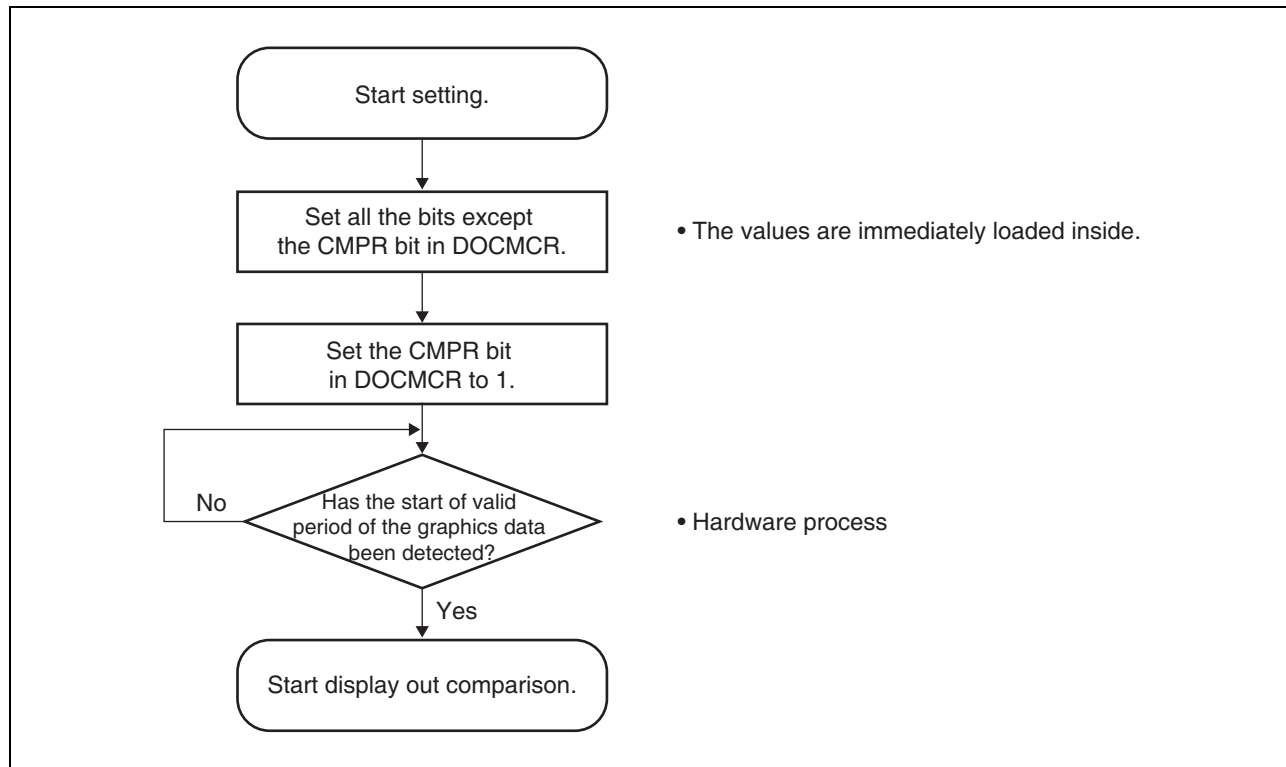


Figure 43.6 Sample Procedure for Starting the Display Out Comparison

(2) Procedure for Changing Register Setting

Figure 43.7 shows a sample procedure for changing the register setting during display out comparison.

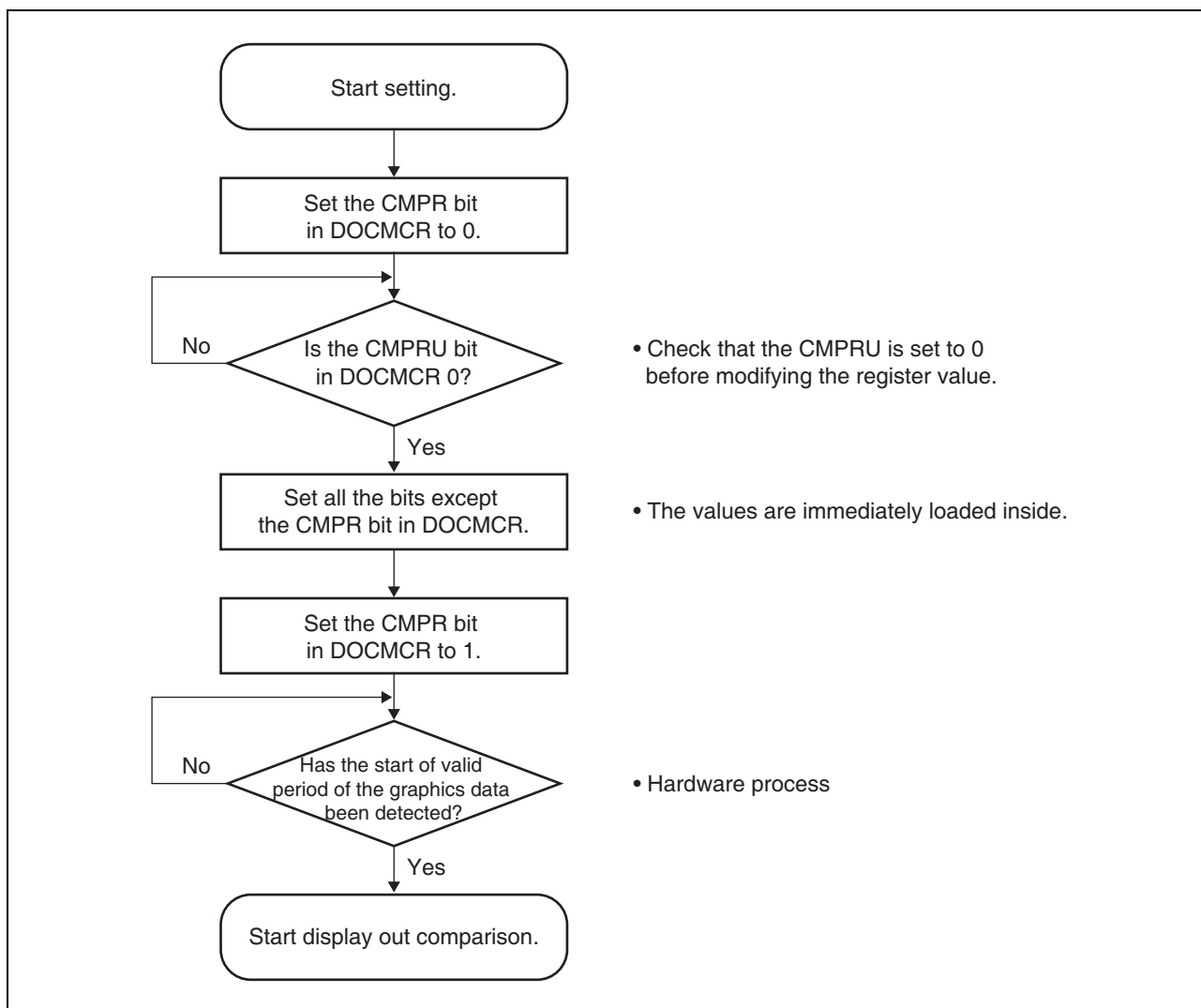


Figure 43.7 Sample Procedure for Changing the Register Setting during Display Out Comparison



## 43.5 Interrupt

The display out comparison mismatch detection interrupt is provided as an interrupt source.

The interrupt request is generated when the CMPIEN bit in DOCMIENR and the CMPST bit in DOCMSTR are both set to 1.

## 43.6 Usage Note

### 43.6.1 Expected CRC Value

When graphics plane 1 to graphics plane 5 are selected, the expected CRC code value (DOCMECR) should be calculated from the graphics data to be used by using the software. When the graphics data after  $\alpha$  blending is selected, the graphics data after  $\alpha$  blending should be selected at the time of debugging and the calculated CRC code value read from DOCMCCR should be used as the expected CRC code value. This is because there is a possibility of mismatching of the result of superimposition of the graphics data by the software and the result of superimposition by the graphics display module due to an error in calculation when the graphics data after  $\alpha$  blending is selected.

### 43.6.2 Expansion Control Functionality

When scaling settings are applied for expansion processing by video display controller 5, this module becomes incapable of generating CRC codes. Stop this module while expansion processing is being applied.

## 44. Renesas Graphics Processor for OpenVG™

### 44.1 Specification

This processor supports OpenVG™1.1, an open source 2D vector graphics API. Dedicated hardware and a programmable shader are used to accelerate OpenVG™ Stage 2 to Stage 8 processing.

Please refer to the Khronos Group Web site for the specifications of OpenVG™1.1.

## 45. JPEG Codec Unit

The JPEG codec unit (JCU) incorporates a JPEG codec conforming to the JPEG baseline compression and decompression standard to provide high-speed compression of image data and high-speed decoding of JPEG data.

### 45.1 Features

The JPEG codec unit has the following features:

- Conforms to the JPEG baseline standard within the range described in this document.  
This module does not support the following basic features:  
Scanning with two elements  
Non-interleave scanning with multiple elements
- Operational precision: Conforming to JPEG Part 2, ISO-IEC10918-2
- Image input/output system: Block interleave method
- Pixel format:  
Compression: YCbCr422 (H = 2:1:1, V = 1:1:1)  
Decompression: YCbCr444 (H = 1:1:1, V = 1:1:1), YCbCr422 (H = 2:1:1, V = 1:1:1), YCbCr411 (H = 4:1:1, V = 1:1:1), YCbCr420 (H = 2:1:1, V = 2:1:1)  
Output pixel format to the buffer: YCbCr422, ARGB8888, RGB565
- Four quantization tables provided
- Four Huffman tables provided (two tables for AC coefficients and two tables for DC coefficients)
- Markers supported: SOI (start of image), SOF0 (start of frame type 0), SOS (start of scan), DQT (define quantization tables), DHT (define Huffman tables), DRI (define restart interval), RSTm (restart marks), and EOI (end of image)
- Image data rate: Max. 133.34 Mbytes/s (at 66.67-MHz operation)
- The buffer size can be reduced by using the mode in which data transfer is temporarily stopped each time the specified number of lines or the specified amount of data is transferred during image data or coded data input/output.
- Processing unit: 8-byte address boundary units can be set
- Image sizes that can be processed: Sizes divisible by the minimum coded unit (MCU): 8 lines by 8 pixels in YCbCr444; 8 lines by 16 pixels in YCbCr422; 8 lines by 32 pixels in YCbCr411; 16 lines by 16 pixels in YCbCr420

Note: • Compression and decompression processing of images in unsupported pixel formats or unsupported image sizes should be avoided.

Figure 45.1 shows a block diagram.

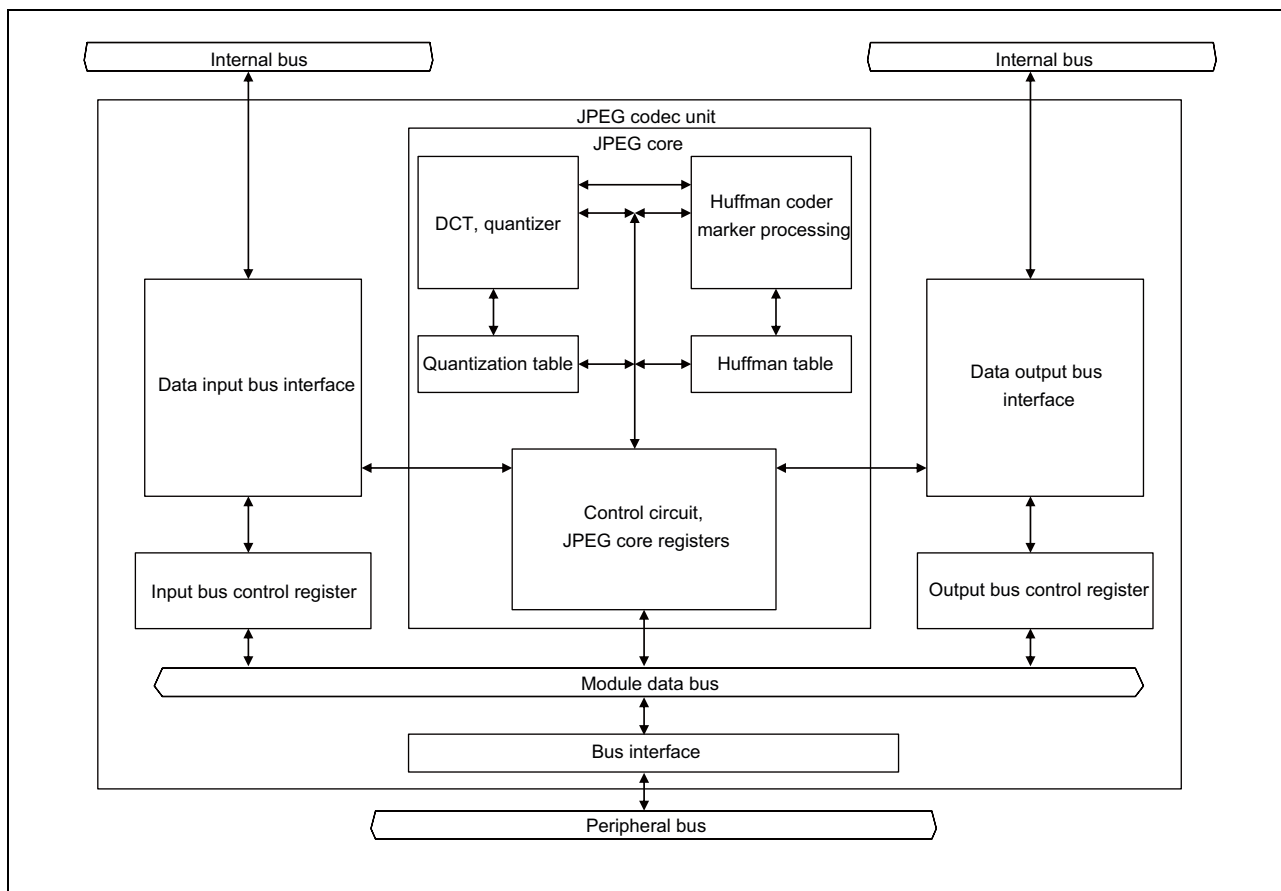


Figure 45.1 Block Diagram

## 45.2 Register Descriptions

Table 45.1 shows the JCU registers.

**Table 45.1 Register Configuration**

Register Name	Abbreviation	R/W	Address	Access Size
JPEG code mode register	JCMOD	R/W	H'E801 7000	8
JPEG code command register	JCCMD	R/W	H'E801 7001	8
JPEG code quantization table number register	JCQTN	R/W	H'E801 7003	8
JPEG code Huffman table number register	JCHTN	R/W	H'E801 7004	8
JPEG code DRI upper register	JCDRIU	R/W	H'E801 7005	8
JPEG code DRI lower register	JCDRID	R/W	H'E801 7006	8
JPEG code vertical size upper register	JCVSZU	R/W	H'E801 7007	8
JPEG code vertical size lower register	JCVSZD	R/W	H'E801 7008	8
JPEG code horizontal size upper register	JCHSZU	R/W	H'E801 7009	8
JPEG code horizontal size lower register	JCHSZD	R/W	H'E801 700A	8
JPEG code data count upper register	JCDTCU	R	H'E801 700B	8
JPEG code data count middle register	JCDTCM	R	H'E801 700C	8
JPEG code data count lower register	JCDTCD	R	H'E801 700D	8
JPEG interrupt enable register 0	JINTE0	R/W	H'E801 700E	8
JPEG interrupt status register 0	JINTS0	R/W	H'E801 700F	8
JPEG code decode error register	JCDERR	R/W	H'E801 7010	8
JPEG code reset register	JCRST	R	H'E801 7011	8
JPEG interface compression control register	JIFECNT	R/W	H'E801 7040	32
JPEG interface compression source address register	JIFESA	R/W	H'E801 7044	32
JPEG interface compression line offset register	JIFESOFST	R/W	H'E801 7048	32
JPEG interface compression destination address register	JIFEDA	R/W	H'E801 704C	32
JPEG interface compression source line count register	JIFESLC	R/W	H'E801 7050	32
JPEG interface compression destination register	JIFEDDC	R/W	H'E801 7054	32
JPEG interface decompression control register	JIFDCNT	R/W	H'E801 7058	32
JPEG interface decompression source address register	JIFDSA	R/W	H'E801 705C	32
JPEG interface decompression destination offset register	JIFDDOFST	R/W	H'E801 7060	32
JPEG interface decompression destination address register	JIFDDA	R/W	H'E801 7064	32
JPEG interface decompression source count register	JIFDSDC	R/W	H'E801 7068	32
JPEG interface decompression destination line count register	JIFDDLCL	R/W	H'E801 706C	32
JPEG interface decompression $\alpha$ setting register	JIFDADT	R/W	H'E801 7070	32
JPEG interrupt enable register 1	JINTE1	R/W	H'E801 708C	32
JPEG interrupt status register 1	JINTS1	R/W	H'E801 7090	32
JPEG input image data CbCr range setting register	JIFESVSZ	R/W	H'E801 7094	32
JPEG output image data CbCr range setting register	JIFESHSZ	R/W	H'E801 7098	32
JPEG code quantization table 0 register	JCQTBL0	R/W	H'E801 7100 to H'E801 713F	8
JPEG code quantization table 1 register	JCQTBL1	R/W	H'E801 7140 to H'E801 717F	8
JPEG code quantization table 2 register	JCQTBL2	R/W	H'E801 7180 to H'E801 71BF	8
JPEG code quantization table 3 register	JCQTBL3	R/W	H'E801 71C0 to H'E801 71FF	8

**Table 45.1 Register Configuration**

Register Name	Abbreviation	R/W	Address	Access Size
JPEG code Huffman table DC0 register	JCHTBD0	W	H'E801 7200 to H'E801 721B	8
JPEG code Huffman table AC0 register	JCHTBA0	W	H'E801 7220 to H'E801 72D1	8
JPEG code Huffman table DC1 register	JCHTBD1	W	H'E801 7300 to H'E801 731B	8
JPEG code Huffman table AC1 register	JCHTBA1	W	H'E801 7320 to H'E801 73D1	8

Note: • For the settings of the JPEG code quantization table and JPEG code Huffman table, see section 45.3.1 (4), Table Setting.

### 45.2.1 JPEG Code Mode Register (JCMOD)

JCMOD sets the operating mode before the JCU starts operation.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	DSP	REDU[2:0]		
Initial value:	0	0	0	0	0	0	0	0
R/W(compress):	R	R	R	R	R/W	R/W	R/W	R/W
R/W(decompress):	R	R	R	R	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
7 to 4	—	All 0		R	Reserved These bits are always read as 0. The write value should always be 0.
3	DSP	0		R/W	Compression/Decompression Set 0: Compression process 1: Decompression process Note: • When changing between processing for compression and for decompression, be sure to reset this module in advance by setting the SRST21 bit in the software reset control register 2 (SWRSTCR2) of the power-down modes.
2 to 0	REDU[2:0]	000	R/W	R	Pixel Format [Compression] 001: YCbCr422 Other than above: Setting prohibited. [Decompression] 000: YCbCr444 001: YCbCr422 110: YCbCr411 010: YCbCr420 Other than above: Error (JCU cannot process normally.)

### 45.2.2 JPEG Code Command Register (JCCMD)

JCCMD sets commands. Bits of this register need not be cleared to 0 after setting a command. Multiple commands must not be set simultaneously.

Bit:	7	6	5	4	3	2	1	0
	BRST	—	—	—	—	JEND	JRST	JSRT
Initial value:	0	0	0	0	0	0	0	0
R/W(compress):	R*/W	R	R	R	R	R*/W	Undefined	R*/W
R/W(decompress):	R*/W	R	R	R	R	R*/W	R*/W	R*/W

Note: \* Values read from these bits are undefined.

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
7	BRST	0		R*/W	<b>Bus Reset</b> Setting this bit to 1 resets the internal circuits. While the JCU is in operation (from setting the JPEG core process start command to writing the last output coded/image data), do not set this bit to 1. For the bus reset processing, see section 45.5, Bus Reset Processing.
6 to 3	—	All 0		R	<b>Reserved</b> These bits are always read as 0. The write value should always be 0.
2	JEND	0		R*/W	<b>Interrupt Request Clear Command</b> This bit is valid only for the interrupt sources corresponding to bits INS6, INS5, and INS3 in JINTS0. To clear an interrupt request, set this bit to 1.
1	JRST	0	Invalid	R*/W	<b>JPEG Core Process Stop Clear Command</b> To clear the process-stopped state caused by requests to read the image size and pixel format (enabled by the INT3 bit in JINTE0), set this bit to 1.
0	JSRT	0		R*/W	<b>JPEG Core Process Start Command</b> To start JPEG core processing, set this bit to 1. Do not write this bit to 1 again while the JCU is in operation.

Note: \* Values read from these bits are undefined.

### 45.2.3 JPEG Code Quantization Table Number Register (JCQTN)

JCQTN sets the quantization table number before compression process is started.

- To use quantization table No. 0 (JCQTBL0) as the first color component, set QT1 to B'00
- To use quantization table No. 1 (JCQTBL1) as the first color component, set QT1 to B'01
- To use quantization table No. 2 (JCQTBL2) as the first color component, set QT1 to B'10
- To use quantization table No. 3 (JCQTBL3) as the first color component, set QT1 to B'11

Bit:	7	6	5	4	3	2	1	0
	—	—	QT3[1:0]	QT2[1:0]	QT1[1:0]			
Initial value:	0	0	0	0	0	0	0	0
R/W(compress):	R	R	R/W	R/W	R/W	R/W	R/W	R/W
R/W(decompress):	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
7, 6	—	All 0		R	Reserved These bits are always read as 0. The write value should always be 0.
5, 4	QT3[1:0]	00	R/W	R	Quantization table number for the third color component
3, 2	QT2[1:0]	00	R/W	R	Quantization table number for the second color component
1, 0	QT1[1:0]	00	R/W	R	Quantization table number for the first color component



### 45.2.4 JPEG Code Huffman Table Number Register (JCHTN)

JCHTN sets the Huffman table number (AC/DC) before compression process is started.

- To use DC/AC Huffman table No. 0 (JCHTBD0 and JCHTBA0) as the first color component, set bits HTA1 and HTD1 to B'0
- To use DC/AC Huffman table No. 1 (JCHTBD1 and JCHTBA1) as the first color component, set bits HTA1 and HTD1 to B'1

Bit:	7	6	5	4	3	2	1	0
	—	—	HTA3	HTD3	HTA2	HTD2	HTA1	HTD1
Initial value:	0	0	0	0	0	0	0	0
R/W(compress):	R	R	R/W	R/W	R/W	R/W	R/W	R/W
R/W(decompress):	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
7, 6	—	All 0		R	Reserved These bits are always read as 0. The write value should always be 0.
5	HTA3	0	R/W	R	Huffman table number (AC) for the third color component
4	HTD3	0	R/W	R	Huffman table number (DC) for the third color component
3	HTA2	0	R/W	R	Huffman table number (AC) for the second color component
2	HTD2	0	R/W	R	Huffman table number (DC) for the second color component
1	HTA1	0	R/W	R	Huffman table number (AC) for the first color component
0	HTD1	0	R/W	R	Huffman table number (DC) for the first color component

### 45.2.5 JPEG Code DRI Upper Register (JCDRIU)

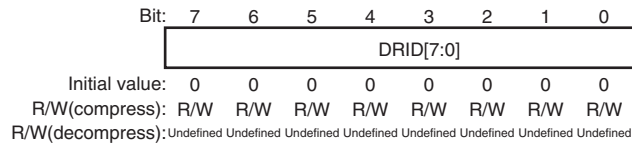
JCDRIU sets the upper bytes of the minimum coded units (MCUs) preceding an RST marker.

Bit:	7	6	5	4	3	2	1	0
	DRIU[7:0]							
Initial value:	0	0	0	0	0	0	0	0
R/W(compress):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W(decompress):	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
7 to 0	DRIU[7:0]	H'00	R/W	Invalid	Upper Bytes of MCUs Preceding RST Marker When both upper and lower bytes are set to H'00, neither a DRI nor an RST marker is placed.

### 45.2.6 JPEG Code DRI Lower Register (JCDRID)

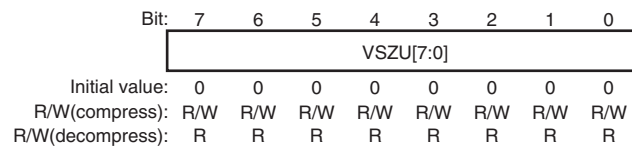
JCDRID sets the lower bytes of MCUs preceding an RST marker.



Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
7 to 0	DRID[7:0]	H'00	R/W	Invalid	Lower Bytes of MCUs Preceding RST Marker When both upper and lower bytes are set to H'00, neither a DRI nor an RST marker is placed.

### 45.2.7 JPEG Code Vertical Size Upper Register (JCVSZU)

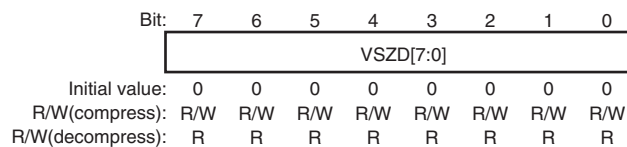
JCVSZU sets the upper bytes of the vertical image size.



Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
7 to 0	VSZU[7:0]	H'00	R/W	R	Upper Bytes of Vertical Image Size In decompression process, a downloaded value from the JPEG coded data is set.

### 45.2.8 JPEG Code Vertical Size Lower Register (JCVSZD)

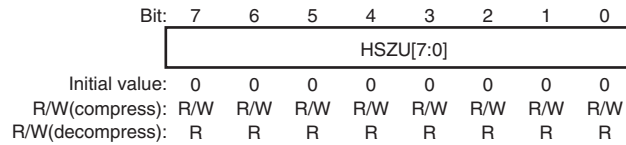
JCVSZD sets the lower bytes of the vertical image size.



Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
7 to 0	VSZD[7:0]	H'00	R/W	R	Lower Bytes of Vertical Image Size In decompression process, a downloaded value from the JPEG coded data is set.

### 45.2.9 JPEG Code Horizontal Size Upper Register (JCHSZU)

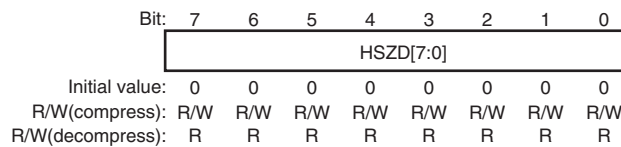
JCHSZU sets the upper bytes of the horizontal image size.



Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
7 to 0	HSZU[7:0]	H'00	R/W	R	Upper Bytes of Horizontal Image Size In decompression process, a downloaded value from the JPEG coded data is set.

### 45.2.10 JPEG Coded Horizontal Size Lower Register (JCHSZD)

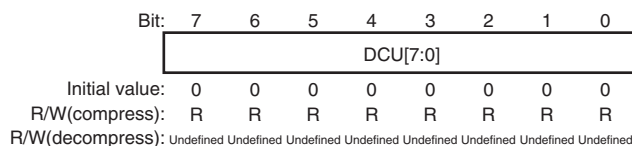
JCHSZD sets the lower bytes of the horizontal image size.



Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
7 to 0	HSZD[7:0]	H'00	R/W	R	Lower Bytes of Horizontal Image Size In decompression process, a downloaded value from the JPEG coded data is set.

### 45.2.11 JPEG Code Data Count Upper Register (JCDTCU)

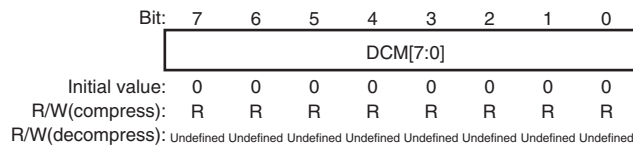
The upper bytes for the counted amount of data to be compressed are set to JCDTCU. The values of this register are reset before compression starts.



Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
7 to 0	DCU[7:0]	H'00	R	Invalid	Upper bytes of the counted amount of data to be compressed

### 45.2.12 JPEG Code Data Count Middle Register (JCDCM)

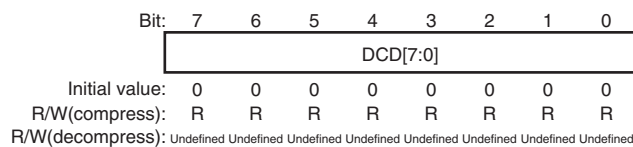
The middle bytes for the counted amount of data to be compressed are set to JCDCM. The values of this register are reset before compression starts.



Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
7 to 0	DCM[7:0]	H'00	R	Invalid	Middle bytes of the counted amount of data to be compressed

### 45.2.13 JPEG Code Data Count Lower Register (JCDCD)

The lower bytes for the counted amount of data to be compressed are set to JCDCD. The values of this register are reset before compression starts.



Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
7 to 0	DCD[7:0]	H'00	R	Invalid	Lower bytes of the counted amount of data to be compressed

### 45.2.14 JPEG Interrupt Enable Register 0 (JINTE0)

JINTE0 enables interrupts.

When any of bits INT7 to INT5 is set to B'1, the INS5 bit in JINTS0 indicates B'1 as the error status upon occurrence of the compression data error, and the ERR bit in JCDERR indicates the particular error code.

Bit:	7	6	5	4	3	2	1	0
	INT7	INT6	INT5	—	INT3	—	—	—
Initial value:	0	0	0	0	0	0	0	0
R/W(compress):	Undefined	Undefined	Undefined	R	Undefined	R	R	R
R/W(decompress):	R/W	R/W	R/W	R	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
7	INT7	0	Invalid	R/W	This bit enables an interrupt to be generated when the number of data in the restart interval of the Huffman-coding segment is not correct in decompression. When this bit is not set to enable interrupt generation, an error code is not returned.
6	INT6	0	Invalid	R/W	This bit enables an interrupt to be generated when the total number of data in the Huffman-coding segment is not correct in decompression. When this bit is not set to enable interrupt generation, an error code is not returned.
5	INT5	0	Invalid	R/W	This bit enables an interrupt to be generated when the final number of MCU data in the Huffman-coding segment is not correct in decompression. When this bit is not set to enable interrupt generation, an error code is not returned.
4	—	0		R	Reserved This bit is always read as 0. The write value should always be 0.
3	INT3	0	Invalid	R/W	This bit enables an interrupt to be generated when it has been determined that the image size and the subsampling setting of the compressed data can be read through analyzing the data.
2 to 0	—	All 0		R	Reserved These bits are always read as 0. The write value should always be 0.

### 45.2.15 JPEG Interrupt Status Register 0 (JINTS0)

JINTS0 identifies the interrupt sources.

The interrupt sources of this register should be cleared by clearing the corresponding interrupt status bits to 0 and setting the relevant bit in JCCMD appropriately.

Bit:	7	6	5	4	3	2	1	0
	—	INS6	INS5	—	INS3	—	—	—
Initial value:	0	0	0	0	0	0	0	0
R/W(compress):	R	R/W*	Undefined	R	Undefined	R	R	R
R/W(decompress):	R	R/W*	R/W*	R	R/W*	R	R	R

Note: \* Clear this bit by writing 0 to it.  
Do not write 1 to this bit.

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
7	—	0		R	Reserved This bit is always read as 0. The write value should always be 0.
6	INS6	0		R/W*	This bit is set to 1 when the JCU completes compression process normally.
5	INS5	0	Invalid	R/W*	This bit is set to 1 when a compressed data error occurs.
4	—	0		R	Reserved This bit is always read as 0. The write value should always be 0.
3	INS3	0	Invalid	R/W*	This bit is set to 1 when the image size and pixel format can be read. When an interrupt occurs, this module stops processing and the state is indicated by the JCRST register. To make the JCU resume processing, set the JPEG core process stop clear command bit (JRST) in JCCMD.
2 to 0	—	All 0		R	Reserved These bits are always read as 0. The write value should always be 0.

Note: \* Clear this bit by writing 0 to it. Do not write 1 to this bit.

### 45.2.16 JPEG Code Decode Error Register (JCDERR)

JCDERR indicates the error code to identify the type of the error which has occurred in the compressed data analysis for decompression. The values of this register are reset before the JCU starts decompression.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	ERR[3:0]			
Initial value:	0	0	0	0	1	0	1	0
R/W(compress):	R	R	R	R	Undefined	Undefined	Undefined	Undefined
R/W(decompress):	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
7 to 4	—	All 0		R	Reserved These bits are always read as 0.
3 to 0	ERR[3:0]	1010	Invalid	R/W	Error Code (See tables 45.3 and 45.4.)

### 45.2.17 JPEG Code Reset Register (JCRST)

JCRST indicates a processing-stopped state caused by requests to read the image size and pixel format (enabled by the INT3 bit in JINTE0). To resume processing, set the JPEG core process stop clear command bit (JRST) in JCCMD.

Bit:	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	RST
Initial value:	0	0	0	0	0	0	0	0
R/W(compress):	R	R	R	R	R	R	R	Undefined
R/W(decompress):	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
7 to 1	—	All 0		R	Reserved These bits are always read as 0.
0	RST	0	Invalid	R/W	Operating State 0: State other than below 1: Suspended state caused by interrupt sources of JINTE0

## 45.2.18 JPEG Interface Compression Control Register (JIFECNT)

JIFECNT controls the compression process.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(compress):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(decompress):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	JOUT RINI	JOUT RCMD	JOUTC	—	JOUTSWAP[2:0]	—	DIN RINI	DIN RCMD	DIN LC	—	DINSWAP[2:0]				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(compress):	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W
R/W(decompress):	R	Undefined	Undefined	Undefined	R	Undefined	Undefined	Undefined	R	Undefined	Undefined	Undefined	R	Undefined	Undefined	Undefined

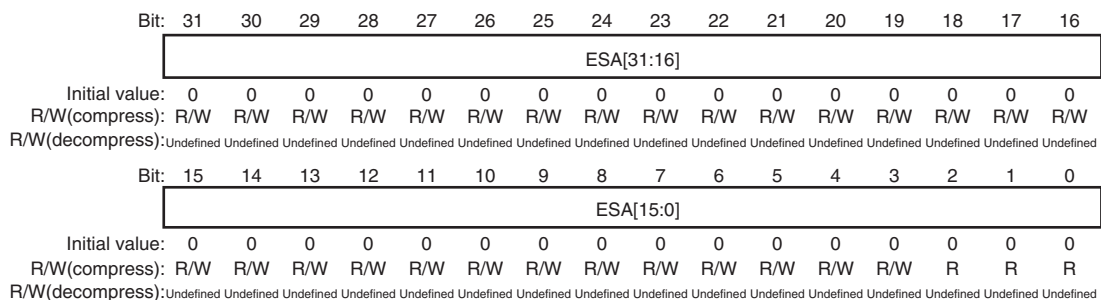
Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
31 to 15	—	All 0		R	Reserved These bits are always read as 0. The write value should always be 0.
14	JOUTRINI	0	R/W	Invalid	Address Initialization when Output Coded Data is Resumed This bit is only valid when the count mode for stopping the output of coded data is on. Set this bit before writing 1 to the data resume command bit. 0: The transfer address is not initialized when the output of coded data is restarted. 1: The transfer address is initialized when the output of coded data is restarted.
13	JOUTR CMD	0	R/W	Invalid	Output Coded Data Resume Command This bit is only valid when the count mode for stopping the output of coded data is on. Setting this bit to 1 resumes writing output coded data. This bit is always read as 0.
12	JOUTC	0	R/W	Invalid	Count Mode Setting for Stopping Output Coded Data 0: Count mode for stopping the output of coded data is off. 1: Count mode for stopping the output of coded data is on.
11	—	0		R	Reserved This bit is always read as 0. The write value should always be 0.
10 to 8	JOUT SWAP[2:0]	000	R/W	Invalid	Byte/Word/Longword Swap Output coded data in compression is swapped. 000: (1) (2) (3) (4) (5) (6) (7) (8) 001: (2) (1) (4) (3) (6) (5) (8) (7) [Byte swap] 010: (3) (4) (1) (2) (7) (8) (5) (6) [Word swap] 011: (4) (3) (2) (1) (8) (7) (6) (5) [Word - byte swap] 100: (5) (6) (7) (8) (1) (2) (3) (4) [Longword swap] 101: (6) (5) (8) (7) (2) (1) (4) (3) [Longword - byte swap] 110: (7) (8) (5) (6) (3) (4) (1) (2) [Longword - word swap] 111: (8) (7) (6) (5) (4) (3) (2) (1) [Longword - word - byte swap]
7	—	0		R	Reserved This bit is always read as 0. The write value should always be 0.



Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
6	DINRINI	0	R/W	Invalid	Address Initialization when Resuming Input of Image Data Lines This bit is only valid when the count mode for stopping the input of image data lines is on. Set this bit before writing 1 to the data-line resume command bit. 0: The transfer address is not initialized when the input of image data lines is restarted. 1: The transfer address is initialized when the input of image data lines is restarted.
5	DINRCMD	0	R/W	Invalid	Input Image Data Lines Resume Command This bit is valid only when the count mode for stopping the input of image data lines is on. Setting this bit to 1 resumes reading input image data. This bit is always read as 0.
4	DINLC	0	R/W	Invalid	Count Mode Setting for Stopping Input Image Data Lines 0: Count mode for stopping the input of image data lines is off. 1: Count mode for stopping the input of image data lines is on.
3	—	0		R	Reserved This bit is always read as 0. The write value should always be 0.
2 to 0	DINSWAP [2:0]	000	R/W	Invalid	Byte/Word Swap Input image data in compression is swapped. 000: (1) (2) (3) (4) (5) (6) (7) (8) 001: (2) (1) (4) (3) (6) (5) (8) (7) [Byte swap] 010: (3) (4) (1) (2) (7) (8) (5) (6) [Word swap] 011: (4) (3) (2) (1) (8) (7) (6) (5) [Word - byte swap] 100: (5) (6) (7) (8) (1) (2) (3) (4) [Longword swap] 101: (6) (5) (8) (7) (2) (1) (4) (3) [Longword - byte swap] 110: (7) (8) (5) (6) (3) (4) (1) (2) [Longword - word swap] 111: (8) (7) (6) (5) (4) (3) (2) (1) [Longword - word - byte swap]

### 45.2.19 JPEG Interface Compression Source Address Register (JIFESA)

JIFESA sets the source address of the input image data. This register should be set in 8-byte units.



Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
31 to 3	ESA[31:3]	H'0000	R/W	Invalid	Input Image Data Source Address (in 8-byte units)
2 to 0	ESA[2:0]	0000	R		The lower three bits should be set to 0.

### 45.2.20 JPEG Interface Compression Line Offset Register (JIFESOFST)

JIFESOFST sets the line offset of the input image data (refer to section 45.3.4, Storing Image Data). This register should be set in 8-byte units.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(compress):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(decompress):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	ESMW[14:0]														
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(compress):	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R
R/W(decompress):	R	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
31 to 15	—	All 0		R	Reserved These bits are always read as 0. The write value should always be 0.
14 to 3	ESMW [14:3]	H'0000	R/W	Invalid	Input Image Data Lines Offset (in 8-byte units) The lower three bits should be set to 0.
2 to 0	ESMW[2:0]		R		

### 45.2.21 JPEG Interface Compression Destination Address Register (JIFEDA)

JIFEDA sets the destination address of the output coded data. This register should be set in 8-byte units.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	EDA[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(compress):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
R/W(decompress):	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EDA[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(compress):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R
R/W(decompress):	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
31 to 3	EDA[31:3]	H'0000	R/W	Invalid	Output Coded Data Destination Address (in 8-byte units)
2 to 0	EDA[2:0]	0000	R		The lower three bits should be set to 0.

### 45.2.22 JPEG Interface Compression Source Line Count Register (JIFESLC)

JIFESLC sets the number of input image data lines when the count mode for stopping the input of image data lines is on (the DINLC bit in JIFECNT is set to 1). This register should be set in 8-line units.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0
R/W(compress):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(decompress):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LINES[15:0]															
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0
R/W(compress):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R
R/W(decompress):	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
31 to 16	—	H'FFF8		R	Reserved Values read from these bits are undefined. The write value should always be 0.
15 to 3	LINES[15:3]	H'FFF8	R/W	Invalid	Number of Input Image Data Lines to be Read (in 8-line units)
2 to 0	LINES[2:0]		R		The lower three bits should be set to 0.

### 45.2.23 JPEG Interface Compression Destination Count Register (JIFEDDC)

JIFEDDC sets the amount of output coded data when the count mode for stopping the output of coded data is on (the JOUTC bit in JIFECNT is set to 1). This register should be set in 8-byte units.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W(compress):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(decompress):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	JDATAS[15:0]															
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0
R/W(compress):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R
R/W(decompress):	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
31 to 16	—	H'FFF8	R	Invalid	Reserved Values read from these bits are undefined. The write value should always be 0.
15 to 3	JDATAS[15:3]	H'FFF8	R/W	Invalid	Amount of Output Coded Data to be Written (in 8-byte units)
2 to 0	JDATAS[2:0]		R		The lower three bits should be set to 0.

## 45.2.24 JPEG Interface Decompression Control Register (JIFDCNT)

JIFDCNT controls the decompression process.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	VINTER[1:0]	HINTER[1:0]	OPF[1:0]	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(compress):	R	R	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	R	R	R	R	R	R	R	R
R/W(decompress):	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	JINRINI	JINRCMD	JINC	—	JINSWAP[2:0]	—	DOUTRINI	DOUTRCMD	DOUTLD	—	DOUTSWAP[2:0]	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(compress):	R	Undefined	Undefined	Undefined	R	Undefined	Undefined	Undefined	R	Undefined	Undefined	Undefined	R	Undefined	Undefined	Undefined
R/W(decompress):	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
31, 30	—	All 0		R	Reserved These bits are always read as 0. The write value should always be 0.
29, 28	VINTER[1:0]	00	Invalid	R/W	Vertical Subsampling Subsamples vertical output image data. 00: No subsampling 01: Subsamples output data into 1/2. 10: Subsamples output data into 1/4. 11: Subsamples output data into 1/8.
27, 26	HINTER[1:0]	00	Invalid	R/W	Horizontal Subsampling Subsamples horizontal output image data. 00: No subsampling 01: Subsamples output data into 1/2. 10: Subsamples output data into 1/4. 11: Subsamples output data into 1/8.
25, 24	OPF[1:0]	00	Invalid	R/W	Specifies output image data pixel format. 00: YCbCr422 01: ARGB8888 10: RGB565 11: Setting prohibited
23 to 15	—	All 0		R	Reserved These bits are always read as 0. The write value should always be 0.
14	JINRINI	0	Invalid	R/W	Address Initialization when Input Coded Data is Resumed This bit is only valid when the count mode for stopping the input of coded data is on. Set this bit before writing 1 to the data resume command bit. 0: The transfer address is not initialized when the input of coded data is restarted. 1: The transfer address is initialized when the input of coded data is restarted.
13	JINRCMD	0	Invalid	R/W	Input Coded Data Resume Command This bit is valid only when the count mode for stopping the input of coded data is on. Setting this bit to 1 resumes reading input coded data. This bit is always read as 0.
12	JINC	0	Invalid	R/W	Count Mode Setting for Stopping Input Coded Data 0: Count mode for stopping the input of coded data is off. 1: Count mode for stopping the input of coded data is on.
11	—	0		R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
10 to 8	JINSWAP [2:0]	000	Invalid	R/W	Byte/Word/Longword Swap Input coded data in decompression is swapped. 000: (1) (2) (3) (4) (5) (6) (7) (8) 001: (2) (1) (4) (3) (6) (5) (8) (7) [Byte swap] 010: (3) (4) (1) (2) (7) (8) (5) (6) [Word swap] 011: (4) (3) (2) (1) (8) (7) (6) (5) [Word - byte swap] 100: (5) (6) (7) (8) (1) (2) (3) (4) [Longword swap] 101: (6) (5) (8) (7) (2) (1) (4) (3) [Longword - byte swap] 110: (7) (8) (5) (6) (3) (4) (1) (2) [Longword - word swap] 111: (8) (7) (6) (5) (4) (3) (2) (1) [Longword - word - byte swap]
7	—	0		R	Reserved This bit is always read as 0. The write value should always be 0.
6	DOUTRINI	0	Invalid	R/W	Address Initialization when Resuming Output of Image Data Lines This bit is only valid when the count mode for stopping the output of image data lines is on. Set this bit before writing 1 to the data-line resume command bit. 0: The transfer address is not initialized when the output of lines of image data is restarted. 1: The transfer address is initialized when the output of lines of image data is restarted.
5	DOUTRCMD	0	Invalid	R/W	Output Image Data Lines Resume Command This bit is valid only when the count mode for stopping the output of image data lines is on. Setting this bit to 1 resumes writing image data. This bit is always read as 0.
4	DOUTLC	0	Invalid	R/W	Count Mode for Stopping Output Image Data Lines 0: Count mode for stopping the output of image data lines is off. 1: Count mode for stopping the output of image data lines is on.
3	—	0		R	Reserved This bit is always read as 0. The write value should always be 0.
2 to 0	DOUTSWAP [2:0]	000	Invalid	R/W	Byte/Word Swap Output image data in decompression is swapped. 000: (1) (2) (3) (4) (5) (6) (7) (8) 001: (2) (1) (4) (3) (6) (5) (8) (7) [Byte swap] 010: (3) (4) (1) (2) (7) (8) (5) (6) [Word swap] 011: (4) (3) (2) (1) (8) (7) (6) (5) [Word - byte swap] 100: (5) (6) (7) (8) (1) (2) (3) (4) [Longword swap] 101: (6) (5) (8) (7) (2) (1) (4) (3) [Longword - byte swap] 110: (7) (8) (5) (6) (3) (4) (1) (2) [Longword - word swap] 111: (8) (7) (6) (5) (4) (3) (2) (1) [Longword - word - byte swap]

### 45.2.25 JPEG Interface Decompression Source Address Register (JIFDSA)

JIFDSA sets the source address of the input coded data. This register should be set in 8-byte units.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DSA[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(compress):	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W(decompress):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DSA[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(compress):	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W(decompress):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
31 to 3	DSA[31:3]	H'0000	Invalid	R/W	Input Coded Data Source Address (in 8-byte units)
2 to 0	DSA[2:0]	0000		R	The lower three bits should be set to 0.

### 45.2.26 JPEG Interface Decompression Line Offset Register (JIFDDOFST)

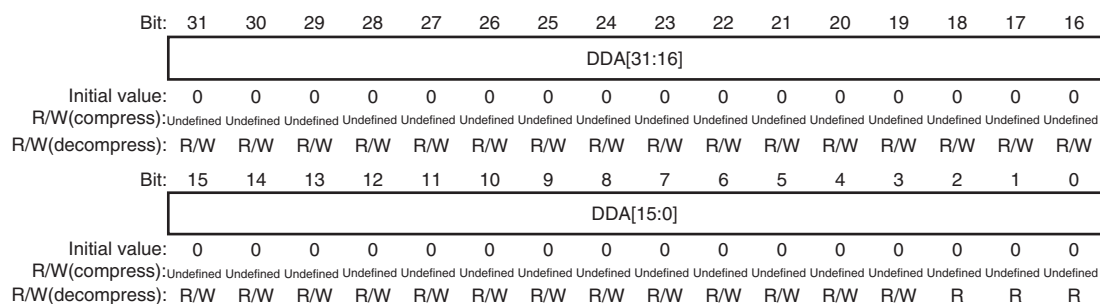
JIFDDOFST sets the line offset of the output image data to be transferred to the external buffer (refer to section 45.3.4, Storing Image Data). This register should be set in 8-byte units.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(compress):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(decompress):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	DDMW[14:0]														
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(compress):	R	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W(decompress):	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
31 to 16	—	All 0		R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 3	DDMW[14:3]	H'0000	Invalid	R/W	Output Image Data Lines Offset (in 8-byte units)
2 to 0	DDMW[2:0]			R	The lower three bits should be set to 0.

## 45.2.27 JPEG Interface Decompression Destination Address Register (JIFDDA)

JIFDDA sets the destination address of the output image data. This register should be set in 8-byte units.



Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
31 to 3	DDA[31:3]	H'0000	Invalid	R/W	Output Image Data Destination Address (in 8-byte units)
2 to 0	DDA[2:0]	0000		R	The lower three bits should be set to 0.

### 45.2.28 JPEG Interface Decompression Source Data Count Register (JIFDSDC)

JIFDSDC sets the amount of input coded data when the count mode for stopping the input of coded data is on (the JINC bit in JIFDCNT is set to 1). This register should be set in 8-byte units.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0
R/W(compress):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(decompress):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	JDATAS[15:0]															
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0
R/W(compress):	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W(decompress):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
31 to 16	—	H'FFF8		R	Reserved Values read from these bits are undefined. The write value should always be 0.
15 to 3	JDATAS[15:3]	H'FFF8	Invalid	R/W	Amount of Input Coded Data to be Read (in 8-byte units)The lower three bits should be set to 0.
2 to 0	JDATAS[2:0]			R	

### 45.2.29 JPEG Interface Decompression Destination Line Count Register (JIFDDLCL)

JIFDDLCL sets the number of lines of output image data when the count mode for stopping the output of image data lines is on (the DOUTLCL bit in JIFECNT is set to 1). This register is used to set the number of lines of output image data in MCU units.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0
R/W(compress):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(decompress):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LINES[15:0]															
Initial value:	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0
R/W(compress):	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W(decompress):	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
31 to 16	—	H'FFF8		R	Reserved Values read from these bits are undefined. The write value should always be 0.
15 to 3	LINES[15:3]	H'FFF8	Invalid	R/W	Specify the number of lines of output image data to be written. The setting is in MCU units. When data are to be output in YCbCr444, YCbCr422 or YCbCr411 format, the number of lines of output image data is <this setting> x 1. When data are to be output in YCbCr420 format, the number of lines of output image data is <this setting> x 2. The lower three bits should be set to 0.
2 to 0	LINES[2:0]			R	



### 45.2.30 JPEG Interface Decompression $\alpha$ Set Register (JIFDADT)

JIFDADT is used to set the  $\alpha$  value when output is in ARGB8888 format.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(compress):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(decompress):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ALPHA[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(compress):	R	R	R	R	R	R	R	R	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W(decompress):	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
31 to 8	—	All 0		R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	ALPHA[7:0]	H'00	Invalid	R/W	Setting of the $\alpha$ value for output in ARGB8888 format.

## 45.2.31 JPEG Interrupt Enable Register 1 (JINTE1)

JINTE1 enables interrupts.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(compress):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(decompress):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	CBTEN	DIN LEN	JOU TEN	—	DBTEN	JINEN	DOU TLEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(compress):	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	Undefined	Undefined	Undefined
R/W(decompress):	R	R	R	R	R	R	R	R	R	Undefined	Undefined	Undefined	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
31 to 7	—	All 0		R	Reserved These bits are always read as 0. The write value should always be 0.
6	CBTEN	0	R/W	Invalid	Enables or disables a data transfer processing interrupt request (JDTI) when the CBTF bit in JINTS1 is set to 1. 0: Disables an interrupt request. 1: Enables an interrupt request.
5	DINLEN	0	R/W	Invalid	Enables or disables a data transfer processing interrupt request (JDTI) when the DINLF bit in JINTS1 is set to 1. 0: Disables an interrupt request. 1: Enables an interrupt request.
4	JOUTEN	0	R/W	Invalid	Enables or disables a data transfer processing interrupt request (JDTI) when the JOUTF bit in JINTS1 is set to 1. 0: Disables an interrupt request. 1: Enables an interrupt request.
3	—	0		R	Reserved This bit is always read as 0. The write value should always be 0.
2	DBTEN	0	Invalid	R/W	Enables or disables a data transfer processing interrupt request (JDTI) when the DBTF bit in JINTS1 is set to 1. 0: Disables an interrupt request. 1: Enables an interrupt request.
1	JINEN	0	Invalid	R/W	Enables or disables a data transfer processing interrupt request (JDTI) when the JINF bit in JINTS1 is set to 1. 0: Disables an interrupt request. 1: Enables an interrupt request.
0	DOUTLEN	0	Invalid	R/W	Enables or disables a data transfer processing interrupt request (JDTI) when the DOUTLF bit in JINTS1 is set to 1. 0: Disables an interrupt request. 1: Enables an interrupt request.

### 45.2.32 JPEG Interrupt Status Register 1 (JINTS1)

JINTS1 indicates the interrupt sources.

The interrupt sources of this register should be cleared by writing 0 to this register.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(compress):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(decompress):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	CBTF	DINLF	JOUTF	—	DBTF	JINF	DOU TLF
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(compress):	R	R	R	R	R	R	R	R	R	R/W*	R/W*	R/W*	R	Undefined	Undefined	Undefined
R/W(decompress):	R	R	R	R	R	R	R	R	R	Undefined	Undefined	Undefined	R	R/W*	R/W*	R/W*

Note: \* When the bit is read as 1, write 0 to clear it.  
When the bit is read as 0, write 1 to it.

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
31 to 7	—	All 0		R	Reserved These bits are always read as 0. The write value should always be 0.
6	CBTF	0	R/W*	Invalid	This bit is set to 1 when the last output coded data is written in compression.
5	DINLF	0	R/W*	Invalid	This bit is set to 1 when the number of input image data lines indicated by JIFESLC is read in compression. This bit is valid only when the DINLC bit in JIFECNT is set to 1.
4	JOUTF	0	R/W*	Invalid	This bit is set to 1 when the amount of output coded data indicated by JIFEDDC is written in compression. This bit is valid only when the JOUTC bit in JIFECNT is set to 1.
3	—	0		R	Reserved This bit is always read as 0. The write value should always be 0.
2	DBTF	0	Invalid	R/W*	This bit is set to 1 when the last output image data is written in decompression.
1	JINF	0	Invalid	R/W*	This bit is set to 1 when the amount of input coded data indicated by JIFSDC is read in decompression. This bit is valid only when the JINC bit in JIFDCNT is set to 1.
0	DOUTLF	0	Invalid	R/W*	In decompression, this bit is set to 1 when the number of lines of output image data indicated by JIFDDL have been written. This bit is only valid when the DOUTC bit in JIFDCNT is set to 1.

Note: \* When the bit is read as 1, write 0 to clear it.  
When the bit is read as 0, write 1 to it.

### 45.2.33 JPEG Input Image Data CbCr Range Setting Register (JIFESVSZ)

JIFESVSZ sets the CbCr range of input image data.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(compress):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(decompress):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DINY CHG	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(compress):	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(decompress):	Undefined	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
31 to 16	—	All 0		R	Reserved These bits are always read as 0. The write value should always be 0.
15	DINYCHG	0	R/W	Invalid	Input Image Data CbCr Range Setting 0: Range from -128 to 127 1: Range from 0 to 255
14 to 0	—	All 0		R	Reserved These bits are always read as 0. The write value should always be 0.

### 45.2.34 JPEG Output Image Data CbCr Range Setting Register (JIFESHSZ)

JIFESHSZ sets the CbCr range of output image data.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(compress):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(decompress):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DOUTY CHG	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W(compress):	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W(decompress):	Undefined	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
31 to 16	—	All 0		R	Reserved These bits are always read as 0. The write value should always be 0.
15	DOUTYCHG	0	Invalid	R/W	Output Image Data CbCr Range Setting 0: Range from -128 to 127 1: Range from 0 to 255
14 to 0	—	All 0		R	Reserved These bits are always read as 0. The write value should always be 0.

## 45.3 Operation

### 45.3.1 Compression

#### (1) Overview of Processing

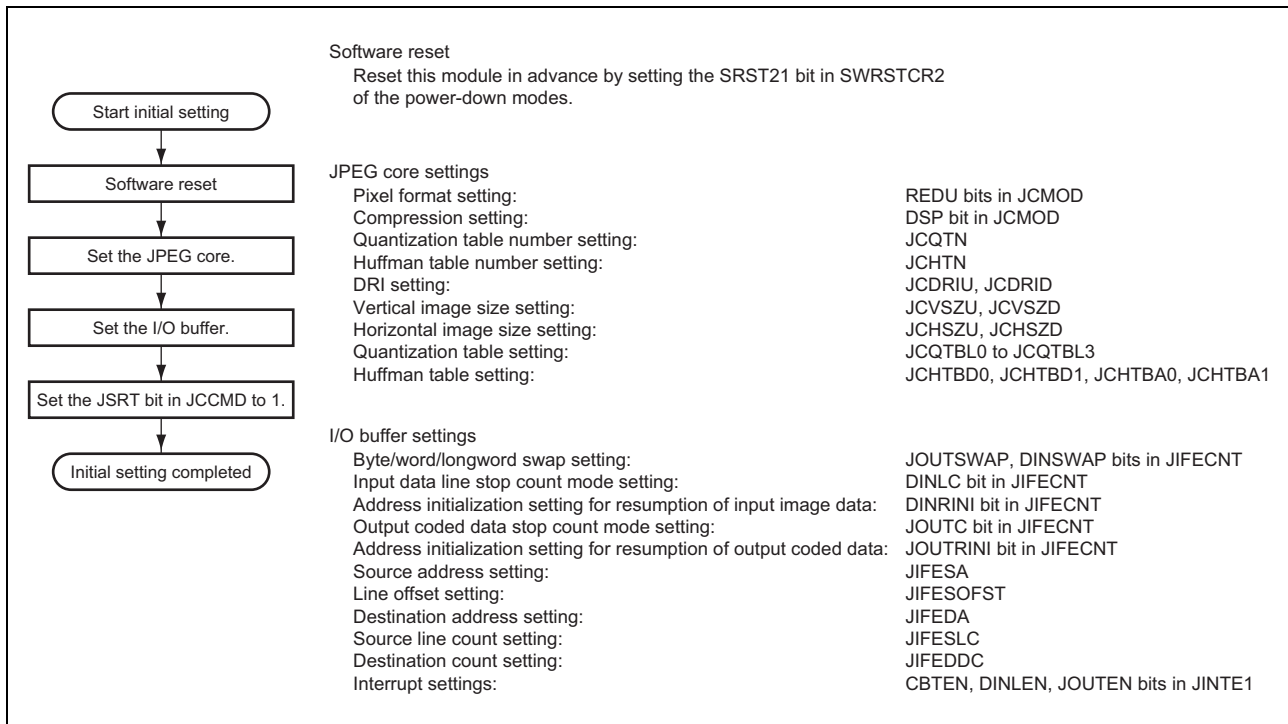
The compression process flows are described below.

1. The JPEG core is activated.  
A marker is output. (After a marker is output, image data can be input.)  
Approximately 30,000 cycles (necessary for making SOI to SOS markers)
2. Image data is transferred in MCUs from the external buffer to the JCU.  
If the count mode for stopping the input of image data lines is on, reading is stopped each time the number of lines set in JIFESLC is read. Reading is resumed by setting the DINRCMD bit in JIFECNT to 1.  
When the DINRINI bit in JIFECNT is zero, the addresses for reading on resumption are continued from the addresses in the previous round of transfer.  
When the DINRINI bit is one, the address set in JIFESA is used on resumption.  
Reading is also stopped when one frame of image data is completely transferred.  
If the count mode for stopping the input of image data lines is off, reading is continued until one frame of image data is completely transferred.
3. Image data is input to the JPEG core.  
The input data is processed in MCUs at any time in the JPEG core.
4. Coded data is transferred from the JCU to the external buffer.  
When the count mode for stopping the output of coded data is on, writing is stopped each time the amount of coded data set in JIFEDDC is written. Writing is resumed by setting the JOUTRCMD bit in JIFECNT to 1.  
When the JOUTRINI bit in JIFECNT is zero, the addresses for writing on resumption are continued from the addresses in the previous round of transfer.  
When the JOUTRINI bit is one, the address set in JIFEDA is used on resumption.  
Writing is also stopped when one frame of coded data is completely transferred.  
If the count mode for stopping the output of coded data is off, writing is continued until one frame of coded data is completely transferred.
5. Compression is completed after one frame of data is processed completely.

## (2) Flowchart (Compression)

### (a) Initial Settings

After completing the JPEG core settings and input/output buffer settings and transferring image data to the external buffer, activate this module by setting the JSRT bit in JCCMD to 1. After the JCU has been activated, the JPEG markers (SOI to SOS) are generated and output. It takes approximately 30,000 cycles to generate the markers.



**Figure 45.2 Compression Initial Setting Flow**

### (b) Compression Process

The compression process flows are described below.

- When JPEG compression process has been completed, the INS6 bit in JINTS0 is set to 1. However, the JCU continues processing since the coded data remains to be transferred. The CBTF bit in JINTS1 is set to 1 when the last coded data is transferred. The interrupt source is cleared by writing 0 to the INS6 bit. However, the interrupt request asserted by this interrupt source cannot be cleared by writing 0 to the INS6 bit. Set an interrupt request clear command (by setting the JEND bit in JCCMD to 1) to clear the interrupt request.
- When the JCU has completed compression and all coded data has been transferred, the CBTF flag in JINTS1 is set to 1. When the CBTEN bit in JINTE1 is 1 here, an interrupt is generated. The interrupt source is cleared by writing 0 to the CBTF flag.
- If the count mode for stopping image data lines is on, when the specified number of image data lines set in JIFESLC has been read, the DINLF flag in JINTS1 is set to 1, and reading is stopped. When the DINLEN bit in JINTE1 is 1 here, an interrupt is generated. An interrupt source is cleared by writing 0 to the DINLEN bit. Setting the DINRCMD bit in JIFECNT to 1 resumes reading.  
When the DINRINI bit in JIFECNT is zero, the addresses for reading on resumption are continued from the addresses in the previous round of transfer.  
When the DINRINI bit is one, the address set in JIFESA is used on resumption.
- If the count mode for stopping the output of coded data is on, when the specified amount of coded data set in JIFEDDC has been written, the JOUTF flag in JINTS1 is set to 1, and writing is stopped. When the JOUTEN bit in JINTE1 is 1 here, an interrupt is generated. An interrupt source is cleared by writing 0 to the JOUTF bit. Setting the JOUTRCMD bit in JIFECNT to 1 resumes writing.

When the JOUTRINI bit in JIFECNT is zero, the addresses for writing on resumption are continued from the addresses in the previous round of transfer.

When the JOUTRINI bit is one, the address set in JIFEDA is used on resumption.

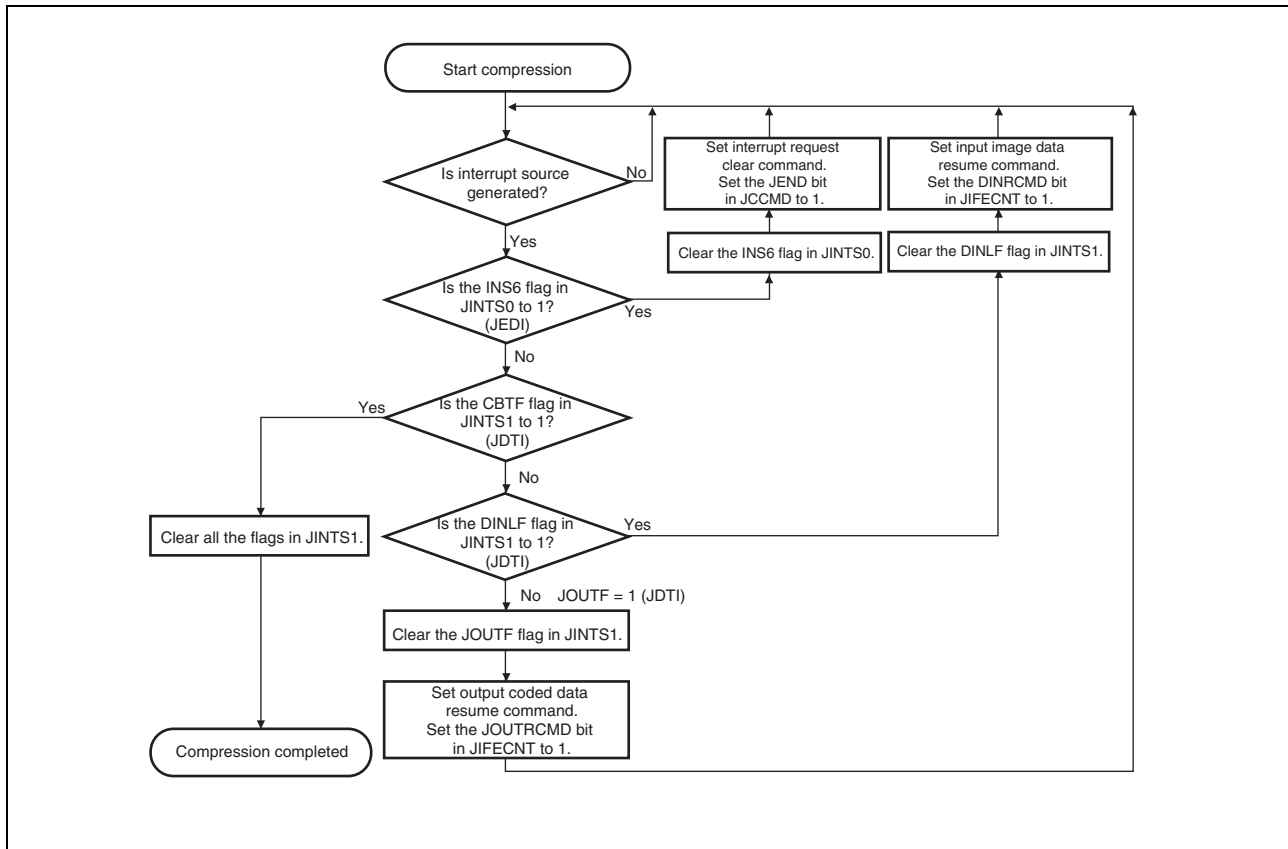


Figure 45.3 Compression Process Flow

### (3) JPEG Coded Data Format

Figure 45.4 shows the data output stream in compression. The amount of coded data from SOI to EOI is indicated by JCDTCU, JCDTCM, and JCDTCD. When both JCDRIU and JCDRID are set to H'0000 0000, the following markers are not output.

- DRI marker
- RST marker (in compressed image data)

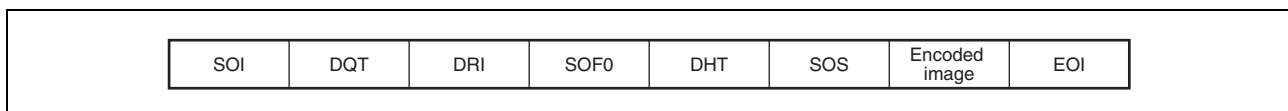


Figure 45.4 JPEG Coded Data Format

DQT: Not output for unused table.

DHT: Output in order DC0, AC0, DC1, and AC1. Not output for unused table.

SOF0: Component identifiers are C1 = first color component, C2 = second color component, and C3 = third color component.

SOS: Scan component selectors are CS1 = first color component, CS2 = second color component, and CS3 = third color component.

**Header Volume (Reference):**

- SOI: 2 bytes (FFD8)
- DQT: 134 bytes when two quantization tables are used, 199 bytes when three quantization tables are used ( $\pm 65$  bytes/table increase or decrease)
- DRI: 6 bytes
- SOF0: 19 bytes (4:2:2)
- DHT: 420 bytes (two tables are used)
- SOS: 14 bytes (4:2:2)
- EOI: 2 bytes (FFD9)

**(4) Table Setting****(a) Quantization Table Specification**

The order of addresses shown in  $8 \times 8$  blocks corresponds to that of the register addresses. Do not access this table while the JCU is in processing.

**Table 45.2 Quantization Table**

00	01	02	03	04	05	06	07
08	09	0A	0B	0C	0D	0E	0F
10	11	12	13	14	15	16	17
18	19	1A	1B	1C	1D	1E	1F
20	21	22	23	24	25	26	27
28	29	2A	2B	2C	2D	2E	2F
30	31	32	33	34	35	36	37
38	39	3A	3B	3C	3D	3E	3F

JCQTBL0 (H'E801 7100) = H'00

JCQTBL0 (H'E801 7101) = H'01

JCQTBL0 (H'E801 7102) = H'02

JCQTBL0 (H'E801 7103) = H'03

:

JCQTBL0 (H'E801 713F) = H'3F

**(b) Huffman Table Specification**

Examples of the Huffman table specification given in the ITU-T T81 Annex K.3.3 recommended by JPEG are shown below. In compression, the following settings must be specified for all the codes so that Huffman codes can be generated for all the group numbers.

- DC Huffman table: The number of codes for each code length is 12.  
The group numbers in order of frequency of occurrence are 12.
- AC Huffman table: The number of codes for each code length is 162.  
The zero run length/the group numbers in order of frequency of occurrence are 162.



Do not access the following tables while the JCU is in processing. In particular, read access is prohibited.

- Table K.3/T81

JCHTBD0 (H'E801 7200) = H'00  
JCHTBD0 (H'E801 7201) = H'01  
JCHTBD0 (H'E801 7202) = H'05  
JCHTBD0 (H'E801 7203) = H'01  
:  
JCHTBD0 (H'E801 721B) = H'0B

- Table K.4/T81

JCHTBD1 (H'E801 7300) = H'00  
JCHTBD1 (H'E801 7301) = H'03  
JCHTBD1 (H'E801 7302) = H'01  
JCHTBD1 (H'E801 7303) = H'01  
:  
JCHTBD1 (H'E801 731B) = H'0B

- Table K.5/T81

JCHTBA0 (H'E801 7220) = H'00  
JCHTBA0 (H'E801 7221) = H'02  
JCHTBA0 (H'E801 7222) = H'01  
JCHTBA0 (H'E801 7223) = H'03  
:  
JCHTBA0 (H'E801 72D1) = H'FA

- Table K.6/T81

JCHTBA1 (H'E801 7320) = H'00  
JCHTBA1 (H'E801 7321) = H'02  
JCHTBA1 (H'E801 7322) = H'01  
JCHTBA1 (H'E801 7323) = H'02  
:  
JCHTBA1 (H'E801 73D1) = H'FA

### (5) Input Pixel Format

Image data in the YCbCr422 format can be input to this module. Allocation of data in the YCbCr422 format can be changed by the DINSWAP bits in JIFECNT as shown below.

- When the DINSWAP bits = 000

b63	b56	b55	b48	b47	b40	b39	b32	b31	b24	b23	b16	b15	b8	b7	b0
Y0 8 bits	Cb0 8 bits	Y1 8 bits	Cr0 8 bits	Y2 8 bits	Cb1 8 bits	Y3 8 bits	Cr1 8 bits								

- When the DINSWAP bits = 001

b63	b56	b55	b48	b47	b40	b39	b32	b31	b24	b23	b16	b15	b8	b7	b0
Cb0 8 bits	Y0 8 bits	Cr0 8 bits	Y1 8 bits	Cb1 8 bits	Y2 8 bits	Cr1 8 bits	Y3 8 bits								

- When the DINSWAP bits = 010

b63	b56	b55	b48	b47	b40	b39	b32	b31	b24	b23	b16	b15	b8	b7	b0
Y1 8 bits	Cr0 8 bits	Y0 8 bits	Cb0 8 bits	Y3 8 bits	Cr1 8 bits	Y2 8 bits	Cb1 8 bits								

- When the DINSWAP bits = 100

b63	b56	b55	b48	b47	b40	b39	b32	b31	b24	b23	b16	b15	b8	b7	b0
Y2 8 bits	Cb1 8 bits	Y3 8 bits	Cr1 8 bits	Y0 8 bits	Cb0 8 bits	Y1 8 bits	Cr0 8 bits								

- When the DINSWAP bits = 101

b63	b56	b55	b48	b47	b40	b39	b32	b31	b24	b23	b16	b15	b8	b7	b0
Cb1 8 bits	Y2 8 bits	Cr1 8 bits	Y3 8 bits	Cb0 8 bits	Y0 8 bits	Cr0 8 bits	Y1 8 bits								

- When the DINSWAP bits = 110

b63	b56	b55	b48	b47	b40	b39	b32	b31	b24	b23	b16	b15	b8	b7	b0
Y3 8 bits	Cr1 8 bits	Y2 8 bits	Cb1 8 bits	Y1 8 bits	Cr0 8 bits	Y0 8 bits	Cb0 8 bits								

- When the DINSWAP bits = 111

b63	b56	b55	b48	b47	b40	b39	b32	b31	b24	b23	b16	b15	b8	b7	b0
Cr1 8 bits	Y3 8 bits	Cb1 8 bits	Y2 8 bits	Cr0 8 bits	Y1 8 bits	Cb0 8 bits	Y0 8 bits								

### (6) Output Coded Data

In the case of compression, coded data are output. This module handles the output of coded data in 16-bit units. For this reason, if the coded data have an odd code length (are fractional), the final code for output will be H'D9FF.

The JOUTSWAP bits in JIFECNT can be used to alter the arrangement of coded data in the output.

## 45.3.2 Decompression

### (1) Overview of Processing

The decompression process flows are described below.

1. The JPEG core is activated.
2. Coded data is transferred from the external buffer to the JCU.  
If the count mode for stopping the input of coded data is on, reading is stopped each time the amount of coded data set in JIFDSLCL is read. Reading is resumed by setting the JINRCMD bit in JIFDCNT to 1. When the JINRINI bit in JIFDCNT is zero, the addresses for reading on resumption are continued from the addresses in the previous round of transfer.  
When the JINRINI bit is one, the address set in JIFDSA is used on resumption. Reading is stopped when the end of the coded data is detected.  
If the count mode for stopping the input of coded data is off, reading is continued until the end of code is detected. With this module, more coded data may be read than the coded data size since coded data reading is continued until the end of code is detected.
3. Coded data is input to the JPEG core.  
The input data is processed in MCUs at any time in the JPEG core.
4. Image data is transferred in MCUs from the JCU to the external buffer.  
When the count mode for stopping the output of image data lines is on, writing is stopped each time the number of image data lines set in JIFDDLCL is written. Writing is resumed by setting the DOUTRCMD bit in JIFECNT to 1. When the DOUTRINI bit in JIFDCNT is zero, the addresses for writing on resumption are continued from the addresses in the previous round of transfer.  
When the DOUTRINI bit is one, the address set in JIFDDA is used on resumption. Writing is stopped when one frame of image data is completely transferred.  
If the count mode for stopping the output of image data lines is off, writing is continued until one frame of image data is completely transferred.
5. Decompression is completed after one frame of data is processed completely.

(a) Initial Settings

- When the INT3 bit in JINTE0 is set to 0:  
After completing the JPEG core settings and input/output buffer settings and transferring coded data to the external buffer, activate this module by setting the JSRT bit in JCCMD to 1.
- When the INT3 bit in JINTE0 is set to 1:  
After completing the JPEG core settings and input buffer settings and transferring coded data to the external buffer, activate this module by setting the JSRT bit in JCCMD to 1.  
When the image size and pixel format become readable after the coded data has been decompressed, the INS3 bit in JINTS0 is set. At this time, decompression is temporarily stopped.  
After the image size and pixel format have been read, set the output buffer.  
Setting the JRST bit in JCCMD to 1 after interrupt handling resumes decompression.

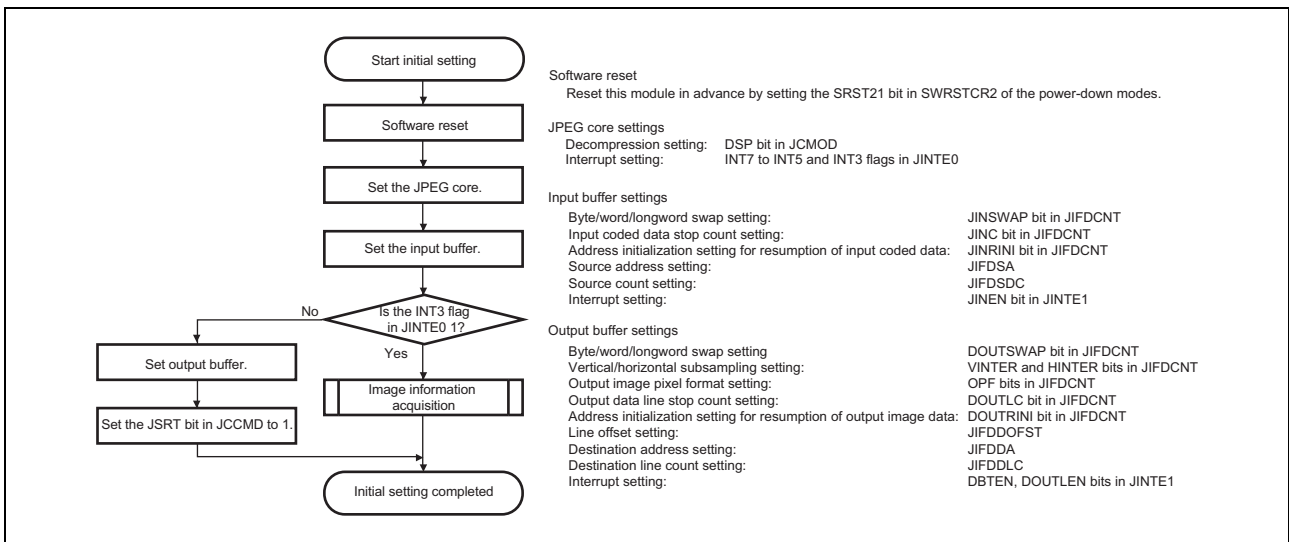


Figure 45.5 Decompression Initial Setting Flow

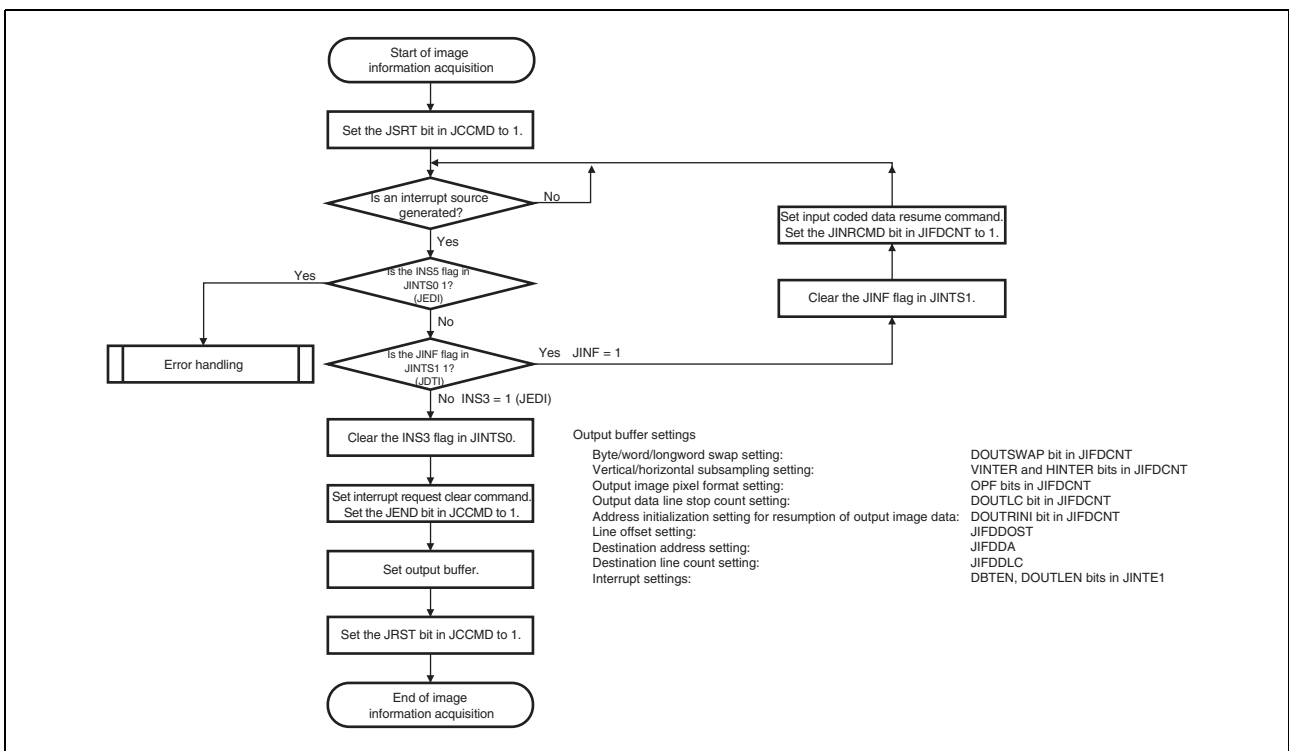


Figure 45.6 Image Information Acquisition Flow

## (b) Decompression Process

The decompression process flows are described below.

- When JPEG decompression process has been completed, the INS6 bit in JINTS0 is set to 1. However, the JCU continues processing since the image data remains to be transferred. The DBTF bit in JINTS1 is set to 1 when the last image data is transferred. The interrupt source is cleared by writing 0 to the INS6 bit. However, the interrupt request asserted by this interrupt source cannot be cleared by writing 0 to the INS6 bit. Set an interrupt request clear command (by setting the JEND bit in JCCMD to 1) to clear the interrupt request.
- When the JCU has completed decompression process and all image data has been transferred, the DBTF flag in JINTS1 is set to 1. When the DBTEN bit in JINTE1 is 1 here, an interrupt is generated. The interrupt source is cleared by writing 0 to the DBTF flag.
- If the count mode for stopping input coded data is on, when the specified amount of coded data set in JIFSDSC have been read, the JINF flag in JINTS1 is set to 1, and reading is stopped. When the JINEN bit in JINTE1 is 1 here, an interrupt is generated. An interrupt source is cleared by writing 0 to the JINF bit. Setting the JINRCMD bit in JIFDCNT to 1 resumes reading.

When the JINRINI bit in JIFDCNT is zero, the addresses for reading on resumption are continued from the addresses in the previous round of transfer.

When the JINRINI bit is one, the address set in JIFDSA is used on resumption.

- If the count mode for stopping the output image data is on, when the specified number of image data lines set in JIFDDL have been written, the DOUTLF flag in JINT1 is set to 1, and writing is stopped. When the DOUTLEN bit in JINTE1 is 1 here, an interrupt is generated. An interrupt source is cleared by writing 0 to the DOUTLF bit. Setting the DOUTRCMD bit in JIFDCNT to 1 resumes writing.

When the DOUTRINI bit in JIFDCNT is zero, the addresses for writing on resumption are continued from the addresses in the previous round of transfer.

When the DOUTRINI bit is one, the address set in JIFDDA is used on resumption.

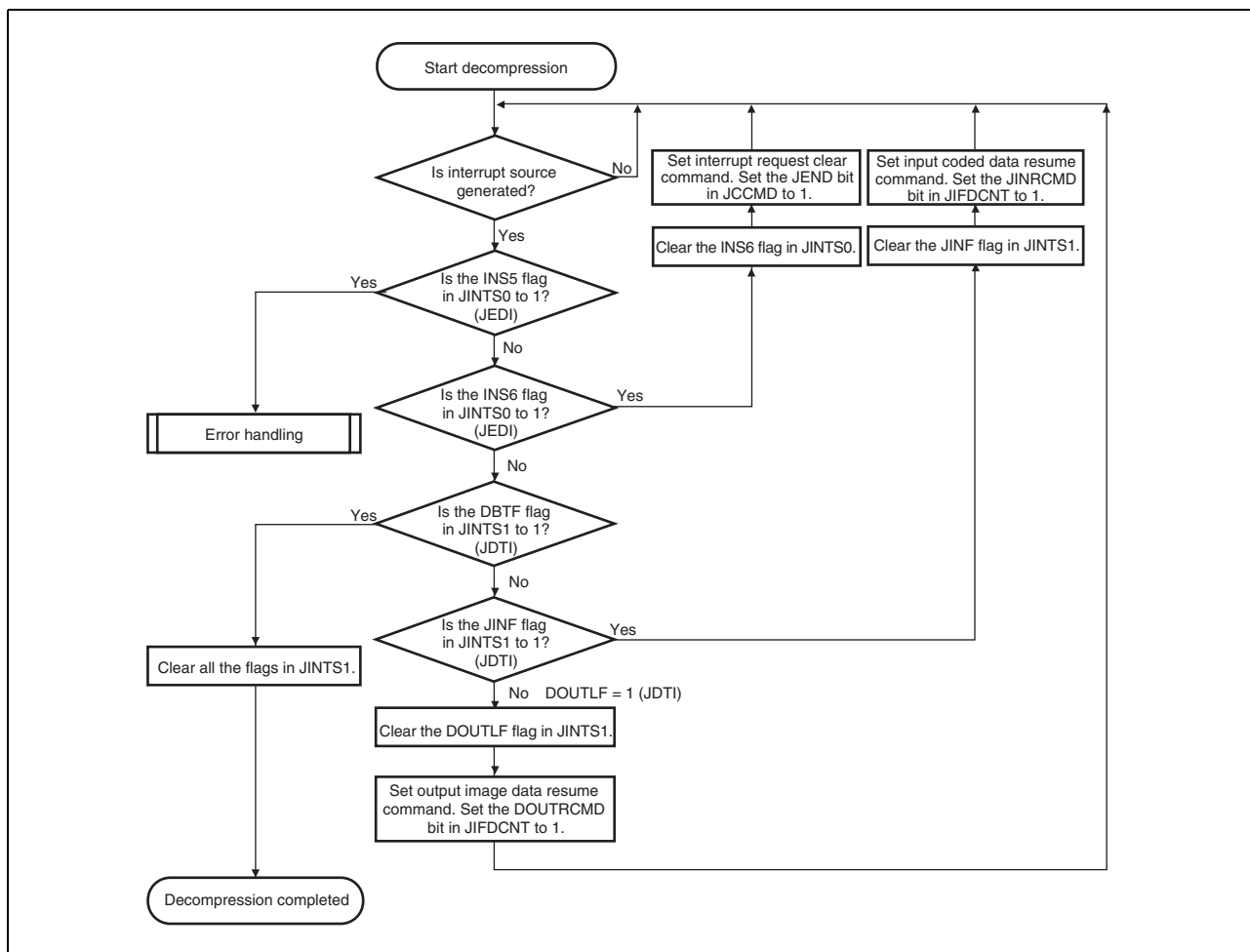
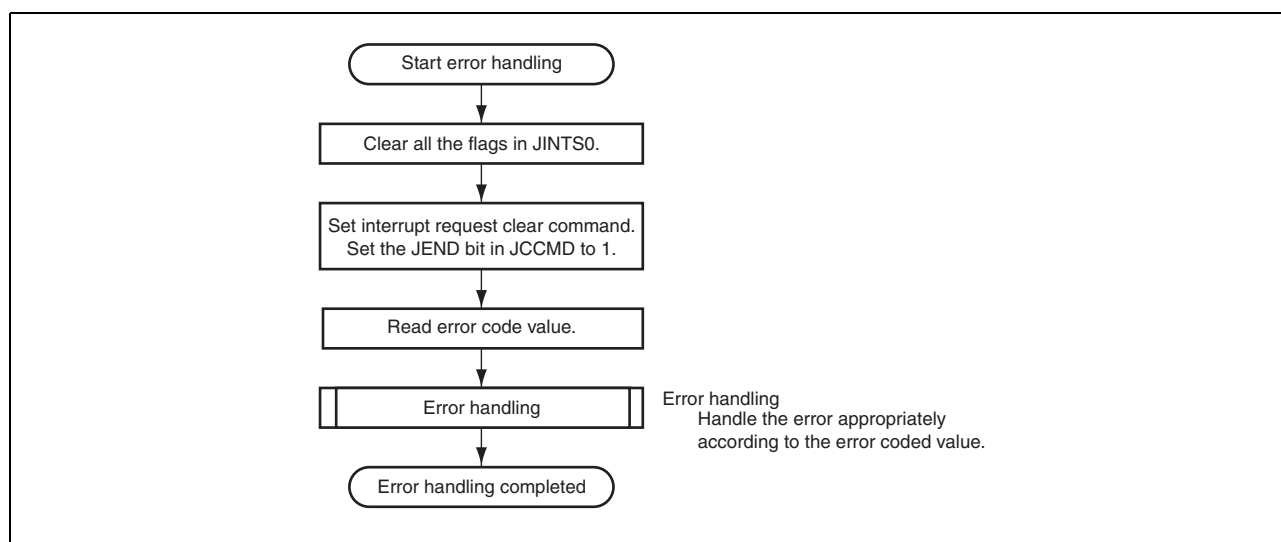


Figure 45.7 Decompression Process Flow

### (c) Error Handling

If the INS5 bit in JINTS0 is 1, it indicates that there is an error in the input JPEG coded data and that the decompression process by this module has been ended. Read the ERR bits in JCDERR to determine the cause of the error. The interrupt signal asserted due to the interrupt source indicated by the INS5 bit cannot be negated by clearing the interrupt status through 0-writing. To clear the interrupt request, set the interrupt request clear command (by setting the JEND bit in JCCMD to 1).

If decompression or compression is to proceed after error handling is completed, start by making the initial settings.



**Figure 45.8 Error Handling Flow**

### (2) Input JPEG Coded Data

Markers to be processed in decompression are SOI, SOF0, SOS, DQT, DHT, DRI, RSTm, and EOI. Other markers except for the error markers shown below are ignored even if they are read.

The JINSWAP bits in JIFDCNT can be used to alter the arrangement for the input of coded data.

### (3) JPEG Decompression Errors

#### (a) Error Marker

If a marker error is found while analyzing compressed data for decompression, the code to identify the error type (shown in Table 45.3) is set to ERR bits in JCDERR. When an error is detected, the JCU generates an interrupt signal and terminates decoding. The stored code value will be set to B'1010 (default value) at the start of processing of the next frame or after a bus reset.

**Table 45.3 Decompression Error Codes**

Code	Description
B'0000	Normal
B'0001	SOI not detected: SOI not detected until EOI detected
B'0010	SOF1 to SOFF detected
B'0011	Unprovided pixel format detected
B'0100	SOF accuracy error: Other than 8 detected
B'0101	DQT accuracy error: Other than 0 detected
B'0110	Component error 1: The number of SOF0 header components detected is other than 1, 3, or 4
B'0111	Component error 2: The number of components differs between SOF0 header and SOS
B'1000	SOF0, DQT, and DHT not detected when SOS detected
B'1001	SOS not detected: SOS not detected until EOI detected
B'1010	EOI not detected (default)
B'1011	Restart interval data number error detected
B'1100	Image size error detected
B'1101	Last MCU data number error detected
B'1110	Block data number error detected

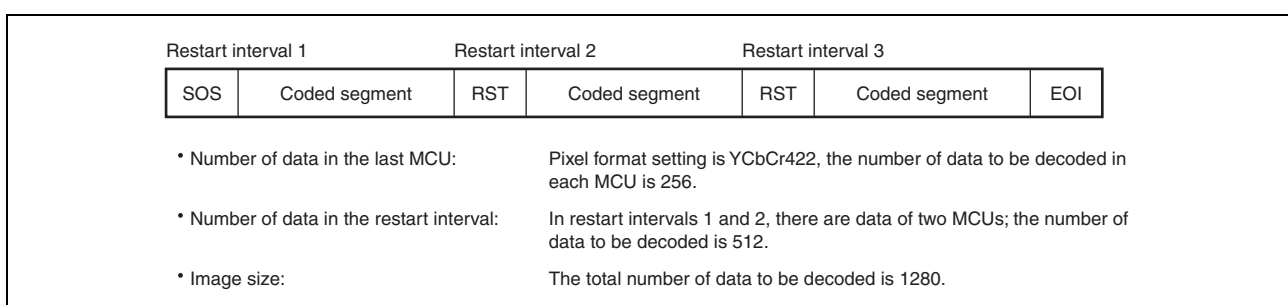
#### (b) Huffman Coded Segment Error

During the compressed data analysis in decompression operation, if there is an increase or decrease in the decoded data count due to an error resulting from bit reversal or missing data in the Huffman-coded segment, determine the error type, and set the error code in the ERR bit in JCDERR. Table 45.4 lists the segment error codes. The error code is set, interrupt signal is issued, and the process is ended only if the bits INT7 to INT5 in JINTE0 corresponding to the detected error is set to 1. The set code value will turn to the default value (B'1010) at the start of processing of the next frame or after a bus reset.

However, in this error detection, if an error in the Huffman-coded segment does not result in an alteration in the decoded data count, the error will go undetected.

[Example]

The number of data in a Huffman coded segment with pixel format setting YCbCr422, DRI = 2, X = 80 pixels, and Y = 8 pixels



**Figure 45.9 Huffman Coded Segment**

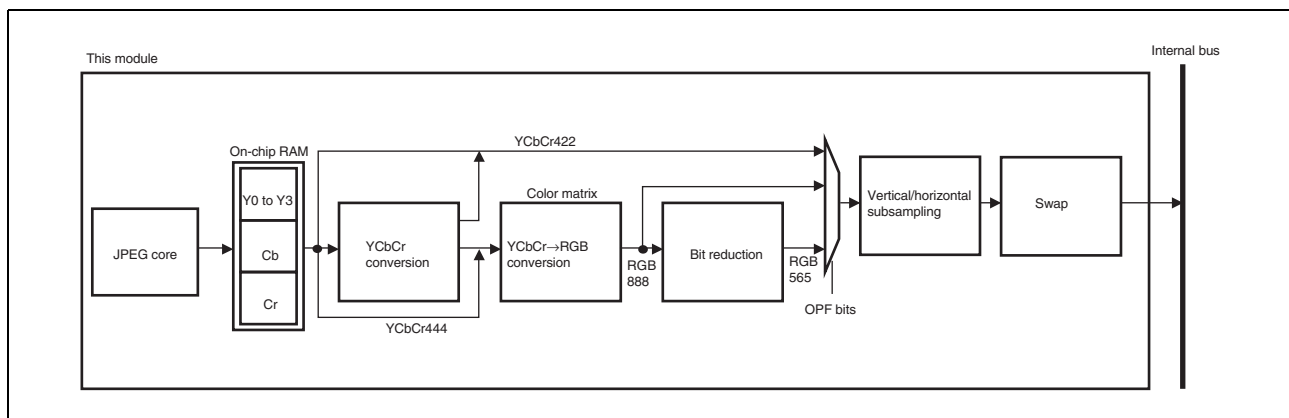


**Table 45.4 Segment Error Codes**

Code	Description
B'0000	Normal
B'1011	Restart interval data number error: The number of data in each interval is compared with the number of data specified by the DRI marker. If an interval has more or less data than is specified by the DRI marker, the decompression error code (1011) is set. The last interval which is shorter than the restart interval is not compared. If the DRI marker segment is not placed or the specified number is 00, an error is not detected even if the RST <sub>m</sub> marker is placed. Also an <i>m</i> which indicates the order of RST <sub>m</sub> marker modulo 8 ( <i>m</i> = 0 to 7) is exempt from the error detection analysis. When the INT7 bit in JINTE0 is set to 0, this error is not detected.
B'1100	Image size error: The data number of an image which is calculated from the number of lines specified by the frame parameter and the number of samples per line is compared with the total number of data from SOS to EOI (in pixel units). If the numbers of data do not match, the decompression error code (1100) is set. When the INT6 bit in JINTE0 is set to 0, this error is not detected. The data number of an image is shown in MCU units. Thus the number of lines and the number of samples per line for calculation need to be shown in MCU units.
B'1101	Last MCU data number error: Whether the number of data in the MCUs at the EOI detection is shown in MCU units is checked and fractions are detected. If error (1100) occurs simultaneously, error (1100) has priority. When the INT5 bit in JINTE0 is set to 0, this error is not detected.
B'1110	Block data number error: Whether a block is an 8 × 8 array is checked; the check is performed for fractions. When bits INT7 to INT5 in JINTE0 are all set to 0, this error is not detected.

### 45.3.3 Output Pixel Format in Decompression

This module is capable of decompressing JPEG encoded data created in the YCbCr444, YCbCr422, YCbCr411 and YCbCr420 formats. The pixel format of the output image will be YCbCr422, ARGB8888, or RGB565. The flow of conversion of decompressed data to the given output pixel format is shown below.

**Figure 45.10 Block Diagram of Output Pixel Format Conversion in Decompression**

#### (1) On-chip RAM

Data decoded by the JPEG core are stored in MCUs on RAM in this module.

#### (2) YCbCr Conversion

When data are to be output in the ARGB8888 or RGB565 format, data in the YCbCr422, YCbCr411 or YCbCr420 format are first converted to the YCbCr444 format.

When data are to be output in the YCbCr422 format, data in the YCbCr444, YCbCr411 or YCbCr420 format are converted to the YCbCr422 format.

Conversion is performed using simple interpolation.

### (3) YCbCr → RGB Conversion

Data in the YCbCr444 format are converted to the RGB888 format. The following formulae are used.

$$R = 1.000Y + 1.402Cr$$

$$G = 1.000Y - 0.344Cb - 0.714Cr$$

$$B = 1.000Y + 1.772Cb$$

### (4) Bit Reduction

RGB888 data is reduced to RGB565 data. The lower three bits of red and blue, and lower two bits of green are removed.

### (5) Output Pixel Format Selection

The pixel format to be output is selected by the OPF bit in JIFDCNT.

Allocation of data (while the DOUTSWAP bits in JIFDCNT = 000) in the pixel format is shown below.

- YCbCr422 (32 bits/pixel)

b31	b24 b23	b16 b15	b8 b7	b0
Y0 8 bits	Cb 8 bits	Y1 8 bits	Cr 8 bits	

- ARGB8888 (32 bits/pixel)

b31	b24 b23	b16 b15	b8 b7	b0
*	Red 8 bits	Green 8 bits	Blue 8 bits	

Note: \* This value is determined by the ALPHA[7:0] bits in JIFDADT.

- RGB565 (16 bits/pixel)

b15	b11 b10	b5 b4	b0
Red 5 bits	Green 6 bits	Blue 5 bits	

### (6) Vertical/Horizontal Subsampling

The output data can be horizontally and vertically subsampled according to the VINTER and HINTER bit setting in JIFDCNT.

Figure 45.11 to Figure 45.13 show line subsampling modes.

For the output formats ARGB8888 and RGB565, one cell represents one pixel in the figures.

For the output format YCbCr422, one cell represents one set of Y0Cb0Y1Cr0 in the figures.

As subsampling is carried out by minimum coded units (MCU), the numbers of the horizontal and vertical block units will vary according to the decompressed pixel format.

Table 45.5 and Table 45.6 show the values of n and m in the figures.

Horizontal:

**Table 45.5 Number of Horizontal Blocks**

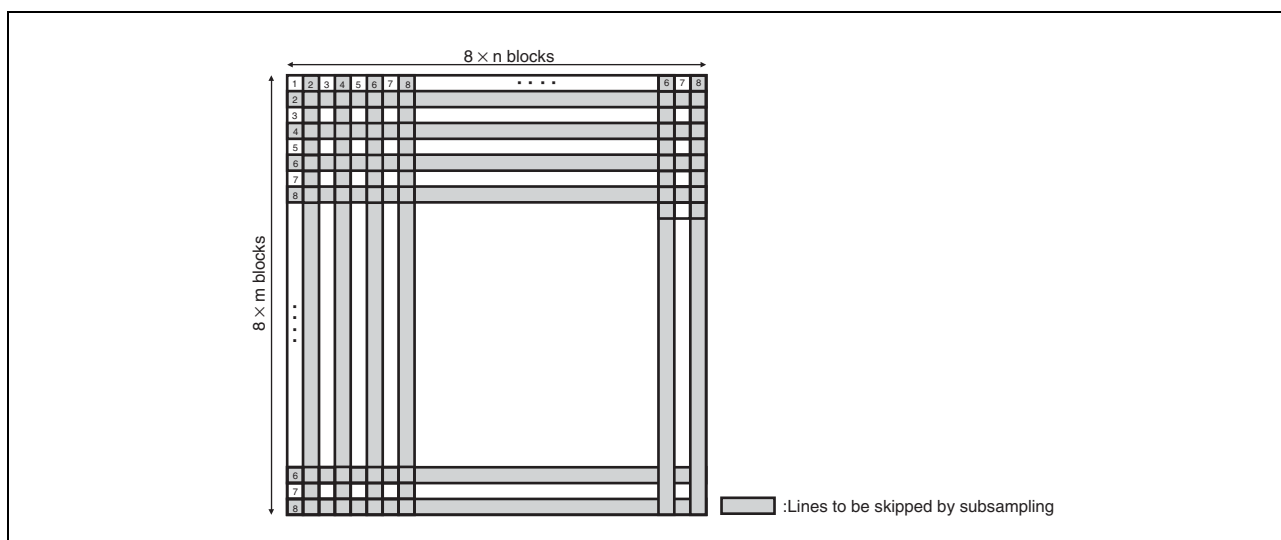
Compression Format	Output Format	n
YCbCr444	YCbCr422	1/2
YCbCr444	ARGB8888, RGB565	1
YCbCr422	YCbCr422	1
YCbCr422	ARGB8888, RGB565	2
YCbCr411	YCbCr422	2
YCbCr411	ARGB8888, RGB565	4
YCbCr420	YCbCr422	1
YCbCr420	ARGB8888, RGB565	2

Vertical:

**Table 45.6 Number of Vertical Blocks**

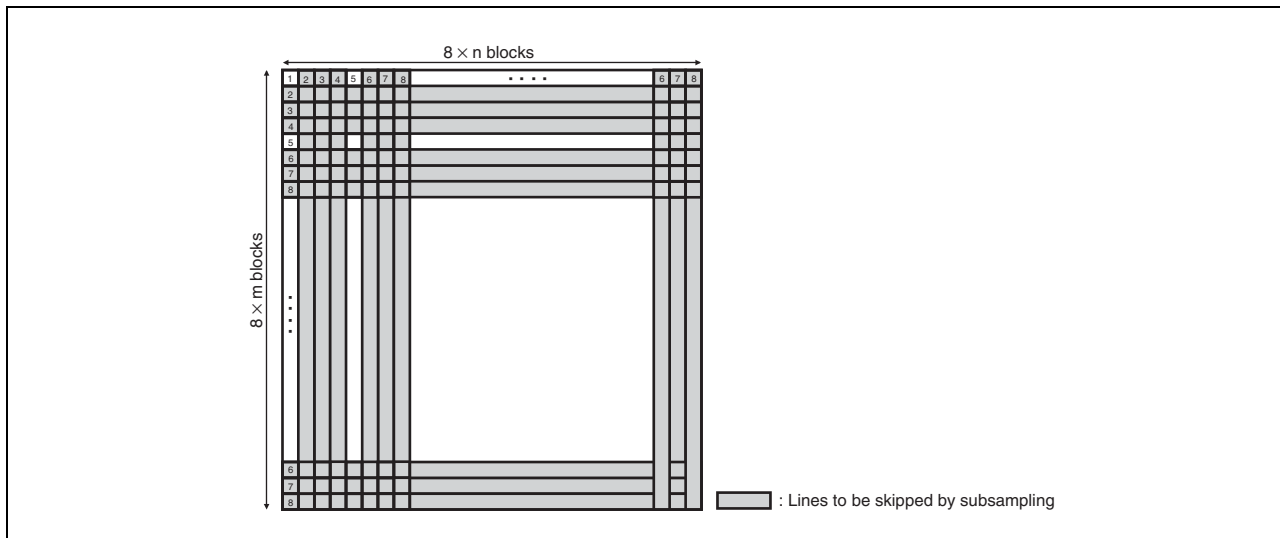
Compression Format	Output Format	m
YCbCr444	YCbCr422	1
YCbCr444	ARGB8888, RGB565	1
YCbCr422	YCbCr422	1
YCbCr422	ARGB8888, RGB565	1
YCbCr411	YCbCr422	1
YCbCr411	ARGB8888, RGB565	1
YCbCr420	YCbCr422	2
YCbCr420	ARGB8888, RGB565	2

- Subsampling into 1/2  
Even lines are skipped by subsampling.



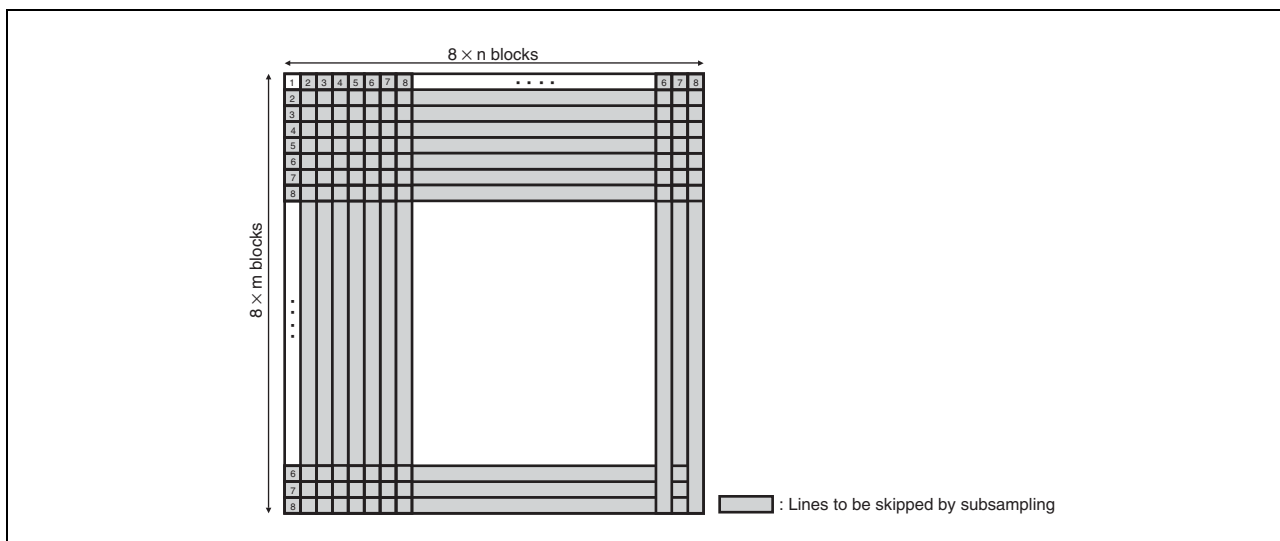
**Figure 45.11 MCU when subsampling into 1/2 is selected**

- Subsampling into 1/4  
The second, third, and fourth lines are skipped by subsampling.



**Figure 45.12 MCU when subsampling into 1/4 is selected**

- Subsampling into 1/8  
The second, third, fourth, fifth, sixth, seventh, and eighth lines are skipped by subsampling.



**Figure 45.13 MCU when subsampling into 1/8 is selected**

(7) Swap

Allocation of data can be changed by the DOUTSWAP bits in JIFECNT.

### 45.3.4 Storing Image Data

Figure 45.14 shows the buffer area for storing the image data.

- Start address  
Compression: JIFESA  
Decompression: JIFDDA
- Horizontal size  
Compression, decompression: JCHSZU, JCHSZD
- Vertical size  
Compression, decompression: JCVSAU, JCVSZD
- Offset  
Compression: JIFESOFST  
Decompression: JIFDDOFST

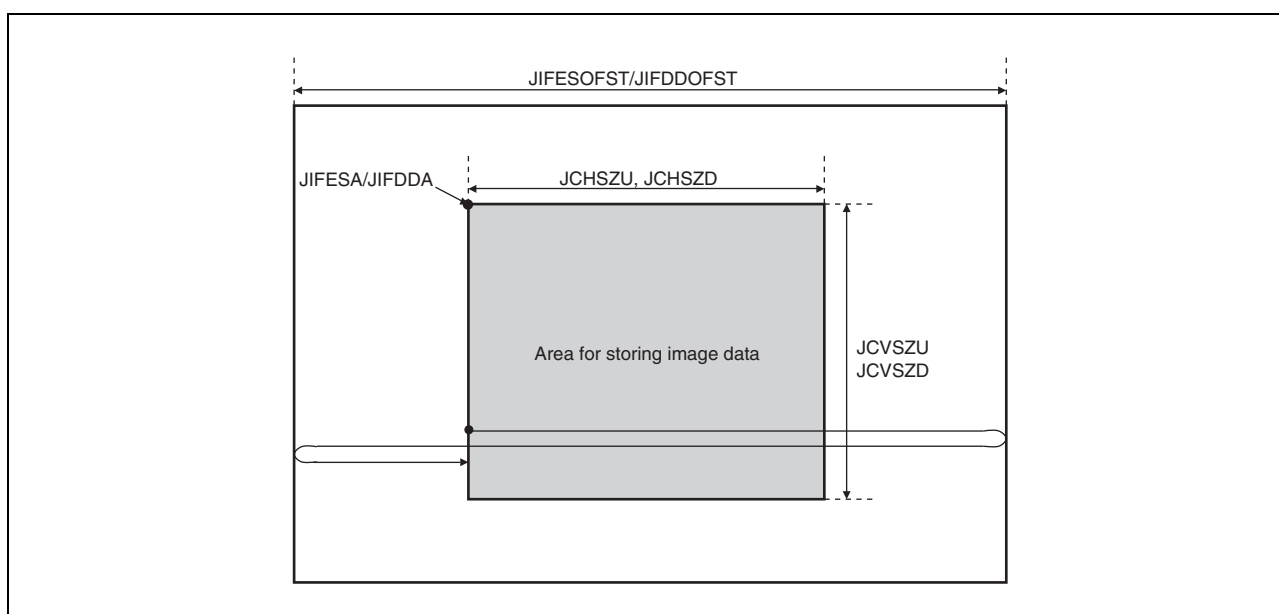


Figure 45.14 Image of Storing Image Data

## 45.4 Interrupts

Two types of interrupt requests, namely compression/decompression process interrupt request (JEDI) and data transfer interrupt request (JDTI), are available in this module. The two types of interrupt requests are each related to multiple sources. The interrupt request cancellation methods differ depending on the source of the interrupt request.

### 45.4.1 Compression/Decompression Process Interrupt Request (JEDI)

The flags in JINTS0 indicate compression/decompression-related sources. The interrupt requests asserted by these interrupt sources cannot be negated by clearing the corresponding interrupt status bits to 0. Issue an interrupt request clear command (by setting the JEND bit in JCCMD to 1) to clear the interrupt request. When a flag in JINTS0 is set to 1, a compression/decompression process interrupt request is sent to the interrupt controller.

#### (1) Compression

- JPEG compression process end

When the INS6 bit in JINTS0 is 1, the JPEG compression process has been successfully completed. After all of the coded data is transferred, the JCU completes compression.

#### (2) Decompression

- JPEG decompression process end

When the INS6 bit in JINTS0 is 1, the JPEG decompression process has been successfully completed. After all of the image data is transferred, JCU completes decompression.

- JPEG decompression error occurrence

When the INS5 bit in JINTS0 is 1, the input JPEG coded data has an error and the JCU has stopped the decompression process. Read the error code (ERR bits in JCDERR) and identify the error source. This interrupt occurs when any of the INT7 to INT5 bits in JINTE0 is 1.

- Request for reading the image size and pixel format

When the INS3 bit in JINTS0 is 1, JPEG coded data has been input and information regarding the image size and pixel format can be read. Since the JPEG decompression process is suspended, resume the JPEG decompression process by setting the process stop clear command after accessing the necessary registers. This interrupt occurs when the INT3 bit in JINTE0 is 1.

### 45.4.2 Data Transfer Interrupt Request (JDTI)

The flags in JINTS1 are the interrupt sources for transferring the image data and coded data. The interrupt requests asserted by these interrupt sources can be negated by clearing the corresponding interrupt status bits to 0.

#### (1) Compression

- Interrupt request generated after the specified number of input image data lines has been read

When the DINLF bit in JINTS1 is 1, the number of image data lines specified by JIFESLC has been transferred; transfer the rest of the image data to the external buffer and resume transferring the data from the external buffer. A data transfer interrupt request is sent when the DINLEN bit in JINTE1 is 1.

- Interrupt request generated after the specified amount of output coded data have been written to

When the JOUTF bit in JINTS1 is 1, the amount of coded data specified by JIFEDDC has been transferred. Secure a space for the next coded data in the external buffer, and resume transfer process. The data transfer interrupt request is sent when the JOUTEN bit in JINTE1 is 1.

- Interrupt request generated after all processes are completed

When the CBTF bit in JINTS1 is 1, the JCU has completed compression and transferred all of the coded data. The data transfer interrupt request is sent when the CBTEN bit in JINTE1 is set to 1.

## (2) Decompression

- Interrupt request generated after the specified number of output image data lines has been written to  
When the DOUTLF bit in the JINTS1 is 1, the number of image data lines specified by JIFDDLCL has been transferred. Secure a space for the next coded data in the external buffer, and resume transfer process. A data transfer interrupt request is sent when the DOUTLEN bit in JINTE1 is 1.
- Interrupt request generated after the specified amount of input coded data has been read  
The JINF bit in JINTS1 becomes 1 when the amount of coded data specified by JIFDSDCL has been transferred. Secure the next coded data in the external buffer, and resume transfer process. A data transfer interrupt is also sent at this time if the JINEN bit in JINTE1 is 1.
- Interrupt request generated after all processes are completed  
The DBTF bit in JINTS1 becomes 1 when the JCU has completed decompression and transferred all of the coded data. A data transfer interrupt request is also sent at this time if the DBTEN bit in JINTE1 is set to 1.

## 45.5 Bus Reset Processing

Issuing the bus reset command (setting the BRST bit in JCCMD to 1) causes a bus reset.

When the JCU is in operation, the bus reset command should not be issued. Registers below are initialized by a bus reset.

- JPEG code data count upper register (JCDTCU)
- JPEG code data count middle register (JCDTCM)
- JPEG code data count lower register (JCDTCD)
- JPEG interrupt status register 0 (JINTS0)
- JPEG code decode error register (JCDERR)
- JPEG code reset register (JCRST)



## 46. Capture Engine Unit

The capture engine unit (CEU) is a capture module that fetches image data externally input and transfers it to the memory. The CEU is connected to the system bus via bus bridge modules.

### 46.1 Features of CEU

Lists the features of CEU as follows.

#### (1) Image data fetch

- Captures an image output from an external module and writes YCbCr data to the memory with it separated into Y data and CbCr data.
- Fetches image data other than YCbCr data, e.g. JPEG data, RGB565, from an externally connected module, such as a camera, and sequentially writes the image data to the memory.
- Fetches an interlace source image in both-field units or one-field units and writes it to the memory. In both-field capture, an image can be stored in the memory as a frame image.

#### (2) Filter processing

- Performs scale-down and removal of high-frequency components (only in the horizontal direction) for an image using internal filters. Note that the scaled-down image must not exceed VGA. The filter processing can be applied to only YCbCr input data.

#### (3) Format conversion

- Converts image data input in the YCbCr422 format into the YCbCr420 format and writes it to the memory. Note that the conversion algorithm is simple skipping in which the chrominance component (CbCr) of the even-numbered lines is skipped.

## 46.2 Functional Overview of CEU

The functional overview of the CEU is shown in Table 46.1, and the main functions and their details are shown in Table 46.1.

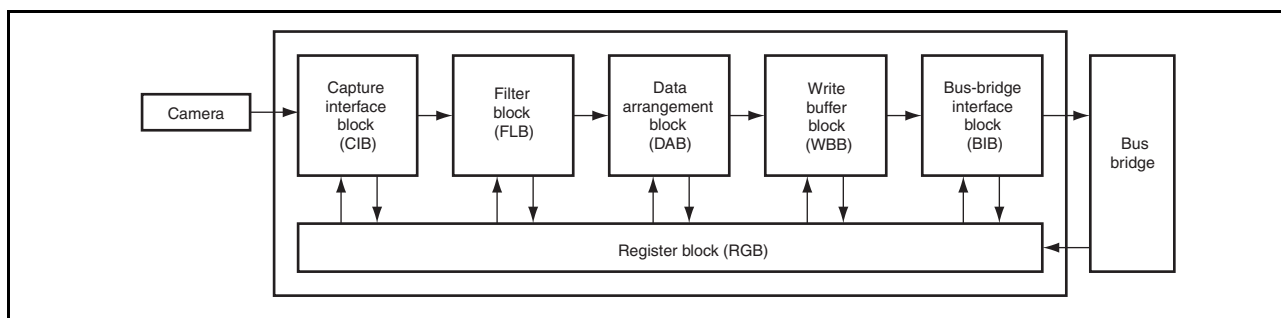
**Table 46.1 Functional Overview of CEU**

Classification	Item	Function	Description	Note	
Connectable camera	Size	WXGA	1,280 pixels × 768 lines	Horizontal: 4-pixel units Vertical: 4-line units The range of the image size that can be input is as follows. Horizontal: 1,280 pixels to 128 pixels Vertical: 768 lines to 96 lines [Note] This depends on the AC characteristics of the device to be connected.	
		XGA	1,024 pixels × 768 lines		
		SVGA	800 pixels × 600 lines		
		WVGA	800 pixels × 480 lines		
		VGA	640 pixels × 480 lines		
		CIF	352 pixels × 288 lines		
		WQVGA	480 pixels × 240 lines		
		QVGA	320 pixels × 240 lines, 240 pixels × 320 lines		
		QCIF	176 pixels × 144 lines		
		QQVGA	160 pixels × 120 lines		
	Sub-QCIF	128 pixels × 96 lines			
	Input format	YCbCr422 8 bits		Cb <sub>0</sub> , Y <sub>0</sub> , Cr <sub>0</sub> , Y <sub>1</sub> ...	Supports clock ratio of 1:1
				Cr <sub>0</sub> , Y <sub>0</sub> , Cb <sub>0</sub> , Y <sub>1</sub> ...	
			Y <sub>0</sub> , Cb <sub>0</sub> , Y <sub>1</sub> , Cr <sub>0</sub> ...		
	Y <sub>0</sub> , Cr <sub>0</sub> , Y <sub>1</sub> , Cb <sub>0</sub> ...				
	YCbCr422 16 bits		{Y <sub>0</sub> , Cb <sub>0</sub> }, {Y <sub>1</sub> , Cr <sub>0</sub> }, ... {Y <sub>0</sub> , Cr <sub>0</sub> }, {Y <sub>1</sub> , Cb <sub>0</sub> }, ...		
	Binary data		Specified amount to be fetched on edges of the sync signal Data is fetched with the horizontal sync signal as an enable signal.	Written sequentially	
Connectable camera	Horizontal and vertical sync signal polarities	Arbitrary	High-active and low-active		
	Capture start location	Arbitrary	Can be specified in camera input clock units	Horizontal: 1-cycle units Vertical: 1-HD (horizontal sync signal) units	
	Number of captured pixels	Arbitrary	Can be specified in 4-pixel units horizontally and in 4-line units vertically		
	Interlace	Both-field capture		Stored as a field image Stored as a frame image	Capture: 2-VD (vertical sync signal) units
One-field capture			Top field or bottom field can be specified	Capture: 1-VD units	
Memory write	Output format	YCbCr422 YCbCr420	YCbCr420 is realized by simple skipping		
Filter function	No scaling or scale-down	Scale-down of captured display	Desired scaling factor from 1/16 to 1 (scaled-down display must not exceed VGA)		
	Low-pass filter		Removal of high-frequency components	Only in the horizontal direction	

**Table 46.2 Main Functions of CEU and Their Details**

Main Function	Detailed Description
Image data fetch	<ul style="list-style-type: none"> <li>• Captures an image output from an external module and writes YCbCr data to the memory with it separated into Y data and CbCr data.</li> <li>• Fetches image data other than YCbCr data, e.g. JPEG data, from an externally connected module, such as a camera, and sequentially writes the image data to the memory.</li> <li>• Fetches an interlace source image in both-field units or one-field units and writes it to the memory. In both-field capture, an image can be stored in the memory as a frame image.</li> </ul>
Filter processing	<p>Performs scale-down and removal of high-frequency components (only in the horizontal direction) for an image using internal filters.</p> <p>Note that the scaled-down image must not exceed VGA. The filter processing can be applied to only YCbCr input data.</p>
Format conversion	<p>Converts image data input in the YCbCr422 format into the YCbCr420 format and writes it to the memory.</p> <p>Note that the conversion algorithm is simple skipping in which the chrominance component (CbCr) of the even-numbered lines is skipped.</p>

Figure 46.1 shows a block diagram of the CEU.

**Figure 46.1 Block Diagram of CEU**

### 46.3 Pin Configuration of CEU

The pin configuration of the CEU is shown in Table 46.3.

**Table 46.3 Pin Configuration of CEU**

Pin Name	Function	I/O	Description
VIO_D15 to VIO_D0*	CEU data bus	Input	Camera image data input to the CEU
VIO_CLK	CEU clock	Input	Camera clock input to the CEU
VIO_VD	CEU vertical sync	Input	Camera vertical sync signal input to the CEU
VIO_HD	CEU horizontal sync	Input	Camera horizontal sync signal input to the CEU
VIO_FLD	Field signal	Input	Field identification signal to the CEU

Note: \* When the distinction according to the bus width for the data bus is not needed, VIO\_D is used in this manual. Otherwise, VIO\_D15 to VIO\_D0 are used.

## 46.4 Register Descriptions of CEU

The register configuration of the CEU is shown in Table 46.4.

Most CEU registers have a 2-plane configuration (plane A and plane B). The CEU switches the planes when using these 2-plane registers. A mirror address, which is an address that can always access the register on the unused plane, is provided for each 2-plane register. Figure 46.2 shows the timing to switch the register planes. The CEU switches the register planes at the same time a VD interrupt is asserted.

In the following register descriptions, "during operation" indicates the period that begins when the CEU is activated by the CE bit in the capture start register (CAPSR) and ends when a capture end interrupt (CPE) of the capture event flag clear register (CETCR) occurs. In the read-only bits in each register, the write value should always be 0. If a value other than 0 is written to any of these bits, correct operation cannot be guaranteed.

**Table 46.4 Register Configuration of CEU**

Register Name	Abbr.	R/W	Addresses			Access Size
			Address (Plane A)	Address (Plane B)	Mirror Address	
CEU Capture start register	CAPSR	R/W	H'E821 0000	—	—	32
CEU Capture control register	CAPCR	R/W	H'E821 0004	—	—	32
CEU Capture interface control register*	CAMCR	R/W	H'E821 0008	—	—	32
CEU Capture interface cycle register*	CMCYR	R/W	H'E821 000C	—	—	32
CEU Capture interface offset register	CAMOR	R/W	H'E821 0010	H'E821 1010	H'E821 2010	32
CEU Capture interface width register	CAPWR	R/W	H'E821 0014	H'E821 1014	H'E821 2014	32
CEU Capture interface input format register	CAIFR	R/W	H'E821 0018	—	—	32
CEU register control register	CRCNTR	R/W	H'E821 0028	—	—	32
CEU register forcible control register	CRCMPR	R/W	H'E821 002C	—	—	32
CEU Capture filter control register	CFLCR	R/W	H'E821 0030	H'E821 1030	H'E821 2030	32
CEU Capture filter size clip register	CFSZR	R/W	H'E821 0034	H'E821 1034	H'E821 2034	32
CEU Capture destination width register	CDWDR	R/W	H'E821 0038	H'E821 1038	H'E821 2038	32
CEU Capture data address Y register	CDAYR	R/W	H'E821 003C	H'E821 103C	H'E821 203C	32
CEU Capture data address C register	CDACR	R/W	H'E821 0040	H'E821 1040	H'E821 2040	32
CEU Capture data bottom-field address Y register	CDBYR	R/W	H'E821 0044	H'E821 1044	H'E821 2044	32
CEU Capture data bottom-field address C register	CDBCR	R/W	H'E821 0048	H'E821 1048	H'E821 2048	32
CEU Capture bundle destination size register	CBDSR	R/W	H'E821 004C	H'E821 104C	H'E821 204C	32
CEU Firewall operation control register	CFWCR	R/W	H'E821 005C	—	—	32
CEU Capture low-pass filter control register	CLFCR	R/W	H'E821 0060	H'E821 1060	H'E821 2060	32
CEU Capture data output control register	CDOCR	R/W	H'E821 0064	H'E821 1064	H'E821 2064	32
CEU Capture event interrupt enable register	CEIER	R/W	H'E821 0070	—	—	32
CEU Capture event flag clear register	CETCR	R/W	H'E821 0074	—	—	32
CEU Capture status register	CSTSR	R	H'E821 007C	—	—	32
CEU Capture data size register	CDSSR	R/W	H'E821 0084	—	—	32
CEU Capture data address Y register 2	CDAYR2	R/W	H'E821 0090	H'E821 1090	H'E821 2090	32
CEU Capture data address C register 2	CDACR2	R/W	H'E821 0094	H'E821 1094	H'E821 2094	32
CEU Capture data bottom-field address Y register 2	CDBYR2	R/W	H'E821 0098	H'E821 1098	H'E821 2098	32
CEU Capture data bottom-field address C register 2	CDBCR2	R/W	H'E821 009C	H'E821 109C	H'E821 209C	32

Note: \* After changing the setting of a register (CAMCR or CMCYR) that is determined by the external module characteristics, do not start capture until at least 10 external input clock cycles have elapsed.

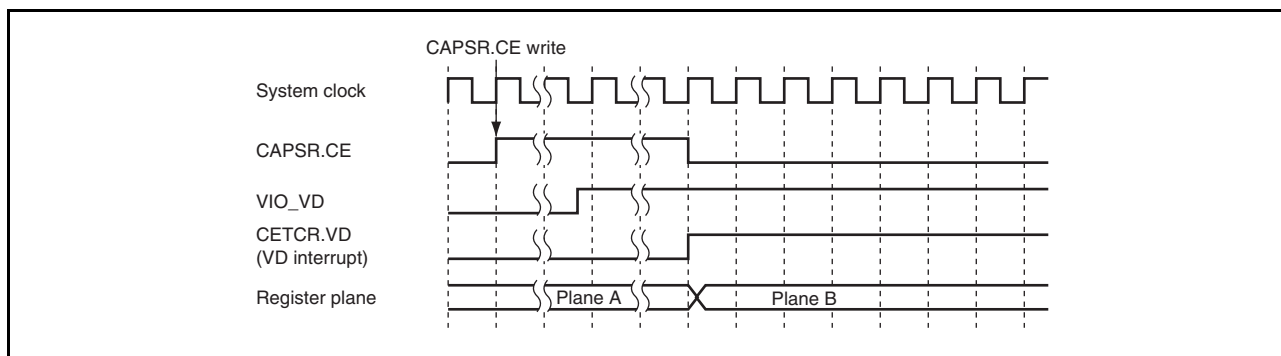


Figure 46.2 Register Plane Switching Timing (VD Polarity is High-Active)

### 46.4.1 Capture Start Register (CAPSR)

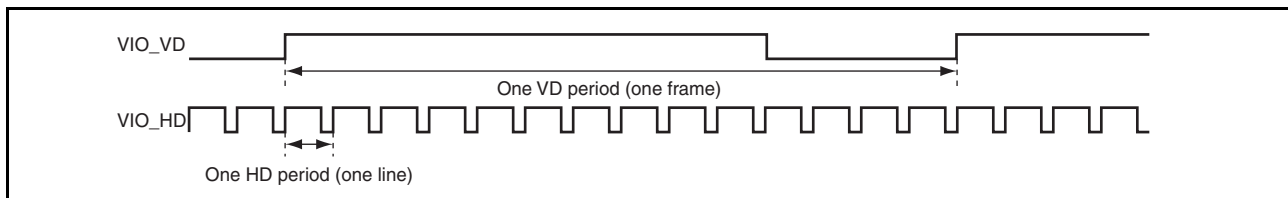
CAPSR captures data input to the CEU from an external module.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CPKIL
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	CPKIL	0	R/W	Write 1 to this bit to perform a software reset of capturing. At a software reset, capturing ends immediately without completing capture operation until the end of a frame. Clear the CE bit to 0 when writing 1 to this bit. Processing of the capture software reset is indicated by this bit being set to 1. When this bit is 1, do not start capturing since reset processing is in progress. When restarting capture operations, after referring to the CPTON bit in CSTSR to ensure that the CEU is halted (in the idle state), wait until this bit is cleared to 0. The timing of restarting capture operations is shown in Figure 46.6. When a software reset is generated by this bit, a capture end interrupt (CPE bit in CETCR) may be output immediately after the software reset. However, such kind of interrupt should be ignored. Also, even if the capture end interrupt is not output, the interrupt source (CPE bit) must be cleared before capturing of the next frame. 0: Normal state 1: Software reset of capturing
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
0	CE	0	R/W	<ul style="list-style-type: none"> <li>• Single-capture           <p>This bit reserves capturing of the next frame. When 1 is written to this bit, the capture of one frame starts from the next VD input, and stops when the one-frame capture end interrupt (CPE bit in CETCR) is asserted (Figure 46.7). To perform capture again, write 1 to this bit. After the VD or HD polarity is changed, do not write 1 to this bit until the next VD interrupt is asserted.</p> <p>As this bit indicates the capture reserve state, this bit is read as 1 after it is set to 1 and until VD is input. When VD is input, this bit returns to 0 and so is read as 0.</p> <p>The capture end is determined by the one-frame capture end interrupt (CPE bit). This is similar in data fetch mode.</p> <p>Registers should be set before the VD interrupt of the frame where capture starts next. The new register settings take effect at the next VD input. When registers are modified during capturing, the register settings take effect from the capture operations of the next VD input. If a setting register to which writing during capturing is prohibited is modified during capturing, an interrupt source (IGRW bit in CETCR) is generated. For details on the interrupt source, see the description on CETCR.</p> </li> <li>• Continuous capture           <p>When this bit is set to 1 while the CTNCP bit in CAPCR is set to 1, continuous capture starts from the next frame (Figure 46.8). Note that this bit is not cleared to 0 but remains as 1. To stop capturing, clear this bit to 0; capturing stops after the current frame is completed.</p> <p>Continuous capture operations are possible in only image capture mode. The start address of the memory to which the captured data is written to must be set for each frame.</p> <p>0: Stops capturing 1: Starts capturing</p> </li> </ul>

When both the VD (vertical sync signal) and HD (horizontal sync signal) polarities are high-active, one frame is defined as a period from a VD rising edge to the next VD rising edge, and one line as a period from an HD rising edge to the next HD rising edge. Figure 46.3 shows the timing of one frame (when both the VD and HD polarities are high-active).

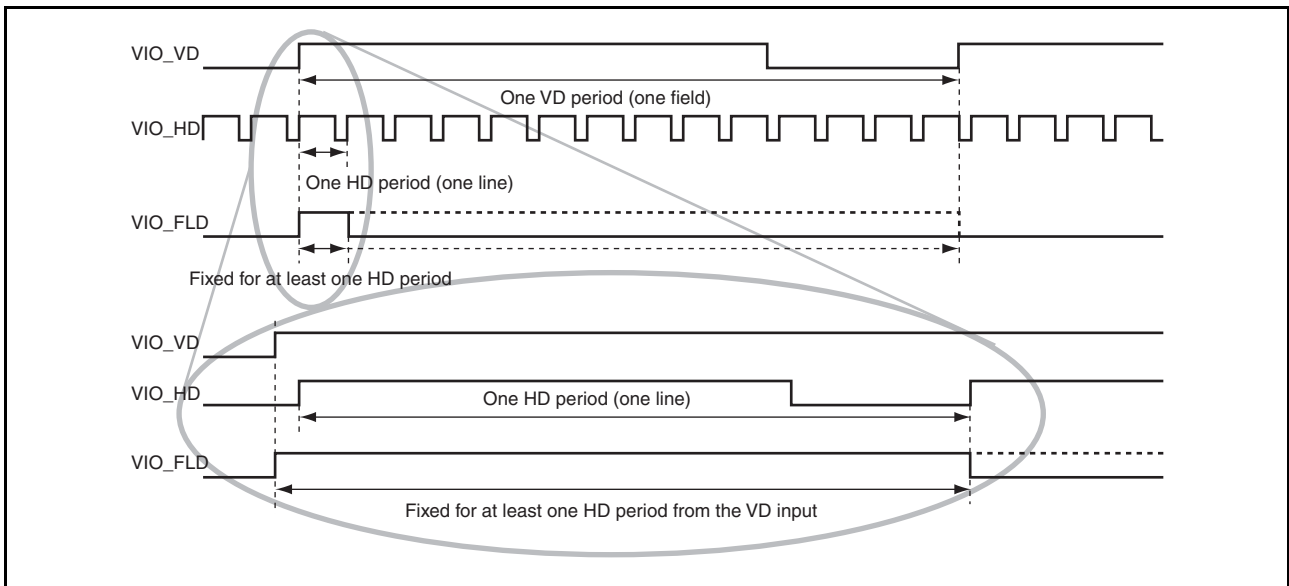


**Figure 46.3** Frame Timing

When both the VD and HD polarities are high-active, similar to one frame, one field is defined as follows:

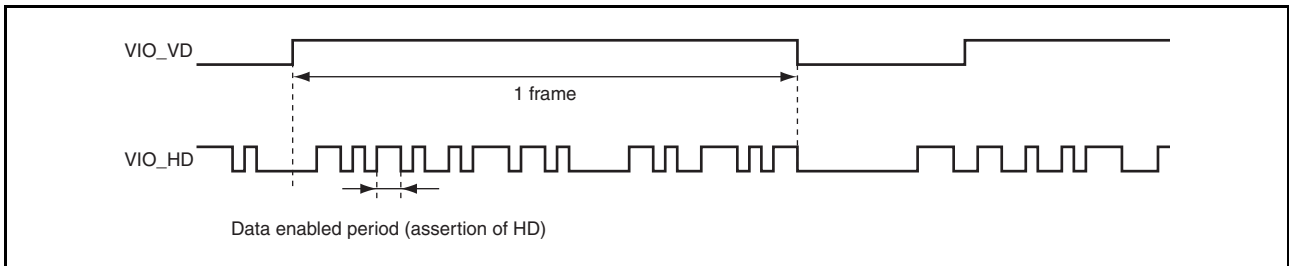
- Period from a VD rising edge to the next VD rising edge
- One line is a period from an HD rising edge to the next HD rising edge

The field identification signal FLD should be fixed for at least 1-HD period from a VD input. Figure 46.4 shows the timing of one field (when both the VD and HD polarities are high-active).

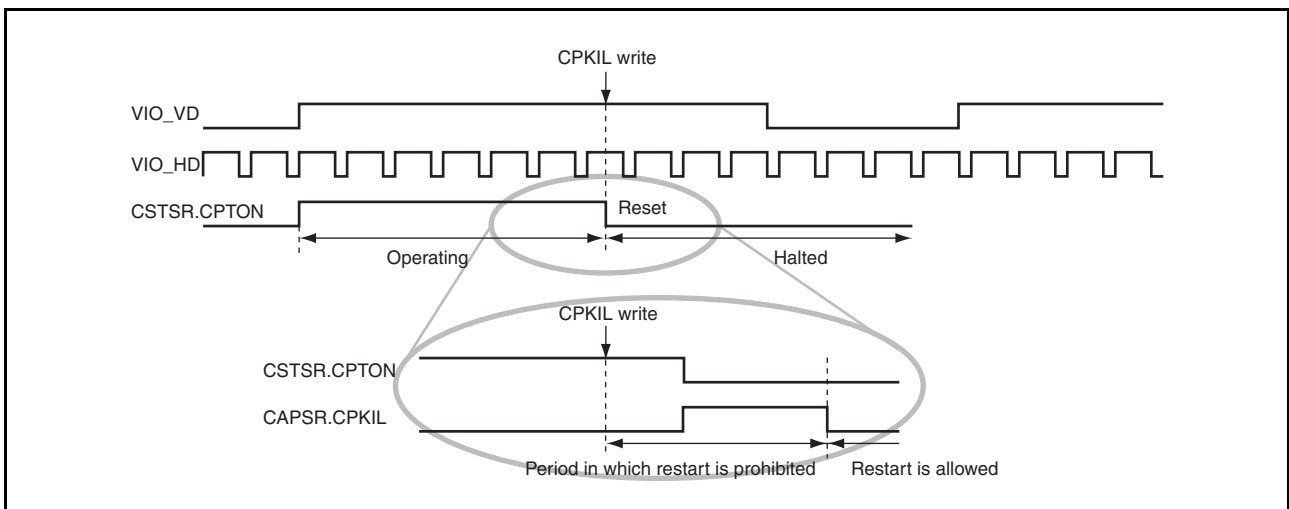


**Figure 46.4 One Field Timing**

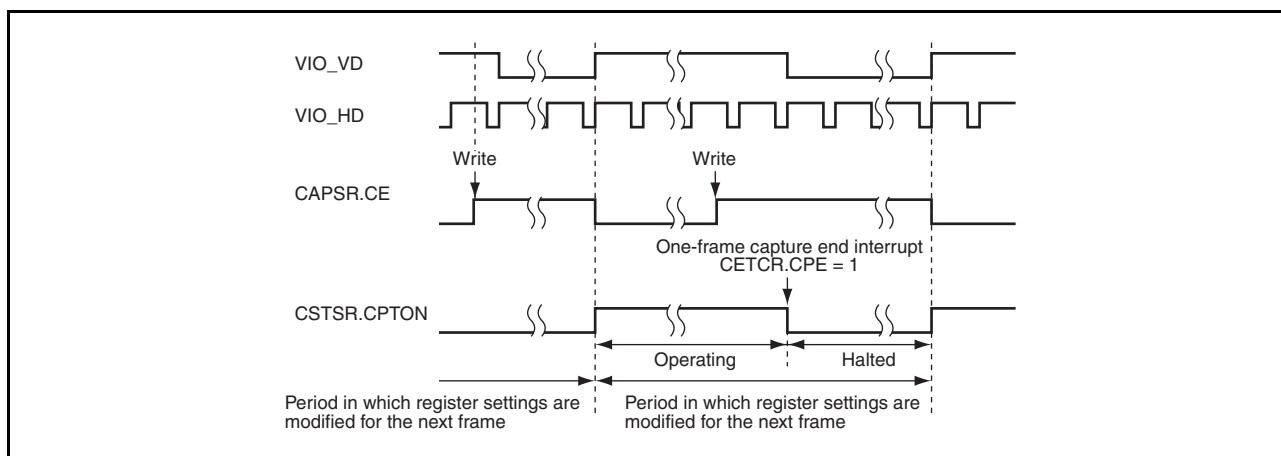
In data enable fetch, one frame is defined as a period from a VD rising edge to the VD falling edge. With the HD as an enable signal (positive polarity), data of a cycle in which the HD is asserted is fetched while the VD is high. Figure 46.5 shows the timing of one frame for data enable fetch.



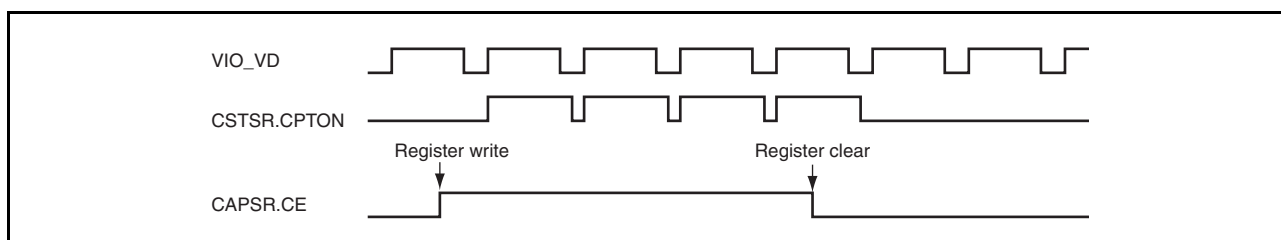
**Figure 46.5 Frame Timing (Data Enable Fetch)**



**Figure 46.6 Timing of Software Reset and Restart of Capturing**



**Figure 46.7** Timing of Modifying CE Bit and Register Setting in One Frame Capture



**Figure 46.8** Continuous-Frame Capture



## 46.4.2 Capture Control Register (CAPCR)

CAPCR sets continuous-frame capture and the frame drop intervals.

Do not modify this register during operation. If this register is modified during operation, correct operation cannot be guaranteed. In addition, the IGRW bit (interrupt source) in CETCR is set to B'1.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	FDRP[7:0]							—	—	MTCM[1:0]	—	—	—	CTNCP		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	FDRP[7:0]	H'00	R/W	<p>These bits set the frame drop interval in continuous-frame capture. When these bits are cleared to 0, frame drop is not performed, and all frames are captured. Figure 46.9 shows the value set in these bits and the timing of captured frames.</p> <p>The frame drop interval unit differs according to the capture setting. Table 46.5 shows the relationship between the capture setting and frame drop interval unit. The image of the frame drop timing for each capture setting when these bits are set to 2 is shown in Figure 46.10.</p> <p>In both-field capture, capturing is performed continuously for 2-VD periods, regardless of whether the second field is the top field or bottom field. In addition, in both-field capture, the frame drop counter is incremented when the first field has been identified as the top field or bottom field, regardless of whether the second field is the top field or bottom field.</p> <p>When 0 is written to the CE bit in CAPSR, capturing terminates after the current frame has been captured for a capture frame. However, for a drop frame, capturing is forcibly terminated in the CEU so no capture end interrupt (CPE bit in CETCR) is output.</p> <p>While CE bit is 1, do not change the setting of these bits.</p> <p>Note: Do not change the setting of these bits during continuous capture operations. To change the setting of these bits, stop continuous capture (CE bit = 0), clear the CTNCP bit (continuous capture) in CAPCR to 0, and then restart continuous capture. Continuous capture is performed during the period of CAPSR.CE = 1 shown in Figure 46.9.</p>
23, 22	—	All 0	R	<p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p>

Bit	Bit Name	Initial Value	R/W	Description
21, 20	MTCM[1:0]	00	R/W	<p>These bits specify the unit for transferring data to a bus bridge module. The efficiency of writing image data can be improved by continuously accessing the addresses. To improve the write efficiency, set these bits to 11. The setting of these bits appear to be unchanged from the outside.</p> <p>00: Transferred to the bus in 32-byte units                      01: Transferred to the bus in 64-byte units                      10: Transferred to the bus in 128-byte units                      11: Transferred to the bus in 256-byte units</p> <p>(1) Image capture                      00: Y data and C data are transferred in 32-byte units                      01: Y data and C data are transferred in 64-byte units                      10: Y data and C data are transferred in 128-byte units                      11: Y data and C data are transferred in 256-byte units</p> <p>(2) Data fetch                      00: Data is transferred in 32-byte units                      01: Data is transferred in 64-byte units                      10: Data is transferred in 128-byte units                      11: Data is transferred in 256-byte units</p>
19 to 17	—	All 0	R	<p>Reserved                      These bits are always read as 0. The write value should always be 0.</p>
16	CTNCP	0	R/W	<p>When capturing is started with this bit set to 1, capturing continues until the CE bit in CAPSR is cleared to 0 or a software reset is initiated by the CPKIL bit in CAPSR (see Figure 46.8). Continuous capture must be set before capturing is started.</p> <p>This bit is modified only after 0 is written to the CE bit to stop capturing. If this bit is modified during capturing, correct operation cannot be guaranteed.</p> <p>In data fetch mode, clear this bit to 0.                      0: One-frame capture when the CE bit is 1                      1: Continuous capture until the CE bit is cleared to 0</p>
15 to 0	—	All 0	R	<p>Reserved                      These bits are always read as 0. The write value should always be 0.</p>

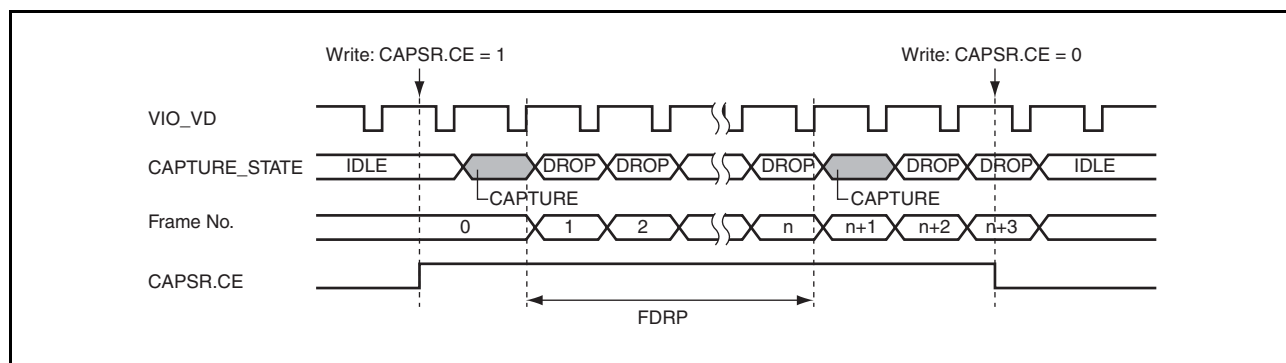


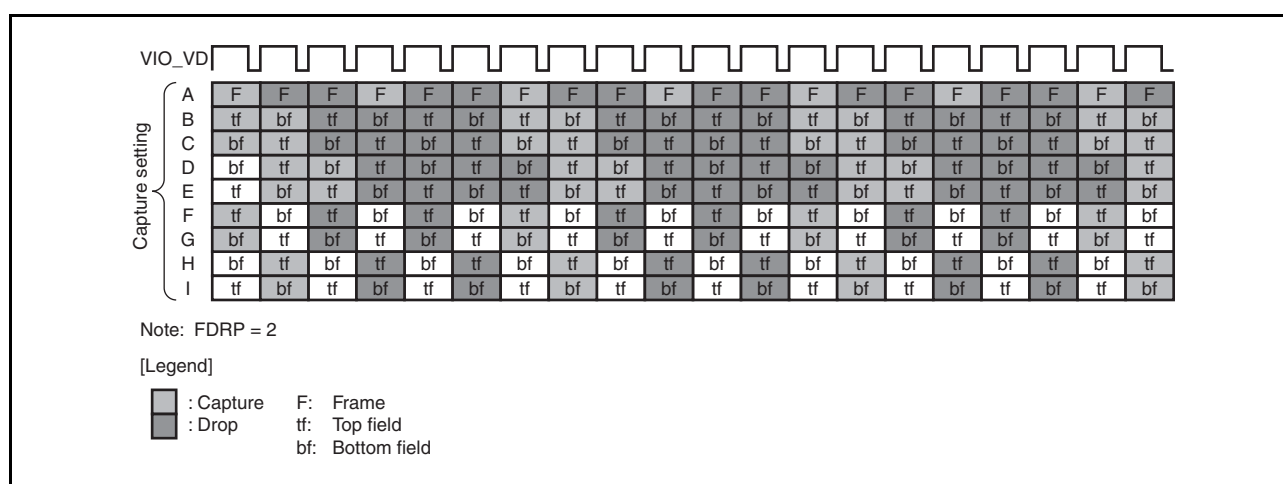
Figure 46.9 Setting of FDRP Bits and Frame Drop Timing

Table 46.5 Relationship between Capture Setting and Frame Drop Interval Unit

Input Mode	Captured Image	First Captured Image	Frame Drop Interval Unit	Capture Setting
Progressive	Frame	Frame immediately after capture start	Frame	A

**Table 46.5 Relationship between Capture Setting and Frame Drop Interval Unit**

Input Mode	Captured Image	First Captured Image	Frame Drop Interval Unit	Capture Setting
Interlace	Both-field (2-VD capture)	Field immediately after capture start	2 fields (first capture field count)	B
		Top field	2 fields (top-field count)	D
		Bottom field	2 fields (bottom-field count)	E
	One-field (1-VD capture)	Field immediately after capture start	First capture field	F
		Top field	Top field	H
		Bottom field	Bottom field	I



**Figure 46.10 Image of Frame Drop Timing for Capture Settings (FDRP Bits = 2)**

### 46.4.3 Capture Interface Control Register (CAMCR)

CAMCR sets the capture interface.

The following items are set by CAMCR.

- Selection between image capture operation or data fetch operation
- Polarities of the vertical and horizontal sync signals
- Input order of image data components (Y, Cb, and Cr) (only for image capture mode)
- Selection of digital image input pins (8 bits or 16 bits)
- Polarity of the field identification signal

CAMCR must be set according to the module connected. In data fetch mode, set the DTARY bits to B'0. Do not modify this register during operation. If this register is modified during operation, correct operation cannot be guaranteed. In addition, the IGRW bit (interrupt source) in CETCR is set to B'1.

Note: After changing the setting of this register, do not start capture until at least 10 external input clock cycles have elapsed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	FLDPOL
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	DTIF	—	—	DTARY[1:0]	—	—	JPG[1:0]	—	—	—	—	VDPOL	HDPOL
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	FLDPOL	0	R/W	Sets the polarity of the field identification signal (FLD) from an external module. 0: When the FLD signal is high-active, the field is detected as the top field and when low-active, the field is detected as the bottom field. 1: When the FLD signal is low-active, the field is detected as the top field and when high-active, the field is detected as the bottom field.
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	DTIF	0	R/W	Sets the digital image input pins from which data is to be captured. 0: Data input to 8-bit digital image input pins is captured 1: Data input to 16-bit digital image input pins is captured
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
9, 8	DTARY[1:0]	00	R/W	<p>These bits set the input order of the luminance component and chrominance component.</p> <p>The order in which the luminance component (Y) and chrominance component (Cb and Cr) are input from an external module differs among modules. The CEU supports the input orders shown in Figure 46.12. Set the corresponding value in these bits. In data fetch mode, set these bits to 00.</p> <p>(1) 8-bit interface            00: Image input data is fetched in the order of Cb<sub>0</sub>, Y<sub>0</sub>, Cr<sub>0</sub>, and Y<sub>1</sub>            01: Image input data is fetched in the order of Cr<sub>0</sub>, Y<sub>0</sub>, Cb<sub>0</sub>, and Y<sub>1</sub>            10: Image input data is fetched in the order of Y<sub>0</sub>, Cb<sub>0</sub>, Y<sub>1</sub>, and Cr<sub>0</sub>            11: Image input data is fetched in the order of Y<sub>0</sub>, Cr<sub>0</sub>, Y<sub>1</sub>, and Cb<sub>0</sub></p> <p>(2) 16-bit interface            00: Image input data is fetched in the order of {Cb0, Y0} and {Cr0, Y1}            01: Image input data is fetched in the order of {Cr0, Y0} and {Cb0, Y1}            10: Image input data is fetched in the order of {Y0, Cb0} and {Y1, Cr0}            11: Image input data is fetched in the order of {Y0, Cr0} and {Y1, Cb0}</p>
7, 6	—	All 0	R	<p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p>
5, 4	JPG[1:0]	00	R/W	<p>These bits select the fetched data type.</p> <p>00: Image capture mode            (input data are separated into Y data and CbCr data for output to the memory)</p> <p>01: Data synchronous fetch mode            (specified size of input data are output to the specified memory addresses in order of input and in synchronization with the sync signal)</p> <p>10: Data enable fetch mode            (input data are fetched with HD as an enable signal and output to the specified addresses in memory in order of input)</p>
3, 2	—	All 0	R	<p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p>
1	VDPOL	0	R/W	<p>Sets the polarity for detection of the vertical sync signal input from an external module.</p> <p>Figure 46.14 shows the relationship between the VD and VD interrupt when high-active is selected.</p> <p>Since a VD interrupt may occur when this bit value is modified, the VD bit in CETCR must always be cleared to 0 when this bit value is changed. Note that this bit is not used in data enable fetch mode.</p> <p>0: Vertical sync signal (VD) from an external module is detected as high-active            1: Vertical sync signal (VD) from an external module is detected as low-active</p>
0	HDPOL	0	R/W	<p>Sets the polarity for detection of the horizontal sync signal input from an external module.</p> <p>Figure 46.15 shows the relationship between the HD and HD interrupt when high-active is selected.</p> <p>Since an HD interrupt may occur when this bit value is modified, the HD bit in CETCR must always be cleared to 0 when this bit value is changed.</p> <p>0: Horizontal sync signal (HD) from an external module is detected as high-active            1: Horizontal sync signal (HD) from an external module is detected as low-active</p>

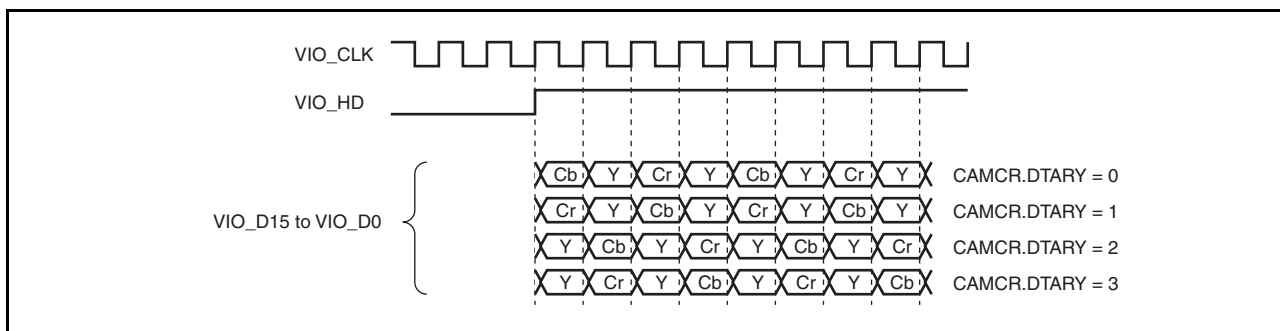


Figure 46.11 Input Order of Image Data

The JPG bit in CAMCR selects whether digital image data is fetched or data such as JPEG is fetched. In addition, when data such as JPEG is fetched, select whether the specified amount of data is continuously fetched in synchronization with the sync signal or data is fetched while the horizontal sync signal is enabled.

In data enable fetch mode, one frame is defined as a period from the rising edge to the falling edge of the vertical sync signal (VD) for data fetching. The horizontal sync signal (HD) is enabled only when the VD is high and treated as an enable signal. Data input in the cycle in which the HD is asserted (high) is fetched and output to the memory continuously.

Figure 46.12 and Figure 46.13 show the interface timing in data enable fetch mode.

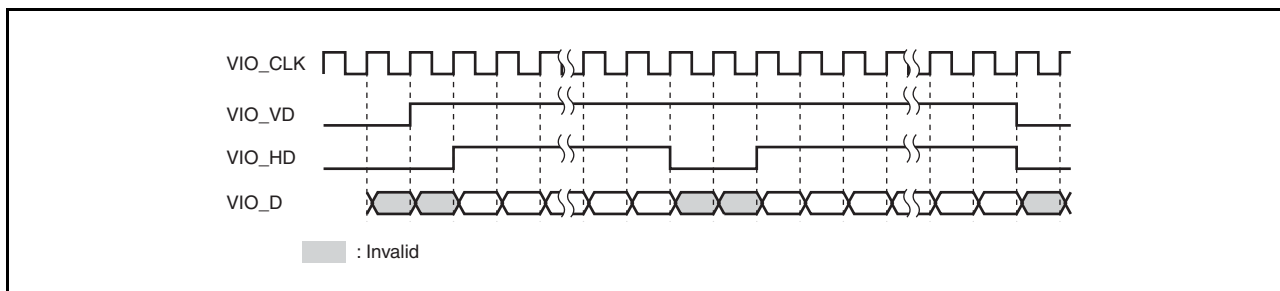


Figure 46.12 Data Enable Fetch Timing (HD Asserted (High) While VD is High)

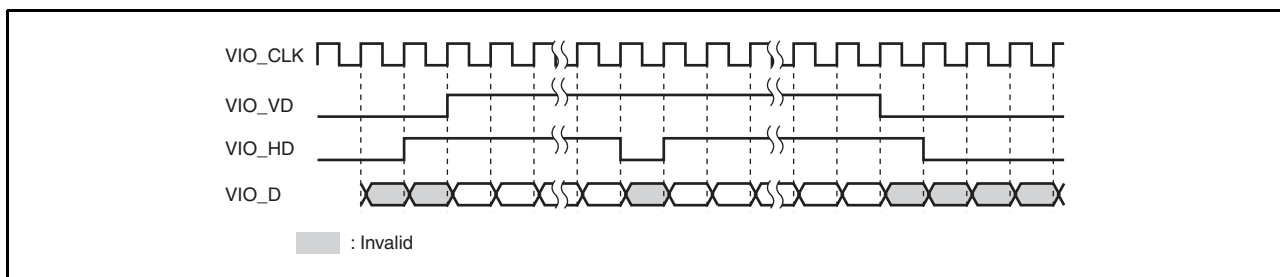
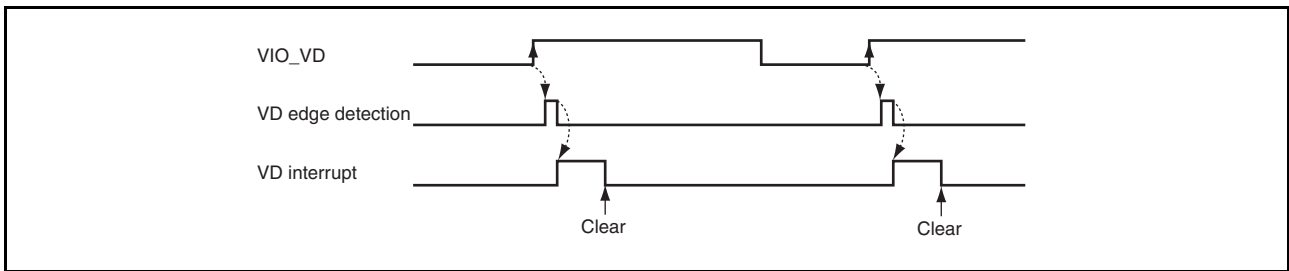
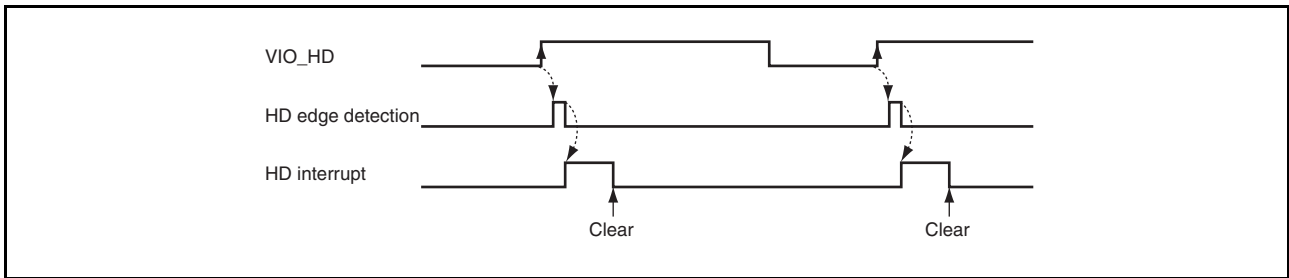


Figure 46.13 Data Enable Fetch Timing (HD Asserted (High) When VD is Not High)

This module starts fetching data at the rising edge of the VD and stops fetching data at the falling edge of the VD in data enable fetch mode. Thus, if the VD remains high and does not go low, end processing does not start. In addition, if the VD remains high and the HD also remains asserted, data continues to be fetched.



**Figure 46.14 Relationship between VIO\_VD and VD Interrupt when VD is High-Active**



**Figure 46.15 Relationship between VIO\_HD and HD Interrupt when HD is High-Active**

#### 46.4.4 Capture Interface Cycle Register (CMCYR)

CMCYR is used to detect an illegal VD and an illegal HD. For HD, the number of cycles from a rising edge of HD to the next rising edge is set (falling edges when low-active is selected for HD). For VD, the number of HD inputs from a rising edge of VD to the next rising edge is set (falling edges when low-active is selected for VD).

Do not modify this register during operation. If this register is modified during operation, correct operation cannot be guaranteed. In addition, the IGRW bit (interrupt source) in CETCR is set to B'1.

Set 0 in all bits of this register, during data enable fetch mode.

Note: After changing the setting of this register, do not start capture until at least 10 external input clock cycles have elapsed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—		VCYL[13:0]													
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—		HCYL[13:0]													
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
29 to 16	VCYL[13:0]	H'0000	R/W	Vertical HD Count of External Module These bits set the number of VD cycles of an external module with the number of HD inputs. The interrupt source bit IGVS in CETCR is set to 1 when the actual number of VD cycles input from the external module differs from this setting. Set these bits for detecting an illegal VD. When these bits are all cleared to 0, the interrupt source bit IGVS in CETCR is not set to 1. Though the interrupt source bit IGVS in CETCR may be set to 1 after the VDPOL bit (VD polarity) in CAMCR is changed, this interrupt should be ignored.
15, 14	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
13 to 0	HCYL[13:0]	H'0000	R/W	Horizontal Cycle Count of External Module These bits set the number of HD cycles of an external module. The interrupt source bit IGHS in CETCR is set to 1 when the actual number of HD cycles input from the external module differs from this setting. Set these bits for detecting an illegal HD. When these bits are all cleared to 0, the interrupt source bit IGHS in CETCR is not set to 1. Though the interrupt source bit IGHS in CETCR may be set to 1 after the HDPOL bit (HD polarity) in CAMCR is changed, this interrupt should be ignored.

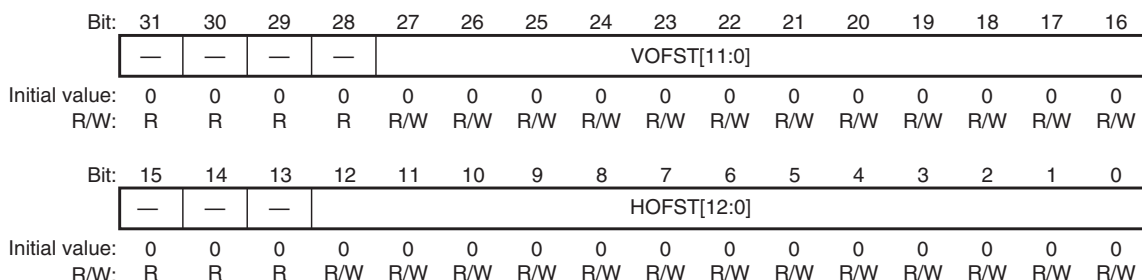


### 46.4.5 Capture Interface Offset Register (CAMOR)

CAMOR sets the location to start capturing when capturing images.

Since the number of HD (horizontal sync signal) inputs from a VD (vertical sync signal) input to the start of a valid image period, and the number of clock cycles from an HD input to the start of a valid image period differ among external modules, these must be set in CAMOR. By setting a value greater than the valid image area, part of the image can be clipped for capture. When fetching data, the setting of this register becomes the number of cycles (HD count) up to the start of a valid data period.

This register is not used, during data enable fetch mode.



Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 16	VOFST[11:0]	H'000	R/W	These bits specify the capture start location in terms of the HD count from a vertical sync signal (1-HD units). The blanking period from a vertical sync signal differs among external modules. Therefore, the vertical capture start location must be specified by these bits in terms of the HD count from a vertical sync signal so that an image can be captured from the valid image area (see Figure 46.16). Some external modules output a vertical sync signal as a data enable signal. In this case, there is no blanking period so these bits must be cleared to 0 (see Figure 46.17).
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 0	HOFST[12:0]	H'0000	R/W	These bits specify the capture start location in terms of the number of clock cycles from a horizontal sync signal (1-cycle units). The blanking period from a horizontal sync signal differs among external modules. Therefore, the horizontal capture start location must be specified by these bits in terms of external input clock cycles from a horizontal sync signal so that an image can be captured from the valid image area. This is similar in data synchronous fetch mode (see Figure 46.18). Some external modules output a horizontal sync signal as a data enable signal. In this case, there is no blanking period so these bits must be cleared to 0 (see Figure 46.19). <b>Note:</b> The first HD (horizontal sync signal) being input simultaneously or after the first VD (vertical sync signal) is the operating condition of the CEU. These inputs are affected by the polarities (set by the VDPOL and HDPOL bits in CAMCR).

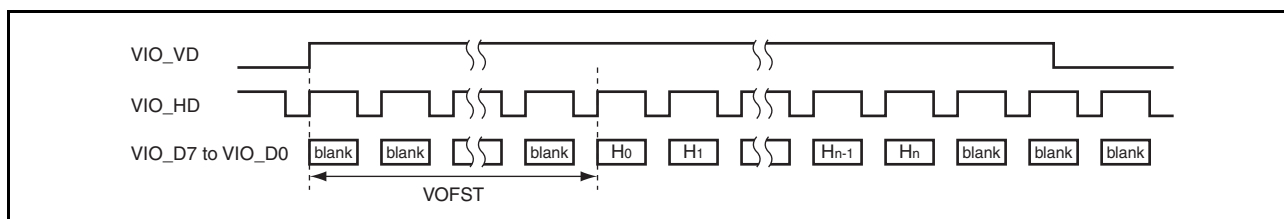


Figure 46.16 Vertical Offset

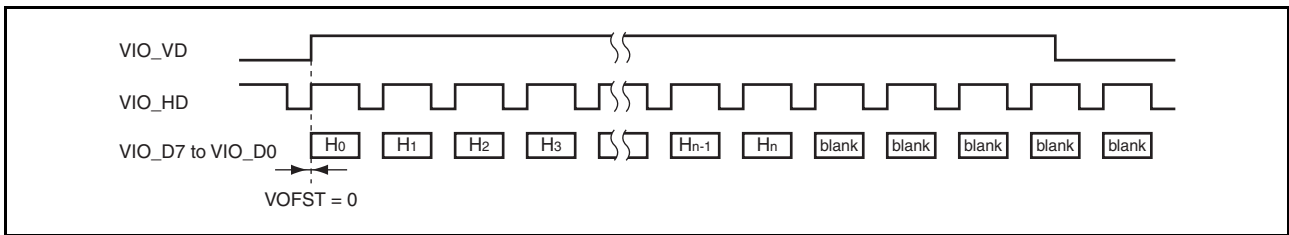


Figure 46.17 Timing when VD is Data Enable Signal

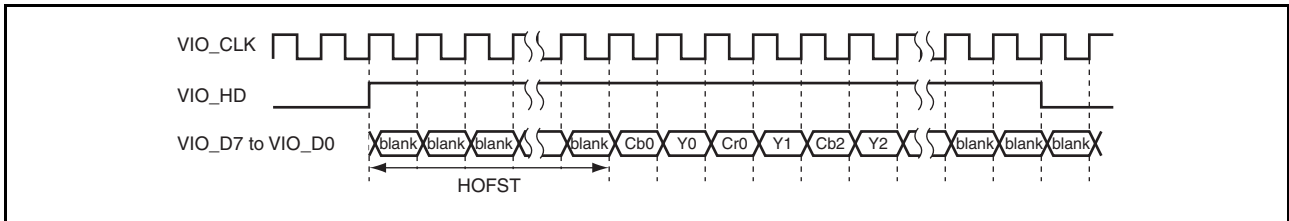


Figure 46.18 Horizontal Offset

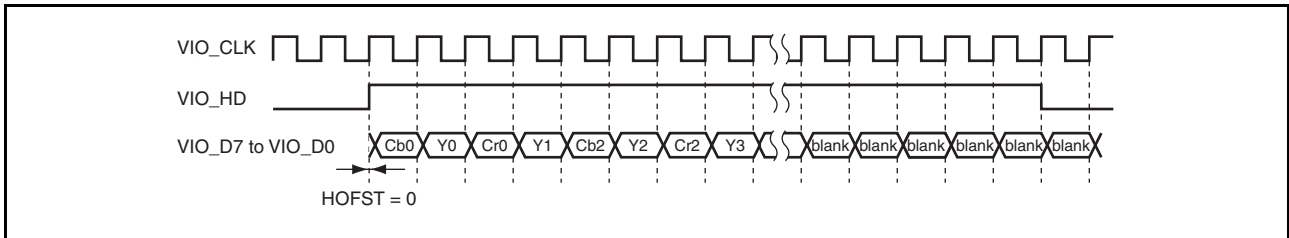


Figure 46.19 Timing when HD is Data Enable Signal (8-bit Interface)

### 46.4.6 Capture Interface Width Register (CAPWR)

CAPWR sets the fetch (capture) cycle width when capturing images.

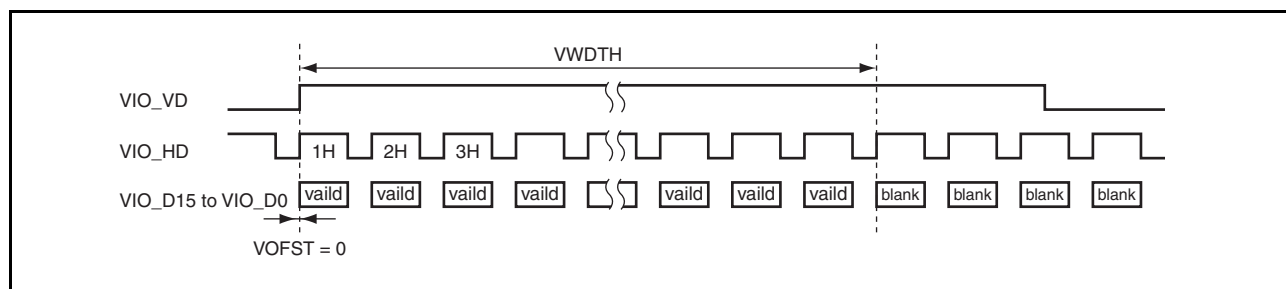
The cycle width unit differs according to the interface and the data type to be captured. For each setting unit, see Table 46.6.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	VWDTH[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	HWDTH[12:0]												
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

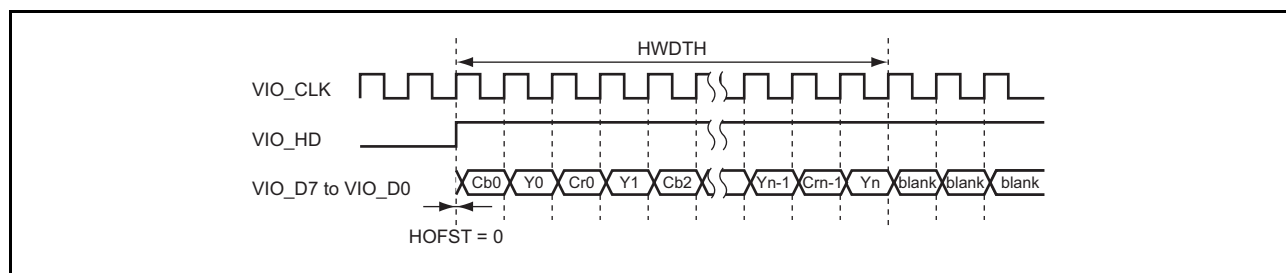
Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 18	VWDTH[11:2]	H'000	R/W	These bits specify the vertical capture period (4-HD units).
17, 16	VWDTH[1:0]		R	These bits specify the number of lines (HD count) to be captured from the location specified by the VOFST bits in CAMOR. Figure 46.20 shows the timing when the vertical blanking period is 0. The CEU captures only the number of lines (HD count) specified by these bits in the vertical direction. Make a setting in the same way to obtain data synchronization. The maximum value to be set is 768 HD (WVGA, VGA).
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 1	HWDTH[12:1]	H'0000	R/W	These bits specify the horizontal capture period.
0	HWDTH[0]		R	These bits specify the number of cycles to be captured from the location specified by the HOFST bits. Figure 46.21 shows the timing when the horizontal blanking period is 0. The CEU captures for only the number of cycles specified by these bits in the horizontal direction. Make a similar setting for data synchronous fetch. The maximum value to be set is as follows: <ul style="list-style-type: none"> <li>• 8-bit interface <ul style="list-style-type: none"> <li>Image capture (8-cycle units) : 2,560 cycles (1,280 pixels)</li> <li>Data synchronous fetch (4-cycle units) : 2,560 cycles (2,560 bytes)</li> </ul> </li> <li>• 16-bit interface <ul style="list-style-type: none"> <li>Image capture (4-cycle units) : 1,280 cycles (1,280 pixels)</li> <li>Data synchronous fetch (2-cycle units) : 1,280 cycles (2,560 bytes)</li> </ul> </li> </ul> Note: In data synchronous fetch mode, set CFSZR and CDWDR according to the values set in this register. For details, see the descriptions on CFSZR and CDWDR.

**Table 46.6 Unit for Setting Fetch (Capture) Cycle Width**

Interface	Vertical Direction		Horizontal Direction	
	Image Capture	Data Synchronous Fetch	Image Capture	Data Synchronous Fetch
8-bit interface	4 HD	4 HD	8 cycles	4 cycles
16-bit interface	4 HD	4 HD	4 cycles	2 cycles



**Figure 46.20 Vertical Capture Timing**



**Figure 46.21 Horizontal Capture Timing (Image Capture with 8-bit Interface)**

### 46.4.7 Capture Interface Input Format Register (CAIFR)

CAIFR sets the input mode (progressive or interlace) for capturing images, the images to be captured (frame, both-field, or one-field), the image from which capturing starts (top field or bottom field), etc. CAIFR is not used in data fetch mode.

Do not modify this register during operation. If this register is modified during operation, correct operation cannot be guaranteed. In addition, the IGRW bit (interrupt source) in CETCR is set to B'1.

The items set by CAIFR are listed in Table 46.7.

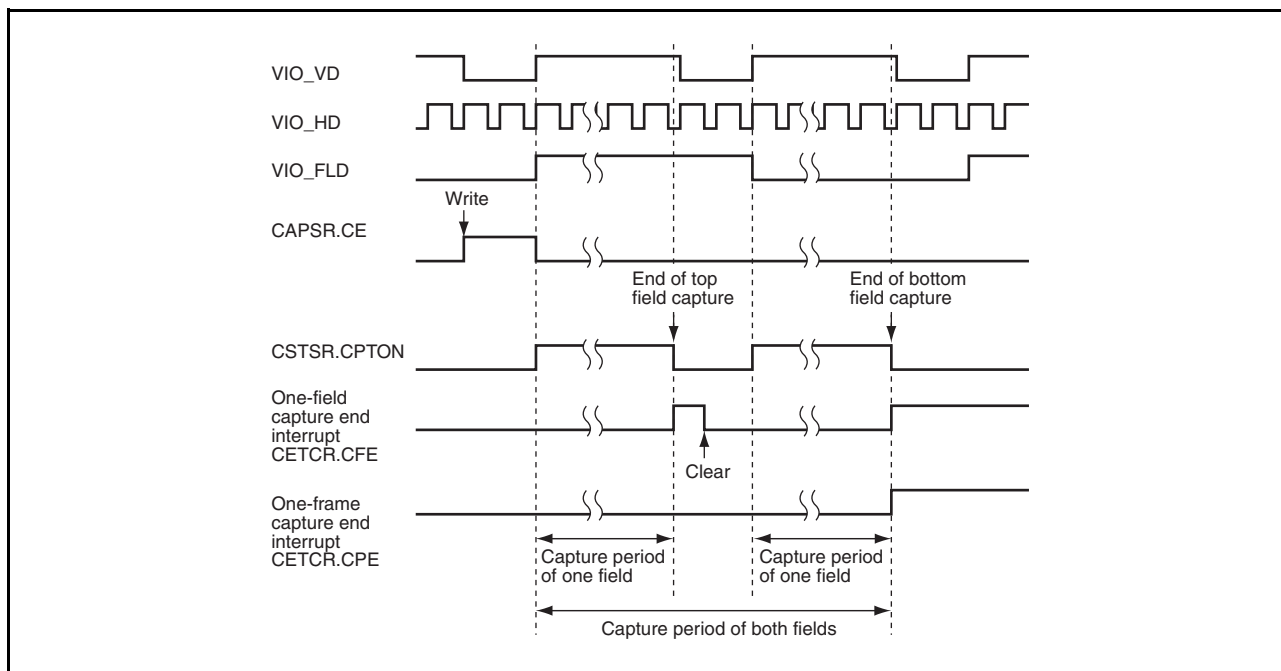
Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	IFS	—	—	—	CIM	—	—	—	FCI[1:0]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	IFS	0	R/W	Sets the input mode for capturing images. 0: Progressive 1: Interlace
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	CIM	0	R/W	Sets the images to be captured. Clear this bit to 0 when the input mode for image capture is progressive (frame image) or when the input mode for image capture is interlace for continuous capture of both the top and bottom fields. Set this bit to 1 when the input mode for image capture is interlace for capture of only a one-field image. 0: Capture of frame image (1 VD) or both-field image (2 VD) 1: Capture of one-field image (1 VD)
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	FCI[1:0]	00	R/W	These bits set the timing to start capturing. The timing to start capturing is set by specifying the image to be captured first. Set these bits to 00 when the input mode is progressive. 00: Capture starts from the VD input immediately after the CEU activation regardless of it being a top or bottom field 01: After the CEU activation, input of a top-field image is waited, and then capture starts from the top field 10: After the CEU activation, input of a bottom-field image is waited, and then capture starts from the bottom field 11: Setting prohibited

**Table 46.7 CAIFR Setting Items**

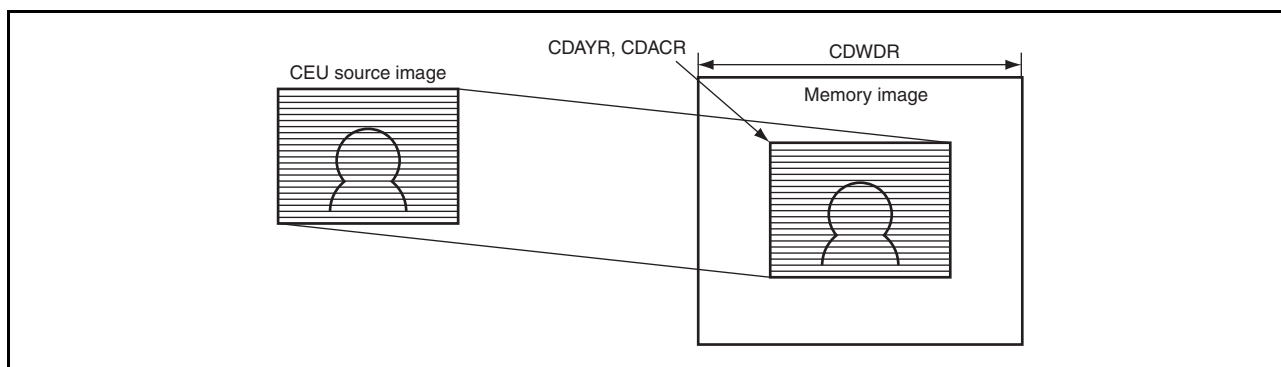
Input Mode	IFS Bit	Captured Image	CIM Bit	Image to Start Capture	FCI Bits		
Progressive	0	Frame	0	Frame immediately after activation	00		
Interlace	1	Both-field (2-VD capture)	0	Field immediately after activation	00		
				Top field	01		
				Bottom field	10		
		Setting prohibited	11				
		One-field (1-VD capture)	1		1	Field immediately after activation	00
						Top field	01
						Bottom field	10
Setting prohibited	11						

In frame image capture and one-field image capture, a one-frame capture end interrupt occurs when capture for 1 VD finishes. In both-field image capture, a one-field capture end interrupt occurs when capture for 1 VD finishes and a one-frame capture end interrupt occurs when capture for 2 VD finishes. At this time, a one-field capture end interrupt occurs simultaneously with a one-frame capture end interrupt. Figure 46.23 shows the timing of a one-frame capture end interrupt and one-field capture end interrupt in both-field image capture.

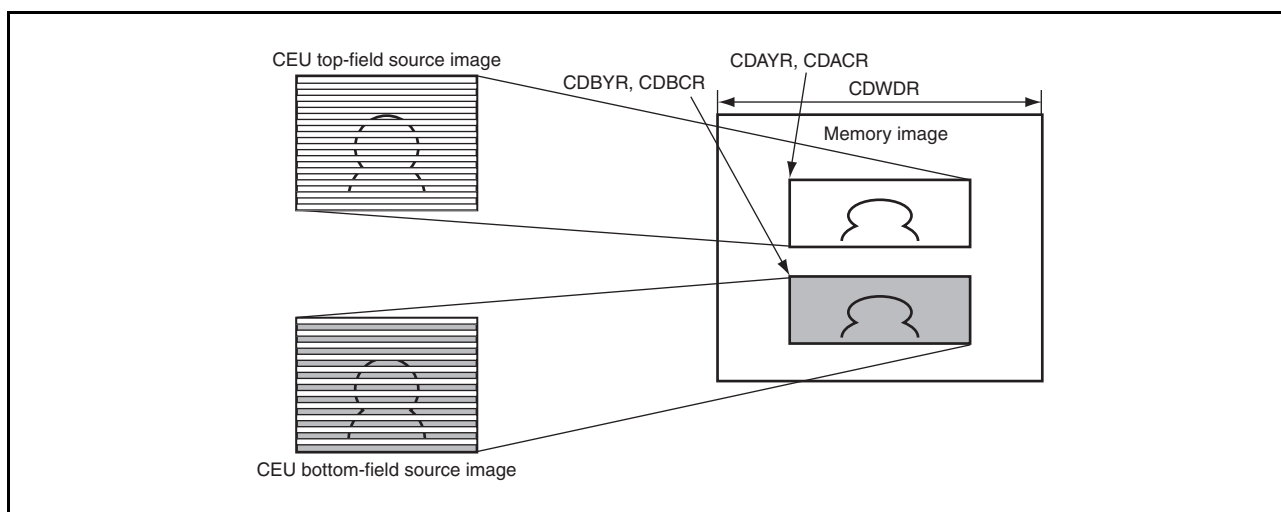


**Figure 46.22 One-Frame Capture End Interrupt and One-Field Capture End Interrupt in Both-Field Image Capture**

A captured frame image or captured one-field image is stored in the memory from the addresses set in CDAYR and CDACR (Figure 46.23). Captured both-field images are stored in different memory areas depending on whether it is a top-field or bottom-field image. A top-field image is stored in the memory from the addresses set in CDAYR and CDACR whereas a bottom-field image is stored in the memory from the addresses set in CDBYR and CDBCRCR (Figure 46.24).

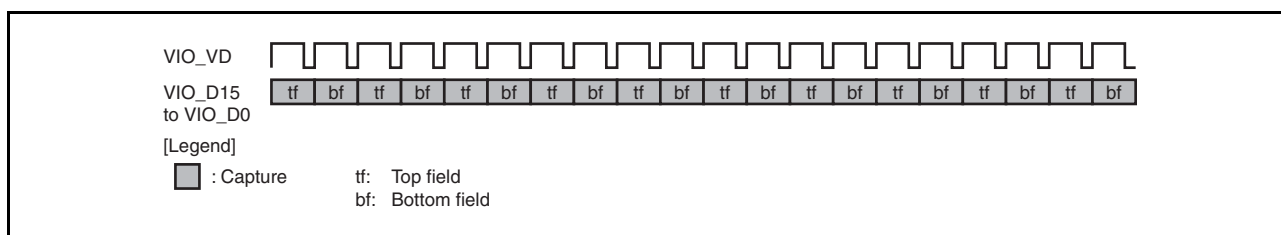


**Figure 46.23 Image of Storing Captured Frame Image or Captured One-Field Image in Memory**

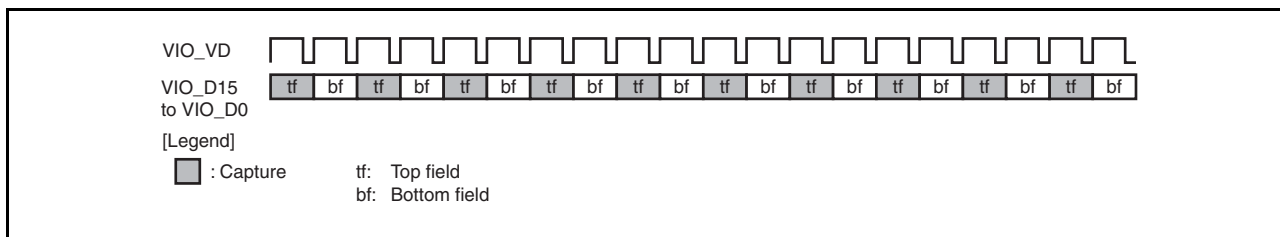


**Figure 46.24 Image of Storing Captured Both-Field Images in Memory**

If the FCI bits are set to B'00 for continuous capture in interlace input mode, images are continuously captured for 2 VD with the first captured field as the reference in both-field image capture (Figure 46.25). In one-field image capture, only the first captured field is continuously captured for 1 VD (Figure 46.26).



**Figure 46.25 Continuous Both-Field Capture in Interlace Mode (Image Immediately after Activation is Top Field (FCI Bits = B'00))**



**Figure 46.26 Continuous One-Field Capture in Interlace Mode (Image Immediately after Activation is Top Field (FCI Bits = B'00))**

**(1) Storage of Interlace Input as Frame Image**

The CEU can store an interlace source image in the memory as a frame image. To store an interlace source image as a frame image, make the following register settings:

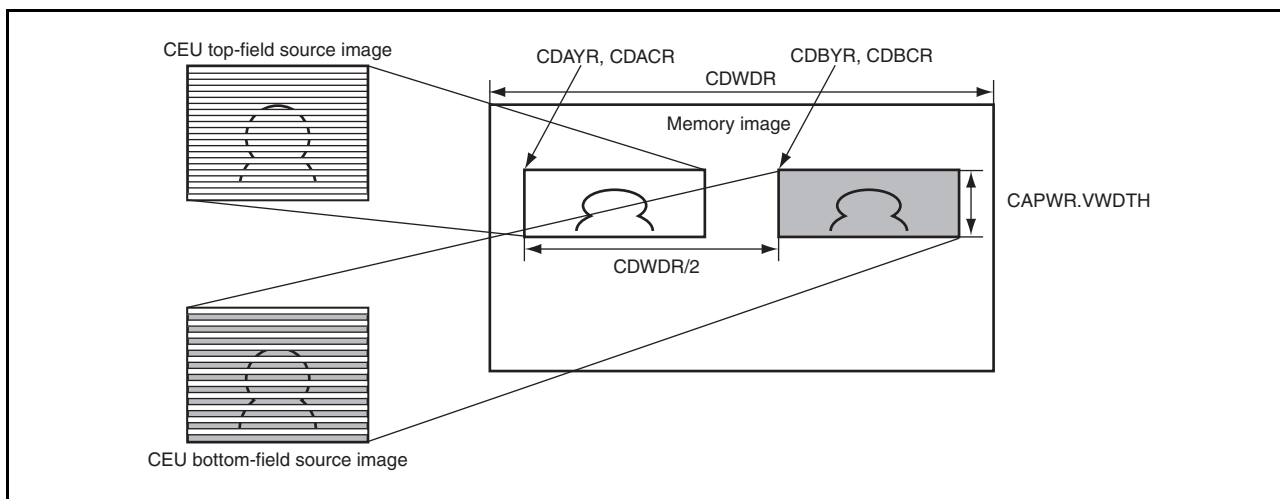
Input mode: Interlace (IFS bit = B'1)

Capture image: Both-field (CIM bit = B'0)

Image to start capture: Any setting other than the prohibited setting (FCI bits = as desired)

Figure 46.27 shows a memory image of capturing both fields of an interlace input and storing it as a frame image in the memory. Set the start addresses of the memory destination for the captured top-field image in CDAYR and CDACR, and the start addresses of the memory destination for the captured bottom-field image in CDBYR and CDBCR. When storing an interlace image as a frame image in the memory, set the horizontal image size of the memory area in CDWDR with the top-field image and bottom-field image placed next to each other as shown in Figure 46.27. In addition, set the number of captured lines of the field image in the VWDTH bits in CAPWR.

A memory image of folding the horizontal image size of the memory area in Figure 46.27 at CDWDR/2 is shown in Figure 46.28. Setting the registers to form the image in Figure 46.27 enables an interlace image to be stored as a frame image in the memory as shown in Figure 46.28.



**Figure 46.27 Image of Storing Captured Both-Fields of Interlace Input in Memory**



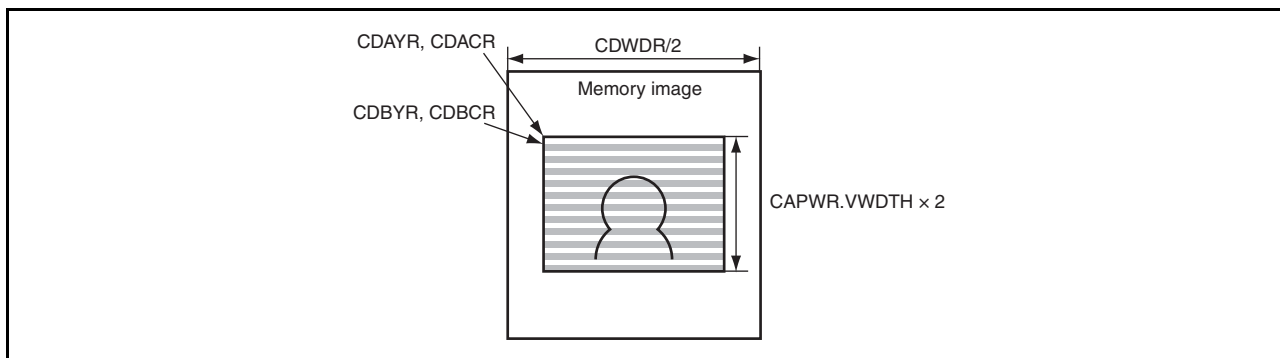


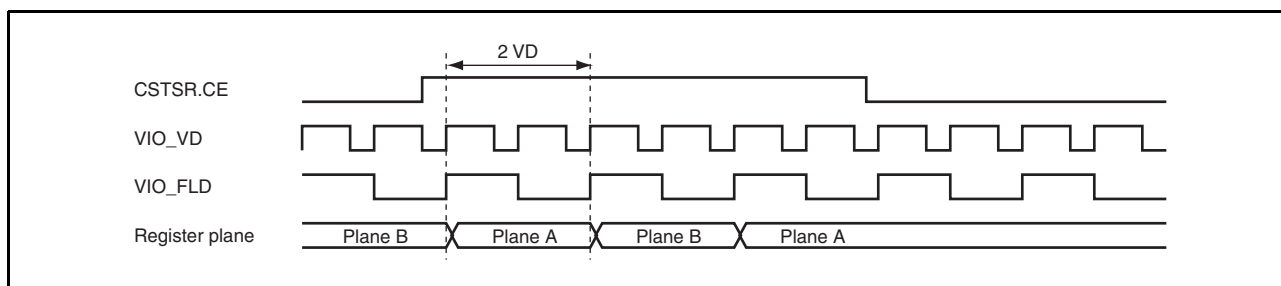
Figure 46.28 Image of Storing Interlace Input as Frame Image in Memory

### 46.4.8 CEU Register Control Register (CRCNTR)

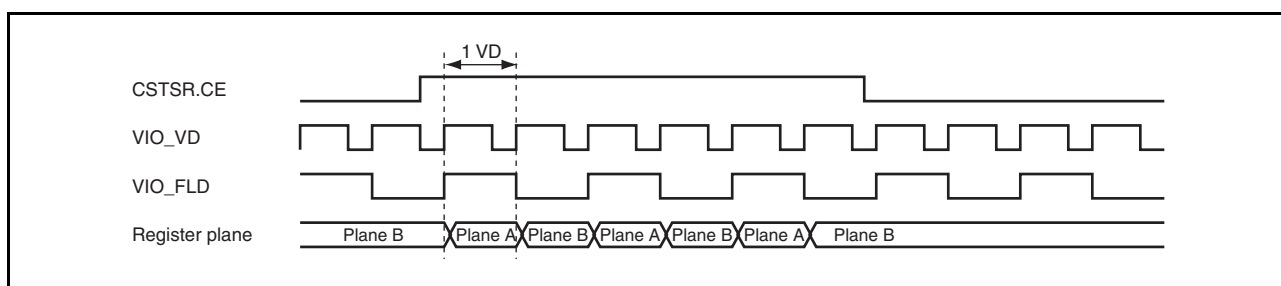
CRCNTR controls switching of the planes of registers with a 2-plane configuration.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	RVS	—	—	RS	RC
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	RVS	0	R/W	Sets the timing to switch the register plane in both-field capture. The setting of this bit is valid only when the RC bit is 1 in both-field capture. 0: Switches the register plane every 2 VD 1: Switches the register plane every 1 VD
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	RS	0	R/W	Specifies which register plane is used by the CEU in synchronization with VD. The setting of this bit is valid only when the RC bit is 0. 0: Uses plane A of the register 1: Uses plane B of the register
0	RC	0	R/W	Specifies switching of the register plane used by the CEU in synchronization with VD. If the register plane is not switched, the register plane specified by the RS bit is used. 0: Uses the specified register plane in synchronization with VD 1: Switches the register plane in synchronization with VD



**Figure 46.29** Timing for Register Plane Switching when RVS Bit is B'0



**Figure 46.30** Timing for Register Plane Switching when RVS Bit is B'1

### 46.4.9 CEU Register Forcible Control Register (CRCMPR)

CRCMPR forcibly controls switching of the planes of registers with a 2-plane configuration. Setting this register enables direct control of register plane switching.

Do not modify this register during operation. If this register is modified during operation, correct operation cannot be guaranteed. In addition, the IGRW bit (interrupt source) in CETCR is set to B'1.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RA
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	RA	0	R/W	Indicates the register plane currently specified. This register value automatically changes in synchronization with VD for starting capture. To start capture with plane A of the register when a setting to switch the register plane in synchronization with VD has been made (RC bit in CRCNTR is 1), specify plane B of the register using this bit. 0: Specifies plane A of the register 1: Specifies plane B of the register

### 46.4.10 Capture Filter Control Register (CFLCR)

CFLCR sets the scale-down factor for the filter to scale images down.

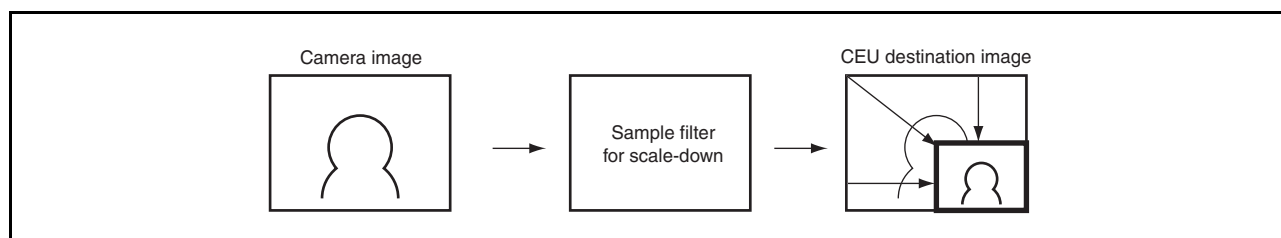
The CEU has an image scale-down filter which can be used to scale down the captured images before storing them in the memory. Set CFLCR to 0 when not performing scale-down (same size output). If a value other than 0 is set in CFLCR, scale-down is performed. In data fetch mode, set CFLCR to 0.

When handling an interlace source image as a frame image, set CFLCR to 0 not to use the filter.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	
	VMANT[3:0]				VFRAC[11:0]												
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	HMANT[3:0]				HFRAC[11:0]												
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	VMANT[3:0]	H'0	R/W	Mantissa Part of Vertical Scale-Down Factor The specifiable range is H'0 to H'F. When H'0 is set for the VMANT bits and H'000 is set for the VFRAC bits, the scale-down filter is not used.
27 to 19 18 to 16	VFRAC[11:3] VFRAC[2:0]	H'000	R/W R	Fraction Part of Vertical Scale-Down Factor The specifiable range is H'000 to H'FF8. The fraction of the scale-down factor that cannot be set with only the VMANT bits must be set with these bits.
15 to 12	HMANT[3:0]	H'0	R/W	Mantissa Part of Horizontal Scale-Down Factor The specifiable range is H'0 to H'F. When H'0 is set for the HMANT bits and H'000 is set for the HFRAC bits, the scale-down filter is not used.
11 to 3 2 to 0	HFRAC[11:3] HFRAC[2:0]	H'000	R/W R	Fraction Part of Horizontal Scale-Down Factor The specifiable range is H'000 to H'FF8. The fraction of the scale-down factor that cannot be set with only the HMANT bits must be specified with these bits.

An image scale-down filter is installed in the CEU, and the captured images can be scaled down and stored in the memory.



**Figure 46.31 Scale-Down of Captured Image**

The formulas for obtaining the MANT (VMANT or HMANT) and FRAC (VFRAC or HFRAC) values from the input pixel count and output pixel count of the filter are shown below. Set the MANT and FRAC bits in order to obtain the desired output pixel count from the number of pixels input to the CEU.

First, calculate preliminary MANT and FRAC values. The parameters needed for calculation are defined as follows:

$$\alpha = MANT \times 4096 + FRAC \quad \dots \text{Formula 1}$$

$$SCL \text{ (scaling factor)} = \frac{4096}{\alpha} \quad \dots \text{Formula 2}$$

Assuming an operator  $\lfloor x \rfloor$  which discards fractions of an integer  $x$ , the MANT and FRAC values can be temporarily set as follows, according to formula 1 and formula 2.

$$MANT = \left\lfloor \frac{1}{SCL} \right\rfloor, \quad FRAC = \left\lfloor 512 \times \left( \frac{1}{SCL} - MANT \right) \right\rfloor \times 8$$

Here, the scaled-down filter output size ( $SIZE_D$ ) can be calculated using the input image size  $S_{in}$  (8-bit interface: half of the CAPWR setting, 16-bit interface: CAPWR setting) in the following formula.

$$SIZE_D = \left\lfloor 1 + \left( \left\lfloor \frac{1}{2} + \frac{S_{in} - 1}{MANT_{pre}} \right\rfloor - 1 \right) \times \frac{MANT_{pre} \times 4096}{\alpha} \right\rfloor \quad \dots \text{Formula 3}$$

$$\left[ \begin{array}{l} MANT_{pre} = 1 \rightarrow (0 \leq MANT < 2) \\ MANT_{pre} = 2 \rightarrow (2 \leq MANT < 4) \\ MANT_{pre} = 4 \rightarrow (4 \leq MANT < 8) \\ MANT_{pre} = 8 \rightarrow (8 \leq MANT) \end{array} \right]$$

The number of output pixels can be obtained by substituting the temporarily calculated MANT, FRAC, and input image size into these formulas. If the calculated number of output pixels is smaller than the number of output pixels used to obtain the preliminary MANT and FRAC values, recalculate with a smaller FRAC ( $\alpha$ ) value, and set the MANT and FRAC values in this register so that a pixel value greater than the desired number of output pixels can be obtained.

Example: Scale down 640 pixels to 480 pixels

$SCL = 480/640 = 3/4$ , and the preliminary settings of  $MANT = 1$ ,  $MANT_{pre} = 1$ , and  $FRAC = H'550$  are made. Substituting these in the following formula results in an output pixel count of 479.

$$SIZE_D = \left\lfloor 1 + \left( \left\lfloor \frac{1}{2} + \frac{S_{in} - 1}{MANT_{pre}} \right\rfloor - 1 \right) \times \frac{MANT_{pre} \times 4096}{\alpha} \right\rfloor \quad \dots \text{Formula 3}$$

Since this output pixel count is smaller than the desired output pixel count of 480, the formula is recalculated with a FRAC value of H'548, a value eight less than the previous time. The obtained result of output pixel count = 480 is equal to the desired output pixel count of 480, so this register is set as  $MANT = 1$  and  $FRAC = H'548$ .

**Table 46.8** Setting Examples for Each Scale-Down Filter Factor

Scale-Down Factor	FRAC		MANT	Input Pixel Count	Output Pixel Count	Clipping Size (CFSZR)
	Decimal	Hexadecimal				
7/8	576	H'240	1	640	560	560
3/4	1352	H'548	1	640	480	480
5/8	2448	H'990	1	640	400	400
1/2	0	H'0	2	640	320	320
3/8	2728	H'AA8	2	640	240	240
1/3	0.0	H'0	3	640	213	212
1/4	0.0	H'0	4	640	160	160
1/5	0.0	H'0	5	640	128	128
1/6	0.0	H'0	6	640	107	104
1/7	0.0	H'0	7	640	91	88
1/8	0.0	H'0	8	640	80	80
1/16	4088	H'FF8	15	640	40	40

Note: This scale-down filter uses a VGA-size line memory for scale-down. Therefore, when an image larger than the VGA size is input for scale-down, settings must be made so that the output image size is equal to or larger than the SubQCIF size and equal to or smaller than the VGA size. When an image is not scaled down (same size output), this restriction does not apply.

#### 46.4.11 Capture Filter Size Clip Register (CFSZR)

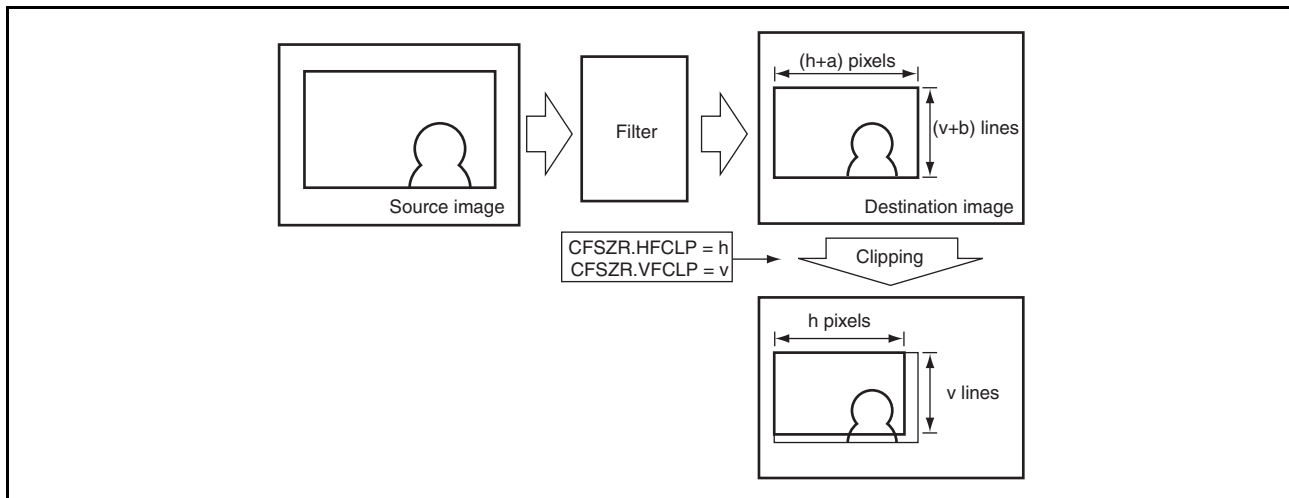
CFSZR sets the clipping size for fine adjustment of the image size output from the filter, and must be set in combination with CFLCR. When clipping the output size of the filter, set the clipping size as a number of pixels, and the setting unit should be four pixels. CFSZR must be set even when scale-down is not performed (same size output).

In data synchronous fetch mode, set CFSZR according to the setting of CAPWR.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	VFCLP[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	HFCLP[11:0]											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 18 17, 16	VFCLP[11:2] VFCLP[1:0]	H'000	R/W R	These bits set the vertical clipping value of the filter output size (4-pixel units).
15 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 2 1, 0	HFCLP[11:2] HFCLP[1:0]	H'000	R/W R	These bits specify the horizontal clipping value of the filter output size (4-pixel units).

The scale-down filter in the CEU may output an odd number of pixels or lines depending on the settings. To adjust the output size of the filter, the CEU clips the destination image by using the number of pixels specified in CFSZR, as shown in Figure 46.32. The clipping size must be specified vertically and horizontally in 4-pixel units.



**Figure 46.32 Clipping of Image Output from Filter**

The pixels to be clipped are counted from the top-left corner of a display. The pixels located to the right of the specified number of pixels or below the specified number of lines are discarded by the clipping function. If the number of pixels specified in CFSZR is larger than that output from the filter, correct operation cannot be guaranteed. To avoid this, the clipping size specified in CFSZR must be equal to or smaller than the number of pixels output from the filter.

Note: In data synchronous fetch mode, the following settings are required. Data cannot be fetched correctly unless the following settings are made.

8-bit interface:  $VFCLP = CAPWR.VWDTH$

$HFCLP = CAPWR.HWDTH/2$

16-bit interface:  $VFCLP = CAPWR.VWDTH$

$HFCLP = CAPWR.HWDTH$

### 46.4.12 Capture Destination Width Register (CDWDR)

CDWDR sets the horizontal image size in the memory area where the captured image is to be output in 4-byte units (4-pixel units).

In data synchronous fetch mode, set CDWDR according to the setting of CAPWR.

This register is not used, during data enable fetch mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	CHDW[12:0]												
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 2 1, 0	CHDW[12:2] CHDW[1:0]	H'0000	R/W R	These bits specify the horizontal image size in the memory area where the captured image is to be stored (4-byte units). The image data captured by the CEU is stored in the memory. If the right end of the captured image does not match the horizontal image size in the memory area as shown in Figure 46.33, some addresses must be skipped at the right end of the image when storing the captured image. Therefore, the horizontal image size in the memory area where the captured image is to be stored must be set in these bits. The maximum value to be set is 8188 bytes (8188 pixels). In data synchronous fetch mode, set as follows: 8-bit interface: CHDW = CAPWR.HWDTH 16-bit interface: CHDW = CAPWR.HWDTH × 2

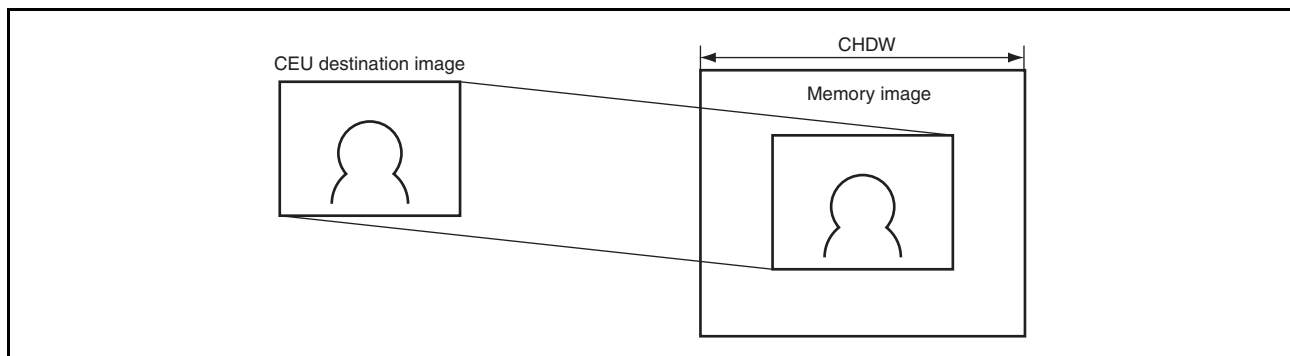
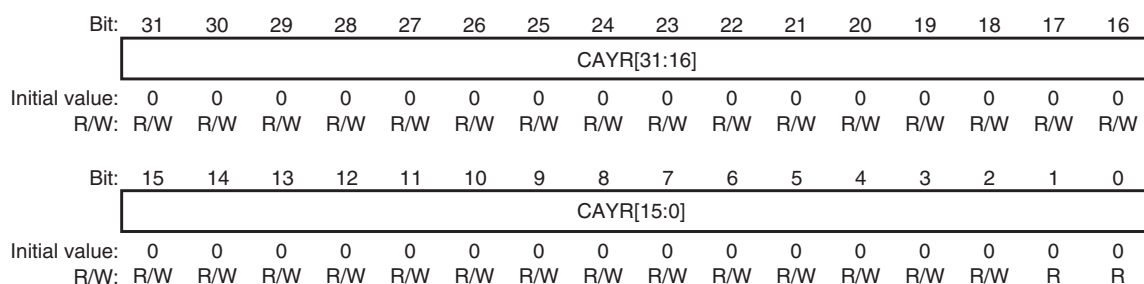


Figure 46.33 Captured Image and Memory Area Image

### 46.4.13 Capture Data Address Y Register (CDAYR)

CDAYR specifies the address where the luminance (Y) component of the captured data is to be stored in frame image capture or one-field image capture, the address where the luminance (Y) component of the captured top field is to be stored in both-field image capture, and the address where the fetched data is to be stored in data fetch. The CEU separates the captured image data into the luminance component data (Y) and the chrominance component data (C), and stores them in the memory via the bus. In frame image capture or one-field image capture, set the start address of the memory area where the Y (luminance) component of the captured data is to be stored by CDAYR. In both-field image capture, set the start address of the memory area where the Y (luminance) component of the captured top-field image is to be stored by CDAYR. In data fetch, set the start address of the memory area where data is to be stored by CDAYR.

Because the address must be specified in 32 bits, the address set by CDAYR must be in longword units. As the setting is in 4-pixel units for image capture and 4-byte units for data fetch, the lower two bits are always fixed to 0.



Bit	Bit Name	Initial Value	R/W	Description
31 to 2 1, 0	CAYR[31:2] CAYR[1:0]	H'0000 0000	R/W R	<ul style="list-style-type: none"> <li>• Frame image capture: These bits set the address for storing the Y (luminance) component data of the captured data (4-pixel units).</li> <li>• One-field image capture: These bits set the address for storing the Y (luminance) component data of the captured data (4-pixel units).</li> <li>• Both-field image capture: These bits set the address for storing the Y (luminance) component data of the captured top-field data (4-pixel units).</li> <li>• Data fetch: These bits set the address for storing data (4-byte units).</li> <li>• Data enable fetch bundle write: These bits set the address for storing data (32-byte units).</li> </ul>

Set the address of the starting point of the memory area where the fetched data is to be stored in this register, as shown in Figure 46.34.

- Frame image capture: Set the address of the starting point of the memory area where the Y component of the captured image is to be stored.
- One-field image capture: Set the address of the starting point of the memory area where the Y component of the captured image is to be stored.
- Both-field image capture: Set the address of the starting point of the memory area where the Y component of the captured top-field image is to be stored.
- Data fetch: Set the address of the starting point of the memory area where the fetched data is to be stored. In data fetch mode, the data is simply stuffed in order from the start address so the end address becomes as follows:

**End address = CDAYR + number of fetched bytes**

- Data enable fetch bundle write: Set the address in 32-byte units.



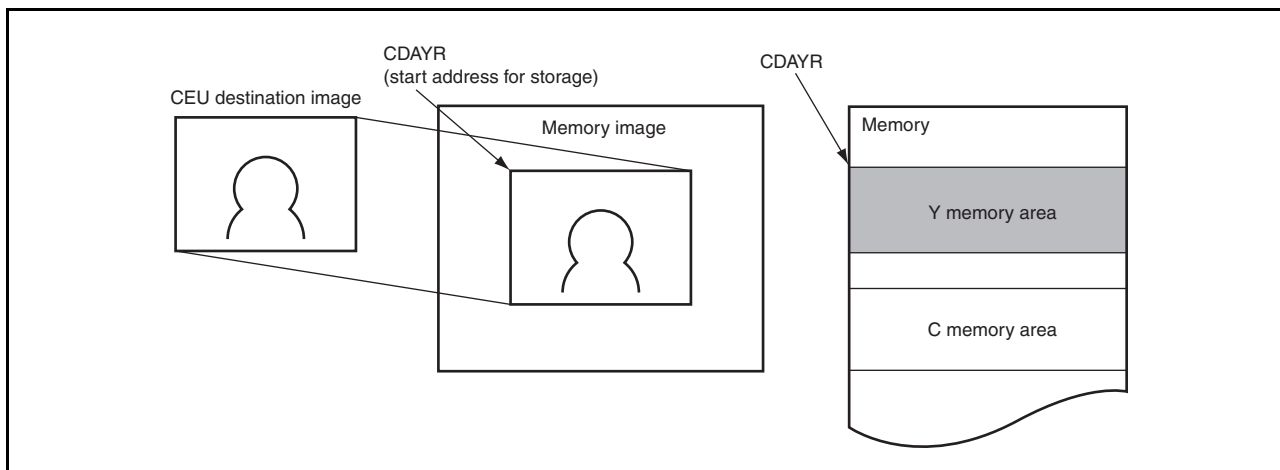


Figure 46.34 Relationship between Captured Image and Y Component Memory Area

### 46.4.14 Capture Data Address C Register (CDACR)

CDACR specifies the address where the chrominance (C) component of the captured data is to be stored in frame image capture or one-field image capture, and the address where the chrominance (C) component of the captured top field is to be stored in both-field image capture. The CEU separates the captured image data into the luminance component data (Y) and the chrominance component data (C), and stores them in the memory via the bus. In frame image capture or one-field image capture, set the start address of the memory area where the C (chrominance) component of the captured data is to be stored by CDACR. In both-field image capture, set the start address of the memory area where the C (chrominance) component of the captured top-field image is to be stored by CDACR. CDACR is not used in data fetch.

Because the address must be specified in 32 bits, the address set by CDACR must be in longword units. As the setting is in 4-pixel units, the lower two bits are always fixed to 0.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CACR[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CACR[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	CACR[31:2]	H'0000 0000	R/W	• Frame image capture: These bits set the address for storing the C (chrominance) component data of the captured data (4-pixel units).
1, 0	CACR[1:0]		R	• One-field image capture: These bits set the address for storing the C (chrominance) component data of the captured data (4-pixel units). • Both-field image capture: These bits set the address for storing the C (chrominance) component data of the captured top-field data (4-pixel units).

Set the address of the starting point of the memory area where the C component of the captured image is to be stored in this register, as shown in Figure 46.35. The C component has an output data format like that in Figure 46.36, and is

saved in the memory in this format.

- Frame image capture: Set the address of the starting point of the memory area where the C component of the captured image is to be stored.
- One-field image capture: Set the address of the starting point of the memory area where the C component of the captured image is to be stored.
- Both-field image capture: Set the address of the starting point of the memory area where the C component of the captured top-field image is to be stored.

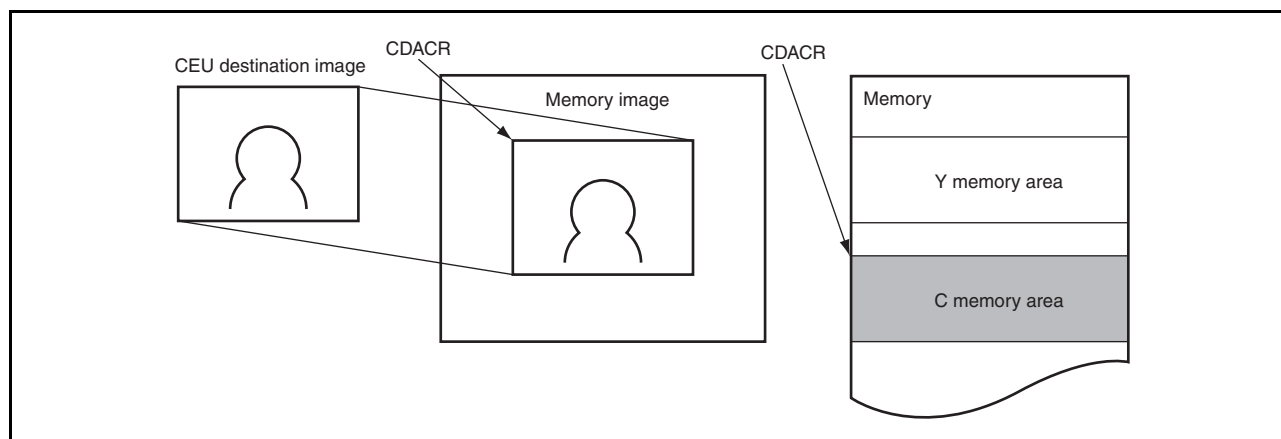


Figure 46.35 Relationship between Captured Image and C Component Memory Area

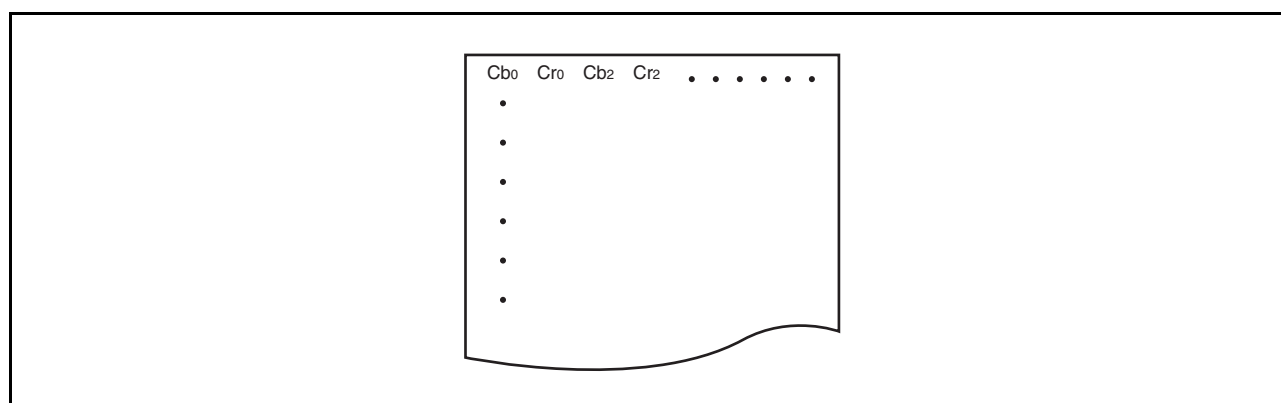
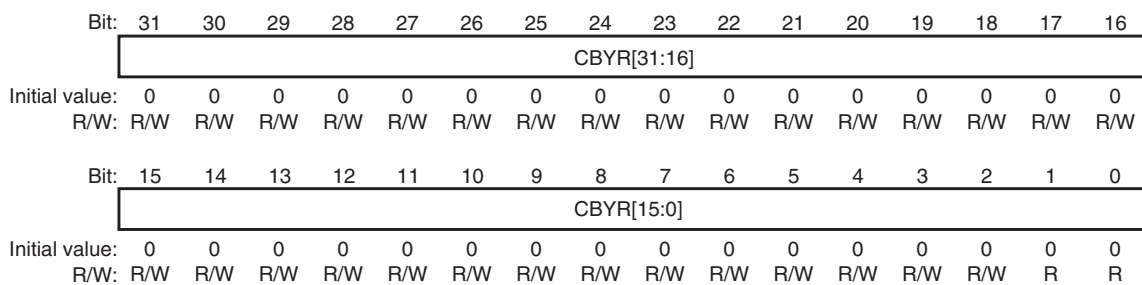


Figure 46.36 Image of Storing C Components in Memory

### 46.4.15 Capture Data Bottom-Field Address Y Register (CDBYR)

CDBYR specifies the address where the luminance (Y) component of the captured bottom-field data is to be stored in both-field image capture. The CEU separates the captured image data into the luminance component data (Y) and the chrominance component data (C), and stores them in the memory via the bus. Set the start address of the memory area where the Y (luminance) component of the captured bottom-field image is to be stored by CDBYR. CDBYR is not used in frame image capture, one-field image capture, or data fetch.

Because the address must be specified in 32 bits, the address set by CDBYR must be in longword units. As the setting is in 4-pixel units, the lower two bits are always fixed to 0.



Bit	Bit Name	Initial Value	R/W	Description
31 to 2	CDBYR[31:2]	H'0000 0000	R/W	These bits set the address for storing the Y (luminance) component data of the captured bottom-field data (4-pixel units).
1, 0	CDBYR[1:0]		R	

Set the address of the starting point of the memory area where the Y component of the captured bottom-field image is to be stored in this register, as shown in Figure 46.37.

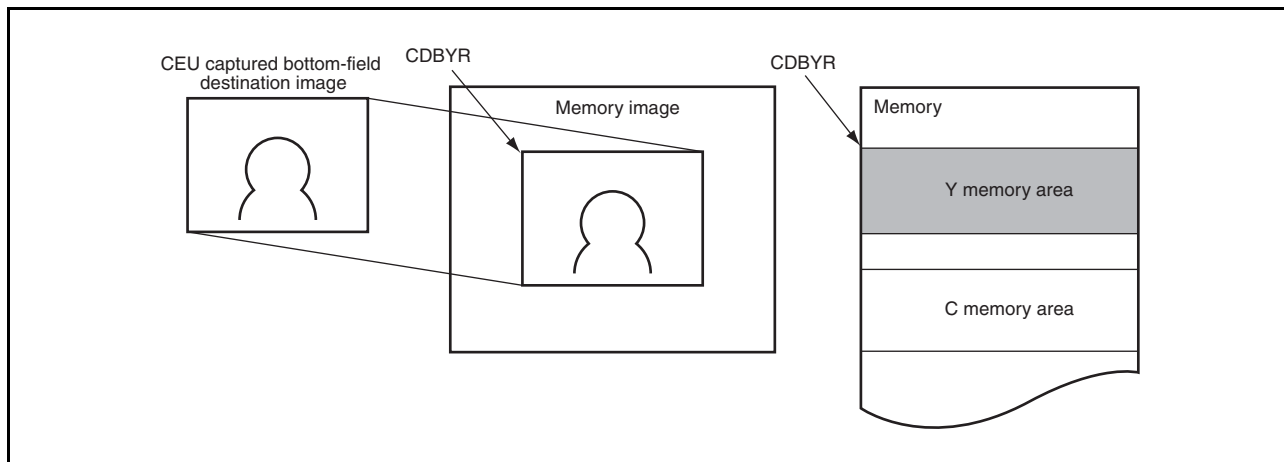
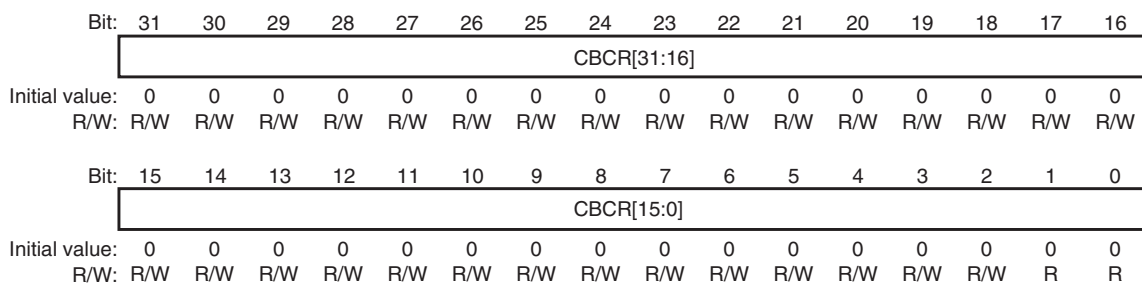


Figure 46.37 Relationship between Captured Bottom-Field Image and Y Component Memory Area

### 46.4.16 Capture Data Bottom-Field Address C Register (CDBCR)

CDBCR specifies the address where the chrominance (C) component of the captured bottom-field data is to be stored in both-field image capture. The CEU separates the captured image data into the luminance component data (Y) and the chrominance component data (C), and stores them in the memory via the bus. Set the start address of the memory area where the C (chrominance) component of the captured bottom-field image is to be stored by CDBCR. CDBCR is not used in frame image capture, one-field image capture, or data fetch.

Because the address must be specified in 32 bits, the address set by CDBCR must be in longword units. As the setting is in 4-pixel units, the lower two bits are always fixed to 0.



Bit	Bit Name	Initial Value	R/W	Description
31 to 2	CBCR[31:2]	H'0000 0000	R/W	These bits set the address for storing the C (chrominance) component data of the captured bottom-field data (4-pixel units).
1, 0	CBCR[1:0]		R	

Set the address of the starting point of the memory area where the C component of the captured bottom-field image is to be stored in this register, as shown in Figure 46.38. The C component has an output data format like that in Figure 46.39, and is saved in the memory in this format.

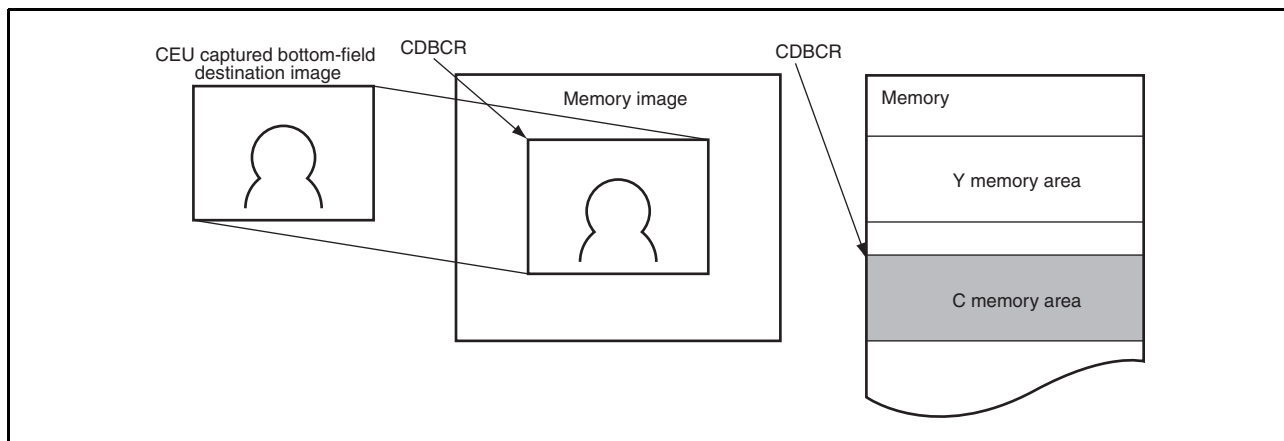


Figure 46.38 Relationship between Captured Bottom-Field Image and C Component Memory Area

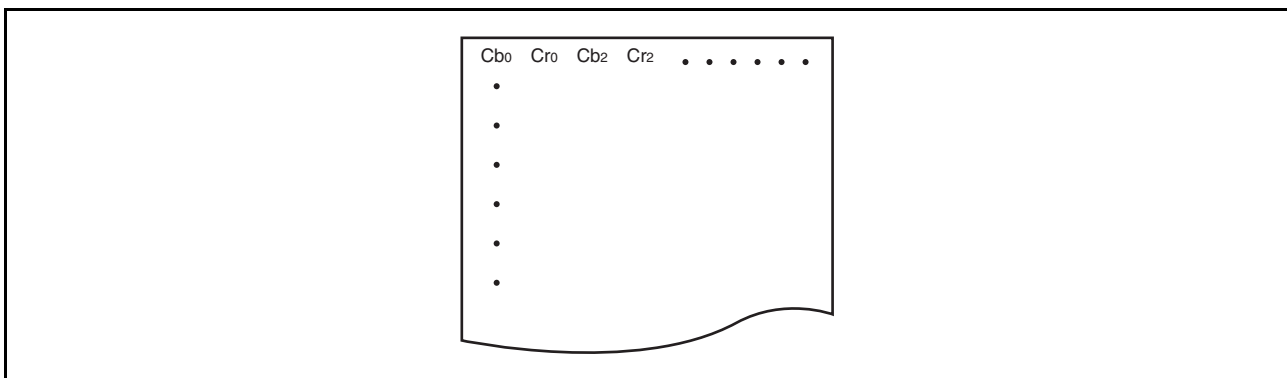


Figure 46.39 Image of Storing C Components in Memory

### 46.4.17 Capture Bundle Destination Size Register (CBDSR)

CBDSR sets the size of output to memory in a bundle write. The number of output lines should be specified for image capture or data synchronous fetch. The number of bytes should be specified for data enable fetch.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	CBVS[22:16]						
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CBVS[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 23	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
22 to 0	CBVS[22:3] CBVS[2:0]	H'000	R/W R	These bits select the number of lines or number of bytes for output to the memory in a bundle write. Image capture and data synchronous fetch: Number of lines for output to the memory in a bundle write. Unit: 8 lines, min.: 8 lines, max.: 768 lines (H'300) Data enable fetch: Number of bytes for output to the memory in a bundle write. Unit: 32 bytes, min.: 512 bytes, max.: 6291456 lines (H'600000)

#### (a) Image capture and data synchronous fetch

Set the number of lines of captured data to be written to the memory by a bundle write as a multiple of eight. This register is valid only when the CBE bit in CDOCR is 1. When the CBE bit in CDOCR is 1 and this register cleared to H'0, this module operates with the number of lines of captured data to be written to the memory as eight. The maximum number of lines that can be set is 768 (H'300). Only bits CBVS[11:3] are valid.

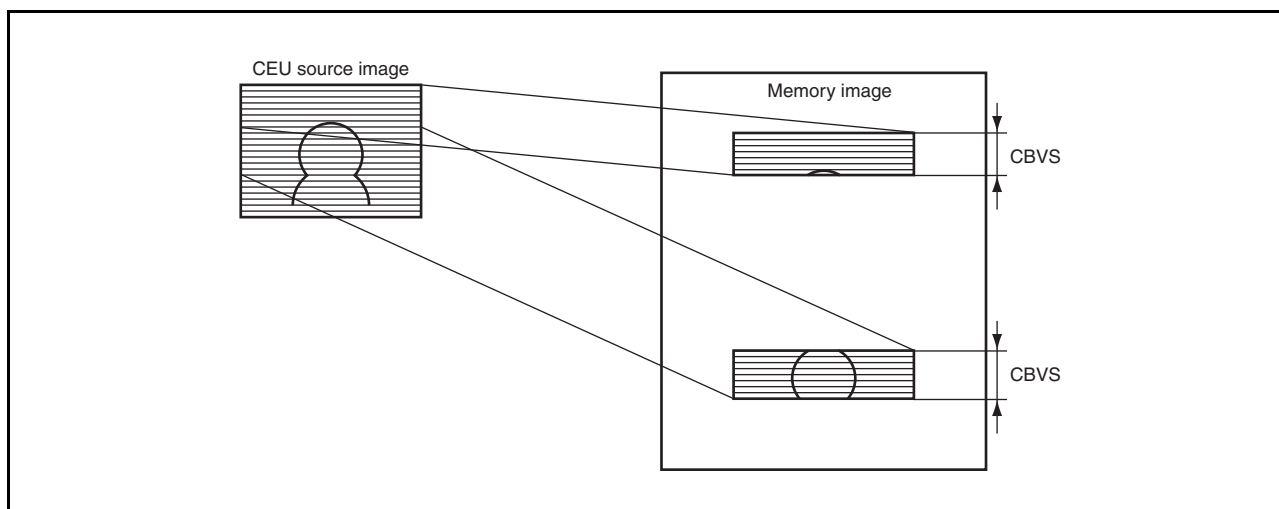


Figure 46.40 Image of Storing Captured Image in Memory by Bundle Write

#### (b) Data enable fetch

Set the number of bytes of captured data to be written to the memory by a bundle write as a multiple of 32. This register is valid only when the CBE bit in CDOCR is 1. The minimum settable size is 512 bytes. When a number smaller than 512 bytes is specified, operation is not guaranteed.

### 46.4.18 Capture Low-Pass Filter Control Register (CLFCR)

CLFCR specifies whether or not to operate the low-pass filter. In data fetch mode, clear the LPF bit to B'0.

The characteristic of the low-pass filter installed in the CEU causes the phase location of the image processed by the low-pass filter to be shifted right by one pixel compared to the raw image.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LPF
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	LPF	0	R/W	Enables or disables operation of the low-pass filter. The low-pass filter removes high-frequency components from the destination image in the horizontal direction. Clear this bit to 0 in data fetch mode. 0: Low-pass filter not used 1: Low-pass filter used (only in the horizontal direction)

### 46.4.19 Firewall Operation Control Register (CFWCR)

CFWCR specifies the upper limit of the write addresses in data enable fetch. When the VD input from an external module does not go low and end notification is not given, this register can prevent writing to memory from being out of control.

This register is enabled only in data enable fetch.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	FWV[26:11]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FWV[10:0]											—	—	—	—	FWE
Initial value:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 5	FWV[26:0]	H'0000008	R/W	These bits specify the upper limit of a write address. Specify the upper 27 bits of the 32-bit address. The upper limit of an address is $FWV[26:0] \ll 5 + H'1F$ .
4 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	FWE	0	R/W	With the setting of FWE = 1, when an address exceeds the value set with FWV, the address is retained and an interrupt source FWF is set. After this, the address is not incremented and data is overwritten on the upper limit address. 0: Firewall is not activated. 1: Firewall is activated.

### 46.4.20 Capture Data Output Control Register (CDOCR)

CDOCR sets the format for outputting captured data to the memory. In data fetch mode, set the CDS bit to B'1.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CBE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CDS	—	COLS	COWS	COBS	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
16	CBE	0	R/W	<p>Controls the number of lines of captured data to be written to the memory.</p> <ul style="list-style-type: none"> <li>Image capture           <p>This bit controls the number of lines of captured data to be written to the memory. When bundle write is set by this register, captured data is written in line units specified by CBDSR to the addresses specified by CDAYR and CDACR, and CDAYR2 and CDACR2 (CDBYR and CDBCR, and CDBYR2 and CDBCR2 for the bottom field in both-field capture) alternately (Figure 46.41). When captured data has been written for the number of lines set by CBDSR, a write end interrupt corresponding to each address setting register occurs. However, after write for one-frame (one-field) capture ends, a bundle write end interrupt does not occur even when bundle write has finished.</p> </li> </ul>
16	CBE	0	R/W	<ul style="list-style-type: none"> <li>Data synchronous fetch           <p>This bit controls the number of lines of captured data to be written to the memory. When bundle write is set by this register, captured data is written in line units specified by CBDSR to the addresses specified by CDAYR and CDAYR2 alternately. When captured data has been written for the number of lines set by CBDSR, a write end interrupt corresponding to each address setting register occurs. However, after write for one-frame capture ends, a bundle write end interrupt does not occur even when bundle write has finished.</p> </li> <li>Data enable fetch           <p>This bit controls the number of bytes of captured data to be written to the memory. When bundle write is set by this register, captured data is written in byte units specified by CBDSR to the addresses specified by CDAYR and CDAYR2 alternately. When captured data has been written for the number of bytes set by CBDSR, a write end interrupt corresponding to each address setting register occurs. Also, only in data enable fetch, a bundle write end interrupt occurs when bundle write has finished after write for one-frame capture ends.</p> <p>Table 46.9 shows the correspondence between address setting registers and write end interrupt sources. Figure 46.42 shows the timing of write end interrupts in image capture and data synchronous fetch. Figure 46.43 shows the timing of write end interrupts in data enable fetch.</p> <p>0: Normal write 1: Bundle write</p> </li> </ul>
15 to 5	—	All 0	R	<p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p>
4	CDS	0	R/W	<p>Sets the image format when outputting the image data captured in the YCbCr422 format to the memory.</p> <p>When 0 is written to this bit, only the luminance component (Y) is output and no chrominance components (Cb and Cr) are output for the odd-numbered lines. With an interlace source image, similarly only the luminance component (Y) is output and no chrominance components (Cb and Cr) are output for the odd-numbered lines of the field. In data fetch mode, set this bit to 1.</p> <p>0: Converts the YCbCr422 format to the YCbCr420 format before outputting data to the memory 1: Outputs data in the YCbCr422 format to the memory without conversion</p>
3	—	0	R	<p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p>
2	COLS	0	R/W	<p>Controls swapping in 32-bit units for data output from the CEU.</p> <p>0: Data is not swapped in 32-bit units 1: Data is swapped in 32-bit units</p>
1	COWS	0	R/W	<p>Controls swapping in 16-bit units for data output from the CEU.</p> <p>0: Data is not swapped in 16-bit units 1: Data is swapped in 16-bit units</p>
0	COBS	0	R/W	<p>Controls swapping in 8-bit units for data output from the CEU.</p> <p>0: Data is not swapped in 8-bit units 1: Data is swapped in 8-bit units</p>



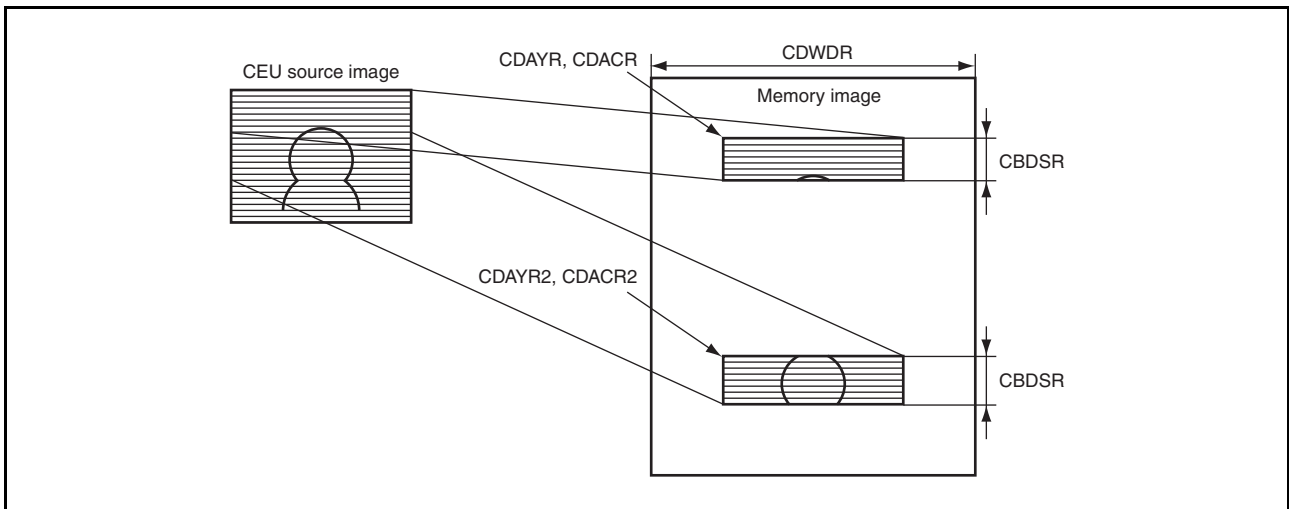


Figure 46.41 Image of Bundle Write to Memory

Table 46.9 Correspondence between Address Setting Registers and Write End Interrupt Sources

Address Setting Registers	Bundle Write End Interrupt Source
CDAYR, CDACR	CPBE1 bit in CETCR
CDAYR2, CDACR2	CPBE2 bit in CETCR
CDBYR, CDBCR	CPBE3 bit in CETCR
CDBYR2, CDBCR2	CPBE4 bit in CETCR

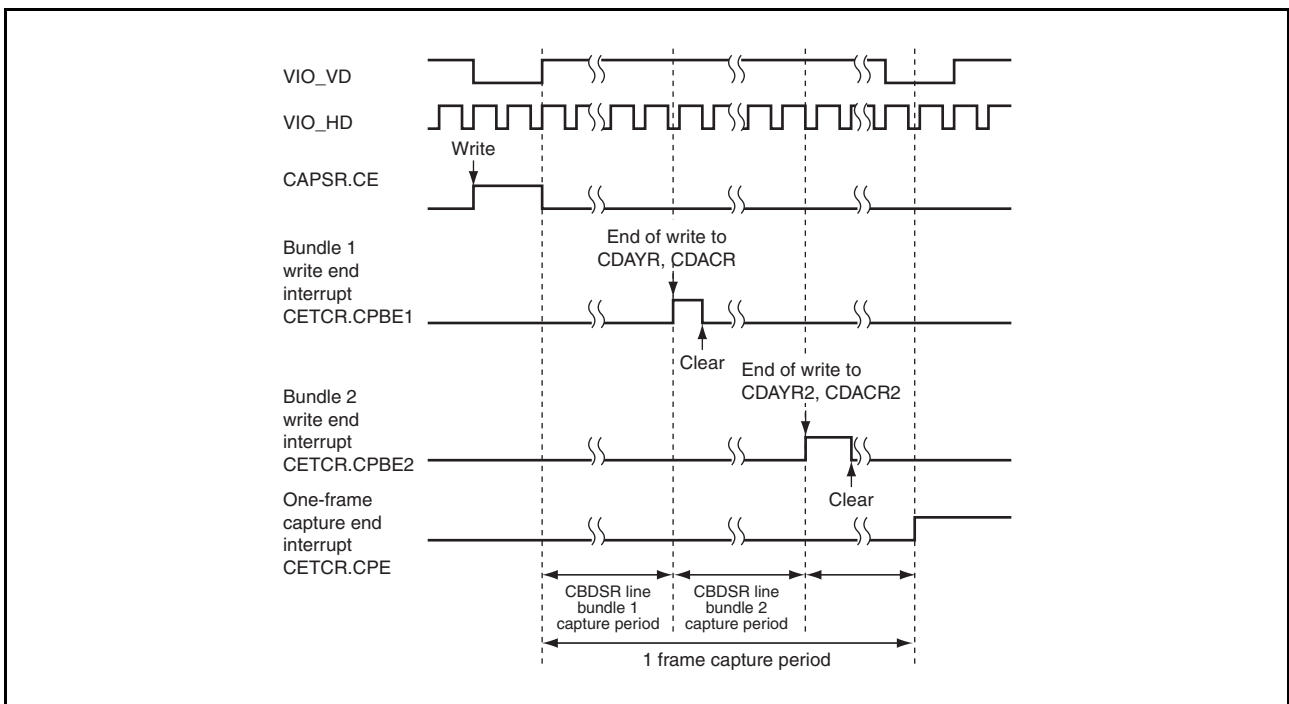
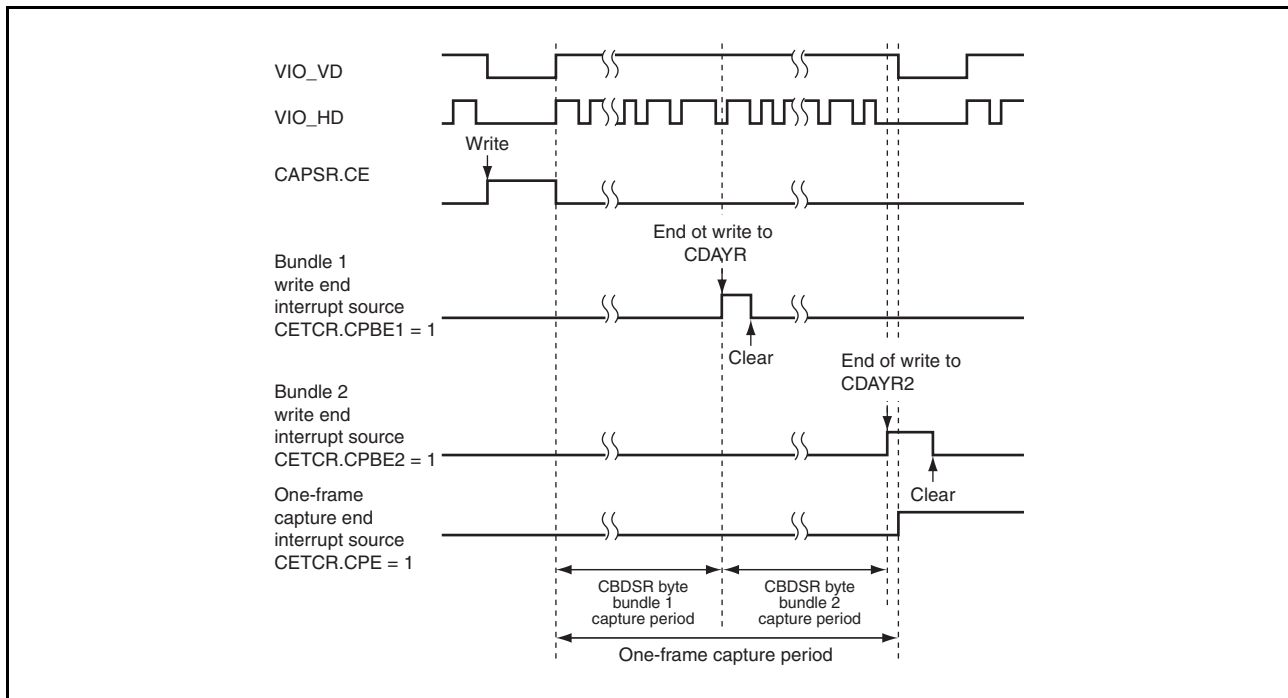


Figure 46.42 Timing of Write End Interrupts (Image Capture, Data Synchronous Fetch)



**Figure 46.43** Timing of Write End Interrupts (Data Enable Fetch)

For data output from the CEU, the COLS, COWS, and COBS bits control swapping in 32-bit, 16-bit, and 8-bit units, respectively. Set these bits when data is misaligned because of a difference in endian. The data swapping bits are shown below. These bits can be set similarly in data fetch mode.

Data can be swapped in 8-bit, 16-bit, 32-bit units, or in 32 bits, 16 bits and 8 bits, as shown in Figure 46.44. To enable data swapping, set the corresponding control bit to B'1.

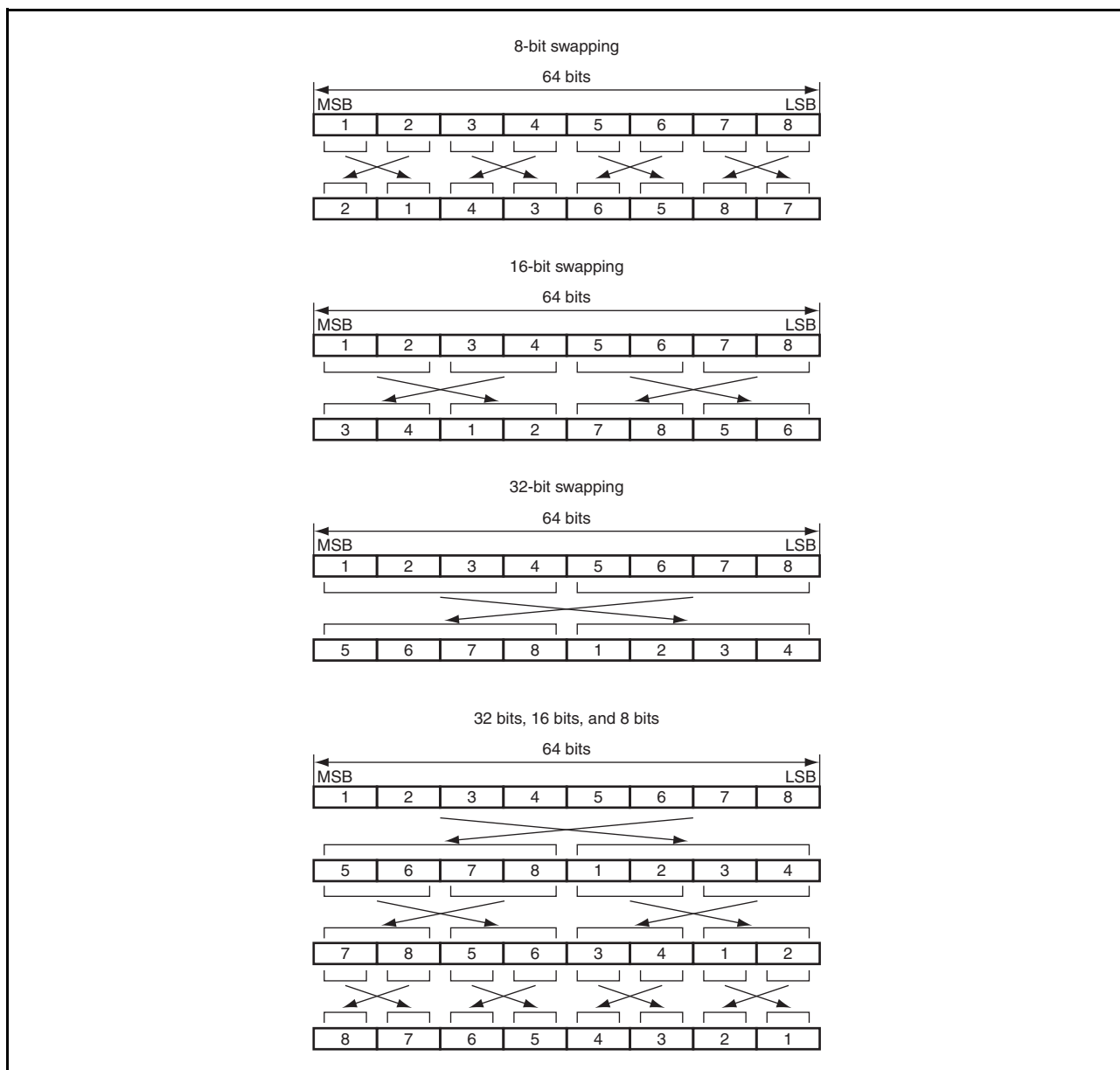


Figure 46.44 Data Swapping by Data Aligner

### 46.4.21 Capture Event Interrupt Enable Register (CEIER)

CEIER enables or disables interrupts of the event flag register that generates CEU interrupts.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	NV DIE	NH DIE	FWFIE	—	—	VB PIE	—	IGV SIE	IGH SIE	CDT OFIE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R	R	R/W	R	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CPBE 4IE	CPBE 3IE	CPBE 2IE	CPBE 1IE	—	—	VDIE	HDIE	—	—	—	IGR WIE	—	—	CF EIE	CP EIE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25	NVDIE	0	R/W	Non-VD Interrupt Enable Disable this interrupt (NVDIE = 0) for data enable fetch. 0: Disables a non-VD interrupt 1: Enables a non-VD interrupt
24	NHDIE	0	R/W	Non-HD Interrupt Enable Disable this interrupt (NHDIE = 0) for data enable fetch. 0: Disables a non-HD interrupt 1: Enables a non-HD interrupt
23	FWFIE	0	R/W	FWF Interrupt Enable 0: Disables a FWF interrupt 1: Enables a FWF interrupt
22, 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	VBPIE	0	R/W	VBP Interrupt Enable 0: Disables a VBP interrupt 1: Enables a VBP interrupt
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
18	IGVSIE	0	R/W	IGVS Interrupt Enable 0: Disables an IGVS interrupt 1: Enables an IGVS interrupt
17	IGHSIE	0	R/W	IGHS Interrupt Enable 0: Disables an IGHS interrupt 1: Enables an IGHS interrupt
16	CDTOFIE	0	R/W	CDTOF Interrupt Enable 0: Disables a CDTOF interrupt 1: Enables a CDTOF interrupt
15	CPBE4IE	0	R/W	CPBE4 Interrupt Enable 0: Disables a CPBE4 interrupt 1: Enables a CPBE4 interrupt
14	CPBE3IE	0	R/W	CPBE3 Interrupt Enable 0: Disables a CPBE3 interrupt 1: Enables a CPBE3 interrupt
13	CPBE2IE	0	R/W	CPBE2 Interrupt Enable 0: Disables a CPBE2 interrupt 1: Enables a CPBE2 interrupt
12	CPBE1IE	0	R/W	CPBE1 Interrupt Enable 0: Disables a CPBE1 interrupt 1: Enables a CPBE1 interrupt
11, 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9	VDIE	0	R/W	VD Interrupt Enable 0: Disables a VD interrupt 1: Enables a VD interrupt
8	HDIE	0	R/W	HD Interrupt Enable 0: Disables an HD interrupt 1: Enables an HD interrupt
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	IGRWIE	0	R/W	Register-Access-During-Capture Interrupt Enable 0: Disables a register-access-during-capture interrupt 1: Enables a register-access-during-capture interrupt
3, 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
1	CFEIE	0	R/W	CFE Interrupt Enable 0: Disables a CFE interrupt 1: Enables a CFE interrupt
0	CPEIE	0	R/W	One-Frame Capture End Interrupt Enable 0: Disables a one-frame capture end interrupt 1: Enables a one-frame capture end interrupt

#### 46.4.22 Capture Event Flag Clear Register (CETCR)

CETCR notifies the CPU of the source of an interrupt that is generated in the CEU. The flags of CETCR can be used as interrupt signals. When the corresponding interrupt is enabled, an interrupt is generated. To clear the interrupt, clear the bit corresponding to the interrupt source to 0. After several cycles have passed after modifying the bit, the interrupt is cleared.

To clear the bit corresponding to the interrupt source to be cleared to 0 and retain that state, write 1 to that bit. For example, to clear only the CPE bit to 0, write H'FFFF FFFE to CETCR.

In CETCR, only bits to which 0 is written are cleared. Bits to which 1 is written retain their current values. To clear an interrupt source, write 0 only to the bit corresponding to the interrupt source to be cleared, and 1 to the other bits.

Note: Since the CETCR value becomes undefined in the following cases, clear all bits in CETCR to 0.

- VD and HD bits immediately after power-on reset or the deep-standby mode is entered
- All bits after the software standby or module standby mode is entered
- VD and HD bits after the polarities of the capture interface sync signals are changed

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	NVD	NHD	FWF	—	—	VBP	—	IGVS	IGHS	CDTOF
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R	R	R/W	R	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CPBE4	CPBE3	CPBE2	CPBE1	—	—	VD	HD	—	—	—	IGRW	—	—	CFE	CPE
Initial value:	0	0	0	0	0	0	—	—	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25	NVD	0	R/W	This bit is used for an interrupt indicating that no VD was input. A non-VD interrupt occurs when the 14-bit internal counter becomes full. Accordingly, this bit is set to 1 when no VD has been input for at least 16,383 lines since the last VD was input.

Bit	Bit Name	Initial Value	R/W	Description
24	NHD	0	R/W	<p>This bit is used for an interrupt indicating that no HD was input. The timing for a non-HD interrupt to occur differs depending on the bit width of the digital image input pins.</p> <p>8-bit digital image input pins: Occurs when the 11-bit internal counter that is incremented every eight cycles becomes full. Accordingly, this bit is set to 1 when no HD has been input for at least 16,376 cycles since the last HD was input.</p> <p>16-bit digital image input pins: Occurs when the 12-bit internal counter that is incremented every four cycles becomes full. Accordingly, this bit is set to 1 when no HD has been input for at least 16,380 cycles since the last HD was input.</p> <p>When connecting a camera whose HD is fixed low when VD is low, this bit may be set to 1.</p> <p>Ignore this interrupt during data enable fetch.</p>
23	FWF	0	R/W	<p>The interrupt is generated when data is written to the address that exceeds the value specified with CFWCR.FMV.</p> <p>This bit is set to 1 when data is written to the address that exceeds the value specified with CFWCR.FMV while CFWCR.FWE = 1.</p>
22, 21	—	0	R	<p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p>
20	VBP	0	R/W	<p>This bit is used for an interrupt indicating that VD has been input while the CEU holds data (insufficient vertical-sync front porch). The conditions for a VBP interrupt to occur are as follows:</p> <ul style="list-style-type: none"> <li>• Condition 1 VD is input when there is captured data within the CEU</li> <li>• Condition 2 The last transfer data cannot be internally detected due to a write buffer overflow or an illegal HD so that the end timing is unclear until the next VD (By generating a VBP interrupt at the VD input timing, capture fail can be announced.)</li> </ul> <p>When a VBP interrupt occurs, a capture end interrupt (CPE bit in CETCR) does not occur and the image of that frame is not captured correctly. Though a capture end interrupt (CPE bit) will occur on rare occasions, it should be ignored in this case. Capturing cannot be performed until the next VD (even if the CE bit (capture reservation signal) in CAPSR is 1, capturing does not start).</p> <p>In the case of condition 2, instead of waiting for a VBP interrupt to occur, execute a software reset (CPKIL bit in CAPSR) to stop capturing and then restart capturing. In this case, since capture operation is terminated without waiting for the next VD, a VBP interrupt does not occur and capturing can be performed from the next VD.</p>
19	—	0	R	<p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p>
18	IGVS	0	R/W	<p>This bit is used for an interrupt generated when the number of VD cycles set in CMCYR differ from the number of VD cycles input from an external module. This bit is set to 1 when there is an illegal VD input from an external module. This bit is set to 1 when the number of HD cycles for the VD input to the CEU differs from the value set in the VCYL bits in CMCYR. Note however that when the VCYL bits are cleared to 0, this interrupt does not occur.</p>
17	IGHS	0	R/W	<p>This bit is used for an interrupt generated when the number of HD cycles set in CMCYR differ from the number of HD cycles input from an external module. This bit is set to 1 when there is an illegal HD input from an external module. This bit is set to 1 when the number of clock cycles for the HD input to the CEU differs from the value set in the HCYL bits in CMCYR. Note however that when the HCYL bits are cleared to 0, this interrupt does not occur.</p>
16	CDTOF	0	R/W	<p>This bit is used for an interrupt indicating that data overflowed in the CRAM of the write buffer.</p> <p>Since data is input at realtime from an external module in capture operations, the frame image is overwritten unless the captured data is transferred from the CEU internal buffer to the memory at a certain or higher transfer rate. This bit is set to 1 when writing the data in the CRAM of the CEU internal write buffer to the bus is not performed within time and data has overflowed.</p>

Bit	Bit Name	Initial Value	R/W	Description
15	CPBE4	0	R/W	<p>This bit is used for an interrupt indicating that writing to CDBYR2 and CDBCR2 in a bundle write has finished.</p> <p>This interrupt is output when the last captured data has been transferred and the end notification received, regardless of the next HD input.</p> <p>This bit is set to 1 when data for the number of lines set in CBDSR has been captured and the last data transfer to the bus has completed.</p> <p>However, this interrupt does not occur when the last captured data in a bundle write is the last captured data of a frame (field).</p>
14	CPBE3	0	R/W	<p>This bit is used for an interrupt indicating that writing to CDBYR and CDBCR in a bundle write has finished.</p> <p>This interrupt is output when the last captured data has been transferred and the end notification received, regardless of the next HD input.</p> <p>This bit is set to 1 when data for the number of lines set in CBDSR has been captured and the last data transfer to the bus has completed.</p> <p>However, this interrupt does not occur when the last captured data in a bundle write is the last captured data of a frame (field).</p>
13	CPBE2	0	R/W	<p>This bit is used for an interrupt indicating that writing to CDAYR2 and CDACR2 in a bundle write has finished.</p> <p>This interrupt is output when the last captured data has been transferred and the end notification received, regardless of the next HD input.</p> <p>This bit is set to 1 when data for the number of lines (number of bytes in data enable fetch) set in CBDSR has been captured and the last data transfer to the bus has completed.</p> <p>However, in image capture or data synchronous fetch, this interrupt does not occur when the last captured data in a bundle write is the last captured data of a frame (field).</p>
12	CPBE1	0	R/W	<p>This bit is used for an interrupt indicating that writing to CDAYR and CDACR in a bundle write has finished.</p> <p>This interrupt is output when the last captured data has been transferred and the end notification received, regardless of the next HD input.</p> <p>This bit is set to 1 when data for the number of lines (number of bytes in data enable fetch) set in CBDSR has been captured and the last data transfer to the bus has completed.</p> <p>However, in image capture or data synchronous fetch, this interrupt does not occur when the last captured data in a bundle write is the last captured data of a frame (field).</p>
11, 10	—	All 0	R	<p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p>
9	VD	Undefined	R/W	<p>This bit is used for an interrupt indicating that VD (vertical sync signal) was input from an external module.</p> <p>This bit is set to 1 when a VD input from an external module is detected.</p> <p>Immediately after the VDPOL bit in CAMCR is modified, a pseudo VD is input and this bit is set to 1. The VD interrupt after the VDPOL bit is modified should be ignored.</p>
8	HD	Undefined	R/W	<p>This bit is used for an interrupt indicating that HD (horizontal sync signal) was input from an external module.</p> <p>This bit is set to 1 when an HD input from an external module is detected.</p> <p>Immediately after the HDPOL bit in CAMCR is modified, a pseudo HD is input and this bit is set to 1. The HD interrupt after the HDPOL bit is modified should be ignored.</p>
7 to 5	—	All 0	R	<p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p>
4	IGRW	0	R/W	<p>This bit is used for an interrupt indicating that during capturing, access was attempted to a register to which writing during operation is prohibited.</p> <p>Among the CEU registers, writing during capturing is prohibited for some registers. Table 46.10 shows which registers can/cannot be written to during capturing. This bit is set to 1 when a register to which writing during capturing is prohibited has been written to.</p>
3, 2	—	All 0	R	<p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p>

Bit	Bit Name	Initial Value	R/W	Description
1	CFE	0	R/W	This bit is used for an interrupt indicating that capturing of one field from an external module has finished. This interrupt is output when the last captured data has been transferred and the end notification received, regardless of the next VD input (see Figure 46.45). This interrupt occurs only in both-field capture mode.
0	CPE	0	R/W	This bit is used for an interrupt indicating that capturing of one frame from an external module has finished. This interrupt is output when the last captured data has been transferred and the end notification received, regardless of the next VD input. This interrupt indicates that capturing of one frame has finished. This bit is set to 1 when the image of the size set in CAPWR is captured and the last data transfer to the bus finished (see Figure 46.46).

**Table 46.10 Registers that Can/Cannot Be Modified during Capturing**

Register Name	Modification during Capturing
CAPSR	Possible
CAPCR	Prohibited
CAMCR	Prohibited
CMCYR	Prohibited
CAMOR	Possible
CAPWR	Possible
CAIFR	Prohibited
CRCNTR	Possible
CRCMPR	Prohibited
CFLCR	Possible
CFSZR	Possible
CDWDR	Possible
CDAYR	Possible
CDACR	Possible
CDBYR	Possible
CDBCR	Possible
CBDSR	Possible
CFWCR	Possible
CLFCR	Possible
CDOCR	Possible
CEIER	Possible
CETCR	Possible
CSTSR	Prohibited
CDSSR	Prohibited
CDAYR2	Possible
CDACR2	Possible
CDBYR2	Possible
CDBCR2	Possible



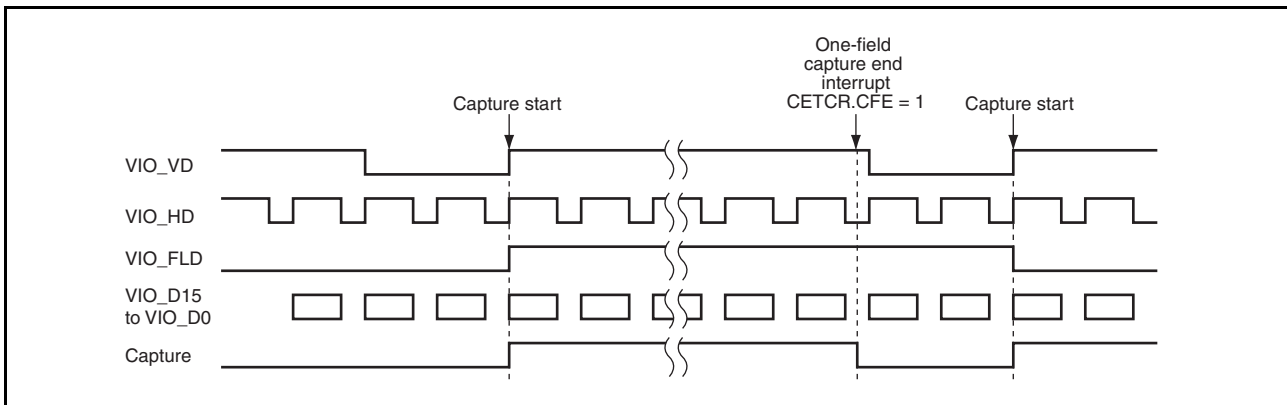


Figure 46.45 CFE Generation Timing

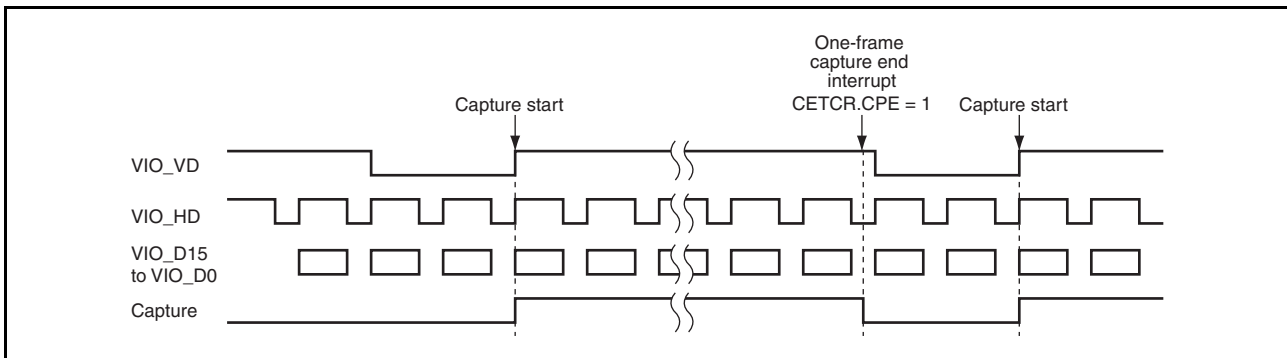


Figure 46.46 CPE Generation Timing

### 46.4.23 Capture Status Register (CSTSR)

CSTSR indicates the internal status of the CEU. CSTSR differs from CETCR in that no interrupt is generated for the events indicated in CSTSR.

The CEU operating/halt state can be confirmed using CSTSR. To confirm the halt state of the CEU, make sure that the status bit (bit 0) indicating that the CEU is operating is cleared to 0 for sure.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	CRST	—	—	—	—	—	—	—	CP FLD
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CP TON
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
24	CRST	0	R	Indicates which register plane is currently used. 0: Plane A of the register is being used 1: Plane B of the register is being used
23 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	CPFLD	0	R	Indicates which field is being captured. 0: Bottom field is being captured 1: Top field is being captured
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	CPTON	0	R	Indicates that the CEU is operating. This bit retains 1 during the period that starts from the internal VD at capture start and ends when a one-frame capture end interrupt occurs. Figure 46.47 shows the CEU operating period.

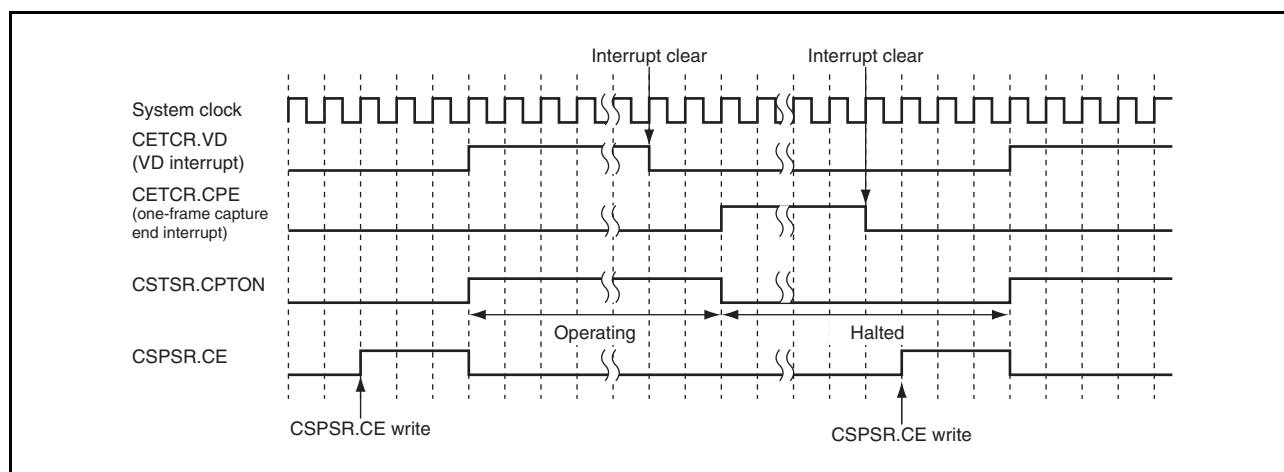


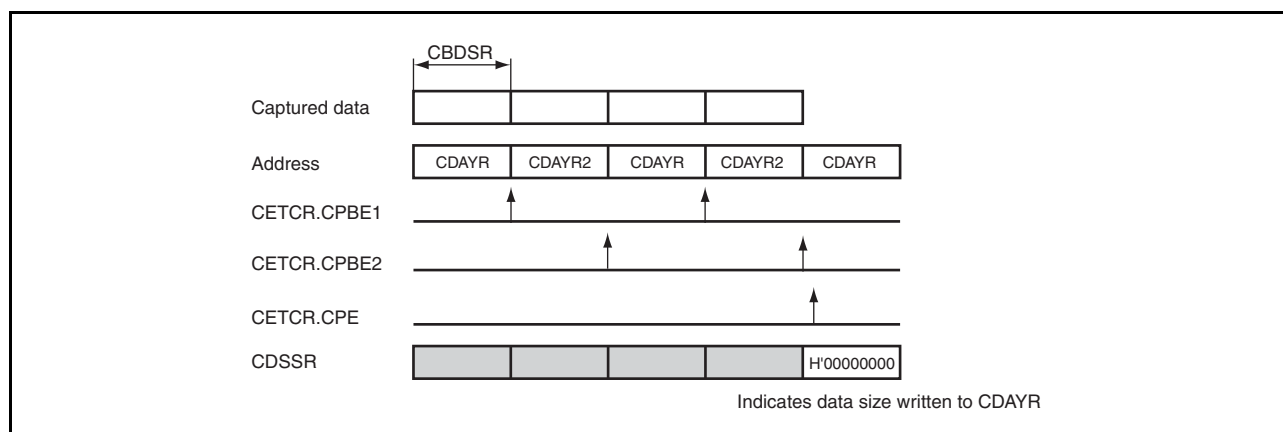
Figure 46.47 Operating Status during Capturing

### 46.4.24 Capture Data Size Register (CDSSR)

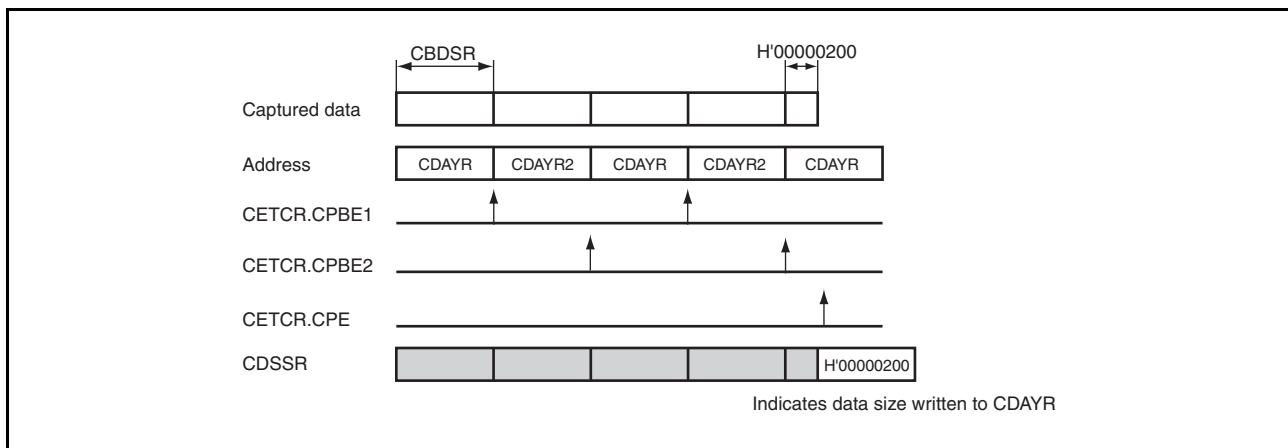
CDSSR indicates the size of data written to the memory in data enable fetch. As this register indicates a correct value at the end of capture, confirm this register when capture is completed.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CDSS[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CDSS[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	CDSS[31:0]	H'0000 0000	R	<p>Indicate the size of data written to the memory in data enable fetch. In a bundle write, size of data written to the selected address at the end of one-frame capture is indicated. In a bundle write, as soon as the number of bytes specified by CBDSR is transferred to the bus, address to which data is written is switched.</p> <p>Therefore, if one-frame capture is completed at the same time as a bundle write is completed, this register indicates H'0000 0000.</p> <p>Figure 46.48 and Figure 46.49 show the overall timing of the CDSSR operation in a bundle write.</p>



**Figure 46.48 Overall Timing of CDSSR Operation in Bundle Write (When Bundle Write End and Capture End Coincide)**



**Figure 46.49 Overall Timing of CDSSR Operation in Bundle Write  
(When Bundle Write End and Capture End Do Not Coincide)**

#### 46.4.25 Capture Data Address Y Register 2 (CDAYR2)

CDAYR2 specifies the address for the luminance (Y) component used in a bundle write and the address for data storage in a bundle write in data fetch. CDAYR2 is used only in a bundle write.

CDAYR2 specifies the address where the Y component of the captured data is to be stored in frame image capture or one-field image capture, the address where the Y component of the captured top field is to be stored in both-field image capture, and the address where data is to be stored in data fetch. The CEU separates the captured image data into the luminance component data (Y) and the chrominance component data (C), and stores them in the memory via the bus. In frame image capture or one-field image capture, set the start address of the memory area where the Y component of the captured data is to be stored by CDAYR2. In both-field image capture, set the start address of the memory area where the Y component of the captured top-field image is to be stored by CDAYR2. In data fetch, set the start address of the memory area to be used for data storage.

The address specified by this register must be in 32-bit units. As the setting is in 4-pixel units, the lower two bits are always fixed to 0.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CAYR2[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CAYR2[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	CAYR2[31:2]	H'0000 0000	R/W	• Frame image capture: These bits set the address for storing the Y component data of the captured data (4-pixel units).
1, 0	CAYR2[1:0]		R	• One-field image capture: These bits set the address for storing the Y component data of the captured data (4-pixel units).
				• Both-field image capture: These bits set the address for storing the Y component data of the captured top-field data (4-pixel units).
				• Data synchronous fetch: These bits set the address for storing data (4-byte units).
				• Dana enable fetch: These bits set the address for storing data (32-byte units).

Set the address of the starting point of the memory area where the Y component of the captured image is to be stored in this register, as shown in Figure 46.50.

- Frame image capture: Set the address of the starting point of the memory area where the Y component of the captured image is to be stored.
- One-field image capture: Set the address of the starting point of the memory area where the Y component of the captured image is to be stored.
- Both-field image capture: Set the address of the starting point of the memory area where the Y component of the captured top-field image is to be stored.

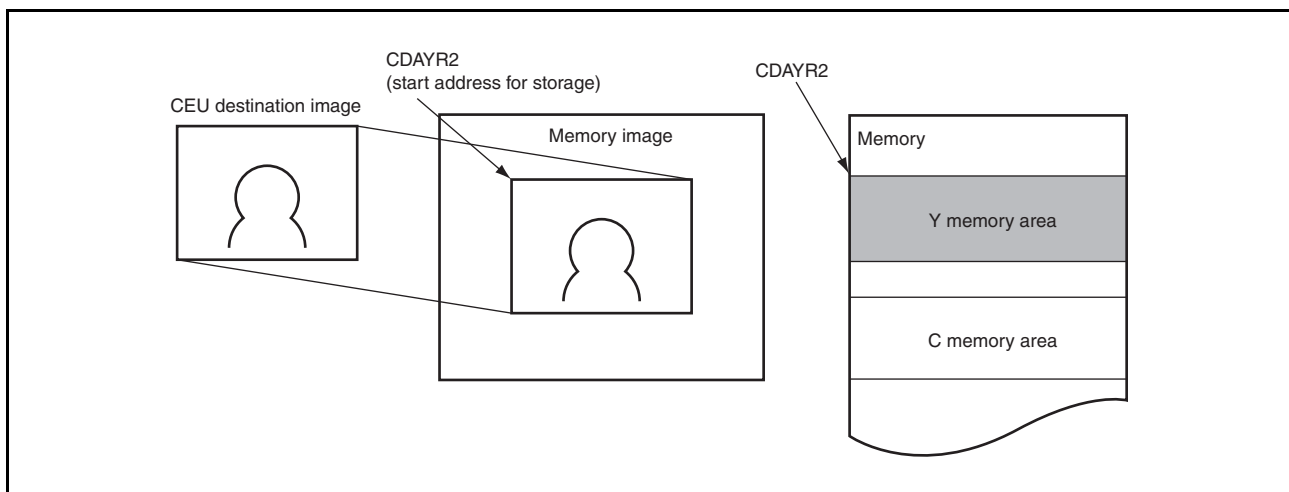


Figure 46.50 Relationship between Captured Image and Y Component Memory Area

### 46.4.26 Capture Data Address C Register 2 (CDACR2)

CDACR2 specifies the address for the chrominance (C) component used in a bundle write. CDACR2 is used only in a bundle write.

CDACR2 specifies the address where the C component of the captured data is to be stored in frame image capture or one-field image capture, and the address where the C component of the captured top field is to be stored in both-field image capture. The CEU separates the captured image data into the luminance component data (Y) and the chrominance component data (C), and stores them in the memory via the bus. In frame image capture or one-field image capture, set the start address of the memory area where the C component of the captured data is to be stored by CDACR2. In both-field image capture, set the start address of the memory area where the C component of the captured top-field image is to be stored by CDACR2. CDACR2 is not used in data fetch.

The address specified by this register must be in 32-bit units. As the setting is in 4-pixel units, the lower two bits are always fixed to 0.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CACR2[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CACR2[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	CACR2[31:2]	H'0000 0000	R/W	<ul style="list-style-type: none"> <li>Frame image capture: These bits set the address for storing the C component data of the captured data (4-pixel units).</li> <li>One-field image capture: These bits set the address for storing the C component data of the captured data (4-pixel units).</li> <li>Both-field image capture: These bits set the address for storing the C component data of the captured top-field data (4-pixel units).</li> </ul>
1, 0	CACR2[1:0]		R	

In this register, set the address of the starting point of the memory area where the captured data is to be stored in a bundle write, as shown in Figure 46.51. The C component has an output data format as shown in Figure 46.52, and is saved in the memory in this format.

- Frame image capture: Set the address of the starting point of the memory area where the C component of the captured image is to be stored.
- One-field image capture: Set the address of the starting point of the memory area where the C component of the captured image is to be stored.
- Both-field image capture: Set the address of the starting point of the memory area where the C component of the captured top-field image is to be stored.

The C component has an output data format as shown in Figure 46.51, and is saved in the memory in this format.

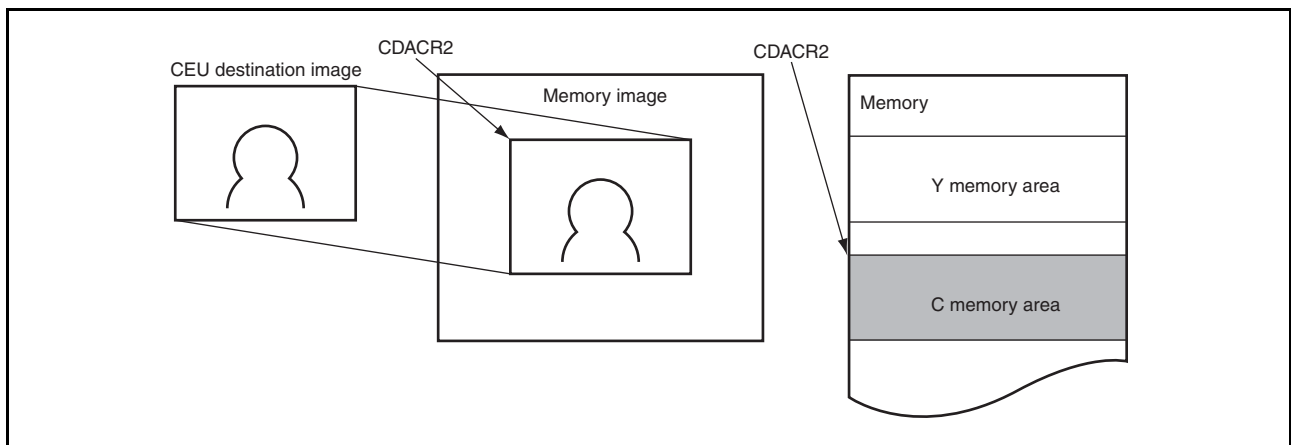


Figure 46.51 Relationship between Captured Image and C Component Memory Area

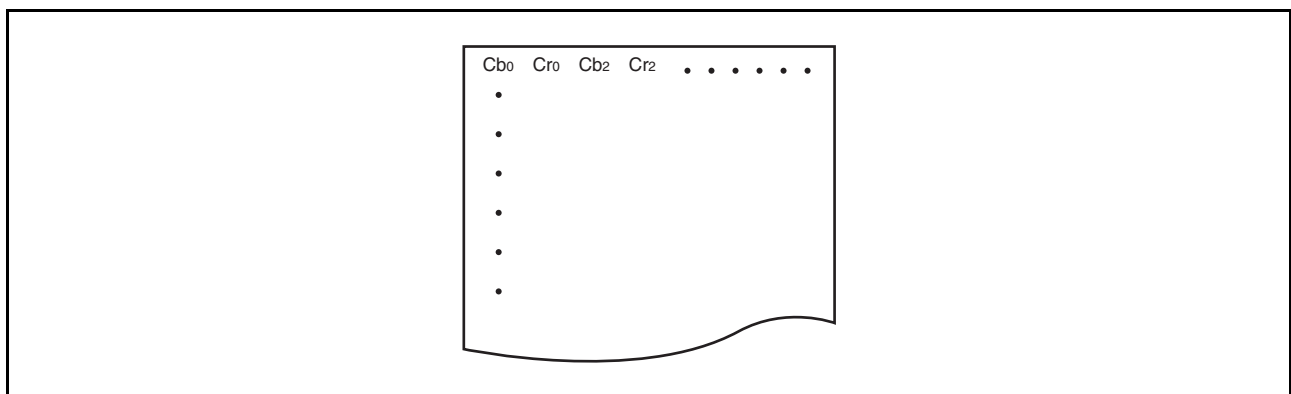


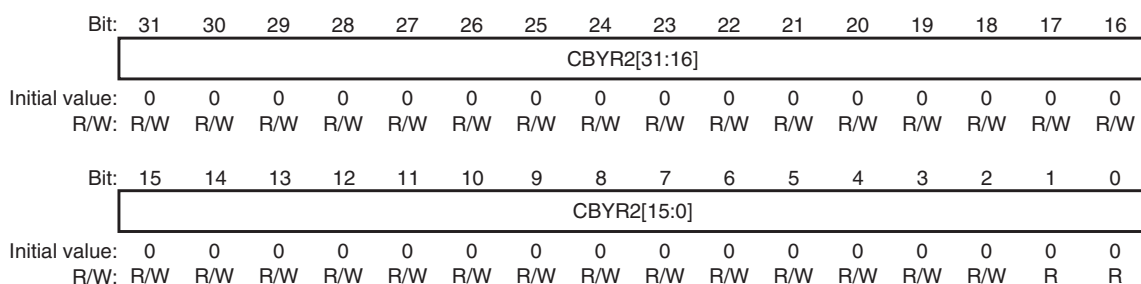
Figure 46.52 Image of Storing C Components in Memory

### 46.4.27 Capture Data Bottom-Field Address Y Register 2 (CDBYR2)

CDBYR2 specifies the address for the luminance (Y) component of the bottom field used in a bundle write. CDBYR2 is used only in a bundle write.

CDBYR2 specifies the address where the Y component of the captured bottom-field data is to be stored in both-field image capture. The CEU separates the captured image data into the luminance component data (Y) and the chrominance component data (C), and stores them in the memory via the bus. Set the start address of the memory area where the Y component of the bottom-field image captured in both-field image capture is to be stored by CDBYR2. CDBYR2 is not used in frame image capture, one-field image capture, or data fetch.

The address specified by this register must be in 32-bit units. As the setting is in 4-pixel units, the lower two bits are always fixed to 0.



Bit	Bit Name	Initial Value	R/W	Description
31 to 2	CBYR2[31:2]	H'0000 0000	R/W	These bits set the address for storing the Y component data of the captured bottom-field data (4-pixel units).
1, 0	CBYR2[1:0]		R	

In this register, set the address of the starting point of the memory area where the Y component of the captured bottom-field image is to be stored in a bundle write, as shown in Figure 46.53.

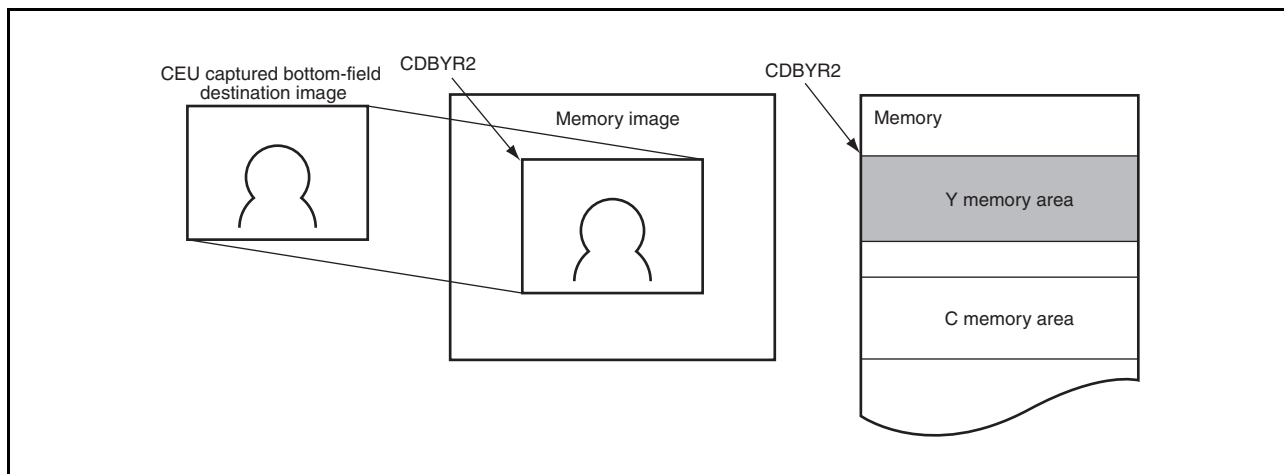


Figure 46.53 Relationship between Captured Bottom-Field Image and Y Component Memory Area

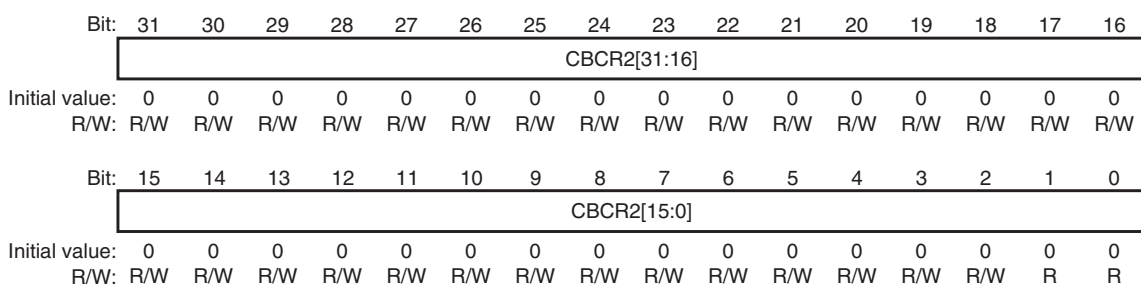


### 46.4.28 Capture Data Bottom-Field Address C Register 2 (CDBCR2)

CDBCR2 specifies the address for the chrominance (C) component of the bottom field used in a bundle write. CDBCR2 is used only in a bundle write.

CDBCR2 specifies the address where the C component of the captured bottom-field data is to be stored in both-field image capture. The CEU separates the captured image data into the luminance component data (Y) and the chrominance component data (C), and stores them in the memory via the bus. Set the start address of the memory area where the C component of the bottom-field image captured in both-field image capture is to be stored by CDBCR2. CDBCR2 is not used in frame image capture, one-field image capture, or data fetch.

The address specified by this register must be in 32-bit units. As the setting is in 4-pixel units, the lower two bits are always fixed to 0.



Bit	Bit Name	Initial Value	R/W	Description
31 to 2	CBCR2[31:2]	H'0000 0000	R/W	These bits set the address for storing the C component data of the captured bottom-field data (4-pixel units).
1, 0	CBCR2[1:0]		R	

In this register, set the address of the starting point of the memory area where the C component of the captured bottom-field image is to be stored in a bundle write, as shown in Figure 46.54. The C component has an output data format as shown in Figure 46.55, and is saved in the memory in this format.

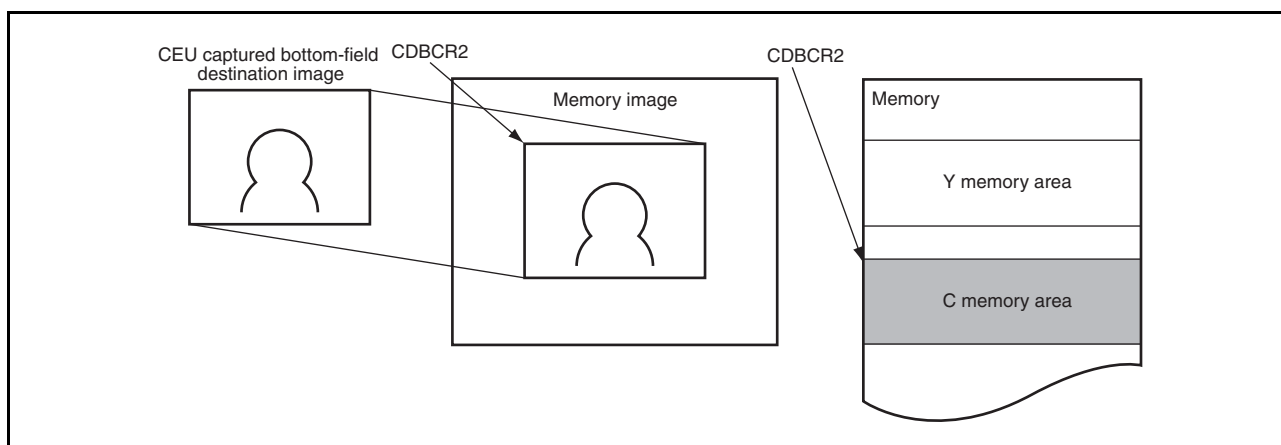


Figure 46.54 Relationship between Captured Bottom-Field Image and C Component Memory Area

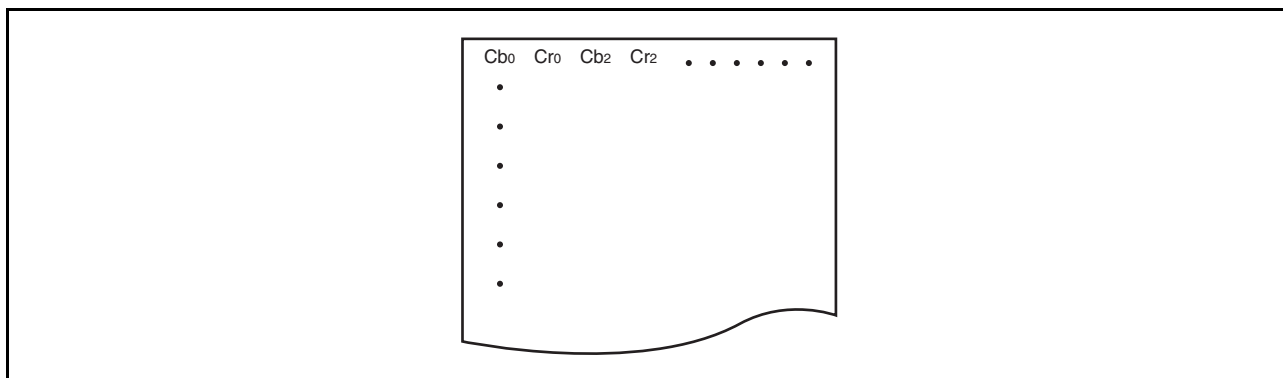


Figure 46.55 Image of Storing C Components in Memory

## 46.5 Usage Notes for CEU

### 46.5.1 Conditions for Connection to an External Module

#### (1) Clock Frequency

The external input clock should have a frequency at least the same as the CEU operating clock frequency ( $B\phi$ ), with jitter on both sides included.

$$\text{CEU operating clock frequency (B}\phi\text{)} \geq \text{External input clock frequency}$$

#### (2) Blanking Period

The period from the last valid pixel in each line to the next horizontal sync signal HD must be at least 20 cycles.

#### (3) Fixed Period of Field Identification Signal

The field identification signal FLD should be fixed for at least 1-HD period since a VD input

## 46.5.2 Restrictions on Input/Output Functions

Table 46.11 lists the restrictions regarding the CEU input/output functions.

**Table 46.11 Restrictions on CEU Input/Output Functions**

Item	Restrictions
External module interface	The operating clock of the external module (VIO_CLK) should always have a frequency at most the same as that of the CEU operating clock (Bφ), with jitter on both sides included.
	Selecting the interface, or modifying the frequency of the external module operating clock or HD/VD polarity must be done when capture operations are halted for sure.
	The capture horizontal size in image capture must be specified as follows: 8-bit interface: 8-cycle units 16-bit interface: 4-cycle units
	The capture horizontal size in data fetch must be specified as follows: 8-bit interface: 4-cycle units 16-bit interface: 2-cycle units
External module interface	The capture vertical size must be specified in 4-line units.
	The maximum number of cycles in the horizontal sync signal period should be 16,375 cycles of external input clock for 8-bit digital image input pins, or 16,379 cycles of external input clock for 16-bit digital image input pins.
	The maximum number of lines (HD count) in the vertical sync signal should be 16,382 lines.
	The minimum number of captured pixels should be sub-QCIF (128 × 96).
	The maximum number of captured pixels should be WVGA (1,280 × 768) pixels.
	Captured size in data enable fetch Maximum: 6 Mbytes (2,048 × 1,536 × 2) Minimum: 16 bytes (must be specified in 4-byte units)
Memory output	The output address must be specified in 32-bit units.
	The horizontal size of the destination image (memory) must be specified in 4-pixel units.
	The number of horizontal output pixels (= horizontal clipping size) must be specified in 4-pixel units.
	The number of vertical output lines (HD) (= vertical clipping size) must be specified in 4-line (HD) units.
	In data enable fetch bundle write, the output address must be specified in 32-byte units.
Internal processing	The filter clipping size must be specified as a value equal to or lower than the actual output size of the filter.

## 46.5.3 Cooperation with Video Display Controller 5

Since the input data are written to the memory separately as Y data and CbCr data in image capture mode, the captured data cannot be displayed in the video display controller 5.

## 47. Pixel Format Converter

The pixel format converter has a color matrix that is used to adjust offsets and nine-axis gain for image data in the YCbCr or RGB format. This enables brightness adjustment, gain adjustment, and YCbCr and RGB mutual conversion. This LSI incorporates two channels of pixel format converters.

### 47.1 Features

- Input pixel data  
[RGB] RGB888 and RGB565  
[YCbCr] YCbCr422
- Output pixel data  
[RGB] ARGB888 and RGB565  
[YCbCr] YCbCr422
- Input buffer: 32-byte FIFO buffer configuration
- Output buffer: 32-byte FIFO buffer configuration
- Interrupt sources: Three  
Input FIFO empty, output FIFO full, and PFV error (input FIFO overflow and output FIFO underflow)
- The input FIFO empty and output FIFO full interrupts can activate the direct memory access controller (DMAC) to transfer data.
- When this module is not used, it can be stopped to reduce power consumption by stopping the clock supply to this module.

Figure 47.1 shows a block diagram of this module.

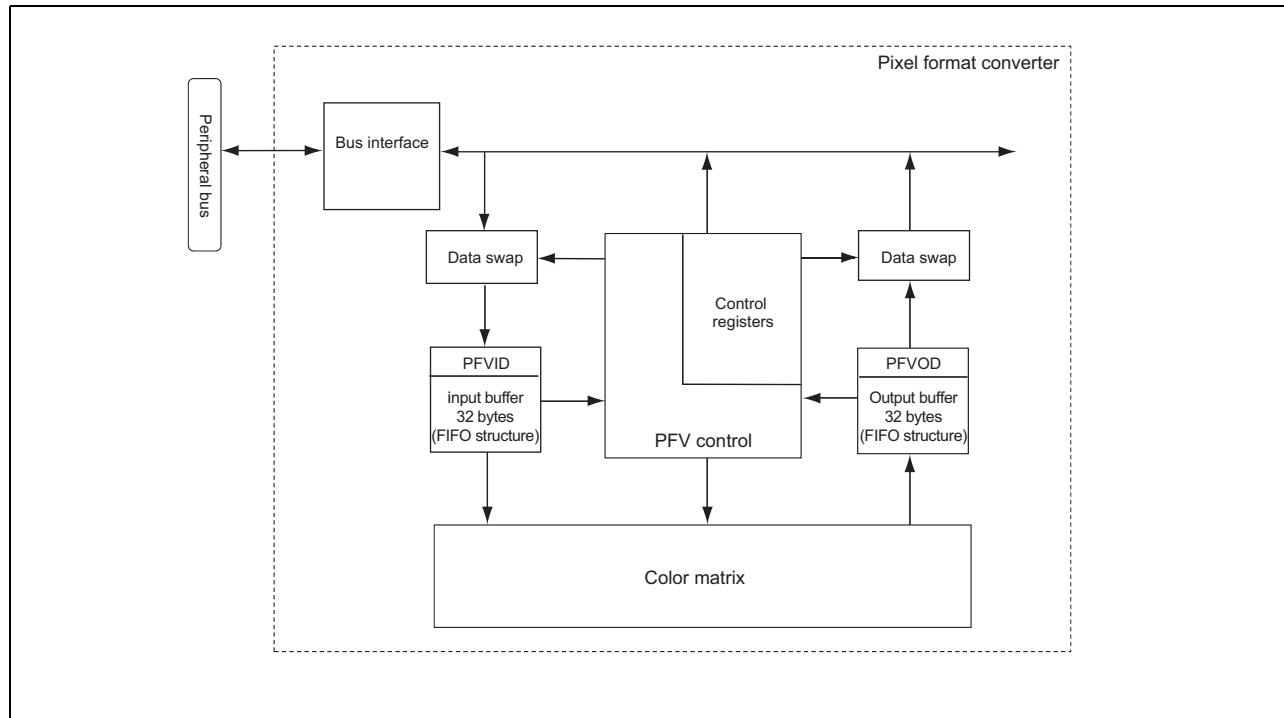


Figure 47.1 Block Diagram

## 47.2 Register Descriptions

This module has the following registers.

**Table 47.1 Register Configuration**

Channel	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
0	PFV control register	PFVCR	R/W	H'0000_0000	H'E820_5000	32
	PFV interrupt control register	PFVICR	R/W	H'0000_0000	H'E820_5004	32
	PFV interrupt status register	PFVISR	R	H'0000_0000	H'E820_5008	32
	PFV input buffer register	PFVID	W	Undefined	H'E820_5020 to H'E820_503C	16*, 32
	PFV output buffer register	PFVOD	R	Undefined	H'E820_5040 to H'E840_505C	16*, 32
	PFV input FIFO status register	PFVIFSR	R/W	H'0000_0000	H'E820_5064	32
	PFV output FIFO status register	PFVOFSR	R/W	H'0000_0000	H'E820_5068	32
	PFV $\alpha$ setting register	PFVACR	R/W	H'0000_0000	H'E820_506C	32
	PFV matrix mode register	PFV_MTX_MODE	R/W	H'0000_0000	H'E820_5070	32
	PFV matrix YG adjustment register 0	PFV_MTX_YG_ADJ0	R/W	H'0080_0100	H'E820_5074	32
	PFV matrix YG adjustment register 1	PFV_MTX_YG_ADJ1	R/W	H'0000_0000	H'E820_5078	32
	PFV matrix CBB adjustment register 0	PFV_MTX_CBB_ADJ0	R/W	H'0080_0000	H'E820_507C	32
	PFV matrix CBB adjustment register 1	PFV_MTX_CBB_ADJ1	R/W	H'0100_0000	H'E820_5080	32
	PFV matrix CRR adjustment register 0	PFV_MTX_CRR_ADJ0	R/W	H'0080_0000	H'E820_5084	32
	PFV matrix CRR adjustment register 1	PFV_MTX_CRR_ADJ1	R/W	H'0000_0100	H'E820_5088	32
	PFV image size setting register	PFVSZR	R/W	H'0000_0000	H'E820_508C	32
1	PFV control register	PFVCR	R/W	H'0000_0000	H'E820_5800	32
	PFV interrupt control register	PFVICR	R/W	H'0000_0000	H'E820_5804	32
	PFV interrupt status register	PFVISR	R	H'0000_0000	H'E820_5808	32
	PFV input buffer register	PFVID	W	Undefined	H'E820_5820 to H'E820_583C	16*, 32
	PFV output buffer register	PFVOD	R	Undefined	H'E820_5840 to H'E840_585C	16*, 32
	PFV input FIFO status register	PFVIFSR	R/W	H'0000_0000	H'E820_5864	32
	PFV output FIFO status register	PFVOFSR	R/W	H'0000_0000	H'E820_5868	32
	PFV $\alpha$ setting register	PFVACR	R/W	H'0000_0000	H'E820_586C	32
	PFV matrix mode register	PFV_MTX_MODE	R/W	H'0000_0000	H'E820_5870	32
	PFV matrix YG adjustment register 0	PFV_MTX_YG_ADJ0	R/W	H'0080_0100	H'E820_5874	32
	PFV matrix YG adjustment register 1	PFV_MTX_YG_ADJ1	R/W	H'0000_0000	H'E820_5878	32
	PFV matrix CBB adjustment register 0	PFV_MTX_CBB_ADJ0	R/W	H'0080_0000	H'E820_587C	32
	PFV matrix CBB adjustment register 1	PFV_MTX_CBB_ADJ1	R/W	H'0100_0000	H'E820_5880	32
	PFV matrix CRR adjustment register 0	PFV_MTX_CRR_ADJ0	R/W	H'0080_0000	H'E820_5884	32
	PFV matrix CRR adjustment register 1	PFV_MTX_CRR_ADJ1	R/W	H'0000_0100	H'E820_5888	32
	PFV image size setting register	PFVSZR	R/W	H'0000_0000	H'E820_588C	32

Note: \* Only address 0 should be accessed when the pixel format is set to RGB565.

### 47.2.1 PFV Control Register (PFVCR)

This register enables operation of this module and makes various settings for input/output data.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PFVE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	DTH_ON	IFMT[1:0]	OFMT[1:0]	—	—	—	—	—	DIN SWAP32[1:0]	DOUT SWAP32[1:0]	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	PFVE	0	R/W	PFV Enable Setting this bit to 1 starts operation of this module. When this bit is cleared to 0 during operation, this module stops operation and the valid data in the input and output FIFOs and the operating blocks in this module are initialized. 0: This module is stopped. 1: This module is ready for operation.
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	DTH_ON	0	R/W	RGB565 Bit Reduction Setting Specifies bit reduction from RGB888 to RGB565 for output data. 0: Round-off mode 1: 2 x 2 pattern dither (the PFVSRZ register should also be set up)
11, 10	IFMT[1:0]	00	R/W	Pixel Format Setting for Input Data Specifies the pixel format for input data. 00: RGB888 01: RGB565 10: Setting prohibited 11: YCbCr422
9, 8	OFMT[1:0]	00	R/W	Pixel Format Setting for Output Data Specifies the pixel format for output data. 00: ARGB8888 01: RGB565 10: Setting prohibited 11: YCbCr422
7 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3, 2	DIN SWAP32 [1:0]	00	R/W	Swap Setting for Input Data Specifies the swap operation for input data. This setting is valid only for 32-bit access. 00: (1) (2) (3) (4) 01: (2) (1) (4) (3) [Byte swap] 10: (3) (4) (1) (2) [Word swap] 11: (4) (3) (2) (1) [Word-byte swap]
1, 0	DOUT SWAP32 [1:0]	00	R/W	Swap Setting for Output Data Specifies the swap operation for output data. This setting is valid only for 32-bit access. 00: (1) (2) (3) (4) 01: (2) (1) (4) (3) [Byte swap] 10: (3) (4) (1) (2) [Word swap] 11: (4) (3) (2) (1) [Word-byte swap]

## 47.2.2 PFV Interrupt Control Register (PFVICR)

This register enables various interrupt requests.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	PFV EEN	IFEN	OFEN	IDTRG[1:0]	ODTRG[1:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 7	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
6	PFVEEN	0	R/W	PFV Error Interrupt Enable Enables or disables the PFV error interrupt request (PFVEI) to be issued when the input FIFO overflow interrupt source flag or output FIFO underflow interrupt source flag is set to 1. 0: PFV error interrupt is disabled. 1: PFV error interrupt is enabled
5	IFEN	0	R/W	Input FIFO Empty Interrupt Enable Enables or disables the input FIFO empty interrupt request (IFEI) to be issued when the input FIFO empty interrupt source flag is set to 1. 0: Input FIFO empty interrupt is disabled. 1: Input FIFO empty interrupt is enabled
4	OFEN	0	R/W	Output FIFO Full Interrupt Enable Enables or disables the output FIFO full interrupt request (OFFI) to be issued when the output FIFO full interrupt source flag is set to 1. 0: Output FIFO full interrupt is disabled. 1: Output FIFO full interrupt is enabled
3, 2	IDTRG[1:0]	All 0	R/W	Input FIFO Byte Count Trigger These bits specify the number of bytes in the input FIFO to be used as the trigger for setting the input FIFO empty interrupt source flag to 1. When the amount of valid data in the input FIFO becomes no larger than the specified trigger count, the input FIFO empty interrupt source flag is set to 1. 00: 30 bytes (2-byte empty space in the input FIFO) 01: 28 bytes (4-byte empty space in the input FIFO) 10: 16 bytes (16-byte empty space in the input FIFO) 11: 0 bytes (32-byte empty space in the input FIFO)
1, 0	ODTRG[1:0]	All 0	R/W	Output FIFO Byte Count Trigger These bits specify the number of bytes in the output FIFO to be used as the trigger for setting the output FIFO full interrupt source flag to 1. When the amount of valid data in the output FIFO becomes no less than the specified trigger count, the output FIFO full interrupt source flag is set to 1. 00: 2 bytes (30-byte empty space in the output FIFO) 01: 4 bytes (28-byte empty space in the output FIFO) 10: 16 bytes (16-byte empty space in the output FIFO) 11: 32 bytes (0-byte empty space in the output FIFO)

### 47.2.3 PFV Interrupt Status Register (PFVISR)

This register is used to check various interrupt source flags.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	IFOVF	OFUDF	IFEMP	OFFUL
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R

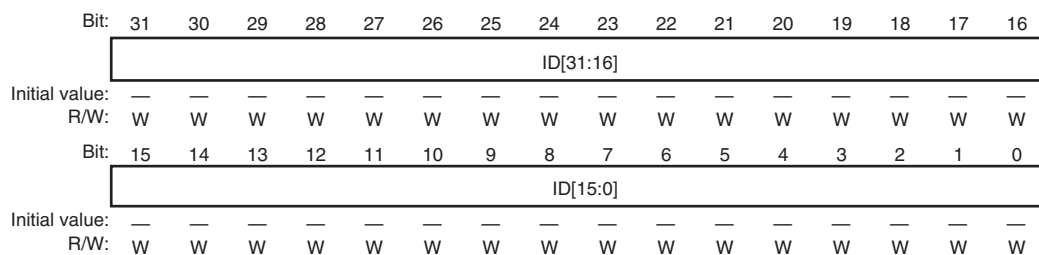
Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	IFOVF	0	R/W	Input FIFO Overflow Interrupt Source Flag Indicates that more data than the amount of empty space has been written to the input FIFO. 0: Indicates that no data has been written to PFVID while the input FIFO does not have enough empty space. [Clearing conditions] <ul style="list-style-type: none"> <li>• Power-on reset</li> <li>• When PFVE = 0</li> <li>• When IFOVF = 1 is read and then 0 is written to IFOVF</li> </ul> 1: Indicates that data has been written to PFVID while the input FIFO does not have enough empty space. [Setting condition] <ul style="list-style-type: none"> <li>• When data is written to PFVID while the input FIFO does not have enough empty space.</li> </ul>
2	OFUDF	0	R/W	Output FIFO Underflow Interrupt Source Flag Indicates that more data than the amount of valid data has been read from the output FIFO. 0: Indicates that no data has been read from PFVOD while the output FIFO does not have enough valid data. [Clearing conditions] <ul style="list-style-type: none"> <li>• Power-on reset</li> <li>• When PFVE = 0</li> <li>• When OFUDF = 1 is read and then 0 is written to OFUDF</li> </ul> 1: Indicates that data has been read from PFVOD while the output FIFO does not have enough valid data. [Setting condition] <ul style="list-style-type: none"> <li>• When data is read from PFVOD while the output FIFO does not have enough valid data.</li> </ul>
1	IFEMP	1	R	Input FIFO Empty Interrupt Source Flag Indicates that the amount of valid data in the input FIFO has become equal to or less than the input FIFO trigger count specified in the IDTRG[1:0] bits. 0: Indicates that the amount of valid data in the input FIFO is larger than the specified input FIFO trigger count. [Clearing conditions] <ul style="list-style-type: none"> <li>• When data is written to the input FIFO and the amount of valid data exceeds the specified input FIFO trigger count.</li> </ul> 1: Indicates that the amount of valid data in the input FIFO is no larger than the specified input FIFO trigger count. [Setting conditions] <ul style="list-style-type: none"> <li>• Power-on reset</li> <li>• When PFVE = 0</li> <li>• When the amount of valid data in the input FIFO becomes less than the specified input FIFO trigger count.</li> </ul>



Bit	Bit Name	Initial Value	R/W	Description
0	OFFUL	0	R	<p>Output FIFO Full Interrupt Source Flag</p> <p>Indicates that the amount of valid data in the output FIFO has become equal to or larger than the output FIFO trigger count specified in the ODTRG[1:0] bits.</p> <p>0: Indicates that the amount of valid data in the output FIFO is less than the specified output FIFO trigger count.</p> <p>[Clearing conditions]</p> <ul style="list-style-type: none"> <li>• Power-on reset</li> <li>• When PFVE = 0</li> <li>• When data is read from the output FIFO until the remaining data becomes less than the specified output FIFO trigger count.</li> </ul> <p>1: Indicates that the amount of valid data in the output FIFO is no less than the specified output FIFO trigger count.</p> <p>[Setting condition]</p> <ul style="list-style-type: none"> <li>• When the amount of valid data in the output FIFO is larger than the specified output FIFO trigger count.</li> </ul>

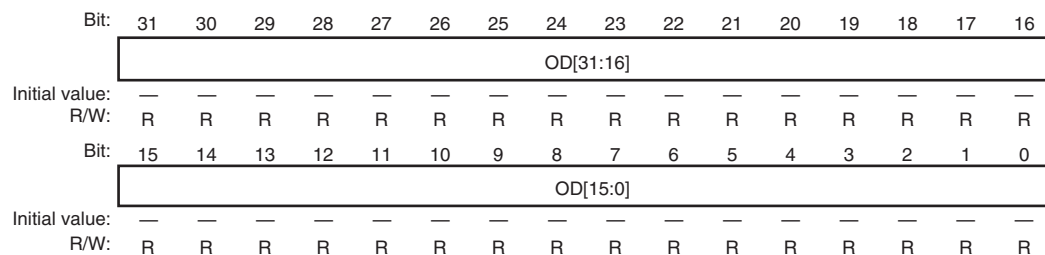
#### 47.2.4 PFV Input Buffer Register (PFVID)

PFVID is 32-byte buffer memory that stores pixel data to be converted. The buffer memory has a FIFO structure (FIFO buffer). If an attempt is made to write data to the input buffer while it does not have empty space, the input FIFO overflow interrupt flag is set to 1; the written data is ignored in this case.



#### 47.2.5 PFV Output Buffer Register (PFVOD)

PFVOD is 32-byte buffer memory that stores the converted pixel data. The buffer memory has a FIFO structure (FIFO buffer). If an attempt is made to read data from the output buffer while it does not have valid data, the output FIFO underflow interrupt flag is set to 1; the read data is undefined in this case.



### 47.2.6 PFV Input FIFO Status Register (PFVIFSR)

This register is used to check the number of valid bytes in the input FIFO.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	IFVD[5:0]					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5 to 0	IFVD[5:0]	All 0	R	Valid Byte Count in Input FIFO These bits indicate the number of valid bytes stored in the input FIFO. H'00 indicates that no input data is stored, and H'20 indicates that the input FIFO is full of data.

### 47.2.7 PFV Output FIFO Status Register (PFVOFSR)

This register is used to check the number of valid bytes in the output FIFO.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	OFVD[5:0]					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5 to 0	OFVD[5:0]	All 0	R	Valid Byte Count in Output FIFO These bits indicate the number of valid bytes stored in the output FIFO. H'00 indicates that no output data is stored, and H'20 indicates that the output FIFO is full of data.

### 47.2.8 PFV $\alpha$ Setting Register (PFVACR)

This register specifies the A value used for output in the ARGB8888 format.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W (compress):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
R/W (decompress):	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ALPHA[7:0]							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W (compress):	R	R	R	R	R	R	R	R	Ignored	Ignored	Ignored	Ignored	Ignored	Ignored	Ignored	Ignored
R/W (decompress):	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W		Description
			Compression	De-compression	
31 to 8	—	All 0		R	Reserved These bits are always read as 0. The write value should always be 0.
7 to 0	ALPHA[7:0]	H'00	Ignored	R/W	These bits specify the A value used for output in the ARGB8888 format.

### 47.2.9 PFV Matrix Mode Register (PFV\_MTX\_MODE)

This register specifies the operation of the color matrix.

Make appropriate settings in this register according to the pixel format of the input/output data.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PFV_MTX_MD[1:0]
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	PFV_MTX_MD [1:0]	00	R/W	Operating Mode These bits specify the operation of the color matrix. Make appropriate settings according to the pixel format (IFMT and OFMT bit settings in PFVCR) of the input/output data. 00: GBR → GBR 01: GBR → YCbCr 10: YCbCr → GBR 11: YCbCr → YCbCr

### 47.2.10 PFV Matrix YG Adjustment Register 0 (PFV\_MTX\_YG\_ADJ0)

This register specifies offset and gain adjustment for output Y/G data.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	PFV_MTX_YG[7:0]							
Initial value:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	PFV_MTX_GG[10:0]										
Initial value:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	PFV_MTX_YG [7:0]	128	R/W	Offset (DC) Adjustment for Output Y/G Data Unsigned (0 (-128) to 128 (0) to 255 (+127) [LSB])
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	PFV_MTX_GG [10:0]	256	R/W	Input Y/G Data Gain Adjustment for Output Y/G Data Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

### 47.2.11 PFV Matrix YG Adjustment Register 1 (PFV\_MTX\_YG\_ADJ1)

This register specifies offset and gain adjustment for output Y/G data.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	PFV_MTX_GB[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	PFV_MTX_GR[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	PFV_MTX_GB [10:0]	0	R/W	Input Cb/B Data Gain Adjustment for Output Y/G Data Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	PFV_MTX_GR [10:0]	0	R/W	Input Cr/R Data Gain Adjustment for Output Y/G Data Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

### 47.2.12 PFV Matrix CBB Adjustment Register 0 (PFV\_MTX\_CBB\_ADJ0)

This register specifies offset and gain adjustment for output Cb/B data.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	PFV_MTX_B[7:0]							
Initial value:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	PFV_MTX_BG[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	PFV_MTX_B [7:0]	128	R/W	Offset (DC) Adjustment for Output Cb/B Data Unsigned (0 (-128) to 128 (0) to 255 (+127) [LSB])
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	PFV_MTX_BG [10:0]	0	R/W	Input Y/G Data Gain Adjustment for Output Cb/B Data Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

### 47.2.13 PFV Matrix CBB Adjustment Register 1 (PFV\_MTX\_CBB\_ADJ1)

This register specifies offset and gain adjustment for output Cb/B data.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	PFV_MTX_BB[10:0]										
Initial value:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	PFV_MTX_BR[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	PFV_MTX_BB [10:0]	256	R/W	Input Cb/B Data Gain Adjustment for Output Cb/B Data Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	PFV_MTX_BR [10:0]	0	R/W	Input Cr/R Data Gain Adjustment for Output Cb/B Data Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

### 47.2.14 PFV Matrix CRR Adjustment Register 0 (PFV\_MTX\_CRR\_ADJ0)

This register specifies offset and gain adjustment for output Cr/R data.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	PFV_MTX_R[7:0]							
Initial value:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	PFV_MTX_RG[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	PFV_MTX_R [7:0]	128	R/W	Offset (DC) Adjustment for Output Cr/R Data Unsigned (0 (-128) to 128 (0) to 255 (+127) [LSB])
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	PFV_MTX_RG [10:0]	0	R/W	Input Y/G Data Gain Adjustment for Output Cr/R Data Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

### 47.2.15 PFV Matrix CRR Adjustment Register 1 (PFV\_MTX\_CRR\_ADJ1)

This register specifies offset and gain adjustment for output Cr/R data.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	PFV_MTX_RB[10:0]										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	PFV_MTX_RR[10:0]										
Initial value:	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	PFV_MTX_RB [10:0]	0	R/W	Input Cb/B Data Gain Adjustment for Output Cr/R Data Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])
15 to 11	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
10 to 0	PFV_MTX_RR [10:0]	256	R/W	Input Cr/R Data Gain Adjustment for Output Cr/R Data Signed (two's complement) (-1024 to +1023 [LSB], 256 [LSB] = 1.0 [times])

### 47.2.16 PFV Image Size Setting Register (PFVSZR)

This register specifies the horizontal and vertical sizes of an image.

Be sure to set up this register when the output pixel format is RGB565 and  $2 \times 2$  pattern dither is used for bit reduction.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PFVSZX[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PFVSZY[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	PFVSZX[15:0]	H'0000	R/W	Horizontal Size These bits specify the horizontal size.
15 to 0	PFVSZY[15:0]	H'0000	R/W	Vertical Size These bits specify the vertical size.

## 47.3 Operation

### 47.3.1 Overview

This module uses a color matrix for offset and nine-axis gain adjustment to apply brightness adjustment, gain adjustment, and YCbCr and RGB mutual conversion for image data in the YCbCr or RGB format.

### 47.3.2 Input/Output Data Format

This module allows selection of input and output data formats from RGB888 (ARGB8888 for output), RGB565, and YCbCr422. When the input and output FIFOs are accessed in 32 bits, data allocation can be modified through the DINSWAP32 and DOUTSWAP32 bit settings in PFVCR as shown below.

#### (1) RGB888 (ARGB8888) 32 bits/pixel

[DINSWAP32 and DOUTSWAP32 = 00 is specified]

b31	b24 b23	b16 b15	b8 b7	b0
Alpha* 8 bits	Red 8 bits	Green 8 bits	Blue 8 bits	

[DINSWAP32 and DOUTSWAP32 = 01 is specified]

b31	b24 b23	b16 b15	b8 b7	b0
Red 8 bits	Alpha* 8 bits	Blue 8 bits	Green 8 bits	

[DINSWAP32 and DOUTSWAP32 = 10 is specified]

b31	b24 b23	b16 b15	b8 b7	b0
Green 8 bits	Blue 8 bits	Alpha* 8 bits	Red 8 bits	

[DINSWAP32 and DOUTSWAP32 = 11 is specified]

b31	b24 b23	b16 b15	b8 b7	b0
Blue 8 bits	Green 8 bits	Red 8 bits	Alpha* 8 bits	

Note: \* Alpha is used only for output and it is the value specified in the ALPHA bits in PFVACR.

#### (2) RGB565 16 bits/pixel

[DINSWAP32 and DOUTSWAP32 = 00 is specified]

b31	b24 b23	b16 b15	b8 b7	b0
RGB565 0 pixels	RGB565 1 pixel			

[DINSWAP32 and DOUTSWAP32 = 01 is prohibited]

[DINSWAP32 and DOUTSWAP32 = 10 is specified]

b31	b24 b23	b16 b15	b8 b7	b0
RGB565 1 pixel	RGB565 0 pixels			

[DINSWAP32 and DOUTSWAP32 = 11 is prohibited]



**(3) YCbCr422 32bits/pixel**

[DINSWAP32 and DOUTSWAP32 = 00 is specified]

b31	b24 b23	b16 b15	b8 b7	b0
Cb 8 bits	Y0 8 bits	Cr 8 bits	Y1 8 bits	

[DINSWAP32 and DOUTSWAP32 = 01 is specified]

b31	b24 b23	b16 b15	b8 b7	b0
Y0 8 bits	Cb 8 bits	Y1 8 bits	Cr 8 bits	

[DINSWAP32 and DOUTSWAP32 = 10 is specified]

b31	b24 b23	b16 b15	b8 b7	b0
Cr 8 bits	Y1 8 bits	Cb 8 bits	Y0 8 bits	

[DINSWAP32 and DOUTSWAP32 = 11 is specified]

b31	b24 b23	b16 b15	b8 b7	b0
Y1 8 bits	Cr 8 bits	Y0 8 bits	Cb 8 bits	

**47.3.3 Color Matrix**

By using a color matrix, input data offsets and nine-axis gain can be adjusted. This enables brightness adjustment, gain adjustment, and YCbCr and GBR mutual conversion.

**(1) GBR to GBR Conversion**

$$YGIN\_A = YGIN + PFV\_MTX\_YG - 128$$

$$CBBIN\_A = CBBIN + PFV\_MTX\_B - 128$$

$$CRRIN\_A = CRRIN + PFV\_MTX\_R - 128$$

$$YGOUT = (PFV\_MTX\_GG \times YGIN\_A + PFV\_MTX\_GB \times CBBIN\_A + PFV\_MTX\_GR \times CRRIN\_A) \div 256$$

$$CBBOUT = (PFV\_MTX\_BG \times YGIN\_A + PFV\_MTX\_BB \times CBBIN\_A + PFV\_MTX\_BR \times CRRIN\_A) \div 256$$

$$CRROUT = (PFV\_MTX\_RG \times YGIN\_A + PFV\_MTX\_RB \times CBBIN\_A + PFV\_MTX\_RR \times CRRIN\_A) \div 256$$

**(2) GBR to YCbCr Conversion**

$$YGIN\_A = YGIN + PFV\_MTX\_YG - 128$$

$$CBBIN\_A = CBBIN + PFV\_MTX\_B - 128$$

$$CRRIN\_A = CRRIN + PFV\_MTX\_R - 128$$

$$YGOUT = (PFV\_MTX\_GG \times YGIN\_A + PFV\_MTX\_GB \times CBBIN\_A + PFV\_MTX\_GR \times CRRIN\_A) \div 256$$

$$CBBOUT = (PFV\_MTX\_BG \times YGIN\_A + PFV\_MTX\_BB \times CBBIN\_A + PFV\_MTX\_BR \times CRRIN\_A) \div 256 + 128$$

$$CRROUT = (PFV\_MTX\_RG \times YGIN\_A + PFV\_MTX\_RB \times CBBIN\_A + PFV\_MTX\_RR \times CRRIN\_A) \div 256 + 128$$

**Table 47.2 Matrix Coefficient (Typical Value) for SMPTE 293M**

	YGIN		CBBIN		CRRIN	
	Coefficient	Set Value	Coefficient	Set Value	Coefficient	Set Value
YGOUT	0.587	PFV_MTX_GG= 150	0.114	PFV_MTX_GB= 29	0.299	PFV_MTX_GR= 77
CBBOUT	-0.331	PFV_MTX_BG= 1963	0.500	PFV_MTX_BB= 128	-0.169	PFV_MTX_BR= 2005
CRROUT	-0.419	PFV_MTX_RG= 1941	-0.081	PFV_MTX_RB= 2027	0.500	PFV_MTX_RR= 128

**(3) YCbCr to GBR Conversion**

$$YGIN\_A = YGIN + PFV\_MTX\_YG - 128$$

$$CBBIN\_A = CBBIN - 128$$

$$CRRIN\_A = CRRIN - 128$$

$$YGOUT = (PFV\_MTX\_GG \times YGIN\_A + PFV\_MTX\_GB \times CBBIN\_A + PFV\_MTX\_GR \times CRRIN\_A) \div 256$$

$$CBBOUT = (PFV\_MTX\_BG \times YGIN\_A + PFV\_MTX\_BB \times CBBIN\_A + PFV\_MTX\_BR \times CRRIN\_A) \div 256$$

$$CRROUT = (PFV\_MTX\_RG \times YGIN\_A + PFV\_MTX\_RB \times CBBIN\_A + PFV\_MTX\_RR \times CRRIN\_A) \div 256$$

**Table 47.3 Matrix Coefficient (Typical Value) for SMPTE 293M**

	YGIN		CBBIN		CRRIN	
	Coefficient	Set Value	Coefficient	Set Value	Coefficient	Set Value
YGOUT	1.000	PFV_MTX_GG= 256	-0.344	PFV_MTX_GB= 1960	-0.714	PFV_MTX_GR= 1865
CBBOUT	1.000	PFV_MTX_BG= 256	1.772	PFV_MTX_BB= 454	0.000	PFV_MTX_BR= 0
CRROUT	1.000	PFV_MTX_RG= 256	0.000	PFV_MTX_RB= 0	1.402	PFV_MTX_RR= 359

**(4) YCbCr to YCbCr Conversion**

$$YGIN\_A = YGIN + PFV\_MTX\_YG - 128$$

$$CBBIN\_A = CBBIN - 128$$

$$CRRIN\_A = CRRIN - 128$$

$$YGOUT = (PFV\_MTX\_GG \times YGIN\_A + PFV\_MTX\_GB \times CBBIN\_A + PFV\_MTX\_GR \times CRRIN\_A) \div 256$$

$$CBBOUT = (PFV\_MTX\_BG \times YGIN\_A + PFV\_MTX\_BB \times CBBIN\_A + PFV\_MTX\_BR \times CRRIN\_A) \div 256 + 128$$

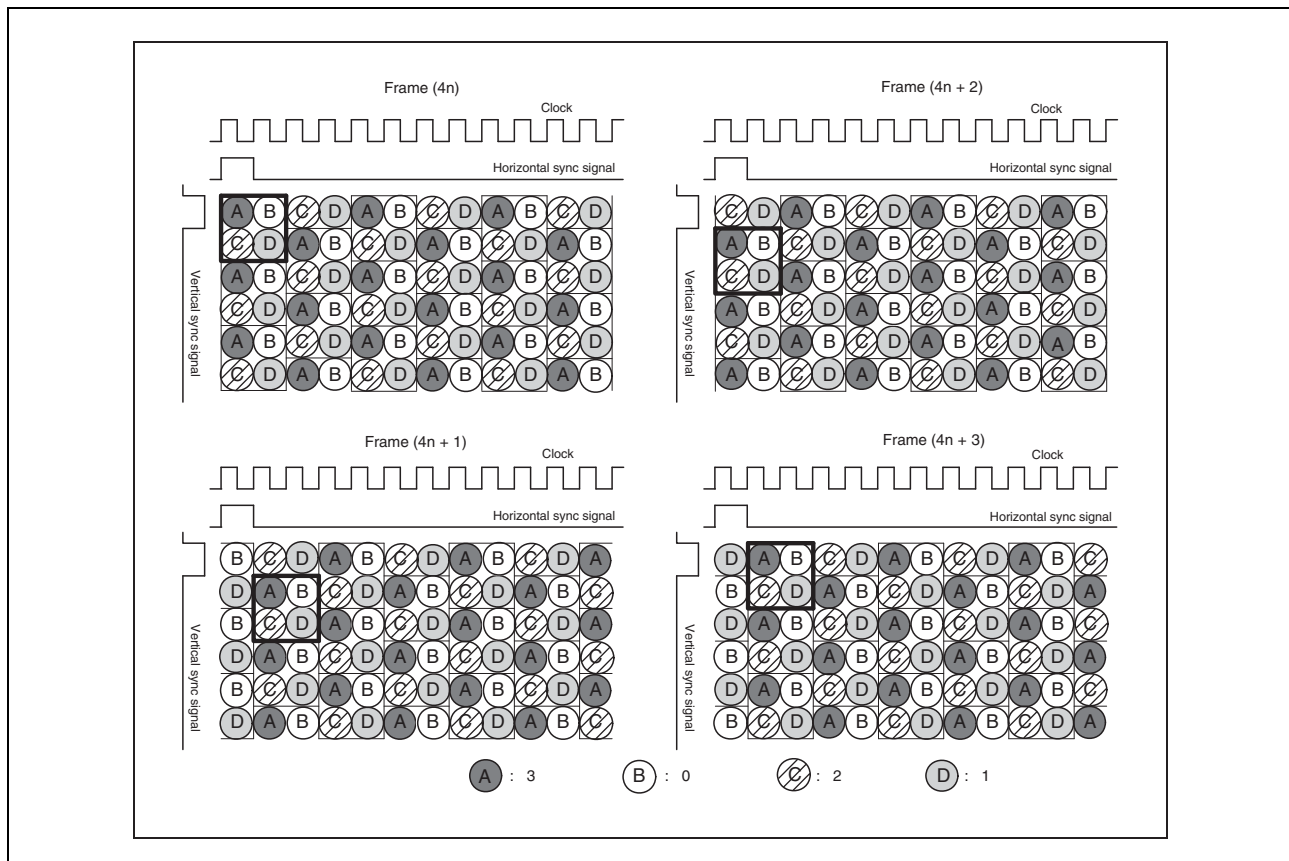
$$CRROUT = (PFV\_MTX\_RG \times YGIN\_A + PFV\_MTX\_RB \times CBBIN\_A + PFV\_MTX\_RR \times CRRIN\_A) \div 256 + 128$$

### 47.3.4 Bit Reduction

For RGB565 output, the number of RGB data bits is reduced from eight bits to six or five bits. The operation mode can be selected from round-off mode and  $2 \times 2$  pattern dither mode through the DTH\_ON bit in PFVCR. When selecting  $2 \times 2$  pattern dither mode, be sure to set up the PFVSZX and PFVSZY bits in PFVSZR.

From the values set in the PFVSZX and PFVSZY bits in PFVSZR, this module generates horizontal and vertical sync signals.

The following shows the specifications of  $2 \times 2$  pattern dither operation.



**Figure 47.2 Specifications of  $2 \times 2$  Pattern Dither Operation**

The conversion equations are as follows.

- Round-off mode

1. 8 bits to 6 bits

$$\text{Output RGB data}[7:2] = \text{Input RGB data}[7:0] \div 4 \text{ (round off to an integer)}$$

2. 8 bits to 5 bits

$$\text{Output RGB data}[7:3] = \text{Input RGB data}[7:0] \div 8 \text{ (round off to an integer)}$$

- $2 \times 2$  pattern dither

1. 8 bits to 6 bits

$$\text{Output RGB data}[7:2] = \text{Input RGB data}[7:0] \div 4 + \text{pattern value at the first decimal place (truncate the number below the decimal point after addition)}$$

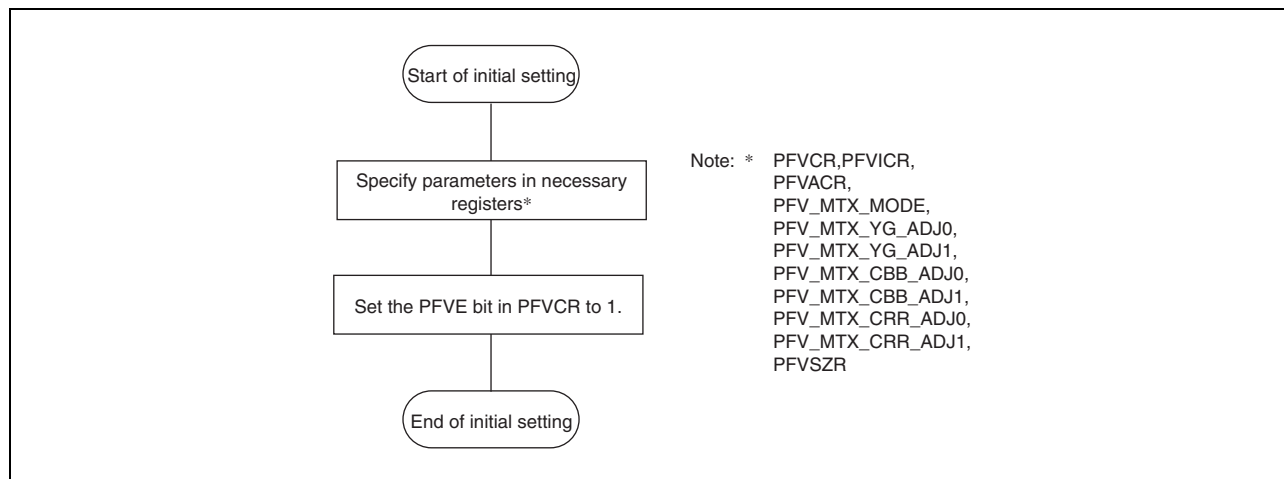
2. 8 bits to 5 bits

$$\text{Output RGB data}[7:3] = \text{Input RGB data}[7:0] \div 32 + \text{pattern value at the first decimal place (truncate the number below the decimal point after addition)}$$

### 47.3.5 Operating Procedures

#### (1) Initial Setting

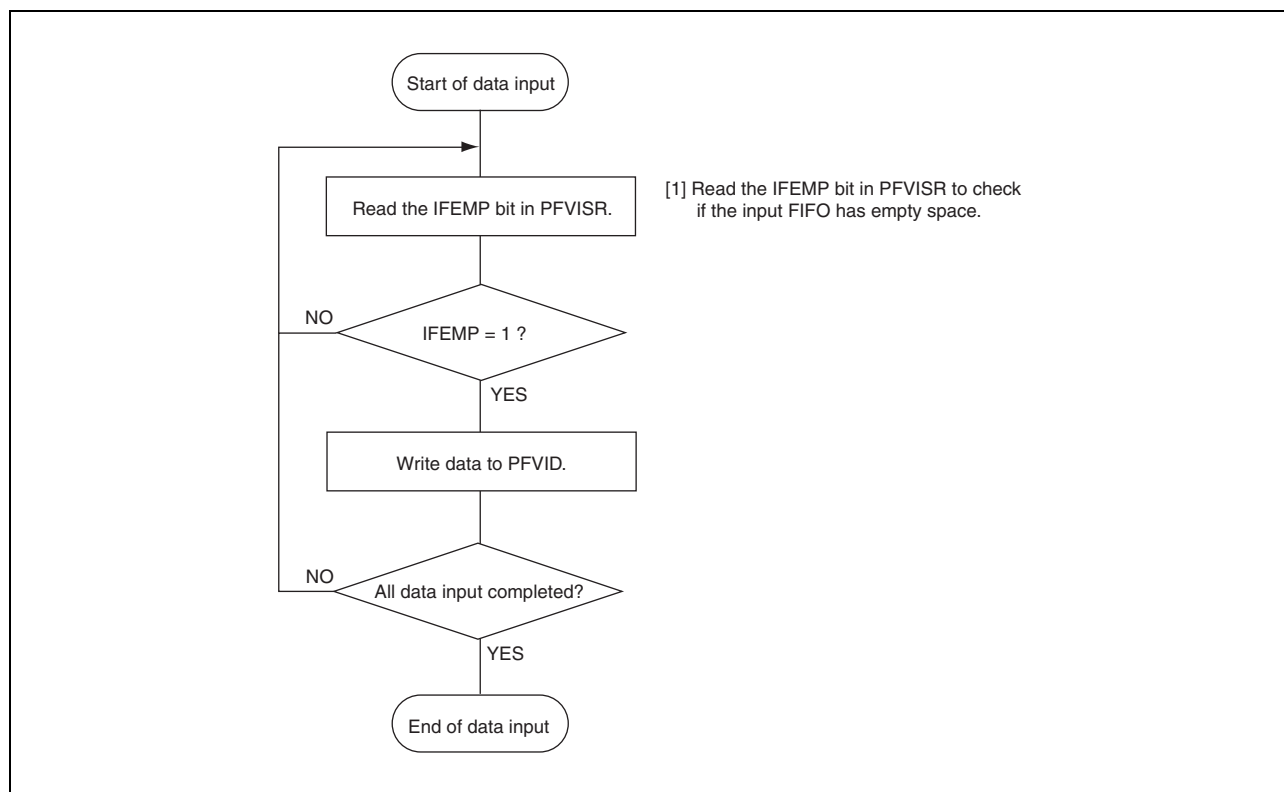
Figure 47.3 is a sample flowchart for initial setting.



**Figure 47.3 Sample Flowchart for Initial Setting**

#### (2) Data Input

Figure 47.4 is a sample flowchart for data input.



**Figure 47.4 Sample Flowchart for Data Input**

(3) Data Output

Figure 47.5 is a sample flowchart for data output.

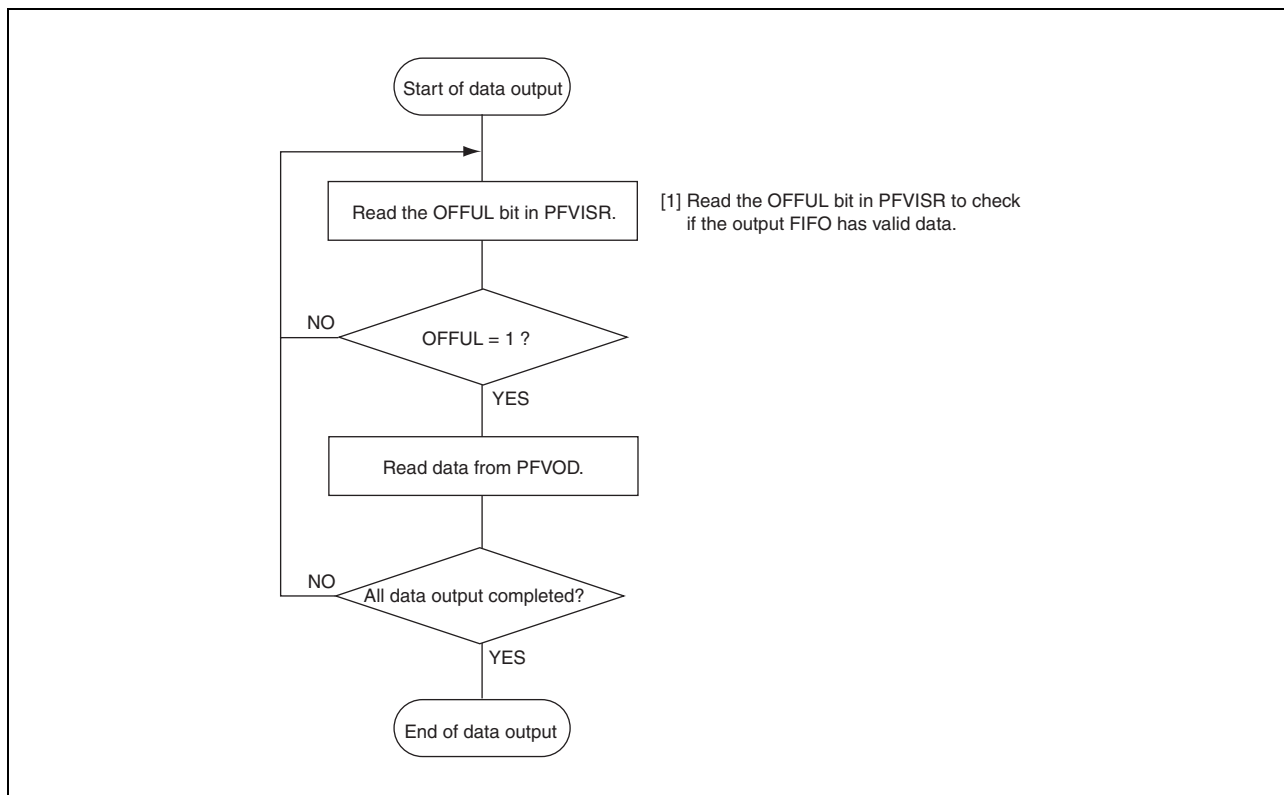


Figure 47.5 Sample Flowchart for Data Output

## 47.4 Interrupt Requests

This module has three interrupt requests: input FIFO empty, output FIFO full, and PFV error. Table 47.4 summarizes the interrupt requests.

**Table 47.4 List of Interrupt Requests**

Interrupt Request	Abbreviation	Interrupt Request Condition	DMAC Activation
Input FIFO empty	IFEI	(PFVISR.IFEMP = 1) & (PFVICR.IFEN = 1)	Possible
Output FIFO full	OFFI	(PFVISR.OFFUL = 1) & (PFVICR.OFEN = 1)	Possible
PFV error	PFVEI	((PFVISR.IFOVF=1)   (PFVISR.OFUDF=1)) & (PFVICR.PFVEEN=1)	Not possible

When an interrupt condition shown in Table 47.4 becomes 1, this module sends an interrupt request. The input FIFO empty and output FIFO full interrupt requests can activate the direct memory access controller when the direct memory access controller is set to allow this. In this case, the interrupts from this module are not sent to the CPU.

## 48. SCUX

The SCUX consists of sampling converters, digital volume units, and a mixer. It is connected to the SSIF module to have an interface with external modules.

### 48.1 Features

#### (1) [SRC] Sampling Rate Conversion

- Asynchronous or synchronous sampling rate conversion is possible\*<sup>1</sup>
- Sampling rate (synchronous mode)\*<sup>2</sup>  
Input [KHz]: 8, 11.025, 12, 16, 22.05, 24, 32, 44.1, 48, 64, 88.2, or 96 is selectable  
Output [KHz]: 8, 16, 24, 32, 44.1, 48, or 96 is selectable
- Sampling rate (asynchronous mode)\*<sup>2</sup>  
Input/output [KHz]: 1 to 96
- Supported bit sizes are 16 bits and 24 bits
- Sound quality: -132 dB or less\*<sup>3</sup>
- 1, 2, 4, 6, or 8 channels are supported\*<sup>4</sup>
- DMA transfer with on-chip memory or external memory and direct transfer with the SSIF module are possible

Note 1. The synchronous mode can be selected only when the SCUX is connected to the FFD and FFU modules. For details, refer to section 48.4.5.

Note 2. The selectable sampling rates depend on the number of used channels and rate ratio. For details, refer to section 48.3.22 and section 48.4.7.

Note 3. The data format is a 24-bit value.

Note 4. The number of selectable channels depends on the sampling rate and route. For details, refer to section 48.4.7.

#### (2) [DVU] Digital Volume and Mute Functions

- Digital volume, volume ramp, and zero cross mute are provided as functions to adjust the volume
- The digital volume is set as a 24-bit fixed-point value within the range from a multiple of 0 to 8 (mute, -120 to 18 dB)
- Volume ramp can be used to perform soft mute, fade in, fade out, and volume change as desired
- The ramp time of volume ramp can be changed and set within the sampling range of  $2^0$  to  $2^{23}$
- Zero cross mute turns the sound mute at the zero cross point of audio data
- Direct transfer with the SSIF module and transfer with the mixer are possible

#### (3) [MIX] Mixer

- Data of two to four source systems can be mixed (added together) into one system
- The ratio to add the sources can be set
- The ratio can be changed dynamically
- Volume ramp enables mixing to be performed (ramp time is variable)
- Only direct transfer with the SSIF module is possible

Figure 48.1 shows a block diagram of the SCUX.

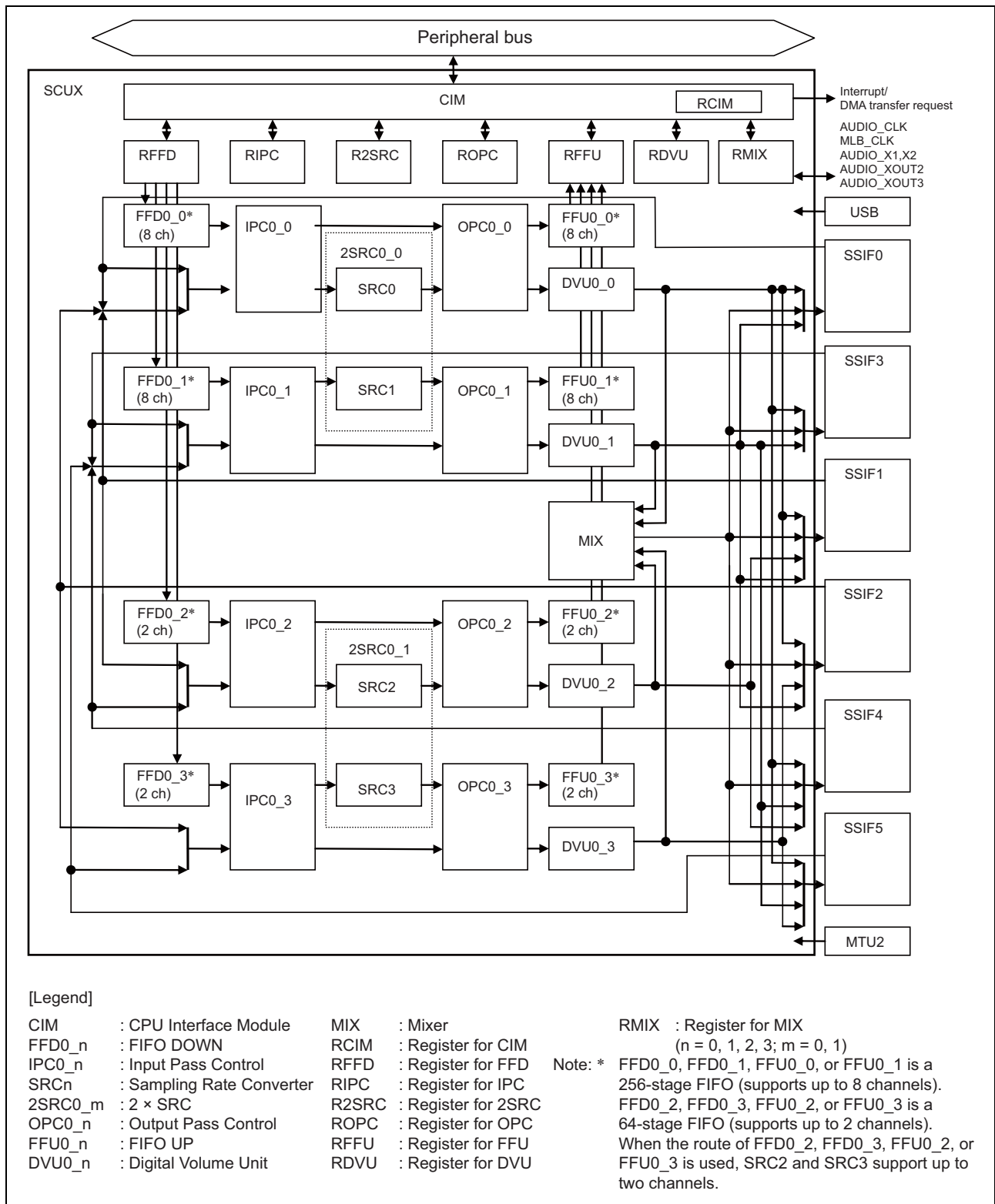


Figure 48.1 Block Diagram of SCUX



## 48.2 Input/Output Pins

**Table 48.1 Pin Configuration**

Name	I/O	Function
AUDIO_CLK	Input	External clock for audio Can be used in generating the sampling timing.
MLB_CLK	Input	External clock for MLB Can be used in generating the sampling timing.
AUDIO_X1	Input	Crystal resonator/external clock for audio Can be used in generating the sampling timing.
AUDIO_X2	Output	
AUDIO_XOUT2	Output	Output clock of AUDIO_X1 divided into two (Output is toggled once per cycle of the AUDIO_X1 clock.)
AUDIO_XOUT3	Output	Output clock of AUDIO_X1 divided into three (Output is toggled once every 1.5 cycles of the AUDIO_X1 clock.)

## 48.3 Register Descriptions

Table 48.2 shows the register configuration.

**Table 48.2 Register Configuration**

Block Name	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
IPC0_0	IPC0_0 initialization register	IPCIR_IPC0_0	R/W	H'00000001	H'E8208000	32
	IPC0_0 pass select register	IPSLR_IPC0_0	R/W	H'00000000	H'E8208004	32
IPC0_1	IPC0_1 initialization register	IPCIR_IPC0_1	R/W	H'00000001	H'E8208100	32
	IPC0_1 pass select register	IPSLR_IPC0_1	R/W	H'00000000	H'E8208104	32
IPC0_2	IPC0_2 initialization register	IPCIR_IPC0_2	R/W	H'00000001	H'E8208200	32
	IPC0_2 pass select register	IPSLR_IPC0_2	R/W	H'00000000	H'E8208204	32
IPC0_3	IPC0_3 initialization register	IPCIR_IPC0_3	R/W	H'00000001	H'E8208300	32
	IPC0_3 pass select register	IPSLR_IPC0_3	R/W	H'00000000	H'E8208304	32
OPC0_0	OPC0_0 initialization register	OPCIR_OPC0_0	R/W	H'00000001	H'E8208400	32
	OPC0_0 pass select register	OPSLR_OPC0_0	R/W	H'00000000	H'E8208404	32
OPC0_1	OPC0_1 initialization register	OPCIR_OPC0_1	R/W	H'00000001	H'E8208500	32
	OPC0_1 pass select register	OPSLR_OPC0_1	R/W	H'00000000	H'E8208504	32
OPC0_2	OPC0_2 initialization register	OPCIR_OPC0_2	R/W	H'00000001	H'E8208600	32
	OPC0_2 pass select register	OPSLR_OPC0_2	R/W	H'00000000	H'E8208604	32
OPC0_3	OPC0_3 initialization register	OPCIR_OPC0_3	R/W	H'00000001	H'E8208700	32
	OPC0_3 pass select register	OPSLR_OPC0_3	R/W	H'00000000	H'E8208704	32
FFD0_0	FFD0_0 FIFO download initialization register	FFDIR_FFD0_0	R/W	H'00000001	H'E8208800	32
	FFD0_0 FIFO download audio information register	FDAIR_FFD0_0	R/W	H'00000000	H'E8208804	32
	FFD0_0 FIFO download request size register	DRQSR_FFD0_0	R/W	H'00000000	H'E8208808	32
	FFD0_0 FIFO download pass register	FFDPR_FFD0_0	R/W	H'00000000	H'E820880C	32
	FFD0_0 FIFO download boot register	FFDBR_FFD0_0	R/W	H'00000000	H'E8208810	32
	FFD0_0 FIFO download event mask register	DEVMR_FFD0_0	R/W	H'00000000	H'E8208814	32
FFD0_1	FFD0_0 FIFO download event clear register	DEVCR_FFD0_0	R/W*1	H'00000000	H'E820881C	32
	FFD0_1 FIFO download initialization register	FFDIR_FFD0_1	R/W	H'00000001	H'E8208900	32
	FFD0_1 FIFO download audio information register	FDAIR_FFD0_1	R/W	H'00000000	H'E8208904	32
	FFD0_1 FIFO download request size register	DRQSR_FFD0_1	R/W	H'00000000	H'E8208908	32
	FFD0_1 FIFO download pass register	FFDPR_FFD0_1	R/W	H'00000000	H'E820890C	32
	FFD0_1 FIFO download boot register	FFDBR_FFD0_1	R/W	H'00000000	H'E8208910	32
FFD0_2	FFD0_1 FIFO download event mask register	DEVMR_FFD0_1	R/W	H'00000000	H'E8208914	32
	FFD0_1 FIFO download event clear register	DEVCR_FFD0_1	R/W*1	H'00000000	H'E820891C	32
	FFD0_2 FIFO download initialization register	FFDIR_FFD0_2	R/W	H'00000001	H'E8208A00	32
	FFD0_2 FIFO download audio information register	FDAIR_FFD0_2	R/W	H'00000000	H'E8208A04	32
	FFD0_2 FIFO download request size register	DRQSR_FFD0_2	R/W	H'00000000	H'E8208A08	32
	FFD0_2 FIFO download pass register	FFDPR_FFD0_2	R/W	H'00000000	H'E8208A0C	32
FFD0_2	FFD0_2 FIFO download boot register	FFDBR_FFD0_2	R/W	H'00000000	H'E8208A10	32
	FFD0_2 FIFO download event mask register	DEVMR_FFD0_2	R/W	H'00000000	H'E8208A14	32
	FFD0_2 FIFO download event clear register	DEVCR_FFD0_2	R/W*1	H'00000000	H'E8208A1C	32

**Table 48.2 Register Configuration**

Block Name	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
FFD0_3	FFD0_3 FIFO download initialization register	FFDIR_FFD0_3	R/W	H'00000001	H'E8208B00	32
	FFD0_3 FIFO download audio information register	FDAIR_FFD0_3	R/W	H'00000000	H'E8208B04	32
	FFD0_3 FIFO download request size register	DRQSR_FFD0_3	R/W	H'00000000	H'E8208B08	32
	FFD0_3 FIFO download pass register	FFDPR_FFD0_3	R/W	H'00000000	H'E8208B0C	32
	FFD0_3 FIFO download boot register	FFDBR_FFD0_3	R/W	H'00000000	H'E8208B10	32
	FFD0_3 FIFO download event mask register	DEVMR_FFD0_3	R/W	H'00000000	H'E8208B14	32
	FFD0_3 FIFO download event clear register	DEVCR_FFD0_3	R/W*1	H'00000000	H'E8208B1C	32
FFU0_0	FFU0_0 FIFO upload initialization register	FFUIR_FFU0_0	R/W	H'00000001	H'E8208C00	32
	FFU0_0 FIFO upload audio information register	FUAIR_FFU0_0	R/W	H'00000000	H'E8208C04	32
	FFU0_0 FIFO upload request size register	URQSR_FFU0_0	R/W	H'00000000	H'E8208C08	32
	FFU0_0 FIFO upload pass register	FFUPR_FFU0_0	R/W	H'00000000	H'E8208C0C	32
	FFU0_0 FIFO upload event mask register	UEVMR_FFU0_0	R/W	H'00000000	H'E8208C10	32
	FFU0_0 FIFO upload event clear register	UEVCR_FFU0_0	R/W*1	H'00000000	H'E8208C18	32
FFU0_1	FFU0_1 FIFO upload initialization register	FFUIR_FFU0_1	R/W	H'00000001	H'E8208D00	32
	FFU0_1 FIFO upload audio information register	FUAIR_FFU0_1	R/W	H'00000000	H'E8208D04	32
	FFU0_1 FIFO upload request size register	URQSR_FFU0_1	R/W	H'00000000	H'E8208D08	32
	FFU0_1 FIFO upload pass register	FFUPR_FFU0_1	R/W	H'00000000	H'E8208D0C	32
	FFU0_1 FIFO upload event mask register	UEVMR_FFU0_1	R/W	H'00000000	H'E8208D10	32
	FFU0_1 FIFO upload event clear register	UEVCR_FFU0_1	R/W*1	H'00000000	H'E8208D18	32
FFU0_2	FFU0_2 FIFO upload initialization register	FFUIR_FFU0_2	R/W	H'00000001	H'E8208E00	32
	FFU0_2 FIFO upload audio information register	FUAIR_FFU0_2	R/W	H'00000000	H'E8208E04	32
	FFU0_2 FIFO upload request size register	URQSR_FFU0_2	R/W	H'00000000	H'E8208E08	32
	FFU0_2 FIFO upload pass register	FFUPR_FFU0_2	R/W	H'00000000	H'E8208E0C	32
	FFU0_2 FIFO upload event mask register	UEVMR_FFU0_2	R/W	H'00000000	H'E8208E10	32
	FFU0_2 FIFO upload event clear register	UEVCR_FFU0_2	R/W*1	H'00000000	H'E8208E18	32
FFU0_3	FFU0_3 FIFO upload initialization register	FFUIR_FFU0_3	R/W	H'00000001	H'E8208F00	32
	FFU0_3 FIFO upload audio information register	FUAIR_FFU0_3	R/W	H'00000000	H'E8208F04	32
	FFU0_3 FIFO upload request size register	URQSR_FFU0_3	R/W	H'00000000	H'E8208F08	32
	FFU0_3 FIFO upload pass register	FFUPR_FFU0_3	R/W	H'00000000	H'E8208F0C	32
	FFU0_3 FIFO upload event mask register	UEVMR_FFU0_3	R/W	H'00000000	H'E8208F10	32
	FFU0_3 FIFO upload event clear register	UEVCR_FFU0_3	R/W*1	H'00000000	H'E8208F18	32

Table 48.2 Register Configuration

Block Name	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
2SRC0_0 (SRC0)	2SRC0_0 initialization register 0	SRCIR0_2SRC0_0	R/W	H'00000001	H'E8209000	32
	2SRC0_0 audio information register 0	SADIR0_2SRC0_0	R/W	H'00000000	H'E8209004	32
	2SRC0_0 bypass register 0	SRCBR0_2SRC0_0	R/W	H'00000000	H'E8209008	32
	2SRC0_0 IFS control register 0	IFSCR0_2SRC0_0	R/W	H'00000000	H'E820900C	32
	2SRC0_0 IFS value setting register 0	IFSVR0_2SRC0_0	R/W	H'00000000	H'E8209010	32
	2SRC0_0 control register 0	SRCCR0_2SRC0_0	R/W	H'00000000	H'E8209014	32
	2SRC0_0 minimum FS setting register 0	MNFSR0_2SRC0_0	R/W	H'00000000	H'E8209018	32
	2SRC0_0 buffer size setting register 0	BFSSR0_2SRC0_0	R/W	H'00000000	H'E820901C	32
	2SRC0_0 SCU2 status register 0	SC2SR0_2SRC0_0	R	H'00000000	H'E8209020	32
	2SRC0_0 wait time setting register 0	WATSR0_2SRC0_0	R/W	H'00000000	H'E8209024	32
	2SRC0_0 event mask register 0	SEVMR0_2SRC0_0	R/W	H'00000000	H'E8209028	32
	2SRC0_0 event clear register 0	SEVCR0_2SRC0_0	R/W*1	H'00000000	H'E8209030	32
2SRC0_0 (SRC1)	2SRC0_0 initialization register 1	SRCIR1_2SRC0_0	R/W	H'00000001	H'E8209034	32
	2SRC0_0 audio information register 1	SADIR1_2SRC0_0	R/W	H'00000000	H'E8209038	32
	2SRC0_0 bypass register 1	SRCBR1_2SRC0_0	R/W	H'00000000	H'E820903C	32
	2SRC0_0 IFS control register 1	IFSCR1_2SRC0_0	R/W	H'00000000	H'E8209040	32
	2SRC0_0 IFS value setting register 1	IFSVR1_2SRC0_0	R/W	H'00000000	H'E8209044	32
	2SRC0_0 control register 1	SRCCR1_2SRC0_0	R/W	H'00000000	H'E8209048	32
	2SRC0_0 minimum FS setting register 1	MNFSR1_2SRC0_0	R/W	H'00000000	H'E820904C	32
	2SRC0_0 buffer size setting register 1	BFSSR1_2SRC0_0	R/W	H'00000000	H'E8209050	32
	2SRC0_0 SCU2 status register 1	SC2SR1_2SRC0_0	R	H'00000000	H'E8209054	32
	2SRC0_0 wait time setting register 1	WATSR1_2SRC0_0	R/W	H'00000000	H'E8209058	32
	2SRC0_0 event mask register 1	SEVMR1_2SRC0_0	R/W	H'00000000	H'E820905C	32
	2SRC0_0 event clear register 1	SEVCR1_2SRC0_0	R/W*1	H'00000000	H'E8209064	32
	2SRC0_0 initialization register RIF	SRCIRR_2SRC0_0	R/W	H'00000001	H'E8209068	32
	2SRC0_1 (SRC2)	2SRC0_1 initialization register 0	SRCIR0_2SRC0_1	R/W	H'00000001	H'E8209100
2SRC0_1 audio information register 0		SADIR0_2SRC0_1	R/W	H'00000000	H'E8209104	32
2SRC0_1 bypass register 0		SRCBR0_2SRC0_1	R/W	H'00000000	H'E8209108	32
2SRC0_1 IFS control register 0		IFSCR0_2SRC0_1	R/W	H'00000000	H'E820910C	32
2SRC0_1 IFS value setting register 0		IFSVR0_2SRC0_1	R/W	H'00000000	H'E8209110	32
2SRC0_1 control register 0		SRCCR0_2SRC0_1	R/W	H'00000000	H'E8209114	32
2SRC0_1 minimum FS setting register 0		MNFSR0_2SRC0_1	R/W	H'00000000	H'E8209118	32
2SRC0_1 buffer size setting register 0		BFSSR0_2SRC0_1	R/W	H'00000000	H'E820911C	32
2SRC0_1 SCU2 status register 0		SC2SR0_2SRC0_1	R	H'00000000	H'E8209120	32
2SRC0_1 wait time setting register 0		WATSR0_2SRC0_1	R/W	H'00000000	H'E8209124	32
2SRC0_1 event mask register 0		SEVMR0_2SRC0_1	R/W	H'00000000	H'E8209128	32
2SRC0_1 event clear register 0		SEVCR0_2SRC0_1	R/W*1	H'00000000	H'E8209130	32

Table 48.2 Register Configuration

Block Name	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
2SRC0_1 (SRC3)	2SRC0_1 initialization register 1	SRCIR1_2SRC0_1	R/W	H'00000001	H'E8209134	32
	2SRC0_1 audio information register 1	SADIR1_2SRC0_1	R/W	H'00000000	H'E8209138	32
	2SRC0_1 bypass register 1	SRCBR1_2SRC0_1	R/W	H'00000000	H'E820913C	32
	2SRC0_1 IFS control register 1	IFSCR1_2SRC0_1	R/W	H'00000000	H'E8209140	32
	2SRC0_1 IFS value setting register 1	IFSVR1_2SRC0_1	R/W	H'00000000	H'E8209144	32
	2SRC0_1 control register 1	SRCCR1_2SRC0_1	R/W	H'00000000	H'E8209148	32
	2SRC0_1 minimum FS setting register 1	MNFSR1_2SRC0_1	R/W	H'00000000	H'E820914C	32
	2SRC0_1 buffer size setting register 1	BFSSR1_2SRC0_1	R/W	H'00000000	H'E8209150	32
	2SRC0_1 SCU2 status register 1	SC2SR1_2SRC0_1	R	H'00000000	H'E8209154	32
	2SRC0_1 wait time setting register 1	WATSR1_2SRC0_1	R/W	H'00000000	H'E8209158	32
	2SRC0_1 event mask register 1	SEVMR1_2SRC0_1	R/W	H'00000000	H'E820915C	32
	2SRC0_1 event clear register 1	SEVCR1_2SRC0_1	R/W*1	H'00000000	H'E8209164	32
	2SRC0_1 initialization register RIF	SRCIRR_2SRC0_1	R/W	H'00000001	H'E8209168	32
	DVU0_0	DVU0_0 initialization register	DVUIR_DVU0_0	R/W	H'00000001	H'E8209200
DVU0_0 audio information register		VADIR_DVU0_0	R/W	H'00000000	H'E8209204	32
DVU0_0 bypass register		DVUBR_DVU0_0	R/W	H'00000000	H'E8209208	32
DVU0_0 control register		DVUCR_DVU0_0	R/W	H'00000000	H'E820920C	32
DVU0_0 zero cross mute control register		ZCMCR_DVU0_0	R/W	H'00000000	H'E8209210	32
DVU0_0 volume ramp control register		VRCTR_DVU0_0	R/W	H'00000000	H'E8209214	32
DVU0_0 volume ramp period register		VRPDR_DVU0_0	R/W	H'00000000	H'E8209218	32
DVU0_0 volume ramp decibel register		VRDBR_DVU0_0	R/W	H'00000000	H'E820921C	32
DVU0_0 volume ramp wait time register		VRWTR_DVU0_0	R/W	H'00000000	H'E8209220	32
DVU0_0 volume value setting 0 register		VOL0R_DVU0_0	R/W	H'00000000	H'E8209224	32
DVU0_0 volume value setting 1 register		VOL1R_DVU0_0	R/W	H'00000000	H'E8209228	32
DVU0_0 volume value setting 2 register		VOL2R_DVU0_0	R/W	H'00000000	H'E820922C	32
DVU0_0 volume value setting 3 register		VOL3R_DVU0_0	R/W	H'00000000	H'E8209230	32
DVU0_0 volume value setting 4 register		VOL4R_DVU0_0	R/W	H'00000000	H'E8209234	32
DVU0_0 volume value setting 5 register		VOL5R_DVU0_0	R/W	H'00000000	H'E8209238	32
DVU0_0 volume value setting 6 register		VOL6R_DVU0_0	R/W	H'00000000	H'E820923C	32
DVU0_0 volume value setting 7 register		VOL7R_DVU0_0	R/W	H'00000000	H'E8209240	32
DVU0_0 enable register		DVUER_DVU0_0	R/W	H'00000000	H'E8209244	32
DVU0_0 status register		DVUSR_DVU0_0	R	H'00000000	H'E8209248	32
DVU0_0 event mask register		VEVMR_DVU0_0	R/W	H'00000000	H'E820924C	32
DVU0_0 event clear register		VEVCR_DVU0_0	R/W*1	H'00000000	H'E8209254	32

**Table 48.2 Register Configuration**

Block Name	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
DVU0_1	DVU0_1 initialization register	DVUIR_DVU0_1	R/W	H'00000001	H'E8209300	32
	DVU0_1 audio information register	VADIR_DVU0_1	R/W	H'00000000	H'E8209304	32
	DVU0_1 bypass register	DVUBR_DVU0_1	R/W	H'00000000	H'E8209308	32
	DVU0_1 control register	DVUCR_DVU0_1	R/W	H'00000000	H'E820930C	32
	DVU0_1 zero cross mute control register	ZCMCR_DVU0_1	R/W	H'00000000	H'E8209310	32
	DVU0_1 volume ramp control register	VRCTR_DVU0_1	R/W	H'00000000	H'E8209314	32
	DVU0_1 volume ramp period register	VRPDR_DVU0_1	R/W	H'00000000	H'E8209318	32
	DVU0_1 volume ramp decibel register	VRDBR_DVU0_1	R/W	H'00000000	H'E820931C	32
	DVU0_1 volume ramp wait time register	VRWTR_DVU0_1	R/W	H'00000000	H'E8209320	32
	DVU0_1 volume value setting 0 register	VOL0R_DVU0_1	R/W	H'00000000	H'E8209324	32
	DVU0_1 volume value setting 1 register	VOL1R_DVU0_1	R/W	H'00000000	H'E8209328	32
	DVU0_1 volume value setting 2 register	VOL2R_DVU0_1	R/W	H'00000000	H'E820932C	32
	DVU0_1 volume value setting 3 register	VOL3R_DVU0_1	R/W	H'00000000	H'E8209330	32
	DVU0_1 volume value setting 4 register	VOL4R_DVU0_1	R/W	H'00000000	H'E8209334	32
	DVU0_1 volume value setting 5 register	VOL5R_DVU0_1	R/W	H'00000000	H'E8209338	32
	DVU0_1 volume value setting 6 register	VOL6R_DVU0_1	R/W	H'00000000	H'E820933C	32
	DVU0_1 volume value setting 7 register	VOL7R_DVU0_1	R/W	H'00000000	H'E8209340	32
	DVU0_1 enable register	DVUER_DVU0_1	R/W	H'00000000	H'E8209344	32
	DVU0_1 status register	DVUSR_DVU0_1	R	H'00000000	H'E8209348	32
	DVU0_1 event mask register	VEVMR_DVU0_1	R/W	H'00000000	H'E820934C	32
DVU0_1 event clear register	VEVCR_DVU0_1	R/W*1	H'00000000	H'E8209354	32	
DVU0_2	DVU0_2 initialization register	DVUIR_DVU0_2	R/W	H'00000001	H'E8209400	32
	DVU0_2 audio information register	VADIR_DVU0_2	R/W	H'00000000	H'E8209404	32
	DVU0_2 bypass register	DVUBR_DVU0_2	R/W	H'00000000	H'E8209408	32
	DVU0_2 control register	DVUCR_DVU0_2	R/W	H'00000000	H'E820940C	32
	DVU0_2 zero cross mute control register	ZCMCR_DVU0_2	R/W	H'00000000	H'E8209410	32
	DVU0_2 volume ramp control register	VRCTR_DVU0_2	R/W	H'00000000	H'E8209414	32
	DVU0_2 volume ramp period register	VRPDR_DVU0_2	R/W	H'00000000	H'E8209418	32
	DVU0_2 volume ramp decibel register	VRDBR_DVU0_2	R/W	H'00000000	H'E820941C	32
	DVU0_2 volume ramp wait time register	VRWTR_DVU0_2	R/W	H'00000000	H'E8209420	32
	DVU0_2 volume value setting 0 register	VOL0R_DVU0_2	R/W	H'00000000	H'E8209424	32
	DVU0_2 volume value setting 1 register	VOL1R_DVU0_2	R/W	H'00000000	H'E8209428	32
	DVU0_2 volume value setting 2 register	VOL2R_DVU0_2	R/W	H'00000000	H'E820942C	32
	DVU0_2 volume value setting 3 register	VOL3R_DVU0_2	R/W	H'00000000	H'E8209430	32
	DVU0_2 volume value setting 4 register	VOL4R_DVU0_2	R/W	H'00000000	H'E8209434	32
	DVU0_2 volume value setting 5 register	VOL5R_DVU0_2	R/W	H'00000000	H'E8209438	32
	DVU0_2 volume value setting 6 register	VOL6R_DVU0_2	R/W	H'00000000	H'E820943C	32
	DVU0_2 volume value setting 7 register	VOL7R_DVU0_2	R/W	H'00000000	H'E8209440	32
	DVU0_2 enable register	DVUER_DVU0_2	R/W	H'00000000	H'E8209444	32
	DVU0_2 status register	DVUSR_DVU0_2	R	H'00000000	H'E8209448	32
	DVU0_2 event mask register	VEVMR_DVU0_2	R/W	H'00000000	H'E820944C	32
DVU0_2 event clear register	VEVCR_DVU0_2	R/W*1	H'00000000	H'E8209454	32	

**Table 48.2 Register Configuration**

Block Name	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
DVU0_3	DVU0_3 initialization register	DVUIR_DVU0_3	R/W	H'00000001	H'E8209500	32
	DVU0_3 audio information register	VADIR_DVU0_3	R/W	H'00000000	H'E8209504	32
	DVU0_3 bypass register	DVUBR_DVU0_3	R/W	H'00000000	H'E8209508	32
	DVU0_3 control register	DVUCR_DVU0_3	R/W	H'00000000	H'E820950C	32
	DVU0_3 zero cross mute control register	ZCMCR_DVU0_3	R/W	H'00000000	H'E8209510	32
	DVU0_3 volume ramp control register	VRCTR_DVU0_3	R/W	H'00000000	H'E8209514	32
	DVU0_3 volume ramp period register	VRPDR_DVU0_3	R/W	H'00000000	H'E8209518	32
	DVU0_3 volume ramp decibel register	VRDBR_DVU0_3	R/W	H'00000000	H'E820951C	32
	DVU0_3 volume ramp wait time register	VRWTR_DVU0_3	R/W	H'00000000	H'E8209520	32
	DVU0_3 volume value setting 0 register	VOL0R_DVU0_3	R/W	H'00000000	H'E8209524	32
	DVU0_3 volume value setting 1 register	VOL1R_DVU0_3	R/W	H'00000000	H'E8209528	32
	DVU0_3 volume value setting 2 register	VOL2R_DVU0_3	R/W	H'00000000	H'E820952C	32
	DVU0_3 volume value setting 3 register	VOL3R_DVU0_3	R/W	H'00000000	H'E8209530	32
	DVU0_3 volume value setting 4 register	VOL4R_DVU0_3	R/W	H'00000000	H'E8209534	32
	DVU0_3 volume value setting 5 register	VOL5R_DVU0_3	R/W	H'00000000	H'E8209538	32
	DVU0_3 volume value setting 6 register	VOL6R_DVU0_3	R/W	H'00000000	H'E820953C	32
	DVU0_3 volume value setting 7 register	VOL7R_DVU0_3	R/W	H'00000000	H'E8209540	32
	DVU0_3 enable register	DVUER_DVU0_3	R/W	H'00000000	H'E8209544	32
	DVU0_3 status register	DVUSR_DVU0_3	R	H'00000000	H'E8209548	32
	DVU0_3 event mask register	VEVMR_DVU0_3	R/W	H'00000000	H'E820954C	32
DVU0_3 event clear register	VEVCR_DVU0_3	R/W*1	H'00000000	H'E8209554	32	
MIX	MIX0_0 initialization register	MIXIR_MIX0_0	R/W	H'00000001	H'E8209600	32
	MIX0_0 audio information register	MADIR_MIX0_0	R/W	H'00000000	H'E8209604	32
	MIX0_0 bypass register	MIXBR_MIX0_0	R/W	H'00000000	H'E8209608	32
	MIX0_0 mode register	MIXMR_MIX0_0	R/W	H'00000000	H'E820960C	32
	MIX0_0 volume period register	MVPDR_MIX0_0	R/W	H'00000000	H'E8209610	32
	MIX0_0 decibel A register	MDBAR_MIX0_0	R/W	H'00000000	H'E8209614	32
	MIX0_0 decibel B register	MDBBR_MIX0_0	R/W	H'00000000	H'E8209618	32
	MIX0_0 decibel C register	MDBCR_MIX0_0	R/W	H'00000000	H'E820961C	32
	MIX0_0 decibel D register	MDBDR_MIX0_0	R/W	H'00000000	H'E8209620	32
	MIX0_0 decibel enable register	MDBER_MIX0_0	R/W	H'00000000	H'E8209624	32
MIX0_0 status register	MIXSR_MIX0_0	R	H'00000000	H'E8209628	32	

**Table 48.2 Register Configuration**

Block Name	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
CIM	Software reset register	SWRSR_CIM	R/W	H'00000001	H'E8209700	32
	DMA control register	DMACR_CIM	R/W	H'00000000	H'E8209704	32
	DMA transfer register for FFD0_0 RAM	DMATD0_CIM	W	—	H'E8209708	16, 32*2
	DMA transfer register for FFD0_1 RAM	DMATD1_CIM	W	—	H'E820970C	16, 32*3
	DMA transfer register for FFD0_2 RAM	DMATD2_CIM	W	—	H'E8209710	16, 32*4
	DMA transfer register for FFD0_3 RAM	DMATD3_CIM	W	—	H'E8209714	16, 32*5
	DMA transfer register for FFU0_0 RAM	DMATU0_CIM	R	—	H'E8209718	16, 32*6
	DMA transfer register for FFU0_1 RAM	DMATU1_CIM	R	—	H'E820971C	16, 32*7
	DMA transfer register for FFU0_2 RAM	DMATU2_CIM	R	—	H'E8209720	16, 32*8
	DMA transfer register for FFU0_3 RAM	DMATU3_CIM	R	—	H'E8209724	16, 32*9
	SSI route select register	SSIRSEL_CIM	R/W	H'00000000	H'E8209738	32
	FFD0_0 timing select register	FDTSEL0_CIM	R/W	H'00000000	H'E820973C	32
	FFD0_1 timing select register	FDTSEL1_CIM	R/W	H'00000000	H'E8209740	32
	FFD0_2 timing select register	FDTSEL2_CIM	R/W	H'00000000	H'E8209744	32
	FFD0_3 timing select register	FDTSEL3_CIM	R/W	H'00000000	H'E8209748	32
	FFU0_0 timing select register	FUTSEL0_CIM	W	H'00000000	H'E820974C	32
	FFU0_1 timing select register	FUTSEL1_CIM	W	H'00000000	H'E8209750	32
	FFU0_2 timing select register	FUTSEL2_CIM	W	H'00000000	H'E8209754	32
	FFU0_3 timing select register	FUTSEL3_CIM	W	H'00000000	H'E8209758	32
	SSI pin mode register	SSIPMD_CIM	R/W	H'00000000	H'E820975C	32
	SSI control register	SSICTRL_CIM	W	H'00000000	H'E8209760	32
	SRC0 route select register	SRCRSEL0_CIM	R/W	H'76543210	H'E8209764	32
	SRC1 route select register	SRCRSEL1_CIM	R/W	H'76543210	H'E8209768	32
	SRC2 route select register	SRCRSEL2_CIM	R/W	H'76543210	H'E820976C	32
	SRC3 route select register	SRCRSEL3_CIM	R/W	H'76543210	H'E8209770	32
	MIX route select register	MIXRSEL_CIM	R/W	H'76543210	H'E8209774	32

Note 1. Only 0 can be written to clear the flag. Writing 0 is ignored.

Note 2. Address H'E8209708 can be accessed in only 16-bit or 32-bit units.

Note 3. Address H'E820970C can be accessed in only 16-bit or 32-bit units.

Note 4. Address H'E8209710 can be accessed in only 16-bit or 32-bit units.

Note 5. Address H'E8209714 can be accessed in only 16-bit or 32-bit units.

Note 6. Address H'E8209718 can be accessed in only 16-bit or 32-bit units.

Note 7. Address H'E820971C can be accessed in only 16-bit or 32-bit units.

Note 8. Address H'E8209720 can be accessed in only 16-bit or 32-bit units.

Note 9. Address H'E8209724 can be accessed in only 16-bit or 32-bit units.



### 48.3.1 IPC0\_n Initialization Register (IPCIR\_IPC0\_n) (n = 0, 1, 2, 3)

IPCIR\_IPC0\_n is a 32-bit readable/writable register that initializes the IPC internal circuit.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	INIT
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	INIT	1	R/W	Initialization When this bit is set to 1, the IPC internal circuit is initialized. To cancel the initialization, set this bit to 0 with it set to 1. 0: Processing State 1: Initialization (sets the initial setting of other registers)

### 48.3.2 IPC0\_n Pass Select Register (IPSLR\_IPC0\_n) (n = 0, 1, 2, 3)

IPSLR\_IPC0\_n is a 32-bit readable/writable register that selects the input source and output destination of the input timing signal and audio data.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	IPC_PASS_SEL	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	IPC_PASS_SEL	0	R/W	IPC Pass Select These bits select the pass. 000: No operation 001: Input Timing Signal/Input Audio Data : EXTERNAL (SSIF) → IPC → SRC (Async mode) 010: Reserved 011: Input Timing Signal : EXTERNAL → IPC → FFD/SRC Input Audio Data : FFD → IPC → SRC (Async mode) 100: Input Timing Signal/Input Audio Data : FFD → IPC → SRC (Sync mode) 101 to 111: No operation

### 48.3.3 OPC0\_n Initialization Register (OPCIR\_OPC0\_n) (n = 0, 1, 2, 3)

OPCIR\_OPC0\_n is a 32-bit readable/writable register that initializes the OPC internal circuit.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	INIT
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	INIT	1	R/W	Initialization When this bit is set to 1, the OPC internal circuit is initialized. To cancel the initialization, set this bit to 0 with it set to 1. 0: Processing State 1: Initialization (sets the initial setting of other registers)

### 48.3.4 OPC0\_n Pass Select Register (OPSLR\_OPC0\_n) (n = 0, 1, 2, 3)

OPSLR\_OPC0\_n is a 32-bit readable/writable register that selects the input source and output destination of the output timing signal and audio data.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	OPC_PASS_SEL	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	OPC_PASS_SEL	0	R/W	OPC Pass Select These bits select the pass. 000: No operation 001: Output Timing Signal : EXTERNAL → OPC → SRC/DVU Output Audio Data : SRC (Async mode) → OPC → DVU 010: Reserved 011: Output Timing Signal : EXTERNAL → OPC → FFU/SRC Output Audio Data : SRC (Async mode) → OPC → FFU 100: Output Timing Signal : FFU → OPC → SRC Output Audio Data : SRC (Sync mode) → OPC → FFU 101 to 111: No operation

### 48.3.5 FFD0\_n FIFO Download Initialization Register (FFDIR\_FFD0\_n) (n = 0, 1, 2, 3)

FFDIR\_FFD0\_n is a 32-bit readable/writable register that initializes the FFD internal circuit.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	INIT
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	INIT	1	R/W	Initialization When this bit is set to 1, the FFD internal circuit is initialized. To cancel the initialization, set this bit to 0 with it set to 1. 0: Processing State 1: Initialization (sets the initial setting of other registers)

### 48.3.6 FFD0\_n FIFO Download Audio Information Register (FDAIR\_FFD0\_n) (n = 0, 1, 2, 3)

FDAIR\_FFD0\_n is a 32-bit readable/writable register that sets the number of FFD data channels.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CHNUM
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	CHNUM	All 0	R/W	Number of Channels These bits set the number of channels. 0000: Zero (None) 0001: 1 channel 0010: 2 channels 0011: Reserved 0100: 4 channels* 0101: Reserved 0110: 6 channels* 0111: Reserved 1000: 8 channels* 1001 to 1111: Reserved Note: * These bits are only settable for FFD0_0 and FFD0_1. Setting these bits in the registers for FFD0_2 and FFD0_3 is prohibited.

### 48.3.7 FFD0\_n FIFO Download Request Size Register (DRQSR\_FFD0\_n) (n = 0, 1, 2, 3)

DRQSR\_FFD0\_n is a 32-bit readable/writable register that sets the request size of data transfer requests.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	SIZE			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	SIZE	All 0	R/W	Request Size These bits set the PCM data size per request. The size must be less than the FIFO size. 0000: 256 data. (FFD0_0,FFD0_1), 64 data. (FFD0_2,FFD0_3) *1 0001: 128 data. (FFD0_0,FFD0_1), 32 data. (FFD0_2,FFD0_3) 0010: 64 data. (FFD0_0,FFD0_1), 16 data. (FFD0_2,FFD0_3) 0011: 32 data. (FFD0_0,FFD0_1), 8 data. (FFD0_2,FFD0_3) 0100: 16 data. (FFD0_0,FFD0_1), 4 data. (FFD0_2,FFD0_3) 0101: 8 data. (FFD0_0,FFD0_1), 2 data. (FFD0_2,FFD0_3) 0110: 4 data. (FFD0_0,FFD0_1), 1 data. (FFD0_2,FFD0_3) 0111: 2 data. (FFD0_0,FFD0_1)*2 1000: 1 data. (FFD0_0,FFD0_1)*2 1001 to 1111: Reserved. Notes: 1. If the request size equals the FIFO size, the request size must be a multiple of the number of channels so that the operation is correct. For example, the request size and FIFO size are 256, the number of channels can be set to 1, 2, 4, or 8. If the number of channels is not set to any of the numbers, FIFO overflow may occur. 2. These bits are only settable for FFD0_0 and FFD0_1. Setting these bits in the registers for FFD0_2 and FFD0_3 is prohibited.

### 48.3.8 FFD0\_n FIFO Download Pass Register (FFDPR\_FFD0\_n) (n = 0, 1, 2, 3)

FFDPR\_FFD0\_n is a 32-bit readable/writable register that sets the input/output route of audio data.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	PASS
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	PASS	All 0	R/W	Data Pass Select These bits set the data pass of audio data. 00: No pass selected 01: CIM → FFD → IPC (Async Mode) *1 10: CIM → FFD → IPC (Sync Mode) *2 11: Reserved Notes: 1.- Async mode: The FFD use external timing. 2.- Sync mode: The FFD use the request which is generated by SRC module. Set Sync mode in SRC.

### 48.3.9 FFD0\_n FIFO Download Boot Register (FFDDBR\_FFD0\_n) (n = 0, 1, 2, 3)

FFDDBR\_FFD0\_n is a 32-bit readable/writable register that controls the starting and stopping of data requests sent to RFFD.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	BOOT
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	BOOT	0	R/W	Data Request Start/Stop This bit controls the start/stop of data request. This bit can be set when the PASS bits in the FFDPR register are set to B'01 or B'10. This bit must be set after setting the FFD register. 0: Stops the data request. 1: Starts the data request.

### 48.3.10 FFD0\_n FIFO Download Event Mask Register (DEVMR\_FFD0\_n) (n = 0, 1, 2, 3)

DEVMR\_FFD0\_n is a 32-bit readable/writable register that controls SCUFDIn interrupt requests.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DEV MUF	DEV MOF	DEV MOL	DEV MIUF	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DEV MRQ	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	DEVMUF	0	R/W	FFD Underflow Mask This bit sets whether to mask the interrupt request from the DEVCUF bit in the DEVCR register. 0: Disables interrupts. 1: Enables interrupts.
30	DEVMOF	0	R/W	FFD Overflow Mask This bit sets whether to mask the interrupt request from the DEVCOF bit in the DEVCR register. 0: Disables interrupts. 1: Enables interrupts.
29	DEVMOL	0	R/W	FFD Overlap Mask This bit sets whether to mask the interrupt request from the DEVCOL bit in the DEVCR register. 0: Disables interrupts. 1: Enables interrupts.
28	DEVMUIF	0	R/W	FFD Initialization Underflow Mask This bit sets whether to mask the interrupt request from the DEVCIUF bit in the DEVCR register. 0: Disables interrupts. 1: Enables interrupts.
27 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15	DEVMRQ	0	R/W	FFD Request Packet Mask This bit sets whether to mask the interrupt request from the DEVCRQ bit in the DEVCR register. 0: Disables interrupts. 1: Enables interrupts.
14 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

### 48.3.11 FFD0\_n FIFO Download Event Clear Register (DEVCR\_FFD0\_n) (n = 0, 1, 2, 3)

DEVCR\_FFD0\_n is a 32-bit readable/writable register that clears SCUFDIn interrupt requests. When an interrupt event is generated, the relevant bit in this register is automatically set to 1 and 1 is retained until 0 is written to that bit.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DEV CUF	DEV COF	DEV COL	DEV CIUF	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/(W)*1	R/(W)*1	R/(W)*1	R/(W)*1	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DEV CRQ	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/(W)*1	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	DEVCUF	0	R/(W)*1	FFD Underflow Clear This is an interrupt flag to indicate whether DPRAM underflow occurs. When it occurs and the first packet has already been written to DPRAM, this bit is set to 1. 0: Clears the flag. 1: Retains the flag.
30	DEVCOF	0	R/(W)*1	FFD Overflow Clear This is an interrupt flag to indicate whether DPRAM overflow occurs. When it occurs, this bit is set to 1. 0: Clears the flag. 1: Retains the flag.
29	DEVCOL	0	R/(W)*1	FFD Overlap Clear This is an interrupt flag to indicate whether FFD overlap occurs. When the next read request occurs while FFD reads data from DPRAM upon the previous read request, the overflow occurs. When it occurs, this bit is set to 1. 0: Clears the flag. 1: Retains the flag.
28	DEVCIUF	0	R/(W)*1	FFD Initialization Underflow Clear This is an interrupt flag to indicate whether DPRAM initialization underflow occurs. When it occurs and the first packet has not been written to DPRAM, this bit is set to 1. 0: Clears the flag. 1: Retains the flag.
27 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15	DEVCRQ	0	R/(W)*1	FFD Request Packet Clear This is an interrupt flag to send a request to write one packet in FFD, to the CPU. When the request occurs, this bit is set to 1. 0: Clears the flag. 1: Retains the flag.
14 to 0	—	All 0	R	Reserved. These bits are always read as 0. The write value should always be 0.

Note 1. Readable/writable. When 0 is written, the bit is initialized. Writing 1 to the bit is ignored. Write 0 only to each bit corresponding to the interrupt source to be cleared; write 1 to other bits.

### 48.3.12 FFU0\_n FIFO Upload Initialization Register (FFUIR\_FFU0\_n) (n = 0, 1, 2, 3)

FFUIR\_FFU0\_n is a 32-bit readable/writable register that initializes the FFU internal circuit.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	INIT
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	INIT	1	R/W	Initialization When this bit is set to 1, the FFU internal circuit is initialized. To cancel the initialization, set this bit to 0 with it set to 1. 0: Processing State 1: Initialization (sets the initial setting of other registers)

### 48.3.13 FFU0\_n FIFO Upload Audio Information Register (FUAIR\_FFU0\_n) (n = 0, 1, 2, 3)

FUAIR\_FFU0\_n is a 32-bit readable/writable register that sets the number of FFU data channels.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	CHNUM			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	CHNUM	All 0	R/W	Number of Channels These bits set the number of channels. For details, see below. 0000: Zero (None) 0001: 1 channel 0010: 2 channels 0011: Reserved 0100: 4 channels* 0101: Reserved 0110: 6 channels* 0111: Reserved 1000: 8 channels* 1001 to 1111: Reserved Note: * These bits are only settable for FFU0_0 and FFU0_1. Setting these bits in the registers for FFU0_2 and FFU0_3 is prohibited.



### 48.3.14 FFU0\_n FIFO Upload Request Size Register (URQSR\_FFU0\_n) (n = 0, 1, 2, 3)

URQSR\_FFU0\_n is a 32-bit readable/writable register that sets the request size of data transfer requests.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	SIZE				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	SIZE	All 0	R/W	Request Size These bits set the PCM data size per request. The size must be less than the FIFO size. 0000: 256 data. (FFU0_0,FFU0_1), 64 data. (FFU0_2,FFU0_3) *1 0001: 128 data. (FFU0_0,FFU0_1), 32 data. (FFU0_2,FFU0_3) 0010: 64 data. (FFU0_0,FFU0_1), 16 data. (FFU0_2,FFU0_3) 0011: 32 data. (FFU0_0,FFU0_1), 8 data. (FFU0_2,FFU0_3) 0100: 16 data. (FFU0_0,FFU0_1), 4 data. (FFU0_2,FFU0_3) 0101: 8 data. (FFU0_0,FFU0_1), 2 data. (FFU0_2,FFU0_3) 0110: 4 data. (FFU0_0,FFU0_1), 1 data. (FFU0_2,FFU0_3) 0111: 2 data. (FFU0_0,FFU0_1)*2 1000: 1 data. (FFU0_0,FFU0_1)*2 1001 to 1111: Reserved. Notes: 1. If the request size equals the FIFO size, the request size must be a multiple of the number of channels so that the operation is correct. For example, the request size and FIFO size are 256, the number of channels can be set to 1, 2, 4, or 8. If the number of channels is not set to any of the numbers, FIFO overflow may occur. 2. These bits are only settable for FFU0_0 and FFU0_1. Setting these bits in the registers for FFU0_2 and FFU0_3 is prohibited.

### 48.3.15 FFU0\_n FIFO Upload Pass Register (FFUPR\_FFU0\_n) (n = 0, 1, 2, 3)

FFUPR\_FFU0\_n is a 32-bit readable/writable register that sets the input/output route of audio data.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	PASS
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	PASS	All 0	R/W	Data Pass Select These bits set the data pass of audio data. 00: No pass selected 01: CIM ← FFU ← OPC (Async Mode) *1 10: CIM ← FFU ← OPC (Sync Mode) *2 11: Reserved Notes: 1. Async mode: The FFU use external timing. 2. Sync mode: The FFU use the request which is generated by SRC module. Set Sync mode in SRC.

### 48.3.16 FFU0\_n FIFO Upload Event Mask Register (UEVMR\_FFU0\_n) (n = 0, 1, 2, 3)

UEVMR\_FFU0\_n is a 32-bit readable/writable register that controls SCUFUIn interrupt requests.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	UEV MUF	UEV MOF	UEV MOL	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	UEV MRQ	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	UEVMUF	0	R/W	FFU Underflow Mask This bit sets whether to mask the interrupt request from the UEVCUF bit in the UEVCR register. 0: Disables interrupts. 1: Enables interrupts.
30	UEVMOF	0	R/W	FFU Overflow Mask This bit sets whether to mask the interrupt request from the UEVCOF bit in the UEVCR register. 0: Disables interrupts. 1: Enables interrupts.
29	UEVMOL	0	R/W	FFU Overlap Mask This bit sets whether to mask the interrupt request from the UEVCOL bit in the UEVCR register. 0: Disables interrupts. 1: Enables interrupts.
28 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15	UEVMRQ	0	R/W	FFU Request Packet Mask This bit sets whether to mask the interrupt request from the UEVCRQ bit in the UEVCR register. 0: Disables interrupts. 1: Enables interrupts.
14 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

### 48.3.17 FFU0\_n FIFO Upload Event Clear Register (UEVCR\_FFU0\_n) (n = 0, 1, 2, 3)

UEVCR\_FFU0\_n is a 32-bit readable/writable register that clears SCUFUIn interrupt requests. When an interrupt event is generated, the relevant bit in this register is automatically set to 1 and 1 is retained until 0 is written to that bit.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	UEV CUF	UEV COF	UEV COL	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/(W)*1	R/(W)*1	R/(W)*1	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	UEV CRQ	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/(W)*1	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	UEVCUF	0	R/(W)*1	FFU Underflow Clear This is an interrupt flag to indicate whether DPRAM underflow occurs. When it occurs, this bit is set to 1. 0: Clears the flag. 1: Retains the flag.
30	UEVCOF	0	R/(W)*1	FFU Overflow Clear This is an interrupt flag to indicate whether DPRAM overflow occurs. When it occurs, this bit is set to 1. 0: Clears the flag. 1: Retains the flag.
29	UEVCOL	0	R/(W)*1	FFU Overlap Clear This is an interrupt flag to indicate whether DPRAM overlap occurs. When the next write request occurs while FFU writes data upon the previous write request, the overlap occurs. When it occurs, this bit is set to 1. 0: Clears the flag. 1: Retains the flag.
28 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15	UEVCRQ	0	R/(W)*1	FFU Request Packet Clear This is an interrupt flag to send a request to read one packet from FFU, to the CPU. When the request occurs, this bit is set to 1. 0: Clears the flag. 1: Retains the flag.
14 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note 1. Readable/writable. When 0 is written, the bit is initialized. Writing 1 to the bit is ignored. Write 0 only to each bit corresponding to the interrupt source to be cleared; write 1 to other bits.

### 48.3.18 2SRC0\_m Initialization Register p (SRCIRp\_2SRC0\_m) (m = 0, 1; p = 0, 1)

SRCIRp\_2SRC0\_m is a 32-bit readable/writable register that initializes the SRC internal circuit.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	INIT
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	INIT	1	R/W	Initialization When this bit is set to 1, the SRC internal circuit is initialized. To cancel the initialization, set this bit to 0 with it set to 1. 0: Processing State 1: Initialization (sets the initial setting of other registers)

### 48.3.19 2SRC0\_m Audio Information Register p (SADIRp\_2SRC0\_m) (m = 0, 1; p = 0, 1)

SADIRp\_2SRC0\_m is a 32-bit readable/writable register that sets the bit length of data and the number of channels.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	OTBL				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	CHNUM			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20 to 16	OTBL	All 0	R/W	Bit Length of Output Audio Data 00000: 24 bits 00001 to 00111: Reserved 01000: 16 bits 01001 to 11111: Reserved
15 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	CHNUM	All 0	R/W	Number of Channels These bits set the number of channels. 0000: Zero (None) 0001: 1 channel 0010: 2 channels 0011: Reserved 0100: 4 channels 0101: Reserved 0110: 6 channels 0111: Reserved 1000: 8 channels 1001 to 1111: Reserved

### 48.3.20 2SRC0\_m Bypass Register p (SRCBRp\_2SRC0\_m) (m = 0, 1; p = 0, 1)

SRCBRp\_2SRC0\_m is a 32-bit readable/writable register that sets the bypass mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	BY PASS
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	BYPASS	0	R/W	<b>Bypass Mode</b> This bit controls the data pass of SRC. The bypass function is only used in asynchronous SRC mode. Therefore, this bit must be set to 0 when synchronous SRC mode is used (bit SRCMD in SRCCRp_2SRC0_m register is 1). 0: SRC function is used. Input data is processed by SRC and then the result data is connected to output data. 1: SRC function is not used (bypass mode). Input data is directly connected to output data.

### 48.3.21 2SRC0\_m IFS Control Register p (IFSCRp\_2SRC0\_m) (m = 0, 1; p = 0, 1)

IFSCRp\_2SRC0\_m is a 32-bit readable/writable register that enables or disables usage of the initial value set by INTIFS.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	INT IFSEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	INTIFSEN	0	R/W	<b>INTIFS Value Enable</b> This bit is enabled only in asynchronous SRC mode (bit SRCMD in SRCCRp_2SRC0_m register is 0). 0: Disables initial value setting in IFSVR register. 1: Enables initial value setting in IFSVR register.

### 48.3.22 2SRC0\_m IFS Value Setting Register p (IFSVRp\_2SRC0\_m) (m = 0, 1; p = 0, 1)

IFSVRp\_2SRC0\_m is a 32-bit readable/writable register that sets the INTIFS value.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	INTIFS											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	INTIFS															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 0	INTIFS	All 0	R/W	Initial Value of FSI These bits set initial value of FSI. Set the input sampling rate and output sampling rate within the restricted range determined by the number of channels. Example: Fin = 32 kHz Fout = 44.1 kHz FSI = $2^{22} * 32000/44100 = 3043485 = 0x02E709D$

When using the SRC in asynchronous mode, INTIFS is given as the initial value of FSI, which is the sampling conversion function. Therefore, after setting the INTIFSEN bit in IFSCRp\_2SRC0\_m to 1, set the INTIFS bits to the desired value. The SRC automatically detects the input sampling frequency and output sampling frequency. Based on this, the FSI value is calculated so that the following equation is satisfied.

$$\frac{F_{in}}{F_{out}} \approx \frac{FSI}{FSO}$$

[Legend]

Fin: Input sampling frequency

Fout: Output sampling frequency

FSI: Input sampling rate

FSO: Output sampling rate (fixed value:  $2^{22} = 0x0400000$ )

When using the SRC in synchronous mode, the value set in INTIFS is used as the FSI value from beginning to end. The value set in INITFS is shown in Table 48.3 to Table 48.8.



**Table 48.3** INITFS Setting (Delay Mode: Normal, Number of Channels: 1 or 2)

Input Sampling Rate [KHz]	Output Sampling Rate [KHz]						
	8	16	24	32	44.1	48	96
8	0x0400000	0x0200000	0x0155555	0x0100000	0x00b9c27	0x00aaaaa	0x0055555
11.025	0x0583333	0x02c1999	0x01d6666	0x0160ccc	0x0100000	0x00eb333	0x0075999
12	0x0600000	0x0300000	0x0200000	0x0180000	0x0116a3b	0x0100000	0x0080000
16	0x0800000	0x0400000	0x02aaaaa	0x0200000	0x017384e	0x0155555	0x00aaaaa
22.05	0x0b06666	0x0583333	0x03acccc	0x02c1999	0x0200000	0x01d6666	0x00eb333
24	0x0c00000	0x0600000	0x0400000	0x0300000	0x022d476	0x0200000	0x0100000
32	0x1000000	0x0800000	0x0555555	0x0400000	0x02e709d	0x02aaaaa	0x0155555
44.1	0x160cccc	0x0b06666	0x0759999	0x0583333	0x0400000	0x03acccc	0x01d6666
48	0x1800000	0x0c00000	0x0800000	0x0600000	0x045a8ec	0x0400000	0x0200000
64	0x2000000	0x1000000	0x0aaaaaa	0x0800000	0x05ce13b	0x0555555	0x02aaaaa
88.2	Setting prohibited*1	0x160cccc	0x0eb3333	0x0b06666	0x0800000	0x0759999	0x03acccc
96	Setting prohibited*1	0x1800000	0x1000000	0x0c00000	0x08b51d9	0x0800000	0x0400000

Note 1. This is because the setting is outside the specification of the SRC module. For details, see Table 48.22.

**Table 48.4** INITFS Setting (Delay Mode: Normal, Number of Channels: 4)

Input Sampling Rate [KHz]	Output Sampling Rate [KHz]						
	8	16	24	32	44.1	48	96
8	0x0400000	0x0200000	0x0155555	0x0100000	0x00b9c27	0x00aaaaa	0x0055555
11.025	0x0583333	0x02c1999	0x01d6666	0x0160ccc	0x0100000	0x00eb333	0x0075999
12	0x0600000	0x0300000	0x0200000	0x0180000	0x0116a3b	0x0100000	0x0080000
16	0x0800000	0x0400000	0x02aaaaa	0x0200000	0x017384e	0x0155555	0x00aaaaa
22.05	0x0b06666	0x0583333	0x03acccc	0x02c1999	0x0200000	0x01d6666	0x00eb333
24	0x0c00000	0x0600000	0x0400000	0x0300000	0x022d476	0x0200000	0x0100000
32	0x1000000	0x0800000	0x0555555	0x0400000	0x02e709d	0x02aaaaa	0x0155555
44.1	Setting prohibited*1	0x0b06666	0x0759999	0x0583333	0x0400000	0x03acccc	0x01d6666
48	Setting prohibited*1	0x0c00000	0x0800000	0x0600000	0x045a8ec	0x0400000	0x0200000
64	Setting prohibited*1	0x1000000	0x0aaaaaa	0x0800000	0x05ce13b	0x0555555	0x02aaaaa
88.2	Setting prohibited*1	Setting prohibited*1	0x0eb3333	0x0b06666	0x0800000	0x0759999	0x03acccc
96	Setting prohibited*1	Setting prohibited*1	0x1000000	0x0c00000	0x08b51d9	0x0800000	0x0400000

Note 1. This is because the setting is outside the specification of the SRC module. For details, see Table 48.22.

**Table 48.5** INITFS Setting (Delay Mode: Normal, Number of Channels: 6)

Input Sampling Rate [KHz]	Output Sampling Rate [KHz]						
	8	16	24	32	44.1	48	96
8	0x0400000	0x0200000	0x0155555	0x0100000	0x00b9c27	0x00aaaaa	Setting prohibited*1
11.025	0x0583333	0x02c1999	0x01d6666	0x0160ccc	0x0100000	0x00eb333	Setting prohibited*1
12	0x0600000	0x0300000	0x0200000	0x0180000	0x0116a3b	0x0100000	Setting prohibited*1
16	0x0800000	0x0400000	0x02aaaaa	0x0200000	0x017384e	0x0155555	Setting prohibited*1
22.05	Setting prohibited*1	0x0583333	0x03acccc	0x02c1999	0x0200000	0x01d6666	Setting prohibited*1
24	Setting prohibited*1	0x0600000	0x0400000	0x0300000	0x022d476	0x0200000	Setting prohibited*1
32	Setting prohibited*1	0x0800000	0x0555555	0x0400000	0x02e709d	0x02aaaaa	Setting prohibited*1
44.1	Setting prohibited*1	Setting prohibited*1	0x0759999	0x0583333	0x0400000	0x03acccc	Setting prohibited*1
48	Setting prohibited*1	Setting prohibited*1	0x0800000	0x0600000	0x045a8ec	0x0400000	Setting prohibited*1
64	Setting prohibited*1*	Setting prohibited*1	0x0aaaaaa	0x0800000	0x05ce13b	0x0555555	Setting prohibited*1
88.2	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1
96	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1

Note 1. This is because the setting is outside the specification of the SRC module. For details, see Table 48.22.

**Table 48.6** INITFS Setting (Delay Mode: Normal, Number of Channels: 8)

Input Sampling Rate [KHz]	Output Sampling Rate [KHz]						
	8	16	24	32	44.1	48	96
8	0x0400000	0x0200000	0x0155555	0x0100000	0x00b9c27	0x00aaaaa	Setting prohibited*1
11.025	0x0583333	0x02c1999	0x01d6666	0x0160ccc	0x0100000	0x00eb333	Setting prohibited*1
12	0x0600000	0x0300000	0x0200000	0x0180000	0x0116a3b	0x0100000	Setting prohibited*1
16	0x0800000	0x0400000	0x02aaaaa	0x0200000	0x017384e	0x0155555	Setting prohibited*1
22.05	Setting prohibited*1	0x0583333	0x03acccc	0x02c1999	0x0200000	0x01d6666	Setting prohibited*1
24	Setting prohibited*1	0x0600000	0x0400000	0x0300000	0x022d476	0x0200000	Setting prohibited*1
32	Setting prohibited*1	0x0800000	0x0555555	0x0400000	0x02e709d	0x02aaaaa	Setting prohibited*1
44.1	Setting prohibited*1	Setting prohibited*1	0x0759999	0x0583333	0x0400000	0x03acccc	Setting prohibited*1
48	Setting prohibited*1	Setting prohibited*1	0x0800000	0x0600000	0x045a8ec	0x0400000	Setting prohibited*1
64	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1
88.2	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1
96	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1

Note 1. This is because the setting is outside the specification of the SRC module. For details, see Table 48.22.

**Table 48.7** INITFS Setting (Delay Mode: Low Delay 1, Number of Channels: 1 or 2)

Input Sampling Rate [KHz]	Output Sampling Rate [KHz]						
	8	16	24	32	44.1	48	96
8	0x0400000	0x0200000	0x0155555	0x0100000	0x00b9c27	0x00aaaaa	0x0055555
11.025	0x0583333	0x02c1999	0x01d6666	0x0160ccc	0x0100000	0x00eb333	0x0075999
12	0x0600000	0x0300000	0x0200000	0x0180000	0x0116a3b	0x0100000	0x0080000
16	0x0800000	0x0400000	0x02aaaaa	0x0200000	0x017384e	0x0155555	0x00aaaaa
22.05	Setting prohibited*1	0x0583333	0x03acccc	0x02c1999	0x0200000	0x01d6666	0x00eb333
24	Setting prohibited*1	0x0600000	0x0400000	0x0300000	0x022d476	0x0200000	0x0100000
32	Setting prohibited*1	0x0800000	0x0555555	0x0400000	0x02e709d	0x02aaaaa	0x0155555
44.1	Setting prohibited*1	Setting prohibited*1	0x0759999	0x0583333	0x0400000	0x03acccc	0x01d6666
48	Setting prohibited*1	Setting prohibited*1	0x0800000	0x0600000	0x045a8ec	0x0400000	0x0200000
64	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	0x0800000	0x05ce13b	0x0555555	0x02aaaaa
88.2	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	0x0800000	0x0759999	0x03acccc
96	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	0x0800000	0x0400000

Note 1. This is because the setting is outside the specification of the SRC module. For details, see Table 48.22.

**Table 48.8** INITFS Setting (Delay Mode: Low Delay 2, Number of Channels: 1 or 2)

Input Sampling Rate [KHz]	Output Sampling Rate [KHz]						
	8	16	24	32	44.1	48	96
8	0x0400000	0x0200000	0x0155555	0x0100000	0x00b9c27	0x00aaaaa	0x0055555
11.025	Setting prohibited*1	0x02c1999	0x01d6666	0x0160ccc	0x0100000	0x00eb333	0x0075999
12	Setting prohibited*1	0x0300000	0x0200000	0x0180000	0x0116a3b	0x0100000	0x0080000
16	Setting prohibited*1	0x0400000	0x02aaaaa	0x0200000	0x017384e	0x0155555	0x00aaaaa
22.05	Setting prohibited*1	Setting prohibited*1	0x03acccc	0x02c1999	0x0200000	0x01d6666	0x00eb333
24	Setting prohibited*1	Setting prohibited*1	0x0400000	0x0300000	0x022d476	0x0200000	0x0100000
32	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	0x0400000	0x02e709d	0x02aaaaa	0x0155555
44.1	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	0x0400000	0x03acccc	0x01d6666
48	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	0x0400000	0x0200000
64	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	0x02aaaaa
88.2	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	0x03acccc
96	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	Setting prohibited*1	0x0400000

Note 1. This is because the setting is outside the specification of the SRC module. For details, see Table 48.22.

### 48.3.23 2SRC0\_m Control Register p (SRCCRp\_2SRC0\_m) (m = 0, 1; p = 0, 1)

SRCCRp\_2SRC0\_m is a 32-bit readable/writable register that selects the low delay mode, and synchronous or asynchronous mode of the SRC.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	WATMD	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	BUFMD	-	-	-	-	-	-	-	-	-	-	-	SRCMD
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20	WATMD	0	R/W	Wait Time Control of SRC 0: Disables wait time control. 1: Enables wait time control.
19 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	—	0	R/W	The write value should always be 1.
15 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12	BUFMD	0	R/W	Low Delay Mode 0: Disables low delay mode. 1: Enables low delay mode.
11 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	—	0	R/W	The write value should always be 1.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	—	0	R/W	The write value should always be 1.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SRCMD	0	R/W	SRC Mode Select 0: Asynchronous SRC 1: Synchronous SRC

### 48.3.24 2SRC0\_m Minimum FS Setting Register p (MNFSRp\_2SRC0\_m) (m = 0, 1; p = 0, 1)

MNFSRp\_2SRC0\_m is a 32-bit readable/writable register that sets the minimum value of FS.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	MINFS											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MINFS															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 28	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
27 to 0	MINFS	All 0	R/W	Minimum Value of FS Ratio These bits set the FS ratio to start fading out the output data to mute under the situation that the input timing stops during SRC processing. This setting value should be less than the value of IFSVRp_2SRC0_m register. Example: Fin = 32 kHz Fout = 44.1 kHz FSI = $2^{22} * 32000/44100 = 3043485 = 0x02E709D$ MINFS = FSI * 98% = 2982615 = 0x02D82D7 Set the ratio which is out of the range of the jitter. (The recommended value is from 90 to 98 percent.)

### 48.3.25 2SRC0\_m Buffer Size Setting Register p (BFSSRp\_2SRC0\_m) (m = 0, 1; p = 0, 1)

BFSSRp\_2SRC0\_m is a 32-bit readable/writable register that sets the buffer size. This register setting is valid only when SRCCRp\_2SRC0\_m.BUFMD = 1.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	BUFDATA									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	BUFIN			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25 to 16	BUFDATA	All 0	R/W	Set the buffer size of 1 channel in DATA RAM. Sets 0x80 for low delay mode 1. Sets 0x40 for low delay mode 2.
15 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	BUFIN	All 0	R/W	Set the buffer size of 1 channel in INDATA RAM. Sets 0x5 for low delay mode.

### 48.3.26 2SRC0\_m SCU2 Status Register p (SC2SRp\_2SRC0\_m) (m = 0, 1; p = 0, 1)

SC2SRp\_2SRC0\_m is a 32-bit read-only register that indicates the status of the SRC.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SRCW STS	SC2 MUTE	SC2STS		SC2FSI											
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SC2FSI															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	SRCWSTS	0	R	Status of Wait Time Operation of SRC 0: Wait for count up to wait time 1: Reached to wait time
30	SC2MUTE	0	R	SRC Mute Status This bit is only used for debug. 0: SRC output data is not fixed and it should be muted. This happens when SRC cannot synchronize the input sampling rate and output sampling rate. 1: SRC output data is fixed.
29, 28	SC2STS	All 0	R	Status of SRC 00: SRC is reset. 01: SRC is initialized. 10: SRC is processing. 11: Reserved
27 to 0	SC2FSI	All 0	R	Latest value of FSI which is calculated by SRC

### 48.3.27 2SRC0\_m Wait Time Setting Register p (WATSRp\_2SRC0\_m) (m = 0, 1; p = 0, 1)

WATSRp\_2SRC0\_m is a 32-bit readable/writable register that sets the wait time for data input to the SRC.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	WTIME							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WTIME															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 0	WTIME	All 0	R/W	These bits set the wait time when WATMD bit of SRCrRp_2SRC0_m register is 1. After the INIT bit of SRCIRp_2SRC0_m register is set to 0, 0 data is stored into the SRC until the number of input data reaches to the setting value of these bits. The data input before the INIT bit is set to 0 is discarded. After the number of input data reaches to the setting value, input data is stored into the SRC.

### 48.3.28 2SRC0\_m Event Mask Register p (SEVMRp\_2SRC0\_m) (m = 0, 1; p = 0, 1)

SEVMRp\_2SRC0\_m is a 32-bit readable/writable register that enables or disables SCUAI<sub>m</sub> interrupt requests. The name of the interrupt source corresponding to the interrupt request from SEVMRp\_2SRC0\_0 is SCUAI0 and that corresponding to the interrupt request from SEVMRp\_2SRC0\_1 is SCUAI1.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	EVMUF	EVMOF	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	EVM WAIT	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	EVMUF	0	R/W	SRC Underflow Mask This bit sets whether to mask the interrupt request from the EVCUF bit in the SEVCRp_2SRC0_m register. 0: Disables interrupts. 1: Enables interrupts.
30	EVMOF	0	R/W	SRC Overflow Mask This bit sets whether to mask the EVCOF bit in the SEVCRp_2SRC0_m register. 0: Disables interrupts. 1: Enables interrupts.
29 to 15	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
14	EVMWAIT	0	R/W	SRC Wait Time Mask This bit sets whether to mask the interrupt request from the EVCWAIT bit of SEVCRp_2SRC0_m register. 0: Disables interrupts. 1: Enables interrupts.
13 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.



### 48.3.29 2SRC0\_m Event Clear Register p (SEVCRp\_2SRC0\_m) (m = 0, 1; p = 0, 1)

SEVCRp\_2SRC0\_m is a 32-bit readable/writable register that clears SCUAI<sub>m</sub> interrupt requests. When an interrupt event is generated, the relevant bit in this register is automatically set to 1 and 1 is retained until 0 is written to that bit.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	EVCUF	EVCOF	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/(W)*1	R/(W)*1	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	EVCWAIT	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/(W)*1	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	EVCUF	0	R/(W)*1	<b>SRC Underflow Clear</b> This is an interrupt flag to indicate whether INDATA RAM underflow occurs. When it occurs, this bit is set to 1. 0: Clears the flag. 1: Retains the flag.
30	EVCOF	0	R/(W)*1	<b>SRC Overflow Clear</b> This is an interrupt flag to indicate whether INDATA RAM overflow occurs. When it occurs, this bit is set to 1. 0: Clears the flag. 1: Retains the flag.
29 to 15	—	All 0	R	<b>Reserved</b> These bits are always read as 0. The write value should always be 0.
14	EVCWAIT	0	R/(W)*1	<b>SRC Wait Time Clear</b> This is an interrupt flag to indicate whether SRC has started processing after the wait time. 0: Clears the flag. 1: Retains the flag.
13 to 0	—	All 0	R	<b>Reserved</b> These bits are always read as 0. The write value should always be 0.

Note 1. Readable/writable. When 0 is written, the bit is initialized. Writing 1 to the bit is ignored. Write 0 only to each bit corresponding to the interrupt source to be cleared; write 1 to other bits.

### 48.3.30 2SRC0\_m Initialization Register RIF (SRCIRR\_2SRC0\_m) (m = 0, 1)

SRCIRR\_2SRC0\_m is a 32-bit readable/writable register that initializes COEF-ROMIF which is mounted on 2SRC0\_m.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	INIT
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	INIT	1	R/W	Initialization When this bit is set to 1, the COEF-ROMIF internal circuit is initialized. To cancel the initialization, set this bit to 0 with it set to 1. 0: Processing State 1: Initialization (sets the initial setting of other registers)

### 48.3.31 DVU0\_n Initialization Register (DVUIR\_DVU0\_n) (n = 0, 1, 2, 3)

DVUIR\_DVU0\_n is a 32-bit readable/writable register that initializes the DVU internal circuit.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	INIT
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	INIT	1	R/W	Initialization When this bit is set to 1, the DVU internal circuit is initialized. To cancel the initialization, set this bit to 0 with it set to 1. 0: Processing State 1: Initialization (sets the initial setting of other registers)

### 48.3.32 DVU0\_n Audio Information Register (VADIR\_DVU0\_n) (n = 0, 1, 2, 3)

VADIR\_DVU0\_n is a 32-bit readable/writable register that sets the bit length of data and the number of channels.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	OTBL				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	CHNUM			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
20 to 16	OTBL	All 0	R/W	Bit Length of Output Audio Data 00000: 24 bits 00001 to 00111: Reserved 01000: 16 bits 01001 to 11111: Reserved
15 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	CHNUM	All 0	R/W	Number of Channels These bits set the number of channels. 0000: Zero (None) 0001: 1 channel 0010: 2 channels 0011: Reserved 0100: 4 channels 0101: Reserved 0110: 6 channels 0111: Reserved 1000: 8 channels 1001 to 1111: Reserved

### 48.3.33 DVU0\_n Bypass Register (DVUBR\_DVU0\_n) (n = 0, 1, 2, 3)

DVUBR\_DVU0\_n is a 32-bit readable/writable register that sets the bypass mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	BY PASS
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	BYPASS	0	R/W	Bypass Mode This bit controls the data pass of DVU. 0: DVU function is used. Input data is processed by DVU and then the result data is connected to output data. 1: DVU function is not used (bypass mode). Input data is directly connected to output data.

### 48.3.34 DVU0\_n Control Register (DVUCR\_DVU0\_n) (n = 0, 1, 2, 3)

DVUCR\_DVU0\_n is a 32-bit readable/writable register that selects the DVU mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	VVMD	-	-	-	VRMD	-	-	-	ZCMD
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	VVMD	0	R/W	Digital Volume Mode Select This bit selects the digital volume function. 0: Disables the digital volume function. 1: Enables the digital volume function.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	VRMD	0	R/W	Volume Ramp Mode Select This bit selects the volume ramp function. 0: Disables the volume ramp function. 1: Enables the volume ramp function.
3 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	ZCMD	0	R/W	Zero Cross Mute Mode Select This bit selects the zero cross mute function. 0: Disables the zero cross mute function. 1: Enables the zero cross mute function.

### 48.3.35 DVU0\_n Zero Cross Mute Control Register (ZCMCR\_DVU0\_n) (n = 0, 1, 2, 3)

ZCMCR\_DVU0\_n is a 32-bit readable/writable register that controls operation of the zero cross mute function for each channel.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	ZCEN7	ZCEN6	ZCEN5	ZCEN4	ZCEN3	ZCEN2	ZCEN1	ZCEN0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	ZCEN7	0	R/W	Zero Cross Mute Enable for Channel 7 This bit controls the operation of the zero cross mute function for channel 7. 0: Disables the operation of the zero cross mute function. 1: Enables the operation the zero cross mute function.
6	ZCEN6	0	R/W	Zero Cross Mute Enable for Channel 6 This bit controls the operation of the zero cross mute function for channel 6. 0: Disables the operation of the zero cross mute function. 1: Enables the operation the zero cross mute function.
5	ZCEN5	0	R/W	Zero Cross Mute Enable for Channel 5 This bit controls the operation of the zero cross mute function for channel 5. 0: Disables the operation of the zero cross mute function. 1: Enables the operation the zero cross mute function.
4	ZCEN4	0	R/W	Zero Cross Mute Enable for Channel 4 This bit controls the operation of the zero cross mute function for channel 4. 0: Disables the operation of the zero cross mute function. 1: Enables the operation the zero cross mute function.
3	ZCEN3	0	R/W	Zero Cross Mute Enable for Channel 3 This bit controls the operation of the zero cross mute function for channel 3. 0: Disables the operation of the zero cross mute function. 1: Enables the operation the zero cross mute function.
2	ZCEN2	0	R/W	Zero Cross Mute Enable for Channel 2 This bit controls the operation of the zero cross mute function for channel 2. 0: Disables the operation of the zero cross mute function. 1: Enables the operation the zero cross mute function.
1	ZCEN1	0	R/W	Zero Cross Mute Enable for Channel 1 This bit controls the operation of the zero cross mute function for channel 1. 0: Disables the operation of the zero cross mute function. 1: Enables the operation the zero cross mute function.
0	ZCEN0	0	R/W	Zero Cross Mute Enable for Channel 0 This bit controls the operation of the zero cross mute function for channel 0. 0: Disables the operation of the zero cross mute function. 1: Enables the operation the zero cross mute function.

### 48.3.36 DVU0\_n Volume Ramp Control Register (VRCTR\_DVU0\_n) (n = 0, 1, 2, 3)

VRCTR\_DVU0\_n is a 32-bit readable/writable register that controls operation of the volume ramp function for each channel.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	VREN7	VREN6	VREN5	VREN4	VREN3	VREN2	VREN1	VREN0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	VREN7	0	R/W	Volume Ramp Enable for Channel 7 This bit controls the operation of the volume ramp function for channel 7. 0: Disables the operation of the volume ramp function. 1: Enables the operation of the volume ramp function.
6	VREN6	0	R/W	Volume Ramp Enable for Channel 6 This bit controls the operation of the volume ramp function for channel 6. 0: Disables the operation of the volume ramp function. 1: Enables the operation of the volume ramp function.
5	VREN5	0	R/W	Volume Ramp Enable for Channel 5 This bit controls the operation of the volume ramp function for channel 5. 0: Disables the operation of the volume ramp function. 1: Enables the operation of the volume ramp function.
4	VREN4	0	R/W	Volume Ramp Enable for Channel 4 This bit controls the operation of the volume ramp function for channel 4. 0: Disables the operation of the volume ramp function. 1: Enables the operation of the volume ramp function.
3	VREN3	0	R/W	Volume Ramp Enable for Channel 3 This bit controls the operation of the volume ramp function for channel 3. 0: Disables the operation of the volume ramp function. 1: Enables the operation of the volume ramp function.
2	VREN2	0	R/W	Volume Ramp Enable for Channel 2 This bit controls the operation of the volume ramp function for channel 2. 0: Disables the operation of the volume ramp function. 1: Enables the operation of the volume ramp function.
1	VREN1	0	R/W	Volume Ramp Enable for Channel 1 This bit controls the operation of the volume ramp function for channel 1. 0: Disables the operation of the volume ramp function. 1: Enables the operation of the volume ramp function.
0	VREN0	0	R/W	Volume Ramp Enable for Channel 0 This bit controls the operation of the volume ramp function for channel 0. 0: Disables the operation of the volume ramp function. 1: Enables the operation of the volume ramp function.

### 48.3.37 DVU0\_n Volume Ramp Period Register (VRPDR\_DVU0\_n) (n = 0, 1, 2, 3)

VRPDR\_DVU0\_n is a 32-bit readable/writable register that sets the ramp period of volume ramp.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	VRPDUP				-	-	-	VRPDDW					
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 13	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
12 to 8	VRPDUP	All 0	R/W	Volume Ramp Period for Volume Up 00000: 1 [sample] (128 dB/1 step) 00001: 2 [sample] (64 dB/1 step) 00010: 4 [sample] (32 dB/1 step) 00011: 8 [sample] (16 dB/1 step) 00100: 16 [sample] (8 dB/1 step) 00101: 32 [sample] (4 dB/1 step) 00110: 64 [sample] (2 dB/1 step) 00111: 128 [sample] (1 dB/1 step) 01000: 256 [sample] (0.5 dB/1 step) 01001: 512 [sample] (0.25 dB/1 step) 01010: 1024 [sample] (0.125 dB/1 step) 01011: 2048 [sample] (0.125 dB/2 steps) 01100: 4096 [sample] (0.125 dB/4 steps) 01101: 8192 [sample] (0.125 dB/8 steps) 01110: 16384 [sample] (0.125 dB/16 steps) 01111: 32768 [sample] (0.125 dB/32 steps) 10000: 65536 [sample] (0.125 dB/64 steps) 10001: 131072 [sample] (0.125 dB/128 steps) 10010: 262144 [sample] (0.125 dB/256 steps) 10011: 524288 [sample] (0.125 dB/512 steps) 10100: 1048576 [sample] (0.125 dB/1024 steps) 10101: 2097152 [sample] (0.125 dB/2048 steps) 10110: 4194304 [sample] (0.125 dB/4096 steps) 10111: 8388608 [sample] (0.125 dB/8192 steps) 11000 to 11111: Reserved
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
4 to 0	VRPDDW	0	R/W	Volume Ramp Period for Volume Down 00000: 1 [sample] (-128 dB/1 step) 00001: 2 [sample] (-64 dB/1 step) 00010: 4 [sample] (-32 dB/1 step) 00011: 8 [sample] (-16 dB/1 step) 00100: 16 [sample] (-8 dB/1 step) 00101: 32 [sample] (-4 dB/1 step) 00110: 64 [sample] (-2 dB/1 step) 00111: 128 [sample] (-1 dB/1 step) 01000: 256 [sample] (-0.5 dB/1 step) 01001: 512 [sample] (-0.25 dB/1 step) 01010: 1024 [sample] (-0.125 dB/1 step) 01011: 2048 [sample] (-0.125 dB/2 steps) 01100: 4096 [sample] (-0.125 dB/4 steps) 01101: 8192 [sample] (-0.125 dB/8 steps) 01110: 16384 [sample] (-0.125 dB/16 steps) 01111: 32768 [sample] (-0.125 dB/32 steps) 10000: 65536 [sample] (-0.125 dB/64 steps) 10001: 131072 [sample] (-0.125 dB/128 steps) 10010: 262144 [sample] (-0.125 dB/256 steps) 10011: 524288 [sample] (-0.125 dB/512 steps) 10100: 1048576 [sample] (-0.125 dB/1024 steps) 10101: 2097152 [sample] (-0.125 dB/2048 steps) 10110: 4194304 [sample] (-0.125 dB/4096 steps) 10111: 8388608 [sample] (-0.125 dB/8192 steps) 11000 to 11111: Reserved

### 48.3.38 DVU0\_n Volume Ramp Decibel Register (VRDBR\_DVU0\_n) (n = 0, 1, 2, 3)

VRDBR\_DVU0\_n is a 32-bit readable/writable register that sets the gain level (decibel units) of volume ramp.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16		
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R		
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0		
	-	-	-	-	-	-	VRDB										-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W		

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
9 to 0	VRDB	All 0	R/W	Decibel of Volume Ramp These bits set the decibel (gain level) of volume ramp. The value can be selected from 1024 points in the range of 0 dB to -∞ dB at intervals of 0.125 dB.

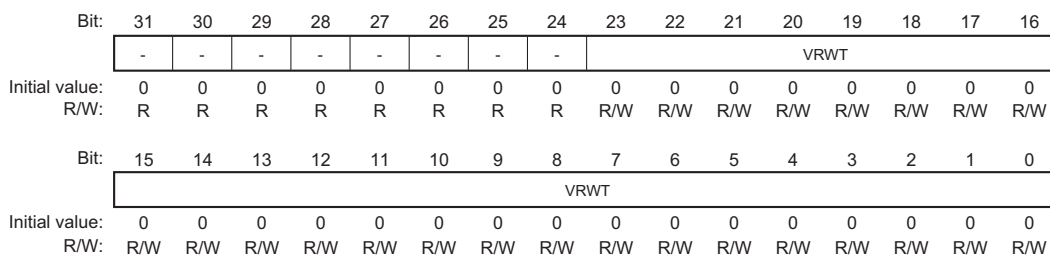
  

Setting Value	Time	Gain[dB]	Setting Value	Time	Gain[dB]
0x000	1	0	.....	.....	.....
.....	.....	.....	0x091	0.125	-18.125
0x031	0.5	-6.125	.....	.....	.....
.....	.....	.....	0x3FE	4.1x10 <sup>-7</sup>	-127.75
0x061	0.25	-12.125	0x3FF	0 (Mute)	-∞



### 48.3.39 DVU0\_n Volume Ramp Wait Time Register (VRWTR\_DVU0\_n) (n = 0, 1, 2, 3)

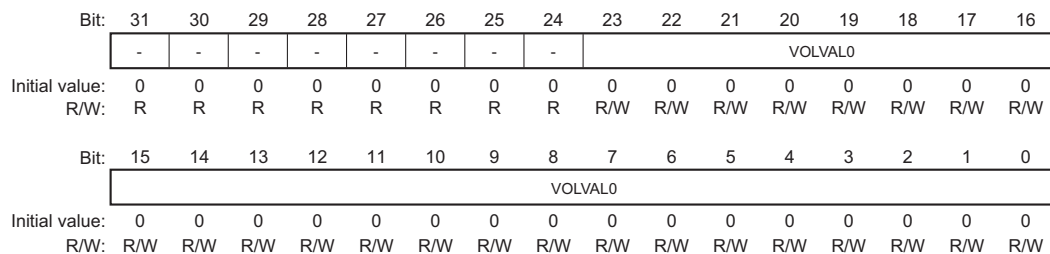
VRWTR\_DVU0\_n is a 32-bit readable/writable register that sets the wait time before executing the volume ramp function.



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 0	VRWT	All 0	R/W	Volume Ramp Wait Time These bits set the wait time before starting the volume ramp function when the setting of VRDBR register is changed. If the internal counter of the DVU logic reached to the value of these bits, the volume ramp function starts operating to change the volume to the value changed in the VRDBR register as a target.

### 48.3.40 DVU0\_n Volume Value Setting 0 Register (VOL0R\_DVU0\_n) (n = 0, 1, 2, 3)

VOL0R\_DVU0\_n is a 32-bit readable/writable register that sets the digital volume value of channel 0.



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 0	VOLVAL0	All 0	R/W	Digital Volume Value for Channel 0 These bits set the digital volume of channel 0. The maximum value is octupled gain (18 dB) and the minimum value is 0 (-∞ dB). VOLVAL0[23] : Sign bit VOLVAL0[22:20] : Integer bits VOLVAL0[19:0] : Decimal bits

Setting Value	Time	Gain[dB]	Setting Value	Time	Gain[dB]
0x7F_FFFF	8	18	0x08_0000	0.5	-6
...	...	...	...	...	...
0x10_0000	1	0	0x00_0001	9.5x10 <sup>-7</sup>	-120
...	...	...	0x00_0000	0	-∞

### 48.3.41 DVU0\_n Volume Value Setting 1 Register (VOL1R\_DVU0\_n) (n = 0, 1, 2, 3)

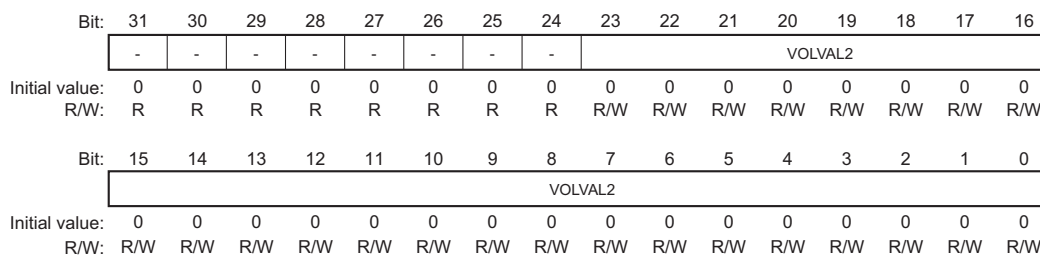
VOL1R\_DVU0\_n is a 32-bit readable/writable register that sets the digital volume value of channel 1.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	VOLVAL1							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VOLVAL1															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description																														
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.																														
23 to 0	VOLVAL1	All 0	R/W	Digital Volume Value for Channel 1 These bits set the digital volume of channel 1. The maximum value is octupled gain (18 dB) and the minimum value is 0 (-∞ dB). VOLVAL1[23] : Sign bit VOLVAL1[22:20] : Integer bits VOLVAL1[19:0] : Decimal bits																														
				<table border="1"> <thead> <tr> <th>Setting Value</th> <th>Time</th> <th>Gain[dB]</th> <th>Setting Value</th> <th>Time</th> <th>Gain[dB]</th> </tr> </thead> <tbody> <tr> <td>0x7F_FFFF</td> <td>8</td> <td>18</td> <td>0x08_0000</td> <td>0.5</td> <td>-6</td> </tr> <tr> <td>...</td> <td>...</td> <td>...</td> <td>...</td> <td>...</td> <td>...</td> </tr> <tr> <td>0x10_0000</td> <td>1</td> <td>0</td> <td>0x00_0001</td> <td>9.5x10<sup>-7</sup></td> <td>-120</td> </tr> <tr> <td>...</td> <td>...</td> <td>...</td> <td>0x00_0000</td> <td>0</td> <td>-∞</td> </tr> </tbody> </table>	Setting Value	Time	Gain[dB]	Setting Value	Time	Gain[dB]	0x7F_FFFF	8	18	0x08_0000	0.5	-6	...	...	...	...	...	...	0x10_0000	1	0	0x00_0001	9.5x10 <sup>-7</sup>	-120	...	...	...	0x00_0000	0	-∞
Setting Value	Time	Gain[dB]	Setting Value	Time	Gain[dB]																													
0x7F_FFFF	8	18	0x08_0000	0.5	-6																													
...	...	...	...	...	...																													
0x10_0000	1	0	0x00_0001	9.5x10 <sup>-7</sup>	-120																													
...	...	...	0x00_0000	0	-∞																													

### 48.3.42 DVU0\_n Volume Value Setting 2 Register (VOL2R\_DVU0\_n) (n = 0, 1, 2, 3)

VOL2R\_DVU0\_n is a 32-bit readable/writable register that sets the digital volume value of channel 2.



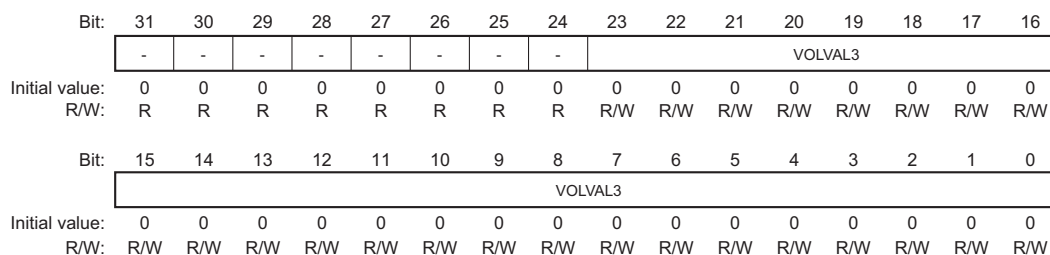
Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 0	VOLVAL2	All 0	R/W	Digital Volume Value for Channel 2 These bits set the digital volume of channel 2. The maximum value is octupled gain (18 dB) and the minimum value is 0 (-∞ dB). VOLVAL2[23] : Sign bit VOLVAL2[22:20] : Integer bits VOLVAL2[19:0] : Decimal bits

Setting Value	Time	Gain[dB]	Setting Value	Time	Gain[dB]
0x7F_FFFF	8	18	0x08_0000	0.5	-6
...	...	...	...	...	...
0x10_0000	1	0	0x00_0001	9.5x10 <sup>-7</sup>	-120
...	...	...	0x00_0000	0	-∞

### 48.3.43 DVU0\_n Volume Value Setting 3 Register (VOL3R\_DVU0\_n) (n = 0, 1, 2, 3)

VOL3R\_DVU0\_n is a 32-bit readable/writable register that sets the digital volume value of channel 3.



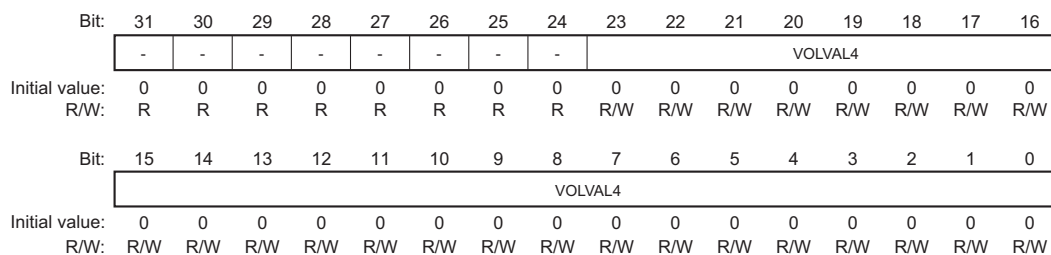
Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 0	VOLVAL3	All 0	R/W	Digital Volume Value for Channel 3 These bits set the digital volume of channel 3. The maximum value is octupled gain (18 dB) and the minimum value is 0 (-∞ dB). VOLVAL3[23] : Sign bit VOLVAL3[22:20] : Integer bits VOLVAL3[19:0] : Decimal bits

Setting Value	Time	Gain[dB]	Setting Value	Time	Gain[dB]
0x7F_FFFF	8	18	0x08_0000	0.5	-6
...	...	...	...	...	...
0x10_0000	1	0	0x00_0001	9.5x10 <sup>-7</sup>	-120
...	...	...	0x00_0000	0	-∞

### 48.3.44 DVU0\_n Volume Value Setting 4 Register (VOL4R\_DVU0\_n) (n = 0, 1, 2, 3)

VOL4R\_DVU0\_n is a 32-bit readable/writable register that sets the digital volume value of channel 4.



Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 0	VOLVAL4	All 0	R/W	Digital Volume Value for Channel 4 These bits set the digital volume of channel 4. The maximum value is octupled gain (18 dB) and the minimum value is 0 (-∞ dB). VOLVAL4[23] : Sign bit VOLVAL4[22:20] : Integer bits VOLVAL4[19:0] : Decimal bits

Setting Value	Time	Gain[dB]	Setting Value	Time	Gain[dB]
0x7F_FFFF	8	18	0x08_0000	0.5	-6
...	...	...	...	...	...
0x10_0000	1	0	0x00_0001	9.5x10 <sup>-7</sup>	-120
...	...	...	0x00_0000	0	-∞

### 48.3.45 DVU0\_n Volume Value Setting 5 Register (VOL5R\_DVU0\_n) (n = 0, 1, 2, 3)

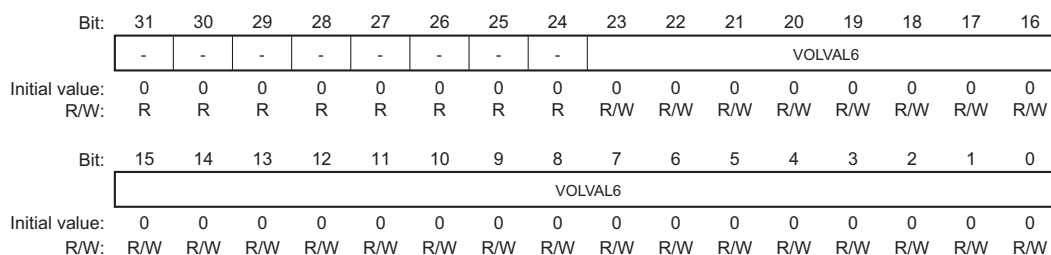
VOL5R\_DVU0\_n is a 32-bit readable/writable register that sets the digital volume value of channel 5.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	VOLVAL5							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VOLVAL5															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description																														
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.																														
23 to 0	VOLVAL5	All 0	R/W	Digital Volume Value for Channel 5 These bits set the digital volume of channel 5. The maximum value is octupled gain (18 dB) and the minimum value is 0 (-∞ dB). VOLVAL5[23] : Sign bit VOLVAL5[22:20] : Integer bits VOLVAL5[19:0] : Decimal bits																														
				<table border="1"> <thead> <tr> <th>Setting Value</th> <th>Time</th> <th>Gain[dB]</th> <th>Setting Value</th> <th>Time</th> <th>Gain[dB]</th> </tr> </thead> <tbody> <tr> <td>0x7F_FFFF</td> <td>8</td> <td>18</td> <td>0x08_0000</td> <td>0.5</td> <td>-6</td> </tr> <tr> <td>...</td> <td>...</td> <td>...</td> <td>...</td> <td>...</td> <td>...</td> </tr> <tr> <td>0x10_0000</td> <td>1</td> <td>0</td> <td>0x00_0001</td> <td>9.5x10<sup>-7</sup></td> <td>-120</td> </tr> <tr> <td>...</td> <td>...</td> <td>...</td> <td>0x00_0000</td> <td>0</td> <td>-∞</td> </tr> </tbody> </table>	Setting Value	Time	Gain[dB]	Setting Value	Time	Gain[dB]	0x7F_FFFF	8	18	0x08_0000	0.5	-6	...	...	...	...	...	...	0x10_0000	1	0	0x00_0001	9.5x10 <sup>-7</sup>	-120	...	...	...	0x00_0000	0	-∞
Setting Value	Time	Gain[dB]	Setting Value	Time	Gain[dB]																													
0x7F_FFFF	8	18	0x08_0000	0.5	-6																													
...	...	...	...	...	...																													
0x10_0000	1	0	0x00_0001	9.5x10 <sup>-7</sup>	-120																													
...	...	...	0x00_0000	0	-∞																													

### 48.3.46 DVU0\_n Volume Value Setting 6 Register (VOL6R\_DVU0\_n) (n = 0, 1, 2, 3)

VOL6R\_DVU0\_n is a 32-bit readable/writable register that sets the digital volume value of channel 6.



Bit	Bit Name	Initial Value	R/W	Description																														
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.																														
23 to 0	VOLVAL6	All 0	R/W	Digital Volume Value for Channel 6 These bits set the digital volume of channel 6. The maximum value is octupled gain (18 dB) and the minimum value is 0 (-∞ dB). VOLVAL6[23] : Sign bit VOLVAL6[22:20] : Integer bits VOLVAL6[19:0] : Decimal bits																														
				<table border="1"> <thead> <tr> <th>Setting Value</th> <th>Time</th> <th>Gain[dB]</th> <th>Setting Value</th> <th>Time</th> <th>Gain[dB]</th> </tr> </thead> <tbody> <tr> <td>0x7F_FFFF</td> <td>8</td> <td>18</td> <td>0x08_0000</td> <td>0.5</td> <td>-6</td> </tr> <tr> <td>...</td> <td>...</td> <td>...</td> <td>...</td> <td>...</td> <td>...</td> </tr> <tr> <td>0x10_0000</td> <td>1</td> <td>0</td> <td>0x00_0001</td> <td>9.5x10<sup>-7</sup></td> <td>-120</td> </tr> <tr> <td>...</td> <td>...</td> <td>...</td> <td>0x00_0000</td> <td>0</td> <td>-∞</td> </tr> </tbody> </table>	Setting Value	Time	Gain[dB]	Setting Value	Time	Gain[dB]	0x7F_FFFF	8	18	0x08_0000	0.5	-6	...	...	...	...	...	...	0x10_0000	1	0	0x00_0001	9.5x10 <sup>-7</sup>	-120	...	...	...	0x00_0000	0	-∞
Setting Value	Time	Gain[dB]	Setting Value	Time	Gain[dB]																													
0x7F_FFFF	8	18	0x08_0000	0.5	-6																													
...	...	...	...	...	...																													
0x10_0000	1	0	0x00_0001	9.5x10 <sup>-7</sup>	-120																													
...	...	...	0x00_0000	0	-∞																													

### 48.3.47 DVU0\_n Volume Value Setting 7 Register (VOL7R\_DVU0\_n) (n = 0, 1, 2, 3)

VOL7R\_DVU0\_n is a 32-bit readable/writable register that sets the digital volume value of channel 7.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	VOLVAL7							
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VOLVAL7															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 0	VOLVAL7	All 0	R/W	Digital Volume Value for Channel7 These bits set the digital volume of channel 7. The maximum value is octupled gain (18 dB) and the minimum value is 0 (-∞ dB). VOLVAL7[23] : Sign bit VOLVAL7[22:20] : Integer bits VOLVAL7[19:0] : Decimal bits

Setting Value	Time	Gain[dB]	Setting Value	Time	Gain[dB]
0x7F_FFFF	8	18	0x08_0000	0.5	-6
...	...	...	...	...	...
0x10_0000	1	0	0x00_0001	9.5x10 <sup>-7</sup>	-120
...	...	...	0x00_0000	0	-∞

### 48.3.48 DVU0\_n Enable Register (DVUER\_DVU0\_n) (n = 0, 1, 2, 3)

DVUER\_DVU0\_n is a 32-bit readable/writable register that enables or disables the DVU register settings.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	DVUEN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	DVUEN	0	R/W	DVU Register Setting Enable This bit controls the setting of DVU registers (ZCMCR, VRCTR, VRPDR, VRDBR, VOL0R, VOL1R, VOL2R, VOL3R, VOL4R, VOL5R, VOL6R, and VOL7R). 0: Disables the setting of DVU registers. 1: Enables the setting of DVU registers.



### 48.3.49 DVU0\_n Status Register (DVUSR\_DVU0\_n) (n = 0, 1, 2, 3)

DVUSR\_DVU0\_n is a 32-bit read-only register that indicates the status of zero cross mute and volume ramp.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	ZSTS7	ZSTS6	ZSTS5	ZSTS4	ZSTS3	ZSTS2	ZSTS1	ZSTS0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	VRSTS		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23	ZSTS7	0	R	Zero Cross Mute Status of Channel 7 This bit indicates the zero cross mute status of channel 7. 0: Not in mute status 1: Mute status
22	ZSTS6	0	R	Zero Cross Mute Status of Channel 6 This bit indicates the zero cross mute status of channel 6. 0: Not in mute status 1: Mute status
21	ZSTS5	0	R	Zero Cross Mute Status of Channel 5 This bit indicates the zero cross mute status of channel 5. 0: Not in mute status 1: Mute status
20	ZSTS4	0	R	Zero Cross Mute Status of Channel 4 This bit indicates the zero cross mute status of channel 4. 0: Not in mute status 1: Mute status
19	ZSTS3	0	R	Zero Cross Mute Status of Channel 3 This bit indicates the zero cross mute status of channel 3. 0: Not in mute status 1: Mute status
18	ZSTS2	0	R	Zero Cross Mute Status of Channel 2 This bit indicates the zero cross mute status of channel 2. 0: Not in mute status 1: Mute status
17	ZSTS1	0	R	Zero Cross Mute Status of Channel 1 This bit indicates the zero cross mute status of channel 1. 0: Not in mute status 1: Mute status
16	ZSTS0	0	R	Zero Cross Mute Status of Channel 0 This bit indicates the zero cross mute status of channel 0. 0: Not in mute status 1: Mute status
15 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2 to 0	VRSTS	All 0	R	Volume Ramp Status These bits indicate the volume ramp status. 000: Mute status 001: Volume ramp down 010: Volume ramp up 011: Level of volume ramp is in the state of the value of VRDBR register 100: Volume of input data is maintained (Volume is not multiplied). 101 to 111: Reserved

### 48.3.50 DVU0\_n Event Mask Register (VEVMR\_DVU0\_n) (n = 0, 1, 2, 3)

VEVMR\_DVU0\_n is a 32-bit readable/writable register that enables or disables SCUDVIn interrupt requests.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	VEVMZCM7	VEVMZCM6	VEVMZCM5	VEVMZCM4	VEVMZCM3	VEVMZCM2	VEVMZCM1	VEVMZCM0	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VEVMVR	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	VEVMZCM7	0	R/W	Zero Cross Mute Status Change Mask of Channel 7 This bit sets whether to mask the interrupt request from the VEVCZCM7 bit in the VEVCR_DVU0_n register. 0: Disables interrupts. 1: Enables interrupts.
30	VEVMZCM6	0	R/W	Zero Cross Mute Status Change Mask of Channel 6 This bit sets whether to mask the interrupt request from the VEVCZCM6 bit in the VEVCR_DVU0_n register. 0: Disables interrupts. 1: Enables interrupts.
29	VEVMZCM5	0	R/W	Zero Cross Mute Status Change Mask of Channel 5 This bit sets whether to mask the interrupt request from the VEVCZCM5 bit in the VEVCR_DVU0_n register. 0: Disables interrupts. 1: Enables interrupts.
28	VEVMZCM4	0	R/W	Zero Cross Mute Status Change Mask of Channel 4 This bit sets whether to mask the interrupt request from the VEVCZCM4 bit in the VEVCR_DVU0_n register. 0: Disables interrupts. 1: Enables interrupts.
27	VEVMZCM3	0	R/W	Zero Cross Mute Status Change Mask of Channel 3 This bit sets whether to mask the interrupt request from the VEVCZCM3 bit in the VEVCR_DVU0_n register. 0: Disables interrupts. 1: Enables interrupts.
26	VEVMZCM2	0	R/W	Zero Cross Mute Status Change Mask of Channel 2 This bit sets whether to mask the interrupt request from the VEVCZCM2 bit in the VEVCR_DVU0_n register. 0: Disables interrupts. 1: Enables interrupts.
25	VEVMZCM1	0	R/W	Zero Cross Mute Status Change Mask of Channel 1 This bit sets whether to mask the interrupt request from the VEVCZCM1 bit in the VEVCR_DVU0_n register. 0: Disables interrupts. 1: Enables interrupts.
24	VEVMZCM0	0	R/W	Zero Cross Mute Status Change Mask of Channel 0 This bit sets whether to mask the interrupt request from the VEVCZCM0 bit in the VEVCR_DVU0_n register. 0: Disables interrupts. 1: Enables interrupts.
23 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15	VEVMVR	0	R/W	Volume Ramp Status Change This bit sets whether to mask the interrupt request from the VEVCVR bit in the VEVCR_DVU0_n register. 0: Disables interrupts. 1: Enables interrupts.

---

Bit	Bit Name	Initial Value	R/W	Description
14 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

---

### 48.3.51 DVU0\_n Event Clear Register (VEVCR\_DVU0\_n) (n = 0, 1, 2, 3)

VEVCR\_DVU0\_n is a 32-bit readable/writable register that clears SCUDVIn interrupt requests. When an interrupt event is generated, the relevant bit in this register is automatically set to 1 and 1 is retained until 0 is written to that bit.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	VEVC ZCM7	VEVC ZCM6	VEVC ZCM5	VEVC ZCM4	VEVC ZCM3	VEVC ZCM2	VEVC ZCM1	VEVC ZCM0	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/(W) <sup>*1</sup>	R/(W) <sup>*1</sup>	R/(W) <sup>*1</sup>	R/(W) <sup>*1</sup>	R/(W) <sup>*1</sup>	R/(W) <sup>*1</sup>	R/(W) <sup>*1</sup>	R/(W) <sup>*1</sup>	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VEV CVR	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/(W) <sup>*1</sup>	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	VEVCZCM7	0	R/(W) <sup>*1</sup>	Zero Cross Mute Status Change Clear of Channel 7 This is an interrupt flag to indicate whether the zero cross mute status change of channel 7 occurs. When it occurs, this bit is set to 1. 0: Clears the flag. 1: Retains the flag.
30	VEVCZCM6	0	R/(W) <sup>*1</sup>	Zero Cross Mute Status Change Clear of Channel 6 This is an interrupt flag to indicate whether the zero cross mute status change of channel 6 occurs. When it occurs, this bit is set to 1. 0: Clears the flag. 1: Retains the flag.
29	VEVCZCM5	0	R/(W) <sup>*1</sup>	Zero Cross Mute Status Change Clear of Channel 5 This is an interrupt flag to indicate whether the zero cross mute status change of channel 5 occurs. When it occurs, this bit is set to 1. 0: Clears the flag. 1: Retains the flag.
28	VEVCZCM4	0	R/(W) <sup>*1</sup>	Zero Cross Mute Status Change Clear of Channel 4 This is an interrupt flag to indicate whether the zero cross mute status change of channel 4 occurs. When it occurs, this bit is set to 1. 0: Clears the flag. 1: Retains the flag.
27	VEVCZCM3	0	R/(W) <sup>*1</sup>	Zero Cross Mute Status Change Clear of Channel 3 This is an interrupt flag to indicate whether the zero cross mute status change of channel 3 occurs. When it occurs, this bit is set to 1. 0: Clears the flag. 1: Retains the flag.
26	VEVCZCM2	0	R/(W) <sup>*1</sup>	Zero Cross Mute Status Change Clear of Channel 2 This is an interrupt flag to indicate whether the zero cross mute status change of channel 2 occurs. When it occurs, this bit is set to 1. 0: Clears the flag. 1: Retains the flag.
25	VEVCZCM1	0	R/(W) <sup>*1</sup>	Zero Cross Mute Status Change Clear of Channel 1 This is an interrupt flag to indicate whether the zero cross mute status change of channel 1 occurs. When it occurs, this bit is set to 1. 0: Clears the flag. 1: Retains the flag.
24	VEVCZCM0	0	R/(W) <sup>*1</sup>	Zero Cross Mute Status Change Clear of Channel 0 This is an interrupt flag to indicate whether the zero cross mute status change of channel 0 occurs. When it occurs, this bit is set to 1. 0: Clears the flag. 1: Retains the flag.
23 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
15	VEVCVR	0	R/(W)*1	Volume Ramp Status Change Clear This is an interrupt flag to judge volume ramp mute occurs or not. When it occurs, this bit is set to 1. 0: Clears the flag. 1: Retains the flag.
14 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note 1. Readable/writable. When 0 is written, the bit is initialized. Writing 1 to the bit is ignored. Write 0 only to each bit corresponding to the interrupt source to be cleared; write 1 to other bits.

### 48.3.52 MIX0\_0 Initialization Register (MIXIR\_MIX0\_0)

MIXIR\_MIX0\_0 is a 32-bit readable/writable register that initializes the MIX internal circuit.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	INIT
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	INIT	1	R/W	Initialization When this bit is set to 1, the MIX internal circuit is initialized. To cancel the initialization, set this bit to 0 with it set to 1. 0: Processing State 1: Initialization (sets the initial setting of other registers)

### 48.3.53 MIX0\_0 Audio Information Register (MADIR\_MIX0\_0)

MADIR\_MIX0\_0 is a 32-bit readable/writable register that sets the number of channels.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	CHNUM			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	CHNUM	All 0	R/W	Number of Channels These bits set the number of channels. 0000: Zero (None) 0001: 1 channel 0010: 2 channels 0011: Reserved 0100: 4 channels 0101: Reserved 0110: 6 channels 0111: Reserved 1000: 8 channels 1001 to 1111: Reserved

### 48.3.54 MIX0\_0 Bypass Register (MIXBR\_MIX0\_0)

MIXBR\_MIX0\_0 is a 32-bit readable/writable register that sets the bypass mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	BPSYS
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	BY PASS
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 18	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
17, 16	BPSYS	All 0	R/W	Bypass System Select These bits select the input data output from the MIX module when the BYPASS bit is 1. 00: Input data of system A 01: Input data of system B 10: Input data of system C 11: Input data of system D
15 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	BYPASS	0	R/W	Bypass Mode This bit controls the data pass of MIX function. 0: MIX function is used. Input data is processed by MIX and then the result data is connected to output data. 1: MIX function is not used (bypass mode). Input data is connected to output data according to the BPSYS bits.

### 48.3.55 MIX0\_0 Mode Register (MIXMR\_MIX0\_0)

MIXMR\_MIX0\_0 is a 32-bit readable/writable register that selects the MIX mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	MIX MODE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	MIXMODE	0	R/W	MIX Mode This bit sets the mix mode. 0: Selects volume step mixer. 1: Selects volume ramp mixer.

### 48.3.56 MIX0\_0 Volume Period Register (MVPDR\_MIX0\_0)

MVPDR\_MIX0\_0 is a 32-bit readable/writable register that sets the ramp period of volume ramp.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	MXPDPUP				-	-	-	-	MXPDDW			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 12	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
11 to 8	MXPDPUP	0	R/W	Volume Up Period These bits set the decibel which changes per sample during volume up. This setting is used when the MIXMODE bit in the MIXMR register is set to 1. 0000: 128 dB/1 sample 0001: 64 dB/1 sample 0010: 32 dB/1 sample 0011: 16 dB/1 sample 0100: 8 dB/1 sample 0101: 4 dB/1 sample 0110: 2 dB/1 sample 0111: 1 dB/1 sample 1000: 0.5 dB/1 sample 1001: 0.25 dB/1 sample 1010: 0.125 dB/1 sample 1011 to 1111: Reserved
7 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	MXPDDW	0	R/W	Volume Down Period These bits set the decibel which changes per sample during volume down. This setting is used when the MIXMODE bit in the MIXMR register is set to 1. 0000: -128 dB/1 sample 0001: -64 dB/1 sample 0010: -32 dB/1 sample 0011: -16 dB/1 sample 0100: -8 dB/1 sample 0101: -4 dB/1 sample 0110: -2 dB/1 sample 0111: -1 dB/1 sample 1000: -0.5 dB/1 sample 1001: -0.25 dB/1 sample 1010: -0.125 dB/1 sample 1011 to 1111: Reserved



### 48.3.57 MIX0\_0 Decibel A Register (MDBAR\_MIX0\_0)

MDBAR\_MIX0\_0 is a 32-bit readable/writable register that sets the gain level (decibel units) of system A.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	MIXDBA									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description																																				
31 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.																																				
9 to 0	MIXDBA	All 0	R/W	Decibel of System A These bits set the decibel (gain level) of volume ramp. The value can be selected from 1024 points in the range of 0 dB to $-\infty$ dB at intervals of 0.125 dB.																																				
				<table border="1"> <thead> <tr> <th>Setting Value</th> <th>Time</th> <th>Gain[dB]</th> <th>Setting Value</th> <th>Time</th> <th>Gain[dB]</th> </tr> </thead> <tbody> <tr> <td>0x000</td> <td>1</td> <td>0</td> <td>.....</td> <td>.....</td> <td>.....</td> </tr> <tr> <td>.....</td> <td>.....</td> <td>.....</td> <td>0x091</td> <td>0.125</td> <td>-18.125</td> </tr> <tr> <td>0x031</td> <td>0.5</td> <td>-6.125</td> <td>.....</td> <td>.....</td> <td>.....</td> </tr> <tr> <td>.....</td> <td>.....</td> <td>.....</td> <td>0x3FE</td> <td><math>4.1 \times 10^{-7}</math></td> <td>-127.75</td> </tr> <tr> <td>0x061</td> <td>0.25</td> <td>-12.125</td> <td>0x3FF</td> <td>0 (Mute)</td> <td><math>-\infty</math></td> </tr> </tbody> </table>	Setting Value	Time	Gain[dB]	Setting Value	Time	Gain[dB]	0x000	1	0	.....	.....	.....	.....	.....	.....	0x091	0.125	-18.125	0x031	0.5	-6.125	.....	.....	.....	.....	.....	.....	0x3FE	$4.1 \times 10^{-7}$	-127.75	0x061	0.25	-12.125	0x3FF	0 (Mute)	$-\infty$
Setting Value	Time	Gain[dB]	Setting Value	Time	Gain[dB]																																			
0x000	1	0	.....	.....	.....																																			
.....	.....	.....	0x091	0.125	-18.125																																			
0x031	0.5	-6.125	.....	.....	.....																																			
.....	.....	.....	0x3FE	$4.1 \times 10^{-7}$	-127.75																																			
0x061	0.25	-12.125	0x3FF	0 (Mute)	$-\infty$																																			

### 48.3.58 MIX0\_0 Decibel B Register (MDBBR\_MIX0\_0)

MDBBR\_MIX0\_0 is a 32-bit readable/writable register that sets the gain level (decibel units) of system B.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	MIXDBB									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description																																				
9 to 0	MIXDBB	All 0	R/W	Decibel of System B These bits set the decibel (gain level) of volume ramp. The value can be selected from 1024 points in the range of 0 dB to $-\infty$ dB at intervals of 0.125 dB.																																				
				<table border="1"> <thead> <tr> <th>Setting Value</th> <th>Time</th> <th>Gain[dB]</th> <th>Setting Value</th> <th>Time</th> <th>Gain[dB]</th> </tr> </thead> <tbody> <tr> <td>0x000</td> <td>1</td> <td>0</td> <td>.....</td> <td>.....</td> <td>.....</td> </tr> <tr> <td>.....</td> <td>.....</td> <td>.....</td> <td>0x091</td> <td>0.125</td> <td>-18.125</td> </tr> <tr> <td>0x031</td> <td>0.5</td> <td>-6.125</td> <td>.....</td> <td>.....</td> <td>.....</td> </tr> <tr> <td>.....</td> <td>.....</td> <td>.....</td> <td>0x3FE</td> <td><math>4.1 \times 10^{-7}</math></td> <td>-127.75</td> </tr> <tr> <td>0x061</td> <td>0.25</td> <td>-12.125</td> <td>0x3FF</td> <td>0 (Mute)</td> <td><math>-\infty</math></td> </tr> </tbody> </table>	Setting Value	Time	Gain[dB]	Setting Value	Time	Gain[dB]	0x000	1	0	.....	.....	.....	.....	.....	.....	0x091	0.125	-18.125	0x031	0.5	-6.125	.....	.....	.....	.....	.....	.....	0x3FE	$4.1 \times 10^{-7}$	-127.75	0x061	0.25	-12.125	0x3FF	0 (Mute)	$-\infty$
Setting Value	Time	Gain[dB]	Setting Value	Time	Gain[dB]																																			
0x000	1	0	.....	.....	.....																																			
.....	.....	.....	0x091	0.125	-18.125																																			
0x031	0.5	-6.125	.....	.....	.....																																			
.....	.....	.....	0x3FE	$4.1 \times 10^{-7}$	-127.75																																			
0x061	0.25	-12.125	0x3FF	0 (Mute)	$-\infty$																																			

### 48.3.59 MIX0\_0 Decibel C Register (MDBCR\_MIX0\_0)

MDBCR\_MIX0\_0 is a 32-bit readable/writable register that sets the gain level (decibel units) of system C.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	-	-	-	-	-	-	MIXDBC									-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

Bit	Bit Name	Initial Value	R/W	Description																																				
31 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.																																				
9 to 0	MIXDBC	All 0	R/W	Decibel of System C These bits set the decibel (gain level) of volume ramp. The value can be selected from 1024 points in the range of 0 dB to $-\infty$ dB at intervals of 0.125 dB.																																				
				<table border="1"> <thead> <tr> <th>Setting Value</th> <th>Time</th> <th>Gain[dB]</th> <th>Setting Value</th> <th>Time</th> <th>Gain[dB]</th> </tr> </thead> <tbody> <tr> <td>0x000</td> <td>1</td> <td>0</td> <td>.....</td> <td>.....</td> <td>.....</td> </tr> <tr> <td>.....</td> <td>.....</td> <td>.....</td> <td>0x091</td> <td>0.125</td> <td>-18.125</td> </tr> <tr> <td>0x031</td> <td>0.5</td> <td>-6.125</td> <td>.....</td> <td>.....</td> <td>.....</td> </tr> <tr> <td>.....</td> <td>.....</td> <td>.....</td> <td>0x3FE</td> <td><math>4.1 \times 10^{-7}</math></td> <td>-127.75</td> </tr> <tr> <td>0x061</td> <td>0.25</td> <td>-12.125</td> <td>0x3FF</td> <td>0 (Mute)</td> <td><math>-\infty</math></td> </tr> </tbody> </table>	Setting Value	Time	Gain[dB]	Setting Value	Time	Gain[dB]	0x000	1	0	.....	.....	.....	.....	.....	.....	0x091	0.125	-18.125	0x031	0.5	-6.125	.....	.....	.....	.....	.....	.....	0x3FE	$4.1 \times 10^{-7}$	-127.75	0x061	0.25	-12.125	0x3FF	0 (Mute)	$-\infty$
Setting Value	Time	Gain[dB]	Setting Value	Time	Gain[dB]																																			
0x000	1	0	.....	.....	.....																																			
.....	.....	.....	0x091	0.125	-18.125																																			
0x031	0.5	-6.125	.....	.....	.....																																			
.....	.....	.....	0x3FE	$4.1 \times 10^{-7}$	-127.75																																			
0x061	0.25	-12.125	0x3FF	0 (Mute)	$-\infty$																																			

### 48.3.60 MIX0\_0 Decibel D Register (MDBDR\_MIX0\_0)

MDBDR\_MIX0\_0 is a 32-bit readable/writable register that sets the gain level (decibel units) of system D.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	MIXDBD									
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description																																				
31 to 10	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.																																				
9 to 0	MIXDBD	All 0	R/W	Decibel of System D These bits set the decibel (gain level) of volume ramp. The value can be selected from 1024 points in the range of 0 dB to -∞ dB at intervals of 0.125 dB.																																				
				<table border="1"> <thead> <tr> <th>Setting Value</th> <th>Time</th> <th>Gain[dB]</th> <th>Setting Value</th> <th>Time</th> <th>Gain[dB]</th> </tr> </thead> <tbody> <tr> <td>0x000</td> <td>1</td> <td>0</td> <td>.....</td> <td>.....</td> <td>.....</td> </tr> <tr> <td>.....</td> <td>.....</td> <td>.....</td> <td>0x091</td> <td>0.125</td> <td>-18.125</td> </tr> <tr> <td>0x031</td> <td>0.5</td> <td>-6.125</td> <td>.....</td> <td>.....</td> <td>.....</td> </tr> <tr> <td>.....</td> <td>.....</td> <td>.....</td> <td>0x3FE</td> <td>4.1x10<sup>-7</sup></td> <td>-127.75</td> </tr> <tr> <td>0x061</td> <td>0.25</td> <td>-12.125</td> <td>0x3FF</td> <td>0 (Mute)</td> <td>-∞</td> </tr> </tbody> </table>	Setting Value	Time	Gain[dB]	Setting Value	Time	Gain[dB]	0x000	1	0	.....	.....	.....	.....	.....	.....	0x091	0.125	-18.125	0x031	0.5	-6.125	.....	.....	.....	.....	.....	.....	0x3FE	4.1x10 <sup>-7</sup>	-127.75	0x061	0.25	-12.125	0x3FF	0 (Mute)	-∞
Setting Value	Time	Gain[dB]	Setting Value	Time	Gain[dB]																																			
0x000	1	0	.....	.....	.....																																			
.....	.....	.....	0x091	0.125	-18.125																																			
0x031	0.5	-6.125	.....	.....	.....																																			
.....	.....	.....	0x3FE	4.1x10 <sup>-7</sup>	-127.75																																			
0x061	0.25	-12.125	0x3FF	0 (Mute)	-∞																																			

### 48.3.61 MIX0\_0 Decibel Enable Register (MDBER\_MIX0\_0)

MDBER\_MIX0\_0 is a 32-bit readable/writable register that enable or disables the MIX decibel register settings.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	MIXDB EN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	MIXDBEN	0	R/W	MIX Decibel Enable This bit controls the decibel value that sets in MDBAR, MDBBR, MDBCR and MDBDR registers. 0: Disables the setting of decibel. 1: Enables the setting of decibel.

### 48.3.62 MIX0\_0 Status Register (MIXSR\_MIX0\_0)

MIXSR\_MIX0\_0 is a 32-bit read-only register that indicates the status of the MIX.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	MIXSTS
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	MIXSTS	All 0	R	MIX Status This bit specifies the status of ramp process. 00: Ramp process is unchanged. 01: Ramp down 10: Ramp up

### 48.3.63 Software Reset Register (SWRSR\_CIM)

SWRSR\_CIM is a 32-bit readable/writable register that initializes the SCUX internal circuit.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	SWRST
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	SWRST	1	R/W	Software Reset While this bit is 0, the SCUX internal circuits are put in the reset state. Registers except this register are reset. Therefore, they should be set again after the reset is canceled. 0: The SCUX is reset. 1: The SCUX is operating.

### 48.3.64 DMA Control Register (DMACR\_CIM)

DMACR\_CIM is a 32-bit readable/writable register that controls DMA execution.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	DMAM DFFU3	DMAM DFFU2	DMAM DFFU1	DMAM DFFU0	DMAM DFFD3	DMAM DFFD2	DMAM DFFD1	DMAM DFFD0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
23 to 16	—	All 0	R/W	Reserved The write value should always be 0.
15 to 8	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
7	DMAMDFFU3	0	R/W	FFU0_3 DMA Mode 0: Disables DMA transfer to DMATD3_CIM. 1: Enables DMA transfer to DMATD3_CIM. Note: When DMA transfer is enabled, disable packet requests sent to the interrupt controller. (Set the UEVMR_FFU0_3.UEVMRQ bit to 0.)
6	DMAMDFFU2	0	R/W	FFU0_2 DMA Mode 0: Disables DMA transfer to DMATD2_CIM. 1: Enables DMA transfer to DMATD2_CIM. Note: When DMA transfer is enabled, disable packet requests sent to the interrupt controller. (Set the UEVMR_FFU0_2.UEVMRQ bit to 0.)
5	DMAMDFFU1	0	R/W	FFU0_1 DMA Mode 0: Disables DMA transfer to DMATD1_CIM. 1: Enables DMA transfer to DMATD1_CIM. Note: When DMA transfer is enabled, disable packet requests sent to the interrupt controller. (Set the UEVMR_FFU0_1.UEVMRQ bit to 0.)
4	DMAMDFFU0	0	R/W	FFU0_0 DMA Mode 0: Disables DMA transfer to DMATD0_CIM. 1: Enables DMA transfer to DMATD0_CIM. Note: When DMA transfer is enabled, disable packet requests sent to the interrupt controller. (Set the UEVMR_FFU0_0.UEVMRQ bit to 0.)
3	DMAMDFFD3	0	R/W	FFD0_3 DMA Mode 0: Disables DMA transfer to DMATD3_CIM. 1: Enables DMA transfer to DMATD3_CIM. Note: When DMA transfer is enabled, disable packet requests sent to the interrupt controller. (Set the DEVMR_FFD0_3.DEVMRQ bit to 0.)
2	DMAMDFFD2	0	R/W	FFD0_2 DMA Mode 0: Disables DMA transfer to DMATD2_CIM. 1: Enables DMA transfer to DMATD2_CIM. Note: When DMA transfer is enabled, disable packet requests sent to the interrupt controller. (Set the DEVMR_FFD0_2.DEVMRQ bit to 0.)
1	DMAMDFFD1	0	R/W	FFD0_1 DMA Mode 0: Disables DMA transfer to DMATD1_CIM. 1: Enables DMA transfer to DMATD1_CIM. Note: When DMA transfer is enabled, disable packet requests sent to the interrupt controller. (Set the DEVMR_FFD0_1.DEVMRQ bit to 0.)
0	DMAMDFFD0	0	R/W	FFD0_0 DMA Mode 0: Disables DMA transfer to DMATD0_CIM. 1: Enables DMA transfer to DMATD0_CIM. Note: When DMA transfer is enabled, disable packet requests sent to the interrupt controller. (Set the DEVMR_FFD0_0.DEVMRQ bit to 0.)

### 48.3.65 DMA Transfer Register for FFD0\_n (DMATDn\_CIM) (n = 0, 1, 2, 3)

DMATDn\_CIM is a 32-bit register that stores the data to be transmitted to FFD0\_n. The data written to DMATDn\_CIM is automatically stored in FFD0\_n. When the number of data bytes in FFD0\_n becomes equal to or smaller than the value set in DRQSR\_FFD0\_0.SIZE, a data transfer request is issued. When the DMA controller is used (DMACR\_CIM.DMAMDFFDn = 1), a transfer request for the number of counts set in DRQSR\_FFD0\_0.SIZE is sent to the DMA controller. When the interrupt controller is used (DEVMR\_FFD0\_n.DEVMRQ = 1), data is written in this register for the number of counts set in DRQSR\_FFD0\_0.SIZE.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

In the SCUX internal block, audio data is handled in the 24-bit big endian format. When this register is accessed in 32-bit units, the lower eight bits are not used and only the upper 24 bits are used, as shown in Figure 48.2. When the SCUX is used in 16-bit mode, the valid data needs to be located at the MSB of audio data. Therefore, when this register is accessed in 16-bit units, the written data is located in the upper 16 bits of audio data and the lower eight bits are fixed to 0 in the CIM block. When the SCUX is used in 16-bit mode and this register is accessed in 32-bit units, since bits [15:8] of the data to be written is passed to the FFD block without change, 0 should be written to bits [15:8].

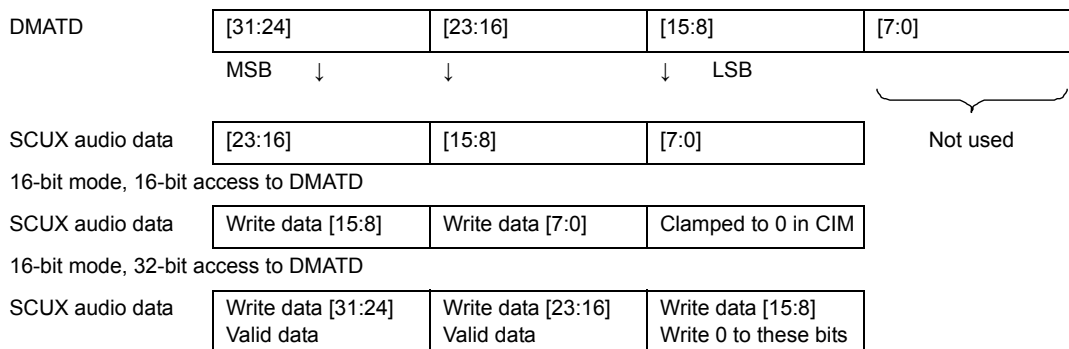


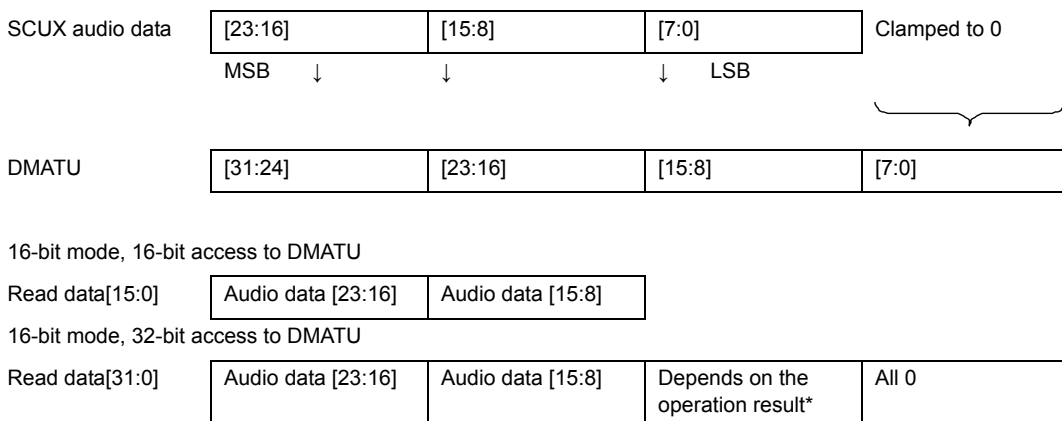
Figure 48.2 Data Alignment of DMATDn\_CIM

### 48.3.66 DMA Transfer Register for FFU0\_n (DMATUn\_CIM) (n = 0, 1, 2, 3)

DMATUn\_CIM is a 32-bit register that stores the data received from FFU0\_n. When data is read from DMATUn\_CIM, the next data is automatically transferred from FFU0\_n. When the number of data bytes in FFU0\_n becomes equal to or greater than the value set in URQSR\_FFU0\_n.SIZE, a data transfer request is issued. When the DMA controller is used (DMACR\_CIM.DMAMDFFU<sub>n</sub> = 1), a transfer request for the number of counts set in URQSR\_FFU0\_n.SIZE is sent to the DMA controller. When the interrupt controller is used (UEVMR\_FFU0\_n.UEVMRQ = 1), data is read from this register for the number of counts set in DRQSR\_FFD0\_0.SIZE.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Initial value:	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

When this register is accessed in 32-bit units, audio data is located in the upper 24 bits and the lower eight bits are always read as 0, as shown in Figure 48.3. When this register is accessed in 16-bit units, the upper 16 bits of audio data are read as valid data. When the SCUX is used in 16-bit mode and this register is accessed in 32-bit units, the lower eight bits are always read as 0 and bits [15:8] of the read data are read as a value dependent on the internal operation result in the SRC.



Note: \* All 1 when the internal operation result of the SRC is 0x7FFFFFFF; otherwise, all 0.  
 Even when used in 16-bit mode, operation is performed in 24-bit units.

Figure 48.3 Data Alignment of DMATUn\_CIM

### 48.3.67 SSI Route Select Register (SSIRSEL\_CIM)

SSIRSEL\_CIM is a 32-bit readable/writable register that selects the SSIF channel to be connected to each SRC during direct transfer with the SSIF module. This register setting has no meaning when a route between the FFD or FFU module is selected. The IPLSR\_IPC0\_n and OPSLR\_OPC0\_n registers are used to select connection between the SSIF or FFD/FFU module. The DVUBR\_DVU0\_n register is used to specify whether the DVU is used or not.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SISEL3		SISEL2		SISEL1		SISEL0		-	-	SOSEL5		-	-	SOSEL4	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	SOSEL3		-	-	SOSEL2		-	-	SOSEL1		-	-	SOSEL0	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	SISEL3	00	R/W	<p>SRC3 SSIF Input Select</p> <p>These bits select the input route of SRC3.</p> <p>00: Uses the input from SSIF2.</p> <p>01: Uses the input from SSIF5.</p> <p>10, 11: Setting prohibited</p> <p>Note: When the route from the FFD module is selected by the IPLSR_IPC0_3 register, this bit setting has no meaning.</p>
29, 28	SISEL2	00	R/W	<p>SRC2 SSIF Input Select</p> <p>These bits select the input route of SRC2.</p> <p>00: Uses the input from SSIF1.</p> <p>01: Uses the input from SSIF4.</p> <p>10, 11: Setting prohibited</p> <p>Note: When the route from the FFD module is selected by the IPLSR_IPC0_2 register, this bit setting has no meaning.</p>
27, 26	SISEL1	00	R/W	<p>SRC1 SSIF Input Select</p> <p>These bits select the input route of SRC1.</p> <p>00: Uses the input from SSIF3.</p> <p>01: Uses the input from SSIF345.*</p> <p>10, 11: Setting prohibited</p> <p>Note: When the route from the FFD module is selected by the IPLSR_IPC0_1 register, this bit setting has no meaning.</p> <p>* These bits can be set to 01 only when the SSI4PMD and SSI5PMD bits of the SSIPMD_CIM register are both set to 01 or 10.</p>
25, 24	SISEL0	00	R/W	<p>SRC0 SSIF Input Select</p> <p>These bits select the input route of SRC0.</p> <p>00: Uses the input from SSIF0.</p> <p>01: Uses the input from SSIF012.*</p> <p>10, 11: Setting prohibited</p> <p>Note: When the route from the FFD module is selected by the IPLSR_IPC0_0 register, this bit setting has no meaning.</p> <p>* These bits can be set to 01 only when the SSI1PMD and SSI2PMD bits of the SSIPMD_CIM register are both set to 01 or 10.</p>
23, 22	—	All 0	R	<p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p>



Bit	Bit Name	Initial Value	R/W	Description
21, 20	SOSEL5	00	R/W	<p>SSIF5 Output Select</p> <p>These bits select the output route to SSIF5.</p> <p>00: Uses the output from SRC3 (DVU0_3).<sup>*1</sup></p> <p>01: Uses the output from SRC1 (DVU0_1).<sup>*2, *4</sup></p> <p>10: Uses the output from SRC0 (DVU0_0).<sup>*3, *4</sup></p> <p>11: Uses the output from MIX.<sup>*4</sup></p> <p>Notes: 1. When the route to the FFU module is selected by the OPSLR_OPC0_3 register, this bit setting has no meaning.</p> <p>2. When the route to the FFU module is selected by the OPSLR_OPC0_1 register, this bit setting has no meaning.</p> <p>3. When the route to the FFU module is selected by the OPSLR_OPC0_0 register, this bit setting has no meaning.</p> <p>4. Valid only when SSIPMD_CIM.SSI345EN = 1.</p>
19, 18	—	All 0	R	<p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p>
17, 16	SOSEL4	00	R/W	<p>SSIF4 Output Select</p> <p>These bits select the output route to SSIF4.</p> <p>00: Uses the output from SRC2 (DVU0_2).<sup>*1</sup></p> <p>01: Uses the output from SRC1 (DVU0_1).<sup>*2, *4</sup></p> <p>10: Uses the output from SRC0 (DVU0_0).<sup>*3, *4</sup></p> <p>11: Uses the output from MIX.<sup>*4</sup></p> <p>Notes: 1. When the route to the FFU module is selected by the OPSLR_OPC0_2 register, this bit setting has no meaning.</p> <p>2. When the route to the FFU module is selected by the OPSLR_OPC0_1 register, this bit setting has no meaning.</p> <p>3. When the route to the FFU module is selected by the OPSLR_OPC0_0 register, this bit setting has no meaning.</p> <p>4. Valid only when SSIPMD_CIM.SSI345EN = 1.</p>
15, 14	—	All 0	R	<p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p>
13, 12	SOSEL3	00	R/W	<p>SSIF3 Output Select</p> <p>These bits select the output route to SSIF3.</p> <p>00: Uses the output from SRC1 (DVU0_1).<sup>*1</sup></p> <p>01: Uses the output from SRC0 (DVU0_0).<sup>*2</sup></p> <p>10: Uses the output from MIX.</p> <p>11: Setting prohibited</p> <p>Notes: 1. When the route to the FFU module is selected by the OPSLR_OPC0_1 register, this bit setting has no meaning.</p> <p>2. When the route to the FFU module is selected by the OPSLR_OPC0_0 register, this bit setting has no meaning.</p>
11, 10	—	All 0	R	<p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p>
9, 8	SOSEL2	00	R/W	<p>SSIF2 Output Select</p> <p>These bits select the output route to SSIF2.</p> <p>00: Uses the output from SRC3 (DVU0_3).<sup>*1</sup></p> <p>01: Uses the output from SRC0 (DVU0_0).<sup>*2, *4</sup></p> <p>10: Uses the output from SRC1 (DVU0_1).<sup>*3, *4</sup></p> <p>11: Uses the output from MIX.<sup>*4</sup></p> <p>Notes: 1. When the route to the FFU module is selected by the OPSLR_OPC0_3 register, this bit setting has no meaning.</p> <p>2. When the route to the FFU module is selected by the OPSLR_OPC0_0 register, this bit setting has no meaning.</p> <p>3. When the route to the FFU module is selected by the OPSLR_OPC0_1 register, this bit setting has no meaning.</p> <p>4. Valid only when SSIPMD_CIM.SSI012EN = 1.</p>
7, 6	—	All 0	R	<p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p>

Bit	Bit Name	Initial Value	R/W	Description
5, 4	SOSEL1	00	R/W	<p>SSIF1 Output Select</p> <p>These bits select the output route to SSIF1.</p> <p>00: Uses the output from SRC2 (DVU0_2).*1</p> <p>01: Uses the output from SRC0 (DVU0_0).*2, *4</p> <p>10: Uses the output from SRC1 (DVU0_1).*3, *4</p> <p>11: Uses the output from MIX.*4</p> <p>Notes: 1. When the route to the FFU module is selected by the OPSLR_OPC0_2 register, this bit setting has no meaning.</p> <p>2. When the route to the FFU module is selected by the OPSLR_OPC0_0 register, this bit setting has no meaning.</p> <p>3. When the route to the FFU module is selected by the OPSLR_OPC0_1 register, this bit setting has no meaning.</p> <p>4. Valid only when SSIPMD_CIM.SSI012EN = 1.</p>
3, 2	—	All 0	R	<p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p>
1, 0	SOSEL0	00	R/W	<p>SSIF0 Output Select</p> <p>These bits select the output route to SSIF0.</p> <p>00: Uses the output from SRC0 (DVU0_0).*1</p> <p>01: Uses the output from SRC1 (DVU0_1).*2</p> <p>10: Uses the output from MIX.</p> <p>11: Setting prohibited.</p> <p>Notes: 1. When the route to the FFU module is selected by the OPSLR_OPC0_0 register, this bit setting has no meaning.</p> <p>2. When the route to the FFU module is selected by the OPSLR_OPC0_1 register, this bit setting has no meaning.</p>

### 48.3.68 FFD0\_n Timing Select Register (FDTSELn\_CIM) (n = 0, 1, 2, 3)

FDTSELn\_CIM is a 32-bit readable/writable register that selects the input timing signal to be used in SRCn (n = 0, 1, 2, 3) when asynchronous mode is selected. This register setting is used only when the route with the FFD module is selected.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	SCKDIV										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	DIVEN	-	-	-	MTUSEL	SCKSEL			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26 to 16	SCKDIV	All 0	R/W	Division Ratio These bits specify the ratio for dividing the input clock specified in the SCKSEL bits to generate the timing signal. Note: Always write 0 to bit 16. (Writing 1 to bit 16 is ignored.) When all of these bits are set to 0, the division ratio is 1. Only the input clock selected by SCKSEL[3] = 0 will be divided. The WS signal selected by SCKSEL[3] = 1 will not be divided. The DIVEN bit must be set to 0 before changing these bits.
15 to 9	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
8	DIVEN	0	R/W	Division Enable Frequency division starts the moment 1 is written to this bit. 1: Starts frequency division. 0: Outputs 0.
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	MTUSEL	0	R/W	MTU Select This bit selects the clock signal specified by setting the SCKSEL bits to 0101 from the multi-function timer pulse unit 2. 0: Uses TIOC3A. 1: Uses TIOC4A
3 to 0	SCKSEL	All 0	R/W	Clock Select These bits select the clock that is used in generating the timing signal. 0000: Uses AUDIO_CLK. 0001: Uses the AUDIO_X1 input. 0010: Uses MLB_CLK. 0011: Uses USB_X1. 0100: Uses the peripheral clock (clkp1 $\phi$ ) divided into 2. 0101: Uses the signal specified by the MTUSEL bit. 0110 and 0111: Setting prohibited 1000: Uses the WS signal of SSIF0. 1001: Uses the WS signal of SSIF1. 1010: Uses the WS signal of SSIF2. 1011: Uses the WS signal of SSIF3. 1100: Uses the WS signal of SSIF4. 1101: Uses the WS signal of SSIF5. 1110 and 1111: Setting prohibited Note: The DIVEN bit must be set to 0 before changing these bits.

The input timing signal selector has the configuration shown in Figure 48.4.

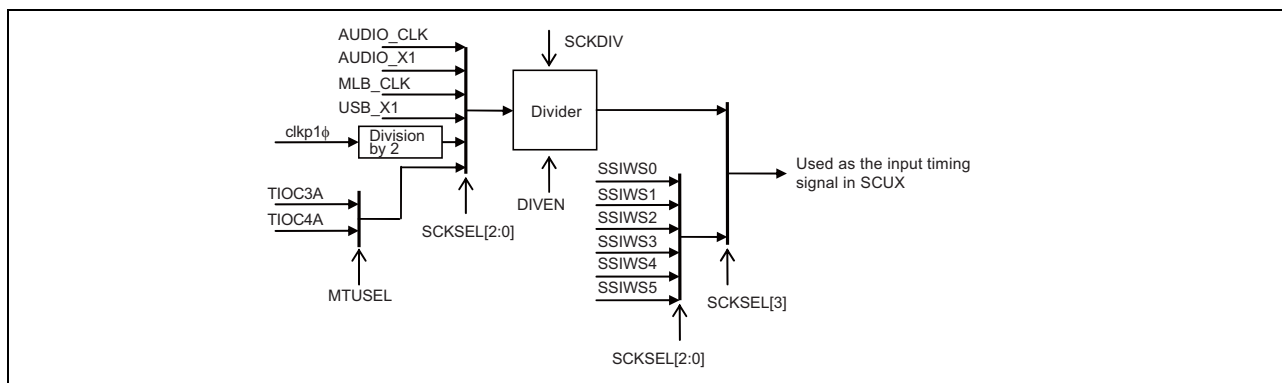


Figure 48.4 Configuration Diagram of Input Timing Signal Selector

### 48.3.69 FFU0\_n Timing Select Register (FDTSELn\_CIM) (n = 0, 1, 2, 3)

FDTSELn\_CIM is a 32-bit writable register that selects the output timing signal to be used in SRCn (n = 0, 1, 2, 3) when asynchronous mode is selected. This register setting is used when the route with the FFU module is selected or when output to the MIX is selected.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	SCKDIV										
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	-	-	-	-	-	W	W	W	W	W	W	W	W	W	W	W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	DIVEN	-	-	-	MTUSEL	SCKSEL			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	-	-	-	-	-	-	-	W	-	-	-	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	—	Reserved The write value should always be 0.
26 to 16	SCKDIV	All 0	W	Division Ratio These bits specify the ratio for dividing the input clock specified in the SCKSEL bits to generate the timing signal. Note: Always write 0 in bit 16. (Writing 1 to bit 16 is ignored.) When all of these bits are set to 0, the division ratio is 1. Only the input clock selected by SCKSEL[3] = 0 will be divided. The WS signal selected by SCKSEL[3] = 1 will not be divided. The DIVEN bit must be set to 0 before changing these bits.
15 to 9	—	All 0	—	Reserved The write value should always be 0.
8	DIVEN	0	W	Division Enable Frequency division starts the moment 1 is written to this bit. 1: Starts frequency division. 0: Outputs 0.
7 to 5	—	All 0	—	Reserved The write value should always be 0.
4	MTUSEL	0	W	MTU Select This bit selects the clock signal specified by setting the SCKSEL bits to 0101 from the multi-function timer pulse unit 2. 0: Uses TIOC3A. 1: Uses TIOC4A
3 to 0	SCKSEL	All 0	W	Clock Select These bits select the clock that is used in generating the timing signal. 0000: Uses AUDIO_CLK. 0001: Uses AUDIO_X1 input. 0010: Uses MLB_CLK. 0011: Uses USB_X1. 0100: Uses the peripheral clock (clkp1 $\phi$ ) divided into 2. 0101: Uses the signal specified by the MTUSEL bit. 0110 and 0111: Setting prohibited 1000: Uses the WS signal of SSIF0. 1001: Uses the WS signal of SSIF1. 1010: Uses the WS signal of SSIF2. 1011: Uses the WS signal of SSIF3. 1100: Uses the WS signal of SSIF4. 1101: Uses the WS signal of SSIF5. 1110 and 1111: Setting prohibited Note: The DIVEN bit must be set to 0 before changing these bits.

The output timing signal selector has the configuration shown in Figure 48.5.

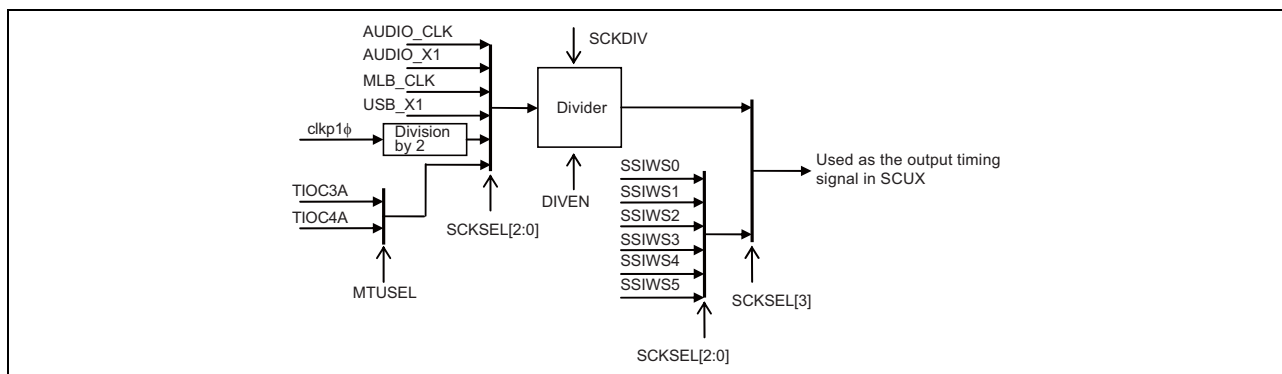


Figure 48.5 Configuration Diagram of Output Timing Signal Selector

### 48.3.70 SSI Pin Mode Register (SSIPMD\_CIM)

SSIPMD\_CIM is a 32-bit readable/writable register that sets the pin mode of the SSIF.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	-	-	-	-	-	-	-	-	-	SSI5 CKS	SSI4 CKS	SSI3 CKS	SSI2 CKS	SSI1 CKS	SSI0 CKS
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SSI3PMD	-	SSI 345EN	SSI4PMD	SSI5PMD	-	-	-	-	SSI 012EN	SSI2PMD	SSI1PMD				
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 22	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
21	SSI5CKS	0	R/W	SSI5 Clock Select This bit selects the clock to be connected to the AUDIO_CLK pin of the SSIF5 module. 0: Selects the AUDIO_CLK input. 1: Selects the MLB_CLK input. Note: When selecting AUDIO_CLK or MLB_CLK as the oversampling clock of the SSIF5 module, SSICR5.CKS = 1 must also be set in the SSIF5 module.
20	SSI4CKS	0	R/W	SSI4 Clock Select This bit selects the clock to be connected to the AUDIO_CLK pin of the SSIF4 module. 0: Selects the AUDIO_CLK input. 1: Selects the MLB_CLK input. Note: When selecting AUDIO_CLK or MLB_CLK as the oversampling clock of the SSIF4 module, SSICR4.CKS = 1 must also be set in the SSIF4 module.
19	SSI3CKS	0	R/W	SSI3 Clock Select This bit selects the clock to be connected to the AUDIO_CLK pin of the SSIF3 module. 0: Selects the AUDIO_CLK input. 1: Selects the MLB_CLK input. Note: When selecting AUDIO_CLK or MLB_CLK as the oversampling clock of the SSIF3 module, SSICR3.CKS = 1 must also be set in the SSIF3 module.
18	SSI2CKS	0	R/W	SSI2 Clock Select This bit selects the clock to be connected to the AUDIO_CLK pin of the SSIF2 module. 0: Selects the AUDIO_CLK input. 1: Selects the MLB_CLK input. Note: When selecting AUDIO_CLK or MLB_CLK as the oversampling clock of the SSIF2 module, SSICR2.CKS = 1 must also be set in the SSIF2 module.
17	SSI1CKS	0	R/W	SSI1 Clock Select This bit selects the clock to be connected to the AUDIO_CLK pin of the SSIF1 module. 0: Selects the AUDIO_CLK input. 1: Selects the MLB_CLK input. Note: When selecting AUDIO_CLK or MLB_CLK as the oversampling clock of the SSIF1 module, SSICR1.CKS = 1 must also be set in the SSIF1 module.

Bit	Bit Name	Initial Value	R/W	Description
16	SSI0CKS	0	R/W	<p>SSIF0 Clock Select</p> <p>This bit selects the clock to be connected to the AUDIO_CLK pin of the SSIF0 module.</p> <p>0: Selects the AUDIO_CLK input. 1: Selects the MLB_CLK input.</p> <p>Note: When selecting AUDIO_CLK or MLB_CLK as the oversampling clock of the SSIF0 module, SSICR0.CKS = 1 must also be set in the SSIF0 module.</p>
15, 14	SSI3PMD	00	R/W	<p>SSIF3 Pin Mode</p> <p>These bits select connection of the SSISCK3 and SSIWS3 pins.</p> <p>00: The pins are used independently. 01: The SSIF0 pins are used in common. Both SSIF0 and SSIF3 are slaves. 10: The SSIF0 pins are used in common. SSIF0 is the master and SSIF3 is the slave. 11: Setting prohibited</p>
13	—	0	R	<p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p>
12	SSI345EN	0	R/W	<p>SSIF345 Enable</p> <p>This bit selects whether the three modules SSIF3, SSIF4, and SSIF5 are used to provide six channels at transmission.</p> <p>0: Not handled as six channels. 1: Handled as six channels.</p> <p>Note: This bit can be set to 1 only when the SSI4PMD and SSI5PMD bits of this register are both set to 01 or 10. Usage of six channels at reception can be specified by SSIRSEL_CIM.SISEL1.</p>
11, 10	SSI4PMD	00	R/W	<p>SSIF4 Pin Mode</p> <p>These bits select connection of the SSISCK4 and SSIWS4 pins.</p> <p>00: The pins are used independently. 01: The SSIF3 pins are used in common. Both SSIF3 and SSIF4 are slaves. 10: The SSIF3 pins are used in common. SSIF3 is the master and SSIF4 is the slave. 11: Setting prohibited</p>
9, 8	SSI5PMD	00	R/W	<p>SSIF5 Pin Mode</p> <p>These bits select connection of the SSISCK5 and SSIWS5 pins.</p> <p>00: The pins are used independently. 01: The SSIF3 pins are used in common. Both SSIF3 and SSIF5 are slaves. 10: The SSIF3 pins are used in common. SSIF3 is the master and SSIF5 is the slave. 11: Setting prohibited</p>
7 to 5	—	All 0	R	<p>Reserved</p> <p>These bits are always read as 0. The write value should always be 0.</p>
4	SSI012EN	0	R/W	<p>SSIF012 Enable</p> <p>This bit selects whether the three modules SSIF0, SSIF1, and SSIF2 are used to provide six channels at transmission.</p> <p>0: Not handled as six channels. 1: Handled as six channels.</p> <p>Note: This bit can be set to 1 only when the SSI1PMD and SSI2PMD bits of this register are both set to 01 or 10. Usage of six channels at reception can be specified by SSIRSEL_CIM.SISEL0.</p>
3, 2	SSI2PMD	00	R/W	<p>SSIF2 Pin Mode</p> <p>These bits select connection of the SSISCK2 and SSIWS2 pins.</p> <p>00: The pins are used independently. 01: The SSIF0 pins are used in common. Both SSIF0 and SSIF2 are slaves. 10: The SSIF0 pins are used in common. SSIF0 is the master and SSIF2 is the slave. 11: Setting prohibited</p>



---

Bit	Bit Name	Initial Value	R/W	Description
1, 0	SSI1PMD	00	R/W	SSIF1 Pin Mode These bits select connection of the SSISCK1 and SSIWS1 pins. 00: The pins are used independently. 01: The SSIF0 pins are used in common. Both SSIF0 and SSIF1 are slaves. 10: The SSIF0 pins are used in common. SSIF0 is the master and SSIF1 is the slave. 11: Setting prohibited

---

### 48.3.71 SSI Control Register (SSICTRL\_CIM)

SSIPCTRL\_CIM is a 32-bit readable/writable register that controls connection with the SSIF module and starting and stopping of the SSIF module.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	SSI3TX	SSI4TX	SSI5TX	-	SSI3RX	SSI4RX	SSI5RX	-	-	-	-	-	-	SSI345TEN	SSI345REN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	-	W	W	W	-	W	W	W	-	-	-	-	-	-	W	W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	SSI0TX	SSI1TX	SSI2TX	-	SSI0RX	SSI1RX	SSI2RX	-	-	-	-	-	-	SSI012TEN	SSI012REN
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	-	W	W	W	-	W	W	W	-	-	-	-	-	-	W	W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	—	Reserved The write value should always be 0.
30	SSI3TX	0	W	SSIF3 Direct Transmission 0: Disables direct transmission to SSIF3. 1: Enables direct transmission to SSIF3.
29	SSI4TX	0	W	SSIF4 Direct Transmission 0: Disables direct transmission to SSIF4. 1: Enables direct transmission to SSIF4.
28	SSI5TX	0	W	SSIF5 Direct Transmission 0: Disables direct transmission to SSIF5. 1: Enables direct transmission to SSIF5.
27	—	0	—	Reserved The write value should always be 0.
26	SSI3RX	0	W	SSIF3 Direct Reception 0: Disables direct reception from SSIF3. 1: Enables direct reception from SSIF3.
25	SSI4RX	0	W	SSIF4 Direct Reception 0: Disables direct reception from SSIF4. 1: Enables direct reception from SSIF4.
24	SSI5RX	0	W	SSIF5 Direct Reception 0: Disables direct reception from SSIF5. 1: Enables direct reception from SSIF5.
23 to 18	—	All 0	—	Reserved The write value should always be 0.
17	SSI345TEN	0	W	SSIF345 Transmission Enable This bit specifies transmission in three modules SSIF3, SSIF4, and SSIF5 to be started or stopped simultaneously. 0: Stops transmission in SSIF3, SSIF4, and SSIF5. 1: Starts transmission in SSIF3, SSIF4, and SSIF5. Note: This function can be used only when SSIPMD_CIM.SSI345EN = 1. When SSIPMD_CIM.SSI345EN = 0, SSIF modules should be controlled using SSICRn.TEN. This bit must be set to 0 when performing direct transmission to SSIF3, SSIF4, or SSIF5.
16	SSI345REN	0	W	SSIF345 Reception Enable This bit specifies reception in three modules SSIF3, SSIF4, and SSIF5 to be started or stopped simultaneously. 0: Stops reception in SSIF3, SSIF4, and SSIF5. 1: Starts reception in SSIF3, SSIF4, and SSIF5. Note: This function can be used only when SSIRSEL_CIM.SISEL1 = 01. When SSIRSEL_CIM.SISEL1 = 00, SSIF modules should be controlled using SSICRn.REN. This bit must be set to 0 when performing direct reception from SSIF3, SSIF4, or SSIF5.

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	—	Reserved The write value should always be 0.
14	SSI0TX	0	W	SSIF0 Direct Transmission 0: Disables direct transmission to SSIF0. 1: Enables direct transmission to SSIF0.
13	SSI1TX	0	W	SSIF1 Direct Transmission 0: Disables direct transmission to SSIF1. 1: Enables direct transmission to SSIF1.
12	SSI2TX	0	W	SSIF2 Direct Transmission 0: Disables direct transmission to SSIF2. 1: Enables direct transmission to SSIF2.
11	—	0	—	Reserved The write value should always be 0.
10	SSI0RX	0	W	SSIF0 Direct Reception 0: Disables direct reception from SSIF0. 1: Enables direct reception from SSIF0.
9	SSI1RX	0	W	SSIF1 Direct Reception 0: Disables direct reception from SSIF1. 1: Enables direct reception from SSIF1.
8	SSI2RX	0	W	SSIF2 Direct Reception 0: Disables direct reception from SSIF2. 1: Enables direct reception from SSIF2.
7 to 2	—	All 0	—	Reserved The write value should always be 0.
1	SSI012TEN	0	W	SSIF012 Transmission Enable This bit specifies transmission in three modules SSIF0, SSIF1, and SSIF2 to be started or stopped simultaneously. 0: Stops transmission in SSIF0, SSIF1, and SSIF2. 1: Starts transmission in SSIF0, SSIF1, and SSIF2. Note: This function can be used only when SSIPMD_CIM.SSI012EN = 1. When SSIPMD_CIM.SSI012EN = 0, SSIF modules should be controlled using SSICRn.TEN. This bit must be set to 0 when performing direct transmission to SSIF0, SSIF1, or SSIF2.
0	SSI012REN	0	W	SSIF012 Reception Enable This bit specifies reception in three modules SSIF0, SSIF1, and SSIF2 to be started or stopped simultaneously. 0: Stops reception in SSIF0, SSIF1, and SSIF2. 1: Starts reception in SSIF0, SSIF1, and SSIF2. Note: This function can be used only when SSIRSEL_CIM.SISEL0 = 01. When SSIRSEL_CIM.SISEL0 = 00, SSIF modules should be controlled using SSICRn.REN. This bit must be set to 0 when performing direct reception from SSIF0, SSIF1, or SSIF2.

### 48.3.72 SRCn Route Select Register (SRCRSELn\_CIM) (n = 0, 1, 2, 3)

SRCRSEL\_CIM is a 32-bit readable/writable register that sets the route for the data that is input to the SRC.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16				
	-	PLACE7				-	PLACE6				-	PLACE5				-	PLACE4			
Initial value:	0	1	1	1	0	1	1	0	0	1	0	1	0	1	0	0				
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W				
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0				
	-	PLACE3				-	PLACE2				-	PLACE1				-	PLACE0			
Initial value:	0	0	1	1	0	0	1	0	0	0	0	1	0	0	0	0				
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W				

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30 to 28	PLACE7	111	R/W	Place 7 Stream data is rearranged before it is input to the SRC. These bits select data on the input side to be output to place 7 on the output side. 000: Data at place 0 on the input side is output to place 7 on the output side. 001: Data at place 1 on the input side is output to place 7 on the output side. 010: Data at place 2 on the input side is output to place 7 on the output side. 011: Data at place 3 on the input side is output to place 7 on the output side. 100: Data at place 4 on the input side is output to place 7 on the output side. 101: Data at place 5 on the input side is output to place 7 on the output side. 110: Data at place 6 on the input side is output to place 7 on the output side. 111: Data at place 7 on the input side is output to place 7 on the output side. Note: This bit setting is used when eight channels are set. When six or less channels are set, set these bits to the same value as the initial value.
27	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
26 to 24	PLACE6	110	R/W	Place 6 Stream data is rearranged before it is input to the SRC. These bits select data on the input side to be output to place 6 on the output side. 000: Data at place 0 on the input side is output to place 6 on the output side. 001: Data at place 1 on the input side is output to place 6 on the output side. 010: Data at place 2 on the input side is output to place 6 on the output side. 011: Data at place 3 on the input side is output to place 6 on the output side. 100: Data at place 4 on the input side is output to place 6 on the output side. 101: Data at place 5 on the input side is output to place 6 on the output side. 110: Data at place 6 on the input side is output to place 6 on the output side. 111: Data at place 7 on the input side is output to place 6 on the output side. Note: This bit setting is used when eight channels are set. When six or less channels are set, set these bits to the same value as the initial value.

Bit	Bit Name	Initial Value	R/W	Description
23	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
22 to 20	PLACE5	101	R/W	Place 5 Stream data is rearranged before it is input to the SRC. These bits select data on the input side to be output to place 5 on the output side. 000: Data at place 0 on the input side is output to place 5 on the output side. 001: Data at place 1 on the input side is output to place 5 on the output side. 010: Data at place 2 on the input side is output to place 5 on the output side. 011: Data at place 3 on the input side is output to place 5 on the output side. 100: Data at place 4 on the input side is output to place 5 on the output side. 101: Data at place 5 on the input side is output to place 5 on the output side. 110: Data at place 6 on the input side is output to place 5 on the output side. 111: Data at place 7 on the input side is output to place 5 on the output side. Note: This bit setting is used when six or more channels are set. When four or less channels are set, set these bits to the same value as the initial value.
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
18 to 16	PLACE4	100	R/W	Place 4 Stream data is rearranged before it is input to the SRC. These bits select data on the input side to be output to place 4 on the output side. 000: Data at place 0 on the input side is output to place 4 on the output side. 001: Data at place 1 on the input side is output to place 4 on the output side. 010: Data at place 2 on the input side is output to place 4 on the output side. 011: Data at place 3 on the input side is output to place 4 on the output side. 100: Data at place 4 on the input side is output to place 4 on the output side. 101: Data at place 5 on the input side is output to place 4 on the output side. 110: Data at place 6 on the input side is output to place 4 on the output side. 111: Data at place 7 on the input side is output to place 4 on the output side. Note: This bit setting is used when six or more channels are set. When four or less channels are set, set these bits to the same value as the initial value.
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
14 to 12	PLACE3	011	R/W	<p>Place 3</p> <p>Stream data is rearranged before it is input to the SRC.</p> <p>These bits select data on the input side to be output to place 3 on the output side.</p> <p>000: Data at place 0 on the input side is output to place 3 on the output side.</p> <p>001: Data at place 1 on the input side is output to place 3 on the output side.</p> <p>010: Data at place 2 on the input side is output to place 3 on the output side.</p> <p>011: Data at place 3 on the input side is output to place 3 on the output side.</p> <p>100: Data at place 4 on the input side is output to place 3 on the output side.</p> <p>101: Data at place 5 on the input side is output to place 3 on the output side.</p> <p>110: Data at place 6 on the input side is output to place 3 on the output side.</p> <p>111: Data at place 7 on the input side is output to place 3 on the output side.</p> <p>Note: This bit setting is used when four or more channels are set. When two or less channels are set, set these bits to the same value as the initial value.</p>
11	—	0	R	<p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p>
10 to 8	PLACE2	010	R/W	<p>Place 2</p> <p>Stream data is rearranged before it is input to the SRC.</p> <p>These bits select data on the input side to be output to place 2 on the output side.</p> <p>000: Data at place 0 on the input side is output to place 2 on the output side.</p> <p>001: Data at place 1 on the input side is output to place 2 on the output side.</p> <p>010: Data at place 2 on the input side is output to place 2 on the output side.</p> <p>011: Data at place 3 on the input side is output to place 2 on the output side.</p> <p>100: Data at place 4 on the input side is output to place 2 on the output side.</p> <p>101: Data at place 5 on the input side is output to place 2 on the output side.</p> <p>110: Data at place 6 on the input side is output to place 2 on the output side.</p> <p>111: Data at place 7 on the input side is output to place 2 on the output side.</p> <p>Note: This bit setting is used when four or more channels are set. When two or less channels are set, set these bits to the same value as the initial value.</p>
7	—	0	R	<p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p>

Bit	Bit Name	Initial Value	R/W	Description
6 to 4	PLACE1	001	R/W	<p>Place 1</p> <p>Stream data is rearranged before it is input to the SRC. These bits select data on the input side to be output to place 1 on the output side.</p> <p>000: Data at place 0 on the input side is output to place 1 on the output side.</p> <p>001: Data at place 1 on the input side is output to place 1 on the output side.</p> <p>010: Data at place 2 on the input side is output to place 1 on the output side.</p> <p>011: Data at place 3 on the input side is output to place 1 on the output side.</p> <p>100: Data at place 4 on the input side is output to place 1 on the output side.</p> <p>101: Data at place 5 on the input side is output to place 1 on the output side.</p> <p>110: Data at place 6 on the input side is output to place 1 on the output side.</p> <p>111: Data at place 7 on the input side is output to place 1 on the output side.</p>
3	—	0	R	<p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p>
2 to 0	PLACE0	000	R/W	<p>Place 0</p> <p>Stream data is rearranged before it is input to the SRC. These bits select data on the input side to be output to place 0 on the output side.</p> <p>000: Data at place 0 on the input side is output to place 0 on the output side.</p> <p>001: Data at place 1 on the input side is output to place 0 on the output side.</p> <p>010: Data at place 2 on the input side is output to place 0 on the output side.</p> <p>011: Data at place 3 on the input side is output to place 0 on the output side.</p> <p>100: Data at place 4 on the input side is output to place 0 on the output side.</p> <p>101: Data at place 5 on the input side is output to place 0 on the output side.</p> <p>110: Data at place 6 on the input side is output to place 0 on the output side.</p> <p>111: Data at place 7 on the input side is output to place 0 on the output side.</p>

### 48.3.73 MIX Route Select Register (MIXRSEL\_CIM)

MIXRSEL\_CIM is a 32-bit readable/writable register that sets the route for the data that is output from the MIX.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	-	PLACE7			-	PLACE6			-	PLACE5			-	PLACE4		
Initial value:	0	1	1	1	0	1	1	0	0	1	0	1	0	1	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	PLACE3			-	PLACE2			-	PLACE1			-	PLACE0		
Initial value:	0	0	1	1	0	0	1	0	0	0	0	1	0	0	0	0
R/W:	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
30 to 28	PLACE7	111	R/W	Place 7 Stream data is rearranged after it is output from the MIX. These bits select data on the input side to be output to place 7 on the output side. 000: Data at place 0 on the input side is output to place 7 on the output side. 001: Data at place 1 on the input side is output to place 7 on the output side. 010: Data at place 2 on the input side is output to place 7 on the output side. 011: Data at place 3 on the input side is output to place 7 on the output side. 100: Data at place 4 on the input side is output to place 7 on the output side. 101: Data at place 5 on the input side is output to place 7 on the output side. 110: Data at place 6 on the input side is output to place 7 on the output side. 111: Data at place 7 on the input side is output to place 7 on the output side. Note: This bit setting is used when eight channels are set. When six or less channels are set, set these bits to the same value as the initial value.
27	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
26 to 24	PLACE6	110	R/W	Place 6 Stream data is rearranged after it is output from the MIX. These bits select data on the input side to be output to place 6 on the output side. 000: Data at place 0 on the input side is output to place 6 on the output side. 001: Data at place 1 on the input side is output to place 6 on the output side. 010: Data at place 2 on the input side is output to place 6 on the output side. 011: Data at place 3 on the input side is output to place 6 on the output side. 100: Data at place 4 on the input side is output to place 6 on the output side. 101: Data at place 5 on the input side is output to place 6 on the output side. 110: Data at place 6 on the input side is output to place 6 on the output side. 111: Data at place 7 on the input side is output to place 6 on the output side. Note: This bit setting is used when eight channels are set. When six or less channels are set, set these bits to the same value as the initial value.



Bit	Bit Name	Initial Value	R/W	Description
23	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
22 to 20	PLACE5	101	R/W	Place 5 Stream data is rearranged after it is output from the MIX. These bits select data on the input side to be output to place 5 on the output side. 000: Data at place 0 on the input side is output to place 5 on the output side. 001: Data at place 1 on the input side is output to place 5 on the output side. 010: Data at place 2 on the input side is output to place 5 on the output side. 011: Data at place 3 on the input side is output to place 5 on the output side. 100: Data at place 4 on the input side is output to place 5 on the output side. 101: Data at place 5 on the input side is output to place 5 on the output side. 110: Data at place 6 on the input side is output to place 5 on the output side. 111: Data at place 7 on the input side is output to place 5 on the output side. Note: This bit setting is used when six or more channels are set. When four or less channels are set, set these bits to the same value as the initial value.
19	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
18 to 16	PLACE4	100	R/W	Place 4 Stream data is rearranged after it is output from the MIX. These bits select data on the input side to be output to place 4 on the output side. 000: Data at place 0 on the input side is output to place 4 on the output side. 001: Data at place 1 on the input side is output to place 4 on the output side. 010: Data at place 2 on the input side is output to place 4 on the output side. 011: Data at place 3 on the input side is output to place 4 on the output side. 100: Data at place 4 on the input side is output to place 4 on the output side. 101: Data at place 5 on the input side is output to place 4 on the output side. 110: Data at place 6 on the input side is output to place 4 on the output side. 111: Data at place 7 on the input side is output to place 4 on the output side. Note: This bit setting is used when six or more channels are set. When four or less channels are set, set these bits to the same value as the initial value.
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

Bit	Bit Name	Initial Value	R/W	Description
14 to 12	PLACE3	011	R/W	<p>Place 3</p> <p>Stream data is rearranged after it is output from the MIX. These bits select data on the input side to be output to place 3 on the output side.</p> <p>000: Data at place 0 on the input side is output to place 3 on the output side.</p> <p>001: Data at place 1 on the input side is output to place 3 on the output side.</p> <p>010: Data at place 2 on the input side is output to place 3 on the output side.</p> <p>011: Data at place 3 on the input side is output to place 3 on the output side.</p> <p>100: Data at place 4 on the input side is output to place 3 on the output side.</p> <p>101: Data at place 5 on the input side is output to place 3 on the output side.</p> <p>110: Data at place 6 on the input side is output to place 3 on the output side.</p> <p>111: Data at place 7 on the input side is output to place 3 on the output side.</p> <p>Note: This bit setting is used when four or more channels are set. When two or less channels are set, set these bits to the same value as the initial value.</p>
11	—	0	R	<p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p>
10 to 8	PLACE2	010	R/W	<p>Place 2</p> <p>Stream data is rearranged after it is output from the MIX. These bits select data on the input side to be output to place 2 on the output side.</p> <p>000: Data at place 0 on the input side is output to place 2 on the output side.</p> <p>001: Data at place 1 on the input side is output to place 2 on the output side.</p> <p>010: Data at place 2 on the input side is output to place 2 on the output side.</p> <p>011: Data at place 3 on the input side is output to place 2 on the output side.</p> <p>100: Data at place 4 on the input side is output to place 2 on the output side.</p> <p>101: Data at place 5 on the input side is output to place 2 on the output side.</p> <p>110: Data at place 6 on the input side is output to place 2 on the output side.</p> <p>111: Data at place 7 on the input side is output to place 2 on the output side.</p> <p>Note: This bit setting is used when four or more channels are set. When two or less channels are set, set these bits to the same value as the initial value.</p>
7	—	0	R	<p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p>

Bit	Bit Name	Initial Value	R/W	Description
6 to 4	PLACE1	001	R/W	<p>Place 1</p> <p>Stream data is rearranged after it is output from the MIX. These bits select data on the input side to be output to place 1 on the output side.</p> <p>000: Data at place 0 on the input side is output to place 1 on the output side.</p> <p>001: Data at place 1 on the input side is output to place 1 on the output side.</p> <p>010: Data at place 2 on the input side is output to place 1 on the output side.</p> <p>011: Data at place 3 on the input side is output to place 1 on the output side.</p> <p>100: Data at place 4 on the input side is output to place 1 on the output side.</p> <p>101: Data at place 5 on the input side is output to place 1 on the output side.</p> <p>110: Data at place 6 on the input side is output to place 1 on the output side.</p> <p>111: Data at place 7 on the input side is output to place 1 on the output side.</p>
3	—	0	R	<p>Reserved</p> <p>This bit is always read as 0. The write value should always be 0.</p>
2 to 0	PLACE0	000	R/W	<p>Place 0</p> <p>Stream data is rearranged after it is output from the MIX. These bits select data on the input side to be output to place 0 on the output side.</p> <p>000: Data at place 0 on the input side is output to place 0 on the output side.</p> <p>001: Data at place 1 on the input side is output to place 0 on the output side.</p> <p>010: Data at place 2 on the input side is output to place 0 on the output side.</p> <p>011: Data at place 3 on the input side is output to place 0 on the output side.</p> <p>100: Data at place 4 on the input side is output to place 0 on the output side.</p> <p>101: Data at place 5 on the input side is output to place 0 on the output side.</p> <p>110: Data at place 6 on the input side is output to place 0 on the output side.</p> <p>111: Data at place 7 on the input side is output to place 0 on the output side.</p>

## 48.4 Operation

### 48.4.1 Initial Setting Procedure

Figure 48.6 to Figure 48.8 show the initial setting procedure of the SCUX. For details on register setting, refer to section 48.3, Register Descriptions.

### 48.4.2 Transfer Start Procedure and Stop Procedure

Figure 48.9 to Figure 48.11 show the transfer start procedure and stop procedure of the SCUX. For details on register setting, refer to section 48.3, Register Descriptions.

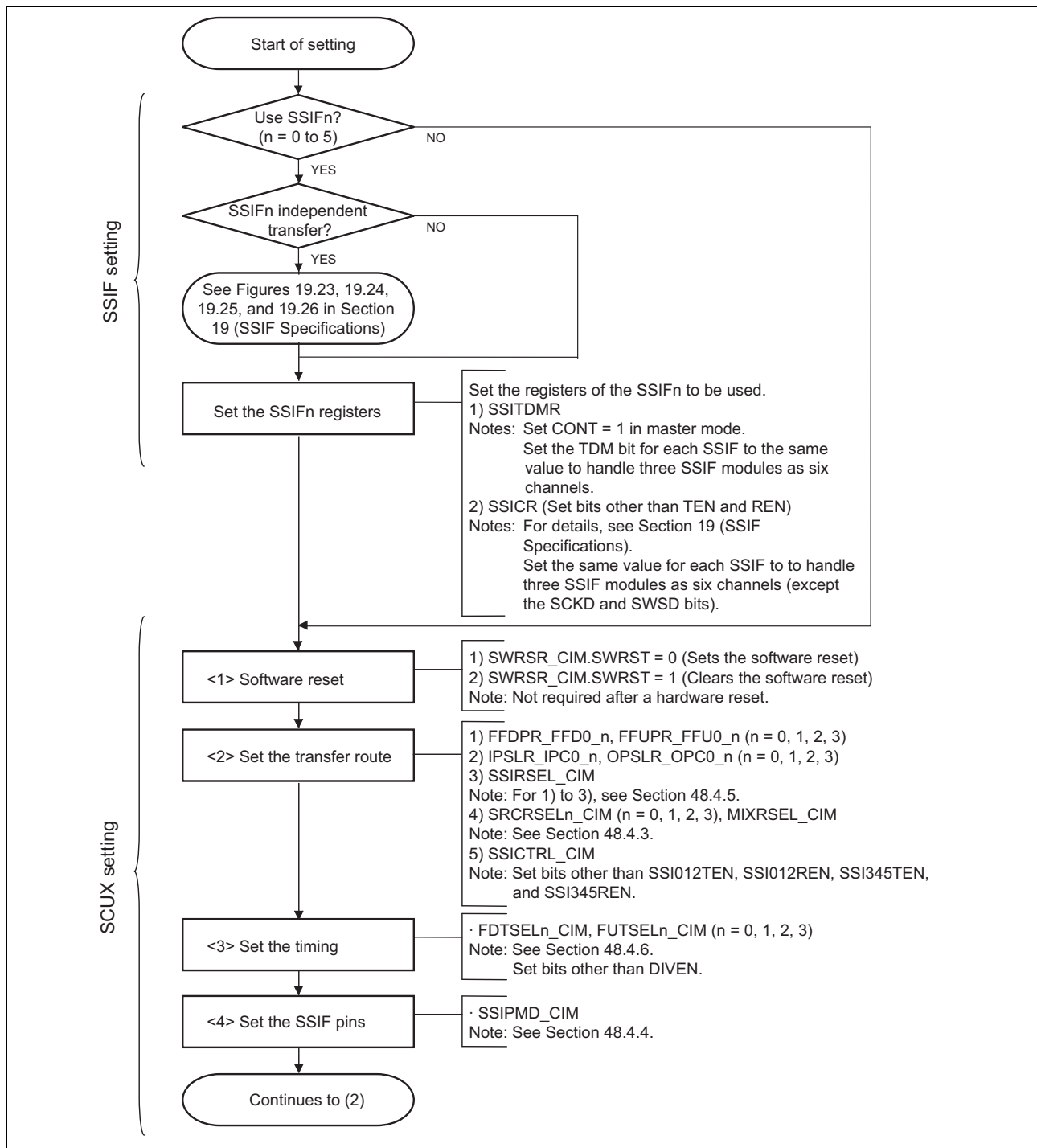


Figure 48.6 Initial Setting Procedure of SCUX (1)

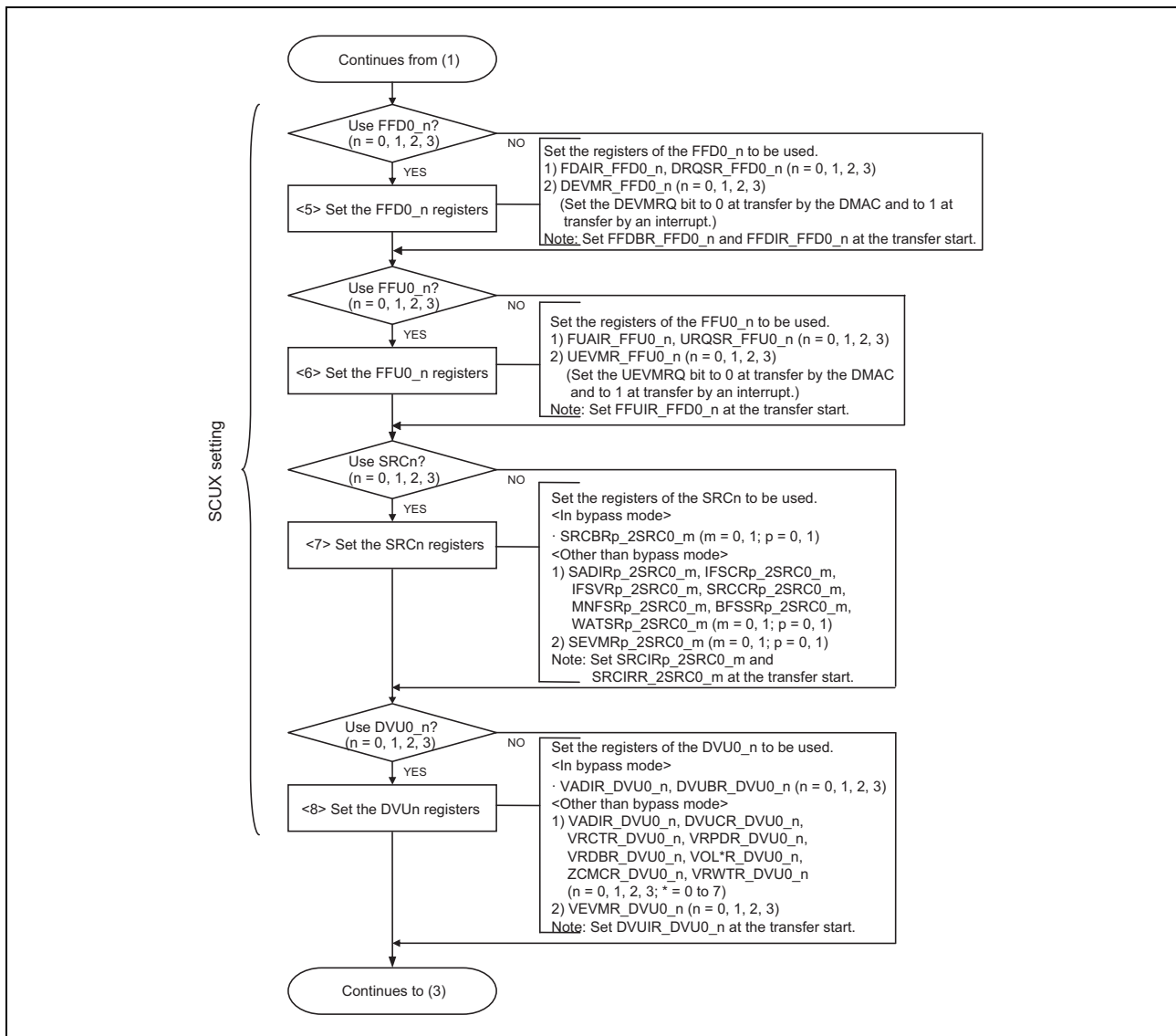


Figure 48.7 Initial Setting Procedure of SCUX (2)

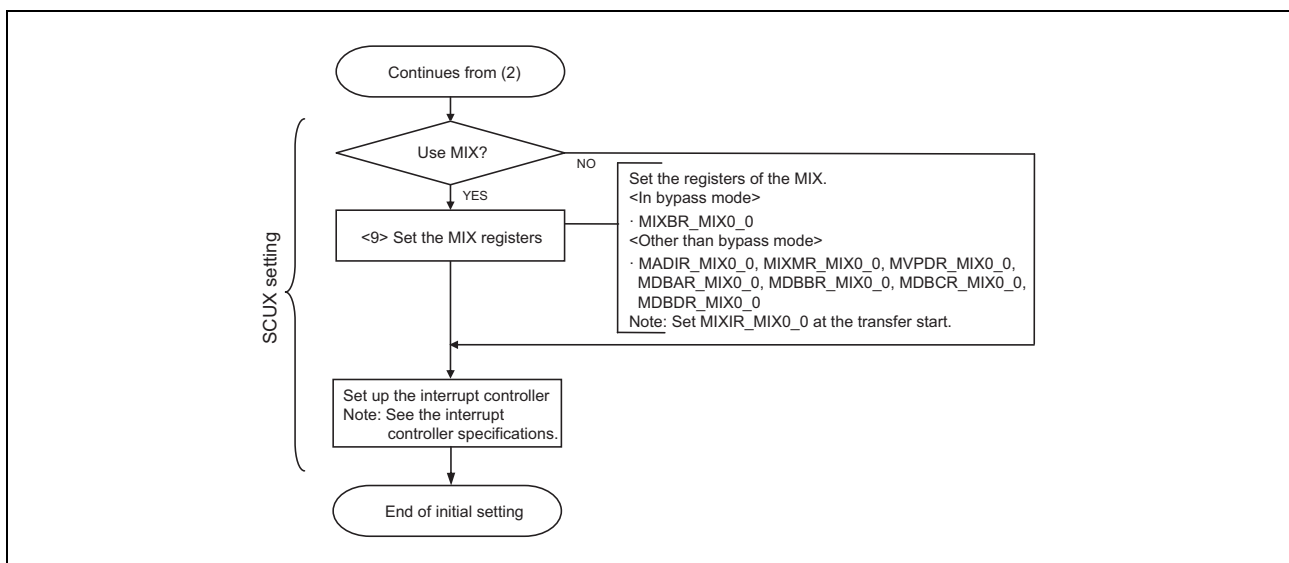


Figure 48.8 Initial Setting Procedure of SCUX (3)

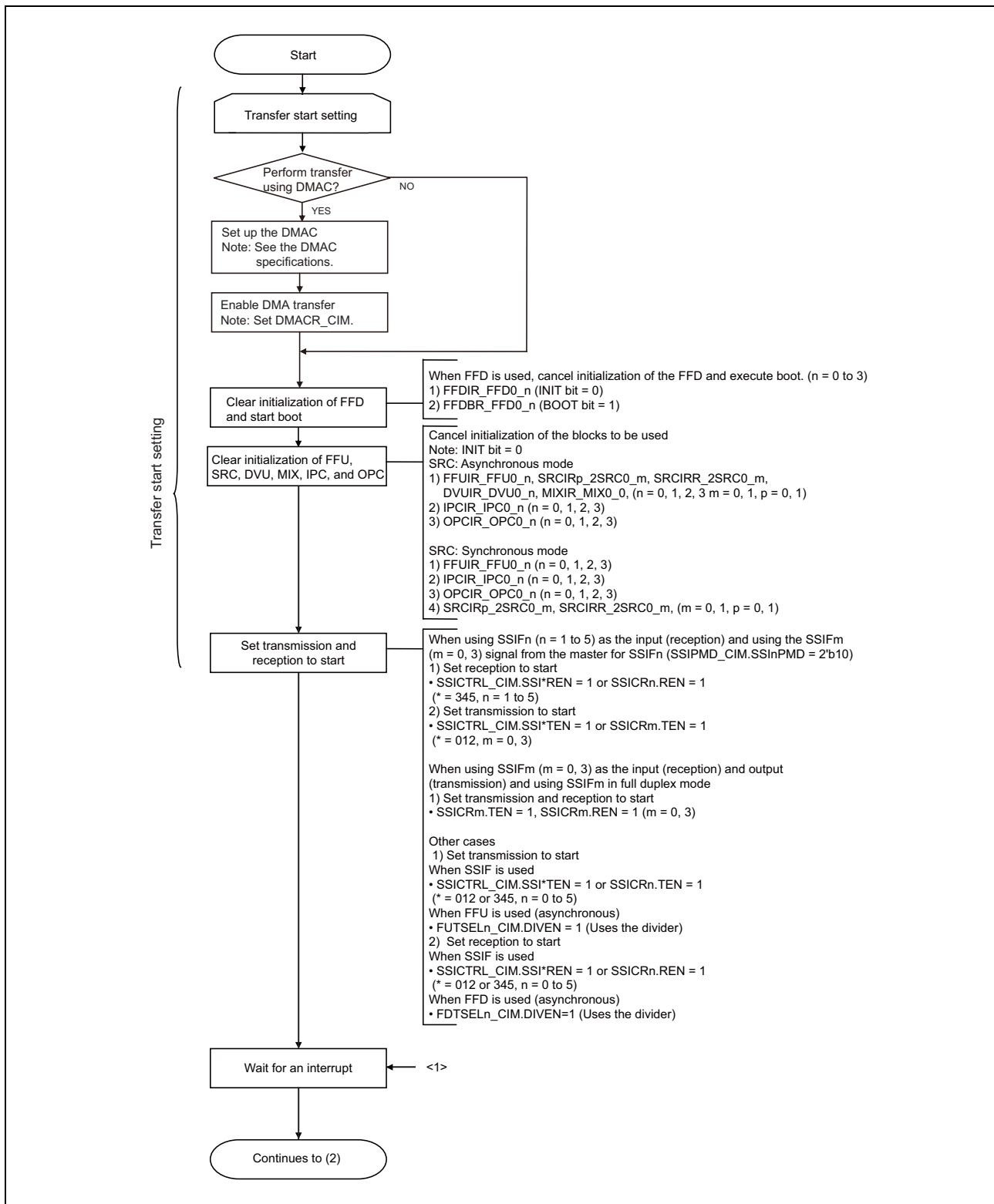


Figure 48.9 Transfer Start Procedure and Stop Procedure of SCUX (1)

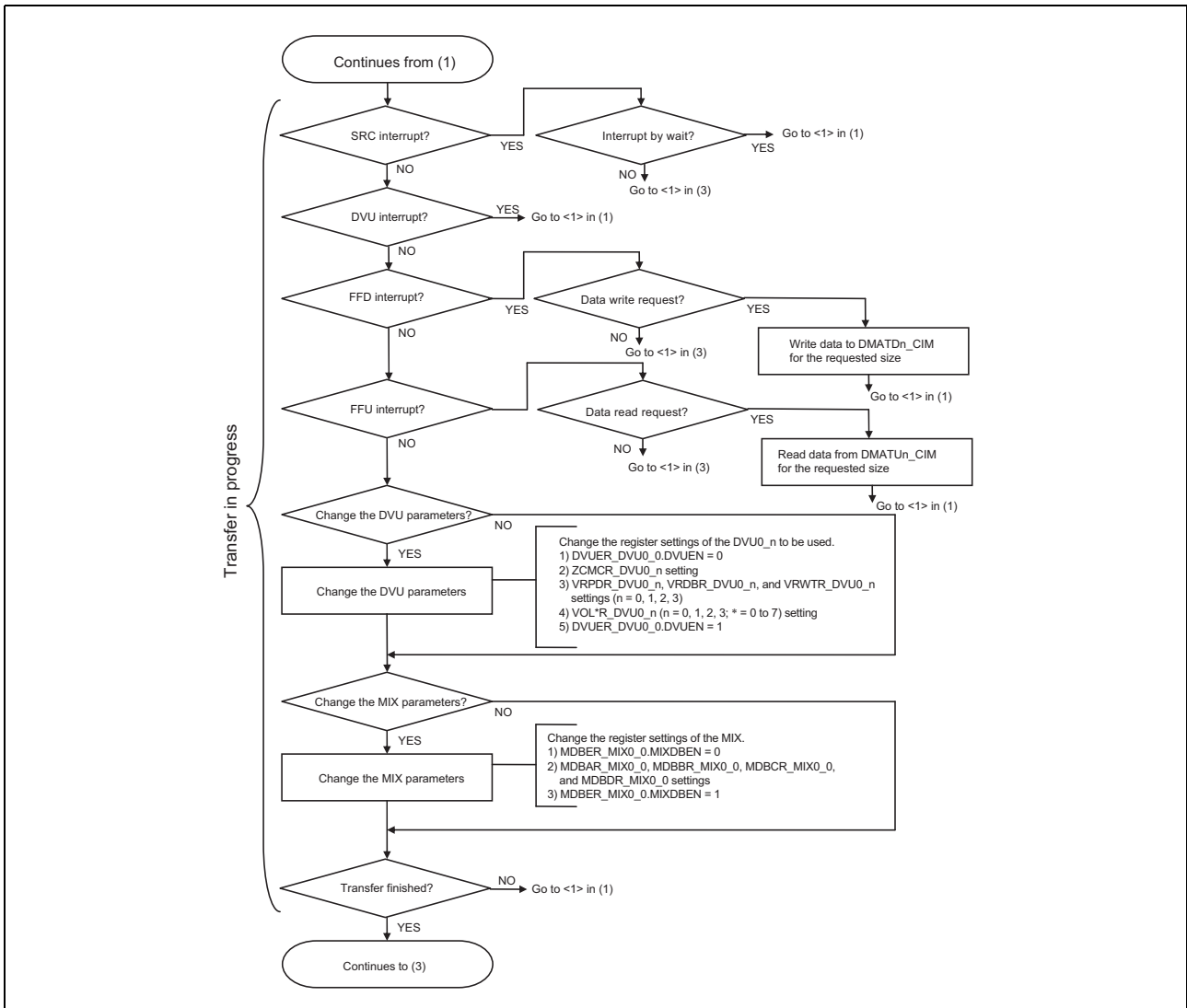


Figure 48.10 Transfer Start Procedure and Stop Procedure of SCUX (2)

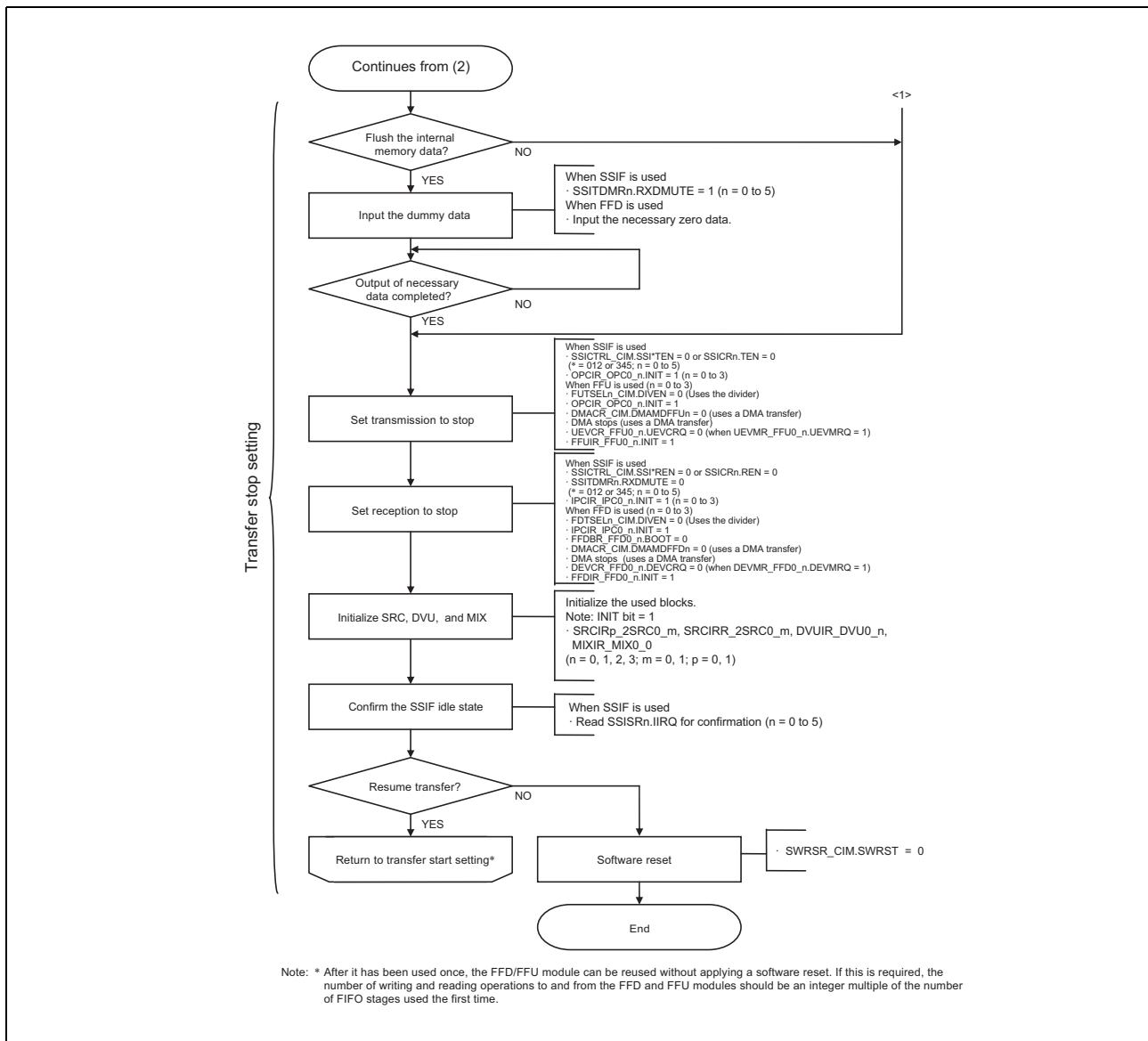



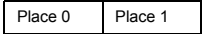

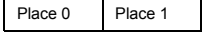
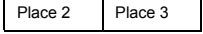
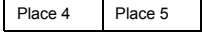



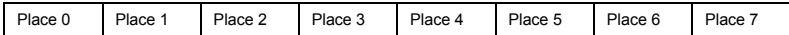
Figure 48.11 Transfer Start Procedure and Stop Procedure of SCUX (3)



### 48.4.3 Data Rearrangement for Each Channel

Data can be rearranged independently for each channel in the SCUX. Data rearrangement can be performed immediately before input to the SRC and immediately after output from the MIX. Table 48.9 to Table 48.11 show definition of data places. Data rearrangement immediately before the SRC is performed as described in section 48.3.72, SRCn Route Select Register (SRCRSELn\_CIM) (n = 0, 1, 2, 3). Data rearrangement immediately after the MIX is performed as described in section 48.3.73, MIX Route Select Register (MIXRSEL\_CIM).

**Table 48.9 Definition of Data Places (1)**

SSIF • Stereo (2 ch)	SSIWS0  SSIDATA0 
SSIF • Stereo × 3 (6 ch)	SSIWS0  SSIDATA0  SSIDATA1  SSIDATA2 
SSIF • TDM (6 ch)	SSIWS0  SSIDATA0 
SSIF • Multichannel (8 ch)	SSIWS0  SSIDATA0 

**Table 48.10 Definition of Data Places (2)**

<p>FFD0_n • DMATDn_CIM 32-bit access (2 ch)</p>	<table style="margin-left: auto; margin-right: auto;"> <tr> <td></td> <td style="text-align: center;">31</td> <td></td> <td style="text-align: center;">8</td> <td style="text-align: center;">7</td> <td style="text-align: center;">0</td> </tr> <tr> <td>1st write data</td> <td style="border: 1px solid black; text-align: center;">Place 0</td> <td style="border: 1px solid black; text-align: center;">0</td> <td></td> <td></td> <td></td> </tr> <tr> <td>2nd write data</td> <td style="border: 1px solid black; text-align: center;">Place 1</td> <td style="border: 1px solid black; text-align: center;">0</td> <td></td> <td></td> <td></td> </tr> <tr> <td>3rd write data</td> <td style="border: 1px solid black; text-align: center;">Place 0</td> <td style="border: 1px solid black; text-align: center;">0</td> <td></td> <td></td> <td></td> </tr> <tr> <td>4th write data</td> <td style="border: 1px solid black; text-align: center;">Place 1</td> <td style="border: 1px solid black; text-align: center;">0</td> <td></td> <td></td> <td></td> </tr> </table>		31		8	7	0	1st write data	Place 0	0				2nd write data	Place 1	0				3rd write data	Place 0	0				4th write data	Place 1	0																											
	31		8	7	0																																																		
1st write data	Place 0	0																																																					
2nd write data	Place 1	0																																																					
3rd write data	Place 0	0																																																					
4th write data	Place 1	0																																																					
<p>FFD0_n • DMATDn_CIM 16-bit access (2 ch)</p>	<table style="margin-left: auto; margin-right: auto;"> <tr> <td></td> <td style="text-align: center;">31</td> <td></td> <td style="text-align: center;">16</td> <td style="text-align: center;">15</td> <td style="text-align: center;">0</td> </tr> <tr> <td>1st write data</td> <td style="border: 1px solid black; text-align: center;">Place 0</td> <td style="border: 1px solid black; text-align: center;">0</td> <td></td> <td></td> <td></td> </tr> <tr> <td>2nd write data</td> <td style="border: 1px solid black; text-align: center;">Place 1</td> <td style="border: 1px solid black; text-align: center;">0</td> <td></td> <td></td> <td></td> </tr> <tr> <td>3rd write data</td> <td style="border: 1px solid black; text-align: center;">Place 0</td> <td style="border: 1px solid black; text-align: center;">0</td> <td></td> <td></td> <td></td> </tr> <tr> <td>4th write data</td> <td style="border: 1px solid black; text-align: center;">Place 1</td> <td style="border: 1px solid black; text-align: center;">0</td> <td></td> <td></td> <td></td> </tr> </table>		31		16	15	0	1st write data	Place 0	0				2nd write data	Place 1	0				3rd write data	Place 0	0				4th write data	Place 1	0																											
	31		16	15	0																																																		
1st write data	Place 0	0																																																					
2nd write data	Place 1	0																																																					
3rd write data	Place 0	0																																																					
4th write data	Place 1	0																																																					
<p>FFD0_n • DMATDn_CIM 32-bit access (8 ch)</p>	<table style="margin-left: auto; margin-right: auto;"> <tr> <td></td> <td style="text-align: center;">31</td> <td></td> <td style="text-align: center;">8</td> <td style="text-align: center;">7</td> <td style="text-align: center;">0</td> </tr> <tr> <td>1st write data</td> <td style="border: 1px solid black; text-align: center;">Place 0</td> <td style="border: 1px solid black; text-align: center;">0</td> <td></td> <td></td> <td></td> </tr> <tr> <td>2nd write data</td> <td style="border: 1px solid black; text-align: center;">Place 1</td> <td style="border: 1px solid black; text-align: center;">0</td> <td></td> <td></td> <td></td> </tr> <tr> <td>3rd write data</td> <td style="border: 1px solid black; text-align: center;">Place 2</td> <td style="border: 1px solid black; text-align: center;">0</td> <td></td> <td></td> <td></td> </tr> <tr> <td>4th write data</td> <td style="border: 1px solid black; text-align: center;">Place 3</td> <td style="border: 1px solid black; text-align: center;">0</td> <td></td> <td></td> <td></td> </tr> <tr> <td>5th write data</td> <td style="border: 1px solid black; text-align: center;">Place 4</td> <td style="border: 1px solid black; text-align: center;">0</td> <td></td> <td></td> <td></td> </tr> <tr> <td>6th write data</td> <td style="border: 1px solid black; text-align: center;">Place 5</td> <td style="border: 1px solid black; text-align: center;">0</td> <td></td> <td></td> <td></td> </tr> <tr> <td>7th write data</td> <td style="border: 1px solid black; text-align: center;">Place 6</td> <td style="border: 1px solid black; text-align: center;">0</td> <td></td> <td></td> <td></td> </tr> <tr> <td>8th write data</td> <td style="border: 1px solid black; text-align: center;">Place 7</td> <td style="border: 1px solid black; text-align: center;">0</td> <td></td> <td></td> <td></td> </tr> </table>		31		8	7	0	1st write data	Place 0	0				2nd write data	Place 1	0				3rd write data	Place 2	0				4th write data	Place 3	0				5th write data	Place 4	0				6th write data	Place 5	0				7th write data	Place 6	0				8th write data	Place 7	0			
	31		8	7	0																																																		
1st write data	Place 0	0																																																					
2nd write data	Place 1	0																																																					
3rd write data	Place 2	0																																																					
4th write data	Place 3	0																																																					
5th write data	Place 4	0																																																					
6th write data	Place 5	0																																																					
7th write data	Place 6	0																																																					
8th write data	Place 7	0																																																					
<p>FFU0_n • DMATUn_CIM 32-bit access (2 ch)</p>	<table style="margin-left: auto; margin-right: auto;"> <tr> <td></td> <td style="text-align: center;">31</td> <td></td> <td style="text-align: center;">8</td> <td style="text-align: center;">7</td> <td style="text-align: center;">0</td> </tr> <tr> <td>1st write data</td> <td style="border: 1px solid black; text-align: center;">Place 0</td> <td style="border: 1px solid black; text-align: center;">0</td> <td></td> <td></td> <td></td> </tr> <tr> <td>2nd write data</td> <td style="border: 1px solid black; text-align: center;">Place 1</td> <td style="border: 1px solid black; text-align: center;">0</td> <td></td> <td></td> <td></td> </tr> <tr> <td>3rd write data</td> <td style="border: 1px solid black; text-align: center;">Place 0</td> <td style="border: 1px solid black; text-align: center;">0</td> <td></td> <td></td> <td></td> </tr> <tr> <td>4th write data</td> <td style="border: 1px solid black; text-align: center;">Place 1</td> <td style="border: 1px solid black; text-align: center;">0</td> <td></td> <td></td> <td></td> </tr> </table>		31		8	7	0	1st write data	Place 0	0				2nd write data	Place 1	0				3rd write data	Place 0	0				4th write data	Place 1	0																											
	31		8	7	0																																																		
1st write data	Place 0	0																																																					
2nd write data	Place 1	0																																																					
3rd write data	Place 0	0																																																					
4th write data	Place 1	0																																																					

**Table 48.11 Definition of Data Places (3)**

<p>FFU0_n • DMATUn_CIM 16-bit access (2 ch)</p>	<table style="margin-left: auto; margin-right: auto;"> <tr> <td></td> <td style="text-align: center;">31</td> <td></td> <td style="text-align: center;">16</td> <td style="text-align: center;">15</td> <td style="text-align: center;">0</td> </tr> <tr> <td>1st write data</td> <td colspan="2" style="border: 1px solid black; text-align: center;">Place 0</td> <td colspan="3" style="border: 1px solid black; text-align: center;">-</td> </tr> <tr> <td>2nd write data</td> <td colspan="2" style="border: 1px solid black; text-align: center;">Place 1</td> <td colspan="3" style="border: 1px solid black; text-align: center;">-</td> </tr> <tr> <td>3rd write data</td> <td colspan="2" style="border: 1px solid black; text-align: center;">Place 0</td> <td colspan="3" style="border: 1px solid black; text-align: center;">-</td> </tr> <tr> <td>4th write data</td> <td colspan="2" style="border: 1px solid black; text-align: center;">Place 1</td> <td colspan="3" style="border: 1px solid black; text-align: center;">-</td> </tr> </table>		31		16	15	0	1st write data	Place 0		-			2nd write data	Place 1		-			3rd write data	Place 0		-			4th write data	Place 1		-																										
	31		16	15	0																																																		
1st write data	Place 0		-																																																				
2nd write data	Place 1		-																																																				
3rd write data	Place 0		-																																																				
4th write data	Place 1		-																																																				
<p>FFU0_n • DMATUn_CIM 32-bit access (8 ch)</p>	<table style="margin-left: auto; margin-right: auto;"> <tr> <td></td> <td style="text-align: center;">31</td> <td></td> <td style="text-align: center;">8</td> <td style="text-align: center;">7</td> <td style="text-align: center;">0</td> </tr> <tr> <td>1st write data</td> <td colspan="2" style="border: 1px solid black; text-align: center;">Place 0</td> <td colspan="3" style="border: 1px solid black; text-align: center;">0</td> </tr> <tr> <td>2nd write data</td> <td colspan="2" style="border: 1px solid black; text-align: center;">Place 1</td> <td colspan="3" style="border: 1px solid black; text-align: center;">0</td> </tr> <tr> <td>3rd write data</td> <td colspan="2" style="border: 1px solid black; text-align: center;">Place 2</td> <td colspan="3" style="border: 1px solid black; text-align: center;">0</td> </tr> <tr> <td>4th write data</td> <td colspan="2" style="border: 1px solid black; text-align: center;">Place 3</td> <td colspan="3" style="border: 1px solid black; text-align: center;">0</td> </tr> <tr> <td>5th write data</td> <td colspan="2" style="border: 1px solid black; text-align: center;">Place 4</td> <td colspan="3" style="border: 1px solid black; text-align: center;">0</td> </tr> <tr> <td>6th write data</td> <td colspan="2" style="border: 1px solid black; text-align: center;">Place 5</td> <td colspan="3" style="border: 1px solid black; text-align: center;">0</td> </tr> <tr> <td>7th write data</td> <td colspan="2" style="border: 1px solid black; text-align: center;">Place 6</td> <td colspan="3" style="border: 1px solid black; text-align: center;">0</td> </tr> <tr> <td>8th write data</td> <td colspan="2" style="border: 1px solid black; text-align: center;">Place 7</td> <td colspan="3" style="border: 1px solid black; text-align: center;">0</td> </tr> </table>		31		8	7	0	1st write data	Place 0		0			2nd write data	Place 1		0			3rd write data	Place 2		0			4th write data	Place 3		0			5th write data	Place 4		0			6th write data	Place 5		0			7th write data	Place 6		0			8th write data	Place 7		0		
	31		8	7	0																																																		
1st write data	Place 0		0																																																				
2nd write data	Place 1		0																																																				
3rd write data	Place 2		0																																																				
4th write data	Place 3		0																																																				
5th write data	Place 4		0																																																				
6th write data	Place 5		0																																																				
7th write data	Place 6		0																																																				
8th write data	Place 7		0																																																				

### 48.4.4 Pin Connection Specifications of SSIF

SSIF0 to SSIF3 and SSIF3 to SSIF5 can be operated in synchronization with the same SSISCK and SSIWS signals. The AUDIO\_CLK input to the SSIF can be selected between external pins AUDIO\_CLK and MLB\_CLK. These settings are made as described in section 48.3.70, SSI Pin Mode Register (SSIPMD\_CIM). Figure 48.12 shows the pin connection specifications of each SSIF module.

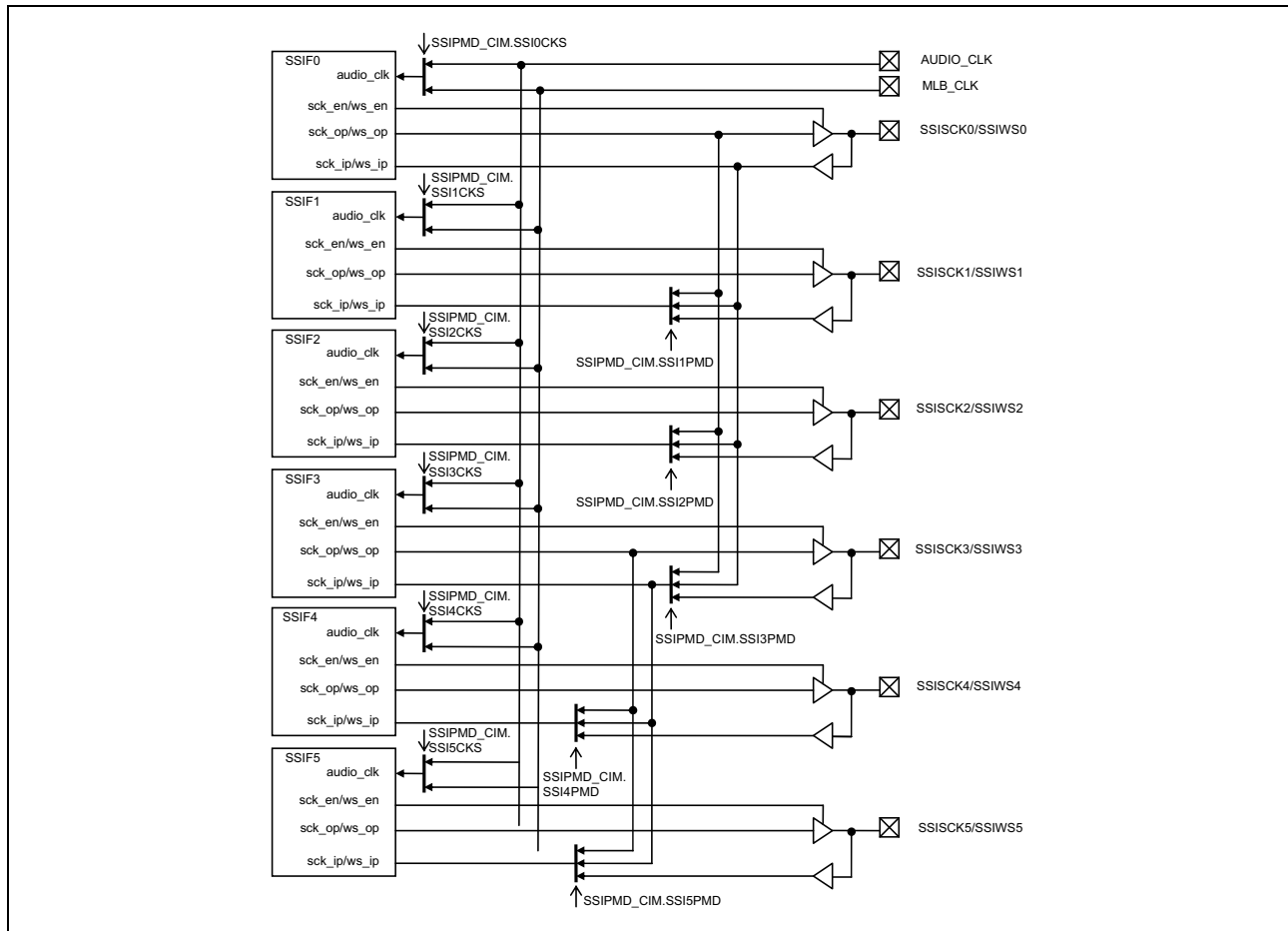


Figure 48.12 Pin Connection Specifications of SSIF

### 48.4.5 Data Transfer Route

Table 48.12 to Table 48.14 list the data transfer routes that can be selected by the SCUX.

Table 48.15 to Table 48.17 show the necessary register settings for selecting each transfer route.

The 2SRC0\_m bypass register p and DVU0\_n bypass register should be set appropriately to bypass the SRC and DVU, respectively.

**Table 48.12 Data Transfer Routes (1)**

Route	No.	SSIF (SRC)	FFD	SRC + DVU	MIX	SSIF (DST)	FFU
SSIF → SRC → SSIF	1	SSIF0	—	SRC0*1 + DVU0_0	—	SSIF0	—
	2	SSIF0	—	SRC0*1 + DVU0_0	—	SSIF3	—
	3	SSIF012 (6 ch)	—	SRC0*1 + DVU0_0	—	SSIF345 (6 ch)	—
	4	SSIF3	—	SRC1*1 + DVU0_1	—	SSIF3	—
	5	SSIF3	—	SRC1*1 + DVU0_1	—	SSIF0	—
	6	SSIF345 (6 ch)	—	SRC1*1 + DVU0_1	—	SSIF012 (6 ch)	—
	7	SSIF1	—	SRC2*1 + DVU0_2	—	SSIF4	—
	8	SSIF4	—	SRC2*1 + DVU0_2	—	SSIF1	—
	9	SSIF2	—	SRC3*1 + DVU0_3	—	SSIF5	—
	10	SSIF5	—	SRC3*1 + DVU0_3	—	SSIF2	—
SSIF → SRC → FFU	1	SSIF0	—	SRC0*1	—	—	FFU0_0
	2	SSIF012 (6 ch)	—	SRC0*1	—	—	FFU0_0
	3	SSIF3	—	SRC1*1	—	—	FFU0_1
	4	SSIF345 (6 ch)	—	SRC1*1	—	—	FFU0_1
	5	SSI1	—	SRC2*1	—	—	FFU0_2
	6	SSI4	—	SRC2*1	—	—	FFU0_2
	7	SSI2	—	SRC3*1	—	—	FFU0_3
	8	SSI5	—	SRC3*1	—	—	FFU0_3

Note 1. Selectable only in asynchronous mode. Not selectable in synchronous mode.

Table 48.13 Data Transfer Routes (2)

Route	No.	SSIF (SRC)	FFD	SRC + DVU	MIX	SSIF (DST)	FFU
SSIF → SRC → MIX → SSIF	1	SSIF0	—	SRC0*1 + DVU0_0	MIX	SSIF0	—
	2	SSIF0	—	SRC0*1 + DVU0_0	MIX	SSIF3	—
	3	SSIF0	—	SRC0*1 + DVU0_0	MIX	SSIF345 (6 ch)	—
	4	SSIF3	—	SRC1*1 + DVU0_1	MIX	SSIF3	—
	5	SSIF3	—	SRC1*1 + DVU0_1	MIX	SSIF0	—
	6	SSIF3	—	SRC1*1 + DVU0_1	MIX	SSIF012 (6 ch)	—
	7	SSIF1	—	SRC2*1 + DVU0_2	MIX	SSIF0	—
	8	SSIF1	—	SRC2*1 + DVU0_2	MIX	SSIF3	—
	9	SSIF1	—	SRC2*1 + DVU0_2	MIX	SSIF345 (6 ch)	—
	10	SSIF4	—	SRC2*1 + DVU0_2	MIX	SSIF0	—
	11	SSIF4	—	SRC2*1 + DVU0_2	MIX	SSIF012 (6 ch)	—
	12	SSIF4	—	SRC2*1 + DVU0_2	MIX	SSIF3	—
	13	SSIF2	—	SRC3*1 + DVU0_3	MIX	SSIF0	—
	14	SSIF2	—	SRC3*1 + DVU0_3	MIX	SSIF3	—
	15	SSIF2	—	SRC3*1 + DVU0_3	MIX	SSIF345 (6 ch)	—
	16	SSIF5	—	SRC3*1 + DVU0_3	MIX	SSIF0	—
	17	SSIF5	—	SRC3*1 + DVU0_3	MIX	SSIF012 (6 ch)	—
	18	SSIF5	—	SRC3*1 + DVU0_3	MIX	SSIF3	—
FFD → SRC → SSIF	1	—	FFD0_0	SRC0*1 + DVU0_0	—	SSIF0	—
	2	—	FFD0_0	SRC0*1 + DVU0_0	—	SSIF012 (6 ch)	—
	3	—	FFD0_0	SRC0*1 + DVU0_0	—	SSIF3	—
	4	—	FFD0_0	SRC0*1 + DVU0_0	—	SSIF345 (6 ch)	—
	5	—	FFD0_1	SRC1*1 + DVU0_1	—	SSIF3	—
	6	—	FFD0_1	SRC1*1 + DVU0_1	—	SSIF345 (6 ch)	—
	7	—	FFD0_1	SRC1*1 + DVU0_1	—	SSIF0	—
	8	—	FFD0_1	SRC1*1 + DVU0_1	—	SSIF012 (6 ch)	—
	9	—	FFD0_2	SRC2*1 + DVU0_2	—	SSIF1	—
	10	—	FFD0_2	SRC2*1 + DVU0_2	—	SSIF4	—
	11	—	FFD0_3	SRC3*1 + DVU0_3	—	SSIF2	—
	12	—	FFD0_3	SRC3*1 + DVU0_3	—	SSIF5	—

Note 1. Selectable only in asynchronous mode. Not selectable in synchronous mode.

Table 48.14 Data Transfer Routes (3)

Route	No.	SSIF (SRC)	FFD	SRC+DVU	MIX	SSIF (DST)	FFU
FFD → SRC → FFU	1	—	FFD0_0	SRC0 (Sync)			FFU0_0
	2	—	FFD0_0	SRC0 (Async)			FFU0_0
	3	—	FFD0_1	SRC1 (Sync)			FFU0_1
	4	—	FFD0_1	SRC1 (Async)			FFU0_1
	5	—	FFD0_2	SRC2 (Sync)			FFU0_2
	6	—	FFD0_2	SRC2 (Async)			FFU0_2
	7	—	FFD0_3	SRC3 (Sync)			FFU0_3
	8	—	FFD0_3	SRC3 (Async)			FFU0_3
FFD → SRC → MIX → SSIF	1	—	FFD0_0	SRC0*1 + DVU0_0	MIX	SSIF0	—
	2	—	FFD0_0	SRC0*1 + DVU0_0	MIX	SSIF012 (6 ch)	—
	3	—	FFD0_0	SRC0*1 + DVU0_0	MIX	SSIF3	—
	4	—	FFD0_0	SRC0*1 + DVU0_0	MIX	SSIF345 (6 ch)	—
	5	—	FFD0_1	SRC1*1 + DVU0_1	MIX	SSIF0	—
	6	—	FFD0_1	SRC1*1 + DVU0_1	MIX	SSIF012 (6 ch)	—
	7	—	FFD0_1	SRC1*1 + DVU0_1	MIX	SSIF3	—
	8	—	FFD0_1	SRC1*1 + DVU0_1	MIX	SSIF345 (6 ch)	—
	9	—	FFD0_2	SRC2*1 + DVU0_2	MIX	SSIF0	—
	10	—	FFD0_2	SRC2*1 + DVU0_2	MIX	SSIF012 (6 ch)	—
	11	—	FFD0_2	SRC2*1 + DVU0_2	MIX	SSIF3	—
	12	—	FFD0_2	SRC2*1 + DVU0_2	MIX	SSIF345 (6 ch)	—
	13	—	FFD0_3	SRC3*1 + DVU0_3	MIX	SSIF0	—
	14	—	FFD0_3	SRC3*1 + DVU0_3	MIX	SSIF012 (6 ch)	—
	15	—	FFD0_3	SRC3*1 + DVU0_3	MIX	SSIF3	—
	16	—	FFD0_3	SRC3*1 + DVU0_3	MIX	SSIF345 (6 ch)	—

Note 1. Selectable only in asynchronous mode. Not selectable in synchronous mode.

**Table 48.15 Route Select Register Settings (1)**

Route	No.	SSIRSEL_CIM										IPSLR_IPC0 <sub>n</sub> SEL	OPSLR_OPC0 <sub>n</sub> SEL	FFDPR_FF00 <sub>n</sub> PASS	FFUPR_FF00 <sub>n</sub> PASS	SSI345EN	SSIPMD_CIM SSI012EN
		SSISEL3	SSISEL2	SSISEL1	SSISEL0	SSOSEL5	SSOSEL4	SSOSEL3	SSOSEL2	SSOSEL1	SSOSEL0						
SSIF → SRC → SSIF	1	-	-	-	00	-	-	-	-	-	00	001	001	00	00	0	0
	2	-	-	-	00	-	-	01	-	-	-	001	001	00	00	0	0
	3	-	-	-	01	10	10	01	-	-	-	001	001	00	00	1	0
	4	-	-	00	-	-	-	00	-	-	-	001	001	00	00	0	0
	5	-	-	00	-	-	-	-	-	-	01	001	001	00	00	0	0
	6	-	-	01	-	-	-	-	10	10	01	001	001	00	00	0	1
	7	-	00	-	-	-	00	-	-	-	-	001	001	00	00	0	0
	8	-	01	-	-	-	-	-	-	-	00	001	001	00	00	0	0
	9	00	-	-	-	00	-	-	-	-	-	001	001	00	00	0	0
	10	01	-	-	-	-	-	-	00	-	-	001	001	00	00	0	0
SSIF → SRC → FFU	1	-	-	-	00	-	-	-	-	-	-	001	011	00	01	0	0
	2	-	-	-	01	-	-	-	-	-	-	001	011	00	01	0	0
	3	-	-	00	-	-	-	-	-	-	-	001	011	00	01	0	0
	4	-	-	01	-	-	-	-	-	-	-	001	011	00	01	0	0
	5	-	00	-	-	-	-	-	-	-	-	001	011	00	01	0	0
	6	-	01	-	-	-	-	-	-	-	-	001	011	00	01	0	0
	7	00	-	-	-	-	-	-	-	-	-	001	011	00	01	0	0
	8	01	-	-	-	-	-	-	-	-	-	001	011	00	01	0	0



**Table 48.16 Route Select Register Settings (2)**

Route	No.	SSIRSEL_CIM										IPSLR_IPC0_n SEL	OPSLR_OPC0 SEL	FFDPR_FF0_n PASS	FFUPR_FF0_n PASS	SS1345EN	SSIPMD_CIM SSI012EN
		SSISEL3	SSISEL2	SSISEL1	SSISEL0	SSOSEL5	SSOSEL4	SSOSEL3	SSOSEL2	SSOSEL1	SSOSEL0						
SSIF → SRC → MIX → SSI	1	-	-	-	00	-	-	-	-	-	10	001	001	00	00	0	0
	2	-	-	-	00	-	-	10	-	-	-	001	001	00	00	0	0
	3	-	-	-	00	11	11	10	-	-	-	001	001	00	00	1	0
	4	-	-	00	-	-	-	10	-	-	-	001	001	00	00	0	0
	5	-	-	00	-	-	-	-	-	-	10	001	001	00	00	0	0
	6	-	-	00	-	-	-	-	11	11	10	001	001	00	00	0	1
	7	-	00	-	-	-	-	-	-	-	10	001	001	00	00	0	0
	8	-	00	-	-	-	-	10	-	-	-	001	001	00	00	0	0
	9	-	00	-	-	11	11	10	-	-	-	001	001	00	00	1	0
	10	-	01	-	-	-	-	-	-	-	10	001	001	00	00	0	0
	11	-	01	-	-	-	-	-	11	11	10	001	001	00	00	0	1
	12	-	01	-	-	-	-	10	-	-	-	001	001	00	00	0	0
	13	00	-	-	-	-	-	-	-	-	10	001	001	00	00	0	0
	14	00	-	-	-	-	-	10	-	-	-	001	001	00	00	0	0
	15	00	-	-	-	11	11	10	-	-	-	001	001	00	00	1	0
	16	01	-	-	-	-	-	-	-	-	10	001	001	00	00	0	0
	17	01	-	-	-	-	-	-	11	11	10	001	001	00	00	0	1
	18	01	-	-	-	-	-	10	-	-	-	001	001	00	00	0	0
FFD → SRC → SSIF	1	-	-	-	-	-	-	-	-	-	00	011	001	01	00	0	0
	2	-	-	-	-	-	-	01	01	00	011	001	01	00	0	1	
	3	-	-	-	-	-	-	01	-	-	-	011	001	01	00	0	0
	4	-	-	-	-	10	10	01	-	-	-	011	001	01	00	1	0
	5	-	-	-	-	-	-	00	-	-	-	011	001	01	00	0	0
	6	-	-	-	-	01	01	00	-	-	-	011	001	01	00	1	0
	7	-	-	-	-	-	-	-	-	-	01	011	001	01	00	0	0
	8	-	-	-	-	-	-	-	10	10	01	011	001	01	00	0	1
	9	-	-	-	-	-	-	-	-	00	-	011	001	01	00	0	0
	10	-	-	-	-	-	-	00	-	-	-	011	001	01	00	0	0
	11	-	-	-	-	-	-	-	00	-	-	011	001	01	00	0	0
	12	-	-	-	-	00	-	-	-	-	-	011	001	01	00	0	0

**Table 48.17 Route Select Register Settings (3)**

Route	No.	SSIRSEL_CIM										IPSLR_IPC0_n	OPSLR_OPC0_n	FFDPR_FF0_n	FFUPR_FF0_n	SSIPMD_CIM		
		SOSEL0	SOSEL1	SOSEL2	SOSEL3	SOSEL4	SOSEL5	SOSEL0	SOSEL1	SOSEL2	SOSEL3	SOSEL4	SOSEL5	IPC_PASS_SEL	OPC_PASS_SEL	PASS	PASS	SS345EN
FFD → SRC → FFU	1	-	-	-	-	-	-	-	-	-	-	-	100	100	10	10	0	0
	2	-	-	-	-	-	-	-	-	-	-	-	011	011	01	01	0	0
	3	-	-	-	-	-	-	-	-	-	-	-	100	100	10	10	0	0
	4	-	-	-	-	-	-	-	-	-	-	-	011	011	01	01	0	0
	5	-	-	-	-	-	-	-	-	-	-	-	100	100	10	10	0	0
	6	-	-	-	-	-	-	-	-	-	-	-	011	011	01	01	0	0
	7	-	-	-	-	-	-	-	-	-	-	-	100	100	10	10	0	0
	8	-	-	-	-	-	-	-	-	-	-	-	011	011	01	01	0	0
FFD → SRC → MIX → SSIF	1	-	-	-	-	-	-	-	-	-	10	011	001	01	00	0	0	
	2	-	-	-	-	-	-	-	11	11	10	011	001	01	00	0	1	
	3	-	-	-	-	-	-	10	-	-	-	011	001	01	00	0	0	
	4	-	-	-	-	11	11	10	-	-	-	011	001	01	00	1	0	
	5	-	-	-	-	-	-	-	-	-	10	011	001	01	00	0	0	
	6	-	-	-	-	-	-	-	11	11	10	011	001	01	00	0	1	
	7	-	-	-	-	-	-	10	-	-	-	011	001	01	00	0	0	
	8	-	-	-	-	11	11	10	-	-	-	011	001	01	00	1	0	
	9	-	-	-	-	-	-	-	-	-	10	011	001	01	00	0	0	
	10	-	-	-	-	-	-	-	11	11	10	011	001	01	00	0	1	
	11	-	-	-	-	-	-	10	-	-	-	011	001	01	00	0	0	
	12	-	-	-	-	11	11	10	-	-	-	011	001	01	00	1	0	
	13	-	-	-	-	-	-	-	-	-	10	011	001	01	00	0	0	
	14	-	-	-	-	-	-	-	11	11	10	011	001	01	00	0	1	
	15	-	-	-	-	-	-	10	-	-	-	011	001	01	00	0	0	
	16	-	-	-	-	11	11	10	-	-	-	011	001	01	00	1	0	

### 48.4.6 Input Timing Signal and Output Timing Signal

The input timing signal and output timing signal need to be set when using the SRC or MIX. Table 48.18 to Table 48.20 show the input/output timing setting necessary for each transfer route in Table 48.12 to Table 48.14. When direct connection between the SRC and SSIF is selected, the WS signal of the relevant SSIF is automatically selected regardless of the setting of the FFD0\_n/FFU0\_n timing select register. When direct connection between the MIX and SSIF is selected, set the prescribed value in the FFU0\_n timing select register. When connection between the FFD and SRC or connection between the SRC and FFU is selected, a desired value can be set in the FFD0\_n/FFU0\_n timing select register and timing signals corresponding to the setting are used.

**Table 48.18 Input/Output Timing Setting (1)**

Route	No.	Input Timing Setting							FDTSELn_CIM	Input Timing Signal	Output Timing Setting						FUTSELn_CIM	Output Timing Signal
		SSICTRL_CIM						SSICTRL_CIM										
		SS5RX	SS4RX	SS3RX	SS2RX	SS1RX	SS0RX	SS6TX			SS4TX	SS3TX	SS2TX	SS1TX	SS0TX			
SSIF → SRC → SSIF	1	—	—	—	—	—	1	Setting unnecessary	SSIWS0*1	—	—	0	—	—	1	Setting unnecessary	SSIWS0*1	
	2	—	—	—	—	—	1	Setting unnecessary	SSIWS0*1	—	—	1	—	—	0	Setting unnecessary	SSIWS3*1	
	3	—	—	—	—	—	1	Setting unnecessary	SSIWS0*1	—	—	1	—	—	0	Setting unnecessary	SSIWS3*1	
	4	—	—	1	—	—	—	Setting unnecessary	SSIWS3*1	—	—	1	—	—	0	Setting unnecessary	SSIWS3*1	
	5	—	—	1	—	—	—	Setting unnecessary	SSIWS3*1	—	—	0	—	—	1	Setting unnecessary	SSIWS0*1	
	6	—	—	1	—	—	—	Setting unnecessary	SSIWS3*1	—	—	0	—	—	1	Setting unnecessary	SSIWS0*1	
	7	—	0	—	—	1	—	Setting unnecessary	SSIWS1*1	—	1	—	—	0	—	Setting unnecessary	SSIWS4*1	
	8	—	1	—	—	0	—	Setting unnecessary	SSIWS4*1	—	0	—	—	1	—	Setting unnecessary	SSIWS1*1	
	9	0	—	—	1	—	—	Setting unnecessary	SSIWS2*1	1	—	—	0	—	—	Setting unnecessary	SSIWS5*1	
	10	1	—	—	0	—	—	Setting unnecessary	SSIWS5*1	0	—	—	1	—	—	Setting unnecessary	SSIWS2*1	
SSIF → SRC → FFU	1	—	—	—	—	—	1	Setting unnecessary	SSIWS0*1	—	—	0	—	—	0	Arbitrary	←	
	2	—	—	—	—	—	1	Setting unnecessary	SSIWS0*1	—	—	0	—	—	0	Arbitrary	←	
	3	—	—	1	—	—	—	Setting unnecessary	SSIWS3*1	—	—	0	—	—	0	Arbitrary	←	
	4	—	—	1	—	—	—	Setting unnecessary	SSIWS3*1	—	—	0	—	—	0	Arbitrary	←	
	5	—	0	—	—	1	—	Setting unnecessary	SSIWS1*1	—	0	—	—	0	—	Arbitrary	←	
	6	—	1	—	—	0	—	Setting unnecessary	SSIWS4*1	—	0	—	—	0	—	Arbitrary	←	
	7	0	—	—	1	—	—	Setting unnecessary	SSIWS2*1	0	—	—	0	—	—	Arbitrary	←	
	8	1	—	—	0	—	—	Setting unnecessary	SSIWS5*1	0	—	—	0	—	—	Arbitrary	←	

Note 1. Automatically selected regardless of the FDTSELn\_CIM and FUTSELn\_CIM settings.

**Table 48.19 Input/Output Timing Setting (2)**

Route	No.	Input Timing Setting							FDTSEL <sub>n</sub> _CIM	Input Timing Signal	Output Timing Setting						FUTSEL <sub>n</sub> _CIM	Output Timing Signal
		SSICTRL_CIM						SSICTRL_CIM										
		SSI5RX	SSI4RX	SSI3RX	SSI2RX	SSI1RX	SSI0RX	SSI5TX			SSI4TX	SSI3TX	SSI2TX	SSI1TX	SSI0TX			
SSIF → SRC → MIX → SSIF	1	—	—	—	—	—	1	Setting unnecessary	SSIWS0*1	—	—	0	—	—	1	SCKSEL: 1000	SSIWS0	
	2	—	—	—	—	—	1	Setting unnecessary	SSIWS0*1	—	—	1	—	—	0	SCKSEL: 1011	SSIWS3	
	3	—	—	—	—	—	1	Setting unnecessary	SSIWS0*1	—	—	1	—	—	0	SCKSEL: 1011	SSIWS3	
	4	—	—	1	—	—	—	Setting unnecessary	SSIWS3*1	—	—	1	—	—	0	SCKSEL: 1011	SSIWS3	
	5	—	—	1	—	—	—	Setting unnecessary	SSIWS3*1	—	—	0	—	—	1	SCKSEL: 1000	SSIWS0	
	6	—	—	1	—	—	—	Setting unnecessary	SSIWS3*1	—	—	0	—	—	1	SCKSEL: 1000	SSIWS0	
	7	—	0	—	—	1	—	Setting unnecessary	SSIWS1*1	—	0	0	—	0	1	SCKSEL: 1000	SSIWS0	
	8	—	0	—	—	1	—	Setting unnecessary	SSIWS1*1	—	0	1	—	0	0	SCKSEL: 1011	SSIWS3	
	9	—	0	—	—	1	—	Setting unnecessary	SSIWS1*1	—	0	1	—	0	0	SCKSEL: 1011	SSIWS3	
	10	—	1	—	—	0	—	Setting unnecessary	SSIWS4*1	—	0	0	—	0	1	SCKSEL: 1000	SSIWS0	
	11	—	1	—	—	0	—	Setting unnecessary	SSIWS4*1	—	0	0	—	0	1	SCKSEL: 1000	SSIWS0	
	12	—	1	—	—	0	—	Setting unnecessary	SSIWS4*1	—	0	1	—	0	0	SCKSEL: 1011	SSIWS3	
	13	0	—	—	1	—	—	Setting unnecessary	SSIWS2*1	0	—	0	0	—	1	SCKSEL: 1000	SSIWS0	
	14	0	—	—	1	—	—	Setting unnecessary	SSIWS2*1	0	—	1	0	—	0	SCKSEL: 1011	SSIWS3	
	15	0	—	—	1	—	—	Setting unnecessary	SSIWS2*1	0	—	1	0	—	0	SCKSEL: 1011	SSIWS3	
	16	1	—	—	0	—	—	Setting unnecessary	SSIWS5*1	0	—	0	0	—	1	SCKSEL: 1000	SSIWS0	
	17	1	—	—	0	—	—	Setting unnecessary	SSIWS5*1	0	—	0	0	—	1	SCKSEL: 1000	SSIWS0	
	18	1	—	—	0	—	—	Setting unnecessary	SSIWS5*1	0	—	1	0	—	0	SCKSEL: 1011	SSIWS3	
FFD → SRC → SSIF	1	—	—	—	—	—	0	Arbitrary	←	—	—	0	—	—	1	Setting unnecessary	SSIWS0*1	
	2	—	—	—	—	—	0	Arbitrary	←	—	—	0	—	—	1	Setting unnecessary	SSIWS0*1	
	3	—	—	—	—	—	0	Arbitrary	←	—	—	1	—	—	0	Setting unnecessary	SSIWS3*1	
	4	—	—	—	—	—	0	Arbitrary	←	—	—	1	—	—	0	Setting unnecessary	SSIWS3*1	
	5	—	—	0	—	—	—	Arbitrary	←	—	—	1	—	—	0	Setting unnecessary	SSIWS3*1	
	6	—	—	0	—	—	—	Arbitrary	←	—	—	1	—	—	0	Setting unnecessary	SSIWS3*1	
	7	—	—	0	—	—	—	Arbitrary	←	—	—	0	—	—	1	Setting unnecessary	SSIWS0*1	
	8	—	—	0	—	—	—	Arbitrary	←	—	—	0	—	—	1	Setting unnecessary	SSIWS0*1	
	9	—	0	—	—	0	—	Arbitrary	←	—	0	—	—	1	—	Setting unnecessary	SSIWS1*1	
	10	—	0	—	—	0	—	Arbitrary	←	—	1	—	—	0	—	Setting unnecessary	SSIWS4*1	
	11	0	—	—	0	—	—	Arbitrary	←	0	—	—	1	—	—	Setting unnecessary	SSIWS2*1	
	12	0	—	—	0	—	—	Arbitrary	←	1	—	—	0	—	—	Setting unnecessary	SSIWS5*1	

Note 1. Automatically selected regardless of the FDTSEL<sub>n</sub>\_CIM and FUTSEL<sub>n</sub>\_CIM settings.

Table 48.20 Input/Output Timing Setting (3)

Route	No.	Input Timing Setting							FDTSELn_CIM	Input Timing Signal	Output Timing Setting							FUTSELn_CIM	Output Timing Signal
		SSICTRL_CIM					SSIO0RX	SSIO1TX			SSICTRL_CIM					SSIO1TX	FUTSELn_CIM		
		SSI5RX	SSI4RX	SSI3RX	SSI2RX	SSI1RX					SSI5TX	SSI4TX	SSI3TX	SSI2TX	SSI1TX				
FFD → SRC → FFU	1	—	—	—	—	—	—	—	Unnecessary	—	—	—	—	—	—	—	Unnecessary		
	2	—	—	—	—	—	0	Arbitrary	←	—	—	0	—	—	0	Arbitrary	←		
	3	—	—	—	—	—	—	—	Unnecessary	—	—	—	—	—	—	—	Unnecessary		
	4	—	—	0	—	—	—	Arbitrary	←	—	—	0	—	—	0	Arbitrary	←		
	5	—	—	—	—	—	—	—	Unnecessary	—	—	—	—	—	—	—	Unnecessary		
	6	—	0	—	—	0	—	Arbitrary	←	—	0	—	—	0	—	Arbitrary	←		
	7	—	—	—	—	—	—	—	Unnecessary	—	—	—	—	—	—	—	Unnecessary		
	8	0	—	—	0	—	—	Arbitrary	←	0	—	—	0	—	—	Arbitrary	←		
FFD → SRC → MIX → SSIF	1	—	—	—	—	—	0	Arbitrary	←	—	—	0	—	—	1	SCKSEL: 1000	SSIWS0		
	2	—	—	—	—	—	0	Arbitrary	←	—	—	0	—	—	1	SCKSEL: 1000	SSIWS0		
	3	—	—	—	—	—	0	Arbitrary	←	—	—	1	—	—	0	SCKSEL: 1011	SSIWS3		
	4	—	—	—	—	—	0	Arbitrary	←	—	—	1	—	—	0	SCKSEL: 1011	SSIWS3		
	5	—	—	0	—	—	—	Arbitrary	←	—	—	0	—	—	1	SCKSEL: 1000	SSIWS0		
	6	—	—	0	—	—	—	Arbitrary	←	—	—	0	—	—	1	SCKSEL: 1000	SSIWS0		
	7	—	—	0	—	—	—	Arbitrary	←	—	—	1	—	—	0	SCKSEL: 1011	SSIWS3		
	8	—	—	0	—	—	—	Arbitrary	←	—	—	1	—	—	0	SCKSEL: 1011	SSIWS3		
	9	—	0	—	—	0	—	Arbitrary	←	—	0	0	—	0	1	SCKSEL: 1000	SSIWS0		
	10	—	0	—	—	0	—	Arbitrary	←	—	0	0	—	0	1	SCKSEL: 1000	SSIWS0		
	11	—	0	—	—	0	—	Arbitrary	←	—	0	1	—	0	0	SCKSEL: 1011	SSIWS3		
	12	—	0	—	—	0	—	Arbitrary	←	—	0	1	—	0	0	SCKSEL: 1011	SSIWS3		
	13	0	—	—	0	—	—	Arbitrary	←	0	—	0	0	—	1	SCKSEL: 1000	SSIWS0		
	14	0	—	—	0	—	—	Arbitrary	←	0	—	0	0	—	1	SCKSEL: 1000	SSIWS0		
	15	0	—	—	0	—	—	Arbitrary	←	0	—	1	0	—	0	SCKSEL: 1011	SSIWS3		
	16	0	—	—	0	—	—	Arbitrary	←	0	—	1	0	—	0	SCKSEL: 1011	SSIWS3		

Note 1. Automatically selected regardless of the FDTSELn\_CIM and FUTSELn\_CIM settings.

### 48.4.7 2SRC (SRC) Block

The 2SRC block consists of two SRCs (Sampling Rate Converters) which implement the sampling rate conversion function. Each SRC has two modes allowing either synchronous or asynchronous sampling rate conversion to be performed.

- Asynchronous or synchronous sampling rate conversion is possible\*<sup>1</sup>
- Sampling rate (synchronous mode)\*<sup>2</sup>  
Input [KHz]: 8, 11.025, 12, 16, 22.05, 24, 32, 44.1, 48, 64, 88.2, or 96 is selectable  
Output [KHz]: 8, 16, 24, 32, 44.1, 48, or 96 is selectable
- Sampling rate (asynchronous mode)\*<sup>2</sup>  
Input/output [KHz]: 1 to 96
- The supported number of bits is 16 bits and 24 bits
- Sound quality: -132 dB or less\*<sup>3</sup>
- Supports 1, 2, 4, 6, or 8 channels\*<sup>4</sup>
- DMA transfer with on-chip memory or external memory and direct transfer with the SSIF module are possible

Note 1. The synchronous mode can be selected only when the SCUX is connected to the FFD and FFU modules.

Note 2. The selectable sampling rates depend on the number of used channels and rate ratio.

Note 3. The data format is a 24-bit value.

Note 4. The number of selectable channels depends on the sampling rate and route.

Figure 48.13 shows a block diagram of the 2SRC (SRC) block. Input audio data indicates the data before executing the SRC function and output audio data indicates the data after executing the SRC function.

In synchronous SRC mode, the 2SRC (SRC) block outputs the input request signal to the FFD block and outputs the output request signal to the FFU block. In asynchronous SRC mode, the 2SRC (SRC) block receives the input timing signal and output timing signal from an external module and automatically detects the input sampling rate and output sampling rate from these timing signals.

Each SRC in the 2SRC (SRC) block can simultaneously process audio data for a maximum of eight channels. However, the maximum sampling rate that can be handled depends on the number of channels used. When SRC2 or SRC3 is used by the route from the FFD or the route to the FFU, the maximum number of processable channels is two. The number of channels is set by `SADIRp_2SRC0_m.CHNUM`. When not in bypass mode (`SRCBRp_2SRC0_m.BYPASS = 0`), unused channels always output 0 after initialization (`SRCIRp_2SRC0_m.INIT = 1`) unless the `CHNUM` setting is changed. In bypass mode, the input audio data is output one cycle later in all channels regardless of the `CHNUM` setting.

The bit width of output audio data can be set to 16 bits or 24 bits by `SADIRp_2SRC0_m.OTBL`. For input audio data, audio data with the same bit width as the `OTBL` bit setting in Table 48.21 should be input.

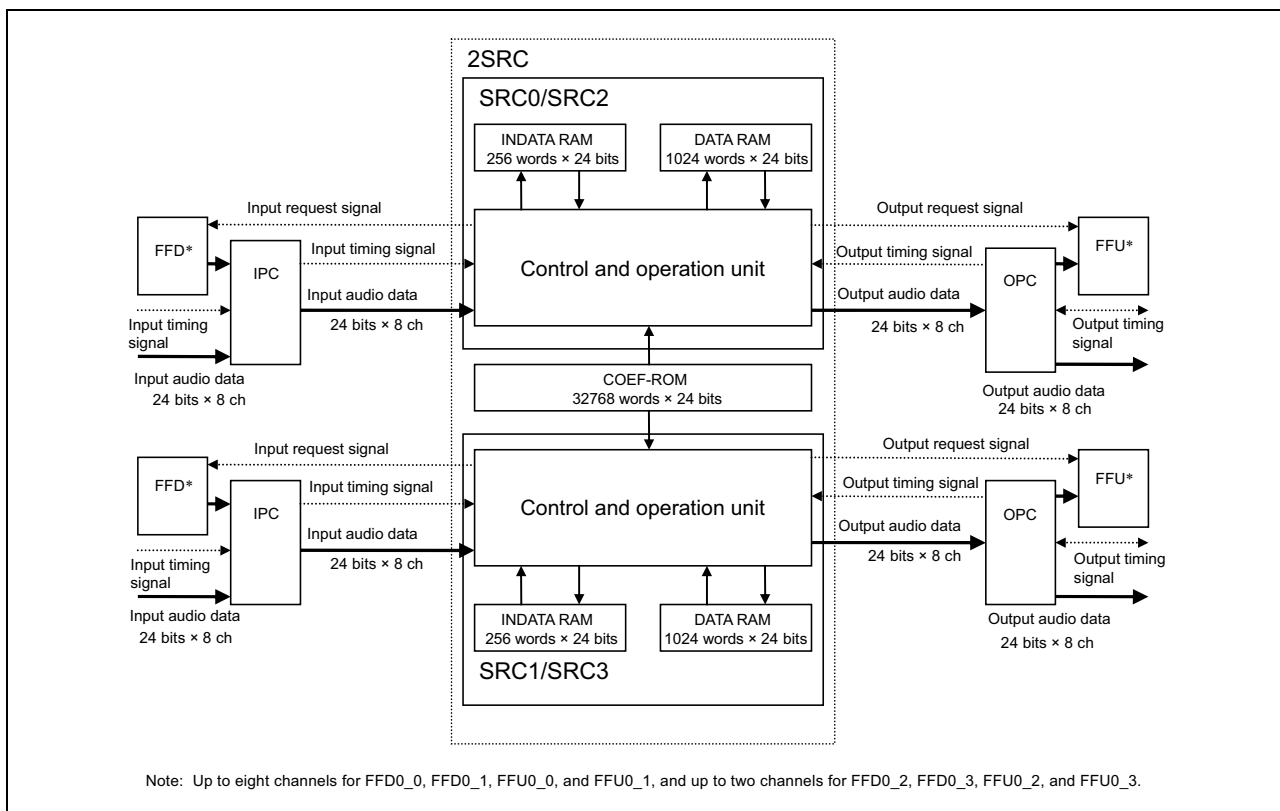


Figure 48.13 Block Diagram of 2SRC Block

Table 48.21 Combinations of Bit Width of Input/Output Audio Data

SADIRp_2SRC0_m.OTBL	Bit Width of Input Audio Data	Bit Width of Output Audio Data
00000 = 24 bits	24 bits	24 bits
01000 = 16 bits	16 bits	16 bits

Table 48.22 shows the functions of the 2SRC (SRC) block.

Table 48.22 2SRC (SRC) Block Functions

Item	Performance					
Type of SRC	Asynchronous SRC/synchronous SRC					
Operation frequency	66 MHz					
Delay mode	Normal				Low delay 1	Low delay 2
Channel number	1 or 2	4	6	8	1 or 2	1 or 2
Sampling rate of input source	8k to 96k [Hz]	8k to 96k [Hz]	8k to 66k [Hz]	8k to 49k [Hz]	8k to 96k [Hz]	8k to 96k [Hz]
Sampling rate of output source	8k to 96k [Hz]	8k to 96k [Hz]	8k to 66k [Hz]	8k to 49k [Hz]	8k to 96k [Hz]	8k to 96k [Hz]
Ratio of input and output sampling rate (FSO/FSI)	16 to 0.125 [time]	16 to 0.25 [time]	16 to 0.375 [time]	16 to 0.5 [time]	16 to 0.5 [time]	16 to 1 [time]
Sound quality	-132 dB					

Table 48.23 shows the latency of the 2SRC (SRC) block.

**Table 48.23 2SRC (SRC) Block Latency**

Item	Bypass	Normal					Low Delay 1		Low Delay 2	
		1	2	4	6	8	1	2	1	2
Channel number	Any	1	2	4	6	8	1	2	1	2
Processing delay	0	641	321	161	102	81	81	81	49	49
Logic delay	1	3	3	3	3	3	3	3	3	3
Output delay	Output delay [sample] = (Processing delay) * (FSO/FSI ratio) + (Logic delay)									

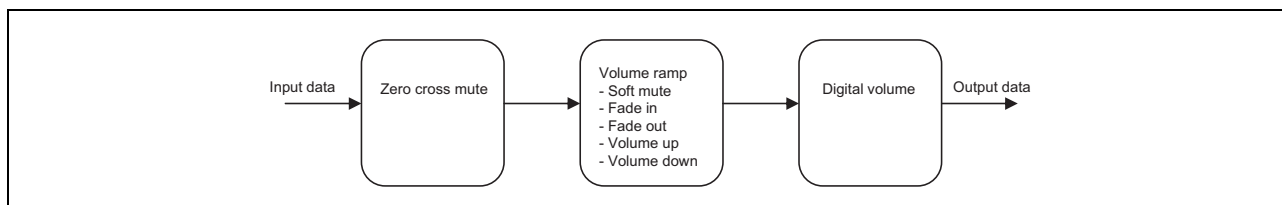
#### 48.4.8 DVU Block

The DVU block is used to control volume and mute. This block can simultaneously process data for a maximum of eight channels.

- Digital volume, volume ramp, and zero cross mute are provided as functions to adjust the volume
- The digital volume is set as a 24-bit fixed-point value within the range from a multiple of 0 to 8 (mute, -120 to 18 dB)
- Volume ramp can be used to perform soft mute, fade in, fade out, and volume change as desired
- The ramp time of volume ramp can be changed and set within the sampling range of  $2^0$  to  $2^{23}$
- Zero cross mute turns the sound mute at the zero cross point of audio data
- Direct transfer with the SSIF module and transfer with the mixer are possible

The number of channels is set by VADIR\_DVU0\_n.CHNUM. For unused channels, the input data that is input to the corresponding channels is output one cycle later. In bypass mode, the input data is output one cycle later in all channels regardless of the CHNUM setting.

Figure 48.14 shows the processing image of the DVU block.



**Figure 48.14 Processing Image of DVU Block**



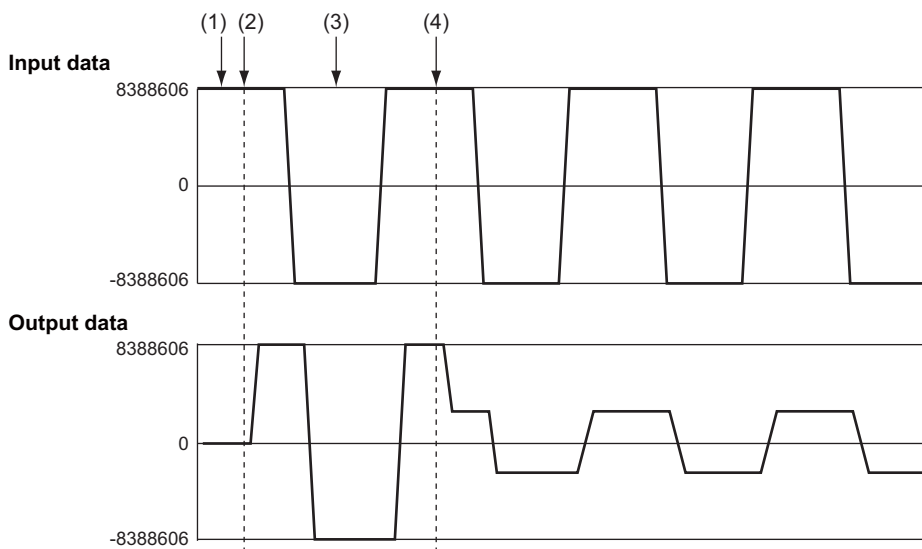
Table 48.24 shows the functions of the DVU block.

**Table 48.24 DVU Block Functions**

Item	Performance
Digital volume	Range: -120 dB to 18 dB ( $9.5 \times 10^{-7}$ -time to 8-time)
Volume ramp	Volume ramp use for many kinds of operation (soft mute, fade in, fade out, volume change by ramp) Ramp time: $2^0/fso$ to $2^{23}/fso$ Example: ( 1/fso: -128 dB/1 step) ( 2/fso: -64 dB/1 step) ( 4/fso: -32 dB/1 step) ..... ( 128/fso: -1 dB/1 step) ( 256/fso: -0.5 dB/1 step) ( 512/fso: -0.25 dB/1 step) ( 1024/fso: -0.125 dB/1 step) ( 2048/fso: -0.125 dB/2 steps) ( 4096/fso: -0.125 dB/4 steps) ..... ( 8388608/fso: -0.125 dB/8192 steps)
Zero cross mute	Mute the signal at zero cross point

Figure 48.15 shows the image of digital volume operation.

DVUBR\_DVU0\_n.BYPASS = 0  
 DVUCR\_DVU0\_n.VVMD = 1 (digital volume mode)  
 VOL0R\_DVU0\_n.VOLVALm (m = arbitrary value from 0 to 7): 24'H10\_0000 (0 dB) → 24'H05\_0F45 (-10 dB)



**Figure 48.15 Digital Volume Operation**

### Operation

- (1) In the initialized state ( $DVUIR\_DVU0\_n.INIT = 1$ ), set the  $VVMD$  bit in  $DVUCR\_DVU0\_n$  to 1 (digital volume mode) and the  $VOLVALm$  bit in  $VOL0R\_DVU0\_n$  to  $24'H10\_0000$  (0 dB). Until the initialized state is cleared, the DVU does not start operation and its output data is 0.
- (2) When the initialized state is cleared ( $DVUIR\_DVU0\_n.INIT = 0$ ), the DVU starts operation. Since the  $VOLVALm$  bit is set to 0 dB, the output data is the same size as the input data.
- (3) Set the  $DVUEN$  bit in  $DVUER\_DVU0\_n$  to 0 and the  $VOLVAL0$  bit in  $VOL0R\_DVU0\_n$  to  $24'H05\_0F45$  (-10 dB).
- (4) The setting to the  $VOLVAL0$  bit in (3) is enabled by setting the  $DVUEN$  bit in  $DVUER\_DVU0\_n$  to 1. The output data is the value obtained by attenuating the input data by -10 dB.

Figure 48.16 shows the image of volume ramp operation.

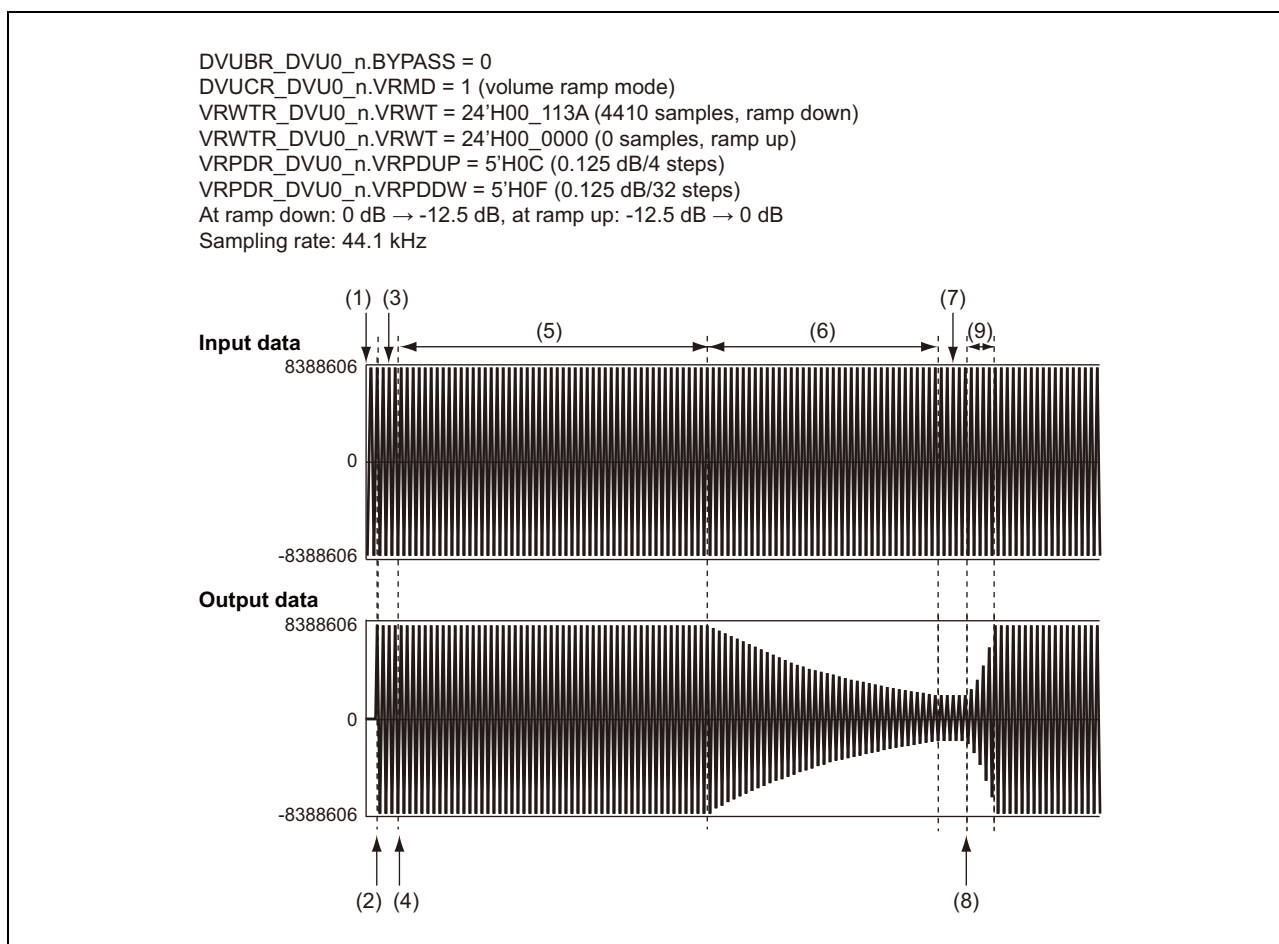


Figure 48.16 Volume Ramp Operation

### Operation

- (1) In the initialized state ( $DVUIR\_DVU0\_n.INIT = 1$ ), set the VRMD bit in  $DVUCR\_DVU0\_n$  to 1 (volume ramp mode) and the VRENm (m: arbitrary value from 0 to 7) bit in  $VRCTR\_DVU0\_n$  to 1. Until the initialized state is cleared, the DVU does not start operation and its output data is 0.
  - (2) When the initialized state is cleared ( $DVUIR\_DVU0\_n.INIT = 0$ ), the DVU starts operation.
- Note: Changes to the settings of the VRDB bit of the  $VRDBR\_DVU0\_n$  register made in the INIT state rapidly become applicable on release from the INIT state (independently of the setting of  $VRWTR\_DVU0\_n$  and  $VRPDR\_DVU0\_n$ ).
- (3) Set the DVUEN bit in  $DVUER\_DVU0\_n$  to 0, the VRPDDW bit in  $VRPDR\_DVU0\_n$  to 5'H0F (0.125 dB/32 steps), the VRDB bit in  $VRDBR\_DVU0\_n$  to 10'H064 (-12.5 dB), and the VRWT bit in  $VRWTR\_DVU0\_n$  to 24'H00\_113A (4410 samples).
  - (4) The setting in (3) is enabled by setting the DVUEN bit in  $DVUER\_DVU0\_n$  to 1.
  - (5) The VDU waits to start the volume ramp operation for  $4410/44100 = 0.1$  second because the sampling rate is set to 44.1 kHz with  $VRWTR\_DVU0\_n.VRWT = 24'H00_113A$  (4410 samples).
  - (6) The VDU starts volume ramp operation after the wait time in (5).  
The output data is the value obtained by attenuating the input data by -12.5 dB after  $12.5 \times 32/0.125 = 3200$  steps because the VRPDDW bit in  $VRPDR\_DVU0\_n$  is set to 5'H0F (0.125 dB/32 steps) according to 0 dB to -12.5 dB ramp down.
  - (7) Set the DVUEN bit in  $DVUER\_DVU0\_n$  to 0, the VRPDUP bit in  $VRPDR\_DVU0\_n$  to 5'H0C (0.125 dB/4 steps), the VRDB bit in  $VRDBR\_DVU0\_n$  to 10'H000 (0 dB), and the VRWT bit in  $VRWTR\_DVU0\_n$  to 24'H00\_0000 (0 sample).
  - (8) The setting in (7) is enabled by setting the DVUEN bit in  $DVUER\_DVU0\_n$  to 1.
  - (9) The VDU starts volume ramp operation without delay because the VRWT bit in  $VRWTR\_DVU0\_n$  is set to 24'H00\_0000 (0 sample).  
The output data is the same size as the input data after  $12.5 \times 4/0.125 = 400$  steps because the VRPDUP bit in  $VRPDR\_DVU0\_n$  is set to 5'H0C (0.125 dB/4 steps) according to -12.5 dB to 0 dB ramp up.

Table 48.25 shows the latency of the DVU block.

**Table 48.25 DVU Block Latency**

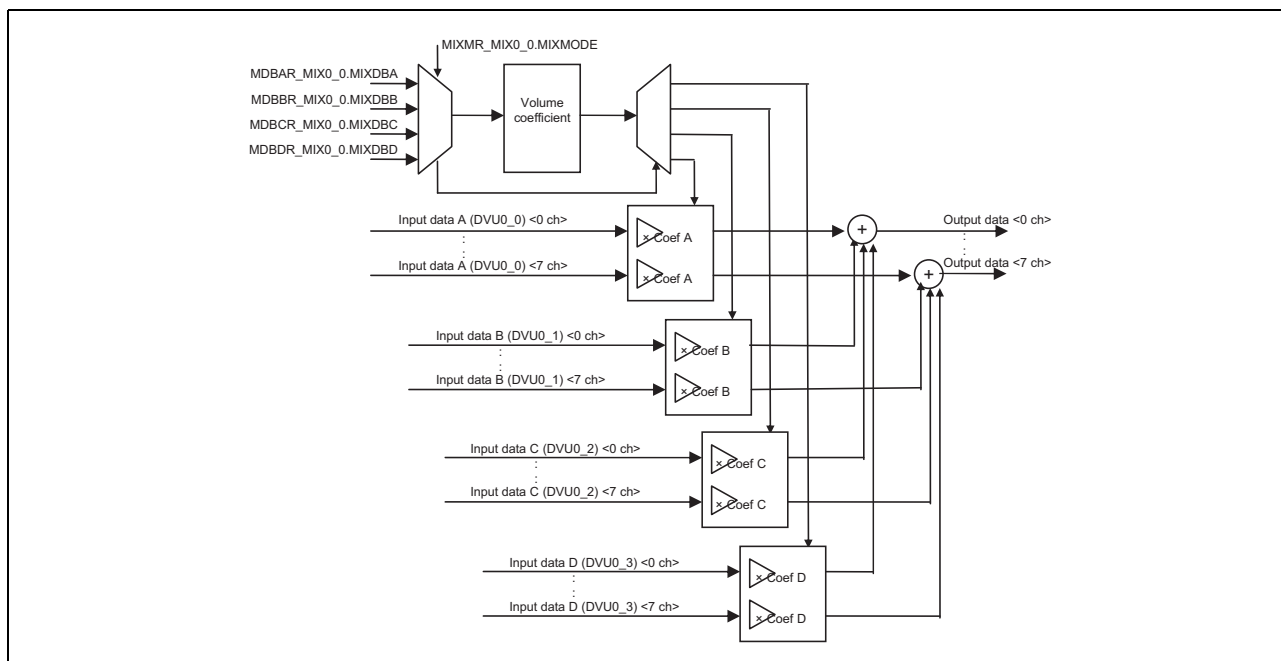
Item	Bypass	DVU Operation
Output delay	1	2

### 48.4.9 MIX Block

The MIX block is used to mix (add together) the input audio data from a maximum of four systems into one system. Each system can simultaneously process data for a maximum of eight channels.

- Data of two to four source systems can be mixed (added together) into one system
- The ratio to add the sources can be set
- The ratio can be changed dynamically
- Volume ramp enables mixing to be performed (ramp time is variable)
- Only direct transfer with the SSIF module is possible

The number of channels is set by `MADIR_MIX0_0.CHNUM`. When not in bypass mode (`MIXBR_MIX0_0.BYPASS = 0`), unused channels always output 0 after initialization (`MIXIR_MIX0_0.INIT = 1`) unless the `CHNUM` setting is changed. In bypass mode, the input data is output one cycle later in all channels regardless of the `CHNUM` setting. Figure 48.17 shows a block diagram of the MIX block.



**Figure 48.17 Block Diagram of MIX Block**

Table 48.26 shows the functions of the MIX block.

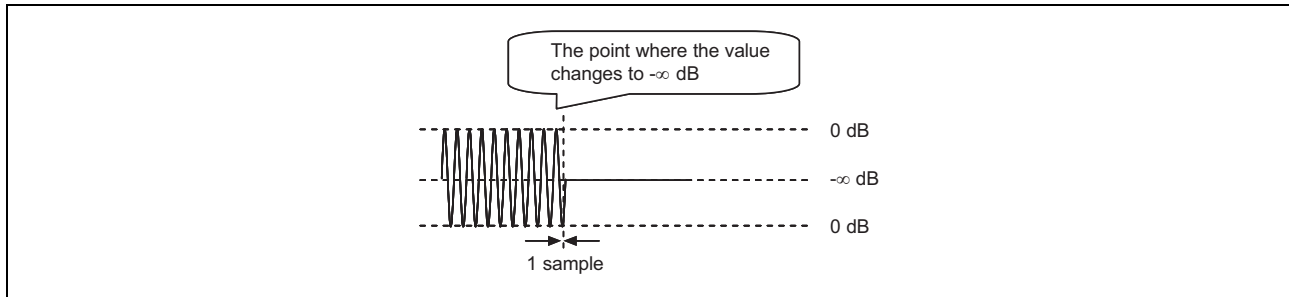
**Table 48.26 MIX Block Functions**

Item	Performance
Volume step mixer	Mix the data of four systems into one system. The volume of each system can be adjusted by step change.
Volume ramp mixer	Mix the data of four systems into one system. The volume of each system can be adjusted by ramp.

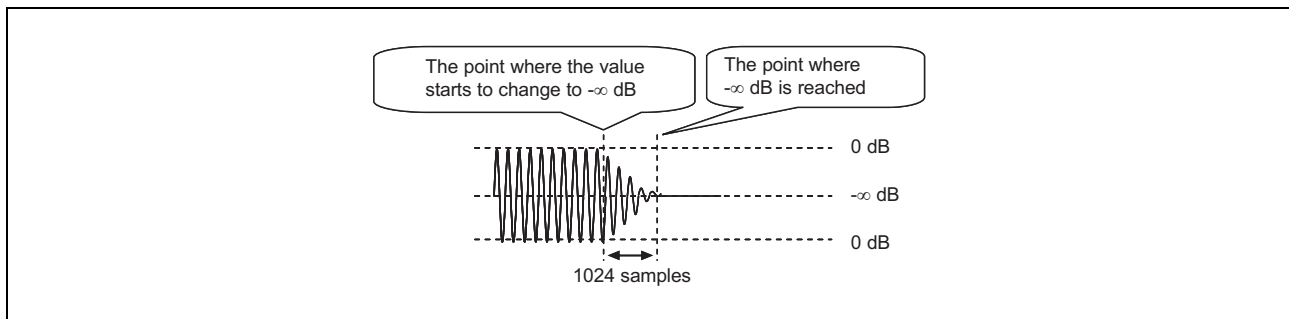
Figure 48.18 shows the image of volume step operation for 1-system data. This figure explains about the operation when the volume is changed from 0 dB to  $-\infty$  dB. The volume can change the target dB for one step each. In the case of the volume step mixer, the volume of four systems can be adjusted at each step change (one sample).

Figure 48.19 shows the image of volume ramp operation for 1-system data. This figure explains about the operation when the volume is changed from 0 dB to  $-\infty$  dB. The volume can be changed by 0.125-dB/step each and it takes 1024 samples to reach  $-\infty$  dB in the maximum case for each system. In the case of the volume ramp mixer, the volume of four

systems can be adjusted at each ramp. In addition, the volume ramp mixer has a mechanism to reduce the noise. If the direction of volume is different among the four systems, it operates from the volume-down systems and after that operates on the volume-up systems automatically. In the case of the same direction, it changes the volume at the same timing.



**Figure 48.18** Volume Step Operation



**Figure 48.19** Volume Ramp Operation

Figure 48.20 shows the image of volume ramp operation for 4-system data.

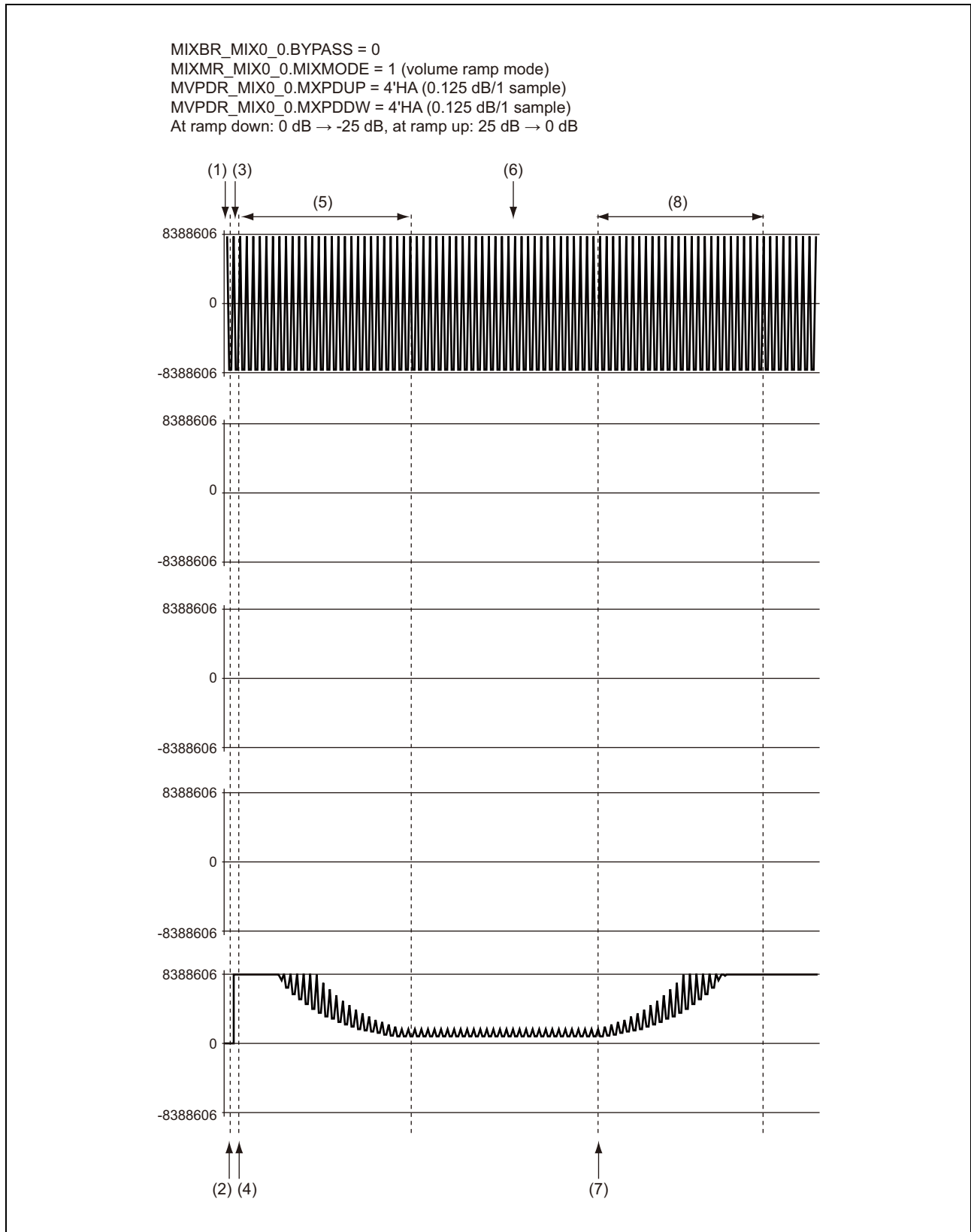


Figure 48.20 Volume Ramp Operation for 4-System Data

### Operation

- (1) In the initialized state (MIXIR\_MIX0\_0.INIT = 1), set the MIXMODE bit in MIXMR\_MIX0\_0 to 1 (volume ramp mode) and the MXPDDW bit in MVPDR\_MIX0\_0 to 4'HA (0.125 dB/1 sample).

Until the initialized state is cleared, the MIX does not start operation and its output data is 0.

- (2) When the initialized state is cleared (MIXIR\_MIX0\_0.INIT = 0), the MIX starts operation.

**Note:** Changes to the settings of the MIXDBA bits of the MDBAR\_MIX0\_0 register, MIXDBB bits of the MDBBR\_MIX0\_0 register, MIXDBC bits of the MDBCR\_MIX0\_0 register and MIXDBD bits of the MDBDR\_MIX0\_0 register made in the INIT state rapidly become applicable on release from the INIT state (independently of the setting of MVPDR\_MIX0\_0).

- (3) Set as follows:

MDBER\_MIX0\_0.MIXDBEN = 0

MDBAR\_MIX0\_0.MIXDBA = 10'H0C8 (-25 dB)

MDBBR\_MIX0\_0.MIXDBB = 10'H0C8 (-25 dB)

MDBCR\_MIX0\_0.MIXDBC = 10'H0C8 (-25 dB)

MDBDR\_MIX0\_0.MIXDBD = 10'H0C8 (-25 dB)

- (4) The setting in (3) is enabled by setting the MIXDBEN bit in MDBER\_MIX0\_0 to 1.

- (5) The output data is the value obtained by attenuating and mixing the input data of each system by -25 dB after  $25/0.125 = 200$  steps because the MXPDDW bit in MVPDR\_MIX0\_0 is set to 4'HA (0.125 dB/1 sample) according to 0 dB to -25 dB ramp down for each system.

- (6) Set as follows:

MDBER\_MIX0\_0.MIXDBEN = 0

MDBAR\_MIX0\_0.MIXDBA = 10'H000 (0 dB)

MDBBR\_MIX0\_0.MIXDBB = 10'H000 (0 dB)

MDBCR\_MIX0\_0.MIXDBC = 10'H000 (0 dB)

MDBDR\_MIX0\_0.MIXDBD = 10'H000 (0 dB)

- (7) The setting in (6) is enabled by setting the MIXDBEN bit in MDBER\_MIX0\_0 to 1.

- (8) The output data is the value obtained by boosting and mixing the input data of each system by 25 dB after  $25/0.125 = 200$  steps because the MXPDU bit in MVPDR\_MIX0\_0 is set to 4'HA (0.125 dB/1 sample) according to -25 dB to 0 dB ramp up for each system.

Table 48.27 shows the latency of the MIX block.

**Table 48.27 MIX Block Latency**

Item	Bypass	MIX Operation
Output delay	1	2

## 49. Sound Generator

This LSI has four channels of on-chip sound generator.

### 49.1 Features

- Capable of adjusting sound volume using 8-bit PWM output
- Selection of operating clocks  
Four types of operating clocks ( $P0\phi/2$ ,  $P0\phi/4$ ,  $P0\phi/8$ , and  $P0\phi/16$ ) can be selected.
- Frequency settings in the 25-Hz to 20-kHz range with precision of 1% or less
- Output stop procedures can be selected
- Automatic attenuator function can be selected
- Interrupt source: one type  
Attenuation end interrupt can be requested
- Module stop mode can be set

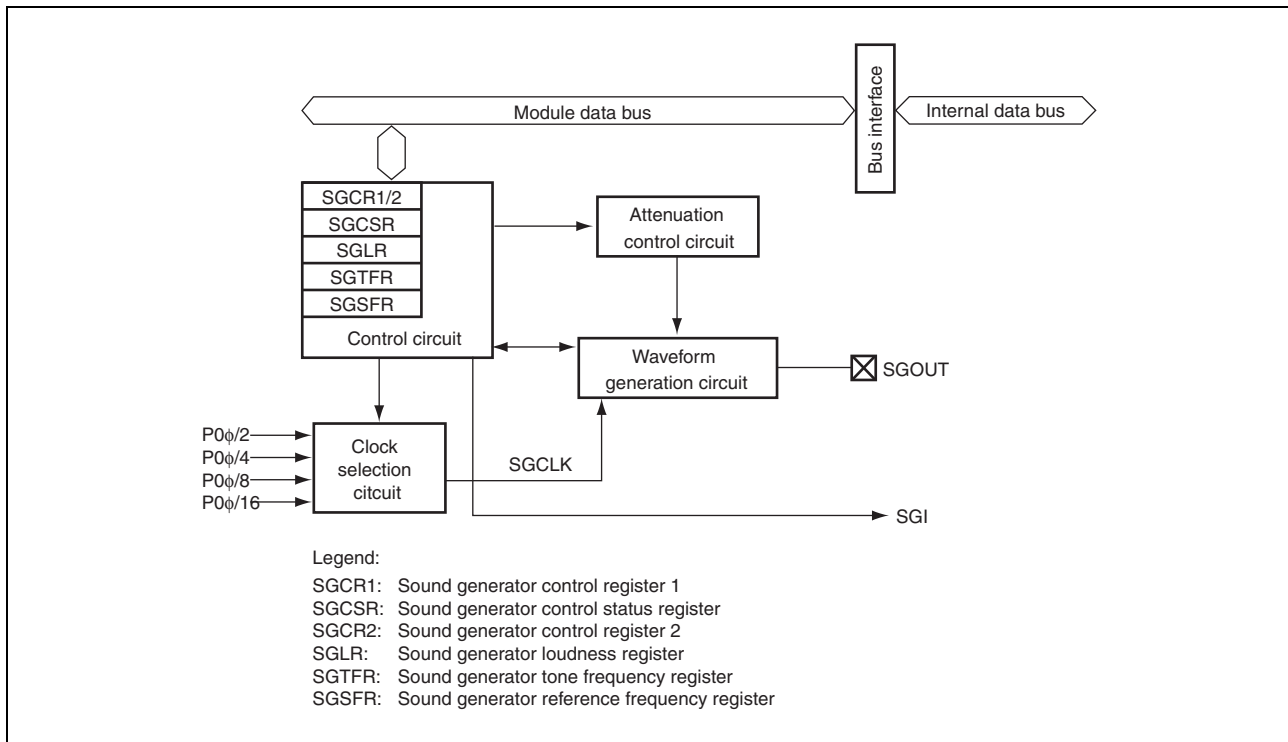


Figure 49.1 Block Diagram



## 49.2 Input/Output Pins

Table 49.1 shows the pin configuration.

**Table 49.1 Pin Configuration**

Name	Abbr.	I/O	Function
Sound generator output pin 0	SGOUT_0	Output	Channel 0 sound generator output
Sound generator output pin 1	SGOUT_1	Output	Channel 1 sound generator output
Sound generator output pin 2	SGOUT_2	Output	Channel 2 sound generator output
Sound generator output pin 3	SGOUT_3	Output	Channel 3 sound generator output

## 49.3 Register Descriptions

Table 49.2 shows the register configuration. Separate explanations for each channel are not given in this section.

**Table 49.2 Register Configuration**

Channel	Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
0	Sound generator control register 1_0	SGCR1_0	R/W	H'00	H'FCFF4800	8
	Sound generator control status register_0	SGCSR_0	R/W	H'00	H'FCFF4801	8
	Sound generator control register 2_0	SGCR2_0	R/W	H'00	H'FCFF4802	8
	Sound generator loudness register_0	SGLR_0	R/W	H'00	H'FCFF4803	8
	Sound generator tone frequency register_0	SGTFR_0	R/W	H'00	H'FCFF4804	8
	Sound generator reference frequency register_0	SGSFR_0	R/W	H'00	H'FCFF4805	8
1	Sound generator control register 1_1	SGCR1_1	R/W	H'00	H'FCFF4A00	8
	Sound generator control status register_1	SGCSR_1	R/W	H'00	H'FCFF4A01	8
	Sound generator control register 2_1	SGCR2_1	R/W	H'00	H'FCFF4A02	8
	Sound generator loudness register_1	SGLR_1	R/W	H'00	H'FCFF4A03	8
	Sound generator tone frequency register_1	SGTFR_1	R/W	H'00	H'FCFF4A04	8
	Sound generator reference frequency register_1	SGSFR_1	R/W	H'00	H'FCFF4A05	8
2	Sound generator control register 1_2	SGCR1_2	R/W	H'00	H'FCFF4C00	8
	Sound generator control status register_2	SGCSR_2	R/W	H'00	H'FCFF4C01	8
	Sound generator control register 2_2	SGCR2_2	R/W	H'00	H'FCFF4C02	8
	Sound generator loudness register_2	SGLR_2	R/W	H'00	H'FCFF4C03	8
	Sound generator tone frequency register_2	SGTFR_2	R/W	H'00	H'FCFF4C04	8
	Sound generator reference frequency register_2	SGSFR_2	R/W	H'00	H'FCFF4C05	8
3	Sound generator control register 1_3	SGCR1_3	R/W	H'00	H'FCFF4E00	8
	Sound generator control status register_3	SGCSR_3	R/W	H'00	H'FCFF4E01	8
	Sound generator control register 2_3	SGCR2_3	R/W	H'00	H'FCFF4E02	8
	Sound generator loudness register_3	SGLR_3	R/W	H'00	H'FCFF4E03	8
	Sound generator tone frequency register_3	SGTFR_3	R/W	H'00	H'FCFF4E04	8
	Sound generator reference frequency register_3	SGSFR_3	R/W	H'00	H'FCFF4E05	8

### 49.3.1 Sound Generator Control Register 1 (SGCR1)

SGCR1 controls the operation of this module.

Bit:	7	6	5	4	3	2	1	0
Bit name:	SGST	STPM	—	SGCK[1]	SGCK[0]	DPF[2]	DPF[1]	DPF[0]
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	SGST	0	R/W	Operation Start Starts/stops the operation. 0: Stops the operation 1: Starts the operation. The stop procedures differ depending on the STPM bit setting even when SGST = 1
6	STPM	0	R/W	Stop Procedure Select Selects the operation stop procedure. 0: Stops when SGST = 0 1: When the attenuator function is on, operation stops if SGST = 0 and SGDEF = 1 When the attenuator function is off, operation stops if SGST = 0 and SGEND = 1
5	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
4 3	SGCK[1:0]	H'0	R/W R/W	Clock Select Selects the operating clock (SGCLK). 00: P0φ/2 01: P0φ/4 10: P0φ/8 11: P0φ/16
2 1 0	DPF[2:0]	H'0	R/W R/W R/W	Attenuator Function Select These bits turn the attenuator function on and off, and select the attenuation cycle. 000: Attenuator function is off. 001: Attenuates at the TONE frequency 010: Attenuates at the TONE frequency/2 011: Attenuates at the TONE frequency/4 100: Attenuates at the TONE frequency/8 101: Attenuates at the TONE frequency/16 110: Attenuates at the TONE frequency/32 111: Setting prohibited

### 49.3.2 Sound Generator Control Status Register (SGCSR)

SGCSR is a status register.

Bit:	7	6	5	4	3	2	1	0
Bit name:	SGIE	SGDEF	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/(W)*	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	SGIE	0	R/W	Interrupt Enable Enables/disables the attenuation end interrupt requests. 0: Disables interrupt requests 1: Enables interrupt requests
6	SGDEF	0	R/(W)*	Attenuation End Flag [Setting condition] • When attenuation operation ends [Clearing conditions] • When 0 is written after SGDEF = 1 is read • When SGLR is written
5 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note: \* Only 0 can be written to clear the flag.

### 49.3.3 Sound Generator Control Register 2 (SGCR2)

SGCR2 is used for the termination of this module.

Bit:	7	6	5	4	3	2	1	0
Bit name:	SGEND	TCHG	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	SGEND	0	R/W	Stop Controls the operation of this module when the attenuator function is off and STPM = 1. 0: Continues operation 1: Stops operation When STPM = 0, the SGST bit controls operation regardless of this bit setting.
6	TCHG	0	R/W	TONE Change Protect Enables/disables writing to the TONE or SFS bits. Bits TONE and SFS can be modified when TCHG = 1. 0: Disables writing to the TONE and SFS bits 1: Enables writing to the TONE and SFS bits
5 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

### 49.3.4 Sound Generator Loudness Register (SGLR)

SGLR sets the SGOUT duty.

Bit:	7	6	5	4	3	2	1	0
Bit name:	LD[7]	LD[6]	LD[5]	LD[4]	LD[3]	LD[2]	LD[1]	LD[0]
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	LD[7:0]	H'00	R/W	Loudness Data These bits store the duty data for pulse output.

### 49.3.5 Sound Generator Tone Frequency Register (SGTFR)

SGTFR sets the tone frequency.

Bit:	7	6	5	4	3	2	1	0
Bit name:	—	TONE[6]	TONE[5]	TONE[4]	TONE[3]	TONE[2]	TONE[1]	TONE[0]
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*

Note: \* These bits can be written to only when TCHG = 1.

Bit	Bit Name	Initial Value	R/W	Description
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6 to 0	TONE[6:0]	H'00	R/(W)*	Tone Frequency Setting These bits set the tone frequency based on the reference frequency specified by the SFS bits. Setting H'00 is prohibited.

Note: \* These bits can be written to only when TCHG = 1.

### 49.3.6 Sound Generator Reference Frequency Register (SGSFR)

SGSFR sets the reference frequency.

Bit:	7	6	5	4	3	2	1	0
Bit name:	SFS[7]	SFS[6]	SFS[5]	SFS[4]	SFS[3]	SFS[2]	SFS[1]	SFS[0]
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*

Note: \* These bits can be written to only when TCHG = 1.

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	SFS[7:0]	H'00	R/(W)*	Reference Frequency Setting These bits set the reference frequency based on the operating clock (SGCLK) specified by the SGCK bits in SGCR1. Setting H'00 is prohibited.

Note: \* These bits can be written to only when TCHG = 1.

## 49.4 Operation

### 49.4.1 Base Operation

#### (1) Initial Settings

Make sure that this module has stopped before setting registers.

The operation stop procedure, operating clock, and attenuator function on/off settings are specified by the STPM, SGCK, and DPF bits in SGCR1, respectively. When the attenuator function is on, the attenuation cycle is selected. The interrupt request is set by the SGIE bit in SGCSR.

#### (2) Activation

The operation of this module is enabled by setting the SGST bit in SGCR1 to 1 and clearing the SGEND bit in SGCR2 to 0. Set the TCHG bit in SGCR2 to 1 to cancel the write protect for SGSFR and SGTFR. Use bits SFS7 to SFS0 in SGSFR to select the reference frequency and bits TONE6 to TONE0 in SGTFR to select the tone frequency. SGLR specifies the output pulse duty. This module starts operation after writing to SGCR2, SGLR, SGTFR, and SGSFR is completed.

#### (3) Stopping

The operation stop procedure of this module is specified by the STPM bit in SGCR1.

When the attenuator function is off and STPM is 0, the operation is stopped by the SGST bit regardless of the SGEND bit setting. When the attenuator function is off and STPM is 1, the operation is stopped by clearing the SGST bit to 0 and setting the SGEND bit to 1.

When the attenuator function is on and STPM is 0, clearing the SGST bit to 0 stops operation even though the automatic attenuation does not end and the SGDEF bit is not set to 1. When the attenuator function is on and STPM is 1, clearing the SGST bit to 0 does not stop operation until the automatic attenuation ends and the SGDEF bit is set to 1.

The conditions for stopping this module are listed in Table 49.3 and examples of stopping operation are shown in Figure 49.2.

**Table 49.3 Stop Conditions**

Attenuator Function Off					Attenuator Function On				
STPM	SGST	SGEND	SGDEF	Operation	STPM	SGST	SGEND	SGDEF	Operation
0	0	x	x	Stopped	0	0	x	x	Stopped
0	1	x	x	Output	0	1	x	x	Output
1	0	0	x	Retained*	1	0	x	0	Retained*
1	0	1	x	Stopped	1	0	x	1	Stopped
1	1	0	x	Output	1	1	x	0	Output
1	1	1	x	Output	1	1	x	1	Output

Legend:

x: Don't care

Note: \* Previous state is held.

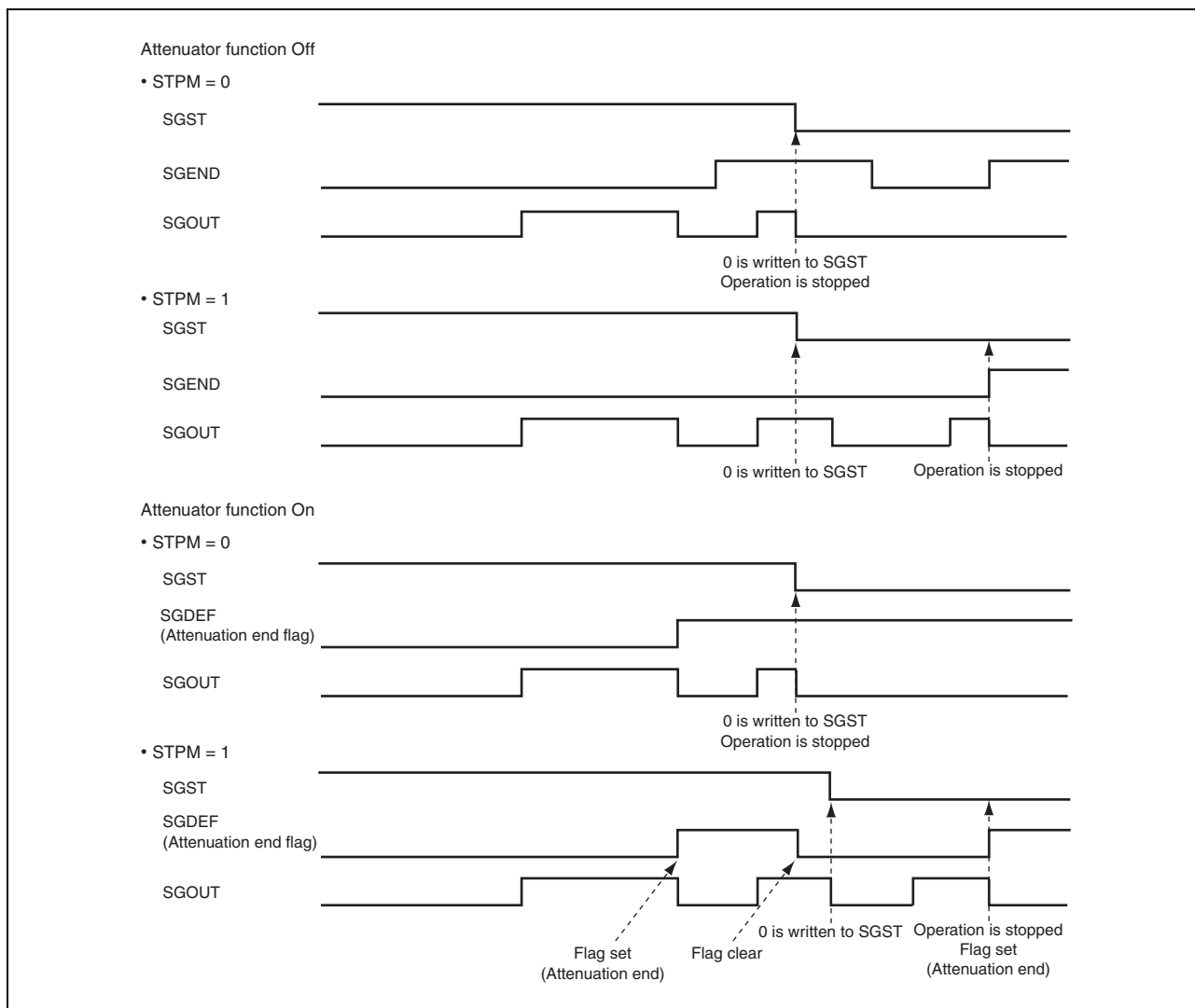


Figure 49.2 Examples of Stopping Operation

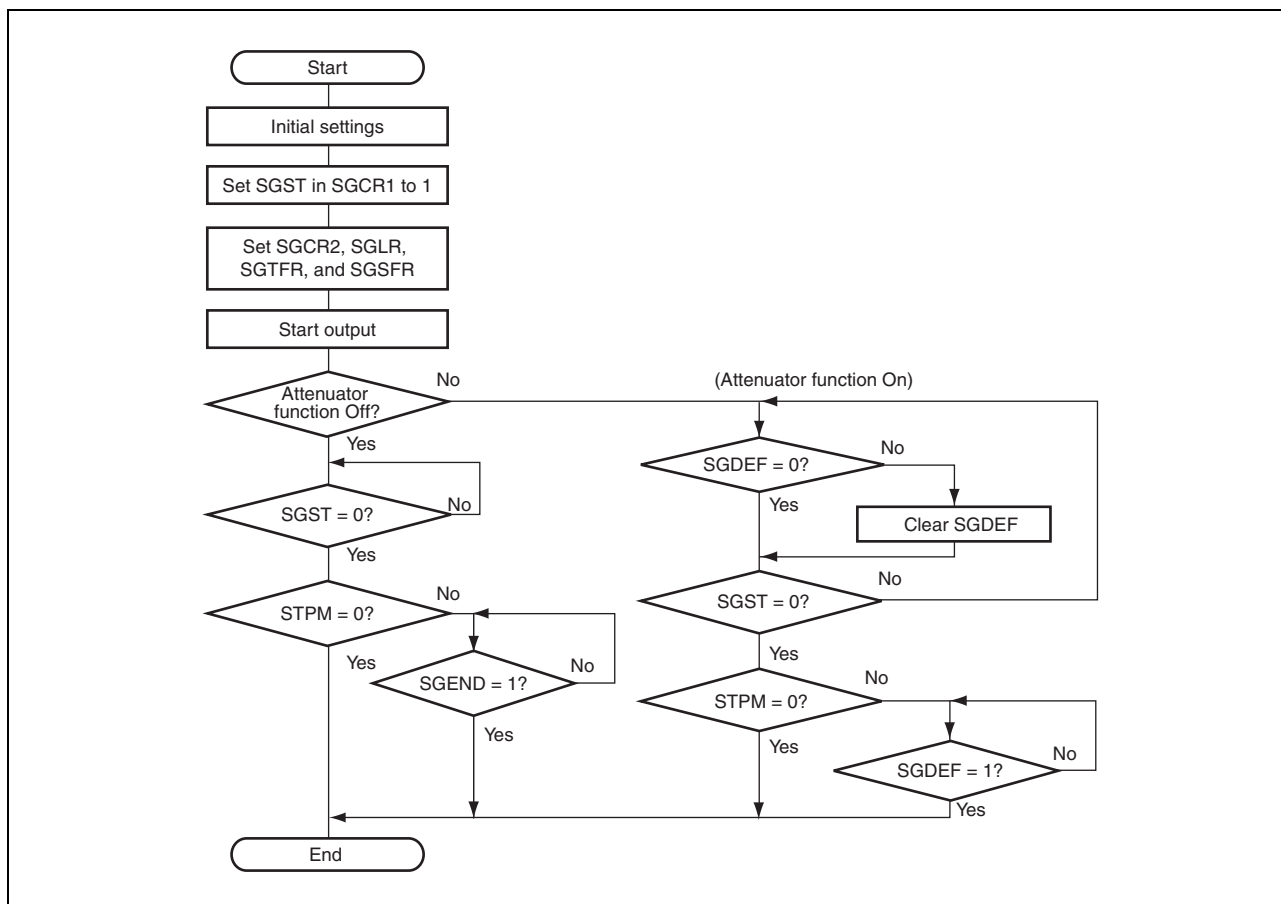


Figure 49.3 Operation Flow



### 49.4.2 Tone Frequency Setting

This module provides tone frequency output in the range of 25 Hz to 20 kHz, with precision of 1% or less. The tone frequency is calculated by:

$$\begin{aligned} \text{Reference frequency (Hz)} &= \text{SGCLK (Hz)}/\text{SFS} \\ \text{Tone frequency (Hz)} &= \text{Reference frequency (Hz)}/(2 \times \text{TONE}) \\ &= \text{SGCLK (Hz)}/(2 \times \text{SFS} \times \text{TONE}) \end{aligned}$$

Setting values of the TONE bits in SGTFR and SFS bits in SGFSR are calculated by:

$$\begin{aligned} \text{SFS} &= \text{SGCLK (Hz)}/\text{reference frequency (Hz)} \quad (0 < \text{SFS} \leq 255) \\ \text{TONE} &= \text{Reference frequency (Hz)}/(2 \times \text{TONE frequency (Hz)}) \quad (0 < \text{TONE} \leq 127) \end{aligned}$$

An example of relationship between the tone frequency and output error is shown in Table 49.4.

**Table 49.4 Relationship between Tone Frequency and Output Error**

Tone Frequency	SFS[7:0]	TONE[6:0]	Error (%)
220.00	B1	6B	0.008
329.63	9E	50	0.007
440.00	4D	7B	0.02
659.26	4F	50	0.007
880.00	40	4A	0.03
1318.50	28	4F	0.005
1760.00	20	4A	0.03
2637.00	14	4F	0.005
3520.00	10	4A	0.03
5274.00	A	4F	0.005
7040.00	8	4A	0.03

Note: • SGCLK = P0φ [MHz]/4 = 33.33 [MHz]/4 = 8.3 [MHz]

Since changing P0φ frequency affects the tone frequency, care should be taken when changing it.

### 49.4.3 Auto Attenuator Function

When the auto attenuator function is on, the loudness data (LD) determines the initial duty cycle for SGOUT. The SGOUT duty cycle is attenuated by a factor of 1/32 using the attenuation cycle set by the DPF bits in SGCR1. The attenuation characteristics are calculated by the equation:

$$LD_n = \text{int} (LD_0 \times (1 - 1/32)^n)$$

LD: SGOUT duty cycle (The initial data is SGLR.)

n: Attenuation cycle number

Figure 49.4 is a graph of attenuation characteristics.

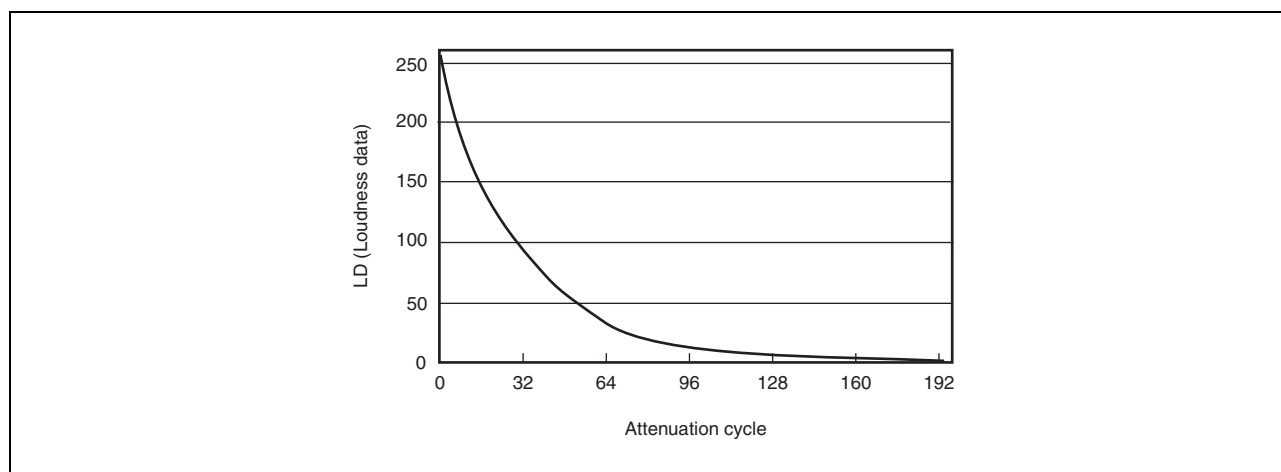


Figure 49.4 Attenuation Characteristics

### 49.4.4 Output Waveform

As shown in Figure 49.5, the output waveform of this module is obtained by synthesis of the on-chip 8-bit PWM pulse output and the tone frequency output. The duty cycle for the on-chip 8-bit PWM pulse output is set by SGLR.

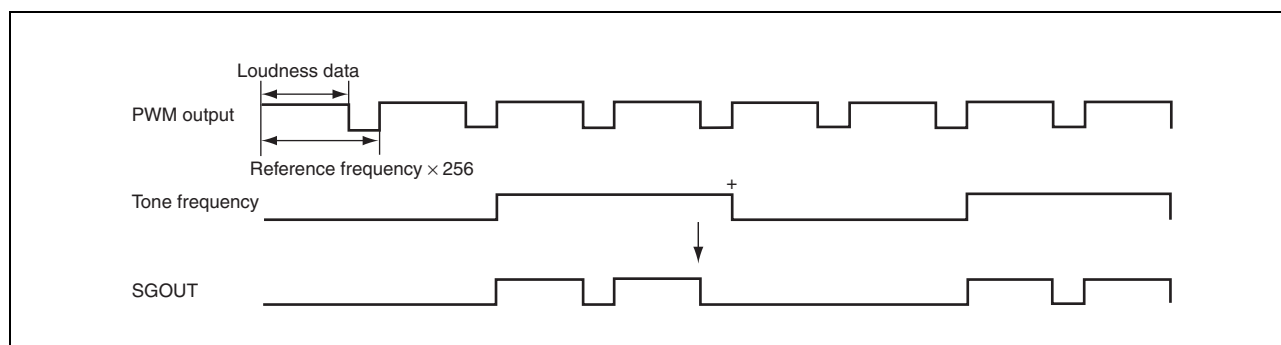


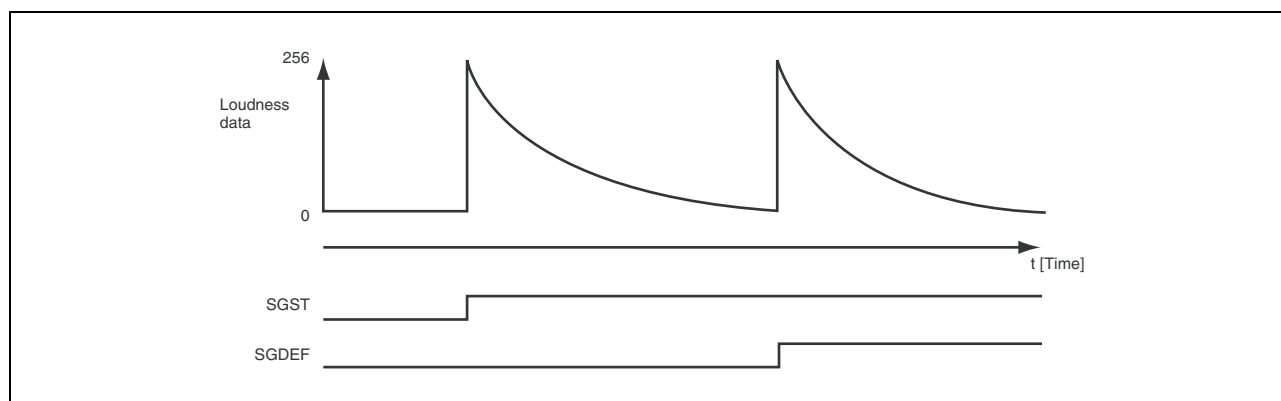
Figure 49.5 Output Waveforms

## 49.5 Interrupt Source

When the attenuator function is on and an automatic attenuation operation is completed (duty cycle for the on-chip PWM pulse output is 0), the SGDEF bit in SGCSR is set. An interrupt request is issued if the SGIE bit in SGCSR is set to 1. When STPM = 0, the SGDEF bit is set only when the first attenuation operation is completed. If the SGDEF bit is cleared during automatic attenuation, the SGDEF bit is set when the next attenuation operation is completed.

**Table 49.5 Interrupt Source**

Name	Interrupt Source	Interrupt Flag
SGDEI	Attenuation end	SGDEF



**Figure 49.6 Attenuation End Flag Setting Timing**

## 49.6 Usage Note

### 49.6.1 Module Stop Mode Settings

The module stop control register enables or disables the operation of this module. With the initial setting, the operation of this module is stopped. Register access is enabled by canceling module stop mode. For details, see [section 55, Power-Down Modes](#).

## 50. SD Host Interface

The contents of this section are available upon non-disclosure agreement.

For details, contact your local sales representatives.

For the features of the SD host interface, consult [section 1, Overview](#).

## 51. MMC Host Interface

The MMC host interface is a host controller conforming to the JEDEC Standard JESD84-A441 that allows connection with devices with the MMC interface.

### 51.1 Features

- Supports 1/4/8-bit MMC bus
- Supports the Single Data Rate only
- MMC clock frequency =  $P1\phi$  frequency/ $2^n$  ( $n = 1$  to  $10$ )
- Supports High Priority Interrupt (HPI)\*
- Supports background operation
- Data buffer: 512 bytes  $\times$  2
- Three types of interrupt requests: normal operation, error/timeout, card detection
- DMA transfer requests: buffer write and buffer read
- Card detection function

Note: \* Supports HPI in the sequences of CMD6, CMD24, CMD25 (pre-defined), and CMD38.

Figure 51.1 shows a block diagram of this module.

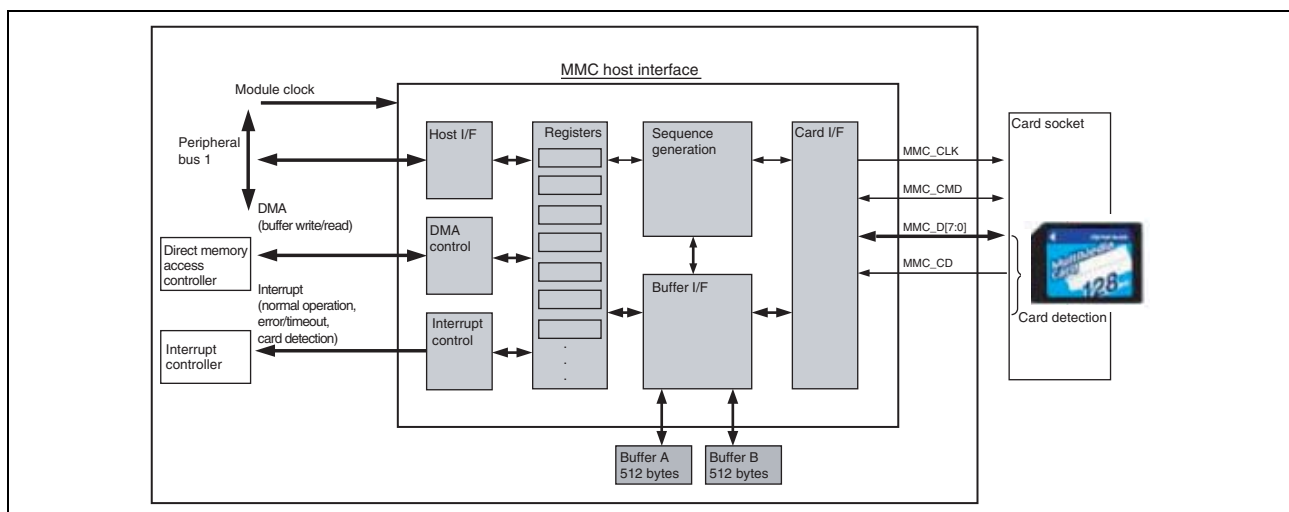


Figure 51.1 Block Diagram of MMC Host Interface

### 51.2 Input/Output Pins

Table 51.1 shows the pin configuration of this module.

Table 51.1 Pin Configuration

Pin Name	I/O	Function
MMC_CLK	Output	MMC clock
MMC_CMD	Input/output	Command/response
MMC_D[7:0]	Input/output	Transmit/receive data
MMC_CD	Input	Card detection*1

Note 1. Check the specifications of the card socket to be used before connection.

### 51.3 Register Descriptions

Table 51.2 shows the register configuration of this module.

**Table 51.2 Register Configuration**

Register Name	Abbreviation	R/W	Address	Access Size
Command setting register	CE_CMD_SET	R/W	H'E804C800	16
			H'E804C802	16
Argument register	CE_ARG	R/W	H'E804C808	32
Argument register for automatically-issued CMD12	CE_ARG_CMD12	R/W	H'E804C80C	32
Command control register	CE_CMD_CTRL	R/W	H'E804C810	32
Transfer block setting register	CE_BLOCK_SET	R/W	H'E804C814	32
Clock control register	CE_CLK_CTRL	R/W	H'E804C818	32
Buffer access configuration register	CE_BUF_ACC	R/W	H'E804C81C	32
Response register 3	CE_RESP3	R	H'E804C820	32
Response register 2	CE_RESP2	R	H'E804C824	32
Response register 1	CE_RESP1	R	H'E804C828	32
Response register 0	CE_RESP0	R	H'E804C82C	32
Response register for automatically-issued CMD12	CE_RESP_CMD12	R	H'E804C830	32
Data register	CE_DATA	R/W	H'E804C834	32
Interrupt flag register	CE_INT	R/W	H'E804C840	32
Interrupt enable register	CE_INT_EN	R/W	H'E804C844	32
Status register 1	CE_HOST_STS1	R	H'E804C848	32
Status register 2	CE_HOST_STS2	R	H'E804C84C	32
DMA mode setting register	CE_DMA_MODE	R/W	H'E804C85C	32
Card detection/port control register	CE_DETECT	R/W	H'E804C870	32
Special mode setting register	CE_ADD_MODE	R/W	H'E804C874	32
Version register	CE_VERSION	R/W	H'E804C87C	32

Note: • Do not access registers other than those shown above.

### 51.3.1 Command Setting Register (CE\_CMD\_SET)

CE\_CMD\_SET sets a command sequence.

The command sequence starts when the settings have been made in bits 31 to 16. Note that writing to CE\_CMD\_SET is disabled while a command sequence is proceeding (i.e., the value of CMDSEQ in CE\_HOST\_STS1 is 1).

CE\_CMD\_SET should be set according to the description in section 51.7.12, Setting Values of CE\_CMD\_SET. This register should be accessed in 16 bits. (The address of bits 31 to 16 is H'E804C800 and that of bits 15 to 0 is H'E804C802.)

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	CMD[5:0]					RTYP[1:0]		RBSY	—	WDAT	DWEN	CMLTE	CMD12 EN	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RIDXC[1:0]		RCRC7C[1:0]		—	CRC 16C	—	CRC STE	TBIT	OPDM	—	—	SBIT	—	DATW[1:0]	
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R	R/W	R	R/W	R/W	R/W	R	R	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31, 30	—	00	R	Reserved These bits are always read as 0. The write value should always be 0.
29 to 24	CMD[5:0]	All 0	R/W	Command Index These bits set a command index ([45:40]). Note: Setting a command index in these bits initiates the command sequence.
23, 22	RTYP[1:0]	00	R/W	Response Type 00: No response 01: 6-byte response (R1, R1b, R3, R4, R5) 10: 17-byte response (R2) 11: Setting prohibited
21	RBSY	0	R/W	Response Busy Select Selects whether "busy" is involved in response reception. 0: No response busy 1: Response busy involved (R1b)
20	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
19	WDAT	0	R/W	Presence/Absence of Data 0: No data 1: With data
18	DWEN	0	R/W	Read/Write (valid when "with data" is selected) 0: Read from the card 1: Write to the card
17	CMLTE	0	R/W	Single/Multi Block Transfer Select (valid when "with data" is selected) 0: Single-block transfer 1: Multi-block transfer
16	CMD12EN	0	R/W	Automatic CMD12 Issuance (valid when multi-block transfer is selected)* 0: Does not issue CMD12 automatically. 1: Issues CMD12 automatically (= automatic CMD12) For details of automatic CMD12 issuance, see section 51.6.4, Automatic CMD12 Issuance. Note: Set the transfer block size to 512 bytes. Set RBSY to 0.

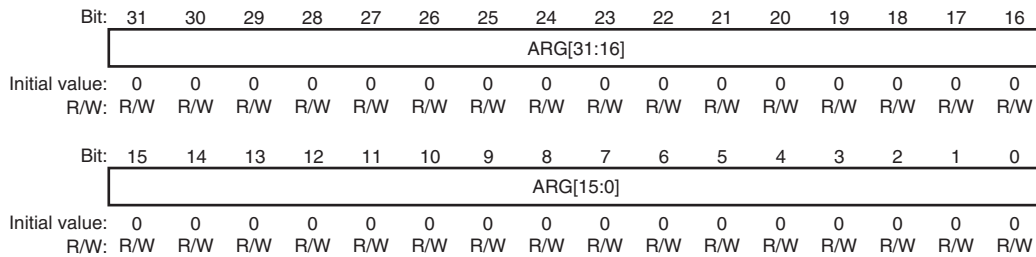


Bit	Bit Name	Initial Value	R/W	Description
15, 14	RIDXCI[1:0]	00	R/W	Response Index Check Specify the items to be checked for [45:40] of a 6-byte response or [133:128] of a 17-byte response. 00: Checks the response index (ensure that the response index matches the command index.) 01: Checks the check bits (ensure that the bits are all 1.) 10: No checking 11: Setting prohibited
13, 12	RCRC7C [1:0]	00	R/W	Response CRC7 Check Specify the items to be checked for [7:1] of a 6-byte response or a 17-byte response. 00: Checks CRC7 (set the response type to 01) 01: Checks the check bits (set the response type to 01) 10: Checks internal CRC7 (R2 only) (set the response type to 10) 11: No checking
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	CRC16C	0	R/W	CRC16 Check in Reception (valid when "with data" and "read" are selected) 0: Checks CRC16 1: Does not check CRC16 (use when CMD14)
9	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
8	CRCSTE	0	R/W	CRC Status Reception (valid when "with data" and "write" are selected) 0: Receives CRC status 1: Does not receive CRC status (use when CMD19)
7	TBIT	0	R/W	Transmission Bit Setting 0: Sets the transmission bit ([46]) to 1. 1: Sets the transmission bit ([46]) to 0.
6	OPDM	0	R/W	Open-Drain Output Mode 0: Normal output 1: Open-drain output Note: This setting is only applied to the MMC_CMD line.
5, 4	—	00	R	Reserved These bits are always read as 0. The write value should always be 0.
3	SBIT	0	R/W	Read Data Start Bit Detection (valid when "with data" and "read" are selected) 0: Detects the start bit when the MMC_D signals set by DATW are all 0. 1: Detects the start bit when MMC_D[0] is 0.
2	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
1, 0	DATW[1:0]	00	R/W	Data Bus Width Setting (valid when "with data" is selected) 00: 1 bit 01: 4 bits 10: 8 bits 11: Setting prohibited

Note: \* When HPI is executed, use multi-block transfer (pre-defined) without setting this bit to 1.

### 51.3.2 Argument Register (CE\_ARG)

CE\_ARG sets the argument for the command to be transmitted. Set CE\_ARG before starting the command sequence.

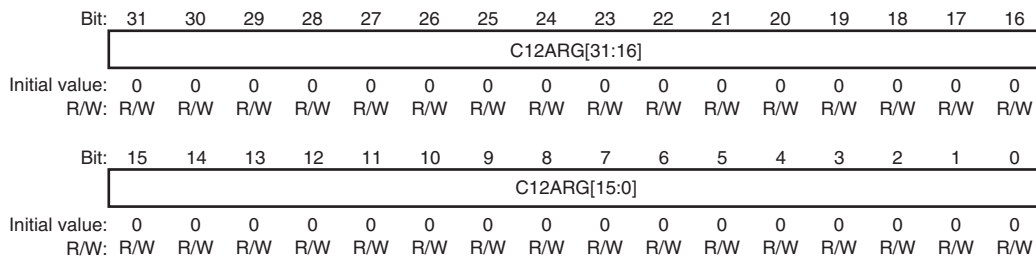


Bit	Bit Name	Initial Value	R/W	Description
31 to 0	ARG[31:0]	H'00000000	R/W	These bits set [39:8] of a command. Note: Set the argument of automatically-issued CMD12 by CE_ARG_CMD12.

### 51.3.3 Argument Register for Automatically-Issued CMD12 (CE\_ARG\_CMD12)

CE\_ARG\_CMD12 is used to set the argument for the automatically-issued CMD12 in multi-block transfer.

For the automatically-issued CMD12, see section 51.6.4, Automatic CMD12 Issuance. Set CE\_ARG\_CMD12 before starting the command sequence.



Bit	Bit Name	Initial Value	R/W	Description
31 to 0	C12ARG[31:0]	H'00000000	R/W	These bits set [39:8] of the automatically-issued CMD12.

### 51.3.4 Command Control Register (CE\_CMD\_CTRL)

CE\_CMD\_CTRL is used to terminate a command sequence forcibly.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	BREAK
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	BREAK	0	R/W	Forcible Termination of Command Sequence To discontinue the current command sequence, write 1 to this bit while the bit is 0 and then write 0. After the above procedure, check if the value of the CMDSEQ bit in CE_HOST_STS1 has become 0 and then perform a software reset. Note: Since a software reset returns the register value to the initial value, the register needs be set again.

### 51.3.5 Transfer Block Setting Register (CE\_BLOCK\_SET)

CE\_BLOCK\_SET specifies the size of the block and the number of blocks for the data to be transferred. Set CE\_BLOCK\_SET before starting the command sequence.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BLKCNT[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BLKSIZ[15:0]															
Initial value:	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 16	BLKCNT [15:0]	H'0000	R/W	Number of Blocks for Transfer Note: This setting is valid for multi-block transfer.
15 to 0	BLKSIZ [15:0]	H'0200	R/W	Transfer Block Size Note: Transfer block size should be set as follows. <ul style="list-style-type: none"> <li>• Single-block transfer: 1 to 512 bytes</li> <li>• Multi-block transfer: 512 bytes</li> </ul>

### 51.3.6 Clock Control Register (CE\_CLK\_CTRL)

CE\_CLK\_CTRL controls the MMC clock and sets timeout values. Do not change the setting of this register while a command sequence is in progress.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	CLKEN	—	—	—	—	CLKDIV[3:0]			
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R/W	R	R	R	R	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	SRSPTO[1:0]		SRBSYTO[3:0]			SRWDTO[3:0]			—	—	—	—		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 25	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
24	CLKEN	0	R/W	MMC Clock Output Control 0: Does not output the MMC clock (tied to low level) 1: Outputs the MMC clock
23 to 20	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
19 to 16	CLKDIV[3:0]	0000	R/W	MMC Clock Frequency Setting 0000: $P1\phi/2$ 0001: $P1\phi/2^2$  0111: $P1\phi/2^8$ 1000: $P1\phi/2^9$ 1001: $P1\phi/2^{10}$ 1010 to 1111: Setting prohibited
15, 14	—	00	R	Reserved These bits are always read as 0. The write value should always be 0.
13, 12	SRSPTO [1:0]	00	R/W	Response Timeout Setting Specifies the timeout period for the RSPTO bit of CE_INT. 00: 64 MMC clock cycles 01: 128 MMC clock cycles 10: 256 MMC clock cycles 11: Setting prohibited
11 to 8	SRBSYTO [3:0]	0000	R/W	Response Busy Timeout Setting Specifies the timeout period for the RBSYTO bit of CE_INT. 0000: $2^{14}$ MMC clock cycles 0001: $2^{15}$ MMC clock cycles : 1110: $2^{28}$ MMC clock cycles 1111: $2^{29}$ MMC clock cycles
7 to 4	SRWDTO [3:0]	0000	R/W	Write Data/Read Data Timeout Setting Specifies the timeout period for the WDATTO and RDATTO bits of CE_INT. 0000: $2^{14}$ MMC clock cycles 0001: $2^{15}$ MMC clock cycles : 1110: $2^{28}$ MMC clock cycles 1111: $2^{29}$ MMC clock cycles
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

### 51.3.7 Buffer Access Configuration Register (CE\_BUF\_ACC)

CE\_BUF\_ACC configures the method of accessing data registers and mode of DMA transfer. Do not change the setting of this register while a command sequence is in progress. For explanation of the buffers, see section 51.6.3, Buffer Structure and Buffer Accesses.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	DMAW EN	DMAR EN	—	—	—	—	—	—	—	ATYP
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 26	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
25	DMAWEN	0	R/W	Buffer Write DMA Transfer Request Enable 0: Disables DMA transfer request for buffer writing 1: Enables DMA transfer request for buffer writing
24	DMAREN	0	R/W	Buffer Read DMA Transfer Request Enable 0: Disables DMA transfer request for buffer reading 1: Enables DMA transfer request for buffer reading
23 to 17	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
16	ATYP	0	R/W	Buffer Access Select 0: When not swapped byte-wise. 1: When swapped byte-wise. Note: For explanation of buffer access, see section 51.6.3, Buffer Structure and Buffer Accesses.
15 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

### 51.3.8 Response Registers 3 to 0 (CE\_RESP3 to CE\_RESP0)

CE\_RESP3 to CE\_RESP0 are the registers for storing the response that has been received. For the formats of response values, see section 51.6.1, Command/Response Formats.

- CE\_RESP3

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RSP[127:112]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RSP[111:96]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	RSP[127:96]	H'00000000	R	[127:96] of a 17-byte response are stored.

- CE\_RESP2

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RSP[95:80]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RSP[79:64]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	RSP[95:64]	H'00000000	R	[95:64] of a 17-byte response are stored.

- CE\_RESP1

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RSP[63:48]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RSP[47:32]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	RSP[63:32]	H'00000000	R	[63:32] of a 17-byte response are stored.

- CE\_RESP0

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RSP[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RSP[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	RSP[31:0]	H'00000000	R	[39:8] of a 6-byte response or [31:0] of a 17-byte response are stored. Note: The response to the automatically-issued CMD12 is stored in CE_RESP_CMD12.

### 51.3.9 Response Register for Automatically-Issued CMD12 (CE\_RESP\_CMD12)

CE\_RESP\_CMD12 is the register for storing the response to the automatically-issued CMD12.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RSP12[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RSP12[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	RSP12[31:0]	H'00000000	R	[39:8] of a response to the automatically-issued CMD12 are stored.

### 51.3.10 Data Register (CE\_DATA)

CE\_DATA is used to access the buffers of this module.

For the write/read data formats, see section 51.6.2, Data Block Format.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DATA[31:16]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DATA[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DATA[31:0]	H'00000000	R/W	Buffer write/read data [31:0]

### 51.3.11 Interrupt Flag Register (CE\_INT)

CE\_INT indicates various status during execution of a command sequence. Each bit is set to 1 when its setting condition has been met. To clear flag(s), write 0 only to the bit(s) to be cleared and write 1 to the other bits.

For the operation in the case of an error or timeout, see section 51.6.7, Operation in the Case of Error/Timeout.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	CMD12 DRE	CMD12 RBE	CMD12 CRE	DTRAN E	BUFR E	BUFW EN	BUFR EN	—	—	RBSY E	CRSP E
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMD VIO	BUF VIO	—	—	WDAT ERR	RDAT ERR	RIDX ERR	RSP ERR	—	—	—	CRCS TO	WDAT TO	RDAT TO	RBSY TO	RSP TO
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The writing value should always be 0.
26	CMD12DRE	0	R/W*1	Automatic CMD12 Issuance & Buffer Read Complete [Setting condition] Response busy for automatically-issued CMD12 and buffer reading have been completed. [Clearing condition] Writing a 0 to this bit Note: When CMD12DRE has been set, CMD12RBE, CMD12CRE, and BUFRE have also been set. So, these bits should be cleared as well.
25	CMD12RBE	0	R/W*1	Automatic CMD12 Issuance Response Busy Complete [Setting condition] Reception of the response and response busy for an automatically-issued CMD12 have been completed. [Clearing condition] Writing a 0 to this bit Note: When CMD12RBE is set, CMD12CRE is also set. So, clear the bit as well. When CMD12RBE is set during a multi-block write, DTRANE is also set. So clear the bit as well.
24	CMD12CRE	0	R/W*1	Automatic CMD12 Response Complete [Setting condition] The response to an automatically-issued CMD12 has been received. [Clearing condition] Writing a 0 to this bit
23	DTRANE	0	R/W*1	Data Transmission Complete [Setting conditions] Transmission of all blocks of data has been completed. <ul style="list-style-type: none"> <li>When configured to receive CRC status: Completion of busy (data busy) after reception of CRC status</li> <li>When configured not to receive CRC status: Completion of data transmission</li> </ul> [Clearing condition] Writing a 0 to this bit
22	BUFRE	0	R/W*1	Buffer Read Complete [Setting condition] All blocks of data have been received and the data have been read from the buffer [Clearing condition] Writing a 0 to this bit



Bit	Bit Name	Initial Value	R/W	Description
21	BUFWEN	0	R/W*1	<p>Buffer Write Ready [Setting condition] The buffer has become empty and ready for writing. [Clearing condition] Writing a 0 to this bit Note: When data is written to CE_DATA by the CPU, clear this bit first, and then write the amount of data equivalent to the block size set in the CE_BLOCK_SET register. Note that this bit is not set when DMA transfer request for buffer writing is enabled.</p>
20	BUFREN	0	R/W*1	<p>Buffer Read Ready [Setting condition] Transfer block size of data have been stored in the buffer and it has become ready for reading [Clearing condition] Writing a 0 to this bit Note: When data is read from CE_DATA by the CPU, clear this bit first, and then read the amount of data equivalent to the block size set in the CE_BLOCK_SET register. Note that this bit is not set when DMA transfer request for buffer reading is enabled.</p>
19, 18	—	00	R	<p>Reserved These bits are always read as 0. The write value should always be 0.</p>
17	RBSYE	0	R/W*1	<p>Response Busy Complete [Setting condition] Reception of a response and response busy have been completed [Clearing condition] Writing a 0 to this bit Note: When RBSYE has been set, CRSPE has also been set. So, this bit should be cleared as well. Completion of reception of the response and response busy for automatically-issued CMD12 is reflected in CMD12RBE.</p>
16	CRSPE	0	R/W*1	<p>Command/Response Complete [Setting conditions] A command has been transmitted or a response has been received  <ul style="list-style-type: none"> <li>When configured not to receive response: A command has been transmitted</li> <li>When configured to receive 6- or 17-byte response: A response has been received</li> </ul>           [Clearing condition] Writing a 0 to this bit Note: Completion of reception of the response to automatically-issued CMD12 is reflected in CMD12CRE.</p>
15	CMDVIO	0	R/W*1	<p>Command Issuance Error [Setting conditions] Illegal setting has been made in CE_CMD_SET or CE_BLOCK_SET  <ul style="list-style-type: none"> <li>During execution of a command sequence: Writing to CMD[5:0] in CE_CMD_SET (The command sequence is not stopped automatically.)</li> <li>At the start of command sequence: Writing to CMD[5:0] in CE_CMD_SET when the registers have been set for one of the following combinations of selection  <ul style="list-style-type: none"> <li>No response + response busy</li> <li>No response + with data</li> <li>No data + automatic CMD12 issuance</li> <li>With data + single-block transfer + automatic CMD12 issuance</li> <li>With data + response busy + automatic CMD12 issuance</li> <li>With data + transfer block size = 0</li> <li>With data + transfer block size <math>\geq</math> 513</li> <li>With data + multi-block transfer + number of blocks for transfer = 0</li> </ul> </li> </ul>           [Clearing condition] Writing a 0 to this bit</p>

Bit	Bit Name	Initial Value	R/W	Description
14	BUFVIO	0	R/W*1	<p>Buffer Access Error [Setting conditions] Illegal buffer access has been attempted.</p> <ul style="list-style-type: none"> <li>CE_DATA has been accessed exceeding the block size set in BLKSIZ[15:0] in CE_BLOCK_SET</li> <li>While data is being read from the card: CE_DATA has been accessed with BUFREN not set (when DMA is used, with no DMA transfer request asserted for buffer reading)</li> <li>While data is being written to the card: CE_DATA has been accessed with BUFWEN not set (when DMA is used, with no DMA transfer request asserted for buffer writing)</li> </ul> <p>[Clearing condition] Writing a 0 to this bit Note: When BUFVIO is set, the command sequence is not stopped automatically. When an error occurs, this bit may be set.</p>
13, 12	—	00	R	<p>Reserved These bits are always read as 0. The write value should always be 0.</p>
11	WDATERR	0	R/W*1	<p>Write Data Error [Setting conditions] Error is found in the data that has been written.</p> <ul style="list-style-type: none"> <li>Error is in the status of the CRC status</li> <li>Error is in the end bit of the CRC status</li> </ul> <p>[Clearing condition] Writing a 0 to this bit Note: When WDATERR is set, the command sequence is stopped automatically.</p>
10	RDATERR	0	R/W*1	<p>Read Data Error [Setting conditions] Error is found in the read data.</p> <ul style="list-style-type: none"> <li>Error is in CRC16 of the read data</li> <li>Error is in the end bit of the read data</li> </ul> <p>[Clearing condition] Writing a 0 to this bit Note: When RDATERR is set, the command sequence is stopped automatically.</p>
9	RIDXERR	0	R/W*1	<p>Response Index Error [Setting condition] Error has been found in the index value of the response.</p> <ul style="list-style-type: none"> <li>When an error has been found in [45:40] of a 6-byte response (including automatically-issued CMD12) or [133:128] of a 17-byte response (The items to be checked are set by RIDXC in CE_CMD_SET.)</li> </ul> <p>[Clearing condition] Writing a 0 to this bit Note: When RIDXERR is set, the command sequence is stopped automatically.</p>
8	RSPERR	0	R/W*1	<p>Response Error [Setting conditions] Error has been found in the response values of the response.</p> <ul style="list-style-type: none"> <li>Transmission bit in the response is high</li> <li>Error is in the end bit of the response</li> <li>When an error has been found in [7:1] of a 6-byte response (including automatically-issued CMD12) or a 17-byte response (The items to be checked are set by RCRC7C in CE_CMD_SET.)</li> </ul> <p>[Clearing condition] Writing a 0 to this bit Note: When RSPERR is set, the command sequence is stopped automatically.</p>
7 to 5	—	All 0	R	<p>Reserved These bits are always read as 0. The write value should always be 0.</p>

Bit	Bit Name	Initial Value	R/W	Description
4	CRCSTO	0	R/W*1	CRC Status Timeout [Setting condition] CRC status could not be received [Clearing condition] Writing a 0 to this bit Note: The command sequence is not stopped even if CRCSTO is set.
3	WDATTO	0	R/W*1	Write Data Timeout [Setting condition] The busy status remains unchanged after the period set by SRWDTO in CE_CLK_CTRL after the CRC status was received. [Clearing condition] Writing a 0 to this bit Note: The command sequence is not stopped even if WDATTO is set.
2	RDATTO	0	R/W*1	Read Data Timeout [Setting conditions] <ul style="list-style-type: none"> <li>Read data could not be received within the period set by SRWDTO in CE_CLK_CTRL after the read command was transmitted</li> <li>Read data could not be received within the period set by SRWDTO in CE_CLK_CTRL after the read data was received.</li> </ul> [Clearing condition] Writing a 0 to this bit Note: The command sequence is not stopped even if RDATTO is set.
1	RBSYTO	0	R/W*1	Response Busy Timeout [Setting condition] The busy status remains unchanged after the period set by SRBSYTO in CE_CLK_CTRL after the command (including automatically-issued CMD12) was transmitted. [Clearing condition] Writing a 0 to this bit Note: The command sequence is not stopped even if RBSYTO is set.
0	RSPTO	0	R/W*1	Response Timeout [Setting condition] Response could not be received within the period set by SRSPTO in CE_CLK_CTRL after the command (including automatically-issued CMD12) was transmitted. [Clearing condition] Writing a 0 to this bit Note: The command sequence is not stopped even if RSPTO is set.

Note 1. A 0 is the only value that can be written to the bit. Writing a 1 is ignored.

### 51.3.12 Interrupt Enable Register (CE\_INT\_EN)

CE\_INT\_EN controls output of the CE\_INT-related interrupt signals. If a flag in CE\_INT is set to 1 while its corresponding bit in CE\_INT\_EN is set to 1, an interrupt request is output. For details on interrupt requests, see section 51.4, Interrupt Requests.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	MCMD12DRE	MCMD12RBE	MCMD12CRE	MDTRANE	MBUFRE	MBUFWEN	MBUFREN	—	—	MRBSYE	MCRSPE
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MCMDVIO	MBUFVIO	—	—	MWDATERR	MRDATERR	MRIDXERR	MRSPERR	—	—	—	MCRCSTO	MWDA TTO	MRDA TTO	MRBSYTO	MRSP TO
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 27	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
26	MCMD12DRE	0	R/W	CMD12DRE Interrupt Enable 0: Disables interrupt output by the CMD12DRE flag 1: Enables interrupt output by the CMD12DRE flag
25	MCMD12RBE	0	R/W	CMD12RBE Interrupt Enable 0: Disables interrupt output by the CMD12RBE flag 1: Enables interrupt output by the CMD12RBE flag
24	MCMD12CRE	0	R/W	CMD12CRE Interrupt Enable 0: Disables interrupt output by the CMD12CRE flag 1: Enables interrupt output by the CMD12CRE flag
23	MDTRANE	0	R/W	DTRANE Interrupt Enable 0: Disables interrupt output by the DTRANE flag 1: Enables interrupt output by the DTRANE flag
22	MBUFRE	0	R/W	BUFRE Interrupt Enable 0: Disables interrupt output by the BUFRE flag 1: Enables interrupt output by the BUFRE flag
21	MBUFWEN	0	R/W	BUFWEN Interrupt Enable 0: Disables interrupt output by the BUFWEN flag 1: Enables interrupt output by the BUFWEN flag
20	MBUFREN	0	R/W	BUFREN Interrupt Enable 0: Disables interrupt output by the BUFREN flag 1: Enables interrupt output by the BUFREN flag
19, 18	—	00	R	Reserved These bits are always read as 0. The write value should always be 0.
17	MRBSYE	0	R/W	RBSYE Interrupt Enable 0: Disables interrupt output by the RBSYE flag 1: Enables interrupt output by the RBSYE flag
16	MCRSPE	0	R/W	CRSPE Interrupt Enable 0: Disables interrupt output by the CRSPE flag 1: Enables interrupt output by the CRSPE flag
15	MCMDVIO	0	R/W	CMDVIO Interrupt Enable 0: Disables interrupt output by the CMDVIO flag 1: Enables interrupt output by the CMDVIO flag
14	MBUFVIO	0	R/W	BUFVIO Interrupt Enable 0: Disables interrupt output by the BUFVIO flag 1: Enables interrupt output by the BUFVIO flag
13, 12	—	00	R	Reserved These bits are always read as 0. The write value should always be 0.
11	MWDATERR	0	R/W	WDATERR Interrupt Enable 0: Disables interrupt output by the WDATERR flag 1: Enables interrupt output by the WDATERR flag

Bit	Bit Name	Initial Value	R/W	Description
10	MRDATERR	0	R/W	RDATERR Interrupt Enable 0: Disables interrupt output by the RDATERR flag 1: Enables interrupt output by the RDATERR flag
9	MRIDXERR	0	R/W	RIDXERR Interrupt Enable 0: Disables interrupt output by the RIDXERR flag 1: Enables interrupt output by the RIDXERR flag
8	MRSPEERR	0	R/W	RSPERR Interrupt Enable 0: Disables interrupt output by the RSPERR flag 1: Enables interrupt output by the RSPERR flag
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	MCRCSTO	0	R/W	CRCSTO Interrupt Enable 0: Disables interrupt output by the CRCSTO flag 1: Enables interrupt output by the CRCSTO flag
3	MWDATTO	0	R/W	WDATTO Interrupt Enable 0: Disables interrupt output by the WDATTO flag 1: Enables interrupt output by the WDATTO flag
2	MRDATTO	0	R/W	RDATTO Interrupt Enable 0: Disables interrupt output by the RDATTO flag 1: Enables interrupt output by the RDATTO flag
1	MRBSYTO	0	R/W	RBSYTO Interrupt Enable 0: Disables interrupt output by the RBSYTO flag 1: Enables interrupt output by the RBSYTO flag
0	MRSPTO	0	R/W	RSPTO Interrupt Enable 0: Disables interrupt output by the RSPTO flag 1: Enables interrupt output by the RSPTO flag

### 51.3.13 Status Register 1 (CE\_HOST\_STS1)

CE\_HOST\_STS1 indicates the number of blocks that have been transferred, states of the MMC\_CMD and MMC\_D lines, index of the received response, and command sequence status.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CMD SEQ	CMD SIG	RSPIDX[5:0]						DATSIG[7:0]							
Initial value:	0	—	0	0	0	0	0	0	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RCVBLK[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	CMDSEQ	0	R	Command Sequence Status 0: Command sequence is in the initial state 1: Command sequence is being executed
30	CMSIG	Undefined	R	MMC_CMD Line Status Indicates the state on the MMC_CMD line.
29 to 24	RSPIDX [5:0]	H'00	R	Response Index Indicate [45:40] of a 6-byte response or [133:128] of a 17-byte response.
23 to 16	DATSIG [7:0]	Undefined	R	MMC_D Status Indicate the states on the MMC_D[7:0] lines. Note: When an error or timeout occurs, MMCDAT[0] may remain 0.
15 to 0	RCVBLK [15:0]	H'0000	R	Number of Transferred Blocks Indicate the number blocks that have been transferred. <ul style="list-style-type: none"> <li>When the DWEN bit in CE_CMD_SET is 0 Number of blocks read from the card</li> <li>When the DWEN bit in CE_CMD_SET is 1 Number of blocks written to the card</li> </ul>

### 51.3.14 Status Register 2 (CE\_HOST\_STS2)

CE\_HOST\_STS2 indicates timeout and error status.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRC STE	CRC 16E	AC12 CRCE	RSP CRC7E	CRC STEBE	RDATEBE	AC12REBE	RSP EBE	AC12IDXE	RSP IDXE	—	—	—	CRCST[2:0]		
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	STRD ATTO	DATBS YTO	CRCST TO	AC12BSYTO	RSPBS YTO	AC12RSPTO	STRS PTO	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	CRCSTE	0	R	CRC Status Error This bit is set to 1 when an error is found in the CRC status value.
30	CRC16E	0	R	Read Data CRC16 Error This bit is set to 1 when an error is found in CRC16 in the read data.
29	AC12CRCE	0	R	Automatic CMD12 Response CRC7 Error This bit is set to 1 when an error is found in [7:1] of the response to the automatically-issued CMD12. Note: The items to be checked are set by RCRC7C in CE_CMD_SET.
28	RSPCRC7E	0	R	Command Response CRC7 Error (other than automatically-issued CMD12) This bit is set to 1 when an error is found in [7:1] of a 6-byte response or a 17-byte response. Note: The items to be checked are set by RCRC7C in CE_CMD_SET.
27	CRCSTEBE	0	R	CRC Status End Bit Error This bit is set to 1 when an error is found in the end bit in CRC status.
26	RDATEBE	0	R	Read Data End Bit Error This bit is set to 1 when an error is found in the end bit in the read data.
25	AC12REBE	0	R	Automatic CMD12 Response End Bit Error This bit is set to 1 when an error is found in the end bit of the response to the automatically-issued CMD12.
24	RSPEBE	0	R	Command Response End Bit Error (other than automatically-issued CMD12) This bit is set to 1 when an error is found in the end bit of the response.
23	AC12IDXE	0	R	Automatic CMD12 Response Index Error This bit is set to 1 when an error is found in [45:40] of the response to the automatically-issued CMD12. Note: The items to be checked are set by RIDXC in CE_CMD_SET.
22	RSPIDXE	0	R	Command Response Index Error (other than automatically-issued CMD12) This bit is set to 1 when an error is found in [45:40] of a 6-byte response or [133:128] of a 17-byte response. Note: The items to be checked are set by RIDXC in CE_CMD_SET.
21 to 19	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
18 to 16	CRCST [2:0]	000	R	CRC Status Indicate the value of the CRC status that has been received.
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	STRDATTO	0	R	Read Data Timeout <ul style="list-style-type: none"> <li>This bit is set to 1 if read data is not received within the period set by the SRWDTO bits in CE_CLK_CTRL after a read command was transmitted.</li> <li>This bit is set to 1 if read data is not received within the period set by the SRWDTO bits in CE_CLK_CTRL after a read data was received.</li> </ul>

Bit	Bit Name	Initial Value	R/W	Description
13	DATBSYTO	0	R	Data Busy Timeout This bit is set to 1 if busy status remains unchanged after the period set by the SRWDTO bits in CE_CLK_CTRL after the CRC status was received.
12	CRCSTTO	0	R	CRC Status Timeout This bit is set to 1 if CRC status could not be received.
11	AC12BSYTO	0	R	Automatic CMD12 Response Busy Timeout This bit is set to 1 if busy state remains unchanged after the period set by the SRBSYTO bits in CE_CLK_CTRL after the automatically-issued CMD12 was transmitted.
10	RSPBSYTO	0	R	Response Busy Timeout This bit is set to 1 if busy state remains unchanged after the period set by the SRBSYTO bits in CE_CLK_CTRL after a command (other than automatically-issued CMD12) was transmitted.
9	AC12RSPTO	0	R	Automatic CMD12 Response Timeout This bit is set to 1 if the response is not received within the period set by the SRSPTO bits in CE_CLK_CTRL after the automatically-issued CMD12 was transmitted.
8	STRSPTO	0	R	Response Timeout This bit is set to 1 if the response is not received within the period set by the SRSPTO bits in CE_CLK_CTRL after a command (other than automatically-issued CMD12) was transmitted.
7 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

### 51.3.15 DMA Mode Setting Register (CE\_DMA\_MODE)

CE\_DMA\_MODE sets the transfer unit for DMA transfer.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DMA SEL
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 1	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
0	DMASEL	0	R/W	DMA Transfer Size Select Selects the transfer unit for CE_DATA read/write DMA transfer. Set this bit in combination with the transfer size set in the DMA channel configuration register. 0: 2-byte (word) or 4-byte (longword) unit 1: 64-byte (longword × 16) unit Note: When a transfer error or timeout occurs during DMA transfer in 64-byte units, leading to the forced termination of transfer, write 0 to this bit and then set it to 1 again. Also do this whenever a software reset is used.



### 51.3.16 Card Detection/Port Control Register (CE\_DETECT)

CE\_DETECT controls card detection. For details on interrupt requests by card detection, see section 51.4, Interrupt Requests.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	CD SIG	CD RISE	CD FALL	—	—	—	—	—	—	MCD RISE	MCD FALL	—	—	—	—
Initial value:	0	—	0	0	0	—	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R/W*	R/W*	R	R	R	R	R	R	R/W	R/W	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 15	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
14	CDSIG	Undefined	R	MMC_CD Pin Status Indication Indicates the MMC_CD pin status.
13	CDRISE	0	R/W*1	MMC_CD Pin Rise Detection Flag [Setting condition] The MMC_CD pin level changes from low to high. [Clearing condition] Writing a 0 to this bit
12	CDFALL	0	R/W*1	MMC_CD Pin Fall Detection Flag [Setting condition] The MMC_CD pin level changes from high to low. [Clearing condition] Writing a 0 to this bit
11	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
10	—	Undefined	R	Reserved The write value should always be 0.
9 to 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	MCDRISE	0	R/W	CDRISE Interrupt Enable 0: Disables interrupt output by the CDRISE flag. 1: Enables interrupt output by the CDRISE flag.
4	MCDFALL	0	R/W	CDFALL Interrupt Enable 0: Disables interrupt output by the CDFALL flag. 1: Enables interrupt output by the CDFALL flag.
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

Note 1. A 0 is the only value that can be written to the bit. Writing a 1 is ignored.

### 51.3.17 Special Mode Setting Register (CE\_ADD\_MODE)

CE\_ADD\_MODE controls the internal clock.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	CLK MAIN	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 20	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
19	CLKMAIN	0	R/W	Internal Clock Control 0: Normal mode 1: Low power consumption mode (only card detection is enabled)
18 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

### 51.3.18 Version Register (CE\_VERSION)

CE\_VERSION indicates the version number and controls software reset of this module.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SW RST	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VERSION[15:0]															
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31	SWRST	0	R/W	Software Reset 0: Software reset cleared (normal operation) 1: Executes software reset. When SWRST is set to 1, all the register values are reset to the initial values. (SWRST is not reset to the initial value)
30 to 16	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	VERSION [15:0]	H'0003	R	Version Information Indicates the version number of this module.

## 51.4 Interrupt Requests

Table 51.3 shows the interrupt request specifications of this module.

This module generates three types of interrupt requests: normal operation, error/timeout, and card detection. When an interrupt flag is set to 1 and also the corresponding interrupt is enabled, an interrupt request is asserted.

**Table 51.3 Specifications of Interrupt Requests**

Flag Register	Bit	Mask Register	Bit	Interrupt Request
CE_INT	CMD12DRE	CE_INT_EN	MCMD12DRE	Normal operation interrupt (MMC2)
	CMD12RBE		MCMD12RBE	
	CMD12CRE		MCMD12CRE	
	DTRANE		MDTRANE	
	BUFRE		MBUFRE	
	BUFWEN		MBUFWEN	
	BUFREN		MBUFREN	
	RBSYE		MRBSYE	
	CRSPE		MCRSPE	
	CMDVIO		MCMDVIO	
	BUFVIO		MBUFVIO	
	WDATERR		MWDATERR	
	RDATERR		MRDATERR	
	RIDXERR		MRIDXERR	
	RSPERR		MRSPELL	
	CRCSTO		MCRCSO	
	WDATTO		MWDATTO	
	RDATTO		MRDATTO	
	RBSYTO		MRBSYTO	
	RSPTO		MRSPTO	
CE_DETECT	CDRISE	CE_DETECT	MCDRISE	Card detection interrupt (MMC0)
	CDFALL		MCDFALL	

## 51.5 DMA Specifications

This module provides two channels for DMA transfer requests: one for buffer reading and the other for buffer writing.

### 51.5.1 DMA for Buffer Writing

The DMA transfer request is asserted for buffer writing when the buffer has become empty while the DMAWEN bit in CE\_BUF\_ACC is 1.

The DMA transfer request stays asserted for the amount of data specified by BLKSIZ (the block size set in CE\_BLOCK\_SET)  $\times$  BLKCNT (the number of blocks for transfer set in CE\_BLOCK\_SET), and negated after the last block has been transferred. Note that the BUFWEN bit in CE\_INT will not be asserted in this case.

If an error occurs during DMA transfer or DMA transfer is forcibly terminated, the command sequence is stopped automatically, which causes the DMA transfer request to be negated.

### 51.5.2 DMA for Buffer Reading

The DMA transfer request is asserted for buffer reading when the buffer stores data of the block size specified in CE\_BLOCK\_SET while the DMAREN bit in CE\_BUF\_ACC is 1.

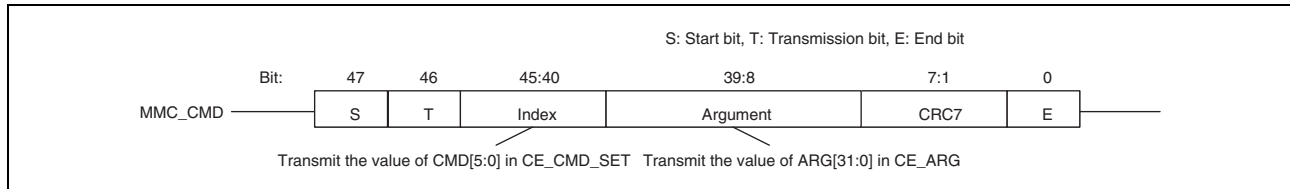
The DMA transfer request stays asserted for the amount of data specified by BLKSIZ (the block size set in CE\_BLOCK\_SET)  $\times$  BLKCNT (the number of blocks for transfer set in CE\_BLOCK\_SET), and negated after the last block has been transferred. Note that the BUFREN bit in CE\_INT will not be asserted in this case.

If an error occurs during DMA transfer or DMA transfer is forcibly terminated, the command sequence is stopped automatically, which causes the DMA transfer request to be negated.

## 51.6 Operation

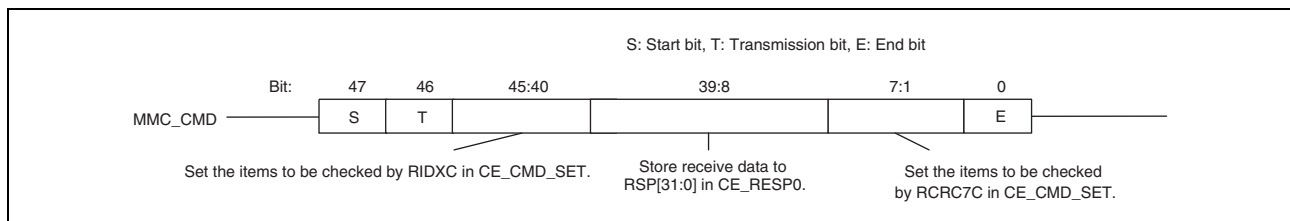
### 51.6.1 Command/Response Formats

Figure 51.2 shows the format of the command to be transferred. The command index that is set in CMD[5:0] of CE\_CMD\_SET and the argument set in ARG[31:0] of CE\_ARG are reflected in the command.

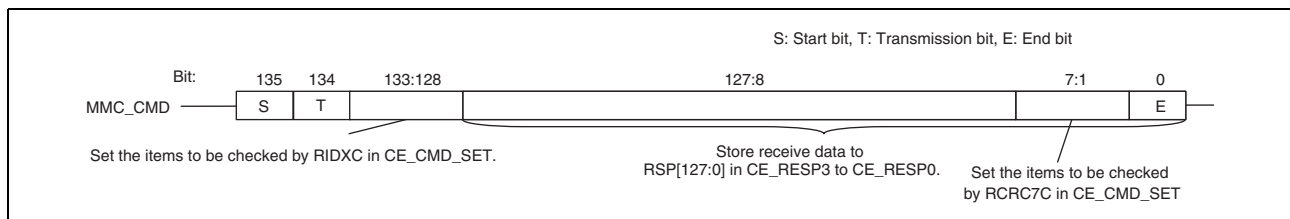


**Figure 51.2 Command Format**

Figure 51.3 and Figure 51.4 show the formats when a 6-byte response and 17-byte response (R2) are received, respectively. The response index is stored in RSPIDX[5:0] of CE\_HOST\_STS1, and the status value of the response is stored to CE\_RESP0 or CE\_RESP3 to CE\_RESP0.



**Figure 51.3 Format of 6-Byte Response**



**Figure 51.4 Format of 17-Byte Response (R2)**

### 51.6.2 Data Block Format

Figure 51.5 shows the format of data blocks. For details on D0 to D3 in Figure 51.5, see section 51.6.3, Buffer Structure and Buffer Accesses. When data is written to a card, data stored in the buffer is transmitted. When data is read from a card, receive data is stored to the buffer.

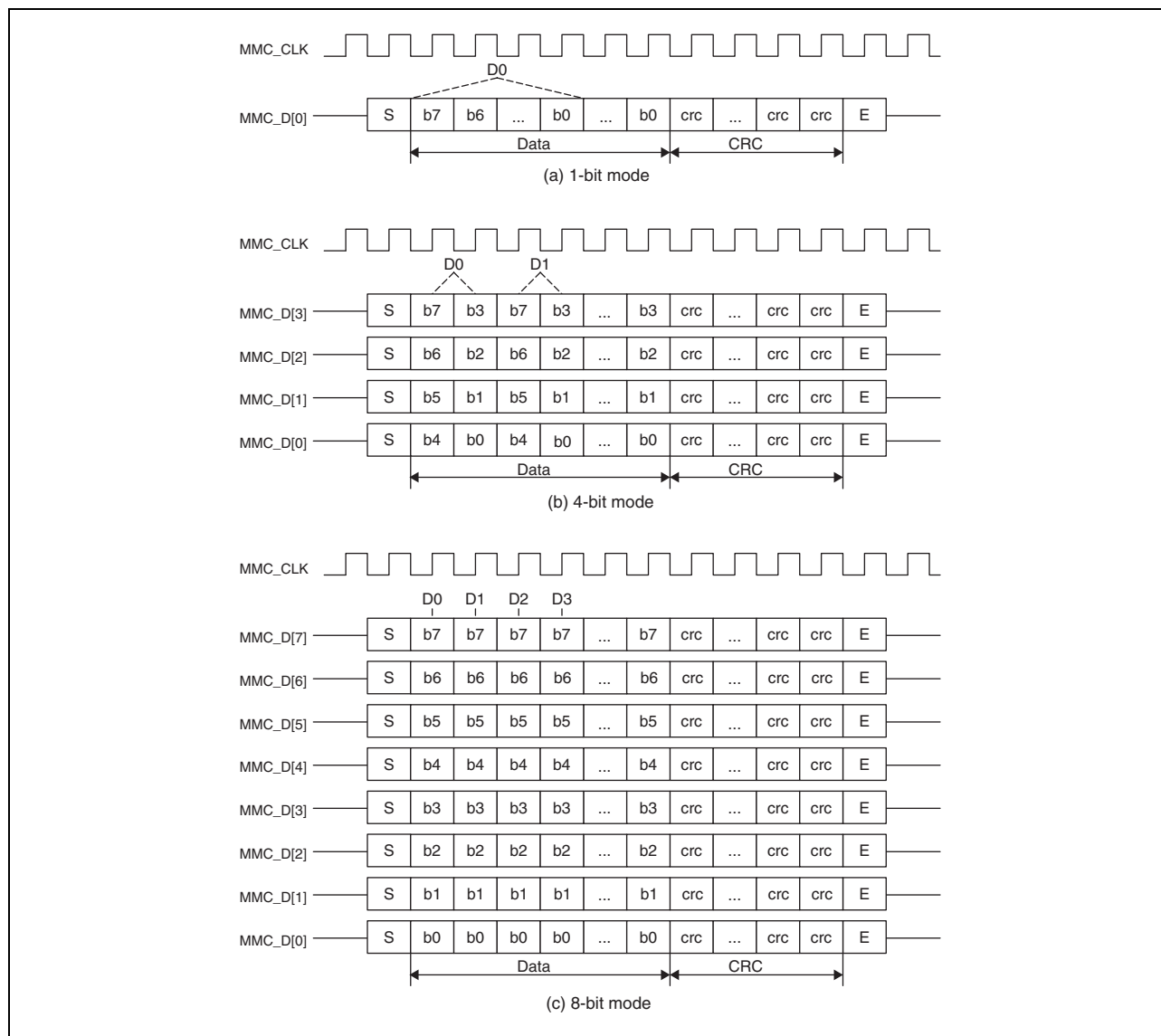


Figure 51.5 Data Block Format

### 51.6.3 Buffer Structure and Buffer Accesses

This module has two units of 512-byte RAM as shown in Figure 51.6. Therefore, for multi-block writing, if one block of data (512 bytes) stored in one buffer has been transmitted but the other buffer is full, the next block of data can be continuously transmitted. For multi-block reading, if one block of receive data (512 bytes) has been stored in one buffer but the other buffer is empty, the next block of receive data can be continuously stored in the buffer.

If neither of the buffers is empty for multi-block reading, the MMC clock is stopped and reception is suspended. When one of the buffers becomes empty, the MMC clock supply is started to start reception.

The buffer is accessed with CE\_DATA. CE\_DATA should be accessed for  $4 \times (n + 1)$  bytes ( $n = 0, 1, 2, 3, \dots, 127$ ).

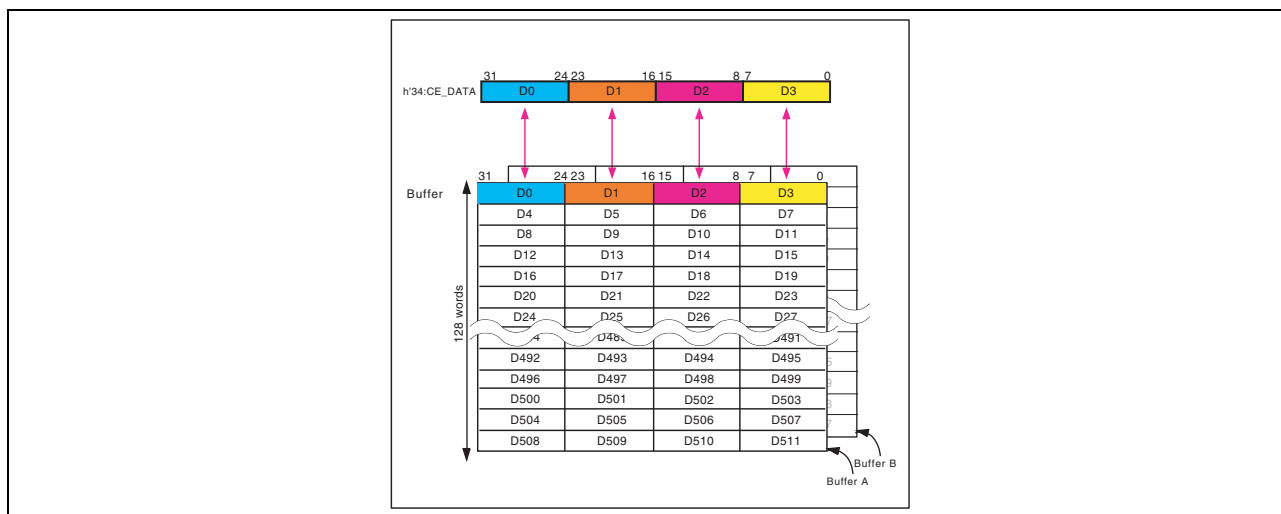


Figure 51.6 Double Buffer Structure

The buffer access select function allows byte-wise swapping of data when the buffer is accessed by writing to or reading from CE\_DATA. This function is enabled by the setting of CE\_BUFF\_ACC. Figure 51.7 shows the specification of byte-wise swapping.

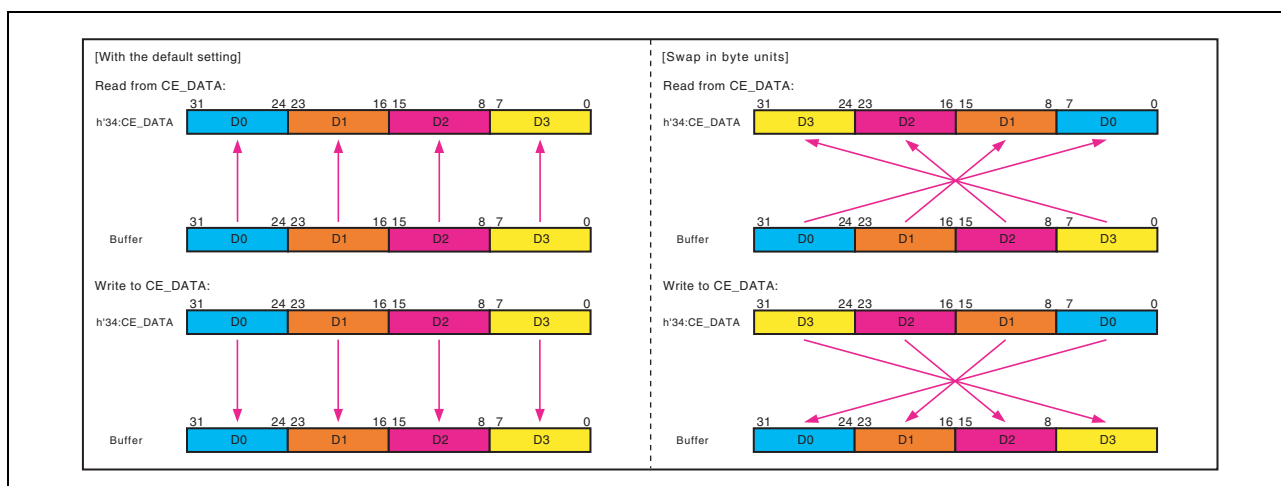
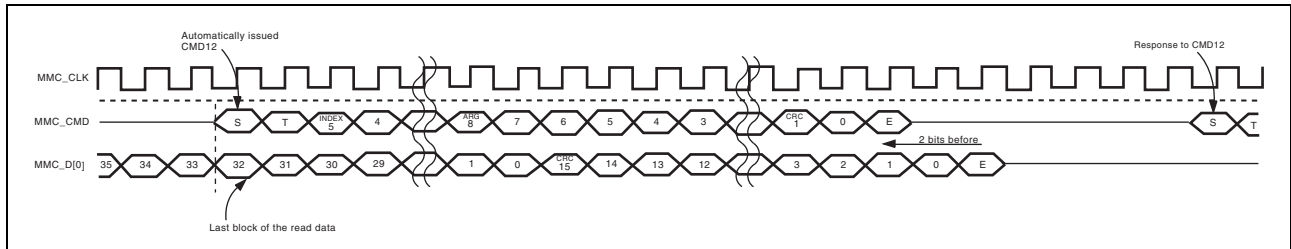


Figure 51.7 Specification of Byte-Wise Swapping

### 51.6.4 Automatic CMD12 Issuance

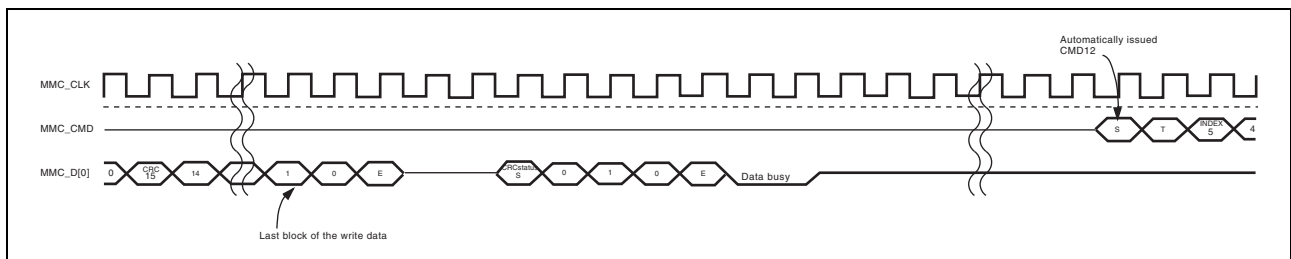
This module automatically issues CMD12 when multi-block transfer is performed with the CMD12EN bit in CE\_CMD\_SET set to 1.

Figure 51.8 shows the timing of automatic CMD12 issuance in multi-block read. CMD12 is issued such that the end bit of the command is sent two bits before the end bit of the data during reception of the last block.



**Figure 51.8** Timing of Automatically-Issued CMD12 in Multi-Block Read (1-Bit Mode)

Figure 51.9 shows the timing of automatic CMD12 issuance in multi-block write. CMD12 is issued after the data busy after transmission of the last block has ended.



**Figure 51.9** Timing of Automatically-Issued CMD12 in Multi-Block Write (1-Bit Mode)

The argument of the automatically-issued CMD12 is set by CE\_ARG\_CMD12. The value of [39:8] of the response to CMD12 is stored to CE\_RESP\_CMD12. The busy status on response reception is received.



### 51.6.5 High Priority Interrupt (HPI)

HPI should be executed as described below with reference to section 51.3, Register Descriptions and section 51.7, Examples of Setting.

#### (1) When HPI is executed during a write to the device

- (a) Terminate the command sequence forcibly.
- (b) Wait for the CMDSEQ bit in CE\_HOST\_STS1 to be cleared to 0.
- (c) Issue CMD12 (R1) to make the device in the rcv state enter the prg state. Note that the device does not output the response when the device is in the prg state before issuance of CMD12(R1).
- (d) Issue CMD13 (R1).
- (e) Issue the HPI command.

#### (2) When HPI is executed during response busy while not writing to the device

- (a) Terminate the command sequence forcibly.
- (b) Wait for the CMDSEQ bit in CE\_HOST\_STS1 to be cleared to 0.
- (c) Issue CMD13 (R1).
- (d) Issue the HPI command\*.

The procedure above is for HPI generated in the CMD6, CMD24, CMD25 (Pre-defined), or CMD38 sequence.

Note: \* CMD12 (R1b) or CMD13 (R1b) depending on the device connected

### 51.6.6 Background Operation

Background operation should be executed as described below with reference to section 51.3, Register Descriptions and section 51.7, Examples of Setting.

To execute background operation, issue CMD6 (R1) to write to the BKOPS\_START byte in the EXT\_CSD register of the device.

Completion of background operation can be determined by checking the device state with CMD13 (R1) issued after CMD6 (R1) or by polling MMCDAT[0]. When the device state is indicated as “tran” or MMCDAT[0] is high, background operation is determined to be completed.

To suspend background operation, use HPI described in section 51.6.5.

### 51.6.7 Operation in the Case of Error/Timeout

This module may not be stopped in case of error occurrence. If the command sequence is in progress when an error occurs (check the CMDSEQ bit in CE\_HOST\_STS1), terminate the sequence forcibly and perform a software reset. The data for transmission or received data that had been stored in the buffers at the time of error occurrence are not guaranteed.

This module is not stopped when a timeout has occurred. To execute the next command after a timeout error has occurred, terminate the sequence forcibly, perform a software reset.

For forcible termination, refer to section 51.7.11, Forcible Termination.

### 51.7 Examples of Setting

This section shows the procedures for executing typical command sequences.

#### 51.7.1 Legends

Figure 51.10 shows the legends for the symbols used in the figures in the following subsections.

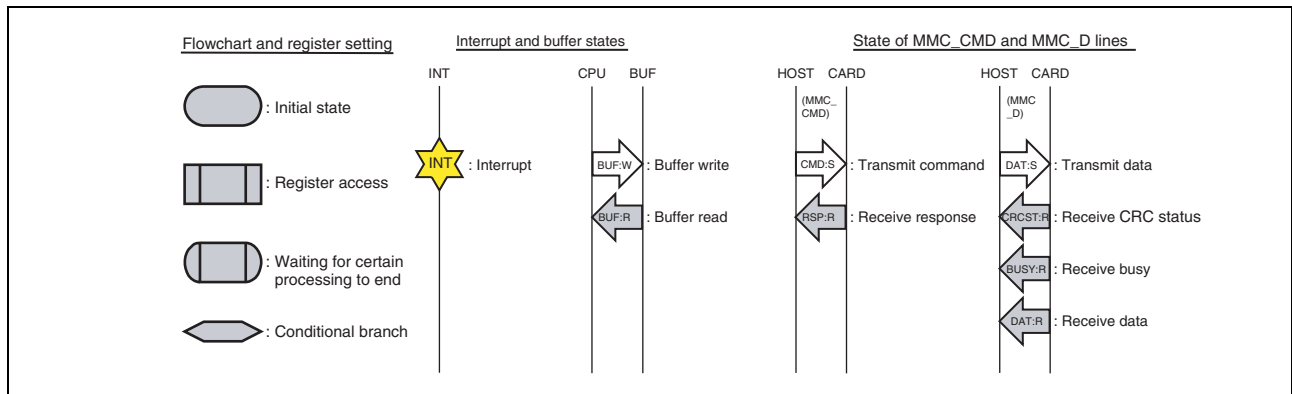


Figure 51.10 Legends for the Symbols Used in the Figures

#### 51.7.2 Command Transmission

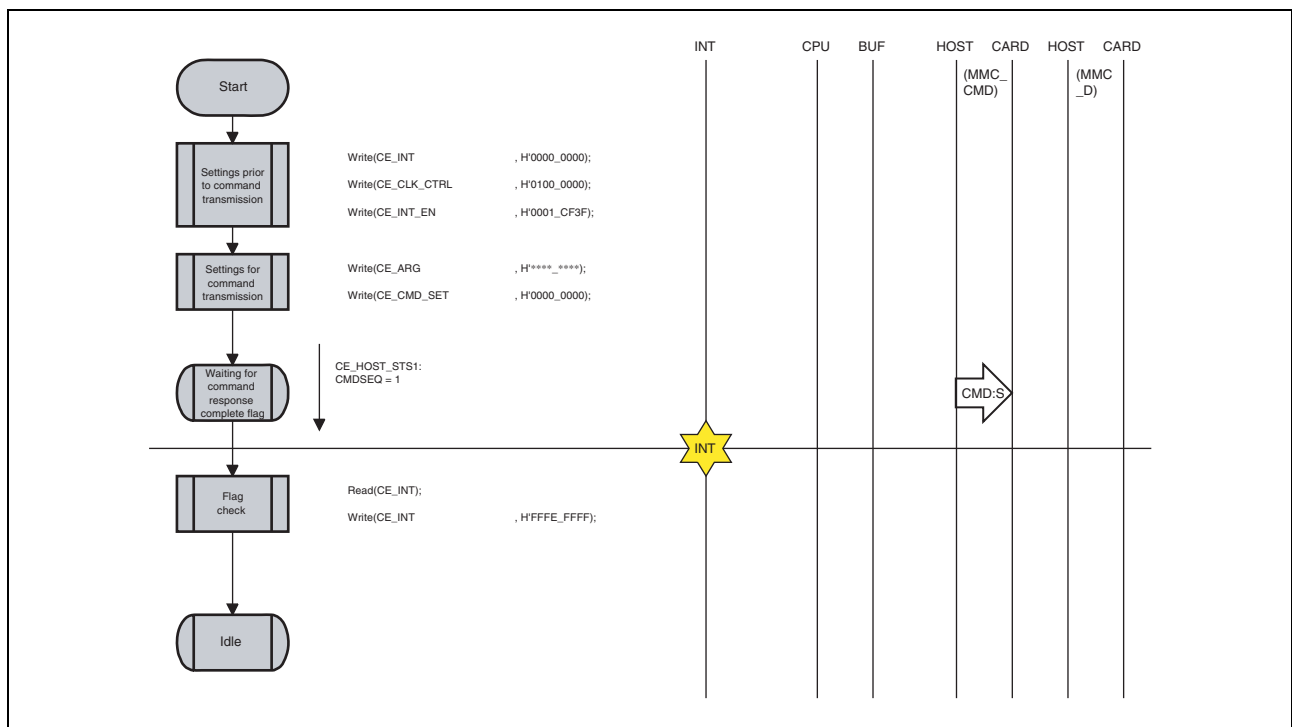


Figure 51.11 Command Transmission (CMD0)

51.7.3 Command Transmission → Response Reception

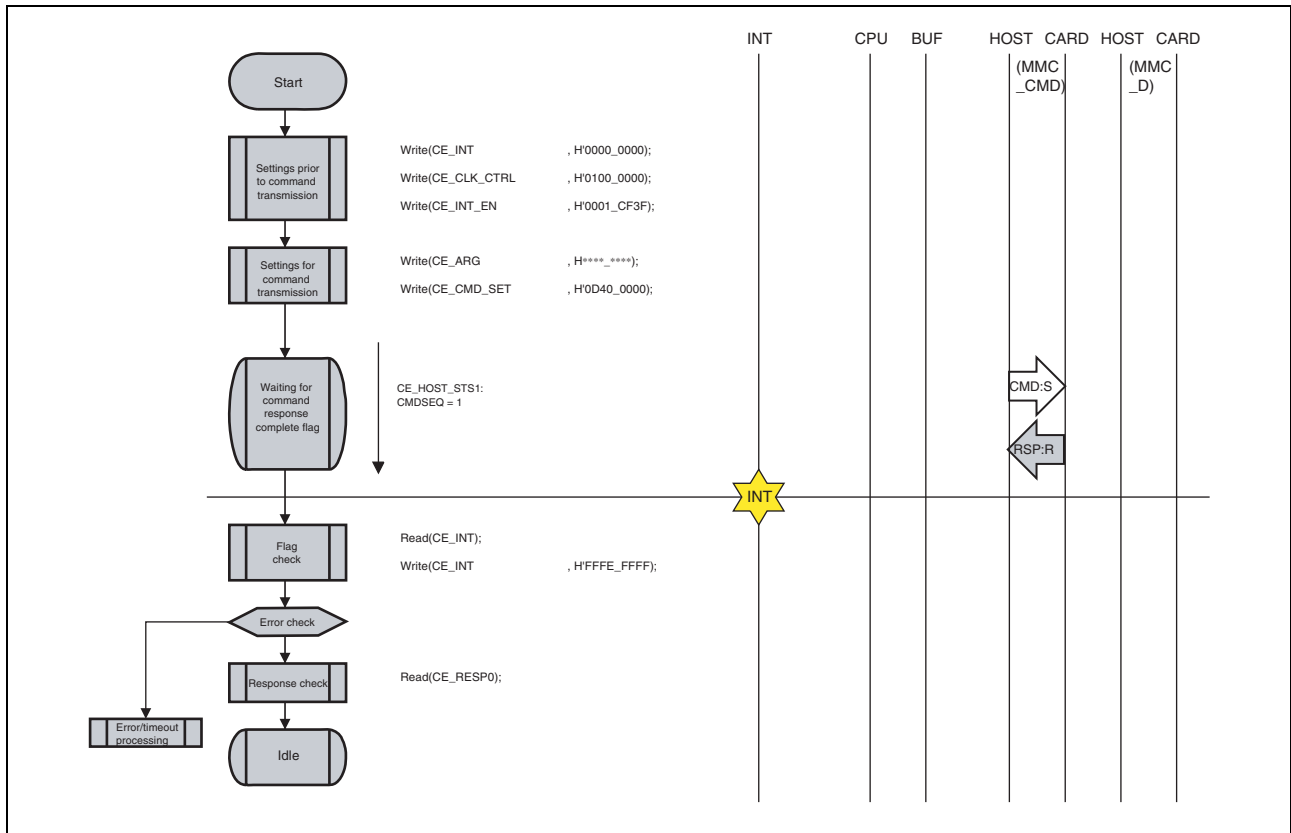


Figure 51.12 Command Transmission → Response Reception (CMD13)

51.7.4 Command Transmission → Response Reception (with Response Busy)

(1) When the busy time period is less than the period set by SRBSYTO in CE\_CLK\_CTRL

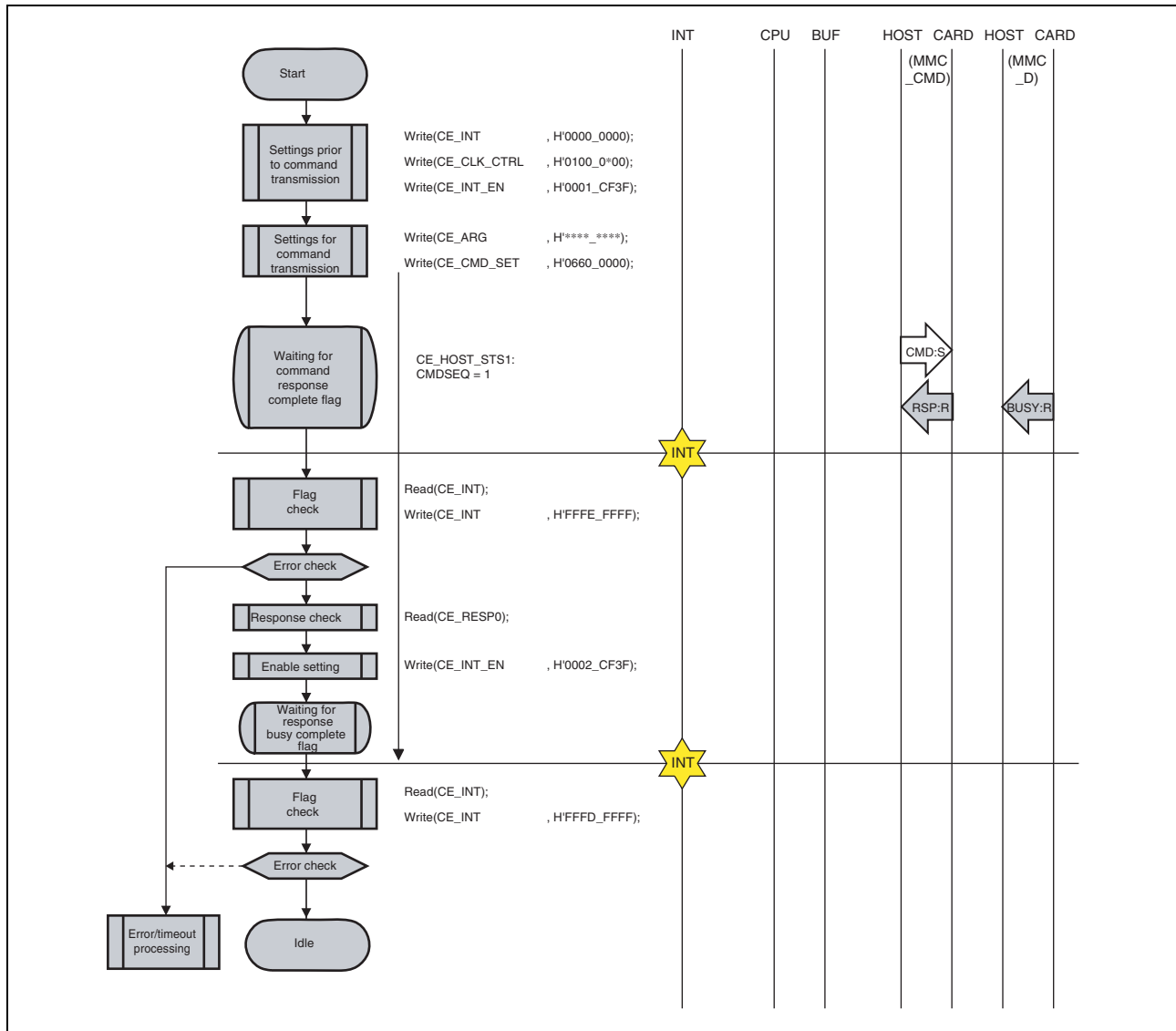


Figure 51.13 Command Transmission → Response Reception (with Response Busy) (CMD6)

(2) When the busy time period may be equal to or beyond the period set by SRBSYTO in CE\_CLK\_CTRL

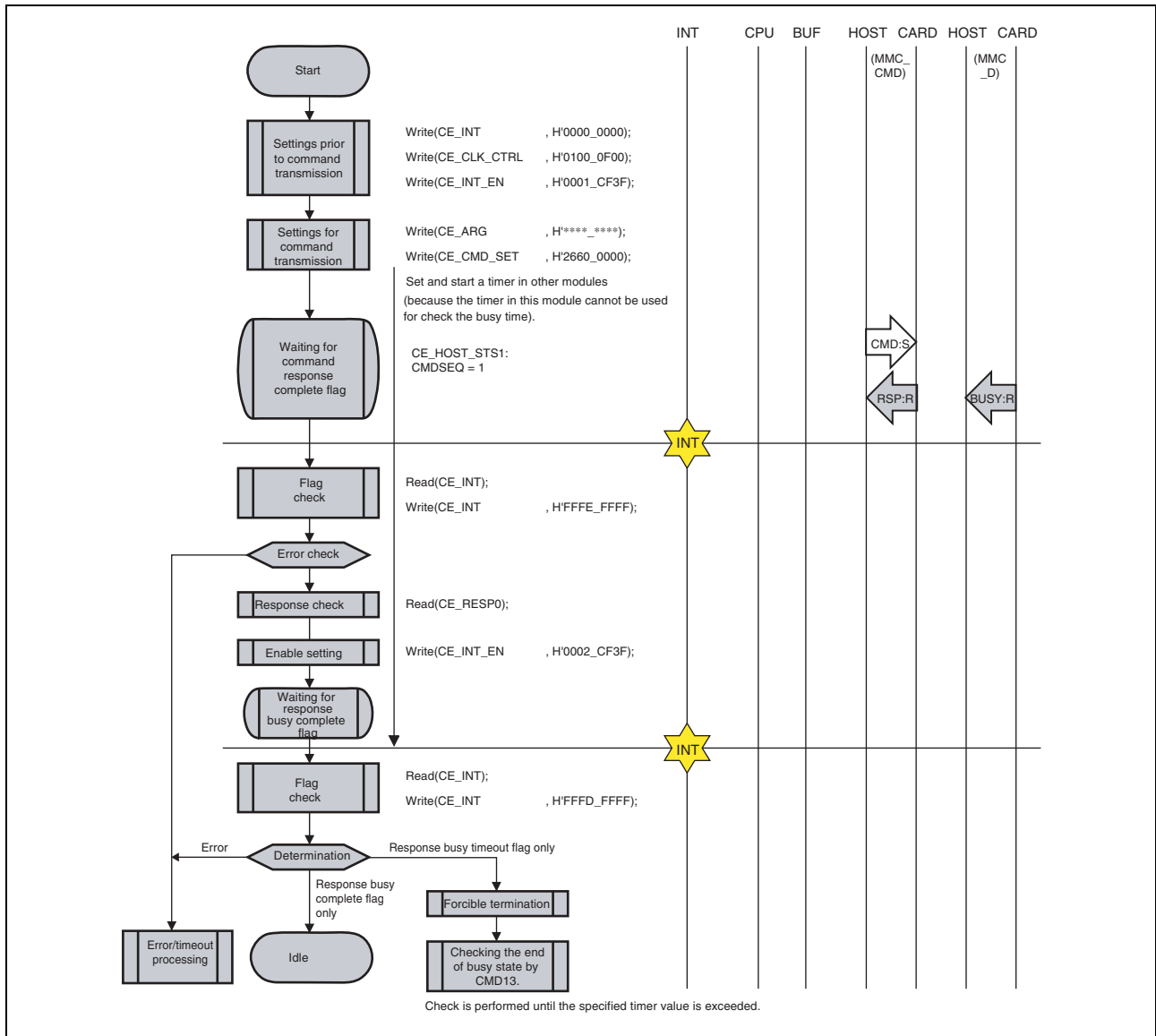


Figure 51.14 Command Transmission → Response Reception (with Response Busy) (CMD38)

51.7.5 Single-Block Read

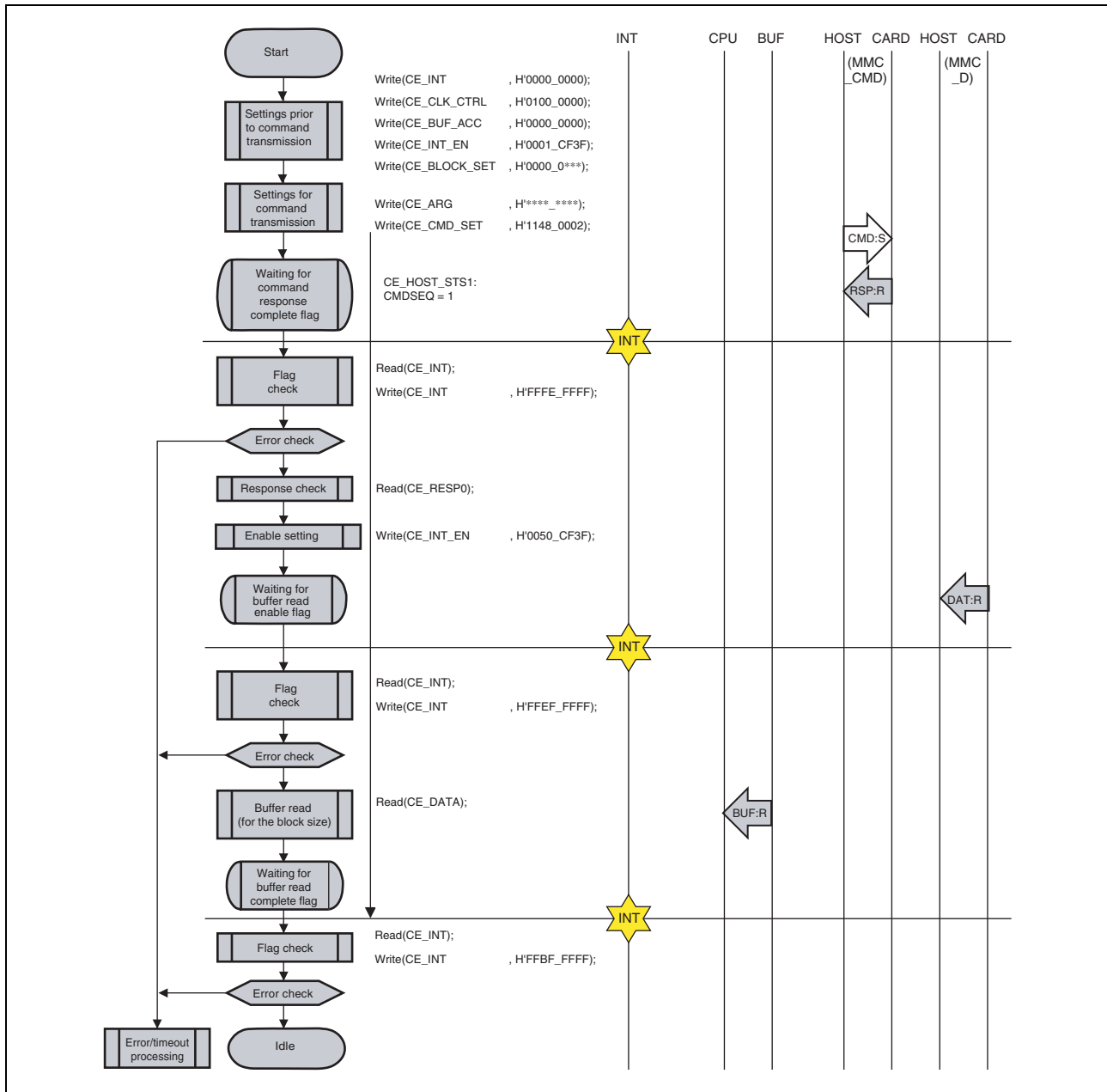


Figure 51.15 Single-Block Read (CMD17)

51.7.6 Multi-Block Read

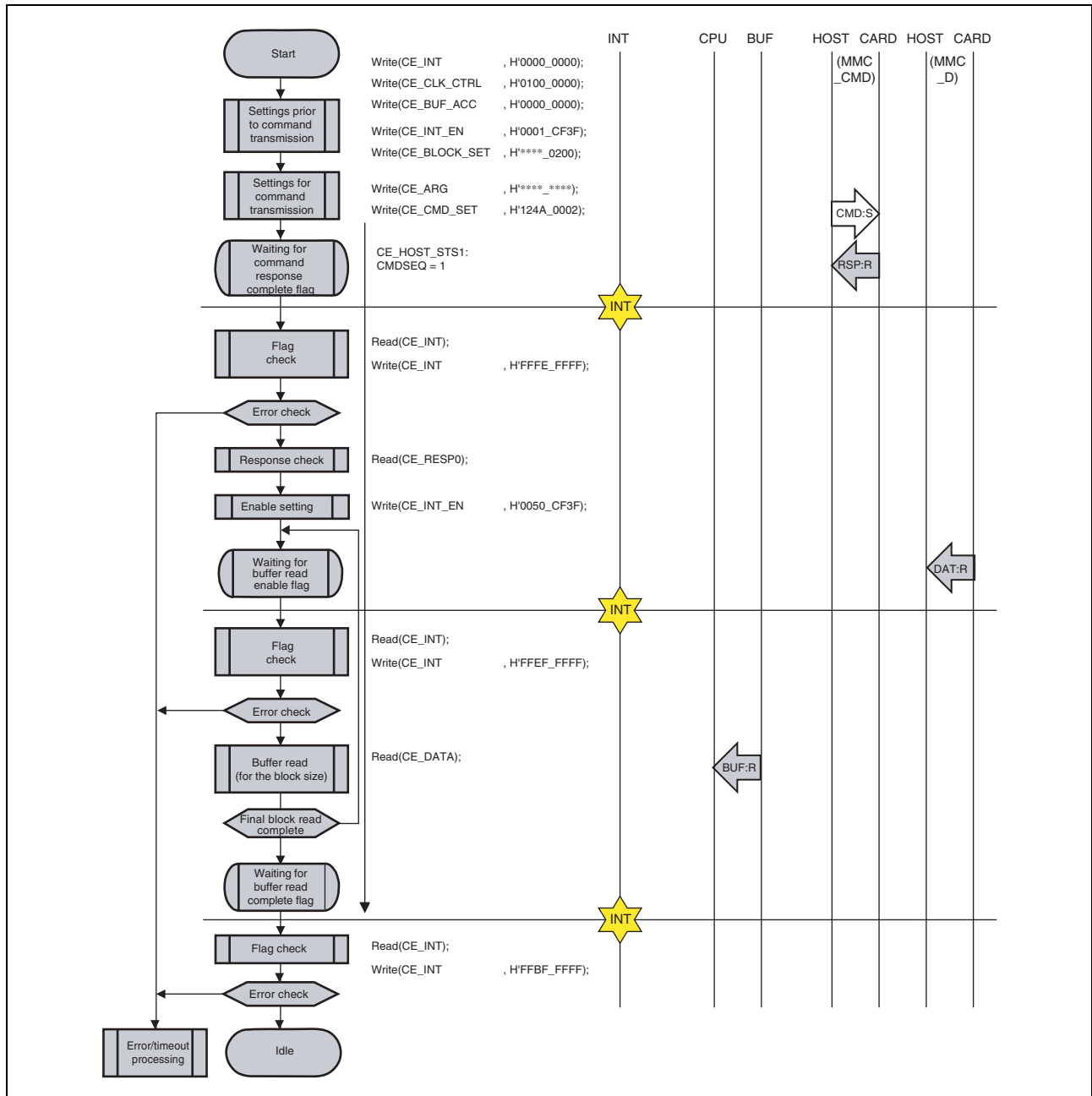


Figure 51.16 Multi-Block Read (CMD18 Pre-Defined)

51.7.7 Multi-Block Read (with Automatic CMD12 Issuance)

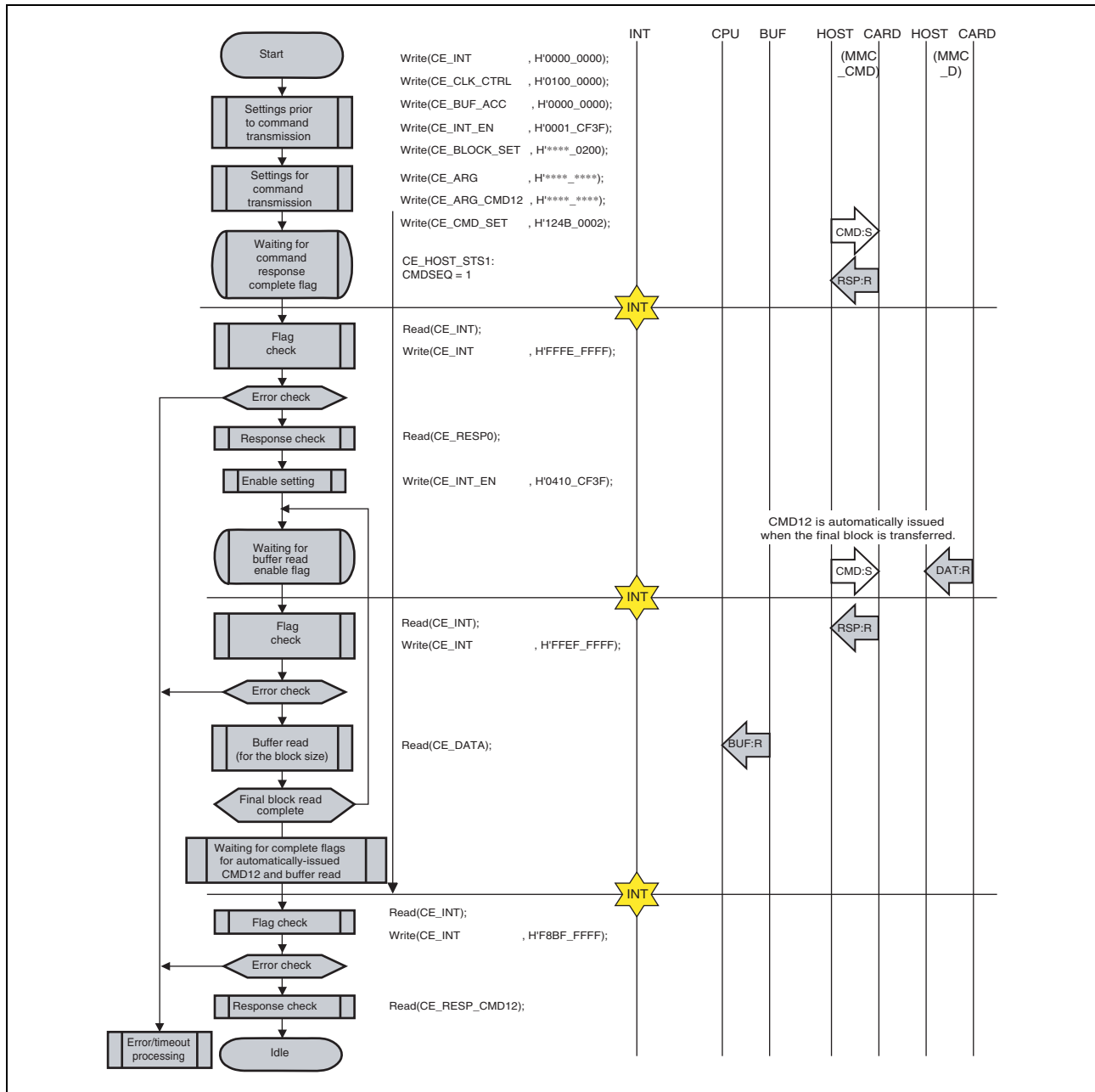


Figure 51.17 Multi-Block Read (with Automatic CMD12 Issuance) (CMD18 Open-Ended)



51.7.8 Single-Block Write

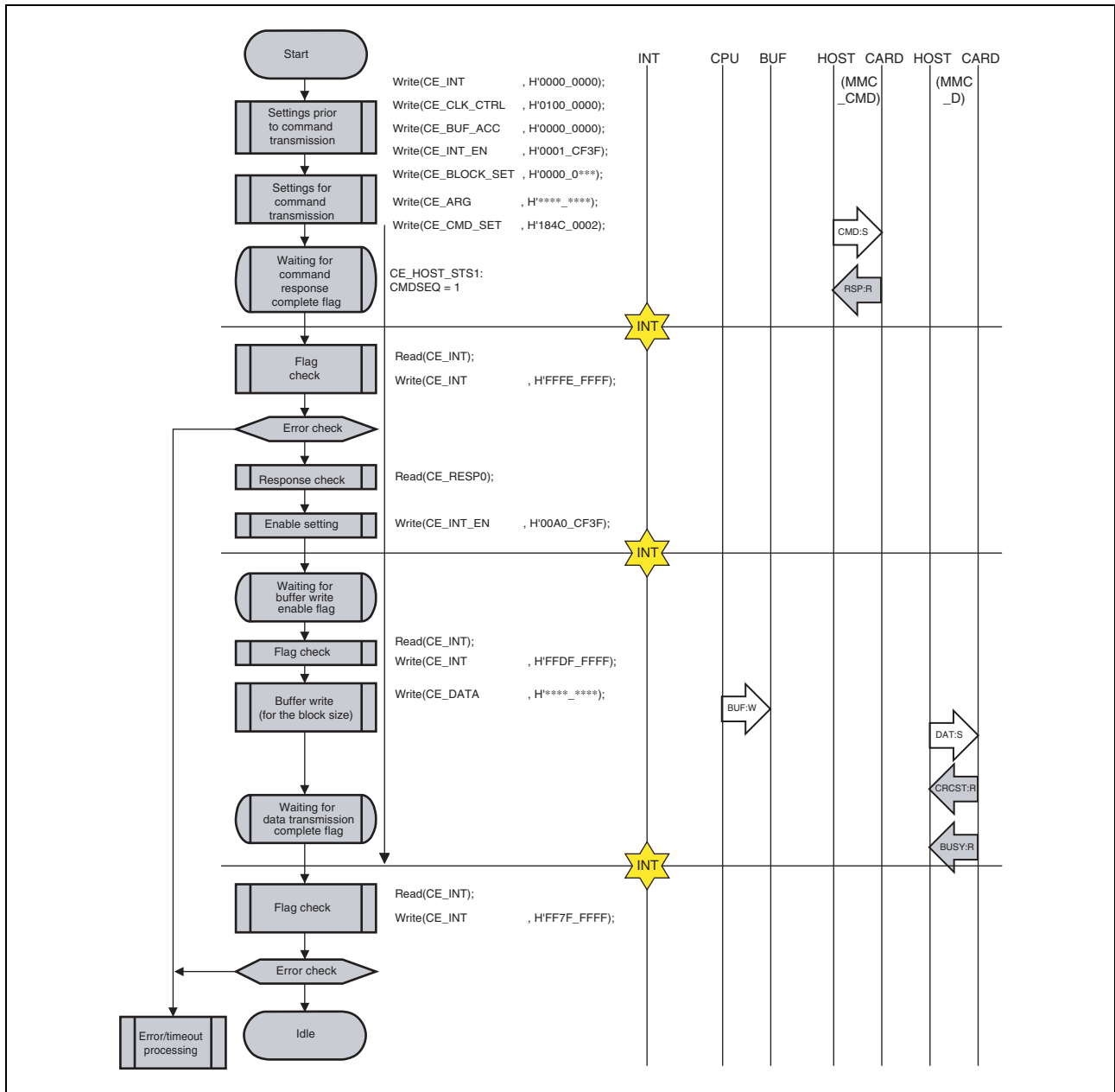


Figure 51.18 Single-Block Write (CMD24)

51.7.9 Multi-Block Write

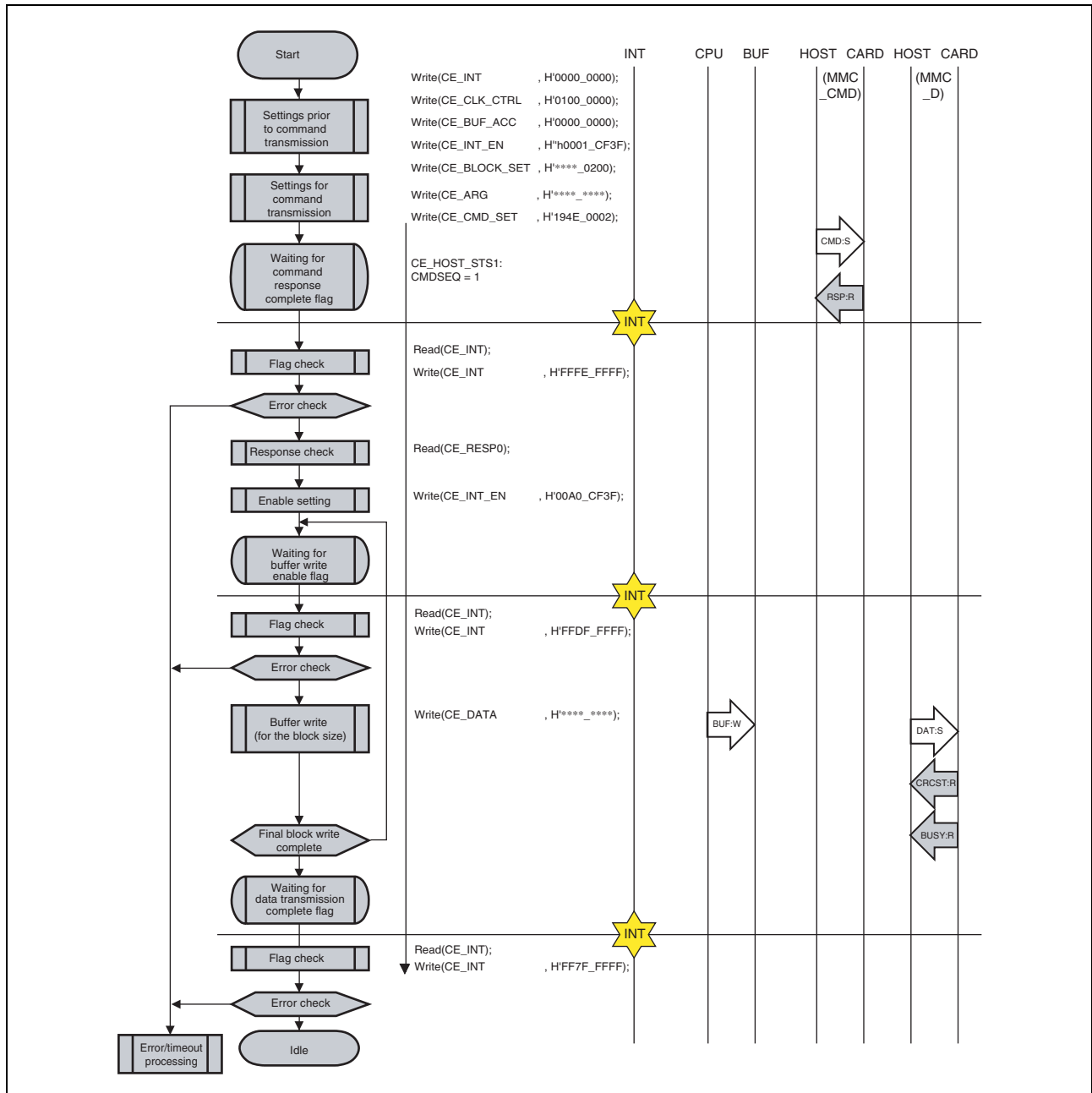


Figure 51.19 Multi-Block Write (CMD25 Pre-Defined)

51.7.10 Multi-Block Write (with Automatic CMD12 Issuance)

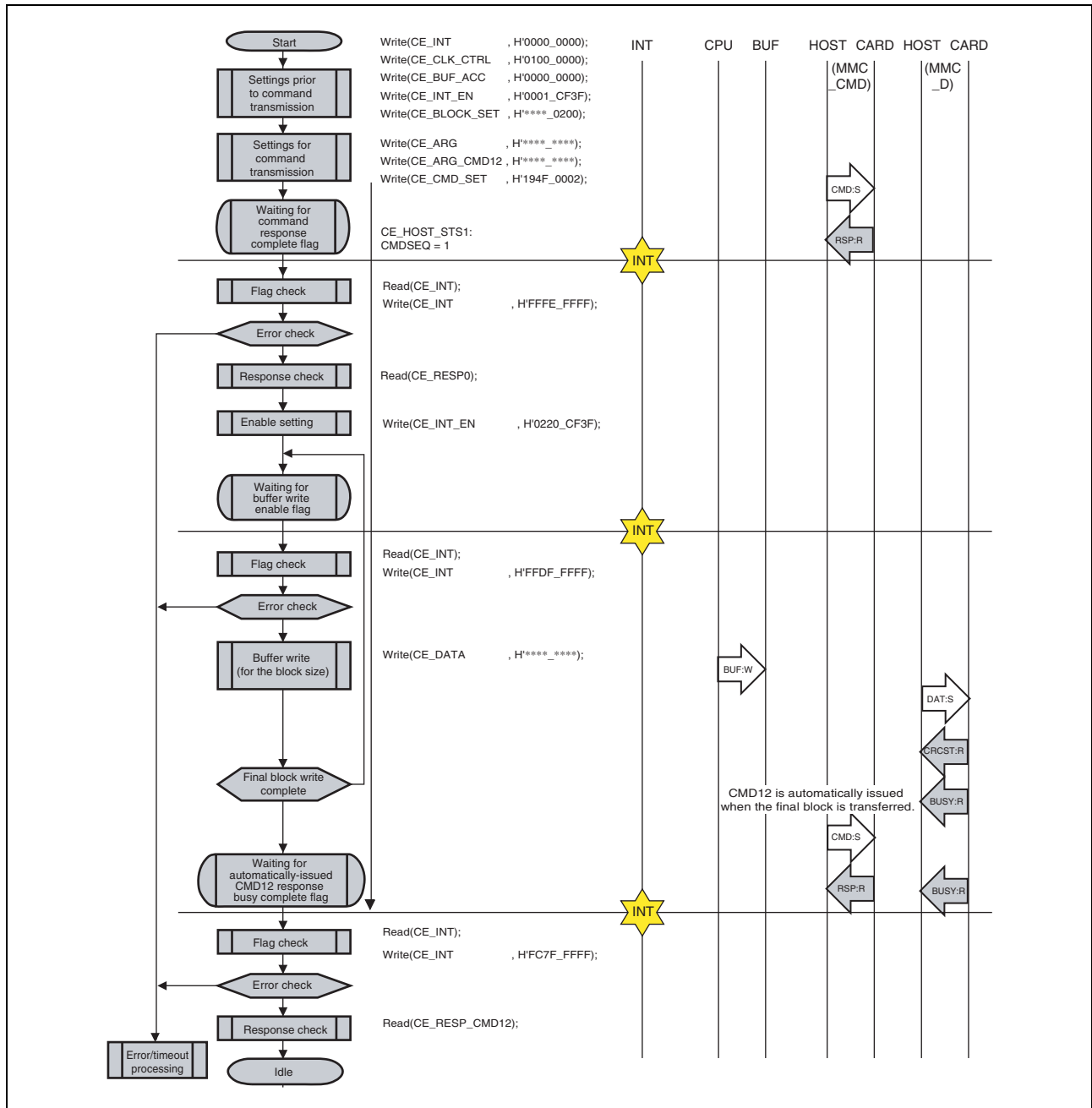


Figure 51.20 Multi-Block Write (with Automatic CMD12 Issuance) (CMD25 Open-Ended)

51.7.11 Forcible Termination

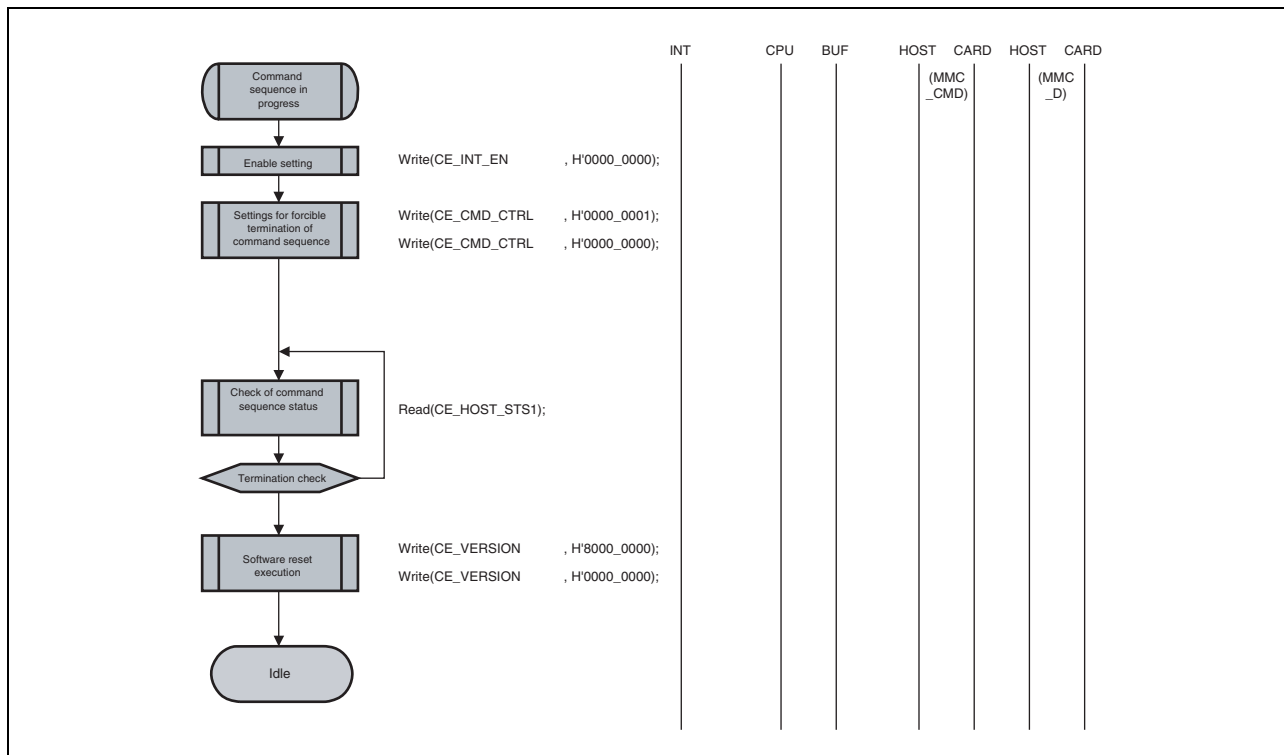


Figure 51.21 Forcible Termination

## 51.7.12 Setting Values of CE\_CMD\_SET

Table 51.4 lists the setting values required to issue commands.

Table 51.4 Setting Values of CE\_CMD\_SET

Command	Response	CE_CMD_SET																		Remarks					
		.	.	CMD[5:0]	RTYP[1:0]	RBSY	.	WDAT	DWEN	CMLTE	CMD12EN	RIDXC[1:0]	RCRC7C[1:0]	.	CRC16C	.	CRCSTE	TBIT	OPDM		.	.	SBIT	.	DATW[1:0]
CMD0	-	0	0	000000	00	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	0	00	
CMD1	R3	0	0	000001	01	0	0	0	0	0	0	01	01	0	0	0	0	0	0	0	0	0	0	00	
CMD2	R2	0	0	000010	10	0	0	0	0	0	0	01	10	0	0	0	0	0	0	0	0	0	0	00	
CMD3	R1	0	0	000011	01	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	0	00	
CMD4	-	0	0	000100	00	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	0	00	
CMD5	R1b	0	0	000101	01	1	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	0	00	
CMD6	R1	0	0	000110	01	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	0	00	Background operation
	R1b	0	0	000110	01	1	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	0	00	
CMD7	R1	0	0	000111	01	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	0	00	
	R1b	0	0	000111	01	1	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	0	00	
CMD8	R1	0	0	001000	01	0	0	1	0	0	0	00	00	0	0	0	0	0	0	0	0	0	0	* **	
CMD9	R2	0	0	001001	10	0	0	0	0	0	0	01	10	0	0	0	0	0	0	0	0	0	0	00	
CMD10	R2	0	0	001010	10	0	0	0	0	0	0	01	10	0	0	0	0	0	0	0	0	0	0	00	
CMD12	R1	0	0	001100	01	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	0	00	
	R1b	0	0	001100	01	1	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	0	00	
CMD13	R1	0	0	001101	01	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	0	00	
	R1b	0	0	001101	01	1	1	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	0	00	
CMD14	R1	0	0	001110	01	0	0	1	0	0	0	00	00	0	1	0	0	0	0	0	0	0	1	0 **	
CMD15	-	0	0	001111	00	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	0	00	
CMD16	R1	0	0	010000	01	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	0	00	
CMD17	R1	0	0	010001	01	0	0	1	0	0	0	00	00	0	0	0	0	0	0	0	0	0	0	* **	
CMD18	R1	0	0	010010	01	0	0	1	0	1	0	00	00	0	0	0	0	0	0	0	0	0	0	* **	Pre-defined Open-ended
	R1	0	0	010010	01	0	0	1	0	1	1	00	00	0	0	0	0	0	0	0	0	0	0	* **	
CMD19	R1	0	0	010011	01	0	0	1	1	0	0	00	00	0	0	0	1	0	0	0	0	0	0	**	
CMD23	R1	0	0	010111	01	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	0	00	
CMD24	R1	0	0	011000	01	0	0	1	1	0	0	00	00	0	0	0	0	0	0	0	0	0	0	* **	
CMD25	R1	0	0	011001	01	0	0	1	1	1	0	00	00	0	0	0	0	0	0	0	0	0	0	* **	Pre-defined Open-ended
	R1	0	0	011001	01	0	0	1	1	1	1	00	00	0	0	0	0	0	0	0	0	0	0	* **	
CMD26	R1	0	0	011010	01	0	0	1	1	0	0	00	00	0	0	0	0	0	0	0	0	0	0	* **	
CMD27	R1	0	0	011011	01	0	0	1	1	0	0	00	00	0	0	0	0	0	0	0	0	0	0	* **	
CMD28	R1b	0	0	011100	01	1	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	0	00	
CMD29	R1b	0	0	011101	01	1	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	0	00	

Command	Response	CE_CMD_SET																				Remarks			
		-	-	CMD[5:0]	RTYP[1:0]	RBSY	-	WDAT	DWEN	CMLTE	CMD12EN	RIDXC[1:0]	RCRC7C[1:0]	-	CRC16C	-	CRCSTE	TBIT	OPDM	-	-		SBIT	-	DATW[1:0]
CMD30	R1	0	0	011110	01	0	0	1	0	0	0	00	00	0	0	0	0	0	0	0	0	0	*	**	
CMD31	R1	0	0	011111	01	0	0	1	0	0	0	00	00	0	0	0	0	0	0	0	0	0	*	**	
CMD35	R1	0	0	100011	01	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	0	00	
CMD36	R1	0	0	100100	01	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	0	00	
CMD38	R1b	0	0	100110	01	1	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	0	00	
CMD39	R4	0	0	100111	01	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	0	00	
CMD40	R5	0	0	101000	01	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	0	00	Send CMD
	R5	0	0	101000	01	0	0	0	0	0	0	00	00	0	0	0	0	1	1	0	0	0	0	00	Send RSP
CMD42	R1	0	0	101010	01	0	0	1	1	0	0	00	00	0	0	0	0	0	0	0	0	0	0	**	
CMD55	R1	0	0	110111	01	0	0	0	0	0	0	00	00	0	0	0	0	0	0	0	0	0	0	00	
CMD56	R1	0	0	111000	01	0	0	1	0	0	0	00	00	0	0	0	0	0	0	0	0	0	*	**	Read
	R1	0	0	111000	01	0	0	1	1	0	0	00	00	0	0	0	0	0	0	0	0	0	*	**	Read

Note: • This module does not support CMD11 and CMD20.

## 51.8 Usage Note

### 51.8.1 Card Detection

The CDRISE and CDFALL bits in CE\_DETECT which are used for the card detection function do not have a chattering elimination function. The chattering elimination processing should be implemented by software.

### 51.8.2 Multi-Block Transfer

When HPI is executed, use pre-defined multi-block transfer.

## 52. Motor Control PWM Timer

This LSI has two channels of on-chip motor control PWM (pulse width modulator) timer with a maximum capability of eight pulse outputs for each channel.

### 52.1 Features

- Maximum of 16 pulse outputs
  - Two 10-bit PWM channels, each with eight outputs
  - 10-bit counter (PWCNT) and cycle register (PWCYR)
  - Duty and output polarity can be set for each output.
- Automatic data transfer in every cycle
  - Each of four duty registers (PWDTR) is provided with buffer registers (PWBFR), with data transferred automatically every cycle.
- Duty cycle configurable
  - A duty cycle of 0% to 100% can be configured by means of a duty register.
- Counting clock selectable
  - There is a choice of five counting clocks (P0 $\phi$ , P0 $\phi$ /2, P0 $\phi$ /4, P0 $\phi$ /8, P0 $\phi$ /16).
- High-speed access via internal 16-bit bus
- Two interrupt sources
  - An interrupt can be requested independently for each channel by a cycle register compare match.
- Automatic transfer of register data
  - Block transfer and one-word data transfer are available by activating the direct memory access controller.
- Module stop mode can be set



Figure 52.1 shows a block diagram of the motor control PWM timer.

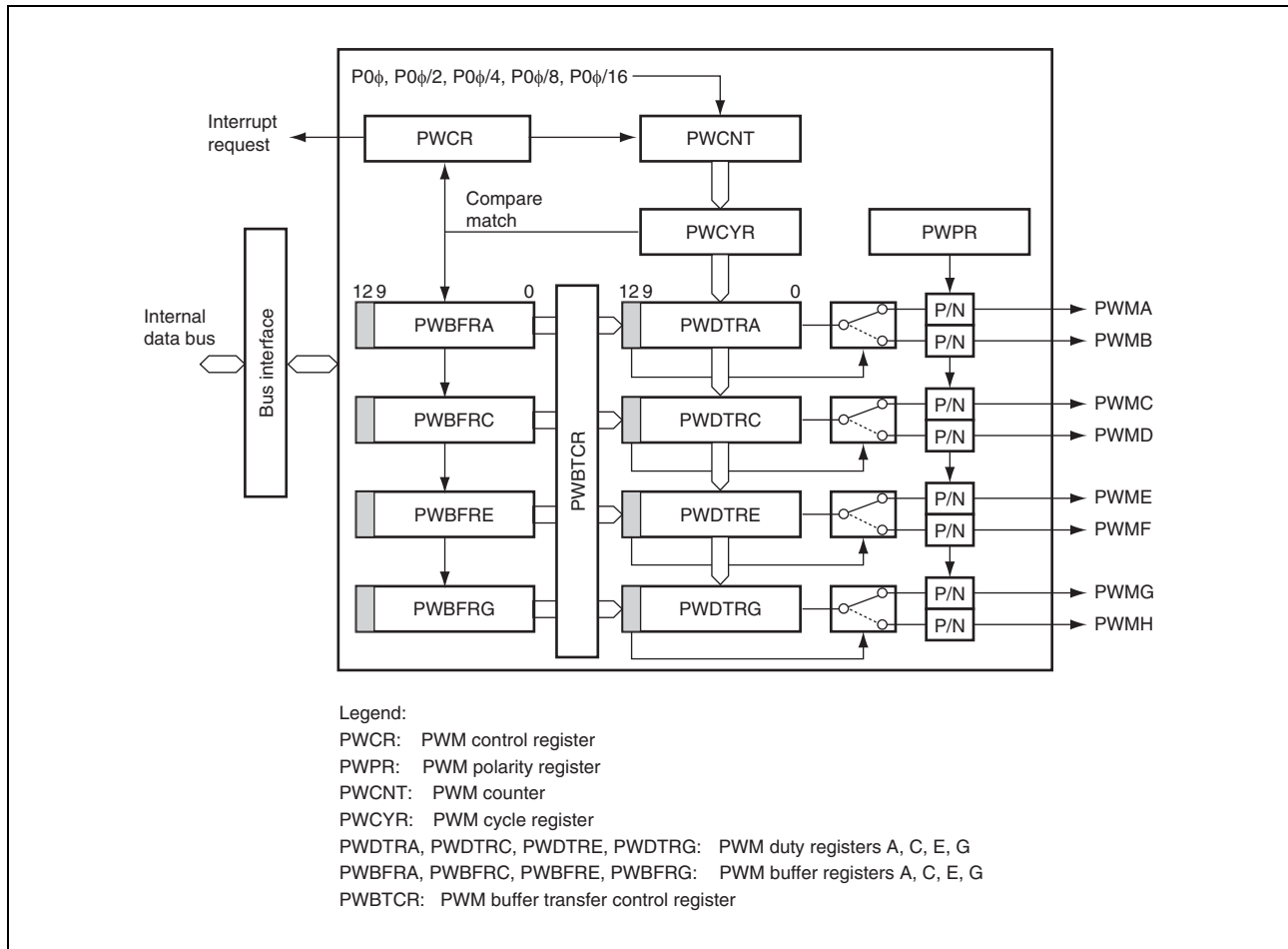


Figure 52.1 Block Diagram of Motor Control PWM Timer

## 52.2 Input/Output Pins

Table 52.1 shows the pin configuration of this module.

**Table 52.1 Pin Configuration**

Channel	Name	Abbrev.	I/O	Function
1	PWM output pin 1A	PWM1A	Output	Channel 1A PWM output
	PWM output pin 1B	PWM1B	Output	Channel 1B PWM output
	PWM output pin 1C	PWM1C	Output	Channel 1C PWM output
	PWM output pin 1D	PWM1D	Output	Channel 1D PWM output
	PWM output pin 1E	PWM1E	Output	Channel 1E PWM output
	PWM output pin 1F	PWM1F	Output	Channel 1F PWM output
	PWM output pin 1G	PWM1G	Output	Channel 1G PWM output
	PWM output pin 1H	PWM1H	Output	Channel 1H PWM output
2	PWM output pin 2A	PWM2A	Output	Channel 2A PWM output
	PWM output pin 2B	PWM2B	Output	Channel 2B PWM output
	PWM output pin 2C	PWM2C	Output	Channel 2C PWM output
	PWM output pin 2D	PWM2D	Output	Channel 2D PWM output
	PWM output pin 2E	PWM2E	Output	Channel 2E PWM output
	PWM output pin 2F	PWM2F	Output	Channel 2F PWM output
	PWM output pin 2G	PWM2G	Output	Channel 2G PWM output
	PWM output pin 2H	PWM2H	Output	Channel 2H PWM output

### 52.3 Register Descriptions

This module has the following registers for each channel as listed in Table 52.2.

**Table 52.2 Register Description**

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
PWM control register_1	PWCR_1	R/W	H'C0	H'FCFF50E0	8
PWM polarity register_1	PWPR_1	R/W	H'00	H'FCFF50E4	8
PWM cycle register_1	PWCYR_1	R/W	H'FFFF	H'FCFF50E6	16
PWM buffer register_1A	PWBFR_1A	R/W	H'EC00	H'FCFF50E8	16
PWM buffer register_1C	PWBFR_1C	R/W	H'EC00	H'FCFF50EA	16
PWM buffer register_1E	PWBFR_1E	R/W	H'EC00	H'FCFF50EC	16
PWM buffer register_1G	PWBFR_1G	R/W	H'EC00	H'FCFF50EE	16
PWM control register_2	PWCR_2	R/W	H'C0	H'FCFF50F0	8
PWM polarity register_2	PWPR_2	R/W	H'00	H'FCFF50F4	8
PWM cycle register_2	PWCYR_2	R/W	H'FFFF	H'FCFF50F6	16
PWM buffer register_2A	PWBFR_2A	R/W	H'EC00	H'FCFF50F8	16
PWM buffer register_2C	PWBFR_2C	R/W	H'EC00	H'FCFF50FA	16
PWM buffer register_2E	PWBFR_2E	R/W	H'EC00	H'FCFF50FC	16
PWM buffer register_2G	PWBFR_2G	R/W	H'EC00	H'FCFF50FE	16
PWM buffer transfer control register	PWBTCR	R/W	H'00	H'FCFF5006	8

### 52.3.1 PWM Control Register\_n (PWCR\_n) (n = 1, 2)

PWCR\_n performs interrupt control, starting/stopping of the counter, and counter clock selection. It also contains a flag that indicates a compare match with the cycle register.

Bit	7	6	5	4	3	2	1	0
Bit Name	—	—	IE	CMF	CST	CKS2	CKS1	CKS0
Initial Value	1	1	0	0	0	0	0	0
R/W	—	—	R/W	R/(W)*	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 1	—	Reserved These bits are always read as 1 and cannot be modified.
5	IE	0	R/W	Interrupt Enable Enables or disables an interrupt request in the event of a compare match with PWCYR_n of the corresponding channel. 0: Interrupt disabled 1: Interrupt enabled
4	CMF	0	R/(W)*	Compare Match Flag Indicates the occurrence of a compare match with PWCYR_n of the corresponding channel. [Setting condition] When PWCNT_n = PWCYR_n – 1 [Clearing conditions] <ul style="list-style-type: none"> <li>• When 0 is written to CMF after reading CMF = 1</li> <li>• When the direct memory access controller is activated by a compare match interrupt, and DMA transfer is executed (When the CPU is used to clear this flag by writing 0 while the corresponding interrupt is enabled, be sure to read the flag after writing 0 to it.)</li> </ul>
3	CST	0	R/W	Counter Start Selects starting or stopping of PWCNT_n of the corresponding channel. 0: PWCNT_n is stopped 1: PWCNT_n is started
2	CKS2	0	R/W	Clock Select These bits select the operating clock for PWCNT_n of the corresponding channel. 000: Counts on P0φ/1 001: Counts on P0φ/2 010: Counts on P0φ/4 011: Counts on P0φ/8 1XX: Counts on P0φ/16
1	CKS1	0	R/W	
0	CKS0	0	R/W	

[Legend]

X: Don't care

Note: \* Only 0 can be written, to clear the flag.

### 52.3.2 PWM Polarity Register\_n (PWPR\_n) (n = 1, 2)

PWPR\_n selects the PWM output polarity.

Bit	7	6	5	4	3	2	1	0
Bit Name	OPSnH	OPSnG	OPSnF	OPSnE	OPSnD	OPSnC	OPSnB	OPSnA
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	OPSnH	0	R/W	Output Polarity Select
6	OPSnG	0	R/W	Each of these bits selects the PWM output polarity. 0: PWM direct output 1: PWM inverse output
5	OPSnF	0	R/W	
4	OPSnE	0	R/W	
3	OPSnD	0	R/W	
2	OPSnC	0	R/W	
1	OPSnB	0	R/W	
0	OPSnA	0	R/W	

(n = 1, 2)

### 52.3.3 PWM Counter\_n (PWCNT\_n) (n = 1, 2)

PWCNT\_n is a 10-bit up-counter incremented by the input clock. The input clock is selected by clock select bits CKS2 to CKS0 in PWCR\_n.

PWCNT\_n can not be directly accessed by the CPU. PWCNT\_n is initialized to H'FC00, when the CST bit in PWCR\_n is cleared to 0.

### 52.3.4 PWM Cycle Register\_n (PWCYR\_n) (n = 1, 2)

PWCYR\_n is a 16-bit readable/writable register that sets the PWM conversion cycle.

Bit:	15	14	13	12	11	10	9	8
	PWC Y15	PWC Y14	PWC Y13	PWC Y12	PWC Y11	PWC Y10	PWC Y9	PWC Y8
Initial value:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	7	6	5	4	3	2	1	0
	PWC Y7	PWC Y6	PWC Y5	PWC Y4	PWC Y3	PWC Y2	PWC Y1	PWC Y0
Initial value:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

When a PWCYR\_n compare match occurs, PWCNT\_n is cleared and data is transferred from the buffer register (PWBFR\_n) to the duty register (PWDTR\_n).

PWCYR\_n should be written to only while PWCNT\_n is stopped. A value of H'FC00 must not be set to PWCYR\_n.

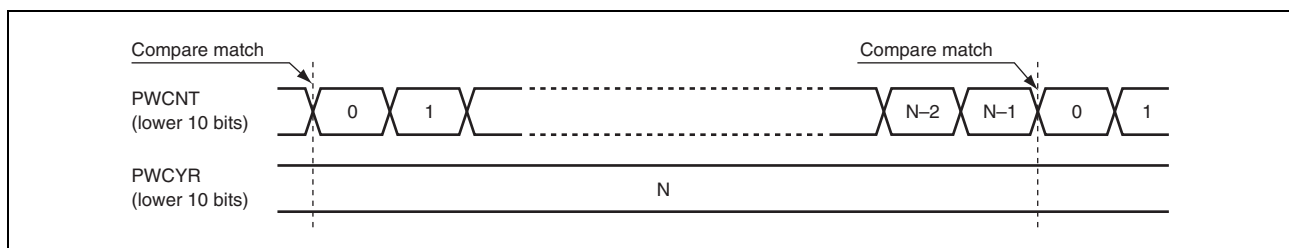


Figure 52.2 Cycle Register Compare Match

### 52.3.5 PWM Duty Registers\_nA, nC, nE, nG (PWDTR\_nA, PWDTR\_nC, PWDTR\_nE, PWDTR\_nG) (n = 1, 2)

There are four PWDTR\_n registers (PWDTR\_nA, PWDTR\_nC, PWDTR\_nE, and PWDTR\_nG). The PWDTR\_nA is used for outputs PWMnA and PWMnB, PWDTR\_nC for outputs PWMnC and PWMnD, PWDTR\_nE for outputs PWMnE and PWMnF, and PWDTR\_nG for outputs PWMnG and PWMnH.

PWDTR\_n can not be directly accessed by the CPU. When a PWCYR\_n compare match occurs, data is transferred from the buffer register (PWBFR\_n) to the duty register (PWDTR\_n). PWDTR\_n is initialized to H'00 when the CST bit is cleared to 0.

Bit	15	14	13	12	11	10	9	8
Bit Name	—	—	—	OTS	—	—	DT9	DT8
Initial Value	—	—	—	0	—	—	0	0
R/W	—	—	—	—	—	—	—	—

Bit	7	6	5	4	3	2	1	0
Bit Name	DT7	DT6	DT5	DT4	DT3	DT2	DT1	DT0
Initial Value	0	0	0	0	0	0	0	0
R/W	—	—	—	—	—	—	—	—

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	—	—	—	Reserved
12	OTS	0	—	Output Terminal Select Selects the pin used for PWM output. Unselected pins output a low level (or a high level when the corresponding bit in PWPR_n is set to 1). For details, see Table 52.3.
11, 10	—	—	—	Reserved
9	DT9	0	—	Duty
8	DT8	0	—	These bits specify the PWM output duty. A high level (or a low level when the corresponding bit in PWPR_n is set to 1) is output from the time PWCNT_n is cleared by a PWCYR_n compare match until a PWDTR_n compare match occurs. When all of the bits are 0, there is no high-level (or low-level when the corresponding bit in PWPR_n is set to 1) output period.
7	DT7	0	—	
6	DT6	0	—	
5	DT5	0	—	
4	DT4	0	—	
3	DT3	0	—	
2	DT2	0	—	
1	DT1	0	—	
0	DT0	0	—	

**Table 52.3 Output Selection by OTS Bit**

Register	Bit 12		Description
	OTS		
PWDTR_1A/PWDTR_2A	0	PWMnA output selected	
	1	PWMnB output selected	
PWDTR_1C/PWDTR_2C	0	PWMnC output selected	
	1	PWMnD output selected	
PWDTR_1E/PWDTR_2E	0	PWMnE output selected	
	1	PWMnF output selected	
PWDTR_1G/PWDTR_2G	0	PWMnG output selected	
	1	PWMnH output selected	

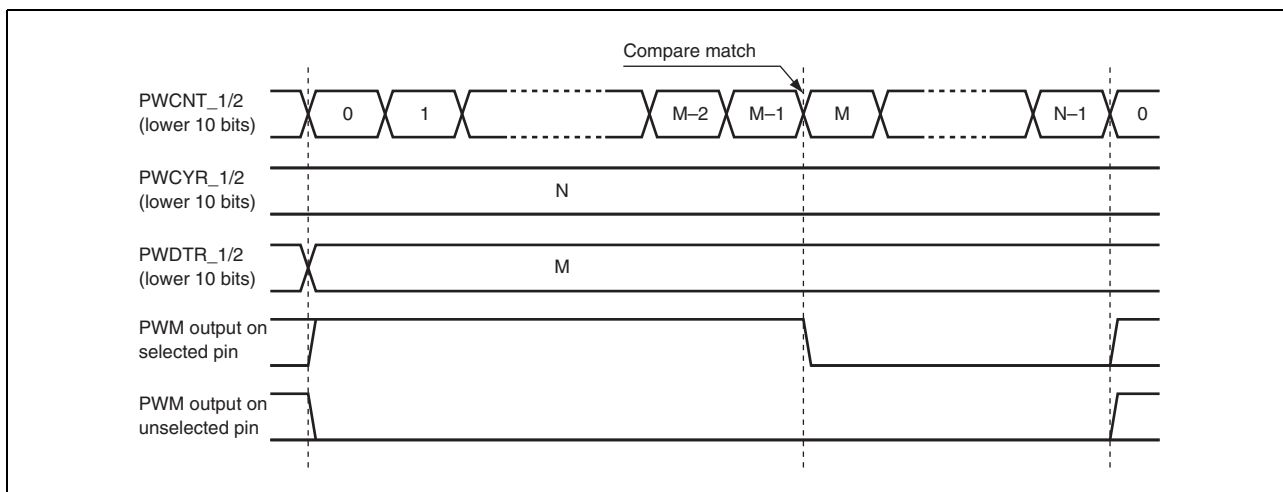


Figure 52.3 Duty Register Compare Match (OPS = 0 in PWPR\_n)

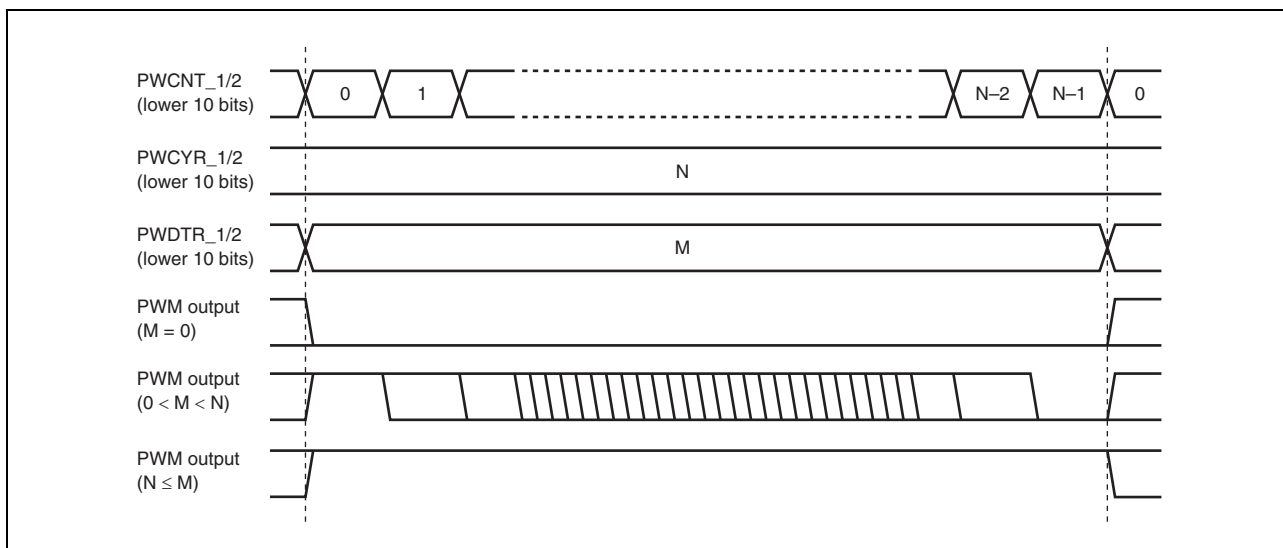


Figure 52.4 Differences in PWM Output According to Duty Register Set Value (OPS = 0 in PWPR\_n)

### 52.3.6 PWM Buffer Registers\_nA, nC, nE, nG (PWBFR\_nA, PWBFR\_nC, PWBFR\_nE, PWBFR\_nG) (n = 1, 2)

There are four PWBFR\_n registers (PWBFR\_nA, PWBFR\_nC, PWBFR\_nE, and PWBFR\_nG). When a PWCYR\_n compare match occurs, data is transferred from the buffer register (PWBFR\_n) to the duty register (PWDTR\_n).

Bit:	15	14	13	12	11	10	9	8
	—	—	—	OTS	—	—	DT9	DT8
Initial Value:	1	1	1	0	1	1	0	0
R/W:	R	R	R	R/W	R	R	R/W	R/W
Bit:	7	6	5	4	3	2	1	0
	DT7	DT6	DT5	DT4	DT3	DT2	DT1	DT0
Initial Value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15 to 13	—	All 1	R	Reserved These bits are always read as 1 and cannot be modified.
12	OTS	0	R/W	Output Terminal Select Holds the data to be sent to bit 12 in PWDTR_n.
11, 10	—	All 1	R	Reserved These bits are always read as 1 and cannot be modified.
9	DT9	0	R/W	Duty These bits hold the data to be sent to bits 9 to 0 in PWDTR_n.
8	DT8	0	R/W	
7	DT7	0	R/W	
6	DT6	0	R/W	
5	DT5	0	R/W	
4	DT4	0	R/W	
3	DT3	0	R/W	
2	DT2	0	R/W	
1	DT1	0	R/W	
0	DT0	0	R/W	



### 52.3.7 PWM Buffer Transfer Control Register (PWBTCCR)

PWBTCCR enables or disables the data transfer from buffer register to duty register with the compare match of PWM counter and PWM cycle register.

Bit	7	6	5	4	3	2	1	0
Bit Name	BTC2G	BTC2E	BTC2C	BTC2A	BTC1G	BTC1E	BTC1C	BTC1A
Initial Value	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	BTC2G	0	R/W	0: Data transfer from PWBFR <sub>n</sub> to PWDTR <sub>n</sub> is enabled with PWCNT <sub>n</sub> and PWCYR <sub>n</sub> compare match
6	BTC2E	0	R/W	1: Data transfer from PWBFR <sub>n</sub> to PWDTR <sub>n</sub> is disabled with PWCNT <sub>n</sub> and PWCYR <sub>n</sub> compare match
5	BTC2C	0	R/W	
4	BTC2A	0	R/W	
3	BTC1G	0	R/W	
2	BTC1E	0	R/W	
1	BTC1C	0	R/W	
0	BTC1A	0	R/W	

## 52.4 Bus Master Interface

### 52.4.1 16-Bit Data Registers

PWCYR\_n and PWBFR\_n are 16-bit registers. These registers are linked to the bus master by a 16-bit data bus, and can be read or written in 16-bit units. They cannot be read or written by 8-bit access; 16-bit access must always be used.

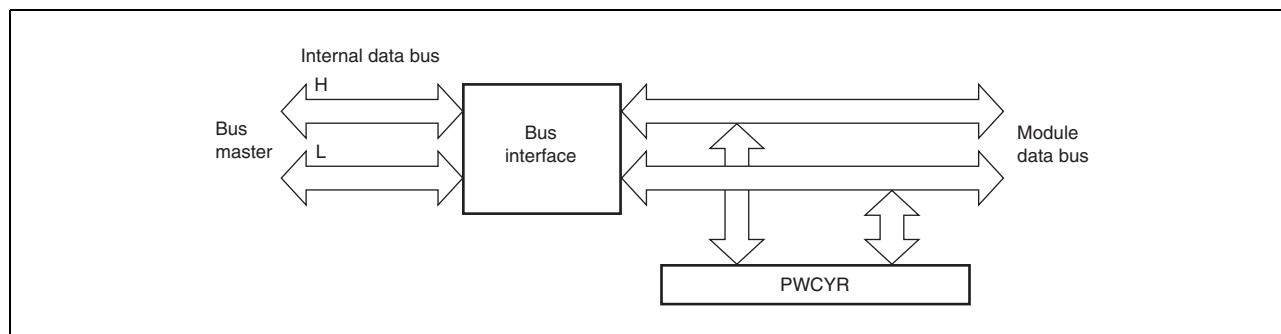


Figure 52.5 16-Bit Register Access Operation (Bus Master ↔ PWCYR\_n (16 Bits))

### 52.4.2 8-Bit Data Registers

PWCR\_n, PWPR\_n, and PWBTCR are 8-bit registers that can be read and written to in 8-bit units. These registers are linked to the bus master by a 16-bit data bus, and can be read or written by 16-bit access; in this case, the lower eight bits are read as H'FF.

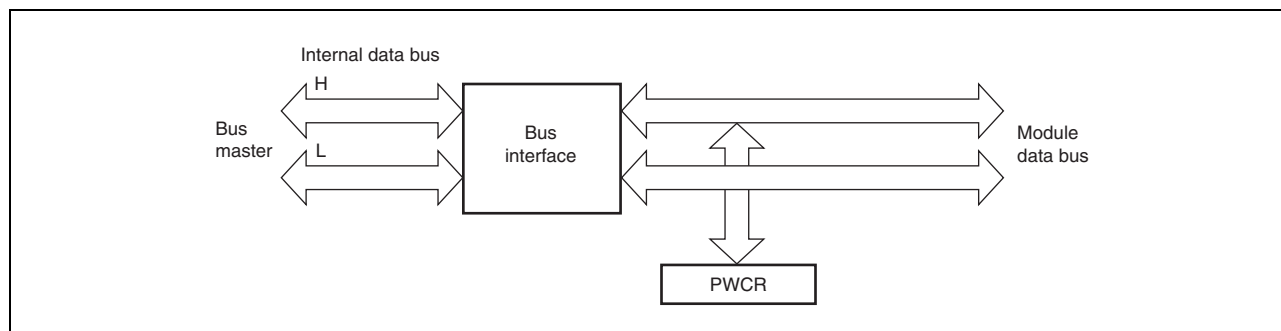


Figure 52.6 8-Bit Register Access Operation (Bus Master ↔ PWCR\_n (Upper Eight Bits))

## 52.5 Operation

### 52.5.1 PWM Operation

PWM waveforms are output from pins PWM1A to PWM1H and PWM2A to PWM2H as shown in Figure 52.7.

#### (1) Initial Settings

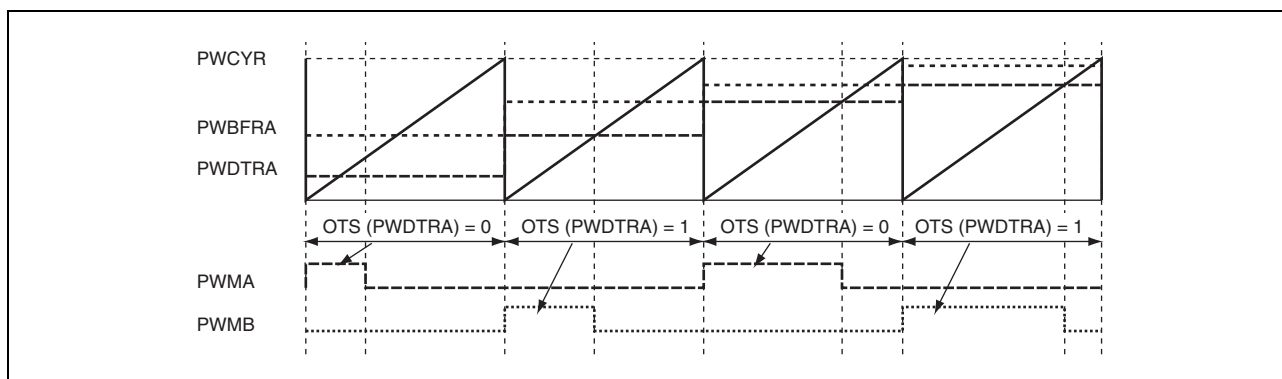
Set the PWM output polarity in PWPR\_n; select the clock to be input to PWCNT\_n with the CKS2 to CKS0 bits in PWCR\_n; set the PWM conversion cycle in PWCYR\_n; and set the first frame of data in PWBFR\_nA, PWBFR\_nC, PWBFR\_nE, and PWBFR\_nG.

#### (2) Activation

When the CST bit in PWCR\_n is set to 1, PWCNT\_n starts counting up. On compare match between PWCNT\_n and PWCYR\_n, data is transferred from the buffer register to the duty register and the CMF bit in PWCR\_n is set to 1. At the same time, if the IE bit in PWCR\_n has been set to 1, an interrupt can be requested or the direct memory access controller can be activated.

#### (3) Waveform Output

The PWM outputs selected by the OTS bits in PWDTR\_nA, PWDTR\_nC, PWDTR\_nE, and PWDTR\_nG go high when a compare match occurs between PWCNT\_n and PWCYR\_n. The PWM outputs not selected by the OTS bit are low. When a compare match occurs between PWCNT\_n and PWDTR\_nA, PWDTR\_nC, PWDTR\_nE, or PWDTR\_nG, the corresponding PWM output goes low. If the corresponding bit in PWPR\_n is set to 1, the output is inverted.



**Figure 52.7** PWM Operation

#### (4) Next Frame

When a compare match occurs between PWCNT\_n and PWCYR\_n, data is transferred from the buffer register to the duty register. PWCNT\_n is reset and starts counting up from H'000. The CMF bit in PWCR\_n is set, and if the IE bit in PWCR\_n has been set, an interrupt can be requested or the direct memory access controller can be activated.

#### (5) Stopping

When the CST bit in PWCR\_n is cleared to 0, PWCNT\_n is reset and stops. All PWM outputs go low (or high if the corresponding bit in PWPR\_n is set to 1).

### 52.5.2 Buffer Transfer Control

Setting a corresponding bit in the PWM buffer transfer control register disables a buffer transfer on compare match. This prevents the output from changing when compare match occurs while the buffer register is being changed. A buffer transfer on compare match is resumed after cleaning the bit.

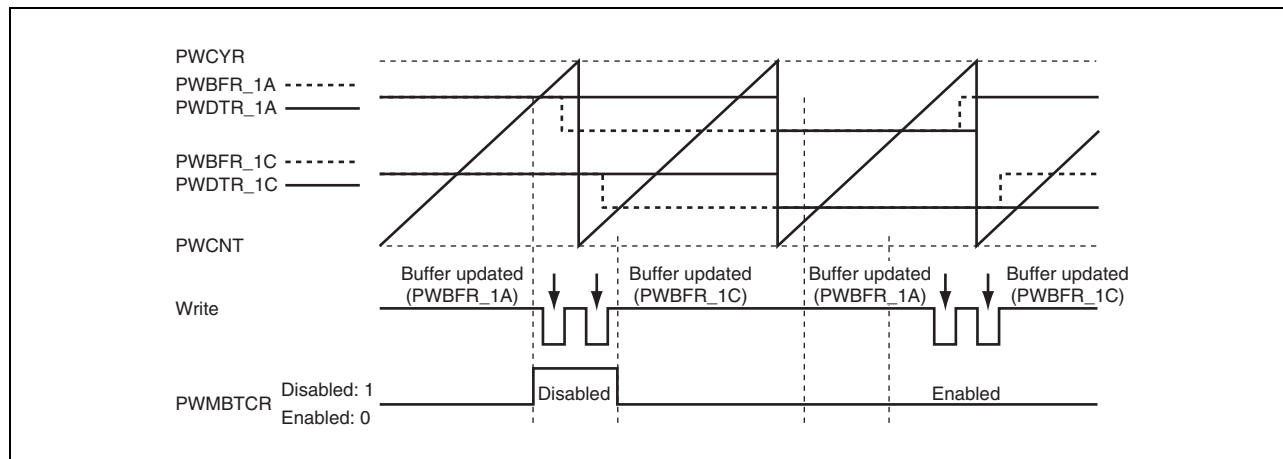


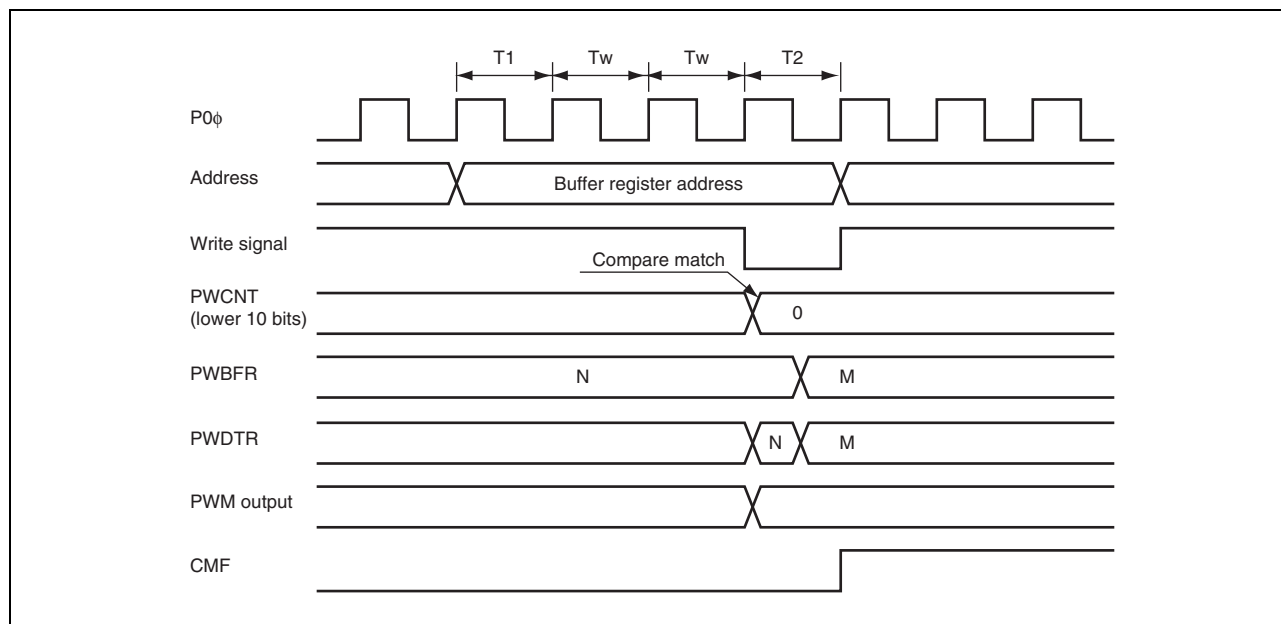
Figure 52.8 Disabling Buffer Transfer

## 52.6 Usage Note

### 52.6.1 Conflict between Buffer Register Write and Compare Match

If a PWBFR\_n write is performed in the state immediately after a cycle register compare match, the buffer register and duty register are both modified. PWM output changed by the cycle register compare match is not changed by modification of the duty register due to conflict. This may result in unanticipated duty output.

Buffer register modification must be completed before automatic transfer by the direct memory access controller, exception handling due to a compare match interrupt, or the occurrence of a cycle register compare match on detection of the rise of CMF (compare match flag) in PWCR\_n.



**Figure 52.9 Conflict between Buffer Register Write and Compare Match**

## 53. On-Chip RAM

This LSI has an on-chip large-capacity RAM for display area and work area (128 Kbytes of this RAM are shared with the on-chip data retention RAM) and an on-chip data retention RAM, which can retain data in deep standby mode. These memory units can be used to store instructions or data.

The operation and write access to the large-capacity RAM (including on-chip data retention RAM) can be enabled or disabled through the RAM enable bits and RAM write enable bits.

The on-chip data retention RAM is assigned to page 0 in the on-chip large-capacity RAM. Retention or non-retention of data by the on-chip data retention RAM in deep standby mode is selectable on a per-page basis.

### 53.1 Features

- Page

- The on-chip large-capacity RAM consists of five pages.
- The on-chip data retention RAM consists of four pages. Page 0 has 16-Kbytes, page 1 has 16-Kbytes, page 2 has 32-Kbytes, and page 3 has 64-Kbytes.

- Memory map

The on-chip RAM is located in the address spaces shown in tables 47.1 to 47.3. Each page of the large-capacity RAM in products with a 10-Mbyte on-chip RAM has a capacity of 2 Mbytes, while each page of the large-capacity RAM in products with a 5-Mbyte on-chip RAM has a capacity of 1 Mbyte. Except for the size, the address maps for products with a 5-Mbyte on-chip RAM and with a 10-Mbyte on-chip RAM are compatible. Furthermore, to obtain a continuous address space for 5-Mbyte RAM compatibility when a 10-Mbyte RAM is in use, each page of the latter is divided into two. If you wish to use the pages continuously when using a 10-Mbyte RAM, use mirror addresses.

**Table 53.1 Address Spaces of On-Chip Large-Capacity RAM**

Page	Address	Mirror Address
Page 0 (2048KB)	H'20000000 to H'200FFFFFFF	H'60000000 to H'600FFFFFFF
	H'20500000 to H'205FFFFFFF	H'60100000 to H'601FFFFFFF
Page 1 (2048KB)	H'20600000 to H'206FFFFFFF	H'60200000 to H'602FFFFFFF
	H'20100000 to H'201FFFFFFF	H'60300000 to H'603FFFFFFF
Page 2 (2048KB)	H'20200000 to H'202FFFFFFF	H'60400000 to H'604FFFFFFF
	H'20700000 to H'207FFFFFFF	H'60500000 to H'605FFFFFFF
Page 3 (2048KB)	H'20800000 to H'208FFFFFFF	H'60600000 to H'606FFFFFFF
	H'20300000 to H'203FFFFFFF	H'60700000 to H'607FFFFFFF
Page 4 (2048KB)	H'20400000 to H'204FFFFFFF	H'60800000 to H'608FFFFFFF
	H'20900000 to H'209FFFFFFF	H'60900000 to H'609FFFFFFF

**Table 53.2 Address Spaces of On-Chip Data Retention RAM**

Page	Address	Mirror Address
Page 0 (16KB)	H'20000000 to H'20003FFF	H'60000000 to H'60003FFF
Page 1 (16KB)	H'20004000 to H'20007FFF	H'60004000 to H'60007FFF
Page 2 (32KB)	H'20008000 to H'2000FFFF	H'60008000 to H'6000FFFF
Page 3 (64KB)	H'20010000 to H'2001FFFF	H'60010000 to H'6001FFFF

- Ports

Each page of the on-chip large-capacity RAM has one read and write port and is connected to the AXI bus. The on-chip RAM for data retention, which has a port independent of the port in page 0, is shared with the read and write port in four pages.

- Method of arbitration

When the same port of the on-chip large-capacity RAM is accessed from different masters simultaneously, the AXI bus performs arbitration in round-robin mode.

- Number of access cycles

The number of cycles for access to read or write is one cycle of  $B\phi$ .

## 53.2 Usage Notes

### 53.2.1 Page Conflict

When the same page of the on-chip large-capacity RAM is accessed from different masters simultaneously, a conflict on the page occurs. Although each access is completed correctly, this kind of conflict degrades the memory access speed. Therefore, it is advisable to provide software measures to prevent such conflicts as far as possible. For example, no conflict will arise if different pages are accessed by each master.

### 53.2.2 Data Retention

Data in the large-capacity RAM (including on-chip data retention RAM) are retained in the states other than power-on reset and deep standby mode. In power-on reset and deep standby mode, these RAMs operate as described below.

#### (1) Power-on Reset

##### (a) On-Chip Large-Capacity RAM (Excluding On-Chip Data Retention RAM)

Data are retained on a power-on reset by disabling the setting of either the VRAME or VRAMWE bit. Data are not retained when the setting of the VRAME and VRAMWE bits are both enabled.

##### (b) On-Chip Data Retention RAM

Data are retained on a power-on reset by disabling the setting of any of the VRAME, VRAMWE, or RRAMWE, excluding the case that deep standby mode is canceled by power-on reset. Data are not retained when the setting of the VRAME, VRAMWE and RRAMWE bits are all enabled.

#### (2) Deep Standby Mode

##### (a) On-Chip Large-Capacity RAM (Excluding On-Chip Data Retention RAM)

Data are not retained.

##### (b) On-Chip Data Retention RAM

Data are retained in deep standby mode by enabling the setting of the RRAMKP bit, excluding the case that deep standby mode is canceled by power-on reset. In the case that deep standby mode is canceled by interrupt or pins for cancelling, power-on reset exception handling is executed, but the data are retained.



## 54. Ports

### 54.1 Features

The 256-pin version of this LSI has eleven ports (JP0 and P0 to P9) and the 324-pin version has thirteen ports (JP0 and P0 to P11). All port pins are multiplexed with other peripheral module pin functions, and by setting the control registers, multiplexed pin functions (alternative functions) can be selected.

#### 54.1.1 Port group

This LSI has the following number of port groups.

**Table 54.1 Port Group**

Package	Number	Name
RZ/A1H 256-pin	11	P0 to P9, JP0
RZ/A1H 324-pin	13	P0 to P11, JP0

#### 54.1.2 Port group index n

Throughout this section, the individual port groups are identified by the index "n" (n = 0 to 11), for example PMCN for the port mode control register, which is used for the Pn pin.

#### 54.1.3 Base address

The addresses of the registers used to control the JTAG ports are given as addresses offset from the base address <JPORTn\_base>. The addresses of the other registers used to control the general ports are given as addresses offset from the base address <PORTn\_base>.

**Table 54.2 Base Addresses <PORTn\_base> and <JPORTn\_base>**

<PORTn_base> address	<JPORTn_base> address
H'FCFE 3000	H'FCFE 7B00

## 54.2 Functional Overview

### 54.2.1 Mode of Pin Functions

Pins can operate in three modes.

Port mode (PM<sub>Cn</sub>.PM<sub>Cnm</sub> bit = 0)

A pin in port mode operates as a general purpose input/output pin. The I/O mode is selected by setting the PM<sub>n</sub>.PM<sub>nm</sub> bit.

Software I/O control alternative mode (PM<sub>Cn</sub>.PM<sub>Cnm</sub> bit = 1, PIP<sub>Cn</sub>.PIP<sub>Cnm</sub> bit = 0)

In this mode, the pins operate as alternative functions.

The I/O mode is selected by setting the PM<sub>n</sub>.PM<sub>nm</sub> bit by using software.

Direct I/O control alternative mode (PM<sub>Cn</sub>.PM<sub>Cnm</sub> bit = 1, PIP<sub>Cn</sub>.PIP<sub>Cnm</sub> bit = 1)

In this mode, the pins operate as alternative functions.

Unlike the software I/O alternative mode, however, the I/O mode is selected by the alternative function.

An overview of the register settings is given in the tables below.

**Table 54.3 Pin Function Configuration (Overview)**

Mode	Bit			I/O
	PM <sub>Cn</sub> . PM <sub>Cnm</sub>	PM <sub>n</sub> . PM <sub>nm</sub>	PIP <sub>Cn</sub> . PIP <sub>Cnm</sub>	
Port mode	0	0	X	O
		1*1		I
Software I/O control alternative mode	1*2	0	0	O
		1		I
Direct I/O control alternative mode		X	1	Controlled by the alternative function

Note 1. The input buffer can be enabled in the port mode only when the PIB<sub>Cn</sub>.PIB<sub>Cnm</sub> bit is set to 1.

Note 2. When the pins are in alternative mode (PM<sub>Cn</sub>.PM<sub>Cnm</sub> = 1), one of the eight alternative functions is selected by the PFC<sub>n</sub>, PFC<sub>En</sub>, and PFCA<sub>En</sub> registers. For details, see Table 54.6, Alternative Mode Selection\*2.

## 54.2.2 Pin Data Input/Output

The registers used for data input/output are described below. The location that is read via the PPRn register differs depending on the pin mode.

- Output data

In the port mode (PMcn.PMCnm = 0), the value of the Pn.Pnm bit is output from the Pn\_m pin.

- Input data

When the PPRn register is read, either the value of the Pn\_m pin, the value of the Pn.Pnm bit, or the value output by the alternative function is returned. Which value is returned depends on the pin mode and setting of several control bits. The different PPRn read values are shown in the table below.

**Table 54.4 PPRnm Read Values**

PMcn. PMcnm	PMn. PMnm	PIBCn. PIBCnm	PIPCn. PIPCnm	Mode	PPRnm Read Value
0	1	0	X	Port input, input buffer disabled	Pn.Pnm register
		1		Port input, input buffer enabled	Pn_m pin
	0	X	Port output	Pn.Pnm register*1	
1	1	X	0	Software I/O control alternative function input	Pn_m pin
	0			Software I/O control alternative function output	Alternative-function internal output signal*1
	X		1	Direct I/O control alternative function	I/O port in alternative mode: • Input: Pn_m pin • Output: Alternative-function internal output signal*1

Note 1. When the PBDCn.PBDCnm bit is 1, the level of the Pn\_m pin is returned by the PPRn.PPRnm bit.

### 54.3 Register Description

The following registers are used for setting the individual pins of the port groups:

**Table 54.5 Register Configuration**

Symbol	Register Name	R/W	16 Bit Access	32 Bit Access	Address
Pn	Port register	R/W	√	—	<PORTn_base> + 0000H + n × 4
PSRn	Port set/reset register	R/W	—	√	<PORTn_base> + 0100H + n × 4
PPRn	Port pin read register	R	√	—	<PORTn_base> + 0200H + n × 4
PMn	Port mode register	R/W	√	—	<PORTn_base> + 0300H + n × 4
PMCn	Port mode control register	R/W	√	—	<PORTn_base> + 0400H + n × 4
PFCn	Port function control register	R/W	√	—	<PORTn_base> + 0500H + n × 4
PFCEn	Port function control expansion register	R/W	√	—	<PORTn_base> + 0600H + n × 4
PNOTn	Port NOT register	W	√	—	<PORTn_base> + 0700H + n × 4
PMSRn	Port mode set/reset register	R/W	—	√	<PORTn_base> + 0800H + n × 4
PMCSRn	Port mode control set/reset register	R/W	—	√	<PORTn_base> + 0900H + n × 4
PFCAEn	Port Function Control Additional Expansion register	R/W	√	—	<PORTn_base> + 0A00H + n × 4
PIBCn	Port input buffer control register	R/W	√	—	<PORTn_base> + 4000H + n × 4
PBDCn	Port bi-direction control register	R/W	√	—	<PORTn_base> + 4100H + n × 4
PIPCn	Port IP control register	R/W	√	—	<PORTn_base> + 4200H + n × 4
JPPR0	Port pin read register	R	√	—	<JPORTn_base> + 0020H
JPMC0	Port mode control register	R/W	√	—	<JPORTn_base> + 0040H
JPMCSR0	Port mode control set/reset register	R/W	—	√	<JPORTn_base> + 0090H
JPIBC0	Port input buffer control register	R/W	√	—	<JPORTn_base> + 0400H
SNCRn	Serial sound interface noise canceler control register	R/W	—	√	<PORTn_base> + 0C00H

#### 54.3.1 Port Register (Pn)

In output port mode (PMCn.PMCnm = 0 and PMn.PMnm = 0), this register holds the Pn.Pnm data to be output via the Pn\_m pin.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Pn15	Pn14	Pn13	Pn12	Pn11	Pn10	Pn9	Pn8	Pn7	Pn6	Pn5	Pn4	Pn3	Pn2	Pn1	Pn0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit Position	Bit Name	Function
15 to 0	Pn[15:0]	These bits set the output level of the Pn_m pin (m = 0 to 15). 0: Outputs low level 1: Outputs high level

### 54.3.2 Port Set and Reset Register (PSRn)

This register provides an alternative method for writing data to the Pn register.

The higher 16 bits of the PSRn register specify whether data can be written to the Pn.Pnm bit specified by the lower 16 bits of the PSRn register.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PSRn 31	PSRn 30	PSRn 29	PSRn 28	PSRn 27	PSRn 26	PSRn 25	PSRn 24	PSRn 23	PSRn 22	PSRn 21	PSRn 20	PSRn 19	PSRn 18	PSRn 17	PSRn 16
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PSRn 15	PSRn 14	PSRn 13	PSRn 12	PSRn 11	PSRn 10	PSRn9 9	PSRn8 8	PSRn7 7	PSRn6 6	PSRn5 5	PSRn4 4	PSRn3 3	PSRn2 2	PSRn1 1	PSRn0 0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit Position	Bit Name	Function
31 to 16	PSRn[31:16]	These are enable bits that specify whether the value of the corresponding lower bit of the PSRn.PSRnm bits is written to the corresponding Pn.Pnm bit. 0: Pn.Pnm is independent of PSRn.PSRnm 1: Pn.Pnm becomes the value of the PSRn.PSRnm bit. Example: If PSRn.PSRn31 = 1, the value of PSRn.PSRn15 is written to Pn.Pn15.
15 to 0	PSRn[15:0]	These are data bits that specify the value of the Pn.Pnm bit if the corresponding higher bit, PSRn.PSRn (m+16), is 1. 0: Pn.Pnm = 0 1: Pn.Pnm = 1

### 54.3.3 Port Pin Read Register (PPRn/JPPR0)

The PPRn register reflects either the level of the Pn\_m pin, the value of the Pn.Pnm bit, or the output level of the alternative function. The value read depends on various register settings as described in Table 54.4, PPRnm Read Values.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PPRn 15	PPRn 14	PPRn 13	PPRn 12	PPRn 11	PPRn 10	PPRn9 9	PPRn8 8	PPRn7 7	PPRn6 6	PPRn5 5	PPRn4 4	PPRn3 3	PPRn2 2	PPRn1 1	PPRn0 0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit Position	Bit Name	Function
15 to 0	PPRn[15:0]	The value of the Pn_m pin, the value of the Pn.Pnm bit, or the output level of the alternative function.

- Notes:
1. For the PPRn register read value, see section 54.2.2, Pin Data Input/Output.
  2. The JPPR0[1:0] bits are control bits in the JTAG port pin read register (JPPR0).

### 54.3.4 Port Mode Register (PMn)

The PMn register specifies whether the Pn\_m pins are in input mode or in output mode.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PMn	PMn	PMn	PMn	PMn	PMn	PMn9	PMn8	PMn7	PMn6	PMn5	PMn4	PMn3	PMn2	PMn1	PMn0
	15	14	13	12	11	10										
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit Position	Bit Name	Function
15 to 0	PMn[15:0]	Specifies input/output mode of the corresponding pin: 0: Output mode (output enabled) 1: Input mode (output disabled)

- Notes:
1. To use the Pn\_m pin in input port mode (PMCn.PMCnm = 0 and PMn.PMnm = 1), the input buffer must be enabled by setting the PIBCn.PIBCnm bit to 1.
  2. After reset is issued, the PIPCn.PIPCnm bit is set to 0 and the PMn.PMnm bit specifies the I/O direction for port mode (PMCn.PMCnm = 0) and alternative mode (PMCn.PMCnm = 1). For the pins whose functions are automatically selected in boot mode (see Table 8.3, Initial States by Areas in Boot Modes 0, 1, and 2 to 5), the I/O direction is controlled by the alternative functions instead of the PMn register.

### 54.3.5 Port Mode Control Register (PMCn/JPMC0)

This register is used to specify whether the Pn\_m pins are in port mode or in alternative mode.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PMCn	PMCn	PMCn	PMCn	PMCn	PMCn	PMCn	PMCn	PMCn	PMCn	PMCn	PMCn	PMCn	PMCn	PMCn	PMCn
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit Position	Bit Name	Function
15 to 0	PMCn[15:0]	Specifies the operation mode of the corresponding pin: 0: Port mode 1: Alternative mode

Caution: The input/output control is not performed just by setting alternative mode (PMCn.PMCnm). Set 1 in the PIPCn.PIPCnm bit too when the I/O control is to be performed by using the alternative function.

Note: The JPMC0[1:0] bits are control bits in the JTAG port mode control register (JPMC0).

### 54.3.6 Port Function Control Register (PFCn)

This register, together with the PFCEn and PFCAEn register, specifies the alternative function of the pins.

The I/O direction of several alternative functions can be controlled directly by using the Pn\_m pin. For these alternative functions, the PIPCN.PIPCnm bit must be set to 1.

For all other alternative functions, the I/O direction is specified by using the PMn.PMnm bit.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PFCn 15	PFCn 14	PFCn 13	PFCn 12	PFCn 11	PFCn 10	PFCn9	PFCn8	PFCn7	PFCn6	PFCn5	PFCn4	PFCn3	PFCn2	PFCn1	PFCn0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit Position	Bit Name	Function
15 to 0	PFCn[15:0]	Specifies the alternative function of a pin.

**Table 54.6 Alternative Mode Selection\*2**

PMcn.PMCnm	PFCAEn.PFCAEnm	PFCEn.PFCEnm	PFCn.PFCnm	Pn.PMnm*1	Function		
0	X	X	X	1	Port mode/input		
				0	Port mode/output		
1	0	0	0	1	Alternative mode/1st alternative function/input		
				0	Alternative mode/1st alternative function/output		
			1	1	Alternative mode/2nd alternative function/input		
				0	Alternative mode/2nd alternative function/output		
			1	0	1	Alternative mode/3rd alternative function/input	
				0	0	Alternative mode/3rd alternative function/output	
			1	0	0	1	Alternative mode/4th alternative function/input
						0	Alternative mode/4th alternative function/output
	1	0	0	0	1	Alternative mode/5th alternative function/input	
					0	Alternative mode/5th alternative function/output	
		1	0	0	1	1	Alternative mode/6th alternative function/input
						0	Alternative mode/6th alternative function/output
		1	0	0	1	1	Alternative mode/7th alternative function/input
						0	Alternative mode/7th alternative function/output
		1	0	0	1	1	Alternative mode/8th alternative function/input
						0	Alternative mode/8th alternative function/output

Note 1. When the PIPCN.PIPCnm bit is set to 1, the I/O direction is directly controlled by the alternative function and the setting of the PMn.PMnm bit becomes invalid.

Note 2. Use alternative functions that are open to users. If a mode to which no alternative function is assigned is selected, correct operation is not guaranteed.

### 54.3.7 Port Function Control Expansion Register (PFCEn)

This register, together with the PFCn and PFCAEn register, specifies the alternative function of the pins.

The I/O direction of several alternative functions can be controlled directly by using the Pn\_m pin. For these alternative functions, the PIPCn.PIPCnm bit must be set to 1.

For all other alternative functions, the I/O direction is specified by using the PMn.PMnm bit.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PFCEn	PFCEn	PFCEn	PFCEn	PFCEn	PFCEn	PFCEn	PFCEn	PFCEn	PFCEn	PFCEn	PFCEn	PFCEn	PFCEn	PFCEn	PFCEn
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit Position	Bit Name	Function
15 to 0	PFCEn[15:0]	Specifies the alternative function of a pin.

### 54.3.8 Port NOT Register (PNOTn)

This register allows a Pn.Pnm bit to be inverted without directly writing to the Pn register.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PNOTn	PNOTn	PNOTn	PNOTn	PNOTn	PNOTn	PNOTn	PNOTn	PNOTn	PNOTn	PNOTn	PNOTn	PNOTn	PNOTn	PNOTn	PNOTn
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit Position	Bit Name	Function
15 to 0	PNOTn[15:0]	Specifies whether to invert Pn.Pnm. 0: Pn.Pnm is not inverted ( $Pnm \rightarrow Pnm$ ) 1: Pn.Pnm is inverted ( $\overline{Pnm} \rightarrow Pnm$ )



### 54.3.9 Port Mode Set and Reset Register (PMSRn)

This register provides an alternative method for writing data to the PMn register.

The higher 16 bits of the PMSRn register specify whether data can be written to the PMn.PMnm bit specified by the lower 16 bits of the PMSRn register.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PMSRn 31	PMSRn 30	PMSRn 29	PMSRn 28	PMSRn 27	PMSRn 26	PMSRn 25	PMSRn 24	PMSRn 23	PMSRn 22	PMSRn 21	PMSRn 20	PMSRn 19	PMSRn 18	PMSRn 17	PMSRn 16
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PMSRn 15	PMSRn 14	PMSRn 13	PMSRn 12	PMSRn 11	PMSRn 10	PMSRn 9	PMSRn 8	PMSRn 7	PMSRn 6	PMSRn 5	PMSRn 4	PMSRn 3	PMSRn 2	PMSRn 1	PMSRn 0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit Position	Bit Name	Function
31 to 16	PMSRn[31:16]	<p>These are enable bits that specify whether the value of the corresponding lower bit of the PMSRn.PMSRnm bits is written to the corresponding PMn.PMnm bit.</p> <p>0: PMn.PMnm does not depend on the PMSRn.PMSRnm bit.</p> <p>1: PMn.PMnm becomes the value of the PMSRn.PMSRnm bit.</p> <p>Example:</p> <p>If PMSRn.PMSRn31 = 1, the value of PMSRn.PMSRn15 is written to PMn.PMn15.</p>
15 to 0	PMSRn[15:0]	<p>These are data bits that specify the value of the PMn.PMnm bit if the corresponding higher bit, PMSRn.PMSRn(m+16), is 1.</p> <p>0: PMn.PMnm = 0</p> <p>1: PMn.PMnm = 1</p>

### 54.3.10 Port Mode Control Set and Reset Register (PMCSRn/JPMCSR0)

This register provides an alternative method for writing data to the PMCn register.

The higher bits of the PMCSRn register specify whether data can be written to the PMCn.PMCnm bit specified by the lower bits of the PMCSRn register.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PMCS Rn31	PMCS Rn30	PMCS Rn29	PMCS Rn28	PMCS Rn27	PMCS Rn26	PMCS Rn25	PMCS Rn24	PMCS Rn23	PMCS Rn22	PMCS Rn21	PMCS Rn20	PMCS Rn19	PMCS Rn18	PMCS Rn17	PMCS Rn16
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PMCS Rn15	PMCS Rn14	PMCS Rn13	PMCS Rn12	PMCS Rn11	PMCS Rn10	PMCS Rn9	PMCS Rn8	PMCS Rn7	PMCS Rn6	PMCS Rn5	PMCS Rn4	PMCS Rn3	PMCS Rn2	PMCS Rn1	PMCS Rn0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit Position	Bit Name	Function
31 to 16	PMCSRn[31:16]	These are enable bits that specify whether the value of the corresponding lower bit of the PMCSRn.PMCn bits is written to the corresponding PMCn.PMCnm bit. 0: PMCn.PMCnm does not depend on the PMCSRn.PMCn bit. 1: PMCn.PMCnm becomes the value of the PMCSRn.PMCn bit. Example: If PMCSRn.PMCn31 = 1, the value of PMCSRn.PMCn15 is written to PMCn.PMCn15.
15 to 0	PMCSRn[15:0]	These are data bits that specify the value of the PMCn.PMCnm bit if the corresponding higher bit, PMCSRn.PMCn(m+16), is 1. 0: PMCn.PMCnm = 0 1: PMCn.PMCnm = 1

Note: The JPMCSR0[1:0] bits are control bits in the JTAG port mode control set/reset register (JPMCSR0).

### 54.3.11 Port Function Control Additional Expansion Register (PFCAEn)

This register, together with the PFCn and PFCEn register, specifies the alternative function of the pins.

The I/O direction of several alternative functions can be controlled directly by using the Pn\_m pin. For these alternative functions, the PIPCN.PIPCNm bit must be set to 1.

For all other alternative functions, the I/O direction is specified by using the PMn.PMnm bit.

After selecting an alternative function by the PFCn.PFCnm, PFCEn.PFCEnm, or PFCAEn.PFCAEnm bit, set the PMCn.PMCnm bit to 1.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PFCAE n15	PFCAE n14	PFCAE n13	PFCAE n12	PFCAE n11	PFCAE n10	PFCAE n9	PFCAE n8	PFCAE n7	PFCAE n6	PFCAE n5	PFCAE n4	PFCAE n3	PFCAE n2	PFCAE n1	PFCAE n0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit Position	Bit Name	Function
15 to 0	PFCAEn[15:0]	Specifies the alternative function of a pin.

Caution: Although some alternative functions are assigned to multiple pins in this LSI, only one of the pins can be used for each alternative function. Do not set the same alternative function in multiple pins. For example, when the a/b/c pin is used as the b pin, the b/d/e pin cannot be used as the b pin. Before using the b/d/e pin, set the pin as an alternative function other than b.

Note: For the allocation of the alternative functions, see the table of the pin functions of each port.

### 54.3.12 Port Input Buffer Control Register (PIBCn)

In input port mode ( $PMn.PMCnm = 0$  and  $PMn.PMnm = 1$ ), this register enables or disables the input buffer for the  $Pn\_m$  pin.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIBCn 15	PIBCn 14	PIBCn 13	PIBCn 12	PIBCn 11	PIBCn 10	PIBCn 9	PIBCn 8	PIBCn 7	PIBCn 6	PIBCn 5	PIBCn 4	PIBCn 3	PIBCn 2	PIBCn 1	PIBCn 0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit Position	Bit Name	Function
15 to 0	PIBCn[15:0]	These bits enable or disable the input buffer. 0: Input buffer disabled 1: Input buffer enabled

Note: When the input buffer is disabled, through current does not flow even when the pin level is Hi-Z. Thus the pin does not need to be fixed to a high or low level externally (except for P2\_12, P2\_15, P3\_1, P3\_3, P3\_9, P5\_09, P6\_2, P6\_4, P7\_8, P8\_02, P8\_7, and P9\_1).

Caution: When the output buffer is enabled and the PBDCn.PBDCnm bit is 1, the input buffer is enabled regardless of this register setting.

### 54.3.13 Port Bidirection Control Register (PBDCn/JPBDC0)

This register enables or disables the input buffer while the output buffer is enabled. When the input buffer is enabled while the output buffer is enabled, the bidirectional mode is entered, allowing the level of the  $Pn\_m$  pin to always be read via the PPRn.PPRnm bit.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PBDCn 15	PBDCn 14	PBDCn 13	PBDCn 12	PBDCn 11	PBDCn 10	PBDCn 9	PBDCn 8	PBDCn 7	PBDCn 6	PBDCn 5	PBDCn 4	PBDCn 3	PBDCn 2	PBDCn 1	PBDCn 0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit Position	Bit Name	Function
15 to 0	PBDCn[15:0]	Enables/disables bidirectional mode of the corresponding pin: 0: Bidirectional mode disabled 1: Bidirectional mode enabled

Caution: When using the  $Pn\_m$  bit as the output of an alternative function ( $PMn.PMCnm = 1$ ,  $PMn.PMnm = 0$ ), the level of the  $Pn\_m$  pin can be read at the PPRn.PPRnm bit by setting the PBDCn.PBDCnm bit to 1. Note, however, that in this case, the level of the  $Pn\_m$  pin will be conveyed to the input of the alternative function that the  $Pn\_m$  pin is being used as.

Note: The JPBDC0[1:0] bits are control bits in the JTAG port bidirection control register (JPBDC0).

### 54.3.14 Port IP Control Register (PIPCn)

This register is used to specify whether the I/O direction of the Pn\_m pin is controlled by the PMn.PMnm bit or by the alternative function in alternative mode (the PMCn.PMCnm bit is 1).

When using an alternative function shown in Table 54.7, set the PIPCn.PIPCnm bit to 0. When using an alternative function that is not shown in Table 54.7, set the PIPCn.PIPCnm bit to 1. When the PIPCn.PIPCnm bit is set to 1, the alternative function controls the I/O direction and the PMn.PMnm bit setting becomes invalid.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIPCn 15	PIPCn 14	PIPCn 13	PIPCn 12	PIPCn 11	PIPCn 10	PIPCn 9	PIPCn 8	PIPCn 7	PIPCn 6	PIPCn 5	PIPCn 4	PIPCn 3	PIPCn 2	PIPCn 1	PIPCn 0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit Position	Bit Name	Function
15 to 0	PIPCn[15:0]	These bits specify how to control the I/O direction in alternative mode. 0: The I/O direction is controlled according to the value of the PMn.PMnm bit (software I/O control). 1: The I/O direction is controlled by the alternative function (direct I/O control).

**Table 54.7 Alternative Functions that PIPCn.PIPCnm Bit Should be Set to 0**

Classification	Pin Name	Remark
Multi-function timer pulse unit 2	TIOC0A	Set the PIPCn.PIPCnm bit to 0 and select input or output by using the PMn.PMnm bit.
	TIOC0B	
	TIOC0C	
	TIOC0D	
	TIOC1A	
	TIOC1B	
	TIOC2A	
	TIOC2B	
	TIOC3A	
	TIOC3B	
	TIOC3C	
	TIOC3D	
	TIOC4A	
	TIOC4B	
	TIOC4C	
	TIOC4D	
LVDS output interface	TXOUT0P	Set the PIPCn.PIPCnm bit to 0 and the PMn.PMnm bit to 1.
	TXOUT0M	
	TXOUT1P	
	TXOUT1M	
	TXOUT2P	
	TXOUT2M	
	TXCLKOUTP	
	TXCLKOUTM	
Serial sound interface	SSITxD0	Set the PIPCn.PIPCnm and PMn.PMnm bits to 0.
	SSITxD1	
	SSITxD3	
	SSITxD4	

### 54.3.15 Serial Sound Interface Noise Canceler Control Register (SNCR)

This register controls the noise canceler in the input route from the LSI pin to a serial sound interface. Each bit can be set only when slave mode is selected for the corresponding channel of the serial sound interface. The bit should be used as it is the initial value (H'00000000) when master mode is selected for the corresponding channel of the serial sound interface.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	ETSEL	SSI5 NCE	SSI4 NCE	SSI3 NCE	SSI2 NCE	SSI1 NCE	SSI0 NCE
R/W:	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit Position	Bit Name	Function
6	ETSEL	Enables or disables the Ethernet controller and EthernetAVB with respect to the pins for MII supported in the Ethernet. 0: The Ethernet controller is enabled and the EthernetAVB is disabled. 1: The EthernetAVB is enabled and the Ethernet controller is disabled.
5	SSI5NCE	Serial Sound Interface Channel 5 Noise Canceler Enable Enables or disables the noise canceler of SSISCK5, SSIWS5, and SSIRxD5. 0: Noise canceler is disabled. 1: Noise canceler is enabled.
4	SSI4NCE	Serial Sound Interface Channel 4 Noise Canceler Enable Enables or disables the noise canceler of SSISCK4, SSIWS4, and SSIDATA4. 0: Noise canceler is disabled. 1: Noise canceler is enabled.
3	SSI3NCE	Serial Sound Interface Channel 3 Noise Canceler Enable Enables or disables the noise canceler of SSISCK3, SSIWS3, and SSIRxD3. 0: Noise canceler is disabled. 1: Noise canceler is enabled.
2	SSI2NCE	Serial Sound Interface Channel 2 Noise Canceler Enable Enables or disables the noise canceler of SSISCK2, SSIWS2, and SSIDATA2. 0: Noise canceler is disabled. 1: Noise canceler is enabled.
1	SSI1NCE	Serial Sound Interface Channel 1 Noise Canceler Enable Enables or disables the noise canceler of SSISCK1, SSIWS1, and SSIRxD1. 0: Noise canceler is disabled. 1: Noise canceler is enabled.
0	SSI0NCE	Serial Sound Interface Channel 0 Noise Canceler Enable Enables or disables the noise canceler of SSISCK0, SSIWS0, and SSIRxD0. 0: Noise canceler is disabled. 1: Noise canceler is enabled.

## 54.4 Port Function

**Table 54.8 Number of Pins and Function of Ports**

Port	Pin Name	Size (Number of Pins)	Input/Output	Note
JTAG Port 0	JP0_0 to 1	2 bits	input	
Port 0	P0_0 to 5	6 bits	input	
Port 1	P1_0 to 15	16 bits	input/open-drain output	Output of only the P1_0 to P1_7 pins is open-drain output.
Port 2	P2_0 to 15	16 bits	input/output	
Port 3	P3_0 to 15	16 bits	input/output	
Port 4	P4_0 to 15	16 bits	input/output	
Port 5	P5_0 to 10	11 bits	input/output	
Port 6	P6_0 to 15	16 bits	input/output	
Port 7	P7_0 to 15	16 bits	input/output	
Port 8	P8_0 to 15	16 bits	input/output	
Port 9	P9_0 to 7	8 bits	input/output	
Port 10	P10_0 to 15	16 bits	input/output	Only in 324-pin package
Port 11	P11_0 to 15	16 bits	input/output	Only in 324-pin package

## 54.5 JTAG Port 0 (JP0)

**Table 54.9 Pin Function (JP0)**

Port Mode		Alternative Mode	
Input	Output	Input	Output
JP0_0		TDI	
JP0_1			TDO

Note: Only input in port mode.

**Table 54.10 Control Registers (JP0)**

Register	Register Size	Valid Bit		Offset Address <JPORn_base>	Initial Value
		Location	R/W		
JPPR0	16	1, 0	R	H'0020	H'0000
JPMC0	16	1, 0	R/W	H'0040	H'FFFF
JPMCSR0	32	17, 6, 1, 0	W, R/W	H'0090	H'0000_FFFF
JPIBC0	16	1, 0	R/W	H'0400	H'0000

Note: When a control register is read, the initial value is read from the fields where no valid bit is allocated. When writing to these fields, write the initial value.

## 54.6 Port 0 (P0)

Table 54.11 Pin Function (P0)

Port Mode		Alternative Mode	
Input	Output	Input	Output
P0_0			
P0_1			
P0_2			
P0_3			
P0_4		RTC_X3	
P0_5			RTC_X4

Note 1. Only input in port mode. P0\_0 to P0\_3 pins have no alternative mode.

Note 2. P0\_0, P0\_1, P0\_2, P0\_3 pins are MD\_BOOT0, MD\_BOOT1, MD\_CLK, MD\_CLKS functions in the state of  $\overline{RES} = L$ .

Table 54.12 Control Registers (P0)

Register	Register Size	Valid Bit		Offset Address <PORTn_base>	Initial Value
		Location	R/W		
PPR0	16	5 to 0	R	H'0200	H'0000
PMC0	16	5, 4	R/W	H'0400	H'0000
PMCSR0	32	21, 20, 5, 4	W, R/W	H'0900	H'0000_0000
PIBC0	16	5 to 0	R/W	H'4000	H'0000

Note: When a control register is read, the initial value is read from the fields where no valid bit is allocated. When writing to these fields, write the initial value.

## 54.7 Port 1 (P1)

Table 54.13 Pin Function (P1)

Port Mode		Alternative Mode															
		1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		8th Alternative	
Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output
P1_0	P1_0	R1IC0SCL		DV0_DATA16		TCLKA		IRQ0		VIO_VD		DV0_VSYNC					
P1_1	P1_1	R1IC0SDA		DV0_DATA17		TCLKC		IRQ1		VIO_HD		DV0_HSYNC					
P1_2	P1_2	R1IC1SCL		DV0_DATA18		FRB		IRQ2						LCD1_EXTCLK			
P1_3	P1_3	R1IC1SDA		DV0_DATA19		ET_COL		IRQ3		$\overline{\text{ADTRG}}$							
P1_4	P1_4	R1IC2SCL		DV0_CLK		CAN1RX		IRQ4						CAN_CLK			
P1_5	P1_5	R1IC2SDA		DV1_CLK		CAN4RX		IRQ5		VIO_CLK				LCD1_EXTCLK			
P1_6	P1_6	R1IC3SCL		DV1_VSYNC		IERxD		IRQ6		VIO_D12		DV0_DATA12					
P1_7	P1_7	R1IC3SDA		DV1_HSYNC		RLIN30RX		IRQ7		VIO_D13		DV0_DATA13					
P1_8		AN0				IRQ2		DREQ0		VIO_D14		DV0_DATA14					
P1_9		AN1				IRQ3				VIO_D15		DV0_DATA15					
P1_10		AN2				IRQ4		TCLKB									
P1_11		AN3				IRQ5		TCLKD									
P1_12		AN4		DV0_VSYNC				VIO_FLD									
P1_13		AN5		DV0_HSYNC				$\overline{\text{WAIT}}$									
P1_14		AN6						ET_COL									
P1_15		AN7						AVB_CAPTURE									

Note: P1\_0 to P1\_7 pins are input pins with open-drain output. P1\_8 to P1\_15 pins are only input.



**Table 54.14 Control Registers (P1)**

Register	Register Size	Valid Bit		Offset Address <PORTn_base>	Initial Value
		Location	R/W		
P1	16	7 to 0	R/W	H'0004	H'0000
PSR1	32	23 to 16, 7 to 0	W, R/W	H'0104	H'0000_0000
PPR1	16	15 to 0	R	H'0204	H'0000
PM1	16	7 to 0	R/W	H'0304	H'FFFF*2
PMC1	16	15 to 0	R/W	H'0404	H'0000
PFC1	16	15 to 0	R/W	H'0504	H'0000
PFCE1	16	15 to 0	R/W	H'0604	H'0000
PNOT1	16	7 to 0	W	H'0704	H'0000
PMSR1	32	23 to 16, 7 to 0	W, R/W	H'0804	H'0000_FFFF
PMCSR1	32	23 to 16, 7 to 0	W, R/W	H'0904	H'0000_0000
PFCAE1	16	15 to 0	R/W	H'0A04	H'0000
PIBC1	16	15 to 0	R/W	H'4004	H'0000
PBDC1	16	15 to 0	R/W	H'4104	H'0000
PIPC1	16	7 to 0	R/W	H'4204	H'FF00*2

Note 1. When a control register is read, the initial value is read from the fields where no valid bit is allocated. When writing to these fields, write the initial value.

Note 2. The values of bits 8 to 15 of the PM1 and PIPC1 registers are fixed. When read, all of these bits are always read as 1.

## 54.8 Port 2 (P2)

Table 54.15 Pin Function (P2)

Port Mode		Alternative Mode															
		1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		8th Alternative	
Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output
P2_0	P2_0	D16		ET_TXCLK		DV0_DATA0		SPBIO00_1		MLB_CLK		IRQ5		VIO_D0		LCD0_DATA16	
P2_1	P2_1	D17			ET_TXER	DV0_DATA1		SPBIO10_1		MLB_DAT		TIOC2A		VIO_D1		LCD0_DATA17	
P2_2	P2_2	D18			ET_TXEN	DV0_DATA2		SPBIO20_1		MLB_SIG		TIOC2B		VIO_D2		LCD0_DATA18	
P2_3	P2_3	D19		ET_CRS		DV0_DATA3		SPBIO30_1		IERxD		CTS1		VIO_D3		LCD0_DATA19	
P2_4	P2_4	D20			ET_TXD0	DV0_DATA4		SSISCK5			SPBCLK_1	SCK1		VIO_D4		LCD0_DATA20	
P2_5	P2_5	D21			ET_TXD1	DV0_DATA5		SSIWS5			SPBSSL_1		TxD1	VIO_D5		LCD0_DATA21	
P2_6	P2_6	D22			ET_TXD2	DV0_DATA6		SSIRxD5				RxD1		VIO_D6		LCD0_DATA22	
P2_7	P2_7	D23			ET_TXD3	DV0_DATA7			SSITxD5		IETxD	RTS1		VIO_D7		LCD0_DATA23	
P2_8	P2_8	D24		ET_RXD0		DV0_DATA8		SSISCK0			LCD0_TCON6		LCD1_DATA8	VIO_D8		RSPCK4	
P2_9	P2_9	D25		ET_RXD1		DV0_DATA9		SSIWS0		RLIN30_RX			LCD1_DATA9	VIO_D9		SSL40	
P2_10	P2_10	D26		ET_RXD2		DV0_DATA10		SSIRxD0			RLIN30_TX		LCD1_DATA10	VIO_D10		MOSI4	
P2_11	P2_11	D27		ET_RXD3		DV0_DATA11			SSITxD0	TIOC1A			LCD1_DATA11	VIO_D11		MISO4	
P2_12	P2_12	D28		RSPCK0		DV0_DATA12		SPBIO01_0		CAN3RX		IRQ6			LCD1_DATA12	TIOC1B	
P2_13	P2_13	D29		SSL00		DV0_DATA13		SPBIO11_0			CAN3TX	SCK0			LCD1_DATA13	IRQ7	
P2_14	P2_14	D30		MOSI0		DV0_DATA14		SPBIO21_0		CAN4RX			TxD0		LCD1_DATA14	IRQ0	
P2_15	P2_15	D31		MISO0		DV0_DATA15		SPBIO31_0		CAN_CLK		RxD0			LCD1_DATA15	IRQ1	

Table 54.16 Control Registers (P2)

Register	Register Size	Valid Bit		Offset Address <PORTn_base>	Initial Value
		Location	R/W		
P2	16	15 to 0	R/W	H'0008	H'0000
PSR2	32	31 to 0	W, R/W	H'0108	H'0000_0000
PPR2	16	15 to 0	R	H'0208	H'0000
PM2	16	15 to 0	R/W	H'0308	H'FFFF
PMC2	16	15 to 0	R/W	H'0408	H'FFFF (In boot mode 1), H'0000 (Other than in boot mode 1)
PFC2	16	15 to 0	R/W	H'0508	H'0000
PFCE2	16	15 to 0	R/W	H'0608	H'0000
PNOT2	16	15 to 0	W	H'0708	H'0000
PMSR2	32	31 to 0	W, R/W	H'0808	H'0000_FFFF
PMCSR2	32	31 to 0	W, R/W	H'0908	H'0000_FFFF (In boot mode 1), H'0000_0000 (Other than in boot mode 1)
PFCAE2	16	15 to 0	R/W	H'0A08	H'0000
PIBC2	16	15 to 0	R/W	H'4008	H'0000
PBDC2	16	15 to 0	R/W	H'4108	H'FFFF (In boot mode 1), H'0000 (Other than in boot mode 1)
PIPC2	16	15 to 0	R/W	H'4208	H'FFFF (In boot mode 1), H'0000 (Other than in boot mode 1)

Note: For the pins whose functions are automatically selected in boot mode, see Table 8.3, Initial States by Areas in Boot Modes 0, 1, and 2 to 5.

## 54.9 Port 3 (P3)

Table 54.17 Pin Function (P3)

Port Mode		Alternative Mode															
		1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		8th Alternative	
Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output
P3_0	P3_0		LCD0_CLK	ET_TXCLK		IRQ2		SCK2		SCI_SCK1			TxD2		PWM2A	RSPCK3	
P3_1	P3_1		LCD0_TCON0		ET_TXER	IRQ6			TxD2		SCI_TXD1	AUDIO_CLK			PWM2B	SSL30	
P3_2	P3_2		LCD0_TCON1		ET_TXEN			RxD2		SCI_RXD1			TEND0		PWM2C	MOSI3	
P3_3	P3_3		LCD0_TCON2	ET_MDIO		IRQ4			$\overline{BS}$	SCI_CTS1/RTS1			DACK0		PWM2D	MISO3	
P3_4	P3_4		LCD0_TCON3	ET_RXCLK		SSISCK1			AUDIO_XOUT2	SCI_SCK0		TIOC3A		SCK3			
P3_5	P3_5		LCD0_TCON4	ET_RXER		SSIWS1			AUDIO_XOUT3		SCI_TXD0	TIOC3B			TxD3		
P3_6	P3_6		LCD0_TCON5	ET_RXDV		SSIRxD1				SCI_RXD0		TIOC3C		RxD3			
P3_7	P3_7		LCD0_TCON6				SSITxD1	LCD1_EXTCLK		SCI_CTS0/RTS0		TIOC3D			$\overline{CS}$ 1		WDTOV_F
P3_8	P3_8		LCD0_DATA0			NAF0					TRACE_DATA0	TIOC4A		SD_CD_1		MMC_CD	
P3_9	P3_9		LCD0_DATA1			NAF1					TRACE_DATA1	TIOC4B		SD_WP_1		IRQ6	
P3_10	P3_10		LCD0_DATA2			NAF2					TRACE_DATA2	TIOC4C		SD_D1_1		MMC_D1	
P3_11	P3_11		LCD0_DATA3			NAF3					TRACE_DATA3	TIOC4D		SD_D0_1		MMC_D0	
P3_12	P3_12		LCD0_DATA4			NAF4									SD_CLK_1		MMC_CLK
P3_13	P3_13		LCD0_DATA5			NAF5			AUDIO_XOUT					SD_CMD_1		MMC_CMD	
P3_14	P3_14		LCD0_DATA6			NAF6					TRACE_CLK			SD_D3_1		MMC_D3	
P3_15	P3_15		LCD0_DATA7			NAF7					TRACE_CTL			SD_D2_1		MMC_D2	

Table 54.18 Control Registers (P3)

Register	Register Size	Valid Bit		Offset Address <PORTn_base>	Initial Value
		Location	R/W		
P3	16	15 to 0	R/W	H'000C	H'0000
PSR3	32	31 to 0	W, R/W	H'010C	H'0000_0000
PPR3	16	15 to 0	R	H'020C	H'0000
PM3	16	15 to 0	R/W	H'030C	H'FFFF
PMC3	16	15 to 0	R/W	H'040C	H'0000
PFC3	16	15 to 0	R/W	H'050C	H'0000
PFCE3	16	15 to 0	R/W	H'060C	H'0000
PNOT3	16	15 to 0	W	H'070C	H'0000
PMSR3	32	31 to 0	W, R/W	H'080C	H'0000_FFFF
PMCSR3	32	31 to 0	W, R/W	H'090C	H'0000_0000
PFCAE3	16	15 to 0	R/W	H'0A0C	H'0000
PIBC3	16	15 to 0	R/W	H'400C	H'0000
PBDC3	16	15 to 0	R/W	H'410C	H'0000
PIPC3	16	15 to 0	R/W	H'420C	H'0000

## 54.10 Port 4 (P4)

Table 54.19 Pin Function (P4)

Port Mode		Alternative Mode																
		1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		8th Alternative		
Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	
P4_0	P4_0		LCD0_DATA8	TIOC0A			$\overline{\text{FRE}}$							RSPCK4		MMC_D4		
P4_1	P4_1		LCD0_DATA9	TIOC0B			FCLE			SCK2				SSL40		MMC_D5		
P4_2	P4_2		LCD0_DATA10	TIOC0C			FALE	CAN3RX				TxD2				MOSI4		
P4_3	P4_3		LCD0_DATA11	TIOC0D			$\overline{\text{FWE}}$		CAN3TX	RxD2					MISO4		MMC_D7	
P4_4	P4_4		LCD0_DATA12	RSPCK1		TIOC4A			PWM2E	SSISCK0					DV0_DATA12			
P4_5	P4_5		LCD0_DATA13	SSL10		TIOC4B			PWM2F	SSIWS0					DV0_DATA13			
P4_6	P4_6		LCD0_DATA14	MOSI1		TIOC4C			PWM2G	SSIRxD0					DV0_DATA14			
P4_7	P4_7		LCD0_DATA15	MISO1		TIOC4D			PWM2H		SSITxD0				DV0_DATA15			
P4_8	P4_8		LCD0_DATA16		LCD1_TCON3	SD_CD_0		MMC_CD		SSISCK5			CAN2TX	SCK0		IRQ0		
P4_9	P4_9		LCD0_DATA17		LCD1_TCON4	SD_WP_0				SSIWS5		CAN2RX			TxD0	IRQ1		
P4_10	P4_10		LCD0_DATA18		LCD1_TCON5	SD_D1_0		MMC_D1		SSIRxD5				RxD0		IRQ2		
P4_11	P4_11		LCD0_DATA19		LCD1_TCON6	SD_D0_0		MMC_D0		SSITxD5			CAN4TX	SCK1		IRQ3		
P4_12	P4_12		LCD0_DATA20		LCD1_CLK	SD_CLK_0		MMC_CLK		SPBIO01_1		SSISCK3			TxD1	IRQ4		
P4_13	P4_13		LCD0_DATA21		LCD1_TCON0	SD_CMD_0		MMC_CMD		SPBIO11_1		SSIWS3		RxD1		IRQ5		
P4_14	P4_14		LCD0_DATA22		LCD1_TCON1	SD_D3_0		MMC_D3		SPBIO21_1	SSIRxD3				TxD2	IRQ6		
P4_15	P4_15		LCD0_DATA23		LCD1_TCON2	SD_D2_0		MMC_D2		SPBIO31_1		SSITxD3	RxD2			IRQ7		

Table 54.20 Control Registers (P4)

Register	Register Size	Valid Bit		Offset Address <PORTn_base>	Initial Value
		Location	R/W		
P4	16	15 to 0	R/W	H'0010	H'0000
PSR4	32	31 to 0	W, R/W	H'0110	H'0000_0000
PPR4	16	15 to 0	R	H'0210	H'0000
PM4	16	15 to 0	R/W	H'0310	H'FFFF
PMC4	16	15 to 0	R/W	H'0410	H'0000
PFC4	16	15 to 0	R/W	H'0510	H'0000
PFCE4	16	15 to 0	R/W	H'0610	H'0000
PNOT4	16	15 to 0	W	H'0710	H'0000
PMSR4	32	31 to 0	W, R/W	H'0810	H'0000_FFFF
PMCSR4	32	31 to 0	W, R/W	H'0910	H'0000_0000
PFCAE4	16	15 to 0	R/W	H'0A10	H'0000
PIBC4	16	15 to 0	R/W	H'4010	H'0000
PBDC4	16	15 to 0	R/W	H'4110	H'0000
PIPC4	16	15 to 0	R/W	H'4210	H'0000

## 54.11 Port 5 (P5)

Table 54.21 Pin Function (P5)

Port Mode		Alternative Mode															
		1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		8th Alternative	
Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output
P5_0	P5_0		TXCLKOUTP		LCD1_DATA0		LCD0_DATA16	DV1_DATA0			TxD4		TIOC0A				RSPCK3
P5_1	P5_1		TXCLKOUTM		LCD1_DATA1		LCD0_DATA17	DV1_DATA1		RxD4			TIOC0B				SSL30
P5_2	P5_2		TXOUT2P		LCD1_DATA2		LCD0_DATA18	DV1_DATA2			SCK3		TIOC1B				MOSI3
P5_3	P5_3		TXOUT2M		LCD1_DATA3		LCD0_DATA19	DV1_DATA3			TxD3		TIOC3C				MISO3
P5_4	P5_4		TXOUT1P		LCD1_DATA4		LCD0_DATA20	DV1_DATA4		RxD3			TIOC3D				DV0_DATA12
P5_5	P5_5		TXOUT1M		LCD1_DATA5		LCD0_DATA21	DV1_DATA5			AUDIO_XOUT		TIOC0C		FCE		DV0_DATA13
P5_6	P5_6		TXOUT0P		LCD1_DATA6		LCD0_DATA22	DV1_DATA6			TxD6	IRQ6		SPDIF_IN			DV0_DATA14
P5_7	P5_7		TXOUT0M		LCD1_DATA7		LCD0_DATA23	DV1_DATA7		RxD6			TIOC0D			SPDIF_OUT	DV0_DATA15
P5_8	P5_8	LCD0_EXTCLK		IRQ0		DV1_CLK				DV0_CLK				$\overline{\text{CS2}}$			
P5_9	P5_9		$\overline{\text{WE2}}$ /DQMUL		ET_MDC	DV0_VSYNC		IRQ2		CAN1RX		IERxD				LCD1_DATA16	
P5_10	P5_10		$\overline{\text{WE3}}$ /DQMUU/AH			DV0_HSYNC				CAN1TX		IETxD				LCD1_DATA17	

Table 54.22 Control Registers (P5)

Register	Register Size	Valid Bit		Offset Address <PORTn_base>	Initial Value
		Location	R/W		
P5	16	10 to 0	R/W	H'0014	H'0000
PSR5	32	26 to 16, 10 to 0	W, R/W	H'0114	H'0000_0000
PPR5	16	10 to 0	R	H'0214	H'0000
PM5	16	10 to 0	R/W	H'0314	H'FFFF
PMC5	16	10 to 0	R/W	H'0414	H'0000
PFC5	16	10 to 0	R/W	H'0514	H'0000
PFCE5	16	10 to 0	R/W	H'0614	H'0000
PNOT5	16	10 to 0	W	H'0714	H'0000
PMSR5	32	26 to 16, 10 to 0	W, R/W	H'0814	H'0000_FFFF
PMCSR5	32	26 to 16, 10 to 0	W, R/W	H'0914	H'0000_0000
PFCAE5	16	10 to 0	R/W	H'0A14	H'0000
PIBC5	16	10 to 0	R/W	H'4014	H'0000
PBDC5	16	10 to 0	R/W	H'4114	H'0000
PIPC5	16	10 to 0	R/W	H'4214	H'0000

Note: When a control register is read, the initial value is read from the fields where no valid bit is allocated. When writing to these fields, write the initial value.

## 54.12 Port 6 (P6)

Table 54.23 Pin Function (P6)

Port Mode		Alternative Mode															
		1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		8th Alternative	
Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output
P6_0	P6_0	D0			LCD1_DATA8	RLIN30 RX		DV0_CLK		TIOC1A		IRQ5		RxD3		DV0_DATA16	
P6_1	P6_1	D1			LCD1_DATA9		RLIN30 TX	IRQ4		TIOC1B		SSIDATA4			TxD3	DV0_DATA17	
P6_2	P6_2	D2			LCD1_DATA10	RLIN31 RX		IRQ7		TCLKA		TIOC2A		RxD2		DV0_DATA18	
P6_3	P6_3	D3			LCD1_DATA11		RLIN31 TX	IRQ2		CTS5		TIOC2B			TxD2	DV0_DATA19	
P6_4	P6_4	D4			LCD1_DATA12	CAN2 RX		IRQ3		RTS5				RSPCK1		DV0_DATA20	
P6_5	P6_5	D5			LCD1_DATA13		CAN2 TX			SCK5				SSL10		DV0_DATA21	
P6_6	P6_6	D6			LCD1_DATA14				LCD0_TCON5		TxD5			MOSI1		DV0_DATA22	
P6_7	P6_7	D7			LCD1_DATA15				LCD0_TCON6	RxD5				MISO1		DV0_DATA23	
P6_8	P6_8	D8	DV0_DATA12					CAN_CLK		SCK0			LCD0_DATA0			IRQ0	
P6_9	P6_9	D9	DV0_DATA13								TxD0		LCD0_DATA1			IRQ1	
P6_10	P6_10	D10	DV0_DATA14						LCD0_TCON5	RxD0			LCD0_DATA2			IRQ2	
P6_11	P6_11	D11	DV0_DATA15						LCD0_TCON6	SCK1			LCD0_DATA3			IRQ3	
P6_12	P6_12	D12	DV0_DATA20								TxD1		LCD0_DATA4			IRQ4	
P6_13	P6_13	D13	DV0_DATA21					SCK6		RxD1			LCD0_DATA5			IRQ5	
P6_14	P6_14	D14	DV0_DATA22						TxD6				LCD0_DATA6			IRQ6	
P6_15	P6_15	D15	DV0_DATA23					RxD6					LCD0_DATA7			IRQ7	

Table 54.24 Control Registers (P6)

Register	Register Size	Valid Bit		Offset Address <PORTn_base>	Initial Value
		Location	R/W		
P6	16	15 to 0	R/W	H'0018	H'0000
PSR6	32	31 to 0	W, R/W	H'0118	H'0000_0000
PPR6	16	15 to 0	R	H'0218	H'0000
PM6	16	15 to 0	R/W	H'0318	H'FFFF
PMC6	16	15 to 0	R/W	H'0418	H'FFFF (In boot mode 0, 1), H'0000 (Other than in boot mode 0 or 1)
PFC6	16	15 to 0	R/W	H'0518	H'0000
PFCE6	16	15 to 0	R/W	H'0618	H'0000
PNOT6	16	15 to 0	W	H'0718	H'0000
PMSR6	32	31 to 0	W, R/W	H'0818	H'0000_FFFF
PMCSR6	32	31 to 0	W, R/W	H'0918	H'0000_FFFF (In boot mode 0 or 1), H'0000_0000 (Other than in boot mode 0 or 1)
PFCAE6	16	15 to 0	R/W	H'0A18	H'0000
PIBC6	16	15 to 0	R/W	H'4018	H'0000
PBDC6	16	15 to 0	R/W	H'4118	H'FFFF (In boot mode 0 or 1), H'0000 (Other than in boot mode 0 or 1)
PIPC6	16	15 to 0	R/W	H'4218	H'FFFF (In boot mode 0 or 1), H'0000 (Other than in boot mode 0 or 1)

Note: For the pins whose functions are automatically selected in boot mode, see Table 8.3, Initial States by Areas in Boot Modes 0, 1, and 2 to 5.

## 54.13 Port 7 (P7)

Table 54.25 Pin Function (P7)

Port Mode		Alternative Mode															
		1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		8th Alternative	
Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output
P7_0	P7_0		CS0	DV0_DATA16			ET_MDC	SCK4			RLIN30 TX			TIOC0A			
P7_1	P7_1		CS3	DV0_DATA17		ET_TXCLK			TxD4	DV0_CLK		SSISCK1		TIOC0B			
P7_2	P7_2		RAS	DV0_DATA18			ET_TXER	RxD4		CAN2RX		SSIWS1		TIOC0C			
P7_3	P7_3		CAS	DV0_DATA19			ET_TXEN	SCK7			CAN2TX	SSIRxD1		TIOC0D			
P7_4	P7_4		CKE	DV0_DATA20			ET_TXD0		TxD7				SSITxD1	TIOC1A			
P7_5	P7_5		RD/W $\bar{R}$	DV0_DATA21			ET_TXD1	RxD7				SSISCK2		TIOC1B			
P7_6	P7_6		WE $\bar{0}$ /DQMLL	DV0_DATA22			ET_TXD2	CTS7				SSIWS2		TIOC2A			
P7_7	P7_7		WE $\bar{1}$ /DQMLU	DV0_DATA23			ET_TXD3	RTS7				SSIDATA2		TIOC2B			
P7_8	P7_8		RD	SSISCK3				CAN0RX						TIOC3A		IRQ1	
P7_9	P7_9		A1	SSIWS3		ET_RXD0			CAN0TX					TIOC3B		IRQ0	
P7_10	P7_10		A2	SSIRxD3		ET_RXD1			CAN1TX					TIOC3C		IRQ2	
P7_11	P7_11		A3		SSITxD3	ET_RXD2		CAN1RX						TIOC3D		IRQ3	
P7_12	P7_12		A4	SSISCK4		ET_RXD3								TIOC4A		IRQ4	
P7_13	P7_13		A5	SSIWS4		ET_MDIO								TIOC4B		IRQ5	
P7_14	P7_14		A6	SSIDATA4		ET_CRS								TIOC4C		IRQ6	
P7_15	P7_15		A7	RSPCK0		ET_RXCLK		CTS5			SCL_TXD0			TIOC4D			

Note: P7\_0 pin is MD\_BOOT2 function in the state of RES = L.

**Table 54.26 Control Registers (P7)**

Register	Register Size	Valid Bit		Offset Address <PORTn_base>	Initial Value
		Location	R/W		
P7	16	15 to 0	R/W	H'001C	H'0000
PSR7	32	31 to 0	W, R/W	H'011C	H'0000_0000
PPR7	16	15 to 0	R	H'021C	H'0000
PM7	16	15 to 0	R/W	H'031C	H'FFFF
PMC7	16	15 to 0	R/W	H'041C	H'FF01 (In boot mode 0), H'FD01 (In boot mode 1), H'0000 (Other than in boot mode 0 or 1)
PFC7	16	15 to 0	R/W	H'051C	H'0000
PFCE7	16	15 to 0	R/W	H'061C	H'0000
PNOT7	16	15 to 0	W	H'071C	H'0000
PMSR7	32	31 to 0	W, R/W	H'081C	H'0000_FFFF
PMCSR7	32	31 to 0	W, R/W	H'091C	H'0000_FF01 (In boot mode 0), H'0000_FD01 (In boot mode 1), H'0000_0000 (Other than in boot mode 0 or 1))
PFCAE7	16	15 to 0	R/W	H'0A1C	H'0000
PIBC7	16	15 to 0	R/W	H'401C	H'0000
PBDC7	16	15 to 0	R/W	H'411C	H'FF01 (In boot mode 0), H'FD01 (In boot mode 1), H'0000 (Other than in boot mode 0 or 1)
PIPC7	16	15 to 0	R/W	H'421C	H'FF01 (In boot mode 0), H'FD01 (In boot mode 1), H'0000 (Other than in boot mode 0 or 1)

Note: For the pins whose functions are automatically selected in boot mode, see Table 8.3, Initial States by Areas in Boot Modes 0, 1, and 2 to 5.



## 54.14 Port 8 (P8)

Table 54.27 Pin Function (P8)

Port Mode		Alternative Mode																
		1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		8th Alternative		
Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	
P8_0	P8_0		A8	SSL00		ET_RXER		SCK5		SCI_SCK0								
P8_1	P8_1		A9	MOSIO		ET_RXDV			TxD5	SCI_RXD0								
P8_2	P8_2		A10	MISO0		AVB_GPTP_EXTERN		RxD5		IRQ0								
P8_3	P8_3		A11	DV1_DATA0		RSPCK2		RTS5				IRQ1		SCK2				
P8_4	P8_4		A12	DV1_DATA1		SSL20						IERxD		RxD2				
P8_5	P8_5		A13	DV1_DATA2		MOSI2												
P8_6	P8_6		A14	DV1_DATA3		MISO2						IETxD		TxD2				
P8_7	P8_7		A15	DV1_DATA4			AUDIO_XOUT	IRQ5		ET_COL								
P8_8	P8_8		A16	DV1_DATA5		SPBIO00_1		SPDIF_IN		TIOC1A			PWM1A		TxD3		SSISCK5	
P8_9	P8_9		A17	DV1_DATA6		SPBIO10_1			SPDIF_OUT	TIOC1B			PWM1B		RxD3		SSIWS5	
P8_10	P8_10		A18	DV1_DATA7		SPBIO20_1		TIOC3A			CAN4_TX		PWM1C		SGOUT_0		SSITxD5	
P8_11	P8_11		A19			SPBIO30_1		TIOC3B		RxD5			PWM1D		SGOUT_1		DV0_CLK	
P8_12	P8_12		A20			SPBCLK_1		TIOC3C		SCK5			PWM1E		SGOUT_2		SSISCK4	
P8_13	P8_13		A21			SPBSSL_1		TIOC3D			TxD5		PWM1F		SGOUT_3		SSIWS4	
P8_14	P8_14		A22	SPBIO01_0		SPBIO00_1		TIOC2A		RSPCK2			PWM1G		TxD4		SSIDATA4	
P8_15	P8_15		A23	SPBIO11_0		SPBIO10_1		TIOC2B		SSL20			PWM1H		RxD4			

Table 54.28 Control Registers (P8)

Register	Register Size	Valid Bit		Offset Address		Initial Value
		Location	R/W	<PORTn_base>		
P8	16	15 to 0	R/W	H'0020		H'0000
PSR8	32	31 to 0	W, R/W	H'0120		H'0000_0000
PPR8	16	15 to 0	R	H'0220		H'0000
PM8	16	15 to 0	R/W	H'0320		H'FFFF
PMC8	16	15 to 0	R/W	H'0420		H'1FFF (In boot mode 0 or 1), H'0000 (Other than in boot mode 0 or 1)
PFC8	16	15 to 0	R/W	H'0520		H'0000
PFCE8	16	15 to 0	R/W	H'0620		H'0000
PNOT8	16	15 to 0	W	H'0720		H'0000
PMSR8	32	31 to 0	W, R/W	H'0820		H'0000_FFFF
PMCSR8	32	31 to 0	W, R/W	H'0920		H'0000_1FFF (In boot mode 0 or 1), H'0000_0000 (Other than in boot mode 0 or 1)
PFCAE8	16	15 to 0	R/W	H'0A20		H'0000
PIBC8	16	15 to 0	R/W	H'4020		H'0000
PBDC8	16	15 to 0	R/W	H'4120		H'1FFF (In boot mode 0 or 1), H'0000 (Other than in boot mode 0 or 1)
PIPC8	16	15 to 0	R/W	H'4220		H'1FFF (In boot mode 0 or 1), H'0000 (Other than in boot mode 0 or 1)

Note: For the pins whose functions are automatically selected in boot mode, see Table 8.3, Initial States by Areas in Boot Modes 0, 1, and 2 to 5.

## 54.15 Port 9 (P9)

Table 54.29 Pin Function (P9)

Port Mode		Alternative Mode															
		1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		8th Alternative	
Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output
P9_0	P9_0		A24	SPBIO21_0			CAN0 TX	TCLKC		MOSI2							
P9_1	P9_1		A25	SPBIO31_0		CAN0 RX		IRQ0		MISO2							
P9_2	P9_2		LCD1_DATA18	SPBCLK_0		RLIN30 TX		SCK1			A0						
P9_3	P9_3		LCD1_DATA19	SPBSSL_0				TxD1									
P9_4	P9_4		LCD1_DATA20	SPBIO00_0				RxD1									
P9_5	P9_5		LCD1_DATA21	SPBIO10_0		SSISCK2		CTS1		CS4							
P9_6	P9_6		LCD1_DATA22	SPBIO20_0		SSIWS2		RTS1		CS5							
P9_7	P9_7		LCD1_DATA23	SPBIO30_0		SSIDATA2		TIOC1A									

Table 54.30 Control Registers (P9)

Register	Register Size	Valid bit		Offset Address <PORTn_base>	Initial value
		Location	R/W		
P9	16	7 to 0	R/W	H'0024	H'0000
PSR9	32	23 to 16, 7 to 0	W, R/W	H'0124	H'0000_0000
PPR9	16	7 to 0	R	H'0224	H'0000
PM9	16	7 to 0	R/W	H'0324	H'FFFF
PMC9	16	7 to 0	R/W	H'0424	H'0000
PFC9	16	7 to 0	R/W	H'0524	H'0000
PFCE9	16	7 to 0	R/W	H'0624	H'0000
PNOT9	16	7 to 0	W	H'0724	H'0000
PMSR9	32	23 to 16, 7 to 0	W, R/W	H'0824	H'0000_FFFF
PMCSR9	32	23 to 16, 7 to 0	W, R/W	H'0924	H'0000_0000
PFCAE9	16	7 to 0	R/W	H'0A24	H'0000
PIBC9	16	7 to 0	R/W	H'4024	H'0000
PBDC9	16	7 to 0	R/W	H'4124	H'0000
PIPC9	16	7 to 0	R/W	H'4224	H'0000

Note: When a control register is read, the initial value is read from the fields where no valid bit is allocated. When writing to these fields, write the initial value.

## 54.16 Port 10 (P10)

Table 54.31 Pin Function (P10)

Port Mode		Alternative Mode															
		1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		8th Alternative	
Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output
P10_0	P10_0	DV0_CLK		TCLKA			PWM2A	ET_TXCLK			LCD0_DATA23	VIO_CLK					
P10_1	P10_1	DV0_VSYNC		TCLKB			PWM2B		ET_TXER		LCD0_DATA22	VIO_VD					
P10_2	P10_2	DV0_HSYNC		TCLKC			PWM2C		ET_TXEN		LCD0_DATA21	VIO_HD					
P10_3	P10_3			TCLKD			PWM2D	ET_CRS			LCD0_DATA20	VIO_FLD					
P10_4	P10_4	DV0_DATA0		TIOC0A			PWM2E		ET_TXD0		LCD0_DATA19	VIO_D0					
P10_5	P10_5	DV0_DATA1		TIOC0B			PWM2F		ET_TXD1		LCD0_DATA18	VIO_D1					
P10_6	P10_6	DV0_DATA2		TIOC0C			PWM2G		ET_TXD2		LCD0_DATA17	VIO_D2					
P10_7	P10_7	DV0_DATA3		TIOC0D			PWM2H		ET_TXD3		LCD0_DATA16	VIO_D3					
P10_8	P10_8	DV0_DATA4		TIOC1A				ET_RXD0			LCD0_DATA15	VIO_D4					
P10_9	P10_9	DV0_DATA5		TIOC1B				ET_RXD1			LCD0_DATA14	VIO_D5					
P10_10	P10_10	DV0_DATA6		TIOC2A				ET_RXD2			LCD0_DATA13	VIO_D6					
P10_11	P10_11	DV0_DATA7		TIOC2B				ET_RXD3			LCD0_DATA12	VIO_D7					
P10_12	P10_12	DV0_DATA8		SSISCK1				RSPCK0			LCD0_DATA11	VIO_D8					
P10_13	P10_13	DV0_DATA9		SSIWS1				SSL00			LCD0_DATA10	VIO_D9					
P10_14	P10_14	DV0_DATA10		SSIRxD1				MOSI0			LCD0_DATA9	VIO_D10					
P10_15	P10_15	DV0_DATA11			SSITxD1			MISO0			LCD0_DATA8	VIO_D11					

Table 54.32 Control Registers (P10)

Register	Register Size	Valid Bit		Offset Address <PORTn_base>	Initial Value
		Location	R/W		
P10	16	15 to 0	R/W	H'0028	H'0000
PSR10	32	31 to 0	W, R/W	H'0128	H'0000_0000
PPR10	16	15 to 0	R	H'0228	H'0000
PM10	16	15 to 0	R/W	H'0328	H'FFFF
PMC10	16	15 to 0	R/W	H'0428	H'0000
PFC10	16	15 to 0	R/W	H'0528	H'0000
PFCE10	16	15 to 0	R/W	H'0628	H'0000
PNOT10	16	15 to 0	W	H'0728	H'0000
PMSR10	32	31 to 0	W, R/W	H'0828	H'0000_FFFF
PMCSR10	32	31 to 0	W, R/W	H'0928	H'0000_0000
PFCAE10	16	15 to 0	R/W	H'0A28	H'0000
PIBC10	16	15 to 0	R/W	H'4028	H'0000
PBDC10	16	15 to 0	R/W	H'4128	H'0000
PIPC10	16	15 to 0	R/W	H'4228	H'0000

## 54.17 Port 11 (P11)

Table 54.33 Pin Function (P11)

Port Mode		Alternative Mode															
		1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		8th Alternative	
Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output	Input	Output
P11_0	P11_0	DV0_DATA12		TIOC4A				SCK6			LCD0_DATA7	VIO_D12					
P11_1	P11_1	DV0_DATA13		TIOC4B					TxD6		LCD0_DATA6	VIO_D13					
P11_2	P11_2	DV0_DATA14		TIOC4C				RxD6			LCD0_DATA5	VIO_D14					
P11_3	P11_3	DV0_DATA15		TIOC4D							LCD0_DATA4	VIO_D15					
P11_4	P11_4	DV0_DATA16		SD_CD_0		SSISCK4		MMC_CD			LCD0_DATA3						
P11_5	P11_5	DV0_DATA17		SD_WP_0		SSISCK4					LCD0_DATA2						
P11_6	P11_6	DV0_DATA18		SD_D1_0		SSIDATA4		MMC_D1			LCD0_DATA1						
P11_7	P11_7	DV0_DATA19		SD_D0_0		CTS5		MMC_D0			LCD0_DATA0						
P11_8	P11_8	DV0_DATA20			SD_CLK_0	RTS5			MMC_CLK		LCD0_TCON6						
P11_9	P11_9	DV0_DATA21		SD_CMD_0		SCK5		MMC_CMD			LCD0_TCON5						
P11_10	P11_10	DV0_DATA22		SD_D3_0			TxD5	MMC_D3			LCD0_TCON4						
P11_11	P11_11	DV0_DATA23		SD_D2_0		RxD5		MMC_D2			LCD0_TCON3						
P11_12	P11_12	CAN1_RX		RSPCK1		IRQ3		MMC_D4			LCD0_TCON2						
P11_13	P11_13		CAN1_TX	SSL10			LCD0_TCON4	MMC_D5			LCD0_TCON1						
P11_14	P11_14	SPDIF_IN		MOSI1			LCD0_TCON5	MMC_D6			LCD0_TCON0						
P11_15	P11_15		SPDIF_OUT	MISO1		IRQ1		MMC_D7			LCD0_CLK						

Table 54.34 Control Registers (P11)

Register	Register Size	Valid Bit		Offset Address <PORTn_base>	Initial Value
		Location	R/W		
P11	16	15 to 0	R/W	H'002C	H'0000
PSR11	32	31 to 0	W, R/W	H'012C	H'0000_0000
PPR11	16	15 to 0	R	H'022C	H'0000
PM11	16	15 to 0	R/W	H'032C	H'FFFF
PMC11	16	15 to 0	R/W	H'042C	H'0000
PFC11	16	15 to 0	R/W	H'052C	H'0000
PFCE11	16	15 to 0	R/W	H'062C	H'0000
PNOT11	16	15 to 0	W	H'072C	H'0000
PMSR11	32	31 to 0	W, R/W	H'082C	H'0000_FFFF
PMCSR11	32	31 to 0	W, R/W	H'092C	H'0000_0000
PFCAE11	16	15 to 0	R/W	H'0A2C	H'0000
PIBC11	16	15 to 0	R/W	H'402C	H'0000
PBDC11	16	15 to 0	R/W	H'412C	H'0000
PIPC11	16	15 to 0	R/W	H'422C	H'0000

### 54.18 Port Control Logical Diagram

The following shows a logical diagram of the port control function.

Note: This figure shows the logic for reference, not the circuit.

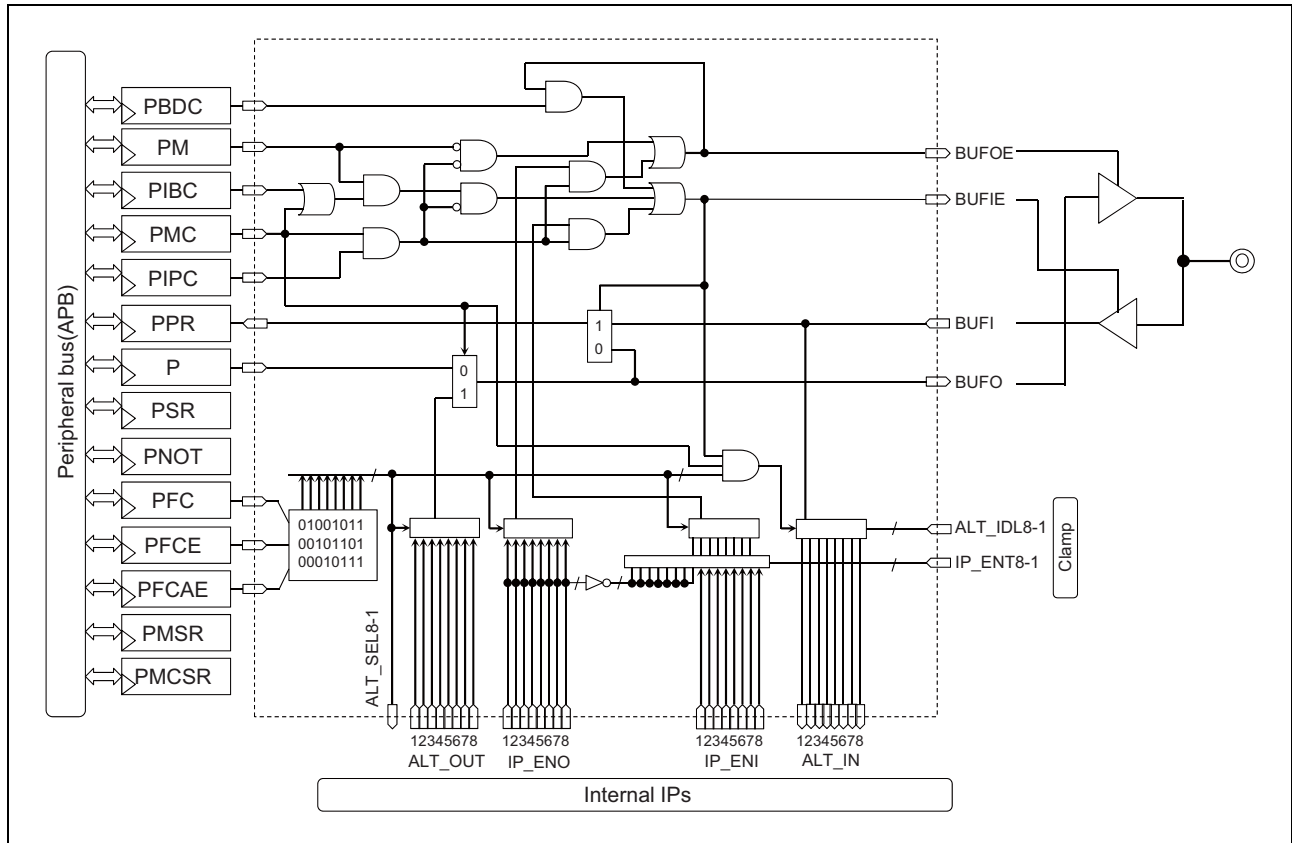


Figure 54.1 Logical Diagram of Port Control

## 54.19 Flowchart Examples of Port Setting

A flowchart example for setting the Pn\_m pin to port mode is shown in (a), an example for setting software I/O control alternative mode is shown in (b), and an example for setting direct I/O control alternative mode is shown in (c).

### (a) Port settings for port mode

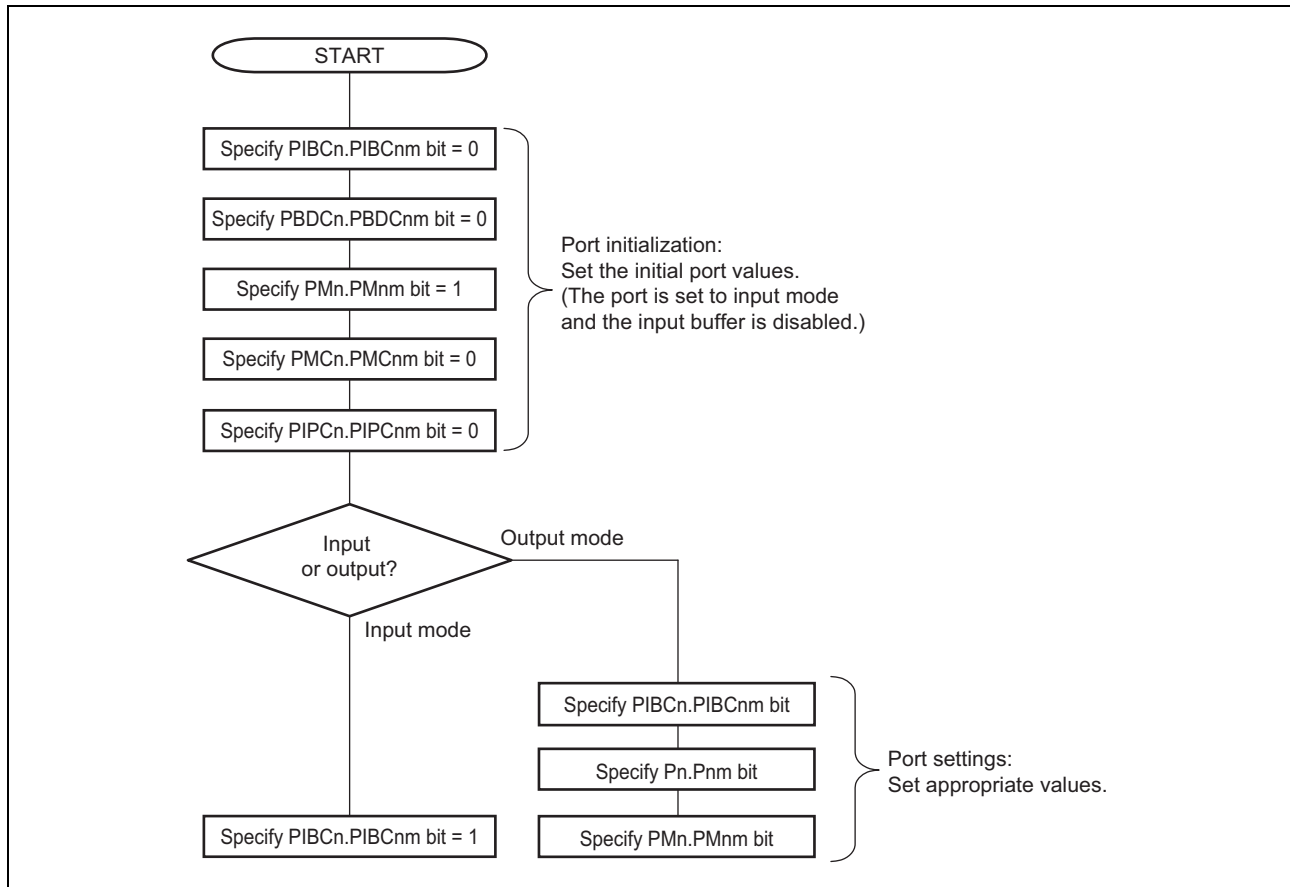
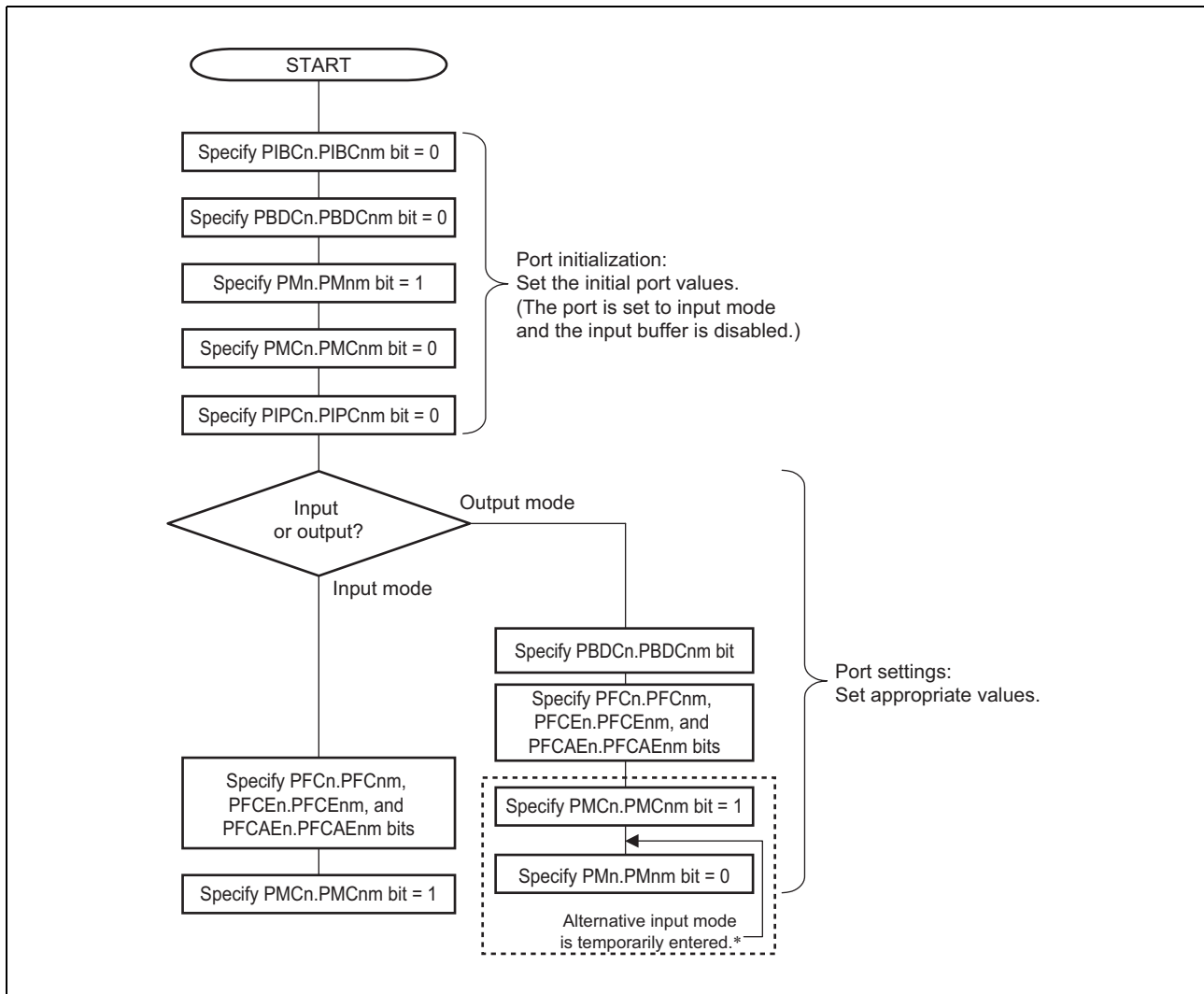


Figure 54.2 Example of Port Settings (in Port Mode) (1/3)

## (b) Port settings for software I/O control alternative mode



**Figure 54.3 Example of Port Settings (in Software I/O Control Alternative Mode) (2/3)**

**Note:** When the Pn\_m pin is set to the output mode in software I/O control alternative mode, the alternative input mode is temporarily entered after the PMCn.PMCnm bit is set to 1 until the PMn.PMnm bit is cleared to 0. Therefore, when an interrupt-related signal is selected as the alternative function for the port pin, appropriate settings should be made so that the interrupt does not occur or is ignored.

## (c) Port settings for direct I/O control alternative mode

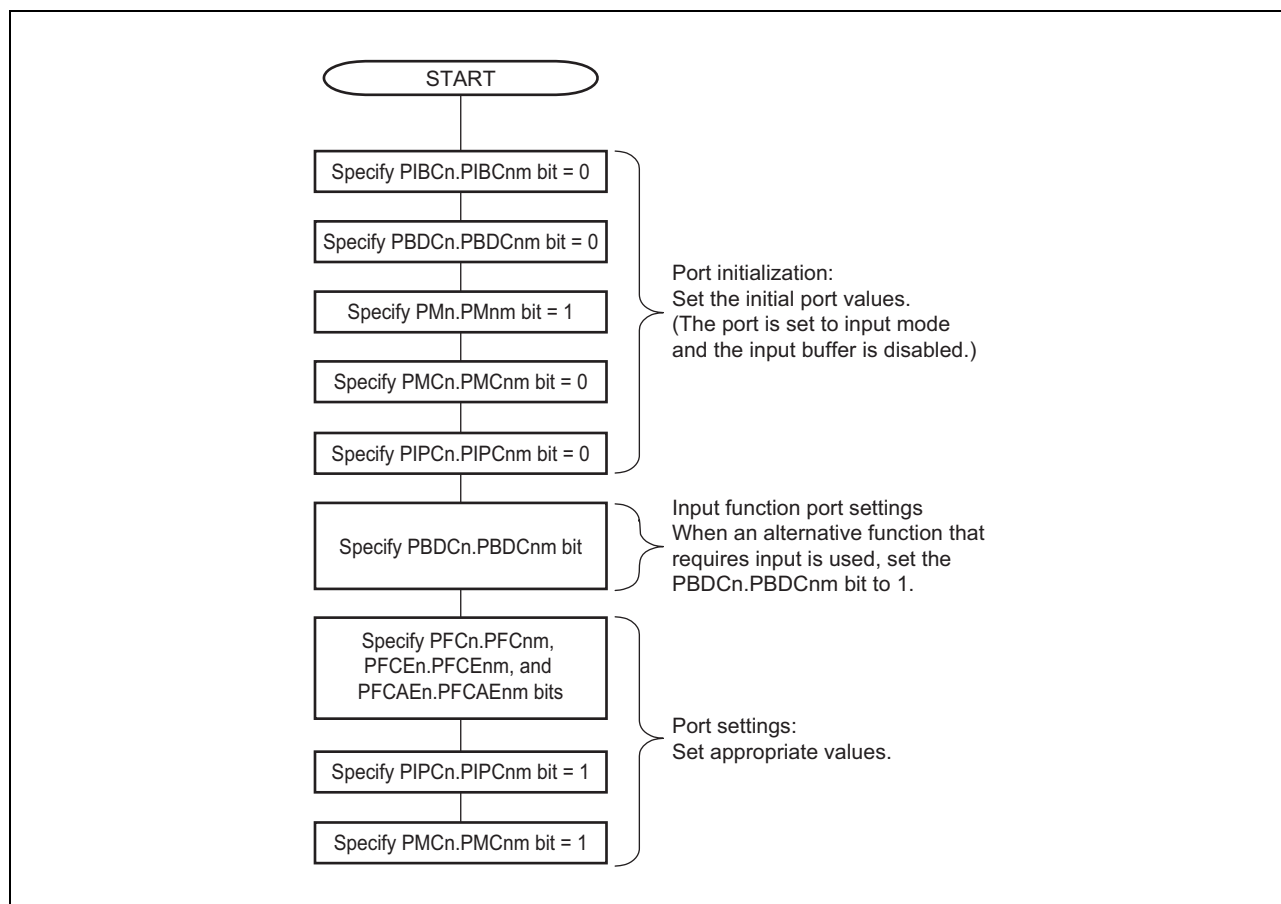


Figure 54.4 Example of Port Settings (in Direct I/O Control Alternative Mode) (3/3)



## 55. Power-Down Modes

This LSI supports sleep mode, software standby mode, deep standby mode, and module standby mode. In power-down modes, functions of CPU, clocks, on-chip memory, or part of on-chip peripheral modules are halted or the power-supply is turned off, through which low power consumption is achieved. These modes are canceled by a reset or interrupt.

### 55.1 Features

#### 55.1.1 States of Processing and Power-Down Modes

##### (1) States of processing

This LSI has three general states of processing: the reset state, the program execution state, and the power-down modes. Figure 55.1 shows transitions between these states.

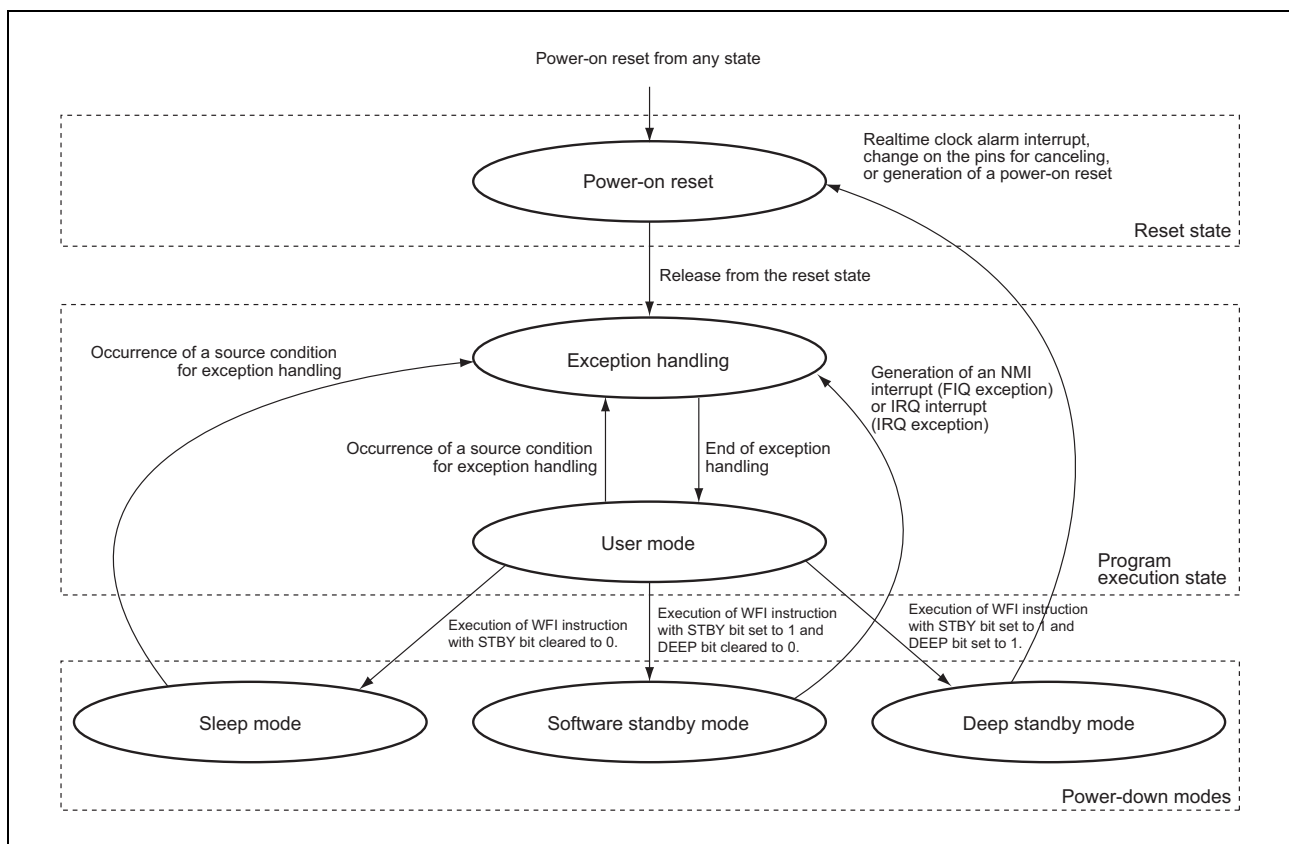


Figure 55.1 Transitions between Processing States

##### (2) Power-down modes

This LSI has the following power-down modes and function:

1. Sleep mode
2. Software standby mode
3. Deep standby mode
4. Module standby function

Table 55.1 shows the transition conditions for entering the modes from the program execution state, as well as the CPU and peripheral module states in each mode and the procedures for canceling each mode.

Table 55.1 States of Power-Down Modes

Power-Down Mode	Transition Conditions	State							
		CPU, CPU Register, Primary Cache, TLB	Secondary Cache	Large-Capacity On-Chip RAM (including on-chip data-retention RAM)	On-Chip Peripheral Modules	Realtime Clock	Internal Power Supply	External Memory	Canceling Procedure
Sleep mode	Execute WFI instruction with STBY bit in STBCR1 cleared to 0	Halted Contents are held.*4	Running	Running	Running	Running*1	Applied	Auto-refresh	<ul style="list-style-type: none"> <li>Interrupt</li> <li>Power-on reset</li> </ul>
Software standby mode	Execute WFI instruction with STBY and DEEP bits in STBCR1 set to 1 and 0, respectively	Halted Contents are held.*4	Halted Contents are held.*4	Halted Contents are held.*5	Halted	Running*1	Applied	Self-refresh	<ul style="list-style-type: none"> <li>NMI or IRQ interrupt</li> <li>Power-on reset</li> </ul>
Deep standby mode	Execute WFI instruction with STBY and DEEP bits in STBCR1 set to 1	Halted Contents are not held.	Halted Contents are not held.	Halted Contents in on-chip data-retention RAM are held*2, and the other contents in large-capacity on-chip RAM are not held.	Halted	Running*1	Shut off	Self-refresh	<ul style="list-style-type: none"> <li>Power-on reset*3</li> <li>Realtime clock alarm interrupt*3</li> <li>Change on the pins for canceling*3</li> </ul>
Module standby mode	Set the MSTP bits in STBCR2 to STBCR13 to 1	Running	Running	Running	Specified module halted	Halted	Applied	Auto-refresh	<ul style="list-style-type: none"> <li>Clear MSTP bit to 0</li> </ul>

Note 1. The realtime clock operates when the START bit in the RCR2 register is set to 1. For details, see section 13, Realtime Clock. When deep standby mode is canceled by a power-on reset, the running state cannot be retained. Make the initial setting for the realtime clock again.

Note 2. Setting the bits RRAMKP3 to RRAMKP0 in the RRAMKP register to 1 enables to retain the data in the corresponding area on the on-chip data-retention RAM during the transition to deep standby mode. When the deep standby mode is canceled by a power-on reset, the retained contents are initialized.

Note 3. Deep standby mode can be canceled by a power-on reset, a realtime clock alarm interrupt, or change on the pins for canceling (P2\_12, P2\_15, P3\_1, P3\_3, P3\_9, P5\_9, P6\_2, P6\_4, P7\_8, P8\_2, P8\_7, P9\_1, and NMI). Even when deep standby mode is canceled by a source other than a power-on reset, the reset exception handling is executed instead of the interrupt exception handling.

Note 4. When sleep mode or software standby mode is canceled by a power-on reset, the retained contents are initialized.

Note 5. By setting the VRAME bit in the SYSCR1 register or VRAMWE bit in the SYSCR2 register to enable accesses, the retained contents are initialized when software standby mode is canceled by a power-on reset. By setting the VRAME bit in the SYSCR1 register or VRAMWE bit in the SYSCR2 register to disable accesses, contents in the large-capacity on-chip RAM (including on-chip data-retention RAM) can be retained when software standby mode is canceled by a power-on reset. Note that the area at addresses from H'2002\_0000 to H'2002\_3FFF is used as working memory for the boot program in boot modes 3 to 5. In boot modes 4 and 5, a 28-Kbyte program is transferred from the external flash memory to the area at addresses from H'2002\_4000 to H'2002\_AFFF. For details, see section 3, Boot Mode.

## 55.2 Register Descriptions

Table 55.2 shows the register configuration.

**Table 55.2 Register Configuration**

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Standby control register 1	STBCR1	R/W	H'00	H'FCFE0020	8
Standby control register 2	STBCR2	R/W	H'6A	H'FCFE0024	8
Standby control register 3	STBCR3	R/W	H'FD	H'FCFE0420	8
Standby control register 4	STBCR4	R/W	H'FF	H'FCFE0424	8
Standby control register 5	STBCR5	R/W	H'FF	H'FCFE0428	8
Standby control register 6	STBCR6	R/W	H'FE	H'FCFE042C	8
Standby control register 7	STBCR7	R/W	H'3F	H'FCFE0430	8
Standby control register 8	STBCR8	R/W	H'FF	H'FCFE0434	8
Standby control register 9	STBCR9	R/W	H'FF	H'FCFE0438	8
Standby control register 10	STBCR10	R/W	H'FF	H'FCFE043C	8
Standby control register 11	STBCR11	R/W	H'FF	H'FCFE0440	8
Standby control register 12	STBCR12	R/W	H'FF	H'FCFE0444	8
Standby control register 13	STBCR13	R/W	H'FF	H'FCFE0470	8
Software reset control register 1	SWRSTCR1	R/W	H'00	H'FCFE0460	8
Software reset control register 2	SWRSTCR2	R/W	H'00	H'FCFE0464	8
Software reset control register 3	SWRSTCR3	R/W	H'80	H'FCFE0468	8
System control register 1	SYSCR1	R/W	H'FF	H'FCFE0400	8
System control register 2	SYSCR2	R/W	H'FF	H'FCFE0404	8
System control register 3	SYSCR3	R/W	H'00	H'FCFE0408	8
CPU status register	CPUSTS	R	H'00	H'FCFE0018	8
Standby request register 1	STBREQ1	R/W	H'00	H'FCFE0030	8
Standby request register 2	STBREQ2	R/W	H'00	H'FCFE0034	8
Standby acknowledge register 1	STBACK1	R	H'00	H'FCFE0040	8
Standby acknowledge register 2	STBACK2	R	H'00	H'FCFE0044	8
On-chip data-retention RAM area setting register	RRAMKP	R/W	H'00	H'FCFF1800	8
Deep standby control register	DSCTR	R/W	H'00	H'FCFF1802	8
Deep standby cancel source select register	DSSSR	R/W	H'0000	H'FCFF1804	16
Deep standby cancel edge select register	DSESR	R/W	H'0000	H'FCFF1806	16
Deep standby cancel source flag register	DSFR	R/W	H'0000	H'FCFF1808	16
XTAL crystal oscillator gain control register	XTALCTR	R/W	H'00	H'FCFF1810	8

### 55.2.1 Standby Control Register 1 (STBCR1)

STBCR1 is an 8-bit readable/writable register that specifies the state of the power-down mode.

Note: • When writing to this register, see section 55.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	STBY	DEEP	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	STBY	0	R/W	Software Standby, Deep Standby
6	DEEP	0	R/W	Specifies transition to software standby mode or deep standby mode. 0x: Executing WFI instruction puts chip into sleep mode. 10: Executing WFI instruction puts chip into software standby mode. 11: Executing WFI instruction puts chip into deep standby mode.
5 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

[Legend]

x: Don't care

### 55.2.2 Standby Control Register 2 (STBCR2)

STBCR2 is an 8-bit readable/writable register that controls the operation of each module.

Note: • When writing to this register, see section 55.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	HIZ	-	-	-	-	-	-	MSTP 20
Initial value:	0	1	1	0	1	0	1	0
R/W:	R/W	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	HIZ	0	R/W	Port High Impedance Selects whether the state of specific output pin is retained or high impedance in software standby mode or deep standby mode. As to which pins are controlled, see section 59.1, Pin States in section 59, States and Handling of Pins. 0: The pin state is retained in software standby mode or deep standby mode. 1: The pin is set to high-impedance in software standby mode or deep standby mode.
6, 5	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
4	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
3	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
2	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
1	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
0	MSTP20	0	R/W	Module Stop 20*1 [not for disclosure] When the MSTP20 bit is set to 1, the clock supply to CoreSight is halted. 0: CoreSight runs. 1: Clock supply to CoreSight is halted.

Note 1. The transition to and release from module stop mode proceed in the steps described in section 55.3.5, Module Standby Function.

### 55.2.3 Standby Control Register 3 (STBCR3)

STBCR3 is an 8-bit readable/writable register that controls the operation of each module.

Note: • When writing to this register, see section 55.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	MSTP 37	MSTP 36	MSTP 35	MSTP 34	MSTP 33	MSTP 32	MSTP 31	MSTP 30
Initial value:	1	1	1	1	1	1	0	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	MSTP37	1	R/W	<b>Module Stop 37</b> When the MSTP37 bit is set to 1, the clock supply to the IEBus™ controller is halted. 0: The IEBus™ controller runs. 1: Clock supply to the IEBus™ controller is halted.
6	MSTP36	1	R/W	<b>Module Stop 36</b> When the MSTP36 bit is set to 1, the clock supply to serial communication interface (IrDA) is halted. 0: Serial communication interface (IrDA) runs. 1: Clock supply to serial communication interface (IrDA) is halted.
5	MSTP35	1	R/W	<b>Module Stop 35</b> When the MSTP35 bit is set to 1, the clock supply to the LIN interface channel 0 is halted. 0: The LIN interface channel 0 runs. 1: Clock supply to the LIN interface channel 0 is halted.
4	MSTP34	1	R/W	<b>Module Stop 34</b> When the MSTP34 bit is set to 1, the clock supply to the LIN interface channel 1 is halted. 0: The LIN interface channel 1 runs. 1: Clock supply to the LIN interface channel 1 is halted.
3	MSTP33	1	R/W	<b>Module Stop 33</b> When the MSTP33 bit is set to 1, the clock supply to the multi-function timer pulse unit 2 is halted. 0: The multi-function timer pulse unit 2 runs. 1: Clock supply to the multi-function timer pulse unit 2 is halted.
2	MSTP32	1	R/W	<b>Module Stop 32</b> When the MSTP32 bit is set to 1, the clock supply to the CAN interface is halted. 0: The CAN interface runs. 1: Clock supply to the CAN interface is halted.
1	MSTP31	0	R/W	<b>Module Stop 31</b> When the MSTP31 bit is set to 1, the clock supply to the A/D converter is halted and the analog power supply to the A/D converter is shut off. 0: Clock signal and analog voltage are supplied to the A/D converter. 1: Supply of clock signal and analog voltage to the A/D converter is shut off.
0	MSTP30	1	R/W	<b>Module Stop 30</b> When the MSTP30 bit is set to 1, the clock supply to the motor control PWM timer is halted. 0: The motor control PWM timer runs. 1: Clock supply to the motor control PWM timer is halted.

### 55.2.4 Standby Control Register 4 (STBCR4)

STBCR4 is an 8-bit readable/writable register that controls the operation of each module.

Note: • When writing to this register, see section 55.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	MSTP 47	MSTP 46	MSTP 45	MSTP 44	MSTP 43	MSTP 42	MSTP 41	MSTP 40
Initial value:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	MSTP47	1	R/W	<b>Module Stop 47</b> When the MSTP47 bit is set to 1, the clock supply to channel 0 of the serial communication interface with FIFO is halted. 0: Channel 0 of the serial communication interface with FIFO runs. 1: Clock supply to channel 0 of the serial communication interface with FIFO is halted.
6	MSTP46	1	R/W	<b>Module Stop 46</b> When the MSTP46 bit is set to 1, the clock supply to channel 1 of the serial communication interface with FIFO is halted. 0: Channel 1 of the serial communication interface with FIFO runs. 1: Clock supply to channel 1 of the serial communication interface with FIFO is halted.
5	MSTP45	1	R/W	<b>Module Stop 45</b> When the MSTP45 bit is set to 1, the clock supply to channel 2 of the serial communication interface with FIFO is halted. 0: Channel 2 of the serial communication interface with FIFO runs. 1: Clock supply to channel 2 of the serial communication interface with FIFO is halted.
4	MSTP44	1	R/W	<b>Module Stop 44</b> When the MSTP44 bit is set to 1, the clock supply to channel 3 of the serial communication interface with FIFO is halted. 0: Channel 3 of the serial communication interface with FIFO runs. 1: Clock supply to channel 3 of the serial communication interface with FIFO is halted.
3	MSTP43	1	R/W	<b>Module Stop 43</b> When the MSTP43 bit is set to 1, the clock supply to channel 4 of the serial communication interface with FIFO is halted. 0: Channel 4 of the serial communication interface with FIFO runs. 1: Clock supply to channel 4 of the serial communication interface with FIFO is halted.
2	MSTP42	1	R/W	<b>Module Stop 42</b> When the MSTP42 bit is set to 1, the clock supply to channel 5 of the serial communication interface with FIFO is halted. 0: Channel 5 of the serial communication interface with FIFO runs. 1: Clock supply to channel 5 of the serial communication interface with FIFO is halted.
1	MSTP41	1	R/W	<b>Module Stop 41</b> When the MSTP41 bit is set to 1, the clock supply to channel 6 of the serial communication interface with FIFO is halted. 0: Channel 6 of the serial communication interface with FIFO runs. 1: Clock supply to channel 6 of the serial communication interface with FIFO is halted.
0	MSTP40	1	R/W	<b>Module Stop 40</b> When the MSTP40 bit is set to 1, the clock supply to channel 7 of the serial communication interface with FIFO is halted. 0: Channel 7 of the serial communication interface with FIFO runs. 1: Clock supply to channel 7 of the serial communication interface with FIFO is halted.

### 55.2.5 Standby Control Register 5 (STBCR5)

STBCR5 is an 8-bit readable/writable register that controls the operation of each module.

Note: • When writing to this register, see section 55.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	MSTP 57	MSTP 56	MSTP 55	MSTP 54	MSTP 53	MSTP 52	MSTP 51	MSTP 50
Initial value:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	MSTP57	1	R/W	Module Stop 57 When the MSTP57 bit is set to 1, the clock supply to channel 0 of the serial communication interface (SCI) is halted. 0: Channel 0 of the serial communication interface (SCI) runs. 1: Clock supply to channel 0 of the serial communication interface (SCI) is halted.
6	MSTP56	1	R/W	Module Stop 56 When the MSTP56 bit is set to 1, the clock supply to channel 1 of the serial communication interface (SCI) is halted. 0: Channel 1 of the serial communication interface (SCI) runs. 1: Clock supply to channel 1 of the serial communication interface (SCI) is halted.
5	MSTP55	1	R/W	Module Stop 55 When the MSTP55 bit is set to 1, the clock supply to channel 0 of the sound generator is halted. 0: Channel 0 of the sound generator runs. 1: Clock supply to channel 0 of the sound generator is halted.
4	MSTP54	1	R/W	Module Stop 54 When the MSTP54 bit is set to 1, the clock supply to channel 1 of the sound generator is halted. 0: Channel 1 of the sound generator runs. 1: Clock supply to channel 1 of the sound generator is halted.
3	MSTP53	1	R/W	Module Stop 53 When the MSTP53 bit is set to 1, the clock supply to channel 2 of the sound generator is halted. 0: Channel 2 of the sound generator runs. 1: Clock supply to channel 2 of the sound generator is halted.
2	MSTP52	1	R/W	Module Stop 52 When the MSTP52 bit is set to 1, the clock supply to channel 3 of the sound generator is halted. 0: Channel 3 of the sound generator runs. 1: Clock supply to channel 3 of the sound generator is halted.
1	MSTP51	1	R/W	Module Stop 51 When the MSTP51 bit is set to 1, the clock supply to channel 0 of the OS timer is halted. 0: Channel 0 of the OS timer runs. 1: Clock supply to channel 0 of the OS timer is halted.
0	MSTP50	1	R/W	Module Stop 50 When the MSTP50 bit is set to 1, the clock supply to channel 1 of the OS timer is halted. 0: Channel 1 of the OS timer runs. 1: Clock supply to channel 1 of the OS timer is halted.



### 55.2.6 Standby Control Register 6 (STBCR6)

STBCR6 is an 8-bit readable/writable register that controls the operation of each module.

Note: • When writing to this register, see section 55.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	MSTP 67	MSTP 66	MSTP 65	MSTP 64	MSTP 63	MSTP 62	MSTP 61	MSTP 60
Initial value:	1	1	1	1	1	1	1	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	MSTP67	1	R/W	Module Stop 67 When the MSTP67 bit is set to 1, the clock supply to the A/D converter is halted. 0: The A/D converter runs. 1: Clock supply to channel 0 of the A/D converter is halted.
6	MSTP66	1	R/W	Module Stop 66*1 When the MSTP66 bit is set to 1, the clock supply to the capture engine unit is halted. 0: The capture engine unit runs. 1: Clock supply to the capture engine unit is halted.
5	MSTP65	1	R/W	Module Stop 65 When the MSTP65 bit is set to 1, the clock supply to channel 0 of the display out comparison unit is halted. 0: Channel 0 of the display out comparison unit runs. 1: Clock supply to channel 0 of the display out comparison unit is halted.
4	MSTP64	1	R/W	Module Stop 64 When the MSTP64 bit is set to 1, the clock supply to channel 1 of the display out comparison unit is halted. 0: Channel 1 of the display out comparison unit runs. 1: Clock supply to channel 1 of the display out comparison unit is halted.
3	MSTP63	1	R/W	Module Stop 63 When the MSTP63 bit is set to 1, the clock supply to channel 0 of the dynamic range compression is halted. 0: Channel 0 of the dynamic range compression runs. 1: Clock supply to channel 0 of the dynamic range compression is halted.
2	MSTP62	1	R/W	Module Stop 62 When the MSTP62 bit is set to 1, the clock supply to channel 1 of the dynamic range compression is halted. 0: Channel 1 of the dynamic range compression runs. 1: Clock supply to channel 1 of the dynamic range compression is halted.
1	MSTP61	1	R/W	Module Stop 61*1 When the MSTP61 bit is set to 1, the clock supply to the JPEG codec unit is halted. 0: The JPEG codec unit runs. 1: Clock supply to the JPEG codec unit is halted.
0	MSTP60	0	R/W	Module Stop 60 When the MSTP60 bit is set to 1, the clock supply to the realtime clock is halted. 0: The realtime clock runs. 1: Clock supply to the realtime clock is halted. Note: When the realtime clock is halted, set the bits in registers shown below. <ul style="list-style-type: none"> <li>• Set bit RTCEN in the control register 2 (RCR2) to 0.</li> <li>• Set bits RCKSEL[1:0] in the control register 5 (RCR5) to 00.</li> </ul> After the settings above, set bit MSTP60 to 1.

Note 1. The transition to and release from module stop mode proceed in the steps described in section 55.3.5, Module Standby Function.

### 55.2.7 Standby Control Register 7 (STBCR7)

STBCR7 is an 8-bit readable/writable register that controls the operation of each module.

Note: • When writing to this register, see section 55.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	MSTP 77	MSTP 76	-	MSTP 74	MSTP 73	-	MSTP 71	MSTP 70
Initial value:	0	0	1	1	1	1	1	1
R/W:	R/W	R/W	R	R	R/W	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	MSTP77	0	R/W	<b>Module Stop 77</b> When the MSTP77 bit is set to 1, the clock supply to channel 0 of the digital video decoder is halted. 0: Channel 0 of the digital video decoder runs. 1: Clock supply to channel 0 of the digital video decoder is halted. Note: When this bit is set to 1, A/D converter A for the input of video signals will stop operating and the LSI will be placed in the power saving state. When the LSI is released from the power saving mode (by clearing this bit to 0), the precision of conversion by A/D converter A cannot be guaranteed until the time for return from the power saving state has elapsed. For details, see section 58, Electrical Characteristics.
6	MSTP76	0	R/W	<b>Module Stop 76</b> When the MSTP76 bit is set to 1, the clock supply to channel 1 of the digital video decoder is halted. 0: Channel 1 of the digital video decoder runs. 1: Clock supply to channel 1 of the digital video decoder is halted. Note: When this bit is set to 1, A/D converter A for the input of video signals will stop operating and the LSI will be placed in the power saving state. When the LSI is released from the power saving mode (by clearing this bit to 0), the precision of conversion by A/D converter A cannot be guaranteed until the time for return from the power saving state has elapsed. For details, see section 58, Electrical Characteristics.
5	—	1	R	<b>Reserved</b> This bit is always read as 1. The write value should always be 1.
4	MSTP74	1	R	<b>Module Stop 74*1</b> When the MSTP74 bit is set to 1, the clock supply to the Ethernet controller is halted. 0: The Ethernet controller runs. 1: Clock supply to the Ethernet controller is halted.
3	MSTP73	1	R/W	<b>Module Stop 73</b> When the MSTP73 bit is set to 1, the clock supply to the NAND flash memory controller 4 is halted. 0: The NAND flash memory controller 4 runs. 1: Clock supply to the NAND flash memory controller 4 is halted.
2	—	1	R	<b>Reserved</b> This bit is always read as 1. The write value should always be 1.
1	MSTP71	1	R/W	<b>Module Stop 71</b> When the MSTP71 bit is set to 1, the clock supply to channel 0 of the USB 2.0 host/function module is halted. 0: Channel 0 of the USB 2.0 host/function module runs. 1: Clock supply to channel 0 of the USB 2.0 host/function module is halted.
0	MSTP70	1	R/W	<b>Module Stop 70</b> When the MSTP70 bit is set to 1, the clock supply to channel 1 of the USB 2.0 host/function module is halted. 0: Channel 1 of the USB 2.0 host/function module runs. 1: Clock supply to channel 1 of the USB 2.0 host/function module is halted.

Note 1. The transition to and release from module stop mode proceed in the steps described in section 55.3.5, Module Standby Function.

## 55.2.8 Standby Control Register 8 (STBCR8)

STBCR8 is an 8-bit readable/writable register that controls the operation of each module.

Note: • When writing to this register, see section 55.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	MSTP 87	MSTP 86	MSTP 85	MSTP 84	MSTP 83	MSTP 82	MST P81	-
Initial value:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
7	MSTP87	1	R/W	Module Stop 87*1 When the MSTP87 bit is set to 1, the clock supply to channel 0 of the image renderer is halted. 0: Channel 0 of the image renderer runs. 1: Clock supply to channel 0 of the image renderer is halted.
6	MSTP86	1	R/W	Module Stop 86 When the MSTP86 bit is set to 1, the clock supply to channel 1 of the image renderer is halted. 0: Channel 1 of the image renderer runs. 1: Clock supply to channel 1 of the image renderer is halted.
5	MSTP85	1	R/W	Module Stop 85*1 When the MSTP85 bit is set to 1, the clock supply to the display image renderer is halted. 0: The display image renderer runs. 1: Clock supply to the display image renderer is halted.
4	MSTP84	1	R/W	Module Stop 84 When the MSTP84 bit is set to 1, the clock supply to the MMC host interface is halted. 0: The MMC host interface runs. 1: Clock supply to the MMC host interface is halted.
3	MSTP83	1	R/W	Module Stop 83*1 When the MSTP83 bit is set to 1, the clock supply to the media local bus is halted. 0: The media local bus runs. 1: Clock supply to the media local bus is halted.
2	MSTP82	1	R/W	Module Stop 82*1 When the MSTP82 bit is set to 1, the clock supply to the EthernetAVB is halted. 0: The EthernetAVB runs. 1: Clock supply to the EthernetAVB is halted.
1	MSTP81	1	R/W	Module Stop 81 When the MSTP81 bit is set to 1, the clock supply to SCUX is halted. 0: SCUX runs. 1: Clock supply to SCUX is halted.
0	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.

Note 1. The transition to and release from module stop mode proceed in the steps described in section 55.3.5, Module Standby Function.

### 55.2.9 Standby Control Register 9 (STBCR9)

STBCR9 is an 8-bit readable/writable register that controls the operation of each module.

Note: • When writing to this register, see section 55.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	MSTP 97	MSTP 96	MSTP 95	MSTP 94	MSTP 93	MSTP 92	MSTP 91	MSTP 90
Initial value:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	MSTP97	1	R/W	Module Stop 97 When the MSTP97 bit is set to 1, the clock supply to channel 0 of the I <sup>2</sup> C bus interface is halted. 0: Channel 0 of the I <sup>2</sup> C bus interface runs. 1: Clock supply to channel 0 of the I <sup>2</sup> C bus interface is halted.
6	MSTP96	1	R/W	Module Stop 96 When the MSTP96 bit is set to 1, the clock supply to channel 1 of the I <sup>2</sup> C bus interface is halted. 0: Channel 1 of the I <sup>2</sup> C bus interface runs. 1: Clock supply to channel 1 of the I <sup>2</sup> C bus interface is halted.
5	MSTP95	1	R/W	Module Stop 95 When the MSTP95 bit is set to 1, the clock supply to channel 2 of the I <sup>2</sup> C bus interface is halted. 0: Channel 2 of the I <sup>2</sup> C bus interface runs. 1: Clock supply to channel 2 of the I <sup>2</sup> C bus interface is halted.
4	MSTP94	1	R/W	Module Stop 94 When the MSTP94 bit is set to 1, the clock supply to channel 3 of the I <sup>2</sup> C bus interface is halted. 0: Channel 3 of the I <sup>2</sup> C bus interface runs. 1: Clock supply to channel 3 of the I <sup>2</sup> C bus interface is halted.
3	MSTP93	1	R/W	Module Stop 93 When the MSTP93 bit is set to 1, the clock supply to channel 0 of the SPI multi I/O bus controller is halted. 0: Channel 04 of the SPI multi I/O bus controller runs. 1: Clock supply to channel 0 of the SPI multi I/O bus controller is halted.
2	MSTP92	1	R/W	Module Stop 92 When the MSTP92 bit is set to 1, the clock supply to channel 1 of the SPI multi I/O bus controller is halted. 0: Channel 1 of the SPI multi I/O bus controller runs. 1: Clock supply to channel 1 of the SPI multi I/O bus controller is halted.
1	MSTP91	1	R/W	Module Stop 91*1 When the MSTP91 bit is set to 1, the clock supply to channel 0 of the video display controller 5 and the LVDS output interface is halted. 0: Channel 0 of the video display controller 5 and the LVDS output interface run. 1: Clock supply to channel 0 of the video display controller 5 and the LVDS output interface is halted.
0	MSTP90	1	R/W	Module Stop 90*1 When the MSTP90 bit is set to 1, the clock supply to channel 1 of the video display controller 5 is halted. 0: Channel 1 of the video display controller 5 runs. 1: Clock supply to channel 1 of the video display controller 5 is halted.

Note 1. The transition to and release from module stop mode proceed in the steps described in section 55.3.5, Module Standby Function.

### 55.2.10 Standby Control Register 10 (STBCR10)

STBCR10 is an 8-bit readable/writable register that controls the operation of each module.

Note: • When writing to this register, see section 55.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	MSTP 107	MSTP 106	MSTP 105	MSTP 104	MSTP 103	MSTP 102	MSTP 101	MSTP 100
Initial value:	1	1	1	1	1	1	1	1
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	MSTP107	1	R/W	Module Stop 107 When the MSTP107 bit is set to 1, the clock supply to channel 0 of the Renesas serial peripheral interface is halted. 0: Channel 0 of the Renesas serial peripheral interface runs. 1: Clock supply to channel 0 of the Renesas serial peripheral interface is halted.
6	MSTP106	1	R/W	Module Stop 106 When the MSTP106 bit is set to 1, the clock supply to channel 1 of the Renesas serial peripheral interface is halted. 0: Channel 1 of the Renesas serial peripheral interface runs. 1: Clock supply to channel 1 of the Renesas serial peripheral interface is halted.
5	MSTP105	1	R/W	Module Stop 105 When the MSTP105 bit is set to 1, the clock supply to channel 2 of the Renesas serial peripheral interface is halted. 0: Channel 2 of the Renesas serial peripheral interface runs. 1: Clock supply to channel 2 of the Renesas serial peripheral interface is halted.
4	MSTP104	1	R/W	Module Stop 104 When the MSTP104 bit is set to 1, the clock supply to channel 3 of the Renesas serial peripheral interface is halted. 0: Channel 3 of the Renesas serial peripheral interface runs. 1: Clock supply to channel 3 of the Renesas serial peripheral interface is halted.
3	MSTP103	1	R/W	Module Stop 103 When the MSTP103 bit is set to 1, the clock supply to channel 4 of the Renesas serial peripheral interface is halted. 0: Channel 4 of the Renesas serial peripheral interface runs. 1: Clock supply to channel 4 of the Renesas serial peripheral interface is halted.
2	MSTP102	1	R/W	Module Stop 102 When the MSTP102 bit is set to 1, the clock supply to the CD-ROM decoder is halted. 0: The CD-ROM decoder runs. 1: Clock supply to the CD-ROM decoder is halted.
1	MSTP101	1	R/W	Module Stop 101 When the MSTP101 bit is set to 1, the clock supply to the Renesas SPDIF interface is halted. 0: The Renesas SPDIF interface runs. 1: Clock supply to the Renesas SPDIF interface is halted.
0	MSTP100	1	R/W	Module Stop 100*1 When the MSTP100 bit is set to 1, the clock supply to the OpenVG™-compliant Renesas graphics processor is halted. 0: The OpenVG™-compliant Renesas graphics processor runs. 1: Clock supply to the OpenVG™-compliant Renesas graphics processor is halted.

Note 1. The transition to and release from module stop mode proceed in the steps described in section 55.3.5, Module Standby Function.

### 55.2.11 Standby Control Register 11 (STBCR11)

STBCR11 is an 8-bit readable/writable register that controls the operation of each module.

Note: • When writing to this register, see section 55.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	-	-	MSTP 115	MSTP 114	MSTP 113	MSTP 112	MSTP 111	MSTP 110
Initial value:	1	1	1	1	1	1	1	1
R/W:	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
5	MSTP115	1	R/W	Module Stop 115 When the MSTP115 bit is set to 1, the clock supply to channel 0 of the serial sound interface is halted. 0: Channel 0 of the serial sound interface runs. 1: Clock supply to channel 0 of the serial sound interface is halted.
4	MSTP114	1	R/W	Module Stop 114 When the MSTP114 bit is set to 1, the clock supply to channel 1 of the serial sound interface is halted. 0: Channel 1 of the serial sound interface runs. 1: Clock supply to channel 1 of the serial sound interface is halted.
3	MSTP113	1	R/W	Module Stop 113 When the MSTP113 bit is set to 1, the clock supply to channel 2 of the serial sound interface is halted. 0: Channel 2 of the serial sound interface runs. 1: Clock supply to channel 2 of the serial sound interface is halted.
2	MSTP112	1	R/W	Module Stop 112 When the MSTP112 bit is set to 1, the clock supply to channel 3 of the serial sound interface is halted. 0: Channel 3 of the serial sound interface runs. 1: Clock supply to channel 3 of the serial sound interface is halted.
1	MSTP111	1	R/W	Module Stop 111 When the MSTP111 bit is set to 1, the clock supply to channel 4 of the serial sound interface is halted. 0: Channel 4 of the serial sound interface runs. 1: Clock supply to channel 4 of the serial sound interface is halted.
0	MSTP110	1	R/W	Module Stop 110 When the MSTP110 bit is set to 1, the clock supply to channel 5 of the serial sound interface is halted. 0: Channel 5 of the serial sound interface runs. 1: Clock supply to channel 5 of the serial sound interface is halted.

### 55.2.12 Standby Control Register 12 (STBCR12)

STBCR12 is an 8-bit readable/writable register that controls the operation of each module.

Note: • When writing to this register, see section 55.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	MSTP 123	MSTP 122	MSTP 121	MSTP 120
Initial value:	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
3	MSTP123	1	R/W	Module Stop 123 When the MSTP123 bit is set to 1, the SD host interface 00 is halted. 0: The SD host interface 00 runs. 1: Clock supply to the SD host interface 00 is halted.
2	MSTP122	1	R/W	Module Stop 122 When the MSTP122 bit is set to 1, the SD host interface 01 is halted. 0: The SD host interface 01 runs. 1: Clock supply to the SD host interface 01 is halted.
1	MSTP121	1	R/W	Module Stop 121 When the MSTP121 bit is set to 1, the SD host interface 10 is halted. 0: The SD host interface 10 runs. 1: Clock supply to the SD host interface 10 is halted.
0	MSTP120	1	R/W	Module Stop 120 When the MSTP120 bit is set to 1, the SD host interface 11 is halted. 0: The SD host interface 11 runs. 1: Clock supply to the SD host interface 11 is halted.

### 55.2.13 Standby Control Register 13 (STBCR13)

STBCR13 is an 8-bit readable/writable register that controls the operation of each module.

Note: • When writing to this register, see section 55.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	MSTP 132	MSTP 131	-
Initial value:	1	1	1	1	1	1	1	1
R/W:	R	R	R	R	R	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 3	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
2	MSTP132	1	R/W	Module Stop 132 When the MSTP132 bit is set to 1, the clock supply to channel 1 of the pixel format converter is halted. 0: Channel 1 of the pixel format converter runs. 1: Clock supply to channel 1 of the pixel format converter is halted.
1	MSTP131	1	R/W	Module Stop 131 When the MSTP131 bit is set to 1, the clock supply to channel 0 of the pixel format converter is halted. 0: Channel 0 of the pixel format converter runs. 1: Clock supply to channel 0 of the pixel format converter is halted.
0	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.



### 55.2.14 Software Reset Control Register 1 (SWRSTCR1)

SWRSTCR1 is an 8-bit readable/writable register that controls a software reset for the serial sound interface and the operation of the crystal resonator for audio.

Note: • When writing to this register, see section 55.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	AXT ALE	SRST 16	SRST 15	SRST 14	SRST 13	SRST 12	SRST 11	-
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
7	AXTALE	0	R/W	AUDIO_X1 Clock Control Controls the function of AUDIO_X1 pin. 0: Runs the on-chip crystal oscillator/enables the external clock input. 1: Halts the on-chip crystal oscillator/disables the external clock input.
6	SRST16	0	R/W	Serial Sound Interface Channel 0 Software Reset Controls the serial sound interface channel 0 reset with software. 0: The serial sound interface channel 0 reset is canceled. 1: The serial sound interface channel 0 is reset.
5	SRST15	0	R/W	Serial Sound Interface Channel 1 Software Reset Controls the serial sound interface channel 1 reset with software. 0: The serial sound interface channel 1 reset is canceled. 1: The serial sound interface channel 1 is reset.
4	SRST14	0	R/W	Serial Sound Interface Channel 2 Software Reset Controls the serial sound interface channel 2 reset with software. 0: The serial sound interface channel 2 reset is canceled. 1: The serial sound interface channel 2 is reset.
3	SRST13	0	R/W	Serial Sound Interface Channel 3 Software Reset Controls the serial sound interface channel 3 reset with software. 0: The serial sound interface channel 3 reset is canceled. 1: The serial sound interface channel 3 is reset.
2	SRST12	0	R/W	Serial Sound Interface Channel 4 Software Reset Controls the serial sound interface channel 4 reset with software. 0: The serial sound interface channel 4 reset is canceled. 1: The serial sound interface channel 4 is reset.
1	SRST11	0	R/W	Serial Sound Interface Channel 5 Software Reset Controls the serial sound interface channel 5 reset with software. 0: The serial sound interface channel 5 reset is canceled. 1: The serial sound interface channel 5 is reset.
0	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

### 55.2.15 Software Reset Control Register 2 (SWRSTCR2)

SWRSTCR2 is an 8-bit readable/writable register that controls a software reset for each module.

Note: • When writing to this register, see section 55.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	SRST 21	-
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	SRST21	0	R/W	JPEG Codec Unit Software Reset Controls the JPEG codec unit reset with software. 0: The JPEG codec unit reset is canceled. 1: The JPEG codec unit is reset.
0	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.

### 55.2.16 Software Reset Control Register 3 (SWRSTCR3)

SWRSTCR3 is an 8-bit readable/writable register that controls a software reset for each module.

Note: • When writing to this register, see section 55.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	SRST 32	-	-
Initial value:	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R/W	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	—	1	R	Reserved This bit is always read as 1. The write value should always be 1.
6 to 3	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
2	SRST32	0	R/W	OpenVG™-Compliant Renesas Graphics Processor Software Reset Controls the OpenVG™-compliant Renesas graphics processor reset with software. 0: The OpenVG™-compliant Renesas graphics processor reset is canceled. 1: The OpenVG™-compliant Renesas graphics processor is reset.
1, 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

### 55.2.17 System Control Register 1 (SYSCR1)

SYSCR1 is an 8-bit readable/writable register that enables or disables access (read and write) to a specified page in the large-capacity on-chip RAM.

When a VRAMEn (n = 0 to 4) bit in SYSCR1 is set to 1, access to page n is enabled. When a VRAMEn bit is cleared to 0, page n cannot be accessed. In this case, an undefined value is returned when reading data or fetching an instruction from page n, and writing to page n is ignored. The initial value of a VRAMEn bit is 1.

SYSCR1 should be set with a program located in an area other than the large-capacity on-chip RAM. Furthermore, an instruction to read SYSCR1 should be located immediately after the instruction to write to SYSCR1. If not, normal access is not guaranteed.

Note: • When writing to this register, see section 55.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	VRAME4	VRAME3	VRAME2	VRAME1	VRAME0
Initial value:	1	1	1	1	1	1	1	1
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
4	VRAME4	1	R/W	RAM Enable 4 (corresponding area: page 4*1 in large-capacity on-chip RAM) 0: Access to page 4 is disabled. 1: Access to page 4 is enabled.
3	VRAME3	1	R/W	RAM Enable 3 (corresponding area: page 3*1 in large-capacity on-chip RAM) 0: Access to page 3 is disabled. 1: Access to page 3 is enabled.
2	VRAME2	1	R/W	RAM Enable 2 (corresponding area: page 2*1 in large-capacity on-chip RAM) 0: Access to page 2 is disabled. 1: Access to page 2 is enabled.
1	VRAME1	1	R/W	RAM Enable 1 (corresponding area: page 1*1 in large-capacity on-chip RAM) 0: Access to page 1 is disabled. 1: Access to page 1 is enabled.
0	VRAME0	1	R/W	RAM Enable 0 (corresponding area: page 0*1 in large-capacity on-chip RAM) 0: Access to page 0 is disabled. 1: Access to page 0 is enabled.

Note 1. For addresses in each page, see section 53, On-Chip RAM.

### 55.2.18 System Control Register 2 (SYSCR2)

SYSCR2 is an 8-bit readable/writable register that enables or disables writing to a specified page in the large-capacity on-chip RAM.

When a VRAMWEn (n = 0 to 4) bit in SYSCR2 is set to 1, writing to page n is enabled. When a VRAMWEn bit is cleared to 0, writing to page n is ignored. The initial value of a VRAMWEn bit is 1.

SYSCR2 should be set with a program located in an area other than the large-capacity on-chip RAM. Furthermore, an instruction to read SYSCR2 should be located immediately after the instruction to write to SYSCR2. If not, normal access is not guaranteed.

Note: • When writing to this register, see section 55.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	VRAM WE4	VRAM WE3	VRAM WE2	VRAM WE1	VRAM WE0
Initial value:	1	1	1	1	1	1	1	1
R/W:	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	—	All 1	R	Reserved These bits are always read as 1. The write value should always be 1.
4	VRAMWE4	1	R/W	RAM Write Enable 4 (corresponding area: page 4*1 in large-capacity on-chip RAM) 0: Writing to page 4 is disabled. 1: Writing to page 4 is enabled.
3	VRAMWE3	1	R/W	RAM Write Enable 3 (corresponding area: page 3*1 in large-capacity on-chip RAM) 0: Writing to page 3 is disabled. 1: Writing to page 3 is enabled.
2	VRAMWE2	1	R/W	RAM Write Enable 2 (corresponding area: page 2*1 in large-capacity on-chip RAM) 0: Writing to page 2 is disabled. 1: Writing to page 2 is enabled.
1	VRAMWE1	1	R/W	RAM Write Enable 1 (corresponding area: page 1*1 in large-capacity on-chip RAM) 0: Writing to page 1 is disabled. 1: Writing to page 1 is enabled.
0	VRAMWE0	1	R/W	RAM Write Enable 0 (corresponding area: page 0*1 in large-capacity on-chip RAM) 0: Writing to page 0 is disabled. 1: Writing to page 0 is enabled.

Note 1. For addresses in each page, see section 53, On-Chip RAM.

### 55.2.19 System Control Register 3 (SYSCR3)

SYSCR3 is an 8-bit readable/writable register that enables or disables writing to a specified page in the on-chip data-retention RAM.

When a RRAMWEn ( $n = 0$  to  $3$ ) bit in SYSCR3 is set to 1, writing to page  $n$  is enabled. When a RRAMWEn bit is cleared to 0, writing to page  $n$  is ignored. The initial value of a RRAMWEn bit is 0.

SYSCR3 should be set with a program located in an area other than the on-chip data-retention RAM.

Note: • When writing to this register, see section 55.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	RRAM WE3	RRAM WE2	RRAM WE1	RRAM WE0
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	RRAMWE3	0	R/W	RAM Write Enable 3 (corresponding area: page $3^{*2}$ in on-chip data-retention RAM) 0: Writing to page 3 is disabled. 1: Writing to page 3 is enabled.
2	RRAMWE2	0	R/W	RAM Write Enable 2 (corresponding area: page $2^{*2}$ in on-chip data-retention RAM) 0: Writing to page 2 is disabled. 1: Writing to page 2 is enabled.
1	RRAMWE1	0	R/W	RAM Write Enable 1 (corresponding area: page $1^{*2}$ in on-chip data-retention RAM) 0: Writing to page 1 is disabled. 1: Writing to page 1 is enabled.
0	RRAMWE0	0	R/W	RAM Write Enable 0 (corresponding area: page $0^{*2}$ in on-chip data-retention RAM) 0: Writing to page 0 is disabled. 1: Writing to page 0 is enabled.

Note 1. For addresses in each page, see section 53, On-Chip RAM.

Note 2. When the VRAME0 bit in SYSCR1 is cleared to 0 (access to page 0 in large-capacity on-chip RAM is invalid), the on-chip data-retention RAM cannot be accessed (read and written), regardless of the setting of this bit.

When the VRAMWE0 bit in SYSCR2 is cleared to 0 (writing to page 0 in large-capacity on-chip RAM is invalid), the on-chip data-retention RAM cannot be written, regardless of the setting of this bit.

### 55.2.20 CPU Status Register (CPUSTS)

CPUSTS is an 8-bit readable register that indicates the status of the CPU.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	ISBUSY	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7 to 5	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
4	ISBUSY	0	R	State during Changing of the Frequency of CPU and Return from Software Standby Indicates that the frequency of CPU is being changed or that return from software standby is in progress. Do not execute a WFI instruction while this bit is 1. 0: The frequency of CPU is not being changed and returning from software standby is not in progress 1: The frequency of CPU is being changed or return from software standby is in progress
3 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

### 55.2.21 Standby Request Register 1 (STBREQ1)

This register is used to request notification of whether a CPU or peripheral module is ready for standby. The CPU or peripheral module returns a standby acknowledgement on receipt of a request for notification if it is ready for standby.

Note: • When writing to this register, see section 55.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	-	-	STBRQ 15	-	STBRQ 13	STBRQ 12	-	STBRQ 10
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R/W	R	R/W	R/W	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	STBRQ15	0	R/W	Standby Request to CoreSight 0: The standby request to CoreSight is invalid. 1: The standby request to CoreSight is valid.*1
4	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
3	STBRQ13	0	R/W	Standby Request to JPEG Codec Unit 0: The standby request to the JPEG codec unit is invalid. 1: The standby request to the JPEG codec unit is valid.*1
2	STBRQ12	0	R/W	Standby Request to EthernetAVB 0: The standby request to the EthernetAVB is invalid. 1: The standby request to the EthernetAVB is valid.*1
1	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
0	STBRQ10	0	R/W	Standby Request to Capture Engine Unit 0: The standby request to the capture engine unit is invalid. 1: The standby request to the capture engine unit is valid.*1

Note 1. When the MSTP bit for the corresponding module is 1, writing 1 to the STBRQ bit has no effect.

## 55.2.22 Standby Request Register 2 (STBREQ2)

This register is used to request notification of whether a peripheral module is ready for standby. The peripheral module returns a standby acknowledgement on receipt of a request for notification if it is ready for standby.

Note: • When writing to this register, see section 55.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	STBRQ 27	STBRQ 26	STBRQ 25	STBRQ 24	STBRQ 23	STBRQ 22	STBRQ 21	STBRQ 20
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7	STBRQ27	0	R/W	Standby Request to the media local bus 0: The standby request to the media local bus is invalid. 1: The standby request to the media local bus is valid.*1
6	STBRQ26	0	R/W	Standby Request to Ethernet Controller 0: The standby request to the Ethernet controller is invalid. 1: The standby request to the Ethernet controller is valid.*1
5	STBRQ25	0	R/W	Standby Request to Channel 0 of Video Display Controller 5 0: The standby request to channel 0 of the video display controller 5 is invalid. 1: The standby request to channel 0 of the video display controller 5 is valid.*1
4	STBRQ24	0	R/W	Standby Request to Channel 1 of Video Display Controller 5 0: The standby request to channel 1 of the video display controller 5 is invalid. 1: The standby request to channel 1 of the video display controller 5 is valid.*1
3	STBRQ23	0	R/W	Standby Request to Channel 0 of Image Renderer 0: The standby request to channel 0 of the image renderer is invalid. 1: The standby request to channel 0 of the image renderer is valid.*1
2	STBRQ22	0	R/W	Standby Request to Channel 1 of Image Renderer 0: The standby request to channel 1 of the image renderer is invalid. 1: The standby request to channel 1 of the image renderer is valid.*1
1	STBRQ21	0	R/W	Standby Request to Display Image Renderer 0: The standby request to the display image renderer is invalid. 1: The standby request to the display image renderer is valid.*1
0	STBRQ20	0	R/W	Standby Request to OpenVG™-Compliant Renesas Graphics Processor 0: The standby request to the OpenVG™-compliant Renesas graphics processor is invalid. 1: The standby request to the OpenVG™-compliant Renesas graphics processor is valid.*1

Note 1. When the MSTP bit for the corresponding module is 1, writing 1 to the STBRQ bit has no effect.



### 55.2.23 Standby Acknowledgment Register 1 (STBACK1)

This register is used to provide notification that a CPU or peripheral module is in the standby ready state. The CPU or peripheral module returns a standby acknowledgement on reception of a standby notification request if it is ready for standby. This register is a read-only register.

Bit:	7	6	5	4	3	2	1	0
	-	-	STBAK 15	-	STBAK 13	STBAK 12	-	STBAK 10
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	STBAK15	0	R	Standby Acknowledgement from CoreSight 0: The standby acknowledgement from CoreSight is invalid. 1: The standby acknowledgement from CoreSight is valid.*1
4	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
3	STBAK13	0	R	Standby Acknowledgement from JPEG Codec Unit 0: The standby acknowledgement from the JPEG codec unit is invalid. 1: The standby acknowledgement from the JPEG codec unit is valid.*1
2	STBAK12	0	R	Standby Acknowledgement from EthernetAVB 0: The standby acknowledgement from the EthernetAVB is invalid. 1: The standby acknowledgement from the EthernetAVB is valid.*1
1	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
0	STBAK10	0	R	Standby Acknowledgement from Capture Engine Unit 0: The standby acknowledgement from the capture engine unit is invalid. 1: The standby acknowledgement from the capture engine unit is valid.*1

Note 1. The bits in STBACK1 are set to 1 when a standby acknowledgement is transmitted from the module while the MSTP bit for the corresponding module is set to 1.

### 55.2.24 Standby Acknowledge Register 2 (STBACK2)

This register is used to provide notification that a peripheral module is in the standby ready state. The peripheral module returns a standby acknowledgement on reception of a standby notification request if it is ready for standby. This register is a read-only register.

Bit:	7	6	5	4	3	2	1	0
	STBAK 27	STBAK 26	STBAK 25	STBAK 24	STBAK 23	STBAK 22	STBAK 21	STBAK 20
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	STBAK27	0	R	Standby Acknowledgement from the media local bus 0: The standby acknowledgement from the media local bus is invalid. 1: The standby acknowledgement from the media local bus is valid.*1
6	STBAK26	0	R	Standby Acknowledgement from Ethernet Controller 0: The standby acknowledgement from the Ethernet controller is invalid. 1: The standby acknowledgement from the Ethernet controller is valid.*1
5	STBAK25	0	R	Standby Acknowledgement from Channel 0 of Video Display Controller 5 0: The standby acknowledgement from channel 0 of the video display controller 5 is invalid. 1: The standby acknowledgement from channel 0 of the video display controller 5 is valid.*1
4	STBAK24	0	R	Standby Acknowledgement from Channel 1 of Video Display Controller 5 0: The standby acknowledgement from channel 1 of the video display controller 5 is invalid. 1: The standby acknowledgement from channel 1 of the video display controller 5 is valid.*1
3	STBAK23	0	R	Standby Acknowledgement from Channel 0 of Image Renderer 0: The standby acknowledgement from channel 0 of the image renderer is invalid. 1: The standby acknowledgement from channel 0 of the image renderer is valid.*1
2	STBAK22	0	R	Standby Acknowledgement from Channel 1 of Image Renderer 0: The standby acknowledgement from channel 1 of the image renderer is invalid. 1: The standby acknowledgement from channel 1 of the image renderer is valid.*1
1	STBAK21	0	R	Standby Acknowledgement from Display Image Renderer 0: The standby acknowledgement from the display image renderer is invalid. 1: The standby acknowledgement from the display image renderer is valid.*1
0	STBAK20	0	R	Standby Acknowledgement from OpenVG™-Compliant Renesas Graphics Processor 0: The standby acknowledgement from the OpenVG™-compliant Renesas graphics processor is invalid. 1: The standby acknowledgement from the OpenVG™-compliant Renesas graphics processor is valid.*1

Note 1. The bits in STBACK2 are set to 1 when a standby acknowledgement is transmitted from the module while the MSTP bit for the corresponding module is set to 1.

### 55.2.25 On-Chip Data-Retention RAM Area Setting Register (RRAMKP)

RRAMKP is an 8-bit readable/writable register that selects whether the contents of the corresponding area of the on-chip data-retention RAM are retained or not in deep standby mode.

When the RRAMKP3 to RRAMKP0 bits are set to 1, the contents of the corresponding area of the on-chip data-retention RAM are retained in deep standby mode. When these bits are cleared to 0, the contents of the corresponding area of the on-chip data-retention RAM are not retained in deep standby mode.

Note: • When writing to this register, see section 55.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	RRAM KP3	RRAM KP2	RRAM KP1	RRAM KP0
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 4	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
3	RRAMKP3	0	R/W	On-Chip Data-Retention RAM Storage Area 3 (corresponding area: page 3*1 in on-chip data-retention RAM) 0: The contents of the on-chip data-retention RAM are not retained in deep standby mode. 1: The contents of the on-chip data-retention RAM are retained in deep standby mode.
2	RRAMKP2	0	R/W	On-Chip Data-Retention RAM Storage Area 2 (corresponding area: page 2*1 in on-chip data-retention RAM) 0: The contents of the on-chip data-retention RAM are not retained in deep standby mode. 1: The contents of the on-chip data-retention RAM are retained in deep standby mode.
1	RRAMKP1	0	R/W	On-Chip Data-Retention RAM Storage Area 1 (corresponding area: page 1*1 in on-chip data-retention RAM) 0: The contents of the on-chip data-retention RAM are not retained in deep standby mode. 1: The contents of the on-chip data-retention RAM are retained in deep standby mode.
0	RRAMKP0	0	R/W	On-Chip Data-Retention RAM Storage Area 0 (corresponding area: page 0*1 in on-chip data-retention RAM) 0: The contents of the on-chip data-retention RAM are not retained in deep standby mode. 1: The contents of the on-chip data-retention RAM are retained in deep standby mode.

Note 1. For addresses in each page, see section 53, On-Chip RAM.

### 55.2.26 Deep Standby Control Register (DSCTR)

DSCTR is an 8-bit readable/writable register that selects whether the states of the external memory control pins are retained or not when returning from deep standby mode and specifies the method to start the LSI.

Note: • When writing to this register, see section 55.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	EBUS KEEPE	RAM BOOT	-	-	-	-	-	-
Initial value:	0	0	0	0	0	0	0	0
R/W:	R/W	R/W	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
7	EBUSKEEPE	0	R/W	Retention of External Memory Control Pin State 0: The state of the external memory control pins is not retained when returning from deep standby mode. 1: The state of the external memory control pins is retained when returning from deep standby mode. Note: The following setting is prohibited: (EBUSKEEPE, RAMBOOT) = (1, 0).
6	RAMBOOT	0	R/W	Selection of Method after Returning from Deep Standby Mode Selects an activation method after returning from deep standby mode. 0: Activated according to the boot mode specified for a reset. 1: The program is read from the on-chip data-retention RAM. Instruction fetch from H'20000000 Note: The following setting is prohibited: (EBUSKEEPE, RAMBOOT) = (1, 0).
5 to 0	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.

### 55.2.27 Deep Standby Cancel Source Select Register (DSSSR)

DSSSR is a 16-bit readable/writable register that consists of the bits for selecting a source to cancel deep standby mode. The realtime clock alarm interrupt or change on the pins for canceling (P2\_12, P2\_15, P3\_1, P3\_3, P3\_9, P5\_9, P6\_2, P6\_4, P7\_8, P8\_2, P8\_7, and P9\_1) can be selected as a cancel source. The pins for canceling can be used for canceling deep standby, regardless of pin function settings in the general I/O port.

Note: • When writing to this register, see section 55.4, Usage Notes.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	P6_2	P3_9	P3_1	P2_12	P8_7	P3_3	NMI	-	RTCAR	P6_4	P5_9	P7_8	P2_15	P9_1	P8_2
lue:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
I/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	P6_2	0	R/W	Cancel by Change on P6_2 0: Deep standby mode is not canceled by change on the P6_2 pin. 1: Deep standby mode is canceled by change on the P6_2 pin.
13	P3_9	0	R/W	Cancel by Change on P3_9 0: Deep standby mode is not canceled by change on the P3_9 pin. 1: Deep standby mode is canceled by change on the P3_9 pin.
12	P3_1	0	R/W	Cancel by Change on P3_1 0: Deep standby mode is not canceled by change on the P3_1 pin. 1: Deep standby mode is canceled by change on the P3_1 pin.
11	P2_12	0	R/W	Cancel by Change on P2_12 0: Deep standby mode is not canceled by change on the P2_12 pin. 1: Deep standby mode is canceled by change on the P2_12 pin.
10	P8_7	0	R/W	Cancel by Change on P8_7 0: Deep standby mode is not canceled by change on the P8_7 pin. 1: Deep standby mode is canceled by change on the P8_7 pin.
9	P3_3	0	R/W	Cancel by Change on P3_3 0: Deep standby mode is not canceled by change on the P3_3 pin. 1: Deep standby mode is canceled by change on the P3_3 pin.
8	NMI	0	R/W	Cancel by Change on NMI 0: Deep standby mode is not canceled by change on the NMI pin. 1: Deep standby mode is canceled by change on the NMI pin.
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	RTCAR	0	R/W	Cancel by Realtime Clock Alarm Interrupt 0: Deep standby mode is not canceled by a realtime clock alarm interrupt. 1: Deep standby mode is canceled by a realtime clock alarm interrupt.
5	P6_4	0	R/W	Cancel by Change on P6_4 0: Deep standby mode is not canceled by change on the P6_4 pin. 1: Deep standby mode is canceled by change on the P6_4 pin.
4	P5_9	0	R/W	Cancel by Change on P5_9 0: Deep standby mode is not canceled by change on the P5_9 pin. 1: Deep standby mode is canceled by change on the P5_9 pin.
3	P7_8	0	R/W	Cancel by Change on P7_8 0: Deep standby mode is not canceled by change on the P7_8 pin. 1: Deep standby mode is canceled by change on the P7_8 pin.
2	P2_15	0	R/W	Cancel by Change on P2_15 0: Deep standby mode is not canceled by change on the P2_15 pin. 1: Deep standby mode is canceled by change on the P2_15 pin.
1	P9_1	0	R/W	Cancel by Change on P9_1 0: Deep standby mode is not canceled by change on the P9_1 pin. 1: Deep standby mode is canceled by change on the P9_1 pin.

---

Bit	Bit Name	Initial Value	R/W	Description
0	P8_2	0	R/W	Cancel by Change on P8_2 0: Deep standby mode is not canceled by change on the P8_2 pin. 1: Deep standby mode is canceled by change on the P8_2 pin.

---

### 55.2.28 Deep Standby Cancel Edge Select Register (DSESR)

DSESR is a 16-bit readable/writable register that consists of the bits for selecting an edge to be detected for the pin specified as a deep standby cancel source with DSSSR. This register setting is always valid for canceling deep standby, regardless of the interrupt controller setting.

Note: • When writing to this register, see section 55.4, Usage Notes.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	P6_2E	P3_9E	P3_1E	P2_12E	P8_7E	P3_3E	NMIE	-	-	P6_4E	P5_9E	P7_8E	P2_15E	P9_1E	P8_2E
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
15	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
14	P6_2E	0	R/W	P6_2 Edge Detection 0: Falling edge of P6_2 is detected. 1: Rising edge of P6_2 is detected.
13	P3_9E	0	R/W	P3_9 Edge Detection 0: Falling edge of P3_9 is detected. 1: Rising edge of P3_9 is detected.
12	P3_1E	0	R/W	P3_1 Edge Detection 0: Falling edge of P3_1 is detected. 1: Rising edge of P3_1 is detected.
11	P2_12E	0	R/W	P2_12 Edge Detection 0: Falling edge of P2_12 is detected. 1: Rising edge of P2_12 is detected.
10	P8_7E	0	R/W	P8_7 Edge Detection 0: Falling edge of P8_7 is detected. 1: Rising edge of P8_7 is detected.
9	P3_3E	0	R/W	P3_3 Edge Detection 0: Falling edge of P3_3 is detected. 1: Rising edge of P3_3 is detected.
8	NMIE	0	R/W	NMI Edge Detection 0: Falling edge of NMI is detected. 1: Rising edge of NMI is detected.
7, 6	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
5	P6_4E	0	R/W	P6_4 Edge Detection 0: Falling edge of P6_4 is detected. 1: Rising edge of P6_4 is detected.
4	P5_9E	0	R/W	P5_9 Edge Detection 0: Falling edge of P5_9 is detected. 1: Rising edge of P5_9 is detected.
3	P7_8E	0	R/W	P7_8 Edge Detection 0: Falling edge of P7_8 is detected. 1: Rising edge of P7_8 is detected.
2	P2_15E	0	R/W	P2_15 Edge Detection 0: Falling edge of P2_15 is detected. 1: Rising edge of P2_15 is detected.
1	P9_1E	0	R/W	P9_1 Edge Detection 0: Falling edge of P9_1 is detected. 1: Rising edge of P9_1 is detected.
0	P8_2E	0	R/W	P8_2 Edge Detection 0: Falling edge of P8_2 is detected. 1: Rising edge of P8_2 is detected.

### 55.2.29 Deep Standby Cancel Source Flag Register (DSFR)

DSFR is a 16-bit readable/writable register composed of two types of bits. One is the flag that confirms which source canceled deep standby mode. The other is the bit that releases the state of pins after canceling deep standby mode. When deep standby mode is canceled by an interrupt (NMI, realtime clock alarm interrupt, or change on the pins for canceling) and changes on the pins for canceling, this register retains the previous data although power-on reset exception handling is executed. When deep standby mode is canceled by a power-on reset, this register is initialized to H'0000.

All flags must be cleared immediately before transition to deep standby mode.

Note: • When writing to this register, see section 55.4, Usage Notes.

Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IO KEEP	P6_2F	P3_9F	P3_1F	P2_12F	P8_7F	P3_3F	NMIF	-	RTC ARF	P6_4F	P5_9F	P7_8F	P2_15F	P9_1F	P8_2F
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*	R/(W)*

Note: \* Only 0 can be written after reading 1 to clear the flag.

Bit	Bit Name	Initial Value	R/W	Description
15	IOKEEP	0	R/(W)*1	Release of Pin State Retention Releases the retention of the pin state after canceling deep standby mode 0: Pin state not retained [Clearing condition] • Writing 0 after reading 1 1: Pin state retained [Setting condition] • When deep standby mode is entered
14	P6_2F	0	R/(W)*1	P6_2 Flag 0: No change on the P6_2 pin 1: Change on the P6_2 pin
13	P3_9F	0	R/(W)*1	P3_9 Flag 0: No change on the P3_9 pin 1: Change on the P3_9 pin
12	P3_1F	0	R/(W)*1	P3_1 Flag 0: No change on the P3_1 pin 1: Change on the P3_1 pin
11	P2_12F	0	R/(W)*1	P2_12 Flag 0: No change on the P2_12 pin 1: Change on the P2_12 pin
10	P8_7F	0	R/(W)*1	P8_7 Flag 0: No change on the P8_7 pin 1: Change on the P8_7 pin
9	P3_3F	0	R/(W)*1	P3_3 Flag 0: No change on the P3_3 pin 1: Change on the P3_3 pin
8	NMIF	0	R/(W)*1	NMI Flag 0: No interrupt on NMI pin 1: Interrupt on NMI pin
7	—	0	R	Reserved This bit is always read as 0. The write value should always be 0.
6	RTCARF	0	R/(W)*1	RTCAR Flag 0: No realtime clock alarm interrupt generated 1: Realtime clock alarm Interrupt generated
5	P6_4F	0	R/(W)*1	P6_4 Flag 0: No change on the P6_4 pin 1: Change on the P6_4 pin



Bit	Bit Name	Initial Value	R/W	Description
4	P5_9F	0	R/(W)*1	P5_9 Flag 0: No change on the P5_9 pin 1: Change on the P5_9 pin
3	P7_8F	0	R/(W)*1	P7_8 Flag 0: No change on the P7_8 pin 1: Change on the P7_8 pin
2	P2_15F	0	R/(W)*1	P2_15 Flag 0: No change on the P2_15 pin 1: Change on the P2_15 pin
1	P9_1F	0	R/(W)*1	P9_1 Flag 0: No change on the P9_1 pin 1: Change on the P9_1 pin
0	P8_2F	0	R/(W)*1	P8_2 Flag 0: No change on the P8_2 pin 1: Change on the P8_2 pin

Note 1. Only 0 can be written after reading 1 to clear the flag.

### 55.2.30 XTAL Crystal Oscillator Gain Control Register (XTALCTR)

XTALCTR is an 8-bit readable/writable register that controls the gain of the crystal oscillator for XTAL and the realtime clock.

If the realtime clock uses the EXTAL input, the GAIN0 bit retains the previous value when software standby mode or deep standby mode is canceled by a source other than a power-on reset. If the realtime clock does not use the EXTAL input, this bit is initialized to 0 when software standby or deep standby mode is entered.

If the realtime clock uses the RTC\_X1 or RTC\_X3 input, the GAIN1 bit retains the previous value when software standby mode or deep standby mode is canceled by a source other than a power-on reset. If the realtime clock does not use the RTC\_X1 or RTC\_X3 input, this bit is initialized to 0 when software standby or deep standby mode is entered.

This bit is initialized to H'00 when software standby or deep standby mode is canceled by a power-on reset.

Note: • When writing to this register, see section 55.4, Usage Notes.

Bit:	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	GAIN1	GAIN0
Initial value:	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R/W	R/W

Bit	Bit Name	Initial Value	R/W	Description
7 to 2	—	All 0	R	Reserved These bits are always read as 0. The write value should always be 0.
1	GAIN1	0	R/W	Realtime Clock Crystal Oscillator (RTC_X3 or RTC_X4 Pin) Gain Select 0: Large gain 1: Small gain
0	GAIN0	0	R/W	XTAL Crystal Oscillator (EXTAL or XTAL Pin) Gain Select 0: Large gain 1: Small gain

## 55.3 Operation

### 55.3.1 Sleep Mode

#### (1) Transition to Sleep Mode

Executing the WFI instruction when the STBY bit in STBCR1 is 0 causes a transition from the program execution state to sleep mode. Although the CPU halts immediately after executing the WFI instruction, the contents of its internal registers remain unchanged. The on-chip peripheral modules continue to run in sleep mode. The clock output from the CKIO pin is continued.

#### (2) Canceling Sleep Mode

Sleep mode is canceled by an interrupt (NMI, IRQ, and on-chip peripheral module) or a power-on reset.

- Canceling by an interrupt  
When an NMI, IRQ, or on-chip peripheral module interrupt occurs, sleep mode is canceled and interrupt exception handling is executed. When the priority level of the generated interrupt is lower than the setting of the interrupt controller execution priority register, or the interrupt by the on-chip peripheral module is disabled on the module side, the interrupt request is not accepted and sleep mode is not canceled. When an interrupt request occurs with an interrupt masked by bits 6 (the FIQ mask bit) and 7 (the IRQ mask bit) in the CPSR register in the CPU, sleep mode is canceled but interrupt exception handling is not executed.
- Canceling by a reset  
Sleep mode is canceled by a power-on reset.

### 55.3.2 Software Standby Mode

#### (1) Transition to Software Standby Mode

The LSI switches from a program execution state to software standby mode by executing the WFI instruction when the STBY bit and DEEP bit in STBCR1 are 1 and 0 respectively. In software standby mode, not only the CPU but also the clock and on-chip peripheral modules halt. The clock output from the CKIO pin also stops.

The contents of the CPU and cache registers remain unchanged. Some registers of on-chip peripheral modules are, however, initialized. As for the states of on-chip peripheral module registers in software standby mode, see section 57.3, Register States.

The CPU takes one cycle to finish writing to STBCR1, and then executes processing for the next instruction. However, it takes one or more cycles to actually write. Therefore, execute a WFI instruction after reading STBCR1 to have the values written to STBCR1 by the CPU to be definitely reflected in the WFI instruction.

The procedure for switching to software standby mode is as follows:

1. Set the standby\_mode\_en bit of the power control register in the PL310 to 1. For the details of this register, see CoreLink Level 2 Cache Controller L2C-310 Technical Reference Manual issued by ARM Ltd.
2. Clear the TME bit in the watchdog timer control/status register (WTCSR) of the watchdog timer to 0 to stop the watchdog timer.
3. Set the timer counter (WTCNT) of the watchdog timer to 0 and the CKS[2:0] bits in WTCSR to appropriate values to secure the specified oscillation settling time.
4. After setting the STBY and DEEP bits in STBCR1 to 1 and 0 respectively, read STBCR. Then, execute a WFI instruction.

## (2) Canceling Software Standby Mode

Software standby mode is canceled by interrupts (NMI or IRQ) or a reset (power-on reset). Clock signal starts to be output from the CKIO pin.

- Canceling by an interrupt

When the falling edge or rising edge of the NMI pin (selected by the NMI edge select bit (NMIE) in the interrupt control register 0 (ICR0) of the interrupt controller) or the falling edge or rising edge of an IRQ pin (IRQ7 to IRQ0) (selected by the IRQn sense select bits (IRQn1S and IRQn0S) in interrupt control register 1 (ICR1) of the interrupt controller) is detected, clock oscillation is started. This clock pulse is supplied only to the oscillation settling counter (watchdog timer) used to count the oscillation settling time.

After the elapse of the time set in the clock select bits (CKS[2:0]) in the watchdog timer control/status register (WTCSR) of the watchdog timer before the transition to software standby mode, the watchdog timer overflow occurs. Since this overflow indicates that the clock has been stabilized, the clock pulse will be supplied to the entire chip after this overflow. Software standby mode is thus cleared and NMI interrupt exception handling (IRQ interrupt exception handling in case of IRQ) is started. If the priority level of the generated interrupt is lower than the setting of the interrupt controller execution priority register, the interrupt request is not accepted and sleep mode is entered. So, cancel software standby mode by an interrupt whose priority level is higher than the setting of the interrupt controller execution priority register.

When canceling software standby mode by the NMI interrupt or IRQ interrupt, set the CKS[2:0] bits so that the watchdog timer overflow period will be equal to or longer than the oscillation settling time.

The clock output phase of the CKIO pin may be unstable immediately after detecting an interrupt and until software standby mode is canceled.

- Canceling by a reset

When the RES pin is driven low, software standby mode is canceled and the LSI enters the power-on reset state. After that, if the RES pin is driven high, the power-on reset exception handling is started.

Keep the RES pin low until the clock oscillation settles. The internal clock will continue to be output to the CKIO pin.

## (3) Note on Transition to Software Standby Mode

Release from software standby mode is triggered by interrupts (NMI or IRQ) or a power-on reset. If, however, a WFI instruction and an interrupt other than NMI and IRQ are generated at the same time, software standby mode may be canceled due to acceptance of the interrupt.

When initiating a transition to software standby mode, make settings so that interrupts other than NMI and IRQ are not generated before execution of the WFI instruction.

## (4) Note on Canceling Software Standby Mode

After software standby mode is canceled, unstable clock pulses are output from the CKIO pin during oscillation settling time. To prevent malfunction due to the unstable pulses, the CKOEN[1:0] bits in FRQCR should be modified.

### 55.3.3 Software Standby Mode Application Example

This example describes a transition to software standby mode on the falling edge of the NMI signal, and cancellation on the rising edge of the NMI signal. The timing is shown in Figure 55.2.

When the NMI pin is changed from high to low level while the NMI edge select bit (NMIE) in the interrupt control register 0 (ICR0) is set to 0 (falling edge detection), the NMI interrupt is accepted. When the NMIE bit is set to 1 (rising edge detection) by the NMI exception service routine, the STBY and DEEP bits in STBCR1 are set to 1 and 0 respectively, and a WFI instruction is executed, software standby mode is entered. Thereafter, software standby mode is canceled when the NMI pin is changed from low to high level.

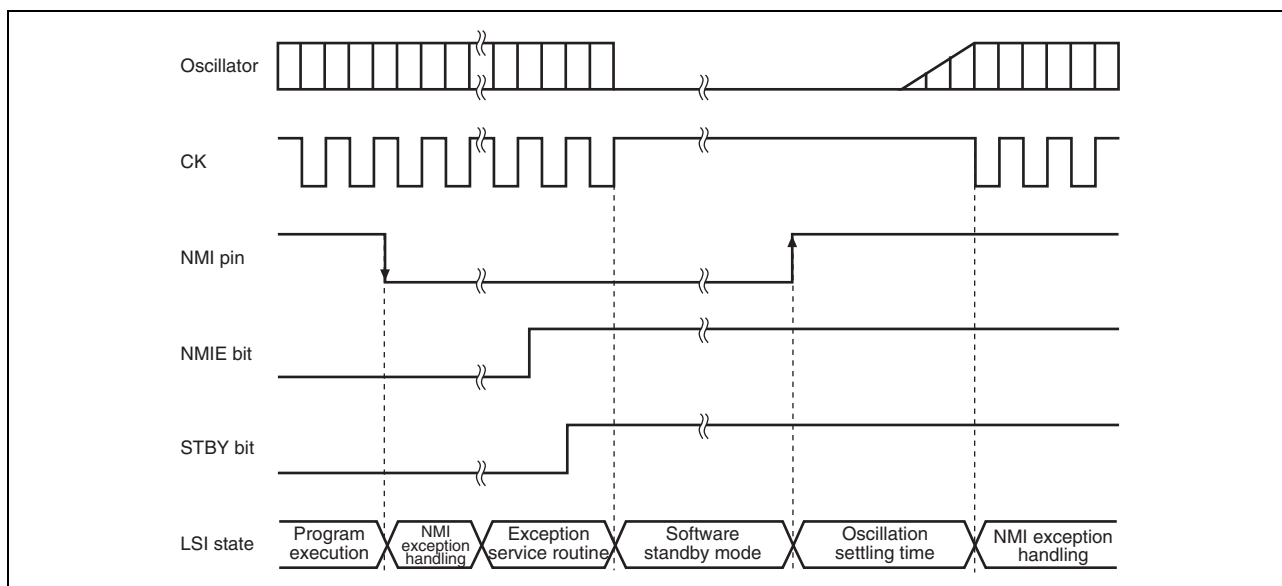


Figure 55.2 NMI Timing in Software Standby Mode (Application Example)

### 55.3.4 Deep Standby Mode

#### (1) Transition to Deep Standby Mode

The LSI switches from a program execution state to deep standby mode by executing the WFI instruction when the STBY bit and DEEP bit in STBCR1 are set to 1. In deep standby mode, not only the CPU, clocks, and on-chip peripheral modules but also power supply is turned off excluding the on-chip data-retention RAM area specified by the RRAMKP3 to RRAMKP0 bits in RRAMKP and realtime clock. This can significantly reduce power consumption. Therefore, data in the registers of the CPU, cache, and on-chip peripheral modules are not retained. Pin state values immediately before the transition to deep standby mode are retained.

The CPU takes one cycle to finish writing to STBCR1, and then executes processing for the next instruction. However, it actually takes one or more cycles to write. Therefore, execute a WFI instruction after reading STBCR1 to reflect the values written to STBCR1 by the CPU in the WFI instruction without fail.

The procedure for switching to deep standby mode is as follows. Figure 55.3 also shows its flowchart.

1. Set the `standby_mode_en` bit of the power control register in the PL310 to 1. For the details of this register, see CoreLink Level 2 Cache Controller L2C-310 Technical Reference Manual issued by ARM Ltd.
2. Set the RRAMKP3 to RRAMKP0 bits in RRAMKP for the corresponding on-chip data-retention RAM area that must be retained. Transfer the programs to be retained to the specified areas of the on-chip data-retention RAM.
3. Set the RAMBOOT and EBUSKEEPE bits in DSCTR to specify the activation method for returning from deep standby mode and to select whether the external memory control pin status is retained or not.
4. When canceling deep standby mode by an interrupt, set the corresponding bit in DSSSR to select the pin or source to cancel deep standby mode. In this case, specify the input signal detection mode for the selected pin with the

corresponding bit in DSESR.

5. Execute read and write of an arbitrary but the same address for each page in the on-chip data-retention RAM area. When this is not executed, data last written may not be written to the on-chip data-retention RAM. If there is a write to the on-chip data-retention RAM after this time, execute this processing after the last write to the on-chip data-retention RAM.
6. Set the STBY and DEEP bits in the STBCR1 register to 1, and then read this register.
7. Read out the DSFR register after clearing the flag in the DSFR register. Then execute the WFI instruction.
8. Set the CPU interface control register (ICCICR) of the interrupt controller to 0 so that an interrupt other than NMI is not notified to the CPU. Then, read the ICCICR register.
9. Execute the WFI instruction.
10. The conflict between the instruction and the NMI interrupt may prevent transition to deep standby mode. After executing the WFI instruction, locate the branch instruction to return to step 9.

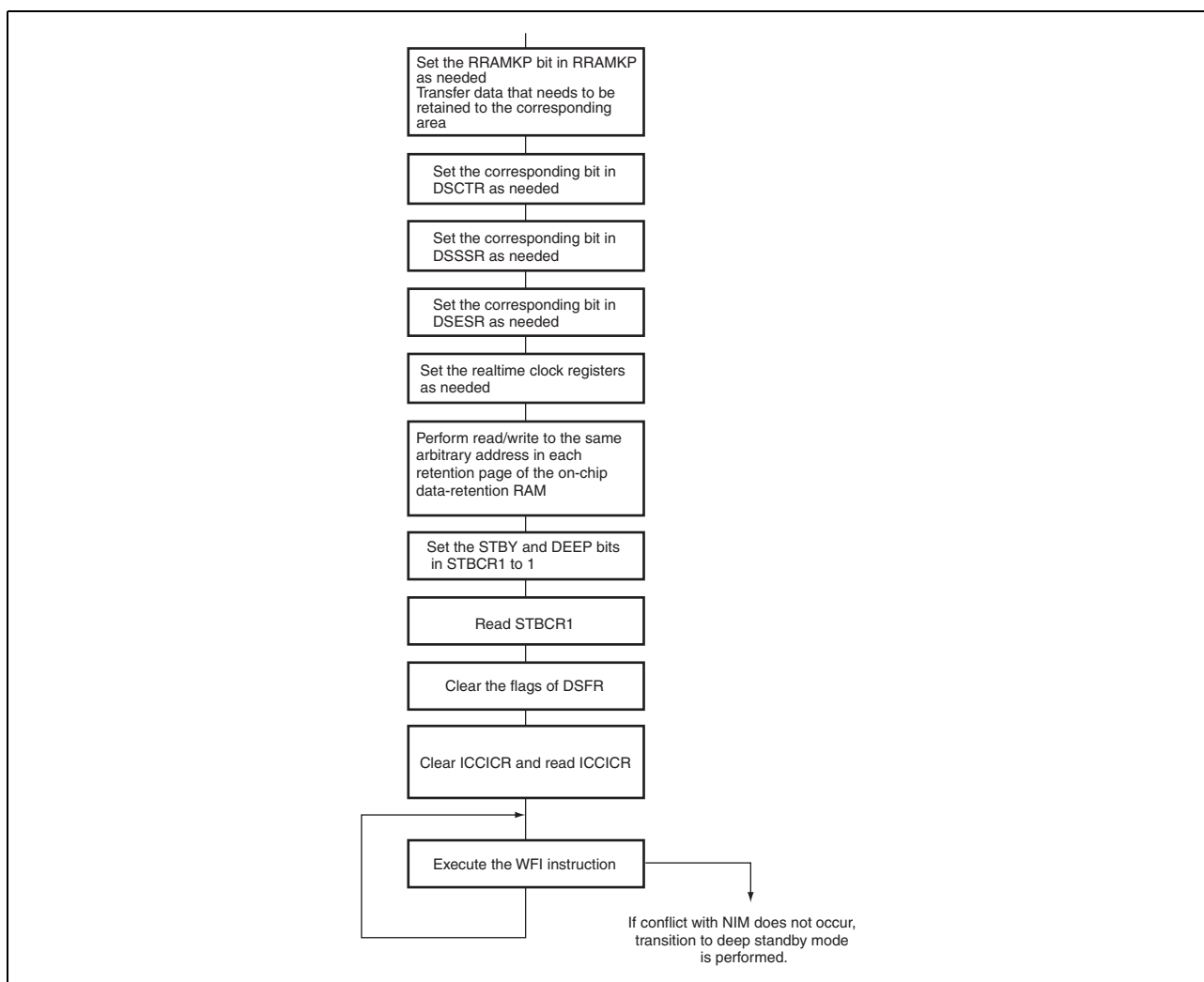
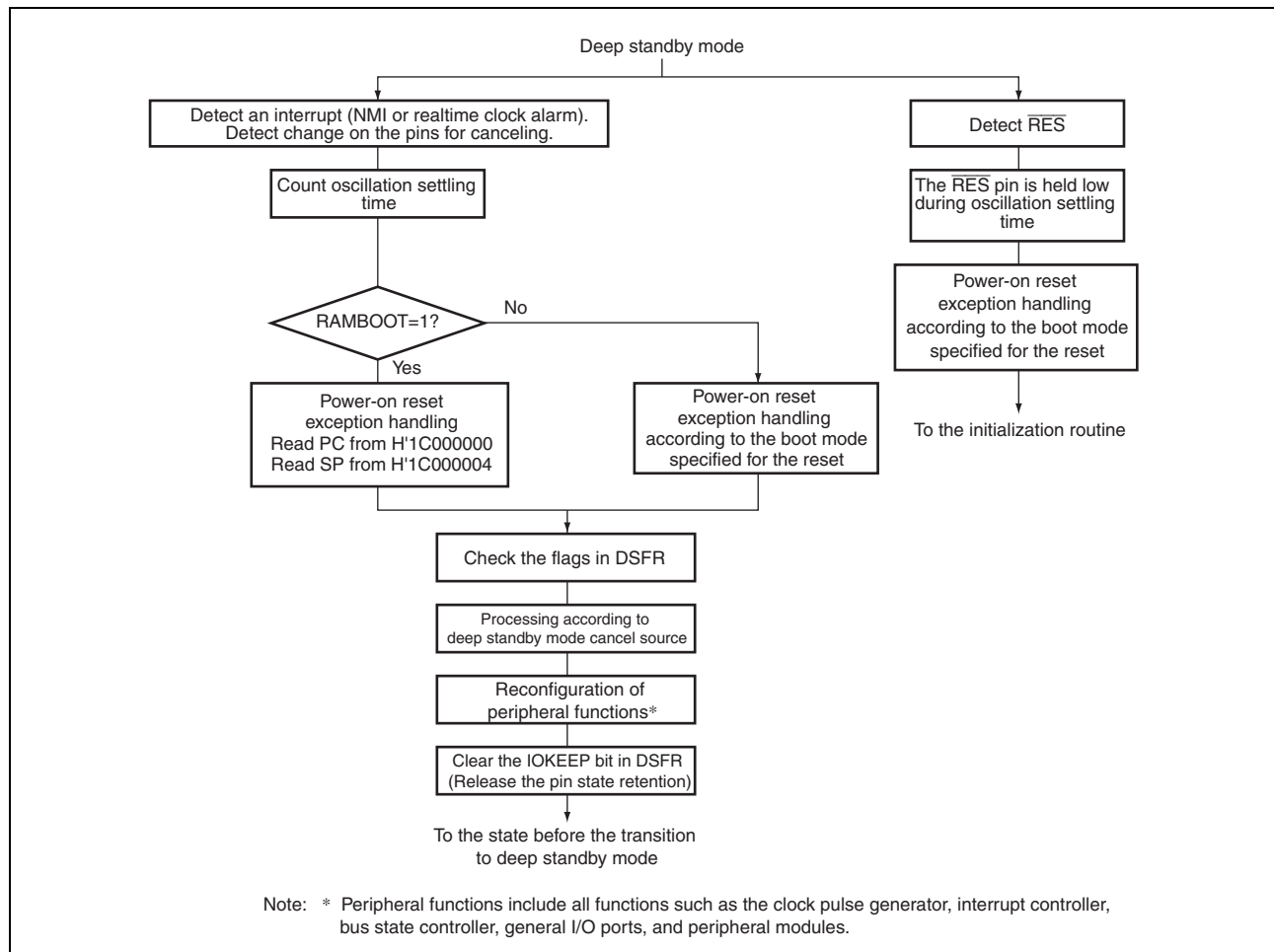


Figure 55.3 Flowchart of Transition to Deep Standby Mode

## (2) Canceling Deep Standby Mode

Deep standby mode is canceled by interrupts (NMI or realtime clock alarm interrupt), change on the pins for canceling, or a reset (power-on reset). The realtime clock alarm interrupt can always cancel deep standby mode regardless of the setting of the running priority register for the interrupt controller. This interrupt can also always cancel this mode do regardless of the states of bit 6 (the FIQ mask bit) and bit 7 (the IRQ mask bit) in the CPSR register of the CPU. When canceling the mode by a source other than a reset, a power-on reset exception handling is executed instead of an interrupt exception handling.

Figure 55.4 shows the flowchart of canceling deep standby mode.



**Figure 55.4** Flowchart of Canceling Deep Standby Mode

- Canceling by a source other than a reset

When the falling or rising edge of the NMI pin (selected by a corresponding bit in DSESR) or falling or rising edge of the pins for canceling (selected by a corresponding bit in DSESR) is detected or the realtime clock alarm interrupt (see section 13.4.4, Alarm Function) is generated, clock oscillation is started after the wait time for the oscillation settling time. After the oscillation settling time has elapsed, deep standby mode is canceled and the power-on reset exception handling is executed.

The clock output phase of the CKIO pin may be unstable immediately after detecting a cancel source and until deep standby mode is canceled.

The detecting of the NMI pin, the pins for canceling, and the realtime clock alarm interrupt becomes enable when the corresponding bits in DSSSR are set. The detected cancel sources are kept, but they are reflected to DSFR after canceling the deep standby mode. When the CPU accepts any interrupts and reads the interrupt response register (ICCIAR), all of the cancel sources that are kept are cleared. When the CPU enters the deep standby mode as the

detected cancel sources are kept, the deep standby mode is canceled immediately after the CPU enters the deep standby mode.

- Canceling with a reset

Driving the RES pin low cancels deep standby mode and causes a transition to the power-on reset state. After this, driving the RES pin high initiates power-on reset exception handling. Output of the internal clock from the CKIO pin also starts by driving the RES pin low.

Keep the RES pin low until the clock oscillation has settled.

### (3) Operation after Canceling Deep Standby Mode

After canceling deep standby mode, the LSI can be activated through the external memory or from the on-chip data-retention RAM, which can be selected by setting the RAMBOOT bit in DSCTR. By setting the EBUSKEEPE bit, the states of the external memory control pins can be retained even after cancellation of deep standby mode. Table 55.3 shows the pin states after cancellation of deep standby mode according to the setting of each bit. Table 55.4 lists the external memory control pins.

**Table 55.3 Pin States after Cancellation of Deep Standby Mode and System Activation Method by the DSCTR Settings**

EBUSKEEPE Bit	RAMBOOT Bit	Activation Method	Pin States After Cancellation of Deep Standby Mode
0	0	External memory	The states of the external memory control pins are not retained. For other pins, the retention of their states is cancelled when the IOKEEP bit is cleared.
0	1	On-chip data-retention RAM	The states of the external memory control pins are not retained. After cancellation of deep standby mode, the retention of the external memory control pin states is cancelled. For other pins, the retention of their states is cancelled when the IOKEEP bit is cleared.
1	0	—	Setting prohibited.
1	1	On-chip data-retention RAM	The states of the external memory control pin are retained. The retention of the states of the external memory control pins and other pins is cancelled when the IOKEEP bit is cleared.

**Table 55.4 External Memory Control Pins in Different Modes**

Boot Mode 0 (CS0 Area: Bus Width: 16 Bits)	Boot Mode 0 (CS0 Area: Bus Width: 32 Bits)	Boot Mode 3 (Serial Flash Boot)	Boot Mode 4 (eSD Boot)	Boot Mode 5 (eMMC Boot)
A[20:1] D[15:0] CS0, RD, CKIO	A[20:2] D[31:0] CS0, RD, CKIO	SPBCLK_0, SPBSSL_0, SPBMO0_0, SPBMO1_0	SD_CLK0, SD_CMD0, SD_D[3:0]0	MMC_CLK, MMC_CMD, MMC_D[3:0]

When deep standby mode is canceled by interrupts (NMI or realtime clock alarm) or changes on the pins for canceling, the deep standby cancel source flag register (DSFR) can be used to confirm which source has canceled the mode. Pins retain the state immediately before the transition to deep standby mode. However, in system activation through the external memory, the retention of the states of the external memory control pins is cancelled so that programs can be fetched after cancellation of deep standby mode. Other pins, after cancellation of deep standby mode, continue to retain the pin states until writing 0 to the IOKEEP bit in DSFR after reading 1 from the same bit. In system activation from the on-chip data-retention RAM, after cancellation of deep standby mode, both the external memory control pins and other pins continues to retain the pin states until writing 0 to the IOKEEP bit in DSFR after reading 1 from the same bit. Reconfiguration of peripheral functions is required to return to the previous state of deep standby mode. Peripheral functions include all functions such as the clock pulse generator, interrupt controller, general I/O ports, and peripheral modules. After the reconfiguration, the retention of the pin state can be canceled and the LSI returns to the state prior to the transition to deep standby mode by reading 1 from the IOKEEP bit in DSFR and then writing 0 to it.

#### (4) Notes on Transition to Deep Standby Mode

If multiple canceling sources have been specified and multiple canceling sources are input, multiple cancel source flags will be set.

### 55.3.5 Module Standby Function

#### (1) Transition to Module Standby Function

The procedure for transitions to the module standby state varies according to whether the STBREQ register has a bit for the corresponding module. Follow the procedures below according to whether this is or is not the case. Disable a module before placing it in the module standby state. In addition, do not access the module's registers while it is in the module standby state.

- (a) Procedure for transition to module standby of modules for which the STBREQ register does not have a corresponding bit

1. Set the MSTP bit of the corresponding module to 1.

- (b) Procedure for transition to module standby of modules for which the STBREQ register has a corresponding bit

1. Set the corresponding bit in the STBREQ register to 1 to generate a request to stop the module.
2. Confirm that the module is ready to be stopped by the corresponding bit in the STBACK register being set to 1.
3. Set the MSTP bit of the corresponding module to 1.

For states of registers in the module standby function, see section 57.3, Register States.

#### (2) Canceling Module Standby Function

Release from the module standby state can be achieved in two ways: starting the module by a power-on reset while the MSTP bit is set to 1, then clearing the MSTP bit to 0, or setting the MSTP bit to 0 to activate the module, setting the MSTP bit to 1 to place the module in the standby state, and then setting the MSTP bit to 0 again. If you use the latter approach, the procedure for release from module standby depends on whether the STBREQ register has a bit for the corresponding module.

- Release from the module standby state after the activation of the module by a power-on reset while the MSTP bit is set to 1

1. Clear the MSTP bit to 0.
2. After that, dummy-read the same register.

- Release from the module standby state after a transition to standby following activation of the module

- (a) Procedure for release from module standby of modules for which the STBREQ register does not have a corresponding bit

1. Clear the MSTP bit to 0, then dummy-read the same register.

- (b) Procedure for release from module standby of modules for which the STBREQ register has a corresponding bit

1. Clear the MSTP bit to 0, then dummy-read the same register.
2. Clear the corresponding bit in the STBREQ register to 0 to cancel the request to stop the module.
3. Confirm that the corresponding bit in the STBACK register has been cleared to 0, indicating the completion of release from standby.



### 55.3.6 Software Reset

Initialization equivalent to a power-on reset is only applied to the selected modules. The procedure for transitions to the software reset state varies according to whether the STBREQ register has a bit for the corresponding module. Follow the procedures below according to whether this is or is not the case.

#### (1) Transition to Software Reset State

- (a) Procedure for transition to the software reset state of modules for which the STBREQ register does not have a corresponding bit

1. Set the SRST bit of the corresponding module to 1.

- (b) Procedure for transition to the software reset state of modules for which the STBREQ register has a corresponding bit

1. Set the corresponding bit in the STBREQ register to 1 to generate a request to stop the module.
2. Confirm that the corresponding bit in the STBREQ register has been set to 1.
3. Set the SRST bit of the corresponding module to 1.

#### (2) Canceling Software Reset

- (a) Procedure for release from the software reset state of modules for which the STBREQ register does not have a corresponding bit

1. Clear the SRST bit of the corresponding module to 0, then dummy-read the same register.

- (b) Procedure for release from the software reset state of modules for which the STBREQ register has a corresponding bit

1. Clear the SRST bit of the corresponding module to 0, then dummy-read the same register.
2. Clear the corresponding bit in the STBREQ register to 0 to cancel the request to stop the module.
3. Confirm that the corresponding bit in the STBACK register has been cleared to 0.

### 55.3.7 Adjustment of XTAL Crystal Oscillator Gain

The gain of the crystal oscillator for XTAL and the realtime clock can be adjusted using the GAIN1 and GAIN0 bits in XTALCTR. To modify the gain of the signal on the EXTAL or XTAL pin, PLL settling time is needed. The settling time is counted using the on-chip watchdog timer. Counting to secure the PLL settling time is not needed when the gain of the signal on the RTC\_X3 or RTC\_X4 pin is changed.

1. The large gain is selected in the initial state.
2. Set the standby\_mode\_en bit of the power control register in the PL310 to 1. For the details of this register, see CoreLink Level 2 Cache Controller L2C-310 Technical Reference Manual issued by ARM Ltd.
3. Set the watchdog timer so that the specified settling time should be obtained and stop the watchdog timer. Specifically, the following settings are necessary:  
TME in WTCSR = 0: Stop the watchdog timer.  
CKS[2:0] in WTCSR: Division ratio for watchdog timer count clock  
WTCNT: Initial counter value  
(The watchdog timer starts counting on the set clock.)
4. Set the GAIN0 bit to the desired value.
5. The LSI is internally stopped and the watchdog timer starts counting. The clock is supplied only to the watchdog timer and other internal clocks are stopped. In this state, the CKIO pin continues to output an unstable clock. To avoid malfunction due to the unstable clock, modify the CKOEN2 bit in FRQCR appropriately. Since this state is equivalent to the software standby mode state, some registers of on-chip peripheral modules are initialized. For details, see section 57.3, Register States.
6. When an overflow occurs on the watchdog timer, the specified clock supply is started and the LSI starts operation. The watchdog timer stops after an overflow.

## 55.4 Usage Notes

### 55.4.1 Usage Notes on Setting Registers

When writing to the registers related to power-down modes, note the following.

When writing to the register related to power-down modes, the CPU, after executing a write instruction, executes the next instruction without waiting for the write operation to complete.

Therefore, to reflect the change specified by writing to the register while the next instruction is executed, insert a dummy read of the same register between the register write instruction and the next instruction.

### 55.4.2 Usage Notes when the Realtime Clock is not Used

When the realtime clock is not used, set the MSTP60 bit in STBCR6 to 1 after setting the bits in registers of the realtime clock. For details, see section 55.2.6, Standby Control Register 6 (STBCR6).

- Set the RTCEN bit in the control register 2 (RCR2) to 0
- Set the RCKSEL[1:0] bits in the control register 5 (RCR5) to 00.

## 56. Debugger Interface

This LSI incorporates a debugger interface to support the boundary scan function and connection to the emulator.

### 56.1 Features

The debugger interface is a serial input/output interface which has a JTAG (Joint Test Action Group, IEEE Std.1149.1 and IEEE Standard Test Access Port and Boundary-Scan Architecture) interface and CoreSight debug interface.

This module has TAP controllers for the boundary scan and CoreSight debug function.

Driving the BSCANP pin high selects the boundary-scan TAP controller, and driving the BSCANP pin low selects the CoreSight debug TAP controller.

Figure 56.1 is a block diagram of this module and Table 56.1 shows the JTAG pin mode.

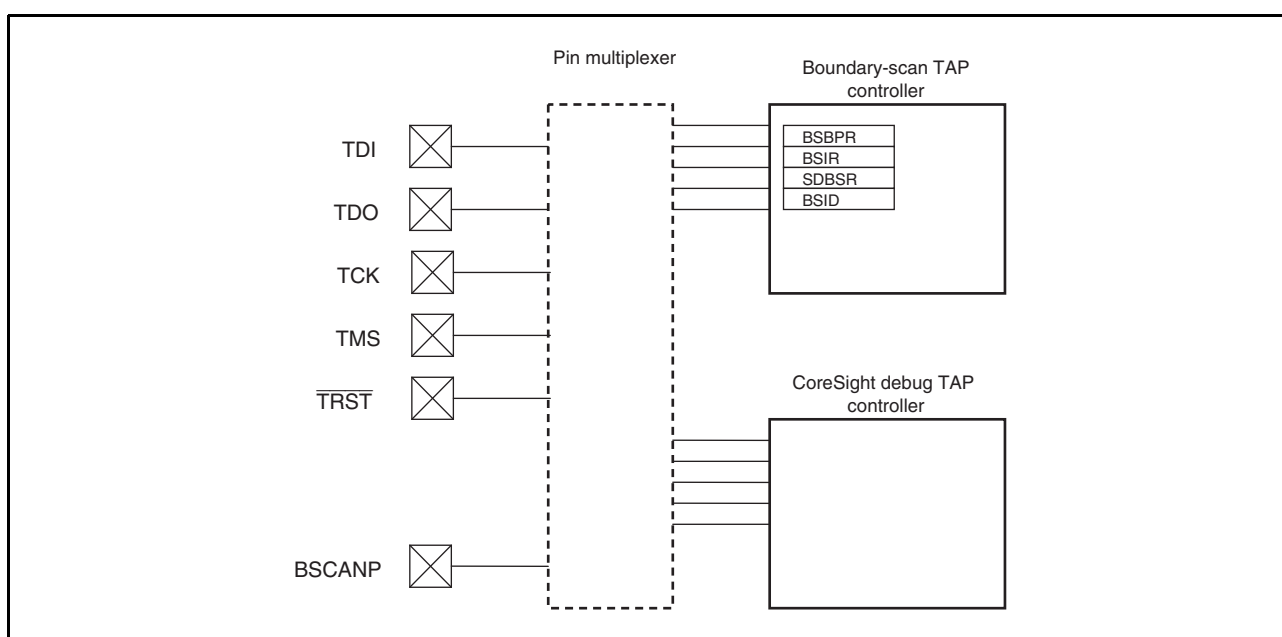


Figure 56.1 Block Diagram

Table 56.1 JTAG Pin Mode

BSCANP	JTAG Pin Mode
0	Normal operation (CoreSight debug mode)
1	Boundary-scan mode

The following lists the features of CoreSight.

- JTAG interface  
Supports JTAG and serial wire debug mode (SWD)
- Trace interface  
Outputs 4-bit × 66-Mbps (33-MHz DDR) trace data  
4-Kbyte Embedded Trace FIFO (ETF)
- Controls by ICE registers  
Control of resetting, debug-enable signal supplied to the CPU, and disabling power shut-off in deep-standby mode during debugging (fake debug mode)

Figure 56.2 is a block diagram of CoreSight, Table 56.2 to Table 56.5 show the input/output pins of the cross-trigger interface (CTI), and Table 56.6 shows the address map for CoreSight. For details of CoreSight other than ICE registers, see the technical reference manual issued by ARM Ltd.

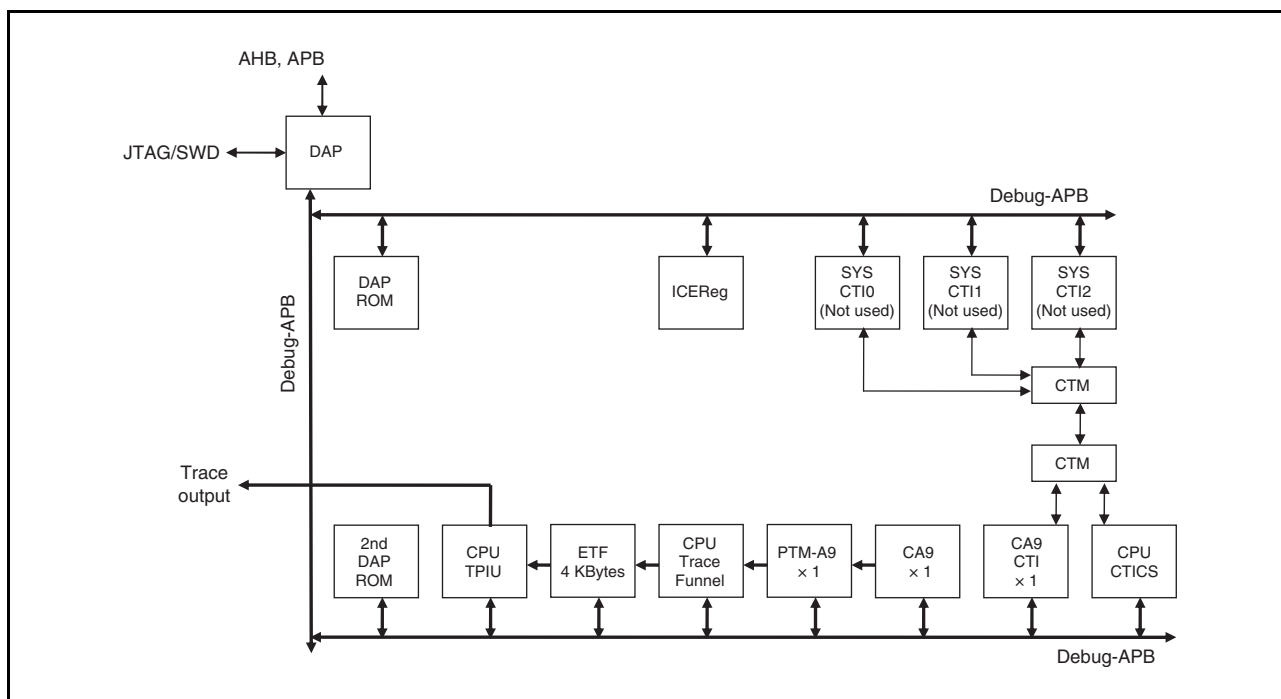


Figure 56.2 Block Diagram of CoreSight

Table 56.2 CA9 CTI Trigger Inputs

Trigger Input Bit	Trigger Signal	Source Device
[7]	Not used	—
[6]	TRIGGER	PTM-A9
[5]	COMMRX	CA9
[4]	COMMTX	CA9
[3]	EXTOUT[1]	PTM-A9
[2]	EXTOUT[0]	PTM-A9
[1]	PMUIRQ	CA9
[0]	DBGACK	CA9

Table 56.3 CA9 CTI Trigger Outputs

Trigger Input Bit	Trigger Signal	Destination Device
[7]	DBGRESTART	CA9
[6]	nCTIIRQ	INTC
[5]	Not used	—
[4]	EXTIN[3]	PTM-A9
[3]	EXTIN[2]	PTM-A9
[2]	EXTIN[1]	PTM-A9
[1]	EXTIN[0]	PTM-A9
[0]	EDBGRQ	CA9

**Table 56.4 CPU CTICS Trigger Inputs**

Trigger Input Bit	Trigger Signal	Source Device
[7]	Not used	—
[6]	Not used	—
[5]	Not used	—
[4]	Not used	—
[3]	ACQCOMP	ETF
[2]	FULL	ETF
[1]	Not used	—
[0]	Not used	—

**Table 56.5 CA9 CTI Trigger Outputs**

Trigger Input Bit	Trigger Signal	Destination Device
[7]	Not used	—
[6]	Not used	—
[5]	Not used	—
[4]	Not used	—
[3]	TRIGIN	CPU TPIU
[2]	FLUSHIN	CPU TPIU
[1]	TRIGIN	ETF
[0]	FLUSHIN	ETF

**Table 56.6 Address Map for CoreSight**

System Address (Viewed from CPU)	Debug-APB Address (Viewed from Debugger)	Module
H'FC000000 to H'FC000FFF	H'80000000 to H'80000FFF	DAP ROM
H'FC001000 to H'FC001FFF	H'80001000 to H'80001FFF	Reserved
H'FC002000 to H'FC002FFF	H'80002000 to H'80002FFF	SYS-CTI0
H'FC003000 to H'FC003FFF	H'80003000 to H'80003FFF	Reserved
H'FC004000 to H'FC004FFF	H'80004000 to H'80004FFF	Reserved
H'FC005000 to H'FC005FFF	H'80005000 to H'80005FFF	Reserved
H'FC006000 to H'FC006FFF	H'80006000 to H'80006FFF	Reserved
H'FC007000 to H'FC007FFF	H'80007000 to H'80007FFF	Reserved
H'FC008000 to H'FC008FFF	H'80008000 to H'80008FFF	SYS-CTI2
H'FC009000 to H'FC009FFF	H'80009000 to H'80009FFF	Reserved
H'FC00A000 to H'FC00AFFF	H'8000A000 to H'8000AFFF	Reserved
H'FC00B000 to H'FC00BFFF	H'8000B000 to H'8000BFFF	Reserved
H'FC00C000 to H'FC00CFFF	H'8000C000 to H'8000CFFF	SYS-CTI1
H'FC00D000 to H'FC00DFFF	H'8000D000 to H'8000DFFF	Reserved
H'FC00E000 to H'FC00EFFF	H'8000E000 to H'8000EFFF	Reserved
H'FC00F000 to H'FC00FFFF	H'8000F000 to H'8000FFFF	ICE registers (ICEReg)
H'FC010000 to H'FC010FFF	H'80010000 to H'80010FFF	Reserved
H'FC011000 to H'FC011FFF	H'80011000 to H'80011FFF	Reserved
H'FC012000 to H'FC012FFF	H'80012000 to H'80012FFF	Reserved
H'FC013000 to H'FC013FFF	H'80013000 to H'80013FFF	Reserved
H'FC014000 to H'FC014FFF	H'80014000 to H'80014FFF	Reserved
H'FC015000 to H'FC015FFF	H'80015000 to H'80015FFF	Reserved

**Table 56.6 Address Map for CoreSight**

System Address (Viewed from CPU)	Debug-APB Address (Viewed from Debugger)	Module
H'FC016000 to H'FC016FFF	H'80016000 to H'80016FFF	Reserved
H'FC017000 to H'FC017FFF	H'80017000 to H'80017FFF	Reserved
H'FC018000 to H'FC018FFF	H'80018000 to H'80018FFF	Reserved
H'FC019000 to H'FC019FFF	H'80019000 to H'80019FFF	Reserved
H'FC01A000 to H'FC01AFFF	H'8001A000 to H'8001AFFF	Reserved
H'FC01B000 to H'FC01BFFF	H'8001B000 to H'8001BFFF	Reserved
H'FC01C000 to H'FC01CFFF	H'8001C000 to H'8001CFFF	Reserved
H'FC01D000 to H'FC01DFFF	H'8001D000 to H'8001DFFF	Reserved
H'FC01E000 to H'FC01EFFF	H'8001E000 to H'8001EFFF	Reserved
H'FC01F000 to H'FC01FFFF	H'8001F000 to H'8001FFFF	Reserved
H'FC020000 to H'FC020FFF	H'80020000 to H'80020FFF	2nd DAP ROM
H'FC021000 to H'FC021FFF	H'80021000 to H'80021FFF	CPU-ETF
H'FC022000 to H'FC022FFF	H'80022000 to H'80022FFF	CPU-CTICS
H'FC023000 to H'FC023FFF	H'80023000 to H'80023FFF	CPU-TPIU
H'FC024000 to H'FC024FFF	H'80024000 to H'80024FFF	CPU-Trace Funnel
H'FC025000 to H'FC025FFF	H'80025000 to H'80025FFF	Reserved
H'FC026000 to H'FC026FFF	H'80026000 to H'80026FFF	Reserved
H'FC027000 to H'FC027FFF	H'80027000 to H'80027FFF	Reserved
H'FC028000 to H'FC028FFF	H'80028000 to H'80028FFF	Reserved
H'FC029000 to H'FC029FFF	H'80029000 to H'80029FFF	Reserved
H'FC02A000 to H'FC02AFFF	H'8002A000 to H'8002AFFF	Reserved
H'FC02B000 to H'FC02BFFF	H'8002B000 to H'8002BFFF	Reserved
H'FC02C000 to H'FC02CFFF	H'8002C000 to H'8002CFFF	Reserved
H'FC02D000 to H'FC02DFFF	H'8002D000 to H'8002DFFF	Reserved
H'FC02E000 to H'FC02EFFF	H'8002E000 to H'8002EFFF	Reserved
H'FC02F000 to H'FC02FFFF	H'8002F000 to H'8002FFFF	Reserved
H'FC030000 to H'FC030FFF	H'80030000 to H'80030FFF	CA9-DBG (CPU0)
H'FC031000 to H'FC031FFF	H'80031000 to H'80031FFF	CA9-PMU (CPU0)
H'FC032000 to H'FC032FFF	H'80032000 to H'80032FFF	Reserved
H'FC033000 to H'FC033FFF	H'80033000 to H'80033FFF	Reserved
H'FC034000 to H'FC034FFF	H'80034000 to H'80034FFF	Reserved
H'FC035000 to H'FC035FFF	H'80035000 to H'80035FFF	Reserved
H'FC036000 to H'FC036FFF	H'80036000 to H'80036FFF	Reserved
H'FC037000 to H'FC037FFF	H'80037000 to H'80037FFF	Reserved
H'FC038000 to H'FC038FFF	H'80038000 to H'80038FFF	CA9 CTI (CPU0)
H'FC039000 to H'FC039FFF	H'80039000 to H'80039FFF	Reserved
H'FC03A000 to H'FC03AFFF	H'8003A000 to H'8003AFFF	Reserved
H'FC03B000 to H'FC03BFFF	H'8003B000 to H'8003BFFF	Reserved
H'FC03C000 to H'FC03CFFF	H'8003C000 to H'8003CFFF	PTM-A9 (CPU0)
H'FC03D000 to H'FC03DFFF	H'8003D000 to H'8003DFFF	Reserved
H'FC03E000 to H'FC03EFFF	H'8003E000 to H'8003EFFF	Reserved
H'FC03F000 to H'FC03FFFF	H'8003F000 to H'8003FFFF	Reserved

## 56.2 Input/Output Pins

Table 56.7 shows the pin configuration of the debugger interface.

**Table 56.7 Pin Configuration**

Name	Pin Name	Input/ Output	Function
Test clock	TCK/SWDCLK	Input	Data is input in serial to this module via the data input pin (TDI) or data is output via the data output pin (TDO) in synchronization with this signal. Functions as the SWDCLK pin in serial wire debug (SWD) mode.
Test mode select	TMS/SWDIO	Input, Input/ Output	Changing this signal in synchronization with the TCK signal determines the state of the TAP controller circuit. Its protocol conforms to the subset of the JTAG standard (IEEE standard 1149.1). Functions as the SWDIO pin in serial wire debug (SWD) mode.
Test reset	$\overline{\text{TRST}}^*$	Input	This signal is received asynchronously with the TCK signal. Asserting this signal resets this module. When power is supplied, this pin should be asserted for a given period regardless of whether or not this module function is used. For details on resetting, see section 56.5.2, Reset Signal Setting.
Test data input	TDI	Input	Data is sent to this module by changing this signal in synchronization with the TCK signal.
Test data output	TDO	Output	Data is read from this module by reading this signal in synchronization with the TCK signal.
Boundary scan setting	BSCANP	Input	Input a high-level signal during boundary-scan testing. Input a low-level signal during normal use.
Clock output	TRACECLK	Output	Trace clock output pin
Enable output	TRACECTL	Output	Trace enable output pin
Data output	TRACEDAT3 to TRACEDATA0	Output	Trace data output pins

Note: \* When designing a board that can use an emulator, the  $\overline{\text{TRST}}$  pin circuit should be designed so that it can be controlled independently and can be set low during the period the  $\overline{\text{RES}}$  pin is asserted low at power-on. When the  $\overline{\text{TRST}}$  pin is not in use, it should be either fixed to low level or connected to the  $\overline{\text{RES}}$  pin (or another pin which operates in the same manner as the  $\overline{\text{RES}}$  pin).



### 56.3 Registers for Boundary-Scan TAP Controller

The boundary-scan TAP controller has the following registers.

**Table 56.8 List of Registers of Boundary-Scan TAP Controller**

Register Name	Abbreviation	R/W	Initial Value	Address	Access Size
Bypass register	BSBPR	—	—	—	—
Instruction register	BSIR	—	H'55	—	—
Boundary scan register	SDBSR	—	—	—	—
ID register	BSID	—	H'08178447	—	—

#### 56.3.1 Bypass Register (BSBPR)

BSBPR is a 1-bit register that cannot be accessed by the CPU. When BSIR is set to BYPASS mode, BSBPR is connected between the TDI and TDO pins. The initial value is undefined.

#### 56.3.2 Instruction Register (BSIR)

BSIR is an 8-bit register. This register is initialized by  $\overline{\text{TRST}}$  assertion or in the TAP test-logic-reset state. BSIR cannot be accessed by the CPU.

Bit	Bit Name	Initial Value	R/W	Description
7 to 0	T[7:0]	01010101	—	Test Instruction The instruction of this module is transferred to BSIR through a serial input from the TDI pin. For commands, see Table 56.9.

**Table 56.9 Commands Supported for Boundary Scan TAP Controller**

Bits 7 to 0								Description
T17	T16	T15	T14	T13	T12	T11	T10	
0	0	0	0	0	0	0	0	EXTEST
0	1	0	0	0	0	0	0	SAMPLE/PRELOAD
0	1	0	1	0	1	0	1	IDCODE (initial value)
1	1	0	1	0	0	0	0	CLAMP
1	0	0	0	0	0	0	0	HIGHZ
1	1	1	1	1	1	1	1	BYPASS
Other than above								Reserved

### 56.3.3 Boundary Scan Register (SDBSR)

SDBSR is a shift register, located on the PAD, for controlling the input/output pins of this LSI. SDBSR cannot be accessed by the CPU. The initial value is undefined. Using the EXTEST, SAMPLE/PRELOAD, CLAMP, and HIGHZ commands, a boundary scan test conforming to the JTAG standard can be carried out. Table 56.10 shows the correspondence between the pins of this LSI and bits of the boundary scan register.

**Table 56.10 Correspondence between Pins of this LSI and Bits of Boundary Scan Registers**

324-Pin Bit Number	256-Pin Bit Number	Pin Name*1	Type	324-Pin Bit Number	256-Pin Bit Number	Pin Name*1	Type
From TDI				434	-	P10_6	OUTPUT
470	374	P3_10	OUTPUT	433	-	P10_6	CONTROL
469	373	P3_10	CONTROL	432	-	P10_6	INPUT
468	372	P3_10	INPUT	431	-	P10_7	OUTPUT
467	371	P3_11	OUTPUT	430	-	P10_7	CONTROL
466	370	P3_11	CONTROL	429	-	P10_7	INPUT
465	369	P3_11	INPUT	428	344	P4_4	OUTPUT
464	368	P3_12	OUTPUT	427	343	P4_4	CONTROL
463	367	P3_12	CONTROL	426	342	P4_4	INPUT
462	366	P3_12	INPUT	425	341	P4_5	OUTPUT
461	365	P3_13	OUTPUT	424	340	P4_5	CONTROL
460	364	P3_13	CONTROL	423	339	P4_5	INPUT
459	363	P3_13	INPUT	422	338	P4_6	OUTPUT
458	362	P3_14	OUTPUT	421	337	P4_6	CONTROL
457	361	P3_14	CONTROL	420	336	P4_6	INPUT
456	360	P3_14	INPUT	419	335	P4_7	OUTPUT
455	359	P3_15	OUTPUT	418	334	P4_7	CONTROL
454	358	P3_15	CONTROL	417	333	P4_7	INPUT
453	357	P3_15	INPUT	416	332	P2_0	OUTPUT
452	356	P4_0	OUTPUT	415	331	P2_0	CONTROL
451	355	P4_0	CONTROL	414	330	P2_0	INPUT
450	354	P4_0	INPUT	413	329	P2_1	OUTPUT
449	353	P4_1	OUTPUT	412	328	P2_1	CONTROL
448	352	P4_1	CONTROL	411	327	P2_1	INPUT
447	351	P4_1	INPUT	410	326	P4_8	OUTPUT
446	350	P4_2	OUTPUT	409	325	P4_8	CONTROL
445	349	P4_2	CONTROL	408	324	P4_8	INPUT
444	348	P4_2	INPUT	407	323	P4_9	OUTPUT
443	-	P10_4	OUTPUT	406	322	P4_9	CONTROL
442	-	P10_4	CONTROL	405	321	P4_9	INPUT
441	-	P10_4	INPUT	404	320	P4_10	OUTPUT
440	347	P4_3	OUTPUT	403	319	P4_10	CONTROL
439	346	P4_3	CONTROL	402	318	P4_10	INPUT
438	345	P4_3	INPUT	401	-	P10_8	OUTPUT
437	-	P10_5	OUTPUT	400	-	P10_8	CONTROL
436	-	P10_5	CONTROL	399	-	P10_8	INPUT
435	-	P10_5	INPUT	398	-	P10_9	OUTPUT

**Table 56.10 Correspondence between Pins of this LSI and Bits of Boundary Scan Registers**

324-Pin Bit Number	256-Pin Bit Number	Pin Name*1	Type	324-Pin Bit Number	256-Pin Bit Number	Pin Name*1	Type
397	-	P10_9	CONTROL	352	-	P10_15	CONTROL
396	-	P10_9	INPUT	351	-	P10_15	INPUT
395	-	P10_10	OUTPUT	350	290	P2_6	OUTPUT
394	-	P10_10	CONTROL	349	289	P2_6	CONTROL
393	-	P10_10	INPUT	348	288	P2_6	INPUT
392	-	P10_11	OUTPUT	347	287	P2_7	OUTPUT
391	-	P10_11	CONTROL	346	286	P2_7	CONTROL
390	-	P10_11	INPUT	345	285	P2_7	INPUT
389	317	P4_11	OUTPUT	344	284	P2_8	OUTPUT
388	316	P4_11	CONTROL	343	283	P2_8	CONTROL
387	315	P4_11	INPUT	342	282	P2_8	INPUT
386	314	P4_12	OUTPUT	341	281	P2_9	OUTPUT
385	313	P4_12	CONTROL	340	280	P2_9	CONTROL
384	312	P4_12	INPUT	339	279	P2_9	INPUT
383	311	P4_13	OUTPUT	338	278	P2_10	OUTPUT
382	310	P4_13	CONTROL	337	277	P2_10	CONTROL
381	309	P4_13	INPUT	336	276	P2_10	INPUT
380	308	P4_14	OUTPUT	335	275	P2_11	OUTPUT
379	307	P4_14	CONTROL	334	274	P2_11	CONTROL
378	306	P4_14	INPUT	333	273	P2_11	INPUT
377	305	P4_15	OUTPUT	332	272	P2_12	OUTPUT
376	304	P4_15	CONTROL	331	271	P2_12	CONTROL
375	303	P4_15	INPUT	330	270	P2_12	INPUT
374	302	P2_2	OUTPUT	329	269	P2_13	OUTPUT
373	301	P2_2	CONTROL	328	268	P2_13	CONTROL
372	300	P2_2	INPUT	327	267	P2_13	INPUT
371	299	P2_3	OUTPUT	326	266	P2_14	OUTPUT
370	298	P2_3	CONTROL	325	265	P2_14	CONTROL
369	297	P2_3	INPUT	324	264	P2_14	INPUT
368	296	P2_4	OUTPUT	323	263	P2_15	OUTPUT
367	295	P2_4	CONTROL	322	262	P2_15	CONTROL
366	294	P2_4	INPUT	321	261	P2_15	INPUT
365	293	P2_5	OUTPUT	320	260	P1_0	OUTPUT
364	292	P2_5	CONTROL	319	259	P1_0	INPUT
363	291	P2_5	INPUT	318	258	P1_1	OUTPUT
362	-	P10_12	OUTPUT	317	257	P1_1	INPUT
361	-	P10_12	CONTROL	316	256	P1_2	OUTPUT
360	-	P10_12	INPUT	315	255	P1_2	INPUT
359	-	P10_13	OUTPUT	314	254	P1_3	OUTPUT
358	-	P10_13	CONTROL	313	253	P1_3	INPUT
357	-	P10_13	INPUT	312	252	P1_4	OUTPUT
356	-	P10_14	OUTPUT	311	251	P1_4	INPUT
355	-	P10_14	CONTROL	310	250	P1_5	OUTPUT
354	-	P10_14	INPUT	309	249	P1_5	INPUT
353	-	P10_15	OUTPUT	308	248	P1_6	OUTPUT

Table 56.10 Correspondence between Pins of this LSI and Bits of Boundary Scan Registers

324-Pin Bit Number	256-Pin Bit Number	Pin Name*1	Type	324-Pin Bit Number	256-Pin Bit Number	Pin Name*1	Type
307	247	P1_6	INPUT	262	202	P9_4	INPUT
306	246	P1_7	OUTPUT	261	201	P9_5	OUTPUT
305	245	P1_7	INPUT	260	200	P9_5	CONTROL
304	244	P0_2	INPUT	259	199	P9_5	INPUT
303	243	P5_0	OUTPUT	258	198	P9_6	OUTPUT
302	242	P5_0	CONTROL	257	197	P9_6	CONTROL
301	241	P5_0	INPUT	256	196	P9_6	INPUT
300	240	P5_1	OUTPUT	255	195	P9_7	OUTPUT
299	239	P5_1	CONTROL	254	194	P9_7	CONTROL
298	238	P5_1	INPUT	253	193	P9_7	INPUT
297	237	P5_2	OUTPUT	252	-	P11_0	OUTPUT
296	236	P5_2	CONTROL	251	-	P11_0	CONTROL
295	235	P5_2	INPUT	250	-	P11_0	INPUT
294	234	P5_3	OUTPUT	249	-	P11_1	OUTPUT
293	233	P5_3	CONTROL	248	-	P11_1	CONTROL
292	232	P5_3	INPUT	247	-	P11_1	INPUT
291	231	P5_4	OUTPUT	246	-	P11_2	OUTPUT
290	230	P5_4	CONTROL	245	-	P11_2	CONTROL
289	229	P5_4	INPUT	244	-	P11_2	INPUT
288	228	P5_5	OUTPUT	243	-	P11_3	OUTPUT
287	227	P5_5	CONTROL	242	-	P11_3	CONTROL
286	226	P5_5	INPUT	241	-	P11_3	INPUT
285	225	P5_6	OUTPUT	240	192	P6_0	OUTPUT
284	224	P5_6	CONTROL	239	191	P6_0	CONTROL
283	223	P5_6	INPUT	238	190	P6_0	INPUT
282	222	P5_7	OUTPUT	237	189	P6_1	OUTPUT
281	221	P5_7	CONTROL	236	188	P6_1	CONTROL
280	220	P5_7	INPUT	235	187	P6_1	INPUT
279	219	P5_8	OUTPUT	234	186	P6_2	OUTPUT
278	218	P5_8	CONTROL	233	185	P6_2	CONTROL
277	217	P5_8	INPUT	232	184	P6_2	INPUT
276	216	P5_9	OUTPUT	231	183	P6_3	OUTPUT
275	215	P5_9	CONTROL	230	182	P6_3	CONTROL
274	214	P5_9	INPUT	229	181	P6_3	INPUT
273	213	P5_10	OUTPUT	228	180	P6_4	OUTPUT
272	212	P5_10	CONTROL	227	179	P6_4	CONTROL
271	211	P5_10	INPUT	226	178	P6_4	INPUT
270	210	P9_2	OUTPUT	225	177	P6_5	OUTPUT
269	209	P9_2	CONTROL	224	176	P6_5	CONTROL
268	208	P9_2	INPUT	223	175	P6_5	INPUT
267	207	P9_3	OUTPUT	222	174	P6_6	OUTPUT
266	206	P9_3	CONTROL	221	173	P6_6	CONTROL
265	205	P9_3	INPUT	220	172	P6_6	INPUT
264	204	P9_4	OUTPUT	219	171	P6_7	OUTPUT
263	203	P9_4	CONTROL	218	170	P6_7	CONTROL

**Table 56.10 Correspondence between Pins of this LSI and Bits of Boundary Scan Registers**

324-Pin Bit Number	256-Pin Bit Number	Pin Name*1	Type	324-Pin Bit Number	256-Pin Bit Number	Pin Name*1	Type
217	169	P6_7	INPUT	172	136	P7_2	INPUT
216	168	P6_8	OUTPUT	171	135	P7_3	OUTPUT
215	167	P6_8	CONTROL	170	134	P7_3	CONTROL
214	166	P6_8	INPUT	169	133	P7_3	INPUT
213	165	P6_9	OUTPUT	168	132	P7_4	OUTPUT
212	164	P6_9	CONTROL	167	131	P7_4	CONTROL
211	163	P6_9	INPUT	166	130	P7_4	INPUT
210	162	P6_10	OUTPUT	165	129	P7_5	OUTPUT
209	161	P6_10	CONTROL	164	128	P7_5	CONTROL
208	160	P6_10	INPUT	163	127	P7_5	INPUT
207	159	P6_11	OUTPUT	162	126	P7_6	OUTPUT
206	158	P6_11	CONTROL	161	125	P7_6	CONTROL
205	157	P6_11	INPUT	160	124	P7_6	INPUT
204	156	P6_12	OUTPUT	159	123	P7_7	OUTPUT
203	155	P6_12	CONTROL	158	122	P7_7	CONTROL
202	154	P6_12	INPUT	157	121	P7_7	INPUT
201	153	P6_13	OUTPUT	156	120	P7_8	OUTPUT
200	152	P6_13	CONTROL	155	119	P7_8	CONTROL
199	151	P6_13	INPUT	154	118	P7_8	INPUT
198	150	P6_14	OUTPUT	153	117	P7_9	OUTPUT
197	149	P6_14	CONTROL	152	116	P7_9	CONTROL
196	148	P6_14	INPUT	151	115	P7_9	INPUT
195	147	P6_15	OUTPUT	150	114	P7_10	OUTPUT
194	146	P6_15	CONTROL	149	113	P7_10	CONTROL
193	145	P6_15	INPUT	148	112	P7_10	INPUT
192	144	P7_0	OUTPUT	147	111	P7_11	OUTPUT
191	143	P7_0	CONTROL	146	110	P7_11	CONTROL
190	142	P7_0	INPUT	145	109	P7_11	INPUT
189	-	P11_12	OUTPUT	144	-	P11_4	OUTPUT
188	-	P11_12	CONTROL	143	-	P11_4	CONTROL
187	-	P11_12	INPUT	142	-	P11_4	INPUT
186	-	P11_13	OUTPUT	141	-	P11_5	OUTPUT
185	-	P11_13	CONTROL	140	-	P11_5	CONTROL
184	-	P11_13	INPUT	139	-	P11_5	INPUT
183	-	P11_14	OUTPUT	138	108	P7_12	OUTPUT
182	-	P11_14	CONTROL	137	107	P7_12	CONTROL
181	-	P11_14	INPUT	136	106	P7_12	INPUT
180	-	P11_15	OUTPUT	135	-	P11_6	OUTPUT
179	-	P11_15	CONTROL	134	-	P11_6	CONTROL
178	-	P11_15	INPUT	133	-	P11_6	INPUT
177	141	P7_1	OUTPUT	132	-	P11_7	OUTPUT
176	140	P7_1	CONTROL	131	-	P11_7	CONTROL
175	139	P7_1	INPUT	130	-	P11_7	INPUT
174	138	P7_2	OUTPUT	129	105	P7_13	OUTPUT
173	137	P7_2	CONTROL	128	104	P7_13	CONTROL

**Table 56.10 Correspondence between Pins of this LSI and Bits of Boundary Scan Registers**

324-Pin Bit Number	256-Pin Bit Number	Pin Name*1	Type	324-Pin Bit Number	256-Pin Bit Number	Pin Name*1	Type
127	103	P7_13	INPUT	82	70	P8_8	INPUT
126	102	P7_14	OUTPUT	81	69	P8_9	OUTPUT
125	101	P7_14	CONTROL	80	68	P8_9	CONTROL
124	100	P7_14	INPUT	79	67	P8_9	INPUT
123	99	P7_15	OUTPUT	78	66	P8_10	OUTPUT
122	98	P7_15	CONTROL	77	65	P8_10	CONTROL
121	97	P7_15	INPUT	76	64	P8_10	INPUT
120	96	P8_0	OUTPUT	75	63	P8_11	OUTPUT
119	95	P8_0	CONTROL	74	62	P8_11	CONTROL
118	94	P8_0	INPUT	73	61	P8_11	INPUT
117	93	P8_1	OUTPUT	72	60	P8_12	OUTPUT
116	92	P8_1	CONTROL	71	59	P8_12	CONTROL
115	91	P8_1	INPUT	70	58	P8_12	INPUT
114	90	P8_2	OUTPUT	69	57	P8_13	OUTPUT
113	89	P8_2	CONTROL	68	56	P8_13	CONTROL
112	88	P8_2	INPUT	67	55	P8_13	INPUT
111	87	P8_3	OUTPUT	66	54	P8_14	OUTPUT
110	86	P8_3	CONTROL	65	53	P8_14	CONTROL
109	85	P8_3	INPUT	64	52	P8_14	INPUT
108	84	P8_4	OUTPUT	63	51	P8_15	OUTPUT
107	83	P8_4	CONTROL	62	50	P8_15	CONTROL
106	82	P8_4	INPUT	61	49	P8_15	INPUT
105	81	P8_5	OUTPUT	60	48	P9_0	OUTPUT
104	80	P8_5	CONTROL	59	47	P9_0	CONTROL
103	79	P8_5	INPUT	58	46	P9_0	INPUT
102	-	P11_8	OUTPUT	57	45	P9_1	OUTPUT
101	-	P11_8	CONTROL	56	44	P9_1	CONTROL
100	-	P11_8	INPUT	55	43	P9_1	INPUT
99	-	P11_9	OUTPUT	54	42	P3_7	OUTPUT
98	-	P11_9	CONTROL	53	41	P3_7	CONTROL
97	-	P11_9	INPUT	52	40	P3_7	INPUT
96	-	P11_10	OUTPUT	51	39	P3_6	OUTPUT
95	-	P11_10	CONTROL	50	38	P3_6	CONTROL
94	-	P11_10	INPUT	49	37	P3_6	INPUT
93	78	P8_6	OUTPUT	48	36	P3_5	OUTPUT
92	77	P8_6	CONTROL	47	35	P3_5	CONTROL
91	76	P8_6	INPUT	46	34	P3_5	INPUT
90	-	P11_11	OUTPUT	45	33	P3_4	OUTPUT
89	-	P11_11	CONTROL	44	32	P3_4	CONTROL
88	-	P11_11	INPUT	43	31	P3_4	INPUT
87	75	P8_7	OUTPUT	42	-	P10_0	OUTPUT
86	74	P8_7	CONTROL	41	-	P10_0	CONTROL
85	73	P8_7	INPUT	40	-	P10_0	INPUT
84	72	P8_8	OUTPUT	39	-	P10_1	OUTPUT
83	71	P8_8	CONTROL	38	-	P10_1	CONTROL

**Table 56.10 Correspondence between Pins of this LSI and Bits of Boundary Scan Registers**

324-Pin Bit Number	256-Pin Bit Number	Pin Name*1	Type	324-Pin Bit Number	256-Pin Bit Number	Pin Name*1	Type
37	-	P10_1	INPUT	18	18	NMI	INPUT
36	-	P10_2	OUTPUT	17	17	P0_4	INPUT
35	-	P10_2	CONTROL	16	16	P0_5	INPUT
34	-	P10_2	INPUT	15	15	P0_0	INPUT
33	-	P10_3	OUTPUT	14	14	P0_1	INPUT
32	-	P10_3	CONTROL	13	13	P1_8	INPUT
31	-	P10_3	INPUT	12	12	P1_9	INPUT
30	30	P3_3	OUTPUT	11	11	P1_10	INPUT
29	29	P3_3	CONTROL	10	10	P1_11	INPUT
28	28	P3_3	INPUT	9	9	P1_12	INPUT
27	27	P3_2	OUTPUT	8	8	P1_13	INPUT
26	26	P3_2	CONTROL	7	7	P1_14	INPUT
25	25	P3_2	INPUT	6	6	P1_15	INPUT
24	24	P3_1	OUTPUT	5	5	P3_8	OUTPUT
23	23	P3_1	CONTROL	4	4	P3_8	CONTROL
22	22	P3_1	INPUT	3	3	P3_8	INPUT
21	21	P3_0	OUTPUT	2	2	P3_9	OUTPUT
20	20	P3_0	CONTROL	1	1	P3_9	CONTROL
19	19	P3_0	INPUT	0	0	P3_9	INPUT

Note 1. The pin names are as listed in the "Port function/dedicated function" column of Table 1.4, List of Pins in section 1.6, List of Pins.

Note 2. Open-drain pin. Setting the data bit for the pin to 0 makes the pin output a low-level signal and setting the data bit for the pin to 1 places the pin in the hi-Z state.

Note 3. Bits of type "control" are active low. When the control bit is at the low level, the corresponding pin becomes an output.

### 56.3.4 ID Register (BSID)

BSID is a 32-bit read-only register that cannot be accessed by the CPU. BSID can only be read from the pins when the IDCODE command is set.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DID[31:16]															
Initial value:	0	0	0	0	1	0	0	0	0	0	0	1	0	1	1	1
R/W:	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DID[15:0]															
Initial value:	1	0	0	0	0	1	0	0	0	1	0	0	0	1	1	1
R/W:	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	DID[31:0]	H'08178447	—	Device IDCODE This is the ID register stipulated by JTAG. The initial value specific to this LSI is H'08178447. Note that the higher-order four bits may be changed depending on the version of the chip.

## 56.4 ICE Registers

The debugger interface has the following ICE registers.

**Table 56.11 Configuration of ICE Registers**

Register Name	Abbreviation	R/W	Initial Value	Address		Access Size
				Viewed from CPU	Viewed from Debugger	
Mode reset control register	ICEREGMDRSTCTL	R/W	H'0000FF1E	H'FC00F000	H'8000F000	32
JTAG trace select register	ICEREGJTTRCSEL	R/W	H'00800000	H'FC00F004	H'8000F004	32
Clock power control register	ICEREGCLKPWRCTRL	R/W	H'00000000	H'FC00F014	H'8000F014	32
Lock access register	ICEREGLOCKACCES	W	—	H'FC00FFB0	H'8000FFB0	32

### 56.4.1 Mode Reset Control Register (ICEREGMDRSTCTL)

ICEREGMDRSTCTL is used to set the debug mode and to perform software reset.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	NIDEN_CPU0	—	—	—	DBGEN_CPU0
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R	R/W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	RSTRB_CPU0_DERSTZ	—	—	—	RSTRB_CPU0_CPURSTZ	—	—	—	RSTRB_SYS_SYSRSTZ	—	RSTRB_CPU_PRSTD_BGZ	RSTRB_CPU_SYSRSTZ	—
Initial value:	1	1	1	1	1	1	1	0	0	0	0	1	1	1	1	0
R/W:	R	R	R	R/W	R	R	R	R/W	R	R	R	R/W	R/W	R/W	R/W	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 21	—	All 0	R	Reserved The read value is always 0.
20	NIDEN_CPU0	0	R/W	CPU Noninvasive Debug Enable Controls the noninvasive debug enable signal (NIDEN/SPNIDEN) supplied to the CPU. This bit is only effective when the PINSETEN bit in ICEREGJTTRCSEL is set to 0.
19 to 17	—	All 0	R	Reserved The read value is always 0.
16	DBGEN_CPU0	0	R/W	CPU Debug Enable Controls the debug enable signal (DBGEN/SPIDEN) supplied to the CPU. This bit is only effective when the PINSETEN bit in ICEREGJTTRCSEL is set to 0.
15 to 13	—	All 1	R	Reserved The read value is always 1.
12	RSTRB_CPU0_DERSTZ	1	R/W	NEON Reset (low-active) Resets the NEON unit in the CPU.
11 to 9	—	All 1	R	Reserved The read value is always 1.
8	RSTRB_CPU0_CPURSTZ	1	R/W	CPU Reset (low-active) Resets all systems on the CPU block other than debugging resources.



Bit	Bit Name	Initial Value	R/W	Description
7 to 5	—	All 0	R	Reserved The read value is always 0.
4	RSTRB_SYS_ SYSRSTZ	1	R/W	System Reset (low-active) Resets all systems on the chip other than debugging resources.
3	—	1	R/W	Reserved The read value is always 1. The write value should always be 1.
2	RSTRB_CPU_ PRSTDBGZ	1	R/W	CPU Subsystem Debug Peripheral Reset (low-active) Resets the debug-APB block in the CPU subsystem.
1	RSTRB_CPU_ SYSRSTZ	1	R/W	CPU Subsystem Reset (low-active) Resets the CPU subsystem other than those for debugging.
0	—	0	R	Reserved The read value is always 0.

### 56.4.2 JTAG Trace Select Register (ICEREGJTTRCSEL)

ICEREGJTTRCSEL is used to control trace data output to the respective pins and debug enable signal supplied to the CPU.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	PINSET EN	—	—	TRCMU X_SEL	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R	R	R/W	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	Bit Name	Initial Value	R/W	Description
31 to 24	—	All 0	R	Reserved The read value is always 0.
23	PINSETEN	1	R/W	Selects control method of debug enable signal supplied to the CPU. 0: Supply of the DBGEN and NIDEN signals is determined by the ICEREGMDRSTCTL register. 1: Supply of the DBGEN and NIDEN signals is determined by the BSCANP pin.
22, 21	—	All 0	R	Reserved The read value is always 0.
20	TRCMUX_ SEL	0	R/W	Sets the priority of the function multiplexed on the pins to which trace output is assigned. 0: The function of the pins depends on the general I/O port setting. 1: The pins act as trace outputs regardless of the general I/O port setting.
19 to 0	—	All 0	R	Reserved The read value is always 0.

### 56.4.3 Clock Power Control Register (ICEREGCLKPWRCTRL)

ICEREGCLKPWRCTRL is used to set the fake debug mode.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	FAKED BGCTRL	—	—	—	—	—	—	FAKED BG
Initial value:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W:	R	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R/W

Bit	Bit Name	Initial Value	R/W	Description
31 to 8	—	All 0	R	Reserved The read value is always 0.
7	FAKEDBGCTRL	0	R/W	Enables or disables the fake debug mode. 0: The power control is determined by the standby control register 1 (STBCR1). 1: The power control is determined by this register.
6 to 1	—	All 0	R	Reserved The read value is always 0.
0	FAKEDBG	0	R/W	Selects whether or not to stop the power supply in deep standby mode. 0: The power supply is actually stopped in deep standby mode. 1: The power supply is not stopped even in deep standby mode.

### 56.4.4 Lock Access Register (ICEREGLOCKACCESS)

ICEREGLOCKACCESS is used to enable access to ICE registers by the CPU.

Bit:	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ICEREGLOCKACCESS															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
Bit:	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ICEREGLOCKACCESS															
Initial value:	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W:	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	Bit Name	Initial Value	R/W	Description
31 to 0	ICEREGLOCKACCESS	—	W	To enable access to ICE registers by the CPU, first write H'C5AC CE55 to this register.

## 56.5 Operation

### 56.5.1 TAP Controller

Figure 56.3 shows the internal states of the TAP controller. State transitions basically conform to the JTAG standard.

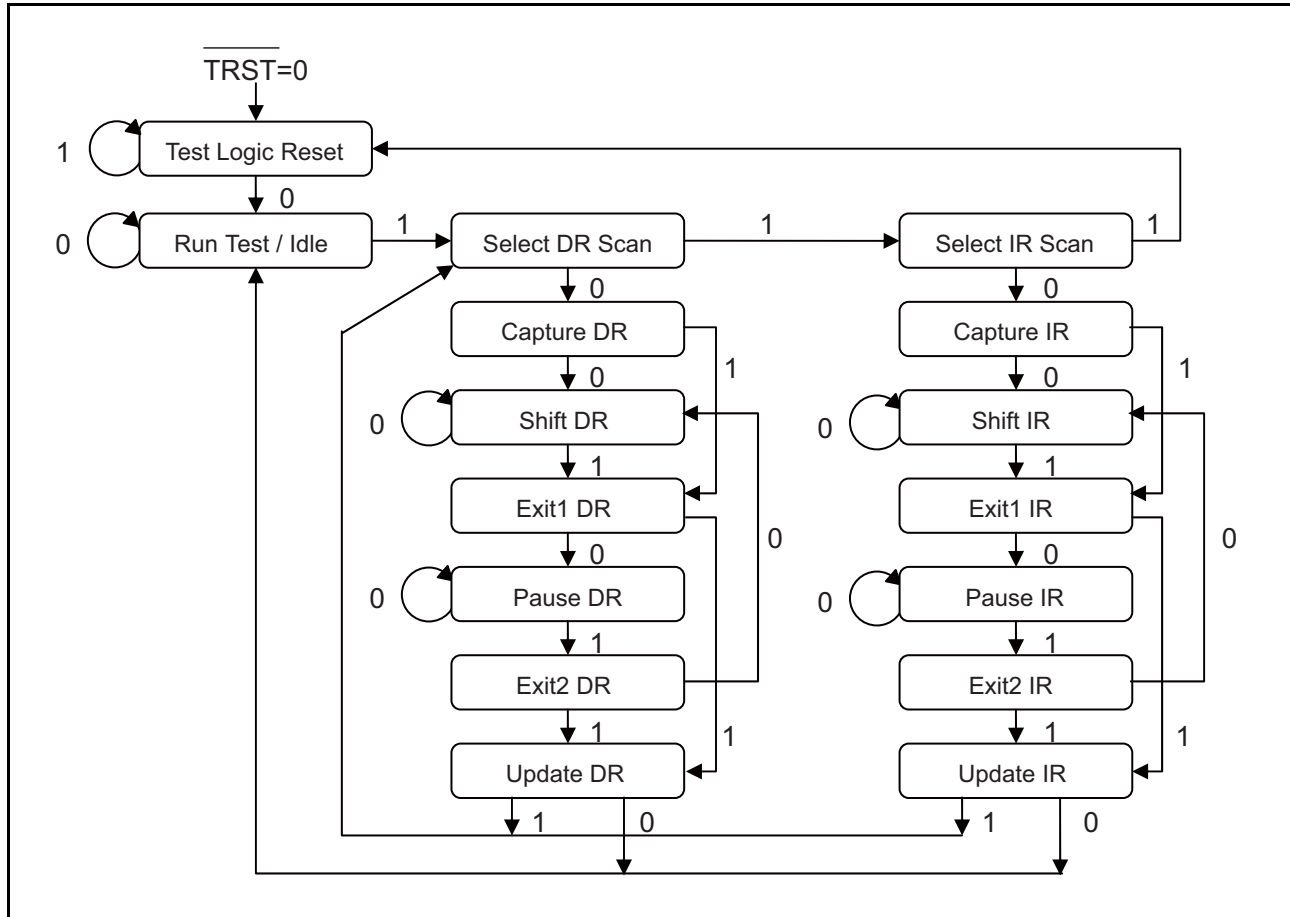


Figure 56.3 State Transitions of TAP Controller

Note: State transition occur according to the TMS value at the rising edge of the TCK signal. The TDI value sampled at the rising edge of the TCK signal and shifted at the falling edge of the TCK signal. TDO value is output at the rising edge of the TCK signal. The TDO signal is in a Hi-Z state for TAP controller states other than Shift-DR and Shift-IR. A transition to the Test-Logic-Reset state by clearing  $\overline{\text{TRST}}$  to 0 is performed asynchronously with the TCK signal.

### 56.5.2 Reset Signal Setting

Table 56.12 Reset Signal Setting

$\overline{\text{RES}}$	$\overline{\text{TRST}}$	Chip State
L	L	The chip is power-on reset and this module is reset.*1
	H	The chip is power-on reset.*1
H	L	This module is only reset.*2
	H	Normal operation*2

Note 1. By asserting the  $\overline{\text{RES}}$  and  $\overline{\text{TRST}}$  signals, the CPU and CoreSight enter the reset state. Set the debugging function while asserting the  $\overline{\text{RES}}$  signal after negating the  $\overline{\text{TRST}}$  signal.

Note 2. When the chip is in CoreSight debugging mode, do not negate the  $\overline{\text{TRST}}$  signal while the  $\overline{\text{RES}}$  signal is at the high-level.

## 56.6 Boundary Scan

Commands can be set in BSIR by this module to place this module's pins in boundary scan mode stipulated by JTAG.

### 56.6.1 Supported Instructions

This LSI supports three mandatory instructions stipulated by JTAG standard (BYPASS, SAMPLE/PRELOAD, and EXTEST) and three option instructions (IDCODE, CLAMP, and HIGHZ).

#### (1) BYPASS

BYPASS is a mandatory instruction that operates the bypass register. This instruction is used to shorten the shift path to speed up serial data transfer involving other LSI on the printed circuit board. This LSI's system circuits are not affected by execution of this instruction.

#### (2) SAMPLE/PRELOAD

SAMPLE/PRELOAD is used to input data from this LSI's internal circuit to the boundary scan register, outputs data from the scan path, and loads data onto the scan path. While this instruction is being executed, signals input to this LSI pins are transmitted directly to the internal circuit, and the values of the internal circuit are directly output externally from the output pins. This LSI's system circuits are not affected by execution of this instruction.

In a SAMPLE operation, a snapshot of a value to be transferred from an input pin to the internal circuit, or a value to be transferred from the internal circuit to an output pin, is latched into the boundary scan register and read from the scan path. Snapshot latching is performed in synchronization with the rising edge of the TCK signal in the Capture-DR state. Snapshot latching does not affect normal operation of this LSI.

In a PRELOAD operation, an initial value is set in the parallel output latch of the boundary scan register from the scan path prior to the EXTEST instruction. Without a PRELOAD operation, when the EXTEST instruction was executed, an undefined value would be output from the output pin until completion of the initial scan sequence (transfer to the output latch) (with the EXTEST instruction, the parallel output latch value is constantly output to the output pin).

#### (3) EXTEST

This instruction is provided to test external circuit when this LSI is mounted on a printed circuit board. When this instruction is executed, output pins are used to output test data (previously set by the SAMPLE/PRELOAD instruction) from the boundary scan register to the printed circuit board, and input pins are used to latch test results into the boundary scan register from the printed circuit board. If testing is carried out by using the EXTEST instruction N times, the Nth test data is scanned-in when test data (N-1) is scanned out. The data loaded into the output pin boundary scan register in the Capture-DR state are not used for external circuit testing (data are replaced by the shift operation).

#### (4) IDCODE

Commands can be set in BSIR via pins of this module to place the module's pins in the IDCODE mode stipulated by JTAG. When this module is initialized ( $\overline{\text{TRST}}$  is asserted or TAP is in the Test-Logic-Reset state), the IDCODE mode is entered.

#### (5) CLAMP, HIGHZ

Commands can be set in BSIR via pins of this module to place the module's pins in the CLAMP or HIGHZ mode stipulated by JTAG.

### 56.6.2 Points for Attention

1. Boundary scan mode does not cover clock-related signals (EXTAL, XTAL, CKIO, AUDIO\_X1, AUDIO\_X2, USB\_X1, USB\_X2, RTC\_X1, and RTC\_X2).
2. Boundary scan mode does not cover reset-related signal ( $\overline{\text{RES}}$ ).
3. Boundary scan mode does not cover this module's signals (TCK, TDI, TDO, TMS,  $\overline{\text{TRST}}$ , and BSCANP).
4. Boundary scan mode does not cover USB-related signals (DP0, DP1, DM0, DM1, VBUS0, VBUS1, and REFRIN).
5. Boundary scan mode does not cover P0\_3.
6. Boundary scan mode does not cover the digital video decoder related signals (VIN1A, VIN2A, VIN1B, VIN2B, VIDEO\_X1, VIDEO\_X2, VRP, VRM, and REXT).
7. Boundary scan mode does not cover AVref (reference voltage pin for the A/D converter).

### 56.7 Usage Notes

1. Once a command of this module is set, it will not be modified unless another command is reissued. If the same command is given continuously, the command must be set after a command (BYPASS, etc.) that does not affect LSI operations is once set.
2. Regardless of whether or not this module is to be used, it should be initialized by asserting the  $\overline{\text{TRST}}$  signal when power is supplied and on release from deep standby mode (excluding fake debug mode) through assertion of the  $\overline{\text{RES}}$  signal.
3. Be sure to wait for at least 200 ns after negating the  $\overline{\text{TRST}}$  signal before starting TAP controller operation.
4. Be sure to fix the TMS pin to the high level until 200 ns have elapsed after negation of the  $\overline{\text{TRST}}$  signal.

## 57. List of Registers

### 57.1 Register Addresses

**Table 57.1 Register Addresses**

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Secondary cache	Cache ID Register	reg0_cache_id	32	H'3FFFF000	32
	Cache Type Register	reg0_cache_type	32	H'3FFFF004	32
	Control Register	reg1_control	32	H'3FFFF100	32
	Auxiliary Control Register	reg1_aux_control	32	H'3FFFF104	32
	Tag RAM Latency Control Register	reg1_tag_ram_control	32	H'3FFFF108	32
	Data RAM Latency Control Register	reg1_data_ram_control	32	H'3FFFF10C	32
	Event Counter Control Register	reg2_ev_counter_ctrl	32	H'3FFFF200	32
	Event Counter Configuration Register 1	reg2_ev_counter1_cfg	32	H'3FFFF204	32
	Event Counter Configuration Register 0	reg2_ev_counter0_cfg	32	H'3FFFF208	32
	Event counter value register 1	reg2_ev_counter1	32	H'3FFFF20C	32
	Event counter value register 0	reg2_ev_counter0	32	H'3FFFF210	32
	Interrupt Mask Register	reg2_int_mask	32	H'3FFFF214	32
	Masked Interrupt Status Register	reg2_int_mask_status	32	H'3FFFF218	32
	Raw Interrupt Status Register	reg2_int_raw_status	32	H'3FFFF21C	32
	Interrupt Clear Register	reg2_int_clear	32	H'3FFFF220	32
	Cache Sync Register	reg7_cache_sync	32	H'3FFFF730	32
	Invalidate Line by PA Register	reg7_inv_pa	32	H'3FFFF770	32
	Invalidate by Way Register	reg7_inv_way	32	H'3FFFF77C	32
	Clean Line by PA Register	reg7_clean_pa	32	H'3FFFF7B0	32
	Clean Line by Set/Way Register	reg7_clean_index	32	H'3FFFF7B8	32
	Clean by Way Register	reg7_clean_way	32	H'3FFFF7BC	32
	Clean and Invalidate Line by PA Register	reg7_clean_inv_pa	32	H'3FFFF7F0	32
	Clean and Invalidate Line by Set/Way Register	reg7_clean_inv_index	32	H'3FFFF7F8	32
	Clean and Invalidate by Way Register	reg7_clean_inv_way	32	H'3FFFF7FC	32
	Data Lockdown 0 Register	reg9_d_lockdown0	32	H'3FFFF900	32
	Instruction Lockdown 0 Register	reg9_i_lockdown0	32	H'3FFFF904	32
	Data Lockdown 1 Register	reg9_d_lockdown1	32	H'3FFFF908	32
	Instruction Lockdown 1 Register	reg9_i_lockdown1	32	H'3FFFF90C	32
	Data Lockdown 2 Register	reg9_d_lockdown2	32	H'3FFFF910	32
	Instruction Lockdown 2 Register	reg9_i_lockdown2	32	H'3FFFF914	32
	Data Lockdown 3 Register	reg9_d_lockdown3	32	H'3FFFF918	32
	Instruction Lockdown 3 Register	reg9_i_lockdown3	32	H'3FFFF91C	32
	Data Lockdown 4 Register	reg9_d_lockdown4	32	H'3FFFF920	32
	Instruction Lockdown 4 Register	reg9_i_lockdown4	32	H'3FFFF924	32
	Data Lockdown 5 Register	reg9_d_lockdown5	32	H'3FFFF928	32
	Instruction Lockdown 5 Register	reg9_i_lockdown5	32	H'3FFFF92C	32
	Data Lockdown 6 Register	reg9_d_lockdown6	32	H'3FFFF930	32
	Instruction Lockdown 6 Register	reg9_i_lockdown6	32	H'3FFFF934	32
	Data Lockdown 7 Register	reg9_d_lockdown7	32	H'3FFFF938	32
	Instruction Lockdown 7 Register	reg9_i_lockdown7	32	H'3FFFF93C	32
	Lockdown by Line Enable Register	reg9_lock_line_en	32	H'3FFFF950	32
	Unlock All Lines Register	reg9_unlock_way	32	H'3FFFF954	32
	Address Filtering Start Register	reg12_addr_filtering_start	32	H'3FFFFC00	32
	Address Filtering End Register	reg12_addr_filtering_end	32	H'3FFFFC04	32
Debug Control Register	reg15_debug_ctrl	32	H'3FFFFF40	32	
Prefetch Control Register	reg15_prefetch_ctrl	32	H'3FFFFF60	32	
Power Control Register	reg15_power_ctrl	32	H'3FFFFF80	32	

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
LSI internal bus	Remap register	RMPR	32	H'FCFE1A00	32
	AXI bus control register 0	AXIBUSCTL0	32	H'FCFE1A04	32
	AXI bus control register 1	AXIBUSCTL1	32	H'FCFE1A08	32
	AXI bus control register 2	AXIBUSCTL2	32	H'FCFE1A0C	32
	AXI bus control register 3	AXIBUSCTL3	32	H'FCFE1A10	32
	AXI bus control register 4	AXIBUSCTL4	32	H'FCFE1A14	32
	AXI bus control register 5	AXIBUSCTL5	32	H'FCFE1A18	32
	AXI bus control register 6	AXIBUSCTL6	32	H'FCFE1A1C	32
	AXI bus control register 7	AXIBUSCTL7	32	H'FCFE1A20	32
	AXI bus control register 8	AXIBUSCTL8	32	H'FCFE1A24	32
	AXI bus control register 9	AXIBUSCTL9	32	H'FCFE1A28	32
	AXI bus control register 10	AXIBUSCTL10	32	H'FCFE1A2C	32
	AXI bus response error interrupt control register 0	AXIRERRCTL0	32	H'FCFE1A30	32
	AXI bus response error interrupt control register 1	AXIRERRCTL1	32	H'FCFE1A34	32
	AXI bus response error interrupt control register 2	AXIRERRCTL2	32	H'FCFE1A38	32
	AXI bus response error interrupt control register 3	AXIRERRCTL3	32	H'FCFE1A3C	32
	AXI bus response error status register 0	AXIRERRST0	32	H'FCFE1A40	32
	AXI bus response error status register 2	AXIRERRST2	32	H'FCFE1A48	32
	AXI bus response error clear register 0	AXIRERRCLR0	32	H'FCFE1A50	32
	AXI bus response error clear register 1	AXIRERRCLR1	32	H'FCFE1A54	32
AXI bus response error clear register 2	AXIRERRCLR2	32	H'FCFE1A58	32	
AXI bus response error clear register 3	AXIRERRCLR3	32	H'FCFE1A5C	32	
Clock pulse generator	Frequency control register	FRQCR	16	H'FCFE0010	16
	Frequency control register 2	FRQCR2	16	H'FCFE0014	16
Interrupt controller	Interrupt control register 0	ICR0	16	H'FCFEF800	16
	Interrupt control register 1	ICR1	16	H'FCFEF802	16
	IRQ interrupt request register	IRQRR	16	H'FCFEF804	16
	Distributor control register	ICDDCR	32	H'E8201000	32
	Interrupt controller type register	ICDICTR	32	H'E8201004	32
	Distributor implementer identification register	ICDIIDR	32	H'E8201008	32
	Interrupt security register 0	ICDISR0	32	H'E8201080	32
	Interrupt security register 1	ICDISR1	32	H'E8201084	32
	Interrupt security register 2	ICDISR2	32	H'E8201088	32
	Interrupt security register 3	ICDISR3	32	H'E820108C	32
	Interrupt security register 4	ICDISR4	32	H'E8201090	32
	Interrupt security register 5	ICDISR5	32	H'E8201094	32
	Interrupt security register 6	ICDISR6	32	H'E8201098	32
	Interrupt security register 7	ICDISR7	32	H'E820109C	32
	Interrupt security register 8	ICDISR8	32	H'E82010A0	32
	Interrupt security register 9	ICDISR9	32	H'E82010A4	32
	Interrupt security register 10	ICDISR10	32	H'E82010A8	32
	Interrupt security register 11	ICDISR11	32	H'E82010AC	32
	Interrupt security register 12	ICDISR12	32	H'E82010B0	32
	Interrupt security register 13	ICDISR13	32	H'E82010B4	32
	Interrupt security register 14	ICDISR14	32	H'E82010B8	32
	Interrupt security register 15	ICDISR15	32	H'E82010BC	32
	Interrupt security register 16	ICDISR16	32	H'E82010C0	32
	Interrupt security register 17	ICDISR17	32	H'E82010C4	32
	Interrupt security register 18	ICDISR18	32	H'E82010C8	32
	Interrupt set-enable register 0	ICDISER0	32	H'E8201100	32
	Interrupt set-enable register 1	ICDISER1	32	H'E8201104	32

**Table 57.1 Register Addresses**

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Interrupt controller	Interrupt set-enable register 2	ICDISER2	32	H'E8201108	32
	Interrupt set-enable register 3	ICDISER3	32	H'E820110C	32
	Interrupt set-enable register 4	ICDISER4	32	H'E8201110	32
	Interrupt set-enable register 5	ICDISER5	32	H'E8201114	32
	Interrupt set-enable register 6	ICDISER6	32	H'E8201118	32
	Interrupt set-enable register 7	ICDISER7	32	H'E820111C	32
	Interrupt set-enable register 8	ICDISER8	32	H'E8201120	32
	Interrupt set-enable register 9	ICDISER9	32	H'E8201124	32
	Interrupt set-enable register 10	ICDISER10	32	H'E8201128	32
	Interrupt set-enable register 11	ICDISER11	32	H'E820112C	32
	Interrupt set-enable register 12	ICDISER12	32	H'E8201130	32
	Interrupt set-enable register 13	ICDISER13	32	H'E8201134	32
	Interrupt set-enable register 14	ICDISER14	32	H'E8201138	32
	Interrupt set-enable register 15	ICDISER15	32	H'E820113C	32
	Interrupt set-enable register 16	ICDISER16	32	H'E8201140	32
	Interrupt set-enable register 17	ICDISER17	32	H'E8201144	32
	Interrupt set-enable register 18	ICDISER18	32	H'E8201148	32
	Interrupt clear-enable register 0	ICDICER0	32	H'E8201180	32
	Interrupt clear-enable register 1	ICDICER1	32	H'E8201184	32
	Interrupt clear-enable register 2	ICDICER2	32	H'E8201188	32
	Interrupt clear-enable register 3	ICDICER3	32	H'E820118C	32
	Interrupt clear-enable register 4	ICDICER4	32	H'E8201190	32
	Interrupt clear-enable register 5	ICDICER5	32	H'E8201194	32
	Interrupt clear-enable register 6	ICDICER6	32	H'E8201198	32
	Interrupt clear-enable register 7	ICDICER7	32	H'E820119C	32
	Interrupt clear-enable register 8	ICDICER8	32	H'E82011A0	32
	Interrupt clear-enable register 9	ICDICER9	32	H'E82011A4	32
	Interrupt clear-enable register 10	ICDICER10	32	H'E82011A8	32
	Interrupt clear-enable register 11	ICDICER11	32	H'E82011AC	32
	Interrupt clear-enable register 12	ICDICER12	32	H'E82011B0	32
	Interrupt clear-enable register 13	ICDICER13	32	H'E82011B4	32
	Interrupt clear-enable register 14	ICDICER14	32	H'E82011B8	32
	Interrupt clear-enable register 15	ICDICER15	32	H'E82011BC	32
	Interrupt clear-enable register 16	ICDICER16	32	H'E82011C0	32
	Interrupt clear-enable register 17	ICDICER17	32	H'E82011C4	32
	Interrupt clear-enable register 18	ICDICER18	32	H'E82011C8	32
	Interrupt set-pending register 0	ICDISPR0	32	H'E8201200	32
	Interrupt set-pending register 1	ICDISPR1	32	H'E8201204	32
	Interrupt set-pending register 2	ICDISPR2	32	H'E8201208	32
	Interrupt set-pending register 3	ICDISPR3	32	H'E820120C	32
	Interrupt set-pending register 4	ICDISPR4	32	H'E8201210	32
	Interrupt set-pending register 5	ICDISPR5	32	H'E8201214	32
	Interrupt set-pending register 6	ICDISPR6	32	H'E8201218	32
	Interrupt set-pending register 7	ICDISPR7	32	H'E820121C	32
	Interrupt set-pending register 8	ICDISPR8	32	H'E8201220	32
Interrupt set-pending register 9	ICDISPR9	32	H'E8201224	32	
Interrupt set-pending register 10	ICDISPR10	32	H'E8201228	32	
Interrupt set-pending register 11	ICDISPR11	32	H'E820122C	32	
Interrupt set-pending register 12	ICDISPR12	32	H'E8201230	32	
Interrupt set-pending register 13	ICDISPR13	32	H'E8201234	32	
Interrupt set-pending register 14	ICDISPR14	32	H'E8201238	32	
Interrupt set-pending register 15	ICDISPR15	32	H'E820123C	32	



Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Interrupt controller	Interrupt set-pending register 16	ICDISPR16	32	H'E8201240	32
	Interrupt set-pending register 17	ICDISPR17	32	H'E8201244	32
	Interrupt set-pending register 18	ICDISPR18	32	H'E8201248	32
	Interrupt clear-pending register 0	ICDICPR0	32	H'E8201280	32
	Interrupt clear-pending register 1	ICDICPR1	32	H'E8201284	32
	Interrupt clear-pending register 2	ICDICPR2	32	H'E8201288	32
	Interrupt clear-pending register 3	ICDICPR3	32	H'E820128C	32
	Interrupt clear-pending register 4	ICDICPR4	32	H'E8201290	32
	Interrupt clear-pending register 5	ICDICPR5	32	H'E8201294	32
	Interrupt clear-pending register 6	ICDICPR6	32	H'E8201298	32
	Interrupt clear-pending register 7	ICDICPR7	32	H'E820129C	32
	Interrupt clear-pending register 8	ICDICPR8	32	H'E82012A0	32
	Interrupt clear-pending register 9	ICDICPR9	32	H'E82012A4	32
	Interrupt clear-pending register 10	ICDICPR10	32	H'E82012A8	32
	Interrupt clear-pending register 11	ICDICPR11	32	H'E82012AC	32
	Interrupt clear-pending register 12	ICDICPR12	32	H'E82012B0	32
	Interrupt clear-pending register 13	ICDICPR13	32	H'E82012B4	32
	Interrupt clear-pending register 14	ICDICPR14	32	H'E82012B8	32
	Interrupt clear-pending register 15	ICDICPR15	32	H'E82012BC	32
	Interrupt clear-pending register 16	ICDICPR16	32	H'E82012C0	32
	Interrupt clear-pending register 17	ICDICPR17	32	H'E82012C4	32
	Interrupt clear-pending register 18	ICDICPR18	32	H'E82012C8	32
	Active bit register 0	ICDABR0	32	H'E8201300	32
	Active bit register 1	ICDABR1	32	H'E8201304	32
	Active bit register 2	ICDABR2	32	H'E8201308	32
	Active bit register 3	ICDABR3	32	H'E820130C	32
	Active bit register 4	ICDABR4	32	H'E8201310	32
	Active bit register 5	ICDABR5	32	H'E8201314	32
	Active bit register 6	ICDABR6	32	H'E8201318	32
	Active bit register 7	ICDABR7	32	H'E820131C	32
	Active bit register 8	ICDABR8	32	H'E8201320	32
	Active bit register 9	ICDABR9	32	H'E8201324	32
	Active bit register 10	ICDABR10	32	H'E8201328	32
	Active bit register 11	ICDABR11	32	H'E820132C	32
	Active bit register 12	ICDABR12	32	H'E8201330	32
	Active bit register 13	ICDABR13	32	H'E8201334	32
	Active bit register 14	ICDABR14	32	H'E8201338	32
	Active bit register 15	ICDABR15	32	H'E820133C	32
	Active bit register 16	ICDABR16	32	H'E8201340	32
	Active bit register 17	ICDABR17	32	H'E8201344	32
	Active bit register 18	ICDABR18	32	H'E8201348	32
	Interrupt priority register 0	ICDIPR0	32	H'E8201400	32
Interrupt priority register 1	ICDIPR1	32	H'E8201404	32	
Interrupt priority register 2	ICDIPR2	32	H'E8201408	32	
Interrupt priority register 3	ICDIPR3	32	H'E820140C	32	
Interrupt priority register 4	ICDIPR4	32	H'E8201410	32	
Interrupt priority register 5	ICDIPR5	32	H'E8201414	32	
Interrupt priority register 6	ICDIPR6	32	H'E8201418	32	
Interrupt priority register 7	ICDIPR7	32	H'E820141C	32	
Interrupt priority register 8	ICDIPR8	32	H'E8201420	32	
Interrupt priority register 9	ICDIPR9	32	H'E8201424	32	
Interrupt priority register 10	ICDIPR10	32	H'E8201428	32	

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Interrupt controller	Interrupt priority register 11	ICDIPR11	32	H'E820142C	32
	Interrupt priority register 12	ICDIPR12	32	H'E8201430	32
	Interrupt priority register 13	ICDIPR13	32	H'E8201434	32
	Interrupt priority register 14	ICDIPR14	32	H'E8201438	32
	Interrupt priority register 15	ICDIPR15	32	H'E820143C	32
	Interrupt priority register 16	ICDIPR16	32	H'E8201440	32
	Interrupt priority register 17	ICDIPR17	32	H'E8201444	32
	Interrupt priority register 18	ICDIPR18	32	H'E8201448	32
	Interrupt priority register 19	ICDIPR19	32	H'E820144C	32
	Interrupt priority register 20	ICDIPR20	32	H'E8201450	32
	Interrupt priority register 21	ICDIPR21	32	H'E8201454	32
	Interrupt priority register 22	ICDIPR22	32	H'E8201458	32
	Interrupt priority register 23	ICDIPR23	32	H'E820145C	32
	Interrupt priority register 24	ICDIPR24	32	H'E8201460	32
	Interrupt priority register 25	ICDIPR25	32	H'E8201464	32
	Interrupt priority register 26	ICDIPR26	32	H'E8201468	32
	Interrupt priority register 27	ICDIPR27	32	H'E820146C	32
	Interrupt priority register 28	ICDIPR28	32	H'E8201470	32
	Interrupt priority register 29	ICDIPR29	32	H'E8201474	32
	Interrupt priority register 30	ICDIPR30	32	H'E8201478	32
	Interrupt priority register 31	ICDIPR31	32	H'E820147C	32
	Interrupt priority register 32	ICDIPR32	32	H'E8201480	32
	Interrupt priority register 33	ICDIPR33	32	H'E8201484	32
	Interrupt priority register 34	ICDIPR34	32	H'E8201488	32
	Interrupt priority register 35	ICDIPR35	32	H'E820148C	32
	Interrupt priority register 36	ICDIPR36	32	H'E8201490	32
	Interrupt priority register 37	ICDIPR37	32	H'E8201494	32
	Interrupt priority register 38	ICDIPR38	32	H'E8201498	32
	Interrupt priority register 39	ICDIPR39	32	H'E820149C	32
	Interrupt priority register 40	ICDIPR40	32	H'E82014A0	32
	Interrupt priority register 41	ICDIPR41	32	H'E82014A4	32
	Interrupt priority register 42	ICDIPR42	32	H'E82014A8	32
	Interrupt priority register 43	ICDIPR43	32	H'E82014AC	32
	Interrupt priority register 44	ICDIPR44	32	H'E82014B0	32
	Interrupt priority register 45	ICDIPR45	32	H'E82014B4	32
	Interrupt priority register 46	ICDIPR46	32	H'E82014B8	32
	Interrupt priority register 47	ICDIPR47	32	H'E82014BC	32
	Interrupt priority register 48	ICDIPR48	32	H'E82014C0	32
	Interrupt priority register 49	ICDIPR49	32	H'E82014C4	32
	Interrupt priority register 50	ICDIPR50	32	H'E82014C8	32
	Interrupt priority register 51	ICDIPR51	32	H'E82014CC	32
	Interrupt priority register 52	ICDIPR52	32	H'E82014D0	32
	Interrupt priority register 53	ICDIPR53	32	H'E82014D4	32
	Interrupt priority register 54	ICDIPR54	32	H'E82014D8	32
	Interrupt priority register 55	ICDIPR55	32	H'E82014DC	32
	Interrupt priority register 56	ICDIPR56	32	H'E82014E0	32
	Interrupt priority register 57	ICDIPR57	32	H'E82014E4	32
	Interrupt priority register 58	ICDIPR58	32	H'E82014E8	32
	Interrupt priority register 59	ICDIPR59	32	H'E82014EC	32
	Interrupt priority register 60	ICDIPR60	32	H'E82014F0	32
	Interrupt priority register 61	ICDIPR61	32	H'E82014F4	32
	Interrupt priority register 62	ICDIPR62	32	H'E82014F8	32

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Interrupt controller	Interrupt priority register 63	ICDIPR63	32	H'E82014FC	32
	Interrupt priority register 64	ICDIPR64	32	H'E8201500	32
	Interrupt priority register 65	ICDIPR65	32	H'E8201504	32
	Interrupt priority register 66	ICDIPR66	32	H'E8201508	32
	Interrupt priority register 67	ICDIPR67	32	H'E820150C	32
	Interrupt priority register 68	ICDIPR68	32	H'E8201510	32
	Interrupt priority register 69	ICDIPR69	32	H'E8201514	32
	Interrupt priority register 70	ICDIPR70	32	H'E8201518	32
	Interrupt priority register 71	ICDIPR71	32	H'E820151C	32
	Interrupt priority register 72	ICDIPR72	32	H'E8201520	32
	Interrupt priority register 73	ICDIPR73	32	H'E8201524	32
	Interrupt priority register 74	ICDIPR74	32	H'E8201528	32
	Interrupt priority register 75	ICDIPR75	32	H'E820152C	32
	Interrupt priority register 76	ICDIPR76	32	H'E8201530	32
	Interrupt priority register 77	ICDIPR77	32	H'E8201534	32
	Interrupt priority register 78	ICDIPR78	32	H'E8201538	32
	Interrupt priority register 79	ICDIPR79	32	H'E820153C	32
	Interrupt priority register 80	ICDIPR80	32	H'E8201540	32
	Interrupt priority register 81	ICDIPR81	32	H'E8201544	32
	Interrupt priority register 82	ICDIPR82	32	H'E8201548	32
	Interrupt priority register 83	ICDIPR83	32	H'E820154C	32
	Interrupt priority register 84	ICDIPR84	32	H'E8201550	32
	Interrupt priority register 85	ICDIPR85	32	H'E8201554	32
	Interrupt priority register 86	ICDIPR86	32	H'E8201558	32
	Interrupt priority register 87	ICDIPR87	32	H'E820155C	32
	Interrupt priority register 88	ICDIPR88	32	H'E8201560	32
	Interrupt priority register 89	ICDIPR89	32	H'E8201564	32
	Interrupt priority register 90	ICDIPR90	32	H'E8201568	32
	Interrupt priority register 91	ICDIPR91	32	H'E820156C	32
	Interrupt priority register 92	ICDIPR92	32	H'E8201570	32
	Interrupt priority register 93	ICDIPR93	32	H'E8201574	32
	Interrupt priority register 94	ICDIPR94	32	H'E8201578	32
	Interrupt priority register 95	ICDIPR95	32	H'E820157C	32
	Interrupt priority register 96	ICDIPR96	32	H'E8201580	32
	Interrupt priority register 97	ICDIPR97	32	H'E8201584	32
	Interrupt priority register 98	ICDIPR98	32	H'E8201588	32
	Interrupt priority register 99	ICDIPR99	32	H'E820158C	32
	Interrupt priority register 100	ICDIPR100	32	H'E8201590	32
	Interrupt priority register 101	ICDIPR101	32	H'E8201594	32
	Interrupt priority register 102	ICDIPR102	32	H'E8201598	32
	Interrupt priority register 103	ICDIPR103	32	H'E820159C	32
	Interrupt priority register 104	ICDIPR104	32	H'E82015A0	32
	Interrupt priority register 105	ICDIPR105	32	H'E82015A4	32
	Interrupt priority register 106	ICDIPR106	32	H'E82015A8	32
	Interrupt priority register 107	ICDIPR107	32	H'E82015AC	32
	Interrupt priority register 108	ICDIPR108	32	H'E82015B0	32
	Interrupt priority register 109	ICDIPR109	32	H'E82015B4	32
	Interrupt priority register 110	ICDIPR110	32	H'E82015B8	32
Interrupt priority register 111	ICDIPR111	32	H'E82015BC	32	
Interrupt priority register 112	ICDIPR112	32	H'E82015C0	32	
Interrupt priority register 113	ICDIPR113	32	H'E82015C4	32	
Interrupt priority register 114	ICDIPR114	32	H'E82015C8	32	

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Interrupt controller	Interrupt priority register 115	ICDIPR115	32	H'E82015CC	32
	Interrupt priority register 116	ICDIPR116	32	H'E82015D0	32
	Interrupt priority register 117	ICDIPR117	32	H'E82015D4	32
	Interrupt priority register 118	ICDIPR118	32	H'E82015D8	32
	Interrupt priority register 119	ICDIPR119	32	H'E82015DC	32
	Interrupt priority register 120	ICDIPR120	32	H'E82015E0	32
	Interrupt priority register 121	ICDIPR121	32	H'E82015E4	32
	Interrupt priority register 122	ICDIPR122	32	H'E82015E8	32
	Interrupt priority register 123	ICDIPR123	32	H'E82015EC	32
	Interrupt priority register 124	ICDIPR124	32	H'E82015F0	32
	Interrupt priority register 125	ICDIPR125	32	H'E82015F4	32
	Interrupt priority register 126	ICDIPR126	32	H'E82015F8	32
	Interrupt priority register 127	ICDIPR127	32	H'E82015FC	32
	Interrupt priority register 128	ICDIPR128	32	H'E8201600	32
	Interrupt priority register 129	ICDIPR129	32	H'E8201604	32
	Interrupt priority register 130	ICDIPR130	32	H'E8201608	32
	Interrupt priority register 131	ICDIPR131	32	H'E820160C	32
	Interrupt priority register 132	ICDIPR132	32	H'E8201610	32
	Interrupt priority register 133	ICDIPR133	32	H'E8201614	32
	Interrupt priority register 134	ICDIPR134	32	H'E8201618	32
	Interrupt priority register 135	ICDIPR135	32	H'E820161C	32
	Interrupt priority register 136	ICDIPR136	32	H'E8201620	32
	Interrupt priority register 137	ICDIPR137	32	H'E8201624	32
	Interrupt priority register 138	ICDIPR138	32	H'E8201628	32
	Interrupt priority register 139	ICDIPR139	32	H'E820162C	32
	Interrupt priority register 140	ICDIPR140	32	H'E8201630	32
	Interrupt priority register 141	ICDIPR141	32	H'E8201634	32
	Interrupt priority register 142	ICDIPR142	32	H'E8201638	32
	Interrupt priority register 143	ICDIPR143	32	H'E820163C	32
	Interrupt priority register 144	ICDIPR144	32	H'E8201640	32
	Interrupt priority register 145	ICDIPR145	32	H'E8201644	32
	Interrupt priority register 146	ICDIPR146	32	H'E8201648	32
	Interrupt processor target register 0	ICDIPTR0	32	H'E8201800	32
	Interrupt processor target register 1	ICDIPTR1	32	H'E8201804	32
	Interrupt processor target register 2	ICDIPTR2	32	H'E8201808	32
	Interrupt processor target register 3	ICDIPTR3	32	H'E820180C	32
	Interrupt processor target register 4	ICDIPTR4	32	H'E8201810	32
	Interrupt processor target register 5	ICDIPTR5	32	H'E8201814	32
	Interrupt processor target register 6	ICDIPTR6	32	H'E8201818	32
	Interrupt processor target register 7	ICDIPTR7	32	H'E820181C	32
	Interrupt processor target register 8	ICDIPTR8	32	H'E8201820	32
	Interrupt processor target register 9	ICDIPTR9	32	H'E8201824	32
	Interrupt processor target register 10	ICDIPTR10	32	H'E8201828	32
Interrupt processor target register 11	ICDIPTR11	32	H'E820182C	32	
Interrupt processor target register 12	ICDIPTR12	32	H'E8201830	32	
Interrupt processor target register 13	ICDIPTR13	32	H'E8201834	32	
Interrupt processor target register 14	ICDIPTR14	32	H'E8201838	32	
Interrupt processor target register 15	ICDIPTR15	32	H'E820183C	32	
Interrupt processor target register 16	ICDIPTR16	32	H'E8201840	32	
Interrupt processor target register 17	ICDIPTR17	32	H'E8201844	32	
Interrupt processor target register 18	ICDIPTR18	32	H'E8201848	32	
Interrupt processor target register 19	ICDIPTR19	32	H'E820184C	32	

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Interrupt controller	Interrupt processor target register 20	ICDIPT20	32	H'E8201850	32
	Interrupt processor target register 21	ICDIPT21	32	H'E8201854	32
	Interrupt processor target register 22	ICDIPT22	32	H'E8201858	32
	Interrupt processor target register 23	ICDIPT23	32	H'E820185C	32
	Interrupt processor target register 24	ICDIPT24	32	H'E8201860	32
	Interrupt processor target register 25	ICDIPT25	32	H'E8201864	32
	Interrupt processor target register 26	ICDIPT26	32	H'E8201868	32
	Interrupt processor target register 27	ICDIPT27	32	H'E820186C	32
	Interrupt processor target register 28	ICDIPT28	32	H'E8201870	32
	Interrupt processor target register 29	ICDIPT29	32	H'E8201874	32
	Interrupt processor target register 30	ICDIPT30	32	H'E8201878	32
	Interrupt processor target register 31	ICDIPT31	32	H'E820187C	32
	Interrupt processor target register 32	ICDIPT32	32	H'E8201880	32
	Interrupt processor target register 33	ICDIPT33	32	H'E8201884	32
	Interrupt processor target register 34	ICDIPT34	32	H'E8201888	32
	Interrupt processor target register 35	ICDIPT35	32	H'E820188C	32
	Interrupt processor target register 36	ICDIPT36	32	H'E8201890	32
	Interrupt processor target register 37	ICDIPT37	32	H'E8201894	32
	Interrupt processor target register 38	ICDIPT38	32	H'E8201898	32
	Interrupt processor target register 39	ICDIPT39	32	H'E820189C	32
	Interrupt processor target register 40	ICDIPT40	32	H'E82018A0	32
	Interrupt processor target register 41	ICDIPT41	32	H'E82018A4	32
	Interrupt processor target register 42	ICDIPT42	32	H'E82018A8	32
	Interrupt processor target register 43	ICDIPT43	32	H'E82018AC	32
	Interrupt processor target register 44	ICDIPT44	32	H'E82018B0	32
	Interrupt processor target register 45	ICDIPT45	32	H'E82018B4	32
	Interrupt processor target register 46	ICDIPT46	32	H'E82018B8	32
	Interrupt processor target register 47	ICDIPT47	32	H'E82018BC	32
	Interrupt processor target register 48	ICDIPT48	32	H'E82018C0	32
	Interrupt processor target register 49	ICDIPT49	32	H'E82018C4	32
	Interrupt processor target register 50	ICDIPT50	32	H'E82018C8	32
	Interrupt processor target register 51	ICDIPT51	32	H'E82018CC	32
	Interrupt processor target register 52	ICDIPT52	32	H'E82018D0	32
	Interrupt processor target register 53	ICDIPT53	32	H'E82018D4	32
	Interrupt processor target register 54	ICDIPT54	32	H'E82018D8	32
	Interrupt processor target register 55	ICDIPT55	32	H'E82018DC	32
	Interrupt processor target register 56	ICDIPT56	32	H'E82018E0	32
	Interrupt processor target register 57	ICDIPT57	32	H'E82018E4	32
	Interrupt processor target register 58	ICDIPT58	32	H'E82018E8	32
	Interrupt processor target register 59	ICDIPT59	32	H'E82018EC	32
	Interrupt processor target register 60	ICDIPT60	32	H'E82018F0	32
	Interrupt processor target register 61	ICDIPT61	32	H'E82018F4	32
	Interrupt processor target register 62	ICDIPT62	32	H'E82018F8	32
	Interrupt processor target register 63	ICDIPT63	32	H'E82018FC	32
	Interrupt processor target register 64	ICDIPT64	32	H'E8201900	32
	Interrupt processor target register 65	ICDIPT65	32	H'E8201904	32
	Interrupt processor target register 66	ICDIPT66	32	H'E8201908	32
	Interrupt processor target register 67	ICDIPT67	32	H'E820190C	32
	Interrupt processor target register 68	ICDIPT68	32	H'E8201910	32
	Interrupt processor target register 69	ICDIPT69	32	H'E8201914	32
	Interrupt processor target register 70	ICDIPT70	32	H'E8201918	32
	Interrupt processor target register 71	ICDIPT71	32	H'E820191C	32

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Interrupt controller	Interrupt processor target register 72	ICDIPTR72	32	H'E8201920	32
	Interrupt processor target register 73	ICDIPTR73	32	H'E8201924	32
	Interrupt processor target register 74	ICDIPTR74	32	H'E8201928	32
	Interrupt processor target register 75	ICDIPTR75	32	H'E820192C	32
	Interrupt processor target register 76	ICDIPTR76	32	H'E8201930	32
	Interrupt processor target register 77	ICDIPTR77	32	H'E8201934	32
	Interrupt processor target register 78	ICDIPTR78	32	H'E8201938	32
	Interrupt processor target register 79	ICDIPTR79	32	H'E820193C	32
	Interrupt processor target register 80	ICDIPTR80	32	H'E8201940	32
	Interrupt processor target register 81	ICDIPTR81	32	H'E8201944	32
	Interrupt processor target register 82	ICDIPTR82	32	H'E8201948	32
	Interrupt processor target register 83	ICDIPTR83	32	H'E820194C	32
	Interrupt processor target register 84	ICDIPTR84	32	H'E8201950	32
	Interrupt processor target register 85	ICDIPTR85	32	H'E8201954	32
	Interrupt processor target register 86	ICDIPTR86	32	H'E8201958	32
	Interrupt processor target register 87	ICDIPTR87	32	H'E820195C	32
	Interrupt processor target register 88	ICDIPTR88	32	H'E8201960	32
	Interrupt processor target register 89	ICDIPTR89	32	H'E8201964	32
	Interrupt processor target register 90	ICDIPTR90	32	H'E8201968	32
	Interrupt processor target register 91	ICDIPTR91	32	H'E820196C	32
	Interrupt processor target register 92	ICDIPTR92	32	H'E8201970	32
	Interrupt processor target register 93	ICDIPTR93	32	H'E8201974	32
	Interrupt processor target register 94	ICDIPTR94	32	H'E8201978	32
	Interrupt processor target register 95	ICDIPTR95	32	H'E820197C	32
	Interrupt processor target register 96	ICDIPTR96	32	H'E8201980	32
	Interrupt processor target register 97	ICDIPTR97	32	H'E8201984	32
	Interrupt processor target register 98	ICDIPTR98	32	H'E8201988	32
	Interrupt processor target register 99	ICDIPTR99	32	H'E820198C	32
	Interrupt processor target register 100	ICDIPTR100	32	H'E8201990	32
	Interrupt processor target register 101	ICDIPTR101	32	H'E8201994	32
	Interrupt processor target register 102	ICDIPTR102	32	H'E8201998	32
	Interrupt processor target register 103	ICDIPTR103	32	H'E820199C	32
	Interrupt processor target register 104	ICDIPTR104	32	H'E82019A0	32
	Interrupt processor target register 105	ICDIPTR105	32	H'E82019A4	32
	Interrupt processor target register 106	ICDIPTR106	32	H'E82019A8	32
	Interrupt processor target register 107	ICDIPTR107	32	H'E82019AC	32
	Interrupt processor target register 108	ICDIPTR108	32	H'E82019B0	32
	Interrupt processor target register 109	ICDIPTR109	32	H'E82019B4	32
	Interrupt processor target register 110	ICDIPTR110	32	H'E82019B8	32
	Interrupt processor target register 111	ICDIPTR111	32	H'E82019BC	32
	Interrupt processor target register 112	ICDIPTR112	32	H'E82019C0	32
	Interrupt processor target register 113	ICDIPTR113	32	H'E82019C4	32
	Interrupt processor target register 114	ICDIPTR114	32	H'E82019C8	32
	Interrupt processor target register 115	ICDIPTR115	32	H'E82019CC	32
	Interrupt processor target register 116	ICDIPTR116	32	H'E82019D0	32
	Interrupt processor target register 117	ICDIPTR117	32	H'E82019D4	32
Interrupt processor target register 118	ICDIPTR118	32	H'E82019D8	32	
Interrupt processor target register 119	ICDIPTR119	32	H'E82019DC	32	
Interrupt processor target register 120	ICDIPTR120	32	H'E82019E0	32	
Interrupt processor target register 121	ICDIPTR121	32	H'E82019E4	32	
Interrupt processor target register 122	ICDIPTR122	32	H'E82019E8	32	
Interrupt processor target register 123	ICDIPTR123	32	H'E82019EC	32	

**Table 57.1 Register Addresses**

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Interrupt controller	Interrupt processor target register 124	ICDIPTR124	32	H'E82019F0	32
	Interrupt processor target register 125	ICDIPTR125	32	H'E82019F4	32
	Interrupt processor target register 126	ICDIPTR126	32	H'E82019F8	32
	Interrupt processor target register 127	ICDIPTR127	32	H'E82019FC	32
	Interrupt processor target register 128	ICDIPTR128	32	H'E8201A00	32
	Interrupt processor target register 129	ICDIPTR129	32	H'E8201A04	32
	Interrupt processor target register 130	ICDIPTR130	32	H'E8201A08	32
	Interrupt processor target register 131	ICDIPTR131	32	H'E8201A0C	32
	Interrupt processor target register 132	ICDIPTR132	32	H'E8201A10	32
	Interrupt processor target register 133	ICDIPTR133	32	H'E8201A14	32
	Interrupt processor target register 134	ICDIPTR134	32	H'E8201A18	32
	Interrupt processor target register 135	ICDIPTR135	32	H'E8201A1C	32
	Interrupt processor target register 136	ICDIPTR136	32	H'E8201A20	32
	Interrupt processor target register 137	ICDIPTR137	32	H'E8201A24	32
	Interrupt processor target register 138	ICDIPTR138	32	H'E8201A28	32
	Interrupt processor target register 139	ICDIPTR139	32	H'E8201A2C	32
	Interrupt processor target register 140	ICDIPTR140	32	H'E8201A30	32
	Interrupt processor target register 141	ICDIPTR141	32	H'E8201A34	32
	Interrupt processor target register 142	ICDIPTR142	32	H'E8201A38	32
	Interrupt processor target register 143	ICDIPTR143	32	H'E8201A3C	32
	Interrupt processor target register 144	ICDIPTR144	32	H'E8201A40	32
	Interrupt processor target register 145	ICDIPTR145	32	H'E8201A44	32
	Interrupt processor target register 146	ICDIPTR146	32	H'E8201A48	32
	Interrupt configuration register 0	ICDICFR0	32	H'E8201C00	32
	Interrupt configuration register 1	ICDICFR1	32	H'E8201C04	32
	Interrupt configuration register 2	ICDICFR2	32	H'E8201C08	32
	Interrupt configuration register 3	ICDICFR3	32	H'E8201C0C	32
	Interrupt configuration register 4	ICDICFR4	32	H'E8201C10	32
	Interrupt configuration register 5	ICDICFR5	32	H'E8201C14	32
	Interrupt configuration register 6	ICDICFR6	32	H'E8201C18	32
	Interrupt configuration register 7	ICDICFR7	32	H'E8201C1C	32
	Interrupt configuration register 8	ICDICFR8	32	H'E8201C20	32
	Interrupt configuration register 9	ICDICFR9	32	H'E8201C24	32
	Interrupt configuration register 10	ICDICFR10	32	H'E8201C28	32
	Interrupt configuration register 11	ICDICFR11	32	H'E8201C2C	32
	Interrupt configuration register 12	ICDICFR12	32	H'E8201C30	32
	Interrupt configuration register 13	ICDICFR13	32	H'E8201C34	32
	Interrupt configuration register 14	ICDICFR14	32	H'E8201C38	32
	Interrupt configuration register 15	ICDICFR15	32	H'E8201C3C	32
	Interrupt configuration register 16	ICDICFR16	32	H'E8201C40	32
	Interrupt configuration register 17	ICDICFR17	32	H'E8201C44	32
	Interrupt configuration register 18	ICDICFR18	32	H'E8201C48	32
	Interrupt configuration register 19	ICDICFR19	32	H'E8201C4C	32
	Interrupt configuration register 20	ICDICFR20	32	H'E8201C50	32
	Interrupt configuration register 21	ICDICFR21	32	H'E8201C54	32
	Interrupt configuration register 22	ICDICFR22	32	H'E8201C58	32
	Interrupt configuration register 23	ICDICFR23	32	H'E8201C5C	32
	Interrupt configuration register 24	ICDICFR24	32	H'E8201C60	32
Interrupt configuration register 25	ICDICFR25	32	H'E8201C64	32	
Interrupt configuration register 26	ICDICFR26	32	H'E8201C68	32	
Interrupt configuration register 27	ICDICFR27	32	H'E8201C6C	32	
Interrupt configuration register 28	ICDICFR28	32	H'E8201C70	32	

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Interrupt controller	Interrupt configuration register 29	ICDICFR29	32	H'E8201C74	32
	Interrupt configuration register 30	ICDICFR30	32	H'E8201C78	32
	Interrupt configuration register 31	ICDICFR31	32	H'E8201C7C	32
	Interrupt configuration register 32	ICDICFR32	32	H'E8201C80	32
	Interrupt configuration register 33	ICDICFR33	32	H'E8201C84	32
	Interrupt configuration register 34	ICDICFR34	32	H'E8201C88	32
	Interrupt configuration register 35	ICDICFR35	32	H'E8201C8C	32
	Interrupt configuration register 36	ICDICFR36	32	H'E8201C90	32
	PPI status register	ppi_status	32	H'E8201D00	32
	SPI status register 0	spi_status0	32	H'E8201D04	32
	SPI status register 1	spi_status1	32	H'E8201D08	32
	SPI status register 2	spi_status2	32	H'E8201D0C	32
	SPI status register 3	spi_status3	32	H'E8201D10	32
	SPI status register 4	spi_status4	32	H'E8201D14	32
	SPI status register 5	spi_status5	32	H'E8201D18	32
	SPI status register 6	spi_status6	32	H'E8201D1C	32
	SPI status register 7	spi_status7	32	H'E8201D20	32
	SPI status register 8	spi_status8	32	H'E8201D24	32
	SPI status register 9	spi_status9	32	H'E8201D28	32
	SPI status register 10	spi_status10	32	H'E8201D2C	32
	SPI status register 11	spi_status11	32	H'E8201D30	32
	SPI status register 12	spi_status12	32	H'E8201D34	32
	SPI status register 13	spi_status13	32	H'E8201D38	32
	SPI status register 14	spi_status14	32	H'E8201D3C	32
	SPI status register 15	spi_status15	32	H'E8201D40	32
	SPI status register 16	spi_status16	32	H'E8201D44	32
	Software generation interrupt register	ICDSGIR	32	H'E8201F00	32
	CPU interface control register	ICCCICR	32	H'E8202000	32
	Interrupt priority mask register	ICCPMR	32	H'E8202004	32
	Binary point register	ICCBPR	32	H'E8202008	32
	Interrupt acknowledge register	ICCIAR	32	H'E820200C	32
	End-of-interrupt register	ICCEOIR	32	H'E8202010	32
	Running priority register	ICCRPR	32	H'E8202014	32
Highest pending interrupt register	ICCHPIR	32	H'E8202018	32	
Aliased binary point register	ICCABPR	32	H'E820201C	32	
CPU interface implementer identification register	ICCIIDR	32	H'E8202020	32	
Bus state controller	Common control register	CMNCR	32	H'3FFFC000	32
	CS0 space bus control register	CS0BCR	32	H'3FFFC004	32
	CS1 space bus control register	CS1BCR	32	H'3FFFC008	32
	CS2 space bus control register	CS2BCR	32	H'3FFFC00C	32
	CS3 space bus control register	CS3BCR	32	H'3FFFC010	32
	CS4 space bus control register	CS4BCR	32	H'3FFFC014	32
	CS5 space bus control register	CS5BCR	32	H'3FFFC018	32
	CS0 space wait control register	CS0WCR	32	H'3FFFC028	32
	CS1 space wait control register	CS1WCR	32	H'3FFFC02C	32
	CS2 space wait control register	CS2WCR	32	H'3FFFC030	32
	CS3 space wait control register	CS3WCR	32	H'3FFFC034	32
	CS4 space wait control register	CS4WCR	32	H'3FFFC038	32
	CS5 space wait control register	CS5WCR	32	H'3FFFC03C	32
	SDRAM control register	SDCR	32	H'3FFFC04C	32
	Refresh timer control/status register	RTCSCR	16	H'3FFFC050	32
	Refresh timer counter	RTCNT	16	H'3FFFC054	32



Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Bus state controller	Refresh time constant register	RTCOR	16	H'3FFFC058	32
	Timeout cycle constant register 0	TOSCOR0	32	H'3FFFC060	32
	Timeout cycle constant register 1	TOSCOR1	32	H'3FFFC064	32
	Timeout cycle constant register 2	TOSCOR2	32	H'3FFFC068	32
	Timeout cycle constant register 3	TOSCOR3	32	H'3FFFC06C	32
	Timeout cycle constant register 4	TOSCOR4	32	H'3FFFC070	32
	Timeout cycle constant register 5	TOSCOR5	32	H'3FFFC074	32
	Timeout status register	TOSTR	32	H'3FFFC080	32
	Timeout enable register	TOENR	32	H'3FFFC084	32
Direct memory access controller	Next0 Source Address Register 0	N0SA_0	32	H'E8200000	32
	Next0 Destination Address Register 0	N0DA_0	32	H'E8200004	32
	Next0 Transaction Byte Register 0	N0TB_0	32	H'E8200008	32
	Next1 Source Address Register 0	N1SA_0	32	H'E820000C	32
	Next1 Destination Address Register 0	N1DA_0	32	H'E8200010	32
	Next1 Transaction Byte Register 0	N1TB_0	32	H'E8200014	32
	Current Source Address Register 0	CRSA_0	32	H'E8200018	32
	Current Destination Address Register 0	CRDA_0	32	H'E820001C	32
	Current Transaction Byte Register 0	CRTB_0	32	H'E8200020	32
	Channel Status Register 0	CHSTAT_0	32	H'E8200024	32
	Channel Control Register 0	CHCTRL_0	32	H'E8200028	32
	Channel Configuration Register 0	CHCFG_0	32	H'E820002C	32
	Channel Interval Register 0	CHITVL_0	32	H'E8200030	32
	Channel Extension Register 0	CHEXT_0	32	H'E8200034	32
	Next Link Address Register 0	NXLA_0	32	H'E8200038	32
	Current Link Address Register 0	CRLA_0	32	H'E820003C	32
	Next0 Source Address Register 1	N0SA_1	32	H'E8200040	32
	Next0 Destination Address Register 1	N0DA_1	32	H'E8200044	32
	Next0 Transaction Byte Register 1	N0TB_1	32	H'E8200048	32
	Next1 Source Address Register 1	N1SA_1	32	H'E820004C	32
	Next1 Destination Address Register 1	N1DA_1	32	H'E8200050	32
	Next1 Transaction Byte Register 1	N1TB_1	32	H'E8200054	32
	Current Source Address Register 1	CRSA_1	32	H'E8200058	32
	Current Destination Address Register 1	CRDA_1	32	H'E820005C	32
	Current Transaction Byte Register 1	CRTB_1	32	H'E8200060	32
	Channel Status Register 1	CHSTAT_1	32	H'E8200064	32
	Channel Control Register 1	CHCTRL_1	32	H'E8200068	32
	Channel Configuration Register 1	CHCFG_1	32	H'E820006C	32
	Channel Interval Register 1	CHITVL_1	32	H'E8200070	32
	Channel Extension Register 1	CHEXT_1	32	H'E8200074	32
	Next Link Address Register 1	NXLA_1	32	H'E8200078	32
	Current Link Address Register 1	CRLA_1	32	H'E820007C	32
	Next0 Source Address Register 2	N0SA_2	32	H'E8200080	32
	Next0 Destination Address Register 2	N0DA_2	32	H'E8200084	32
	Next0 Transaction Byte Register 2	N0TB_2	32	H'E8200088	32
	Next1 Source Address Register 2	N1SA_2	32	H'E820008C	32
	Next1 Destination Address Register 2	N1DA_2	32	H'E8200090	32
	Next1 Transaction Byte Register 2	N1TB_2	32	H'E8200094	32
	Current Source Address Register 2	CRSA_2	32	H'E8200098	32
	Current Destination Address Register 2	CRDA_2	32	H'E820009C	32
Current Transaction Byte Register 2	CRTB_2	32	H'E82000A0	32	
Channel Status Register 2	CHSTAT_2	32	H'E82000A4	32	
Channel Control Register 2	CHCTRL_2	32	H'E82000A8	32	

**Table 57.1 Register Addresses**

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Direct memory access controller	Channel Configuration Register 2	CHCFG_2	32	H'E82000AC	32
	Channel Interval Register 2	CHITVL_2	32	H'E82000B0	32
	Channel Extension Register 2	CHEXT_2	32	H'E82000B4	32
	Next Link Address Resister 2	NXLA_2	32	H'E82000B8	32
	Current Link Address Resister 2	CRLA_2	32	H'E82000BC	32
	Next0 Source Address Register 3	N0SA_3	32	H'E82000C0	32
	Next0 Destination Address Register 3	N0DA_3	32	H'E82000C4	32
	Next0 Transaction Byte Register 3	N0TB_3	32	H'E82000C8	32
	Next1 Source Address Register 3	N1SA_3	32	H'E82000CC	32
	Next1 Destination Address Resister 3	N1DA_3	32	H'E82000D0	32
	Next1 Transaction Byte Register 3	N1TB_3	32	H'E82000D4	32
	Current Source Address Register 3	CRSA_3	32	H'E82000D8	32
	Current Destination Address Register 3	CRDA_3	32	H'E82000DC	32
	Current Transaction Byte Register 3	CRTB_3	32	H'E82000E0	32
	Channel Status Register 3	CHSTAT_3	32	H'E82000E4	32
	Channel Control Register 3	CHCTRL_3	32	H'E82000E8	32
	Channel Configuration Register 3	CHCFG_3	32	H'E82000EC	32
	Channel Interval Register 3	CHITVL_3	32	H'E82000F0	32
	Channel Extension Register 3	CHEXT_3	32	H'E82000F4	32
	Next Link Address Resister 3	NXLA_3	32	H'E82000F8	32
	Current Link Address Resister 3	CRLA_3	32	H'E82000FC	32
	Next0 Source Address Register 4	N0SA_4	32	H'E8200100	32
	Next0 Destination Address Register 4	N0DA_4	32	H'E8200104	32
	Next0 Transaction Byte Register 4	N0TB_4	32	H'E8200108	32
	Next1 Source Address Register 4	N1SA_4	32	H'E820010C	32
	Next1 Destination Address Resister 4	N1DA_4	32	H'E8200110	32
	Next1 Transaction Byte Register 4	N1TB_4	32	H'E8200114	32
	Current Source Address Register 4	CRSA_4	32	H'E8200118	32
	Current Destination Address Register 4	CRDA_4	32	H'E820011C	32
	Current Transaction Byte Register 4	CRTB_4	32	H'E8200120	32
	Channel Status Register 4	CHSTAT_4	32	H'E8200124	32
	Channel Control Register 4	CHCTRL_4	32	H'E8200128	32
	Channel Configuration Register 4	CHCFG_4	32	H'E820012C	32
	Channel Interval Register 4	CHITVL_4	32	H'E8200130	32
	Channel Extension Register 4	CHEXT_4	32	H'E8200134	32
	Next Link Address Resister 4	NXLA_4	32	H'E8200138	32
	Current Link Address Resister 4	CRLA_4	32	H'E820013C	32
	Next0 Source Address Register 5	N0SA_5	32	H'E8200140	32
	Next0 Destination Address Register 5	N0DA_5	32	H'E8200144	32
	Next0 Transaction Byte Register 5	N0TB_5	32	H'E8200148	32
	Next1 Source Address Register 5	N1SA_5	32	H'E820014C	32
	Next1 Destination Address Resister 5	N1DA_5	32	H'E8200150	32
	Next1 Transaction Byte Register 5	N1TB_5	32	H'E8200154	32
	Current Source Address Register 5	CRSA_5	32	H'E8200158	32
	Current Destination Address Register 5	CRDA_5	32	H'E820015C	32
	Current Transaction Byte Register 5	CRTB_5	32	H'E8200160	32
	Channel Status Register 5	CHSTAT_5	32	H'E8200164	32
	Channel Control Register 5	CHCTRL_5	32	H'E8200168	32
Channel Configuration Register 5	CHCFG_5	32	H'E820016C	32	
Channel Interval Register 5	CHITVL_5	32	H'E8200170	32	
Channel Extension Register 5	CHEXT_5	32	H'E8200174	32	
Next Link Address Resister 5	NXLA_5	32	H'E8200178	32	
Current Link Address Resister 5	CRLA_5	32	H'E820017C	32	

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Direct memory access controller	Next0 Source Address Register 6	N0SA_6	32	H'E8200180	32
	Next0 Destination Address Register 6	N0DA_6	32	H'E8200184	32
	Next0 Transaction Byte Register 6	N0TB_6	32	H'E8200188	32
	Next1 Source Address Register 6	N1SA_6	32	H'E820018C	32
	Next1 Destination Address Register 6	N1DA_6	32	H'E8200190	32
	Next1 Transaction Byte Register 6	N1TB_6	32	H'E8200194	32
	Current Source Address Register 6	CRSA_6	32	H'E8200198	32
	Current Destination Address Register 6	CRDA_6	32	H'E820019C	32
	Current Transaction Byte Register 6	CRTB_6	32	H'E82001A0	32
	Channel Status Register 6	CHSTAT_6	32	H'E82001A4	32
	Channel Control Register 6	CHCTRL_6	32	H'E82001A8	32
	Channel Configuration Register 6	CHCFG_6	32	H'E82001AC	32
	Channel Interval Register 6	CHITVL_6	32	H'E82001B0	32
	Channel Extension Register 6	CHEXT_6	32	H'E82001B4	32
	Next Link Address Register 6	NXLA_6	32	H'E82001B8	32
	Current Link Address Register 6	CRLA_6	32	H'E82001BC	32
	Next0 Source Address Register 7	N0SA_7	32	H'E82001C0	32
	Next0 Destination Address Register 7	N0DA_7	32	H'E82001C4	32
	Next0 Transaction Byte Register 7	N0TB_7	32	H'E82001C8	32
	Next1 Source Address Register 7	N1SA_7	32	H'E82001CC	32
	Next1 Destination Address Register 7	N1DA_7	32	H'E82001D0	32
	Next1 Transaction Byte Register 7	N1TB_7	32	H'E82001D4	32
	Current Source Address Register 7	CRSA_7	32	H'E82001D8	32
	Current Destination Address Register 7	CRDA_7	32	H'E82001DC	32
	Current Transaction Byte Register 7	CRTB_7	32	H'E82001E0	32
	Channel Status Register 7	CHSTAT_7	32	H'E82001E4	32
	Channel Control Register 7	CHCTRL_7	32	H'E82001E8	32
	Channel Configuration Register 7	CHCFG_7	32	H'E82001EC	32
	Channel Interval Register 7	CHITVL_7	32	H'E82001F0	32
	Channel Extension Register 7	CHEXT_7	32	H'E82001F4	32
	Next Link Address Register 7	NXLA_7	32	H'E82001F8	32
	Current Link Address Register 7	CRLA_7	32	H'E82001FC	32
	DMA Control Register 0-7	DCTRL_0_7	32	H'E8200300	32
	DMA Status EN Register 0-7	DSTAT_EN_0_7	32	H'E8200310	32
	DMA Status ER Register 0-7	DSTAT_ER_0_7	32	H'E8200314	32
	DMA Status END Register 0-7	DSTAT_END_0_7	32	H'E8200318	32
	DMA Status TC Register 0-7	DSTAT_TC_0_7	32	H'E820031C	32
	DMA Status SUS Register 0-7	DSTAT_SUS_0_7	32	H'E8200320	32
	Next0 Source Address Register 8	N0SA_8	32	H'E8200400	32
	Next0 Destination Address Register 8	N0DA_8	32	H'E8200404	32
	Next0 Transaction Byte Register 8	N0TB_8	32	H'E8200408	32
	Next1 Source Address Register 8	N1SA_8	32	H'E820040C	32
	Next1 Destination Address Register 8	N1DA_8	32	H'E8200410	32
	Next1 Transaction Byte Register 8	N1TB_8	32	H'E8200414	32
	Current Source Address Register 8	CRSA_8	32	H'E8200418	32
	Current Destination Address Register 8	CRDA_8	32	H'E820041C	32
	Current Transaction Byte Register 8	CRTB_8	32	H'E8200420	32
	Channel Status Register 8	CHSTAT_8	32	H'E8200424	32
	Channel Control Register 8	CHCTRL_8	32	H'E8200428	32
	Channel Configuration Register 8	CHCFG_8	32	H'E820042C	32
	Channel Interval Register 8	CHITVL_8	32	H'E8200430	32
	Channel Extension Register 8	CHEXT_8	32	H'E8200434	32
Next Link Address Register 8	NXLA_8	32	H'E8200438	32	

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Direct memory access controller	Current Link Address Register 8	CRLA_8	32	H'E820043C	32
	Next0 Source Address Register 9	NOSA_9	32	H'E8200440	32
	Next0 Destination Address Register 9	NODA_9	32	H'E8200444	32
	Next0 Transaction Byte Register 9	N0TB_9	32	H'E8200448	32
	Next1 Source Address Register 9	N1SA_9	32	H'E820044C	32
	Next1 Destination Address Register 9	N1DA_9	32	H'E8200450	32
	Next1 Transaction Byte Register 9	N1TB_9	32	H'E8200454	32
	Current Source Address Register 9	CRSA_9	32	H'E8200458	32
	Current Destination Address Register 9	CRDA_9	32	H'E820045C	32
	Current Transaction Byte Register 9	CRTB_9	32	H'E8200460	32
	Channel Status Register 9	CHSTAT_9	32	H'E8200464	32
	Channel Control Register 9	CHCTRL_9	32	H'E8200468	32
	Channel Configuration Register 9	CHCFG_9	32	H'E820046C	32
	Channel Interval Register 9	CHITVL_9	32	H'E8200470	32
	Channel Extension Register 9	CHEXT_9	32	H'E8200474	32
	Next Link Address Register 9	NXLA_9	32	H'E8200478	32
	Current Link Address Register 9	CRLA_9	32	H'E820047C	32
	Next0 Source Address Register 10	NOSA_10	32	H'E8200480	32
	Next0 Destination Address Register 10	NODA_10	32	H'E8200484	32
	Next0 Transaction Byte Register 10	N0TB_10	32	H'E8200488	32
	Next1 Source Address Register 10	N1SA_10	32	H'E820048C	32
	Next1 Destination Address Register 10	N1DA_10	32	H'E8200490	32
	Next1 Transaction Byte Register 10	N1TB_10	32	H'E8200494	32
	Current Source Address Register 10	CRSA_10	32	H'E8200498	32
	Current Destination Address Register 10	CRDA_10	32	H'E820049C	32
	Current Transaction Byte Register 10	CRTB_10	32	H'E82004A0	32
	Channel Status Register 10	CHSTAT_10	32	H'E82004A4	32
	Channel Control Register 10	CHCTRL_10	32	H'E82004A8	32
	Channel Configuration Register 10	CHCFG_10	32	H'E82004AC	32
	Channel Interval Register 10	CHITVL_10	32	H'E82004B0	32
	Channel Extension Register 10	CHEXT_10	32	H'E82004B4	32
	Next Link Address Register 10	NXLA_10	32	H'E82004B8	32
	Current Link Address Register 10	CRLA_10	32	H'E82004BC	32
	Next0 Source Address Register 11	NOSA_11	32	H'E82004C0	32
	Next0 Destination Address Register 11	NODA_11	32	H'E82004C4	32
	Next0 Transaction Byte Register 11	N0TB_11	32	H'E82004C8	32
	Next1 Source Address Register 11	N1SA_11	32	H'E82004CC	32
	Next1 Destination Address Register 11	N1DA_11	32	H'E82004D0	32
	Next1 Transaction Byte Register 11	N1TB_11	32	H'E82004D4	32
	Current Source Address Register 11	CRSA_11	32	H'E82004D8	32
	Current Destination Address Register 11	CRDA_11	32	H'E82004DC	32
	Current Transaction Byte Register 11	CRTB_11	32	H'E82004E0	32
	Channel Status Register 11	CHSTAT_11	32	H'E82004E4	32
	Channel Control Register 11	CHCTRL_11	32	H'E82004E8	32
	Channel Configuration Register 11	CHCFG_11	32	H'E82004EC	32
	Channel Interval Register 11	CHITVL_11	32	H'E82004F0	32
	Channel Extension Register 11	CHEXT_11	32	H'E82004F4	32
	Next Link Address Register 11	NXLA_11	32	H'E82004F8	32
Current Link Address Register 11	CRLA_11	32	H'E82004FC	32	
Next0 Source Address Register 12	NOSA_12	32	H'E8200500	32	
Next0 Destination Address Register 12	NODA_12	32	H'E8200504	32	
Next0 Transaction Byte Register 12	N0TB_12	32	H'E8200508	32	
Next1 Source Address Register 12	N1SA_12	32	H'E820050C	32	

**Table 57.1 Register Addresses**

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Direct memory access controller	Next1 Destination Address Register 12	N1DA_12	32	H'E8200510	32
	Next1 Transaction Byte Register 12	N1TB_12	32	H'E8200514	32
	Current Source Address Register 12	CRSA_12	32	H'E8200518	32
	Current Destination Address Register 12	CRDA_12	32	H'E820051C	32
	Current Transaction Byte Register 12	CRTB_12	32	H'E8200520	32
	Channel Status Register 12	CHSTAT_12	32	H'E8200524	32
	Channel Control Register 12	CHCTRL_12	32	H'E8200528	32
	Channel Configuration Register 12	CHCFG_12	32	H'E820052C	32
	Channel Interval Register 12	CHITVL_12	32	H'E8200530	32
	Channel Extension Register 12	CHEXT_12	32	H'E8200534	32
	Next Link Address Register 12	NXLA_12	32	H'E8200538	32
	Current Link Address Register 12	CRLA_12	32	H'E820053C	32
	Next0 Source Address Register 13	N0SA_13	32	H'E8200540	32
	Next0 Destination Address Register 13	N0DA_13	32	H'E8200544	32
	Next0 Transaction Byte Register 13	N0TB_13	32	H'E8200548	32
	Next1 Source Address Register 13	N1SA_13	32	H'E820054C	32
	Next1 Destination Address Register 13	N1DA_13	32	H'E8200550	32
	Next1 Transaction Byte Register 13	N1TB_13	32	H'E8200554	32
	Current Source Address Register 13	CRSA_13	32	H'E8200558	32
	Current Destination Address Register 13	CRDA_13	32	H'E820055C	32
	Current Transaction Byte Register 13	CRTB_13	32	H'E8200560	32
	Channel Status Register 13	CHSTAT_13	32	H'E8200564	32
	Channel Control Register 13	CHCTRL_13	32	H'E8200568	32
	Channel Configuration Register 13	CHCFG_13	32	H'E820056C	32
	Channel Interval Register 13	CHITVL_13	32	H'E8200570	32
	Channel Extension Register 13	CHEXT_13	32	H'E8200574	32
	Next Link Address Register 13	NXLA_13	32	H'E8200578	32
	Current Link Address Register 13	CRLA_13	32	H'E820057C	32
	Next0 Source Address Register 14	N0SA_14	32	H'E8200580	32
	Next0 Destination Address Register 14	N0DA_14	32	H'E8200584	32
	Next0 Transaction Byte Register 14	N0TB_14	32	H'E8200588	32
	Next1 Source Address Register 14	N1SA_14	32	H'E820058C	32
	Next1 Destination Address Register 14	N1DA_14	32	H'E8200590	32
	Next1 Transaction Byte Register 14	N1TB_14	32	H'E8200594	32
	Current Source Address Register 14	CRSA_14	32	H'E8200598	32
	Current Destination Address Register 14	CRDA_14	32	H'E820059C	32
	Current Transaction Byte Register 14	CRTB_14	32	H'E82005A0	32
	Channel Status Register 14	CHSTAT_14	32	H'E82005A4	32
	Channel Control Register 14	CHCTRL_14	32	H'E82005A8	32
	Channel Configuration Register 14	CHCFG_14	32	H'E82005AC	32
	Channel Interval Register 14	CHITVL_14	32	H'E82005B0	32
	Channel Extension Register 14	CHEXT_14	32	H'E82005B4	32
	Next Link Address Register 14	NXLA_14	32	H'E82005B8	32
	Current Link Address Register 14	CRLA_14	32	H'E82005BC	32
	Next0 Source Address Register 15	N0SA_15	32	H'E82005C0	32
	Next0 Destination Address Register 15	N0DA_15	32	H'E82005C4	32
	Next0 Transaction Byte Register 15	N0TB_15	32	H'E82005C8	32
	Next1 Source Address Register 15	N1SA_15	32	H'E82005CC	32
Next1 Destination Address Register 15	N1DA_15	32	H'E82005D0	32	
Next1 Transaction Byte Register 15	N1TB_15	32	H'E82005D4	32	
Current Source Address Register 15	CRSA_15	32	H'E82005D8	32	
Current Destination Address Register 15	CRDA_15	32	H'E82005DC	32	

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Direct memory access controller	Current Transaction Byte Register 15	CRTB_15	32	H'E82005E0	32
	Channel Status Register 15	CHSTAT_15	32	H'E82005E4	32
	Channel Control Register 15	CHCTRL_15	32	H'E82005E8	32
	Channel Configuration Register 15	CHCFG_15	32	H'E82005EC	32
	Channel Interval Register 15	CHITVL_15	32	H'E82005F0	32
	Channel Extension Register 15	CHEXT_15	32	H'E82005F4	32
	Next Link Address Resister 15	NXLA_15	32	H'E82005F8	32
	Current Link Address Resister 15	CRLA_15	32	H'E82005FC	32
	DMA Control Register 8-15	DCTRL_8_15	32	H'E8200700	32
	DMA Status EN Register 8-15	DSTAT_EN_8_15	32	H'E8200710	32
	DMA Status ER Register 8-15	DSTAT_ER_8_15	32	H'E8200714	32
	DMA Status END Register 8-15	DSTAT_END_8_15	32	H'E8200718	32
	DMA Status TC Register 8-15	DSTAT_TC_8_15	32	H'E820071C	32
	DMA Status SUS Register 8-15	DSTAT_SUS_8_15	32	H'E8200720	32
	DMA extended resource selector 0	DMARS0	32	H'FCFE1000	32
	DMA extended resource selector 1	DMARS1	32	H'FCFE1004	32
	DMA extended resource selector 2	DMARS2	32	H'FCFE1008	32
	DMA extended resource selector 3	DMARS3	32	H'FCFE100C	32
	DMA extended resource selector 4	DMARS4	32	H'FCFE1010	32
	DMA extended resource selector 5	DMARS5	32	H'FCFE1014	32
DMA extended resource selector 6	DMARS6	32	H'FCFE1018	32	
DMA extended resource selector 7	DMARS7	32	H'FCFE101C	32	
Multi-function timer pulse unit 2	Timer control register_0	TCR_0	8	H'FCFF0300	8
	Timer mode register_0	TMDR_0	8	H'FCFF0301	8
	Timer I/O control register H_0	TIORH_0	8	H'FCFF0302	8
	Timer I/O control register L_0	TIORL_0	8	H'FCFF0303	8
	Timer interrupt enable register_0	TIER_0	8	H'FCFF0304	8
	Timer status register_0	TSR_0	8	H'FCFF0305	8
	Timer counter_0	TCNT_0	16	H'FCFF0306	16
	Timer general register A_0	TGRA_0	16	H'FCFF0308	16
	Timer general register B_0	TGRB_0	16	H'FCFF030A	16
	Timer general register C_0	TGRC_0	16	H'FCFF030C	16
	Timer general register D_0	TGRD_0	16	H'FCFF030E	16
	Timer general register E_0	TGRE_0	16	H'FCFF0320	16
	Timer general register F_0	TGRF_0	16	H'FCFF0322	16
	Timer interrupt enable register 2_0	TIER2_0	8	H'FCFF0324	8
	Timer status register 2_0	TSR2_0	8	H'FCFF0325	8
	Timer buffer operation transfer mode register_0	TBTM_0	8	H'FCFF0326	8
	Timer control register_1	TCR_1	8	H'FCFF0380	8
	Timer mode register_1	TMDR_1	8	H'FCFF0381	8
	Timer I/O control register _1	TIOR_1	8	H'FCFF0382	8
	Timer interrupt enable register_1	TIER_1	8	H'FCFF0384	8
	Timer status register_1	TSR_1	8	H'FCFF0385	8
	Timer counter_1	TCNT_1	16	H'FCFF0386	16
	Timer general register A_1	TGRA_1	16	H'FCFF0388	16
	Timer general register B_1	TGRB_1	16	H'FCFF038A	16
	Timer input capture control register	TICCR	8	H'FCFF0390	8
	Timer control register_2	TCR_2	8	H'FCFF0000	8
	Timer mode register_2	TMDR_2	8	H'FCFF0001	8
	Timer I/O control register _2	TIOR_2	8	H'FCFF0002	8
	Timer interrupt enable register_2	TIER_2	8	H'FCFF0004	8
	Timer status register_2	TSR_2	8	H'FCFF0005	8

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Multi-function timer pulse unit 2	Timer counter_2	TCNT_2	16	H'FCFF0006	16
	Timer general register A_2	TGRA_2	16	H'FCFF0008	16
	Timer general register B_2	TGRB_2	16	H'FCFF000A	16
	Timer control register_3	TCR_3	8	H'FCFF0200	8
	Timer mode register_3	TMDR_3	8	H'FCFF0202	8
	Timer I/O control register H_3	TIORH_3	8	H'FCFF0204	8
	Timer I/O control register L_3	TIORL_3	8	H'FCFF0205	8
	Timer interrupt enable register_3	TIER_3	8	H'FCFF0208	8
	Timer status register_3	TSR_3	8	H'FCFF022C	8
	Timer counter_3	TCNT_3	16	H'FCFF0210	16
	Timer general register A_3	TGRA_3	16	H'FCFF0218	16
	Timer general register B_3	TGRB_3	16	H'FCFF021A	16
	Timer general register C_3	TGRC_3	16	H'FCFF0224	16
	Timer general register D_3	TGRD_3	16	H'FCFF0226	16
	Timer buffer operation transfer mode register_3	TBTM_3	8	H'FCFF0238	8
	Timer control register_4	TCR_4	8	H'FCFF0201	8
	Timer mode register_4	TMDR_4	8	H'FCFF0203	8
	Timer I/O control register H_4	TIORH_4	8	H'FCFF0206	8
	Timer I/O control register L_4	TIORL_4	8	H'FCFF0207	8
	Timer interrupt enable register_4	TIER_4	8	H'FCFF0209	8
	Timer status register_4	TSR_4	8	H'FCFF022D	8
	Timer counter_4	TCNT_4	16	H'FCFF0212	16
	Timer general register A_4	TGRA_4	16	H'FCFF021C	16
	Timer general register B_4	TGRB_4	16	H'FCFF021E	16
	Timer general register C_4	TGRC_4	16	H'FCFF0228	16
	Timer general register D_4	TGRD_4	16	H'FCFF022A	16
	Timer buffer operation transfer mode register_4	TBTM_4	8	H'FCFF0239	8
	Timer A/D converter start request control register	TADCR	16	H'FCFF0240	16
	Timer A/D converter start request cycle set register A_4	TADCORA_4	16	H'FCFF0244	16
	Timer A/D converter start request cycle set register B_4	TADCORB_4	16	H'FCFF0246	16
	Timer A/D converter start request cycle set buffer register A_4	TADCOBRA_4	16	H'FCFF0248	16
	Timer A/D converter start request cycle set buffer register B_4	TADCOBRA_4	16	H'FCFF024A	16
	Timer start register	TSTR	8	H'FCFF0280	8
	Timer synchronous register	TSYR	8	H'FCFF0281	8
	Timer read/write enable register	TRWER	8	H'FCFF0284	8
	Timer output master enable register	TOER	8	H'FCFF020A	8
	Timer output control register 1	TOCR1	8	H'FCFF020E	8
	Timer output control register 2	TOCR2	8	H'FCFF020F	8
	Timer gate control register	TGCR	8	H'FCFF020D	8
	Timer cycle data register	TCDR	16	H'FCFF0214	16
	Timer dead time data register	TDDR	16	H'FCFF0216	16
	Timer subcounter	TCNTS	16	H'FCFF0220	16
	Timer cycle buffer register	TCBR	16	H'FCFF0222	16
	Timer interrupt skipping set register	TITCR	8	H'FCFF0230	8
Timer interrupt skipping counter	TITCNT	8	H'FCFF0231	8	
Timer buffer transfer set register	TBTER	8	H'FCFF0232	8	
Timer dead time enable register	TDER	8	H'FCFF0234	8	
Timer waveform control register	TWCR	8	H'FCFF0260	8	
Timer output level buffer register	TOLBR	8	H'FCFF0236	8	
OS timer	OSTM0 compare register	OSTM0CMP	32	H'FCFEC000	32
	OSTM0 counter register	OSTM0CNT	32	H'FCFEC004	32
	OSTM0 count enable status register	OSTM0TE	8	H'FCFEC010	8

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
OS timer	OSTM0 count start trigger register	OSTM0TS	8	H'FCFEC014	8
	OSTM0 count stop trigger register	OSTM0TT	8	H'FCFEC018	8
	OSTM0 control register	OSTM0CTL	8	H'FCFEC020	8
	OSTM1 compare register	OSTM1CMP	32	H'FCFEC400	32
	OSTM1 counter register	OSTM1CNT	32	H'FCFEC404	32
	OSTM1 count enable status register	OSTM1TE	8	H'FCFEC410	8
	OSTM1 count start trigger register	OSTM1TS	8	H'FCFEC414	8
	OSTM1 count stop trigger register	OSTM1TT	8	H'FCFEC418	8
	OSTM1 control register	OSTM1CTL	8	H'FCFEC420	8
Watchdog timer	Watchdog timer counter	WTCNT	8	H'FCFE0002	16
	Watchdog timer control/status register	WTCSR	8	H'FCFE0000	16
	Watchdog reset control/status register	WRCSR	8	H'FCFE0004	16
Realtime clock	64-Hz counter	R64CNT	8	H'FCFF1000	8
	Second counter	RSECCNT	8	H'FCFF1002	8
	Minute counter	RMINCNT	8	H'FCFF1004	8
	Hour counter	RHRCNT	8	H'FCFF1006	8
	Day of week counter	RWKCNT	8	H'FCFF1008	8
	Day counter	RDAYCNT	8	H'FCFF100A	8
	Month counter	RMONCNT	8	H'FCFF100C	8
	Year counter	RYRCNT	16	H'FCFF100E	16
	Second alarm register	RSECAR	8	H'FCFF1010	8
	Minute alarm register	RMINAR	8	H'FCFF1012	8
	Hour alarm register	RHRAR	8	H'FCFF1014	8
	Day of week alarm register	RWKAR	8	H'FCFF1016	8
	Day alarm register	RDAYAR	8	H'FCFF1018	8
	Month alarm register	RMONAR	8	H'FCFF101A	8
	Year alarm register	RYRAR	16	H'FCFF1020	16
	Control register 1	RCR1	8	H'FCFF101C	8
	Control register 2	RCR2	8	H'FCFF101E	8
	Control register 3	RCR3	8	H'FCFF1024	8
	Control register 5	RCR5	8	H'FCFF1026	8
	Frequency register H	RFRH	16	H'FCFF102A	16
Frequency register L	RFRL	16	H'FCFF102C	16	
Serial communication interface with FIFO	Serial mode register_0	SCSMR_0	16	H'E8007000	16
	Bit rate register_0	SCBRR_0	8	H'E8007004	8
	Serial control register_0	SCSCR_0	16	H'E8007008	16
	Transmit FIFO data register_0	SCFTDR_0	8	H'E800700C	8
	Serial status register_0	SCFSR_0	16	H'E8007010	16
	Receive FIFO data register_0	SCFRDR_0	8	H'E8007014	8
	FIFO control register_0	SCFCR_0	16	H'E8007018	16
	FIFO data count set register_0	SCFDR_0	16	H'E800701C	16
	Serial port register_0	SCSPTR_0	16	H'E8007020	16
	Line status register_0	SCLSR_0	16	H'E8007024	16
	Serial extension mode register_0	SCEMR_0	16	H'E8007028	16
	Serial mode register_1	SCSMR_1	16	H'E8007800	16
	Bit rate register_1	SCBRR_1	8	H'E8007804	8
	Serial control register_1	SCSCR_1	16	H'E8007808	16
	Transmit FIFO data register_1	SCFTDR_1	8	H'E800780C	8
	Serial status register_1	SCFSR_1	16	H'E8007810	16
	Receive FIFO data register_1	SCFRDR_1	8	H'E8007814	8
	FIFO control register_1	SCFCR_1	16	H'E8007818	16
	FIFO data count set register_1	SCFDR_1	16	H'E800781C	16



**Table 57.1 Register Addresses**

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Serial communication interface with FIFO	Serial port register_1	SCSPTR_1	16	H'E8007820	16
	Line status register_1	SCLSR_1	16	H'E8007824	16
	Serial extension mode register_1	SCEMR_1	16	H'E8007828	16
	Serial mode register_2	SCSMR_2	16	H'E8008000	16
	Bit rate register_2	SCBRR_2	8	H'E8008004	8
	Serial control register_2	SCSCR_2	16	H'E8008008	16
	Transmit FIFO data register_2	SCFTDR_2	8	H'E800800C	8
	Serial status register_2	SCFSR_2	16	H'E8008010	16
	Receive FIFO data register_2	SCFRDR_2	8	H'E8008014	8
	FIFO control register_2	SCFCR_2	16	H'E8008018	16
	FIFO data count set register_2	SCFDR_2	16	H'E800801C	16
	Serial port register_2	SCSPTR_2	16	H'E8008020	16
	Line status register_2	SCLSR_2	16	H'E8008024	16
	Serial extension mode register_2	SCEMR_2	16	H'E8008028	16
	Serial mode register_3	SCSMR_3	16	H'E8008800	16
	Bit rate register_3	SCBRR_3	8	H'E8008804	8
	Serial control register_3	SCSCR_3	16	H'E8008808	16
	Transmit FIFO data register_3	SCFTDR_3	8	H'E800880C	8
	Serial status register_3	SCFSR_3	16	H'E8008810	16
	Receive FIFO data register_3	SCFRDR_3	8	H'E8008814	8
	FIFO control register_3	SCFCR_3	16	H'E8008818	16
	FIFO data count set register_3	SCFDR_3	16	H'E800881C	16
	Serial port register_3	SCSPTR_3	16	H'E8008820	16
	Line status register_3	SCLSR_3	16	H'E8008824	16
	Serial extension mode register_3	SCEMR_3	16	H'E8008828	16
	Serial mode register_4	SCSMR_4	16	H'E8009000	16
	Bit rate register_4	SCBRR_4	8	H'E8009004	8
	Serial control register_4	SCSCR_4	16	H'E8009008	16
	Transmit FIFO data register_4	SCFTDR_4	8	H'E800900C	8
	Serial status register_4	SCFSR_4	16	H'E8009010	16
	Receive FIFO data register_4	SCFRDR_4	8	H'E8009014	8
	FIFO control register_4	SCFCR_4	16	H'E8009018	16
	FIFO data count set register_4	SCFDR_4	16	H'E800901C	16
	Serial port register_4	SCSPTR_4	16	H'E8009020	16
	Line status register_4	SCLSR_4	16	H'E8009024	16
	Serial extension mode register_4	SCEMR_4	16	H'E8009028	16
	Serial mode register_5	SCSMR_5	16	H'E8009800	16
	Bit rate register_5	SCBRR_5	8	H'E8009804	8
	Serial control register_5	SCSCR_5	16	H'E8009808	16
	Transmit FIFO data register_5	SCFTDR_5	8	H'E800980C	8
	Serial status register_5	SCFSR_5	16	H'E8009810	16
	Receive FIFO data register_5	SCFRDR_5	8	H'E8009814	8
FIFO control register_5	SCFCR_5	16	H'E8009818	16	
FIFO data count set register_5	SCFDR_5	16	H'E800981C	16	
Serial port register_5	SCSPTR_5	16	H'E8009820	16	
Line status register_5	SCLSR_5	16	H'E8009824	16	
Serial extension mode register_5	SCEMR_5	16	H'E8009828	16	
Serial mode register_6	SCSMR_6	16	H'E800A000	16	
Bit rate register_6	SCBRR_6	8	H'E800A004	8	
Serial control register_6	SCSCR_6	16	H'E800A008	16	
Transmit FIFO data register_6	SCFTDR_6	8	H'E800A00C	8	
Serial status register_6	SCFSR_6	16	H'E800A010	16	

**Table 57.1 Register Addresses**

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Serial communication interface with FIFO	Receive FIFO data register_6	SCFRDR_6	8	H'E800A014	8
	FIFO control register_6	SCFCR_6	16	H'E800A018	16
	FIFO data count set register_6	SCFDR_6	16	H'E800A01C	16
	Serial port register_6	SCSPTR_6	16	H'E800A020	16
	Line status register_6	SCLSR_6	16	H'E800A024	16
	Serial extension mode register_6	SCEMR_6	16	H'E800A028	16
	Serial mode register_7	SCSMR_7	16	H'E800A800	16
	Bit rate register_7	SCBRR_7	8	H'E800A804	8
	Serial control register_7	SCSCR_7	16	H'E800A808	16
	Transmit FIFO data register_7	SCFTDR_7	8	H'E800A80C	8
	Serial status register_7	SCFSR_7	16	H'E800A810	16
	Receive FIFO data register_7	SCFRDR_7	8	H'E800A814	8
	FIFO control register_7	SCFCR_7	16	H'E800A818	16
	FIFO data count set register_7	SCFDR_7	16	H'E800A81C	16
	Serial port register_7	SCSPTR_7	16	H'E800A820	16
	Line status register_7	SCLSR_7	16	H'E800A824	16
	Serial extension mode register_7	SCEMR_7	16	H'E800A828	16
Serial communications interface	Serial mode register 0	SMR0	8	H'E800B000	8
	Bit rate register 0	BRR0	8	H'E800B001	8
	Serial control register 0	SCR0	8	H'E800B002	8
	Transmit data register 0	TDR0	8	H'E800B003	8
	Serial status register 0	SSR0	8	H'E800B004	8
	Receive data register 0	RDR0	8	H'E800B005	8
	Smart card mode register 0	SCMR0	8	H'E800B006	8
	Serial extended mode register 0	SEMR0	8	H'E800B007	8
	Noise filter setting register 0	SNFR0	8	H'E800B008	8
	Extended function control register 0	SECR0	8	H'E800B00D	8
	Serial mode register 1	SMR1	8	H'E800B800	8
	Bit rate register 1	BRR1	8	H'E800B801	8
	Serial control register 1	SCR1	8	H'E800B802	8
	Transmit data register 1	TDR1	8	H'E800B803	8
	Serial status register 1	SSR1	8	H'E800B804	8
	Receive data register 1	RDR1	8	H'E800B805	8
	Smart card mode register 1	SCMR1	8	H'E800B806	8
	Serial extended mode register 1	SEMR1	8	H'E800B807	8
	Noise filter setting register 1	SNFR1	8	H'E800B808	8
	Extended function control register 1	SECR1	8	H'E800B80D	8
IrDA control register	IRCR	8	H'E8014000	8	

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Renesas serial peripheral interface	Control register_0	SPCR_0	8	H'E800C800	8
	Slave select polarity register_0	SSLP_0	8	H'E800C801	8
	Pin control register_0	SPPCR_0	8	H'E800C802	8
	Status register_0	SPSR_0	8	H'E800C803	8
	Data register_0	SPDR_0	32	H'E800C804	8, 16, 32
	Sequence control register_0	SPSCR_0	8	H'E800C808	8
	Sequence status register_0	SPSSR_0	8	H'E800C809	8
	Bit rate register_0	SPBR_0	8	H'E800C80A	8
	Data control register_0	SPDCR_0	8	H'E800C80B	8
	Clock delay register_0	SPCKD_0	8	H'E800C80C	8
	Slave select negation delay register_0	SSLND_0	8	H'E800C80D	8
	Next-access delay register_0	SPND_0	8	H'E800C80E	8
	Command register 0_0	SPCMD0_0	16	H'E800C810	16
	Command register 1_0	SPCMD1_0	16	H'E800C812	16
	Command register 2_0	SPCMD2_0	16	H'E800C814	16
	Command register 3_0	SPCMD3_0	16	H'E800C816	16
	Buffer control register_0	SPBFCR_0	8	H'E800C820	8
	Buffer data count setting register_0	SPBFDR_0	16	H'E800C822	16
	Control register_1	SPCR_1	8	H'E800D000	8
	Slave select polarity register_1	SSLP_1	8	H'E800D001	8
	Pin control register_1	SPPCR_1	8	H'E800D002	8
	Status register_1	SPSR_1	8	H'E800D003	8
	Data register_1	SPDR_1	32	H'E800D004	8, 16, 32
	Sequence control register_1	SPSCR_1	8	H'E800D008	8
	Sequence status register_1	SPSSR_1	8	H'E800D009	8
	Bit rate register_1	SPBR_1	8	H'E800D00A	8
	Data control register_1	SPDCR_1	8	H'E800D00B	8
	Clock delay register_1	SPCKD_1	8	H'E800D00C	8
	Slave select negation delay register_1	SSLND_1	8	H'E800D00D	8
	Next-access delay register_1	SPND_1	8	H'E800D00E	8
	Command register 0_1	SPCMD0_1	16	H'E800D010	16
	Command register 1_1	SPCMD1_1	16	H'E800D012	16
	Command register 2_1	SPCMD2_1	16	H'E800D014	16
	Command register 3_1	SPCMD3_1	16	H'E800D016	16
	Buffer control register_1	SPBFCR_1	8	H'E800D020	8
	Buffer data count setting register_1	SPBFDR_1	16	H'E800D022	16
	Control register_2	SPCR_2	8	H'E800D800	8
	Slave select polarity register_2	SSLP_2	8	H'E800D801	8
	Pin control register_2	SPPCR_2	8	H'E800D802	8
	Status register_2	SPSR_2	8	H'E800D803	8
	Data register_2	SPDR_2	32	H'E800D804	8, 16, 32
	Sequence control register_2	SPSCR_2	8	H'E800D808	8
	Sequence status register_2	SPSSR_2	8	H'E800D809	8
	Bit rate register_2	SPBR_2	8	H'E800D80A	8
Data control register_2	SPDCR_2	8	H'E800D80B	8	
Clock delay register_2	SPCKD_2	8	H'E800D80C	8	
Slave select negation delay register_2	SSLND_2	8	H'E800D80D	8	

**Table 57.1 Register Addresses**

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Renesas serial peripheral interface	Next-access delay register_2	SPND_2	8	H'E800D80E	8
	Command register 0_2	SPCMD0_2	16	H'E800D810	16
	Command register 1_2	SPCMD1_2	16	H'E800D812	16
	Command register 2_2	SPCMD2_2	16	H'E800D814	16
	Command register 3_2	SPCMD3_2	16	H'E800D816	16
	Buffer control register_2	SPBFCR_2	8	H'E800D820	8
	Buffer data count setting register_2	SPBFDR_2	16	H'E800D822	16
	Control register_3	SPCR_3	8	H'E800E000	8
	Slave select polarity register_3	SSLP_3	8	H'E800E001	8
	Pin control register_3	SPPCR_3	8	H'E800E002	8
	Status register_3	SPSR_3	8	H'E800E003	8
	Data register_3	SPDR_3	32	H'E800E004	8, 16, 32
	Sequence control register_3	SPSCR_3	8	H'E800E008	8
	Sequence status register_3	SPSSR_3	8	H'E800E009	8
	Bit rate register_3	SPBR_3	8	H'E800E00A	8
	Data control register_3	SPDCR_3	8	H'E800E00B	8
	Clock delay register_3	SPCKD_3	8	H'E800E00C	8
	Slave select negation delay register_3	SSLND_3	8	H'E800E00D	8
	Next-access delay register_3	SPND_3	8	H'E800E00E	8
	Command register 0_3	SPCMD0_3	16	H'E800E010	16
	Command register 1_3	SPCMD1_3	16	H'E800E012	16
	Command register 2_3	SPCMD2_3	16	H'E800E014	16
	Command register 3_3	SPCMD3_3	16	H'E800E016	16
	Buffer control register_3	SPBFCR_3	8	H'E800E020	8
	Buffer data count setting register_3	SPBFDR_3	16	H'E800E022	16
	Control register_4	SPCR_4	8	H'E800E800	8
	Slave select polarity register_4	SSLP_4	8	H'E800E801	8
	Pin control register_4	SPPCR_4	8	H'E800E802	8
	Status register_4	SPSR_4	8	H'E800E803	8
	Data register_4	SPDR_4	32	H'E800E804	8, 16, 32
	Sequence control register_4	SPSCR_4	8	H'E800E808	8
	Sequence status register_4	SPSSR_4	8	H'E800E809	8
	Bit rate register_4	SPBR_4	8	H'E800E80A	8
	Data control register_4	SPDCR_4	8	H'E800E80B	8
	Clock delay register_4	SPCKD_4	8	H'E800E80C	8
	Slave select negation delay register_4	SSLND_4	8	H'E800E80D	8
	Next-access delay register_4	SPND_4	8	H'E800E80E	8
	Command register 0_4	SPCMD0_4	16	H'E800E810	16
	Command register 1_4	SPCMD1_4	16	H'E800E812	16
	Command register 2_4	SPCMD2_4	16	H'E800E814	16
	Command register 3_4	SPCMD3_4	16	H'E800E818	16
	Buffer control register_4	SPBFCR_4	8	H'E800E820	8
	Buffer data count setting register_4	SPBFDR_4	16	H'E800E822	16

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
SPI multi I/O bus controller	Common control register_0	CMNCR_0	32	H'3FEFA000	32
	SSL delay register_0	SSLDR_0	32	H'3FEFA004	32
	Bit rate register_0	SPBCR_0	32	H'3FEFA008	32
	Data read control register_0	DRCR_0	32	H'3FEFA00C	32
	Data read command setting register_0	DRCMR_0	32	H'3FEFA010	32
	Data read extended address setting register_0	DREAR_0	32	H'3FEFA014	32
	Data read option setting register_0	DROPR_0	32	H'3FEFA018	32
	Data read enable setting register_0	DREN_R_0	32	H'3FEFA01C	32
	SPI mode control register_0	SMCR_0	32	H'3FEFA020	32
	SPI mode command setting register_0	SMCMR_0	32	H'3FEFA024	32
	SPI mode address setting register_0	SMADR_0	32	H'3FEFA028	32
	SPI mode option setting register_0	SMOPR_0	32	H'3FEFA02C	32
	SPI mode enable setting register_0	SMENR_0	32	H'3FEFA030	32
	SPI mode read data register 0_0	SMRDR0_0	32	H'3FEFA038	8, 16, 32
	SPI mode read data register 1_0	SMRDR1_0	32	H'3FEFA03C	8, 16, 32
	SPI mode write data register 0_0	SMWDR0_0	32	H'3FEFA040	8, 16, 32
	SPI mode write data register 1_0	SMWDR1_0	32	H'3FEFA044	8, 16, 32
	Common status register_0	CMNSR_0	32	H'3FEFA048	32
	SPI AC input characteristics adjustment register_0	CKDLY_0	32	H'3FEFA050	32
	Data read dummy cycle setting register_0	DRDMCR_0	32	H'3FEFA058	32
	Data read DDR enable register_0	DRDREN_R_0	32	H'3FEFA05C	32
	SPI mode dummy cycle setting register_0	SMDMCR_0	32	H'3FEFA060	32
	SPI mode DDR enable register_0	SMDREN_R_0	32	H'3FEFA064	32
	SPI AC output characteristics adjustment register_0	SPODLY_0	32	H'3FEFA068	32
	Common control register_1	CMNCR_1	32	H'3FEFB000	32
	SSL delay register_1	SSLDR_1	32	H'3FEFB004	32
	Bit rate register_1	SPBCR_1	32	H'3FEFB008	32
	Data read control register_1	DRCR_1	32	H'3FEFB00C	32
	Data read command setting register_1	DRCMR_1	32	H'3FEFB010	32
	Data read extended address setting register_1	DREAR_1	32	H'3FEFB014	32
	Data read option setting register_1	DROPR_1	32	H'3FEFB018	32
	Data read enable setting register_1	DREN_R_1	32	H'3FEFB01C	32
	SPI mode control register_1	SMCR_1	32	H'3FEFB020	32
	SPI mode command setting register_1	SMCMR_1	32	H'3FEFB024	32
	SPI mode address setting register_1	SMADR_1	32	H'3FEFB028	32
	SPI mode option setting register_1	SMOPR_1	32	H'3FEFB02C	32
	SPI mode enable setting register_1	SMENR_1	32	H'3FEFB030	32
	SPI mode read data register 0_1	SMRDR0_1	32	H'3FEFB038	8, 16, 32
	SPI mode read data register 1_1	SMRDR1_1	32	H'3FEFB03C	8, 16, 32
	SPI mode write data register 0_1	SMWDR0_1	32	H'3FEFB040	8, 16, 32
	SPI mode write data register 1_1	SMWDR1_1	32	H'3FEFB044	8, 16, 32
	Common status register_1	CMNSR_1	32	H'3FEFB048	32
	SPI AC input characteristics adjustment register_1	CKDLY_1	32	H'3FEFA050	32
	Data read dummy cycle setting register_1	DRDMCR_1	32	H'3FEFB058	32
Data read DDR enable register_1	DRDREN_R_1	32	H'3FEFB05C	32	
SPI mode dummy cycle setting register_1	SMDMCR_1	32	H'3FEFB060	32	
SPI mode DDR enable register_1	SMDREN_R_1	32	H'3FEFB064	32	
SPI AC output characteristics adjustment register_1	SPODLY_1	32	H'3FEFA068	32	
I <sup>2</sup> C bus interface	I <sup>2</sup> C bus control register 1_0	RIIC0CR1	32	H'FCFEE000	8, 16, 32
	I <sup>2</sup> C bus control register 2_0	RIIC0CR2	32	H'FCFEE004	8, 16, 32
	I <sup>2</sup> C bus mode register 1_0	RIIC0MR1	32	H'FCFEE008	8, 16, 32
	I <sup>2</sup> C bus mode register 2_0	RIIC0MR2	32	H'FCFEE00C	8, 16, 32

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
I <sup>2</sup> C bus interface	I <sup>2</sup> C bus mode register 3_0	RIIC0MR3	32	H'FCFEE010	8, 16, 32
	I <sup>2</sup> C bus function enable register_0	RIIC0FER	32	H'FCFEE014	8, 16, 32
	I <sup>2</sup> C bus status enable register_0	RIIC0SER	32	H'FCFEE018	8, 16, 32
	I <sup>2</sup> C bus interrupt enable register_0	RIIC0IER	32	H'FCFEE01C	8, 16, 32
	I <sup>2</sup> C bus status register 1_0	RIIC0SR1	32	H'FCFEE020	8, 16, 32
	I <sup>2</sup> C bus status register 2_0	RIIC0SR2	32	H'FCFEE024	8, 16, 32
	I <sup>2</sup> C slave address register 0_0	RIIC0SAR0	32	H'FCFEE028	8, 16, 32
	I <sup>2</sup> C slave address register 1_0	RIIC0SAR1	32	H'FCFEE02C	8, 16, 32
	I <sup>2</sup> C slave address register 2_0	RIIC0SAR2	32	H'FCFEE030	8, 16, 32
	I <sup>2</sup> C bus bit rate low-level register_0	RIIC0BRL	32	H'FCFEE034	8, 16, 32
	I <sup>2</sup> C bus bit rate high-level register_0	RIIC0BRH	32	H'FCFEE038	8, 16, 32
	I <sup>2</sup> C bus transmit data register_0	RIIC0DRT	32	H'FCFEE03C	8, 16, 32
	I <sup>2</sup> C bus transmit data register_0	RIIC0DRR	32	H'FCFEE040	8, 16, 32
	I <sup>2</sup> C bus control register 1_1	RIIC1CR1	32	H'FCFEE400	8, 16, 32
	I <sup>2</sup> C bus control register 2_1	RIIC1CR2	32	H'FCFEE404	8, 16, 32
	I <sup>2</sup> C bus mode register 1_1	RIIC1MR1	32	H'FCFEE408	8, 16, 32
	I <sup>2</sup> C bus mode register 2_1	RIIC1MR2	32	H'FCFEE40C	8, 16, 32
	I <sup>2</sup> C bus mode register 3_1	RIIC1MR3	32	H'FCFEE410	8, 16, 32
	I <sup>2</sup> C bus function enable register_1	RIIC1FER	32	H'FCFEE414	8, 16, 32
	I <sup>2</sup> C bus status enable register_1	RIIC1SER	32	H'FCFEE418	8, 16, 32
	I <sup>2</sup> C bus interrupt enable register_1	RIIC1IER	32	H'FCFEE41C	8, 16, 32
	I <sup>2</sup> C bus status register 1_1	RIIC1SR1	32	H'FCFEE420	8, 16, 32
	I <sup>2</sup> C bus status register 2_1	RIIC1SR2	32	H'FCFEE424	8, 16, 32
	I <sup>2</sup> C slave address register 0_1	RIIC1SAR0	32	H'FCFEE428	8, 16, 32
	I <sup>2</sup> C slave address register 1_1	RIIC1SAR1	32	H'FCFEE42C	8, 16, 32
	I <sup>2</sup> C slave address register 2_1	RIIC1SAR2	32	H'FCFEE430	8, 16, 32
	I <sup>2</sup> C bus bit rate low-level register_1	RIIC1BRL	32	H'FCFEE434	8, 16, 32
	I <sup>2</sup> C bus bit rate high-level register_1	RIIC1BRH	32	H'FCFEE438	8, 16, 32
	I <sup>2</sup> C bus transmit data register_1	RIIC1DRT	32	H'FCFEE43C	8, 16, 32
	I <sup>2</sup> C bus receive data register_1	RIIC1DRR	32	H'FCFEE440	8, 16, 32
	I <sup>2</sup> C bus control register 1_2	RIIC2CR1	32	H'FCFEE800	8, 16, 32
	I <sup>2</sup> C bus control register 2_2	RIIC2CR2	32	H'FCFEE804	8, 16, 32
	I <sup>2</sup> C bus mode register 1_2	RIIC2MR1	32	H'FCFEE808	8, 16, 32
	I <sup>2</sup> C bus mode register 2_2	RIIC2MR2	32	H'FCFEE80C	8, 16, 32
	I <sup>2</sup> C bus mode register 3_2	RIIC2MR3	32	H'FCFEE810	8, 16, 32
	I <sup>2</sup> C bus function enable register_2	RIIC2FER	32	H'FCFEE814	8, 16, 32
	I <sup>2</sup> C bus status enable register_2	RIIC2SER	32	H'FCFEE818	8, 16, 32
	I <sup>2</sup> C bus interrupt enable register_2	RIIC2IER	32	H'FCFEE81C	8, 16, 32
	I <sup>2</sup> C bus status register 1_2	RIIC2SR1	32	H'FCFEE820	8, 16, 32
	I <sup>2</sup> C bus status register 2_2	RIIC2SR2	32	H'FCFEE824	8, 16, 32
	I <sup>2</sup> C slave address register 0_2	RIIC2SAR0	32	H'FCFEE828	8, 16, 32
	I <sup>2</sup> C slave address register 1_2	RIIC2SAR1	32	H'FCFEE82C	8, 16, 32
	I <sup>2</sup> C slave address register 2_2	RIIC2SAR2	32	H'FCFEE830	8, 16, 32
	I <sup>2</sup> C bus bit rate low-level register_2	RIIC2BRL	32	H'FCFEE834	8, 16, 32
I <sup>2</sup> C bus bit rate high-level register_2	RIIC2BRH	32	H'FCFEE838	8, 16, 32	
I <sup>2</sup> C bus transmit data register_2	RIIC2DRT	32	H'FCFEE83C	8, 16, 32	
I <sup>2</sup> C bus receive data register_2	RIIC2DRR	32	H'FCFEE840	8, 16, 32	
I <sup>2</sup> C bus control register 1_3	RIIC3CR1	32	H'FCFEEC00	8, 16, 32	
I <sup>2</sup> C bus control register 2_3	RIIC3CR2	32	H'FCFEEC04	8, 16, 32	
I <sup>2</sup> C bus mode register 1_3	RIIC3MR1	32	H'FCFEEC08	8, 16, 32	
I <sup>2</sup> C bus mode register 2_3	RIIC3MR2	32	H'FCFEEC0C	8, 16, 32	
I <sup>2</sup> C bus mode register 3_3	RIIC3MR3	32	H'FCFEEC10	8, 16, 32	

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
I <sup>2</sup> C bus interface	I <sup>2</sup> C bus function enable register_3	RIIC3FER	32	H'FCFEEC14	8, 16, 32
	I <sup>2</sup> C bus status enable register_3	RIIC3SER	32	H'FCFEEC18	8, 16, 32
	I <sup>2</sup> C bus interrupt enable register_3	RIIC3IER	32	H'FCFEEC1C	8, 16, 32
	I <sup>2</sup> C bus status register 1_3	RIIC3SR1	32	H'FCFEEC20	8, 16, 32
	I <sup>2</sup> C bus status register 2_3	RIIC3SR2	32	H'FCFEEC24	8, 16, 32
	I <sup>2</sup> C slave address register 0_3	RIIC3SAR0	32	H'FCFEEC28	8, 16, 32
	I <sup>2</sup> C slave address register 1_3	RIIC3SAR1	32	H'FCFEEC2C	8, 16, 32
	I <sup>2</sup> C slave address register 2_3	RIIC3SAR2	32	H'FCFEEC30	8, 16, 32
	I <sup>2</sup> C bus bit rate low-level register_3	RIIC3BRL	32	H'FCFEEC34	8, 16, 32
	I <sup>2</sup> C bus bit rate high-level register_3	RIIC3BRH	32	H'FCFEEC38	8, 16, 32
	I <sup>2</sup> C bus transmit data register_3	RIIC3DRT	32	H'FCFEEC3C	8, 16, 32
	I <sup>2</sup> C bus receive data register_3	RIIC3DRR	32	H'FCFEEC40	8, 16, 32
Serial sound interface	Control register_0	SSICR_0	32	H'E820B000	32
	Status register_0	SSISR_0	32	H'E820B004	32
	FIFO control register_0	SSIFCR_0	32	H'E820B010	32
	FIFO status register_0	SSIFSR_0	32	H'E820B014	32
	Transmit FIFO data register_0	SSIFTDR_0	32	H'E820B018	32
	Receive FIFO data register_0	SSIFRDR_0	32	H'E820B01C	32
	TDM mode register_0	SSITDMR_0	32	H'E820B020	32
	FC control register_0	SSIFCCR_0	32	H'E820B024	32
	FC mode register_0	SSIFCMR_0	32	H'E820B028	32
	FC status register_0	SSIFCSR_0	32	H'E820B02C	32
	Control register_1	SSICR_1	32	H'E820B800	32
	Status register_1	SSISR_1	32	H'E820B804	32
	FIFO control register_1	SSIFCR_1	32	H'E820B810	32
	FIFO status register_1	SSIFSR_1	32	H'E820B814	32
	Transmit FIFO data register_1	SSIFTDR_1	32	H'E820B818	32
	Receive FIFO data register_1	SSIFRDR_1	32	H'E820B81C	32
	TDM mode register_1	SSITDMR_1	32	H'E820B820	32
	FC control register_1	SSIFCCR_1	32	H'E820B824	32
	FC mode register_1	SSIFCMR_1	32	H'E820B828	32
	FC status register_1	SSIFCSR_1	32	H'E820B82C	32
	Control register_2	SSICR_2	32	H'E820C000	32
	Status register_2	SSISR_2	32	H'E820C004	32
	FIFO control register_2	SSIFCR_2	32	H'E820C010	32
	FIFO status register_2	SSIFSR_2	32	H'E820C014	32
	Transmit FIFO data register_2	SSIFTDR_2	32	H'E820C018	32
	Receive FIFO data register_2	SSIFRDR_2	32	H'E820C01C	32
	TDM mode register_2	SSITDMR_2	32	H'E820C020	32
	FC control register_2	SSIFCCR_2	32	H'E820C024	32
	FC mode register_2	SSIFCMR_2	32	H'E820C028	32
	FC status register_2	SSIFCSR_2	32	H'E820C02C	32
	Control register_3	SSICR_3	32	H'E820C800	32
	Status register_3	SSISR_3	32	H'E820C804	32
	FIFO control register_3	SSIFCR_3	32	H'E820C810	32
	FIFO status register_3	SSIFSR_3	32	H'E820C814	32
	Transmit FIFO data register_3	SSIFTDR_3	32	H'E820C818	32
	Receive FIFO data register_3	SSIFRDR_3	32	H'E820C81C	32
TDM mode register_3	SSITDMR_3	32	H'E820C820	32	
FC control register_3	SSIFCCR_3	32	H'E820C824	32	
FC mode register_3	SSIFCMR_3	32	H'E820C828	32	
FC status register_3	SSIFCSR_3	32	H'E820C82C	32	

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Serial sound interface	Control register_4	SSICR_4	32	H'E820D000	32
	Status register_4	SSISR_4	32	H'E820D004	32
	FIFO control register_4	SSIFCR_4	32	H'E820D010	32
	FIFO status register_4	SSIFSR_4	32	H'E820D014	32
	Transmit FIFO data register_4	SSIFTDR_4	32	H'E820D018	32
	Receive FIFO data register_4	SSIFRDR_4	32	H'E820D01C	32
	TDM mode register_4	SSITDMR_4	32	H'E820D020	32
	FC control register_4	SSIFCCR_4	32	H'E820D024	32
	FC mode register_4	SSIFCMR_4	32	H'E820D028	32
	FC status register_4	SSIFCSR_4	32	H'E820D02C	32
	Control register_5	SSICR_5	32	H'E820D800	32
	Status register_5	SSISR_5	32	H'E820D804	32
	FIFO control register_5	SSIFCR_5	32	H'E820D810	32
	FIFO status register_5	SSIFSR_5	32	H'E820D814	32
	Transmit FIFO data register_5	SSIFTDR_5	32	H'E820D818	32
	Receive FIFO data register_5	SSIFRDR_5	32	H'E820D81C	32
	TDM mode register_5	SSITDMR_5	32	H'E820D820	32
	FC control register_5	SSIFCCR_5	32	H'E820D824	32
	FC mode register_5	SSIFCMR_5	32	H'E820D828	32
	FC status register_5	SSIFCSR_5	32	H'E820D82C	32
Media local bus	Device Control Cfg Register	DCCR	32	H'E8034000	32
	System Status Cfg Register	SSCR	32	H'E8034004	32
	System Data Cfg Register	SDCR	32	H'E8034008	32
	System Mask Cfg Register	SMCR	32	H'E803400C	32
	Version Control Cfg Register	VCCR	32	H'E803401C	32
	Synchronous Base Address Cfg Register	SBCR	32	H'E8034020	32
	Asynchronous Base Address Cfg Register	ABCR	32	H'E8034024	32
	Control Base Address Cfg Register	CBCR	32	H'E8034028	32
	Isochronous Base Address Cfg Register	IBCR	32	H'E803402C	32
	Channel Interrupt Cfg Register	CICR	32	H'E8034030	32
	Channel 0 Entry Cfg Register	CECR0	32	H'E8034040	32
	Channel 0 Status Cfg Register	CSCR0	32	H'E8034044	32
	Channel 0 Current Buffer Cfg Register	CCBCR0	32	H'E8034048	32
	Channel 0 Next Buffer Cfg Register	CNBCR0	32	H'E803404C	32
	Channel 1 Entry Cfg Register	CECR1	32	H'E8034050	32
	Channel 1 Status Cfg Register	CSCR1	32	H'E8034054	32
	Channel 1 Current Buffer Cfg Register	CCBCR1	32	H'E8034058	32
	Channel 1 Next Buffer Cfg Register	CNBCR1	32	H'E803405C	32
	Channel 2 Entry Cfg Register	CECR2	32	H'E8034060	32
	Channel 2 Status Cfg Register	CSCR2	32	H'E8034064	32
	Channel 2 Current Buffer Cfg Register	CCBCR2	32	H'E8034068	32
	Channel 2 Next Buffer Cfg Register	CNBCR2	32	H'E803406C	32
	Channel 3 Entry Cfg Register	CECR3	32	H'E8034070	32
	Channel 3 Status Cfg Register	CSCR3	32	H'E8034074	32
	Channel 3 Current Buffer Cfg Register	CCBCR3	32	H'E8034078	32
	Channel 3 Next Buffer Cfg Register	CNBCR3	32	H'E803407C	32
	Channel 4 Entry Cfg Register	CECR4	32	H'E8034080	32
	Channel 4 Status Cfg Register	CSCR4	32	H'E8034084	32
	Channel 4 Current Buffer Cfg Register	CCBCR4	32	H'E8034088	32
	Channel 4 Next Buffer Cfg Register	CNBCR4	32	H'E803408C	32
Channel 5 Entry Cfg Register	CECR5	32	H'E8034090	32	
Channel 5 Status Cfg Register	CSCR5	32	H'E8034094	32	



Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Media local bus	Channel 5 Current Buffer Cfg Register	CCBCR5	32	H'E8034098	32
	Channel 5 Next Buffer Cfg Register	CNBCR5	32	H'E803409C	32
	Channel 6 Entry Cfg Register	CECR6	32	H'E80340A0	32
	Channel 6 Status Cfg Register	CSCR6	32	H'E80340A4	32
	Channel 6 Current Buffer Cfg Register	CCBCR6	32	H'E80340A8	32
	Channel 6 Next Buffer Cfg Register	CNBCR6	32	H'E80340AC	32
	Channel 7 Entry Cfg Register	CECR7	32	H'E80340B0	32
	Channel 7 Status Cfg Register	CSCR7	32	H'E80340B4	32
	Channel 7 Current Buffer Cfg Register	CCBCR7	32	H'E80340B8	32
	Channel 7 Next Buffer Cfg Register	CNBCR7	32	H'E80340BC	32
	Channel 8 Entry Cfg Register	CECR8	32	H'E80340C0	32
	Channel 8 Status Cfg Register	CSCR8	32	H'E80340C4	32
	Channel 8 Current Buffer Cfg Register	CCBCR8	32	H'E80340C8	32
	Channel 8 Next Buffer Cfg Register	CNBCR8	32	H'E80340CC	32
	Channel 9 Entry Cfg Register	CECR9	32	H'E80340D0	32
	Channel 9 Status Cfg Register	CSCR9	32	H'E80340D4	32
	Channel 9 Current Buffer Cfg Register	CCBCR9	32	H'E80340D8	32
	Channel 9 Next Buffer Cfg Register	CNBCR9	32	H'E80340DC	32
	Channel 10 Entry Cfg Register	CECR10	32	H'E80340E0	32
	Channel 10 Status Cfg Register	CSCR10	32	H'E80340E4	32
	Channel 10 Current Buffer Cfg Register	CCBCR10	32	H'E80340E8	32
	Channel 10 Next Buffer Cfg Register	CNBCR10	32	H'E80340EC	32
	Channel 11 Entry Cfg Register	CECR11	32	H'E80340F0	32
	Channel 11 Status Cfg Register	CSCR11	32	H'E80340F4	32
	Channel 11 Current Buffer Cfg Register	CCBCR11	32	H'E80340F8	32
	Channel 11 Next Buffer Cfg Register	CNBCR11	32	H'E80340FC	32
	Channel 12 Entry Cfg Register	CECR12	32	H'E8034100	32
	Channel 12 Status Cfg Register	CSCR12	32	H'E8034104	32
	Channel 12 Current Buffer Cfg Register	CCBCR12	32	H'E8034108	32
	Channel 12 Next Buffer Cfg Register	CNBCR12	32	H'E803410C	32
	Channel 13 Entry Cfg Register	CECR13	32	H'E8034110	32
	Channel 13 Status Cfg Register	CSCR13	32	H'E8034114	32
	Channel 13 Current Buffer Cfg Register	CCBCR13	32	H'E8034118	32
	Channel 13 Next Buffer Cfg Register	CNBCR13	32	H'E803411C	32
	Channel 14 Entry Cfg Register	CECR14	32	H'E8034120	32
	Channel 14 Status Cfg Register	CSCR14	32	H'E8034124	32
	Channel 14 Current Buffer Cfg Register	CCBCR14	32	H'E8034128	32
	Channel 14 Next Buffer Cfg Register	CNBCR14	32	H'E803412C	32
	Channel 15 Entry Cfg Register	CECR15	32	H'E8034130	32
	Channel 15 Status Cfg Register	CSCR15	32	H'E8034134	32
	Channel 15 Current Buffer Cfg Register	CCBCR15	32	H'E8034138	32
	Channel 15 Next Buffer Cfg Register	CNBCR15	32	H'E803413C	32
	Channel 16 Entry Cfg Register	CECR16	32	H'E8034140	32
	Channel 16 Status Cfg Register	CSCR16	32	H'E8034144	32
	Channel 16 Current Buffer Cfg Register	CCBCR16	32	H'E8034148	32
	Channel 16 Next Buffer Cfg Register	CNBCR16	32	H'E803414C	32
	Channel 17 Entry Cfg Register	CECR17	32	H'E8034150	32
	Channel 17 Status Cfg Register	CSCR17	32	H'E8034154	32
Channel 17 Current Buffer Cfg Register	CCBCR17	32	H'E8034158	32	
Channel 17 Next Buffer Cfg Register	CNBCR17	32	H'E803415C	32	
Channel 18 Entry Cfg Register	CECR18	32	H'E8034160	32	
Channel 18 Status Cfg Register	CSCR18	32	H'E8034164	32	

**Table 57.1 Register Addresses**

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Media local bus	Channel 18 Current Buffer Cfg Register	CCBCR18	32	H'E8034168	32
	Channel 18 Next Buffer Cfg Register	CNBCR18	32	H'E803416C	32
	Channel 19 Entry Cfg Register	CECR19	32	H'E8034170	32
	Channel 19 Status Cfg Register	CSCR19	32	H'E8034174	32
	Channel 19 Current Buffer Cfg Register	CCBCR19	32	H'E8034178	32
	Channel 19 Next Buffer Cfg Register	CNBCR19	32	H'E803417C	32
	Channel 20 Entry Cfg Register	CECR20	32	H'E8034180	32
	Channel 20 Status Cfg Register	CSCR20	32	H'E8034184	32
	Channel 20 Current Buffer Cfg Register	CCBCR20	32	H'E8034188	32
	Channel 20 Next Buffer Cfg Register	CNBCR20	32	H'E803418C	32
	Channel 21 Entry Cfg Register	CECR21	32	H'E8034190	32
	Channel 21 Status Cfg Register	CSCR21	32	H'E8034194	32
	Channel 21 Current Buffer Cfg Register	CCBCR21	32	H'E8034198	32
	Channel 21 Next Buffer Cfg Register	CNBCR21	32	H'E803419C	32
	Channel 22 Entry Cfg Register	CECR22	32	H'E80341A0	32
	Channel 22 Status Cfg Register	CSCR22	32	H'E80341A4	32
	Channel 22 Current Buffer Cfg Register	CCBCR22	32	H'E80341A8	32
	Channel 22 Next Buffer Cfg Register	CNBCR22	32	H'E80341AC	32
	Channel 23 Entry Cfg Register	CECR23	32	H'E80341B0	32
	Channel 23 Status Cfg Register	CSCR23	32	H'E80341B4	32
	Channel 23 Current Buffer Cfg Register	CCBCR23	32	H'E80341B8	32
	Channel 23 Next Buffer Cfg Register	CNBCR23	32	H'E80341BC	32
	Channel 24 Entry Cfg Register	CECR24	32	H'E80341C0	32
	Channel 24 Status Cfg Register	CSCR24	32	H'E80341C4	32
	Channel 24 Current Buffer Cfg Register	CCBCR24	32	H'E80341C8	32
	Channel 24 Next Buffer Cfg Register	CNBCR24	32	H'E80341CC	32
	Channel 25 Entry Cfg Register	CECR25	32	H'E80341D0	32
	Channel 25 Status Cfg Register	CSCR25	32	H'E80341D4	32
	Channel 25 Current Buffer Cfg Register	CCBCR25	32	H'E80341D8	32
	Channel 25 Next Buffer Cfg Register	CNBCR25	32	H'E80341DC	32
	Channel 26 Entry Cfg Register	CECR26	32	H'E80341E0	32
	Channel 26 Status Cfg Register	CSCR26	32	H'E80341E4	32
	Channel 26 Current Buffer Cfg Register	CCBCR26	32	H'E80341E8	32
	Channel 26 Next Buffer Cfg Register	CNBCR26	32	H'E80341EC	32
	Channel 27 Entry Cfg Register	CECR27	32	H'E80341F0	32
	Channel 27 Status Cfg Register	CSCR27	32	H'E80341F4	32
	Channel 27 Current Buffer Cfg Register	CCBCR27	32	H'E80341F8	32
	Channel 27 Next Buffer Cfg Register	CNBCR27	32	H'E80341FC	32
	Channel 28 Entry Cfg Register	CECR28	32	H'E8034200	32
	Channel 28 Status Cfg Register	CSCR28	32	H'E8034204	32
	Channel 28 Current Buffer Cfg Register	CCBCR28	32	H'E8034208	32
	Channel 28 Next Buffer Cfg Register	CNBCR28	32	H'E803420C	32
	Channel 29 Entry Cfg Register	CECR29	32	H'E8034210	32
	Channel 29 Status Cfg Register	CSCR29	32	H'E8034214	32
Channel 29 Current Buffer Cfg Register	CCBCR29	32	H'E8034218	32	
Channel 29 Next Buffer Cfg Register	CNBCR29	32	H'E803421C	32	
Channel 30 Entry Cfg Register	CECR30	32	H'E8034220	32	
Channel 30 Status Cfg Register	CSCR30	32	H'E8034224	32	
Channel 30 Current Buffer Cfg Register	CCBCR30	32	H'E8034228	32	
Channel 30 Next Buffer Cfg Register	CNBCR30	32	H'E803422C	32	
Local Channel 0 Buffer Cfg Register	LCBCR0	32	H'E8034280	32	
Local Channel 1 Buffer Cfg Register	LCBCR1	32	H'E8034284	32	

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Media local bus	Local Channel 2 Buffer Cfg Register	LCBCR2	32	H'E8034288	32
	Local Channel 3 Buffer Cfg Register	LCBCR3	32	H'E803428C	32
	Local Channel 4 Buffer Cfg Register	LCBCR4	32	H'E8034290	32
	Local Channel 5 Buffer Cfg Register	LCBCR5	32	H'E8034294	32
	Local Channel 6 Buffer Cfg Register	LCBCR6	32	H'E8034298	32
	Local Channel 7 Buffer Cfg Register	LCBCR7	32	H'E803429C	32
	Local Channel 8 Buffer Cfg Register	LCBCR8	32	H'E80342A0	32
	Local Channel 9 Buffer Cfg Register	LCBCR9	32	H'E80342A4	32
	Local Channel 10 Buffer Cfg Register	LCBCR10	32	H'E80342A8	32
	Local Channel 11 Buffer Cfg Register	LCBCR11	32	H'E80342AC	32
	Local Channel 12 Buffer Cfg Register	LCBCR12	32	H'E80342B0	32
	Local Channel 13 Buffer Cfg Register	LCBCR13	32	H'E80342B4	32
	Local Channel 14 Buffer Cfg Register	LCBCR14	32	H'E80342B8	32
	Local Channel 15 Buffer Cfg Register	LCBCR15	32	H'E80342BC	32
	Local Channel 16 Buffer Cfg Register	LCBCR16	32	H'E80342C0	32
	Local Channel 17 Buffer Cfg Register	LCBCR17	32	H'E80342C4	32
	Local Channel 18 Buffer Cfg Register	LCBCR18	32	H'E80342C8	32
	Local Channel 19 Buffer Cfg Register	LCBCR19	32	H'E80342CC	32
	Local Channel 20 Buffer Cfg Register	LCBCR20	32	H'E80342D0	32
	Local Channel 21 Buffer Cfg Register	LCBCR21	32	H'E80342D4	32
	Local Channel 22 Buffer Cfg Register	LCBCR22	32	H'E80342D8	32
	Local Channel 23 Buffer Cfg Register	LCBCR23	32	H'E80342DC	32
	Local Channel 24 Buffer Cfg Register	LCBCR24	32	H'E80342E0	32
	Local Channel 25 Buffer Cfg Register	LCBCR25	32	H'E80342E4	32
	Local Channel 26 Buffer Cfg Register	LCBCR26	32	H'E80342E8	32
	Local Channel 27 Buffer Cfg Register	LCBCR27	32	H'E80342EC	32
	Local Channel 28 Buffer Cfg Register	LCBCR28	32	H'E80342F0	32
	Local Channel 29 Buffer Cfg Register	LCBCR29	32	H'E80342F4	32
	Local Channel 30 Buffer Cfg Register	LCBCR30	32	H'E80342F8	32
	CAN interface	Channel m configuration register (m = 0 to 4)	RSCAN0CmCFG (m = 0 to 4)	32	H'E803A000 + m * H'0010
Channel m control register (m = 0 to 4)		RSCAN0CmCTR (m = 0 to 4)	32	H'E803A004 + m * H'0010	8, 16, 32
Channel m status register (m = 0 to 4)		RSCAN0CmSTS (m = 0 to 4)	32	H'E803A008 + m * H'0010	8, 16, 32
Channel m error flag register (m = 0 to 4)		RSCAN0CmERFL (m = 0 to 4)	32	H'E803A00C + m * H'0010	8, 16, 32
Global configuration register		RSCAN0GCFG	32	H'E803A084	8, 16, 32
Global control register		RSCAN0GCTR	32	H'E803A088	8, 16, 32
Global status register		RSCAN0GSTS	32	H'E803A08C	8, 16, 32
Global error flag register		RSCAN0GERFL	32	H'E803A090	8, 16, 32
Global timestamp counter register		RSCAN0GTSC	32	H'E803A094	16, 32
Receive rule entry control register		RSCAN0GAFLECTR	32	H'E803A098	8, 16, 32
Receive rule configuration register 0		RSCAN0GAFLCFG0	32	H'E803A09C	8, 16, 32
Receive rule configuration register 1		RSCAN0GAFLCFG1	32	H'E803A0A0	8, 16, 32
Receive buffer number register		RSCAN0RMNB	32	H'E803A0A4	8, 16, 32
Receive buffer new data register y (y = 0 to 2)		RSCAN0RMNDY (y = 0 to 2)	32	H'E803A0A8 + y * H'0004	8, 16, 32
Receive FIFO buffer configuration and control register x (x = 0 to 7)		RSCAN0RFCCx (x = 0 to 7)	32	H'E803A0B8 + x * H'0004	8, 16, 32
Receive FIFO buffer status register x (x = 0 to 7)		RSCAN0RFSTSx (x = 0 to 7)	32	H'E803A0D8 + x * H'0004	8, 16, 32
Receive FIFO buffer pointer control register x (x = 0 to 7)		RSCAN0RFPCTRx (x = 0 to 7)	32	H'E803A0F8 + x * H'0004	8, 16, 32
Transmit/receive FIFO buffer configuration and control register k (k = 0 to 14)		RSCAN0CFCCk (k = 0 to 14)	32	H'E803A118 + k * H'0004	8, 16, 32

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
CAN interface	Transmit/receive FIFO buffer status register k (k = 0 to 14)	RSCAN0CFSTSk (k = 0 to 14)	32	H'E803A178 + k * H'0004	8, 16, 32
	Transmit/receive FIFO buffer pointer control register k (k = 0 to 14)	RSCAN0CFPCTRk (k = 0 to 14)	32	H'E803A1D8 + k * H'0004	8, 16, 32
	FIFO empty status register	RSCAN0FESTS	32	H'E803A238	8, 16, 32
	FIFO full status register	RSCAN0FFSTS	32	H'E803A23C	8, 16, 32
	FIFO Msg lost status register	RSCAN0FMSTS	32	H'E803A240	8, 16, 32
	Receive FIFO buffer interrupt flag status register	RSCAN0RFISTS	32	H'E803A244	8, 16, 32
	Transmit/receive FIFO buffer RX interrupt flag status register	RSCAN0CFRISTS	32	H'E803A248	8, 16, 32
	Transmit/receive FIFO buffer TX interrupt flag status register	RSCAN0CFTISTS	32	H'E803A24C	8, 16, 32
	Transmit buffer control register p (p = 0 to 79)	RSCAN0TMCp (p = 0 to 79)	8	H'E803A250 + p * H'0001	8
	Transmit buffer status register p (p = 0 to 79)	RSCAN0TMSTSp (p = 0 to 79)	8	H'E803A2D0 + p * H'0001	8
	Transmit buffer transmit request status register y (y = 0 to 2)	RSCAN0TMRSTSy (y = 0 to 2)	32	H'E803A350 + y * H'0004	8, 16, 32
	Transmit buffer transmit abort request status register y (y = 0 to 2)	RSCAN0TMTARSTSy (y = 0 to 2)	32	H'E803A360 + y * H'0004	8, 16, 32
	Transmit buffer transmit complete status register y (y = 0 to 2)	RSCAN0TMTCASTSy (y = 0 to 2)	32	H'E803A370 + y * H'0004	8, 16, 32
	Transmit buffer transmit abort status register y (y = 0 to 2)	RSCAN0TMTASTSy (y = 0 to 2)	32	H'E803A380 + y * H'0004	8, 16, 32
	Transmit buffer interrupt enable configuration register y (y = 0 to 2)	RSCAN0TMIECy (y = 0 to 2)	32	H'E803A390 + y * H'0004	8, 16, 32
	Transmit queue configuration and control register m (m = 0 to 4)	RSCAN0TXQCCm (m = 0 to 4)	32	H'E803A3A0 + m * H'0010	8, 16, 32
	Transmit queue status register m (m = 0 to 4)	RSCAN0TXQSTSm (m = 0 to 4)	32	H'E803A3C0 + m * H'0004	8, 16, 32
	Transmit queue pointer control register m (m = 0 to 4)	RSCAN0TXQPCTRM (m = 0 to 4)	32	H'E803A3E0 + m * H'0004	8, 16, 32
	Transmit history configuration and control register m (m = 0 to 4)	RSCAN0THLCCm (m = 0 to 4)	32	H'E803A400 + m * H'0004	8, 16, 32
	Transmit history status register m (m = 0 to 4)	RSCAN0THLSTSm (m = 0 to 4)	32	H'E803A420 + m * H'0004	8, 16, 32
	Transmit history pointer control register m (m = 0 to 4)	RSCAN0THLPCTRM (m = 0 to 4)	32	H'E803A440 + m * H'0004	8, 16, 32
	Global TX interrupt status register 0	RSCAN0GTINTSTS0	32	H'E803A460	8, 16, 32
	Global TX interrupt status register 1	RSCAN0GTINTSTS1	32	H'E803A464	8, 16, 32
	Global test configuration register	RSCAN0GTSTCFG	32	H'E803A468	8, 16, 32
	Global test control register	RSCAN0GTSTCTR	32	H'E803A46C	8, 16, 32
	Global lock key register	RSCAN0GLOCKK	32	H'E803A47C	16, 32
	Receive rule ID register j (j = 0 to 15)	RSCAN0GAFLIDj (j = 0 to 15)	32	H'E803A500 + j * H'0010	8, 16, 32
	Receive rule mask register j (j = 0 to 15)	RSCAN0GAFLMj (j = 0 to 15)	32	H'E803A504 + j * H'0010	8, 16, 32
	Receive rule pointer 0 register j (j = 0 to 15)	RSCAN0GAFLP0j (j = 0 to 15)	32	H'E803A508 + j * H'0010	8, 16, 32
	Receive rule pointer 1 register j (j = 0 to 15)	RSCAN0GAFLP1j (j = 0 to 15)	32	H'E803A50C + j * H'0010	8, 16, 32
	Receive buffer ID register q (q = 0 to 79)	RSCAN0RMIDq (q = 0 to 79)	32	H'E803A600 + q * H'0010	8, 16, 32
	Receive buffer pointer register q (q = 0 to 79)	RSCAN0RMPTRq (q = 0 to 79)	32	H'E803A604 + q * H'0010	8, 16, 32
	Receive buffer data field 0 register q (q = 0 to 79)	RSCAN0RMDf0q (q = 0 to 79)	32	H'E803A608 + q * H'0010	8, 16, 32
	Receive buffer data field 1 register q (q = 0 to 79)	RSCAN0RMDf1q (q = 0 to 79)	32	H'E803A60C + q * H'0010	8, 16, 32
Receive FIFO buffer access ID register x (x = 0 to 7)	RSCAN0RFIDx (x = 0 to 7)	32	H'E803AE00 + x * H'0010	8, 16, 32	
Receive FIFO buffer access pointer register x (x = 0 to 7)	RSCAN0RFPTRx (x = 0 to 7)	32	H'E803AE04 + x * H'0010	8, 16, 32	
Receive FIFO buffer access data field 0 register x (x = 0 to 7)	RSCAN0RFDf0x (x = 0 to 7)	32	H'E803AE08 + x * H'0010	8, 16, 32	

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
CAN interface	Receive FIFO buffer access data field 1 register x (x = 0 to 7)	RSCAN0RFDF1x (x = 0 to 7)	32	H'E803AE0C + x * H'0010	8, 16, 32
	Transmit/receive FIFO buffer access ID register k (k = 0 to 14)	RSCAN0CFIDk (k = 0 to 14)	32	H'E803AE80 + k * H'0010	8, 16, 32
	Transmit/receive FIFO buffer access pointer register k (k = 0 to 14)	RSCAN0CFPTRk (k = 0 to 14)	32	H'E803AE84 + k * H'0010	8, 16, 32
	Transmit/receive FIFO buffer access data field 0 register k (k = 0 to 14)	RSCAN0CFDF0k (k = 0 to 14)	32	H'E803AE88 + k * H'0010	8, 16, 32
	Transmit/receive FIFO buffer access data field 1 register k (k = 0 to 14)	RSCAN0CFDF1k (k = 0 to 14)	32	H'E803AE8C + k * H'0010	8, 16, 32
	Transmit buffer ID register p (p = 0 to 79)	RSCAN0TMIDp (p = 0 to 79)	32	H'E803B000 + p * H'0010	8, 16, 32
	Transmit buffer pointer register p (p = 0 to 79)	RSCAN0TMPTRp (p = 0 to 79)	32	H'E803B004 + p * H'0010	8, 16, 32
	Transmit buffer data field 0 register p (p = 0 to 79)	RSCAN0TMDf0p (p = 0 to 79)	32	H'E803B008 + p * H'0010	8, 16, 32
	Transmit buffer data field 1 register p (p = 0 to 79)	RSCAN0TMDf1p (p = 0 to 79)	32	H'E803B00C + p * H'0010	8, 16, 32
	Transmit history access register m (m = 0 to 4)	RSCAN0THLACCm (m = 0 to 4)	32	H'E803B800 + m * H'0004	8, 16, 32
IEBus controller	IEBB0 bus control register	IEBB0BCR	8	H'FCFEF000	8
	IEBB0 power save register	IEBB0PSR	8	H'FCFEF004	8
	IEBB0 unit address register	IEBB0UAR	16	H'FCFEF008	16
	IEBB0 slave address register	IEBB0SAR	16	H'FCFEF00C	16
	IEBB0 partner address register	IEBB0PAR	16	H'FCFEF010	16
	IEBB0 reception slave address register	IEBB0RSA	16	H'FCFEF014	16
	IEBB0 control data register	IEBB0CDR	8	H'FCFEF018	8
	IEBB0 transmission control data register	IEBB0TCD	8	H'FCFEF01C	8
	IEBB0 reception control data register	IEBB0RCD	8	H'FCFEF020	8
	IEBB0 message length register	IEBB0DLR	8	H'FCFEF024	8
	IEBB0 transmission message length register	IEBB0TDL	8	H'FCFEF028	8
	IEBB0 reception message length register	IEBB0RDL	8	H'FCFEF02C	8
	IEBB0 clock selection register	IEBB0CKS	8	H'FCFEF030	8
	IEBB0 transfer mode setting register	IEBB0TMS	8	H'FCFEF034	8
	IEBB0 pointer clear register	IEBB0PCR	8	H'FCFEF038	8
	IEBB0 buffer status register	IEBB0BSR	16	H'FCFEF03C	16
	IEBB0 slave status register	IEBB0SSR	8	H'FCFEF040	8
	IEBB0 unit status register	IEBB0USR	8	H'FCFEF044	8
	IEBB0 interrupt status register	IEBB0ISR	8	H'FCFEF048	8
	IEBB0 error status register	IEBB0ESR	8	H'FCFEF04C	8
	IEBB0 field status register	IEBB0FSR	8	H'FCFEF050	8
	IEBB0 success count register	IEBB0SCR	8	H'FCFEF054	8
	IEBB0 communication count register	IEBB0CCR	8	H'FCFEF058	8
	IEBB0 status clear register 0	IEBB0STC0	8	H'FCFEF05C	8
IEBB0 status clear register 1	IEBB0STC1	8	H'FCFEF060	8	
IEBB0 data register	IEBB0DR	8	H'FCFEF064	8	

**Table 57.1 Register Addresses**

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Renesas SPDIF interface	Transmitter channel 1 audio register	TLCA	32	H'E8012000	32
	Transmitter channel 2 audio register	TRCA	32	H'E8012004	32
	Transmitter channel 1 status register	TLCS	32	H'E8012008	32
	Transmitter channel 2 status register	TRCS	32	H'E801200C	32
	Transmitter user data register	TUI	32	H'E8012010	32
	Receiver channel 1 audio register	RLCA	32	H'E8012014	32
	Receiver channel 2 audio register	RRCA	32	H'E8012018	32
	Receiver channel 1 status register	RLCS	32	H'E801201C	32
	Receiver channel 2 status register	RRCS	32	H'E8012020	32
	Receiver user data register	RUI	32	H'E8012024	32
	Control register	CTRL	32	H'E8012028	32
	Status register	STAT	32	H'E801202C	32
	Transmitter DMA audio data register	TDAD	32	H'E8012030	32
	Receiver DMA audio data register	RDAD	32	H'E8012034	32
CD-ROM decoder	Enable control register	CROMEN	8	H'E8005000	8
	Sync code-based synchronization control register	CROMSY0	8	H'E8005001	8
	Decoding mode control register	CROMCTL0	8	H'E8005002	8
	EDC/ECC check control register	CROMCTL1	8	H'E8005003	8
	Automatic decoding stop control register	CROMCTL3	8	H'E8005005	8
	Decoding option setting control register	CROMCTL4	8	H'E8005006	8
	HEAD20 to HEAD22 representation control register	CROMCTL5	8	H'E8005007	8
	Sync code status register	CROMST0	8	H'E8005008	8
	Post-ECC header error status register	CROMST1	8	H'E8005009	8
	Post-ECC subheader error status register	CROMST3	8	H'E800500B	8
	Header/subheader validity check status register	CROMST4	8	H'E800500C	8
	Mode determination and link sector detection status register	CROMST5	8	H'E800500D	8
	ECC/EDC error status register	CROMST6	8	H'E800500E	8
	Buffer status register	CBUFST0	8	H'E8005014	8
	Decoding stoppage source status register	CBUFST1	8	H'E8005015	8
	Buffer overflow status register	CBUFST2	8	H'E8005016	8
	Pre-ECC correction header: minutes data register	HEAD00	8	H'E8005018	8
	Pre-ECC correction header: seconds data register	HEAD01	8	H'E8005019	8
	Pre-ECC correction header: frames (1/75 second) data register	HEAD02	8	H'E800501A	8
	Pre-ECC correction header: mode data register	HEAD03	8	H'E800501B	8
	Pre-ECC correction subheader: file number (byte 16) data register	SHEAD00	8	H'E800501C	8
	Pre-ECC correction subheader: channel number (byte 17) data register	SHEAD01	8	H'E800501D	8
	Pre-ECC correction subheader: sub-mode (byte 18) data register	SHEAD02	8	H'E800501E	8
Pre-ECC correction subheader: data type (byte 19) data register	SHEAD03	8	H'E800501F	8	
Pre-ECC correction subheader: file number (byte 20) data register	SHEAD04	8	H'E8005020	8	
Pre-ECC correction subheader: channel number (byte 21) data register	SHEAD05	8	H'E8005021	8	

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
CD-ROM decoder	Pre-ECC correction subheader: sub-mode (byte 22) data register	SHEAD06	8	H'E8005022	8
	Pre-ECC correction subheader: data type (byte 23) data register	SHEAD07	8	H'E8005023	8
	Post-ECC correction header: minutes data register	HEAD20	8	H'E8005024	8
	Post-ECC correction header: seconds data register	HEAD21	8	H'E8005025	8
	Post-ECC correction header: frames (1/75 second) data register	HEAD22	8	H'E8005026	8
	Post-ECC correction header: mode data register	HEAD23	8	H'E8005027	8
	Post-ECC correction subheader: file number (byte 16) data register	SHEAD20	8	H'E8005028	8
	Post-ECC correction subheader: channel number (byte 17) data register	SHEAD21	8	H'E8005029	8
	Post-ECC correction subheader: sub-mode (byte 18) data register	SHEAD22	8	H'E800502A	8
	Post-ECC correction subheader: data type (byte 19) data register	SHEAD23	8	H'E800502B	8
	Post-ECC correction subheader: file number (byte 20) data register	SHEAD24	8	H'E800502C	8
	Post-ECC correction subheader: channel number (byte 21) data register	SHEAD25	8	H'E800502D	8
	Post-ECC correction subheader: sub-mode (byte 22) data register	SHEAD26	8	H'E800502E	8
	Post-ECC correction subheader: data type (byte 23) data register	SHEAD27	8	H'E800502F	8
	Automatic buffering setting control register	CBUFCTL0	8	H'E8005040	8
	Automatic buffering start sector setting: minutes control register	CBUFCTL1	8	H'E8005041	8
	Automatic buffering start sector setting: seconds control register	CBUFCTL2	8	H'E8005042	8
	Automatic buffering start sector setting: frames control register	CBUFCTL3	8	H'E8005043	8
	ISY interrupt source mask control register	CROMST0M	8	H'E8005045	8
	CD-ROM decoder reset control register	ROMDECRST	8	H'E8005100	8
	CD-ROM decoder reset status register	RSTSTAT	8	H'E8005101	8
	Serial sound interface data control register	SSI	8	H'E8005102	8
	Interrupt flag register	INTHOLD	8	H'E8005108	8
	Interrupt source mask control register	INHINT	8	H'E8005109	8
	CD-ROM decoder stream data input register	STRMDIN0	16	H'E8005200	16 (R/W), 32 (W)
	CD-ROM decoder stream data input register	STRMDIN2	16	H'E8005202	16
	CD-ROM decoder stream data output register	STRMDOUT0	16	H'E8005204	16
	LIN interface channel 0	LIN wake-up baud rate selector register	RLN30LWBR	8	H'FCFE9001
LIN baud rate prescaler 0 register		RLN30LBRP0	8	H'FCFE9002	8
LIN baud rate prescaler 1 register		RLN30LBRP1	8	H'FCFE9003	8
LIN self-test control register		RLN30LSTC	8	H'FCFE9004	8
LIN mode register		RLN30LMD	8	H'FCFE9008	8
LIN break field configuration register		RLN30LBFC	8	H'FCFE9009	8
LIN space configuration register		RLN30LSC	8	H'FCFE900A	8
LIN wake-up configuration register		RLN30LWUP	8	H'FCFE900B	8
LIN interrupt enable register		RLN30LIE	8	H'FCFE900C	8
LIN error detection enable register		RLN30LEDE	8	H'FCFE900D	8
LIN control register		RLN30LCUC	8	H'FCFE900E	8
LIN transmission control register		RLN30LTRC	8	H'FCFE9010	8
LIN mode status register		RLN30LMST	8	H'FCFE9011	8
LIN status register		RLN30LST	8	H'FCFE9012	8
LIN error status register		RLN30LEST	8	H'FCFE9013	8
LIN data field configuration register		RLN30LDFC	8	H'FCFE9014	8
LIN ID buffer register		RLN30LIDB	8	H'FCFE9015	8

**Table 57.1 Register Addresses**

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
LIN interface channel 0	LIN checksum buffer register	RLN30LCBR	8	H'FCFE9016	8
	LIN data buffer 1 register	RLN30LDB1	8	H'FCFE9018	8
	LIN data buffer 2 register	RLN30LDB2	8	H'FCFE9019	8
	LIN data buffer 3 register	RLN30LDB3	8	H'FCFE901A	8
	LIN data buffer 4 register	RLN30LDB4	8	H'FCFE901B	8
	LIN data buffer 5 register	RLN30LDB5	8	H'FCFE901C	8
	LIN data buffer 6 register	RLN30LDB6	8	H'FCFE901D	8
	LIN data buffer 7 register	RLN30LDB7	8	H'FCFE901E	8
	LIN data buffer 8 register	RLN30LDB8	8	H'FCFE901F	8
LIN interface channel 1	LIN wake-up baud rate selector register	RLN31LWBR	8	H'FCFE9801	8
	LIN baud rate prescaler 0 register	RLN31LBRP0	8	H'FCFE9802	8
	LIN baud rate prescaler 1 register	RLN31LBRP1	8	H'FCFE9803	8
	LIN self-test control register	RLN31LSTC	8	H'FCFE9804	8
	LIN mode register	RLN31LMD	8	H'FCFE9808	8
	LIN break field configuration register	RLN31LBFC	8	H'FCFE9809	8
	LIN space configuration register	RLN31LSC	8	H'FCFE980A	8
	LIN wake-up configuration register	RLN31LWUP	8	H'FCFE980B	8
	LIN interrupt enable register	RLN31LIE	8	H'FCFE980C	8
	LIN error detection enable register	RLN31LEDE	8	H'FCFE980D	8
	LIN control register	RLN31LCUC	8	H'FCFE980E	8
	LIN transmission control register	RLN31LTRC	8	H'FCFE9810	8
	LIN mode status register	RLN31LMST	8	H'FCFE9811	8
	LIN status register	RLN31LST	8	H'FCFE9812	8
	LIN error status register	RLN31LEST	8	H'FCFE9813	8
	LIN data field configuration register	RLN31LDFC	8	H'FCFE9814	8
	LIN ID buffer register	RLN31LIDB	8	H'FCFE9815	8
	LIN checksum buffer register	RLN31LCBR	8	H'FCFE9816	8
	LIN data buffer 1 register	RLN31LDB1	8	H'FCFE9818	8
	LIN data buffer 2 register	RLN31LDB2	8	H'FCFE9819	8
	LIN data buffer 3 register	RLN31LDB3	8	H'FCFE981A	8
	LIN data buffer 4 register	RLN31LDB4	8	H'FCFE981B	8
	LIN data buffer 5 register	RLN31LDB5	8	H'FCFE981C	8
	LIN data buffer 6 register	RLN31LDB6	8	H'FCFE981D	8
	LIN data buffer 7 register	RLN31LDB7	8	H'FCFE981E	8
	LIN data buffer 8 register	RLN31LDB8	8	H'FCFE981F	8



Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Ethernet controller	Software reset register	ARSTR	32	H'E8204800	32
	E-MAC mode register	ECMR0	32	H'E8203500	32
	E-MAC status register	ECSR0	32	H'E8203510	32
	E-MAC interrupt permission register	ECSIPR0	32	H'E8203518	32
	PHY interface register	PIR0	32	H'E8203520	32
	MAC address high register	MAHR0	32	H'E82035C0	32
	MAC address low register	MALR0	32	H'E82035C8	32
	Receive frame length register	RFLR0	32	H'E8203508	32
	CRC error frame receive counter register	CEFCR0	32	H'E8203740	32
	Frame receive error counter register	FRECR0	32	H'E8203748	32
	Too-short frame receive counter register	TSFRCR0	32	H'E8203750	32
	Too-long frame receive counter register	TLFRCR0	32	H'E8203758	32
	Residual-bit frame receive counter register	RFCR0	32	H'E8203760	32
	Multicast address frame receive counter register	MAFCR0	32	H'E8203778	32
	Automatic PAUSE frame register	APR0	32	H'E8203554	32
	Manual PAUSE frame register	MPR0	32	H'E8203558	32
	Automatic PAUSE frame retransmit count register	TPAUSER0	32	H'E8203564	32
	PAUSE frame transmit counter register	PFTCR0	32	H'E820355C	32
	PAUSE frame receive counter register	PFCR0	32	H'E8203560	32
	TSU counter reset register	TSU_CTRST	32	H'E8204804	32
	VLANtag set register	TSU_VTAG0	32	H'E8204858	32
	CAM entry table busy register	TSU_ADSBSY	32	H'E8204860	32
	CAM entry table enable register	TSU_TEN	32	H'E8204864	32
	CAM entry table 0H register	TSU_ADRH0	32	H'E8204900	32
	CAM entry table 1H register	TSU_ADRH1	32	H'E8204908	32
	CAM entry table 2H register	TSU_ADRH2	32	H'E8204910	32
	CAM entry table 3H register	TSU_ADRH3	32	H'E8204918	32
	CAM entry table 4H register	TSU_ADRH4	32	H'E8204920	32
	CAM entry table 5H register	TSU_ADRH5	32	H'E8204928	32
	CAM entry table 6H register	TSU_ADRH6	32	H'E8204930	32
	CAM entry table 7H register	TSU_ADRH7	32	H'E8204938	32
	CAM entry table 8H register	TSU_ADRH8	32	H'E8204940	32
	CAM entry table 9H register	TSU_ADRH9	32	H'E8204948	32
	CAM entry table 10H register	TSU_ADRH10	32	H'E8204950	32
	CAM entry table 11H register	TSU_ADRH11	32	H'E8204958	32
	CAM entry table 12H register	TSU_ADRH12	32	H'E8204960	32
	CAM entry table 13H register	TSU_ADRH13	32	H'E8204968	32
	CAM entry table 14H register	TSU_ADRH14	32	H'E8204970	32
	CAM entry table 15H register	TSU_ADRH15	32	H'E8204978	32
	CAM entry table 16H register	TSU_ADRH16	32	H'E8204980	32
	CAM entry table 17H register	TSU_ADRH17	32	H'E8204988	32
	CAM entry table 18H register	TSU_ADRH18	32	H'E8204990	32
	CAM entry table 19H register	TSU_ADRH19	32	H'E8204998	32
	CAM entry table 20H register	TSU_ADRH20	32	H'E82049A0	32
	CAM entry table 21H register	TSU_ADRH21	32	H'E82049A8	32
	CAM entry table 22H register	TSU_ADRH22	32	H'E82049B0	32
	CAM entry table 23H register	TSU_ADRH23	32	H'E82049B8	32
	CAM entry table 24H register	TSU_ADRH24	32	H'E82049C0	32
CAM entry table 25H register	TSU_ADRH25	32	H'E82049C8	32	
CAM entry table 26H register	TSU_ADRH26	32	H'E82049D0	32	
CAM entry table 27H register	TSU_ADRH27	32	H'E82049D8	32	
CAM entry table 28H register	TSU_ADRH28	32	H'E82049E0	32	

**Table 57.1 Register Addresses**

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Ethernet controller	CAM entry table 29H register	TSU_ADRH29	32	H'E82049E8	32
	CAM entry table 30H register	TSU_ADRH30	32	H'E82049F0	32
	CAM entry table 31H register	TSU_ADRH31	32	H'E82049F8	32
	CAM entry table 0L register	TSU_ADRL0	32	H'E8204904	32
	CAM entry table 1L register	TSU_ADRL1	32	H'E820490C	32
	CAM entry table 2L register	TSU_ADRL2	32	H'E8204914	32
	CAM entry table 3L register	TSU_ADRL3	32	H'E820491C	32
	CAM entry table 4L register	TSU_ADRL4	32	H'E8204924	32
	CAM entry table 5L register	TSU_ADRL5	32	H'E820492C	32
	CAM entry table 6L register	TSU_ADRL6	32	H'E8204934	32
	CAM entry table 7L register	TSU_ADRL7	32	H'E820493C	32
	CAM entry table 8L register	TSU_ADRL8	32	H'E8204944	32
	CAM entry table 9L register	TSU_ADRL9	32	H'E820494C	32
	CAM entry table 10L register	TSU_ADRL10	32	H'E8204954	32
	CAM entry table 11L register	TSU_ADRL11	32	H'E820495C	32
	CAM entry table 12L register	TSU_ADRL12	32	H'E8204964	32
	CAM entry table 13L register	TSU_ADRL13	32	H'E820496C	32
	CAM entry table 14L register	TSU_ADRL14	32	H'E8204974	32
	CAM entry table 15L register	TSU_ADRL15	32	H'E820497C	32
	CAM entry table 16L register	TSU_ADRL16	32	H'E8204984	32
	CAM entry table 17L register	TSU_ADRL17	32	H'E820498C	32
	CAM entry table 18L register	TSU_ADRL18	32	H'E8204994	32
	CAM entry table 19L register	TSU_ADRL19	32	H'E820499C	32
	CAM entry table 20L register	TSU_ADRL20	32	H'E82049A4	32
	CAM entry table 21L register	TSU_ADRL21	32	H'E82049AC	32
	CAM entry table 22L register	TSU_ADRL22	32	H'E82049B4	32
	CAM entry table 23L register	TSU_ADRL23	32	H'E82049BC	32
	CAM entry table 24L register	TSU_ADRL24	32	H'E82049C4	32
	CAM entry table 25L register	TSU_ADRL25	32	H'E82049CC	32
	CAM entry table 26L register	TSU_ADRL26	32	H'E82049D4	32
	CAM entry table 27L register	TSU_ADRL27	32	H'E82049DC	32
	CAM entry table 28L register	TSU_ADRL28	32	H'E82049E4	32
	CAM entry table 29L register	TSU_ADRL29	32	H'E82049EC	32
	CAM entry table 30L register	TSU_ADRL30	32	H'E82049F4	32
	CAM entry table 31L register	TSU_ADRL31	32	H'E82049FC	32
	Transmit frame counter register	TXNLCR0	32	H'E8204880	32
	Transmit frame counter register	TXALCR0	32	H'E8204884	32
	Receive frame counter register	RXNLCR0	32	H'E8204888	32
	Receive frame counter register	RXALCR0	32	H'E820488C	32
	E-DMAC start register	EDSR0	32	H'E8203000	32
	E-DMAC mode register	EDMR0	32	H'E8203400	32
	E-DMAC transmit request register	EDTRR0	32	H'E8203408	32
	E-DMAC receive request register	EDRRR0	32	H'E8203410	32
	E-MAC/E-DMAC status register	EESR0	32	H'E8203428	32
E-MAC/E-DMAC status interrupt permission register	EESIPR0	32	H'E8203430	32	
Transmit descriptor list start address register	TDLAR0	32	H'E8203010	32	
Transmit descriptor fetch address register	TDFAR0	32	H'E8203014	32	
Transmit descriptor finished address register	TDFXR0	32	H'E8203018	32	
Transmit descriptor final flag register	TDFFR0	32	H'E820301C	32	
Receive descriptor list start address register	RDLAR0	32	H'E8203030	32	
Receive descriptor fetch address register	RDFAR0	32	H'E8203034	32	
Receive descriptor finished address register	RDFXR0	32	H'E8203038	32	

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Ethernet controller	Receive descriptor final flag register	RDFFR0	32	H'E820303C	32
	Transmit/receive status copy enable register	TRSCER0	32	H'E8203438	32
	Receive missed-frame counter register	RMFCR0	32	H'E8203440	32
	Transmit FIFO threshold register	TFTR0	32	H'E8203448	32
	FIFO depth register	FDR0	32	H'E8203450	32
	Receiving method control register	RMCR0	32	H'E8203458	32
	Receive data padding insert register	RPADIR0	32	H'E8203460	32
	Overflow alert FIFO threshold register	FCFTR0	32	H'E8203468	32
	Intelligent checksum mode register	CMSR	32	H'E82034E4	32
	Intelligent checksum skipped bytes monitor register	CSSBM	32	H'E82034E8	32
	Intelligent checksum monitor register	CSSMR	32	H'E82034EC	32
A/D converter	A/D data register A	ADDRA	16	H'E8005800	16
	A/D data register B	ADDRB	16	H'E8005802	16
	A/D data register C	ADDRC	16	H'E8005804	16
	A/D data register D	ADDRD	16	H'E8005806	16
	A/D data register E	ADDRE	16	H'E8005808	16
	A/D data register F	ADDRF	16	H'E800580A	16
	A/D data register G	ADDRG	16	H'E800580C	16
	A/D data register H	ADDRH	16	H'E800580E	16
	A/D comparison upper limit value register A	ADCMPHA	16	H'E8005820	16
	A/D comparison lower limit value register A	ADCMPLA	16	H'E8005822	16
	A/D comparison upper limit value register B	ADCMPHB	16	H'E8005824	16
	A/D comparison lower limit value register B	ADCMPLB	16	H'E8005826	16
	A/D comparison upper limit value register C	ADCMPHC	16	H'E8005828	16
	A/D comparison lower limit value register C	ADCMPLC	16	H'E800582A	16
	A/D comparison upper limit value register D	ADCMPHD	16	H'E800582C	16
	A/D comparison lower limit value register D	ADCMPLD	16	H'E800582E	16
	A/D comparison upper limit value register E	ADCMPE	16	H'E8005830	16
	A/D comparison lower limit value register E	ADCMPE	16	H'E8005832	16
	A/D comparison upper limit value register F	ADCMPHF	16	H'E8005834	16
	A/D comparison lower limit value register F	ADCMPLF	16	H'E8005836	16
	A/D comparison upper limit value register G	ADCMPHG	16	H'E8005838	16
	A/D comparison lower limit value register G	ADCMPLG	16	H'E800583A	16
	A/D comparison upper limit value register H	ADCMPHH	16	H'E800583C	16
	A/D comparison lower limit value register H	ADCMPLH	16	H'E800583E	16
A/D control/status register	ADCSR	16	H'E8005860	16	
A/D comparison interrupt enable register	ADCMPE	16	H'E8005862	16	
A/D comparison status register	ADCMPSR	16	H'E8005864	16	
NAND flash memory controller	Common control register	FLCMNCR	32	H'FCFF4000	32
	Command control register	FLCMDCR	32	H'FCFF4004	32
	Command code register	FLCMCDR	32	H'FCFF4008	32
	Address register	FLADR	32	H'FCFF400C	32
	Address register 2	FLADR2	32	H'FCFF403C	32
	Data register	FLDATAR	32	H'FCFF4010	32
	Data counter register	FLDTCNTR	32	H'FCFF4014	32
	Interrupt DMA control register	FLINTDMACR	32	H'FCFF4018	32
	Ready busy timeout setting register	FLBSYTMR	32	H'FCFF401C	32
	Ready busy timeout counter	FLBSYCNT	32	H'FCFF4020	32
	Data FIFO register	FLDTFIFO	32	H'FCFF4050	32
	Transfer control register	FLTRCR	8	H'FCFF402C	8

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
USB2.0 host/function module	System configuration control register_0	SYSCFG0_0	16	H'E8010000	16
	CPU bus wait setting register_0	BUSWAIT_0	16	H'E8010002	16
	System configuration status register_0	SYSSTS0_0	16	H'E8010004	16
	Device state control register_0_0	DVSTCTR0_0	16	H'E8010008	16
	Test mode register_0	TESTMODE_0	16	H'E801000C	16
	DMA0-FIFO bus configuration register_0	D0FBCFG_0	16	H'E8010010	16
	DMA1-FIFO bus configuration register_0	D1FBCFG_0	16	H'E8010012	16
	CFIFO port register_0	CFIFO_0	32	H'E8010014	8, 16, 32
	D0FIFO port register_0	D0FIFO_0	32	H'E8010018	8, 16, 32
	D1FIFO port register_0	D1FIFO_0	32	H'E801001C	8, 16, 32
	CFIFO port select register_0	CFIFOSEL_0	16	H'E8010020	16
	CFIFO port control register_0	CFIFOCTR_0	16	H'E8010022	16
	D0FIFO port select register_0	D0FIFOSEL_0	16	H'E8010028	16
	D0FIFO port control register_0	D0FIFOCTR_0	16	H'E801002A	16
	D1FIFO port select register_0	D1FIFOSEL_0	16	H'E801002C	16
	D1FIFO port control register_0	D1FIFOCTR_0	16	H'E801002E	16
	Interrupt enable register_0_0	INTENB0_0	16	H'E8010030	16
	Interrupt enable register_1_0	INTENB1_0	16	H'E8010032	16
	BRDY interrupt enable register_0	BRDYENB_0	16	H'E8010036	16
	NRDY interrupt enable register_0	NRDYENB_0	16	H'E8010038	16
	BEMP interrupt enable register_0	BEMPENB_0	16	H'E801003A	16
	SOF output configuration register_0	SOFCFG_0	16	H'E801003C	16
	Interrupt status register_0_0	INTSTS0_0	16	H'E8010040	16
	Interrupt status register_1_0	INTSTS1_0	16	H'E8010042	16
	BRDY interrupt status register_0	BRDYSTS_0	16	H'E8010046	16
	NRDY interrupt status register_0	NRDYSTS_0	16	H'E8010048	16
	BEMP interrupt status register_0	BEMPSTS_0	16	H'E801004A	16
	Frame number register_0	FRMNUM_0	16	H'E801004C	16
	μFrame number register_0	UFRMNUM_0	16	H'E801004E	16
	USB address register_0	USBADDR_0	16	H'E8010050	16
	USB request type register_0	USBREQ_0	16	H'E8010054	16
	USB request value register_0	USBVAL_0	16	H'E8010056	16
	USB request index register_0	USBINDX_0	16	H'E8010058	16
	USB request length register_0	USBLENG_0	16	H'E801005A	16
	DCP configuration register_0	DCPCFG_0	16	H'E801005C	16
	DCP maximum packet size register_0	DCPMAXP_0	16	H'E801005E	16
	DCP control register_0	DCPCTR_0	16	H'E8010060	16
	Pipe window select register_0	PIPESEL_0	16	H'E8010064	16
	Pipe configuration register_0	PIPECFG_0	16	H'E8010068	16
	Pipe buffer setting register_0	PIPEBUF_0	16	H'E801006A	16
	Pipe maximum packet size register_0	PEMAXP_0	16	H'E801006C	16
	Pipe timing control register_0	PIPEPERI_0	16	H'E801006E	16
	Pipe 1 control register_0	PIPE1CTR_0	16	H'E8010070	16
	Pipe 2 control register_0	PIPE2CTR_0	16	H'E8010072	16
Pipe 3 control register_0	PIPE3CTR_0	16	H'E8010074	16	
Pipe 4 control register_0	PIPE4CTR_0	16	H'E8010076	16	
Pipe 5 control register_0	PIPE5CTR_0	16	H'E8010078	16	
Pipe 6 control register_0	PIPE6CTR_0	16	H'E801007A	16	
Pipe 7 control register_0	PIPE7CTR_0	16	H'E801007C	16	
Pipe 8 control register_0	PIPE8CTR_0	16	H'E801007E	16	
Pipe 9 control register_0	PIPE9CTR_0	16	H'E8010080	16	
Pipe A control register_0	PIPEACTR_0	16	H'E8010082	16	

**Table 57.1 Register Addresses**

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
USB2.0 host/function module	Pipe B control register_0	PIPEBCTR_0	16	H'E8010084	16
	Pipe C control register_0	PIPECCTR_0	16	H'E8010086	16
	Pipe D control register_0	PIPEDCTR_0	16	H'E8010088	16
	Pipe E control register_0	PIPEECTR_0	16	H'E801008A	16
	Pipe F control register_0	PIPEFCTR_0	16	H'E801008C	16
	Pipe 1 transaction counter enable register_0	PIPE1TRE_0	16	H'E8010090	16
	Pipe 1 transaction counter register_0	PIPE1TRN_0	16	H'E8010092	16
	Pipe 2 transaction counter enable register_0	PIPE2TRE_0	16	H'E8010094	16
	Pipe 2 transaction counter register_0	PIPE2TRN_0	16	H'E8010096	16
	Pipe 3 transaction counter enable register_0	PIPE3TRE_0	16	H'E8010098	16
	Pipe 3 transaction counter register_0	PIPE3TRN_0	16	H'E801009A	16
	Pipe 4 transaction counter enable register_0	PIPE4TRE_0	16	H'E801009C	16
	Pipe 4 transaction counter register_0	PIPE4TRN_0	16	H'E801009E	16
	Pipe 5 transaction counter enable register_0	PIPE5TRE_0	16	H'E80100A0	16
	Pipe 5 transaction counter register_0	PIPE5TRN_0	16	H'E80100A2	16
	Pipe B transaction counter enable register_0	PIPEBTRE_0	16	H'E80100A4	16
	Pipe B transaction counter register_0	PIPEBTRN_0	16	H'E80100A6	16
	Pipe C transaction counter enable register_0	PIPECTRE_0	16	H'E80100A8	16
	Pipe C transaction counter register_0	PIPECTRN_0	16	H'E80100AA	16
	Pipe D transaction counter enable register_0	PIPEDTRE_0	16	H'E80100AC	16
	Pipe D transaction counter register_0	PIPEDTRN_0	16	H'E80100AE	16
	Pipe E transaction counter enable register_0	PIPEETRE_0	16	H'E80100B0	16
	Pipe E transaction counter register_0	PIPEETRN_0	16	H'E80100B2	16
	Pipe F transaction counter enable register_0	PIPEFTRE_0	16	H'E80100B4	16
	Pipe F transaction counter register_0	PIPEFTRN_0	16	H'E80100B6	16
	Pipe 9 transaction counter enable register_0	PIPE9TRE_0	16	H'E80100B8	16
	Pipe 9 transaction counter register_0	PIPE9TRN_0	16	H'E80100BA	16
	Pipe A transaction counter enable register_0	PIPEATRE_0	16	H'E80100BC	16
	Pipe A transaction counter register_0	PIPEATRN_0	16	H'E80100BE	16
	Device address 0 configuration register_0	DEVADD0_0	16	H'E80100D0	16
	Device address 1 configuration register_0	DEVADD1_0	16	H'E80100D2	16
	Device address 2 configuration register_0	DEVADD2_0	16	H'E80100D4	16
	Device address 3 configuration register_0	DEVADD3_0	16	H'E80100D6	16
	Device address 4 configuration register_0	DEVADD4_0	16	H'E80100D8	16
	Device address 5 configuration register_0	DEVADD5_0	16	H'E80100DA	16
	Device address 6 configuration register_0	DEVADD6_0	16	H'E80100DC	16
	Device address 7 configuration register_0	DEVADD7_0	16	H'E80100DE	16
	Device address 8 configuration register_0	DEVADD8_0	16	H'E80100E0	16
	Device address 9 configuration register_0	DEVADD9_0	16	H'E80100E2	16
	Device address A configuration register_0	DEVADDA_0	16	H'E80100E4	16
	Suspend mode register_0	SUSPMODE_0	16	H'E8010102	16
	D0FIFO continuous transfer port register 0_0	D0FIFOB0_0	32	H'E8010160	32
	D0FIFO continuous transfer port register 1_0	D0FIFOB1_0	32	H'E8010164	32
D0FIFO continuous transfer port register 2_0	D0FIFOB2_0	32	H'E8010168	32	
D0FIFO continuous transfer port register 3_0	D0FIFOB3_0	32	H'E801016C	32	
D0FIFO continuous transfer port register 4_0	D0FIFOB4_0	32	H'E8010170	32	
D0FIFO continuous transfer port register 5_0	D0FIFOB5_0	32	H'E8010174	32	
D0FIFO continuous transfer port register 6_0	D0FIFOB6_0	32	H'E8010178	32	
D0FIFO continuous transfer port register 7_0	D0FIFOB7_0	32	H'E801017C	32	
D1FIFO continuous transfer port register 0_0	D1FIFOB0_0	32	H'E8010180	32	
D1FIFO continuous transfer port register 1_0	D1FIFOB1_0	32	H'E8010184	32	
D1FIFO continuous transfer port register 2_0	D1FIFOB2_0	32	H'E8010188	32	

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
USB2.0 host/function module	D1FIFO continuous transfer port register 3_0	D1FIFOB3_0	32	H'E801018C	32
	D1FIFO continuous transfer port register 4_0	D1FIFOB4_0	32	H'E8010190	32
	D1FIFO continuous transfer port register 5_0	D1FIFOB5_0	32	H'E8010194	32
	D1FIFO continuous transfer port register 6_0	D1FIFOB6_0	32	H'E8010198	32
	D1FIFO continuous transfer port register 7_0	D1FIFOB7_0	32	H'E801019C	32
	System configuration control register_1	SYSCFG0_1	16	H'E8207000	16
	CPU bus wait setting register_1	BUSWAIT_1	16	H'E8207002	16
	System configuration status register_1	SYSSTS0_1	16	H'E8207004	16
	Device state control register 0_1	DVSTCTR0_1	16	H'E8207008	16
	Test mode register_1	TESTMODE_1	16	H'E820700C	16
	DMA0-FIFO bus configuration register_1	D0FBCFG_1	16	H'E8207010	16
	DMA1-FIFO bus configuration register_1	D1FBCFG_1	16	H'E8207012	16
	CFIFO port register_1	CFIFO_1	32	H'E8207014	8, 16, 32
	D0FIFO port register_1	D0FIFO_1	32	H'E8207018	8, 16, 32
	D1FIFO port register_1	D1FIFO_1	32	H'E820701C	8, 16, 32
	CFIFO port select register_1	CFIFOSEL_1	16	H'E8207020	16
	CFIFO port control register_1	CFIFOCTR_1	16	H'E8207022	16
	D0FIFO port select register_1	D0FIFOSEL_1	16	H'E8207028	16
	D0FIFO port control register_1	D0FIFOCTR_1	16	H'E820702A	16
	D1FIFO port select register_1	D1FIFOSEL_1	16	H'E820702C	16
	D1FIFO port control register_1	D1FIFOCTR_1	16	H'E820702E	16
	Interrupt enable register 0_1	INTENB0_1	16	H'E8207030	16
	Interrupt enable register 1_1	INTENB1_1	16	H'E8207032	16
	BRDY interrupt enable register_1	BRDYENB_1	16	H'E8207036	16
	NRDY interrupt enable register_1	NRDYENB_1	16	H'E8207038	16
	BEMP interrupt enable register_1	BEMPENB_1	16	H'E820703A	16
	SOF output configuration register_1	SOFCFG_1	16	H'E820703C	16
	Interrupt status register 0_1	INTSTS0_1	16	H'E8207040	16
	Interrupt status register 1_1	INTSTS1_1	16	H'E8207042	16
	BRDY interrupt status register_1	BRDYSTS_1	16	H'E8207046	16
	NRDY interrupt status register_1	NRDYSTS_1	16	H'E8207048	16
	BEMP interrupt status register_1	BEMPSTS_1	16	H'E820704A	16
	Frame number register_1	FRMNUM_1	16	H'E820704C	16
	μFrame number register_1	UFRMNUM_1	16	H'E820704E	16
	USB address register_1	USBADDR_1	16	H'E8207050	16
	USB request type register_1	USBREQ_1	16	H'E8207054	16
	USB request value register_1	USBVAL_1	16	H'E8207056	16
	USB request index register_1	USBINDX_1	16	H'E8207058	16
	USB request length register_1	USBLENG_1	16	H'E820705A	16
	DCP configuration register_1	DCPCFG_1	16	H'E820705C	16
	DCP maximum packet size register_1	DCPMAXP_1	16	H'E820705E	16
	DCP control register_1	DCPCTR_1	16	H'E8207060	16
	Pipe window select register_1	PIPESEL_1	16	H'E8207064	16
	Pipe configuration register_1	PIPECFG_1	16	H'E8207068	16
	Pipe buffer setting register_1	PIPEBUF_1	16	H'E820706A	16
	Pipe maximum packet size register_1	PEMAXP_1	16	H'E820706C	16
	Pipe timing control register_1	PIPEPERI_1	16	H'E820706E	16
	Pipe 1 control register_1	PIPE1CTR_1	16	H'E8207070	16
	Pipe 2 control register_1	PIPE2CTR_1	16	H'E8207072	16
	Pipe 3 control register_1	PIPE3CTR_1	16	H'E8207074	16
Pipe 4 control register_1	PIPE4CTR_1	16	H'E8207076	16	
Pipe 5 control register_1	PIPE5CTR_1	16	H'E8207078	16	

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
USB2.0 host/function module	Pipe 6 control register_1	PIPE6CTR_1	16	H'E820707A	16
	Pipe 7 control register_1	PIPE7CTR_1	16	H'E820707C	16
	Pipe 8 control register_1	PIPE8CTR_1	16	H'E820707E	16
	Pipe 9 control register_1	PIPE9CTR_1	16	H'E8207080	16
	Pipe A control register_1	PIPEACTR_1	16	H'E8207082	16
	Pipe B control register_1	PIPEBCTR_1	16	H'E8207084	16
	Pipe C control register_1	PIPECCTR_1	16	H'E8207086	16
	Pipe D control register_1	PIPEDCTR_1	16	H'E8207088	16
	Pipe E control register_1	PIPEECTR_1	16	H'E820708A	16
	Pipe F control register_1	PIPEFCTR_1	16	H'E820708C	16
	Pipe 1 transaction counter enable register_1	PIPE1TRE_1	16	H'E8207090	16
	Pipe 1 transaction counter register_1	PIPE1TRN_1	16	H'E8207092	16
	Pipe 2 transaction counter enable register_1	PIPE2TRE_1	16	H'E8207094	16
	Pipe 2 transaction counter register_1	PIPE2TRN_1	16	H'E8207096	16
	Pipe 3 transaction counter enable register_1	PIPE3TRE_1	16	H'E8207098	16
	Pipe 3 transaction counter register_1	PIPE3TRN_1	16	H'E820709A	16
	Pipe 4 transaction counter enable register_1	PIPE4TRE_1	16	H'E820709C	16
	Pipe 4 transaction counter register_1	PIPE4TRN_1	16	H'E820709E	16
	Pipe 5 transaction counter enable register_1	PIPE5TRE_1	16	H'E82070A0	16
	Pipe 5 transaction counter register_1	PIPE5TRN_1	16	H'E82070A2	16
	Pipe B transaction counter enable register_1	PIPEBTRE_1	16	H'E82070A4	16
	Pipe B transaction counter register_1	PIPEBTRN_1	16	H'E82070A6	16
	Pipe C transaction counter enable register_1	PIPECTRE_1	16	H'E82070A8	16
	Pipe C transaction counter register_1	PIPECTRN_1	16	H'E82070AA	16
	Pipe D transaction counter enable register_1	PIPEDTRE_1	16	H'E82070AC	16
	Pipe D transaction counter register_1	PIPEDTRN_1	16	H'E82070AE	16
	Pipe E transaction counter enable register_1	PIPEETRE_1	16	H'E82070B0	16
	Pipe E transaction counter register_1	PIPEETRN_1	16	H'E82070B2	16
	Pipe F transaction counter enable register_1	PIPEFTRE_1	16	H'E82070B4	16
	Pipe F transaction counter register_1	PIPEFTRN_1	16	H'E82070B6	16
	Pipe 9 transaction counter enable register_1	PIPE9TRE_1	16	H'E82070B8	16
	Pipe 9 transaction counter register_1	PIPE9TRN_1	16	H'E82070BA	16
	Pipe A transaction counter enable register_1	PIPEATRE_1	16	H'E82070BC	16
	Pipe A transaction counter register_1	PIPEATR_1	16	H'E82070BE	16
	Device address 0 configuration register_1	DEVADD0_1	16	H'E82070D0	16
	Device address 1 configuration register_1	DEVADD1_1	16	H'E82070D2	16
	Device address 2 configuration register_1	DEVADD2_1	16	H'E82070D4	16
	Device address 3 configuration register_1	DEVADD3_1	16	H'E82070D6	16
	Device address 4 configuration register_1	DEVADD4_1	16	H'E82070D8	16
	Device address 5 configuration register_1	DEVADD5_1	16	H'E82070DA	16
	Device address 6 configuration register_1	DEVADD6_1	16	H'E82070DC	16
	Device address 7 configuration register_1	DEVADD7_1	16	H'E82070DE	16
	Device address 8 configuration register_1	DEVADD8_1	16	H'E82070E0	16
	Device address 9 configuration register_1	DEVADD9_1	16	H'E82070E2	16
Device address A configuration register_1	DEVADDA_1	16	H'E82070E4	16	
Suspend mode register_1	SUSPMODE_1	16	H'E8207102	16	
D0FIFO continuous transfer port register 0_1	D0FIFOB0_1	32	H'E8207160	32	
D0FIFO continuous transfer port register 1_1	D0FIFOB1_1	32	H'E8207164	32	
D0FIFO continuous transfer port register 2_1	D0FIFOB2_1	32	H'E8207168	32	
D0FIFO continuous transfer port register 3_1	D0FIFOB3_1	32	H'E820716C	32	
D0FIFO continuous transfer port register 4_1	D0FIFOB4_1	32	H'E8207170	32	
D0FIFO continuous transfer port register 5_1	D0FIFOB5_1	32	H'E8207174	32	

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
USB2.0 host/function module	D0FIFO continuous transfer port register 6_1	D0FIFOB6_1	32	H'E8207178	32
	D0FIFO continuous transfer port register 7_1	D0FIFOB7_1	32	H'E820717C	32
	D1FIFO continuous transfer port register 0_1	D1FIFOB0_1	32	H'E8207180	32
	D1FIFO continuous transfer port register 1_1	D1FIFOB1_1	32	H'E8207184	32
	D1FIFO continuous transfer port register 2_1	D1FIFOB2_1	32	H'E8207188	32
	D1FIFO continuous transfer port register 3_1	D1FIFOB3_1	32	H'E820718C	32
	D1FIFO continuous transfer port register 4_1	D1FIFOB4_1	32	H'E8207190	32
	D1FIFO continuous transfer port register 5_1	D1FIFOB5_1	32	H'E8207194	32
	D1FIFO continuous transfer port register 6_1	D1FIFOB6_1	32	H'E8207198	32
	D1FIFO continuous transfer port register 7_1	D1FIFOB7_1	32	H'E820719C	32
Digital video decoder	ADC control register 1_0	ADCCR1_0	16	H'FCFFB808	16
	Timing generation control register 1_0	TGCR1_0	16	H'FCFFB80E	16
	Timing generation control register 2_0	TGCR2_0	16	H'FCFFB810	16
	Timing generation control register 3_0	TGCR3_0	16	H'FCFFB812	16
	Sync separation control register 1_0	SYNSCR1_0	16	H'FCFFB81A	16
	Sync separation control register 2_0	SYNSCR2_0	16	H'FCFFB81C	16
	Sync separation control register 3_0	SYNSCR3_0	16	H'FCFFB81E	16
	Sync separation control register 4_0	SYNSCR4_0	16	H'FCFFB820	16
	Sync separation control register 5_0	SYNSCR5_0	16	H'FCFFB822	16
	Horizontal AFC control register 1_0	HAFCCR1_0	16	H'FCFFB824	16
	Horizontal AFC control register 2_0	HAFCCR2_0	16	H'FCFFB826	16
	Horizontal AFC control register 3_0	HAFCCR3_0	16	H'FCFFB828	16
	Vertical countdown control register 1_0	VCDWCR1_0	16	H'FCFFB82A	16
	Digital clamp control register 1_0	DCPCR1_0	16	H'FCFFB830	16
	Digital clamp control register 2_0	DCPCR2_0	16	H'FCFFB832	16
	Digital clamp control register 3_0	DCPCR3_0	16	H'FCFFB834	16
	Digital clamp control register 4_0	DCPCR4_0	16	H'FCFFB836	16
	Digital clamp control register 5_0	DCPCR5_0	16	H'FCFFB838	16
	Digital clamp control register 6_0	DCPCR6_0	16	H'FCFFB83A	16
	Digital clamp control register 7_0	DCPCR7_0	16	H'FCFFB83C	16
	Digital clamp control register 8_0	DCPCR8_0	16	H'FCFFB83E	16
	Noise detection control register_0	NSDCR_0	16	H'FCFFB840	16
	Burst lock/chroma decoding control register_0	BTLCR_0	16	H'FCFFB842	16
	Burst gate pulse control register_0	BTGPCR_0	16	H'FCFFB844	16
	ACC control register 1_0	ACCCR1_0	16	H'FCFFB846	16
	ACC control register 2_0	ACCCR2_0	16	H'FCFFB848	16
	ACC control register 3_0	ACCCR3_0	16	H'FCFFB84A	16
	TINT control register_0	TINTCR_0	16	H'FCFFB84C	16
	Y/C delay/chroma decoding control register_0	YDCR_0	16	H'FCFFB84E	16
	AGC control register 1_0	AGCCR1_0	16	H'FCFFB850	16
	AGC control register 2_0	AGCCR2_0	16	H'FCFFB852	16
	Peak limiter control register_0	PKLIMITCR_0	16	H'FCFFB854	16
	Over-range control register 1_0	RGORCR1_0	16	H'FCFFB856	16
	Over-range control register 2_0	RGORCR2_0	16	H'FCFFB858	16
	Over-range control register 3_0	RGORCR3_0	16	H'FCFFB85A	16
	Over-range control register 4_0	RGORCR4_0	16	H'FCFFB85C	16
	Over-range control register 5_0	RGORCR5_0	16	H'FCFFB85E	16
	Over-range control register 6_0	RGORCR6_0	16	H'FCFFB860	16
	Over-range control register 7_0	RGORCR7_0	16	H'FCFFB862	16
	Feedback control register for horizontal AFC phase comparator_0	AFCPFCR_0	16	H'FCFFB87C	16
Register update enable register_0	RUPDCR_0	16	H'FCFFB87E	16	
Sync separation status/vertical cycle read register_0	VSYNSCR_0	16	H'FCFFB880	16	



Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Digital video decoder	Horizontal cycle read register_0	HSYNCSR_0	16	H'FCFFB882	16
	Digital clamp read register 1_0	DCPSR1_0	16	H'FCFFB884	16
	Digital clamp read register 2_0	DCPSR2_0	16	H'FCFFB886	16
	Noise detection read register_0	NSDSR_0	16	H'FCFFB88C	16
	Chroma decoding read register 1_0	CROMASR1_0	16	H'FCFFB88E	16
	Chroma decoding read register 2_0	CROMASR2_0	16	H'FCFFB890	16
	Sync separation read register_0	SYNCSR_0	16	H'FCFFB892	16
	AGC control read register 1_0	AGCSR1_0	16	H'FCFFB894	16
	AGC control read register 2_0	AGCSR2_0	16	H'FCFFB896	16
	Y/C separation control register 3_0	YCSCR3_0	16	H'FCFFB904	16
	Y/C separation control register 4_0	YCSCR4_0	16	H'FCFFB906	16
	Y/C separation control register 5_0	YCSCR5_0	16	H'FCFFB908	16
	Y/C separation control register 6_0	YCSCR6_0	16	H'FCFFB90A	16
	Y/C separation control register 7_0	YCSCR7_0	16	H'FCFFB90C	16
	Y/C separation control register 8_0	YCSCR8_0	16	H'FCFFB90E	16
	Y/C separation control register 9_0	YCSCR9_0	16	H'FCFFB910	16
	Y/C separation control register 11_0	YCSCR11_0	16	H'FCFFB914	16
	Y/C separation control register 12_0	YCSCR12_0	16	H'FCFFB916	16
	Digital clamp control register 9_0	DCPCR9_0	16	H'FCFFB980	16
	Chroma filter TAP coefficient (WA_F0) register for Y/C separation_0	YCTWA_F0_0	16	H'FCFFB992	16
	Chroma filter TAP coefficient (WA_F1) register for Y/C separation_0	YCTWA_F1_0	16	H'FCFFB994	16
	Chroma filter TAP coefficient (WA_F2) register for Y/C separation_0	YCTWA_F2_0	16	H'FCFFB996	16
	Chroma filter TAP coefficient (WA_F3) register for Y/C separation_0	YCTWA_F3_0	16	H'FCFFB998	16
	Chroma filter TAP coefficient (WA_F4) register for Y/C separation_0	YCTWA_F4_0	16	H'FCFFB99A	16
	Chroma filter TAP coefficient (WA_F5) register for Y/C separation_0	YCTWA_F5_0	16	H'FCFFB99C	16
	Chroma filter TAP coefficient (WA_F6) register for Y/C separation_0	YCTWA_F6_0	16	H'FCFFB99E	16
	Chroma filter TAP coefficient (WA_F7) register for Y/C separation_0	YCTWA_F7_0	16	H'FCFFB9A0	16
	Chroma filter TAP coefficient (WA_F8) register for Y/C separation_0	YCTWA_F8_0	16	H'FCFFB9A2	16
	Chroma filter TAP coefficient (WB_F0) register for Y/C separation_0	YCTWB_F0_0	16	H'FCFFB9A4	16
	Chroma filter TAP coefficient (WB_F1) register for Y/C separation_0	YCTWB_F1_0	16	H'FCFFB9A6	16
	Chroma filter TAP coefficient (WB_F2) register for Y/C separation_0	YCTWB_F2_0	16	H'FCFFB9A8	16
	Chroma filter TAP coefficient (WB_F3) register for Y/C separation_0	YCTWB_F3_0	16	H'FCFFB9AA	16
	Chroma filter TAP coefficient (WB_F4) register for Y/C separation_0	YCTWB_F4_0	16	H'FCFFB9AC	16
	Chroma filter TAP coefficient (WB_F5) register for Y/C separation_0	YCTWB_F5_0	16	H'FCFFB9AE	16
	Chroma filter TAP coefficient (WB_F6) register for Y/C separation_0	YCTWB_F6_0	16	H'FCFFB9B0	16
	Chroma filter TAP coefficient (WB_F7) register for Y/C separation_0	YCTWB_F7_0	16	H'FCFFB9B2	16
	Chroma filter TAP coefficient (WB_F8) register for Y/C separation_0	YCTWB_F8_0	16	H'FCFFB9B4	16
	Chroma filter TAP coefficient (NA_F0) register for Y/C separation_0	YCTNA_F0_0	16	H'FCFFB9B6	16
	Chroma filter TAP coefficient (NA_F1) register for Y/C separation_0	YCTNA_F1_0	16	H'FCFFB9B8	16
	Chroma filter TAP coefficient (NA_F2) register for Y/C separation_0	YCTNA_F2_0	16	H'FCFFB9BA	16

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Digital video decoder	Chroma filter TAP coefficient (NA_F3) register for Y/C separation_0	YCTNA_F3_0	16	H'FCFFB9BC	16
	Chroma filter TAP coefficient (NA_F4) register for Y/C separation_0	YCTNA_F4_0	16	H'FCFFB9BE	16
	Chroma filter TAP coefficient (NA_F5) register for Y/C separation_0	YCTNA_F5_0	16	H'FCFFB9C0	16
	Chroma filter TAP coefficient (NA_F6) register for Y/C separation_0	YCTNA_F6_0	16	H'FCFFB9C2	16
	Chroma filter TAP coefficient (NA_F7) register for Y/C separation_0	YCTNA_F7_0	16	H'FCFFB9C4	16
	Chroma filter TAP coefficient (NA_F8) register for Y/C separation_0	YCTNA_F8_0	16	H'FCFFB9C6	16
	Chroma filter TAP coefficient (NB_F0) register for Y/C separation_0	YCTNB_F0_0	16	H'FCFFB9C8	16
	Chroma filter TAP coefficient (NB_F1) register for Y/C separation_0	YCTNB_F1_0	16	H'FCFFB9CA	16
	Chroma filter TAP coefficient (NB_F2) register for Y/C separation_0	YCTNB_F2_0	16	H'FCFFB9CC	16
	Chroma filter TAP coefficient (NB_F3) register for Y/C separation_0	YCTNB_F3_0	16	H'FCFFB9CE	16
	Chroma filter TAP coefficient (NB_F4) register for Y/C separation_0	YCTNB_F4_0	16	H'FCFFB9D0	16
	Chroma filter TAP coefficient (NB_F5) register for Y/C separation_0	YCTNB_F5_0	16	H'FCFFB9D2	16
	Chroma filter TAP coefficient (NB_F6) register for Y/C separation_0	YCTNB_F6_0	16	H'FCFFB9D4	16
	Chroma filter TAP coefficient (NB_F7) register for Y/C separation_0	YCTNB_F7_0	16	H'FCFFB9D6	16
	Chroma filter TAP coefficient (NB_F8) register for Y/C separation_0	YCTNB_F8_0	16	H'FCFFB9D8	16
	Luminance (Y) signal gain control register_0	YGAINCR_0	16	H'FCFFBA00	16
	Color difference (Cb) signal gain control register_0	CBGAINCR_0	16	H'FCFFBA02	16
	Color difference (Cr) signal gain control register_0	CRGAINCR_0	16	H'FCFFBA04	16
	PGA register update_0	PGA_UPDATE_0	16	H'FCFFBA80	16
	PGA control register_0	PGACR_0	16	H'FCFFBA82	16
	ADC control register 2_0	ADCCR2_0	16	H'FCFFBA84	16
	ADC control register 1_1	ADCCR1_1	16	H'FCFFA008	16
	Timing generation control register 1_1	TGCR1_1	16	H'FCFFA00E	16
	Timing generation control register 2_1	TGCR2_1	16	H'FCFFA010	16
	Timing generation control register 3_1	TGCR3_1	16	H'FCFFA012	16
	Sync separation control register 1_1	SYNSCR1_1	16	H'FCFFA01A	16
	Sync separation control register 2_1	SYNSCR2_1	16	H'FCFFA01C	16
	Sync separation control register 3_1	SYNSCR3_1	16	H'FCFFA01E	16
	Sync separation control register 4_1	SYNSCR4_1	16	H'FCFFA020	16
	Sync separation control register 5_1	SYNSCR5_1	16	H'FCFFA022	16
	Horizontal AFC control register 1_1	HAFCCR1_1	16	H'FCFFA024	16
	Horizontal AFC control register 2_1	HAFCCR2_1	16	H'FCFFA026	16
	Horizontal AFC control register 3_1	HAFCCR3_1	16	H'FCFFA028	16
	Vertical countdown control register 1_1	VCDWCR1_1	16	H'FCFFA02A	16
	Digital clamp control register 1_1	DCPCR1_1	16	H'FCFFA030	16
	Digital clamp control register 2_1	DCPCR2_1	16	H'FCFFA032	16
	Digital clamp control register 3_1	DCPCR3_1	16	H'FCFFA034	16
	Digital clamp control register 4_1	DCPCR4_1	16	H'FCFFA036	16
	Digital clamp control register 5_1	DCPCR5_1	16	H'FCFFA038	16
	Digital clamp control register 6_1	DCPCR6_1	16	H'FCFFA03A	16
	Digital clamp control register 7_1	DCPCR7_1	16	H'FCFFA03C	16
	Digital clamp control register 8_1	DCPCR8_1	16	H'FCFFA03E	16
Noise detection control register_1	NSDCR_1	16	H'FCFFA040	16	
Burst lock/chroma decoding control register_1	BTLCR_1	16	H'FCFFA042	16	

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Digital video decoder	Burst gate pulse control register_1	BTGPCR_1	16	H'FCFFA044	16
	ACC control register 1_1	ACCCR1_1	16	H'FCFFA046	16
	ACC control register 2_1	ACCCR2_1	16	H'FCFFA048	16
	ACC control register 3_1	ACCCR3_1	16	H'FCFFA04A	16
	TINT control register_1	TINTCR_1	16	H'FCFFA04C	16
	Y/C delay/chroma decoding control register_1	YCDCR_1	16	H'FCFFA04E	16
	AGC control register 1_1	AGCCR1_1	16	H'FCFFA050	16
	AGC control register 2_1	AGCCR2_1	16	H'FCFFA052	16
	Peak limiter control register_1	PKLIMITCR_1	16	H'FCFFA054	16
	Over-range control register 1_1	RGORCR1_1	16	H'FCFFA056	16
	Over-range control register 2_1	RGORCR2_1	16	H'FCFFA058	16
	Over-range control register 3_1	RGORCR3_1	16	H'FCFFA05A	16
	Over-range control register 4_1	RGORCR4_1	16	H'FCFFA05C	16
	Over-range control register 5_1	RGORCR5_1	16	H'FCFFA05E	16
	Over-range control register 6_1	RGORCR6_1	16	H'FCFFA060	16
	Over-range control register 7_1	RGORCR7_1	16	H'FCFFA062	16
	Feedback control register for horizontal AFC phase comparator_1	AFCPFCR_1	16	H'FCFFA07C	16
	Register update enable register_1	RUPDCR_1	16	H'FCFFA07E	16
	Sync separation status/vertical cycle read register_1	VSYNCSR_1	16	H'FCFFA080	16
	Horizontal cycle read register_1	HSYNCSR_1	16	H'FCFFA082	16
	Digital clamp read register 1_1	DCPSR1_1	16	H'FCFFA084	16
	Digital clamp read register 2_1	DCPSR2_1	16	H'FCFFA086	16
	Noise detection read register_1	NSDSR_1	16	H'FCFFA08C	16
	Chroma decoding read register 1_1	CROMASR1_1	16	H'FCFFA08E	16
	Chroma decoding read register 2_1	CROMASR2_1	16	H'FCFFA090	16
	Sync separation read register_1	SYNCSR_1	16	H'FCFFA092	16
	AGC control read register 1_1	AGCCSR1_1	16	H'FCFFA094	16
	AGC control read register 2_1	AGCCSR2_1	16	H'FCFFA096	16
	Y/C separation control register 3_1	YCSCR3_1	16	H'FCFFA104	16
	Y/C separation control register 4_1	YCSCR4_1	16	H'FCFFA106	16
	Y/C separation control register 5_1	YCSCR5_1	16	H'FCFFA108	16
	Y/C separation control register 6_1	YCSCR6_1	16	H'FCFFA10A	16
	Y/C separation control register 7_1	YCSCR7_1	16	H'FCFFA10C	16
	Y/C separation control register 8_1	YCSCR8_1	16	H'FCFFA10E	16
	Y/C separation control register 9_1	YCSCR9_1	16	H'FCFFA110	16
	Y/C separation control register 11_1	YCSCR11_1	16	H'FCFFA114	16
	Y/C separation control register 12_1	YCSCR12_1	16	H'FCFFA116	16
	Digital clamp control register 9_1	DCPCR9_1	16	H'FCFFA180	16
	Chroma filter TAP coefficient (WA_F0) register for Y/C separation_1	YCTWA_F0_1	16	H'FCFFA192	16
	Chroma filter TAP coefficient (WA_F1) register for Y/C separation_1	YCTWA_F1_1	16	H'FCFFA194	16
	Chroma filter TAP coefficient (WA_F2) register for Y/C separation_1	YCTWA_F2_1	16	H'FCFFA196	16
	Chroma filter TAP coefficient (WA_F3) register for Y/C separation_1	YCTWA_F3_1	16	H'FCFFA198	16
Chroma filter TAP coefficient (WA_F4) register for Y/C separation_1	YCTWA_F4_1	16	H'FCFFA19A	16	
Chroma filter TAP coefficient (WA_F5) register for Y/C separation_1	YCTWA_F5_1	16	H'FCFFA19C	16	
Chroma filter TAP coefficient (WA_F6) register for Y/C separation_1	YCTWA_F6_1	16	H'FCFFA19E	16	
Chroma filter TAP coefficient (WA_F7) register for Y/C separation_1	YCTWA_F7_1	16	H'FCFFA1A0	16	
Chroma filter TAP coefficient (WA_F8) register for Y/C separation_1	YCTWA_F8_1	16	H'FCFFA1A2	16	

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Digital video decoder	Chroma filter TAP coefficient (WB_F0) register for Y/C separation_1	YCTWB_F0_1	16	H'FCFFA1A4	16
	Chroma filter TAP coefficient (WB_F1) register for Y/C separation_1	YCTWB_F1_1	16	H'FCFFA1A6	16
	Chroma filter TAP coefficient (WB_F2) register for Y/C separation_1	YCTWB_F2_1	16	H'FCFFA1A8	16
	Chroma filter TAP coefficient (WB_F3) register for Y/C separation_1	YCTWB_F3_1	16	H'FCFFA1AA	16
	Chroma filter TAP coefficient (WB_F4) register for Y/C separation_1	YCTWB_F4_1	16	H'FCFFA1AC	16
	Chroma filter TAP coefficient (WB_F5) register for Y/C separation_1	YCTWB_F5_1	16	H'FCFFA1AE	16
	Chroma filter TAP coefficient (WB_F6) register for Y/C separation_1	YCTWB_F6_1	16	H'FCFFA1B0	16
	Chroma filter TAP coefficient (WB_F7) register for Y/C separation_1	YCTWB_F7_1	16	H'FCFFA1B2	16
	Chroma filter TAP coefficient (WB_F8) register for Y/C separation_1	YCTWB_F8_1	16	H'FCFFA1B4	16
	Chroma filter TAP coefficient (NA_F0) register for Y/C separation_1	YCTNA_F0_1	16	H'FCFFA1B6	16
	Chroma filter TAP coefficient (NA_F1) register for Y/C separation_1	YCTNA_F1_1	16	H'FCFFA1B8	16
	Chroma filter TAP coefficient (NA_F2) register for Y/C separation_1	YCTNA_F2_1	16	H'FCFFA1BA	16
	Chroma filter TAP coefficient (NA_F3) register for Y/C separation_1	YCTNA_F3_1	16	H'FCFFA1BC	16
	Chroma filter TAP coefficient (NA_F4) register for Y/C separation_1	YCTNA_F4_1	16	H'FCFFA1BE	16
	Chroma filter TAP coefficient (NA_F5) register for Y/C separation_1	YCTNA_F5_1	16	H'FCFFA1C0	16
	Chroma filter TAP coefficient (NA_F6) register for Y/C separation_1	YCTNA_F6_1	16	H'FCFFA1C2	16
	Chroma filter TAP coefficient (NA_F7) register for Y/C separation_1	YCTNA_F7_1	16	H'FCFFA1C4	16
	Chroma filter TAP coefficient (NA_F8) register for Y/C separation_1	YCTNA_F8_1	16	H'FCFFA1C6	16
	Chroma filter TAP coefficient (NB_F0) register for Y/C separation_1	YCTNB_F0_1	16	H'FCFFA1C8	16
	Chroma filter TAP coefficient (NB_F1) register for Y/C separation_1	YCTNB_F1_1	16	H'FCFFA1CA	16
	Chroma filter TAP coefficient (NB_F2) register for Y/C separation_1	YCTNB_F2_1	16	H'FCFFA1CC	16
	Chroma filter TAP coefficient (NB_F3) register for Y/C separation_1	YCTNB_F3_1	16	H'FCFFA1CE	16
	Chroma filter TAP coefficient (NB_F4) register for Y/C separation_1	YCTNB_F4_1	16	H'FCFFA1D0	16
	Chroma filter TAP coefficient (NB_F5) register for Y/C separation_1	YCTNB_F5_1	16	H'FCFFA1D2	16
	Chroma filter TAP coefficient (NB_F6) register for Y/C separation_1	YCTNB_F6_1	16	H'FCFFA1D4	16
	Chroma filter TAP coefficient (NB_F7) register for Y/C separation_1	YCTNB_F7_1	16	H'FCFFA1D6	16
	Chroma filter TAP coefficient (NB_F8) register for Y/C separation_1	YCTNB_F8_1	16	H'FCFFA1D8	16
	Luminance (Y) signal gain control register_1	YGAINCR_1	16	H'FCFFA200	16
	Color difference (Cb) signal gain control register_1	CBGAINCR_1	16	H'FCFFA202	16
	Color difference (Cr) signal gain control register_1	CRGAINCR_1	16	H'FCFFA204	16
	PGA register update_1	PGA_UPDATE_1	16	H'FCFFA280	16
	PGA control register_1	PGACR_1	16	H'FCFFA282	16
ADC control register 2_1	ADCCR2_1	16	H'FCFFA284	16	

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Video display controller 5 channel 0	External input block register update control register	INP_UPDATE	32	H'FCFF7400	32
	Input select control register	INP_SEL_CNT	32	H'FCFF7404	32
	External input sync signal control register	INP_EXT_SYNC_CNT	32	H'FCFF7408	32
	Vsync signal phase adjustment register	INP_VSYNC_PH_ADJ	32	H'FCFF740C	32
	Sync signal phase adjustment register	INP_DLY_ADJ	32	H'FCFF7410	32
	Image quality adjustment block register update control register	IMGCNT_UPDATE	32	H'FCFF7480	32
	NR control register 0	IMGCNT_NR_CNT0	32	H'FCFF7484	32
	NR control register 1	IMGCNT_NR_CNT1	32	H'FCFF7488	32
	Image quality adjustment block matrix mode register	IMGCNT_MTX_MODE	32	H'FCFF74A0	32
	Image quality adjustment block matrix YG adjustment register 0	IMGCNT_MTX_YG_ADJ0	32	H'FCFF74A4	32
	Image quality adjustment block matrix YG adjustment register 1	IMGCNT_MTX_YG_ADJ1	32	H'FCFF74A8	32
	Image quality adjustment block matrix CBB adjustment register 0	IMGCNT_MTX_CBB_ADJ0	32	H'FCFF74AC	32
	Image quality adjustment block matrix CBB adjustment register 1	IMGCNT_MTX_CBB_ADJ1	32	H'FCFF74B0	32
	Image quality adjustment block matrix CRR adjustment register 0	IMGCNT_MTX_CRR_ADJ0	32	H'FCFF74B4	32
	Image quality adjustment block matrix CRR adjustment register 1	IMGCNT_MTX_CRR_ADJ1	32	H'FCFF74B8	32
	Dynamic range compression register	IMGCNT_DRC_REG	32	H'FCFF74C0	32
	SCL0 register update control register (SC0)	SC0_SCL0_UPDATE	32	H'FCFF7500	32
	Mask control register (SC0)	SC0_SCL0_FRC1	32	H'FCFF7504	32
	Missing Vsync compensation control register (SC0)	SC0_SCL0_FRC2	32	H'FCFF7508	32
	Output sync select register (SC0)	SC0_SCL0_FRC3	32	H'FCFF750C	32
	Free-running period control register (SC0)	SC0_SCL0_FRC4	32	H'FCFF7510	32
	Output delay control register (SC0)	SC0_SCL0_FRC5	32	H'FCFF7514	32
	Full-screen vertical size register (SC0)	SC0_SCL0_FRC6	32	H'FCFF7518	32
	Full-screen horizontal size register (SC0)	SC0_SCL0_FRC7	32	H'FCFF751C	32
	Vsync detection register (SC0)	SC0_SCL0_FRC9	32	H'FCFF7524	32
	Status monitor 0 register (SC0)	SC0_SCL0_MON0	16	H'FCFF7528	16
	Interrupt control register (SC0)	SC0_SCL0_INT	16	H'FCFF752A	16
	Scaling-down control register (SC0)	SC0_SCL0_DS1	32	H'FCFF752C	32
	Vertical capture size register (SC0)	SC0_SCL0_DS2	32	H'FCFF7530	32
	Horizontal capture size register (SC0)	SC0_SCL0_DS3	32	H'FCFF7534	32
	Horizontal scale down register (SC0)	SC0_SCL0_DS4	32	H'FCFF7538	32
	Initial vertical phase register (SC0)	SC0_SCL0_DS5	32	H'FCFF753C	32
	Vertical scaling register (SC0)	SC0_SCL0_DS6	32	H'FCFF7540	32
	Scaling-down control block output size register (SC0)	SC0_SCL0_DS7	32	H'FCFF7544	32
	Scaling-up control register (SC0)	SC0_SCL0_US1	32	H'FCFF7548	32
	Output image vertical size register (SC0)	SC0_SCL0_US2	32	H'FCFF754C	32
	Output image horizontal size register (SC0)	SC0_SCL0_US3	32	H'FCFF7550	32
	Scaling-up control block input size register (SC0)	SC0_SCL0_US4	32	H'FCFF7554	32
	Horizontal scale up register (SC0)	SC0_SCL0_US5	32	H'FCFF7558	32
	Horizontal scale up initial phase register (SC0)	SC0_SCL0_US6	32	H'FCFF755C	32
	Trimming register (SC0)	SC0_SCL0_US7	32	H'FCFF7560	32
	Frame buffer read select register (SC0)	SC0_SCL0_US8	32	H'FCFF7564	32
	Background color register (SC0)	SC0_SCL0_OVR1	32	H'FCFF756C	32
	SCL1 register update control register (SC0)	SC0_SCL1_UPDATE	32	H'FCFF7580	32
Writing mode register (SC0)	SC0_SCL1_WR1	32	H'FCFF7588	32	
Write address register 1T (SC0)	SC0_SCL1_WR2	32	H'FCFF758C	32	
Write address register 2T (SC0)	SC0_SCL1_WR3	32	H'FCFF7590	32	
Write address register 3T (SC0)	SC0_SCL1_WR4	32	H'FCFF7594	32	
Frame sub-sampling register (SC0)	SC0_SCL1_WR5	32	H'FCFF759C	32	
Bit reduction register (SC0)	SC0_SCL1_WR6	32	H'FCFF75A0	32	
Write detection register (SC0)	SC0_SCL1_WR7	32	H'FCFF75A4	32	
Write address register 1B (SC0)	SC0_SCL1_WR8	32	H'FCFF75A8	32	

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Video display controller 5 channel 0	Write address register 2B (SC0)	SC0_SCL1_WR9	32	H'FCFF75AC	32
	Write address register 3B (SC0)	SC0_SCL1_WR10	32	H'FCFF75B0	32
	Write detection register B (SC0)	SC0_SCL1_WR11	32	H'FCFF75B4	32
	Status monitor 1 register (SC0)	SC0_SCL1_MON1	32	H'FCFF75B8	32
	Pointer buffer 0 register (SC0)	SC0_SCL1_PBUF0	32	H'FCFF75BC	32
	Pointer buffer 1 register (SC0)	SC0_SCL1_PBUF1	32	H'FCFF75C0	32
	Pointer buffer 2 register (SC0)	SC0_SCL1_PBUF2	32	H'FCFF75C4	32
	Pointer buffer 3 register (SC0)	SC0_SCL1_PBUF3	32	H'FCFF75C8	32
	Pointer buffer and field information register (SC0)	SC0_SCL1_PBUF_FLD	32	H'FCFF75CC	32
	Pointer buffer control register (SC0)	SC0_SCL1_PBUF_CNT	32	H'FCFF75D0	32
	Graphics 0 register update control register	GR0_UPDATE	32	H'FCFF7600	32
	Frame buffer read control register (graphics 0)	GR0_FLM_RD	32	H'FCFF7604	32
	Frame buffer control register 1 (graphics 0)	GR0_FLM1	32	H'FCFF7608	32
	Frame buffer control register 2 (graphics 0)	GR0_FLM2	32	H'FCFF760C	32
	Frame buffer control register 3 (graphics 0)	GR0_FLM3	32	H'FCFF7610	32
	Frame buffer control register 4 (graphics 0)	GR0_FLM4	32	H'FCFF7614	32
	Frame buffer control register 5 (graphics 0)	GR0_FLM5	32	H'FCFF7618	32
	Frame buffer control register 6 (graphics 0)	GR0_FLM6	32	H'FCFF761C	32
	Alpha blending control register 1 (graphics 0)	GR0_AB1	32	H'FCFF7620	32
	Alpha blending control register 2 (graphics 0)	GR0_AB2	32	H'FCFF7624	32
	Alpha blending control register 3 (graphics 0)	GR0_AB3	32	H'FCFF7628	32
	Alpha blending control register 7 (graphics 0)	GR0_AB7	32	H'FCFF7638	32
	Alpha blending control register 8 (graphics 0)	GR0_AB8	32	H'FCFF763C	32
	Alpha blending control register 9 (graphics 0)	GR0_AB9	32	H'FCFF7640	32
	Alpha blending control register 10 (graphics 0)	GR0_AB10	32	H'FCFF7644	32
	Alpha blending control register 11 (graphics 0)	GR0_AB11	32	H'FCFF7648	32
	Background color control register (graphics 0)	GR0_BASE	32	H'FCFF764C	32
	CLUT table control register (graphics 0)	GR0_CLUT	32	H'FCFF7650	32
	SCL0 register update control register (SC1)	SC1_SCL0_UPDATE	32	H'FCFF7C00	32
	Mask control register (SC1)	SC1_SCL0_FRC1	32	H'FCFF7C04	32
	Missing Vsync compensation control register (SC1)	SC1_SCL0_FRC2	32	H'FCFF7C08	32
	Output sync select register (SC1)	SC1_SCL0_FRC3	32	H'FCFF7C0C	32
	Free-running period control register (SC1)	SC1_SCL0_FRC4	32	H'FCFF7C10	32
	Output delay control register (SC1)	SC1_SCL0_FRC5	32	H'FCFF7C14	32
	Full-screen vertical size register (SC1)	SC1_SCL0_FRC6	32	H'FCFF7C18	32
	Full-screen horizontal size register (SC1)	SC1_SCL0_FRC7	32	H'FCFF7C1C	32
	Vsync detection register (SC1)	SC1_SCL0_FRC9	32	H'FCFF7C24	32
	Status monitor 0 register (SC1)	SC1_SCL0_MON0	16	H'FCFF7C28	16
	Interrupt control register (SC1)	SC1_SCL0_INT	16	H'FCFF7C2A	16
	Scaling-down control register (SC1)	SC1_SCL0_DS1	32	H'FCFF7C2C	32
	Vertical capture size register (SC1)	SC1_SCL0_DS2	32	H'FCFF7C30	32
	Horizontal capture size register (SC1)	SC1_SCL0_DS3	32	H'FCFF7C34	32
	Horizontal scale down register (SC1)	SC1_SCL0_DS4	32	H'FCFF7C38	32
	Initial vertical phase register (SC1)	SC1_SCL0_DS5	32	H'FCFF7C3C	32
	Vertical scaling register (SC1)	SC1_SCL0_DS6	32	H'FCFF7C40	32
	Scaling-down control block output size register (SC1)	SC1_SCL0_DS7	32	H'FCFF7C44	32
	Scaling-up control register (SC1)	SC1_SCL0_US1	32	H'FCFF7C48	32
	Output image vertical size register (SC1)	SC1_SCL0_US2	32	H'FCFF7C4C	32
	Output image horizontal size register (SC1)	SC1_SCL0_US3	32	H'FCFF7C50	32
	Scaling-up control block input size register (SC1)	SC1_SCL0_US4	32	H'FCFF7C54	32
	Horizontal scale up register (SC1)	SC1_SCL0_US5	32	H'FCFF7C58	32
	Horizontal scale up initial phase register (SC1)	SC1_SCL0_US6	32	H'FCFF7C5C	32

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Video display controller 5 channel 0	Trimming register (SC1)	SC1_SCL0_US7	32	H'FCFF7C60	32
	Frame buffer read select register (SC1)	SC1_SCL0_US8	32	H'FCFF7C64	32
	Background color register (SC1)	SC1_SCL0_OVR1	32	H'FCFF7C6C	32
	SCL1 register update control register (SC1)	SC1_SCL1_UPDATE	32	H'FCFF7C80	32
	Writing mode register (SC1)	SC1_SCL1_WR1	32	H'FCFF7C88	32
	Write address register 1T (SC1)	SC1_SCL1_WR2	32	H'FCFF7C8C	32
	Write address register 2T (SC1)	SC1_SCL1_WR3	32	H'FCFF7C90	32
	Write address register 3T (SC1)	SC1_SCL1_WR4	32	H'FCFF7C94	32
	Frame sub-sampling register (SC1)	SC1_SCL1_WR5	32	H'FCFF7C9C	32
	Bit reduction register (SC1)	SC1_SCL1_WR6	32	H'FCFF7CA0	32
	Write detection register (SC1)	SC1_SCL1_WR7	32	H'FCFF7CA4	32
	Write address register 1B (SC1)	SC1_SCL1_WR8	32	H'FCFF7CA8	32
	Write address register 2B (SC1)	SC1_SCL1_WR9	32	H'FCFF7CAC	32
	Write address register 3B (SC1)	SC1_SCL1_WR10	32	H'FCFF7CB0	32
	Write detection register B (SC1)	SC1_SCL1_WR11	32	H'FCFF7CB4	32
	Status monitor 1 register (SC1)	SC1_SCL1_MON1	32	H'FCFF7CB8	32
	Pointer buffer 0 register (SC1)	SC1_SCL1_PBUF0	32	H'FCFF7CBC	32
	Pointer buffer 1 register (SC1)	SC1_SCL1_PBUF1	32	H'FCFF7CC0	32
	Pointer buffer 2 register (SC1)	SC1_SCL1_PBUF2	32	H'FCFF7CC4	32
	Pointer buffer 3 register (SC1)	SC1_SCL1_PBUF3	32	H'FCFF7CC8	32
	Pointer buffer and field information register (SC1)	SC1_SCL1_PBUF_FLD	32	H'FCFF7CCC	32
	Pointer buffer control register (SC1)	SC1_SCL1_PBUF_CNT	32	H'FCFF7CD0	32
	Graphics 1 register update control register	GR1_UPDATE	32	H'FCFF7D00	32
	Frame buffer read control register (graphics 1)	GR1_FLM_RD	32	H'FCFF7D04	32
	Frame buffer control register 1 (graphics 1)	GR1_FLM1	32	H'FCFF7D08	32
	Frame buffer control register 2 (graphics 1)	GR1_FLM2	32	H'FCFF7D0C	32
	Frame buffer control register 3 (graphics 1)	GR1_FLM3	32	H'FCFF7D10	32
	Frame buffer control register 4 (graphics 1)	GR1_FLM4	32	H'FCFF7D14	32
	Frame buffer control register 5 (graphics 1)	GR1_FLM5	32	H'FCFF7D18	32
	Frame buffer control register 6 (graphics 1)	GR1_FLM6	32	H'FCFF7D1C	32
	Alpha blending control register 1 (graphics 1)	GR1_AB1	32	H'FCFF7D20	32
	Alpha blending control register 2 (graphics 1)	GR1_AB2	32	H'FCFF7D24	32
	Alpha blending control register 3 (graphics 1)	GR1_AB3	32	H'FCFF7D28	32
	Alpha blending control register 4 (graphics 1)	GR1_AB4	32	H'FCFF7D2C	32
	Alpha blending control register 5 (graphics 1)	GR1_AB5	32	H'FCFF7D30	32
	Alpha blending control register 6 (graphics 1)	GR1_AB6	32	H'FCFF7D34	32
	Alpha blending control register 7 (graphics 1)	GR1_AB7	32	H'FCFF7D38	32
	Alpha blending control register 8 (graphics 1)	GR1_AB8	32	H'FCFF7D3C	32
	Alpha blending control register 9 (graphics 1)	GR1_AB9	32	H'FCFF7D40	32
	Alpha blending control register 10 (graphics 1)	GR1_AB10	32	H'FCFF7D44	32
	Alpha blending control register 11 (graphics 1)	GR1_AB11	32	H'FCFF7D48	32
	Background color control register (graphics 1)	GR1_BASE	32	H'FCFF7D4C	32
	CLUT table control register (graphics 1)	GR1_CLUT	32	H'FCFF7D50	32
	Status monitor register (graphics 1)	GR1_MON	32	H'FCFF7D54	32
	Register update control register in image quality improver (image quality improver 0)	ADJ0_UPDATE	32	H'FCFF7680	32
	Black stretch register (image quality improver 0)	ADJ0_BKSTR_SET	32	H'FCFF7684	32
	Enhancer timing adjustment register 1 (image quality improver 0)	ADJ0_ENH_TIM1	32	H'FCFF7688	32
	Enhancer timing adjustment register 2 (image quality improver 0)	ADJ0_ENH_TIM2	32	H'FCFF768C	32
	Enhancer timing adjustment register 3 (image quality improver 0)	ADJ0_ENH_TIM3	32	H'FCFF7690	32
	Enhancer sharpness register 1 (image quality improver 0)	ADJ0_ENH_SHP1	32	H'FCFF7694	32
Enhancer sharpness register 2 (image quality improver 0)	ADJ0_ENH_SHP2	32	H'FCFF7698	32	
Enhancer sharpness register 3 (image quality improver 0)	ADJ0_ENH_SHP3	32	H'FCFF769C	32	

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Video display controller 5 channel 0	Enhancer sharpness register 4 (image quality improver 0)	ADJ0_ENH_SHP4	32	H'FCFF76A0	32
	Enhancer sharpness register 5 (image quality improver 0)	ADJ0_ENH_SHP5	32	H'FCFF76A4	32
	Enhancer sharpness register 6 (image quality improver 0)	ADJ0_ENH_SHP6	32	H'FCFF76A8	32
	Enhancer LTI register 1 (image quality improver 0)	ADJ0_ENH_LTI1	32	H'FCFF76AC	32
	Enhancer LTI register 2 (image quality improver 0)	ADJ0_ENH_LTI2	32	H'FCFF76B0	32
	Matrix mode register in image quality improver (image quality improver 0)	ADJ0_MTX_MODE	32	H'FCFF76B4	32
	Matrix YG control register 0 in image quality improver (image quality improver 0)	ADJ0_MTX_YG_ADJ0	32	H'FCFF76B8	32
	Matrix YG control register 1 in image quality improver (image quality improver 0)	ADJ0_MTX_YG_ADJ1	32	H'FCFF76BC	32
	Matrix CBB control register 0 in image quality improver (image quality improver 0)	ADJ0_MTX_CBB_ADJ0	32	H'FCFF76C0	32
	Matrix CBB control register 1 in image quality improver (image quality improver 0)	ADJ0_MTX_CBB_ADJ1	32	H'FCFF76C4	32
	Matrix CRR control register 0 in image quality improver (image quality improver 0)	ADJ0_MTX_CRR_ADJ0	32	H'FCFF76C8	32
	Matrix CRR control register 1 in image quality improver (image quality improver 0)	ADJ0_MTX_CRR_ADJ1	32	H'FCFF76CC	32
	Register update control register in image quality improver (image quality improver 1)	ADJ1_UPDATE	32	H'FCFF7D80	32
	Black stretch register (image quality improver 1)	ADJ1_BKSTR_SET	32	H'FCFF7D84	32
	Enhancer timing adjustment register 1 (image quality improver 1)	ADJ1_ENH_TIM1	32	H'FCFF7D88	32
	Enhancer timing adjustment register 2 (image quality improver 1)	ADJ1_ENH_TIM2	32	H'FCFF7D8C	32
	Enhancer timing adjustment register 3 (image quality improver 1)	ADJ1_ENH_TIM3	32	H'FCFF7D90	32
	Enhancer sharpness register 1 (image quality improver 1)	ADJ1_ENH_SHP1	32	H'FCFF7D94	32
	Enhancer sharpness register 2 (image quality improver 1)	ADJ1_ENH_SHP2	32	H'FCFF7D98	32
	Enhancer sharpness register 3 (image quality improver 1)	ADJ1_ENH_SHP3	32	H'FCFF7D9C	32
	Enhancer sharpness register 4 (image quality improver 1)	ADJ1_ENH_SHP4	32	H'FCFF7DA0	32
	Enhancer sharpness register 5 (image quality improver 1)	ADJ1_ENH_SHP5	32	H'FCFF7DA4	32
	Enhancer sharpness register 6 (image quality improver 1)	ADJ1_ENH_SHP6	32	H'FCFF7DA8	32
	Enhancer LTI register 1 (image quality improver 1)	ADJ1_ENH_LTI1	32	H'FCFF7DAC	32
	Enhancer LTI register 2 (image quality improver 1)	ADJ1_ENH_LTI2	32	H'FCFF7DB0	32
	Matrix mode register in image quality improver (image quality improver 1)	ADJ1_MTX_MODE	32	H'FCFF7DB4	32
	Matrix YG control register 0 in image quality improver (image quality improver 1)	ADJ1_MTX_YG_ADJ0	32	H'FCFF7DB8	32
	Matrix YG control register 1 in image quality improver (image quality improver 1)	ADJ1_MTX_YG_ADJ1	32	H'FCFF7DBC	32
	Matrix CBB control register 0 in image quality improver (image quality improver 1)	ADJ1_MTX_CBB_ADJ0	32	H'FCFF7DC0	32
	Matrix CBB control register 1 in image quality improver (image quality improver 1)	ADJ1_MTX_CBB_ADJ1	32	H'FCFF7DC4	32
	Matrix CRR control register 0 in image quality improver (image quality improver 1)	ADJ1_MTX_CRR_ADJ0	32	H'FCFF7DC8	32
	Matrix CRR control register 1 in image quality improver (image quality improver 1)	ADJ1_MTX_CRR_ADJ1	32	H'FCFF7DCC	32
	Graphics 2 register update control register	GR2_UPDATE	32	H'FCFF7700	32
	Frame buffer read control register (Graphics 2)	GR2_FLM_RD	32	H'FCFF7704	32
	Frame buffer control register 1 (Graphics 2)	GR2_FLM1	32	H'FCFF7708	32
	Frame buffer control register 2 (Graphics 2)	GR2_FLM2	32	H'FCFF770C	32
	Frame buffer control register 3 (Graphics 2)	GR2_FLM3	32	H'FCFF7710	32
	Frame buffer control register 4 (Graphics 2)	GR2_FLM4	32	H'FCFF7714	32
	Frame buffer control register 5 (Graphics 2)	GR2_FLM5	32	H'FCFF7718	32
	Frame buffer control register 6 (Graphics 2)	GR2_FLM6	32	H'FCFF771C	32
	Alpha blending control register 1 (Graphics 2)	GR2_AB1	32	H'FCFF7720	32
	Alpha blending control register 2 (Graphics 2)	GR2_AB2	32	H'FCFF7724	32
	Alpha blending control register 3 (Graphics 2)	GR2_AB3	32	H'FCFF7728	32
	Alpha blending control register 4 (Graphics 2)	GR2_AB4	32	H'FCFF772C	32
	Alpha blending control register 5 (Graphics 2)	GR2_AB5	32	H'FCFF7730	32
	Alpha blending control register 6 (Graphics 2)	GR2_AB6	32	H'FCFF7734	32



**Table 57.1 Register Addresses**

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Video display controller 5 channel 0	Alpha blending control register 7 (Graphics 2)	GR2_AB7	32	H'FCFF7738	32
	Alpha blending control register 8 (Graphics 2)	GR2_AB8	32	H'FCFF773C	32
	Alpha blending control register 9 (Graphics 2)	GR2_AB9	32	H'FCFF7740	32
	Alpha blending control register 10 (Graphics 2)	GR2_AB10	32	H'FCFF7744	32
	Alpha blending control register 11 (Graphics 2)	GR2_AB11	32	H'FCFF7748	32
	Background color control register (Graphics 2)	GR2_BASE	32	H'FCFF774C	32
	CLUT table control register (Graphics 2)	GR2_CLUT	32	H'FCFF7750	32
	Status monitor register (Graphics 2)	GR2_MON	32	H'FCFF7754	32
	Graphics 3 register update control register	GR3_UPDATE	32	H'FCFF7780	32
	Frame buffer read control register (Graphics 3)	GR3_FLM_RD	32	H'FCFF7784	32
	Frame buffer control register 1 (Graphics 3)	GR3_FLM1	32	H'FCFF7788	32
	Frame buffer control register 2 (Graphics 3)	GR3_FLM2	32	H'FCFF778C	32
	Frame buffer control register 3 (Graphics 3)	GR3_FLM3	32	H'FCFF7790	32
	Frame buffer control register 4 (Graphics 3)	GR3_FLM4	32	H'FCFF7794	32
	Frame buffer control register 5 (Graphics 3)	GR3_FLM5	32	H'FCFF7798	32
	Frame buffer control register 6 (Graphics 3)	GR3_FLM6	32	H'FCFF779C	32
	Alpha blending control register 1 (Graphics 3)	GR3_AB1	32	H'FCFF77A0	32
	Alpha blending control register 2 (Graphics 3)	GR3_AB2	32	H'FCFF77A4	32
	Alpha blending control register 3 (Graphics 3)	GR3_AB3	32	H'FCFF77A8	32
	Alpha blending control register 4 (Graphics 3)	GR3_AB4	32	H'FCFF77AC	32
	Alpha blending control register 5 (Graphics 3)	GR3_AB5	32	H'FCFF77B0	32
	Alpha blending control register 6 (Graphics 3)	GR3_AB6	32	H'FCFF77B4	32
	Alpha blending control register 7 (Graphics 3)	GR3_AB7	32	H'FCFF77B8	32
	Alpha blending control register 8 (Graphics 3)	GR3_AB8	32	H'FCFF77BC	32
	Alpha blending control register 9 (Graphics 3)	GR3_AB9	32	H'FCFF77C0	32
	Alpha blending control register 10 (Graphics 3)	GR3_AB10	32	H'FCFF77C4	32
	Alpha blending control register 11 (Graphics 3)	GR3_AB11	32	H'FCFF77C8	32
	Background color control register (Graphics 3)	GR3_BASE	32	H'FCFF77CC	32
	CLUT table and interrupt control register (Graphics 3)	GR3_CLUT_INT	32	H'FCFF77D0	32
	Status monitor register (Graphics 3)	GR3_MON	32	H'FCFF77D4	32
	VIN synthesizer register update control register	GR_VIN_UPDATE	32	H'FCFF7E00	32
	Alpha blending control register 1 (VIN synthesizer)	GR_VIN_AB1	32	H'FCFF7E20	32
	Alpha blending control register 2 (VIN synthesizer)	GR_VIN_AB2	32	H'FCFF7E24	32
	Alpha blending control register 3 (VIN synthesizer)	GR_VIN_AB3	32	H'FCFF7E28	32
	Alpha blending control register 4 (VIN synthesizer)	GR_VIN_AB4	32	H'FCFF7E2C	32
	Alpha blending control register 5 (VIN synthesizer)	GR_VIN_AB5	32	H'FCFF7E30	32
	Alpha blending control register 6 (VIN synthesizer)	GR_VIN_AB6	32	H'FCFF7E34	32
	Alpha blending control register 7 (VIN synthesizer)	GR_VIN_AB7	32	H'FCFF7E38	32
	Background color control register (VIN synthesizer)	GR_VIN_BASE	32	H'FCFF7E4C	32
	Status monitor register (VIN synthesizer)	GR_VIN_MON	32	H'FCFF7E54	32
	SCL0 register update control register (OIR)	OIR_SCL0_UPDATE	32	H'FCFF7E80	32
	Mask control register (OIR)	OIR_SCL0_FRC1	32	H'FCFF7E84	32
	Missing Vsync compensation control register (OIR)	OIR_SCL0_FRC2	32	H'FCFF7E88	32
	Output sync select register (OIR)	OIR_SCL0_FRC3	32	H'FCFF7E8C	32
Free-running period control register (OIR)	OIR_SCL0_FRC4	32	H'FCFF7E90	32	
Output delay control register (OIR)	OIR_SCL0_FRC5	32	H'FCFF7E94	32	
Full-screen vertical size register (OIR)	OIR_SCL0_FRC6	32	H'FCFF7E98	32	
Full-screen horizontal size register (OIR)	OIR_SCL0_FRC7	32	H'FCFF7E9C	32	
Scaling-down control register (OIR)	OIR_SCL0_DS1	32	H'FCFF7EAC	32	
Vertical capture size register (OIR)	OIR_SCL0_DS2	32	H'FCFF7EB0	32	
Horizontal capture size register (OIR)	OIR_SCL0_DS3	32	H'FCFF7EB4	32	
Capture control block output size register (OIR)	OIR_SCL0_DS7	32	H'FCFF7EC4	32	

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Video display controller 5 channel 0	Scaling-up control register (OIR)	OIR_SCL0_US1	32	H'FCFF7EC8	32
	Output image vertical size register (OIR)	OIR_SCL0_US2	32	H'FCFF7ECC	32
	Output image horizontal size register (OIR)	OIR_SCL0_US3	32	H'FCFF7ED0	32
	Frame buffer read select register (OIR)	OIR_SCL0_US8	32	H'FCFF7EE4	32
	Background color register (OIR)	OIR_SCL0_OVR1	32	H'FCFF7EEC	32
	SCL1 register update control register (OIR)	OIR_SCL1_UPDATE	32	H'FCFF7F00	32
	Writing mode register (OIR)	OIR_SCL1_WR1	32	H'FCFF7F08	32
	Write address register 1 (OIR)	OIR_SCL1_WR2	32	H'FCFF7F0C	32
	Write address register 2 (OIR)	OIR_SCL1_WR3	32	H'FCFF7F10	32
	Write address register 3 (OIR)	OIR_SCL1_WR4	32	H'FCFF7F14	32
	Frame sub-sampling register (OIR)	OIR_SCL1_WR5	32	H'FCFF7F1C	32
	Bit reduction register (OIR)	OIR_SCL1_WR6	32	H'FCFF7F20	32
	Write detection register (OIR)	OIR_SCL1_WR7	32	H'FCFF7F24	32
	Graphics (OIR) register update control register	GR_OIR_UPDATE	32	H'FCFF7F80	32
	Frame buffer read control register (Graphics (OIR))	GR_OIR_FLM_RD	32	H'FCFF7F84	32
	Frame buffer control register 1 (Graphics (OIR))	GR_OIR_FLM1	32	H'FCFF7F88	32
	Frame buffer control register 2 (Graphics (OIR))	GR_OIR_FLM2	32	H'FCFF7F8C	32
	Frame buffer control register 3 (Graphics (OIR))	GR_OIR_FLM3	32	H'FCFF7F90	32
	Frame buffer control register 4 (Graphics (OIR))	GR_OIR_FLM4	32	H'FCFF7F94	32
	Frame buffer control register 5 (Graphics (OIR))	GR_OIR_FLM5	32	H'FCFF7F98	32
	Frame buffer control register 6 (Graphics (OIR))	GR_OIR_FLM6	32	H'FCFF7F9C	32
	Alpha blending control register 1 (Graphics (OIR))	GR_OIR_AB1	32	H'FCFF7FA0	32
	Alpha blending control register 2 (Graphics (OIR))	GR_OIR_AB2	32	H'FCFF7FA4	32
	Alpha blending control register 3 (Graphics (OIR))	GR_OIR_AB3	32	H'FCFF7FA8	32
	Alpha blending control register 7 (Graphics (OIR))	GR_OIR_AB7	32	H'FCFF7FB8	32
	Alpha blending control register 8 (Graphics (OIR))	GR_OIR_AB8	32	H'FCFF7FBC	32
	Alpha blending control register 9 (Graphics (OIR))	GR_OIR_AB9	32	H'FCFF7FC0	32
	Alpha blending control register 10 (Graphics (OIR))	GR_OIR_AB10	32	H'FCFF7FC4	32
	Alpha blending control register 11 (Graphics (OIR))	GR_OIR_AB11	32	H'FCFF7FC8	32
	Background color control register (Graphics (OIR))	GR_OIR_BASE	32	H'FCFF7FCC	32
	CLUT table control register (Graphics (OIR))	GR_OIR_CLUT	32	H'FCFF7FD0	32
	Status monitor register (Graphics (OIR))	GR_OIR_MON	32	H'FCFF7FD4	32
	Register update control register G in gamma correction block	GAM_G_UPDATE	32	H'FCFF7800	32
	Function switch register in gamma correction block	GAM_SW	32	H'FCFF7804	32
	Table setting register G1 in gamma correction block	GAM_G_LUT1	32	H'FCFF7808	32
	Table setting register G2 in gamma correction block	GAM_G_LUT2	32	H'FCFF780C	32
	Table setting register G3 in gamma correction block	GAM_G_LUT3	32	H'FCFF7810	32
	Table setting register G4 in gamma correction block	GAM_G_LUT4	32	H'FCFF7814	32
	Table setting register G5 in gamma correction block	GAM_G_LUT5	32	H'FCFF7818	32
	Table setting register G6 in gamma correction block	GAM_G_LUT6	32	H'FCFF781C	32
	Table setting register G7 in gamma correction block	GAM_G_LUT7	32	H'FCFF7820	32
	Table setting register G8 in gamma correction block	GAM_G_LUT8	32	H'FCFF7824	32
	Table setting register G9 in gamma correction block	GAM_G_LUT9	32	H'FCFF7828	32
	Table setting register G10 in gamma correction block	GAM_G_LUT10	32	H'FCFF782C	32
Table setting register G11 in gamma correction block	GAM_G_LUT11	32	H'FCFF7830	32	
Table setting register G12 in gamma correction block	GAM_G_LUT12	32	H'FCFF7834	32	
Table setting register G13 in gamma correction block	GAM_G_LUT13	32	H'FCFF7838	32	
Table setting register G14 in gamma correction block	GAM_G_LUT14	32	H'FCFF783C	32	
Table setting register G15 in gamma correction block	GAM_G_LUT15	32	H'FCFF7840	32	
Table setting register G16 in gamma correction block	GAM_G_LUT16	32	H'FCFF7844	32	
Area setting register G1 in gamma correction block	GAM_G_AREA1	32	H'FCFF7848	32	
Area setting register G2 in gamma correction block	GAM_G_AREA2	32	H'FCFF784C	32	

**Table 57.1 Register Addresses**

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Video display controller 5 channel 0	Area setting register G3 in gamma correction block	GAM_G_AREA3	32	H'FCFF7850	32
	Area setting register G4 in gamma correction block	GAM_G_AREA4	32	H'FCFF7854	32
	Area setting register G5 in gamma correction block	GAM_G_AREA5	32	H'FCFF7858	32
	Area setting register G6 in gamma correction block	GAM_G_AREA6	32	H'FCFF785C	32
	Area setting register G7 in gamma correction block	GAM_G_AREA7	32	H'FCFF7860	32
	Area setting register G8 in gamma correction block	GAM_G_AREA8	32	H'FCFF7864	32
	Register update control register B in gamma correction block	GAM_B_UPDATE	32	H'FCFF7880	32
	Table setting register B1 in gamma correction block	GAM_B_LUT1	32	H'FCFF7888	32
	Table setting register B2 in gamma correction block	GAM_B_LUT2	32	H'FCFF788C	32
	Table setting register B3 in gamma correction block	GAM_B_LUT3	32	H'FCFF7890	32
	Table setting register B4 in gamma correction block	GAM_B_LUT4	32	H'FCFF7894	32
	Table setting register B5 in gamma correction block	GAM_B_LUT5	32	H'FCFF7898	32
	Table setting register B6 in gamma correction block	GAM_B_LUT6	32	H'FCFF789C	32
	Table setting register B7 in gamma correction block	GAM_B_LUT7	32	H'FCFF78A0	32
	Table setting register B8 in gamma correction block	GAM_B_LUT8	32	H'FCFF78A4	32
	Table setting register B9 in gamma correction block	GAM_B_LUT9	32	H'FCFF78A8	32
	Table setting register B10 in gamma correction block	GAM_B_LUT10	32	H'FCFF78AC	32
	Table setting register B11 in gamma correction block	GAM_B_LUT11	32	H'FCFF78B0	32
	Table setting register B12 in gamma correction block	GAM_B_LUT12	32	H'FCFF78B4	32
	Table setting register B13 in gamma correction block	GAM_B_LUT13	32	H'FCFF78B8	32
	Table setting register B14 in gamma correction block	GAM_B_LUT14	32	H'FCFF78BC	32
	Table setting register B15 in gamma correction block	GAM_B_LUT15	32	H'FCFF78C0	32
	Table setting register B16 in gamma correction block	GAM_B_LUT16	32	H'FCFF78C4	32
	Area setting register B1 in gamma correction block	GAM_B_AREA1	32	H'FCFF78C8	32
	Area setting register B2 in gamma correction block	GAM_B_AREA2	32	H'FCFF78CC	32
	Area setting register B3 in gamma correction block	GAM_B_AREA3	32	H'FCFF78D0	32
	Area setting register B4 in gamma correction block	GAM_B_AREA4	32	H'FCFF78D4	32
	Area setting register B5 in gamma correction block	GAM_B_AREA5	32	H'FCFF78D8	32
	Area setting register B6 in gamma correction block	GAM_B_AREA6	32	H'FCFF78DC	32
	Area setting register B7 in gamma correction block	GAM_B_AREA7	32	H'FCFF78E0	32
	Area setting register B8 in gamma correction block	GAM_B_AREA8	32	H'FCFF78E4	32
	Register update control register R in gamma correction block	GAM_R_UPDATE	32	H'FCFF7900	32
	Table setting register R1 in gamma correction block	GAM_R_LUT1	32	H'FCFF7908	32
	Table setting register R2 in gamma correction block	GAM_R_LUT2	32	H'FCFF790C	32
	Table setting register R3 in gamma correction block	GAM_R_LUT3	32	H'FCFF7910	32
	Table setting register R4 in gamma correction block	GAM_R_LUT4	32	H'FCFF7914	32
	Table setting register R5 in gamma correction block	GAM_R_LUT5	32	H'FCFF7918	32
	Table setting register R6 in gamma correction block	GAM_R_LUT6	32	H'FCFF791C	32
	Table setting register R7 in gamma correction block	GAM_R_LUT7	32	H'FCFF7920	32
	Table setting register R8 in gamma correction block	GAM_R_LUT8	32	H'FCFF7924	32
	Table setting register R9 in gamma correction block	GAM_R_LUT9	32	H'FCFF7928	32
	Table setting register R10 in gamma correction block	GAM_R_LUT10	32	H'FCFF792C	32
	Table setting register R11 in gamma correction block	GAM_R_LUT11	32	H'FCFF7930	32
	Table setting register R12 in gamma correction block	GAM_R_LUT12	32	H'FCFF7934	32
	Table setting register R13 in gamma correction block	GAM_R_LUT13	32	H'FCFF7938	32
	Table setting register R14 in gamma correction block	GAM_R_LUT14	32	H'FCFF793C	32
	Table setting register R15 in gamma correction block	GAM_R_LUT15	32	H'FCFF7940	32
	Table setting register R16 in gamma correction block	GAM_R_LUT16	32	H'FCFF7944	32
	Area setting register R1 in gamma correction block	GAM_R_AREA1	32	H'FCFF7948	32
	Area setting register R2 in gamma correction block	GAM_R_AREA2	32	H'FCFF794C	32
	Area setting register R3 in gamma correction block	GAM_R_AREA3	32	H'FCFF7950	32
	Area setting register R4 in gamma correction block	GAM_R_AREA4	32	H'FCFF7954	32

**Table 57.1 Register Addresses**

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Video display controller 5 channel 0	Area setting register R5 in gamma correction block	GAM_R_AREA5	32	H'FCFF7958	32
	Area setting register R6 in gamma correction block	GAM_R_AREA6	32	H'FCFF795C	32
	Area setting register R7 in gamma correction block	GAM_R_AREA7	32	H'FCFF7960	32
	Area setting register R8 in gamma correction block	GAM_R_AREA8	32	H'FCFF7964	32
	TCON register update control register	TCON_UPDATE	32	H'FCFF7980	32
	TCON reference timing setting register	TCON_TIM	32	H'FCFF7984	32
	TCON vertical timing setting register A1	TCON_TIM_STVA1	32	H'FCFF7988	32
	TCON vertical timing setting register A2	TCON_TIM_STVA2	32	H'FCFF798C	32
	TCON vertical timing setting register B1	TCON_TIM_STVB1	32	H'FCFF7990	32
	TCON vertical timing setting register B2	TCON_TIM_STVB2	32	H'FCFF7994	32
	TCON horizontal timing setting register STH1	TCON_TIM_STH1	32	H'FCFF7998	32
	TCON horizontal timing setting register STH2	TCON_TIM_STH2	32	H'FCFF799C	32
	TCON horizontal timing setting register STB1	TCON_TIM_STB1	32	H'FCFF79A0	32
	TCON horizontal timing setting register STB2	TCON_TIM_STB2	32	H'FCFF79A4	32
	TCON horizontal timing setting register CPV1	TCON_TIM_CPV1	32	H'FCFF79A8	32
	TCON horizontal timing setting register CPV2	TCON_TIM_CPV2	32	H'FCFF79AC	32
	TCON horizontal timing setting register POLA1	TCON_TIM_POLA1	32	H'FCFF79B0	32
	TCON horizontal timing setting register POLA2	TCON_TIM_POLA2	32	H'FCFF79B4	32
	TCON horizontal timing setting register POLB1	TCON_TIM_POLB1	32	H'FCFF79B8	32
	TCON horizontal timing setting register POLB2	TCON_TIM_POLB2	32	H'FCFF79BC	32
	TCON data enable polarity setting register	TCON_TIM_DE	32	H'FCFF79C0	32
	Register update control register in output controller	OUT_UPDATE	32	H'FCFF7A00	32
	Output interface register	OUT_SET	32	H'FCFF7A04	32
	Brightness (DC) correction register 1	OUT_BRIGHT1	32	H'FCFF7A08	32
	Brightness (DC) correction register 2	OUT_BRIGHT2	32	H'FCFF7A0C	32
	Contrast (gain) correction register	OUT_CONTRAST	32	H'FCFF7A10	32
	Panel dither register	OUT_PDTHA	32	H'FCFF7A14	32
	Output phase control register	OUT_CLK_PHASE	32	H'FCFF7A24	32
	Interrupt control register 1	SYSCNT_INT1	32	H'FCFF7A80	32
	Interrupt control register 2	SYSCNT_INT2	32	H'FCFF7A84	32
	Interrupt control register 3	SYSCNT_INT3	32	H'FCFF7A88	32
	Interrupt control register 4	SYSCNT_INT4	32	H'FCFF7A8C	32
	Interrupt control register 5	SYSCNT_INT5	32	H'FCFF7A90	32
	Interrupt control register 6	SYSCNT_INT6	32	H'FCFF7A94	32
Panel clock control register	SYSCNT_PANEL_CLK	16	H'FCFF7A98	16	
CLUT table read select signal status register	SYSCNT_CLUT	16	H'FCFF7A9A	16	

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Video display controller 5 channel 1	External input block register update control register	INP_UPDATE	32	H'FCFF9400	32
	Input select control register	INP_SEL_CNT	32	H'FCFF9404	32
	External input sync signal control register	INP_EXT_SYNC_CNT	32	H'FCFF9408	32
	Vsync signal phase adjustment register	INP_VSYNC_PH_ADJ	32	H'FCFF940C	32
	Sync signal phase adjustment register	INP_DLY_ADJ	32	H'FCFF9410	32
	Image quality adjustment block register update control register	IMGCNT_UPDATE	32	H'FCFF9480	32
	NR control register 0	IMGCNT_NR_CNT0	32	H'FCFF9484	32
	NR control register 1	IMGCNT_NR_CNT1	32	H'FCFF9488	32
	Image quality adjustment block matrix mode register	IMGCNT_MTX_MODE	32	H'FCFF94A0	32
	Image quality adjustment block matrix YG adjustment register 0	IMGCNT_MTX_YG_ADJ0	32	H'FCFF94A4	32
	Image quality adjustment block matrix YG adjustment register 1	IMGCNT_MTX_YG_ADJ1	32	H'FCFF94A8	32
	Image quality adjustment block matrix CBB adjustment register 0	IMGCNT_MTX_CBB_ADJ0	32	H'FCFF94AC	32
	Image quality adjustment block matrix CBB adjustment register 1	IMGCNT_MTX_CBB_ADJ1	32	H'FCFF94B0	32
	Image quality adjustment block matrix CRR adjustment register 0	IMGCNT_MTX_CRR_ADJ0	32	H'FCFF94B4	32
	Image quality adjustment block matrix CRR adjustment register 1	IMGCNT_MTX_CRR_ADJ1	32	H'FCFF94B8	32
	Dynamic range compression register	IMGCNT_DRC_REG	32	H'FCFF94C0	32
	SCL0 register update control register (SC0)	SC0_SCL0_UPDATE	32	H'FCFF9500	32
	Mask control register (SC0)	SC0_SCL0_FRC1	32	H'FCFF9504	32
	Missing Vsync compensation control register (SC0)	SC0_SCL0_FRC2	32	H'FCFF9508	32
	Output sync select register (SC0)	SC0_SCL0_FRC3	32	H'FCFF950C	32
	Free-running period control register (SC0)	SC0_SCL0_FRC4	32	H'FCFF9510	32
	Output delay control register (SC0)	SC0_SCL0_FRC5	32	H'FCFF9514	32
	Full-screen vertical size register (SC0)	SC0_SCL0_FRC6	32	H'FCFF9518	32
	Full-screen horizontal size register (SC0)	SC0_SCL0_FRC7	32	H'FCFF951C	32
	Vsync detection register (SC0)	SC0_SCL0_FRC9	32	H'FCFF9524	32
	Status monitor 0 register (SC0)	SC0_SCL0_MON0	16	H'FCFF9528	16
	Interrupt control register (SC0)	SC0_SCL0_INT	16	H'FCFF952A	16
	Scaling-down control register (SC0)	SC0_SCL0_DS1	32	H'FCFF952C	32
	Vertical capture size register (SC0)	SC0_SCL0_DS2	32	H'FCFF9530	32
	Horizontal capture size register (SC0)	SC0_SCL0_DS3	32	H'FCFF9534	32
	Horizontal scale down register (SC0)	SC0_SCL0_DS4	32	H'FCFF9538	32
	Initial vertical phase register (SC0)	SC0_SCL0_DS5	32	H'FCFF953C	32
	Vertical scaling register (SC0)	SC0_SCL0_DS6	32	H'FCFF9540	32
	Scaling-down control block output size register (SC0)	SC0_SCL0_DS7	32	H'FCFF9544	32
	Scaling-up control register (SC0)	SC0_SCL0_US1	32	H'FCFF9548	32
	Output image vertical size register (SC0)	SC0_SCL0_US2	32	H'FCFF954C	32
	Output image horizontal size register (SC0)	SC0_SCL0_US3	32	H'FCFF9550	32
	Scaling-up control block input size register (SC0)	SC0_SCL0_US4	32	H'FCFF9554	32
	Horizontal scale up register (SC0)	SC0_SCL0_US5	32	H'FCFF9558	32
	Horizontal scale up initial phase register (SC0)	SC0_SCL0_US6	32	H'FCFF955C	32
	Trimming register (SC0)	SC0_SCL0_US7	32	H'FCFF9560	32
	Frame buffer read select register (SC0)	SC0_SCL0_US8	32	H'FCFF9564	32
	Background color register (SC0)	SC0_SCL0_OVR1	32	H'FCFF956C	32
	SCL1 register update control register (SC0)	SC0_SCL1_UPDATE	32	H'FCFF9580	32
Writing mode register (SC0)	SC0_SCL1_WR1	32	H'FCFF9588	32	
Write address register 1T (SC0)	SC0_SCL1_WR2	32	H'FCFF958C	32	
Write address register 2T (SC0)	SC0_SCL1_WR3	32	H'FCFF9590	32	
Write address register 3T (SC0)	SC0_SCL1_WR4	32	H'FCFF9594	32	
Frame sub-sampling register (SC0)	SC0_SCL1_WR5	32	H'FCFF959C	32	
Bit reduction register (SC0)	SC0_SCL1_WR6	32	H'FCFF95A0	32	
Write detection register (SC0)	SC0_SCL1_WR7	32	H'FCFF95A4	32	
Write address register 1B (SC0)	SC0_SCL1_WR8	32	H'FCFF95A8	32	

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Video display controller 5 channel 1	Write address register 2B (SC0)	SC0_SCL1_WR9	32	H'FCFF95AC	32
	Write address register 3B (SC0)	SC0_SCL1_WR10	32	H'FCFF95B0	32
	Write detection register B (SC0)	SC0_SCL1_WR11	32	H'FCFF95B4	32
	Status monitor 1 register (SC0)	SC0_SCL1_MON1	32	H'FCFF95B8	32
	Pointer buffer 0 register (SC0)	SC0_SCL1_PBUF0	32	H'FCFF95BC	32
	Pointer buffer 1 register (SC0)	SC0_SCL1_PBUF1	32	H'FCFF95C0	32
	Pointer buffer 2 register (SC0)	SC0_SCL1_PBUF2	32	H'FCFF95C4	32
	Pointer buffer 3 register (SC0)	SC0_SCL1_PBUF3	32	H'FCFF95C8	32
	Pointer buffer and field information register (SC0)	SC0_SCL1_PBUF_FLD	32	H'FCFF95CC	32
	Pointer buffer control register (SC0)	SC0_SCL1_PBUF_CNT	32	H'FCFF95D0	32
	Graphics 0 register update control register	GR0_UPDATE	32	H'FCFF9600	32
	Frame buffer read control register (graphics 0)	GR0_FLM_RD	32	H'FCFF9604	32
	Frame buffer control register 1 (graphics 0)	GR0_FLM1	32	H'FCFF9608	32
	Frame buffer control register 2 (graphics 0)	GR0_FLM2	32	H'FCFF960C	32
	Frame buffer control register 3 (graphics 0)	GR0_FLM3	32	H'FCFF9610	32
	Frame buffer control register 4 (graphics 0)	GR0_FLM4	32	H'FCFF9614	32
	Frame buffer control register 5 (graphics 0)	GR0_FLM5	32	H'FCFF9618	32
	Frame buffer control register 6 (graphics 0)	GR0_FLM6	32	H'FCFF961C	32
	Alpha blending control register 1 (graphics 0)	GR0_AB1	32	H'FCFF9620	32
	Alpha blending control register 2 (graphics 0)	GR0_AB2	32	H'FCFF9624	32
	Alpha blending control register 3 (graphics 0)	GR0_AB3	32	H'FCFF9628	32
	Alpha blending control register 7 (graphics 0)	GR0_AB7	32	H'FCFF9638	32
	Alpha blending control register 8 (graphics 0)	GR0_AB8	32	H'FCFF963C	32
	Alpha blending control register 9 (graphics 0)	GR0_AB9	32	H'FCFF9640	32
	Alpha blending control register 10 (graphics 0)	GR0_AB10	32	H'FCFF9644	32
	Alpha blending control register 11 (graphics 0)	GR0_AB11	32	H'FCFF9648	32
	Background color control register (graphics 0)	GR0_BASE	32	H'FCFF964C	32
	CLUT table control register (graphics 0)	GR0_CLUT	32	H'FCFF9650	32
	SCL0 register update control register (SC1)	SC1_SCL0_UPDATE	32	H'FCFF9C00	32
	Mask control register (SC1)	SC1_SCL0_FRC1	32	H'FCFF9C04	32
	Missing Vsync compensation control register (SC1)	SC1_SCL0_FRC2	32	H'FCFF9C08	32
	Output sync select register (SC1)	SC1_SCL0_FRC3	32	H'FCFF9C0C	32
	Free-running period control register (SC1)	SC1_SCL0_FRC4	32	H'FCFF9C10	32
	Output delay control register (SC1)	SC1_SCL0_FRC5	32	H'FCFF9C14	32
	Full-screen vertical size register (SC1)	SC1_SCL0_FRC6	32	H'FCFF9C18	32
	Full-screen horizontal size register (SC1)	SC1_SCL0_FRC7	32	H'FCFF9C1C	32
	Vsync detection register (SC1)	SC1_SCL0_FRC9	32	H'FCFF9C24	32
	Status monitor 0 register (SC1)	SC1_SCL0_MON0	16	H'FCFF9C28	16
	Interrupt control register (SC1)	SC1_SCL0_INT	16	H'FCFF9C2A	16
	Scaling-down control register (SC1)	SC1_SCL0_DS1	32	H'FCFF9C2C	32
	Vertical capture size register (SC1)	SC1_SCL0_DS2	32	H'FCFF9C30	32
	Horizontal capture size register (SC1)	SC1_SCL0_DS3	32	H'FCFF9C34	32
	Horizontal scale down register (SC1)	SC1_SCL0_DS4	32	H'FCFF9C38	32
	Initial vertical phase register (SC1)	SC1_SCL0_DS5	32	H'FCFF9C3C	32
	Vertical scaling register (SC1)	SC1_SCL0_DS6	32	H'FCFF9C40	32
	Scaling-down control block output size register (SC1)	SC1_SCL0_DS7	32	H'FCFF9C44	32
	Scaling-up control register (SC1)	SC1_SCL0_US1	32	H'FCFF9C48	32
	Output image vertical size register (SC1)	SC1_SCL0_US2	32	H'FCFF9C4C	32
	Output image horizontal size register (SC1)	SC1_SCL0_US3	32	H'FCFF9C50	32
	Scaling-up control block input size register (SC1)	SC1_SCL0_US4	32	H'FCFF9C54	32
	Horizontal scale up register (SC1)	SC1_SCL0_US5	32	H'FCFF9C58	32
	Horizontal scale up initial phase register (SC1)	SC1_SCL0_US6	32	H'FCFF9C5C	32

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Video display controller 5 channel 1	Trimming register (SC1)	SC1_SCL0_US7	32	H'FCFF9C60	32
	Frame buffer read select register (SC1)	SC1_SCL0_US8	32	H'FCFF9C64	32
	Background color register (SC1)	SC1_SCL0_OVR1	32	H'FCFF9C6C	32
	SCL1 register update control register (SC1)	SC1_SCL1_UPDATE	32	H'FCFF9C80	32
	Writing mode register (SC1)	SC1_SCL1_WR1	32	H'FCFF9C88	32
	Write address register 1T (SC1)	SC1_SCL1_WR2	32	H'FCFF9C8C	32
	Write address register 2T (SC1)	SC1_SCL1_WR3	32	H'FCFF9C90	32
	Write address register 3T (SC1)	SC1_SCL1_WR4	32	H'FCFF9C94	32
	Frame sub-sampling register (SC1)	SC1_SCL1_WR5	32	H'FCFF9C9C	32
	Bit reduction register (SC1)	SC1_SCL1_WR6	32	H'FCFF9CA0	32
	Write detection register (SC1)	SC1_SCL1_WR7	32	H'FCFF9CA4	32
	Write address register 1B (SC1)	SC1_SCL1_WR8	32	H'FCFF9CA8	32
	Write address register 2B (SC1)	SC1_SCL1_WR9	32	H'FCFF9CAC	32
	Write address register 3B (SC1)	SC1_SCL1_WR10	32	H'FCFF9CB0	32
	Write detection register B (SC1)	SC1_SCL1_WR11	32	H'FCFF9CB4	32
	Status monitor 1 register (SC1)	SC1_SCL1_MON1	32	H'FCFF9CB8	32
	Pointer buffer 0 register (SC1)	SC1_SCL1_PBUF0	32	H'FCFF9CBC	32
	Pointer buffer 1 register (SC1)	SC1_SCL1_PBUF1	32	H'FCFF9CC0	32
	Pointer buffer 2 register (SC1)	SC1_SCL1_PBUF2	32	H'FCFF9CC4	32
	Pointer buffer 3 register (SC1)	SC1_SCL1_PBUF3	32	H'FCFF9CC8	32
	Pointer buffer and field information register (SC1)	SC1_SCL1_PBUF_FLD	32	H'FCFF9CCC	32
	Pointer buffer control register (SC1)	SC1_SCL1_PBUF_CNT	32	H'FCFF9CD0	32
	Graphics 1 register update control register	GR1_UPDATE	32	H'FCFF9D00	32
	Frame buffer read control register (graphics 1)	GR1_FLM_RD	32	H'FCFF9D04	32
	Frame buffer control register 1 (graphics 1)	GR1_FLM1	32	H'FCFF9D08	32
	Frame buffer control register 2 (graphics 1)	GR1_FLM2	32	H'FCFF9D0C	32
	Frame buffer control register 3 (graphics 1)	GR1_FLM3	32	H'FCFF9D10	32
	Frame buffer control register 4 (graphics 1)	GR1_FLM4	32	H'FCFF9D14	32
	Frame buffer control register 5 (graphics 1)	GR1_FLM5	32	H'FCFF9D18	32
	Frame buffer control register 6 (graphics 1)	GR1_FLM6	32	H'FCFF9D1C	32
	Alpha blending control register 1 (graphics 1)	GR1_AB1	32	H'FCFF9D20	32
	Alpha blending control register 2 (graphics 1)	GR1_AB2	32	H'FCFF9D24	32
	Alpha blending control register 3 (graphics 1)	GR1_AB3	32	H'FCFF9D28	32
	Alpha blending control register 4 (graphics 1)	GR1_AB4	32	H'FCFF9D2C	32
	Alpha blending control register 5 (graphics 1)	GR1_AB5	32	H'FCFF9D30	32
	Alpha blending control register 6 (graphics 1)	GR1_AB6	32	H'FCFF9D34	32
	Alpha blending control register 7 (graphics 1)	GR1_AB7	32	H'FCFF9D38	32
	Alpha blending control register 8 (graphics 1)	GR1_AB8	32	H'FCFF9D3C	32
	Alpha blending control register 9 (graphics 1)	GR1_AB9	32	H'FCFF9D40	32
	Alpha blending control register 10 (graphics 1)	GR1_AB10	32	H'FCFF9D44	32
	Alpha blending control register 11 (graphics 1)	GR1_AB11	32	H'FCFF9D48	32
	Background color control register (graphics 1)	GR1_BASE	32	H'FCFF9D4C	32
	CLUT table control register (graphics 1)	GR1_CLUT	32	H'FCFF9D50	32
	Status monitor register (graphics 1)	GR1_MON	32	H'FCFF9D54	32
	Register update control register in image quality improver (image quality improver 0)	ADJ0_UPDATE	32	H'FCFF9680	32
	Black stretch register (image quality improver 0)	ADJ0_BKSTR_SET	32	H'FCFF9684	32
	Enhancer timing adjustment register 1 (image quality improver 0)	ADJ0_ENH_TIM1	32	H'FCFF9688	32
	Enhancer timing adjustment register 2 (image quality improver 0)	ADJ0_ENH_TIM2	32	H'FCFF968C	32
	Enhancer timing adjustment register 3 (image quality improver 0)	ADJ0_ENH_TIM3	32	H'FCFF9690	32
	Enhancer sharpness register 1 (image quality improver 0)	ADJ0_ENH_SHP1	32	H'FCFF9694	32
	Enhancer sharpness register 2 (image quality improver 0)	ADJ0_ENH_SHP2	32	H'FCFF9698	32
	Enhancer sharpness register 3 (image quality improver 0)	ADJ0_ENH_SHP3	32	H'FCFF969C	32

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Video display controller 5 channel 1	Enhancer sharpness register 4 (image quality improver 0)	ADJ0_ENH_SHP4	32	H'FCFF96A0	32
	Enhancer sharpness register 5 (image quality improver 0)	ADJ0_ENH_SHP5	32	H'FCFF96A4	32
	Enhancer sharpness register 6 (image quality improver 0)	ADJ0_ENH_SHP6	32	H'FCFF96A8	32
	Enhancer LTI register 1 (image quality improver 0)	ADJ0_ENH_LTI1	32	H'FCFF96AC	32
	Enhancer LTI register 2 (image quality improver 0)	ADJ0_ENH_LTI2	32	H'FCFF96B0	32
	Matrix mode register in image quality improver (image quality improver 0)	ADJ0_MTX_MODE	32	H'FCFF96B4	32
	Matrix YG control register 0 in image quality improver (image quality improver 0)	ADJ0_MTX_YG_ADJ0	32	H'FCFF96B8	32
	Matrix YG control register 1 in image quality improver (image quality improver 0)	ADJ0_MTX_YG_ADJ1	32	H'FCFF96BC	32
	Matrix CBB control register 0 in image quality improver (image quality improver 0)	ADJ0_MTX_CBB_ADJ0	32	H'FCFF96C0	32
	Matrix CBB control register 1 in image quality improver (image quality improver 0)	ADJ0_MTX_CBB_ADJ1	32	H'FCFF96C4	32
	Matrix CRR control register 0 in image quality improver (image quality improver 0)	ADJ0_MTX_CRR_ADJ0	32	H'FCFF96C8	32
	Matrix CRR control register 1 in image quality improver (image quality improver 0)	ADJ0_MTX_CRR_ADJ1	32	H'FCFF96CC	32
	Register update control register in image quality improver (image quality improver 1)	ADJ1_UPDATE	32	H'FCFF9D80	32
	Black stretch register (image quality improver 1)	ADJ1_BKSTR_SET	32	H'FCFF9D84	32
	Enhancer timing adjustment register 1 (image quality improver 1)	ADJ1_ENH_TIM1	32	H'FCFF9D88	32
	Enhancer timing adjustment register 2 (image quality improver 1)	ADJ1_ENH_TIM2	32	H'FCFF9D8C	32
	Enhancer timing adjustment register 3 (image quality improver 1)	ADJ1_ENH_TIM3	32	H'FCFF9D90	32
	Enhancer sharpness register 1 (image quality improver 1)	ADJ1_ENH_SHP1	32	H'FCFF9D94	32
	Enhancer sharpness register 2 (image quality improver 1)	ADJ1_ENH_SHP2	32	H'FCFF9D98	32
	Enhancer sharpness register 3 (image quality improver 1)	ADJ1_ENH_SHP3	32	H'FCFF9D9C	32
	Enhancer sharpness register 4 (image quality improver 1)	ADJ1_ENH_SHP4	32	H'FCFF9DA0	32
	Enhancer sharpness register 5 (image quality improver 1)	ADJ1_ENH_SHP5	32	H'FCFF9DA4	32
	Enhancer sharpness register 6 (image quality improver 1)	ADJ1_ENH_SHP6	32	H'FCFF9DA8	32
	Enhancer LTI register 1 (image quality improver 1)	ADJ1_ENH_LTI1	32	H'FCFF9DAC	32
	Enhancer LTI register 2 (image quality improver 1)	ADJ1_ENH_LTI2	32	H'FCFF9DB0	32
	Matrix mode register in image quality improver (image quality improver 1)	ADJ1_MTX_MODE	32	H'FCFF9DB4	32
	Matrix YG control register 0 in image quality improver (image quality improver 1)	ADJ1_MTX_YG_ADJ0	32	H'FCFF9DB8	32
	Matrix YG control register 1 in image quality improver (image quality improver 1)	ADJ1_MTX_YG_ADJ1	32	H'FCFF9DBC	32
	Matrix CBB control register 0 in image quality improver (image quality improver 1)	ADJ1_MTX_CBB_ADJ0	32	H'FCFF9DC0	32
	Matrix CBB control register 1 in image quality improver (image quality improver 1)	ADJ1_MTX_CBB_ADJ1	32	H'FCFF9DC4	32
	Matrix CRR control register 0 in image quality improver (image quality improver 1)	ADJ1_MTX_CRR_ADJ0	32	H'FCFF9DC8	32
	Matrix CRR control register 1 in image quality improver (image quality improver 1)	ADJ1_MTX_CRR_ADJ1	32	H'FCFF9DCC	32
	Graphics 2 register update control register	GR2_UPDATE	32	H'FCFF9700	32
	Frame buffer read control register (Graphics 2)	GR2_FLM_RD	32	H'FCFF9704	32
	Frame buffer control register 1 (Graphics 2)	GR2_FLM1	32	H'FCFF9708	32
	Frame buffer control register 2 (Graphics 2)	GR2_FLM2	32	H'FCFF970C	32
	Frame buffer control register 3 (Graphics 2)	GR2_FLM3	32	H'FCFF9710	32
	Frame buffer control register 4 (Graphics 2)	GR2_FLM4	32	H'FCFF9714	32
	Frame buffer control register 5 (Graphics 2)	GR2_FLM5	32	H'FCFF9718	32
	Frame buffer control register 6 (Graphics 2)	GR2_FLM6	32	H'FCFF971C	32
	Alpha blending control register 1 (Graphics 2)	GR2_AB1	32	H'FCFF9720	32
	Alpha blending control register 2 (Graphics 2)	GR2_AB2	32	H'FCFF9724	32
	Alpha blending control register 3 (Graphics 2)	GR2_AB3	32	H'FCFF9728	32
	Alpha blending control register 4 (Graphics 2)	GR2_AB4	32	H'FCFF972C	32
Alpha blending control register 5 (Graphics 2)	GR2_AB5	32	H'FCFF9730	32	
Alpha blending control register 6 (Graphics 2)	GR2_AB6	32	H'FCFF9734	32	



Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Video display controller 5 channel 1	Alpha blending control register 7 (Graphics 2)	GR2_AB7	32	H'FCFF9738	32
	Alpha blending control register 8 (Graphics 2)	GR2_AB8	32	H'FCFF973C	32
	Alpha blending control register 9 (Graphics 2)	GR2_AB9	32	H'FCFF9740	32
	Alpha blending control register 10 (Graphics 2)	GR2_AB10	32	H'FCFF9744	32
	Alpha blending control register 11 (Graphics 2)	GR2_AB11	32	H'FCFF9748	32
	Background color control register (Graphics 2)	GR2_BASE	32	H'FCFF974C	32
	CLUT table control register (Graphics 2)	GR2_CLUT	32	H'FCFF9750	32
	Status monitor register (Graphics 2)	GR2_MON	32	H'FCFF9754	32
	Graphics 3 register update control register	GR3_UPDATE	32	H'FCFF9780	32
	Frame buffer read control register (Graphics 3)	GR3_FLM_RD	32	H'FCFF9784	32
	Frame buffer control register 1 (Graphics 3)	GR3_FLM1	32	H'FCFF9788	32
	Frame buffer control register 2 (Graphics 3)	GR3_FLM2	32	H'FCFF978C	32
	Frame buffer control register 3 (Graphics 3)	GR3_FLM3	32	H'FCFF9790	32
	Frame buffer control register 4 (Graphics 3)	GR3_FLM4	32	H'FCFF9794	32
	Frame buffer control register 5 (Graphics 3)	GR3_FLM5	32	H'FCFF9798	32
	Frame buffer control register 6 (Graphics 3)	GR3_FLM6	32	H'FCFF979C	32
	Alpha blending control register 1 (Graphics 3)	GR3_AB1	32	H'FCFF97A0	32
	Alpha blending control register 2 (Graphics 3)	GR3_AB2	32	H'FCFF97A4	32
	Alpha blending control register 3 (Graphics 3)	GR3_AB3	32	H'FCFF97A8	32
	Alpha blending control register 4 (Graphics 3)	GR3_AB4	32	H'FCFF97AC	32
	Alpha blending control register 5 (Graphics 3)	GR3_AB5	32	H'FCFF97B0	32
	Alpha blending control register 6 (Graphics 3)	GR3_AB6	32	H'FCFF97B4	32
	Alpha blending control register 7 (Graphics 3)	GR3_AB7	32	H'FCFF97B8	32
	Alpha blending control register 8 (Graphics 3)	GR3_AB8	32	H'FCFF97BC	32
	Alpha blending control register 9 (Graphics 3)	GR3_AB9	32	H'FCFF97C0	32
	Alpha blending control register 10 (Graphics 3)	GR3_AB10	32	H'FCFF97C4	32
	Alpha blending control register 11 (Graphics 3)	GR3_AB11	32	H'FCFF97C8	32
	Background color control register (Graphics 3)	GR3_BASE	32	H'FCFF97CC	32
	CLUT table and interrupt control register (Graphics 3)	GR3_CLUT_INT	32	H'FCFF97D0	32
	Status monitor register (Graphics 3)	GR3_MON	32	H'FCFF97D4	32
	VIN synthesizer register update control register	GR_VIN_UPDATE	32	H'FCFF9E00	32
	Alpha blending control register 1 (VIN synthesizer)	GR_VIN_AB1	32	H'FCFF9E20	32
	Alpha blending control register 2 (VIN synthesizer)	GR_VIN_AB2	32	H'FCFF9E24	32
	Alpha blending control register 3 (VIN synthesizer)	GR_VIN_AB3	32	H'FCFF9E28	32
	Alpha blending control register 4 (VIN synthesizer)	GR_VIN_AB4	32	H'FCFF9E2C	32
	Alpha blending control register 5 (VIN synthesizer)	GR_VIN_AB5	32	H'FCFF9E30	32
	Alpha blending control register 6 (VIN synthesizer)	GR_VIN_AB6	32	H'FCFF9E34	32
	Alpha blending control register 7 (VIN synthesizer)	GR_VIN_AB7	32	H'FCFF9E38	32
	Background color control register (VIN synthesizer)	GR_VIN_BASE	32	H'FCFF9E4C	32
	Status monitor register (VIN synthesizer)	GR_VIN_MON	32	H'FCFF9E54	32
	SCL0 register update control register (OIR)	OIR_SCL0_UPDATE	32	H'FCFF9E80	32
	Mask control register (OIR)	OIR_SCL0_FRC1	32	H'FCFF9E84	32
	Missing Vsync compensation control register (OIR)	OIR_SCL0_FRC2	32	H'FCFF9E88	32
	Output sync select register (OIR)	OIR_SCL0_FRC3	32	H'FCFF9E8C	32
Free-running period control register (OIR)	OIR_SCL0_FRC4	32	H'FCFF9E90	32	
Output delay control register (OIR)	OIR_SCL0_FRC5	32	H'FCFF9E94	32	
Full-screen vertical size register (OIR)	OIR_SCL0_FRC6	32	H'FCFF9E98	32	
Full-screen horizontal size register (OIR)	OIR_SCL0_FRC7	32	H'FCFF9E9C	32	
Scaling-down control register (OIR)	OIR_SCL0_DS1	32	H'FCFF9EAC	32	
Vertical capture size register (OIR)	OIR_SCL0_DS2	32	H'FCFF9EB0	32	
Horizontal capture size register (OIR)	OIR_SCL0_DS3	32	H'FCFF9EB4	32	
Capture control block output size register (OIR)	OIR_SCL0_DS7	32	H'FCFF9EC4	32	

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Video display controller 5 channel 1	Scaling-up control register (OIR)	OIR_SCL0_US1	32	H'FCFF9EC8	32
	Output image vertical size register (OIR)	OIR_SCL0_US2	32	H'FCFF9ECC	32
	Output image horizontal size register (OIR)	OIR_SCL0_US3	32	H'FCFF9ED0	32
	Frame buffer read select register (OIR)	OIR_SCL0_US8	32	H'FCFF9EE4	32
	Background color register (OIR)	OIR_SCL0_OVR1	32	H'FCFF9EEC	32
	SCL1 register update control register (OIR)	OIR_SCL1_UPDATE	32	H'FCFF9F00	32
	Writing mode register (OIR)	OIR_SCL1_WR1	32	H'FCFF9F08	32
	Write address register 1 (OIR)	OIR_SCL1_WR2	32	H'FCFF9F0C	32
	Write address register 2 (OIR)	OIR_SCL1_WR3	32	H'FCFF9F10	32
	Write address register 3 (OIR)	OIR_SCL1_WR4	32	H'FCFF9F14	32
	Frame sub-sampling register (OIR)	OIR_SCL1_WR5	32	H'FCFF9F1C	32
	Bit reduction register (OIR)	OIR_SCL1_WR6	32	H'FCFF9F20	32
	Write detection register (OIR)	OIR_SCL1_WR7	32	H'FCFF9F24	32
	Graphics (OIR) register update control register	GR_OIR_UPDATE	32	H'FCFF9F80	32
	Frame buffer read control register (Graphics (OIR))	GR_OIR_FLM_RD	32	H'FCFF9F84	32
	Frame buffer control register 1 (Graphics (OIR))	GR_OIR_FLM1	32	H'FCFF9F88	32
	Frame buffer control register 2 (Graphics (OIR))	GR_OIR_FLM2	32	H'FCFF9F8C	32
	Frame buffer control register 3 (Graphics (OIR))	GR_OIR_FLM3	32	H'FCFF9F90	32
	Frame buffer control register 4 (Graphics (OIR))	GR_OIR_FLM4	32	H'FCFF9F94	32
	Frame buffer control register 5 (Graphics (OIR))	GR_OIR_FLM5	32	H'FCFF9F98	32
	Frame buffer control register 6 (Graphics (OIR))	GR_OIR_FLM6	32	H'FCFF9F9C	32
	Alpha blending control register 1 (Graphics (OIR))	GR_OIR_AB1	32	H'FCFF9FA0	32
	Alpha blending control register 2 (Graphics (OIR))	GR_OIR_AB2	32	H'FCFF9FA4	32
	Alpha blending control register 3 (Graphics (OIR))	GR_OIR_AB3	32	H'FCFF9FA8	32
	Alpha blending control register 7 (Graphics (OIR))	GR_OIR_AB7	32	H'FCFF9FB8	32
	Alpha blending control register 8 (Graphics (OIR))	GR_OIR_AB8	32	H'FCFF9FBC	32
	Alpha blending control register 9 (Graphics (OIR))	GR_OIR_AB9	32	H'FCFF9FC0	32
	Alpha blending control register 10 (Graphics (OIR))	GR_OIR_AB10	32	H'FCFF9FC4	32
	Alpha blending control register 11 (Graphics (OIR))	GR_OIR_AB11	32	H'FCFF9FC8	32
	Background color control register (Graphics (OIR))	GR_OIR_BASE	32	H'FCFF9FCC	32
	CLUT table control register (Graphics (OIR))	GR_OIR_CLUT	32	H'FCFF9FD0	32
	Status monitor register (Graphics (OIR))	GR_OIR_MON	32	H'FCFF9FD4	32
	Register update control register G in gamma correction block	GAM_G_UPDATE	32	H'FCFF9800	32
	Function switch register in gamma correction block	GAM_SW	32	H'FCFF9804	32
	Table setting register G1 in gamma correction block	GAM_G_LUT1	32	H'FCFF9808	32
	Table setting register G2 in gamma correction block	GAM_G_LUT2	32	H'FCFF980C	32
	Table setting register G3 in gamma correction block	GAM_G_LUT3	32	H'FCFF9810	32
	Table setting register G4 in gamma correction block	GAM_G_LUT4	32	H'FCFF9814	32
	Table setting register G5 in gamma correction block	GAM_G_LUT5	32	H'FCFF9818	32
	Table setting register G6 in gamma correction block	GAM_G_LUT6	32	H'FCFF981C	32
	Table setting register G7 in gamma correction block	GAM_G_LUT7	32	H'FCFF9820	32
	Table setting register G8 in gamma correction block	GAM_G_LUT8	32	H'FCFF9824	32
	Table setting register G9 in gamma correction block	GAM_G_LUT9	32	H'FCFF9828	32
	Table setting register G10 in gamma correction block	GAM_G_LUT10	32	H'FCFF982C	32
	Table setting register G11 in gamma correction block	GAM_G_LUT11	32	H'FCFF9830	32
	Table setting register G12 in gamma correction block	GAM_G_LUT12	32	H'FCFF9834	32
Table setting register G13 in gamma correction block	GAM_G_LUT13	32	H'FCFF9838	32	
Table setting register G14 in gamma correction block	GAM_G_LUT14	32	H'FCFF983C	32	
Table setting register G15 in gamma correction block	GAM_G_LUT15	32	H'FCFF9840	32	
Table setting register G16 in gamma correction block	GAM_G_LUT16	32	H'FCFF9844	32	
Area setting register G1 in gamma correction block	GAM_G_AREA1	32	H'FCFF9848	32	
Area setting register G2 in gamma correction block	GAM_G_AREA2	32	H'FCFF984C	32	

**Table 57.1 Register Addresses**

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Video display controller 5 channel 1	Area setting register G3 in gamma correction block	GAM_G_AREA3	32	H'FCFF9850	32
	Area setting register G4 in gamma correction block	GAM_G_AREA4	32	H'FCFF9854	32
	Area setting register G5 in gamma correction block	GAM_G_AREA5	32	H'FCFF9858	32
	Area setting register G6 in gamma correction block	GAM_G_AREA6	32	H'FCFF985C	32
	Area setting register G7 in gamma correction block	GAM_G_AREA7	32	H'FCFF9860	32
	Area setting register G8 in gamma correction block	GAM_G_AREA8	32	H'FCFF9864	32
	Register update control register B in gamma correction block	GAM_B_UPDATE	32	H'FCFF9880	32
	Table setting register B1 in gamma correction block	GAM_B_LUT1	32	H'FCFF9888	32
	Table setting register B2 in gamma correction block	GAM_B_LUT2	32	H'FCFF988C	32
	Table setting register B3 in gamma correction block	GAM_B_LUT3	32	H'FCFF9890	32
	Table setting register B4 in gamma correction block	GAM_B_LUT4	32	H'FCFF9894	32
	Table setting register B5 in gamma correction block	GAM_B_LUT5	32	H'FCFF9898	32
	Table setting register B6 in gamma correction block	GAM_B_LUT6	32	H'FCFF989C	32
	Table setting register B7 in gamma correction block	GAM_B_LUT7	32	H'FCFF98A0	32
	Table setting register B8 in gamma correction block	GAM_B_LUT8	32	H'FCFF98A4	32
	Table setting register B9 in gamma correction block	GAM_B_LUT9	32	H'FCFF98A8	32
	Table setting register B10 in gamma correction block	GAM_B_LUT10	32	H'FCFF98AC	32
	Table setting register B11 in gamma correction block	GAM_B_LUT11	32	H'FCFF98B0	32
	Table setting register B12 in gamma correction block	GAM_B_LUT12	32	H'FCFF98B4	32
	Table setting register B13 in gamma correction block	GAM_B_LUT13	32	H'FCFF98B8	32
	Table setting register B14 in gamma correction block	GAM_B_LUT14	32	H'FCFF98BC	32
	Table setting register B15 in gamma correction block	GAM_B_LUT15	32	H'FCFF98C0	32
	Table setting register B16 in gamma correction block	GAM_B_LUT16	32	H'FCFF98C4	32
	Area setting register B1 in gamma correction block	GAM_B_AREA1	32	H'FCFF98C8	32
	Area setting register B2 in gamma correction block	GAM_B_AREA2	32	H'FCFF98CC	32
	Area setting register B3 in gamma correction block	GAM_B_AREA3	32	H'FCFF98D0	32
	Area setting register B4 in gamma correction block	GAM_B_AREA4	32	H'FCFF98D4	32
	Area setting register B5 in gamma correction block	GAM_B_AREA5	32	H'FCFF98D8	32
	Area setting register B6 in gamma correction block	GAM_B_AREA6	32	H'FCFF98DC	32
	Area setting register B7 in gamma correction block	GAM_B_AREA7	32	H'FCFF98E0	32
	Area setting register B8 in gamma correction block	GAM_B_AREA8	32	H'FCFF98E4	32
	Register update control register R in gamma correction block	GAM_R_UPDATE	32	H'FCFF9900	32
	Table setting register R1 in gamma correction block	GAM_R_LUT1	32	H'FCFF9908	32
	Table setting register R2 in gamma correction block	GAM_R_LUT2	32	H'FCFF990C	32
	Table setting register R3 in gamma correction block	GAM_R_LUT3	32	H'FCFF9910	32
	Table setting register R4 in gamma correction block	GAM_R_LUT4	32	H'FCFF9914	32
	Table setting register R5 in gamma correction block	GAM_R_LUT5	32	H'FCFF9918	32
	Table setting register R6 in gamma correction block	GAM_R_LUT6	32	H'FCFF991C	32
	Table setting register R7 in gamma correction block	GAM_R_LUT7	32	H'FCFF9920	32
	Table setting register R8 in gamma correction block	GAM_R_LUT8	32	H'FCFF9924	32
	Table setting register R9 in gamma correction block	GAM_R_LUT9	32	H'FCFF9928	32
	Table setting register R10 in gamma correction block	GAM_R_LUT10	32	H'FCFF992C	32
	Table setting register R11 in gamma correction block	GAM_R_LUT11	32	H'FCFF9930	32
	Table setting register R12 in gamma correction block	GAM_R_LUT12	32	H'FCFF9934	32
	Table setting register R13 in gamma correction block	GAM_R_LUT13	32	H'FCFF9938	32
	Table setting register R14 in gamma correction block	GAM_R_LUT14	32	H'FCFF993C	32
	Table setting register R15 in gamma correction block	GAM_R_LUT15	32	H'FCFF9940	32
	Table setting register R16 in gamma correction block	GAM_R_LUT16	32	H'FCFF9944	32
	Area setting register R1 in gamma correction block	GAM_R_AREA1	32	H'FCFF9948	32
	Area setting register R2 in gamma correction block	GAM_R_AREA2	32	H'FCFF994C	32
	Area setting register R3 in gamma correction block	GAM_R_AREA3	32	H'FCFF9950	32
	Area setting register R4 in gamma correction block	GAM_R_AREA4	32	H'FCFF9954	32

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Video display controller 5 channel 1	Area setting register R5 in gamma correction block	GAM_R_AREA5	32	H'FCFF9958	32
	Area setting register R6 in gamma correction block	GAM_R_AREA6	32	H'FCFF995C	32
	Area setting register R7 in gamma correction block	GAM_R_AREA7	32	H'FCFF9960	32
	Area setting register R8 in gamma correction block	GAM_R_AREA8	32	H'FCFF9964	32
	TCON register update control register	TCON_UPDATE	32	H'FCFF9980	32
	TCON reference timing setting register	TCON_TIM	32	H'FCFF9984	32
	TCON vertical timing setting register A1	TCON_TIM_STVA1	32	H'FCFF9988	32
	TCON vertical timing setting register A2	TCON_TIM_STVA2	32	H'FCFF998C	32
	TCON vertical timing setting register B1	TCON_TIM_STVB1	32	H'FCFF9990	32
	TCON vertical timing setting register B2	TCON_TIM_STVB2	32	H'FCFF9994	32
	TCON horizontal timing setting register STH1	TCON_TIM_STH1	32	H'FCFF9998	32
	TCON horizontal timing setting register STH2	TCON_TIM_STH2	32	H'FCFF999C	32
	TCON horizontal timing setting register STB1	TCON_TIM_STB1	32	H'FCFF99A0	32
	TCON horizontal timing setting register STB2	TCON_TIM_STB2	32	H'FCFF99A4	32
	TCON horizontal timing setting register CPV1	TCON_TIM_CPV1	32	H'FCFF99A8	32
	TCON horizontal timing setting register CPV2	TCON_TIM_CPV2	32	H'FCFF99AC	32
	TCON horizontal timing setting register POLA1	TCON_TIM_POLA1	32	H'FCFF99B0	32
	TCON horizontal timing setting register POLA2	TCON_TIM_POLA2	32	H'FCFF99B4	32
	TCON horizontal timing setting register POLB1	TCON_TIM_POLB1	32	H'FCFF99B8	32
	TCON horizontal timing setting register POLB2	TCON_TIM_POLB2	32	H'FCFF99BC	32
	TCON data enable polarity setting register	TCON_TIM_DE	32	H'FCFF99C0	32
	Register update control register in output controller	OUT_UPDATE	32	H'FCFF9A00	32
	Output interface register	OUT_SET	32	H'FCFF9A04	32
	Brightness (DC) correction register 1	OUT_BRIGHT1	32	H'FCFF9A08	32
	Brightness (DC) correction register 2	OUT_BRIGHT2	32	H'FCFF9A0C	32
	Contrast (gain) correction register	OUT_CONTRAST	32	H'FCFF9A10	32
	Panel dither register	OUT_PDTHA	32	H'FCFF9A14	32
	Output phase control register	OUT_CLK_PHASE	32	H'FCFF9A24	32
	Interrupt control register 1	SYSCNT_INT1	32	H'FCFF9A80	32
	Interrupt control register 2	SYSCNT_INT2	32	H'FCFF9A84	32
	Interrupt control register 3	SYSCNT_INT3	32	H'FCFF9A88	32
	Interrupt control register 4	SYSCNT_INT4	32	H'FCFF9A8C	32
Interrupt control register 5	SYSCNT_INT5	32	H'FCFF9A90	32	
Interrupt control register 6	SYSCNT_INT6	32	H'FCFF9A94	32	
Panel clock control register	SYSCNT_PANEL_CLK	16	H'FCFF9A98	16	
CLUT table read select signal status register	SYSCNT_CLUT	16	H'FCFF9A9A	16	
LVDS output interface	LVDS register update control register	LVDS_UPDATE	32	H'FCFF7A30	32
	LVDS format conversion register L	LVDSFCL	32	H'FCFF7A34	32
	LVDS clock select register	LCLKSELR	32	H'FCFF7A50	32
	LVDSPLL setting register	LPLLSETR	32	H'FCFF7A54	32
	LVDS PHY characteristics switching register	LPHYACC	32	H'FCFF7A5C	32
Image renderer channel 0	Control register	CR	32	H'FCFF3008	32
	Status register	SR	32	H'FCFF300C	32
	Status clear register	SRCLR	32	H'FCFF3010	32
	Interrupt control register	ICR	32	H'FCFF3014	32
	Interrupt mask register	IMR	32	H'FCFF3018	32
	DL status register	DLPR	32	H'FCFF3020	32
	DL start address register	DLSAR	32	H'FCFF3030	32
	Destination start address register	DSAR	32	H'FCFF3034	32
	Destination stride register	DSTR	32	H'FCFF303C	32
	Destination start address register 2	DSAR2	32	H'FCFF3048	32
	DL start address register 2	DLSAR2	32	H'FCFF304C	32

**Table 57.1 Register Addresses**

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Image renderer channel 0	Triangle mode register	TRIMR	32	H'FCFF3060	32
	Triangle set register	TRIMSR	32	H'FCFF3064	32
	Triangle clear register	TRIMCR	32	H'FCFF3068	32
	Triangle color register	TRICR	32	H'FCFF306C	32
	Source and destination coordinate decimal point register	UVDPOR	32	H'FCFF3070	32
	Source width register	SUSR	32	H'FCFF3074	32
	Source height register	SVSR	32	H'FCFF3078	32
	Min. clipping X register	XMINR	32	H'FCFF3080	32
	Min. clipping Y register	YMINR	32	H'FCFF3084	32
	Max. clipping X register	XMAXR	32	H'FCFF3088	32
	Max. clipping Y register	YMAXR	32	H'FCFF308C	32
	Mesh generation X size register	AMXSR	32	H'FCFF3090	32
	Mesh generation Y size register	AMYSR	32	H'FCFF3094	32
	Mesh generation X start register	AMXOR	32	H'FCFF3098	32
	Mesh generation Y start register	AMYOR	32	H'FCFF309C	32
	Memory access control register 1	MACR1	32	H'FCFF3100	32
	Start line specification register	LSPR	32	H'FCFF3A00	32
	End line specification register	LEPR	32	H'FCFF3A04	32
	Mesh size register	LMSR	32	H'FCFF3A08	32
	Image renderer channel 1	Line memory control register	LMCR	32	H'FCFF3A20
Line memory pre-clip start register		LMSPPCR	32	H'FCFF3A24	32
Line memory pre-clip end register		LMEPPCR	32	H'FCFF3A28	32
Control register		CR	32	H'FCFFE008	32
Status register		SR	32	H'FCFFE00C	32
Status clear register		SRCR	32	H'FCFFE010	32
Interrupt control register		ICR	32	H'FCFFE014	32
Interrupt mask register		IMR	32	H'FCFFE018	32
DL status register		DLPR	32	H'FCFFE020	32
DL start address register		DLSAR	32	H'FCFFE030	32
Destination start address register		DSAR	32	H'FCFFE034	32
Destination stride register		DSTR	32	H'FCFFE03C	32
Destination start address register 2		DSAR2	32	H'FCFFE048	32
DL start address register 2		DLSAR2	32	H'FCFFE04C	32
Triangle mode register		TRIMR	32	H'FCFFE060	32
Triangle set register		TRIMSR	32	H'FCFFE064	32
Triangle clear register		TRIMCR	32	H'FCFFE068	32
Triangle color register		TRICR	32	H'FCFFE06C	32
Source and destination coordinate decimal point register		UVDPOR	32	H'FCFFE070	32
Source width register		SUSR	32	H'FCFFE074	32
Source height register		SVSR	32	H'FCFFE078	32
Min. clipping X register		XMINR	32	H'FCFFE080	32
Min. clipping Y register		YMINR	32	H'FCFFE084	32
Max. clipping X register		XMAXR	32	H'FCFFE088	32
Max. clipping Y register		YMAXR	32	H'FCFFE08C	32
Mesh generation X size register		AMXSR	32	H'FCFFE090	32
Mesh generation Y size register		AMYSR	32	H'FCFFE094	32
Mesh generation X start register		AMXOR	32	H'FCFFE098	32
Mesh generation Y start register		AMYOR	32	H'FCFFE09C	32
Memory access control register 1		MACR1	32	H'FCFFE100	32
Start line specification register		LSPR	32	H'FCFFEA00	32
End line specification register		LEPR	32	H'FCFFEA04	32
Mesh size register		LMSR	32	H'FCFFEA08	32

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Image renderer channel 1	Line memory control register	LMCR	32	H'FCFFEA20	32
	Line memory pre-clip start register	LMSPPCR	32	H'FCFFEA24	32
	Line memory pre-clip end register	LMEPPCR	32	H'FCFFEA28	32
Image renderer for display	Control register	CR	32	H'FCFF2008	32
	Status register	SR	32	H'FCFF200C	32
	Status clear register	SRCLR	32	H'FCFF2010	32
	Interrupt control register	ICR	32	H'FCFF2014	32
	Interrupt mask register	IMR	32	H'FCFF2018	32
	DL status register	DLPR	32	H'FCFF2020	32
	DL start address register	DLSAR	32	H'FCFF2030	32
	Destination start address register	DSAR	32	H'FCFF2034	32
	Destination stride register	DSTR	32	H'FCFF203C	32
	Destination start address register 2	DSAR2	32	H'FCFF2048	32
	Triangle mode register	TRIMR	32	H'FCFF2060	32
	Triangle set register	TRIMSR	32	H'FCFF2064	32
	Triangle clear register	TRIMCR	32	H'FCFF2068	32
	Triangle color register	TRICR	32	H'FCFF206C	32
	Source and destination coordinate decimal point register	UVDPOR	32	H'FCFF2070	32
	Source width register	SUSR	32	H'FCFF2074	32
	Source height register	SVSR	32	H'FCFF2078	32
	Min. clipping X register	XMINR	32	H'FCFF2080	32
	Min. clipping Y register	YMINR	32	H'FCFF2084	32
	Max. clipping X register	XMAXR	32	H'FCFF2088	32
	Max. clipping Y register	YMAXR	32	H'FCFF208C	32
	Mesh generation X size register	AMXSR	32	H'FCFF2090	32
	Mesh generation Y size register	AMYSR	32	H'FCFF2094	32
	Mesh generation X start register	AMXOR	32	H'FCFF2098	32
	Mesh generation Y start register	AMYOR	32	H'FCFF209C	32
	Memory access control register 1	MACR1	32	H'FCFF20A0	32
	Start line specification register	LSPR	32	H'FCFF2A00	32
	End line specification register	LEPR	32	H'FCFF2A04	32
Mesh size register	LMSR	32	H'FCFF2A08	32	
Display out comparison unit channel 0	Control register	DOCMCR	32	H'FCFFA800	32
	Status register	DOCMSTR	32	H'FCFFA804	32
	Status clear register	DOCMCLSTR	32	H'FCFFA808	32
	Interrupt enable register	DOCMENR	32	H'FCFFA80C	32
	Operation parameter setting register	DOCMPMR	32	H'FCFFA814	32
	Expected CRC code register	DOCMCCR	32	H'FCFFA818	32
	Calculated CRC code value register	DOCMCCR	32	H'FCFFA81C	32
	Horizontal start position setting register	DOCMSPXR	32	H'FCFFA820	32
	Vertical start position setting register	DOCMSPYR	32	H'FCFFA824	32
	Horizontal size setting register	DOCMSZXR	32	H'FCFFA828	32
	Vertical size setting register	DOCMSZYR	32	H'FCFFA82C	32
	CRC code initialization register	DOCMCRIR	32	H'FCFFA830	32
Display out comparison unit channel 1	Control register	DOCMCR	32	H'FCFFB000	32
	Status register	DOCMSTR	32	H'FCFFB004	32
	Status clear register	DOCMCLSTR	32	H'FCFFB008	32
	Interrupt enable register	DOCMENR	32	H'FCFFB00C	32
	Operation parameter setting register	DOCMPMR	32	H'FCFFB014	32
	Expected CRC code register	DOCMCCR	32	H'FCFFB018	32
	Calculated CRC code value register	DOCMCCR	32	H'FCFFB01C	32
	Horizontal start position setting register	DOCMSPXR	32	H'FCFFB020	32

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Display out comparison unit channel 1	Vertical start position setting register	DOCMSPYR	32	H'FCFFB024	32
	Horizontal size setting register	DOCMSZXR	32	H'FCFFB028	32
	Vertical size setting register	DOCMSZYR	32	H'FCFFB02C	32
	CRC code initialization register	DOCMCRCIR	32	H'FCFFB030	32
JPEG codec unit	JPEG code mode register	JCMOD	8	H'E8017000	8
	JPEG code command register	JCCMD	8	H'E8017001	8
	JPEG code quantization table number register	JCQTN	8	H'E8017003	8
	JPEG code Huffman table number register	JCHTN	8	H'E8017004	8
	JPEG code DRI upper register	JCDRIU	8	H'E8017005	8
	JPEG code DRI lower register	JCDRID	8	H'E8017006	8
	JPEG code vertical size upper register	JCVSZU	8	H'E8017007	8
	JPEG code vertical size lower register	JCVSZD	8	H'E8017008	8
	JPEG code horizontal size upper register	JCHSZU	8	H'E8017009	8
	JPEG code horizontal size lower register	JCHSZD	8	H'E801700A	8
	JPEG code data count upper register	JCDTCU	8	H'E801700B	8
	JPEG code data count middle register	JCDTCM	8	H'E801700C	8
	JPEG code data count lower register	JCDTCD	8	H'E801700D	8
	JPEG interrupt enable register 0	JINTE0	8	H'E801700E	8
	JPEG interrupt status register 0	JINTS0	8	H'E801700F	8
	JPEG code decode error register	JCDERR	8	H'E8017010	8
	JPEG code reset register	JCRST	8	H'E8017011	8
	JPEG interface compression control register	JIFECNT	32	H'E8017040	32
	JPEG interface compression source address register	JIFESA	32	H'E8017044	32
	JPEG interface compression line offset register	JIFESOFST	32	H'E8017048	32
	JPEG interface compression destination address register	JIFEDA	32	H'E801704C	32
	JPEG interface compression source line count register	JIFESLC	32	H'E8017050	32
	JPEG interface compression destination register	JIFEDDC	32	H'E8017054	32
	JPEG interface decompression control register	JIFDCNT	32	H'E8017058	32
	JPEG interface decompression source address register	JIFDSA	32	H'E801705C	32
	JPEG interface decompression destination offset register	JIFDDOFST	32	H'E8017060	32
	JPEG interface decompression destination address register	JIFDDA	32	H'E8017064	32
	JPEG interface decompression source count register	JIFSDSC	32	H'E8017068	32
	JPEG interface decompression destination line count register	JIFDDLCL	32	H'E801706C	32
	JPEG interface decompression $\alpha$ setting register	JIFDADT	32	H'E8017070	32
	JPEG interrupt enable register 1	JINTE1	32	H'E801708C	32
	JPEG interrupt status register 1	JINTS1	32	H'E8017090	32
	JPEG input image data CbCr range setting register	JIFESVSZ	32	H'E8017094	32
JPEG output image data CbCr range setting register	JIFESHSHZ	32	H'E8017098	32	
JPEG code quantization table 0 register	JCQTBL0	512	H'E8017100 to H'E801713F	8	
JPEG code quantization table 1 register	JCQTBL1	512	H'E8017140 to H'E801717F	8	
JPEG code quantization table 2 register	JCQTBL2	512	H'E8017180 to H'E80171BF	8	
JPEG code quantization table 3 register	JCQTBL3	512	H'E80171C0 to H'E80171FF	8	
JPEG code Huffman table DC0 register	JCHTBDO	224	H'E8017200 to H'E801721B	8	
JPEG code Huffman table AC0 register	JCHTBDO	1416	H'E8017220 to H'E80172D1	8	

**Table 57.1 Register Addresses**

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
JPEG codec unit	JPEG code Huffman table DC1 register	JCHTBD1	224	H'E8017300 to H'E801731B	8
	JPEG code Huffman table AC1 register	JCHTBA1	1416	H'E8017320 to H'E80173D1	8
Capture engine unit	CEU Capture start register	CAPSR	32	H'E8210000	32
	CEU Capture control register	CAPCR	32	H'E8210004	32
	CEU Capture interface control register	CAMCR	32	H'E8210008	32
	CEU Capture interface cycle register	CMCYR	32	H'E821000C	32
	CEU Capture interface offset register	CAMOR	32	Plane A: H'E8210010 Plane B: H'E8211010 Mirror: H'E8212010	32
	CEU Capture interface width register	CAPWR	32	Plane A: H'E8210014 Plane B: H'E8211014 Mirror: H'E8212014	32
	CEU Capture interface input format register	CAIFR	32	H'E8210018	32
	CEU register control register	CRCNTR	32	H'E8210028	32
	CEU register forcible control register	CRCMPR	32	H'E821002C	32
	CEU Capture filter control register	CFLCR	32	Plane A: H'E8210030 Plane B: H'E8211030 Mirror: H'E8212030	32
	CEU Capture filter size clip register	CFSZR	32	Plane A: H'E8210034 Plane B: H'E8211034 Mirror: H'E8212034	32
	CEU Capture destination width register	CDWDR	32	Plane A: H'E8210038 Plane B: H'E8211038 Mirror: H'E8212038	32
	CEU Capture data address Y register	CDAYR	32	Plane A: H'E821003C Plane B: H'E821103C Mirror: H'E821203C	32
	CEU Capture data address C register	CDACR	32	Plane A: H'E8210040 Plane B: H'E8211040 Mirror: H'E8212040	32
	CEU Capture data bottom-field address Y register	CDBYR	32	Plane A: H'E8210044 Plane B: H'E8211044 Mirror: H'E8212044	32
	CEU Capture data bottom-field address C register	CDBCR	32	Plane A: H'E8210048 Plane B: H'E8211048 Mirror: H'E8212048	32
	CEU Capture bundle destination size register	CBDSR	32	Plane A: H'E821004C Plane B: H'E821104C Mirror: H'E821204C	32
	CEU Firewall operation control register	CFWCR	32	H'E821005C	32



Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Capture engine unit	CEU Capture low-pass filter control register	CLFCR	32	Plane A: H'E8210060 Plane B: H'E8211060 Mirror: H'E8212060	32
	CEU Capture data output control register	CDOCR	32	Plane A: H'E8210064 Plane B: H'E8211064 Mirror: H'E8212064	32
	CEU Capture event interrupt enable register	CEIER	32	H'E8210070	32
	CEU Capture event flag clear register	CETCR	32	H'E8210074	32
	CEU Capture status register	CSTSR	32	H'E821007C	32
	CEU Capture data size register	CDSSR	32	H'E8210084	32
	CEU Capture data address Y register 2	CDAYR2	32	Plane A: H'E8210090 Plane B: H'E8211090 Mirror: H'E8212090	32
	CEU Capture data address C register 2	CDACR2	32	Plane A: H'E8210094 Plane B: H'E8211094 Mirror: H'E8212094	32
	CEU Capture data bottom-field address Y register 2	CDBYR2	32	Plane A: H'E8210098 Plane B: H'E8211098 Mirror: H'E8212098	32
CEU Capture data bottom-field address C register 2	CDBCR2	32	Plane A: H'E821009C Plane B: H'E821109C Mirror: H'E821209C	32	
Pixel format converter channel 0	PFV control register	PFVCR	32	H'E8205000	32
	PFV interrupt control register	PFVICR	32	H'E8205004	32
	PFV interrupt status register	PFVISR	32	H'E8205008	32
	PFV input buffer register	PFVID	32	H'E8205020 to H'E820503C	16, 32
	PFV output buffer register	PFVOD	32	H'E8205040 to H'E840505C	16, 32
	PFV input FIFO status register	PFVIFSR	32	H'E8205064	32
	PFV output FIFO status register	PFVOFSR	32	H'E8205068	32
	PFV $\alpha$ setting register	PFVACR	32	H'E820506C	32
	PFV matrix mode register	PFV_MTX_MODE	32	H'E8205070	32
	PFV matrix YG adjustment register 0	PFV_MTX_YG_ADJ0	32	H'E8205074	32
	PFV matrix YG adjustment register 1	PFV_MTX_YG_ADJ1	32	H'E8205078	32
	PFV matrix CBB adjustment register 0	PFV_MTX_CBB_ADJ0	32	H'E820507C	32
	PFV matrix CBB adjustment register 1	PFV_MTX_CBB_ADJ1	32	H'E8205080	32
	PFV matrix CRR adjustment register 0	PFV_MTX_CRR_ADJ0	32	H'E8205084	32
	PFV matrix CRR adjustment register 1	PFV_MTX_CRR_ADJ1	32	H'E8205088	32
PFV image size setting register	PFVSZR	32	H'E820508C	32	
Pixel format converter channel 1	PFV control register	PFVCR	32	H'E8205800	32
	PFV interrupt control register	PFVICR	32	H'E8205804	32
	PFV interrupt status register	PFVISR	32	H'E8205808	32
	PFV input buffer register	PFVID	32	H'E8205820 to H'E820583C	16, 32
	PFV output buffer register	PFVOD	32	H'E8205840 to H'E840585C	16, 32
	PFV input FIFO status register	PFVIFSR	32	H'E8205864	32

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Pixel format converter channel 1	PFV output FIFO status register	PFVOFSR	32	H'E8205868	32
	PFV $\alpha$ setting register	PFVACR	32	H'E820586C	32
	PFV matrix mode register	PFV_MTX_MODE	32	H'E8205870	32
	PFV matrix YG adjustment register 0	PFV_MTX_YG_ADJ0	32	H'E8205874	32
	PFV matrix YG adjustment register 1	PFV_MTX_YG_ADJ1	32	H'E8205878	32
	PFV matrix CBB adjustment register 0	PFV_MTX_CBB_ADJ0	32	H'E820587C	32
	PFV matrix CBB adjustment register 1	PFV_MTX_CBB_ADJ1	32	H'E8205880	32
	PFV matrix CRR adjustment register 0	PFV_MTX_CRR_ADJ0	32	H'E8205884	32
	PFV matrix CRR adjustment register 1	PFV_MTX_CRR_ADJ1	32	H'E8205888	32
	PFV image size setting register	PFVSZR	32	H'E820588C	32
SCUX	IPC0_0 Initialization Register	IPCIR_IPC0_0	32	H'E8208000	32
	IPC0_0 Pass Select Register	IPSLR_IPC0_0	32	H'E8208004	32
	IPC0_1 Initialization Register	IPCIR_IPC0_1	32	H'E8208100	32
	IPC0_1 Pass Select Register	IPSLR_IPC0_1	32	H'E8208104	32
	IPC0_2 Initialization Register	IPCIR_IPC0_2	32	H'E8208200	32
	IPC0_2 Pass Select Register	IPSLR_IPC0_2	32	H'E8208204	32
	IPC0_3 Initialization Register	IPCIR_IPC0_3	32	H'E8208300	32
	IPC0_3 Pass Select Register	IPSLR_IPC0_3	32	H'E8208304	32
	OPC0_0 Initialization Register	OPCIR_OPC0_0	32	H'E8208400	32
	OPC0_0 Pass Select Register	OPSLR_OPC0_0	32	H'E8208404	32
	OPC0_1 Initialization Register	OPCIR_OPC0_1	32	H'E8208500	32
	OPC0_1 Pass Select Register	OPSLR_OPC0_1	32	H'E8208504	32
	OPC0_2 Initialization Register	OPCIR_OPC0_2	32	H'E8208600	32
	OPC0_2 Pass Select Register	OPSLR_OPC0_2	32	H'E8208604	32
	OPC0_3 Initialization Register	OPCIR_OPC0_3	32	H'E8208700	32
	OPC0_3 Pass Select Register	OPSLR_OPC0_3	32	H'E8208704	32
	FFD0_0 FIFO Download Initialization Register	FFDIR_FFD0_0	32	H'E8208800	32
	FFD0_0 FIFO Download Audio Information Register	FDAIR_FFD0_0	32	H'E8208804	32
	FFD0_0 FIFO Download Request Size Register	DRQSR_FFD0_0	32	H'E8208808	32
	FFD0_0 FIFO Download Pass Register	FFDPR_FFD0_0	32	H'E820880C	32
	FFD0_0 FIFO Download Boot Register	FFDBR_FFD0_0	32	H'E8208810	32
	FFD0_0 FIFO Download Event Mask Register	DEVMR_FFD0_0	32	H'E8208814	32
	FFD0_0 FIFO Download Event Clear Register	DEVCR_FFD0_0	32	H'E820881C	32
	FFD0_1 FIFO Download Initialization Register	FFDIR_FFD0_1	32	H'E8208900	32
	FFD0_1 FIFO Download Audio Information Register	FDAIR_FFD0_1	32	H'E8208904	32
	FFD0_1 FIFO Download Request Size Register	DRQSR_FFD0_1	32	H'E8208908	32
	FFD0_1 FIFO Download Pass Register	FFDPR_FFD0_1	32	H'E820890C	32
	FFD0_1 FIFO Download Boot Register	FFDBR_FFD0_1	32	H'E8208910	32
	FFD0_1 FIFO Download Event Mask Register	DEVMR_FFD0_1	32	H'E8208914	32
	FFD0_1 FIFO Download Event Clear Register	DEVCR_FFD0_1	32	H'E820891C	32
	FFD0_2 FIFO Download Initialization Register	FFDIR_FFD0_2	32	H'E8208A00	32
	FFD0_2 FIFO Download Audio Information Register	FDAIR_FFD0_2	32	H'E8208A04	32
	FFD0_2 FIFO Download Request Size Register	DRQSR_FFD0_2	32	H'E8208A08	32
	FFD0_2 FIFO Download Pass Register	FFDPR_FFD0_2	32	H'E8208A0C	32
	FFD0_2 FIFO Download Boot Register	FFDBR_FFD0_2	32	H'E8208A10	32
	FFD0_2 FIFO Download Event Mask Register	DEVMR_FFD0_2	32	H'E8208A14	32
	FFD0_2 FIFO Download Event Clear Register	DEVCR_FFD0_2	32	H'E8208A1C	32
	FFD0_3 FIFO Download Initialization Register	FFDIR_FFD0_3	32	H'E8208B00	32
	FFD0_3 FIFO Download Audio Information Register	FDAIR_FFD0_3	32	H'E8208B04	32
	FFD0_3 FIFO Download Request Size Register	DRQSR_FFD0_3	32	H'E8208B08	32
	FFD0_3 FIFO Download Pass Register	FFDPR_FFD0_3	32	H'E8208B0C	32
	FFD0_3 FIFO Download Boot Register	FFDBR_FFD0_3	32	H'E8208B10	32

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
SCUX	FFD0_3 FIFO Download Event Mask Register	DEVMR_FFD0_3	32	H'E8208B14	32
	FFD0_3 FIFO Download Event Clear Register	DEVCR_FFD0_3	32	H'E8208B1C	32
	FFU0_0 FIFO Upload Initialization Register	FFUIR_FFU0_0	32	H'E8208C00	32
	FFU0_0 FIFO Upload Audio Information Register	FUAIR_FFU0_0	32	H'E8208C04	32
	FFU0_0 FIFO Upload Request Size Register	URQSR_FFU0_0	32	H'E8208C08	32
	FFU0_0 FIFO Upload Pass Register	FFUPR_FFU0_0	32	H'E8208C0C	32
	FFU0_0 FIFO Upload Event Mask Register	UEVMR_FFU0_0	32	H'E8208C10	32
	FFU0_0 FIFO Upload Event Clear Register	UEVCR_FFU0_0	32	H'E8208C18	32
	FFU0_1 FIFO Upload Initialization Register	FFUIR_FFU0_1	32	H'E8208D00	32
	FFU0_1 FIFO Upload Audio Information Register	FUAIR_FFU0_1	32	H'E8208D04	32
	FFU0_1 FIFO Upload Request Size Register	URQSR_FFU0_1	32	H'E8208D08	32
	FFU0_1 FIFO Upload Pass Register	FFUPR_FFU0_1	32	H'E8208D0C	32
	FFU0_1 FIFO Upload Event Mask Register	UEVMR_FFU0_1	32	H'E8208D10	32
	FFU0_1 FIFO Upload Event Clear Register	UEVCR_FFU0_1	32	H'E8208D18	32
	FFU0_2 FIFO Upload Initialization Register	FFUIR_FFU0_2	32	H'E8208E00	32
	FFU0_2 FIFO Upload Audio Information Register	FUAIR_FFU0_2	32	H'E8208E04	32
	FFU0_2 FIFO Upload Request Size Register	URQSR_FFU0_2	32	H'E8208E08	32
	FFU0_2 FIFO Upload Pass Register	FFUPR_FFU0_2	32	H'E8208E0C	32
	FFU0_2 FIFO Upload Event Mask Register	UEVMR_FFU0_2	32	H'E8208E10	32
	FFU0_2 FIFO Upload Event Clear Register	UEVCR_FFU0_2	32	H'E8208E18	32
	FFU0_3 FIFO Upload Initialization Register	FFUIR_FFU0_3	32	H'E8208F00	32
	FFU0_3 FIFO Upload Audio Information Register	FUAIR_FFU0_3	32	H'E8208F04	32
	FFU0_3 FIFO Upload Request Size Register	URQSR_FFU0_3	32	H'E8208F08	32
	FFU0_3 FIFO Upload Pass Register	FFUPR_FFU0_3	32	H'E8208F0C	32
	FFU0_3 FIFO Upload Event Mask Register	UEVMR_FFU0_3	32	H'E8208F10	32
	FFU0_3 FIFO Upload Event Clear Register	UEVCR_FFU0_3	32	H'E8208F18	32
	2SRC0_0 Initialization Register 0	SRCIR0_2SRC0_0	32	H'E8209000	32
	2SRC0_0 Audio Information Register 0	SADIR0_2SRC0_0	32	H'E8209004	32
	2SRC0_0 Bypass Register 0	SRCBR0_2SRC0_0	32	H'E8209008	32
	2SRC0_0 IFS Control Register 0	IFSCR0_2SRC0_0	32	H'E820900C	32
	2SRC0_0 IFS Value Setting Register 0	IFSVR0_2SRC0_0	32	H'E8209010	32
	2SRC0_0 Control Register 0	SRCCR0_2SRC0_0	32	H'E8209014	32
	2SRC0_0 Minimum FS Setting Register 0	MNFSR0_2SRC0_0	32	H'E8209018	32
	2SRC0_0 Buffer Size Setting Register 0	BFSSR0_2SRC0_0	32	H'E820901C	32
	2SRC0_0 SCU2 Status Register 0	SC2SR0_2SRC0_0	32	H'E8209020	32
	2SRC0_0 Wait Time Setting Register 0	WATSR0_2SRC0_0	32	H'E8209024	32
	2SRC0_0 Event Mask Register 0	SEVMR0_2SRC0_0	32	H'E8209028	32
	2SRC0_0 Event Clear Register 0	SEVCR0_2SRC0_0	32	H'E8209030	32
	2SRC0_0 Initialization Register 1	SRCIR1_2SRC0_0	32	H'E8209034	32
	2SRC0_0 Audio Information Register 1	SADIR1_2SRC0_0	32	H'E8209038	32
	2SRC0_0 Bypass Register 1	SRCBR1_2SRC0_0	32	H'E820903C	32
	2SRC0_0 IFS Control Register 1	IFSCR1_2SRC0_0	32	H'E8209040	32
	2SRC0_0 IFS Value Setting Register 1	IFSVR1_2SRC0_0	32	H'E8209044	32
	2SRC0_0 Control Register 1	SRCCR1_2SRC0_0	32	H'E8209048	32
	2SRC0_0 Minimum FS Setting Register 1	MNFSR1_2SRC0_0	32	H'E820904C	32
	2SRC0_0 Buffer Size Setting Register 1	BFSSR1_2SRC0_0	32	H'E8209050	32
	2SRC0_0 SCU2 Status Register 1	SC2SR1_2SRC0_0	32	H'E8209054	32
	2SRC0_0 Wait Time Setting Register 1	WATSR1_2SRC0_0	32	H'E8209058	32
	2SRC0_0 Event Mask Register 1	SEVMR1_2SRC0_0	32	H'E820905C	32
	2SRC0_0 Event Clear Register 1	SEVCR1_2SRC0_0	32	H'E8209064	32
	2SRC0_0 Initialization Register RIF	SRCIRR_2SRC0_0	32	H'E8209068	32
2SRC0_1 Initialization Register 0	SRCIR0_2SRC0_1	32	H'E8209100	32	

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
SCUX	2SRC0_1 Audio Information Register 0	SADIR0_2SRC0_1	32	H'E8209104	32
	2SRC0_1 Bypass Register 0	SRCBR0_2SRC0_1	32	H'E8209108	32
	2SRC0_1 IFS Control Register 0	IFSCR0_2SRC0_1	32	H'E820910C	32
	2SRC0_1 IFS Value Setting Register 0	IFSVR0_2SRC0_1	32	H'E8209110	32
	2SRC0_1 Control Register 0	SRCCR0_2SRC0_1	32	H'E8209114	32
	2SRC0_1 Minimum FS Setting Register 0	MNFSR0_2SRC0_1	32	H'E8209118	32
	2SRC0_1 Buffer Size Setting Register 0	BFSSR0_2SRC0_1	32	H'E820911C	32
	2SRC0_1 SCU2 Status Register 0	SC2SR0_2SRC0_1	32	H'E8209120	32
	2SRC0_1 Wait Time Setting Register 0	WATSR0_2SRC0_1	32	H'E8209124	32
	2SRC0_1 Event Mask Register 0	SEVMR0_2SRC0_1	32	H'E8209128	32
	2SRC0_1 Event Clear Register 0	SEVCR0_2SRC0_1	32	H'E8209130	32
	2SRC0_1 Initialization Register 1	SRCIR1_2SRC0_1	32	H'E8209134	32
	2SRC0_1 Audio Information Register 1	SADIR1_2SRC0_1	32	H'E8209138	32
	2SRC0_1 Bypass Register 1	SRCBR1_2SRC0_1	32	H'E820913C	32
	2SRC0_1 IFS Control Register 1	IFSCR1_2SRC0_1	32	H'E8209140	32
	2SRC0_1 IFS Value Setting Register 1	IFSVR1_2SRC0_1	32	H'E8209144	32
	2SRC0_1 Control Register 1	SRCCR1_2SRC0_1	32	H'E8209148	32
	2SRC0_1 Minimum FS Setting Register 1	MNFSR1_2SRC0_1	32	H'E820914C	32
	2SRC0_1 Buffer Size Setting Register 1	BFSSR1_2SRC0_1	32	H'E8209150	32
	2SRC0_1 SCU2 Status Register 1	SC2SR1_2SRC0_1	32	H'E8209154	32
	2SRC0_1 Wait Time Setting Register 1	WATSR1_2SRC0_1	32	H'E8209158	32
	2SRC0_1 Event Mask Register 1	SEVMR1_2SRC0_1	32	H'E820915C	32
	2SRC0_1 Event Clear Register 1	SEVCR1_2SRC0_1	32	H'E8209164	32
	2SRC0_1 Initialization Register RIF	SRCIRR_2SRC0_1	32	H'E8209168	32
	DVU0_0 Initialization Register	DVUIR_DVU0_0	32	H'E8209200	32
	DVU0_0 Audio Information Register	VADIR_DVU0_0	32	H'E8209204	32
	DVU0_0 Bypass Register	DVUBR_DVU0_0	32	H'E8209208	32
	DVU0_0 Control Register	DVUCR_DVU0_0	32	H'E820920C	32
	DVU0_0 Zero Cross Mute Control Register	ZCMCR_DVU0_0	32	H'E8209210	32
	DVU0_0 Volume Ramp Control Register	VRCTR_DVU0_0	32	H'E8209214	32
	DVU0_0 Volume Ramp Period Register	VRPDR_DVU0_0	32	H'E8209218	32
	DVU0_0 Volume Ramp Decibel Register	VRDBR_DVU0_0	32	H'E820921C	32
	DVU0_0 Volume Ramp Wait Time Register	VRWTR_DVU0_0	32	H'E8209220	32
	DVU0_0 Volume Value Setting 0 Register	VOL0R_DVU0_0	32	H'E8209224	32
	DVU0_0 Volume Value Setting 1 Register	VOL1R_DVU0_0	32	H'E8209228	32
	DVU0_0 Volume Value Setting 2 Register	VOL2R_DVU0_0	32	H'E820922C	32
	DVU0_0 Volume Value Setting 3 Register	VOL3R_DVU0_0	32	H'E8209230	32
	DVU0_0 Volume Value Setting 4 Register	VOL4R_DVU0_0	32	H'E8209234	32
	DVU0_0 Volume Value Setting 5 Register	VOL5R_DVU0_0	32	H'E8209238	32
	DVU0_0 Volume Value Setting 6 Register	VOL6R_DVU0_0	32	H'E820923C	32
	DVU0_0 Volume Value Setting 7 Register	VOL7R_DVU0_0	32	H'E8209240	32
	DVU0_0 Enable Register	DVUER_DVU0_0	32	H'E8209244	32
	DVU0_0 Status Register	DVUSR_DVU0_0	32	H'E8209248	32
	DVU0_0 Event Mask Register	VEVMR_DVU0_0	32	H'E820924C	32
	DVU0_0 Event Clear Register	VEVCR_DVU0_0	32	H'E8209254	32
	DVU0_1 Initialization Register	DVUIR_DVU0_1	32	H'E8209300	32
DVU0_1 Audio Information Register	VADIR_DVU0_1	32	H'E8209304	32	
DVU0_1 Bypass Register	DVUBR_DVU0_1	32	H'E8209308	32	
DVU0_1 Control Register	DVUCR_DVU0_1	32	H'E820930C	32	
DVU0_1 Zero Cross Mute Control Register	ZCMCR_DVU0_1	32	H'E8209310	32	
DVU0_1 Volume Ramp Control Register	VRCTR_DVU0_1	32	H'E8209314	32	
DVU0_1 Volume Ramp Period Register	VRPDR_DVU0_1	32	H'E8209318	32	

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
SCUX	DVU0_1 Volume Ramp Decibel Register	VRDBR_DVU0_1	32	H'E820931C	32
	DVU0_1 Volume Ramp Wait Time Register	VRWTR_DVU0_1	32	H'E8209320	32
	DVU0_1 Volume Value Setting 0 Register	VOL0R_DVU0_1	32	H'E8209324	32
	DVU0_1 Volume Value Setting 1 Register	VOL1R_DVU0_1	32	H'E8209328	32
	DVU0_1 Volume Value Setting 2 Register	VOL2R_DVU0_1	32	H'E820932C	32
	DVU0_1 Volume Value Setting 3 Register	VOL3R_DVU0_1	32	H'E8209330	32
	DVU0_1 Volume Value Setting 4 Register	VOL4R_DVU0_1	32	H'E8209334	32
	DVU0_1 Volume Value Setting 5 Register	VOL5R_DVU0_1	32	H'E8209338	32
	DVU0_1 Volume Value Setting 6 Register	VOL6R_DVU0_1	32	H'E820933C	32
	DVU0_1 Volume Value Setting 7 Register	VOL7R_DVU0_1	32	H'E8209340	32
	DVU0_1 Enable Register	DVUER_DVU0_1	32	H'E8209344	32
	DVU0_1 Status Register	DVUSR_DVU0_1	32	H'E8209348	32
	DVU0_1 Event Mask Register	VEVMR_DVU0_1	32	H'E820934C	32
	DVU0_1 Event Clear Register	VEVCR_DVU0_1	32	H'E8209354	32
	DVU0_2 Initialization Register	DVUIR_DVU0_2	32	H'E8209400	32
	DVU0_2 Audio Information Register	VADIR_DVU0_2	32	H'E8209404	32
	DVU0_2 Bypass Register	DVUBR_DVU0_2	32	H'E8209408	32
	DVU0_2 Control Register	DVUCR_DVU0_2	32	H'E820940C	32
	DVU0_2 Zero Cross Mute Control Register	ZCMCR_DVU0_2	32	H'E8209410	32
	DVU0_2 Volume Ramp Control Register	VRCTR_DVU0_2	32	H'E8209414	32
	DVU0_2 Volume Ramp Period Register	VRPDR_DVU0_2	32	H'E8209418	32
	DVU0_2 Volume Ramp Decibel Register	VRDBR_DVU0_2	32	H'E820941C	32
	DVU0_2 Volume Ramp Wait Time Register	VRWTR_DVU0_2	32	H'E8209420	32
	DVU0_2 Volume Value Setting 0 Register	VOL0R_DVU0_2	32	H'E8209424	32
	DVU0_2 Volume Value Setting 1 Register	VOL1R_DVU0_2	32	H'E8209428	32
	DVU0_2 Volume Value Setting 2 Register	VOL2R_DVU0_2	32	H'E820942C	32
	DVU0_2 Volume Value Setting 3 Register	VOL3R_DVU0_2	32	H'E8209430	32
	DVU0_2 Volume Value Setting 4 Register	VOL4R_DVU0_2	32	H'E8209434	32
	DVU0_2 Volume Value Setting 5 Register	VOL5R_DVU0_2	32	H'E8209438	32
	DVU0_2 Volume Value Setting 6 Register	VOL6R_DVU0_2	32	H'E820943C	32
	DVU0_2 Volume Value Setting 7 Register	VOL7R_DVU0_2	32	H'E8209440	32
	DVU0_2 Enable Register	DVUER_DVU0_2	32	H'E8209444	32
	DVU0_2 Status Register	DVUSR_DVU0_2	32	H'E8209448	32
	DVU0_2 Event Mask Register	VEVMR_DVU0_2	32	H'E820944C	32
	DVU0_2 Event Clear Register	VEVCR_DVU0_2	32	H'E8209454	32
	DVU0_3 Initialization Register	DVUIR_DVU0_3	32	H'E8209500	32
	DVU0_3 Audio Information Register	VADIR_DVU0_3	32	H'E8209504	32
	DVU0_3 Bypass Register	DVUBR_DVU0_3	32	H'E8209508	32
	DVU0_3 Control Register	DVUCR_DVU0_3	32	H'E820950C	32
	DVU0_3 Zero Cross Mute Control Register	ZCMCR_DVU0_3	32	H'E8209510	32
	DVU0_3 Volume Ramp Control Register	VRCTR_DVU0_3	32	H'E8209514	32
	DVU0_3 Volume Ramp Period Register	VRPDR_DVU0_3	32	H'E8209518	32
DVU0_3 Volume Ramp Decibel Register	VRDBR_DVU0_3	32	H'E820951C	32	
DVU0_3 Volume Ramp Wait Time Register	VRWTR_DVU0_3	32	H'E8209520	32	
DVU0_3 Volume Value Setting 0 Register	VOL0R_DVU0_3	32	H'E8209524	32	
DVU0_3 Volume Value Setting 1 Register	VOL1R_DVU0_3	32	H'E8209528	32	
DVU0_3 Volume Value Setting 2 Register	VOL2R_DVU0_3	32	H'E820952C	32	
DVU0_3 Volume Value Setting 3 Register	VOL3R_DVU0_3	32	H'E8209530	32	
DVU0_3 Volume Value Setting 4 Register	VOL4R_DVU0_3	32	H'E8209534	32	
DVU0_3 Volume Value Setting 5 Register	VOL5R_DVU0_3	32	H'E8209538	32	
DVU0_3 Volume Value Setting 6 Register	VOL6R_DVU0_3	32	H'E820953C	32	
DVU0_3 Volume Value Setting 7 Register	VOL7R_DVU0_3	32	H'E8209540	32	

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
SCUX	DVU0_3 Enable Register	DVUER_DVU0_3	32	H'E8209544	32
	DVU0_3 Status Register	DVUSR_DVU0_3	32	H'E8209548	32
	DVU0_3 Event Mask Register	VEVMR_DVU0_3	32	H'E820954C	32
	DVU0_3 Event Clear Register	VEVCR_DVU0_3	32	H'E8209554	32
	MIX0_0 Initialization Register	MIXIR_MIX0_0	32	H'E8209600	32
	MIX0_0 Audio Information Register	MADIR_MIX0_0	32	H'E8209604	32
	MIX0_0 Bypass Register	MIXBR_MIX0_0	32	H'E8209608	32
	MIX0_0 Mode Register	MIXMR_MIX0_0	32	H'E820960C	32
	MIX0_0 Volume Period Register	MVPDR_MIX0_0	32	H'E8209610	32
	MIX0_0 Decibel A Register	MDBAR_MIX0_0	32	H'E8209614	32
	MIX0_0 Decibel B Register	MDBBR_MIX0_0	32	H'E8209618	32
	MIX0_0 Decibel C Register	MDBCR_MIX0_0	32	H'E820961C	32
	MIX0_0 Decibel D Register	MDBDR_MIX0_0	32	H'E8209620	32
	MIX0_0 Decibel Enable Register	MDBER_MIX0_0	32	H'E8209624	32
	MIX0_0 Status Register	MIXSR_MIX0_0	32	H'E8209628	32
	Software Reset Register	SWRSR_CIM	32	H'E8209700	32
	DMA Control Register	DMACR_CIM	32	H'E8209704	32
	DMA Transfer Register for FFD0_0 RAM	DMATD0_CIM	32	H'E8209708	16, 32
	DMA Transfer Register for FFD0_1 RAM	DMATD1_CIM	32	H'E820970C	16, 32
	DMA Transfer Register for FFD0_2 RAM	DMATD2_CIM	32	H'E8209710	16, 32
	DMA Transfer Register for FFD0_3 RAM	DMATD3_CIM	32	H'E8209714	16, 32
	DMA Transfer Register for FFU0_0 RAM	DMATU0_CIM	32	H'E8209718	16, 32
	DMA Transfer Register for FFU0_1 RAM	DMATU1_CIM	32	H'E820971C	16, 32
	DMA Transfer Register for FFU0_2 RAM	DMATU2_CIM	32	H'E8209720	16, 32
	DMA Transfer Register for FFU0_3 RAM	DMATU3_CIM	32	H'E8209724	16, 32
	SSI route select register	SSIRSEL_CIM	32	H'E8209738	32
	FFD0_0 timing select register	FDTSEL0_CIM	32	H'E820973C	32
	FFD0_1 timing select register	FDTSEL1_CIM	32	H'E8209740	32
	FFD0_2 timing select register	FDTSEL2_CIM	32	H'E8209744	32
	FFD0_3 timing select register	FDTSEL3_CIM	32	H'E8209748	32
	FFU0_0 timing select register	FUTSEL0_CIM	32	H'E820974C	32
	FFU0_1 timing select register	FUTSEL1_CIM	32	H'E8209750	32
	FFU0_2 timing select register	FUTSEL2_CIM	32	H'E8209754	32
	FFU0_3 timing select register	FUTSEL3_CIM	32	H'E8209758	32
SSI pin mode register	SSIPMD_CIM	32	H'E820975C	32	
SSI control register	SSICTRL_CIM	32	H'E8209760	32	
SRC0 route select register	SRCRSEL0_CIM	32	H'E8209764	32	
SRC1 route select register	SRCRSEL1_CIM	32	H'E8209768	32	
SRC2 route select register	SRCRSEL2_CIM	32	H'E820976C	32	
SRC3 route select register	SRCRSEL3_CIM	32	H'E8209770	32	
MIX route select register	MIXRSEL_CIM	32	H'E8209774	32	
Sound generator	Sound generator control register 1_0	SGCR1_0	8	H'FCFF4800	8
	Sound generator control status register_0	SGCSR_0	8	H'FCFF4801	8
	Sound generator control register 2_0	SGCR2_0	8	H'FCFF4802	8
	Sound generator loudness register_0	SGLR_0	8	H'FCFF4803	8
	Sound generator tone frequency register_0	SGTFR_0	8	H'FCFF4804	8
	Sound generator reference frequency register_0	SGSFR_0	8	H'FCFF4805	8
	Sound generator control register 1_1	SGCR1_1	8	H'FCFF4A00	8
	Sound generator control status register_1	SGCSR_1	8	H'FCFF4A01	8
	Sound generator control register 2_1	SGCR2_1	8	H'FCFF4A02	8
	Sound generator loudness register_1	SGLR_1	8	H'FCFF4A03	8
	Sound generator tone frequency register_1	SGTFR_1	8	H'FCFF4A04	8

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size	
Sound generator	Sound generator reference frequency register_1	SGSFR_1	8	H'FCFF4A05	8	
	Sound generator control register_1_2	SGCR1_2	8	H'FCFF4C00	8	
	Sound generator control status register_2	SGCSR_2	8	H'FCFF4C01	8	
	Sound generator control register_2_2	SGCR2_2	8	H'FCFF4C02	8	
	Sound generator loudness register_2	SGLR_2	8	H'FCFF4C03	8	
	Sound generator tone frequency register_2	SGTFR_2	8	H'FCFF4C04	8	
	Sound generator reference frequency register_2	SGSFR_2	8	H'FCFF4C05	8	
	Sound generator control register_1_3	SGCR1_3	8	H'FCFF4E00	8	
	Sound generator control status register_3	SGCSR_3	8	H'FCFF4E01	8	
	Sound generator control register_2_3	SGCR2_3	8	H'FCFF4E02	8	
	Sound generator loudness register_3	SGLR_3	8	H'FCFF4E03	8	
	Sound generator tone frequency register_3	SGTFR_3	8	H'FCFF4E04	8	
	Sound generator reference frequency register_3	SGSFR_3	8	H'FCFF4E05	8	
MMC host interface	Command setting register	CE_CMD_SET	32	H'E804C800	16	
				H'E804C802	16	
	Argument register	CE_ARG	32	H'E804C808	32	
	Argument register for automatically-issued CMD12	CE_ARG_CMD12	32	H'E804C80C	32	
	Command control register	CE_CMD_CTRL	32	H'E804C810	32	
	Transfer block setting register	CE_BLOCK_SET	32	H'E804C814	32	
	Clock control register	CE_CLK_CTRL	32	H'E804C818	32	
	Buffer access configuration register	CE_BUF_ACC	32	H'E804C81C	32	
	Response register 3	CE_RESP3	32	H'E804C820	32	
	Response register 2	CE_RESP2	32	H'E804C824	32	
	Response register 1	CE_RESP1	32	H'E804C828	32	
	Response register 0	CE_RESP0	32	H'E804C82C	32	
	Response register for automatically-issued CMD12	CE_RESP_CMD12	32	H'E804C830	32	
	Data register	CE_DATA	32	H'E804C834	32	
	Interrupt flag register	CE_INT	32	H'E804C840	32	
	Interrupt enable register	CE_INT_EN	32	H'E804C844	32	
	Status register 1	CE_HOST_STS1	32	H'E804C848	32	
	Status register 2	CE_HOST_STS2	32	H'E804C84C	32	
	DMA mode setting register	CE_DMA_MODE	32	H'E804C85C	32	
	Card detection/port control register	CE_DETECT	32	H'E804C870	32	
Special mode setting register	CE_ADD_MODE	32	H'E804C874	32		
Version register	CE_VERSION	32	H'E804C87C	32		
Motor control PWM timer	PWM control register_1	PWCR_1	8	H'FCFF50E0	8	
	PWM polarity register_1	PWPR_1	8	H'FCFF50E4	8	
	PWM cycle register_1	PWCYR_1	16	H'FCFF50E6	16	
	PWM buffer register_1A	PWBFR_1A	16	H'FCFF50E8	16	
	PWM buffer register_1C	PWBFR_1C	16	H'FCFF50EA	16	
	PWM buffer register_1E	PWBFR_1E	16	H'FCFF50EC	16	
	PWM buffer register_1G	PWBFR_1G	16	H'FCFF50EE	16	
	PWM control register_2	PWCR_2	8	H'FCFF50F0	8	
	PWM polarity register_2	PWPR_2	8	H'FCFF50F4	8	
	PWM cycle register_2	PWCYR_2	16	H'FCFF50F6	16	
	PWM buffer register_2A	PWBFR_2A	16	H'FCFF50F8	16	
	PWM buffer register_2C	PWBFR_2C	16	H'FCFF50FA	16	
	PWM buffer register_2E	PWBFR_2E	16	H'FCFF50FC	16	
	PWM buffer register_2G	PWBFR_2G	16	H'FCFF50FE	16	
	PWM buffer transfer control register	PWBTCR	8	H'FCFF5006	8	
	Ports	Port register_1	P1	16	H'FCFE3004	16
		Port register_2	P2	16	H'FCFE3008	16

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Ports	Port register_3	P3	16	H'FCFE300C	16
	Port register_4	P4	16	H'FCFE3010	16
	Port register_5	P5	16	H'FCFE3014	16
	Port register_6	P6	16	H'FCFE3018	16
	Port register_7	P7	16	H'FCFE301C	16
	Port register_8	P8	16	H'FCFE3020	16
	Port register_9	P9	16	H'FCFE3024	16
	Port register_10	P10	16	H'FCFE3028	16
	Port register_11	P11	16	H'FCFE302C	16
	Port set/reset register_1	PSR1	32	H'FCFE3104	32
	Port set/reset register_2	PSR2	32	H'FCFE3108	32
	Port set/reset register_3	PSR3	32	H'FCFE310C	32
	Port set/reset register_4	PSR4	32	H'FCFE3110	32
	Port set/reset register_5	PSR5	32	H'FCFE3114	32
	Port set/reset register_6	PSR6	32	H'FCFE3118	32
	Port set/reset register_7	PSR7	32	H'FCFE311C	32
	Port set/reset register_8	PSR8	32	H'FCFE3120	32
	Port set/reset register_9	PSR9	32	H'FCFE3124	32
	Port set/reset register_10	PSR10	32	H'FCFE3128	32
	Port set/reset register_11	PSR11	32	H'FCFE312C	32
	Port pin read register_0	PPR0	16	H'FCFE3200	16
	Port pin read register_1	PPR1	16	H'FCFE3204	16
	Port pin read register_2	PPR2	16	H'FCFE3208	16
	Port pin read register_3	PPR3	16	H'FCFE320C	16
	Port pin read register_4	PPR4	16	H'FCFE3210	16
	Port pin read register_5	PPR5	16	H'FCFE3214	16
	Port pin read register_6	PPR6	16	H'FCFE3218	16
	Port pin read register_7	PPR7	16	H'FCFE321C	16
	Port pin read register_8	PPR8	16	H'FCFE3220	16
	Port pin read register_9	PPR9	16	H'FCFE3224	16
	Port pin read register_10	PPR10	16	H'FCFE3228	16
	Port pin read register_11	PPR11	16	H'FCFE322C	16
	Port mode register_1	PM1	16	H'FCFE3304	16
	Port mode register_2	PM2	16	H'FCFE3308	16
	Port mode register_3	PM3	16	H'FCFE330C	16
	Port mode register_4	PM4	16	H'FCFE3310	16
	Port mode register_5	PM5	16	H'FCFE3314	16
	Port mode register_6	PM6	16	H'FCFE3318	16
	Port mode register_7	PM7	16	H'FCFE331C	16
	Port mode register_8	PM8	16	H'FCFE3320	16
	Port mode register_9	PM9	16	H'FCFE3324	16
	Port mode register_10	PM10	16	H'FCFE3328	16
	Port mode register_11	PM11	16	H'FCFE332C	16
	Port mode control register_0	PMC0	16	H'FCFE3400	16
	Port mode control register_1	PMC1	16	H'FCFE3404	16
	Port mode control register_2	PMC2	16	H'FCFE3408	16
	Port mode control register_3	PMC3	16	H'FCFE340C	16
	Port mode control register_4	PMC4	16	H'FCFE3410	16
	Port mode control register_5	PMC5	16	H'FCFE3414	16
	Port mode control register_6	PMC6	16	H'FCFE3418	16
Port mode control register_7	PMC7	16	H'FCFE341C	16	
Port mode control register_8	PMC8	16	H'FCFE3420	16	



Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Ports	Port mode control register_9	PMC9	16	H'FCFE3424	16
	Port mode control register_10	PMC10	16	H'FCFE3428	16
	Port mode control register_11	PMC11	16	H'FCFE342C	16
	Port function control register_1	PFC1	16	H'FCFE3504	16
	Port function control register_2	PFC2	16	H'FCFE3508	16
	Port function control register_3	PFC3	16	H'FCFE350C	16
	Port function control register_4	PFC4	16	H'FCFE3510	16
	Port function control register_5	PFC5	16	H'FCFE3514	16
	Port function control register_6	PFC6	16	H'FCFE3518	16
	Port function control register_7	PFC7	16	H'FCFE351C	16
	Port function control register_8	PFC8	16	H'FCFE3520	16
	Port function control register_9	PFC9	16	H'FCFE3524	16
	Port function control register_10	PFC10	16	H'FCFE3528	16
	Port function control register_11	PFC11	16	H'FCFE352C	16
	Port function control expansion register_1	PFCE1	16	H'FCFE3604	16
	Port function control expansion register_2	PFCE2	16	H'FCFE3608	16
	Port function control expansion register_3	PFCE3	16	H'FCFE360C	16
	Port function control expansion register_4	PFCE4	16	H'FCFE3610	16
	Port function control expansion register_5	PFCE5	16	H'FCFE3614	16
	Port function control expansion register_6	PFCE6	16	H'FCFE3618	16
	Port function control expansion register_7	PFCE7	16	H'FCFE361C	16
	Port function control expansion register_8	PFCE8	16	H'FCFE3620	16
	Port function control expansion register_9	PFCE9	16	H'FCFE3624	16
	Port function control expansion register_10	PFCE10	16	H'FCFE3628	16
	Port function control expansion register_11	PFCE11	16	H'FCFE362C	16
	Port NOT register_1	PNOT1	16	H'FCFE3704	16
	Port NOT register_2	PNOT2	16	H'FCFE3708	16
	Port NOT register_3	PNOT3	16	H'FCFE370C	16
	Port NOT register_4	PNOT4	16	H'FCFE3710	16
	Port NOT register_5	PNOT5	16	H'FCFE3714	16
	Port NOT register_6	PNOT6	16	H'FCFE3718	16
	Port NOT register_7	PNOT7	16	H'FCFE371C	16
	Port NOT register_8	PNOT8	16	H'FCFE3720	16
	Port NOT register_9	PNOT9	16	H'FCFE3724	16
	Port NOT register_10	PNOT10	16	H'FCFE3728	16
	Port NOT register_11	PNOT11	16	H'FCFE372C	16
	Port mode set/reset register_1	PMSR1	32	H'FCFE3804	32
	Port mode set/reset register_2	PMSR2	32	H'FCFE3808	32
	Port mode set/reset register_3	PMSR3	32	H'FCFE380C	32
	Port mode set/reset register_4	PMSR4	32	H'FCFE3810	32
	Port mode set/reset register_5	PMSR5	32	H'FCFE3814	32
	Port mode set/reset register_6	PMSR6	32	H'FCFE3818	32
	Port mode set/reset register_7	PMSR7	32	H'FCFE381C	32
	Port mode set/reset register_8	PMSR8	32	H'FCFE3820	32
Port mode set/reset register_9	PMSR9	32	H'FCFE3824	32	
Port mode set/reset register_10	PMSR10	32	H'FCFE3828	32	
Port mode set/reset register_11	PMSR11	32	H'FCFE382C	32	
Port mode control set/reset register_0	PMCSR0	32	H'FCFE3900	32	
Port mode control set/reset register_1	PMCSR1	32	H'FCFE3904	32	
Port mode control set/reset register_2	PMCSR2	32	H'FCFE3908	32	

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Ports	Port mode control set/reset register_3	PMCSR3	32	H'FCFE390C	32
	Port mode control set/reset register_4	PMCSR4	32	H'FCFE3910	32
	Port mode control set/reset register_5	PMCSR5	32	H'FCFE3914	32
	Port mode control set/reset register_6	PMCSR6	32	H'FCFE3918	32
	Port mode control set/reset register_7	PMCSR7	32	H'FCFE391C	32
	Port mode control set/reset register_8	PMCSR8	32	H'FCFE3920	32
	Port mode control set/reset register_9	PMCSR9	32	H'FCFE3924	32
	Port mode control set/reset register_10	PMCSR10	32	H'FCFE3928	32
	Port mode control set/reset register_11	PMCSR11	32	H'FCFE392C	32
	Port function control additional expansion register_1	PFCAE1	16	H'FCFE3A04	16
	Port function control additional expansion register_2	PFCAE2	16	H'FCFE3A08	16
	Port function control additional expansion register_3	PFCAE3	16	H'FCFE3A0C	16
	Port function control additional expansion register_4	PFCAE4	16	H'FCFE3A10	16
	Port function control additional expansion register_5	PFCAE5	16	H'FCFE3A14	16
	Port function control additional expansion register_6	PFCAE6	16	H'FCFE3A18	16
	Port function control additional expansion register_7	PFCAE7	16	H'FCFE3A1C	16
	Port function control additional expansion register_8	PFCAE8	16	H'FCFE3A20	16
	Port function control additional expansion register_9	PFCAE9	16	H'FCFE3A24	16
	Port function control additional expansion register_10	PFCAE10	16	H'FCFE3A28	16
	Port function control additional expansion register_11	PFCAE11	16	H'FCFE3A2C	16
	Port input buffer control register_0	PIBC0	16	H'FCFE7000	16
	Port input buffer control register_1	PIBC1	16	H'FCFE7004	16
	Port input buffer control register_2	PIBC2	16	H'FCFE7008	16
	Port input buffer control register_3	PIBC3	16	H'FCFE700C	16
	Port input buffer control register_4	PIBC4	16	H'FCFE7010	16
	Port input buffer control register_5	PIBC5	16	H'FCFE7014	16
	Port input buffer control register_6	PIBC6	16	H'FCFE7018	16
	Port input buffer control register_7	PIBC7	16	H'FCFE701C	16
	Port input buffer control register_8	PIBC8	16	H'FCFE7020	16
	Port input buffer control register_9	PIBC9	16	H'FCFE7024	16
	Port input buffer control register_10	PIBC10	16	H'FCFE7028	16
	Port input buffer control register_11	PIBC11	16	H'FCFE702C	16
	Port bi-direction control register_1	PBDC1	16	H'FCFE7104	16
	Port bi-direction control register_2	PBDC2	16	H'FCFE7108	16
	Port bi-direction control register_3	PBDC3	16	H'FCFE710C	16
	Port bi-direction control register_4	PBDC4	16	H'FCFE7110	16
	Port bi-direction control register_5	PBDC5	16	H'FCFE7114	16
	Port bi-direction control register_6	PBDC6	16	H'FCFE7118	16
	Port bi-direction control register_7	PBDC7	16	H'FCFE711C	16
	Port bi-direction control register_8	PBDC8	16	H'FCFE7120	16
	Port bi-direction control register_9	PBDC9	16	H'FCFE7124	16
	Port bi-direction control register_10	PBDC10	16	H'FCFE7128	16
	Port bi-direction control register_11	PBDC11	16	H'FCFE712C	16
	Port IP control register_1	PIPC1	16	H'FCFE7204	16
Port IP control register_2	PIPC2	16	H'FCFE7208	16	
Port IP control register_3	PIPC3	16	H'FCFE720C	16	
Port IP control register_4	PIPC4	16	H'FCFE7210	16	
Port IP control register_5	PIPC5	16	H'FCFE7214	16	
Port IP control register_6	PIPC6	16	H'FCFE7218	16	
Port IP control register_7	PIPC7	16	H'FCFE721C	16	
Port IP control register_8	PIPC8	16	H'FCFE7220	16	

**Table 57.1 Register Addresses**

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Ports	Port IP control register_9	PIPC9	16	H'FCFE7224	16
	Port IP control register_10	PIPC10	16	H'FCFE7228	16
	Port IP control register_11	PIPC11	16	H'FCFE722C	16
	Port pin read register_J0	JPPR0	16	H'FCFE7B20	16
	Port mode control register_J0	JPMC0	16	H'FCFE7B40	16
	Port mode control set/reset register_J0	JPMCSR0	32	H'FCFE7B90	32
	Port input buffer control register_J0	JPIBC0	16	H'FCFE7F00	16
	Serial sound interface noise canceler control register	SNCR	32	H'FCFE3C00	32
Power-down modes	Standby control register 1	STBCR1	8	H'FCFE0020	8
	Standby control register 2	STBCR2	8	H'FCFE0024	8
	Standby control register 3	STBCR3	8	H'FCFE0420	8
	Standby control register 4	STBCR4	8	H'FCFE0424	8
	Standby control register 5	STBCR5	8	H'FCFE0428	8
	Standby control register 6	STBCR6	8	H'FCFE042C	8
	Standby control register 7	STBCR7	8	H'FCFE0430	8
	Standby control register 8	STBCR8	8	H'FCFE0434	8
	Standby control register 9	STBCR9	8	H'FCFE0438	8
	Standby control register 10	STBCR10	8	H'FCFE043C	8
	Standby control register 11	STBCR11	8	H'FCFE0440	8
	Standby control register 12	STBCR12	8	H'FCFE0444	8
	Standby control register 13	STBCR13	8	H'FCFE0470	8
	Software reset control register 1	SWRSTCR1	8	H'FCFE0460	8
	Software reset control register 2	SWRSTCR2	8	H'FCFE0464	8
	Software reset control register 3	SWRSTCR3	8	H'FCFE0468	8
	System control register 1	SYSCR1	8	H'FCFE0400	8
	System control register 2	SYSCR2	8	H'FCFE0404	8
	System control register 3	SYSCR3	8	H'FCFE0408	8
	CPU status register	CPUSTS	8	H'FCFE0018	8
	Standby request register 1	STBREQ1	8	H'FCFE0030	8
	Standby request register 2	STBREQ2	8	H'FCFE0034	8
	Standby acknowledge register 1	STBACK1	8	H'FCFE0040	8
	Standby acknowledge register 2	STBACK2	8	H'FCFE0044	8
	On-chip data-retention RAM area setting register	RRAMKP	8	H'FCFF1800	8
	Deep standby control register	DSCTR	8	H'FCFF1802	8
	Deep standby cancel source select register	DSSSR	16	H'FCFF1804	16
	Deep standby cancel edge select register	DSESR	16	H'FCFF1806	16
	Deep standby cancel source flag register	DSFR	16	H'FCFF1808	16
	XTAL crystal oscillator gain control register	XTALCTR	8	H'FCFF1810	8

**Table 57.1 Register Addresses**

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Debugger interface	DAPROM Peripheral ID4 Register	DAPROM_PERIPHID4	32	H'FC000FD0	32
	DAPROM Peripheral ID0 Register	DAPROM_PERIPHID0	32	H'FC000FE0	32
	DAPROM Peripheral ID1 Register	DAPROM_PERIPHID1	32	H'FC000FE4	32
	DAPROM Peripheral ID2 Register	DAPROM_PERIPHID2	32	H'FC000FE8	32
	DAPROM Peripheral ID3 Register	DAPROM_PERIPHID3	32	H'FC000FEC	32
	DAPROM Component ID0 Register	DAPROM_COMPID0	32	H'FC000FF0	32
	DAPROM Component ID1 Register	DAPROM_COMPID1	32	H'FC000FF4	32
	DAPROM Component ID2 Register	DAPROM_COMPID2	32	H'FC000FF8	32
	DAPROM Component ID3 Register	DAPROM_COMPID3	32	H'FC000FFC	32
	Mode reset control register	ICEREGMDRSTCTL	32	H'FC00F000	32
	JTAG trace select register	ICEREGJTTRCSEL	32	H'FC00F004	32
	Clock power control register	ICEREGCLKPWRCTRL	32	H'FC00F014	32
	Lock access register	ICEREGLOCKACCES	32	H'FC00FFB0	32
	2ndDAPROM Peripheral ID4 Register	2ndDAPROM_PERIPHID4	32	H'FC020FD0	32
	2ndDAPROM Peripheral ID0 Register	2ndDAPROM_PERIPHID0	32	H'FC020FE0	32
	2ndDAPROM Peripheral ID1 Register	2ndDAPROM_PERIPHID1	32	H'FC020FE4	32
	2ndDAPROM Peripheral ID2 Register	2ndDAPROM_PERIPHID2	32	H'FC020FE8	32
	2ndDAPROM Peripheral ID3 Register	2ndDAPROM_PERIPHID3	32	H'FC020FEC	32
	2ndDAPROM Component ID0 Register	2ndDAPROM_COMPID0	32	H'FC020FF0	32
	2ndDAPROM Component ID1 Register	2ndDAPROM_COMPID1	32	H'FC020FF4	32
	2ndDAPROM Component ID2 Register	2ndDAPROM_COMPID2	32	H'FC020FF8	32
	2ndDAPROM Component ID3 Register	2ndDAPROM_COMPID3	32	H'FC020FFC	32
	CPU-ETF RAM Size Register	CPU_ETF_RSZ	32	H'FC021004	32

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Debugger interface	CPU-ETF Status Register	CPU_ETF_STS	32	H'FC02100C	32
	CPU-ETF RAM Read Data Register	CPU_ETF_RRD	32	H'FC021010	32
	CPU-ETF RAM Read Pointer Register	CPU_ETF_RRP	32	H'FC021014	32
	CPU-ETF RAM Write Pointer Register	CPU_ETF_RWP	32	H'FC021018	32
	CPU-ETF Trigger Counter Register	CPU_ETF_TRG	32	H'FC02101C	32
	CPU-ETF Control Register	CPU_ETF_CTL	32	H'FC021020	32
	CPU-ETF RAM Write Data Register	CPU_ETF_RWD	32	H'FC021024	32
	CPU-ETF Mode Register	CPU_ETF_MODE	32	H'FC021028	32
	CPU-ETF Latched Buffer Fill Level Register	CPU_ETF_LBUFLEVEL	32	H'FC02102C	32
	CPU-ETF Current Buffer Fill Level Register	CPU_ETF_CBUFLEVEL	32	H'FC021030	32
	CPU-ETF Buffer Level Water Mark Register	CPU_ETF_BUFWM	32	H'FC021034	32
	CPU-ETF RAM Read Pointer High Register	CPU_ETF_RRPHI	32	H'FC021038	32
	CPU-ETF RAM Write Pointer High Register	CPU_ETF_RWPHI	32	H'FC02103C	32
	CPU-ETF Formatter and Flush Status Register	CPU_ETF_FF SR	32	H'FC021300	32
	CPU-ETF Formatter and Flush Control Register	CPU_ETF_FF CR	32	H'FC021304	32
	CPU-ETF Periodic Synchronization Counter Register	CPU_ETF_PSCR	32	H'FC021308	32
	CPU-ETF Claim Tag Set Register	CPU_ETF_CLAIMSET	32	H'FC021FA0	32
	CPU-ETF Claim Tag Clear Register	CPU_ETF_CLAIMCLR	32	H'FC021FA4	32
	CPU-ETF Lock Access Register	CPU_ETF_LAR	32	H'FC021FB0	32
	CPU-ETF Lock Status Register	CPU_ETF_LSR	32	H'FC021FB4	32
	CPU-ETF Authentication Status Register	CPU_ETF_AUTHSTATUS	32	H'FC021FB8	32
	CPU-ETF Device Configuration Register	CPU_ETF_DEVID	32	H'FC021FC8	32
	CPU-ETF Device Type Identifier Register	CPU_ETF_DEVTYPE	32	H'FC021FCC	32
	CPU-ETF Peripheral ID4 Register	CPU_ETF_PERIPHID4	32	H'FC021FD0	32
	CPU-ETF Peripheral ID0 Register	CPU_ETF_PERIPHID0	32	H'FC021FE0	32
	CPU-ETF Peripheral ID1 Register	CPU_ETF_PERIPHID1	32	H'FC021FE4	32
	CPU-ETF Peripheral ID2 Register	CPU_ETF_PERIPHID2	32	H'FC021FE8	32
	CPU-ETF Peripheral ID3 Register	CPU_ETF_PERIPHID3	32	H'FC021FEC	32
	CPU-ETF Component ID0 Register	CPU_ETF_COMPID0	32	H'FC021FF0	32
	CPU-ETF Component ID1 Register	CPU_ETF_COMPID1	32	H'FC021FF4	32
	CPU-ETF Component ID2 Register	CPU_ETF_COMPID2	32	H'FC021FF8	32
	CPU-ETF Component ID3 Register	CPU_ETF_COMPID3	32	H'FC021FFC	32
	CPU-CTICS CTI Control Register	CPU_CTICS_CTICONTROL	32	H'FC022000	32
	CPU-CTICS CTI Interrupt Acknowledge Register	CPU_CTICS_CTIINTACK	32	H'FC022010	32
	CPU-CTICS CTI Application Trigger Set Register	CPU_CTICS_CTIAPPSET	32	H'FC022014	32
	CPU-CTICS CTI Application Trigger Clear Register	CPU_CTICS_CTIAPPCLEAR	32	H'FC022018	32
	CPU-CTICS CTI Application Pulse Register	CPU_CTICS_CTIAPPULSE	32	H'FC02201C	32
	CPU-CTICS CTI Trigger to Channel Enable Register0	CPU_CTICS_CTIINEN0	32	H'FC022020	32
	CPU-CTICS CTI Trigger to Channel Enable Register1	CPU_CTICS_CTIINEN1	32	H'FC022024	32
	CPU-CTICS CTI Trigger to Channel Enable Register2	CPU_CTICS_CTIINEN2	32	H'FC022028	32
	CPU-CTICS CTI Trigger to Channel Enable Register3	CPU_CTICS_CTIINEN3	32	H'FC02202C	32
	CPU-CTICS CTI Trigger to Channel Enable Register4	CPU_CTICS_CTIINEN4	32	H'FC022030	32
	CPU-CTICS CTI Trigger to Channel Enable Register5	CPU_CTICS_CTIINEN5	32	H'FC022034	32
	CPU-CTICS CTI Trigger to Channel Enable Register6	CPU_CTICS_CTIINEN6	32	H'FC022038	32
	CPU-CTICS CTI Trigger to Channel Enable Register7	CPU_CTICS_CTIINEN7	32	H'FC02203C	32
	CPU-CTICS CTI Channel to Trigger Enable Register0	CPU_CTICS_CTIOUTEN0	32	H'FC0220A0	32
	CPU-CTICS CTI Channel to Trigger Enable Register1	CPU_CTICS_CTIOUTEN1	32	H'FC0220A4	32
	CPU-CTICS CTI Channel to Trigger Enable Register2	CPU_CTICS_CTIOUTEN2	32	H'FC0220A8	32
	CPU-CTICS CTI Channel to Trigger Enable Register3	CPU_CTICS_CTIOUTEN3	32	H'FC0220AC	32
	CPU-CTICS CTI Channel to Trigger Enable Register4	CPU_CTICS_CTIOUTEN4	32	H'FC0220B0	32
CPU-CTICS CTI Channel to Trigger Enable Register5	CPU_CTICS_CTIOUTEN5	32	H'FC0220B4	32	
CPU-CTICS CTI Channel to Trigger Enable Register6	CPU_CTICS_CTIOUTEN6	32	H'FC0220B8	32	

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Debugger interface	CPU-CTICS CTI Channel to Trigger Enable Register7	CPU_CTICS_CTIOUTEN7	32	H'FC0220BC	32
	CPU-CTICS CTI Trigger In Status Register	CPU_CTICS_CTITRIGINSTATUS	32	H'FC022130	32
	CPU-CTICS CTI Trigger Out Status Register	CPU_CTICS_CTITRIGOUTSTATUS	32	H'FC022134	32
	CPU-CTICS CTI Channel In Status Register	CPU_CTICS_CTICHINSTATUS	32	H'FC022138	32
	CPU-CTICS CTI Channel Out Status Register	CPU_CTICS_CTICHOUTSTATUS	32	H'FC02213C	32
	CPU-CTICS Enable CTI Channel Gate Register	CPU_CTICS_CTIGATE	32	H'FC022140	32
	CPU-CTICS External Multiplexor Control Register	CPU_CTICS_ASICCTL	32	H'FC022144	32
	CPU-CTICS Claim Tag Set Register	CPU_CTICS_CLAIMSET	32	H'FC022FA0	32
	CPU-CTICS Claim Tag Clear Register	CPU_CTICS_CLAIMCLR	32	H'FC022FA4	32
	CPU-CTICS Lock Access Register	CPU_CTICS_LAR	32	H'FC022FB0	32
	CPU-CTICS Lock Status Register	CPU_CTICS_LSR	32	H'FC022FB4	32
	CPU-CTICS Authentication Status Register	CPU_CTICS_AUTHSTATUS	32	H'FC022FB8	32
	CPU-CTICS Device Configuration Register	CPU_CTICS_DEVID	32	H'FC022FC8	32
	CPU-CTICS Device Type Identifier Register	CPU_CTICS_DEVTYPE	32	H'FC022FCC	32
	CPU-CTICS Peripheral ID4 Register	CPU_CTICS_PERIPHID4	32	H'FC022FD0	32
	CPU-CTICS Peripheral ID0 Register	CPU_CTICS_PERIPHID0	32	H'FC022FE0	32
	CPU-CTICS Peripheral ID1 Register	CPU_CTICS_PERIPHID1	32	H'FC022FE4	32
	CPU-CTICS Peripheral ID2 Register	CPU_CTICS_PERIPHID2	32	H'FC022FE8	32
	CPU-CTICS Peripheral ID3 Register	CPU_CTICS_PERIPHID3	32	H'FC022FEC	32
	CPU-CTICS Component ID0 Register	CPU_CTICS_COMPID0	32	H'FC022FF0	32
	CPU-CTICS Component ID1 Register	CPU_CTICS_COMPID1	32	H'FC022FF4	32
	CPU-CTICS Component ID2 Register	CPU_CTICS_COMPID2	32	H'FC022FF8	32
	CPU-CTICS Component ID3 Register	CPU_CTICS_COMPID3	32	H'FC022FFC	32
	CPU-TPIU Supported Port Size Register	CPU_TPIU_Supported port sizes	32	H'FC023000	32
	CPU-TPIU Current Port Size Register	CPU_TPIU_Current port size	32	H'FC023004	32
	CPU-TPIU Trigger Modes Register	CPU_TPIU_Supported trigger modes	32	H'FC023100	32
	CPU-TPIU Trigger Counter Register	CPU_TPIU_Trigger counter value	32	H'FC023104	32
	CPU-TPIU Trigger Multiplier Register	CPU_TPIU_Trigger multiplier	32	H'FC023108	32
	CPU-TPIU Supported Test Patterns/Modes Register	CPU_TPIU_Supported test pattern/modes	32	H'FC023200	32
	CPU-TPIU Current Test Patterns/Modes Register	CPU_TPIU_Current test pattern/mode	32	H'FC023204	32
	CPU-TPIU TPIU Test Pattern Repeat Register	CPU_TPIU_Test pattern repeat counter	32	H'FC023208	32
	CPU-TPIU Formatter and Flush Status Register	CPU_TPIU_Formatter and flush status	32	H'FC023300	32
	CPU-TPIU Formatter and Flush Control Register	CPU_TPIU_Formatter and flush control	32	H'FC023304	32
	CPU-TPIU Formatter Synchronization counter Register	CPU_TPIU_Formatter synchronization counter	32	H'FC023308	32
	CPU-TPIU Claim Tag Set Register	CPU_TPIU_CLAIMSET	32	H'FC023FA0	32
	CPU-TPIU Claim Tag Clear Register	CPU_TPIU_CLAIMCLR	32	H'FC023FA4	32
	CPU-TPIU Lock Access Register	CPU_TPIU_LAR	32	H'FC023FB0	32
	CPU-TPIU Lock Status Register	CPU_TPIU_LSR	32	H'FC023FB4	32
	CPU-TPIU Authentication Status Register	CPU_TPIU_AUTHSTATUS	32	H'FC023FB8	32
	CPU-TPIU Device Configuration Register	CPU_TPIU_DEVID	32	H'FC023FC8	32
	CPU-TPIU Device Type Identifier Register	CPU_TPIU_DEVTYPE	32	H'FC023FCC	32
	CPU-TPIU Peripheral ID4 Register	CPU_TPIU_PERIPHID4	32	H'FC023FD0	32
CPU-TPIU Peripheral ID0 Register	CPU_TPIU_PERIPHID0	32	H'FC023FE0	32	
CPU-TPIU Peripheral ID1 Register	CPU_TPIU_PERIPHID1	32	H'FC023FE4	32	
CPU-TPIU Peripheral ID2 Register	CPU_TPIU_PERIPHID2	32	H'FC023FE8	32	

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Debugger interface	CPU-TPIU Peripheral ID3 Register	CPU_TPIU_PERIPHID3	32	H'FC023FEC	32
	CPU-TPIU Component ID0 Register	CPU_TPIU_COMPID0	32	H'FC023FF0	32
	CPU-TPIU Component ID1 Register	CPU_TPIU_COMPID1	32	H'FC023FF4	32
	CPU-TPIU Component ID2 Register	CPU_TPIU_COMPID2	32	H'FC023FF8	32
	CPU-TPIU Component ID3 Register	CPU_TPIU_COMPID3	32	H'FC023FFC	32
	CPU-TraceFunnel CSTF Control Register	CPU_TraceFunnel_FUNCCTL	32	H'FC024000	32
	CPU-TraceFunnel CSTF Priority Control Register	CPU_TraceFunnel_PRICTL	32	H'FC024004	32
	CPU-TraceFunnel Claim Tag Set Register	CPU_TraceFunnel_CLAIMSET	32	H'FC024FA0	32
	CPU-TraceFunnel Claim Tag Clear Register	CPU_TraceFunnel_CLAIMCLR	32	H'FC024FA4	32
	CPU-TraceFunnel Lock Access Register	CPU_TraceFunnel_LAR	32	H'FC024FB0	32
	CPU-TraceFunnel Lock Status Register	CPU_TraceFunnel_LSR	32	H'FC024FB4	32
	CPU-TraceFunnel Authentication Status Register	CPU_TraceFunnel_AUTHSTATUS	32	H'FC024FB8	32
	CPU-TraceFunnel Device Configuration Register	CPU_TraceFunnel_DEVID	32	H'FC024FC8	32
	CPU-TraceFunnel Device Type Identifier Register	CPU_TraceFunnel_DEVTYPE	32	H'FC024FCC	32
	CPU-TraceFunnel Peripheral ID4 Register	CPU_TraceFunnel_PERIPHID4	32	H'FC024FD0	32
	CPU-TraceFunnel Peripheral ID0 Register	CPU_TraceFunnel_PERIPHID0	32	H'FC024FE0	32
	CPU-TraceFunnel Peripheral ID1 Register	CPU_TraceFunnel_PERIPHID1	32	H'FC024FE4	32
	CPU-TraceFunnel Peripheral ID2 Register	CPU_TraceFunnel_PERIPHID2	32	H'FC024FE8	32
	CPU-TraceFunnel Peripheral ID3 Register	CPU_TraceFunnel_PERIPHID3	32	H'FC024FEC	32
	CPU-TraceFunnel Component ID0 Register	CPU_TraceFunnel_COMPID0	32	H'FC024FF0	32
	CPU-TraceFunnel Component ID1 Register	CPU_TraceFunnel_COMPID1	32	H'FC024FF4	32
	CPU-TraceFunnel Component ID2 Register	CPU_TraceFunnel_COMPID2	32	H'FC024FF8	32
	CPU-TraceFunnel Component ID3 Register	CPU_TraceFunnel_COMPID3	32	H'FC024FFC	32
	CA9-DBG Debug ID Register	CA9_DBG_DBGIDR	32	H'FC030000	32
	CA9-DBG Watchpoint Fault Address Register	CA9_DBG_DBGWFAR	32	H'FC030018	32
	CA9-DBG Vector Catch Register	CA9_DBG_DBGVCR	32	H'FC03001C	32
	CA9-DBG Host to Target Data Transfer Register	CA9_DBG_DBGDTRRText	32	H'FC030080	32
	CA9-DBG Instruction Transfer/Program Counter Sampling Register	CA9_DBG_DBGITR/DBGPCSR	32	H'FC030084	32
	CA9-DBG Debug Status and Control Register	CA9_DBG_DBGDSCRText	32	H'FC030088	32
	CA9-DBG Target to Host Data Transfer Register	CA9_DBG_DBGDTRTText	32	H'FC03008C	32
	CA9-DBG Debug Run Control Register	CA9_DBG_DBGDRCR	32	H'FC030090	32
	CA9-DBG Breakpoint Value Register 0	CA9_DBG_DBGBVR0	32	H'FC030100	32
	CA9-DBG Breakpoint Value Register 1	CA9_DBG_DBGBVR1	32	H'FC030104	32
	CA9-DBG Breakpoint Value Register 2	CA9_DBG_DBGBVR2	32	H'FC030108	32
	CA9-DBG Breakpoint Value Register 3	CA9_DBG_DBGBVR3	32	H'FC03010C	32
	CA9-DBG Breakpoint Value Register 4	CA9_DBG_DBGBVR4	32	H'FC030110	32
	CA9-DBG Breakpoint Value Register 5	CA9_DBG_DBGBVR5	32	H'FC030114	32
	CA9-DBG Breakpoint Control Register 0	CA9_DBG_DBGBCR0	32	H'FC030140	32
	CA9-DBG Breakpoint Control Register 1	CA9_DBG_DBGBCR1	32	H'FC030144	32
	CA9-DBG Breakpoint Control Register 2	CA9_DBG_DBGBCR2	32	H'FC030148	32
	CA9-DBG Breakpoint Control Register 3	CA9_DBG_DBGBCR3	32	H'FC03014C	32
	CA9-DBG Breakpoint Control Register 4	CA9_DBG_DBGBCR4	32	H'FC030150	32
	CA9-DBG Breakpoint Control Register 5	CA9_DBG_DBGBCR5	32	H'FC030154	32
	CA9-DBG Watchpoint Value Register 0	CA9_DBG_DBGWVR0	32	H'FC030180	32
	CA9-DBG Watchpoint Value Register 1	CA9_DBG_DBGWVR1	32	H'FC030184	32
	CA9-DBG Watchpoint Value Register 2	CA9_DBG_DBGWVR2	32	H'FC030188	32
CA9-DBG Watchpoint Value Register 3	CA9_DBG_DBGWVR3	32	H'FC03018C	32	
CA9-DBG Watchpoint Control Register 0	CA9_DBG_DBGWCR0	32	H'FC0301C0	32	

**Table 57.1 Register Addresses**

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Debugger interface	CA9-DBG Watchpoint Control Register 1	CA9_DBG_DBGWCR1	32	H'FC0301C4	32
	CA9-DBG Watchpoint Control Register 2	CA9_DBG_DBGWCR2	32	H'FC0301C8	32
	CA9-DBG Watchpoint Control Register 3	CA9_DBG_DBGWCR3	32	H'FC0301CC	32
	CA9-DBG Main ID Register	CA9_DBG_MIDR	32	H'FC030D00	32
	CA9-DBG Cache Type Register	CA9_DBG_CTR	32	H'FC030D04	32
	CA9-DBG TLB Type Register	CA9_DBG_TLBTR	32	H'FC030D0C	32
	CA9-DBG Multiprocessor Affinity Register	CA9_DBG_MPIDR	32	H'FC030D10	32
	CA9-DBG Revision ID register	CA9_DBG_REVIDR	32	H'FC030D14	32
	CA9-DBG Processor Feature Register 0	CA9_DBG_ID_PFR0	32	H'FC030D20	32
	CA9-DBG Processor Feature Register 1	CA9_DBG_ID_PFR1	32	H'FC030D24	32
	CA9-DBG Debug Feature Register 0	CA9_DBG_ID_DFR0	32	H'FC030D28	32
	CA9-DBG Memory Model Feature Register 0	CA9_DBG_ID_MMFR0	32	H'FC030D30	32
	CA9-DBG Memory Model Feature Register 1	CA9_DBG_ID_MMFR1	32	H'FC030D34	32
	CA9-DBG Memory Model Feature Register 2	CA9_DBG_ID_MMFR2	32	H'FC030D38	32
	CA9-DBG Memory Model Feature Register 3	CA9_DBG_ID_MMFR3	32	H'FC030D3C	32
	CA9-DBG Instruction Set Attribute Register 0	CA9_DBG_ID_ISAR0	32	H'FC030D40	32
	CA9-DBG Instruction Set Attribute Register 1	CA9_DBG_ID_ISAR1	32	H'FC030D44	32
	CA9-DBG Instruction Set Attribute Register 2	CA9_DBG_ID_ISAR2	32	H'FC030D48	32
	CA9-DBG Instruction Set Attribute Register 3	CA9_DBG_ID_ISAR3	32	H'FC030D4C	32
	CA9-DBG Instruction Set Attribute Register 4	CA9_DBG_ID_ISAR4	32	H'FC030D50	32
	CA9-DBG Claim Tag Set Register	CA9_DBG_CLAIMSET	32	H'FC030FA0	32
	CA9-DBG Claim Tag Clear Register	CA9_DBG_CLAIMCLR	32	H'FC030FA4	32
	CA9-DBG Lock Access Register	CA9_DBG_LAR	32	H'FC030FB0	32
	CA9-DBG Lock Status Register	CA9_DBG_LSR	32	H'FC030FB4	32
	CA9-DBG Authentication Status Register	CA9_DBG_AUTHSTATUS	32	H'FC030FB8	32
	CA9-DBG Device Configuration Register	CA9_DBG_DEVID	32	H'FC030FC8	32
	CA9-DBG Device Type Identifier Register	CA9_DBG_DEVTYPE	32	H'FC030FCC	32
	CA9-DBG Peripheral ID4 Register	CA9_DBG_PERIPHID4	32	H'FC030FD0	32
	CA9-DBG Peripheral ID0 Register	CA9_DBG_PERIPHID0	32	H'FC030FE0	32
	CA9-DBG Peripheral ID1 Register	CA9_DBG_PERIPHID1	32	H'FC030FE4	32
	CA9-DBG Peripheral ID2 Register	CA9_DBG_PERIPHID2	32	H'FC030FE8	32
	CA9-DBG Peripheral ID3 Register	CA9_DBG_PERIPHID3	32	H'FC030FEC	32
	CA9-DBG Component ID0 Register	CA9_DBG_COMPID0	32	H'FC030FF0	32
	CA9-DBG Component ID1 Register	CA9_DBG_COMPID1	32	H'FC030FF4	32
	CA9-DBG Component ID2 Register	CA9_DBG_COMPID2	32	H'FC030FF8	32
	CA9-DBG Component ID3 Register	CA9_DBG_COMPID3	32	H'FC030FFC	32
	CA9-PMU Event Count Register 0	CA9_PMU_PMXEVCNTR0	32	H'FC031000	32
	CA9-PMU Event Count Register 1	CA9_PMU_PMXEVCNTR1	32	H'FC031004	32
	CA9-PMU Event Count Register 2	CA9_PMU_PMXEVCNTR2	32	H'FC031008	32
	CA9-PMU Event Count Register 3	CA9_PMU_PMXEVCNTR3	32	H'FC03100C	32
	CA9-PMU Event Count Register 4	CA9_PMU_PMXEVCNTR4	32	H'FC031010	32
	CA9-PMU Event Count Register 5	CA9_PMU_PMXEVCNTR5	32	H'FC031014	32
	CA9-PMU Cycle Count Register	CA9_PMU_PMCCNTR	32	H'FC03107C	32
	CA9-PMU Event Counter Selection Register 0	CA9_PMU_PMXEVTYPER0	32	H'FC031400	32
CA9-PMU Event Counter Selection Register 1	CA9_PMU_PMXEVTYPER1	32	H'FC031404	32	
CA9-PMU Event Counter Selection Register 2	CA9_PMU_PMXEVTYPER2	32	H'FC031408	32	
CA9-PMU Event Counter Selection Register 3	CA9_PMU_PMXEVTYPER3	32	H'FC03140C	32	
CA9-PMU Event Counter Selection Register 4	CA9_PMU_PMXEVTYPER4	32	H'FC031410	32	
CA9-PMU Event Counter Selection Register 5	CA9_PMU_PMXEVTYPER5	32	H'FC031414	32	
CA9-PMU Count Enable Set Register	CA9_PMU_PMCNTENSET	32	H'FC031C00	32	
CA9-PMU Count Enable Clear Register	CA9_PMU_PMCNTENCLR	32	H'FC031C20	32	
CA9-PMU Interrupt Enable Set Register	CA9_PMU_PMINTENSET	32	H'FC031C40	32	



Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Debugger interface	CA9-PMU Interrupt Enable Clear Register	CA9_PMU_PMINTENCLR	32	H'FC031C60	32
	CA9-PMU Overflow Flag Status Register	CA9_PMU_PMOVSR	32	H'FC031C80	32
	CA9-PMU Software Increment Register	CA9_PMU_PMSWINC	32	H'FC031CA0	32
	CA9-PMU Performance Monitor Control Register	CA9_PMU_PMCR	32	H'FC031E04	32
	CA9-PMU User Enable Register	CA9_PMU_PMUSERENR	32	H'FC031E08	32
	CA9-PMU Claim Tag Set Register	CA9_PMU_CLAIMSET	32	H'FC031FA0	32
	CA9-PMU Claim Tag Clear Register	CA9_PMU_CLAIMCLR	32	H'FC031FA4	32
	CA9-PMU Lock Access Register	CA9_PMU_LAR	32	H'FC031FB0	32
	CA9-PMU Lock Status Register	CA9_PMU_LSR	32	H'FC031FB4	32
	CA9-PMU Authentication Status Register	CA9_PMU_AUTHSTATUS	32	H'FC031FB8	32
	CA9-PMU Device Configuration Register	CA9_PMU_DEVID	32	H'FC031FC8	32
	CA9-PMU Device Type Identifier Register	CA9_PMU_DEVTYPE	32	H'FC031FCC	32
	CA9-PMU Peripheral ID4 Register	CA9_PMU_PERIPHID4	32	H'FC031FD0	32
	CA9-PMU Peripheral ID0 Register	CA9_PMU_PERIPHID0	32	H'FC031FE0	32
	CA9-PMU Peripheral ID1 Register	CA9_PMU_PERIPHID1	32	H'FC031FE4	32
	CA9-PMU Peripheral ID2 Register	CA9_PMU_PERIPHID2	32	H'FC031FE8	32
	CA9-PMU Peripheral ID3 Register	CA9_PMU_PERIPHID3	32	H'FC031FEC	32
	CA9-PMU Component ID0 Register	CA9_PMU_COMPID0	32	H'FC031FF0	32
	CA9-PMU Component ID1 Register	CA9_PMU_COMPID1	32	H'FC031FF4	32
	CA9-PMU Component ID2 Register	CA9_PMU_COMPID2	32	H'FC031FF8	32
	CA9-PMU Component ID3 Register	CA9_PMU_COMPID3	32	H'FC031FFC	32
	CA9-CTI CTI Control Register	CA9_CTI_CTICONTROL	32	H'FC038000	32
	CA9-CTI CTI Interrupt Acknowledge Register	CA9_CTI_CTIINTACK	32	H'FC038010	32
	CA9-CTI CTI Application Trigger Set Register	CA9_CTI_CTIAPPSET	32	H'FC038014	32
	CA9-CTI CTI Application Trigger Clear Register	CA9_CTI_CTIAPPCLEAR	32	H'FC038018	32
	CA9-CTI CTI Application Pulse Register	CA9_CTI_CTIAPPULSE	32	H'FC03801C	32
	CA9-CTI CTI Trigger to Channel Enable Register0	CA9_CTI_CTIINEN0	32	H'FC038020	32
	CA9-CTI CTI Trigger to Channel Enable Register1	CA9_CTI_CTIINEN1	32	H'FC038024	32
	CA9-CTI CTI Trigger to Channel Enable Register2	CA9_CTI_CTIINEN2	32	H'FC038028	32
	CA9-CTI CTI Trigger to Channel Enable Register3	CA9_CTI_CTIINEN3	32	H'FC03802C	32
	CA9-CTI CTI Trigger to Channel Enable Register4	CA9_CTI_CTIINEN4	32	H'FC038030	32
	CA9-CTI CTI Trigger to Channel Enable Register5	CA9_CTI_CTIINEN5	32	H'FC038034	32
	CA9-CTI CTI Trigger to Channel Enable Register6	CA9_CTI_CTIINEN6	32	H'FC038038	32
	CA9-CTI CTI Trigger to Channel Enable Register7	CA9_CTI_CTIINEN7	32	H'FC03803C	32
	CA9-CTI CTI Channel to Trigger Enable Register0	CA9_CTI_CTIOUTEN0	32	H'FC0380A0	32
	CA9-CTI CTI Channel to Trigger Enable Register1	CA9_CTI_CTIOUTEN1	32	H'FC0380A4	32
	CA9-CTI CTI Channel to Trigger Enable Register2	CA9_CTI_CTIOUTEN2	32	H'FC0380A8	32
	CA9-CTI CTI Channel to Trigger Enable Register3	CA9_CTI_CTIOUTEN3	32	H'FC0380AC	32
	CA9-CTI CTI Channel to Trigger Enable Register4	CA9_CTI_CTIOUTEN4	32	H'FC0380B0	32
	CA9-CTI CTI Channel to Trigger Enable Register5	CA9_CTI_CTIOUTEN5	32	H'FC0380B4	32
	CA9-CTI CTI Channel to Trigger Enable Register6	CA9_CTI_CTIOUTEN6	32	H'FC0380B8	32
	CA9-CTI CTI Channel to Trigger Enable Register7	CA9_CTI_CTIOUTEN7	32	H'FC0380BC	32
	CA9-CTI CTI Trigger In Status Register	CA9_CTI_CTIINTRIGINSTATUS	32	H'FC038130	32
	CA9-CTI CTI Trigger Out Status Register	CA9_CTI_CTIINTRIGOUTSTATUS	32	H'FC038134	32
	CA9-CTI CTI Channel In Status Register	CA9_CTI_CTIICHINSTATUS	32	H'FC038138	32
	CA9-CTI CTI Channel Out Status Register	CA9_CTI_CTIICHOUTSTATUS	32	H'FC03813C	32
	CA9-CTI Enable CTI Channel Gate Register	CA9_CTI_CTIIGATE	32	H'FC038140	32
	CA9-CTI External Multiplexor Control Register	CA9_CTI_ASICCTL	32	H'FC038144	32
	CA9-CTI Claim Tag Set Register	CA9_CTI_CLAIMSET	32	H'FC038FA0	32
	CA9-CTI Claim Tag Clear Register	CA9_CTI_CLAIMCLR	32	H'FC038FA4	32
CA9-CTI Lock Access Register	CA9_CTI_LAR	32	H'FC038FB0	32	
CA9-CTI Lock Status Register	CA9_CTI_LSR	32	H'FC038FB4	32	

Table 57.1 Register Addresses

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Debugger interface	CA9-CTI Authentication Status Register	CA9_CTI_AUTHSTATUS	32	H'FC038FB8	32
	CA9-CTI Device Configuration Register	CA9_CTI_DEVID	32	H'FC038FC8	32
	CA9-CTI Device Type Identifier Register	CA9_CTI_DEVTYPE	32	H'FC038FCC	32
	CA9-CTI Peripheral ID4 Register	CA9_CTI_PERIPHID4	32	H'FC038FD0	32
	CA9-CTI Peripheral ID0 Register	CA9_CTI_PERIPHID0	32	H'FC038FE0	32
	CA9-CTI Peripheral ID1 Register	CA9_CTI_PERIPHID1	32	H'FC038FE4	32
	CA9-CTI Peripheral ID2 Register	CA9_CTI_PERIPHID2	32	H'FC038FE8	32
	CA9-CTI Peripheral ID3 Register	CA9_CTI_PERIPHID3	32	H'FC038FEC	32
	CA9-CTI Component ID0 Register	CA9_CTI_COMPID0	32	H'FC038FF0	32
	CA9-CTI Component ID1 Register	CA9_CTI_COMPID1	32	H'FC038FF4	32
	CA9-CTI Component ID2 Register	CA9_CTI_COMPID2	32	H'FC038FF8	32
	CA9-CTI Component ID3 Register	CA9_CTI_COMPID3	32	H'FC038FFC	32
	PTM-A9 Main Control Register	PTM_A9_ETMCR	32	H'FC03C000	32
	PTM-A9 Configuration Code Register	PTM_A9_ETMCCR	32	H'FC03C004	32
	PTM-A9 Trigger Event Register	PTM_A9_ETMTRIGGER	32	H'FC03C008	32
	PTM-A9 Status Register	PTM_A9_ETMSR	32	H'FC03C010	32
	PTM-A9 System Configuration Register	PTM_A9_ETMSCR	32	H'FC03C014	32
	PTM-A9 TraceEnable Start/Stop Control Register	PTM_A9_ETMTSSCR	32	H'FC03C018	32
	PTM-A9 TraceEnable Event Register	PTM_A9_ETMTSEVR	32	H'FC03C020	32
	PTM-A9 TraceEnable Control Register 1	PTM_A9_ETMTECR1	32	H'FC03C024	32
	PTM-A9 Address Comparator Value Register 1	PTM_A9_ETMACVR1	32	H'FC03C040	32
	PTM-A9 Address Comparator Value Register 2	PTM_A9_ETMACVR2	32	H'FC03C044	32
	PTM-A9 Address Comparator Value Register 3	PTM_A9_ETMACVR3	32	H'FC03C048	32
	PTM-A9 Address Comparator Value Register 4	PTM_A9_ETMACVR4	32	H'FC03C04C	32
	PTM-A9 Address Comparator Value Register 5	PTM_A9_ETMACVR5	32	H'FC03C050	32
	PTM-A9 Address Comparator Value Register 6	PTM_A9_ETMACVR6	32	H'FC03C054	32
	PTM-A9 Address Comparator Value Register 7	PTM_A9_ETMACVR7	32	H'FC03C058	32
	PTM-A9 Address Comparator Value Register 8	PTM_A9_ETMACVR8	32	H'FC03C05C	32
	PTM-A9 Address Comparator Access Type Register 1	PTM_A9_ETMACTR1	32	H'FC03C080	32
	PTM-A9 Address Comparator Access Type Register 2	PTM_A9_ETMACTR2	32	H'FC03C084	32
	PTM-A9 Address Comparator Access Type Register 3	PTM_A9_ETMACTR3	32	H'FC03C088	32
	PTM-A9 Address Comparator Access Type Register 4	PTM_A9_ETMACTR4	32	H'FC03C08C	32
	PTM-A9 Address Comparator Access Type Register 5	PTM_A9_ETMACTR5	32	H'FC03C090	32
	PTM-A9 Address Comparator Access Type Register 6	PTM_A9_ETMACTR6	32	H'FC03C094	32
	PTM-A9 Address Comparator Access Type Register 7	PTM_A9_ETMACTR7	32	H'FC03C098	32
	PTM-A9 Address Comparator Access Type Register 8	PTM_A9_ETMACTR8	32	H'FC03C09C	32
	PTM-A9 Counter Reload Value Register 1	PTM_A9_ETMCNTRLDVR1	32	H'FC03C140	32
	PTM-A9 Counter Reload Value Register 2	PTM_A9_ETMCNTRLDVR2	32	H'FC03C144	32
	PTM-A9 Counter Enable Event Register 1	PTM_A9_ETMCNTENR1	32	H'FC03C150	32
	PTM-A9 Counter Enable Event Register 2	PTM_A9_ETMCNTENR2	32	H'FC03C154	32
	PTM-A9 Counter Reload Event Register 1	PTM_A9_ETMCNTRLDEV1	32	H'FC03C160	32
	PTM-A9 Counter Reload Event Register 2	PTM_A9_ETMCNTRLDEV2	32	H'FC03C164	32
	PTM-A9 Counter Value Register 1	PTM_A9_ETMCNTVR1	32	H'FC03C170	32
	PTM-A9 Counter Value Register 2	PTM_A9_ETMCNTVR2	32	H'FC03C174	32
	PTM-A9 State 1 to State 2 Transition Event Register	PTM_A9_ETMSQ12EVR	32	H'FC03C180	32
	PTM-A9 State 2 to State 1 Transition Event Register	PTM_A9_ETMSQ21EVR	32	H'FC03C184	32
	PTM-A9 State 2 to State 3 Transition Event Register	PTM_A9_ETMSQ23EVR	32	H'FC03C188	32
	PTM-A9 State 3 to State 1 Transition Event Register	PTM_A9_ETMSQ31EVR	32	H'FC03C18C	32
	PTM-A9 State 3 to State 2 Transition Event Register	PTM_A9_ETMSQ32EVR	32	H'FC03C190	32
	PTM-A9 State 1 to State 3 Transition Event Register	PTM_A9_ETMSQ13EVR	32	H'FC03C194	32
PTM-A9 Current Sequencer State Register	PTM_A9_ETMSQR	32	H'FC03C19C	32	
PTM-A9 External Output Event Register 1	PTM_A9_ETMEXTOUTEVR1	32	H'FC03C1A0	32	

**Table 57.1 Register Addresses**

Module	Register Name	Abbreviation	Number of Bits	Address	Access Size
Debugger interface	PTM-A9 External Output Event Register 2	PTM_A9_ETMEXTOUTEVR2	32	H'FC03C1A4	32
	PTM-A9 Context ID Comparator Value 1 Register	PTM_A9_ETMCIDCVR1	32	H'FC03C1B0	32
	PTM-A9 Context ID Comparator Mask Register	PTM_A9_ETMCIDCMR	32	H'FC03C1BC	32
	PTM-A9 Synchronization Frequency Register	PTM_A9_ETMSYNCFR	32	H'FC03C1E0	32
	PTM-A9 ID Register	PTM_A9_ETMIDR	32	H'FC03C1E4	32
	PTM-A9 Configuration Code Extension Register	PTM_A9_ETMCCER	32	H'FC03C1E8	32
	PTM-A9 Extended External Input Selection Register	PTM_A9_ETMEXTINSELR	32	H'FC03C1EC	32
	PTM-A9 Timestamp Event Register	PTM_A9_ETMTSEVR	32	H'FC03C1F8	32
	PTM-A9 Auxiliary Control Register	PTM_A9_ETMAUXCR	32	H'FC03C1FC	32
	PTM-A9 CoreSight Trace ID Register	PTM_A9_ETMTRACEIDR	32	H'FC03C200	32
	PTM-A9 OS Lock Status Register	PTM_A9_OSLSR	32	H'FC03C304	32
	PTM-A9 Claim Tag Set Register	PTM_A9_CLAIMSET	32	H'FC031FA0	32
	PTM-A9 Claim Tag Clear Register	PTM_A9_CLAIMCLR	32	H'FC031FA4	32
	PTM-A9 Lock Access Register	PTM_A9_LAR	32	H'FC031FB0	32
	PTM-A9 Lock Status Register	PTM_A9_LSR	32	H'FC031FB4	32
	PTM-A9 Authentication Status Register	PTM_A9_AUTHSTATUS	32	H'FC031FB8	32
	PTM-A9 Device Type Identifier Register	PTM_A9_DEVTTYPE	32	H'FC031FCC	32
	PTM-A9 Peripheral ID4 Register	PTM_A9_PERIPID4	32	H'FC031FD0	32
	PTM-A9 Peripheral ID0 Register	PTM_A9_PERIPID0	32	H'FC031FE0	32
	PTM-A9 Peripheral ID1 Register	PTM_A9_PERIPID1	32	H'FC031FE4	32
	PTM-A9 Peripheral ID2 Register	PTM_A9_PERIPID2	32	H'FC031FE8	32
	PTM-A9 Peripheral ID3 Register	PTM_A9_PERIPID3	32	H'FC031FEC	32
	PTM-A9 Component ID0 Register	PTM_A9_COMPID0	32	H'FC031FF0	32
	PTM-A9 Component ID1 Register	PTM_A9_COMPID1	32	H'FC031FF4	32
	PTM-A9 Component ID2 Register	PTM_A9_COMPID2	32	H'FC031FF8	32
	PTM-A9 Component ID3 Register	PTM_A9_COMPID3	32	H'FC031FFC	32
	BSID register	BSID	32	H'FCFE1804	32

## 57.2 Register Bits

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Secondary cache	reg0_cache_id	Implementer[7]	Implementer[6]	Implementer[5]	Implementer[4]	Implementer[3]	Implementer[2]	Implementer[1]	Implementer[0]
		-	-	-	-	-	-	-	-
		CACHE ID[5]	CACHE ID[4]	CACHE ID[3]	CACHE ID[2]	CACHE ID[1]	CACHE ID[0]	Part Number[3]	Part Number[2]
	reg0_cache_type	Part Number[1]	Part Number[0]	RTL release[5]	RTL release[4]	RTL release[3]	RTL release[2]	RTL release[1]	RTL release[0]
		Data banking	-	-	ctype[3]	ctype[2]	ctype[1]	ctype[0]	-
		Dsize[4]	Dsize[3]	Dsize[2]	Dsize[1]	Dsize[0]	L2 associativity	-	-
	reg1_control	-	-	L2 cache line length[1]	L2 cache line length[0]	lsize[4]	lsize[3]	lsize[2]	lsize[1]
		-	-	-	-	-	-	L2 cache line length[1]	L2 cache line length[0]
		-	-	-	-	-	-	-	-
	reg1_aux_control	-	Early BRESP enable	Instruction prefetch enable	Data prefetch enable	Non-secure interrupt access control	Non-secure lockdown enable	Cache replacement policy	Force write allocate[1]
		Force write allocate[0]	Shared attribute override enable	Parity enable	Event monitor bus enable	Way-size[2]	Way-size[1]	Way-size[0]	Associativity
		-	-	Shared Attribute Invalidate Enable	Exclusive cache configuration	Store buffer device limitation Enable	High Priority for SO and Dev Reads Enable	-	-
		-	-	-	-	-	-	-	Full Line of Zero Enable
	reg1_tag_ram_control	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		0	RAM read access latency[2]	RAM read access latency[1]	RAM read access latency[0]	0	RAM write access latency[2]	RAM write access latency[1]	RAM write access latency[0]
	reg1_data_ram_control	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		0	RAM read access latency[2]	RAM read access latency[1]	RAM read access latency[0]	0	RAM write access latency[2]	RAM write access latency[1]	RAM write access latency[0]
	reg2_ev_counter_ctrl	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	Counter reset[1]	Counter reset[0]	Event counter enable
	reg2_ev_counter_cfg	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	Counter event source[3]	Counter event source[2]	Counter event source[1]	Counter event source[0]	Event counter interrupt generation[1]	Event counter interrupt generation[0]
	reg2_ev_counter0_cfg	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	Counter event source[3]	Counter event source[2]	Counter event source[1]	Counter event source[0]	Event counter interrupt generation[1]	Event counter interrupt generation[0]
	reg2_ev_counter1	Counter value[31]	Counter value[30]	Counter value[29]	Counter value[28]	Counter value[27]	Counter value[26]	Counter value[25]	Counter value[24]
		Counter value[23]	Counter value[22]	Counter value[21]	Counter value[20]	Counter value[19]	Counter value[18]	Counter value[17]	Counter value[16]
		Counter value[15]	Counter value[14]	Counter value[13]	Counter value[12]	Counter value[11]	Counter value[10]	Counter value[9]	Counter value[8]
		Counter value[7]	Counter value[6]	Counter value[5]	Counter value[4]	Counter value[3]	Counter value[2]	Counter value[1]	Counter value[0]
	reg2_ev_counter0	Counter value[31]	Counter value[30]	Counter value[29]	Counter value[28]	Counter value[27]	Counter value[26]	Counter value[25]	Counter value[24]
		Counter value[23]	Counter value[22]	Counter value[21]	Counter value[20]	Counter value[19]	Counter value[18]	Counter value[17]	Counter value[16]
		Counter value[15]	Counter value[14]	Counter value[13]	Counter value[12]	Counter value[11]	Counter value[10]	Counter value[9]	Counter value[8]
		Counter value[7]	Counter value[6]	Counter value[5]	Counter value[4]	Counter value[3]	Counter value[2]	Counter value[1]	Counter value[0]
	reg2_int_mask	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	DECERR
		SLVERR	ERRRD	ERRRT	ERRWD	ERRWT	PARRD	PARRT	ECNTR

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Secondary cache	reg2_int_mask_status	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	DECERR
		SLVERR	ERRRD	ERRRT	ERRWD	ERRWT	PARRD	PARRT	ECNTR
	reg2_int_raw_status	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	DECERR
		SLVERR	ERRRD	ERRRT	ERRWD	ERRWT	PARRD	PARRT	ECNTR
	reg2_int_clear	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	DECERR
		SLVERR	ERRRD	ERRRT	ERRWD	ERRWT	PARRD	PARRT	ECNTR
	reg7_cache_sync	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	C
	reg7_inv_pa	TAG[17]	TAG[16]	TAG[15]	TAG[14]	TAG[13]	TAG[12]	TAG[11]	TAG[10]
		TAG[9]	TAG[8]	TAG[7]	TAG[6]	TAG[5]	TAG[4]	TAG[3]	TAG[2]
		TAG[1]	TAG[0]	INDEX[8]	INDEX[7]	INDEX[6]	INDEX[5]	INDEX[4]	INDEX[3]
		INDEX[2]	INDEX[1]	INDEX[0]	-	-	-	-	C
	reg7_inv_way	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Way bits[7]	Way bits[6]	Way bits[5]	Way bits[4]	Way bits[3]	Way bits[2]	Way bits[1]	Way bits[0]
	reg7_clean_pa	TAG[17]	TAG[16]	TAG[15]	TAG[14]	TAG[13]	TAG[12]	TAG[11]	TAG[10]
		TAG[9]	TAG[8]	TAG[7]	TAG[6]	TAG[5]	TAG[4]	TAG[3]	TAG[2]
		TAG[1]	TAG[0]	INDEX[8]	INDEX[7]	INDEX[6]	INDEX[5]	INDEX[4]	INDEX[3]
		INDEX[2]	INDEX[1]	INDEX[0]	-	-	-	-	C
	reg7_clean_index	-	Way[2]	Way[1]	Way[0]	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	INDEX[8]	INDEX[7]	INDEX[6]	INDEX[5]	INDEX[4]	INDEX[3]
		INDEX[2]	INDEX[1]	INDEX[0]	-	-	-	-	C
	reg7_clean_way	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Way bits[7]	Way bits[6]	Way bits[5]	Way bits[4]	Way bits[3]	Way bits[2]	Way bits[1]	Way bits[0]
	reg7_clean_inv_pa	TAG[17]	TAG[16]	TAG[15]	TAG[14]	TAG[13]	TAG[12]	TAG[11]	TAG[10]
		TAG[9]	TAG[8]	TAG[7]	TAG[6]	TAG[5]	TAG[4]	TAG[3]	TAG[2]
		TAG[1]	TAG[0]	INDEX[8]	INDEX[7]	INDEX[6]	INDEX[5]	INDEX[4]	INDEX[3]
		INDEX[2]	INDEX[1]	INDEX[0]	-	-	-	-	C
	reg7_clean_inv_index	-	Way[2]	Way[1]	Way[0]	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	INDEX[8]	INDEX[7]	INDEX[6]	INDEX[5]	INDEX[4]	INDEX[3]
		INDEX[2]	INDEX[1]	INDEX[0]	-	-	-	-	C
	reg7_clean_inv_way	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Way_bits[7]	Way_bits[6]	Way_bits[5]	Way_bits[4]	Way_bits[3]	Way_bits[2]	Way_bits[1]	Way_bits[0]
	reg9_d_lockdown0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		DATALOCK000[7]	DATALOCK000[6]	DATALOCK000[5]	DATALOCK000[4]	DATALOCK000[3]	DATALOCK000[2]	DATALOCK000[1]	DATALOCK000[0]
	reg9_i_lockdown0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		INSTRLOCK000 [7]	INSTRLOCK000 [6]	INSTRLOCK000 [5]	INSTRLOCK000 [4]	INSTRLOCK000 [3]	INSTRLOCK000 [2]	INSTRLOCK000 [1]	INSTRLOCK000 [0]

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Secondary cache	reg9_d_lockdown1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		DATALOCK001[7]	DATALOCK001[6]	DATALOCK001[5]	DATALOCK001[4]	DATALOCK001[3]	DATALOCK001[2]	DATALOCK001[1]	DATALOCK001[0]
	reg9_i_lockdown1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		INSTRLOCK001 [7]	INSTRLOCK001 [6]	INSTRLOCK001 [5]	INSTRLOCK001 [4]	INSTRLOCK001 [3]	INSTRLOCK001 [2]	INSTRLOCK001 [1]	INSTRLOCK001 [0]
	reg9_d_lockdown2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		DATALOCK002[7]	DATALOCK002[6]	DATALOCK002[5]	DATALOCK002[4]	DATALOCK002[3]	DATALOCK002[2]	DATALOCK002[1]	DATALOCK002[0]
	reg9_i_lockdown2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		INSTRLOCK002 [7]	INSTRLOCK002 [6]	INSTRLOCK002 [5]	INSTRLOCK002 [4]	INSTRLOCK002 [3]	INSTRLOCK002 [2]	INSTRLOCK002 [1]	INSTRLOCK002 [0]
	reg9_d_lockdown3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		DATALOCK003[7]	DATALOCK003[6]	DATALOCK003[5]	DATALOCK003[4]	DATALOCK003[3]	DATALOCK003[2]	DATALOCK003[1]	DATALOCK003[0]
	reg9_i_lockdown3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		INSTRLOCK003 [7]	INSTRLOCK003 [6]	INSTRLOCK003 [5]	INSTRLOCK003 [4]	INSTRLOCK003 [3]	INSTRLOCK003 [2]	INSTRLOCK003 [1]	INSTRLOCK003 [0]
	reg9_d_lockdown4	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		DATALOCK004[7]	DATALOCK004[6]	DATALOCK004[5]	DATALOCK004[4]	DATALOCK004[3]	DATALOCK004[2]	DATALOCK004[1]	DATALOCK004[0]
	reg9_i_lockdown4	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		INSTRLOCK004 [7]	INSTRLOCK004 [6]	INSTRLOCK004 [5]	INSTRLOCK004 [4]	INSTRLOCK004 [3]	INSTRLOCK004 [2]	INSTRLOCK004 [1]	INSTRLOCK004 [0]
	reg9_d_lockdown5	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		DATALOCK005[7]	DATALOCK005[6]	DATALOCK005[5]	DATALOCK005[4]	DATALOCK005[3]	DATALOCK005[2]	DATALOCK005[1]	DATALOCK005[0]
	reg9_i_lockdown5	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		INSTRLOCK005 [7]	INSTRLOCK005 [6]	INSTRLOCK005 [5]	INSTRLOCK005 [4]	INSTRLOCK005 [3]	INSTRLOCK005 [2]	INSTRLOCK005 [1]	INSTRLOCK005 [0]
	reg9_d_lockdown6	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		DATALOCK006[7]	DATALOCK006[6]	DATALOCK006[5]	DATALOCK006[4]	DATALOCK006[3]	DATALOCK006[2]	DATALOCK006[1]	DATALOCK006[0]
	reg9_i_lockdown6	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		INSTRLOCK006 [7]	INSTRLOCK006 [6]	INSTRLOCK006 [5]	INSTRLOCK006 [4]	INSTRLOCK006 [3]	INSTRLOCK006 [2]	INSTRLOCK006 [1]	INSTRLOCK006 [0]
	reg9_d_lockdown7	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		DATALOCK007[7]	DATALOCK007[6]	DATALOCK007[5]	DATALOCK007[4]	DATALOCK007[3]	DATALOCK007[2]	DATALOCK007[1]	DATALOCK007[0]

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Secondary cache	reg9_i_lockdown7	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		INSTRLOCK007[7]	INSTRLOCK007[6]	INSTRLOCK007[5]	INSTRLOCK007[4]	INSTRLOCK007[3]	INSTRLOCK007[2]	INSTRLOCK007[1]	INSTRLOCK007[0]	
	reg9_lock_line_en	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	lockdown_by_line_enable	
	reg9_unlock_way	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		unlock_all_lines_by_way_operation[7]	unlock_all_lines_by_way_operation[6]	unlock_all_lines_by_way_operation[5]	unlock_all_lines_by_way_operation[4]	unlock_all_lines_by_way_operation[3]	unlock_all_lines_by_way_operation[2]	unlock_all_lines_by_way_operation[1]	unlock_all_lines_by_way_operation[0]	
	reg12_addr_filtering_start	address_filtering_start[11]	address_filtering_start[10]	address_filtering_start[9]	address_filtering_start[8]	address_filtering_start[7]	address_filtering_start[6]	address_filtering_start[5]	address_filtering_start[4]	
		address_filtering_start[3]	address_filtering_start[2]	address_filtering_start[1]	address_filtering_start[0]	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	address_filtering_enable	
	reg12_addr_filtering_end	address_filtering_end[11]	address_filtering_end[10]	address_filtering_end[9]	address_filtering_end[8]	address_filtering_end[7]	address_filtering_end[6]	address_filtering_end[5]	address_filtering_end[4]	
		address_filtering_end[3]	address_filtering_end[2]	address_filtering_end[1]	address_filtering_end[0]	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
	reg15_debug_ctrl	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	SPNIDEN	DWB	DCL	
	reg15_prefetch_ctrl	-	Double linefill enable	Instruction prefetch enable	Data prefetch enable	Double linefill on WRAP read disable	-	-	Prefetch drop enable	
		Incr double Linefill enable	-	Not same ID on exclusive sequence enable	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	Prefetch offset[4]	Prefetch offset[3]	Prefetch offset[2]	Prefetch offset[1]	Prefetch offset[0]	
	reg15_power_ctrl	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	dynamic_clk_gating_en	standby_mode_en	
	LSI internal bus	RMPR	-	-	-	-	-	-	-	
			-	-	-	-	-	-	-	
			-	-	-	-	-	-	-	
			-	-	-	-	-	-	AXI128	-
		AXIBUSCTL0	-	-	-	-	JCUARCACHE[3]	JCUARCACHE[2]	JCUARCACHE[1]	JCUARCACHE[0]
			-	-	-	-	JCUAWCACHE[3]	JCUAWCACHE[2]	JCUAWCACHE[1]	JCUAWCACHE[0]
			-	-	-	-	ETHARCACHE[3]	ETHARCACHE[2]	ETHARCACHE[1]	ETHARCACHE[0]
			-	-	-	-	ETHAWCACHE[3]	ETHAWCACHE[2]	ETHAWCACHE[1]	ETHAWCACHE[0]
		AXIBUSCTL1	-	-	-	-	IMR20ARCACHE[3]	IMR20ARCACHE[2]	IMR20ARCACHE[1]	IMR20ARCACHE[0]
			-	-	-	-	IMR20AWCACHE[3]	IMR20AWCACHE[2]	IMR20AWCACHE[1]	IMR20AWCACHE[0]
-			-	-	-	IMR21ARCACHE[3]	IMR21ARCACHE[2]	IMR21ARCACHE[1]	IMR21ARCACHE[0]	
-			-	-	-	IMR21AWCACHE[3]	IMR21AWCACHE[2]	IMR21AWCACHE[1]	IMR21AWCACHE[0]	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
LSI internal bus	AXIBUSCTL2	-	-	-	-	IMRDARCACHE [3]	IMRDARCACHE [2]	IMRDARCACHE [1]	IMRDARCACHE [0]
		-	-	-	-	IMRDAWCACHE [3]	IMRDAWCACHE [2]	IMRDAWCACHE [1]	IMRDAWCACHE [0]
		-	-	-	-	-	-	-	-
		-	-	-	-	CEUAWCACHE [3]	CEUAWCACHE [2]	CEUAWCACHE [1]	CEUAWCACHE [0]
	AXIBUSCTL3	-	-	-	-	RGP640AR CACHE[3]	RGP640AR CACHE[2]	RGP640AR CACHE[1]	RGP640AR CACHE[0]
		-	-	-	-	-	-	-	-
		-	-	-	-	RGP641AR CACHE[3]	RGP641AR CACHE[2]	RGP641AR CACHE[1]	RGP641AR CACHE[0]
		-	-	-	-	RGP641AW CACHE[3]	RGP641AW CACHE[2]	RGP641AW CACHE[1]	RGP641AW CACHE[0]
	AXIBUSCTL4	-	-	-	-	RGP642AR CACHE[3]	RGP642AR CACHE[2]	RGP642AR CACHE[1]	RGP642AR CACHE[0]
		-	-	-	-	RGP642AW CACHE[3]	RGP642AW CACHE[2]	RGP642AW CACHE[1]	RGP642AW CACHE[0]
		-	-	-	-	RGP1280AR CACHE[3]	RGP1280AR CACHE[2]	RGP1280AR CACHE[1]	RGP1280AR CACHE[0]
		-	-	-	-	-	-	-	-
	AXIBUSCTL5	-	-	-	-	RGP1281AR CACHE[3]	RGP1281AR CACHE[2]	RGP1281AR CACHE[1]	RGP1281AR CACHE[0]
		-	-	-	-	RGP1281AW CACHE[3]	RGP1281AW CACHE[2]	RGP1281AW CACHE[1]	RGP1281AW CACHE[0]
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	MLBAXCACHE[1]	MLBAXCACHE[0]
	AXIBUSCTL6	-	-	-	-	VDC501AR CACHE[3]	VDC501AR CACHE[2]	VDC501AR CACHE[1]	VDC501AR CACHE[0]
		-	-	-	-	VDC501AW CACHE[3]	VDC501AW CACHE[2]	VDC501AW CACHE[1]	VDC501AW CACHE[0]
		-	-	-	-	VDC502AR CACHE[3]	VDC502AR CACHE[2]	VDC502AR CACHE[1]	VDC502AR CACHE[0]
		-	-	-	-	-	-	-	-
	AXIBUSCTL7	-	-	-	-	VDC503AR CACHE[3]	VDC503AR CACHE[2]	VDC503AR CACHE[1]	VDC503AR CACHE[0]
		-	-	-	-	VDC503AW CACHE[3]	VDC503AW CACHE[2]	VDC503AW CACHE[1]	VDC503AW CACHE[0]
		-	-	-	-	VDC504AR CACHE[3]	VDC504AR CACHE[2]	VDC504AR CACHE[1]	VDC504AR CACHE[0]
		-	-	-	-	-	-	-	-
	AXIBUSCTL8	-	-	-	-	VDC505AR CACHE[3]	VDC505AR CACHE[2]	VDC505AR CACHE[1]	VDC505AR CACHE[0]
		-	-	-	-	VDC505AW CACHE[3]	VDC505AW CACHE[2]	VDC505AW CACHE[1]	VDC505AW CACHE[0]
		-	-	-	-	VDC511AR CACHE[3]	VDC511AR CACHE[2]	VDC511AR CACHE[1]	VDC511AR CACHE[0]
		-	-	-	-	VDC511AW CACHE[3]	VDC511AW CACHE[2]	VDC511AW CACHE[1]	VDC511AW CACHE[0]
	AXIBUSCTL9	-	-	-	-	VDC512AR CACHE[3]	VDC512AR CACHE[2]	VDC512AR CACHE[1]	VDC512AR CACHE[0]
		-	-	-	-	-	-	-	-
		-	-	-	-	VDC513AR CACHE[3]	VDC513AR CACHE[2]	VDC513AR CACHE[1]	VDC513AR CACHE[0]
		-	-	-	-	VDC513AW CACHE[3]	VDC513AW CACHE[2]	VDC513AW CACHE[1]	VDC513AW CACHE[0]
	AXIBUSCTL10	-	-	-	-	VDC514AR CACHE[3]	VDC514AR CACHE[2]	VDC514AR CACHE[1]	VDC514AR CACHE[0]
		-	-	-	-	-	-	-	-
		-	-	-	-	VDC515AR CACHE[3]	VDC515AR CACHE[2]	VDC515AR CACHE[1]	VDC515AR CACHE[0]
		-	-	-	-	VDC515AW CACHE[3]	VDC515AW CACHE[2]	VDC515AW CACHE[1]	VDC515AW CACHE[0]
	AXIRERRCTL0	-	-	-	JCURERREN	-	-	-	ETHRERREN
		-	-	-	IMR20RERREN	-	-	-	IMR21RERREN
		-	-	-	IMRDREERREN	-	-	-	CEURERREN
		-	-	-	-	-	-	-	-



Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
LSI internal bus	AXIRERRCTL1	-	-	-	RGP640RERREN	-	-	-	RGP641RERREN
		-	-	-	RGP642RERREN	-	-	-	RGP1280RERREN
		-	-	-	RGP1281RERREN	-	-	-	-
		-	-	-	-	-	-	-	-
	AXIRERRCTL2	-	-	-	VDC501RERREN	-	-	-	VDC502RERREN
		-	-	-	VDC503RERREN	-	-	-	VDC504RERREN
		-	-	-	VDC505RERREN	-	-	-	-
		-	-	-	-	-	-	-	-
	AXIRERRCTL3	-	-	-	VDC511RERREN	-	-	-	VDC512RERREN
		-	-	-	VDC513RERREN	-	-	-	VDC514RERREN
		-	-	-	VDC515RERREN	-	-	-	-
		-	-	-	-	-	-	-	-
	AXIRERRST0	JCURRESP[1]	JCURRESP[0]	JCUBRESP[1]	JCUBRESP[0]	ETHRESP[1]	ETHRESP[0]	ETHBRESP[1]	ETHBRESP[0]
		IMR20RRESP[1]	IMR20RRESP[0]	IMR20BRESP[1]	IMR20BRESP[0]	IMR21RRESP[1]	IMR21RRESP[0]	IMR21BRESP[1]	IMR21BRESP[0]
		IMRDRRESP[1]	IMRDRRESP[0]	IMRDBRESP[1]	IMRDBRESP[0]	-	-	CEUBRESP[1]	CEUBRESP[0]
		-	-	-	-	-	-	-	-
	AXIRERRST1	RGP640RRESP [1]	RGP640RRESP [0]	-	-	RGP641RRESP [1]	RGP641RRESP [0]	RGP641BRESP [1]	RGP641BRESP [0]
		RGP642RRESP [1]	RGP642RRESP [0]	RGP642BRESP [1]	RGP642BRESP [0]	RGP1280RRESP [1]	RGP1280RRESP [0]	-	-
		RGP1281RRESP [1]	RGP1281RRESP [0]	RGP1281BRESP [1]	RGP1281BRESP [0]	-	-	-	-
		-	-	-	-	-	-	-	-
	AXIRERRST2	VDC501RRESP [1]	VDC501RRESP [0]	VDC501BRESP [1]	VDC501BRESP [0]	VDC502RRESP [1]	VDC502RRESP [0]	-	-
		VDC503RRESP [1]	VDC503RRESP [0]	VDC503BRESP [1]	VDC503BRESP [0]	VDC504RRESP [1]	VDC504RRESP [0]	-	-
		VDC505RRESP [1]	VDC505RRESP [0]	VDC505BRESP [1]	VDC505BRESP [0]	-	-	-	-
		-	-	-	-	-	-	-	-
	AXIRERRST3	VDC511RRESP [1]	VDC511RRESP [0]	VDC511BRESP [1]	VDC511BRESP [0]	VDC512RRESP [1]	VDC512RRESP [0]	-	-
		VDC513RRESP [1]	VDC513RRESP [0]	VDC513BRESP [1]	VDC513BRESP [0]	VDC514RRESP [1]	VDC514RRESP [0]	-	-
		VDC515RRESP [1]	VDC515RRESP [0]	VDC515BRESP [1]	VDC515BRESP [0]	-	-	-	-
		-	-	-	-	-	-	-	-
	AXIRERRCLR0	-	JCURRESPCLR	-	JCUBRESPCLR	-	ETHRESPCLR	-	ETHBRESPCLR
		-	IMR20RRESP CLR	-	IMR20BRESP CLR	-	IMR21RRESP CLR	-	IMR21BRESP CLR
		-	IMRDRRESPCLR	-	IMRDBRESPCLR	-	-	-	CEUBRESPCLR
		-	-	-	-	-	-	-	-
	AXIRERRCLR1	-	RGP640RRESP CLR	-	-	-	RGP641RRESP CLR	-	RGP641BRESP CLR
		-	RGP642RRESP CLR	-	RGP642BRESP CLR	-	RGP1280RRESPCLR	-	-
		-	RGP1281RRESPCLR	-	RGP1281BRESPCLR	-	-	-	-
		-	-	-	-	-	-	-	-
	AXIRERRCLR2	-	VDC501RRESP CLR	-	VDC501BRESP CLR	-	VDC502RRESP CLR	-	-
		-	VDC503RRESP CLR	-	VDC503BRESP CLR	-	VDC504RRESP CLR	-	-
		-	VDC505RRESP CLR	-	VDC505BRESP CLR	-	-	-	-
		-	-	-	-	-	-	-	-
	AXIRERRCLR3	-	VDC511RRESP CLR	-	VDC511BRESP CLR	-	VDC512RRESP CLR	-	-
		-	VDC513RRESP CLR	-	VDC513BRESP CLR	-	VDC514RRESP CLR	-	-
		-	VDC515RRESP CLR	-	VDC515BRESP CLR	-	-	-	-
		-	-	-	-	-	-	-	-
Clock pulse generator	FRQCR	-	CKOEN2	CKOEN[1]	CKOEN[0]	-	-	IFC[1]	IFC[0]
		-	-	-	-	-	-	-	-
	FRQCR2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	GFC[1]	GFC[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Interrupt controller	ICR0	NMIL	-	-	-	-	-	-	NMIE
		-	-	-	-	-	-	-	-
	ICR1	IRQ71S	IRQ70S	IRQ61S	IRQ60S	IRQ51S	IRQ50S	IRQ41S	IRQ40S
		IRQ31S	IRQ30S	IRQ21S	IRQ20S	IRQ11S	IRQ10S	IRQ01S	IRQ00S
	IRQRR	-	-	-	-	-	-	-	-
		IRQ7F	IRQ6F	IRQ5F	IRQ4F	IRQ3F	IRQ2F	IRQ1F	IRQ0F
	ICDDCR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	Enable
	ICDICTR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		LSP[4]	LSP[3]	LSP[2]	LSP[1]	LSP[0]	SecurityExtn	-	-
	ICDIIDR	CPUNumber[2]	CPUNumber[1]	CPUNumber[0]	ITLinesNumber[4]	ITLinesNumber[3]	ITLinesNumber[2]	ITLinesNumber[1]	ITLinesNumber[0]
		ProductID[7]	ProductID[6]	ProductID[5]	ProductID[4]	ProductID[3]	ProductID[2]	ProductID[1]	ProductID[0]
		-	-	-	-	Variant[3]	Variant[2]	Variant[1]	Variant[0]
		Revision[3]	Revision[2]	Revision[1]	Revision[0]	Implementer[11]	Implementer[10]	Implementer[9]	Implementer[8]
	ICDISERn n = 0 to 18	Implementer[7]	Implementer[6]	Implementer[5]	Implementer[4]	Implementer[3]	Implementer[2]	Implementer[1]	Implementer[0]
		Security status bits[31]	Security status bits[30]	Security status bits[29]	Security status bits[28]	Security status bits[27]	Security status bits[26]	Security status bits[25]	Security status bits[24]
		Security status bits[23]	Security status bits[22]	Security status bits[21]	Security status bits[20]	Security status bits[19]	Security status bits[18]	Security status bits[17]	Security status bits[16]
		Security status bits[15]	Security status bits[14]	Security status bits[13]	Security status bits[12]	Security status bits[11]	Security status bits[10]	Security status bits[9]	Security status bits[8]
	ICDISERn n = 0 to 18	Security status bits[7]	Security status bits[6]	Security status bits[5]	Security status bits[4]	Security status bits[3]	Security status bits[2]	Security status bits[1]	Security status bits[0]
		Set-enable bits[31]	Set-enable bits[30]	Set-enable bits[29]	Set-enable bits[28]	Set-enable bits[27]	Set-enable bits[26]	Set-enable bits[25]	Set-enable bits[24]
		Set-enable bits[23]	Set-enable bits[22]	Set-enable bits[21]	Set-enable bits[20]	Set-enable bits[19]	Set-enable bits[18]	Set-enable bits[17]	Set-enable bits[16]
		Set-enable bits[15]	Set-enable bits[14]	Set-enable bits[13]	Set-enable bits[12]	Set-enable bits[11]	Set-enable bits[10]	Set-enable bits[9]	Set-enable bits[8]
	ICDICERn n = 0 to 18	Set-enable bits[7]	Set-enable bits[6]	Set-enable bits[5]	Set-enable bits[4]	Set-enable bits[3]	Set-enable bits[2]	Set-enable bits[1]	Set-enable bits[0]
		Clear-enable bits[31]	Clear-enable bits[30]	Clear-enable bits[29]	Clear-enable bits[28]	Clear-enable bits[27]	Clear-enable bits[26]	Clear-enable bits[25]	Clear-enable bits[24]
		Clear-enable bits[23]	Clear-enable bits[22]	Clear-enable bits[21]	Clear-enable bits[20]	Clear-enable bits[19]	Clear-enable bits[18]	Clear-enable bits[17]	Clear-enable bits[16]
		Clear-enable bits[15]	Clear-enable bits[14]	Clear-enable bits[13]	Clear-enable bits[12]	Clear-enable bits[11]	Clear-enable bits[10]	Clear-enable bits[9]	Clear-enable bits[8]
	ICDISPRn n = 0 to 18	Clear-enable bits[7]	Clear-enable bits[6]	Clear-enable bits[5]	Clear-enable bits[4]	Clear-enable bits[3]	Clear-enable bits[2]	Clear-enable bits[1]	Clear-enable bits[0]
		Set-pending bits[31]	Set-pending bits[30]	Set-pending bits[29]	Set-pending bits[28]	Set-pending bits[27]	Set-pending bits[26]	Set-pending bits[25]	Set-pending bits[24]
		Set-pending bits[23]	Set-pending bits[22]	Set-pending bits[21]	Set-pending bits[20]	Set-pending bits[19]	Set-pending bits[18]	Set-pending bits[17]	Set-pending bits[16]
		Set-pending bits[15]	Set-pending bits[14]	Set-pending bits[13]	Set-pending bits[12]	Set-pending bits[11]	Set-pending bits[10]	Set-pending bits[9]	Set-pending bits[8]
	ICDICPRn n = 0 to 18	Set-pending bits[7]	Set-pending bits[6]	Set-pending bits[5]	Set-pending bits[4]	Set-pending bits[3]	Set-pending bits[2]	Set-pending bits[1]	Set-pending bits[0]
		Clear-pending bits[31]	Clear-pending bits[30]	Clear-pending bits[29]	Clear-pending bits[28]	Clear-pending bits[27]	Clear-pending bits[26]	Clear-pending bits[25]	Clear-pending bits[24]
		Clear-pending bits[23]	Clear-pending bits[22]	Clear-pending bits[21]	Clear-pending bits[20]	Clear-pending bits[19]	Clear-pending bits[18]	Clear-pending bits[17]	Clear-pending bits[16]
		Clear-pending bits[15]	Clear-pending bits[14]	Clear-pending bits[13]	Clear-pending bits[12]	Clear-pending bits[11]	Clear-pending bits[10]	Clear-pending bits[9]	Clear-pending bits[8]
	ICDICPRn n = 0 to 18	Clear-pending bits[7]	Clear-pending bits[6]	Clear-pending bits[5]	Clear-pending bits[4]	Clear-pending bits[3]	Clear-pending bits[2]	Clear-pending bits[1]	Clear-pending bits[0]

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Interrupt controller	ICDABRn n = 0 to 18	Active bits[31]	Active bits[30]	Active bits[29]	Active bits[28]	Active bits[27]	Active bits[26]	Active bits[25]	Active bits[24]
		Active bits[23]	Active bits[22]	Active bits[21]	Active bits[20]	Active bits[19]	Active bits[18]	Active bits[17]	Active bits[16]
		Active bits[15]	Active bits[14]	Active bits[13]	Active bits[12]	Active bits[11]	Active bits[10]	Active bits[9]	Active bits[8]
		Active bits[7]	Active bits[6]	Active bits[5]	Active bits[4]	Active bits[3]	Active bits[2]	Active bits[1]	Active bits[0]
	ICDIPRn n = 0 to 146	Priority, byte offset 3[7]	Priority, byte offset 3[6]	Priority, byte offset 3[5]	Priority, byte offset 3[4]	Priority, byte offset 3[3]	Priority, byte offset 3[2]	Priority, byte offset 3[1]	Priority, byte offset 3[0]
		Priority, byte offset 2[7]	Priority, byte offset 2[6]	Priority, byte offset 2[5]	Priority, byte offset 2[4]	Priority, byte offset 2[3]	Priority, byte offset 2[2]	Priority, byte offset 2[1]	Priority, byte offset 2[0]
		Priority, byte offset 1[7]	Priority, byte offset 1[6]	Priority, byte offset 1[5]	Priority, byte offset 1[4]	Priority, byte offset 1[3]	Priority, byte offset 1[2]	Priority, byte offset 1[1]	Priority, byte offset 1[0]
		Priority, byte offset 0[7]	Priority, byte offset 0[6]	Priority, byte offset 0[5]	Priority, byte offset 0[4]	Priority, byte offset 0[3]	Priority, byte offset 0[2]	Priority, byte offset 0[1]	Priority, byte offset 0[0]
	ICDIPTRn n = 0 to 146	CPU targets, byte offset 3[7]	CPU targets, byte offset 3[6]	CPU targets, byte offset 3[5]	CPU targets, byte offset 3[4]	CPU targets, byte offset 3[3]	CPU targets, byte offset 3[2]	CPU targets, byte offset 3[1]	CPU targets, byte offset 3[0]
		CPU targets, byte offset 2[7]	CPU targets, byte offset 2[6]	CPU targets, byte offset 2[5]	CPU targets, byte offset 2[4]	CPU targets, byte offset 2[3]	CPU targets, byte offset 2[2]	CPU targets, byte offset 2[1]	CPU targets, byte offset 2[0]
		CPU targets, byte offset 1[7]	CPU targets, byte offset 1[6]	CPU targets, byte offset 1[5]	CPU targets, byte offset 1[4]	CPU targets, byte offset 1[3]	CPU targets, byte offset 1[2]	CPU targets, byte offset 1[1]	CPU targets, byte offset 1[0]
		CPU targets, byte offset 0[7]	CPU targets, byte offset 0[6]	CPU targets, byte offset 0[5]	CPU targets, byte offset 0[4]	CPU targets, byte offset 0[3]	CPU targets, byte offset 0[2]	CPU targets, byte offset 0[1]	CPU targets, byte offset 0[0]
	ICDIFFRn n = 0 to 36	Int_config[1], field 15	Int_config[0], field 15	Int_config[1], field 14	Int_config[0], field 14	Int_config[1], field 13	Int_config[0], field 13	Int_config[1], field 12	Int_config[0], field 12
		Int_config[1], field 11	Int_config[0], field 11	Int_config[1], field 10	Int_config[0], field 10	Int_config[1], field 9	Int_config[0], field 9	Int_config[1], field 8	Int_config[0], field 8
		Int_config[1], field 7	Int_config[0], field 7	Int_config[1], field 6	Int_config[0], field 6	Int_config[1], field 5	Int_config[0], field 5	Int_config[1], field 4	Int_config[0], field 4
		Int_config[1], field 3	Int_config[0], field 3	Int_config[1], field 2	Int_config[0], field 2	Int_config[1], field 1	Int_config[0], field 1	Int_config[1], field 0	Int_config[0], field 0
	PPI Status Register	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		ppi_status[15]	ppi_status[14]	ppi_status[13]	ppi_status[12]	ppi_status[11]	ppi_status[10]	ppi_status[9]	ppi_status[8]
		ppi_status[7]	ppi_status[6]	ppi_status[5]	ppi_status[4]	ppi_status[3]	ppi_status[2]	ppi_status[1]	ppi_status[0]
	SPI Status Registersn n = 0 to 16	spi_status[31]	spi_status[30]	spi_status[29]	spi_status[28]	spi_status[27]	spi_status[26]	spi_status[25]	spi_status[24]
		spi_status[23]	spi_status[22]	spi_status[21]	spi_status[20]	spi_status[19]	spi_status[18]	spi_status[17]	spi_status[16]
		spi_status[15]	spi_status[14]	spi_status[13]	spi_status[12]	spi_status[11]	spi_status[10]	spi_status[9]	spi_status[8]
		spi_status[7]	spi_status[6]	spi_status[5]	spi_status[4]	spi_status[3]	spi_status[2]	spi_status[1]	spi_status[0]
	ICDSGR	-	-	-	-	-	-	TargetListFilter[1]	TargetListFilter[0]
		CPUTargetList[7]	CPUTargetList[6]	CPUTargetList[5]	CPUTargetList[4]	CPUTargetList[3]	CPUTargetList[2]	CPUTargetList[1]	CPUTargetList[0]
		SATT	-	-	-	-	-	-	-
	ICCICR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	SBPR	FIQEn	AckCtI	EnableNS	EnableS
	ICCPMR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Priority[7]	Priority[6]	Priority[5]	Priority[4]	Priority[3]	Priority[2]	Priority[1]	Priority[0]
	ICCBPR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	Binary point[2]	Binary point[1]	Binary point[0]
	ICCIAR	-	-	-	-	-	-	-	-
-		-	-	-	-	-	-	-	
-		-	-	CPUID[2]	CPUID[1]	CPUID[0]	ACKINTID[9]	ACKINTID[8]	
ACKINTID[7]		ACKINTID[6]	ACKINTID[5]	ACKINTID[4]	ACKINTID[3]	ACKINTID[2]	ACKINTID[1]	ACKINTID[0]	
ICCEOIR	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	
	-	-	-	CPUID[2]	CPUID[1]	CPUID[0]	EOINTID[9]	EOINTID[8]	
	EOINTID[7]	EOINTID[6]	EOINTID[5]	EOINTID[4]	EOINTID[3]	EOINTID[2]	EOINTID[1]	EOINTID[0]	
ICCRPR	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	
	Priority[7]	Priority[6]	Priority[5]	Priority[4]	Priority[3]	Priority[2]	Priority[1]	Priority[0]	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Interrupt controller	ICCHPIR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	CPUID[2]	CPUID[1]	CPUID[0]	PENDINTID[9]	PENDINTID[8]
		PENDINTID[7]	PENDINTID[6]	PENDINTID[5]	PENDINTID[4]	PENDINTID[3]	PENDINTID[2]	PENDINTID[1]	PENDINTID[0]
	ICCABPR								
	ICCIIDR	ProductID[11]	ProductID[10]	ProductID[9]	ProductID[8]	ProductID[7]	ProductID[6]	ProductID[5]	ProductID[4]
		ProductID[3]	ProductID[2]	ProductID[1]	ProductID[0]	Architecture version[3]	Architecture version[2]	Architecture version[1]	Architecture version[0]
		Revision[3]	Revision[2]	Revision[1]	Revision[0]	Implementer[11]	Implementer[10]	Implementer[9]	Implementer[8]
		Implementer[7]	Implementer[6]	Implementer[5]	Implementer[4]	Implementer[3]	Implementer[2]	Implementer[1]	Implementer[0]
	Bus state controller	CMNCR	-	-	-	TL0	-	-	-
-			-	-	-	-	-	-	-
-			-	-	-	-	DPRTY[1]	DPRTY[0]	-
-			-	-	-	-	-	HIZMEM	HIZCNT
CS0BCR		-	IWW[2]	IWW[1]	IWW[0]	IWRWD[2]	IWRWD[1]	IWRWD[0]	IWRWS[2]
		IWRWS[1]	IWRWS[0]	IWRRD[2]	IWRRD[1]	IWRRD[0]	IWRRS[2]	IWRRS[1]	IWRRS[0]
		-	TYPE[2]	TYPE[1]	TYPE[0]	-	BSZ[1]	BSZ[0]	-
		-	-	-	-	-	-	-	-
CS1BCR		-	IWW[2]	IWW[1]	IWW[0]	IWRWD[2]	IWRWD[1]	IWRWD[0]	IWRWS[2]
		IWRWS[1]	IWRWS[0]	IWRRD[2]	IWRRD[1]	IWRRD[0]	IWRRS[2]	IWRRS[1]	IWRRS[0]
		-	TYPE[2]	TYPE[1]	TYPE[0]	-	BSZ[1]	BSZ[0]	-
		-	-	-	-	-	-	-	-
CS2BCR		-	IWW[2]	IWW[1]	IWW[0]	IWRWD[2]	IWRWD[1]	IWRWD[0]	IWRWS[2]
		IWRWS[1]	IWRWS[0]	IWRRD[2]	IWRRD[1]	IWRRD[0]	IWRRS[2]	IWRRS[1]	IWRRS[0]
		-	TYPE[2]	TYPE[1]	TYPE[0]	-	BSZ[1]	BSZ[0]	-
		-	-	-	-	-	-	-	-
CS3BCR		-	IWW[2]	IWW[1]	IWW[0]	IWRWD[2]	IWRWD[1]	IWRWD[0]	IWRWS[2]
		IWRWS[1]	IWRWS[0]	IWRRD[2]	IWRRD[1]	IWRRD[0]	IWRRS[2]	IWRRS[1]	IWRRS[0]
		-	TYPE[2]	TYPE[1]	TYPE[0]	-	BSZ[1]	BSZ[0]	-
		-	-	-	-	-	-	-	-
CS4BCR		-	IWW[2]	IWW[1]	IWW[0]	IWRWD[2]	IWRWD[1]	IWRWD[0]	IWRWS[2]
		IWRWS[1]	IWRWS[0]	IWRRD[2]	IWRRD[1]	IWRRD[0]	IWRRS[2]	IWRRS[1]	IWRRS[0]
		-	TYPE[2]	TYPE[1]	TYPE[0]	-	BSZ[1]	BSZ[0]	-
		-	-	-	-	-	-	-	-
CS5BCR		-	IWW[2]	IWW[1]	IWW[0]	IWRWD[2]	IWRWD[1]	IWRWD[0]	IWRWS[2]
		IWRWS[1]	IWRWS[0]	IWRRD[2]	IWRRD[1]	IWRRD[0]	IWRRS[2]	IWRRS[1]	IWRRS[0]
		-	TYPE[2]	TYPE[1]	TYPE[0]	-	BSZ[1]	BSZ[0]	-
		-	-	-	-	-	-	-	-
CS0WCR Normal space, SRAM with byte selection, MPX-I/O		-	-	-	-	-	-	-	-
		-	-	-	BAS	-	-	-	-
		-	-	-	SW[1]	SW[0]	WR[3]	WR[2]	WR[1]
		WR[0]	WM	-	-	-	-	HW[1]	HW[0]
CS0WCR Burst ROM (clock asynchronous)		-	-	-	-	-	-	-	-
		-	-	BST[1]	BST[0]	-	-	BW[1]	BW[0]
		W[0]	WM	-	-	-	W[3]	W[2]	W[1]
CS0WCR Burst ROM (clock synchronous)		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	BW[1]	BW[0]
		-	-	-	-	-	W[3]	W[2]	W[1]
		W[0]	WM	-	-	-	-	-	-
CS1WCR		-	-	-	-	-	-	-	-
		-	-	-	BAS	-	WW[2]	WW[1]	WW[0]
		-	-	-	SW[1]	SW[0]	WR[3]	WR[2]	WR[1]
		WR[0]	WM	-	-	-	-	HW[1]	HW[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Bus state controller	CS2WCR Normal space, SRAM with byte selection, MPX-I/O	-	-	-	-	-	-	-	-	
		-	-	-	BAS	-	-	-	-	
		-	-	-	-	-	-	WR[3]	WR[2]	WR[1]
		WR[0]	WM	-	-	-	-	-	-	-
	CS2WCR Burst ROM (clock synchronous)	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	A2CL[1]
		A2CL[0]	-	-	-	-	-	-	-	-
	CS3WCR Normal space, SRAM with byte selection, MPX-I/O	-	-	-	-	BAS	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	WR[3]	WR[2]	WR[1]
		WR[0]	WM	-	-	-	-	-	-	-
	CS3WCR Burst ROM (clock synchronous)	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	WTRP[1]	WTRP[0]	-	WTRCD[1]	WTRCD[0]	-	-	A3CL[1]
		A3CL[0]	-	-	TRWL[1]	TRWL[0]	-	WTRC[1]	WTRC[0]	-
	CS4WCR Normal space, SRAM with byte selection, MPX-I/O	-	-	-	-	BAS	-	WW[2]	WW[1]	WW[0]
		-	-	-	-	SW[1]	SW[0]	WR[3]	WR[2]	WR[1]
		-	-	-	-	-	-	-	HW[1]	HW[0]
		WR[0]	WM	-	-	-	-	-	-	-
	CS4WCR Burst ROM (clock asynchronous)	-	-	-	-	-	-	-	-	-
		-	-	BST[1]	BST[0]	-	-	-	BW[1]	BW[0]
		-	-	-	-	SW[1]	SW[0]	W[3]	W[2]	W[1]
		W[0]	WM	-	-	-	-	-	HW[1]	HW[0]
	CS5WCR Normal space, SRAM with byte selection, MPX-I/O	-	-	-	SZSEL	MPXW/BAS	-	WW[2]	WW[1]	WW[0]
		-	-	-	-	SW[1]	SW[0]	WR[3]	WR[2]	WR[1]
		-	-	-	-	-	-	-	HW[1]	HW[0]
		WR[0]	WM	-	-	-	-	-	-	-
	SDCR	-	-	-	-	-	-	-	-	-
		-	-	-	-	A2ROW[1]	A2ROW[0]	-	A2COL[1]	A2COL[0]
		-	-	DEEP	-	-	RFSH	RMODE	PDOWN	BACTV
		-	-	-	-	A3ROW[1]	A3ROW[0]	-	A3COL[1]	A3COL[0]
	RTCSR	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		CMF	CMIE	CKS[2]	CKS[1]	CKS[0]	RRC[2]	RRC[1]	RRC[0]	-
	RTCNT	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
	RTCOR	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
TOSCOR0	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	
TOSCOR1	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Bus state controller	TOSCOR2	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
	TOSCOR3	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
	TOSCOR4	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
	TOSCOR5	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
	TOSTR	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	CS5TOSTF	CS4TOSTF	CS3TOSTF	CS2TOSTF	CS1TOSTF	CS0TOSTF	
	TOENR	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	CS5TOEN	CS4TOEN	CS3TOEN	CS2TOEN	CS1TOEN	CS0TOEN	
	Direct memory access controller	NOSA_0	SA[31]	SA[30]	SA[29]	SA[28]	SA[27]	SA[26]	SA[25]	SA[24]
			SA[23]	SA[22]	SA[21]	SA[20]	SA[19]	SA[18]	SA[17]	SA[16]
			SA[15]	SA[14]	SA[13]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
			SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
NODA_0		DA[31]	DA[30]	DA[29]	DA[28]	DA[27]	DA[26]	DA[25]	DA[24]	
		DA[23]	DA[22]	DA[21]	DA[20]	DA[19]	DA[18]	DA[17]	DA[16]	
		DA[15]	DA[14]	DA[13]	DA[12]	DA[11]	DA[10]	DA[9]	DA[8]	
		DA[7]	DA[6]	DA[5]	DA[4]	DA[3]	DA[2]	DA[1]	DA[0]	
NOTB_0		TB[31]	TB[30]	TB[29]	TB[28]	TB[27]	TB[26]	TB[25]	TB[24]	
		TB[23]	TB[22]	TB[21]	TB[20]	TB[19]	TB[18]	TB[17]	TB[16]	
		TB[15]	TB[14]	TB[13]	TB[12]	TB[11]	TB[10]	TB[9]	TB[8]	
		TB[7]	TB[6]	TB[5]	TB[4]	TB[3]	TB[2]	TB[1]	TB[0]	
NISA_0		SA[31]	SA[30]	SA[29]	SA[28]	SA[27]	SA[26]	SA[25]	SA[24]	
		SA[23]	SA[22]	SA[21]	SA[20]	SA[19]	SA[18]	SA[17]	SA[16]	
		SA[15]	SA[14]	SA[13]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]	
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]	
NIDA_0		DA[31]	DA[30]	DA[29]	DA[28]	DA[27]	DA[26]	DA[25]	DA[24]	
		DA[23]	DA[22]	DA[21]	DA[20]	DA[19]	DA[18]	DA[17]	DA[16]	
		DA[15]	DA[14]	DA[13]	DA[12]	DA[11]	DA[10]	DA[9]	DA[8]	
		DA[7]	DA[6]	DA[5]	DA[4]	DA[3]	DA[2]	DA[1]	DA[0]	
NITB_0		TB[31]	TB[30]	TB[29]	TB[28]	TB[27]	TB[26]	TB[25]	TB[24]	
		TB[23]	TB[22]	TB[21]	TB[20]	TB[19]	TB[18]	TB[17]	TB[16]	
		TB[15]	TB[14]	TB[13]	TB[12]	TB[11]	TB[10]	TB[9]	TB[8]	
		TB[7]	TB[6]	TB[5]	TB[4]	TB[3]	TB[2]	TB[1]	TB[0]	
CRSA_0		CRSA[31]	CRSA[30]	CRSA[29]	CRSA[28]	CRSA[27]	CRSA[26]	CRSA[25]	CRSA[24]	
		CRSA[23]	CRSA[22]	CRSA[21]	CRSA[20]	CRSA[19]	CRSA[18]	CRSA[17]	CRSA[16]	
		CRSA[15]	CRSA[14]	CRSA[13]	CRSA[12]	CRSA[11]	CRSA[10]	CRSA[9]	CRSA[8]	
		CRSA[7]	CRSA[6]	CRSA[5]	CRSA[4]	CRSA[3]	CRSA[2]	CRSA[1]	CRSA[0]	
CRDA_0		CRDA[31]	CRDA[30]	CRDA[29]	CRDA[28]	CRDA[27]	CRDA[26]	CRDA[25]	CRDA[24]	
		CRDA[23]	CRDA[22]	CRDA[21]	CRDA[20]	CRDA[19]	CRDA[18]	CRDA[17]	CRDA[16]	
		CRDA[15]	CRDA[14]	CRDA[13]	CRDA[12]	CRDA[11]	CRDA[10]	CRDA[9]	CRDA[8]	
		CRDA[7]	CRDA[6]	CRDA[5]	CRDA[4]	CRDA[3]	CRDA[2]	CRDA[1]	CRDA[0]	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Direct memory access controller	CRTB_0	CRTB[31]	CRTB[30]	CRTB[29]	CRTB[28]	CRTB[27]	CRTB[26]	CRTB[25]	CRTB[24]	
		CRTB[23]	CRTB[22]	CRTB[21]	CRTB[20]	CRTB[19]	CRTB[18]	CRTB[17]	CRTB[16]	
		CRTB[15]	CRTB[14]	CRTB[13]	CRTB[12]	CRTB[11]	CRTB[10]	CRTB[9]	CRTB[8]	
		CRTB[7]	CRTB[6]	CRTB[5]	CRTB[4]	CRTB[3]	CRTB[2]	CRTB[1]	CRTB[0]	
	CHSTAT_0	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	INTMSK
		-	-	-	-	MODE	DER	DW	DL	-
		SR	TC	END	ER	SUS	TACT	RQST	EN	-
	CHCTRL_0	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	CLRINTMSK	SETINTMSK
		-	-	-	-	-	-	-	CLRSUS	SETSUS
		-	CLRTC	CLREND	CLRRQ	SWRST	STG	CLREN	SETEN	-
	CHCFG_0	DMS	REN	RSW	RSEL	SBE	-	-	-	DEM
		-	TM	DAD	SAD	DDS[3]	DDS[2]	DDS[1]	DDS[0]	-
		SDS[3]	SDS[2]	SDS[1]	SDS[0]	-	AM[2]	AM[1]	AM[0]	-
		-	LVL	HIEN	LOEN	REQD	SEL[2]	SEL[1]	SEL[0]	-
	CHITVL_0	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		ITVL[15]	ITVL[14]	ITVL[13]	ITVL[12]	ITVL[11]	ITVL[10]	ITVL[9]	ITVL[8]	-
		ITVL[7]	ITVL[6]	ITVL[5]	ITVL[4]	ITVL[3]	ITVL[2]	ITVL[1]	ITVL[0]	-
	CHEXT_0	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		DCA[3]	DCA[2]	DCA[1]	DCA[0]	-	-	-	-	-
		SCA[3]	SCA[2]	SCA[1]	SCA[0]	-	-	-	-	-
	NXLA_0	NXLA[31]	NXLA[30]	NXLA[29]	NXLA[28]	NXLA[27]	NXLA[26]	NXLA[25]	NXLA[24]	-
		NXLA[23]	NXLA[22]	NXLA[21]	NXLA[20]	NXLA[19]	NXLA[18]	NXLA[17]	NXLA[16]	-
		NXLA[15]	NXLA[14]	NXLA[13]	NXLA[12]	NXLA[11]	NXLA[10]	NXLA[9]	NXLA[8]	-
		NXLA[7]	NXLA[6]	NXLA[5]	NXLA[4]	NXLA[3]	NXLA[2]	NXLA[1]	NXLA[0]	-
	CRLA_0	CRLA[31]	CRLA[30]	CRLA[29]	CRLA[28]	CRLA[27]	CRLA[26]	CRLA[25]	CRLA[24]	-
		CRLA[23]	CRLA[22]	CRLA[21]	CRLA[20]	CRLA[19]	CRLA[18]	CRLA[17]	CRLA[16]	-
		CRLA[15]	CRLA[14]	CRLA[13]	CRLA[12]	CRLA[11]	CRLA[10]	CRLA[9]	CRLA[8]	-
		CRLA[7]	CRLA[6]	CRLA[5]	CRLA[4]	CRLA[3]	CRLA[2]	CRLA[1]	CRLA[0]	-
	NOSA_1	SA[31]	SA[30]	SA[29]	SA[28]	SA[27]	SA[26]	SA[25]	SA[24]	-
		SA[23]	SA[22]	SA[21]	SA[20]	SA[19]	SA[18]	SA[17]	SA[16]	-
		SA[15]	SA[14]	SA[13]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]	-
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]	-
	NODA_1	DA[31]	DA[30]	DA[29]	DA[28]	DA[27]	DA[26]	DA[25]	DA[24]	-
		DA[23]	DA[22]	DA[21]	DA[20]	DA[19]	DA[18]	DA[17]	DA[16]	-
		DA[15]	DA[14]	DA[13]	DA[12]	DA[11]	DA[10]	DA[9]	DA[8]	-
		DA[7]	DA[6]	DA[5]	DA[4]	DA[3]	DA[2]	DA[1]	DA[0]	-
	NOTB_1	TB[31]	TB[30]	TB[29]	TB[28]	TB[27]	TB[26]	TB[25]	TB[24]	-
		TB[23]	TB[22]	TB[21]	TB[20]	TB[19]	TB[18]	TB[17]	TB[16]	-
		TB[15]	TB[14]	TB[13]	TB[12]	TB[11]	TB[10]	TB[9]	TB[8]	-
		TB[7]	TB[6]	TB[5]	TB[4]	TB[3]	TB[2]	TB[1]	TB[0]	-
	N1SA_1	SA[31]	SA[30]	SA[29]	SA[28]	SA[27]	SA[26]	SA[25]	SA[24]	-
		SA[23]	SA[22]	SA[21]	SA[20]	SA[19]	SA[18]	SA[17]	SA[16]	-
		SA[15]	SA[14]	SA[13]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]	-
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]	-
	N1DA_1	DA[31]	DA[30]	DA[29]	DA[28]	DA[27]	DA[26]	DA[25]	DA[24]	-
		DA[23]	DA[22]	DA[21]	DA[20]	DA[19]	DA[18]	DA[17]	DA[16]	-
		DA[15]	DA[14]	DA[13]	DA[12]	DA[11]	DA[10]	DA[9]	DA[8]	-
		DA[7]	DA[6]	DA[5]	DA[4]	DA[3]	DA[2]	DA[1]	DA[0]	-
	N1TB_1	TB[31]	TB[30]	TB[29]	TB[28]	TB[27]	TB[26]	TB[25]	TB[24]	-
		TB[23]	TB[22]	TB[21]	TB[20]	TB[19]	TB[18]	TB[17]	TB[16]	-
		TB[15]	TB[14]	TB[13]	TB[12]	TB[11]	TB[10]	TB[9]	TB[8]	-
		TB[7]	TB[6]	TB[5]	TB[4]	TB[3]	TB[2]	TB[1]	TB[0]	-

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Direct memory access controller	CRSA_1	CRSA[31]	CRSA[30]	CRSA[29]	CRSA[28]	CRSA[27]	CRSA[26]	CRSA[25]	CRSA[24]
		CRSA[23]	CRSA[22]	CRSA[21]	CRSA[20]	CRSA[19]	CRSA[18]	CRSA[17]	CRSA[16]
		CRSA[15]	CRSA[14]	CRSA[13]	CRSA[12]	CRSA[11]	CRSA[10]	CRSA[9]	CRSA[8]
		CRSA[7]	CRSA[6]	CRSA[5]	CRSA[4]	CRSA[3]	CRSA[2]	CRSA[1]	CRSA[0]
	CRDA_1	CRDA[31]	CRDA[30]	CRDA[29]	CRDA[28]	CRDA[27]	CRDA[26]	CRDA[25]	CRDA[24]
		CRDA[23]	CRDA[22]	CRDA[21]	CRDA[20]	CRDA[19]	CRDA[18]	CRDA[17]	CRDA[16]
		CRDA[15]	CRDA[14]	CRDA[13]	CRDA[12]	CRDA[11]	CRDA[10]	CRDA[9]	CRDA[8]
		CRDA[7]	CRDA[6]	CRDA[5]	CRDA[4]	CRDA[3]	CRDA[2]	CRDA[1]	CRDA[0]
	CRTB_1	CRTB[31]	CRTB[30]	CRTB[29]	CRTB[28]	CRTB[27]	CRTB[26]	CRTB[25]	CRTB[24]
		CRTB[23]	CRTB[22]	CRTB[21]	CRTB[20]	CRTB[19]	CRTB[18]	CRTB[17]	CRTB[16]
		CRTB[15]	CRTB[14]	CRTB[13]	CRTB[12]	CRTB[11]	CRTB[10]	CRTB[9]	CRTB[8]
	CHSTAT_1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	INTMSK
		-	-	-	-	MODE	DER	DW	DL
		SR	TC	END	ER	SUS	TACT	RQST	EN
	CHCTRL_1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	CLRINTMSK	SETINTMSK
		-	-	-	-	-	-	CLRSUS	SETSUS
		-	CLRTC	CLREND	CLRRQ	SWRST	STG	CLREN	SETEN
	CHCFG_1	DMS	REN	RSW	RSEL	SBE	-	-	DEM
		-	TM	DAD	SAD	DDS[3]	DDS[2]	DDS[1]	DDS[0]
		SDS[3]	SDS[2]	SDS[1]	SDS[0]	-	AM[2]	AM[1]	AM[0]
		-	LVL	HIEN	LOEN	REQD	SEL[2]	SEL[1]	SEL[0]
	CHITVL_1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		ITVL[15]	ITVL[14]	ITVL[13]	ITVL[12]	ITVL[11]	ITVL[10]	ITVL[9]	ITVL[8]
	CHEXT_1	ITVL[7]	ITVL[6]	ITVL[5]	ITVL[4]	ITVL[3]	ITVL[2]	ITVL[1]	ITVL[0]
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		DCA[3]	DCA[2]	DCA[1]	DCA[0]	-	-	-	-
	NXLA_1	SCA[3]	SCA[2]	SCA[1]	SCA[0]	-	-	-	-
		NXLA[31]	NXLA[30]	NXLA[29]	NXLA[28]	NXLA[27]	NXLA[26]	NXLA[25]	NXLA[24]
		NXLA[23]	NXLA[22]	NXLA[21]	NXLA[20]	NXLA[19]	NXLA[18]	NXLA[17]	NXLA[16]
		NXLA[15]	NXLA[14]	NXLA[13]	NXLA[12]	NXLA[11]	NXLA[10]	NXLA[9]	NXLA[8]
	CRLA_1	NXLA[7]	NXLA[6]	NXLA[5]	NXLA[4]	NXLA[3]	NXLA[2]	NXLA[1]	NXLA[0]
		CRLA[31]	CRLA[30]	CRLA[29]	CRLA[28]	CRLA[27]	CRLA[26]	CRLA[25]	CRLA[24]
		CRLA[23]	CRLA[22]	CRLA[21]	CRLA[20]	CRLA[19]	CRLA[18]	CRLA[17]	CRLA[16]
		CRLA[15]	CRLA[14]	CRLA[13]	CRLA[12]	CRLA[11]	CRLA[10]	CRLA[9]	CRLA[8]
	NOSA_2	CRLA[7]	CRLA[6]	CRLA[5]	CRLA[4]	CRLA[3]	CRLA[2]	CRLA[1]	CRLA[0]
		SA[31]	SA[30]	SA[29]	SA[28]	SA[27]	SA[26]	SA[25]	SA[24]
		SA[23]	SA[22]	SA[21]	SA[20]	SA[19]	SA[18]	SA[17]	SA[16]
		SA[15]	SA[14]	SA[13]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
	NODA_2	SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
		DA[31]	DA[30]	DA[29]	DA[28]	DA[27]	DA[26]	DA[25]	DA[24]
		DA[23]	DA[22]	DA[21]	DA[20]	DA[19]	DA[18]	DA[17]	DA[16]
		DA[15]	DA[14]	DA[13]	DA[12]	DA[11]	DA[10]	DA[9]	DA[8]
	NOTB_2	DA[7]	DA[6]	DA[5]	DA[4]	DA[3]	DA[2]	DA[1]	DA[0]
		TB[31]	TB[30]	TB[29]	TB[28]	TB[27]	TB[26]	TB[25]	TB[24]
		TB[23]	TB[22]	TB[21]	TB[20]	TB[19]	TB[18]	TB[17]	TB[16]
		TB[15]	TB[14]	TB[13]	TB[12]	TB[11]	TB[10]	TB[9]	TB[8]
	NISA_2	TB[7]	TB[6]	TB[5]	TB[4]	TB[3]	TB[2]	TB[1]	TB[0]
		SA[31]	SA[30]	SA[29]	SA[28]	SA[27]	SA[26]	SA[25]	SA[24]
		SA[23]	SA[22]	SA[21]	SA[20]	SA[19]	SA[18]	SA[17]	SA[16]
		SA[15]	SA[14]	SA[13]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]



Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Direct memory access controller	N1DA_2	DA[31]	DA[30]	DA[29]	DA[28]	DA[27]	DA[26]	DA[25]	DA[24]
		DA[23]	DA[22]	DA[21]	DA[20]	DA[19]	DA[18]	DA[17]	DA[16]
		DA[15]	DA[14]	DA[13]	DA[12]	DA[11]	DA[10]	DA[9]	DA[8]
		DA[7]	DA[6]	DA[5]	DA[4]	DA[3]	DA[2]	DA[1]	DA[0]
	N1TB_2	TB[31]	TB[30]	TB[29]	TB[28]	TB[27]	TB[26]	TB[25]	TB[24]
		TB[23]	TB[22]	TB[21]	TB[20]	TB[19]	TB[18]	TB[17]	TB[16]
		TB[15]	TB[14]	TB[13]	TB[12]	TB[11]	TB[10]	TB[9]	TB[8]
		TB[7]	TB[6]	TB[5]	TB[4]	TB[3]	TB[2]	TB[1]	TB[0]
	CRSA_2	CRSA[31]	CRSA[30]	CRSA[29]	CRSA[28]	CRSA[27]	CRSA[26]	CRSA[25]	CRSA[24]
		CRSA[23]	CRSA[22]	CRSA[21]	CRSA[20]	CRSA[19]	CRSA[18]	CRSA[17]	CRSA[16]
		CRSA[15]	CRSA[14]	CRSA[13]	CRSA[12]	CRSA[11]	CRSA[10]	CRSA[9]	CRSA[8]
		CRSA[7]	CRSA[6]	CRSA[5]	CRSA[4]	CRSA[3]	CRSA[2]	CRSA[1]	CRSA[0]
	CRDA_2	CRDA[31]	CRDA[30]	CRDA[29]	CRDA[28]	CRDA[27]	CRDA[26]	CRDA[25]	CRDA[24]
		CRDA[23]	CRDA[22]	CRDA[21]	CRDA[20]	CRDA[19]	CRDA[18]	CRDA[17]	CRDA[16]
		CRDA[15]	CRDA[14]	CRDA[13]	CRDA[12]	CRDA[11]	CRDA[10]	CRDA[9]	CRDA[8]
		CRDA[7]	CRDA[6]	CRDA[5]	CRDA[4]	CRDA[3]	CRDA[2]	CRDA[1]	CRDA[0]
	CRTB_2	CRTB[31]	CRTB[30]	CRTB[29]	CRTB[28]	CRTB[27]	CRTB[26]	CRTB[25]	CRTB[24]
		CRTB[23]	CRTB[22]	CRTB[21]	CRTB[20]	CRTB[19]	CRTB[18]	CRTB[17]	CRTB[16]
		CRTB[15]	CRTB[14]	CRTB[13]	CRTB[12]	CRTB[11]	CRTB[10]	CRTB[9]	CRTB[8]
		CRTB[7]	CRTB[6]	CRTB[5]	CRTB[4]	CRTB[3]	CRTB[2]	CRTB[1]	CRTB[0]
	CHSTAT_2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	INTMSK
		-	-	-	-	MODE	DER	DW	DL
		SR	TC	END	ER	SUS	TACT	RQST	EN
	CHCTRL_2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	CLRINTMSK	SETINTMSK
		-	-	-	-	-	-	CLRSUS	SETSUS
		-	CLRTC	CLREND	CLRRQ	SWRST	STG	CLREN	SETEN
	CHCFG_2	DMS	REN	RSW	RSEL	SBE	-	-	DEM
		-	TM	DAD	SAD	DDS[3]	DDS[2]	DDS[1]	DDS[0]
		SDS[3]	SDS[2]	SDS[1]	SDS[0]	-	AM[2]	AM[1]	AM[0]
		-	LVL	HIEN	LOEN	REQD	SEL[2]	SEL[1]	SEL[0]
	CHITVL_2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		ITVL[15]	ITVL[14]	ITVL[13]	ITVL[12]	ITVL[11]	ITVL[10]	ITVL[9]	ITVL[8]
		ITVL[7]	ITVL[6]	ITVL[5]	ITVL[4]	ITVL[3]	ITVL[2]	ITVL[1]	ITVL[0]
	CHEXT_2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		DCA[3]	DCA[2]	DCA[1]	DCA[0]	-	-	-	-
		SCA[3]	SCA[2]	SCA[1]	SCA[0]	-	-	-	-
	NXLA_2	NXLA[31]	NXLA[30]	NXLA[29]	NXLA[28]	NXLA[27]	NXLA[26]	NXLA[25]	NXLA[24]
		NXLA[23]	NXLA[22]	NXLA[21]	NXLA[20]	NXLA[19]	NXLA[18]	NXLA[17]	NXLA[16]
		NXLA[15]	NXLA[14]	NXLA[13]	NXLA[12]	NXLA[11]	NXLA[10]	NXLA[9]	NXLA[8]
		NXLA[7]	NXLA[6]	NXLA[5]	NXLA[4]	NXLA[3]	NXLA[2]	NXLA[1]	NXLA[0]
	CRLA_2	CRLA[31]	CRLA[30]	CRLA[29]	CRLA[28]	CRLA[27]	CRLA[26]	CRLA[25]	CRLA[24]
		CRLA[23]	CRLA[22]	CRLA[21]	CRLA[20]	CRLA[19]	CRLA[18]	CRLA[17]	CRLA[16]
		CRLA[15]	CRLA[14]	CRLA[13]	CRLA[12]	CRLA[11]	CRLA[10]	CRLA[9]	CRLA[8]
		CRLA[7]	CRLA[6]	CRLA[5]	CRLA[4]	CRLA[3]	CRLA[2]	CRLA[1]	CRLA[0]
	NOSA_3	SA[31]	SA[30]	SA[29]	SA[28]	SA[27]	SA[26]	SA[25]	SA[24]
		SA[23]	SA[22]	SA[21]	SA[20]	SA[19]	SA[18]	SA[17]	SA[16]
		SA[15]	SA[14]	SA[13]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	NODA_3	DA[31]	DA[30]	DA[29]	DA[28]	DA[27]	DA[26]	DA[25]	DA[24]
		DA[23]	DA[22]	DA[21]	DA[20]	DA[19]	DA[18]	DA[17]	DA[16]
		DA[15]	DA[14]	DA[13]	DA[12]	DA[11]	DA[10]	DA[9]	DA[8]
		DA[7]	DA[6]	DA[5]	DA[4]	DA[3]	DA[2]	DA[1]	DA[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Direct memory access controller	NOTB_3	TB[31]	TB[30]	TB[29]	TB[28]	TB[27]	TB[26]	TB[25]	TB[24]
		TB[23]	TB[22]	TB[21]	TB[20]	TB[19]	TB[18]	TB[17]	TB[16]
		TB[15]	TB[14]	TB[13]	TB[12]	TB[11]	TB[10]	TB[9]	TB[8]
		TB[7]	TB[6]	TB[5]	TB[4]	TB[3]	TB[2]	TB[1]	TB[0]
	N1SA_3	SA[31]	SA[30]	SA[29]	SA[28]	SA[27]	SA[26]	SA[25]	SA[24]
		SA[23]	SA[22]	SA[21]	SA[20]	SA[19]	SA[18]	SA[17]	SA[16]
		SA[15]	SA[14]	SA[13]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	N1DA_3	DA[31]	DA[30]	DA[29]	DA[28]	DA[27]	DA[26]	DA[25]	DA[24]
		DA[23]	DA[22]	DA[21]	DA[20]	DA[19]	DA[18]	DA[17]	DA[16]
		DA[15]	DA[14]	DA[13]	DA[12]	DA[11]	DA[10]	DA[9]	DA[8]
		DA[7]	DA[6]	DA[5]	DA[4]	DA[3]	DA[2]	DA[1]	DA[0]
	N1TB_3	TB[31]	TB[30]	TB[29]	TB[28]	TB[27]	TB[26]	TB[25]	TB[24]
		TB[23]	TB[22]	TB[21]	TB[20]	TB[19]	TB[18]	TB[17]	TB[16]
		TB[15]	TB[14]	TB[13]	TB[12]	TB[11]	TB[10]	TB[9]	TB[8]
		TB[7]	TB[6]	TB[5]	TB[4]	TB[3]	TB[2]	TB[1]	TB[0]
	CRSA_3	CRSA[31]	CRSA[30]	CRSA[29]	CRSA[28]	CRSA[27]	CRSA[26]	CRSA[25]	CRSA[24]
		CRSA[23]	CRSA[22]	CRSA[21]	CRSA[20]	CRSA[19]	CRSA[18]	CRSA[17]	CRSA[16]
		CRSA[15]	CRSA[14]	CRSA[13]	CRSA[12]	CRSA[11]	CRSA[10]	CRSA[9]	CRSA[8]
		CRSA[7]	CRSA[6]	CRSA[5]	CRSA[4]	CRSA[3]	CRSA[2]	CRSA[1]	CRSA[0]
	CRDA_3	CRDA[31]	CRDA[30]	CRDA[29]	CRDA[28]	CRDA[27]	CRDA[26]	CRDA[25]	CRDA[24]
		CRDA[23]	CRDA[22]	CRDA[21]	CRDA[20]	CRDA[19]	CRDA[18]	CRDA[17]	CRDA[16]
		CRDA[15]	CRDA[14]	CRDA[13]	CRDA[12]	CRDA[11]	CRDA[10]	CRDA[9]	CRDA[8]
		CRDA[7]	CRDA[6]	CRDA[5]	CRDA[4]	CRDA[3]	CRDA[2]	CRDA[1]	CRDA[0]
	CRTB_3	CRTB[31]	CRTB[30]	CRTB[29]	CRTB[28]	CRTB[27]	CRTB[26]	CRTB[25]	CRTB[24]
		CRTB[23]	CRTB[22]	CRTB[21]	CRTB[20]	CRTB[19]	CRTB[18]	CRTB[17]	CRTB[16]
		CRTB[15]	CRTB[14]	CRTB[13]	CRTB[12]	CRTB[11]	CRTB[10]	CRTB[9]	CRTB[8]
		CRTB[7]	CRTB[6]	CRTB[5]	CRTB[4]	CRTB[3]	CRTB[2]	CRTB[1]	CRTB[0]
	CHSTAT_3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	INTMSK
		-	-	-	-	MODE	DER	DW	DL
		SR	TC	END	ER	SUS	TACT	RQST	EN
	CHCTRL_3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	CLRINTMSK	SETINTMSK
		-	-	-	-	-	-	CLRSUS	SETSUS
		-	CLRTC	CLREND	CLRRQ	SWRST	STG	CLREN	SETEN
	CHCFG_3	DMS	REN	RSW	RSEL	SBE	-	-	DEM
		-	TM	DAD	SAD	DDS[3]	DDS[2]	DDS[1]	DDS[0]
		SDS[3]	SDS[2]	SDS[1]	SDS[0]	-	AM[2]	AM[1]	AM[0]
		-	LVL	HIEN	LOEN	REQD	SEL[2]	SEL[1]	SEL[0]
	CHITVL_3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		ITVL[15]	ITVL[14]	ITVL[13]	ITVL[12]	ITVL[11]	ITVL[10]	ITVL[9]	ITVL[8]
		ITVL[7]	ITVL[6]	ITVL[5]	ITVL[4]	ITVL[3]	ITVL[2]	ITVL[1]	ITVL[0]
	CHEXT_3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		DCA[3]	DCA[2]	DCA[1]	DCA[0]	-	-	-	-
		SCA[3]	SCA[2]	SCA[1]	SCA[0]	-	-	-	-
	NXLA_3	NXLA[31]	NXLA[30]	NXLA[29]	NXLA[28]	NXLA[27]	NXLA[26]	NXLA[25]	NXLA[24]
		NXLA[23]	NXLA[22]	NXLA[21]	NXLA[20]	NXLA[19]	NXLA[18]	NXLA[17]	NXLA[16]
		NXLA[15]	NXLA[14]	NXLA[13]	NXLA[12]	NXLA[11]	NXLA[10]	NXLA[9]	NXLA[8]
		NXLA[7]	NXLA[6]	NXLA[5]	NXLA[4]	NXLA[3]	NXLA[2]	NXLA[1]	NXLA[0]
	CRLA_3	CRLA[31]	CRLA[30]	CRLA[29]	CRLA[28]	CRLA[27]	CRLA[26]	CRLA[25]	CRLA[24]
		CRLA[23]	CRLA[22]	CRLA[21]	CRLA[20]	CRLA[19]	CRLA[18]	CRLA[17]	CRLA[16]
		CRLA[15]	CRLA[14]	CRLA[13]	CRLA[12]	CRLA[11]	CRLA[10]	CRLA[9]	CRLA[8]
		CRLA[7]	CRLA[6]	CRLA[5]	CRLA[4]	CRLA[3]	CRLA[2]	CRLA[1]	CRLA[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Direct memory access controller	NOSA_4	SA[31]	SA[30]	SA[29]	SA[28]	SA[27]	SA[26]	SA[25]	SA[24]
		SA[23]	SA[22]	SA[21]	SA[20]	SA[19]	SA[18]	SA[17]	SA[16]
		SA[15]	SA[14]	SA[13]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	NODA_4	DA[31]	DA[30]	DA[29]	DA[28]	DA[27]	DA[26]	DA[25]	DA[24]
		DA[23]	DA[22]	DA[21]	DA[20]	DA[19]	DA[18]	DA[17]	DA[16]
		DA[15]	DA[14]	DA[13]	DA[12]	DA[11]	DA[10]	DA[9]	DA[8]
		DA[7]	DA[6]	DA[5]	DA[4]	DA[3]	DA[2]	DA[1]	DA[0]
	NOTB_4	TB[31]	TB[30]	TB[29]	TB[28]	TB[27]	TB[26]	TB[25]	TB[24]
		TB[23]	TB[22]	TB[21]	TB[20]	TB[19]	TB[18]	TB[17]	TB[16]
		TB[15]	TB[14]	TB[13]	TB[12]	TB[11]	TB[10]	TB[9]	TB[8]
		TB[7]	TB[6]	TB[5]	TB[4]	TB[3]	TB[2]	TB[1]	TB[0]
	NISA_4	SA[31]	SA[30]	SA[29]	SA[28]	SA[27]	SA[26]	SA[25]	SA[24]
		SA[23]	SA[22]	SA[21]	SA[20]	SA[19]	SA[18]	SA[17]	SA[16]
		SA[15]	SA[14]	SA[13]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	NIDA_4	DA[31]	DA[30]	DA[29]	DA[28]	DA[27]	DA[26]	DA[25]	DA[24]
		DA[23]	DA[22]	DA[21]	DA[20]	DA[19]	DA[18]	DA[17]	DA[16]
		DA[15]	DA[14]	DA[13]	DA[12]	DA[11]	DA[10]	DA[9]	DA[8]
		DA[7]	DA[6]	DA[5]	DA[4]	DA[3]	DA[2]	DA[1]	DA[0]
	NITB_4	TB[31]	TB[30]	TB[29]	TB[28]	TB[27]	TB[26]	TB[25]	TB[24]
		TB[23]	TB[22]	TB[21]	TB[20]	TB[19]	TB[18]	TB[17]	TB[16]
		TB[15]	TB[14]	TB[13]	TB[12]	TB[11]	TB[10]	TB[9]	TB[8]
		TB[7]	TB[6]	TB[5]	TB[4]	TB[3]	TB[2]	TB[1]	TB[0]
	CRSA_4	CRSA[31]	CRSA[30]	CRSA[29]	CRSA[28]	CRSA[27]	CRSA[26]	CRSA[25]	CRSA[24]
		CRSA[23]	CRSA[22]	CRSA[21]	CRSA[20]	CRSA[19]	CRSA[18]	CRSA[17]	CRSA[16]
		CRSA[15]	CRSA[14]	CRSA[13]	CRSA[12]	CRSA[11]	CRSA[10]	CRSA[9]	CRSA[8]
		CRSA[7]	CRSA[6]	CRSA[5]	CRSA[4]	CRSA[3]	CRSA[2]	CRSA[1]	CRSA[0]
	CRDA_4	CRDA[31]	CRDA[30]	CRDA[29]	CRDA[28]	CRDA[27]	CRDA[26]	CRDA[25]	CRDA[24]
		CRDA[23]	CRDA[22]	CRDA[21]	CRDA[20]	CRDA[19]	CRDA[18]	CRDA[17]	CRDA[16]
		CRDA[15]	CRDA[14]	CRDA[13]	CRDA[12]	CRDA[11]	CRDA[10]	CRDA[9]	CRDA[8]
		CRDA[7]	CRDA[6]	CRDA[5]	CRDA[4]	CRDA[3]	CRDA[2]	CRDA[1]	CRDA[0]
	CRTB_4	CRTB[31]	CRTB[30]	CRTB[29]	CRTB[28]	CRTB[27]	CRTB[26]	CRTB[25]	CRTB[24]
		CRTB[23]	CRTB[22]	CRTB[21]	CRTB[20]	CRTB[19]	CRTB[18]	CRTB[17]	CRTB[16]
		CRTB[15]	CRTB[14]	CRTB[13]	CRTB[12]	CRTB[11]	CRTB[10]	CRTB[9]	CRTB[8]
		CRTB[7]	CRTB[6]	CRTB[5]	CRTB[4]	CRTB[3]	CRTB[2]	CRTB[1]	CRTB[0]
	CHSTAT_4	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	INTMSK
		-	-	-	-	MODE	DER	DW	DL
		SR	TC	END	ER	SUS	TACT	RQST	EN
	CHCTRL_4	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	CLRINTMSK	SETINTMSK
		-	-	-	-	-	-	CLRSUS	SETSUS
		-	CLRTC	CLREND	CLRRQ	SWRST	STG	CLREN	SETEN
	CHCFG_4	DMS	REN	RSW	RSEL	SBE	-	-	DEM
		-	TM	DAD	SAD	DDS[3]	DDS[2]	DDS[1]	DDS[0]
		SDS[3]	SDS[2]	SDS[1]	SDS[0]	-	AM[2]	AM[1]	AM[0]
		-	LVL	HIEN	LOEN	REQD	SEL[2]	SEL[1]	SEL[0]
	CHITVL_4	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		ITVL[15]	ITVL[14]	ITVL[13]	ITVL[12]	ITVL[11]	ITVL[10]	ITVL[9]	ITVL[8]
		ITVL[7]	ITVL[6]	ITVL[5]	ITVL[4]	ITVL[3]	ITVL[2]	ITVL[1]	ITVL[0]
	CHEXT_4	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		DCA[3]	DCA[2]	DCA[1]	DCA[0]	-	-	-	-
		SCA[3]	SCA[2]	SCA[1]	SCA[0]	-	-	-	-

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Direct memory access controller	NXLA_4	NXLA[31]	NXLA[30]	NXLA[29]	NXLA[28]	NXLA[27]	NXLA[26]	NXLA[25]	NXLA[24]
		NXLA[23]	NXLA[22]	NXLA[21]	NXLA[20]	NXLA[19]	NXLA[18]	NXLA[17]	NXLA[16]
		NXLA[15]	NXLA[14]	NXLA[13]	NXLA[12]	NXLA[11]	NXLA[10]	NXLA[9]	NXLA[8]
		NXLA[7]	NXLA[6]	NXLA[5]	NXLA[4]	NXLA[3]	NXLA[2]	NXLA[1]	NXLA[0]
	CRLA_4	CRLA[31]	CRLA[30]	CRLA[29]	CRLA[28]	CRLA[27]	CRLA[26]	CRLA[25]	CRLA[24]
		CRLA[23]	CRLA[22]	CRLA[21]	CRLA[20]	CRLA[19]	CRLA[18]	CRLA[17]	CRLA[16]
		CRLA[15]	CRLA[14]	CRLA[13]	CRLA[12]	CRLA[11]	CRLA[10]	CRLA[9]	CRLA[8]
		CRLA[7]	CRLA[6]	CRLA[5]	CRLA[4]	CRLA[3]	CRLA[2]	CRLA[1]	CRLA[0]
	NOSA_5	SA[31]	SA[30]	SA[29]	SA[28]	SA[27]	SA[26]	SA[25]	SA[24]
		SA[23]	SA[22]	SA[21]	SA[20]	SA[19]	SA[18]	SA[17]	SA[16]
		SA[15]	SA[14]	SA[13]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	NODA_5	DA[31]	DA[30]	DA[29]	DA[28]	DA[27]	DA[26]	DA[25]	DA[24]
		DA[23]	DA[22]	DA[21]	DA[20]	DA[19]	DA[18]	DA[17]	DA[16]
		DA[15]	DA[14]	DA[13]	DA[12]	DA[11]	DA[10]	DA[9]	DA[8]
		DA[7]	DA[6]	DA[5]	DA[4]	DA[3]	DA[2]	DA[1]	DA[0]
	NOTB_5	TB[31]	TB[30]	TB[29]	TB[28]	TB[27]	TB[26]	TB[25]	TB[24]
		TB[23]	TB[22]	TB[21]	TB[20]	TB[19]	TB[18]	TB[17]	TB[16]
		TB[15]	TB[14]	TB[13]	TB[12]	TB[11]	TB[10]	TB[9]	TB[8]
		TB[7]	TB[6]	TB[5]	TB[4]	TB[3]	TB[2]	TB[1]	TB[0]
	NISA_5	SA[31]	SA[30]	SA[29]	SA[28]	SA[27]	SA[26]	SA[25]	SA[24]
		SA[23]	SA[22]	SA[21]	SA[20]	SA[19]	SA[18]	SA[17]	SA[16]
		SA[15]	SA[14]	SA[13]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	NIDA_5	DA[31]	DA[30]	DA[29]	DA[28]	DA[27]	DA[26]	DA[25]	DA[24]
		DA[23]	DA[22]	DA[21]	DA[20]	DA[19]	DA[18]	DA[17]	DA[16]
		DA[15]	DA[14]	DA[13]	DA[12]	DA[11]	DA[10]	DA[9]	DA[8]
		DA[7]	DA[6]	DA[5]	DA[4]	DA[3]	DA[2]	DA[1]	DA[0]
	N1TB_5	TB[31]	TB[30]	TB[29]	TB[28]	TB[27]	TB[26]	TB[25]	TB[24]
		TB[23]	TB[22]	TB[21]	TB[20]	TB[19]	TB[18]	TB[17]	TB[16]
		TB[15]	TB[14]	TB[13]	TB[12]	TB[11]	TB[10]	TB[9]	TB[8]
		TB[7]	TB[6]	TB[5]	TB[4]	TB[3]	TB[2]	TB[1]	TB[0]
	CRSA_5	CRSA[31]	CRSA[30]	CRSA[29]	CRSA[28]	CRSA[27]	CRSA[26]	CRSA[25]	CRSA[24]
		CRSA[23]	CRSA[22]	CRSA[21]	CRSA[20]	CRSA[19]	CRSA[18]	CRSA[17]	CRSA[16]
		CRSA[15]	CRSA[14]	CRSA[13]	CRSA[12]	CRSA[11]	CRSA[10]	CRSA[9]	CRSA[8]
		CRSA[7]	CRSA[6]	CRSA[5]	CRSA[4]	CRSA[3]	CRSA[2]	CRSA[1]	CRSA[0]
	CRDA_5	CRDA[31]	CRDA[30]	CRDA[29]	CRDA[28]	CRDA[27]	CRDA[26]	CRDA[25]	CRDA[24]
		CRDA[23]	CRDA[22]	CRDA[21]	CRDA[20]	CRDA[19]	CRDA[18]	CRDA[17]	CRDA[16]
		CRDA[15]	CRDA[14]	CRDA[13]	CRDA[12]	CRDA[11]	CRDA[10]	CRDA[9]	CRDA[8]
		CRDA[7]	CRDA[6]	CRDA[5]	CRDA[4]	CRDA[3]	CRDA[2]	CRDA[1]	CRDA[0]
	CRTB_5	CRTB[31]	CRTB[30]	CRTB[29]	CRTB[28]	CRTB[27]	CRTB[26]	CRTB[25]	CRTB[24]
		CRTB[23]	CRTB[22]	CRTB[21]	CRTB[20]	CRTB[19]	CRTB[18]	CRTB[17]	CRTB[16]
		CRTB[15]	CRTB[14]	CRTB[13]	CRTB[12]	CRTB[11]	CRTB[10]	CRTB[9]	CRTB[8]
		CRTB[7]	CRTB[6]	CRTB[5]	CRTB[4]	CRTB[3]	CRTB[2]	CRTB[1]	CRTB[0]
	CHSTAT_5	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	INTMSK
		-	-	-	-	MODE	DER	DW	DL
		SR	TC	END	ER	SUS	TACT	RQST	EN
	CHCTRL_5	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	CLRINTMSK	SETINTMSK
		-	-	-	-	-	-	CLRSUS	SETSUS
		-	CLRTC	CLREND	CLRRQ	SWRST	STG	CLREN	SETEN
	CHCFG_5	DMS	REN	RSW	RSEL	SBE	-	-	DEM
		-	TM	DAD	SAD	DDS[3]	DDS[2]	DDS[1]	DDS[0]
		SDS[3]	SDS[2]	SDS[1]	SDS[0]	-	AM[2]	AM[1]	AM[0]
		-	LVL	HIEN	LOEN	REQD	SEL[2]	SEL[1]	SEL[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Direct memory access controller	CHIITVL_5	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		ITVL[15]	ITVL[14]	ITVL[13]	ITVL[12]	ITVL[11]	ITVL[10]	ITVL[9]	ITVL[8]
		ITVL[7]	ITVL[6]	ITVL[5]	ITVL[4]	ITVL[3]	ITVL[2]	ITVL[1]	ITVL[0]
	CHEXT_5	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		DCA[3]	DCA[2]	DCA[1]	DCA[0]	-	-	-	-
		SCA[3]	SCA[2]	SCA[1]	SCA[0]	-	-	-	-
	NXLA_5	NXLA[31]	NXLA[30]	NXLA[29]	NXLA[28]	NXLA[27]	NXLA[26]	NXLA[25]	NXLA[24]
		NXLA[23]	NXLA[22]	NXLA[21]	NXLA[20]	NXLA[19]	NXLA[18]	NXLA[17]	NXLA[16]
		NXLA[15]	NXLA[14]	NXLA[13]	NXLA[12]	NXLA[11]	NXLA[10]	NXLA[9]	NXLA[8]
		NXLA[7]	NXLA[6]	NXLA[5]	NXLA[4]	NXLA[3]	NXLA[2]	NXLA[1]	NXLA[0]
	CRLA_5	CRLA[31]	CRLA[30]	CRLA[29]	CRLA[28]	CRLA[27]	CRLA[26]	CRLA[25]	CRLA[24]
		CRLA[23]	CRLA[22]	CRLA[21]	CRLA[20]	CRLA[19]	CRLA[18]	CRLA[17]	CRLA[16]
		CRLA[15]	CRLA[14]	CRLA[13]	CRLA[12]	CRLA[11]	CRLA[10]	CRLA[9]	CRLA[8]
		CRLA[7]	CRLA[6]	CRLA[5]	CRLA[4]	CRLA[3]	CRLA[2]	CRLA[1]	CRLA[0]
	NOSA_6	SA[31]	SA[30]	SA[29]	SA[28]	SA[27]	SA[26]	SA[25]	SA[24]
		SA[23]	SA[22]	SA[21]	SA[20]	SA[19]	SA[18]	SA[17]	SA[16]
		SA[15]	SA[14]	SA[13]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	NODA_6	DA[31]	DA[30]	DA[29]	DA[28]	DA[27]	DA[26]	DA[25]	DA[24]
		DA[23]	DA[22]	DA[21]	DA[20]	DA[19]	DA[18]	DA[17]	DA[16]
		DA[15]	DA[14]	DA[13]	DA[12]	DA[11]	DA[10]	DA[9]	DA[8]
		DA[7]	DA[6]	DA[5]	DA[4]	DA[3]	DA[2]	DA[1]	DA[0]
	NOTB_6	TB[31]	TB[30]	TB[29]	TB[28]	TB[27]	TB[26]	TB[25]	TB[24]
		TB[23]	TB[22]	TB[21]	TB[20]	TB[19]	TB[18]	TB[17]	TB[16]
		TB[15]	TB[14]	TB[13]	TB[12]	TB[11]	TB[10]	TB[9]	TB[8]
		TB[7]	TB[6]	TB[5]	TB[4]	TB[3]	TB[2]	TB[1]	TB[0]
	NISA_6	SA[31]	SA[30]	SA[29]	SA[28]	SA[27]	SA[26]	SA[25]	SA[24]
		SA[23]	SA[22]	SA[21]	SA[20]	SA[19]	SA[18]	SA[17]	SA[16]
		SA[15]	SA[14]	SA[13]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	NIDA_6	DA[31]	DA[30]	DA[29]	DA[28]	DA[27]	DA[26]	DA[25]	DA[24]
		DA[23]	DA[22]	DA[21]	DA[20]	DA[19]	DA[18]	DA[17]	DA[16]
		DA[15]	DA[14]	DA[13]	DA[12]	DA[11]	DA[10]	DA[9]	DA[8]
		DA[7]	DA[6]	DA[5]	DA[4]	DA[3]	DA[2]	DA[1]	DA[0]
	N1TB_6	TB[31]	TB[30]	TB[29]	TB[28]	TB[27]	TB[26]	TB[25]	TB[24]
		TB[23]	TB[22]	TB[21]	TB[20]	TB[19]	TB[18]	TB[17]	TB[16]
		TB[15]	TB[14]	TB[13]	TB[12]	TB[11]	TB[10]	TB[9]	TB[8]
		TB[7]	TB[6]	TB[5]	TB[4]	TB[3]	TB[2]	TB[1]	TB[0]
	CRSA_6	CRSA[31]	CRSA[30]	CRSA[29]	CRSA[28]	CRSA[27]	CRSA[26]	CRSA[25]	CRSA[24]
		CRSA[23]	CRSA[22]	CRSA[21]	CRSA[20]	CRSA[19]	CRSA[18]	CRSA[17]	CRSA[16]
		CRSA[15]	CRSA[14]	CRSA[13]	CRSA[12]	CRSA[11]	CRSA[10]	CRSA[9]	CRSA[8]
		CRSA[7]	CRSA[6]	CRSA[5]	CRSA[4]	CRSA[3]	CRSA[2]	CRSA[1]	CRSA[0]
	CRDA_6	CRDA[31]	CRDA[30]	CRDA[29]	CRDA[28]	CRDA[27]	CRDA[26]	CRDA[25]	CRDA[24]
		CRDA[23]	CRDA[22]	CRDA[21]	CRDA[20]	CRDA[19]	CRDA[18]	CRDA[17]	CRDA[16]
		CRDA[15]	CRDA[14]	CRDA[13]	CRDA[12]	CRDA[11]	CRDA[10]	CRDA[9]	CRDA[8]
		CRDA[7]	CRDA[6]	CRDA[5]	CRDA[4]	CRDA[3]	CRDA[2]	CRDA[1]	CRDA[0]
	CRTB_6	CRTB[31]	CRTB[30]	CRTB[29]	CRTB[28]	CRTB[27]	CRTB[26]	CRTB[25]	CRTB[24]
		CRTB[23]	CRTB[22]	CRTB[21]	CRTB[20]	CRTB[19]	CRTB[18]	CRTB[17]	CRTB[16]
		CRTB[15]	CRTB[14]	CRTB[13]	CRTB[12]	CRTB[11]	CRTB[10]	CRTB[9]	CRTB[8]
		CRTB[7]	CRTB[6]	CRTB[5]	CRTB[4]	CRTB[3]	CRTB[2]	CRTB[1]	CRTB[0]
	CHSTAT_6	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	INTMSK
		-	-	-	-	MODE	DER	DW	DL
		SR	TC	END	ER	SUS	TACT	RQST	EN

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Direct memory access controller	CHCTRL_6	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	CLRINTMSK	SETINTMSK
		-	-	-	-	-	-	-	CLRSUS	SETSUS
		-	CLRTC	CLREND	CLRRQ	SWRST	STG	CLREN	SETEN	
	CHCFG_6	DMS	REN	RSW	RSEL	SBE	-	-	-	DEM
		-	TM	DAD	SAD	DDS[3]	DDS[2]	DDS[1]	DDS[0]	
		SDS[3]	SDS[2]	SDS[1]	SDS[0]	-	AM[2]	AM[1]	AM[0]	
		-	LVL	HIEN	LOEN	REQD	SEL[2]	SEL[1]	SEL[0]	
	CHITVL_6	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		ITVL[15]	ITVL[14]	ITVL[13]	ITVL[12]	ITVL[11]	ITVL[10]	ITVL[9]	ITVL[8]	
	CHEXT_6	ITVL[7]	ITVL[6]	ITVL[5]	ITVL[4]	ITVL[3]	ITVL[2]	ITVL[1]	ITVL[0]	
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		DCA[3]	DCA[2]	DCA[1]	DCA[0]	-	-	-	-	-
	NXLA_6	SCA[3]	SCA[2]	SCA[1]	SCA[0]	-	-	-	-	-
		NXLA[31]	NXLA[30]	NXLA[29]	NXLA[28]	NXLA[27]	NXLA[26]	NXLA[25]	NXLA[24]	
		NXLA[23]	NXLA[22]	NXLA[21]	NXLA[20]	NXLA[19]	NXLA[18]	NXLA[17]	NXLA[16]	
		NXLA[15]	NXLA[14]	NXLA[13]	NXLA[12]	NXLA[11]	NXLA[10]	NXLA[9]	NXLA[8]	
	CRLA_6	NXLA[7]	NXLA[6]	NXLA[5]	NXLA[4]	NXLA[3]	NXLA[2]	NXLA[1]	NXLA[0]	
		CRLA[31]	CRLA[30]	CRLA[29]	CRLA[28]	CRLA[27]	CRLA[26]	CRLA[25]	CRLA[24]	
		CRLA[23]	CRLA[22]	CRLA[21]	CRLA[20]	CRLA[19]	CRLA[18]	CRLA[17]	CRLA[16]	
		CRLA[15]	CRLA[14]	CRLA[13]	CRLA[12]	CRLA[11]	CRLA[10]	CRLA[9]	CRLA[8]	
	NOSA_7	CRLA[7]	CRLA[6]	CRLA[5]	CRLA[4]	CRLA[3]	CRLA[2]	CRLA[1]	CRLA[0]	
		SA[31]	SA[30]	SA[29]	SA[28]	SA[27]	SA[26]	SA[25]	SA[24]	
		SA[23]	SA[22]	SA[21]	SA[20]	SA[19]	SA[18]	SA[17]	SA[16]	
		SA[15]	SA[14]	SA[13]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]	
	NODA_7	SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]	
		DA[31]	DA[30]	DA[29]	DA[28]	DA[27]	DA[26]	DA[25]	DA[24]	
		DA[23]	DA[22]	DA[21]	DA[20]	DA[19]	DA[18]	DA[17]	DA[16]	
		DA[15]	DA[14]	DA[13]	DA[12]	DA[11]	DA[10]	DA[9]	DA[8]	
	NOTB_7	DA[7]	DA[6]	DA[5]	DA[4]	DA[3]	DA[2]	DA[1]	DA[0]	
		TB[31]	TB[30]	TB[29]	TB[28]	TB[27]	TB[26]	TB[25]	TB[24]	
		TB[23]	TB[22]	TB[21]	TB[20]	TB[19]	TB[18]	TB[17]	TB[16]	
		TB[15]	TB[14]	TB[13]	TB[12]	TB[11]	TB[10]	TB[9]	TB[8]	
	NISA_7	TB[7]	TB[6]	TB[5]	TB[4]	TB[3]	TB[2]	TB[1]	TB[0]	
		SA[31]	SA[30]	SA[29]	SA[28]	SA[27]	SA[26]	SA[25]	SA[24]	
		SA[23]	SA[22]	SA[21]	SA[20]	SA[19]	SA[18]	SA[17]	SA[16]	
		SA[15]	SA[14]	SA[13]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]	
	NIDA_7	SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]	
		DA[31]	DA[30]	DA[29]	DA[28]	DA[27]	DA[26]	DA[25]	DA[24]	
		DA[23]	DA[22]	DA[21]	DA[20]	DA[19]	DA[18]	DA[17]	DA[16]	
		DA[15]	DA[14]	DA[13]	DA[12]	DA[11]	DA[10]	DA[9]	DA[8]	
	NITB_7	DA[7]	DA[6]	DA[5]	DA[4]	DA[3]	DA[2]	DA[1]	DA[0]	
		TB[31]	TB[30]	TB[29]	TB[28]	TB[27]	TB[26]	TB[25]	TB[24]	
		TB[23]	TB[22]	TB[21]	TB[20]	TB[19]	TB[18]	TB[17]	TB[16]	
		TB[15]	TB[14]	TB[13]	TB[12]	TB[11]	TB[10]	TB[9]	TB[8]	
	CRSA_7	TB[7]	TB[6]	TB[5]	TB[4]	TB[3]	TB[2]	TB[1]	TB[0]	
		CRSA[31]	CRSA[30]	CRSA[29]	CRSA[28]	CRSA[27]	CRSA[26]	CRSA[25]	CRSA[24]	
		CRSA[23]	CRSA[22]	CRSA[21]	CRSA[20]	CRSA[19]	CRSA[18]	CRSA[17]	CRSA[16]	
		CRSA[15]	CRSA[14]	CRSA[13]	CRSA[12]	CRSA[11]	CRSA[10]	CRSA[9]	CRSA[8]	
	CRDA_7	CRSA[7]	CRSA[6]	CRSA[5]	CRSA[4]	CRSA[3]	CRSA[2]	CRSA[1]	CRSA[0]	
CRDA[31]		CRDA[30]	CRDA[29]	CRDA[28]	CRDA[27]	CRDA[26]	CRDA[25]	CRDA[24]		
CRDA[23]		CRDA[22]	CRDA[21]	CRDA[20]	CRDA[19]	CRDA[18]	CRDA[17]	CRDA[16]		
CRDA[15]		CRDA[14]	CRDA[13]	CRDA[12]	CRDA[11]	CRDA[10]	CRDA[9]	CRDA[8]		
CRDA_7	CRDA[7]	CRDA[6]	CRDA[5]	CRDA[4]	CRDA[3]	CRDA[2]	CRDA[1]	CRDA[0]		

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Direct memory access controller	CRTB_7	CRTB[31]	CRTB[30]	CRTB[29]	CRTB[28]	CRTB[27]	CRTB[26]	CRTB[25]	CRTB[24]	
		CRTB[23]	CRTB[22]	CRTB[21]	CRTB[20]	CRTB[19]	CRTB[18]	CRTB[17]	CRTB[16]	
		CRTB[15]	CRTB[14]	CRTB[13]	CRTB[12]	CRTB[11]	CRTB[10]	CRTB[9]	CRTB[8]	
		CRTB[7]	CRTB[6]	CRTB[5]	CRTB[4]	CRTB[3]	CRTB[2]	CRTB[1]	CRTB[0]	
	CHSTAT_7	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	INTMSK
		-	-	-	-	MODE	DER	DW	DL	-
		SR	TC	END	ER	SUS	TACT	RQST	EN	-
	CHCTRL_7	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	CLRINTMSK	SETINTMSK
		-	-	-	-	-	-	-	CLRSUS	SETSUS
		-	CLRTC	CLREND	CLRRQ	SWRST	STG	CLREN	SETEN	-
	CHCFG_7	DMS	REN	RSW	RSEL	SBE	-	-	-	DEM
		-	TM	DAD	SAD	DDS[3]	DDS[2]	DDS[1]	DDS[0]	-
		SDS[3]	SDS[2]	SDS[1]	SDS[0]	-	-	AM[2]	AM[1]	AM[0]
		-	LVL	HIEN	LOEN	REQD	SEL[2]	SEL[1]	SEL[0]	-
	CHITVL_7	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		ITVL[15]	ITVL[14]	ITVL[13]	ITVL[12]	ITVL[11]	ITVL[10]	ITVL[9]	ITVL[8]	-
		ITVL[7]	ITVL[6]	ITVL[5]	ITVL[4]	ITVL[3]	ITVL[2]	ITVL[1]	ITVL[0]	-
	CHEXT_7	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		DCA[3]	DCA[2]	DCA[1]	DCA[0]	-	-	-	-	-
		SCA[3]	SCA[2]	SCA[1]	SCA[0]	-	-	-	-	-
	NXLA_7	NXLA[31]	NXLA[30]	NXLA[29]	NXLA[28]	NXLA[27]	NXLA[26]	NXLA[25]	NXLA[24]	-
		NXLA[23]	NXLA[22]	NXLA[21]	NXLA[20]	NXLA[19]	NXLA[18]	NXLA[17]	NXLA[16]	-
		NXLA[15]	NXLA[14]	NXLA[13]	NXLA[12]	NXLA[11]	NXLA[10]	NXLA[9]	NXLA[8]	-
		NXLA[7]	NXLA[6]	NXLA[5]	NXLA[4]	NXLA[3]	NXLA[2]	NXLA[1]	NXLA[0]	-
	CRLA_7	CRLA[31]	CRLA[30]	CRLA[29]	CRLA[28]	CRLA[27]	CRLA[26]	CRLA[25]	CRLA[24]	-
		CRLA[23]	CRLA[22]	CRLA[21]	CRLA[20]	CRLA[19]	CRLA[18]	CRLA[17]	CRLA[16]	-
		CRLA[15]	CRLA[14]	CRLA[13]	CRLA[12]	CRLA[11]	CRLA[10]	CRLA[9]	CRLA[8]	-
		CRLA[7]	CRLA[6]	CRLA[5]	CRLA[4]	CRLA[3]	CRLA[2]	CRLA[1]	CRLA[0]	-
	DCTRL_0_7	LWCA[3]	LWCA[2]	LWCA[1]	LWCA[0]	-	-	-	-	-
		LDCA[3]	LDCA[2]	LDCA[1]	LDCA[0]	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	LVINT	PR	-
	DSTAT_EN_0_7	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		EN7	EN6	EN5	EN4	EN3	EN2	EN1	EN0	-
	DSTAT_ER_0_7	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		ER7	ER6	ER5	ER4	ER3	ER2	ER1	ER0	-
	DSTAT_END_0_7	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		END7	END6	END5	END4	END3	END2	END1	END0	-
	DSTAT_TC_0_7	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		TC7	TC6	TC5	TC4	TC3	TC2	TC1	TC0	-
	DSTAT_SUS_0_7	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		SUS7	SUS6	SUS5	SUS4	SUS3	SUS2	SUS1	SUS0	-

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Direct memory access controller	NOSA_8	SA[31]	SA[30]	SA[29]	SA[28]	SA[27]	SA[26]	SA[25]	SA[24]
		SA[23]	SA[22]	SA[21]	SA[20]	SA[19]	SA[18]	SA[17]	SA[16]
		SA[15]	SA[14]	SA[13]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	NODA_8	DA[31]	DA[30]	DA[29]	DA[28]	DA[27]	DA[26]	DA[25]	DA[24]
		DA[23]	DA[22]	DA[21]	DA[20]	DA[19]	DA[18]	DA[17]	DA[16]
		DA[15]	DA[14]	DA[13]	DA[12]	DA[11]	DA[10]	DA[9]	DA[8]
		DA[7]	DA[6]	DA[5]	DA[4]	DA[3]	DA[2]	DA[1]	DA[0]
	NOTB_8	TB[31]	TB[30]	TB[29]	TB[28]	TB[27]	TB[26]	TB[25]	TB[24]
		TB[23]	TB[22]	TB[21]	TB[20]	TB[19]	TB[18]	TB[17]	TB[16]
		TB[15]	TB[14]	TB[13]	TB[12]	TB[11]	TB[10]	TB[9]	TB[8]
		TB[7]	TB[6]	TB[5]	TB[4]	TB[3]	TB[2]	TB[1]	TB[0]
	NISA_8	SA[31]	SA[30]	SA[29]	SA[28]	SA[27]	SA[26]	SA[25]	SA[24]
		SA[23]	SA[22]	SA[21]	SA[20]	SA[19]	SA[18]	SA[17]	SA[16]
		SA[15]	SA[14]	SA[13]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	NIDA_8	DA[31]	DA[30]	DA[29]	DA[28]	DA[27]	DA[26]	DA[25]	DA[24]
		DA[23]	DA[22]	DA[21]	DA[20]	DA[19]	DA[18]	DA[17]	DA[16]
		DA[15]	DA[14]	DA[13]	DA[12]	DA[11]	DA[10]	DA[9]	DA[8]
		DA[7]	DA[6]	DA[5]	DA[4]	DA[3]	DA[2]	DA[1]	DA[0]
	NITB_8	TB[31]	TB[30]	TB[29]	TB[28]	TB[27]	TB[26]	TB[25]	TB[24]
		TB[23]	TB[22]	TB[21]	TB[20]	TB[19]	TB[18]	TB[17]	TB[16]
		TB[15]	TB[14]	TB[13]	TB[12]	TB[11]	TB[10]	TB[9]	TB[8]
		TB[7]	TB[6]	TB[5]	TB[4]	TB[3]	TB[2]	TB[1]	TB[0]
	CRSA_8	CRSA[31]	CRSA[30]	CRSA[29]	CRSA[28]	CRSA[27]	CRSA[26]	CRSA[25]	CRSA[24]
		CRSA[23]	CRSA[22]	CRSA[21]	CRSA[20]	CRSA[19]	CRSA[18]	CRSA[17]	CRSA[16]
		CRSA[15]	CRSA[14]	CRSA[13]	CRSA[12]	CRSA[11]	CRSA[10]	CRSA[9]	CRSA[8]
		CRSA[7]	CRSA[6]	CRSA[5]	CRSA[4]	CRSA[3]	CRSA[2]	CRSA[1]	CRSA[0]
	CRDA_8	CRDA[31]	CRDA[30]	CRDA[29]	CRDA[28]	CRDA[27]	CRDA[26]	CRDA[25]	CRDA[24]
		CRDA[23]	CRDA[22]	CRDA[21]	CRDA[20]	CRDA[19]	CRDA[18]	CRDA[17]	CRDA[16]
		CRDA[15]	CRDA[14]	CRDA[13]	CRDA[12]	CRDA[11]	CRDA[10]	CRDA[9]	CRDA[8]
		CRDA[7]	CRDA[6]	CRDA[5]	CRDA[4]	CRDA[3]	CRDA[2]	CRDA[1]	CRDA[0]
	CRTB_8	CRTB[31]	CRTB[30]	CRTB[29]	CRTB[28]	CRTB[27]	CRTB[26]	CRTB[25]	CRTB[24]
		CRTB[23]	CRTB[22]	CRTB[21]	CRTB[20]	CRTB[19]	CRTB[18]	CRTB[17]	CRTB[16]
		CRTB[15]	CRTB[14]	CRTB[13]	CRTB[12]	CRTB[11]	CRTB[10]	CRTB[9]	CRTB[8]
		CRTB[7]	CRTB[6]	CRTB[5]	CRTB[4]	CRTB[3]	CRTB[2]	CRTB[1]	CRTB[0]
	CHSTAT_8	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	INTMSK
		-	-	-	-	MODE	DER	DW	DL
		SR	TC	END	ER	SUS	TACT	RQST	EN
	CHCTRL_8	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	CLRINTMSK	SETINTMSK
		-	-	-	-	-	-	CLRSUS	SETSUS
		-	CLRTC	CLREND	CLRRQ	SWRST	STG	CLREN	SETEN
	CHCFG_8	DMS	REN	RSW	RSEL	SBE	-	-	DEM
		-	TM	DAD	SAD	DDS[3]	DDS[2]	DDS[1]	DDS[0]
		SDS[3]	SDS[2]	SDS[1]	SDS[0]	-	AM[2]	AM[1]	AM[0]
		-	LVL	HIEN	LOEN	REQD	SEL[2]	SEL[1]	SEL[0]
	CHITVL_8	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		ITVL[15]	ITVL[14]	ITVL[13]	ITVL[12]	ITVL[11]	ITVL[10]	ITVL[9]	ITVL[8]
		ITVL[7]	ITVL[6]	ITVL[5]	ITVL[4]	ITVL[3]	ITVL[2]	ITVL[1]	ITVL[0]
	CHEXT_8	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		DCA[3]	DCA[2]	DCA[1]	DCA[0]	-	-	-	-
		SCA[3]	SCA[2]	SCA[1]	SCA[0]	-	-	-	-



Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Direct memory access controller	NXLA_8	NXLA[31]	NXLA[30]	NXLA[29]	NXLA[28]	NXLA[27]	NXLA[26]	NXLA[25]	NXLA[24]
		NXLA[23]	NXLA[22]	NXLA[21]	NXLA[20]	NXLA[19]	NXLA[18]	NXLA[17]	NXLA[16]
		NXLA[15]	NXLA[14]	NXLA[13]	NXLA[12]	NXLA[11]	NXLA[10]	NXLA[9]	NXLA[8]
		NXLA[7]	NXLA[6]	NXLA[5]	NXLA[4]	NXLA[3]	NXLA[2]	NXLA[1]	NXLA[0]
	CRLA_8	CRLA[31]	CRLA[30]	CRLA[29]	CRLA[28]	CRLA[27]	CRLA[26]	CRLA[25]	CRLA[24]
		CRLA[23]	CRLA[22]	CRLA[21]	CRLA[20]	CRLA[19]	CRLA[18]	CRLA[17]	CRLA[16]
		CRLA[15]	CRLA[14]	CRLA[13]	CRLA[12]	CRLA[11]	CRLA[10]	CRLA[9]	CRLA[8]
		CRLA[7]	CRLA[6]	CRLA[5]	CRLA[4]	CRLA[3]	CRLA[2]	CRLA[1]	CRLA[0]
	NOSA_9	SA[31]	SA[30]	SA[29]	SA[28]	SA[27]	SA[26]	SA[25]	SA[24]
		SA[23]	SA[22]	SA[21]	SA[20]	SA[19]	SA[18]	SA[17]	SA[16]
		SA[15]	SA[14]	SA[13]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	NODA_9	DA[31]	DA[30]	DA[29]	DA[28]	DA[27]	DA[26]	DA[25]	DA[24]
		DA[23]	DA[22]	DA[21]	DA[20]	DA[19]	DA[18]	DA[17]	DA[16]
		DA[15]	DA[14]	DA[13]	DA[12]	DA[11]	DA[10]	DA[9]	DA[8]
		DA[7]	DA[6]	DA[5]	DA[4]	DA[3]	DA[2]	DA[1]	DA[0]
	NOTB_9	TB[31]	TB[30]	TB[29]	TB[28]	TB[27]	TB[26]	TB[25]	TB[24]
		TB[23]	TB[22]	TB[21]	TB[20]	TB[19]	TB[18]	TB[17]	TB[16]
		TB[15]	TB[14]	TB[13]	TB[12]	TB[11]	TB[10]	TB[9]	TB[8]
		TB[7]	TB[6]	TB[5]	TB[4]	TB[3]	TB[2]	TB[1]	TB[0]
	NISA_9	SA[31]	SA[30]	SA[29]	SA[28]	SA[27]	SA[26]	SA[25]	SA[24]
		SA[23]	SA[22]	SA[21]	SA[20]	SA[19]	SA[18]	SA[17]	SA[16]
		SA[15]	SA[14]	SA[13]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	NIDA_9	DA[31]	DA[30]	DA[29]	DA[28]	DA[27]	DA[26]	DA[25]	DA[24]
		DA[23]	DA[22]	DA[21]	DA[20]	DA[19]	DA[18]	DA[17]	DA[16]
		DA[15]	DA[14]	DA[13]	DA[12]	DA[11]	DA[10]	DA[9]	DA[8]
		DA[7]	DA[6]	DA[5]	DA[4]	DA[3]	DA[2]	DA[1]	DA[0]
	N1TB_9	TB[31]	TB[30]	TB[29]	TB[28]	TB[27]	TB[26]	TB[25]	TB[24]
		TB[23]	TB[22]	TB[21]	TB[20]	TB[19]	TB[18]	TB[17]	TB[16]
		TB[15]	TB[14]	TB[13]	TB[12]	TB[11]	TB[10]	TB[9]	TB[8]
		TB[7]	TB[6]	TB[5]	TB[4]	TB[3]	TB[2]	TB[1]	TB[0]
	CRSA_9	CRSA[31]	CRSA[30]	CRSA[29]	CRSA[28]	CRSA[27]	CRSA[26]	CRSA[25]	CRSA[24]
		CRSA[23]	CRSA[22]	CRSA[21]	CRSA[20]	CRSA[19]	CRSA[18]	CRSA[17]	CRSA[16]
		CRSA[15]	CRSA[14]	CRSA[13]	CRSA[12]	CRSA[11]	CRSA[10]	CRSA[9]	CRSA[8]
		CRSA[7]	CRSA[6]	CRSA[5]	CRSA[4]	CRSA[3]	CRSA[2]	CRSA[1]	CRSA[0]
	CRDA_9	CRDA[31]	CRDA[30]	CRDA[29]	CRDA[28]	CRDA[27]	CRDA[26]	CRDA[25]	CRDA[24]
		CRDA[23]	CRDA[22]	CRDA[21]	CRDA[20]	CRDA[19]	CRDA[18]	CRDA[17]	CRDA[16]
		CRDA[15]	CRDA[14]	CRDA[13]	CRDA[12]	CRDA[11]	CRDA[10]	CRDA[9]	CRDA[8]
		CRDA[7]	CRDA[6]	CRDA[5]	CRDA[4]	CRDA[3]	CRDA[2]	CRDA[1]	CRDA[0]
	CRTB_9	CRTB[31]	CRTB[30]	CRTB[29]	CRTB[28]	CRTB[27]	CRTB[26]	CRTB[25]	CRTB[24]
		CRTB[23]	CRTB[22]	CRTB[21]	CRTB[20]	CRTB[19]	CRTB[18]	CRTB[17]	CRTB[16]
		CRTB[15]	CRTB[14]	CRTB[13]	CRTB[12]	CRTB[11]	CRTB[10]	CRTB[9]	CRTB[8]
		CRTB[7]	CRTB[6]	CRTB[5]	CRTB[4]	CRTB[3]	CRTB[2]	CRTB[1]	CRTB[0]
	CHSTAT_9	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	INTMSK
		-	-	-	-	MODE	DER	DW	DL
		SR	TC	END	ER	SUS	TACT	RQST	EN
	CHCTRL_9	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	CLRINTMSK	SETINTMSK
		-	-	-	-	-	-	CLRSUS	SETSUS
		-	CLRTC	CLREND	CLRRQ	SWRST	STG	CLREN	SETEN
	CHCFG_9	DMS	REN	RSW	RSEL	SBE	-	-	DEM
		-	TM	DAD	SAD	DDS[3]	DDS[2]	DDS[1]	DDS[0]
		SDS[3]	SDS[2]	SDS[1]	SDS[0]	-	AM[2]	AM[1]	AM[0]
		-	LVL	HIEN	LOEN	REQD	SEL[2]	SEL[1]	SEL[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Direct memory access controller	CHIITVL_9	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		ITVL[15]	ITVL[14]	ITVL[13]	ITVL[12]	ITVL[11]	ITVL[10]	ITVL[9]	ITVL[8]
		ITVL[7]	ITVL[6]	ITVL[5]	ITVL[4]	ITVL[3]	ITVL[2]	ITVL[1]	ITVL[0]
	CHEXT_9	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		DCA[3]	DCA[2]	DCA[1]	DCA[0]	-	-	-	-
		SCA[3]	SCA[2]	SCA[1]	SCA[0]	-	-	-	-
	NXLA_9	NXLA[31]	NXLA[30]	NXLA[29]	NXLA[28]	NXLA[27]	NXLA[26]	NXLA[25]	NXLA[24]
		NXLA[23]	NXLA[22]	NXLA[21]	NXLA[20]	NXLA[19]	NXLA[18]	NXLA[17]	NXLA[16]
		NXLA[15]	NXLA[14]	NXLA[13]	NXLA[12]	NXLA[11]	NXLA[10]	NXLA[9]	NXLA[8]
		NXLA[7]	NXLA[6]	NXLA[5]	NXLA[4]	NXLA[3]	NXLA[2]	NXLA[1]	NXLA[0]
	CRLA_9	CRLA[31]	CRLA[30]	CRLA[29]	CRLA[28]	CRLA[27]	CRLA[26]	CRLA[25]	CRLA[24]
		CRLA[23]	CRLA[22]	CRLA[21]	CRLA[20]	CRLA[19]	CRLA[18]	CRLA[17]	CRLA[16]
		CRLA[15]	CRLA[14]	CRLA[13]	CRLA[12]	CRLA[11]	CRLA[10]	CRLA[9]	CRLA[8]
		CRLA[7]	CRLA[6]	CRLA[5]	CRLA[4]	CRLA[3]	CRLA[2]	CRLA[1]	CRLA[0]
	NOSA_10	SA[31]	SA[30]	SA[29]	SA[28]	SA[27]	SA[26]	SA[25]	SA[24]
		SA[23]	SA[22]	SA[21]	SA[20]	SA[19]	SA[18]	SA[17]	SA[16]
		SA[15]	SA[14]	SA[13]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	NODA_10	DA[31]	DA[30]	DA[29]	DA[28]	DA[27]	DA[26]	DA[25]	DA[24]
		DA[23]	DA[22]	DA[21]	DA[20]	DA[19]	DA[18]	DA[17]	DA[16]
		DA[15]	DA[14]	DA[13]	DA[12]	DA[11]	DA[10]	DA[9]	DA[8]
		DA[7]	DA[6]	DA[5]	DA[4]	DA[3]	DA[2]	DA[1]	DA[0]
	NOTB_10	TB[31]	TB[30]	TB[29]	TB[28]	TB[27]	TB[26]	TB[25]	TB[24]
		TB[23]	TB[22]	TB[21]	TB[20]	TB[19]	TB[18]	TB[17]	TB[16]
		TB[15]	TB[14]	TB[13]	TB[12]	TB[11]	TB[10]	TB[9]	TB[8]
		TB[7]	TB[6]	TB[5]	TB[4]	TB[3]	TB[2]	TB[1]	TB[0]
	NISA_10	SA[31]	SA[30]	SA[29]	SA[28]	SA[27]	SA[26]	SA[25]	SA[24]
		SA[23]	SA[22]	SA[21]	SA[20]	SA[19]	SA[18]	SA[17]	SA[16]
		SA[15]	SA[14]	SA[13]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	NIDA_10	DA[31]	DA[30]	DA[29]	DA[28]	DA[27]	DA[26]	DA[25]	DA[24]
		DA[23]	DA[22]	DA[21]	DA[20]	DA[19]	DA[18]	DA[17]	DA[16]
		DA[15]	DA[14]	DA[13]	DA[12]	DA[11]	DA[10]	DA[9]	DA[8]
		DA[7]	DA[6]	DA[5]	DA[4]	DA[3]	DA[2]	DA[1]	DA[0]
	N1TB_10	TB[31]	TB[30]	TB[29]	TB[28]	TB[27]	TB[26]	TB[25]	TB[24]
		TB[23]	TB[22]	TB[21]	TB[20]	TB[19]	TB[18]	TB[17]	TB[16]
		TB[15]	TB[14]	TB[13]	TB[12]	TB[11]	TB[10]	TB[9]	TB[8]
		TB[7]	TB[6]	TB[5]	TB[4]	TB[3]	TB[2]	TB[1]	TB[0]
	CRSA_10	CRSA[31]	CRSA[30]	CRSA[29]	CRSA[28]	CRSA[27]	CRSA[26]	CRSA[25]	CRSA[24]
		CRSA[23]	CRSA[22]	CRSA[21]	CRSA[20]	CRSA[19]	CRSA[18]	CRSA[17]	CRSA[16]
		CRSA[15]	CRSA[14]	CRSA[13]	CRSA[12]	CRSA[11]	CRSA[10]	CRSA[9]	CRSA[8]
		CRSA[7]	CRSA[6]	CRSA[5]	CRSA[4]	CRSA[3]	CRSA[2]	CRSA[1]	CRSA[0]
	CRDA_10	CRDA[31]	CRDA[30]	CRDA[29]	CRDA[28]	CRDA[27]	CRDA[26]	CRDA[25]	CRDA[24]
		CRDA[23]	CRDA[22]	CRDA[21]	CRDA[20]	CRDA[19]	CRDA[18]	CRDA[17]	CRDA[16]
		CRDA[15]	CRDA[14]	CRDA[13]	CRDA[12]	CRDA[11]	CRDA[10]	CRDA[9]	CRDA[8]
		CRDA[7]	CRDA[6]	CRDA[5]	CRDA[4]	CRDA[3]	CRDA[2]	CRDA[1]	CRDA[0]
	CRTB_10	CRTB[31]	CRTB[30]	CRTB[29]	CRTB[28]	CRTB[27]	CRTB[26]	CRTB[25]	CRTB[24]
		CRTB[23]	CRTB[22]	CRTB[21]	CRTB[20]	CRTB[19]	CRTB[18]	CRTB[17]	CRTB[16]
		CRTB[15]	CRTB[14]	CRTB[13]	CRTB[12]	CRTB[11]	CRTB[10]	CRTB[9]	CRTB[8]
		CRTB[7]	CRTB[6]	CRTB[5]	CRTB[4]	CRTB[3]	CRTB[2]	CRTB[1]	CRTB[0]
	CHSTAT_10	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	INTMSK
		-	-	-	-	MODE	DER	DW	DL
		SR	TC	END	ER	SUS	TACT	RQST	EN

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0		
Direct memory access controller	CHCTRL_10	-	-	-	-	-	-	-	-		
		-	-	-	-	-	-	-	CLRINTMSK	SETINTMSK	
		-	-	-	-	-	-	-	-	CLRSUS	SETSUS
		-	CLRTC	CLREND	CLRRQ	SWRST	STG	CLREN	SETEN		
	CHCFG_10	DMS	REN	RSW	RSEL	SBE	-	-	-	DEM	
		-	TM	DAD	SAD	DDS[3]	DDS[2]	DDS[1]	DDS[0]		
		SDS[3]	SDS[2]	SDS[1]	SDS[0]	-	AM[2]	AM[1]	AM[0]		
		-	LVL	HIEN	LOEN	REQD	SEL[2]	SEL[1]	SEL[0]		
	CHITVL_10	-	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	-	
		ITVL[15]	ITVL[14]	ITVL[13]	ITVL[12]	ITVL[11]	ITVL[10]	ITVL[9]	ITVL[8]		
	CHEXT_10	ITVL[7]	ITVL[6]	ITVL[5]	ITVL[4]	ITVL[3]	ITVL[2]	ITVL[1]	ITVL[0]		
		-	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	-	
		DCA[3]	DCA[2]	DCA[1]	DCA[0]	-	-	-	-	-	
	NXLA_10	SCA[3]	SCA[2]	SCA[1]	SCA[0]	-	-	-	-	-	
		NXLA[31]	NXLA[30]	NXLA[29]	NXLA[28]	NXLA[27]	NXLA[26]	NXLA[25]	NXLA[24]		
		NXLA[23]	NXLA[22]	NXLA[21]	NXLA[20]	NXLA[19]	NXLA[18]	NXLA[17]	NXLA[16]		
		NXLA[15]	NXLA[14]	NXLA[13]	NXLA[12]	NXLA[11]	NXLA[10]	NXLA[9]	NXLA[8]		
	CRLA_10	NXLA[7]	NXLA[6]	NXLA[5]	NXLA[4]	NXLA[3]	NXLA[2]	NXLA[1]	NXLA[0]		
		CRLA[31]	CRLA[30]	CRLA[29]	CRLA[28]	CRLA[27]	CRLA[26]	CRLA[25]	CRLA[24]		
		CRLA[23]	CRLA[22]	CRLA[21]	CRLA[20]	CRLA[19]	CRLA[18]	CRLA[17]	CRLA[16]		
		CRLA[15]	CRLA[14]	CRLA[13]	CRLA[12]	CRLA[11]	CRLA[10]	CRLA[9]	CRLA[8]		
	NOSA_11	CRLA[7]	CRLA[6]	CRLA[5]	CRLA[4]	CRLA[3]	CRLA[2]	CRLA[1]	CRLA[0]		
		SA[31]	SA[30]	SA[29]	SA[28]	SA[27]	SA[26]	SA[25]	SA[24]		
		SA[23]	SA[22]	SA[21]	SA[20]	SA[19]	SA[18]	SA[17]	SA[16]		
		SA[15]	SA[14]	SA[13]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]		
	NODA_11	SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]		
		DA[31]	DA[30]	DA[29]	DA[28]	DA[27]	DA[26]	DA[25]	DA[24]		
		DA[23]	DA[22]	DA[21]	DA[20]	DA[19]	DA[18]	DA[17]	DA[16]		
		DA[15]	DA[14]	DA[13]	DA[12]	DA[11]	DA[10]	DA[9]	DA[8]		
	NOTB_11	DA[7]	DA[6]	DA[5]	DA[4]	DA[3]	DA[2]	DA[1]	DA[0]		
		TB[31]	TB[30]	TB[29]	TB[28]	TB[27]	TB[26]	TB[25]	TB[24]		
		TB[23]	TB[22]	TB[21]	TB[20]	TB[19]	TB[18]	TB[17]	TB[16]		
		TB[15]	TB[14]	TB[13]	TB[12]	TB[11]	TB[10]	TB[9]	TB[8]		
	NISA_11	TB[7]	TB[6]	TB[5]	TB[4]	TB[3]	TB[2]	TB[1]	TB[0]		
		SA[31]	SA[30]	SA[29]	SA[28]	SA[27]	SA[26]	SA[25]	SA[24]		
		SA[23]	SA[22]	SA[21]	SA[20]	SA[19]	SA[18]	SA[17]	SA[16]		
		SA[15]	SA[14]	SA[13]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]		
	NIDA_11	SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]		
		DA[31]	DA[30]	DA[29]	DA[28]	DA[27]	DA[26]	DA[25]	DA[24]		
		DA[23]	DA[22]	DA[21]	DA[20]	DA[19]	DA[18]	DA[17]	DA[16]		
		DA[15]	DA[14]	DA[13]	DA[12]	DA[11]	DA[10]	DA[9]	DA[8]		
	NITB_11	DA[7]	DA[6]	DA[5]	DA[4]	DA[3]	DA[2]	DA[1]	DA[0]		
		TB[31]	TB[30]	TB[29]	TB[28]	TB[27]	TB[26]	TB[25]	TB[24]		
		TB[23]	TB[22]	TB[21]	TB[20]	TB[19]	TB[18]	TB[17]	TB[16]		
		TB[15]	TB[14]	TB[13]	TB[12]	TB[11]	TB[10]	TB[9]	TB[8]		
	CRSA_11	TB[7]	TB[6]	TB[5]	TB[4]	TB[3]	TB[2]	TB[1]	TB[0]		
		CRSA[31]	CRSA[30]	CRSA[29]	CRSA[28]	CRSA[27]	CRSA[26]	CRSA[25]	CRSA[24]		
		CRSA[23]	CRSA[22]	CRSA[21]	CRSA[20]	CRSA[19]	CRSA[18]	CRSA[17]	CRSA[16]		
CRSA[15]		CRSA[14]	CRSA[13]	CRSA[12]	CRSA[11]	CRSA[10]	CRSA[9]	CRSA[8]			
CRDA_11	CRSA[7]	CRSA[6]	CRSA[5]	CRSA[4]	CRSA[3]	CRSA[2]	CRSA[1]	CRSA[0]			
	CRDA[31]	CRDA[30]	CRDA[29]	CRDA[28]	CRDA[27]	CRDA[26]	CRDA[25]	CRDA[24]			
	CRDA[23]	CRDA[22]	CRDA[21]	CRDA[20]	CRDA[19]	CRDA[18]	CRDA[17]	CRDA[16]			
	CRDA[15]	CRDA[14]	CRDA[13]	CRDA[12]	CRDA[11]	CRDA[10]	CRDA[9]	CRDA[8]			
CRDA_11	CRDA[7]	CRDA[6]	CRDA[5]	CRDA[4]	CRDA[3]	CRDA[2]	CRDA[1]	CRDA[0]			

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Direct memory access controller	CRTB_11	CRTB[31]	CRTB[30]	CRTB[29]	CRTB[28]	CRTB[27]	CRTB[26]	CRTB[25]	CRTB[24]	
		CRTB[23]	CRTB[22]	CRTB[21]	CRTB[20]	CRTB[19]	CRTB[18]	CRTB[17]	CRTB[16]	
		CRTB[15]	CRTB[14]	CRTB[13]	CRTB[12]	CRTB[11]	CRTB[10]	CRTB[9]	CRTB[8]	
		CRTB[7]	CRTB[6]	CRTB[5]	CRTB[4]	CRTB[3]	CRTB[2]	CRTB[1]	CRTB[0]	
	CHSTAT_11	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	INTMSK
		-	-	-	-	MODE	DER	DW	DL	-
		SR	TC	END	ER	SUS	TACT	RQST	EN	-
	CHCTRL_11	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	CLRINTMSK	SETINTMSK
		-	-	-	-	-	-	-	CLRSUS	SETSUS
		-	CLRTC	CLREND	CLRRQ	SWRST	STG	CLREN	SETEN	-
	CHCFG_11	DMS	REN	RSW	RSEL	SBE	-	-	-	DEM
		-	TM	DAD	SAD	DDS[3]	DDS[2]	DDS[1]	DDS[0]	-
		SDS[3]	SDS[2]	SDS[1]	SDS[0]	-	-	AM[2]	AM[1]	AM[0]
		-	LVL	HIEN	LOEN	REQD	SEL[2]	SEL[1]	SEL[0]	-
	CHITVL_11	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		ITVL[15]	ITVL[14]	ITVL[13]	ITVL[12]	ITVL[11]	ITVL[10]	ITVL[9]	ITVL[8]	-
		ITVL[7]	ITVL[6]	ITVL[5]	ITVL[4]	ITVL[3]	ITVL[2]	ITVL[1]	ITVL[0]	-
	CHEXT_11	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		DCA[3]	DCA[2]	DCA[1]	DCA[0]	-	-	-	-	-
		SCA[3]	SCA[2]	SCA[1]	SCA[0]	-	-	-	-	-
	NXLA_11	NXLA[31]	NXLA[30]	NXLA[29]	NXLA[28]	NXLA[27]	NXLA[26]	NXLA[25]	NXLA[24]	-
		NXLA[23]	NXLA[22]	NXLA[21]	NXLA[20]	NXLA[19]	NXLA[18]	NXLA[17]	NXLA[16]	-
		NXLA[15]	NXLA[14]	NXLA[13]	NXLA[12]	NXLA[11]	NXLA[10]	NXLA[9]	NXLA[8]	-
		NXLA[7]	NXLA[6]	NXLA[5]	NXLA[4]	NXLA[3]	NXLA[2]	NXLA[1]	NXLA[0]	-
	CRLA_11	CRLA[31]	CRLA[30]	CRLA[29]	CRLA[28]	CRLA[27]	CRLA[26]	CRLA[25]	CRLA[24]	-
		CRLA[23]	CRLA[22]	CRLA[21]	CRLA[20]	CRLA[19]	CRLA[18]	CRLA[17]	CRLA[16]	-
		CRLA[15]	CRLA[14]	CRLA[13]	CRLA[12]	CRLA[11]	CRLA[10]	CRLA[9]	CRLA[8]	-
		CRLA[7]	CRLA[6]	CRLA[5]	CRLA[4]	CRLA[3]	CRLA[2]	CRLA[1]	CRLA[0]	-
	NOSA_12	SA[31]	SA[30]	SA[29]	SA[28]	SA[27]	SA[26]	SA[25]	SA[24]	-
		SA[23]	SA[22]	SA[21]	SA[20]	SA[19]	SA[18]	SA[17]	SA[16]	-
		SA[15]	SA[14]	SA[13]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]	-
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]	-
	NODA_12	DA[31]	DA[30]	DA[29]	DA[28]	DA[27]	DA[26]	DA[25]	DA[24]	-
		DA[23]	DA[22]	DA[21]	DA[20]	DA[19]	DA[18]	DA[17]	DA[16]	-
		DA[15]	DA[14]	DA[13]	DA[12]	DA[11]	DA[10]	DA[9]	DA[8]	-
		DA[7]	DA[6]	DA[5]	DA[4]	DA[3]	DA[2]	DA[1]	DA[0]	-
	NOTB_12	TB[31]	TB[30]	TB[29]	TB[28]	TB[27]	TB[26]	TB[25]	TB[24]	-
		TB[23]	TB[22]	TB[21]	TB[20]	TB[19]	TB[18]	TB[17]	TB[16]	-
		TB[15]	TB[14]	TB[13]	TB[12]	TB[11]	TB[10]	TB[9]	TB[8]	-
		TB[7]	TB[6]	TB[5]	TB[4]	TB[3]	TB[2]	TB[1]	TB[0]	-
	NISA_12	SA[31]	SA[30]	SA[29]	SA[28]	SA[27]	SA[26]	SA[25]	SA[24]	-
		SA[23]	SA[22]	SA[21]	SA[20]	SA[19]	SA[18]	SA[17]	SA[16]	-
		SA[15]	SA[14]	SA[13]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]	-
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]	-
	N1DA_12	DA[31]	DA[30]	DA[29]	DA[28]	DA[27]	DA[26]	DA[25]	DA[24]	-
		DA[23]	DA[22]	DA[21]	DA[20]	DA[19]	DA[18]	DA[17]	DA[16]	-
		DA[15]	DA[14]	DA[13]	DA[12]	DA[11]	DA[10]	DA[9]	DA[8]	-
		DA[7]	DA[6]	DA[5]	DA[4]	DA[3]	DA[2]	DA[1]	DA[0]	-
	N1TB_12	TB[31]	TB[30]	TB[29]	TB[28]	TB[27]	TB[26]	TB[25]	TB[24]	-
		TB[23]	TB[22]	TB[21]	TB[20]	TB[19]	TB[18]	TB[17]	TB[16]	-
		TB[15]	TB[14]	TB[13]	TB[12]	TB[11]	TB[10]	TB[9]	TB[8]	-
		TB[7]	TB[6]	TB[5]	TB[4]	TB[3]	TB[2]	TB[1]	TB[0]	-

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Direct memory access controller	CRSA_12	CRSA[31]	CRSA[30]	CRSA[29]	CRSA[28]	CRSA[27]	CRSA[26]	CRSA[25]	CRSA[24]
		CRSA[23]	CRSA[22]	CRSA[21]	CRSA[20]	CRSA[19]	CRSA[18]	CRSA[17]	CRSA[16]
		CRSA[15]	CRSA[14]	CRSA[13]	CRSA[12]	CRSA[11]	CRSA[10]	CRSA[9]	CRSA[8]
		CRSA[7]	CRSA[6]	CRSA[5]	CRSA[4]	CRSA[3]	CRSA[2]	CRSA[1]	CRSA[0]
	CRDA_12	CRDA[31]	CRDA[30]	CRDA[29]	CRDA[28]	CRDA[27]	CRDA[26]	CRDA[25]	CRDA[24]
		CRDA[23]	CRDA[22]	CRDA[21]	CRDA[20]	CRDA[19]	CRDA[18]	CRDA[17]	CRDA[16]
		CRDA[15]	CRDA[14]	CRDA[13]	CRDA[12]	CRDA[11]	CRDA[10]	CRDA[9]	CRDA[8]
		CRDA[7]	CRDA[6]	CRDA[5]	CRDA[4]	CRDA[3]	CRDA[2]	CRDA[1]	CRDA[0]
	CRTB_12	CRTB[31]	CRTB[30]	CRTB[29]	CRTB[28]	CRTB[27]	CRTB[26]	CRTB[25]	CRTB[24]
		CRTB[23]	CRTB[22]	CRTB[21]	CRTB[20]	CRTB[19]	CRTB[18]	CRTB[17]	CRTB[16]
		CRTB[15]	CRTB[14]	CRTB[13]	CRTB[12]	CRTB[11]	CRTB[10]	CRTB[9]	CRTB[8]
		CRTB[7]	CRTB[6]	CRTB[5]	CRTB[4]	CRTB[3]	CRTB[2]	CRTB[1]	CRTB[0]
	CHSTAT_12	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	INTMSK
		-	-	-	-	MODE	DER	DW	DL
		SR	TC	END	ER	SUS	TACT	RQST	EN
	CHCTRL_12	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	CLRINTMSK	SETINTMSK
		-	-	-	-	-	-	CLRSUS	SETSUS
		-	CLRTC	CLREND	CLRRQ	SWRST	STG	CLREN	SETEN
	CHCFG_12	DMS	REN	RSW	RSEL	SBE	-	-	DEM
		-	TM	DAD	SAD	DDS[3]	DDS[2]	DDS[1]	DDS[0]
		SDS[3]	SDS[2]	SDS[1]	SDS[0]	-	AM[2]	AM[1]	AM[0]
		-	LVL	HIEN	LOEN	REQD	SEL[2]	SEL[1]	SEL[0]
	CHITVL_12	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		ITVL[15]	ITVL[14]	ITVL[13]	ITVL[12]	ITVL[11]	ITVL[10]	ITVL[9]	ITVL[8]
		ITVL[7]	ITVL[6]	ITVL[5]	ITVL[4]	ITVL[3]	ITVL[2]	ITVL[1]	ITVL[0]
	CHEXT_12	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		DCA[3]	DCA[2]	DCA[1]	DCA[0]	-	-	-	-
		SCA[3]	SCA[2]	SCA[1]	SCA[0]	-	-	-	-
	NXLA_12	NXLA[31]	NXLA[30]	NXLA[29]	NXLA[28]	NXLA[27]	NXLA[26]	NXLA[25]	NXLA[24]
		NXLA[23]	NXLA[22]	NXLA[21]	NXLA[20]	NXLA[19]	NXLA[18]	NXLA[17]	NXLA[16]
		NXLA[15]	NXLA[14]	NXLA[13]	NXLA[12]	NXLA[11]	NXLA[10]	NXLA[9]	NXLA[8]
		NXLA[7]	NXLA[6]	NXLA[5]	NXLA[4]	NXLA[3]	NXLA[2]	NXLA[1]	NXLA[0]
	CRLA_12	CRLA[31]	CRLA[30]	CRLA[29]	CRLA[28]	CRLA[27]	CRLA[26]	CRLA[25]	CRLA[24]
		CRLA[23]	CRLA[22]	CRLA[21]	CRLA[20]	CRLA[19]	CRLA[18]	CRLA[17]	CRLA[16]
		CRLA[15]	CRLA[14]	CRLA[13]	CRLA[12]	CRLA[11]	CRLA[10]	CRLA[9]	CRLA[8]
		CRLA[7]	CRLA[6]	CRLA[5]	CRLA[4]	CRLA[3]	CRLA[2]	CRLA[1]	CRLA[0]
	NOSA_13	SA[31]	SA[30]	SA[29]	SA[28]	SA[27]	SA[26]	SA[25]	SA[24]
		SA[23]	SA[22]	SA[21]	SA[20]	SA[19]	SA[18]	SA[17]	SA[16]
		SA[15]	SA[14]	SA[13]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	NODA_13	DA[31]	DA[30]	DA[29]	DA[28]	DA[27]	DA[26]	DA[25]	DA[24]
		DA[23]	DA[22]	DA[21]	DA[20]	DA[19]	DA[18]	DA[17]	DA[16]
		DA[15]	DA[14]	DA[13]	DA[12]	DA[11]	DA[10]	DA[9]	DA[8]
		DA[7]	DA[6]	DA[5]	DA[4]	DA[3]	DA[2]	DA[1]	DA[0]
	NOTB_13	TB[31]	TB[30]	TB[29]	TB[28]	TB[27]	TB[26]	TB[25]	TB[24]
		TB[23]	TB[22]	TB[21]	TB[20]	TB[19]	TB[18]	TB[17]	TB[16]
		TB[15]	TB[14]	TB[13]	TB[12]	TB[11]	TB[10]	TB[9]	TB[8]
		TB[7]	TB[6]	TB[5]	TB[4]	TB[3]	TB[2]	TB[1]	TB[0]
	NISA_13	SA[31]	SA[30]	SA[29]	SA[28]	SA[27]	SA[26]	SA[25]	SA[24]
		SA[23]	SA[22]	SA[21]	SA[20]	SA[19]	SA[18]	SA[17]	SA[16]
		SA[15]	SA[14]	SA[13]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Direct memory access controller	N1DA_13	DA[31]	DA[30]	DA[29]	DA[28]	DA[27]	DA[26]	DA[25]	DA[24]
		DA[23]	DA[22]	DA[21]	DA[20]	DA[19]	DA[18]	DA[17]	DA[16]
		DA[15]	DA[14]	DA[13]	DA[12]	DA[11]	DA[10]	DA[9]	DA[8]
		DA[7]	DA[6]	DA[5]	DA[4]	DA[3]	DA[2]	DA[1]	DA[0]
	N1TB_13	TB[31]	TB[30]	TB[29]	TB[28]	TB[27]	TB[26]	TB[25]	TB[24]
		TB[23]	TB[22]	TB[21]	TB[20]	TB[19]	TB[18]	TB[17]	TB[16]
		TB[15]	TB[14]	TB[13]	TB[12]	TB[11]	TB[10]	TB[9]	TB[8]
		TB[7]	TB[6]	TB[5]	TB[4]	TB[3]	TB[2]	TB[1]	TB[0]
	CRSA_13	CRSA[31]	CRSA[30]	CRSA[29]	CRSA[28]	CRSA[27]	CRSA[26]	CRSA[25]	CRSA[24]
		CRSA[23]	CRSA[22]	CRSA[21]	CRSA[20]	CRSA[19]	CRSA[18]	CRSA[17]	CRSA[16]
		CRSA[15]	CRSA[14]	CRSA[13]	CRSA[12]	CRSA[11]	CRSA[10]	CRSA[9]	CRSA[8]
		CRSA[7]	CRSA[6]	CRSA[5]	CRSA[4]	CRSA[3]	CRSA[2]	CRSA[1]	CRSA[0]
	CRDA_13	CRDA[31]	CRDA[30]	CRDA[29]	CRDA[28]	CRDA[27]	CRDA[26]	CRDA[25]	CRDA[24]
		CRDA[23]	CRDA[22]	CRDA[21]	CRDA[20]	CRDA[19]	CRDA[18]	CRDA[17]	CRDA[16]
		CRDA[15]	CRDA[14]	CRDA[13]	CRDA[12]	CRDA[11]	CRDA[10]	CRDA[9]	CRDA[8]
		CRDA[7]	CRDA[6]	CRDA[5]	CRDA[4]	CRDA[3]	CRDA[2]	CRDA[1]	CRDA[0]
	CRTB_13	CRTB[31]	CRTB[30]	CRTB[29]	CRTB[28]	CRTB[27]	CRTB[26]	CRTB[25]	CRTB[24]
		CRTB[23]	CRTB[22]	CRTB[21]	CRTB[20]	CRTB[19]	CRTB[18]	CRTB[17]	CRTB[16]
		CRTB[15]	CRTB[14]	CRTB[13]	CRTB[12]	CRTB[11]	CRTB[10]	CRTB[9]	CRTB[8]
		CRTB[7]	CRTB[6]	CRTB[5]	CRTB[4]	CRTB[3]	CRTB[2]	CRTB[1]	CRTB[0]
	CHSTAT_13	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	INTMSK
		-	-	-	-	MODE	DER	DW	DL
		SR	TC	END	ER	SUS	TACT	RQST	EN
	CHCTRL_13	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	CLRINTMSK	SETINTMSK
		-	-	-	-	-	-	CLRSUS	SETSUS
		-	CLRTC	CLREND	CLRRQ	SWRST	STG	CLREN	SETEN
	CHCFG_13	DMS	REN	RSW	RSEL	SBE	-	-	DEM
		-	TM	DAD	SAD	DDS[3]	DDS[2]	DDS[1]	DDS[0]
		SDS[3]	SDS[2]	SDS[1]	SDS[0]	-	AM[2]	AM[1]	AM[0]
		-	LVL	HIEN	LOEN	REQD	SEL[2]	SEL[1]	SEL[0]
	CHITVL_13	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		ITVL[15]	ITVL[14]	ITVL[13]	ITVL[12]	ITVL[11]	ITVL[10]	ITVL[9]	ITVL[8]
		ITVL[7]	ITVL[6]	ITVL[5]	ITVL[4]	ITVL[3]	ITVL[2]	ITVL[1]	ITVL[0]
	CHEXT_13	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		DCA[3]	DCA[2]	DCA[1]	DCA[0]	-	-	-	-
		SCA[3]	SCA[2]	SCA[1]	SCA[0]	-	-	-	-
	NXLA_13	NXLA[31]	NXLA[30]	NXLA[29]	NXLA[28]	NXLA[27]	NXLA[26]	NXLA[25]	NXLA[24]
		NXLA[23]	NXLA[22]	NXLA[21]	NXLA[20]	NXLA[19]	NXLA[18]	NXLA[17]	NXLA[16]
		NXLA[15]	NXLA[14]	NXLA[13]	NXLA[12]	NXLA[11]	NXLA[10]	NXLA[9]	NXLA[8]
		NXLA[7]	NXLA[6]	NXLA[5]	NXLA[4]	NXLA[3]	NXLA[2]	NXLA[1]	NXLA[0]
	CRLA_13	CRLA[31]	CRLA[30]	CRLA[29]	CRLA[28]	CRLA[27]	CRLA[26]	CRLA[25]	CRLA[24]
		CRLA[23]	CRLA[22]	CRLA[21]	CRLA[20]	CRLA[19]	CRLA[18]	CRLA[17]	CRLA[16]
		CRLA[15]	CRLA[14]	CRLA[13]	CRLA[12]	CRLA[11]	CRLA[10]	CRLA[9]	CRLA[8]
		CRLA[7]	CRLA[6]	CRLA[5]	CRLA[4]	CRLA[3]	CRLA[2]	CRLA[1]	CRLA[0]
	NOSA_14	SA[31]	SA[30]	SA[29]	SA[28]	SA[27]	SA[26]	SA[25]	SA[24]
		SA[23]	SA[22]	SA[21]	SA[20]	SA[19]	SA[18]	SA[17]	SA[16]
		SA[15]	SA[14]	SA[13]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	NODA_14	DA[31]	DA[30]	DA[29]	DA[28]	DA[27]	DA[26]	DA[25]	DA[24]
		DA[23]	DA[22]	DA[21]	DA[20]	DA[19]	DA[18]	DA[17]	DA[16]
		DA[15]	DA[14]	DA[13]	DA[12]	DA[11]	DA[10]	DA[9]	DA[8]
		DA[7]	DA[6]	DA[5]	DA[4]	DA[3]	DA[2]	DA[1]	DA[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Direct memory access controller	NOTB_14	TB[31]	TB[30]	TB[29]	TB[28]	TB[27]	TB[26]	TB[25]	TB[24]
		TB[23]	TB[22]	TB[21]	TB[20]	TB[19]	TB[18]	TB[17]	TB[16]
		TB[15]	TB[14]	TB[13]	TB[12]	TB[11]	TB[10]	TB[9]	TB[8]
		TB[7]	TB[6]	TB[5]	TB[4]	TB[3]	TB[2]	TB[1]	TB[0]
	N1SA_14	SA[31]	SA[30]	SA[29]	SA[28]	SA[27]	SA[26]	SA[25]	SA[24]
		SA[23]	SA[22]	SA[21]	SA[20]	SA[19]	SA[18]	SA[17]	SA[16]
		SA[15]	SA[14]	SA[13]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	N1DA_14	DA[31]	DA[30]	DA[29]	DA[28]	DA[27]	DA[26]	DA[25]	DA[24]
		DA[23]	DA[22]	DA[21]	DA[20]	DA[19]	DA[18]	DA[17]	DA[16]
		DA[15]	DA[14]	DA[13]	DA[12]	DA[11]	DA[10]	DA[9]	DA[8]
		DA[7]	DA[6]	DA[5]	DA[4]	DA[3]	DA[2]	DA[1]	DA[0]
	N1TB_14	TB[31]	TB[30]	TB[29]	TB[28]	TB[27]	TB[26]	TB[25]	TB[24]
		TB[23]	TB[22]	TB[21]	TB[20]	TB[19]	TB[18]	TB[17]	TB[16]
		TB[15]	TB[14]	TB[13]	TB[12]	TB[11]	TB[10]	TB[9]	TB[8]
		TB[7]	TB[6]	TB[5]	TB[4]	TB[3]	TB[2]	TB[1]	TB[0]
	CRSA_14	CRSA[31]	CRSA[30]	CRSA[29]	CRSA[28]	CRSA[27]	CRSA[26]	CRSA[25]	CRSA[24]
		CRSA[23]	CRSA[22]	CRSA[21]	CRSA[20]	CRSA[19]	CRSA[18]	CRSA[17]	CRSA[16]
		CRSA[15]	CRSA[14]	CRSA[13]	CRSA[12]	CRSA[11]	CRSA[10]	CRSA[9]	CRSA[8]
		CRSA[7]	CRSA[6]	CRSA[5]	CRSA[4]	CRSA[3]	CRSA[2]	CRSA[1]	CRSA[0]
	CRDA_14	CRDA[31]	CRDA[30]	CRDA[29]	CRDA[28]	CRDA[27]	CRDA[26]	CRDA[25]	CRDA[24]
		CRDA[23]	CRDA[22]	CRDA[21]	CRDA[20]	CRDA[19]	CRDA[18]	CRDA[17]	CRDA[16]
		CRDA[15]	CRDA[14]	CRDA[13]	CRDA[12]	CRDA[11]	CRDA[10]	CRDA[9]	CRDA[8]
		CRDA[7]	CRDA[6]	CRDA[5]	CRDA[4]	CRDA[3]	CRDA[2]	CRDA[1]	CRDA[0]
	CRTB_14	CRTB[31]	CRTB[30]	CRTB[29]	CRTB[28]	CRTB[27]	CRTB[26]	CRTB[25]	CRTB[24]
		CRTB[23]	CRTB[22]	CRTB[21]	CRTB[20]	CRTB[19]	CRTB[18]	CRTB[17]	CRTB[16]
		CRTB[15]	CRTB[14]	CRTB[13]	CRTB[12]	CRTB[11]	CRTB[10]	CRTB[9]	CRTB[8]
		CRTB[7]	CRTB[6]	CRTB[5]	CRTB[4]	CRTB[3]	CRTB[2]	CRTB[1]	CRTB[0]
	CHSTAT_14	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	INTMSK
		-	-	-	-	MODE	DER	DW	DL
		SR	TC	END	ER	SUS	TACT	RQST	EN
	CHCTRL_14	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	CLRINTMSK	SETINTMSK
		-	-	-	-	-	-	CLRSUS	SETSUS
		-	CLRTC	CLREND	CLRRQ	SWRST	STG	CLREN	SETEN
	CHCFG_14	DMS	REN	RSW	RSEL	SBE	-	-	DEM
		-	TM	DAD	SAD	DDS[3]	DDS[2]	DDS[1]	DDS[0]
		SDS[3]	SDS[2]	SDS[1]	SDS[0]	-	AM[2]	AM[1]	AM[0]
		-	LVL	HIEN	LOEN	REQD	SEL[2]	SEL[1]	SEL[0]
	CHITVL_14	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		ITVL[15]	ITVL[14]	ITVL[13]	ITVL[12]	ITVL[11]	ITVL[10]	ITVL[9]	ITVL[8]
		ITVL[7]	ITVL[6]	ITVL[5]	ITVL[4]	ITVL[3]	ITVL[2]	ITVL[1]	ITVL[0]
	CHEXT_14	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		DCA[3]	DCA[2]	DCA[1]	DCA[0]	-	-	-	-
		SCA[3]	SCA[2]	SCA[1]	SCA[0]	-	-	-	-
	NXLA_14	NXLA[31]	NXLA[30]	NXLA[29]	NXLA[28]	NXLA[27]	NXLA[26]	NXLA[25]	NXLA[24]
		NXLA[23]	NXLA[22]	NXLA[21]	NXLA[20]	NXLA[19]	NXLA[18]	NXLA[17]	NXLA[16]
		NXLA[15]	NXLA[14]	NXLA[13]	NXLA[12]	NXLA[11]	NXLA[10]	NXLA[9]	NXLA[8]
		NXLA[7]	NXLA[6]	NXLA[5]	NXLA[4]	NXLA[3]	NXLA[2]	NXLA[1]	NXLA[0]
	CRLA_14	CRLA[31]	CRLA[30]	CRLA[29]	CRLA[28]	CRLA[27]	CRLA[26]	CRLA[25]	CRLA[24]
		CRLA[23]	CRLA[22]	CRLA[21]	CRLA[20]	CRLA[19]	CRLA[18]	CRLA[17]	CRLA[16]
		CRLA[15]	CRLA[14]	CRLA[13]	CRLA[12]	CRLA[11]	CRLA[10]	CRLA[9]	CRLA[8]
		CRLA[7]	CRLA[6]	CRLA[5]	CRLA[4]	CRLA[3]	CRLA[2]	CRLA[1]	CRLA[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Direct memory access controller	NOSA_15	SA[31]	SA[30]	SA[29]	SA[28]	SA[27]	SA[26]	SA[25]	SA[24]
		SA[23]	SA[22]	SA[21]	SA[20]	SA[19]	SA[18]	SA[17]	SA[16]
		SA[15]	SA[14]	SA[13]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	NODA_15	DA[31]	DA[30]	DA[29]	DA[28]	DA[27]	DA[26]	DA[25]	DA[24]
		DA[23]	DA[22]	DA[21]	DA[20]	DA[19]	DA[18]	DA[17]	DA[16]
		DA[15]	DA[14]	DA[13]	DA[12]	DA[11]	DA[10]	DA[9]	DA[8]
		DA[7]	DA[6]	DA[5]	DA[4]	DA[3]	DA[2]	DA[1]	DA[0]
	NOTB_15	TB[31]	TB[30]	TB[29]	TB[28]	TB[27]	TB[26]	TB[25]	TB[24]
		TB[23]	TB[22]	TB[21]	TB[20]	TB[19]	TB[18]	TB[17]	TB[16]
		TB[15]	TB[14]	TB[13]	TB[12]	TB[11]	TB[10]	TB[9]	TB[8]
		TB[7]	TB[6]	TB[5]	TB[4]	TB[3]	TB[2]	TB[1]	TB[0]
	NISA_15	SA[31]	SA[30]	SA[29]	SA[28]	SA[27]	SA[26]	SA[25]	SA[24]
		SA[23]	SA[22]	SA[21]	SA[20]	SA[19]	SA[18]	SA[17]	SA[16]
		SA[15]	SA[14]	SA[13]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	NIDA_15	DA[31]	DA[30]	DA[29]	DA[28]	DA[27]	DA[26]	DA[25]	DA[24]
		DA[23]	DA[22]	DA[21]	DA[20]	DA[19]	DA[18]	DA[17]	DA[16]
		DA[15]	DA[14]	DA[13]	DA[12]	DA[11]	DA[10]	DA[9]	DA[8]
		DA[7]	DA[6]	DA[5]	DA[4]	DA[3]	DA[2]	DA[1]	DA[0]
	NITB_15	TB[31]	TB[30]	TB[29]	TB[28]	TB[27]	TB[26]	TB[25]	TB[24]
		TB[23]	TB[22]	TB[21]	TB[20]	TB[19]	TB[18]	TB[17]	TB[16]
		TB[15]	TB[14]	TB[13]	TB[12]	TB[11]	TB[10]	TB[9]	TB[8]
		TB[7]	TB[6]	TB[5]	TB[4]	TB[3]	TB[2]	TB[1]	TB[0]
	CRSA_15	CRSA[31]	CRSA[30]	CRSA[29]	CRSA[28]	CRSA[27]	CRSA[26]	CRSA[25]	CRSA[24]
		CRSA[23]	CRSA[22]	CRSA[21]	CRSA[20]	CRSA[19]	CRSA[18]	CRSA[17]	CRSA[16]
		CRSA[15]	CRSA[14]	CRSA[13]	CRSA[12]	CRSA[11]	CRSA[10]	CRSA[9]	CRSA[8]
		CRSA[7]	CRSA[6]	CRSA[5]	CRSA[4]	CRSA[3]	CRSA[2]	CRSA[1]	CRSA[0]
	CRDA_15	CRDA[31]	CRDA[30]	CRDA[29]	CRDA[28]	CRDA[27]	CRDA[26]	CRDA[25]	CRDA[24]
		CRDA[23]	CRDA[22]	CRDA[21]	CRDA[20]	CRDA[19]	CRDA[18]	CRDA[17]	CRDA[16]
		CRDA[15]	CRDA[14]	CRDA[13]	CRDA[12]	CRDA[11]	CRDA[10]	CRDA[9]	CRDA[8]
		CRDA[7]	CRDA[6]	CRDA[5]	CRDA[4]	CRDA[3]	CRDA[2]	CRDA[1]	CRDA[0]
	CRTB_15	CRTB[31]	CRTB[30]	CRTB[29]	CRTB[28]	CRTB[27]	CRTB[26]	CRTB[25]	CRTB[24]
		CRTB[23]	CRTB[22]	CRTB[21]	CRTB[20]	CRTB[19]	CRTB[18]	CRTB[17]	CRTB[16]
		CRTB[15]	CRTB[14]	CRTB[13]	CRTB[12]	CRTB[11]	CRTB[10]	CRTB[9]	CRTB[8]
		CRTB[7]	CRTB[6]	CRTB[5]	CRTB[4]	CRTB[3]	CRTB[2]	CRTB[1]	CRTB[0]
	CHSTAT_15	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	INTMSK
		-	-	-	-	MODE	DER	DW	DL
		SR	TC	END	ER	SUS	TACT	RQST	EN
	CHCTRL_15	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	CLRINTMSK	SETINTMSK
		-	-	-	-	-	-	CLRSUS	SETSUS
		-	CLRTC	CLREND	CLRRQ	SWRST	STG	CLREN	SETEN
	CHCFG_15	DMS	REN	RSW	RSEL	SBE	-	-	DEM
		-	TM	DAD	SAD	DDS[3]	DDS[2]	DDS[1]	DDS[0]
		SDS[3]	SDS[2]	SDS[1]	SDS[0]	-	AM[2]	AM[1]	AM[0]
		-	LVL	HIEN	LOEN	REQD	SEL[2]	SEL[1]	SEL[0]
	CHITVL_15	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		ITVL[15]	ITVL[14]	ITVL[13]	ITVL[12]	ITVL[11]	ITVL[10]	ITVL[9]	ITVL[8]
		ITVL[7]	ITVL[6]	ITVL[5]	ITVL[4]	ITVL[3]	ITVL[2]	ITVL[1]	ITVL[0]
	CHEXT_15	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		DCA[3]	DCA[2]	DCA[1]	DCA[0]	-	-	-	-
		SCA[3]	SCA[2]	SCA[1]	SCA[0]	-	-	-	-



**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Direct memory access controller	NXLA_15	NXLA[31]	NXLA[30]	NXLA[29]	NXLA[28]	NXLA[27]	NXLA[26]	NXLA[25]	NXLA[24]
		NXLA[23]	NXLA[22]	NXLA[21]	NXLA[20]	NXLA[19]	NXLA[18]	NXLA[17]	NXLA[16]
		NXLA[15]	NXLA[14]	NXLA[13]	NXLA[12]	NXLA[11]	NXLA[10]	NXLA[9]	NXLA[8]
		NXLA[7]	NXLA[6]	NXLA[5]	NXLA[4]	NXLA[3]	NXLA[2]	NXLA[1]	NXLA[0]
	CRLA_15	CRLA[31]	CRLA[30]	CRLA[29]	CRLA[28]	CRLA[27]	CRLA[26]	CRLA[25]	CRLA[24]
		CRLA[23]	CRLA[22]	CRLA[21]	CRLA[20]	CRLA[19]	CRLA[18]	CRLA[17]	CRLA[16]
		CRLA[15]	CRLA[14]	CRLA[13]	CRLA[12]	CRLA[11]	CRLA[10]	CRLA[9]	CRLA[8]
		CRLA[7]	CRLA[6]	CRLA[5]	CRLA[4]	CRLA[3]	CRLA[2]	CRLA[1]	CRLA[0]
	DCTRL_8_15	LWCA[3]	LWCA[2]	LWCA[1]	LWCA[0]	-	-	-	-
		LDCA[3]	LDCA[2]	LDCA[1]	LDCA[0]	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	LVINT	PR
	DSTAT_EN_8_15	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		EN15	EN14	EN13	EN12	EN11	EN10	EN9	EN8
	DSTAT_ER_8_15	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		ER15	ER14	ER13	ER12	ER11	ER10	ER9	ER8
	DSTAT_END_8_15	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		END15	END14	END13	END12	END11	END10	END9	END8
	DSTAT_TC_8_15	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		TC15	TC14	TC13	TC12	TC11	TC10	TC9	TC8
	DSTAT_SUS_8_15	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		SUS15	SUS14	SUS13	SUS12	SUS11	SUS10	SUS9	SUS8
	DMARS0	-	-	-	-	-	-	-	-
		CH1_MID[5]	CH1_MID[4]	CH1_MID[3]	CH1_MID[2]	CH1_MID[1]	CH1_MID[0]	CH1_RID[1]	CH1_RID[0]
		-	-	-	-	-	-	-	CH0_MID[6]
		CH0_MID[5]	CH0_MID[4]	CH0_MID[3]	CH0_MID[2]	CH0_MID[1]	CH0_MID[0]	CH0_RID[1]	CH0_RID[0]
	DMARS1	-	-	-	-	-	-	-	-
		CH3_MID[5]	CH3_MID[4]	CH3_MID[3]	CH3_MID[2]	CH3_MID[1]	CH3_MID[0]	CH3_RID[1]	CH3_RID[0]
		-	-	-	-	-	-	-	CH2_MID[6]
		CH2_MID[5]	CH2_MID[4]	CH2_MID[3]	CH2_MID[2]	CH2_MID[1]	CH2_MID[0]	CH2_RID[1]	CH2_RID[0]
	DMARS2	-	-	-	-	-	-	-	-
		CH5_MID[5]	CH5_MID[4]	CH5_MID[3]	CH5_MID[2]	CH5_MID[1]	CH5_MID[0]	CH5_RID[1]	CH5_RID[0]
		-	-	-	-	-	-	-	CH4_MID[6]
		CH4_MID[5]	CH4_MID[4]	CH4_MID[3]	CH4_MID[2]	CH4_MID[1]	CH4_MID[0]	CH4_RID[1]	CH4_RID[0]
	DMARS3	-	-	-	-	-	-	-	-
		CH7_MID[5]	CH7_MID[4]	CH7_MID[3]	CH7_MID[2]	CH7_MID[1]	CH7_MID[0]	CH7_RID[1]	CH7_RID[0]
		-	-	-	-	-	-	-	CH6_MID[6]
		CH6_MID[5]	CH6_MID[4]	CH6_MID[3]	CH6_MID[2]	CH6_MID[1]	CH6_MID[0]	CH6_RID[1]	CH6_RID[0]
	DMARS4	-	-	-	-	-	-	-	-
		CH9_MID[5]	CH9_MID[4]	CH9_MID[3]	CH9_MID[2]	CH9_MID[1]	CH9_MID[0]	CH9_RID[1]	CH9_RID[0]
		-	-	-	-	-	-	-	CH8_MID[6]
		CH8_MID[5]	CH8_MID[4]	CH8_MID[3]	CH8_MID[2]	CH8_MID[1]	CH8_MID[0]	CH8_RID[1]	CH8_RID[0]
	DMARS5	-	-	-	-	-	-	-	-
		CH11_MID[5]	CH11_MID[4]	CH11_MID[3]	CH11_MID[2]	CH11_MID[1]	CH11_MID[0]	CH11_RID[1]	CH11_RID[0]
		-	-	-	-	-	-	-	CH10_MID[6]
		CH10_MID[5]	CH10_MID[4]	CH10_MID[3]	CH10_MID[2]	CH10_MID[1]	CH10_MID[0]	CH10_RID[1]	CH10_RID[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Direct memory access controller	DMARS6	-	-	-	-	-	-	-	CH13_MID[6]
		CH13_MID[5]	CH13_MID[4]	CH13_MID[3]	CH13_MID[2]	CH13_MID[1]	CH13_MID[0]	CH13_RID[1]	CH13_RID[0]
		-	-	-	-	-	-	-	CH12_MID[6]
	DMARS7	CH12_MID[5]	CH12_MID[4]	CH12_MID[3]	CH12_MID[2]	CH12_MID[1]	CH12_MID[0]	CH12_RID[1]	CH12_RID[0]
		-	-	-	-	-	-	-	CH15_MID[6]
		CH15_MID[5]	CH15_MID[4]	CH15_MID[3]	CH15_MID[2]	CH15_MID[1]	CH15_MID[0]	CH15_RID[1]	CH15_RID[0]
		-	-	-	-	-	-	-	CH14_MID[6]
Multi-function timer pulse unit 2	TCR_0	CCLR[2]	CCLR[1]	CCLR[0]	CKEG[1]	CKEG[0]	TPSC[2]	TPSC[1]	TPSC[0]
	TMDR_0	-	BFE	BFB	BFA	MD[3]	MD[2]	MD[1]	MD[0]
	TIORH_0	IOB[3]	IOB[2]	IOB[1]	IOB[0]	IOA[3]	IOA[2]	IOA[1]	IOA[0]
	TIORL_0	IOD[3]	IOD[2]	IOD[1]	IOD[0]	IOC[3]	IOC[2]	IOC[1]	IOC[0]
	TIER_0	TTGE	-	-	TCIEV	TGIED	TGIEC	TGIEB	TGIEA
	TSR_0	-	-	-	TCFV	TGFD	TGFC	TGFB	TGFA
	TCNT_0								
	TGRA_0								
	TGRB_0								
	TGRC_0								
	TGRD_0								
	TGRE_0								
	TGRF_0								
	TIER2_0	TTGE2	-	-	-	-	-	TGIEF	TGIEE
	TSR2_0	-	-	-	-	-	-	TGFF	TGFE
	TBTM_0	-	-	-	-	-	TTSE	TTSB	T TSA
	TCR_1	-	CCLR[1]	CCLR[0]	CKEG[1]	CKEG[0]	TPSC[2]	TPSC[1]	TPSC[0]
	TMDR_1	-	-	-	-	MD[3]	MD[2]	MD[1]	MD[0]
	TIOR_1	IOB[3]	IOB[2]	IOB[1]	IOB[0]	IOA[3]	IOA[2]	IOA[1]	IOA[0]
	TIER_1	TTGE	-	TCIEU	TCIEV	-	-	TGIEB	TGIEA
	TSR_1	TCFD	-	TCFU	TCFV	-	-	TGFB	TGFA
	TCNT_1								
	TGRA_1								
	TGRB_1								
	TICCR	-	-	-	-	I2BE	I2AE	I1BE	I1AE
	TCR_2	-	CCLR[1]	CCLR[0]	CKEG[1]	CKEG[0]	TPSC[2]	TPSC[1]	TPSC[0]
	TMDR_2	-	-	-	-	MD[3]	MD[2]	MD[1]	MD[0]
TIOR_2	IOB[3]	IOB[2]	IOB[1]	IOB[0]	IOA[3]	IOA[2]	IOA[1]	IOA[0]	
TIER_2	TTGE	-	TCIEU	TCIEV	-	-	TGIEB	TGIEA	
TSR_2	TCFD	-	TCFU	TCFV	-	-	TGFB	TGFA	
TCNT_2									
TGRA_2									
TGRB_2									
TCR_3	CCLR[2]	CCLR[1]	CCLR[0]	CKEG[1]	CKEG[0]	TPSC[2]	TPSC[1]	TPSC[0]	
TMDR_3	-	-	BFB	BFA	MD[3]	MD[2]	MD[1]	MD[0]	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Multi-function timer pulse unit 2	TIORH_3	IOB[3]	IOB[2]	IOB[1]	IOB[0]	IOA[3]	IOA[2]	IOA[1]	IOA[0]	
	TIORL_3	IOD[3]	IOD[2]	IOD[1]	IOD[0]	IOC[3]	IOC[2]	IOC[1]	IOC[0]	
	TIER_3	TTGE	-	-	TCIEV	TGIED	TGIEC	TGIEB	TGIEA	
	TSR_3	TCFD	-	-	TCFV	TGFD	TGFC	TGFB	TGFA	
	TCNT_3									
	TGRA_3									
	TGRB_3									
	TGRC_3									
	TGRD_3									
	TBTM_3	-	-	-	-	-	-	-	TTSB	TTSA
	TCR_4	CCLR[2]	CCLR[1]	CCLR[0]	CKEG[1]	CKEG[0]	TPSC[2]	TPSC[1]	TPSC[0]	
	TMDR_4	-	-	BFB	BFA	MD[3]	MD[2]	MD[1]	MD[0]	
	TIORH_4	IOB[3]	IOB[2]	IOB[1]	IOB[0]	IOA[3]	IOA[2]	IOA[1]	IOA[0]	
	TIORL_4	IOD[3]	IOD[2]	IOD[1]	IOD[0]	IOC[3]	IOC[2]	IOC[1]	IOC[0]	
	TIER_4	TTGE	TTGE2	-	TCIEV	TGIED	TGIEC	TGIEB	TGIEA	
	TSR_4	TCFD	-	-	TCFV	TGFD	TGFC	TGFB	TGFA	
	TCNT_4									
	TGRA_4									
	TGRB_4									
	TGRC_4									
	TGRD_4									
	TBTM_4	-	-	-	-	-	-	-	TTSB	TTSA
	TADCR	BF[1] UT4AE	BF[0] DT4AE	- UT4BE	- DT4BE	- ITA3AE	- ITA4VE	- ITB3AE	- ITB4VE	- ITB4VE
	TADCORA_4									
	TADCORB_4									
	TADCOBRA_4									
	TADCOBRB_4									
	TSTR	CST4	CST3	-	-	-	CST2	CST1	CST0	
	TSYR	SYNC4	SYNC3	-	-	-	SYNC2	SYNC1	SYNC0	
	TRWER	-	-	-	-	-	-	-	-	RWE
	TOER	-	-	OE4D	OE4C	OE3D	OE4B	OE4A	OE3B	
	TOCR1	-	PSYE	-	-	TOCL	TOCS	OLSN	OLSP	
	TOCR2	BF[1]	BF[0]	OLS3N	OLS3P	OLS2N	OLS2P	OLS1N	OLS1P	
	TGCR	-	BDC	N	P	FB	WF	VF	UF	
	TCDR									
	TDDR									
	TCNTS									
	TCBR									

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Multi-function timer pulse unit 2	TITCR	T3AEN	3ACOR[2]	3ACOR[1]	3ACOR[0]	T4VEN	4VCOR[2]	4VCOR[1]	4VCOR[0]	
	TITCNT	-	3ACNT[2]	3ACNT[1]	3ACNT[0]	-	4VCNT[2]	4VCNT[1]	4VCNT[0]	
	TBTER	-	-	-	-	-	-	BTE[1]	BTE[0]	
	TDER	-	-	-	-	-	-	-	TDER	
	TWCR	CCE	-	-	-	-	-	-	WRE	
	TOLBR	-	-	OLS3N	OLS3P	OLS2N	OLS2P	OLS1N	OLS1P	
OS timer	OSTM0CMP	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
	OSTM0CNT	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
	OSTM0TE	-	-	-	-	-	-	-	OSTM0TE	
	OSTM0TS	-	-	-	-	-	-	-	OSTM0TS	
	OSTM0TT	-	-	-	-	-	-	-	OSTM0TT	
	OSTM0CTL	-	-	-	-	-	-	OSTM0MD1	OSTM0MD0	
	OSTM1CMP	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
	OSTM1CNT	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
OSTM1TE	-	-	-	-	-	-	-	OSTM1TE		
OSTM1TS	-	-	-	-	-	-	-	OSTM1TS		
OSTM1TT	-	-	-	-	-	-	-	OSTM1TT		
OSTM1CTL	-	-	-	-	-	-	OSTM1MD1	OSTM1MD0		
Watchdog timer	WTCNT	-	-	-	-	-	-	-	-	
	WTCNR	IOVF	WT/IT	TME	-	-	CKS[2]	CKS[1]	CKS[0]	
	WRCSR	WVOF	RSTE	-	-	-	-	-	-	
Realtime clock	R64CNT	-	1Hz	2Hz	4Hz	8Hz	16Hz	32Hz	64Hz	
	RSECCNT	-	10 seconds[2]	10 seconds[1]	10 seconds[0]	1 second[3]	1 second[2]	1 second[1]	1 second[0]	
	RMINCNT	-	10 minutes[2]	10 minutes[1]	10 minutes[0]	1 minute[3]	1 minute[2]	1 minute[1]	1 minute[0]	
	RHRCNT	-	-	10 hours[1]	10 hours[0]	1 hour[3]	1 hour[2]	1 hour[1]	1 hour[0]	
	RWKCNT	-	-	-	-	-	Day[2]	Day[1]	Day[0]	
	RDAYCNT	-	-	10 days[1]	10 days[0]	1 day[3]	1 day[2]	1 day[1]	1 day[0]	
	RMONCNT	-	-	-	10 months	1 month[3]	1 month[2]	1 month[1]	1 month[0]	
	RYRCNT	1000 years[3]	1000 years[2]	1000 years[1]	1000 years[0]	100 years[3]	100 years[2]	100 years[1]	100 years[0]	
		10 years[3]	10 years[2]	10 years[1]	10 years[0]	1 year[3]	1 year[2]	1 year[1]	1 year[0]	
	RSECAR	ENB	10 seconds[2]	10 seconds[1]	10 seconds[0]	1 second[3]	1 second[2]	1 second[1]	1 second[0]	
	RMINAR	ENB	10 minutes[2]	10 minutes[1]	10 minutes[0]	1 minute[3]	1 minute[2]	1 minute[1]	1 minute[0]	
	RHRAR	ENB	-	10 hours[1]	10 hours[0]	1 hour[3]	1 hour[2]	1 hour[1]	1 hour[0]	
	RWKAR	ENB	-	-	-	-	Day[2]	Day[1]	Day[0]	
	RDAYAR	ENB	-	10 days[1]	10 days[0]	1 day[3]	1 day[2]	1 day[1]	1 day[0]	
	RMONAR	ENB	-	-	10 months	1 month[3]	1 month[2]	1 month[1]	1 month[0]	
	RYRAR	1000 years[3]	1000 years[2]	1000 years[1]	1000 years[0]	100 years[3]	100 years[2]	100 years[1]	100 years[0]	
		10 years[3]	10 years[2]	10 years[1]	10 years[0]	1 year[3]	1 year[2]	1 year[1]	1 year[0]	
	RCR1	CF	-	-	CIE	AIE	-	-	AF	
	RCR2	PEF	PES[2]	PES[1]	PES[0]	RTCEN	ADJ	RESET	START	
	RCR3	ENB	-	-	-	-	-	-	-	
	RCR5	-	-	-	-	-	-	RCKSEL[1]	RCKSEL[0]	
	RFRH	SEL64	-	-	-	-	-	-	-	
		-	-	-	-	-	RFC[18]	RFC[17]	RFC[16]	
	RFRL	RFC[15]	RFC[14]	RFC[13]	RFC[12]	RFC[11]	RFC[10]	RFC[9]	RFC[8]	
		RFC[7]	RFC[6]	RFC[5]	RFC[4]	RFC[3]	RFC[2]	RFC[1]	RFC[0]	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Serial communication interface with FIFO	SCSMR_0	-	-	-	-	-	-	-	-
		C/A	CHR	PE	O/E	STOP	-	CKS[1]	CKS[0]
	SCBRR_0								
	SCSCR_0	-	-	-	-	-	-	-	-
		TIE	RIE	TE	RE	REIE	-	CKE[1]	CKE[0]
	SCFTDR_0								
	SCFSR_0	PER[3]	PER[2]	PER[1]	PER[0]	FER[3]	FER[2]	FER[1]	FER[0]
		ER	TEND	TDFE	BRK	FER	PER	RDF	DR
	SCFRDR_0								
	SCFCR_0	-	-	-	-	-	RSTRG[2]	RSTRG[1]	RSTRG[0]
		RTRG[1]	RTRG[0]	TTRG[1]	TTRG[0]	MCE	TFRST	RFRST	LOOP
	SCFDR_0	-	-	-	T[4]	T[3]	T[2]	T[1]	T[0]
		-	-	-	R[4]	R[3]	R[2]	R[1]	R[0]
	SCSPTR_0	-	-	-	-	-	-	-	-
		RTSIO	RTSDT	CTSIO	CTSDT	SCKIO	SCKDT	SPB2IO	SPB2DT
	SCLSR_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	ORER
	SCEMR_0	-	-	-	-	-	-	-	-
		BGDM	-	-	-	-	-	-	ABCS
	SCSMR_1	-	-	-	-	-	-	-	-
		C/A	CHR	PE	O/E	STOP	-	CKS[1]	CKS[0]
	SCBRR_1								
	SCSCR_1	-	-	-	-	-	-	-	-
		TIE	RIE	TE	RE	REIE	-	CKE[1]	CKE[0]
	SCFTDR_1								
	SCFSR_1	PER[3]	PER[2]	PER[1]	PER[0]	FER[3]	FER[2]	FER[1]	FER[0]
		ER	TEND	TDFE	BRK	FER	PER	RDF	DR
	SCFRDR_1								
	SCFCR_1	-	-	-	-	-	RSTRG[2]	RSTRG[1]	RSTRG[0]
		RTRG[1]	RTRG[0]	TTRG[1]	TTRG[0]	MCE	TFRST	RFRST	LOOP
	SCFDR_1	-	-	-	T[4]	T[3]	T[2]	T[1]	T[0]
		-	-	-	R[4]	R[3]	R[2]	R[1]	R[0]
	SCSPTR_1	-	-	-	-	-	-	-	-
		RTSIO	RTSDT	CTSIO	CTSDT	SCKIO	SCKDT	SPB2IO	SPB2DT
	SCLSR_1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	ORER
	SCEMR_1	-	-	-	-	-	-	-	-
		BGDM	-	-	-	-	-	-	ABCS
	SCSMR_2	-	-	-	-	-	-	-	-
		C/A	CHR	PE	O/E	STOP	-	CKS[1]	CKS[0]
	SCBRR_2								
	SCSCR_2	-	-	-	-	-	-	-	-
		TIE	RIE	TE	RE	REIE	-	CKE[1]	CKE[0]
	SCFTDR_2								
	SCFSR_2	PER[3]	PER[2]	PER[1]	PER[0]	FER[3]	FER[2]	FER[1]	FER[0]
		ER	TEND	TDFE	BRK	FER	PER	RDF	DR
	SCFRDR_2								
	SCFCR_2	-	-	-	-	-	RSTRG[2]	RSTRG[1]	RSTRG[0]
		RTRG[1]	RTRG[0]	TTRG[1]	TTRG[0]	MCE	TFRST	RFRST	LOOP
	SCFDR_2	-	-	-	T[4]	T[3]	T[2]	T[1]	T[0]
		-	-	-	R[4]	R[3]	R[2]	R[1]	R[0]
	SCSPTR_2	-	-	-	-	-	-	-	-
		RTSIO	RTSDT	CTSIO	CTSDT	SCKIO	SCKDT	SPB2IO	SPB2DT
	SCLSR_2	-	-	-	-	-	-	-	-
	-	-	-	-	-	-	-	ORER	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Serial communication interface with FIFO	SCEMR_2	-	-	-	-	-	-	-	-
		BGDM	-	-	-	-	-	-	ABCS
	SCSMR_3	-	-	-	-	-	-	-	-
		C/A	CHR	PE	O/E	STOP	-	CKS[1]	CKS[0]
	SCBRR_3	-	-	-	-	-	-	-	-
	SCSCR_3	-	-	-	-	-	-	-	-
		TIE	RIE	TE	RE	REIE	-	CKE[1]	CKE[0]
	SCFTDR_3	-	-	-	-	-	-	-	-
	SCFSR_3	PER[3]	PER[2]	PER[1]	PER[0]	FER[3]	FER[2]	FER[1]	FER[0]
		ER	TEND	TDFE	BRK	FER	PER	RDF	DR
	SCFRDR_3	-	-	-	-	-	-	-	-
	SCFCR_3	-	-	-	-	-	RSTRG[2]	RSTRG[1]	RSTRG[0]
		RTRG[1]	RTRG[0]	TTRG[1]	TTRG[0]	MCE	TFRST	RFRST	LOOP
	SCFDR_3	-	-	-	T[4]	T[3]	T[2]	T[1]	T[0]
		-	-	-	R[4]	R[3]	R[2]	R[1]	R[0]
	SCSPTR_3	-	-	-	-	-	-	-	-
		RTSIO	RTSDT	CTSIO	CTSDT	SCKIO	SCKDT	SPB2IO	SPB2DT
	SCLSR_3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	ORER
	SCEMR_3	-	-	-	-	-	-	-	-
		BGDM	-	-	-	-	-	-	ABCS
	SCSMR_4	-	-	-	-	-	-	-	-
		C/A	CHR	PE	O/E	STOP	-	CKS[1]	CKS[0]
	SCBRR_4	-	-	-	-	-	-	-	-
	SCSCR_4	-	-	-	-	-	-	-	-
		TIE	RIE	TE	RE	REIE	-	CKE[1]	CKE[0]
	SCFTDR_4	-	-	-	-	-	-	-	-
	SCFSR_4	PER[3]	PER[2]	PER[1]	PER[0]	FER[3]	FER[2]	FER[1]	FER[0]
		ER	TEND	TDFE	BRK	FER	PER	RDF	DR
	SCFRDR_4	-	-	-	-	-	-	-	-
	SCFCR_4	-	-	-	-	-	RSTRG[2]	RSTRG[1]	RSTRG[0]
		RTRG[1]	RTRG[0]	TTRG[1]	TTRG[0]	MCE	TFRST	RFRST	LOOP
	SCFDR_4	-	-	-	T[4]	T[3]	T[2]	T[1]	T[0]
		-	-	-	R[4]	R[3]	R[2]	R[1]	R[0]
	SCSPTR_4	-	-	-	-	-	-	-	-
		RTSIO	RTSDT	CTSIO	CTSDT	SCKIO	SCKDT	SPB2IO	SPB2DT
	SCLSR_4	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	ORER
	SCEMR_4	-	-	-	-	-	-	-	-
		BGDM	-	-	-	-	-	-	ABCS
	SCSMR_5	-	-	-	-	-	-	-	-
		C/A	CHR	PE	O/E	STOP	-	CKS[1]	CKS[0]
	SCBRR_5	-	-	-	-	-	-	-	-
	SCSCR_5	-	-	-	-	-	-	-	-
		TIE	RIE	TE	RE	REIE	-	CKE[1]	CKE[0]
	SCFTDR_5	-	-	-	-	-	-	-	-
	SCFSR_5	PER[3]	PER[2]	PER[1]	PER[0]	FER[3]	FER[2]	FER[1]	FER[0]
		ER	TEND	TDFE	BRK	FER	PER	RDF	DR
	SCFRDR_5	-	-	-	-	-	-	-	-
	SCFCR_5	-	-	-	-	-	RSTRG[2]	RSTRG[1]	RSTRG[0]
		RTRG[1]	RTRG[0]	TTRG[1]	TTRG[0]	MCE	TFRST	RFRST	LOOP
	SCFDR_5	-	-	-	T[4]	T[3]	T[2]	T[1]	T[0]
		-	-	-	R[4]	R[3]	R[2]	R[1]	R[0]
	SCSPTR_5	-	-	-	-	-	-	-	-
		RTSIO	RTSDT	CTSIO	CTSDT	SCKIO	SCKDT	SPB2IO	SPB2DT
	SCLSR_5	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	ORER

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Serial communication interface with FIFO	SCEMR_5	-	-	-	-	-	-	-	-	
		BGDM	-	-	-	-	-	-	ABCS	
	SCSMR_6	-	-	-	-	-	-	-	-	
		C/A	CHR	PE	O/E	STOP	-	-	CKS[1]	CKS[0]
	SCBRR_6	-	-	-	-	-	-	-	-	
	SCSCR_6	-	-	-	-	-	-	-	-	-
		TIE	RIE	TE	RE	REIE	-	-	CKE[1]	CKE[0]
	SCFTDR_6	-	-	-	-	-	-	-	-	
	SCFSR_6	PER[3]	PER[2]	PER[1]	PER[0]	FER[3]	FER[2]	FER[1]	FER[0]	
		ER	TEND	TDFE	BRK	FER	PER	RDF	DR	
	SCFRDR_6	-	-	-	-	-	-	-	-	
	SCFCR_6	-	-	-	-	-	RSTRG[2]	RSTRG[1]	RSTRG[0]	
		RTRG[1]	RTRG[0]	TTRG[1]	TTRG[0]	MCE	TFRST	RFRST	LOOP	
	SCFDR_6	-	-	-	T[4]	T[3]	T[2]	T[1]	T[0]	
		-	-	-	R[4]	R[3]	R[2]	R[1]	R[0]	
	SCSPTR_6	-	-	-	-	-	-	-	-	
		RTSIO	RTSDT	CTSIO	CTSDT	SCKIO	SCKDT	SPB2IO	SPB2DT	
	SCLSR_6	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	ORER	
	SCEMR_6	-	-	-	-	-	-	-	-	
		BGDM	-	-	-	-	-	-	ABCS	
	SCSMR_7	-	-	-	-	-	-	-	-	
		C/A	CHR	PE	O/E	STOP	-	-	CKS[1]	CKS[0]
	SCBRR_7	-	-	-	-	-	-	-	-	
	SCSCR_7	-	-	-	-	-	-	-	-	
		TIE	RIE	TE	RE	REIE	-	-	CKE[1]	CKE[0]
	SCFTDR_7	-	-	-	-	-	-	-	-	
	SCFSR_7	PER[3]	PER[2]	PER[1]	PER[0]	FER[3]	FER[2]	FER[1]	FER[0]	
		ER	TEND	TDFE	BRK	FER	PER	RDF	DR	
	SCFRDR_7	-	-	-	-	-	-	-	-	
	SCFCR_7	-	-	-	-	-	RSTRG[2]	RSTRG[1]	RSTRG[0]	
		RTRG[1]	RTRG[0]	TTRG[1]	TTRG[0]	MCE	TFRST	RFRST	LOOP	
	SCFDR_7	-	-	-	T[4]	T[3]	T[2]	T[1]	T[0]	
		-	-	-	R[4]	R[3]	R[2]	R[1]	R[0]	
	SCSPTR_7	-	-	-	-	-	-	-	-	
		RTSIO	RTSDT	CTSIO	CTSDT	SCKIO	SCKDT	SPB2IO	SPB2DT	
	SCLSR_7	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	ORER	
	SCEMR_7	-	-	-	-	-	-	-	-	
		BGDM	-	-	-	-	-	-	ABCS	
Serial communications interface	SMR0	CM	CHR	PE	PM	STOP	MP	CKS[1]	CKS[0]	
	SMR0	GM	BLK	PE	PM	BCP[1]	BCP[0]	CKS[1]	CKS[0]	
	BRR0	BRR[7]	BRR[6]	BRR[5]	BRR[4]	BRR[3]	BRR[2]	BRR[1]	BRR[0]	
	SCR0	TIE	RIE	TE	RE	MPIE	TEIE	CKE[1]	CKE[0]	
	TDR0	TDR[7]	TDR[6]	TDR[5]	TDR[4]	TDR[3]	TDR[2]	TDR[1]	TDR[0]	
	SSR0	-	-	ORER	FER	PER	TEND	MPB	MPBT	
	SSR0	-	-	ORER	ERS	PER	TEND	MPB	MPBT	
	RDR0	RDR[7]	RDR[6]	RDR[5]	RDR[4]	RDR[3]	RDR[2]	RDR[1]	RDR[0]	
	SCMR0	BCP2	-	-	-	SDIR	SINV	-	SMIF	
	SEMR0	-	-	NFEN	ABCS	-	-	-	-	
	SNFR0	-	-	-	-	-	NFCS[2]	NFCS[1]	NFCS[0]	
	SECR0	-	-	-	-	-	-	CTSE	-	
	SMR1	CM	CHR	PE	PM	STOP	MP	CKS[1]	CKS[0]	
	SMR1	GM	BLK	PE	PM	BCP[1]	BCP[0]	CKS[1]	CKS[0]	
	BRR1	BRR[7]	BRR[6]	BRR[5]	BRR[4]	BRR[3]	BRR[2]	BRR[1]	BRR[0]	
	SCR1	TIE	RIE	TE	RE	MPIE	TEIE	CKE[1]	CKE[0]	
TDR1	TDR[7]	TDR[6]	TDR[5]	TDR[4]	TDR[3]	TDR[2]	TDR[1]	TDR[0]		

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Serial communications interface	SSR1	-	-	ORER	FER	PER	TEND	MPB	MPBT
	SSR1	-	-	ORER	ERS	PER	TEND	MPB	MPBT
	RDR1	RDR[7]	RDR[6]	RDR[5]	RDR[4]	RDR[3]	RDR[2]	RDR[1]	RDR[0]
	SCMR1	BCP2	-	-	-	SDIR	SINV	-	SMIF
	SEMR1	-	-	NFEN	ABCS	-	-	-	-
	SNFR1	-	-	-	-	-	NFCS[2]	NFCS[1]	NFCS[0]
	SECR1	-	-	-	-	-	-	CTSE	-
IRCR	IRE	IRCKS[2]	IRCKS[1]	IRCKS[0]	IRTXINV	IRRXINV	-	-	
Renesas serial peripheral interface	SPCR_0	SPRIE	SPE	SPTIE	SPEIE	MSTR	MODFEN	-	-
	SSLP_0	-	-	-	-	-	-	-	SSL0P
	SPPCR_0	-	-	MOIFE	MOIFV	-	-	-	SPLP
	SPSR_0	SPRF	TEND	SPTF	-	-	MODF	-	OVRF
	SPDR_0	SPD31	SPD30	SPD29	SPD28	SPD27	SPD26	SPD25	SPD24
		SPD23	SPD22	SPD21	SPD20	SPD19	SPD18	SPD17	SPD16
		SPD15	SPD14	SPD13	SPD12	SPD11	SPD10	SPD9	SPD8
		SPD7	SPD6	SPD5	SPD4	SPD3	SPD2	SPD1	SPD0
	SPSCR_0	-	-	-	-	-	-	SPSLN1	SPSLN0
	SPSSR_0	-	-	-	-	-	-	SPCP1	SPCP0
	SPBR_0	SPR7	SPR6	SPR5	SPR4	SPR3	SPR2	SPR1	SPR0
	SPDCR_0	TXDMY	SPLW1	SPLW0	-	-	-	-	-
	SPCKD_0	-	-	-	-	-	SCKDL2	SCKDL1	SCKDL0
	SSLND_0	-	-	-	-	-	SLNDL2	SLNDL1	SLNDL0
	SPND_0	-	-	-	-	-	SPNDL2	SPNDL1	SPNDL0
	SPCMD0_0	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0
		SSLKP	-	-	-	BRDV1	BRDV0	CPOL	CPHA
	SPCMD1_0	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0
		SSLKP	-	-	-	BRDV1	BRDV0	CPOL	CPHA
	SPCMD2_0	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0
		SSLKP	-	-	-	BRDV1	BRDV0	CPOL	CPHA
	SPCMD3_0	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0
		SSLKP	-	-	-	BRDV1	BRDV0	CPOL	CPHA
	SPBFCR_0	TXRST	RXRST	TXTRG[1]	TXTRG[0]	-	RXTRG[2]	RXTRG[1]	RXTRG[0]
	SPBFDR_0	-	-	-	-	T[3]	T[2]	T[1]	T[0]
		-	-	R[5]	R[4]	R[3]	R[2]	R[1]	R[0]
	SPCR_1	SPRIE	SPE	SPTIE	SPEIE	MSTR	MODFEN	-	-
	SSLP_1	-	-	-	-	-	-	-	SSL0P
	SPPCR_1	-	-	MOIFE	MOIFV	-	-	-	SPLP
	SPSR_1	SPRF	TEND	SPTF	-	-	MODF	-	OVRF
	SPDR_1	SPD31	SPD30	SPD29	SPD28	SPD27	SPD26	SPD25	SPD24
		SPD23	SPD22	SPD21	SPD20	SPD19	SPD18	SPD17	SPD16
		SPD15	SPD14	SPD13	SPD12	SPD11	SPD10	SPD9	SPD8
		SPD7	SPD6	SPD5	SPD4	SPD3	SPD2	SPD1	SPD0
	SPSCR_1	-	-	-	-	-	-	SPSLN1	SPSLN0
	SPSSR_1	-	-	-	-	-	-	SPCP1	SPCP0
	SPBR_1	SPR7	SPR6	SPR5	SPR4	SPR3	SPR2	SPR1	SPR0
	SPDCR_1	TXDMY	SPLW1	SPLW0	-	-	-	-	-
	SPCKD_1	-	-	-	-	-	SCKDL2	SCKDL1	SCKDL0
	SSLND_1	-	-	-	-	-	SLNDL2	SLNDL1	SLNDL0
	SPND_1	-	-	-	-	-	SPNDL2	SPNDL1	SPNDL0
	SPCMD0_1	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0
		SSLKP	-	-	-	BRDV1	BRDV0	CPOL	CPHA
	SPCMD1_1	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0
		SSLKP	-	-	-	BRDV1	BRDV0	CPOL	CPHA
	SPCMD2_1	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0
		SSLKP	-	-	-	BRDV1	BRDV0	CPOL	CPHA
SPCMD3_1	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0	
	SSLKP	-	-	-	BRDV1	BRDV0	CPOL	CPHA	



Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Renesas serial peripheral interface	SPBFCR_1	TXRST	RXRST	TXTRG[1]	TXTRG[0]	-	RXTRG[2]	RXTRG[1]	RXTRG[0]
	SPBFDR_1	-	-	-	-	T[3]	T[2]	T[1]	T[0]
		-	-	R[5]	R[4]	R[3]	R[2]	R[1]	R[0]
	SPCR_2	SPRIE	SPE	SPTIE	SPEIE	MSTR	MODFEN	-	-
	SSLP_2	-	-	-	-	-	-	-	SSL0P
	SPPCR_2	-	-	MOIFE	MOIFV	-	-	-	SPLP
	SPSR_2	SPRF	TEND	SPTFEF	-	-	MODF	-	OVRF
	SPDR_2	SPD31	SPD30	SPD29	SPD28	SPD27	SPD26	SPD25	SPD24
		SPD23	SPD22	SPD21	SPD20	SPD19	SPD18	SPD17	SPD16
		SPD15	SPD14	SPD13	SPD12	SPD11	SPD10	SPD9	SPD8
		SPD7	SPD6	SPD5	SPD4	SPD3	SPD2	SPD1	SPD0
	SPSCR_2	-	-	-	-	-	-	SPSLN1	SPSLN0
	SPSSR_2	-	-	-	-	-	-	SPCP1	SPCP0
	SPBR_2	SPR7	SPR6	SPR5	SPR4	SPR3	SPR2	SPR1	SPR0
	SPDCR_2	TXDMY	SPLW1	SPLW0	-	-	-	-	-
	SPCKD_2	-	-	-	-	-	SCKDL2	SCKDL1	SCKDL0
	SSLND_2	-	-	-	-	-	SLNDL2	SLNDL1	SLNDL0
	SPND_2	-	-	-	-	-	SPNDL2	SPNDL1	SPNDL0
	SPCMD0_2	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0
		SSLKP	-	-	-	BRDV1	BRDV0	CPOL	CPHA
	SPCMD1_2	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0
		SSLKP	-	-	-	BRDV1	BRDV0	CPOL	CPHA
	SPCMD2_2	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0
		SSLKP	-	-	-	BRDV1	BRDV0	CPOL	CPHA
	SPCMD3_2	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0
		SSLKP	-	-	-	BRDV1	BRDV0	CPOL	CPHA
	SPBFCR_2	TXRST	RXRST	TXTRG[1]	TXTRG[0]	-	RXTRG[2]	RXTRG[1]	RXTRG[0]
	SPBFDR_2	-	-	-	-	T[3]	T[2]	T[1]	T[0]
		-	-	R[5]	R[4]	R[3]	R[2]	R[1]	R[0]
	SPCR_3	SPRIE	SPE	SPTIE	SPEIE	MSTR	MODFEN	-	-
	SSLP_3	-	-	-	-	-	-	-	SSL0P
	SPPCR_3	-	-	MOIFE	MOIFV	-	-	-	SPLP
	SPSR_3	SPRF	TEND	SPTFEF	-	-	MODF	-	OVRF
	SPDR_3	SPD31	SPD30	SPD29	SPD28	SPD27	SPD26	SPD25	SPD24
		SPD23	SPD22	SPD21	SPD20	SPD19	SPD18	SPD17	SPD16
		SPD15	SPD14	SPD13	SPD12	SPD11	SPD10	SPD9	SPD8
		SPD7	SPD6	SPD5	SPD4	SPD3	SPD2	SPD1	SPD0
	SPSCR_3	-	-	-	-	-	-	SPSLN1	SPSLN0
	SPSSR_3	-	-	-	-	-	-	SPCP1	SPCP0
	SPBR_3	SPR7	SPR6	SPR5	SPR4	SPR3	SPR2	SPR1	SPR0
	SPDCR_3	TXDMY	SPLW1	SPLW0	-	-	-	-	-
	SPCKD_3	-	-	-	-	-	SCKDL2	SCKDL1	SCKDL0
	SSLND_3	-	-	-	-	-	SLNDL2	SLNDL1	SLNDL0
	SPND_3	-	-	-	-	-	SPNDL2	SPNDL1	SPNDL0
	SPCMD0_3	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0
		SSLKP	-	-	-	BRDV1	BRDV0	CPOL	CPHA
	SPCMD1_3	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0
		SSLKP	-	-	-	BRDV1	BRDV0	CPOL	CPHA
	SPCMD2_3	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0
		SSLKP	-	-	-	BRDV1	BRDV0	CPOL	CPHA
	SPCMD3_3	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0
		SSLKP	-	-	-	BRDV1	BRDV0	CPOL	CPHA
	SPBFCR_3	TXRST	RXRST	TXTRG[1]	TXTRG[0]	-	RXTRG[2]	RXTRG[1]	RXTRG[0]
	SPBFDR_3	-	-	-	-	T[3]	T[2]	T[1]	T[0]
		-	-	R[5]	R[4]	R[3]	R[2]	R[1]	R[0]
	SPCR_4	SPRIE	SPE	SPTIE	SPEIE	MSTR	MODFEN	-	-
	SSLP_4	-	-	-	-	-	-	-	SSL0P

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Renesas serial peripheral interface	SPPCR_4	-	-	MOIFE	MOIFV	-	-	-	SPLP	
	SPSR_4	SPRF	TEND	SPTF	-	-	MODF	-	OVRF	
	SPDR_4	SPD31	SPD30	SPD29	SPD28	SPD27	SPD26	SPD25	SPD24	
		SPD23	SPD22	SPD21	SPD20	SPD19	SPD18	SPD17	SPD16	
		SPD15	SPD14	SPD13	SPD12	SPD11	SPD10	SPD9	SPD8	
		SPD7	SPD6	SPD5	SPD4	SPD3	SPD2	SPD1	SPD0	
	SPSCR_4	-	-	-	-	-	-	SPSLN1	SPSLN0	
	SPSSR_4	-	-	-	-	-	-	SPCP1	SPCP0	
	SPBR_4	SPR7	SPR6	SPR5	SPR4	SPR3	SPR2	SPR1	SPR0	
	SPDCR_4	TXDMY	SPLW1	SPLW0	-	-	-	-	-	
	SPCKD_4	-	-	-	-	-	SCKDL2	SCKDL1	SCKDL0	
	SSLND_4	-	-	-	-	-	SLNDL2	SLNDL1	SLNDL0	
	SPND_4	-	-	-	-	-	SPNDL2	SPNDL1	SPNDL0	
	SPCMD0_4	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0	
		SSLKP	-	-	-	BRDV1	BRDV0	CPOL	CPHA	
	SPCMD1_4	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0	
		SSLKP	-	-	-	BRDV1	BRDV0	CPOL	CPHA	
	SPCMD2_4	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0	
		SSLKP	-	-	-	BRDV1	BRDV0	CPOL	CPHA	
	SPCMD3_4	SCKDEN	SLNDEN	SPNDEN	LSBF	SPB3	SPB2	SPB1	SPB0	
		SSLKP	-	-	-	BRDV1	BRDV0	CPOL	CPHA	
	SPBFDR_4	TXRST	RXRST	TXTRG[1]	TXTRG[0]	-	RXTRG[2]	RXTRG[1]	RXTRG[0]	
	SPBFDR_4	-	-	-	-	T[3]	T[2]	T[1]	T[0]	
		-	-	R[5]	R[4]	R[3]	R[2]	R[1]	R[0]	
	SPI multi I/O bus controller	CMNCR_0	MD	-	-	-	-	-	-	SFDE
			MOIO3[1]	MOIO3[0]	MOIO2[1]	MOIO2[0]	MOIO1[1]	MOIO1[0]	MOIO0[1]	MOIO0[0]
			IO3FV[1]	IO3FV[0]	IO2FV[1]	IO2FV[0]	-	-	IO0FV[1]	IO0FV[0]
SSLDL_0		-	CPHAT	CPHAR	SSLP	CPOL	-	BSZ[1]	BSZ[0]	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	SPNDL[2]	SPNDL[1]	SPNDL[0]	
		-	-	-	-	-	SLNDL[2]	SLNDL[1]	SLNDL[0]	
SPBCR_0		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		SPBR[7]	SPBR[6]	SPBR[5]	SPBR[4]	SPBR[3]	SPBR[2]	SPBR[1]	SPBR[0]	
DRCR_0		-	-	-	-	-	BRDV[1]	BRDV[0]	-	
		-	-	-	-	-	-	-	SSLN	
		-	-	-	-	RBURST[3]	RBURST[2]	RBURST[1]	RBURST[0]	
		-	-	-	-	-	-	RCF	RBE	
DRCMR_0		-	-	-	-	-	-	-	SSLE	
		CMD[7]	CMD[6]	CMD[5]	CMD[4]	CMD[3]	CMD[2]	CMD[1]	CMD[0]	
		-	-	-	-	-	-	-	-	
DREAR_0		OCMD[7]	OCMD[6]	OCMD[5]	OCMD[4]	OCMD[3]	OCMD[2]	OCMD[1]	OCMD[0]	
		-	-	-	-	-	-	-	-	
		EAV[7]	EAV[6]	EAV[5]	EAV[4]	EAV[3]	EAV[2]	EAV[1]	EAV[0]	
		-	-	-	-	-	-	-	-	
DROPR_0		-	-	-	-	-	EAC[2]	EAC[1]	EAC[0]	
		OPD3[7]	OPD3[6]	OPD3[5]	OPD3[4]	OPD3[3]	OPD3[2]	OPD3[1]	OPD3[0]	
		OPD2[7]	OPD2[6]	OPD2[5]	OPD2[4]	OPD2[3]	OPD2[2]	OPD2[1]	OPD2[0]	
		OPD1[7]	OPD1[6]	OPD1[5]	OPD1[4]	OPD1[3]	OPD1[2]	OPD1[1]	OPD1[0]	
DRENDR_0		OPD0[7]	OPD0[6]	OPD0[5]	OPD0[4]	OPD0[3]	OPD0[2]	OPD0[1]	OPD0[0]	
		CDB[1]	CDB[0]	OCDB[1]	OCDB[0]	-	-	ADB[1]	ADB[0]	
	-	-	OPDB[1]	OPDB[0]	-	-	DRDB[1]	DRDB[0]		
	DME	CDE	-	OCDE	ADE[3]	ADE[2]	ADE[1]	ADE[0]		
-	-	OPDE[1]	OPDE[0]	-	-	-	-			

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
SPI multi I/O bus controller	SMCR_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	SSKLP
		-	-	-	-	-	SPIRE	SPIWE	SPIE
	SMCMR_0	-	-	-	-	-	-	-	-
		CMD[7]	CMD[6]	CMD[5]	CMD[4]	CMD[3]	CMD[2]	CMD[1]	CMD[0]
		-	-	-	-	-	-	-	-
	SMADR_0	OCMD[7]	OCMD[6]	OCMD[5]	OCMD[4]	OCMD[3]	OCMD[2]	OCMD[1]	OCMD[0]
		ADR[31]	ADR[30]	ADR[29]	ADR[28]	ADR[27]	ADR[26]	ADR[25]	ADR[24]
		ADR[23]	ADR[22]	ADR[21]	ADR[20]	ADR[19]	ADR[18]	ADR[17]	ADR[16]
		ADR[15]	ADR[14]	ADR[13]	ADR[12]	ADR[11]	ADR[10]	ADR[9]	ADR[8]
	SMOPR_0	ADR[7]	ADR[6]	ADR[5]	ADR[4]	ADR[3]	ADR[2]	ADR[1]	ADR[0]
		OPD3[7]	OPD3[6]	OPD3[5]	OPD3[4]	OPD3[3]	OPD3[2]	OPD3[1]	OPD3[0]
		OPD2[7]	OPD2[6]	OPD2[5]	OPD2[4]	OPD2[3]	OPD2[2]	OPD2[1]	OPD2[0]
		OPD1[7]	OPD1[6]	OPD1[5]	OPD1[4]	OPD1[3]	OPD1[2]	OPD1[1]	OPD1[0]
	SMENR_0	OPD0[7]	OPD0[6]	OPD0[5]	OPD0[4]	OPD0[3]	OPD0[2]	OPD0[1]	OPD0[0]
		CDB[1]	CDB[0]	OCDB[1]	OCDB[0]	-	-	ADB[1]	ADB[0]
		-	-	OPDB[1]	OPDB[0]	-	-	SPIDB[1]	SPIDB[0]
		DME	CDE	-	OCDE	ADE[3]	ADE[2]	ADE[1]	ADE[0]
	SMRDR0_0	OPDE[3]	OPDE[2]	OPDE[1]	OPDE[0]	SPIDE[3]	SPIDE[2]	SPIDE[1]	SPIDE[0]
		RDATA0[31]	RDATA0[30]	RDATA0[29]	RDATA0[28]	RDATA0[27]	RDATA0[26]	RDATA0[25]	RDATA0[24]
		RDATA0[23]	RDATA0[22]	RDATA0[21]	RDATA0[20]	RDATA0[19]	RDATA0[18]	RDATA0[17]	RDATA0[16]
		RDATA0[15]	RDATA0[14]	RDATA0[13]	RDATA0[12]	RDATA0[11]	RDATA0[10]	RDATA0[9]	RDATA0[8]
	SMRDR1_0	RDATA0[7]	RDATA0[6]	RDATA0[5]	RDATA0[4]	RDATA0[3]	RDATA0[2]	RDATA0[1]	RDATA0[0]
		RDATA1[31]	RDATA1[30]	RDATA1[29]	RDATA1[28]	RDATA1[27]	RDATA1[26]	RDATA1[25]	RDATA1[24]
		RDATA1[23]	RDATA1[22]	RDATA1[21]	RDATA1[20]	RDATA1[19]	RDATA1[18]	RDATA1[17]	RDATA1[16]
		RDATA1[15]	RDATA1[14]	RDATA1[13]	RDATA1[12]	RDATA1[11]	RDATA1[10]	RDATA1[9]	RDATA1[8]
	SMWDR0_0	RDATA1[7]	RDATA1[6]	RDATA1[5]	RDATA1[4]	RDATA1[3]	RDATA1[2]	RDATA1[1]	RDATA1[0]
		WDATA0[31]	WDATA0[30]	WDATA0[29]	WDATA0[28]	WDATA0[27]	WDATA0[26]	WDATA0[25]	WDATA0[24]
		WDATA0[23]	WDATA0[22]	WDATA0[21]	WDATA0[20]	WDATA0[19]	WDATA0[18]	WDATA0[17]	WDATA0[16]
		WDATA0[15]	WDATA0[14]	WDATA0[13]	WDATA0[12]	WDATA0[11]	WDATA0[10]	WDATA0[9]	WDATA0[8]
	SMWDR1_0	WDATA0[7]	WDATA0[6]	WDATA0[5]	WDATA0[4]	WDATA0[3]	WDATA0[2]	WDATA0[1]	WDATA0[0]
		WDATA1[31]	WDATA1[30]	WDATA1[29]	WDATA1[28]	WDATA1[27]	WDATA1[26]	WDATA1[25]	WDATA1[24]
		WDATA1[23]	WDATA1[22]	WDATA1[21]	WDATA1[20]	WDATA1[19]	WDATA1[18]	WDATA1[17]	WDATA1[16]
		WDATA1[15]	WDATA1[14]	WDATA1[13]	WDATA1[12]	WDATA1[11]	WDATA1[10]	WDATA1[9]	WDATA1[8]
	CMNSR_0	WDATA1[7]	WDATA1[6]	WDATA1[5]	WDATA1[4]	WDATA1[3]	WDATA1[2]	WDATA1[1]	WDATA1[0]
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	SSLF	TEND
	CKDLY_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		GB[7]	GB[6]	GB[5]	GB[4]	GB[3]	GB[2]	GB[1]	GB[0]
	DRDMCR_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	DMDB[1]	DMDB[0]
		-	-	-	-	-	-	-	-
		-	-	-	-	-	DMCYC[2]	DMCYC[1]	DMCYC[0]
	DRDREN_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	ADDRE
		-	-	-	OPDRE	-	-	-	DRDRE
	SMDMCR_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	DMDB[1]	DMDB[0]
		-	-	-	-	-	-	-	-
		-	-	-	-	-	DMCYC[2]	DMCYC[1]	DMCYC[0]
	SMDREN_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	ADDRE
		-	-	-	OPDRE	-	-	-	DRDRE

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
SPI multi I/O bus controller	SPOPLY_0	GB[7]	GB[6]	GB[5]	GB[4]	GB[3]	GB[2]	GB[1]	GB[0]	
		-	-	-	-	-	-	-	-	
		SPOPLY[15]	SPOPLY[14]	SPOPLY[13]	SPOPLY[12]	SPOPLY[11]	SPOPLY[10]	SPOPLY[9]	SPOPLY[8]	
	CMNCR_1	SPOPLY[7]	SPOPLY[6]	SPOPLY[5]	SPOPLY[4]	SPOPLY[3]	SPOPLY[2]	SPOPLY[1]	SPOPLY[0]	
		MD	-	-	-	-	-	-	SFDE	
		MOIO3[1]	MOIO3[0]	MOIO2[1]	MOIO2[0]	MOIO1[1]	MOIO1[0]	MOIO0[1]	MOIO0[0]	
		IO3FV[1]	IO3FV[0]	IO2FV[1]	IO2FV[0]	-	-	IO0FV[1]	IO0FV[0]	
	SSLDL_1	-	CPHAT	CPHAR	SSLP	CPOL	-	BSZ[1]	BSZ[0]	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	SPNDL[2]	SPNDL[1]	SPNDL[0]	
		-	-	-	-	-	SLNDL[2]	SLNDL[1]	SLNDL[0]	
	SPBCR_1	-	-	-	-	-	-	SCKDL[2]	SCKDL[1]	SCKDL[0]
		-	-	-	-	-	-	-	-	
		SPBR[7]	SPBR[6]	SPBR[5]	SPBR[4]	SPBR[3]	SPBR[2]	SPBR[1]	SPBR[0]	
		-	-	-	-	-	-	BRDV[1]	BRDV[0]	
	DRCCR_1	-	-	-	-	-	-	-	SSLN	
		-	-	-	-	RBURST[3]	RBURST[2]	RBURST[1]	RBURST[0]	
		-	-	-	-	-	-	RCF	RBE	
		-	-	-	-	-	-	-	SSLE	
	DRCMR_1	-	-	-	-	-	-	-	-	
		CMD[7]	CMD[6]	CMD[5]	CMD[4]	CMD[3]	CMD[2]	CMD[1]	CMD[0]	
		-	-	-	-	-	-	-	-	
	DREAR_1	OCMD[7]	OCMD[6]	OCMD[5]	OCMD[4]	OCMD[3]	OCMD[2]	OCMD[1]	OCMD[0]	
		-	-	-	-	-	-	-	-	
		EAV[7]	EAV[6]	EAV[5]	EAV[4]	EAV[3]	EAV[2]	EAV[1]	EAV[0]	
	DROPR_1	-	-	-	-	-	-	-	-	
		OPD3[7]	OPD3[6]	OPD3[5]	OPD3[4]	OPD3[3]	OPD3[2]	OPD3[1]	OPD3[0]	
		OPD2[7]	OPD2[6]	OPD2[5]	OPD2[4]	OPD2[3]	OPD2[2]	OPD2[1]	OPD2[0]	
		OPD1[7]	OPD1[6]	OPD1[5]	OPD1[4]	OPD1[3]	OPD1[2]	OPD1[1]	OPD1[0]	
	DREN_1	OPD0[7]	OPD0[6]	OPD0[5]	OPD0[4]	OPD0[3]	OPD0[2]	OPD0[1]	OPD0[0]	
		CDB[1]	CDB[0]	OCDB[1]	OCDB[0]	-	-	ADB[1]	ADB[0]	
		-	-	OPDB[1]	OPDB[0]	-	-	DRDB[1]	DRDB[0]	
		DME	CDE	-	OCDE	ADE[3]	ADE[2]	ADE[1]	ADE[0]	
	SMCR_1	OPDE[3]	OPDE[2]	OPDE[1]	OPDE[0]	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	SPIRE	SPIWE	SPIE	
	SMCMR_1	-	-	-	-	-	-	-	-	
		CMD[7]	CMD[6]	CMD[5]	CMD[4]	CMD[3]	CMD[2]	CMD[1]	CMD[0]	
		-	-	-	-	-	-	-	-	
	SMADR_1	OCMD[7]	OCMD[6]	OCMD[5]	OCMD[4]	OCMD[3]	OCMD[2]	OCMD[1]	OCMD[0]	
		ADR[31]	ADR[30]	ADR[29]	ADR[28]	ADR[27]	ADR[26]	ADR[25]	ADR[24]	
		ADR[23]	ADR[22]	ADR[21]	ADR[20]	ADR[19]	ADR[18]	ADR[17]	ADR[16]	
		ADR[15]	ADR[14]	ADR[13]	ADR[12]	ADR[11]	ADR[10]	ADR[9]	ADR[8]	
	SMOPR_1	ADR[7]	ADR[6]	ADR[5]	ADR[4]	ADR[3]	ADR[2]	ADR[1]	ADR[0]	
		OPD3[7]	OPD3[6]	OPD3[5]	OPD3[4]	OPD3[3]	OPD3[2]	OPD3[1]	OPD3[0]	
		OPD2[7]	OPD2[6]	OPD2[5]	OPD2[4]	OPD2[3]	OPD2[2]	OPD2[1]	OPD2[0]	
		OPD1[7]	OPD1[6]	OPD1[5]	OPD1[4]	OPD1[3]	OPD1[2]	OPD1[1]	OPD1[0]	
	SMENR_1	OPD0[7]	OPD0[6]	OPD0[5]	OPD0[4]	OPD0[3]	OPD0[2]	OPD0[1]	OPD0[0]	
		CDB[1]	CDB[0]	OCDB[1]	OCDB[0]	-	-	ADB[1]	ADB[0]	
		-	-	OPDB[1]	OPDB[0]	-	-	SPIDB[1]	SPIDB[0]	
		DME	CDE	-	OCDE	ADE[3]	ADE[2]	ADE[1]	ADE[0]	
	SMRDR0_1	OPDE[3]	OPDE[2]	OPDE[1]	OPDE[0]	SPIDE[3]	SPIDE[2]	SPIDE[1]	SPIDE[0]	
		RDATA0[31]	RDATA0[30]	RDATA0[29]	RDATA0[28]	RDATA0[27]	RDATA0[26]	RDATA0[25]	RDATA0[24]	
		RDATA0[23]	RDATA0[22]	RDATA0[21]	RDATA0[20]	RDATA0[19]	RDATA0[18]	RDATA0[17]	RDATA0[16]	
		RDATA0[15]	RDATA0[14]	RDATA0[13]	RDATA0[12]	RDATA0[11]	RDATA0[10]	RDATA0[9]	RDATA0[8]	
	SMRDR0_1	RDATA0[7]	RDATA0[6]	RDATA0[5]	RDATA0[4]	RDATA0[3]	RDATA0[2]	RDATA0[1]	RDATA0[0]	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0		
SPI multi I/O bus controller	SMRDR1_1	RDATA1[31]	RDATA1[30]	RDATA1[29]	RDATA1[28]	RDATA1[27]	RDATA1[26]	RDATA1[25]	RDATA1[24]		
		RDATA1[23]	RDATA1[22]	RDATA1[21]	RDATA1[20]	RDATA1[19]	RDATA1[18]	RDATA1[17]	RDATA1[16]		
		RDATA1[15]	RDATA1[14]	RDATA1[13]	RDATA1[12]	RDATA1[11]	RDATA1[10]	RDATA1[9]	RDATA1[8]		
	SMWDR0_1	WDATA0[31]	WDATA0[30]	WDATA0[29]	WDATA0[28]	WDATA0[27]	WDATA0[26]	WDATA0[25]	WDATA0[24]		
		WDATA0[23]	WDATA0[22]	WDATA0[21]	WDATA0[20]	WDATA0[19]	WDATA0[18]	WDATA0[17]	WDATA0[16]		
		WDATA0[15]	WDATA0[14]	WDATA0[13]	WDATA0[12]	WDATA0[11]	WDATA0[10]	WDATA0[9]	WDATA0[8]		
		WDATA0[7]	WDATA0[6]	WDATA0[5]	WDATA0[4]	WDATA0[3]	WDATA0[2]	WDATA0[1]	WDATA0[0]		
	SMWDR1_1	WDATA1[31]	WDATA1[30]	WDATA1[29]	WDATA1[28]	WDATA1[27]	WDATA1[26]	WDATA1[25]	WDATA1[24]		
		WDATA1[23]	WDATA1[22]	WDATA1[21]	WDATA1[20]	WDATA1[19]	WDATA1[18]	WDATA1[17]	WDATA1[16]		
		WDATA1[15]	WDATA1[14]	WDATA1[13]	WDATA1[12]	WDATA1[11]	WDATA1[10]	WDATA1[9]	WDATA1[8]		
		WDATA1[7]	WDATA1[6]	WDATA1[5]	WDATA1[4]	WDATA1[3]	WDATA1[2]	WDATA1[1]	WDATA1[0]		
	CMNSR_1	-	-	-	-	-	-	-	-		
		-	-	-	-	-	-	-	-		
		-	-	-	-	-	-	-	-		
		-	-	-	-	-	-	SSLF	TEND		
	CKDLY_1	-	-	-	-	-	-	-	-		
		-	-	-	-	-	-	-	-		
		GB[7]	GB[6]	GB[5]	GB[4]	GB[3]	GB[2]	GB[1]	GB[0]		
	DRDMCR_1	-	-	-	-	-	-	-	-		
		-	-	-	-	-	-	DMDB[1]	DMDB[0]		
		-	-	-	-	-	-	-	-		
		-	-	-	-	-	DMCYC[2]	DMCYC[1]	DMCYC[0]		
	DRDREN_1	-	-	-	-	-	-	-	-		
		-	-	-	-	-	-	-	ADDRE		
		-	-	-	OPDRE	-	-	-	DRDRE		
	SMDMCR_1	-	-	-	-	-	-	-	-		
		-	-	-	-	-	-	DMDB[1]	DMDB[0]		
		-	-	-	-	-	DMCYC[2]	DMCYC[1]	DMCYC[0]		
	SMDREN_1	-	-	-	-	-	-	-	-		
		-	-	-	-	-	-	-	-		
		-	-	-	-	-	-	-	ADDRE		
		-	-	-	OPDRE	-	-	-	DRDRE		
	SPODLY_1	GB[7]	GB[6]	GB[5]	GB[4]	GB[3]	GB[2]	GB[1]	GB[0]		
		-	-	-	-	-	-	-	-		
		SPODLY[15]	SPODLY[14]	SPODLY[13]	SPODLY[12]	SPODLY[11]	SPODLY[10]	SPODLY[9]	SPODLY[8]		
		SPODLY[7]	SPODLY[6]	SPODLY[5]	SPODLY[4]	SPODLY[3]	SPODLY[2]	SPODLY[1]	SPODLY[0]		
	I2C bus interface	RIICOCR1	ICE	IICRST	CLO	SOWP	SCLO	SDAO	SCLI	SDAI	
		RIICOCR2	BBSY	MST	TRS	-	SP	RS	ST	-	
		RIICOMR1	MTWP	CKS[2]	CKS[1]	CKS[0]	BCWP	BC[2]	BC[1]	BC[0]	
		RIICOMR2	DLCS	SDDL[2]	SDDL[1]	SDDL[0]	-	TMOH	TMOL	TMOS	
		RIICOMR3	SMBE	WAIT	RDRFS	ACKWP	ACKBT	ACKBR	NF[1]	NF[0]	
		RIICOFER	-	SCLE	NFE	NACKE	SALE	NALE	MALE	TMOE	
		RIICOSER	HOAE	-	DIDE	-	GCE	SAR2E	SAR1E	SAR0E	
		RIICQIER	TIE	TEIE	RIE	NAKIE	SPIE	STIE	ALIE	TMOIE	
		RIICOSR1	HOA	-	DID	-	GCA	AAS2	AAS1	AAS0	
		RIICOSR2	TDRE	TEND	RDRF	NACKF	STOP	START	AL	TMOF	
		RIICOSAR0	FS0	-	-	-	-	-	-	SVA[9]	SVA[8]
			SVA[7]	SVA[6]	SVA[5]	SVA[4]	SVA[3]	SVA[2]	SVA[1]	SVA0	
RIICOSAR1		FS1	-	-	-	-	-	-	SVA[9]	SVA[8]	
		SVA[7]	SVA[6]	SVA[5]	SVA[4]	SVA[3]	SVA[2]	SVA[1]	SVA0		
RIICOSAR2		FS2	-	-	-	-	-	-	SVA[9]	SVA[8]	
		SVA[7]	SVA[6]	SVA[5]	SVA[4]	SVA[3]	SVA[2]	SVA[1]	SVA0		
RIICOBRL		-	-	-	BRL[4]	BRL[3]	BRL[2]	BRL[1]	BRL[0]		
RIICOBRLH		-	-	-	BRH[4]	BRH[3]	BRH[2]	BRH[1]	BRH[0]		
RIICODRT		DRT[7]	DRT[6]	DRT[5]	DRT[4]	DRT[3]	DRT[2]	DRT[1]	DRT[0]		
RIICODRR		DRR[7]	DRR[6]	DRR[5]	DRR[4]	DRR[3]	DRR[2]	DRR[1]	DRR[0]		

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
I2C bus interface	RIIC1CR1	ICE	IICRST	CLO	SOWP	SCLO	SDAO	SCLI	SDAI	
	RIIC1CR2	BBSY	MST	TRS	-	SP	RS	ST	-	
	RIIC1MR1	MTWP	CK[S]2	CK[S]1	CK[S]0	BCWP	BC[2]	BC[1]	BC[0]	
	RIIC1MR2	DLCS	SDDL[2]	SDDL[1]	SDDL[0]	-	TMOH	TMOL	TMOS	
	RIIC1MR3	SMBE	WAIT	RDRFS	ACKWP	ACKBT	ACKBR	NF[1]	NF[0]	
	RIIC1FER	-	SCLE	NFE	NACKE	SALE	NALE	MALE	TMOE	
	RIIC1SER	HOAE	-	DIDE	-	GCE	SAR2E	SAR1E	SAR0E	
	RIIC1IER	TIE	TEIE	RIE	NAKIE	SPIE	STIE	ALIE	TMOIE	
	RIIC1SR1	HOA	-	DID	-	GCA	AAS2	AAS1	AAS0	
	RIIC1SR2	TDRE	TEND	RDRF	NACKF	STOP	START	AL	TMOF	
	RIIC1SAR0	FS0	-	-	-	-	-	-	SVA[9]	SVA[8]
		SVA[7]	SVA[6]	SVA[5]	SVA[4]	SVA[3]	SVA[2]	SVA[1]	SVA0	
	RIIC1SAR1	FS1	-	-	-	-	-	-	SVA[9]	SVA[8]
		SVA[7]	SVA[6]	SVA[5]	SVA[4]	SVA[3]	SVA[2]	SVA[1]	SVA0	
	RIIC1SAR2	FS2	-	-	-	-	-	-	SVA[9]	SVA[8]
		SVA[7]	SVA[6]	SVA[5]	SVA[4]	SVA[3]	SVA[2]	SVA[1]	SVA0	
	RIIC1BRL	-	-	-	BRL[4]	BRL[3]	BRL[2]	BRL[1]	BRL[0]	
	RIIC1BRH	-	-	-	BRH[4]	BRH[3]	BRH[2]	BRH[1]	BRH[0]	
	RIIC1DRT	DRT[7]	DRT[6]	DRT[5]	DRT[4]	DRT[3]	DRT[2]	DRT[1]	DRT[0]	
	RIIC1DRR	DRR[7]	DRR[6]	DRR[5]	DRR[4]	DRR[3]	DRR[2]	DRR[1]	DRR[0]	
	RIIC2CR1	ICE	IICRST	CLO	SOWP	SCLO	SDAO	SCLI	SDAI	
	RIIC2CR2	BBSY	MST	TRS	-	SP	RS	ST	-	
	RIIC2MR1	MTWP	CK[S]2	CK[S]1	CK[S]0	BCWP	BC[2]	BC[1]	BC[0]	
	RIIC2MR2	DLCS	SDDL[2]	SDDL[1]	SDDL[0]	-	TMOH	TMOL	TMOS	
	RIIC2MR3	SMBE	WAIT	RDRFS	ACKWP	ACKBT	ACKBR	NF[1]	NF[0]	
	RIIC2FER	-	SCLE	NFE	NACKE	SALE	NALE	MALE	TMOE	
	RIIC2SER	HOAE	-	DIDE	-	GCE	SAR2E	SAR1E	SAR0E	
	RIIC2IER	TIE	TEIE	RIE	NAKIE	SPIE	STIE	ALIE	TMOIE	
	RIIC2SR1	HOA	-	DID	-	GCA	AAS2	AAS1	AAS0	
	RIIC2SR2	TDRE	TEND	RDRF	NACKF	STOP	START	AL	TMOF	
	RIIC2SAR0	FS0	-	-	-	-	-	-	SVA[9]	SVA[8]
		SVA[7]	SVA[6]	SVA[5]	SVA[4]	SVA[3]	SVA[2]	SVA[1]	SVA0	
	RIIC2SAR1	FS1	-	-	-	-	-	-	SVA[9]	SVA[8]
		SVA[7]	SVA[6]	SVA[5]	SVA[4]	SVA[3]	SVA[2]	SVA[1]	SVA0	
	RIIC2SAR2	FS2	-	-	-	-	-	-	SVA[9]	SVA[8]
		SVA[7]	SVA[6]	SVA[5]	SVA[4]	SVA[3]	SVA[2]	SVA[1]	SVA0	
	RIIC2BRL	-	-	-	BRL[4]	BRL[3]	BRL[2]	BRL[1]	BRL[0]	
	RIIC2BRH	-	-	-	BRH[4]	BRH[3]	BRH[2]	BRH[1]	BRH[0]	
	RIIC2DRT	DRT[7]	DRT[6]	DRT[5]	DRT[4]	DRT[3]	DRT[2]	DRT[1]	DRT[0]	
	RIIC2DRR	DRR[7]	DRR[6]	DRR[5]	DRR[4]	DRR[3]	DRR[2]	DRR[1]	DRR[0]	
	RIIC3CR1	ICE	IICRST	CLO	SOWP	SCLO	SDAO	SCLI	SDAI	
	RIIC3CR2	BBSY	MST	TRS	-	SP	RS	ST	-	
	RIIC3MR1	MTWP	CK[S]2	CK[S]1	CK[S]0	BCWP	BC[2]	BC[1]	BC[0]	
	RIIC3MR2	DLCS	SDDL[2]	SDDL[1]	SDDL[0]	-	TMOH	TMOL	TMOS	
	RIIC3MR3	SMBE	WAIT	RDRFS	ACKWP	ACKBT	ACKBR	NF[1]	NF[0]	
	RIIC3FER	-	SCLE	NFE	NACKE	SALE	NALE	MALE	TMOE	
	RIIC3SER	HOAE	-	DIDE	-	GCE	SAR2E	SAR1E	SAR0E	
	RIIC3IER	TIE	TEIE	RIE	NAKIE	SPIE	STIE	ALIE	TMOIE	
RIIC3SR1	HOA	-	DID	-	GCA	AAS2	AAS1	AAS0		
RIIC3SR2	TDRE	TEND	RDRF	NACKF	STOP	START	AL	TMOF		
RIIC3SAR0	FS0	-	-	-	-	-	-	SVA[9]	SVA[8]	
	SVA[7]	SVA[6]	SVA[5]	SVA[4]	SVA[3]	SVA[2]	SVA[1]	SVA0		
RIIC3SAR1	FS1	-	-	-	-	-	-	SVA[9]	SVA[8]	
	SVA[7]	SVA[6]	SVA[5]	SVA[4]	SVA[3]	SVA[2]	SVA[1]	SVA0		
RIIC3SAR2	FS2	-	-	-	-	-	-	SVA[9]	SVA[8]	
	SVA[7]	SVA[6]	SVA[5]	SVA[4]	SVA[3]	SVA[2]	SVA[1]	SVA0		
RIIC3BRL	-	-	-	BRL[4]	BRL[3]	BRL[2]	BRL[1]	BRL[0]		
RIIC3BRH	-	-	-	BRH[4]	BRH[3]	BRH[2]	BRH[1]	BRH[0]		
RIIC3DRT	DRT[7]	DRT[6]	DRT[5]	DRT[4]	DRT[3]	DRT[2]	DRT[1]	DRT[0]		
RIIC3DRR	DRR[7]	DRR[6]	DRR[5]	DRR[4]	DRR[3]	DRR[2]	DRR[1]	DRR[0]		

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Serial sound interface	SSICR_0	-	CKS	TUIEN	TOIEN	RUIEN	ROIEN	IEN	-
		CHNL[1]	CHNL[0]	DWL[2]	DWL[1]	DWL[0]	SWL[2]	SWL[1]	SWL[0]
		SCKD	SWSD	SCKP	SWSP	SPDP	SDTA	PDTA	DEL
		CKDV[3]	CKDV[2]	CKDV[1]	CKDV[0]	MUEN	-	TEN	REN
	SSISR_0	-	-	TUIRQ	TOIRQ	RUIRQ	ROIHQ	IIRQ	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	TCHNO[1]	TCHNO[0]	TSWNO	RCHNO[1]	RCHNO[0]	RSWNO	IDST
	SSIFCR_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		TTRG[1]	TTRG[0]	RTRG[1]	RTRG[0]	TIE	RIE	TFRST	RFRST
	SSIFSR_0	-	-	-	-	TDC[3]	TDC[2]	TDC[1]	TDC[0]
		-	-	-	-	-	-	-	TDE
		-	-	-	-	RDC[3]	RDC[2]	RDC[1]	RDC[0]
		-	-	-	-	-	-	-	RDF
	SSIFTDR_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	SSIFRDR_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	SSITDMR_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	RXDMUTE	-
		-	-	-	-	-	-	-	CONT
		-	-	-	-	-	-	-	TDM
	SSIFCCR_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	FIEN
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	FCEN
	SSIFCMR_0	-	-	MAXV[13]	MAXV[12]	MAXV[11]	MAXV[10]	MAXV[9]	MAXV[8]
		MAXV[7]	MAXV[6]	MAXV[5]	MAXV[4]	MAXV[3]	MAXV[2]	MAXV[1]	MAXV[0]
		-	-	MINV[13]	MINV[12]	MINV[11]	MINV[10]	MINV[9]	MINV[8]
		MINV[7]	MINV[6]	MINV[5]	MINV[4]	MINV[3]	MINV[2]	MINV[1]	MINV[0]
	SSIFCSR_0	-	-	-	-	-	-	-	FCIRQ
		-	-	-	-	-	-	-	-
		-	-	VALUE[13]	VALUE[12]	VALUE[11]	VALUE[10]	VALUE[9]	VALUE[8]
		VALUE[7]	VALUE[6]	VALUE[5]	VALUE[4]	VALUE[3]	VALUE[2]	VALUE[1]	VALUE[0]
	SSICR_1	-	CKS	TUIEN	TOIEN	RUIEN	ROIEN	IEN	-
		CHNL[1]	CHNL[0]	DWL[2]	DWL[1]	DWL[0]	SWL[2]	SWL[1]	SWL[0]
		SCKD	SWSD	SCKP	SWSP	SPDP	SDTA	PDTA	DEL
		CKDV[3]	CKDV[2]	CKDV[1]	CKDV[0]	MUEN	-	TEN	REN
	SSISR_1	-	-	TUIRQ	TOIRQ	RUIRQ	ROIHQ	IIRQ	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	TCHNO[1]	TCHNO[0]	TSWNO	RCHNO[1]	RCHNO[0]	RSWNO	IDST
	SSIFCR_1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		TTRG[1]	TTRG[0]	RTRG[1]	RTRG[0]	TIE	RIE	TFRST	RFRST
	SSIFSR_1	-	-	-	-	TDC[3]	TDC[2]	TDC[1]	TDC[0]
		-	-	-	-	-	-	-	TDE
		-	-	-	-	RDC[3]	RDC[2]	RDC[1]	RDC[0]
		-	-	-	-	-	-	-	RDF

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Serial sound interface	SSIFDR_1								
	SSIFRDR_1								
	SSITDMR_1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	RXDMUTE
		-	-	-	-	-	-	-	CONT
		-	-	-	-	-	-	-	TDM
	SSIFCCR_1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	FIEN
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	FCEN
	SSIFCMR_1	-	-	MAXV[13]	MAXV[12]	MAXV[11]	MAXV[10]	MAXV[9]	MAXV[8]
		MAXV[7]	MAXV[6]	MAXV[5]	MAXV[4]	MAXV[3]	MAXV[2]	MAXV[1]	MAXV[0]
		-	-	MINV[13]	MINV[12]	MINV[11]	MINV[10]	MINV[9]	MINV[8]
		MINV[7]	MINV[6]	MINV[5]	MINV[4]	MINV[3]	MINV[2]	MINV[1]	MINV[0]
	SSIFCSR_1	-	-	-	-	-	-	-	FCIRQ
		-	-	-	-	-	-	-	-
		-	-	VALUE[13]	VALUE[12]	VALUE[11]	VALUE[10]	VALUE[9]	VALUE[8]
		VALUE[7]	VALUE[6]	VALUE[5]	VALUE[4]	VALUE[3]	VALUE[2]	VALUE[1]	VALUE[0]
	SSICR_2	-	CKS	TUIEN	TOIEN	RUIEN	ROIEN	IEN	-
		CHNL[1]	CHNL[0]	DWL[2]	DWL[1]	DWL[0]	SWL[2]	SWL[1]	SWL[0]
		SCKD	SWSD	SCKP	SWSP	SPDP	SDTA	PDTA	DEL
		CKDV[3]	CKDV[2]	CKDV[1]	CKDV[0]	MUEN	-	TEN	REN
	SSISR_2	-	-	TUIRQ	TOIRQ	RUIRQ	ROIHQ	IIRQ	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	TCHNO[1]	TCHNO[0]	TSWNO	RCHNO[1]	RCHNO[0]	RSWNO	IDST
	SSIFCR_2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		TTRG[1]	TTRG[0]	RTRG[1]	RTRG[0]	TIE	RIE	TFRST	RFRST
	SSIFSR_2	-	-	-	-	TDC[3]	TDC[2]	TDC[1]	TDC[0]
		-	-	-	-	-	-	-	TDE
		-	-	-	-	RDC[3]	RDC[2]	RDC[1]	RDC[0]
		-	-	-	-	-	-	-	RDF
	SSIFDR_2								
	SSIFRDR_2								
	SSITDMR_2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	RXDMUTE
		-	-	-	-	-	-	-	CONT
		-	-	-	-	-	-	-	TDM
	SSIFCCR_2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	FIEN
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	FCEN



Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Serial sound interface	SSIFCMR_2	-	-	MAXV[13]	MAXV[12]	MAXV[11]	MAXV[10]	MAXV[9]	MAXV[8]	
		MAXV[7]	MAXV[6]	MAXV[5]	MAXV[4]	MAXV[3]	MAXV[2]	MAXV[1]	MAXV[0]	
		-	-	MINV[13]	MINV[12]	MINV[11]	MINV[10]	MINV[9]	MINV[8]	
		MINV[7]	MINV[6]	MINV[5]	MINV[4]	MINV[3]	MINV[2]	MINV[1]	MINV[0]	
	SSIFCSR_2	-	-	-	-	-	-	-	-	FCIRQ
		-	-	-	-	-	-	-	-	-
		-	-	VALUE[13]	VALUE[12]	VALUE[11]	VALUE[10]	VALUE[9]	VALUE[8]	-
		VALUE[7]	VALUE[6]	VALUE[5]	VALUE[4]	VALUE[3]	VALUE[2]	VALUE[1]	VALUE[0]	-
	SSICR_3	-	CKS	TUIEN	TOIEN	RUIEN	ROIEN	IIEN	-	-
		CHNL[1]	CHNL[0]	DWL[2]	DWL[1]	DWL[0]	SWL[2]	SWL[1]	SWL[0]	-
		SCKD	SWSD	SCKP	SWSP	SPDP	SDTA	PDTA	DEL	-
		CKDV[3]	CKDV[2]	CKDV[1]	CKDV[0]	MUEN	-	TEN	REN	-
	SSISR_3	-	-	TUIRQ	TOIRQ	RUIRQ	ROIHQ	IIRQ	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	TCHNO[1]	TCHNO[0]	TSWNO	RCHNO[1]	RCHNO[0]	RSWNO	IDST	-
	SSIFCR_3	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		TTRG[1]	TTRG[0]	RTRG[1]	RTRG[0]	TIE	RIE	TFRST	RFRST	-
	SSIFSR_3	-	-	-	-	TDC[3]	TDC[2]	TDC[1]	TDC[0]	-
		-	-	-	-	-	-	-	-	TDE
		-	-	-	-	RDC[3]	RDC[2]	RDC[1]	RDC[0]	-
		-	-	-	-	-	-	-	-	RDF
	SSIFTDR_3	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
	SSIFRDR_3	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
	SSITDMR_3	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	RXDMUTE	-	-
		-	-	-	-	-	-	-	CONT	-
		-	-	-	-	-	-	-	TDM	-
	SSIFCCR_3	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	FIEN
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	FCEN
	SSIFCMR_3	-	-	MAXV[13]	MAXV[12]	MAXV[11]	MAXV[10]	MAXV[9]	MAXV[8]	-
		MAXV[7]	MAXV[6]	MAXV[5]	MAXV[4]	MAXV[3]	MAXV[2]	MAXV[1]	MAXV[0]	-
		-	-	MINV[13]	MINV[12]	MINV[11]	MINV[10]	MINV[9]	MINV[8]	-
		MINV[7]	MINV[6]	MINV[5]	MINV[4]	MINV[3]	MINV[2]	MINV[1]	MINV[0]	-
	SSIFCSR_3	-	-	-	-	-	-	-	-	FCIRQ
		-	-	-	-	-	-	-	-	-
		-	-	VALUE[13]	VALUE[12]	VALUE[11]	VALUE[10]	VALUE[9]	VALUE[8]	-
		VALUE[7]	VALUE[6]	VALUE[5]	VALUE[4]	VALUE[3]	VALUE[2]	VALUE[1]	VALUE[0]	-
	SSICR_4	-	CKS	TUIEN	TOIEN	RUIEN	ROIEN	IIEN	-	-
		CHNL[1]	CHNL[0]	DWL[2]	DWL[1]	DWL[0]	SWL[2]	SWL[1]	SWL[0]	-
		SCKD	SWSD	SCKP	SWSP	SPDP	SDTA	PDTA	DEL	-
		CKDV[3]	CKDV[2]	CKDV[1]	CKDV[0]	MUEN	-	TEN	REN	-
	SSISR_4	-	-	TUIRQ	TOIRQ	RUIRQ	ROIHQ	IIRQ	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	TCHNO[1]	TCHNO[0]	TSWNO	RCHNO[1]	RCHNO[0]	RSWNO	IDST	-

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Serial sound interface	SSIFCR_4	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		TTRG[1]	TTRG[0]	RTRG[1]	RTRG[0]	TIE	RIE	TFRST	RFRST
	SSIFSR_4	-	-	-	-	TDC[3]	TDC[2]	TDC[1]	TDC[0]
		-	-	-	-	-	-	-	TDE
		-	-	-	-	RDC[3]	RDC[2]	RDC[1]	RDC[0]
		-	-	-	-	-	-	-	RDF
	SSIFTDR_4	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	SSIFRDR_4	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	SSITDMR_4	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	RXDMUTE	-
		-	-	-	-	-	-	-	CONT
		-	-	-	-	-	-	-	TDM
	SSIFCCR_4	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	FIEN
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	FCEN
	SSIFCMR_4	-	-	MAXV[13]	MAXV[12]	MAXV[11]	MAXV[10]	MAXV[9]	MAXV[8]
		MAXV[7]	MAXV[6]	MAXV[5]	MAXV[4]	MAXV[3]	MAXV[2]	MAXV[1]	MAXV[0]
		-	-	MINV[13]	MINV[12]	MINV[11]	MINV[10]	MINV[9]	MINV[8]
		MINV[7]	MINV[6]	MINV[5]	MINV[4]	MINV[3]	MINV[2]	MINV[1]	MINV[0]
	SSIFCSR_4	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	VALUE[13]	VALUE[12]	VALUE[11]	VALUE[10]	VALUE[9]	VALUE[8]
		VALUE[7]	VALUE[6]	VALUE[5]	VALUE[4]	VALUE[3]	VALUE[2]	VALUE[1]	VALUE[0]
	SSICR_5	-	CKS	TUIEN	TOIEN	RUIEN	ROIEN	IIEN	-
		CHNL[1]	CHNL[0]	DWL[2]	DWL[1]	DWL[0]	SWL[2]	SWL[1]	SWL[0]
		SCKD	SWSD	SCKP	SWSP	SPDP	SDTA	PDTA	DEL
		CKDV[3]	CKDV[2]	CKDV[1]	CKDV[0]	MUEN	-	TEN	REN
	SSISR_5	-	-	TUIRQ	TOIRQ	RUIRQ	ROIHQ	IIRQ	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	TCHNO[1]	TCHNO[0]	TSWNO	RCHNO[1]	RCHNO[0]	RSWNO	IDST
	SSIFCR_5	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		TTRG[1]	TTRG[0]	RTRG[1]	RTRG[0]	TIE	RIE	TFRST	RFRST
	SSIFSR_5	-	-	-	-	TDC[3]	TDC[2]	TDC[1]	TDC[0]
		-	-	-	-	-	-	-	TDE
		-	-	-	-	RDC[3]	RDC[2]	RDC[1]	RDC[0]
		-	-	-	-	-	-	-	RDF
	SSIFTDR_5	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	SSIFRDR_5	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Serial sound interface	SSITDMR_5	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	RXDMUTE	
		-	-	-	-	-	-	-	-	CONT
		-	-	-	-	-	-	-	-	TDM
	SSIFCCR_5	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	FIEN
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	FCEN
	SSIFCMR_5	-	-	MAXV[13]	MAXV[12]	MAXV[11]	MAXV[10]	MAXV[9]	MAXV[8]	MAXV[7]
		MAXV[7]	MAXV[6]	MAXV[5]	MAXV[4]	MAXV[3]	MAXV[2]	MAXV[1]	MAXV[0]	-
		-	-	MINV[13]	MINV[12]	MINV[11]	MINV[10]	MINV[9]	MINV[8]	MINV[7]
		MINV[7]	MINV[6]	MINV[5]	MINV[4]	MINV[3]	MINV[2]	MINV[1]	MINV[0]	-
	SSIFCSR_5	-	-	-	-	-	-	-	-	FCIRQ
		-	-	-	-	-	-	-	-	-
		-	-	VALUE[13]	VALUE[12]	VALUE[11]	VALUE[10]	VALUE[9]	VALUE[8]	VALUE[7]
		VALUE[7]	VALUE[6]	VALUE[5]	VALUE[4]	VALUE[3]	VALUE[2]	VALUE[1]	VALUE[0]	-
Media local bus	DCCR	MDE	LBM	MCS[1]	MCS[0]	M5PS	MLK	MLE	MHRE	
		MRS	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		MDA[8]	MDA[7]	MDA[6]	MDA[5]	MDA[4]	MDA[3]	MDA[2]	MDA[1]	
	SSCR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		SSRE	SDMU	SDML	SDSC	SDCS	SDNU	SDNL	SDR	
	SDCR	MSD[31]	MSD[30]	MSD[29]	MSD[28]	MSD[27]	MSD[26]	MSD[25]	MSD[24]	
		MSD[23]	MSD[22]	MSD[21]	MSD[20]	MSD[19]	MSD[18]	MSD[17]	MSD[16]	
		MSD[15]	MSD[14]	MSD[13]	MSD[12]	MSD[11]	MSD[10]	MSD[9]	MSD[8]	
		MSD[7]	MSD[6]	MSD[5]	MSD[4]	MSD[3]	MSD[2]	MSD[1]	MSD[0]	
	SMCR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	SMMU	SMML	SMSC	SMCS	SMNU	SMNL	SMR	
	VCCR	UMA[7]	UMA[6]	UMA[5]	UMA[4]	UMA[3]	UMA[2]	UMA[1]	UMA[0]	
		UMI[7]	UMI[6]	UMI[5]	UMI[4]	UMI[3]	UMI[2]	UMI[1]	UMI[0]	
		MMA[7]	MMA[6]	MMA[5]	MMA[4]	MMA[3]	MMA[2]	MMA[1]	MMA[0]	
		MMI[7]	MMI[6]	MMI[5]	MMI[4]	MMI[3]	MMI[2]	MMI[1]	MMI[0]	
	SBCR	SRBA[31]	SRBA[30]	SRBA[29]	SRBA[28]	SRBA[27]	SRBA[26]	SRBA[25]	SRBA[24]	
		SRBA[23]	SRBA[22]	SRBA[21]	SRBA[20]	SRBA[19]	SRBA[18]	SRBA[17]	SRBA[16]	
		STBA[31]	STBA[30]	STBA[29]	STBA[28]	STBA[27]	STBA[26]	STBA[25]	STBA[24]	
		STBA[23]	STBA[22]	STBA[21]	STBA[20]	STBA[19]	STBA[18]	STBA[17]	STBA[16]	
	ABCR	ARBA[31]	ARBA[30]	ARBA[29]	ARBA[28]	ARBA[27]	ARBA[26]	ARBA[25]	ARBA[24]	
		ARBA[23]	ARBA[22]	ARBA[21]	ARBA[20]	ARBA[19]	ARBA[18]	ARBA[17]	ARBA[16]	
		ATBA[31]	ATBA[30]	ATBA[29]	ATBA[28]	ATBA[27]	ATBA[26]	ATBA[25]	ATBA[24]	
		ATBA[23]	ATBA[22]	ATBA[21]	ATBA[20]	ATBA[19]	ATBA[18]	ATBA[17]	ATBA[16]	
	CBCR	CRBA[31]	CRBA[30]	CRBA[29]	CRBA[28]	CRBA[27]	CRBA[26]	CRBA[25]	CRBA[24]	
		CRBA[23]	CRBA[22]	CRBA[21]	CRBA[20]	CRBA[19]	CRBA[18]	CRBA[17]	CRBA[16]	
		CTBA[31]	CTBA[30]	CTBA[29]	CTBA[28]	CTBA[27]	CTBA[26]	CTBA[25]	CTBA[24]	
		CTBA[23]	CTBA[22]	CTBA[21]	CTBA[20]	CTBA[19]	CTBA[18]	CTBA[17]	CTBA[16]	
	IBCR	IRBA[31]	IRBA[30]	IRBA[29]	IRBA[28]	IRBA[27]	IRBA[26]	IRBA[25]	IRBA[24]	
		IRBA[23]	IRBA[22]	IRBA[21]	IRBA[20]	IRBA[19]	IRBA[18]	IRBA[17]	IRBA[16]	
		ITBA[31]	ITBA[30]	ITBA[29]	ITBA[28]	ITBA[27]	ITBA[26]	ITBA[25]	ITBA[24]	
		ITBA[23]	ITBA[22]	ITBA[21]	ITBA[20]	ITBA[19]	ITBA[18]	ITBA[17]	ITBA[16]	
	CICR	-	CnSU[30]	CnSU[29]	CnSU[28]	CnSU[27]	CnSU[26]	CnSU[25]	CnSU[24]	
		CnSU[23]	CnSU[22]	CnSU[21]	CnSU[20]	CnSU[19]	CnSU[18]	CnSU[17]	CnSU[16]	
		CnSU[15]	CnSU[14]	CnSU[13]	CnSU[12]	CnSU[11]	CnSU[10]	CnSU[9]	CnSU[8]	
		CnSU[7]	CnSU[6]	CnSU[5]	CnSU[4]	CnSU[3]	CnSU[2]	CnSU[1]	CnSU[0]	
		-	-	-	-	-	-	-	-	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Media local bus	CECR0	CE	TR	CT[1]	CT[0]	PCE_FSE	MDS[1]	MDS[0]	-	
		MASK[7]	MASK[6]	MASK[5]	MASK[4]	MASK[3]	MASK[2]	MASK[1]	MASK[0]	
		FSCD	-	-	PCTH_FSPC[4]	PCTH_FSPC[3]	PCTH_FSPC[2]	PCTH_FSPC[1]	PCTH_FSPC[0]	
		CA[8]	CA[7]	CA[6]	CA[5]	CA[4]	CA[3]	CA[2]	CA[1]	
	CSCR0	BM	BF	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GB	RDY
		STS[15]	STS[14]	STS[13]	STS[12]	STS[11]	STS[10]	STS[9]	STS[8]	STS[7]
		STS[7]	STS[6]	STS[5]	STS[4]	STS[3]	STS[2]	STS[1]	STS[0]	-
	CCBCR0	BCA[15]	BCA[14]	BCA[13]	BCA[12]	BCA[11]	BCA[10]	BCA[9]	BCA[8]	BCA[7]
		BCA[7]	BCA[6]	BCA[5]	BCA[4]	BCA[3]	BCA[2]	BCA[1]	BCA[0]	-
		BFA[15]	BFA[14]	BFA[13]	BFA[12]	BFA[11]	BFA[10]	BFA[9]	BFA[8]	BFA[7]
		BFA[7]	BFA[6]	BFA[5]	BFA[4]	BFA[3]	BFA[2]	BFA[1]	BFA[0]	-
	CNBCR0	BSA[15]	BSA[14]	BSA[13]	BSA[12]	BSA[11]	BSA[10]	BSA[9]	BSA[8]	BSA[7]
		BSA[7]	BSA[6]	BSA[5]	BSA[4]	BSA[3]	BSA[2]	-	-	-
		BEA[15]	BEA[14]	BEA[13]	BEA[12]	BEA[11]	BEA[10]	BEA[9]	BEA[8]	BEA[7]
		BEA[7]	BEA[6]	BEA[5]	BEA[4]	BEA[3]	BEA[2]	BEA[1]	BEA[0]	-
	CECR1	CE	TR	CT[1]	CT[0]	PCE_FSE	MDS[1]	MDS[0]	-	-
		MASK[7]	MASK[6]	MASK[5]	MASK[4]	MASK[3]	MASK[2]	MASK[1]	MASK[0]	-
		FSCD	-	-	PCTH_FSPC[4]	PCTH_FSPC[3]	PCTH_FSPC[2]	PCTH_FSPC[1]	PCTH_FSPC[0]	-
		CA[8]	CA[7]	CA[6]	CA[5]	CA[4]	CA[3]	CA[2]	CA[1]	-
	CSCR1	BM	BF	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GB	RDY
		STS[15]	STS[14]	STS[13]	STS[12]	STS[11]	STS[10]	STS[9]	STS[8]	STS[7]
		STS[7]	STS[6]	STS[5]	STS[4]	STS[3]	STS[2]	STS[1]	STS[0]	-
	CCBCR1	BCA[15]	BCA[14]	BCA[13]	BCA[12]	BCA[11]	BCA[10]	BCA[9]	BCA[8]	BCA[7]
		BCA[7]	BCA[6]	BCA[5]	BCA[4]	BCA[3]	BCA[2]	BCA[1]	BCA[0]	-
		BFA[15]	BFA[14]	BFA[13]	BFA[12]	BFA[11]	BFA[10]	BFA[9]	BFA[8]	BFA[7]
		BFA[7]	BFA[6]	BFA[5]	BFA[4]	BFA[3]	BFA[2]	BFA[1]	BFA[0]	-
	CNBCR1	BSA[15]	BSA[14]	BSA[13]	BSA[12]	BSA[11]	BSA[10]	BSA[9]	BSA[8]	BSA[7]
		BSA[7]	BSA[6]	BSA[5]	BSA[4]	BSA[3]	BSA[2]	-	-	-
		BEA[15]	BEA[14]	BEA[13]	BEA[12]	BEA[11]	BEA[10]	BEA[9]	BEA[8]	BEA[7]
		BEA[7]	BEA[6]	BEA[5]	BEA[4]	BEA[3]	BEA[2]	BEA[1]	BEA[0]	-
	CECR2	CE	TR	CT[1]	CT[0]	PCE_FSE	MDS[1]	MDS[0]	-	-
		MASK[7]	MASK[6]	MASK[5]	MASK[4]	MASK[3]	MASK[2]	MASK[1]	MASK[0]	-
		FSCD	-	-	PCTH_FSPC[4]	PCTH_FSPC[3]	PCTH_FSPC[2]	PCTH_FSPC[1]	PCTH_FSPC[0]	-
		CA[8]	CA[7]	CA[6]	CA[5]	CA[4]	CA[3]	CA[2]	CA[1]	-
	CSCR2	BM	BF	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GB	RDY
		STS[15]	STS[14]	STS[13]	STS[12]	STS[11]	STS[10]	STS[9]	STS[8]	STS[7]
		STS[7]	STS[6]	STS[5]	STS[4]	STS[3]	STS[2]	STS[1]	STS[0]	-
	CCBCR2	BCA[15]	BCA[14]	BCA[13]	BCA[12]	BCA[11]	BCA[10]	BCA[9]	BCA[8]	BCA[7]
		BCA[7]	BCA[6]	BCA[5]	BCA[4]	BCA[3]	BCA[2]	BCA[1]	BCA[0]	-
		BFA[15]	BFA[14]	BFA[13]	BFA[12]	BFA[11]	BFA[10]	BFA[9]	BFA[8]	BFA[7]
		BFA[7]	BFA[6]	BFA[5]	BFA[4]	BFA[3]	BFA[2]	BFA[1]	BFA[0]	-
	CNBCR2	BSA[15]	BSA[14]	BSA[13]	BSA[12]	BSA[11]	BSA[10]	BSA[9]	BSA[8]	BSA[7]
		BSA[7]	BSA[6]	BSA[5]	BSA[4]	BSA[3]	BSA[2]	-	-	-
		BEA[15]	BEA[14]	BEA[13]	BEA[12]	BEA[11]	BEA[10]	BEA[9]	BEA[8]	BEA[7]
		BEA[7]	BEA[6]	BEA[5]	BEA[4]	BEA[3]	BEA[2]	BEA[1]	BEA[0]	-
	CECR3	CE	TR	CT[1]	CT[0]	PCE_FSE	MDS[1]	MDS[0]	-	-
		MASK[7]	MASK[6]	MASK[5]	MASK[4]	MASK[3]	MASK[2]	MASK[1]	MASK[0]	-
		FSCD	-	-	PCTH_FSPC[4]	PCTH_FSPC[3]	PCTH_FSPC[2]	PCTH_FSPC[1]	PCTH_FSPC[0]	-
		CA[8]	CA[7]	CA[6]	CA[5]	CA[4]	CA[3]	CA[2]	CA[1]	-
	CSCR3	BM	BF	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GB	RDY
		STS[15]	STS[14]	STS[13]	STS[12]	STS[11]	STS[10]	STS[9]	STS[8]	STS[7]
		STS[7]	STS[6]	STS[5]	STS[4]	STS[3]	STS[2]	STS[1]	STS[0]	-

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Media local bus	CCBCR3	BCA[15]	BCA[14]	BCA[13]	BCA[12]	BCA[11]	BCA[10]	BCA[9]	BCA[8]
		BCA[7]	BCA[6]	BCA[5]	BCA[4]	BCA[3]	BCA[2]	BCA[1]	BCA[0]
		BFA[15]	BFA[14]	BFA[13]	BFA[12]	BFA[11]	BFA[10]	BFA[9]	BFA[8]
	CNBCR3	BSA[15]	BSA[14]	BSA[13]	BSA[12]	BSA[11]	BSA[10]	BSA[9]	BSA[8]
		BSA[7]	BSA[6]	BSA[5]	BSA[4]	BSA[3]	BSA[2]	-	-
		BEA[15]	BEA[14]	BEA[13]	BEA[12]	BEA[11]	BEA[10]	BEA[9]	BEA[8]
	CECR4	CE	TR	CT[1]	CT[0]	PCE_FSE	MDS[1]	MDS[0]	-
		MASK[7]	MASK[6]	MASK[5]	MASK[4]	MASK[3]	MASK[2]	MASK[1]	MASK[0]
		FSCD	-	-	PCTH_FSPC[4]	PCTH_FSPC[3]	PCTH_FSPC[2]	PCTH_FSPC[1]	PCTH_FSPC[0]
	CSCR4	CA[8]	CA[7]	CA[6]	CA[5]	CA[4]	CA[3]	CA[2]	CA[1]
		BM	BF	-	-	-	-	-	-
		-	-	-	-	-	-	GB	RDY
	CCBCR4	STS[15]	STS[14]	STS[13]	STS[12]	STS[11]	STS[10]	STS[9]	STS[8]
		STS[7]	STS[6]	STS[5]	STS[4]	STS[3]	STS[2]	STS[1]	STS[0]
		BCA[15]	BCA[14]	BCA[13]	BCA[12]	BCA[11]	BCA[10]	BCA[9]	BCA[8]
	CNBCR4	BCA[7]	BCA[6]	BCA[5]	BCA[4]	BCA[3]	BCA[2]	BCA[1]	BCA[0]
		BFA[15]	BFA[14]	BFA[13]	BFA[12]	BFA[11]	BFA[10]	BFA[9]	BFA[8]
		BFA[7]	BFA[6]	BFA[5]	BFA[4]	BFA[3]	BFA[2]	BFA[1]	BFA[0]
	CECR5	BSA[15]	BSA[14]	BSA[13]	BSA[12]	BSA[11]	BSA[10]	BSA[9]	BSA[8]
		BSA[7]	BSA[6]	BSA[5]	BSA[4]	BSA[3]	BSA[2]	-	-
		BEA[15]	BEA[14]	BEA[13]	BEA[12]	BEA[11]	BEA[10]	BEA[9]	BEA[8]
	CSCR5	BEA[7]	BEA[6]	BEA[5]	BEA[4]	BEA[3]	BEA[2]	BEA[1]	BEA[0]
		CE	TR	CT[1]	CT[0]	PCE_FSE	MDS[1]	MDS[0]	-
		MASK[7]	MASK[6]	MASK[5]	MASK[4]	MASK[3]	MASK[2]	MASK[1]	MASK[0]
	CCBCR5	FSCD	-	-	PCTH_FSPC[4]	PCTH_FSPC[3]	PCTH_FSPC[2]	PCTH_FSPC[1]	PCTH_FSPC[0]
		CA[8]	CA[7]	CA[6]	CA[5]	CA[4]	CA[3]	CA[2]	CA[1]
		BM	BF	-	-	-	-	-	-
	CNBCR5	-	-	-	-	-	-	GB	RDY
		STS[15]	STS[14]	STS[13]	STS[12]	STS[11]	STS[10]	STS[9]	STS[8]
		STS[7]	STS[6]	STS[5]	STS[4]	STS[3]	STS[2]	STS[1]	STS[0]
	CECR6	BCA[15]	BCA[14]	BCA[13]	BCA[12]	BCA[11]	BCA[10]	BCA[9]	BCA[8]
		BCA[7]	BCA[6]	BCA[5]	BCA[4]	BCA[3]	BCA[2]	BCA[1]	BCA[0]
		BFA[15]	BFA[14]	BFA[13]	BFA[12]	BFA[11]	BFA[10]	BFA[9]	BFA[8]
	CNBCR6	BFA[7]	BFA[6]	BFA[5]	BFA[4]	BFA[3]	BFA[2]	BFA[1]	BFA[0]
		BSA[15]	BSA[14]	BSA[13]	BSA[12]	BSA[11]	BSA[10]	BSA[9]	BSA[8]
		BSA[7]	BSA[6]	BSA[5]	BSA[4]	BSA[3]	BSA[2]	-	-
	CECR6	BEA[15]	BEA[14]	BEA[13]	BEA[12]	BEA[11]	BEA[10]	BEA[9]	BEA[8]
		BEA[7]	BEA[6]	BEA[5]	BEA[4]	BEA[3]	BEA[2]	BEA[1]	BEA[0]
		CE	TR	CT[1]	CT[0]	PCE_FSE	MDS[1]	MDS[0]	-
	CSCR6	MASK[7]	MASK[6]	MASK[5]	MASK[4]	MASK[3]	MASK[2]	MASK[1]	MASK[0]
		FSCD	-	-	PCTH_FSPC[4]	PCTH_FSPC[3]	PCTH_FSPC[2]	PCTH_FSPC[1]	PCTH_FSPC[0]
		CA[8]	CA[7]	CA[6]	CA[5]	CA[4]	CA[3]	CA[2]	CA[1]
	CCBCR6	BM	BF	-	-	-	-	-	-
		-	-	-	-	-	-	GB	RDY
		STS[15]	STS[14]	STS[13]	STS[12]	STS[11]	STS[10]	STS[9]	STS[8]
	CNBCR6	STS[7]	STS[6]	STS[5]	STS[4]	STS[3]	STS[2]	STS[1]	STS[0]
		BCA[15]	BCA[14]	BCA[13]	BCA[12]	BCA[11]	BCA[10]	BCA[9]	BCA[8]
		BCA[7]	BCA[6]	BCA[5]	BCA[4]	BCA[3]	BCA[2]	BCA[1]	BCA[0]
	CECR6	BFA[15]	BFA[14]	BFA[13]	BFA[12]	BFA[11]	BFA[10]	BFA[9]	BFA[8]
		BFA[7]	BFA[6]	BFA[5]	BFA[4]	BFA[3]	BFA[2]	BFA[1]	BFA[0]
		BSA[15]	BSA[14]	BSA[13]	BSA[12]	BSA[11]	BSA[10]	BSA[9]	BSA[8]
	CNBCR6	BSA[7]	BSA[6]	BSA[5]	BSA[4]	BSA[3]	BSA[2]	-	-
		BEA[15]	BEA[14]	BEA[13]	BEA[12]	BEA[11]	BEA[10]	BEA[9]	BEA[8]
		BEA[7]	BEA[6]	BEA[5]	BEA[4]	BEA[3]	BEA[2]	BEA[1]	BEA[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Media local bus	CECR7	CE	TR	CT[1]	CT[0]	PCE_FSE	MDS[1]	MDS[0]	-	
		MASK[7]	MASK[6]	MASK[5]	MASK[4]	MASK[3]	MASK[2]	MASK[1]	MASK[0]	
		FSCD	-	-	PCTH_FSPC[4]	PCTH_FSPC[3]	PCTH_FSPC[2]	PCTH_FSPC[1]	PCTH_FSPC[0]	
		CA[8]	CA[7]	CA[6]	CA[5]	CA[4]	CA[3]	CA[2]	CA[1]	
	CSCR7	BM	BF	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GB	RDY
		STS[15]	STS[14]	STS[13]	STS[12]	STS[11]	STS[10]	STS[9]	STS[8]	STS[7]
		STS[7]	STS[6]	STS[5]	STS[4]	STS[3]	STS[2]	STS[1]	STS[0]	
	CCBCR7	BCA[15]	BCA[14]	BCA[13]	BCA[12]	BCA[11]	BCA[10]	BCA[9]	BCA[8]	BCA[7]
		BCA[7]	BCA[6]	BCA[5]	BCA[4]	BCA[3]	BCA[2]	BCA[1]	BCA[0]	
		BFA[15]	BFA[14]	BFA[13]	BFA[12]	BFA[11]	BFA[10]	BFA[9]	BFA[8]	BFA[7]
		BFA[7]	BFA[6]	BFA[5]	BFA[4]	BFA[3]	BFA[2]	BFA[1]	BFA[0]	
	CNBCR7	BSA[15]	BSA[14]	BSA[13]	BSA[12]	BSA[11]	BSA[10]	BSA[9]	BSA[8]	BSA[7]
		BSA[7]	BSA[6]	BSA[5]	BSA[4]	BSA[3]	BSA[2]	-	-	
		BEA[15]	BEA[14]	BEA[13]	BEA[12]	BEA[11]	BEA[10]	BEA[9]	BEA[8]	BEA[7]
		BEA[7]	BEA[6]	BEA[5]	BEA[4]	BEA[3]	BEA[2]	BEA[1]	BEA[0]	
	CECR8	CE	TR	CT[1]	CT[0]	PCE_FSE	MDS[1]	MDS[0]	-	
		MASK[7]	MASK[6]	MASK[5]	MASK[4]	MASK[3]	MASK[2]	MASK[1]	MASK[0]	
		FSCD	-	-	PCTH_FSPC[4]	PCTH_FSPC[3]	PCTH_FSPC[2]	PCTH_FSPC[1]	PCTH_FSPC[0]	
		CA[8]	CA[7]	CA[6]	CA[5]	CA[4]	CA[3]	CA[2]	CA[1]	
	CSCR8	BM	BF	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GB	RDY
		STS[15]	STS[14]	STS[13]	STS[12]	STS[11]	STS[10]	STS[9]	STS[8]	STS[7]
		STS[7]	STS[6]	STS[5]	STS[4]	STS[3]	STS[2]	STS[1]	STS[0]	
	CCBCR8	BCA[15]	BCA[14]	BCA[13]	BCA[12]	BCA[11]	BCA[10]	BCA[9]	BCA[8]	BCA[7]
		BCA[7]	BCA[6]	BCA[5]	BCA[4]	BCA[3]	BCA[2]	BCA[1]	BCA[0]	
		BFA[15]	BFA[14]	BFA[13]	BFA[12]	BFA[11]	BFA[10]	BFA[9]	BFA[8]	BFA[7]
		BFA[7]	BFA[6]	BFA[5]	BFA[4]	BFA[3]	BFA[2]	BFA[1]	BFA[0]	
	CNBCR8	BSA[15]	BSA[14]	BSA[13]	BSA[12]	BSA[11]	BSA[10]	BSA[9]	BSA[8]	BSA[7]
		BSA[7]	BSA[6]	BSA[5]	BSA[4]	BSA[3]	BSA[2]	-	-	
		BEA[15]	BEA[14]	BEA[13]	BEA[12]	BEA[11]	BEA[10]	BEA[9]	BEA[8]	BEA[7]
		BEA[7]	BEA[6]	BEA[5]	BEA[4]	BEA[3]	BEA[2]	BEA[1]	BEA[0]	
	CECR9	CE	TR	CT[1]	CT[0]	PCE_FSE	MDS[1]	MDS[0]	-	
		MASK[7]	MASK[6]	MASK[5]	MASK[4]	MASK[3]	MASK[2]	MASK[1]	MASK[0]	
		FSCD	-	-	PCTH_FSPC[4]	PCTH_FSPC[3]	PCTH_FSPC[2]	PCTH_FSPC[1]	PCTH_FSPC[0]	
		CA[8]	CA[7]	CA[6]	CA[5]	CA[4]	CA[3]	CA[2]	CA[1]	
	CSCR9	BM	BF	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GB	RDY
		STS[15]	STS[14]	STS[13]	STS[12]	STS[11]	STS[10]	STS[9]	STS[8]	STS[7]
		STS[7]	STS[6]	STS[5]	STS[4]	STS[3]	STS[2]	STS[1]	STS[0]	
	CCBCR9	BCA[15]	BCA[14]	BCA[13]	BCA[12]	BCA[11]	BCA[10]	BCA[9]	BCA[8]	BCA[7]
		BCA[7]	BCA[6]	BCA[5]	BCA[4]	BCA[3]	BCA[2]	BCA[1]	BCA[0]	
		BFA[15]	BFA[14]	BFA[13]	BFA[12]	BFA[11]	BFA[10]	BFA[9]	BFA[8]	BFA[7]
		BFA[7]	BFA[6]	BFA[5]	BFA[4]	BFA[3]	BFA[2]	BFA[1]	BFA[0]	
CNBCR9	BSA[15]	BSA[14]	BSA[13]	BSA[12]	BSA[11]	BSA[10]	BSA[9]	BSA[8]	BSA[7]	
	BSA[7]	BSA[6]	BSA[5]	BSA[4]	BSA[3]	BSA[2]	-	-		
	BEA[15]	BEA[14]	BEA[13]	BEA[12]	BEA[11]	BEA[10]	BEA[9]	BEA[8]	BEA[7]	
	BEA[7]	BEA[6]	BEA[5]	BEA[4]	BEA[3]	BEA[2]	BEA[1]	BEA[0]		
CECR10	CE	TR	CT[1]	CT[0]	PCE_FSE	MDS[1]	MDS[0]	-		
	MASK[7]	MASK[6]	MASK[5]	MASK[4]	MASK[3]	MASK[2]	MASK[1]	MASK[0]		
	FSCD	-	-	PCTH_FSPC[4]	PCTH_FSPC[3]	PCTH_FSPC[2]	PCTH_FSPC[1]	PCTH_FSPC[0]		
	CA[8]	CA[7]	CA[6]	CA[5]	CA[4]	CA[3]	CA[2]	CA[1]		
CSCR10	BM	BF	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	GB	RDY	
	STS[15]	STS[14]	STS[13]	STS[12]	STS[11]	STS[10]	STS[9]	STS[8]	STS[7]	
	STS[7]	STS[6]	STS[5]	STS[4]	STS[3]	STS[2]	STS[1]	STS[0]		

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Media local bus	CCBCR10	BCA[15]	BCA[14]	BCA[13]	BCA[12]	BCA[11]	BCA[10]	BCA[9]	BCA[8]
		BCA[7]	BCA[6]	BCA[5]	BCA[4]	BCA[3]	BCA[2]	BCA[1]	BCA[0]
		BFA[15]	BFA[14]	BFA[13]	BFA[12]	BFA[11]	BFA[10]	BFA[9]	BFA[8]
	CNBCR10	BFA[7]	BFA[6]	BFA[5]	BFA[4]	BFA[3]	BFA[2]	BFA[1]	BFA[0]
		BSA[15]	BSA[14]	BSA[13]	BSA[12]	BSA[11]	BSA[10]	BSA[9]	BSA[8]
		BSA[7]	BSA[6]	BSA[5]	BSA[4]	BSA[3]	BSA[2]	-	-
		BEA[15]	BEA[14]	BEA[13]	BEA[12]	BEA[11]	BEA[10]	BEA[9]	BEA[8]
	CECR11	BEA[7]	BEA[6]	BEA[5]	BEA[4]	BEA[3]	BEA[2]	BEA[1]	BEA[0]
		CE	TR	CT[1]	CT[0]	PCE_FSE	MDS[1]	MDS[0]	-
		MASK[7]	MASK[6]	MASK[5]	MASK[4]	MASK[3]	MASK[2]	MASK[1]	MASK[0]
		FSCD	-	-	PCTH_FSPC[4]	PCTH_FSPC[3]	PCTH_FSPC[2]	PCTH_FSPC[1]	PCTH_FSPC[0]
	CSCR11	CA[8]	CA[7]	CA[6]	CA[5]	CA[4]	CA[3]	CA[2]	CA[1]
		BM	BF	-	-	-	-	-	-
		-	-	-	-	-	-	GB	RDY
		STS[15]	STS[14]	STS[13]	STS[12]	STS[11]	STS[10]	STS[9]	STS[8]
	CCBCR11	STS[7]	STS[6]	STS[5]	STS[4]	STS[3]	STS[2]	STS[1]	STS[0]
		BCA[15]	BCA[14]	BCA[13]	BCA[12]	BCA[11]	BCA[10]	BCA[9]	BCA[8]
		BCA[7]	BCA[6]	BCA[5]	BCA[4]	BCA[3]	BCA[2]	BCA[1]	BCA[0]
		BFA[15]	BFA[14]	BFA[13]	BFA[12]	BFA[11]	BFA[10]	BFA[9]	BFA[8]
	CNBCR11	BFA[7]	BFA[6]	BFA[5]	BFA[4]	BFA[3]	BFA[2]	BFA[1]	BFA[0]
		BSA[15]	BSA[14]	BSA[13]	BSA[12]	BSA[11]	BSA[10]	BSA[9]	BSA[8]
		BSA[7]	BSA[6]	BSA[5]	BSA[4]	BSA[3]	BSA[2]	-	-
		BEA[15]	BEA[14]	BEA[13]	BEA[12]	BEA[11]	BEA[10]	BEA[9]	BEA[8]
	CECR12	BEA[7]	BEA[6]	BEA[5]	BEA[4]	BEA[3]	BEA[2]	BEA[1]	BEA[0]
		CE	TR	CT[1]	CT[0]	PCE_FSE	MDS[1]	MDS[0]	-
		MASK[7]	MASK[6]	MASK[5]	MASK[4]	MASK[3]	MASK[2]	MASK[1]	MASK[0]
		FSCD	-	-	PCTH_FSPC[4]	PCTH_FSPC[3]	PCTH_FSPC[2]	PCTH_FSPC[1]	PCTH_FSPC[0]
	CSCR12	CA[8]	CA[7]	CA[6]	CA[5]	CA[4]	CA[3]	CA[2]	CA[1]
		BM	BF	-	-	-	-	-	-
		-	-	-	-	-	-	GB	RDY
		STS[15]	STS[14]	STS[13]	STS[12]	STS[11]	STS[10]	STS[9]	STS[8]
	CCBCR12	STS[7]	STS[6]	STS[5]	STS[4]	STS[3]	STS[2]	STS[1]	STS[0]
		BCA[15]	BCA[14]	BCA[13]	BCA[12]	BCA[11]	BCA[10]	BCA[9]	BCA[8]
		BCA[7]	BCA[6]	BCA[5]	BCA[4]	BCA[3]	BCA[2]	BCA[1]	BCA[0]
		BFA[15]	BFA[14]	BFA[13]	BFA[12]	BFA[11]	BFA[10]	BFA[9]	BFA[8]
	CNBCR12	BFA[7]	BFA[6]	BFA[5]	BFA[4]	BFA[3]	BFA[2]	BFA[1]	BFA[0]
		BSA[15]	BSA[14]	BSA[13]	BSA[12]	BSA[11]	BSA[10]	BSA[9]	BSA[8]
		BSA[7]	BSA[6]	BSA[5]	BSA[4]	BSA[3]	BSA[2]	-	-
		BEA[15]	BEA[14]	BEA[13]	BEA[12]	BEA[11]	BEA[10]	BEA[9]	BEA[8]
	CECR13	BEA[7]	BEA[6]	BEA[5]	BEA[4]	BEA[3]	BEA[2]	BEA[1]	BEA[0]
		CE	TR	CT[1]	CT[0]	PCE_FSE	MDS[1]	MDS[0]	-
		MASK[7]	MASK[6]	MASK[5]	MASK[4]	MASK[3]	MASK[2]	MASK[1]	MASK[0]
		FSCD	-	-	PCTH_FSPC[4]	PCTH_FSPC[3]	PCTH_FSPC[2]	PCTH_FSPC[1]	PCTH_FSPC[0]
	CSCR13	CA[8]	CA[7]	CA[6]	CA[5]	CA[4]	CA[3]	CA[2]	CA[1]
		BM	BF	-	-	-	-	-	-
		-	-	-	-	-	-	GB	RDY
		STS[15]	STS[14]	STS[13]	STS[12]	STS[11]	STS[10]	STS[9]	STS[8]
	CCBCR13	STS[7]	STS[6]	STS[5]	STS[4]	STS[3]	STS[2]	STS[1]	STS[0]
		BCA[15]	BCA[14]	BCA[13]	BCA[12]	BCA[11]	BCA[10]	BCA[9]	BCA[8]
		BCA[7]	BCA[6]	BCA[5]	BCA[4]	BCA[3]	BCA[2]	BCA[1]	BCA[0]
		BFA[15]	BFA[14]	BFA[13]	BFA[12]	BFA[11]	BFA[10]	BFA[9]	BFA[8]
	CNBCR13	BFA[7]	BFA[6]	BFA[5]	BFA[4]	BFA[3]	BFA[2]	BFA[1]	BFA[0]
BSA[15]		BSA[14]	BSA[13]	BSA[12]	BSA[11]	BSA[10]	BSA[9]	BSA[8]	
BSA[7]		BSA[6]	BSA[5]	BSA[4]	BSA[3]	BSA[2]	-	-	
BEA[15]		BEA[14]	BEA[13]	BEA[12]	BEA[11]	BEA[10]	BEA[9]	BEA[8]	
		BEA[7]	BEA[6]	BEA[5]	BEA[4]	BEA[3]	BEA[2]	BEA[1]	BEA[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Media local bus	CECR14	CE	TR	CT[1]	CT[0]	PCE_FSE	MDS[1]	MDS[0]	-
		MASK[7]	MASK[6]	MASK[5]	MASK[4]	MASK[3]	MASK[2]	MASK[1]	MASK[0]
		FSCD	-	-	PCTH_FSPC[4]	PCTH_FSPC[3]	PCTH_FSPC[2]	PCTH_FSPC[1]	PCTH_FSPC[0]
		CA[8]	CA[7]	CA[6]	CA[5]	CA[4]	CA[3]	CA[2]	CA[1]
	CSCR14	BM	BF	-	-	-	-	-	-
		-	-	-	-	-	-	GB	RDY
		STS[15]	STS[14]	STS[13]	STS[12]	STS[11]	STS[10]	STS[9]	STS[8]
		STS[7]	STS[6]	STS[5]	STS[4]	STS[3]	STS[2]	STS[1]	STS[0]
	CCBCR14	BCA[15]	BCA[14]	BCA[13]	BCA[12]	BCA[11]	BCA[10]	BCA[9]	BCA[8]
		BCA[7]	BCA[6]	BCA[5]	BCA[4]	BCA[3]	BCA[2]	BCA[1]	BCA[0]
		BFA[15]	BFA[14]	BFA[13]	BFA[12]	BFA[11]	BFA[10]	BFA[9]	BFA[8]
		BFA[7]	BFA[6]	BFA[5]	BFA[4]	BFA[3]	BFA[2]	BFA[1]	BFA[0]
	CNBCR14	BSA[15]	BSA[14]	BSA[13]	BSA[12]	BSA[11]	BSA[10]	BSA[9]	BSA[8]
		BSA[7]	BSA[6]	BSA[5]	BSA[4]	BSA[3]	BSA[2]	-	-
		BEA[15]	BEA[14]	BEA[13]	BEA[12]	BEA[11]	BEA[10]	BEA[9]	BEA[8]
		BEA[7]	BEA[6]	BEA[5]	BEA[4]	BEA[3]	BEA[2]	BEA[1]	BEA[0]
	CECR15	CE	TR	CT[1]	CT[0]	PCE_FSE	MDS[1]	MDS[0]	-
		MASK[7]	MASK[6]	MASK[5]	MASK[4]	MASK[3]	MASK[2]	MASK[1]	MASK[0]
		FSCD	-	-	PCTH_FSPC[4]	PCTH_FSPC[3]	PCTH_FSPC[2]	PCTH_FSPC[1]	PCTH_FSPC[0]
		CA[8]	CA[7]	CA[6]	CA[5]	CA[4]	CA[3]	CA[2]	CA[1]
	CSCR15	BM	BF	-	-	-	-	-	-
		-	-	-	-	-	-	GB	RDY
		STS[15]	STS[14]	STS[13]	STS[12]	STS[11]	STS[10]	STS[9]	STS[8]
		STS[7]	STS[6]	STS[5]	STS[4]	STS[3]	STS[2]	STS[1]	STS[0]
	CCBCR15	BCA[15]	BCA[14]	BCA[13]	BCA[12]	BCA[11]	BCA[10]	BCA[9]	BCA[8]
		BCA[7]	BCA[6]	BCA[5]	BCA[4]	BCA[3]	BCA[2]	BCA[1]	BCA[0]
		BFA[15]	BFA[14]	BFA[13]	BFA[12]	BFA[11]	BFA[10]	BFA[9]	BFA[8]
		BFA[7]	BFA[6]	BFA[5]	BFA[4]	BFA[3]	BFA[2]	BFA[1]	BFA[0]
	CNBCR15	BSA[15]	BSA[14]	BSA[13]	BSA[12]	BSA[11]	BSA[10]	BSA[9]	BSA[8]
		BSA[7]	BSA[6]	BSA[5]	BSA[4]	BSA[3]	BSA[2]	-	-
		BEA[15]	BEA[14]	BEA[13]	BEA[12]	BEA[11]	BEA[10]	BEA[9]	BEA[8]
		BEA[7]	BEA[6]	BEA[5]	BEA[4]	BEA[3]	BEA[2]	BEA[1]	BEA[0]
	CECR16	CE	TR	CT[1]	CT[0]	PCE_FSE	MDS[1]	MDS[0]	-
		MASK[7]	MASK[6]	MASK[5]	MASK[4]	MASK[3]	MASK[2]	MASK[1]	MASK[0]
		FSCD	-	-	PCTH_FSPC[4]	PCTH_FSPC[3]	PCTH_FSPC[2]	PCTH_FSPC[1]	PCTH_FSPC[0]
		CA[8]	CA[7]	CA[6]	CA[5]	CA[4]	CA[3]	CA[2]	CA[1]
	CSCR16	BM	BF	-	-	-	-	-	-
		-	-	-	-	-	-	GB	RDY
		STS[15]	STS[14]	STS[13]	STS[12]	STS[11]	STS[10]	STS[9]	STS[8]
		STS[7]	STS[6]	STS[5]	STS[4]	STS[3]	STS[2]	STS[1]	STS[0]
	CCBCR16	BCA[15]	BCA[14]	BCA[13]	BCA[12]	BCA[11]	BCA[10]	BCA[9]	BCA[8]
		BCA[7]	BCA[6]	BCA[5]	BCA[4]	BCA[3]	BCA[2]	BCA[1]	BCA[0]
		BFA[15]	BFA[14]	BFA[13]	BFA[12]	BFA[11]	BFA[10]	BFA[9]	BFA[8]
		BFA[7]	BFA[6]	BFA[5]	BFA[4]	BFA[3]	BFA[2]	BFA[1]	BFA[0]
	CNBCR16	BSA[15]	BSA[14]	BSA[13]	BSA[12]	BSA[11]	BSA[10]	BSA[9]	BSA[8]
		BSA[7]	BSA[6]	BSA[5]	BSA[4]	BSA[3]	BSA[2]	-	-
		BEA[15]	BEA[14]	BEA[13]	BEA[12]	BEA[11]	BEA[10]	BEA[9]	BEA[8]
		BEA[7]	BEA[6]	BEA[5]	BEA[4]	BEA[3]	BEA[2]	BEA[1]	BEA[0]
	CECR17	CE	TR	CT[1]	CT[0]	PCE_FSE	MDS[1]	MDS[0]	-
		MASK[7]	MASK[6]	MASK[5]	MASK[4]	MASK[3]	MASK[2]	MASK[1]	MASK[0]
		FSCD	-	-	PCTH_FSPC[4]	PCTH_FSPC[3]	PCTH_FSPC[2]	PCTH_FSPC[1]	PCTH_FSPC[0]
		CA[8]	CA[7]	CA[6]	CA[5]	CA[4]	CA[3]	CA[2]	CA[1]
	CSCR17	BM	BF	-	-	-	-	-	-
		-	-	-	-	-	-	GB	RDY
		STS[15]	STS[14]	STS[13]	STS[12]	STS[11]	STS[10]	STS[9]	STS[8]
		STS[7]	STS[6]	STS[5]	STS[4]	STS[3]	STS[2]	STS[1]	STS[0]



Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Media local bus	CCBCR17	BCA[15]	BCA[14]	BCA[13]	BCA[12]	BCA[11]	BCA[10]	BCA[9]	BCA[8]
		BCA[7]	BCA[6]	BCA[5]	BCA[4]	BCA[3]	BCA[2]	BCA[1]	BCA[0]
		BFA[15]	BFA[14]	BFA[13]	BFA[12]	BFA[11]	BFA[10]	BFA[9]	BFA[8]
	CNBCR17	BFA[7]	BFA[6]	BFA[5]	BFA[4]	BFA[3]	BFA[2]	BFA[1]	BFA[0]
		BSA[15]	BSA[14]	BSA[13]	BSA[12]	BSA[11]	BSA[10]	BSA[9]	BSA[8]
		BSA[7]	BSA[6]	BSA[5]	BSA[4]	BSA[3]	BSA[2]	-	-
	CECR18	BEA[15]	BEA[14]	BEA[13]	BEA[12]	BEA[11]	BEA[10]	BEA[9]	BEA[8]
		BEA[7]	BEA[6]	BEA[5]	BEA[4]	BEA[3]	BEA[2]	BEA[1]	BEA[0]
		CE	TR	CT[1]	CT[0]	PCE_FSE	MDS[1]	MDS[0]	-
	CSCR18	MASK[7]	MASK[6]	MASK[5]	MASK[4]	MASK[3]	MASK[2]	MASK[1]	MASK[0]
		FSCD	-	-	PCTH_FSPC[4]	PCTH_FSPC[3]	PCTH_FSPC[2]	PCTH_FSPC[1]	PCTH_FSPC[0]
		CA[8]	CA[7]	CA[6]	CA[5]	CA[4]	CA[3]	CA[2]	CA[1]
	CCBCR18	BM	BF	-	-	-	-	-	-
		-	-	-	-	-	-	GB	RDY
		STS[15]	STS[14]	STS[13]	STS[12]	STS[11]	STS[10]	STS[9]	STS[8]
	CNBCR18	STS[7]	STS[6]	STS[5]	STS[4]	STS[3]	STS[2]	STS[1]	STS[0]
		BCA[15]	BCA[14]	BCA[13]	BCA[12]	BCA[11]	BCA[10]	BCA[9]	BCA[8]
		BCA[7]	BCA[6]	BCA[5]	BCA[4]	BCA[3]	BCA[2]	BCA[1]	BCA[0]
	CECR19	BFA[15]	BFA[14]	BFA[13]	BFA[12]	BFA[11]	BFA[10]	BFA[9]	BFA[8]
		BFA[7]	BFA[6]	BFA[5]	BFA[4]	BFA[3]	BFA[2]	BFA[1]	BFA[0]
		BSA[15]	BSA[14]	BSA[13]	BSA[12]	BSA[11]	BSA[10]	BSA[9]	BSA[8]
	CSCR19	BSA[7]	BSA[6]	BSA[5]	BSA[4]	BSA[3]	BSA[2]	-	-
		BEA[15]	BEA[14]	BEA[13]	BEA[12]	BEA[11]	BEA[10]	BEA[9]	BEA[8]
		BEA[7]	BEA[6]	BEA[5]	BEA[4]	BEA[3]	BEA[2]	BEA[1]	BEA[0]
	CCBCR19	CE	TR	CT[1]	CT[0]	PCE_FSE	MDS[1]	MDS[0]	-
		MASK[7]	MASK[6]	MASK[5]	MASK[4]	MASK[3]	MASK[2]	MASK[1]	MASK[0]
		FSCD	-	-	PCTH_FSPC[4]	PCTH_FSPC[3]	PCTH_FSPC[2]	PCTH_FSPC[1]	PCTH_FSPC[0]
	CNBCR19	CA[8]	CA[7]	CA[6]	CA[5]	CA[4]	CA[3]	CA[2]	CA[1]
		BM	BF	-	-	-	-	-	-
		-	-	-	-	-	-	GB	RDY
	CECR20	STS[15]	STS[14]	STS[13]	STS[12]	STS[11]	STS[10]	STS[9]	STS[8]
		STS[7]	STS[6]	STS[5]	STS[4]	STS[3]	STS[2]	STS[1]	STS[0]
		BCA[15]	BCA[14]	BCA[13]	BCA[12]	BCA[11]	BCA[10]	BCA[9]	BCA[8]
	CSCR20	BCA[7]	BCA[6]	BCA[5]	BCA[4]	BCA[3]	BCA[2]	BCA[1]	BCA[0]
		BFA[15]	BFA[14]	BFA[13]	BFA[12]	BFA[11]	BFA[10]	BFA[9]	BFA[8]
		BFA[7]	BFA[6]	BFA[5]	BFA[4]	BFA[3]	BFA[2]	BFA[1]	BFA[0]
	CNBCR20	BSA[15]	BSA[14]	BSA[13]	BSA[12]	BSA[11]	BSA[10]	BSA[9]	BSA[8]
		BSA[7]	BSA[6]	BSA[5]	BSA[4]	BSA[3]	BSA[2]	-	-
		BEA[15]	BEA[14]	BEA[13]	BEA[12]	BEA[11]	BEA[10]	BEA[9]	BEA[8]
	CCBCR20	BEA[7]	BEA[6]	BEA[5]	BEA[4]	BEA[3]	BEA[2]	BEA[1]	BEA[0]
		CE	TR	CT[1]	CT[0]	PCE_FSE	MDS[1]	MDS[0]	-
		MASK[7]	MASK[6]	MASK[5]	MASK[4]	MASK[3]	MASK[2]	MASK[1]	MASK[0]
	CSCR20	FSCD	-	-	PCTH_FSPC[4]	PCTH_FSPC[3]	PCTH_FSPC[2]	PCTH_FSPC[1]	PCTH_FSPC[0]
		CA[8]	CA[7]	CA[6]	CA[5]	CA[4]	CA[3]	CA[2]	CA[1]
		BM	BF	-	-	-	-	-	-
	CNBCR20	-	-	-	-	-	-	GB	RDY
		STS[15]	STS[14]	STS[13]	STS[12]	STS[11]	STS[10]	STS[9]	STS[8]
		STS[7]	STS[6]	STS[5]	STS[4]	STS[3]	STS[2]	STS[1]	STS[0]
	CCBCR20	BCA[15]	BCA[14]	BCA[13]	BCA[12]	BCA[11]	BCA[10]	BCA[9]	BCA[8]
		BCA[7]	BCA[6]	BCA[5]	BCA[4]	BCA[3]	BCA[2]	BCA[1]	BCA[0]
		BFA[15]	BFA[14]	BFA[13]	BFA[12]	BFA[11]	BFA[10]	BFA[9]	BFA[8]
	CNBCR20	BFA[7]	BFA[6]	BFA[5]	BFA[4]	BFA[3]	BFA[2]	BFA[1]	BFA[0]
		BSA[15]	BSA[14]	BSA[13]	BSA[12]	BSA[11]	BSA[10]	BSA[9]	BSA[8]
		BSA[7]	BSA[6]	BSA[5]	BSA[4]	BSA[3]	BSA[2]	-	-
	CCBCR20	BEA[15]	BEA[14]	BEA[13]	BEA[12]	BEA[11]	BEA[10]	BEA[9]	BEA[8]
		BEA[7]	BEA[6]	BEA[5]	BEA[4]	BEA[3]	BEA[2]	BEA[1]	BEA[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Media local bus	CECR21	CE	TR	CT[1]	CT[0]	PCE_FSE	MDS[1]	MDS[0]	-	
		MASK[7]	MASK[6]	MASK[5]	MASK[4]	MASK[3]	MASK[2]	MASK[1]	MASK[0]	
		FSCD	-	-	PCTH_FSPC[4]	PCTH_FSPC[3]	PCTH_FSPC[2]	PCTH_FSPC[1]	PCTH_FSPC[0]	
		CA[8]	CA[7]	CA[6]	CA[5]	CA[4]	CA[3]	CA[2]	CA[1]	
	CSCR21	BM	BF	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GB	RDY
		STS[15]	STS[14]	STS[13]	STS[12]	STS[11]	STS[10]	STS[9]	STS[8]	STS[7]
		STS[7]	STS[6]	STS[5]	STS[4]	STS[3]	STS[2]	STS[1]	STS[0]	
	CCBCR21	BCA[15]	BCA[14]	BCA[13]	BCA[12]	BCA[11]	BCA[10]	BCA[9]	BCA[8]	BCA[7]
		BCA[7]	BCA[6]	BCA[5]	BCA[4]	BCA[3]	BCA[2]	BCA[1]	BCA[0]	
		BFA[15]	BFA[14]	BFA[13]	BFA[12]	BFA[11]	BFA[10]	BFA[9]	BFA[8]	BFA[7]
		BFA[7]	BFA[6]	BFA[5]	BFA[4]	BFA[3]	BFA[2]	BFA[1]	BFA[0]	
	CNBCR21	BSA[15]	BSA[14]	BSA[13]	BSA[12]	BSA[11]	BSA[10]	BSA[9]	BSA[8]	BSA[7]
		BSA[7]	BSA[6]	BSA[5]	BSA[4]	BSA[3]	BSA[2]	-	-	
		BEA[15]	BEA[14]	BEA[13]	BEA[12]	BEA[11]	BEA[10]	BEA[9]	BEA[8]	BEA[7]
		BEA[7]	BEA[6]	BEA[5]	BEA[4]	BEA[3]	BEA[2]	BEA[1]	BEA[0]	
	CECR22	CE	TR	CT[1]	CT[0]	PCE_FSE	MDS[1]	MDS[0]	-	
		MASK[7]	MASK[6]	MASK[5]	MASK[4]	MASK[3]	MASK[2]	MASK[1]	MASK[0]	
		FSCD	-	-	PCTH_FSPC[4]	PCTH_FSPC[3]	PCTH_FSPC[2]	PCTH_FSPC[1]	PCTH_FSPC[0]	
		CA[8]	CA[7]	CA[6]	CA[5]	CA[4]	CA[3]	CA[2]	CA[1]	
	CSCR22	BM	BF	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GB	RDY
		STS[15]	STS[14]	STS[13]	STS[12]	STS[11]	STS[10]	STS[9]	STS[8]	STS[7]
		STS[7]	STS[6]	STS[5]	STS[4]	STS[3]	STS[2]	STS[1]	STS[0]	
	CCBCR22	BCA[15]	BCA[14]	BCA[13]	BCA[12]	BCA[11]	BCA[10]	BCA[9]	BCA[8]	BCA[7]
		BCA[7]	BCA[6]	BCA[5]	BCA[4]	BCA[3]	BCA[2]	BCA[1]	BCA[0]	
		BFA[15]	BFA[14]	BFA[13]	BFA[12]	BFA[11]	BFA[10]	BFA[9]	BFA[8]	BFA[7]
		BFA[7]	BFA[6]	BFA[5]	BFA[4]	BFA[3]	BFA[2]	BFA[1]	BFA[0]	
	CNBCR22	BSA[15]	BSA[14]	BSA[13]	BSA[12]	BSA[11]	BSA[10]	BSA[9]	BSA[8]	BSA[7]
		BSA[7]	BSA[6]	BSA[5]	BSA[4]	BSA[3]	BSA[2]	-	-	
		BEA[15]	BEA[14]	BEA[13]	BEA[12]	BEA[11]	BEA[10]	BEA[9]	BEA[8]	BEA[7]
		BEA[7]	BEA[6]	BEA[5]	BEA[4]	BEA[3]	BEA[2]	BEA[1]	BEA[0]	
	CECR23	CE	TR	CT[1]	CT[0]	PCE_FSE	MDS[1]	MDS[0]	-	
		MASK[7]	MASK[6]	MASK[5]	MASK[4]	MASK[3]	MASK[2]	MASK[1]	MASK[0]	
		FSCD	-	-	PCTH_FSPC[4]	PCTH_FSPC[3]	PCTH_FSPC[2]	PCTH_FSPC[1]	PCTH_FSPC[0]	
		CA[8]	CA[7]	CA[6]	CA[5]	CA[4]	CA[3]	CA[2]	CA[1]	
	CSCR23	BM	BF	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GB	RDY
		STS[15]	STS[14]	STS[13]	STS[12]	STS[11]	STS[10]	STS[9]	STS[8]	STS[7]
		STS[7]	STS[6]	STS[5]	STS[4]	STS[3]	STS[2]	STS[1]	STS[0]	
	CCBCR23	BCA[15]	BCA[14]	BCA[13]	BCA[12]	BCA[11]	BCA[10]	BCA[9]	BCA[8]	BCA[7]
		BCA[7]	BCA[6]	BCA[5]	BCA[4]	BCA[3]	BCA[2]	BCA[1]	BCA[0]	
		BFA[15]	BFA[14]	BFA[13]	BFA[12]	BFA[11]	BFA[10]	BFA[9]	BFA[8]	BFA[7]
		BFA[7]	BFA[6]	BFA[5]	BFA[4]	BFA[3]	BFA[2]	BFA[1]	BFA[0]	
	CNBCR23	BSA[15]	BSA[14]	BSA[13]	BSA[12]	BSA[11]	BSA[10]	BSA[9]	BSA[8]	BSA[7]
		BSA[7]	BSA[6]	BSA[5]	BSA[4]	BSA[3]	BSA[2]	-	-	
		BEA[15]	BEA[14]	BEA[13]	BEA[12]	BEA[11]	BEA[10]	BEA[9]	BEA[8]	BEA[7]
		BEA[7]	BEA[6]	BEA[5]	BEA[4]	BEA[3]	BEA[2]	BEA[1]	BEA[0]	
CECR24	CE	TR	CT[1]	CT[0]	PCE_FSE	MDS[1]	MDS[0]	-		
	MASK[7]	MASK[6]	MASK[5]	MASK[4]	MASK[3]	MASK[2]	MASK[1]	MASK[0]		
	FSCD	-	-	PCTH_FSPC[4]	PCTH_FSPC[3]	PCTH_FSPC[2]	PCTH_FSPC[1]	PCTH_FSPC[0]		
	CA[8]	CA[7]	CA[6]	CA[5]	CA[4]	CA[3]	CA[2]	CA[1]		
CSCR24	BM	BF	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	GB	RDY	
	STS[15]	STS[14]	STS[13]	STS[12]	STS[11]	STS[10]	STS[9]	STS[8]	STS[7]	
	STS[7]	STS[6]	STS[5]	STS[4]	STS[3]	STS[2]	STS[1]	STS[0]		

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Media local bus	CCBCR24	BCA[15]	BCA[14]	BCA[13]	BCA[12]	BCA[11]	BCA[10]	BCA[9]	BCA[8]
		BCA[7]	BCA[6]	BCA[5]	BCA[4]	BCA[3]	BCA[2]	BCA[1]	BCA[0]
		BFA[15]	BFA[14]	BFA[13]	BFA[12]	BFA[11]	BFA[10]	BFA[9]	BFA[8]
	CNBCR24	BFA[7]	BFA[6]	BFA[5]	BFA[4]	BFA[3]	BFA[2]	BFA[1]	BFA[0]
		BSA[15]	BSA[14]	BSA[13]	BSA[12]	BSA[11]	BSA[10]	BSA[9]	BSA[8]
		BSA[7]	BSA[6]	BSA[5]	BSA[4]	BSA[3]	BSA[2]	-	-
		BEA[15]	BEA[14]	BEA[13]	BEA[12]	BEA[11]	BEA[10]	BEA[9]	BEA[8]
	CECR25	BEA[7]	BEA[6]	BEA[5]	BEA[4]	BEA[3]	BEA[2]	BEA[1]	BEA[0]
		CE	TR	CT[1]	CT[0]	PCE_FSE	MDS[1]	MDS[0]	-
		MASK[7]	MASK[6]	MASK[5]	MASK[4]	MASK[3]	MASK[2]	MASK[1]	MASK[0]
		FSCD	-	-	PCTH_FSPC[4]	PCTH_FSPC[3]	PCTH_FSPC[2]	PCTH_FSPC[1]	PCTH_FSPC[0]
	CSCR25	CA[8]	CA[7]	CA[6]	CA[5]	CA[4]	CA[3]	CA[2]	CA[1]
		BM	BF	-	-	-	-	-	-
		-	-	-	-	-	-	GB	RDY
		STS[15]	STS[14]	STS[13]	STS[12]	STS[11]	STS[10]	STS[9]	STS[8]
	CCBCR25	STS[7]	STS[6]	STS[5]	STS[4]	STS[3]	STS[2]	STS[1]	STS[0]
		BCA[15]	BCA[14]	BCA[13]	BCA[12]	BCA[11]	BCA[10]	BCA[9]	BCA[8]
		BCA[7]	BCA[6]	BCA[5]	BCA[4]	BCA[3]	BCA[2]	BCA[1]	BCA[0]
		BFA[15]	BFA[14]	BFA[13]	BFA[12]	BFA[11]	BFA[10]	BFA[9]	BFA[8]
	CNBCR25	BFA[7]	BFA[6]	BFA[5]	BFA[4]	BFA[3]	BFA[2]	BFA[1]	BFA[0]
		BSA[15]	BSA[14]	BSA[13]	BSA[12]	BSA[11]	BSA[10]	BSA[9]	BSA[8]
		BSA[7]	BSA[6]	BSA[5]	BSA[4]	BSA[3]	BSA[2]	-	-
		BEA[15]	BEA[14]	BEA[13]	BEA[12]	BEA[11]	BEA[10]	BEA[9]	BEA[8]
	CECR26	BEA[7]	BEA[6]	BEA[5]	BEA[4]	BEA[3]	BEA[2]	BEA[1]	BEA[0]
		CE	TR	CT[1]	CT[0]	PCE_FSE	MDS[1]	MDS[0]	-
		MASK[7]	MASK[6]	MASK[5]	MASK[4]	MASK[3]	MASK[2]	MASK[1]	MASK[0]
		FSCD	-	-	PCTH_FSPC[4]	PCTH_FSPC[3]	PCTH_FSPC[2]	PCTH_FSPC[1]	PCTH_FSPC[0]
	CSCR26	CA[8]	CA[7]	CA[6]	CA[5]	CA[4]	CA[3]	CA[2]	CA[1]
		BM	BF	-	-	-	-	-	-
		-	-	-	-	-	-	GB	RDY
		STS[15]	STS[14]	STS[13]	STS[12]	STS[11]	STS[10]	STS[9]	STS[8]
	CCBCR26	STS[7]	STS[6]	STS[5]	STS[4]	STS[3]	STS[2]	STS[1]	STS[0]
		BCA[15]	BCA[14]	BCA[13]	BCA[12]	BCA[11]	BCA[10]	BCA[9]	BCA[8]
		BCA[7]	BCA[6]	BCA[5]	BCA[4]	BCA[3]	BCA[2]	BCA[1]	BCA[0]
		BFA[15]	BFA[14]	BFA[13]	BFA[12]	BFA[11]	BFA[10]	BFA[9]	BFA[8]
	CNBCR26	BFA[7]	BFA[6]	BFA[5]	BFA[4]	BFA[3]	BFA[2]	BFA[1]	BFA[0]
		BSA[15]	BSA[14]	BSA[13]	BSA[12]	BSA[11]	BSA[10]	BSA[9]	BSA[8]
		BSA[7]	BSA[6]	BSA[5]	BSA[4]	BSA[3]	BSA[2]	-	-
		BEA[15]	BEA[14]	BEA[13]	BEA[12]	BEA[11]	BEA[10]	BEA[9]	BEA[8]
	CECR27	BEA[7]	BEA[6]	BEA[5]	BEA[4]	BEA[3]	BEA[2]	BEA[1]	BEA[0]
		CE	TR	CT[1]	CT[0]	PCE_FSE	MDS[1]	MDS[0]	-
		MASK[7]	MASK[6]	MASK[5]	MASK[4]	MASK[3]	MASK[2]	MASK[1]	MASK[0]
		FSCD	-	-	PCTH_FSPC[4]	PCTH_FSPC[3]	PCTH_FSPC[2]	PCTH_FSPC[1]	PCTH_FSPC[0]
	CSCR27	CA[8]	CA[7]	CA[6]	CA[5]	CA[4]	CA[3]	CA[2]	CA[1]
		BM	BF	-	-	-	-	-	-
		-	-	-	-	-	-	GB	RDY
		STS[15]	STS[14]	STS[13]	STS[12]	STS[11]	STS[10]	STS[9]	STS[8]
	CCBCR27	STS[7]	STS[6]	STS[5]	STS[4]	STS[3]	STS[2]	STS[1]	STS[0]
		BCA[15]	BCA[14]	BCA[13]	BCA[12]	BCA[11]	BCA[10]	BCA[9]	BCA[8]
		BCA[7]	BCA[6]	BCA[5]	BCA[4]	BCA[3]	BCA[2]	BCA[1]	BCA[0]
		BFA[15]	BFA[14]	BFA[13]	BFA[12]	BFA[11]	BFA[10]	BFA[9]	BFA[8]
	CNBCR27	BFA[7]	BFA[6]	BFA[5]	BFA[4]	BFA[3]	BFA[2]	BFA[1]	BFA[0]
		BSA[15]	BSA[14]	BSA[13]	BSA[12]	BSA[11]	BSA[10]	BSA[9]	BSA[8]
		BSA[7]	BSA[6]	BSA[5]	BSA[4]	BSA[3]	BSA[2]	-	-
BEA[15]		BEA[14]	BEA[13]	BEA[12]	BEA[11]	BEA[10]	BEA[9]	BEA[8]	
		BEA[7]	BEA[6]	BEA[5]	BEA[4]	BEA[3]	BEA[2]	BEA[1]	BEA[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Media local bus	CECR28	CE	TR	CT[1]	CT[0]	PCE_FSE	MDS[1]	MDS[0]	-
		MASK[7]	MASK[6]	MASK[5]	MASK[4]	MASK[3]	MASK[2]	MASK[1]	MASK[0]
		FSCD	-	-	PCTH_FSPC[4]	PCTH_FSPC[3]	PCTH_FSPC[2]	PCTH_FSPC[1]	PCTH_FSPC[0]
		CA[8]	CA[7]	CA[6]	CA[5]	CA[4]	CA[3]	CA[2]	CA[1]
	CSCR28	BM	BF	-	-	-	-	-	-
		-	-	-	-	-	-	GB	RDY
		STS[15]	STS[14]	STS[13]	STS[12]	STS[11]	STS[10]	STS[9]	STS[8]
		STS[7]	STS[6]	STS[5]	STS[4]	STS[3]	STS[2]	STS[1]	STS[0]
	CCBCR28	BCA[15]	BCA[14]	BCA[13]	BCA[12]	BCA[11]	BCA[10]	BCA[9]	BCA[8]
		BCA[7]	BCA[6]	BCA[5]	BCA[4]	BCA[3]	BCA[2]	BCA[1]	BCA[0]
		BFA[15]	BFA[14]	BFA[13]	BFA[12]	BFA[11]	BFA[10]	BFA[9]	BFA[8]
		BFA[7]	BFA[6]	BFA[5]	BFA[4]	BFA[3]	BFA[2]	BFA[1]	BFA[0]
	CNBCR28	BSA[15]	BSA[14]	BSA[13]	BSA[12]	BSA[11]	BSA[10]	BSA[9]	BSA[8]
		BSA[7]	BSA[6]	BSA[5]	BSA[4]	BSA[3]	BSA[2]	-	-
		BEA[15]	BEA[14]	BEA[13]	BEA[12]	BEA[11]	BEA[10]	BEA[9]	BEA[8]
		BEA[7]	BEA[6]	BEA[5]	BEA[4]	BEA[3]	BEA[2]	BEA[1]	BEA[0]
	CECR29	CE	TR	CT[1]	CT[0]	PCE_FSE	MDS[1]	MDS[0]	-
		MASK[7]	MASK[6]	MASK[5]	MASK[4]	MASK[3]	MASK[2]	MASK[1]	MASK[0]
		FSCD	-	-	PCTH_FSPC[4]	PCTH_FSPC[3]	PCTH_FSPC[2]	PCTH_FSPC[1]	PCTH_FSPC[0]
		CA[8]	CA[7]	CA[6]	CA[5]	CA[4]	CA[3]	CA[2]	CA[1]
	CSCR29	BM	BF	-	-	-	-	-	-
		-	-	-	-	-	-	GB	RDY
		STS[15]	STS[14]	STS[13]	STS[12]	STS[11]	STS[10]	STS[9]	STS[8]
		STS[7]	STS[6]	STS[5]	STS[4]	STS[3]	STS[2]	STS[1]	STS[0]
	CCBCR29	BCA[15]	BCA[14]	BCA[13]	BCA[12]	BCA[11]	BCA[10]	BCA[9]	BCA[8]
		BCA[7]	BCA[6]	BCA[5]	BCA[4]	BCA[3]	BCA[2]	BCA[1]	BCA[0]
		BFA[15]	BFA[14]	BFA[13]	BFA[12]	BFA[11]	BFA[10]	BFA[9]	BFA[8]
		BFA[7]	BFA[6]	BFA[5]	BFA[4]	BFA[3]	BFA[2]	BFA[1]	BFA[0]
	CNBCR29	BSA[15]	BSA[14]	BSA[13]	BSA[12]	BSA[11]	BSA[10]	BSA[9]	BSA[8]
		BSA[7]	BSA[6]	BSA[5]	BSA[4]	BSA[3]	BSA[2]	-	-
		BEA[15]	BEA[14]	BEA[13]	BEA[12]	BEA[11]	BEA[10]	BEA[9]	BEA[8]
		BEA[7]	BEA[6]	BEA[5]	BEA[4]	BEA[3]	BEA[2]	BEA[1]	BEA[0]
	CECR30	CE	TR	CT[1]	CT[0]	PCE_FSE	MDS[1]	MDS[0]	-
		MASK[7]	MASK[6]	MASK[5]	MASK[4]	MASK[3]	MASK[2]	MASK[1]	MASK[0]
		FSCD	-	-	PCTH_FSPC[4]	PCTH_FSPC[3]	PCTH_FSPC[2]	PCTH_FSPC[1]	PCTH_FSPC[0]
		CA[8]	CA[7]	CA[6]	CA[5]	CA[4]	CA[3]	CA[2]	CA[1]
	CSCR30	BM	BF	-	-	-	-	-	-
		-	-	-	-	-	-	GB	RDY
		STS[15]	STS[14]	STS[13]	STS[12]	STS[11]	STS[10]	STS[9]	STS[8]
		STS[7]	STS[6]	STS[5]	STS[4]	STS[3]	STS[2]	STS[1]	STS[0]
	CCBCR30	BCA[15]	BCA[14]	BCA[13]	BCA[12]	BCA[11]	BCA[10]	BCA[9]	BCA[8]
		BCA[7]	BCA[6]	BCA[5]	BCA[4]	BCA[3]	BCA[2]	BCA[1]	BCA[0]
		BFA[15]	BFA[14]	BFA[13]	BFA[12]	BFA[11]	BFA[10]	BFA[9]	BFA[8]
		BFA[7]	BFA[6]	BFA[5]	BFA[4]	BFA[3]	BFA[2]	BFA[1]	BFA[0]
	CNBCR30	BSA[15]	BSA[14]	BSA[13]	BSA[12]	BSA[11]	BSA[10]	BSA[9]	BSA[8]
		BSA[7]	BSA[6]	BSA[5]	BSA[4]	BSA[3]	BSA[2]	-	-
		BEA[15]	BEA[14]	BEA[13]	BEA[12]	BEA[11]	BEA[10]	BEA[9]	BEA[8]
		BEA[7]	BEA[6]	BEA[5]	BEA[4]	BEA[3]	BEA[2]	BEA[1]	BEA[0]
	LCBCR0	TH[9]	TH[8]	TH[7]	TH[6]	TH[5]	TH[4]	TH[3]	TH[2]
		TH[1]	TH[0]	BD[8]	BD[7]	BD[6]	BD[5]	BD[4]	BD[3]
		BD[2]	BD[1]	BD[0]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	LCBCR1	TH[9]	TH[8]	TH[7]	TH[6]	TH[5]	TH[4]	TH[3]	TH[2]
		TH[1]	TH[0]	BD[8]	BD[7]	BD[6]	BD[5]	BD[4]	BD[3]
		BD[2]	BD[1]	BD[0]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Media local bus	LCBCR2	TH[9]	TH[8]	TH[7]	TH[6]	TH[5]	TH[4]	TH[3]	TH[2]
		TH[1]	TH[0]	BD[8]	BD[7]	BD[6]	BD[5]	BD[4]	BD[3]
		BD[2]	BD[1]	BD[0]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	LCBCR3	TH[9]	TH[8]	TH[7]	TH[6]	TH[5]	TH[4]	TH[3]	TH[2]
		TH[1]	TH[0]	BD[8]	BD[7]	BD[6]	BD[5]	BD[4]	BD[3]
		BD[2]	BD[1]	BD[0]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	LCBCR4	TH[9]	TH[8]	TH[7]	TH[6]	TH[5]	TH[4]	TH[3]	TH[2]
		TH[1]	TH[0]	BD[8]	BD[7]	BD[6]	BD[5]	BD[4]	BD[3]
		BD[2]	BD[1]	BD[0]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	LCBCR5	TH[9]	TH[8]	TH[7]	TH[6]	TH[5]	TH[4]	TH[3]	TH[2]
		TH[1]	TH[0]	BD[8]	BD[7]	BD[6]	BD[5]	BD[4]	BD[3]
		BD[2]	BD[1]	BD[0]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	LCBCR6	TH[9]	TH[8]	TH[7]	TH[6]	TH[5]	TH[4]	TH[3]	TH[2]
		TH[1]	TH[0]	BD[8]	BD[7]	BD[6]	BD[5]	BD[4]	BD[3]
		BD[2]	BD[1]	BD[0]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	LCBCR7	TH[9]	TH[8]	TH[7]	TH[6]	TH[5]	TH[4]	TH[3]	TH[2]
		TH[1]	TH[0]	BD[8]	BD[7]	BD[6]	BD[5]	BD[4]	BD[3]
		BD[2]	BD[1]	BD[0]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	LCBCR8	TH[9]	TH[8]	TH[7]	TH[6]	TH[5]	TH[4]	TH[3]	TH[2]
		TH[1]	TH[0]	BD[8]	BD[7]	BD[6]	BD[5]	BD[4]	BD[3]
		BD[2]	BD[1]	BD[0]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	LCBCR9	TH[9]	TH[8]	TH[7]	TH[6]	TH[5]	TH[4]	TH[3]	TH[2]
		TH[1]	TH[0]	BD[8]	BD[7]	BD[6]	BD[5]	BD[4]	BD[3]
		BD[2]	BD[1]	BD[0]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	LCBCR10	TH[9]	TH[8]	TH[7]	TH[6]	TH[5]	TH[4]	TH[3]	TH[2]
		TH[1]	TH[0]	BD[8]	BD[7]	BD[6]	BD[5]	BD[4]	BD[3]
		BD[2]	BD[1]	BD[0]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	LCBCR11	TH[9]	TH[8]	TH[7]	TH[6]	TH[5]	TH[4]	TH[3]	TH[2]
		TH[1]	TH[0]	BD[8]	BD[7]	BD[6]	BD[5]	BD[4]	BD[3]
		BD[2]	BD[1]	BD[0]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	LCBCR12	TH[9]	TH[8]	TH[7]	TH[6]	TH[5]	TH[4]	TH[3]	TH[2]
		TH[1]	TH[0]	BD[8]	BD[7]	BD[6]	BD[5]	BD[4]	BD[3]
		BD[2]	BD[1]	BD[0]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	LCBCR13	TH[9]	TH[8]	TH[7]	TH[6]	TH[5]	TH[4]	TH[3]	TH[2]
		TH[1]	TH[0]	BD[8]	BD[7]	BD[6]	BD[5]	BD[4]	BD[3]
		BD[2]	BD[1]	BD[0]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	LCBCR14	TH[9]	TH[8]	TH[7]	TH[6]	TH[5]	TH[4]	TH[3]	TH[2]
		TH[1]	TH[0]	BD[8]	BD[7]	BD[6]	BD[5]	BD[4]	BD[3]
		BD[2]	BD[1]	BD[0]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	LCBCR15	TH[9]	TH[8]	TH[7]	TH[6]	TH[5]	TH[4]	TH[3]	TH[2]
		TH[1]	TH[0]	BD[8]	BD[7]	BD[6]	BD[5]	BD[4]	BD[3]
		BD[2]	BD[1]	BD[0]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Media local bus	LCBCR16	TH[9]	TH[8]	TH[7]	TH[6]	TH[5]	TH[4]	TH[3]	TH[2]
		TH[1]	TH[0]	BD[8]	BD[7]	BD[6]	BD[5]	BD[4]	BD[3]
		BD[2]	BD[1]	BD[0]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	LCBCR17	TH[9]	TH[8]	TH[7]	TH[6]	TH[5]	TH[4]	TH[3]	TH[2]
		TH[1]	TH[0]	BD[8]	BD[7]	BD[6]	BD[5]	BD[4]	BD[3]
		BD[2]	BD[1]	BD[0]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	LCBCR18	TH[9]	TH[8]	TH[7]	TH[6]	TH[5]	TH[4]	TH[3]	TH[2]
		TH[1]	TH[0]	BD[8]	BD[7]	BD[6]	BD[5]	BD[4]	BD[3]
		BD[2]	BD[1]	BD[0]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	LCBCR19	TH[9]	TH[8]	TH[7]	TH[6]	TH[5]	TH[4]	TH[3]	TH[2]
		TH[1]	TH[0]	BD[8]	BD[7]	BD[6]	BD[5]	BD[4]	BD[3]
		BD[2]	BD[1]	BD[0]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	LCBCR20	TH[9]	TH[8]	TH[7]	TH[6]	TH[5]	TH[4]	TH[3]	TH[2]
		TH[1]	TH[0]	BD[8]	BD[7]	BD[6]	BD[5]	BD[4]	BD[3]
		BD[2]	BD[1]	BD[0]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	LCBCR21	TH[9]	TH[8]	TH[7]	TH[6]	TH[5]	TH[4]	TH[3]	TH[2]
		TH[1]	TH[0]	BD[8]	BD[7]	BD[6]	BD[5]	BD[4]	BD[3]
		BD[2]	BD[1]	BD[0]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	LCBCR22	TH[9]	TH[8]	TH[7]	TH[6]	TH[5]	TH[4]	TH[3]	TH[2]
		TH[1]	TH[0]	BD[8]	BD[7]	BD[6]	BD[5]	BD[4]	BD[3]
		BD[2]	BD[1]	BD[0]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	LCBCR23	TH[9]	TH[8]	TH[7]	TH[6]	TH[5]	TH[4]	TH[3]	TH[2]
		TH[1]	TH[0]	BD[8]	BD[7]	BD[6]	BD[5]	BD[4]	BD[3]
		BD[2]	BD[1]	BD[0]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	LCBCR24	TH[9]	TH[8]	TH[7]	TH[6]	TH[5]	TH[4]	TH[3]	TH[2]
		TH[1]	TH[0]	BD[8]	BD[7]	BD[6]	BD[5]	BD[4]	BD[3]
		BD[2]	BD[1]	BD[0]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	LCBCR25	TH[9]	TH[8]	TH[7]	TH[6]	TH[5]	TH[4]	TH[3]	TH[2]
		TH[1]	TH[0]	BD[8]	BD[7]	BD[6]	BD[5]	BD[4]	BD[3]
		BD[2]	BD[1]	BD[0]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	LCBCR26	TH[9]	TH[8]	TH[7]	TH[6]	TH[5]	TH[4]	TH[3]	TH[2]
		TH[1]	TH[0]	BD[8]	BD[7]	BD[6]	BD[5]	BD[4]	BD[3]
		BD[2]	BD[1]	BD[0]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	LCBCR27	TH[9]	TH[8]	TH[7]	TH[6]	TH[5]	TH[4]	TH[3]	TH[2]
		TH[1]	TH[0]	BD[8]	BD[7]	BD[6]	BD[5]	BD[4]	BD[3]
		BD[2]	BD[1]	BD[0]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	LCBCR28	TH[9]	TH[8]	TH[7]	TH[6]	TH[5]	TH[4]	TH[3]	TH[2]
		TH[1]	TH[0]	BD[8]	BD[7]	BD[6]	BD[5]	BD[4]	BD[3]
		BD[2]	BD[1]	BD[0]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	LCBCR29	TH[9]	TH[8]	TH[7]	TH[6]	TH[5]	TH[4]	TH[3]	TH[2]
		TH[1]	TH[0]	BD[8]	BD[7]	BD[6]	BD[5]	BD[4]	BD[3]
		BD[2]	BD[1]	BD[0]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]
	LCBCR30	TH[9]	TH[8]	TH[7]	TH[6]	TH[5]	TH[4]	TH[3]	TH[2]
		TH[1]	TH[0]	BD[8]	BD[7]	BD[6]	BD[5]	BD[4]	BD[3]
		BD[2]	BD[1]	BD[0]	SA[12]	SA[11]	SA[10]	SA[9]	SA[8]
		SA[7]	SA[6]	SA[5]	SA[4]	SA[3]	SA[2]	SA[1]	SA[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
CAN interface	RSCAN0CmCFG (m = 0 to 4)	-	-	-	-	-	-	SJW[1]	SJW[0]	
		-	TSEG2[2]	TSEG2[1]	TSEG2[0]	TSEG1[3]	TSEG1[2]	TSEG1[1]	TSEG1[0]	
		-	-	-	-	-	-	BRP[9]	BRP[8]	
		BRP[7]	BRP[6]	BRP[5]	BRP[4]	BRP[3]	BRP[2]	BRP[1]	BRP[0]	
	RSCAN0CmCTR (m = 0 to 4)	-	-	-	-	-	-	CTMS[1]	CTMS[0]	CTME
		ERRD	BOM[1]	BOM[0]	-	-	-	-	-	TAIE
		ALIE	BLIE	OLIE	BORIE	BOEIE	EPIE	EWIE	BEIE	
		-	-	-	-	RTBO	CSLPR	CHMDC[1]	CHMDC[0]	
	RSCAN0CmSTS (m = 0 to 4)	TEC[7]	TEC[6]	TEC[5]	TEC[4]	TEC[3]	TEC[2]	TEC[1]	TEC[0]	
		REC[7]	REC[6]	REC[5]	REC[4]	REC[3]	REC[2]	REC[1]	REC[0]	
		-	-	-	-	-	-	-	-	-
		COMSTS	RECSTS	TRMSTS	BOSTS	EPSTS	CSLPSTS	CHLTSTS	CRSTSTS	
	RSCAN0CmERFL (m = 0 to 4)	-	CRCREG[14]	CRCREG[13]	CRCREG[12]	CRCREG[11]	CRCREG[10]	CRCREG[9]	CRCREG[8]	
		CRCREG[7]	CRCREG[6]	CRCREG[5]	CRCREG[4]	CRCREG[3]	CRCREG[2]	CRCREG[1]	CRCREG[0]	
		-	ADERR	B0ERR	B1ERR	CERR	AERR	FERR	SERR	
		ALF	BLF	OVLf	BORF	BOEF	EPF	EPF	BEF	
	RSCAN0GCFG	ITRCP[15]	ITRCP[14]	ITRCP[13]	ITRCP[12]	ITRCP[11]	ITRCP[10]	ITRCP[9]	ITRCP[8]	
		ITRCP[7]	ITRCP[6]	ITRCP[5]	ITRCP[4]	ITRCP[3]	ITRCP[2]	ITRCP[1]	ITRCP[0]	
		TSBTCS[2]	TSBTCS[1]	TSBTCS[0]	TSSS	TSP[3]	TSP[2]	TSP[1]	TSP[0]	
		-	-	-	DCS	MME	DRE	DCE	TPRI	
	RSCAN0GCTR	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	TSRST
		-	-	-	-	-	THLEIE	MEIE	DEIE	
		-	-	-	-	-	GSLPR	GMDC[1]	GMDC[0]	
	RSCAN0GSTS	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	GRAMINIT	GSLPSTS	GHLTSTS	GRSTSTS	
	RSCAN0GERFL	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	THLES	MES	DEF	
	RSCAN0GTSC	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		TS[15]	TS[14]	TS[13]	TS[12]	TS[11]	TS[10]	TS[9]	TS[8]	
		TS[7]	TS[6]	TS[5]	TS[4]	TS[3]	TS[2]	TS[1]	TS[0]	
	RSCAN0GAFLECTR	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	AFLDAE
		-	-	-	AFLPN[4]	AFLPN[3]	AFLPN[2]	AFLPN[1]	AFLPN[0]	
	RSCAN0GAFLCFG0	RNC0[7]	RNC0[6]	RNC0[5]	RNC0[4]	RNC0[3]	RNC0[2]	RNC0[1]	RNC0[0]	
		RNC1[7]	RNC1[6]	RNC1[5]	RNC1[4]	RNC1[3]	RNC1[2]	RNC1[1]	RNC1[0]	
		RNC2[7]	RNC2[6]	RNC2[5]	RNC2[4]	RNC2[3]	RNC2[2]	RNC2[1]	RNC2[0]	
		RNC3[7]	RNC3[6]	RNC3[5]	RNC3[4]	RNC3[3]	RNC3[2]	RNC3[1]	RNC3[0]	
	RSCAN0GAFLCFG1	RNC4[7]	RNC4[6]	RNC4[5]	RNC4[4]	RNC4[3]	RNC4[2]	RNC4[1]	RNC4[0]	
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
	RSCAN0RMNB	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		NRXMB[7]	NRXMB[6]	NRXMB[5]	NRXMB[4]	NRXMB[3]	NRXMB[2]	NRXMB[1]	NRXMB[0]	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
CAN interface	RSCAN0RMND0	RMNS[31]	RMNS[30]	RMNS[29]	RMNS[28]	RMNS[27]	RMNS[26]	RMNS[25]	RMNS[24]
		RMNS[23]	RMNS[22]	RMNS[21]	RMNS[20]	RMNS[19]	RMNS[18]	RMNS[17]	RMNS[16]
		RMNS[15]	RMNS[14]	RMNS[13]	RMNS[12]	RMNS[11]	RMNS[10]	RMNS[9]	RMNS[8]
		RMNS[7]	RMNS[6]	RMNS[5]	RMNS[4]	RMNS[3]	RMNS[2]	RMNS[1]	RMNS[0]
	RSCAN0RMND1	RMNS[63]	RMNS[62]	RMNS[61]	RMNS[60]	RMNS[59]	RMNS[58]	RMNS[57]	RMNS[56]
		RMNS[55]	RMNS[54]	RMNS[53]	RMNS[52]	RMNS[51]	RMNS[50]	RMNS[49]	RMNS[48]
		RMNS[47]	RMNS[46]	RMNS[45]	RMNS[44]	RMNS[43]	RMNS[42]	RMNS[41]	RMNS[40]
		RMNS[39]	RMNS[38]	RMNS[37]	RMNS[36]	RMNS[35]	RMNS[34]	RMNS[33]	RMNS[32]
	RSCAN0RMND2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		RMNS[79]	RMNS[78]	RMNS[77]	RMNS[76]	RMNS[75]	RMNS[74]	RMNS[73]	RMNS[72]
		RMNS[71]	RMNS[70]	RMNS[69]	RMNS[68]	RMNS[67]	RMNS[66]	RMNS[65]	RMNS[64]
	RSCAN0RFCCx (x = 0 to 7)	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		RFIGCV[2]	RFIGCV[1]	RFIGCV[0]	RFIM	-	RFDC[2]	RFDC[1]	RFDC[0]
		-	-	-	-	-	-	RFIE	RFE
	RSCAN0RFSTx (x = 0 to 7)	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		RFMC[7]	RFMC[6]	RFMC[5]	RFMC[4]	RFMC[3]	RFMC[2]	RFMC[1]	RFMC[0]
		-	-	-	-	RFIF	RFMLT	RFLL	RFEMP
	RSCAN0RFPCTRx (x = 0 to 7)	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		RFPC[7]	RFPC[6]	RFPC[5]	RFPC[4]	RFPC[3]	RFPC[2]	RFPC[1]	RFPC[0]
	RSCAN0CFCK (k = 0 to 14)	CFITT[7]	CFITT[6]	CFITT[5]	CFITT[4]	CFITT[3]	CFITT[2]	CFITT[1]	CFITT[0]
		CFTML[3]	CFTML[2]	CFTML[1]	CFTML[0]	CFITR	CFITSS	CFM[1]	CFM[0]
		CFIGCV[2]	CFIGCV[1]	CFIGCV[0]	CFIM	-	CFDC[2]	CFDC[1]	CFDC[0]
		-	-	-	-	-	CFXIE	CFXIE	CFE
	RSCAN0CFSTk (k = 0 to 14)	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		CFMC[7]	CFMC[6]	CFMC[5]	CFMC[4]	CFMC[3]	CFMC[2]	CFMC[1]	CFMC[0]
		-	-	-	CFXIF	CFXIF	CFMLT	CFLL	CFEMP
	RSCAN0CFPCTRk (k = 0 to 14)	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		CFPC[7]	CFPC[6]	CFPC[5]	CFPC[4]	CFPC[3]	CFPC[2]	CFPC[1]	CFPC[0]
	RSCAN0FESTS	-	-	-	-	-	-	-	-
		-	CF14EMP	CF13EMP	CF12EMP	CF11EMP	CF10EMP	CF9EMP	CF8EMP
		CF7EMP	CF6EMP	CF5EMP	CF4EMP	CF3EMP	CF2EMP	CF1EMP	CF0EMP
		RF7EMP	RF6EMP	RF5EMP	RF4EMP	RF3EMP	RF2EMP	RF1EMP	RF0EMP
	RSCAN0FFSTS	-	-	-	-	-	-	-	-
		-	CF14FLL	CF13FLL	CF12FLL	CF11FLL	CF10FLL	CF9FLL	CF8FLL
		CF7FLL	CF6FLL	CF5FLL	CF4FLL	CF3FLL	CF2FLL	CF1FLL	CF0FLL
		RF7FLL	RF6FLL	RF5FLL	RF4FLL	RF3FLL	RF2FLL	RF1FLL	RF0FLL
	RSCAN0FMSTS	-	-	-	-	-	-	-	-
		-	CF14MLT	CF13MLT	CF12MLT	CF11MLT	CF10MLT	CF9MLT	CF8MLT
		CF7MLT	CF6MLT	CF5MLT	CF4MLT	CF3MLT	CF2MLT	CF1MLT	CF0MLT
		RF7MLT	RF6MLT	RF5MLT	RF4MLT	RF3MLT	RF2MLT	RF1MLT	RF0MLT
	RSCAN0RFISTS	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
-		-	-	-	-	-	-	-	
RF7IF		RF6IF	RF5IF	RF4IF	RF3IF	RF2IF	RF1IF	RF0IF	
RSCAN0CFRISTS	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	
	-	CF14RXIF	CF13RXIF	CF12RXIF	CF11RXIF	CF10RXIF	CF9RXIF	CF8RXIF	
	CF7RXIF	CF6RXIF	CF5RXIF	CF4RXIF	CF3RXIF	CF2RXIF	CF1RXIF	CF0RXIF	



**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
CAN interface	RSCAN0CTISIS	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	CF14TXIF	CF13TXIF	CF12TXIF	CF11TXIF	CF10TXIF	CF9TXIF	CF8TXIF	
		CF7TXIF	CF6TXIF	CF5TXIF	CF4TXIF	CF3TXIF	CF2TXIF	CF1TXIF	CF0TXIF	
	RSCAN0TMCp (p = 0 to 79)	-	-	-	-	-	-	TMOM	TMTAR	TMTR
		-	-	-	-	-	-	-	-	-
	RSCAN0TMTSp (p = 0 to 79)	-	-	-	TMTARM	TMTRM	TMTRF[1]	TMTRF[0]	TMTSTS	-
		-	-	-	-	-	-	-	-	-
	RSCAN0TMRSTS0	TMRSTS[31]	TMRSTS[30]	TMRSTS[29]	TMRSTS[28]	TMRSTS[27]	TMRSTS[26]	TMRSTS[25]	TMRSTS[24]	
		TMRSTS[23]	TMRSTS[22]	TMRSTS[21]	TMRSTS[20]	TMRSTS[19]	TMRSTS[18]	TMRSTS[17]	TMRSTS[16]	
		TMRSTS[15]	TMRSTS[14]	TMRSTS[13]	TMRSTS[12]	TMRSTS[11]	TMRSTS[10]	TMRSTS[9]	TMRSTS[8]	
		TMRSTS[7]	TMRSTS[6]	TMRSTS[5]	TMRSTS[4]	TMRSTS[3]	TMRSTS[2]	TMRSTS[1]	TMRSTS[0]	
	RSCAN0TMRSTS1	TMRSTS[63]	TMRSTS[62]	TMRSTS[61]	TMRSTS[60]	TMRSTS[59]	TMRSTS[58]	TMRSTS[57]	TMRSTS[56]	
		TMRSTS[55]	TMRSTS[54]	TMRSTS[53]	TMRSTS[52]	TMRSTS[51]	TMRSTS[50]	TMRSTS[49]	TMRSTS[48]	
		TMRSTS[47]	TMRSTS[46]	TMRSTS[45]	TMRSTS[44]	TMRSTS[43]	TMRSTS[42]	TMRSTS[41]	TMRSTS[40]	
		TMRSTS[39]	TMRSTS[38]	TMRSTS[37]	TMRSTS[36]	TMRSTS[35]	TMRSTS[34]	TMRSTS[33]	TMRSTS[32]	
	RSCAN0TMRSTS2	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		TMRSTS[79]	TMRSTS[78]	TMRSTS[77]	TMRSTS[76]	TMRSTS[75]	TMRSTS[74]	TMRSTS[73]	TMRSTS[72]	
		TMRSTS[71]	TMRSTS[70]	TMRSTS[69]	TMRSTS[68]	TMRSTS[67]	TMRSTS[66]	TMRSTS[65]	TMRSTS[64]	
	RSCAN0TMTARSTS 0	TMTARSTS[31]	TMTARSTS[30]	TMTARSTS[29]	TMTARSTS[28]	TMTARSTS[27]	TMTARSTS[26]	TMTARSTS[25]	TMTARSTS[24]	
		TMTARSTS[23]	TMTARSTS[22]	TMTARSTS[21]	TMTARSTS[20]	TMTARSTS[19]	TMTARSTS[18]	TMTARSTS[17]	TMTARSTS[16]	
		TMTARSTS[15]	TMTARSTS[14]	TMTARSTS[13]	TMTARSTS[12]	TMTARSTS[11]	TMTARSTS[10]	TMTARSTS[9]	TMTARSTS[8]	
		TMTARSTS[7]	TMTARSTS[6]	TMTARSTS[5]	TMTARSTS[4]	TMTARSTS[3]	TMTARSTS[2]	TMTARSTS[1]	TMTARSTS[0]	
	RSCAN0TMTARSTS 1	TMTARSTS[63]	TMTARSTS[62]	TMTARSTS[61]	TMTARSTS[60]	TMTARSTS[59]	TMTARSTS[58]	TMTARSTS[57]	TMTARSTS[56]	
		TMTARSTS[55]	TMTARSTS[54]	TMTARSTS[53]	TMTARSTS[52]	TMTARSTS[51]	TMTARSTS[50]	TMTARSTS[49]	TMTARSTS[48]	
		TMTARSTS[47]	TMTARSTS[46]	TMTARSTS[45]	TMTARSTS[44]	TMTARSTS[43]	TMTARSTS[42]	TMTARSTS[41]	TMTARSTS[40]	
		TMTARSTS[39]	TMTARSTS[38]	TMTARSTS[37]	TMTARSTS[36]	TMTARSTS[35]	TMTARSTS[34]	TMTARSTS[33]	TMTARSTS[32]	
	RSCAN0TMTARSTS 2	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		TMTARSTS[79]	TMTARSTS[78]	TMTARSTS[77]	TMTARSTS[76]	TMTARSTS[75]	TMTARSTS[74]	TMTARSTS[73]	TMTARSTS[72]	
		TMTARSTS[71]	TMTARSTS[70]	TMTARSTS[69]	TMTARSTS[68]	TMTARSTS[67]	TMTARSTS[66]	TMTARSTS[65]	TMTARSTS[64]	
	RSCAN0TMTCCSTS0	TMTCSTS[31]	TMTCSTS[30]	TMTCSTS[29]	TMTCSTS[28]	TMTCSTS[27]	TMTCSTS[26]	TMTCSTS[25]	TMTCSTS[24]	
		TMTCSTS[23]	TMTCSTS[22]	TMTCSTS[21]	TMTCSTS[20]	TMTCSTS[19]	TMTCSTS[18]	TMTCSTS[17]	TMTCSTS[16]	
		TMTCSTS[15]	TMTCSTS[14]	TMTCSTS[13]	TMTCSTS[12]	TMTCSTS[11]	TMTCSTS[10]	TMTCSTS[9]	TMTCSTS[8]	
		TMTCSTS[7]	TMTCSTS[6]	TMTCSTS[5]	TMTCSTS[4]	TMTCSTS[3]	TMTCSTS[2]	TMTCSTS[1]	TMTCSTS[0]	
	RSCAN0TMTCCSTS1	TMTCSTS[63]	TMTCSTS[62]	TMTCSTS[61]	TMTCSTS[60]	TMTCSTS[59]	TMTCSTS[58]	TMTCSTS[57]	TMTCSTS[56]	
		TMTCSTS[55]	TMTCSTS[54]	TMTCSTS[53]	TMTCSTS[52]	TMTCSTS[51]	TMTCSTS[50]	TMTCSTS[49]	TMTCSTS[48]	
		TMTCSTS[47]	TMTCSTS[46]	TMTCSTS[45]	TMTCSTS[44]	TMTCSTS[43]	TMTCSTS[42]	TMTCSTS[41]	TMTCSTS[40]	
		TMTCSTS[39]	TMTCSTS[38]	TMTCSTS[37]	TMTCSTS[36]	TMTCSTS[35]	TMTCSTS[34]	TMTCSTS[33]	TMTCSTS[32]	
	RSCAN0TMTCCSTS2	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		TMTCSTS[79]	TMTCSTS[78]	TMTCSTS[77]	TMTCSTS[76]	TMTCSTS[75]	TMTCSTS[74]	TMTCSTS[73]	TMTCSTS[72]	
		TMTCSTS[71]	TMTCSTS[70]	TMTCSTS[69]	TMTCSTS[68]	TMTCSTS[67]	TMTCSTS[66]	TMTCSTS[65]	TMTCSTS[64]	
	RSCAN0TMTASTS0	TMTASTS[31]	TMTASTS[30]	TMTASTS[29]	TMTASTS[28]	TMTASTS[27]	TMTASTS[26]	TMTASTS[25]	TMTASTS[24]	
		TMTASTS[23]	TMTASTS[22]	TMTASTS[21]	TMTASTS[20]	TMTASTS[19]	TMTASTS[18]	TMTASTS[17]	TMTASTS[16]	
		TMTASTS[15]	TMTASTS[14]	TMTASTS[13]	TMTASTS[12]	TMTASTS[11]	TMTASTS[10]	TMTASTS[9]	TMTASTS[8]	
		TMTASTS[7]	TMTASTS[6]	TMTASTS[5]	TMTASTS[4]	TMTASTS[3]	TMTASTS[2]	TMTASTS[1]	TMTASTS[0]	
	RSCAN0TMTASTS1	TMTASTS[63]	TMTASTS[62]	TMTASTS[61]	TMTASTS[60]	TMTASTS[59]	TMTASTS[58]	TMTASTS[57]	TMTASTS[56]	
		TMTASTS[55]	TMTASTS[54]	TMTASTS[53]	TMTASTS[52]	TMTASTS[51]	TMTASTS[50]	TMTASTS[49]	TMTASTS[48]	
		TMTASTS[47]	TMTASTS[46]	TMTASTS[45]	TMTASTS[44]	TMTASTS[43]	TMTASTS[42]	TMTASTS[41]	TMTASTS[40]	
		TMTASTS[39]	TMTASTS[38]	TMTASTS[37]	TMTASTS[36]	TMTASTS[35]	TMTASTS[34]	TMTASTS[33]	TMTASTS[32]	
	RSCAN0TMTASTS2	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		TMTASTS[79]	TMTASTS[78]	TMTASTS[77]	TMTASTS[76]	TMTASTS[75]	TMTASTS[74]	TMTASTS[73]	TMTASTS[72]	
		TMTASTS[71]	TMTASTS[70]	TMTASTS[69]	TMTASTS[68]	TMTASTS[67]	TMTASTS[66]	TMTASTS[65]	TMTASTS[64]	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
CAN interface	RSCAN0TMIEC0	TMIE[31]	TMIE[30]	TMIE[29]	TMIE[28]	TMIE[27]	TMIE[26]	TMIE[25]	TMIE[24]
		TMIE[23]	TMIE[22]	TMIE[21]	TMIE[20]	TMIE[19]	TMIE[18]	TMIE[17]	TMIE[16]
		TMIE[15]	TMIE[14]	TMIE[13]	TMIE[12]	TMIE[11]	TMIE[10]	TMIE[9]	TMIE[8]
		TMIE[7]	TMIE[6]	TMIE[5]	TMIE[4]	TMIE[3]	TMIE[2]	TMIE[1]	TMIE[0]
	RSCAN0TMIEC1	TMIE[63]	TMIE[62]	TMIE[61]	TMIE[60]	TMIE[59]	TMIE[58]	TMIE[57]	TMIE[56]
		TMIE[55]	TMIE[54]	TMIE[53]	TMIE[52]	TMIE[51]	TMIE[50]	TMIE[49]	TMIE[48]
		TMIE[47]	TMIE[46]	TMIE[45]	TMIE[44]	TMIE[43]	TMIE[42]	TMIE[41]	TMIE[40]
		TMIE[39]	TMIE[38]	TMIE[37]	TMIE[36]	TMIE[35]	TMIE[34]	TMIE[33]	TMIE[32]
	RSCAN0TMIEC2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		TMIE[79]	TMIE[78]	TMIE[77]	TMIE[76]	TMIE[75]	TMIE[74]	TMIE[73]	TMIE[72]
		TMIE[71]	TMIE[70]	TMIE[69]	TMIE[68]	TMIE[67]	TMIE[66]	TMIE[65]	TMIE[64]
	RSCAN0TXQCCm (m = 0 to 4)	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	TXQIM	TXQIE	TXQDC[3]	TXQDC[2]	TXQDC[1]	TXQDC[0]
		-	-	-	-	-	-	-	TXQE
	RSCAN0TXQSTSm (m = 0 to 4)	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	TXQIF	TXQFLL	TXQEMP
	RSCAN0TXQPCTR m (m = 0 to 4)	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		TXQPC[7]	TXQPC[6]	TXQPC[5]	TXQPC[4]	TXQPC[3]	TXQPC[2]	TXQPC[1]	TXQPC[0]
	RSCAN0THLCCm (m = 0 to 4)	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	THLDTE	THLIM	THLIE
		-	-	-	-	-	-	-	THLE
	RSCAN0THLSTSm (m = 0 to 4)	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	THLMC[4]	THLMC[3]	THLMC[2]	THLMC[1]	THLMC[0]
		-	-	-	-	THLIF	THLELT	THLFLL	THLEMP
	RSCAN0THLPCTRm (m = 0 to 4)	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		THLPC[7]	THLPC[6]	THLPC[5]	THLPC[4]	THLPC[3]	THLPC[2]	THLPC[1]	THLPC[0]
	RSCAN0GTINTSTS0	-	-	-	THIF3	CFTIF3	TQIF3	TAIF3	TSIF3
		-	-	-	THIF2	CFTIF2	TQIF2	TAIF2	TSIF2
		-	-	-	THIF1	CFTIF1	TQIF1	TAIF1	TSIF1
		-	-	-	THIF0	CFTIF0	TQIF0	TAIF0	TSIF0
	RSCAN0GTINTSTS1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	THIF4	CFTIF4	TQIF4	TAIF4	TSIF4
	RSCAN0GTSTCFG	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	C4ICBCE	C3ICBCE	C2ICBCE	C1ICBCE	COICBCE
	RSCAN0GTSTCTR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	ICBCTME
	RSCAN0GLOCKK	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		LOCK[15]	LOCK[14]	LOCK[13]	LOCK[12]	LOCK[11]	LOCK[10]	LOCK[9]	LOCK[8]
		LOCK[7]	LOCK[6]	LOCK[5]	LOCK[4]	LOCK[3]	LOCK[2]	LOCK[1]	LOCK[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
CAN interface	RSCAN0GAFLIDj (j = 0 to 15)	GAFLIDE	GAFLRTR	GAFLLB	GAFLID[28]	GAFLID[27]	GAFLID[26]	GAFLID[25]	GAFLID[24]
		GAFLID[23]	GAFLID[22]	GAFLID[21]	GAFLID[20]	GAFLID[19]	GAFLID[18]	GAFLID[17]	GAFLID[16]
		GAFLID[15]	GAFLID[14]	GAFLID[13]	GAFLID[12]	GAFLID[11]	GAFLID[10]	GAFLID[9]	GAFLID[8]
		GAFLID[7]	GAFLID[6]	GAFLID[5]	GAFLID[4]	GAFLID[3]	GAFLID[2]	GAFLID[1]	GAFLID[0]
	RSCAN0GAFLMj (j = 0 to 15)	GAFLIDEM	GAFLRTRM	-	GAFLIDM[28]	GAFLIDM[27]	GAFLIDM[26]	GAFLIDM[25]	GAFLIDM[24]
		GAFLIDM[23]	GAFLIDM[22]	GAFLIDM[21]	GAFLIDM[20]	GAFLIDM[19]	GAFLIDM[18]	GAFLIDM[17]	GAFLIDM[16]
		GAFLIDM[15]	GAFLIDM[14]	GAFLIDM[13]	GAFLIDM[12]	GAFLIDM[11]	GAFLIDM[10]	GAFLIDM[9]	GAFLIDM[8]
		GAFLIDM[7]	GAFLIDM[6]	GAFLIDM[5]	GAFLIDM[4]	GAFLIDM[3]	GAFLIDM[2]	GAFLIDM[1]	GAFLIDM[0]
	RSCAN0GAFLP0j (j = 0 to 15)	GAFLDLC[3]	GAFLDLC[2]	GAFLDLC[1]	GAFLDLC[0]	GAFLPTR[11]	GAFLPTR[10]	GAFLPTR[9]	GAFLPTR[8]
		GAFLPTR[7]	GAFLPTR[6]	GAFLPTR[5]	GAFLPTR[4]	GAFLPTR[3]	GAFLPTR[2]	GAFLPTR[1]	GAFLPTR[0]
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	RSCAN0GAFLP1j (j = 0 to 15)	-	-	-	-	-	-	-	-
		-	GAFLFDP[22]	GAFLFDP[21]	GAFLFDP[20]	GAFLFDP[19]	GAFLFDP[18]	GAFLFDP[17]	GAFLFDP[16]
		GAFLFDP[15]	GAFLFDP[14]	GAFLFDP[13]	GAFLFDP[12]	GAFLFDP[11]	GAFLFDP[10]	GAFLFDP[9]	GAFLFDP[8]
		GAFLFDP[7]	GAFLFDP[6]	GAFLFDP[5]	GAFLFDP[4]	GAFLFDP[3]	GAFLFDP[2]	GAFLFDP[1]	GAFLFDP[0]
	RSCAN0RMIDq (q = 0 to 79)	PMIDE	PMRTR	-	RMID[28]	RMID[27]	RMID[26]	RMID[25]	RMID[24]
		RMID[23]	RMID[22]	RMID[21]	RMID[20]	RMID[19]	RMID[18]	RMID[17]	RMID[16]
		RMID[15]	RMID[14]	RMID[13]	RMID[12]	RMID[11]	RMID[10]	RMID[9]	RMID[8]
		RMID[7]	RMID[6]	RMID[5]	RMID[4]	RMID[3]	RMID[2]	RMID[1]	RMID[0]
	RSCAN0RMPTRq (q = 0 to 79)	RMDLC[3]	RMDLC[2]	RMDLC[1]	RMDLC[0]	RMPTR[11]	RMPTR[10]	RMPTR[9]	RMPTR[8]
		RMPTR[7]	RMPTR[6]	RMPTR[5]	RMPTR[4]	RMPTR[3]	RMPTR[2]	RMPTR[1]	RMPTR[0]
		RMTS[15]	RMTS[14]	RMTS[13]	RMTS[12]	RMTS[11]	RMTS[10]	RMTS[9]	RMTS[8]
		RMTS[7]	RMTS[6]	RMTS[5]	RMTS[4]	RMTS[3]	RMTS[2]	RMTS[1]	RMTS[0]
	RSCAN0RMDf0q (q = 0 to 79)	RMDB3[7]	RMDB3[6]	RMDB3[5]	RMDB3[4]	RMDB3[3]	RMDB3[2]	RMDB3[1]	RMDB3[0]
		RMDB2[7]	RMDB2[6]	RMDB2[5]	RMDB2[4]	RMDB2[3]	RMDB2[2]	RMDB2[1]	RMDB2[0]
		RMDB1[7]	RMDB1[6]	RMDB1[5]	RMDB1[4]	RMDB1[3]	RMDB1[2]	RMDB1[1]	RMDB1[0]
		RMDB0[7]	RMDB0[6]	RMDB0[5]	RMDB0[4]	RMDB0[3]	RMDB0[2]	RMDB0[1]	RMDB0[0]
	RSCAN0RMDf1q (q = 0 to 79)	RMDB7[7]	RMDB7[6]	RMDB7[5]	RMDB7[4]	RMDB7[3]	RMDB7[2]	RMDB7[1]	RMDB7[0]
		RMDB6[7]	RMDB6[6]	RMDB6[5]	RMDB6[4]	RMDB6[3]	RMDB6[2]	RMDB6[1]	RMDB6[0]
		RMDB5[7]	RMDB5[6]	RMDB5[5]	RMDB5[4]	RMDB5[3]	RMDB5[2]	RMDB5[1]	RMDB5[0]
		RMDB4[7]	RMDB4[6]	RMDB4[5]	RMDB4[4]	RMDB4[3]	RMDB4[2]	RMDB4[1]	RMDB4[0]
	RSCAN0RFIDx (x = 0 to 7)	RFIDE	RFTRTR	-	RFID[28]	RFID[27]	RFID[26]	RFID[25]	RFID[24]
		RFID[23]	RFID[22]	RFID[21]	RFID[20]	RFID[19]	RFID[18]	RFID[17]	RFID[16]
		RFID[15]	RFID[14]	RFID[13]	RFID[12]	RFID[11]	RFID[10]	RFID[9]	RFID[8]
		RFID[7]	RFID[6]	RFID[5]	RFID[4]	RFID[3]	RFID[2]	RFID[1]	RFID[0]
	RSCAN0RFPTRx (x = 0 to 7)	RFIDLC[3]	RFIDLC[2]	RFIDLC[1]	RFIDLC[0]	RFPTR[11]	RFPTR[10]	RFPTR[9]	RFPTR[8]
		RFPTR[7]	RFPTR[6]	RFPTR[5]	RFPTR[4]	RFPTR[3]	RFPTR[2]	RFPTR[1]	RFPTR[0]
		RFTS[15]	RFTS[14]	RFTS[13]	RFTS[12]	RFTS[11]	RFTS[10]	RFTS[9]	RFTS[8]
		RFTS[7]	RFTS[6]	RFTS[5]	RFTS[4]	RFTS[3]	RFTS[2]	RFTS[1]	RFTS[0]
	RSCAN0RFDB0x (x = 0 to 7)	RFDB3[7]	RFDB3[6]	RFDB3[5]	RFDB3[4]	RFDB3[3]	RFDB3[2]	RFDB3[1]	RFDB3[0]
		RFDB2[7]	RFDB2[6]	RFDB2[5]	RFDB2[4]	RFDB2[3]	RFDB2[2]	RFDB2[1]	RFDB2[0]
		RFDB1[7]	RFDB1[6]	RFDB1[5]	RFDB1[4]	RFDB1[3]	RFDB1[2]	RFDB1[1]	RFDB1[0]
		RFDB0[7]	RFDB0[6]	RFDB0[5]	RFDB0[4]	RFDB0[3]	RFDB0[2]	RFDB0[1]	RFDB0[0]
	RSCAN0RFDB1x (x = 0 to 7)	RFDB7[7]	RFDB7[6]	RFDB7[5]	RFDB7[4]	RFDB7[3]	RFDB7[2]	RFDB7[1]	RFDB7[0]
		RFDB6[7]	RFDB6[6]	RFDB6[5]	RFDB6[4]	RFDB6[3]	RFDB6[2]	RFDB6[1]	RFDB6[0]
		RFDB5[7]	RFDB5[6]	RFDB5[5]	RFDB5[4]	RFDB5[3]	RFDB5[2]	RFDB5[1]	RFDB5[0]
		RFDB4[7]	RFDB4[6]	RFDB4[5]	RFDB4[4]	RFDB4[3]	RFDB4[2]	RFDB4[1]	RFDB4[0]
	RSCAN0CFIDk (k = 0 to 14)	CFIDE	CFRTR	THLEN	CFID[28]	CFID[27]	CFID[26]	CFID[25]	CFID[24]
		CFID[23]	CFID[22]	CFID[21]	CFID[20]	CFID[19]	CFID[18]	CFID[17]	CFID[16]
		CFID[15]	CFID[14]	CFID[13]	CFID[12]	CFID[11]	CFID[10]	CFID[9]	CFID[8]
		CFID[7]	CFID[6]	CFID[5]	CFID[4]	CFID[3]	CFID[2]	CFID[1]	CFID[0]
	RSCAN0CFPTRk (k = 0 to 14)	CFDLC[3]	CFDLC[2]	CFDLC[1]	CFDLC[0]	CFPTR[11]	CFPTR[10]	CFPTR[9]	CFPTR[8]
		CFPTR[7]	CFPTR[6]	CFPTR[5]	CFPTR[4]	CFPTR[3]	CFPTR[2]	CFPTR[1]	CFPTR[0]
		CFTS[15]	CFTS[14]	CFTS[13]	CFTS[12]	CFTS[11]	CFTS[10]	CFTS[9]	CFTS[8]
		CFTS[7]	CFTS[6]	CFTS[5]	CFTS[4]	CFTS[3]	CFTS[2]	CFTS[1]	CFTS[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
CAN interface	RSCANOCDF0k (k = 0 to 14)	CFDB3[7]	CFDB3[6]	CFDB3[5]	CFDB3[4]	CFDB3[3]	CFDB3[2]	CFDB3[1]	CFDB3[0]	
		CFDB2[7]	CFDB2[6]	CFDB2[5]	CFDB2[4]	CFDB2[3]	CFDB2[2]	CFDB2[1]	CFDB2[0]	
		CFDB1[7]	CFDB1[6]	CFDB1[5]	CFDB1[4]	CFDB1[3]	CFDB1[2]	CFDB1[1]	CFDB1[0]	
		CFDB0[7]	CFDB0[6]	CFDB0[5]	CFDB0[4]	CFDB0[3]	CFDB0[2]	CFDB0[1]	CFDB0[0]	
	RSCANOCDF1k (k = 0 to 14)	CFDB7[7]	CFDB7[6]	CFDB7[5]	CFDB7[4]	CFDB7[3]	CFDB7[2]	CFDB7[1]	CFDB7[0]	
		CFDB6[7]	CFDB6[6]	CFDB6[5]	CFDB6[4]	CFDB6[3]	CFDB6[2]	CFDB6[1]	CFDB6[0]	
		CFDB5[7]	CFDB5[6]	CFDB5[5]	CFDB5[4]	CFDB5[3]	CFDB5[2]	CFDB5[1]	CFDB5[0]	
		CFDB4[7]	CFDB4[6]	CFDB4[5]	CFDB4[4]	CFDB4[3]	CFDB4[2]	CFDB4[1]	CFDB4[0]	
	RSCAN0TMIDp (p = 0 to 79)	TMIDE	TMRTR	THLEN	TMID[28]	TMID[27]	TMID[26]	TMID[25]	TMID[24]	
		TMID[23]	TMID[22]	TMID[21]	TMID[20]	TMID[19]	TMID[18]	TMID[17]	TMID[16]	
		TMID[15]	TMID[14]	TMID[13]	TMID[12]	TMID[11]	TMID[10]	TMID[9]	TMID[8]	
		TMID[7]	TMID[6]	TMID[5]	TMID[4]	TMID[3]	TMID[2]	TMID[1]	TMID[0]	
	RSCAN0TMPTRp (p = 0 to 79)	TM DLC[3]	TM DLC[2]	TM DLC[1]	TM DLC[0]	-	-	-	-	
		TM PTR[7]	TM PTR[6]	TM PTR[5]	TM PTR[4]	TM PTR[3]	TM PTR[2]	TM PTR[1]	TM PTR[0]	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
	RSCAN0TMDB0p (p = 0 to 79)	TMDB3[7]	TMDB3[6]	TMDB3[5]	TMDB3[4]	TMDB3[3]	TMDB3[2]	TMDB3[1]	TMDB3[0]	
		TMDB2[7]	TMDB2[6]	TMDB2[5]	TMDB2[4]	TMDB2[3]	TMDB2[2]	TMDB2[1]	TMDB2[0]	
		TMDB1[7]	TMDB1[6]	TMDB1[5]	TMDB1[4]	TMDB1[3]	TMDB1[2]	TMDB1[1]	TMDB1[0]	
		TMDB0[7]	TMDB0[6]	TMDB0[5]	TMDB0[4]	TMDB0[3]	TMDB0[2]	TMDB0[1]	TMDB0[0]	
	RSCAN0TMDB1p (p = 0 to 79)	TMDB7[7]	TMDB7[6]	TMDB7[5]	TMDB7[4]	TMDB7[3]	TMDB7[2]	TMDB7[1]	TMDB7[0]	
		TMDB6[7]	TMDB6[6]	TMDB6[5]	TMDB6[4]	TMDB6[3]	TMDB6[2]	TMDB6[1]	TMDB6[0]	
		TMDB5[7]	TMDB5[6]	TMDB5[5]	TMDB5[4]	TMDB5[3]	TMDB5[2]	TMDB5[1]	TMDB5[0]	
		TMDB4[7]	TMDB4[6]	TMDB4[5]	TMDB4[4]	TMDB4[3]	TMDB4[2]	TMDB4[1]	TMDB4[0]	
	RSCAN0THLACm (m = 0 to 4)	-	-	-	-	-	-	-	-	
		TID[7]	TID[6]	TID[5]	TID[4]	TID[3]	TID[2]	TID[1]	TID[0]	
		-	BN[3]	BN[2]	BN[1]	BN[0]	BT[2]	BT[1]	BT[0]	
		-	-	-	-	-	-	-	-	
	IEBus controller	IEBB0BCR	IEBB0PW	IEBB0MSRQ	IEBB0ALRQ	IEBB0STXE	IEBB0SRXE	-	-	-
		IEBB0PSR	IEBB0CLKE	IEBB0CMD	-	-	-	-	-	
		IEBB0UAR	-	-	-	-	-	-	-	
		IEBB0SAR	-	-	-	-	-	-	-	
		IEBB0PAR	-	-	-	-	-	-	-	
		IEBB0RSA	-	-	-	-	-	-	-	
		IEBB0CDR	-	-	-	-	IEBB0SLCD3	IEBB0SLCD2	IEBB0SLCD1	IEBB0SLCD0
		IEBB0TCD	-	-	-	-	IEBB0SLTD3	IEBB0SLTD2	IEBB0SLTD1	IEBB0SLTD0
		IEBB0RCD	-	-	-	-	IEBB0SLRD3	IEBB0SLRD2	IEBB0SLRD1	IEBB0SLRD0
		IEBB0DLR	-	-	-	-	-	-	-	-
		IEBB0TDL	-	-	-	-	-	-	-	-
		IEBB0RDL	-	-	-	-	-	-	-	-
IEBB0CKS		-	-	-	IEBB0PRS	-	IEBB0BRS2	IEBB0BRS1	IEBB0BRS0	
IEBB0TMS		IEBB0FMDE	IEBB0SLR1	IEBB0SLR10	IEBB0SLT11	IEBB0SLT10	IEBB0ALC2	IEBB0ALC1	IEBB0ALC0	
IEBB0PCR		IEBB0CRPT	IEBB0CTPT	-	-	-	-	-	-	
IEBB0BSR		IEBB0RFLF	IEBB0OVR	-	IEBB0SRFP4	IEBB0SRFP3	IEBB0SRFP2	IEBB0SRFP1	IEBB0SRFP0	
		IEBB0TFLF	IEBB0OVW	-	IEBB0STFP4	IEBB0STFP3	IEBB0STFP2	IEBB0STFP1	IEBB0STFP0	
IEBB0SSR		-	-	-	IEBB0SSLF	-	IEBB0STLF	IEBB0SRXF	IEBB0STXF	
IEBB0USR		-	IEBB0SRQF	IEBB0ARBF	IEBB0ALTF	IEBB0ACKF	IEBB0LCKF	-	-	
IEBB0ISR		-	IEBB0IEBE	IEBB0STRF	IEBB0STSF	IEBB0ETRF	IEBB0EFMF	-	IEBB0FOVE	
IEBB0ESR		IEBB0TIME	IEBB0PARE	IEBB0NACE	IEBB0UNRE	IEBB0OVRE	-	IEBB0ABTE	IEBB0TRDE	
IEBB0FSR		IEBB0TRF	IEBB0TRF	-	-	-	-	IEBB0SFS1	IEBB0SFS0	
IEBB0SCR		-	-	-	-	-	-	-	-	
IEBB0CCR		-	-	-	-	-	-	-	-	
IEBB0STC0		IEBB0CLTM	IEBB0CLPA	IEBB0CLNC	IEBB0CLUR	IEBB0CLOV	-	IEBB0CLAB	IEBB0CLTR	
IEBB0STC1		-	-	-	-	-	-	-	IEBB0CLFF	
IEBB0DR		-	-	-	-	-	-	-	-	

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Renesas SPDIF interface	TLCA	-	-	-	-	-	-	-	-
	TRCA	-	-	-	-	-	-	-	-
	TLCS	-	-	CLAC[1]	CLAC[0]	FS[3]	FS[2]	FS[1]	FS[0]
		CHNO[3]	CHNO[2]	CHNO[1]	CHNO[0]	SRCNO[3]	SRCNO[2]	SRCNO[1]	SRCNO[0]
		CATCD[7]	CATCD[6]	CATCD[5]	CATCD[4]	CATCD[3]	CATCD[2]	CATCD[1]	CATCD[0]
		-	-	CTL[4]	CTL[3]	CTL[2]	CTL[1]	CTL[0]	-
	TRCS	-	-	CLAC[1]	CLAC[0]	FS[3]	FS[2]	FS[1]	FS[0]
		CHNO[3]	CHNO[2]	CHNO[1]	CHNO[0]	SRCNO[3]	SRCNO[2]	SRCNO[1]	SRCNO[0]
		CATCD[7]	CATCD[6]	CATCD[5]	CATCD[4]	CATCD[3]	CATCD[2]	CATCD[1]	CATCD[0]
		-	-	CTL[4]	CTL[3]	CTL[2]	CTL[1]	CTL[0]	-
	TUI								
	RLCA	-	-	-	-	-	-	-	-
	RRCA	-	-	-	-	-	-	-	-
	RLCS	-	-	CLAC[1]	CLAC[0]	FS[3]	FS[2]	FS[1]	FS[0]
		CHNO[3]	CHNO[2]	CHNO[1]	CHNO[0]	SRCNO[3]	SRCNO[2]	SRCNO[1]	SRCNO[0]
		CATCD[7]	CATCD[6]	CATCD[5]	CATCD[4]	CATCD[3]	CATCD[2]	CATCD[1]	CATCD[0]
		-	-	CTL[4]	CTL[3]	CTL[2]	CTL[1]	CTL[0]	-
	RRCS	-	-	CLAC[1]	CLAC[0]	FS[3]	FS[2]	FS[1]	FS[0]
		CHNO[3]	CHNO[2]	CHNO[1]	CHNO[0]	SRCNO[3]	SRCNO[2]	SRCNO[1]	SRCNO[0]
		CATCD[7]	CATCD[6]	CATCD[5]	CATCD[4]	CATCD[3]	CATCD[2]	CATCD[1]	CATCD[0]
		-	-	CTL[4]	CTL[3]	CTL[2]	CTL[1]	CTL[0]	-
	RUI								
	CTRL	-	-	-	CKS	-	PB	RASS[1]	RASS[0]
		TASS[1]	TASS[0]	RDE	TDE	NCSI	AOS	RME	TME
		REIE	TEIE	UBOI	UBUI	CREI	PAEI	PREI	CSEI
		ABOI	ABUI	RUII	TUII	RCSI	RCBI	TCSI	TCBI
	STAT	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	CMD
		RIS	TIS	UBO	UBU	CE	PARE	PREE	CSE
		ABO	ABU	RUIR	TUIR	CSRX	CBRX	CSTX	CBTX
	TDAD	-	-	-	-	-	-	-	-
	RDAD	-	-	-	-	-	-	-	-

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
CD-ROM decoder	CROMEN	SUBC_EN	CROM_EN	CROM_STP	-	-	-	-	-
	CROMSY0	SY_AUT	SY_IEN	SY_DEN	-	-	-	-	-
	CROMCTL0	MD_DESC	-	MD_AUTO	MD_AUTOS1	MD_AUTOS2	MD_SEC[2]	MD_SEC[1]	MD_SEC[0]
	CROMCTL1	M2F2EDC	MD_DEC[2]	MD_DEC[1]	MD_DEC[0]	-	-	MD_PQREP[1]	MD_PQREP[0]
	CROMCTL3	STP_ECC	STP_EDC	-	STP_MD	STP_MIN	-	-	-
	CROMCTL4	-	LINK2	-	EROSEL	NO_ECC	-	-	-
	CROMCTL5	-	-	-	-	-	-	-	MSF_LBA_SEL
	CROMST0	-	-	ST_SYIL	ST_SYNO	ST_BLKs	ST_BLKL	ST_SECS	ST_SECL
	CROMST1	-	-	-	-	ER2_HEAD0	ER2_HEAD1	ER2_HEAD2	ER2_HEAD3
	CROMST3	ER2_SHEAD0	ER2_SHEAD1	ER2_SHEAD2	ER2_SHEAD3	ER2_SHEAD4	ER2_SHEAD5	ER2_SHEAD6	ER2_SHEAD7
	CROMST4	NG_MD	NG_MDCMP1	NG_MDCMP2	NG_MDCMP3	NG_MDCMP4	NG_MDDEF	NG_MDTIM1	NG_MDTIM2
	CROMST5	ST_AMD[2]	ST_AMD[1]	ST_AMD[0]	ST_MDx	LINK_ON	LINK_DET	LINK_SDET	LINK_OUT1
	CROMST6	ST_ERR	-	ST_ECCABT	ST_ECCNG	ST_ECCP	ST_ECCQ	ST_EDC1	ST_EDC2
	CBUFST0	BUF_REF	BUF_ACT	-	-	-	-	-	-
	CBUFST1	BUF_ECC	BUF_EDC	-	BUF_MD	BUF_MIN	-	-	-
	CBUFST2	BUF_NG	-	-	-	-	-	-	-
	HEAD00	HEAD00[7]	HEAD00[6]	HEAD00[5]	HEAD00[4]	HEAD00[3]	HEAD00[2]	HEAD00[1]	HEAD00[0]
	HEAD01	HEAD01[7]	HEAD01[6]	HEAD01[5]	HEAD01[4]	HEAD01[3]	HEAD01[2]	HEAD01[1]	HEAD01[0]
	HEAD02	HEAD02[7]	HEAD02[6]	HEAD02[5]	HEAD02[4]	HEAD02[3]	HEAD02[2]	HEAD02[1]	HEAD02[0]
	HEAD03	HEAD03[7]	HEAD03[6]	HEAD03[5]	HEAD03[4]	HEAD03[3]	HEAD03[2]	HEAD03[1]	HEAD03[0]
	SHEAD00	SHEAD00[7]	SHEAD00[6]	SHEAD00[5]	SHEAD00[4]	SHEAD00[3]	SHEAD00[2]	SHEAD00[1]	SHEAD00[0]
	SHEAD01	SHEAD01[7]	SHEAD01[6]	SHEAD01[5]	SHEAD01[4]	SHEAD01[3]	SHEAD01[2]	SHEAD01[1]	SHEAD01[0]
	SHEAD02	SHEAD02[7]	SHEAD02[6]	SHEAD02[5]	SHEAD02[4]	SHEAD02[3]	SHEAD02[2]	SHEAD02[1]	SHEAD02[0]
	SHEAD03	SHEAD03[7]	SHEAD03[6]	SHEAD03[5]	SHEAD03[4]	SHEAD03[3]	SHEAD03[2]	SHEAD03[1]	SHEAD03[0]
	SHEAD04	SHEAD04[7]	SHEAD04[6]	SHEAD04[5]	SHEAD04[4]	SHEAD04[3]	SHEAD04[2]	SHEAD04[1]	SHEAD04[0]
	SHEAD05	SHEAD05[7]	SHEAD05[6]	SHEAD05[5]	SHEAD05[4]	SHEAD05[3]	SHEAD05[2]	SHEAD05[1]	SHEAD05[0]
	SHEAD06	SHEAD06[7]	SHEAD06[6]	SHEAD06[5]	SHEAD06[4]	SHEAD06[3]	SHEAD06[2]	SHEAD06[1]	SHEAD06[0]
	SHEAD07	SHEAD07[7]	SHEAD07[6]	SHEAD07[5]	SHEAD07[4]	SHEAD07[3]	SHEAD07[2]	SHEAD07[1]	SHEAD07[0]
	HEAD20	HEAD20[7]	HEAD20[6]	HEAD20[5]	HEAD20[4]	HEAD20[3]	HEAD20[2]	HEAD20[1]	HEAD20[0]
	HEAD21	HEAD21[7]	HEAD21[6]	HEAD21[5]	HEAD21[4]	HEAD21[3]	HEAD21[2]	HEAD21[1]	HEAD21[0]
	HEAD22	HEAD22[7]	HEAD22[6]	HEAD22[5]	HEAD22[4]	HEAD22[3]	HEAD22[2]	HEAD22[1]	HEAD22[0]
	HEAD23	HEAD23[7]	HEAD23[6]	HEAD23[5]	HEAD23[4]	HEAD23[3]	HEAD23[2]	HEAD23[1]	HEAD23[0]
	SHEAD20	SHEAD20[7]	SHEAD20[6]	SHEAD20[5]	SHEAD20[4]	SHEAD20[3]	SHEAD20[2]	SHEAD20[1]	SHEAD20[0]
	SHEAD21	SHEAD21[7]	SHEAD21[6]	SHEAD21[5]	SHEAD21[4]	SHEAD21[3]	SHEAD21[2]	SHEAD21[1]	SHEAD21[0]
	SHEAD22	SHEAD22[7]	SHEAD22[6]	SHEAD22[5]	SHEAD22[4]	SHEAD22[3]	SHEAD22[2]	SHEAD22[1]	SHEAD22[0]
	SHEAD23	SHEAD23[7]	SHEAD23[6]	SHEAD23[5]	SHEAD23[4]	SHEAD23[3]	SHEAD23[2]	SHEAD23[1]	SHEAD23[0]
	SHEAD24	SHEAD24[7]	SHEAD24[6]	SHEAD24[5]	SHEAD24[4]	SHEAD24[3]	SHEAD24[2]	SHEAD24[1]	SHEAD24[0]
	SHEAD25	SHEAD25[7]	SHEAD25[6]	SHEAD25[5]	SHEAD25[4]	SHEAD25[3]	SHEAD25[2]	SHEAD25[1]	SHEAD25[0]
	SHEAD26	SHEAD26[7]	SHEAD26[6]	SHEAD26[5]	SHEAD26[4]	SHEAD26[3]	SHEAD26[2]	SHEAD26[1]	SHEAD26[0]
	SHEAD27	SHEAD27[7]	SHEAD27[6]	SHEAD27[5]	SHEAD27[4]	SHEAD27[3]	SHEAD27[2]	SHEAD27[1]	SHEAD27[0]
	CBUFCTL0	CBUF_AUT	CBUF_EN	-	CBUF_MD[1]	CBUF_MD[0]	CBUF_TS	CBUF_Q	-
	CBUFCTL1	BS_MIN[7]	BS_MIN[6]	BS_MIN[5]	BS_MIN[4]	BS_MIN[3]	BS_MIN[2]	BS_MIN[1]	BS_MIN[0]
	CBUFCTL2	BS_SEC[7]	BS_SEC[6]	BS_SEC[5]	BS_SEC[4]	BS_SEC[3]	BS_SEC[2]	BS_SEC[1]	BS_SEC[0]
	CBUFCTL3	BS_FRM[7]	BS_FRM[6]	BS_FRM[5]	BS_FRM[4]	BS_FRM[3]	BS_FRM[2]	BS_FRM[1]	BS_FRM[0]
	CROMST0M	-	-	ST_SYILM	ST_SYNOM	ST_BLKSM	ST_BLKLM	ST_SECSM	ST_SECLM
	ROMDECRST	LOGICRST	RAMRST	-	-	-	-	-	-
	RSTSTAT	RAMCLRST	-	-	-	-	-	-	-
	SSI	BYTEND	BITEND	BUFEND0[1]	BUFEND0[0]	BUFEND1[1]	BUFEND1[0]	-	-
	INTHOLD	ISEC	ITARG	ISY	IERR	IBUF	IREADY	-	-
	INHINT	INHISEC	INHITARG	INHISY	INHIERR	INHIBUF	INHIREADY	PREINHREQDM	PREINHIREADY
	STRMDIN0	STRMDIN[31]	STRMDIN[30]	STRMDIN[29]	STRMDIN[28]	STRMDIN[27]	STRMDIN[26]	STRMDIN[25]	STRMDIN[24]
		STRMDIN[23]	STRMDIN[22]	STRMDIN[21]	STRMDIN[20]	STRMDIN[19]	STRMDIN[18]	STRMDIN[17]	STRMDIN[16]
	STRMDIN2	STRMDIN[15]	STRMDIN[14]	STRMDIN[13]	STRMDIN[12]	STRMDIN[11]	STRMDIN[10]	STRMDIN[9]	STRMDIN[8]
		STRMDIN[7]	STRMDIN[6]	STRMDIN[5]	STRMDIN[4]	STRMDIN[3]	STRMDIN[2]	STRMDIN[1]	STRMDIN[0]
	STRMDOUT0	STRMDOUT[15]	STRMDOUT[14]	STRMDOUT[13]	STRMDOUT[12]	STRMDOUT[11]	STRMDOUT[10]	STRMDOUT[9]	STRMDOUT[8]
		STRMDOUT[7]	STRMDOUT[6]	STRMDOUT[5]	STRMDOUT[4]	STRMDOUT[3]	STRMDOUT[2]	STRMDOUT[1]	STRMDOUT[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
LIN interface channel 0 LIN master	RLN30LWBR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		NSPB[3]	NSPB[2]	NSPB[1]	NSPB[0]	LPRS[2]	LPRS[1]	LPRS[0]	LWBR0
	RLN30LBRP0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		LBRP0[7]	LBRP0[6]	LBRP0[5]	LBRP0[4]	LBRP0[3]	LBRP0[2]	LBRP0[1]	LBRP0[0]
	RLN30LBRP1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		LBRP1[7]	LBRP1[6]	LBRP1[5]	LBRP1[4]	LBRP1[3]	LBRP1[2]	LBRP1[1]	LBRP1[0]
	RLN30LSTC	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	LSTM
	RLN30LMD	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	LRDNFS	LIOS	LCKS[1]	LCKS[0]	LMD[1]	LMD[0]
	RLN30LBFC	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	BDT[1]	BDT[0]	BLT[3]	BLT[2]	BLT[1]	BLT[0]
	RLN30LSC	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	IBS[1]	IBS[0]	-	IBHS[2]	IBHS[1]	IBHS[0]
	RLN30LWUP	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		WUTL[3]	WUTL[2]	WUTL[1]	WUTL[0]	-	-	-	-
	RLN30LIE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	SHIE	ERRIE	FRCIE	FTCIE
	RLN30LEDE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		LTES	-	-	-	FERE	FTERE	PBERE	BERE
	RLN30LCUC	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	OM1	OM0
	RLN30LTRC	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	RTS	FTS
	RLN30LMST	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	OMM1	OMM0

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
LIN interface channel 0 LIN master	RLN30LST	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		HTRC	D1RC	-	-	ERR	-	FRC	FTC	
	RLN30LEST	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		RPER	-	CSER	-	FER	FTER	PBER	BER	
	RLN30LDFC	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		LSS	FSM	CSM	RFT	RFDL[3]	RFDL[2]	RFDL[1]	RFDL[0]	
	RLN30LIDB	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		IDP1	IDP0	ID[5]	ID[4]	ID[3]	ID[2]	ID[1]	ID[0]	
	RLN30LCBR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		CKSM[7]	CKSM[6]	CKSM[5]	CKSM[4]	CKSM[3]	CKSM[2]	CKSM[1]	CKSM[0]	
	RLN30LDBR1	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]	
	RLN30LDBR2	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]	
	RLN30LDBR3	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]	
	RLN30LDBR4	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]	
	RLN30LDBR5	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]	
	RLN30LDBR6	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]	
	RLN30LDBR7	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]	
	RLN30LDBR8	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]	
	LIN interface channel 1 LIN master	RLN31LWBR	-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			NSPB[3]	NSPB[2]	NSPB[1]	NSPB[0]	LPRS[2]	LPRS[1]	LPRS[0]	LWBR0



Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
LIN interface channel 1 LIN master	RLN31LBRP0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		LBRP0[7]	LBRP0[6]	LBRP0[5]	LBRP0[4]	LBRP0[3]	LBRP0[2]	LBRP0[1]	LBRP0[0]
	RLN31LBRP1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		LBRP1[7]	LBRP1[6]	LBRP1[5]	LBRP1[4]	LBRP1[3]	LBRP1[2]	LBRP1[1]	LBRP1[0]
	RLN31LSTC	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	LSTM
	RLN31LMD	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	LRDNFS	LIOS	LCKS[1]	LCKS[0]	LMD[1]	LMD[0]
	RLN31LBFC	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	BDT[1]	BDT[0]	BLT[3]	BLT[2]	BLT[1]	BLT[0]
	RLN31LSC	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	IBS[1]	IBS[0]	-	IBHS[2]	IBHS[1]	IBHS[0]
	RLN31LWUP	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		WUTL[3]	WUTL[2]	WUTL[1]	WUTL[0]	-	-	-	-
	RLN31LIE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	SHIE	ERRIE	FRCIE	FTCIE
	RLN31LEDE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		LTES	-	-	-	FERE	FTERE	PBERE	BERE
	RLN31LCUC	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	OM1	OM0
	RLN31LTRC	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	RTS	FTS
	RLN31LMST	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	OMM1	OMM0
	RLN31LST	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		HTRC	D1RC	-	-	ERR	-	FRC	FTC
	RLN31LEST	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		RPER	-	CSER	-	FER	FTER	PBER	BER

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
LIN interface channel 1 LIN master	RLN31LDFC	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		LSS	FSM	CSM	RFT	RFDL[3]	RFDL[2]	RFDL[1]	RFDL[0]	
	RLN31LIDB	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		IDP1	IDP0	ID[5]	ID[4]	ID[3]	ID[2]	ID[1]	ID[0]	
	RLN31LCBR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		CKSM[7]	CKSM[6]	CKSM[5]	CKSM[4]	CKSM[3]	CKSM[2]	CKSM[1]	CKSM[0]	
	RLN31LDBR1	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]	
	RLN31LDBR2	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]	
	RLN31LDBR3	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]	
	RLN31LDBR4	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]	
	RLN31LDBR5	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]	
	RLN31LDBR6	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]	
	RLN31LDBR7	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]	
	RLN31LDBR8	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]	
	LIN interface channel 0 LIN slave	RLN30LWBR	-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			NSPB[3]	NSPB[2]	NSPB[1]	NSPB[0]	LPRS[2]	LPRS[1]	LPRS[0]	-
		RLN30LBRP0	-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			BRP[7]	BRP[6]	BRP[5]	BRP[4]	BRP[3]	BRP[2]	BRP[1]	BRP[0]
		RLN30LBRP1	-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			BRP[15]	BRP[14]	BRP[13]	BRP[12]	BRP[11]	BRP[10]	BRP[9]	BRP[8]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
LIN interface channel 0 LIN slave	RLN30LSTC	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	LSTM
	RLN30LMD	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	LRDNFS	LIOS	-	-	LMD[1]	LMD[0]
	RLN30LBFC	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	LBLT
	RLN30LSC	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	IBS[1]	IBS[0]	-	IBHS[2]	IBHS[1]	IBHS[0]
	RLN30LWUP	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	WUTL[3]	WUTL[2]	WUTL[1]	WUTL[0]	-	-	-
	RLN30LIE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	SHIE	ERRIE	FRCIE	FTCIE
	RLN30LEDE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	LTES	IPERE	-	SFERE	FERE	-	BERE
	RLN30LCUC	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	OM1	OM0
	RLN30LTRC	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	LNRR	RTS
	RLN30LMST	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	OMM1	OMM0
	RLN30LST	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	HTRC	D1RC	-	-	ERR	-	FRC
	RLN30LEST	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	RPER	IPER	CSER	SFER	FER	TER	-
	RLN30LDFC	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	LSS	-	LCS	RCDS	RFDL[3]	RFDL[2]	RFDL[1]
	RLN30LIDB	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	IDP1	IDP0	ID[5]	ID[4]	ID[3]	ID[2]	ID[1]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
LIN interface channel 0 LIN slave	RLN30LCBR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		CKSM[7]	CKSM[6]	CKSM[5]	CKSM[4]	CKSM[3]	CKSM[2]	CKSM[1]	CKSM[0]	
	RLN30LDBR1	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]	
	RLN30LDBR2	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]	
	RLN30LDBR3	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]	
	RLN30LDBR4	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]	
	RLN30LDBR5	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]	
	RLN30LDBR6	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]	
	RLN30LDBR7	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]	
	RLN30LDBR8	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]	
	LIN interface channel 1 LIN slave	RLN31LWBR	-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			NSPB[3]	NSPB[2]	NSPB[1]	NSPB[0]	LPRS[2]	LPRS[1]	LPRS[0]	-
		RLN31LBRP0	-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			BRP[7]	BRP[6]	BRP[5]	BRP[4]	BRP[3]	BRP[2]	BRP[1]	BRP[0]
		RLN31LBRP1	-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			BRP[15]	BRP[14]	BRP[13]	BRP[12]	BRP[11]	BRP[10]	BRP[9]	BRP[8]
RLN31LSTC		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	LSTM	
RLN31LMD	-	-	-	-	-	-	-	-		
	-	-	-	-	-	-	-	-		
	-	-	-	-	-	-	-	-		
	-	-	LRDNFS	LIOS	-	-	-	LMD[1]	LMD[0]	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
LIN interface channel 1 LIN slave	RLN31LBFC	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	LBLT
	RLN31LSC	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	IBS[1]	IBS[0]	-	IBHS[2]	IBHS[1]	IBHS[0]
	RLN31LWUP	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		WUTL[3]	WUTL[2]	WUTL[1]	WUTL[0]	-	-	-	-
	RLN31LIE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	SHIE	ERRIE	FRCIE	FTCIE
	RLN31LEDE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		LTES	IPERE	-	SFERE	FERE	-	-	BERE
	RLN31LCUC	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	OM1	OM0
	RLN31LTRC	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	LNRR	RTS	FTS
	RLN31LMST	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	OMM1	OMM0
	RLN31LST	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		HTRC	D1RC	-	-	ERR	-	FRC	FTC
	RLN31LEST	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		RPER	IPER	CSER	SFER	FER	TER	-	BER
	RLN31LDFC	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		LSS	-	LCS	RCDS	RFDL[3]	RFDL[2]	RFDL[1]	RFDL[0]
	RLN31LIDB	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		IDP1	IDP0	ID[5]	ID[4]	ID[3]	ID[2]	ID[1]	ID[0]
	RLN31LCBR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		CKSM[7]	CKSM[6]	CKSM[5]	CKSM[4]	CKSM[3]	CKSM[2]	CKSM[1]	CKSM[0]
	RLN31LDBR1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
LIN interface channel 1 LIN slave	RLN31LDBR2	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]	
	RLN31LDBR3	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]	
	RLN31LDBR4	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]	
	RLN31LDBR5	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]	
	RLN31LDBR6	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]	
	RLN31LDBR7	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]	
	RLN31LDBR8	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]	
	LIN interface channel 0 UART	RLN30LWBR	-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			NSPB[3]	NSPB[2]	NSPB[1]	NSPB[0]	LPRS[2]	LPRS[1]	LPRS[0]	-
		RLN30LBRP0	-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			BRP[7]	BRP[6]	BRP[5]	BRP[4]	BRP[3]	BRP[2]	BRP[1]	BRP[0]
RLN30LBRP1		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		BRP[15]	BRP[14]	BRP[13]	BRP[12]	BRP[11]	BRP[10]	BRP[9]	BRP[8]	
RLN30LMD		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	LRDNFS	-	-	-	LMD[1]	LMD[0]	
RLN30LBFC		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	UTPS	URPS	UPS[1]	UPS[0]	USBLS	UBOS	UBLS	
RLN30LSC		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	IBS[1]	IBS[0]	-	-	-	-	
RLN30LEDE	-	-	-	-	-	-	-	-		
	-	-	-	-	-	-	-	-		
	-	-	-	-	-	-	-	-		
	-	-	-	-	FERE	OERE	-	BERE		

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
LIN interface channel 0 UART	RLN30LCUC	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	OM0
	RLN30LTRC	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	RTS	-
	RLN30LMST	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	OMM0
	RLN30LST	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	URS	UTS	ERR	-	-	FTC
	RLN30LEST	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	UPER	IDMT	EXBT	FER	OER	-	BER
	RLN30LDFC	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	UTSW	-	MDL[3]	MDL[2]	MDL[1]	MDL[0]
	RLN30LIDB	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		ID[7]	ID[6]	ID[5]	ID[4]	ID[3]	ID[2]	ID[1]	ID[0]
	RLN30LUDB	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		UDB[7]	UDB[6]	UDB[5]	UDB[4]	UDB[3]	UDB[2]	UDB[1]	UDB[0]
	RLN30LDBR1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]
	RLN30LDBR2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]
	RLN30LDBR3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]
	RLN30LDBR4	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]
	RLN30LDBR5	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]
	RLN30LDBR6	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
LIN interface channel 0 UART	RLN30LDBR7	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]	
	RLN30LDBR8	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]	
	RLN30LUOER	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	UROE	UTOE	
	RLN30LUOR1	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	UECD	UTIGTS	UEBDCE	UEBDL	UEBE	
	RLN30LUTDR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	UTD[8]	
		UTD[7]	UTD[6]	UTD[5]	UTD[4]	UTD[3]	UTD[2]	UTD[1]	UTD[0]	
	RLN30LURDR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	URD[8]	
		URD[7]	URD[6]	URD[5]	URD[4]	URD[3]	URD[2]	URD[1]	URD[0]	
	RLN30LUWDR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	UWTD[8]	
		UWTD[7]	UWTD[6]	UWTD[5]	UWTD[4]	UWTD[3]	UWTD[2]	UWTD[1]	UWTD[0]	
	LIN interface channel 1 UART	RLN31LWBR	-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			NSPB[3]	NSPB[2]	NSPB[1]	NSPB[0]	LPRS[2]	LPRS[1]	LPRS[0]	-
		RLN31LBRP0	-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			BRP[7]	BRP[6]	BRP[5]	BRP[4]	BRP[3]	BRP[2]	BRP[1]	BRP[0]
RLN31LBRP1		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		BRP[15]	BRP[14]	BRP[13]	BRP[12]	BRP[11]	BRP[10]	BRP[9]	BRP[8]	
RLN31LMD		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	LRDNFS	-	-	-	LMD[1]	LMD[0]	
RLN31LBFC		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	UTPS	URPS	UPS[1]	UPS[0]	USBLS	UBOS	UBLS	
RLN31LSC		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	IBS[1]	IBS[0]	-	-	-	-	
RLN31LEDE	-	-	-	-	-	-	-	-		
	-	-	-	-	-	-	-	-		
	-	-	-	-	-	-	-	-		
	-	-	-	-	FERE	OERE	-	BERE		



Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
LIN interface channel 1 UART	RLN31LCUC	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	OM0
	RLN31LTRC	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	RTS	-
	RLN31LMST	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	OMM0
	RLN31LST	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	URS	UTS	ERR	-	-	FTC
	RLN31LEST	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	UPER	IDMT	EXBT	FER	OER	-	BER
	RLN31LDFC	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	UTSW	-	MDL[3]	MDL[2]	MDL[1]	MDL[0]
	RLN31LIDB	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		ID[7]	ID[6]	ID[5]	ID[4]	ID[3]	ID[2]	ID[1]	ID[0]
	RLN31LUDB	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		UDB[7]	UDB[6]	UDB[5]	UDB[4]	UDB[3]	UDB[2]	UDB[1]	UDB[0]
	RLN31LDBR1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]
	RLN31LDBR2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]
	RLN31LDBR3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]
	RLN31LDBR4	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]
	RLN31LDBR5	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]
	RLN31LDBR6	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
LIN interface channel 1 UART	RLN31LDBR7	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]
	RLN31LDBR8	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		LDB[7]	LDB[6]	LDB[5]	LDB[4]	LDB[3]	LDB[2]	LDB[1]	LDB[0]
	RLN31LUOER	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	UROE	UTOE
	RLN31LUOR1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	UECD	UTIGTS	UEBDCE	UEBDL	UEBE
	RLN31LUTDR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	UTD[8]
		UTD[7]	UTD[6]	UTD[5]	UTD[4]	UTD[3]	UTD[2]	UTD[1]	UTD[0]
	RLN31LURDR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	URD[8]
		URD[7]	URD[6]	URD[5]	URD[4]	URD[3]	URD[2]	URD[1]	URD[0]
	RLN31LUWDR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	UWTD[8]
		UWTD[7]	UWTD[6]	UWTD[5]	UWTD[4]	UWTD[3]	UWTD[2]	UWTD[1]	UWTD[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Ethernet controller	ARSTR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	ARST	
	ECMR0	-	-	-	-	-	-	TRCCM	-	-
		RCSC	-	DPAD	RZPF	ZPF	PFR	RXF	TXXF	-
		-	-	MCT	-	-	-	-	-	-
		-	RE	TE	-	-	-	DM	PRM	-
	ECSR0	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	PFROI	-	-	-	-	ICD
	ECSIPR0	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	PFROIIP	-	-	-	-	ICDIP
	PIR0	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	MDI	MDO	MMD	MDC	-
	MAHR0	MA[47]	MA[46]	MA[45]	MA[44]	MA[43]	MA[42]	MA[41]	MA[40]	-
		MA[39]	MA[38]	MA[37]	MA[36]	MA[35]	MA[34]	MA[33]	MA[32]	-
		MA[31]	MA[30]	MA[29]	MA[28]	MA[27]	MA[26]	MA[25]	MA[24]	-
		MA[23]	MA[22]	MA[21]	MA[20]	MA[19]	MA[18]	MA[17]	MA[16]	-
	MALR0	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		MA[15]	MA[14]	MA[13]	MA[12]	MA[11]	MA[10]	MA[9]	MA[8]	-
		MA[7]	MA[6]	MA[5]	MA[4]	MA[3]	MA[2]	MA[1]	MA[0]	-
	RFLR0	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	RFL[17]	RFL[16]	-
		RFL[15]	RFL[14]	RFL[13]	RFL[12]	RFL[11]	RFL[10]	RFL[9]	RFL[8]	-
		RFL[7]	RFL[6]	RFL[5]	RFL[4]	RFL[3]	RFL[2]	RFL[1]	RFL[0]	-
	CEFCR0	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		CEFC[15]	CEFC[14]	CEFC[13]	CEFC[12]	CEFC[11]	CEFC[10]	CEFC[9]	CEFC[8]	-
		CEFC[7]	CEFC[6]	CEFC[5]	CEFC[4]	CEFC[3]	CEFC[2]	CEFC[1]	CEFC[0]	-
	FRECR0	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		FREC[15]	FREC[14]	FREC[13]	FREC[12]	FREC[11]	FREC[10]	FREC[9]	FREC[8]	-
		FREC[7]	FREC[6]	FREC[5]	FREC[4]	FREC[3]	FREC[2]	FREC[1]	FREC[0]	-
	TSFCR0	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		TSFC[15]	TSFC[14]	TSFC[13]	TSFC[12]	TSFC[11]	TSFC[10]	TSFC[9]	TSFC[8]	-
		TSFC[7]	TSFC[6]	TSFC[5]	TSFC[4]	TSFC[3]	TSFC[2]	TSFC[1]	TSFC[0]	-
	TLFCR0	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		TLFC[15]	TLFC[14]	TLFC[13]	TLFC[12]	TLFC[11]	TLFC[10]	TLFC[9]	TLFC[8]	-
		TLFC[7]	TLFC[6]	TLFC[5]	TLFC[4]	TLFC[3]	TLFC[2]	TLFC[1]	TLFC[0]	-

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Ethernet controller	RFCR0	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		RFC[15]	RFC[14]	RFC[13]	RFC[12]	RFC[11]	RFC[10]	RFC[9]	RFC[8]	
	RFC[7]	RFC[6]	RFC[5]	RFC[4]	RFC[3]	RFC[2]	RFC[1]	RFC[0]		
	MAFCR0	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		MAFC[15]	MAFC[14]	MAFC[13]	MAFC[12]	MAFC[11]	MAFC[10]	MAFC[9]	MAFC[8]	
	MAFC[7]	MAFC[6]	MAFC[5]	MAFC[4]	MAFC[3]	MAFC[2]	MAFC[1]	MAFC[0]		
	APR0	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		AP[15]	AP[14]	AP[13]	AP[12]	AP[11]	AP[10]	AP[9]	AP[8]	
	AP[7]	AP[6]	AP[5]	AP[4]	AP[3]	AP[2]	AP[1]	AP[0]		
	MPR0	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		MP[15]	MP[14]	MP[13]	MP[12]	MP[11]	MP[10]	MP[9]	MP[8]	
	MP[7]	MP[6]	MP[5]	MP[4]	MP[3]	MP[2]	MP[1]	MP[0]		
	TPAUSER0	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		TPAUSE[15]	TPAUSE[14]	TPAUSE[13]	TPAUSE[12]	TPAUSE[11]	TPAUSE[10]	TPAUSE[9]	TPAUSE[8]	
	TPAUSE[7]	TPAUSE[6]	TPAUSE[5]	TPAUSE[4]	TPAUSE[3]	TPAUSE[2]	TPAUSE[1]	TPAUSE[0]		
	PFTCR0	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		PFTXC[15]	PFTXC[14]	PFTXC[13]	PFTXC[12]	PFTXC[11]	PFTXC[10]	PFTXC[9]	PFTXC[8]	
	PFTXC[7]	PFTXC[6]	PFTXC[5]	PFTXC[4]	PFTXC[3]	PFTXC[2]	PFTXC[1]	PFTXC[0]		
	PFRCR0	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		PFRXC[15]	PFRXC[14]	PFRXC[13]	PFRXC[12]	PFRXC[11]	PFRXC[10]	PFRXC[9]	PFRXC[8]	
	PFRXC[7]	PFRXC[6]	PFRXC[5]	PFRXC[4]	PFRXC[3]	PFRXC[2]	PFRXC[1]	PFRXC[0]		
	TSU_CTRST	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	CTRST
		-	-	-	-	-	-	-	-	-
	TSU_VTAG0	VTAG0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	VID0[11]	VID0[10]	VID0[9]	VID0[8]	
		VID0[7]	VID0[6]	VID0[5]	VID0[4]	VID0[3]	VID0[2]	VID0[1]	VID0[0]	
	TSU_ADSBSY	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	ADSBSY
	TSU_TEN	TEN0	TEN1	TEN2	TEN3	TEN4	TEN5	TEN6	TEN7	
		TEN8	TEN9	TEN10	TEN11	TEN12	TEN13	TEN14	TEN15	
		TEN16	TEN17	TEN18	TEN19	TEN20	TEN21	TEN22	TEN23	
		TEN24	TEN25	TEN26	TEN27	TEN28	TEN29	TEN30	TEN31	
	TSU_ADRH0	ADRH0[31]	ADRH0[30]	ADRH0[29]	ADRH0[28]	ADRH0[27]	ADRH0[26]	ADRH0[25]	ADRH0[24]	
		ADRH0[23]	ADRH0[22]	ADRH0[21]	ADRH0[20]	ADRH0[19]	ADRH0[18]	ADRH0[17]	ADRH0[16]	
		ADRH0[15]	ADRH0[14]	ADRH0[13]	ADRH0[12]	ADRH0[11]	ADRH0[10]	ADRH0[9]	ADRH0[8]	
		ADRH0[7]	ADRH0[6]	ADRH0[5]	ADRH0[4]	ADRH0[3]	ADRH0[2]	ADRH0[1]	ADRH0[0]	
	TSU_ADRH1	ADRH1[31]	ADRH1[30]	ADRH1[29]	ADRH1[28]	ADRH1[27]	ADRH1[26]	ADRH1[25]	ADRH1[24]	
		ADRH1[23]	ADRH1[22]	ADRH1[21]	ADRH1[20]	ADRH1[19]	ADRH1[18]	ADRH1[17]	ADRH1[16]	
		ADRH1[15]	ADRH1[14]	ADRH1[13]	ADRH1[12]	ADRH1[11]	ADRH1[10]	ADRH1[9]	ADRH1[8]	
		ADRH1[7]	ADRH1[6]	ADRH1[5]	ADRH1[4]	ADRH1[3]	ADRH1[2]	ADRH1[1]	ADRH1[0]	
	TSU_ADRH2	ADRH2[31]	ADRH2[30]	ADRH2[29]	ADRH2[28]	ADRH2[27]	ADRH2[26]	ADRH2[25]	ADRH2[24]	
		ADRH2[23]	ADRH2[22]	ADRH2[21]	ADRH2[20]	ADRH2[19]	ADRH2[18]	ADRH2[17]	ADRH2[16]	
		ADRH2[15]	ADRH2[14]	ADRH2[13]	ADRH2[12]	ADRH2[11]	ADRH2[10]	ADRH2[9]	ADRH2[8]	
		ADRH2[7]	ADRH2[6]	ADRH2[5]	ADRH2[4]	ADRH2[3]	ADRH2[2]	ADRH2[1]	ADRH2[0]	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Ethernet controller	TSU_ADRH3	ADRH3[31]	ADRH3[30]	ADRH3[29]	ADRH3[28]	ADRH3[27]	ADRH3[26]	ADRH3[25]	ADRH3[24]
		ADRH3[23]	ADRH3[22]	ADRH3[21]	ADRH3[20]	ADRH3[19]	ADRH3[18]	ADRH3[17]	ADRH3[16]
		ADRH3[15]	ADRH3[14]	ADRH3[13]	ADRH3[12]	ADRH3[11]	ADRH3[10]	ADRH3[9]	ADRH3[8]
		ADRH3[7]	ADRH3[6]	ADRH3[5]	ADRH3[4]	ADRH3[3]	ADRH3[2]	ADRH3[1]	ADRH3[0]
	TSU_ADRH4	ADRH4[31]	ADRH4[30]	ADRH4[29]	ADRH4[28]	ADRH4[27]	ADRH4[26]	ADRH4[25]	ADRH4[24]
		ADRH4[23]	ADRH4[22]	ADRH4[21]	ADRH4[20]	ADRH4[19]	ADRH4[18]	ADRH4[17]	ADRH4[16]
		ADRH4[15]	ADRH4[14]	ADRH4[13]	ADRH4[12]	ADRH4[11]	ADRH4[10]	ADRH4[9]	ADRH4[8]
		ADRH4[7]	ADRH4[6]	ADRH4[5]	ADRH4[4]	ADRH4[3]	ADRH4[2]	ADRH4[1]	ADRH4[0]
	TSU_ADRH5	ADRH5[31]	ADRH5[30]	ADRH5[29]	ADRH5[28]	ADRH5[27]	ADRH5[26]	ADRH5[25]	ADRH5[24]
		ADRH5[23]	ADRH5[22]	ADRH5[21]	ADRH5[20]	ADRH5[19]	ADRH5[18]	ADRH5[17]	ADRH5[16]
		ADRH5[15]	ADRH5[14]	ADRH5[13]	ADRH5[12]	ADRH5[11]	ADRH5[10]	ADRH5[9]	ADRH5[8]
		ADRH5[7]	ADRH5[6]	ADRH5[5]	ADRH5[4]	ADRH5[3]	ADRH5[2]	ADRH5[1]	ADRH5[0]
	TSU_ADRH6	ADRH6[31]	ADRH6[30]	ADRH6[29]	ADRH6[28]	ADRH6[27]	ADRH6[26]	ADRH6[25]	ADRH6[24]
		ADRH6[23]	ADRH6[22]	ADRH6[21]	ADRH6[20]	ADRH6[19]	ADRH6[18]	ADRH6[17]	ADRH6[16]
		ADRH6[15]	ADRH6[14]	ADRH6[13]	ADRH6[12]	ADRH6[11]	ADRH6[10]	ADRH6[9]	ADRH6[8]
		ADRH6[7]	ADRH6[6]	ADRH6[5]	ADRH6[4]	ADRH6[3]	ADRH6[2]	ADRH6[1]	ADRH6[0]
	TSU_ADRH7	ADRH7[31]	ADRH7[30]	ADRH7[29]	ADRH7[28]	ADRH7[27]	ADRH7[26]	ADRH7[25]	ADRH7[24]
		ADRH7[23]	ADRH7[22]	ADRH7[21]	ADRH7[20]	ADRH7[19]	ADRH7[18]	ADRH7[17]	ADRH7[16]
		ADRH7[15]	ADRH7[14]	ADRH7[13]	ADRH7[12]	ADRH7[11]	ADRH7[10]	ADRH7[9]	ADRH7[8]
		ADRH7[7]	ADRH7[6]	ADRH7[5]	ADRH7[4]	ADRH7[3]	ADRH7[2]	ADRH7[1]	ADRH7[0]
	TSU_ADRH8	ADRH8[31]	ADRH8[30]	ADRH8[29]	ADRH8[28]	ADRH8[27]	ADRH8[26]	ADRH8[25]	ADRH8[24]
		ADRH8[23]	ADRH8[22]	ADRH8[21]	ADRH8[20]	ADRH8[19]	ADRH8[18]	ADRH8[17]	ADRH8[16]
		ADRH8[15]	ADRH8[14]	ADRH8[13]	ADRH8[12]	ADRH8[11]	ADRH8[10]	ADRH8[9]	ADRH8[8]
		ADRH8[7]	ADRH8[6]	ADRH8[5]	ADRH8[4]	ADRH8[3]	ADRH8[2]	ADRH8[1]	ADRH8[0]
	TSU_ADRH9	ADRH9[31]	ADRH9[30]	ADRH9[29]	ADRH9[28]	ADRH9[27]	ADRH9[26]	ADRH9[25]	ADRH9[24]
		ADRH9[23]	ADRH9[22]	ADRH9[21]	ADRH9[20]	ADRH9[19]	ADRH9[18]	ADRH9[17]	ADRH9[16]
		ADRH9[15]	ADRH9[14]	ADRH9[13]	ADRH9[12]	ADRH9[11]	ADRH9[10]	ADRH9[9]	ADRH9[8]
		ADRH9[7]	ADRH9[6]	ADRH9[5]	ADRH9[4]	ADRH9[3]	ADRH9[2]	ADRH9[1]	ADRH9[0]
	TSU_ADRH10	ADRH10[31]	ADRH10[30]	ADRH10[29]	ADRH10[28]	ADRH10[27]	ADRH10[26]	ADRH10[25]	ADRH10[24]
		ADRH10[23]	ADRH10[22]	ADRH10[21]	ADRH10[20]	ADRH10[19]	ADRH10[18]	ADRH10[17]	ADRH10[16]
		ADRH10[15]	ADRH10[14]	ADRH10[13]	ADRH10[12]	ADRH10[11]	ADRH10[10]	ADRH10[9]	ADRH10[8]
		ADRH10[7]	ADRH10[6]	ADRH10[5]	ADRH10[4]	ADRH10[3]	ADRH10[2]	ADRH10[1]	ADRH10[0]
	TSU_ADRH11	ADRH11[31]	ADRH11[30]	ADRH11[29]	ADRH11[28]	ADRH11[27]	ADRH11[26]	ADRH11[25]	ADRH11[24]
		ADRH11[23]	ADRH11[22]	ADRH11[21]	ADRH11[20]	ADRH11[19]	ADRH11[18]	ADRH11[17]	ADRH11[16]
		ADRH11[15]	ADRH11[14]	ADRH11[13]	ADRH11[12]	ADRH11[11]	ADRH11[10]	ADRH11[9]	ADRH11[8]
		ADRH11[7]	ADRH11[6]	ADRH11[5]	ADRH11[4]	ADRH11[3]	ADRH11[2]	ADRH11[1]	ADRH11[0]
	TSU_ADRH12	ADRH12[31]	ADRH12[30]	ADRH12[29]	ADRH12[28]	ADRH12[27]	ADRH12[26]	ADRH12[25]	ADRH12[24]
		ADRH12[23]	ADRH12[22]	ADRH12[21]	ADRH12[20]	ADRH12[19]	ADRH12[18]	ADRH12[17]	ADRH12[16]
		ADRH12[15]	ADRH12[14]	ADRH12[13]	ADRH12[12]	ADRH12[11]	ADRH12[10]	ADRH12[9]	ADRH12[8]
		ADRH12[7]	ADRH12[6]	ADRH12[5]	ADRH12[4]	ADRH12[3]	ADRH12[2]	ADRH12[1]	ADRH12[0]
	TSU_ADRH13	ADRH13[31]	ADRH13[30]	ADRH13[29]	ADRH13[28]	ADRH13[27]	ADRH13[26]	ADRH13[25]	ADRH13[24]
		ADRH13[23]	ADRH13[22]	ADRH13[21]	ADRH13[20]	ADRH13[19]	ADRH13[18]	ADRH13[17]	ADRH13[16]
		ADRH13[15]	ADRH13[14]	ADRH13[13]	ADRH13[12]	ADRH13[11]	ADRH13[10]	ADRH13[9]	ADRH13[8]
		ADRH13[7]	ADRH13[6]	ADRH13[5]	ADRH13[4]	ADRH13[3]	ADRH13[2]	ADRH13[1]	ADRH13[0]
	TSU_ADRH14	ADRH14[31]	ADRH14[30]	ADRH14[29]	ADRH14[28]	ADRH14[27]	ADRH14[26]	ADRH14[25]	ADRH14[24]
		ADRH14[23]	ADRH14[22]	ADRH14[21]	ADRH14[20]	ADRH14[19]	ADRH14[18]	ADRH14[17]	ADRH14[16]
		ADRH14[15]	ADRH14[14]	ADRH14[13]	ADRH14[12]	ADRH14[11]	ADRH14[10]	ADRH14[9]	ADRH14[8]
		ADRH14[7]	ADRH14[6]	ADRH14[5]	ADRH14[4]	ADRH14[3]	ADRH14[2]	ADRH14[1]	ADRH14[0]
	TSU_ADRH15	ADRH15[31]	ADRH15[30]	ADRH15[29]	ADRH15[28]	ADRH15[27]	ADRH15[26]	ADRH15[25]	ADRH15[24]
		ADRH15[23]	ADRH15[22]	ADRH15[21]	ADRH15[20]	ADRH15[19]	ADRH15[18]	ADRH15[17]	ADRH15[16]
		ADRH15[15]	ADRH15[14]	ADRH15[13]	ADRH15[12]	ADRH15[11]	ADRH15[10]	ADRH15[9]	ADRH15[8]
		ADRH15[7]	ADRH15[6]	ADRH15[5]	ADRH15[4]	ADRH15[3]	ADRH15[2]	ADRH15[1]	ADRH15[0]
	TSU_ADRH16	ADRH16[31]	ADRH16[30]	ADRH16[29]	ADRH16[28]	ADRH16[27]	ADRH16[26]	ADRH16[25]	ADRH16[24]
		ADRH16[23]	ADRH16[22]	ADRH16[21]	ADRH16[20]	ADRH16[19]	ADRH16[18]	ADRH16[17]	ADRH16[16]
		ADRH16[15]	ADRH16[14]	ADRH16[13]	ADRH16[12]	ADRH16[11]	ADRH16[10]	ADRH16[9]	ADRH16[8]
		ADRH16[7]	ADRH16[6]	ADRH16[5]	ADRH16[4]	ADRH16[3]	ADRH16[2]	ADRH16[1]	ADRH16[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Ethernet controller	TSU_ADRH17	ADRH17[31]	ADRH17[30]	ADRH17[29]	ADRH17[28]	ADRH17[27]	ADRH17[26]	ADRH17[25]	ADRH17[24]
		ADRH17[23]	ADRH17[22]	ADRH17[21]	ADRH17[20]	ADRH17[19]	ADRH17[18]	ADRH17[17]	ADRH17[16]
		ADRH17[15]	ADRH17[14]	ADRH17[13]	ADRH17[12]	ADRH17[11]	ADRH17[10]	ADRH17[9]	ADRH17[8]
		ADRH17[7]	ADRH17[6]	ADRH17[5]	ADRH17[4]	ADRH17[3]	ADRH17[2]	ADRH17[1]	ADRH17[0]
	TSU_ADRH18	ADRH18[31]	ADRH18[30]	ADRH18[29]	ADRH18[28]	ADRH18[27]	ADRH18[26]	ADRH18[25]	ADRH18[24]
		ADRH18[23]	ADRH18[22]	ADRH18[21]	ADRH18[20]	ADRH18[19]	ADRH18[18]	ADRH18[17]	ADRH18[16]
		ADRH18[15]	ADRH18[14]	ADRH18[13]	ADRH18[12]	ADRH18[11]	ADRH18[10]	ADRH18[9]	ADRH18[8]
		ADRH18[7]	ADRH18[6]	ADRH18[5]	ADRH18[4]	ADRH18[3]	ADRH18[2]	ADRH18[1]	ADRH18[0]
	TSU_ADRH19	ADRH19[31]	ADRH19[30]	ADRH19[29]	ADRH19[28]	ADRH19[27]	ADRH19[26]	ADRH19[25]	ADRH19[24]
		ADRH19[23]	ADRH19[22]	ADRH19[21]	ADRH19[20]	ADRH19[19]	ADRH19[18]	ADRH19[17]	ADRH19[16]
		ADRH19[15]	ADRH19[14]	ADRH19[13]	ADRH19[12]	ADRH19[11]	ADRH19[10]	ADRH19[9]	ADRH19[8]
		ADRH19[7]	ADRH19[6]	ADRH19[5]	ADRH19[4]	ADRH19[3]	ADRH19[2]	ADRH19[1]	ADRH19[0]
	TSU_ADRH20	ADRH20[31]	ADRH20[30]	ADRH20[29]	ADRH20[28]	ADRH20[27]	ADRH20[26]	ADRH20[25]	ADRH20[24]
		ADRH20[23]	ADRH20[22]	ADRH20[21]	ADRH20[20]	ADRH20[19]	ADRH20[18]	ADRH20[17]	ADRH20[16]
		ADRH20[15]	ADRH20[14]	ADRH20[13]	ADRH20[12]	ADRH20[11]	ADRH20[10]	ADRH20[9]	ADRH20[8]
		ADRH20[7]	ADRH20[6]	ADRH20[5]	ADRH20[4]	ADRH20[3]	ADRH20[2]	ADRH20[1]	ADRH20[0]
	TSU_ADRH21	ADRH21[31]	ADRH21[30]	ADRH21[29]	ADRH21[28]	ADRH21[27]	ADRH21[26]	ADRH21[25]	ADRH21[24]
		ADRH21[23]	ADRH21[22]	ADRH21[21]	ADRH21[20]	ADRH21[19]	ADRH21[18]	ADRH21[17]	ADRH21[16]
		ADRH21[15]	ADRH21[14]	ADRH21[13]	ADRH21[12]	ADRH21[11]	ADRH21[10]	ADRH21[9]	ADRH21[8]
		ADRH21[7]	ADRH21[6]	ADRH21[5]	ADRH21[4]	ADRH21[3]	ADRH21[2]	ADRH21[1]	ADRH21[0]
	TSU_ADRH22	ADRH22[31]	ADRH22[30]	ADRH22[29]	ADRH22[28]	ADRH22[27]	ADRH22[26]	ADRH22[25]	ADRH22[24]
		ADRH22[23]	ADRH22[22]	ADRH22[21]	ADRH22[20]	ADRH22[19]	ADRH22[18]	ADRH22[17]	ADRH22[16]
		ADRH22[15]	ADRH22[14]	ADRH22[13]	ADRH22[12]	ADRH22[11]	ADRH22[10]	ADRH22[9]	ADRH22[8]
		ADRH22[7]	ADRH22[6]	ADRH22[5]	ADRH22[4]	ADRH22[3]	ADRH22[2]	ADRH22[1]	ADRH22[0]
	TSU_ADRH23	ADRH23[31]	ADRH23[30]	ADRH23[29]	ADRH23[28]	ADRH23[27]	ADRH23[26]	ADRH23[25]	ADRH23[24]
		ADRH23[23]	ADRH23[22]	ADRH23[21]	ADRH23[20]	ADRH23[19]	ADRH23[18]	ADRH23[17]	ADRH23[16]
		ADRH23[15]	ADRH23[14]	ADRH23[13]	ADRH23[12]	ADRH23[11]	ADRH23[10]	ADRH23[9]	ADRH23[8]
		ADRH23[7]	ADRH23[6]	ADRH23[5]	ADRH23[4]	ADRH23[3]	ADRH23[2]	ADRH23[1]	ADRH23[0]
	TSU_ADRH24	ADRH24[31]	ADRH24[30]	ADRH24[29]	ADRH24[28]	ADRH24[27]	ADRH24[26]	ADRH24[25]	ADRH24[24]
		ADRH24[23]	ADRH24[22]	ADRH24[21]	ADRH24[20]	ADRH24[19]	ADRH24[18]	ADRH24[17]	ADRH24[16]
		ADRH24[15]	ADRH24[14]	ADRH24[13]	ADRH24[12]	ADRH24[11]	ADRH24[10]	ADRH24[9]	ADRH24[8]
		ADRH24[7]	ADRH24[6]	ADRH24[5]	ADRH24[4]	ADRH24[3]	ADRH24[2]	ADRH24[1]	ADRH24[0]
	TSU_ADRH25	ADRH25[31]	ADRH25[30]	ADRH25[29]	ADRH25[28]	ADRH25[27]	ADRH25[26]	ADRH25[25]	ADRH25[24]
		ADRH25[23]	ADRH25[22]	ADRH25[21]	ADRH25[20]	ADRH25[19]	ADRH25[18]	ADRH25[17]	ADRH25[16]
		ADRH25[15]	ADRH25[14]	ADRH25[13]	ADRH25[12]	ADRH25[11]	ADRH25[10]	ADRH25[9]	ADRH25[8]
		ADRH25[7]	ADRH25[6]	ADRH25[5]	ADRH25[4]	ADRH25[3]	ADRH25[2]	ADRH25[1]	ADRH25[0]
	TSU_ADRH26	ADRH26[31]	ADRH26[30]	ADRH26[29]	ADRH26[28]	ADRH26[27]	ADRH26[26]	ADRH26[25]	ADRH26[24]
		ADRH26[23]	ADRH26[22]	ADRH26[21]	ADRH26[20]	ADRH26[19]	ADRH26[18]	ADRH26[17]	ADRH26[16]
		ADRH26[15]	ADRH26[14]	ADRH26[13]	ADRH26[12]	ADRH26[11]	ADRH26[10]	ADRH26[9]	ADRH26[8]
		ADRH26[7]	ADRH26[6]	ADRH26[5]	ADRH26[4]	ADRH26[3]	ADRH26[2]	ADRH26[1]	ADRH26[0]
	TSU_ADRH27	ADRH27[31]	ADRH27[30]	ADRH27[29]	ADRH27[28]	ADRH27[27]	ADRH27[26]	ADRH27[25]	ADRH27[24]
		ADRH27[23]	ADRH27[22]	ADRH27[21]	ADRH27[20]	ADRH27[19]	ADRH27[18]	ADRH27[17]	ADRH27[16]
		ADRH27[15]	ADRH27[14]	ADRH27[13]	ADRH27[12]	ADRH27[11]	ADRH27[10]	ADRH27[9]	ADRH27[8]
		ADRH27[7]	ADRH27[6]	ADRH27[5]	ADRH27[4]	ADRH27[3]	ADRH27[2]	ADRH27[1]	ADRH27[0]
	TSU_ADRH28	ADRH28[31]	ADRH28[30]	ADRH28[29]	ADRH28[28]	ADRH28[27]	ADRH28[26]	ADRH28[25]	ADRH28[24]
		ADRH28[23]	ADRH28[22]	ADRH28[21]	ADRH28[20]	ADRH28[19]	ADRH28[18]	ADRH28[17]	ADRH28[16]
		ADRH28[15]	ADRH28[14]	ADRH28[13]	ADRH28[12]	ADRH28[11]	ADRH28[10]	ADRH28[9]	ADRH28[8]
		ADRH28[7]	ADRH28[6]	ADRH28[5]	ADRH28[4]	ADRH28[3]	ADRH28[2]	ADRH28[1]	ADRH28[0]
TSU_ADRH29	ADRH29[31]	ADRH29[30]	ADRH29[29]	ADRH29[28]	ADRH29[27]	ADRH29[26]	ADRH29[25]	ADRH29[24]	
	ADRH29[23]	ADRH29[22]	ADRH29[21]	ADRH29[20]	ADRH29[19]	ADRH29[18]	ADRH29[17]	ADRH29[16]	
	ADRH29[15]	ADRH29[14]	ADRH29[13]	ADRH29[12]	ADRH29[11]	ADRH29[10]	ADRH29[9]	ADRH29[8]	
	ADRH29[7]	ADRH29[6]	ADRH29[5]	ADRH29[4]	ADRH29[3]	ADRH29[2]	ADRH29[1]	ADRH29[0]	
TSU_ADRH30	ADRH30[31]	ADRH30[30]	ADRH30[29]	ADRH30[28]	ADRH30[27]	ADRH30[26]	ADRH30[25]	ADRH30[24]	
	ADRH30[23]	ADRH30[22]	ADRH30[21]	ADRH30[20]	ADRH30[19]	ADRH30[18]	ADRH30[17]	ADRH30[16]	
	ADRH30[15]	ADRH30[14]	ADRH30[13]	ADRH30[12]	ADRH30[11]	ADRH30[10]	ADRH30[9]	ADRH30[8]	
	ADRH30[7]	ADRH30[6]	ADRH30[5]	ADRH30[4]	ADRH30[3]	ADRH30[2]	ADRH30[1]	ADRH30[0]	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Ethernet controller	TSU_ADRH31	ADRH31[31]	ADRH31[30]	ADRH31[29]	ADRH31[28]	ADRH31[27]	ADRH31[26]	ADRH31[25]	ADRH31[24]	
		ADRH31[23]	ADRH31[22]	ADRH31[21]	ADRH31[20]	ADRH31[19]	ADRH31[18]	ADRH31[17]	ADRH31[16]	
		ADRH31[15]	ADRH31[14]	ADRH31[13]	ADRH31[12]	ADRH31[11]	ADRH31[10]	ADRH31[9]	ADRH31[8]	
		ADRH31[7]	ADRH31[6]	ADRH31[5]	ADRH31[4]	ADRH31[3]	ADRH31[2]	ADRH31[1]	ADRH31[0]	
	TSU_ADRL0	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		ADRL0[15]	ADRL0[14]	ADRL0[13]	ADRL0[12]	ADRL0[11]	ADRL0[10]	ADRL0[9]	ADRL0[8]	ADRL0[7]
		ADRL0[7]	ADRL0[6]	ADRL0[5]	ADRL0[4]	ADRL0[3]	ADRL0[2]	ADRL0[1]	ADRL0[0]	-
	TSU_ADRL1	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		ADRL1[15]	ADRL1[14]	ADRL1[13]	ADRL1[12]	ADRL1[11]	ADRL1[10]	ADRL1[9]	ADRL1[8]	ADRL1[7]
	TSU_ADRL2	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		ADRL2[15]	ADRL2[14]	ADRL2[13]	ADRL2[12]	ADRL2[11]	ADRL2[10]	ADRL2[9]	ADRL2[8]	ADRL2[7]
	TSU_ADRL3	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		ADRL3[15]	ADRL3[14]	ADRL3[13]	ADRL3[12]	ADRL3[11]	ADRL3[10]	ADRL3[9]	ADRL3[8]	ADRL3[7]
	TSU_ADRL4	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		ADRL4[15]	ADRL4[14]	ADRL4[13]	ADRL4[12]	ADRL4[11]	ADRL4[10]	ADRL4[9]	ADRL4[8]	ADRL4[7]
	TSU_ADRL5	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		ADRL5[15]	ADRL5[14]	ADRL5[13]	ADRL5[12]	ADRL5[11]	ADRL5[10]	ADRL5[9]	ADRL5[8]	ADRL5[7]
	TSU_ADRL6	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		ADRL6[15]	ADRL6[14]	ADRL6[13]	ADRL6[12]	ADRL6[11]	ADRL6[10]	ADRL6[9]	ADRL6[8]	ADRL6[7]
	TSU_ADRL7	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		ADRL7[15]	ADRL7[14]	ADRL7[13]	ADRL7[12]	ADRL7[11]	ADRL7[10]	ADRL7[9]	ADRL7[8]	ADRL7[7]
	TSU_ADRL8	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		ADRL8[15]	ADRL8[14]	ADRL8[13]	ADRL8[12]	ADRL8[11]	ADRL8[10]	ADRL8[9]	ADRL8[8]	ADRL8[7]
	TSU_ADRL9	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		ADRL9[15]	ADRL9[14]	ADRL9[13]	ADRL9[12]	ADRL9[11]	ADRL9[10]	ADRL9[9]	ADRL9[8]	ADRL9[7]
	TSU_ADRL10	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		ADRL10[15]	ADRL10[14]	ADRL10[13]	ADRL10[12]	ADRL10[11]	ADRL10[10]	ADRL10[9]	ADRL10[8]	ADRL10[7]
	TSU_ADRL11	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		ADRL11[15]	ADRL11[14]	ADRL11[13]	ADRL11[12]	ADRL11[11]	ADRL11[10]	ADRL11[9]	ADRL11[8]	ADRL11[7]
	TSU_ADRL12	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		ADRL12[15]	ADRL12[14]	ADRL12[13]	ADRL12[12]	ADRL12[11]	ADRL12[10]	ADRL12[9]	ADRL12[8]	ADRL12[7]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Ethernet controller	TSU_ADRL13	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		ADRL13[15]	ADRL13[14]	ADRL13[13]	ADRL13[12]	ADRL13[11]	ADRL13[10]	ADRL13[9]	ADRL13[8]
	ADRL13[7]	ADRL13[6]	ADRL13[5]	ADRL13[4]	ADRL13[3]	ADRL13[2]	ADRL13[1]	ADRL13[0]	
	TSU_ADRL14	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		ADRL14[15]	ADRL14[14]	ADRL14[13]	ADRL14[12]	ADRL14[11]	ADRL14[10]	ADRL14[9]	ADRL14[8]
	ADRL14[7]	ADRL14[6]	ADRL14[5]	ADRL14[4]	ADRL14[3]	ADRL14[2]	ADRL14[1]	ADRL14[0]	
	TSU_ADRL15	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		ADRL15[15]	ADRL15[14]	ADRL15[13]	ADRL15[12]	ADRL15[11]	ADRL15[10]	ADRL15[9]	ADRL15[8]
	ADRL15[7]	ADRL15[6]	ADRL15[5]	ADRL15[4]	ADRL15[3]	ADRL15[2]	ADRL15[1]	ADRL15[0]	
	TSU_ADRL16	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		ADRL16[15]	ADRL16[14]	ADRL16[13]	ADRL16[12]	ADRL16[11]	ADRL16[10]	ADRL16[9]	ADRL16[8]
	ADRL16[7]	ADRL16[6]	ADRL16[5]	ADRL16[4]	ADRL16[3]	ADRL16[2]	ADRL16[1]	ADRL16[0]	
	TSU_ADRL17	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		ADRL17[15]	ADRL17[14]	ADRL17[13]	ADRL17[12]	ADRL17[11]	ADRL17[10]	ADRL17[9]	ADRL17[8]
	ADRL17[7]	ADRL17[6]	ADRL17[5]	ADRL17[4]	ADRL17[3]	ADRL17[2]	ADRL17[1]	ADRL17[0]	
	TSU_ADRL18	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		ADRL18[15]	ADRL18[14]	ADRL18[13]	ADRL18[12]	ADRL18[11]	ADRL18[10]	ADRL18[9]	ADRL18[8]
	ADRL18[7]	ADRL18[6]	ADRL18[5]	ADRL18[4]	ADRL18[3]	ADRL18[2]	ADRL18[1]	ADRL18[0]	
	TSU_ADRL19	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		ADRL19[15]	ADRL19[14]	ADRL19[13]	ADRL19[12]	ADRL19[11]	ADRL19[10]	ADRL19[9]	ADRL19[8]
	ADRL19[7]	ADRL19[6]	ADRL19[5]	ADRL19[4]	ADRL19[3]	ADRL19[2]	ADRL19[1]	ADRL19[0]	
	TSU_ADRL20	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		ADRL20[15]	ADRL20[14]	ADRL20[13]	ADRL20[12]	ADRL20[11]	ADRL20[10]	ADRL20[9]	ADRL20[8]
	ADRL20[7]	ADRL20[6]	ADRL20[5]	ADRL20[4]	ADRL20[3]	ADRL20[2]	ADRL20[1]	ADRL20[0]	
	TSU_ADRL21	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		ADRL21[15]	ADRL21[14]	ADRL21[13]	ADRL21[12]	ADRL21[11]	ADRL21[10]	ADRL21[9]	ADRL21[8]
	ADRL21[7]	ADRL21[6]	ADRL21[5]	ADRL21[4]	ADRL21[3]	ADRL21[2]	ADRL21[1]	ADRL21[0]	
	TSU_ADRL22	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		ADRL22[15]	ADRL22[14]	ADRL22[13]	ADRL22[12]	ADRL22[11]	ADRL22[10]	ADRL22[9]	ADRL22[8]
	ADRL22[7]	ADRL22[6]	ADRL22[5]	ADRL22[4]	ADRL22[3]	ADRL22[2]	ADRL22[1]	ADRL22[0]	
	TSU_ADRL23	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		ADRL23[15]	ADRL23[14]	ADRL23[13]	ADRL23[12]	ADRL23[11]	ADRL23[10]	ADRL23[9]	ADRL23[8]
	ADRL23[7]	ADRL23[6]	ADRL23[5]	ADRL23[4]	ADRL23[3]	ADRL23[2]	ADRL23[1]	ADRL23[0]	
	TSU_ADRL24	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		ADRL24[15]	ADRL24[14]	ADRL24[13]	ADRL24[12]	ADRL24[11]	ADRL24[10]	ADRL24[9]	ADRL24[8]
	ADRL24[7]	ADRL24[6]	ADRL24[5]	ADRL24[4]	ADRL24[3]	ADRL24[2]	ADRL24[1]	ADRL24[0]	
TSU_ADRL25	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	
	ADRL25[15]	ADRL25[14]	ADRL25[13]	ADRL25[12]	ADRL25[11]	ADRL25[10]	ADRL25[9]	ADRL25[8]	
ADRL25[7]	ADRL25[6]	ADRL25[5]	ADRL25[4]	ADRL25[3]	ADRL25[2]	ADRL25[1]	ADRL25[0]		
TSU_ADRL26	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	
	ADRL26[15]	ADRL26[14]	ADRL26[13]	ADRL26[12]	ADRL26[11]	ADRL26[10]	ADRL26[9]	ADRL26[8]	
ADRL26[7]	ADRL26[6]	ADRL26[5]	ADRL26[4]	ADRL26[3]	ADRL26[2]	ADRL26[1]	ADRL26[0]		



Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Ethernet controller	TSU_ADRL27	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		ADRL27[15]	ADRL27[14]	ADRL27[13]	ADRL27[12]	ADRL27[11]	ADRL27[10]	ADRL27[9]	ADRL27[8]
	ADRL27[7]	ADRL27[6]	ADRL27[5]	ADRL27[4]	ADRL27[3]	ADRL27[2]	ADRL27[1]	ADRL27[0]	
	TSU_ADRL28	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		ADRL28[15]	ADRL28[14]	ADRL28[13]	ADRL28[12]	ADRL28[11]	ADRL28[10]	ADRL28[9]	ADRL28[8]
	ADRL28[7]	ADRL28[6]	ADRL28[5]	ADRL28[4]	ADRL28[3]	ADRL28[2]	ADRL28[1]	ADRL28[0]	
	TSU_ADRL29	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		ADRL29[15]	ADRL29[14]	ADRL29[13]	ADRL29[12]	ADRL29[11]	ADRL29[10]	ADRL29[9]	ADRL29[8]
	ADRL29[7]	ADRL29[6]	ADRL29[5]	ADRL29[4]	ADRL29[3]	ADRL29[2]	ADRL29[1]	ADRL29[0]	
	TSU_ADRL30	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		ADRL30[15]	ADRL30[14]	ADRL30[13]	ADRL30[12]	ADRL30[11]	ADRL30[10]	ADRL30[9]	ADRL30[8]
	ADRL30[7]	ADRL30[6]	ADRL30[5]	ADRL30[4]	ADRL30[3]	ADRL30[2]	ADRL30[1]	ADRL30[0]	
	TSU_ADRL31	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		ADRL31[15]	ADRL31[14]	ADRL31[13]	ADRL31[12]	ADRL31[11]	ADRL31[10]	ADRL31[9]	ADRL31[8]
	ADRL31[7]	ADRL31[6]	ADRL31[5]	ADRL31[4]	ADRL31[3]	ADRL31[2]	ADRL31[1]	ADRL31[0]	
	TXNLCR0	NTC0[31]	NTC0[30]	NTC0[29]	NTC0[28]	NTC0[27]	NTC0[26]	NTC0[25]	NTC0[24]
		NTC0[23]	NTC0[22]	NTC0[21]	NTC0[20]	NTC0[19]	NTC0[18]	NTC0[17]	NTC0[16]
		NTC0[15]	NTC0[14]	NTC0[13]	NTC0[12]	NTC0[11]	NTC0[10]	NTC0[9]	NTC0[8]
		NTC0[7]	NTC0[6]	NTC0[5]	NTC0[4]	NTC0[3]	NTC0[2]	NTC0[1]	NTC0[0]
	TXALCR0	TC0[31]	TC0[30]	TC0[29]	TC0[28]	TC0[27]	TC0[26]	TC0[25]	TC0[24]
		TC0[23]	TC0[22]	TC0[21]	TC0[20]	TC0[19]	TC0[18]	TC0[17]	TC0[16]
		TC0[15]	TC0[14]	TC0[13]	TC0[12]	TC0[11]	TC0[10]	TC0[9]	TC0[8]
		TC0[7]	TC0[6]	TC0[5]	TC0[4]	TC0[3]	TC0[2]	TC0[1]	TC0[0]
	RXNLCR0	NRC0[31]	NRC0[30]	NRC0[29]	NRC0[28]	NRC0[27]	NRC0[26]	NRC0[25]	NRC0[24]
		NRC0[23]	NRC0[22]	NRC0[21]	NRC0[20]	NRC0[19]	NRC0[18]	NRC0[17]	NRC0[16]
		NRC0[15]	NRC0[14]	NRC0[13]	NRC0[12]	NRC0[11]	NRC0[10]	NRC0[9]	NRC0[8]
		NRC0[7]	NRC0[6]	NRC0[5]	NRC0[4]	NRC0[3]	NRC0[2]	NRC0[1]	NRC0[0]
	RXALCR0	RC0[31]	RC0[30]	RC0[29]	RC0[28]	RC0[27]	RC0[26]	RC0[25]	RC0[24]
		RC0[23]	RC0[22]	RC0[21]	RC0[20]	RC0[19]	RC0[18]	RC0[17]	RC0[16]
		RC0[15]	RC0[14]	RC0[13]	RC0[12]	RC0[11]	RC0[10]	RC0[9]	RC0[8]
		RC0[7]	RC0[6]	RC0[5]	RC0[4]	RC0[3]	RC0[2]	RC0[1]	RC0[0]
	EDSR0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	ENT	ENR
	EDMR0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	DE	DL[1]	DL[0]	-	-	SWRT	SWRR
	EDTRR0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	TR[1]	TR[0]
	EDRRR0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	RR
	EESR0	TWB[1]	TWB[0]	TC[1]	TUC	ROC	TABT	RABT	RFCOF
		-	ECI	TC[0]	TDE	TFUF	FR	RDE	RFOF
		-	-	-	-	-	-	-	-
		RMAF	-	-	RRF	RTLf	RTSF	PRE	CERF

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Ethernet controller	EESIPR0	TWB1IP	TWB0IP	TC1IP	TUCIP	ROCIP	TABTIP	RABTIP	RFCOFIP
		-	ECIIP	TC0IP	TDEIP	TFUFIP	FRIP	RDEIP	RFOFIP
		-	-	-	-	-	-	-	-
		RMAFIP	-	-	RRFIP	RTLFIIP	RTSFIP	PREIP	CERFIP
	TDLAR0	TDLA[31]	TDLA[30]	TDLA[29]	TDLA[28]	TDLA[27]	TDLA[26]	TDLA[25]	TDLA[24]
		TDLA[23]	TDLA[22]	TDLA[21]	TDLA[20]	TDLA[19]	TDLA[18]	TDLA[17]	TDLA[16]
		TDLA[15]	TDLA[14]	TDLA[13]	TDLA[12]	TDLA[11]	TDLA[10]	TDLA[9]	TDLA[8]
		TDLA[7]	TDLA[6]	TDLA[5]	TDLA[4]	TDLA[3]	TDLA[2]	TDLA[1]	TDLA[0]
	TDFAR0	TDFA[31]	TDFA[30]	TDFA[29]	TDFA[28]	TDFA[27]	TDFA[26]	TDFA[25]	TDFA[24]
		TDFA[23]	TDFA[22]	TDFA[21]	TDFA[20]	TDFA[19]	TDFA[18]	TDFA[17]	TDFA[16]
		TDFA[15]	TDFA[14]	TDFA[13]	TDFA[12]	TDFA[11]	TDFA[10]	TDFA[9]	TDFA[8]
		TDFA[7]	TDFA[6]	TDFA[5]	TDFA[4]	TDFA[3]	TDFA[2]	TDFA[1]	TDFA[0]
	TDFXR0	TDFX[31]	TDFX[30]	TDFX[29]	TDFX[28]	TDFX[27]	TDFX[26]	TDFX[25]	TDFX[24]
		TDFX[23]	TDFX[22]	TDFX[21]	TDFX[20]	TDFX[19]	TDFX[18]	TDFX[17]	TDFX[16]
		TDFX[15]	TDFX[14]	TDFX[13]	TDFX[12]	TDFX[11]	TDFX[10]	TDFX[9]	TDFX[8]
		TDFX[7]	TDFX[6]	TDFX[5]	TDFX[4]	TDFX[3]	TDFX[2]	TDFX[1]	TDFX[0]
	TDFFR0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	TDLF
	RDLAR0	RDLA[31]	RDLA[30]	RDLA[29]	RDLA[28]	RDLA[27]	RDLA[26]	RDLA[25]	RDLA[24]
		RDLA[23]	RDLA[22]	RDLA[21]	RDLA[20]	RDLA[19]	RDLA[18]	RDLA[17]	RDLA[16]
		RDLA[15]	RDLA[14]	RDLA[13]	RDLA[12]	RDLA[11]	RDLA[10]	RDLA[9]	RDLA[8]
		RDLA[7]	RDLA[6]	RDLA[5]	RDLA[4]	RDLA[3]	RDLA[2]	RDLA[1]	RDLA[0]
	RDFAR0	RDFA[31]	RDFA[30]	RDFA[29]	RDFA[28]	RDFA[27]	RDFA[26]	RDFA[25]	RDFA[24]
		RDFA[23]	RDFA[22]	RDFA[21]	RDFA[20]	RDFA[19]	RDFA[18]	RDFA[17]	RDFA[16]
		RDFA[15]	RDFA[14]	RDFA[13]	RDFA[12]	RDFA[11]	RDFA[10]	RDFA[9]	RDFA[8]
		RDFA[7]	RDFA[6]	RDFA[5]	RDFA[4]	RDFA[3]	RDFA[2]	RDFA[1]	RDFA[0]
	RDFXR0	RDFX[31]	RDFX[30]	RDFX[29]	RDFX[28]	RDFX[27]	RDFX[26]	RDFX[25]	RDFX[24]
		RDFX[23]	RDFX[22]	RDFX[21]	RDFX[20]	RDFX[19]	RDFX[18]	RDFX[17]	RDFX[16]
		RDFX[15]	RDFX[14]	RDFX[13]	RDFX[12]	RDFX[11]	RDFX[10]	RDFX[9]	RDFX[8]
		RDFX[7]	RDFX[6]	RDFX[5]	RDFX[4]	RDFX[3]	RDFX[2]	RDFX[1]	RDFX[0]
	RDFFR0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	RDLF
	TRSCER0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	TABTCE	RABTCE
		-	-	-	-	-	-	-	-
		RMAFCE	-	-	RRFCE	RTLFCCE	RTSFCE	PRECE	CERFCE
	RMFCR0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		MFC[15]	MFC[14]	MFC[13]	MFC[12]	MFC[11]	MFC[10]	MFC[9]	MFC[8]
		MFC[7]	MFC[6]	MFC[5]	MFC[4]	MFC[3]	MFC[2]	MFC[1]	MFC[0]
	TFTR0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	TFT[10]	TFT[9]	TFT[8]
		TFT[7]	TFT[6]	TFT[5]	TFT[4]	TFT[3]	TFT[2]	TFT[1]	TFT[0]
	FDR0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	TFD[2]	TFD[1]	TFD[0]
		-	-	-	RFD[4]	RFD[3]	RFD[2]	RFD[1]	RFD[0]
	RMCRO	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	RNC

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Ethernet controller	RPADIR0	-	-	-	-	-	-	-	-
		-	-	-	PADS[4]	PADS[3]	PADS[2]	PADS[1]	PADS[0]
		PADR[15]	PADR[14]	PADR[13]	PADR[12]	PADR[11]	PADR[10]	PADR[9]	PADR[8]
		PADR[7]	PADR[6]	PADR[5]	PADR[4]	PADR[3]	PADR[2]	PADR[1]	PADR[0]
	FCFTR0	-	-	-	-	-	-	-	-
		-	-	-	RFF[4]	RFF[3]	RFF[2]	RFF[1]	RFF[0]
		-	-	-	-	-	-	-	-
		RFD[7]	RFD[6]	RFD[5]	RFD[4]	RFD[3]	RFD[2]	RFD[1]	RFD[0]
	CSMR	CSEBL	CAMD	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	SB[5]	SB[4]	SB[3]	SB[2]	SB[1]	SB[0]
	CSSBM	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	SBM[5]	SBM[4]	SBM[3]	SBM[2]	SBM[1]	SBM[0]
	CSSMR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		CS[15]	CS[14]	CS[13]	CS[12]	CS[11]	CS[10]	CS[9]	CS[8]
		CS[7]	CS[6]	CS[5]	CS[4]	CS[3]	CS[2]	CS[1]	CS[0]
A/D converter	ADDRA	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	ADDRB	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	ADDRC	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	ADDRD	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	ADDRE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	ADDRF	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	ADDRG	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	ADDRH	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	ADCMPHA	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	ADCMPLA	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	ADCMPHB	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	ADCMPLB	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	ADCMPHC	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	ADCMPLC	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	ADCMPHD	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	ADCMPLD	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	ADCMPHE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	ADCMPL E	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	ADCMPHF	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
A/D converter	ADCMPLF	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
	ADCMPLHG	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
	ADCMPLG	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
	ADCMPLHH	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
	ADCMPLH	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
	ADCSR	ADF	ADIE	ADST	TRGS[3]	TRGS[2]	TRGS[1]	TRGS[0]	CKS[2]	
		CKS[1]	CKS[0]	MDS[2]	MDS[1]	MDS[0]	CH[2]	CH[1]	CH[0]	
	ADCMPEP	HLMENH	HLMENG	HLMENF	HLMENE	HLMEND	HLMENC	HLMENB	HLMENA	
		LLMENH	LLMENG	LLMENF	LLMENE	LLMEND	LLMENC	LLMENB	LLMENA	
ADCMPSR	HOVRH	HOVRG	HOVRF	HOVRE	HOVRD	HOVRC	HOVRB	HOVRA		
	LUDRH	LUDRG	LUDRF	LUDRE	LUDRD	LUDRC	LUDRB	LUDRA		
NAND flash memory controller	FLCMNCR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	SNAND	QTSEL	-	
		-	-	-	-	ACM[1]	ACM[0]	NANDWF	-	
		-	-	-	-	CE	-	-	-	
	FLCMDRC	ADRCNT2	-	-	-	-	-	-	CDSRC	DOSR
		-	-	SELRW	DOADR	ADRCNT[1]	ADRCNT[0]	DOCMD2	DOCMD1	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
	FLCMCDR	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		CMD2[7]	CMD2[6]	CMD2[5]	CMD2[4]	CMD2[3]	CMD2[2]	CMD2[1]	CMD2[0]	
	FLADR	CMD1[7]	CMD1[6]	CMD1[5]	CMD1[4]	CMD1[3]	CMD1[2]	CMD1[1]	CMD1[0]	
		ADR4[7]	ADR4[6]	ADR4[5]	ADR4[4]	ADR4[3]	ADR4[2]	ADR4[1]	ADR4[0]	
		ADR3[7]	ADR3[6]	ADR3[5]	ADR3[4]	ADR3[3]	ADR3[2]	ADR3[1]	ADR3[0]	
		ADR2[7]	ADR2[6]	ADR2[5]	ADR2[4]	ADR2[3]	ADR2[2]	ADR2[1]	ADR2[0]	
	FLADR2	ADR1[7]	ADR1[6]	ADR1[5]	ADR1[4]	ADR1[3]	ADR1[2]	ADR1[1]	ADR1[0]	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
	FLDTCNTR	ADR5[7]	ADR5[6]	ADR5[5]	ADR5[4]	ADR5[3]	ADR5[2]	ADR5[1]	ADR5[0]	
		-	-	-	-	-	-	-	-	
		DTFLW[7]	DTFLW[6]	DTFLW[5]	DTFLW[4]	DTFLW[3]	DTFLW[2]	DTFLW[1]	DTFLW[0]	
		-	-	-	-	DTCNT[11]	DTCNT[10]	DTCNT[9]	DTCNT[8]	
	FLDATAR	DTCNT[7]	DTCNT[6]	DTCNT[5]	DTCNT[4]	DTCNT[3]	DTCNT[2]	DTCNT[1]	DTCNT[0]	
		DT4[7]	DT4[6]	DT4[5]	DT4[4]	DT4[3]	DT4[2]	DT4[1]	DT4[0]	
		DT3[7]	DT3[6]	DT3[5]	DT3[4]	DT3[3]	DT3[2]	DT3[1]	DT3[0]	
		DT2[7]	DT2[6]	DT2[5]	DT2[4]	DT2[3]	DT2[2]	DT2[1]	DT2[0]	
	FLINTDMACR	DT1[7]	DT1[6]	DT1[5]	DT1[4]	DT1[3]	DT1[2]	DT1[1]	DT1[0]	
		-	-	-	-	-	-	-	-	
		-	-	FIFOTRG[1]	FIFOTRG[0]	-	AC0CLR	-	DREQ0EN	
		BTOERB	-	TRREQF0	STERINTE	RBERINTE	TEINTE	-	STERB	
	FLBSYMR	-	-	-	-	-	-	-	-	
		-	-	-	-	RBTMOUT[19]	RBTMOUT[18]	RBTMOUT[17]	RBTMOUT[16]	
		RBTMOUT[15]	RBTMOUT[14]	RBTMOUT[13]	RBTMOUT[12]	RBTMOUT[11]	RBTMOUT[10]	RBTMOUT[9]	RBTMOUT[8]	
	FLBSYCNTR	RBTMOUT[7]	RBTMOUT[6]	RBTMOUT[5]	RBTMOUT[4]	RBTMOUT[3]	RBTMOUT[2]	RBTMOUT[1]	RBTMOUT[0]	
		STAT[7]	STAT[6]	STAT[5]	STAT[4]	STAT[3]	STAT[2]	STAT[1]	STAT[0]	
		-	-	-	-	RBTMOUT[19]	RBTMOUT[18]	RBTMOUT[17]	RBTMOUT[16]	
		RBTMOUT[15]	RBTMOUT[14]	RBTMOUT[13]	RBTMOUT[12]	RBTMOUT[11]	RBTMOUT[10]	RBTMOUT[9]	RBTMOUT[8]	
	FLDTFIFO	RBTMOUT[7]	RBTMOUT[6]	RBTMOUT[5]	RBTMOUT[4]	RBTMOUT[3]	RBTMOUT[2]	RBTMOUT[1]	RBTMOUT[0]	
		RBTMOUT[15]	RBTMOUT[14]	RBTMOUT[13]	RBTMOUT[12]	RBTMOUT[11]	RBTMOUT[10]	RBTMOUT[9]	RBTMOUT[8]	
		RBTMOUT[7]	RBTMOUT[6]	RBTMOUT[5]	RBTMOUT[4]	RBTMOUT[3]	RBTMOUT[2]	RBTMOUT[1]	RBTMOUT[0]	
		DTFO[31]	DTFO[30]	DTFO[29]	DTFO[28]	DTFO[27]	DTFO[26]	DTFO[25]	DTFO[24]	
FLDTFIFO	DTFO[23]	DTFO[22]	DTFO[21]	DTFO[20]	DTFO[19]	DTFO[18]	DTFO[17]	DTFO[16]		
	DTFO[15]	DTFO[14]	DTFO[13]	DTFO[12]	DTFO[11]	DTFO[10]	DTFO[9]	DTFO[8]		
	DTFO[7]	DTFO[6]	DTFO[5]	DTFO[4]	DTFO[3]	DTFO[2]	DTFO[1]	DTFO[0]		
	DTFO[7]	DTFO[6]	DTFO[5]	DTFO[4]	DTFO[3]	DTFO[2]	DTFO[1]	DTFO[0]		

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
NAND flash memory controller	FLTRCR	-	-	-	-	-	-	TREND	TRSTRT
USB2.0 host/ function module	SYSCFG0_0	-	-	-	-	-	SCKE	-	-
		HSE	DCFM	DRPD	DPRPU	-	UCKSEL	UPLLE	USBE
	BUSWAIT_0	-	-	-	-	-	-	-	-
		-	-	BWAIT[5]	BWAIT[4]	BWAIT[3]	BWAIT[2]	BWAIT[1]	BWAIT[0]
	SYSSTS0_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	LNST[1]	LNST[0]
	DVSTCTR0_0	-	-	-	-	-	-	-	WKUP
		RWUPE	USBRST	RESUME	UACT	-	RHST[2]	RHST[1]	RHST[0]
	TESTMODE_0	-	-	-	-	-	-	-	-
		-	-	-	-	UTST[3]	UTST[2]	UTST[1]	UTST[0]
	D0FBCFG_0	-	-	DFACC[1]	DFACC[0]	-	-	-	-
		-	-	-	TENDE	-	-	-	-
	D1FBCFG_0	-	-	DFACC[1]	DFACC[0]	-	-	-	-
		-	-	-	TENDE	-	-	-	-
	CFIFO_0	FIFOPORT[31]	FIFOPORT[30]	FIFOPORT[29]	FIFOPORT[28]	FIFOPORT[27]	FIFOPORT[26]	FIFOPORT[25]	FIFOPORT[24]
		FIFOPORT[23]	FIFOPORT[22]	FIFOPORT[21]	FIFOPORT[20]	FIFOPORT[19]	FIFOPORT[18]	FIFOPORT[17]	FIFOPORT[16]
		FIFOPORT[15]	FIFOPORT[14]	FIFOPORT[13]	FIFOPORT[12]	FIFOPORT[11]	FIFOPORT[10]	FIFOPORT[9]	FIFOPORT[8]
		FIFOPORT[7]	FIFOPORT[6]	FIFOPORT[5]	FIFOPORT[4]	FIFOPORT[3]	FIFOPORT[2]	FIFOPORT[1]	FIFOPORT[0]
	D0FIFO_0	FIFOPORT[31]	FIFOPORT[30]	FIFOPORT[29]	FIFOPORT[28]	FIFOPORT[27]	FIFOPORT[26]	FIFOPORT[25]	FIFOPORT[24]
		FIFOPORT[23]	FIFOPORT[22]	FIFOPORT[21]	FIFOPORT[20]	FIFOPORT[19]	FIFOPORT[18]	FIFOPORT[17]	FIFOPORT[16]
		FIFOPORT[15]	FIFOPORT[14]	FIFOPORT[13]	FIFOPORT[12]	FIFOPORT[11]	FIFOPORT[10]	FIFOPORT[9]	FIFOPORT[8]
		FIFOPORT[7]	FIFOPORT[6]	FIFOPORT[5]	FIFOPORT[4]	FIFOPORT[3]	FIFOPORT[2]	FIFOPORT[1]	FIFOPORT[0]
	D1FIFO_0	FIFOPORT[31]	FIFOPORT[30]	FIFOPORT[29]	FIFOPORT[28]	FIFOPORT[27]	FIFOPORT[26]	FIFOPORT[25]	FIFOPORT[24]
		FIFOPORT[23]	FIFOPORT[22]	FIFOPORT[21]	FIFOPORT[20]	FIFOPORT[19]	FIFOPORT[18]	FIFOPORT[17]	FIFOPORT[16]
		FIFOPORT[15]	FIFOPORT[14]	FIFOPORT[13]	FIFOPORT[12]	FIFOPORT[11]	FIFOPORT[10]	FIFOPORT[9]	FIFOPORT[8]
		FIFOPORT[7]	FIFOPORT[6]	FIFOPORT[5]	FIFOPORT[4]	FIFOPORT[3]	FIFOPORT[2]	FIFOPORT[1]	FIFOPORT[0]
	CFIFOSEL_0	RCNT	REW	-	-	MBW[1]	MBW[0]	-	BIGEND
		-	-	ISEL	-	CURPIPE[3]	CURPIPE[2]	CURPIPE[1]	CURPIPE[0]
	CFIFOCTR_0	BVAL	BCLR	FRDY	-	DTLN[11]	DTLN[10]	DTLN[9]	DTLN[8]
		DTLN[7]	DTLN[6]	DTLN[5]	DTLN[4]	DTLN[3]	DTLN[2]	DTLN[1]	DTLN[0]
	D0FIFOSEL_0	RCNT	REW	DCLRM	DREQE	MBW[1]	MBW[0]	-	BIGEND
		-	-	-	-	CURPIPE[3]	CURPIPE[2]	CURPIPE[1]	CURPIPE[0]
	D0FIFOCTR_0	BVAL	BCLR	FRDY	-	DTLN[11]	DTLN[10]	DTLN[9]	DTLN[8]
		DTLN[7]	DTLN[6]	DTLN[5]	DTLN[4]	DTLN[3]	DTLN[2]	DTLN[1]	DTLN[0]
	D1FIFOSEL_0	RCNT	REW	DCLRM	DREQE	MBW[1]	MBW[0]	-	BIGEND
		-	-	-	-	CURPIPE[3]	CURPIPE[2]	CURPIPE[1]	CURPIPE[0]
	D1FIFOCTR_0	BVAL	BCLR	FRDY	-	DTLN[11]	DTLN[10]	DTLN[9]	DTLN[8]
		DTLN[7]	DTLN[6]	DTLN[5]	DTLN[4]	DTLN[3]	DTLN[2]	DTLN[1]	DTLN[0]
	INTENB0_0	VBSE	RSME	SOFE	DVSE	CTRE	BEMPE	NRDYE	BRDYE
		-	-	-	-	-	-	-	-
	INTENB1_0	-	BCHGE	-	DTCHE	ATTICHE	-	-	-
		-	EOFERRE	SIGNE	SACKE	-	-	-	-
	BRDYENB_0	PIPEBRDYE[15]	PIPEBRDYE[14]	PIPEBRDYE[13]	PIPEBRDYE[12]	PIPEBRDYE[11]	PIPEBRDYE[10]	PIPEBRDYE[9]	PIPEBRDYE[8]
		PIPEBRDYE[7]	PIPEBRDYE[6]	PIPEBRDYE[5]	PIPEBRDYE[4]	PIPEBRDYE[3]	PIPEBRDYE[2]	PIPEBRDYE[1]	PIPEBRDYE[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
USB2.0 host/ function module	NRDYENB_0	PIPENRDYE[15]	PIPENRDYE[14]	PIPENRDYE[13]	PIPENRDYE[12]	PIPENRDYE[11]	PIPENRDYE[10]	PIPENRDYE[9]	PIPENRDYE[8]
		PIPENRDYE[7]	PIPENRDYE[6]	PIPENRDYE[5]	PIPENRDYE[4]	PIPENRDYE[3]	PIPENRDYE[2]	PIPENRDYE[1]	PIPENRDYE[0]
	BEMPENB_0	PIPEBEMPE[15]	PIPEBEMPE[14]	PIPEBEMPE[13]	PIPEBEMPE[12]	PIPEBEMPE[11]	PIPEBEMPE[10]	PIPEBEMPE[9]	PIPEBEMPE[8]
		PIPEBEMPE[7]	PIPEBEMPE[6]	PIPEBEMPE[5]	PIPEBEMPE[4]	PIPEBEMPE[3]	PIPEBEMPE[2]	PIPEBEMPE[1]	PIPEBEMPE[0]
	SOFCFG_0	-	-	-	-	-	-	-	TRNENSEL
		-	BRDYM	-	-	-	-	-	-
	INTSTS0_0	VBINT	RESM	SOFR	DVST	CTRRT	BEMP	NRDY	BRDY
		VBSTS	DVSQ[2]	DVSQ[1]	DVSQ[0]	VALID	CTSQ[2]	CTSQ[1]	CTSQ[0]
	INTSTS1_0	-	BCHG	-	DTCH	ATTCH	-	-	-
		-	EOFERR	SIGN	SACK	-	-	-	-
	BRDYSTS_0	PIPEBRDY[15]	PIPEBRDY[14]	PIPEBRDY[13]	PIPEBRDY[12]	PIPEBRDY[11]	PIPEBRDY[10]	PIPEBRDY[9]	PIPEBRDY[8]
		PIPEBRDY[7]	PIPEBRDY[6]	PIPEBRDY[5]	PIPEBRDY[4]	PIPEBRDY[3]	PIPEBRDY[2]	PIPEBRDY[1]	PIPEBRDY[0]
	NRDYSTS_0	PIPENRDY[15]	PIPENRDY[14]	PIPENRDY[13]	PIPENRDY[12]	PIPENRDY[11]	PIPENRDY[10]	PIPENRDY[9]	PIPENRDY[8]
		PIPENRDY[7]	PIPENRDY[6]	PIPENRDY[5]	PIPENRDY[4]	PIPENRDY[3]	PIPENRDY[2]	PIPENRDY[1]	PIPENRDY[0]
	BEMPSTS_0	PIPEBEMP[15]	PIPEBEMP[14]	PIPEBEMP[13]	PIPEBEMP[12]	PIPEBEMP[11]	PIPEBEMP[10]	PIPEBEMP[9]	PIPEBEMP[8]
		PIPEBEMP[7]	PIPEBEMP[6]	PIPEBEMP[5]	PIPEBEMP[4]	PIPEBEMP[3]	PIPEBEMP[2]	PIPEBEMP[1]	PIPEBEMP[0]
	FRMNUM_0	OVRN	CRCE	-	-	-	FRNM[10]	FRNM[9]	FRNM[8]
		FRNM[7]	FRNM[6]	FRNM[5]	FRNM[4]	FRNM[3]	FRNM[2]	FRNM[1]	FRNM[0]
	UFRMNUM_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	UFRNM[2]	UFRNM[1]	UFRNM[0]
	USBADDR_0	-	-	-	-	-	-	-	-
		-	USBADDR[6]	USBADDR[5]	USBADDR[4]	USBADDR[3]	USBADDR[2]	USBADDR[1]	USBADDR[0]
	USBREQ_0	bRequest[7]	bRequest[6]	bRequest[5]	bRequest[4]	bRequest[3]	bRequest[2]	bRequest[1]	bRequest[0]
		bmRequestType[7]	bmRequestType[6]	bmRequestType[5]	bmRequestType[4]	bmRequestType[3]	bmRequestType[2]	bmRequestType[1]	bmRequestType[0]
	USBVAL_0	wValue[15]	wValue[14]	wValue[13]	wValue[12]	wValue[11]	wValue[10]	wValue[9]	wValue[8]
		wValue[7]	wValue[6]	wValue[5]	wValue[4]	wValue[3]	wValue[2]	wValue[1]	wValue[0]
	USBINDX_0	wIndex[15]	wIndex[14]	wIndex[13]	wIndex[12]	wIndex[11]	wIndex[10]	wIndex[9]	wIndex[8]
		wIndex[7]	wIndex[6]	wIndex[5]	wIndex[4]	wIndex[3]	wIndex[2]	wIndex[1]	wIndex[0]
	USBLENG_0	wLength[15]	wLength[14]	wLength[13]	wLength[12]	wLength[11]	wLength[10]	wLength[9]	wLength[8]
		wLength[7]	wLength[6]	wLength[5]	wLength[4]	wLength[3]	wLength[2]	wLength[1]	wLength[0]
	DCPCFG_0	-	-	-	-	-	-	-	CNTMD
		SHTNAK	-	-	DIR	-	-	-	-
	DCPMAXP_0	DEVSEL[3]	DEVSEL[2]	DEVSEL[1]	DEVSEL[0]	-	-	-	-
		-	MXPS[6]	MXPS[5]	MXPS[4]	MXPS[3]	MXPS[2]	MXPS[1]	MXPS[0]
	DCPCTR_0	BSTS	SUREQ	CSCLR	CSSTS	SUREQCLR	-	-	SQCLR
		SQSET	SQMON	PBUSY	PINGE	-	CCPL	PID[1]	PID[0]
	PIPESEL_0	-	-	-	-	-	-	-	-
		-	-	-	-	PIPESEL[3]	PIPESEL[2]	PIPESEL[1]	PIPESEL[0]
	PIPECFG_0	TYPE[1]	TYPE[0]	-	-	-	BFRE	DBLB	CNTMD
		SHTNAK	-	-	DIR	EPNUM[3]	EPNUM[2]	EPNUM[1]	EPNUM[0]
	PIPEBUF_0	-	BUFSIZE[4]	BUFSIZE[3]	BUFSIZE[2]	BUFSIZE[1]	BUFSIZE[0]	-	-
		BUFNMB[7]	BUFNMB[6]	BUFNMB[5]	BUFNMB[4]	BUFNMB[3]	BUFNMB[2]	BUFNMB[1]	BUFNMB[0]
	PIPEMAXP_0	DEVSEL[3]	DEVSEL[2]	DEVSEL[1]	DEVSEL[0]	-	MXPS[10]	MXPS[9]	MXPS[8]
		MXPS[7]	MXPS[6]	MXPS[5]	MXPS[4]	MXPS[3]	MXPS[2]	MXPS[1]	MXPS[0]
	PIPEPERI_0	-	-	-	IFIS	-	-	-	-
		-	-	-	-	-	IITV[2]	IITV[1]	IITV[0]
	PIPE1CTR_0	BSTS	INBUFM	CSCLR	CSSTS	-	ATREPM	ACLRM	SQCLR
		SQSET	SQMON	PBUSY	-	-	-	PID[1]	PID[0]
	PIPE2CTR_0	BSTS	INBUFM	CSCLR	CSSTS	-	ATREPM	ACLRM	SQCLR
		SQSET	SQMON	PBUSY	-	-	-	PID[1]	PID[0]
	PIPE3CTR_0	BSTS	INBUFM	CSCLR	CSSTS	-	ATREPM	ACLRM	SQCLR
		SQSET	SQMON	PBUSY	-	-	-	PID[1]	PID[0]
	PIPE4CTR_0	BSTS	INBUFM	CSCLR	CSSTS	-	ATREPM	ACLRM	SQCLR
		SQSET	SQMON	PBUSY	-	-	-	PID[1]	PID[0]
	PIPE5CTR_0	BSTS	INBUFM	CSCLR	CSSTS	-	ATREPM	ACLRM	SQCLR
		SQSET	SQMON	PBUSY	-	-	-	PID[1]	PID[0]

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
USB2.0 host/ function module	PIPE6CTR_0	BSTS	-	CSCLR	CSSTS	-	-	ACLRM	SQCLR	
		SQSET	SQMON	PBUSY	-	-	-	PID[1]	PID[0]	
	PIPE7CTR_0	BSTS	-	CSCLR	CSSTS	-	-	ACLRM	SQCLR	
		SQSET	SQMON	PBUSY	-	-	-	PID[1]	PID[0]	
	PIPE8CTR_0	BSTS	-	CSCLR	CSSTS	-	-	ACLRM	SQCLR	
		SQSET	SQMON	PBUSY	-	-	-	PID[1]	PID[0]	
	PIPE9CTR_0	BSTS	INBUFM	CSCLR	CSSTS	-	-	ATREPM	ACLRM	SQCLR
		SQSET	SQMON	PBUSY	-	-	-	PID[1]	PID[0]	
	PIPEACTR_0	BSTS	INBUFM	CSCLR	CSSTS	-	-	ATREPM	ACLRM	SQCLR
		SQSET	SQMON	PBUSY	-	-	-	PID[1]	PID[0]	
	PIPEBCTR_0	BSTS	INBUFM	CSCLR	CSSTS	-	-	ATREPM	ACLRM	SQCLR
		SQSET	SQMON	PBUSY	-	-	-	PID[1]	PID[0]	
	PIPECCTR_0	BSTS	INBUFM	CSCLR	CSSTS	-	-	ATREPM	ACLRM	SQCLR
		SQSET	SQMON	PBUSY	-	-	-	PID[1]	PID[0]	
	PIPEDCTR_0	BSTS	INBUFM	CSCLR	CSSTS	-	-	ATREPM	ACLRM	SQCLR
		SQSET	SQMON	PBUSY	-	-	-	PID[1]	PID[0]	
	PIPEECTR_0	BSTS	INBUFM	CSCLR	CSSTS	-	-	ATREPM	ACLRM	SQCLR
		SQSET	SQMON	PBUSY	-	-	-	PID[1]	PID[0]	
	PIPEFCTR_0	BSTS	INBUFM	CSCLR	CSSTS	-	-	ATREPM	ACLRM	SQCLR
		SQSET	SQMON	PBUSY	-	-	-	PID[1]	PID[0]	
	PIPE1TRE_0	-	-	-	-	-	-	-	TRENB	TRCLR
		-	-	-	-	-	-	-	-	-
	PIPE1TRN_0	TRNCNT[15]	TRNCNT[14]	TRNCNT[13]	TRNCNT[12]	TRNCNT[11]	TRNCNT[10]	TRNCNT[9]	TRNCNT[8]	TRNCNT[7]
		TRNCNT[7]	TRNCNT[6]	TRNCNT[5]	TRNCNT[4]	TRNCNT[3]	TRNCNT[2]	TRNCNT[1]	TRNCNT[0]	-
	PIPE2TRE_0	-	-	-	-	-	-	-	TRENB	TRCLR
		-	-	-	-	-	-	-	-	-
	PIPE2TRN_0	TRNCNT[15]	TRNCNT[14]	TRNCNT[13]	TRNCNT[12]	TRNCNT[11]	TRNCNT[10]	TRNCNT[9]	TRNCNT[8]	TRNCNT[7]
		TRNCNT[7]	TRNCNT[6]	TRNCNT[5]	TRNCNT[4]	TRNCNT[3]	TRNCNT[2]	TRNCNT[1]	TRNCNT[0]	-
	PIPE3TRE_0	-	-	-	-	-	-	-	TRENB	TRCLR
		-	-	-	-	-	-	-	-	-
	PIPE3TRN_0	TRNCNT[15]	TRNCNT[14]	TRNCNT[13]	TRNCNT[12]	TRNCNT[11]	TRNCNT[10]	TRNCNT[9]	TRNCNT[8]	TRNCNT[7]
		TRNCNT[7]	TRNCNT[6]	TRNCNT[5]	TRNCNT[4]	TRNCNT[3]	TRNCNT[2]	TRNCNT[1]	TRNCNT[0]	-
	PIPE4TRE_0	-	-	-	-	-	-	-	TRENB	TRCLR
		-	-	-	-	-	-	-	-	-
	PIPE4TRN_0	TRNCNT[15]	TRNCNT[14]	TRNCNT[13]	TRNCNT[12]	TRNCNT[11]	TRNCNT[10]	TRNCNT[9]	TRNCNT[8]	TRNCNT[7]
		TRNCNT[7]	TRNCNT[6]	TRNCNT[5]	TRNCNT[4]	TRNCNT[3]	TRNCNT[2]	TRNCNT[1]	TRNCNT[0]	-
	PIPE5TRE_0	-	-	-	-	-	-	-	TRENB	TRCLR
		-	-	-	-	-	-	-	-	-
	PIPE5TRN_0	TRNCNT[15]	TRNCNT[14]	TRNCNT[13]	TRNCNT[12]	TRNCNT[11]	TRNCNT[10]	TRNCNT[9]	TRNCNT[8]	TRNCNT[7]
		TRNCNT[7]	TRNCNT[6]	TRNCNT[5]	TRNCNT[4]	TRNCNT[3]	TRNCNT[2]	TRNCNT[1]	TRNCNT[0]	-
	PIPEBTRE_0	-	-	-	-	-	-	-	TRENB	TRCLR
		-	-	-	-	-	-	-	-	-
	PIPEBTRN_0	TRNCNT[15]	TRNCNT[14]	TRNCNT[13]	TRNCNT[12]	TRNCNT[11]	TRNCNT[10]	TRNCNT[9]	TRNCNT[8]	TRNCNT[7]
		TRNCNT[7]	TRNCNT[6]	TRNCNT[5]	TRNCNT[4]	TRNCNT[3]	TRNCNT[2]	TRNCNT[1]	TRNCNT[0]	-
	PIPECTRE_0	-	-	-	-	-	-	-	TRENB	TRCLR
		-	-	-	-	-	-	-	-	-
	PIPECTRN_0	TRNCNT[15]	TRNCNT[14]	TRNCNT[13]	TRNCNT[12]	TRNCNT[11]	TRNCNT[10]	TRNCNT[9]	TRNCNT[8]	TRNCNT[7]
		TRNCNT[7]	TRNCNT[6]	TRNCNT[5]	TRNCNT[4]	TRNCNT[3]	TRNCNT[2]	TRNCNT[1]	TRNCNT[0]	-
	PIPEDTRE_0	-	-	-	-	-	-	-	TRENB	TRCLR
		-	-	-	-	-	-	-	-	-
	PIPEDTRN_0	TRNCNT[15]	TRNCNT[14]	TRNCNT[13]	TRNCNT[12]	TRNCNT[11]	TRNCNT[10]	TRNCNT[9]	TRNCNT[8]	TRNCNT[7]
		TRNCNT[7]	TRNCNT[6]	TRNCNT[5]	TRNCNT[4]	TRNCNT[3]	TRNCNT[2]	TRNCNT[1]	TRNCNT[0]	-
	PIPEETRE_0	-	-	-	-	-	-	-	TRENB	TRCLR
		-	-	-	-	-	-	-	-	-
	PIPEETRN_0	TRNCNT[15]	TRNCNT[14]	TRNCNT[13]	TRNCNT[12]	TRNCNT[11]	TRNCNT[10]	TRNCNT[9]	TRNCNT[8]	TRNCNT[7]
		TRNCNT[7]	TRNCNT[6]	TRNCNT[5]	TRNCNT[4]	TRNCNT[3]	TRNCNT[2]	TRNCNT[1]	TRNCNT[0]	-

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
USB2.0 host/ function module	PIPEFTRE_0	-	-	-	-	-	-	TRENB	TRCLR	
		-	-	-	-	-	-	-	-	
	PIPEFTRN_0	TRNCNT[15]	TRNCNT[14]	TRNCNT[13]	TRNCNT[12]	TRNCNT[11]	TRNCNT[10]	TRNCNT[9]	TRNCNT[8]	
		TRNCNT[7]	TRNCNT[6]	TRNCNT[5]	TRNCNT[4]	TRNCNT[3]	TRNCNT[2]	TRNCNT[1]	TRNCNT[0]	
	PIPE9TRE_0	-	-	-	-	-	-	-	TRENB	TRCLR
		-	-	-	-	-	-	-	-	-
	PIPE9TRN_0	TRNCNT[15]	TRNCNT[14]	TRNCNT[13]	TRNCNT[12]	TRNCNT[11]	TRNCNT[10]	TRNCNT[9]	TRNCNT[8]	
		TRNCNT[7]	TRNCNT[6]	TRNCNT[5]	TRNCNT[4]	TRNCNT[3]	TRNCNT[2]	TRNCNT[1]	TRNCNT[0]	
	PIPEATRE_0	-	-	-	-	-	-	-	TRENB	TRCLR
		-	-	-	-	-	-	-	-	-
	PIPEATRN_0	TRNCNT[15]	TRNCNT[14]	TRNCNT[13]	TRNCNT[12]	TRNCNT[11]	TRNCNT[10]	TRNCNT[9]	TRNCNT[8]	
		TRNCNT[7]	TRNCNT[6]	TRNCNT[5]	TRNCNT[4]	TRNCNT[3]	TRNCNT[2]	TRNCNT[1]	TRNCNT[0]	
	DEVADD0_0	-	UPPHUB[3]	UPPHUB[2]	UPPHUB[1]	UPPHUB[0]	HUBPORT[2]	HUBPORT[1]	HUBPORT[0]	
		USBSPD[1]	USBSPD[0]	-	-	-	-	-	-	
	DEVADD1_0	-	UPPHUB[3]	UPPHUB[2]	UPPHUB[1]	UPPHUB[0]	HUBPORT[2]	HUBPORT[1]	HUBPORT[0]	
		USBSPD[1]	USBSPD[0]	-	-	-	-	-	-	
	DEVADD2_0	-	UPPHUB[3]	UPPHUB[2]	UPPHUB[1]	UPPHUB[0]	HUBPORT[2]	HUBPORT[1]	HUBPORT[0]	
		USBSPD[1]	USBSPD[0]	-	-	-	-	-	-	
	DEVADD3_0	-	UPPHUB[3]	UPPHUB[2]	UPPHUB[1]	UPPHUB[0]	HUBPORT[2]	HUBPORT[1]	HUBPORT[0]	
		USBSPD[1]	USBSPD[0]	-	-	-	-	-	-	
	DEVADD4_0	-	UPPHUB[3]	UPPHUB[2]	UPPHUB[1]	UPPHUB[0]	HUBPORT[2]	HUBPORT[1]	HUBPORT[0]	
		USBSPD[1]	USBSPD[0]	-	-	-	-	-	-	
	DEVADD5_0	-	UPPHUB[3]	UPPHUB[2]	UPPHUB[1]	UPPHUB[0]	HUBPORT[2]	HUBPORT[1]	HUBPORT[0]	
		USBSPD[1]	USBSPD[0]	-	-	-	-	-	-	
	DEVADD6_0	-	HPPHUB[3]	HPPHUB[2]	HPPHUB[1]	HPPHUB[0]	HUBPORT[2]	HUBPORT[1]	HUBPORT[0]	
		USBSPD[1]	UPPHUB[3]	UPPHUB[2]	UPPHUB[1]	UPPHUB[0]	-	-	-	
	DEVADD7_0	-	HPPHUB[3]	HPPHUB[2]	HPPHUB[1]	HPPHUB[0]	HUBPORT[2]	HUBPORT[1]	HUBPORT[0]	
		USBSPD[1]	USBSPD[0]	-	-	-	-	-	-	
	DEVADD8_0	-	UPPHUB[3]	UPPHUB[2]	UPPHUB[1]	UPPHUB[0]	HUBPORT[2]	HUBPORT[1]	HUBPORT[0]	
		USBSPD[1]	USBSPD[0]	-	-	-	-	-	-	
	DEVADD9_0	-	UPPHUB[3]	UPPHUB[2]	UPPHUB[1]	UPPHUB[0]	HUBPORT[2]	HUBPORT[1]	HUBPORT[0]	
		USBSPD[1]	USBSPD[0]	-	-	-	-	-	-	
	DEVADDA_0	-	UPPHUB[3]	UPPHUB[2]	UPPHUB[1]	UPPHUB[0]	HUBPORT[2]	HUBPORT[1]	HUBPORT[0]	
		USBSPD[1]	USBSPD[0]	-	-	-	-	-	-	
	SUSPMODE_0	-	SUSPM	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
	DOFIFOB0_0	FIFOPORT[31]	FIFOPORT[30]	FIFOPORT[29]	FIFOPORT[28]	FIFOPORT[27]	FIFOPORT[26]	FIFOPORT[25]	FIFOPORT[24]	
		FIFOPORT[23]	FIFOPORT[22]	FIFOPORT[21]	FIFOPORT[20]	FIFOPORT[19]	FIFOPORT[18]	FIFOPORT[17]	FIFOPORT[16]	
		FIFOPORT[15]	FIFOPORT[14]	FIFOPORT[13]	FIFOPORT[12]	FIFOPORT[11]	FIFOPORT[10]	FIFOPORT[9]	FIFOPORT[8]	
		FIFOPORT[7]	FIFOPORT[6]	FIFOPORT[5]	FIFOPORT[4]	FIFOPORT[3]	FIFOPORT[2]	FIFOPORT[1]	FIFOPORT[0]	
	DOFIFOB1_0	FIFOPORT[31]	FIFOPORT[30]	FIFOPORT[29]	FIFOPORT[28]	FIFOPORT[27]	FIFOPORT[26]	FIFOPORT[25]	FIFOPORT[24]	
		FIFOPORT[23]	FIFOPORT[22]	FIFOPORT[21]	FIFOPORT[20]	FIFOPORT[19]	FIFOPORT[18]	FIFOPORT[17]	FIFOPORT[16]	
		FIFOPORT[15]	FIFOPORT[14]	FIFOPORT[13]	FIFOPORT[12]	FIFOPORT[11]	FIFOPORT[10]	FIFOPORT[9]	FIFOPORT[8]	
		FIFOPORT[7]	FIFOPORT[6]	FIFOPORT[5]	FIFOPORT[4]	FIFOPORT[3]	FIFOPORT[2]	FIFOPORT[1]	FIFOPORT[0]	
	DOFIFOB2_0	FIFOPORT[31]	FIFOPORT[30]	FIFOPORT[29]	FIFOPORT[28]	FIFOPORT[27]	FIFOPORT[26]	FIFOPORT[25]	FIFOPORT[24]	
		FIFOPORT[23]	FIFOPORT[22]	FIFOPORT[21]	FIFOPORT[20]	FIFOPORT[19]	FIFOPORT[18]	FIFOPORT[17]	FIFOPORT[16]	
		FIFOPORT[15]	FIFOPORT[14]	FIFOPORT[13]	FIFOPORT[12]	FIFOPORT[11]	FIFOPORT[10]	FIFOPORT[9]	FIFOPORT[8]	
		FIFOPORT[7]	FIFOPORT[6]	FIFOPORT[5]	FIFOPORT[4]	FIFOPORT[3]	FIFOPORT[2]	FIFOPORT[1]	FIFOPORT[0]	
DOFIFOB3_0	FIFOPORT[31]	FIFOPORT[30]	FIFOPORT[29]	FIFOPORT[28]	FIFOPORT[27]	FIFOPORT[26]	FIFOPORT[25]	FIFOPORT[24]		
	FIFOPORT[23]	FIFOPORT[22]	FIFOPORT[21]	FIFOPORT[20]	FIFOPORT[19]	FIFOPORT[18]	FIFOPORT[17]	FIFOPORT[16]		
	FIFOPORT[15]	FIFOPORT[14]	FIFOPORT[13]	FIFOPORT[12]	FIFOPORT[11]	FIFOPORT[10]	FIFOPORT[9]	FIFOPORT[8]		
	FIFOPORT[7]	FIFOPORT[6]	FIFOPORT[5]	FIFOPORT[4]	FIFOPORT[3]	FIFOPORT[2]	FIFOPORT[1]	FIFOPORT[0]		
DOFIFOB4_0	FIFOPORT[31]	FIFOPORT[30]	FIFOPORT[29]	FIFOPORT[28]	FIFOPORT[27]	FIFOPORT[26]	FIFOPORT[25]	FIFOPORT[24]		
	FIFOPORT[23]	FIFOPORT[22]	FIFOPORT[21]	FIFOPORT[20]	FIFOPORT[19]	FIFOPORT[18]	FIFOPORT[17]	FIFOPORT[16]		
	FIFOPORT[15]	FIFOPORT[14]	FIFOPORT[13]	FIFOPORT[12]	FIFOPORT[11]	FIFOPORT[10]	FIFOPORT[9]	FIFOPORT[8]		
	FIFOPORT[7]	FIFOPORT[6]	FIFOPORT[5]	FIFOPORT[4]	FIFOPORT[3]	FIFOPORT[2]	FIFOPORT[1]	FIFOPORT[0]		



Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
USB2.0 host/ function module	D0FIFOB5_0	FIFOPORT[31]	FIFOPORT[30]	FIFOPORT[29]	FIFOPORT[28]	FIFOPORT[27]	FIFOPORT[26]	FIFOPORT[25]	FIFOPORT[24]	
		FIFOPORT[23]	FIFOPORT[22]	FIFOPORT[21]	FIFOPORT[20]	FIFOPORT[19]	FIFOPORT[18]	FIFOPORT[17]	FIFOPORT[16]	
		FIFOPORT[15]	FIFOPORT[14]	FIFOPORT[13]	FIFOPORT[12]	FIFOPORT[11]	FIFOPORT[10]	FIFOPORT[9]	FIFOPORT[8]	
		FIFOPORT[7]	FIFOPORT[6]	FIFOPORT[5]	FIFOPORT[4]	FIFOPORT[3]	FIFOPORT[2]	FIFOPORT[1]	FIFOPORT[0]	
	D0FIFOB6_0	FIFOPORT[31]	FIFOPORT[30]	FIFOPORT[29]	FIFOPORT[28]	FIFOPORT[27]	FIFOPORT[26]	FIFOPORT[25]	FIFOPORT[24]	
		FIFOPORT[23]	FIFOPORT[22]	FIFOPORT[21]	FIFOPORT[20]	FIFOPORT[19]	FIFOPORT[18]	FIFOPORT[17]	FIFOPORT[16]	
		FIFOPORT[15]	FIFOPORT[14]	FIFOPORT[13]	FIFOPORT[12]	FIFOPORT[11]	FIFOPORT[10]	FIFOPORT[9]	FIFOPORT[8]	
		FIFOPORT[7]	FIFOPORT[6]	FIFOPORT[5]	FIFOPORT[4]	FIFOPORT[3]	FIFOPORT[2]	FIFOPORT[1]	FIFOPORT[0]	
	D0FIFOB7_0	FIFOPORT[31]	FIFOPORT[30]	FIFOPORT[29]	FIFOPORT[28]	FIFOPORT[27]	FIFOPORT[26]	FIFOPORT[25]	FIFOPORT[24]	
		FIFOPORT[23]	FIFOPORT[22]	FIFOPORT[21]	FIFOPORT[20]	FIFOPORT[19]	FIFOPORT[18]	FIFOPORT[17]	FIFOPORT[16]	
		FIFOPORT[15]	FIFOPORT[14]	FIFOPORT[13]	FIFOPORT[12]	FIFOPORT[11]	FIFOPORT[10]	FIFOPORT[9]	FIFOPORT[8]	
		FIFOPORT[7]	FIFOPORT[6]	FIFOPORT[5]	FIFOPORT[4]	FIFOPORT[3]	FIFOPORT[2]	FIFOPORT[1]	FIFOPORT[0]	
	D1FIFOB0_0	FIFOPORT[31]	FIFOPORT[30]	FIFOPORT[29]	FIFOPORT[28]	FIFOPORT[27]	FIFOPORT[26]	FIFOPORT[25]	FIFOPORT[24]	
		FIFOPORT[23]	FIFOPORT[22]	FIFOPORT[21]	FIFOPORT[20]	FIFOPORT[19]	FIFOPORT[18]	FIFOPORT[17]	FIFOPORT[16]	
		FIFOPORT[15]	FIFOPORT[14]	FIFOPORT[13]	FIFOPORT[12]	FIFOPORT[11]	FIFOPORT[10]	FIFOPORT[9]	FIFOPORT[8]	
		FIFOPORT[7]	FIFOPORT[6]	FIFOPORT[5]	FIFOPORT[4]	FIFOPORT[3]	FIFOPORT[2]	FIFOPORT[1]	FIFOPORT[0]	
	D1FIFOB1_0	FIFOPORT[31]	FIFOPORT[30]	FIFOPORT[29]	FIFOPORT[28]	FIFOPORT[27]	FIFOPORT[26]	FIFOPORT[25]	FIFOPORT[24]	
		FIFOPORT[23]	FIFOPORT[22]	FIFOPORT[21]	FIFOPORT[20]	FIFOPORT[19]	FIFOPORT[18]	FIFOPORT[17]	FIFOPORT[16]	
		FIFOPORT[15]	FIFOPORT[14]	FIFOPORT[13]	FIFOPORT[12]	FIFOPORT[11]	FIFOPORT[10]	FIFOPORT[9]	FIFOPORT[8]	
		FIFOPORT[7]	FIFOPORT[6]	FIFOPORT[5]	FIFOPORT[4]	FIFOPORT[3]	FIFOPORT[2]	FIFOPORT[1]	FIFOPORT[0]	
	D1FIFOB2_0	FIFOPORT[31]	FIFOPORT[30]	FIFOPORT[29]	FIFOPORT[28]	FIFOPORT[27]	FIFOPORT[26]	FIFOPORT[25]	FIFOPORT[24]	
		FIFOPORT[23]	FIFOPORT[22]	FIFOPORT[21]	FIFOPORT[20]	FIFOPORT[19]	FIFOPORT[18]	FIFOPORT[17]	FIFOPORT[16]	
		FIFOPORT[15]	FIFOPORT[14]	FIFOPORT[13]	FIFOPORT[12]	FIFOPORT[11]	FIFOPORT[10]	FIFOPORT[9]	FIFOPORT[8]	
		FIFOPORT[7]	FIFOPORT[6]	FIFOPORT[5]	FIFOPORT[4]	FIFOPORT[3]	FIFOPORT[2]	FIFOPORT[1]	FIFOPORT[0]	
	D1FIFOB3_0	FIFOPORT[31]	FIFOPORT[30]	FIFOPORT[29]	FIFOPORT[28]	FIFOPORT[27]	FIFOPORT[26]	FIFOPORT[25]	FIFOPORT[24]	
		FIFOPORT[23]	FIFOPORT[22]	FIFOPORT[21]	FIFOPORT[20]	FIFOPORT[19]	FIFOPORT[18]	FIFOPORT[17]	FIFOPORT[16]	
		FIFOPORT[15]	FIFOPORT[14]	FIFOPORT[13]	FIFOPORT[12]	FIFOPORT[11]	FIFOPORT[10]	FIFOPORT[9]	FIFOPORT[8]	
		FIFOPORT[7]	FIFOPORT[6]	FIFOPORT[5]	FIFOPORT[4]	FIFOPORT[3]	FIFOPORT[2]	FIFOPORT[1]	FIFOPORT[0]	
	D1FIFOB4_0	FIFOPORT[31]	FIFOPORT[30]	FIFOPORT[29]	FIFOPORT[28]	FIFOPORT[27]	FIFOPORT[26]	FIFOPORT[25]	FIFOPORT[24]	
		FIFOPORT[23]	FIFOPORT[22]	FIFOPORT[21]	FIFOPORT[20]	FIFOPORT[19]	FIFOPORT[18]	FIFOPORT[17]	FIFOPORT[16]	
		FIFOPORT[15]	FIFOPORT[14]	FIFOPORT[13]	FIFOPORT[12]	FIFOPORT[11]	FIFOPORT[10]	FIFOPORT[9]	FIFOPORT[8]	
		FIFOPORT[7]	FIFOPORT[6]	FIFOPORT[5]	FIFOPORT[4]	FIFOPORT[3]	FIFOPORT[2]	FIFOPORT[1]	FIFOPORT[0]	
	D1FIFOB5_0	FIFOPORT[31]	FIFOPORT[30]	FIFOPORT[29]	FIFOPORT[28]	FIFOPORT[27]	FIFOPORT[26]	FIFOPORT[25]	FIFOPORT[24]	
		FIFOPORT[23]	FIFOPORT[22]	FIFOPORT[21]	FIFOPORT[20]	FIFOPORT[19]	FIFOPORT[18]	FIFOPORT[17]	FIFOPORT[16]	
		FIFOPORT[15]	FIFOPORT[14]	FIFOPORT[13]	FIFOPORT[12]	FIFOPORT[11]	FIFOPORT[10]	FIFOPORT[9]	FIFOPORT[8]	
		FIFOPORT[7]	FIFOPORT[6]	FIFOPORT[5]	FIFOPORT[4]	FIFOPORT[3]	FIFOPORT[2]	FIFOPORT[1]	FIFOPORT[0]	
	D1FIFOB6_0	FIFOPORT[31]	FIFOPORT[30]	FIFOPORT[29]	FIFOPORT[28]	FIFOPORT[27]	FIFOPORT[26]	FIFOPORT[25]	FIFOPORT[24]	
		FIFOPORT[23]	FIFOPORT[22]	FIFOPORT[21]	FIFOPORT[20]	FIFOPORT[19]	FIFOPORT[18]	FIFOPORT[17]	FIFOPORT[16]	
		FIFOPORT[15]	FIFOPORT[14]	FIFOPORT[13]	FIFOPORT[12]	FIFOPORT[11]	FIFOPORT[10]	FIFOPORT[9]	FIFOPORT[8]	
		FIFOPORT[7]	FIFOPORT[6]	FIFOPORT[5]	FIFOPORT[4]	FIFOPORT[3]	FIFOPORT[2]	FIFOPORT[1]	FIFOPORT[0]	
	D1FIFOB7_0	FIFOPORT[31]	FIFOPORT[30]	FIFOPORT[29]	FIFOPORT[28]	FIFOPORT[27]	FIFOPORT[26]	FIFOPORT[25]	FIFOPORT[24]	
		FIFOPORT[23]	FIFOPORT[22]	FIFOPORT[21]	FIFOPORT[20]	FIFOPORT[19]	FIFOPORT[18]	FIFOPORT[17]	FIFOPORT[16]	
		FIFOPORT[15]	FIFOPORT[14]	FIFOPORT[13]	FIFOPORT[12]	FIFOPORT[11]	FIFOPORT[10]	FIFOPORT[9]	FIFOPORT[8]	
		FIFOPORT[7]	FIFOPORT[6]	FIFOPORT[5]	FIFOPORT[4]	FIFOPORT[3]	FIFOPORT[2]	FIFOPORT[1]	FIFOPORT[0]	
	SYSCFG0_1	-	-	-	-	-	-	SCKE	-	-
		HSE	DCFM	DRPD	DPRPU	-	-	-	-	USBE
	BUSWAIT_1	-	-	-	-	-	-	-	-	-
		-	-	BWAIT[5]	BWAIT[4]	BWAIT[3]	BWAIT[2]	BWAIT[1]	BWAIT[0]	-
	SYSSTS0_1	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	LNST[1]	LNST[0]	-
	DVSTCTRO_1	-	-	-	-	-	-	-	-	WKUP
		RWUPE	USBRST	RESUME	UACT	-	RHST[2]	RHST[1]	RHST[0]	-
	TESTMODE_1	-	-	-	-	-	-	-	-	-
		-	-	-	-	UTST[3]	UTST[2]	UTST[1]	UTST[0]	-
	D0FBCFG_1	-	-	DFACC[1]	DFACC[0]	-	-	-	-	-
		-	-	-	TENDE	-	-	-	-	-

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
USB2.0 host/ function module	D1FBCFG_1	-	-	DFACC[1]	DFACC[0]	-	-	-	-
		-	-	-	TENDE	-	-	-	-
	CFIFO_1	FIFOPORT[31]	FIFOPORT[30]	FIFOPORT[29]	FIFOPORT[28]	FIFOPORT[27]	FIFOPORT[26]	FIFOPORT[25]	FIFOPORT[24]
		FIFOPORT[23]	FIFOPORT[22]	FIFOPORT[21]	FIFOPORT[20]	FIFOPORT[19]	FIFOPORT[18]	FIFOPORT[17]	FIFOPORT[16]
		FIFOPORT[15]	FIFOPORT[14]	FIFOPORT[13]	FIFOPORT[12]	FIFOPORT[11]	FIFOPORT[10]	FIFOPORT[9]	FIFOPORT[8]
		FIFOPORT[7]	FIFOPORT[6]	FIFOPORT[5]	FIFOPORT[4]	FIFOPORT[3]	FIFOPORT[2]	FIFOPORT[1]	FIFOPORT[0]
	D0FIFO_1	FIFOPORT[31]	FIFOPORT[30]	FIFOPORT[29]	FIFOPORT[28]	FIFOPORT[27]	FIFOPORT[26]	FIFOPORT[25]	FIFOPORT[24]
		FIFOPORT[23]	FIFOPORT[22]	FIFOPORT[21]	FIFOPORT[20]	FIFOPORT[19]	FIFOPORT[18]	FIFOPORT[17]	FIFOPORT[16]
		FIFOPORT[15]	FIFOPORT[14]	FIFOPORT[13]	FIFOPORT[12]	FIFOPORT[11]	FIFOPORT[10]	FIFOPORT[9]	FIFOPORT[8]
		FIFOPORT[7]	FIFOPORT[6]	FIFOPORT[5]	FIFOPORT[4]	FIFOPORT[3]	FIFOPORT[2]	FIFOPORT[1]	FIFOPORT[0]
	D1FIFO_1	FIFOPORT[31]	FIFOPORT[30]	FIFOPORT[29]	FIFOPORT[28]	FIFOPORT[27]	FIFOPORT[26]	FIFOPORT[25]	FIFOPORT[24]
		FIFOPORT[23]	FIFOPORT[22]	FIFOPORT[21]	FIFOPORT[20]	FIFOPORT[19]	FIFOPORT[18]	FIFOPORT[17]	FIFOPORT[16]
		FIFOPORT[15]	FIFOPORT[14]	FIFOPORT[13]	FIFOPORT[12]	FIFOPORT[11]	FIFOPORT[10]	FIFOPORT[9]	FIFOPORT[8]
		FIFOPORT[7]	FIFOPORT[6]	FIFOPORT[5]	FIFOPORT[4]	FIFOPORT[3]	FIFOPORT[2]	FIFOPORT[1]	FIFOPORT[0]
	CFIFOSEL_1	RCNT	REW	-	-	MBW[1]	MBW[0]	-	BIGEND
		-	-	ISEL	-	CURPIPE[3]	CURPIPE[2]	CURPIPE[1]	CURPIPE[0]
	CFIFOCTR_1	BVAL	BCLR	FRDY	-	DTLN[11]	DTLN[10]	DTLN[9]	DTLN[8]
		DTLN[7]	DTLN[6]	DTLN[5]	DTLN[4]	DTLN[3]	DTLN[2]	DTLN[1]	DTLN[0]
	D0FIFOSEL_1	RCNT	REW	DCLRM	DREQE	MBW[1]	MBW[0]	-	BIGEND
		-	-	-	-	CURPIPE[3]	CURPIPE[2]	CURPIPE[1]	CURPIPE[0]
	D0FIFOCTR_1	BVAL	BCLR	FRDY	-	DTLN[11]	DTLN[10]	DTLN[9]	DTLN[8]
		DTLN[7]	DTLN[6]	DTLN[5]	DTLN[4]	DTLN[3]	DTLN[2]	DTLN[1]	DTLN[0]
	D1FIFOSEL_1	RCNT	REW	DCLRM	DREQE	MBW[1]	MBW[0]	-	BIGEND
		-	-	-	-	CURPIPE[3]	CURPIPE[2]	CURPIPE[1]	CURPIPE[0]
	D1FIFOCTR_1	BVAL	BCLR	FRDY	-	DTLN[11]	DTLN[10]	DTLN[9]	DTLN[8]
		DTLN[7]	DTLN[6]	DTLN[5]	DTLN[4]	DTLN[3]	DTLN[2]	DTLN[1]	DTLN[0]
	INTENB0_1	VBSE	RSME	SOFE	DVSE	CTRE	BEMPE	NRDYE	BRDYE
		-	-	-	-	-	-	-	-
	INTENB1_1	-	BCHGE	-	DTCHE	ATTICHE	-	-	-
		-	EOFERRE	SIGNE	SACKE	-	-	-	-
	BRDYENB_1	PIPEBRDYE[15]	PIPEBRDYE[14]	PIPEBRDYE[13]	PIPEBRDYE[12]	PIPEBRDYE[11]	PIPEBRDYE[10]	PIPEBRDYE[9]	PIPEBRDYE[8]
		PIPEBRDYE[7]	PIPEBRDYE[6]	PIPEBRDYE[5]	PIPEBRDYE[4]	PIPEBRDYE[3]	PIPEBRDYE[2]	PIPEBRDYE[1]	PIPEBRDYE[0]
	NRDYENB_1	PIPENRDYE[15]	PIPENRDYE[14]	PIPENRDYE[13]	PIPENRDYE[12]	PIPENRDYE[11]	PIPENRDYE[10]	PIPENRDYE[9]	PIPENRDYE[8]
		PIPENRDYE[7]	PIPENRDYE[6]	PIPENRDYE[5]	PIPENRDYE[4]	PIPENRDYE[3]	PIPENRDYE[2]	PIPENRDYE[1]	PIPENRDYE[0]
	BEMPENB_1	PIPEBEMPE[15]	PIPEBEMPE[14]	PIPEBEMPE[13]	PIPEBEMPE[12]	PIPEBEMPE[11]	PIPEBEMPE[10]	PIPEBEMPE[9]	PIPEBEMPE[8]
		PIPEBEMPE[7]	PIPEBEMPE[6]	PIPEBEMPE[5]	PIPEBEMPE[4]	PIPEBEMPE[3]	PIPEBEMPE[2]	PIPEBEMPE[1]	PIPEBEMPE[0]
	SOFCFG_1	-	-	-	-	-	-	-	TRNENSEL
		-	BRDYM	-	-	-	-	-	-
	INTSTS0_1	VBINT	RESM	SOFR	DVST	CTRT	BEMP	NRDY	BRDY
		VBSTS	DVSQ[2]	DVSQ[1]	DVSQ[0]	VALID	CTSQ[2]	CTSQ[1]	CTSQ[0]
	INTSTS1_1	-	BCHG	-	DTCH	ATTCH	-	-	-
		-	EOFERR	SIGN	SACK	-	-	-	-
	BRDYSTS_1	PIPEBRDY[15]	PIPEBRDY[14]	PIPEBRDY[13]	PIPEBRDY[12]	PIPEBRDY[11]	PIPEBRDY[10]	PIPEBRDY[9]	PIPEBRDY[8]
		PIPEBRDY[7]	PIPEBRDY[6]	PIPEBRDY[5]	PIPEBRDY[4]	PIPEBRDY[3]	PIPEBRDY[2]	PIPEBRDY[1]	PIPEBRDY[0]
	NRDYSTS_1	PIPENRDY[15]	PIPENRDY[14]	PIPENRDY[13]	PIPENRDY[12]	PIPENRDY[11]	PIPENRDY[10]	PIPENRDY[9]	PIPENRDY[8]
		PIPENRDY[7]	PIPENRDY[6]	PIPENRDY[5]	PIPENRDY[4]	PIPENRDY[3]	PIPENRDY[2]	PIPENRDY[1]	PIPENRDY[0]
	BEMPSTS_1	PIPEBEMP[15]	PIPEBEMP[14]	PIPEBEMP[13]	PIPEBEMP[12]	PIPEBEMP[11]	PIPEBEMP[10]	PIPEBEMP[9]	PIPEBEMP[8]
		PIPEBEMP[7]	PIPEBEMP[6]	PIPEBEMP[5]	PIPEBEMP[4]	PIPEBEMP[3]	PIPEBEMP[2]	PIPEBEMP[1]	PIPEBEMP[0]
	FRMNUM_1	OVRN	CRCE	-	-	-	FRNM[10]	FRNM[9]	FRNM[8]
		FRNM[7]	FRNM[6]	FRNM[5]	FRNM[4]	FRNM[3]	FRNM[2]	FRNM[1]	FRNM[0]
	UFRMNUM_1	-	-	-	-	-	-	-	-
		-	-	-	-	-	UFRNM[2]	UFRNM[1]	UFRNM[0]
	USBADDR_1	-	-	-	-	-	-	-	-
		-	USBADDR[6]	USBADDR[5]	USBADDR[4]	USBADDR[3]	USBADDR[2]	USBADDR[1]	USBADDR[0]
	USBREQ_1	bRequest[7]	bRequest[6]	bRequest[5]	bRequest[4]	bRequest[3]	bRequest[2]	bRequest[1]	bRequest[0]
		bmRequestType[7]	bmRequestType[6]	bmRequestType[5]	bmRequestType[4]	bmRequestType[3]	bmRequestType[2]	bmRequestType[1]	bmRequestType[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
USB2.0 host/ function module	USBVAL_1	wValue[15]	wValue[14]	wValue[13]	wValue[12]	wValue[11]	wValue[10]	wValue[9]	wValue[8]
		wValue[7]	wValue[6]	wValue[5]	wValue[4]	wValue[3]	wValue[2]	wValue[1]	wValue[0]
	USBINDX_1	wIndex[15]	wIndex[14]	wIndex[13]	wIndex[12]	wIndex[11]	wIndex[10]	wIndex[9]	wIndex[8]
		wIndex[7]	wIndex[6]	wIndex[5]	wIndex[4]	wIndex[3]	wIndex[2]	wIndex[1]	wIndex[0]
	USBLENG_1	wLength[15]	wLength[14]	wLength[13]	wLength[12]	wLength[11]	wLength[10]	wLength[9]	wLength[8]
		wLength[7]	wLength[6]	wLength[5]	wLength[4]	wLength[3]	wLength[2]	wLength[1]	wLength[0]
	DCPCFG_1	-	-	-	-	-	-	-	CNTMD
		SHTNAK	-	-	DIR	-	-	-	-
	DCPMAXP_1	DEVSEL[3]	DEVSEL[2]	DEVSEL[1]	DEVSEL[0]	-	-	-	-
		-	MXPS[6]	MXPS[5]	MXPS[4]	MXPS[3]	MXPS[2]	MXPS[1]	MXPS[0]
DPCCTR_1	BSTS	SUREQ	CSCLR	CSSTS	SUREQCLR	-	-	SQCLR	
	SQSET	SQMON	PBUSY	PINGE	-	CCPL	PID[1]	PID[0]	
PIPESEL_1	-	-	-	-	-	-	-	-	
	-	-	-	-	PIPESEL[3]	PIPESEL[2]	PIPESEL[1]	PIPESEL[0]	
PIPECFG_1	TYPE[1]	TYPE[0]	-	-	-	BFRE	DBLB	CNTMD	
	SHTNAK	-	-	DIR	EPNUM[3]	EPNUM[2]	EPNUM[1]	EPNUM[0]	
PIPEBUF_1	-	BUFSIZE[4]	BUFSIZE[3]	BUFSIZE[2]	BUFSIZE[1]	BUFSIZE[0]	-	-	
	BUFNMB[7]	BUFNMB[6]	BUFNMB[5]	BUFNMB[4]	BUFNMB[3]	BUFNMB[2]	BUFNMB[1]	BUFNMB[0]	
PEPEMAXP_1	DEVSEL[3]	DEVSEL[2]	DEVSEL[1]	DEVSEL[0]	-	MXPS[10]	MXPS[9]	MXPS[8]	
	MXPS[7]	MXPS[6]	MXPS[5]	MXPS[4]	MXPS[3]	MXPS[2]	MXPS[1]	MXPS[0]	
PIPEPERI_1	-	-	-	IFIS	-	-	-	-	
	-	-	-	-	-	IITV[2]	IITV[1]	IITV[0]	
PIPE1CTR_1	BSTS	INBUFM	CSCLR	CSSTS	-	ATREPM	ACLRM	SQCLR	
	SQSET	SQMON	PBUSY	-	-	-	PID[1]	PID[0]	
PIPE2CTR_1	BSTS	INBUFM	CSCLR	CSSTS	-	ATREPM	ACLRM	SQCLR	
	SQSET	SQMON	PBUSY	-	-	-	PID[1]	PID[0]	
PIPE3CTR_1	BSTS	INBUFM	CSCLR	CSSTS	-	ATREPM	ACLRM	SQCLR	
	SQSET	SQMON	PBUSY	-	-	-	PID[1]	PID[0]	
PIPE4CTR_1	BSTS	INBUFM	CSCLR	CSSTS	-	ATREPM	ACLRM	SQCLR	
	SQSET	SQMON	PBUSY	-	-	-	PID[1]	PID[0]	
PIPE5CTR_1	BSTS	INBUFM	CSCLR	CSSTS	-	ATREPM	ACLRM	SQCLR	
	SQSET	SQMON	PBUSY	-	-	-	PID[1]	PID[0]	
PIPE6CTR_1	BSTS	-	CSCLR	CSSTS	-	-	ACLRM	SQCLR	
	SQSET	SQMON	PBUSY	-	-	-	PID[1]	PID[0]	
PIPE7CTR_1	BSTS	-	CSCLR	CSSTS	-	-	ACLRM	SQCLR	
	SQSET	SQMON	PBUSY	-	-	-	PID[1]	PID[0]	
PIPE8CTR_1	BSTS	-	CSCLR	CSSTS	-	-	ACLRM	SQCLR	
	SQSET	SQMON	PBUSY	-	-	-	PID[1]	PID[0]	
PIPE9CTR_1	BSTS	INBUFM	CSCLR	CSSTS	-	ATREPM	ACLRM	SQCLR	
	SQSET	SQMON	PBUSY	-	-	-	PID[1]	PID[0]	
PIPEACTR_1	BSTS	INBUFM	CSCLR	CSSTS	-	ATREPM	ACLRM	SQCLR	
	SQSET	SQMON	PBUSY	-	-	-	PID[1]	PID[0]	
PIPEBCTR_1	BSTS	INBUFM	CSCLR	CSSTS	-	ATREPM	ACLRM	SQCLR	
	SQSET	SQMON	PBUSY	-	-	-	PID[1]	PID[0]	
PIPECCTR_1	BSTS	INBUFM	CSCLR	CSSTS	-	ATREPM	ACLRM	SQCLR	
	SQSET	SQMON	PBUSY	-	-	-	PID[1]	PID[0]	
PIPEDCTR_1	BSTS	INBUFM	CSCLR	CSSTS	-	ATREPM	ACLRM	SQCLR	
	SQSET	SQMON	PBUSY	-	-	-	PID[1]	PID[0]	
PIPEECTR_1	BSTS	INBUFM	CSCLR	CSSTS	-	ATREPM	ACLRM	SQCLR	
	SQSET	SQMON	PBUSY	-	-	-	PID[1]	PID[0]	
PIPEFCTR_1	BSTS	INBUFM	CSCLR	CSSTS	-	ATREPM	ACLRM	SQCLR	
	SQSET	SQMON	PBUSY	-	-	-	PID[1]	PID[0]	
PIPE1TRE_1	-	-	-	-	-	-	TRENB	TRCLR	
	-	-	-	-	-	-	-	-	
PIPE1TRN_1	TRNCNT[15]	TRNCNT[14]	TRNCNT[13]	TRNCNT[12]	TRNCNT[11]	TRNCNT[10]	TRNCNT[9]	TRNCNT[8]	
	TRNCNT[7]	TRNCNT[6]	TRNCNT[5]	TRNCNT[4]	TRNCNT[3]	TRNCNT[2]	TRNCNT[1]	TRNCNT[0]	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
USB2.0 host/ function module	PIPE2TRE_1	-	-	-	-	-	-	TRENB	TRCLR
		-	-	-	-	-	-	-	-
	PIPE2TRN_1	TRNCNT[15]	TRNCNT[14]	TRNCNT[13]	TRNCNT[12]	TRNCNT[11]	TRNCNT[10]	TRNCNT[9]	TRNCNT[8]
		TRNCNT[7]	TRNCNT[6]	TRNCNT[5]	TRNCNT[4]	TRNCNT[3]	TRNCNT[2]	TRNCNT[1]	TRNCNT[0]
	PIPE3TRE_1	-	-	-	-	-	-	TRENB	TRCLR
		-	-	-	-	-	-	-	-
	PIPE3TRN_1	TRNCNT[15]	TRNCNT[14]	TRNCNT[13]	TRNCNT[12]	TRNCNT[11]	TRNCNT[10]	TRNCNT[9]	TRNCNT[8]
		TRNCNT[7]	TRNCNT[6]	TRNCNT[5]	TRNCNT[4]	TRNCNT[3]	TRNCNT[2]	TRNCNT[1]	TRNCNT[0]
	PIPE4TRE_1	-	-	-	-	-	-	TRENB	TRCLR
		-	-	-	-	-	-	-	-
	PIPE4TRN_1	TRNCNT[15]	TRNCNT[14]	TRNCNT[13]	TRNCNT[12]	TRNCNT[11]	TRNCNT[10]	TRNCNT[9]	TRNCNT[8]
		TRNCNT[7]	TRNCNT[6]	TRNCNT[5]	TRNCNT[4]	TRNCNT[3]	TRNCNT[2]	TRNCNT[1]	TRNCNT[0]
	PIPE5TRE_1	-	-	-	-	-	-	TRENB	TRCLR
		-	-	-	-	-	-	-	-
	PIPE5TRN_1	TRNCNT[15]	TRNCNT[14]	TRNCNT[13]	TRNCNT[12]	TRNCNT[11]	TRNCNT[10]	TRNCNT[9]	TRNCNT[8]
		TRNCNT[7]	TRNCNT[6]	TRNCNT[5]	TRNCNT[4]	TRNCNT[3]	TRNCNT[2]	TRNCNT[1]	TRNCNT[0]
	PIPE8TRE_1	-	-	-	-	-	-	TRENB	TRCLR
		-	-	-	-	-	-	-	-
	PIPE8TRN_1	TRNCNT[15]	TRNCNT[14]	TRNCNT[13]	TRNCNT[12]	TRNCNT[11]	TRNCNT[10]	TRNCNT[9]	TRNCNT[8]
		TRNCNT[7]	TRNCNT[6]	TRNCNT[5]	TRNCNT[4]	TRNCNT[3]	TRNCNT[2]	TRNCNT[1]	TRNCNT[0]
	PIPECTRE_1	-	-	-	-	-	-	TRENB	TRCLR
		-	-	-	-	-	-	-	-
	PIPECTRN_1	TRNCNT[15]	TRNCNT[14]	TRNCNT[13]	TRNCNT[12]	TRNCNT[11]	TRNCNT[10]	TRNCNT[9]	TRNCNT[8]
		TRNCNT[7]	TRNCNT[6]	TRNCNT[5]	TRNCNT[4]	TRNCNT[3]	TRNCNT[2]	TRNCNT[1]	TRNCNT[0]
	PIPEDTRE_1	-	-	-	-	-	-	TRENB	TRCLR
		-	-	-	-	-	-	-	-
	PIPEDTRN_1	TRNCNT[15]	TRNCNT[14]	TRNCNT[13]	TRNCNT[12]	TRNCNT[11]	TRNCNT[10]	TRNCNT[9]	TRNCNT[8]
		TRNCNT[7]	TRNCNT[6]	TRNCNT[5]	TRNCNT[4]	TRNCNT[3]	TRNCNT[2]	TRNCNT[1]	TRNCNT[0]
	PIPEETRE_1	-	-	-	-	-	-	TRENB	TRCLR
		-	-	-	-	-	-	-	-
	PIPEETRN_1	TRNCNT[15]	TRNCNT[14]	TRNCNT[13]	TRNCNT[12]	TRNCNT[11]	TRNCNT[10]	TRNCNT[9]	TRNCNT[8]
		TRNCNT[7]	TRNCNT[6]	TRNCNT[5]	TRNCNT[4]	TRNCNT[3]	TRNCNT[2]	TRNCNT[1]	TRNCNT[0]
	PIPEFTRE_1	-	-	-	-	-	-	TRENB	TRCLR
		-	-	-	-	-	-	-	-
	PIPEFTRN_1	TRNCNT[15]	TRNCNT[14]	TRNCNT[13]	TRNCNT[12]	TRNCNT[11]	TRNCNT[10]	TRNCNT[9]	TRNCNT[8]
		TRNCNT[7]	TRNCNT[6]	TRNCNT[5]	TRNCNT[4]	TRNCNT[3]	TRNCNT[2]	TRNCNT[1]	TRNCNT[0]
	PIPE9TRE_1	-	-	-	-	-	-	TRENB	TRCLR
		-	-	-	-	-	-	-	-
	PIPE9TRN_1	TRNCNT[15]	TRNCNT[14]	TRNCNT[13]	TRNCNT[12]	TRNCNT[11]	TRNCNT[10]	TRNCNT[9]	TRNCNT[8]
		TRNCNT[7]	TRNCNT[6]	TRNCNT[5]	TRNCNT[4]	TRNCNT[3]	TRNCNT[2]	TRNCNT[1]	TRNCNT[0]
	PIPEATRE_1	-	-	-	-	-	-	TRENB	TRCLR
		-	-	-	-	-	-	-	-
	PIPEATRN_1	TRNCNT[15]	TRNCNT[14]	TRNCNT[13]	TRNCNT[12]	TRNCNT[11]	TRNCNT[10]	TRNCNT[9]	TRNCNT[8]
		TRNCNT[7]	TRNCNT[6]	TRNCNT[5]	TRNCNT[4]	TRNCNT[3]	TRNCNT[2]	TRNCNT[1]	TRNCNT[0]
	DEVADD0_1	-	UPPHUB[3]	UPPHUB[2]	UPPHUB[1]	UPPHUB[0]	HUBPORT[2]	HUBPORT[1]	HUBPORT[0]
		USBSPD[1]	USBSPD[0]	-	-	-	-	-	-
	DEVADD1_1	-	UPPHUB[3]	UPPHUB[2]	UPPHUB[1]	UPPHUB[0]	HUBPORT[2]	HUBPORT[1]	HUBPORT[0]
		USBSPD[1]	USBSPD[0]	-	-	-	-	-	-
	DEVADD2_1	-	UPPHUB[3]	UPPHUB[2]	UPPHUB[1]	UPPHUB[0]	HUBPORT[2]	HUBPORT[1]	HUBPORT[0]
		USBSPD[1]	USBSPD[0]	-	-	-	-	-	-
	DEVADD3_1	-	UPPHUB[3]	UPPHUB[2]	UPPHUB[1]	UPPHUB[0]	HUBPORT[2]	HUBPORT[1]	HUBPORT[0]
		USBSPD[1]	USBSPD[0]	-	-	-	-	-	-
	DEVADD4_1	-	UPPHUB[3]	UPPHUB[2]	UPPHUB[1]	UPPHUB[0]	HUBPORT[2]	HUBPORT[1]	HUBPORT[0]
		USBSPD[1]	USBSPD[0]	-	-	-	-	-	-
	DEVADD5_1	-	UPPHUB[3]	UPPHUB[2]	UPPHUB[1]	UPPHUB[0]	HUBPORT[2]	HUBPORT[1]	HUBPORT[0]
		USBSPD[1]	USBSPD[0]	-	-	-	-	-	-



Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0		
USB2.0 host/ function module	D1FIFOB3_1	FIFOPORT[31]	FIFOPORT[30]	FIFOPORT[29]	FIFOPORT[28]	FIFOPORT[27]	FIFOPORT[26]	FIFOPORT[25]	FIFOPORT[24]		
		FIFOPORT[23]	FIFOPORT[22]	FIFOPORT[21]	FIFOPORT[20]	FIFOPORT[19]	FIFOPORT[18]	FIFOPORT[17]	FIFOPORT[16]		
		FIFOPORT[15]	FIFOPORT[14]	FIFOPORT[13]	FIFOPORT[12]	FIFOPORT[11]	FIFOPORT[10]	FIFOPORT[9]	FIFOPORT[8]		
		FIFOPORT[7]	FIFOPORT[6]	FIFOPORT[5]	FIFOPORT[4]	FIFOPORT[3]	FIFOPORT[2]	FIFOPORT[1]	FIFOPORT[0]		
	D1FIFOB4_1	FIFOPORT[31]	FIFOPORT[30]	FIFOPORT[29]	FIFOPORT[28]	FIFOPORT[27]	FIFOPORT[26]	FIFOPORT[25]	FIFOPORT[24]		
		FIFOPORT[23]	FIFOPORT[22]	FIFOPORT[21]	FIFOPORT[20]	FIFOPORT[19]	FIFOPORT[18]	FIFOPORT[17]	FIFOPORT[16]		
		FIFOPORT[15]	FIFOPORT[14]	FIFOPORT[13]	FIFOPORT[12]	FIFOPORT[11]	FIFOPORT[10]	FIFOPORT[9]	FIFOPORT[8]		
		FIFOPORT[7]	FIFOPORT[6]	FIFOPORT[5]	FIFOPORT[4]	FIFOPORT[3]	FIFOPORT[2]	FIFOPORT[1]	FIFOPORT[0]		
	D1FIFOB5_1	FIFOPORT[31]	FIFOPORT[30]	FIFOPORT[29]	FIFOPORT[28]	FIFOPORT[27]	FIFOPORT[26]	FIFOPORT[25]	FIFOPORT[24]		
		FIFOPORT[23]	FIFOPORT[22]	FIFOPORT[21]	FIFOPORT[20]	FIFOPORT[19]	FIFOPORT[18]	FIFOPORT[17]	FIFOPORT[16]		
		FIFOPORT[15]	FIFOPORT[14]	FIFOPORT[13]	FIFOPORT[12]	FIFOPORT[11]	FIFOPORT[10]	FIFOPORT[9]	FIFOPORT[8]		
		FIFOPORT[7]	FIFOPORT[6]	FIFOPORT[5]	FIFOPORT[4]	FIFOPORT[3]	FIFOPORT[2]	FIFOPORT[1]	FIFOPORT[0]		
	D1FIFOB6_1	FIFOPORT[31]	FIFOPORT[30]	FIFOPORT[29]	FIFOPORT[28]	FIFOPORT[27]	FIFOPORT[26]	FIFOPORT[25]	FIFOPORT[24]		
		FIFOPORT[23]	FIFOPORT[22]	FIFOPORT[21]	FIFOPORT[20]	FIFOPORT[19]	FIFOPORT[18]	FIFOPORT[17]	FIFOPORT[16]		
		FIFOPORT[15]	FIFOPORT[14]	FIFOPORT[13]	FIFOPORT[12]	FIFOPORT[11]	FIFOPORT[10]	FIFOPORT[9]	FIFOPORT[8]		
		FIFOPORT[7]	FIFOPORT[6]	FIFOPORT[5]	FIFOPORT[4]	FIFOPORT[3]	FIFOPORT[2]	FIFOPORT[1]	FIFOPORT[0]		
	D1FIFOB7_1	FIFOPORT[31]	FIFOPORT[30]	FIFOPORT[29]	FIFOPORT[28]	FIFOPORT[27]	FIFOPORT[26]	FIFOPORT[25]	FIFOPORT[24]		
		FIFOPORT[23]	FIFOPORT[22]	FIFOPORT[21]	FIFOPORT[20]	FIFOPORT[19]	FIFOPORT[18]	FIFOPORT[17]	FIFOPORT[16]		
		FIFOPORT[15]	FIFOPORT[14]	FIFOPORT[13]	FIFOPORT[12]	FIFOPORT[11]	FIFOPORT[10]	FIFOPORT[9]	FIFOPORT[8]		
		FIFOPORT[7]	FIFOPORT[6]	FIFOPORT[5]	FIFOPORT[4]	FIFOPORT[3]	FIFOPORT[2]	FIFOPORT[1]	FIFOPORT[0]		
	Digital video decoder	ADCCR1_0	-	-	-	-	-	-	-	AGCMODE	
			-	-	-	-	-	-	-	-	
		TGCR1_0	-	-	-	-	-	-	-	-	SRCLEFT[8]
			SRCLEFT[7]	SRCLEFT[6]	SRCLEFT[5]	SRCLEFT[4]	SRCLEFT[3]	SRCLEFT[2]	SRCLEFT[1]	SRCLEFT[0]	-
TGCR2_0		SRCTOP[5]	SRCTOP[4]	SRCTOP[3]	SRCTOP[2]	SRCTOP[1]	SRCTOP[0]	SRCHEIGHT[9]	SRCHEIGHT[8]	SRCHEIGHT[7]	
		SRCHEIGHT[7]	SRCHEIGHT[6]	SRCHEIGHT[5]	SRCHEIGHT[4]	SRCHEIGHT[3]	SRCHEIGHT[2]	SRCHEIGHT[1]	SRCHEIGHT[0]	-	
TGCR3_0		-	-	-	-	-	-	SRCWIDTH[10]	SRCWIDTH[9]	SRCWIDTH[8]	
		SRCWIDTH[7]	SRCWIDTH[6]	SRCWIDTH[5]	SRCWIDTH[4]	SRCWIDTH[3]	SRCWIDTH[2]	SRCWIDTH[1]	SRCWIDTH[0]	-	
SYNSCR1_0		LPFVSYNC[2]	LPFVSYNC[1]	LPFVSYNC[0]	LPFHSYNC[2]	LPFHSYNC[1]	LPFHSYNC[0]	-	-	-	
		VELOCITYSHIFT_H[3]	VELOCITYSHIFT_H[2]	VELOCITYSHIFT_H[1]	VELOCITYSHIFT_H[0]	SLICERMODE_H[1]	SLICERMODE_H[0]	SLICERMODE_V[1]	SLICERMODE_V[0]	-	
SYNSCR2_0		-	-	-	-	SYNMAXDUTY_H[5]	SYNMAXDUTY_H[4]	SYNMAXDUTY_H[3]	SYNMAXDUTY_H[2]	SYNMAXDUTY_H[1]	
		SYNMAXDUTY_H[1]	SYNMAXDUTY_H[0]	SYNMINDUTY_H[5]	SYNMINDUTY_H[4]	SYNMINDUTY_H[3]	SYNMINDUTY_H[2]	SYNMINDUTY_H[1]	SYNMINDUTY_H[0]	-	
SYNSCR3_0		-	-	SSCLIPSEL[3]	SSCLIPSEL[2]	SSCLIPSEL[1]	SSCLIPSEL[0]	CSYNCSLICE_H[9]	CSYNCSLICE_H[8]	CSYNCSLICE_H[7]	
		CSYNCSLICE_H[7]	CSYNCSLICE_H[6]	CSYNCSLICE_H[5]	CSYNCSLICE_H[4]	CSYNCSLICE_H[3]	CSYNCSLICE_H[2]	CSYNCSLICE_H[1]	CSYNCSLICE_H[0]	-	
SYNSCR4_0		-	-	-	-	SYNMAXDUTY_V[5]	SYNMAXDUTY_V[4]	SYNMAXDUTY_V[3]	SYNMAXDUTY_V[2]	SYNMAXDUTY_V[1]	
		SYNMAXDUTY_V[1]	SYNMAXDUTY_V[0]	SYNMINDUTY_V[5]	SYNMINDUTY_V[4]	SYNMINDUTY_V[3]	SYNMINDUTY_V[2]	SYNMINDUTY_V[1]	SYNMINDUTY_V[0]	-	
SYNSCR5_0		VSYNCDelay	VSYNCSLICE[4]	VSYNCSLICE[3]	VSYNCSLICE[2]	VSYNCSLICE[1]	VSYNCSLICE[0]	CSYNCSLICE_V[9]	CSYNCSLICE_V[8]	CSYNCSLICE_V[7]	
		CSYNCSLICE_V[7]	CSYNCSLICE_V[6]	CSYNCSLICE_V[5]	CSYNCSLICE_V[4]	CSYNCSLICE_V[3]	CSYNCSLICE_V[2]	CSYNCSLICE_V[1]	CSYNCSLICE_V[0]	-	
HAFCCR1_0		HAFCGAIN[3]	HAFCGAIN[2]	HAFCGAIN[1]	HAFCGAIN[0]	-	HAFCFREERUN	HAFCTYP[9]	HAFCTYP[8]	HAFCTYP[7]	
		HAFCTYP[7]	HAFCTYP[6]	HAFCTYP[5]	HAFCTYP[4]	HAFCTYP[3]	HAFCTYP[2]	HAFCTYP[1]	HAFCTYP[0]	-	
HAFCCR2_0		HAFSTART[3]	HAFSTART[2]	HAFSTART[1]	HAFSTART[0]	NOX2HOSC	DOX2HOSC	HAFCMAX[9]	HAFCMAX[8]	HAFCMAX[7]	
		HAFCMAX[7]	HAFCMAX[6]	HAFCMAX[5]	HAFCMAX[4]	HAFCMAX[3]	HAFCMAX[2]	HAFCMAX[1]	HAFCMAX[0]	-	
HAFCCR3_0		HAFCEM[3]	HAFCEM[2]	HAFCEM[1]	HAFCEM[0]	HAFCMODE[1]	HAFCMODE[0]	HAFCMIN[9]	HAFCMIN[8]	HAFCMIN[7]	
		HAFCMIN[7]	HAFCMIN[6]	HAFCMIN[5]	HAFCMIN[4]	HAFCMIN[3]	HAFCMIN[2]	HAFCMIN[1]	HAFCMIN[0]	-	
VCDWCR1_0		VCDFREERUN	NOVCD50	NOVCD60	VCDDEFAULT[1]	VCDDEFAULT[0]	VCDWINDOW[5]	VCDWINDOW[4]	VCDWINDOW[3]	VCDWINDOW[2]	
		VCDWINDOW[2]	VCDWINDOW[1]	VCDWINDOW[0]	VCDOFFSET[4]	VCDOFFSET[3]	VCDOFFSET[2]	VCDOFFSET[1]	VCDOFFSET[0]	-	
DCPCR1_0		DCPMODE_Y	-	-	-	-	DCPCHECK	-	BLANKLEVEL_Y[9]	BLANKLEVEL_Y[8]	
		BLANKLEVEL_Y[7]	BLANKLEVEL_Y[6]	BLANKLEVEL_Y[5]	BLANKLEVEL_Y[4]	BLANKLEVEL_Y[3]	BLANKLEVEL_Y[2]	BLANKLEVEL_Y[1]	BLANKLEVEL_Y[0]	-	
DCPCR2_0		DCPMODE_C	-	-	-	-	BLANKLEVEL_CB[5]	BLANKLEVEL_CB[4]	BLANKLEVEL_CB[3]	BLANKLEVEL_CB[2]	
		BLANKLEVEL_CB[1]	BLANKLEVEL_CB[0]	BLANKLEVEL_CR[5]	BLANKLEVEL_CR[4]	BLANKLEVEL_CR[3]	BLANKLEVEL_CR[2]	BLANKLEVEL_CR[1]	BLANKLEVEL_CR[0]	-	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Digital video decoder	DCPCR3_0	-	DCPREPONSE [2]	DCPREPONSE [1]	DCPREPONSE [0]	-	-	-	-
		-	-	-	-	-	-	-	-
	DCPCR4_0	DCPSTART[5]	DCPSTART[4]	DCPSTART[3]	DCPSTART[2]	DCPSTART[1]	DCPSTART[0]	-	-
		-	-	-	-	-	-	-	-
	DCPCR5_0	DCPEND[5]	DCPEND[4]	DCPEND[3]	DCPEND[2]	DCPEND[1]	DCPEND[0]	-	-
		-	-	-	-	-	-	-	-
	DCPCR6_0	-	DCPWIDTH[6]	DCPWIDTH[5]	DCPWIDTH[4]	DCPWIDTH[3]	DCPWIDTH[2]	DCPWIDTH[1]	DCPWIDTH[0]
		-	-	-	-	-	-	-	-
	DCPCR7_0	DCPPOS_Y[7]	DCPPOS_Y[6]	DCPPOS_Y[5]	DCPPOS_Y[4]	DCPPOS_Y[3]	DCPPOS_Y[2]	DCPPOS_Y[1]	DCPPOS_Y[0]
		-	-	-	-	-	-	-	-
	DCPCR8_0	DCPPOS_C[7]	DCPPOS_C[6]	DCPPOS_C[5]	DCPPOS_C[4]	DCPPOS_C[3]	DCPPOS_C[2]	DCPPOS_C[1]	DCPPOS_C[0]
		-	-	-	-	-	-	-	-
	NSDCR_0	-	-	ACFINPUT[1]	ACFINPUT[0]	-	-	-	ACFLAGTIME[4]
		ACFLAGTIME[3]	ACFLAGTIME[2]	ACFLAGTIME[1]	ACFLAGTIME[0]	-	-	ACFFILTER[1]	ACFFILTER[0]
	BTLCR_0	LOCKRANGE[1]	LOCKRANGE[0]	LOOPGAIN[1]	LOOPGAIN[0]	LOCKLIMIT[1]	LOCKLIMIT[0]	BCOFREERUN	-
		DEFAULTSYS[1]	DEFAULTSYS[0]	NONTSC358	NONTSC443	NOPALM	NOPALN	NOPAL443	NOSECAM
	BTGPCR_0	BGPCHK	BGPWIDTH[6]	BGPWIDTH[5]	BGPWIDTH[4]	BGPWIDTH[3]	BGPWIDTH[2]	BGPWIDTH[1]	BGPWIDTH[0]
		BGPSTART[7]	BGPSTART[6]	BGPSTART[5]	BGPSTART[4]	BGPSTART[3]	BGPSTART[2]	BGPSTART[1]	BGPSTART[0]
	ACCCR1_0	KILLEROFFSET [3]	KILLEROFFSET [2]	KILLEROFFSET [1]	KILLEROFFSET [0]	ACCMODE	ACCMAXGAIN [1]	ACCMAXGAIN [0]	ACCLEVEL[8]
		ACCLEVEL[7]	ACCLEVEL[6]	ACCLEVEL[5]	ACCLEVEL[4]	ACCLEVEL[3]	ACCLEVEL[2]	ACCLEVEL[1]	ACCLEVEL[0]
	ACCCR2_0	-	-	-	-	-	CHROMASUB GAIN[1]	CHROMASUB GAIN[0]	CHROMAMAIN GAIN[8]
		CHROMAMAIN GAIN[7]	CHROMAMAIN GAIN[6]	CHROMAMAIN GAIN[5]	CHROMAMAIN GAIN[4]	CHROMAMAIN GAIN[3]	CHROMAMAIN GAIN[2]	CHROMAMAIN GAIN[1]	CHROMAMAIN GAIN[0]
	ACCCR3_0	ACCRESPOSE [1]	ACCRESPOSE [0]	ACCPRECS[5]	ACCPRECS[4]	ACCPRECS[3]	ACCPRECS[2]	ACCPRECS[1]	ACCPRECS[0]
		KILLERMODE	KILLERLEVEL[5]	KILLERLEVEL[4]	KILLERLEVEL[3]	KILLERLEVEL[2]	KILLERLEVEL[1]	KILLERLEVEL[0]	-
	TINTCR_0	TINTSUB[5]	TINTSUB[4]	TINTSUB[3]	TINTSUB[2]	TINTSUB[1]	TINTSUB[0]	TINTMAIN[9]	TINTMAIN[8]
		TINTMAIN[7]	TINTMAIN[6]	TINTMAIN[5]	TINTMAIN[4]	TINTMAIN[3]	TINTMAIN[2]	TINTMAIN[1]	TINTMAIN[0]
	YDCR_0	-	-	-	-	-	-	-	LUMADELAY[4]
		LUMADELAY[3]	LUMADELAY[2]	LUMADELAY[1]	LUMADELAY[0]	-	CHROMALPF	DEMOMODE[1]	DEMOMODE[0]
	AGCCR1_0	-	-	DOREDUCE	NOREDUCE	AGCRESPONSE [2]	AGCRESPONSE [1]	AGCRESPONSE [0]	AGCLEVEL[8]
		AGCLEVEL[7]	AGCLEVEL[6]	AGCLEVEL[5]	AGCLEVEL[4]	AGCLEVEL[3]	AGCLEVEL[2]	AGCLEVEL[1]	AGCLEVEL[0]
	AGCCR2_0	-	-	AGCPRECS[5]	AGCPRECS[4]	AGCPRECS[3]	AGCPRECS[2]	AGCPRECS[1]	AGCPRECS[0]
		-	-	-	-	-	-	-	-
	PKLIMITCR_0	PEAKLEVEL[1]	PEAKLEVEL[0]	PEAKATTACK[1]	PEAKATTACK[0]	PEAKRELEASE[1]	PEAKRELEASE[0]	PEAKRATIO[1]	PEAKRATIO[0]
		MAXPEAK SAMPLES[7]	MAXPEAK SAMPLES[6]	MAXPEAK SAMPLES[5]	MAXPEAK SAMPLES[4]	MAXPEAK SAMPLES[3]	MAXPEAK SAMPLES[2]	MAXPEAK SAMPLES[1]	MAXPEAK SAMPLES[0]
	RGORCR1_0	-	-	-	-	-	-	RADJ_O_LEVEL[9]	RADJ_O_LEVEL[8]
		RADJ_O_LEVEL[7]	RADJ_O_LEVEL[6]	RADJ_O_LEVEL[5]	RADJ_O_LEVEL[4]	RADJ_O_LEVEL[3]	RADJ_O_LEVEL[2]	RADJ_O_LEVEL[1]	RADJ_O_LEVEL[0]
	RGORCR2_0	-	-	-	-	-	-	RADJ_U_LEVEL[9]	RADJ_U_LEVEL[8]
		RADJ_U_LEVEL[7]	RADJ_U_LEVEL[6]	RADJ_U_LEVEL[5]	RADJ_U_LEVEL[4]	RADJ_U_LEVEL[3]	RADJ_U_LEVEL[2]	RADJ_U_LEVEL[1]	RADJ_U_LEVEL[0]
	RGORCR3_0	-	-	-	-	-	-	RADJ_O_LEVEL[9]	RADJ_O_LEVEL[8]
		RADJ_O_LEVEL[7]	RADJ_O_LEVEL[6]	RADJ_O_LEVEL[5]	RADJ_O_LEVEL[4]	RADJ_O_LEVEL[3]	RADJ_O_LEVEL[2]	RADJ_O_LEVEL[1]	RADJ_O_LEVEL[0]
	RGORCR4_0	-	-	-	-	-	-	RADJ_U_LEVEL[9]	RADJ_U_LEVEL[8]
		RADJ_U_LEVEL[7]	RADJ_U_LEVEL[6]	RADJ_U_LEVEL[5]	RADJ_U_LEVEL[4]	RADJ_U_LEVEL[3]	RADJ_U_LEVEL[2]	RADJ_U_LEVEL[1]	RADJ_U_LEVEL[0]
	RGORCR5_0	-	-	-	-	-	-	RADJ_O_LEVEL[9]	RADJ_O_LEVEL[8]
		RADJ_O_LEVEL[7]	RADJ_O_LEVEL[6]	RADJ_O_LEVEL[5]	RADJ_O_LEVEL[4]	RADJ_O_LEVEL[3]	RADJ_O_LEVEL[2]	RADJ_O_LEVEL[1]	RADJ_O_LEVEL[0]
	RGORCR6_0	-	-	-	-	-	-	RADJ_U_LEVEL[9]	RADJ_U_LEVEL[8]
		RADJ_U_LEVEL[7]	RADJ_U_LEVEL[6]	RADJ_U_LEVEL[5]	RADJ_U_LEVEL[4]	RADJ_U_LEVEL[3]	RADJ_U_LEVEL[2]	RADJ_U_LEVEL[1]	RADJ_U_LEVEL[0]
	RGORCR7_0	-	TEST_MONI[2]	TEST_MONI[1]	TEST_MONI[0]	RADJ_MIX_K_FIX[2]	RADJ_MIX_K_FIX[1]	RADJ_MIX_K_FIX[0]	-
		-	-	-	-	-	UCMP_SW	DCMP_SW	HWIDE_SW

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Digital video decoder	AFCPFCR_0	-	-	-	-	-	-	-	-
		-	-	-	PHDET_FIX	-	PHDET_DIV[2]	PHDET_DIV[1]	PHDET_DIV[0]
	RUPDCR_0	NEWSETTING	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	VSYNCSR_0	FHCOUNT_L	FHLOCK	ISNOISY	FHMODE	NOSIGNAL	FVLOCK	FVMODE	INTERLACED
		FVCOUNT[7]	FVCOUNT[6]	FVCOUNT[5]	FVCOUNT[4]	FVCOUNT[3]	FVCOUNT[2]	FVCOUNT[1]	FVCOUNT[0]
	HSYNCSR_0	FHCOUNT_H[16]	FHCOUNT_H[15]	FHCOUNT_H[14]	FHCOUNT_H[13]	FHCOUNT_H[12]	FHCOUNT_H[11]	FHCOUNT_H[10]	FHCOUNT_H[9]
		FHCOUNT_H[8]	FHCOUNT_H[7]	FHCOUNT_H[6]	FHCOUNT_H[5]	FHCOUNT_H[4]	FHCOUNT_H[3]	FHCOUNT_H[2]	FHCOUNT_H[1]
	DCPSR1_0	CLAMPLEVEL_CB[5]	CLAMPLEVEL_CB[4]	CLAMPLEVEL_CB[3]	CLAMPLEVEL_CB[2]	CLAMPLEVEL_CB[1]	CLAMPLEVEL_CB[0]	CLAMPLEVEL_Y[9]	CLAMPLEVEL_Y[8]
		CLAMPLEVEL_Y[7]	CLAMPLEVEL_Y[6]	CLAMPLEVEL_Y[5]	CLAMPLEVEL_Y[4]	CLAMPLEVEL_Y[3]	CLAMPLEVEL_Y[2]	CLAMPLEVEL_Y[1]	CLAMPLEVEL_Y[0]
	DCPSR2_0	CLAMPLEVEL_CR[5]	CLAMPLEVEL_CR[4]	CLAMPLEVEL_CR[3]	CLAMPLEVEL_CR[2]	CLAMPLEVEL_CR[1]	CLAMPLEVEL_CR[0]	-	-
		-	-	-	-	-	-	-	-
	NSDSR_0	ACFSTRENGTH[15]	ACFSTRENGTH[14]	ACFSTRENGTH[13]	ACFSTRENGTH[12]	ACFSTRENGTH[11]	ACFSTRENGTH[10]	ACFSTRENGTH[9]	ACFSTRENGTH[8]
		ACFSTRENGTH[7]	ACFSTRENGTH[6]	ACFSTRENGTH[5]	ACFSTRENGTH[4]	ACFSTRENGTH[3]	ACFSTRENGTH[2]	ACFSTRENGTH[1]	ACFSTRENGTH[0]
	CROMASR1_0	COLORSYS[1]	COLORSYS[0]	FSCMODE	FSCLOCK	NOBURST	ACCSUBGAIN[1]	ACCSUBGAIN[0]	ACCMaingain[8]
		ACCMaingain[7]	ACCMaingain[6]	ACCMaingain[5]	ACCMaingain[4]	ACCMaingain[3]	ACCMaingain[2]	ACCMaingain[1]	ACCMaingain[0]
	CROMASR2_0	-	-	-	ISSECAM	ISPAL	ISNTSC	-	-
		LOCKLEVEL[7]	LOCKLEVEL[6]	LOCKLEVEL[5]	LOCKLEVEL[4]	LOCKLEVEL[3]	LOCKLEVEL[2]	LOCKLEVEL[1]	LOCKLEVEL[0]
	SYNCSR_0	-	-	-	ISREDUCED	-	-	SYNCDPTH[9]	SYNCDPTH[8]
		SYNCDPTH[7]	SYNCDPTH[6]	SYNCDPTH[5]	SYNCDPTH[4]	SYNCDPTH[3]	SYNCDPTH[2]	SYNCDPTH[1]	SYNCDPTH[0]
	AGCCSR1_0	HIGHSAMPLES[7]	HIGHSAMPLES[6]	HIGHSAMPLES[5]	HIGHSAMPLES[4]	HIGHSAMPLES[3]	HIGHSAMPLES[2]	HIGHSAMPLES[1]	HIGHSAMPLES[0]
		PEAKSAMPLES[7]	PEAKSAMPLES[6]	PEAKSAMPLES[5]	PEAKSAMPLES[4]	PEAKSAMPLES[3]	PEAKSAMPLES[2]	PEAKSAMPLES[1]	PEAKSAMPLES[0]
	AGCCSR2_0	-	-	-	-	-	-	-	AGCCONVERGE
		AGCGAIN[7]	AGCGAIN[6]	AGCGAIN[5]	AGCGAIN[4]	AGCGAIN[3]	AGCGAIN[2]	AGCGAIN[1]	AGCGAIN[0]
	YCSR3_0	K15[3]	K15[2]	K15[1]	K15[0]	K13[5]	K13[4]	K13[3]	K13[2]
		K13[1]	K13[0]	K11[5]	K11[4]	K11[3]	K11[2]	K11[1]	K11[0]
	YCSR4_0	K16[3]	K16[2]	K16[1]	K16[0]	K14[5]	K14[4]	K14[3]	K14[2]
		K14[1]	K14[0]	K12[5]	K12[4]	K12[3]	K12[2]	K12[1]	K12[0]
	YCSR5_0	K22A[7]	K22A[6]	K22A[5]	K22A[4]	K22A[3]	K22A[2]	K22A[1]	K22A[0]
		-	-	K21A[5]	K21A[4]	K21A[3]	K21A[2]	K21A[1]	K21A[0]
	YCSR6_0	K22B[7]	K22B[6]	K22B[5]	K22B[4]	K22B[3]	K22B[2]	K22B[1]	K22B[0]
		-	-	K21B[5]	K21B[4]	K21B[3]	K21B[2]	K21B[1]	K21B[0]
	YCSR7_0	K23B[3]	K23B[2]	K23B[1]	K23B[0]	K23A[3]	K23A[2]	K23A[1]	K23A[0]
		-	-	-	K24[4]	K24[3]	K24[2]	K24[1]	K24[0]
	YCSR8_0	HVPF_NARROW	HVPF_NARROW	HVPF1_9TAP_ON	HVPF1_9TAP_ON	HFIL_TAP_SEL	-	-	-
		-	-	-	-	-	-	-	-
	YCSR9_0	DET2_ON	-	-	-	HSEL_MIX_Y[3]	HSEL_MIX_Y[2]	HSEL_MIX_Y[1]	HSEL_MIX_Y[0]
		VSEL_MIX_Y[3]	VSEL_MIX_Y[2]	VSEL_MIX_Y[1]	VSEL_MIX_Y[0]	HVSEL_MIX_Y[3]	HVSEL_MIX_Y[2]	HVSEL_MIX_Y[1]	HVSEL_MIX_Y[0]
	YCSR11_0	-	-	-	-	-	-	-	V_Y_LEVEL[8]
		V_Y_LEVEL[7]	V_Y_LEVEL[6]	V_Y_LEVEL[5]	V_Y_LEVEL[4]	V_Y_LEVEL[3]	V_Y_LEVEL[2]	V_Y_LEVEL[1]	V_Y_LEVEL[0]
	YCSR12_0	DET2_MIX_C[3]	DET2_MIX_C[2]	DET2_MIX_C[1]	DET2_MIX_C[0]	DET2_MIX_Y[3]	DET2_MIX_Y[2]	DET2_MIX_Y[1]	DET2_MIX_Y[0]
		-	-	-	-	FIL2_MODE_2D[1]	FIL2_MODE_2D[0]	-	FIL2_NARROW_2D
	DCPCR9_0	-	-	-	CLP_HOLD_ON_Y	CLP_HOLD_ON_CB	CLP_HOLD_ON_CR	-	-
		-	-	-	-	-	-	-	-
	YCTWA_F0_0	-	-	-	FIL2_2D_WA_F0[12]	FIL2_2D_WA_F0[11]	FIL2_2D_WA_F0[10]	FIL2_2D_WA_F0[9]	FIL2_2D_WA_F0[8]
		FIL2_2D_WA_F0[7]	FIL2_2D_WA_F0[6]	FIL2_2D_WA_F0[5]	FIL2_2D_WA_F0[4]	FIL2_2D_WA_F0[3]	FIL2_2D_WA_F0[2]	FIL2_2D_WA_F0[1]	FIL2_2D_WA_F0[0]
	YCTWA_F1_0	-	-	-	FIL2_2D_WA_F1[12]	FIL2_2D_WA_F1[11]	FIL2_2D_WA_F1[10]	FIL2_2D_WA_F1[9]	FIL2_2D_WA_F1[8]
		FIL2_2D_WA_F1[7]	FIL2_2D_WA_F1[6]	FIL2_2D_WA_F1[5]	FIL2_2D_WA_F1[4]	FIL2_2D_WA_F1[3]	FIL2_2D_WA_F1[2]	FIL2_2D_WA_F1[1]	FIL2_2D_WA_F1[0]
	YCTWA_F2_0	-	-	-	FIL2_2D_WA_F2[12]	FIL2_2D_WA_F2[11]	FIL2_2D_WA_F2[10]	FIL2_2D_WA_F2[9]	FIL2_2D_WA_F2[8]
		FIL2_2D_WA_F2[7]	FIL2_2D_WA_F2[6]	FIL2_2D_WA_F2[5]	FIL2_2D_WA_F2[4]	FIL2_2D_WA_F2[3]	FIL2_2D_WA_F2[2]	FIL2_2D_WA_F2[1]	FIL2_2D_WA_F2[0]



Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Digital video decoder	YCTWA_F3_0	-	-	-	FIL2_2D_WA_F3[12]	FIL2_2D_WA_F3[11]	FIL2_2D_WA_F3[10]	FIL2_2D_WA_F3[9]	FIL2_2D_WA_F3[8]
		FIL2_2D_WA_F3[7]	FIL2_2D_WA_F3[6]	FIL2_2D_WA_F3[5]	FIL2_2D_WA_F3[4]	FIL2_2D_WA_F3[3]	FIL2_2D_WA_F3[2]	FIL2_2D_WA_F3[1]	FIL2_2D_WA_F3[0]
	YCTWA_F4_0	-	-	-	FIL2_2D_WA_F4[12]	FIL2_2D_WA_F4[11]	FIL2_2D_WA_F4[10]	FIL2_2D_WA_F4[9]	FIL2_2D_WA_F4[8]
		FIL2_2D_WA_F4[7]	FIL2_2D_WA_F4[6]	FIL2_2D_WA_F4[5]	FIL2_2D_WA_F4[4]	FIL2_2D_WA_F4[3]	FIL2_2D_WA_F4[2]	FIL2_2D_WA_F4[1]	FIL2_2D_WA_F4[0]
	YCTWA_F5_0	-	-	-	FIL2_2D_WA_F5[12]	FIL2_2D_WA_F5[11]	FIL2_2D_WA_F5[10]	FIL2_2D_WA_F5[9]	FIL2_2D_WA_F5[8]
		FIL2_2D_WA_F5[7]	FIL2_2D_WA_F5[6]	FIL2_2D_WA_F5[5]	FIL2_2D_WA_F5[4]	FIL2_2D_WA_F5[3]	FIL2_2D_WA_F5[2]	FIL2_2D_WA_F5[1]	FIL2_2D_WA_F5[0]
	YCTWA_F6_0	-	-	-	FIL2_2D_WA_F6[12]	FIL2_2D_WA_F6[11]	FIL2_2D_WA_F6[10]	FIL2_2D_WA_F6[9]	FIL2_2D_WA_F6[8]
		FIL2_2D_WA_F6[7]	FIL2_2D_WA_F6[6]	FIL2_2D_WA_F6[5]	FIL2_2D_WA_F6[4]	FIL2_2D_WA_F6[3]	FIL2_2D_WA_F6[2]	FIL2_2D_WA_F6[1]	FIL2_2D_WA_F6[0]
	YCTWA_F7_0	-	-	-	FIL2_2D_WA_F7[12]	FIL2_2D_WA_F7[11]	FIL2_2D_WA_F7[10]	FIL2_2D_WA_F7[9]	FIL2_2D_WA_F7[8]
		FIL2_2D_WA_F7[7]	FIL2_2D_WA_F7[6]	FIL2_2D_WA_F7[5]	FIL2_2D_WA_F7[4]	FIL2_2D_WA_F7[3]	FIL2_2D_WA_F7[2]	FIL2_2D_WA_F7[1]	FIL2_2D_WA_F7[0]
	YCTWA_F8_0	-	-	-	FIL2_2D_WA_F8[12]	FIL2_2D_WA_F8[11]	FIL2_2D_WA_F8[10]	FIL2_2D_WA_F8[9]	FIL2_2D_WA_F8[8]
		FIL2_2D_WA_F8[7]	FIL2_2D_WA_F8[6]	FIL2_2D_WA_F8[5]	FIL2_2D_WA_F8[4]	FIL2_2D_WA_F8[3]	FIL2_2D_WA_F8[2]	FIL2_2D_WA_F8[1]	FIL2_2D_WA_F8[0]
	YCTWB_F0_0	-	-	-	FIL2_2D_WB_F0[12]	FIL2_2D_WB_F0[11]	FIL2_2D_WB_F0[10]	FIL2_2D_WB_F0[9]	FIL2_2D_WB_F0[8]
		FIL2_2D_WB_F0[7]	FIL2_2D_WB_F0[6]	FIL2_2D_WB_F0[5]	FIL2_2D_WB_F0[4]	FIL2_2D_WB_F0[3]	FIL2_2D_WB_F0[2]	FIL2_2D_WB_F0[1]	FIL2_2D_WB_F0[0]
	YCTWB_F1_0	-	-	-	FIL2_2D_WB_F1[12]	FIL2_2D_WB_F1[11]	FIL2_2D_WB_F1[10]	FIL2_2D_WB_F1[9]	FIL2_2D_WB_F1[8]
		FIL2_2D_WB_F1[7]	FIL2_2D_WB_F1[6]	FIL2_2D_WB_F1[5]	FIL2_2D_WB_F1[4]	FIL2_2D_WB_F1[3]	FIL2_2D_WB_F1[2]	FIL2_2D_WB_F1[1]	FIL2_2D_WB_F1[0]
	YCTWB_F2_0	-	-	-	FIL2_2D_WB_F2[12]	FIL2_2D_WB_F2[11]	FIL2_2D_WB_F2[10]	FIL2_2D_WB_F2[9]	FIL2_2D_WB_F2[8]
		FIL2_2D_WB_F2[7]	FIL2_2D_WB_F2[6]	FIL2_2D_WB_F2[5]	FIL2_2D_WB_F2[4]	FIL2_2D_WB_F2[3]	FIL2_2D_WB_F2[2]	FIL2_2D_WB_F2[1]	FIL2_2D_WB_F2[0]
	YCTWB_F3_0	-	-	-	FIL2_2D_WB_F3[12]	FIL2_2D_WB_F3[11]	FIL2_2D_WB_F3[10]	FIL2_2D_WB_F3[9]	FIL2_2D_WB_F3[8]
		FIL2_2D_WB_F3[7]	FIL2_2D_WB_F3[6]	FIL2_2D_WB_F3[5]	FIL2_2D_WB_F3[4]	FIL2_2D_WB_F3[3]	FIL2_2D_WB_F3[2]	FIL2_2D_WB_F3[1]	FIL2_2D_WB_F3[0]
	YCTWB_F4_0	-	-	-	FIL2_2D_WB_F4[12]	FIL2_2D_WB_F4[11]	FIL2_2D_WB_F4[10]	FIL2_2D_WB_F4[9]	FIL2_2D_WB_F4[8]
		FIL2_2D_WB_F4[7]	FIL2_2D_WB_F4[6]	FIL2_2D_WB_F4[5]	FIL2_2D_WB_F4[4]	FIL2_2D_WB_F4[3]	FIL2_2D_WB_F4[2]	FIL2_2D_WB_F4[1]	FIL2_2D_WB_F4[0]
	YCTWB_F5_0	-	-	-	FIL2_2D_WB_F5[12]	FIL2_2D_WB_F5[11]	FIL2_2D_WB_F5[10]	FIL2_2D_WB_F5[9]	FIL2_2D_WB_F5[8]
		FIL2_2D_WB_F5[7]	FIL2_2D_WB_F5[6]	FIL2_2D_WB_F5[5]	FIL2_2D_WB_F5[4]	FIL2_2D_WB_F5[3]	FIL2_2D_WB_F5[2]	FIL2_2D_WB_F5[1]	FIL2_2D_WB_F5[0]
	YCTWB_F6_0	-	-	-	FIL2_2D_WB_F6[12]	FIL2_2D_WB_F6[11]	FIL2_2D_WB_F6[10]	FIL2_2D_WB_F6[9]	FIL2_2D_WB_F6[8]
		FIL2_2D_WB_F6[7]	FIL2_2D_WB_F6[6]	FIL2_2D_WB_F6[5]	FIL2_2D_WB_F6[4]	FIL2_2D_WB_F6[3]	FIL2_2D_WB_F6[2]	FIL2_2D_WB_F6[1]	FIL2_2D_WB_F6[0]
	YCTWB_F7_0	-	-	-	FIL2_2D_WB_F7[12]	FIL2_2D_WB_F7[11]	FIL2_2D_WB_F7[10]	FIL2_2D_WB_F7[9]	FIL2_2D_WB_F7[8]
		FIL2_2D_WB_F7[7]	FIL2_2D_WB_F7[6]	FIL2_2D_WB_F7[5]	FIL2_2D_WB_F7[4]	FIL2_2D_WB_F7[3]	FIL2_2D_WB_F7[2]	FIL2_2D_WB_F7[1]	FIL2_2D_WB_F7[0]
	YCTWB_F8_0	-	-	-	FIL2_2D_WB_F8[12]	FIL2_2D_WB_F8[11]	FIL2_2D_WB_F8[10]	FIL2_2D_WB_F8[9]	FIL2_2D_WB_F8[8]
		FIL2_2D_WB_F8[7]	FIL2_2D_WB_F8[6]	FIL2_2D_WB_F8[5]	FIL2_2D_WB_F8[4]	FIL2_2D_WB_F8[3]	FIL2_2D_WB_F8[2]	FIL2_2D_WB_F8[1]	FIL2_2D_WB_F8[0]
	YCTNA_F0_0	-	-	-	FIL2_2D_NA_F0[12]	FIL2_2D_NA_F0[11]	FIL2_2D_NA_F0[10]	FIL2_2D_NA_F0[9]	FIL2_2D_NA_F0[8]
		FIL2_2D_NA_F0[7]	FIL2_2D_NA_F0[6]	FIL2_2D_NA_F0[5]	FIL2_2D_NA_F0[4]	FIL2_2D_NA_F0[3]	FIL2_2D_NA_F0[2]	FIL2_2D_NA_F0[1]	FIL2_2D_NA_F0[0]
	YCTNA_F1_0	-	-	-	FIL2_2D_NA_F1[12]	FIL2_2D_NA_F1[11]	FIL2_2D_NA_F1[10]	FIL2_2D_NA_F1[9]	FIL2_2D_NA_F1[8]
		FIL2_2D_NA_F1[7]	FIL2_2D_NA_F1[6]	FIL2_2D_NA_F1[5]	FIL2_2D_NA_F1[4]	FIL2_2D_NA_F1[3]	FIL2_2D_NA_F1[2]	FIL2_2D_NA_F1[1]	FIL2_2D_NA_F1[0]
	YCTNA_F2_0	-	-	-	FIL2_2D_NA_F2[12]	FIL2_2D_NA_F2[11]	FIL2_2D_NA_F2[10]	FIL2_2D_NA_F2[9]	FIL2_2D_NA_F2[8]
		FIL2_2D_NA_F2[7]	FIL2_2D_NA_F2[6]	FIL2_2D_NA_F2[5]	FIL2_2D_NA_F2[4]	FIL2_2D_NA_F2[3]	FIL2_2D_NA_F2[2]	FIL2_2D_NA_F2[1]	FIL2_2D_NA_F2[0]
	YCTNA_F3_0	-	-	-	FIL2_2D_NA_F3[12]	FIL2_2D_NA_F3[11]	FIL2_2D_NA_F3[10]	FIL2_2D_NA_F3[9]	FIL2_2D_NA_F3[8]
		FIL2_2D_NA_F3[7]	FIL2_2D_NA_F3[6]	FIL2_2D_NA_F3[5]	FIL2_2D_NA_F3[4]	FIL2_2D_NA_F3[3]	FIL2_2D_NA_F3[2]	FIL2_2D_NA_F3[1]	FIL2_2D_NA_F3[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Digital video decoder	YCTNA_F4_0	-	-	-	FIL2_2D_NA_F4[12]	FIL2_2D_NA_F4[11]	FIL2_2D_NA_F4[10]	FIL2_2D_NA_F4[9]	FIL2_2D_NA_F4[8]
		FIL2_2D_NA_F4[7]	FIL2_2D_NA_F4[6]	FIL2_2D_NA_F4[5]	FIL2_2D_NA_F4[4]	FIL2_2D_NA_F4[3]	FIL2_2D_NA_F4[2]	FIL2_2D_NA_F4[1]	FIL2_2D_NA_F4[0]
YCTNA_F5_0	YCTNA_F5_0	-	-	-	FIL2_2D_NA_F5[12]	FIL2_2D_NA_F5[11]	FIL2_2D_NA_F5[10]	FIL2_2D_NA_F5[9]	FIL2_2D_NA_F5[8]
		FIL2_2D_NA_F5[7]	FIL2_2D_NA_F5[6]	FIL2_2D_NA_F5[5]	FIL2_2D_NA_F5[4]	FIL2_2D_NA_F5[3]	FIL2_2D_NA_F5[2]	FIL2_2D_NA_F5[1]	FIL2_2D_NA_F5[0]
YCTNA_F6_0	YCTNA_F6_0	-	-	-	FIL2_2D_NA_F6[12]	FIL2_2D_NA_F6[11]	FIL2_2D_NA_F6[10]	FIL2_2D_NA_F6[9]	FIL2_2D_NA_F6[8]
		FIL2_2D_NA_F6[7]	FIL2_2D_NA_F6[6]	FIL2_2D_NA_F6[5]	FIL2_2D_NA_F6[4]	FIL2_2D_NA_F6[3]	FIL2_2D_NA_F6[2]	FIL2_2D_NA_F6[1]	FIL2_2D_NA_F6[0]
YCTNA_F7_0	YCTNA_F7_0	-	-	-	FIL2_2D_NA_F7[12]	FIL2_2D_NA_F7[11]	FIL2_2D_NA_F7[10]	FIL2_2D_NA_F7[9]	FIL2_2D_NA_F7[8]
		FIL2_2D_NA_F7[7]	FIL2_2D_NA_F7[6]	FIL2_2D_NA_F7[5]	FIL2_2D_NA_F7[4]	FIL2_2D_NA_F7[3]	FIL2_2D_NA_F7[2]	FIL2_2D_NA_F7[1]	FIL2_2D_NA_F7[0]
YCTNA_F8_0	YCTNA_F8_0	-	-	-	FIL2_2D_NA_F8[12]	FIL2_2D_NA_F8[11]	FIL2_2D_NA_F8[10]	FIL2_2D_NA_F8[9]	FIL2_2D_NA_F8[8]
		FIL2_2D_NA_F8[7]	FIL2_2D_NA_F8[6]	FIL2_2D_NA_F8[5]	FIL2_2D_NA_F8[4]	FIL2_2D_NA_F8[3]	FIL2_2D_NA_F8[2]	FIL2_2D_NA_F8[1]	FIL2_2D_NA_F8[0]
YCTNB_F0_0	YCTNB_F0_0	-	-	-	FIL2_2D_NB_F0[12]	FIL2_2D_NB_F0[11]	FIL2_2D_NB_F0[10]	FIL2_2D_NB_F0[9]	FIL2_2D_NB_F0[8]
		FIL2_2D_NB_F0[7]	FIL2_2D_NB_F0[6]	FIL2_2D_NB_F0[5]	FIL2_2D_NB_F0[4]	FIL2_2D_NB_F0[3]	FIL2_2D_NB_F0[2]	FIL2_2D_NB_F0[1]	FIL2_2D_NB_F0[0]
YCTNB_F1_0	YCTNB_F1_0	-	-	-	FIL2_2D_NB_F1[12]	FIL2_2D_NB_F1[11]	FIL2_2D_NB_F1[10]	FIL2_2D_NB_F1[9]	FIL2_2D_NB_F1[8]
		FIL2_2D_NB_F1[7]	FIL2_2D_NB_F1[6]	FIL2_2D_NB_F1[5]	FIL2_2D_NB_F1[4]	FIL2_2D_NB_F1[3]	FIL2_2D_NB_F1[2]	FIL2_2D_NB_F1[1]	FIL2_2D_NB_F1[0]
YCTNB_F2_0	YCTNB_F2_0	-	-	-	FIL2_2D_NB_F2[12]	FIL2_2D_NB_F2[11]	FIL2_2D_NB_F2[10]	FIL2_2D_NB_F2[9]	FIL2_2D_NB_F2[8]
		FIL2_2D_NB_F2[7]	FIL2_2D_NB_F2[6]	FIL2_2D_NB_F2[5]	FIL2_2D_NB_F2[4]	FIL2_2D_NB_F2[3]	FIL2_2D_NB_F2[2]	FIL2_2D_NB_F2[1]	FIL2_2D_NB_F2[0]
YCTNB_F3_0	YCTNB_F3_0	-	-	-	FIL2_2D_NB_F3[12]	FIL2_2D_NB_F3[11]	FIL2_2D_NB_F3[10]	FIL2_2D_NB_F3[9]	FIL2_2D_NB_F3[8]
		FIL2_2D_NB_F3[7]	FIL2_2D_NB_F3[6]	FIL2_2D_NB_F3[5]	FIL2_2D_NB_F3[4]	FIL2_2D_NB_F3[3]	FIL2_2D_NB_F3[2]	FIL2_2D_NB_F3[1]	FIL2_2D_NB_F3[0]
YCTNB_F4_0	YCTNB_F4_0	-	-	-	FIL2_2D_NB_F4[12]	FIL2_2D_NB_F4[11]	FIL2_2D_NB_F4[10]	FIL2_2D_NB_F4[9]	FIL2_2D_NB_F4[8]
		FIL2_2D_NB_F4[7]	FIL2_2D_NB_F4[6]	FIL2_2D_NB_F4[5]	FIL2_2D_NB_F4[4]	FIL2_2D_NB_F4[3]	FIL2_2D_NB_F4[2]	FIL2_2D_NB_F4[1]	FIL2_2D_NB_F4[0]
YCTNB_F5_0	YCTNB_F5_0	-	-	-	FIL2_2D_NB_F5[12]	FIL2_2D_NB_F5[11]	FIL2_2D_NB_F5[10]	FIL2_2D_NB_F5[9]	FIL2_2D_NB_F5[8]
		FIL2_2D_NB_F5[7]	FIL2_2D_NB_F5[6]	FIL2_2D_NB_F5[5]	FIL2_2D_NB_F5[4]	FIL2_2D_NB_F5[3]	FIL2_2D_NB_F5[2]	FIL2_2D_NB_F5[1]	FIL2_2D_NB_F5[0]
YCTNB_F6_0	YCTNB_F6_0	-	-	-	FIL2_2D_NB_F6[12]	FIL2_2D_NB_F6[11]	FIL2_2D_NB_F6[10]	FIL2_2D_NB_F6[9]	FIL2_2D_NB_F6[8]
		FIL2_2D_NB_F6[7]	FIL2_2D_NB_F6[6]	FIL2_2D_NB_F6[5]	FIL2_2D_NB_F6[4]	FIL2_2D_NB_F6[3]	FIL2_2D_NB_F6[2]	FIL2_2D_NB_F6[1]	FIL2_2D_NB_F6[0]
YCTNB_F7_0	YCTNB_F7_0	-	-	-	FIL2_2D_NB_F7[12]	FIL2_2D_NB_F7[11]	FIL2_2D_NB_F7[10]	FIL2_2D_NB_F7[9]	FIL2_2D_NB_F7[8]
		FIL2_2D_NB_F7[7]	FIL2_2D_NB_F7[6]	FIL2_2D_NB_F7[5]	FIL2_2D_NB_F7[4]	FIL2_2D_NB_F7[3]	FIL2_2D_NB_F7[2]	FIL2_2D_NB_F7[1]	FIL2_2D_NB_F7[0]
YCTNB_F8_0	YCTNB_F8_0	-	-	-	FIL2_2D_NB_F8[12]	FIL2_2D_NB_F8[11]	FIL2_2D_NB_F8[10]	FIL2_2D_NB_F8[9]	FIL2_2D_NB_F8[8]
		FIL2_2D_NB_F8[7]	FIL2_2D_NB_F8[6]	FIL2_2D_NB_F8[5]	FIL2_2D_NB_F8[4]	FIL2_2D_NB_F8[3]	FIL2_2D_NB_F8[2]	FIL2_2D_NB_F8[1]	FIL2_2D_NB_F8[0]
YGAINCR_0	YGAINCR_0	-	-	-	-	-	-	Y_GAIN2[9]	Y_GAIN2[8]
		Y_GAIN2[7]	Y_GAIN2[6]	Y_GAIN2[5]	Y_GAIN2[4]	Y_GAIN2[3]	Y_GAIN2[2]	Y_GAIN2[1]	Y_GAIN2[0]
CBGAINCR_0	CBGAINCR_0	-	-	-	-	-	-	CB_GAIN2[9]	CB_GAIN2[8]
		CB_GAIN2[7]	CB_GAIN2[6]	CB_GAIN2[5]	CB_GAIN2[4]	CB_GAIN2[3]	CB_GAIN2[2]	CB_GAIN2[1]	CB_GAIN2[0]
CRGAINCR_0	CRGAINCR_0	-	-	-	-	-	-	CR_GAIN2[9]	CR_GAIN2[8]
		CR_GAIN2[7]	CR_GAIN2[6]	CR_GAIN2[5]	CR_GAIN2[4]	CR_GAIN2[3]	CR_GAIN2[2]	CR_GAIN2[1]	CR_GAIN2[0]
PGA_UPDATE_0	PGA_UPDATE_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	PGA_VEN
PGACR_0	PGACR_0	-	PGA_GAIN_SEL	PGA_GAIN[5]	PGA_GAIN[4]	PGA_GAIN[3]	PGA_GAIN[2]	PGA_GAIN[1]	PGA_GAIN[0]
		-	-	-	-	-	-	-	-
ADCCR2_0	ADCCR2_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	ADC_VINSEL
ADCCR1_1	ADCCR1_1	-	-	-	-	-	-	-	AGCMODE
		-	-	-	-	-	-	-	-

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Digital video decoder	TGCR1_1	-	-	-	-	-	-	-	SRCLEFT[8]	
		SRCLEFT[7]	SRCLEFT[6]	SRCLEFT[5]	SRCLEFT[4]	SRCLEFT[3]	SRCLEFT[2]	SRCLEFT[1]	SRCLEFT[0]	
	TGCR2_1	SRCRTOPTOP[5]	SRCRTOPTOP[4]	SRCRTOPTOP[3]	SRCRTOPTOP[2]	SRCRTOPTOP[1]	SRCRTOPTOP[0]	SRCRHEIGHT[9]	SRCRHEIGHT[8]	
		SRCRHEIGHT[7]	SRCRHEIGHT[6]	SRCRHEIGHT[5]	SRCRHEIGHT[4]	SRCRHEIGHT[3]	SRCRHEIGHT[2]	SRCRHEIGHT[1]	SRCRHEIGHT[0]	
	TGCR3_1	-	-	-	-	-	-	SRCRWIDTH[10]	SRCRWIDTH[9]	SRCRWIDTH[8]
		SRCRWIDTH[7]	SRCRWIDTH[6]	SRCRWIDTH[5]	SRCRWIDTH[4]	SRCRWIDTH[3]	SRCRWIDTH[2]	SRCRWIDTH[1]	SRCRWIDTH[0]	
	SYNSCR1_1	LPFVSYNC[2]	LPFVSYNC[1]	LPFVSYNC[0]	LPFHSYNC[2]	LPFHSYNC[1]	LPFHSYNC[0]	-	-	
		VELOCITYSHIFT_H[3]	VELOCITYSHIFT_H[2]	VELOCITYSHIFT_H[1]	VELOCITYSHIFT_H[0]	SLICERMODE_H[1]	SLICERMODE_H[0]	SLICERMODE_V[1]	SLICERMODE_V[0]	
	SYNSCR2_1	-	-	-	-	SYNCRMAXDUTY_H[5]	SYNCRMAXDUTY_H[4]	SYNCRMAXDUTY_H[3]	SYNCRMAXDUTY_H[2]	
		SYNCRMAXDUTY_H[1]	SYNCRMAXDUTY_H[0]	SYNCRMINDUTY_H[5]	SYNCRMINDUTY_H[4]	SYNCRMINDUTY_H[3]	SYNCRMINDUTY_H[2]	SYNCRMINDUTY_H[1]	SYNCRMINDUTY_H[0]	
	SYNSCR3_1	-	-	SSCLIPSEL[3]	SSCLIPSEL[2]	SSCLIPSEL[1]	SSCLIPSEL[0]	CSYNCSLICE_H[9]	CSYNCSLICE_H[8]	
		CSYNCSLICE_H[7]	CSYNCSLICE_H[6]	CSYNCSLICE_H[5]	CSYNCSLICE_H[4]	CSYNCSLICE_H[3]	CSYNCSLICE_H[2]	CSYNCSLICE_H[1]	CSYNCSLICE_H[0]	
	SYNSCR4_1	-	-	-	-	SYNCRMAXDUTY_V[5]	SYNCRMAXDUTY_V[4]	SYNCRMAXDUTY_V[3]	SYNCRMAXDUTY_V[2]	
		SYNCRMAXDUTY_V[1]	SYNCRMAXDUTY_V[0]	SYNCRMINDUTY_V[5]	SYNCRMINDUTY_V[4]	SYNCRMINDUTY_V[3]	SYNCRMINDUTY_V[2]	SYNCRMINDUTY_V[1]	SYNCRMINDUTY_V[0]	
	SYNSCR5_1	VSYNCRDELAY	VSYNCRSLICE[4]	VSYNCRSLICE[3]	VSYNCRSLICE[2]	VSYNCRSLICE[1]	VSYNCRSLICE[0]	CSYNCSLICE_V[9]	CSYNCSLICE_V[8]	
		CSYNCSLICE_V[7]	CSYNCSLICE_V[6]	CSYNCSLICE_V[5]	CSYNCSLICE_V[4]	CSYNCSLICE_V[3]	CSYNCSLICE_V[2]	CSYNCSLICE_V[1]	CSYNCSLICE_V[0]	
	HAFCCR1_1	HAFCRGAIN[3]	HAFCRGAIN[2]	HAFCRGAIN[1]	HAFCRGAIN[0]	-	HAFCRFREERUN	HAFCRCTYP[9]	HAFCRCTYP[8]	
		HAFCRCTYP[7]	HAFCRCTYP[6]	HAFCRCTYP[5]	HAFCRCTYP[4]	HAFCRCTYP[3]	HAFCRCTYP[2]	HAFCRCTYP[1]	HAFCRCTYP[0]	
	HAFCCR2_1	HAFCRSTART[3]	HAFCRSTART[2]	HAFCRSTART[1]	HAFCRSTART[0]	NOXZHOSC	DOX2HOSC	HAFCRMAX[9]	HAFCRMAX[8]	
		HAFCRMAX[7]	HAFCRMAX[6]	HAFCRMAX[5]	HAFCRMAX[4]	HAFCRMAX[3]	HAFCRMAX[2]	HAFCRMAX[1]	HAFCRMAX[0]	
	HAFCCR3_1	HAFCREND[3]	HAFCREND[2]	HAFCREND[1]	HAFCREND[0]	HAFCRMODE[1]	HAFCRMODE[0]	HAFCRMIN[9]	HAFCRMIN[8]	
		HAFCRMIN[7]	HAFCRMIN[6]	HAFCRMIN[5]	HAFCRMIN[4]	HAFCRMIN[3]	HAFCRMIN[2]	HAFCRMIN[1]	HAFCRMIN[0]	
	VCDWCR1_1	VCDFREERUN	NOVCD50	NOVCD60	VCDDEFAULT[1]	VCDDEFAULT[0]	VCDWINDOW[5]	VCDWINDOW[4]	VCDWINDOW[3]	
		VCDWINDOW[2]	VCDWINDOW[1]	VCDWINDOW[0]	VCDOFFSET[4]	VCDOFFSET[3]	VCDOFFSET[2]	VCDOFFSET[1]	VCDOFFSET[0]	
	DCPCR1_1	DCPCRMODE_Y	-	-	-	-	DCPCRCHECK	-	BLANKLEVEL_Y[9]	BLANKLEVEL_Y[8]
		BLANKLEVEL_Y[7]	BLANKLEVEL_Y[6]	BLANKLEVEL_Y[5]	BLANKLEVEL_Y[4]	BLANKLEVEL_Y[3]	BLANKLEVEL_Y[2]	BLANKLEVEL_Y[1]	BLANKLEVEL_Y[0]	
	DCPCR2_1	DCPCRMODE_C	-	-	-	-	BLANKLEVEL_CB[5]	BLANKLEVEL_CB[4]	BLANKLEVEL_CB[3]	BLANKLEVEL_CB[2]
		BLANKLEVEL_CB[1]	BLANKLEVEL_CB[0]	BLANKLEVEL_CR[5]	BLANKLEVEL_CR[4]	BLANKLEVEL_CR[3]	BLANKLEVEL_CR[2]	BLANKLEVEL_CR[1]	BLANKLEVEL_CR[0]	
	DCPCR3_1	-	DCPCRRESPONSE[2]	DCPCRRESPONSE[1]	DCPCRRESPONSE[0]	-	-	-	-	
		-	-	-	-	-	-	-	-	
	DCPCR4_1	DCPCRSTART[5]	DCPCRSTART[4]	DCPCRSTART[3]	DCPCRSTART[2]	DCPCRSTART[1]	DCPCRSTART[0]	-	-	
		-	-	-	-	-	-	-	-	
	DCPCR5_1	DCPCRPEND[5]	DCPCRPEND[4]	DCPCRPEND[3]	DCPCRPEND[2]	DCPCRPEND[1]	DCPCRPEND[0]	-	-	
		-	-	-	-	-	-	-	-	
	DCPCR6_1	-	DCPCRWIDTH[6]	DCPCRWIDTH[5]	DCPCRWIDTH[4]	DCPCRWIDTH[3]	DCPCRWIDTH[2]	DCPCRWIDTH[1]	DCPCRWIDTH[0]	
		-	-	-	-	-	-	-	-	
	DCPCR7_1	DCPCRPOS_Y[7]	DCPCRPOS_Y[6]	DCPCRPOS_Y[5]	DCPCRPOS_Y[4]	DCPCRPOS_Y[3]	DCPCRPOS_Y[2]	DCPCRPOS_Y[1]	DCPCRPOS_Y[0]	
		-	-	-	-	-	-	-	-	
	DCPCR8_1	DCPCRPOS_C[7]	DCPCRPOS_C[6]	DCPCRPOS_C[5]	DCPCRPOS_C[4]	DCPCRPOS_C[3]	DCPCRPOS_C[2]	DCPCRPOS_C[1]	DCPCRPOS_C[0]	
		-	-	-	-	-	-	-	-	
	NSDCR_1	-	-	ACFINPUT[1]	ACFINPUT[0]	-	-	-	ACFLAGTIME[4]	
		ACFLAGTIME[3]	ACFLAGTIME[2]	ACFLAGTIME[1]	ACFLAGTIME[0]	-	-	ACFFILTER[1]	ACFFILTER[0]	
	BTLCR_1	LOCKRANGE[1]	LOCKRANGE[0]	LOOPGAIN[1]	LOOPGAIN[0]	LOCKLIMIT[1]	LOCKLIMIT[0]	BCOFFREERUN	-	
		DEFAULTSYS[1]	DEFAULTSYS[0]	NONTSC358	NONTSC443	NOPALM	NOPALN	NOPAL443	NOSECAM	
	BTGPCR_1	BGPCCHECK	BGPWIDTH[6]	BGPWIDTH[5]	BGPWIDTH[4]	BGPWIDTH[3]	BGPWIDTH[2]	BGPWIDTH[1]	BGPWIDTH[0]	
		BGPSTART[7]	BGPSTART[6]	BGPSTART[5]	BGPSTART[4]	BGPSTART[3]	BGPSTART[2]	BGPSTART[1]	BGPSTART[0]	
	ACCCR1_1	KILLEROFFSET[3]	KILLEROFFSET[2]	KILLEROFFSET[1]	KILLEROFFSET[0]	ACCMODE	ACCMAXGAIN[1]	ACCMAXGAIN[0]	ACCLEVEL[8]	
		ACCLEVEL[7]	ACCLEVEL[6]	ACCLEVEL[5]	ACCLEVEL[4]	ACCLEVEL[3]	ACCLEVEL[2]	ACCLEVEL[1]	ACCLEVEL[0]	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Digital video decoder	ACCCR2_1	-	-	-	-	-	CHROMASUB GAIN[1]	CHROMASUB GAIN[0]	CHROMAMAIN GAIN[8]
		CHROMAMAIN GAIN[7]	CHROMAMAIN GAIN[6]	CHROMAMAIN GAIN[5]	CHROMAMAIN GAIN[4]	CHROMAMAIN GAIN[3]	CHROMAMAIN GAIN[2]	CHROMAMAIN GAIN[1]	CHROMAMAIN GAIN[0]
	ACCCR3_1	ACCRESPONSE [1]	ACCRESPONSE [0]	ACCPRECIS[5]	ACCPRECIS[4]	ACCPRECIS[3]	ACCPRECIS[2]	ACCPRECIS[1]	ACCPRECIS[0]
		KILLERMODE	KILLERLEVEL[5]	KILLERLEVEL[4]	KILLERLEVEL[3]	KILLERLEVEL[2]	KILLERLEVEL[1]	KILLERLEVEL[0]	-
	TINTCR_1	TINTSUB[5]	TINTSUB[4]	TINTSUB[3]	TINTSUB[2]	TINTSUB[1]	TINTSUB[0]	TINTMAIN[9]	TINTMAIN[8]
		TINTMAIN[7]	TINTMAIN[6]	TINTMAIN[5]	TINTMAIN[4]	TINTMAIN[3]	TINTMAIN[2]	TINTMAIN[1]	TINTMAIN[0]
	YCDCR_1	-	-	-	-	-	-	-	LUMADELAY[4]
		LUMADELAY[3]	LUMADELAY[2]	LUMADELAY[1]	LUMADELAY[0]	-	CHROMALPF	DEMOMODE[1]	DEMOMODE[0]
	AGCCR1_1	-	-	DOREDUCE	NOREDUCE	AGCRESPONSE [2]	AGCRESPONSE [1]	AGCRESPONSE [0]	AGCLEVEL[8]
		AGCLEVEL[7]	AGCLEVEL[6]	AGCLEVEL[5]	AGCLEVEL[4]	AGCLEVEL[3]	AGCLEVEL[2]	AGCLEVEL[1]	AGCLEVEL[0]
	AGCCR2_1	-	-	AGCPRECIS[5]	AGCPRECIS[4]	AGCPRECIS[3]	AGCPRECIS[2]	AGCPRECIS[1]	AGCPRECIS[0]
		-	-	-	-	-	-	-	-
	PKLIMITCR_1	PEAKLEVEL[1]	PEAKLEVEL[0]	PEAKATTACK[1]	PEAKATTACK[0]	PEAKRELEASE[1]	PEAKRELEASE[0]	PEAKRATIO[1]	PEAKRATIO[0]
		MAXPEAK SAMPLES[7]	MAXPEAK SAMPLES[6]	MAXPEAK SAMPLES[5]	MAXPEAK SAMPLES[4]	MAXPEAK SAMPLES[3]	MAXPEAK SAMPLES[2]	MAXPEAK SAMPLES[1]	MAXPEAK SAMPLES[0]
	RGORCR1_1	-	-	-	-	-	-	RADJ_O_LEVEL[9]	RADJ_O_LEVEL[8]
		RADJ_O_LEVEL[7]	RADJ_O_LEVEL[6]	RADJ_O_LEVEL[5]	RADJ_O_LEVEL[4]	RADJ_O_LEVEL[3]	RADJ_O_LEVEL[2]	RADJ_O_LEVEL[1]	RADJ_O_LEVEL[0]
	RGORCR2_1	-	-	-	-	-	-	RADJ_U_LEVEL[9]	RADJ_U_LEVEL[8]
		RADJ_U_LEVEL[7]	RADJ_U_LEVEL[6]	RADJ_U_LEVEL[5]	RADJ_U_LEVEL[4]	RADJ_U_LEVEL[3]	RADJ_U_LEVEL[2]	RADJ_U_LEVEL[1]	RADJ_U_LEVEL[0]
	RGORCR3_1	-	-	-	-	-	-	RADJ_O_LEVEL[9]	RADJ_O_LEVEL[8]
		RADJ_O_LEVEL[7]	RADJ_O_LEVEL[6]	RADJ_O_LEVEL[5]	RADJ_O_LEVEL[4]	RADJ_O_LEVEL[3]	RADJ_O_LEVEL[2]	RADJ_O_LEVEL[1]	RADJ_O_LEVEL[0]
	RGORCR4_1	-	-	-	-	-	-	RADJ_U_LEVEL[9]	RADJ_U_LEVEL[8]
		RADJ_U_LEVEL[7]	RADJ_U_LEVEL[6]	RADJ_U_LEVEL[5]	RADJ_U_LEVEL[4]	RADJ_U_LEVEL[3]	RADJ_U_LEVEL[2]	RADJ_U_LEVEL[1]	RADJ_U_LEVEL[0]
	RGORCR5_1	-	-	-	-	-	-	RADJ_O_LEVEL[9]	RADJ_O_LEVEL[8]
		RADJ_O_LEVEL[7]	RADJ_O_LEVEL[6]	RADJ_O_LEVEL[5]	RADJ_O_LEVEL[4]	RADJ_O_LEVEL[3]	RADJ_O_LEVEL[2]	RADJ_O_LEVEL[1]	RADJ_O_LEVEL[0]
	RGORCR6_1	-	-	-	-	-	-	RADJ_U_LEVEL[9]	RADJ_U_LEVEL[8]
		RADJ_U_LEVEL[7]	RADJ_U_LEVEL[6]	RADJ_U_LEVEL[5]	RADJ_U_LEVEL[4]	RADJ_U_LEVEL[3]	RADJ_U_LEVEL[2]	RADJ_U_LEVEL[1]	RADJ_U_LEVEL[0]
	RGORCR7_1	-	TEST_MONI[2]	TEST_MONI[1]	TEST_MONI[0]	RADJ_MIX_K_FIX[2]	RADJ_MIX_K_FIX[1]	RADJ_MIX_K_FIX[0]	-
		-	-	-	-	-	UCMP_SW	DCMP_SW	HWISE_SW
	AFPCFCR_1	-	-	-	-	-	-	-	-
		-	-	-	PHDET_FIX	-	PHDET_DIV[2]	PHDET_DIV[1]	PHDET_DIV[0]
	RUPDCR_1	NEWSETTING	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	VSYNCSR_1	FHCOUNT_L	FHLOCK	ISNOISY	FHMODE	NOSIGNAL	FVLOCK	FVMODE	INTERLACED
		FVCOUNT[7]	FVCOUNT[6]	FVCOUNT[5]	FVCOUNT[4]	FVCOUNT[3]	FVCOUNT[2]	FVCOUNT[1]	FVCOUNT[0]
	HSYNSR_1	FHCOUNT_H[16]	FHCOUNT_H[15]	FHCOUNT_H[14]	FHCOUNT_H[13]	FHCOUNT_H[12]	FHCOUNT_H[11]	FHCOUNT_H[10]	FHCOUNT_H[9]
		FHCOUNT_H[8]	FHCOUNT_H[7]	FHCOUNT_H[6]	FHCOUNT_H[5]	FHCOUNT_H[4]	FHCOUNT_H[3]	FHCOUNT_H[2]	FHCOUNT_H[1]
	DCPSR1_1	CLAMPLEVEL_CB[5]	CLAMPLEVEL_CB[4]	CLAMPLEVEL_CB[3]	CLAMPLEVEL_CB[2]	CLAMPLEVEL_CB[1]	CLAMPLEVEL_CB[0]	CLAMPLEVEL_Y[9]	CLAMPLEVEL_Y[8]
		CLAMPLEVEL_Y[7]	CLAMPLEVEL_Y[6]	CLAMPLEVEL_Y[5]	CLAMPLEVEL_Y[4]	CLAMPLEVEL_Y[3]	CLAMPLEVEL_Y[2]	CLAMPLEVEL_Y[1]	CLAMPLEVEL_Y[0]
	DCPSR2_1	CLAMPLEVEL_CR[5]	CLAMPLEVEL_CR[4]	CLAMPLEVEL_CR[3]	CLAMPLEVEL_CR[2]	CLAMPLEVEL_CR[1]	CLAMPLEVEL_CR[0]	-	-
		-	-	-	-	-	-	-	-
	NSDSR_1	ACFSTRENGTH [15]	ACFSTRENGTH [14]	ACFSTRENGTH [13]	ACFSTRENGTH [12]	ACFSTRENGTH [11]	ACFSTRENGTH [10]	ACFSTRENGTH [9]	ACFSTRENGTH [8]
		ACFSTRENGTH [7]	ACFSTRENGTH [6]	ACFSTRENGTH [5]	ACFSTRENGTH [4]	ACFSTRENGTH [3]	ACFSTRENGTH [2]	ACFSTRENGTH [1]	ACFSTRENGTH [0]
	CROMASR1_1	COLORSYS[1]	COLORSYS[0]	FSCMODE	FSCLOCK	NOBURST	ACCSUBGAIN[1]	ACCSUBGAIN[0]	ACCMINGAIN[8]
		ACCMINGAIN[7]	ACCMINGAIN[6]	ACCMINGAIN[5]	ACCMINGAIN[4]	ACCMINGAIN[3]	ACCMINGAIN[2]	ACCMINGAIN[1]	ACCMINGAIN[0]
	CROMASR2_1	-	-	-	ISSECAM	ISPAL	ISNTSC	-	-
		LOCKLEVEL[7]	LOCKLEVEL[6]	LOCKLEVEL[5]	LOCKLEVEL[4]	LOCKLEVEL[3]	LOCKLEVEL[2]	LOCKLEVEL[1]	LOCKLEVEL[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Digital video decoder	SYNCSR_1	-	-	-	ISREDUCED	-	-	SYNCDPTH[9]	SYNCDPTH[0]
		SYNCDPTH[7]	SYNCDPTH[6]	SYNCDPTH[5]	SYNCDPTH[4]	SYNCDPTH[3]	SYNCDPTH[2]	SYNCDPTH[1]	SYNCDPTH[0]
	AGCCSR_1	HIGHSAMPLES[7]	HIGHSAMPLES[6]	HIGHSAMPLES[5]	HIGHSAMPLES[4]	HIGHSAMPLES[3]	HIGHSAMPLES[2]	HIGHSAMPLES[1]	HIGHSAMPLES[0]
		PEAKSAMPLES[7]	PEAKSAMPLES[6]	PEAKSAMPLES[5]	PEAKSAMPLES[4]	PEAKSAMPLES[3]	PEAKSAMPLES[2]	PEAKSAMPLES[1]	PEAKSAMPLES[0]
	AGCCSR2_1	-	-	-	-	-	-	-	AGCCONVERGE
		AGCGAIN[7]	AGCGAIN[6]	AGCGAIN[5]	AGCGAIN[4]	AGCGAIN[3]	AGCGAIN[2]	AGCGAIN[1]	AGCGAIN[0]
	YCSR3_1	K15[3]	K15[2]	K15[1]	K15[0]	K13[5]	K13[4]	K13[3]	K13[2]
		K13[1]	K13[0]	K11[5]	K11[4]	K11[3]	K11[2]	K11[1]	K11[0]
	YCSR4_1	K16[3]	K16[2]	K16[1]	K16[0]	K14[5]	K14[4]	K14[3]	K14[2]
		K14[1]	K14[0]	K12[5]	K12[4]	K12[3]	K12[2]	K12[1]	K12[0]
YCSR5_1	K22A[7]	K22A[6]	K22A[5]	K22A[4]	K22A[3]	K22A[2]	K22A[1]	K22A[0]	
	-	-	K21A[5]	K21A[4]	K21A[3]	K21A[2]	K21A[1]	K21A[0]	
YCSR6_1	K22B[7]	K22B[6]	K22B[5]	K22B[4]	K22B[3]	K22B[2]	K22B[1]	K22B[0]	
	-	-	K21B[5]	K21B[4]	K21B[3]	K21B[2]	K21B[1]	K21B[0]	
YCSR7_1	K23B[3]	K23B[2]	K23B[1]	K23B[0]	K23A[3]	K23A[2]	K23A[1]	K23A[0]	
	-	-	-	K24[4]	K24[3]	K24[2]	K24[1]	K24[0]	
YCSR8_1	HBPF_NARROW	HVBPF_NARROW	HBPF1_9TAP_ON	HVBPF1_9TAP_ON	HFIL_TAP_SEL	-	-	-	
	-	-	-	-	-	-	-	-	
YCSR9_1	DET2_ON	-	-	-	HSEL_MIX_Y[3]	HSEL_MIX_Y[2]	HSEL_MIX_Y[1]	HSEL_MIX_Y[0]	
	VSEL_MIX_Y[3]	VSEL_MIX_Y[2]	VSEL_MIX_Y[1]	VSEL_MIX_Y[0]	HVSEL_MIX_Y[3]	HVSEL_MIX_Y[2]	HVSEL_MIX_Y[1]	HVSEL_MIX_Y[0]	
YCSR11_1	-	-	-	-	-	-	-	V_Y_LEVEL[8]	
	V_Y_LEVEL[7]	V_Y_LEVEL[6]	V_Y_LEVEL[5]	V_Y_LEVEL[4]	V_Y_LEVEL[3]	V_Y_LEVEL[2]	V_Y_LEVEL[1]	V_Y_LEVEL[0]	
YCSR12_1	DET2_MIX_C[3]	DET2_MIX_C[2]	DET2_MIX_C[1]	DET2_MIX_C[0]	DET2_MIX_Y[3]	DET2_MIX_Y[2]	DET2_MIX_Y[1]	DET2_MIX_Y[0]	
	-	-	-	-	FIL2_MODE_2D[1]	FIL2_MODE_2D[0]	-	FIL2_NARROW_2D	
DCPCR9_1	-	-	-	CLP_HOLD_ON_Y	CLP_HOLD_ON_CB	CLP_HOLD_ON_CR	-	-	
	-	-	-	-	-	-	-	-	
YCTWA_F0_1	-	-	-	FIL2_2D_WA_F0[12]	FIL2_2D_WA_F0[11]	FIL2_2D_WA_F0[10]	FIL2_2D_WA_F0[9]	FIL2_2D_WA_F0[8]	
	FIL2_2D_WA_F0[7]	FIL2_2D_WA_F0[6]	FIL2_2D_WA_F0[5]	FIL2_2D_WA_F0[4]	FIL2_2D_WA_F0[3]	FIL2_2D_WA_F0[2]	FIL2_2D_WA_F0[1]	FIL2_2D_WA_F0[0]	
YCTWA_F1_1	-	-	-	FIL2_2D_WA_F1[12]	FIL2_2D_WA_F1[11]	FIL2_2D_WA_F1[10]	FIL2_2D_WA_F1[9]	FIL2_2D_WA_F1[8]	
	FIL2_2D_WA_F1[7]	FIL2_2D_WA_F1[6]	FIL2_2D_WA_F1[5]	FIL2_2D_WA_F1[4]	FIL2_2D_WA_F1[3]	FIL2_2D_WA_F1[2]	FIL2_2D_WA_F1[1]	FIL2_2D_WA_F1[0]	
YCTWA_F2_1	-	-	-	FIL2_2D_WA_F2[12]	FIL2_2D_WA_F2[11]	FIL2_2D_WA_F2[10]	FIL2_2D_WA_F2[9]	FIL2_2D_WA_F2[8]	
	FIL2_2D_WA_F2[7]	FIL2_2D_WA_F2[6]	FIL2_2D_WA_F2[5]	FIL2_2D_WA_F2[4]	FIL2_2D_WA_F2[3]	FIL2_2D_WA_F2[2]	FIL2_2D_WA_F2[1]	FIL2_2D_WA_F2[0]	
YCTWA_F3_1	-	-	-	FIL2_2D_WA_F3[12]	FIL2_2D_WA_F3[11]	FIL2_2D_WA_F3[10]	FIL2_2D_WA_F3[9]	FIL2_2D_WA_F3[8]	
	FIL2_2D_WA_F3[7]	FIL2_2D_WA_F3[6]	FIL2_2D_WA_F3[5]	FIL2_2D_WA_F3[4]	FIL2_2D_WA_F3[3]	FIL2_2D_WA_F3[2]	FIL2_2D_WA_F3[1]	FIL2_2D_WA_F3[0]	
YCTWA_F4_1	-	-	-	FIL2_2D_WA_F4[12]	FIL2_2D_WA_F4[11]	FIL2_2D_WA_F4[10]	FIL2_2D_WA_F4[9]	FIL2_2D_WA_F4[8]	
	FIL2_2D_WA_F4[7]	FIL2_2D_WA_F4[6]	FIL2_2D_WA_F4[5]	FIL2_2D_WA_F4[4]	FIL2_2D_WA_F4[3]	FIL2_2D_WA_F4[2]	FIL2_2D_WA_F4[1]	FIL2_2D_WA_F4[0]	
YCTWA_F5_1	-	-	-	FIL2_2D_WA_F5[12]	FIL2_2D_WA_F5[11]	FIL2_2D_WA_F5[10]	FIL2_2D_WA_F5[9]	FIL2_2D_WA_F5[8]	
	FIL2_2D_WA_F5[7]	FIL2_2D_WA_F5[6]	FIL2_2D_WA_F5[5]	FIL2_2D_WA_F5[4]	FIL2_2D_WA_F5[3]	FIL2_2D_WA_F5[2]	FIL2_2D_WA_F5[1]	FIL2_2D_WA_F5[0]	
YCTWA_F6_1	-	-	-	FIL2_2D_WA_F6[12]	FIL2_2D_WA_F6[11]	FIL2_2D_WA_F6[10]	FIL2_2D_WA_F6[9]	FIL2_2D_WA_F6[8]	
	FIL2_2D_WA_F6[7]	FIL2_2D_WA_F6[6]	FIL2_2D_WA_F6[5]	FIL2_2D_WA_F6[4]	FIL2_2D_WA_F6[3]	FIL2_2D_WA_F6[2]	FIL2_2D_WA_F6[1]	FIL2_2D_WA_F6[0]	
YCTWA_F7_1	-	-	-	FIL2_2D_WA_F7[12]	FIL2_2D_WA_F7[11]	FIL2_2D_WA_F7[10]	FIL2_2D_WA_F7[9]	FIL2_2D_WA_F7[8]	
	FIL2_2D_WA_F7[7]	FIL2_2D_WA_F7[6]	FIL2_2D_WA_F7[5]	FIL2_2D_WA_F7[4]	FIL2_2D_WA_F7[3]	FIL2_2D_WA_F7[2]	FIL2_2D_WA_F7[1]	FIL2_2D_WA_F7[0]	
YCTWA_F8_1	-	-	-	FIL2_2D_WA_F8[12]	FIL2_2D_WA_F8[11]	FIL2_2D_WA_F8[10]	FIL2_2D_WA_F8[9]	FIL2_2D_WA_F8[8]	
	FIL2_2D_WA_F8[7]	FIL2_2D_WA_F8[6]	FIL2_2D_WA_F8[5]	FIL2_2D_WA_F8[4]	FIL2_2D_WA_F8[3]	FIL2_2D_WA_F8[2]	FIL2_2D_WA_F8[1]	FIL2_2D_WA_F8[0]	
YCTWB_F0_1	-	-	-	FIL2_2D_WB_F0[12]	FIL2_2D_WB_F0[11]	FIL2_2D_WB_F0[10]	FIL2_2D_WB_F0[9]	FIL2_2D_WB_F0[8]	
	FIL2_2D_WB_F0[7]	FIL2_2D_WB_F0[6]	FIL2_2D_WB_F0[5]	FIL2_2D_WB_F0[4]	FIL2_2D_WB_F0[3]	FIL2_2D_WB_F0[2]	FIL2_2D_WB_F0[1]	FIL2_2D_WB_F0[0]	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Digital video decoder	YCTWB_F1_1	-	-	-	FIL2_2D_WB_F1[12]	FIL2_2D_WB_F1[11]	FIL2_2D_WB_F1[10]	FIL2_2D_WB_F1[9]	FIL2_2D_WB_F1[8]
		FIL2_2D_WB_F1[7]	FIL2_2D_WB_F1[6]	FIL2_2D_WB_F1[5]	FIL2_2D_WB_F1[4]	FIL2_2D_WB_F1[3]	FIL2_2D_WB_F1[2]	FIL2_2D_WB_F1[1]	FIL2_2D_WB_F1[0]
	YCTWB_F2_1	-	-	-	FIL2_2D_WB_F2[12]	FIL2_2D_WB_F2[11]	FIL2_2D_WB_F2[10]	FIL2_2D_WB_F2[9]	FIL2_2D_WB_F2[8]
		FIL2_2D_WB_F2[7]	FIL2_2D_WB_F2[6]	FIL2_2D_WB_F2[5]	FIL2_2D_WB_F2[4]	FIL2_2D_WB_F2[3]	FIL2_2D_WB_F2[2]	FIL2_2D_WB_F2[1]	FIL2_2D_WB_F2[0]
	YCTWB_F3_1	-	-	-	FIL2_2D_WB_F3[12]	FIL2_2D_WB_F3[11]	FIL2_2D_WB_F3[10]	FIL2_2D_WB_F3[9]	FIL2_2D_WB_F3[8]
		FIL2_2D_WB_F3[7]	FIL2_2D_WB_F3[6]	FIL2_2D_WB_F3[5]	FIL2_2D_WB_F3[4]	FIL2_2D_WB_F3[3]	FIL2_2D_WB_F3[2]	FIL2_2D_WB_F3[1]	FIL2_2D_WB_F3[0]
	YCTWB_F4_1	-	-	-	FIL2_2D_WB_F4[12]	FIL2_2D_WB_F4[11]	FIL2_2D_WB_F4[10]	FIL2_2D_WB_F4[9]	FIL2_2D_WB_F4[8]
		FIL2_2D_WB_F4[7]	FIL2_2D_WB_F4[6]	FIL2_2D_WB_F4[5]	FIL2_2D_WB_F4[4]	FIL2_2D_WB_F4[3]	FIL2_2D_WB_F4[2]	FIL2_2D_WB_F4[1]	FIL2_2D_WB_F4[0]
	YCTWB_F5_1	-	-	-	FIL2_2D_WB_F5[12]	FIL2_2D_WB_F5[11]	FIL2_2D_WB_F5[10]	FIL2_2D_WB_F5[9]	FIL2_2D_WB_F5[8]
		FIL2_2D_WB_F5[7]	FIL2_2D_WB_F5[6]	FIL2_2D_WB_F5[5]	FIL2_2D_WB_F5[4]	FIL2_2D_WB_F5[3]	FIL2_2D_WB_F5[2]	FIL2_2D_WB_F5[1]	FIL2_2D_WB_F5[0]
	YCTWB_F6_1	-	-	-	FIL2_2D_WB_F6[12]	FIL2_2D_WB_F6[11]	FIL2_2D_WB_F6[10]	FIL2_2D_WB_F6[9]	FIL2_2D_WB_F6[8]
		FIL2_2D_WB_F6[7]	FIL2_2D_WB_F6[6]	FIL2_2D_WB_F6[5]	FIL2_2D_WB_F6[4]	FIL2_2D_WB_F6[3]	FIL2_2D_WB_F6[2]	FIL2_2D_WB_F6[1]	FIL2_2D_WB_F6[0]
	YCTWB_F7_1	-	-	-	FIL2_2D_WB_F7[12]	FIL2_2D_WB_F7[11]	FIL2_2D_WB_F7[10]	FIL2_2D_WB_F7[9]	FIL2_2D_WB_F7[8]
		FIL2_2D_WB_F7[7]	FIL2_2D_WB_F7[6]	FIL2_2D_WB_F7[5]	FIL2_2D_WB_F7[4]	FIL2_2D_WB_F7[3]	FIL2_2D_WB_F7[2]	FIL2_2D_WB_F7[1]	FIL2_2D_WB_F7[0]
	YCTWB_F8_1	-	-	-	FIL2_2D_WB_F8[12]	FIL2_2D_WB_F8[11]	FIL2_2D_WB_F8[10]	FIL2_2D_WB_F8[9]	FIL2_2D_WB_F8[8]
		FIL2_2D_WB_F8[7]	FIL2_2D_WB_F8[6]	FIL2_2D_WB_F8[5]	FIL2_2D_WB_F8[4]	FIL2_2D_WB_F8[3]	FIL2_2D_WB_F8[2]	FIL2_2D_WB_F8[1]	FIL2_2D_WB_F8[0]
	YCTNA_F0_1	-	-	-	FIL2_2D_NA_F0[12]	FIL2_2D_NA_F0[11]	FIL2_2D_NA_F0[10]	FIL2_2D_NA_F0[9]	FIL2_2D_NA_F0[8]
		FIL2_2D_NA_F0[7]	FIL2_2D_NA_F0[6]	FIL2_2D_NA_F0[5]	FIL2_2D_NA_F0[4]	FIL2_2D_NA_F0[3]	FIL2_2D_NA_F0[2]	FIL2_2D_NA_F0[1]	FIL2_2D_NA_F0[0]
	YCTNA_F1_1	-	-	-	FIL2_2D_NA_F1[12]	FIL2_2D_NA_F1[11]	FIL2_2D_NA_F1[10]	FIL2_2D_NA_F1[9]	FIL2_2D_NA_F1[8]
		FIL2_2D_NA_F1[7]	FIL2_2D_NA_F1[6]	FIL2_2D_NA_F1[5]	FIL2_2D_NA_F1[4]	FIL2_2D_NA_F1[3]	FIL2_2D_NA_F1[2]	FIL2_2D_NA_F1[1]	FIL2_2D_NA_F1[0]
	YCTNA_F2_1	-	-	-	FIL2_2D_NA_F2[12]	FIL2_2D_NA_F2[11]	FIL2_2D_NA_F2[10]	FIL2_2D_NA_F2[9]	FIL2_2D_NA_F2[8]
		FIL2_2D_NA_F2[7]	FIL2_2D_NA_F2[6]	FIL2_2D_NA_F2[5]	FIL2_2D_NA_F2[4]	FIL2_2D_NA_F2[3]	FIL2_2D_NA_F2[2]	FIL2_2D_NA_F2[1]	FIL2_2D_NA_F2[0]
	YCTNA_F3_1	-	-	-	FIL2_2D_NA_F3[12]	FIL2_2D_NA_F3[11]	FIL2_2D_NA_F3[10]	FIL2_2D_NA_F3[9]	FIL2_2D_NA_F3[8]
		FIL2_2D_NA_F3[7]	FIL2_2D_NA_F3[6]	FIL2_2D_NA_F3[5]	FIL2_2D_NA_F3[4]	FIL2_2D_NA_F3[3]	FIL2_2D_NA_F3[2]	FIL2_2D_NA_F3[1]	FIL2_2D_NA_F3[0]
	YCTNA_F4_1	-	-	-	FIL2_2D_NA_F4[12]	FIL2_2D_NA_F4[11]	FIL2_2D_NA_F4[10]	FIL2_2D_NA_F4[9]	FIL2_2D_NA_F4[8]
		FIL2_2D_NA_F4[7]	FIL2_2D_NA_F4[6]	FIL2_2D_NA_F4[5]	FIL2_2D_NA_F4[4]	FIL2_2D_NA_F4[3]	FIL2_2D_NA_F4[2]	FIL2_2D_NA_F4[1]	FIL2_2D_NA_F4[0]
	YCTNA_F5_1	-	-	-	FIL2_2D_NA_F5[12]	FIL2_2D_NA_F5[11]	FIL2_2D_NA_F5[10]	FIL2_2D_NA_F5[9]	FIL2_2D_NA_F5[8]
		FIL2_2D_NA_F5[7]	FIL2_2D_NA_F5[6]	FIL2_2D_NA_F5[5]	FIL2_2D_NA_F5[4]	FIL2_2D_NA_F5[3]	FIL2_2D_NA_F5[2]	FIL2_2D_NA_F5[1]	FIL2_2D_NA_F5[0]
	YCTNA_F6_1	-	-	-	FIL2_2D_NA_F6[12]	FIL2_2D_NA_F6[11]	FIL2_2D_NA_F6[10]	FIL2_2D_NA_F6[9]	FIL2_2D_NA_F6[8]
		FIL2_2D_NA_F6[7]	FIL2_2D_NA_F6[6]	FIL2_2D_NA_F6[5]	FIL2_2D_NA_F6[4]	FIL2_2D_NA_F6[3]	FIL2_2D_NA_F6[2]	FIL2_2D_NA_F6[1]	FIL2_2D_NA_F6[0]
	YCTNA_F7_1	-	-	-	FIL2_2D_NA_F7[12]	FIL2_2D_NA_F7[11]	FIL2_2D_NA_F7[10]	FIL2_2D_NA_F7[9]	FIL2_2D_NA_F7[8]
		FIL2_2D_NA_F7[7]	FIL2_2D_NA_F7[6]	FIL2_2D_NA_F7[5]	FIL2_2D_NA_F7[4]	FIL2_2D_NA_F7[3]	FIL2_2D_NA_F7[2]	FIL2_2D_NA_F7[1]	FIL2_2D_NA_F7[0]
	YCTNA_F8_1	-	-	-	FIL2_2D_NA_F8[12]	FIL2_2D_NA_F8[11]	FIL2_2D_NA_F8[10]	FIL2_2D_NA_F8[9]	FIL2_2D_NA_F8[8]
		FIL2_2D_NA_F8[7]	FIL2_2D_NA_F8[6]	FIL2_2D_NA_F8[5]	FIL2_2D_NA_F8[4]	FIL2_2D_NA_F8[3]	FIL2_2D_NA_F8[2]	FIL2_2D_NA_F8[1]	FIL2_2D_NA_F8[0]
	YCTNB_F0_1	-	-	-	FIL2_2D_NB_F0[12]	FIL2_2D_NB_F0[11]	FIL2_2D_NB_F0[10]	FIL2_2D_NB_F0[9]	FIL2_2D_NB_F0[8]
		FIL2_2D_NB_F0[7]	FIL2_2D_NB_F0[6]	FIL2_2D_NB_F0[5]	FIL2_2D_NB_F0[4]	FIL2_2D_NB_F0[3]	FIL2_2D_NB_F0[2]	FIL2_2D_NB_F0[1]	FIL2_2D_NB_F0[0]
	YCTNB_F1_1	-	-	-	FIL2_2D_NB_F1[12]	FIL2_2D_NB_F1[11]	FIL2_2D_NB_F1[10]	FIL2_2D_NB_F1[9]	FIL2_2D_NB_F1[8]
		FIL2_2D_NB_F1[7]	FIL2_2D_NB_F1[6]	FIL2_2D_NB_F1[5]	FIL2_2D_NB_F1[4]	FIL2_2D_NB_F1[3]	FIL2_2D_NB_F1[2]	FIL2_2D_NB_F1[1]	FIL2_2D_NB_F1[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0		
Digital video decoder	YCTNB_F2_1	-	-	-	FIL2_2D_NB_F2[12]	FIL2_2D_NB_F2[11]	FIL2_2D_NB_F2[10]	FIL2_2D_NB_F2[9]	FIL2_2D_NB_F2[8]		
		FIL2_2D_NB_F2[7]	FIL2_2D_NB_F2[6]	FIL2_2D_NB_F2[5]	FIL2_2D_NB_F2[4]	FIL2_2D_NB_F2[3]	FIL2_2D_NB_F2[2]	FIL2_2D_NB_F2[1]	FIL2_2D_NB_F2[0]		
	YCTNB_F3_1	-	-	-	FIL2_2D_NB_F3[12]	FIL2_2D_NB_F3[11]	FIL2_2D_NB_F3[10]	FIL2_2D_NB_F3[9]	FIL2_2D_NB_F3[8]		
		FIL2_2D_NB_F3[7]	FIL2_2D_NB_F3[6]	FIL2_2D_NB_F3[5]	FIL2_2D_NB_F3[4]	FIL2_2D_NB_F3[3]	FIL2_2D_NB_F3[2]	FIL2_2D_NB_F3[1]	FIL2_2D_NB_F3[0]		
	YCTNB_F4_1	-	-	-	FIL2_2D_NB_F4[12]	FIL2_2D_NB_F4[11]	FIL2_2D_NB_F4[10]	FIL2_2D_NB_F4[9]	FIL2_2D_NB_F4[8]		
		FIL2_2D_NB_F4[7]	FIL2_2D_NB_F4[6]	FIL2_2D_NB_F4[5]	FIL2_2D_NB_F4[4]	FIL2_2D_NB_F4[3]	FIL2_2D_NB_F4[2]	FIL2_2D_NB_F4[1]	FIL2_2D_NB_F4[0]		
	YCTNB_F5_1	-	-	-	FIL2_2D_NB_F5[12]	FIL2_2D_NB_F5[11]	FIL2_2D_NB_F5[10]	FIL2_2D_NB_F5[9]	FIL2_2D_NB_F5[8]		
		FIL2_2D_NB_F5[7]	FIL2_2D_NB_F5[6]	FIL2_2D_NB_F5[5]	FIL2_2D_NB_F5[4]	FIL2_2D_NB_F5[3]	FIL2_2D_NB_F5[2]	FIL2_2D_NB_F5[1]	FIL2_2D_NB_F5[0]		
	YCTNB_F6_1	-	-	-	FIL2_2D_NB_F6[12]	FIL2_2D_NB_F6[11]	FIL2_2D_NB_F6[10]	FIL2_2D_NB_F6[9]	FIL2_2D_NB_F6[8]		
		FIL2_2D_NB_F6[7]	FIL2_2D_NB_F6[6]	FIL2_2D_NB_F6[5]	FIL2_2D_NB_F6[4]	FIL2_2D_NB_F6[3]	FIL2_2D_NB_F6[2]	FIL2_2D_NB_F6[1]	FIL2_2D_NB_F6[0]		
	YCTNB_F7_1	-	-	-	FIL2_2D_NB_F7[12]	FIL2_2D_NB_F7[11]	FIL2_2D_NB_F7[10]	FIL2_2D_NB_F7[9]	FIL2_2D_NB_F7[8]		
		FIL2_2D_NB_F7[7]	FIL2_2D_NB_F7[6]	FIL2_2D_NB_F7[5]	FIL2_2D_NB_F7[4]	FIL2_2D_NB_F7[3]	FIL2_2D_NB_F7[2]	FIL2_2D_NB_F7[1]	FIL2_2D_NB_F7[0]		
	YCTNB_F8_1	-	-	-	FIL2_2D_NB_F8[12]	FIL2_2D_NB_F8[11]	FIL2_2D_NB_F8[10]	FIL2_2D_NB_F8[9]	FIL2_2D_NB_F8[8]		
		FIL2_2D_NB_F8[7]	FIL2_2D_NB_F8[6]	FIL2_2D_NB_F8[5]	FIL2_2D_NB_F8[4]	FIL2_2D_NB_F8[3]	FIL2_2D_NB_F8[2]	FIL2_2D_NB_F8[1]	FIL2_2D_NB_F8[0]		
	YGAINCR_1	-	-	-	-	-	-	-	Y_GAIN2[9]	Y_GAIN2[8]	
		Y_GAIN2[7]	Y_GAIN2[6]	Y_GAIN2[5]	Y_GAIN2[4]	Y_GAIN2[3]	Y_GAIN2[2]	Y_GAIN2[1]	Y_GAIN2[0]		
	CBGAINCR_1	-	-	-	-	-	-	-	CB_GAIN2[9]	CB_GAIN2[8]	
		CB_GAIN2[7]	CB_GAIN2[6]	CB_GAIN2[5]	CB_GAIN2[4]	CB_GAIN2[3]	CB_GAIN2[2]	CB_GAIN2[1]	CB_GAIN2[0]		
	CRGAINCR_1	-	-	-	-	-	-	-	CR_GAIN2[9]	CR_GAIN2[8]	
		CR_GAIN2[7]	CR_GAIN2[6]	CR_GAIN2[5]	CR_GAIN2[4]	CR_GAIN2[3]	CR_GAIN2[2]	CR_GAIN2[1]	CR_GAIN2[0]		
	PGA_UPDATE_1	-	-	-	-	-	-	-	-	PGA_VEN	
		-	-	-	-	-	-	-	-	-	
	PGACR_1	-	PGA_GAIN_SEL	PGA_GAIN[5]	PGA_GAIN[4]	PGA_GAIN[3]	PGA_GAIN[2]	PGA_GAIN[1]	PGA_GAIN[0]		
		-	-	-	-	-	-	-	-	-	
	ADCCR2_1	-	-	-	-	-	-	-	-	ADC_VINSEL	
		-	-	-	-	-	-	-	-	-	
	Video display controller 5 channel 0	INP_UPDATE	-	-	-	-	-	-	-	-	
			-	-	-	-	-	-	-	-	
			-	-	-	-	-	-	-	-	
			-	-	-	INP_EXT_UPDATE	-	-	-	INP_IMG_UPDATE	
		INP_SEL_CNT	-	-	-	-	-	-	-	-	
			-	-	-	INP_SEL	-	-	-	-	
			-	INP_FORMAT[2]	INP_FORMAT[1]	INP_FORMAT[0]	-	-	-	INP_PXD_EDGE	
			-	-	-	INP_VS_EDGE	-	-	-	INP_HS_EDGE	
		INP_EXT_SYNC_CNT	-	-	-	INP_ENDIAN_ON	-	-	-	INP_SWAP_ON	
			-	-	-	INP_VS_INV	-	-	-	INP_HS_INV	
			-	-	-	-	-	-	-	INP_H_EDGE_SEL	
			-	-	-	INP_F525_625	-	-	INP_H_POS[1]	INP_H_POS[0]	
		INP_VSYNC_PH_ADJ	-	-	-	-	-	-	-	INP_FH50[9]	INP_FH50[8]
			INP_FH50[7]	INP_FH50[6]	INP_FH50[5]	INP_FH50[4]	INP_FH50[3]	INP_FH50[2]	INP_FH50[1]	INP_FH50[0]	
			-	-	-	-	-	-	-	INP_FH25[9]	INP_FH25[8]
			INP_FH25[7]	INP_FH25[6]	INP_FH25[5]	INP_FH25[4]	INP_FH25[3]	INP_FH25[2]	INP_FH25[1]	INP_FH25[0]	
		INP_DLY_ADJ	-	-	-	-	-	INP_VS_DLY_L[2]	INP_VS_DLY_L[1]	INP_VS_DLY_L[0]	
			INP_FLD_DLY[7]	INP_FLD_DLY[6]	INP_FLD_DLY[5]	INP_FLD_DLY[4]	INP_FLD_DLY[3]	INP_FLD_DLY[2]	INP_FLD_DLY[1]	INP_FLD_DLY[0]	
			INP_VS_DLY[7]	INP_VS_DLY[6]	INP_VS_DLY[5]	INP_VS_DLY[4]	INP_VS_DLY[3]	INP_VS_DLY[2]	INP_VS_DLY[1]	INP_VS_DLY[0]	
			INP_HS_DLY[7]	INP_HS_DLY[6]	INP_HS_DLY[5]	INP_HS_DLY[4]	INP_HS_DLY[3]	INP_HS_DLY[2]	INP_HS_DLY[1]	INP_HS_DLY[0]	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Video display controller 5 channel 0	IMGCNT_UPDATE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	IMGCNT_VEN
	IMGCNT_NR_CNT0	-	-	-	-	-	-	-	-
		-	-	-	NR1D_MD	-	-	-	NR1D_ON
		-	NR1D_Y_TH[6]	NR1D_Y_TH[5]	NR1D_Y_TH[4]	NR1D_Y_TH[3]	NR1D_Y_TH[2]	NR1D_Y_TH[1]	NR1D_Y_TH[0]
		-	-	NR1D_Y_TAP[1]	NR1D_Y_TAP[0]	-	-	NR1D_Y_GAIN[1]	NR1D_Y_GAIN[0]
	IMGCNT_NR_CNT1	-	NR1D_CB_TH[6]	NR1D_CB_TH[5]	NR1D_CB_TH[4]	NR1D_CB_TH[3]	NR1D_CB_TH[2]	NR1D_CB_TH[1]	NR1D_CB_TH[0]
		-	-	NR1D_CB_TAP[1]	NR1D_CB_TAP[0]	-	-	NR1D_CB_GAIN [1]	NR1D_CB_GAIN [0]
		-	NR1D_CR_TH[6]	NR1D_CR_TH[5]	NR1D_CR_TH[4]	NR1D_CR_TH[3]	NR1D_CR_TH[2]	NR1D_CR_TH[1]	NR1D_CR_TH[0]
		-	-	NR1D_CR_TAP[1]	NR1D_CR_TAP[0]	-	-	NR1D_CR_GAIN [1]	NR1D_CR_GAIN [0]
	IMGCNT_MTX_MODE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	IMGCNT_MTX_MD[1]	IMGCNT_MTX_MD[0]
	IMGCNT_MTX_YG_ADJ0	-	-	-	-	-	-	-	-
		IMGCNT_MTX_YG[7]	IMGCNT_MTX_YG[6]	IMGCNT_MTX_YG[5]	IMGCNT_MTX_YG[4]	IMGCNT_MTX_YG[3]	IMGCNT_MTX_YG[2]	IMGCNT_MTX_YG[1]	IMGCNT_MTX_YG[0]
		-	-	-	-	-	IMGCNT_MTX_GG[10]	IMGCNT_MTX_GG[9]	IMGCNT_MTX_GG[8]
	IMGCNT_MTX_YG_ADJ1	IMGCNT_MTX_GG[7]	IMGCNT_MTX_GG[6]	IMGCNT_MTX_GG[5]	IMGCNT_MTX_GG[4]	IMGCNT_MTX_GG[3]	IMGCNT_MTX_GG[2]	IMGCNT_MTX_GG[1]	IMGCNT_MTX_GG[0]
		-	-	-	-	-	IMGCNT_MTX_GB[10]	IMGCNT_MTX_GB[9]	IMGCNT_MTX_GB[8]
		IMGCNT_MTX_GB[7]	IMGCNT_MTX_GB[6]	IMGCNT_MTX_GB[5]	IMGCNT_MTX_GB[4]	IMGCNT_MTX_GB[3]	IMGCNT_MTX_GB[2]	IMGCNT_MTX_GB[1]	IMGCNT_MTX_GB[0]
	IMGCNT_MTX_YG_CBB_ADJ0	-	-	-	-	-	-	-	-
		IMGCNT_MTX_GR[7]	IMGCNT_MTX_GR[6]	IMGCNT_MTX_GR[5]	IMGCNT_MTX_GR[4]	IMGCNT_MTX_GR[3]	IMGCNT_MTX_GR[2]	IMGCNT_MTX_GR[1]	IMGCNT_MTX_GR[0]
		-	-	-	-	-	IMGCNT_MTX_BB[10]	IMGCNT_MTX_BB[9]	IMGCNT_MTX_BB[8]
	IMGCNT_MTX_YG_CBB_ADJ1	IMGCNT_MTX_BB[7]	IMGCNT_MTX_BB[6]	IMGCNT_MTX_BB[5]	IMGCNT_MTX_BB[4]	IMGCNT_MTX_BB[3]	IMGCNT_MTX_BB[2]	IMGCNT_MTX_BB[1]	IMGCNT_MTX_BB[0]
		-	-	-	-	-	IMGCNT_MTX_BR[10]	IMGCNT_MTX_BR[9]	IMGCNT_MTX_BR[8]
		IMGCNT_MTX_BR[7]	IMGCNT_MTX_BR[6]	IMGCNT_MTX_BR[5]	IMGCNT_MTX_BR[4]	IMGCNT_MTX_BR[3]	IMGCNT_MTX_BR[2]	IMGCNT_MTX_BR[1]	IMGCNT_MTX_BR[0]
	IMGCNT_MTX_CRR_ADJ0	-	-	-	-	-	-	-	-
		IMGCNT_MTX_RG[7]	IMGCNT_MTX_RG[6]	IMGCNT_MTX_RG[5]	IMGCNT_MTX_RG[4]	IMGCNT_MTX_RG[3]	IMGCNT_MTX_RG[2]	IMGCNT_MTX_RG[1]	IMGCNT_MTX_RG[0]
		-	-	-	-	-	IMGCNT_MTX_RB[10]	IMGCNT_MTX_RB[9]	IMGCNT_MTX_RB[8]
	IMGCNT_MTX_CRR_ADJ1	IMGCNT_MTX_RB[7]	IMGCNT_MTX_RB[6]	IMGCNT_MTX_RB[5]	IMGCNT_MTX_RB[4]	IMGCNT_MTX_RB[3]	IMGCNT_MTX_RB[2]	IMGCNT_MTX_RB[1]	IMGCNT_MTX_RB[0]
		-	-	-	-	-	IMGCNT_MTX_RR[10]	IMGCNT_MTX_RR[9]	IMGCNT_MTX_RR[8]
		IMGCNT_MTX_RR[7]	IMGCNT_MTX_RR[6]	IMGCNT_MTX_RR[5]	IMGCNT_MTX_RR[4]	IMGCNT_MTX_RR[3]	IMGCNT_MTX_RR[2]	IMGCNT_MTX_RR[1]	IMGCNT_MTX_RR[0]
	IMGCNT_DRC_REG	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	DRC_EN



Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 0	SC0_SCL0_UPDATE	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	SC0_SCL0_VEN_D	SC0_SCL0_VEN_C	-	-	-	SC0_SCL0_UPDATE	
		-	-	-	SC0_SCL0_VEN_B	-	-	-	SC0_SCL0_VEN_A	
	SC0_SCL0_FRC1	SC0_RES_VMASK[15]	SC0_RES_VMASK[14]	SC0_RES_VMASK[13]	SC0_RES_VMASK[12]	SC0_RES_VMASK[11]	SC0_RES_VMASK[10]	SC0_RES_VMASK[9]	SC0_RES_VMASK[8]	
		SC0_RES_VMASK[7]	SC0_RES_VMASK[6]	SC0_RES_VMASK[5]	SC0_RES_VMASK[4]	SC0_RES_VMASK[3]	SC0_RES_VMASK[2]	SC0_RES_VMASK[1]	SC0_RES_VMASK[0]	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	SC0_RES_VMASK_ON	
	SC0_SCL0_FRC2	SC0_RES_VLACK[15]	SC0_RES_VLACK[14]	SC0_RES_VLACK[13]	SC0_RES_VLACK[12]	SC0_RES_VLACK[11]	SC0_RES_VLACK[10]	SC0_RES_VLACK[9]	SC0_RES_VLACK[8]	
		SC0_RES_VLACK[7]	SC0_RES_VLACK[6]	SC0_RES_VLACK[5]	SC0_RES_VLACK[4]	SC0_RES_VLACK[3]	SC0_RES_VLACK[2]	SC0_RES_VLACK[1]	SC0_RES_VLACK[0]	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	SC0_RES_VLACK_ON	
	SC0_SCL0_FRC3	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	SC0_RES_VS_IN_SEL	
		-	-	-	-	-	-	-	SC0_RES_VS_SEL	
	SC0_SCL0_FRC4	-	-	-	-	-	-	SC0_RES_FV[10]	SC0_RES_FV[9]	SC0_RES_FV[8]
		SC0_RES_FV[7]	SC0_RES_FV[6]	SC0_RES_FV[5]	SC0_RES_FV[4]	SC0_RES_FV[3]	SC0_RES_FV[2]	SC0_RES_FV[1]	SC0_RES_FV[0]	
		-	-	-	-	-	SC0_RES_FH[10]	SC0_RES_FH[9]	SC0_RES_FH[8]	
		SC0_RES_FH[7]	SC0_RES_FH[6]	SC0_RES_FH[5]	SC0_RES_FH[4]	SC0_RES_FH[3]	SC0_RES_FH[2]	SC0_RES_FH[1]	SC0_RES_FH[0]	
	SC0_SCL0_FRC5	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		SC0_RES_VSDLY[7]	SC0_RES_VSDLY[6]	SC0_RES_VSDLY[5]	SC0_RES_VSDLY[4]	SC0_RES_VSDLY[3]	SC0_RES_VSDLY[2]	SC0_RES_VSDLY[1]	SC0_RES_VSDLY[0]	
	SC0_SCL0_FRC6	-	-	-	-	-	-	SC0_RES_F_VS[10]	SC0_RES_F_VS[9]	SC0_RES_F_VS[8]
		SC0_RES_F_VS[7]	SC0_RES_F_VS[6]	SC0_RES_F_VS[5]	SC0_RES_F_VS[4]	SC0_RES_F_VS[3]	SC0_RES_F_VS[2]	SC0_RES_F_VS[1]	SC0_RES_F_VS[0]	
		-	-	-	-	-	SC0_RES_F_VW[10]	SC0_RES_F_VW[9]	SC0_RES_F_VW[8]	
		SC0_RES_F_VW[7]	SC0_RES_F_VW[6]	SC0_RES_F_VW[5]	SC0_RES_F_VW[4]	SC0_RES_F_VW[3]	SC0_RES_F_VW[2]	SC0_RES_F_VW[1]	SC0_RES_F_VW[0]	
	SC0_SCL0_FRC7	-	-	-	-	-	-	SC0_RES_F_HS[10]	SC0_RES_F_HS[9]	SC0_RES_F_HS[8]
		SC0_RES_F_HS[7]	SC0_RES_F_HS[6]	SC0_RES_F_HS[5]	SC0_RES_F_HS[4]	SC0_RES_F_HS[3]	SC0_RES_F_HS[2]	SC0_RES_F_HS[1]	SC0_RES_F_HS[0]	
		-	-	-	-	-	SC0_RES_F_HW[10]	SC0_RES_F_HW[9]	SC0_RES_F_HW[8]	
		SC0_RES_F_HW[7]	SC0_RES_F_HW[6]	SC0_RES_F_HW[5]	SC0_RES_F_HW[4]	SC0_RES_F_HW[3]	SC0_RES_F_HW[2]	SC0_RES_F_HW[1]	SC0_RES_F_HW[0]	
	SC0_SCL0_FRC9	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	SC0_RES_QVLOCK	-	-	-	-	SC0_RES_QVLACK
	SC0_SCL0_MON0	-	-	-	-	-	-	SC0_RES_LIN_STAT[10]	SC0_RES_LIN_STAT[9]	SC0_RES_LIN_STAT[8]
		SC0_RES_LIN_STAT[7]	SC0_RES_LIN_STAT[6]	SC0_RES_LIN_STAT[5]	SC0_RES_LIN_STAT[4]	SC0_RES_LIN_STAT[3]	SC0_RES_LIN_STAT[2]	SC0_RES_LIN_STAT[1]	SC0_RES_LIN_STAT[0]	
	SC0_SCL0_INT	-	-	-	-	-	-	SC0_RES_LINE[10]	SC0_RES_LINE[9]	SC0_RES_LINE[8]
		SC0_RES_LINE[7]	SC0_RES_LINE[6]	SC0_RES_LINE[5]	SC0_RES_LINE[4]	SC0_RES_LINE[3]	SC0_RES_LINE[2]	SC0_RES_LINE[1]	SC0_RES_LINE[0]	
	SC0_SCL0_DS1	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	SC0_RES_DS_V_ON	-	-	-	-	SC0_RES_DS_H_ON

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 0	SC0_SCL0_DS2	-	-	-	-	-	SC0_RES_VS[10]	SC0_RES_VS[9]	SC0_RES_VS[8]	
		SC0_RES_VS[7]	SC0_RES_VS[6]	SC0_RES_VS[5]	SC0_RES_VS[4]	SC0_RES_VS[3]	SC0_RES_VS[2]	SC0_RES_VS[1]	SC0_RES_VS[0]	
		-	-	-	-	-	SC0_RES_VW[10]	SC0_RES_VW[9]	SC0_RES_VW[8]	
		SC0_RES_VW[7]	SC0_RES_VW[6]	SC0_RES_VW[5]	SC0_RES_VW[4]	SC0_RES_VW[3]	SC0_RES_VW[2]	SC0_RES_VW[1]	SC0_RES_VW[0]	
	SC0_SCL0_DS3	-	-	-	-	-	-	SC0_RES_HS[10]	SC0_RES_HS[9]	SC0_RES_HS[8]
		SC0_RES_HS[7]	SC0_RES_HS[6]	SC0_RES_HS[5]	SC0_RES_HS[4]	SC0_RES_HS[3]	SC0_RES_HS[2]	SC0_RES_HS[1]	SC0_RES_HS[0]	
		-	-	-	-	-	SC0_RES_HW[10]	SC0_RES_HW[9]	SC0_RES_HW[8]	
		SC0_RES_HW[7]	SC0_RES_HW[6]	SC0_RES_HW[5]	SC0_RES_HW[4]	SC0_RES_HW[3]	SC0_RES_HW[2]	SC0_RES_HW[1]	SC0_RES_HW[0]	
	SC0_SCL0_DS4	-	-	SC0_RES_PFIL_SEL	SC0_RES_DS_H_INTERPOTYP	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		SC0_RES_DS_H_RATIO[15]	SC0_RES_DS_H_RATIO[14]	SC0_RES_DS_H_RATIO[13]	SC0_RES_DS_H_RATIO[12]	SC0_RES_DS_H_RATIO[11]	SC0_RES_DS_H_RATIO[10]	SC0_RES_DS_H_RATIO[9]	SC0_RES_DS_H_RATIO[8]	
		SC0_RES_DS_H_RATIO[7]	SC0_RES_DS_H_RATIO[6]	SC0_RES_DS_H_RATIO[5]	SC0_RES_DS_H_RATIO[4]	SC0_RES_DS_H_RATIO[3]	SC0_RES_DS_H_RATIO[2]	SC0_RES_DS_H_RATIO[1]	SC0_RES_DS_H_RATIO[0]	
	SC0_SCL0_DS5	-	-	-	SC0_RES_V_INTERPOTYP	SC0_RES_TOP_INIPHASE[11]	SC0_RES_TOP_INIPHASE[10]	SC0_RES_TOP_INIPHASE[9]	SC0_RES_TOP_INIPHASE[8]	
		SC0_RES_TOP_INIPHASE[7]	SC0_RES_TOP_INIPHASE[6]	SC0_RES_TOP_INIPHASE[5]	SC0_RES_TOP_INIPHASE[4]	SC0_RES_TOP_INIPHASE[3]	SC0_RES_TOP_INIPHASE[2]	SC0_RES_TOP_INIPHASE[1]	SC0_RES_TOP_INIPHASE[0]	
		-	-	-	-	SC0_RES_BTM_INIPHASE[11]	SC0_RES_BTM_INIPHASE[10]	SC0_RES_BTM_INIPHASE[9]	SC0_RES_BTM_INIPHASE[8]	
		SC0_RES_BTM_INIPHASE[7]	SC0_RES_BTM_INIPHASE[6]	SC0_RES_BTM_INIPHASE[5]	SC0_RES_BTM_INIPHASE[4]	SC0_RES_BTM_INIPHASE[3]	SC0_RES_BTM_INIPHASE[2]	SC0_RES_BTM_INIPHASE[1]	SC0_RES_BTM_INIPHASE[0]	
	SC0_SCL0_DS6	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		SC0_RES_V_RATIO[15]	SC0_RES_V_RATIO[14]	SC0_RES_V_RATIO[13]	SC0_RES_V_RATIO[12]	SC0_RES_V_RATIO[11]	SC0_RES_V_RATIO[10]	SC0_RES_V_RATIO[9]	SC0_RES_V_RATIO[8]	
		SC0_RES_V_RATIO[7]	SC0_RES_V_RATIO[6]	SC0_RES_V_RATIO[5]	SC0_RES_V_RATIO[4]	SC0_RES_V_RATIO[3]	SC0_RES_V_RATIO[2]	SC0_RES_V_RATIO[1]	SC0_RES_V_RATIO[0]	
	SC0_SCL0_DS7	-	-	-	-	-	-	SC0_RES_OUT_VW[10]	SC0_RES_OUT_VW[9]	SC0_RES_OUT_VW[8]
		SC0_RES_OUT_VW[7]	SC0_RES_OUT_VW[6]	SC0_RES_OUT_VW[5]	SC0_RES_OUT_VW[4]	SC0_RES_OUT_VW[3]	SC0_RES_OUT_VW[2]	SC0_RES_OUT_VW[1]	SC0_RES_OUT_VW[0]	
		-	-	-	-	-	SC0_RES_OUT_HW[10]	SC0_RES_OUT_HW[9]	SC0_RES_OUT_HW[8]	
		SC0_RES_OUT_HW[7]	SC0_RES_OUT_HW[6]	SC0_RES_OUT_HW[5]	SC0_RES_OUT_HW[4]	SC0_RES_OUT_HW[3]	SC0_RES_OUT_HW[2]	SC0_RES_OUT_HW[1]	SC0_RES_OUT_HW[0]	
	SC0_SCL0_US1	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	SC0_RES_US_V_ON	-	-	-	-	SC0_RES_US_H_ON
	SC0_SCL0_US2	-	-	-	-	-	-	SC0_RES_P_VS[10]	SC0_RES_P_VS[9]	SC0_RES_P_VS[8]
		SC0_RES_P_VS[7]	SC0_RES_P_VS[6]	SC0_RES_P_VS[5]	SC0_RES_P_VS[4]	SC0_RES_P_VS[3]	SC0_RES_P_VS[2]	SC0_RES_P_VS[1]	SC0_RES_P_VS[0]	
		-	-	-	-	-	SC0_RES_P_VW[10]	SC0_RES_P_VW[9]	SC0_RES_P_VW[8]	
		SC0_RES_P_VW[7]	SC0_RES_P_VW[6]	SC0_RES_P_VW[5]	SC0_RES_P_VW[4]	SC0_RES_P_VW[3]	SC0_RES_P_VW[2]	SC0_RES_P_VW[1]	SC0_RES_P_VW[0]	
	SC0_SCL0_US3	-	-	-	-	-	-	SC0_RES_P_HS[10]	SC0_RES_P_HS[9]	SC0_RES_P_HS[8]
		SC0_RES_P_HS[7]	SC0_RES_P_HS[6]	SC0_RES_P_HS[5]	SC0_RES_P_HS[4]	SC0_RES_P_HS[3]	SC0_RES_P_HS[2]	SC0_RES_P_HS[1]	SC0_RES_P_HS[0]	
		-	-	-	-	-	SC0_RES_P_HW[10]	SC0_RES_P_HW[9]	SC0_RES_P_HW[8]	
		SC0_RES_P_HW[7]	SC0_RES_P_HW[6]	SC0_RES_P_HW[5]	SC0_RES_P_HW[4]	SC0_RES_P_HW[3]	SC0_RES_P_HW[2]	SC0_RES_P_HW[1]	SC0_RES_P_HW[0]	
	SC0_SCL0_US4	-	-	-	-	-	-	SC0_RES_IN_VW[10]	SC0_RES_IN_VW[9]	SC0_RES_IN_VW[8]
		SC0_RES_IN_VW[7]	SC0_RES_IN_VW[6]	SC0_RES_IN_VW[5]	SC0_RES_IN_VW[4]	SC0_RES_IN_VW[3]	SC0_RES_IN_VW[2]	SC0_RES_IN_VW[1]	SC0_RES_IN_VW[0]	
		-	-	-	-	-	SC0_RES_IN_HW[10]	SC0_RES_IN_HW[9]	SC0_RES_IN_HW[8]	
		SC0_RES_IN_HW[7]	SC0_RES_IN_HW[6]	SC0_RES_IN_HW[5]	SC0_RES_IN_HW[4]	SC0_RES_IN_HW[3]	SC0_RES_IN_HW[2]	SC0_RES_IN_HW[1]	SC0_RES_IN_HW[0]	

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 0	SC0_SCL0_US5	-	-	-	-	-	-	-	-	
	SC0_RES_US_H_RATIO[15]	SC0_RES_US_H_RATIO[14]	SC0_RES_US_H_RATIO[13]	SC0_RES_US_H_RATIO[12]	SC0_RES_US_H_RATIO[11]	SC0_RES_US_H_RATIO[10]	SC0_RES_US_H_RATIO[9]	SC0_RES_US_H_RATIO[8]	SC0_RES_US_H_RATIO[0]	
	SC0_RES_US_H_RATIO[7]	SC0_RES_US_H_RATIO[6]	SC0_RES_US_H_RATIO[5]	SC0_RES_US_H_RATIO[4]	SC0_RES_US_H_RATIO[3]	SC0_RES_US_H_RATIO[2]	SC0_RES_US_H_RATIO[1]	SC0_RES_US_H_RATIO[0]	SC0_RES_US_H_RATIO[0]	
	SC0_SCL0_US6	-	-	-	SC0_RES_US_H_INTE RPTYP	SC0_RES_US_HT_INIPHASE[11]	SC0_RES_US_HT_INI PHASE[10]	SC0_RES_US_HT_INI PHASE[9]	SC0_RES_US_HT_INI PHASE[8]	SC0_RES_US_HT_INI PHASE[0]
	SC0_RES_US_HT_INI PHASE[7]	SC0_RES_US_HT_INI PHASE[6]	SC0_RES_US_HT_INI PHASE[5]	SC0_RES_US_HT_INI PHASE[4]	SC0_RES_US_HT_INI PHASE[3]	SC0_RES_US_HT_INI PHASE[2]	SC0_RES_US_HT_INI PHASE[1]	SC0_RES_US_HT_INI PHASE[0]	SC0_RES_US_HT_INI PHASE[0]	
	-	-	-	-	SC0_RES_US_HB_INI PHASE[11]	SC0_RES_US_HB_INI PHASE[10]	SC0_RES_US_HB_INI PHASE[9]	SC0_RES_US_HB_INI PHASE[8]	SC0_RES_US_HB_INI PHASE[0]	
	SC0_RES_US_HB_INI PHASE[7]	SC0_RES_US_HB_INI PHASE[6]	SC0_RES_US_HB_INI PHASE[5]	SC0_RES_US_HB_INI PHASE[4]	SC0_RES_US_HB_INI PHASE[3]	SC0_RES_US_HB_INI PHASE[2]	SC0_RES_US_HB_INI PHASE[1]	SC0_RES_US_HB_INI PHASE[0]	SC0_RES_US_HB_INI PHASE[0]	
	SC0_SCL0_US7	-	-	-	-	-	-	-	-	
	SC0_RES_HRCUT [7]	SC0_RES_HRCUT [6]	SC0_RES_HRCUT [5]	SC0_RES_HRCUT [4]	SC0_RES_HRCUT [3]	SC0_RES_HRCUT [2]	SC0_RES_HRCUT [1]	SC0_RES_HRCUT [0]	SC0_RES_HRCUT [0]	
	SC0_RES_VRCUT [7]	SC0_RES_VRCUT [6]	SC0_RES_VRCUT [5]	SC0_RES_VRCUT [4]	SC0_RES_VRCUT [3]	SC0_RES_VRCUT [2]	SC0_RES_VRCUT [1]	SC0_RES_VRCUT [0]	SC0_RES_VRCUT [0]	
SC0_SCL0_US8	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-		
-	-	-	-	-	-	-	-	-		
-	-	-	-	SC0_RES_IBUS_SYN C_SEL	-	-	-	-	SC0_RES_DISP_ON	
SC0_SCL0_OVR1	-	-	-	-	-	-	-	-	-	
SC0_RES_BK_COL_R[7]	SC0_RES_BK_COL_R[6]	SC0_RES_BK_COL_R[5]	SC0_RES_BK_COL_R[4]	SC0_RES_BK_COL_R[3]	SC0_RES_BK_COL_R[2]	SC0_RES_BK_COL_R[1]	SC0_RES_BK_COL_R[0]	SC0_RES_BK_COL_R[0]		
SC0_RES_BK_COL_G[7]	SC0_RES_BK_COL_G[6]	SC0_RES_BK_COL_G[5]	SC0_RES_BK_COL_G[4]	SC0_RES_BK_COL_G[3]	SC0_RES_BK_COL_G[2]	SC0_RES_BK_COL_G[1]	SC0_RES_BK_COL_G[0]	SC0_RES_BK_COL_G[0]		
SC0_RES_BK_COL_B[7]	SC0_RES_BK_COL_B[6]	SC0_RES_BK_COL_B[5]	SC0_RES_BK_COL_B[4]	SC0_RES_BK_COL_B[3]	SC0_RES_BK_COL_B[2]	SC0_RES_BK_COL_B[1]	SC0_RES_BK_COL_B[0]	SC0_RES_BK_COL_B[0]		
SC0_SCL1_UPDATE	-	-	-	-	-	-	-	-	-	
-	-	-	-	SC0_SCL1_UPDATE_B	-	-	-	-	SC0_SCL1_UPDATE_A	
-	-	-	-	-	-	-	-	-	-	
-	-	-	-	SC0_SCL1_VEN_B	-	-	-	-	SC0_SCL1_VEN_A	
SC0_SCL1_WR1	-	-	-	-	-	-	-	-	-	
-	-	-	-	-	-	SC0_RES_WRSWA[2]	SC0_RES_WRSWA[1]	SC0_RES_WRSWA[0]	SC0_RES_WRSWA[0]	
-	-	-	-	-	-	-	-	-	-	
SC0_RES_TB_ADD_MOD	SC0_RES_DS_WR_MD[2]	SC0_RES_DS_WR_MD[1]	SC0_RES_DS_WR_MD[0]	SC0_RES_MD[1]	SC0_RES_MD[0]	SC0_RES_LOOP	SC0_RES_BST_MD	SC0_RES_BST_MD		
SC0_SCL1_WR2	SC0_RES_BASE [31]	SC0_RES_BASE [30]	SC0_RES_BASE [29]	SC0_RES_BASE [28]	SC0_RES_BASE [27]	SC0_RES_BASE [26]	SC0_RES_BASE [25]	SC0_RES_BASE [24]		
SC0_RES_BAS E[23]	SC0_RES_BASE [22]	SC0_RES_BASE [21]	SC0_RES_BASE [20]	SC0_RES_BASE [19]	SC0_RES_BASE [18]	SC0_RES_BASE [17]	SC0_RES_BASE [16]	SC0_RES_BASE [16]		
SC0_RES_BASE [15]	SC0_RES_BASE [14]	SC0_RES_BASE [13]	SC0_RES_BASE [12]	SC0_RES_BASE [11]	SC0_RES_BASE [10]	SC0_RES_BASE [9]	SC0_RES_BASE [8]	SC0_RES_BASE [8]		
SC0_RES_BASE [7]	SC0_RES_BASE [6]	SC0_RES_BASE [5]	SC0_RES_BASE [4]	SC0_RES_BASE [3]	SC0_RES_BASE [2]	SC0_RES_BASE [1]	SC0_RES_BASE [0]	SC0_RES_BASE [0]		
SC0_SCL1_WR3	-	SC0_RES_LN_OFF[14]	SC0_RES_LN_OFF[13]	SC0_RES_LN_OFF[12]	SC0_RES_LN_OFF[11]	SC0_RES_LN_OFF[10]	SC0_RES_LN_OFF[9]	SC0_RES_LN_OFF[8]		
SC0_RES_LN_OFF[7]	SC0_RES_LN_OFF[6]	SC0_RES_LN_OFF[5]	SC0_RES_LN_OFF[4]	SC0_RES_LN_OFF[3]	SC0_RES_LN_OFF[2]	SC0_RES_LN_OFF[1]	SC0_RES_LN_OFF[0]	SC0_RES_LN_OFF[0]		
-	-	-	-	-	-	-	SC0_RES_FLM_NUM [9]	SC0_RES_FLM_NUM [8]		
SC0_RES_FLM_NUM [7]	SC0_RES_FLM_NUM [6]	SC0_RES_FLM_NUM [5]	SC0_RES_FLM_NUM [4]	SC0_RES_FLM_NUM [3]	SC0_RES_FLM_NUM [2]	SC0_RES_FLM_NUM [1]	SC0_RES_FLM_NUM [0]	SC0_RES_FLM_NUM [0]		
SC0_SCL1_WR4	-	-	-	-	-	-	-	-		
-	SC0_RES_FLM_OFF[22]	SC0_RES_FLM_OFF[21]	SC0_RES_FLM_OFF[20]	SC0_RES_FLM_OFF[19]	SC0_RES_FLM_OFF[18]	SC0_RES_FLM_OFF[17]	SC0_RES_FLM_OFF[16]	SC0_RES_FLM_OFF[16]		
SC0_RES_FLM_OFF[15]	SC0_RES_FLM_OFF[14]	SC0_RES_FLM_OFF[13]	SC0_RES_FLM_OFF[12]	SC0_RES_FLM_OFF[11]	SC0_RES_FLM_OFF[10]	SC0_RES_FLM_OFF[9]	SC0_RES_FLM_OFF[8]	SC0_RES_FLM_OFF[8]		
SC0_RES_FLM_OFF[7]	SC0_RES_FLM_OFF[6]	SC0_RES_FLM_OFF[5]	SC0_RES_FLM_OFF[4]	SC0_RES_FLM_OFF[3]	SC0_RES_FLM_OFF[2]	SC0_RES_FLM_OFF[1]	SC0_RES_FLM_OFF[0]	SC0_RES_FLM_OFF[0]		

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Video display controller 5 channel 0	SC0_SCL1_WR5	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	SC0_RES_INTER	-	-	SC0_RES_FS_RATE[1]	SC0_RES_FS_RATE[0]
		-	-	-	SC0_RES_FLD_SEL	-	-	-	SC0_RES_WENB
	SC0_SCL1_WR6	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	SC0_RES_DTH_ON	-	-	-	SC0_RES_BITDEC_ON
	SC0_SCL1_WR7	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	SC0_RES_OVERFLOW
		-	-	-	-	-	-	SC0_RES_FLM_CNT[9]	SC0_RES_FLM_CNT[8]
		SC0_RES_FLM_CNT[7]	SC0_RES_FLM_CNT[6]	SC0_RES_FLM_CNT[5]	SC0_RES_FLM_CNT[4]	SC0_RES_FLM_CNT[3]	SC0_RES_FLM_CNT[2]	SC0_RES_FLM_CNT[1]	SC0_RES_FLM_CNT[0]
	SC0_SCL1_WR8	SC0_RES_BASE_B[31]	SC0_RES_BASE_B[30]	SC0_RES_BASE_B[29]	SC0_RES_BASE_B[28]	SC0_RES_BASE_B[27]	SC0_RES_BASE_B[26]	SC0_RES_BASE_B[25]	SC0_RES_BASE_B[24]
		SC0_RES_BASE_B[23]	SC0_RES_BASE_B[22]	SC0_RES_BASE_B[21]	SC0_RES_BASE_B[20]	SC0_RES_BASE_B[19]	SC0_RES_BASE_B[18]	SC0_RES_BASE_B[17]	SC0_RES_BASE_B[16]
		SC0_RES_BASE_B[15]	SC0_RES_BASE_B[14]	SC0_RES_BASE_B[13]	SC0_RES_BASE_B[12]	SC0_RES_BASE_B[11]	SC0_RES_BASE_B[10]	SC0_RES_BASE_B[9]	SC0_RES_BASE_B[8]
		SC0_RES_BASE_B[7]	SC0_RES_BASE_B[6]	SC0_RES_BASE_B[5]	SC0_RES_BASE_B[4]	SC0_RES_BASE_B[3]	SC0_RES_BASE_B[2]	SC0_RES_BASE_B[1]	SC0_RES_BASE_B[0]
	SC0_SCL1_WR9	-	SC0_RES_LN_OFF_B[14]	SC0_RES_LN_OFF_B[13]	SC0_RES_LN_OFF_B[12]	SC0_RES_LN_OFF_B[11]	SC0_RES_LN_OFF_B[10]	SC0_RES_LN_OFF_B[9]	SC0_RES_LN_OFF_B[8]
		SC0_RES_LN_OFF_B[7]	SC0_RES_LN_OFF_B[6]	SC0_RES_LN_OFF_B[5]	SC0_RES_LN_OFF_B[4]	SC0_RES_LN_OFF_B[3]	SC0_RES_LN_OFF_B[2]	SC0_RES_LN_OFF_B[1]	SC0_RES_LN_OFF_B[0]
		-	-	-	-	-	-	SC0_RES_FLM_NUM_B[9]	SC0_RES_FLM_NUM_B[8]
		SC0_RES_FLM_NUM_B[7]	SC0_RES_FLM_NUM_B[6]	SC0_RES_FLM_NUM_B[5]	SC0_RES_FLM_NUM_B[4]	SC0_RES_FLM_NUM_B[3]	SC0_RES_FLM_NUM_B[2]	SC0_RES_FLM_NUM_B[1]	SC0_RES_FLM_NUM_B[0]
	SC0_SCL1_WR10	-	-	-	-	-	-	-	-
		-	SC0_RES_FLM_OFF_B[22]	SC0_RES_FLM_OFF_B[21]	SC0_RES_FLM_OFF_B[20]	SC0_RES_FLM_OFF_B[19]	SC0_RES_FLM_OFF_B[18]	SC0_RES_FLM_OFF_B[17]	SC0_RES_FLM_OFF_B[16]
		SC0_RES_FLM_OFF_B[15]	SC0_RES_FLM_OFF_B[14]	SC0_RES_FLM_OFF_B[13]	SC0_RES_FLM_OFF_B[12]	SC0_RES_FLM_OFF_B[11]	SC0_RES_FLM_OFF_B[10]	SC0_RES_FLM_OFF_B[9]	SC0_RES_FLM_OFF_B[8]
		SC0_RES_FLM_OFF_B[7]	SC0_RES_FLM_OFF_B[6]	SC0_RES_FLM_OFF_B[5]	SC0_RES_FLM_OFF_B[4]	SC0_RES_FLM_OFF_B[3]	SC0_RES_FLM_OFF_B[2]	SC0_RES_FLM_OFF_B[1]	SC0_RES_FLM_OFF_B[0]
	SC0_SCL1_WR11	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	SC0_RES_FLM_CNT_B[9]	SC0_RES_FLM_CNT_B[8]
		SC0_RES_FLM_CNT_B[7]	SC0_RES_FLM_CNT_B[6]	SC0_RES_FLM_CNT_B[5]	SC0_RES_FLM_CNT_B[4]	SC0_RES_FLM_CNT_B[3]	SC0_RES_FLM_CNT_B[2]	SC0_RES_FLM_CNT_B[1]	SC0_RES_FLM_CNT_B[0]
	SC0_SCL1_MON1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	SC0_PBUF_NUM[1]	SC0_PBUF_NUM[0]
		-	-	-	-	-	-	-	-
	SC0_SCL1_PBUF0	SC0_PBUF0_ADD[31]	SC0_PBUF0_ADD[30]	SC0_PBUF0_ADD[29]	SC0_PBUF0_ADD[28]	SC0_PBUF0_ADD[27]	SC0_PBUF0_ADD[26]	SC0_PBUF0_ADD[25]	SC0_PBUF0_ADD[24]
		SC0_PBUF0_ADD[23]	SC0_PBUF0_ADD[22]	SC0_PBUF0_ADD[21]	SC0_PBUF0_ADD[20]	SC0_PBUF0_ADD[19]	SC0_PBUF0_ADD[18]	SC0_PBUF0_ADD[17]	SC0_PBUF0_ADD[16]
		SC0_PBUF0_ADD[15]	SC0_PBUF0_ADD[14]	SC0_PBUF0_ADD[13]	SC0_PBUF0_ADD[12]	SC0_PBUF0_ADD[11]	SC0_PBUF0_ADD[10]	SC0_PBUF0_ADD[9]	SC0_PBUF0_ADD[8]
		SC0_PBUF0_ADD[7]	SC0_PBUF0_ADD[6]	SC0_PBUF0_ADD[5]	SC0_PBUF0_ADD[4]	SC0_PBUF0_ADD[3]	SC0_PBUF0_ADD[2]	SC0_PBUF0_ADD[1]	SC0_PBUF0_ADD[0]
	SC0_SCL1_PBUF1	SC0_PBUF1_ADD[31]	SC0_PBUF1_ADD[30]	SC0_PBUF1_ADD[29]	SC0_PBUF1_ADD[28]	SC0_PBUF1_ADD[27]	SC0_PBUF1_ADD[26]	SC0_PBUF1_ADD[25]	SC0_PBUF1_ADD[24]
		SC0_PBUF1_ADD[23]	SC0_PBUF1_ADD[22]	SC0_PBUF1_ADD[21]	SC0_PBUF1_ADD[20]	SC0_PBUF1_ADD[19]	SC0_PBUF1_ADD[18]	SC0_PBUF1_ADD[17]	SC0_PBUF1_ADD[16]
		SC0_PBUF1_ADD[15]	SC0_PBUF1_ADD[14]	SC0_PBUF1_ADD[13]	SC0_PBUF1_ADD[12]	SC0_PBUF1_ADD[11]	SC0_PBUF1_ADD[10]	SC0_PBUF1_ADD[9]	SC0_PBUF1_ADD[8]
		SC0_PBUF1_ADD[7]	SC0_PBUF1_ADD[6]	SC0_PBUF1_ADD[5]	SC0_PBUF1_ADD[4]	SC0_PBUF1_ADD[3]	SC0_PBUF1_ADD[2]	SC0_PBUF1_ADD[1]	SC0_PBUF1_ADD[0]

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 0	SC0_SCL1_PBUF2	SC0_PBUF2_ADD[31]	SC0_PBUF2_ADD[30]	SC0_PBUF2_ADD[29]	SC0_PBUF2_ADD[28]	SC0_PBUF2_ADD[27]	SC0_PBUF2_ADD[26]	SC0_PBUF2_ADD[25]	SC0_PBUF2_ADD[24]	
		SC0_PBUF2_ADD[23]	SC0_PBUF2_ADD[22]	SC0_PBUF2_ADD[21]	SC0_PBUF2_ADD[20]	SC0_PBUF2_ADD[19]	SC0_PBUF2_ADD[18]	SC0_PBUF2_ADD[17]	SC0_PBUF2_ADD[16]	
		SC0_PBUF2_ADD[15]	SC0_PBUF2_ADD[14]	SC0_PBUF2_ADD[13]	SC0_PBUF2_ADD[12]	SC0_PBUF2_ADD[11]	SC0_PBUF2_ADD[10]	SC0_PBUF2_ADD[9]	SC0_PBUF2_ADD[8]	
		SC0_PBUF2_ADD[7]	SC0_PBUF2_ADD[6]	SC0_PBUF2_ADD[5]	SC0_PBUF2_ADD[4]	SC0_PBUF2_ADD[3]	SC0_PBUF2_ADD[2]	SC0_PBUF2_ADD[1]	SC0_PBUF2_ADD[0]	
	SC0_SCL1_PBUF3	SC0_PBUF3_ADD[31]	SC0_PBUF3_ADD[30]	SC0_PBUF3_ADD[29]	SC0_PBUF3_ADD[28]	SC0_PBUF3_ADD[27]	SC0_PBUF3_ADD[26]	SC0_PBUF3_ADD[25]	SC0_PBUF3_ADD[24]	
		SC0_PBUF3_ADD[23]	SC0_PBUF3_ADD[22]	SC0_PBUF3_ADD[21]	SC0_PBUF3_ADD[20]	SC0_PBUF3_ADD[19]	SC0_PBUF3_ADD[18]	SC0_PBUF3_ADD[17]	SC0_PBUF3_ADD[16]	
		SC0_PBUF3_ADD[15]	SC0_PBUF3_ADD[14]	SC0_PBUF3_ADD[13]	SC0_PBUF3_ADD[12]	SC0_PBUF3_ADD[11]	SC0_PBUF3_ADD[10]	SC0_PBUF3_ADD[9]	SC0_PBUF3_ADD[8]	
		SC0_PBUF3_ADD[7]	SC0_PBUF3_ADD[6]	SC0_PBUF3_ADD[5]	SC0_PBUF3_ADD[4]	SC0_PBUF3_ADD[3]	SC0_PBUF3_ADD[2]	SC0_PBUF3_ADD[1]	SC0_PBUF3_ADD[0]	
	SC0_SCL1_PBUF_FLD	-	-	-	-	-	-	-	-	SC0_FLD_INF3
		-	-	-	-	-	-	-	-	SC0_FLD_INF2
		-	-	-	-	-	-	-	-	SC0_FLD_INF1
		-	-	-	-	-	-	-	-	SC0_FLD_INF0
	SC0_SCL1_PBUF_CNT	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	SC0_PBUF_RST
		-	-	-	-	-	-	-	-	-
	GR0_UPDATE	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	GR0_UPDATE
		-	-	-	GR0_P_VEN	-	-	-	-	GR0_IBUS_VEN
	GR0_FLM_RD	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	GR0_R_ENB
	GR0_FLM1	GR0_FLD_SEL	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	GR0_LN_OFF_DIR
		-	-	-	-	-	-	GR0_FLM_SEL[1]	GR0_FLM_SEL[0]	-
		-	-	-	GR0_JMR_FLM_INV	-	-	-	-	GR0_BST_MD
	GR0_FLM2	GR0_BASE[31]	GR0_BASE[30]	GR0_BASE[29]	GR0_BASE[28]	GR0_BASE[27]	GR0_BASE[26]	GR0_BASE[25]	GR0_BASE[24]	
		GR0_BASE[23]	GR0_BASE[22]	GR0_BASE[21]	GR0_BASE[20]	GR0_BASE[19]	GR0_BASE[18]	GR0_BASE[17]	GR0_BASE[16]	
		GR0_BASE[15]	GR0_BASE[14]	GR0_BASE[13]	GR0_BASE[12]	GR0_BASE[11]	GR0_BASE[10]	GR0_BASE[9]	GR0_BASE[8]	
		GR0_BASE[7]	GR0_BASE[6]	GR0_BASE[5]	GR0_BASE[4]	GR0_BASE[3]	GR0_BASE[2]	GR0_BASE[1]	GR0_BASE[0]	
	GR0_FLM3	GR0_FLD_NXT	GR0_LN_OFF[14]	GR0_LN_OFF[13]	GR0_LN_OFF[12]	GR0_LN_OFF[11]	GR0_LN_OFF[10]	GR0_LN_OFF[9]	GR0_LN_OFF[8]	
		GR0_LN_OFF[7]	GR0_LN_OFF[6]	GR0_LN_OFF[5]	GR0_LN_OFF[4]	GR0_LN_OFF[3]	GR0_LN_OFF[2]	GR0_LN_OFF[1]	GR0_LN_OFF[0]	
		-	-	-	-	-	-	GR0_FLM_NUM[9]	GR0_FLM_NUM[8]	
		GR0_FLM_NUM[7]	GR0_FLM_NUM[6]	GR0_FLM_NUM[5]	GR0_FLM_NUM[4]	GR0_FLM_NUM[3]	GR0_FLM_NUM[2]	GR0_FLM_NUM[1]	GR0_FLM_NUM[0]	
	GR0_FLM4	-	-	-	-	-	-	-	-	
		-	GR0_FLM_OFF[22]	GR0_FLM_OFF[21]	GR0_FLM_OFF[20]	GR0_FLM_OFF[19]	GR0_FLM_OFF[18]	GR0_FLM_OFF[17]	GR0_FLM_OFF[16]	
		GR0_FLM_OFF[15]	GR0_FLM_OFF[14]	GR0_FLM_OFF[13]	GR0_FLM_OFF[12]	GR0_FLM_OFF[11]	GR0_FLM_OFF[10]	GR0_FLM_OFF[9]	GR0_FLM_OFF[8]	
		GR0_FLM_OFF[7]	GR0_FLM_OFF[6]	GR0_FLM_OFF[5]	GR0_FLM_OFF[4]	GR0_FLM_OFF[3]	GR0_FLM_OFF[2]	GR0_FLM_OFF[1]	GR0_FLM_OFF[0]	
	GR0_FLM5	-	-	-	-	-	GR0_FLM_LNUM[10]	GR0_FLM_LNUM[9]	GR0_FLM_LNUM[8]	
		GR0_FLM_LNUM[7]	GR0_FLM_LNUM[6]	GR0_FLM_LNUM[5]	GR0_FLM_LNUM[4]	GR0_FLM_LNUM[3]	GR0_FLM_LNUM[2]	GR0_FLM_LNUM[1]	GR0_FLM_LNUM[0]	
		-	-	-	-	-	GR0_FLM_LOOP[10]	GR0_FLM_LOOP[9]	GR0_FLM_LOOP[8]	
		GR0_FLM_LOOP[7]	GR0_FLM_LOOP[6]	GR0_FLM_LOOP[5]	GR0_FLM_LOOP[4]	GR0_FLM_LOOP[3]	GR0_FLM_LOOP[2]	GR0_FLM_LOOP[1]	GR0_FLM_LOOP[0]	
	GR0_FLM6	GR0_FORMAT[3]	GR0_FORMAT[2]	GR0_FORMAT[1]	GR0_FORMAT[0]	-	GR0_HW[10]	GR0_HW[9]	GR0_HW[8]	
		GR0_HW[7]	GR0_HW[6]	GR0_HW[5]	GR0_HW[4]	GR0_HW[3]	GR0_HW[2]	GR0_HW[1]	GR0_HW[0]	
		GR0_YCC_SWAP[2]	GR0_YCC_SWAP[1]	GR0_YCC_SWAP[0]	GR0_RDSWA[2]	GR0_RDSWA[1]	GR0_RDSWA[0]	-	GR0_CN444_MD	
		-	-	GR0_STA_POS[5]	GR0_STA_POS[4]	GR0_STA_POS[3]	GR0_STA_POS[2]	GR0_STA_POS[1]	GR0_STA_POS[0]	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Video display controller 5 channel 0	GR0_AB1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	GR0_GRC_DISP_ON	-	-	GR0_DISP_SEL[1]	GR0_DISP_SEL[0]
	GR0_AB2	-	-	-	-	-	GR0_GRC_VS[10]	GR0_GRC_VS[9]	GR0_GRC_VS[8]
		GR0_GRC_VS[7]	GR0_GRC_VS[6]	GR0_GRC_VS[5]	GR0_GRC_VS[4]	GR0_GRC_VS[3]	GR0_GRC_VS[2]	GR0_GRC_VS[1]	GR0_GRC_VS[0]
		-	-	-	-	-	GR0_GRC_VW[10]	GR0_GRC_VW[9]	GR0_GRC_VW[8]
		GR0_GRC_VW[7]	GR0_GRC_VW[6]	GR0_GRC_VW[5]	GR0_GRC_VW[4]	GR0_GRC_VW[3]	GR0_GRC_VW[2]	GR0_GRC_VW[1]	GR0_GRC_VW[0]
	GR0_AB3	-	-	-	-	-	GR0_GRC_HS[10]	GR0_GRC_HS[9]	GR0_GRC_HS[8]
		GR0_GRC_HS[7]	GR0_GRC_HS[6]	GR0_GRC_HS[5]	GR0_GRC_HS[4]	GR0_GRC_HS[3]	GR0_GRC_HS[2]	GR0_GRC_HS[1]	GR0_GRC_HS[0]
		-	-	-	-	-	GR0_GRC_HW[10]	GR0_GRC_HW[9]	GR0_GRC_HW[8]
		GR0_GRC_HW[7]	GR0_GRC_HW[6]	GR0_GRC_HW[5]	GR0_GRC_HW[4]	GR0_GRC_HW[3]	GR0_GRC_HW[2]	GR0_GRC_HW[1]	GR0_GRC_HW[0]
	GR0_AB7	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GR0_CK_ON
	GR0_AB8	GR0_CK_KCLUT[7]	GR0_CK_KCLUT[6]	GR0_CK_KCLUT[5]	GR0_CK_KCLUT[4]	GR0_CK_KCLUT[3]	GR0_CK_KCLUT[2]	GR0_CK_KCLUT[1]	GR0_CK_KCLUT[0]
		GR0_CK_KG[7]	GR0_CK_KG[6]	GR0_CK_KG[5]	GR0_CK_KG[4]	GR0_CK_KG[3]	GR0_CK_KG[2]	GR0_CK_KG[1]	GR0_CK_KG[0]
		GR0_CK_KB[7]	GR0_CK_KB[6]	GR0_CK_KB[5]	GR0_CK_KB[4]	GR0_CK_KB[3]	GR0_CK_KB[2]	GR0_CK_KB[1]	GR0_CK_KB[0]
		GR0_CK_KR[7]	GR0_CK_KR[6]	GR0_CK_KR[5]	GR0_CK_KR[4]	GR0_CK_KR[3]	GR0_CK_KR[2]	GR0_CK_KR[1]	GR0_CK_KR[0]
	GR0_AB9	GR0_CK_A[7]	GR0_CK_A[6]	GR0_CK_A[5]	GR0_CK_A[4]	GR0_CK_A[3]	GR0_CK_A[2]	GR0_CK_A[1]	GR0_CK_A[0]
		GR0_CK_G[7]	GR0_CK_G[6]	GR0_CK_G[5]	GR0_CK_G[4]	GR0_CK_G[3]	GR0_CK_G[2]	GR0_CK_G[1]	GR0_CK_G[0]
		GR0_CK_B[7]	GR0_CK_B[6]	GR0_CK_B[5]	GR0_CK_B[4]	GR0_CK_B[3]	GR0_CK_B[2]	GR0_CK_B[1]	GR0_CK_B[0]
		GR0_CK_R[7]	GR0_CK_R[6]	GR0_CK_R[5]	GR0_CK_R[4]	GR0_CK_R[3]	GR0_CK_R[2]	GR0_CK_R[1]	GR0_CK_R[0]
	GR0_AB10	GR0_A0[7]	GR0_A0[6]	GR0_A0[5]	GR0_A0[4]	GR0_A0[3]	GR0_A0[2]	GR0_A0[1]	GR0_A0[0]
		GR0_G0[7]	GR0_G0[6]	GR0_G0[5]	GR0_G0[4]	GR0_G0[3]	GR0_G0[2]	GR0_G0[1]	GR0_G0[0]
		GR0_B0[7]	GR0_B0[6]	GR0_B0[5]	GR0_B0[4]	GR0_B0[3]	GR0_B0[2]	GR0_B0[1]	GR0_B0[0]
		GR0_R0[7]	GR0_R0[6]	GR0_R0[5]	GR0_R0[4]	GR0_R0[3]	GR0_R0[2]	GR0_R0[1]	GR0_R0[0]
	GR0_AB11	GR0_A1[7]	GR0_A1[6]	GR0_A1[5]	GR0_A1[4]	GR0_A1[3]	GR0_A1[2]	GR0_A1[1]	GR0_A1[0]
		GR0_G1[7]	GR0_G1[6]	GR0_G1[5]	GR0_G1[4]	GR0_G1[3]	GR0_G1[2]	GR0_G1[1]	GR0_G1[0]
		GR0_B1[7]	GR0_B1[6]	GR0_B1[5]	GR0_B1[4]	GR0_B1[3]	GR0_B1[2]	GR0_B1[1]	GR0_B1[0]
		GR0_R1[7]	GR0_R1[6]	GR0_R1[5]	GR0_R1[4]	GR0_R1[3]	GR0_R1[2]	GR0_R1[1]	GR0_R1[0]
	GR0_BASE	-	-	-	-	-	-	-	-
		GR0_BASE_G[7]	GR0_BASE_G[6]	GR0_BASE_G[5]	GR0_BASE_G[4]	GR0_BASE_G[3]	GR0_BASE_G[2]	GR0_BASE_G[1]	GR0_BASE_G[0]
		GR0_BASE_B[7]	GR0_BASE_B[6]	GR0_BASE_B[5]	GR0_BASE_B[4]	GR0_BASE_B[3]	GR0_BASE_B[2]	GR0_BASE_B[1]	GR0_BASE_B[0]
		GR0_BASE_R[7]	GR0_BASE_R[6]	GR0_BASE_R[5]	GR0_BASE_R[4]	GR0_BASE_R[3]	GR0_BASE_R[2]	GR0_BASE_R[1]	GR0_BASE_R[0]
	GR0_CLUT	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GR0_CLT_SEL
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	SC1_SCL0_UPDATE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	SC1_SCL0_VEN_D	SC1_SCL0_VEN_C	-	-	-	SC1_SCL0_UPDATE
		-	-	-	SC1_SCL0_VEN_B	-	-	-	SC1_SCL0_VEN_A
	SC1_SCL0_FRC1	SC1_RES_VMASK[15]	SC1_RES_VMASK[14]	SC1_RES_VMASK[13]	SC1_RES_VMASK[12]	SC1_RES_VMASK[11]	SC1_RES_VMASK[10]	SC1_RES_VMASK[9]	SC1_RES_VMASK[8]
		SC1_RES_VMASK[7]	SC1_RES_VMASK[6]	SC1_RES_VMASK[5]	SC1_RES_VMASK[4]	SC1_RES_VMASK[3]	SC1_RES_VMASK[2]	SC1_RES_VMASK[1]	SC1_RES_VMASK[0]
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	SC1_RES_VMASK_ON

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 0	SC1_SCL0_FRC2	SC1_RES_VLACK[15]	SC1_RES_VLACK[14]	SC1_RES_VLACK[13]	SC1_RES_VLACK[12]	SC1_RES_VLACK[11]	SC1_RES_VLACK[10]	SC1_RES_VLACK[9]	SC1_RES_VLACK[8]	
		SC1_RES_VLACK[7]	SC1_RES_VLACK[6]	SC1_RES_VLACK[5]	SC1_RES_VLACK[4]	SC1_RES_VLACK[3]	SC1_RES_VLACK[2]	SC1_RES_VLACK[1]	SC1_RES_VLACK[0]	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	SC1_RES_VLACK_ON	
	SC1_SCL0_FRC3	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	SC1_RES_VS_IN_SEL	
		-	-	-	-	-	-	-	SC1_RES_VS_SEL	
	SC1_SCL0_FRC4	-	-	-	-	-	-	SC1_RES_FV[10]	SC1_RES_FV[9]	SC1_RES_FV[8]
		SC1_RES_FV[7]	SC1_RES_FV[6]	SC1_RES_FV[5]	SC1_RES_FV[4]	SC1_RES_FV[3]	SC1_RES_FV[2]	SC1_RES_FV[1]	SC1_RES_FV[0]	
		-	-	-	-	-	SC1_RES_FH[10]	SC1_RES_FH[9]	SC1_RES_FH[8]	
		SC1_RES_FH[7]	SC1_RES_FH[6]	SC1_RES_FH[5]	SC1_RES_FH[4]	SC1_RES_FH[3]	SC1_RES_FH[2]	SC1_RES_FH[1]	SC1_RES_FH[0]	
	SC1_SCL0_FRC5	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	SC1_RES_FLD_DLY_SEL	
		SC1_RES_VSDLY[7]	SC1_RES_VSDLY[6]	SC1_RES_VSDLY[5]	SC1_RES_VSDLY[4]	SC1_RES_VSDLY[3]	SC1_RES_VSDLY[2]	SC1_RES_VSDLY[1]	SC1_RES_VSDLY[0]	
	SC1_SCL0_FRC6	-	-	-	-	-	-	SC1_RES_F_VS[10]	SC1_RES_F_VS[9]	SC1_RES_F_VS[8]
		SC1_RES_F_VS[7]	SC1_RES_F_VS[6]	SC1_RES_F_VS[5]	SC1_RES_F_VS[4]	SC1_RES_F_VS[3]	SC1_RES_F_VS[2]	SC1_RES_F_VS[1]	SC1_RES_F_VS[0]	
		-	-	-	-	-	SC1_RES_F_VW[10]	SC1_RES_F_VW[9]	SC1_RES_F_VW[8]	
		SC1_RES_F_VW[7]	SC1_RES_F_VW[6]	SC1_RES_F_VW[5]	SC1_RES_F_VW[4]	SC1_RES_F_VW[3]	SC1_RES_F_VW[2]	SC1_RES_F_VW[1]	SC1_RES_F_VW[0]	
	SC1_SCL0_FRC7	-	-	-	-	-	-	SC1_RES_F_HS[10]	SC1_RES_F_HS[9]	SC1_RES_F_HS[8]
		SC1_RES_F_HS[7]	SC1_RES_F_HS[6]	SC1_RES_F_HS[5]	SC1_RES_F_HS[4]	SC1_RES_F_HS[3]	SC1_RES_F_HS[2]	SC1_RES_F_HS[1]	SC1_RES_F_HS[0]	
		-	-	-	-	-	SC1_RES_F_HW[10]	SC1_RES_F_HW[9]	SC1_RES_F_HW[8]	
		SC1_RES_F_HW[7]	SC1_RES_F_HW[6]	SC1_RES_F_HW[5]	SC1_RES_F_HW[4]	SC1_RES_F_HW[3]	SC1_RES_F_HW[2]	SC1_RES_F_HW[1]	SC1_RES_F_HW[0]	
	SC1_SCL0_FRC9	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	SC1_RES_QVLOCK	-	-	-	-	SC1_RES_QVLOCK
	SC1_SCL0_MON0	-	-	-	-	-	-	SC1_RES_LIN_STAT[10]	SC1_RES_LIN_STAT[9]	SC1_RES_LIN_STAT[8]
		SC1_RES_LIN_STAT[7]	SC1_RES_LIN_STAT[6]	SC1_RES_LIN_STAT[5]	SC1_RES_LIN_STAT[4]	SC1_RES_LIN_STAT[3]	SC1_RES_LIN_STAT[2]	SC1_RES_LIN_STAT[1]	SC1_RES_LIN_STAT[0]	
	SC1_SCL0_INT	-	-	-	-	-	-	SC1_RES_LINE[10]	SC1_RES_LINE[9]	SC1_RES_LINE[8]
		SC1_RES_LINE[7]	SC1_RES_LINE[6]	SC1_RES_LINE[5]	SC1_RES_LINE[4]	SC1_RES_LINE[3]	SC1_RES_LINE[2]	SC1_RES_LINE[1]	SC1_RES_LINE[0]	
	SC1_SCL0_DS1	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	SC1_RES_DS_V_ON	-	-	-	-	SC1_RES_DS_H_ON
	SC1_SCL0_DS2	-	-	-	-	-	-	SC1_RES_VS[10]	SC1_RES_VS[9]	SC1_RES_VS[8]
		SC1_RES_VS[7]	SC1_RES_VS[6]	SC1_RES_VS[5]	SC1_RES_VS[4]	SC1_RES_VS[3]	SC1_RES_VS[2]	SC1_RES_VS[1]	SC1_RES_VS[0]	
		-	-	-	-	-	SC1_RES_VW[10]	SC1_RES_VW[9]	SC1_RES_VW[8]	
		SC1_RES_VW[7]	SC1_RES_VW[6]	SC1_RES_VW[5]	SC1_RES_VW[4]	SC1_RES_VW[3]	SC1_RES_VW[2]	SC1_RES_VW[1]	SC1_RES_VW[0]	
	SC1_SCL0_DS3	-	-	-	-	-	-	SC1_RES_HS[10]	SC1_RES_HS[9]	SC1_RES_HS[8]
		SC1_RES_HS[7]	SC1_RES_HS[6]	SC1_RES_HS[5]	SC1_RES_HS[4]	SC1_RES_HS[3]	SC1_RES_HS[2]	SC1_RES_HS[1]	SC1_RES_HS[0]	
		-	-	-	-	-	SC1_RES_HW[10]	SC1_RES_HW[9]	SC1_RES_HW[8]	
		SC1_RES_HW[7]	SC1_RES_HW[6]	SC1_RES_HW[5]	SC1_RES_HW[4]	SC1_RES_HW[3]	SC1_RES_HW[2]	SC1_RES_HW[1]	SC1_RES_HW[0]	

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Video display controller 5 channel 0	SC1_SCL0_DS4	-	-	SC1_RES_PFIL_SEL	SC1_RES_DS_H_INT ERPOTYP	-	-	-	-
		-	-	-	-	-	-	-	-
		SC1_RES_DS_H_RATIO[15]	SC1_RES_DS_H_RATIO[14]	SC1_RES_DS_H_RATIO[13]	SC1_RES_DS_H_RATIO[12]	SC1_RES_DS_H_RATIO[11]	SC1_RES_DS_H_RATIO[10]	SC1_RES_DS_H_RATIO[9]	SC1_RES_DS_H_RATIO[8]
	SC1_RES_DS_H_RATIO[7]	SC1_RES_DS_H_RATIO[6]	SC1_RES_DS_H_RATIO[5]	SC1_RES_DS_H_RATIO[4]	SC1_RES_DS_H_RATIO[3]	SC1_RES_DS_H_RATIO[2]	SC1_RES_DS_H_RATIO[1]	SC1_RES_DS_H_RATIO[0]	
	SC1_SCL0_DS5	-	-	-	SC1_RES_V_INTERPOTYP	SC1_RES_TOP_INIPHASE[11]	SC1_RES_TOP_INIPHASE[10]	SC1_RES_TOP_INIPHASE[9]	SC1_RES_TOP_INIPHASE[8]
		SC1_RES_TOP_INIPHASE[7]	SC1_RES_TOP_INIPHASE[6]	SC1_RES_TOP_INIPHASE[5]	SC1_RES_TOP_INIPHASE[4]	SC1_RES_TOP_INIPHASE[3]	SC1_RES_TOP_INIPHASE[2]	SC1_RES_TOP_INIPHASE[1]	SC1_RES_TOP_INIPHASE[0]
		-	-	-	-	SC1_RES_BTM_INIPHASE[11]	SC1_RES_BTM_INIPHASE[10]	SC1_RES_BTM_INIPHASE[9]	SC1_RES_BTM_INIPHASE[8]
		SC1_RES_BTM_INIPHASE[7]	SC1_RES_BTM_INIPHASE[6]	SC1_RES_BTM_INIPHASE[5]	SC1_RES_BTM_INIPHASE[4]	SC1_RES_BTM_INIPHASE[3]	SC1_RES_BTM_INIPHASE[2]	SC1_RES_BTM_INIPHASE[1]	SC1_RES_BTM_INIPHASE[0]
	SC1_SCL0_DS6	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		SC1_RES_V_RATIO[15]	SC1_RES_V_RATIO[14]	SC1_RES_V_RATIO[13]	SC1_RES_V_RATIO[12]	SC1_RES_V_RATIO[11]	SC1_RES_V_RATIO[10]	SC1_RES_V_RATIO[9]	SC1_RES_V_RATIO[8]
	SC1_RES_V_RATIO[7]	SC1_RES_V_RATIO[6]	SC1_RES_V_RATIO[5]	SC1_RES_V_RATIO[4]	SC1_RES_V_RATIO[3]	SC1_RES_V_RATIO[2]	SC1_RES_V_RATIO[1]	SC1_RES_V_RATIO[0]	
	SC1_SCL0_DS7	-	-	-	-	-	SC1_RES_OUT_VW[10]	SC1_RES_OUT_VW[9]	SC1_RES_OUT_VW[8]
		SC1_RES_OUT_VW[7]	SC1_RES_OUT_VW[6]	SC1_RES_OUT_VW[5]	SC1_RES_OUT_VW[4]	SC1_RES_OUT_VW[3]	SC1_RES_OUT_VW[2]	SC1_RES_OUT_VW[1]	SC1_RES_OUT_VW[0]
		-	-	-	-	-	SC1_RES_OUT_HW[10]	SC1_RES_OUT_HW[9]	SC1_RES_OUT_HW[8]
		SC1_RES_OUT_HW[7]	SC1_RES_OUT_HW[6]	SC1_RES_OUT_HW[5]	SC1_RES_OUT_HW[4]	SC1_RES_OUT_HW[3]	SC1_RES_OUT_HW[2]	SC1_RES_OUT_HW[1]	SC1_RES_OUT_HW[0]
	SC1_SCL0_US1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	SC1_RES_US_V_ON	-	-	-	SC1_RES_US_H_ON
	SC1_SCL0_US2	-	-	-	-	-	SC1_RES_P_VS[10]	SC1_RES_P_VS[9]	SC1_RES_P_VS[8]
		SC1_RES_P_VS[7]	SC1_RES_P_VS[6]	SC1_RES_P_VS[5]	SC1_RES_P_VS[4]	SC1_RES_P_VS[3]	SC1_RES_P_VS[2]	SC1_RES_P_VS[1]	SC1_RES_P_VS[0]
		-	-	-	-	-	SC1_RES_P_VW[10]	SC1_RES_P_VW[9]	SC1_RES_P_VW[8]
		SC1_RES_P_VW[7]	SC1_RES_P_VW[6]	SC1_RES_P_VW[5]	SC1_RES_P_VW[4]	SC1_RES_P_VW[3]	SC1_RES_P_VW[2]	SC1_RES_P_VW[1]	SC1_RES_P_VW[0]
	SC1_SCL0_US3	-	-	-	-	-	SC1_RES_P_HS[10]	SC1_RES_P_HS[9]	SC1_RES_P_HS[8]
		SC1_RES_P_HS[7]	SC1_RES_P_HS[6]	SC1_RES_P_HS[5]	SC1_RES_P_HS[4]	SC1_RES_P_HS[3]	SC1_RES_P_HS[2]	SC1_RES_P_HS[1]	SC1_RES_P_HS[0]
		-	-	-	-	-	SC1_RES_P_HW[10]	SC1_RES_P_HW[9]	SC1_RES_P_HW[8]
		SC1_RES_P_HW[7]	SC1_RES_P_HW[6]	SC1_RES_P_HW[5]	SC1_RES_P_HW[4]	SC1_RES_P_HW[3]	SC1_RES_P_HW[2]	SC1_RES_P_HW[1]	SC1_RES_P_HW[0]
	SC1_SCL0_US4	-	-	-	-	-	SC1_RES_IN_VW[10]	SC1_RES_IN_VW[9]	SC1_RES_IN_VW[8]
		SC1_RES_IN_VW[7]	SC1_RES_IN_VW[6]	SC1_RES_IN_VW[5]	SC1_RES_IN_VW[4]	SC1_RES_IN_VW[3]	SC1_RES_IN_VW[2]	SC1_RES_IN_VW[1]	SC1_RES_IN_VW[0]
		-	-	-	-	-	SC1_RES_IN_HW[10]	SC1_RES_IN_HW[9]	SC1_RES_IN_HW[8]
		SC1_RES_IN_HW[7]	SC1_RES_IN_HW[6]	SC1_RES_IN_HW[5]	SC1_RES_IN_HW[4]	SC1_RES_IN_HW[3]	SC1_RES_IN_HW[2]	SC1_RES_IN_HW[1]	SC1_RES_IN_HW[0]
	SC1_SCL0_US5	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		SC1_RES_US_H_RATIO[15]	SC1_RES_US_H_RATIO[14]	SC1_RES_US_H_RATIO[13]	SC1_RES_US_H_RATIO[12]	SC1_RES_US_H_RATIO[11]	SC1_RES_US_H_RATIO[10]	SC1_RES_US_H_RATIO[9]	SC1_RES_US_H_RATIO[8]
	SC1_RES_US_H_RATIO[7]	SC1_RES_US_H_RATIO[6]	SC1_RES_US_H_RATIO[5]	SC1_RES_US_H_RATIO[4]	SC1_RES_US_H_RATIO[3]	SC1_RES_US_H_RATIO[2]	SC1_RES_US_H_RATIO[1]	SC1_RES_US_H_RATIO[0]	
	SC1_SCL0_US6	-	-	-	SC1_RES_US_H_INTE R POTYP	SC1_RES_US_HT_INI PHASE[11]	SC1_RES_US_HT_INI PHASE[10]	SC1_RES_US_HT_INI PHASE[9]	SC1_RES_US_HT_INI PHASE[8]
		SC1_RES_US_HT_INI PHASE[7]	SC1_RES_US_HT_INI PHASE[6]	SC1_RES_US_HT_INI PHASE[5]	SC1_RES_US_HT_INI PHASE[4]	SC1_RES_US_HT_INI PHASE[3]	SC1_RES_US_HT_INI PHASE[2]	SC1_RES_US_HT_INI PHASE[1]	SC1_RES_US_HT_INI PHASE[0]
		-	-	-	-	SC1_RES_US_HB_INI PHASE[11]	SC1_RES_US_HB_INI PHASE[10]	SC1_RES_US_HB_INI PHASE[9]	SC1_RES_US_HB_INI PHASE[8]
		SC1_RES_US_HB_INI PHASE[7]	SC1_RES_US_HB_INI PHASE[6]	SC1_RES_US_HB_INI PHASE[5]	SC1_RES_US_HB_INI PHASE[4]	SC1_RES_US_HB_INI PHASE[3]	SC1_RES_US_HB_INI PHASE[2]	SC1_RES_US_HB_INI PHASE[1]	SC1_RES_US_HB_INI PHASE[0]



Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Video display controller 5 channel 0	SC1_SCL0_US7	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		SC1_RES_HCUR[7]	SC1_RES_HCUR[6]	SC1_RES_HCUR[5]	SC1_RES_HCUR[4]	SC1_RES_HCUR[3]	SC1_RES_HCUR[2]	SC1_RES_HCUR[1]	SC1_RES_HCUR[0]
	SC1_SCL0_US8	SC1_RES_VCUR[7]	SC1_RES_VCUR[6]	SC1_RES_VCUR[5]	SC1_RES_VCUR[4]	SC1_RES_VCUR[3]	SC1_RES_VCUR[2]	SC1_RES_VCUR[1]	SC1_RES_VCUR[0]
		-	-	-	-	-	-	-	-
		-	-	-	SC1_RES_IBUS_SYN_C_SEL	-	-	-	SC1_RES_DISP_ON
	SC1_SCL0_OVR1	-	-	-	-	-	-	-	-
		SC1_RES_BK_COL_R[7]	SC1_RES_BK_COL_R[6]	SC1_RES_BK_COL_R[5]	SC1_RES_BK_COL_R[4]	SC1_RES_BK_COL_R[3]	SC1_RES_BK_COL_R[2]	SC1_RES_BK_COL_R[1]	SC1_RES_BK_COL_R[0]
		SC1_RES_BK_COL_G[7]	SC1_RES_BK_COL_G[6]	SC1_RES_BK_COL_G[5]	SC1_RES_BK_COL_G[4]	SC1_RES_BK_COL_G[3]	SC1_RES_BK_COL_G[2]	SC1_RES_BK_COL_G[1]	SC1_RES_BK_COL_G[0]
SC1_SCL1_UPDATE	SC1_RES_BK_COL_B[7]	SC1_RES_BK_COL_B[6]	SC1_RES_BK_COL_B[5]	SC1_RES_BK_COL_B[4]	SC1_RES_BK_COL_B[3]	SC1_RES_BK_COL_B[2]	SC1_RES_BK_COL_B[1]	SC1_RES_BK_COL_B[0]	
	-	-	-	-	-	-	-	-	
	-	-	-	SC1_SCL1_UPDATE_B	-	-	-	SC1_SCL1_UPDATE_A	
SC1_SCL1_WR1	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	
	-	-	-	SC1_SCL1_VEN_B	-	-	-	SC1_SCL1_VEN_A	
SC1_SCL1_WR2	-	-	-	-	-	-	-	-	
	SC1_RES_TB_ADD_MOD	SC1_RES_DS_WR_MD[2]	SC1_RES_DS_WR_MD[1]	SC1_RES_DS_WR_MD[0]	SC1_RES_MD[1]	SC1_RES_MD[0]	SC1_RES_LOOP	SC1_RES_BST_MD	
	SC1_RES_BASE[31]	SC1_RES_BASE[30]	SC1_RES_BASE[29]	SC1_RES_BASE[28]	SC1_RES_BASE[27]	SC1_RES_BASE[26]	SC1_RES_BASE[25]	SC1_RES_BASE[24]	
SC1_SCL1_WR3	SC1_RES_BASE[23]	SC1_RES_BASE[22]	SC1_RES_BASE[21]	SC1_RES_BASE[20]	SC1_RES_BASE[19]	SC1_RES_BASE[18]	SC1_RES_BASE[17]	SC1_RES_BASE[16]	
	SC1_RES_BASE[15]	SC1_RES_BASE[14]	SC1_RES_BASE[13]	SC1_RES_BASE[12]	SC1_RES_BASE[11]	SC1_RES_BASE[10]	SC1_RES_BASE[9]	SC1_RES_BASE[8]	
	SC1_RES_BASE[7]	SC1_RES_BASE[6]	SC1_RES_BASE[5]	SC1_RES_BASE[4]	SC1_RES_BASE[3]	SC1_RES_BASE[2]	SC1_RES_BASE[1]	SC1_RES_BASE[0]	
SC1_SCL1_WR4	-	SC1_RES_LN_OFF[14]	SC1_RES_LN_OFF[13]	SC1_RES_LN_OFF[12]	SC1_RES_LN_OFF[11]	SC1_RES_LN_OFF[10]	SC1_RES_LN_OFF[9]	SC1_RES_LN_OFF[8]	
	SC1_RES_LN_OFF[7]	SC1_RES_LN_OFF[6]	SC1_RES_LN_OFF[5]	SC1_RES_LN_OFF[4]	SC1_RES_LN_OFF[3]	SC1_RES_LN_OFF[2]	SC1_RES_LN_OFF[1]	SC1_RES_LN_OFF[0]	
	-	-	-	-	-	-	SC1_RES_FLM_NUM[9]	SC1_RES_FLM_NUM[8]	
SC1_SCL1_WR5	SC1_RES_FLM_NUM[7]	SC1_RES_FLM_NUM[6]	SC1_RES_FLM_NUM[5]	SC1_RES_FLM_NUM[4]	SC1_RES_FLM_NUM[3]	SC1_RES_FLM_NUM[2]	SC1_RES_FLM_NUM[1]	SC1_RES_FLM_NUM[0]	
	-	-	-	-	-	-	-	-	
	-	SC1_RES_FLM_OFF[22]	SC1_RES_FLM_OFF[21]	SC1_RES_FLM_OFF[20]	SC1_RES_FLM_OFF[19]	SC1_RES_FLM_OFF[18]	SC1_RES_FLM_OFF[17]	SC1_RES_FLM_OFF[16]	
SC1_SCL1_WR6	SC1_RES_FLM_OFF[15]	SC1_RES_FLM_OFF[14]	SC1_RES_FLM_OFF[13]	SC1_RES_FLM_OFF[12]	SC1_RES_FLM_OFF[11]	SC1_RES_FLM_OFF[10]	SC1_RES_FLM_OFF[9]	SC1_RES_FLM_OFF[8]	
	SC1_RES_FLM_OFF[7]	SC1_RES_FLM_OFF[6]	SC1_RES_FLM_OFF[5]	SC1_RES_FLM_OFF[4]	SC1_RES_FLM_OFF[3]	SC1_RES_FLM_OFF[2]	SC1_RES_FLM_OFF[1]	SC1_RES_FLM_OFF[0]	
	-	-	-	SC1_RES_INTER	-	-	SC1_RES_FS_RATE[1]	SC1_RES_FS_RATE[0]	
SC1_SCL1_WR7	-	-	-	SC1_RES_FLD_SEL	-	-	-	SC1_RES_WENB	
	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	SC1_RES_DTH_ON	
SC1_SCL1_WR8	-	-	-	-	-	-	-	SC1_RES_BITDEC_ON	
	-	-	-	-	-	-	-	SC1_RES_OVERFLOW	
	SC1_RES_FLM_CNT[7]	SC1_RES_FLM_CNT[6]	SC1_RES_FLM_CNT[5]	SC1_RES_FLM_CNT[4]	SC1_RES_FLM_CNT[3]	SC1_RES_FLM_CNT[2]	SC1_RES_FLM_CNT[1]	SC1_RES_FLM_CNT[0]	

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Video display controller 5 channel 0	SC1_SCL1_WR8	SC1_RES_BASE_B[31]	SC1_RES_BASE_B[30]	SC1_RES_BASE_B[29]	SC1_RES_BASE_B[28]	SC1_RES_BASE_B[27]	SC1_RES_BASE_B[26]	SC1_RES_BASE_B[25]	SC1_RES_BASE_B[24]
		SC1_RES_BASE_B[23]	SC1_RES_BASE_B[22]	SC1_RES_BASE_B[21]	SC1_RES_BASE_B[20]	SC1_RES_BASE_B[19]	SC1_RES_BASE_B[18]	SC1_RES_BASE_B[17]	SC1_RES_BASE_B[16]
		SC1_RES_BASE_B[15]	SC1_RES_BASE_B[14]	SC1_RES_BASE_B[13]	SC1_RES_BASE_B[12]	SC1_RES_BASE_B[11]	SC1_RES_BASE_B[10]	SC1_RES_BASE_B[9]	SC1_RES_BASE_B[8]
		SC1_RES_BASE_B[7]	SC1_RES_BASE_B[6]	SC1_RES_BASE_B[5]	SC1_RES_BASE_B[4]	SC1_RES_BASE_B[3]	SC1_RES_BASE_B[2]	SC1_RES_BASE_B[1]	SC1_RES_BASE_B[0]
	SC1_SCL1_WR9	-	SC1_RES_LN_OFF_B[14]	SC1_RES_LN_OFF_B[13]	SC1_RES_LN_OFF_B[12]	SC1_RES_LN_OFF_B[11]	SC1_RES_LN_OFF_B[10]	SC1_RES_LN_OFF_B[9]	SC1_RES_LN_OFF_B[8]
		SC1_RES_LN_OFF_B[7]	SC1_RES_LN_OFF_B[6]	SC1_RES_LN_OFF_B[5]	SC1_RES_LN_OFF_B[4]	SC1_RES_LN_OFF_B[3]	SC1_RES_LN_OFF_B[2]	SC1_RES_LN_OFF_B[1]	SC1_RES_LN_OFF_B[0]
		-	-	-	-	-	-	SC1_RES_FLM_NUM_B[9]	SC1_RES_FLM_NUM_B[8]
		SC1_RES_FLM_NUM_B[7]	SC1_RES_FLM_NUM_B[6]	SC1_RES_FLM_NUM_B[5]	SC1_RES_FLM_NUM_B[4]	SC1_RES_FLM_NUM_B[3]	SC1_RES_FLM_NUM_B[2]	SC1_RES_FLM_NUM_B[1]	SC1_RES_FLM_NUM_B[0]
	SC1_SCL1_WR10	-	-	-	-	-	-	-	-
		-	SC1_RES_FLM_OFF_B[22]	SC1_RES_FLM_OFF_B[21]	SC1_RES_FLM_OFF_B[20]	SC1_RES_FLM_OFF_B[19]	SC1_RES_FLM_OFF_B[18]	SC1_RES_FLM_OFF_B[17]	SC1_RES_FLM_OFF_B[16]
		SC1_RES_FLM_OFF_B[15]	SC1_RES_FLM_OFF_B[14]	SC1_RES_FLM_OFF_B[13]	SC1_RES_FLM_OFF_B[12]	SC1_RES_FLM_OFF_B[11]	SC1_RES_FLM_OFF_B[10]	SC1_RES_FLM_OFF_B[9]	SC1_RES_FLM_OFF_B[8]
	SC1_SCL1_WR11	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		SC1_RES_FLM_CNT_B[7]	SC1_RES_FLM_CNT_B[6]	SC1_RES_FLM_CNT_B[5]	SC1_RES_FLM_CNT_B[4]	SC1_RES_FLM_CNT_B[3]	SC1_RES_FLM_CNT_B[2]	SC1_RES_FLM_CNT_B[1]	SC1_RES_FLM_CNT_B[0]
	SC1_SCL1_MON1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	SC1_PBUF_NUM[1]	SC1_PBUF_NUM[0]
		-	-	-	-	-	-	-	-
	SC1_SCL1_PBUF0	SC1_PBUF0_ADD[31]	SC1_PBUF0_ADD[30]	SC1_PBUF0_ADD[29]	SC1_PBUF0_ADD[28]	SC1_PBUF0_ADD[27]	SC1_PBUF0_ADD[26]	SC1_PBUF0_ADD[25]	SC1_PBUF0_ADD[24]
		SC1_PBUF0_ADD[23]	SC1_PBUF0_ADD[22]	SC1_PBUF0_ADD[21]	SC1_PBUF0_ADD[20]	SC1_PBUF0_ADD[19]	SC1_PBUF0_ADD[18]	SC1_PBUF0_ADD[17]	SC1_PBUF0_ADD[16]
		SC1_PBUF0_ADD[15]	SC1_PBUF0_ADD[14]	SC1_PBUF0_ADD[13]	SC1_PBUF0_ADD[12]	SC1_PBUF0_ADD[11]	SC1_PBUF0_ADD[10]	SC1_PBUF0_ADD[9]	SC1_PBUF0_ADD[8]
		SC1_PBUF0_ADD[7]	SC1_PBUF0_ADD[6]	SC1_PBUF0_ADD[5]	SC1_PBUF0_ADD[4]	SC1_PBUF0_ADD[3]	SC1_PBUF0_ADD[2]	SC1_PBUF0_ADD[1]	SC1_PBUF0_ADD[0]
	SC1_SCL1_PBUF1	SC1_PBUF1_ADD[31]	SC1_PBUF1_ADD[30]	SC1_PBUF1_ADD[29]	SC1_PBUF1_ADD[28]	SC1_PBUF1_ADD[27]	SC1_PBUF1_ADD[26]	SC1_PBUF1_ADD[25]	SC1_PBUF1_ADD[24]
		SC1_PBUF1_ADD[23]	SC1_PBUF1_ADD[22]	SC1_PBUF1_ADD[21]	SC1_PBUF1_ADD[20]	SC1_PBUF1_ADD[19]	SC1_PBUF1_ADD[18]	SC1_PBUF1_ADD[17]	SC1_PBUF1_ADD[16]
		SC1_PBUF1_ADD[15]	SC1_PBUF1_ADD[14]	SC1_PBUF1_ADD[13]	SC1_PBUF1_ADD[12]	SC1_PBUF1_ADD[11]	SC1_PBUF1_ADD[10]	SC1_PBUF1_ADD[9]	SC1_PBUF1_ADD[8]
		SC1_PBUF1_ADD[7]	SC1_PBUF1_ADD[6]	SC1_PBUF1_ADD[5]	SC1_PBUF1_ADD[4]	SC1_PBUF1_ADD[3]	SC1_PBUF1_ADD[2]	SC1_PBUF1_ADD[1]	SC1_PBUF1_ADD[0]
	SC1_SCL1_PBUF2	SC1_PBUF2_ADD[31]	SC1_PBUF2_ADD[30]	SC1_PBUF2_ADD[29]	SC1_PBUF2_ADD[28]	SC1_PBUF2_ADD[27]	SC1_PBUF2_ADD[26]	SC1_PBUF2_ADD[25]	SC1_PBUF2_ADD[24]
		SC1_PBUF2_ADD[23]	SC1_PBUF2_ADD[22]	SC1_PBUF2_ADD[21]	SC1_PBUF2_ADD[20]	SC1_PBUF2_ADD[19]	SC1_PBUF2_ADD[18]	SC1_PBUF2_ADD[17]	SC1_PBUF2_ADD[16]
		SC1_PBUF2_ADD[15]	SC1_PBUF2_ADD[14]	SC1_PBUF2_ADD[13]	SC1_PBUF2_ADD[12]	SC1_PBUF2_ADD[11]	SC1_PBUF2_ADD[10]	SC1_PBUF2_ADD[9]	SC1_PBUF2_ADD[8]
		SC1_PBUF2_ADD[7]	SC1_PBUF2_ADD[6]	SC1_PBUF2_ADD[5]	SC1_PBUF2_ADD[4]	SC1_PBUF2_ADD[3]	SC1_PBUF2_ADD[2]	SC1_PBUF2_ADD[1]	SC1_PBUF2_ADD[0]
	SC1_SCL1_PBUF3	SC1_PBUF3_ADD[31]	SC1_PBUF3_ADD[30]	SC1_PBUF3_ADD[29]	SC1_PBUF3_ADD[28]	SC1_PBUF3_ADD[27]	SC1_PBUF3_ADD[26]	SC1_PBUF3_ADD[25]	SC1_PBUF3_ADD[24]
		SC1_PBUF3_ADD[23]	SC1_PBUF3_ADD[22]	SC1_PBUF3_ADD[21]	SC1_PBUF3_ADD[20]	SC1_PBUF3_ADD[19]	SC1_PBUF3_ADD[18]	SC1_PBUF3_ADD[17]	SC1_PBUF3_ADD[16]
		SC1_PBUF3_ADD[15]	SC1_PBUF3_ADD[14]	SC1_PBUF3_ADD[13]	SC1_PBUF3_ADD[12]	SC1_PBUF3_ADD[11]	SC1_PBUF3_ADD[10]	SC1_PBUF3_ADD[9]	SC1_PBUF3_ADD[8]
		SC1_PBUF3_ADD[7]	SC1_PBUF3_ADD[6]	SC1_PBUF3_ADD[5]	SC1_PBUF3_ADD[4]	SC1_PBUF3_ADD[3]	SC1_PBUF3_ADD[2]	SC1_PBUF3_ADD[1]	SC1_PBUF3_ADD[0]
	SC1_SCL1_PBUF_FLD	-	-	-	-	-	-	-	SC1_FLD_INF3
		-	-	-	-	-	-	-	SC1_FLD_INF2
		-	-	-	-	-	-	-	SC1_FLD_INF1
		-	-	-	-	-	-	-	SC1_FLD_INF0

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 0	SC1_SCL1_PBUF_CNT	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	SC1_PBUF_RST	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
	GR1_UPDATE	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	GR1_UPDATE
		-	-	-	-	GR1_P_VEN	-	-	-	GR1_IBUS_VEN
	GR1_FLM_RD	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	GR1_R_ENB
	GR1_FLM1	GR1_FLD_SEL	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	GR1_LN_OFF_DIR
		-	-	-	-	-	-	-	GR1_FLM_SEL[1]	GR1_FLM_SEL[0]
		-	-	-	-	GR1_IMR_FLM_INV	-	-	-	GR1_BST_MD
	GR1_FLM2	GR1_BASE[31]	GR1_BASE[30]	GR1_BASE[29]	GR1_BASE[28]	GR1_BASE[27]	GR1_BASE[26]	GR1_BASE[25]	GR1_BASE[24]	
		GR1_BASE[23]	GR1_BASE[22]	GR1_BASE[21]	GR1_BASE[20]	GR1_BASE[19]	GR1_BASE[18]	GR1_BASE[17]	GR1_BASE[16]	
		GR1_BASE[15]	GR1_BASE[14]	GR1_BASE[13]	GR1_BASE[12]	GR1_BASE[11]	GR1_BASE[10]	GR1_BASE[9]	GR1_BASE[8]	
		GR1_BASE[7]	GR1_BASE[6]	GR1_BASE[5]	GR1_BASE[4]	GR1_BASE[3]	GR1_BASE[2]	GR1_BASE[1]	GR1_BASE[0]	
	GR1_FLM3	GR1_FLD_NXT	GR1_LN_OFF[14]	GR1_LN_OFF[13]	GR1_LN_OFF[12]	GR1_LN_OFF[11]	GR1_LN_OFF[10]	GR1_LN_OFF[9]	GR1_LN_OFF[8]	
		GR1_LN_OFF[7]	GR1_LN_OFF[6]	GR1_LN_OFF[5]	GR1_LN_OFF[4]	GR1_LN_OFF[3]	GR1_LN_OFF[2]	GR1_LN_OFF[1]	GR1_LN_OFF[0]	
		-	-	-	-	-	-	GR1_FLM_NUM[9]	GR1_FLM_NUM[8]	
		GR1_FLM_NUM[7]	GR1_FLM_NUM[6]	GR1_FLM_NUM[5]	GR1_FLM_NUM[4]	GR1_FLM_NUM[3]	GR1_FLM_NUM[2]	GR1_FLM_NUM[1]	GR1_FLM_NUM[0]	
	GR1_FLM4	-	-	-	-	-	-	-	-	
		-	GR1_FLM_OFF[22]	GR1_FLM_OFF[21]	GR1_FLM_OFF[20]	GR1_FLM_OFF[19]	GR1_FLM_OFF[18]	GR1_FLM_OFF[17]	GR1_FLM_OFF[16]	
		GR1_FLM_OFF[15]	GR1_FLM_OFF[14]	GR1_FLM_OFF[13]	GR1_FLM_OFF[12]	GR1_FLM_OFF[11]	GR1_FLM_OFF[10]	GR1_FLM_OFF[9]	GR1_FLM_OFF[8]	
		GR1_FLM_OFF[7]	GR1_FLM_OFF[6]	GR1_FLM_OFF[5]	GR1_FLM_OFF[4]	GR1_FLM_OFF[3]	GR1_FLM_OFF[2]	GR1_FLM_OFF[1]	GR1_FLM_OFF[0]	
	GR1_FLM5	-	-	-	-	-	GR1_FLM_LNUM[10]	GR1_FLM_LNUM[9]	GR1_FLM_LNUM[8]	
		GR1_FLM_LNUM[7]	GR1_FLM_LNUM[6]	GR1_FLM_LNUM[5]	GR1_FLM_LNUM[4]	GR1_FLM_LNUM[3]	GR1_FLM_LNUM[2]	GR1_FLM_LNUM[1]	GR1_FLM_LNUM[0]	
		-	-	-	-	-	GR1_FLM_LOOP[10]	GR1_FLM_LOOP[9]	GR1_FLM_LOOP[8]	
		GR1_FLM_LOOP[7]	GR1_FLM_LOOP[6]	GR1_FLM_LOOP[5]	GR1_FLM_LOOP[4]	GR1_FLM_LOOP[3]	GR1_FLM_LOOP[2]	GR1_FLM_LOOP[1]	GR1_FLM_LOOP[0]	
	GR1_FLM6	GR1_FORMAT[3]	GR1_FORMAT[2]	GR1_FORMAT[1]	GR1_FORMAT[0]	-	GR1_HW[10]	GR1_HW[9]	GR1_HW[8]	
		GR1_HW[7]	GR1_HW[6]	GR1_HW[5]	GR1_HW[4]	GR1_HW[3]	GR1_HW[2]	GR1_HW[1]	GR1_HW[0]	
		GR1_YCC_SWAP[2]	GR1_YCC_SWAP[1]	GR1_YCC_SWAP[0]	GR1_RDSWA[2]	GR1_RDSWA[1]	GR1_RDSWA[0]	-	GR1_CNV444_MD	
		-	-	GR1_STA_POS[5]	GR1_STA_POS[4]	GR1_STA_POS[3]	GR1_STA_POS[2]	GR1_STA_POS[1]	GR1_STA_POS[0]	
	GR1_AB1	-	-	GR1_CUS_CON_ON	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		GR1_ARC_MUL	GR1_ACALC_MD	-	GR1_ARC_ON	-	-	-	GR1_ARC_DISP_ON	
		-	-	-	GR1_GRC_DISP_ON	-	-	GR1_DISP_SEL[1]	GR1_DISP_SEL[0]	
	GR1_AB2	-	-	-	-	-	GR1_GRC_VS[10]	GR1_GRC_VS[9]	GR1_GRC_VS[8]	
		GR1_GRC_VS[7]	GR1_GRC_VS[6]	GR1_GRC_VS[5]	GR1_GRC_VS[4]	GR1_GRC_VS[3]	GR1_GRC_VS[2]	GR1_GRC_VS[1]	GR1_GRC_VS[0]	
		-	-	-	-	-	GR1_GRC_VW[10]	GR1_GRC_VW[9]	GR1_GRC_VW[8]	
		GR1_GRC_VW[7]	GR1_GRC_VW[6]	GR1_GRC_VW[5]	GR1_GRC_VW[4]	GR1_GRC_VW[3]	GR1_GRC_VW[2]	GR1_GRC_VW[1]	GR1_GRC_VW[0]	
	GR1_AB3	-	-	-	-	-	GR1_GRC_HS[10]	GR1_GRC_HS[9]	GR1_GRC_HS[8]	
		GR1_GRC_HS[7]	GR1_GRC_HS[6]	GR1_GRC_HS[5]	GR1_GRC_HS[4]	GR1_GRC_HS[3]	GR1_GRC_HS[2]	GR1_GRC_HS[1]	GR1_GRC_HS[0]	
		-	-	-	-	-	GR1_GRC_HW[10]	GR1_GRC_HW[9]	GR1_GRC_HW[8]	
		GR1_GRC_HW[7]	GR1_GRC_HW[6]	GR1_GRC_HW[5]	GR1_GRC_HW[4]	GR1_GRC_HW[3]	GR1_GRC_HW[2]	GR1_GRC_HW[1]	GR1_GRC_HW[0]	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Video display controller 5 channel 0	GR1_AB4	-	-	-	-	-	GR1_ARC_VS[10]	GR1_ARC_VS[9]	GR1_ARC_VS[0]
		GR1_ARC_VS[7]	GR1_ARC_VS[6]	GR1_ARC_VS[5]	GR1_ARC_VS[4]	GR1_ARC_VS[3]	GR1_ARC_VS[2]	GR1_ARC_VS[1]	GR1_ARC_VS[0]
		-	-	-	-	-	GR1_ARC_VW[10]	GR1_ARC_VW[9]	GR1_ARC_VW[8]
		GR1_ARC_VW[7]	GR1_ARC_VW[6]	GR1_ARC_VW[5]	GR1_ARC_VW[4]	GR1_ARC_VW[3]	GR1_ARC_VW[2]	GR1_ARC_VW[1]	GR1_ARC_VW[0]
	GR1_AB5	-	-	-	-	-	GR1_ARC_HS[10]	GR1_ARC_HS[9]	GR1_ARC_HS[8]
		GR1_ARC_HS[7]	GR1_ARC_HS[6]	GR1_ARC_HS[5]	GR1_ARC_HS[4]	GR1_ARC_HS[3]	GR1_ARC_HS[2]	GR1_ARC_HS[1]	GR1_ARC_HS[0]
		-	-	-	-	-	GR1_ARC_HW[10]	GR1_ARC_HW[9]	GR1_ARC_HW[8]
		GR1_ARC_HW[7]	GR1_ARC_HW[6]	GR1_ARC_HW[5]	GR1_ARC_HW[4]	GR1_ARC_HW[3]	GR1_ARC_HW[2]	GR1_ARC_HW[1]	GR1_ARC_HW[0]
	GR1_AB6	-	-	-	-	-	-	-	GR1_ARC_MODE
		GR1_ARC_COEF[7]	GR1_ARC_COEF[6]	GR1_ARC_COEF[5]	GR1_ARC_COEF[4]	GR1_ARC_COEF[3]	GR1_ARC_COEF[2]	GR1_ARC_COEF[1]	GR1_ARC_COEF[0]
		-	-	-	-	-	-	-	-
		GR1_ARC_RATE [7]	GR1_ARC_RATE [6]	GR1_ARC_RATE [5]	GR1_ARC_RATE [4]	GR1_ARC_RATE [3]	GR1_ARC_RATE [2]	GR1_ARC_RATE [1]	GR1_ARC_RATE [0]
	GR1_AB7	-	-	-	-	-	-	-	-
		GR1_ARC_DEF [7]	GR1_ARC_DEF [6]	GR1_ARC_DEF [5]	GR1_ARC_DEF [4]	GR1_ARC_DEF [3]	GR1_ARC_DEF [2]	GR1_ARC_DEF [1]	GR1_ARC_DEF [0]
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GR1_CK_ON
	GR1_AB8	GR1_CK_KCLUT [7]	GR1_CK_KCLUT [6]	GR1_CK_KCLUT [5]	GR1_CK_KCLUT [4]	GR1_CK_KCLUT [3]	GR1_CK_KCLUT [2]	GR1_CK_KCLUT [1]	GR1_CK_KCLUT [0]
		GR1_CK_KG[7]	GR1_CK_KG[6]	GR1_CK_KG[5]	GR1_CK_KG[4]	GR1_CK_KG[3]	GR1_CK_KG[2]	GR1_CK_KG[1]	GR1_CK_KG[0]
		GR1_CK_KB[7]	GR1_CK_KB[6]	GR1_CK_KB[5]	GR1_CK_KB[4]	GR1_CK_KB[3]	GR1_CK_KB[2]	GR1_CK_KB[1]	GR1_CK_KB[0]
		GR1_CK_KR[7]	GR1_CK_KR[6]	GR1_CK_KR[5]	GR1_CK_KR[4]	GR1_CK_KR[3]	GR1_CK_KR[2]	GR1_CK_KR[1]	GR1_CK_KR[0]
	GR1_AB9	GR1_CK_A[7]	GR1_CK_A[6]	GR1_CK_A[5]	GR1_CK_A[4]	GR1_CK_A[3]	GR1_CK_A[2]	GR1_CK_A[1]	GR1_CK_A[0]
		GR1_CK_G[7]	GR1_CK_G[6]	GR1_CK_G[5]	GR1_CK_G[4]	GR1_CK_G[3]	GR1_CK_G[2]	GR1_CK_G[1]	GR1_CK_G[0]
		GR1_CK_B[7]	GR1_CK_B[6]	GR1_CK_B[5]	GR1_CK_B[4]	GR1_CK_B[3]	GR1_CK_B[2]	GR1_CK_B[1]	GR1_CK_B[0]
		GR1_CK_R[7]	GR1_CK_R[6]	GR1_CK_R[5]	GR1_CK_R[4]	GR1_CK_R[3]	GR1_CK_R[2]	GR1_CK_R[1]	GR1_CK_R[0]
	GR1_AB10	GR1_A0[7]	GR1_A0[6]	GR1_A0[5]	GR1_A0[4]	GR1_A0[3]	GR1_A0[2]	GR1_A0[1]	GR1_A0[0]
		GR1_G0[7]	GR1_G0[6]	GR1_G0[5]	GR1_G0[4]	GR1_G0[3]	GR1_G0[2]	GR1_G0[1]	GR1_G0[0]
		GR1_B0[7]	GR1_B0[6]	GR1_B0[5]	GR1_B0[4]	GR1_B0[3]	GR1_B0[2]	GR1_B0[1]	GR1_B0[0]
		GR1_R0[7]	GR1_R0[6]	GR1_R0[5]	GR1_R0[4]	GR1_R0[3]	GR1_R0[2]	GR1_R0[1]	GR1_R0[0]
	GR1_AB11	GR1_A1[7]	GR1_A1[6]	GR1_A1[5]	GR1_A1[4]	GR1_A1[3]	GR1_A1[2]	GR1_A1[1]	GR1_A1[0]
		GR1_G1[7]	GR1_G1[6]	GR1_G1[5]	GR1_G1[4]	GR1_G1[3]	GR1_G1[2]	GR1_G1[1]	GR1_G1[0]
		GR1_B1[7]	GR1_B1[6]	GR1_B1[5]	GR1_B1[4]	GR1_B1[3]	GR1_B1[2]	GR1_B1[1]	GR1_B1[0]
		GR1_R1[7]	GR1_R1[6]	GR1_R1[5]	GR1_R1[4]	GR1_R1[3]	GR1_R1[2]	GR1_R1[1]	GR1_R1[0]
	GR1_BASE	-	-	-	-	-	-	-	-
		GR1_BASE_G[7]	GR1_BASE_G[6]	GR1_BASE_G[5]	GR1_BASE_G[4]	GR1_BASE_G[3]	GR1_BASE_G[2]	GR1_BASE_G[1]	GR1_BASE_G[0]
		GR1_BASE_B[7]	GR1_BASE_B[6]	GR1_BASE_B[5]	GR1_BASE_B[4]	GR1_BASE_B[3]	GR1_BASE_B[2]	GR1_BASE_B[1]	GR1_BASE_B[0]
	GR1_CLUT	GR1_BASE_R[7]	GR1_BASE_R[6]	GR1_BASE_R[5]	GR1_BASE_R[4]	GR1_BASE_R[3]	GR1_BASE_R[2]	GR1_BASE_R[1]	GR1_BASE_R[0]
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GR1_CLT_SEL
	GR1_MON	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GR1_ARC_ST
	ADJ0_UPDATE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	ADJ0_VEN
	ADJ0_BKSTR_SET	-	-	-	-	-	-	-	BKSTR_ON
		BKSTR_ST[3]	BKSTR_ST[2]	BKSTR_ST[1]	BKSTR_ST[0]	BKSTR_D[3]	BKSTR_D[2]	BKSTR_D[1]	BKSTR_D[0]
		-	-	-	BKSTR_T1[4]	BKSTR_T1[3]	BKSTR_T1[2]	BKSTR_T1[1]	BKSTR_T1[0]
		-	-	-	BKSTR_T2[4]	BKSTR_T2[3]	BKSTR_T2[2]	BKSTR_T2[1]	BKSTR_T2[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 0	ADJ0_ENH_TIM1	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	ENH_MD	-	-	-	ENH_DISP_ON	
	ADJ0_ENH_TIM2	-	-	-	-	-	-	ENH_VS[10]	ENH_VS[9]	ENH_VS[8]
		ENH_VS[7]	ENH_VS[6]	ENH_VS[5]	ENH_VS[4]	ENH_VS[3]	ENH_VS[2]	ENH_VS[1]	ENH_VS[0]	
		-	-	-	-	-	ENH_VW[10]	ENH_VW[9]	ENH_VW[8]	
		ENH_VW[7]	ENH_VW[6]	ENH_VW[5]	ENH_VW[4]	ENH_VW[3]	ENH_VW[2]	ENH_VW[1]	ENH_VW[0]	
	ADJ0_ENH_TIM3	-	-	-	-	-	-	ENH_HS[10]	ENH_HS[9]	ENH_HS[8]
		ENH_HS[7]	ENH_HS[6]	ENH_HS[5]	ENH_HS[4]	ENH_HS[3]	ENH_HS[2]	ENH_HS[1]	ENH_HS[0]	
		-	-	-	-	-	ENH_HW[10]	ENH_HW[9]	ENH_HW[8]	
		ENH_HW[7]	ENH_HW[6]	ENH_HW[5]	ENH_HW[4]	ENH_HW[3]	ENH_HW[2]	ENH_HW[1]	ENH_HW[0]	
	ADJ0_ENH_SHP1	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	SHP_H_ON	
		-	-	-	-	-	-	-	-	
		-	SHP_H1_CORE[6]	SHP_H1_CORE[5]	SHP_H1_CORE[4]	SHP_H1_CORE[3]	SHP_H1_CORE[2]	SHP_H1_CORE[1]	SHP_H1_CORE[0]	
	ADJ0_ENH_SHP2	SHP_H1_CLIP_O[7]	SHP_H1_CLIP_O[6]	SHP_H1_CLIP_O[5]	SHP_H1_CLIP_O[4]	SHP_H1_CLIP_O[3]	SHP_H1_CLIP_O[2]	SHP_H1_CLIP_O[1]	SHP_H1_CLIP_O[0]	
		SHP_H1_CLIP_U[7]	SHP_H1_CLIP_U[6]	SHP_H1_CLIP_U[5]	SHP_H1_CLIP_U[4]	SHP_H1_CLIP_U[3]	SHP_H1_CLIP_U[2]	SHP_H1_CLIP_U[1]	SHP_H1_CLIP_U[0]	
		SHP_H1_GAIN_O[7]	SHP_H1_GAIN_O[6]	SHP_H1_GAIN_O[5]	SHP_H1_GAIN_O[4]	SHP_H1_GAIN_O[3]	SHP_H1_GAIN_O[2]	SHP_H1_GAIN_O[1]	SHP_H1_GAIN_O[0]	
		SHP_H1_GAIN_U[7]	SHP_H1_GAIN_U[6]	SHP_H1_GAIN_U[5]	SHP_H1_GAIN_U[4]	SHP_H1_GAIN_U[3]	SHP_H1_GAIN_U[2]	SHP_H1_GAIN_U[1]	SHP_H1_GAIN_U[0]	
	ADJ0_ENH_SHP3	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	SHP_H2_LPF_SEL	
		-	-	-	-	-	-	-	-	
		-	SHP_H2_CORE[6]	SHP_H2_CORE[5]	SHP_H2_CORE[4]	SHP_H2_CORE[3]	SHP_H2_CORE[2]	SHP_H2_CORE[1]	SHP_H2_CORE[0]	
	ADJ0_ENH_SHP4	SHP_H2_CLIP_O[7]	SHP_H2_CLIP_O[6]	SHP_H2_CLIP_O[5]	SHP_H2_CLIP_O[4]	SHP_H2_CLIP_O[3]	SHP_H2_CLIP_O[2]	SHP_H2_CLIP_O[1]	SHP_H2_CLIP_O[0]	
		SHP_H2_CLIP_U[7]	SHP_H2_CLIP_U[6]	SHP_H2_CLIP_U[5]	SHP_H2_CLIP_U[4]	SHP_H2_CLIP_U[3]	SHP_H2_CLIP_U[2]	SHP_H2_CLIP_U[1]	SHP_H2_CLIP_U[0]	
		SHP_H2_GAIN_O[7]	SHP_H2_GAIN_O[6]	SHP_H2_GAIN_O[5]	SHP_H2_GAIN_O[4]	SHP_H2_GAIN_O[3]	SHP_H2_GAIN_O[2]	SHP_H2_GAIN_O[1]	SHP_H2_GAIN_O[0]	
		SHP_H2_GAIN_U[7]	SHP_H2_GAIN_U[6]	SHP_H2_GAIN_U[5]	SHP_H2_GAIN_U[4]	SHP_H2_GAIN_U[3]	SHP_H2_GAIN_U[2]	SHP_H2_GAIN_U[1]	SHP_H2_GAIN_U[0]	
	ADJ0_ENH_SHP5	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	SHP_H3_CORE[6]	SHP_H3_CORE[5]	SHP_H3_CORE[4]	SHP_H3_CORE[3]	SHP_H3_CORE[2]	SHP_H3_CORE[1]	SHP_H3_CORE[0]	
	ADJ0_ENH_SHP6	SHP_H3_CLIP_O[7]	SHP_H3_CLIP_O[6]	SHP_H3_CLIP_O[5]	SHP_H3_CLIP_O[4]	SHP_H3_CLIP_O[3]	SHP_H3_CLIP_O[2]	SHP_H3_CLIP_O[1]	SHP_H3_CLIP_O[0]	
		SHP_H3_CLIP_U[7]	SHP_H3_CLIP_U[6]	SHP_H3_CLIP_U[5]	SHP_H3_CLIP_U[4]	SHP_H3_CLIP_U[3]	SHP_H3_CLIP_U[2]	SHP_H3_CLIP_U[1]	SHP_H3_CLIP_U[0]	
		SHP_H3_GAIN_O[7]	SHP_H3_GAIN_O[6]	SHP_H3_GAIN_O[5]	SHP_H3_GAIN_O[4]	SHP_H3_GAIN_O[3]	SHP_H3_GAIN_O[2]	SHP_H3_GAIN_O[1]	SHP_H3_GAIN_O[0]	
		SHP_H3_GAIN_U[7]	SHP_H3_GAIN_U[6]	SHP_H3_GAIN_U[5]	SHP_H3_GAIN_U[4]	SHP_H3_GAIN_U[3]	SHP_H3_GAIN_U[2]	SHP_H3_GAIN_U[1]	SHP_H3_GAIN_U[0]	
	ADJ0_ENH_LTI1	LTI_H_ON	-	-	-	-	-	-	LTI_H2_LPF_SEL	
		LTI_H2_INC_ZERO[7]	LTI_H2_INC_ZERO[6]	LTI_H2_INC_ZERO[5]	LTI_H2_INC_ZERO[4]	LTI_H2_INC_ZERO[3]	LTI_H2_INC_ZERO[2]	LTI_H2_INC_ZERO[1]	LTI_H2_INC_ZERO[0]	
		LTI_H2_GAIN[7]	LTI_H2_GAIN[6]	LTI_H2_GAIN[5]	LTI_H2_GAIN[4]	LTI_H2_GAIN[3]	LTI_H2_GAIN[2]	LTI_H2_GAIN[1]	LTI_H2_GAIN[0]	
		LTI_H2_CORE[7]	LTI_H2_CORE[6]	LTI_H2_CORE[5]	LTI_H2_CORE[4]	LTI_H2_CORE[3]	LTI_H2_CORE[2]	LTI_H2_CORE[1]	LTI_H2_CORE[0]	
	ADJ0_ENH_LTI2	-	-	-	-	-	-	-	LTI_H4_MEDIAN_TAP_SEL	
		LTI_H4_INC_ZERO[7]	LTI_H4_INC_ZERO[6]	LTI_H4_INC_ZERO[5]	LTI_H4_INC_ZERO[4]	LTI_H4_INC_ZERO[3]	LTI_H4_INC_ZERO[2]	LTI_H4_INC_ZERO[1]	LTI_H4_INC_ZERO[0]	
		LTI_H4_GAIN[7]	LTI_H4_GAIN[6]	LTI_H4_GAIN[5]	LTI_H4_GAIN[4]	LTI_H4_GAIN[3]	LTI_H4_GAIN[2]	LTI_H4_GAIN[1]	LTI_H4_GAIN[0]	
		LTI_H4_CORE[7]	LTI_H4_CORE[6]	LTI_H4_CORE[5]	LTI_H4_CORE[4]	LTI_H4_CORE[3]	LTI_H4_CORE[2]	LTI_H4_CORE[1]	LTI_H4_CORE[0]	
	ADJ0_MTX_MODE	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	ADJ0_MTX_MD[1]	ADJ0_MTX_MD[0]	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 0	ADJ0_MTX_YG_ADJ0	-	-	-	-	-	-	-	-	
		ADJ0_MTX_YG[7]	ADJ0_MTX_YG[6]	ADJ0_MTX_YG[5]	ADJ0_MTX_YG[4]	ADJ0_MTX_YG[3]	ADJ0_MTX_YG[2]	ADJ0_MTX_YG[1]	ADJ0_MTX_YG[0]	
		-	-	-	-	-	ADJ0_MTX_GG[10]	ADJ0_MTX_GG[9]	ADJ0_MTX_GG[8]	
		ADJ0_MTX_GG[7]	ADJ0_MTX_GG[6]	ADJ0_MTX_GG[5]	ADJ0_MTX_GG[4]	ADJ0_MTX_GG[3]	ADJ0_MTX_GG[2]	ADJ0_MTX_GG[1]	ADJ0_MTX_GG[0]	
	ADJ0_MTX_YG_ADJ1	-	-	-	-	-	-	ADJ0_MTX_GB[10]	ADJ0_MTX_GB[9]	ADJ0_MTX_GB[8]
		ADJ0_MTX_GB[7]	ADJ0_MTX_GB[6]	ADJ0_MTX_GB[5]	ADJ0_MTX_GB[4]	ADJ0_MTX_GB[3]	ADJ0_MTX_GB[2]	ADJ0_MTX_GB[1]	ADJ0_MTX_GB[0]	
		-	-	-	-	-	ADJ0_MTX_GR[10]	ADJ0_MTX_GR[9]	ADJ0_MTX_GR[8]	
		ADJ0_MTX_GR[7]	ADJ0_MTX_GR[6]	ADJ0_MTX_GR[5]	ADJ0_MTX_GR[4]	ADJ0_MTX_GR[3]	ADJ0_MTX_GR[2]	ADJ0_MTX_GR[1]	ADJ0_MTX_GR[0]	
	ADJ0_MTX_CBB_ADJ0	-	-	-	-	-	-	-	-	-
		ADJ0_MTX_B[7]	ADJ0_MTX_B[6]	ADJ0_MTX_B[5]	ADJ0_MTX_B[4]	ADJ0_MTX_B[3]	ADJ0_MTX_B[2]	ADJ0_MTX_B[1]	ADJ0_MTX_B[0]	
		-	-	-	-	-	ADJ0_MTX_BG[10]	ADJ0_MTX_BG[9]	ADJ0_MTX_BG[8]	
		ADJ0_MTX_BG[7]	ADJ0_MTX_BG[6]	ADJ0_MTX_BG[5]	ADJ0_MTX_BG[4]	ADJ0_MTX_BG[3]	ADJ0_MTX_BG[2]	ADJ0_MTX_BG[1]	ADJ0_MTX_BG[0]	
	ADJ0_MTX_CBB_ADJ1	-	-	-	-	-	-	ADJ0_MTX_BB[10]	ADJ0_MTX_BB[9]	ADJ0_MTX_BB[8]
		ADJ0_MTX_BB[7]	ADJ0_MTX_BB[6]	ADJ0_MTX_BB[5]	ADJ0_MTX_BB[4]	ADJ0_MTX_BB[3]	ADJ0_MTX_BB[2]	ADJ0_MTX_BB[1]	ADJ0_MTX_BB[0]	
		-	-	-	-	-	ADJ0_MTX_BR[10]	ADJ0_MTX_BR[9]	ADJ0_MTX_BR[8]	
		ADJ0_MTX_BR[7]	ADJ0_MTX_BR[6]	ADJ0_MTX_BR[5]	ADJ0_MTX_BR[4]	ADJ0_MTX_BR[3]	ADJ0_MTX_BR[2]	ADJ0_MTX_BR[1]	ADJ0_MTX_BR[0]	
	ADJ0_MTX_CRR_ADJ0	-	-	-	-	-	-	-	-	-
		ADJ0_MTX_R[7]	ADJ0_MTX_R[6]	ADJ0_MTX_R[5]	ADJ0_MTX_R[4]	ADJ0_MTX_R[3]	ADJ0_MTX_R[2]	ADJ0_MTX_R[1]	ADJ0_MTX_R[0]	
		-	-	-	-	-	ADJ0_MTX_RG[10]	ADJ0_MTX_RG[9]	ADJ0_MTX_RG[8]	
		ADJ0_MTX_RG[7]	ADJ0_MTX_RG[6]	ADJ0_MTX_RG[5]	ADJ0_MTX_RG[4]	ADJ0_MTX_RG[3]	ADJ0_MTX_RG[2]	ADJ0_MTX_RG[1]	ADJ0_MTX_RG[0]	
	ADJ0_MTX_CRR_ADJ1	-	-	-	-	-	-	ADJ0_MTX_RB[10]	ADJ0_MTX_RB[9]	ADJ0_MTX_RB[8]
		ADJ0_MTX_RB[7]	ADJ0_MTX_RB[6]	ADJ0_MTX_RB[5]	ADJ0_MTX_RB[4]	ADJ0_MTX_RB[3]	ADJ0_MTX_RB[2]	ADJ0_MTX_RB[1]	ADJ0_MTX_RB[0]	
		-	-	-	-	-	ADJ0_MTX_RR[10]	ADJ0_MTX_RR[9]	ADJ0_MTX_RR[8]	
		ADJ0_MTX_RR[7]	ADJ0_MTX_RR[6]	ADJ0_MTX_RR[5]	ADJ0_MTX_RR[4]	ADJ0_MTX_RR[3]	ADJ0_MTX_RR[2]	ADJ0_MTX_RR[1]	ADJ0_MTX_RR[0]	
	ADJ1_UPDATE	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	ADJ1_VEN
	ADJ1_BKSTR_SET	-	-	-	-	-	-	-	-	BKSTR_ON
		BKSTR_ST[3]	BKSTR_ST[2]	BKSTR_ST[1]	BKSTR_ST[0]	BKSTR_D[3]	BKSTR_D[2]	BKSTR_D[1]	BKSTR_D[0]	
		-	-	-	BKSTR_T1[4]	BKSTR_T1[3]	BKSTR_T1[2]	BKSTR_T1[1]	BKSTR_T1[0]	
		-	-	-	BKSTR_T2[4]	BKSTR_T2[3]	BKSTR_T2[2]	BKSTR_T2[1]	BKSTR_T2[0]	
	ADJ1_ENH_TIM1	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	ENH_MD	-	-	-	-	ENH_DISP_ON
	ADJ1_ENH_TIM2	-	-	-	-	-	-	ENH_VS[10]	ENH_VS[9]	ENH_VS[8]
		ENH_VS[7]	ENH_VS[6]	ENH_VS[5]	ENH_VS[4]	ENH_VS[3]	ENH_VS[2]	ENH_VS[1]	ENH_VS[0]	
		-	-	-	-	-	ENH_VW[10]	ENH_VW[9]	ENH_VW[8]	
		ENH_VW[7]	ENH_VW[6]	ENH_VW[5]	ENH_VW[4]	ENH_VW[3]	ENH_VW[2]	ENH_VW[1]	ENH_VW[0]	
	ADJ1_ENH_TIM3	-	-	-	-	-	-	ENH_HS[10]	ENH_HS[9]	ENH_HS[8]
		ENH_HS[7]	ENH_HS[6]	ENH_HS[5]	ENH_HS[4]	ENH_HS[3]	ENH_HS[2]	ENH_HS[1]	ENH_HS[0]	
		-	-	-	-	-	ENH_HW[10]	ENH_HW[9]	ENH_HW[8]	
		ENH_HW[7]	ENH_HW[6]	ENH_HW[5]	ENH_HW[4]	ENH_HW[3]	ENH_HW[2]	ENH_HW[1]	ENH_HW[0]	
	ADJ1_ENH_SHP1	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	SHP_H_ON
		-	-	-	-	-	-	-	-	-
		-	SHP_H1_CORE[6]	SHP_H1_CORE[5]	SHP_H1_CORE[4]	SHP_H1_CORE[3]	SHP_H1_CORE[2]	SHP_H1_CORE[1]	SHP_H1_CORE[0]	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 0	ADJ1_ENH_SHP2	SHP_H1_CLIP_O[7]	SHP_H1_CLIP_O[6]	SHP_H1_CLIP_O[5]	SHP_H1_CLIP_O[4]	SHP_H1_CLIP_O[3]	SHP_H1_CLIP_O[2]	SHP_H1_CLIP_O[1]	SHP_H1_CLIP_O[0]	
		SHP_H1_CLIP_U [7]	SHP_H1_CLIP_U [6]	SHP_H1_CLIP_U [5]	SHP_H1_CLIP_U [4]	SHP_H1_CLIP_U [3]	SHP_H1_CLIP_U [2]	SHP_H1_CLIP_U [1]	SHP_H1_CLIP_U [0]	
		SHP_H1_GAIN_O[7]	SHP_H1_GAIN_O[6]	SHP_H1_GAIN_O[5]	SHP_H1_GAIN_O[4]	SHP_H1_GAIN_O[3]	SHP_H1_GAIN_O[2]	SHP_H1_GAIN_O[1]	SHP_H1_GAIN_O[0]	
		SHP_H1_GAIN_U[7]	SHP_H1_GAIN_U[6]	SHP_H1_GAIN_U[5]	SHP_H1_GAIN_U[4]	SHP_H1_GAIN_U[3]	SHP_H1_GAIN_U[2]	SHP_H1_GAIN_U[1]	SHP_H1_GAIN_U[0]	
	ADJ1_ENH_SHP3	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	SHP_H2_LPF_SEL
		-	-	-	-	-	-	-	-	-
		-	SHP_H2_CORE[6]	SHP_H2_CORE[5]	SHP_H2_CORE[4]	SHP_H2_CORE[3]	SHP_H2_CORE[2]	SHP_H2_CORE[1]	SHP_H2_CORE[0]	-
	ADJ1_ENH_SHP4	SHP_H2_CLIP_O[7]	SHP_H2_CLIP_O[6]	SHP_H2_CLIP_O[5]	SHP_H2_CLIP_O[4]	SHP_H2_CLIP_O[3]	SHP_H2_CLIP_O[2]	SHP_H2_CLIP_O[1]	SHP_H2_CLIP_O[0]	-
		SHP_H2_CLIP_U [7]	SHP_H2_CLIP_U [6]	SHP_H2_CLIP_U [5]	SHP_H2_CLIP_U [4]	SHP_H2_CLIP_U [3]	SHP_H2_CLIP_U [2]	SHP_H2_CLIP_U [1]	SHP_H2_CLIP_U [0]	-
		SHP_H2_GAIN_O[7]	SHP_H2_GAIN_O[6]	SHP_H2_GAIN_O[5]	SHP_H2_GAIN_O[4]	SHP_H2_GAIN_O[3]	SHP_H2_GAIN_O[2]	SHP_H2_GAIN_O[1]	SHP_H2_GAIN_O[0]	-
		SHP_H2_GAIN_U[7]	SHP_H2_GAIN_U[6]	SHP_H2_GAIN_U[5]	SHP_H2_GAIN_U[4]	SHP_H2_GAIN_U[3]	SHP_H2_GAIN_U[2]	SHP_H2_GAIN_U[1]	SHP_H2_GAIN_U[0]	-
ADJ1_ENH_SHP5	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	
	-	SHP_H3_CORE[6]	SHP_H3_CORE[5]	SHP_H3_CORE[4]	SHP_H3_CORE[3]	SHP_H3_CORE[2]	SHP_H3_CORE[1]	SHP_H3_CORE[0]	-	
ADJ1_ENH_SHP6	SHP_H3_CLIP_O[7]	SHP_H3_CLIP_O[6]	SHP_H3_CLIP_O[5]	SHP_H3_CLIP_O[4]	SHP_H3_CLIP_O[3]	SHP_H3_CLIP_O[2]	SHP_H3_CLIP_O[1]	SHP_H3_CLIP_O[0]	-	
	SHP_H3_CLIP_U [7]	SHP_H3_CLIP_U [6]	SHP_H3_CLIP_U [5]	SHP_H3_CLIP_U [4]	SHP_H3_CLIP_U [3]	SHP_H3_CLIP_U [2]	SHP_H3_CLIP_U [1]	SHP_H3_CLIP_U [0]	-	
	SHP_H3_GAIN_O[7]	SHP_H3_GAIN_O[6]	SHP_H3_GAIN_O[5]	SHP_H3_GAIN_O[4]	SHP_H3_GAIN_O[3]	SHP_H3_GAIN_O[2]	SHP_H3_GAIN_O[1]	SHP_H3_GAIN_O[0]	-	
	SHP_H3_GAIN_U[7]	SHP_H3_GAIN_U[6]	SHP_H3_GAIN_U[5]	SHP_H3_GAIN_U[4]	SHP_H3_GAIN_U[3]	SHP_H3_GAIN_U[2]	SHP_H3_GAIN_U[1]	SHP_H3_GAIN_U[0]	-	
ADJ1_ENH_LTI1	LTI_H_ON	-	-	-	-	-	-	-	LTI_H2_LPF_SEL	
	LTI_H2_INC_ZERO[7]	LTI_H2_INC_ZERO[6]	LTI_H2_INC_ZERO[5]	LTI_H2_INC_ZERO[4]	LTI_H2_INC_ZERO[3]	LTI_H2_INC_ZERO[2]	LTI_H2_INC_ZERO[1]	LTI_H2_INC_ZERO[0]	-	
	LTI_H2_GAIN[7]	LTI_H2_GAIN[6]	LTI_H2_GAIN[5]	LTI_H2_GAIN[4]	LTI_H2_GAIN[3]	LTI_H2_GAIN[2]	LTI_H2_GAIN[1]	LTI_H2_GAIN[0]	-	
	LTI_H2_CORE[7]	LTI_H2_CORE[6]	LTI_H2_CORE[5]	LTI_H2_CORE[4]	LTI_H2_CORE[3]	LTI_H2_CORE[2]	LTI_H2_CORE[1]	LTI_H2_CORE[0]	-	
ADJ1_ENH_LTI2	-	-	-	-	-	-	-	-	LTI_H4_MEDIAN_TAP_SEL	
	LTI_H4_INC_ZERO[7]	LTI_H4_INC_ZERO[6]	LTI_H4_INC_ZERO[5]	LTI_H4_INC_ZERO[4]	LTI_H4_INC_ZERO[3]	LTI_H4_INC_ZERO[2]	LTI_H4_INC_ZERO[1]	LTI_H4_INC_ZERO[0]	-	
	LTI_H4_GAIN[7]	LTI_H4_GAIN[6]	LTI_H4_GAIN[5]	LTI_H4_GAIN[4]	LTI_H4_GAIN[3]	LTI_H4_GAIN[2]	LTI_H4_GAIN[1]	LTI_H4_GAIN[0]	-	
	LTI_H4_CORE[7]	LTI_H4_CORE[6]	LTI_H4_CORE[5]	LTI_H4_CORE[4]	LTI_H4_CORE[3]	LTI_H4_CORE[2]	LTI_H4_CORE[1]	LTI_H4_CORE[0]	-	
ADJ1_MTX_MODE	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	ADJ1_MTX_MD[1]	ADJ1_MTX_MD[0]	-	
ADJ1_MTX_YG_ADJ0	-	-	-	-	-	-	-	-	-	
	ADJ1_MTX_YG[7]	ADJ1_MTX_YG[6]	ADJ1_MTX_YG[5]	ADJ1_MTX_YG[4]	ADJ1_MTX_YG[3]	ADJ1_MTX_YG[2]	ADJ1_MTX_YG[1]	ADJ1_MTX_YG[0]	-	
	-	-	-	-	-	ADJ1_MTX_GG [10]	ADJ1_MTX_GG[9]	ADJ1_MTX_GG[8]	-	
	ADJ1_MTX_GG[7]	ADJ1_MTX_GG[6]	ADJ1_MTX_GG[5]	ADJ1_MTX_GG[4]	ADJ1_MTX_GG[3]	ADJ1_MTX_GG[2]	ADJ1_MTX_GG[1]	ADJ1_MTX_GG[0]	-	
ADJ1_MTX_YG_ADJ1	-	-	-	-	-	ADJ1_MTX_GB [10]	ADJ1_MTX_GB[9]	ADJ1_MTX_GB[8]	-	
	ADJ1_MTX_GB[7]	ADJ1_MTX_GB[6]	ADJ1_MTX_GB[5]	ADJ1_MTX_GB[4]	ADJ1_MTX_GB[3]	ADJ1_MTX_GB[2]	ADJ1_MTX_GB[1]	ADJ1_MTX_GB[0]	-	
	-	-	-	-	-	ADJ1_MTX_GR [10]	ADJ1_MTX_GR[9]	ADJ1_MTX_GR[8]	-	
	ADJ1_MTX_GR[7]	ADJ1_MTX_GR[6]	ADJ1_MTX_GR[5]	ADJ1_MTX_GR[4]	ADJ1_MTX_GR[3]	ADJ1_MTX_GR[2]	ADJ1_MTX_GR[1]	ADJ1_MTX_GR[0]	-	
ADJ1_MTX_CBB_ADJ0	-	-	-	-	-	-	-	-	-	
	ADJ1_MTX_B[7]	ADJ1_MTX_B[6]	ADJ1_MTX_B[5]	ADJ1_MTX_B[4]	ADJ1_MTX_B[3]	ADJ1_MTX_B[2]	ADJ1_MTX_B[1]	ADJ1_MTX_B[0]	-	
	-	-	-	-	-	ADJ1_MTX_BG [10]	ADJ1_MTX_BG[9]	ADJ1_MTX_BG[8]	-	
	ADJ1_MTX_BG[7]	ADJ1_MTX_BG[6]	ADJ1_MTX_BG[5]	ADJ1_MTX_BG[4]	ADJ1_MTX_BG[3]	ADJ1_MTX_BG[2]	ADJ1_MTX_BG[1]	ADJ1_MTX_BG[0]	-	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 0	ADJ1_MTX_CBB_ADJ1	-	-	-	-	-	ADJ1_MTX_BB [10]	ADJ1_MTX_BB[9]	ADJ1_MTX_BB[8]	
		ADJ1_MTX_BB[7]	ADJ1_MTX_BB[6]	ADJ1_MTX_BB[5]	ADJ1_MTX_BB[4]	ADJ1_MTX_BB[3]	ADJ1_MTX_BB[2]	ADJ1_MTX_BB[1]	ADJ1_MTX_BB[0]	
		-	-	-	-	-	ADJ1_MTX_BR [10]	ADJ1_MTX_BR[9]	ADJ1_MTX_BR[8]	
		ADJ1_MTX_BR[7]	ADJ1_MTX_BR[6]	ADJ1_MTX_BR[5]	ADJ1_MTX_BR[4]	ADJ1_MTX_BR[3]	ADJ1_MTX_BR[2]	ADJ1_MTX_BR[1]	ADJ1_MTX_BR[0]	
	ADJ1_MTX_CRR_ADJ0	-	-	-	-	-	-	-	-	
		ADJ1_MTX_R[7]	ADJ1_MTX_R[6]	ADJ1_MTX_R[5]	ADJ1_MTX_R[4]	ADJ1_MTX_R[3]	ADJ1_MTX_R[2]	ADJ1_MTX_R[1]	ADJ1_MTX_R[0]	
		-	-	-	-	-	ADJ1_MTX_RG [10]	ADJ1_MTX_RG[9]	ADJ1_MTX_RG[8]	
		ADJ1_MTX_RG[7]	ADJ1_MTX_RG[6]	ADJ1_MTX_RG[5]	ADJ1_MTX_RG[4]	ADJ1_MTX_RG[3]	ADJ1_MTX_RG[2]	ADJ1_MTX_RG[1]	ADJ1_MTX_RG[0]	
	ADJ1_MTX_CRR_ADJ1	-	-	-	-	-	-	ADJ1_MTX_RB [10]	ADJ1_MTX_RB[9]	ADJ1_MTX_RB[8]
		ADJ1_MTX_RB[7]	ADJ1_MTX_RB[6]	ADJ1_MTX_RB[5]	ADJ1_MTX_RB[4]	ADJ1_MTX_RB[3]	ADJ1_MTX_RB[2]	ADJ1_MTX_RB[1]	ADJ1_MTX_RB[0]	
		-	-	-	-	-	ADJ1_MTX_RR [10]	ADJ1_MTX_RR[9]	ADJ1_MTX_RR[8]	
		ADJ1_MTX_RR[7]	ADJ1_MTX_RR[6]	ADJ1_MTX_RR[5]	ADJ1_MTX_RR[4]	ADJ1_MTX_RR[3]	ADJ1_MTX_RR[2]	ADJ1_MTX_RR[1]	ADJ1_MTX_RR[0]	
	GR2_UPDATE	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	GR2_UPDATE	
		-	-	-	GR2_P_VEN	-	-	-	GR2_IBUS_VEN	
	GR2_FLM_RD	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	GR2_R_ENB	
	GR2_FLM1	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	GR2_LN_OFF_DIR	
		-	-	-	-	-	-	GR2_FLM_SEL[1]	GR2_FLM_SEL[0]	
		-	-	-	-	-	-	-	GR2_BST_MD	
	GR2_FLM2	GR2_BASE[31]	GR2_BASE[30]	GR2_BASE[29]	GR2_BASE[28]	GR2_BASE[27]	GR2_BASE[26]	GR2_BASE[25]	GR2_BASE[24]	
		GR2_BASE[23]	GR2_BASE[22]	GR2_BASE[21]	GR2_BASE[20]	GR2_BASE[19]	GR2_BASE[18]	GR2_BASE[17]	GR2_BASE[16]	
		GR2_BASE[15]	GR2_BASE[14]	GR2_BASE[13]	GR2_BASE[12]	GR2_BASE[11]	GR2_BASE[10]	GR2_BASE[9]	GR2_BASE[8]	
		GR2_BASE[7]	GR2_BASE[6]	GR2_BASE[5]	GR2_BASE[4]	GR2_BASE[3]	GR2_BASE[2]	GR2_BASE[1]	GR2_BASE[0]	
	GR2_FLM3	-	GR2_LN_OFF[14]	GR2_LN_OFF[13]	GR2_LN_OFF[12]	GR2_LN_OFF[11]	GR2_LN_OFF[10]	GR2_LN_OFF[9]	GR2_LN_OFF[8]	
		GR2_LN_OFF[7]	GR2_LN_OFF[6]	GR2_LN_OFF[5]	GR2_LN_OFF[4]	GR2_LN_OFF[3]	GR2_LN_OFF[2]	GR2_LN_OFF[1]	GR2_LN_OFF[0]	
		-	-	-	-	-	-	GR2_FLM_NUM [9]	GR2_FLM_NUM [8]	
		GR2_FLM_NUM [7]	GR2_FLM_NUM [6]	GR2_FLM_NUM [5]	GR2_FLM_NUM [4]	GR2_FLM_NUM [3]	GR2_FLM_NUM [2]	GR2_FLM_NUM [1]	GR2_FLM_NUM [0]	
	GR2_FLM4	-	-	-	-	-	-	-	-	
		-	GR2_FLM_OFF [22]	GR2_FLM_OFF [21]	GR2_FLM_OFF [20]	GR2_FLM_OFF [19]	GR2_FLM_OFF [18]	GR2_FLM_OFF [17]	GR2_FLM_OFF [16]	
		GR2_FLM_OFF [15]	GR2_FLM_OFF [14]	GR2_FLM_OFF [13]	GR2_FLM_OFF [12]	GR2_FLM_OFF [11]	GR2_FLM_OFF [10]	GR2_FLM_OFF[9]	GR2_FLM_OFF[8]	
		GR2_FLM_OFF[7]	GR2_FLM_OFF[6]	GR2_FLM_OFF[5]	GR2_FLM_OFF[4]	GR2_FLM_OFF[3]	GR2_FLM_OFF[2]	GR2_FLM_OFF[1]	GR2_FLM_OFF[0]	
	GR2_FLM5	-	-	-	-	-	GR2_FLM_LNUM[10]	GR2_FLM_LNUM[9]	GR2_FLM_LNUM[8]	
		GR2_FLM_LNUM[7]	GR2_FLM_LNUM[6]	GR2_FLM_LNUM[5]	GR2_FLM_LNUM[4]	GR2_FLM_LNUM[3]	GR2_FLM_LNUM[2]	GR2_FLM_LNUM[1]	GR2_FLM_LNUM[0]	
		-	-	-	-	-	GR2_FLM_LOOP [10]	GR2_FLM_LOOP [9]	GR2_FLM_LOOP [8]	
		GR2_FLM_LOOP [7]	GR2_FLM_LOOP [6]	GR2_FLM_LOOP [5]	GR2_FLM_LOOP [4]	GR2_FLM_LOOP [3]	GR2_FLM_LOOP [2]	GR2_FLM_LOOP [1]	GR2_FLM_LOOP [0]	
	GR2_FLM6	GR2_FORMAT[3]	GR2_FORMAT[2]	GR2_FORMAT[1]	GR2_FORMAT[0]	-	GR2_HW[10]	GR2_HW[9]	GR2_HW[8]	
		GR2_HW[7]	GR2_HW[6]	GR2_HW[5]	GR2_HW[4]	GR2_HW[3]	GR2_HW[2]	GR2_HW[1]	GR2_HW[0]	
		-	-	-	GR2_RDSWA[2]	GR2_RDSWA[1]	GR2_RDSWA[0]	-	-	
		-	-	GR2_STA_POS[5]	GR2_STA_POS[4]	GR2_STA_POS[3]	GR2_STA_POS[2]	GR2_STA_POS[1]	GR2_STA_POS[0]	
GR2_AB1	-	-	-	-	-	-	-	-		
	-	-	-	-	-	-	-	-		
	GR2_ARC_MUL	GR2_ACALC_MD	-	GR2_ARC_ON	-	-	-	GR2_ARC_DISP_ON		
-	-	-	GR2_GRC_DISP_ON	-	-	GR2_DISP_SEL[1]	GR2_DISP_SEL[0]			



Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Video display controller 5 channel 0	GR2_AB2	-	-	-	-	-	GR2_GRC_VS[10]	GR2_GRC_VS[9]	GR2_GRC_VS[8]
		GR2_GRC_VS[7]	GR2_GRC_VS[6]	GR2_GRC_VS[5]	GR2_GRC_VS[4]	GR2_GRC_VS[3]	GR2_GRC_VS[2]	GR2_GRC_VS[1]	GR2_GRC_VS[0]
		-	-	-	-	-	GR2_GRC_VW[10]	GR2_GRC_VW[9]	GR2_GRC_VW[8]
		GR2_GRC_VW[7]	GR2_GRC_VW[6]	GR2_GRC_VW[5]	GR2_GRC_VW[4]	GR2_GRC_VW[3]	GR2_GRC_VW[2]	GR2_GRC_VW[1]	GR2_GRC_VW[0]
	GR2_AB3	-	-	-	-	-	GR2_GRC_HS[10]	GR2_GRC_HS[9]	GR2_GRC_HS[8]
		GR2_GRC_HS[7]	GR2_GRC_HS[6]	GR2_GRC_HS[5]	GR2_GRC_HS[4]	GR2_GRC_HS[3]	GR2_GRC_HS[2]	GR2_GRC_HS[1]	GR2_GRC_HS[0]
		-	-	-	-	-	GR2_GRC_HW[10]	GR2_GRC_HW[9]	GR2_GRC_HW[8]
		GR2_GRC_HW[7]	GR2_GRC_HW[6]	GR2_GRC_HW[5]	GR2_GRC_HW[4]	GR2_GRC_HW[3]	GR2_GRC_HW[2]	GR2_GRC_HW[1]	GR2_GRC_HW[0]
	GR2_AB4	-	-	-	-	-	GR2_ARC_VS[10]	GR2_ARC_VS[9]	GR2_ARC_VS[8]
		GR2_ARC_VS[7]	GR2_ARC_VS[6]	GR2_ARC_VS[5]	GR2_ARC_VS[4]	GR2_ARC_VS[3]	GR2_ARC_VS[2]	GR2_ARC_VS[1]	GR2_ARC_VS[0]
		-	-	-	-	-	GR2_ARC_VW[10]	GR2_ARC_VW[9]	GR2_ARC_VW[8]
		GR2_ARC_VW[7]	GR2_ARC_VW[6]	GR2_ARC_VW[5]	GR2_ARC_VW[4]	GR2_ARC_VW[3]	GR2_ARC_VW[2]	GR2_ARC_VW[1]	GR2_ARC_VW[0]
	GR2_AB5	-	-	-	-	-	GR2_ARC_HS[10]	GR2_ARC_HS[9]	GR2_ARC_HS[8]
		GR2_ARC_HS[7]	GR2_ARC_HS[6]	GR2_ARC_HS[5]	GR2_ARC_HS[4]	GR2_ARC_HS[3]	GR2_ARC_HS[2]	GR2_ARC_HS[1]	GR2_ARC_HS[0]
		-	-	-	-	-	GR2_ARC_HW[10]	GR2_ARC_HW[9]	GR2_ARC_HW[8]
		GR2_ARC_HW[7]	GR2_ARC_HW[6]	GR2_ARC_HW[5]	GR2_ARC_HW[4]	GR2_ARC_HW[3]	GR2_ARC_HW[2]	GR2_ARC_HW[1]	GR2_ARC_HW[0]
	GR2_AB6	-	-	-	-	-	-	-	GR2_ARC_MODE
		GR2_ARC_COEF[7]	GR2_ARC_COEF[6]	GR2_ARC_COEF[5]	GR2_ARC_COEF[4]	GR2_ARC_COEF[3]	GR2_ARC_COEF[2]	GR2_ARC_COEF[1]	GR2_ARC_COEF[0]
		-	-	-	-	-	-	-	-
		GR2_ARC_RATE[7]	GR2_ARC_RATE[6]	GR2_ARC_RATE[5]	GR2_ARC_RATE[4]	GR2_ARC_RATE[3]	GR2_ARC_RATE[2]	GR2_ARC_RATE[1]	GR2_ARC_RATE[0]
	GR2_AB7	-	-	-	-	-	-	-	-
		GR2_ARC_DEF[7]	GR2_ARC_DEF[6]	GR2_ARC_DEF[5]	GR2_ARC_DEF[4]	GR2_ARC_DEF[3]	GR2_ARC_DEF[2]	GR2_ARC_DEF[1]	GR2_ARC_DEF[0]
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GR2_CK_ON
	GR2_AB8	GR2_CK_KCLUT[7]	GR2_CK_KCLUT[6]	GR2_CK_KCLUT[5]	GR2_CK_KCLUT[4]	GR2_CK_KCLUT[3]	GR2_CK_KCLUT[2]	GR2_CK_KCLUT[1]	GR2_CK_KCLUT[0]
		GR2_CK_KG[7]	GR2_CK_KG[6]	GR2_CK_KG[5]	GR2_CK_KG[4]	GR2_CK_KG[3]	GR2_CK_KG[2]	GR2_CK_KG[1]	GR2_CK_KG[0]
		GR2_CK_KB[7]	GR2_CK_KB[6]	GR2_CK_KB[5]	GR2_CK_KB[4]	GR2_CK_KB[3]	GR2_CK_KB[2]	GR2_CK_KB[1]	GR2_CK_KB[0]
		GR2_CK_KR[7]	GR2_CK_KR[6]	GR2_CK_KR[5]	GR2_CK_KR[4]	GR2_CK_KR[3]	GR2_CK_KR[2]	GR2_CK_KR[1]	GR2_CK_KR[0]
	GR2_AB9	GR2_CK_A[7]	GR2_CK_A[6]	GR2_CK_A[5]	GR2_CK_A[4]	GR2_CK_A[3]	GR2_CK_A[2]	GR2_CK_A[1]	GR2_CK_A[0]
		GR2_CK_G[7]	GR2_CK_G[6]	GR2_CK_G[5]	GR2_CK_G[4]	GR2_CK_G[3]	GR2_CK_G[2]	GR2_CK_G[1]	GR2_CK_G[0]
		GR2_CK_B[7]	GR2_CK_B[6]	GR2_CK_B[5]	GR2_CK_B[4]	GR2_CK_B[3]	GR2_CK_B[2]	GR2_CK_B[1]	GR2_CK_B[0]
		GR2_CK_R[7]	GR2_CK_R[6]	GR2_CK_R[5]	GR2_CK_R[4]	GR2_CK_R[3]	GR2_CK_R[2]	GR2_CK_R[1]	GR2_CK_R[0]
	GR2_AB10	GR2_A0[7]	GR2_A0[6]	GR2_A0[5]	GR2_A0[4]	GR2_A0[3]	GR2_A0[2]	GR2_A0[1]	GR2_A0[0]
		GR2_G0[7]	GR2_G0[6]	GR2_G0[5]	GR2_G0[4]	GR2_G0[3]	GR2_G0[2]	GR2_G0[1]	GR2_G0[0]
		GR2_B0[7]	GR2_B0[6]	GR2_B0[5]	GR2_B0[4]	GR2_B0[3]	GR2_B0[2]	GR2_B0[1]	GR2_B0[0]
		GR2_R0[7]	GR2_R0[6]	GR2_R0[5]	GR2_R0[4]	GR2_R0[3]	GR2_R0[2]	GR2_R0[1]	GR2_R0[0]
	GR2_AB11	GR2_A1[7]	GR2_A1[6]	GR2_A1[5]	GR2_A1[4]	GR2_A1[3]	GR2_A1[2]	GR2_A1[1]	GR2_A1[0]
		GR2_G1[7]	GR2_G1[6]	GR2_G1[5]	GR2_G1[4]	GR2_G1[3]	GR2_G1[2]	GR2_G1[1]	GR2_G1[0]
		GR2_B1[7]	GR2_B1[6]	GR2_B1[5]	GR2_B1[4]	GR2_B1[3]	GR2_B1[2]	GR2_B1[1]	GR2_B1[0]
		GR2_R1[7]	GR2_R1[6]	GR2_R1[5]	GR2_R1[4]	GR2_R1[3]	GR2_R1[2]	GR2_R1[1]	GR2_R1[0]
	GR2_BASE	-	-	-	-	-	-	-	-
		GR2_BASE_G[7]	GR2_BASE_G[6]	GR2_BASE_G[5]	GR2_BASE_G[4]	GR2_BASE_G[3]	GR2_BASE_G[2]	GR2_BASE_G[1]	GR2_BASE_G[0]
		GR2_BASE_B[7]	GR2_BASE_B[6]	GR2_BASE_B[5]	GR2_BASE_B[4]	GR2_BASE_B[3]	GR2_BASE_B[2]	GR2_BASE_B[1]	GR2_BASE_B[0]
		GR2_BASE_R[7]	GR2_BASE_R[6]	GR2_BASE_R[5]	GR2_BASE_R[4]	GR2_BASE_R[3]	GR2_BASE_R[2]	GR2_BASE_R[1]	GR2_BASE_R[0]
	GR2_CLUT	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GR2_CLT_SEL
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	GR2_MON	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GR2_ARC_ST

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Video display controller 5 channel 0	GR3_UPDATE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GR3_UPDATE
		-	-	-	GR3_P_VEN	-	-	-	GR3_IBUS_VEN
	GR3_FLM_RD	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GR3_R_ENB
	GR3_FLM1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GR3_LN_OFF_DIR
		-	-	-	-	-	-	GR3_FLM_SEL[1]	GR3_FLM_SEL[0]
		-	-	-	-	-	-	-	GR3_BST_MD
	GR3_FLM2	GR3_BASE[31]	GR3_BASE[30]	GR3_BASE[29]	GR3_BASE[28]	GR3_BASE[27]	GR3_BASE[26]	GR3_BASE[25]	GR3_BASE[24]
		GR3_BASE[23]	GR3_BASE[22]	GR3_BASE[21]	GR3_BASE[20]	GR3_BASE[19]	GR3_BASE[18]	GR3_BASE[17]	GR3_BASE[16]
		GR3_BASE[15]	GR3_BASE[14]	GR3_BASE[13]	GR3_BASE[12]	GR3_BASE[11]	GR3_BASE[10]	GR3_BASE[9]	GR3_BASE[8]
		GR3_BASE[7]	GR3_BASE[6]	GR3_BASE[5]	GR3_BASE[4]	GR3_BASE[3]	GR3_BASE[2]	GR3_BASE[1]	GR3_BASE[0]
	GR3_FLM3	-	GR3_LN_OFF[14]	GR3_LN_OFF[13]	GR3_LN_OFF[12]	GR3_LN_OFF[11]	GR3_LN_OFF[10]	GR3_LN_OFF[9]	GR3_LN_OFF[8]
		GR3_LN_OFF[7]	GR3_LN_OFF[6]	GR3_LN_OFF[5]	GR3_LN_OFF[4]	GR3_LN_OFF[3]	GR3_LN_OFF[2]	GR3_LN_OFF[1]	GR3_LN_OFF[0]
		-	-	-	-	-	-	GR3_FLM_NUM[9]	GR3_FLM_NUM[8]
		GR3_FLM_NUM[7]	GR3_FLM_NUM[6]	GR3_FLM_NUM[5]	GR3_FLM_NUM[4]	GR3_FLM_NUM[3]	GR3_FLM_NUM[2]	GR3_FLM_NUM[1]	GR3_FLM_NUM[0]
	GR3_FLM4	-	-	-	-	-	-	-	-
		-	GR3_FLM_OFF[22]	GR3_FLM_OFF[21]	GR3_FLM_OFF[20]	GR3_FLM_OFF[19]	GR3_FLM_OFF[18]	GR3_FLM_OFF[17]	GR3_FLM_OFF[16]
		GR3_FLM_OFF[15]	GR3_FLM_OFF[14]	GR3_FLM_OFF[13]	GR3_FLM_OFF[12]	GR3_FLM_OFF[11]	GR3_FLM_OFF[10]	GR3_FLM_OFF[9]	GR3_FLM_OFF[8]
		GR3_FLM_OFF[7]	GR3_FLM_OFF[6]	GR3_FLM_OFF[5]	GR3_FLM_OFF[4]	GR3_FLM_OFF[3]	GR3_FLM_OFF[2]	GR3_FLM_OFF[1]	GR3_FLM_OFF[0]
	GR3_FLM5	-	-	-	-	-	GR3_FLM_LNUM[10]	GR3_FLM_LNUM[9]	GR3_FLM_LNUM[8]
		GR3_FLM_LNUM[7]	GR3_FLM_LNUM[6]	GR3_FLM_LNUM[5]	GR3_FLM_LNUM[4]	GR3_FLM_LNUM[3]	GR3_FLM_LNUM[2]	GR3_FLM_LNUM[1]	GR3_FLM_LNUM[0]
		-	-	-	-	-	GR3_FLM_LOOP[10]	GR3_FLM_LOOP[9]	GR3_FLM_LOOP[8]
		GR3_FLM_LOOP[7]	GR3_FLM_LOOP[6]	GR3_FLM_LOOP[5]	GR3_FLM_LOOP[4]	GR3_FLM_LOOP[3]	GR3_FLM_LOOP[2]	GR3_FLM_LOOP[1]	GR3_FLM_LOOP[0]
	GR3_FLM6	GR3_FORMAT[3]	GR3_FORMAT[2]	GR3_FORMAT[1]	GR3_FORMAT[0]	-	GR3_HW[10]	GR3_HW[9]	GR3_HW[8]
		GR3_HW[7]	GR3_HW[6]	GR3_HW[5]	GR3_HW[4]	GR3_HW[3]	GR3_HW[2]	GR3_HW[1]	GR3_HW[0]
		-	-	-	GR3_RDSWA[2]	GR3_RDSWA[1]	GR3_RDSWA[0]	-	-
		-	-	GR3_STA_POS[5]	GR3_STA_POS[4]	GR3_STA_POS[3]	GR3_STA_POS[2]	GR3_STA_POS[1]	GR3_STA_POS[0]
	GR3_AB1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		GR3_ARC_MUL	GR3_ACALC_MD	-	GR3_ARC_ON	-	-	-	GR3_ARC_DISP_ON
		-	-	-	GR3_GRC_DISP_ON	-	-	GR3_DISP_SEL[1]	GR3_DISP_SEL[0]
	GR3_AB2	-	-	-	-	-	GR3_GRC_VS[10]	GR3_GRC_VS[9]	GR3_GRC_VS[8]
		GR3_GRC_VS[7]	GR3_GRC_VS[6]	GR3_GRC_VS[5]	GR3_GRC_VS[4]	GR3_GRC_VS[3]	GR3_GRC_VS[2]	GR3_GRC_VS[1]	GR3_GRC_VS[0]
		-	-	-	-	-	GR3_GRC_VW[10]	GR3_GRC_VW[9]	GR3_GRC_VW[8]
		GR3_GRC_VW[7]	GR3_GRC_VW[6]	GR3_GRC_VW[5]	GR3_GRC_VW[4]	GR3_GRC_VW[3]	GR3_GRC_VW[2]	GR3_GRC_VW[1]	GR3_GRC_VW[0]
	GR3_AB3	-	-	-	-	-	GR3_GRC_HS[10]	GR3_GRC_HS[9]	GR3_GRC_HS[8]
		GR3_GRC_HS[7]	GR3_GRC_HS[6]	GR3_GRC_HS[5]	GR3_GRC_HS[4]	GR3_GRC_HS[3]	GR3_GRC_HS[2]	GR3_GRC_HS[1]	GR3_GRC_HS[0]
		-	-	-	-	-	GR3_GRC_HW[10]	GR3_GRC_HW[9]	GR3_GRC_HW[8]
		GR3_GRC_HW[7]	GR3_GRC_HW[6]	GR3_GRC_HW[5]	GR3_GRC_HW[4]	GR3_GRC_HW[3]	GR3_GRC_HW[2]	GR3_GRC_HW[1]	GR3_GRC_HW[0]
	GR3_AB4	-	-	-	-	-	GR3_ARC_VS[10]	GR3_ARC_VS[9]	GR3_ARC_VS[8]
		GR3_ARC_VS[7]	GR3_ARC_VS[6]	GR3_ARC_VS[5]	GR3_ARC_VS[4]	GR3_ARC_VS[3]	GR3_ARC_VS[2]	GR3_ARC_VS[1]	GR3_ARC_VS[0]
		-	-	-	-	-	GR3_ARC_VW[10]	GR3_ARC_VW[9]	GR3_ARC_VW[8]
		GR3_ARC_VW[7]	GR3_ARC_VW[6]	GR3_ARC_VW[5]	GR3_ARC_VW[4]	GR3_ARC_VW[3]	GR3_ARC_VW[2]	GR3_ARC_VW[1]	GR3_ARC_VW[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Video display controller 5 channel 0	GR3_AB5	-	-	-	-	-	GR3_ARC_HS[10]	GR3_ARC_HS[9]	GR3_ARC_HS[0]
		GR3_ARC_HS[7]	GR3_ARC_HS[6]	GR3_ARC_HS[5]	GR3_ARC_HS[4]	GR3_ARC_HS[3]	GR3_ARC_HS[2]	GR3_ARC_HS[1]	GR3_ARC_HS[0]
		-	-	-	-	-	GR3_ARC_HW[10]	GR3_ARC_HW[9]	GR3_ARC_HW[8]
		GR3_ARC_HW[7]	GR3_ARC_HW[6]	GR3_ARC_HW[5]	GR3_ARC_HW[4]	GR3_ARC_HW[3]	GR3_ARC_HW[2]	GR3_ARC_HW[1]	GR3_ARC_HW[0]
	GR3_AB6	-	-	-	-	-	-	-	GR3_ARC_MODE
		GR3_ARC_COEF[7]	GR3_ARC_COEF[6]	GR3_ARC_COEF[5]	GR3_ARC_COEF[4]	GR3_ARC_COEF[3]	GR3_ARC_COEF[2]	GR3_ARC_COEF[1]	GR3_ARC_COEF[0]
		-	-	-	-	-	-	-	-
		GR3_ARC_RATE[7]	GR3_ARC_RATE[6]	GR3_ARC_RATE[5]	GR3_ARC_RATE[4]	GR3_ARC_RATE[3]	GR3_ARC_RATE[2]	GR3_ARC_RATE[1]	GR3_ARC_RATE[0]
	GR3_AB7	-	-	-	-	-	-	-	-
		GR3_ARC_DEF[7]	GR3_ARC_DEF[6]	GR3_ARC_DEF[5]	GR3_ARC_DEF[4]	GR3_ARC_DEF[3]	GR3_ARC_DEF[2]	GR3_ARC_DEF[1]	GR3_ARC_DEF[0]
		-	-	-	-	-	-	-	GR3_CK_ON
	GR3_AB8	GR3_CK_KCLUT[7]	GR3_CK_KCLUT[6]	GR3_CK_KCLUT[5]	GR3_CK_KCLUT[4]	GR3_CK_KCLUT[3]	GR3_CK_KCLUT[2]	GR3_CK_KCLUT[1]	GR3_CK_KCLUT[0]
		GR3_CK_KG[7]	GR3_CK_KG[6]	GR3_CK_KG[5]	GR3_CK_KG[4]	GR3_CK_KG[3]	GR3_CK_KG[2]	GR3_CK_KG[1]	GR3_CK_KG[0]
		GR3_CK_KB[7]	GR3_CK_KB[6]	GR3_CK_KB[5]	GR3_CK_KB[4]	GR3_CK_KB[3]	GR3_CK_KB[2]	GR3_CK_KB[1]	GR3_CK_KB[0]
		GR3_CK_KR[7]	GR3_CK_KR[6]	GR3_CK_KR[5]	GR3_CK_KR[4]	GR3_CK_KR[3]	GR3_CK_KR[2]	GR3_CK_KR[1]	GR3_CK_KR[0]
	GR3_AB9	GR3_CK_A[7]	GR3_CK_A[6]	GR3_CK_A[5]	GR3_CK_A[4]	GR3_CK_A[3]	GR3_CK_A[2]	GR3_CK_A[1]	GR3_CK_A[0]
		GR3_CK_G[7]	GR3_CK_G[6]	GR3_CK_G[5]	GR3_CK_G[4]	GR3_CK_G[3]	GR3_CK_G[2]	GR3_CK_G[1]	GR3_CK_G[0]
		GR3_CK_B[7]	GR3_CK_B[6]	GR3_CK_B[5]	GR3_CK_B[4]	GR3_CK_B[3]	GR3_CK_B[2]	GR3_CK_B[1]	GR3_CK_B[0]
		GR3_CK_R[7]	GR3_CK_R[6]	GR3_CK_R[5]	GR3_CK_R[4]	GR3_CK_R[3]	GR3_CK_R[2]	GR3_CK_R[1]	GR3_CK_R[0]
	GR3_AB10	GR3_A0[7]	GR3_A0[6]	GR3_A0[5]	GR3_A0[4]	GR3_A0[3]	GR3_A0[2]	GR3_A0[1]	GR3_A0[0]
		GR3_G0[7]	GR3_G0[6]	GR3_G0[5]	GR3_G0[4]	GR3_G0[3]	GR3_G0[2]	GR3_G0[1]	GR3_G0[0]
		GR3_B0[7]	GR3_B0[6]	GR3_B0[5]	GR3_B0[4]	GR3_B0[3]	GR3_B0[2]	GR3_B0[1]	GR3_B0[0]
		GR3_R0[7]	GR3_R0[6]	GR3_R0[5]	GR3_R0[4]	GR3_R0[3]	GR3_R0[2]	GR3_R0[1]	GR3_R0[0]
	GR3_AB11	GR3_A1[7]	GR3_A1[6]	GR3_A1[5]	GR3_A1[4]	GR3_A1[3]	GR3_A1[2]	GR3_A1[1]	GR3_A1[0]
		GR3_G1[7]	GR3_G1[6]	GR3_G1[5]	GR3_G1[4]	GR3_G1[3]	GR3_G1[2]	GR3_G1[1]	GR3_G1[0]
		GR3_B1[7]	GR3_B1[6]	GR3_B1[5]	GR3_B1[4]	GR3_B1[3]	GR3_B1[2]	GR3_B1[1]	GR3_B1[0]
		GR3_R1[7]	GR3_R1[6]	GR3_R1[5]	GR3_R1[4]	GR3_R1[3]	GR3_R1[2]	GR3_R1[1]	GR3_R1[0]
	GR3_BASE	-	-	-	-	-	-	-	-
		GR3_BASE_G[7]	GR3_BASE_G[6]	GR3_BASE_G[5]	GR3_BASE_G[4]	GR3_BASE_G[3]	GR3_BASE_G[2]	GR3_BASE_G[1]	GR3_BASE_G[0]
		GR3_BASE_B[7]	GR3_BASE_B[6]	GR3_BASE_B[5]	GR3_BASE_B[4]	GR3_BASE_B[3]	GR3_BASE_B[2]	GR3_BASE_B[1]	GR3_BASE_B[0]
	GR3_CLUT_INT	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GR3_CLT_SEL
		-	-	-	-	-	GR3_LINE[10]	GR3_LINE[9]	GR3_LINE[8]
		GR3_LINE[7]	GR3_LINE[6]	GR3_LINE[5]	GR3_LINE[4]	GR3_LINE[3]	GR3_LINE[2]	GR3_LINE[1]	GR3_LINE[0]
	GR3_MON	-	-	-	-	-	GR3_LIN_STAT[10]	GR3_LIN_STAT[9]	GR3_LIN_STAT[8]
		GR3_LIN_STAT[7]	GR3_LIN_STAT[6]	GR3_LIN_STAT[5]	GR3_LIN_STAT[4]	GR3_LIN_STAT[3]	GR3_LIN_STAT[2]	GR3_LIN_STAT[1]	GR3_LIN_STAT[0]
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GR3_ARC_ST
	GR_VIN_UPDATE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GR_VIN_UPDATE
		-	-	-	GR_VIN_P_VEN	-	-	-	-
	GR_VIN_AB1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	GR_VIN_ARC_DISP_ON	-	-	-	GR_VIN_ARC_DISP_ON
		-	-	-	GR_VIN_GRC_DISP_ON	-	GR_VIN_SCL_UND_SEL	GR_VIN_DISP_SEL[1]	GR_VIN_DISP_SEL[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 0	GR_VIN_AB2	-	-	-	-	-	GR_VIN_GRC_VS[10]	GR_VIN_GRC_VS[9]	GR_VIN_GRC_VS[8]	
		GR_VIN_GRC_VS[7]	GR_VIN_GRC_VS[6]	GR_VIN_GRC_VS[5]	GR_VIN_GRC_VS[4]	GR_VIN_GRC_VS[3]	GR_VIN_GRC_VS[2]	GR_VIN_GRC_VS[1]	GR_VIN_GRC_VS[0]	
		-	-	-	-	-	GR_VIN_GRC_VW[10]	GR_VIN_GRC_VW[9]	GR_VIN_GRC_VW[8]	
		GR_VIN_GRC_VW[7]	GR_VIN_GRC_VW[6]	GR_VIN_GRC_VW[5]	GR_VIN_GRC_VW[4]	GR_VIN_GRC_VW[3]	GR_VIN_GRC_VW[2]	GR_VIN_GRC_VW[1]	GR_VIN_GRC_VW[0]	
	GR_VIN_AB3	-	-	-	-	-	-	GR_VIN_GRC_HS[10]	GR_VIN_GRC_HS[9]	GR_VIN_GRC_HS[8]
		GR_VIN_GRC_HS[7]	GR_VIN_GRC_HS[6]	GR_VIN_GRC_HS[5]	GR_VIN_GRC_HS[4]	GR_VIN_GRC_HS[3]	GR_VIN_GRC_HS[2]	GR_VIN_GRC_HS[1]	GR_VIN_GRC_HS[0]	
		-	-	-	-	-	GR_VIN_GRC_HW[10]	GR_VIN_GRC_HW[9]	GR_VIN_GRC_HW[8]	
		GR_VIN_GRC_HW[7]	GR_VIN_GRC_HW[6]	GR_VIN_GRC_HW[5]	GR_VIN_GRC_HW[4]	GR_VIN_GRC_HW[3]	GR_VIN_GRC_HW[2]	GR_VIN_GRC_HW[1]	GR_VIN_GRC_HW[0]	
	GR_VIN_AB4	-	-	-	-	-	-	GR_VIN_ARC_VS[10]	GR_VIN_ARC_VS[9]	GR_VIN_ARC_VS[8]
		GR_VIN_ARC_VS[7]	GR_VIN_ARC_VS[6]	GR_VIN_ARC_VS[5]	GR_VIN_ARC_VS[4]	GR_VIN_ARC_VS[3]	GR_VIN_ARC_VS[2]	GR_VIN_ARC_VS[1]	GR_VIN_ARC_VS[0]	
		-	-	-	-	-	GR_VIN_ARC_VW[10]	GR_VIN_ARC_VW[9]	GR_VIN_ARC_VW[8]	
		GR_VIN_ARC_VW[7]	GR_VIN_ARC_VW[6]	GR_VIN_ARC_VW[5]	GR_VIN_ARC_VW[4]	GR_VIN_ARC_VW[3]	GR_VIN_ARC_VW[2]	GR_VIN_ARC_VW[1]	GR_VIN_ARC_VW[0]	
	GR_VIN_AB5	-	-	-	-	-	-	GR_VIN_ARC_HS[10]	GR_VIN_ARC_HS[9]	GR_VIN_ARC_HS[8]
		GR_VIN_ARC_HS[7]	GR_VIN_ARC_HS[6]	GR_VIN_ARC_HS[5]	GR_VIN_ARC_HS[4]	GR_VIN_ARC_HS[3]	GR_VIN_ARC_HS[2]	GR_VIN_ARC_HS[1]	GR_VIN_ARC_HS[0]	
		-	-	-	-	-	GR_VIN_ARC_HW[10]	GR_VIN_ARC_HW[9]	GR_VIN_ARC_HW[8]	
		GR_VIN_ARC_HW[7]	GR_VIN_ARC_HW[6]	GR_VIN_ARC_HW[5]	GR_VIN_ARC_HW[4]	GR_VIN_ARC_HW[3]	GR_VIN_ARC_HW[2]	GR_VIN_ARC_HW[1]	GR_VIN_ARC_HW[0]	
	GR_VIN_AB6	-	-	-	-	-	-	-	-	GR_VIN_ARC_MODE
		GR_VIN_ARC_COEF[7]	GR_VIN_ARC_COEF[6]	GR_VIN_ARC_COEF[5]	GR_VIN_ARC_COEF[4]	GR_VIN_ARC_COEF[3]	GR_VIN_ARC_COEF[2]	GR_VIN_ARC_COEF[1]	GR_VIN_ARC_COEF[0]	
		-	-	-	-	-	-	-	-	
		GR_VIN_ARC_RATE[7]	GR_VIN_ARC_RATE[6]	GR_VIN_ARC_RATE[5]	GR_VIN_ARC_RATE[4]	GR_VIN_ARC_RATE[3]	GR_VIN_ARC_RATE[2]	GR_VIN_ARC_RATE[1]	GR_VIN_ARC_RATE[0]	
	GR_VIN_AB7	-	-	-	-	-	-	-	-	-
		GR_VIN_ARC_DEF[7]	GR_VIN_ARC_DEF[6]	GR_VIN_ARC_DEF[5]	GR_VIN_ARC_DEF[4]	GR_VIN_ARC_DEF[3]	GR_VIN_ARC_DEF[2]	GR_VIN_ARC_DEF[1]	GR_VIN_ARC_DEF[0]	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
	GR_VIN_BASE	-	-	-	-	-	-	-	-	
		GR_VIN_BASE_G[7]	GR_VIN_BASE_G[6]	GR_VIN_BASE_G[5]	GR_VIN_BASE_G[4]	GR_VIN_BASE_G[3]	GR_VIN_BASE_G[2]	GR_VIN_BASE_G[1]	GR_VIN_BASE_G[0]	
		GR_VIN_BASE_B[7]	GR_VIN_BASE_B[6]	GR_VIN_BASE_B[5]	GR_VIN_BASE_B[4]	GR_VIN_BASE_B[3]	GR_VIN_BASE_B[2]	GR_VIN_BASE_B[1]	GR_VIN_BASE_B[0]	
	GR_VIN_MON	GR_VIN_BASE_R[7]	GR_VIN_BASE_R[6]	GR_VIN_BASE_R[5]	GR_VIN_BASE_R[4]	GR_VIN_BASE_R[3]	GR_VIN_BASE_R[2]	GR_VIN_BASE_R[1]	GR_VIN_BASE_R[0]	
		-	-	-	-	-	-	-	-	
	OIR_SCL0_UPDATE	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	OIR_SCL0_VEN_D	OIR_SCL0_VEN_C	-	-	-	OIR_SCL0_UPDATE	
		-	-	-	OIR_SCL0_VEN_B	-	-	-	OIR_SCL0_VEN_A	
	OIR_SCL0_FRC1	OIR_RES_VMASK[15]	OIR_RES_VMASK[14]	OIR_RES_VMASK[13]	OIR_RES_VMASK[12]	OIR_RES_VMASK[11]	OIR_RES_VMASK[10]	OIR_RES_VMASK[9]	OIR_RES_VMASK[8]	
		OIR_RES_VMASK[7]	OIR_RES_VMASK[6]	OIR_RES_VMASK[5]	OIR_RES_VMASK[4]	OIR_RES_VMASK[3]	OIR_RES_VMASK[2]	OIR_RES_VMASK[1]	OIR_RES_VMASK[0]	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	OIR_RES_VMASK_ON	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 0	OIR_SCL0_FRC2	OIR_RES_VLACK[15]	OIR_RES_VLACK[14]	OIR_RES_VLACK[13]	OIR_RES_VLACK[12]	OIR_RES_VLACK[11]	OIR_RES_VLACK[10]	OIR_RES_VLACK[9]	OIR_RES_VLACK[8]	
		OIR_RES_VLACK[7]	OIR_RES_VLACK[6]	OIR_RES_VLACK[5]	OIR_RES_VLACK[4]	OIR_RES_VLACK[3]	OIR_RES_VLACK[2]	OIR_RES_VLACK[1]	OIR_RES_VLACK[0]	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	OIR_RES_VLACK_ON	
	OIR_SCL0_FRC3	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	OIR_RES_EN	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	OIR_RES_VS_SEL	
	OIR_SCL0_FRC4	-	-	-	-	-	-	OIR_RES_FV[10]	OIR_RES_FV[9]	OIR_RES_FV[8]
		OIR_RES_FV[7]	OIR_RES_FV[6]	OIR_RES_FV[5]	OIR_RES_FV[4]	OIR_RES_FV[3]	OIR_RES_FV[2]	OIR_RES_FV[1]	OIR_RES_FV[0]	
		-	-	-	-	-	OIR_RES_FH[10]	OIR_RES_FH[9]	OIR_RES_FH[8]	
		OIR_RES_FH[7]	OIR_RES_FH[6]	OIR_RES_FH[5]	OIR_RES_FH[4]	OIR_RES_FH[3]	OIR_RES_FH[2]	OIR_RES_FH[1]	OIR_RES_FH[0]	
	OIR_SCL0_FRC5	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		OIR_RES_VSDLY[7]	OIR_RES_VSDLY[6]	OIR_RES_VSDLY[5]	OIR_RES_VSDLY[4]	OIR_RES_VSDLY[3]	OIR_RES_VSDLY[2]	OIR_RES_VSDLY[1]	OIR_RES_VSDLY[0]	
	OIR_SCL0_FRC6	-	-	-	-	-	-	OIR_RES_F_VS [10]	OIR_RES_F_VS [9]	OIR_RES_F_VS [8]
		OIR_RES_F_VS [7]	OIR_RES_F_VS [6]	OIR_RES_F_VS [5]	OIR_RES_F_VS [4]	OIR_RES_F_VS [3]	OIR_RES_F_VS [2]	OIR_RES_F_VS [1]	OIR_RES_F_VS [0]	
		-	-	-	-	-	OIR_RES_F_VW [10]	OIR_RES_F_VW [9]	OIR_RES_F_VW [8]	
		OIR_RES_F_VW [7]	OIR_RES_F_VW [6]	OIR_RES_F_VW [5]	OIR_RES_F_VW [4]	OIR_RES_F_VW [3]	OIR_RES_F_VW [2]	OIR_RES_F_VW [1]	OIR_RES_F_VW [0]	
	OIR_SCL0_FRC7	-	-	-	-	-	-	OIR_RES_F_HS [10]	OIR_RES_F_HS [9]	OIR_RES_F_HS [8]
		OIR_RES_F_HS [7]	OIR_RES_F_HS [6]	OIR_RES_F_HS [5]	OIR_RES_F_HS [4]	OIR_RES_F_HS [3]	OIR_RES_F_HS [2]	OIR_RES_F_HS [1]	OIR_RES_F_HS [0]	
		-	-	-	-	-	OIR_RES_F_HW [10]	OIR_RES_F_HW [9]	OIR_RES_F_HW [8]	
		OIR_RES_F_HW [7]	OIR_RES_F_HW [6]	OIR_RES_F_HW [5]	OIR_RES_F_HW [4]	OIR_RES_F_HW [3]	OIR_RES_F_HW [2]	OIR_RES_F_HW [1]	OIR_RES_F_HW [0]	
	OIR_SCL0_DS1	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	OIR_RES_DS_V_ON	-	-	-	-	OIR_RES_DS_H_ON
	OIR_SCL0_DS2	-	-	-	-	-	-	OIR_RES_VS[10]	OIR_RES_VS[9]	OIR_RES_VS[8]
		OIR_RES_VS[7]	OIR_RES_VS[6]	OIR_RES_VS[5]	OIR_RES_VS[4]	OIR_RES_VS[3]	OIR_RES_VS[2]	OIR_RES_VS[1]	OIR_RES_VS[0]	
		-	-	-	-	-	OIR_RES_VW[10]	OIR_RES_VW[9]	OIR_RES_VW[8]	
		OIR_RES_VW[7]	OIR_RES_VW[6]	OIR_RES_VW[5]	OIR_RES_VW[4]	OIR_RES_VW[3]	OIR_RES_VW[2]	OIR_RES_VW[1]	OIR_RES_VW[0]	
	OIR_SCL0_DS3	-	-	-	-	-	-	OIR_RES_HS[10]	OIR_RES_HS[9]	OIR_RES_HS[8]
		OIR_RES_HS[7]	OIR_RES_HS[6]	OIR_RES_HS[5]	OIR_RES_HS[4]	OIR_RES_HS[3]	OIR_RES_HS[2]	OIR_RES_HS[1]	OIR_RES_HS[0]	
		-	-	-	-	-	OIR_RES_HW[10]	OIR_RES_HW[9]	OIR_RES_HW[8]	
		OIR_RES_HW[7]	OIR_RES_HW[6]	OIR_RES_HW[5]	OIR_RES_HW[4]	OIR_RES_HW[3]	OIR_RES_HW[2]	OIR_RES_HW[1]	OIR_RES_HW[0]	
	OIR_SCL0_DS7	-	-	-	-	-	-	OIR_RES_OUT_VW[10]	OIR_RES_OUT_VW[9]	OIR_RES_OUT_VW[8]
		OIR_RES_OUT_VW[7]	OIR_RES_OUT_VW[6]	OIR_RES_OUT_VW[5]	OIR_RES_OUT_VW[4]	OIR_RES_OUT_VW[3]	OIR_RES_OUT_VW[2]	OIR_RES_OUT_VW[1]	OIR_RES_OUT_VW[0]	
		-	-	-	-	-	OIR_RES_OUT_HW[10]	OIR_RES_OUT_HW[9]	OIR_RES_OUT_HW[8]	
		OIR_RES_OUT_HW[7]	OIR_RES_OUT_HW[6]	OIR_RES_OUT_HW[5]	OIR_RES_OUT_HW[4]	OIR_RES_OUT_HW[3]	OIR_RES_OUT_HW[2]	OIR_RES_OUT_HW[1]	OIR_RES_OUT_HW[0]	
	OIR_SCL0_US1	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	OIR_RES_US_V_ON	-	-	-	-	OIR_RES_US_H_ON

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 0	OIR_SCL0_US2	-	-	-	-	-	OIR_RES_P_VS [10]	OIR_RES_P_VS [9]	OIR_RES_P_VS [8]	
		OIR_RES_P_VS [7]	OIR_RES_P_VS [6]	OIR_RES_P_VS [5]	OIR_RES_P_VS [4]	OIR_RES_P_VS [3]	OIR_RES_P_VS [2]	OIR_RES_P_VS [1]	OIR_RES_P_VS [0]	
		-	-	-	-	-	OIR_RES_P_VW [10]	OIR_RES_P_VW [9]	OIR_RES_P_VW [8]	
		OIR_RES_P_VW [7]	OIR_RES_P_VW [6]	OIR_RES_P_VW [5]	OIR_RES_P_VW [4]	OIR_RES_P_VW [3]	OIR_RES_P_VW [2]	OIR_RES_P_VW [1]	OIR_RES_P_VW [0]	
	OIR_SCL0_US3	-	-	-	-	-	-	OIR_RES_P_HS [10]	OIR_RES_P_HS [9]	OIR_RES_P_HS [8]
		OIR_RES_P_HS [7]	OIR_RES_P_HS [6]	OIR_RES_P_HS [5]	OIR_RES_P_HS [4]	OIR_RES_P_HS [3]	OIR_RES_P_HS [2]	OIR_RES_P_HS [1]	OIR_RES_P_HS [0]	
		-	-	-	-	-	OIR_RES_P_HW [10]	OIR_RES_P_HW [9]	OIR_RES_P_HW [8]	
		OIR_RES_P_HW [7]	OIR_RES_P_HW [6]	OIR_RES_P_HW [5]	OIR_RES_P_HW [4]	OIR_RES_P_HW [3]	OIR_RES_P_HW [2]	OIR_RES_P_HW [1]	OIR_RES_P_HW [0]	
	OIR_SCL0_US8	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	OIR_RES_IBUS_SYN C_SEL	-	-	-	-	
	OIR_SCL0_OVR1	-	-	-	-	-	-	-	-	
		OIR_RES_BK_COL_R[7]	OIR_RES_BK_COL_R[6]	OIR_RES_BK_COL_R[5]	OIR_RES_BK_COL_R[4]	OIR_RES_BK_COL_R[3]	OIR_RES_BK_COL_R[2]	OIR_RES_BK_COL_R[1]	OIR_RES_BK_COL_R[0]	
		OIR_RES_BK_COL_G[7]	OIR_RES_BK_COL_G[6]	OIR_RES_BK_COL_G[5]	OIR_RES_BK_COL_G[4]	OIR_RES_BK_COL_G[3]	OIR_RES_BK_COL_G[2]	OIR_RES_BK_COL_G[1]	OIR_RES_BK_COL_G[0]	
		OIR_RES_BK_COL_B[7]	OIR_RES_BK_COL_B[6]	OIR_RES_BK_COL_B[5]	OIR_RES_BK_COL_B[4]	OIR_RES_BK_COL_B[3]	OIR_RES_BK_COL_B[2]	OIR_RES_BK_COL_B[1]	OIR_RES_BK_COL_B[0]	
	OIR_SCL1_UPDATE	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	OIR_SCL1_UPDATE_A	
		-	-	-	-	-	-	-	-	
		-	-	-	OIR_SCL1_VEN_B	-	-	-	OIR_SCL1_VEN_A	
	OIR_SCL1_WR1	-	-	-	-	-	-	-	-	
		-	-	-	-	-	OIR_RES_WRSWA[2]	OIR_RES_WRSWA[1]	OIR_RES_WRSWA[0]	
		-	-	-	-	-	-	-	-	
		-	-	-	-	OIR_RES_MD[1]	OIR_RES_MD[0]	-	OIR_RES_BST_MD	
	OIR_SCL1_WR2	OIR_RES_BASE [31]	OIR_RES_BASE [30]	OIR_RES_BASE [29]	OIR_RES_BASE [28]	OIR_RES_BASE [27]	OIR_RES_BASE [26]	OIR_RES_BASE [25]	OIR_RES_BASE [24]	
		OIR_RES_BASE [23]	OIR_RES_BASE [22]	OIR_RES_BASE [21]	OIR_RES_BASE [20]	OIR_RES_BASE [19]	OIR_RES_BASE [18]	OIR_RES_BASE [17]	OIR_RES_BASE [16]	
		OIR_RES_BASE [15]	OIR_RES_BASE [14]	OIR_RES_BASE [13]	OIR_RES_BASE [12]	OIR_RES_BASE [11]	OIR_RES_BASE [10]	OIR_RES_BASE [9]	OIR_RES_BASE [8]	
		OIR_RES_BASE [7]	OIR_RES_BASE [6]	OIR_RES_BASE [5]	OIR_RES_BASE [4]	OIR_RES_BASE [3]	OIR_RES_BASE [2]	OIR_RES_BASE [1]	OIR_RES_BASE [0]	
	OIR_SCL1_WR3	-	OIR_RES_LN_OFF[14]	OIR_RES_LN_OFF[13]	OIR_RES_LN_OFF[12]	OIR_RES_LN_OFF[11]	OIR_RES_LN_OFF[10]	OIR_RES_LN_OFF[9]	OIR_RES_LN_OFF[8]	
		OIR_RES_LN_OFF[7]	OIR_RES_LN_OFF[6]	OIR_RES_LN_OFF[5]	OIR_RES_LN_OFF[4]	OIR_RES_LN_OFF[3]	OIR_RES_LN_OFF[2]	OIR_RES_LN_OFF[1]	OIR_RES_LN_OFF[0]	
		-	-	-	-	-	-	OIR_RES_FLM_NUM[9]	OIR_RES_FLM_NUM[8]	
		OIR_RES_FLM_NUM[7]	OIR_RES_FLM_NUM[6]	OIR_RES_FLM_NUM[5]	OIR_RES_FLM_NUM[4]	OIR_RES_FLM_NUM[3]	OIR_RES_FLM_NUM[2]	OIR_RES_FLM_NUM[1]	OIR_RES_FLM_NUM[0]	
	OIR_SCL1_WR4	-	-	-	-	-	-	-	-	
		-	OIR_RES_FLM_OFF[22]	OIR_RES_FLM_OFF[21]	OIR_RES_FLM_OFF[20]	OIR_RES_FLM_OFF[19]	OIR_RES_FLM_OFF[18]	OIR_RES_FLM_OFF[17]	OIR_RES_FLM_OFF[16]	
		OIR_RES_FLM_OFF[15]	OIR_RES_FLM_OFF[14]	OIR_RES_FLM_OFF[13]	OIR_RES_FLM_OFF[12]	OIR_RES_FLM_OFF[11]	OIR_RES_FLM_OFF[10]	OIR_RES_FLM_OFF[9]	OIR_RES_FLM_OFF[8]	
		OIR_RES_FLM_OFF[7]	OIR_RES_FLM_OFF[6]	OIR_RES_FLM_OFF[5]	OIR_RES_FLM_OFF[4]	OIR_RES_FLM_OFF[3]	OIR_RES_FLM_OFF[2]	OIR_RES_FLM_OFF[1]	OIR_RES_FLM_OFF[0]	
	OIR_SCL1_WR5	-	-	-	-	-	-	-	-	
		-	-	-	OIR_RES_INTER	-	-	-	-	
		-	-	-	-	-	-	OIR_RES_FS_RATE[1]	OIR_RES_FS_RATE[0]	
		-	-	-	-	-	-	-	OIR_RES_WENB	

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Video display controller 5 channel 0	OIR_SCL1_WR6	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	OIR_RES_DTH_ON	-	-	-	OIR_RES_BITDEC_ON
	OIR_SCL1_WR7	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	OIR_RES_OVERFLOW
		-	-	-	-	-	-	OIR_RES_FLM_CNT[9]	OIR_RES_FLM_CNT[8]
		OIR_RES_FLM_CNT[7]	OIR_RES_FLM_CNT[6]	OIR_RES_FLM_CNT[5]	OIR_RES_FLM_CNT[4]	OIR_RES_FLM_CNT[3]	OIR_RES_FLM_CNT[2]	OIR_RES_FLM_CNT[1]	OIR_RES_FLM_CNT[0]
	GR_OIR_UPDATE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GR_OIR_UPDATE
		-	-	-	GR_OIR_P_VEN	-	-	-	GR_OIR_IBUS_VEN
	GR_OIR_FLM_RD	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GR_OIR_R_ENB
	GR_OIR_FLM1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GR_OIR_LN_OFF_DIR
		-	-	-	-	-	-	GR_OIR_FLM_SEL[1]	GR_OIR_FLM_SEL[0]
		-	-	-	GR_OIR_IMR_FLM_INV	-	-	-	GR_OIR_BST_MD
	GR_OIR_FLM2	GR_OIR_BASE[31]	GR_OIR_BASE[30]	GR_OIR_BASE[29]	GR_OIR_BASE[28]	GR_OIR_BASE[27]	GR_OIR_BASE[26]	GR_OIR_BASE[25]	GR_OIR_BASE[24]
		GR_OIR_BASE[23]	GR_OIR_BASE[22]	GR_OIR_BASE[21]	GR_OIR_BASE[20]	GR_OIR_BASE[19]	GR_OIR_BASE[18]	GR_OIR_BASE[17]	GR_OIR_BASE[16]
		GR_OIR_BASE[15]	GR_OIR_BASE[14]	GR_OIR_BASE[13]	GR_OIR_BASE[12]	GR_OIR_BASE[11]	GR_OIR_BASE[10]	GR_OIR_BASE[9]	GR_OIR_BASE[8]
		GR_OIR_BASE[7]	GR_OIR_BASE[6]	GR_OIR_BASE[5]	GR_OIR_BASE[4]	GR_OIR_BASE[3]	GR_OIR_BASE[2]	GR_OIR_BASE[1]	GR_OIR_BASE[0]
	GR_OIR_FLM3	-	GR_OIR_LN_OFF[14]	GR_OIR_LN_OFF[13]	GR_OIR_LN_OFF[12]	GR_OIR_LN_OFF[11]	GR_OIR_LN_OFF[10]	GR_OIR_LN_OFF[9]	GR_OIR_LN_OFF[8]
		GR_OIR_LN_OFF[7]	GR_OIR_LN_OFF[6]	GR_OIR_LN_OFF[5]	GR_OIR_LN_OFF[4]	GR_OIR_LN_OFF[3]	GR_OIR_LN_OFF[2]	GR_OIR_LN_OFF[1]	GR_OIR_LN_OFF[0]
		-	-	-	-	-	-	GR_OIR_FLM_NUM[9]	GR_OIR_FLM_NUM[8]
		GR_OIR_FLM_NUM[7]	GR_OIR_FLM_NUM[6]	GR_OIR_FLM_NUM[5]	GR_OIR_FLM_NUM[4]	GR_OIR_FLM_NUM[3]	GR_OIR_FLM_NUM[2]	GR_OIR_FLM_NUM[1]	GR_OIR_FLM_NUM[0]
	GR_OIR_FLM4	-	-	-	-	-	-	-	-
		-	GR_OIR_FLM_OFF[22]	GR_OIR_FLM_OFF[21]	GR_OIR_FLM_OFF[20]	GR_OIR_FLM_OFF[19]	GR_OIR_FLM_OFF[18]	GR_OIR_FLM_OFF[17]	GR_OIR_FLM_OFF[16]
		GR_OIR_FLM_OFF[15]	GR_OIR_FLM_OFF[14]	GR_OIR_FLM_OFF[13]	GR_OIR_FLM_OFF[12]	GR_OIR_FLM_OFF[11]	GR_OIR_FLM_OFF[10]	GR_OIR_FLM_OFF[9]	GR_OIR_FLM_OFF[8]
		GR_OIR_FLM_OFF[7]	GR_OIR_FLM_OFF[6]	GR_OIR_FLM_OFF[5]	GR_OIR_FLM_OFF[4]	GR_OIR_FLM_OFF[3]	GR_OIR_FLM_OFF[2]	GR_OIR_FLM_OFF[1]	GR_OIR_FLM_OFF[0]
	GR_OIR_FLM5	-	-	-	-	-	GR_OIR_FLM_LNUM[10]	GR_OIR_FLM_LNUM[9]	GR_OIR_FLM_LNUM[8]
		GR_OIR_FLM_LNUM[7]	GR_OIR_FLM_LNUM[6]	GR_OIR_FLM_LNUM[5]	GR_OIR_FLM_LNUM[4]	GR_OIR_FLM_LNUM[3]	GR_OIR_FLM_LNUM[2]	GR_OIR_FLM_LNUM[1]	GR_OIR_FLM_LNUM[0]
		-	-	-	-	-	GR_OIR_FLM_LOOP[10]	GR_OIR_FLM_LOOP[9]	GR_OIR_FLM_LOOP[8]
		GR_OIR_FLM_LOOP[7]	GR_OIR_FLM_LOOP[6]	GR_OIR_FLM_LOOP[5]	GR_OIR_FLM_LOOP[4]	GR_OIR_FLM_LOOP[3]	GR_OIR_FLM_LOOP[2]	GR_OIR_FLM_LOOP[1]	GR_OIR_FLM_LOOP[0]
	GR_OIR_FLM6	GR_OIR_FORMAT[3]	GR_OIR_FORMAT[2]	GR_OIR_FORMAT[1]	GR_OIR_FORMAT[0]	-	GR_OIR_HW[10]	GR_OIR_HW[9]	GR_OIR_HW[8]
		GR_OIR_HW[7]	GR_OIR_HW[6]	GR_OIR_HW[5]	GR_OIR_HW[4]	GR_OIR_HW[3]	GR_OIR_HW[2]	GR_OIR_HW[1]	GR_OIR_HW[0]
		-	-	-	GR_OIR_RDSWA[2]	GR_OIR_RDSWA[1]	GR_OIR_RDSWA[0]	-	-
		-	-	GR_OIR_STA_POS[5]	GR_OIR_STA_POS[4]	GR_OIR_STA_POS[3]	GR_OIR_STA_POS[2]	GR_OIR_STA_POS[1]	GR_OIR_STA_POS[0]
	GR_OIR_AB1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	GR_OIR_GRC_DISP_ON	-	-	GR_OIR_DISP_SEL[1]	GR_OIR_DISP_SEL[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 0	GR_OIR_AB2	-	-	-	-	-	GR_OIR_GRC_VS[10]	GR_OIR_GRC_VS[9]	GR_OIR_GRC_VS[8]	
		GR_OIR_GRC_VS[7]	GR_OIR_GRC_VS[6]	GR_OIR_GRC_VS[5]	GR_OIR_GRC_VS[4]	GR_OIR_GRC_VS[3]	GR_OIR_GRC_VS[2]	GR_OIR_GRC_VS[1]	GR_OIR_GRC_VS[0]	
		-	-	-	-	-	GR_OIR_GRC_VW[10]	GR_OIR_GRC_VW[9]	GR_OIR_GRC_VW[8]	
		GR_OIR_GRC_VW[7]	GR_OIR_GRC_VW[6]	GR_OIR_GRC_VW[5]	GR_OIR_GRC_VW[4]	GR_OIR_GRC_VW[3]	GR_OIR_GRC_VW[2]	GR_OIR_GRC_VW[1]	GR_OIR_GRC_VW[0]	
	GR_OIR_AB3	-	-	-	-	-	-	GR_OIR_GRC_HS[10]	GR_OIR_GRC_HS[9]	GR_OIR_GRC_HS[8]
		GR_OIR_GRC_HS[7]	GR_OIR_GRC_HS[6]	GR_OIR_GRC_HS[5]	GR_OIR_GRC_HS[4]	GR_OIR_GRC_HS[3]	GR_OIR_GRC_HS[2]	GR_OIR_GRC_HS[1]	GR_OIR_GRC_HS[0]	
		-	-	-	-	-	GR_OIR_GRC_HW[10]	GR_OIR_GRC_HW[9]	GR_OIR_GRC_HW[8]	
		GR_OIR_GRC_HW[7]	GR_OIR_GRC_HW[6]	GR_OIR_GRC_HW[5]	GR_OIR_GRC_HW[4]	GR_OIR_GRC_HW[3]	GR_OIR_GRC_HW[2]	GR_OIR_GRC_HW[1]	GR_OIR_GRC_HW[0]	
	GR_OIR_AB7	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	GR_OIR_CK_ON
	GR_OIR_AB8	GR_OIR_CK_KCLUT[7]	GR_OIR_CK_KCLUT[6]	GR_OIR_CK_KCLUT[5]	GR_OIR_CK_KCLUT[4]	GR_OIR_CK_KCLUT[3]	GR_OIR_CK_KCLUT[2]	GR_OIR_CK_KCLUT[1]	GR_OIR_CK_KCLUT[0]	
		GR_OIR_CK_KG[7]	GR_OIR_CK_KG[6]	GR_OIR_CK_KG[5]	GR_OIR_CK_KG[4]	GR_OIR_CK_KG[3]	GR_OIR_CK_KG[2]	GR_OIR_CK_KG[1]	GR_OIR_CK_KG[0]	
		GR_OIR_CK_KB[7]	GR_OIR_CK_KB[6]	GR_OIR_CK_KB[5]	GR_OIR_CK_KB[4]	GR_OIR_CK_KB[3]	GR_OIR_CK_KB[2]	GR_OIR_CK_KB[1]	GR_OIR_CK_KB[0]	
		GR_OIR_CK_KR[7]	GR_OIR_CK_KR[6]	GR_OIR_CK_KR[5]	GR_OIR_CK_KR[4]	GR_OIR_CK_KR[3]	GR_OIR_CK_KR[2]	GR_OIR_CK_KR[1]	GR_OIR_CK_KR[0]	
	GR_OIR_AB9	GR_OIR_CK_A[7]	GR_OIR_CK_A[6]	GR_OIR_CK_A[5]	GR_OIR_CK_A[4]	GR_OIR_CK_A[3]	GR_OIR_CK_A[2]	GR_OIR_CK_A[1]	GR_OIR_CK_A[0]	
		GR_OIR_CK_G[7]	GR_OIR_CK_G[6]	GR_OIR_CK_G[5]	GR_OIR_CK_G[4]	GR_OIR_CK_G[3]	GR_OIR_CK_G[2]	GR_OIR_CK_G[1]	GR_OIR_CK_G[0]	
		GR_OIR_CK_B[7]	GR_OIR_CK_B[6]	GR_OIR_CK_B[5]	GR_OIR_CK_B[4]	GR_OIR_CK_B[3]	GR_OIR_CK_B[2]	GR_OIR_CK_B[1]	GR_OIR_CK_B[0]	
		GR_OIR_CK_R[7]	GR_OIR_CK_R[6]	GR_OIR_CK_R[5]	GR_OIR_CK_R[4]	GR_OIR_CK_R[3]	GR_OIR_CK_R[2]	GR_OIR_CK_R[1]	GR_OIR_CK_R[0]	
	GR_OIR_AB10	GR_OIR_A0[7]	GR_OIR_A0[6]	GR_OIR_A0[5]	GR_OIR_A0[4]	GR_OIR_A0[3]	GR_OIR_A0[2]	GR_OIR_A0[1]	GR_OIR_A0[0]	
		GR_OIR_G0[7]	GR_OIR_G0[6]	GR_OIR_G0[5]	GR_OIR_G0[4]	GR_OIR_G0[3]	GR_OIR_G0[2]	GR_OIR_G0[1]	GR_OIR_G0[0]	
		GR_OIR_B0[7]	GR_OIR_B0[6]	GR_OIR_B0[5]	GR_OIR_B0[4]	GR_OIR_B0[3]	GR_OIR_B0[2]	GR_OIR_B0[1]	GR_OIR_B0[0]	
		GR_OIR_R0[7]	GR_OIR_R0[6]	GR_OIR_R0[5]	GR_OIR_R0[4]	GR_OIR_R0[3]	GR_OIR_R0[2]	GR_OIR_R0[1]	GR_OIR_R0[0]	
	GR_OIR_AB11	GR_OIR_A1[7]	GR_OIR_A1[6]	GR_OIR_A1[5]	GR_OIR_A1[4]	GR_OIR_A1[3]	GR_OIR_A1[2]	GR_OIR_A1[1]	GR_OIR_A1[0]	
		GR_OIR_G1[7]	GR_OIR_G1[6]	GR_OIR_G1[5]	GR_OIR_G1[4]	GR_OIR_G1[3]	GR_OIR_G1[2]	GR_OIR_G1[1]	GR_OIR_G1[0]	
		GR_OIR_B1[7]	GR_OIR_B1[6]	GR_OIR_B1[5]	GR_OIR_B1[4]	GR_OIR_B1[3]	GR_OIR_B1[2]	GR_OIR_B1[1]	GR_OIR_B1[0]	
		GR_OIR_R1[7]	GR_OIR_R1[6]	GR_OIR_R1[5]	GR_OIR_R1[4]	GR_OIR_R1[3]	GR_OIR_R1[2]	GR_OIR_R1[1]	GR_OIR_R1[0]	
	GR_OIR_BASE	-	-	-	-	-	-	-	-	
		GR_OIR_BASE_G[7]	GR_OIR_BASE_G[6]	GR_OIR_BASE_G[5]	GR_OIR_BASE_G[4]	GR_OIR_BASE_G[3]	GR_OIR_BASE_G[2]	GR_OIR_BASE_G[1]	GR_OIR_BASE_G[0]	
		GR_OIR_BASE_B[7]	GR_OIR_BASE_B[6]	GR_OIR_BASE_B[5]	GR_OIR_BASE_B[4]	GR_OIR_BASE_B[3]	GR_OIR_BASE_B[2]	GR_OIR_BASE_B[1]	GR_OIR_BASE_B[0]	
		GR_OIR_BASE_R[7]	GR_OIR_BASE_R[6]	GR_OIR_BASE_R[5]	GR_OIR_BASE_R[4]	GR_OIR_BASE_R[3]	GR_OIR_BASE_R[2]	GR_OIR_BASE_R[1]	GR_OIR_BASE_R[0]	
	GR_OIR_CLUT	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	GR_OIR_CLT_SEL	
		-	-	-	-	-	GR_OIR_LINE[10]	GR_OIR_LINE[9]	GR_OIR_LINE[8]	
		GR_OIR_LINE[7]	GR_OIR_LINE[6]	GR_OIR_LINE[5]	GR_OIR_LINE[4]	GR_OIR_LINE[3]	GR_OIR_LINE[2]	GR_OIR_LINE[1]	GR_OIR_LINE[0]	
	GR_OIR_MON	-	-	-	-	-	OIR_LIN_STAT[10]	OIR_LIN_STAT[9]	OIR_LIN_STAT[8]	
		OIR_LIN_STAT[7]	OIR_LIN_STAT[6]	OIR_LIN_STAT[5]	OIR_LIN_STAT[4]	OIR_LIN_STAT[3]	OIR_LIN_STAT[2]	OIR_LIN_STAT[1]	OIR_LIN_STAT[0]	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
	GAM_G_UPDATE	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	GAM_G_VEN	
GAM_SW	-	-	-	-	-	-	-	-		
	-	-	-	-	-	-	-	-		
	-	-	-	-	-	-	-	-		
	-	-	-	-	-	-	-	GAM_ON		



Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 0	GAM_G_LUT1	-	-	-	-	-	GAM_G_GAIN_00[10]	GAM_G_GAIN_00[9]	GAM_G_GAIN_00[8]	
		GAM_G_GAIN_00[7]	GAM_G_GAIN_00[6]	GAM_G_GAIN_00[5]	GAM_G_GAIN_00[4]	GAM_G_GAIN_00[3]	GAM_G_GAIN_00[2]	GAM_G_GAIN_00[1]	GAM_G_GAIN_00[0]	
		-	-	-	-	-	GAM_G_GAIN_01[10]	GAM_G_GAIN_01[9]	GAM_G_GAIN_01[8]	
		GAM_G_GAIN_01[7]	GAM_G_GAIN_01[6]	GAM_G_GAIN_01[5]	GAM_G_GAIN_01[4]	GAM_G_GAIN_01[3]	GAM_G_GAIN_01[2]	GAM_G_GAIN_01[1]	GAM_G_GAIN_01[0]	
	GAM_G_LUT2	-	-	-	-	-	-	GAM_G_GAIN_02[10]	GAM_G_GAIN_02[9]	GAM_G_GAIN_02[8]
		GAM_G_GAIN_02[7]	GAM_G_GAIN_02[6]	GAM_G_GAIN_02[5]	GAM_G_GAIN_02[4]	GAM_G_GAIN_02[3]	GAM_G_GAIN_02[2]	GAM_G_GAIN_02[1]	GAM_G_GAIN_02[0]	
		-	-	-	-	-	-	GAM_G_GAIN_03[10]	GAM_G_GAIN_03[9]	GAM_G_GAIN_03[8]
		GAM_G_GAIN_03[7]	GAM_G_GAIN_03[6]	GAM_G_GAIN_03[5]	GAM_G_GAIN_03[4]	GAM_G_GAIN_03[3]	GAM_G_GAIN_03[2]	GAM_G_GAIN_03[1]	GAM_G_GAIN_03[0]	
	GAM_G_LUT3	-	-	-	-	-	-	GAM_G_GAIN_04[10]	GAM_G_GAIN_04[9]	GAM_G_GAIN_04[8]
		GAM_G_GAIN_04[7]	GAM_G_GAIN_04[6]	GAM_G_GAIN_04[5]	GAM_G_GAIN_04[4]	GAM_G_GAIN_04[3]	GAM_G_GAIN_04[2]	GAM_G_GAIN_04[1]	GAM_G_GAIN_04[0]	
		-	-	-	-	-	-	GAM_G_GAIN_05[10]	GAM_G_GAIN_05[9]	GAM_G_GAIN_05[8]
		GAM_G_GAIN_05[7]	GAM_G_GAIN_05[6]	GAM_G_GAIN_05[5]	GAM_G_GAIN_05[4]	GAM_G_GAIN_05[3]	GAM_G_GAIN_05[2]	GAM_G_GAIN_05[1]	GAM_G_GAIN_05[0]	
	GAM_G_LUT4	-	-	-	-	-	-	GAM_G_GAIN_06[10]	GAM_G_GAIN_06[9]	GAM_G_GAIN_06[8]
		GAM_G_GAIN_06[7]	GAM_G_GAIN_06[6]	GAM_G_GAIN_06[5]	GAM_G_GAIN_06[4]	GAM_G_GAIN_06[3]	GAM_G_GAIN_06[2]	GAM_G_GAIN_06[1]	GAM_G_GAIN_06[0]	
		-	-	-	-	-	-	GAM_G_GAIN_07[10]	GAM_G_GAIN_07[9]	GAM_G_GAIN_07[8]
		GAM_G_GAIN_07[7]	GAM_G_GAIN_07[6]	GAM_G_GAIN_07[5]	GAM_G_GAIN_07[4]	GAM_G_GAIN_07[3]	GAM_G_GAIN_07[2]	GAM_G_GAIN_07[1]	GAM_G_GAIN_07[0]	
	GAM_G_LUT5	-	-	-	-	-	-	GAM_G_GAIN_08[10]	GAM_G_GAIN_08[9]	GAM_G_GAIN_08[8]
		GAM_G_GAIN_08[7]	GAM_G_GAIN_08[6]	GAM_G_GAIN_08[5]	GAM_G_GAIN_08[4]	GAM_G_GAIN_08[3]	GAM_G_GAIN_08[2]	GAM_G_GAIN_08[1]	GAM_G_GAIN_08[0]	
		-	-	-	-	-	-	GAM_G_GAIN_09[10]	GAM_G_GAIN_09[9]	GAM_G_GAIN_09[8]
		GAM_G_GAIN_09[7]	GAM_G_GAIN_09[6]	GAM_G_GAIN_09[5]	GAM_G_GAIN_09[4]	GAM_G_GAIN_09[3]	GAM_G_GAIN_09[2]	GAM_G_GAIN_09[1]	GAM_G_GAIN_09[0]	
	GAM_G_LUT6	-	-	-	-	-	-	GAM_G_GAIN_10[10]	GAM_G_GAIN_10[9]	GAM_G_GAIN_10[8]
		GAM_G_GAIN_10[7]	GAM_G_GAIN_10[6]	GAM_G_GAIN_10[5]	GAM_G_GAIN_10[4]	GAM_G_GAIN_10[3]	GAM_G_GAIN_10[2]	GAM_G_GAIN_10[1]	GAM_G_GAIN_10[0]	
		-	-	-	-	-	-	GAM_G_GAIN_11[10]	GAM_G_GAIN_11[9]	GAM_G_GAIN_11[8]
		GAM_G_GAIN_11[7]	GAM_G_GAIN_11[6]	GAM_G_GAIN_11[5]	GAM_G_GAIN_11[4]	GAM_G_GAIN_11[3]	GAM_G_GAIN_11[2]	GAM_G_GAIN_11[1]	GAM_G_GAIN_11[0]	
	GAM_G_LUT7	-	-	-	-	-	-	GAM_G_GAIN_12[10]	GAM_G_GAIN_12[9]	GAM_G_GAIN_12[8]
		GAM_G_GAIN_12[7]	GAM_G_GAIN_12[6]	GAM_G_GAIN_12[5]	GAM_G_GAIN_12[4]	GAM_G_GAIN_12[3]	GAM_G_GAIN_12[2]	GAM_G_GAIN_12[1]	GAM_G_GAIN_12[0]	
		-	-	-	-	-	-	GAM_G_GAIN_13[10]	GAM_G_GAIN_13[9]	GAM_G_GAIN_13[8]
		GAM_G_GAIN_13[7]	GAM_G_GAIN_13[6]	GAM_G_GAIN_13[5]	GAM_G_GAIN_13[4]	GAM_G_GAIN_13[3]	GAM_G_GAIN_13[2]	GAM_G_GAIN_13[1]	GAM_G_GAIN_13[0]	
	GAM_G_LUT8	-	-	-	-	-	-	GAM_G_GAIN_14[10]	GAM_G_GAIN_14[9]	GAM_G_GAIN_14[8]
		GAM_G_GAIN_14[7]	GAM_G_GAIN_14[6]	GAM_G_GAIN_14[5]	GAM_G_GAIN_14[4]	GAM_G_GAIN_14[3]	GAM_G_GAIN_14[2]	GAM_G_GAIN_14[1]	GAM_G_GAIN_14[0]	
		-	-	-	-	-	-	GAM_G_GAIN_15[10]	GAM_G_GAIN_15[9]	GAM_G_GAIN_15[8]
		GAM_G_GAIN_15[7]	GAM_G_GAIN_15[6]	GAM_G_GAIN_15[5]	GAM_G_GAIN_15[4]	GAM_G_GAIN_15[3]	GAM_G_GAIN_15[2]	GAM_G_GAIN_15[1]	GAM_G_GAIN_15[0]	
	GAM_G_LUT9	-	-	-	-	-	-	GAM_G_GAIN_16[10]	GAM_G_GAIN_16[9]	GAM_G_GAIN_16[8]
		GAM_G_GAIN_16[7]	GAM_G_GAIN_16[6]	GAM_G_GAIN_16[5]	GAM_G_GAIN_16[4]	GAM_G_GAIN_16[3]	GAM_G_GAIN_16[2]	GAM_G_GAIN_16[1]	GAM_G_GAIN_16[0]	
		-	-	-	-	-	-	GAM_G_GAIN_17[10]	GAM_G_GAIN_17[9]	GAM_G_GAIN_17[8]
		GAM_G_GAIN_17[7]	GAM_G_GAIN_17[6]	GAM_G_GAIN_17[5]	GAM_G_GAIN_17[4]	GAM_G_GAIN_17[3]	GAM_G_GAIN_17[2]	GAM_G_GAIN_17[1]	GAM_G_GAIN_17[0]	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 0	GAM_G_LUT10	-	-	-	-	-	GAM_G_GAIN_18[10]	GAM_G_GAIN_18[9]	GAM_G_GAIN_18[8]	
		GAM_G_GAIN_18[7]	GAM_G_GAIN_18[6]	GAM_G_GAIN_18[5]	GAM_G_GAIN_18[4]	GAM_G_GAIN_18[3]	GAM_G_GAIN_18[2]	GAM_G_GAIN_18[1]	GAM_G_GAIN_18[0]	
		-	-	-	-	-	GAM_G_GAIN_19[10]	GAM_G_GAIN_19[9]	GAM_G_GAIN_19[8]	
		GAM_G_GAIN_19[7]	GAM_G_GAIN_19[6]	GAM_G_GAIN_19[5]	GAM_G_GAIN_19[4]	GAM_G_GAIN_19[3]	GAM_G_GAIN_19[2]	GAM_G_GAIN_19[1]	GAM_G_GAIN_19[0]	
	GAM_G_LUT11	-	-	-	-	-	-	GAM_G_GAIN_20[10]	GAM_G_GAIN_20[9]	GAM_G_GAIN_20[8]
		GAM_G_GAIN_20[7]	GAM_G_GAIN_20[6]	GAM_G_GAIN_20[5]	GAM_G_GAIN_20[4]	GAM_G_GAIN_20[3]	GAM_G_GAIN_20[2]	GAM_G_GAIN_20[1]	GAM_G_GAIN_20[0]	
		-	-	-	-	-	GAM_G_GAIN_21[10]	GAM_G_GAIN_21[9]	GAM_G_GAIN_21[8]	
		GAM_G_GAIN_21[7]	GAM_G_GAIN_21[6]	GAM_G_GAIN_21[5]	GAM_G_GAIN_21[4]	GAM_G_GAIN_21[3]	GAM_G_GAIN_21[2]	GAM_G_GAIN_21[1]	GAM_G_GAIN_21[0]	
	GAM_G_LUT12	-	-	-	-	-	-	GAM_G_GAIN_22[10]	GAM_G_GAIN_22[9]	GAM_G_GAIN_22[8]
		GAM_G_GAIN_22[7]	GAM_G_GAIN_22[6]	GAM_G_GAIN_22[5]	GAM_G_GAIN_22[4]	GAM_G_GAIN_22[3]	GAM_G_GAIN_22[2]	GAM_G_GAIN_22[1]	GAM_G_GAIN_22[0]	
		-	-	-	-	-	GAM_G_GAIN_23[10]	GAM_G_GAIN_23[9]	GAM_G_GAIN_23[8]	
		GAM_G_GAIN_23[7]	GAM_G_GAIN_23[6]	GAM_G_GAIN_23[5]	GAM_G_GAIN_23[4]	GAM_G_GAIN_23[3]	GAM_G_GAIN_23[2]	GAM_G_GAIN_23[1]	GAM_G_GAIN_23[0]	
	GAM_G_LUT13	-	-	-	-	-	-	GAM_G_GAIN_24[10]	GAM_G_GAIN_24[9]	GAM_G_GAIN_24[8]
		GAM_G_GAIN_24[7]	GAM_G_GAIN_24[6]	GAM_G_GAIN_24[5]	GAM_G_GAIN_24[4]	GAM_G_GAIN_24[3]	GAM_G_GAIN_24[2]	GAM_G_GAIN_24[1]	GAM_G_GAIN_24[0]	
		-	-	-	-	-	GAM_G_GAIN_25[10]	GAM_G_GAIN_25[9]	GAM_G_GAIN_25[8]	
		GAM_G_GAIN_25[7]	GAM_G_GAIN_25[6]	GAM_G_GAIN_25[5]	GAM_G_GAIN_25[4]	GAM_G_GAIN_25[3]	GAM_G_GAIN_25[2]	GAM_G_GAIN_25[1]	GAM_G_GAIN_25[0]	
	GAM_G_LUT14	-	-	-	-	-	-	GAM_G_GAIN_26[10]	GAM_G_GAIN_26[9]	GAM_G_GAIN_26[8]
		GAM_G_GAIN_26[7]	GAM_G_GAIN_26[6]	GAM_G_GAIN_26[5]	GAM_G_GAIN_26[4]	GAM_G_GAIN_26[3]	GAM_G_GAIN_26[2]	GAM_G_GAIN_26[1]	GAM_G_GAIN_26[0]	
		-	-	-	-	-	GAM_G_GAIN_27[10]	GAM_G_GAIN_27[9]	GAM_G_GAIN_27[8]	
		GAM_G_GAIN_27[7]	GAM_G_GAIN_27[6]	GAM_G_GAIN_27[5]	GAM_G_GAIN_27[4]	GAM_G_GAIN_27[3]	GAM_G_GAIN_27[2]	GAM_G_GAIN_27[1]	GAM_G_GAIN_27[0]	
	GAM_G_LUT15	-	-	-	-	-	-	GAM_G_GAIN_28[10]	GAM_G_GAIN_28[9]	GAM_G_GAIN_28[8]
		GAM_G_GAIN_28[7]	GAM_G_GAIN_28[6]	GAM_G_GAIN_28[5]	GAM_G_GAIN_28[4]	GAM_G_GAIN_28[3]	GAM_G_GAIN_28[2]	GAM_G_GAIN_28[1]	GAM_G_GAIN_28[0]	
		-	-	-	-	-	GAM_G_GAIN_29[10]	GAM_G_GAIN_29[9]	GAM_G_GAIN_29[8]	
		GAM_G_GAIN_29[7]	GAM_G_GAIN_29[6]	GAM_G_GAIN_29[5]	GAM_G_GAIN_29[4]	GAM_G_GAIN_29[3]	GAM_G_GAIN_29[2]	GAM_G_GAIN_29[1]	GAM_G_GAIN_29[0]	
	GAM_G_LUT16	-	-	-	-	-	-	GAM_G_GAIN_30[10]	GAM_G_GAIN_30[9]	GAM_G_GAIN_30[8]
		GAM_G_GAIN_30[7]	GAM_G_GAIN_30[6]	GAM_G_GAIN_30[5]	GAM_G_GAIN_30[4]	GAM_G_GAIN_30[3]	GAM_G_GAIN_30[2]	GAM_G_GAIN_30[1]	GAM_G_GAIN_30[0]	
		-	-	-	-	-	GAM_G_GAIN_31[10]	GAM_G_GAIN_31[9]	GAM_G_GAIN_31[8]	
		GAM_G_GAIN_31[7]	GAM_G_GAIN_31[6]	GAM_G_GAIN_31[5]	GAM_G_GAIN_31[4]	GAM_G_GAIN_31[3]	GAM_G_GAIN_31[2]	GAM_G_GAIN_31[1]	GAM_G_GAIN_31[0]	
	GAM_G_AREA1	-	-	-	-	-	-	-	-	-
		GAM_G_TH_01[7]	GAM_G_TH_01[6]	GAM_G_TH_01[5]	GAM_G_TH_01[4]	GAM_G_TH_01[3]	GAM_G_TH_01[2]	GAM_G_TH_01[1]	GAM_G_TH_01[0]	
		GAM_G_TH_02[7]	GAM_G_TH_02[6]	GAM_G_TH_02[5]	GAM_G_TH_02[4]	GAM_G_TH_02[3]	GAM_G_TH_02[2]	GAM_G_TH_02[1]	GAM_G_TH_02[0]	
	GAM_G_AREA2	GAM_G_TH_03[7]	GAM_G_TH_03[6]	GAM_G_TH_03[5]	GAM_G_TH_03[4]	GAM_G_TH_03[3]	GAM_G_TH_03[2]	GAM_G_TH_03[1]	GAM_G_TH_03[0]	
		GAM_G_TH_04[7]	GAM_G_TH_04[6]	GAM_G_TH_04[5]	GAM_G_TH_04[4]	GAM_G_TH_04[3]	GAM_G_TH_04[2]	GAM_G_TH_04[1]	GAM_G_TH_04[0]	
		GAM_G_TH_05[7]	GAM_G_TH_05[6]	GAM_G_TH_05[5]	GAM_G_TH_05[4]	GAM_G_TH_05[3]	GAM_G_TH_05[2]	GAM_G_TH_05[1]	GAM_G_TH_05[0]	
		GAM_G_TH_06[7]	GAM_G_TH_06[6]	GAM_G_TH_06[5]	GAM_G_TH_06[4]	GAM_G_TH_06[3]	GAM_G_TH_06[2]	GAM_G_TH_06[1]	GAM_G_TH_06[0]	
	GAM_G_AREA3	GAM_G_TH_07[7]	GAM_G_TH_07[6]	GAM_G_TH_07[5]	GAM_G_TH_07[4]	GAM_G_TH_07[3]	GAM_G_TH_07[2]	GAM_G_TH_07[1]	GAM_G_TH_07[0]	
		GAM_G_TH_08[7]	GAM_G_TH_08[6]	GAM_G_TH_08[5]	GAM_G_TH_08[4]	GAM_G_TH_08[3]	GAM_G_TH_08[2]	GAM_G_TH_08[1]	GAM_G_TH_08[0]	
		GAM_G_TH_09[7]	GAM_G_TH_09[6]	GAM_G_TH_09[5]	GAM_G_TH_09[4]	GAM_G_TH_09[3]	GAM_G_TH_09[2]	GAM_G_TH_09[1]	GAM_G_TH_09[0]	
		GAM_G_TH_10[7]	GAM_G_TH_10[6]	GAM_G_TH_10[5]	GAM_G_TH_10[4]	GAM_G_TH_10[3]	GAM_G_TH_10[2]	GAM_G_TH_10[1]	GAM_G_TH_10[0]	
		GAM_G_TH_11[7]	GAM_G_TH_11[6]	GAM_G_TH_11[5]	GAM_G_TH_11[4]	GAM_G_TH_11[3]	GAM_G_TH_11[2]	GAM_G_TH_11[1]	GAM_G_TH_11[0]	

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Video display controller 5 channel 0	GAM_G_AREA4	GAM_G_TH_12[7]	GAM_G_TH_12[6]	GAM_G_TH_12[5]	GAM_G_TH_12[4]	GAM_G_TH_12[3]	GAM_G_TH_12[2]	GAM_G_TH_12[1]	GAM_G_TH_12[0]
		GAM_G_TH_13[7]	GAM_G_TH_13[6]	GAM_G_TH_13[5]	GAM_G_TH_13[4]	GAM_G_TH_13[3]	GAM_G_TH_13[2]	GAM_G_TH_13[1]	GAM_G_TH_13[0]
		GAM_G_TH_14[7]	GAM_G_TH_14[6]	GAM_G_TH_14[5]	GAM_G_TH_14[4]	GAM_G_TH_14[3]	GAM_G_TH_14[2]	GAM_G_TH_14[1]	GAM_G_TH_14[0]
		GAM_G_TH_15[7]	GAM_G_TH_15[6]	GAM_G_TH_15[5]	GAM_G_TH_15[4]	GAM_G_TH_15[3]	GAM_G_TH_15[2]	GAM_G_TH_15[1]	GAM_G_TH_15[0]
	GAM_G_AREA5	GAM_G_TH_16[7]	GAM_G_TH_16[6]	GAM_G_TH_16[5]	GAM_G_TH_16[4]	GAM_G_TH_16[3]	GAM_G_TH_16[2]	GAM_G_TH_16[1]	GAM_G_TH_16[0]
		GAM_G_TH_17[7]	GAM_G_TH_17[6]	GAM_G_TH_17[5]	GAM_G_TH_17[4]	GAM_G_TH_17[3]	GAM_G_TH_17[2]	GAM_G_TH_17[1]	GAM_G_TH_17[0]
		GAM_G_TH_18[7]	GAM_G_TH_18[6]	GAM_G_TH_18[5]	GAM_G_TH_18[4]	GAM_G_TH_18[3]	GAM_G_TH_18[2]	GAM_G_TH_18[1]	GAM_G_TH_18[0]
		GAM_G_TH_19[7]	GAM_G_TH_19[6]	GAM_G_TH_19[5]	GAM_G_TH_19[4]	GAM_G_TH_19[3]	GAM_G_TH_19[2]	GAM_G_TH_19[1]	GAM_G_TH_19[0]
	GAM_G_AREA6	GAM_G_TH_20[7]	GAM_G_TH_20[6]	GAM_G_TH_20[5]	GAM_G_TH_20[4]	GAM_G_TH_20[3]	GAM_G_TH_20[2]	GAM_G_TH_20[1]	GAM_G_TH_20[0]
		GAM_G_TH_21[7]	GAM_G_TH_21[6]	GAM_G_TH_21[5]	GAM_G_TH_21[4]	GAM_G_TH_21[3]	GAM_G_TH_21[2]	GAM_G_TH_21[1]	GAM_G_TH_21[0]
		GAM_G_TH_22[7]	GAM_G_TH_22[6]	GAM_G_TH_22[5]	GAM_G_TH_22[4]	GAM_G_TH_22[3]	GAM_G_TH_22[2]	GAM_G_TH_22[1]	GAM_G_TH_22[0]
		GAM_G_TH_23[7]	GAM_G_TH_23[6]	GAM_G_TH_23[5]	GAM_G_TH_23[4]	GAM_G_TH_23[3]	GAM_G_TH_23[2]	GAM_G_TH_23[1]	GAM_G_TH_23[0]
	GAM_G_AREA7	GAM_G_TH_24[7]	GAM_G_TH_24[6]	GAM_G_TH_24[5]	GAM_G_TH_24[4]	GAM_G_TH_24[3]	GAM_G_TH_24[2]	GAM_G_TH_24[1]	GAM_G_TH_24[0]
		GAM_G_TH_25[7]	GAM_G_TH_25[6]	GAM_G_TH_25[5]	GAM_G_TH_25[4]	GAM_G_TH_25[3]	GAM_G_TH_25[2]	GAM_G_TH_25[1]	GAM_G_TH_25[0]
		GAM_G_TH_26[7]	GAM_G_TH_26[6]	GAM_G_TH_26[5]	GAM_G_TH_26[4]	GAM_G_TH_26[3]	GAM_G_TH_26[2]	GAM_G_TH_26[1]	GAM_G_TH_26[0]
		GAM_G_TH_27[7]	GAM_G_TH_27[6]	GAM_G_TH_27[5]	GAM_G_TH_27[4]	GAM_G_TH_27[3]	GAM_G_TH_27[2]	GAM_G_TH_27[1]	GAM_G_TH_27[0]
	GAM_G_AREA8	GAM_G_TH_28[7]	GAM_G_TH_28[6]	GAM_G_TH_28[5]	GAM_G_TH_28[4]	GAM_G_TH_28[3]	GAM_G_TH_28[2]	GAM_G_TH_28[1]	GAM_G_TH_28[0]
		GAM_G_TH_29[7]	GAM_G_TH_29[6]	GAM_G_TH_29[5]	GAM_G_TH_29[4]	GAM_G_TH_29[3]	GAM_G_TH_29[2]	GAM_G_TH_29[1]	GAM_G_TH_29[0]
		GAM_G_TH_30[7]	GAM_G_TH_30[6]	GAM_G_TH_30[5]	GAM_G_TH_30[4]	GAM_G_TH_30[3]	GAM_G_TH_30[2]	GAM_G_TH_30[1]	GAM_G_TH_30[0]
		GAM_G_TH_31[7]	GAM_G_TH_31[6]	GAM_G_TH_31[5]	GAM_G_TH_31[4]	GAM_G_TH_31[3]	GAM_G_TH_31[2]	GAM_G_TH_31[1]	GAM_G_TH_31[0]
	GAM_B_UPDATE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GAM_B_VEN
	GAM_B_LUT1	-	-	-	-	-	GAM_B_GAIN_00[10]	GAM_B_GAIN_00[9]	GAM_B_GAIN_00[8]
		GAM_B_GAIN_00[7]	GAM_B_GAIN_00[6]	GAM_B_GAIN_00[5]	GAM_B_GAIN_00[4]	GAM_B_GAIN_00[3]	GAM_B_GAIN_00[2]	GAM_B_GAIN_00[1]	GAM_B_GAIN_00[0]
		-	-	-	-	-	GAM_B_GAIN_01[10]	GAM_B_GAIN_01[9]	GAM_B_GAIN_01[8]
		GAM_B_GAIN_01[7]	GAM_B_GAIN_01[6]	GAM_B_GAIN_01[5]	GAM_B_GAIN_01[4]	GAM_B_GAIN_01[3]	GAM_B_GAIN_01[2]	GAM_B_GAIN_01[1]	GAM_B_GAIN_01[0]
	GAM_B_LUT2	-	-	-	-	-	GAM_B_GAIN_02[10]	GAM_B_GAIN_02[9]	GAM_B_GAIN_02[8]
		GAM_B_GAIN_02[7]	GAM_B_GAIN_02[6]	GAM_B_GAIN_02[5]	GAM_B_GAIN_02[4]	GAM_B_GAIN_02[3]	GAM_B_GAIN_02[2]	GAM_B_GAIN_02[1]	GAM_B_GAIN_02[0]
		-	-	-	-	-	GAM_B_GAIN_03[10]	GAM_B_GAIN_03[9]	GAM_B_GAIN_03[8]
		GAM_B_GAIN_03[7]	GAM_B_GAIN_03[6]	GAM_B_GAIN_03[5]	GAM_B_GAIN_03[4]	GAM_B_GAIN_03[3]	GAM_B_GAIN_03[2]	GAM_B_GAIN_03[1]	GAM_B_GAIN_03[0]
	GAM_B_LUT3	-	-	-	-	-	GAM_B_GAIN_04[10]	GAM_B_GAIN_04[9]	GAM_B_GAIN_04[8]
		GAM_B_GAIN_04[7]	GAM_B_GAIN_04[6]	GAM_B_GAIN_04[5]	GAM_B_GAIN_04[4]	GAM_B_GAIN_04[3]	GAM_B_GAIN_04[2]	GAM_B_GAIN_04[1]	GAM_B_GAIN_04[0]
		-	-	-	-	-	GAM_B_GAIN_05[10]	GAM_B_GAIN_05[9]	GAM_B_GAIN_05[8]
		GAM_B_GAIN_05[7]	GAM_B_GAIN_05[6]	GAM_B_GAIN_05[5]	GAM_B_GAIN_05[4]	GAM_B_GAIN_05[3]	GAM_B_GAIN_05[2]	GAM_B_GAIN_05[1]	GAM_B_GAIN_05[0]
	GAM_B_LUT4	-	-	-	-	-	GAM_B_GAIN_06[10]	GAM_B_GAIN_06[9]	GAM_B_GAIN_06[8]
		GAM_B_GAIN_06[7]	GAM_B_GAIN_06[6]	GAM_B_GAIN_06[5]	GAM_B_GAIN_06[4]	GAM_B_GAIN_06[3]	GAM_B_GAIN_06[2]	GAM_B_GAIN_06[1]	GAM_B_GAIN_06[0]
		-	-	-	-	-	GAM_B_GAIN_07[10]	GAM_B_GAIN_07[9]	GAM_B_GAIN_07[8]
		GAM_B_GAIN_07[7]	GAM_B_GAIN_07[6]	GAM_B_GAIN_07[5]	GAM_B_GAIN_07[4]	GAM_B_GAIN_07[3]	GAM_B_GAIN_07[2]	GAM_B_GAIN_07[1]	GAM_B_GAIN_07[0]
GAM_B_LUT5	-	-	-	-	-	GAM_B_GAIN_08[10]	GAM_B_GAIN_08[9]	GAM_B_GAIN_08[8]	
	GAM_B_GAIN_08[7]	GAM_B_GAIN_08[6]	GAM_B_GAIN_08[5]	GAM_B_GAIN_08[4]	GAM_B_GAIN_08[3]	GAM_B_GAIN_08[2]	GAM_B_GAIN_08[1]	GAM_B_GAIN_08[0]	
	-	-	-	-	-	GAM_B_GAIN_09[10]	GAM_B_GAIN_09[9]	GAM_B_GAIN_09[8]	
	GAM_B_GAIN_09[7]	GAM_B_GAIN_09[6]	GAM_B_GAIN_09[5]	GAM_B_GAIN_09[4]	GAM_B_GAIN_09[3]	GAM_B_GAIN_09[2]	GAM_B_GAIN_09[1]	GAM_B_GAIN_09[0]	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Video display controller 5 channel 0	GAM_B_LUT6	-	-	-	-	-	GAM_B_GAIN_10[10]	GAM_B_GAIN_10[9]	GAM_B_GAIN_10[8]
		GAM_B_GAIN_10[7]	GAM_B_GAIN_10[6]	GAM_B_GAIN_10[5]	GAM_B_GAIN_10[4]	GAM_B_GAIN_10[3]	GAM_B_GAIN_10[2]	GAM_B_GAIN_10[1]	GAM_B_GAIN_10[0]
		-	-	-	-	-	GAM_B_GAIN_11[10]	GAM_B_GAIN_11[9]	GAM_B_GAIN_11[8]
		GAM_B_GAIN_11[7]	GAM_B_GAIN_11[6]	GAM_B_GAIN_11[5]	GAM_B_GAIN_11[4]	GAM_B_GAIN_11[3]	GAM_B_GAIN_11[2]	GAM_B_GAIN_11[1]	GAM_B_GAIN_11[0]
	GAM_B_LUT7	-	-	-	-	-	GAM_B_GAIN_12[10]	GAM_B_GAIN_12[9]	GAM_B_GAIN_12[8]
		GAM_B_GAIN_12[7]	GAM_B_GAIN_12[6]	GAM_B_GAIN_12[5]	GAM_B_GAIN_12[4]	GAM_B_GAIN_12[3]	GAM_B_GAIN_12[2]	GAM_B_GAIN_12[1]	GAM_B_GAIN_12[0]
		-	-	-	-	-	GAM_B_GAIN_13[10]	GAM_B_GAIN_13[9]	GAM_B_GAIN_13[8]
		GAM_B_GAIN_13[7]	GAM_B_GAIN_13[6]	GAM_B_GAIN_13[5]	GAM_B_GAIN_13[4]	GAM_B_GAIN_13[3]	GAM_B_GAIN_13[2]	GAM_B_GAIN_13[1]	GAM_B_GAIN_13[0]
	GAM_B_LUT8	-	-	-	-	-	GAM_B_GAIN_14[10]	GAM_B_GAIN_14[9]	GAM_B_GAIN_14[8]
		GAM_B_GAIN_14[7]	GAM_B_GAIN_14[6]	GAM_B_GAIN_14[5]	GAM_B_GAIN_14[4]	GAM_B_GAIN_14[3]	GAM_B_GAIN_14[2]	GAM_B_GAIN_14[1]	GAM_B_GAIN_14[0]
		-	-	-	-	-	GAM_B_GAIN_15[10]	GAM_B_GAIN_15[9]	GAM_B_GAIN_15[8]
		GAM_B_GAIN_15[7]	GAM_B_GAIN_15[6]	GAM_B_GAIN_15[5]	GAM_B_GAIN_15[4]	GAM_B_GAIN_15[3]	GAM_B_GAIN_15[2]	GAM_B_GAIN_15[1]	GAM_B_GAIN_15[0]
	GAM_B_LUT9	-	-	-	-	-	GAM_B_GAIN_16[10]	GAM_B_GAIN_16[9]	GAM_B_GAIN_16[8]
		GAM_B_GAIN_16[7]	GAM_B_GAIN_16[6]	GAM_B_GAIN_16[5]	GAM_B_GAIN_16[4]	GAM_B_GAIN_16[3]	GAM_B_GAIN_16[2]	GAM_B_GAIN_16[1]	GAM_B_GAIN_16[0]
		-	-	-	-	-	GAM_B_GAIN_17[10]	GAM_B_GAIN_17[9]	GAM_B_GAIN_17[8]
		GAM_B_GAIN_17[7]	GAM_B_GAIN_17[6]	GAM_B_GAIN_17[5]	GAM_B_GAIN_17[4]	GAM_B_GAIN_17[3]	GAM_B_GAIN_17[2]	GAM_B_GAIN_17[1]	GAM_B_GAIN_17[0]
	GAM_B_LUT10	-	-	-	-	-	GAM_B_GAIN_18[10]	GAM_B_GAIN_18[9]	GAM_B_GAIN_18[8]
		GAM_B_GAIN_18[7]	GAM_B_GAIN_18[6]	GAM_B_GAIN_18[5]	GAM_B_GAIN_18[4]	GAM_B_GAIN_18[3]	GAM_B_GAIN_18[2]	GAM_B_GAIN_18[1]	GAM_B_GAIN_18[0]
		-	-	-	-	-	GAM_B_GAIN_19[10]	GAM_B_GAIN_19[9]	GAM_B_GAIN_19[8]
		GAM_B_GAIN_19[7]	GAM_B_GAIN_19[6]	GAM_B_GAIN_19[5]	GAM_B_GAIN_19[4]	GAM_B_GAIN_19[3]	GAM_B_GAIN_19[2]	GAM_B_GAIN_19[1]	GAM_B_GAIN_19[0]
	GAM_B_LUT11	-	-	-	-	-	GAM_B_GAIN_20[10]	GAM_B_GAIN_20[9]	GAM_B_GAIN_20[8]
		GAM_B_GAIN_20[7]	GAM_B_GAIN_20[6]	GAM_B_GAIN_20[5]	GAM_B_GAIN_20[4]	GAM_B_GAIN_20[3]	GAM_B_GAIN_20[2]	GAM_B_GAIN_20[1]	GAM_B_GAIN_20[0]
		-	-	-	-	-	GAM_B_GAIN_21[10]	GAM_B_GAIN_21[9]	GAM_B_GAIN_21[8]
		GAM_B_GAIN_21[7]	GAM_B_GAIN_21[6]	GAM_B_GAIN_21[5]	GAM_B_GAIN_21[4]	GAM_B_GAIN_21[3]	GAM_B_GAIN_21[2]	GAM_B_GAIN_21[1]	GAM_B_GAIN_21[0]
	GAM_B_LUT12	-	-	-	-	-	GAM_B_GAIN_22[10]	GAM_B_GAIN_22[9]	GAM_B_GAIN_22[8]
		GAM_B_GAIN_22[7]	GAM_B_GAIN_22[6]	GAM_B_GAIN_22[5]	GAM_B_GAIN_22[4]	GAM_B_GAIN_22[3]	GAM_B_GAIN_22[2]	GAM_B_GAIN_22[1]	GAM_B_GAIN_22[0]
		-	-	-	-	-	GAM_B_GAIN_23[10]	GAM_B_GAIN_23[9]	GAM_B_GAIN_23[8]
		GAM_B_GAIN_23[7]	GAM_B_GAIN_23[6]	GAM_B_GAIN_23[5]	GAM_B_GAIN_23[4]	GAM_B_GAIN_23[3]	GAM_B_GAIN_23[2]	GAM_B_GAIN_23[1]	GAM_B_GAIN_23[0]
	GAM_B_LUT13	-	-	-	-	-	GAM_B_GAIN_24[10]	GAM_B_GAIN_24[9]	GAM_B_GAIN_24[8]
		GAM_B_GAIN_24[7]	GAM_B_GAIN_24[6]	GAM_B_GAIN_24[5]	GAM_B_GAIN_24[4]	GAM_B_GAIN_24[3]	GAM_B_GAIN_24[2]	GAM_B_GAIN_24[1]	GAM_B_GAIN_24[0]
		-	-	-	-	-	GAM_B_GAIN_25[10]	GAM_B_GAIN_25[9]	GAM_B_GAIN_25[8]
		GAM_B_GAIN_25[7]	GAM_B_GAIN_25[6]	GAM_B_GAIN_25[5]	GAM_B_GAIN_25[4]	GAM_B_GAIN_25[3]	GAM_B_GAIN_25[2]	GAM_B_GAIN_25[1]	GAM_B_GAIN_25[0]
	GAM_B_LUT14	-	-	-	-	-	GAM_B_GAIN_26[10]	GAM_B_GAIN_26[9]	GAM_B_GAIN_26[8]
		GAM_B_GAIN_26[7]	GAM_B_GAIN_26[6]	GAM_B_GAIN_26[5]	GAM_B_GAIN_26[4]	GAM_B_GAIN_26[3]	GAM_B_GAIN_26[2]	GAM_B_GAIN_26[1]	GAM_B_GAIN_26[0]
		-	-	-	-	-	GAM_B_GAIN_27[10]	GAM_B_GAIN_27[9]	GAM_B_GAIN_27[8]
		GAM_B_GAIN_27[7]	GAM_B_GAIN_27[6]	GAM_B_GAIN_27[5]	GAM_B_GAIN_27[4]	GAM_B_GAIN_27[3]	GAM_B_GAIN_27[2]	GAM_B_GAIN_27[1]	GAM_B_GAIN_27[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Video display controller 5 channel 0	GAM_B_LUT15	-	-	-	-	-	GAM_B_GAIN_28[10]	GAM_B_GAIN_28[9]	GAM_B_GAIN_28[8]
		GAM_B_GAIN_28[7]	GAM_B_GAIN_28[6]	GAM_B_GAIN_28[5]	GAM_B_GAIN_28[4]	GAM_B_GAIN_28[3]	GAM_B_GAIN_28[2]	GAM_B_GAIN_28[1]	GAM_B_GAIN_28[0]
		-	-	-	-	-	GAM_B_GAIN_29[10]	GAM_B_GAIN_29[9]	GAM_B_GAIN_29[8]
		GAM_B_GAIN_29[7]	GAM_B_GAIN_29[6]	GAM_B_GAIN_29[5]	GAM_B_GAIN_29[4]	GAM_B_GAIN_29[3]	GAM_B_GAIN_29[2]	GAM_B_GAIN_29[1]	GAM_B_GAIN_29[0]
	GAM_B_LUT16	-	-	-	-	-	GAM_B_GAIN_30[10]	GAM_B_GAIN_30[9]	GAM_B_GAIN_30[8]
		GAM_B_GAIN_30[7]	GAM_B_GAIN_30[6]	GAM_B_GAIN_30[5]	GAM_B_GAIN_30[4]	GAM_B_GAIN_30[3]	GAM_B_GAIN_30[2]	GAM_B_GAIN_30[1]	GAM_B_GAIN_30[0]
		-	-	-	-	-	GAM_B_GAIN_31[10]	GAM_B_GAIN_31[9]	GAM_B_GAIN_31[8]
		GAM_B_GAIN_31[7]	GAM_B_GAIN_31[6]	GAM_B_GAIN_31[5]	GAM_B_GAIN_31[4]	GAM_B_GAIN_31[3]	GAM_B_GAIN_31[2]	GAM_B_GAIN_31[1]	GAM_B_GAIN_31[0]
	GAM_B_AREA1	-	-	-	-	-	-	-	-
		GAM_B_TH_01[7]	GAM_B_TH_01[6]	GAM_B_TH_01[5]	GAM_B_TH_01[4]	GAM_B_TH_01[3]	GAM_B_TH_01[2]	GAM_B_TH_01[1]	GAM_B_TH_01[0]
		GAM_B_TH_02[7]	GAM_B_TH_02[6]	GAM_B_TH_02[5]	GAM_B_TH_02[4]	GAM_B_TH_02[3]	GAM_B_TH_02[2]	GAM_B_TH_02[1]	GAM_B_TH_02[0]
		GAM_B_TH_03[7]	GAM_B_TH_03[6]	GAM_B_TH_03[5]	GAM_B_TH_03[4]	GAM_B_TH_03[3]	GAM_B_TH_03[2]	GAM_B_TH_03[1]	GAM_B_TH_03[0]
	GAM_B_AREA2	GAM_B_TH_04[7]	GAM_B_TH_04[6]	GAM_B_TH_04[5]	GAM_B_TH_04[4]	GAM_B_TH_04[3]	GAM_B_TH_04[2]	GAM_B_TH_04[1]	GAM_B_TH_04[0]
		GAM_B_TH_05[7]	GAM_B_TH_05[6]	GAM_B_TH_05[5]	GAM_B_TH_05[4]	GAM_B_TH_05[3]	GAM_B_TH_05[2]	GAM_B_TH_05[1]	GAM_B_TH_05[0]
		GAM_B_TH_06[7]	GAM_B_TH_06[6]	GAM_B_TH_06[5]	GAM_B_TH_06[4]	GAM_B_TH_06[3]	GAM_B_TH_06[2]	GAM_B_TH_06[1]	GAM_B_TH_06[0]
		GAM_B_TH_07[7]	GAM_B_TH_07[6]	GAM_B_TH_07[5]	GAM_B_TH_07[4]	GAM_B_TH_07[3]	GAM_B_TH_07[2]	GAM_B_TH_07[1]	GAM_B_TH_07[0]
	GAM_B_AREA3	GAM_B_TH_08[7]	GAM_B_TH_08[6]	GAM_B_TH_08[5]	GAM_B_TH_08[4]	GAM_B_TH_08[3]	GAM_B_TH_08[2]	GAM_B_TH_08[1]	GAM_B_TH_08[0]
		GAM_B_TH_09[7]	GAM_B_TH_09[6]	GAM_B_TH_09[5]	GAM_B_TH_09[4]	GAM_B_TH_09[3]	GAM_B_TH_09[2]	GAM_B_TH_09[1]	GAM_B_TH_09[0]
		GAM_B_TH_10[7]	GAM_B_TH_10[6]	GAM_B_TH_10[5]	GAM_B_TH_10[4]	GAM_B_TH_10[3]	GAM_B_TH_10[2]	GAM_B_TH_10[1]	GAM_B_TH_10[0]
		GAM_B_TH_11[7]	GAM_B_TH_11[6]	GAM_B_TH_11[5]	GAM_B_TH_11[4]	GAM_B_TH_11[3]	GAM_B_TH_11[2]	GAM_B_TH_11[1]	GAM_B_TH_11[0]
	GAM_B_AREA4	GAM_B_TH_12[7]	GAM_B_TH_12[6]	GAM_B_TH_12[5]	GAM_B_TH_12[4]	GAM_B_TH_12[3]	GAM_B_TH_12[2]	GAM_B_TH_12[1]	GAM_B_TH_12[0]
		GAM_B_TH_13[7]	GAM_B_TH_13[6]	GAM_B_TH_13[5]	GAM_B_TH_13[4]	GAM_B_TH_13[3]	GAM_B_TH_13[2]	GAM_B_TH_13[1]	GAM_B_TH_13[0]
		GAM_B_TH_14[7]	GAM_B_TH_14[6]	GAM_B_TH_14[5]	GAM_B_TH_14[4]	GAM_B_TH_14[3]	GAM_B_TH_14[2]	GAM_B_TH_14[1]	GAM_B_TH_14[0]
		GAM_B_TH_15[7]	GAM_B_TH_15[6]	GAM_B_TH_15[5]	GAM_B_TH_15[4]	GAM_B_TH_15[3]	GAM_B_TH_15[2]	GAM_B_TH_15[1]	GAM_B_TH_15[0]
	GAM_B_AREA5	GAM_B_TH_16[7]	GAM_B_TH_16[6]	GAM_B_TH_16[5]	GAM_B_TH_16[4]	GAM_B_TH_16[3]	GAM_B_TH_16[2]	GAM_B_TH_16[1]	GAM_B_TH_16[0]
		GAM_B_TH_17[7]	GAM_B_TH_17[6]	GAM_B_TH_17[5]	GAM_B_TH_17[4]	GAM_B_TH_17[3]	GAM_B_TH_17[2]	GAM_B_TH_17[1]	GAM_B_TH_17[0]
		GAM_B_TH_18[7]	GAM_B_TH_18[6]	GAM_B_TH_18[5]	GAM_B_TH_18[4]	GAM_B_TH_18[3]	GAM_B_TH_18[2]	GAM_B_TH_18[1]	GAM_B_TH_18[0]
		GAM_B_TH_19[7]	GAM_B_TH_19[6]	GAM_B_TH_19[5]	GAM_B_TH_19[4]	GAM_B_TH_19[3]	GAM_B_TH_19[2]	GAM_B_TH_19[1]	GAM_B_TH_19[0]
	GAM_B_AREA6	GAM_B_TH_20[7]	GAM_B_TH_20[6]	GAM_B_TH_20[5]	GAM_B_TH_20[4]	GAM_B_TH_20[3]	GAM_B_TH_20[2]	GAM_B_TH_20[1]	GAM_B_TH_20[0]
		GAM_B_TH_21[7]	GAM_B_TH_21[6]	GAM_B_TH_21[5]	GAM_B_TH_21[4]	GAM_B_TH_21[3]	GAM_B_TH_21[2]	GAM_B_TH_21[1]	GAM_B_TH_21[0]
		GAM_B_TH_22[7]	GAM_B_TH_22[6]	GAM_B_TH_22[5]	GAM_B_TH_22[4]	GAM_B_TH_22[3]	GAM_B_TH_22[2]	GAM_B_TH_22[1]	GAM_B_TH_22[0]
		GAM_B_TH_23[7]	GAM_B_TH_23[6]	GAM_B_TH_23[5]	GAM_B_TH_23[4]	GAM_B_TH_23[3]	GAM_B_TH_23[2]	GAM_B_TH_23[1]	GAM_B_TH_23[0]
	GAM_B_AREA7	GAM_B_TH_24[7]	GAM_B_TH_24[6]	GAM_B_TH_24[5]	GAM_B_TH_24[4]	GAM_B_TH_24[3]	GAM_B_TH_24[2]	GAM_B_TH_24[1]	GAM_B_TH_24[0]
		GAM_B_TH_25[7]	GAM_B_TH_25[6]	GAM_B_TH_25[5]	GAM_B_TH_25[4]	GAM_B_TH_25[3]	GAM_B_TH_25[2]	GAM_B_TH_25[1]	GAM_B_TH_25[0]
		GAM_B_TH_26[7]	GAM_B_TH_26[6]	GAM_B_TH_26[5]	GAM_B_TH_26[4]	GAM_B_TH_26[3]	GAM_B_TH_26[2]	GAM_B_TH_26[1]	GAM_B_TH_26[0]
		GAM_B_TH_27[7]	GAM_B_TH_27[6]	GAM_B_TH_27[5]	GAM_B_TH_27[4]	GAM_B_TH_27[3]	GAM_B_TH_27[2]	GAM_B_TH_27[1]	GAM_B_TH_27[0]
	GAM_B_AREA8	GAM_B_TH_28[7]	GAM_B_TH_28[6]	GAM_B_TH_28[5]	GAM_B_TH_28[4]	GAM_B_TH_28[3]	GAM_B_TH_28[2]	GAM_B_TH_28[1]	GAM_B_TH_28[0]
		GAM_B_TH_29[7]	GAM_B_TH_29[6]	GAM_B_TH_29[5]	GAM_B_TH_29[4]	GAM_B_TH_29[3]	GAM_B_TH_29[2]	GAM_B_TH_29[1]	GAM_B_TH_29[0]
		GAM_B_TH_30[7]	GAM_B_TH_30[6]	GAM_B_TH_30[5]	GAM_B_TH_30[4]	GAM_B_TH_30[3]	GAM_B_TH_30[2]	GAM_B_TH_30[1]	GAM_B_TH_30[0]
		GAM_B_TH_31[7]	GAM_B_TH_31[6]	GAM_B_TH_31[5]	GAM_B_TH_31[4]	GAM_B_TH_31[3]	GAM_B_TH_31[2]	GAM_B_TH_31[1]	GAM_B_TH_31[0]
	GAM_R_UPDATE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GAM_R_VEN
	GAM_R_LUT1	-	-	-	-	-	GAM_R_GAIN_00[10]	GAM_R_GAIN_00[9]	GAM_R_GAIN_00[8]
		GAM_R_GAIN_00[7]	GAM_R_GAIN_00[6]	GAM_R_GAIN_00[5]	GAM_R_GAIN_00[4]	GAM_R_GAIN_00[3]	GAM_R_GAIN_00[2]	GAM_R_GAIN_00[1]	GAM_R_GAIN_00[0]
		-	-	-	-	-	GAM_R_GAIN_01[10]	GAM_R_GAIN_01[9]	GAM_R_GAIN_01[8]
		GAM_R_GAIN_01[7]	GAM_R_GAIN_01[6]	GAM_R_GAIN_01[5]	GAM_R_GAIN_01[4]	GAM_R_GAIN_01[3]	GAM_R_GAIN_01[2]	GAM_R_GAIN_01[1]	GAM_R_GAIN_01[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Video display controller 5 channel 0	GAM_R_LUT2	-	-	-	-	-	GAM_R_GAIN_02[10]	GAM_R_GAIN_02[9]	GAM_R_GAIN_02[8]
		GAM_R_GAIN_02[7]	GAM_R_GAIN_02[6]	GAM_R_GAIN_02[5]	GAM_R_GAIN_02[4]	GAM_R_GAIN_02[3]	GAM_R_GAIN_02[2]	GAM_R_GAIN_02[1]	GAM_R_GAIN_02[0]
		-	-	-	-	-	GAM_R_GAIN_03[10]	GAM_R_GAIN_03[9]	GAM_R_GAIN_03[8]
		GAM_R_GAIN_03[7]	GAM_R_GAIN_03[6]	GAM_R_GAIN_03[5]	GAM_R_GAIN_03[4]	GAM_R_GAIN_03[3]	GAM_R_GAIN_03[2]	GAM_R_GAIN_03[1]	GAM_R_GAIN_03[0]
	GAM_R_LUT3	-	-	-	-	-	GAM_R_GAIN_04[10]	GAM_R_GAIN_04[9]	GAM_R_GAIN_04[8]
		GAM_R_GAIN_04[7]	GAM_R_GAIN_04[6]	GAM_R_GAIN_04[5]	GAM_R_GAIN_04[4]	GAM_R_GAIN_04[3]	GAM_R_GAIN_04[2]	GAM_R_GAIN_04[1]	GAM_R_GAIN_04[0]
		-	-	-	-	-	GAM_R_GAIN_05[10]	GAM_R_GAIN_05[9]	GAM_R_GAIN_05[8]
		GAM_R_GAIN_05[7]	GAM_R_GAIN_05[6]	GAM_R_GAIN_05[5]	GAM_R_GAIN_05[4]	GAM_R_GAIN_05[3]	GAM_R_GAIN_05[2]	GAM_R_GAIN_05[1]	GAM_R_GAIN_05[0]
	GAM_R_LUT4	-	-	-	-	-	GAM_R_GAIN_06[10]	GAM_R_GAIN_06[9]	GAM_R_GAIN_06[8]
		GAM_R_GAIN_06[7]	GAM_R_GAIN_06[6]	GAM_R_GAIN_06[5]	GAM_R_GAIN_06[4]	GAM_R_GAIN_06[3]	GAM_R_GAIN_06[2]	GAM_R_GAIN_06[1]	GAM_R_GAIN_06[0]
		-	-	-	-	-	GAM_R_GAIN_07[10]	GAM_R_GAIN_07[9]	GAM_R_GAIN_07[8]
		GAM_R_GAIN_07[7]	GAM_R_GAIN_07[6]	GAM_R_GAIN_07[5]	GAM_R_GAIN_07[4]	GAM_R_GAIN_07[3]	GAM_R_GAIN_07[2]	GAM_R_GAIN_07[1]	GAM_R_GAIN_07[0]
	GAM_R_LUT5	-	-	-	-	-	GAM_R_GAIN_08[10]	GAM_R_GAIN_08[9]	GAM_R_GAIN_08[8]
		GAM_R_GAIN_08[7]	GAM_R_GAIN_08[6]	GAM_R_GAIN_08[5]	GAM_R_GAIN_08[4]	GAM_R_GAIN_08[3]	GAM_R_GAIN_08[2]	GAM_R_GAIN_08[1]	GAM_R_GAIN_08[0]
		-	-	-	-	-	GAM_R_GAIN_09[10]	GAM_R_GAIN_09[9]	GAM_R_GAIN_09[8]
		GAM_R_GAIN_09[7]	GAM_R_GAIN_09[6]	GAM_R_GAIN_09[5]	GAM_R_GAIN_09[4]	GAM_R_GAIN_09[3]	GAM_R_GAIN_09[2]	GAM_R_GAIN_09[1]	GAM_R_GAIN_09[0]
	GAM_R_LUT6	-	-	-	-	-	GAM_R_GAIN_10[10]	GAM_R_GAIN_10[9]	GAM_R_GAIN_10[8]
		GAM_R_GAIN_10[7]	GAM_R_GAIN_10[6]	GAM_R_GAIN_10[5]	GAM_R_GAIN_10[4]	GAM_R_GAIN_10[3]	GAM_R_GAIN_10[2]	GAM_R_GAIN_10[1]	GAM_R_GAIN_10[0]
		-	-	-	-	-	GAM_R_GAIN_11[10]	GAM_R_GAIN_11[9]	GAM_R_GAIN_11[8]
		GAM_R_GAIN_11[7]	GAM_R_GAIN_11[6]	GAM_R_GAIN_11[5]	GAM_R_GAIN_11[4]	GAM_R_GAIN_11[3]	GAM_R_GAIN_11[2]	GAM_R_GAIN_11[1]	GAM_R_GAIN_11[0]
	GAM_R_LUT7	-	-	-	-	-	GAM_R_GAIN_12[10]	GAM_R_GAIN_12[9]	GAM_R_GAIN_12[8]
		GAM_R_GAIN_12[7]	GAM_R_GAIN_12[6]	GAM_R_GAIN_12[5]	GAM_R_GAIN_12[4]	GAM_R_GAIN_12[3]	GAM_R_GAIN_12[2]	GAM_R_GAIN_12[1]	GAM_R_GAIN_12[0]
		-	-	-	-	-	GAM_R_GAIN_13[10]	GAM_R_GAIN_13[9]	GAM_R_GAIN_13[8]
		GAM_R_GAIN_13[7]	GAM_R_GAIN_13[6]	GAM_R_GAIN_13[5]	GAM_R_GAIN_13[4]	GAM_R_GAIN_13[3]	GAM_R_GAIN_13[2]	GAM_R_GAIN_13[1]	GAM_R_GAIN_13[0]
	GAM_R_LUT8	-	-	-	-	-	GAM_R_GAIN_14[10]	GAM_R_GAIN_14[9]	GAM_R_GAIN_14[8]
		GAM_R_GAIN_14[7]	GAM_R_GAIN_14[6]	GAM_R_GAIN_14[5]	GAM_R_GAIN_14[4]	GAM_R_GAIN_14[3]	GAM_R_GAIN_14[2]	GAM_R_GAIN_14[1]	GAM_R_GAIN_14[0]
		-	-	-	-	-	GAM_R_GAIN_15[10]	GAM_R_GAIN_15[9]	GAM_R_GAIN_15[8]
		GAM_R_GAIN_15[7]	GAM_R_GAIN_15[6]	GAM_R_GAIN_15[5]	GAM_R_GAIN_15[4]	GAM_R_GAIN_15[3]	GAM_R_GAIN_15[2]	GAM_R_GAIN_15[1]	GAM_R_GAIN_15[0]
	GAM_R_LUT9	-	-	-	-	-	GAM_R_GAIN_16[10]	GAM_R_GAIN_16[9]	GAM_R_GAIN_16[8]
		GAM_R_GAIN_16[7]	GAM_R_GAIN_16[6]	GAM_R_GAIN_16[5]	GAM_R_GAIN_16[4]	GAM_R_GAIN_16[3]	GAM_R_GAIN_16[2]	GAM_R_GAIN_16[1]	GAM_R_GAIN_16[0]
		-	-	-	-	-	GAM_R_GAIN_17[10]	GAM_R_GAIN_17[9]	GAM_R_GAIN_17[8]
		GAM_R_GAIN_17[7]	GAM_R_GAIN_17[6]	GAM_R_GAIN_17[5]	GAM_R_GAIN_17[4]	GAM_R_GAIN_17[3]	GAM_R_GAIN_17[2]	GAM_R_GAIN_17[1]	GAM_R_GAIN_17[0]
	GAM_R_LUT10	-	-	-	-	-	GAM_R_GAIN_18[10]	GAM_R_GAIN_18[9]	GAM_R_GAIN_18[8]
		GAM_R_GAIN_18[7]	GAM_R_GAIN_18[6]	GAM_R_GAIN_18[5]	GAM_R_GAIN_18[4]	GAM_R_GAIN_18[3]	GAM_R_GAIN_18[2]	GAM_R_GAIN_18[1]	GAM_R_GAIN_18[0]
		-	-	-	-	-	GAM_R_GAIN_19[10]	GAM_R_GAIN_19[9]	GAM_R_GAIN_19[8]
		GAM_R_GAIN_19[7]	GAM_R_GAIN_19[6]	GAM_R_GAIN_19[5]	GAM_R_GAIN_19[4]	GAM_R_GAIN_19[3]	GAM_R_GAIN_19[2]	GAM_R_GAIN_19[1]	GAM_R_GAIN_19[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Video display controller 5 channel 0	GAM_R_LUT11	-	-	-	-	-	GAM_R_GAIN_20[10]	GAM_R_GAIN_20[9]	GAM_R_GAIN_20[8]
		GAM_R_GAIN_20[7]	GAM_R_GAIN_20[6]	GAM_R_GAIN_20[5]	GAM_R_GAIN_20[4]	GAM_R_GAIN_20[3]	GAM_R_GAIN_20[2]	GAM_R_GAIN_20[1]	GAM_R_GAIN_20[0]
		-	-	-	-	-	GAM_R_GAIN_21[10]	GAM_R_GAIN_21[9]	GAM_R_GAIN_21[8]
		GAM_R_GAIN_21[7]	GAM_R_GAIN_21[6]	GAM_R_GAIN_21[5]	GAM_R_GAIN_21[4]	GAM_R_GAIN_21[3]	GAM_R_GAIN_21[2]	GAM_R_GAIN_21[1]	GAM_R_GAIN_21[0]
	GAM_R_LUT12	-	-	-	-	-	GAM_R_GAIN_22[10]	GAM_R_GAIN_22[9]	GAM_R_GAIN_22[8]
		GAM_R_GAIN_22[7]	GAM_R_GAIN_22[6]	GAM_R_GAIN_22[5]	GAM_R_GAIN_22[4]	GAM_R_GAIN_22[3]	GAM_R_GAIN_22[2]	GAM_R_GAIN_22[1]	GAM_R_GAIN_22[0]
		-	-	-	-	-	GAM_R_GAIN_23[10]	GAM_R_GAIN_23[9]	GAM_R_GAIN_23[8]
		GAM_R_GAIN_23[7]	GAM_R_GAIN_23[6]	GAM_R_GAIN_23[5]	GAM_R_GAIN_23[4]	GAM_R_GAIN_23[3]	GAM_R_GAIN_23[2]	GAM_R_GAIN_23[1]	GAM_R_GAIN_23[0]
	GAM_R_LUT13	-	-	-	-	-	GAM_R_GAIN_24[10]	GAM_R_GAIN_24[9]	GAM_R_GAIN_24[8]
		GAM_R_GAIN_24[7]	GAM_R_GAIN_24[6]	GAM_R_GAIN_24[5]	GAM_R_GAIN_24[4]	GAM_R_GAIN_24[3]	GAM_R_GAIN_24[2]	GAM_R_GAIN_24[1]	GAM_R_GAIN_24[0]
		-	-	-	-	-	GAM_R_GAIN_25[10]	GAM_R_GAIN_25[9]	GAM_R_GAIN_25[8]
		GAM_R_GAIN_25[7]	GAM_R_GAIN_25[6]	GAM_R_GAIN_25[5]	GAM_R_GAIN_25[4]	GAM_R_GAIN_25[3]	GAM_R_GAIN_25[2]	GAM_R_GAIN_25[1]	GAM_R_GAIN_25[0]
	GAM_R_LUT14	-	-	-	-	-	GAM_R_GAIN_26[10]	GAM_R_GAIN_26[9]	GAM_R_GAIN_26[8]
		GAM_R_GAIN_26[7]	GAM_R_GAIN_26[6]	GAM_R_GAIN_26[5]	GAM_R_GAIN_26[4]	GAM_R_GAIN_26[3]	GAM_R_GAIN_26[2]	GAM_R_GAIN_26[1]	GAM_R_GAIN_26[0]
		-	-	-	-	-	GAM_R_GAIN_27[10]	GAM_R_GAIN_27[9]	GAM_R_GAIN_27[8]
		GAM_R_GAIN_27[7]	GAM_R_GAIN_27[6]	GAM_R_GAIN_27[5]	GAM_R_GAIN_27[4]	GAM_R_GAIN_27[3]	GAM_R_GAIN_27[2]	GAM_R_GAIN_27[1]	GAM_R_GAIN_27[0]
	GAM_R_LUT15	-	-	-	-	-	GAM_R_GAIN_28[10]	GAM_R_GAIN_28[9]	GAM_R_GAIN_28[8]
		GAM_R_GAIN_28[7]	GAM_R_GAIN_28[6]	GAM_R_GAIN_28[5]	GAM_R_GAIN_28[4]	GAM_R_GAIN_28[3]	GAM_R_GAIN_28[2]	GAM_R_GAIN_28[1]	GAM_R_GAIN_28[0]
		-	-	-	-	-	GAM_R_GAIN_29[10]	GAM_R_GAIN_29[9]	GAM_R_GAIN_29[8]
GAM_R_GAIN_29[7]		GAM_R_GAIN_29[6]	GAM_R_GAIN_29[5]	GAM_R_GAIN_29[4]	GAM_R_GAIN_29[3]	GAM_R_GAIN_29[2]	GAM_R_GAIN_29[1]	GAM_R_GAIN_29[0]	
GAM_R_LUT16	-	-	-	-	-	GAM_R_GAIN_30[10]	GAM_R_GAIN_30[9]	GAM_R_GAIN_30[8]	
	GAM_R_GAIN_30[7]	GAM_R_GAIN_30[6]	GAM_R_GAIN_30[5]	GAM_R_GAIN_30[4]	GAM_R_GAIN_30[3]	GAM_R_GAIN_30[2]	GAM_R_GAIN_30[1]	GAM_R_GAIN_30[0]	
	-	-	-	-	-	GAM_R_GAIN_31[10]	GAM_R_GAIN_31[9]	GAM_R_GAIN_31[8]	
	GAM_R_GAIN_31[7]	GAM_R_GAIN_31[6]	GAM_R_GAIN_31[5]	GAM_R_GAIN_31[4]	GAM_R_GAIN_31[3]	GAM_R_GAIN_31[2]	GAM_R_GAIN_31[1]	GAM_R_GAIN_31[0]	
GAM_R_AREA1	-	-	-	-	-	-	-	-	
	GAM_R_TH_01[7]	GAM_R_TH_01[6]	GAM_R_TH_01[5]	GAM_R_TH_01[4]	GAM_R_TH_01[3]	GAM_R_TH_01[2]	GAM_R_TH_01[1]	GAM_R_TH_01[0]	
	GAM_R_TH_02[7]	GAM_R_TH_02[6]	GAM_R_TH_02[5]	GAM_R_TH_02[4]	GAM_R_TH_02[3]	GAM_R_TH_02[2]	GAM_R_TH_02[1]	GAM_R_TH_02[0]	
	GAM_R_TH_03[7]	GAM_R_TH_03[6]	GAM_R_TH_03[5]	GAM_R_TH_03[4]	GAM_R_TH_03[3]	GAM_R_TH_03[2]	GAM_R_TH_03[1]	GAM_R_TH_03[0]	
GAM_R_AREA2	GAM_R_TH_04[7]	GAM_R_TH_04[6]	GAM_R_TH_04[5]	GAM_R_TH_04[4]	GAM_R_TH_04[3]	GAM_R_TH_04[2]	GAM_R_TH_04[1]	GAM_R_TH_04[0]	
	GAM_R_TH_05[7]	GAM_R_TH_05[6]	GAM_R_TH_05[5]	GAM_R_TH_05[4]	GAM_R_TH_05[3]	GAM_R_TH_05[2]	GAM_R_TH_05[1]	GAM_R_TH_05[0]	
	GAM_R_TH_06[7]	GAM_R_TH_06[6]	GAM_R_TH_06[5]	GAM_R_TH_06[4]	GAM_R_TH_06[3]	GAM_R_TH_06[2]	GAM_R_TH_06[1]	GAM_R_TH_06[0]	
	GAM_R_TH_07[7]	GAM_R_TH_07[6]	GAM_R_TH_07[5]	GAM_R_TH_07[4]	GAM_R_TH_07[3]	GAM_R_TH_07[2]	GAM_R_TH_07[1]	GAM_R_TH_07[0]	
GAM_R_AREA3	GAM_R_TH_08[7]	GAM_R_TH_08[6]	GAM_R_TH_08[5]	GAM_R_TH_08[4]	GAM_R_TH_08[3]	GAM_R_TH_08[2]	GAM_R_TH_08[1]	GAM_R_TH_08[0]	
	GAM_R_TH_09[7]	GAM_R_TH_09[6]	GAM_R_TH_09[5]	GAM_R_TH_09[4]	GAM_R_TH_09[3]	GAM_R_TH_09[2]	GAM_R_TH_09[1]	GAM_R_TH_09[0]	
	GAM_R_TH_10[7]	GAM_R_TH_10[6]	GAM_R_TH_10[5]	GAM_R_TH_10[4]	GAM_R_TH_10[3]	GAM_R_TH_10[2]	GAM_R_TH_10[1]	GAM_R_TH_10[0]	
	GAM_R_TH_11[7]	GAM_R_TH_11[6]	GAM_R_TH_11[5]	GAM_R_TH_11[4]	GAM_R_TH_11[3]	GAM_R_TH_11[2]	GAM_R_TH_11[1]	GAM_R_TH_11[0]	
GAM_R_AREA4	GAM_R_TH_12[7]	GAM_R_TH_12[6]	GAM_R_TH_12[5]	GAM_R_TH_12[4]	GAM_R_TH_12[3]	GAM_R_TH_12[2]	GAM_R_TH_12[1]	GAM_R_TH_12[0]	
	GAM_R_TH_13[7]	GAM_R_TH_13[6]	GAM_R_TH_13[5]	GAM_R_TH_13[4]	GAM_R_TH_13[3]	GAM_R_TH_13[2]	GAM_R_TH_13[1]	GAM_R_TH_13[0]	
	GAM_R_TH_14[7]	GAM_R_TH_14[6]	GAM_R_TH_14[5]	GAM_R_TH_14[4]	GAM_R_TH_14[3]	GAM_R_TH_14[2]	GAM_R_TH_14[1]	GAM_R_TH_14[0]	
	GAM_R_TH_15[7]	GAM_R_TH_15[6]	GAM_R_TH_15[5]	GAM_R_TH_15[4]	GAM_R_TH_15[3]	GAM_R_TH_15[2]	GAM_R_TH_15[1]	GAM_R_TH_15[0]	
GAM_R_AREA5	GAM_R_TH_16[7]	GAM_R_TH_16[6]	GAM_R_TH_16[5]	GAM_R_TH_16[4]	GAM_R_TH_16[3]	GAM_R_TH_16[2]	GAM_R_TH_16[1]	GAM_R_TH_16[0]	
	GAM_R_TH_17[7]	GAM_R_TH_17[6]	GAM_R_TH_17[5]	GAM_R_TH_17[4]	GAM_R_TH_17[3]	GAM_R_TH_17[2]	GAM_R_TH_17[1]	GAM_R_TH_17[0]	
	GAM_R_TH_18[7]	GAM_R_TH_18[6]	GAM_R_TH_18[5]	GAM_R_TH_18[4]	GAM_R_TH_18[3]	GAM_R_TH_18[2]	GAM_R_TH_18[1]	GAM_R_TH_18[0]	
	GAM_R_TH_19[7]	GAM_R_TH_19[6]	GAM_R_TH_19[5]	GAM_R_TH_19[4]	GAM_R_TH_19[3]	GAM_R_TH_19[2]	GAM_R_TH_19[1]	GAM_R_TH_19[0]	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Video display controller 5 channel 0	GAM_R_AREA6	GAM_R_TH_20[7]	GAM_R_TH_20[6]	GAM_R_TH_20[5]	GAM_R_TH_20[4]	GAM_R_TH_20[3]	GAM_R_TH_20[2]	GAM_R_TH_20[1]	GAM_R_TH_20[0]
		GAM_R_TH_21[7]	GAM_R_TH_21[6]	GAM_R_TH_21[5]	GAM_R_TH_21[4]	GAM_R_TH_21[3]	GAM_R_TH_21[2]	GAM_R_TH_21[1]	GAM_R_TH_21[0]
		GAM_R_TH_22[7]	GAM_R_TH_22[6]	GAM_R_TH_22[5]	GAM_R_TH_22[4]	GAM_R_TH_22[3]	GAM_R_TH_22[2]	GAM_R_TH_22[1]	GAM_R_TH_22[0]
		GAM_R_TH_23[7]	GAM_R_TH_23[6]	GAM_R_TH_23[5]	GAM_R_TH_23[4]	GAM_R_TH_23[3]	GAM_R_TH_23[2]	GAM_R_TH_23[1]	GAM_R_TH_23[0]
	GAM_R_AREA7	GAM_R_TH_24[7]	GAM_R_TH_24[6]	GAM_R_TH_24[5]	GAM_R_TH_24[4]	GAM_R_TH_24[3]	GAM_R_TH_24[2]	GAM_R_TH_24[1]	GAM_R_TH_24[0]
		GAM_R_TH_25[7]	GAM_R_TH_25[6]	GAM_R_TH_25[5]	GAM_R_TH_25[4]	GAM_R_TH_25[3]	GAM_R_TH_25[2]	GAM_R_TH_25[1]	GAM_R_TH_25[0]
		GAM_R_TH_26[7]	GAM_R_TH_26[6]	GAM_R_TH_26[5]	GAM_R_TH_26[4]	GAM_R_TH_26[3]	GAM_R_TH_26[2]	GAM_R_TH_26[1]	GAM_R_TH_26[0]
		GAM_R_TH_27[7]	GAM_R_TH_27[6]	GAM_R_TH_27[5]	GAM_R_TH_27[4]	GAM_R_TH_27[3]	GAM_R_TH_27[2]	GAM_R_TH_27[1]	GAM_R_TH_27[0]
	GAM_R_AREA8	GAM_R_TH_28[7]	GAM_R_TH_28[6]	GAM_R_TH_28[5]	GAM_R_TH_28[4]	GAM_R_TH_28[3]	GAM_R_TH_28[2]	GAM_R_TH_28[1]	GAM_R_TH_28[0]
		GAM_R_TH_29[7]	GAM_R_TH_29[6]	GAM_R_TH_29[5]	GAM_R_TH_29[4]	GAM_R_TH_29[3]	GAM_R_TH_29[2]	GAM_R_TH_29[1]	GAM_R_TH_29[0]
		GAM_R_TH_30[7]	GAM_R_TH_30[6]	GAM_R_TH_30[5]	GAM_R_TH_30[4]	GAM_R_TH_30[3]	GAM_R_TH_30[2]	GAM_R_TH_30[1]	GAM_R_TH_30[0]
		GAM_R_TH_31[7]	GAM_R_TH_31[6]	GAM_R_TH_31[5]	GAM_R_TH_31[4]	GAM_R_TH_31[3]	GAM_R_TH_31[2]	GAM_R_TH_31[1]	GAM_R_TH_31[0]
	TCON_UPDATE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	TCON_VEN
	TCON_TIM	-	-	-	-	-	TCON_HALF[10]	TCON_HALF[9]	TCON_HALF[8]
		TCON_HALF[7]	TCON_HALF[6]	TCON_HALF[5]	TCON_HALF[4]	TCON_HALF[3]	TCON_HALF[2]	TCON_HALF[1]	TCON_HALF[0]
		-	-	-	-	-	TCON_OFFSET[10]	TCON_OFFSET[9]	TCON_OFFSET[8]
		TCON_OFFSET[7]	TCON_OFFSET[6]	TCON_OFFSET[5]	TCON_OFFSET[4]	TCON_OFFSET[3]	TCON_OFFSET[2]	TCON_OFFSET[1]	TCON_OFFSET[0]
	TCON_TIM_STVA1	-	-	-	-	-	TCON_STVA_VS[10]	TCON_STVA_VS[9]	TCON_STVA_VS[8]
		TCON_STVA_VS[7]	TCON_STVA_VS[6]	TCON_STVA_VS[5]	TCON_STVA_VS[4]	TCON_STVA_VS[3]	TCON_STVA_VS[2]	TCON_STVA_VS[1]	TCON_STVA_VS[0]
		-	-	-	-	-	TCON_STVA_VW[10]	TCON_STVA_VW[9]	TCON_STVA_VW[8]
		TCON_STVA_VW[7]	TCON_STVA_VW[6]	TCON_STVA_VW[5]	TCON_STVA_VW[4]	TCON_STVA_VW[3]	TCON_STVA_VW[2]	TCON_STVA_VW[1]	TCON_STVA_VW[0]
	TCON_TIM_STVA2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	TCON_STVA_INV	-	TCON_STVA_SEL[2]	TCON_STVA_SEL[1]	TCON_STVA_SEL[0]
	TCON_TIM_STVB1	-	-	-	-	-	TCON_STVB_VS[10]	TCON_STVB_VS[9]	TCON_STVB_VS[8]
		TCON_STVB_VS[7]	TCON_STVB_VS[6]	TCON_STVB_VS[5]	TCON_STVB_VS[4]	TCON_STVB_VS[3]	TCON_STVB_VS[2]	TCON_STVB_VS[1]	TCON_STVB_VS[0]
		-	-	-	-	-	TCON_STVB_VW[10]	TCON_STVB_VW[9]	TCON_STVB_VW[8]
		TCON_STVB_VW[7]	TCON_STVB_VW[6]	TCON_STVB_VW[5]	TCON_STVB_VW[4]	TCON_STVB_VW[3]	TCON_STVB_VW[2]	TCON_STVB_VW[1]	TCON_STVB_VW[0]
	TCON_TIM_STVB2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	TCON_STVB_INV	-	TCON_STVB_SEL[2]	TCON_STVB_SEL[1]	TCON_STVB_SEL[0]
	TCON_TIM_STH1	-	-	-	-	-	TCON_STH_HS[10]	TCON_STH_HS[9]	TCON_STH_HS[8]
		TCON_STH_HS[7]	TCON_STH_HS[6]	TCON_STH_HS[5]	TCON_STH_HS[4]	TCON_STH_HS[3]	TCON_STH_HS[2]	TCON_STH_HS[1]	TCON_STH_HS[0]
		-	-	-	-	-	TCON_STH_HW[10]	TCON_STH_HW[9]	TCON_STH_HW[8]
		TCON_STH_HW[7]	TCON_STH_HW[6]	TCON_STH_HW[5]	TCON_STH_HW[4]	TCON_STH_HW[3]	TCON_STH_HW[2]	TCON_STH_HW[1]	TCON_STH_HW[0]
	TCON_TIM_STH2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	TCON_STH_HS_SEL
		-	-	-	TCON_STH_INV	-	TCON_STH_SEL[2]	TCON_STH_SEL[1]	TCON_STH_SEL[0]
	TCON_TIM_STB1	-	-	-	-	-	TCON_STB_HS[10]	TCON_STB_HS[9]	TCON_STB_HS[8]
		TCON_STB_HS[7]	TCON_STB_HS[6]	TCON_STB_HS[5]	TCON_STB_HS[4]	TCON_STB_HS[3]	TCON_STB_HS[2]	TCON_STB_HS[1]	TCON_STB_HS[0]
		-	-	-	-	-	TCON_STB_HW[10]	TCON_STB_HW[9]	TCON_STB_HW[8]
		TCON_STB_HW[7]	TCON_STB_HW[6]	TCON_STB_HW[5]	TCON_STB_HW[4]	TCON_STB_HW[3]	TCON_STB_HW[2]	TCON_STB_HW[1]	TCON_STB_HW[0]



Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 0	TCON_TIM_STB2	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	TCON_STB_HS_SEL	
		-	-	-	TCON_STB_INV	-	TCON_STB_SEL[2]	TCON_STB_SEL[1]	TCON_STB_SEL[0]	
	TCON_TIM_CPV1	-	-	-	-	-	-	TCON_CPV_HS[10]	TCON_CPV_HS[9]	TCON_CPV_HS[8]
		TCON_CPV_HS[7]	TCON_CPV_HS[6]	TCON_CPV_HS[5]	TCON_CPV_HS[4]	TCON_CPV_HS[3]	TCON_CPV_HS[2]	TCON_CPV_HS[1]	TCON_CPV_HS[0]	
		-	-	-	-	-	TCON_CPV_HW[10]	TCON_CPV_HW[9]	TCON_CPV_HW[8]	
		TCON_CPV_HW[7]	TCON_CPV_HW[6]	TCON_CPV_HW[5]	TCON_CPV_HW[4]	TCON_CPV_HW[3]	TCON_CPV_HW[2]	TCON_CPV_HW[1]	TCON_CPV_HW[0]	
	TCON_TIM_CPV2	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	TCON_CPV_HS_SEL	
		-	-	-	TCON_CPV_INV	-	TCON_CPV_SEL[2]	TCON_CPV_SEL[1]	TCON_CPV_SEL[0]	
	TCON_TIM_POLA1	-	-	-	-	-	-	TCON_POLA_HS[10]	TCON_POLA_HS[9]	TCON_POLA_HS[8]
		TCON_POLA_HS[7]	TCON_POLA_HS[6]	TCON_POLA_HS[5]	TCON_POLA_HS[4]	TCON_POLA_HS[3]	TCON_POLA_HS[2]	TCON_POLA_HS[1]	TCON_POLA_HS[0]	
		-	-	-	-	-	TCON_POLA_HW[10]	TCON_POLA_HW[9]	TCON_POLA_HW[8]	
		TCON_POLA_HW[7]	TCON_POLA_HW[6]	TCON_POLA_HW[5]	TCON_POLA_HW[4]	TCON_POLA_HW[3]	TCON_POLA_HW[2]	TCON_POLA_HW[1]	TCON_POLA_HW[0]	
	TCON_TIM_POLA2	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	TCON_POLA_MD[1]	TCON_POLA_MD[0]	-	-	-	TCON_POLA_HS_SEL	
		-	-	-	TCON_POLA_INV	-	TCON_POLA_SEL[2]	TCON_POLA_SEL[1]	TCON_POLA_SEL[0]	
	TCON_TIM_POLB1	-	-	-	-	-	-	TCON_POLB_HS[10]	TCON_POLB_HS[9]	TCON_POLB_HS[8]
		TCON_POLB_HS[7]	TCON_POLB_HS[6]	TCON_POLB_HS[5]	TCON_POLB_HS[4]	TCON_POLB_HS[3]	TCON_POLB_HS[2]	TCON_POLB_HS[1]	TCON_POLB_HS[0]	
		-	-	-	-	-	TCON_POLB_HW[10]	TCON_POLB_HW[9]	TCON_POLB_HW[8]	
		TCON_POLB_HW[7]	TCON_POLB_HW[6]	TCON_POLB_HW[5]	TCON_POLB_HW[4]	TCON_POLB_HW[3]	TCON_POLB_HW[2]	TCON_POLB_HW[1]	TCON_POLB_HW[0]	
	TCON_TIM_POLB2	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	TCON_POLB_MD[1]	TCON_POLB_MD[0]	-	-	-	TCON_POLB_HS_SEL	
		-	-	-	TCON_POLB_INV	-	TCON_POLB_SEL[2]	TCON_POLB_SEL[1]	TCON_POLB_SEL[0]	
	TCON_TIM_DE	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	TCON_DE_INV	
		-	-	-	-	-	-	-	-	
	OUT_UPDATE	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	OUTCNT_VEN	
		-	-	-	-	-	-	-	-	
	OUT_SET	-	-	-	OUT_ENDIAN_ON	-	-	-	OUT_SWAP_ON	
		-	-	-	-	-	-	-	-	
		-	-	OUT_FORMAT[1]	OUT_FORMAT[0]	-	-	OUT_FRQ_SEL[1]	OUT_FRQ_SEL[0]	
		-	-	-	OUT_DIR_SEL	-	-	OUT_PHASE[1]	OUT_PHASE[0]	
	OUT_BRIGHT1	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	PBRT_G[9]	PBRT_G[8]	
		PBRT_G[7]	PBRT_G[6]	PBRT_G[5]	PBRT_G[4]	PBRT_G[3]	PBRT_G[2]	PBRT_G[1]	PBRT_G[0]	
	OUT_BRIGHT2	-	-	-	-	-	-	PBRT_B[9]	PBRT_B[8]	
		PBRT_B[7]	PBRT_B[6]	PBRT_B[5]	PBRT_B[4]	PBRT_B[3]	PBRT_B[2]	PBRT_B[1]	PBRT_B[0]	
		-	-	-	-	-	-	PBRT_R[9]	PBRT_R[8]	
		PBRT_R[7]	PBRT_R[6]	PBRT_R[5]	PBRT_R[4]	PBRT_R[3]	PBRT_R[2]	PBRT_R[1]	PBRT_R[0]	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 0	OUT_CONTRAST	-	-	-	-	-	-	-	-	
		CONT_G[7]	CONT_G[6]	CONT_G[5]	CONT_G[4]	CONT_G[3]	CONT_G[2]	CONT_G[1]	CONT_G[0]	
		CONT_B[7]	CONT_B[6]	CONT_B[5]	CONT_B[4]	CONT_B[3]	CONT_B[2]	CONT_B[1]	CONT_B[0]	
		CONT_R[7]	CONT_R[6]	CONT_R[5]	CONT_R[4]	CONT_R[3]	CONT_R[2]	CONT_R[1]	CONT_R[0]	
	OUT_PDTHA	-	-	-	-	-	-	-	-	
		-	-	PDTH_SEL[1]	PDTH_SEL[0]	-	-	PDTH_FORMAT[1]	PDTH_FORMAT[0]	
		-	-	PDTH_PA[1]	PDTH_PA[0]	-	-	PDTH_PB[1]	PDTH_PB[0]	
		-	-	PDTH_PC[1]	PDTH_PC[0]	-	-	PDTH_PD[1]	PDTH_PD[0]	
	OUT_CLK_PHASE	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	OUTCNT_FRONT_GAM	-	-	-	OUTCNT_LCD_EDGE	
		-	OUTCNT_STVA_EDGE	OUTCNT_STVB_EDGE	OUTCNT_STH_EDGE	OUTCNT_STB_EDGE	OUTCNT_CPV_EDGE	OUTCNT_POLA_EDGE	OUTCNT_POLB_EDGE	
	SYSCNT_INT1	-	-	-	INT_STA7	-	-	-	INT_STA6	
		-	-	-	INT_STA5	-	-	-	INT_STA4	
		-	-	-	INT_STA3	-	-	-	INT_STA2	
		-	-	-	INT_STA1	-	-	-	INT_STA0	
	SYSCNT_INT2	-	-	-	INT_STA15	-	-	-	INT_STA14	
		-	-	-	INT_STA13	-	-	-	INT_STA12	
		-	-	-	INT_STA11	-	-	-	INT_STA10	
		-	-	-	INT_STA9	-	-	-	INT_STA8	
	SYSCNT_INT3	-	-	-	-	-	-	-	INT_STA22	
		-	-	-	INT_STA21	-	-	-	INT_STA20	
		-	-	-	INT_STA19	-	-	-	INT_STA18	
		-	-	-	INT_STA17	-	-	-	INT_STA16	
	SYSCNT_INT4	-	-	-	INT_OUT7_ON	-	-	-	INT_OUT6_ON	
		-	-	-	INT_OUT5_ON	-	-	-	INT_OUT4_ON	
		-	-	-	INT_OUT3_ON	-	-	-	INT_OUT2_ON	
		-	-	-	INT_OUT1_ON	-	-	-	INT_OUT0_ON	
	SYSCNT_INT5	-	-	-	INT_OUT15_ON	-	-	-	INT_OUT14_ON	
		-	-	-	INT_OUT13_ON	-	-	-	INT_OUT12_ON	
		-	-	-	INT_OUT11_ON	-	-	-	INT_OUT10_ON	
		-	-	-	INT_OUT9_ON	-	-	-	INT_OUT8_ON	
	SYSCNT_INT6	-	-	-	-	-	-	-	INT_OUT22_ON	
		-	-	-	INT_OUT21_ON	-	-	-	INT_OUT20_ON	
		-	-	-	INT_OUT19_ON	-	-	-	INT_OUT18_ON	
		-	-	-	INT_OUT17_ON	-	-	-	INT_OUT16_ON	
	SYSCNT_PANEL_CLK	-	-	PANEL_IJKSEL[1]	PANEL_IJKSEL[0]	PANEL_OCKSEL[1]	PANEL_OCKSEL[0]	-	PANEL_IKEN	
		-	-	PANEL_DCDR[5]	PANEL_DCDR[4]	PANEL_DCDR[3]	PANEL_DCDR[2]	PANEL_DCDR[1]	PANEL_DCDR[0]	
	SYSCNT_CLUT	-	GR_OIR_CLT_SEL_ST	-	GR3_CLT_SEL_ST	-	-	-	GR2_CLT_SEL_ST	
		-	-	-	GR1_CLT_SEL_ST	-	-	-	GR0_CLT_SEL_ST	
	Video display controller 5 channel 1	INP_UPDATE	-	-	-	-	-	-	-	
			-	-	-	-	-	-	-	
			-	-	-	-	-	-	-	
			-	-	-	INP_EXT_UPDATE	-	-	-	INP_IMG_UPDATE
		INP_SEL_CNT	-	-	-	-	-	-	-	-
			-	-	-	INP_SEL	-	-	-	-
			-	INP_FORMAT[2]	INP_FORMAT[1]	INP_FORMAT[0]	-	-	-	INP_PXD_EDGE
			-	-	-	INP_VS_EDGE	-	-	-	INP_HS_EDGE
		INP_EXT_SYNC_CNT	-	-	-	INP_ENDIAN_ON	-	-	-	INP_SWAP_ON
			-	-	-	INP_VS_INV	-	-	-	INP_HS_INV
			-	-	-	-	-	-	-	INP_H_EDGE_SEL
			-	-	-	INP_F525_625	-	-	INP_H_POS[1]	INP_H_POS[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 1	INP_VSYNC_PH_ADJ	-	-	-	-	-	-	INP_FH50[9]	INP_FH50[8]	
		INP_FH50[7]	INP_FH50[6]	INP_FH50[5]	INP_FH50[4]	INP_FH50[3]	INP_FH50[2]	INP_FH50[1]	INP_FH50[0]	
		-	-	-	-	-	-	INP_FH25[9]	INP_FH25[8]	
		INP_FH25[7]	INP_FH25[6]	INP_FH25[5]	INP_FH25[4]	INP_FH25[3]	INP_FH25[2]	INP_FH25[1]	INP_FH25[0]	
	INP_DLY_ADJ	-	-	-	-	-	-	INP_VS_DLY_L[2]	INP_VS_DLY_L[1]	INP_VS_DLY_L[0]
		INP_FLD_DLY[7]	INP_FLD_DLY[6]	INP_FLD_DLY[5]	INP_FLD_DLY[4]	INP_FLD_DLY[3]	INP_FLD_DLY[2]	INP_FLD_DLY[1]	INP_FLD_DLY[0]	
		INP_VS_DLY[7]	INP_VS_DLY[6]	INP_VS_DLY[5]	INP_VS_DLY[4]	INP_VS_DLY[3]	INP_VS_DLY[2]	INP_VS_DLY[1]	INP_VS_DLY[0]	
		INP_HS_DLY[7]	INP_HS_DLY[6]	INP_HS_DLY[5]	INP_HS_DLY[4]	INP_HS_DLY[3]	INP_HS_DLY[2]	INP_HS_DLY[1]	INP_HS_DLY[0]	
	IMGCNT_UPDATE	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	IMGCNT_VEN	
	IMGCNT_NR_CNT0	-	-	-	-	-	-	-	-	
		-	-	-	NR1D_MD	-	-	-	-	NR1D_ON
		-	NR1D_Y_TH[6]	NR1D_Y_TH[5]	NR1D_Y_TH[4]	NR1D_Y_TH[3]	NR1D_Y_TH[2]	NR1D_Y_TH[1]	NR1D_Y_TH[0]	
		-	-	NR1D_Y_TAP[1]	NR1D_Y_TAP[0]	-	-	NR1D_Y_GAIN[1]	NR1D_Y_GAIN[0]	
	IMGCNT_NR_CNT1	-	NR1D_CB_TH[6]	NR1D_CB_TH[5]	NR1D_CB_TH[4]	NR1D_CB_TH[3]	NR1D_CB_TH[2]	NR1D_CB_TH[1]	NR1D_CB_TH[0]	
		-	-	NR1D_CB_TAP[1]	NR1D_CB_TAP[0]	-	-	NR1D_CB_GAIN[1]	NR1D_CB_GAIN[0]	
		-	NR1D_CR_TH[6]	NR1D_CR_TH[5]	NR1D_CR_TH[4]	NR1D_CR_TH[3]	NR1D_CR_TH[2]	NR1D_CR_TH[1]	NR1D_CR_TH[0]	
		-	-	NR1D_CR_TAP[1]	NR1D_CR_TAP[0]	-	-	NR1D_CR_GAIN[1]	NR1D_CR_GAIN[0]	
	IMGCNT_MTX_MODE	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	IMGCNT_MTX_MD[1]	IMGCNT_MTX_MD[0]	
	IMGCNT_MTX_YG_ADJ0	-	-	-	-	-	-	-	-	
		IMGCNT_MTX_YG[7]	IMGCNT_MTX_YG[6]	IMGCNT_MTX_YG[5]	IMGCNT_MTX_YG[4]	IMGCNT_MTX_YG[3]	IMGCNT_MTX_YG[2]	IMGCNT_MTX_YG[1]	IMGCNT_MTX_YG[0]	
		-	-	-	-	-	IMGCNT_MTX_GG[10]	IMGCNT_MTX_GG[9]	IMGCNT_MTX_GG[8]	
		IMGCNT_MTX_GG[7]	IMGCNT_MTX_GG[6]	IMGCNT_MTX_GG[5]	IMGCNT_MTX_GG[4]	IMGCNT_MTX_GG[3]	IMGCNT_MTX_GG[2]	IMGCNT_MTX_GG[1]	IMGCNT_MTX_GG[0]	
	IMGCNT_MTX_YG_ADJ1	-	-	-	-	-	IMGCNT_MTX_GB[10]	IMGCNT_MTX_GB[9]	IMGCNT_MTX_GB[8]	
		IMGCNT_MTX_GB[7]	IMGCNT_MTX_GB[6]	IMGCNT_MTX_GB[5]	IMGCNT_MTX_GB[4]	IMGCNT_MTX_GB[3]	IMGCNT_MTX_GB[2]	IMGCNT_MTX_GB[1]	IMGCNT_MTX_GB[0]	
		-	-	-	-	-	IMGCNT_MTX_GR[10]	IMGCNT_MTX_GR[9]	IMGCNT_MTX_GR[8]	
		IMGCNT_MTX_GR[7]	IMGCNT_MTX_GR[6]	IMGCNT_MTX_GR[5]	IMGCNT_MTX_GR[4]	IMGCNT_MTX_GR[3]	IMGCNT_MTX_GR[2]	IMGCNT_MTX_GR[1]	IMGCNT_MTX_GR[0]	
	IMGCNT_MTX_CBB_ADJ0	-	-	-	-	-	-	-	-	
		IMGCNT_MTX_B[7]	IMGCNT_MTX_B[6]	IMGCNT_MTX_B[5]	IMGCNT_MTX_B[4]	IMGCNT_MTX_B[3]	IMGCNT_MTX_B[2]	IMGCNT_MTX_B[1]	IMGCNT_MTX_B[0]	
		-	-	-	-	-	IMGCNT_MTX_BG[10]	IMGCNT_MTX_BG[9]	IMGCNT_MTX_BG[8]	
		IMGCNT_MTX_BG[7]	IMGCNT_MTX_BG[6]	IMGCNT_MTX_BG[5]	IMGCNT_MTX_BG[4]	IMGCNT_MTX_BG[3]	IMGCNT_MTX_BG[2]	IMGCNT_MTX_BG[1]	IMGCNT_MTX_BG[0]	
	IMGCNT_MTX_CBB_ADJ1	-	-	-	-	-	IMGCNT_MTX_BB[10]	IMGCNT_MTX_BB[9]	IMGCNT_MTX_BB[8]	
		IMGCNT_MTX_BB[7]	IMGCNT_MTX_BB[6]	IMGCNT_MTX_BB[5]	IMGCNT_MTX_BB[4]	IMGCNT_MTX_BB[3]	IMGCNT_MTX_BB[2]	IMGCNT_MTX_BB[1]	IMGCNT_MTX_BB[0]	
		-	-	-	-	-	IMGCNT_MTX_BR[10]	IMGCNT_MTX_BR[9]	IMGCNT_MTX_BR[8]	
		IMGCNT_MTX_BR[7]	IMGCNT_MTX_BR[6]	IMGCNT_MTX_BR[5]	IMGCNT_MTX_BR[4]	IMGCNT_MTX_BR[3]	IMGCNT_MTX_BR[2]	IMGCNT_MTX_BR[1]	IMGCNT_MTX_BR[0]	
	IMGCNT_MTX_CRR_ADJ0	-	-	-	-	-	-	-	-	
		IMGCNT_MTX_R[7]	IMGCNT_MTX_R[6]	IMGCNT_MTX_R[5]	IMGCNT_MTX_R[4]	IMGCNT_MTX_R[3]	IMGCNT_MTX_R[2]	IMGCNT_MTX_R[1]	IMGCNT_MTX_R[0]	
		-	-	-	-	-	IMGCNT_MTX_RG[10]	IMGCNT_MTX_RG[9]	IMGCNT_MTX_RG[8]	
		IMGCNT_MTX_RG[7]	IMGCNT_MTX_RG[6]	IMGCNT_MTX_RG[5]	IMGCNT_MTX_RG[4]	IMGCNT_MTX_RG[3]	IMGCNT_MTX_RG[2]	IMGCNT_MTX_RG[1]	IMGCNT_MTX_RG[0]	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Video display controller 5 channel 1	IMGCNT_MTX_CRR_ADJ1	-	-	-	-	-	IMGCNT_MTX_RB[10]	IMGCNT_MTX_RB[9]	IMGCNT_MTX_RB[8]
		IMGCNT_MTX_RB[7]	IMGCNT_MTX_RB[6]	IMGCNT_MTX_RB[5]	IMGCNT_MTX_RB[4]	IMGCNT_MTX_RB[3]	IMGCNT_MTX_RB[2]	IMGCNT_MTX_RB[1]	IMGCNT_MTX_RB[0]
		-	-	-	-	-	IMGCNT_MTX_RR[10]	IMGCNT_MTX_RR[9]	IMGCNT_MTX_RR[8]
		IMGCNT_MTX_RR[7]	IMGCNT_MTX_RR[6]	IMGCNT_MTX_RR[5]	IMGCNT_MTX_RR[4]	IMGCNT_MTX_RR[3]	IMGCNT_MTX_RR[2]	IMGCNT_MTX_RR[1]	IMGCNT_MTX_RR[0]
	IMGCNT_DRC_REG	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	DRC_EN
	SC0_SCL0_UPDATE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	SC0_SCL0_VEN_D	SC0_SCL0_VEN_C	-	-	-	SC0_SCL0_UPDATE
		-	-	-	SC0_SCL0_VEN_B	-	-	-	SC0_SCL0_VEN_A
	SC0_SCL0_FRC1	SC0_RES_VMASK[15]	SC0_RES_VMASK[14]	SC0_RES_VMASK[13]	SC0_RES_VMASK[12]	SC0_RES_VMASK[11]	SC0_RES_VMASK[10]	SC0_RES_VMASK[9]	SC0_RES_VMASK[8]
		SC0_RES_VMASK[7]	SC0_RES_VMASK[6]	SC0_RES_VMASK[5]	SC0_RES_VMASK[4]	SC0_RES_VMASK[3]	SC0_RES_VMASK[2]	SC0_RES_VMASK[1]	SC0_RES_VMASK[0]
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	SC0_RES_VMASK_ON
	SC0_SCL0_FRC2	SC0_RES_VLACK[15]	SC0_RES_VLACK[14]	SC0_RES_VLACK[13]	SC0_RES_VLACK[12]	SC0_RES_VLACK[11]	SC0_RES_VLACK[10]	SC0_RES_VLACK[9]	SC0_RES_VLACK[8]
		SC0_RES_VLACK[7]	SC0_RES_VLACK[6]	SC0_RES_VLACK[5]	SC0_RES_VLACK[4]	SC0_RES_VLACK[3]	SC0_RES_VLACK[2]	SC0_RES_VLACK[1]	SC0_RES_VLACK[0]
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	SC0_RES_VLACK_ON
	SC0_SCL0_FRC3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	SC0_RES_VS_IN_SE
		-	-	-	-	-	-	-	SC0_RES_VS_SEL
	SC0_SCL0_FRC4	-	-	-	-	-	SC0_RES_FV[10]	SC0_RES_FV[9]	SC0_RES_FV[8]
		SC0_RES_FV[7]	SC0_RES_FV[6]	SC0_RES_FV[5]	SC0_RES_FV[4]	SC0_RES_FV[3]	SC0_RES_FV[2]	SC0_RES_FV[1]	SC0_RES_FV[0]
		-	-	-	-	-	SC0_RES_FH[10]	SC0_RES_FH[9]	SC0_RES_FH[8]
		SC0_RES_FH[7]	SC0_RES_FH[6]	SC0_RES_FH[5]	SC0_RES_FH[4]	SC0_RES_FH[3]	SC0_RES_FH[2]	SC0_RES_FH[1]	SC0_RES_FH[0]
	SC0_SCL0_FRC5	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	SC0_RES_FLD_DLY_SEL
		SC0_RES_VSDLY[7]	SC0_RES_VSDLY[6]	SC0_RES_VSDLY[5]	SC0_RES_VSDLY[4]	SC0_RES_VSDLY[3]	SC0_RES_VSDLY[2]	SC0_RES_VSDLY[1]	SC0_RES_VSDLY[0]
	SC0_SCL0_FRC6	-	-	-	-	-	SC0_RES_F_VS[10]	SC0_RES_F_VS[9]	SC0_RES_F_VS[8]
		SC0_RES_F_VS[7]	SC0_RES_F_VS[6]	SC0_RES_F_VS[5]	SC0_RES_F_VS[4]	SC0_RES_F_VS[3]	SC0_RES_F_VS[2]	SC0_RES_F_VS[1]	SC0_RES_F_VS[0]
		-	-	-	-	-	SC0_RES_F_VW[10]	SC0_RES_F_VW[9]	SC0_RES_F_VW[8]
		SC0_RES_F_VW[7]	SC0_RES_F_VW[6]	SC0_RES_F_VW[5]	SC0_RES_F_VW[4]	SC0_RES_F_VW[3]	SC0_RES_F_VW[2]	SC0_RES_F_VW[1]	SC0_RES_F_VW[0]
	SC0_SCL0_FRC7	-	-	-	-	-	SC0_RES_F_HS[10]	SC0_RES_F_HS[9]	SC0_RES_F_HS[8]
		SC0_RES_F_HS[7]	SC0_RES_F_HS[6]	SC0_RES_F_HS[5]	SC0_RES_F_HS[4]	SC0_RES_F_HS[3]	SC0_RES_F_HS[2]	SC0_RES_F_HS[1]	SC0_RES_F_HS[0]
		-	-	-	-	-	SC0_RES_F_HW[10]	SC0_RES_F_HW[9]	SC0_RES_F_HW[8]
		SC0_RES_F_HW[7]	SC0_RES_F_HW[6]	SC0_RES_F_HW[5]	SC0_RES_F_HW[4]	SC0_RES_F_HW[3]	SC0_RES_F_HW[2]	SC0_RES_F_HW[1]	SC0_RES_F_HW[0]
	SC0_SCL0_FRC9	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	SC0_RES_QVLOCK	-	-	-	SC0_RES_QVLACK

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 1	SC0_SCL0_MON0	-	-	-	-	-	SC0_RES_LIN_STAT[10]	SC0_RES_LIN_STAT[9]	SC0_RES_LIN_STAT[8]	
		SC0_RES_LIN_STAT[7]	SC0_RES_LIN_STAT[6]	SC0_RES_LIN_STAT[5]	SC0_RES_LIN_STAT[4]	SC0_RES_LIN_STAT[3]	SC0_RES_LIN_STAT[2]	SC0_RES_LIN_STAT[1]	SC0_RES_LIN_STAT[0]	
	SC0_SCL0_INT	-	-	-	-	-	-	SC0_RES_LINE[10]	SC0_RES_LINE[9]	SC0_RES_LINE[8]
		SC0_RES_LINE[7]	SC0_RES_LINE[6]	SC0_RES_LINE[5]	SC0_RES_LINE[4]	SC0_RES_LINE[3]	SC0_RES_LINE[2]	SC0_RES_LINE[1]	SC0_RES_LINE[0]	
	SC0_SCL0_DS1	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	SC0_RES_DS_V_ON	-	-	-	-	SC0_RES_DS_H_ON
	SC0_SCL0_DS2	-	-	-	-	-	-	SC0_RES_VS[10]	SC0_RES_VS[9]	SC0_RES_VS[8]
		SC0_RES_VS[7]	SC0_RES_VS[6]	SC0_RES_VS[5]	SC0_RES_VS[4]	SC0_RES_VS[3]	SC0_RES_VS[2]	SC0_RES_VS[1]	SC0_RES_VS[0]	
		-	-	-	-	-	-	SC0_RES_VW[10]	SC0_RES_VW[9]	SC0_RES_VW[8]
		SC0_RES_VW[7]	SC0_RES_VW[6]	SC0_RES_VW[5]	SC0_RES_VW[4]	SC0_RES_VW[3]	SC0_RES_VW[2]	SC0_RES_VW[1]	SC0_RES_VW[0]	
	SC0_SCL0_DS3	-	-	-	-	-	-	SC0_RES_HS[10]	SC0_RES_HS[9]	SC0_RES_HS[8]
		SC0_RES_HS[7]	SC0_RES_HS[6]	SC0_RES_HS[5]	SC0_RES_HS[4]	SC0_RES_HS[3]	SC0_RES_HS[2]	SC0_RES_HS[1]	SC0_RES_HS[0]	
		-	-	-	-	-	-	SC0_RES_HW[10]	SC0_RES_HW[9]	SC0_RES_HW[8]
		SC0_RES_HW[7]	SC0_RES_HW[6]	SC0_RES_HW[5]	SC0_RES_HW[4]	SC0_RES_HW[3]	SC0_RES_HW[2]	SC0_RES_HW[1]	SC0_RES_HW[0]	
	SC0_SCL0_DS4	-	-	SC0_RES_PFIL_SEL	SC0_RES_DS_H_INT_ERPOTYP	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		SC0_RES_DS_H_RATIO[15]	SC0_RES_DS_H_RATIO[14]	SC0_RES_DS_H_RATIO[13]	SC0_RES_DS_H_RATIO[12]	SC0_RES_DS_H_RATIO[11]	SC0_RES_DS_H_RATIO[10]	SC0_RES_DS_H_RATIO[9]	SC0_RES_DS_H_RATIO[8]	
		SC0_RES_DS_H_RATIO[7]	SC0_RES_DS_H_RATIO[6]	SC0_RES_DS_H_RATIO[5]	SC0_RES_DS_H_RATIO[4]	SC0_RES_DS_H_RATIO[3]	SC0_RES_DS_H_RATIO[2]	SC0_RES_DS_H_RATIO[1]	SC0_RES_DS_H_RATIO[0]	
	SC0_SCL0_DS5	-	-	-	SC0_RES_V_INTERPOTYP	SC0_RES_TOP_INIPHASE[11]	SC0_RES_TOP_INIPHASE[10]	SC0_RES_TOP_INIPHASE[9]	SC0_RES_TOP_INIPHASE[8]	
		SC0_RES_TOP_INIPHASE[7]	SC0_RES_TOP_INIPHASE[6]	SC0_RES_TOP_INIPHASE[5]	SC0_RES_TOP_INIPHASE[4]	SC0_RES_TOP_INIPHASE[3]	SC0_RES_TOP_INIPHASE[2]	SC0_RES_TOP_INIPHASE[1]	SC0_RES_TOP_INIPHASE[0]	
		-	-	-	-	SC0_RES_BTM_INIPHASE[11]	SC0_RES_BTM_INIPHASE[10]	SC0_RES_BTM_INIPHASE[9]	SC0_RES_BTM_INIPHASE[8]	
		SC0_RES_BTM_INIPHASE[7]	SC0_RES_BTM_INIPHASE[6]	SC0_RES_BTM_INIPHASE[5]	SC0_RES_BTM_INIPHASE[4]	SC0_RES_BTM_INIPHASE[3]	SC0_RES_BTM_INIPHASE[2]	SC0_RES_BTM_INIPHASE[1]	SC0_RES_BTM_INIPHASE[0]	
	SC0_SCL0_DS6	-	-	-	-	-	-	-	-	
		SC0_RES_V_RATIO[15]	SC0_RES_V_RATIO[14]	SC0_RES_V_RATIO[13]	SC0_RES_V_RATIO[12]	SC0_RES_V_RATIO[11]	SC0_RES_V_RATIO[10]	SC0_RES_V_RATIO[9]	SC0_RES_V_RATIO[8]	
		SC0_RES_V_RATIO[7]	SC0_RES_V_RATIO[6]	SC0_RES_V_RATIO[5]	SC0_RES_V_RATIO[4]	SC0_RES_V_RATIO[3]	SC0_RES_V_RATIO[2]	SC0_RES_V_RATIO[1]	SC0_RES_V_RATIO[0]	
	SC0_SCL0_DS7	-	-	-	-	-	-	SC0_RES_OUT_VW[10]	SC0_RES_OUT_VW[9]	SC0_RES_OUT_VW[8]
		SC0_RES_OUT_VW[7]	SC0_RES_OUT_VW[6]	SC0_RES_OUT_VW[5]	SC0_RES_OUT_VW[4]	SC0_RES_OUT_VW[3]	SC0_RES_OUT_VW[2]	SC0_RES_OUT_VW[1]	SC0_RES_OUT_VW[0]	
		-	-	-	-	-	-	SC0_RES_OUT_HW[10]	SC0_RES_OUT_HW[9]	SC0_RES_OUT_HW[8]
		SC0_RES_OUT_HW[7]	SC0_RES_OUT_HW[6]	SC0_RES_OUT_HW[5]	SC0_RES_OUT_HW[4]	SC0_RES_OUT_HW[3]	SC0_RES_OUT_HW[2]	SC0_RES_OUT_HW[1]	SC0_RES_OUT_HW[0]	
	SC0_SCL0_US1	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	SC0_RES_US_V_ON	-	-	-	-	SC0_RES_US_H_ON
	SC0_SCL0_US2	-	-	-	-	-	-	SC0_RES_P_VS[10]	SC0_RES_P_VS[9]	SC0_RES_P_VS[8]
		SC0_RES_P_VS[7]	SC0_RES_P_VS[6]	SC0_RES_P_VS[5]	SC0_RES_P_VS[4]	SC0_RES_P_VS[3]	SC0_RES_P_VS[2]	SC0_RES_P_VS[1]	SC0_RES_P_VS[0]	
		-	-	-	-	-	-	SC0_RES_P_VW[10]	SC0_RES_P_VW[9]	SC0_RES_P_VW[8]
		SC0_RES_P_VW[7]	SC0_RES_P_VW[6]	SC0_RES_P_VW[5]	SC0_RES_P_VW[4]	SC0_RES_P_VW[3]	SC0_RES_P_VW[2]	SC0_RES_P_VW[1]	SC0_RES_P_VW[0]	
	SC0_SCL0_US3	-	-	-	-	-	-	SC0_RES_P_HS[10]	SC0_RES_P_HS[9]	SC0_RES_P_HS[8]
		SC0_RES_P_HS[7]	SC0_RES_P_HS[6]	SC0_RES_P_HS[5]	SC0_RES_P_HS[4]	SC0_RES_P_HS[3]	SC0_RES_P_HS[2]	SC0_RES_P_HS[1]	SC0_RES_P_HS[0]	
		-	-	-	-	-	-	SC0_RES_P_HW[10]	SC0_RES_P_HW[9]	SC0_RES_P_HW[8]
		SC0_RES_P_HW[7]	SC0_RES_P_HW[6]	SC0_RES_P_HW[5]	SC0_RES_P_HW[4]	SC0_RES_P_HW[3]	SC0_RES_P_HW[2]	SC0_RES_P_HW[1]	SC0_RES_P_HW[0]	

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Video display controller 5 channel 1	SC0_SCL0_US4	-	-	-	-	-	SC0_RES_IN_VW[10]	SC0_RES_IN_VW[9]	SC0_RES_IN_VW[8]
		SC0_RES_IN_VW[7]	SC0_RES_IN_VW[6]	SC0_RES_IN_VW[5]	SC0_RES_IN_VW[4]	SC0_RES_IN_VW[3]	SC0_RES_IN_VW[2]	SC0_RES_IN_VW[1]	SC0_RES_IN_VW[0]
		-	-	-	-	-	SC0_RES_IN_HW[10]	SC0_RES_IN_HW[9]	SC0_RES_IN_HW[8]
		SC0_RES_IN_HW[7]	SC0_RES_IN_HW[6]	SC0_RES_IN_HW[5]	SC0_RES_IN_HW[4]	SC0_RES_IN_HW[3]	SC0_RES_IN_HW[2]	SC0_RES_IN_HW[1]	SC0_RES_IN_HW[0]
	SC0_SCL0_US5	-	-	-	-	-	-	-	-
		SC0_RES_US_H_RA_TIO[15]	SC0_RES_US_H_RA_TIO[14]	SC0_RES_US_H_RA_TIO[13]	SC0_RES_US_H_RA_TIO[12]	SC0_RES_US_H_RA_TIO[11]	SC0_RES_US_H_RA_TIO[10]	SC0_RES_US_H_RA_TIO[9]	SC0_RES_US_H_RA_TIO[8]
		SC0_RES_US_H_RA_TIO[7]	SC0_RES_US_H_RA_TIO[6]	SC0_RES_US_H_RA_TIO[5]	SC0_RES_US_H_RA_TIO[4]	SC0_RES_US_H_RA_TIO[3]	SC0_RES_US_H_RA_TIO[2]	SC0_RES_US_H_RA_TIO[1]	SC0_RES_US_H_RA_TIO[0]
	SC0_SCL0_US6	-	-	-	SC0_RES_US_H_INT_ERPOTYP	SC0_RES_US_HT_INIPHASE[11]	SC0_RES_US_HT_INIPHASE[10]	SC0_RES_US_HT_INIPHASE[9]	SC0_RES_US_HT_INIPHASE[8]
		SC0_RES_US_HT_INIPHASE[7]	SC0_RES_US_HT_INIPHASE[6]	SC0_RES_US_HT_INIPHASE[5]	SC0_RES_US_HT_INIPHASE[4]	SC0_RES_US_HT_INIPHASE[3]	SC0_RES_US_HT_INIPHASE[2]	SC0_RES_US_HT_INIPHASE[1]	SC0_RES_US_HT_INIPHASE[0]
		-	-	-	-	SC0_RES_US_HB_INIPHASE[11]	SC0_RES_US_HB_INIPHASE[10]	SC0_RES_US_HB_INIPHASE[9]	SC0_RES_US_HB_INIPHASE[8]
SC0_RES_US_HB_INIPHASE[7]		SC0_RES_US_HB_INIPHASE[6]	SC0_RES_US_HB_INIPHASE[5]	SC0_RES_US_HB_INIPHASE[4]	SC0_RES_US_HB_INIPHASE[3]	SC0_RES_US_HB_INIPHASE[2]	SC0_RES_US_HB_INIPHASE[1]	SC0_RES_US_HB_INIPHASE[0]	
SC0_SCL0_US7	-	-	-	-	-	-	-	-	
	SC0_RES_HCURT [7]	SC0_RES_HCURT [6]	SC0_RES_HCURT [5]	SC0_RES_HCURT [4]	SC0_RES_HCURT [3]	SC0_RES_HCURT [2]	SC0_RES_HCURT [1]	SC0_RES_HCURT [0]	
	SC0_RES_VCURT [7]	SC0_RES_VCURT [6]	SC0_RES_VCURT [5]	SC0_RES_VCURT [4]	SC0_RES_VCURT [3]	SC0_RES_VCURT [2]	SC0_RES_VCURT [1]	SC0_RES_VCURT [0]	
SC0_SCL0_US8	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	
	-	-	-	SC0_RES_IBUS_SYN_C_SEL	-	-	-	-	SC0_RES_DISP_ON
SC0_SCL0_OVR1	-	-	-	-	-	-	-	-	
	SC0_RES_BK_COL_R[7]	SC0_RES_BK_COL_R[6]	SC0_RES_BK_COL_R[5]	SC0_RES_BK_COL_R[4]	SC0_RES_BK_COL_R[3]	SC0_RES_BK_COL_R[2]	SC0_RES_BK_COL_R[1]	SC0_RES_BK_COL_R[0]	
	SC0_RES_BK_COL_G[7]	SC0_RES_BK_COL_G[6]	SC0_RES_BK_COL_G[5]	SC0_RES_BK_COL_G[4]	SC0_RES_BK_COL_G[3]	SC0_RES_BK_COL_G[2]	SC0_RES_BK_COL_G[1]	SC0_RES_BK_COL_G[0]	
SC0_SCL1_UPDATE	-	-	-	-	-	-	-	-	
	-	-	-	SC0_SCL1_UPDATE_B	-	-	-	SC0_SCL1_UPDATE_A	
	-	-	-	-	-	-	-	-	
	-	-	-	SC0_SCL1_VEN_B	-	-	-	SC0_SCL1_VEN_A	
SC0_SCL1_WR1	-	-	-	-	-	-	-	-	
	-	-	-	-	-	SC0_RES_WRSWA[2]	SC0_RES_WRSWA[1]	SC0_RES_WRSWA[0]	
	SC0_RES_TB_ADD_MOD	SC0_RES_DS_WR_MD[2]	SC0_RES_DS_WR_MD[1]	SC0_RES_DS_WR_MD[0]	SC0_RES_MD[1]	SC0_RES_MD[0]	SC0_RES_LOOP	SC0_RES_BST_MD	
SC0_SCL1_WR2	SC0_RES_BASE [31]	SC0_RES_BASE [30]	SC0_RES_BASE [29]	SC0_RES_BASE [28]	SC0_RES_BASE [27]	SC0_RES_BASE [26]	SC0_RES_BASE [25]	SC0_RES_BASE [24]	
	SC0_RES_BASE [23]	SC0_RES_BASE [22]	SC0_RES_BASE [21]	SC0_RES_BASE [20]	SC0_RES_BASE [19]	SC0_RES_BASE [18]	SC0_RES_BASE [17]	SC0_RES_BASE [16]	
	SC0_RES_BASE [15]	SC0_RES_BASE [14]	SC0_RES_BASE [13]	SC0_RES_BASE [12]	SC0_RES_BASE [11]	SC0_RES_BASE [10]	SC0_RES_BASE [9]	SC0_RES_BASE [8]	
	SC0_RES_BASE [7]	SC0_RES_BASE [6]	SC0_RES_BASE [5]	SC0_RES_BASE [4]	SC0_RES_BASE [3]	SC0_RES_BASE [2]	SC0_RES_BASE [1]	SC0_RES_BASE [0]	
SC0_SCL1_WR3	-	SC0_RES_LN_OFF[14]	SC0_RES_LN_OFF[13]	SC0_RES_LN_OFF[12]	SC0_RES_LN_OFF[11]	SC0_RES_LN_OFF[10]	SC0_RES_LN_OFF[9]	SC0_RES_LN_OFF[8]	
	SC0_RES_LN_OFF[7]	SC0_RES_LN_OFF[6]	SC0_RES_LN_OFF[5]	SC0_RES_LN_OFF[4]	SC0_RES_LN_OFF[3]	SC0_RES_LN_OFF[2]	SC0_RES_LN_OFF[1]	SC0_RES_LN_OFF[0]	
	-	-	-	-	-	-	SC0_RES_FLM_NUM [9]	SC0_RES_FLM_NUM [8]	
	SC0_RES_FLM_NUM [7]	SC0_RES_FLM_NUM [6]	SC0_RES_FLM_NUM [5]	SC0_RES_FLM_NUM [4]	SC0_RES_FLM_NUM [3]	SC0_RES_FLM_NUM [2]	SC0_RES_FLM_NUM [1]	SC0_RES_FLM_NUM [0]	

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Video display controller 5 channel 1	SC0_SCL1_WR4	-	-	-	-	-	-	-	-
		-	SC0_RES_FLM_OF F[22]	SC0_RES_FLM_OF F[21]	SC0_RES_FLM_OF F[20]	SC0_RES_FLM_OF F[19]	SC0_RES_FLM_OF F[18]	SC0_RES_FLM_OF F[17]	SC0_RES_FLM_OF F[16]
		SC0_RES_FLM_OF F[15]	SC0_RES_FLM_OF F[14]	SC0_RES_FLM_OF F[13]	SC0_RES_FLM_OF F[12]	SC0_RES_FLM_OF F[11]	SC0_RES_FLM_OF F[10]	SC0_RES_FLM_OF F[9]	SC0_RES_FLM_OF F[8]
		SC0_RES_FLM_OF F[7]	SC0_RES_FLM_OF F[6]	SC0_RES_FLM_OF F[5]	SC0_RES_FLM_OF F[4]	SC0_RES_FLM_OF F[3]	SC0_RES_FLM_OF F[2]	SC0_RES_FLM_OF F[1]	SC0_RES_FLM_OF F[0]
	SC0_SCL1_WR5	-	-	-	-	-	-	-	-
		-	-	-	SC0_RES_INTER	-	-	SC0_RES_FS_RATE[1]	SC0_RES_FS_RATE[0]
		-	-	-	SC0_RES_FLD_SEL	-	-	-	SC0_RES_WENB
	SC0_SCL1_WR6	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	SC0_RES_DTH_ON	-	-	-	SC0_RES_BITDEC_ON
	SC0_SCL1_WR7	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	SC0_RES_OVERFLOW
		-	-	-	-	-	-	SC0_RES_FLM_CN T[9]	SC0_RES_FLM_CN T[8]
	SC0_SCL1_WR8	SC0_RES_BASE_B[31]	SC0_RES_BASE_B[30]	SC0_RES_BASE_B[29]	SC0_RES_BASE_B[28]	SC0_RES_BASE_B[27]	SC0_RES_BASE_B[26]	SC0_RES_BASE_B[25]	SC0_RES_BASE_B[24]
		SC0_RES_BASE_B[23]	SC0_RES_BASE_B[22]	SC0_RES_BASE_B[21]	SC0_RES_BASE_B[20]	SC0_RES_BASE_B[19]	SC0_RES_BASE_B[18]	SC0_RES_BASE_B[17]	SC0_RES_BASE_B[16]
		SC0_RES_BASE_B[15]	SC0_RES_BASE_B[14]	SC0_RES_BASE_B[13]	SC0_RES_BASE_B[12]	SC0_RES_BASE_B[11]	SC0_RES_BASE_B[10]	SC0_RES_BASE_B[9]	SC0_RES_BASE_B[8]
		SC0_RES_BASE_B[7]	SC0_RES_BASE_B[6]	SC0_RES_BASE_B[5]	SC0_RES_BASE_B[4]	SC0_RES_BASE_B[3]	SC0_RES_BASE_B[2]	SC0_RES_BASE_B[1]	SC0_RES_BASE_B[0]
	SC0_SCL1_WR9	-	SC0_RES_LN_OFF_B[14]	SC0_RES_LN_OFF_B[13]	SC0_RES_LN_OFF_B[12]	SC0_RES_LN_OFF_B[11]	SC0_RES_LN_OFF_B[10]	SC0_RES_LN_OFF_B[9]	SC0_RES_LN_OFF_B[8]
		SC0_RES_LN_OFF_B[7]	SC0_RES_LN_OFF_B[6]	SC0_RES_LN_OFF_B[5]	SC0_RES_LN_OFF_B[4]	SC0_RES_LN_OFF_B[3]	SC0_RES_LN_OFF_B[2]	SC0_RES_LN_OFF_B[1]	SC0_RES_LN_OFF_B[0]
		-	-	-	-	-	-	SC0_RES_FLM_NUM_B[9]	SC0_RES_FLM_NUM_B[8]
		SC0_RES_FLM_NUM_B[7]	SC0_RES_FLM_NUM_B[6]	SC0_RES_FLM_NUM_B[5]	SC0_RES_FLM_NUM_B[4]	SC0_RES_FLM_NUM_B[3]	SC0_RES_FLM_NUM_B[2]	SC0_RES_FLM_NUM_B[1]	SC0_RES_FLM_NUM_B[0]
	SC0_SCL1_WR10	-	-	-	-	-	-	-	-
		-	SC0_RES_FLM_OF F_B[22]	SC0_RES_FLM_OF F_B[21]	SC0_RES_FLM_OF F_B[20]	SC0_RES_FLM_OF F_B[19]	SC0_RES_FLM_OF F_B[18]	SC0_RES_FLM_OF F_B[17]	SC0_RES_FLM_OF F_B[16]
		SC0_RES_FLM_OF F_B[15]	SC0_RES_FLM_OF F_B[14]	SC0_RES_FLM_OF F_B[13]	SC0_RES_FLM_OF F_B[12]	SC0_RES_FLM_OF F_B[11]	SC0_RES_FLM_OF F_B[10]	SC0_RES_FLM_OF F_B[9]	SC0_RES_FLM_OF F_B[8]
		SC0_RES_FLM_OF F_B[7]	SC0_RES_FLM_OF F_B[6]	SC0_RES_FLM_OF F_B[5]	SC0_RES_FLM_OF F_B[4]	SC0_RES_FLM_OF F_B[3]	SC0_RES_FLM_OF F_B[2]	SC0_RES_FLM_OF F_B[1]	SC0_RES_FLM_OF F_B[0]
	SC0_SCL1_WR11	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	SC0_RES_FLM_CN T_B[9]	SC0_RES_FLM_CN T_B[8]
	SC0_SCL1_MON1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	SC0_PBUF_NUM[1]	SC0_PBUF_NUM[0]
		-	-	-	-	-	-	-	-
	SC0_SCL1_PBUF0	SC0_PBUF0_ADD[31]	SC0_PBUF0_ADD[30]	SC0_PBUF0_ADD[29]	SC0_PBUF0_ADD[28]	SC0_PBUF0_ADD[27]	SC0_PBUF0_ADD[26]	SC0_PBUF0_ADD[25]	SC0_PBUF0_ADD[24]
		SC0_PBUF0_ADD[23]	SC0_PBUF0_ADD[22]	SC0_PBUF0_ADD[21]	SC0_PBUF0_ADD[20]	SC0_PBUF0_ADD[19]	SC0_PBUF0_ADD[18]	SC0_PBUF0_ADD[17]	SC0_PBUF0_ADD[16]
		SC0_PBUF0_ADD[15]	SC0_PBUF0_ADD[14]	SC0_PBUF0_ADD[13]	SC0_PBUF0_ADD[12]	SC0_PBUF0_ADD[11]	SC0_PBUF0_ADD[10]	SC0_PBUF0_ADD[9]	SC0_PBUF0_ADD[8]
		SC0_PBUF0_ADD[7]	SC0_PBUF0_ADD[6]	SC0_PBUF0_ADD[5]	SC0_PBUF0_ADD[4]	SC0_PBUF0_ADD[3]	SC0_PBUF0_ADD[2]	SC0_PBUF0_ADD[1]	SC0_PBUF0_ADD[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 1	SC0_SCL1_PBUF1	SC0_PBUF1_ADD[31]	SC0_PBUF1_ADD[30]	SC0_PBUF1_ADD[29]	SC0_PBUF1_ADD[28]	SC0_PBUF1_ADD[27]	SC0_PBUF1_ADD[26]	SC0_PBUF1_ADD[25]	SC0_PBUF1_ADD[24]	
		SC0_PBUF1_ADD[23]	SC0_PBUF1_ADD[22]	SC0_PBUF1_ADD[21]	SC0_PBUF1_ADD[20]	SC0_PBUF1_ADD[19]	SC0_PBUF1_ADD[18]	SC0_PBUF1_ADD[17]	SC0_PBUF1_ADD[16]	
		SC0_PBUF1_ADD[15]	SC0_PBUF1_ADD[14]	SC0_PBUF1_ADD[13]	SC0_PBUF1_ADD[12]	SC0_PBUF1_ADD[11]	SC0_PBUF1_ADD[10]	SC0_PBUF1_ADD[9]	SC0_PBUF1_ADD[8]	
		SC0_PBUF1_ADD[7]	SC0_PBUF1_ADD[6]	SC0_PBUF1_ADD[5]	SC0_PBUF1_ADD[4]	SC0_PBUF1_ADD[3]	SC0_PBUF1_ADD[2]	SC0_PBUF1_ADD[1]	SC0_PBUF1_ADD[0]	
	SC0_SCL1_PBUF2	SC0_PBUF2_ADD[31]	SC0_PBUF2_ADD[30]	SC0_PBUF2_ADD[29]	SC0_PBUF2_ADD[28]	SC0_PBUF2_ADD[27]	SC0_PBUF2_ADD[26]	SC0_PBUF2_ADD[25]	SC0_PBUF2_ADD[24]	
		SC0_PBUF2_ADD[23]	SC0_PBUF2_ADD[22]	SC0_PBUF2_ADD[21]	SC0_PBUF2_ADD[20]	SC0_PBUF2_ADD[19]	SC0_PBUF2_ADD[18]	SC0_PBUF2_ADD[17]	SC0_PBUF2_ADD[16]	
		SC0_PBUF2_ADD[15]	SC0_PBUF2_ADD[14]	SC0_PBUF2_ADD[13]	SC0_PBUF2_ADD[12]	SC0_PBUF2_ADD[11]	SC0_PBUF2_ADD[10]	SC0_PBUF2_ADD[9]	SC0_PBUF2_ADD[8]	
		SC0_PBUF2_ADD[7]	SC0_PBUF2_ADD[6]	SC0_PBUF2_ADD[5]	SC0_PBUF2_ADD[4]	SC0_PBUF2_ADD[3]	SC0_PBUF2_ADD[2]	SC0_PBUF2_ADD[1]	SC0_PBUF2_ADD[0]	
	SC0_SCL1_PBUF3	SC0_PBUF3_ADD[31]	SC0_PBUF3_ADD[30]	SC0_PBUF3_ADD[29]	SC0_PBUF3_ADD[28]	SC0_PBUF3_ADD[27]	SC0_PBUF3_ADD[26]	SC0_PBUF3_ADD[25]	SC0_PBUF3_ADD[24]	
		SC0_PBUF3_ADD[23]	SC0_PBUF3_ADD[22]	SC0_PBUF3_ADD[21]	SC0_PBUF3_ADD[20]	SC0_PBUF3_ADD[19]	SC0_PBUF3_ADD[18]	SC0_PBUF3_ADD[17]	SC0_PBUF3_ADD[16]	
		SC0_PBUF3_ADD[15]	SC0_PBUF3_ADD[14]	SC0_PBUF3_ADD[13]	SC0_PBUF3_ADD[12]	SC0_PBUF3_ADD[11]	SC0_PBUF3_ADD[10]	SC0_PBUF3_ADD[9]	SC0_PBUF3_ADD[8]	
		SC0_PBUF3_ADD[7]	SC0_PBUF3_ADD[6]	SC0_PBUF3_ADD[5]	SC0_PBUF3_ADD[4]	SC0_PBUF3_ADD[3]	SC0_PBUF3_ADD[2]	SC0_PBUF3_ADD[1]	SC0_PBUF3_ADD[0]	
	SC0_SCL1_PBUF_FLD	-	-	-	-	-	-	-	-	SC0_FLD_INF3
		-	-	-	-	-	-	-	-	SC0_FLD_INF2
		-	-	-	-	-	-	-	-	SC0_FLD_INF1
		-	-	-	-	-	-	-	-	SC0_FLD_INF0
	SC0_SCL1_PBUF_CNT	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	SC0_PBUF_RST
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
	GR0_UPDATE	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	GR0_UPDATE
		-	-	-	GR0_P_VEN	-	-	-	-	GR0_IBUS_VEN
	GR0_FLM_RD	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	GR0_R_ENB
	GR0_FLM1	GR0_FLD_SEL	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	GR0_LN_OFF_DIR
		-	-	-	-	-	-	GR0_FLM_SEL[1]	GR0_FLM_SEL[0]	-
		-	-	-	GR0_IMR_FLM_INV	-	-	-	-	GR0_BST_MD
	GR0_FLM2	GR0_BASE[31]	GR0_BASE[30]	GR0_BASE[29]	GR0_BASE[28]	GR0_BASE[27]	GR0_BASE[26]	GR0_BASE[25]	GR0_BASE[24]	
		GR0_BASE[23]	GR0_BASE[22]	GR0_BASE[21]	GR0_BASE[20]	GR0_BASE[19]	GR0_BASE[18]	GR0_BASE[17]	GR0_BASE[16]	
		GR0_BASE[15]	GR0_BASE[14]	GR0_BASE[13]	GR0_BASE[12]	GR0_BASE[11]	GR0_BASE[10]	GR0_BASE[9]	GR0_BASE[8]	
		GR0_BASE[7]	GR0_BASE[6]	GR0_BASE[5]	GR0_BASE[4]	GR0_BASE[3]	GR0_BASE[2]	GR0_BASE[1]	GR0_BASE[0]	
	GR0_FLM3	GR0_FLD_NXT	GR0_LN_OFF[14]	GR0_LN_OFF[13]	GR0_LN_OFF[12]	GR0_LN_OFF[11]	GR0_LN_OFF[10]	GR0_LN_OFF[9]	GR0_LN_OFF[8]	
		GR0_LN_OFF[7]	GR0_LN_OFF[6]	GR0_LN_OFF[5]	GR0_LN_OFF[4]	GR0_LN_OFF[3]	GR0_LN_OFF[2]	GR0_LN_OFF[1]	GR0_LN_OFF[0]	
		-	-	-	-	-	-	GR0_FLM_NUM[9]	GR0_FLM_NUM[8]	
		GR0_FLM_NUM[7]	GR0_FLM_NUM[6]	GR0_FLM_NUM[5]	GR0_FLM_NUM[4]	GR0_FLM_NUM[3]	GR0_FLM_NUM[2]	GR0_FLM_NUM[1]	GR0_FLM_NUM[0]	
	GR0_FLM4	-	-	-	-	-	-	-	-	
		-	GR0_FLM_OFF[22]	GR0_FLM_OFF[21]	GR0_FLM_OFF[20]	GR0_FLM_OFF[19]	GR0_FLM_OFF[18]	GR0_FLM_OFF[17]	GR0_FLM_OFF[16]	
		GR0_FLM_OFF[15]	GR0_FLM_OFF[14]	GR0_FLM_OFF[13]	GR0_FLM_OFF[12]	GR0_FLM_OFF[11]	GR0_FLM_OFF[10]	GR0_FLM_OFF[9]	GR0_FLM_OFF[8]	
		GR0_FLM_OFF[7]	GR0_FLM_OFF[6]	GR0_FLM_OFF[5]	GR0_FLM_OFF[4]	GR0_FLM_OFF[3]	GR0_FLM_OFF[2]	GR0_FLM_OFF[1]	GR0_FLM_OFF[0]	



Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Video display controller 5 channel 1	GR0_FLM5	-	-	-	-	-	GR0_FLM_LNUM[10]	GR0_FLM_LNUM[9]	GR0_FLM_LNUM[8]
		GR0_FLM_LNUM[7]	GR0_FLM_LNUM[6]	GR0_FLM_LNUM[5]	GR0_FLM_LNUM[4]	GR0_FLM_LNUM[3]	GR0_FLM_LNUM[2]	GR0_FLM_LNUM[1]	GR0_FLM_LNUM[0]
		-	-	-	-	-	GR0_FLM_LOOP[10]	GR0_FLM_LOOP[9]	GR0_FLM_LOOP[8]
		GR0_FLM_LOOP[7]	GR0_FLM_LOOP[6]	GR0_FLM_LOOP[5]	GR0_FLM_LOOP[4]	GR0_FLM_LOOP[3]	GR0_FLM_LOOP[2]	GR0_FLM_LOOP[1]	GR0_FLM_LOOP[0]
	GR0_FLM6	GR0_FORMAT[3]	GR0_FORMAT[2]	GR0_FORMAT[1]	GR0_FORMAT[0]	-	GR0_HW[10]	GR0_HW[9]	GR0_HW[8]
		GR0_HW[7]	GR0_HW[6]	GR0_HW[5]	GR0_HW[4]	GR0_HW[3]	GR0_HW[2]	GR0_HW[1]	GR0_HW[0]
		GR0_YCC_SWAP[2]	GR0_YCC_SWAP[1]	GR0_YCC_SWAP[0]	GR0_RDSWA[2]	GR0_RDSWA[1]	GR0_RDSWA[0]	-	GR0_CNV444_MD
		-	-	GR0_STA_POS[5]	GR0_STA_POS[4]	GR0_STA_POS[3]	GR0_STA_POS[2]	GR0_STA_POS[1]	GR0_STA_POS[0]
	GR0_AB1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	GR0_GRC_DISP_ON	-	-	GR0_DISP_SEL[1]	GR0_DISP_SEL[0]
	GR0_AB2	-	-	-	-	-	GR0_GRC_VS[10]	GR0_GRC_VS[9]	GR0_GRC_VS[8]
		GR0_GRC_VS[7]	GR0_GRC_VS[6]	GR0_GRC_VS[5]	GR0_GRC_VS[4]	GR0_GRC_VS[3]	GR0_GRC_VS[2]	GR0_GRC_VS[1]	GR0_GRC_VS[0]
		-	-	-	-	-	GR0_GRC_VW[10]	GR0_GRC_VW[9]	GR0_GRC_VW[8]
		GR0_GRC_VW[7]	GR0_GRC_VW[6]	GR0_GRC_VW[5]	GR0_GRC_VW[4]	GR0_GRC_VW[3]	GR0_GRC_VW[2]	GR0_GRC_VW[1]	GR0_GRC_VW[0]
	GR0_AB3	-	-	-	-	-	GR0_GRC_HS[10]	GR0_GRC_HS[9]	GR0_GRC_HS[8]
		GR0_GRC_HS[7]	GR0_GRC_HS[6]	GR0_GRC_HS[5]	GR0_GRC_HS[4]	GR0_GRC_HS[3]	GR0_GRC_HS[2]	GR0_GRC_HS[1]	GR0_GRC_HS[0]
		-	-	-	-	-	GR0_GRC_HW[10]	GR0_GRC_HW[9]	GR0_GRC_HW[8]
		GR0_GRC_HW[7]	GR0_GRC_HW[6]	GR0_GRC_HW[5]	GR0_GRC_HW[4]	GR0_GRC_HW[3]	GR0_GRC_HW[2]	GR0_GRC_HW[1]	GR0_GRC_HW[0]
	GR0_AB7	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GR0_CK_ON
	GR0_AB8	GR0_CK_KCLUT[7]	GR0_CK_KCLUT[6]	GR0_CK_KCLUT[5]	GR0_CK_KCLUT[4]	GR0_CK_KCLUT[3]	GR0_CK_KCLUT[2]	GR0_CK_KCLUT[1]	GR0_CK_KCLUT[0]
		GR0_CK_KG[7]	GR0_CK_KG[6]	GR0_CK_KG[5]	GR0_CK_KG[4]	GR0_CK_KG[3]	GR0_CK_KG[2]	GR0_CK_KG[1]	GR0_CK_KG[0]
		GR0_CK_KB[7]	GR0_CK_KB[6]	GR0_CK_KB[5]	GR0_CK_KB[4]	GR0_CK_KB[3]	GR0_CK_KB[2]	GR0_CK_KB[1]	GR0_CK_KB[0]
		GR0_CK_KR[7]	GR0_CK_KR[6]	GR0_CK_KR[5]	GR0_CK_KR[4]	GR0_CK_KR[3]	GR0_CK_KR[2]	GR0_CK_KR[1]	GR0_CK_KR[0]
	GR0_AB9	GR0_CK_A[7]	GR0_CK_A[6]	GR0_CK_A[5]	GR0_CK_A[4]	GR0_CK_A[3]	GR0_CK_A[2]	GR0_CK_A[1]	GR0_CK_A[0]
		GR0_CK_G[7]	GR0_CK_G[6]	GR0_CK_G[5]	GR0_CK_G[4]	GR0_CK_G[3]	GR0_CK_G[2]	GR0_CK_G[1]	GR0_CK_G[0]
		GR0_CK_B[7]	GR0_CK_B[6]	GR0_CK_B[5]	GR0_CK_B[4]	GR0_CK_B[3]	GR0_CK_B[2]	GR0_CK_B[1]	GR0_CK_B[0]
		GR0_CK_R[7]	GR0_CK_R[6]	GR0_CK_R[5]	GR0_CK_R[4]	GR0_CK_R[3]	GR0_CK_R[2]	GR0_CK_R[1]	GR0_CK_R[0]
	GR0_AB10	GR0_A0[7]	GR0_A0[6]	GR0_A0[5]	GR0_A0[4]	GR0_A0[3]	GR0_A0[2]	GR0_A0[1]	GR0_A0[0]
		GR0_G0[7]	GR0_G0[6]	GR0_G0[5]	GR0_G0[4]	GR0_G0[3]	GR0_G0[2]	GR0_G0[1]	GR0_G0[0]
		GR0_B0[7]	GR0_B0[6]	GR0_B0[5]	GR0_B0[4]	GR0_B0[3]	GR0_B0[2]	GR0_B0[1]	GR0_B0[0]
		GR0_R0[7]	GR0_R0[6]	GR0_R0[5]	GR0_R0[4]	GR0_R0[3]	GR0_R0[2]	GR0_R0[1]	GR0_R0[0]
	GR0_AB11	GR0_A1[7]	GR0_A1[6]	GR0_A1[5]	GR0_A1[4]	GR0_A1[3]	GR0_A1[2]	GR0_A1[1]	GR0_A1[0]
		GR0_G1[7]	GR0_G1[6]	GR0_G1[5]	GR0_G1[4]	GR0_G1[3]	GR0_G1[2]	GR0_G1[1]	GR0_G1[0]
		GR0_B1[7]	GR0_B1[6]	GR0_B1[5]	GR0_B1[4]	GR0_B1[3]	GR0_B1[2]	GR0_B1[1]	GR0_B1[0]
		GR0_R1[7]	GR0_R1[6]	GR0_R1[5]	GR0_R1[4]	GR0_R1[3]	GR0_R1[2]	GR0_R1[1]	GR0_R1[0]
	GR0_BASE	-	-	-	-	-	-	-	-
		GR0_BASE_G[7]	GR0_BASE_G[6]	GR0_BASE_G[5]	GR0_BASE_G[4]	GR0_BASE_G[3]	GR0_BASE_G[2]	GR0_BASE_G[1]	GR0_BASE_G[0]
		GR0_BASE_B[7]	GR0_BASE_B[6]	GR0_BASE_B[5]	GR0_BASE_B[4]	GR0_BASE_B[3]	GR0_BASE_B[2]	GR0_BASE_B[1]	GR0_BASE_B[0]
	GR0_CLUT	GR0_BASE_R[7]	GR0_BASE_R[6]	GR0_BASE_R[5]	GR0_BASE_R[4]	GR0_BASE_R[3]	GR0_BASE_R[2]	GR0_BASE_R[1]	GR0_BASE_R[0]
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GR0_CLT_SEL
		-	-	-	-	-	-	-	-
	SC1_SCL0_UPDATE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	SC1_SCL0_VEN_D	SC1_SCL0_VEN_C	-	-	-	SC1_SCL0_UPDATE
		-	-	-	SC1_SCL0_VEN_B	-	-	-	SC1_SCL0_VEN_A

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 1	SC1_SCL0_FRC1	SC1_RES_VMASK[15]	SC1_RES_VMASK[14]	SC1_RES_VMASK[13]	SC1_RES_VMASK[12]	SC1_RES_VMASK[11]	SC1_RES_VMASK[10]	SC1_RES_VMASK[9]	SC1_RES_VMASK[8]	
		SC1_RES_VMASK[7]	SC1_RES_VMASK[6]	SC1_RES_VMASK[5]	SC1_RES_VMASK[4]	SC1_RES_VMASK[3]	SC1_RES_VMASK[2]	SC1_RES_VMASK[1]	SC1_RES_VMASK[0]	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	SC1_RES_VMASK_ON	
	SC1_SCL0_FRC2	SC1_RES_VLACK[15]	SC1_RES_VLACK[14]	SC1_RES_VLACK[13]	SC1_RES_VLACK[12]	SC1_RES_VLACK[11]	SC1_RES_VLACK[10]	SC1_RES_VLACK[9]	SC1_RES_VLACK[8]	
		SC1_RES_VLACK[7]	SC1_RES_VLACK[6]	SC1_RES_VLACK[5]	SC1_RES_VLACK[4]	SC1_RES_VLACK[3]	SC1_RES_VLACK[2]	SC1_RES_VLACK[1]	SC1_RES_VLACK[0]	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	SC1_RES_VLACK_ON	
	SC1_SCL0_FRC3	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	SC1_RES_VS_IN_SEL	
	SC1_SCL0_FRC4	-	-	-	-	-	-	SC1_RES_FV[10]	SC1_RES_FV[9]	SC1_RES_FV[8]
		SC1_RES_FV[7]	SC1_RES_FV[6]	SC1_RES_FV[5]	SC1_RES_FV[4]	SC1_RES_FV[3]	SC1_RES_FV[2]	SC1_RES_FV[1]	SC1_RES_FV[0]	
		-	-	-	-	-	SC1_RES_FH[10]	SC1_RES_FH[9]	SC1_RES_FH[8]	
		SC1_RES_FH[7]	SC1_RES_FH[6]	SC1_RES_FH[5]	SC1_RES_FH[4]	SC1_RES_FH[3]	SC1_RES_FH[2]	SC1_RES_FH[1]	SC1_RES_FH[0]	
	SC1_SCL0_FRC5	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	SC1_RES_FLD_DLY_SEL	
		SC1_RES_VSDLY[7]	SC1_RES_VSDLY[6]	SC1_RES_VSDLY[5]	SC1_RES_VSDLY[4]	SC1_RES_VSDLY[3]	SC1_RES_VSDLY[2]	SC1_RES_VSDLY[1]	SC1_RES_VSDLY[0]	
	SC1_SCL0_FRC6	-	-	-	-	-	-	SC1_RES_F_VS[10]	SC1_RES_F_VS[9]	SC1_RES_F_VS[8]
		SC1_RES_F_VS[7]	SC1_RES_F_VS[6]	SC1_RES_F_VS[5]	SC1_RES_F_VS[4]	SC1_RES_F_VS[3]	SC1_RES_F_VS[2]	SC1_RES_F_VS[1]	SC1_RES_F_VS[0]	
		-	-	-	-	-	SC1_RES_F_VW[10]	SC1_RES_F_VW[9]	SC1_RES_F_VW[8]	
		SC1_RES_F_VW[7]	SC1_RES_F_VW[6]	SC1_RES_F_VW[5]	SC1_RES_F_VW[4]	SC1_RES_F_VW[3]	SC1_RES_F_VW[2]	SC1_RES_F_VW[1]	SC1_RES_F_VW[0]	
	SC1_SCL0_FRC7	-	-	-	-	-	-	SC1_RES_F_HS[10]	SC1_RES_F_HS[9]	SC1_RES_F_HS[8]
		SC1_RES_F_HS[7]	SC1_RES_F_HS[6]	SC1_RES_F_HS[5]	SC1_RES_F_HS[4]	SC1_RES_F_HS[3]	SC1_RES_F_HS[2]	SC1_RES_F_HS[1]	SC1_RES_F_HS[0]	
		-	-	-	-	-	SC1_RES_F_HW[10]	SC1_RES_F_HW[9]	SC1_RES_F_HW[8]	
		SC1_RES_F_HW[7]	SC1_RES_F_HW[6]	SC1_RES_F_HW[5]	SC1_RES_F_HW[4]	SC1_RES_F_HW[3]	SC1_RES_F_HW[2]	SC1_RES_F_HW[1]	SC1_RES_F_HW[0]	
	SC1_SCL0_FRC9	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	SC1_RES_QVLOCK	-	-	-	-	SC1_RES_QVLOCK
	SC1_SCL0_MON0	-	-	-	-	-	-	SC1_RES_LIN_STAT[10]	SC1_RES_LIN_STAT[9]	SC1_RES_LIN_STAT[8]
		SC1_RES_LIN_STAT[7]	SC1_RES_LIN_STAT[6]	SC1_RES_LIN_STAT[5]	SC1_RES_LIN_STAT[4]	SC1_RES_LIN_STAT[3]	SC1_RES_LIN_STAT[2]	SC1_RES_LIN_STAT[1]	SC1_RES_LIN_STAT[0]	
	SC1_SCL0_INT	-	-	-	-	-	-	SC1_RES_LINE[10]	SC1_RES_LINE[9]	SC1_RES_LINE[8]
		SC1_RES_LINE[7]	SC1_RES_LINE[6]	SC1_RES_LINE[5]	SC1_RES_LINE[4]	SC1_RES_LINE[3]	SC1_RES_LINE[2]	SC1_RES_LINE[1]	SC1_RES_LINE[0]	
	SC1_SCL0_DS1	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	SC1_RES_DS_V_ON	-	-	-	-	SC1_RES_DS_H_ON
	SC1_SCL0_DS2	-	-	-	-	-	-	SC1_RES_VS[10]	SC1_RES_VS[9]	SC1_RES_VS[8]
		SC1_RES_VS[7]	SC1_RES_VS[6]	SC1_RES_VS[5]	SC1_RES_VS[4]	SC1_RES_VS[3]	SC1_RES_VS[2]	SC1_RES_VS[1]	SC1_RES_VS[0]	
		-	-	-	-	-	SC1_RES_VW[10]	SC1_RES_VW[9]	SC1_RES_VW[8]	
		SC1_RES_VW[7]	SC1_RES_VW[6]	SC1_RES_VW[5]	SC1_RES_VW[4]	SC1_RES_VW[3]	SC1_RES_VW[2]	SC1_RES_VW[1]	SC1_RES_VW[0]	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 1	SC1_SCL0_DS3	-	-	-	-	-	SC1_RES_HS[10]	SC1_RES_HS[9]	SC1_RES_HS[8]	
		SC1_RES_HS[7]	SC1_RES_HS[6]	SC1_RES_HS[5]	SC1_RES_HS[4]	SC1_RES_HS[3]	SC1_RES_HS[2]	SC1_RES_HS[1]	SC1_RES_HS[0]	
		-	-	-	-	-	SC1_RES_HW[10]	SC1_RES_HW[9]	SC1_RES_HW[8]	
		SC1_RES_HW[7]	SC1_RES_HW[6]	SC1_RES_HW[5]	SC1_RES_HW[4]	SC1_RES_HW[3]	SC1_RES_HW[2]	SC1_RES_HW[1]	SC1_RES_HW[0]	
	SC1_SCL0_DS4	-	-	SC1_RES_PFIL_SEL	SC1_RES_DS_H_INT ERPOTYP	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		SC1_RES_DS_H_RATIO[15]	SC1_RES_DS_H_RATIO[14]	SC1_RES_DS_H_RATIO[13]	SC1_RES_DS_H_RATIO[12]	SC1_RES_DS_H_RATIO[11]	SC1_RES_DS_H_RATIO[10]	SC1_RES_DS_H_RATIO[9]	SC1_RES_DS_H_RATIO[8]	
	SC1_SCL0_DS5	SC1_RES_DS_H_RATIO[7]	SC1_RES_DS_H_RATIO[6]	SC1_RES_DS_H_RATIO[5]	SC1_RES_DS_H_RATIO[4]	SC1_RES_DS_H_RATIO[3]	SC1_RES_DS_H_RATIO[2]	SC1_RES_DS_H_RATIO[1]	SC1_RES_DS_H_RATIO[0]	
		-	-	-	SC1_RES_V_INTERPOTYP	SC1_RES_TOP_INIPHASE[11]	SC1_RES_TOP_INIPHASE[10]	SC1_RES_TOP_INIPHASE[9]	SC1_RES_TOP_INIPHASE[8]	
		SC1_RES_TOP_INIPHASE[7]	SC1_RES_TOP_INIPHASE[6]	SC1_RES_TOP_INIPHASE[5]	SC1_RES_TOP_INIPHASE[4]	SC1_RES_TOP_INIPHASE[3]	SC1_RES_TOP_INIPHASE[2]	SC1_RES_TOP_INIPHASE[1]	SC1_RES_TOP_INIPHASE[0]	
		-	-	-	-	SC1_RES_BTM_INIPHASE[11]	SC1_RES_BTM_INIPHASE[10]	SC1_RES_BTM_INIPHASE[9]	SC1_RES_BTM_INIPHASE[8]	
	SC1_SCL0_DS6	SC1_RES_BTM_INIPHASE[7]	SC1_RES_BTM_INIPHASE[6]	SC1_RES_BTM_INIPHASE[5]	SC1_RES_BTM_INIPHASE[4]	SC1_RES_BTM_INIPHASE[3]	SC1_RES_BTM_INIPHASE[2]	SC1_RES_BTM_INIPHASE[1]	SC1_RES_BTM_INIPHASE[0]	
		-	-	-	-	-	-	-	-	
		SC1_RES_V_RATIO[15]	SC1_RES_V_RATIO[14]	SC1_RES_V_RATIO[13]	SC1_RES_V_RATIO[12]	SC1_RES_V_RATIO[11]	SC1_RES_V_RATIO[10]	SC1_RES_V_RATIO[9]	SC1_RES_V_RATIO[8]	
	SC1_SCL0_DS7	SC1_RES_V_RATIO[7]	SC1_RES_V_RATIO[6]	SC1_RES_V_RATIO[5]	SC1_RES_V_RATIO[4]	SC1_RES_V_RATIO[3]	SC1_RES_V_RATIO[2]	SC1_RES_V_RATIO[1]	SC1_RES_V_RATIO[0]	
		-	-	-	-	-	SC1_RES_OUT_VW[10]	SC1_RES_OUT_VW[9]	SC1_RES_OUT_VW[8]	
		SC1_RES_OUT_VW[7]	SC1_RES_OUT_VW[6]	SC1_RES_OUT_VW[5]	SC1_RES_OUT_VW[4]	SC1_RES_OUT_VW[3]	SC1_RES_OUT_VW[2]	SC1_RES_OUT_VW[1]	SC1_RES_OUT_VW[0]	
		-	-	-	-	-	SC1_RES_OUT_HW[10]	SC1_RES_OUT_HW[9]	SC1_RES_OUT_HW[8]	
	SC1_SCL0_US1	SC1_RES_OUT_HW[7]	SC1_RES_OUT_HW[6]	SC1_RES_OUT_HW[5]	SC1_RES_OUT_HW[4]	SC1_RES_OUT_HW[3]	SC1_RES_OUT_HW[2]	SC1_RES_OUT_HW[1]	SC1_RES_OUT_HW[0]	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	SC1_RES_US_V_ON	-	-	-	-	SC1_RES_US_H_ON
	SC1_SCL0_US2	-	-	-	-	-	SC1_RES_P_VS[10]	SC1_RES_P_VS[9]	SC1_RES_P_VS[8]	
		SC1_RES_P_VS[7]	SC1_RES_P_VS[6]	SC1_RES_P_VS[5]	SC1_RES_P_VS[4]	SC1_RES_P_VS[3]	SC1_RES_P_VS[2]	SC1_RES_P_VS[1]	SC1_RES_P_VS[0]	
		-	-	-	-	-	SC1_RES_P_VW[10]	SC1_RES_P_VW[9]	SC1_RES_P_VW[8]	
		SC1_RES_P_VW[7]	SC1_RES_P_VW[6]	SC1_RES_P_VW[5]	SC1_RES_P_VW[4]	SC1_RES_P_VW[3]	SC1_RES_P_VW[2]	SC1_RES_P_VW[1]	SC1_RES_P_VW[0]	
	SC1_SCL0_US3	-	-	-	-	-	SC1_RES_P_HS[10]	SC1_RES_P_HS[9]	SC1_RES_P_HS[8]	
		SC1_RES_P_HS[7]	SC1_RES_P_HS[6]	SC1_RES_P_HS[5]	SC1_RES_P_HS[4]	SC1_RES_P_HS[3]	SC1_RES_P_HS[2]	SC1_RES_P_HS[1]	SC1_RES_P_HS[0]	
		-	-	-	-	-	SC1_RES_P_HW[10]	SC1_RES_P_HW[9]	SC1_RES_P_HW[8]	
		SC1_RES_P_HW[7]	SC1_RES_P_HW[6]	SC1_RES_P_HW[5]	SC1_RES_P_HW[4]	SC1_RES_P_HW[3]	SC1_RES_P_HW[2]	SC1_RES_P_HW[1]	SC1_RES_P_HW[0]	
	SC1_SCL0_US4	-	-	-	-	-	SC1_RES_IN_VW[10]	SC1_RES_IN_VW[9]	SC1_RES_IN_VW[8]	
		SC1_RES_IN_VW[7]	SC1_RES_IN_VW[6]	SC1_RES_IN_VW[5]	SC1_RES_IN_VW[4]	SC1_RES_IN_VW[3]	SC1_RES_IN_VW[2]	SC1_RES_IN_VW[1]	SC1_RES_IN_VW[0]	
		-	-	-	-	-	SC1_RES_IN_HW[10]	SC1_RES_IN_HW[9]	SC1_RES_IN_HW[8]	
		SC1_RES_IN_HW[7]	SC1_RES_IN_HW[6]	SC1_RES_IN_HW[5]	SC1_RES_IN_HW[4]	SC1_RES_IN_HW[3]	SC1_RES_IN_HW[2]	SC1_RES_IN_HW[1]	SC1_RES_IN_HW[0]	
	SC1_SCL0_US5	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		SC1_RES_US_H_RATIO[15]	SC1_RES_US_H_RATIO[14]	SC1_RES_US_H_RATIO[13]	SC1_RES_US_H_RATIO[12]	SC1_RES_US_H_RATIO[11]	SC1_RES_US_H_RATIO[10]	SC1_RES_US_H_RATIO[9]	SC1_RES_US_H_RATIO[8]	
		SC1_RES_US_H_RATIO[7]	SC1_RES_US_H_RATIO[6]	SC1_RES_US_H_RATIO[5]	SC1_RES_US_H_RATIO[4]	SC1_RES_US_H_RATIO[3]	SC1_RES_US_H_RATIO[2]	SC1_RES_US_H_RATIO[1]	SC1_RES_US_H_RATIO[0]	

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 1	SC1_SCL0_US6	-	-	-	SC1_RES_US_H_INT ERPOTYP	SC1_RES_US_HT_INIPHASE[11]	SC1_RES_US_HT_INIPHASE[10]	SC1_RES_US_HT_INIPHASE[9]	SC1_RES_US_HT_INIPHASE[8]	
		SC1_RES_US_HT_INIPHASE[7]	SC1_RES_US_HT_INIPHASE[6]	SC1_RES_US_HT_INIPHASE[5]	SC1_RES_US_HT_INIPHASE[4]	SC1_RES_US_HT_INIPHASE[3]	SC1_RES_US_HT_INIPHASE[2]	SC1_RES_US_HT_INIPHASE[1]	SC1_RES_US_HT_INIPHASE[0]	
		-	-	-	-	SC1_RES_US_HB_INIPHASE[11]	SC1_RES_US_HB_INIPHASE[10]	SC1_RES_US_HB_INIPHASE[9]	SC1_RES_US_HB_INIPHASE[8]	
		SC1_RES_US_HB_INIPHASE[7]	SC1_RES_US_HB_INIPHASE[6]	SC1_RES_US_HB_INIPHASE[5]	SC1_RES_US_HB_INIPHASE[4]	SC1_RES_US_HB_INIPHASE[3]	SC1_RES_US_HB_INIPHASE[2]	SC1_RES_US_HB_INIPHASE[1]	SC1_RES_US_HB_INIPHASE[0]	
	SC1_SCL0_US7	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		SC1_RES_HCUT [7]	SC1_RES_HCUT [6]	SC1_RES_HCUT [5]	SC1_RES_HCUT [4]	SC1_RES_HCUT [3]	SC1_RES_HCUT [2]	SC1_RES_HCUT [1]	SC1_RES_HCUT [0]	
		SC1_RES_VCUT [7]	SC1_RES_VCUT [6]	SC1_RES_VCUT [5]	SC1_RES_VCUT [4]	SC1_RES_VCUT [3]	SC1_RES_VCUT [2]	SC1_RES_VCUT [1]	SC1_RES_VCUT [0]	
	SC1_SCL0_US8	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	SC1_RES_IBUS_SYN C_SEL	-	-	-	SC1_RES_DISP_ON	
	SC1_SCL0_OVR1	-	-	-	-	-	-	-	-	
		SC1_RES_BK_COL_R[7]	SC1_RES_BK_COL_R[6]	SC1_RES_BK_COL_R[5]	SC1_RES_BK_COL_R[4]	SC1_RES_BK_COL_R[3]	SC1_RES_BK_COL_R[2]	SC1_RES_BK_COL_R[1]	SC1_RES_BK_COL_R[0]	
		SC1_RES_BK_COL_G[7]	SC1_RES_BK_COL_G[6]	SC1_RES_BK_COL_G[5]	SC1_RES_BK_COL_G[4]	SC1_RES_BK_COL_G[3]	SC1_RES_BK_COL_G[2]	SC1_RES_BK_COL_G[1]	SC1_RES_BK_COL_G[0]	
		SC1_RES_BK_COL_B[7]	SC1_RES_BK_COL_B[6]	SC1_RES_BK_COL_B[5]	SC1_RES_BK_COL_B[4]	SC1_RES_BK_COL_B[3]	SC1_RES_BK_COL_B[2]	SC1_RES_BK_COL_B[1]	SC1_RES_BK_COL_B[0]	
	SC1_SCL1_UPDATE	-	-	-	-	-	-	-	-	
		-	-	-	SC1_SCL1_UPDATE_B	-	-	-	SC1_SCL1_UPDATE_A	
		-	-	-	-	-	-	-	-	
		-	-	-	SC1_SCL1_VEN_B	-	-	-	SC1_SCL1_VEN_A	
	SC1_SCL1_WR1	-	-	-	-	-	-	SC1_RES_WRSWA[2]	SC1_RES_WRSWA[1]	SC1_RES_WRSWA[0]
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		SC1_RES_TB_ADD_MOD	SC1_RES_DS_WR_MD[2]	SC1_RES_DS_WR_MD[1]	SC1_RES_DS_WR_MD[0]	SC1_RES_MD[1]	SC1_RES_MD[0]	SC1_RES_LOOP	SC1_RES_BST_MD	
	SC1_SCL1_WR2	SC1_RES_BASE [31]	SC1_RES_BASE [30]	SC1_RES_BASE [29]	SC1_RES_BASE [28]	SC1_RES_BASE [27]	SC1_RES_BASE [26]	SC1_RES_BASE [25]	SC1_RES_BASE [24]	
		SC1_RES_BASE [23]	SC1_RES_BASE [22]	SC1_RES_BASE [21]	SC1_RES_BASE [20]	SC1_RES_BASE [19]	SC1_RES_BASE [18]	SC1_RES_BASE [17]	SC1_RES_BASE [16]	
		SC1_RES_BASE [15]	SC1_RES_BASE [14]	SC1_RES_BASE [13]	SC1_RES_BASE [12]	SC1_RES_BASE [11]	SC1_RES_BASE [10]	SC1_RES_BASE [9]	SC1_RES_BASE [8]	
		SC1_RES_BASE [7]	SC1_RES_BASE [6]	SC1_RES_BASE [5]	SC1_RES_BASE [4]	SC1_RES_BASE [3]	SC1_RES_BASE [2]	SC1_RES_BASE [1]	SC1_RES_BASE [0]	
	SC1_SCL1_WR3	-	SC1_RES_LN_OFF[14]	SC1_RES_LN_OFF[13]	SC1_RES_LN_OFF[12]	SC1_RES_LN_OFF[11]	SC1_RES_LN_OFF[10]	SC1_RES_LN_OFF[9]	SC1_RES_LN_OFF[8]	
		SC1_RES_LN_OFF[7]	SC1_RES_LN_OFF[6]	SC1_RES_LN_OFF[5]	SC1_RES_LN_OFF[4]	SC1_RES_LN_OFF[3]	SC1_RES_LN_OFF[2]	SC1_RES_LN_OFF[1]	SC1_RES_LN_OFF[0]	
		-	-	-	-	-	-	SC1_RES_FLM_NUM [9]	SC1_RES_FLM_NUM [8]	
		SC1_RES_FLM_NUM [7]	SC1_RES_FLM_NUM [6]	SC1_RES_FLM_NUM [5]	SC1_RES_FLM_NUM [4]	SC1_RES_FLM_NUM [3]	SC1_RES_FLM_NUM [2]	SC1_RES_FLM_NUM [1]	SC1_RES_FLM_NUM [0]	
	SC1_SCL1_WR4	-	-	-	-	-	-	-	-	
		-	SC1_RES_FLM_OFF[22]	SC1_RES_FLM_OFF[21]	SC1_RES_FLM_OFF[20]	SC1_RES_FLM_OFF[19]	SC1_RES_FLM_OFF[18]	SC1_RES_FLM_OFF[17]	SC1_RES_FLM_OFF[16]	
		SC1_RES_FLM_OFF[15]	SC1_RES_FLM_OFF[14]	SC1_RES_FLM_OFF[13]	SC1_RES_FLM_OFF[12]	SC1_RES_FLM_OFF[11]	SC1_RES_FLM_OFF[10]	SC1_RES_FLM_OFF[9]	SC1_RES_FLM_OFF[8]	
		SC1_RES_FLM_OFF[7]	SC1_RES_FLM_OFF[6]	SC1_RES_FLM_OFF[5]	SC1_RES_FLM_OFF[4]	SC1_RES_FLM_OFF[3]	SC1_RES_FLM_OFF[2]	SC1_RES_FLM_OFF[1]	SC1_RES_FLM_OFF[0]	
	SC1_SCL1_WR5	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	SC1_RES_INTER	-	-	SC1_RES_FS_RATE[1]	SC1_RES_FS_RATE[0]	
		-	-	-	SC1_RES_FLD_SEL	-	-	-	SC1_RES_WENB	

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 1	SC1_SCL1_WR6	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	SC1_RES_DTH_ON	-	-	-	-	SC1_RES_BITDEC_ON
	SC1_SCL1_WR7	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	SC1_RES_OVERFLOW
		-	-	-	-	-	-	SC1_RES_FLM_CNT[9]	SC1_RES_FLM_CNT[8]	SC1_RES_FLM_CNT[7]
		SC1_RES_FLM_CNT[7]	SC1_RES_FLM_CNT[6]	SC1_RES_FLM_CNT[5]	SC1_RES_FLM_CNT[4]	SC1_RES_FLM_CNT[3]	SC1_RES_FLM_CNT[2]	SC1_RES_FLM_CNT[1]	SC1_RES_FLM_CNT[0]	
	SC1_SCL1_WR8	SC1_RES_BASE_B[31]	SC1_RES_BASE_B[30]	SC1_RES_BASE_B[29]	SC1_RES_BASE_B[28]	SC1_RES_BASE_B[27]	SC1_RES_BASE_B[26]	SC1_RES_BASE_B[25]	SC1_RES_BASE_B[24]	SC1_RES_BASE_B[23]
		SC1_RES_BASE_B[23]	SC1_RES_BASE_B[22]	SC1_RES_BASE_B[21]	SC1_RES_BASE_B[20]	SC1_RES_BASE_B[19]	SC1_RES_BASE_B[18]	SC1_RES_BASE_B[17]	SC1_RES_BASE_B[16]	SC1_RES_BASE_B[15]
		SC1_RES_BASE_B[15]	SC1_RES_BASE_B[14]	SC1_RES_BASE_B[13]	SC1_RES_BASE_B[12]	SC1_RES_BASE_B[11]	SC1_RES_BASE_B[10]	SC1_RES_BASE_B[9]	SC1_RES_BASE_B[8]	SC1_RES_BASE_B[7]
		SC1_RES_BASE_B[7]	SC1_RES_BASE_B[6]	SC1_RES_BASE_B[5]	SC1_RES_BASE_B[4]	SC1_RES_BASE_B[3]	SC1_RES_BASE_B[2]	SC1_RES_BASE_B[1]	SC1_RES_BASE_B[0]	SC1_RES_BASE_B[0]
	SC1_SCL1_WR9	-	SC1_RES_LN_OFF_B[14]	SC1_RES_LN_OFF_B[13]	SC1_RES_LN_OFF_B[12]	SC1_RES_LN_OFF_B[11]	SC1_RES_LN_OFF_B[10]	SC1_RES_LN_OFF_B[9]	SC1_RES_LN_OFF_B[8]	SC1_RES_LN_OFF_B[7]
		SC1_RES_LN_OFF_B[7]	SC1_RES_LN_OFF_B[6]	SC1_RES_LN_OFF_B[5]	SC1_RES_LN_OFF_B[4]	SC1_RES_LN_OFF_B[3]	SC1_RES_LN_OFF_B[2]	SC1_RES_LN_OFF_B[1]	SC1_RES_LN_OFF_B[0]	SC1_RES_LN_OFF_B[0]
		-	-	-	-	-	-	SC1_RES_FLM_NUM_B[9]	SC1_RES_FLM_NUM_B[8]	SC1_RES_FLM_NUM_B[7]
		SC1_RES_FLM_NUM_B[7]	SC1_RES_FLM_NUM_B[6]	SC1_RES_FLM_NUM_B[5]	SC1_RES_FLM_NUM_B[4]	SC1_RES_FLM_NUM_B[3]	SC1_RES_FLM_NUM_B[2]	SC1_RES_FLM_NUM_B[1]	SC1_RES_FLM_NUM_B[0]	SC1_RES_FLM_NUM_B[0]
	SC1_SCL1_WR10	-	-	-	-	-	-	-	-	-
		-	SC1_RES_FLM_OFF_B[22]	SC1_RES_FLM_OFF_B[21]	SC1_RES_FLM_OFF_B[20]	SC1_RES_FLM_OFF_B[19]	SC1_RES_FLM_OFF_B[18]	SC1_RES_FLM_OFF_B[17]	SC1_RES_FLM_OFF_B[16]	SC1_RES_FLM_OFF_B[15]
		SC1_RES_FLM_OFF_B[15]	SC1_RES_FLM_OFF_B[14]	SC1_RES_FLM_OFF_B[13]	SC1_RES_FLM_OFF_B[12]	SC1_RES_FLM_OFF_B[11]	SC1_RES_FLM_OFF_B[10]	SC1_RES_FLM_OFF_B[9]	SC1_RES_FLM_OFF_B[8]	SC1_RES_FLM_OFF_B[7]
		SC1_RES_FLM_OFF_B[7]	SC1_RES_FLM_OFF_B[6]	SC1_RES_FLM_OFF_B[5]	SC1_RES_FLM_OFF_B[4]	SC1_RES_FLM_OFF_B[3]	SC1_RES_FLM_OFF_B[2]	SC1_RES_FLM_OFF_B[1]	SC1_RES_FLM_OFF_B[0]	SC1_RES_FLM_OFF_B[0]
	SC1_SCL1_WR11	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	SC1_RES_FLM_CNT_B[9]	SC1_RES_FLM_CNT_B[8]	SC1_RES_FLM_CNT_B[7]
		SC1_RES_FLM_CNT_B[7]	SC1_RES_FLM_CNT_B[6]	SC1_RES_FLM_CNT_B[5]	SC1_RES_FLM_CNT_B[4]	SC1_RES_FLM_CNT_B[3]	SC1_RES_FLM_CNT_B[2]	SC1_RES_FLM_CNT_B[1]	SC1_RES_FLM_CNT_B[0]	SC1_RES_FLM_CNT_B[0]
	SC1_SCL1_MON1	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	SC1_PBUF_NUM[1]	SC1_PBUF_NUM[0]	SC1_PBUF_NUM[0]
		-	-	-	-	-	-	-	-	-
	SC1_SCL1_PBUF0	SC1_PBUF0_ADD[31]	SC1_PBUF0_ADD[30]	SC1_PBUF0_ADD[29]	SC1_PBUF0_ADD[28]	SC1_PBUF0_ADD[27]	SC1_PBUF0_ADD[26]	SC1_PBUF0_ADD[25]	SC1_PBUF0_ADD[24]	SC1_PBUF0_ADD[23]
		SC1_PBUF0_ADD[23]	SC1_PBUF0_ADD[22]	SC1_PBUF0_ADD[21]	SC1_PBUF0_ADD[20]	SC1_PBUF0_ADD[19]	SC1_PBUF0_ADD[18]	SC1_PBUF0_ADD[17]	SC1_PBUF0_ADD[16]	SC1_PBUF0_ADD[15]
		SC1_PBUF0_ADD[15]	SC1_PBUF0_ADD[14]	SC1_PBUF0_ADD[13]	SC1_PBUF0_ADD[12]	SC1_PBUF0_ADD[11]	SC1_PBUF0_ADD[10]	SC1_PBUF0_ADD[9]	SC1_PBUF0_ADD[8]	SC1_PBUF0_ADD[7]
		SC1_PBUF0_ADD[7]	SC1_PBUF0_ADD[6]	SC1_PBUF0_ADD[5]	SC1_PBUF0_ADD[4]	SC1_PBUF0_ADD[3]	SC1_PBUF0_ADD[2]	SC1_PBUF0_ADD[1]	SC1_PBUF0_ADD[0]	SC1_PBUF0_ADD[0]
	SC1_SCL1_PBUF1	SC1_PBUF1_ADD[31]	SC1_PBUF1_ADD[30]	SC1_PBUF1_ADD[29]	SC1_PBUF1_ADD[28]	SC1_PBUF1_ADD[27]	SC1_PBUF1_ADD[26]	SC1_PBUF1_ADD[25]	SC1_PBUF1_ADD[24]	SC1_PBUF1_ADD[23]
		SC1_PBUF1_ADD[23]	SC1_PBUF1_ADD[22]	SC1_PBUF1_ADD[21]	SC1_PBUF1_ADD[20]	SC1_PBUF1_ADD[19]	SC1_PBUF1_ADD[18]	SC1_PBUF1_ADD[17]	SC1_PBUF1_ADD[16]	SC1_PBUF1_ADD[15]
		SC1_PBUF1_ADD[15]	SC1_PBUF1_ADD[14]	SC1_PBUF1_ADD[13]	SC1_PBUF1_ADD[12]	SC1_PBUF1_ADD[11]	SC1_PBUF1_ADD[10]	SC1_PBUF1_ADD[9]	SC1_PBUF1_ADD[8]	SC1_PBUF1_ADD[7]
		SC1_PBUF1_ADD[7]	SC1_PBUF1_ADD[6]	SC1_PBUF1_ADD[5]	SC1_PBUF1_ADD[4]	SC1_PBUF1_ADD[3]	SC1_PBUF1_ADD[2]	SC1_PBUF1_ADD[1]	SC1_PBUF1_ADD[0]	SC1_PBUF1_ADD[0]
	SC1_SCL1_PBUF2	SC1_PBUF2_ADD[31]	SC1_PBUF2_ADD[30]	SC1_PBUF2_ADD[29]	SC1_PBUF2_ADD[28]	SC1_PBUF2_ADD[27]	SC1_PBUF2_ADD[26]	SC1_PBUF2_ADD[25]	SC1_PBUF2_ADD[24]	SC1_PBUF2_ADD[23]
		SC1_PBUF2_ADD[23]	SC1_PBUF2_ADD[22]	SC1_PBUF2_ADD[21]	SC1_PBUF2_ADD[20]	SC1_PBUF2_ADD[19]	SC1_PBUF2_ADD[18]	SC1_PBUF2_ADD[17]	SC1_PBUF2_ADD[16]	SC1_PBUF2_ADD[15]
		SC1_PBUF2_ADD[15]	SC1_PBUF2_ADD[14]	SC1_PBUF2_ADD[13]	SC1_PBUF2_ADD[12]	SC1_PBUF2_ADD[11]	SC1_PBUF2_ADD[10]	SC1_PBUF2_ADD[9]	SC1_PBUF2_ADD[8]	SC1_PBUF2_ADD[7]
		SC1_PBUF2_ADD[7]	SC1_PBUF2_ADD[6]	SC1_PBUF2_ADD[5]	SC1_PBUF2_ADD[4]	SC1_PBUF2_ADD[3]	SC1_PBUF2_ADD[2]	SC1_PBUF2_ADD[1]	SC1_PBUF2_ADD[0]	SC1_PBUF2_ADD[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 1	SC1_SCL1_PBUF3	SC1_PBUF3_ADD[31]	SC1_PBUF3_ADD[30]	SC1_PBUF3_ADD[29]	SC1_PBUF3_ADD[28]	SC1_PBUF3_ADD[27]	SC1_PBUF3_ADD[26]	SC1_PBUF3_ADD[25]	SC1_PBUF3_ADD[24]	
		SC1_PBUF3_ADD[23]	SC1_PBUF3_ADD[22]	SC1_PBUF3_ADD[21]	SC1_PBUF3_ADD[20]	SC1_PBUF3_ADD[19]	SC1_PBUF3_ADD[18]	SC1_PBUF3_ADD[17]	SC1_PBUF3_ADD[16]	
		SC1_PBUF3_ADD[15]	SC1_PBUF3_ADD[14]	SC1_PBUF3_ADD[13]	SC1_PBUF3_ADD[12]	SC1_PBUF3_ADD[11]	SC1_PBUF3_ADD[10]	SC1_PBUF3_ADD[9]	SC1_PBUF3_ADD[8]	
		SC1_PBUF3_ADD[7]	SC1_PBUF3_ADD[6]	SC1_PBUF3_ADD[5]	SC1_PBUF3_ADD[4]	SC1_PBUF3_ADD[3]	SC1_PBUF3_ADD[2]	SC1_PBUF3_ADD[1]	SC1_PBUF3_ADD[0]	
	SC1_SCL1_PBUF_FLD	-	-	-	-	-	-	-	-	SC1_FLD_INF3
		-	-	-	-	-	-	-	-	SC1_FLD_INF2
		-	-	-	-	-	-	-	-	SC1_FLD_INF1
		-	-	-	-	-	-	-	-	SC1_FLD_INF0
	SC1_SCL1_PBUF_CNT	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	SC1_PBUF_RST
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
	GR1_UPDATE	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	GR1_UPDATE
		-	-	-	GR1_P_VEN	-	-	-	-	GR1_JBUS_VEN
	GR1_FLM_RD	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	GR1_R_ENB
	GR1_FLM1	GR1_FLD_SEL	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	GR1_LN_OFF_DIR
		-	-	-	-	-	-	GR1_FLM_SEL[1]	GR1_FLM_SEL[0]	-
		-	-	-	GR1_JMR_FLM_INV	-	-	-	-	GR1_BST_MD
	GR1_FLM2	GR1_BASE[31]	GR1_BASE[30]	GR1_BASE[29]	GR1_BASE[28]	GR1_BASE[27]	GR1_BASE[26]	GR1_BASE[25]	GR1_BASE[24]	-
		GR1_BASE[23]	GR1_BASE[22]	GR1_BASE[21]	GR1_BASE[20]	GR1_BASE[19]	GR1_BASE[18]	GR1_BASE[17]	GR1_BASE[16]	-
		GR1_BASE[15]	GR1_BASE[14]	GR1_BASE[13]	GR1_BASE[12]	GR1_BASE[11]	GR1_BASE[10]	GR1_BASE[9]	GR1_BASE[8]	-
		GR1_BASE[7]	GR1_BASE[6]	GR1_BASE[5]	GR1_BASE[4]	GR1_BASE[3]	GR1_BASE[2]	GR1_BASE[1]	GR1_BASE[0]	-
	GR1_FLM3	GR1_FLD_NXT	GR1_LN_OFF[14]	GR1_LN_OFF[13]	GR1_LN_OFF[12]	GR1_LN_OFF[11]	GR1_LN_OFF[10]	GR1_LN_OFF[9]	GR1_LN_OFF[8]	-
		GR1_LN_OFF[7]	GR1_LN_OFF[6]	GR1_LN_OFF[5]	GR1_LN_OFF[4]	GR1_LN_OFF[3]	GR1_LN_OFF[2]	GR1_LN_OFF[1]	GR1_LN_OFF[0]	-
		-	-	-	-	-	-	GR1_FLM_NUM[9]	GR1_FLM_NUM[8]	-
		GR1_FLM_NUM[7]	GR1_FLM_NUM[6]	GR1_FLM_NUM[5]	GR1_FLM_NUM[4]	GR1_FLM_NUM[3]	GR1_FLM_NUM[2]	GR1_FLM_NUM[1]	GR1_FLM_NUM[0]	-
	GR1_FLM4	-	-	-	-	-	-	-	-	-
		-	GR1_FLM_OFF[22]	GR1_FLM_OFF[21]	GR1_FLM_OFF[20]	GR1_FLM_OFF[19]	GR1_FLM_OFF[18]	GR1_FLM_OFF[17]	GR1_FLM_OFF[16]	-
		GR1_FLM_OFF[15]	GR1_FLM_OFF[14]	GR1_FLM_OFF[13]	GR1_FLM_OFF[12]	GR1_FLM_OFF[11]	GR1_FLM_OFF[10]	GR1_FLM_OFF[9]	GR1_FLM_OFF[8]	-
		GR1_FLM_OFF[7]	GR1_FLM_OFF[6]	GR1_FLM_OFF[5]	GR1_FLM_OFF[4]	GR1_FLM_OFF[3]	GR1_FLM_OFF[2]	GR1_FLM_OFF[1]	GR1_FLM_OFF[0]	-
	GR1_FLM5	-	-	-	-	-	GR1_FLM_LNUM[10]	GR1_FLM_LNUM[9]	GR1_FLM_LNUM[8]	-
		GR1_FLM_LNUM[7]	GR1_FLM_LNUM[6]	GR1_FLM_LNUM[5]	GR1_FLM_LNUM[4]	GR1_FLM_LNUM[3]	GR1_FLM_LNUM[2]	GR1_FLM_LNUM[1]	GR1_FLM_LNUM[0]	-
		-	-	-	-	-	GR1_FLM_LOOP[9]	GR1_FLM_LOOP[8]	GR1_FLM_LOOP[7]	-
		GR1_FLM_LOOP[7]	GR1_FLM_LOOP[6]	GR1_FLM_LOOP[5]	GR1_FLM_LOOP[4]	GR1_FLM_LOOP[3]	GR1_FLM_LOOP[2]	GR1_FLM_LOOP[1]	GR1_FLM_LOOP[0]	-
	GR1_FLM6	GR1_FORMAT[3]	GR1_FORMAT[2]	GR1_FORMAT[1]	GR1_FORMAT[0]	-	GR1_HW[10]	GR1_HW[9]	GR1_HW[8]	-
		GR1_HW[7]	GR1_HW[6]	GR1_HW[5]	GR1_HW[4]	GR1_HW[3]	GR1_HW[2]	GR1_HW[1]	GR1_HW[0]	-
		GR1_YCC_SWAP[2]	GR1_YCC_SWAP[1]	GR1_YCC_SWAP[0]	GR1_RDSWA[2]	GR1_RDSWA[1]	GR1_RDSWA[0]	-	GR1_CNV444_MD	-
		-	-	GR1_STA_POS[5]	GR1_STA_POS[4]	GR1_STA_POS[3]	GR1_STA_POS[2]	GR1_STA_POS[1]	GR1_STA_POS[0]	-
	GR1_AB1	-	-	GR1_CUS_CON_ON	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		GR1_ARC_MUL	GR1_ACALC_MD	-	GR1_ARC_ON	-	-	-	-	GR1_ARC_DISP_ON
		-	-	-	GR1_GRC_DISP_ON	-	-	GR1_DISP_SEL[1]	GR1_DISP_SEL[0]	-

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Video display controller 5 channel 1	GR1_AB2	-	-	-	-	-	GR1_GRC_VS[10]	GR1_GRC_VS[9]	GR1_GRC_VS[8]
		GR1_GRC_VS[7]	GR1_GRC_VS[6]	GR1_GRC_VS[5]	GR1_GRC_VS[4]	GR1_GRC_VS[3]	GR1_GRC_VS[2]	GR1_GRC_VS[1]	GR1_GRC_VS[0]
		-	-	-	-	-	GR1_GRC_VW[10]	GR1_GRC_VW[9]	GR1_GRC_VW[8]
		GR1_GRC_VW[7]	GR1_GRC_VW[6]	GR1_GRC_VW[5]	GR1_GRC_VW[4]	GR1_GRC_VW[3]	GR1_GRC_VW[2]	GR1_GRC_VW[1]	GR1_GRC_VW[0]
	GR1_AB3	-	-	-	-	-	GR1_GRC_HS[10]	GR1_GRC_HS[9]	GR1_GRC_HS[8]
		GR1_GRC_HS[7]	GR1_GRC_HS[6]	GR1_GRC_HS[5]	GR1_GRC_HS[4]	GR1_GRC_HS[3]	GR1_GRC_HS[2]	GR1_GRC_HS[1]	GR1_GRC_HS[0]
		-	-	-	-	-	GR1_GRC_HW[10]	GR1_GRC_HW[9]	GR1_GRC_HW[8]
		GR1_GRC_HW[7]	GR1_GRC_HW[6]	GR1_GRC_HW[5]	GR1_GRC_HW[4]	GR1_GRC_HW[3]	GR1_GRC_HW[2]	GR1_GRC_HW[1]	GR1_GRC_HW[0]
	GR1_AB4	-	-	-	-	-	GR1_ARC_VS[10]	GR1_ARC_VS[9]	GR1_ARC_VS[8]
		GR1_ARC_VS[7]	GR1_ARC_VS[6]	GR1_ARC_VS[5]	GR1_ARC_VS[4]	GR1_ARC_VS[3]	GR1_ARC_VS[2]	GR1_ARC_VS[1]	GR1_ARC_VS[0]
		-	-	-	-	-	GR1_ARC_VW[10]	GR1_ARC_VW[9]	GR1_ARC_VW[8]
		GR1_ARC_VW[7]	GR1_ARC_VW[6]	GR1_ARC_VW[5]	GR1_ARC_VW[4]	GR1_ARC_VW[3]	GR1_ARC_VW[2]	GR1_ARC_VW[1]	GR1_ARC_VW[0]
	GR1_AB5	-	-	-	-	-	GR1_ARC_HS[10]	GR1_ARC_HS[9]	GR1_ARC_HS[8]
		GR1_ARC_HS[7]	GR1_ARC_HS[6]	GR1_ARC_HS[5]	GR1_ARC_HS[4]	GR1_ARC_HS[3]	GR1_ARC_HS[2]	GR1_ARC_HS[1]	GR1_ARC_HS[0]
		-	-	-	-	-	GR1_ARC_HW[10]	GR1_ARC_HW[9]	GR1_ARC_HW[8]
		GR1_ARC_HW[7]	GR1_ARC_HW[6]	GR1_ARC_HW[5]	GR1_ARC_HW[4]	GR1_ARC_HW[3]	GR1_ARC_HW[2]	GR1_ARC_HW[1]	GR1_ARC_HW[0]
	GR1_AB6	-	-	-	-	-	-	-	GR1_ARC_MODE
		GR1_ARC_COEF[7]	GR1_ARC_COEF[6]	GR1_ARC_COEF[5]	GR1_ARC_COEF[4]	GR1_ARC_COEF[3]	GR1_ARC_COEF[2]	GR1_ARC_COEF[1]	GR1_ARC_COEF[0]
		-	-	-	-	-	-	-	-
		GR1_ARC_RATE[7]	GR1_ARC_RATE[6]	GR1_ARC_RATE[5]	GR1_ARC_RATE[4]	GR1_ARC_RATE[3]	GR1_ARC_RATE[2]	GR1_ARC_RATE[1]	GR1_ARC_RATE[0]
	GR1_AB7	-	-	-	-	-	-	-	-
		GR1_ARC_DEF[7]	GR1_ARC_DEF[6]	GR1_ARC_DEF[5]	GR1_ARC_DEF[4]	GR1_ARC_DEF[3]	GR1_ARC_DEF[2]	GR1_ARC_DEF[1]	GR1_ARC_DEF[0]
		-	-	-	-	-	-	-	GR1_CK_ON
	GR1_AB8	GR1_CK_KCLUT[7]	GR1_CK_KCLUT[6]	GR1_CK_KCLUT[5]	GR1_CK_KCLUT[4]	GR1_CK_KCLUT[3]	GR1_CK_KCLUT[2]	GR1_CK_KCLUT[1]	GR1_CK_KCLUT[0]
		GR1_CK_KG[7]	GR1_CK_KG[6]	GR1_CK_KG[5]	GR1_CK_KG[4]	GR1_CK_KG[3]	GR1_CK_KG[2]	GR1_CK_KG[1]	GR1_CK_KG[0]
		GR1_CK_KB[7]	GR1_CK_KB[6]	GR1_CK_KB[5]	GR1_CK_KB[4]	GR1_CK_KB[3]	GR1_CK_KB[2]	GR1_CK_KB[1]	GR1_CK_KB[0]
		GR1_CK_KR[7]	GR1_CK_KR[6]	GR1_CK_KR[5]	GR1_CK_KR[4]	GR1_CK_KR[3]	GR1_CK_KR[2]	GR1_CK_KR[1]	GR1_CK_KR[0]
	GR1_AB9	GR1_CK_A[7]	GR1_CK_A[6]	GR1_CK_A[5]	GR1_CK_A[4]	GR1_CK_A[3]	GR1_CK_A[2]	GR1_CK_A[1]	GR1_CK_A[0]
		GR1_CK_G[7]	GR1_CK_G[6]	GR1_CK_G[5]	GR1_CK_G[4]	GR1_CK_G[3]	GR1_CK_G[2]	GR1_CK_G[1]	GR1_CK_G[0]
		GR1_CK_B[7]	GR1_CK_B[6]	GR1_CK_B[5]	GR1_CK_B[4]	GR1_CK_B[3]	GR1_CK_B[2]	GR1_CK_B[1]	GR1_CK_B[0]
		GR1_CK_R[7]	GR1_CK_R[6]	GR1_CK_R[5]	GR1_CK_R[4]	GR1_CK_R[3]	GR1_CK_R[2]	GR1_CK_R[1]	GR1_CK_R[0]
	GR1_AB10	GR1_A0[7]	GR1_A0[6]	GR1_A0[5]	GR1_A0[4]	GR1_A0[3]	GR1_A0[2]	GR1_A0[1]	GR1_A0[0]
		GR1_G0[7]	GR1_G0[6]	GR1_G0[5]	GR1_G0[4]	GR1_G0[3]	GR1_G0[2]	GR1_G0[1]	GR1_G0[0]
		GR1_B0[7]	GR1_B0[6]	GR1_B0[5]	GR1_B0[4]	GR1_B0[3]	GR1_B0[2]	GR1_B0[1]	GR1_B0[0]
		GR1_R0[7]	GR1_R0[6]	GR1_R0[5]	GR1_R0[4]	GR1_R0[3]	GR1_R0[2]	GR1_R0[1]	GR1_R0[0]
	GR1_AB11	GR1_A1[7]	GR1_A1[6]	GR1_A1[5]	GR1_A1[4]	GR1_A1[3]	GR1_A1[2]	GR1_A1[1]	GR1_A1[0]
		GR1_G1[7]	GR1_G1[6]	GR1_G1[5]	GR1_G1[4]	GR1_G1[3]	GR1_G1[2]	GR1_G1[1]	GR1_G1[0]
		GR1_B1[7]	GR1_B1[6]	GR1_B1[5]	GR1_B1[4]	GR1_B1[3]	GR1_B1[2]	GR1_B1[1]	GR1_B1[0]
		GR1_R1[7]	GR1_R1[6]	GR1_R1[5]	GR1_R1[4]	GR1_R1[3]	GR1_R1[2]	GR1_R1[1]	GR1_R1[0]
	GR1_BASE	-	-	-	-	-	-	-	-
		GR1_BASE_G[7]	GR1_BASE_G[6]	GR1_BASE_G[5]	GR1_BASE_G[4]	GR1_BASE_G[3]	GR1_BASE_G[2]	GR1_BASE_G[1]	GR1_BASE_G[0]
		GR1_BASE_B[7]	GR1_BASE_B[6]	GR1_BASE_B[5]	GR1_BASE_B[4]	GR1_BASE_B[3]	GR1_BASE_B[2]	GR1_BASE_B[1]	GR1_BASE_B[0]
		GR1_BASE_R[7]	GR1_BASE_R[6]	GR1_BASE_R[5]	GR1_BASE_R[4]	GR1_BASE_R[3]	GR1_BASE_R[2]	GR1_BASE_R[1]	GR1_BASE_R[0]
	GR1_CLUT	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GR1_CLT_SEL
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	GR1_MON	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GR1_ARC_ST

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 1	ADJO_UPDATE	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	ADJO_VEN	
	ADJO_BKSTR_SET	-	-	-	-	-	-	-	-	BKSTR_ON
		BKSTR_ST[3]	BKSTR_ST[2]	BKSTR_ST[1]	BKSTR_ST[0]	BKSTR_D[3]	BKSTR_D[2]	BKSTR_D[1]	BKSTR_D[0]	BKSTR_D[0]
		-	-	-	BKSTR_T1[4]	BKSTR_T1[3]	BKSTR_T1[2]	BKSTR_T1[1]	BKSTR_T1[0]	BKSTR_T1[0]
		-	-	-	BKSTR_T2[4]	BKSTR_T2[3]	BKSTR_T2[2]	BKSTR_T2[1]	BKSTR_T2[0]	BKSTR_T2[0]
	ADJO_ENH_TIM1	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	ENH_MD	-	-	-	-	ENH_DISP_ON
	ADJO_ENH_TIM2	-	-	-	-	-	-	ENH_VS[10]	ENH_VS[9]	ENH_VS[8]
		ENH_VS[7]	ENH_VS[6]	ENH_VS[5]	ENH_VS[4]	ENH_VS[3]	ENH_VS[2]	ENH_VS[1]	ENH_VS[0]	ENH_VS[0]
		-	-	-	-	-	-	ENH_VW[10]	ENH_VW[9]	ENH_VW[8]
		ENH_VW[7]	ENH_VW[6]	ENH_VW[5]	ENH_VW[4]	ENH_VW[3]	ENH_VW[2]	ENH_VW[1]	ENH_VW[0]	ENH_VW[0]
	ADJO_ENH_TIM3	-	-	-	-	-	-	ENH_HS[10]	ENH_HS[9]	ENH_HS[8]
		ENH_HS[7]	ENH_HS[6]	ENH_HS[5]	ENH_HS[4]	ENH_HS[3]	ENH_HS[2]	ENH_HS[1]	ENH_HS[0]	ENH_HS[0]
		-	-	-	-	-	-	ENH_HW[10]	ENH_HW[9]	ENH_HW[8]
		ENH_HW[7]	ENH_HW[6]	ENH_HW[5]	ENH_HW[4]	ENH_HW[3]	ENH_HW[2]	ENH_HW[1]	ENH_HW[0]	ENH_HW[0]
	ADJO_ENH_SHP1	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	SHP_H_ON
		-	-	-	-	-	-	-	-	-
		-	SHP_H1_CORE[6]	SHP_H1_CORE[5]	SHP_H1_CORE[4]	SHP_H1_CORE[3]	SHP_H1_CORE[2]	SHP_H1_CORE[1]	SHP_H1_CORE[0]	SHP_H1_CORE[0]
	ADJO_ENH_SHP2	SHP_H1_CLIP_O[7]	SHP_H1_CLIP_O[6]	SHP_H1_CLIP_O[5]	SHP_H1_CLIP_O[4]	SHP_H1_CLIP_O[3]	SHP_H1_CLIP_O[2]	SHP_H1_CLIP_O[1]	SHP_H1_CLIP_O[0]	SHP_H1_CLIP_O[0]
		SHP_H1_CLIP_U [7]	SHP_H1_CLIP_U [6]	SHP_H1_CLIP_U [5]	SHP_H1_CLIP_U [4]	SHP_H1_CLIP_U [3]	SHP_H1_CLIP_U [2]	SHP_H1_CLIP_U [1]	SHP_H1_CLIP_U [0]	SHP_H1_CLIP_U [0]
		SHP_H1_GAIN_O[7]	SHP_H1_GAIN_O[6]	SHP_H1_GAIN_O[5]	SHP_H1_GAIN_O[4]	SHP_H1_GAIN_O[3]	SHP_H1_GAIN_O[2]	SHP_H1_GAIN_O[1]	SHP_H1_GAIN_O[0]	SHP_H1_GAIN_O[0]
		SHP_H1_GAIN_U[7]	SHP_H1_GAIN_U[6]	SHP_H1_GAIN_U[5]	SHP_H1_GAIN_U[4]	SHP_H1_GAIN_U[3]	SHP_H1_GAIN_U[2]	SHP_H1_GAIN_U[1]	SHP_H1_GAIN_U[0]	SHP_H1_GAIN_U[0]
	ADJO_ENH_SHP3	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	SHP_H2_LPF_SEL
		-	-	-	-	-	-	-	-	-
		-	SHP_H2_CORE[6]	SHP_H2_CORE[5]	SHP_H2_CORE[4]	SHP_H2_CORE[3]	SHP_H2_CORE[2]	SHP_H2_CORE[1]	SHP_H2_CORE[0]	SHP_H2_CORE[0]
	ADJO_ENH_SHP4	SHP_H2_CLIP_O[7]	SHP_H2_CLIP_O[6]	SHP_H2_CLIP_O[5]	SHP_H2_CLIP_O[4]	SHP_H2_CLIP_O[3]	SHP_H2_CLIP_O[2]	SHP_H2_CLIP_O[1]	SHP_H2_CLIP_O[0]	SHP_H2_CLIP_O[0]
		SHP_H2_CLIP_U [7]	SHP_H2_CLIP_U [6]	SHP_H2_CLIP_U [5]	SHP_H2_CLIP_U [4]	SHP_H2_CLIP_U [3]	SHP_H2_CLIP_U [2]	SHP_H2_CLIP_U [1]	SHP_H2_CLIP_U [0]	SHP_H2_CLIP_U [0]
		SHP_H2_GAIN_O[7]	SHP_H2_GAIN_O[6]	SHP_H2_GAIN_O[5]	SHP_H2_GAIN_O[4]	SHP_H2_GAIN_O[3]	SHP_H2_GAIN_O[2]	SHP_H2_GAIN_O[1]	SHP_H2_GAIN_O[0]	SHP_H2_GAIN_O[0]
		SHP_H2_GAIN_U[7]	SHP_H2_GAIN_U[6]	SHP_H2_GAIN_U[5]	SHP_H2_GAIN_U[4]	SHP_H2_GAIN_U[3]	SHP_H2_GAIN_U[2]	SHP_H2_GAIN_U[1]	SHP_H2_GAIN_U[0]	SHP_H2_GAIN_U[0]
	ADJO_ENH_SHP5	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	SHP_H3_CORE[6]	SHP_H3_CORE[5]	SHP_H3_CORE[4]	SHP_H3_CORE[3]	SHP_H3_CORE[2]	SHP_H3_CORE[1]	SHP_H3_CORE[0]	SHP_H3_CORE[0]
	ADJO_ENH_SHP6	SHP_H3_CLIP_O[7]	SHP_H3_CLIP_O[6]	SHP_H3_CLIP_O[5]	SHP_H3_CLIP_O[4]	SHP_H3_CLIP_O[3]	SHP_H3_CLIP_O[2]	SHP_H3_CLIP_O[1]	SHP_H3_CLIP_O[0]	SHP_H3_CLIP_O[0]
		SHP_H3_CLIP_U [7]	SHP_H3_CLIP_U [6]	SHP_H3_CLIP_U [5]	SHP_H3_CLIP_U [4]	SHP_H3_CLIP_U [3]	SHP_H3_CLIP_U [2]	SHP_H3_CLIP_U [1]	SHP_H3_CLIP_U [0]	SHP_H3_CLIP_U [0]
		SHP_H3_GAIN_O[7]	SHP_H3_GAIN_O[6]	SHP_H3_GAIN_O[5]	SHP_H3_GAIN_O[4]	SHP_H3_GAIN_O[3]	SHP_H3_GAIN_O[2]	SHP_H3_GAIN_O[1]	SHP_H3_GAIN_O[0]	SHP_H3_GAIN_O[0]
		SHP_H3_GAIN_U[7]	SHP_H3_GAIN_U[6]	SHP_H3_GAIN_U[5]	SHP_H3_GAIN_U[4]	SHP_H3_GAIN_U[3]	SHP_H3_GAIN_U[2]	SHP_H3_GAIN_U[1]	SHP_H3_GAIN_U[0]	SHP_H3_GAIN_U[0]
	ADJO_ENH_LTI1	LTI_H_ON	-	-	-	-	-	-	-	LTI_H2_LPF_SEL
		LTI_H2_INC_ZERO[7]	LTI_H2_INC_ZERO[6]	LTI_H2_INC_ZERO[5]	LTI_H2_INC_ZERO[4]	LTI_H2_INC_ZERO[3]	LTI_H2_INC_ZERO[2]	LTI_H2_INC_ZERO[1]	LTI_H2_INC_ZERO[0]	LTI_H2_INC_ZERO[0]
		LTI_H2_GAIN[7]	LTI_H2_GAIN[6]	LTI_H2_GAIN[5]	LTI_H2_GAIN[4]	LTI_H2_GAIN[3]	LTI_H2_GAIN[2]	LTI_H2_GAIN[1]	LTI_H2_GAIN[0]	LTI_H2_GAIN[0]
		LTI_H2_CORE[7]	LTI_H2_CORE[6]	LTI_H2_CORE[5]	LTI_H2_CORE[4]	LTI_H2_CORE[3]	LTI_H2_CORE[2]	LTI_H2_CORE[1]	LTI_H2_CORE[0]	LTI_H2_CORE[0]



Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 1	ADJ0_ENH_LTI2	-	-	-	-	-	-	-	LTI_H4_MEDIAN_TAP_SEL	
		LTI_H4_INC_ZERO[7]	LTI_H4_INC_ZERO[6]	LTI_H4_INC_ZERO[5]	LTI_H4_INC_ZERO[4]	LTI_H4_INC_ZERO[3]	LTI_H4_INC_ZERO[2]	LTI_H4_INC_ZERO[1]	LTI_H4_INC_ZERO[0]	
		LTI_H4_GAIN[7]	LTI_H4_GAIN[6]	LTI_H4_GAIN[5]	LTI_H4_GAIN[4]	LTI_H4_GAIN[3]	LTI_H4_GAIN[2]	LTI_H4_GAIN[1]	LTI_H4_GAIN[0]	
		LTI_H4_CORE[7]	LTI_H4_CORE[6]	LTI_H4_CORE[5]	LTI_H4_CORE[4]	LTI_H4_CORE[3]	LTI_H4_CORE[2]	LTI_H4_CORE[1]	LTI_H4_CORE[0]	
	ADJ0_MTX_MODE	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	ADJ0_MTX_MD[1]	-	ADJ0_MTX_MD[0]
	ADJ0_MTX_YG_ADJ0	-	-	-	-	-	-	-	-	-
		ADJ0_MTX_YG[7]	ADJ0_MTX_YG[6]	ADJ0_MTX_YG[5]	ADJ0_MTX_YG[4]	ADJ0_MTX_YG[3]	ADJ0_MTX_YG[2]	ADJ0_MTX_YG[1]	ADJ0_MTX_YG[0]	-
		-	-	-	-	-	ADJ0_MTX_GG[10]	ADJ0_MTX_GG[9]	-	ADJ0_MTX_GG[8]
	ADJ0_MTX_YG_ADJ1	ADJ0_MTX_GG[7]	ADJ0_MTX_GG[6]	ADJ0_MTX_GG[5]	ADJ0_MTX_GG[4]	ADJ0_MTX_GG[3]	ADJ0_MTX_GG[2]	ADJ0_MTX_GG[1]	-	ADJ0_MTX_GG[0]
		-	-	-	-	-	ADJ0_MTX_GB[10]	ADJ0_MTX_GB[9]	-	ADJ0_MTX_GB[8]
		ADJ0_MTX_GB[7]	ADJ0_MTX_GB[6]	ADJ0_MTX_GB[5]	ADJ0_MTX_GB[4]	ADJ0_MTX_GB[3]	ADJ0_MTX_GB[2]	ADJ0_MTX_GB[1]	-	ADJ0_MTX_GB[0]
	ADJ0_MTX_CBB_ADJ0	-	-	-	-	-	-	ADJ0_MTX_GR[10]	ADJ0_MTX_GR[9]	ADJ0_MTX_GR[8]
		ADJ0_MTX_GR[7]	ADJ0_MTX_GR[6]	ADJ0_MTX_GR[5]	ADJ0_MTX_GR[4]	ADJ0_MTX_GR[3]	ADJ0_MTX_GR[2]	ADJ0_MTX_GR[1]	-	ADJ0_MTX_GR[0]
		-	-	-	-	-	-	-	-	-
	ADJ0_MTX_CBB_ADJ1	-	-	-	-	-	-	ADJ0_MTX_BB[10]	ADJ0_MTX_BB[9]	ADJ0_MTX_BB[8]
		ADJ0_MTX_BB[7]	ADJ0_MTX_BB[6]	ADJ0_MTX_BB[5]	ADJ0_MTX_BB[4]	ADJ0_MTX_BB[3]	ADJ0_MTX_BB[2]	ADJ0_MTX_BB[1]	-	ADJ0_MTX_BB[0]
		-	-	-	-	-	ADJ0_MTX_BR[10]	ADJ0_MTX_BR[9]	-	ADJ0_MTX_BR[8]
	ADJ0_MTX_CRR_ADJ0	ADJ0_MTX_BR[7]	ADJ0_MTX_BR[6]	ADJ0_MTX_BR[5]	ADJ0_MTX_BR[4]	ADJ0_MTX_BR[3]	ADJ0_MTX_BR[2]	ADJ0_MTX_BR[1]	-	ADJ0_MTX_BR[0]
		-	-	-	-	-	-	-	-	-
		ADJ0_MTX_BG[7]	ADJ0_MTX_BG[6]	ADJ0_MTX_BG[5]	ADJ0_MTX_BG[4]	ADJ0_MTX_BG[3]	ADJ0_MTX_BG[2]	ADJ0_MTX_BG[1]	-	ADJ0_MTX_BG[0]
	ADJ0_MTX_CRR_ADJ1	-	-	-	-	-	-	ADJ0_MTX_RG[10]	ADJ0_MTX_RG[9]	ADJ0_MTX_RG[8]
		ADJ0_MTX_RR[7]	ADJ0_MTX_RR[6]	ADJ0_MTX_RR[5]	ADJ0_MTX_RR[4]	ADJ0_MTX_RR[3]	ADJ0_MTX_RR[2]	ADJ0_MTX_RR[1]	-	ADJ0_MTX_RR[0]
		-	-	-	-	-	-	-	-	-
	ADJ1_UPDATE	ADJ0_MTX_RG[7]	ADJ0_MTX_RG[6]	ADJ0_MTX_RG[5]	ADJ0_MTX_RG[4]	ADJ0_MTX_RG[3]	ADJ0_MTX_RG[2]	ADJ0_MTX_RG[1]	-	ADJ0_MTX_RG[0]
		-	-	-	-	-	ADJ0_MTX_RB[10]	ADJ0_MTX_RB[9]	-	ADJ0_MTX_RB[8]
		ADJ0_MTX_RB[7]	ADJ0_MTX_RB[6]	ADJ0_MTX_RB[5]	ADJ0_MTX_RB[4]	ADJ0_MTX_RB[3]	ADJ0_MTX_RB[2]	ADJ0_MTX_RB[1]	-	ADJ0_MTX_RB[0]
	ADJ1_BKSTR_SET	-	-	-	-	-	-	ADJ0_MTX_RR[10]	ADJ0_MTX_RR[9]	ADJ0_MTX_RR[8]
		BKSTR_ST[3]	BKSTR_ST[2]	BKSTR_ST[1]	BKSTR_ST[0]	BKSTR_D[3]	BKSTR_D[2]	BKSTR_D[1]	-	BKSTR_D[0]
		-	-	-	BKSTR_T1[4]	BKSTR_T1[3]	BKSTR_T1[2]	BKSTR_T1[1]	-	BKSTR_T1[0]
	ADJ1_ENH_TIM1	-	-	-	BKSTR_T2[4]	BKSTR_T2[3]	BKSTR_T2[2]	BKSTR_T2[1]	-	BKSTR_T2[0]
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
	ADJ1_ENH_TIM2	-	-	-	ENH_MD	-	-	-	-	ENH_DISP_ON
		ENH_VS[7]	ENH_VS[6]	ENH_VS[5]	ENH_VS[4]	ENH_VS[3]	ENH_VS[2]	ENH_VS[1]	-	ENH_VS[0]
		-	-	-	-	-	ENH_VW[10]	ENH_VW[9]	-	ENH_VW[8]
	ADJ1_ENH_TIM3	ENH_VW[7]	ENH_VW[6]	ENH_VW[5]	ENH_VW[4]	ENH_VW[3]	ENH_VW[2]	ENH_VW[1]	-	ENH_VW[0]
		-	-	-	-	-	ENH_HS[10]	ENH_HS[9]	-	ENH_HS[8]
		ENH_HS[7]	ENH_HS[6]	ENH_HS[5]	ENH_HS[4]	ENH_HS[3]	ENH_HS[2]	ENH_HS[1]	-	ENH_HS[0]
	ADJ1_ENH_TIM3	-	-	-	-	-	ENH_HW[10]	ENH_HW[9]	-	ENH_HW[8]
		-	-	-	-	-	-	-	-	-
		ENH_HW[7]	ENH_HW[6]	ENH_HW[5]	ENH_HW[4]	ENH_HW[3]	ENH_HW[2]	ENH_HW[1]	-	ENH_HW[0]

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Video display controller 5 channel 1	ADJ1_ENH_SHP1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	SHP_H_ON
		-	-	-	-	-	-	-	-
		-	SHP_H1_CORE[6]	SHP_H1_CORE[5]	SHP_H1_CORE[4]	SHP_H1_CORE[3]	SHP_H1_CORE[2]	SHP_H1_CORE[1]	SHP_H1_CORE[0]
	ADJ1_ENH_SHP2	SHP_H1_CLIP_O[7]	SHP_H1_CLIP_O[6]	SHP_H1_CLIP_O[5]	SHP_H1_CLIP_O[4]	SHP_H1_CLIP_O[3]	SHP_H1_CLIP_O[2]	SHP_H1_CLIP_O[1]	SHP_H1_CLIP_O[0]
		SHP_H1_CLIP_U [7]	SHP_H1_CLIP_U [6]	SHP_H1_CLIP_U [5]	SHP_H1_CLIP_U [4]	SHP_H1_CLIP_U [3]	SHP_H1_CLIP_U [2]	SHP_H1_CLIP_U [1]	SHP_H1_CLIP_U [0]
		SHP_H1_GAIN_O[7]	SHP_H1_GAIN_O[6]	SHP_H1_GAIN_O[5]	SHP_H1_GAIN_O[4]	SHP_H1_GAIN_O[3]	SHP_H1_GAIN_O[2]	SHP_H1_GAIN_O[1]	SHP_H1_GAIN_O[0]
		SHP_H1_GAIN_U[7]	SHP_H1_GAIN_U[6]	SHP_H1_GAIN_U[5]	SHP_H1_GAIN_U[4]	SHP_H1_GAIN_U[3]	SHP_H1_GAIN_U[2]	SHP_H1_GAIN_U[1]	SHP_H1_GAIN_U[0]
	ADJ1_ENH_SHP3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	SHP_H2_LPF_SEL
		-	-	-	-	-	-	-	-
		-	SHP_H2_CORE[6]	SHP_H2_CORE[5]	SHP_H2_CORE[4]	SHP_H2_CORE[3]	SHP_H2_CORE[2]	SHP_H2_CORE[1]	SHP_H2_CORE[0]
	ADJ1_ENH_SHP4	SHP_H2_CLIP_O[7]	SHP_H2_CLIP_O[6]	SHP_H2_CLIP_O[5]	SHP_H2_CLIP_O[4]	SHP_H2_CLIP_O[3]	SHP_H2_CLIP_O[2]	SHP_H2_CLIP_O[1]	SHP_H2_CLIP_O[0]
		SHP_H2_CLIP_U [7]	SHP_H2_CLIP_U [6]	SHP_H2_CLIP_U [5]	SHP_H2_CLIP_U [4]	SHP_H2_CLIP_U [3]	SHP_H2_CLIP_U [2]	SHP_H2_CLIP_U [1]	SHP_H2_CLIP_U [0]
		SHP_H2_GAIN_O[7]	SHP_H2_GAIN_O[6]	SHP_H2_GAIN_O[5]	SHP_H2_GAIN_O[4]	SHP_H2_GAIN_O[3]	SHP_H2_GAIN_O[2]	SHP_H2_GAIN_O[1]	SHP_H2_GAIN_O[0]
		SHP_H2_GAIN_U[7]	SHP_H2_GAIN_U[6]	SHP_H2_GAIN_U[5]	SHP_H2_GAIN_U[4]	SHP_H2_GAIN_U[3]	SHP_H2_GAIN_U[2]	SHP_H2_GAIN_U[1]	SHP_H2_GAIN_U[0]
	ADJ1_ENH_SHP5	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	SHP_H3_CORE[6]	SHP_H3_CORE[5]	SHP_H3_CORE[4]	SHP_H3_CORE[3]	SHP_H3_CORE[2]	SHP_H3_CORE[1]	SHP_H3_CORE[0]
	ADJ1_ENH_SHP6	SHP_H3_CLIP_O[7]	SHP_H3_CLIP_O[6]	SHP_H3_CLIP_O[5]	SHP_H3_CLIP_O[4]	SHP_H3_CLIP_O[3]	SHP_H3_CLIP_O[2]	SHP_H3_CLIP_O[1]	SHP_H3_CLIP_O[0]
		SHP_H3_CLIP_U [7]	SHP_H3_CLIP_U [6]	SHP_H3_CLIP_U [5]	SHP_H3_CLIP_U [4]	SHP_H3_CLIP_U [3]	SHP_H3_CLIP_U [2]	SHP_H3_CLIP_U [1]	SHP_H3_CLIP_U [0]
		SHP_H3_GAIN_O[7]	SHP_H3_GAIN_O[6]	SHP_H3_GAIN_O[5]	SHP_H3_GAIN_O[4]	SHP_H3_GAIN_O[3]	SHP_H3_GAIN_O[2]	SHP_H3_GAIN_O[1]	SHP_H3_GAIN_O[0]
		SHP_H3_GAIN_U[7]	SHP_H3_GAIN_U[6]	SHP_H3_GAIN_U[5]	SHP_H3_GAIN_U[4]	SHP_H3_GAIN_U[3]	SHP_H3_GAIN_U[2]	SHP_H3_GAIN_U[1]	SHP_H3_GAIN_U[0]
	ADJ1_ENH_LTI1	LTI_H_ON	-	-	-	-	-	-	LTI_H2_LPF_SEL
		LTI_H2_INC_ZERO[7]	LTI_H2_INC_ZERO[6]	LTI_H2_INC_ZERO[5]	LTI_H2_INC_ZERO[4]	LTI_H2_INC_ZERO[3]	LTI_H2_INC_ZERO[2]	LTI_H2_INC_ZERO[1]	LTI_H2_INC_ZERO[0]
		LTI_H2_GAIN[7]	LTI_H2_GAIN[6]	LTI_H2_GAIN[5]	LTI_H2_GAIN[4]	LTI_H2_GAIN[3]	LTI_H2_GAIN[2]	LTI_H2_GAIN[1]	LTI_H2_GAIN[0]
		LTI_H2_CORE[7]	LTI_H2_CORE[6]	LTI_H2_CORE[5]	LTI_H2_CORE[4]	LTI_H2_CORE[3]	LTI_H2_CORE[2]	LTI_H2_CORE[1]	LTI_H2_CORE[0]
	ADJ1_ENH_LTI2	-	-	-	-	-	-	-	LTI_H4_MEDIAN_TAP_SEL
		LTI_H4_INC_ZERO[7]	LTI_H4_INC_ZERO[6]	LTI_H4_INC_ZERO[5]	LTI_H4_INC_ZERO[4]	LTI_H4_INC_ZERO[3]	LTI_H4_INC_ZERO[2]	LTI_H4_INC_ZERO[1]	LTI_H4_INC_ZERO[0]
		LTI_H4_GAIN[7]	LTI_H4_GAIN[6]	LTI_H4_GAIN[5]	LTI_H4_GAIN[4]	LTI_H4_GAIN[3]	LTI_H4_GAIN[2]	LTI_H4_GAIN[1]	LTI_H4_GAIN[0]
		LTI_H4_CORE[7]	LTI_H4_CORE[6]	LTI_H4_CORE[5]	LTI_H4_CORE[4]	LTI_H4_CORE[3]	LTI_H4_CORE[2]	LTI_H4_CORE[1]	LTI_H4_CORE[0]
	ADJ1_MTX_MODE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	ADJ1_MTX_MD[1]	ADJ1_MTX_MD[0]
	ADJ1_MTX_YG_ADJ0	-	-	-	-	-	-	-	-
		ADJ1_MTX_YG[7]	ADJ1_MTX_YG[6]	ADJ1_MTX_YG[5]	ADJ1_MTX_YG[4]	ADJ1_MTX_YG[3]	ADJ1_MTX_YG[2]	ADJ1_MTX_YG[1]	ADJ1_MTX_YG[0]
		-	-	-	-	-	ADJ1_MTX_GG [10]	ADJ1_MTX_GG[9]	ADJ1_MTX_GG[8]
	ADJ1_MTX_YG_ADJ1	ADJ1_MTX_GG[7]	ADJ1_MTX_GG[6]	ADJ1_MTX_GG[5]	ADJ1_MTX_GG[4]	ADJ1_MTX_GG[3]	ADJ1_MTX_GG[2]	ADJ1_MTX_GG[1]	ADJ1_MTX_GG[0]
		-	-	-	-	-	ADJ1_MTX_GB [10]	ADJ1_MTX_GB[9]	ADJ1_MTX_GB[8]
		ADJ1_MTX_GB[7]	ADJ1_MTX_GB[6]	ADJ1_MTX_GB[5]	ADJ1_MTX_GB[4]	ADJ1_MTX_GB[3]	ADJ1_MTX_GB[2]	ADJ1_MTX_GB[1]	ADJ1_MTX_GB[0]
		-	-	-	-	-	ADJ1_MTX_GR [10]	ADJ1_MTX_GR[9]	ADJ1_MTX_GR[8]
	ADJ1_MTX_YG_ADJ1	ADJ1_MTX_GR[7]	ADJ1_MTX_GR[6]	ADJ1_MTX_GR[5]	ADJ1_MTX_GR[4]	ADJ1_MTX_GR[3]	ADJ1_MTX_GR[2]	ADJ1_MTX_GR[1]	ADJ1_MTX_GR[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 1	ADJ1_MTX_CBB_ADJ0	-	-	-	-	-	-	-	-	
		ADJ1_MTX_B[7]	ADJ1_MTX_B[6]	ADJ1_MTX_B[5]	ADJ1_MTX_B[4]	ADJ1_MTX_B[3]	ADJ1_MTX_B[2]	ADJ1_MTX_B[1]	ADJ1_MTX_B[0]	
		-	-	-	-	-	ADJ1_MTX_BG[10]	ADJ1_MTX_BG[9]	ADJ1_MTX_BG[8]	
		ADJ1_MTX_BG[7]	ADJ1_MTX_BG[6]	ADJ1_MTX_BG[5]	ADJ1_MTX_BG[4]	ADJ1_MTX_BG[3]	ADJ1_MTX_BG[2]	ADJ1_MTX_BG[1]	ADJ1_MTX_BG[0]	
	ADJ1_MTX_CBB_ADJ1	-	-	-	-	-	-	ADJ1_MTX_BB[10]	ADJ1_MTX_BB[9]	ADJ1_MTX_BB[8]
		ADJ1_MTX_BB[7]	ADJ1_MTX_BB[6]	ADJ1_MTX_BB[5]	ADJ1_MTX_BB[4]	ADJ1_MTX_BB[3]	ADJ1_MTX_BB[2]	ADJ1_MTX_BB[1]	ADJ1_MTX_BB[0]	
		-	-	-	-	-	-	ADJ1_MTX_BR[10]	ADJ1_MTX_BR[9]	ADJ1_MTX_BR[8]
		ADJ1_MTX_BR[7]	ADJ1_MTX_BR[6]	ADJ1_MTX_BR[5]	ADJ1_MTX_BR[4]	ADJ1_MTX_BR[3]	ADJ1_MTX_BR[2]	ADJ1_MTX_BR[1]	ADJ1_MTX_BR[0]	
	ADJ1_MTX_CRR_ADJ0	-	-	-	-	-	-	-	-	
		ADJ1_MTX_R[7]	ADJ1_MTX_R[6]	ADJ1_MTX_R[5]	ADJ1_MTX_R[4]	ADJ1_MTX_R[3]	ADJ1_MTX_R[2]	ADJ1_MTX_R[1]	ADJ1_MTX_R[0]	
		-	-	-	-	-	ADJ1_MTX_RG[10]	ADJ1_MTX_RG[9]	ADJ1_MTX_RG[8]	
	ADJ1_MTX_CRR_ADJ1	ADJ1_MTX_RG[7]	ADJ1_MTX_RG[6]	ADJ1_MTX_RG[5]	ADJ1_MTX_RG[4]	ADJ1_MTX_RG[3]	ADJ1_MTX_RG[2]	ADJ1_MTX_RG[1]	ADJ1_MTX_RG[0]	
		-	-	-	-	-	-	ADJ1_MTX_RB[10]	ADJ1_MTX_RB[9]	ADJ1_MTX_RB[8]
		ADJ1_MTX_RB[7]	ADJ1_MTX_RB[6]	ADJ1_MTX_RB[5]	ADJ1_MTX_RB[4]	ADJ1_MTX_RB[3]	ADJ1_MTX_RB[2]	ADJ1_MTX_RB[1]	ADJ1_MTX_RB[0]	
		-	-	-	-	-	-	ADJ1_MTX_RR[10]	ADJ1_MTX_RR[9]	ADJ1_MTX_RR[8]
	GR2_UPDATE	ADJ1_MTX_RR[7]	ADJ1_MTX_RR[6]	ADJ1_MTX_RR[5]	ADJ1_MTX_RR[4]	ADJ1_MTX_RR[3]	ADJ1_MTX_RR[2]	ADJ1_MTX_RR[1]	ADJ1_MTX_RR[0]	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	GR2_P_VEN	-	-	-	-	GR2_UPDATE
	GR2_FLM_RD	-	-	-	-	-	-	-	-	GR2_IBUS_VEN
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	GR2_R_ENB
	GR2_FLM1	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	GR2_LN_OFF_DIR
		-	-	-	-	-	-	-	GR2_FLM_SEL[1]	GR2_FLM_SEL[0]
		-	-	-	-	-	-	-	-	GR2_BST_MD
	GR2_FLM2	GR2_BASE[31]	GR2_BASE[30]	GR2_BASE[29]	GR2_BASE[28]	GR2_BASE[27]	GR2_BASE[26]	GR2_BASE[25]	GR2_BASE[24]	
		GR2_BASE[23]	GR2_BASE[22]	GR2_BASE[21]	GR2_BASE[20]	GR2_BASE[19]	GR2_BASE[18]	GR2_BASE[17]	GR2_BASE[16]	
		GR2_BASE[15]	GR2_BASE[14]	GR2_BASE[13]	GR2_BASE[12]	GR2_BASE[11]	GR2_BASE[10]	GR2_BASE[9]	GR2_BASE[8]	
		GR2_BASE[7]	GR2_BASE[6]	GR2_BASE[5]	GR2_BASE[4]	GR2_BASE[3]	GR2_BASE[2]	GR2_BASE[1]	GR2_BASE[0]	
	GR2_FLM3	-	GR2_LN_OFF[14]	GR2_LN_OFF[13]	GR2_LN_OFF[12]	GR2_LN_OFF[11]	GR2_LN_OFF[10]	GR2_LN_OFF[9]	GR2_LN_OFF[8]	
		GR2_LN_OFF[7]	GR2_LN_OFF[6]	GR2_LN_OFF[5]	GR2_LN_OFF[4]	GR2_LN_OFF[3]	GR2_LN_OFF[2]	GR2_LN_OFF[1]	GR2_LN_OFF[0]	
		-	-	-	-	-	-	GR2_FLM_NUM[9]	GR2_FLM_NUM[8]	
		GR2_FLM_NUM[7]	GR2_FLM_NUM[6]	GR2_FLM_NUM[5]	GR2_FLM_NUM[4]	GR2_FLM_NUM[3]	GR2_FLM_NUM[2]	GR2_FLM_NUM[1]	GR2_FLM_NUM[0]	
	GR2_FLM4	-	-	-	-	-	-	-	-	
		-	GR2_FLM_OFF[22]	GR2_FLM_OFF[21]	GR2_FLM_OFF[20]	GR2_FLM_OFF[19]	GR2_FLM_OFF[18]	GR2_FLM_OFF[17]	GR2_FLM_OFF[16]	
		GR2_FLM_OFF[15]	GR2_FLM_OFF[14]	GR2_FLM_OFF[13]	GR2_FLM_OFF[12]	GR2_FLM_OFF[11]	GR2_FLM_OFF[10]	GR2_FLM_OFF[9]	GR2_FLM_OFF[8]	
		GR2_FLM_OFF[7]	GR2_FLM_OFF[6]	GR2_FLM_OFF[5]	GR2_FLM_OFF[4]	GR2_FLM_OFF[3]	GR2_FLM_OFF[2]	GR2_FLM_OFF[1]	GR2_FLM_OFF[0]	
	GR2_FLM5	-	-	-	-	-	GR2_FLM_LNUM[10]	GR2_FLM_LNUM[9]	GR2_FLM_LNUM[8]	
		GR2_FLM_LNUM[7]	GR2_FLM_LNUM[6]	GR2_FLM_LNUM[5]	GR2_FLM_LNUM[4]	GR2_FLM_LNUM[3]	GR2_FLM_LNUM[2]	GR2_FLM_LNUM[1]	GR2_FLM_LNUM[0]	
		-	-	-	-	-	GR2_FLM_LOOP[10]	GR2_FLM_LOOP[9]	GR2_FLM_LOOP[8]	
		GR2_FLM_LOOP[7]	GR2_FLM_LOOP[6]	GR2_FLM_LOOP[5]	GR2_FLM_LOOP[4]	GR2_FLM_LOOP[3]	GR2_FLM_LOOP[2]	GR2_FLM_LOOP[1]	GR2_FLM_LOOP[0]	
	GR2_FLM6	GR2_FORMAT[3]	GR2_FORMAT[2]	GR2_FORMAT[1]	GR2_FORMAT[0]	-	GR2_HW[10]	GR2_HW[9]	GR2_HW[8]	
GR2_HW[7]		GR2_HW[6]	GR2_HW[5]	GR2_HW[4]	GR2_HW[3]	GR2_HW[2]	GR2_HW[1]	GR2_HW[0]		
-		-	-	GR2_RDSWA[2]	GR2_RDSWA[1]	GR2_RDSWA[0]	-	-		
-		-	GR2_STA_POS[5]	GR2_STA_POS[4]	GR2_STA_POS[3]	GR2_STA_POS[2]	GR2_STA_POS[1]	GR2_STA_POS[0]		

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 1	GR2_AB1	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		GR2_ARC_MUL	GR2_ACALC_MD	-	GR2_ARC_ON	-	-	-	GR2_ARC_DISP_ON	
	GR2_AB2	-	-	-	-	-	-	GR2_GRC_VS[10]	GR2_GRC_VS[9]	GR2_GRC_VS[8]
		GR2_GRC_VS[7]	GR2_GRC_VS[6]	GR2_GRC_VS[5]	GR2_GRC_VS[4]	GR2_GRC_VS[3]	GR2_GRC_VS[2]	GR2_GRC_VS[1]	GR2_GRC_VS[0]	
		-	-	-	-	-	GR2_GRC_VW [10]	GR2_GRC_VW[9]	GR2_GRC_VW[8]	
		GR2_GRC_VW[7]	GR2_GRC_VW[6]	GR2_GRC_VW[5]	GR2_GRC_VW[4]	GR2_GRC_VW[3]	GR2_GRC_VW[2]	GR2_GRC_VW[1]	GR2_GRC_VW[0]	
	GR2_AB3	-	-	-	-	-	-	GR2_GRC_HS [10]	GR2_GRC_HS[9]	GR2_GRC_HS[8]
		GR2_GRC_HS[7]	GR2_GRC_HS[6]	GR2_GRC_HS[5]	GR2_GRC_HS[4]	GR2_GRC_HS[3]	GR2_GRC_HS[2]	GR2_GRC_HS[1]	GR2_GRC_HS[0]	
		-	-	-	-	-	GR2_GRC_HW [10]	GR2_GRC_HW[9]	GR2_GRC_HW[8]	
		GR2_GRC_HW[7]	GR2_GRC_HW[6]	GR2_GRC_HW[5]	GR2_GRC_HW[4]	GR2_GRC_HW[3]	GR2_GRC_HW[2]	GR2_GRC_HW[1]	GR2_GRC_HW[0]	
	GR2_AB4	-	-	-	-	-	-	GR2_ARC_VS[10]	GR2_ARC_VS[9]	GR2_ARC_VS[8]
		GR2_ARC_VS[7]	GR2_ARC_VS[6]	GR2_ARC_VS[5]	GR2_ARC_VS[4]	GR2_ARC_VS[3]	GR2_ARC_VS[2]	GR2_ARC_VS[1]	GR2_ARC_VS[0]	
		-	-	-	-	-	GR2_ARC_VW [10]	GR2_ARC_VW[9]	GR2_ARC_VW[8]	
		GR2_ARC_VW[7]	GR2_ARC_VW[6]	GR2_ARC_VW[5]	GR2_ARC_VW[4]	GR2_ARC_VW[3]	GR2_ARC_VW[2]	GR2_ARC_VW[1]	GR2_ARC_VW[0]	
	GR2_AB5	-	-	-	-	-	-	GR2_ARC_HS[10]	GR2_ARC_HS[9]	GR2_ARC_HS[8]
		GR2_ARC_HS[7]	GR2_ARC_HS[6]	GR2_ARC_HS[5]	GR2_ARC_HS[4]	GR2_ARC_HS[3]	GR2_ARC_HS[2]	GR2_ARC_HS[1]	GR2_ARC_HS[0]	
		-	-	-	-	-	GR2_ARC_HW [10]	GR2_ARC_HW[9]	GR2_ARC_HW[8]	
		GR2_ARC_HW[7]	GR2_ARC_HW[6]	GR2_ARC_HW[5]	GR2_ARC_HW[4]	GR2_ARC_HW[3]	GR2_ARC_HW[2]	GR2_ARC_HW[1]	GR2_ARC_HW[0]	
	GR2_AB6	-	-	-	-	-	-	-	-	GR2_ARC_MODE
		GR2_ARC_COEF[7]	GR2_ARC_COEF[6]	GR2_ARC_COEF[5]	GR2_ARC_COEF[4]	GR2_ARC_COEF[3]	GR2_ARC_COEF[2]	GR2_ARC_COEF[1]	GR2_ARC_COEF[0]	
		-	-	-	-	-	-	-	-	
		GR2_ARC_RATE [7]	GR2_ARC_RATE [6]	GR2_ARC_RATE [5]	GR2_ARC_RATE [4]	GR2_ARC_RATE [3]	GR2_ARC_RATE [2]	GR2_ARC_RATE [1]	GR2_ARC_RATE [0]	
	GR2_AB7	-	-	-	-	-	-	-	-	
		GR2_ARC_DEF[7]	GR2_ARC_DEF[6]	GR2_ARC_DEF[5]	GR2_ARC_DEF[4]	GR2_ARC_DEF[3]	GR2_ARC_DEF[2]	GR2_ARC_DEF[1]	GR2_ARC_DEF[0]	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	GR2_CK_ON	
	GR2_AB8	GR2_CK_KCLUT [7]	GR2_CK_KCLUT [6]	GR2_CK_KCLUT [5]	GR2_CK_KCLUT [4]	GR2_CK_KCLUT [3]	GR2_CK_KCLUT [2]	GR2_CK_KCLUT [1]	GR2_CK_KCLUT [0]	
		GR2_CK_KG[7]	GR2_CK_KG[6]	GR2_CK_KG[5]	GR2_CK_KG[4]	GR2_CK_KG[3]	GR2_CK_KG[2]	GR2_CK_KG[1]	GR2_CK_KG[0]	
		GR2_CK_KB[7]	GR2_CK_KB[6]	GR2_CK_KB[5]	GR2_CK_KB[4]	GR2_CK_KB[3]	GR2_CK_KB[2]	GR2_CK_KB[1]	GR2_CK_KB[0]	
		GR2_CK_KR[7]	GR2_CK_KR[6]	GR2_CK_KR[5]	GR2_CK_KR[4]	GR2_CK_KR[3]	GR2_CK_KR[2]	GR2_CK_KR[1]	GR2_CK_KR[0]	
	GR2_AB9	GR2_CK_A[7]	GR2_CK_A[6]	GR2_CK_A[5]	GR2_CK_A[4]	GR2_CK_A[3]	GR2_CK_A[2]	GR2_CK_A[1]	GR2_CK_A[0]	
		GR2_CK_G[7]	GR2_CK_G[6]	GR2_CK_G[5]	GR2_CK_G[4]	GR2_CK_G[3]	GR2_CK_G[2]	GR2_CK_G[1]	GR2_CK_G[0]	
		GR2_CK_B[7]	GR2_CK_B[6]	GR2_CK_B[5]	GR2_CK_B[4]	GR2_CK_B[3]	GR2_CK_B[2]	GR2_CK_B[1]	GR2_CK_B[0]	
		GR2_CK_R[7]	GR2_CK_R[6]	GR2_CK_R[5]	GR2_CK_R[4]	GR2_CK_R[3]	GR2_CK_R[2]	GR2_CK_R[1]	GR2_CK_R[0]	
	GR2_AB10	GR2_A0[7]	GR2_A0[6]	GR2_A0[5]	GR2_A0[4]	GR2_A0[3]	GR2_A0[2]	GR2_A0[1]	GR2_A0[0]	
		GR2_G0[7]	GR2_G0[6]	GR2_G0[5]	GR2_G0[4]	GR2_G0[3]	GR2_G0[2]	GR2_G0[1]	GR2_G0[0]	
		GR2_B0[7]	GR2_B0[6]	GR2_B0[5]	GR2_B0[4]	GR2_B0[3]	GR2_B0[2]	GR2_B0[1]	GR2_B0[0]	
		GR2_R0[7]	GR2_R0[6]	GR2_R0[5]	GR2_R0[4]	GR2_R0[3]	GR2_R0[2]	GR2_R0[1]	GR2_R0[0]	
	GR2_AB11	GR2_A1[7]	GR2_A1[6]	GR2_A1[5]	GR2_A1[4]	GR2_A1[3]	GR2_A1[2]	GR2_A1[1]	GR2_A1[0]	
		GR2_G1[7]	GR2_G1[6]	GR2_G1[5]	GR2_G1[4]	GR2_G1[3]	GR2_G1[2]	GR2_G1[1]	GR2_G1[0]	
		GR2_B1[7]	GR2_B1[6]	GR2_B1[5]	GR2_B1[4]	GR2_B1[3]	GR2_B1[2]	GR2_B1[1]	GR2_B1[0]	
		GR2_R1[7]	GR2_R1[6]	GR2_R1[5]	GR2_R1[4]	GR2_R1[3]	GR2_R1[2]	GR2_R1[1]	GR2_R1[0]	
	GR2_BASE	-	-	-	-	-	-	-	-	
		GR2_BASE_G[7]	GR2_BASE_G[6]	GR2_BASE_G[5]	GR2_BASE_G[4]	GR2_BASE_G[3]	GR2_BASE_G[2]	GR2_BASE_G[1]	GR2_BASE_G[0]	
		GR2_BASE_B[7]	GR2_BASE_B[6]	GR2_BASE_B[5]	GR2_BASE_B[4]	GR2_BASE_B[3]	GR2_BASE_B[2]	GR2_BASE_B[1]	GR2_BASE_B[0]	
		GR2_BASE_R[7]	GR2_BASE_R[6]	GR2_BASE_R[5]	GR2_BASE_R[4]	GR2_BASE_R[3]	GR2_BASE_R[2]	GR2_BASE_R[1]	GR2_BASE_R[0]	
	GR2_CLUT	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	GR2_CLT_SEL	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 1	GR2_MON	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	GR2_ARC_ST	
	GR3_UPDATE	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	GR3_UPDATE
		-	-	-	GR3_P_VEN	-	-	-	-	GR3_IBUS_VEN
	GR3_FLM_RD	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	GR3_R_ENB
	GR3_FLM1	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	GR3_LN_OFF_DIR
		-	-	-	-	-	-	-	GR3_FLM_SEL[1]	GR3_FLM_SEL[0]
		-	-	-	-	-	-	-	-	GR3_BST_MD
	GR3_FLM2	GR3_BASE[31]	GR3_BASE[30]	GR3_BASE[29]	GR3_BASE[28]	GR3_BASE[27]	GR3_BASE[26]	GR3_BASE[25]	GR3_BASE[24]	GR3_BASE[23]
		GR3_BASE[23]	GR3_BASE[22]	GR3_BASE[21]	GR3_BASE[20]	GR3_BASE[19]	GR3_BASE[18]	GR3_BASE[17]	GR3_BASE[16]	GR3_BASE[15]
		GR3_BASE[15]	GR3_BASE[14]	GR3_BASE[13]	GR3_BASE[12]	GR3_BASE[11]	GR3_BASE[10]	GR3_BASE[9]	GR3_BASE[8]	GR3_BASE[7]
		GR3_BASE[7]	GR3_BASE[6]	GR3_BASE[5]	GR3_BASE[4]	GR3_BASE[3]	GR3_BASE[2]	GR3_BASE[1]	GR3_BASE[0]	-
	GR3_FLM3	-	GR3_LN_OFF[14]	GR3_LN_OFF[13]	GR3_LN_OFF[12]	GR3_LN_OFF[11]	GR3_LN_OFF[10]	GR3_LN_OFF[9]	GR3_LN_OFF[8]	GR3_LN_OFF[7]
		GR3_LN_OFF[7]	GR3_LN_OFF[6]	GR3_LN_OFF[5]	GR3_LN_OFF[4]	GR3_LN_OFF[3]	GR3_LN_OFF[2]	GR3_LN_OFF[1]	GR3_LN_OFF[0]	-
		-	-	-	-	-	-	GR3_FLM_NUM [9]	GR3_FLM_NUM [8]	GR3_FLM_NUM [7]
		GR3_FLM_NUM [7]	GR3_FLM_NUM [6]	GR3_FLM_NUM [5]	GR3_FLM_NUM [4]	GR3_FLM_NUM [3]	GR3_FLM_NUM [2]	GR3_FLM_NUM [1]	GR3_FLM_NUM [0]	-
	GR3_FLM4	-	-	-	-	-	-	-	-	-
		-	GR3_FLM_OFF [22]	GR3_FLM_OFF [21]	GR3_FLM_OFF [20]	GR3_FLM_OFF [19]	GR3_FLM_OFF [18]	GR3_FLM_OFF [17]	GR3_FLM_OFF [16]	GR3_FLM_OFF [15]
		GR3_FLM_OFF [15]	GR3_FLM_OFF [14]	GR3_FLM_OFF [13]	GR3_FLM_OFF [12]	GR3_FLM_OFF [11]	GR3_FLM_OFF [10]	GR3_FLM_OFF [9]	GR3_FLM_OFF [8]	GR3_FLM_OFF [7]
		GR3_FLM_OFF [7]	GR3_FLM_OFF [6]	GR3_FLM_OFF [5]	GR3_FLM_OFF [4]	GR3_FLM_OFF [3]	GR3_FLM_OFF [2]	GR3_FLM_OFF [1]	GR3_FLM_OFF [0]	-
	GR3_FLM5	-	-	-	-	-	-	GR3_FLM_LNUM[10]	GR3_FLM_LNUM[9]	GR3_FLM_LNUM[8]
		GR3_FLM_LNUM[7]	GR3_FLM_LNUM[6]	GR3_FLM_LNUM[5]	GR3_FLM_LNUM[4]	GR3_FLM_LNUM[3]	GR3_FLM_LNUM[2]	GR3_FLM_LNUM[1]	GR3_FLM_LNUM[0]	-
		-	-	-	-	-	GR3_FLM_LOOP [9]	GR3_FLM_LOOP [8]	GR3_FLM_LOOP [7]	GR3_FLM_LOOP [6]
		GR3_FLM_LOOP [7]	GR3_FLM_LOOP [6]	GR3_FLM_LOOP [5]	GR3_FLM_LOOP [4]	GR3_FLM_LOOP [3]	GR3_FLM_LOOP [2]	GR3_FLM_LOOP [1]	GR3_FLM_LOOP [0]	-
	GR3_FLM6	GR3_FORMAT[3]	GR3_FORMAT[2]	GR3_FORMAT[1]	GR3_FORMAT[0]	-	GR3_HW[10]	GR3_HW[9]	GR3_HW[8]	GR3_HW[7]
		GR3_HW[7]	GR3_HW[6]	GR3_HW[5]	GR3_HW[4]	GR3_HW[3]	GR3_HW[2]	GR3_HW[1]	GR3_HW[0]	-
		-	-	-	GR3_RDSWA[2]	GR3_RDSWA[1]	GR3_RDSWA[0]	-	-	-
		-	-	GR3_STA_POS[5]	GR3_STA_POS[4]	GR3_STA_POS[3]	GR3_STA_POS[2]	GR3_STA_POS[1]	GR3_STA_POS[0]	-
	GR3_AB1	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		GR3_ARC_MUL	GR3_ACALC_MD	-	GR3_ARC_ON	-	-	-	GR3_ARC_DISP_ON	-
		-	-	-	GR3_GRC_DISP_ON	-	-	-	GR3_DISP_SEL[1]	GR3_DISP_SEL[0]
	GR3_AB2	-	-	-	-	-	-	GR3_GRC_VS[10]	GR3_GRC_VS[9]	GR3_GRC_VS[8]
		GR3_GRC_VS[7]	GR3_GRC_VS[6]	GR3_GRC_VS[5]	GR3_GRC_VS[4]	GR3_GRC_VS[3]	GR3_GRC_VS[2]	GR3_GRC_VS[1]	GR3_GRC_VS[0]	-
		-	-	-	-	-	GR3_GRC_VW [10]	GR3_GRC_VW [9]	GR3_GRC_VW [8]	GR3_GRC_VW [7]
		GR3_GRC_VW [7]	GR3_GRC_VW [6]	GR3_GRC_VW [5]	GR3_GRC_VW [4]	GR3_GRC_VW [3]	GR3_GRC_VW [2]	GR3_GRC_VW [1]	GR3_GRC_VW [0]	-
	GR3_AB3	-	-	-	-	-	-	GR3_GRC_HS[10]	GR3_GRC_HS[9]	GR3_GRC_HS[8]
		GR3_GRC_HS[7]	GR3_GRC_HS[6]	GR3_GRC_HS[5]	GR3_GRC_HS[4]	GR3_GRC_HS[3]	GR3_GRC_HS[2]	GR3_GRC_HS[1]	GR3_GRC_HS[0]	-
		-	-	-	-	-	GR3_GRC_HW [10]	GR3_GRC_HW [9]	GR3_GRC_HW [8]	GR3_GRC_HW [7]
		GR3_GRC_HW [7]	GR3_GRC_HW [6]	GR3_GRC_HW [5]	GR3_GRC_HW [4]	GR3_GRC_HW [3]	GR3_GRC_HW [2]	GR3_GRC_HW [1]	GR3_GRC_HW [0]	-

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Video display controller 5 channel 1	GR3_AB4	-	-	-	-	-	GR3_ARC_VS[10]	GR3_ARC_VS[9]	GR3_ARC_VS[8]
		GR3_ARC_VS[7]	GR3_ARC_VS[6]	GR3_ARC_VS[5]	GR3_ARC_VS[4]	GR3_ARC_VS[3]	GR3_ARC_VS[2]	GR3_ARC_VS[1]	GR3_ARC_VS[0]
		-	-	-	-	-	GR3_ARC_VW[10]	GR3_ARC_VW[9]	GR3_ARC_VW[8]
		GR3_ARC_VW[7]	GR3_ARC_VW[6]	GR3_ARC_VW[5]	GR3_ARC_VW[4]	GR3_ARC_VW[3]	GR3_ARC_VW[2]	GR3_ARC_VW[1]	GR3_ARC_VW[0]
	GR3_AB5	-	-	-	-	-	GR3_ARC_HS[10]	GR3_ARC_HS[9]	GR3_ARC_HS[8]
		GR3_ARC_HS[7]	GR3_ARC_HS[6]	GR3_ARC_HS[5]	GR3_ARC_HS[4]	GR3_ARC_HS[3]	GR3_ARC_HS[2]	GR3_ARC_HS[1]	GR3_ARC_HS[0]
		-	-	-	-	-	GR3_ARC_HW[10]	GR3_ARC_HW[9]	GR3_ARC_HW[8]
		GR3_ARC_HW[7]	GR3_ARC_HW[6]	GR3_ARC_HW[5]	GR3_ARC_HW[4]	GR3_ARC_HW[3]	GR3_ARC_HW[2]	GR3_ARC_HW[1]	GR3_ARC_HW[0]
	GR3_AB6	-	-	-	-	-	-	-	GR3_ARC_MODE
		GR3_ARC_COEF[7]	GR3_ARC_COEF[6]	GR3_ARC_COEF[5]	GR3_ARC_COEF[4]	GR3_ARC_COEF[3]	GR3_ARC_COEF[2]	GR3_ARC_COEF[1]	GR3_ARC_COEF[0]
		-	-	-	-	-	-	-	-
		GR3_ARC_RATE[7]	GR3_ARC_RATE[6]	GR3_ARC_RATE[5]	GR3_ARC_RATE[4]	GR3_ARC_RATE[3]	GR3_ARC_RATE[2]	GR3_ARC_RATE[1]	GR3_ARC_RATE[0]
	GR3_AB7	-	-	-	-	-	-	-	-
		GR3_ARC_DEF[7]	GR3_ARC_DEF[6]	GR3_ARC_DEF[5]	GR3_ARC_DEF[4]	GR3_ARC_DEF[3]	GR3_ARC_DEF[2]	GR3_ARC_DEF[1]	GR3_ARC_DEF[0]
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GR3_CK_ON
	GR3_AB8	GR3_CK_KCLUT[7]	GR3_CK_KCLUT[6]	GR3_CK_KCLUT[5]	GR3_CK_KCLUT[4]	GR3_CK_KCLUT[3]	GR3_CK_KCLUT[2]	GR3_CK_KCLUT[1]	GR3_CK_KCLUT[0]
		GR3_CK_KG[7]	GR3_CK_KG[6]	GR3_CK_KG[5]	GR3_CK_KG[4]	GR3_CK_KG[3]	GR3_CK_KG[2]	GR3_CK_KG[1]	GR3_CK_KG[0]
		GR3_CK_KB[7]	GR3_CK_KB[6]	GR3_CK_KB[5]	GR3_CK_KB[4]	GR3_CK_KB[3]	GR3_CK_KB[2]	GR3_CK_KB[1]	GR3_CK_KB[0]
		GR3_CK_KR[7]	GR3_CK_KR[6]	GR3_CK_KR[5]	GR3_CK_KR[4]	GR3_CK_KR[3]	GR3_CK_KR[2]	GR3_CK_KR[1]	GR3_CK_KR[0]
	GR3_AB9	GR3_CK_A[7]	GR3_CK_A[6]	GR3_CK_A[5]	GR3_CK_A[4]	GR3_CK_A[3]	GR3_CK_A[2]	GR3_CK_A[1]	GR3_CK_A[0]
		GR3_CK_G[7]	GR3_CK_G[6]	GR3_CK_G[5]	GR3_CK_G[4]	GR3_CK_G[3]	GR3_CK_G[2]	GR3_CK_G[1]	GR3_CK_G[0]
		GR3_CK_B[7]	GR3_CK_B[6]	GR3_CK_B[5]	GR3_CK_B[4]	GR3_CK_B[3]	GR3_CK_B[2]	GR3_CK_B[1]	GR3_CK_B[0]
		GR3_CK_R[7]	GR3_CK_R[6]	GR3_CK_R[5]	GR3_CK_R[4]	GR3_CK_R[3]	GR3_CK_R[2]	GR3_CK_R[1]	GR3_CK_R[0]
	GR3_AB10	GR3_A0[7]	GR3_A0[6]	GR3_A0[5]	GR3_A0[4]	GR3_A0[3]	GR3_A0[2]	GR3_A0[1]	GR3_A0[0]
		GR3_G0[7]	GR3_G0[6]	GR3_G0[5]	GR3_G0[4]	GR3_G0[3]	GR3_G0[2]	GR3_G0[1]	GR3_G0[0]
		GR3_B0[7]	GR3_B0[6]	GR3_B0[5]	GR3_B0[4]	GR3_B0[3]	GR3_B0[2]	GR3_B0[1]	GR3_B0[0]
		GR3_R0[7]	GR3_R0[6]	GR3_R0[5]	GR3_R0[4]	GR3_R0[3]	GR3_R0[2]	GR3_R0[1]	GR3_R0[0]
	GR3_AB11	GR3_A1[7]	GR3_A1[6]	GR3_A1[5]	GR3_A1[4]	GR3_A1[3]	GR3_A1[2]	GR3_A1[1]	GR3_A1[0]
		GR3_G1[7]	GR3_G1[6]	GR3_G1[5]	GR3_G1[4]	GR3_G1[3]	GR3_G1[2]	GR3_G1[1]	GR3_G1[0]
		GR3_B1[7]	GR3_B1[6]	GR3_B1[5]	GR3_B1[4]	GR3_B1[3]	GR3_B1[2]	GR3_B1[1]	GR3_B1[0]
		GR3_R1[7]	GR3_R1[6]	GR3_R1[5]	GR3_R1[4]	GR3_R1[3]	GR3_R1[2]	GR3_R1[1]	GR3_R1[0]
	GR3_BASE	-	-	-	-	-	-	-	-
		GR3_BASE_G[7]	GR3_BASE_G[6]	GR3_BASE_G[5]	GR3_BASE_G[4]	GR3_BASE_G[3]	GR3_BASE_G[2]	GR3_BASE_G[1]	GR3_BASE_G[0]
		GR3_BASE_B[7]	GR3_BASE_B[6]	GR3_BASE_B[5]	GR3_BASE_B[4]	GR3_BASE_B[3]	GR3_BASE_B[2]	GR3_BASE_B[1]	GR3_BASE_B[0]
		GR3_BASE_R[7]	GR3_BASE_R[6]	GR3_BASE_R[5]	GR3_BASE_R[4]	GR3_BASE_R[3]	GR3_BASE_R[2]	GR3_BASE_R[1]	GR3_BASE_R[0]
	GR3_CLUT_INT	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GR3_CLT_SEL
		-	-	-	-	-	GR3_LINE[10]	GR3_LINE[9]	GR3_LINE[8]
		GR3_LINE[7]	GR3_LINE[6]	GR3_LINE[5]	GR3_LINE[4]	GR3_LINE[3]	GR3_LINE[2]	GR3_LINE[1]	GR3_LINE[0]
	GR3_MON	-	-	-	-	-	GR3_LIN_STAT[10]	GR3_LIN_STAT[9]	GR3_LIN_STAT[8]
		GR3_LIN_STAT[7]	GR3_LIN_STAT[6]	GR3_LIN_STAT[5]	GR3_LIN_STAT[4]	GR3_LIN_STAT[3]	GR3_LIN_STAT[2]	GR3_LIN_STAT[1]	GR3_LIN_STAT[0]
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GR3_ARC_ST
	GR_VIN_UPDATE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GR_VIN_UPDATE
		-	-	-	GR_VIN_P_VEN	-	-	-	-
	GR_VIN_AB1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	GR_VIN_ARC_ON	-	-	-	GR_VIN_ARC_DISP_ON
		-	-	-	GR_VIN_GRC_DISP_ON	-	GR_VIN_SCL_UND_SEL	GR_VIN_DISP_SEL[1]	GR_VIN_DISP_SEL[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 1	GR_VIN_AB2	-	-	-	-	-	GR_VIN_GRC_VS[10]	GR_VIN_GRC_VS[9]	GR_VIN_GRC_VS[8]	
		GR_VIN_GRC_VS[7]	GR_VIN_GRC_VS[6]	GR_VIN_GRC_VS[5]	GR_VIN_GRC_VS[4]	GR_VIN_GRC_VS[3]	GR_VIN_GRC_VS[2]	GR_VIN_GRC_VS[1]	GR_VIN_GRC_VS[0]	
		-	-	-	-	-	GR_VIN_GRC_VW[10]	GR_VIN_GRC_VW[9]	GR_VIN_GRC_VW[8]	
		GR_VIN_GRC_VW[7]	GR_VIN_GRC_VW[6]	GR_VIN_GRC_VW[5]	GR_VIN_GRC_VW[4]	GR_VIN_GRC_VW[3]	GR_VIN_GRC_VW[2]	GR_VIN_GRC_VW[1]	GR_VIN_GRC_VW[0]	
	GR_VIN_AB3	-	-	-	-	-	-	GR_VIN_GRC_HS[10]	GR_VIN_GRC_HS[9]	GR_VIN_GRC_HS[8]
		GR_VIN_GRC_HS[7]	GR_VIN_GRC_HS[6]	GR_VIN_GRC_HS[5]	GR_VIN_GRC_HS[4]	GR_VIN_GRC_HS[3]	GR_VIN_GRC_HS[2]	GR_VIN_GRC_HS[1]	GR_VIN_GRC_HS[0]	
		-	-	-	-	-	-	GR_VIN_GRC_HW[10]	GR_VIN_GRC_HW[9]	GR_VIN_GRC_HW[8]
		GR_VIN_GRC_HW[7]	GR_VIN_GRC_HW[6]	GR_VIN_GRC_HW[5]	GR_VIN_GRC_HW[4]	GR_VIN_GRC_HW[3]	GR_VIN_GRC_HW[2]	GR_VIN_GRC_HW[1]	GR_VIN_GRC_HW[0]	
	GR_VIN_AB4	-	-	-	-	-	-	GR_VIN_ARC_VS[10]	GR_VIN_ARC_VS[9]	GR_VIN_ARC_VS[8]
		GR_VIN_ARC_VS[7]	GR_VIN_ARC_VS[6]	GR_VIN_ARC_VS[5]	GR_VIN_ARC_VS[4]	GR_VIN_ARC_VS[3]	GR_VIN_ARC_VS[2]	GR_VIN_ARC_VS[1]	GR_VIN_ARC_VS[0]	
		-	-	-	-	-	-	GR_VIN_ARC_VW[10]	GR_VIN_ARC_VW[9]	GR_VIN_ARC_VW[8]
		GR_VIN_ARC_VW[7]	GR_VIN_ARC_VW[6]	GR_VIN_ARC_VW[5]	GR_VIN_ARC_VW[4]	GR_VIN_ARC_VW[3]	GR_VIN_ARC_VW[2]	GR_VIN_ARC_VW[1]	GR_VIN_ARC_VW[0]	
	GR_VIN_AB5	-	-	-	-	-	-	GR_VIN_ARC_HS[10]	GR_VIN_ARC_HS[9]	GR_VIN_ARC_HS[8]
		GR_VIN_ARC_HS[7]	GR_VIN_ARC_HS[6]	GR_VIN_ARC_HS[5]	GR_VIN_ARC_HS[4]	GR_VIN_ARC_HS[3]	GR_VIN_ARC_HS[2]	GR_VIN_ARC_HS[1]	GR_VIN_ARC_HS[0]	
		-	-	-	-	-	-	GR_VIN_ARC_HW[10]	GR_VIN_ARC_HW[9]	GR_VIN_ARC_HW[8]
		GR_VIN_ARC_HW[7]	GR_VIN_ARC_HW[6]	GR_VIN_ARC_HW[5]	GR_VIN_ARC_HW[4]	GR_VIN_ARC_HW[3]	GR_VIN_ARC_HW[2]	GR_VIN_ARC_HW[1]	GR_VIN_ARC_HW[0]	
	GR_VIN_AB6	-	-	-	-	-	-	-	-	GR_VIN_ARC_MODE
		GR_VIN_ARC_COE F[7]	GR_VIN_ARC_COE F[6]	GR_VIN_ARC_COE F[5]	GR_VIN_ARC_COE F[4]	GR_VIN_ARC_COE F[3]	GR_VIN_ARC_COE F[2]	GR_VIN_ARC_COE F[1]	GR_VIN_ARC_COE F[0]	
		-	-	-	-	-	-	-	-	
		GR_VIN_ARC_RAT E[7]	GR_VIN_ARC_RAT E[6]	GR_VIN_ARC_RAT E[5]	GR_VIN_ARC_RAT E[4]	GR_VIN_ARC_RAT E[3]	GR_VIN_ARC_RAT E[2]	GR_VIN_ARC_RAT E[1]	GR_VIN_ARC_RAT E[0]	
	GR_VIN_AB7	-	-	-	-	-	-	-	-	-
		GR_VIN_ARC_DEF [7]	GR_VIN_ARC_DEF [6]	GR_VIN_ARC_DEF [5]	GR_VIN_ARC_DEF [4]	GR_VIN_ARC_DEF [3]	GR_VIN_ARC_DEF [2]	GR_VIN_ARC_DEF [1]	GR_VIN_ARC_DEF [0]	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
	GR_VIN_BASE	-	-	-	-	-	-	-	-	-
		GR_VIN_BASE_G[7]	GR_VIN_BASE_G[6]	GR_VIN_BASE_G[5]	GR_VIN_BASE_G[4]	GR_VIN_BASE_G[3]	GR_VIN_BASE_G[2]	GR_VIN_BASE_G[1]	GR_VIN_BASE_G[0]	
		GR_VIN_BASE_B[7]	GR_VIN_BASE_B[6]	GR_VIN_BASE_B[5]	GR_VIN_BASE_B[4]	GR_VIN_BASE_B[3]	GR_VIN_BASE_B[2]	GR_VIN_BASE_B[1]	GR_VIN_BASE_B[0]	
		GR_VIN_BASE_R[7]	GR_VIN_BASE_R[6]	GR_VIN_BASE_R[5]	GR_VIN_BASE_R[4]	GR_VIN_BASE_R[3]	GR_VIN_BASE_R[2]	GR_VIN_BASE_R[1]	GR_VIN_BASE_R[0]	
	GR_VIN_MON	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	GR_VIN_ARC_ST
	OIR_SCL0_UPDATE	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	OIR_SCL0_VEN_D	OIR_SCL0_VEN_C	-	-	-	-	OIR_SCL0_UPDATE
		-	-	-	OIR_SCL0_VEN_B	-	-	-	-	OIR_SCL0_VEN_A
	OIR_SCL0_FRC1	OIR_RES_VMASK[15]	OIR_RES_VMASK[14]	OIR_RES_VMASK[13]	OIR_RES_VMASK[12]	OIR_RES_VMASK[11]	OIR_RES_VMASK[10]	OIR_RES_VMASK[9]	OIR_RES_VMASK[8]	
		OIR_RES_VMASK[7]	OIR_RES_VMASK[6]	OIR_RES_VMASK[5]	OIR_RES_VMASK[4]	OIR_RES_VMASK[3]	OIR_RES_VMASK[2]	OIR_RES_VMASK[1]	OIR_RES_VMASK[0]	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	OIR_RES_VMASK_ON	

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 1	OIR_SCL0_FRC2	OIR_RES_VLACK[15]	OIR_RES_VLACK[14]	OIR_RES_VLACK[13]	OIR_RES_VLACK[12]	OIR_RES_VLACK[11]	OIR_RES_VLACK[10]	OIR_RES_VLACK[9]	OIR_RES_VLACK[8]	
		OIR_RES_VLACK[7]	OIR_RES_VLACK[6]	OIR_RES_VLACK[5]	OIR_RES_VLACK[4]	OIR_RES_VLACK[3]	OIR_RES_VLACK[2]	OIR_RES_VLACK[1]	OIR_RES_VLACK[0]	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	OIR_RES_VLACK_ON	
	OIR_SCL0_FRC3	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	OIR_RES_EN	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	OIR_RES_VS_SEL	
	OIR_SCL0_FRC4	-	-	-	-	-	-	OIR_RES_FV[10]	OIR_RES_FV[9]	OIR_RES_FV[8]
		OIR_RES_FV[7]	OIR_RES_FV[6]	OIR_RES_FV[5]	OIR_RES_FV[4]	OIR_RES_FV[3]	OIR_RES_FV[2]	OIR_RES_FV[1]	OIR_RES_FV[0]	
		-	-	-	-	-	OIR_RES_FH[10]	OIR_RES_FH[9]	OIR_RES_FH[8]	
		OIR_RES_FH[7]	OIR_RES_FH[6]	OIR_RES_FH[5]	OIR_RES_FH[4]	OIR_RES_FH[3]	OIR_RES_FH[2]	OIR_RES_FH[1]	OIR_RES_FH[0]	
	OIR_SCL0_FRC5	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		OIR_RES_VSDLY[7]	OIR_RES_VSDLY[6]	OIR_RES_VSDLY[5]	OIR_RES_VSDLY[4]	OIR_RES_VSDLY[3]	OIR_RES_VSDLY[2]	OIR_RES_VSDLY[1]	OIR_RES_VSDLY[0]	
	OIR_SCL0_FRC6	-	-	-	-	-	-	OIR_RES_F_VS [10]	OIR_RES_F_VS [9]	OIR_RES_F_VS [8]
		OIR_RES_F_VS [7]	OIR_RES_F_VS [6]	OIR_RES_F_VS [5]	OIR_RES_F_VS [4]	OIR_RES_F_VS [3]	OIR_RES_F_VS [2]	OIR_RES_F_VS [1]	OIR_RES_F_VS [0]	
		-	-	-	-	-	OIR_RES_F_VW [10]	OIR_RES_F_VW [9]	OIR_RES_F_VW [8]	
		OIR_RES_F_VW [7]	OIR_RES_F_VW [6]	OIR_RES_F_VW [5]	OIR_RES_F_VW [4]	OIR_RES_F_VW [3]	OIR_RES_F_VW [2]	OIR_RES_F_VW [1]	OIR_RES_F_VW [0]	
	OIR_SCL0_FRC7	-	-	-	-	-	-	OIR_RES_F_HS [10]	OIR_RES_F_HS [9]	OIR_RES_F_HS [8]
		OIR_RES_F_HS [7]	OIR_RES_F_HS [6]	OIR_RES_F_HS [5]	OIR_RES_F_HS [4]	OIR_RES_F_HS [3]	OIR_RES_F_HS [2]	OIR_RES_F_HS [1]	OIR_RES_F_HS [0]	
		-	-	-	-	-	OIR_RES_F_HW [10]	OIR_RES_F_HW [9]	OIR_RES_F_HW [8]	
		OIR_RES_F_HW [7]	OIR_RES_F_HW [6]	OIR_RES_F_HW [5]	OIR_RES_F_HW [4]	OIR_RES_F_HW [3]	OIR_RES_F_HW [2]	OIR_RES_F_HW [1]	OIR_RES_F_HW [0]	
	OIR_SCL0_DS1	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	OIR_RES_DS_V_ON	-	-	-	OIR_RES_DS_H_ON	
	OIR_SCL0_DS2	-	-	-	-	-	-	OIR_RES_VS[10]	OIR_RES_VS[9]	OIR_RES_VS[8]
		OIR_RES_VS[7]	OIR_RES_VS[6]	OIR_RES_VS[5]	OIR_RES_VS[4]	OIR_RES_VS[3]	OIR_RES_VS[2]	OIR_RES_VS[1]	OIR_RES_VS[0]	
		-	-	-	-	-	OIR_RES_VW[10]	OIR_RES_VW[9]	OIR_RES_VW[8]	
		OIR_RES_VW[7]	OIR_RES_VW[6]	OIR_RES_VW[5]	OIR_RES_VW[4]	OIR_RES_VW[3]	OIR_RES_VW[2]	OIR_RES_VW[1]	OIR_RES_VW[0]	
	OIR_SCL0_DS3	-	-	-	-	-	-	OIR_RES_HS[10]	OIR_RES_HS[9]	OIR_RES_HS[8]
		OIR_RES_HS[7]	OIR_RES_HS[6]	OIR_RES_HS[5]	OIR_RES_HS[4]	OIR_RES_HS[3]	OIR_RES_HS[2]	OIR_RES_HS[1]	OIR_RES_HS[0]	
		-	-	-	-	-	OIR_RES_HW[10]	OIR_RES_HW[9]	OIR_RES_HW[8]	
		OIR_RES_HW[7]	OIR_RES_HW[6]	OIR_RES_HW[5]	OIR_RES_HW[4]	OIR_RES_HW[3]	OIR_RES_HW[2]	OIR_RES_HW[1]	OIR_RES_HW[0]	
	OIR_SCL0_DS7	-	-	-	-	-	-	OIR_RES_OUT_VW[10]	OIR_RES_OUT_VW[9]	OIR_RES_OUT_VW[8]
		OIR_RES_OUT_VW[7]	OIR_RES_OUT_VW[6]	OIR_RES_OUT_VW[5]	OIR_RES_OUT_VW[4]	OIR_RES_OUT_VW[3]	OIR_RES_OUT_VW[2]	OIR_RES_OUT_VW[1]	OIR_RES_OUT_VW[0]	
		-	-	-	-	-	OIR_RES_OUT_HW[10]	OIR_RES_OUT_HW[9]	OIR_RES_OUT_HW[8]	
		OIR_RES_OUT_HW[7]	OIR_RES_OUT_HW[6]	OIR_RES_OUT_HW[5]	OIR_RES_OUT_HW[4]	OIR_RES_OUT_HW[3]	OIR_RES_OUT_HW[2]	OIR_RES_OUT_HW[1]	OIR_RES_OUT_HW[0]	
	OIR_SCL0_US1	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	OIR_RES_US_V_ON	-	-	-	OIR_RES_US_H_ON	



Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 1	OIR_SCL0_US2	-	-	-	-	-	OIR_RES_P_VS[10]	OIR_RES_P_VS[9]	OIR_RES_P_VS[8]	
		OIR_RES_P_VS[7]	OIR_RES_P_VS[6]	OIR_RES_P_VS[5]	OIR_RES_P_VS[4]	OIR_RES_P_VS[3]	OIR_RES_P_VS[2]	OIR_RES_P_VS[1]	OIR_RES_P_VS[0]	
		-	-	-	-	-	OIR_RES_P_VW[10]	OIR_RES_P_VW[9]	OIR_RES_P_VW[8]	
		OIR_RES_P_VW[7]	OIR_RES_P_VW[6]	OIR_RES_P_VW[5]	OIR_RES_P_VW[4]	OIR_RES_P_VW[3]	OIR_RES_P_VW[2]	OIR_RES_P_VW[1]	OIR_RES_P_VW[0]	
	OIR_SCL0_US3	-	-	-	-	-	-	OIR_RES_P_HS[10]	OIR_RES_P_HS[9]	OIR_RES_P_HS[8]
		OIR_RES_P_HS[7]	OIR_RES_P_HS[6]	OIR_RES_P_HS[5]	OIR_RES_P_HS[4]	OIR_RES_P_HS[3]	OIR_RES_P_HS[2]	OIR_RES_P_HS[1]	OIR_RES_P_HS[0]	
		-	-	-	-	-	OIR_RES_P_HW[10]	OIR_RES_P_HW[9]	OIR_RES_P_HW[8]	
		OIR_RES_P_HW[7]	OIR_RES_P_HW[6]	OIR_RES_P_HW[5]	OIR_RES_P_HW[4]	OIR_RES_P_HW[3]	OIR_RES_P_HW[2]	OIR_RES_P_HW[1]	OIR_RES_P_HW[0]	
	OIR_SCL0_US8	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	OIR_RES_IBUS_SYN_C_SEL	-	-	-	-	
	OIR_SCL0_OVR1	-	-	-	-	-	-	-	-	
		OIR_RES_BK_COL_R[7]	OIR_RES_BK_COL_R[6]	OIR_RES_BK_COL_R[5]	OIR_RES_BK_COL_R[4]	OIR_RES_BK_COL_R[3]	OIR_RES_BK_COL_R[2]	OIR_RES_BK_COL_R[1]	OIR_RES_BK_COL_R[0]	
		OIR_RES_BK_COL_G[7]	OIR_RES_BK_COL_G[6]	OIR_RES_BK_COL_G[5]	OIR_RES_BK_COL_G[4]	OIR_RES_BK_COL_G[3]	OIR_RES_BK_COL_G[2]	OIR_RES_BK_COL_G[1]	OIR_RES_BK_COL_G[0]	
		OIR_RES_BK_COL_B[7]	OIR_RES_BK_COL_B[6]	OIR_RES_BK_COL_B[5]	OIR_RES_BK_COL_B[4]	OIR_RES_BK_COL_B[3]	OIR_RES_BK_COL_B[2]	OIR_RES_BK_COL_B[1]	OIR_RES_BK_COL_B[0]	
	OIR_SCL1_UPDATE	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	OIR_SCL1_UPDATE_A	
		-	-	-	-	-	-	-	-	
		-	-	-	OIR_SCL1_VEN_B	-	-	-	OIR_SCL1_VEN_A	
	OIR_SCL1_WR1	-	-	-	-	-	-	-	-	
		-	-	-	-	-	OIR_RES_WRSWA[2]	OIR_RES_WRSWA[1]	OIR_RES_WRSWA[0]	
		-	-	-	-	-	-	-	-	
		-	-	-	-	OIR_RES_MD[1]	OIR_RES_MD[0]	-	OIR_RES_BST_MD	
	OIR_SCL1_WR2	OIR_RES_BASE[31]	OIR_RES_BASE[30]	OIR_RES_BASE[29]	OIR_RES_BASE[28]	OIR_RES_BASE[27]	OIR_RES_BASE[26]	OIR_RES_BASE[25]	OIR_RES_BASE[24]	
		OIR_RES_BASE[23]	OIR_RES_BASE[22]	OIR_RES_BASE[21]	OIR_RES_BASE[20]	OIR_RES_BASE[19]	OIR_RES_BASE[18]	OIR_RES_BASE[17]	OIR_RES_BASE[16]	
		OIR_RES_BASE[15]	OIR_RES_BASE[14]	OIR_RES_BASE[13]	OIR_RES_BASE[12]	OIR_RES_BASE[11]	OIR_RES_BASE[10]	OIR_RES_BASE[9]	OIR_RES_BASE[8]	
		OIR_RES_BASE[7]	OIR_RES_BASE[6]	OIR_RES_BASE[5]	OIR_RES_BASE[4]	OIR_RES_BASE[3]	OIR_RES_BASE[2]	OIR_RES_BASE[1]	OIR_RES_BASE[0]	
	OIR_SCL1_WR3	-	OIR_RES_LN_OFF[14]	OIR_RES_LN_OFF[13]	OIR_RES_LN_OFF[12]	OIR_RES_LN_OFF[11]	OIR_RES_LN_OFF[10]	OIR_RES_LN_OFF[9]	OIR_RES_LN_OFF[8]	
		OIR_RES_LN_OFF[7]	OIR_RES_LN_OFF[6]	OIR_RES_LN_OFF[5]	OIR_RES_LN_OFF[4]	OIR_RES_LN_OFF[3]	OIR_RES_LN_OFF[2]	OIR_RES_LN_OFF[1]	OIR_RES_LN_OFF[0]	
		-	-	-	-	-	-	OIR_RES_FLM_NUM[9]	OIR_RES_FLM_NUM[8]	
		OIR_RES_FLM_NUM[7]	OIR_RES_FLM_NUM[6]	OIR_RES_FLM_NUM[5]	OIR_RES_FLM_NUM[4]	OIR_RES_FLM_NUM[3]	OIR_RES_FLM_NUM[2]	OIR_RES_FLM_NUM[1]	OIR_RES_FLM_NUM[0]	
	OIR_SCL1_WR4	-	-	-	-	-	-	-	-	
		-	OIR_RES_FLM_OFF[22]	OIR_RES_FLM_OFF[21]	OIR_RES_FLM_OFF[20]	OIR_RES_FLM_OFF[19]	OIR_RES_FLM_OFF[18]	OIR_RES_FLM_OFF[17]	OIR_RES_FLM_OFF[16]	
		OIR_RES_FLM_OFF[15]	OIR_RES_FLM_OFF[14]	OIR_RES_FLM_OFF[13]	OIR_RES_FLM_OFF[12]	OIR_RES_FLM_OFF[11]	OIR_RES_FLM_OFF[10]	OIR_RES_FLM_OFF[9]	OIR_RES_FLM_OFF[8]	
		OIR_RES_FLM_OFF[7]	OIR_RES_FLM_OFF[6]	OIR_RES_FLM_OFF[5]	OIR_RES_FLM_OFF[4]	OIR_RES_FLM_OFF[3]	OIR_RES_FLM_OFF[2]	OIR_RES_FLM_OFF[1]	OIR_RES_FLM_OFF[0]	
	OIR_SCL1_WR5	-	-	-	-	-	-	-	-	
		-	-	-	OIR_RES_INTER	-	-	-	-	
		-	-	-	-	-	-	OIR_RES_FS_RATE[1]	OIR_RES_FS_RATE[0]	
		-	-	-	-	-	-	-	OIR_RES_WENB	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 1	OIR_SCL1_WR6	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	OIR_RES_DTH_ON	-	-	-	-	OIR_RES_BITDEC_ON
	OIR_SCL1_WR7	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	OIR_RES_OVERFLOW
		-	-	-	-	-	-	OIR_RES_FLM_CNT [9]	OIR_RES_FLM_CNT [8]	-
		OIR_RES_FLM_CNT [7]	OIR_RES_FLM_CNT [6]	OIR_RES_FLM_CNT [5]	OIR_RES_FLM_CNT [4]	OIR_RES_FLM_CNT [3]	OIR_RES_FLM_CNT [2]	OIR_RES_FLM_CNT [1]	OIR_RES_FLM_CNT [0]	-
	GR_OIR_UPDATE	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	GR_OIR_UPDATE
		-	-	-	GR_OIR_P_VEN	-	-	-	-	GR_OIR_IBUS_VEN
	GR_OIR_FLM_RD	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	GR_OIR_R_ENB
	GR_OIR_FLM1	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	GR_OIR_LN_OFF_DIR
		-	-	-	-	-	-	GR_OIR_FLM_SEL [1]	GR_OIR_FLM_SEL [0]	-
		-	-	-	GR_OIR_IMR_FLM_INV	-	-	-	-	GR_OIR_BST_MD
	GR_OIR_FLM2	GR_OIR_BASE [31]	GR_OIR_BASE [30]	GR_OIR_BASE [29]	GR_OIR_BASE [28]	GR_OIR_BASE [27]	GR_OIR_BASE [26]	GR_OIR_BASE [25]	GR_OIR_BASE [24]	-
		GR_OIR_BASE [23]	GR_OIR_BASE [22]	GR_OIR_BASE [21]	GR_OIR_BASE [20]	GR_OIR_BASE [19]	GR_OIR_BASE [18]	GR_OIR_BASE [17]	GR_OIR_BASE [16]	-
		GR_OIR_BASE [15]	GR_OIR_BASE [14]	GR_OIR_BASE [13]	GR_OIR_BASE [12]	GR_OIR_BASE [11]	GR_OIR_BASE [10]	GR_OIR_BASE [9]	GR_OIR_BASE [8]	-
		GR_OIR_BASE [7]	GR_OIR_BASE [6]	GR_OIR_BASE [5]	GR_OIR_BASE [4]	GR_OIR_BASE [3]	GR_OIR_BASE [2]	GR_OIR_BASE [1]	GR_OIR_BASE [0]	-
	GR_OIR_FLM3	-	GR_OIR_LN_OFF[14]	GR_OIR_LN_OFF[13]	GR_OIR_LN_OFF[12]	GR_OIR_LN_OFF[11]	GR_OIR_LN_OFF[10]	GR_OIR_LN_OFF[9]	GR_OIR_LN_OFF[8]	-
		GR_OIR_LN_OFF[7]	GR_OIR_LN_OFF[6]	GR_OIR_LN_OFF[5]	GR_OIR_LN_OFF[4]	GR_OIR_LN_OFF[3]	GR_OIR_LN_OFF[2]	GR_OIR_LN_OFF[1]	GR_OIR_LN_OFF[0]	-
		-	-	-	-	-	-	GR_OIR_FLM_NUM[9]	GR_OIR_FLM_NUM[8]	-
		GR_OIR_FLM_NUM[7]	GR_OIR_FLM_NUM[6]	GR_OIR_FLM_NUM[5]	GR_OIR_FLM_NUM[4]	GR_OIR_FLM_NUM[3]	GR_OIR_FLM_NUM[2]	GR_OIR_FLM_NUM[1]	GR_OIR_FLM_NUM[0]	-
	GR_OIR_FLM4	-	-	-	-	-	-	-	-	-
		-	GR_OIR_FLM_OFF [22]	GR_OIR_FLM_OFF [21]	GR_OIR_FLM_OFF [20]	GR_OIR_FLM_OFF [19]	GR_OIR_FLM_OFF [18]	GR_OIR_FLM_OFF [17]	GR_OIR_FLM_OFF [16]	-
		GR_OIR_FLM_OFF [15]	GR_OIR_FLM_OFF [14]	GR_OIR_FLM_OFF [13]	GR_OIR_FLM_OFF [12]	GR_OIR_FLM_OFF [11]	GR_OIR_FLM_OFF [10]	GR_OIR_FLM_OFF [9]	GR_OIR_FLM_OFF [8]	-
		GR_OIR_FLM_OFF [7]	GR_OIR_FLM_OFF [6]	GR_OIR_FLM_OFF [5]	GR_OIR_FLM_OFF [4]	GR_OIR_FLM_OFF [3]	GR_OIR_FLM_OFF [2]	GR_OIR_FLM_OFF [1]	GR_OIR_FLM_OFF [0]	-
	GR_OIR_FLM5	-	-	-	-	-	-	GR_OIR_FLM_LNU M[10]	GR_OIR_FLM_LNU M[9]	GR_OIR_FLM_LNU M[8]
		GR_OIR_FLM_LNU M[7]	GR_OIR_FLM_LNU M[6]	GR_OIR_FLM_LNU M[5]	GR_OIR_FLM_LNU M[4]	GR_OIR_FLM_LNU M[3]	GR_OIR_FLM_LNU M[2]	GR_OIR_FLM_LNU M[1]	GR_OIR_FLM_LNU M[0]	-
		-	-	-	-	-	-	GR_OIR_FLM_LOO P[10]	GR_OIR_FLM_LOO P[9]	GR_OIR_FLM_LOO P[8]
		GR_OIR_FLM_LOO P[7]	GR_OIR_FLM_LOO P[6]	GR_OIR_FLM_LOO P[5]	GR_OIR_FLM_LOO P[4]	GR_OIR_FLM_LOO P[3]	GR_OIR_FLM_LOO P[2]	GR_OIR_FLM_LOO P[1]	GR_OIR_FLM_LOO P[0]	-
	GR_OIR_FLM6	GR_OIR_FORMAT[3]	GR_OIR_FORMAT[2]	GR_OIR_FORMAT[1]	GR_OIR_FORMAT[0]	-	GR_OIR_HW [10]	GR_OIR_HW[9]	GR_OIR_HW[8]	-
		GR_OIR_HW[7]	GR_OIR_HW[6]	GR_OIR_HW[5]	GR_OIR_HW[4]	GR_OIR_HW[3]	GR_OIR_HW[2]	GR_OIR_HW[1]	GR_OIR_HW[0]	-
		-	-	-	GR_OIR_RDSWA[2]	GR_OIR_RDSWA[1]	GR_OIR_RDSWA[0]	-	-	-
		-	-	GR_OIR_STA_POS [5]	GR_OIR_STA_POS [4]	GR_OIR_STA_POS [3]	GR_OIR_STA_POS [2]	GR_OIR_STA_POS [1]	GR_OIR_STA_POS [0]	-

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Video display controller 5 channel 1	GR_OIR_AB1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	GR_OIR_GRC_DISP_ON	-	-	GR_OIR_DISP_SEL[1]	GR_OIR_DISP_SEL[0]
	GR_OIR_AB2	-	-	-	-	-	GR_OIR_GRC_VS[10]	GR_OIR_GRC_VS[9]	GR_OIR_GRC_VS[8]
		GR_OIR_GRC_VS[7]	GR_OIR_GRC_VS[6]	GR_OIR_GRC_VS[5]	GR_OIR_GRC_VS[4]	GR_OIR_GRC_VS[3]	GR_OIR_GRC_VS[2]	GR_OIR_GRC_VS[1]	GR_OIR_GRC_VS[0]
		-	-	-	-	-	GR_OIR_GRC_VW[10]	GR_OIR_GRC_VW[9]	GR_OIR_GRC_VW[8]
		GR_OIR_GRC_VW[7]	GR_OIR_GRC_VW[6]	GR_OIR_GRC_VW[5]	GR_OIR_GRC_VW[4]	GR_OIR_GRC_VW[3]	GR_OIR_GRC_VW[2]	GR_OIR_GRC_VW[1]	GR_OIR_GRC_VW[0]
	GR_OIR_AB3	-	-	-	-	-	GR_OIR_GRC_HS[10]	GR_OIR_GRC_HS[9]	GR_OIR_GRC_HS[8]
		GR_OIR_GRC_HS[7]	GR_OIR_GRC_HS[6]	GR_OIR_GRC_HS[5]	GR_OIR_GRC_HS[4]	GR_OIR_GRC_HS[3]	GR_OIR_GRC_HS[2]	GR_OIR_GRC_HS[1]	GR_OIR_GRC_HS[0]
		-	-	-	-	-	GR_OIR_GRC_HW[10]	GR_OIR_GRC_HW[9]	GR_OIR_GRC_HW[8]
		GR_OIR_GRC_HW[7]	GR_OIR_GRC_HW[6]	GR_OIR_GRC_HW[5]	GR_OIR_GRC_HW[4]	GR_OIR_GRC_HW[3]	GR_OIR_GRC_HW[2]	GR_OIR_GRC_HW[1]	GR_OIR_GRC_HW[0]
	GR_OIR_AB7	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GR_OIR_CK_ON
	GR_OIR_AB8	GR_OIR_CK_KCLUT[7]	GR_OIR_CK_KCLUT[6]	GR_OIR_CK_KCLUT[5]	GR_OIR_CK_KCLUT[4]	GR_OIR_CK_KCLUT[3]	GR_OIR_CK_KCLUT[2]	GR_OIR_CK_KCLUT[1]	GR_OIR_CK_KCLUT[0]
		GR_OIR_CK_KG[7]	GR_OIR_CK_KG[6]	GR_OIR_CK_KG[5]	GR_OIR_CK_KG[4]	GR_OIR_CK_KG[3]	GR_OIR_CK_KG[2]	GR_OIR_CK_KG[1]	GR_OIR_CK_KG[0]
		GR_OIR_CK_KB[7]	GR_OIR_CK_KB[6]	GR_OIR_CK_KB[5]	GR_OIR_CK_KB[4]	GR_OIR_CK_KB[3]	GR_OIR_CK_KB[2]	GR_OIR_CK_KB[1]	GR_OIR_CK_KB[0]
		GR_OIR_CK_KR[7]	GR_OIR_CK_KR[6]	GR_OIR_CK_KR[5]	GR_OIR_CK_KR[4]	GR_OIR_CK_KR[3]	GR_OIR_CK_KR[2]	GR_OIR_CK_KR[1]	GR_OIR_CK_KR[0]
	GR_OIR_AB9	GR_OIR_CK_A[7]	GR_OIR_CK_A[6]	GR_OIR_CK_A[5]	GR_OIR_CK_A[4]	GR_OIR_CK_A[3]	GR_OIR_CK_A[2]	GR_OIR_CK_A[1]	GR_OIR_CK_A[0]
		GR_OIR_CK_G[7]	GR_OIR_CK_G[6]	GR_OIR_CK_G[5]	GR_OIR_CK_G[4]	GR_OIR_CK_G[3]	GR_OIR_CK_G[2]	GR_OIR_CK_G[1]	GR_OIR_CK_G[0]
		GR_OIR_CK_B[7]	GR_OIR_CK_B[6]	GR_OIR_CK_B[5]	GR_OIR_CK_B[4]	GR_OIR_CK_B[3]	GR_OIR_CK_B[2]	GR_OIR_CK_B[1]	GR_OIR_CK_B[0]
		GR_OIR_CK_R[7]	GR_OIR_CK_R[6]	GR_OIR_CK_R[5]	GR_OIR_CK_R[4]	GR_OIR_CK_R[3]	GR_OIR_CK_R[2]	GR_OIR_CK_R[1]	GR_OIR_CK_R[0]
	GR_OIR_AB10	GR_OIR_A0[7]	GR_OIR_A0[6]	GR_OIR_A0[5]	GR_OIR_A0[4]	GR_OIR_A0[3]	GR_OIR_A0[2]	GR_OIR_A0[1]	GR_OIR_A0[0]
		GR_OIR_G0[7]	GR_OIR_G0[6]	GR_OIR_G0[5]	GR_OIR_G0[4]	GR_OIR_G0[3]	GR_OIR_G0[2]	GR_OIR_G0[1]	GR_OIR_G0[0]
		GR_OIR_B0[7]	GR_OIR_B0[6]	GR_OIR_B0[5]	GR_OIR_B0[4]	GR_OIR_B0[3]	GR_OIR_B0[2]	GR_OIR_B0[1]	GR_OIR_B0[0]
		GR_OIR_R0[7]	GR_OIR_R0[6]	GR_OIR_R0[5]	GR_OIR_R0[4]	GR_OIR_R0[3]	GR_OIR_R0[2]	GR_OIR_R0[1]	GR_OIR_R0[0]
	GR_OIR_AB11	GR_OIR_A1[7]	GR_OIR_A1[6]	GR_OIR_A1[5]	GR_OIR_A1[4]	GR_OIR_A1[3]	GR_OIR_A1[2]	GR_OIR_A1[1]	GR_OIR_A1[0]
		GR_OIR_G1[7]	GR_OIR_G1[6]	GR_OIR_G1[5]	GR_OIR_G1[4]	GR_OIR_G1[3]	GR_OIR_G1[2]	GR_OIR_G1[1]	GR_OIR_G1[0]
		GR_OIR_B1[7]	GR_OIR_B1[6]	GR_OIR_B1[5]	GR_OIR_B1[4]	GR_OIR_B1[3]	GR_OIR_B1[2]	GR_OIR_B1[1]	GR_OIR_B1[0]
		GR_OIR_R1[7]	GR_OIR_R1[6]	GR_OIR_R1[5]	GR_OIR_R1[4]	GR_OIR_R1[3]	GR_OIR_R1[2]	GR_OIR_R1[1]	GR_OIR_R1[0]
	GR_OIR_BASE	-	-	-	-	-	-	-	-
		GR_OIR_BASE_G[7]	GR_OIR_BASE_G[6]	GR_OIR_BASE_G[5]	GR_OIR_BASE_G[4]	GR_OIR_BASE_G[3]	GR_OIR_BASE_G[2]	GR_OIR_BASE_G[1]	GR_OIR_BASE_G[0]
		GR_OIR_BASE_B[7]	GR_OIR_BASE_B[6]	GR_OIR_BASE_B[5]	GR_OIR_BASE_B[4]	GR_OIR_BASE_B[3]	GR_OIR_BASE_B[2]	GR_OIR_BASE_B[1]	GR_OIR_BASE_B[0]
	GR_OIR_CLUT	GR_OIR_BASE_R[7]	GR_OIR_BASE_R[6]	GR_OIR_BASE_R[5]	GR_OIR_BASE_R[4]	GR_OIR_BASE_R[3]	GR_OIR_BASE_R[2]	GR_OIR_BASE_R[1]	GR_OIR_BASE_R[0]
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GR_OIR_CLT_SEL
		GR_OIR_LINE[7]	GR_OIR_LINE[6]	GR_OIR_LINE[5]	GR_OIR_LINE[4]	GR_OIR_LINE[3]	GR_OIR_LINE[2]	GR_OIR_LINE[1]	GR_OIR_LINE[0]
	GR_OIR_MON	-	-	-	-	-	OIR_LIN_STAT[10]	OIR_LIN_STAT[9]	OIR_LIN_STAT[8]
		OIR_LIN_STAT[7]	OIR_LIN_STAT[6]	OIR_LIN_STAT[5]	OIR_LIN_STAT[4]	OIR_LIN_STAT[3]	OIR_LIN_STAT[2]	OIR_LIN_STAT[1]	OIR_LIN_STAT[0]
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	GAM_G_UPDATE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GAM_G_VEN

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 1	GAM_SW	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	GAM_ON	
	GAM_G_LUT1	-	-	-	-	-	-	GAM_G_GAIN_00[10]	GAM_G_GAIN_00[9]	GAM_G_GAIN_00[8]
		GAM_G_GAIN_00[7]	GAM_G_GAIN_00[6]	GAM_G_GAIN_00[5]	GAM_G_GAIN_00[4]	GAM_G_GAIN_00[3]	GAM_G_GAIN_00[2]	GAM_G_GAIN_00[1]	GAM_G_GAIN_00[0]	
		-	-	-	-	-	GAM_G_GAIN_01[10]	GAM_G_GAIN_01[9]	GAM_G_GAIN_01[8]	
		GAM_G_GAIN_01[7]	GAM_G_GAIN_01[6]	GAM_G_GAIN_01[5]	GAM_G_GAIN_01[4]	GAM_G_GAIN_01[3]	GAM_G_GAIN_01[2]	GAM_G_GAIN_01[1]	GAM_G_GAIN_01[0]	
	GAM_G_LUT2	-	-	-	-	-	-	GAM_G_GAIN_02[10]	GAM_G_GAIN_02[9]	GAM_G_GAIN_02[8]
		GAM_G_GAIN_02[7]	GAM_G_GAIN_02[6]	GAM_G_GAIN_02[5]	GAM_G_GAIN_02[4]	GAM_G_GAIN_02[3]	GAM_G_GAIN_02[2]	GAM_G_GAIN_02[1]	GAM_G_GAIN_02[0]	
		-	-	-	-	-	GAM_G_GAIN_03[10]	GAM_G_GAIN_03[9]	GAM_G_GAIN_03[8]	
		GAM_G_GAIN_03[7]	GAM_G_GAIN_03[6]	GAM_G_GAIN_03[5]	GAM_G_GAIN_03[4]	GAM_G_GAIN_03[3]	GAM_G_GAIN_03[2]	GAM_G_GAIN_03[1]	GAM_G_GAIN_03[0]	
	GAM_G_LUT3	-	-	-	-	-	-	GAM_G_GAIN_04[10]	GAM_G_GAIN_04[9]	GAM_G_GAIN_04[8]
		GAM_G_GAIN_04[7]	GAM_G_GAIN_04[6]	GAM_G_GAIN_04[5]	GAM_G_GAIN_04[4]	GAM_G_GAIN_04[3]	GAM_G_GAIN_04[2]	GAM_G_GAIN_04[1]	GAM_G_GAIN_04[0]	
		-	-	-	-	-	GAM_G_GAIN_05[10]	GAM_G_GAIN_05[9]	GAM_G_GAIN_05[8]	
		GAM_G_GAIN_05[7]	GAM_G_GAIN_05[6]	GAM_G_GAIN_05[5]	GAM_G_GAIN_05[4]	GAM_G_GAIN_05[3]	GAM_G_GAIN_05[2]	GAM_G_GAIN_05[1]	GAM_G_GAIN_05[0]	
	GAM_G_LUT4	-	-	-	-	-	-	GAM_G_GAIN_06[10]	GAM_G_GAIN_06[9]	GAM_G_GAIN_06[8]
		GAM_G_GAIN_06[7]	GAM_G_GAIN_06[6]	GAM_G_GAIN_06[5]	GAM_G_GAIN_06[4]	GAM_G_GAIN_06[3]	GAM_G_GAIN_06[2]	GAM_G_GAIN_06[1]	GAM_G_GAIN_06[0]	
		-	-	-	-	-	GAM_G_GAIN_07[10]	GAM_G_GAIN_07[9]	GAM_G_GAIN_07[8]	
		GAM_G_GAIN_07[7]	GAM_G_GAIN_07[6]	GAM_G_GAIN_07[5]	GAM_G_GAIN_07[4]	GAM_G_GAIN_07[3]	GAM_G_GAIN_07[2]	GAM_G_GAIN_07[1]	GAM_G_GAIN_07[0]	
	GAM_G_LUT5	-	-	-	-	-	-	GAM_G_GAIN_08[10]	GAM_G_GAIN_08[9]	GAM_G_GAIN_08[8]
		GAM_G_GAIN_08[7]	GAM_G_GAIN_08[6]	GAM_G_GAIN_08[5]	GAM_G_GAIN_08[4]	GAM_G_GAIN_08[3]	GAM_G_GAIN_08[2]	GAM_G_GAIN_08[1]	GAM_G_GAIN_08[0]	
		-	-	-	-	-	GAM_G_GAIN_09[10]	GAM_G_GAIN_09[9]	GAM_G_GAIN_09[8]	
		GAM_G_GAIN_09[7]	GAM_G_GAIN_09[6]	GAM_G_GAIN_09[5]	GAM_G_GAIN_09[4]	GAM_G_GAIN_09[3]	GAM_G_GAIN_09[2]	GAM_G_GAIN_09[1]	GAM_G_GAIN_09[0]	
	GAM_G_LUT6	-	-	-	-	-	-	GAM_G_GAIN_10[10]	GAM_G_GAIN_10[9]	GAM_G_GAIN_10[8]
		GAM_G_GAIN_10[7]	GAM_G_GAIN_10[6]	GAM_G_GAIN_10[5]	GAM_G_GAIN_10[4]	GAM_G_GAIN_10[3]	GAM_G_GAIN_10[2]	GAM_G_GAIN_10[1]	GAM_G_GAIN_10[0]	
		-	-	-	-	-	GAM_G_GAIN_11[10]	GAM_G_GAIN_11[9]	GAM_G_GAIN_11[8]	
		GAM_G_GAIN_11[7]	GAM_G_GAIN_11[6]	GAM_G_GAIN_11[5]	GAM_G_GAIN_11[4]	GAM_G_GAIN_11[3]	GAM_G_GAIN_11[2]	GAM_G_GAIN_11[1]	GAM_G_GAIN_11[0]	
	GAM_G_LUT7	-	-	-	-	-	-	GAM_G_GAIN_12[10]	GAM_G_GAIN_12[9]	GAM_G_GAIN_12[8]
		GAM_G_GAIN_12[7]	GAM_G_GAIN_12[6]	GAM_G_GAIN_12[5]	GAM_G_GAIN_12[4]	GAM_G_GAIN_12[3]	GAM_G_GAIN_12[2]	GAM_G_GAIN_12[1]	GAM_G_GAIN_12[0]	
		-	-	-	-	-	GAM_G_GAIN_13[10]	GAM_G_GAIN_13[9]	GAM_G_GAIN_13[8]	
		GAM_G_GAIN_13[7]	GAM_G_GAIN_13[6]	GAM_G_GAIN_13[5]	GAM_G_GAIN_13[4]	GAM_G_GAIN_13[3]	GAM_G_GAIN_13[2]	GAM_G_GAIN_13[1]	GAM_G_GAIN_13[0]	
	GAM_G_LUT8	-	-	-	-	-	-	GAM_G_GAIN_14[10]	GAM_G_GAIN_14[9]	GAM_G_GAIN_14[8]
		GAM_G_GAIN_14[7]	GAM_G_GAIN_14[6]	GAM_G_GAIN_14[5]	GAM_G_GAIN_14[4]	GAM_G_GAIN_14[3]	GAM_G_GAIN_14[2]	GAM_G_GAIN_14[1]	GAM_G_GAIN_14[0]	
		-	-	-	-	-	GAM_G_GAIN_15[10]	GAM_G_GAIN_15[9]	GAM_G_GAIN_15[8]	
		GAM_G_GAIN_15[7]	GAM_G_GAIN_15[6]	GAM_G_GAIN_15[5]	GAM_G_GAIN_15[4]	GAM_G_GAIN_15[3]	GAM_G_GAIN_15[2]	GAM_G_GAIN_15[1]	GAM_G_GAIN_15[0]	

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 1	GAM_G_LUT9	-	-	-	-	-	GAM_G_GAIN_16[10]	GAM_G_GAIN_16[9]	GAM_G_GAIN_16[8]	
		GAM_G_GAIN_16[7]	GAM_G_GAIN_16[6]	GAM_G_GAIN_16[5]	GAM_G_GAIN_16[4]	GAM_G_GAIN_16[3]	GAM_G_GAIN_16[2]	GAM_G_GAIN_16[1]	GAM_G_GAIN_16[0]	
		-	-	-	-	-	GAM_G_GAIN_17[10]	GAM_G_GAIN_17[9]	GAM_G_GAIN_17[8]	
		GAM_G_GAIN_17[7]	GAM_G_GAIN_17[6]	GAM_G_GAIN_17[5]	GAM_G_GAIN_17[4]	GAM_G_GAIN_17[3]	GAM_G_GAIN_17[2]	GAM_G_GAIN_17[1]	GAM_G_GAIN_17[0]	
	GAM_G_LUT10	-	-	-	-	-	-	GAM_G_GAIN_18[10]	GAM_G_GAIN_18[9]	GAM_G_GAIN_18[8]
		GAM_G_GAIN_18[7]	GAM_G_GAIN_18[6]	GAM_G_GAIN_18[5]	GAM_G_GAIN_18[4]	GAM_G_GAIN_18[3]	GAM_G_GAIN_18[2]	GAM_G_GAIN_18[1]	GAM_G_GAIN_18[0]	
		-	-	-	-	-	-	GAM_G_GAIN_19[10]	GAM_G_GAIN_19[9]	GAM_G_GAIN_19[8]
		GAM_G_GAIN_19[7]	GAM_G_GAIN_19[6]	GAM_G_GAIN_19[5]	GAM_G_GAIN_19[4]	GAM_G_GAIN_19[3]	GAM_G_GAIN_19[2]	GAM_G_GAIN_19[1]	GAM_G_GAIN_19[0]	
	GAM_G_LUT11	-	-	-	-	-	-	GAM_G_GAIN_20[10]	GAM_G_GAIN_20[9]	GAM_G_GAIN_20[8]
		GAM_G_GAIN_20[7]	GAM_G_GAIN_20[6]	GAM_G_GAIN_20[5]	GAM_G_GAIN_20[4]	GAM_G_GAIN_20[3]	GAM_G_GAIN_20[2]	GAM_G_GAIN_20[1]	GAM_G_GAIN_20[0]	
		-	-	-	-	-	-	GAM_G_GAIN_21[10]	GAM_G_GAIN_21[9]	GAM_G_GAIN_21[8]
		GAM_G_GAIN_21[7]	GAM_G_GAIN_21[6]	GAM_G_GAIN_21[5]	GAM_G_GAIN_21[4]	GAM_G_GAIN_21[3]	GAM_G_GAIN_21[2]	GAM_G_GAIN_21[1]	GAM_G_GAIN_21[0]	
	GAM_G_LUT12	-	-	-	-	-	-	GAM_G_GAIN_22[10]	GAM_G_GAIN_22[9]	GAM_G_GAIN_22[8]
		GAM_G_GAIN_22[7]	GAM_G_GAIN_22[6]	GAM_G_GAIN_22[5]	GAM_G_GAIN_22[4]	GAM_G_GAIN_22[3]	GAM_G_GAIN_22[2]	GAM_G_GAIN_22[1]	GAM_G_GAIN_22[0]	
		-	-	-	-	-	-	GAM_G_GAIN_23[10]	GAM_G_GAIN_23[9]	GAM_G_GAIN_23[8]
		GAM_G_GAIN_23[7]	GAM_G_GAIN_23[6]	GAM_G_GAIN_23[5]	GAM_G_GAIN_23[4]	GAM_G_GAIN_23[3]	GAM_G_GAIN_23[2]	GAM_G_GAIN_23[1]	GAM_G_GAIN_23[0]	
	GAM_G_LUT13	-	-	-	-	-	-	GAM_G_GAIN_24[10]	GAM_G_GAIN_24[9]	GAM_G_GAIN_24[8]
		GAM_G_GAIN_24[7]	GAM_G_GAIN_24[6]	GAM_G_GAIN_24[5]	GAM_G_GAIN_24[4]	GAM_G_GAIN_24[3]	GAM_G_GAIN_24[2]	GAM_G_GAIN_24[1]	GAM_G_GAIN_24[0]	
		-	-	-	-	-	-	GAM_G_GAIN_25[10]	GAM_G_GAIN_25[9]	GAM_G_GAIN_25[8]
		GAM_G_GAIN_25[7]	GAM_G_GAIN_25[6]	GAM_G_GAIN_25[5]	GAM_G_GAIN_25[4]	GAM_G_GAIN_25[3]	GAM_G_GAIN_25[2]	GAM_G_GAIN_25[1]	GAM_G_GAIN_25[0]	
	GAM_G_LUT14	-	-	-	-	-	-	GAM_G_GAIN_26[10]	GAM_G_GAIN_26[9]	GAM_G_GAIN_26[8]
		GAM_G_GAIN_26[7]	GAM_G_GAIN_26[6]	GAM_G_GAIN_26[5]	GAM_G_GAIN_26[4]	GAM_G_GAIN_26[3]	GAM_G_GAIN_26[2]	GAM_G_GAIN_26[1]	GAM_G_GAIN_26[0]	
		-	-	-	-	-	-	GAM_G_GAIN_27[10]	GAM_G_GAIN_27[9]	GAM_G_GAIN_27[8]
		GAM_G_GAIN_27[7]	GAM_G_GAIN_27[6]	GAM_G_GAIN_27[5]	GAM_G_GAIN_27[4]	GAM_G_GAIN_27[3]	GAM_G_GAIN_27[2]	GAM_G_GAIN_27[1]	GAM_G_GAIN_27[0]	
	GAM_G_LUT15	-	-	-	-	-	-	GAM_G_GAIN_28[10]	GAM_G_GAIN_28[9]	GAM_G_GAIN_28[8]
		GAM_G_GAIN_28[7]	GAM_G_GAIN_28[6]	GAM_G_GAIN_28[5]	GAM_G_GAIN_28[4]	GAM_G_GAIN_28[3]	GAM_G_GAIN_28[2]	GAM_G_GAIN_28[1]	GAM_G_GAIN_28[0]	
		-	-	-	-	-	-	GAM_G_GAIN_29[10]	GAM_G_GAIN_29[9]	GAM_G_GAIN_29[8]
		GAM_G_GAIN_29[7]	GAM_G_GAIN_29[6]	GAM_G_GAIN_29[5]	GAM_G_GAIN_29[4]	GAM_G_GAIN_29[3]	GAM_G_GAIN_29[2]	GAM_G_GAIN_29[1]	GAM_G_GAIN_29[0]	
	GAM_G_LUT16	-	-	-	-	-	-	GAM_G_GAIN_30[10]	GAM_G_GAIN_30[9]	GAM_G_GAIN_30[8]
		GAM_G_GAIN_30[7]	GAM_G_GAIN_30[6]	GAM_G_GAIN_30[5]	GAM_G_GAIN_30[4]	GAM_G_GAIN_30[3]	GAM_G_GAIN_30[2]	GAM_G_GAIN_30[1]	GAM_G_GAIN_30[0]	
		-	-	-	-	-	-	GAM_G_GAIN_31[10]	GAM_G_GAIN_31[9]	GAM_G_GAIN_31[8]
		GAM_G_GAIN_31[7]	GAM_G_GAIN_31[6]	GAM_G_GAIN_31[5]	GAM_G_GAIN_31[4]	GAM_G_GAIN_31[3]	GAM_G_GAIN_31[2]	GAM_G_GAIN_31[1]	GAM_G_GAIN_31[0]	
	GAM_G_AREA1	-	-	-	-	-	-	-	-	-
		GAM_G_TH_01[7]	GAM_G_TH_01[6]	GAM_G_TH_01[5]	GAM_G_TH_01[4]	GAM_G_TH_01[3]	GAM_G_TH_01[2]	GAM_G_TH_01[1]	GAM_G_TH_01[0]	
		GAM_G_TH_02[7]	GAM_G_TH_02[6]	GAM_G_TH_02[5]	GAM_G_TH_02[4]	GAM_G_TH_02[3]	GAM_G_TH_02[2]	GAM_G_TH_02[1]	GAM_G_TH_02[0]	
		GAM_G_TH_03[7]	GAM_G_TH_03[6]	GAM_G_TH_03[5]	GAM_G_TH_03[4]	GAM_G_TH_03[3]	GAM_G_TH_03[2]	GAM_G_TH_03[1]	GAM_G_TH_03[0]	
	GAM_G_AREA2	GAM_G_TH_04[7]	GAM_G_TH_04[6]	GAM_G_TH_04[5]	GAM_G_TH_04[4]	GAM_G_TH_04[3]	GAM_G_TH_04[2]	GAM_G_TH_04[1]	GAM_G_TH_04[0]	
		GAM_G_TH_05[7]	GAM_G_TH_05[6]	GAM_G_TH_05[5]	GAM_G_TH_05[4]	GAM_G_TH_05[3]	GAM_G_TH_05[2]	GAM_G_TH_05[1]	GAM_G_TH_05[0]	
		GAM_G_TH_06[7]	GAM_G_TH_06[6]	GAM_G_TH_06[5]	GAM_G_TH_06[4]	GAM_G_TH_06[3]	GAM_G_TH_06[2]	GAM_G_TH_06[1]	GAM_G_TH_06[0]	
		GAM_G_TH_07[7]	GAM_G_TH_07[6]	GAM_G_TH_07[5]	GAM_G_TH_07[4]	GAM_G_TH_07[3]	GAM_G_TH_07[2]	GAM_G_TH_07[1]	GAM_G_TH_07[0]	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Video display controller 5 channel 1	GAM_G_AREA3	GAM_G_TH_08[7]	GAM_G_TH_08[6]	GAM_G_TH_08[5]	GAM_G_TH_08[4]	GAM_G_TH_08[3]	GAM_G_TH_08[2]	GAM_G_TH_08[1]	GAM_G_TH_08[0]
		GAM_G_TH_09[7]	GAM_G_TH_09[6]	GAM_G_TH_09[5]	GAM_G_TH_09[4]	GAM_G_TH_09[3]	GAM_G_TH_09[2]	GAM_G_TH_09[1]	GAM_G_TH_09[0]
		GAM_G_TH_10[7]	GAM_G_TH_10[6]	GAM_G_TH_10[5]	GAM_G_TH_10[4]	GAM_G_TH_10[3]	GAM_G_TH_10[2]	GAM_G_TH_10[1]	GAM_G_TH_10[0]
		GAM_G_TH_11[7]	GAM_G_TH_11[6]	GAM_G_TH_11[5]	GAM_G_TH_11[4]	GAM_G_TH_11[3]	GAM_G_TH_11[2]	GAM_G_TH_11[1]	GAM_G_TH_11[0]
	GAM_G_AREA4	GAM_G_TH_12[7]	GAM_G_TH_12[6]	GAM_G_TH_12[5]	GAM_G_TH_12[4]	GAM_G_TH_12[3]	GAM_G_TH_12[2]	GAM_G_TH_12[1]	GAM_G_TH_12[0]
		GAM_G_TH_13[7]	GAM_G_TH_13[6]	GAM_G_TH_13[5]	GAM_G_TH_13[4]	GAM_G_TH_13[3]	GAM_G_TH_13[2]	GAM_G_TH_13[1]	GAM_G_TH_13[0]
		GAM_G_TH_14[7]	GAM_G_TH_14[6]	GAM_G_TH_14[5]	GAM_G_TH_14[4]	GAM_G_TH_14[3]	GAM_G_TH_14[2]	GAM_G_TH_14[1]	GAM_G_TH_14[0]
		GAM_G_TH_15[7]	GAM_G_TH_15[6]	GAM_G_TH_15[5]	GAM_G_TH_15[4]	GAM_G_TH_15[3]	GAM_G_TH_15[2]	GAM_G_TH_15[1]	GAM_G_TH_15[0]
	GAM_G_AREA5	GAM_G_TH_16[7]	GAM_G_TH_16[6]	GAM_G_TH_16[5]	GAM_G_TH_16[4]	GAM_G_TH_16[3]	GAM_G_TH_16[2]	GAM_G_TH_16[1]	GAM_G_TH_16[0]
		GAM_G_TH_17[7]	GAM_G_TH_17[6]	GAM_G_TH_17[5]	GAM_G_TH_17[4]	GAM_G_TH_17[3]	GAM_G_TH_17[2]	GAM_G_TH_17[1]	GAM_G_TH_17[0]
		GAM_G_TH_18[7]	GAM_G_TH_18[6]	GAM_G_TH_18[5]	GAM_G_TH_18[4]	GAM_G_TH_18[3]	GAM_G_TH_18[2]	GAM_G_TH_18[1]	GAM_G_TH_18[0]
		GAM_G_TH_19[7]	GAM_G_TH_19[6]	GAM_G_TH_19[5]	GAM_G_TH_19[4]	GAM_G_TH_19[3]	GAM_G_TH_19[2]	GAM_G_TH_19[1]	GAM_G_TH_19[0]
	GAM_G_AREA6	GAM_G_TH_20[7]	GAM_G_TH_20[6]	GAM_G_TH_20[5]	GAM_G_TH_20[4]	GAM_G_TH_20[3]	GAM_G_TH_20[2]	GAM_G_TH_20[1]	GAM_G_TH_20[0]
		GAM_G_TH_21[7]	GAM_G_TH_21[6]	GAM_G_TH_21[5]	GAM_G_TH_21[4]	GAM_G_TH_21[3]	GAM_G_TH_21[2]	GAM_G_TH_21[1]	GAM_G_TH_21[0]
		GAM_G_TH_22[7]	GAM_G_TH_22[6]	GAM_G_TH_22[5]	GAM_G_TH_22[4]	GAM_G_TH_22[3]	GAM_G_TH_22[2]	GAM_G_TH_22[1]	GAM_G_TH_22[0]
		GAM_G_TH_23[7]	GAM_G_TH_23[6]	GAM_G_TH_23[5]	GAM_G_TH_23[4]	GAM_G_TH_23[3]	GAM_G_TH_23[2]	GAM_G_TH_23[1]	GAM_G_TH_23[0]
	GAM_G_AREA7	GAM_G_TH_24[7]	GAM_G_TH_24[6]	GAM_G_TH_24[5]	GAM_G_TH_24[4]	GAM_G_TH_24[3]	GAM_G_TH_24[2]	GAM_G_TH_24[1]	GAM_G_TH_24[0]
		GAM_G_TH_25[7]	GAM_G_TH_25[6]	GAM_G_TH_25[5]	GAM_G_TH_25[4]	GAM_G_TH_25[3]	GAM_G_TH_25[2]	GAM_G_TH_25[1]	GAM_G_TH_25[0]
		GAM_G_TH_26[7]	GAM_G_TH_26[6]	GAM_G_TH_26[5]	GAM_G_TH_26[4]	GAM_G_TH_26[3]	GAM_G_TH_26[2]	GAM_G_TH_26[1]	GAM_G_TH_26[0]
		GAM_G_TH_27[7]	GAM_G_TH_27[6]	GAM_G_TH_27[5]	GAM_G_TH_27[4]	GAM_G_TH_27[3]	GAM_G_TH_27[2]	GAM_G_TH_27[1]	GAM_G_TH_27[0]
	GAM_G_AREA8	GAM_G_TH_28[7]	GAM_G_TH_28[6]	GAM_G_TH_28[5]	GAM_G_TH_28[4]	GAM_G_TH_28[3]	GAM_G_TH_28[2]	GAM_G_TH_28[1]	GAM_G_TH_28[0]
		GAM_G_TH_29[7]	GAM_G_TH_29[6]	GAM_G_TH_29[5]	GAM_G_TH_29[4]	GAM_G_TH_29[3]	GAM_G_TH_29[2]	GAM_G_TH_29[1]	GAM_G_TH_29[0]
		GAM_G_TH_30[7]	GAM_G_TH_30[6]	GAM_G_TH_30[5]	GAM_G_TH_30[4]	GAM_G_TH_30[3]	GAM_G_TH_30[2]	GAM_G_TH_30[1]	GAM_G_TH_30[0]
		GAM_G_TH_31[7]	GAM_G_TH_31[6]	GAM_G_TH_31[5]	GAM_G_TH_31[4]	GAM_G_TH_31[3]	GAM_G_TH_31[2]	GAM_G_TH_31[1]	GAM_G_TH_31[0]
	GAM_B_UPDATE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GAM_B_VEN
	GAM_B_LUT1	-	-	-	-	-	GAM_B_GAIN_00[10]	GAM_B_GAIN_00[9]	GAM_B_GAIN_00[8]
		GAM_B_GAIN_00[7]	GAM_B_GAIN_00[6]	GAM_B_GAIN_00[5]	GAM_B_GAIN_00[4]	GAM_B_GAIN_00[3]	GAM_B_GAIN_00[2]	GAM_B_GAIN_00[1]	GAM_B_GAIN_00[0]
		-	-	-	-	-	GAM_B_GAIN_01[10]	GAM_B_GAIN_01[9]	GAM_B_GAIN_01[8]
		GAM_B_GAIN_01[7]	GAM_B_GAIN_01[6]	GAM_B_GAIN_01[5]	GAM_B_GAIN_01[4]	GAM_B_GAIN_01[3]	GAM_B_GAIN_01[2]	GAM_B_GAIN_01[1]	GAM_B_GAIN_01[0]
	GAM_B_LUT2	-	-	-	-	-	GAM_B_GAIN_02[10]	GAM_B_GAIN_02[9]	GAM_B_GAIN_02[8]
		GAM_B_GAIN_02[7]	GAM_B_GAIN_02[6]	GAM_B_GAIN_02[5]	GAM_B_GAIN_02[4]	GAM_B_GAIN_02[3]	GAM_B_GAIN_02[2]	GAM_B_GAIN_02[1]	GAM_B_GAIN_02[0]
		-	-	-	-	-	GAM_B_GAIN_03[10]	GAM_B_GAIN_03[9]	GAM_B_GAIN_03[8]
		GAM_B_GAIN_03[7]	GAM_B_GAIN_03[6]	GAM_B_GAIN_03[5]	GAM_B_GAIN_03[4]	GAM_B_GAIN_03[3]	GAM_B_GAIN_03[2]	GAM_B_GAIN_03[1]	GAM_B_GAIN_03[0]
	GAM_B_LUT3	-	-	-	-	-	GAM_B_GAIN_04[10]	GAM_B_GAIN_04[9]	GAM_B_GAIN_04[8]
		GAM_B_GAIN_04[7]	GAM_B_GAIN_04[6]	GAM_B_GAIN_04[5]	GAM_B_GAIN_04[4]	GAM_B_GAIN_04[3]	GAM_B_GAIN_04[2]	GAM_B_GAIN_04[1]	GAM_B_GAIN_04[0]
		-	-	-	-	-	GAM_B_GAIN_05[10]	GAM_B_GAIN_05[9]	GAM_B_GAIN_05[8]
		GAM_B_GAIN_05[7]	GAM_B_GAIN_05[6]	GAM_B_GAIN_05[5]	GAM_B_GAIN_05[4]	GAM_B_GAIN_05[3]	GAM_B_GAIN_05[2]	GAM_B_GAIN_05[1]	GAM_B_GAIN_05[0]
GAM_B_LUT4	-	-	-	-	-	GAM_B_GAIN_06[10]	GAM_B_GAIN_06[9]	GAM_B_GAIN_06[8]	
	GAM_B_GAIN_06[7]	GAM_B_GAIN_06[6]	GAM_B_GAIN_06[5]	GAM_B_GAIN_06[4]	GAM_B_GAIN_06[3]	GAM_B_GAIN_06[2]	GAM_B_GAIN_06[1]	GAM_B_GAIN_06[0]	
	-	-	-	-	-	GAM_B_GAIN_07[10]	GAM_B_GAIN_07[9]	GAM_B_GAIN_07[8]	
	GAM_B_GAIN_07[7]	GAM_B_GAIN_07[6]	GAM_B_GAIN_07[5]	GAM_B_GAIN_07[4]	GAM_B_GAIN_07[3]	GAM_B_GAIN_07[2]	GAM_B_GAIN_07[1]	GAM_B_GAIN_07[0]	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 1	GAM_B_LUT5	-	-	-	-	-	GAM_B_GAIN_08[10]	GAM_B_GAIN_08[9]	GAM_B_GAIN_08[8]	
		GAM_B_GAIN_08[7]	GAM_B_GAIN_08[6]	GAM_B_GAIN_08[5]	GAM_B_GAIN_08[4]	GAM_B_GAIN_08[3]	GAM_B_GAIN_08[2]	GAM_B_GAIN_08[1]	GAM_B_GAIN_08[0]	
		-	-	-	-	-	GAM_B_GAIN_09[10]	GAM_B_GAIN_09[9]	GAM_B_GAIN_09[8]	
		GAM_B_GAIN_09[7]	GAM_B_GAIN_09[6]	GAM_B_GAIN_09[5]	GAM_B_GAIN_09[4]	GAM_B_GAIN_09[3]	GAM_B_GAIN_09[2]	GAM_B_GAIN_09[1]	GAM_B_GAIN_09[0]	
	GAM_B_LUT6	-	-	-	-	-	-	GAM_B_GAIN_10[10]	GAM_B_GAIN_10[9]	GAM_B_GAIN_10[8]
		GAM_B_GAIN_10[7]	GAM_B_GAIN_10[6]	GAM_B_GAIN_10[5]	GAM_B_GAIN_10[4]	GAM_B_GAIN_10[3]	GAM_B_GAIN_10[2]	GAM_B_GAIN_10[1]	GAM_B_GAIN_10[0]	
		-	-	-	-	-	-	GAM_B_GAIN_11[10]	GAM_B_GAIN_11[9]	GAM_B_GAIN_11[8]
		GAM_B_GAIN_11[7]	GAM_B_GAIN_11[6]	GAM_B_GAIN_11[5]	GAM_B_GAIN_11[4]	GAM_B_GAIN_11[3]	GAM_B_GAIN_11[2]	GAM_B_GAIN_11[1]	GAM_B_GAIN_11[0]	
	GAM_B_LUT7	-	-	-	-	-	-	GAM_B_GAIN_12[10]	GAM_B_GAIN_12[9]	GAM_B_GAIN_12[8]
		GAM_B_GAIN_12[7]	GAM_B_GAIN_12[6]	GAM_B_GAIN_12[5]	GAM_B_GAIN_12[4]	GAM_B_GAIN_12[3]	GAM_B_GAIN_12[2]	GAM_B_GAIN_12[1]	GAM_B_GAIN_12[0]	
		-	-	-	-	-	-	GAM_B_GAIN_13[10]	GAM_B_GAIN_13[9]	GAM_B_GAIN_13[8]
		GAM_B_GAIN_13[7]	GAM_B_GAIN_13[6]	GAM_B_GAIN_13[5]	GAM_B_GAIN_13[4]	GAM_B_GAIN_13[3]	GAM_B_GAIN_13[2]	GAM_B_GAIN_13[1]	GAM_B_GAIN_13[0]	
	GAM_B_LUT8	-	-	-	-	-	-	GAM_B_GAIN_14[10]	GAM_B_GAIN_14[9]	GAM_B_GAIN_14[8]
		GAM_B_GAIN_14[7]	GAM_B_GAIN_14[6]	GAM_B_GAIN_14[5]	GAM_B_GAIN_14[4]	GAM_B_GAIN_14[3]	GAM_B_GAIN_14[2]	GAM_B_GAIN_14[1]	GAM_B_GAIN_14[0]	
		-	-	-	-	-	-	GAM_B_GAIN_15[10]	GAM_B_GAIN_15[9]	GAM_B_GAIN_15[8]
		GAM_B_GAIN_15[7]	GAM_B_GAIN_15[6]	GAM_B_GAIN_15[5]	GAM_B_GAIN_15[4]	GAM_B_GAIN_15[3]	GAM_B_GAIN_15[2]	GAM_B_GAIN_15[1]	GAM_B_GAIN_15[0]	
	GAM_B_LUT9	-	-	-	-	-	-	GAM_B_GAIN_16[10]	GAM_B_GAIN_16[9]	GAM_B_GAIN_16[8]
		GAM_B_GAIN_16[7]	GAM_B_GAIN_16[6]	GAM_B_GAIN_16[5]	GAM_B_GAIN_16[4]	GAM_B_GAIN_16[3]	GAM_B_GAIN_16[2]	GAM_B_GAIN_16[1]	GAM_B_GAIN_16[0]	
		-	-	-	-	-	-	GAM_B_GAIN_17[10]	GAM_B_GAIN_17[9]	GAM_B_GAIN_17[8]
		GAM_B_GAIN_17[7]	GAM_B_GAIN_17[6]	GAM_B_GAIN_17[5]	GAM_B_GAIN_17[4]	GAM_B_GAIN_17[3]	GAM_B_GAIN_17[2]	GAM_B_GAIN_17[1]	GAM_B_GAIN_17[0]	
	GAM_B_LUT10	-	-	-	-	-	-	GAM_B_GAIN_18[10]	GAM_B_GAIN_18[9]	GAM_B_GAIN_18[8]
		GAM_B_GAIN_18[7]	GAM_B_GAIN_18[6]	GAM_B_GAIN_18[5]	GAM_B_GAIN_18[4]	GAM_B_GAIN_18[3]	GAM_B_GAIN_18[2]	GAM_B_GAIN_18[1]	GAM_B_GAIN_18[0]	
		-	-	-	-	-	-	GAM_B_GAIN_19[10]	GAM_B_GAIN_19[9]	GAM_B_GAIN_19[8]
		GAM_B_GAIN_19[7]	GAM_B_GAIN_19[6]	GAM_B_GAIN_19[5]	GAM_B_GAIN_19[4]	GAM_B_GAIN_19[3]	GAM_B_GAIN_19[2]	GAM_B_GAIN_19[1]	GAM_B_GAIN_19[0]	
	GAM_B_LUT11	-	-	-	-	-	-	GAM_B_GAIN_20[10]	GAM_B_GAIN_20[9]	GAM_B_GAIN_20[8]
		GAM_B_GAIN_20[7]	GAM_B_GAIN_20[6]	GAM_B_GAIN_20[5]	GAM_B_GAIN_20[4]	GAM_B_GAIN_20[3]	GAM_B_GAIN_20[2]	GAM_B_GAIN_20[1]	GAM_B_GAIN_20[0]	
		-	-	-	-	-	-	GAM_B_GAIN_21[10]	GAM_B_GAIN_21[9]	GAM_B_GAIN_21[8]
		GAM_B_GAIN_21[7]	GAM_B_GAIN_21[6]	GAM_B_GAIN_21[5]	GAM_B_GAIN_21[4]	GAM_B_GAIN_21[3]	GAM_B_GAIN_21[2]	GAM_B_GAIN_21[1]	GAM_B_GAIN_21[0]	
	GAM_B_LUT12	-	-	-	-	-	-	GAM_B_GAIN_22[10]	GAM_B_GAIN_22[9]	GAM_B_GAIN_22[8]
		GAM_B_GAIN_22[7]	GAM_B_GAIN_22[6]	GAM_B_GAIN_22[5]	GAM_B_GAIN_22[4]	GAM_B_GAIN_22[3]	GAM_B_GAIN_22[2]	GAM_B_GAIN_22[1]	GAM_B_GAIN_22[0]	
		-	-	-	-	-	-	GAM_B_GAIN_23[10]	GAM_B_GAIN_23[9]	GAM_B_GAIN_23[8]
		GAM_B_GAIN_23[7]	GAM_B_GAIN_23[6]	GAM_B_GAIN_23[5]	GAM_B_GAIN_23[4]	GAM_B_GAIN_23[3]	GAM_B_GAIN_23[2]	GAM_B_GAIN_23[1]	GAM_B_GAIN_23[0]	
	GAM_B_LUT13	-	-	-	-	-	-	GAM_B_GAIN_24[10]	GAM_B_GAIN_24[9]	GAM_B_GAIN_24[8]
		GAM_B_GAIN_24[7]	GAM_B_GAIN_24[6]	GAM_B_GAIN_24[5]	GAM_B_GAIN_24[4]	GAM_B_GAIN_24[3]	GAM_B_GAIN_24[2]	GAM_B_GAIN_24[1]	GAM_B_GAIN_24[0]	
		-	-	-	-	-	-	GAM_B_GAIN_25[10]	GAM_B_GAIN_25[9]	GAM_B_GAIN_25[8]
		GAM_B_GAIN_25[7]	GAM_B_GAIN_25[6]	GAM_B_GAIN_25[5]	GAM_B_GAIN_25[4]	GAM_B_GAIN_25[3]	GAM_B_GAIN_25[2]	GAM_B_GAIN_25[1]	GAM_B_GAIN_25[0]	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Video display controller 5 channel 1	GAM_B_LUT14	-	-	-	-	-	GAM_B_GAIN_26[10]	GAM_B_GAIN_26[9]	GAM_B_GAIN_26[8]
		GAM_B_GAIN_26[7]	GAM_B_GAIN_26[6]	GAM_B_GAIN_26[5]	GAM_B_GAIN_26[4]	GAM_B_GAIN_26[3]	GAM_B_GAIN_26[2]	GAM_B_GAIN_26[1]	GAM_B_GAIN_26[0]
		-	-	-	-	-	GAM_B_GAIN_27[10]	GAM_B_GAIN_27[9]	GAM_B_GAIN_27[8]
		GAM_B_GAIN_27[7]	GAM_B_GAIN_27[6]	GAM_B_GAIN_27[5]	GAM_B_GAIN_27[4]	GAM_B_GAIN_27[3]	GAM_B_GAIN_27[2]	GAM_B_GAIN_27[1]	GAM_B_GAIN_27[0]
	GAM_B_LUT15	-	-	-	-	-	GAM_B_GAIN_28[10]	GAM_B_GAIN_28[9]	GAM_B_GAIN_28[8]
		GAM_B_GAIN_28[7]	GAM_B_GAIN_28[6]	GAM_B_GAIN_28[5]	GAM_B_GAIN_28[4]	GAM_B_GAIN_28[3]	GAM_B_GAIN_28[2]	GAM_B_GAIN_28[1]	GAM_B_GAIN_28[0]
		-	-	-	-	-	GAM_B_GAIN_29[10]	GAM_B_GAIN_29[9]	GAM_B_GAIN_29[8]
		GAM_B_GAIN_29[7]	GAM_B_GAIN_29[6]	GAM_B_GAIN_29[5]	GAM_B_GAIN_29[4]	GAM_B_GAIN_29[3]	GAM_B_GAIN_29[2]	GAM_B_GAIN_29[1]	GAM_B_GAIN_29[0]
	GAM_B_LUT16	-	-	-	-	-	GAM_B_GAIN_30[10]	GAM_B_GAIN_30[9]	GAM_B_GAIN_30[8]
		GAM_B_GAIN_30[7]	GAM_B_GAIN_30[6]	GAM_B_GAIN_30[5]	GAM_B_GAIN_30[4]	GAM_B_GAIN_30[3]	GAM_B_GAIN_30[2]	GAM_B_GAIN_30[1]	GAM_B_GAIN_30[0]
		-	-	-	-	-	GAM_B_GAIN_31[10]	GAM_B_GAIN_31[9]	GAM_B_GAIN_31[8]
		GAM_B_GAIN_31[7]	GAM_B_GAIN_31[6]	GAM_B_GAIN_31[5]	GAM_B_GAIN_31[4]	GAM_B_GAIN_31[3]	GAM_B_GAIN_31[2]	GAM_B_GAIN_31[1]	GAM_B_GAIN_31[0]
	GAM_B_AREA1	-	-	-	-	-	-	-	-
		GAM_B_TH_01[7]	GAM_B_TH_01[6]	GAM_B_TH_01[5]	GAM_B_TH_01[4]	GAM_B_TH_01[3]	GAM_B_TH_01[2]	GAM_B_TH_01[1]	GAM_B_TH_01[0]
		GAM_B_TH_02[7]	GAM_B_TH_02[6]	GAM_B_TH_02[5]	GAM_B_TH_02[4]	GAM_B_TH_02[3]	GAM_B_TH_02[2]	GAM_B_TH_02[1]	GAM_B_TH_02[0]
		GAM_B_TH_03[7]	GAM_B_TH_03[6]	GAM_B_TH_03[5]	GAM_B_TH_03[4]	GAM_B_TH_03[3]	GAM_B_TH_03[2]	GAM_B_TH_03[1]	GAM_B_TH_03[0]
	GAM_B_AREA2	GAM_B_TH_04[7]	GAM_B_TH_04[6]	GAM_B_TH_04[5]	GAM_B_TH_04[4]	GAM_B_TH_04[3]	GAM_B_TH_04[2]	GAM_B_TH_04[1]	GAM_B_TH_04[0]
		GAM_B_TH_05[7]	GAM_B_TH_05[6]	GAM_B_TH_05[5]	GAM_B_TH_05[4]	GAM_B_TH_05[3]	GAM_B_TH_05[2]	GAM_B_TH_05[1]	GAM_B_TH_05[0]
		GAM_B_TH_06[7]	GAM_B_TH_06[6]	GAM_B_TH_06[5]	GAM_B_TH_06[4]	GAM_B_TH_06[3]	GAM_B_TH_06[2]	GAM_B_TH_06[1]	GAM_B_TH_06[0]
		GAM_B_TH_07[7]	GAM_B_TH_07[6]	GAM_B_TH_07[5]	GAM_B_TH_07[4]	GAM_B_TH_07[3]	GAM_B_TH_07[2]	GAM_B_TH_07[1]	GAM_B_TH_07[0]
	GAM_B_AREA3	GAM_B_TH_08[7]	GAM_B_TH_08[6]	GAM_B_TH_08[5]	GAM_B_TH_08[4]	GAM_B_TH_08[3]	GAM_B_TH_08[2]	GAM_B_TH_08[1]	GAM_B_TH_08[0]
		GAM_B_TH_09[7]	GAM_B_TH_09[6]	GAM_B_TH_09[5]	GAM_B_TH_09[4]	GAM_B_TH_09[3]	GAM_B_TH_09[2]	GAM_B_TH_09[1]	GAM_B_TH_09[0]
		GAM_B_TH_10[7]	GAM_B_TH_10[6]	GAM_B_TH_10[5]	GAM_B_TH_10[4]	GAM_B_TH_10[3]	GAM_B_TH_10[2]	GAM_B_TH_10[1]	GAM_B_TH_10[0]
		GAM_B_TH_11[7]	GAM_B_TH_11[6]	GAM_B_TH_11[5]	GAM_B_TH_11[4]	GAM_B_TH_11[3]	GAM_B_TH_11[2]	GAM_B_TH_11[1]	GAM_B_TH_11[0]
	GAM_B_AREA4	GAM_B_TH_12[7]	GAM_B_TH_12[6]	GAM_B_TH_12[5]	GAM_B_TH_12[4]	GAM_B_TH_12[3]	GAM_B_TH_12[2]	GAM_B_TH_12[1]	GAM_B_TH_12[0]
		GAM_B_TH_13[7]	GAM_B_TH_13[6]	GAM_B_TH_13[5]	GAM_B_TH_13[4]	GAM_B_TH_13[3]	GAM_B_TH_13[2]	GAM_B_TH_13[1]	GAM_B_TH_13[0]
		GAM_B_TH_14[7]	GAM_B_TH_14[6]	GAM_B_TH_14[5]	GAM_B_TH_14[4]	GAM_B_TH_14[3]	GAM_B_TH_14[2]	GAM_B_TH_14[1]	GAM_B_TH_14[0]
		GAM_B_TH_15[7]	GAM_B_TH_15[6]	GAM_B_TH_15[5]	GAM_B_TH_15[4]	GAM_B_TH_15[3]	GAM_B_TH_15[2]	GAM_B_TH_15[1]	GAM_B_TH_15[0]
	GAM_B_AREA5	GAM_B_TH_16[7]	GAM_B_TH_16[6]	GAM_B_TH_16[5]	GAM_B_TH_16[4]	GAM_B_TH_16[3]	GAM_B_TH_16[2]	GAM_B_TH_16[1]	GAM_B_TH_16[0]
		GAM_B_TH_17[7]	GAM_B_TH_17[6]	GAM_B_TH_17[5]	GAM_B_TH_17[4]	GAM_B_TH_17[3]	GAM_B_TH_17[2]	GAM_B_TH_17[1]	GAM_B_TH_17[0]
		GAM_B_TH_18[7]	GAM_B_TH_18[6]	GAM_B_TH_18[5]	GAM_B_TH_18[4]	GAM_B_TH_18[3]	GAM_B_TH_18[2]	GAM_B_TH_18[1]	GAM_B_TH_18[0]
		GAM_B_TH_19[7]	GAM_B_TH_19[6]	GAM_B_TH_19[5]	GAM_B_TH_19[4]	GAM_B_TH_19[3]	GAM_B_TH_19[2]	GAM_B_TH_19[1]	GAM_B_TH_19[0]
	GAM_B_AREA6	GAM_B_TH_20[7]	GAM_B_TH_20[6]	GAM_B_TH_20[5]	GAM_B_TH_20[4]	GAM_B_TH_20[3]	GAM_B_TH_20[2]	GAM_B_TH_20[1]	GAM_B_TH_20[0]
		GAM_B_TH_21[7]	GAM_B_TH_21[6]	GAM_B_TH_21[5]	GAM_B_TH_21[4]	GAM_B_TH_21[3]	GAM_B_TH_21[2]	GAM_B_TH_21[1]	GAM_B_TH_21[0]
		GAM_B_TH_22[7]	GAM_B_TH_22[6]	GAM_B_TH_22[5]	GAM_B_TH_22[4]	GAM_B_TH_22[3]	GAM_B_TH_22[2]	GAM_B_TH_22[1]	GAM_B_TH_22[0]
		GAM_B_TH_23[7]	GAM_B_TH_23[6]	GAM_B_TH_23[5]	GAM_B_TH_23[4]	GAM_B_TH_23[3]	GAM_B_TH_23[2]	GAM_B_TH_23[1]	GAM_B_TH_23[0]
	GAM_B_AREA7	GAM_B_TH_24[7]	GAM_B_TH_24[6]	GAM_B_TH_24[5]	GAM_B_TH_24[4]	GAM_B_TH_24[3]	GAM_B_TH_24[2]	GAM_B_TH_24[1]	GAM_B_TH_24[0]
		GAM_B_TH_25[7]	GAM_B_TH_25[6]	GAM_B_TH_25[5]	GAM_B_TH_25[4]	GAM_B_TH_25[3]	GAM_B_TH_25[2]	GAM_B_TH_25[1]	GAM_B_TH_25[0]
		GAM_B_TH_26[7]	GAM_B_TH_26[6]	GAM_B_TH_26[5]	GAM_B_TH_26[4]	GAM_B_TH_26[3]	GAM_B_TH_26[2]	GAM_B_TH_26[1]	GAM_B_TH_26[0]
		GAM_B_TH_27[7]	GAM_B_TH_27[6]	GAM_B_TH_27[5]	GAM_B_TH_27[4]	GAM_B_TH_27[3]	GAM_B_TH_27[2]	GAM_B_TH_27[1]	GAM_B_TH_27[0]
	GAM_B_AREA8	GAM_B_TH_28[7]	GAM_B_TH_28[6]	GAM_B_TH_28[5]	GAM_B_TH_28[4]	GAM_B_TH_28[3]	GAM_B_TH_28[2]	GAM_B_TH_28[1]	GAM_B_TH_28[0]
		GAM_B_TH_29[7]	GAM_B_TH_29[6]	GAM_B_TH_29[5]	GAM_B_TH_29[4]	GAM_B_TH_29[3]	GAM_B_TH_29[2]	GAM_B_TH_29[1]	GAM_B_TH_29[0]
		GAM_B_TH_30[7]	GAM_B_TH_30[6]	GAM_B_TH_30[5]	GAM_B_TH_30[4]	GAM_B_TH_30[3]	GAM_B_TH_30[2]	GAM_B_TH_30[1]	GAM_B_TH_30[0]
		GAM_B_TH_31[7]	GAM_B_TH_31[6]	GAM_B_TH_31[5]	GAM_B_TH_31[4]	GAM_B_TH_31[3]	GAM_B_TH_31[2]	GAM_B_TH_31[1]	GAM_B_TH_31[0]
	GAM_R_UPDATE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GAM_R_VEN



Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Video display controller 5 channel 1	GAM_R_LUT1	-	-	-	-	-	GAM_R_GAIN_00[10]	GAM_R_GAIN_00[9]	GAM_R_GAIN_00[8]
		GAM_R_GAIN_00[7]	GAM_R_GAIN_00[6]	GAM_R_GAIN_00[5]	GAM_R_GAIN_00[4]	GAM_R_GAIN_00[3]	GAM_R_GAIN_00[2]	GAM_R_GAIN_00[1]	GAM_R_GAIN_00[0]
		-	-	-	-	-	GAM_R_GAIN_01[10]	GAM_R_GAIN_01[9]	GAM_R_GAIN_01[8]
	GAM_R_LUT2	GAM_R_GAIN_01[7]	GAM_R_GAIN_01[6]	GAM_R_GAIN_01[5]	GAM_R_GAIN_01[4]	GAM_R_GAIN_01[3]	GAM_R_GAIN_01[2]	GAM_R_GAIN_01[1]	GAM_R_GAIN_01[0]
		-	-	-	-	-	GAM_R_GAIN_02[10]	GAM_R_GAIN_02[9]	GAM_R_GAIN_02[8]
		GAM_R_GAIN_02[7]	GAM_R_GAIN_02[6]	GAM_R_GAIN_02[5]	GAM_R_GAIN_02[4]	GAM_R_GAIN_02[3]	GAM_R_GAIN_02[2]	GAM_R_GAIN_02[1]	GAM_R_GAIN_02[0]
	GAM_R_LUT3	-	-	-	-	-	GAM_R_GAIN_03[10]	GAM_R_GAIN_03[9]	GAM_R_GAIN_03[8]
		GAM_R_GAIN_03[7]	GAM_R_GAIN_03[6]	GAM_R_GAIN_03[5]	GAM_R_GAIN_03[4]	GAM_R_GAIN_03[3]	GAM_R_GAIN_03[2]	GAM_R_GAIN_03[1]	GAM_R_GAIN_03[0]
		-	-	-	-	-	GAM_R_GAIN_04[10]	GAM_R_GAIN_04[9]	GAM_R_GAIN_04[8]
	GAM_R_LUT4	GAM_R_GAIN_04[7]	GAM_R_GAIN_04[6]	GAM_R_GAIN_04[5]	GAM_R_GAIN_04[4]	GAM_R_GAIN_04[3]	GAM_R_GAIN_04[2]	GAM_R_GAIN_04[1]	GAM_R_GAIN_04[0]
		-	-	-	-	-	GAM_R_GAIN_05[10]	GAM_R_GAIN_05[9]	GAM_R_GAIN_05[8]
		GAM_R_GAIN_05[7]	GAM_R_GAIN_05[6]	GAM_R_GAIN_05[5]	GAM_R_GAIN_05[4]	GAM_R_GAIN_05[3]	GAM_R_GAIN_05[2]	GAM_R_GAIN_05[1]	GAM_R_GAIN_05[0]
	GAM_R_LUT5	-	-	-	-	-	GAM_R_GAIN_06[10]	GAM_R_GAIN_06[9]	GAM_R_GAIN_06[8]
		GAM_R_GAIN_06[7]	GAM_R_GAIN_06[6]	GAM_R_GAIN_06[5]	GAM_R_GAIN_06[4]	GAM_R_GAIN_06[3]	GAM_R_GAIN_06[2]	GAM_R_GAIN_06[1]	GAM_R_GAIN_06[0]
		-	-	-	-	-	GAM_R_GAIN_07[10]	GAM_R_GAIN_07[9]	GAM_R_GAIN_07[8]
	GAM_R_LUT6	GAM_R_GAIN_07[7]	GAM_R_GAIN_07[6]	GAM_R_GAIN_07[5]	GAM_R_GAIN_07[4]	GAM_R_GAIN_07[3]	GAM_R_GAIN_07[2]	GAM_R_GAIN_07[1]	GAM_R_GAIN_07[0]
		-	-	-	-	-	GAM_R_GAIN_08[10]	GAM_R_GAIN_08[9]	GAM_R_GAIN_08[8]
		GAM_R_GAIN_08[7]	GAM_R_GAIN_08[6]	GAM_R_GAIN_08[5]	GAM_R_GAIN_08[4]	GAM_R_GAIN_08[3]	GAM_R_GAIN_08[2]	GAM_R_GAIN_08[1]	GAM_R_GAIN_08[0]
	GAM_R_LUT7	-	-	-	-	-	GAM_R_GAIN_09[10]	GAM_R_GAIN_09[9]	GAM_R_GAIN_09[8]
		GAM_R_GAIN_09[7]	GAM_R_GAIN_09[6]	GAM_R_GAIN_09[5]	GAM_R_GAIN_09[4]	GAM_R_GAIN_09[3]	GAM_R_GAIN_09[2]	GAM_R_GAIN_09[1]	GAM_R_GAIN_09[0]
		-	-	-	-	-	GAM_R_GAIN_10[10]	GAM_R_GAIN_10[9]	GAM_R_GAIN_10[8]
	GAM_R_LUT8	GAM_R_GAIN_10[7]	GAM_R_GAIN_10[6]	GAM_R_GAIN_10[5]	GAM_R_GAIN_10[4]	GAM_R_GAIN_10[3]	GAM_R_GAIN_10[2]	GAM_R_GAIN_10[1]	GAM_R_GAIN_10[0]
		-	-	-	-	-	GAM_R_GAIN_11[10]	GAM_R_GAIN_11[9]	GAM_R_GAIN_11[8]
		GAM_R_GAIN_11[7]	GAM_R_GAIN_11[6]	GAM_R_GAIN_11[5]	GAM_R_GAIN_11[4]	GAM_R_GAIN_11[3]	GAM_R_GAIN_11[2]	GAM_R_GAIN_11[1]	GAM_R_GAIN_11[0]
	GAM_R_LUT9	-	-	-	-	-	GAM_R_GAIN_12[10]	GAM_R_GAIN_12[9]	GAM_R_GAIN_12[8]
		GAM_R_GAIN_12[7]	GAM_R_GAIN_12[6]	GAM_R_GAIN_12[5]	GAM_R_GAIN_12[4]	GAM_R_GAIN_12[3]	GAM_R_GAIN_12[2]	GAM_R_GAIN_12[1]	GAM_R_GAIN_12[0]
		-	-	-	-	-	GAM_R_GAIN_13[10]	GAM_R_GAIN_13[9]	GAM_R_GAIN_13[8]
	GAM_R_LUT10	GAM_R_GAIN_13[7]	GAM_R_GAIN_13[6]	GAM_R_GAIN_13[5]	GAM_R_GAIN_13[4]	GAM_R_GAIN_13[3]	GAM_R_GAIN_13[2]	GAM_R_GAIN_13[1]	GAM_R_GAIN_13[0]
		-	-	-	-	-	GAM_R_GAIN_14[10]	GAM_R_GAIN_14[9]	GAM_R_GAIN_14[8]
		GAM_R_GAIN_14[7]	GAM_R_GAIN_14[6]	GAM_R_GAIN_14[5]	GAM_R_GAIN_14[4]	GAM_R_GAIN_14[3]	GAM_R_GAIN_14[2]	GAM_R_GAIN_14[1]	GAM_R_GAIN_14[0]
	GAM_R_LUT11	-	-	-	-	-	GAM_R_GAIN_15[10]	GAM_R_GAIN_15[9]	GAM_R_GAIN_15[8]
		GAM_R_GAIN_15[7]	GAM_R_GAIN_15[6]	GAM_R_GAIN_15[5]	GAM_R_GAIN_15[4]	GAM_R_GAIN_15[3]	GAM_R_GAIN_15[2]	GAM_R_GAIN_15[1]	GAM_R_GAIN_15[0]
		-	-	-	-	-	GAM_R_GAIN_16[10]	GAM_R_GAIN_16[9]	GAM_R_GAIN_16[8]
	GAM_R_LUT12	GAM_R_GAIN_16[7]	GAM_R_GAIN_16[6]	GAM_R_GAIN_16[5]	GAM_R_GAIN_16[4]	GAM_R_GAIN_16[3]	GAM_R_GAIN_16[2]	GAM_R_GAIN_16[1]	GAM_R_GAIN_16[0]
		-	-	-	-	-	GAM_R_GAIN_17[10]	GAM_R_GAIN_17[9]	GAM_R_GAIN_17[8]
		GAM_R_GAIN_17[7]	GAM_R_GAIN_17[6]	GAM_R_GAIN_17[5]	GAM_R_GAIN_17[4]	GAM_R_GAIN_17[3]	GAM_R_GAIN_17[2]	GAM_R_GAIN_17[1]	GAM_R_GAIN_17[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 1	GAM_R_LUT10	-	-	-	-	-	GAM_R_GAIN_18[10]	GAM_R_GAIN_18[9]	GAM_R_GAIN_18[8]	
		GAM_R_GAIN_18[7]	GAM_R_GAIN_18[6]	GAM_R_GAIN_18[5]	GAM_R_GAIN_18[4]	GAM_R_GAIN_18[3]	GAM_R_GAIN_18[2]	GAM_R_GAIN_18[1]	GAM_R_GAIN_18[0]	
		-	-	-	-	-	GAM_R_GAIN_19[10]	GAM_R_GAIN_19[9]	GAM_R_GAIN_19[8]	
	GAM_R_LUT11	GAM_R_GAIN_19[7]	GAM_R_GAIN_19[6]	GAM_R_GAIN_19[5]	GAM_R_GAIN_19[4]	GAM_R_GAIN_19[3]	GAM_R_GAIN_19[2]	GAM_R_GAIN_19[1]	GAM_R_GAIN_19[0]	
		-	-	-	-	-	GAM_R_GAIN_20[10]	GAM_R_GAIN_20[9]	GAM_R_GAIN_20[8]	
		GAM_R_GAIN_20[7]	GAM_R_GAIN_20[6]	GAM_R_GAIN_20[5]	GAM_R_GAIN_20[4]	GAM_R_GAIN_20[3]	GAM_R_GAIN_20[2]	GAM_R_GAIN_20[1]	GAM_R_GAIN_20[0]	
	GAM_R_LUT12	-	-	-	-	-	GAM_R_GAIN_21[10]	GAM_R_GAIN_21[9]	GAM_R_GAIN_21[8]	
		GAM_R_GAIN_21[7]	GAM_R_GAIN_21[6]	GAM_R_GAIN_21[5]	GAM_R_GAIN_21[4]	GAM_R_GAIN_21[3]	GAM_R_GAIN_21[2]	GAM_R_GAIN_21[1]	GAM_R_GAIN_21[0]	
		-	-	-	-	-	GAM_R_GAIN_22[10]	GAM_R_GAIN_22[9]	GAM_R_GAIN_22[8]	
	GAM_R_LUT13	GAM_R_GAIN_22[7]	GAM_R_GAIN_22[6]	GAM_R_GAIN_22[5]	GAM_R_GAIN_22[4]	GAM_R_GAIN_22[3]	GAM_R_GAIN_22[2]	GAM_R_GAIN_22[1]	GAM_R_GAIN_22[0]	
		-	-	-	-	-	GAM_R_GAIN_23[10]	GAM_R_GAIN_23[9]	GAM_R_GAIN_23[8]	
		GAM_R_GAIN_23[7]	GAM_R_GAIN_23[6]	GAM_R_GAIN_23[5]	GAM_R_GAIN_23[4]	GAM_R_GAIN_23[3]	GAM_R_GAIN_23[2]	GAM_R_GAIN_23[1]	GAM_R_GAIN_23[0]	
	GAM_R_LUT14	-	-	-	-	-	GAM_R_GAIN_24[10]	GAM_R_GAIN_24[9]	GAM_R_GAIN_24[8]	
		GAM_R_GAIN_24[7]	GAM_R_GAIN_24[6]	GAM_R_GAIN_24[5]	GAM_R_GAIN_24[4]	GAM_R_GAIN_24[3]	GAM_R_GAIN_24[2]	GAM_R_GAIN_24[1]	GAM_R_GAIN_24[0]	
		-	-	-	-	-	GAM_R_GAIN_25[10]	GAM_R_GAIN_25[9]	GAM_R_GAIN_25[8]	
	GAM_R_LUT15	GAM_R_GAIN_25[7]	GAM_R_GAIN_25[6]	GAM_R_GAIN_25[5]	GAM_R_GAIN_25[4]	GAM_R_GAIN_25[3]	GAM_R_GAIN_25[2]	GAM_R_GAIN_25[1]	GAM_R_GAIN_25[0]	
		-	-	-	-	-	GAM_R_GAIN_26[10]	GAM_R_GAIN_26[9]	GAM_R_GAIN_26[8]	
		GAM_R_GAIN_26[7]	GAM_R_GAIN_26[6]	GAM_R_GAIN_26[5]	GAM_R_GAIN_26[4]	GAM_R_GAIN_26[3]	GAM_R_GAIN_26[2]	GAM_R_GAIN_26[1]	GAM_R_GAIN_26[0]	
	GAM_R_LUT16	-	-	-	-	-	GAM_R_GAIN_27[10]	GAM_R_GAIN_27[9]	GAM_R_GAIN_27[8]	
		GAM_R_GAIN_27[7]	GAM_R_GAIN_27[6]	GAM_R_GAIN_27[5]	GAM_R_GAIN_27[4]	GAM_R_GAIN_27[3]	GAM_R_GAIN_27[2]	GAM_R_GAIN_27[1]	GAM_R_GAIN_27[0]	
		-	-	-	-	-	GAM_R_GAIN_28[10]	GAM_R_GAIN_28[9]	GAM_R_GAIN_28[8]	
	GAM_R_AREA1	GAM_R_GAIN_28[7]	GAM_R_GAIN_28[6]	GAM_R_GAIN_28[5]	GAM_R_GAIN_28[4]	GAM_R_GAIN_28[3]	GAM_R_GAIN_28[2]	GAM_R_GAIN_28[1]	GAM_R_GAIN_28[0]	
		-	-	-	-	-	GAM_R_GAIN_29[10]	GAM_R_GAIN_29[9]	GAM_R_GAIN_29[8]	
		GAM_R_GAIN_29[7]	GAM_R_GAIN_29[6]	GAM_R_GAIN_29[5]	GAM_R_GAIN_29[4]	GAM_R_GAIN_29[3]	GAM_R_GAIN_29[2]	GAM_R_GAIN_29[1]	GAM_R_GAIN_29[0]	
	GAM_R_AREA2	-	-	-	-	-	GAM_R_GAIN_30[10]	GAM_R_GAIN_30[9]	GAM_R_GAIN_30[8]	
		GAM_R_GAIN_30[7]	GAM_R_GAIN_30[6]	GAM_R_GAIN_30[5]	GAM_R_GAIN_30[4]	GAM_R_GAIN_30[3]	GAM_R_GAIN_30[2]	GAM_R_GAIN_30[1]	GAM_R_GAIN_30[0]	
		-	-	-	-	-	GAM_R_GAIN_31[10]	GAM_R_GAIN_31[9]	GAM_R_GAIN_31[8]	
	GAM_R_AREA3	GAM_R_GAIN_31[7]	GAM_R_GAIN_31[6]	GAM_R_GAIN_31[5]	GAM_R_GAIN_31[4]	GAM_R_GAIN_31[3]	GAM_R_GAIN_31[2]	GAM_R_GAIN_31[1]	GAM_R_GAIN_31[0]	
		-	-	-	-	-	-	-	-	
		GAM_R_TH_01[7]	GAM_R_TH_01[6]	GAM_R_TH_01[5]	GAM_R_TH_01[4]	GAM_R_TH_01[3]	GAM_R_TH_01[2]	GAM_R_TH_01[1]	GAM_R_TH_01[0]	
	GAM_R_AREA1	GAM_R_TH_02[7]	GAM_R_TH_02[6]	GAM_R_TH_02[5]	GAM_R_TH_02[4]	GAM_R_TH_02[3]	GAM_R_TH_02[2]	GAM_R_TH_02[1]	GAM_R_TH_02[0]	
		GAM_R_TH_03[7]	GAM_R_TH_03[6]	GAM_R_TH_03[5]	GAM_R_TH_03[4]	GAM_R_TH_03[3]	GAM_R_TH_03[2]	GAM_R_TH_03[1]	GAM_R_TH_03[0]	
		GAM_R_TH_04[7]	GAM_R_TH_04[6]	GAM_R_TH_04[5]	GAM_R_TH_04[4]	GAM_R_TH_04[3]	GAM_R_TH_04[2]	GAM_R_TH_04[1]	GAM_R_TH_04[0]	
	GAM_R_AREA2	GAM_R_TH_05[7]	GAM_R_TH_05[6]	GAM_R_TH_05[5]	GAM_R_TH_05[4]	GAM_R_TH_05[3]	GAM_R_TH_05[2]	GAM_R_TH_05[1]	GAM_R_TH_05[0]	
		GAM_R_TH_06[7]	GAM_R_TH_06[6]	GAM_R_TH_06[5]	GAM_R_TH_06[4]	GAM_R_TH_06[3]	GAM_R_TH_06[2]	GAM_R_TH_06[1]	GAM_R_TH_06[0]	
		GAM_R_TH_07[7]	GAM_R_TH_07[6]	GAM_R_TH_07[5]	GAM_R_TH_07[4]	GAM_R_TH_07[3]	GAM_R_TH_07[2]	GAM_R_TH_07[1]	GAM_R_TH_07[0]	
	GAM_R_AREA3	GAM_R_TH_08[7]	GAM_R_TH_08[6]	GAM_R_TH_08[5]	GAM_R_TH_08[4]	GAM_R_TH_08[3]	GAM_R_TH_08[2]	GAM_R_TH_08[1]	GAM_R_TH_08[0]	
		GAM_R_TH_09[7]	GAM_R_TH_09[6]	GAM_R_TH_09[5]	GAM_R_TH_09[4]	GAM_R_TH_09[3]	GAM_R_TH_09[2]	GAM_R_TH_09[1]	GAM_R_TH_09[0]	
		GAM_R_TH_10[7]	GAM_R_TH_10[6]	GAM_R_TH_10[5]	GAM_R_TH_10[4]	GAM_R_TH_10[3]	GAM_R_TH_10[2]	GAM_R_TH_10[1]	GAM_R_TH_10[0]	
			GAM_R_TH_11[7]	GAM_R_TH_11[6]	GAM_R_TH_11[5]	GAM_R_TH_11[4]	GAM_R_TH_11[3]	GAM_R_TH_11[2]	GAM_R_TH_11[1]	GAM_R_TH_11[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Video display controller 5 channel 1	GAM_R_AREA4	GAM_R_TH_12[7]	GAM_R_TH_12[6]	GAM_R_TH_12[5]	GAM_R_TH_12[4]	GAM_R_TH_12[3]	GAM_R_TH_12[2]	GAM_R_TH_12[1]	GAM_R_TH_12[0]
		GAM_R_TH_13[7]	GAM_R_TH_13[6]	GAM_R_TH_13[5]	GAM_R_TH_13[4]	GAM_R_TH_13[3]	GAM_R_TH_13[2]	GAM_R_TH_13[1]	GAM_R_TH_13[0]
		GAM_R_TH_14[7]	GAM_R_TH_14[6]	GAM_R_TH_14[5]	GAM_R_TH_14[4]	GAM_R_TH_14[3]	GAM_R_TH_14[2]	GAM_R_TH_14[1]	GAM_R_TH_14[0]
		GAM_R_TH_15[7]	GAM_R_TH_15[6]	GAM_R_TH_15[5]	GAM_R_TH_15[4]	GAM_R_TH_15[3]	GAM_R_TH_15[2]	GAM_R_TH_15[1]	GAM_R_TH_15[0]
	GAM_R_AREA5	GAM_R_TH_16[7]	GAM_R_TH_16[6]	GAM_R_TH_16[5]	GAM_R_TH_16[4]	GAM_R_TH_16[3]	GAM_R_TH_16[2]	GAM_R_TH_16[1]	GAM_R_TH_16[0]
		GAM_R_TH_17[7]	GAM_R_TH_17[6]	GAM_R_TH_17[5]	GAM_R_TH_17[4]	GAM_R_TH_17[3]	GAM_R_TH_17[2]	GAM_R_TH_17[1]	GAM_R_TH_17[0]
		GAM_R_TH_18[7]	GAM_R_TH_18[6]	GAM_R_TH_18[5]	GAM_R_TH_18[4]	GAM_R_TH_18[3]	GAM_R_TH_18[2]	GAM_R_TH_18[1]	GAM_R_TH_18[0]
		GAM_R_TH_19[7]	GAM_R_TH_19[6]	GAM_R_TH_19[5]	GAM_R_TH_19[4]	GAM_R_TH_19[3]	GAM_R_TH_19[2]	GAM_R_TH_19[1]	GAM_R_TH_19[0]
	GAM_R_AREA6	GAM_R_TH_20[7]	GAM_R_TH_20[6]	GAM_R_TH_20[5]	GAM_R_TH_20[4]	GAM_R_TH_20[3]	GAM_R_TH_20[2]	GAM_R_TH_20[1]	GAM_R_TH_20[0]
		GAM_R_TH_21[7]	GAM_R_TH_21[6]	GAM_R_TH_21[5]	GAM_R_TH_21[4]	GAM_R_TH_21[3]	GAM_R_TH_21[2]	GAM_R_TH_21[1]	GAM_R_TH_21[0]
		GAM_R_TH_22[7]	GAM_R_TH_22[6]	GAM_R_TH_22[5]	GAM_R_TH_22[4]	GAM_R_TH_22[3]	GAM_R_TH_22[2]	GAM_R_TH_22[1]	GAM_R_TH_22[0]
		GAM_R_TH_23[7]	GAM_R_TH_23[6]	GAM_R_TH_23[5]	GAM_R_TH_23[4]	GAM_R_TH_23[3]	GAM_R_TH_23[2]	GAM_R_TH_23[1]	GAM_R_TH_23[0]
	GAM_R_AREA7	GAM_R_TH_24[7]	GAM_R_TH_24[6]	GAM_R_TH_24[5]	GAM_R_TH_24[4]	GAM_R_TH_24[3]	GAM_R_TH_24[2]	GAM_R_TH_24[1]	GAM_R_TH_24[0]
		GAM_R_TH_25[7]	GAM_R_TH_25[6]	GAM_R_TH_25[5]	GAM_R_TH_25[4]	GAM_R_TH_25[3]	GAM_R_TH_25[2]	GAM_R_TH_25[1]	GAM_R_TH_25[0]
		GAM_R_TH_26[7]	GAM_R_TH_26[6]	GAM_R_TH_26[5]	GAM_R_TH_26[4]	GAM_R_TH_26[3]	GAM_R_TH_26[2]	GAM_R_TH_26[1]	GAM_R_TH_26[0]
		GAM_R_TH_27[7]	GAM_R_TH_27[6]	GAM_R_TH_27[5]	GAM_R_TH_27[4]	GAM_R_TH_27[3]	GAM_R_TH_27[2]	GAM_R_TH_27[1]	GAM_R_TH_27[0]
	GAM_R_AREA8	GAM_R_TH_28[7]	GAM_R_TH_28[6]	GAM_R_TH_28[5]	GAM_R_TH_28[4]	GAM_R_TH_28[3]	GAM_R_TH_28[2]	GAM_R_TH_28[1]	GAM_R_TH_28[0]
		GAM_R_TH_29[7]	GAM_R_TH_29[6]	GAM_R_TH_29[5]	GAM_R_TH_29[4]	GAM_R_TH_29[3]	GAM_R_TH_29[2]	GAM_R_TH_29[1]	GAM_R_TH_29[0]
		GAM_R_TH_30[7]	GAM_R_TH_30[6]	GAM_R_TH_30[5]	GAM_R_TH_30[4]	GAM_R_TH_30[3]	GAM_R_TH_30[2]	GAM_R_TH_30[1]	GAM_R_TH_30[0]
		GAM_R_TH_31[7]	GAM_R_TH_31[6]	GAM_R_TH_31[5]	GAM_R_TH_31[4]	GAM_R_TH_31[3]	GAM_R_TH_31[2]	GAM_R_TH_31[1]	GAM_R_TH_31[0]
	TCON_UPDATE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	TCON_VEN
	TCON_TIM	-	-	-	-	-	TCON_HALF[10]	TCON_HALF[9]	TCON_HALF[8]
		TCON_HALF[7]	TCON_HALF[6]	TCON_HALF[5]	TCON_HALF[4]	TCON_HALF[3]	TCON_HALF[2]	TCON_HALF[1]	TCON_HALF[0]
		-	-	-	-	-	TCON_OFFSET[10]	TCON_OFFSET[9]	TCON_OFFSET[8]
		TCON_OFFSET[7]	TCON_OFFSET[6]	TCON_OFFSET[5]	TCON_OFFSET[4]	TCON_OFFSET[3]	TCON_OFFSET[2]	TCON_OFFSET[1]	TCON_OFFSET[0]
	TCON_TIM_STVA1	-	-	-	-	-	TCON_STVA_VS[10]	TCON_STVA_VS[9]	TCON_STVA_VS[8]
		TCON_STVA_VS[7]	TCON_STVA_VS[6]	TCON_STVA_VS[5]	TCON_STVA_VS[4]	TCON_STVA_VS[3]	TCON_STVA_VS[2]	TCON_STVA_VS[1]	TCON_STVA_VS[0]
		-	-	-	-	-	TCON_STVA_VW[10]	TCON_STVA_VW[9]	TCON_STVA_VW[8]
		TCON_STVA_VW[7]	TCON_STVA_VW[6]	TCON_STVA_VW[5]	TCON_STVA_VW[4]	TCON_STVA_VW[3]	TCON_STVA_VW[2]	TCON_STVA_VW[1]	TCON_STVA_VW[0]
	TCON_TIM_STVA2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	TCON_STVA_INV	-	TCON_STVA_SEL[2]	TCON_STVA_SEL[1]	TCON_STVA_SEL[0]
	TCON_TIM_STVB1	-	-	-	-	-	TCON_STVB_VS[10]	TCON_STVB_VS[9]	TCON_STVB_VS[8]
		TCON_STVB_VS[7]	TCON_STVB_VS[6]	TCON_STVB_VS[5]	TCON_STVB_VS[4]	TCON_STVB_VS[3]	TCON_STVB_VS[2]	TCON_STVB_VS[1]	TCON_STVB_VS[0]
		-	-	-	-	-	TCON_STVB_VW[10]	TCON_STVB_VW[9]	TCON_STVB_VW[8]
		TCON_STVB_VW[7]	TCON_STVB_VW[6]	TCON_STVB_VW[5]	TCON_STVB_VW[4]	TCON_STVB_VW[3]	TCON_STVB_VW[2]	TCON_STVB_VW[1]	TCON_STVB_VW[0]
	TCON_TIM_STVB2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	TCON_STVB_INV	-	TCON_STVB_SEL[2]	TCON_STVB_SEL[1]	TCON_STVB_SEL[0]
	TCON_TIM_STH1	-	-	-	-	-	TCON_STH_HS[10]	TCON_STH_HS[9]	TCON_STH_HS[8]
		TCON_STH_HS[7]	TCON_STH_HS[6]	TCON_STH_HS[5]	TCON_STH_HS[4]	TCON_STH_HS[3]	TCON_STH_HS[2]	TCON_STH_HS[1]	TCON_STH_HS[0]
		-	-	-	-	-	TCON_STH_HW[10]	TCON_STH_HW[9]	TCON_STH_HW[8]
		TCON_STH_HW[7]	TCON_STH_HW[6]	TCON_STH_HW[5]	TCON_STH_HW[4]	TCON_STH_HW[3]	TCON_STH_HW[2]	TCON_STH_HW[1]	TCON_STH_HW[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Video display controller 5 channel 1	TCON_TIM_STH2	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	TCON_STH_HS_SEL	
		-	-	-	TCON_STH_INV	-	TCON_STH_SEL [2]	TCON_STH_SEL [1]	TCON_STH_SEL [0]	
	TCON_TIM_STB1	-	-	-	-	-	-	TCON_STB_HS [10]	TCON_STB_HS[9]	TCON_STB_HS[8]
		TCON_STB_HS[7]	TCON_STB_HS[6]	TCON_STB_HS[5]	TCON_STB_HS[4]	TCON_STB_HS[3]	TCON_STB_HS[2]	TCON_STB_HS[1]	TCON_STB_HS[0]	
		-	-	-	-	-	TCON_STB_HW [10]	TCON_STB_HW [9]	TCON_STB_HW [8]	
		TCON_STB_HW [7]	TCON_STB_HW [6]	TCON_STB_HW [5]	TCON_STB_HW [4]	TCON_STB_HW [3]	TCON_STB_HW [2]	TCON_STB_HW [1]	TCON_STB_HW [0]	
	TCON_TIM_STB2	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	TCON_STB_HS_SEL	
		-	-	-	TCON_STB_INV	-	TCON_STB_SEL [2]	TCON_STB_SEL [1]	TCON_STB_SEL [0]	
	TCON_TIM_CPV1	-	-	-	-	-	-	TCON_CPV_HS [10]	TCON_CPV_HS [9]	TCON_CPV_HS [8]
		TCON_CPV_HS[7]	TCON_CPV_HS[6]	TCON_CPV_HS[5]	TCON_CPV_HS[4]	TCON_CPV_HS[3]	TCON_CPV_HS[2]	TCON_CPV_HS[1]	TCON_CPV_HS[0]	
		-	-	-	-	-	TCON_CPV_HW [10]	TCON_CPV_HW [9]	TCON_CPV_HW [8]	
		TCON_CPV_HW [7]	TCON_CPV_HW [6]	TCON_CPV_HW [5]	TCON_CPV_HW [4]	TCON_CPV_HW [3]	TCON_CPV_HW [2]	TCON_CPV_HW [1]	TCON_CPV_HW [0]	
	TCON_TIM_CPV2	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	TCON_CPV_HS_SEL	
		-	-	-	TCON_CPV_INV	-	TCON_CPV_SEL [2]	TCON_CPV_SEL [1]	TCON_CPV_SEL [0]	
	TCON_TIM_POLA1	-	-	-	-	-	-	TCON_POLA_HS[10]	TCON_POLA_HS[9]	TCON_POLA_HS[8]
		TCON_POLA_HS[7]	TCON_POLA_HS[6]	TCON_POLA_HS[5]	TCON_POLA_HS[4]	TCON_POLA_HS[3]	TCON_POLA_HS[2]	TCON_POLA_HS[1]	TCON_POLA_HS[0]	
		-	-	-	-	-	TCON_POLA_HW[10]	TCON_POLA_HW[9]	TCON_POLA_HW[8]	
		TCON_POLA_HW[7]	TCON_POLA_HW[6]	TCON_POLA_HW[5]	TCON_POLA_HW[4]	TCON_POLA_HW[3]	TCON_POLA_HW[2]	TCON_POLA_HW[1]	TCON_POLA_HW[0]	
	TCON_TIM_POLA2	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	TCON_POLA_MD[1]	TCON_POLA_MD[0]	-	-	-	TCON_POLA_HS_SEL	
		-	-	-	TCON_POLA_INV	-	TCON_POLA_SEL[2]	TCON_POLA_SEL[1]	TCON_POLA_SEL[0]	
	TCON_TIM_POLB1	-	-	-	-	-	-	TCON_POLB_HS[10]	TCON_POLB_HS[9]	TCON_POLB_HS[8]
		TCON_POLB_HS[7]	TCON_POLB_HS[6]	TCON_POLB_HS[5]	TCON_POLB_HS[4]	TCON_POLB_HS[3]	TCON_POLB_HS[2]	TCON_POLB_HS[1]	TCON_POLB_HS[0]	
		-	-	-	-	-	TCON_POLB_HW[10]	TCON_POLB_HW[9]	TCON_POLB_HW[8]	
		TCON_POLB_HW[7]	TCON_POLB_HW[6]	TCON_POLB_HW[5]	TCON_POLB_HW[4]	TCON_POLB_HW[3]	TCON_POLB_HW[2]	TCON_POLB_HW[1]	TCON_POLB_HW[0]	
	TCON_TIM_POLB2	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	TCON_POLB_MD[1]	TCON_POLB_MD[0]	-	-	-	TCON_POLB_HS_SEL	
		-	-	-	TCON_POLB_INV	-	TCON_POLB_SEL[2]	TCON_POLB_SEL[1]	TCON_POLB_SEL[0]	
	TCON_TIM_DE	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	TCON_DE_INV	
	OUT_UPDATE	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	OUTCNT_VEN	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Video display controller 5 channel 1	OUT_SET	-	-	-	OUT_ENDIAN_ON	-	-	-	OUT_SWAP_ON
		-	-	-	-	-	-	-	-
		-	-	OUT_FORMAT[1]	OUT_FORMAT[0]	-	-	OUT_FRQ_SEL[1]	OUT_FRQ_SEL[0]
		-	-	-	OUT_DIR_SEL	-	-	OUT_PHASE[1]	OUT_PHASE[0]
	OUT_BRIGHT1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		PBRT_G[7]	PBRT_G[6]	PBRT_G[5]	PBRT_G[4]	PBRT_G[3]	PBRT_G[2]	PBRT_G[1]	PBRT_G[0]
	OUT_BRIGHT2	-	-	-	-	-	-	-	-
		PBRT_B[7]	PBRT_B[6]	PBRT_B[5]	PBRT_B[4]	PBRT_B[3]	PBRT_B[2]	PBRT_B[1]	PBRT_B[0]
		-	-	-	-	-	-	PBRT_R[9]	PBRT_R[8]
	OUT_CONTRAST	-	-	-	-	-	-	-	-
		CONT_G[7]	CONT_G[6]	CONT_G[5]	CONT_G[4]	CONT_G[3]	CONT_G[2]	CONT_G[1]	CONT_G[0]
		CONT_B[7]	CONT_B[6]	CONT_B[5]	CONT_B[4]	CONT_B[3]	CONT_B[2]	CONT_B[1]	CONT_B[0]
		CONT_R[7]	CONT_R[6]	CONT_R[5]	CONT_R[4]	CONT_R[3]	CONT_R[2]	CONT_R[1]	CONT_R[0]
	OUT_PDTHA	-	-	-	-	-	-	-	-
		-	-	PDTH_SEL[1]	PDTH_SEL[0]	-	-	PDTH_FORMAT[1]	PDTH_FORMAT[0]
		-	-	PDTH_PA[1]	PDTH_PA[0]	-	-	PDTH_PB[1]	PDTH_PB[0]
		-	-	PDTH_PC[1]	PDTH_PC[0]	-	-	PDTH_PD[1]	PDTH_PD[0]
	OUT_CLK_PHASE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	OUTCNT_FRONT_GAM	-	-	-	OUTCNT_LCD_EDGE
		-	OUTCNT_STVA_EDGE	OUTCNT_STVB_EDGE	OUTCNT_STH_EDGE	OUTCNT_STB_EDGE	OUTCNT_CPV_EDGE	OUTCNT_POLA_EDGE	OUTCNT_POLB_EDGE
	SYSCNT_INT1	-	-	-	INT_STA7	-	-	-	INT_STA6
		-	-	-	INT_STA5	-	-	-	INT_STA4
		-	-	-	INT_STA3	-	-	-	INT_STA2
		-	-	-	INT_STA1	-	-	-	INT_STA0
	SYSCNT_INT2	-	-	-	INT_STA15	-	-	-	INT_STA14
		-	-	-	INT_STA13	-	-	-	INT_STA12
		-	-	-	INT_STA11	-	-	-	INT_STA10
		-	-	-	INT_STA9	-	-	-	INT_STA8
	SYSCNT_INT3	-	-	-	-	-	-	-	INT_STA22
		-	-	-	INT_STA21	-	-	-	INT_STA20
		-	-	-	INT_STA19	-	-	-	INT_STA18
		-	-	-	INT_STA17	-	-	-	INT_STA16
	SYSCNT_INT4	-	-	-	INT_OUT7_ON	-	-	-	INT_OUT6_ON
		-	-	-	INT_OUT5_ON	-	-	-	INT_OUT4_ON
		-	-	-	INT_OUT3_ON	-	-	-	INT_OUT2_ON
		-	-	-	INT_OUT1_ON	-	-	-	INT_OUT0_ON
	SYSCNT_INT5	-	-	-	INT_OUT15_ON	-	-	-	INT_OUT14_ON
		-	-	-	INT_OUT13_ON	-	-	-	INT_OUT12_ON
		-	-	-	INT_OUT11_ON	-	-	-	INT_OUT10_ON
		-	-	-	INT_OUT9_ON	-	-	-	INT_OUT8_ON
	SYSCNT_INT6	-	-	-	-	-	-	-	INT_OUT22_ON
		-	-	-	INT_OUT21_ON	-	-	-	INT_OUT20_ON
		-	-	-	INT_OUT19_ON	-	-	-	INT_OUT18_ON
		-	-	-	INT_OUT17_ON	-	-	-	INT_OUT16_ON
	SYSCNT_PANEL_CLK	-	-	PANEL_JCKSEL[1]	PANEL_JCKSEL[0]	PANEL_OCKSEL[1]	PANEL_OCKSEL[0]	-	PANEL_ICKEN
		-	-	PANEL_DCDR[5]	PANEL_DCDR[4]	PANEL_DCDR[3]	PANEL_DCDR[2]	PANEL_DCDR[1]	PANEL_DCDR[0]
	SYSCNT_CLUT	-	GR_OIR_CLT_SEL_ST	-	GR3_CLT_SEL_ST	-	-	-	GR2_CLT_SEL_ST
		-	-	-	GR1_CLT_SEL_ST	-	-	-	GR0_CLT_SEL_ST

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0		
LVDS output interface	LVDS_UPDATE	-	-	-	-	-	-	-	-		
		-	-	-	-	-	-	-	LVDS_UPDATE		
		-	-	-	-	-	-	-	-		
		-	-	-	-	-	-	-	-		
	LVDSFCL	-	-	-	-	SYNC_MODE	-	-	-	-	
		SYNC_POL[1]	SYNC_POL[0]	-	-	-	-	-	-	-	
		-	-	-	-	LVDS_SEL2[3]	LVDS_SEL2[2]	LVDS_SEL2[1]	LVDS_SEL2[0]		
	LVDS_SEL[3]	LVDS_SEL1[3]	LVDS_SEL1[2]	LVDS_SEL1[1]	LVDS_SEL1[0]	LVDS_SEL0[3]	LVDS_SEL0[2]	LVDS_SEL0[1]	LVDS_SEL0[0]		
		-	-	-	-	-	LVDS_IN_CLK_SEL[2]	LVDS_IN_CLK_SEL[1]	LVDS_IN_CLK_SEL[0]		
		-	-	-	-	-	-	LVDS_IDIV_SET[1]	LVDS_IDIV_SET[0]		
	LVDSPLL_TST[5]	LVDSPLL_TST[5]	LVDSPLL_TST[4]	LVDSPLL_TST[3]	LVDSPLL_TST[2]	LVDSPLL_TST[1]	LVDSPLL_TST[0]	LVDS_ODIV_SET[1]	LVDS_ODIV_SET[0]		
		-	-	-	LVDS_CLK_EN	-	-	LVDS_VDC_SEL	-		
		-	-	-	-	-	-	-	-		
	LVDSPLL_BP	LVDSPLL_BP	-	-	-	-	-	LVDSPLL_FD[10]	LVDSPLL_FD[9]	LVDSPLL_FD[8]	
		LVDSPLL_FD[7]	LVDSPLL_FD[6]	LVDSPLL_FD[5]	LVDSPLL_FD[4]	LVDSPLL_FD[3]	LVDSPLL_FD[2]	LVDSPLL_FD[1]	LVDSPLL_FD[0]		
		-	-	-	LVDSPLL_RD[4]	LVDSPLL_RD[3]	LVDSPLL_RD[2]	LVDSPLL_RD[1]	LVDSPLL_RD[0]		
		-	-	LVDSPLL_OD[1]	LVDSPLL_OD[0]	-	-	-	LVDSPLL_PD		
	LVDSPLL_OD[1]	-	-	-	-	-	-	-	-		
		-	-	-	-	-	-	-	-		
		-	-	-	-	-	-	-	-		
		-	-	-	-	-	-	SKEWC[1]	SKEWC[0]		
	Image renderer channel 0	CR	-	-	-	-	-	-	-	-	
			-	-	-	-	-	-	-	-	
			SWRST	-	-	-	-	-	-	-	
		SR	-	-	-	-	-	-	SFE	ARS	RS
			-	-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-	-
		SFS	SFS	DSA	REN	-	-	-	INT	IER	TRA
-			-	-	-	-	-	-	-	-	
-			-	-	-	-	-	-	-	-	
SRCR		-	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	INTCLR	IERCLR	TRACLR	
ICR		-	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	INTENB	IERENB	TRAENB	
IMR		-	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	INM	IEM	TRAM	
DLP		DLP[31]	DLP[30]	DLP[29]	DLP[28]	DLP[27]	DLP[26]	DLP[25]	DLP[24]		
		DLP[23]	DLP[22]	DLP[21]	DLP[20]	DLP[19]	DLP[18]	DLP[17]	DLP[16]		
		DLP[15]	DLP[14]	DLP[13]	DLP[12]	DLP[11]	DLP[10]	DLP[9]	DLP[8]		
		DLP[7]	DLP[6]	DLP[5]	DLP[4]	DLP[3]	DLP[2]	DLP[1]	DLP[0]		
DLSA		DLSA[28]	DLSA[27]	DLSA[26]	DLSA[25]	DLSA[24]	DLSA[23]	DLSA[22]	DLSA[21]		
		DLSA[20]	DLSA[19]	DLSA[18]	DLSA[17]	DLSA[16]	DLSA[15]	DLSA[14]	DLSA[13]		
		DLSA[12]	DLSA[11]	DLSA[10]	DLSA[9]	DLSA[8]	DLSA[7]	DLSA[6]	DLSA[5]		
		DLSA[4]	DLSA[3]	DLSA[2]	DLSA[1]	DLSA[0]	-	-	-		
DSA		DSA[26]	DSA[25]	DSA[24]	DSA[23]	DSA[22]	DSA[21]	DSA[20]	DSA[19]		
		DSA[18]	DSA[17]	DSA[16]	DSA[15]	DSA[14]	DSA[13]	DSA[12]	DSA[11]		
		DSA[10]	DSA[9]	DSA[8]	DSA[7]	DSA[6]	DSA[5]	DSA[4]	DSA[3]		
		DSA[2]	DSA[1]	DSA[0]	-	-	-	-	-		
DSTR		-	-	-	-	-	-	-	-		
		-	-	-	-	-	-	-	-		
		-	-	DST[13]	DST[12]	DST[11]	DST[10]	DST[9]	DST[8]		
		DST[7]	DST[6]	DST[5]	DST[4]	DST[3]	DST[2]	DST[1]	DST[0]		

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Image renderer channel 0	DSAR2	DSA2[26]	DSA2[25]	DSA2[24]	DSA2[23]	DSA2[22]	DSA2[21]	DSA2[20]	DSA2[19]	
		DSA2[18]	DSA2[17]	DSA2[16]	DSA2[15]	DSA2[14]	DSA2[13]	DSA2[12]	DSA2[11]	
		DSA2[10]	DSA2[9]	DSA2[8]	DSA2[7]	DSA2[6]	DSA2[5]	DSA2[4]	DSA2[3]	
		DSA2[2]	DSA2[1]	DSA2[0]	-	-	-	-	-	
	DLSAR2	DLSA2[28]	DLSA2[27]	DLSA2[26]	DLSA2[25]	DLSA2[24]	DLSA2[23]	DLSA2[22]	DLSA2[21]	
		DLSA2[20]	DLSA2[19]	DLSA2[18]	DLSA2[17]	DLSA2[16]	DLSA2[15]	DLSA2[14]	DLSA2[13]	
		DLSA2[12]	DLSA2[11]	DLSA2[10]	DLSA2[9]	DLSA2[8]	DLSA2[7]	DLSA2[6]	DLSA2[5]	
		DLSA2[4]	DLSA2[3]	DLSA2[2]	DLSA2[1]	DLSA2[0]	-	-	-	
	TRIMR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	TCM	DUDVM	DXDYM	AUTOSG	AUTODG	BFE	TME	
	TRIMSR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	TCMS	DUDVMS	DXDYMS	AUTOSGS	AUTODGS	BFES	TMES	
	TRIMCR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	TCMC	DUDVMC	DXDYMC	AUTOSGC	AUTODGC	BFEC	TMEC	
	TRICR	YCFORM	-	-	-	-	-	-	-	
		TCV[7]	TCV[6]	TCV[5]	TCV[4]	TCV[3]	TCV[2]	TCV[1]	TCV[0]	
		TCU[7]	TCU[6]	TCU[5]	TCU[4]	TCU[3]	TCU[2]	TCU[1]	TCU[0]	
		TCY[7]	TCY[6]	TCY[5]	TCY[4]	TCY[3]	TCY[2]	TCY[1]	TCY[0]	
	UVDPOR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	DDP	
		-	-	-	-	-	UVDPO[2]	UVDPO[1]	UVDPO[0]	
	SUSR	-	-	-	-	-	-	SUW[10]	SUW[9]	SUW[8]
		SUW[7]	SUW[6]	SUW[5]	SUW[4]	SUW[3]	SUW[2]	SUW[1]	SUW[0]	
		-	-	-	-	-	SVW[10]	SVW[9]	SVW[8]	
		SVW[7]	SVW[6]	SVW[5]	SVW[4]	SVW[3]	SVW[2]	SVW[1]	SVW[0]	
	SVSR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	SVS[10]	SVS[9]	SVS[8]	
		SVS[7]	SVS[6]	SVS[5]	SVS[4]	SVS[3]	SVS[2]	SVS[1]	SVS[0]	
	XMINR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	XMIN[12]	XMIN[11]	XMIN[10]	XMIN[9]	XMIN[8]	
		XMIN[7]	XMIN[6]	XMIN[5]	XMIN[4]	XMIN[3]	XMIN[2]	XMIN[1]	XMIN[0]	
	YMINR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	YMIN[12]	YMIN[11]	YMIN[10]	YMIN[9]	YMIN[8]	
		YMIN[7]	YMIN[6]	YMIN[5]	YMIN[4]	YMIN[3]	YMIN[2]	YMIN[1]	YMIN[0]	
	XMAXR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	XMAX[12]	XMAX[11]	XMAX[10]	XMAX[9]	XMAX[8]	
		XMAX[7]	XMAX[6]	XMAX[5]	XMAX[4]	XMAX[3]	XMAX[2]	XMAX[1]	XMAX[0]	
	YMAXR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	YMAX[12]	YMAX[11]	YMAX[10]	YMAX[9]	YMAX[8]	
		YMAX[7]	YMAX[6]	YMAX[5]	YMAX[4]	YMAX[3]	YMAX[2]	YMAX[1]	YMAX[0]	
	AMXSR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	AMXS[12]	AMXS[11]	AMXS[10]	AMXS[9]	AMXS[8]	
		AMXS[7]	AMXS[6]	AMXS[5]	AMXS[4]	AMXS[3]	AMXS[2]	AMXS[1]	AMXS[0]	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0		
Image renderer channel 0	AMYSR	-	-	-	-	-	-	-	-		
		-	-	-	-	-	-	-	-		
		-	-	-	AMYS[12]	AMYS[11]	AMYS[10]	AMYS[9]	AMYS[8]		
		AMYS[7]	AMYS[6]	AMYS[5]	AMYS[4]	AMYS[3]	AMYS[2]	AMYS[1]	AMYS[0]		
	AMXOR	-	-	-	-	-	-	-	-		
		-	-	-	-	-	-	-	-		
		-	-	-	AMXO[12]	AMXO[11]	AMXO[10]	AMXO[9]	AMXO[8]		
		AMXO[7]	AMXO[6]	AMXO[5]	AMXO[4]	AMXO[3]	AMXO[2]	AMXO[1]	AMXO[0]		
	AMYOR	-	-	-	-	-	-	-	-		
		-	-	-	-	-	-	-	-		
		-	-	-	AMYO[12]	AMYO[11]	AMYO[10]	AMYO[9]	AMYO[8]		
		AMYO[7]	AMYO[6]	AMYO[5]	AMYO[4]	AMYO[3]	AMYO[2]	AMYO[1]	AMYO[0]		
	MACR1	-	-	-	-	-	-	-	-		
		-	-	-	-	-	-	-	-		
		-	-	-	EMAM	-	-	-	-		
		-	-	-	-	-	-	-	-		
	LSPR	-	-	-	-	-	-	-	-		
		-	-	-	-	-	-	-	-		
		-	-	-	-	-	-	LSPR[9]	LSPR[8]		
		LSPR[7]	LSPR[6]	LSPR[5]	LSPR[4]	LSPR[3]	LSPR[2]	LSPR[1]	LSPR[0]		
	LEPR	-	-	-	-	-	-	-	-		
		-	-	-	-	-	-	-	-		
		-	-	-	-	-	-	LEPR[9]	LEPR[8]		
		LEPR[7]	LEPR[6]	LEPR[5]	LEPR[4]	LEPR[3]	LEPR[2]	LEPR[1]	LEPR[0]		
	LMSR	-	-	-	-	-	-	-	-		
		-	-	-	-	-	-	-	-		
		-	-	-	-	-	-	LMSR[2]	LMSR[1]	LMSR[0]	
		-	-	-	-	-	-	-	-		
	LMCR	-	-	-	-	-	-	-	-		
		-	-	-	-	-	-	-	-		
		-	-	-	-	-	-	-	DATASEL		
		-	-	-	-	-	-	-	CLKSEL		
	LMSPPCR	-	-	-	-	-	-	-	-		
		-	-	-	-	-	-	-	-		
		-	-	-	-	-	-	SPPC[10]	SPPC[9]	SPPC[8]	
		SPPC[7]	SPPC[6]	SPPC[5]	SPPC[4]	SPPC[3]	SPPC[2]	SPPC[1]	SPPC[0]		
	LMEPPCR	-	-	-	-	-	-	-	-		
		-	-	-	-	-	-	-	-		
		-	-	-	-	-	-	EPPC[10]	EPPC[9]	EPPC[8]	
		EPPC[7]	EPPC[6]	EPPC[5]	EPPC[4]	EPPC[3]	EPPC[2]	EPPC[1]	EPPC[0]		
	Image renderer channel 1	CR	-	-	-	-	-	-	-	-	
			-	-	-	-	-	-	-	-	
			SWRST	-	-	-	-	-	-	-	
		SR	-	-	-	-	-	-	SFE	ARS	RS
			-	-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-	-
			SFS	DSA	REN	-	-	-	INT	IER	TRA
		SRCR	-	-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-	-
			-	-	-	-	-	-	INTCLR	IERCLR	TRACLR
		ICR	-	-	-	-	-	-	-	-	-
-	-		-	-	-	-	-	-	-		
-	-		-	-	-	-	-	-	-		
-	-		-	-	-	-	INTENB	IERENB	TRAENB		



Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Image renderer channel 1	IMR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	INM	IEM
	DLPR	DLP[31]	DLP[30]	DLP[29]	DLP[28]	DLP[27]	DLP[26]	DLP[25]	DLP[24]
		DLP[23]	DLP[22]	DLP[21]	DLP[20]	DLP[19]	DLP[18]	DLP[17]	DLP[16]
		DLP[15]	DLP[14]	DLP[13]	DLP[12]	DLP[11]	DLP[10]	DLP[9]	DLP[8]
		DLP[7]	DLP[6]	DLP[5]	DLP[4]	DLP[3]	DLP[2]	DLP[1]	DLP[0]
	DLSAR	DLSA[28]	DLSA[27]	DLSA[26]	DLSA[25]	DLSA[24]	DLSA[23]	DLSA[22]	DLSA[21]
		DLSA[20]	DLSA[19]	DLSA[18]	DLSA[17]	DLSA[16]	DLSA[15]	DLSA[14]	DLSA[13]
		DLSA[12]	DLSA[11]	DLSA[10]	DLSA[9]	DLSA[8]	DLSA[7]	DLSA[6]	DLSA[5]
		DLSA[4]	DLSA[3]	DLSA[2]	DLSA[1]	DLSA[0]	-	-	-
	DSAR	DSA[26]	DSA[25]	DSA[24]	DSA[23]	DSA[22]	DSA[21]	DSA[20]	DSA[19]
		DSA[18]	DSA[17]	DSA[16]	DSA[15]	DSA[14]	DSA[13]	DSA[12]	DSA[11]
		DSA[10]	DSA[9]	DSA[8]	DSA[7]	DSA[6]	DSA[5]	DSA[4]	DSA[3]
		DSA[2]	DSA[1]	DSA[0]	-	-	-	-	-
	DSTR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	DST[13]	DST[12]	DST[11]	DST[10]	DST[9]	DST[8]
		DST[7]	DST[6]	DST[5]	DST[4]	DST[3]	DST[2]	DST[1]	DST[0]
	DSAR2	DSA2[26]	DSA2[25]	DSA2[24]	DSA2[23]	DSA2[22]	DSA2[21]	DSA2[20]	DSA2[19]
		DSA2[18]	DSA2[17]	DSA2[16]	DSA2[15]	DSA2[14]	DSA2[13]	DSA2[12]	DSA2[11]
		DSA2[10]	DSA2[9]	DSA2[8]	DSA2[7]	DSA2[6]	DSA2[5]	DSA2[4]	DSA2[3]
		DSA2[2]	DSA2[1]	DSA2[0]	-	-	-	-	-
	DLSA2	DLSA2[28]	DLSA2[27]	DLSA2[26]	DLSA2[25]	DLSA2[24]	DLSA2[23]	DLSA2[22]	DLSA2[21]
		DLSA2[20]	DLSA2[19]	DLSA2[18]	DLSA2[17]	DLSA2[16]	DLSA2[15]	DLSA2[14]	DLSA2[13]
		DLSA2[12]	DLSA2[11]	DLSA2[10]	DLSA2[9]	DLSA2[8]	DLSA2[7]	DLSA2[6]	DLSA2[5]
		DLSA2[4]	DLSA2[3]	DLSA2[2]	DLSA2[1]	DLSA2[0]	-	-	-
	TRIMR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	TCM	DUDVM	DXDYM	AUTOSG	AUTODG	BFE	TME
	TRIMSR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	TCMS	DUDVMS	DXDYMS	AUTOSGS	AUTODGS	BFES	TMES
	TRIMCR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	TCMC	DUDVMC	DXDYMC	AUTOSGC	AUTODGC	BFEC	TMEC
	TRICR	YCFORM	-	-	-	-	-	-	-
		TCV[7]	TCV[6]	TCV[5]	TCV[4]	TCV[3]	TCV[2]	TCV[1]	TCV[0]
		TCU[7]	TCU[6]	TCU[5]	TCU[4]	TCU[3]	TCU[2]	TCU[1]	TCU[0]
		TCY[7]	TCY[6]	TCY[5]	TCY[4]	TCY[3]	TCY[2]	TCY[1]	TCY[0]
	UVDPOR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	DDP
		-	-	-	-	-	UVDPO[2]	UVDPO[1]	UVDPO[0]
	SUSR	-	-	-	-	-	SUW[10]	SUW[9]	SUW[8]
		SUW[7]	SUW[6]	SUW[5]	SUW[4]	SUW[3]	SUW[2]	SUW[1]	SUW[0]
		-	-	-	-	-	SVW[10]	SVW[9]	SVW[8]
		SVW[7]	SVW[6]	SVW[5]	SVW[4]	SVW[3]	SVW[2]	SVW[1]	SVW[0]
	SVSR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	SVS[10]	SVS[9]	SVS[8]
		SVS[7]	SVS[6]	SVS[5]	SVS[4]	SVS[3]	SVS[2]	SVS[1]	SVS[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Image renderer channel 1	XMINR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	XMIN[12]	XMIN[11]	XMIN[10]	XMIN[9]	XMIN[8]
		XMIN[7]	XMIN[6]	XMIN[5]	XMIN[4]	XMIN[3]	XMIN[2]	XMIN[1]	XMIN[0]
	YMINR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	YMIN[12]	YMIN[11]	YMIN[10]	YMIN[9]	YMIN[8]
		YMIN[7]	YMIN[6]	YMIN[5]	YMIN[4]	YMIN[3]	YMIN[2]	YMIN[1]	YMIN[0]
	XMAXR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	XMAX[12]	XMAX[11]	XMAX[10]	XMAX[9]	XMAX[8]
		XMAX[7]	XMAX[6]	XMAX[5]	XMAX[4]	XMAX[3]	XMAX[2]	XMAX[1]	XMAX[0]
	YMAXR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	YMAX[12]	YMAX[11]	YMAX[10]	YMAX[9]	YMAX[8]
		YMAX[7]	YMAX[6]	YMAX[5]	YMAX[4]	YMAX[3]	YMAX[2]	YMAX[1]	YMAX[0]
	AMXSR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	AMXS[12]	AMXS[11]	AMXS[10]	AMXS[9]	AMXS[8]
		AMXS[7]	AMXS[6]	AMXS[5]	AMXS[4]	AMXS[3]	AMXS[2]	AMXS[1]	AMXS[0]
	AMYSR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	AMYS[12]	AMYS[11]	AMYS[10]	AMYS[9]	AMYS[8]
		AMYS[7]	AMYS[6]	AMYS[5]	AMYS[4]	AMYS[3]	AMYS[2]	AMYS[1]	AMYS[0]
	AMXOR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	AMXO[12]	AMXO[11]	AMXO[10]	AMXO[9]	AMXO[8]
		AMXO[7]	AMXO[6]	AMXO[5]	AMXO[4]	AMXO[3]	AMXO[2]	AMXO[1]	AMXO[0]
	AMYOR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	AMYO[12]	AMYO[11]	AMYO[10]	AMYO[9]	AMYO[8]
		AMYO[7]	AMYO[6]	AMYO[5]	AMYO[4]	AMYO[3]	AMYO[2]	AMYO[1]	AMYO[0]
	MACR1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	EMAM	-	-	-	-
		-	-	-	-	-	-	-	-
	LSPR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	LSPR[9]	LSPR[8]
		LSPR[7]	LSPR[6]	LSPR[5]	LSPR[4]	LSPR[3]	LSPR[2]	LSPR[1]	LSPR[0]
	LEPR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	LEPR[9]	LEPR[8]
		LEPR[7]	LEPR[6]	LEPR[5]	LEPR[4]	LEPR[3]	LEPR[2]	LEPR[1]	LEPR[0]
	LMSR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	LMSR[2]	LMSR[1]	LMSR[0]
		-	-	-	-	-	-	-	-
	LMCR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	DATASEL
		-	-	-	-	-	-	-	CLKSEL
	LMSPPCR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	SPPC[10]	SPPC[9]	SPPC[8]
		SPPC[7]	SPPC[6]	SPPC[5]	SPPC[4]	SPPC[3]	SPPC[2]	SPPC[1]	SPPC[0]
	LMEPPCR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	EPPC[10]	EPPC[9]	EPPC[8]
		EPPC[7]	EPPC[6]	EPPC[5]	EPPC[4]	EPPC[3]	EPPC[2]	EPPC[1]	EPPC[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Image renderer for display	CR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		SWRST	-	-	-	-	-	-	-
		-	-	-	-	-	-	ARS	RS
	SR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	DSA	REN	-	-	-	INT	IER
	SRCR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	INTCLR	IERCLR
	ICR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	INTENB	IERENB
	IMR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	INM	IEM
	DLPR	DLP[31]	DLP[30]	DLP[29]	DLP[28]	DLP[27]	DLP[26]	DLP[25]	DLP[24]
		DLP[23]	DLP[22]	DLP[21]	DLP[20]	DLP[19]	DLP[18]	DLP[17]	DLP[16]
		DLP[15]	DLP[14]	DLP[13]	DLP[12]	DLP[11]	DLP[10]	DLP[9]	DLP[8]
		DLP[7]	DLP[6]	DLP[5]	DLP[4]	DLP[3]	DLP[2]	DLP[1]	DLP[0]
	DLSAR	DLSA[28]	DLSA[27]	DLSA[26]	DLSA[25]	DLSA[24]	DLSA[23]	DLSA[22]	DLSA[21]
		DLSA[20]	DLSA[19]	DLSA[18]	DLSA[17]	DLSA[16]	DLSA[15]	DLSA[14]	DLSA[13]
		DLSA[12]	DLSA[11]	DLSA[10]	DLSA[9]	DLSA[8]	DLSA[7]	DLSA[6]	DLSA[5]
		DLSA[4]	DLSA[3]	DLSA[2]	DLSA[1]	DLSA[0]	-	-	-
	DSAR	DSA[26]	DSA[25]	DSA[24]	DSA[23]	DSA[22]	DSA[21]	DSA[20]	DSA[19]
		DSA[18]	DSA[17]	DSA[16]	DSA[15]	DSA[14]	DSA[13]	DSA[12]	DSA[11]
		DSA[10]	DSA[9]	DSA[8]	DSA[7]	DSA[6]	DSA[5]	DSA[4]	DSA[3]
		DSA[2]	DSA[1]	DSA[0]	-	-	-	-	-
	DSTR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	DST[13]	DST[12]	DST[11]	DST[10]	DST[9]	DST[8]
		DST[7]	DST[6]	DST[5]	DST[4]	DST[3]	DST[2]	DST[1]	DST[0]
	DSAR2	DSA2[26]	DSA2[25]	DSA2[24]	DSA2[23]	DSA2[22]	DSA2[21]	DSA2[20]	DSA2[19]
		DSA2[18]	DSA2[17]	DSA2[16]	DSA2[15]	DSA2[14]	DSA2[13]	DSA2[12]	DSA2[11]
		DSA2[10]	DSA2[9]	DSA2[8]	DSA2[7]	DSA2[6]	DSA2[5]	DSA2[4]	DSA2[3]
		DSA2[2]	DSA2[1]	DSA2[0]	-	-	-	-	-
	TRIMR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		CFS	CAS[3]	CAS[2]	CAS[1]	CAS[0]	-	DTHE	-
		-	TCM	DUDVM	DXDYM	AUTOSG	AUTODG	BFE	TME
	TRIMSR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		CFSS	CASS3	CASS2	CASS1	CASS0	-	DTHES	-
		-	TCMS	DUDVMS	DXDYMS	AUTOSGS	AUTODGS	BFES	TMES
	TRIMCR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		CFSC	CASC3	CASC2	CASC1	CASC0	-	DTHEC	-
		-	TCMC	DUDVMC	DXDYMC	AUTOSGC	AUTODGC	BFEC	TMEC

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Image renderer for display	TRICR	TCB[7]	TCB[6]	TCB[5]	TCB[4]	TCB[3]	TCB[2]	TCB[1]	TCB[0]	
		TCG[7]	TCG[6]	TCG[5]	TCG[4]	TCG[3]	TCG[2]	TCG[1]	TCG[0]	
		TCR[7]	TCR[6]	TCR[5]	TCR[4]	TCR[3]	TCR[2]	TCR[1]	TCR[0]	
		TCA[7]	TCA[6]	TCA[5]	TCA[4]	TCA[3]	TCA[2]	TCA[1]	TCA[0]	
	UVDPOR	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	DDP
		-	-	-	-	-	UVDPO[2]	UVDPO[1]	UVDPO[0]	-
	SUSR	-	-	-	-	-	-	SUW[10]	SUW[9]	SUW[8]
		SUW[7]	SUW[6]	SUW[5]	SUW[4]	SUW[3]	SUW[2]	SUW[1]	SUW[0]	-
		-	-	-	-	-	-	SVW[10]	SVW[9]	SVW[8]
		SVW[7]	SVW[6]	SVW[5]	SVW[4]	SVW[3]	SVW[2]	SVW[1]	SVW[0]	-
	SVSR	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	SVS[10]	SVS[9]	SVS[8]
		SVS[7]	SVS[6]	SVS[5]	SVS[4]	SVS[3]	SVS[2]	SVS[1]	SVS[0]	-
	XMINR	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	XMIN[12]	XMIN[11]	XMIN[10]	XMIN[9]	XMIN[8]	-
		XMIN[7]	XMIN[6]	XMIN[5]	XMIN[4]	XMIN[3]	XMIN[2]	XMIN[1]	XMIN[0]	-
	YMINR	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	YMIN[12]	YMIN[11]	YMIN[10]	YMIN[9]	YMIN[8]	-
		YMIN[7]	YMIN[6]	YMIN[5]	YMIN[4]	YMIN[3]	YMIN[2]	YMIN[1]	YMIN[0]	-
	XMAXR	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	XMAX[12]	XMAX[11]	XMAX[10]	XMAX[9]	XMAX[8]	-
		XMAX[7]	XMAX[6]	XMAX[5]	XMAX[4]	XMAX[3]	XMAX[2]	XMAX[1]	XMAX[0]	-
	YMAXR	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	YMAX[12]	YMAX[11]	YMAX[10]	YMAX[9]	YMAX[8]	-
		YMAX[7]	YMAX[6]	YMAX[5]	YMAX[4]	YMAX[3]	YMAX[2]	YMAX[1]	YMAX[0]	-
	AMXSR	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	AMXS[12]	AMXS[11]	AMXS[10]	AMXS[9]	AMXS[8]	-
		AMXS[7]	AMXS[6]	AMXS[5]	AMXS[4]	AMXS[3]	AMXS[2]	AMXS[1]	AMXS[0]	-
	AMYSR	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	AMYS[12]	AMYS[11]	AMYS[10]	AMYS[9]	AMYS[8]	-
		AMYS[7]	AMYS[6]	AMYS[5]	AMYS[4]	AMYS[3]	AMYS[2]	AMYS[1]	AMYS[0]	-
	AMXOR	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	AMXO[12]	AMXO[11]	AMXO[10]	AMXO[9]	AMXO[8]	-
		AMXO[7]	AMXO[6]	AMXO[5]	AMXO[4]	AMXO[3]	AMXO[2]	AMXO[1]	AMXO[0]	-
	AMYOR	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	AMYO[12]	AMYO[11]	AMYO[10]	AMYO[9]	AMYO[8]	-
		AMYO[7]	AMYO[6]	AMYO[5]	AMYO[4]	AMYO[3]	AMYO[2]	AMYO[1]	AMYO[0]	-
	MACR1	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	EMAM	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
	LSPR	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	LSPR[9]	LSPR[8]
		LSPR[7]	LSPR[6]	LSPR[5]	LSPR[4]	LSPR[3]	LSPR[2]	LSPR[1]	LSPR[0]	-

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Image renderer for display	LEPR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	LEPR[9]	LEPR[8]	
		LEPR[7]	LEPR[6]	LEPR[5]	LEPR[4]	LEPR[3]	LEPR[2]	LEPR[1]	LEPR[0]	
	LMSR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	LMSR[2]	LMSR[1]	LMSR[0]	
Display out comparison unit channel 0	DOCMCR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	CMPRU	
		-	-	-	-	-	-	-	CMPR	
		-	-	-	-	-	-	-	-	
	DOCMSTR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	CMPOST	
	DOCMCLSTR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	CMPCCLST	
	DOCMENR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	CMPIEN	
	DOCMPMR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	CMPDFF[1]	CMPDFF[0]	CMGBT
		CMPDFA[7]	CMPDFA[6]	CMPDFA[5]	CMPDFA[4]	CMPDFA[3]	CMPDFA[2]	CMPDFA[1]	CMPDFA[0]	
		CMPPDAUF	CMPCVF[1]	CMPCVF[0]	-	CMPSLIP[3]	CMPSLIP[2]	CMPSLIP[1]	CMPSLIP[0]	
	DOCMCCRRCR	CMPECRC[31]	CMPECRC[30]	CMPECRC[29]	CMPECRC[28]	CMPECRC[27]	CMPECRC[26]	CMPECRC[25]	CMPECRC[24]	
		CMPECRC[23]	CMPECRC[22]	CMPECRC[21]	CMPECRC[20]	CMPECRC[19]	CMPECRC[18]	CMPECRC[17]	CMPECRC[16]	
		CMPECRC[15]	CMPECRC[14]	CMPECRC[13]	CMPECRC[12]	CMPECRC[11]	CMPECRC[10]	CMPECRC[9]	CMPECRC[8]	
		CMPECRC[7]	CMPECRC[6]	CMPECRC[5]	CMPECRC[4]	CMPECRC[3]	CMPECRC[2]	CMPECRC[1]	CMPECRC[0]	
	DOCMCCRRCR	CMPCRCR[31]	CMPCRCR[30]	CMPCRCR[29]	CMPCRCR[28]	CMPCRCR[27]	CMPCRCR[26]	CMPCRCR[25]	CMPCRCR[24]	
		CMPCRCR[23]	CMPCRCR[22]	CMPCRCR[21]	CMPCRCR[20]	CMPCRCR[19]	CMPCRCR[18]	CMPCRCR[17]	CMPCRCR[16]	
		CMPCRCR[15]	CMPCRCR[14]	CMPCRCR[13]	CMPCRCR[12]	CMPCRCR[11]	CMPCRCR[10]	CMPCRCR[9]	CMPCRCR[8]	
		CMPCRCR[7]	CMPCRCR[6]	CMPCRCR[5]	CMPCRCR[4]	CMPCRCR[3]	CMPCRCR[2]	CMPCRCR[1]	CMPCRCR[0]	
	DOCMSPXR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	CMSPSX[10]	CMSPSX[9]	CMSPSX[8]
		CMSPSX[7]	CMSPSX[6]	CMSPSX[5]	CMSPSX[4]	CMSPSX[3]	CMSPSX[2]	CMSPSX[1]	CMSPSX[0]	
	DOCMSPYR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	CMSPSY[10]	CMSPSY[9]	CMSPSY[8]
		CMSPSY[7]	CMSPSY[6]	CMSPSY[5]	CMSPSY[4]	CMSPSY[3]	CMSPSY[2]	CMSPSY[1]	CMSPSY[0]	
	DOCMSZXR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	CMPSZX[10]	CMPSZX[9]	CMPSZX[8]
		CMPSZX[7]	CMPSZX[6]	CMPSZX[5]	CMPSZX[4]	CMPSZX[3]	CMPSZX[2]	CMPSZX[1]	CMPSZX[0]	
	DOCMSZYR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	CMPSZY[10]	CMPSZY[9]	CMPSZY[8]
		CMPSZY[7]	CMPSZY[6]	CMPSZY[5]	CMPSZY[4]	CMPSZY[3]	CMPSZY[2]	CMPSZY[1]	CMPSZY[0]	
	DOCMCRICR	CRCINI[31]	CRCINI[30]	CRCINI[29]	CRCINI[28]	CRCINI[27]	CRCINI[26]	CRCINI[25]	CRCINI[24]	
		CRCINI[23]	CRCINI[22]	CRCINI[21]	CRCINI[20]	CRCINI[19]	CRCINI[18]	CRCINI[17]	CRCINI[16]	
		CRCINI[15]	CRCINI[14]	CRCINI[13]	CRCINI[12]	CRCINI[11]	CRCINI[10]	CRCINI[9]	CRCINI[8]	
		CRCINI[7]	CRCINI[6]	CRCINI[5]	CRCINI[4]	CRCINI[3]	CRCINI[2]	CRCINI[1]	CRCINI[0]	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Display out comparison unit channel 1	DOCMCR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
	DOCMSTR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
	DOCMCLSTR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
	DOCMENR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
	DOCMPMR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		CMPDFA[7]	CMPDFA[6]	CMPDFA[5]	CMPDFA[4]	CMPDFA[3]	CMPDFA[2]	CMPDFA[1]	CMPDFA[0]	
		CMPDAUF	CMPCVF[1]	CMPCVF[0]	-	CMPSERP[3]	CMPSERP[2]	CMPSERP[1]	CMPSERP[0]	
	DOCMCRCR	CMPECRC[31]	CMPECRC[30]	CMPECRC[29]	CMPECRC[28]	CMPECRC[27]	CMPECRC[26]	CMPECRC[25]	CMPECRC[24]	
		CMPECRC[23]	CMPECRC[22]	CMPECRC[21]	CMPECRC[20]	CMPECRC[19]	CMPECRC[18]	CMPECRC[17]	CMPECRC[16]	
		CMPECRC[15]	CMPECRC[14]	CMPECRC[13]	CMPECRC[12]	CMPECRC[11]	CMPECRC[10]	CMPECRC[9]	CMPECRC[8]	
		CMPECRC[7]	CMPECRC[6]	CMPECRC[5]	CMPECRC[4]	CMPECRC[3]	CMPECRC[2]	CMPECRC[1]	CMPECRC[0]	
	DOCMCCRCR	CMPCRC[31]	CMPCRC[30]	CMPCRC[29]	CMPCRC[28]	CMPCRC[27]	CMPCRC[26]	CMPCRC[25]	CMPCRC[24]	
		CMPCRC[23]	CMPCRC[22]	CMPCRC[21]	CMPCRC[20]	CMPCRC[19]	CMPCRC[18]	CMPCRC[17]	CMPCRC[16]	
		CMPCRC[15]	CMPCRC[14]	CMPCRC[13]	CMPCRC[12]	CMPCRC[11]	CMPCRC[10]	CMPCRC[9]	CMPCRC[8]	
		CMPCRC[7]	CMPCRC[6]	CMPCRC[5]	CMPCRC[4]	CMPCRC[3]	CMPCRC[2]	CMPCRC[1]	CMPCRC[0]	
	DOCMSPXR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		CMSPSX[7]	CMSPSX[6]	CMSPSX[5]	CMSPSX[4]	CMSPSX[3]	CMSPSX[2]	CMSPSX[1]	CMSPSX[0]	
	DOCMSPYR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		CMSPSY[7]	CMSPSY[6]	CMSPSY[5]	CMSPSY[4]	CMSPSY[3]	CMSPSY[2]	CMSPSY[1]	CMSPSY[0]	
	DOCMSZXR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		CMPSZX[7]	CMPSZX[6]	CMPSZX[5]	CMPSZX[4]	CMPSZX[3]	CMPSZX[2]	CMPSZX[1]	CMPSZX[0]	
	DOCMSZYR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		CMPSZY[7]	CMPSZY[6]	CMPSZY[5]	CMPSZY[4]	CMPSZY[3]	CMPSZY[2]	CMPSZY[1]	CMPSZY[0]	
	DOCMCIRCIR	CRCINI[31]	CRCINI[30]	CRCINI[29]	CRCINI[28]	CRCINI[27]	CRCINI[26]	CRCINI[25]	CRCINI[24]	
		CRCINI[23]	CRCINI[22]	CRCINI[21]	CRCINI[20]	CRCINI[19]	CRCINI[18]	CRCINI[17]	CRCINI[16]	
		CRCINI[15]	CRCINI[14]	CRCINI[13]	CRCINI[12]	CRCINI[11]	CRCINI[10]	CRCINI[9]	CRCINI[8]	
		CRCINI[7]	CRCINI[6]	CRCINI[5]	CRCINI[4]	CRCINI[3]	CRCINI[2]	CRCINI[1]	CRCINI[0]	
	JPEG codec unit	JCMOD	-	-	-	-	DSP	REDU[2]	REDU[1]	REDU[0]
		JCCMD	BRST	-	-	-	-	JEND	JRST	JSRT
		JCQTN	-	-	QT3[1]	QT3[0]	QT2[1]	QT2[0]	QT1[1]	QT1[0]
		JCHTN	-	-	HTA3	HTD3	HTA2	HTD2	HTA1	HTD1
		JCDRIU	DRIU[7]	DRIU[6]	DRIU[5]	DRIU[4]	DRIU[3]	DRIU[2]	DRIU[1]	DRIU[0]
		JCDRID	DRID[7]	DRID[6]	DRID[5]	DRID[4]	DRID[3]	DRID[2]	DRID[1]	DRID[0]
		JCVSZU	VSZU[7]	VSZU[6]	VSZU[5]	VSZU[4]	VSZU[3]	VSZU[2]	VSZU[1]	VSZU[0]
		JCVSZD	VSZD[7]	VSZD[6]	VSZD[5]	VSZD[4]	VSZD[3]	VSZD[2]	VSZD[1]	VSZD[0]
		JCHSZU	HSZU[7]	HSZU[6]	HSZU[5]	HSZU[4]	HSZU[3]	HSZU[2]	HSZU[1]	HSZU[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
JPEG codec unit	JCHSZD	HSZD[7]	HSZD[6]	HSZD[5]	HSZD[4]	HSZD[3]	HSZD[2]	HSZD[1]	HSZD[0]	
	JCDTCU	DCU[7]	DCU[6]	DCU[5]	DCU[4]	DCU[3]	DCU[2]	DCU[1]	DCU[0]	
	JCDTCM	DCM[7]	DCM[6]	DCM[5]	DCM[4]	DCM[3]	DCM[2]	DCM[1]	DCM[0]	
	JCDTCD	DCD[7]	DCD[6]	DCD[5]	DCD[4]	DCD[3]	DCD[2]	DCD[1]	DCD[0]	
	JINTE0	INT7	INT6	INT5	-	INT3	-	-	-	
	JINTS0	-	INS6	INS5	-	INS3	-	-	-	
	JCDERR	-	-	-	-	ERR[3]	ERR[2]	ERR[1]	ERR[0]	
	JCRST	-	-	-	-	-	-	-	RST	
	JIFECNT	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	JOUTRINI	JOUTRCMD	JOUTC	-	JOUTSWAP[2]	JOUTSWAP[1]	JOUTSWAP[0]	-
		-	DINRINI	DINRCMD	DINLC	-	DINSWAP[2]	DINSWAP[1]	DINSWAP[0]	-
	JIFESA	ESA[31]	ESA[30]	ESA[29]	ESA[28]	ESA[27]	ESA[26]	ESA[25]	ESA[24]	ESA[24]
		ESA[23]	ESA[22]	ESA[21]	ESA[20]	ESA[19]	ESA[18]	ESA[17]	ESA[16]	ESA[16]
		ESA[15]	ESA[14]	ESA[13]	ESA[12]	ESA[11]	ESA[10]	ESA[9]	ESA[8]	ESA[8]
		ESA[7]	ESA[6]	ESA[5]	ESA[4]	ESA[3]	ESA[2]	ESA[1]	ESA[0]	ESA[0]
	JIFESOFST	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	ESMW[14]	ESMW[13]	ESMW[12]	ESMW[11]	ESMW[10]	ESMW[9]	ESMW[8]	ESMW[8]
		ESMW[7]	ESMW[6]	ESMW[5]	ESMW[4]	ESMW[3]	ESMW[2]	ESMW[1]	ESMW[0]	ESMW[0]
	JIFEDA	EDA[31]	EDA[30]	EDA[29]	EDA[28]	EDA[27]	EDA[26]	EDA[25]	EDA[24]	EDA[24]
		EDA[23]	EDA[22]	EDA[21]	EDA[20]	EDA[19]	EDA[18]	EDA[17]	EDA[16]	EDA[16]
		EDA[15]	EDA[14]	EDA[13]	EDA[12]	EDA[11]	EDA[10]	EDA[9]	EDA[8]	EDA[8]
		EDA[7]	EDA[6]	EDA[5]	EDA[4]	EDA[3]	EDA[2]	EDA[1]	EDA[0]	EDA[0]
	JIFESLC	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		LINES[15]	LINES[14]	LINES[13]	LINES[12]	LINES[11]	LINES[10]	LINES[9]	LINES[8]	LINES[8]
	JIFEDDC	LINES[7]	LINES[6]	LINES[5]	LINES[4]	LINES[3]	LINES[2]	LINES[1]	LINES[0]	LINES[0]
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		JDATAS[15]	JDATAS[14]	JDATAS[13]	JDATAS[12]	JDATAS[11]	JDATAS[10]	JDATAS[9]	JDATAS[8]	JDATAS[8]
	JIFDCNT	JDATAS[7]	JDATAS[6]	JDATAS[5]	JDATAS[4]	JDATAS[3]	JDATAS[2]	JDATAS[1]	JDATAS[0]	JDATAS[0]
		-	-	VINTER[1]	VINTER[0]	HINTER[1]	HINTER[0]	OPF[1]	OPF[0]	OPF[0]
		-	-	-	-	-	-	-	-	-
		-	JINRINI	JINRCMD	JINC	-	JINSWAP[2]	JINSWAP[1]	JINSWAP[0]	JINSWAP[0]
	JIFDSA	-	DOUTRINI	DOUTRCMD	DOUTLC	-	DOUTSWAP[2]	DOUTSWAP[1]	DOUTSWAP[0]	DOUTSWAP[0]
		DSA[31]	DSA[30]	DSA[29]	DSA[28]	DSA[27]	DSA[26]	DSA[25]	DSA[24]	DSA[24]
		DSA[23]	DSA[22]	DSA[21]	DSA[20]	DSA[19]	DSA[18]	DSA[17]	DSA[16]	DSA[16]
		DSA[15]	DSA[14]	DSA[13]	DSA[12]	DSA[11]	DSA[10]	DSA[9]	DSA[8]	DSA[8]
	JIFDDOFST	DSA[7]	DSA[6]	DSA[5]	DSA[4]	DSA[3]	DSA[2]	DSA[1]	DSA[0]	DSA[0]
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	DDMW[14]	DDMW[13]	DDMW[12]	DDMW[11]	DDMW[10]	DDMW[9]	DDMW[8]	DDMW[8]
	JIFDDA	DDMW[7]	DDMW[6]	DDMW[5]	DDMW[4]	DDMW[3]	DDMW[2]	DDMW[1]	DDMW[0]	DDMW[0]
		DDA[31]	DDA[30]	DDA[29]	DDA[28]	DDA[27]	DDA[26]	DDA[25]	DDA[24]	DDA[24]
		DDA[23]	DDA[22]	DDA[21]	DDA[20]	DDA[19]	DDA[18]	DDA[17]	DDA[16]	DDA[16]
		DDA[15]	DDA[14]	DDA[13]	DDA[12]	DDA[11]	DDA[10]	DDA[9]	DDA[8]	DDA[8]
	JIFSDC	DDA[7]	DDA[6]	DDA[5]	DDA[4]	DDA[3]	DDA[2]	DDA[1]	DDA[0]	DDA[0]
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		JDATAS[15]	JDATAS[14]	JDATAS[13]	JDATAS[12]	JDATAS[11]	JDATAS[10]	JDATAS[9]	JDATAS[8]	JDATAS[8]
	JIFDLC	JDATAS[7]	JDATAS[6]	JDATAS[5]	JDATAS[4]	JDATAS[3]	JDATAS[2]	JDATAS[1]	JDATAS[0]	JDATAS[0]
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		LINES[15]	LINES[14]	LINES[13]	LINES[12]	LINES[11]	LINES[10]	LINES[9]	LINES[8]	LINES[8]
	JIFDLC	LINES[7]	LINES[6]	LINES[5]	LINES[4]	LINES[3]	LINES[2]	LINES[1]	LINES[0]	LINES[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
JPEG codec unit	JIFDADT	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		ALPHA[7]	ALPHA[6]	ALPHA[5]	ALPHA[4]	ALPHA[3]	ALPHA[2]	ALPHA[1]	ALPHA[0]	
	JINTE1	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	CBTEN	DINLEN	JOUTEN	-	DBTEN	JINEN	DOUTLEN	
	JINTS1	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	CBTF	DINLF	JOUTF	-	DBTF	JINF	DOUTF	
	JIFESVSZ	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		DINYCHG	-	-	-	-	-	-	-	
	JIFESHSZ	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		DOUYCHG	-	-	-	-	-	-	-	
	Capture engine unit	CAPSR	-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	CPKIL
			-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	CE
		CAPCR	FDRP[7]	FDRP[6]	FDRP[5]	FDRP[4]	FDRP[3]	FDRP[2]	FDRP[1]	FDRP[0]
			-	-	MTCM[1]	MTCM[0]	-	-	-	CTNCP
-			-	-	-	-	-	-	-	
-			-	-	-	-	-	-	-	
CAMCR		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	FLDPOL	
		-	-	-	DTIF	-	-	DTARY[1]	DTARY[0]	
		-	-	JPG[1]	JPG[0]	-	-	VDPOL	HDPOL	
CMCYR		-	-	VCYL[13]	VCYL[12]	VCYL[11]	VCYL[10]	VCYL[9]	VCYL[8]	
		VCYL[7]	VCYL[6]	VCYL[5]	VCYL[4]	VCYL[3]	VCYL[2]	VCYL[1]	VCYL[0]	
		-	-	HCYL[13]	HCYL[12]	HCYL[11]	HCYL[10]	HCYL[9]	HCYL[8]	
		HCYL[7]	HCYL[6]	HCYL[5]	HCYL[4]	HCYL[3]	HCYL[2]	HCYL[1]	HCYL[0]	
CAMOR		-	-	-	-	VOFST[11]	VOFST[10]	VOFST[9]	VOFST[8]	
		VOFST[7]	VOFST[6]	VOFST[5]	VOFST[4]	VOFST[3]	VOFST[2]	VOFST[1]	VOFST[0]	
		-	-	-	HOFST[12]	HOFST[11]	HOFST[10]	HOFST[9]	HOFST[8]	
		HOFST[7]	HOFST[6]	HOFST[5]	HOFST[4]	HOFST[3]	HOFST[2]	HOFST[1]	HOFST[0]	
CAPIWR		-	-	-	-	VWDTH[11]	VWDTH[10]	VWDTH[9]	VWDTH[8]	
		VWDTH[7]	VWDTH[6]	VWDTH[5]	VWDTH[4]	VWDTH[3]	VWDTH[2]	VWDTH[1]	VWDTH[0]	
		-	-	-	HWDTH[12]	HWDTH[11]	HWDTH[10]	HWDTH[9]	HWDTH[8]	
		HWDTH[7]	HWDTH[6]	HWDTH[5]	HWDTH[4]	HWDTH[3]	HWDTH[2]	HWDTH[1]	HWDTH[0]	
CAIFR		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	IFS	
		-	-	-	CIM	-	-	FCI[1]	FCI[0]	
CRCNTR		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	RVS	-	-	RS	RC	
CRCMPR		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	RA	



Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Capture engine unit	CFLCR	VMANT[3]	VMANT[2]	VMANT[1]	VMANT[0]	VFRAC[11]	VFRAC[10]	VFRAC[9]	VFRAC[8]
		VFRAC[7]	VFRAC[6]	VFRAC[5]	VFRAC[4]	VFRAC[3]	VFRAC[2]	VFRAC[1]	VFRAC[0]
		HMANT[3]	HMANT[2]	HMANT[1]	HMANT[0]	HFRAC[11]	HFRAC[10]	HFRAC[9]	HFRAC[8]
		HFRAC[7]	HFRAC[6]	HFRAC[5]	HFRAC[4]	HFRAC[3]	HFRAC[2]	HFRAC[1]	HFRAC[0]
	CFSZR	-	-	-	-	VFCLP[11]	VFCLP[10]	VFCLP[9]	VFCLP[8]
		VFCLP[7]	VFCLP[6]	VFCLP[5]	VFCLP[4]	VFCLP[3]	VFCLP[2]	VFCLP[1]	VFCLP[0]
		-	-	-	-	HFCLP[11]	HFCLP[10]	HFCLP[9]	HFCLP[8]
		HFCLP[7]	HFCLP[6]	HFCLP[5]	HFCLP[4]	HFCLP[3]	HFCLP[2]	HFCLP[1]	HFCLP[0]
	CDWDR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	CHDW[12]	CHDW[11]	CHDW[10]	CHDW[9]	CHDW[8]
		CHDW[7]	CHDW[6]	CHDW[5]	CHDW[4]	CHDW[3]	CHDW[2]	CHDW[1]	CHDW[0]
	CDAYR	CAYR[31]	CAYR[30]	CAYR[29]	CAYR[28]	CAYR[27]	CAYR[26]	CAYR[25]	CAYR[24]
		CAYR[23]	CAYR[22]	CAYR[21]	CAYR[20]	CAYR[19]	CAYR[18]	CAYR[17]	CAYR[16]
		CAYR[15]	CAYR[14]	CAYR[13]	CAYR[12]	CAYR[11]	CAYR[10]	CAYR[9]	CAYR[8]
		CAYR[7]	CAYR[6]	CAYR[5]	CAYR[4]	CAYR[3]	CAYR[2]	CAYR[1]	CAYR[0]
	CDACR	CACR[31]	CACR[30]	CACR[29]	CACR[28]	CACR[27]	CACR[26]	CACR[25]	CACR[24]
		CACR[23]	CACR[22]	CACR[21]	CACR[20]	CACR[19]	CACR[18]	CACR[17]	CACR[16]
		CACR[15]	CACR[14]	CACR[13]	CACR[12]	CACR[11]	CACR[10]	CACR[9]	CACR[8]
		CACR[7]	CACR[6]	CACR[5]	CACR[4]	CACR[3]	CACR[2]	CACR[1]	CACR[0]
	CDBYR	CBYR[31]	CBYR[30]	CBYR[29]	CBYR[28]	CBYR[27]	CBYR[26]	CBYR[25]	CBYR[24]
		CBYR[23]	CBYR[22]	CBYR[21]	CBYR[20]	CBYR[19]	CBYR[18]	CBYR[17]	CBYR[16]
		CBYR[15]	CBYR[14]	CBYR[13]	CBYR[12]	CBYR[11]	CBYR[10]	CBYR[9]	CBYR[8]
		CBYR[7]	CBYR[6]	CBYR[5]	CBYR[4]	CBYR[3]	CBYR[2]	CBYR[1]	CBYR[0]
	CBCR	CBCR[31]	CBCR[30]	CBCR[29]	CBCR[28]	CBCR[27]	CBCR[26]	CBCR[25]	CBCR[24]
		CBCR[23]	CBCR[22]	CBCR[21]	CBCR[20]	CBCR[19]	CBCR[18]	CBCR[17]	CBCR[16]
		CBCR[15]	CBCR[14]	CBCR[13]	CBCR[12]	CBCR[11]	CBCR[10]	CBCR[9]	CBCR[8]
		CBCR[7]	CBCR[6]	CBCR[5]	CBCR[4]	CBCR[3]	CBCR[2]	CBCR[1]	CBCR[0]
	CBDSR	-	-	-	-	-	-	-	-
		-	CBVS[22]	CBVS[21]	CBVS[20]	CBVS[19]	CBVS[18]	CBVS[17]	CBVS[16]
		CBVS[15]	CBVS[14]	CBVS[13]	CBVS[12]	CBVS[11]	CBVS[10]	CBVS[9]	CBVS[8]
		CBVS[7]	CBVS[6]	CBVS[5]	CBVS[4]	CBVS[3]	CBVS[2]	CBVS[1]	CBVS[0]
	CFWCR	FWV[26]	FWV[25]	FWV[24]	FWV[23]	FWV[22]	FWV[21]	FWV[20]	FWV[19]
		FWV[18]	FWV[17]	FWV[16]	FWV[15]	FWV[14]	FWV[13]	FWV[12]	FWV[11]
		FWV[10]	FWV[9]	FWV[8]	FWV[7]	FWV[6]	FWV[5]	FWV[4]	FWV[3]
		FWV[2]	FWV[1]	FWV[0]	-	-	-	-	FWE
	CLFCR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	LPF
	CDOCR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	CBE
		-	-	-	-	-	-	-	-
		-	-	-	CDS	-	COLS	COWS	COBS
	CEIER	-	-	-	-	-	-	NVDIE	NHDIE
		FWFIE	-	-	VBPIE	-	IGVSIE	IGHSIE	CDTOFIE
		CPBE4IE	CPBE3IE	CPBE2IE	CPBE1IE	-	-	VDIE	HDIE
		-	-	-	IGRWIE	-	-	CFEIE	CPEIE
	CETCR	-	-	-	-	-	-	NVD	NHD
		FWF	-	-	VBP	-	IGVS	IGHS	CDTOF
		CPBE4	CPBE3	CPBE2	CPBE1	-	-	VD	HD
		-	-	-	IGRW	-	-	CFE	CPE
	CSTSR	-	-	-	-	-	-	-	CRST
		-	-	-	-	-	-	-	CPFLD
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	CPTON

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Capture engine unit	CDSSR	CDSS[31]	CDSS[30]	CDSS[29]	CDSS[28]	CDSS[27]	CDSS[26]	CDSS[25]	CDSS[24]	
		CDSS[23]	CDSS[22]	CDSS[21]	CDSS[20]	CDSS[19]	CDSS[18]	CDSS[17]	CDSS[16]	
		CDSS[15]	CDSS[14]	CDSS[13]	CDSS[12]	CDSS[11]	CDSS[10]	CDSS[9]	CDSS[8]	
		CDSS[7]	CDSS[6]	CDSS[5]	CDSS[4]	CDSS[3]	CDSS[2]	CDSS[1]	CDSS[0]	
	CDAYR2	CAYR2[31]	CAYR2[30]	CAYR2[29]	CAYR2[28]	CAYR2[27]	CAYR2[26]	CAYR2[25]	CAYR2[24]	
		CAYR2[23]	CAYR2[22]	CAYR2[21]	CAYR2[20]	CAYR2[19]	CAYR2[18]	CAYR2[17]	CAYR2[16]	
		CAYR2[15]	CAYR2[14]	CAYR2[13]	CAYR2[12]	CAYR2[11]	CAYR2[10]	CAYR2[9]	CAYR2[8]	
		CAYR2[7]	CAYR2[6]	CAYR2[5]	CAYR2[4]	CAYR2[3]	CAYR2[2]	CAYR2[1]	CAYR2[0]	
	CDACR2	CACR2[31]	CACR2[30]	CACR2[29]	CACR2[28]	CACR2[27]	CACR2[26]	CACR2[25]	CACR2[24]	
		CACR2[23]	CACR2[22]	CACR2[21]	CACR2[20]	CACR2[19]	CACR2[18]	CACR2[17]	CACR2[16]	
		CACR2[15]	CACR2[14]	CACR2[13]	CACR2[12]	CACR2[11]	CACR2[10]	CACR2[9]	CACR2[8]	
		CACR2[7]	CACR2[6]	CACR2[5]	CACR2[4]	CACR2[3]	CACR2[2]	CACR2[1]	CACR2[0]	
	CDBYR2	CBYR2[31]	CBYR2[30]	CBYR2[29]	CBYR2[28]	CBYR2[27]	CBYR2[26]	CBYR2[25]	CBYR2[24]	
		CBYR2[23]	CBYR2[22]	CBYR2[21]	CBYR2[20]	CBYR2[19]	CBYR2[18]	CBYR2[17]	CBYR2[16]	
		CBYR2[15]	CBYR2[14]	CBYR2[13]	CBYR2[12]	CBYR2[11]	CBYR2[10]	CBYR2[9]	CBYR2[8]	
		CBYR2[7]	CBYR2[6]	CBYR2[5]	CBYR2[4]	CBYR2[3]	CBYR2[2]	CBYR2[1]	CBYR2[0]	
	CBCR2	CBCR2[31]	CBCR2[30]	CBCR2[29]	CBCR2[28]	CBCR2[27]	CBCR2[26]	CBCR2[25]	CBCR2[24]	
		CBCR2[23]	CBCR2[22]	CBCR2[21]	CBCR2[20]	CBCR2[19]	CBCR2[18]	CBCR2[17]	CBCR2[16]	
		CBCR2[15]	CBCR2[14]	CBCR2[13]	CBCR2[12]	CBCR2[11]	CBCR2[10]	CBCR2[9]	CBCR2[8]	
		CBCR2[7]	CBCR2[6]	CBCR2[5]	CBCR2[4]	CBCR2[3]	CBCR2[2]	CBCR2[1]	CBCR2[0]	
	Pixel format converter channel 0	PFVCR	-	-	-	-	-	-	-	
			-	-	-	-	-	-	-	
			-	-	-	DTH_ON	IFMT[1]	IFMT[0]	OFMT[1]	OFMT[0]
			-	-	DINSWAP16	DOUTSWAP16	DINSWAP32[1]	DINSWAP32[0]	DOUTSWAP32[1]	DOUTSWAP32[0]
		PFVICR	-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			-	PFVEEN	IFEN	OFEN	IDTRG[1]	IDTRG[0]	ODTRG[1]	ODTRG[0]
		PFVISR	-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			-	-	-	-	IFOVF	OFUFD	IFEMP	OFFUL
PFVID		ID[31]	ID[30]	ID[29]	ID[28]	ID[27]	ID[26]	ID[25]	ID[24]	
		ID[23]	ID[22]	ID[21]	ID[20]	ID[19]	ID[18]	ID[17]	ID[16]	
		ID[15]	ID[14]	ID[13]	ID[12]	ID[11]	ID[10]	ID[9]	ID[8]	
		ID[7]	ID[6]	ID[5]	ID[4]	ID[3]	ID[2]	ID[1]	ID[0]	
PFVOD		OD[31]	OD[30]	OD[29]	OD[28]	OD[27]	OD[26]	OD[25]	OD[24]	
		OD[23]	OD[22]	OD[21]	OD[20]	OD[19]	OD[18]	OD[17]	OD[16]	
		OD[15]	OD[14]	OD[13]	OD[12]	OD[11]	OD[10]	OD[9]	OD[8]	
		OD[7]	OD[6]	OD[5]	OD[4]	OD[3]	OD[2]	OD[1]	OD[0]	
PFVIFSR		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	IFVD[5]	IFVD[4]	IFVD[3]	IFVD[2]	IFVD[1]	IFVD[0]	
PFVOFSR		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	OFVD[5]	OFVD[4]	OFVD[3]	OFVD[2]	OFVD[1]	OFVD[0]	
PFVACR		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		ALPHA[7]	ALPHA[6]	ALPHA[5]	ALPHA[4]	ALPHA[3]	ALPHA[2]	ALPHA[1]	ALPHA[0]	
PFV_MTX_MODE		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	PFV_MTX_MD[1]	PFV_MTX_MD[0]	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Pixel format converter channel 0	PFV_MTX_YG_ADJ0	-	-	-	-	-	-	-	-	
		PFV_MTX_YG[7]	PFV_MTX_YG[6]	PFV_MTX_YG[5]	PFV_MTX_YG[4]	PFV_MTX_YG[3]	PFV_MTX_YG[2]	PFV_MTX_YG[1]	PFV_MTX_YG[0]	
		-	-	-	-	-	PFV_MTX_GG[10]	PFV_MTX_GG[9]	PFV_MTX_GG[8]	
		PFV_MTX_GG[7]	PFV_MTX_GG[6]	PFV_MTX_GG[5]	PFV_MTX_GG[4]	PFV_MTX_GG[3]	PFV_MTX_GG[2]	PFV_MTX_GG[1]	PFV_MTX_GG[0]	
	PFV_MTX_YG_ADJ1	-	-	-	-	-	-	PFV_MTX_GB[10]	PFV_MTX_GB[9]	PFV_MTX_GB[8]
		PFV_MTX_GB[7]	PFV_MTX_GB[6]	PFV_MTX_GB[5]	PFV_MTX_GB[4]	PFV_MTX_GB[3]	PFV_MTX_GB[2]	PFV_MTX_GB[1]	PFV_MTX_GB[0]	
		-	-	-	-	-	PFV_MTX_GR[10]	PFV_MTX_GR[9]	PFV_MTX_GR[8]	
		PFV_MTX_GR[7]	PFV_MTX_GR[6]	PFV_MTX_GR[5]	PFV_MTX_GR[4]	PFV_MTX_GR[3]	PFV_MTX_GR[2]	PFV_MTX_GR[1]	PFV_MTX_GR[0]	
	PFV_MTX_CBB_ADJ0	-	-	-	-	-	-	-	-	
		PFV_MTX_B[7]	PFV_MTX_B[6]	PFV_MTX_B[5]	PFV_MTX_B[4]	PFV_MTX_B[3]	PFV_MTX_B[2]	PFV_MTX_B[1]	PFV_MTX_B[0]	
		-	-	-	-	-	PFV_MTX_BG[10]	PFV_MTX_BG[9]	PFV_MTX_BG[8]	
		PFV_MTX_BG[7]	PFV_MTX_BG[6]	PFV_MTX_BG[5]	PFV_MTX_BG[4]	PFV_MTX_BG[3]	PFV_MTX_BG[2]	PFV_MTX_BG[1]	PFV_MTX_BG[0]	
	PFV_MTX_CBB_ADJ1	-	-	-	-	-	-	PFV_MTX_BB[10]	PFV_MTX_BB[9]	PFV_MTX_BB[8]
		PFV_MTX_BB[7]	PFV_MTX_BB[6]	PFV_MTX_BB[5]	PFV_MTX_BB[4]	PFV_MTX_BB[3]	PFV_MTX_BB[2]	PFV_MTX_BB[1]	PFV_MTX_BB[0]	
		-	-	-	-	-	PFV_MTX_BR[10]	PFV_MTX_BR[9]	PFV_MTX_BR[8]	
		PFV_MTX_BR[7]	PFV_MTX_BR[6]	PFV_MTX_BR[5]	PFV_MTX_BR[4]	PFV_MTX_BR[3]	PFV_MTX_BR[2]	PFV_MTX_BR[1]	PFV_MTX_BR[0]	
	PFV_MTX_CRR_ADJ0	-	-	-	-	-	-	-	-	
		PFV_MTX_R[7]	PFV_MTX_R[6]	PFV_MTX_R[5]	PFV_MTX_R[4]	PFV_MTX_R[3]	PFV_MTX_R[2]	PFV_MTX_R[1]	PFV_MTX_R[0]	
		-	-	-	-	-	PFV_MTX_RG[10]	PFV_MTX_RG[9]	PFV_MTX_RG[8]	
		PFV_MTX_RG[7]	PFV_MTX_RG[6]	PFV_MTX_RG[5]	PFV_MTX_RG[4]	PFV_MTX_RG[3]	PFV_MTX_RG[2]	PFV_MTX_RG[1]	PFV_MTX_RG[0]	
	PFV_MTX_CRR_ADJ1	-	-	-	-	-	-	PFV_MTX_RB[10]	PFV_MTX_RB[9]	PFV_MTX_RB[8]
		PFV_MTX_RB[7]	PFV_MTX_RB[6]	PFV_MTX_RB[5]	PFV_MTX_RB[4]	PFV_MTX_RB[3]	PFV_MTX_RB[2]	PFV_MTX_RB[1]	PFV_MTX_RB[0]	
		-	-	-	-	-	PFV_MTX_RR[10]	PFV_MTX_RR[9]	PFV_MTX_RR[8]	
		PFV_MTX_RR[7]	PFV_MTX_RR[6]	PFV_MTX_RR[5]	PFV_MTX_RR[4]	PFV_MTX_RR[3]	PFV_MTX_RR[2]	PFV_MTX_RR[1]	PFV_MTX_RR[0]	
	PFV_SZR	PFV_SZX[15]	PFV_SZX[14]	PFV_SZX[13]	PFV_SZX[12]	PFV_SZX[11]	PFV_SZX[10]	PFV_SZX[9]	PFV_SZX[8]	
		PFV_SZX[7]	PFV_SZX[6]	PFV_SZX[5]	PFV_SZX[4]	PFV_SZX[3]	PFV_SZX[2]	PFV_SZX[1]	PFV_SZX[0]	
		PFV_SZY[15]	PFV_SZY[14]	PFV_SZY[13]	PFV_SZY[12]	PFV_SZY[11]	PFV_SZY[10]	PFV_SZY[9]	PFV_SZY[8]	
		PFV_SZY[7]	PFV_SZY[6]	PFV_SZY[5]	PFV_SZY[4]	PFV_SZY[3]	PFV_SZY[2]	PFV_SZY[1]	PFV_SZY[0]	
	Pixel format converter channel 1	PFVCR	-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			-	-	-	DTH_ON	IFMT[1]	IFMT[0]	OFMT[1]	OFMT[0]
			-	-	DINSWAP16	DOUWSWAP16	DINSWAP32[1]	DINSWAP32[0]	DOUWSWAP32[1]	DOUWSWAP32[0]
		PFVICR	-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			-	PFVEEN	IFEN	OFEN	IDTRG[1]	IDTRG[0]	ODTRG[1]	ODTRG[0]
		PFVISR	-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			-	-	-	-	IFOVF	OFUDF	IFEMP	OFFUL
		PFVID	ID[31]	ID[30]	ID[29]	ID[28]	ID[27]	ID[26]	ID[25]	ID[24]
			ID[23]	ID[22]	ID[21]	ID[20]	ID[19]	ID[18]	ID[17]	ID[16]
			ID[15]	ID[14]	ID[13]	ID[12]	ID[11]	ID[10]	ID[9]	ID[8]
			ID[7]	ID[6]	ID[5]	ID[4]	ID[3]	ID[2]	ID[1]	ID[0]
		PFVOD	OD[31]	OD[30]	OD[29]	OD[28]	OD[27]	OD[26]	OD[25]	OD[24]
			OD[23]	OD[22]	OD[21]	OD[20]	OD[19]	OD[18]	OD[17]	OD[16]
			OD[15]	OD[14]	OD[13]	OD[12]	OD[11]	OD[10]	OD[9]	OD[8]
			OD[7]	OD[6]	OD[5]	OD[4]	OD[3]	OD[2]	OD[1]	OD[0]
PFVIFSR	-	-	-	-	-	-	-	-		
	-	-	-	-	-	-	-	-		
	-	-	-	-	-	-	-	-		
	-	-	IFVD[5]	IFVD[4]	IFVD[3]	IFVD[2]	IFVD[1]	IFVD[0]		
PFVOFSR	-	-	-	-	-	-	-	-		
	-	-	-	-	-	-	-	-		
	-	-	-	-	-	-	-	-		
	-	-	OFVD[5]	OFVD[4]	OFVD[3]	OFVD[2]	OFVD[1]	OFVD[0]		

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Pixel format converter channel 1	PFVACR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		ALPHA[7]	ALPHA[6]	ALPHA[5]	ALPHA[4]	ALPHA[3]	ALPHA[2]	ALPHA[1]	ALPHA[0]	
	PFV_MTX_MODE	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	PFV_MTX_MD[1]	PFV_MTX_MD[0]	
	PFV_MTX_YG_ADJ0	-	-	-	-	-	-	-	-	
		PFV_MTX_YG[7]	PFV_MTX_YG[6]	PFV_MTX_YG[5]	PFV_MTX_YG[4]	PFV_MTX_YG[3]	PFV_MTX_YG[2]	PFV_MTX_YG[1]	PFV_MTX_YG[0]	
		-	-	-	-	-	PFV_MTX_GG[10]	PFV_MTX_GG[9]	PFV_MTX_GG[8]	
		PFV_MTX_GG[7]	PFV_MTX_GG[6]	PFV_MTX_GG[5]	PFV_MTX_GG[4]	PFV_MTX_GG[3]	PFV_MTX_GG[2]	PFV_MTX_GG[1]	PFV_MTX_GG[0]	
	PFV_MTX_YG_ADJ1	-	-	-	-	-	PFV_MTX_GB[10]	PFV_MTX_GB[9]	PFV_MTX_GB[8]	
		PFV_MTX_GB[7]	PFV_MTX_GB[6]	PFV_MTX_GB[5]	PFV_MTX_GB[4]	PFV_MTX_GB[3]	PFV_MTX_GB[2]	PFV_MTX_GB[1]	PFV_MTX_GB[0]	
		-	-	-	-	-	PFV_MTX_GR[10]	PFV_MTX_GR[9]	PFV_MTX_GR[8]	
		PFV_MTX_GR[7]	PFV_MTX_GR[6]	PFV_MTX_GR[5]	PFV_MTX_GR[4]	PFV_MTX_GR[3]	PFV_MTX_GR[2]	PFV_MTX_GR[1]	PFV_MTX_GR[0]	
	PFV_MTX_CBB_ADJ0	-	-	-	-	-	-	-	-	
		PFV_MTX_B[7]	PFV_MTX_B[6]	PFV_MTX_B[5]	PFV_MTX_B[4]	PFV_MTX_B[3]	PFV_MTX_B[2]	PFV_MTX_B[1]	PFV_MTX_B[0]	
		-	-	-	-	-	PFV_MTX_BG[10]	PFV_MTX_BG[9]	PFV_MTX_BG[8]	
		PFV_MTX_BG[7]	PFV_MTX_BG[6]	PFV_MTX_BG[5]	PFV_MTX_BG[4]	PFV_MTX_BG[3]	PFV_MTX_BG[2]	PFV_MTX_BG[1]	PFV_MTX_BG[0]	
	PFV_MTX_CBB_ADJ1	-	-	-	-	-	PFV_MTX_BB[10]	PFV_MTX_BB[9]	PFV_MTX_BB[8]	
		PFV_MTX_BB[7]	PFV_MTX_BB[6]	PFV_MTX_BB[5]	PFV_MTX_BB[4]	PFV_MTX_BB[3]	PFV_MTX_BB[2]	PFV_MTX_BB[1]	PFV_MTX_BB[0]	
		-	-	-	-	-	PFV_MTX_BR[10]	PFV_MTX_BR[9]	PFV_MTX_BR[8]	
		PFV_MTX_BR[7]	PFV_MTX_BR[6]	PFV_MTX_BR[5]	PFV_MTX_BR[4]	PFV_MTX_BR[3]	PFV_MTX_BR[2]	PFV_MTX_BR[1]	PFV_MTX_BR[0]	
	PFV_MTX_CRR_ADJ0	-	-	-	-	-	-	-	-	
		PFV_MTX_R[7]	PFV_MTX_R[6]	PFV_MTX_R[5]	PFV_MTX_R[4]	PFV_MTX_R[3]	PFV_MTX_R[2]	PFV_MTX_R[1]	PFV_MTX_R[0]	
		-	-	-	-	-	PFV_MTX_RG[10]	PFV_MTX_RG[9]	PFV_MTX_RG[8]	
		PFV_MTX_RG[7]	PFV_MTX_RG[6]	PFV_MTX_RG[5]	PFV_MTX_RG[4]	PFV_MTX_RG[3]	PFV_MTX_RG[2]	PFV_MTX_RG[1]	PFV_MTX_RG[0]	
	PFV_MTX_CRR_ADJ1	-	-	-	-	-	PFV_MTX_RB[10]	PFV_MTX_RB[9]	PFV_MTX_RB[8]	
		PFV_MTX_RB[7]	PFV_MTX_RB[6]	PFV_MTX_RB[5]	PFV_MTX_RB[4]	PFV_MTX_RB[3]	PFV_MTX_RB[2]	PFV_MTX_RB[1]	PFV_MTX_RB[0]	
		-	-	-	-	-	PFV_MTX_RR[10]	PFV_MTX_RR[9]	PFV_MTX_RR[8]	
		PFV_MTX_RR[7]	PFV_MTX_RR[6]	PFV_MTX_RR[5]	PFV_MTX_RR[4]	PFV_MTX_RR[3]	PFV_MTX_RR[2]	PFV_MTX_RR[1]	PFV_MTX_RR[0]	
	PFVSRZ	PFVSRZ[15]	PFVSRZ[14]	PFVSRZ[13]	PFVSRZ[12]	PFVSRZ[11]	PFVSRZ[10]	PFVSRZ[9]	PFVSRZ[8]	
		PFVSRZ[7]	PFVSRZ[6]	PFVSRZ[5]	PFVSRZ[4]	PFVSRZ[3]	PFVSRZ[2]	PFVSRZ[1]	PFVSRZ[0]	
		PFVSRZY[15]	PFVSRZY[14]	PFVSRZY[13]	PFVSRZY[12]	PFVSRZY[11]	PFVSRZY[10]	PFVSRZY[9]	PFVSRZY[8]	
		PFVSRZY[7]	PFVSRZY[6]	PFVSRZY[5]	PFVSRZY[4]	PFVSRZY[3]	PFVSRZY[2]	PFVSRZY[1]	PFVSRZY[0]	
	SCUX	IPICR_IPC0_0	-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	INIT
		IPSLR_IPC0_0	-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			-	-	-	-	-	IPC_PASS_SEL[2]	IPC_PASS_SEL[1]	IPC_PASS_SEL[0]
		IPICR_IPC0_1	-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	INIT
			-	-	-	-	-	-	-	-
		IPSLR_IPC0_1	-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			-	-	-	-	-	IPC_PASS_SEL[2]	IPC_PASS_SEL[1]	IPC_PASS_SEL[0]
		IPICR_IPC0_2	-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	-
			-	-	-	-	-	-	-	INIT

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
SCUX	IPSLR_IPC0_2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	IPC_PASS_SEL[2]	IPC_PASS_SEL[1]	IPC_PASS_SEL[0]
	IPCIR_IPC0_3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	INIT
	IPSLR_IPC0_3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	IPC_PASS_SEL[2]	IPC_PASS_SEL[1]	IPC_PASS_SEL[0]
	OPCIR_OPC0_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	INIT
	OPSLR_OPC0_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	OPC_PASS_SEL [2]	OPC_PASS_SEL [1]	OPC_PASS_SEL [0]
	OPCIR_OPC0_1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	INIT
	OPSLR_OPC0_1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	OPC_PASS_SEL [2]	OPC_PASS_SEL [1]	OPC_PASS_SEL [0]
	OPCIR_OPC0_2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	INIT
	OPSLR_OPC0_2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	OPC_PASS_SEL [2]	OPC_PASS_SEL [1]	OPC_PASS_SEL [0]
	OPCIR_OPC0_3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	INIT
	OPSLR_OPC0_3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	OPC_PASS_SEL [2]	OPC_PASS_SEL [1]	OPC_PASS_SEL [0]
	FFDIR_FFD0_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	INIT
	FDAIR_FFD0_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	CHNUM[3]	CHNUM[2]	CHNUM[1]

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
SCUX	DRQSR_FFD0_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	SIZE[3]	SIZE[2]	SIZE[1]	SIZE[0]
	FFDPR_FFD0_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	PASS[1]	PASS[0]
	FFDBR_FFD0_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	BOOT
	DEVMR_FFD0_0	DEVMUF	DEVMOF	DEVMOL	DEVMUIF	-	-	-	-
		-	-	-	-	-	-	-	-
		DEVMRQ	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	DEVCR_FFD0_0	DEVCUF	DEVCOF	DEVCOL	DEVCIUF	-	-	-	-
		-	-	-	-	-	-	-	-
		DEVCRQ	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	FFDIR_FFD0_1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	INIT
	FDAIR_FFD0_1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	CHNUM[3]	CHNUM[2]	CHNUM[1]
	DRQSR_FFD0_1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	SIZE[3]	SIZE[2]	SIZE[1]
	FFDPR_FFD0_1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	PASS[1]	PASS[0]
	FFDBR_FFD0_1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	BOOT
	DEVMR_FFD0_1	DEVMUF	DEVMOF	DEVMOL	DEVMUIF	-	-	-	-
		-	-	-	-	-	-	-	-
		DEVMRQ	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	DEVCR_FFD0_1	DEVCUF	DEVCOF	DEVCOL	DEVCIUF	-	-	-	-
		-	-	-	-	-	-	-	-
		DEVCRQ	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	FFDIR_FFD0_2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	INIT
	FDAIR_FFD0_2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	CHNUM[3]	CHNUM[2]	CHNUM[1]

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
SCUX	DRQSR_FFD0_2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	SIZE[3]	SIZE[2]	SIZE[1]	SIZE[0]
	FFDPR_FFD0_2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	PASS[1]	PASS[0]
	FFDBR_FFD0_2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	BOOT
	DEVMR_FFD0_2	DEVMUF	DEVMOF	DEVMOL	DEVMUIF	-	-	-	-
		-	-	-	-	-	-	-	-
		DEVMRQ	-	-	-	-	-	-	-
	DEVCR_FFD0_2	DEVCUF	DEVCOF	DEVCOL	DEVCIUF	-	-	-	-
		-	-	-	-	-	-	-	-
		DEVCRQ	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	FFDIR_FFD0_3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	INIT
	FDAIR_FFD0_3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	CHNUM[3]	CHNUM[2]	CHNUM[1]
	DRQSR_FFD0_3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	SIZE[3]	SIZE[2]	SIZE[1]	SIZE[0]
	FFDPR_FFD0_3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	PASS[1]	PASS[0]
	FFDBR_FFD0_3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	BOOT
	DEVMR_FFD0_3	DEVMUF	DEVMOF	DEVMOL	DEVMUIF	-	-	-	-
		-	-	-	-	-	-	-	-
		DEVMRQ	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	DEVCR_FFD0_3	DEVCUF	DEVCOF	DEVCOL	DEVCIUF	-	-	-	-
		-	-	-	-	-	-	-	-
		DEVCRQ	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	FFUIR_FFU0_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	INIT
	FUAIR_FFU0_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	CHNUM[3]	CHNUM[2]	CHNUM[1]

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
SCUX	URQSR_FFU0_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	SIZE[3]	SIZE[2]	SIZE[1]	SIZE[0]
	FFUPR_FFU0_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	PASS[1]	PASS[0]
	UEVMR_FFU0_0	UEVMUF	UEVMOF	UEVMOL	-	-	-	-	-
		-	-	-	-	-	-	-	-
		UEVMRQ	-	-	-	-	-	-	-
	UEVCR_FFU0_0	UEVCUF	UEVCOF	UEVCOL	-	-	-	-	-
		-	-	-	-	-	-	-	-
		UEVCRQ	-	-	-	-	-	-	-
	FFUIR_FFU0_1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	INIT
	FUAIR_FFU0_1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	CHNUM[3]	CHNUM[2]	CHNUM[1]
	URQSR_FFU0_1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	SIZE[3]	SIZE[2]	SIZE[1]
	FFUPR_FFU0_1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	PASS[1]	PASS[0]
	UEVMR_FFU0_1	UEVMUF	UEVMOF	UEVMOL	-	-	-	-	-
		-	-	-	-	-	-	-	-
		UEVMRQ	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	UEVCR_FFU0_1	UEVCUF	UEVCOF	UEVCOL	-	-	-	-	-
		-	-	-	-	-	-	-	-
		UEVCRQ	-	-	-	-	-	-	-
	FFUIR_FFU0_2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	INIT
	FUAIR_FFU0_2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	CHNUM[3]	CHNUM[2]	CHNUM[1]
	URQSR_FFU0_2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	SIZE[3]	SIZE[2]	SIZE[1]
	FFUPR_FFU0_2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	PASS[1]	PASS[0]



Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
SCUX	UEVMR_FFU0_2	UEVMUF	UEVMOF	UEVMOL	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		UEVMRQ	-	-	-	-	-	-	-	
	UEVCR_FFU0_2	UEVCUF	UEVCOF	UEVCOL	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		UEVCRQ	-	-	-	-	-	-	-	
	FFUIR_FFU0_3	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	INIT	
	FUAIR_FFU0_3	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	CHNUM[3]	CHNUM[2]	CHNUM[1]	CHNUM[0]	
	URQSR_FFU0_3	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	SIZE[3]	SIZE[2]	SIZE[1]	SIZE[0]	
	FFUPR_FFU0_3	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	PASS[1]	PASS[0]	
	UEVMR_FFU0_3	UEVMUF	UEVMOF	UEVMOL	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		UEVMRQ	-	-	-	-	-	-	-	
	UEVCR_FFU0_3	UEVCUF	UEVCOF	UEVCOL	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		UEVCRQ	-	-	-	-	-	-	-	
	SRCIR0_2SRC0_0	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	INIT	
	SADIR0_2SRC0_0	-	-	-	-	OTBL[4]	OTBL[3]	OTBL[2]	OTBL[1]	OTBL[0]
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	CHNUM[3]	CHNUM[2]	CHNUM[1]	CHNUM[0]
	SRCBR0_2SRC0_0	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	BYPASS
	IFSCR0_2SRC0_0	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	INTIFSEN
	IFSVR0_2SRC0_0	-	-	-	-	-	INTIFS[27]	INTIFS[26]	INTIFS[25]	INTIFS[24]
		INTIFS[23]	INTIFS[22]	INTIFS[21]	INTIFS[20]	INTIFS[19]	INTIFS[18]	INTIFS[17]	INTIFS[16]	-
		INTIFS[15]	INTIFS[14]	INTIFS[13]	INTIFS[11]	INTIFS[11]	INTIFS[10]	INTIFS[9]	INTIFS[8]	-
	SRCCR0_2SRC0_0	INTIFS[7]	INTIFS[6]	INTIFS[5]	INTIFS[4]	INTIFS[3]	INTIFS[2]	INTIFS[1]	INTIFS[0]	-
		-	-	-	-	-	-	-	-	-
		-	-	-	WATMD	-	-	-	-	-
	-	-	-	BUFMD	-	-	-	-	-	
	-	-	-	-	-	-	-	-	SRCMD	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
SCUX	MNFSR0_2SRC0_0	-	-	-	-	MINFS[27]	MINFS[26]	MINFS[25]	MINFS[24]	
		MINFS[23]	MINFS[22]	MINFS[21]	MINFS[20]	MINFS[19]	MINFS[18]	MINFS[17]	MINFS[16]	
		MINFS[15]	MINFS[14]	MINFS[13]	MINFS[11]	MINFS[11]	MINFS[10]	MINFS[9]	MINFS[8]	
		MINFS[7]	MINFS[6]	MINFS[5]	MINFS[4]	MINFS[3]	MINFS[2]	MINFS[1]	MINFS[0]	
	BFSSR0_2SRC0_0	-	-	-	-	-	-	-	BUFDATA[9]	BUFDATA[8]
		BUFDATA[7]	BUFDATA[6]	BUFDATA[5]	BUFDATA[4]	BUFDATA[3]	BUFDATA[1]	BUFDATA[1]	BUFDATA[0]	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	BUFIN[3]	BUFIN[2]	BUFIN[1]	BUFIN[0]	-
	SC2SR0_2SRC0_0	SRCWSTS	SC2MUTE	SC2STS[1]	SC2STS[0]	SC2FSI[27]	SC2FSI[26]	SC2FSI[25]	SC2FSI[24]	-
		SC2FSI[23]	SC2FSI[22]	SC2FSI[21]	SC2FSI[20]	SC2FSI[19]	SC2FSI[18]	SC2FSI[17]	SC2FSI[16]	-
		SC2FSI[15]	SC2FSI[14]	SC2FSI[13]	SC2FSI[11]	SC2FSI[11]	SC2FSI[10]	SC2FSI[9]	SC2FSI[8]	-
		SC2FSI[7]	SC2FSI[6]	SC2FSI[5]	SC2FSI[4]	SC2FSI[3]	SC2FSI[2]	SC2FSI[1]	SC2FSI[0]	-
	WATSR0_2SRC0_0	-	-	-	-	-	-	-	-	-
		WTIME[23]	WTIME[22]	WTIME[21]	WTIME[20]	WTIME[19]	WTIME[18]	WTIME[17]	WTIME[16]	-
		WTIME[15]	WTIME[14]	WTIME[13]	WTIME[11]	WTIME[11]	WTIME[10]	WTIME[9]	WTIME[8]	-
		WTIME[7]	WTIME[6]	WTIME[5]	WTIME[4]	WTIME[3]	WTIME[2]	WTIME[1]	WTIME[0]	-
	SEVMR0_2SRC0_0	EVMUF	EVMOF	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	EVMWAIT	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
	SEVCR0_2SRC0_0	EVCUF	EVCOF	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	EVCWAIT	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
	SRCIR1_2SRC0_0	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	INIT
	SADIR1_2SRC0_0	-	-	-	-	-	-	-	-	-
		-	-	-	OTBL[4]	OTBL[3]	OTBL[2]	OTBL[1]	OTBL[0]	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	CHNUM[3]	CHNUM[2]	CHNUM[1]	CHNUM[0]	-
	SRCBR1_2SRC0_0	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	BYPASS
	IFSCR1_2SRC0_0	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	INTIFSEN
	IFSVR1_2SRC0_0	-	-	-	-	-	INTIFS[27]	INTIFS[26]	INTIFS[25]	INTIFS[24]
		INTIFS[23]	INTIFS[22]	INTIFS[21]	INTIFS[20]	INTIFS[19]	INTIFS[18]	INTIFS[17]	INTIFS[16]	-
		INTIFS[15]	INTIFS[14]	INTIFS[13]	INTIFS[11]	INTIFS[11]	INTIFS[10]	INTIFS[9]	INTIFS[8]	-
		INTIFS[7]	INTIFS[6]	INTIFS[5]	INTIFS[4]	INTIFS[3]	INTIFS[2]	INTIFS[1]	INTIFS[0]	-
	SRCCR1_2SRC0_0	-	-	-	-	-	-	-	-	-
		-	-	-	WATMD	-	-	-	-	-
		-	-	-	BUFMD	-	-	-	-	-
		-	-	-	-	-	-	-	-	SRCMD
	MNFSR1_2SRC0_0	-	-	-	-	-	MINFS[27]	MINFS[26]	MINFS[25]	MINFS[24]
		MINFS[23]	MINFS[22]	MINFS[21]	MINFS[20]	MINFS[19]	MINFS[18]	MINFS[17]	MINFS[16]	-
		MINFS[15]	MINFS[14]	MINFS[13]	MINFS[11]	MINFS[11]	MINFS[10]	MINFS[9]	MINFS[8]	-
		MINFS[7]	MINFS[6]	MINFS[5]	MINFS[4]	MINFS[3]	MINFS[2]	MINFS[1]	MINFS[0]	-
	BFSSR1_2SRC0_0	-	-	-	-	-	-	-	BUFDATA[9]	BUFDATA[8]
		BUFDATA[7]	BUFDATA[6]	BUFDATA[5]	BUFDATA[4]	BUFDATA[3]	BUFDATA[1]	BUFDATA[1]	BUFDATA[0]	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	BUFIN[3]	BUFIN[2]	BUFIN[1]	BUFIN[0]	-

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
SCUX	SC2SR1_2SRC0_0	SRCWSTS	SC2MUTE	SC2STS[1]	SC2STS[0]	SC2FSI[27]	SC2FSI[26]	SC2FSI[25]	SC2FSI[24]	
		SC2FSI[23]	SC2FSI[22]	SC2FSI[21]	SC2FSI[20]	SC2FSI[19]	SC2FSI[18]	SC2FSI[17]	SC2FSI[16]	
		SC2FSI[15]	SC2FSI[14]	SC2FSI[13]	SC2FSI[11]	SC2FSI[11]	SC2FSI[10]	SC2FSI[9]	SC2FSI[8]	
		SC2FSI[7]	SC2FSI[6]	SC2FSI[5]	SC2FSI[4]	SC2FSI[3]	SC2FSI[2]	SC2FSI[1]	SC2FSI[0]	
	WATSR1_2SRC0_0	-	-	-	-	-	-	-	-	-
		WTIME[23]	WTIME[22]	WTIME[21]	WTIME[20]	WTIME[19]	WTIME[18]	WTIME[17]	WTIME[16]	-
		WTIME[15]	WTIME[14]	WTIME[13]	WTIME[11]	WTIME[11]	WTIME[10]	WTIME[9]	WTIME[8]	-
		WTIME[7]	WTIME[6]	WTIME[5]	WTIME[4]	WTIME[3]	WTIME[2]	WTIME[1]	WTIME[0]	-
	SEVMR1_2SRC0_0	EVMUF	EVMOF	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	EVMWAIT	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
	SEVCR1_2SRC0_0	EVCUF	EVCOF	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	EVCWAIT	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
	SRCIRR_2SRC0_0	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	INIT
	SRCIRO_2SRC0_1	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	INIT
	SADIR0_2SRC0_1	-	-	-	-	-	-	-	-	-
		-	-	-	-	OTBL[4]	OTBL[3]	OTBL[2]	OTBL[1]	OTBL[0]
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	CHNUM[3]	CHNUM[2]	CHNUM[1]	CHNUM[0]
	SRCBR0_2SRC0_1	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	BYPASS
	IFSCR0_2SRC0_1	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	INTIFSEN
	IFSVR0_2SRC0_1	-	-	-	-	-	INTIFS[27]	INTIFS[26]	INTIFS[25]	INTIFS[24]
		INTIFS[23]	INTIFS[22]	INTIFS[21]	INTIFS[20]	INTIFS[19]	INTIFS[18]	INTIFS[17]	INTIFS[16]	-
		INTIFS[15]	INTIFS[14]	INTIFS[13]	INTIFS[11]	INTIFS[11]	INTIFS[10]	INTIFS[9]	INTIFS[8]	-
		INTIFS[7]	INTIFS[6]	INTIFS[5]	INTIFS[4]	INTIFS[3]	INTIFS[2]	INTIFS[1]	INTIFS[0]	-
	SRCCR0_2SRC0_1	-	-	-	-	-	-	-	-	-
		-	-	-	-	WATMD	-	-	-	-
		-	-	-	-	BUFMD	-	-	-	-
		-	-	-	-	-	-	-	-	SRCMD
	MNFSR0_2SRC0_1	-	-	-	-	-	MINFS[27]	MINFS[26]	MINFS[25]	MINFS[24]
		MINFS[23]	MINFS[22]	MINFS[21]	MINFS[20]	MINFS[19]	MINFS[18]	MINFS[17]	MINFS[16]	-
		MINFS[15]	MINFS[14]	MINFS[13]	MINFS[11]	MINFS[11]	MINFS[10]	MINFS[9]	MINFS[8]	-
		MINFS[7]	MINFS[6]	MINFS[5]	MINFS[4]	MINFS[3]	MINFS[2]	MINFS[1]	MINFS[0]	-
	BFSSR0_2SRC0_1	-	-	-	-	-	-	-	BUFDATA[9]	BUFDATA[8]
		BUFDATA[7]	BUFDATA[6]	BUFDATA[5]	BUFDATA[4]	BUFDATA[3]	BUFDATA[1]	BUFDATA[1]	BUFDATA[0]	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	BUFIN[3]	BUFIN[2]	BUFIN[1]	BUFIN[0]	-
	SC2SR0_2SRC0_1	SRCWSTS	SC2MUTE	SC2STS[1]	SC2STS[0]	SC2FSI[27]	SC2FSI[26]	SC2FSI[25]	SC2FSI[24]	-
		SC2FSI[23]	SC2FSI[22]	SC2FSI[21]	SC2FSI[20]	SC2FSI[19]	SC2FSI[18]	SC2FSI[17]	SC2FSI[16]	-
		SC2FSI[15]	SC2FSI[14]	SC2FSI[13]	SC2FSI[11]	SC2FSI[11]	SC2FSI[10]	SC2FSI[9]	SC2FSI[8]	-
		SC2FSI[7]	SC2FSI[6]	SC2FSI[5]	SC2FSI[4]	SC2FSI[3]	SC2FSI[2]	SC2FSI[1]	SC2FSI[0]	-

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
SCUX	WATSR0_2SRC0_1	-	-	-	-	-	-	-	-	
		WTIME[23]	WTIME[22]	WTIME[21]	WTIME[20]	WTIME[19]	WTIME[18]	WTIME[17]	WTIME[16]	
		WTIME[15]	WTIME[14]	WTIME[13]	WTIME[11]	WTIME[11]	WTIME[10]	WTIME[9]	WTIME[8]	
		WTIME[7]	WTIME[6]	WTIME[5]	WTIME[4]	WTIME[3]	WTIME[2]	WTIME[1]	WTIME[0]	
	SEVMR0_2SRC0_1	EVMUF	EVMOF	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	EVMWAIT	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
	SEVCR0_2SRC0_1	EVCUF	EVCOF	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	EVCWAIT	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
	SRCIR1_2SRC0_1	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	INIT
	SADIR1_2SRC0_1	-	-	-	-	-	-	-	-	-
		-	-	-	OTBL[4]	OTBL[3]	OTBL[2]	OTBL[1]	OTBL[0]	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	CHNUM[3]	CHNUM[2]	CHNUM[1]	CHNUM[0]	-
	SRCBR1_2SRC0_1	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	BYPASS
	IFSCR1_2SRC0_1	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	INTIFSEN
	IFSVR1_2SRC0_1	-	-	-	-	INTIFS[27]	INTIFS[26]	INTIFS[25]	INTIFS[24]	-
		INTIFS[23]	INTIFS[22]	INTIFS[21]	INTIFS[20]	INTIFS[19]	INTIFS[18]	INTIFS[17]	INTIFS[16]	-
		INTIFS[15]	INTIFS[14]	INTIFS[13]	INTIFS[11]	INTIFS[11]	INTIFS[10]	INTIFS[9]	INTIFS[8]	-
		INTIFS[7]	INTIFS[6]	INTIFS[5]	INTIFS[4]	INTIFS[3]	INTIFS[2]	INTIFS[1]	INTIFS[0]	-
	SRCCR1_2SRC0_1	-	-	-	-	-	-	-	-	-
		-	-	-	WATMD	-	-	-	-	-
		-	-	-	BUFMD	-	-	-	-	-
		-	-	-	-	-	-	-	-	SRCMD
	MNFSR1_2SRC0_1	-	-	-	-	MINFS[27]	MINFS[26]	MINFS[25]	MINFS[24]	-
		MINFS[23]	MINFS[22]	MINFS[21]	MINFS[20]	MINFS[19]	MINFS[18]	MINFS[17]	MINFS[16]	-
		MINFS[15]	MINFS[14]	MINFS[13]	MINFS[11]	MINFS[11]	MINFS[10]	MINFS[9]	MINFS[8]	-
		MINFS[7]	MINFS[6]	MINFS[5]	MINFS[4]	MINFS[3]	MINFS[2]	MINFS[1]	MINFS[0]	-
	BFSSR1_2SRC0_1	-	-	-	-	-	-	-	BUFDATA[9]	BUFDATA[8]
		BUFDATA[7]	BUFDATA[6]	BUFDATA[5]	BUFDATA[4]	BUFDATA[3]	BUFDATA[1]	BUFDATA[1]	BUFDATA[0]	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	BUFIN[3]	BUFIN[2]	BUFIN[1]	BUFIN[0]	-
	SC2SR1_2SRC0_1	SC2WSTS	SC2MUTE	SC2STS[1]	SC2STS[0]	SC2FSI[27]	SC2FSI[26]	SC2FSI[25]	SC2FSI[24]	-
		SC2FSI[23]	SC2FSI[22]	SC2FSI[21]	SC2FSI[20]	SC2FSI[19]	SC2FSI[18]	SC2FSI[17]	SC2FSI[16]	-
		SC2FSI[15]	SC2FSI[14]	SC2FSI[13]	SC2FSI[11]	SC2FSI[11]	SC2FSI[10]	SC2FSI[9]	SC2FSI[8]	-
		SC2FSI[7]	SC2FSI[6]	SC2FSI[5]	SC2FSI[4]	SC2FSI[3]	SC2FSI[2]	SC2FSI[1]	SC2FSI[0]	-
	WATSR1_2SRC0_1	-	-	-	-	-	-	-	-	-
		WTIME[23]	WTIME[22]	WTIME[21]	WTIME[20]	WTIME[19]	WTIME[18]	WTIME[17]	WTIME[16]	-
		WTIME[15]	WTIME[14]	WTIME[13]	WTIME[11]	WTIME[11]	WTIME[10]	WTIME[9]	WTIME[8]	-
		WTIME[7]	WTIME[6]	WTIME[5]	WTIME[4]	WTIME[3]	WTIME[2]	WTIME[1]	WTIME[0]	-
	SEVMR1_2SRC0_1	EVMUF	EVMOF	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	EVMWAIT	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
SCUX	SEVCR1_2SRC0_1	EVCUF	EVCOF	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	EVCWAIT	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
	SRCIRR_2SRC0_1	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	INIT
	DVUIR_DVU0_0	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	INIT
	VADIR_DVU0_0	-	-	-	-	-	-	-	-	-
		-	-	-	-	OTBL[4]	OTBL[3]	OTBL[2]	OTBL[1]	OTBL[0]
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	CHNUM[3]	CHNUM[2]	CHNUM[1]	CHNUM[0]
	DVUJR_DVU0_0	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	BYPASS
	DVUCR_DVU0_0	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	VVMD
		-	-	-	-	VRMD	-	-	-	ZCMD
	ZCMCR_DVU0_0	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		ZCEN7	ZCEN6	ZCEN5	ZCEN4	ZCEN3	ZCEN2	ZCEN1	ZCEN0	
	VRCTR_DVU0_0	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		VREN7	VREN6	VREN5	VREN4	VREN3	VREN2	VREN1	VREN0	
	VRPDR_DVU0_0	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	VRPDUP[4]	VRPDUP[3]	VRPDUP[1]	VRPDUP[1]	VRPDUP[0]	
		-	-	-	VRPDDW[4]	VRPDDW[3]	VRPDDW[2]	VRPDDW[1]	VRPDDW[0]	
	VRDBR_DVU0_0	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	VRDB[9]	VRDB[8]	
		VRDB[7]	VRDB[6]	VRDB[5]	VRDB[4]	VRDB[3]	VRDB[2]	VRDB[1]	VRDB[0]	
	VRWTR_DVU0_0	-	-	-	-	-	-	-	-	-
		VRWT[23]	VRWT[22]	VRWT[21]	VRWT[20]	VRWT[19]	VRWT[18]	VRWT[17]	VRWT[16]	
		VRWT[15]	VRWT[14]	VRWT[13]	VRWT[11]	VRWT[11]	VRWT[10]	VRWT[9]	VRWT[8]	
		VRWT[7]	VRWT[6]	VRWT[5]	VRWT[4]	VRWT[3]	VRWT[2]	VRWT[1]	VRWT[0]	
	VOL0R_DVU0_0	-	-	-	-	-	-	-	-	-
		VOLVAL0[23]	VOLVAL0[22]	VOLVAL0[21]	VOLVAL0[20]	VOLVAL0[19]	VOLVAL0[18]	VOLVAL0[17]	VOLVAL0[16]	
		VOLVAL0[15]	VOLVAL0[14]	VOLVAL0[13]	VOLVAL0[11]	VOLVAL0[11]	VOLVAL0[10]	VOLVAL0[9]	VOLVAL0[8]	
		VOLVAL0[7]	VOLVAL0[6]	VOLVAL0[5]	VOLVAL0[4]	VOLVAL0[3]	VOLVAL0[2]	VOLVAL0[1]	VOLVAL0[0]	
	VOL1R_DVU0_0	-	-	-	-	-	-	-	-	-
		VOLVAL1[23]	VOLVAL1[22]	VOLVAL1[21]	VOLVAL1[20]	VOLVAL1[19]	VOLVAL1[18]	VOLVAL1[17]	VOLVAL1[16]	
		VOLVAL1[15]	VOLVAL1[14]	VOLVAL1[13]	VOLVAL1[11]	VOLVAL1[11]	VOLVAL1[10]	VOLVAL1[9]	VOLVAL1[8]	
		VOLVAL1[7]	VOLVAL1[6]	VOLVAL1[5]	VOLVAL1[4]	VOLVAL1[3]	VOLVAL1[2]	VOLVAL1[1]	VOLVAL1[0]	
	VOL2R_DVU0_0	-	-	-	-	-	-	-	-	-
		VOLVAL2[23]	VOLVAL2[22]	VOLVAL2[21]	VOLVAL2[20]	VOLVAL2[19]	VOLVAL2[18]	VOLVAL2[17]	VOLVAL2[16]	
		VOLVAL2[15]	VOLVAL2[14]	VOLVAL2[13]	VOLVAL2[11]	VOLVAL2[11]	VOLVAL2[10]	VOLVAL2[9]	VOLVAL2[8]	
		VOLVAL2[7]	VOLVAL2[6]	VOLVAL2[5]	VOLVAL2[4]	VOLVAL2[3]	VOLVAL2[2]	VOLVAL2[1]	VOLVAL2[0]	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
SCUX	VOL3R_DVU0_0	-	-	-	-	-	-	-	-	
		VOLVAL3[23]	VOLVAL3[22]	VOLVAL3[21]	VOLVAL3[20]	VOLVAL3[19]	VOLVAL3[18]	VOLVAL3[17]	VOLVAL3[16]	
		VOLVAL3[15]	VOLVAL3[14]	VOLVAL3[13]	VOLVAL3[11]	VOLVAL3[11]	VOLVAL3[10]	VOLVAL3[9]	VOLVAL3[8]	
		VOLVAL3[7]	VOLVAL3[6]	VOLVAL3[5]	VOLVAL3[4]	VOLVAL3[3]	VOLVAL3[2]	VOLVAL3[1]	VOLVAL3[0]	
	VOL4R_DVU0_0	-	-	-	-	-	-	-	-	-
		VOLVAL4[23]	VOLVAL4[22]	VOLVAL4[21]	VOLVAL4[20]	VOLVAL4[19]	VOLVAL4[18]	VOLVAL4[17]	VOLVAL4[16]	
		VOLVAL4[15]	VOLVAL4[14]	VOLVAL4[13]	VOLVAL4[11]	VOLVAL4[11]	VOLVAL4[10]	VOLVAL4[9]	VOLVAL4[8]	
		VOLVAL4[7]	VOLVAL4[6]	VOLVAL4[5]	VOLVAL4[4]	VOLVAL4[3]	VOLVAL4[2]	VOLVAL4[1]	VOLVAL4[0]	
	VOL5R_DVU0_0	-	-	-	-	-	-	-	-	-
		VOLVAL5[23]	VOLVAL5[22]	VOLVAL5[21]	VOLVAL5[20]	VOLVAL5[19]	VOLVAL5[18]	VOLVAL5[17]	VOLVAL5[16]	
		VOLVAL5[15]	VOLVAL5[14]	VOLVAL5[13]	VOLVAL5[11]	VOLVAL5[11]	VOLVAL5[10]	VOLVAL5[9]	VOLVAL5[8]	
		VOLVAL5[7]	VOLVAL5[6]	VOLVAL5[5]	VOLVAL5[4]	VOLVAL5[3]	VOLVAL5[2]	VOLVAL5[1]	VOLVAL5[0]	
	VOL6R_DVU0_0	-	-	-	-	-	-	-	-	-
		VOLVAL6[23]	VOLVAL6[22]	VOLVAL6[21]	VOLVAL6[20]	VOLVAL6[19]	VOLVAL6[18]	VOLVAL6[17]	VOLVAL6[16]	
		VOLVAL6[15]	VOLVAL6[14]	VOLVAL6[13]	VOLVAL6[11]	VOLVAL6[11]	VOLVAL6[10]	VOLVAL6[9]	VOLVAL6[8]	
		VOLVAL6[7]	VOLVAL6[6]	VOLVAL6[5]	VOLVAL6[4]	VOLVAL6[3]	VOLVAL6[2]	VOLVAL6[1]	VOLVAL6[0]	
	VOL7R_DVU0_0	-	-	-	-	-	-	-	-	-
		VOLVAL7[23]	VOLVAL7[22]	VOLVAL7[21]	VOLVAL7[20]	VOLVAL7[19]	VOLVAL7[18]	VOLVAL7[17]	VOLVAL7[16]	
		VOLVAL7[15]	VOLVAL7[14]	VOLVAL7[13]	VOLVAL7[11]	VOLVAL7[11]	VOLVAL7[10]	VOLVAL7[9]	VOLVAL7[8]	
		VOLVAL7[7]	VOLVAL7[6]	VOLVAL7[5]	VOLVAL7[4]	VOLVAL7[3]	VOLVAL7[2]	VOLVAL7[1]	VOLVAL7[0]	
	DVUER_DVU0_0	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	DVUEN
	DVUSR_DVU0_0	-	-	-	-	-	-	-	-	-
		ZSTS7	ZSTS6	ZSTS5	ZSTS4	ZSTS3	ZSTS2	ZSTS1	ZSTS0	
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	VRSTS[2]	VRSTS[1]	VRSTS[0]	
	VEVMR_DVU0_0	VEVMZCM7	VEVMZCM6	VEVMZCM5	VEVMZCM4	VEVMZCM3	VEVMZCM2	VEVMZCM1	VEVMZCM0	
		-	-	-	-	-	-	-	-	
		VEVMVR	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
	VEVCR_DVU0_0	VEVCZCM7	VEVCZCM6	VEVCZCM5	VEVCZCM4	VEVCZCM3	VEVCZCM2	VEVCZCM1	VEVCZCM0	
		-	-	-	-	-	-	-	-	
		VEVCVR	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
	DVUIR_DVU0_1	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	INIT	
	VADIR_DVU0_1	-	-	-	-	-	-	-	-	
		-	-	-	OTBL[4]	OTBL[3]	OTBL[2]	OTBL[1]	OTBL[0]	
		-	-	-	-	-	-	-	-	
		-	-	-	-	CHNUM[3]	CHNUM[2]	CHNUM[1]	CHNUM[0]	
	DVUBR_DVU0_1	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	BYPASS	
	DVUCR_DVU0_1	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	VVMD	
		-	-	-	VRMD	-	-	-	ZCMD	
	ZCMCR_DVU0_1	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		ZCEN7	ZCEN6	ZCEN5	ZCEN4	ZCEN3	ZCEN2	ZCEN1	ZCEN0	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
SCUX	VRCTR_DVU0_1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		VREN7	VREN6	VREN5	VREN4	VREN3	VREN2	VREN1	VREN0
	VRPDR_DVU0_1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	VRPDUP[4]	VRPDUP[3]	VRPDUP[1]	VRPDUP[1]	VRPDUP[0]
		-	-	-	VRPDDW[4]	VRPDDW[3]	VRPDDW[2]	VRPDDW[1]	VRPDDW[0]
	VRDBR_DVU0_1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	VRDB[9]	VRDB[8]
		VRDB[7]	VRDB[6]	VRDB[5]	VRDB[4]	VRDB[3]	VRDB[2]	VRDB[1]	VRDB[0]
	VRWTR_DVU0_1	-	-	-	-	-	-	-	-
		VRWT[23]	VRWT[22]	VRWT[21]	VRWT[20]	VRWT[19]	VRWT[18]	VRWT[17]	VRWT[16]
		VRWT[15]	VRWT[14]	VRWT[13]	VRWT[11]	VRWT[11]	VRWT[10]	VRWT[9]	VRWT[8]
		VRWT[7]	VRWT[6]	VRWT[5]	VRWT[4]	VRWT[3]	VRWT[2]	VRWT[1]	VRWT[0]
	VOL0R_DVU0_1	-	-	-	-	-	-	-	-
		VOLVAL0[23]	VOLVAL0[22]	VOLVAL0[21]	VOLVAL0[20]	VOLVAL0[19]	VOLVAL0[18]	VOLVAL0[17]	VOLVAL0[16]
		VOLVAL0[15]	VOLVAL0[14]	VOLVAL0[13]	VOLVAL0[11]	VOLVAL0[11]	VOLVAL0[10]	VOLVAL0[9]	VOLVAL0[8]
		VOLVAL0[7]	VOLVAL0[6]	VOLVAL0[5]	VOLVAL0[4]	VOLVAL0[3]	VOLVAL0[2]	VOLVAL0[1]	VOLVAL0[0]
	VOL1R_DVU0_1	-	-	-	-	-	-	-	-
		VOLVAL1[23]	VOLVAL1[22]	VOLVAL1[21]	VOLVAL1[20]	VOLVAL1[19]	VOLVAL1[18]	VOLVAL1[17]	VOLVAL1[16]
		VOLVAL1[15]	VOLVAL1[14]	VOLVAL1[13]	VOLVAL1[11]	VOLVAL1[11]	VOLVAL1[10]	VOLVAL1[9]	VOLVAL1[8]
		VOLVAL1[7]	VOLVAL1[6]	VOLVAL1[5]	VOLVAL1[4]	VOLVAL1[3]	VOLVAL1[2]	VOLVAL1[1]	VOLVAL1[0]
	VOL2R_DVU0_1	-	-	-	-	-	-	-	-
		VOLVAL2[23]	VOLVAL2[22]	VOLVAL2[21]	VOLVAL2[20]	VOLVAL2[19]	VOLVAL2[18]	VOLVAL2[17]	VOLVAL2[16]
		VOLVAL2[15]	VOLVAL2[14]	VOLVAL2[13]	VOLVAL2[11]	VOLVAL2[11]	VOLVAL2[10]	VOLVAL2[9]	VOLVAL2[8]
		VOLVAL2[7]	VOLVAL2[6]	VOLVAL2[5]	VOLVAL2[4]	VOLVAL2[3]	VOLVAL2[2]	VOLVAL2[1]	VOLVAL2[0]
	VOL3R_DVU0_1	-	-	-	-	-	-	-	-
		VOLVAL3[23]	VOLVAL3[22]	VOLVAL3[21]	VOLVAL3[20]	VOLVAL3[19]	VOLVAL3[18]	VOLVAL3[17]	VOLVAL3[16]
		VOLVAL3[15]	VOLVAL3[14]	VOLVAL3[13]	VOLVAL3[11]	VOLVAL3[11]	VOLVAL3[10]	VOLVAL3[9]	VOLVAL3[8]
		VOLVAL3[7]	VOLVAL3[6]	VOLVAL3[5]	VOLVAL3[4]	VOLVAL3[3]	VOLVAL3[2]	VOLVAL3[1]	VOLVAL3[0]
	VOL4R_DVU0_1	-	-	-	-	-	-	-	-
		VOLVAL4[23]	VOLVAL4[22]	VOLVAL4[21]	VOLVAL4[20]	VOLVAL4[19]	VOLVAL4[18]	VOLVAL4[17]	VOLVAL4[16]
		VOLVAL4[15]	VOLVAL4[14]	VOLVAL4[13]	VOLVAL4[11]	VOLVAL4[11]	VOLVAL4[10]	VOLVAL4[9]	VOLVAL4[8]
		VOLVAL4[7]	VOLVAL4[6]	VOLVAL4[5]	VOLVAL4[4]	VOLVAL4[3]	VOLVAL4[2]	VOLVAL4[1]	VOLVAL4[0]
	VOL5R_DVU0_1	-	-	-	-	-	-	-	-
		VOLVAL5[23]	VOLVAL5[22]	VOLVAL5[21]	VOLVAL5[20]	VOLVAL5[19]	VOLVAL5[18]	VOLVAL5[17]	VOLVAL5[16]
		VOLVAL5[15]	VOLVAL5[14]	VOLVAL5[13]	VOLVAL5[11]	VOLVAL5[11]	VOLVAL5[10]	VOLVAL5[9]	VOLVAL5[8]
		VOLVAL5[7]	VOLVAL5[6]	VOLVAL5[5]	VOLVAL5[4]	VOLVAL5[3]	VOLVAL5[2]	VOLVAL5[1]	VOLVAL5[0]
	VOL6R_DVU0_1	-	-	-	-	-	-	-	-
		VOLVAL6[23]	VOLVAL6[22]	VOLVAL6[21]	VOLVAL6[20]	VOLVAL6[19]	VOLVAL6[18]	VOLVAL6[17]	VOLVAL6[16]
		VOLVAL6[15]	VOLVAL6[14]	VOLVAL6[13]	VOLVAL6[11]	VOLVAL6[11]	VOLVAL6[10]	VOLVAL6[9]	VOLVAL6[8]
		VOLVAL6[7]	VOLVAL6[6]	VOLVAL6[5]	VOLVAL6[4]	VOLVAL6[3]	VOLVAL6[2]	VOLVAL6[1]	VOLVAL6[0]
	VOL7R_DVU0_1	-	-	-	-	-	-	-	-
		VOLVAL7[23]	VOLVAL7[22]	VOLVAL7[21]	VOLVAL7[20]	VOLVAL7[19]	VOLVAL7[18]	VOLVAL7[17]	VOLVAL7[16]
		VOLVAL7[15]	VOLVAL7[14]	VOLVAL7[13]	VOLVAL7[11]	VOLVAL7[11]	VOLVAL7[10]	VOLVAL7[9]	VOLVAL7[8]
		VOLVAL7[7]	VOLVAL7[6]	VOLVAL7[5]	VOLVAL7[4]	VOLVAL7[3]	VOLVAL7[2]	VOLVAL7[1]	VOLVAL7[0]
	DVUER_DVU0_1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	DVUEN
	DVUSR_DVU0_1	-	-	-	-	-	-	-	-
		ZSTS7	ZSTS6	ZSTS5	ZSTS4	ZSTS3	ZSTS2	ZSTS1	ZSTS0
		-	-	-	-	-	-	-	-
		-	-	-	-	-	VRSTS[2]	VRSTS[1]	VRSTS[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
SCUX	VEVMR_DVU0_1	VEVMZCM7	VEVMZCM6	VEVMZCM5	VEVMZCM4	VEVMZCM3	VEVMZCM2	VEVMZCM1	VEVMZCM0
		-	-	-	-	-	-	-	-
		VEVMVR	-	-	-	-	-	-	-
	VEVCV_DVU0_1	VEVCZCM7	VEVCZCM6	VEVCZCM5	VEVCZCM4	VEVCZCM3	VEVCZCM2	VEVCZCM1	VEVCZCM0
		-	-	-	-	-	-	-	-
		VEVCVR	-	-	-	-	-	-	-
	DVUIR_DVU0_2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	INIT
	VADIR_DVU0_2	-	-	-	-	-	-	-	-
		-	-	-	OTBL[4]	OTBL[3]	OTBL[2]	OTBL[1]	OTBL[0]
		-	-	-	-	CHNUM[3]	CHNUM[2]	CHNUM[1]	CHNUM[0]
	DVUBR_DVU0_2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	BYPASS
	DVUCR_DVU0_2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	VVMD
		-	-	-	VRMD	-	-	-	ZCMD
	ZCMCR_DVU0_2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		ZCEN7	ZCEN6	ZCEN5	ZCEN4	ZCEN3	ZCEN2	ZCEN1	ZCEN0
	VRCTR_DVU0_2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		VREN7	VREN6	VREN5	VREN4	VREN3	VREN2	VREN1	VREN0
	VRPDR_DVU0_2	-	-	-	-	-	-	-	-
		-	-	-	VRPDUP[4]	VRPDUP[3]	VRPDUP[1]	VRPDUP[1]	VRPDUP[0]
		-	-	-	VRPDDW[4]	VRPDDW[3]	VRPDDW[2]	VRPDDW[1]	VRPDDW[0]
	VRDBR_DVU0_2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		VRDB[7]	VRDB[6]	VRDB[5]	VRDB[4]	VRDB[3]	VRDB[2]	VRDB[1]	VRDB[0]
	VRWTR_DVU0_2	-	-	-	-	-	-	-	-
		VRWT[23]	VRWT[22]	VRWT[21]	VRWT[20]	VRWT[19]	VRWT[18]	VRWT[17]	VRWT[16]
		VRWT[15]	VRWT[14]	VRWT[13]	VRWT[11]	VRWT[11]	VRWT[10]	VRWT[9]	VRWT[8]
	VOL0R_DVU0_2	-	-	-	-	-	-	-	-
		VOLVAL0[23]	VOLVAL0[22]	VOLVAL0[21]	VOLVAL0[20]	VOLVAL0[19]	VOLVAL0[18]	VOLVAL0[17]	VOLVAL0[16]
		VOLVAL0[15]	VOLVAL0[14]	VOLVAL0[13]	VOLVAL0[11]	VOLVAL0[11]	VOLVAL0[10]	VOLVAL0[9]	VOLVAL0[8]
	VOL1R_DVU0_2	-	-	-	-	-	-	-	-
		VOLVAL1[23]	VOLVAL1[22]	VOLVAL1[21]	VOLVAL1[20]	VOLVAL1[19]	VOLVAL1[18]	VOLVAL1[17]	VOLVAL1[16]
		VOLVAL1[15]	VOLVAL1[14]	VOLVAL1[13]	VOLVAL1[11]	VOLVAL1[11]	VOLVAL1[10]	VOLVAL1[9]	VOLVAL1[8]
	VOL2R_DVU0_2	-	-	-	-	-	-	-	-
		VOLVAL2[23]	VOLVAL2[22]	VOLVAL2[21]	VOLVAL2[20]	VOLVAL2[19]	VOLVAL2[18]	VOLVAL2[17]	VOLVAL2[16]
		VOLVAL2[15]	VOLVAL2[14]	VOLVAL2[13]	VOLVAL2[11]	VOLVAL2[11]	VOLVAL2[10]	VOLVAL2[9]	VOLVAL2[8]
		-	-	-	-	-	-	-	-
		VOLVAL2[7]	VOLVAL2[6]	VOLVAL2[5]	VOLVAL2[4]	VOLVAL2[3]	VOLVAL2[2]	VOLVAL2[1]	VOLVAL2[0]



Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
SCUX	VOL3R_DVU0_2	-	-	-	-	-	-	-	-
		VOLVAL3[23]	VOLVAL3[22]	VOLVAL3[21]	VOLVAL3[20]	VOLVAL3[19]	VOLVAL3[18]	VOLVAL3[17]	VOLVAL3[16]
		VOLVAL3[15]	VOLVAL3[14]	VOLVAL3[13]	VOLVAL3[11]	VOLVAL3[11]	VOLVAL3[10]	VOLVAL3[9]	VOLVAL3[8]
		VOLVAL3[7]	VOLVAL3[6]	VOLVAL3[5]	VOLVAL3[4]	VOLVAL3[3]	VOLVAL3[2]	VOLVAL3[1]	VOLVAL3[0]
	VOL4R_DVU0_2	-	-	-	-	-	-	-	-
		VOLVAL4[23]	VOLVAL4[22]	VOLVAL4[21]	VOLVAL4[20]	VOLVAL4[19]	VOLVAL4[18]	VOLVAL4[17]	VOLVAL4[16]
		VOLVAL4[15]	VOLVAL4[14]	VOLVAL4[13]	VOLVAL4[11]	VOLVAL4[11]	VOLVAL4[10]	VOLVAL4[9]	VOLVAL4[8]
		VOLVAL4[7]	VOLVAL4[6]	VOLVAL4[5]	VOLVAL4[4]	VOLVAL4[3]	VOLVAL4[2]	VOLVAL4[1]	VOLVAL4[0]
	VOL5R_DVU0_2	-	-	-	-	-	-	-	-
		VOLVAL5[23]	VOLVAL5[22]	VOLVAL5[21]	VOLVAL5[20]	VOLVAL5[19]	VOLVAL5[18]	VOLVAL5[17]	VOLVAL5[16]
		VOLVAL5[15]	VOLVAL5[14]	VOLVAL5[13]	VOLVAL5[11]	VOLVAL5[11]	VOLVAL5[10]	VOLVAL5[9]	VOLVAL5[8]
		VOLVAL5[7]	VOLVAL5[6]	VOLVAL5[5]	VOLVAL5[4]	VOLVAL5[3]	VOLVAL5[2]	VOLVAL5[1]	VOLVAL5[0]
	VOL6R_DVU0_2	-	-	-	-	-	-	-	-
		VOLVAL6[23]	VOLVAL6[22]	VOLVAL6[21]	VOLVAL6[20]	VOLVAL6[19]	VOLVAL6[18]	VOLVAL6[17]	VOLVAL6[16]
		VOLVAL6[15]	VOLVAL6[14]	VOLVAL6[13]	VOLVAL6[11]	VOLVAL6[11]	VOLVAL6[10]	VOLVAL6[9]	VOLVAL6[8]
		VOLVAL6[7]	VOLVAL6[6]	VOLVAL6[5]	VOLVAL6[4]	VOLVAL6[3]	VOLVAL6[2]	VOLVAL6[1]	VOLVAL6[0]
	VOL7R_DVU0_2	-	-	-	-	-	-	-	-
		VOLVAL7[23]	VOLVAL7[22]	VOLVAL7[21]	VOLVAL7[20]	VOLVAL7[19]	VOLVAL7[18]	VOLVAL7[17]	VOLVAL7[16]
		VOLVAL7[15]	VOLVAL7[14]	VOLVAL7[13]	VOLVAL7[11]	VOLVAL7[11]	VOLVAL7[10]	VOLVAL7[9]	VOLVAL7[8]
		VOLVAL7[7]	VOLVAL7[6]	VOLVAL7[5]	VOLVAL7[4]	VOLVAL7[3]	VOLVAL7[2]	VOLVAL7[1]	VOLVAL7[0]
	DVUER_DVU0_2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	DVUEN
	DVUSR_DVU0_2	-	-	-	-	-	-	-	-
		ZSTS7	ZSTS6	ZSTS5	ZSTS4	ZSTS3	ZSTS2	ZSTS1	ZSTS0
		-	-	-	-	-	-	-	-
		-	-	-	-	-	VRSTS[2]	VRSTS[1]	VRSTS[0]
	VEVMR_DVU0_2	VEVMZCM7	VEVMZCM6	VEVMZCM5	VEVMZCM4	VEVMZCM3	VEVMZCM2	VEVMZCM1	VEVMZCM0
		-	-	-	-	-	-	-	-
		VEVMVR	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	VEVCR_DVU0_2	VEVCZCM7	VEVCZCM6	VEVCZCM5	VEVCZCM4	VEVCZCM3	VEVCZCM2	VEVCZCM1	VEVCZCM0
		-	-	-	-	-	-	-	-
		VEVCVR	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	DVUIR_DVU0_3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	INIT
	VADIR_DVU0_3	-	-	-	-	-	-	-	-
		-	-	-	OTBL[4]	OTBL[3]	OTBL[2]	OTBL[1]	OTBL[0]
		-	-	-	-	-	-	-	-
		-	-	-	-	CHNUM[3]	CHNUM[2]	CHNUM[1]	CHNUM[0]
	DVUBR_DVU0_3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	BYPASS
	DVUCR_DVU0_3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	VVMD
		-	-	-	VRMD	-	-	-	ZCMD
	ZCMCR_DVU0_3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		ZCEN7	ZCEN6	ZCEN5	ZCEN4	ZCEN3	ZCEN2	ZCEN1	ZCEN0

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
SCUX	VRCTR_DVU0_3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		VREN7	VREN6	VREN5	VREN4	VREN3	VREN2	VREN1	VREN0
	VRPDR_DVU0_3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	VRPDUP[4]	VRPDUP[3]	VRPDUP[1]	VRPDUP[1]	VRPDUP[0]
		-	-	-	VRPDDW[4]	VRPDDW[3]	VRPDDW[2]	VRPDDW[1]	VRPDDW[0]
	VRDBR_DVU0_3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	VRDB[9]	VRDB[8]
		VRDB[7]	VRDB[6]	VRDB[5]	VRDB[4]	VRDB[3]	VRDB[2]	VRDB[1]	VRDB[0]
	VRWTR_DVU0_3	-	-	-	-	-	-	-	-
		VRWT[23]	VRWT[22]	VRWT[21]	VRWT[20]	VRWT[19]	VRWT[18]	VRWT[17]	VRWT[16]
		VRWT[15]	VRWT[14]	VRWT[13]	VRWT[11]	VRWT[11]	VRWT[10]	VRWT[9]	VRWT[8]
		VRWT[7]	VRWT[6]	VRWT[5]	VRWT[4]	VRWT[3]	VRWT[2]	VRWT[1]	VRWT[0]
	VOL0R_DVU0_3	-	-	-	-	-	-	-	-
		VOLVAL0[23]	VOLVAL0[22]	VOLVAL0[21]	VOLVAL0[20]	VOLVAL0[19]	VOLVAL0[18]	VOLVAL0[17]	VOLVAL0[16]
		VOLVAL0[15]	VOLVAL0[14]	VOLVAL0[13]	VOLVAL0[11]	VOLVAL0[11]	VOLVAL0[10]	VOLVAL0[9]	VOLVAL0[8]
		VOLVAL0[7]	VOLVAL0[6]	VOLVAL0[5]	VOLVAL0[4]	VOLVAL0[3]	VOLVAL0[2]	VOLVAL0[1]	VOLVAL0[0]
	VOL1R_DVU0_3	-	-	-	-	-	-	-	-
		VOLVAL1[23]	VOLVAL1[22]	VOLVAL1[21]	VOLVAL1[20]	VOLVAL1[19]	VOLVAL1[18]	VOLVAL1[17]	VOLVAL1[16]
		VOLVAL1[15]	VOLVAL1[14]	VOLVAL1[13]	VOLVAL1[11]	VOLVAL1[11]	VOLVAL1[10]	VOLVAL1[9]	VOLVAL1[8]
		VOLVAL1[7]	VOLVAL1[6]	VOLVAL1[5]	VOLVAL1[4]	VOLVAL1[3]	VOLVAL1[2]	VOLVAL1[1]	VOLVAL1[0]
	VOL2R_DVU0_3	-	-	-	-	-	-	-	-
		VOLVAL2[23]	VOLVAL2[22]	VOLVAL2[21]	VOLVAL2[20]	VOLVAL2[19]	VOLVAL2[18]	VOLVAL2[17]	VOLVAL2[16]
		VOLVAL2[15]	VOLVAL2[14]	VOLVAL2[13]	VOLVAL2[11]	VOLVAL2[11]	VOLVAL2[10]	VOLVAL2[9]	VOLVAL2[8]
		VOLVAL2[7]	VOLVAL2[6]	VOLVAL2[5]	VOLVAL2[4]	VOLVAL2[3]	VOLVAL2[2]	VOLVAL2[1]	VOLVAL2[0]
	VOL3R_DVU0_3	-	-	-	-	-	-	-	-
		VOLVAL3[23]	VOLVAL3[22]	VOLVAL3[21]	VOLVAL3[20]	VOLVAL3[19]	VOLVAL3[18]	VOLVAL3[17]	VOLVAL3[16]
		VOLVAL3[15]	VOLVAL3[14]	VOLVAL3[13]	VOLVAL3[11]	VOLVAL3[11]	VOLVAL3[10]	VOLVAL3[9]	VOLVAL3[8]
		VOLVAL3[7]	VOLVAL3[6]	VOLVAL3[5]	VOLVAL3[4]	VOLVAL3[3]	VOLVAL3[2]	VOLVAL3[1]	VOLVAL3[0]
	VOL4R_DVU0_3	-	-	-	-	-	-	-	-
		VOLVAL4[23]	VOLVAL4[22]	VOLVAL4[21]	VOLVAL4[20]	VOLVAL4[19]	VOLVAL4[18]	VOLVAL4[17]	VOLVAL4[16]
		VOLVAL4[15]	VOLVAL4[14]	VOLVAL4[13]	VOLVAL4[11]	VOLVAL4[11]	VOLVAL4[10]	VOLVAL4[9]	VOLVAL4[8]
		VOLVAL4[7]	VOLVAL4[6]	VOLVAL4[5]	VOLVAL4[4]	VOLVAL4[3]	VOLVAL4[2]	VOLVAL4[1]	VOLVAL4[0]
	VOL5R_DVU0_3	-	-	-	-	-	-	-	-
		VOLVAL5[23]	VOLVAL5[22]	VOLVAL5[21]	VOLVAL5[20]	VOLVAL5[19]	VOLVAL5[18]	VOLVAL5[17]	VOLVAL5[16]
		VOLVAL5[15]	VOLVAL5[14]	VOLVAL5[13]	VOLVAL5[11]	VOLVAL5[11]	VOLVAL5[10]	VOLVAL5[9]	VOLVAL5[8]
		VOLVAL5[7]	VOLVAL5[6]	VOLVAL5[5]	VOLVAL5[4]	VOLVAL5[3]	VOLVAL5[2]	VOLVAL5[1]	VOLVAL5[0]
	VOL6R_DVU0_3	-	-	-	-	-	-	-	-
		VOLVAL6[23]	VOLVAL6[22]	VOLVAL6[21]	VOLVAL6[20]	VOLVAL6[19]	VOLVAL6[18]	VOLVAL6[17]	VOLVAL6[16]
		VOLVAL6[15]	VOLVAL6[14]	VOLVAL6[13]	VOLVAL6[11]	VOLVAL6[11]	VOLVAL6[10]	VOLVAL6[9]	VOLVAL6[8]
		VOLVAL6[7]	VOLVAL6[6]	VOLVAL6[5]	VOLVAL6[4]	VOLVAL6[3]	VOLVAL6[2]	VOLVAL6[1]	VOLVAL6[0]
	VOL7R_DVU0_3	-	-	-	-	-	-	-	-
		VOLVAL7[23]	VOLVAL7[22]	VOLVAL7[21]	VOLVAL7[20]	VOLVAL7[19]	VOLVAL7[18]	VOLVAL7[17]	VOLVAL7[16]
		VOLVAL7[15]	VOLVAL7[14]	VOLVAL7[13]	VOLVAL7[11]	VOLVAL7[11]	VOLVAL7[10]	VOLVAL7[9]	VOLVAL7[8]
		VOLVAL7[7]	VOLVAL7[6]	VOLVAL7[5]	VOLVAL7[4]	VOLVAL7[3]	VOLVAL7[2]	VOLVAL7[1]	VOLVAL7[0]
	DVUER_DVU0_3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	DVUEN
	DVUSR_DVU0_3	-	-	-	-	-	-	-	-
		ZSTS7	ZSTS6	ZSTS5	ZSTS4	ZSTS3	ZSTS2	ZSTS1	ZSTS0
		-	-	-	-	-	-	-	-
		-	-	-	-	-	VRSTS[2]	VRSTS[1]	VRSTS[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
SCUX	VEVMR_DVU0_3	VEVMZCM7	VEVMZCM6	VEVMZCM5	VEVMZCM4	VEVMZCM3	VEVMZCM2	VEVMZCM1	VEVMZCM0
		-	-	-	-	-	-	-	-
		VEVMVR	-	-	-	-	-	-	-
	VEVCR_DVU0_3	VEVCZCM7	VEVCZCM6	VEVCZCM5	VEVCZCM4	VEVCZCM3	VEVCZCM2	VEVCZCM1	VEVCZCM0
		-	-	-	-	-	-	-	-
		VEVCVR	-	-	-	-	-	-	-
	MIXIR_MIX0_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	INIT
	MADIR_MIX0_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	CHNUM[3]	CHNUM[2]	CHNUM[1]	CHNUM[0]
	MIXBR_MIX0_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	BPSYS[1]	BPSYS[0]
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	BYPASS
	MIXMR_MIX0_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	MIXMODE
	MVPDR_MIX0_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	MXPDUP[3]	MXPDUP[2]	MXPDUP[1]	MXPDUP[0]
		-	-	-	-	MXPDDW[3]	MXPDDW[2]	MXPDDW[1]	MXPDDW[0]
	MDBAR_MIX0_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	MIXDBA[9]	MIXDBA[8]
		MIXDBA[7]	MIXDBA[6]	MIXDBA[5]	MIXDBA[4]	MIXDBA[3]	MIXDBA[2]	MIXDBA[1]	MIXDBA[0]
	MDBBR_MIX0_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	MIXDBB[9]	MIXDBB[8]
		MIXDBB[7]	MIXDBB[6]	MIXDBB[5]	MIXDBB[4]	MIXDBB[3]	MIXDBB[2]	MIXDBB[1]	MIXDBB[0]
	MDBCR_MIX0_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	MIXDBC[9]	MIXDBC[8]
		MIXDBC[7]	MIXDBC[6]	MIXDBC[5]	MIXDBC[4]	MIXDBC[3]	MIXDBC[2]	MIXDBC[1]	MIXDBC[0]
	MDBDR_MIX0_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	MIXDBD[9]	MIXDBD[8]
		MIXDBD[7]	MIXDBD[6]	MIXDBD[5]	MIXDBD[4]	MIXDBD[3]	MIXDBD[2]	MIXDBD[1]	MIXDBD[0]
	MDBER_MIX0_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	MIXDBEN
	MIXSR_MIX0_0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	MIXSTS[1]	MIXSTS[0]
	SWRSR_CIM	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	SWRST

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
SCUX	DMACR_CIM	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		DMAMDFFU3	DMAMDFFU2	DMAMDFFU1	DMAMDFFU0	DMAMDFFD3	DMAMDFFD2	DMAMDFFD1	DMAMDFFD0
	DMATD0_CIM								
	DMATD1_CIM								
	DMATD2_CIM								
	DMATD3_CIM								
	DMATU0_CIM								
	DMATU1_CIM								
	DMATU2_CIM								
	DMATU3_CIM								
	SSIRSEL_CIM	SISEL3[1]	SISEL3[0]	SISEL2[1]	SISEL2[0]	SISEL1[1]	SISEL1[0]	SISEL0[1]	SISEL0[0]
		-	-	SOSEL5[1]	SOSEL5[0]	-	-	SOSEL4[1]	SOSEL4[0]
		-	-	SOSEL3[1]	SOSEL3[0]	-	-	SOSEL2[1]	SOSEL2[0]
		-	-	SOSEL1[1]	SOSEL1[0]	-	-	SOSEL0[1]	SOSEL0[0]
	FDTSEL0_CIM	-	-	-	-	-	SCKDIV[11]	SCKDIV[10]	SCKDIV[9]
		SCKDIV[8]	SCKDIV[7]	SCKDIV[6]	SCKDIV[5]	SCKDIV[4]	SCKDIV[3]	SCKDIV[2]	SCKDIV[1]
		SCKDIV[0]	-	-	-	-	-	-	DIVEN
		-	-	-	MTUSEL	SCKSEL[3]	SCKSEL[2]	SCKSEL[1]	SCKSEL[0]
	FDTSEL1_CIM	-	-	-	-	-	SCKDIV[10]	SCKDIV[9]	SCKDIV[8]
		SCKDIV[7]	SCKDIV[6]	SCKDIV[5]	SCKDIV[4]	SCKDIV[3]	SCKDIV[2]	SCKDIV[1]	SCKDIV[0]
		-	-	-	-	-	-	-	DIVEN
		-	-	-	MTUSEL	SCKSEL[3]	SCKSEL[2]	SCKSEL[1]	SCKSEL[0]
	FDTSEL2_CIM	-	-	-	-	-	SCKDIV[10]	SCKDIV[9]	SCKDIV[8]
		SCKDIV[7]	SCKDIV[6]	SCKDIV[5]	SCKDIV[4]	SCKDIV[3]	SCKDIV[2]	SCKDIV[1]	SCKDIV[0]
		-	-	-	-	-	-	-	DIVEN
		-	-	-	MTUSEL	SCKSEL[3]	SCKSEL[2]	SCKSEL[1]	SCKSEL[0]
	FDTSEL3_CIM	-	-	-	-	-	SCKDIV[10]	SCKDIV[9]	SCKDIV[8]
		SCKDIV[7]	SCKDIV[6]	SCKDIV[5]	SCKDIV[4]	SCKDIV[3]	SCKDIV[2]	SCKDIV[1]	SCKDIV[0]
		-	-	-	-	-	-	-	DIVEN
		-	-	-	MTUSEL	SCKSEL[3]	SCKSEL[2]	SCKSEL[1]	SCKSEL[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
SCUX	FUTSEL0_CIM	-	-	-	-	-	SCKDIV[10]	SCKDIV[9]	SCKDIV[8]	
		SCKDIV[7]	SCKDIV[6]	SCKDIV[5]	SCKDIV[4]	SCKDIV[3]	SCKDIV[2]	SCKDIV[1]	SCKDIV[0]	
		-	-	-	-	-	-	-	DIVEN	
		-	-	-	MTUSEL	SCKSEL[3]	SCKSEL[2]	SCKSEL[1]	SCKSEL[0]	
	FUTSEL1_CIM	-	-	-	-	-	-	SCKDIV[10]	SCKDIV[9]	SCKDIV[8]
		SCKDIV[7]	SCKDIV[6]	SCKDIV[5]	SCKDIV[4]	SCKDIV[3]	SCKDIV[2]	SCKDIV[1]	SCKDIV[0]	
		-	-	-	-	-	-	-	DIVEN	
		-	-	-	MTUSEL	SCKSEL[3]	SCKSEL[2]	SCKSEL[1]	SCKSEL[0]	
	FUTSEL2_CIM	-	-	-	-	-	-	SCKDIV[10]	SCKDIV[9]	SCKDIV[8]
		SCKDIV[7]	SCKDIV[6]	SCKDIV[5]	SCKDIV[4]	SCKDIV[3]	SCKDIV[2]	SCKDIV[1]	SCKDIV[0]	
		-	-	-	-	-	-	-	DIVEN	
		-	-	-	MTUSEL	SCKSEL[3]	SCKSEL[2]	SCKSEL[1]	SCKSEL[0]	
	FUTSEL3_CIM	-	-	-	-	-	-	SCKDIV[10]	SCKDIV[9]	SCKDIV[8]
		SCKDIV[7]	SCKDIV[6]	SCKDIV[5]	SCKDIV[4]	SCKDIV[3]	SCKDIV[2]	SCKDIV[1]	SCKDIV[0]	
		-	-	-	-	-	-	-	DIVEN	
		-	-	-	MTUSEL	SCKSEL[3]	SCKSEL[2]	SCKSEL[1]	SCKSEL[0]	
	SSIPMD_CIM	-	-	-	-	-	-	-	-	-
		-	-	SSI5CKS	SSI4CKS	SSI3CKS	SSI2CKS	SSI1CKS	SSI0CKS	-
		SSI3PMD[1]	SSI3PMD[0]	-	SSI345EN	SSI4PMD[1]	SSI4PMD[0]	SSI5PMD[1]	SSI5PMD[0]	-
		-	-	-	SSI012EN	SSI2PMD[1]	SSI2PMD[0]	SSI1PMD[1]	SSI1PMD[0]	-
	SSICTRL_CIM	-	SSI3TX	SSI4TX	SSI5TX	-	-	SSI3RX	SSI4RX	SSI5RX
		-	-	-	-	-	-	-	SSI345TEN	SSI345REN
		-	SSI0TX	SSI1TX	SSI2TX	-	-	SSI0RX	SSI1RX	SSI2RX
		-	-	-	-	-	-	-	SSI012TEN	SSI012REN
	SRCRSEL0_CIM	-	PLACE7[2]	PLACE7[1]	PLACE7[0]	-	-	PLACE6[2]	PLACE6[1]	PLACE6[0]
		-	PLACE5[2]	PLACE5[1]	PLACE5[0]	-	-	PLACE4[2]	PLACE4[1]	PLACE4[0]
		-	PLACE3[2]	PLACE3[1]	PLACE3[0]	-	-	PLACE2[2]	PLACE2[1]	PLACE2[0]
		-	PLACE1[2]	PLACE1[1]	PLACE1[0]	-	-	PLACE0[2]	PLACE0[1]	PLACE0[0]
	SRCRSEL1_CIM	-	PLACE7[2]	PLACE7[1]	PLACE7[0]	-	-	PLACE6[2]	PLACE6[1]	PLACE6[0]
		-	PLACE5[2]	PLACE5[1]	PLACE5[0]	-	-	PLACE4[2]	PLACE4[1]	PLACE4[0]
		-	PLACE3[2]	PLACE3[1]	PLACE3[0]	-	-	PLACE2[2]	PLACE2[1]	PLACE2[0]
		-	PLACE1[2]	PLACE1[1]	PLACE1[0]	-	-	PLACE0[2]	PLACE0[1]	PLACE0[0]
	SRCRSEL2_CIM	-	PLACE7[2]	PLACE7[1]	PLACE7[0]	-	-	PLACE6[2]	PLACE6[1]	PLACE6[0]
		-	PLACE5[2]	PLACE5[1]	PLACE5[0]	-	-	PLACE4[2]	PLACE4[1]	PLACE4[0]
		-	PLACE3[2]	PLACE3[1]	PLACE3[0]	-	-	PLACE2[2]	PLACE2[1]	PLACE2[0]
		-	PLACE1[2]	PLACE1[1]	PLACE1[0]	-	-	PLACE0[2]	PLACE0[1]	PLACE0[0]
	SRCRSEL3_CIM	-	PLACE7[2]	PLACE7[1]	PLACE7[0]	-	-	PLACE6[2]	PLACE6[1]	PLACE6[0]
		-	PLACE5[2]	PLACE5[1]	PLACE5[0]	-	-	PLACE4[2]	PLACE4[1]	PLACE4[0]
		-	PLACE3[2]	PLACE3[1]	PLACE3[0]	-	-	PLACE2[2]	PLACE2[1]	PLACE2[0]
		-	PLACE1[2]	PLACE1[1]	PLACE1[0]	-	-	PLACE0[2]	PLACE0[1]	PLACE0[0]
	MIXRSEL_CIM	-	PLACE7[2]	PLACE7[1]	PLACE7[0]	-	-	PLACE6[2]	PLACE6[1]	PLACE6[0]
		-	PLACE5[2]	PLACE5[1]	PLACE5[0]	-	-	PLACE4[2]	PLACE4[1]	PLACE4[0]
		-	PLACE3[2]	PLACE3[1]	PLACE3[0]	-	-	PLACE2[2]	PLACE2[1]	PLACE2[0]
		-	PLACE1[2]	PLACE1[1]	PLACE1[0]	-	-	PLACE0[2]	PLACE0[1]	PLACE0[0]
	Sound generator	SGCR1_0	SGST	STPM	-	SGCK[1]	SGCK[0]	DPF[2]	DPF[1]	DPF[0]
		SGCSR_0	SGIE	SGDEF	-	-	-	-	-	-
		SGCR2_0	SGEND	TCHG	-	-	-	-	-	-
		SGLR_0	LD[7]	LD[6]	LD[5]	LD[4]	LD[3]	LD[2]	LD[1]	LD[0]
		SGTFR_0	-	TONE[6]	TONE[5]	TONE[4]	TONE[3]	TONE[2]	TONE[1]	TONE[0]
		SGSFR_0	SFS[7]	SFS[6]	SFS[5]	SFS[4]	SFS[3]	SFS[2]	SFS[1]	SFS[0]
		SGCR1_1	SGST	STPM	-	SGCK[1]	SGCK[0]	DPF[2]	DPF[1]	DPF[0]
		SGCSR_1	SGIE	SGDEF	-	-	-	-	-	-
		SGCR2_1	SGEND	TCHG	-	-	-	-	-	-
		SGLR_1	LD[7]	LD[6]	LD[5]	LD[4]	LD[3]	LD[2]	LD[1]	LD[0]
		SGTFR_1	-	TONE[6]	TONE[5]	TONE[4]	TONE[3]	TONE[2]	TONE[1]	TONE[0]
		SGSFR_1	SFS[7]	SFS[6]	SFS[5]	SFS[4]	SFS[3]	SFS[2]	SFS[1]	SFS[0]
		SGCR1_2	SGST	STPM	-	SGCK[1]	SGCK[0]	DPF[2]	DPF[1]	DPF[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Sound generator	SGCSR_2	SGIE	SGDEF	-	-	-	-	-	-
	SGCR2_2	SGEND	TCHG	-	-	-	-	-	-
	SGLR_2	LD[7]	LD[6]	LD[5]	LD[4]	LD[3]	LD[2]	LD[1]	LD[0]
	SGTFR_2	-	STONE[6]	STONE[5]	STONE[4]	STONE[3]	STONE[2]	STONE[1]	STONE[0]
	SGSFR_2	SFS[7]	SFS[6]	SFS[5]	SFS[4]	SFS[3]	SFS[2]	SFS[1]	SFS[0]
	SGCR1_3	SGST	STPM	-	SGCK[1]	SGCK[0]	DPF[2]	DPF[1]	DPF[0]
	SGCSR_3	SGIE	SGDEF	-	-	-	-	-	-
	SGCR2_3	SGEND	TCHG	-	-	-	-	-	-
	SGLR_3	LD[7]	LD[6]	LD[5]	LD[4]	LD[3]	LD[2]	LD[1]	LD[0]
	SGTFR_3	-	STONE[6]	STONE[5]	STONE[4]	STONE[3]	STONE[2]	STONE[1]	STONE[0]
SGSFR_3	SFS[7]	SFS[6]	SFS[5]	SFS[4]	SFS[3]	SFS[2]	SFS[1]	SFS[0]	
MMC host interface	CE_CMD_SET	-	-	CMD[5]	CMD[4]	CMD[3]	CMD[2]	CMD[1]	CMD[0]
		RTYP[1]	RTYP[0]	RBSY	-	WDAT	DWEN	CMLTE	CMD12EN
		RIDXC[1]	RIDXC[0]	RCRC7C[1]	RCRC7C[0]	-	-	-	CRCSTE
		TBIT	OPDM	-	-	SBIT	-	DATW[1]	DATW[0]
	CE_ARG	ARG[31]	ARG[30]	ARG[29]	ARG[28]	ARG[27]	ARG[26]	ARG[25]	ARG[24]
		ARG[23]	ARG[22]	ARG[21]	ARG[20]	ARG[19]	ARG[18]	ARG[17]	ARG[16]
		ARG[15]	ARG[14]	ARG[13]	ARG[12]	ARG[11]	ARG[10]	ARG[9]	ARG[8]
		ARG[7]	ARG[6]	ARG[5]	ARG[4]	ARG[3]	ARG[2]	ARG[1]	ARG[0]
	CE_ARG_CMD12	C12ARG[31]	C12ARG[30]	C12ARG[29]	C12ARG[28]	C12ARG[27]	C12ARG[26]	C12ARG[25]	C12ARG[24]
		C12ARG[23]	C12ARG[22]	C12ARG[21]	C12ARG[20]	C12ARG[19]	C12ARG[18]	C12ARG[17]	C12ARG[16]
		C12ARG[15]	C12ARG[14]	C12ARG[13]	C12ARG[12]	C12ARG[11]	C12ARG[10]	C12ARG[9]	C12ARG[8]
		C12ARG[7]	C12ARG[6]	C12ARG[5]	C12ARG[4]	C12ARG[3]	C12ARG[2]	C12ARG[1]	C12ARG[0]
	CE_CMD_CTRL	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	BREAK
	CE_BLOCK_SET	BLKCNT[15]	BLKCNT[14]	BLKCNT[13]	BLKCNT[12]	BLKCNT[11]	BLKCNT[10]	BLKCNT[9]	BLKCNT[8]
		BLKCNT[7]	BLKCNT[6]	BLKCNT[5]	BLKCNT[4]	BLKCNT[3]	BLKCNT[2]	BLKCNT[1]	BLKCNT[0]
		BLKSIZ[15]	BLKSIZ[14]	BLKSIZ[13]	BLKSIZ[12]	BLKSIZ[11]	BLKSIZ[10]	BLKSIZ[9]	BLKSIZ[8]
		BLKSIZ[7]	BLKSIZ[6]	BLKSIZ[5]	BLKSIZ[4]	BLKSIZ[3]	BLKSIZ[2]	BLKSIZ[1]	BLKSIZ[0]
	CE_CLK_CTRL	-	-	-	-	-	-	-	CLKEN
		-	-	-	-	CLKDIV[3]	CLKDIV[2]	CLKDIV[1]	CLKDIV[0]
		-	-	SRSPTO[1]	SRSPTO[0]	SRBSYTO[3]	SRBSYTO[2]	SRBSYTO[1]	SRBSYTO[0]
		SRWDTO[3]	SRWDTO[2]	SRWDTO[1]	SRWDTO[0]	-	-	-	-
	CE_BUF_ACC	-	-	-	-	-	-	DMAWEN	DMAREN
		-	-	-	-	-	-	BUSW	ATYP
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	CE_RESP3	RSP[127]	RSP[126]	RSP[125]	RSP[124]	RSP[123]	RSP[122]	RSP[121]	RSP[120]
		RSP[119]	RSP[118]	RSP[117]	RSP[116]	RSP[115]	RSP[114]	RSP[113]	RSP[112]
		RSP[111]	RSP[110]	RSP[109]	RSP[108]	RSP[107]	RSP[106]	RSP[105]	RSP[104]
		RSP[103]	RSP[102]	RSP[101]	RSP[100]	RSP[99]	RSP[98]	RSP[97]	RSP[96]
	CE_RESP2	RSP[95]	RSP[94]	RSP[93]	RSP[92]	RSP[91]	RSP[90]	RSP[89]	RSP[88]
		RSP[87]	RSP[86]	RSP[85]	RSP[84]	RSP[83]	RSP[82]	RSP[81]	RSP[80]
		RSP[79]	RSP[78]	RSP[77]	RSP[76]	RSP[75]	RSP[74]	RSP[73]	RSP[72]
		RSP[71]	RSP[70]	RSP[69]	RSP[68]	RSP[67]	RSP[66]	RSP[65]	RSP[64]
	CE_RESP1	RSP[63]	RSP[62]	RSP[61]	RSP[60]	RSP[59]	RSP[58]	RSP[57]	RSP[56]
		RSP[55]	RSP[54]	RSP[53]	RSP[52]	RSP[51]	RSP[50]	RSP[49]	RSP[48]
		RSP[47]	RSP[46]	RSP[45]	RSP[44]	RSP[43]	RSP[42]	RSP[41]	RSP[40]
	CE_RESP0	RSP[39]	RSP[38]	RSP[37]	RSP[36]	RSP[35]	RSP[34]	RSP[33]	RSP[32]
		RSP[31]	RSP[30]	RSP[29]	RSP[28]	RSP[27]	RSP[26]	RSP[25]	RSP[24]
		RSP[23]	RSP[22]	RSP[21]	RSP[20]	RSP[19]	RSP[18]	RSP[17]	RSP[16]
		RSP[15]	RSP[14]	RSP[13]	RSP[12]	RSP[11]	RSP[10]	RSP[9]	RSP[8]
		RSP[7]	RSP[6]	RSP[5]	RSP[4]	RSP[3]	RSP[2]	RSP[1]	RSP[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
MMC host interface	CE_RESP_CMD12	RSP12[31]	RSP12[30]	RSP12[29]	RSP12[28]	RSP12[27]	RSP12[26]	RSP12[25]	RSP12[24]	
		RSP12[23]	RSP12[22]	RSP12[21]	RSP12[20]	RSP12[19]	RSP12[18]	RSP12[17]	RSP12[16]	
		RSP12[15]	RSP12[14]	RSP12[13]	RSP12[12]	RSP12[11]	RSP12[10]	RSP12[9]	RSP12[8]	
		RSP12[7]	RSP12[6]	RSP12[5]	RSP12[4]	RSP12[3]	RSP12[2]	RSP12[1]	RSP12[0]	
	CE_DATA	DATA[31]	DATA[30]	DATA[29]	DATA[28]	DATA[27]	DATA[26]	DATA[25]	DATA[24]	
		DATA[23]	DATA[22]	DATA[21]	DATA[20]	DATA[19]	DATA[18]	DATA[17]	DATA[16]	
		DATA[15]	DATA[14]	DATA[13]	DATA[12]	DATA[11]	DATA[10]	DATA[9]	DATA[8]	
		DATA[7]	DATA[6]	DATA[5]	DATA[4]	DATA[3]	DATA[2]	DATA[1]	DATA[0]	
	CE_INT	-	-	-	-	-	-	CMD12DRE	CMD12RBE	CMD12CRE
		DTRANE	BUFRE	BUFVEN	BUFREN	-	-	-	RBSYE	CRSPE
		CMDVIO	BUFVIO	-	-	WDATERR	RDATERR	RIDXERR	RSPERR	-
		-	-	-	CRCSTO	WDATTO	RDATTO	RBSYTO	RSPTO	-
	CE_INT_EN	-	-	-	-	-	-	MCMD12DRE	MCMD12RBE	MCMD12CRE
		MDTRANE	MBUFRE	MBUFVEN	MBUFREN	-	-	-	MRBSYE	MCRSPE
		MCMDVIO	MBUFVIO	-	-	MWDATERR	MRDATERR	MRIDXERR	MRSPEERR	-
		-	-	-	MCRCSO	MWDATTO	MRDATTO	MRBSYTO	MRSPTO	-
	CE_HOST_STS1	CMDSEQ	CMDSIG	RSPIDX[5]	RSPIDX[4]	RSPIDX[3]	RSPIDX[2]	RSPIDX[1]	RSPIDX[0]	-
		DATSIG[7]	DATSIG[6]	DATSIG[5]	DATSIG[4]	DATSIG[3]	DATSIG[2]	DATSIG[1]	DATSIG[0]	-
		RCVBLK[15]	RCVBLK[14]	RCVBLK[13]	RCVBLK[12]	RCVBLK[11]	RCVBLK[10]	RCVBLK[9]	RCVBLK[8]	-
		RCVBLK[7]	RCVBLK[6]	RCVBLK[5]	RCVBLK[4]	RCVBLK[3]	RCVBLK[2]	RCVBLK[1]	RCVBLK[0]	-
	CE_HOST_STS2	CRCSTE	CRC16E	AC12CRCE	RSPCRC7E	CRCSTEBE	RDATEBE	AC12REBE	RSPPEBE	-
		AC12DXE	RSPIDXE	-	-	-	CRCST[2]	CRCST[1]	CRCST[0]	-
		-	STRDATTO	DATBSYTO	CRCSTTO	AC12BSYTO	RSPBSYTO	AC12RSPTO	STRSPTO	-
		-	-	-	-	-	-	-	-	-
	CE_DMA_MODE	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	DMASEL
	CE_DETECT	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	CDSIG	CDRISE	CDFALL	-	-	-	-	-
		-	-	MCDRISE	MCDFALL	-	-	-	-	-
	CE_ADD_MODE	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	CLKMAIN	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
	CE_VERSION	SWRST	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		VERSION[15]	VERSION[14]	VERSION[13]	VERSION[12]	VERSION[11]	VERSION[10]	VERSION[9]	VERSION[8]	-
		VERSION[7]	VERSION[6]	VERSION[5]	VERSION[4]	VERSION[3]	VERSION[2]	VERSION[1]	VERSION[0]	-
	Motor control PWM timer	PWCR_1	-	-	IE	CMF	CST	CKS2	CKS1	CKS0
		PWPR_1	OPS1H	OPS1G	OPS1F	OPS1E	OPS1D	OPS1C	OPS1B	OPS1A
		PWCYR_1	PWCY15	PWCY14	PWCY13	PWCY12	PWCY11	PWCY10	PWCY9	PWCY8
			PWCY7	PWCY6	PWCY5	PWCY4	PWCY3	PWCY2	PWCY1	PWCY0
		PWBFR_1A	-	-	-	OTS	-	-	DT9	DT8
			DT7	DT6	DT5	DT4	DT3	DT2	DT1	DT0
		PWBFR_1C	-	-	-	OTS	-	-	DT9	DT8
			DT7	DT6	DT5	DT4	DT3	DT2	DT1	DT0
		PWBFR_1E	-	-	-	OTS	-	-	DT9	DT8
			DT7	DT6	DT5	DT4	DT3	DT2	DT1	DT0
		PWBFR_1G	-	-	-	OTS	-	-	DT9	DT8
			DT7	DT6	DT5	DT4	DT3	DT2	DT1	DT0
PWCR_2		-	-	IE	CMF	CST	CKS2	CKS1	CKS0	
PWPR_2		OPS2H	OPS2G	OPS2F	OPS2E	OPS2D	OPS2C	OPS2B	OPS2A	
PWCYR_2		PWCY15	PWCY14	PWCY13	PWCY12	PWCY11	PWCY10	PWCY9	PWCY8	
		PWCY7	PWCY6	PWCY5	PWCY4	PWCY3	PWCY2	PWCY1	PWCY0	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Motor control PWM timer	PWBFR_2A	-	-	-	OTS	-	-	DT9	DT8	
		DT7	DT6	DT5	DT4	DT3	DT2	DT1	DT0	
	PWBFR_2C	-	-	-	OTS	-	-	DT9	DT8	
		DT7	DT6	DT5	DT4	DT3	DT2	DT1	DT0	
	PWBFR_2E	-	-	-	OTS	-	-	DT9	DT8	
		DT7	DT6	DT5	DT4	DT3	DT2	DT1	DT0	
	PWBFR_2G	-	-	-	OTS	-	-	DT9	DT8	
		DT7	DT6	DT5	DT4	DT3	DT2	DT1	DT0	
	PWBTCR	-	-	-	-	-	-	-	-	-
		BTC2G	BTC2E	BTC2C	BTC2A	BTC1G	BTC1E	BTC1C	BTC1A	
Ports	P1	-	-	-	-	-	-	-	-	
		P1[7]	P1[6]	P1[5]	P1[4]	P1[3]	P1[2]	P1[1]	P1[0]	
	P2	P2[15]	P2[14]	P2[13]	P2[12]	P2[11]	P2[10]	P2[9]	P2[8]	
		P2[7]	P2[6]	P2[5]	P2[4]	P2[3]	P2[2]	P2[1]	P2[0]	
	P3	P3[15]	P3[14]	P3[13]	P3[12]	P3[11]	P3[10]	P3[9]	P3[8]	
		P3[7]	P3[6]	P3[5]	P3[4]	P3[3]	P3[2]	P3[1]	P3[0]	
	P4	P4[15]	P4[14]	P4[13]	P4[12]	P4[11]	P4[10]	P4[9]	P4[8]	
		P4[7]	P4[6]	P4[5]	P4[4]	P4[3]	P4[2]	P4[1]	P4[0]	
	P5	-	-	-	-	-	P5[10]	P5[9]	P5[8]	
		P5[7]	P5[6]	P5[5]	P5[4]	P5[3]	P5[2]	P5[1]	P5[0]	
	P6	P6[15]	P6[14]	P6[13]	P6[12]	P6[11]	P6[10]	P6[9]	P6[8]	
		P6[7]	P6[6]	P6[5]	P6[4]	P6[3]	P6[2]	P6[1]	P6[0]	
	P7	P7[15]	P7[14]	P7[13]	P7[12]	P7[11]	P7[10]	P7[9]	P7[8]	
		P7[7]	P7[6]	P7[5]	P7[4]	P7[3]	P7[2]	P7[1]	P7[0]	
	P8	P8[15]	P8[14]	P8[13]	P8[12]	P8[11]	P8[10]	P8[9]	P8[8]	
		P8[7]	P8[6]	P8[5]	P8[4]	P8[3]	P8[2]	P8[1]	P8[0]	
	P9	-	-	-	-	-	-	-	-	
		P9[7]	P9[6]	P9[5]	P9[4]	P9[3]	P9[2]	P9[1]	P9[0]	
	P10	P10[15]	P10[14]	P10[13]	P10[12]	P10[11]	P10[10]	P10[9]	P10[8]	
		P10[7]	P10[6]	P10[5]	P10[4]	P10[3]	P10[2]	P10[1]	P10[0]	
	P11	P11[15]	P11[14]	P11[13]	P11[12]	P11[11]	P11[10]	P11[9]	P11[8]	
		P11[7]	P11[6]	P11[5]	P11[4]	P11[3]	P11[2]	P11[1]	P11[0]	
	PSR1	-	-	-	-	-	-	-	-	
		PSR1[23]	PSR1[22]	PSR1[21]	PSR1[20]	PSR1[19]	PSR1[18]	PSR1[17]	PSR1[16]	
		-	-	-	-	-	-	-	-	
		PSR1[7]	PSR1[6]	PSR1[5]	PSR1[4]	PSR1[3]	PSR1[2]	PSR1[1]	PSR1[0]	
	PSR2	PSR2[31]	PSR2[30]	PSR2[29]	PSR2[28]	PSR2[27]	PSR2[26]	PSR2[25]	PSR2[24]	
		PSR2[23]	PSR2[22]	PSR2[21]	PSR2[20]	PSR2[19]	PSR2[18]	PSR2[17]	PSR2[16]	
		PSR2[15]	PSR2[14]	PSR2[13]	PSR2[12]	PSR2[11]	PSR2[10]	PSR2[9]	PSR2[8]	
		PSR2[7]	PSR2[6]	PSR2[5]	PSR2[4]	PSR2[3]	PSR2[2]	PSR2[1]	PSR2[0]	
	PSR3	PSR3[31]	PSR3[30]	PSR3[29]	PSR3[28]	PSR3[27]	PSR3[26]	PSR3[25]	PSR3[24]	
		PSR3[23]	PSR3[22]	PSR3[21]	PSR3[20]	PSR3[19]	PSR3[18]	PSR3[17]	PSR3[16]	
		PSR3[15]	PSR3[14]	PSR3[13]	PSR3[12]	PSR3[11]	PSR3[10]	PSR3[9]	PSR3[8]	
		PSR3[7]	PSR3[6]	PSR3[5]	PSR3[4]	PSR3[3]	PSR3[2]	PSR3[1]	PSR3[0]	
	PSR4	PSR4[31]	PSR4[30]	PSR4[29]	PSR4[28]	PSR4[27]	PSR4[26]	PSR4[25]	PSR4[24]	
		PSR4[23]	PSR4[22]	PSR4[21]	PSR4[20]	PSR4[19]	PSR4[18]	PSR4[17]	PSR4[16]	
		PSR4[15]	PSR4[14]	PSR4[13]	PSR4[12]	PSR4[11]	PSR4[10]	PSR4[9]	PSR4[8]	
		PSR4[7]	PSR4[6]	PSR4[5]	PSR4[4]	PSR4[3]	PSR4[2]	PSR4[1]	PSR4[0]	
	PSR5	-	-	-	-	-	PSR5[26]	PSR5[25]	PSR5[24]	
		PSR5[23]	PSR5[22]	PSR5[21]	PSR5[20]	PSR5[19]	PSR5[18]	PSR5[17]	PSR5[16]	
		-	-	-	-	-	PSR5[10]	PSR5[9]	PSR5[8]	
		PSR5[7]	PSR5[6]	PSR5[5]	PSR5[4]	PSR5[3]	PSR5[2]	PSR5[1]	PSR5[0]	
	PSR6	PSR6[31]	PSR6[30]	PSR6[29]	PSR6[28]	PSR6[27]	PSR6[26]	PSR6[25]	PSR6[24]	
		PSR6[23]	PSR6[22]	PSR6[21]	PSR6[20]	PSR6[19]	PSR6[18]	PSR6[17]	PSR6[16]	
		PSR6[15]	PSR6[14]	PSR6[13]	PSR6[12]	PSR6[11]	PSR6[10]	PSR6[9]	PSR6[8]	
		PSR6[7]	PSR6[6]	PSR6[5]	PSR6[4]	PSR6[3]	PSR6[2]	PSR6[1]	PSR6[0]	



Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Ports	PSR7	PSR7[31]	PSR7[30]	PSR7[29]	PSR7[28]	PSR7[27]	PSR7[26]	PSR7[25]	PSR7[24]
		PSR7[23]	PSR7[22]	PSR7[21]	PSR7[20]	PSR7[19]	PSR7[18]	PSR7[17]	PSR7[16]
		PSR7[15]	PSR7[14]	PSR7[13]	PSR7[12]	PSR7[11]	PSR7[10]	PSR7[9]	PSR7[8]
		PSR7[7]	PSR7[6]	PSR7[5]	PSR7[4]	PSR7[3]	PSR7[2]	PSR7[1]	PSR7[0]
	PSR8	PSR8[31]	PSR8[30]	PSR8[29]	PSR8[28]	PSR8[27]	PSR8[26]	PSR8[25]	PSR8[24]
		PSR8[23]	PSR8[22]	PSR8[21]	PSR8[20]	PSR8[19]	PSR8[18]	PSR8[17]	PSR8[16]
		PSR8[15]	PSR8[14]	PSR8[13]	PSR8[12]	PSR8[11]	PSR8[10]	PSR8[9]	PSR8[8]
		PSR8[7]	PSR8[6]	PSR8[5]	PSR8[4]	PSR8[3]	PSR8[2]	PSR8[1]	PSR8[0]
	PSR9	-	-	-	-	-	-	-	-
		PSR9[23]	PSR9[22]	PSR9[21]	PSR9[20]	PSR9[19]	PSR9[18]	PSR9[17]	PSR9[16]
		-	-	-	-	-	-	-	-
		PSR9[7]	PSR9[6]	PSR9[5]	PSR9[4]	PSR9[3]	PSR9[2]	PSR9[1]	PSR9[0]
	PSR10	PSR10[31]	PSR10[30]	PSR10[29]	PSR10[28]	PSR10[27]	PSR10[26]	PSR10[25]	PSR10[24]
		PSR10[23]	PSR10[22]	PSR10[21]	PSR10[20]	PSR10[19]	PSR10[18]	PSR10[17]	PSR10[16]
		PSR10[15]	PSR10[14]	PSR10[13]	PSR10[12]	PSR10[11]	PSR10[10]	PSR10[9]	PSR10[8]
		PSR10[7]	PSR10[6]	PSR10[5]	PSR10[4]	PSR10[3]	PSR10[2]	PSR10[1]	PSR10[0]
	PSR11	PSR11[31]	PSR11[30]	PSR11[29]	PSR11[28]	PSR11[27]	PSR11[26]	PSR11[25]	PSR11[24]
		PSR11[23]	PSR11[22]	PSR11[21]	PSR11[20]	PSR11[19]	PSR11[18]	PSR11[17]	PSR11[16]
		PSR11[15]	PSR11[14]	PSR11[13]	PSR11[12]	PSR11[11]	PSR11[10]	PSR11[9]	PSR11[8]
		PSR11[7]	PSR11[6]	PSR11[5]	PSR11[4]	PSR11[3]	PSR11[2]	PSR11[1]	PSR11[0]
	PPR0	-	-	-	-	-	-	-	-
		-	-	PPR0[5]	PPR0[4]	PPR0[3]	PPR0[2]	PPR0[1]	PPR0[0]
	PPR1	PPR1[15]	PPR1[14]	PPR1[13]	PPR1[12]	PPR1[11]	PPR1[10]	PPR1[9]	PPR1[8]
		PPR1[7]	PPR1[6]	PPR1[5]	PPR1[4]	PPR1[3]	PPR1[2]	PPR1[1]	PPR1[0]
	PPR2	PPR2[15]	PPR2[14]	PPR2[13]	PPR2[12]	PPR2[11]	PPR2[10]	PPR2[9]	PPR2[8]
		PPR2[7]	PPR2[6]	PPR2[5]	PPR2[4]	PPR2[3]	PPR2[2]	PPR2[1]	PPR2[0]
	PPR3	PPR3[15]	PPR3[14]	PPR3[13]	PPR3[12]	PPR3[11]	PPR3[10]	PPR3[9]	PPR3[8]
		PPR3[7]	PPR3[6]	PPR3[5]	PPR3[4]	PPR3[3]	PPR3[2]	PPR3[1]	PPR3[0]
	PPR4	PPR4[15]	PPR4[14]	PPR4[13]	PPR4[12]	PPR4[11]	PPR4[10]	PPR4[9]	PPR4[8]
		PPR4[7]	PPR4[6]	PPR4[5]	PPR4[4]	PPR4[3]	PPR4[2]	PPR4[1]	PPR4[0]
	PPR5	-	-	-	-	-	PPR5[10]	PPR5[9]	PPR5[8]
		PPR5[7]	PPR5[6]	PPR5[5]	PPR5[4]	PPR5[3]	PPR5[2]	PPR5[1]	PPR5[0]
	PPR6	PPR6[15]	PPR6[14]	PPR6[13]	PPR6[12]	PPR6[11]	PPR6[10]	PPR6[9]	PPR6[8]
		PPR6[7]	PPR6[6]	PPR6[5]	PPR6[4]	PPR6[3]	PPR6[2]	PPR6[1]	PPR6[0]
	PPR7	PPR7[15]	PPR7[14]	PPR7[13]	PPR7[12]	PPR7[11]	PPR7[10]	PPR7[9]	PPR7[8]
		PPR7[7]	PPR7[6]	PPR7[5]	PPR7[4]	PPR7[3]	PPR7[2]	PPR7[1]	PPR7[0]
	PPR8	PPR8[15]	PPR8[14]	PPR8[13]	PPR8[12]	PPR8[11]	PPR8[10]	PPR8[9]	PPR8[8]
		PPR8[7]	PPR8[6]	PPR8[5]	PPR8[4]	PPR8[3]	PPR8[2]	PPR8[1]	PPR8[0]
	PPR9	-	-	-	-	-	-	-	-
		PPR9[7]	PPR9[6]	PPR9[5]	PPR9[4]	PPR9[3]	PPR9[2]	PPR9[1]	PPR9[0]
	PPR10	PPR10[15]	PPR10[14]	PPR10[13]	PPR10[12]	PPR10[11]	PPR10[10]	PPR10[9]	PPR10[8]
		PPR10[7]	PPR10[6]	PPR10[5]	PPR10[4]	PPR10[3]	PPR10[2]	PPR10[1]	PPR10[0]
	PPR11	PPR11[15]	PPR11[14]	PPR11[13]	PPR11[12]	PPR11[11]	PPR11[10]	PPR11[9]	PPR11[8]
		PPR11[7]	PPR11[6]	PPR11[5]	PPR11[4]	PPR11[3]	PPR11[2]	PPR11[1]	PPR11[0]
	PM1	-	-	-	-	-	-	-	-
		PM1[7]	PM1[6]	PM1[5]	PM1[4]	PM1[3]	PM1[2]	PM1[1]	PM1[0]
	PM2	PM2[15]	PM2[14]	PM2[13]	PM2[12]	PM2[11]	PM2[10]	PM2[9]	PM2[8]
		PM2[7]	PM2[6]	PM2[5]	PM2[4]	PM2[3]	PM2[2]	PM2[1]	PM2[0]
	PM3	PM3[15]	PM3[14]	PM3[13]	PM3[12]	PM3[11]	PM3[10]	PM3[9]	PM3[8]
		PM3[7]	PM3[6]	PM3[5]	PM3[4]	PM3[3]	PM3[2]	PM3[1]	PM3[0]
	PM4	PM4[15]	PM4[14]	PM4[13]	PM4[12]	PM4[11]	PM4[10]	PM4[9]	PM4[8]
		PM4[7]	PM4[6]	PM4[5]	PM4[4]	PM4[3]	PM4[2]	PM4[1]	PM4[0]
	PM5	-	-	-	-	-	PM5[10]	PM5[9]	PM5[8]
		PM5[7]	PM5[6]	PM5[5]	PM5[4]	PM5[3]	PM5[2]	PM5[1]	PM5[0]
PM6	PM6[15]	PM6[14]	PM6[13]	PM6[12]	PM6[11]	PM6[10]	PM6[9]	PM6[8]	
	PM6[7]	PM6[6]	PM6[5]	PM6[4]	PM6[3]	PM6[2]	PM6[1]	PM6[0]	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Ports	PM7	PM7[15]	PM7[14]	PM7[13]	PM7[12]	PM7[11]	PM7[10]	PM7[9]	PM7[8]
		PM7[7]	PM7[6]	PM7[5]	PM7[4]	PM7[3]	PM7[2]	PM7[1]	PM7[0]
	PM8	PM8[15]	PM8[14]	PM8[13]	PM8[12]	PM8[11]	PM8[10]	PM8[9]	PM8[8]
		PM8[7]	PM8[6]	PM8[5]	PM8[4]	PM8[3]	PM8[2]	PM8[1]	PM8[0]
	PM9	-	-	-	-	-	-	-	-
		PM9[7]	PM9[6]	PM9[5]	PM9[4]	PM9[3]	PM9[2]	PM9[1]	PM9[0]
	PM10	PM10[15]	PM10[14]	PM10[13]	PM10[12]	PM10[11]	PM10[10]	PM10[9]	PM10[8]
		PM10[7]	PM10[6]	PM10[5]	PM10[4]	PM10[3]	PM10[2]	PM10[1]	PM10[0]
	PM11	PM11[15]	PM11[14]	PM11[13]	PM11[12]	PM11[11]	PM11[10]	PM11[9]	PM11[8]
		PM11[7]	PM11[6]	PM11[5]	PM11[4]	PM11[3]	PM11[2]	PM11[1]	PM11[0]
	PMC0	-	-	-	-	-	-	-	-
		-	-	PMC0[5]	PMC0[4]	-	-	-	-
	PMC1	PMC1[15]	PMC1[14]	PMC1[13]	PMC1[12]	PMC1[11]	PMC1[10]	PMC1[9]	PMC1[8]
		PMC1[7]	PMC1[6]	PMC1[5]	PMC1[4]	PMC1[3]	PMC1[2]	PMC1[1]	PMC1[0]
	PMC2	PMC2[15]	PMC2[14]	PMC2[13]	PMC2[12]	PMC2[11]	PMC2[10]	PMC2[9]	PMC2[8]
		PMC2[7]	PMC2[6]	PMC2[5]	PMC2[4]	PMC2[3]	PMC2[2]	PMC2[1]	PMC2[0]
	PMC3	PMC3[15]	PMC3[14]	PMC3[13]	PMC3[12]	PMC3[11]	PMC3[10]	PMC3[9]	PMC3[8]
		PMC3[7]	PMC3[6]	PMC3[5]	PMC3[4]	PMC3[3]	PMC3[2]	PMC3[1]	PMC3[0]
	PMC4	PMC4[15]	PMC4[14]	PMC4[13]	PMC4[12]	PMC4[11]	PMC4[10]	PMC4[9]	PMC4[8]
		PMC4[7]	PMC4[6]	PMC4[5]	PMC4[4]	PMC4[3]	PMC4[2]	PMC4[1]	PMC4[0]
	PMC5	-	-	-	-	-	PMC5[10]	PMC5[9]	PMC5[8]
		PMC5[7]	PMC5[6]	PMC5[5]	PMC5[4]	PMC5[3]	PMC5[2]	PMC5[1]	PMC5[0]
	PMC6	PMC6[15]	PMC6[14]	PMC6[13]	PMC6[12]	PMC6[11]	PMC6[10]	PMC6[9]	PMC6[8]
		PMC6[7]	PMC6[6]	PMC6[5]	PMC6[4]	PMC6[3]	PMC6[2]	PMC6[1]	PMC6[0]
	PMC7	PMC7[15]	PMC7[14]	PMC7[13]	PMC7[12]	PMC7[11]	PMC7[10]	PMC7[9]	PMC7[8]
		PMC7[7]	PMC7[6]	PMC7[5]	PMC7[4]	PMC7[3]	PMC7[2]	PMC7[1]	PMC7[0]
	PMC8	PMC8[15]	PMC8[14]	PMC8[13]	PMC8[12]	PMC8[11]	PMC8[10]	PMC8[9]	PMC8[8]
		PMC8[7]	PMC8[6]	PMC8[5]	PMC8[4]	PMC8[3]	PMC8[2]	PMC8[1]	PMC8[0]
	PMC9	-	-	-	-	-	-	-	-
		PMC9[7]	PMC9[6]	PMC9[5]	PMC9[4]	PMC9[3]	PMC9[2]	PMC9[1]	PMC9[0]
	PMC10	PMC10[15]	PMC10[14]	PMC10[13]	PMC10[12]	PMC10[11]	PMC10[10]	PMC10[9]	PMC10[8]
		PMC10[7]	PMC10[6]	PMC10[5]	PMC10[4]	PMC10[3]	PMC10[2]	PMC10[1]	PMC10[0]
	PMC11	PMC11[15]	PMC11[14]	PMC11[13]	PMC11[12]	PMC11[11]	PMC11[10]	PMC11[9]	PMC11[8]
		PMC11[7]	PMC11[6]	PMC11[5]	PMC11[4]	PMC11[3]	PMC11[2]	PMC11[1]	PMC11[0]
	PFC1	PFC1[15]	PFC1[14]	PFC1[13]	PFC1[12]	PFC1[11]	PFC1[10]	PFC1[9]	PFC1[8]
		PFC1[7]	PFC1[6]	PFC1[5]	PFC1[4]	PFC1[3]	PFC1[2]	PFC1[1]	PFC1[0]
	PFC2	PFC2[15]	PFC2[14]	PFC2[13]	PFC2[12]	PFC2[11]	PFC2[10]	PFC2[9]	PFC2[8]
		PFC2[7]	PFC2[6]	PFC2[5]	PFC2[4]	PFC2[3]	PFC2[2]	PFC2[1]	PFC2[0]
	PFC3	PFC3[15]	PFC3[14]	PFC3[13]	PFC3[12]	PFC3[11]	PFC3[10]	PFC3[9]	PFC3[8]
		PFC3[7]	PFC3[6]	PFC3[5]	PFC3[4]	PFC3[3]	PFC3[2]	PFC3[1]	PFC3[0]
	PFC4	PFC4[15]	PFC4[14]	PFC4[13]	PFC4[12]	PFC4[11]	PFC4[10]	PFC4[9]	PFC4[8]
		PFC4[7]	PFC4[6]	PFC4[5]	PFC4[4]	PFC4[3]	PFC4[2]	PFC4[1]	PFC4[0]
	PFC5	-	-	-	-	-	PFC5[10]	PFC5[9]	PFC5[8]
		PFC5[7]	PFC5[6]	PFC5[5]	PFC5[4]	PFC5[3]	PFC5[2]	PFC5[1]	PFC5[0]
	PFC6	PFC6[15]	PFC6[14]	PFC6[13]	PFC6[12]	PFC6[11]	PFC6[10]	PFC6[9]	PFC6[8]
		PFC6[7]	PFC6[6]	PFC6[5]	PFC6[4]	PFC6[3]	PFC6[2]	PFC6[1]	PFC6[0]
	PFC7	PFC7[15]	PFC7[14]	PFC7[13]	PFC7[12]	PFC7[11]	PFC7[10]	PFC7[9]	PFC7[8]
		PFC7[7]	PFC7[6]	PFC7[5]	PFC7[4]	PFC7[3]	PFC7[2]	PFC7[1]	PFC7[0]
	PFC8	PFC8[15]	PFC8[14]	PFC8[13]	PFC8[12]	PFC8[11]	PFC8[10]	PFC8[9]	PFC8[8]
		PFC8[7]	PFC8[6]	PFC8[5]	PFC8[4]	PFC8[3]	PFC8[2]	PFC8[1]	PFC8[0]
	PFC9	-	-	-	-	-	-	-	-
		PFC9[7]	PFC9[6]	PFC9[5]	PFC9[4]	PFC9[3]	PFC9[2]	PFC9[1]	PFC9[0]
	PFC10	PFC10[15]	PFC10[14]	PFC10[13]	PFC10[12]	PFC10[11]	PFC10[10]	PFC10[9]	PFC10[8]
		PFC10[7]	PFC10[6]	PFC10[5]	PFC10[4]	PFC10[3]	PFC10[2]	PFC10[1]	PFC10[0]
	PFC11	PFC11[15]	PFC11[14]	PFC11[13]	PFC11[12]	PFC11[11]	PFC11[10]	PFC11[9]	PFC11[8]
		PFC11[7]	PFC11[6]	PFC11[5]	PFC11[4]	PFC11[3]	PFC11[2]	PFC11[1]	PFC11[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Ports	PFCE1	PFCE1[15]	PFCE1[14]	PFCE1[13]	PFCE1[12]	PFCE1[11]	PFCE1[10]	PFCE1[9]	PFCE1[8]
		PFCE1[7]	PFCE1[6]	PFCE1[5]	PFCE1[4]	PFCE1[3]	PFCE1[2]	PFCE1[1]	PFCE1[0]
	PFCE2	PFCE2[15]	PFCE2[14]	PFCE2[13]	PFCE2[12]	PFCE2[11]	PFCE2[10]	PFCE2[9]	PFCE2[8]
		PFCE2[7]	PFCE2[6]	PFCE2[5]	PFCE2[4]	PFCE2[3]	PFCE2[2]	PFCE2[1]	PFCE2[0]
	PFCE3	PFCE3[15]	PFCE3[14]	PFCE3[13]	PFCE3[12]	PFCE3[11]	PFCE3[10]	PFCE3[9]	PFCE3[8]
		PFCE3[7]	PFCE3[6]	PFCE3[5]	PFCE3[4]	PFCE3[3]	PFCE3[2]	PFCE3[1]	PFCE3[0]
	PFCE4	PFCE4[15]	PFCE4[14]	PFCE4[13]	PFCE4[12]	PFCE4[11]	PFCE4[10]	PFCE4[9]	PFCE4[8]
		PFCE4[7]	PFCE4[6]	PFCE4[5]	PFCE4[4]	PFCE4[3]	PFCE4[2]	PFCE4[1]	PFCE4[0]
	PFCE5	-	-	-	-	-	PFCE5[10]	PFCE5[9]	PFCE5[8]
		PFCE5[7]	PFCE5[6]	PFCE5[5]	PFCE5[4]	PFCE5[3]	PFCE5[2]	PFCE5[1]	PFCE5[0]
	PFCE6	PFCE6[15]	PFCE6[14]	PFCE6[13]	PFCE6[12]	PFCE6[11]	PFCE6[10]	PFCE6[9]	PFCE6[8]
		PFCE6[7]	PFCE6[6]	PFCE6[5]	PFCE6[4]	PFCE6[3]	PFCE6[2]	PFCE6[1]	PFCE6[0]
	PFCE7	PFCE7[15]	PFCE7[14]	PFCE7[13]	PFCE7[12]	PFCE7[11]	PFCE7[10]	PFCE7[9]	PFCE7[8]
		PFCE7[7]	PFCE7[6]	PFCE7[5]	PFCE7[4]	PFCE7[3]	PFCE7[2]	PFCE7[1]	PFCE7[0]
	PFCE8	PFCE8[15]	PFCE8[14]	PFCE8[13]	PFCE8[12]	PFCE8[11]	PFCE8[10]	PFCE8[9]	PFCE8[8]
		PFCE8[7]	PFCE8[6]	PFCE8[5]	PFCE8[4]	PFCE8[3]	PFCE8[2]	PFCE8[1]	PFCE8[0]
	PFCE9	-	-	-	-	-	-	-	-
		PFCE9[7]	PFCE9[6]	PFCE9[5]	PFCE9[4]	PFCE9[3]	PFCE9[2]	PFCE9[1]	PFCE9[0]
	PFCE10	PFCE10[15]	PFCE10[14]	PFCE10[13]	PFCE10[12]	PFCE10[11]	PFCE10[10]	PFCE10[9]	PFCE10[8]
		PFCE10[7]	PFCE10[6]	PFCE10[5]	PFCE10[4]	PFCE10[3]	PFCE10[2]	PFCE10[1]	PFCE10[0]
	PFCE11	PFCE11[15]	PFCE11[14]	PFCE11[13]	PFCE11[12]	PFCE11[11]	PFCE11[10]	PFCE11[9]	PFCE11[8]
		PFCE11[7]	PFCE11[6]	PFCE11[5]	PFCE11[4]	PFCE11[3]	PFCE11[2]	PFCE11[1]	PFCE11[0]
	PNOT1	-	-	-	-	-	-	-	-
		PNOT1[7]	PNOT1[6]	PNOT1[5]	PNOT1[4]	PNOT1[3]	PNOT1[2]	PNOT1[1]	PNOT1[0]
	PNOT2	PNOT2[15]	PNOT2[14]	PNOT2[13]	PNOT2[12]	PNOT2[11]	PNOT2[10]	PNOT2[9]	PNOT2[8]
		PNOT2[7]	PNOT2[6]	PNOT2[5]	PNOT2[4]	PNOT2[3]	PNOT2[2]	PNOT2[1]	PNOT2[0]
	PNOT3	PNOT3[15]	PNOT3[14]	PNOT3[13]	PNOT3[12]	PNOT3[11]	PNOT3[10]	PNOT3[9]	PNOT3[8]
		PNOT3[7]	PNOT3[6]	PNOT3[5]	PNOT3[4]	PNOT3[3]	PNOT3[2]	PNOT3[1]	PNOT3[0]
	PNOT4	PNOT4[15]	PNOT4[14]	PNOT4[13]	PNOT4[12]	PNOT4[11]	PNOT4[10]	PNOT4[9]	PNOT4[8]
		PNOT4[7]	PNOT4[6]	PNOT4[5]	PNOT4[4]	PNOT4[3]	PNOT4[2]	PNOT4[1]	PNOT4[0]
	PNOT5	-	-	-	-	-	PNOT5[10]	PNOT5[9]	PNOT5[8]
		PNOT5[7]	PNOT5[6]	PNOT5[5]	PNOT5[4]	PNOT5[3]	PNOT5[2]	PNOT5[1]	PNOT5[0]
	PNOT6	PNOT6[15]	PNOT6[14]	PNOT6[13]	PNOT6[12]	PNOT6[11]	PNOT6[10]	PNOT6[9]	PNOT6[8]
		PNOT6[7]	PNOT6[6]	PNOT6[5]	PNOT6[4]	PNOT6[3]	PNOT6[2]	PNOT6[1]	PNOT6[0]
	PNOT7	PNOT7[15]	PNOT7[14]	PNOT7[13]	PNOT7[12]	PNOT7[11]	PNOT7[10]	PNOT7[9]	PNOT7[8]
		PNOT7[7]	PNOT7[6]	PNOT7[5]	PNOT7[4]	PNOT7[3]	PNOT7[2]	PNOT7[1]	PNOT7[0]
	PNOT8	PNOT8[15]	PNOT8[14]	PNOT8[13]	PNOT8[12]	PNOT8[11]	PNOT8[10]	PNOT8[9]	PNOT8[8]
		PNOT8[7]	PNOT8[6]	PNOT8[5]	PNOT8[4]	PNOT8[3]	PNOT8[2]	PNOT8[1]	PNOT8[0]
	PNOT9	-	-	-	-	-	-	-	-
		PNOT9[7]	PNOT9[6]	PNOT9[5]	PNOT9[4]	PNOT9[3]	PNOT9[2]	PNOT9[1]	PNOT9[0]
	PNOT10	PNOT10[15]	PNOT10[14]	PNOT10[13]	PNOT10[12]	PNOT10[11]	PNOT10[10]	PNOT10[9]	PNOT10[8]
		PNOT10[7]	PNOT10[6]	PNOT10[5]	PNOT10[4]	PNOT10[3]	PNOT10[2]	PNOT10[1]	PNOT10[0]
	PNOT11	PNOT11[15]	PNOT11[14]	PNOT11[13]	PNOT11[12]	PNOT11[11]	PNOT11[10]	PNOT11[9]	PNOT11[8]
		PNOT11[7]	PNOT11[6]	PNOT11[5]	PNOT11[4]	PNOT11[3]	PNOT11[2]	PNOT11[1]	PNOT11[0]
	PMSR1	-	-	-	-	-	-	-	-
		PMSR1[23]	PMSR1[22]	PMSR1[21]	PMSR1[20]	PMSR1[19]	PMSR1[18]	PMSR1[17]	PMSR1[16]
		-	-	-	-	-	-	-	-
	PMSR2	PMSR1[7]	PMSR1[6]	PMSR1[5]	PMSR1[4]	PMSR1[3]	PMSR1[2]	PMSR1[1]	PMSR1[0]
		PMSR2[31]	PMSR2[30]	PMSR2[29]	PMSR2[28]	PMSR2[27]	PMSR2[26]	PMSR2[25]	PMSR2[24]
		PMSR2[23]	PMSR2[22]	PMSR2[21]	PMSR2[20]	PMSR2[19]	PMSR2[18]	PMSR2[17]	PMSR2[16]
		PMSR2[15]	PMSR2[14]	PMSR2[13]	PMSR2[12]	PMSR2[11]	PMSR2[10]	PMSR2[9]	PMSR2[8]
	PMSR3	PMSR2[7]	PMSR2[6]	PMSR2[5]	PMSR2[4]	PMSR2[3]	PMSR2[2]	PMSR2[1]	PMSR2[0]
		PMSR3[31]	PMSR3[30]	PMSR3[29]	PMSR3[28]	PMSR3[27]	PMSR3[26]	PMSR3[25]	PMSR3[24]
		PMSR3[23]	PMSR3[22]	PMSR3[21]	PMSR3[20]	PMSR3[19]	PMSR3[18]	PMSR3[17]	PMSR3[16]
		PMSR3[15]	PMSR3[14]	PMSR3[13]	PMSR3[12]	PMSR3[11]	PMSR3[10]	PMSR3[9]	PMSR3[8]
			PMSR3[7]	PMSR3[6]	PMSR3[5]	PMSR3[4]	PMSR3[3]	PMSR3[2]	PMSR3[1]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Ports	PMSR4	PMSR4[31]	PMSR4[30]	PMSR4[29]	PMSR4[28]	PMSR4[27]	PMSR4[26]	PMSR4[25]	PMSR4[24]	
		PMSR4[23]	PMSR4[22]	PMSR4[21]	PMSR4[20]	PMSR4[19]	PMSR4[18]	PMSR4[17]	PMSR4[16]	
		PMSR4[15]	PMSR4[14]	PMSR4[13]	PMSR4[12]	PMSR4[11]	PMSR4[10]	PMSR4[9]	PMSR4[8]	
		PMSR4[7]	PMSR4[6]	PMSR4[5]	PMSR4[4]	PMSR4[3]	PMSR4[2]	PMSR4[1]	PMSR4[0]	
	PMSR5	-	-	-	-	-	-	PMSR5[26]	PMSR5[25]	PMSR5[24]
		PMSR5[23]	PMSR5[22]	PMSR5[21]	PMSR5[20]	PMSR5[19]	PMSR5[18]	PMSR5[17]	PMSR5[16]	
		-	-	-	-	-	-	PMSR5[10]	PMSR5[9]	PMSR5[8]
		PMSR5[7]	PMSR5[6]	PMSR5[5]	PMSR5[4]	PMSR5[3]	PMSR5[2]	PMSR5[1]	PMSR5[0]	
	PMSR6	PMSR6[31]	PMSR6[30]	PMSR6[29]	PMSR6[28]	PMSR6[27]	PMSR6[26]	PMSR6[25]	PMSR6[24]	
		PMSR6[23]	PMSR6[22]	PMSR6[21]	PMSR6[20]	PMSR6[19]	PMSR6[18]	PMSR6[17]	PMSR6[16]	
		PMSR6[15]	PMSR6[14]	PMSR6[13]	PMSR6[12]	PMSR6[11]	PMSR6[10]	PMSR6[9]	PMSR6[8]	
		PMSR6[7]	PMSR6[6]	PMSR6[5]	PMSR6[4]	PMSR6[3]	PMSR6[2]	PMSR6[1]	PMSR6[0]	
	PMSR7	PMSR7[31]	PMSR7[30]	PMSR7[29]	PMSR7[28]	PMSR7[27]	PMSR7[26]	PMSR7[25]	PMSR7[24]	
		PMSR7[23]	PMSR7[22]	PMSR7[21]	PMSR7[20]	PMSR7[19]	PMSR7[18]	PMSR7[17]	PMSR7[16]	
		PMSR7[15]	PMSR7[14]	PMSR7[13]	PMSR7[12]	PMSR7[11]	PMSR7[10]	PMSR7[9]	PMSR7[8]	
		PMSR7[7]	PMSR7[6]	PMSR7[5]	PMSR7[4]	PMSR7[3]	PMSR7[2]	PMSR7[1]	PMSR7[0]	
	PMSR8	PMSR8[31]	PMSR8[30]	PMSR8[29]	PMSR8[28]	PMSR8[27]	PMSR8[26]	PMSR8[25]	PMSR8[24]	
		PMSR8[23]	PMSR8[22]	PMSR8[21]	PMSR8[20]	PMSR8[19]	PMSR8[18]	PMSR8[17]	PMSR8[16]	
		PMSR8[15]	PMSR8[14]	PMSR8[13]	PMSR8[12]	PMSR8[11]	PMSR8[10]	PMSR8[9]	PMSR8[8]	
		PMSR8[7]	PMSR8[6]	PMSR8[5]	PMSR8[4]	PMSR8[3]	PMSR8[2]	PMSR8[1]	PMSR8[0]	
	PMSR9	-	-	-	-	-	-	-	-	
		PMSR9[23]	PMSR9[22]	PMSR9[21]	PMSR9[20]	PMSR9[19]	PMSR9[18]	PMSR9[17]	PMSR9[16]	
		-	-	-	-	-	-	-	-	
		PMSR9[7]	PMSR9[6]	PMSR9[5]	PMSR9[4]	PMSR9[3]	PMSR9[2]	PMSR9[1]	PMSR9[0]	
	PMSR10	PMSR10[31]	PMSR10[30]	PMSR10[29]	PMSR10[28]	PMSR10[27]	PMSR10[26]	PMSR10[25]	PMSR10[24]	
		PMSR10[23]	PMSR10[22]	PMSR10[21]	PMSR10[20]	PMSR10[19]	PMSR10[18]	PMSR10[17]	PMSR10[16]	
		PMSR10[15]	PMSR10[14]	PMSR10[13]	PMSR10[12]	PMSR10[11]	PMSR10[10]	PMSR10[9]	PMSR10[8]	
		PMSR10[7]	PMSR10[6]	PMSR10[5]	PMSR10[4]	PMSR10[3]	PMSR10[2]	PMSR10[1]	PMSR10[0]	
	PMSR11	PMSR11[31]	PMSR11[30]	PMSR11[29]	PMSR11[28]	PMSR11[27]	PMSR11[26]	PMSR11[25]	PMSR11[24]	
		PMSR11[23]	PMSR11[22]	PMSR11[21]	PMSR11[20]	PMSR11[19]	PMSR11[18]	PMSR11[17]	PMSR11[16]	
		PMSR11[15]	PMSR11[14]	PMSR11[13]	PMSR11[12]	PMSR11[11]	PMSR11[10]	PMSR11[9]	PMSR11[8]	
		PMSR11[7]	PMSR11[6]	PMSR11[5]	PMSR11[4]	PMSR11[3]	PMSR11[2]	PMSR11[1]	PMSR11[0]	
	PMCSR0	-	-	-	-	-	-	-	-	
		-	-	PMCSR0[21]	PMCSR0[20]	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	PMCSR0[5]	PMCSR0[4]	-	-	-	-	
	PMCSR1	-	-	-	-	-	-	-	-	
		PMCSR1[23]	PMCSR1[22]	PMCSR1[21]	PMCSR1[20]	PMCSR1[19]	PMCSR1[18]	PMCSR1[17]	PMCSR1[16]	
		-	-	-	-	-	-	-	-	
	PMCSR2	PMCSR1[7]	PMCSR1[6]	PMCSR1[5]	PMCSR1[4]	PMCSR1[3]	PMCSR1[2]	PMCSR1[1]	PMCSR1[0]	
		PMCSR2[31]	PMCSR2[30]	PMCSR2[29]	PMCSR2[28]	PMCSR2[27]	PMCSR2[26]	PMCSR2[25]	PMCSR2[24]	
		PMCSR2[23]	PMCSR2[22]	PMCSR2[21]	PMCSR2[20]	PMCSR2[19]	PMCSR2[18]	PMCSR2[17]	PMCSR2[16]	
		PMCSR2[15]	PMCSR2[14]	PMCSR2[13]	PMCSR2[12]	PMCSR2[11]	PMCSR2[10]	PMCSR2[9]	PMCSR2[8]	
	PMCSR3	PMCSR2[7]	PMCSR2[6]	PMCSR2[5]	PMCSR2[4]	PMCSR2[3]	PMCSR2[2]	PMCSR2[1]	PMCSR2[0]	
		PMCSR3[31]	PMCSR3[30]	PMCSR3[29]	PMCSR3[28]	PMCSR3[27]	PMCSR3[26]	PMCSR3[25]	PMCSR3[24]	
		PMCSR3[23]	PMCSR3[22]	PMCSR3[21]	PMCSR3[20]	PMCSR3[19]	PMCSR3[18]	PMCSR3[17]	PMCSR3[16]	
		PMCSR3[15]	PMCSR3[14]	PMCSR3[13]	PMCSR3[12]	PMCSR3[11]	PMCSR3[10]	PMCSR3[9]	PMCSR3[8]	
	PMCSR4	PMCSR3[7]	PMCSR3[6]	PMCSR3[5]	PMCSR3[4]	PMCSR3[3]	PMCSR3[2]	PMCSR3[1]	PMCSR3[0]	
		PMCSR4[31]	PMCSR4[30]	PMCSR4[29]	PMCSR4[28]	PMCSR4[27]	PMCSR4[26]	PMCSR4[25]	PMCSR4[24]	
		PMCSR4[23]	PMCSR4[22]	PMCSR4[21]	PMCSR4[20]	PMCSR4[19]	PMCSR4[18]	PMCSR4[17]	PMCSR4[16]	
		PMCSR4[15]	PMCSR4[14]	PMCSR4[13]	PMCSR4[12]	PMCSR4[11]	PMCSR4[10]	PMCSR4[9]	PMCSR4[8]	
	PMCSR5	PMCSR4[7]	PMCSR4[6]	PMCSR4[5]	PMCSR4[4]	PMCSR4[3]	PMCSR4[2]	PMCSR4[1]	PMCSR4[0]	
		-	-	-	-	-	-	-	-	
		PMCSR5[23]	PMCSR5[22]	PMCSR5[21]	PMCSR5[20]	PMCSR5[19]	PMCSR5[18]	PMCSR5[17]	PMCSR5[16]	
		-	-	-	-	-	-	-	-	
	PMCSR5	PMCSR5[7]	PMCSR5[6]	PMCSR5[5]	PMCSR5[4]	PMCSR5[3]	PMCSR5[2]	PMCSR5[1]	PMCSR5[0]	
		-	-	-	-	-	-	-	-	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Ports	PMCSR6	PMCSR6[31]	PMCSR6[30]	PMCSR6[29]	PMCSR6[28]	PMCSR6[27]	PMCSR6[26]	PMCSR6[25]	PMCSR6[24]
		PMCSR6[23]	PMCSR6[22]	PMCSR6[21]	PMCSR6[20]	PMCSR6[19]	PMCSR6[18]	PMCSR6[17]	PMCSR6[16]
		PMCSR6[15]	PMCSR6[14]	PMCSR6[13]	PMCSR6[12]	PMCSR6[11]	PMCSR6[10]	PMCSR6[9]	PMCSR6[8]
		PMCSR6[7]	PMCSR6[6]	PMCSR6[5]	PMCSR6[4]	PMCSR6[3]	PMCSR6[2]	PMCSR6[1]	PMCSR6[0]
	PMCSR7	PMCSR7[31]	PMCSR7[30]	PMCSR7[29]	PMCSR7[28]	PMCSR7[27]	PMCSR7[26]	PMCSR7[25]	PMCSR7[24]
		PMCSR7[23]	PMCSR7[22]	PMCSR7[21]	PMCSR7[20]	PMCSR7[19]	PMCSR7[18]	PMCSR7[17]	PMCSR7[16]
		PMCSR7[15]	PMCSR7[14]	PMCSR7[13]	PMCSR7[12]	PMCSR7[11]	PMCSR7[10]	PMCSR7[9]	PMCSR7[8]
		PMCSR7[7]	PMCSR7[6]	PMCSR7[5]	PMCSR7[4]	PMCSR7[3]	PMCSR7[2]	PMCSR7[1]	PMCSR7[0]
	PMCSR8	PMCSR8[31]	PMCSR8[30]	PMCSR8[29]	PMCSR8[28]	PMCSR8[27]	PMCSR8[26]	PMCSR8[25]	PMCSR8[24]
		PMCSR8[23]	PMCSR8[22]	PMCSR8[21]	PMCSR8[20]	PMCSR8[19]	PMCSR8[18]	PMCSR8[17]	PMCSR8[16]
		PMCSR8[15]	PMCSR8[14]	PMCSR8[13]	PMCSR8[12]	PMCSR8[11]	PMCSR8[10]	PMCSR8[9]	PMCSR8[8]
		PMCSR8[7]	PMCSR8[6]	PMCSR8[5]	PMCSR8[4]	PMCSR8[3]	PMCSR8[2]	PMCSR8[1]	PMCSR8[0]
	PMCSR9	-	-	-	-	-	-	-	-
		PMCSR9[23]	PMCSR9[22]	PMCSR9[21]	PMCSR9[20]	PMCSR9[19]	PMCSR9[18]	PMCSR9[17]	PMCSR9[16]
		-	-	-	-	-	-	-	-
		PMCSR9[7]	PMCSR9[6]	PMCSR9[5]	PMCSR9[4]	PMCSR9[3]	PMCSR9[2]	PMCSR9[1]	PMCSR9[0]
	PMCSR10	PMCSR10[31]	PMCSR10[30]	PMCSR10[29]	PMCSR10[28]	PMCSR10[27]	PMCSR10[26]	PMCSR10[25]	PMCSR10[24]
		PMCSR10[23]	PMCSR10[22]	PMCSR10[21]	PMCSR10[20]	PMCSR10[19]	PMCSR10[18]	PMCSR10[17]	PMCSR10[16]
		PMCSR10[15]	PMCSR10[14]	PMCSR10[13]	PMCSR10[12]	PMCSR10[11]	PMCSR10[10]	PMCSR10[9]	PMCSR10[8]
		PMCSR10[7]	PMCSR10[6]	PMCSR10[5]	PMCSR10[4]	PMCSR10[3]	PMCSR10[2]	PMCSR10[1]	PMCSR10[0]
	PMCSR11	PMCSR11[31]	PMCSR11[30]	PMCSR11[29]	PMCSR11[28]	PMCSR11[27]	PMCSR11[26]	PMCSR11[25]	PMCSR11[24]
		PMCSR11[23]	PMCSR11[22]	PMCSR11[21]	PMCSR11[20]	PMCSR11[19]	PMCSR11[18]	PMCSR11[17]	PMCSR11[16]
		PMCSR11[15]	PMCSR11[14]	PMCSR11[13]	PMCSR11[12]	PMCSR11[11]	PMCSR11[10]	PMCSR11[9]	PMCSR11[8]
		PMCSR11[7]	PMCSR11[6]	PMCSR11[5]	PMCSR11[4]	PMCSR11[3]	PMCSR11[2]	PMCSR11[1]	PMCSR11[0]
	PFCAE1	PFCAE1[15]	PFCAE1[14]	PFCAE1[13]	PFCAE1[12]	PFCAE1[11]	PFCAE1[10]	PFCAE1[9]	PFCAE1[8]
		PFCAE1[7]	PFCAE1[6]	PFCAE1[5]	PFCAE1[4]	PFCAE1[3]	PFCAE1[2]	PFCAE1[1]	PFCAE1[0]
	PFCAE2	PFCAE2[15]	PFCAE2[14]	PFCAE2[13]	PFCAE2[12]	PFCAE2[11]	PFCAE2[10]	PFCAE2[9]	PFCAE2[8]
		PFCAE2[7]	PFCAE2[6]	PFCAE2[5]	PFCAE2[4]	PFCAE2[3]	PFCAE2[2]	PFCAE2[1]	PFCAE2[0]
	PFCAE3	PFCAE3[15]	PFCAE3[14]	PFCAE3[13]	PFCAE3[12]	PFCAE3[11]	PFCAE3[10]	PFCAE3[9]	PFCAE3[8]
		PFCAE3[7]	PFCAE3[6]	PFCAE3[5]	PFCAE3[4]	PFCAE3[3]	PFCAE3[2]	PFCAE3[1]	PFCAE3[0]
	PFCAE4	PFCAE4[15]	PFCAE4[14]	PFCAE4[13]	PFCAE4[12]	PFCAE4[11]	PFCAE4[10]	PFCAE4[9]	PFCAE4[8]
		PFCAE4[7]	PFCAE4[6]	PFCAE4[5]	PFCAE4[4]	PFCAE4[3]	PFCAE4[2]	PFCAE4[1]	PFCAE4[0]
	PFCAE5	-	-	-	-	-	PFCAE5[10]	PFCAE5[9]	PFCAE5[8]
		PFCAE5[7]	PFCAE5[6]	PFCAE5[5]	PFCAE5[4]	PFCAE5[3]	PFCAE5[2]	PFCAE5[1]	PFCAE5[0]
	PFCAE6	PFCAE6[15]	PFCAE6[14]	PFCAE6[13]	PFCAE6[12]	PFCAE6[11]	PFCAE6[10]	PFCAE6[9]	PFCAE6[8]
		PFCAE6[7]	PFCAE6[6]	PFCAE6[5]	PFCAE6[4]	PFCAE6[3]	PFCAE6[2]	PFCAE6[1]	PFCAE6[0]
	PFCAE7	PFCAE7[15]	PFCAE7[14]	PFCAE7[13]	PFCAE7[12]	PFCAE7[11]	PFCAE7[10]	PFCAE7[9]	PFCAE7[8]
		PFCAE7[7]	PFCAE7[6]	PFCAE7[5]	PFCAE7[4]	PFCAE7[3]	PFCAE7[2]	PFCAE7[1]	PFCAE7[0]
	PFCAE8	PFCAE8[15]	PFCAE8[14]	PFCAE8[13]	PFCAE8[12]	PFCAE8[11]	PFCAE8[10]	PFCAE8[9]	PFCAE8[8]
		PFCAE8[7]	PFCAE8[6]	PFCAE8[5]	PFCAE8[4]	PFCAE8[3]	PFCAE8[2]	PFCAE8[1]	PFCAE8[0]
	PFCAE9	-	-	-	-	-	-	-	-
		PFCAE9[7]	PFCAE9[6]	PFCAE9[5]	PFCAE9[4]	PFCAE9[3]	PFCAE9[2]	PFCAE9[1]	PFCAE9[0]
	PFCAE10	PFCAE10[15]	PFCAE10[14]	PFCAE10[13]	PFCAE10[12]	PFCAE10[11]	PFCAE10[10]	PFCAE10[9]	PFCAE10[8]
		PFCAE10[7]	PFCAE10[6]	PFCAE10[5]	PFCAE10[4]	PFCAE10[3]	PFCAE10[2]	PFCAE10[1]	PFCAE10[0]
PFCAE11	PFCAE11[15]	PFCAE11[14]	PFCAE11[13]	PFCAE11[12]	PFCAE11[11]	PFCAE11[10]	PFCAE11[9]	PFCAE11[8]	
	PFCAE11[7]	PFCAE11[6]	PFCAE11[5]	PFCAE11[4]	PFCAE11[3]	PFCAE11[2]	PFCAE11[1]	PFCAE11[0]	
PIBC0	-	-	-	-	-	-	-	-	
	-	-	PIBC0[5]	PIBC0[4]	PIBC0[3]	PIBC0[2]	PIBC0[1]	PIBC0[0]	
PIBC1	PIBC1[15]	PIBC1[14]	PIBC1[13]	PIBC1[12]	PIBC1[11]	PIBC1[10]	PIBC1[9]	PIBC1[8]	
	PIBC1[7]	PIBC1[6]	PIBC1[5]	PIBC1[4]	PIBC1[3]	PIBC1[2]	PIBC1[1]	PIBC1[0]	
PIBC2	PIBC2[15]	PIBC2[14]	PIBC2[13]	PIBC2[12]	PIBC2[11]	PIBC2[10]	PIBC2[9]	PIBC2[8]	
	PIBC2[7]	PIBC2[6]	PIBC2[5]	PIBC2[4]	PIBC2[3]	PIBC2[2]	PIBC2[1]	PIBC2[0]	
PIBC3	PIBC3[15]	PIBC3[14]	PIBC3[13]	PIBC3[12]	PIBC3[11]	PIBC3[10]	PIBC3[9]	PIBC3[8]	
	PIBC3[7]	PIBC3[6]	PIBC3[5]	PIBC3[4]	PIBC3[3]	PIBC3[2]	PIBC3[1]	PIBC3[0]	
PIBC4	PIBC4[15]	PIBC4[14]	PIBC4[13]	PIBC4[12]	PIBC4[11]	PIBC4[10]	PIBC4[9]	PIBC4[8]	
	PIBC4[7]	PIBC4[6]	PIBC4[5]	PIBC4[4]	PIBC4[3]	PIBC4[2]	PIBC4[1]	PIBC4[0]	

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Ports	PIBC5	-	-	-	-	-	PIBC5[10]	PIBC5[9]	PIBC5[8]
		PIBC5[7]	PIBC5[6]	PIBC5[5]	PIBC5[4]	PIBC5[3]	PIBC5[2]	PIBC5[1]	PIBC5[0]
	PIBC6	PIBC6[15]	PIBC6[14]	PIBC6[13]	PIBC6[12]	PIBC6[11]	PIBC6[10]	PIBC6[9]	PIBC6[8]
		PIBC6[7]	PIBC6[6]	PIBC6[5]	PIBC6[4]	PIBC6[3]	PIBC6[2]	PIBC6[1]	PIBC6[0]
	PIBC7	PIBC7[15]	PIBC7[14]	PIBC7[13]	PIBC7[12]	PIBC7[11]	PIBC7[10]	PIBC7[9]	PIBC7[8]
		PIBC7[7]	PIBC7[6]	PIBC7[5]	PIBC7[4]	PIBC7[3]	PIBC7[2]	PIBC7[1]	PIBC7[0]
	PIBC8	PIBC8[15]	PIBC8[14]	PIBC8[13]	PIBC8[12]	PIBC8[11]	PIBC8[10]	PIBC8[9]	PIBC8[8]
		PIBC8[7]	PIBC8[6]	PIBC8[5]	PIBC8[4]	PIBC8[3]	PIBC8[2]	PIBC8[1]	PIBC8[0]
	PIBC9	-	-	-	-	-	-	-	-
		PIBC9[7]	PIBC9[6]	PIBC9[5]	PIBC9[4]	PIBC9[3]	PIBC9[2]	PIBC9[1]	PIBC9[0]
	PIBC10	PIBC10[15]	PIBC10[14]	PIBC10[13]	PIBC10[12]	PIBC10[11]	PIBC10[10]	PIBC10[9]	PIBC10[8]
		PIBC10[7]	PIBC10[6]	PIBC10[5]	PIBC10[4]	PIBC10[3]	PIBC10[2]	PIBC10[1]	PIBC10[0]
	PIBC11	PIBC11[15]	PIBC11[14]	PIBC11[13]	PIBC11[12]	PIBC11[11]	PIBC11[10]	PIBC11[9]	PIBC11[8]
		PIBC11[7]	PIBC11[6]	PIBC11[5]	PIBC11[4]	PIBC11[3]	PIBC11[2]	PIBC11[1]	PIBC11[0]
	PBDC1	PBDC1[15]	PBDC1[14]	PBDC1[13]	PBDC1[12]	PBDC1[11]	PBDC1[10]	PBDC1[9]	PBDC1[8]
		PBDC1[7]	PBDC1[6]	PBDC1[5]	PBDC1[4]	PBDC1[3]	PBDC1[2]	PBDC1[1]	PBDC1[0]
	PBDC2	PBDC2[15]	PBDC2[14]	PBDC2[13]	PBDC2[12]	PBDC2[11]	PBDC2[10]	PBDC2[9]	PBDC2[8]
		PBDC2[7]	PBDC2[6]	PBDC2[5]	PBDC2[4]	PBDC2[3]	PBDC2[2]	PBDC2[1]	PBDC2[0]
	PBDC3	PBDC3[15]	PBDC3[14]	PBDC3[13]	PBDC3[12]	PBDC3[11]	PBDC3[10]	PBDC3[9]	PBDC3[8]
		PBDC3[7]	PBDC3[6]	PBDC3[5]	PBDC3[4]	PBDC3[3]	PBDC3[2]	PBDC3[1]	PBDC3[0]
	PBDC4	PBDC4[15]	PBDC4[14]	PBDC4[13]	PBDC4[12]	PBDC4[11]	PBDC4[10]	PBDC4[9]	PBDC4[8]
		PBDC4[7]	PBDC4[6]	PBDC4[5]	PBDC4[4]	PBDC4[3]	PBDC4[2]	PBDC4[1]	PBDC4[0]
	PBDC5	-	-	-	-	-	PBDC5[10]	PBDC5[9]	PBDC5[8]
		PBDC5[7]	PBDC5[6]	PBDC5[5]	PBDC5[4]	PBDC5[3]	PBDC5[2]	PBDC5[1]	PBDC5[0]
	PBDC6	PBDC6[15]	PBDC6[14]	PBDC6[13]	PBDC6[12]	PBDC6[11]	PBDC6[10]	PBDC6[9]	PBDC6[8]
		PBDC6[7]	PBDC6[6]	PBDC6[5]	PBDC6[4]	PBDC6[3]	PBDC6[2]	PBDC6[1]	PBDC6[0]
	PBDC7	PBDC7[15]	PBDC7[14]	PBDC7[13]	PBDC7[12]	PBDC7[11]	PBDC7[10]	PBDC7[9]	PBDC7[8]
		PBDC7[7]	PBDC7[6]	PBDC7[5]	PBDC7[4]	PBDC7[3]	PBDC7[2]	PBDC7[1]	PBDC7[0]
	PBDC8	PBDC8[15]	PBDC8[14]	PBDC8[13]	PBDC8[12]	PBDC8[11]	PBDC8[10]	PBDC8[9]	PBDC8[8]
		PBDC8[7]	PBDC8[6]	PBDC8[5]	PBDC8[4]	PBDC8[3]	PBDC8[2]	PBDC8[1]	PBDC8[0]
	PBDC9	-	-	-	-	-	-	-	-
		PBDC9[7]	PBDC9[6]	PBDC9[5]	PBDC9[4]	PBDC9[3]	PBDC9[2]	PBDC9[1]	PBDC9[0]
	PBDC10	PBDC10[15]	PBDC10[14]	PBDC10[13]	PBDC10[12]	PBDC10[11]	PBDC10[10]	PBDC10[9]	PBDC10[8]
		PBDC10[7]	PBDC10[6]	PBDC10[5]	PBDC10[4]	PBDC10[3]	PBDC10[2]	PBDC10[1]	PBDC10[0]
	PBDC11	PBDC11[15]	PBDC11[14]	PBDC11[13]	PBDC11[12]	PBDC11[11]	PBDC11[10]	PBDC11[9]	PBDC11[8]
		PBDC11[7]	PBDC11[6]	PBDC11[5]	PBDC11[4]	PBDC11[3]	PBDC11[2]	PBDC11[1]	PBDC11[0]
	PIPC1	-	-	-	-	-	-	-	-
		PIPC1[7]	PIPC1[6]	PIPC1[5]	PIPC1[4]	PIPC1[3]	PIPC1[2]	PIPC1[1]	PIPC1[0]
	PIPC2	PIPC2[15]	PIPC2[14]	PIPC2[13]	PIPC2[12]	PIPC2[11]	PIPC2[10]	PIPC2[9]	PIPC2[8]
		PIPC2[7]	PIPC2[6]	PIPC2[5]	PIPC2[4]	PIPC2[3]	PIPC2[2]	PIPC2[1]	PIPC2[0]
	PIPC3	PIPC3[15]	PIPC3[14]	PIPC3[13]	PIPC3[12]	PIPC3[11]	PIPC3[10]	PIPC3[9]	PIPC3[8]
		PIPC3[7]	PIPC3[6]	PIPC3[5]	PIPC3[4]	PIPC3[3]	PIPC3[2]	PIPC3[1]	PIPC3[0]
	PIPC4	PIPC4[15]	PIPC4[14]	PIPC4[13]	PIPC4[12]	PIPC4[11]	PIPC4[10]	PIPC4[9]	PIPC4[8]
		PIPC4[7]	PIPC4[6]	PIPC4[5]	PIPC4[4]	PIPC4[3]	PIPC4[2]	PIPC4[1]	PIPC4[0]
	PIPC5	-	-	-	-	-	PIPC5[10]	PIPC5[9]	PIPC5[8]
		PIPC5[7]	PIPC5[6]	PIPC5[5]	PIPC5[4]	PIPC5[3]	PIPC5[2]	PIPC5[1]	PIPC5[0]
	PIPC6	PIPC6[15]	PIPC6[14]	PIPC6[13]	PIPC6[12]	PIPC6[11]	PIPC6[10]	PIPC6[9]	PIPC6[8]
		PIPC6[7]	PIPC6[6]	PIPC6[5]	PIPC6[4]	PIPC6[3]	PIPC6[2]	PIPC6[1]	PIPC6[0]
	PIPC7	PIPC7[15]	PIPC7[14]	PIPC7[13]	PIPC7[12]	PIPC7[11]	PIPC7[10]	PIPC7[9]	PIPC7[8]
		PIPC7[7]	PIPC7[6]	PIPC7[5]	PIPC7[4]	PIPC7[3]	PIPC7[2]	PIPC7[1]	PIPC7[0]
	PIPC8	PIPC8[15]	PIPC8[14]	PIPC8[13]	PIPC8[12]	PIPC8[11]	PIPC8[10]	PIPC8[9]	PIPC8[8]
		PIPC8[7]	PIPC8[6]	PIPC8[5]	PIPC8[4]	PIPC8[3]	PIPC8[2]	PIPC8[1]	PIPC8[0]
	PIPC9	-	-	-	-	-	-	-	-
		PIPC9[7]	PIPC9[6]	PIPC9[5]	PIPC9[4]	PIPC9[3]	PIPC9[2]	PIPC9[1]	PIPC9[0]
	PIPC10	PIPC10[15]	PIPC10[14]	PIPC10[13]	PIPC10[12]	PIPC10[11]	PIPC10[10]	PIPC10[9]	PIPC10[8]
		PIPC10[7]	PIPC10[6]	PIPC10[5]	PIPC10[4]	PIPC10[3]	PIPC10[2]	PIPC10[1]	PIPC10[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Ports	PIPC11	PIPC11[15]	PIPC11[14]	PIPC11[13]	PIPC11[12]	PIPC11[11]	PIPC11[10]	PIPC11[9]	PIPC11[8]
		PIPC11[7]	PIPC11[6]	PIPC11[5]	PIPC11[4]	PIPC11[3]	PIPC11[2]	PIPC11[1]	PIPC11[0]
	JPPR0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	JPPR0[1]	JPPR0[0]
	JPMC0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	JPMC0[1]	JPMC0[0]
	JPMCSR0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	JPMCSR0[17]	JPMCSR0[16]
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	JPMCSR0[1]	JPMCSR0[0]
	JPIBC0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	JPIBC0[1]	JPIBC0[0]
	SNCR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	ETSEL	SSI5NCE	SSI4NCE	SSI3NCE	SSI2NCE	SSI1NCE	SSI0NCE
Power-down modes	STBCR1	STBY	DEEP	-	-	-	-	-	
	STBCR2	HIZ	-	-	-	-	-	MSTP20	
	STBCR3	MSTP37	MSTP36	MSTP35	MSTP34	MSTP33	MSTP32	MSTP31	
	STBCR4	MSTP47	MSTP46	MSTP45	MSTP44	MSTP43	MSTP42	MSTP41	
	STBCR5	MSTP57	MSTP56	MSTP55	MSTP54	MSTP53	MSTP52	MSTP51	
	STBCR6	MSTP67	MSTP66	MSTP65	MSTP64	MSTP63	MSTP62	MSTP61	
	STBCR7	MSTP77	MSTP76	-	MSTP74	MSTP73	-	MSTP71	
	STBCR8	MSTP87	MSTP86	MSTP85	MSTP84	MSTP83	MSTP82	MSTP81	
	STBCR9	MSTP97	MSTP96	MSTP95	MSTP94	MSTP93	MSTP92	MSTP91	
	STBCR10	MSTP107	MSTP106	MSTP105	MSTP104	MSTP103	MSTP102	MSTP101	
	STBCR11	-	-	MSTP115	MSTP114	MSTP113	MSTP112	MSTP111	
	STBCR12	-	-	-	-	MSTP123	MSTP122	MSTP121	
	STBCR13	-	-	-	-	-	MSTP132	MSTP131	
	SWRSTCR1	AXTALE	SRST16	SRST15	SRST14	SRST13	SRST12	SRST11	
	SWRSTCR2	-	-	-	-	-	-	SRST21	
	SWRSTCR3	-	-	-	-	-	SRST32	-	
	SYSCR1	-	-	-	VFRAME4	VFRAME3	VFRAME2	VFRAME1	
	SYSCR2	-	-	-	VRAMWE4	VRAMWE3	VRAMWE2	VRAMWE1	
	SYSCR3	-	-	-	-	RRAMWE3	RRAMWE2	RRAMWE1	
	CPUSTS	-	-	-	IBUSY	-	-	-	
	STBRQ1	-	-	STBRQ15	-	STBRQ13	STBRQ12	-	
	STBRQ2	STBRQ27	STBRQ26	STBRQ25	STBRQ24	STBRQ23	STBRQ22	STBRQ21	
	STBACK1	-	-	STBAK15	-	STBAK13	STBAK12	-	
	STBACK2	STBAK27	STBAK26	STBAK25	STBAK24	STBAK23	STBAK22	STBAK21	
	RRAMKP	-	-	-	-	RRAMKP3	RRAMKP2	RRAMKP1	
	DSCTR	EBUSKEEPE	RAMBOOT	-	-	-	-	-	
	DSSSR	-	P6_2	P3_9	P3_1	P2_12	P8_7	P3_3	
		-	RTCAR	P6_4	P5_9	P7_8	P2_15	P9_1	
	DSESR	-	P6_2E	P3_9E	P3_1E	P2_12E	P8_7E	P3_3E	
		-	-	P6_4E	P5_9E	P7_8E	P2_15E	P9_1E	
	DSFR	IOKEEP	P6_2F	P3_9F	P3_1F	P2_12F	P8_7F	P3_3F	
		-	RTCARF	P6_4F	P5_9F	P7_8F	P2_15F	P9_1F	
XTALCTR	-	-	-	-	-	-	GAIN1		
Debugger interface	DAPROM_PERIPHID4	-	-	-	-	-	-		
		-	-	-	-	-	-		
		-	-	-	-	-	-		
		4KB_count[3]	4KB_count[2]	4KB_count[1]	4KB_count[0]	JEP106_code[3]	JEP106_code[2]	JEP106_code[1]	
	DAPROM_PERIPHID0	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	
		Part_Number[7]	Part_Number[6]	Part_Number[5]	Part_Number[4]	Part_Number[3]	Part_Number[2]	Part_Number[1]	
							Part_Number[0]		

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Debugger interface	DAPROM_PERIPHID1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		JEP106_id[3]	JEP106_id[2]	JEP106_id[1]	JEP106_id[0]	Part_Number[11]	Part_Number[10]	Part_Number[9]	Part_Number[8]
	DAPROM_PERIPHID2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Revision[3]	Revision[2]	Revision[1]	Revision[0]	JEDEC	JEP106_id[6]	JEP106_id[5]	JEP106_id[4]
	DAPROM_PERIPHID3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		RevAnd[3]	RevAnd[2]	RevAnd[1]	RevAnd[0]	CUSTOM[3]	CUSTOM[2]	CUSTOM[1]	CUSTOM[0]
	DAPROM_COMPID0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Preamble[7]	Preamble[6]	Preamble[5]	Preamble[4]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]
	DAPROM_COMPID1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Class[3]	Class[2]	Class[1]	Class[0]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]
	DAPROM_COMPID2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Preamble[7]	Preamble[6]	Preamble[5]	Preamble[4]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]
	DAPROM_COMPID3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Preamble[7]	Preamble[6]	Preamble[5]	Preamble[4]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]
	ICEREGMDRSTCTL	-	-	-	-	-	-	-	-
		-	-	-	NIDEN_CPU0	-	-	-	DBGEN_CPU0
		-	-	-	RSTRB_CPU0_DERSTZ	-	-	-	RSTRB_CPU0_CPURSTZ
		-	-	-	RSTRB_SYS_SYSRSTZ	-	RSTRB_CPU_PRSTDBGZ	RSTRB_CPU_SYSRSTZ	-
	ICEREGJTRCSEL	-	-	-	-	-	-	-	-
		PINSETEN	-	-	TRCMUX_SEL	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
	ICEREGCLKPWR CTRL	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		FAKEDBGCTRL	-	-	-	-	-	-	FAKEDBG
	ICEREGLOCKACCES	ICEREGLOCKACCES S[31]	ICEREGLOCKACCES S[30]	ICEREGLOCKACCES S[29]	ICEREGLOCKACCES S[28]	ICEREGLOCKACCES S[27]	ICEREGLOCKACCES S[26]	ICEREGLOCKACCES S[25]	ICEREGLOCKACCES S[24]
		ICEREGLOCKACCES S[23]	ICEREGLOCKACCES S[22]	ICEREGLOCKACCES S[21]	ICEREGLOCKACCES S[20]	ICEREGLOCKACCES S[19]	ICEREGLOCKACCES S[18]	ICEREGLOCKACCES S[17]	ICEREGLOCKACCES S[16]
		ICEREGLOCKACCES S[15]	ICEREGLOCKACCES S[14]	ICEREGLOCKACCES S[13]	ICEREGLOCKACCES S[12]	ICEREGLOCKACCES S[11]	ICEREGLOCKACCES S[10]	ICEREGLOCKACCES S[9]	ICEREGLOCKACCES S[8]
		ICEREGLOCKACCES S[7]	ICEREGLOCKACCES S[6]	ICEREGLOCKACCES S[5]	ICEREGLOCKACCES S[4]	ICEREGLOCKACCES S[3]	ICEREGLOCKACCES S[2]	ICEREGLOCKACCES S[1]	ICEREGLOCKACCES S[0]
	2ndDAPROM_PERIPHID4	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		4KB_count[3]	4KB_count[2]	4KB_count[1]	4KB_count[0]	JEP106_code[3]	JEP106_code[2]	JEP106_code[1]	JEP106_code[0]
	2ndDAPROM_PERIPHID0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Part_Number[7]	Part_Number[6]	Part_Number[5]	Part_Number[4]	Part_Number[3]	Part_Number[2]	Part_Number[1]	Part_Number[0]



Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Debugger interface	2ndDAPROM_PERIPHID1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		JEP106_id[3]	JEP106_id[2]	JEP106_id[1]	JEP106_id[0]	Part_Number [11]	Part_Number [10]	Part_Number[9]	Part_Number[8]
	2ndDAPROM_PERIPHID2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Revision[3]	Revision[2]	Revision[1]	Revision[0]	JEDEC	JEP106_id[6]	JEP106_id[5]	JEP106_id[4]
	2ndDAPROM_PERIPHID3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		RevAnd[3]	RevAnd[2]	RevAnd[1]	RevAnd[0]	CUSTOM[3]	CUSTOM[2]	CUSTOM[1]	CUSTOM[0]
	2ndDAPROM_COMPID0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Preamble[7]	Preamble[6]	Preamble[5]	Preamble[4]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]
	2ndDAPROM_COMPID1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Class[3]	Class[2]	Class[1]	Class[0]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]
	2ndDAPROM_COMPID2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Preamble[7]	Preamble[6]	Preamble[5]	Preamble[4]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]
	2ndDAPROM_COMPID3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Preamble[7]	Preamble[6]	Preamble[5]	Preamble[4]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]
	CPU ETF_RSZ	-	RSZ[30]	RSZ[29]	RSZ[28]	RSZ[27]	RSZ[26]	RSZ[25]	RSZ[24]
		RSZ[23]	RSZ[22]	RSZ[21]	RSZ[20]	RSZ[19]	RSZ[18]	RSZ[17]	RSZ[16]
		RSZ[15]	RSZ[14]	RSZ[13]	RSZ[12]	RSZ[11]	RSZ[10]	RSZ[9]	RSZ[8]
		RSZ[7]	RSZ[6]	RSZ[5]	RSZ[4]	RSZ[3]	RSZ[2]	RSZ[1]	RSZ[0]
	CPU ETF_STS	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	MemErr	Empty	FltEmpty	TMCReady	Triggered	Full
	CPU ETF_RRD	RRD[31]	RRD[30]	RRD[29]	RRD[28]	RRD[27]	RRD[26]	RRD[25]	RRD[24]
		RRD[23]	RRD[22]	RRD[21]	RRD[20]	RRD[19]	RRD[18]	RRD[17]	RRD[16]
		RRD[15]	RRD[14]	RRD[13]	RRD[12]	RRD[11]	RRD[10]	RRD[9]	RRD[8]
		RRD[7]	RRD[6]	RRD[5]	RRD[4]	RRD[3]	RRD[2]	RRD[1]	RRD[0]
	CPU ETF_RRP	RRP[31]	RRP[30]	RRP[29]	RRP[28]	RRP[27]	RRP[26]	RRP[25]	RRP[24]
		RRP[23]	RRP[22]	RRP[21]	RRP[20]	RRP[19]	RRP[18]	RRP[17]	RRP[16]
		RRP[15]	RRP[14]	RRP[13]	RRP[12]	RRP[11]	RRP[10]	RRP[9]	RRP[8]
		RRP[7]	RRP[6]	RRP[5]	RRP[4]	RRP[3]	RRP[2]	RRP[1]	RRP[0]
	CPU ETF_RWP	RWP[31]	RWP[30]	RWP[29]	RWP[28]	RWP[27]	RWP[26]	RWP[25]	RWP[24]
		RWP[23]	RWP[22]	RWP[21]	RWP[20]	RWP[19]	RWP[18]	RWP[17]	RWP[16]
		RWP[15]	RWP[14]	RWP[13]	RWP[12]	RWP[11]	RWP[10]	RWP[9]	RWP[8]
		RWP[7]	RWP[6]	RWP[5]	RWP[4]	RWP[3]	RWP[2]	RWP[1]	RWP[0]
	CPU ETF_TRG	TRG[31]	TRG[30]	TRG[29]	TRG[28]	TRG[27]	TRG[26]	TRG[25]	TRG[24]
		TRG[23]	TRG[22]	TRG[21]	TRG[20]	TRG[19]	TRG[18]	TRG[17]	TRG[16]
		TRG[15]	TRG[14]	TRG[13]	TRG[12]	TRG[11]	TRG[10]	TRG[9]	TRG[8]
		TRG[7]	TRG[6]	TRG[5]	TRG[4]	TRG[3]	TRG[2]	TRG[1]	TRG[0]
	CPU ETF_CTL	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	TraceCaptEn

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Debugger interface	CPU ETF_RWD	RWD[31]	RWD[30]	RWD[29]	RWD[28]	RWD[27]	RWD[26]	RWD[25]	RWD[24]	
		RWD[23]	RWD[22]	RWD[21]	RWD[20]	RWD[19]	RWD[18]	RWD[17]	RWD[16]	
		RWD[15]	RWD[14]	RWD[13]	RWD[12]	RWD[11]	RWD[10]	RWD[9]	RWD[8]	
		RWD[7]	RWD[6]	RWD[5]	RWD[4]	RWD[3]	RWD[2]	RWD[1]	RWD[0]	
	CPU ETF_MODE	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	MODE[1]	MODE[0]
	CPU ETF_LBUFLEVEL	LBUFLEVEL[31]	LBUFLEVEL[30]	LBUFLEVEL[29]	LBUFLEVEL[28]	LBUFLEVEL[27]	LBUFLEVEL[26]	LBUFLEVEL[25]	LBUFLEVEL[24]	LBUFLEVEL[23]
		LBUFLEVEL[23]	LBUFLEVEL[22]	LBUFLEVEL[21]	LBUFLEVEL[20]	LBUFLEVEL[19]	LBUFLEVEL[18]	LBUFLEVEL[17]	LBUFLEVEL[16]	LBUFLEVEL[15]
		LBUFLEVEL[15]	LBUFLEVEL[14]	LBUFLEVEL[13]	LBUFLEVEL[12]	LBUFLEVEL[11]	LBUFLEVEL[10]	LBUFLEVEL[9]	LBUFLEVEL[8]	LBUFLEVEL[7]
		LBUFLEVEL[7]	LBUFLEVEL[6]	LBUFLEVEL[5]	LBUFLEVEL[4]	LBUFLEVEL[3]	LBUFLEVEL[2]	LBUFLEVEL[1]	LBUFLEVEL[0]	-
	CPU ETF_CBUFLEVEL	CBUFLEVEL[31]	CBUFLEVEL[30]	CBUFLEVEL[29]	CBUFLEVEL[28]	CBUFLEVEL[27]	CBUFLEVEL[26]	CBUFLEVEL[25]	CBUFLEVEL[24]	CBUFLEVEL[23]
		CBUFLEVEL[23]	CBUFLEVEL[22]	CBUFLEVEL[21]	CBUFLEVEL[20]	CBUFLEVEL[19]	CBUFLEVEL[18]	CBUFLEVEL[17]	CBUFLEVEL[16]	CBUFLEVEL[15]
		CBUFLEVEL[15]	CBUFLEVEL[14]	CBUFLEVEL[13]	CBUFLEVEL[12]	CBUFLEVEL[11]	CBUFLEVEL[10]	CBUFLEVEL[9]	CBUFLEVEL[8]	CBUFLEVEL[7]
		CBUFLEVEL[7]	CBUFLEVEL[6]	CBUFLEVEL[5]	CBUFLEVEL[4]	CBUFLEVEL[3]	CBUFLEVEL[2]	CBUFLEVEL[1]	CBUFLEVEL[0]	-
	CPU ETF_BUFWM	BUFWM[31]	BUFWM[30]	BUFWM[29]	BUFWM[28]	BUFWM[27]	BUFWM[26]	BUFWM[25]	BUFWM[24]	BUFWM[23]
		BUFWM[23]	BUFWM[22]	BUFWM[21]	BUFWM[20]	BUFWM[19]	BUFWM[18]	BUFWM[17]	BUFWM[16]	BUFWM[15]
		BUFWM[15]	BUFWM[14]	BUFWM[13]	BUFWM[12]	BUFWM[11]	BUFWM[10]	BUFWM[9]	BUFWM[8]	BUFWM[7]
		BUFWM[7]	BUFWM[6]	BUFWM[5]	BUFWM[4]	BUFWM[3]	BUFWM[2]	BUFWM[1]	BUFWM[0]	-
	CPU ETF_RRPHI	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		RRPHI[7]	RRPHI[6]	RRPHI[5]	RRPHI[4]	RRPHI[3]	RRPHI[2]	RRPHI[1]	RRPHI[0]	-
	CPU ETF_RWPHI	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		RWPHI[7]	RWPHI[6]	RWPHI[5]	RWPHI[4]	RWPHI[3]	RWPHI[2]	RWPHI[1]	RWPHI[0]	-
	CPU ETF_FFISR	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	FiStopped	FiInProg	-
	CPU ETF_FFCCR	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	DrainBuffer	StopOnTrigEvt	StopOnFI	-	TrigOnFI	TrigOnTrigEvt	TrigOnTrigIn	-
		-	FlushMan	FOnTrigEvt	FOnFIIn	-	-	EnTI	EnFt	-
	CPU ETF_PSCR	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	PSCnt[4]	PSCnt[3]	PSCnt[2]	PSCnt[1]	PSCnt[0]	-
	CPU ETF_CLAIMSET	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	CLAIMSET[3]	CLAIMSET[2]	CLAIMSET[1]	CLAIMSET[0]	-
	CPU ETF_CLAIMCLR	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	CLAIMCLR[3]	CLAIMCLR[2]	CLAIMCLR[1]	CLAIMCLR[0]	-
	CPU ETF_LAR	ACCESS_W[31]	ACCESS_W[30]	ACCESS_W[29]	ACCESS_W[28]	ACCESS_W[27]	ACCESS_W[26]	ACCESS_W[25]	ACCESS_W[24]	ACCESS_W[23]
		ACCESS_W[23]	ACCESS_W[22]	ACCESS_W[21]	ACCESS_W[20]	ACCESS_W[19]	ACCESS_W[18]	ACCESS_W[17]	ACCESS_W[16]	ACCESS_W[15]
		ACCESS_W[15]	ACCESS_W[14]	ACCESS_W[13]	ACCESS_W[12]	ACCESS_W[11]	ACCESS_W[10]	ACCESS_W[9]	ACCESS_W[8]	ACCESS_W[7]
		ACCESS_W[7]	ACCESS_W[6]	ACCESS_W[5]	ACCESS_W[4]	ACCESS_W[3]	ACCESS_W[2]	ACCESS_W[1]	ACCESS_W[0]	-
	CPU ETF_LSR	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	LOCKTYPE	LOCKGRANT	LOCKEXIST	-

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Debugger interface	CPU ETF AUTHSTATUS	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		SNID[1]	SNID[0]	SID[1]	SID[0]	NSNID[1]	NSNID[0]	NSID[1]	NSID[0]
	CPU ETF_DEVID	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	WBUF_DEPTH[2]	WBUF_DEPTH[1]	WBUF_DEPTH[0]	MEMWIDTH[2]	MEMWIDTH[1]	MEMWIDTH[0]
		CONFIGTYPE[1]	CONFIGTYPE[0]	CLKSCHEME	ATBINPORT COUNT[4]	ATBINPORT COUNT[3]	ATBINPORT COUNT[2]	ATBINPORT COUNT[1]	ATBINPORT COUNT[0]
	CPU ETF_DEVTYPE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Sub_type[3]	Sub_type[2]	Sub_type[1]	Sub_type[0]	Major_type[3]	Major_type[2]	Major_type[1]	Major_type[0]
	CPU ETF_PERIPHID4	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		4KB_count[3]	4KB_count[2]	4KB_count[1]	4KB_count[0]	JEP106_code[3]	JEP106_code[2]	JEP106_code[1]	JEP106_code[0]
	CPU ETF_PERIPHID0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Part_Number[7]	Part_Number[6]	Part_Number[5]	Part_Number[4]	Part_Number[3]	Part_Number[2]	Part_Number[1]	Part_Number[0]
	CPU ETF_PERIPHID1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		JEP106_id[3]	JEP106_id[2]	JEP106_id[1]	JEP106_id[0]	Part_Number[11]	Part_Number[10]	Part_Number[9]	Part_Number[8]
	CPU ETF_PERIPHID2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Revision[3]	Revision[2]	Revision[1]	Revision[0]	JEDEC	JEP106_id[6]	JEP106_id[5]	JEP106_id[4]
	CPU ETF_PERIPHID3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		RevAnd[3]	RevAnd[2]	RevAnd[1]	RevAnd[0]	Customer Modified[3]	Customer Modified[2]	Customer Modified[1]	Customer Modified[0]
	CPU ETF_COMPID0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Preamble[7]	Preamble[6]	Preamble[5]	Preamble[4]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]
	CPU ETF_COMPID1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Class[3]	Class[2]	Class[1]	Class[0]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]
	CPU ETF_COMPID2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Preamble[7]	Preamble[6]	Preamble[5]	Preamble[4]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]
	CPU ETF_COMPID3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Preamble[7]	Preamble[6]	Preamble[5]	Preamble[4]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]
	CPU_CTICS_CTICONTROL	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GLBEN

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Debugger interface	CPU_CTICS_CTIINTACK	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		INTACK[7]	INTACK[6]	INTACK[5]	INTACK[4]	INTACK[3]	INTACK[2]	INTACK[1]	INTACK[0]
	CPU_CTICS_CTIAPPSET	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	APPSET[3]	APPSET[2]	APPSET[1]	APPSET[0]
	CPU_CTICS_CTIAPPCLEAR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	APPCLEAR[3]	APPCLEAR[2]	APPCLEAR[1]	APPCLEAR[0]
	CPU_CTICS_CTIAPPULSE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	APPULSE[3]	APPULSE[2]	APPULSE[1]	APPULSE[0]
	CPU_CTICS_CTIINEN0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	TRIGINEN[3]	TRIGINEN[2]	TRIGINEN[1]	TRIGINEN[0]
	CPU_CTICS_CTIINEN1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	TRIGINEN[3]	TRIGINEN[2]	TRIGINEN[1]	TRIGINEN[0]
	CPU_CTICS_CTIINEN2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	TRIGINEN[3]	TRIGINEN[2]	TRIGINEN[1]	TRIGINEN[0]
	CPU_CTICS_CTIINEN3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	TRIGINEN[3]	TRIGINEN[2]	TRIGINEN[1]	TRIGINEN[0]
	CPU_CTICS_CTIINEN4	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	TRIGINEN[3]	TRIGINEN[2]	TRIGINEN[1]	TRIGINEN[0]
	CPU_CTICS_CTIINEN5	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	TRIGINEN[3]	TRIGINEN[2]	TRIGINEN[1]	TRIGINEN[0]
	CPU_CTICS_CTIINEN6	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	TRIGINEN[3]	TRIGINEN[2]	TRIGINEN[1]	TRIGINEN[0]
	CPU_CTICS_CTIINEN7	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	TRIGINEN[3]	TRIGINEN[2]	TRIGINEN[1]	TRIGINEN[0]
	CPU_CTICS_CTIOUTEN0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	TRIGOUTEN[3]	TRIGOUTEN[2]	TRIGOUTEN[1]	TRIGOUTEN[0]
	CPU_CTICS_CTIOUTEN1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	TRIGOUTEN[3]	TRIGOUTEN[2]	TRIGOUTEN[1]	TRIGOUTEN[0]

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Debugger interface	CPU_CTICS_CTIOUTEN2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	TRIGOUTEN[3]	TRIGOUTEN[2]	TRIGOUTEN[1]	TRIGOUTEN[0]
	CPU_CTICS_CTIOUTEN3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	TRIGOUTEN[3]	TRIGOUTEN[2]	TRIGOUTEN[1]	TRIGOUTEN[0]
	CPU_CTICS_CTIOUTEN4	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	TRIGOUTEN[3]	TRIGOUTEN[2]	TRIGOUTEN[1]	TRIGOUTEN[0]
	CPU_CTICS_CTIOUTEN5	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	TRIGOUTEN[3]	TRIGOUTEN[2]	TRIGOUTEN[1]	TRIGOUTEN[0]
	CPU_CTICS_CTIOUTEN6	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	TRIGOUTEN[3]	TRIGOUTEN[2]	TRIGOUTEN[1]	TRIGOUTEN[0]
	CPU_CTICS_CTIOUTEN7	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	TRIGOUTEN[3]	TRIGOUTEN[2]	TRIGOUTEN[1]	TRIGOUTEN[0]
	CPU_CTICS_CTIIRIGINSTATUS	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		TRIGINSTATUS[7]	TRIGINSTATUS[6]	TRIGINSTATUS[5]	TRIGINSTATUS[4]	TRIGINSTATUS[3]	TRIGINSTATUS[2]	TRIGINSTATUS[1]	TRIGINSTATUS[0]
	CPU_CTICS_CTIIRIGOUT STATUS	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		TRIGOUT STATUS[7]	TRIGOUT STATUS[6]	TRIGOUT STATUS[5]	TRIGOUT STATUS[4]	TRIGOUT STATUS[3]	TRIGOUT STATUS[2]	TRIGOUT STATUS[1]	TRIGOUT STATUS[0]
	CPU_CTICS_CTIICHINSTATUS	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		CTCHINSTATUS[3]	CTCHINSTATUS[2]	CTCHINSTATUS[1]	CTCHINSTATUS[0]				
	CPU_CTICS_CTIICHOUTSTATUS	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		CTCHOUT STATUS[3]	CTCHOUT STATUS[2]	CTCHOUT STATUS[1]	CTCHOUT STATUS[0]				
	CPU_CTICS_CTIIGATE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		CTIGATEEN3	CTIGATEEN2	CTIGATEEN1	CTIGATEEN0				
	CPU_CTICS_ASICCTL	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		ASICCTL[3]	ASICCTL[2]	ASICCTL[1]	ASICCTL[0]				
	CPU_CTICS_CLAIMSET	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		CLAIMSET[3]	CLAIMSET[2]	CLAIMSET[1]	CLAIMSET[0]				

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Debugger interface	CPU_CTICS_CLAIMCLR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	CLAIMCLR[3]	CLAIMCLR[2]	CLAIMCLR[1]	CLAIMCLR[0]
	CPU_CTICS_LAR	ACCESS_W[31]	ACCESS_W[30]	ACCESS_W[29]	ACCESS_W[28]	ACCESS_W[27]	ACCESS_W[26]	ACCESS_W[25]	ACCESS_W[24]
		ACCESS_W[23]	ACCESS_W[22]	ACCESS_W[21]	ACCESS_W[20]	ACCESS_W[19]	ACCESS_W[18]	ACCESS_W[17]	ACCESS_W[16]
		ACCESS_W[15]	ACCESS_W[14]	ACCESS_W[13]	ACCESS_W[12]	ACCESS_W[11]	ACCESS_W[10]	ACCESS_W[9]	ACCESS_W[8]
		ACCESS_W[7]	ACCESS_W[6]	ACCESS_W[5]	ACCESS_W[4]	ACCESS_W[3]	ACCESS_W[2]	ACCESS_W[1]	ACCESS_W[0]
	CPU_CTICS_LSR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	LOCKTYPE	LOCKGRANT	LOCKEXIST
	CPU_CTICS_AUTHSTATUS	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		SNID[1]	SNID[0]	SID[1]	SID[0]	NSNID[1]	NSNID[0]	NSID[1]	NSID[0]
	CPU_CTICS_DEVID	-	-	-	-	-	-	-	-
		-	-	-	-	CHANWIDTH[3]	CHANWIDTH[2]	CHANWIDTH[1]	CHANWIDTH[0]
		TRIGWIDTH[7]	TRIGWIDTH[6]	TRIGWIDTH[5]	TRIGWIDTH[4]	TRIGWIDTH[3]	TRIGWIDTH[2]	TRIGWIDTH[1]	TRIGWIDTH[0]
		-	-	-	-	EXTMUXNUM[3]	EXTMUXNUM[2]	EXTMUXNUM[1]	EXTMUXNUM[0]
	CPU_CTICS_DEVTYPE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Sub_type[3]	Sub_type[2]	Sub_type[1]	Sub_type[0]	Major_type[3]	Major_type[2]	Major_type[1]	Major_type[0]
	CPU_CTICS_PERIPHID4	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		4KB_count[3]	4KB_count[2]	4KB_count[1]	4KB_count[0]	JEP106_code[3]	JEP106_code[2]	JEP106_code[1]	JEP106_code[0]
	CPU_CTICS_PERIPHID0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Part_Number[7]	Part_Number[6]	Part_Number[5]	Part_Number[4]	Part_Number[3]	Part_Number[2]	Part_Number[1]	Part_Number[0]
	CPU_CTICS_PERIPHID1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		JEP106_id[3]	JEP106_id[2]	JEP106_id[1]	JEP106_id[0]	Part_Number [11]	Part_Number [10]	Part_Number[9]	Part_Number[8]
	CPU_CTICS_PERIPHID2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Revision[3]	Revision[2]	Revision[1]	Revision[0]	JEDEC	JEP106_id[6]	JEP106_id[5]	JEP106_id[4]
	CPU_CTICS_PERIPHID3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		RevAnd[3]	RevAnd[2]	RevAnd[1]	RevAnd[0]	CUSTOM[3]	CUSTOM[2]	CUSTOM[1]	CUSTOM[0]
	CPU_CTICS_COMPID0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Preamble[7]	Preamble[6]	Preamble[5]	Preamble[4]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]
	CPU_CTICS_COMPID1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Class[3]	Class[2]	Class[1]	Class[0]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]
	CPU_CTICS_COMPID2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Preamble[7]	Preamble[6]	Preamble[5]	Preamble[4]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Debugger interface	CPU_CTICS_COMPID3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Preamble[7]	Preamble[6]	Preamble[5]	Preamble[4]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]
	CPU_TPIU_Supportedportsizes	SIZE[31]	SIZE[30]	SIZE[29]	SIZE[28]	SIZE[27]	SIZE[26]	SIZE[25]	SIZE[24]
		SIZE[23]	SIZE[22]	SIZE[21]	SIZE[20]	SIZE[19]	SIZE[18]	SIZE[17]	SIZE[16]
		SIZE[15]	SIZE[14]	SIZE[13]	SIZE[12]	SIZE[11]	SIZE[10]	SIZE[9]	SIZE[8]
		SIZE[7]	SIZE[6]	SIZE[5]	SIZE[4]	SIZE[3]	SIZE[2]	SIZE[1]	SIZE[0]
	CPU_TPIU_Currentportsize	SIZE[31]	SIZE[30]	SIZE[29]	SIZE[28]	SIZE[27]	SIZE[26]	SIZE[25]	SIZE[24]
		SIZE[23]	SIZE[22]	SIZE[21]	SIZE[20]	SIZE[19]	SIZE[18]	SIZE[17]	SIZE[16]
		SIZE[15]	SIZE[14]	SIZE[13]	SIZE[12]	SIZE[11]	SIZE[10]	SIZE[9]	SIZE[8]
		SIZE[7]	SIZE[6]	SIZE[5]	SIZE[4]	SIZE[3]	SIZE[2]	SIZE[1]	SIZE[0]
	CPU_TPIU_Supportedtrigge r modes	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	TrgRun	Triggered
		-	-	-	-	-	-	-	TCount8
		-	-	-	Multipliers[4]	Multipliers[3]	Multipliers[2]	Multipliers[1]	Multipliers[0]
	CPU_TPIU_TriggerCounter value	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		TrigCount[7]	TrigCount[6]	TrigCount[5]	TrigCount[4]	TrigCount[3]	TrigCount[2]	TrigCount[1]	TrigCount[0]
	CPU_TPIU_Triggermultiplier	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	TrgRun	Triggered
		-	-	-	-	-	-	-	TCount8
		-	-	-	Multipliers[4]	Multipliers[3]	Multipliers[2]	Multipliers[1]	Multipliers[0]
	CPU_TPIU_Supportedtestpattern/ modes	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	Mode[1]	Mode[0]
		-	-	-	-	-	-	-	-
		-	-	-	-	Pattern[3]	Pattern[2]	Pattern[1]	Pattern[0]
	CPU_TPIU_Currenttestpattern/ mode	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	Mode[1]	Mode[0]
		-	-	-	-	-	-	-	-
		-	-	-	-	Pattern[3]	Pattern[2]	Pattern[1]	Pattern[0]
	CPU_TPIU_Testpatternrepeat counter	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		PattCount[7]	PattCount[6]	PattCount[5]	PattCount[4]	PattCount[3]	PattCount[2]	PattCount[1]	PattCount[0]
	CPU_TPIU_Formatterandflush status	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	Mode[1]	Mode[0]
		-	-	-	-	-	-	-	-
		-	-	StopTrig	StopFl	-	TCP	FTS	FIP
	CPU_TPIU_Formatterandflush control	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	TrigFl	TrigEvt	TrigIn
		-	FOnMan	FOnTrig	FOnFlIn	-	-	EnFCont	EnFTC
	CPU_TPIU_Formatter synchronization counter	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	CycCount[11]	CycCount[10]	CycCount[9]	CycCount[8]
		CycCount[7]	CycCount[6]	CycCount[5]	CycCount[4]	CycCount[3]	CycCount[2]	CycCount[1]	CycCount[0]
	CPU_TPIU_CLAIMSET	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	CLAIMSET[3]	CLAIMSET[2]	CLAIMSET[1]	CLAIMSET[0]
	CPU_TPIU_CLAIMCLR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	CLAIMCLR[3]	CLAIMCLR[2]	CLAIMCLR[1]	CLAIMCLR[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Debugger interface	CPU_TPIU_LAR	ACCESS_W[31]	ACCESS_W[30]	ACCESS_W[29]	ACCESS_W[28]	ACCESS_W[27]	ACCESS_W[26]	ACCESS_W[25]	ACCESS_W[24]
		ACCESS_W[23]	ACCESS_W[22]	ACCESS_W[21]	ACCESS_W[20]	ACCESS_W[19]	ACCESS_W[18]	ACCESS_W[17]	ACCESS_W[16]
		ACCESS_W[15]	ACCESS_W[14]	ACCESS_W[13]	ACCESS_W[12]	ACCESS_W[11]	ACCESS_W[10]	ACCESS_W[9]	ACCESS_W[8]
		ACCESS_W[7]	ACCESS_W[6]	ACCESS_W[5]	ACCESS_W[4]	ACCESS_W[3]	ACCESS_W[2]	ACCESS_W[1]	ACCESS_W[0]
	CPU_TPIU_LSR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	LOCKTYPE	LOCKGRANT	LOCKEXIST
	CPU_TPIU_AUTHSTATUS	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		SNID[1]	SNID[0]	SID[1]	SID[0]	NSNID[1]	NSNID[0]	NSID[1]	NSID[0]
	CPU_TPIU_DEVID	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	ID[12]	ID[11]	ID[10]	ID[9]	ID[8]
		ID[7]	ID[6]	ID[5]	ID[4]	ID[3]	ID[2]	ID[1]	ID[0]
	CPU_TPIU_DEVTYP	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Sub_type[3]	Sub_type[2]	Sub_type[1]	Sub_type[0]	Main_type[3]	Main_type[2]	Main_type[1]	Main_type[0]
	CPU_TPIU_PERIPHID4	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		4KB_count[3]	4KB_count[2]	4KB_count[1]	4KB_count[0]	JEP106_code[3]	JEP106_code[2]	JEP106_code[1]	JEP106_code[0]
	CPU_TPIU_PERIPHID0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Part_Number[7]	Part_Number[6]	Part_Number[5]	Part_Number[4]	Part_Number[3]	Part_Number[2]	Part_Number[1]	Part_Number[0]
	CPU_TPIU_PERIPHID1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		JEP106_id[3]	JEP106_id[2]	JEP106_id[1]	JEP106_id[0]	Part_Number[11]	Part_Number[10]	Part_Number[9]	Part_Number[8]
	CPU_TPIU_PERIPHID2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Revision[3]	Revision[2]	Revision[1]	Revision[0]	JEDEC	JEP106_id[6]	JEP106_id[5]	JEP106_id[4]
	CPU_TPIU_PERIPHID3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		RevAnd[3]	RevAnd[2]	RevAnd[1]	RevAnd[0]	Customer Modified[3]	Customer Modified[2]	Customer Modified[1]	Customer Modified[0]
	CPU_TPIU_COMPID0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Preamble[7]	Preamble[6]	Preamble[5]	Preamble[4]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]
	CPU_TPIU_COMPID1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Class[3]	Class[2]	Class[1]	Class[0]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]
	CPU_TPIU_COMPID2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Preamble[7]	Preamble[6]	Preamble[5]	Preamble[4]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]
	CPU_TPIU_COMPID3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Preamble[7]	Preamble[6]	Preamble[5]	Preamble[4]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]



Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Debugger interface	CPU_TraceFunnel_FUNCCTL	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	HT[3]	HT[2]	HT[1]	HT[0]
		EnS7	EnS6	EnS5	EnS4	EnS3	EnS2	EnS1	EnS0
	CPU_TraceFunnel_PRICTL	-	-	-	-	-	-	-	-
		PriPort7[2]	PriPort7[1]	PriPort7[0]	PriPort6[2]	PriPort6[1]	PriPort6[0]	PriPort5[2]	PriPort5[1]
		PriPort5[0]	PriPort4[2]	PriPort4[1]	PriPort4[0]	PriPort3[2]	PriPort3[1]	PriPort3[0]	PriPort2[2]
	CPU_TraceFunnel_CLAIMSET	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	CLAIMSET[3]	CLAIMSET[2]	CLAIMSET[1]	CLAIMSET[0]
	CPU_TraceFunnel_CLAIMCLR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	CLAIMCLR[3]	CLAIMCLR[2]	CLAIMCLR[1]	CLAIMCLR[0]
	CPU_TraceFunnel_LAR	ACCESS_W[31]	ACCESS_W[30]	ACCESS_W[29]	ACCESS_W[28]	ACCESS_W[27]	ACCESS_W[26]	ACCESS_W[25]	ACCESS_W[24]
		ACCESS_W[23]	ACCESS_W[22]	ACCESS_W[21]	ACCESS_W[20]	ACCESS_W[19]	ACCESS_W[18]	ACCESS_W[17]	ACCESS_W[16]
		ACCESS_W[15]	ACCESS_W[14]	ACCESS_W[13]	ACCESS_W[12]	ACCESS_W[11]	ACCESS_W[10]	ACCESS_W[9]	ACCESS_W[8]
		ACCESS_W[7]	ACCESS_W[6]	ACCESS_W[5]	ACCESS_W[4]	ACCESS_W[3]	ACCESS_W[2]	ACCESS_W[1]	ACCESS_W[0]
	CPU_TraceFunnel_LSR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	LOCKTYPE	LOCKGRANT	LOCKEXIST
	CPU_TraceFunnel_AUTHSTATUS	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		SNID[1]	SNID[0]	SID[1]	SID[0]	NSNID[1]	NSNID[0]	NSID[1]	NSID[0]
	CPU_TraceFunnel_DEVID	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		ID[7]	ID[6]	ID[5]	ID[4]	ID[3]	ID[2]	ID[1]	ID[0]
	CPU_TraceFunnel_DEVTYPE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		TYPE[7]	TYPE[6]	TYPE[5]	TYPE[4]	TYPE[3]	TYPE[2]	TYPE[1]	TYPE[0]
	CPU_TraceFunnel_PERIPHID4	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		4KB_count[3]	4KB_count[2]	4KB_count[1]	4KB_count[0]	JEP106_code[3]	JEP106_code[2]	JEP106_code[1]	JEP106_code[0]
	CPU_TraceFunnel_PERIPHID0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Part_Number[7]	Part_Number[6]	Part_Number[5]	Part_Number[4]	Part_Number[3]	Part_Number[2]	Part_Number[1]	Part_Number[0]
	CPU_TraceFunnel_PERIPHID1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		JEP106_id[3]	JEP106_id[2]	JEP106_id[1]	JEP106_id[0]	Part_Number[11]	Part_Number[10]	Part_Number[9]	Part_Number[8]
	CPU_TraceFunnel_PERIPHID2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Revision[3]	Revision[2]	Revision[1]	Revision[0]	JEDEC	JEP106_id[6]	JEP106_id[5]	JEP106_id[4]
	CPU_TraceFunnel_PERIPHID3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		RevAnd[3]	RevAnd[2]	RevAnd[1]	RevAnd[0]	CUSTOM[3]	CUSTOM[2]	CUSTOM[1]	CUSTOM[0]

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Debugger interface	CPU_TraceFunnel_COMPID0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Preamble[7]	Preamble[6]	Preamble[5]	Preamble[4]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]
	CPU_TraceFunnel_COMPID1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Class[3]	Class[2]	Class[1]	Class[0]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]
	CPU_TraceFunnel_COMPID2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Preamble[7]	Preamble[6]	Preamble[5]	Preamble[4]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]
	CPU_TraceFunnel_COMPID3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Preamble[7]	Preamble[6]	Preamble[5]	Preamble[4]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]
	CA9_DBG_DBGDIDR	WRPs[3]	WRPs[2]	WRPs[1]	WRPs[0]	BRPs[3]	BRPs[2]	BRPs[1]	BRPs[0]
		CTX_CMPs[3]	CTX_CMPs[2]	CTX_CMPs[1]	CTX_CMPs[0]	Version[3]	Version[2]	Version[1]	Version[0]
		DEVID_imp	nSUHD_imp	PCSR_imp	SE_imp	-	-	-	-
		Variant[3]	Variant[2]	Variant[1]	Variant[0]	Revision[3]	Revision[2]	Revision[1]	Revision[0]
	CA9_DBG_DBGWFAR	ADDRESS[31]	ADDRESS[30]	ADDRESS[29]	ADDRESS[28]	ADDRESS[27]	ADDRESS[26]	ADDRESS[25]	ADDRESS[24]
		ADDRESS[23]	ADDRESS[22]	ADDRESS[21]	ADDRESS[20]	ADDRESS[19]	ADDRESS[18]	ADDRESS[17]	ADDRESS[16]
		ADDRESS[15]	ADDRESS[14]	ADDRESS[13]	ADDRESS[12]	ADDRESS[11]	ADDRESS[10]	ADDRESS[9]	ADDRESS[8]
		ADDRESS[7]	ADDRESS[6]	ADDRESS[5]	ADDRESS[4]	ADDRESS[3]	ADDRESS[2]	ADDRESS[1]	ADDRESS[0]
	CA9_DBG_DBGVCR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		FIQ	IRQ	-	DataAbort	PrefetchAbort	SVC	Undef	Reset
	CA9_DBG_DBGDTRRNext	Host_to_target_data[31]	Host_to_target_data[30]	Host_to_target_data[29]	Host_to_target_data[28]	Host_to_target_data[27]	Host_to_target_data[26]	Host_to_target_data[25]	Host_to_target_data[24]
		Host_to_target_data[23]	Host_to_target_data[22]	Host_to_target_data[21]	Host_to_target_data[20]	Host_to_target_data[19]	Host_to_target_data[18]	Host_to_target_data[17]	Host_to_target_data[16]
		Host_to_target_data[15]	Host_to_target_data[14]	Host_to_target_data[13]	Host_to_target_data[12]	Host_to_target_data[11]	Host_to_target_data[10]	Host_to_target_data[9]	Host_to_target_data[8]
		Host_to_target_data[7]	Host_to_target_data[6]	Host_to_target_data[5]	Host_to_target_data[4]	Host_to_target_data[3]	Host_to_target_data[2]	Host_to_target_data[1]	Host_to_target_data[0]
	CA9_DBG_DBGITR	Instruction[31]	Instruction[30]	Instruction[29]	Instruction[28]	Instruction[27]	Instruction[26]	Instruction[25]	Instruction[24]
		Instruction[23]	Instruction[22]	Instruction[21]	Instruction[20]	Instruction[19]	Instruction[18]	Instruction[17]	Instruction[16]
		Instruction[15]	Instruction[14]	Instruction[13]	Instruction[12]	Instruction[11]	Instruction[10]	Instruction[9]	Instruction[8]
		Instruction[7]	Instruction[6]	Instruction[5]	Instruction[4]	Instruction[3]	Instruction[2]	Instruction[1]	Instruction[0]
	CA9_DBG_DBGPCSR	PC[29]	PC[28]	PC[27]	PC[26]	PC[25]	PC[24]	PC[23]	PC[22]
		PC[21]	PC[20]	PC[19]	PC[18]	PC[17]	PC[16]	PC[15]	PC[14]
		PC[13]	PC[12]	PC[11]	PC[10]	PC[9]	PC[8]	PC[7]	PC[6]
		PC[5]	PC[4]	PC[3]	PC[2]	PC[1]	PC[0]	Meaning_of_PC[1]	Meaning_of_PC[0]
CA9_DBG_DBGDSCRExt	-	Rxfull	Txfull	-	RXfull_I	TXfull_I	PipeAdv	InstrCompl_I	
	-	-	ExtDCCmode[1]	ExtDCCmode[0]	ADAdiscard	NS	SPNIDdis	SPIDdis	
	MDBGGen	HDBGGen	ITRen	UDCCdis	INTdis	DBGack	-	UND_I	
	ADABORT_I	SDABORT_I	MOE[3]	MOE[2]	MOE[1]	MOE[0]	RESTARTED	HALTED	
CA9_DBG_DBGDTRTXext	Target_to_host_data[31]	Target_to_host_data[30]	Target_to_host_data[29]	Target_to_host_data[28]	Target_to_host_data[27]	Target_to_host_data[26]	Target_to_host_data[25]	Target_to_host_data[24]	
	Target_to_host_data[23]	Target_to_host_data[22]	Target_to_host_data[21]	Target_to_host_data[20]	Target_to_host_data[19]	Target_to_host_data[18]	Target_to_host_data[17]	Target_to_host_data[16]	
	Target_to_host_data[15]	Target_to_host_data[14]	Target_to_host_data[13]	Target_to_host_data[12]	Target_to_host_data[11]	Target_to_host_data[10]	Target_to_host_data[9]	Target_to_host_data[8]	
	Target_to_host_data[7]	Target_to_host_data[6]	Target_to_host_data[5]	Target_to_host_data[4]	Target_to_host_data[3]	Target_to_host_data[2]	Target_to_host_data[1]	Target_to_host_data[0]	
CA9_DBG_DBGDRCR	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	
	-	-	-	-	-	-	-	-	
	-	-	-	Cancel BIU Requests	Clear Sticky Pipeline Advanceflag	Clear Sticky Exceptions flags	Restart request	Halt request	



Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Debugger interface	CA9_DBG_DBGWVR2	Watchpoint[31]	Watchpoint[30]	Watchpoint[29]	Watchpoint[28]	Watchpoint[27]	Watchpoint[26]	Watchpoint[25]	Watchpoint[24]
		Watchpoint[23]	Watchpoint[22]	Watchpoint[21]	Watchpoint[20]	Watchpoint[19]	Watchpoint[18]	Watchpoint[17]	Watchpoint[16]
		Watchpoint[15]	Watchpoint[14]	Watchpoint[13]	Watchpoint[12]	Watchpoint[11]	Watchpoint[10]	Watchpoint[9]	Watchpoint[8]
		Watchpoint[7]	Watchpoint[6]	Watchpoint[5]	Watchpoint[4]	Watchpoint[3]	Watchpoint[2]	Watchpoint[1]	Watchpoint[0]
	CA9_DBG_DBGWVR3	Watchpoint[31]	Watchpoint[30]	Watchpoint[29]	Watchpoint[28]	Watchpoint[27]	Watchpoint[26]	Watchpoint[25]	Watchpoint[24]
		Watchpoint[23]	Watchpoint[22]	Watchpoint[21]	Watchpoint[20]	Watchpoint[19]	Watchpoint[18]	Watchpoint[17]	Watchpoint[16]
		Watchpoint[15]	Watchpoint[14]	Watchpoint[13]	Watchpoint[12]	Watchpoint[11]	Watchpoint[10]	Watchpoint[9]	Watchpoint[8]
		Watchpoint[7]	Watchpoint[6]	Watchpoint[5]	Watchpoint[4]	Watchpoint[3]	Watchpoint[2]	Watchpoint[1]	Watchpoint[0]
	CA9_DBG_DBGWCR0	-	-	-	address_mask[4]	address_mask[3]	address_mask[2]	address_mask[1]	address_mask[0]
		-	-	-	E	Linked BRP[3]	Linked BRP[2]	Linked BRP[1]	Linked BRP[0]
		Sec_state_cont[1]	Sec_state_cont[0]	-	-	-	-	-	Byte Add_sel[3]
		Byte add_sel[2]	Byte Add_sel[1]	Byte Add_sel[0]	L/S[1]	L/S[0]	SP[1]	SP[0]	W
	CA9_DBG_DBGWCR1	-	-	-	address_mask[4]	address_mask[3]	address_mask[2]	address_mask[1]	address_mask[0]
		-	-	-	E	Linked BRP[3]	Linked BRP[2]	Linked BRP[1]	Linked BRP[0]
		Sec_state_cont[1]	Sec_state_cont[0]	-	-	-	-	-	Byte Add_sel[3]
		Byte Add_sel[2]	Byte Add_sel[1]	Byte Add_sel[0]	L/S[1]	L/S[0]	SP[1]	SP[0]	W
	CA9_DBG_DBGWCR2	-	-	-	address_mask[4]	address_mask[3]	address_mask[2]	address_mask[1]	address_mask[0]
		-	-	-	E	Linked BRP[3]	Linked BRP[2]	Linked BRP[1]	Linked BRP[0]
		Sec_state_cont[1]	Sec_state_cont[0]	-	-	-	-	-	Byte Add_sel[3]
		Byte Add_sel[2]	Byte Add_sel[1]	Byte Add_sel[0]	L/S[1]	L/S[0]	SP[1]	SP[0]	W
	CA9_DBG_DBGWCR3	-	-	-	address_mask[4]	address_mask[3]	address_mask[2]	address_mask[1]	address_mask[0]
		-	-	-	E	Linked BRP[3]	Linked BRP[2]	Linked BRP[1]	Linked BRP[0]
		Sec_state_cont[1]	Sec_state_cont[0]	-	-	-	-	-	Byte Add_sel[3]
		Byte Add_sel[2]	Byte Add_sel[1]	Byte Add_sel[0]	L/S[1]	L/S[0]	SP[1]	SP[0]	W
	CA9_DBG_MIDR	Implementer[7]	Implementer[6]	Implementer[5]	Implementer[4]	Implementer[3]	Implementer[2]	Implementer[1]	Implementer[0]
		Variant[3]	Variant[2]	Variant[1]	Variant[0]	Architecture[3]	Architecture[2]	Architecture[1]	Architecture[0]
		part number[11]	part number[10]	part number[9]	part number[8]	part number[7]	part number[6]	part number[5]	part number[4]
		part number[3]	part number[2]	part number[1]	part number[0]	Revision[3]	Revision[2]	Revision[1]	Revision[0]
	CA9_DBG_CTR	-	-	-	-	CWG[3]	CWG[2]	CWG[1]	CWG[0]
		ERG[3]	ERG[2]	ERG[1]	ERG[0]	DminLine[3]	DminLine[2]	DminLine[1]	DminLine[0]
		-	-	-	-	-	-	-	-
		-	-	-	-	IminLine[3]	IminLine[2]	IminLine[1]	IminLine[0]
	CA9_DBG_TLBTR	-	-	-	-	-	-	-	-
		ILSize[7]	ILSize[6]	ILSize[5]	ILSize[4]	ILSize[3]	ILSize[2]	ILSize[1]	ILSize[0]
		DLSize[7]	DLSize[6]	DLSize[5]	DLSize[4]	DLSize[3]	DLSize[2]	DLSize[1]	DLSize[0]
		-	-	-	-	-	-	TLB_size	uU
	CA9_DBG_MPIDR	-	U	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	ClusterID[3]	ClusterID[2]	ClusterID[1]	ClusterID[0]
		-	-	-	-	-	-	CPUID[1]	CPUID[0]
	CA9_DBG_REVIDR	ID[31]	ID[30]	ID[29]	ID[28]	ID[27]	ID[26]	ID[25]	ID[24]
		ID[23]	ID[22]	ID[21]	ID[20]	ID[19]	ID[18]	ID[17]	ID[16]
		ID[15]	ID[14]	ID[13]	ID[12]	ID[11]	ID[10]	ID[9]	ID[8]
		ID[7]	ID[6]	ID[5]	ID[4]	ID[3]	ID[2]	ID[1]	ID[0]
	CA9_DBG_ID_PFR0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		State3[3]	State3[2]	State3[1]	State3[0]	State2[3]	State2[2]	State2[1]	State2[0]
		State1[3]	State1[2]	State1[1]	State1[0]	State0[3]	State0[2]	State0[1]	State0[0]
	CA9_DBG_ID_PFR1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	Mprofile[3]	Mprofile[2]	Mprofile[1]	Mprofile[0]
		Security[3]	Security[2]	Security[1]	Security[0]	model[3]	model[2]	model[1]	model[0]

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Debugger interface	CA9_DBG_ID_DFR0	-	-	-	-	-	-	-	-
		Debug Model, M profile[3]	Debug Model, M profile[2]	Debug Model, M profile[1]	Debug Model, M profile[0]	Memory-mapped trace model[3]	Memory-mapped trace model[2]	Memory-mapped trace model[1]	Memory-mapped trace model[0]
		Coprocessor trace model[3]	Coprocessor trace model[2]	Coprocessor trace model[1]	Coprocessor trace model[0]	Memory-mapped debug model[3]	Memory-mapped debug model[2]	Memory-mapped debug model[1]	Memory-mapped debug model[0]
		Coprocessor Secure debug model[3]	Coprocessor Secure debug model[2]	Coprocessor Secure debug model[1]	Coprocessor Secure debug model[0]	Coprocessor debug model[3]	Coprocessor debug model[2]	Coprocessor debug model[1]	Coprocessor debug model[0]
CA9_DBG_ID_MMFR0	CA9_DBG_ID_MMFR0	Innermost shareability[3]	Innermost shareability[2]	Innermost shareability[1]	Innermost shareability[0]	FCSE support[3]	FCSE support[2]	FCSE support[1]	FCSE support[0]
		Auxiliary registers[3]	Auxiliary registers[2]	Auxiliary registers[1]	Auxiliary registers[0]	TCM support[3]	TCM support[2]	TCM support[1]	TCM support[0]
		Share ability levels[3]	Share ability levels[2]	Share ability levels[1]	Share ability levels[0]	Outermost shareability[3]	Outermost shareability[2]	Outermost shareability[1]	Outermost shareability[0]
		PMSA support[3]	PMSA support[2]	PMSA support[1]	PMSA support[0]	VMSA support[3]	VMSA support[2]	VMSA support[1]	VMSA support[0]
CA9_DBG_ID_MMFR1	CA9_DBG_ID_MMFR1	Branch Predictor[3]	Branch Predictor[2]	Branch Predictor[1]	Branch Predictor[0]	L1 cache Testand Clean[3]	L1 cache Testand Clean[2]	L1 cache Testand Clean[1]	L1 cache Testand Clean[0]
		L1 unified cache[3]	L1 unified cache[2]	L1 unified cache[1]	L1 unified cache[0]	L1 Harvard cache[3]	L1 Harvard cache[2]	L1 Harvard cache[1]	L1 Harvard cache[0]
		L1 unified cache s/w[3]	L1 unified cache s/w[2]	L1 unified cache s/w[1]	L1 unified cache s/w[0]	L1 Harvard cache s/w[3]	L1 Harvard cache s/w[2]	L1 Harvard cache s/w[1]	L1 Harvard cache s/w[0]
		L1 unified cache VA[3]	L1 unified cache VA[2]	L1 unified cache VA[1]	L1 unified cache VA[0]	L1 Harvard cache VA[3]	L1 Harvard cache VA[2]	L1 Harvard cache VA[1]	L1 Harvard cache VA[0]
CA9_DBG_ID_MMFR2	CA9_DBG_ID_MMFR2	HW access flag[3]	HW access flag[2]	HW access flag[1]	HW access flag[0]	WFI stall[3]	WFI stall[2]	WFI stall[1]	WFI stall[0]
		Mem barrier[3]	Mem barrier[2]	Mem barrier[1]	Mem barrier[0]	Unified TLB[3]	Unified TLB[2]	Unified TLB[1]	Unified TLB[0]
		Harvard TLB[3]	Harvard TLB[2]	Harvard TLB[1]	Harvard TLB[0]	L1 Harvard range[3]	L1 Harvard range[2]	L1 Harvard range[1]	L1 Harvard range[0]
		L1 Harvard bg prefetch[3]	L1 Harvard bg prefetch[2]	L1 Harvard bg prefetch[1]	L1 Harvard bg prefetch[0]	L1 Harvard fg prefetch[3]	L1 Harvard fg prefetch[2]	L1 Harvard fg prefetch[1]	L1 Harvard fg prefetch[0]
CA9_DBG_ID_MMFR3	CA9_DBG_ID_MMFR3	Supersection support[3]	Supersection support[2]	Supersection support[1]	Supersection support[0]	-	-	-	-
		Coherent walk[3]	Coherent walk[2]	Coherent walk[1]	Coherent walk[0]	-	-	-	-
		Maintenance broadcast[3]	Maintenance broadcast[2]	Maintenance broadcast[1]	Maintenance broadcast[0]	BP maintain[3]	BP maintain[2]	BP maintain[1]	BP maintain[0]
		Cache maintenance s/w[3]	Cache maintenance s/w[2]	Cache maintenance s/w[1]	Cache maintenance s/w[0]	Cache maintenance MVA[3]	Cache maintenance MVA[2]	Cache maintenance MVA[1]	Cache maintenance MVA[0]
CA9_DBG_ID_ISAR0	CA9_DBG_ID_ISAR0	-	-	-	-	Divide_insts[3]	Divide_insts[2]	Divide_insts[1]	Divide_insts[0]
		Debug_insts[3]	Debug_insts[2]	Debug_insts[1]	Debug_insts[0]	Coproc_insts[3]	Coproc_insts[2]	Coproc_insts[1]	Coproc_insts[0]
		CmpBranch_insts[3]	CmpBranch_insts[2]	CmpBranch_insts[1]	CmpBranch_insts[0]	Bitfield_insts[3]	Bitfield_insts[2]	Bitfield_insts[1]	Bitfield_insts[0]
		BitCount_insts[3]	BitCount_insts[2]	BitCount_insts[1]	BitCount_insts[0]	Swap_insts[3]	Swap_insts[2]	Swap_insts[1]	Swap_insts[0]
CA9_DBG_ID_ISAR1	CA9_DBG_ID_ISAR1	Jazelle_insts[3]	Jazelle_insts[2]	Jazelle_insts[1]	Jazelle_insts[0]	Interwork_insts[3]	Interwork_insts[2]	Interwork_insts[1]	Interwork_insts[0]
		Immediate_insts[3]	Immediate_insts[2]	Immediate_insts[1]	Immediate_insts[0]	IfThen_insts[3]	IfThen_insts[2]	IfThen_insts[1]	IfThen_insts[0]
		Extend_insts[3]	Extend_insts[2]	Extend_insts[1]	Extend_insts[0]	Except_AR_insts[3]	Except_AR_insts[2]	Except_AR_insts[1]	Except_AR_insts[0]
		Except_insts[3]	Except_insts[2]	Except_insts[1]	Except_insts[0]	Endian_insts[3]	Endian_insts[2]	Endian_insts[1]	Endian_insts[0]
CA9_DBG_ID_ISAR2	CA9_DBG_ID_ISAR2	Reversal_insts[3]	Reversal_insts[2]	Reversal_insts[1]	Reversal_insts[0]	PSR_AR_insts[3]	PSR_AR_insts[2]	PSR_AR_insts[1]	PSR_AR_insts[0]
		MultU_insts[3]	MultU_insts[2]	MultU_insts[1]	MultU_insts[0]	Mult_insts[3]	Mult_insts[2]	Mult_insts[1]	Mult_insts[0]
		Mult_insts[3]	Mult_insts[2]	Mult_insts[1]	Mult_insts[0]	MultiAccessInt_insts[3]	MultiAccessInt_insts[2]	MultiAccessInt_insts[1]	MultiAccessInt_insts[0]
		MemHint_insts[3]	MemHint_insts[2]	MemHint_insts[1]	MemHint_insts[0]	LoadStore_insts[3]	LoadStore_insts[2]	LoadStore_insts[1]	LoadStore_insts[0]
CA9_DBG_ID_ISAR3	CA9_DBG_ID_ISAR3	ThumbEE_extn_insts[3]	ThumbEE_extn_insts[2]	ThumbEE_extn_insts[1]	ThumbEE_extn_insts[0]	TrueNOP_insts[3]	TrueNOP_insts[2]	TrueNOP_insts[1]	TrueNOP_insts[0]
		ThumbCopy_insts[3]	ThumbCopy_insts[2]	ThumbCopy_insts[1]	ThumbCopy_insts[0]	TabBranch_insts[3]	TabBranch_insts[2]	TabBranch_insts[1]	TabBranch_insts[0]
		SynchPrim_insts[3]	SynchPrim_insts[2]	SynchPrim_insts[1]	SynchPrim_insts[0]	SVC_insts[3]	SVC_insts[2]	SVC_insts[1]	SVC_insts[0]
		SIMD_insts[3]	SIMD_insts[2]	SIMD_insts[1]	SIMD_insts[0]	Saturate_insts[3]	Saturate_insts[2]	Saturate_insts[1]	Saturate_insts[0]
CA9_DBG_ID_ISAR4	CA9_DBG_ID_ISAR4	SWP_frac[3]	SWP_frac[2]	SWP_frac[1]	SWP_frac[0]	PSR_M_insts[3]	PSR_M_insts[2]	PSR_M_insts[1]	PSR_M_insts[0]
		SynchPrim_insts_frac[3]	SynchPrim_insts_frac[2]	SynchPrim_insts_frac[1]	SynchPrim_insts_frac[0]	Barrier_insts[3]	Barrier_insts[2]	Barrier_insts[1]	Barrier_insts[0]
		SMC_insts[3]	SMC_insts[2]	SMC_insts[1]	SMC_insts[0]	Writeback_insts[3]	Writeback_insts[2]	Writeback_insts[1]	Writeback_insts[0]
		WithShifts_insts[3]	WithShifts_insts[2]	WithShifts_insts[1]	WithShifts_insts[0]	Unpriv_insts[3]	Unpriv_insts[2]	Unpriv_insts[1]	Unpriv_insts[0]
CA9_DBG_CLAIMSET	CA9_DBG_CLAIMSET	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		CLAIMSET[7]	CLAIMSET[6]	CLAIMSET[5]	CLAIMSET[4]	CLAIMSET[3]	CLAIMSET[2]	CLAIMSET[1]	CLAIMSET[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Debugger interface	CA9_DBG_CLAIMCLR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		CLAIMCLR[7]	CLAIMCLR[6]	CLAIMCLR[5]	CLAIMCLR[4]	CLAIMCLR[3]	CLAIMCLR[2]	CLAIMCLR[1]	CLAIMCLR[0]
	CA9_DBG_LAR	ACCESS_W[31]	ACCESS_W[30]	ACCESS_W[29]	ACCESS_W[28]	ACCESS_W[27]	ACCESS_W[26]	ACCESS_W[25]	ACCESS_W[24]
		ACCESS_W[23]	ACCESS_W[22]	ACCESS_W[21]	ACCESS_W[20]	ACCESS_W[19]	ACCESS_W[18]	ACCESS_W[17]	ACCESS_W[16]
		ACCESS_W[15]	ACCESS_W[14]	ACCESS_W[13]	ACCESS_W[12]	ACCESS_W[11]	ACCESS_W[10]	ACCESS_W[9]	ACCESS_W[8]
		ACCESS_W[7]	ACCESS_W[6]	ACCESS_W[5]	ACCESS_W[4]	ACCESS_W[3]	ACCESS_W[2]	ACCESS_W[1]	ACCESS_W[0]
	CA9_DBG_LSR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	LOCKTYPE	LOCKGRANT	LOCKEXIST
	CA9_DBG_AUTHSTATUS	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		SNID[1]	SNID[0]	SID[1]	SID[0]	NSNID[1]	NSNID[0]	NSID[1]	NSID[0]
	CA9_DBG_DEVID	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	PCsample[3]	PCsample[2]	PCsample[1]	PCsample[0]
	CA9_DBG_DEVTYPE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Sub_type[3]	Sub_type[2]	Sub_type[1]	Sub_type[0]	Main_class[3]	Main_class[2]	Main_class[1]	Main_class[0]
	CA9_DBG_PERIPHID4	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		4KB_count[3]	4KB_count[2]	4KB_count[1]	4KB_count[0]	JEP106_code[3]	JEP106_code[2]	JEP106_code[1]	JEP106_code[0]
	CA9_DBG_PERIPHID0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Part_Number[7]	Part_Number[6]	Part_Number[5]	Part_Number[4]	Part_Number[3]	Part_Number[2]	Part_Number[1]	Part_Number[0]
	CA9_DBG_PERIPHID1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		JEP106_id[3]	JEP106_id[2]	JEP106_id[1]	JEP106_id[0]	Part_Number [11]	Part_Number [10]	Part_Number[9]	Part_Number[8]
	CA9_DBG_PERIPHID2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Revision[3]	Revision[2]	Revision[1]	Revision[0]	JEDEC	JEP106_id[6]	JEP106_id[5]	JEP106_id[4]
	CA9_DBG_PERIPHID3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		RevAnd[3]	RevAnd[2]	RevAnd[1]	RevAnd[0]	CUSTOM[3]	CUSTOM[2]	CUSTOM[1]	CUSTOM[0]
	CA9_DBG_COMPID0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Preamble[7]	Preamble[6]	Preamble[5]	Preamble[4]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]
	CA9_DBG_COMPID1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Class[3]	Class[2]	Class[1]	Class[0]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]
	CA9_DBG_COMPID2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Preamble[7]	Preamble[6]	Preamble[5]	Preamble[4]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Debugger interface	CA9_DBG_COMPID3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Preamble[7]	Preamble[6]	Preamble[5]	Preamble[4]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]
	CA9_PMU_PMXEVCNTR0	PMNX[31]	PMNX[30]	PMNX[29]	PMNX[28]	PMNX[27]	PMNX[26]	PMNX[25]	PMNX[24]
		PMNX[23]	PMNX[22]	PMNX[21]	PMNX[20]	PMNX[19]	PMNX[18]	PMNX[17]	PMNX[16]
		PMNX[15]	PMNX[14]	PMNX[13]	PMNX[12]	PMNX[11]	PMNX[10]	PMNX[9]	PMNX[8]
		PMNX[7]	PMNX[6]	PMNX[5]	PMNX[4]	PMNX[3]	PMNX[2]	PMNX[1]	PMNX[0]
	CA9_PMU_PMXEVCNTR1	PMNX[31]	PMNX[30]	PMNX[29]	PMNX[28]	PMNX[27]	PMNX[26]	PMNX[25]	PMNX[24]
		PMNX[23]	PMNX[22]	PMNX[21]	PMNX[20]	PMNX[19]	PMNX[18]	PMNX[17]	PMNX[16]
		PMNX[15]	PMNX[14]	PMNX[13]	PMNX[12]	PMNX[11]	PMNX[10]	PMNX[9]	PMNX[8]
		PMNX[7]	PMNX[6]	PMNX[5]	PMNX[4]	PMNX[3]	PMNX[2]	PMNX[1]	PMNX[0]
	CA9_PMU_PMXEVCNTR2	PMNX[31]	PMNX[30]	PMNX[29]	PMNX[28]	PMNX[27]	PMNX[26]	PMNX[25]	PMNX[24]
		PMNX[23]	PMNX[22]	PMNX[21]	PMNX[20]	PMNX[19]	PMNX[18]	PMNX[17]	PMNX[16]
		PMNX[15]	PMNX[14]	PMNX[13]	PMNX[12]	PMNX[11]	PMNX[10]	PMNX[9]	PMNX[8]
		PMNX[7]	PMNX[6]	PMNX[5]	PMNX[4]	PMNX[3]	PMNX[2]	PMNX[1]	PMNX[0]
	CA9_PMU_PMXEVCNTR3	PMNX[31]	PMNX[30]	PMNX[29]	PMNX[28]	PMNX[27]	PMNX[26]	PMNX[25]	PMNX[24]
		PMNX[23]	PMNX[22]	PMNX[21]	PMNX[20]	PMNX[19]	PMNX[18]	PMNX[17]	PMNX[16]
		PMNX[15]	PMNX[14]	PMNX[13]	PMNX[12]	PMNX[11]	PMNX[10]	PMNX[9]	PMNX[8]
		PMNX[7]	PMNX[6]	PMNX[5]	PMNX[4]	PMNX[3]	PMNX[2]	PMNX[1]	PMNX[0]
	CA9_PMU_PMXEVCNTR4	PMNX[31]	PMNX[30]	PMNX[29]	PMNX[28]	PMNX[27]	PMNX[26]	PMNX[25]	PMNX[24]
		PMNX[23]	PMNX[22]	PMNX[21]	PMNX[20]	PMNX[19]	PMNX[18]	PMNX[17]	PMNX[16]
		PMNX[15]	PMNX[14]	PMNX[13]	PMNX[12]	PMNX[11]	PMNX[10]	PMNX[9]	PMNX[8]
		PMNX[7]	PMNX[6]	PMNX[5]	PMNX[4]	PMNX[3]	PMNX[2]	PMNX[1]	PMNX[0]
	CA9_PMU_PMXEVCNTR5	PMNX[31]	PMNX[30]	PMNX[29]	PMNX[28]	PMNX[27]	PMNX[26]	PMNX[25]	PMNX[24]
		PMNX[23]	PMNX[22]	PMNX[21]	PMNX[20]	PMNX[19]	PMNX[18]	PMNX[17]	PMNX[16]
		PMNX[15]	PMNX[14]	PMNX[13]	PMNX[12]	PMNX[11]	PMNX[10]	PMNX[9]	PMNX[8]
		PMNX[7]	PMNX[6]	PMNX[5]	PMNX[4]	PMNX[3]	PMNX[2]	PMNX[1]	PMNX[0]
	CA9_PMU_PMXCCNTR	CCNT[31]	CCNT[30]	CCNT[29]	CCNT[28]	CCNT[27]	CCNT[26]	CCNT[25]	CCNT[24]
		CCNT[23]	CCNT[22]	CCNT[21]	CCNT[20]	CCNT[19]	CCNT[18]	CCNT[17]	CCNT[16]
		CCNT[15]	CCNT[14]	CCNT[13]	CCNT[12]	CCNT[11]	CCNT[10]	CCNT[9]	CCNT[8]
		CCNT[7]	CCNT[6]	CCNT[5]	CCNT[4]	CCNT[3]	CCNT[2]	CCNT[1]	CCNT[0]
	CA9_PMU_PMXEVTYPER0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	SEL[4]	SEL[3]	SEL[2]	SEL[1]	SEL[0]
	CA9_PMU_PMXEVTYPER1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	SEL[4]	SEL[3]	SEL[2]	SEL[1]	SEL[0]
	CA9_PMU_PMXEVTYPER2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	SEL[4]	SEL[3]	SEL[2]	SEL[1]	SEL[0]
	CA9_PMU_PMXEVTYPER3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	SEL[4]	SEL[3]	SEL[2]	SEL[1]	SEL[0]
	CA9_PMU_PMXEVTYPER4	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	SEL[4]	SEL[3]	SEL[2]	SEL[1]	SEL[0]
	CA9_PMU_PMXEVTYPER5	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	SEL[4]	SEL[3]	SEL[2]	SEL[1]	SEL[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Debugger interface	CA9_PMU_PMCNTENSET	C	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	-
		-	-	P5	P4	P3	P2	P1	P0	
	CA9_PMU_PMCNTENCLR	C	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	P5	P4	P3	P2	P1	P0	
	CA9_PMU_PMINTENSET	C	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	P5	P4	P3	P2	P1	P0	
	CA9_PMU_PMINTENCLR	C	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	P5	P4	P3	P2	P1	P0	
	CA9_PMU_PMINTENCLR	C	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	P5	P4	P3	P2	P1	P0	
	CA9_PMU_PMOVSR	C	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	P5	P4	P3	P2	P1	P0	
	CA9_PMU_PMSWINC	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	P5	P4	P3	P2	P1	P0	
	CA9_PMU_PMCR	IMP[7]	IMP[6]	IMP[5]	IMP[4]	IMP[3]	IMP[2]	IMP[1]	IMP[0]	
		IDCODE[7]	IDCODE[6]	IDCODE[5]	IDCODE[4]	IDCODE[3]	IDCODE[2]	IDCODE[1]	IDCODE[0]	
		N[4]	N[3]	N[2]	N[1]	N[0]	-	-	-	
		-	-	DP	X	D	C	P	E	
	CA9_PMU_CLAIMSET	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		CLAIMSET[7]	CLAIMSET[6]	CLAIMSET[5]	CLAIMSET[4]	CLAIMSET[3]	CLAIMSET[2]	CLAIMSET[1]	CLAIMSET[0]	
	CA9_PMU_CLAIMCLR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		CLAIMCLR[7]	CLAIMCLR[6]	CLAIMCLR[5]	CLAIMCLR[4]	CLAIMCLR[3]	CLAIMCLR[2]	CLAIMCLR[1]	CLAIMCLR[0]	
	CA9_PMU_LAR	ACCESS_W[31]	ACCESS_W[30]	ACCESS_W[29]	ACCESS_W[28]	ACCESS_W[27]	ACCESS_W[26]	ACCESS_W[25]	ACCESS_W[24]	
		ACCESS_W[23]	ACCESS_W[22]	ACCESS_W[21]	ACCESS_W[20]	ACCESS_W[19]	ACCESS_W[18]	ACCESS_W[17]	ACCESS_W[16]	
		ACCESS_W[15]	ACCESS_W[14]	ACCESS_W[13]	ACCESS_W[12]	ACCESS_W[11]	ACCESS_W[10]	ACCESS_W[9]	ACCESS_W[8]	
		ACCESS_W[7]	ACCESS_W[6]	ACCESS_W[5]	ACCESS_W[4]	ACCESS_W[3]	ACCESS_W[2]	ACCESS_W[1]	ACCESS_W[0]	
	CA9_PMU_LSR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	LOCKTYPE	LOCKGRANT	LOCKEXIST	
	CA9_PMU_AUTHSTATUS	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		SNID[1]	SNID[0]	SID[1]	SID[0]	NSNID[1]	NSNID[0]	NSID[1]	NSID[0]	
	CA9_PMU_DEVID	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		ID[7]	ID[6]	ID[5]	ID[4]	ID[3]	ID[2]	ID[1]	ID[0]	



Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Debugger interface	CA9_PMU_DEVTYPE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Sub_type[3]	Sub_type[2]	Sub_type[1]	Sub_type[0]	Main_class[3]	Main_class[2]	Main_class[1]	Main_class[0]
	CA9_PMU_PERIPHID4	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		4KB_count[3]	4KB_count[2]	4KB_count[1]	4KB_count[0]	JEP106_code[3]	JEP106_code[2]	JEP106_code[1]	JEP106_code[0]
	CA9_PMU_PERIPHID0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Part_Number[7]	Part_Number[6]	Part_Number[5]	Part_Number[4]	Part_Number[3]	Part_Number[2]	Part_Number[1]	Part_Number[0]
	CA9_PMU_PERIPHID1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		JEP106_id[3]	JEP106_id[2]	JEP106_id[1]	JEP106_id[0]	Part_Number[11]	Part_Number[10]	Part_Number[9]	Part_Number[8]
	CA9_PMU_PERIPHID2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Revision[3]	Revision[2]	Revision[1]	Revision[0]	JEDEC	JEP106_id[6]	JEP106_id[5]	JEP106_id[4]
	CA9_PMU_PERIPHID3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		RevAnd[3]	RevAnd[2]	RevAnd[1]	RevAnd[0]	CUSTOM[3]	CUSTOM[2]	CUSTOM[1]	CUSTOM[0]
	CA9_PMU_COMPID0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Preamble[7]	Preamble[6]	Preamble[5]	Preamble[4]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]
	CA9_PMU_COMPID1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Class[3]	Class[2]	Class[1]	Class[0]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]
	CA9_PMU_COMPID2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Preamble[7]	Preamble[6]	Preamble[5]	Preamble[4]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]
	CA9_PMU_COMPID3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Preamble[7]	Preamble[6]	Preamble[5]	Preamble[4]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]
	CA9_CTL_CTICONTROL	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	GLBEN
	CA9_CTL_CTIINTACK	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		INTACK[7]	INTACK[6]	INTACK[5]	INTACK[4]	INTACK[3]	INTACK[2]	INTACK[1]	INTACK[0]
	CA9_CTL_CTIAPPSET	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	APPSET[3]	APPSET[2]	APPSET[1]	APPSET[0]
	CA9_CTL_CTIAPPCLEAR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	APPCLEAR[3]	APPCLEAR[2]	APPCLEAR[1]	APPCLEAR[0]

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Debugger interface	CA9_CTL_CTIAPPULSE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	APPULSE[3]	APPULSE[2]	APPULSE[1]	APPULSE[0]
	CA9_CTL_CTIINEN0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	TRIGINEN[3]	TRIGINEN[2]	TRIGINEN[1]	TRIGINEN[0]
	CA9_CTL_CTIINEN1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	TRIGINEN[3]	TRIGINEN[2]	TRIGINEN[1]	TRIGINEN[0]
	CA9_CTL_CTIINEN2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	TRIGINEN[3]	TRIGINEN[2]	TRIGINEN[1]	TRIGINEN[0]
	CA9_CTL_CTIINEN3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	TRIGINEN[3]	TRIGINEN[2]	TRIGINEN[1]	TRIGINEN[0]
	CA9_CTL_CTIINEN4	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	TRIGINEN[3]	TRIGINEN[2]	TRIGINEN[1]	TRIGINEN[0]
	CA9_CTL_CTIINEN5	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	TRIGINEN[3]	TRIGINEN[2]	TRIGINEN[1]	TRIGINEN[0]
	CA9_CTL_CTIINEN6	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	TRIGINEN[3]	TRIGINEN[2]	TRIGINEN[1]	TRIGINEN[0]
	CA9_CTL_CTIINEN7	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	TRIGINEN[3]	TRIGINEN[2]	TRIGINEN[1]	TRIGINEN[0]
	CA9_CTL_CTIOUTEN0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	TRIGOUTEN[3]	TRIGOUTEN[2]	TRIGOUTEN[1]	TRIGOUTEN[0]
	CA9_CTL_CTIOUTEN1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	TRIGOUTEN[3]	TRIGOUTEN[2]	TRIGOUTEN[1]	TRIGOUTEN[0]
	CA9_CTL_CTIOUTEN2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	TRIGOUTEN[3]	TRIGOUTEN[2]	TRIGOUTEN[1]	TRIGOUTEN[0]
	CA9_CTL_CTIOUTEN3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	TRIGOUTEN[3]	TRIGOUTEN[2]	TRIGOUTEN[1]	TRIGOUTEN[0]
	CA9_CTL_CTIOUTEN4	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	TRIGOUTEN[3]	TRIGOUTEN[2]	TRIGOUTEN[1]	TRIGOUTEN[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Debugger interface	CA9_CTL_CTIOUTEN5	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	TRIGOUTEN[3]	TRIGOUTEN[2]	TRIGOUTEN[1]	TRIGOUTEN[0]
	CA9_CTL_CTIOUTEN6	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	TRIGOUTEN[3]	TRIGOUTEN[2]	TRIGOUTEN[1]	TRIGOUTEN[0]
	CA9_CTL_CTIOUTEN7	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	TRIGOUTEN[3]	TRIGOUTEN[2]	TRIGOUTEN[1]	TRIGOUTEN[0]
	CA9_CTL_CTIIRIGINSTATUS	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		TRIGINSTATUS[7]	TRIGINSTATUS[6]	TRIGINSTATUS[5]	TRIGINSTATUS[4]	TRIGINSTATUS[3]	TRIGINSTATUS[2]	TRIGINSTATUS[1]	TRIGINSTATUS[0]
	CA9_CTL_CTIIRIGOUTSTATUS	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		TRIGOUTSTATUS[7]	TRIGOUTSTATUS[6]	TRIGOUTSTATUS[5]	TRIGOUTSTATUS[4]	TRIGOUTSTATUS[3]	TRIGOUTSTATUS[2]	TRIGOUTSTATUS[1]	TRIGOUTSTATUS[0]
	CA9_CTL_CTICTCHINSTATUS	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	CTCHINSTATUS[3]	CTCHINSTATUS[2]	CTCHINSTATUS[1]	CTCHINSTATUS[0]
	CA9_CTL_CTICTCHOUTSTATUS	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	CTCHOUTSTATUS[3]	CTCHOUTSTATUS[2]	CTCHOUTSTATUS[1]	CTCHOUTSTATUS[0]
	CA9_CTL_CTIIGATE	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	CTIGATEEN3	CTIGATEEN2	CTIGATEEN1	CTIGATEEN0
	CA9_CTL_ASICCTL	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	ASICCTL[3]	ASICCTL[2]	ASICCTL[1]	ASICCTL[0]
	CA9_CTL_CLAIMSET	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	CLAIMSET[3]	CLAIMSET[2]	CLAIMSET[1]	CLAIMSET[0]
	CA9_CTL_CLAIMCLR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	CLAIMCLR[3]	CLAIMCLR[2]	CLAIMCLR[1]	CLAIMCLR[0]
	CA9_CTL_LAR	ACCESS_W[31]	ACCESS_W[30]	ACCESS_W[29]	ACCESS_W[28]	ACCESS_W[27]	ACCESS_W[26]	ACCESS_W[25]	ACCESS_W[24]
		ACCESS_W[23]	ACCESS_W[22]	ACCESS_W[21]	ACCESS_W[20]	ACCESS_W[19]	ACCESS_W[18]	ACCESS_W[17]	ACCESS_W[16]
		ACCESS_W[15]	ACCESS_W[14]	ACCESS_W[13]	ACCESS_W[12]	ACCESS_W[11]	ACCESS_W[10]	ACCESS_W[9]	ACCESS_W[8]
		ACCESS_W[7]	ACCESS_W[6]	ACCESS_W[5]	ACCESS_W[4]	ACCESS_W[3]	ACCESS_W[2]	ACCESS_W[1]	ACCESS_W[0]
	CA9_CTL_LSR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	LOCKTYPE	LOCKGRANT	LOCKEXIST

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Debugger interface	CA9_CTL_AUTHSTATUS	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		SNID[1]	SNID[0]	SID[1]	SID[0]	NSNID[1]	NSNID[0]	NSID[1]	NSID[0]
	CA9_CTL_DEVID	-	-	-	-	-	-	-	-
		-	-	-	-	CHANWIDTH[3]	CHANWIDTH[2]	CHANWIDTH[1]	CHANWIDTH[0]
		TRIGWIDTH[7]	TRIGWIDTH[6]	TRIGWIDTH[5]	TRIGWIDTH[4]	TRIGWIDTH[3]	TRIGWIDTH[2]	TRIGWIDTH[1]	TRIGWIDTH[0]
	CA9_CTL_DEVTYPE	-	-	-	-	EXTMUXNUM[3]	EXTMUXNUM[2]	EXTMUXNUM[1]	EXTMUXNUM[0]
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Sub_type[3]	Sub_type[2]	Sub_type[1]	Sub_type[0]	Major_type[3]	Major_type[2]	Major_type[1]	Major_type[0]
	CA9_CTL_PERIPHID4	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		4KB_count[3]	4KB_count[2]	4KB_count[1]	4KB_count[0]	JEP106_code[3]	JEP106_code[2]	JEP106_code[1]	JEP106_code[0]
	CA9_CTL_PERIPHID0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Part_Number[7]	Part_Number[6]	Part_Number[5]	Part_Number[4]	Part_Number[3]	Part_Number[2]	Part_Number[1]	Part_Number[0]
	CA9_CTL_PERIPHID1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		JEP106_id[3]	JEP106_id[2]	JEP106_id[1]	JEP106_id[0]	Part_Number [11]	Part_Number [10]	Part_Number[9]	Part_Number[8]
	CA9_CTL_PERIPHID2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Revision[3]	Revision[2]	Revision[1]	Revision[0]	JEDEC	JEP106_id[6]	JEP106_id[5]	JEP106_id[4]
	CA9_CTL_PERIPHID3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		RevAnd[3]	RevAnd[2]	RevAnd[1]	RevAnd[0]	CUSTOM[3]	CUSTOM[2]	CUSTOM[1]	CUSTOM[0]
	CA9_CTL_COMPID0	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Preamble[7]	Preamble[6]	Preamble[5]	Preamble[4]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]
	CA9_CTL_COMPID1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Class[3]	Class[2]	Class[1]	Class[0]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]
	CA9_CTL_COMPID2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Preamble[7]	Preamble[6]	Preamble[5]	Preamble[4]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]
	CA9_CTL_COMPID3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Preamble[7]	Preamble[6]	Preamble[5]	Preamble[4]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]
	PTM_A9_ETMCR	-	-	Return stack enable	Timestamp enable	Processor select[2]	Processor select [1]	Processor select [0]	-
		-	-	-	-	-	-	-	-
		Context IDsize[1]	Context IDsize[0]	-	CycleAccurate	-	ProgBit	Debug request control	Branch Broadcast
		Stall processor	-	-	-	-	-	-	PowerDown

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Debugger interface	PTM_A9_ETMCCR	ID Register present	-	-	-	Software access support	Trace stop/start block present	Number of Context ID comparators[1]	Number of Context ID comparators[0]
		FIFOFULL logic	Number of external outputs[2]	Number of external outputs[1]	Number of external outputs[0]	Number of external inputs[2]	Number of external inputs[1]	Number of external inputs[0]	Sequencer
		Numberofcounters[2]	Numberofcounters[1]	Numberofcounters[0]	-	-	-	-	-
		-	-	-	-	Number of pairs of address comparators[3]	Number of pairs of address comparators[2]	Number of pairs of address comparators[1]	Number of pairs of address comparators[0]
	PTM_A9_ETMTRIGGER	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Trigger event[15]	Trigger Event[14]	Trigger Event[13]	Trigger Event[12]	Trigger Event[11]	Trigger Event[10]	Trigger Event[9]	Trigger Event[8]
		Trigger Event[7]	Trigger Event[6]	Trigger Event[5]	Trigger Event[4]	Trigger Event[3]	Trigger Event[2]	Trigger Event[1]	Trigger Event[0]
	PTM_A9_ETMSR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	Trigger flag	Trace start/stop resource status	ProgBit value	Untraced overflow flag
	PTM_A9_ETMSCR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	MAXCORES[2]	MAXCORES[1]	MAXCORES[0]	-	-	-	FIFOFULL supported
		-	-	-	-	-	-	-	-
	PTM_A9_ETMTSSCR	-	-	-	-	-	-	-	-
		Stop addresses[7]	Stop addresses[6]	Stop addresses[5]	Stop addresses[4]	Stop addresses[3]	Stop addresses[2]	Stop addresses[1]	Stop addresses[0]
		-	-	-	-	-	-	-	-
		Start addresses[7]	Start addresses[6]	Start addresses[5]	Start addresses[4]	Start addresses[3]	Start addresses[2]	Start addresses[1]	Start addresses[0]
	PTM_A9_ETMTEEVR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		Trace Enable event[15]	Trace Enable event[14]	Trace Enable event[13]	Trace Enable event[12]	Trace Enable event[11]	Trace Enable event[10]	Trace Enable event[9]	Trace Enable event[8]
		Trace Enable event[7]	Trace Enable event[6]	Trace Enable event[5]	Trace Enable event[4]	Trace Enable event[3]	Trace Enable event[2]	Trace Enable event[1]	Trace Enable event[0]
	PTM_A9_ETMTECR1	-	-	-	-	-	-	-	Trace control enable
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	Addresscom parators[3]	Addresscom parators[2]	Addresscom parators[1]	Addresscom parators[0]
	PTM_A9_ETMACVR1	Address[31]	Address[30]	Address[29]	Address[28]	Address[27]	Address[26]	Address[25]	Address[24]
		Address[23]	Address[22]	Address[21]	Address[20]	Address[19]	Address[18]	Address[17]	Address[16]
		Address[15]	Address[14]	Address[13]	Address[12]	Address[11]	Address[10]	Address[9]	Address[8]
		Address[7]	Address[6]	Address[5]	Address[4]	Address[3]	Address[2]	Address[1]	Address[0]
	PTM_A9_ETMACVR2	Address[31]	Address[30]	Address[29]	Address[28]	Address[27]	Address[26]	Address[25]	Address[24]
		Address[23]	Address[22]	Address[21]	Address[20]	Address[19]	Address[18]	Address[17]	Address[16]
		Address[15]	Address[14]	Address[13]	Address[12]	Address[11]	Address[10]	Address[9]	Address[8]
		Address[7]	Address[6]	Address[5]	Address[4]	Address[3]	Address[2]	Address[1]	Address[0]
	PTM_A9_ETMACVR3	Address[31]	Address[30]	Address[29]	Address[28]	Address[27]	Address[26]	Address[25]	Address[24]
		Address[23]	Address[22]	Address[21]	Address[20]	Address[19]	Address[18]	Address[17]	Address[16]
		Address[15]	Address[14]	Address[13]	Address[12]	Address[11]	Address[10]	Address[9]	Address[8]
		Address[7]	Address[6]	Address[5]	Address[4]	Address[3]	Address[2]	Address[1]	Address[0]
	PTM_A9_ETMACVR4	Address[31]	Address[30]	Address[29]	Address[28]	Address[27]	Address[26]	Address[25]	Address[24]
		Address[23]	Address[22]	Address[21]	Address[20]	Address[19]	Address[18]	Address[17]	Address[16]
		Address[15]	Address[14]	Address[13]	Address[12]	Address[11]	Address[10]	Address[9]	Address[8]
		Address[7]	Address[6]	Address[5]	Address[4]	Address[3]	Address[2]	Address[1]	Address[0]
	PTM_A9_ETMACVR5	Address[31]	Address[30]	Address[29]	Address[28]	Address[27]	Address[26]	Address[25]	Address[24]
		Address[23]	Address[22]	Address[21]	Address[20]	Address[19]	Address[18]	Address[17]	Address[16]
		Address[15]	Address[14]	Address[13]	Address[12]	Address[11]	Address[10]	Address[9]	Address[8]
		Address[7]	Address[6]	Address[5]	Address[4]	Address[3]	Address[2]	Address[1]	Address[0]
	PTM_A9_ETMACVR6	Address[31]	Address[30]	Address[29]	Address[28]	Address[27]	Address[26]	Address[25]	Address[24]
		Address[23]	Address[22]	Address[21]	Address[20]	Address[19]	Address[18]	Address[17]	Address[16]
		Address[15]	Address[14]	Address[13]	Address[12]	Address[11]	Address[10]	Address[9]	Address[8]
		Address[7]	Address[6]	Address[5]	Address[4]	Address[3]	Address[2]	Address[1]	Address[0]

**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Debugger interface	PTM_A9_ETMACVR7	Address[31]	Address[30]	Address[29]	Address[28]	Address[27]	Address[26]	Address[25]	Address[24]
		Address[23]	Address[22]	Address[21]	Address[20]	Address[19]	Address[18]	Address[17]	Address[16]
		Address[15]	Address[14]	Address[13]	Address[12]	Address[11]	Address[10]	Address[9]	Address[8]
		Address[7]	Address[6]	Address[5]	Address[4]	Address[3]	Address[2]	Address[1]	Address[0]
	PTM_A9_ETMACVR8	Address[31]	Address[30]	Address[29]	Address[28]	Address[27]	Address[26]	Address[25]	Address[24]
		Address[23]	Address[22]	Address[21]	Address[20]	Address[19]	Address[18]	Address[17]	Address[16]
		Address[15]	Address[14]	Address[13]	Address[12]	Address[11]	Address[10]	Address[9]	Address[8]
		Address[7]	Address[6]	Address[5]	Address[4]	Address[3]	Address[2]	Address[1]	Address[0]
	PTM_A9_ETMACR1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	Secure Level[1]	Secure Level[0]	Context ID[1]	Context ID[0]
		-	-	-	-	-	Access Type[2]	Access Type[1]	Access Type[0]
	PTM_A9_ETMACR2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	Secure Level[1]	Secure Level[0]	Context ID[1]	Context ID[0]
		-	-	-	-	-	Access Type[2]	Access Type[1]	Access Type[0]
	PTM_A9_ETMACR3	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	Secure Level[1]	Secure Level[0]	Context ID[1]	Context ID[0]
		-	-	-	-	-	Access Type[2]	Access Type[1]	Access Type[0]
	PTM_A9_ETMACR4	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	Secure Level[1]	Secure Level[0]	Context ID[1]	Context ID[0]
		-	-	-	-	-	Access Type[2]	Access Type[1]	Access Type[0]
	PTM_A9_ETMACR5	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	Secure Level[1]	Secure Level[0]	Context ID[1]	Context ID[0]
		-	-	-	-	-	Access Type[2]	Access Type[1]	Access Type[0]
	PTM_A9_ETMACR6	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	Secure Level[1]	Secure Level[0]	Context ID[1]	Context ID[0]
		-	-	-	-	-	Access Type[2]	Access Type[1]	Access Type[0]
	PTM_A9_ETMACR7	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	Secure Level[1]	Secure Level[0]	Context ID[1]	Context ID[0]
		-	-	-	-	-	Access Type[2]	Access Type[1]	Access Type[0]
	PTM_A9_ETMACR8	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	Secure Level[1]	Secure Level[0]	Context ID[1]	Context ID[0]
		-	-	-	-	-	Access Type[2]	Access Type[1]	Access Type[0]
	PTM_A9_ETMCNTRLDVR1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		value[15]	value[14]	value[13]	value[12]	value[11]	value[10]	value[9]	value[8]
		value[7]	value[6]	value[5]	value[4]	value[3]	value[2]	value[1]	value[0]
	PTM_A9_ETMCNTRLDVR2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		value[15]	value[14]	value[13]	value[12]	value[11]	value[10]	value[9]	value[8]
		value[7]	value[6]	value[5]	value[4]	value[3]	value[2]	value[1]	value[0]
	PTM_A9_ETMCNTENR1	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	event[16]
		event[15]	event[14]	event[13]	event[12]	event[11]	event[10]	event[9]	event[8]
		event[7]	event[6]	event[5]	event[4]	event[3]	event[2]	event[1]	event[0]
	PTM_A9_ETMCNTENR2	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	event[16]
		event[15]	event[14]	event[13]	event[12]	event[11]	event[10]	event[9]	event[8]
		event[7]	event[6]	event[5]	event[4]	event[3]	event[2]	event[1]	event[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0
Debugger interface	PTM_A9_ETMCNTRLDEVR1	-	-	-	-	-	-	-	-
		event[15]	event[14]	event[13]	event[12]	event[11]	event[10]	event[9]	event[8]
		event[7]	event[6]	event[5]	event[4]	event[3]	event[2]	event[1]	event[0]
	PTM_A9_ETMCNTRLDEVR2	-	-	-	-	-	-	-	-
		event[15]	event[14]	event[13]	event[12]	event[11]	event[10]	event[9]	event[8]
		event[7]	event[6]	event[5]	event[4]	event[3]	event[2]	event[1]	event[0]
	PTM_A9_ETMCNTRV1	-	-	-	-	-	-	-	-
		value[15]	value[14]	value[13]	value[12]	value[11]	value[10]	value[9]	value[8]
		value[7]	value[6]	value[5]	value[4]	value[3]	value[2]	value[1]	value[0]
	PTM_A9_ETMCNTRV2	-	-	-	-	-	-	-	-
		value[15]	value[14]	value[13]	value[12]	value[11]	value[10]	value[9]	value[8]
		value[7]	value[6]	value[5]	value[4]	value[3]	value[2]	value[1]	value[0]
	PTM_A9_ETMSQ12EVR	-	-	-	-	-	-	-	-
		event[15]	event[14]	event[13]	event[12]	event[11]	event[10]	event[9]	event[8]
		event[7]	event[6]	event[5]	event[4]	event[3]	event[2]	event[1]	event[0]
	PTM_A9_ETMSQ21EVR	-	-	-	-	-	-	-	-
		event[15]	event[14]	event[13]	event[12]	event[11]	event[10]	event[9]	event[8]
		event[7]	event[6]	event[5]	event[4]	event[3]	event[2]	event[1]	event[0]
	PTM_A9_ETMSQ23EVR	-	-	-	-	-	-	-	-
		event[15]	event[14]	event[13]	event[12]	event[11]	event[10]	event[9]	event[8]
		event[7]	event[6]	event[5]	event[4]	event[3]	event[2]	event[1]	event[0]
	PTM_A9_ETMSQ31EVR	-	-	-	-	-	-	-	-
		event[15]	event[14]	event[13]	event[12]	event[11]	event[10]	event[9]	event[8]
		event[7]	event[6]	event[5]	event[4]	event[3]	event[2]	event[1]	event[0]
	PTM_A9_ETMSQ32EVR	-	-	-	-	-	-	-	-
		event[15]	event[14]	event[13]	event[12]	event[11]	event[10]	event[9]	event[8]
		event[7]	event[6]	event[5]	event[4]	event[3]	event[2]	event[1]	event[0]
	PTM_A9_ETMSQ13EVR	-	-	-	-	-	-	-	-
		event[15]	event[14]	event[13]	event[12]	event[11]	event[10]	event[9]	event[8]
		event[7]	event[6]	event[5]	event[4]	event[3]	event[2]	event[1]	event[0]
	PTM_A9_ETMSQR	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-
		-	-	-	-	-	-	state[1]	state[0]
	PTM_A9_ETMEXTOUTEVR1	-	-	-	-	-	-	-	-
		event[15]	event[14]	event[13]	event[12]	event[11]	event[10]	event[9]	event[8]
		event[7]	event[6]	event[5]	event[4]	event[3]	event[2]	event[1]	event[0]
	PTM_A9_ETMEXTOUTEVR2	-	-	-	-	-	-	-	-
		event[15]	event[14]	event[13]	event[12]	event[11]	event[10]	event[9]	event[8]
		event[7]	event[6]	event[5]	event[4]	event[3]	event[2]	event[1]	event[0]
	PTM_A9_ETMCDIVR1	value[31]	value[30]	value[29]	value[28]	value[27]	value[26]	value[25]	value[24]
		value[23]	value[22]	value[21]	value[20]	value[19]	value[18]	value[17]	value[16]
		value[15]	value[14]	value[13]	value[12]	value[11]	value[10]	value[9]	value[8]
		value[7]	value[6]	value[5]	value[4]	value[3]	value[2]	value[1]	value[0]

Table 57.2 Register Bits

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Debugger interface	PTM_A9_ETMCIIDCMR	mask[31]	mask[30]	mask[29]	mask[28]	mask[27]	mask[26]	mask[25]	mask[24]	
		mask[23]	mask[22]	mask[21]	mask[20]	mask[19]	mask[18]	mask[17]	mask[16]	
		mask[15]	mask[14]	mask[13]	mask[12]	mask[11]	mask[10]	mask[9]	mask[8]	
		mask[7]	mask[6]	mask[5]	mask[4]	mask[3]	mask[2]	mask[1]	mask[0]	
	PTM_A9_ETMSYNCFR	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	frequency[11]	frequency[10]	frequency[9]	frequency[8]	-
		frequency[7]	frequency[6]	frequency[5]	frequency[4]	frequency[3]	frequency[2]	frequency[1]	frequency[0]	-
	PTM_A9_ETMIDR	code[7]	code[6]	code[5]	code[4]	code[3]	code[2]	code[1]	code[0]	-
		-	-	-	-	Security	32-bitThumb	-	-	-
		-	-	-	-	Major[7]	Major[6]	Major[5]	Major[4]	-
		Minor[3]	Minor[2]	Minor[1]	Minor[0]	revision[3]	revision[2]	revision[1]	revision[0]	-
	PTM_A9_ETMCCER	-	-	-	-	-	-	-	Timestamps for DMB/DSB	DMB/DSB treated as waypoint
		Return stack implemented	Timestamping implemented	-	-	-	-	-	-	-
		Number of instrumentation resources[2]	Number of instrumentation resources[1]	Number of instrumentation resources[0]	-	-	Extended external input bus size[7]	Extended external input bus size[6]	Extended external input bus size[5]	-
		Extended external input bus size[4]	Extended external input bus size[3]	Extended external input bus size[2]	Extended external input bus size[1]	Extended external input bus size[0]	Number of extended external input selectors[2]	Number of extended external input selectors[1]	Number of extended external input selectors[0]	-
	PTM_A9_ETMEXTINSELR	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	Second extended external input selector[5]	Second extended external input selector[4]	Second extended external input selector[3]	Second extended external input selector[2]	Second extended external input selector[1]	Second extended external input selector[0]	-
		-	-	First extended external input selector[5]	First extended external input selector[4]	First extended external input selector[3]	First extended external input selector[2]	First extended external input selector[1]	First extended external input selector[0]	-
	PTM_A9_ETMTSEVR	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	event[16]
		event[15]	event[14]	event[13]	event[12]	event[11]	event[10]	event[9]	event[8]	-
		event[7]	event[6]	event[5]	event[4]	event[3]	event[2]	event[1]	event[0]	-
	PTM_A9_ETMAUXCR	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	Force synchronization packet insertion	Disable waypoint update packet	Disable timestamps on barriers	Disable forced overflow	-
	PTM_A9_ETMTRACEIDR	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		ID[7]	ID[6]	ID[5]	ID[4]	ID[3]	ID[2]	ID[1]	ID[0]	-
	PTM_A9_OSLSR	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	Lock access mechanism indicator	32-bit access required	PTM trace registers locked/unlocked	Lock access mechanism indicator	-
	PTM_A9_CLAIMSET	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		CLAIMSET[7]	CLAIMSET[6]	CLAIMSET[5]	CLAIMSET[4]	CLAIMSET[3]	CLAIMSET[2]	CLAIMSET[1]	CLAIMSET[0]	-
	PTM_A9_CLAIMCLR	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		CLAIMCLR[7]	CLAIMCLR[6]	CLAIMCLR[5]	CLAIMCLR[4]	CLAIMCLR[3]	CLAIMCLR[2]	CLAIMCLR[1]	CLAIMCLR[0]	-
	PTM_A9_LAR	ACCESS_W[31]	ACCESS_W[30]	ACCESS_W[29]	ACCESS_W[28]	ACCESS_W[27]	ACCESS_W[26]	ACCESS_W[25]	ACCESS_W[24]	-
		ACCESS_W[23]	ACCESS_W[22]	ACCESS_W[21]	ACCESS_W[20]	ACCESS_W[19]	ACCESS_W[18]	ACCESS_W[17]	ACCESS_W[16]	-
		ACCESS_W[15]	ACCESS_W[14]	ACCESS_W[13]	ACCESS_W[12]	ACCESS_W[11]	ACCESS_W[10]	ACCESS_W[9]	ACCESS_W[8]	-
		ACCESS_W[7]	ACCESS_W[6]	ACCESS_W[5]	ACCESS_W[4]	ACCESS_W[3]	ACCESS_W[2]	ACCESS_W[1]	ACCESS_W[0]	-



**Table 57.2 Register Bits**

Module	Register Abbreviation	Bits 31/23/15/7	Bits 30/22/14/6	Bits 29/21/13/5	Bits 28/20/12/4	Bits 27/19/11/3	Bits 26/18/10/2	Bits 25/17/9/1	Bits 24/16/8/0	
Debugger interface	PTM_A9_LSR	-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	-	-	
		-	-	-	-	-	-	LOCKTYPE	LOCKGRANT	LOCKEXIST
	PTM_A9_AUTHSTATUS	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		SNID[1]	SNID[0]	SID[1]	SID[0]	NSNID[1]	NSNID[0]	NSID[1]	NSID[0]	
	PTM_A9_DEVTYPE	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		Sub_type[3]	Sub_type[2]	Sub_type[1]	Sub_type[0]	Main_type[3]	Main_type[2]	Main_type[1]	Main_type[0]	
	PTM_A9_PERIPHID4	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		4KB_count[3]	4KB_count[2]	4KB_count[1]	4KB_count[0]	JEP106_code[3]	JEP106_code[2]	JEP106_code[1]	JEP106_code[0]	
	PTM_A9_PERIPHID0	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		Part_Number[7]	Part_Number[6]	Part_Number[5]	Part_Number[4]	Part_Number[3]	Part_Number[2]	Part_Number[1]	Part_Number[0]	
	PTM_A9_PERIPHID1	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		JEP106_id[3]	JEP106_id[2]	JEP106_id[1]	JEP106_id[0]	Part_Number[11]	Part_Number[10]	Part_Number[9]	Part_Number[8]	
	PTM_A9_PERIPHID2	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		Revision[3]	Revision[2]	Revision[1]	Revision[0]	JEDEC	JEP106_id[6]	JEP106_id[5]	JEP106_id[4]	
	PTM_A9_PERIPHID3	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		RevAnd[3]	RevAnd[2]	RevAnd[1]	RevAnd[0]	CUSTOM[3]	CUSTOM[2]	CUSTOM[1]	CUSTOM[0]	
	PTM_A9_COMPID0	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		Preamble[7]	Preamble[6]	Preamble[5]	Preamble[4]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]	
	PTM_A9_COMPID1	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		Class[3]	Class[2]	Class[1]	Class[0]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]	
	PTM_A9_COMPID2	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		Preamble[7]	Preamble[6]	Preamble[5]	Preamble[4]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]	
	PTM_A9_COMPID3	-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		-	-	-	-	-	-	-	-	-
		Preamble[7]	Preamble[6]	Preamble[5]	Preamble[4]	Preamble[3]	Preamble[2]	Preamble[1]	Preamble[0]	
	BSID	BSID[31]	BSID[30]	BSID[29]	BSID[28]	BSID[27]	BSID[26]	BSID[25]	BSID[24]	
		BSID[23]	BSID[22]	BSID[21]	BSID[20]	BSID[19]	BSID[18]	BSID[17]	BSID[16]	
		BSID[15]	BSID[14]	BSID[13]	BSID[12]	BSID[11]	BSID[10]	BSID[9]	BSID[8]	
		BSID[7]	BSID[6]	BSID[5]	BSID[4]	BSID[3]	BSID[2]	BSID[1]	BSID[0]	

## 57.3 Register States

Table 57.3 Register States

Module	Register Name	Power-On Reset	Deep Standby	Software Standby	Module Standby	Sleep
Secondary cache	All registers	Initialized	Initialized	Retained	-	Retained
LSI internal bus	All registers	Initialized	Initialized	Retained	-	Retained
Clock pulse generator	FRQCR, FRQCR2	Initialized*1	Initialized	Retained	-	Retained
Interrupt controller	All registers	Initialized	Initialized	Retained	-	Retained
Bus state controller	RTCSR	Initialized	Initialized	Retained	-	Retained*2
	RTCNT	Initialized	Initialized	Retained	-	Retained*3
	All registers other than above	Initialized	Initialized	Retained	-	Retained
Direct memory access controller	All registers	Initialized	Initialized	Retained	-	Retained*4
Multi-function timer pulse unit 2	All registers	Initialized	Initialized	Initialized	Retained	Retained
OS timer	All registers	Initialized	Initialized	Retained	Retained	Retained
Watchdog timer	WRCSR	Initialized*1	Initialized	Retained	-	Retained
	All registers other than above	Initialized	Initialized	Retained	-	Retained
Realtime clock	R64CNT	Retained*3	Retained*3	Retained*3	Retained	Retained*3
	RSECCNT					
	RMINCNT					
	RHRCNT					
	RWKCNT					
	RDAYCNT					
	RMONCNT					
	RYRCNT					
	RSECAR	Retained	Retained	Retained	Retained	Retained
	RMINAR					
	RHRAR					
	RWKAR					
	RDAYAR					
	RMONAR					
	RYRAR					
	RCR1	Initialized	Initialized	Retained	Retained	Retained
	RCR2	Initialized	Initialized*5	Retained	Retained	Retained
	RCR3	Initialized	Initialized	Retained	Retained	Retained
	RCR5	Retained	Retained	Retained	Retained	Retained
	RFRH	Retained	Retained	Retained	Retained	Retained
RFRL	Retained	Retained	Retained	Retained	Retained	
Serial communication interface with FIFO	All registers	Initialized	Initialized	Retained	Retained	Retained
Serial communications interface	All registers	Initialized	Initialized	Retained	Retained	Retained
Renesas serial peripheral interface	All registers	Initialized	Initialized	Retained	Retained	Retained
SPI multi I/O bus controller	All registers	Initialized	Initialized	Retained	Retained	Retained
I <sup>2</sup> C bus interface	All registers	Initialized	Initialized	Retained	Retained	Retained
Serial sound interface	All registers	Initialized	Initialized	Retained	Retained	Retained
Media local bus	All registers	Initialized	Initialized	Retained	Retained	Retained
CAN interface	All registers	Initialized	Initialized	Retained	Retained	Retained
IEBus controller	All registers	Initialized	Initialized	Retained	Retained	Retained
Renesas SPDIF interface	All registers	Initialized	Initialized	Retained	Retained	Retained
CD-ROM decoder	All registers	Initialized	Initialized	Retained	Retained	Retained
LIN interface	All registers	Initialized	Initialized	Retained	Retained	Retained

**Table 57.3 Register States**

Module	Register Name	Power-On Reset	Deep Standby	Software Standby	Module Standby	Sleep
EthernetAVB	All registers	Initialized	Initialized	Retained	Retained	Retained
Ethernet controller	All registers	Initialized	Initialized	Retained	Retained	Retained
A/D converter	All registers	Initialized	Initialized	Initialized	Initialized	Retained
NAND flash memory controller	All registers	Initialized	Initialized	Retained	Retained	Retained
USB2.0 host/function module	All registers	Initialized	Initialized	Retained	Retained	Retained
Digital video decoder	All registers	Initialized	Initialized	Retained	Retained	Retained
Video display controller 5	All registers	Initialized	Initialized	Retained	Retained	Retained
LVDS output interface	All registers	Initialized	Initialized	Retained	Retained	Retained
Image renderer	All registers	Initialized	Initialized	Retained	Retained	Retained
Image renderer for display	All registers	Initialized	Initialized	Retained	Retained	Retained
Display out comparison unit	All registers	Initialized	Initialized	Retained	Retained	Retained
JPEG codec unit	All registers	Initialized	Initialized	Retained	Retained	Retained
Capture engine unit	All registers	Initialized	Initialized	Retained	Retained	Retained
Pixel format converter	All registers	Initialized	Initialized	Retained	Retained	Retained
SCUX	All registers	Initialized	Initialized	Retained	Retained	Retained
Sound generator	All registers	Initialized	Initialized	Retained	Retained	Retained
MMC host interface	All registers	Initialized	Initialized	Retained	Retained	Retained
Motor control PWM timer	All registers	Initialized	Initialized	Retained	Retained	Retained
Ports	All registers	Initialized	Initialized	Retained	-	Retained
Power-down modes	DSFR	Initialized	Retained	Retained	-	Retained
	XTALCTR	Initialized*1	Retained*6	Retained*6	-	Retained
	All registers other than above	Initialized	Initialized	Retained	-	Retained
Debugger interface	SDIR*7	Retained	Initialized	Retained	Retained	Retained
	Registers in CA9 PMU*8	Initialized	Initialized	Retained	-	Retained
	All registers other than above*9	Retained	Initialized*10	Retained	Retained	Retained

Note 1. Retained on internal power-on reset by the watchdog timer.

Note 2. Flag processing continues.

Note 3. Counting up continues.

Note 4. Transfer can proceed.

Note 5. The value of RTCEN bit is retained.

Note 6. The GAIN0 bit is initialized when the realtime clock is not using the EXTAL pin. The GAIN1 bit is initialized when the realtime clock is not using the RTC\_X1 or RTC\_X3 pin.

Note 7. Initialized on assertion of  $\overline{\text{TRST}}$  or when the TAP controller is placed in Test-Logic-Reset state.

Note 8. Access to these registers from the I/O area (SLV6) is not possible while TRST is asserted.

Note 9. Initialized on assertion of  $\overline{\text{TRST}}$ .

Note 10. Retained in FAKE debug mode.

## 58. Electrical Characteristics

### 58.1 Absolute Maximum Ratings

**Table 58.1 Absolute Maximum Ratings**

Item	Symbol	Value	Unit	
Power supply voltage (I/O)	PV <sub>CC</sub>	-0.3 to 4.2	V	
Power supply voltage (Internal)	V <sub>CC</sub>	-0.3 to 1.6	V	
PLL power supply voltage	PLL <sub>VCC</sub>	-0.3 to 1.6	V	
Analog power supply voltage	AV <sub>CC</sub>	-0.3 to 4.2	V	
Analog reference voltage	AV <sub>ref</sub>	-0.3 to AV <sub>CC</sub> +0.3	V	
USB transceiver analog power supply voltage (I/O)	USBAP <sub>VCC</sub>	-0.3 to 4.2	V	
USB transceiver digital power supply voltage (I/O) Note: Products in BGA packages do not have this pin.	USBDP <sub>VCC</sub>	-0.3 to 4.2	V	
USB transceiver analog power supply voltage (internal)	USBA <sub>VCC</sub>	-0.3 to 1.6	V	
USB transceiver digital power supply voltage (internal) Note: Products in BGA packages do not have this pin.	USBD <sub>VCC</sub>	-0.3 to 1.6	V	
Power supply for USB 480 MHz (internal) Note: Products in BGA packages do not have this pin.	USB <sub>UVCC</sub>	-0.3 to 1.6	V	
A/D converter power supply voltage for video signal input	VDA <sub>VCC</sub>	-0.3 to 4.2	V	
LVDS analog power supply voltage	LVDSAP <sub>VCC</sub>	-0.3 to 4.2	V	
LVDS PLL power supply voltage	LVDSPLL <sub>VCC</sub>	-0.3 to 1.6	V	
Input voltage	VBUS	V <sub>in</sub>	-0.3 to 5.5	V
	Other input pins	V <sub>in</sub>	-0.3- to 3.3-V power supply (PV <sub>CC</sub> , AV <sub>CC</sub> , USBAP <sub>VCC</sub> , USBDP <sub>VCC</sub> , VDA <sub>VCC</sub> , LVDSAP <sub>VCC</sub> ) +0.3	V
Operating temperature	Ambient temperature	T <sub>a</sub>	-40 to +85	°C
	Junction temperature	T <sub>j</sub>	-40 to +125	°C
Storage temperature	T <sub>stg</sub>	-55 to +125	°C	

Caution: Permanent damage to the LSI may result if absolute maximum ratings are exceeded.

### 58.2 Power-On/Power-Off Sequence

The 1.2-V power supply (V<sub>CC</sub>, PLL<sub>VCC</sub>, USBA<sub>VCC</sub>, USBD<sub>VCC</sub>, USB<sub>UVCC</sub>, and LVDSPLL<sub>VCC</sub>) and 3.3-V power supply (PV<sub>CC</sub>, AV<sub>CC</sub>, USBAP<sub>VCC</sub>, USBDP<sub>VCC</sub>, VDA<sub>VCC</sub>, and LVDSAP<sub>VCC</sub>) can be turned on and off in any order.

When turning on the power, be sure to drive both the  $\overline{\text{TRST}}$  and  $\overline{\text{RES}}$  pins low; otherwise, the output pins and input/output pins output undefined levels, resulting in system malfunction.

When turning off the power, drive the  $\overline{\text{TRST}}$  and  $\overline{\text{RES}}$  pins low if the undefined output may cause a problem.

### 58.3 DC Characteristics

- Conditions used to obtain DC characteristics (2) in Table 58.2 other than current consumption  
 $V_{CC} = USBDV_{CC} = USBUV_{CC} = 1.10$  to  $1.26$  V,  $PV_{CC} = USBDPV_{CC} = 3.0$  to  $3.6$  V,  $PLL_{VCC} = 1.10$  to  $1.26$  V,  
 $AV_{CC} = 3.0$  to  $3.6$  V,  $USBAPV_{CC} = 3.0$  to  $3.6$  V,  $USBAV_{CC} = 1.10$  to  $1.26$  V,  $VDAV_{CC} = 3.0$  to  $3.6$  V,  
 $LVDSAPV_{CC} = 3.0$  to  $3.6$  V,  $LVDSPLL_{VCC} = 1.10$  to  $1.26$  V,  
 $V_{SS} = AV_{SS} = USBDV_{SS} = USBAV_{SS} = USBDPV_{SS} = USBAPV_{SS} = USBUV_{SS} = VDAV_{SS} = LVDSAPV_{SS} = 0$  V,  
 $T_a = -40$  to  $85$  °C,  $T_j = -40$  to  $125$  °C
- Conditions used to obtain DC characteristics (2) in Table 58.2 for current consumption  
 $V_{CC} = USBDV_{CC} = USBUV_{CC} = 1.18$  V,  $PV_{CC} = USBDPV_{CC} = 3.3$  V,  $PLL_{VCC} = 1.18$  V,  $AV_{CC} = 3.3$  V,  
 $USBAPV_{CC} = 3.3$  V,  $USBAV_{CC} = 1.18$  V,  $VDAV_{CC} = 3.3$  V,  $LVDSAPV_{CC} = 3.3$  V,  $LVDSPLL_{VCC} = 1.18$  V,  
 $V_{SS} = AV_{SS} = USBDV_{SS} = USBAV_{SS} = USBDPV_{SS} = USBAPV_{SS} = USBUV_{SS} = VDAV_{SS} = LVDSAPV_{SS} = 0$  V,  
 $AV_{ref} = 3.3$  V,  $VBUS = 5.0$  V,  
 $T_a = -40$  to  $85$  °C,  $T_j = -40$  to  $125$  °C  
 $I_{\phi} = 400.00$  MHz,  $B_{\phi} = 133.33$  MHz,  $P1_{\phi} = 66.67$  MHz,  $P0_{\phi} = 33.33$  MHz

Note: Products in BGA packages do not have  $USBDV_{CC}$ ,  $USBUV_{CC}$ ,  $USBDPV_{CC}$ ,  $USBDV_{SS}$ ,  $USBAV_{SS}$ ,  $USBDPV_{SS}$ ,  $USBAPV_{SS}$ ,  $USBUV_{SS}$ , and  $LVDSAPV_{SS}$  pins.

**Table 58.2 DC Characteristics (1) [Common Items]**

Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions	
Power supply voltage	$PV_{CC}$	3.0	3.3	3.6	V		
	$V_{CC}$	1.10	1.18	1.26	V		
PLL power supply voltage	$PLL_{VCC}$	1.10	1.18	1.26	V		
Analog power supply voltage	$AV_{CC}$	3.0	3.3	3.6	V		
USB power supply voltage Note: Products in BGA packages do not have USB $DPV_{CC}$ , USB $DV_{CC}$ , and USB $UV_{CC}$ pins.	$USBAPV_{CC}$	3.0	3.3	3.6	V		
	$USBAV_{CC}$	1.10	1.18	1.26	V		
	$USBDV_{CC}$						
	$USBUV_{CC}$						
A/D converter power supply voltage for video signal input	$VDAV_{CC}$	3.0	3.3	3.6	V		
LVDS analog power supply voltage	$LVDSAPV_{CC}$	3.0	3.3	3.6	V		
LVDS PLL power supply voltage	$LVDSPLL_{VCC}$	1.10	1.18	1.26	V		
Input leakage current	All input pins	$ I_{in} $	—	—	1.0	$\mu$ A	$V_{in} = 0.5$ to $PV_{CC} - 0.5$ V
Three-state leakage current	All input/output pins, all output pins (except P1_0 to P7_0) (off state)	$ I_{STT} $	—	—	1.0	$\mu$ A	$V_{in} = 0.5$ to $PV_{CC} - 0.5$ V
	P1_0 to P7_0		—	—	10	$\mu$ A	
Input capacitance	all input/output pins, all input pins	$C_{in}$	—	—	10	pF	

**Table 58.2 DC Characteristics (2) [Current Consumption]**

Item	Power Supply	Symbol	Typ.	Max.	Unit	Test Conditions
Current consumption in normal operation	V <sub>CC</sub>	I <sub>CC</sub>	520	850	mA	
	PLL <sub>VCC</sub>	PLL <sub>I</sub> CC	6.4	8	mA	
	PV <sub>CC</sub>	PI <sub>CC</sub>	100*1	—	mA	
	AV <sub>CC</sub>	AI <sub>CC</sub>	2.7	4	mA	During A/D conversion
	AV <sub>ref</sub>	AI <sub>ref</sub>	0.6	1	mA	During A/D conversion
	USBAV <sub>CC</sub>	UA <sub>I</sub> CC	5	6	mA	When the USB host/function is in use.
	USBDV <sub>CC</sub> + USBUV <sub>CC</sub>	UDI <sub>CC</sub> *2	17	25	mA	In USB high-speed operation (2ch)
	USBAPV <sub>CC</sub>	UA <sub>PI</sub> CC	3.3	4	mA	When the USB host/function is in use.
	USBDPV <sub>CC</sub>	UD <sub>PI</sub> CC*3	70*1	—	mA	In USB high-speed operation (2ch)
	VDAV <sub>CC</sub>	VDA <sub>I</sub> CC	45	60	mA	
	LVDSAPV <sub>CC</sub>	LVDSAP <sub>I</sub> CC	30	35	mA	During LVDS transfer
	LVDSPLL <sub>VCC</sub>	LVDSPLL <sub>I</sub> CC	2	2.5	mA	
	VBUS	VI <sub>CC</sub>	8	10	μA	
Current consumption in sleep mode	V <sub>CC</sub>	I <sub>sleep</sub>	400	660	mA	
	For the other power supply, the current consumption is the same as in normal operation.					
Current consumption in software standby mode	Ta > 50 °C	V <sub>CC</sub> + PLL <sub>VCC</sub> + USBAV <sub>CC</sub> + USBDV <sub>CC</sub> + USBUV <sub>CC</sub> + LVDSPLL <sub>VCC</sub>	I <sub>sstby</sub>	72	320	mA
		PV <sub>CC</sub> + AV <sub>CC</sub> + AV <sub>ref</sub> + USBAPV <sub>CC</sub> + USBDPV <sub>CC</sub> + VDAV <sub>CC</sub> + LVDSAPV <sub>CC</sub>	PI <sub>sstby</sub>	25	32	μA
		VBUS	VI <sub>sstby</sub>	8	10	μA
	Ta ≤ 50 °C	V <sub>CC</sub> + PLL <sub>VCC</sub> + USBAV <sub>CC</sub> + USBDV <sub>CC</sub> + USBUV <sub>CC</sub> + LVDSPLL <sub>VCC</sub>	I <sub>sstby</sub>	20	160	mA
		PV <sub>CC</sub> + AV <sub>CC</sub> + AV <sub>ref</sub> + USBAPV <sub>CC</sub> + USBDPV <sub>CC</sub> + VDAV <sub>CC</sub> + LVDSAPV <sub>CC</sub>	PI <sub>sstby</sub>	18	21	μA
		VBUS	VI <sub>sstby</sub>	8	10	μA

Table 58.2 DC Characteristics (2) [Current Consumption]

Item		Power Supply	Symbol	Typ.	Max.	Unit	Test Conditions
Current consumption in deep standby mode	Ta > 50 °C	V <sub>CC</sub> + PLL <sub>VCC</sub> + USB <sub>AVCC</sub> + USB <sub>DVCC</sub> + USB <sub>UVCC</sub> + LVDS <sub>PLLCC</sub>	I <sub>dstby</sub>	124	180	μA	RAM 0 Kbytes retained, RTC_X1 selected
				148	204	μA	RAM 16 Kbytes retained, RTC_X1 selected
				164	228	μA	RAM 32 Kbytes retained, RTC_X1 selected
				200	276	μA	RAM 64 Kbytes retained, RTC_X1 selected
				289	372	μA	RAM 128 Kbytes retained, RTC_X1 selected
				When the EXTAL 13 MHz is selected, 5 μA and 7 μA are added to the "Typ." and "Max." values above, respectively. When the RTC_X3 is selected, 2 μA and 3 μA are added to the "Typ." and "Max." values above, respectively.			
	Ta ≤ 50 °C	PV <sub>CC</sub> + AV <sub>CC</sub> + AV <sub>ref</sub> + USB <sub>APVCC</sub> + USB <sub>DPVCC</sub> + V <sub>DAVCC</sub> + LVDS <sub>APVCC</sub>	P <sub>I</sub> dstby	18	26	μA	RTC is not operating
				25	36	μA	RTC_X1 selected
				0.6	—	mA	RTC_X3 selected, small gain*1
				1	—	mA	EXTAL 13 MHz selected, small gain*1
				8	10	μA	
				VBUS	V <sub>I</sub> dstby	8	10
Current consumption in deep standby mode	Ta > 50 °C	V <sub>CC</sub> + PLL <sub>VCC</sub> + USB <sub>AVCC</sub> + USB <sub>DVCC</sub> + USB <sub>UVCC</sub> + LVDS <sub>PLLCC</sub>	I <sub>dstby</sub>	40	70	μA	RAM 0 Kbytes retained, RTC_X1 selected
				50	90	μA	RAM 16 Kbytes retained, RTC_X1 selected
				60	110	μA	RAM 32 Kbytes retained, RTC_X1 selected
				80	150	μA	RAM 64 Kbytes retained, RTC_X1 selected
				120	230	μA	RAM 128 Kbytes retained, RTC_X1 selected
				When the EXTAL 13 MHz is selected, 5 μA and 7 μA are added to the "Typ." and "Max." values above, respectively. When the RTC_X3 is selected, 2 μA and 3 μA are added to the "Typ." and "Max." values above, respectively.			
	Ta ≤ 50 °C	PV <sub>CC</sub> + AV <sub>CC</sub> + AV <sub>ref</sub> + USB <sub>APVCC</sub> + USB <sub>DPVCC</sub> + V <sub>DAVCC</sub> + LVDS <sub>APVCC</sub>	P <sub>I</sub> dstby	15	19	μA	RTC is not operating
				22	29	μA	RTC_X1 selected
				0.6	—	mA	RTC_X3 selected, small gain*1
				1	—	mA	EXTAL 13 MHz selected, small gain*1
				8	10	μA	
				VBUS	V <sub>I</sub> dstby	8	10

Note 1. Reference value. The actual operating current greatly depends on the system (such as slow rising/falling edges caused by IO load and toggle frequency). Be sure to determine the value using the actual system.

Note 2. In the products in BGA packages, UD<sub>I</sub>cc is added to I<sub>cc</sub>.

Note 3. In the products in BGA packages, UD<sub>P</sub>cc is added to P<sub>I</sub>cc.

**Table 58.2 DC Characteristics (3) [Except I<sup>2</sup>C Bus Interface, and USB 2.0 Host/Function Module-Related Pins]**

Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions	
Input high voltage*	V <sub>IH</sub>	2.2	—	PV <sub>CC</sub> + 0.3	V		
Input low voltage*	V <sub>IL</sub>	-0.3	—	0.8	V		
Schmitt trigger input characteristics	V <sub>T+</sub>	PV <sub>CC</sub> × 0.66	—	—	V		
	V <sub>T-</sub>	—	—	0.8	V		
	V <sub>T+</sub> - V <sub>T-</sub>	0.2	—	—	V		
Output high voltage	V <sub>OH</sub>	PV <sub>CC</sub> - 0.5	—	—	V	I <sub>OH</sub> = -2.0 mA	
Output low voltage	V <sub>OL</sub>	—	—	0.4	V	I <sub>OL</sub> = 2.0 mA	
RAM standby voltage	Software standby mode (large-capacity on-chip RAM)	V <sub>RAMS</sub>	0.85	—	—	V	Measured with V <sub>CC</sub> as parameter
	Deep standby mode (only the on-chip RAM for data retention)	V <sub>RAMD</sub>	1.10	—	—	V	

Note: \* Except for the Schmitt pins and the input of data for the boundary scanning of pins P2\_0 to P2\_15, and P6\_0 to P6\_15.

**Table 58.2 DC Characteristics (4) [I<sup>2</sup>C Bus Interface Related Pins\*]**

Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Input high voltage	V <sub>IH</sub>	PV <sub>CC</sub> × 0.7	—	PV <sub>CC</sub> + 0.3	V	
Input low voltage	V <sub>IL</sub>	-0.3	—	PV <sub>CC</sub> × 0.3	V	
Schmitt trigger input characteristics	V <sub>IH</sub> - V <sub>IL</sub>	PV <sub>CC</sub> × 0.05	—	—	V	
Output low voltage	V <sub>OL</sub>	—	—	0.4	V	I <sub>OL</sub> = 3.0 mA

Note: \* The P1\_0 to P7\_0 pins are open-drain pins.

**Table 58.2 DC Characteristics (5) [USB 2.0 Host/Function Module-Related Pins\*]**

Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Reference resistance	R <sub>REF</sub>	5.6 kΩ ± 1%	5.6 kΩ ± 1%	5.6 kΩ ± 1%		
Input high voltage (VBUS1, VBUS0)	V <sub>IH</sub>	4.02	—	5.25	V	
Input low voltage (VBUS1, VBUS0)	V <sub>IL</sub>	-0.3	—	0.5	V	
Input high voltage (USB_X1)	V <sub>IH</sub>	PV <sub>CC</sub> - 0.5	—	PV <sub>CC</sub> + 0.3	V	
Input low voltage (USB_X1)	V <sub>IL</sub>	-0.3	—	0.5	V	

Note: \* REFRIN, VBUS1, VBUS0, USB\_X1, and USB\_X2 pins



**Table 58.2 DC Characteristics (6) [USB 2.0 Host/Function Module-Related Pins\* (Low-Speed, Full-Speed, and High-Speed Common Items)]**

Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
DP pull-up resistance (when function is selected)	R <sub>pu</sub>	0.900	—	1.575	kΩ	In idle mode
		1.425	—	3.090	kΩ	In transmit/receive mode
DP and DM pull-down resistance (when host is selected)	R <sub>pd</sub>	14.25	—	24.80	kΩ	

Note: \* DP1, DP0, DM1, and DM pins

**Table 58.2 DC Characteristics (7) [USB 2.0 Host/Function Module-Related Pins\* (Low-Speed and Full-Speed)]**

Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Input high voltage	V <sub>IH</sub>	2.0	—	—	V	
Input low voltage	V <sub>IL</sub>	—	—	0.8	V	
Differential input sensitivity	V <sub>DI</sub>	0.2	—	—	V	(DP) – (DM)
Differential common mode range	V <sub>CM</sub>	0.8	—	2.5	V	
Output high voltage	V <sub>OH</sub>	2.8	—	3.6	V	I <sub>OH</sub> = –200 μA
Output low voltage	V <sub>OL</sub>	0.0	—	0.3	V	I <sub>OL</sub> = 2 mA
Output signal crossover voltage	V <sub>CRS</sub>	1.3	—	2.0	V	C <sub>L</sub> = 50 pF (full-speed) C <sub>L</sub> = 200 to 600 pF (low-speed)

Note: \* DP1, DP0, DM1, and DM pins

**Table 58.2 DC Characteristics (8) [USB 2.0 Host/Function Module-Related Pins\* (High-Speed)]**

Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Squelch detection threshold voltage (differential voltage)	V <sub>HSSQ</sub>	100	—	150	mV	
Common mode voltage range	V <sub>HSCM</sub>	–50	—	500	mV	
Idle state	V <sub>HSOI</sub>	–10.0	—	10.0	mV	
Output high voltage	V <sub>HSOH</sub>	360	—	440	mV	
Output low voltage	V <sub>HSOL</sub>	–10.0	—	10.0	mV	
Chirp J output voltage (difference)	V <sub>CHIRPJ</sub>	700	—	1100	mV	
Chirp K output voltage (difference)	V <sub>CHIRPK</sub>	–900	—	–500	mV	

Note: \* DP1, DP0, DM1, and DM pins

**Table 58.2 DC Characteristics (9) [LVDS-Related Pins\*]**

Item	Symbol	Min.	Typ.	Max.	Unit	Test Conditions
Reference resistance	R <sub>LVDSREF</sub>		5.6kΩ±1%			
Differential Output Voltage	V <sub>OD</sub>	250	350	450	mV	R <sub>L</sub> = 100Ω
Difference V <sub>OD</sub> between 'H' and 'L'	ΔV <sub>OD</sub>	—	—	50	mV	R <sub>L</sub> = 100Ω
Offset (Common Mode) Voltage	V <sub>OS</sub>	1.125	1.25	1.375	V	R <sub>L</sub> = 100Ω
Difference V <sub>OS</sub> between 'H' and 'L'	ΔV <sub>OS</sub>	—	—	50	mV	R <sub>L</sub> = 100Ω

Note: \* LVDSREFRIN, TXCLKOUTP, TXCLKOUTM, TXOUT2P to TXOUT0P, and TXOUT2M to TXOUT0M pins

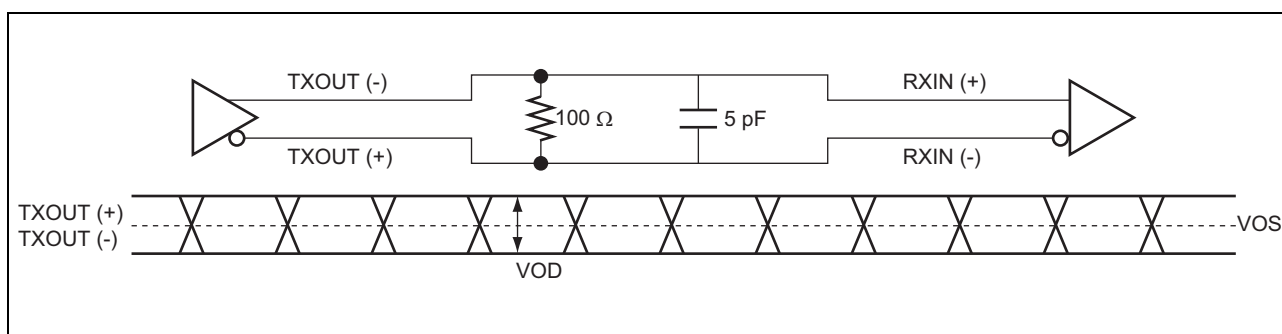


Figure 58.1 LVDS Output Waveform

Table 58.3 Permissible Output Currents

Item	Symbol	Min.	Typ.	Max.	Unit
Permissible output low current (per pin)	P1_0 to P7_0	—	—	10	mA
	Output pins other than above	—	—	2	mA
Permissible output high current (per pin)	$-I_{OH}$	—	—	2	mA
Permissible output current (total)	$\Sigma I_O$	—	—	150	mA

Caution: To protect the LSI's reliability, do not exceed the output current values in Table 58.3.

## 58.4 AC Characteristics

Signals input to this LSI are basically handled as signals in synchronization with a clock. The setup and hold times for input pins must be followed.

Conditions for AC characteristics:  $V_{CC} = USBDV_{CC} = USBUV_{CC} = 1.10$  to  $1.26$  V,  $PV_{CC} = USBDPV_{CC} = 3.0$  to  $3.6$  V,  $PLL_{VCC} = 1.10$  to  $1.26$  V,  $AV_{CC} = 3.0$  to  $3.6$  V,  $USBAPV_{CC} = 3.0$  to  $3.6$  V,  $USBV_{CC} = 1.10$  to  $1.26$  V,  $VDAV_{CC} = 3.0$  to  $3.6$  V,  $LVDSAPV_{CC} = 3.0$  to  $3.6$  V,  $LVDSPLL_{VCC} = 1.10$  to  $1.26$  V,  
 $V_{SS} = AV_{SS} = USBDV_{SS} = USBV_{SS} = USBDPV_{SS} = USBAPV_{SS} = USBUV_{SS} = VDAV_{SS} = LVDSAPV_{SS} = 0$  V,  
 $T_a = -40$  to  $85$  °C,  $T_j = -40$  to  $125$  °C

Note: Products in BGA packages do not have  $USBDV_{CC}$ ,  $USBV_{CC}$ ,  $USBDPV_{CC}$ ,  $USBV_{SS}$ ,  $USBV_{SS}$ ,  $USBAPV_{SS}$ ,  $USBV_{SS}$ , and  $LVDSAPV_{SS}$  pins.

**Table 58.4 Operating Frequency**

Item		Symbol	Min.	Max.	Unit	Remarks
Operating frequency	CPU clock ( $I\phi$ )	f	100.00	400.00	MHz	
	Image processing clock ( $G\phi$ )		100.00	266.67	MHz	
	Internal bus clock ( $B\phi$ )		100.00	133.33	MHz	
	Peripheral clock 1 ( $P1\phi$ )		50.00	66.67	MHz	
	Peripheral clock 0 ( $P0\phi$ )		25.00	33.33	MHz	

## 58.4.1 Clock Timing

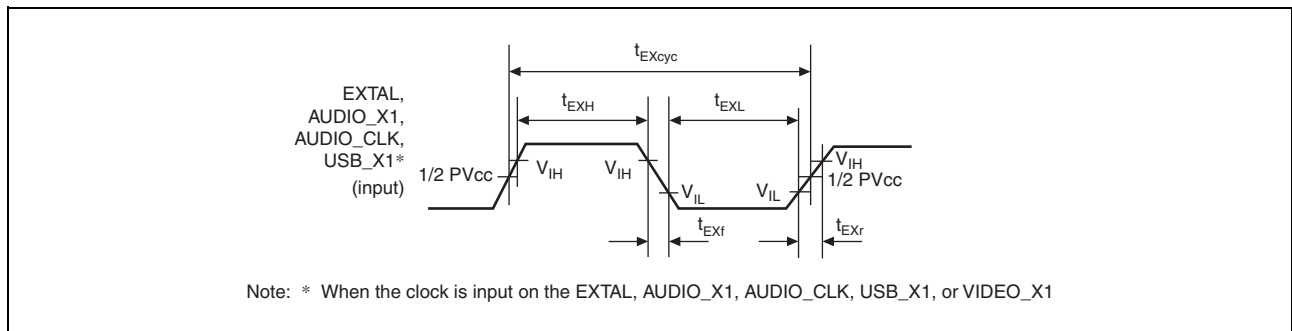
Table 58.5 Clock Timing

Item	Symbol	Min.	Max.	Unit	Figure
EXTAL clock input frequency (when the clock is supplied to USB 2.0 host/function module)	f <sub>EX</sub>	12MHz ± 100ppm			Figure 58.2
EXTAL clock input frequency (when the clock isn't supplied to USB 2.0 host/function module)		10.00	13.33	MHz	
EXTAL clock input cycle time (when the clock isn't supplied to USB 2.0 host/function module)	t <sub>EXcyc</sub>	75.00	100.00	ns	
AUDIO_X1 clock input frequency (crystal resonator connected)	f <sub>EX</sub>	10.00	50.00	MHz	
AUDIO_X1 clock input cycle time (crystal resonator connected)	t <sub>EXcyc</sub>	20.00	100.00	ns	
AUDIO_X1, AUDIO_CLK clock input frequency (external clock input)	f <sub>EX</sub>	1.00	50.00	MHz	
AUDIO_X1, AUDIO_CLK clock input cycle time (external clock input)	t <sub>EXcyc</sub>	20.00	1000.00	ns	
USB_X1 clock input frequency (when the 48-MHz clock is supplied to USB 2.0 host/function module and high-speed transfer function is used)	f <sub>EX</sub>	48 MHz ±100 ppm			
USB_X1 clock input frequency (when the 48-MHz clock is supplied to USB 2.0 host/function module, high-speed transfer function is not used, and host controller function is used)		48 MHz ±500 ppm			
USB_X1 clock input frequency (when the 48-MHz clock is supplied to USB 2.0 host/function module, high-speed transfer function is not used, and host controller function is not used)		48 MHz ±2500 ppm			
VIDEO_X1 clock input frequency		27MHz ±50 ppm*			
EXTAL, AUDIO_X1, AUDIO_CLK, USB_X1 clock input low pulse width	t <sub>EXL</sub>	0.4	0.6	t <sub>EXcyc</sub>	
VIDEO_X1 clock input low pulse width		0.45	0.55		
EXTAL, AUDIO_X1, AUDIO_CLK, USB_X1 clock input high pulse width	t <sub>EXH</sub>	0.4	0.6	t <sub>EXcyc</sub>	
VIDEO_X1 clock input high pulse width		0.45	0.55		
EXTAL, AUDIO_X1, AUDIO_CLK, USB_X1 clock input rise time	t <sub>EXr</sub>	—	4	ns	
VIDEO_X1 clock input rise time		—	3		
EXTAL, AUDIO_X1, AUDIO_CLK, USB_X1 clock input fall time	t <sub>EXf</sub>	—	4	ns	
VIDEO_X1 clock input fall time		—	3		
CKIO clock output frequency	f <sub>OP</sub>	50.00	66.67	MHz	
CKIO clock output cycle time	t <sub>cyc</sub>	15.00	20.00	ns	Figure 58.3 (1) and Figure 58.3 (2)
CKIO clock output low pulse width 1	t <sub>CKOL1</sub>	t <sub>cyc</sub> /2 – t <sub>CKOr1</sub>	—	ns	Figure 58.3 (1)
CKIO clock output high pulse width 1	t <sub>CKOH1</sub>	t <sub>cyc</sub> /2 – t <sub>CKOr1</sub>	—	ns	
CKIO clock output rise time 1	t <sub>CKOr1</sub>	—	3	ns	
CKIO clock output fall time 1	t <sub>CKOf1</sub>	—	3	ns	
CKIO clock output low pulse width 2	t <sub>CKOL2</sub>	t <sub>cyc</sub> /2 – t <sub>CKOr2</sub>	—	ns	Figure 58.3 (2)
CKIO clock output high pulse width 2	t <sub>CKOH2</sub>	t <sub>cyc</sub> /2 – t <sub>CKOr2</sub>	—	ns	
CKIO clock output rise time 2	t <sub>CKOr2</sub>	—	2	ns	
CKIO clock output fall time 2	t <sub>CKOf2</sub>	—	2	ns	

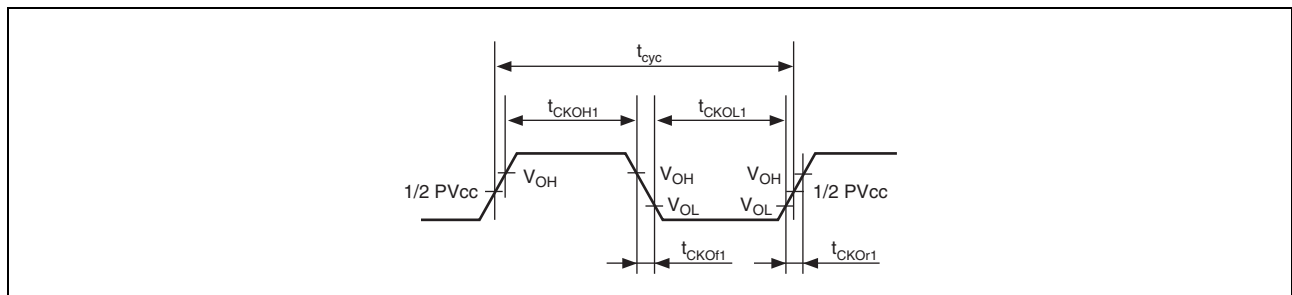
**Table 58.5 Clock Timing**

Item	Symbol	Min.	Max.	Unit	Figure
Power-on oscillation settling time	$t_{OSC1}$	10	—	ms	Figure 58.4
Oscillation settling time 1 on return from standby	$t_{OSC2}$	10	—	ms	Figure 58.5
RTC_X1 real time clock oscillation settling time	$t_{ROSC}$	—	3	s	Figure 58.7
RTC_X3 real time clock oscillation settling time	$t_{ROSC}$	—	10	ms	Figure 58.7
Mode hold time	$t_{MDH}$	200	—	ns	Figure 58.4 and Figure 58.5

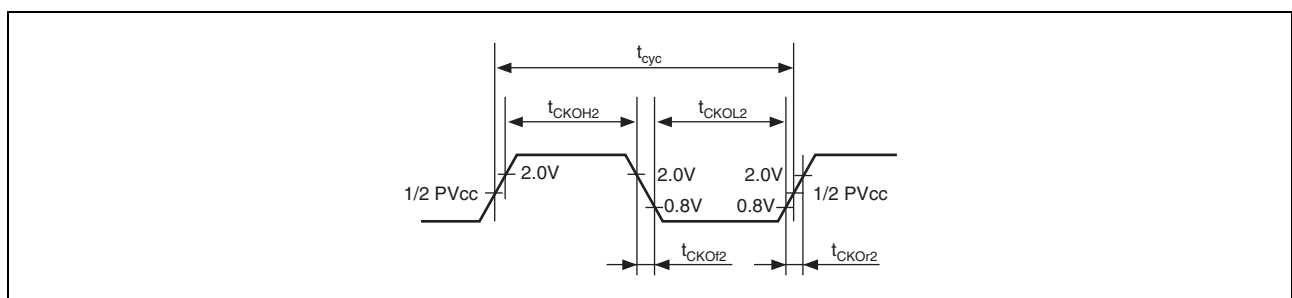
Note: \* Reference value. The accuracy of the clock signal affects the quality of images output by the digital video decoder. Input clock signals that are as accurate as is possible.



**Figure 58.2 EXTAL, AUDIO\_X1, AUDIO\_CLK, USB\_X1, and VIDEO\_X1 Clock Input Timing**



**Figure 58.3 (1) CKIO Clock Output Timing 1**



**Figure 58.3 (2) CKIO Clock Output Timing 2**

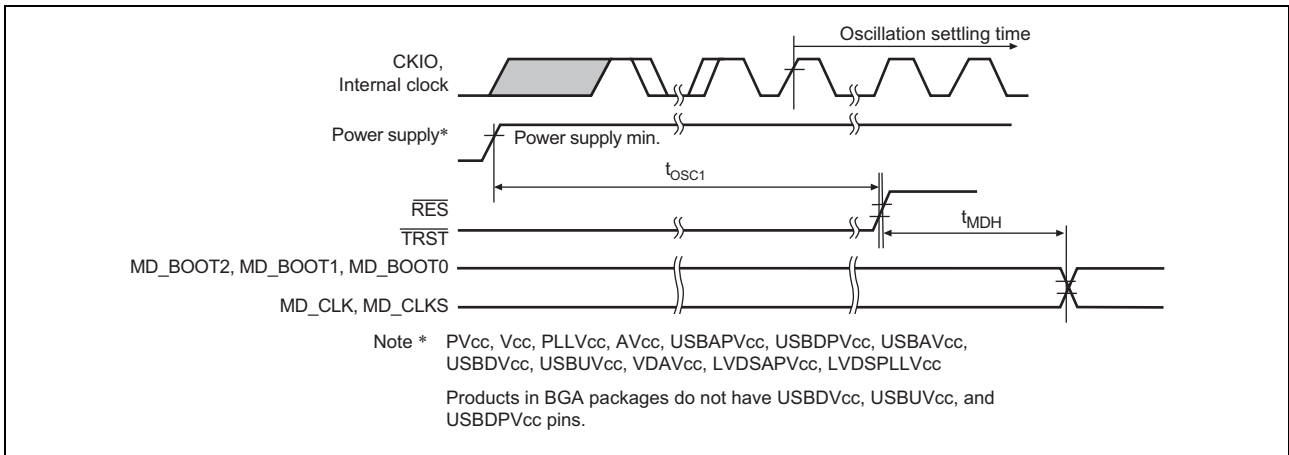


Figure 58.4 Power-On Oscillation Settling Time

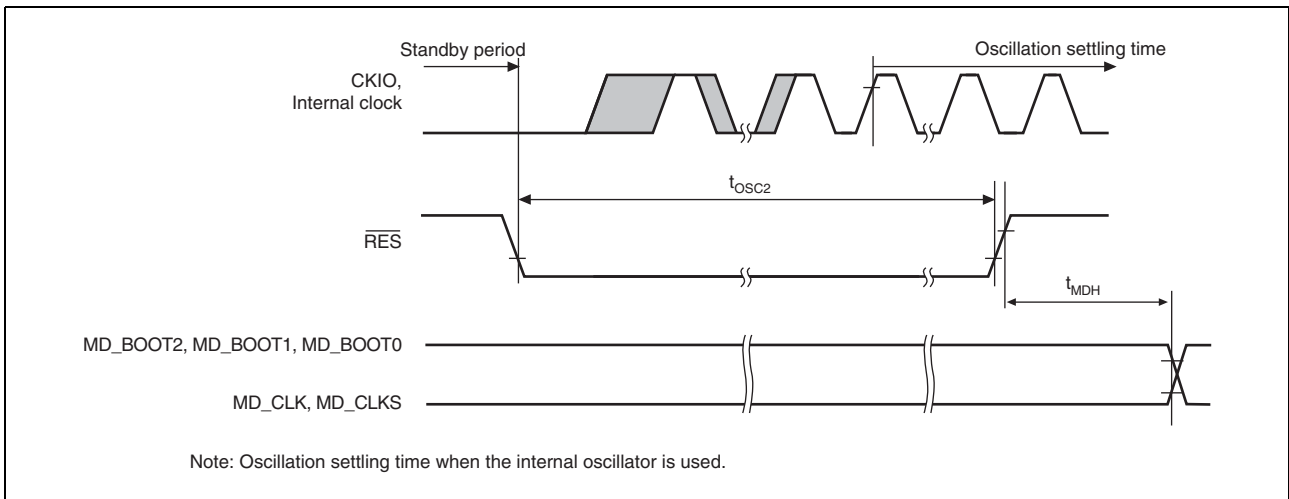


Figure 58.5 Oscillation Settling Time on Return from Standby (Return by Reset)

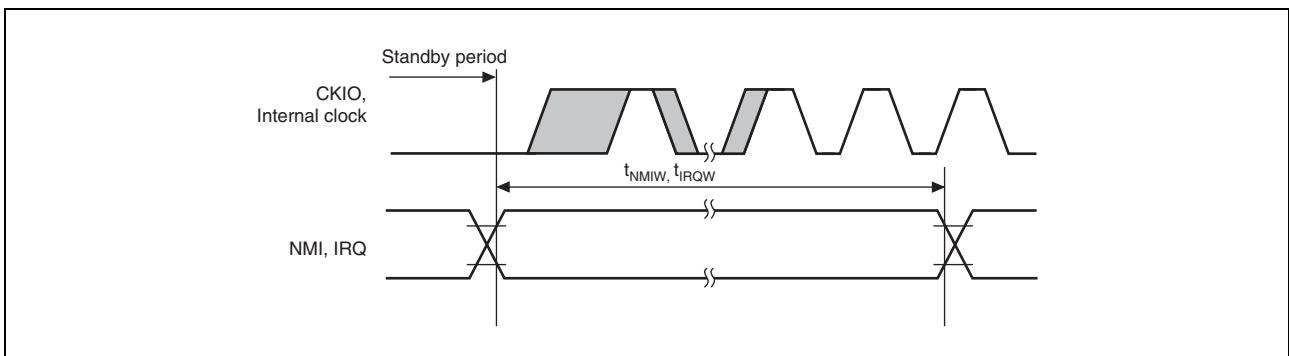
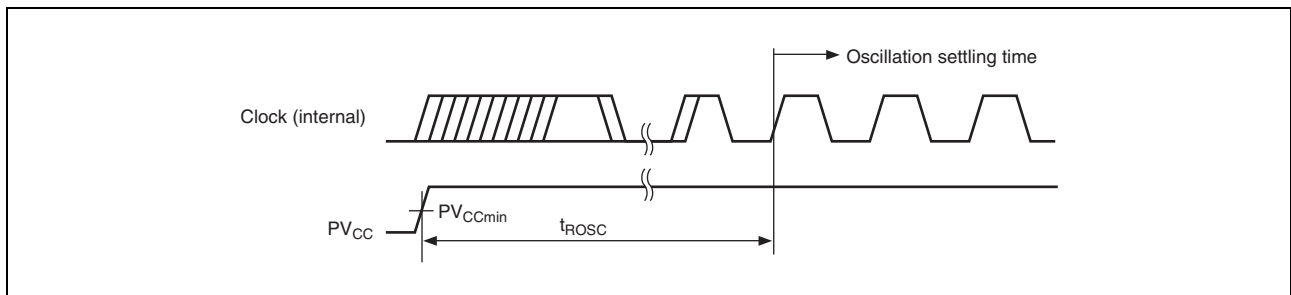


Figure 58.6 Oscillation Settling Time on Return from Standby (Return by NMI or IRQ)

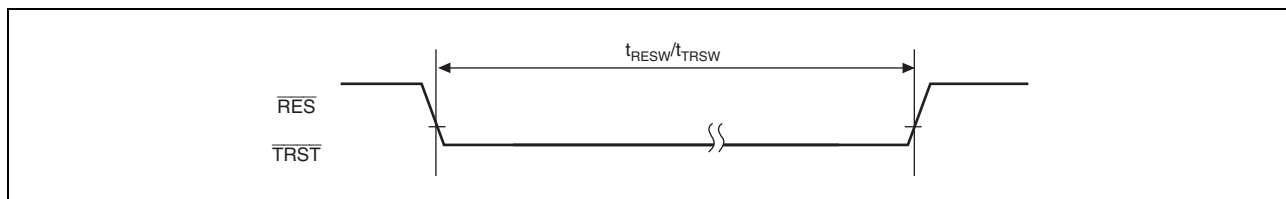


**Figure 58.7 Real Time Clock Oscillation Settling Time**

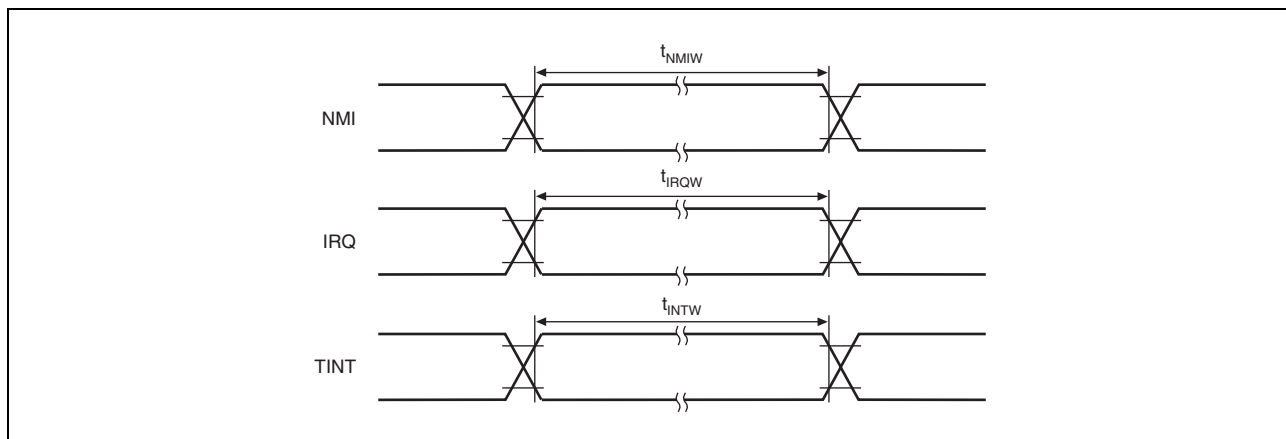
### 58.4.2 Control Signal Timing

**Table 58.6 Control Signal Timing**

Item		Symbol	Min.	Max.	Unit	Figure
RES pulse width	Exit from standby mode	$t_{RESW}$	10	—	ms	Figure 58.8 (1)
	Other than above		20	—	$t_{cyc}$	
TRST pulse width		$t_{TRSW}$	20	—	$t_{cyc}$	Figure 58.8 (2) and Figure 58.6
NMI pulse width		$t_{NMIW}$	20	—	$t_{cyc}$	
IRQ pulse width		$t_{IRQW}$	20	—	$t_{cyc}$	
TINT pulse width		$t_{TINTW}$	20	—	$t_{cyc}$	



**Figure 58.8 (1) Reset Input Timing**



**Figure 58.8 (2) Interrupt Signal Input Timing**



## 58.4.3 Bus Timing

Table 58.7 Bus Timing

Item	Symbol	CKIO = 66.67 MHz*1		Unit	Figure
		Min.	Max.		
Address delay time 1	t <sub>AD1</sub>	0/2*3	12	ns	Figure 58.9 to Figure 58.33
Address delay time 2	t <sub>AD2</sub>	1/2t <sub>cyc</sub>	1/2t <sub>cyc</sub> + 12	ns	Figure 58.16
Address setup time	t <sub>AS</sub>	0	—	ns	Figure 58.9 to Figure 58.12, Figure 58.16
Chip enable setup time	t <sub>CS</sub>	0	—	ns	Figure 58.9 to Figure 58.12, Figure 58.16
Address hold time	t <sub>AH</sub>	0	—	ns	Figure 58.9 to Figure 58.12
$\overline{\text{BS}}$ delay time	t <sub>BSD</sub>	—	12	ns	Figure 58.9 to Figure 58.30
$\overline{\text{CS}}$ delay time 1	t <sub>CSD1</sub>	0/2*3	12	ns	Figure 58.9 to Figure 58.33
Read write delay time 1	t <sub>RWD1</sub>	0/2*3	12	ns	Figure 58.9 to Figure 58.33
Read strobe delay time	t <sub>RSD</sub>	1/2t <sub>cyc</sub>	1/2t <sub>cyc</sub> + 12	ns	Figure 58.9 to Figure 58.16
Read data setup time 1	t <sub>RDS1</sub>	1/2t <sub>cyc</sub> + 5	—	ns	Figure 58.9 to Figure 58.15
Read data setup time 2	t <sub>RDS2</sub>	7	—	ns	Figure 58.17 to Figure 58.20, Figure 58.25 to Figure 58.27
Read data setup time 3	t <sub>RDS3</sub>	1/2t <sub>cyc</sub> + 5	—	ns	Figure 58.16
Read data hold time 1	t <sub>RDH1</sub>	0	—	ns	Figure 58.9 to Figure 58.15
Read data hold time 2	t <sub>RDH2</sub>	2	—	ns	Figure 58.17 to Figure 58.20, Figure 58.25 to Figure 58.27
Read data hold time 3	t <sub>RDH3</sub>	0	—	ns	Figure 58.16
Write enable delay time 1	t <sub>WED1</sub>	1/2t <sub>cyc</sub>	1/2t <sub>cyc</sub> + 12	ns	Figure 58.9 to Figure 58.14
Write enable delay time 2	t <sub>WED2</sub>	—	12	ns	Figure 58.15
Write data delay time 1	t <sub>WDD1</sub>	—	12	ns	Figure 58.9 to Figure 58.15
Write data delay time 2	t <sub>WDD2</sub>	—	12	ns	Figure 58.21 to Figure 58.24, Figure 58.28 to Figure 58.30
Write data hold time 1	t <sub>WDH1</sub>	1	—	ns	Figure 58.9 to Figure 58.15
Write data hold time 2	t <sub>WDH2</sub>	2	—	ns	Figure 58.21 to Figure 58.24, Figure 58.28 to Figure 58.30
Write data hold time 4	t <sub>WDH4</sub>	0	—	ns	Figure 58.9 to Figure 58.13
$\overline{\text{WAIT}}$ setup time	t <sub>WTS</sub>	1/2t <sub>cyc</sub> + 4.5	—	ns	Figure 58.10 to Figure 58.16
$\overline{\text{WAIT}}$ hold time	t <sub>WTH</sub>	1/2t <sub>cyc</sub> + 3.5	—	ns	Figure 58.10 to Figure 58.16
$\overline{\text{RAS}}$ delay time 1	t <sub>RASD1</sub>	2	12	ns	Figure 58.17 to Figure 58.33
$\overline{\text{CAS}}$ delay time 1	t <sub>CASD1</sub>	2	12	ns	Figure 58.17 to Figure 58.33
DQM delay time 1	t <sub>DQMD1</sub>	2	12	ns	Figure 58.17 to Figure 58.30
CKE delay time 1	t <sub>CKED1</sub>	2	12	ns	Figure 58.32
AH delay time	t <sub>AHD</sub>	1/2t <sub>cyc</sub>	1/2t <sub>cyc</sub> + 12	ns	Figure 58.13
Multiplexed address delay time	t <sub>MAD</sub>	—	12	ns	Figure 58.13
Multiplexed address hold time	t <sub>MAH</sub>	1	—	ns	Figure 58.13
Address setup time for AH	t <sub>AVVH</sub>	1/2t <sub>cyc</sub> - 2	—	ns	Figure 58.13
DACK, TEND delay time	t <sub>DACD</sub>	Refer to the direct memory access controller timing		ns	Figure 58.9 to Figure 58.30

Note 1. The maximum value (fmax) of CKIO (external bus clock) depends on the number of wait cycles and the system configuration of your board.

Note 2. 1/2 t<sub>cyc</sub> indicated in minimum and maximum values for the item of delay, setup, and hold times represents a half cycle from the rising edge with a clock. That is, 1/2 t<sub>cyc</sub> describes a reference of the falling edge with a clock.

Note 3. Values when SDRAM is used.

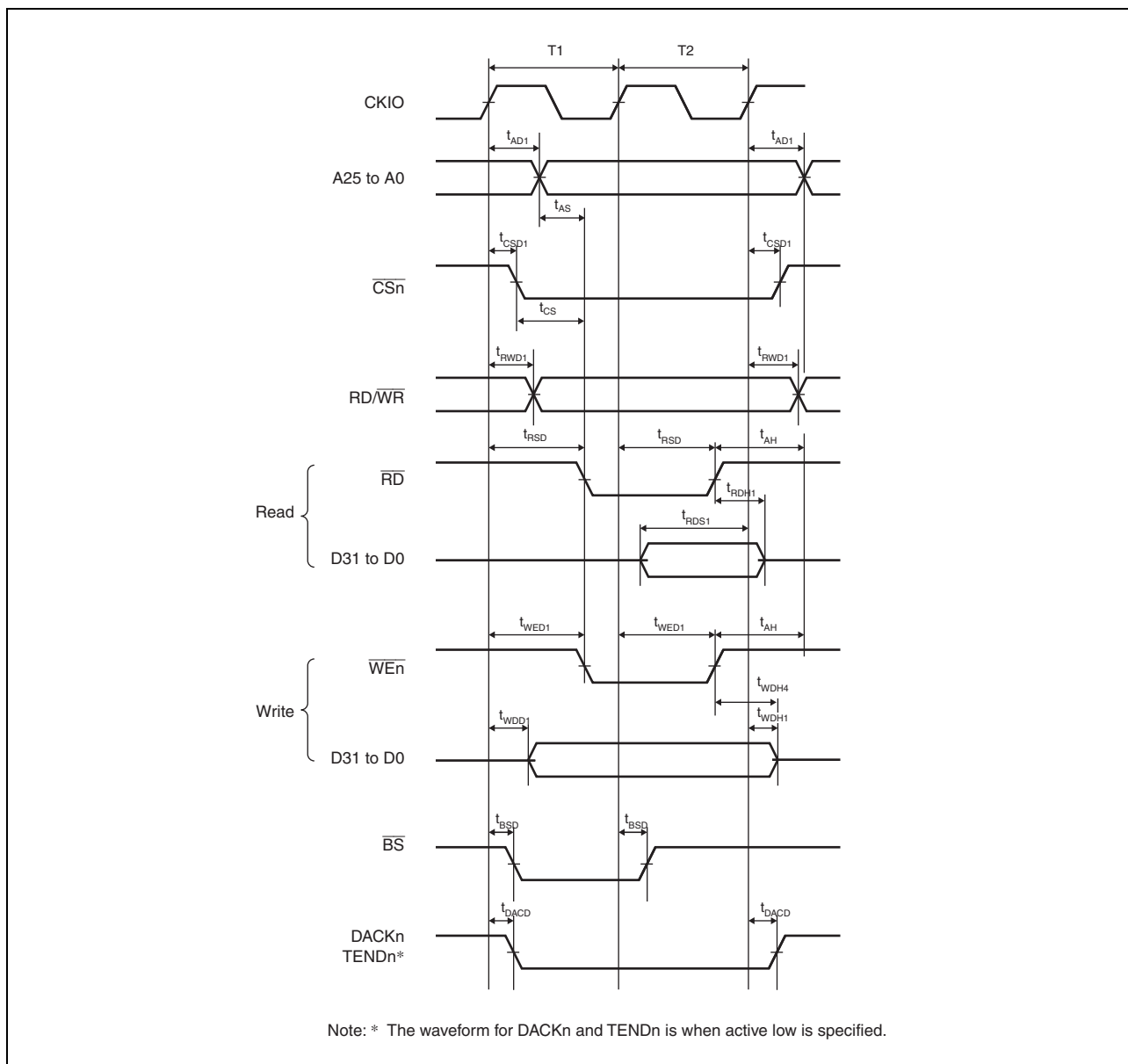


Figure 58.9 Basic Bus Timing for Normal Space (No Wait)

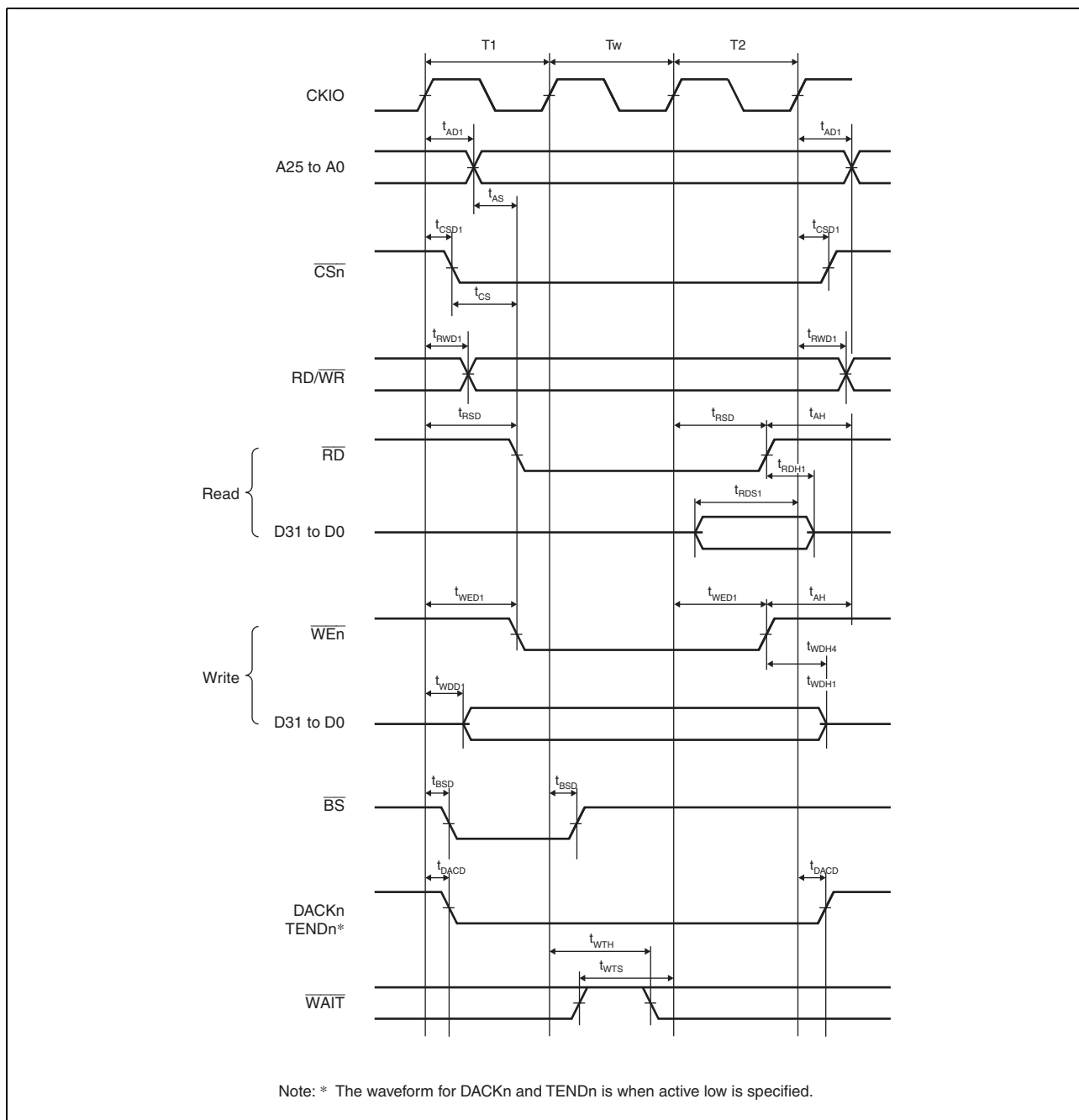


Figure 58.10 Basic Bus Timing for Normal Space (One Software Wait Cycle)

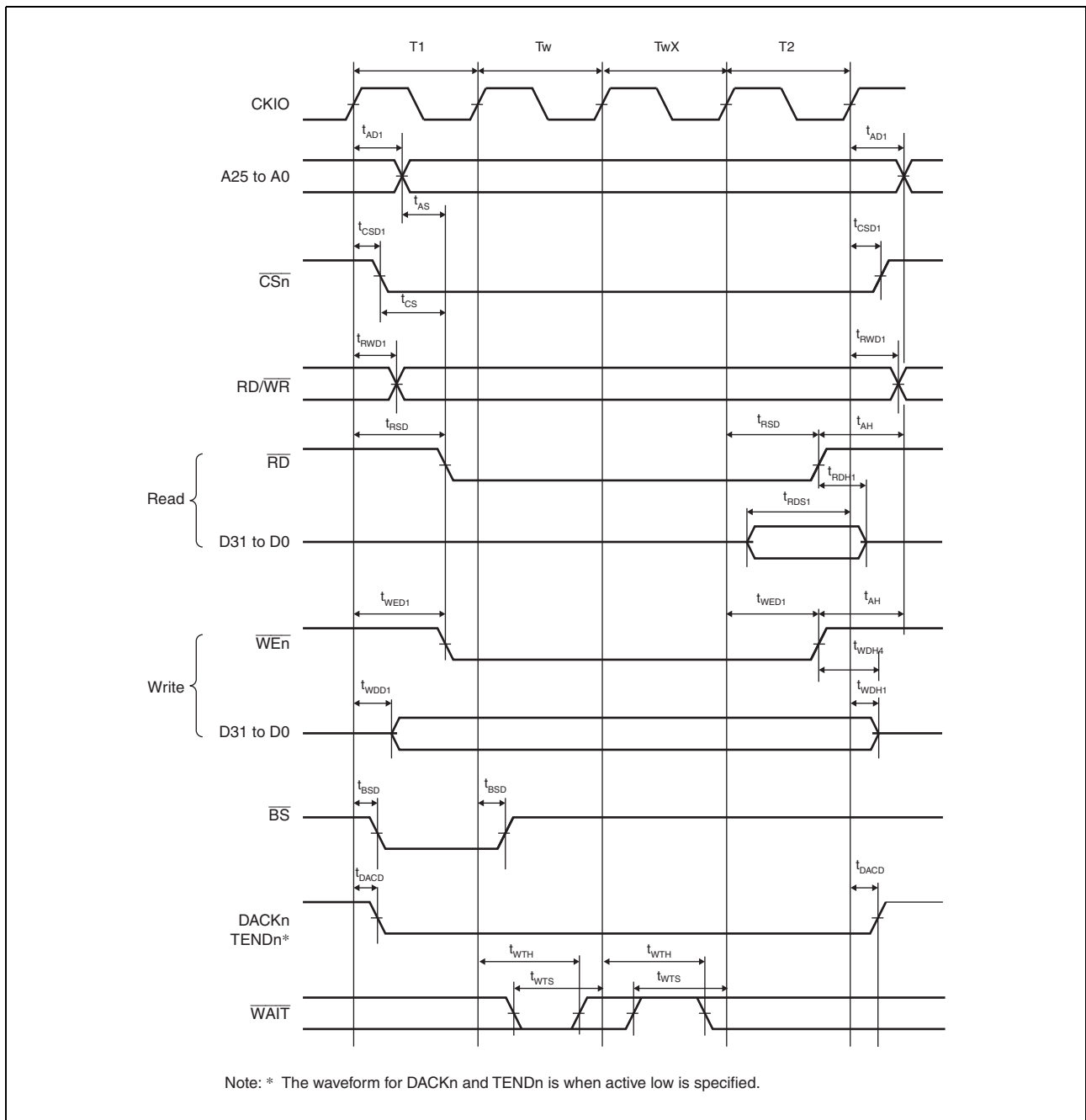


Figure 58.11 Basic Bus Timing for Normal Space (One Software Wait Cycle, One External Wait Cycle)

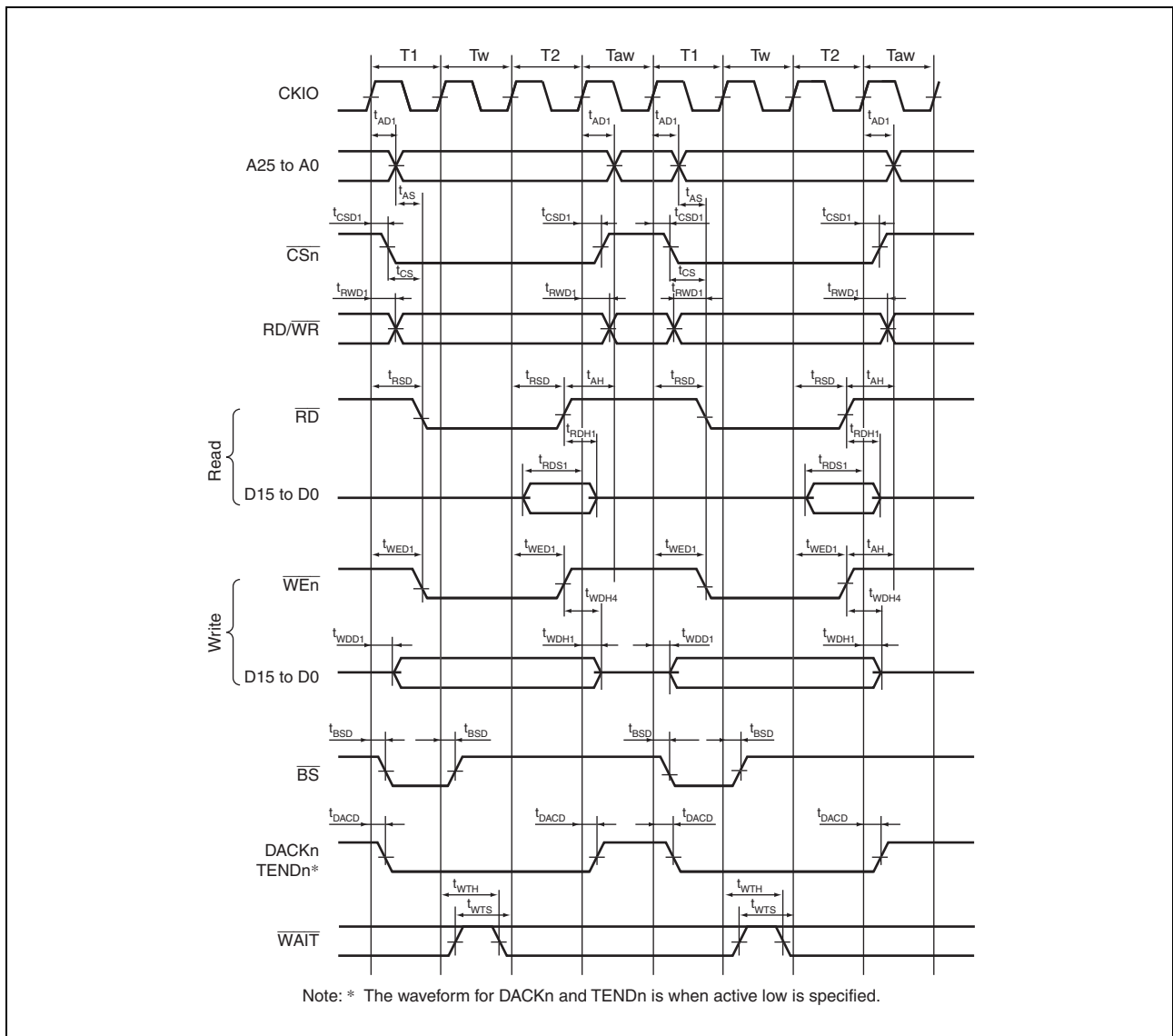
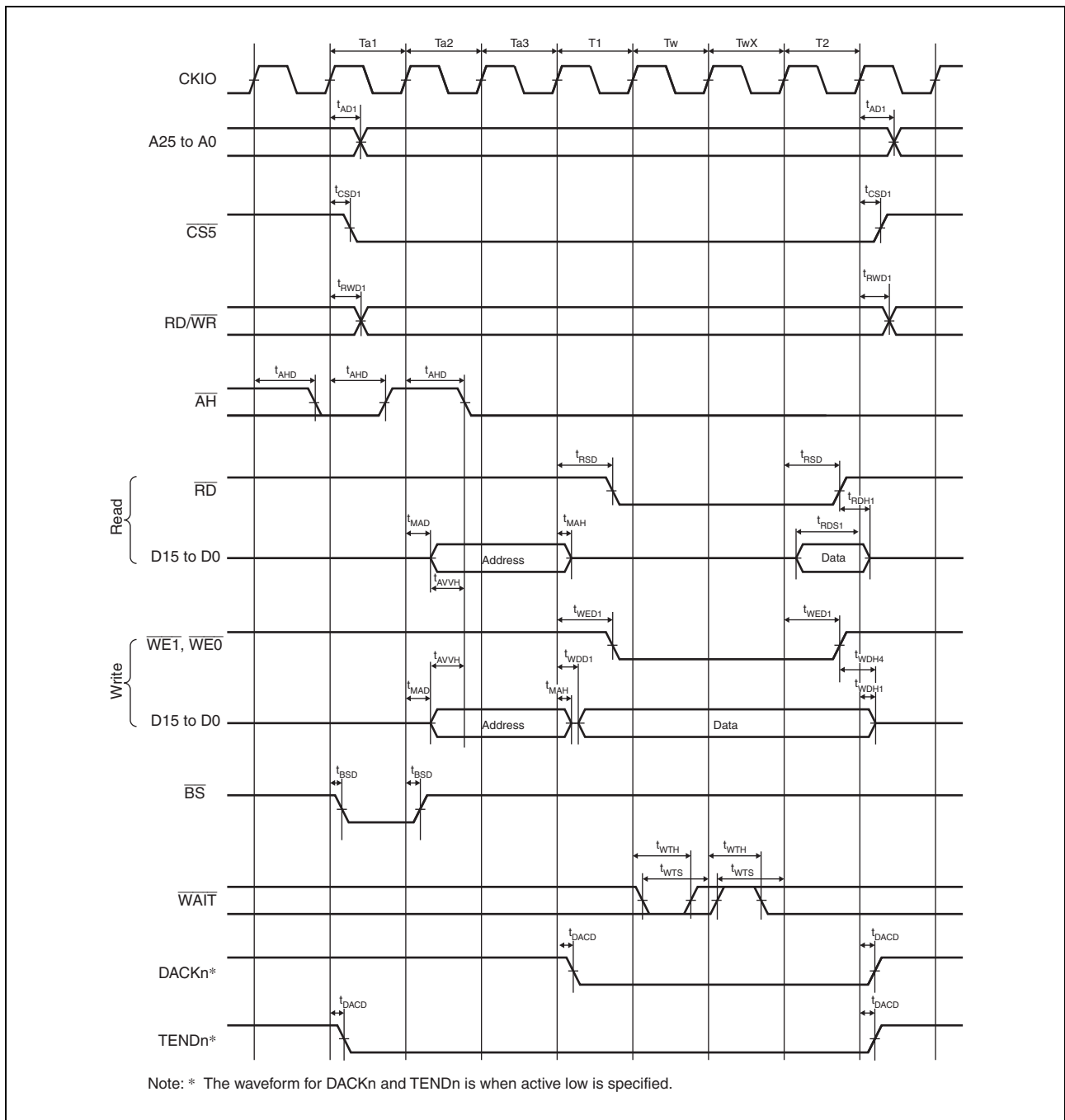
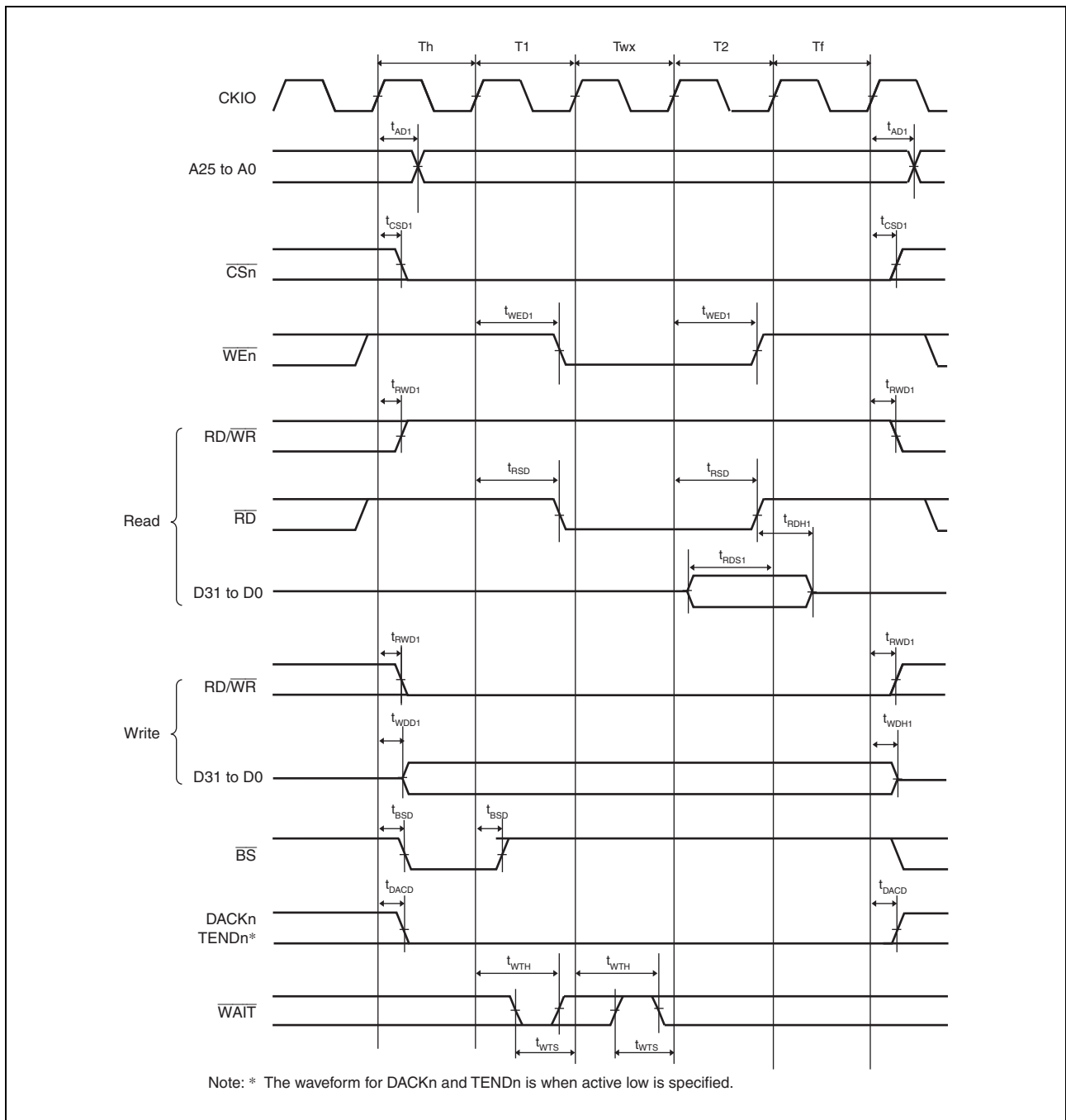


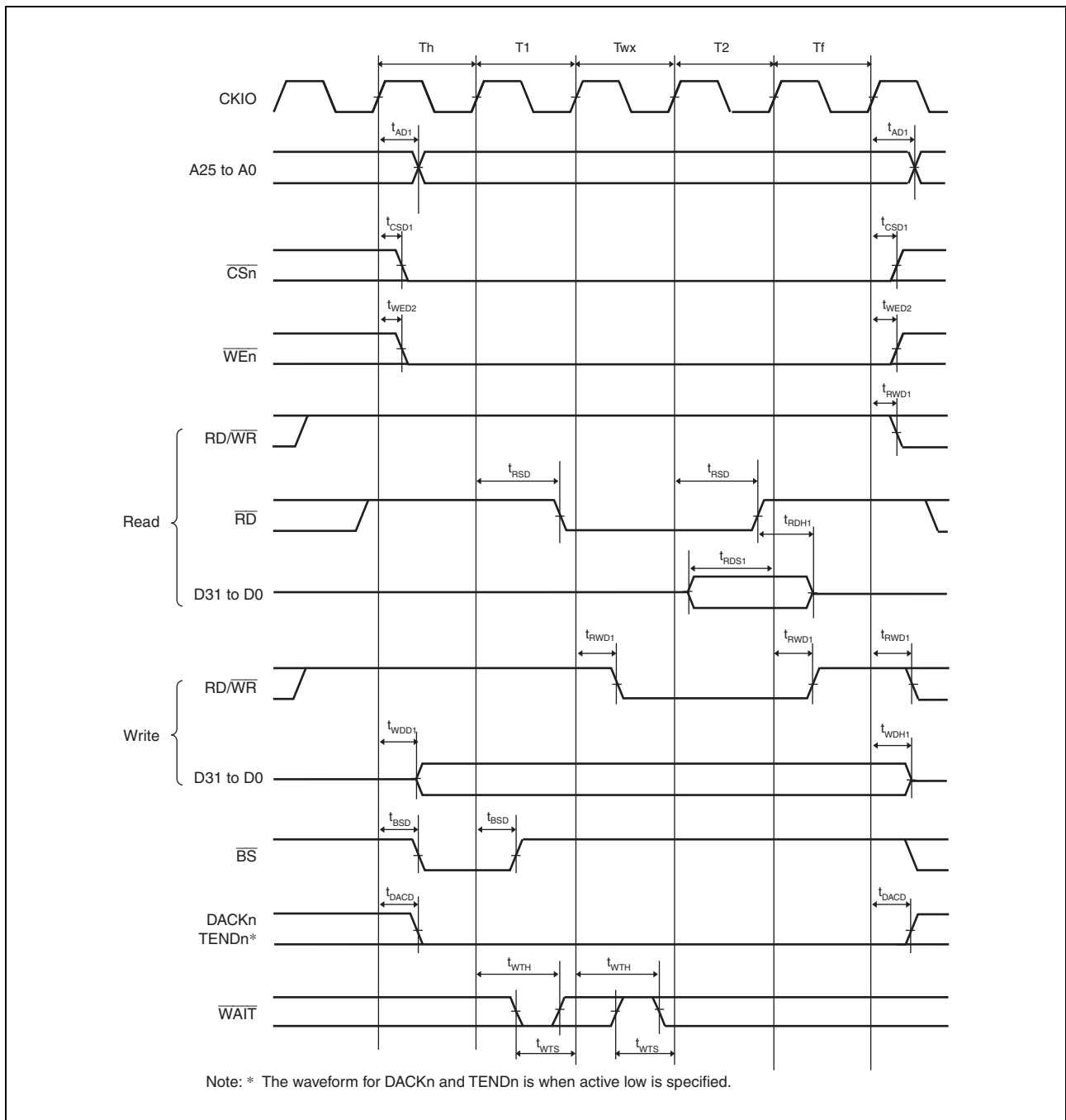
Figure 58.12 Basic Bus Timing for Normal Space (One Software Wait Cycle, External Wait Cycle Valid (WM Bit = 0), No Idle Cycle)



**Figure 58.13 MPX-I/O Interface Bus Cycle (Three Address Cycles, One Software Wait Cycle, One External Wait Cycle)**



**Figure 58.14 Bus Cycle of SRAM with Byte Selection (SW = 1 Cycle, HW = 1 Cycle, One Asynchronous External Wait Cycle, BAS = 0 (Write Cycle UB/LB Control))**



**Figure 58.15 Bus Cycle of SRAM with Byte Selection (SW = 1 Cycle, HW = 1 Cycle, One Asynchronous External Wait Cycle, BAS = 1 (Write Cycle WE Control))**



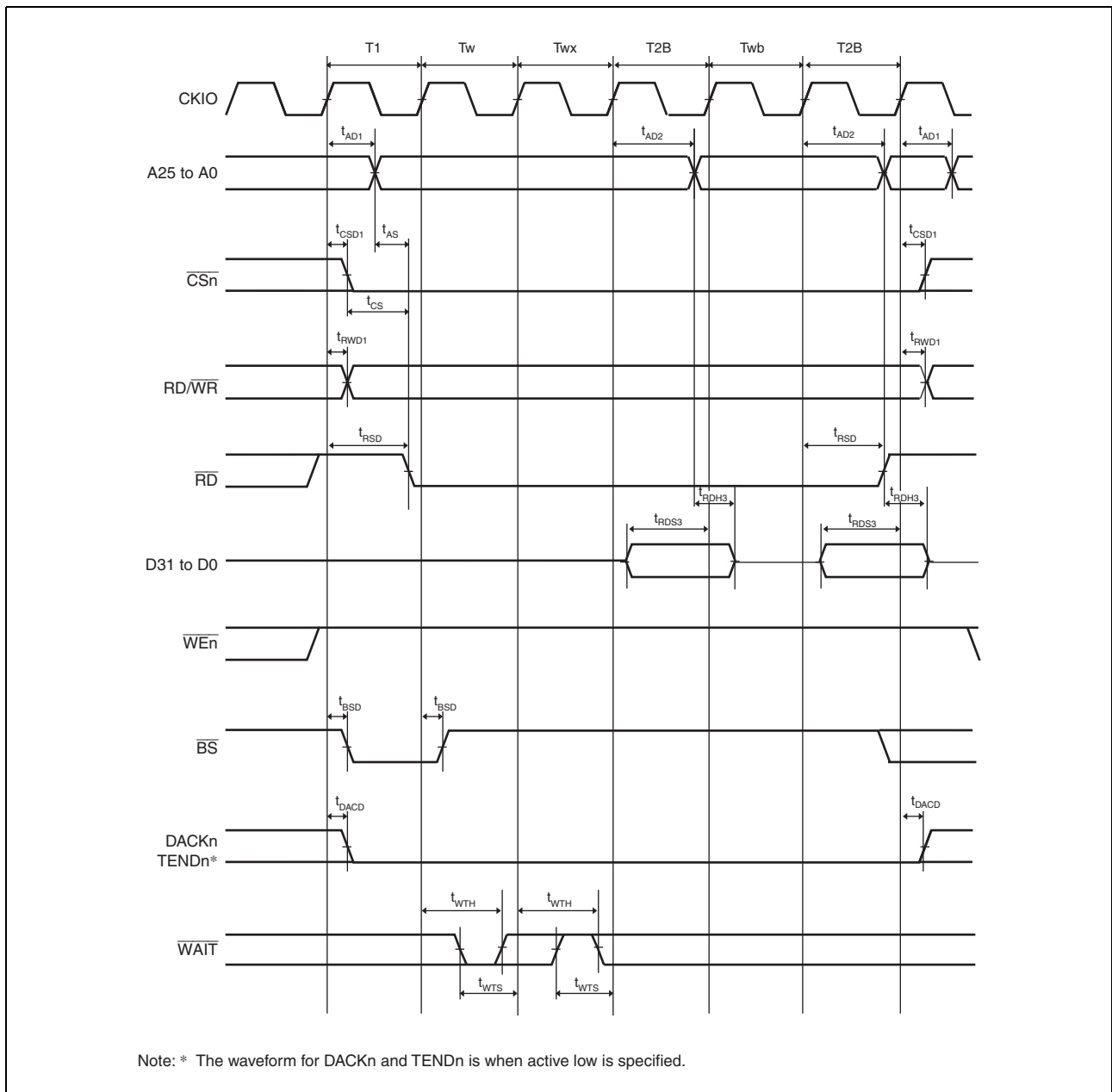
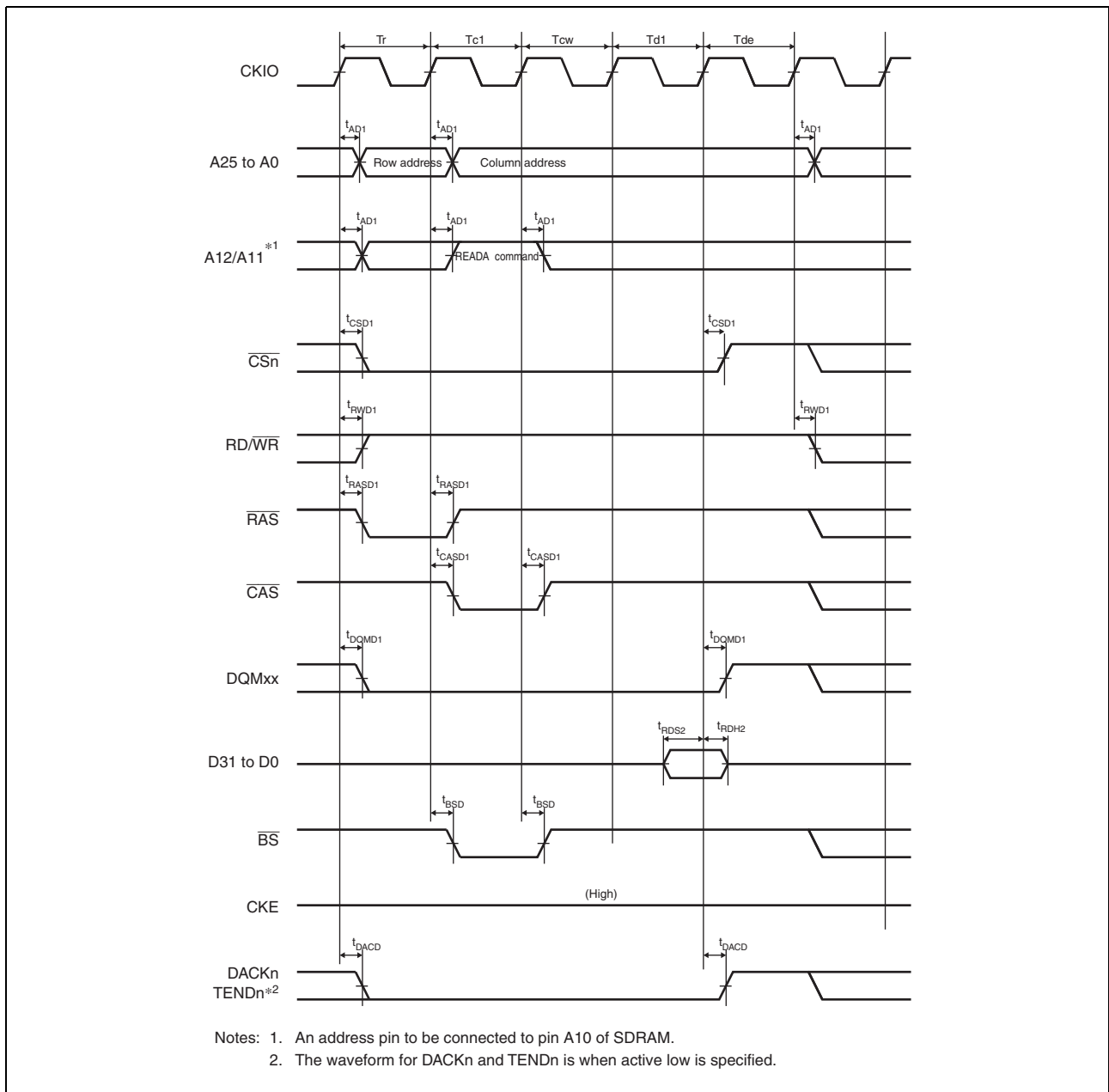
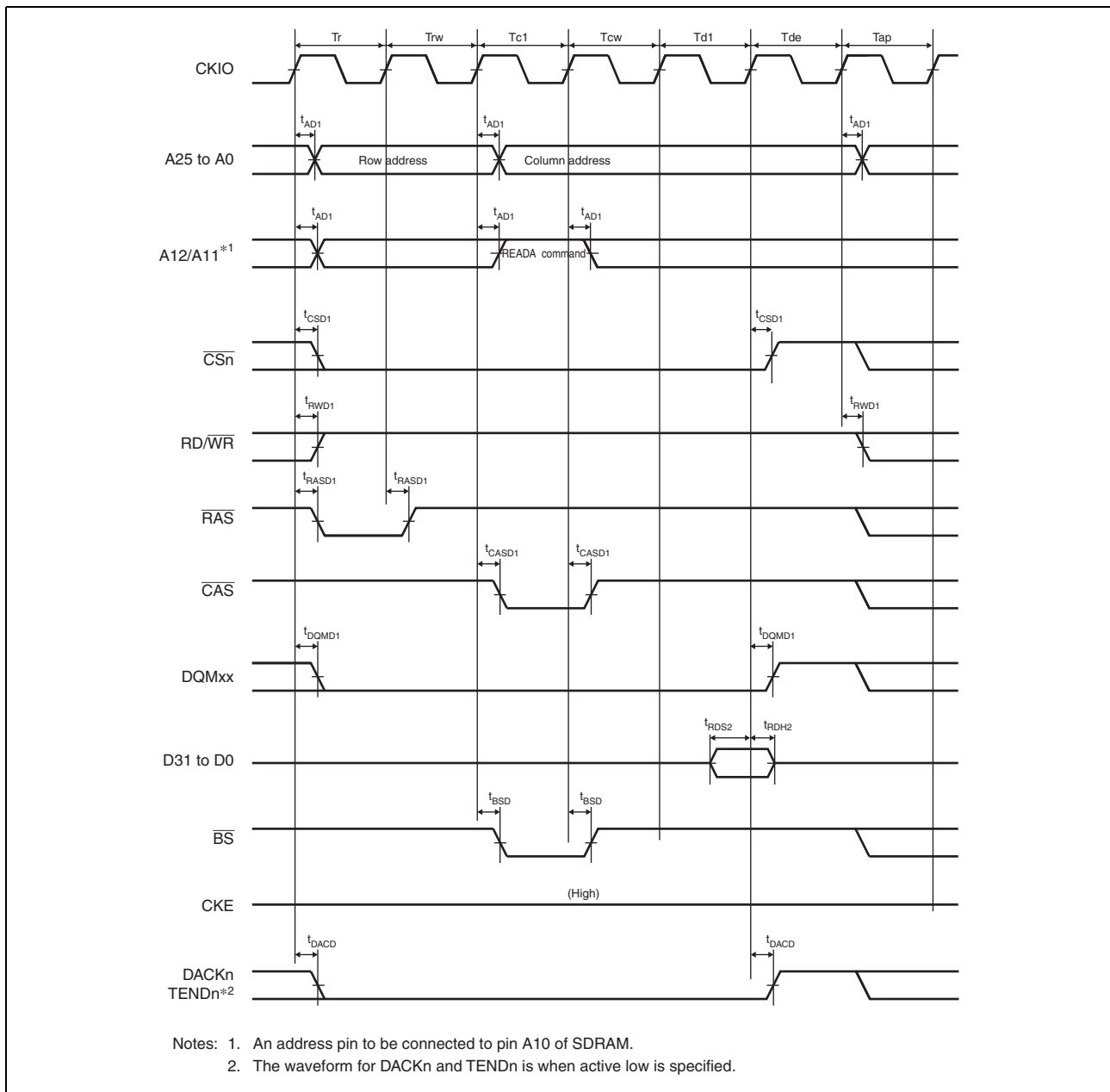


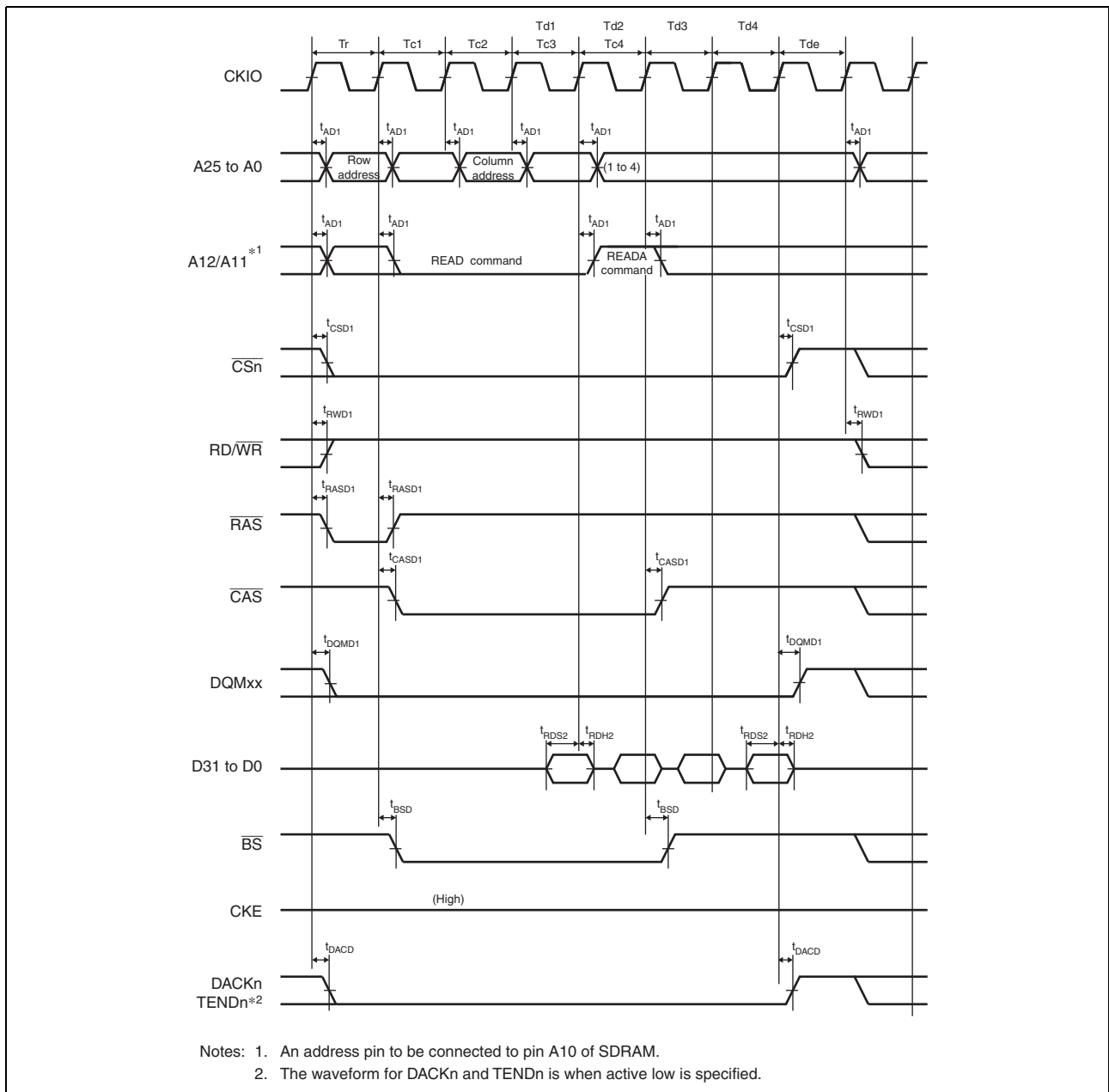
Figure 58.16 Burst ROM Read Cycle (One Software Wait Cycle, One Asynchronous External Burst Wait Cycle, Two Burst)



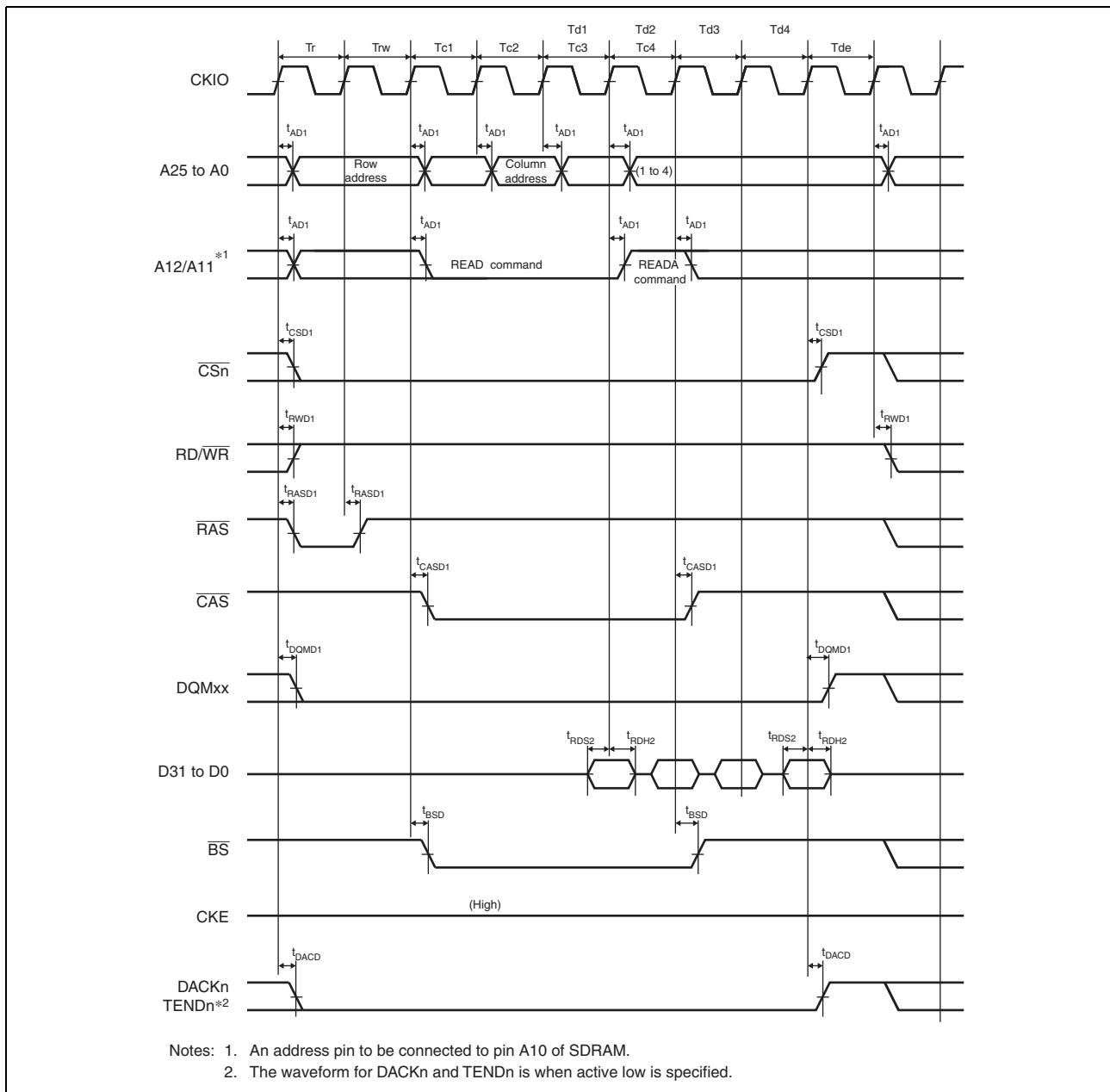
**Figure 58.17 Synchronous DRAM Single Read Bus Cycle (Auto Precharge, CAS Latency 2, WTRCD = 0 Cycle, WTRP = 0 Cycle)**



**Figure 58.18 Synchronous DRAM Single Read Bus Cycle (Auto Precharge, CAS Latency 2, WTRCD = 1 Cycle, WTRP = 1 Cycle)**



**Figure 58.19 Synchronous DRAM Burst Read Bus Cycle (Four Read Cycles) (Auto Precharge, CAS Latency 2, WTRCD = 0 Cycle, WTRP = 1 Cycle)**



**Figure 58.20 Synchronous DRAM Burst Read Bus Cycle (Four Read Cycles) (Auto Precharge, CAS Latency 2, WTRCD = 1 Cycle, WTRP = 0 Cycle)**

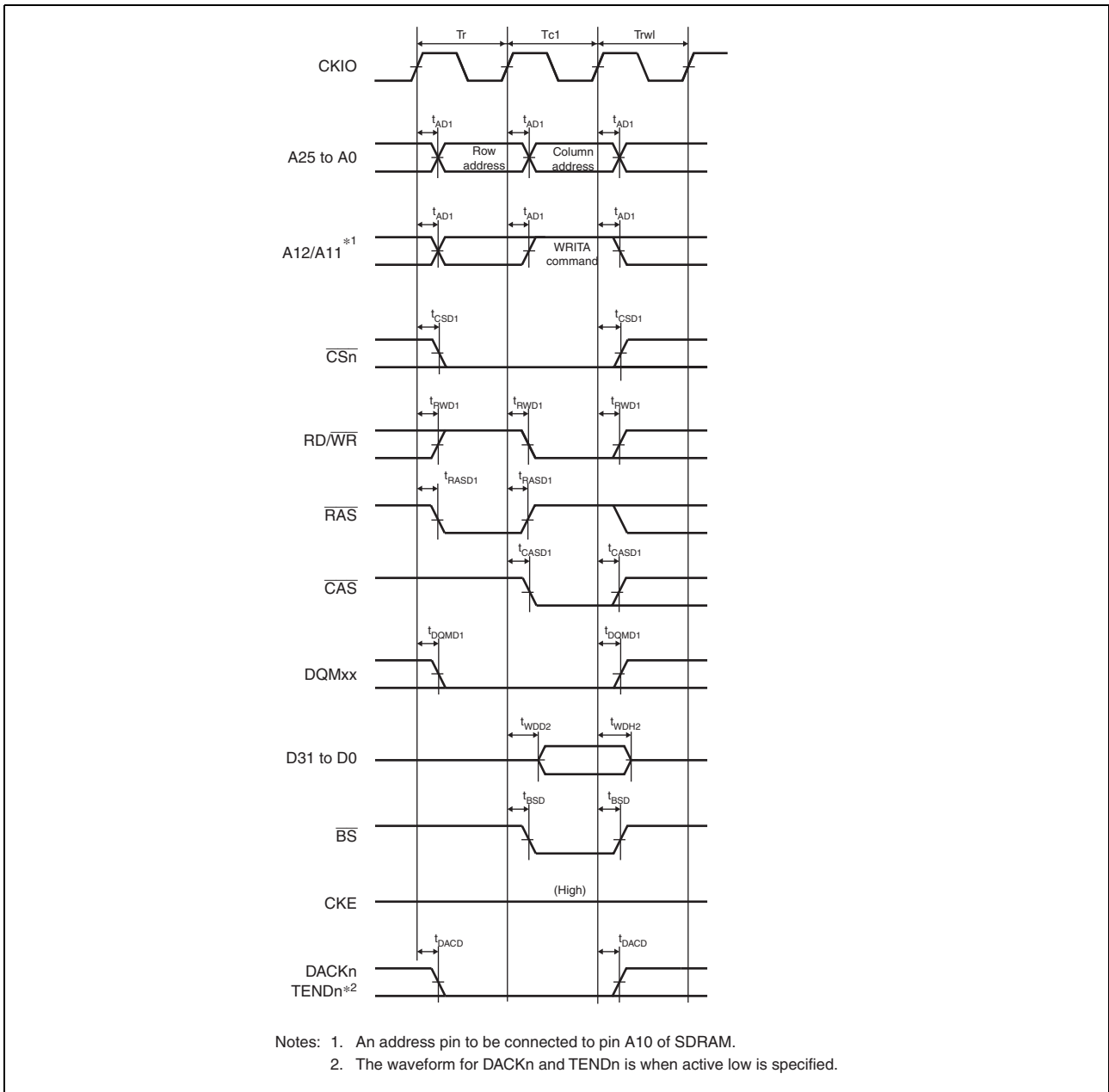
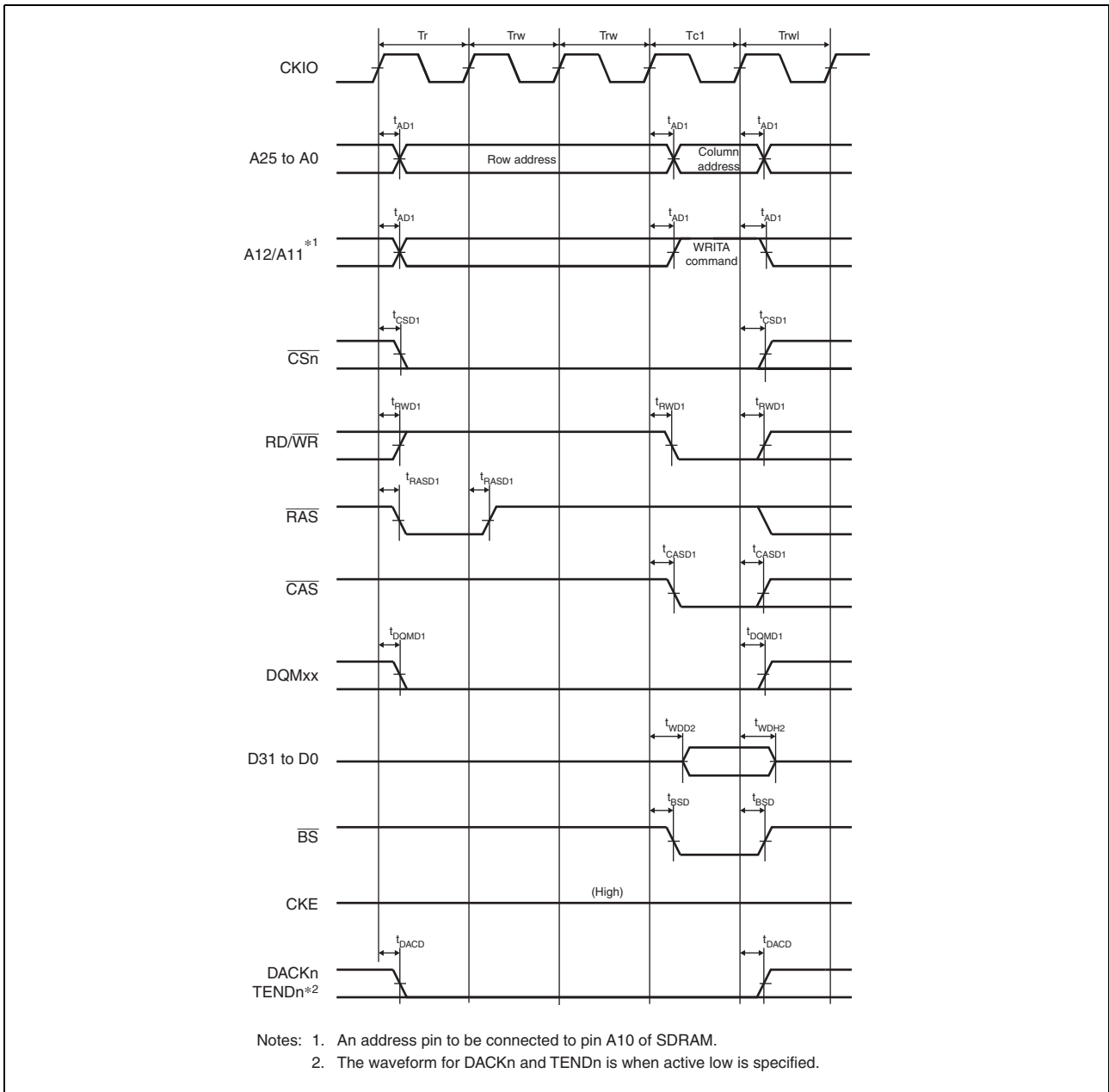
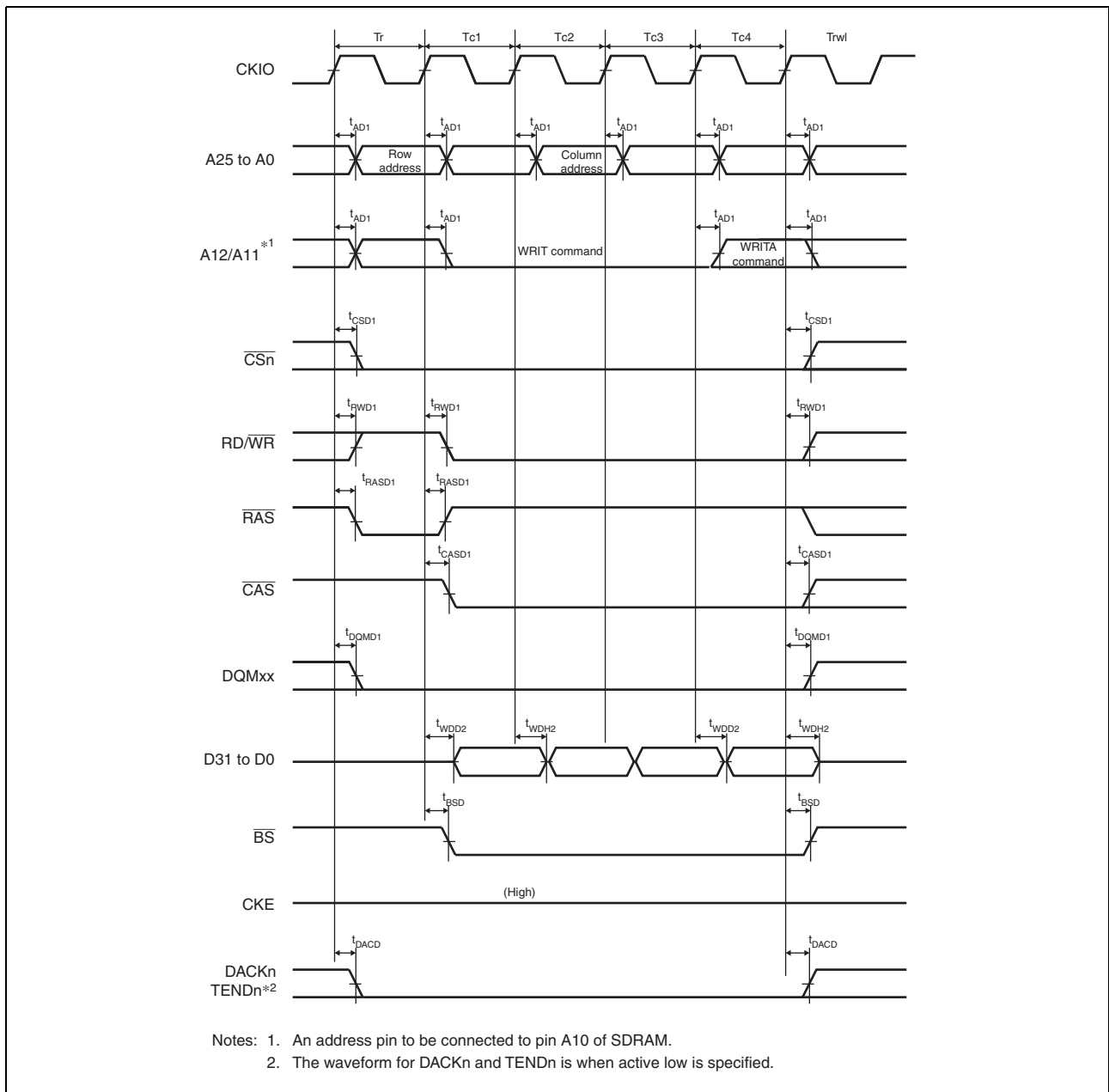


Figure 58.21 Synchronous DRAM Single Write Bus Cycle (Auto Precharge, TRWL = 1 Cycle)

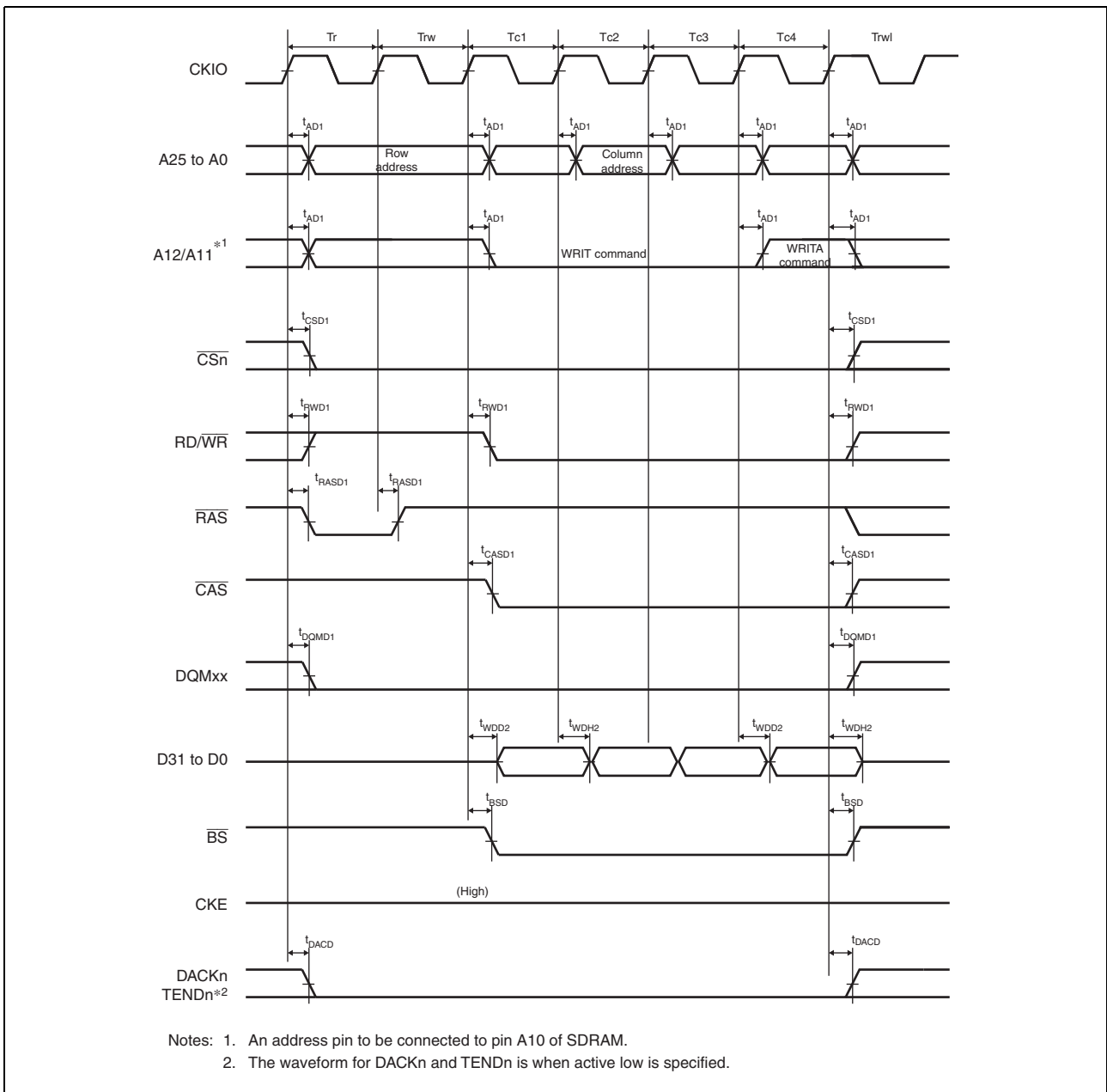


**Figure 58.22 Synchronous DRAM Single Write Bus Cycle (Auto Precharge, WTRCD = 2 Cycles, TRWL = 1 Cycle)**

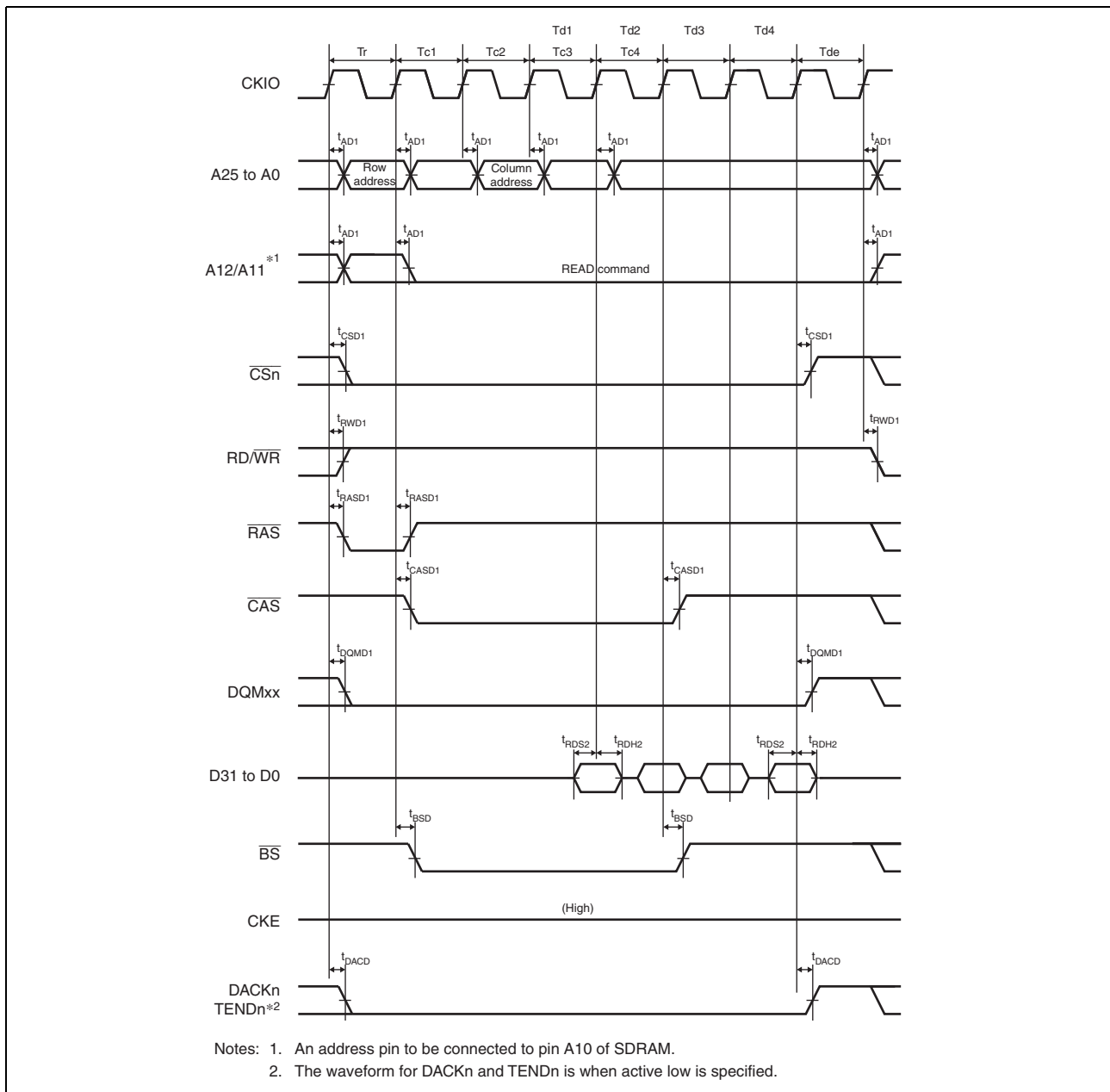


**Figure 58.23 Synchronous DRAM Burst Write Bus Cycle (Four Write Cycles) (Auto Precharge, WTRCD = 0 Cycle, TRWL = 1 Cycle)**

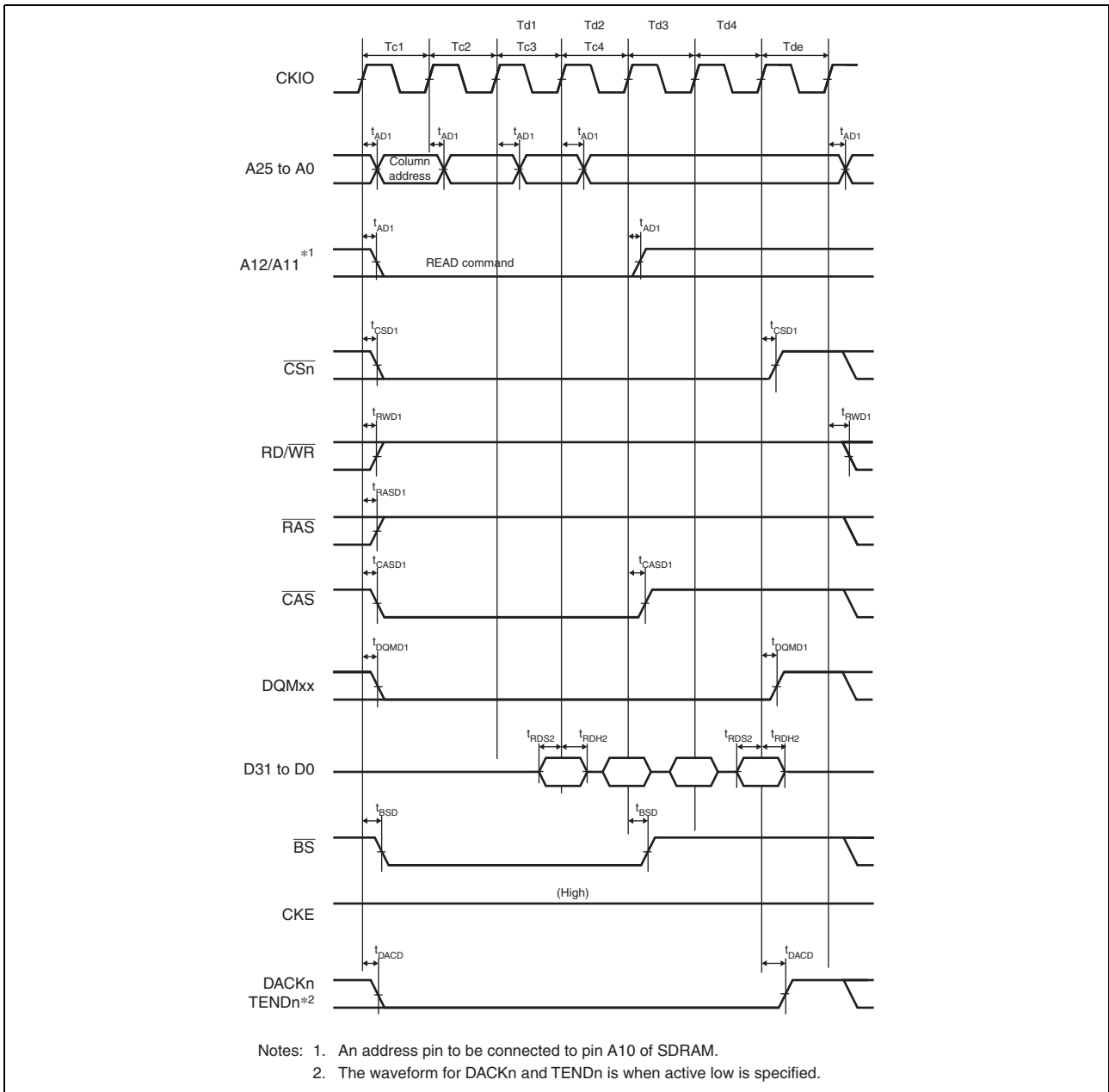




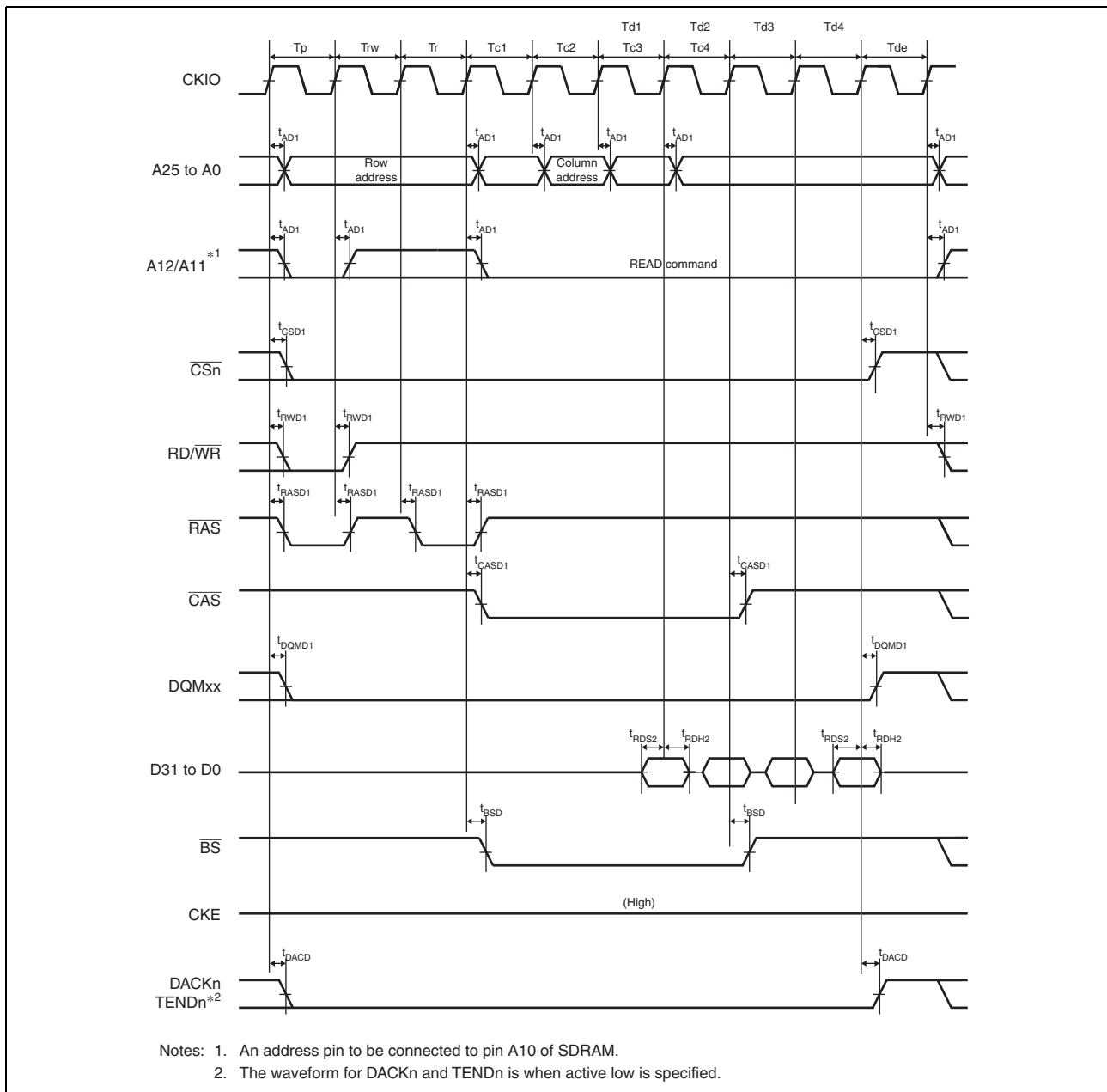
**Figure 58.24 Synchronous DRAM Burst Write Bus Cycle (Four Write Cycles) (Auto Precharge, WTRCD = 1 Cycle, TRWL = 1 Cycle)**



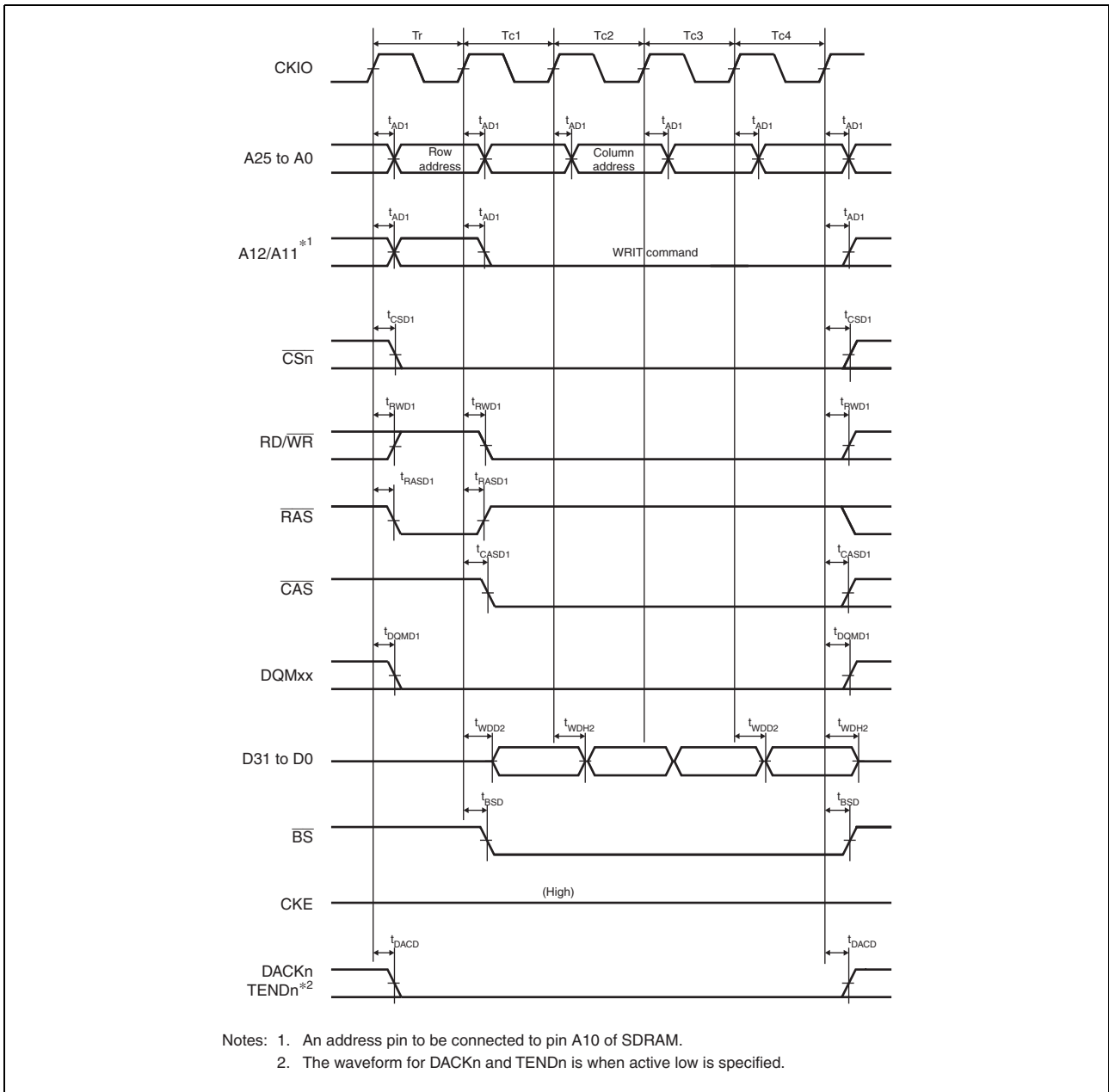
**Figure 58.25 Synchronous DRAM Burst Read Bus Cycle (Four Read Cycles) (Bank Active Mode: ACT + READ Commands, CAS Latency 2, WTRCD = 0 Cycle)**



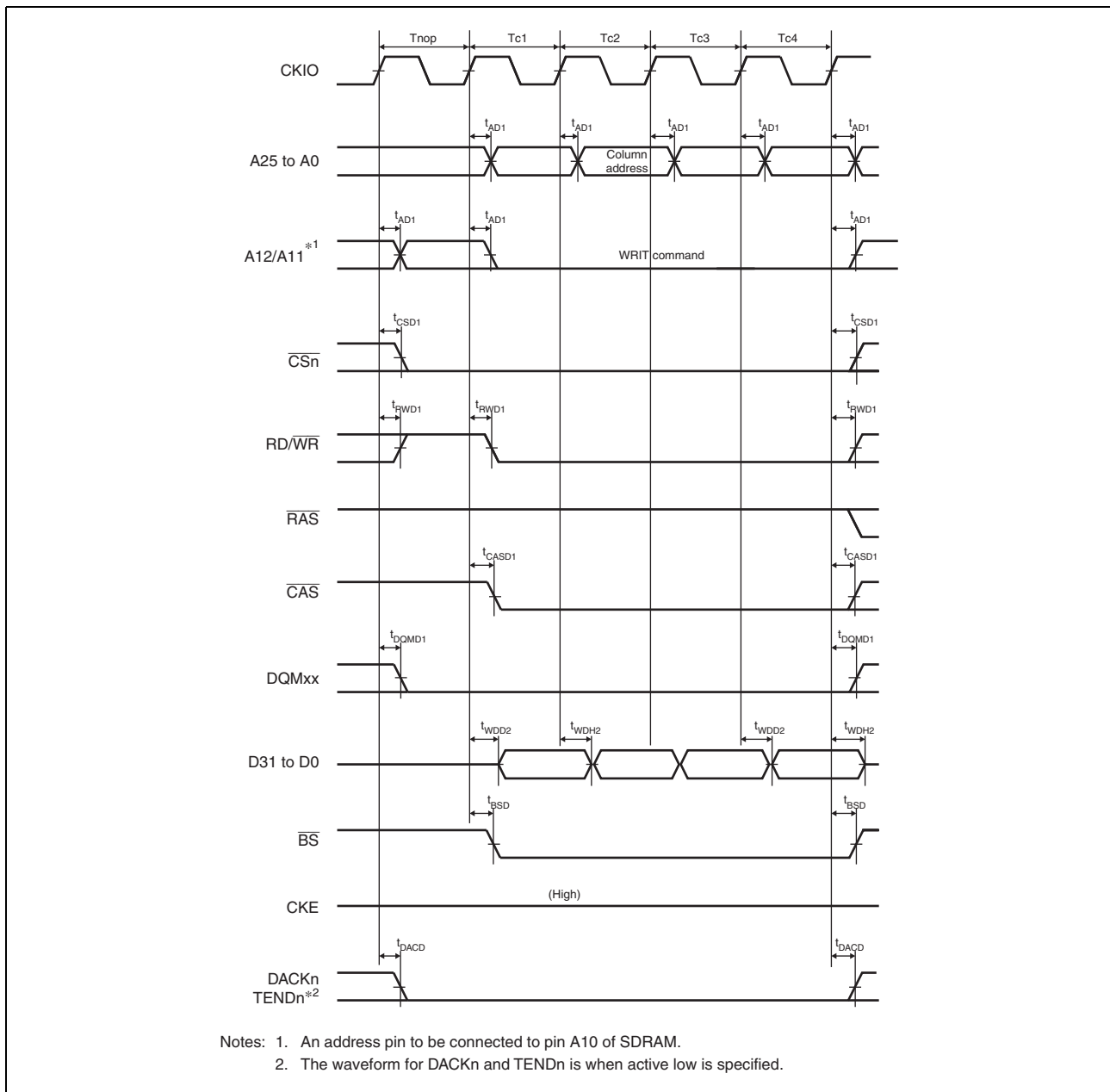
**Figure 58.26 Synchronous DRAM Burst Read Bus Cycle (Four Read Cycles) (Bank Active Mode: READ Command, Same Row Address, CAS Latency 2, WTRCD = 0 Cycle)**



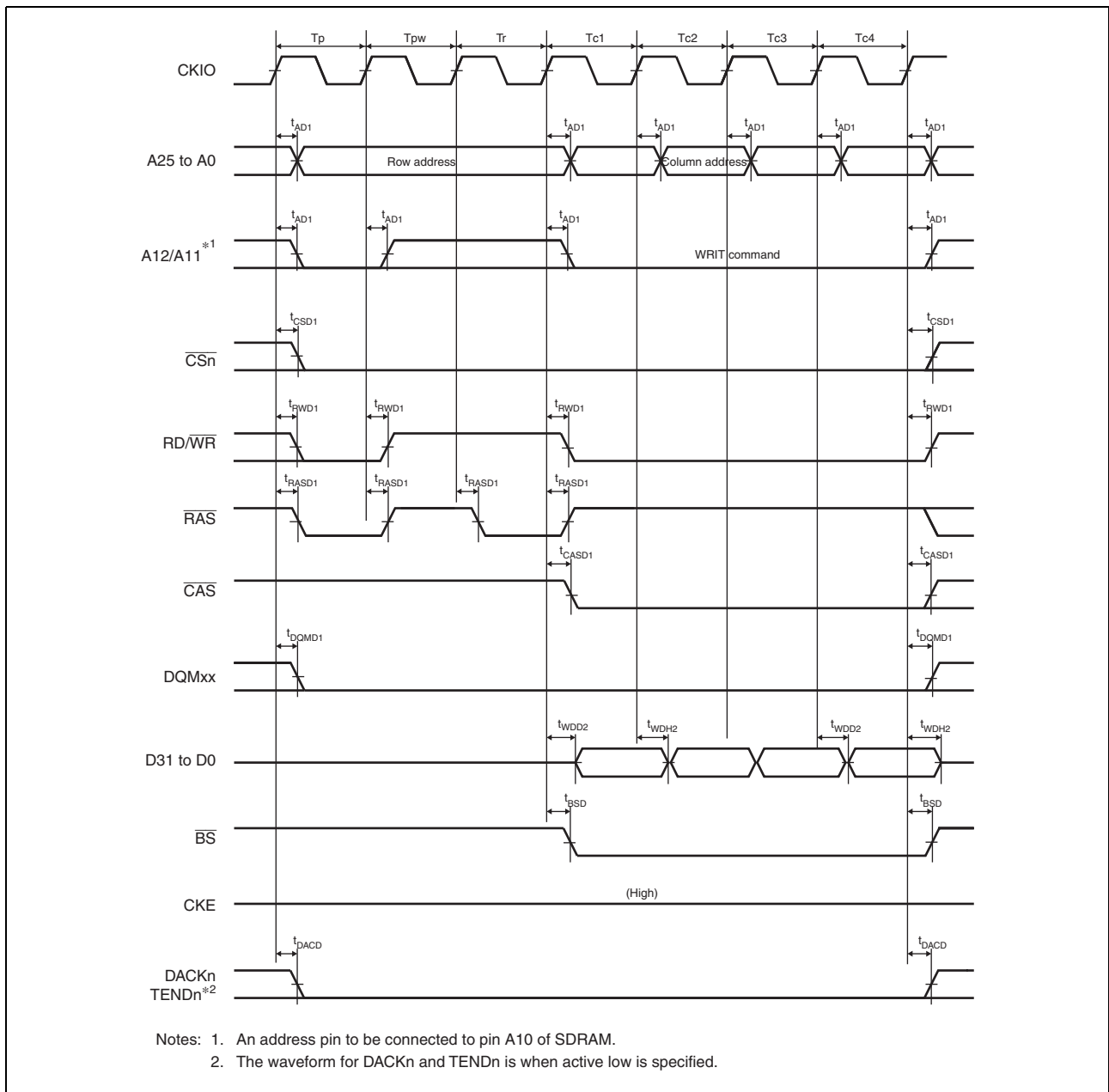
**Figure 58.27 Synchronous DRAM Burst Read Bus Cycle (Four Read Cycles) (Bank Active Mode: PRE + ACT + READ Commands, Different Row Addresses, CAS Latency 2, WTRCD = 0 Cycle)**



**Figure 58.28 Synchronous DRAM Burst Write Bus Cycle (Four Write Cycles) (Bank Active Mode: ACT + WRITE Commands, WTRCD = 0 Cycle, TRWL = 0 Cycle)**



**Figure 58.29 Synchronous DRAM Burst Write Bus Cycle (Four Write Cycles) (Bank Active Mode: WRITE Command, Same Row Address, WTRCD = 0 Cycle, TRWL = 0 Cycle)**



**Figure 58.30 Synchronous DRAM Burst Write Bus Cycle (Four Write Cycles) (Bank Active Mode: PRE + ACT + WRITE Commands, Different Row Addresses, WTRCD = 0 Cycle, TRWL = 0 Cycle)**

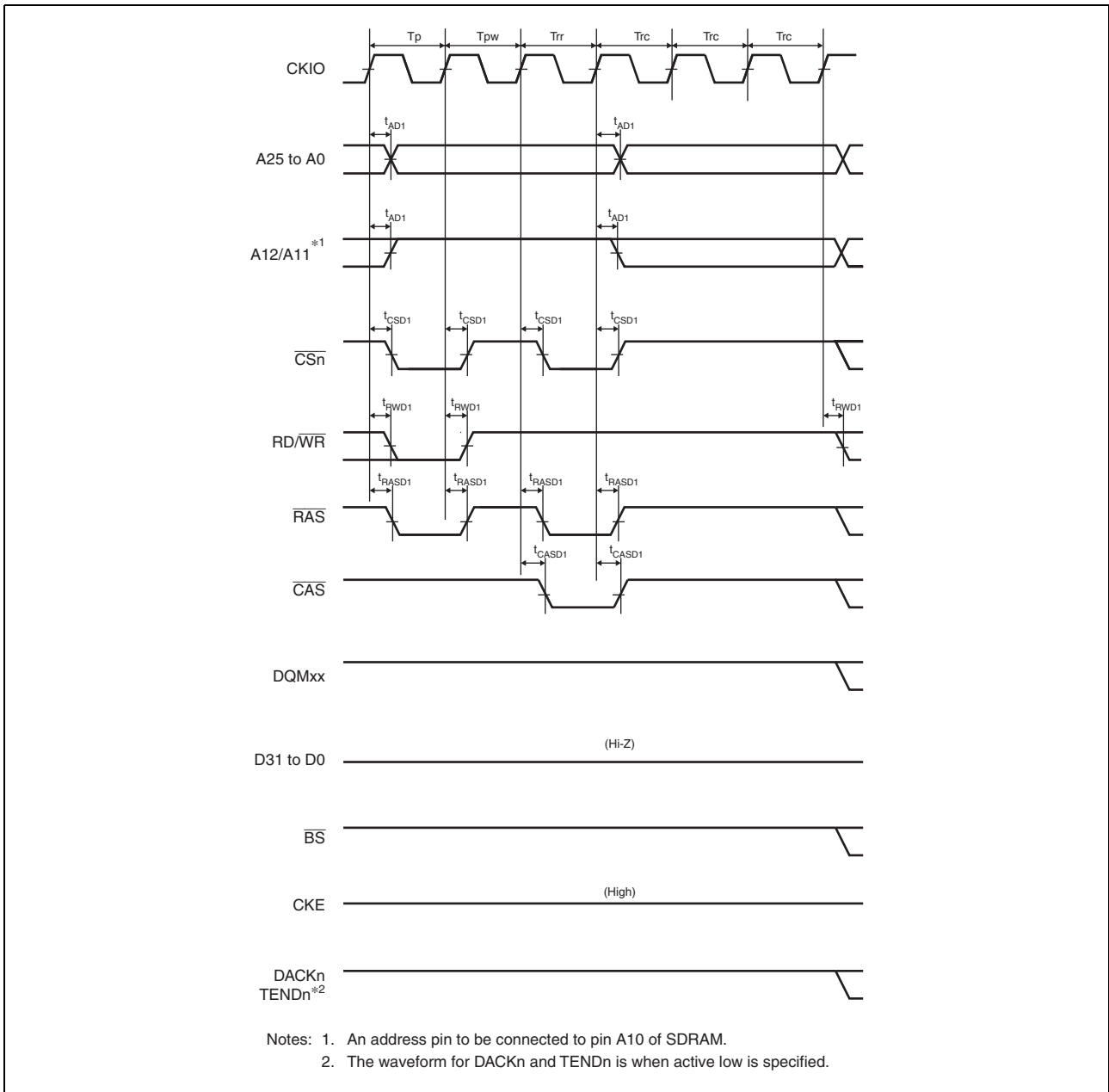


Figure 58.31 Synchronous DRAM Auto-Refreshing Timing (WTRP = 1 Cycle, WTRC = 3 Cycles)



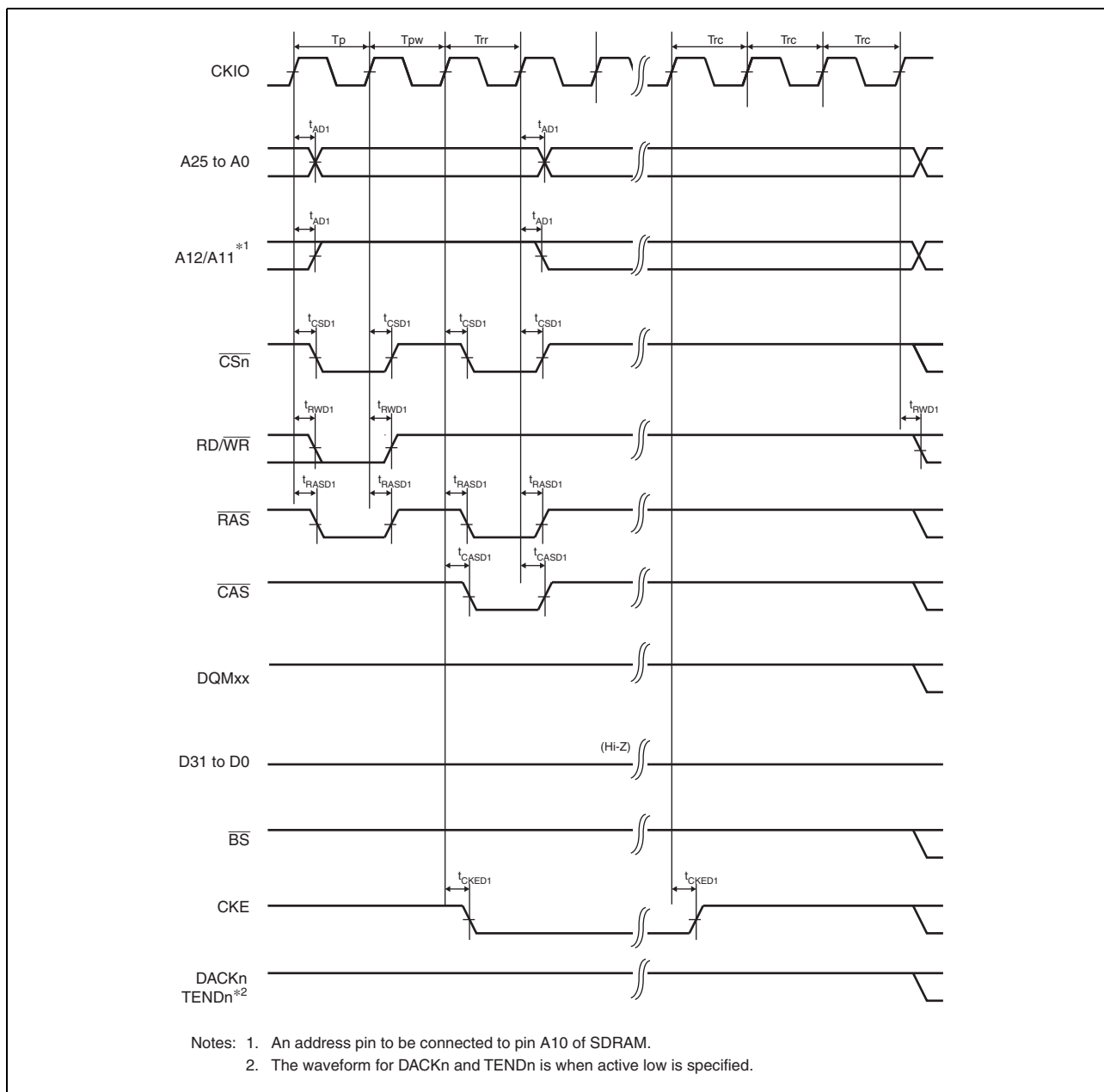


Figure 58.32 Synchronous DRAM Self-Refreshing Timing (WTRP = 1 Cycle)

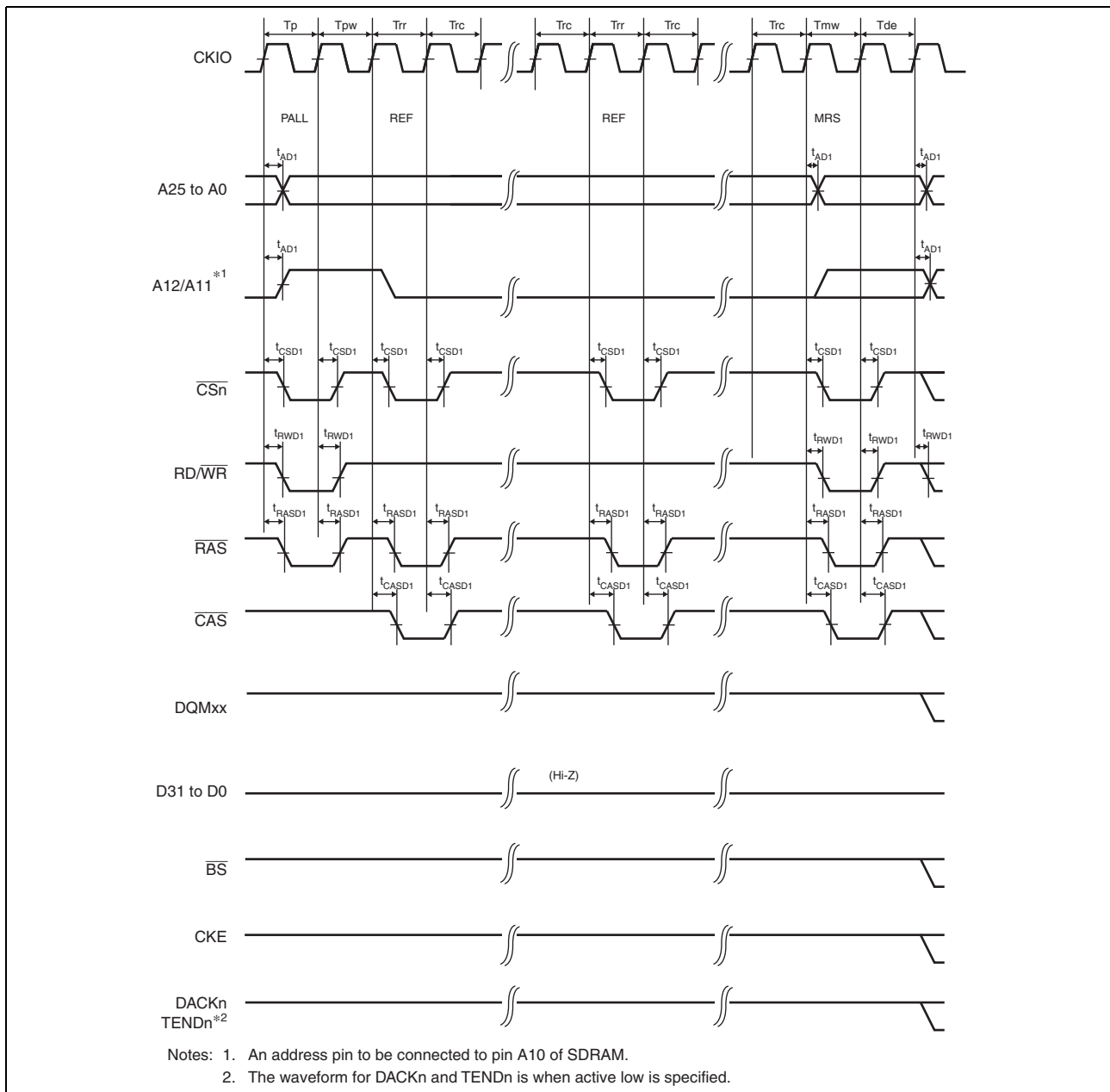
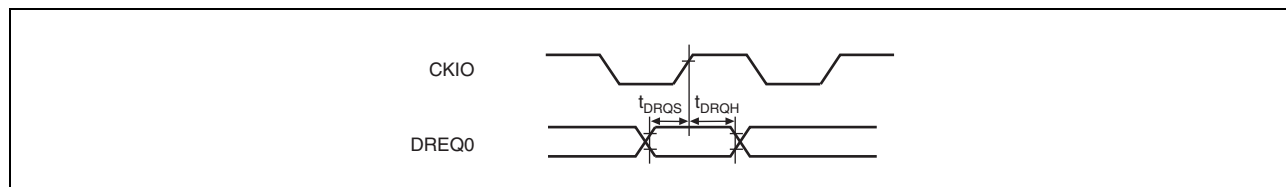


Figure 58.33 Synchronous DRAM Mode Register Write Timing (WTRP = 1 Cycle)

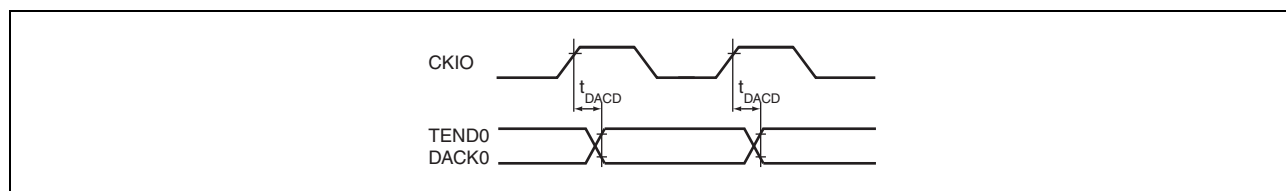
### 58.4.4 Direct Memory Access Controller Timing

**Table 58.8 Direct Memory Access Controller Timing**

Item	Symbol	Min.	Max.	Unit	Figure
DREQ setup time	$t_{DRQS}$	5.5	—	ns	Figure 58.34
DREQ hold time	$t_{DRQH}$	2.5	—		
DACK, TEND delay time	$t_{DACD}$	0	12		Figure 58.35



**Figure 58.34 DREQ Input Timing**



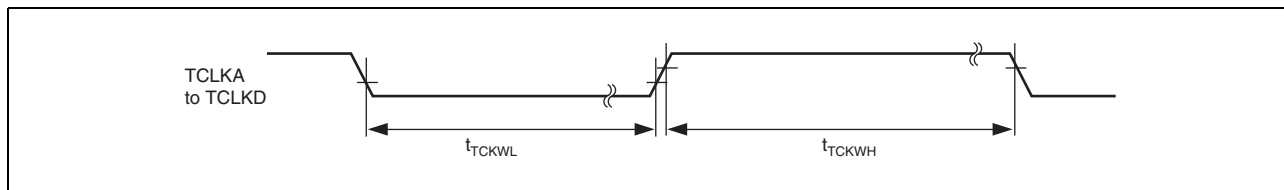
**Figure 58.35 DACK, TEND Output Timing**

### 58.4.5 Multi-Function Timer Pulse Unit 2 Timing

**Table 58.9 Multi-Function Timer Pulse Unit 2 Timing**

Item	Symbol	Min.	Max.	Unit	Figure
Timer clock pulse width (single edge)	$t_{TCKWH/L}$	1.5	—	$t_{p0cyc}$	Figure 58.36
Timer clock pulse width (both edges)	$t_{TCKWH/L}$	2.5	—	$t_{p0cyc}$	
Timer clock pulse width (phase counting mode)	$t_{TCKWH/L}$	2.5	—	$t_{p0cyc}$	

Note:  $t_{p0cyc}$  indicates peripheral clock (P0 $\phi$ ) cycle.

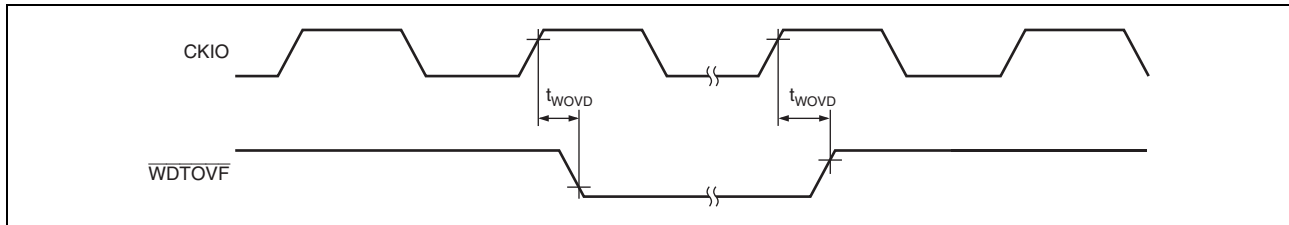


**Figure 58.36 Clock Input Timing**

## 58.4.6 Watchdog Timer Timing

Table 58.10 Watchdog Timer Timing

Item	Symbol	Min.	Max.	Unit	Figure
$\overline{\text{WDTOVF}}$ delay time	$t_{\text{WOVD}}$	—	100	ns	Figure 58.37

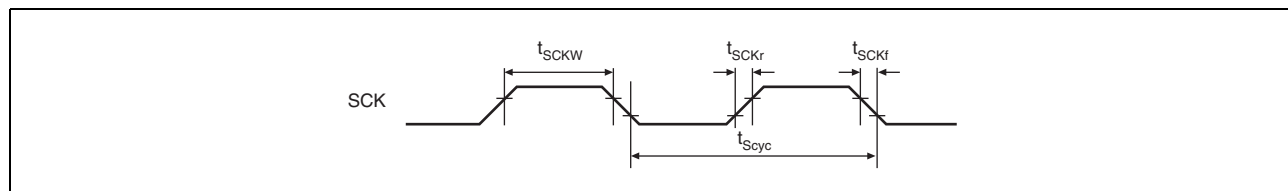
Figure 58.37  $\overline{\text{WDTOVF}}$  Output Timing

### 58.4.7 Serial Communication Interface with FIFO Timing

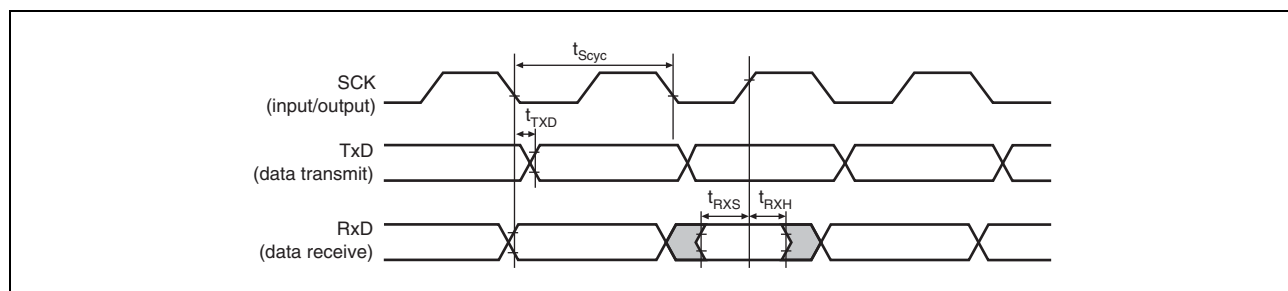
**Table 58.11 Serial Communication Interface with FIFO Timing**

Item	Symbol	Min.	Max.	Unit	Figure	
Input clock cycle	(clocked synchronous)	$t_{Scyc}$	12	—	$t_{p1cyc}$	Figure 58.38
	(asynchronous)		4	—	$t_{p1cyc}$	
Input clock rise time	$t_{SCKr}$	—	1.5	$t_{p1cyc}$		
Input clock fall time	$t_{SCKf}$	—	1.5	$t_{p1cyc}$		
Input clock width	$t_{SCKW}$	0.4	0.6	$t_{Scyc}$		
Transmit data delay time (clocked synchronous)	$t_{TXD}$	—	$3 t_{p1cyc} + 15$	ns	Figure 58.39	
Receive data setup time (clocked synchronous)	$t_{RXS}$	$4 t_{p1cyc} + 15$	—	ns		
Receive data hold time (clocked synchronous)	$t_{RXH}$	$1 t_{p1cyc} + 15$	—	ns		

Note:  $t_{p1cyc}$  indicates the peripheral clock 1 (P1 $\phi$ ) cycle.



**Figure 58.38 SCK Input Clock Timing**



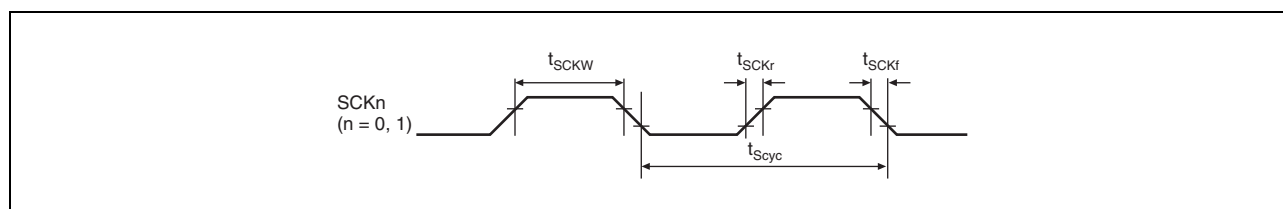
**Figure 58.39 Transmit/Receive Data Input/Output Timing in Clocked Synchronous Mode**

### 58.4.8 Serial Communication Interface Timing

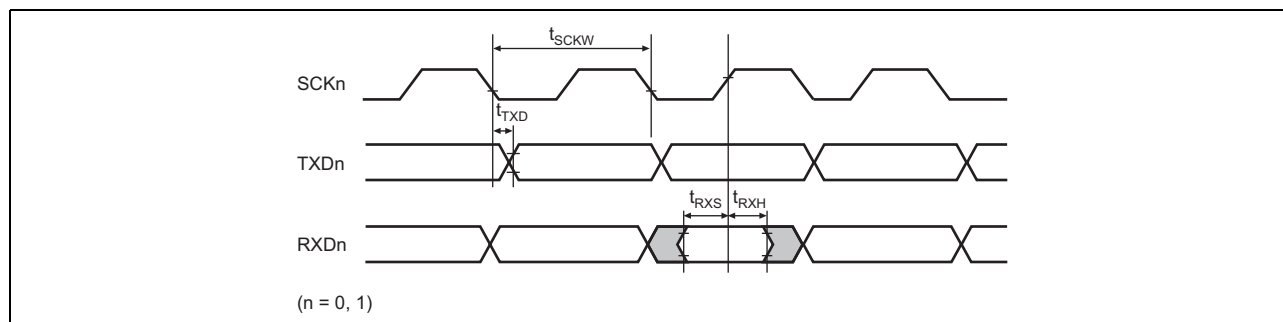
**Table 58.12 Serial Communication Interface Timing**

Item	Symbol	Min.	Max.	Unit	Figure	
Input clock cycle	(asynchronous)	$t_{Sycyc}$	4	—	$t_{P1cyc}$	Figure 58.40
	(clocked synchronous)	6	—			
Input clock pulse width	$t_{SCKW}$	0.4	0.6	$t_{Sycyc}$		
Input clock rise time	$t_{SCKr}$	—	20	ns		
Input clock fall time	$t_{SCKf}$	—	20	ns		
Output clock cycle	(asynchronous)	$t_{Sycyc}$	16	—	$t_{P1cyc}$	
	(clocked synchronous)	4	—			
Output clock pulse width	$t_{SCKW}$	0.4	0.6	$t_{Sycyc}$		
Output clock rise time	$t_{SCKr}$	—	20	ns		
Output clock fall time	$t_{SCKf}$	—	20	ns		
Transmit data delay time	(clocked synchronous) $t_{TXD}$	—	40	ns	Figure 58.41	
Receive data setup time	(clocked synchronous) $t_{RXS}$	40	—	ns		
Receive data hold time	(clocked synchronous) $t_{RXH}$	40	—	ns		

Note:  $t_{P1cyc}$  indicates the peripheral clock 1 (P1 $\phi$ ) cycle.



**Figure 58.40 SCK Input Clock Timing**



**Figure 58.41 Transmit/Receive Data Input/Output Timing in Clocked Synchronous Mode**

## 58.4.9 Renesas Serial Peripheral Interface Timing

Table 58.13 Renesas Serial Peripheral Interface Timing

Item		Symbol	Min.	Max.	Unit	Figure
RSPCK clock cycle	Master	$t_{SPcyc}$	2	4096	$t_{cyc}$	Figure 58.42
	Slave		8	4096		
RSPCK clock high pulse width	Master	$t_{SPCKWH}$	0.4	—	$t_{SPcyc}$	
	Slave		0.4	—		
RSPCK clock low pulse width	Master	$t_{SPCKWL}$	0.4	—	$t_{SPcyc}$	
	Slave		0.4	—		
Data input setup time	Master	$t_{su}$	15	—	ns	Figure 58.43 to Figure 58.46
	Slave		0	—	$t_{cyc}$	
Data input hold time	Master	$t_{H}$	0	—	ns	
	Slave		4	—	$t_{cyc}$	
SSL setup time	Master	$t_{LEAD}$	$1 \times t_{SPcyc} - 20$	$8 \times t_{SPcyc}$	ns	
	Slave		4	—	$t_{cyc}$	
SSL hold time	Master	$t_{LAG}$	$1 \times t_{SPcyc}$	$8 \times t_{SPcyc} + 20$	ns	
	Slave		4	—	$t_{cyc}$	
Data output delay time	Master	$t_{OD}$	—	21	ns	
	Slave		—	4	$t_{cyc}$	
Data output hold time	Master	$t_{OH}$	5	—	ns	
	Slave		3	—	$t_{cyc}$	
Continuous transmission delay time	Master	$t_{TD}$	$1 \times t_{SPcyc} + 2 t_{cyc}$	$8 \times t_{SPcyc} + 2 \times t_{cyc}$	ns	
	Slave		$4 \times t_{cyc}$	—		
Slave access time		$t_{SA}$	—	4	$t_{cyc}$	Figure 58.45 and Figure 58.46
Slave out release time		$t_{REL}$	—	3	$t_{cyc}$	

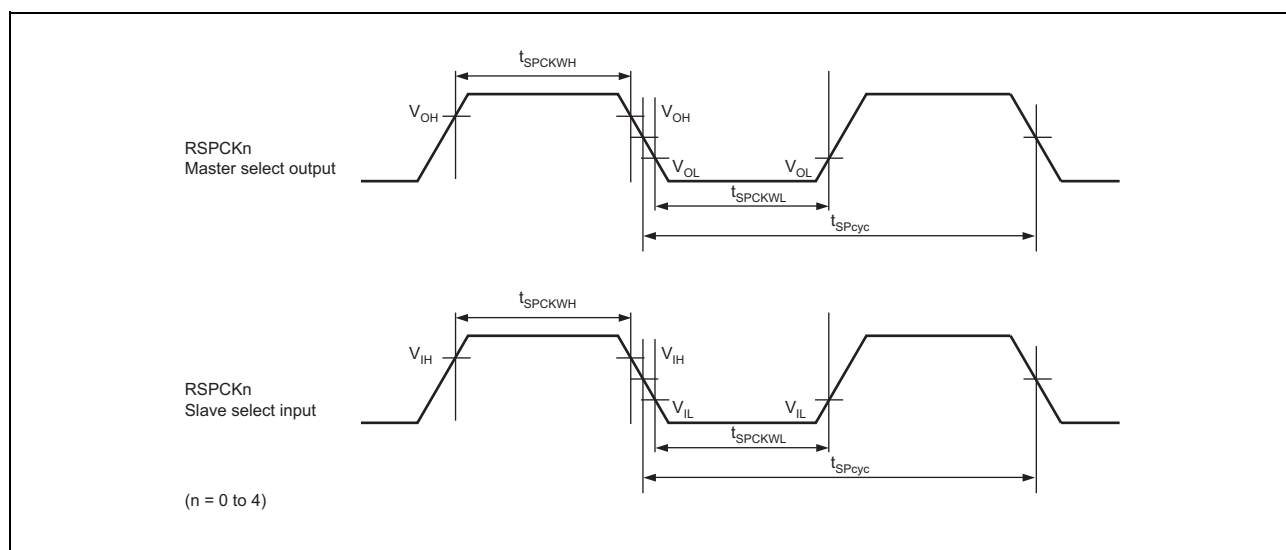


Figure 58.42 Clock Timing



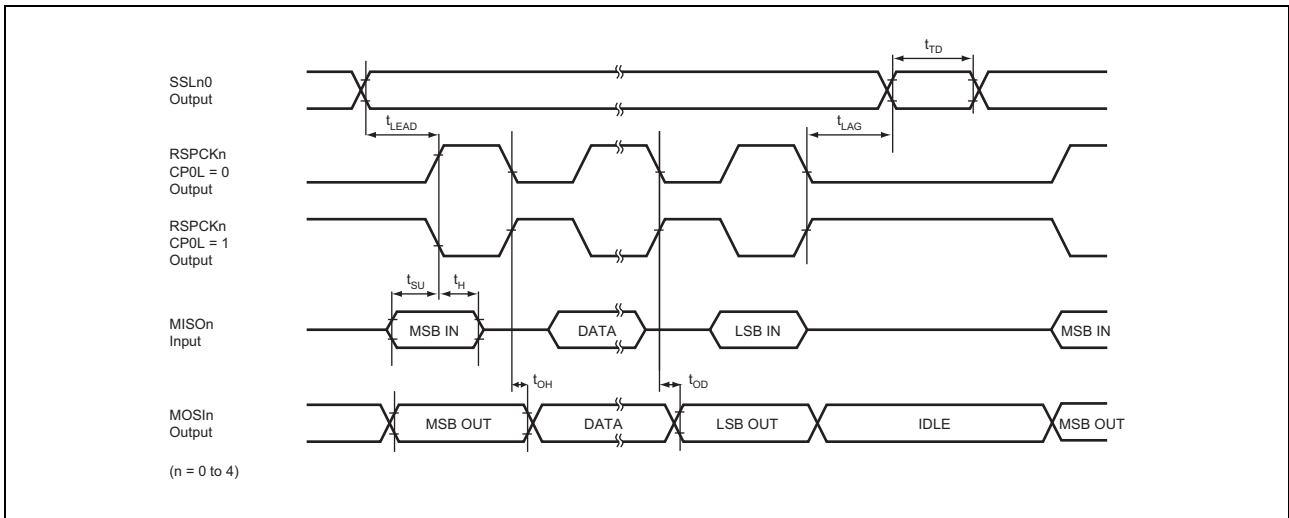


Figure 58.43 Transmission and Reception Timing (Master, CPHA = 0)

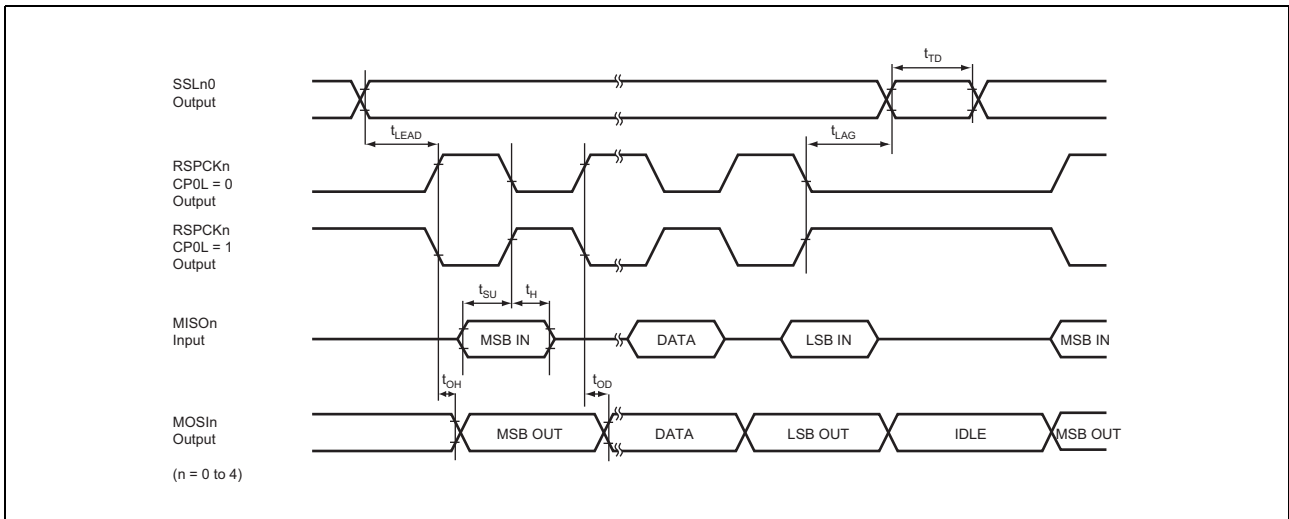


Figure 58.44 Transmission and Reception Timing (Master, CPHA = 1)

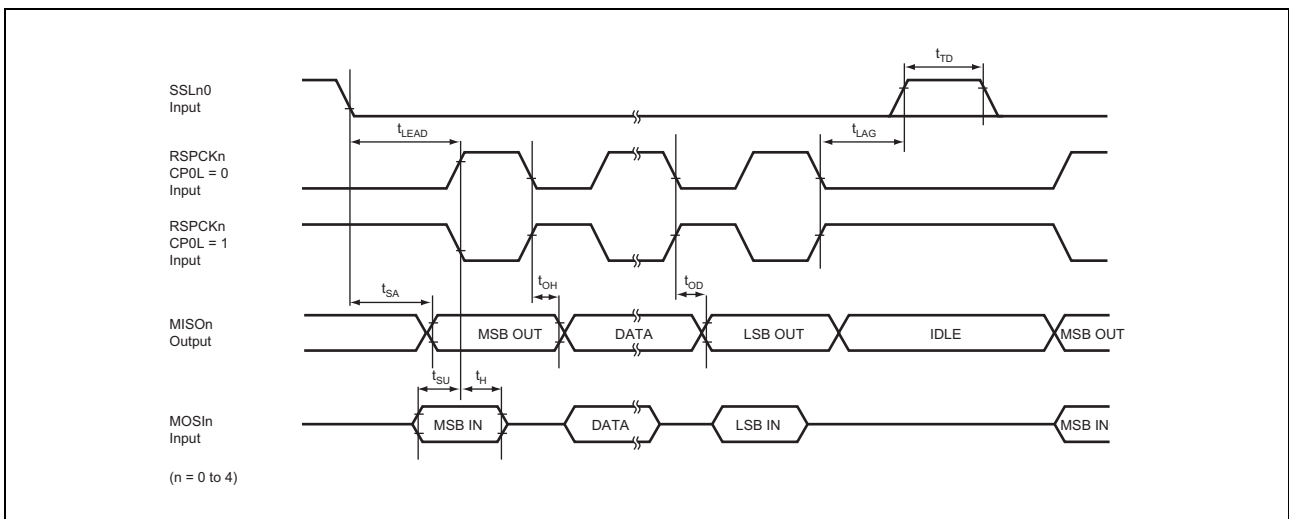


Figure 58.45 Transmission and Reception Timing (Slave, CPHA = 0)

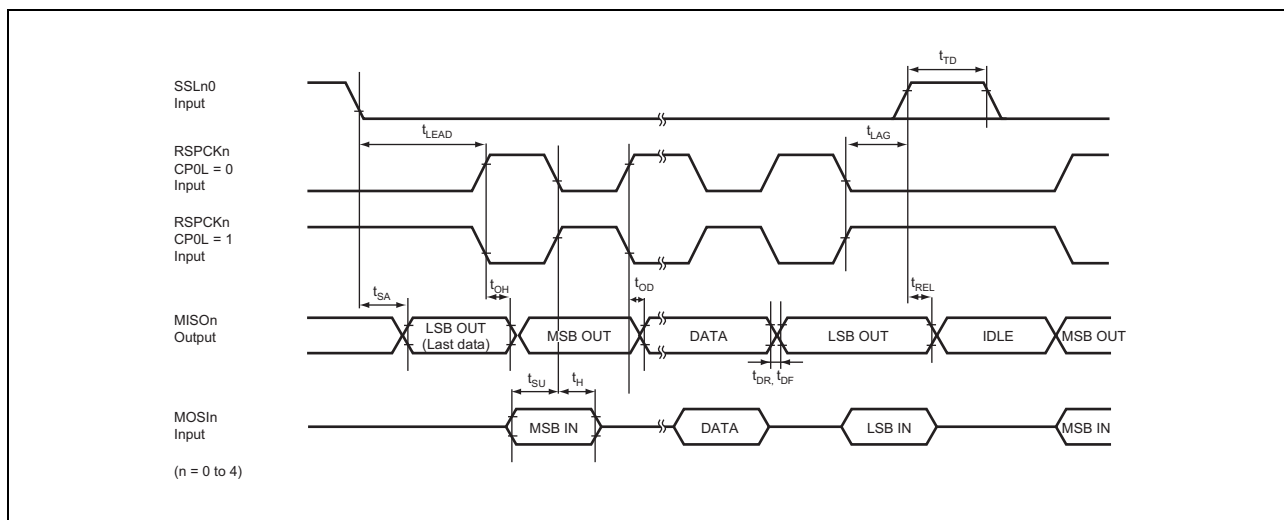


Figure 58.46 Transmission and Reception Timing (Slave, CPHA = 1)

## 58.4.10 SPI Multi I/O Bus Controller Timing

Table 58.14 SPI Multi I/O Bus Controller Timing

Item	Symbol	Min.	Max.	Unit	Figure	
SPBCLK clock cycle	$t_{SPBcyc}$	2	4080	$t_{cyc}$	Figure 58.47	
SPBCLK high pulse width	$t_{SPBWH}$	0.475	0.525	$t_{SPBcyc}$		
SPBCLK low pulse width	$t_{SPBWL}$	0.475	0.525	$t_{SPBcyc}$		
SPBCLK rise time	$t_{SPBR}$	—	3	ns		
SPBCLK fall time	$t_{SPBF}$	—	3	ns		
Data input setup time	CKDLY = B'0100 (initial value)	$t_{SU}$	10.0	—	ns	Figure 58.48 and Figure 58.49
	CKDLY = B'1000		4.2	—	ns	
Data input hold time	Common to CKDLY = B'0100 and B'1000	$t_H$	0.0	—	ns	
SSL setup time		$t_{LEAD}$	$1 \times t_{SPBcyc} - 3$	$8 \times t_{SPBcyc} + 3$	ns	
SSL hold time		$t_{LAG}$	$1.5 \times t_{SPBcyc} - 3$	$8.5 \times t_{SPBcyc} + 3$	ns	
Continuous transfer delay time		$t_{TD}$	1	8	$t_{SPBcyc}$	
Data output delay time	SPODLY = H'0000 (initial value)	$t_{OD}$	—	3.6	ns	
	SPODLY = H'6363		—	7.0		
Data output hold time	SPODLY = H'0000 (initial value)	$t_{OH}$	-1.6	—	ns	
	SPODLY = H'6363		1.0	—		
Data output buffer on time	SPODLY = H'0000 (initial value)	$t_{BON}$	—	3.6	ns	Figure 58.50, Figure 58.51, Figure 58.52, and Figure 58.53
	SPODLY = H'6363		—	7.0		
Data output buffer off time	SPODLY = H'0000 (initial value)	$t_{BOFF}$	-7.0	0	ns	
	SPODLY = H'6363		1.0	7.0		

Note:  $t_{cyc}$  indicates the bus clock ( $B\phi$ ) cycle.

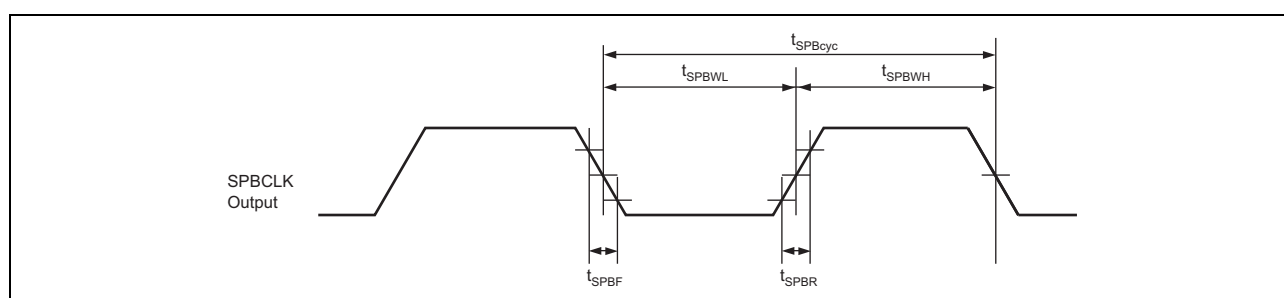


Figure 58.47 Clock Timing

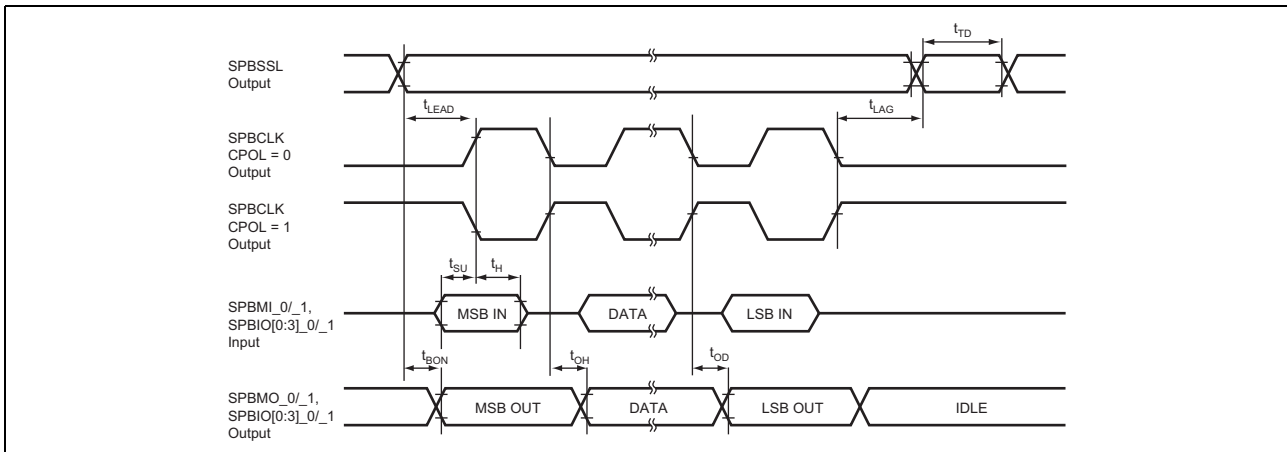


Figure 58.48 SDR Transfer Format Transmission and Reception Timing (CPHAT = 0, CPHAR = 0)

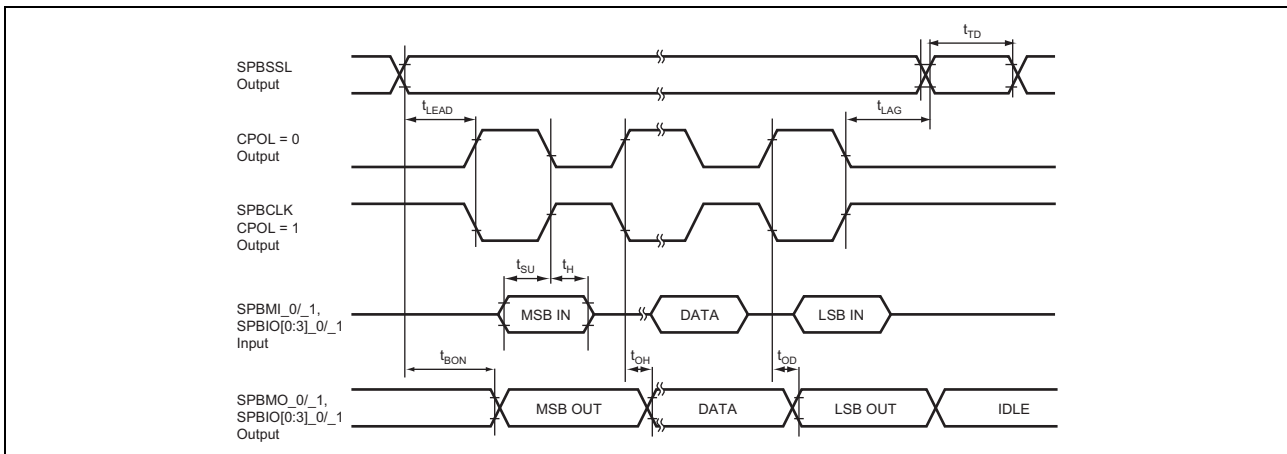


Figure 58.49 SDR Transfer Format Transmission and Reception Timing (CPHAT = 1, CPHAR = 1)

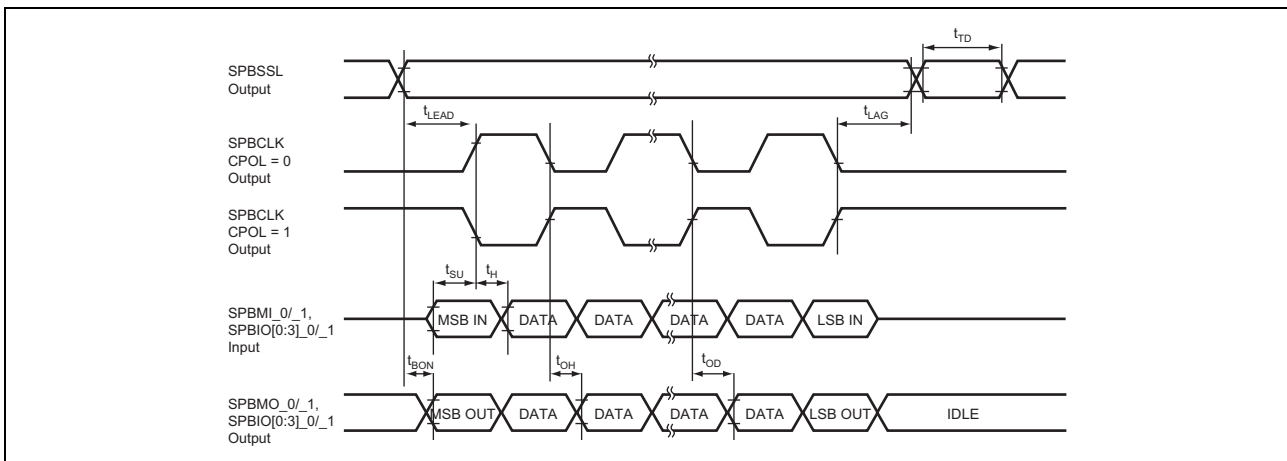


Figure 58.50 DDR Transfer Format Transmission and Reception Timing (CPHAT = 0, CPHAR = 0)

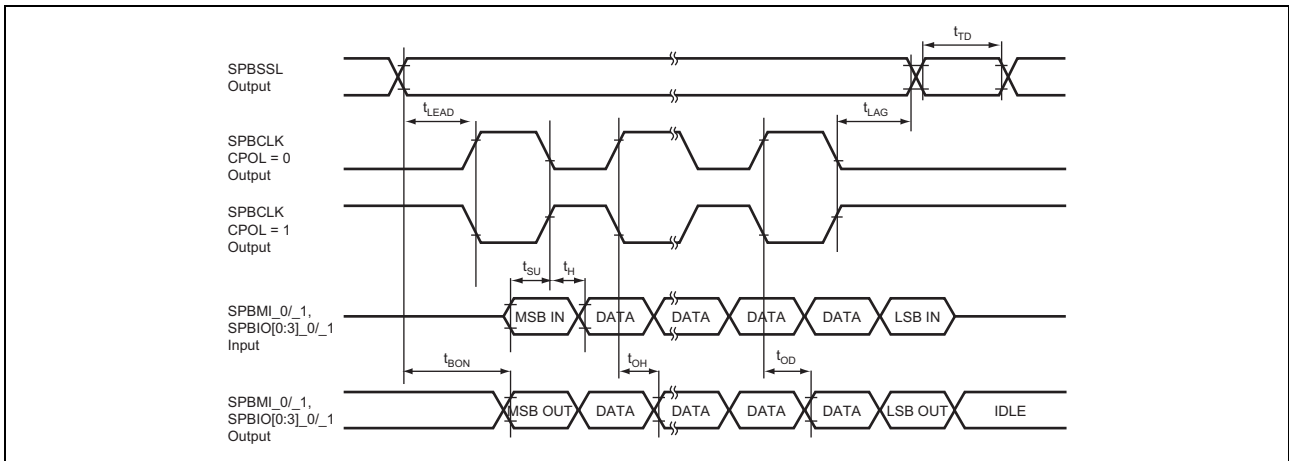


Figure 58.51 DDR Transfer Format Transmission and Reception Timing (CPHAT = 1, CPHAR = 1)

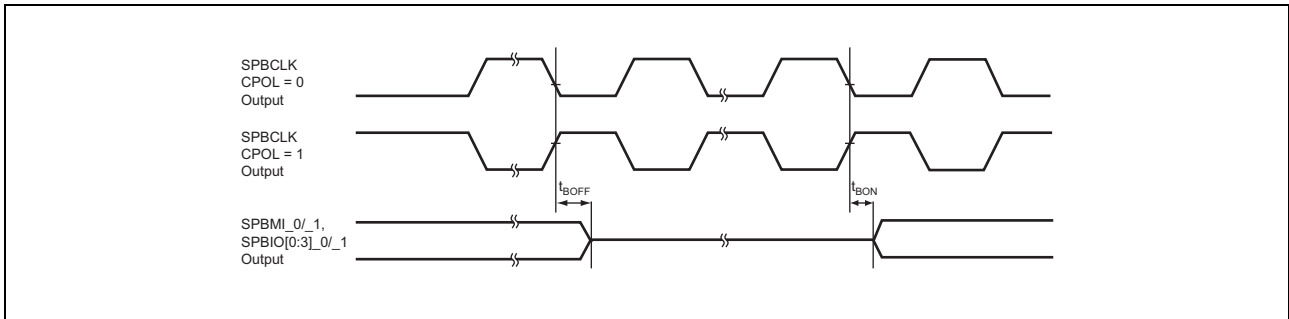


Figure 58.52 Timing for Switching the Buffers on and off (CPHAT = 0, CPHAR = 0)

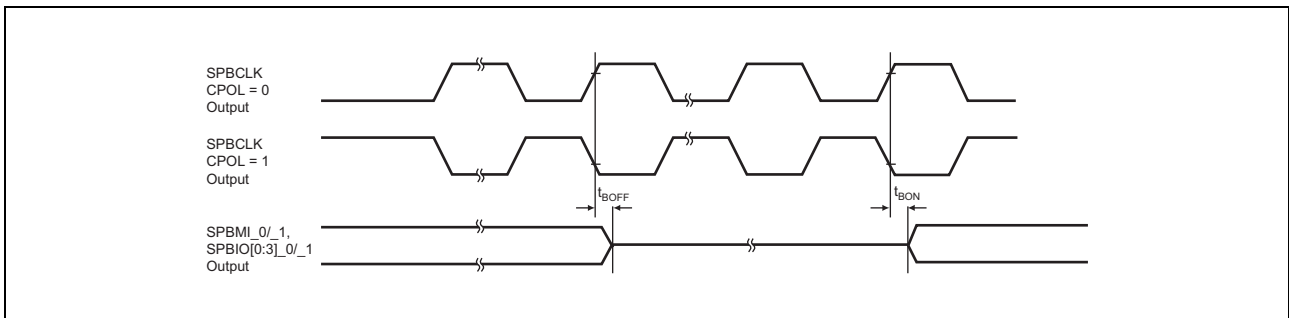


Figure 58.53 Timing for Switching the Buffers on and off (CPHAT = 1, CPHAR = 1)

58.4.11 I<sup>2</sup>C Bus Interface TimingTable 58.15 I<sup>2</sup>C Bus Interface Timing

Item	Symbol	I/O	Standard mode (Sm)		Fast mode (Fm)		Unit
			Min.	Max.	Min.	Max.	
SCL clock frequency	f <sub>CLK</sub>	I/O	0	100	0	400	kHz
Bus free time (between stop and start condition)	t <sub>BUF</sub>	I/O	4.7	-	1.3	-	μs
Hold time*1	t <sub>HD:STA</sub>	I/O	4.0	-	0.6	-	μs
Low period of SCL clock	t <sub>LOW</sub>	I/O	4.7	-	1.3	-	μs
High period of SCL clock	t <sub>HIGH</sub>	I/O	4.0	-	0.6	-	μs
Setup time for start/restart condition	t <sub>SU:STA</sub>	I/O	4.7	-	0.6	-	μs
Data hold time (I <sup>2</sup> C bus device)	t <sub>HD:DAT</sub>	I/O	0*2	-	0*2	-	μs
Data setup time	t <sub>SU:DAT</sub>	I/O	250	-	100*3	-	ns
SDA and SCL signal rise time	t <sub>R</sub>	Input	-	1000	20	300	ns
SDA and SCL signal fall time*3	t <sub>F</sub>	Input	-	300	20 × (PV <sub>CC</sub> /5.5 V)	300	ns
		Output	-	250	20 × (PV <sub>CC</sub> /5.5 V)	250	ns
Setup time for STOP condition	t <sub>SU:STO</sub>	I/O	4.0	-	0.6	-	μs
Capacitive load for each bus line	C <sub>b</sub>	-	-	400*4	-	400*4	pF
Pulse width of spikes that must be suppressed by the input filter	t <sub>SP</sub>	Input	-	-	0	50*5	ns

In the above table and subsequently, SCL and SDA refer to the RIICnSCL and RIICnSDA signals, respectively.

Note 1. The first clock pulse is generated on the SCL line after the start condition has been issued and the hold time has elapsed.

Note 2. This module requires a minimum of 300 ns hold time internally for the SDA signal to handle the period over which the falling edge of SCL has not reached a defined level (time until the CnSCL signal reaches V<sub>IL</sub> (max.) from V<sub>IH</sub> (min.)).

Note 3. The fast-mode I<sup>2</sup>C bus device can be used in the standard mode I<sup>2</sup>C bus system. In this case, the minimum value of the data setup time (t<sub>SU:DAT</sub> (min.) 250 [ns]) must be satisfied.

If the system does not extend the low period of SCL clock (t<sub>LOW</sub>), this condition is automatically satisfied. If the system extends the low period of SCL clock (t<sub>LOW</sub>), transmit the subsequent data bit to the SDA line before the SCL line is released (t<sub>R</sub> (max.) + t<sub>SU:DAT</sub> (min.) = 1000 + 250 = 1250 [ns]; (standard mode I<sup>2</sup>C bus specification)).

Note 4. Total capacitance of one bus line. The allowable maximum bus capacitance may differ from this specification, depending on the actual operating voltage and frequency of an application. For techniques to cope with a large bus capacitance, see the I<sup>2</sup>C bus specification provided by NXP Semiconductors.

Note 5. Noise is removed by the analog and digital input filters. The level of noise reduction of the digital input filter is determined by the period of internal reference clock (IICφ) and the NF[1:0] bits in RIICnMR3. For details, refer to section 18, I<sup>2</sup>C Bus Interface.

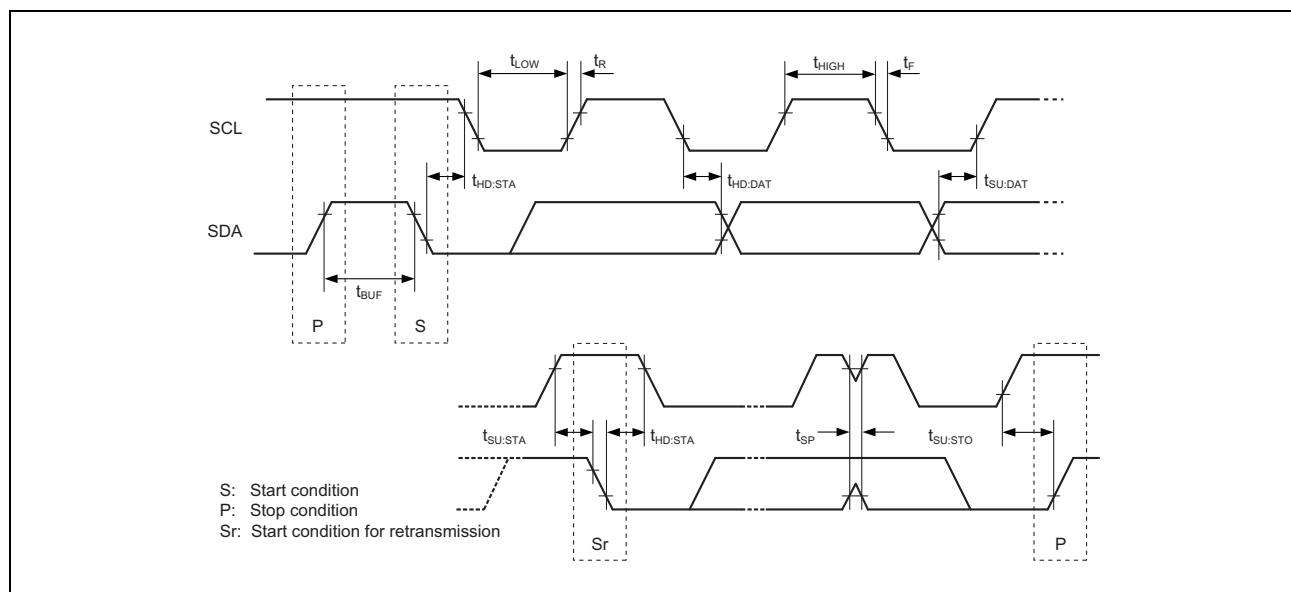
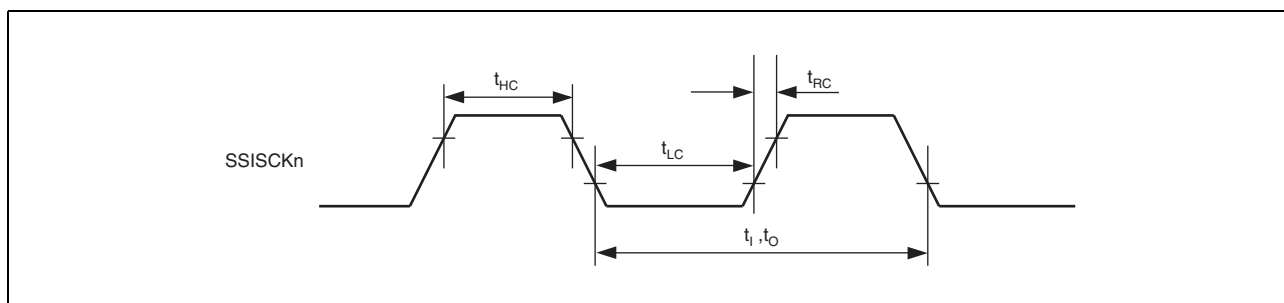


Figure 58.54 Input/Output Timing

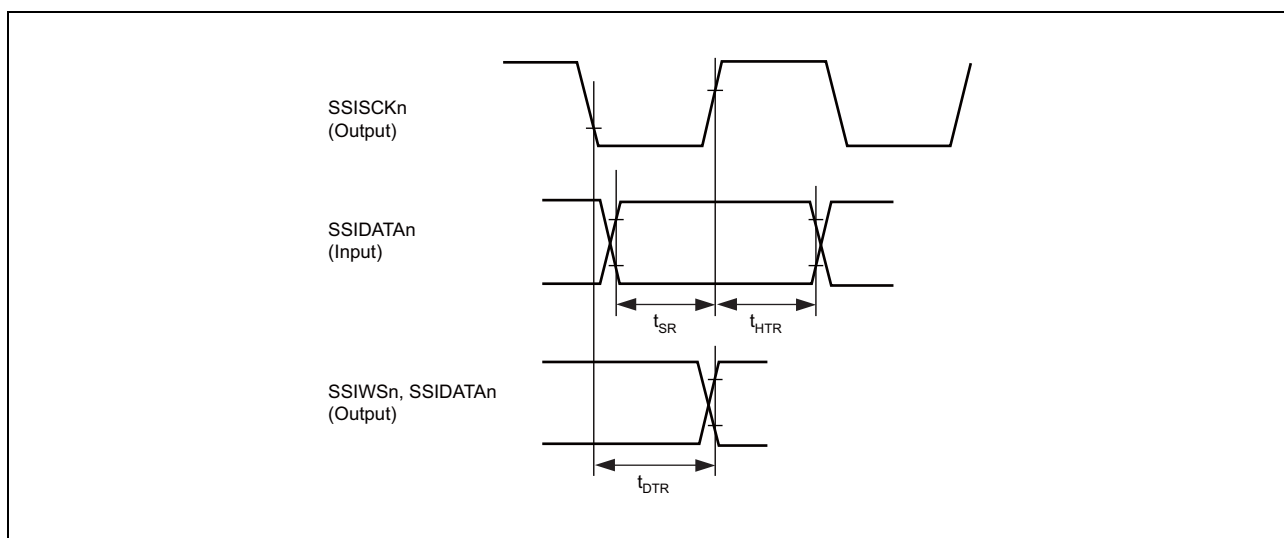
### 58.4.12 Serial Sound Interface Timing

**Table 58.16 Serial Sound Interface Timing**

Item	Symbol	Min.	Max.	Unit	Remarks	Figure
Output clock cycle	$t_o$	80	64000	ns	Output	Figure 58.55
Input clock cycle	$t_i$	80	64000	ns	Input	
Clock high	$t_{HC}$	32	—	ns	Bidirectional	
Clock low	$t_{LC}$	32	—	ns		
Clock rise time	$t_{RC}$	—	25	ns	Output	
Delay	Noise canceler not in use	$t_{DTR}$	-5	25	ns	Figure 58.56, Figure 58.57, Figure 58.58, Figure 58.59, and Figure 58.60
	Noise canceler in use		10	45	ns	
Setup time	$t_{SR}$	25	—	ns		
Hold time	$t_{HTR}$	5	—	ns		



**Figure 58.55 Clock Input/Output Timing**



**Figure 58.56 Transmission and Reception Timing (Master, SSICR\_n.SCKP = 0)**

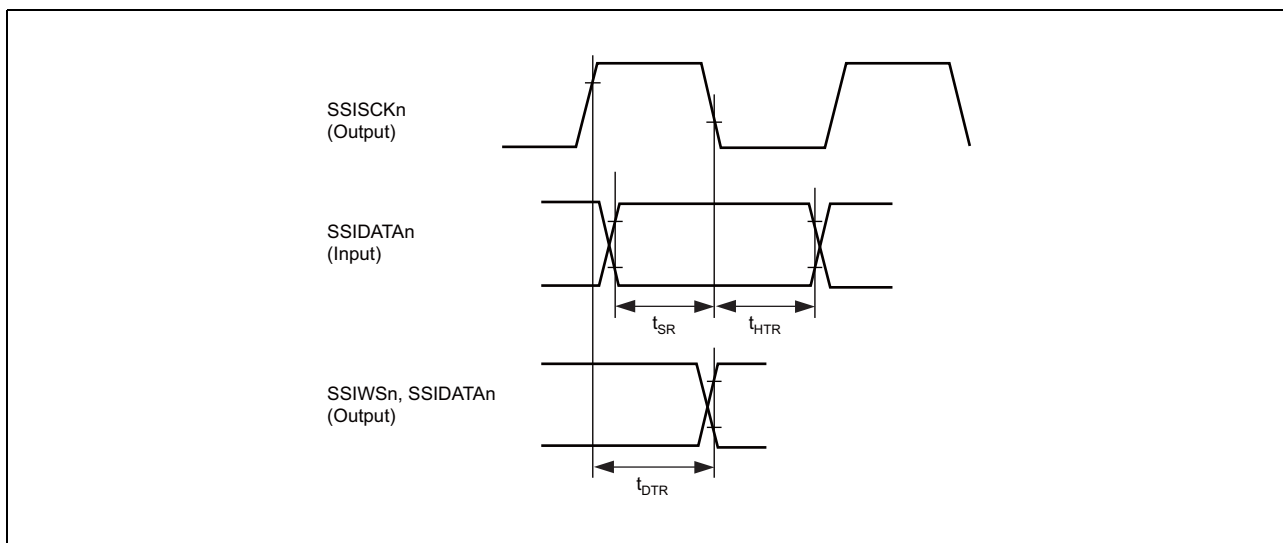


Figure 58.57 Transmission and Reception Timing (Master, SSICR\_n.SCKP = 1)

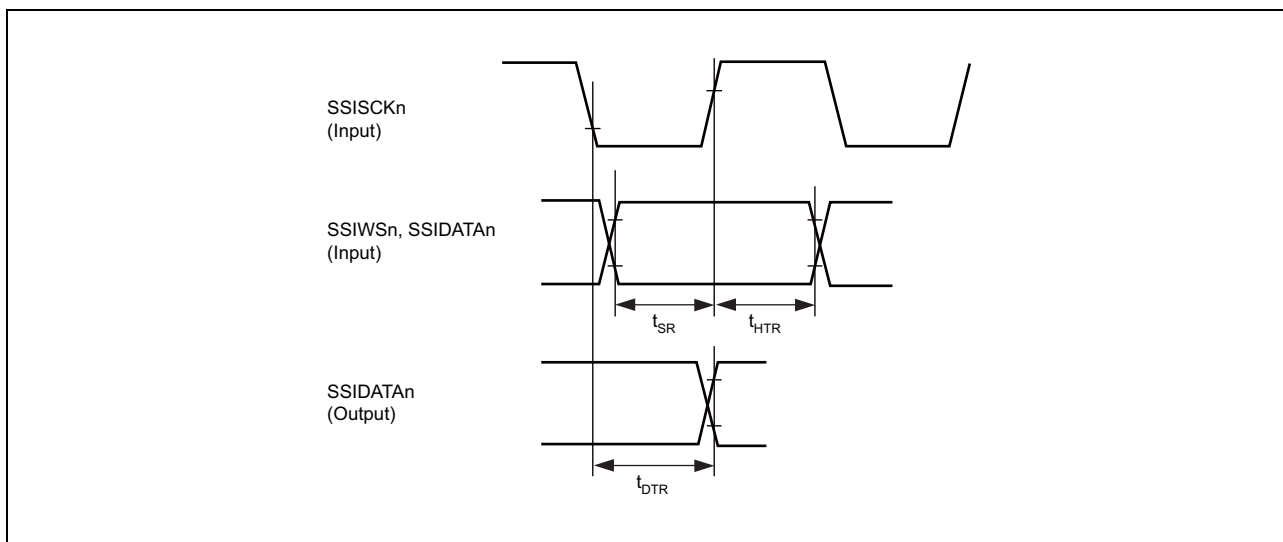


Figure 58.58 Transmission and Reception Timing (Slave, SSICR\_n.SCKP = 0)

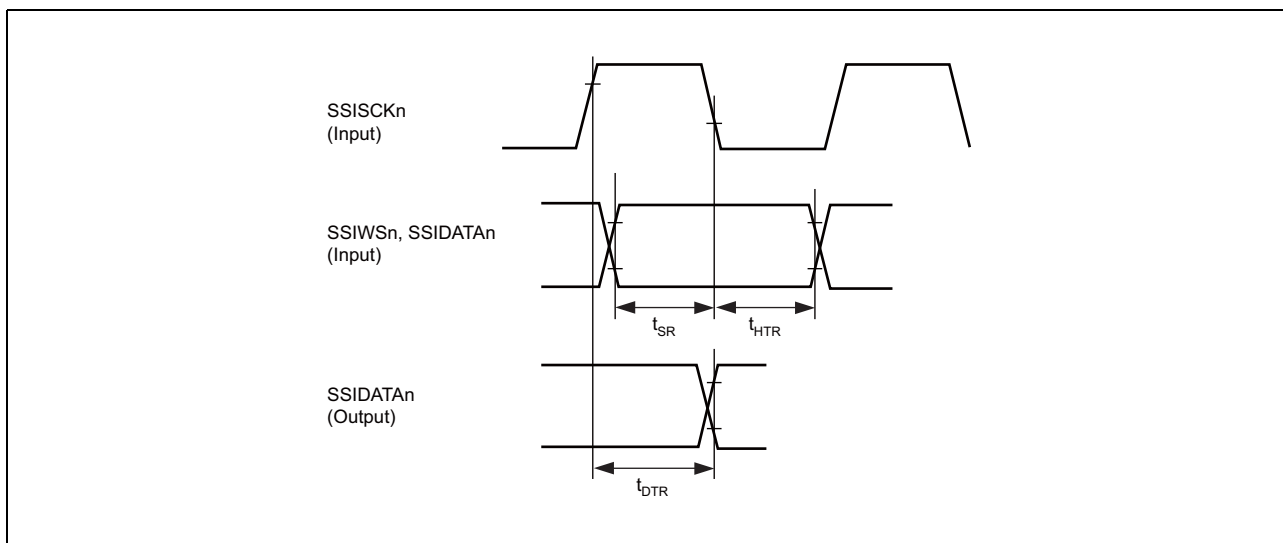
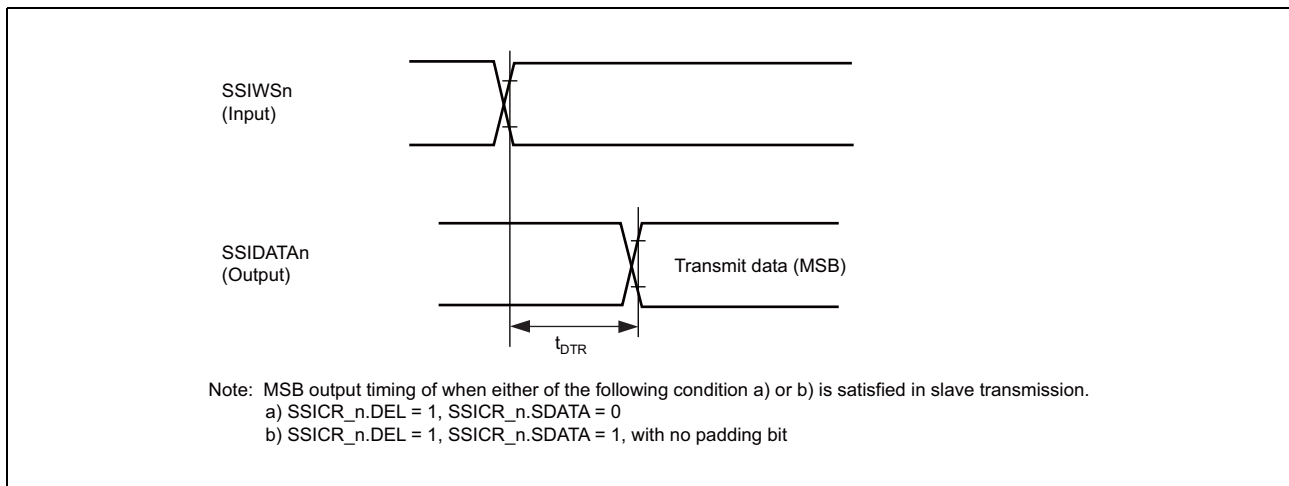


Figure 58.59 Transmission and Reception Timing (Slave, SSICR\_n.SCKP = 1)





**Figure 58.60** Transmission Timing (Slave, in Synchronization with SSIWSn)

## 58.4.13 Media Local Bus Timing

Table 58.17 Media Local Bus Timing

Item	Symbol	Min.	Typ	Max.	Unit	Remarks	Figure
Input clock frequency (256 × FS)	$f_i$	11.2640	12.2880	12.3136	MHz		Figure 58.61
Input clock cycle (256 × FS)	$t_i$	—	81	—	ns		
Input clock high level (256 × FS)	$t_{HC}$	30	36.5	—	ns		
Input clock low level (256 × FS)	$t_{LC}$	30	35.5	—	ns		
Input clock frequency (512 × FS)	$f_i$	22.5280	24.5760	24.6272	MHz		
Input clock cycle (512 × FS)	$t_i$	—	40	—	ns		
Input clock high level (512 × FS)	$t_{HC}$	14	16.5	—	ns		
Input clock low level (512 × FS)	$t_{LC}$	14	16.5	—	ns		
Input clock frequency (1024 × FS)	$f_i$	45.0560	49.1520	49.2544	MHz		
Input clock cycle (1024 × FS)	$t_i$	—	20.3	—	ns		
Input clock high level (1024 × FS)	$t_{HC}$	9.3	10.2	—	ns		
Input clock low level (1024 × FS)	$t_{LC}$	6.1	7.3	—	ns		
Input clock rise time	$t_{RC}$	—	—	1	ns	$V_{IL}$ to $V_{IH}$	
Input clock fall time	$t_{FC}$	—	—	1	ns	$V_{IH}$ to $V_{IL}$	
Delay time (clock signal rising)	$t_{DTR}$	—	—	8.0	ns	Output load: 20 pF	
Delay time (clock signal falling)	$t_{DTF}$	0	—	$t_{LC}$	ns		
Setup time	$t_{SR}$	1	—	—	ns		
Hold time	$t_{HTR}$	2	—	—	ns		

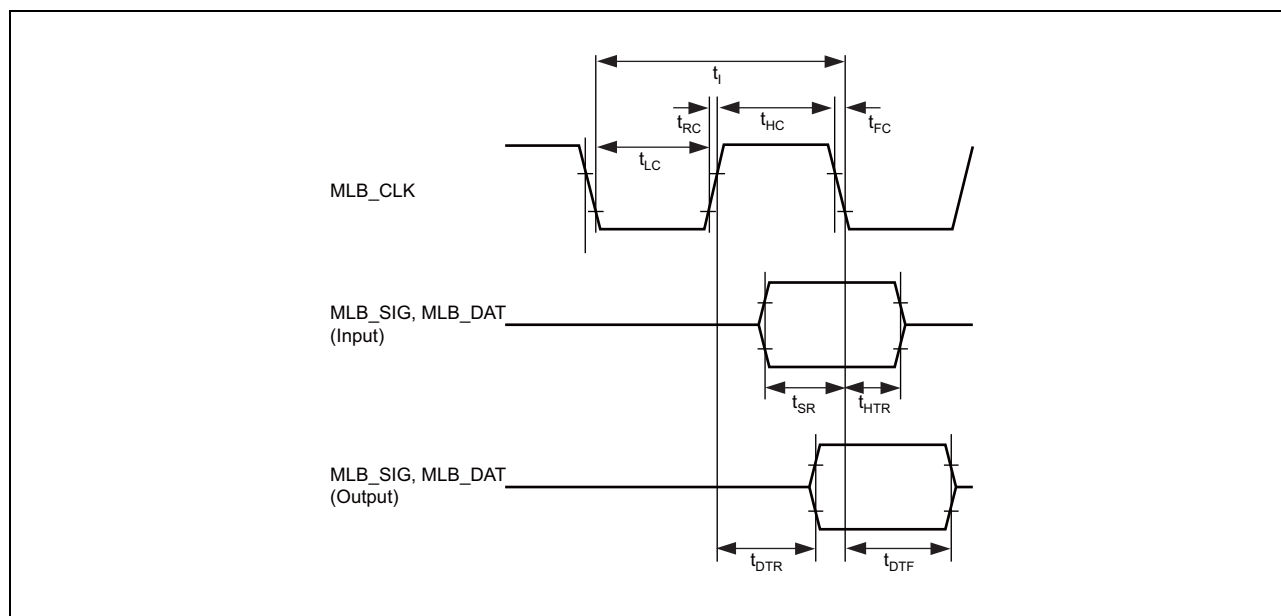


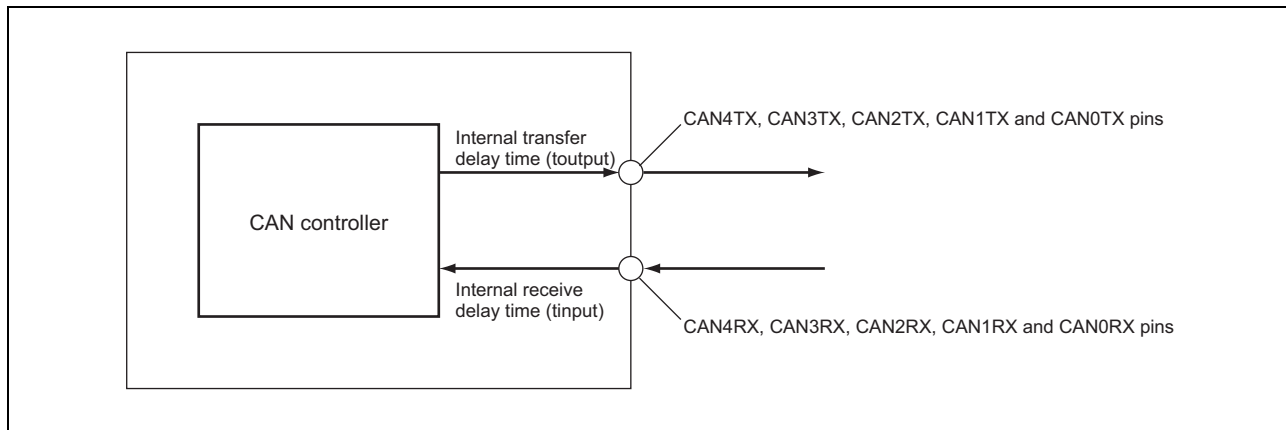
Figure 58.61 Interface Timing

58.4.14 CAN Interface Timing

**Table 58.18 CAN Interface Timing**

Item	Symbol	Min.	Max.	Unit	Figure
Internal delay time	t <sub>node</sub>	—	100	ns	Figure 58.62
Transmission rate		—	1	Mbps	

Internal delay time (t<sub>node</sub>) = Internal transfer delay time (t<sub>output</sub>) + Internal receive delay time (t<sub>input</sub>)



**Figure 58.62 CAN Interface Condition**

## 58.4.15 Ethernet Controller and EthernetAVB Timing

Table 58.19 Ethernet Controller Timing

Item	Symbol	Min.	Max.	Unit	Figure
ET_TXCLK cycle time	$t_{Tcyc}$	40	—	ns	Figure 58.63,
ET_TXCLK high level width	$t_{TCKWH}$	$0.35 \times t_{Tcyc}$	—	ns	Figure 58.64,
ET_TXCLK low level width	$t_{TCKWL}$	$0.35 \times t_{Tcyc}$	—	ns	Figure 58.65, and
ET_TXEN output delay time	$t_{TEND}$	0	25	ns	Figure 58.67
ET_TXD[3:0] output delay time	$t_{TDD}$	0	25	ns	
ET_RXCLK cycle time	$t_{Rcyc}$	40	—	ns	
ET_RXCLK high level width	$t_{RCKWH}$	$0.35 \times t_{Rcyc}$	—	ns	
ET_RXCLK low level width	$t_{RCKWL}$	$0.35 \times t_{Rcyc}$	—	ns	
ET_RXDV setup time	$t_{RDVS}$	10	—	ns	
ET_RXDV hold time	$t_{RDVH}$	10	—	ns	
ET_RXD[3:0] setup time	$t_{RDDS}$	10	—	ns	
ET_RXD[3:0] hold time	$t_{RDDH}$	10	—	ns	
ET_RXER setup time	$t_{RERS}$	10	—	ns	
ET_RXER hold time	$t_{RERH}$	10	—	ns	
AVB_GPTP_EXTERN cycle time	$t_{Gcyc}$	40	—	ns	
AVB_GPTP_EXTERN high level width	$t_{GCKWH}$	$0.35 \times t_{Gcyc}$	—	ns	
AVB_GPTP_EXTERN low level width	$t_{GCKWL}$	$0.35 \times t_{Gcyc}$	—	ns	
AVB_CAPTURE high level width	$t_{CAPWH}$	$2 \times t_{Ccyc}^*$	—	ns	

Note: \* This is the cycle time of the clock selected by the CSEL bit in the AVB-DMAC mode register (CCC).

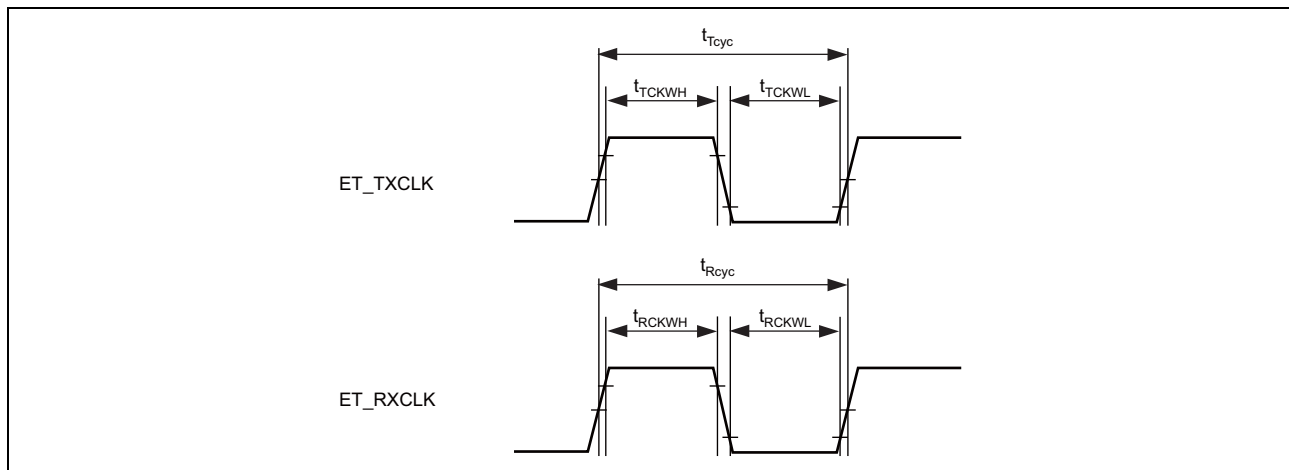


Figure 58.63 MII Clock Timing

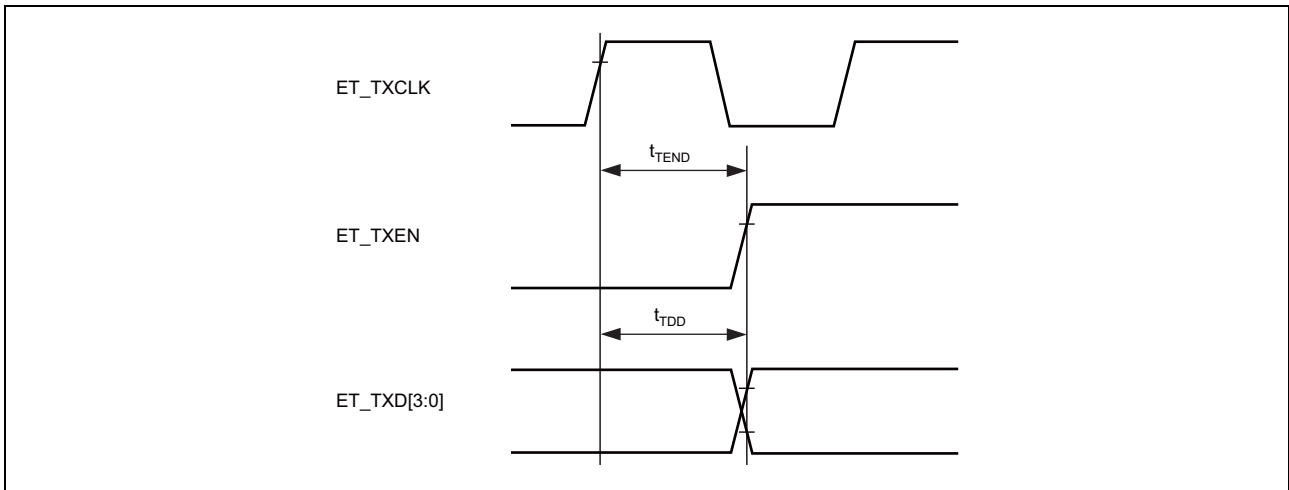


Figure 58.64 MII Transmit Data Timing

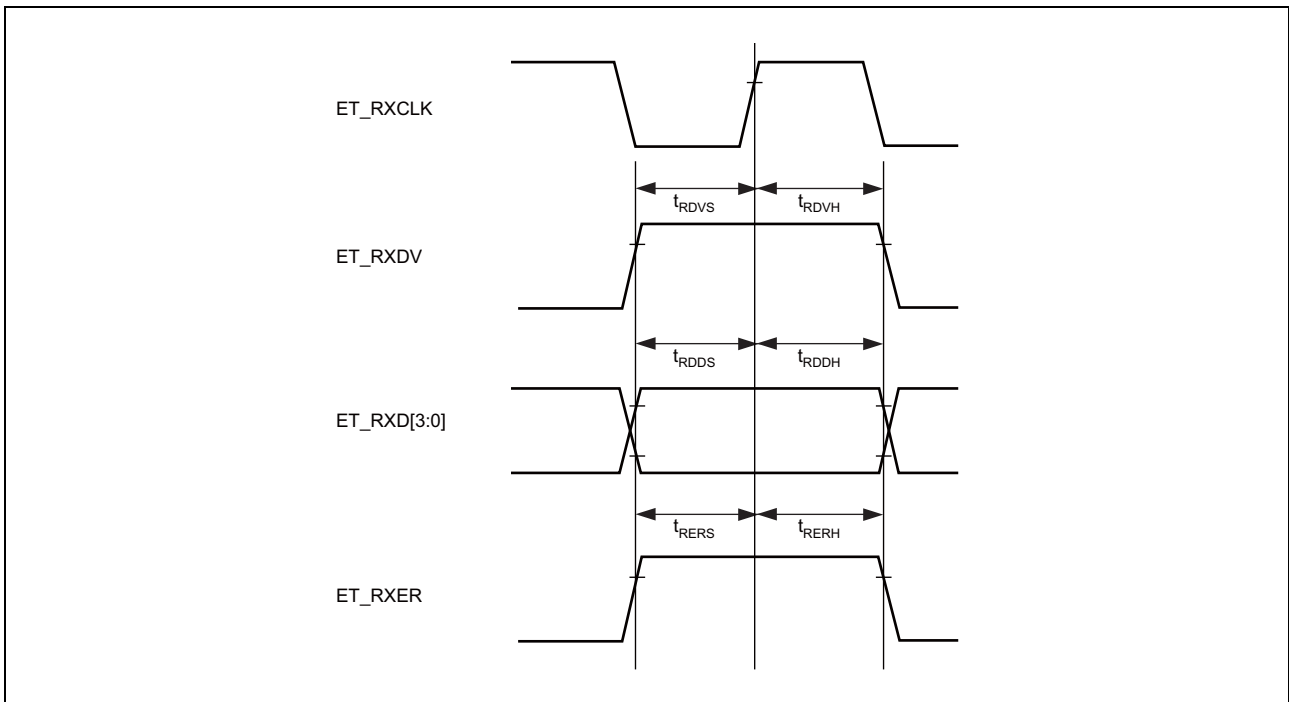


Figure 58.65 MII Receive Data Timing

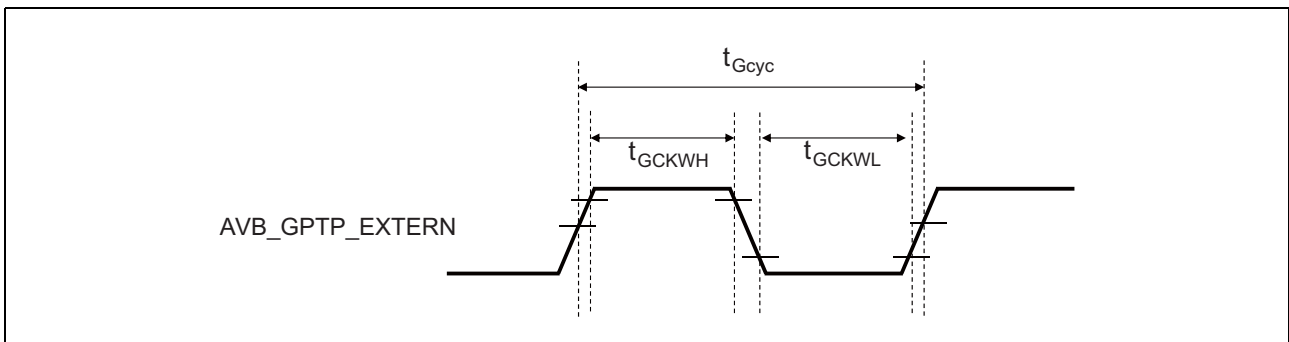


Figure 58.66 gPTP Timer External Clock Timing

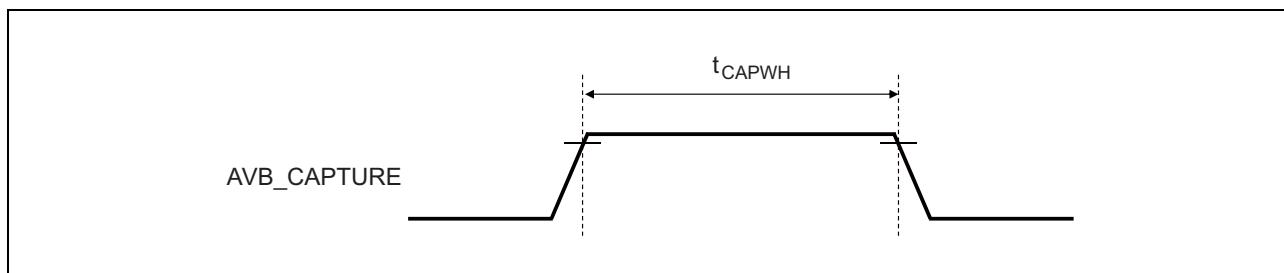


Figure 58.67 Timer Capture Signal Timing

58.4.16 A/D Converter Timing

Table 58.20 A/D Converter Timing

Module	Item	Symbol	Min.	Max.	Unit	Figure
A/D converter	Trigger input setup time	$t_{TRGS}$	17	—	ns	Figure 58.68

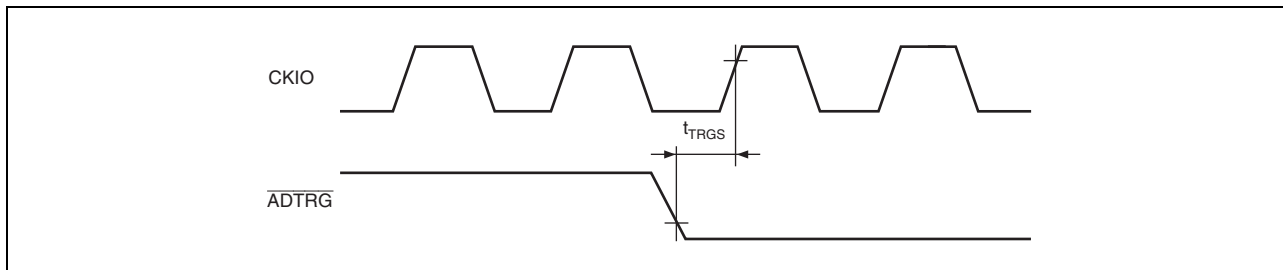


Figure 58.68 A/D Converter External Trigger Input Timing

## 58.4.17 NAND Type Flash Memory Controller Timing

Table 58.21 NAND Type Flash Memory Interface Timing

Item	Symbol	Min.	Max.	Unit	Figure
Command output setup time	t <sub>NCDS</sub>	$2 \times t_{f_{cyc}} - 10$	—	ns	Figure 58.69 and Figure 58.73
Command output hold time	t <sub>NCDH</sub>	$1.5 \times t_{f_{cyc}} - 5$	—	ns	Figure 58.73
Data output setup time	t <sub>NDOS</sub>	$0.5 \times t_{w_{f_{cyc}}} - 5$	—	ns	Figure 58.69, Figure 58.70, Figure 58.72, and Figure 58.73
Data output hold time	t <sub>NDOH</sub>	$0.5 \times t_{w_{f_{cyc}}} - 10$	—	ns	Figure 58.72, and Figure 58.73
Command to address transition time 1	t <sub>NCAD1</sub>	$1.5 \times t_{f_{cyc}} - 10$	—	ns	Figure 58.69 and Figure 58.70
Command to address transition time 2	t <sub>NCAD2</sub>	$2 \times t_{f_{cyc}} - 10$	—	ns	Figure 58.70
FWE cycle time	t <sub>NWC</sub>	$t_{w_{f_{cyc}}} - 5$	—	ns	Figure 58.70 and Figure 58.72
FWE low pulse width	t <sub>NWP</sub>	$0.5 \times t_{w_{f_{cyc}}} - 5$	—	ns	Figure 58.69, Figure 58.70, Figure 58.72, and Figure 58.73
FWE high pulse width	t <sub>NWH</sub>	$0.5 \times t_{w_{f_{cyc}}} - 5$	—	ns	Figure 58.70 and Figure 58.72
Address to ready/busy transition time	t <sub>NADRB</sub>	—	$32 \times t_{p0cyc}$	ns	Figure 58.70 and Figure 58.71
Command to ready/busy transition time	t <sub>NCDRB</sub>	—	$10 \times t_{p0cyc}$	ns	Figure 58.70 and Figure 58.71
Ready/busy to data read transition time 1	t <sub>NRBDR1</sub>	$1.5 \times t_{f_{cyc}}$	—	ns	Figure 58.71
Ready/busy to data read transition time 2	t <sub>NRBDR2</sub>	$32 \times t_{p0cyc}$	—	ns	
FRE cycle time	t <sub>NSCC</sub>	$t_{w_{f_{cyc}}} - 5$	—	ns	
FRE low pulse width	t <sub>NSP</sub>	$0.5 \times t_{w_{f_{cyc}}} - 5$	—	ns	Figure 58.71 and Figure 58.73
FRE high pulse width	t <sub>NSPH</sub>	$0.5 \times t_{w_{f_{cyc}}} - 5$	—	ns	Figure 58.71
Read data setup time	t <sub>NRDS</sub>	16	—	ns	Figure 58.71 and Figure 58.73
Read data hold time	t <sub>NRDH</sub>	5	—	ns	Figure 58.71 and Figure 58.73
Data write setup time	t <sub>NDWS</sub>	$32 \times t_{p0cyc}$	—	ns	Figure 58.72
Command to status read transition time	t <sub>NCDSR</sub>	$4 \times t_{f_{cyc}}$	—	ns	Figure 58.73
Command output off to status read transition time	t <sub>NCDFSR</sub>	$3.5 \times t_{f_{cyc}}$	—	ns	
Status read setup time	t <sub>NSTS</sub>	$2.5 \times t_{f_{cyc}}$	—	ns	

Note: t<sub>f<sub>cyc</sub></sub> indicates the period of one cycle of the FLCTL clock.

t<sub>w<sub>f<sub>cyc</sub></sub></sub> indicates the period of one cycle of the FLCTL clock when the value of the NANDWF bit is 0. On the other hand, t<sub>w<sub>f<sub>cyc</sub></sub></sub> indicates the period of two cycles of the FLCTL clock when the value of the NANDWF bit is 1.

t<sub>p0cyc</sub> indicates the period of one cycle of the peripheral clock (P0φ).



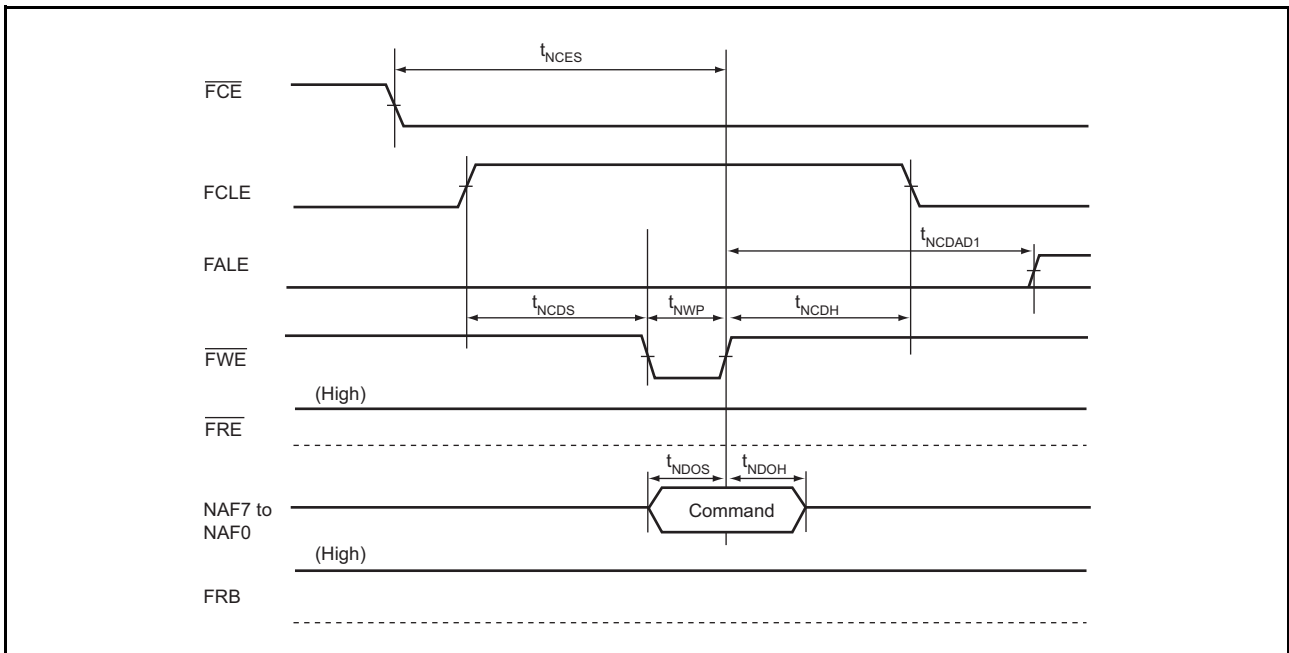


Figure 58.69 NAND Type Flash Memory Command Issuance Timing

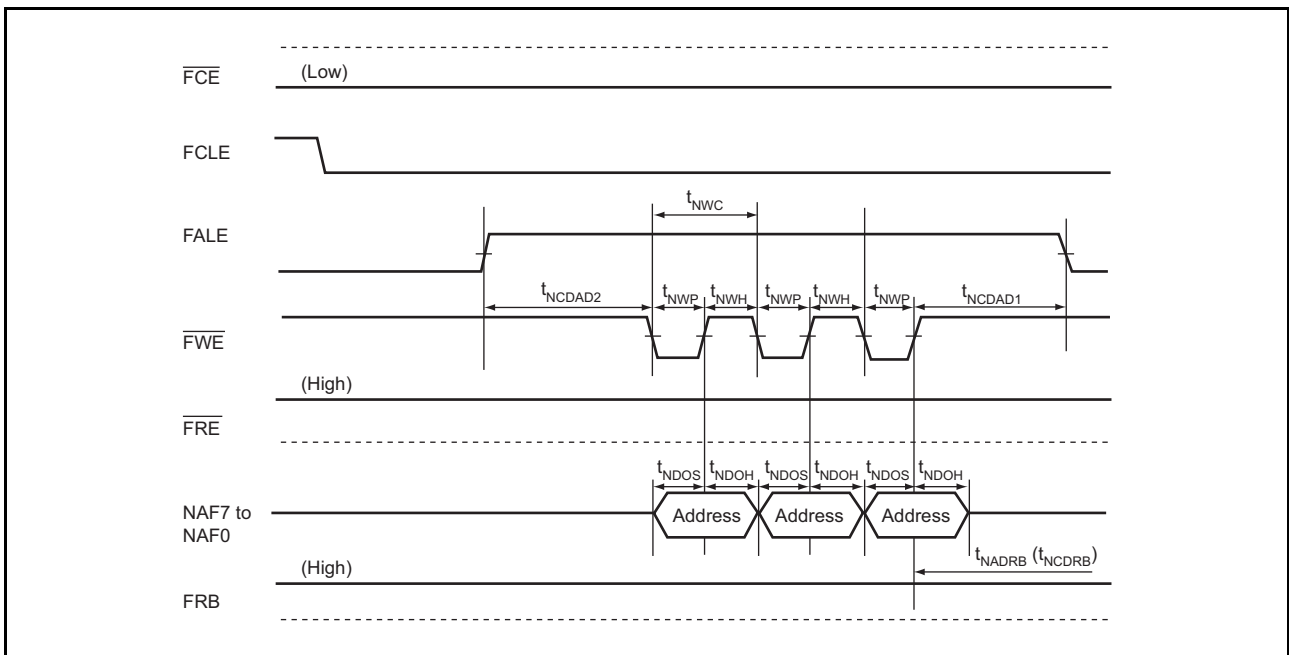


Figure 58.70 NAND Type Flash Memory Address Issuance Timing

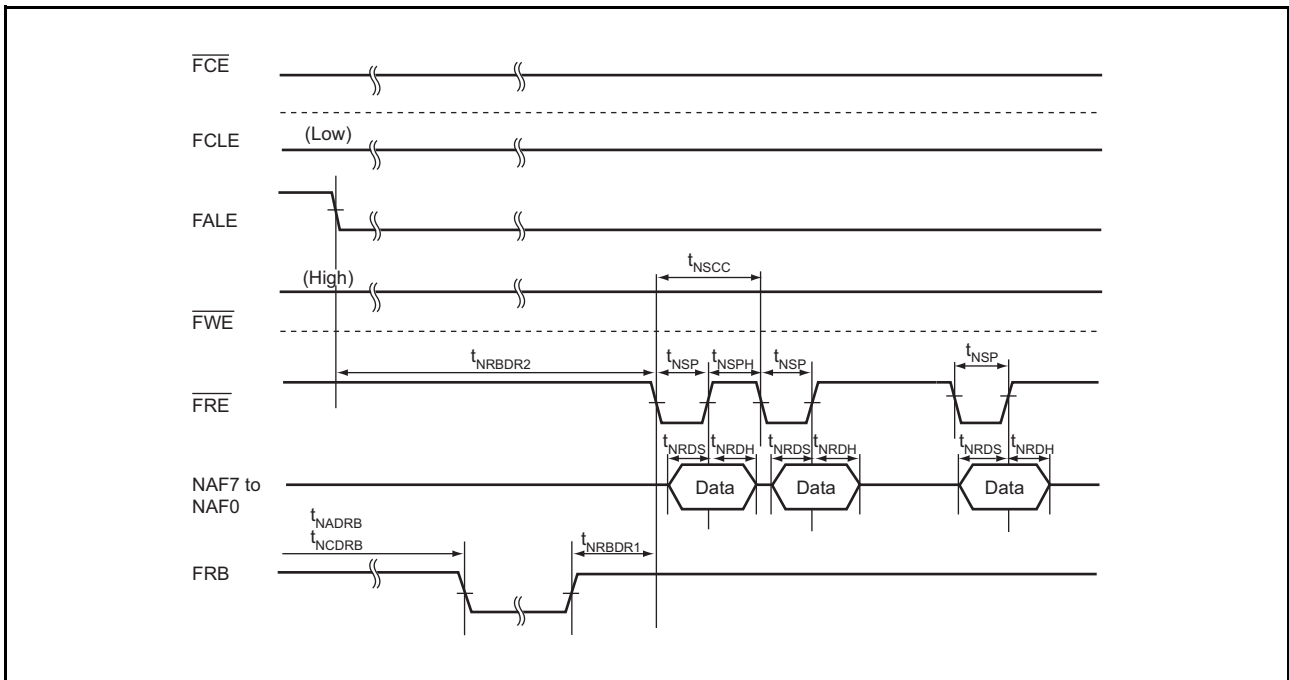


Figure 58.71 NAND Type Flash Memory Data Read Timing

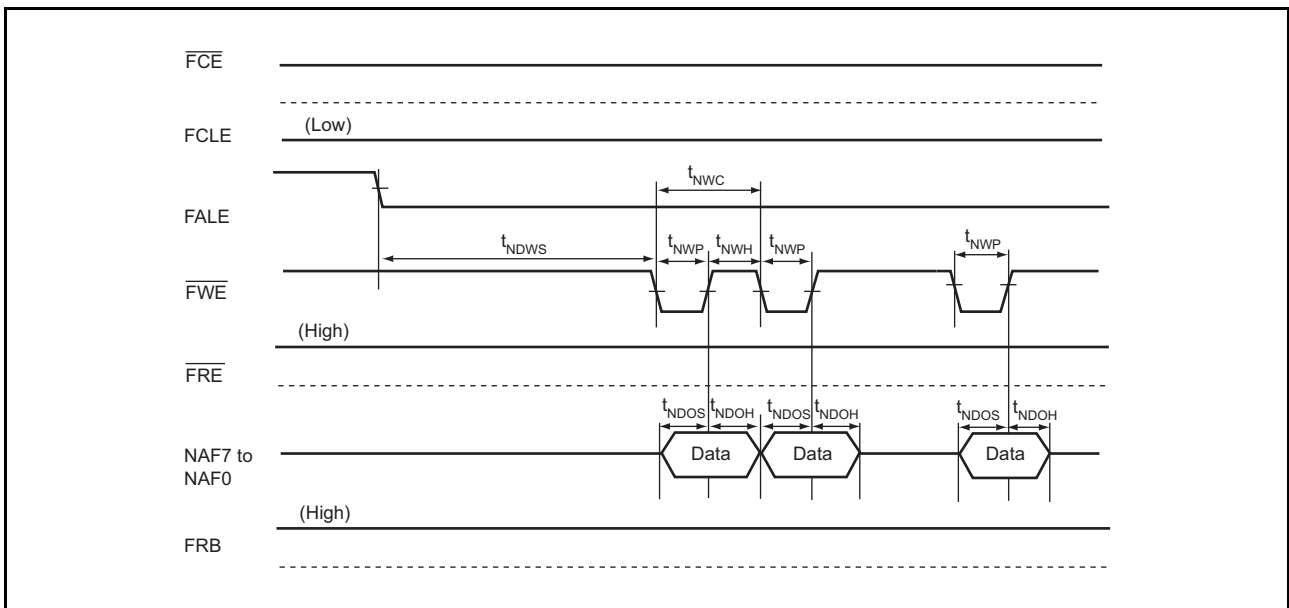


Figure 58.72 NAND Type Flash Memory Data Write Timing

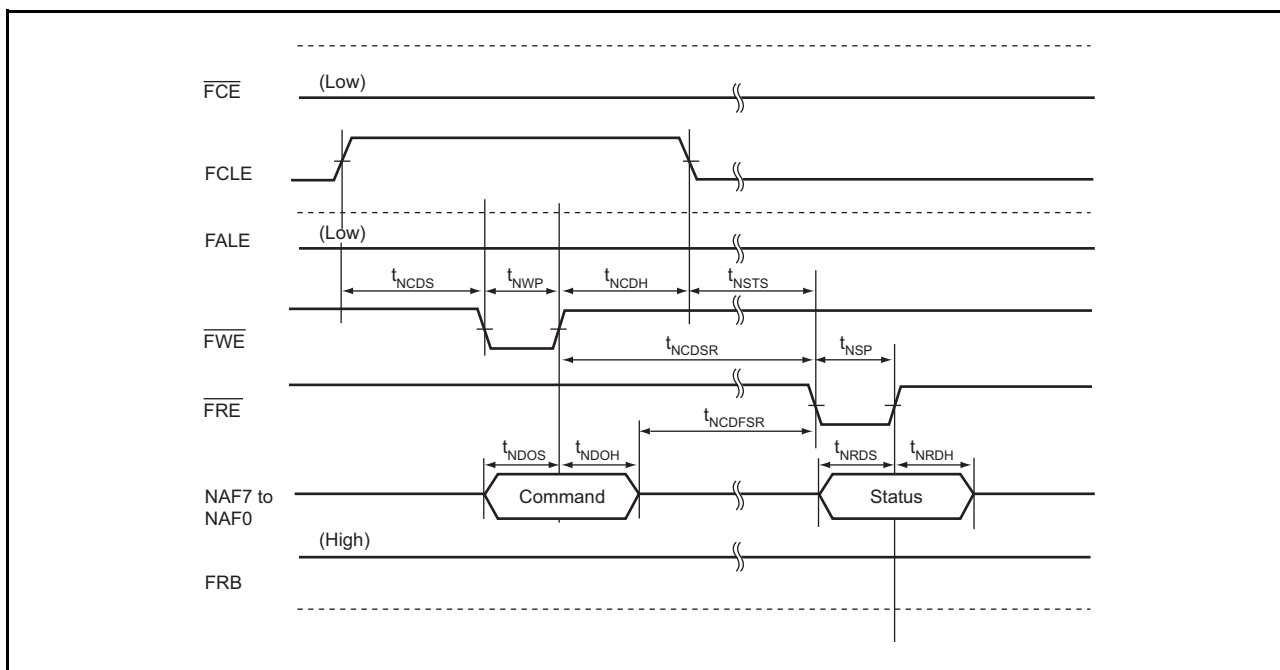
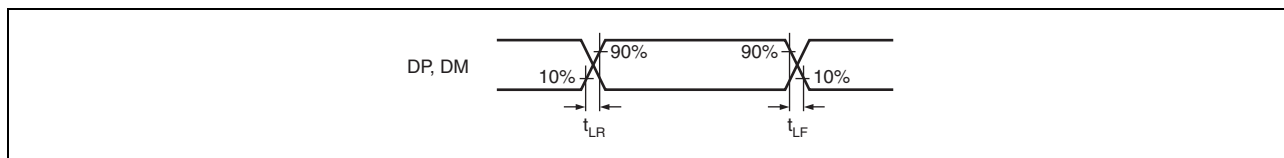


Figure 58.73 NAND Type Flash Memory Status Read Timing

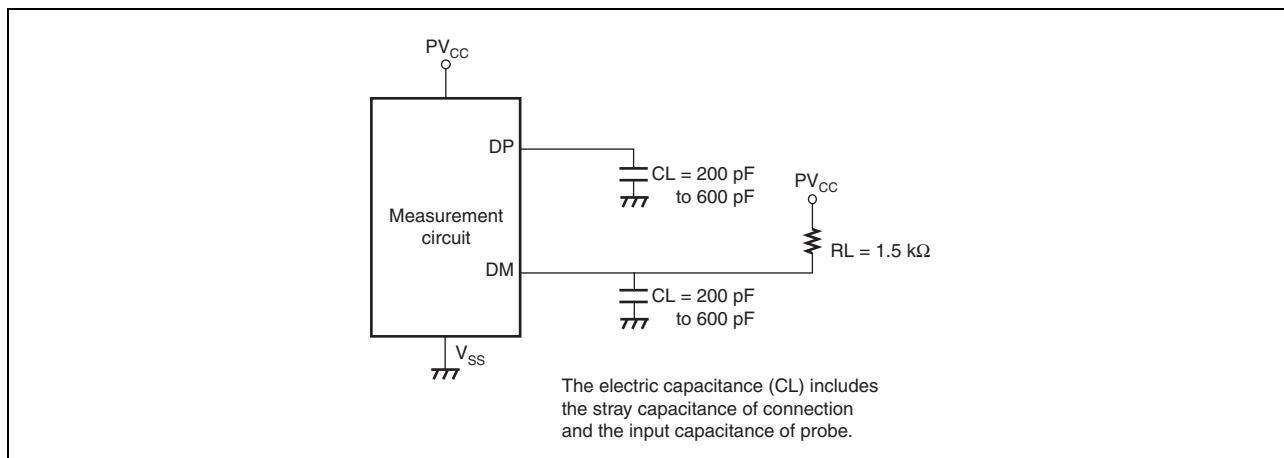
58.4.18 USB 2.0 Host/Function Module Timing

**Table 58.22 USB Transceiver Timing (Low-Speed)**

Item	Symbol	Min.	Max.	Unit	Figure
Rise time	$t_{LR}$	75	300	ns	Figure 58.74
Fall time	$t_{LF}$	75	300	ns	
Rise/fall time lag	$t_{LR}/t_{LF}$	80	125	%	



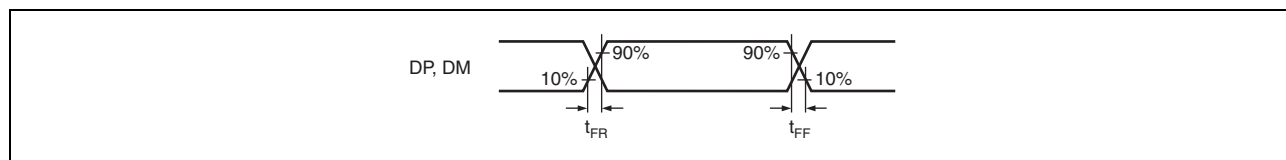
**Figure 58.74 DP1, DP0, DM1, and DM0 Output Timing (Low-Speed)**



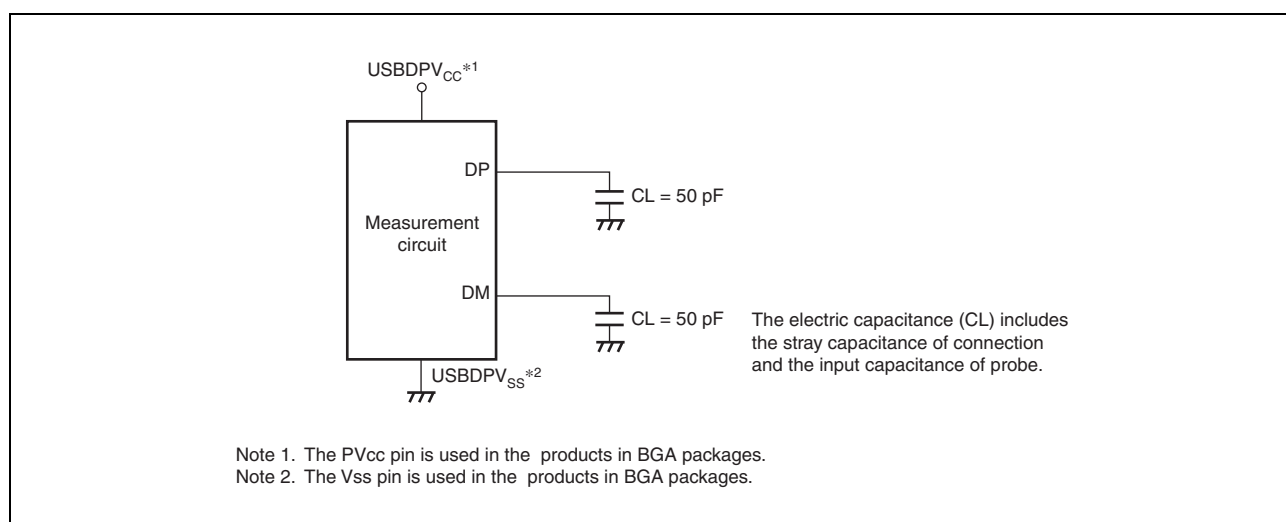
**Figure 58.75 Measurement Circuit (Low-Speed)**

**Table 58.23 USB Transceiver Timing (Full-Speed)**

Item	Symbol	Min.	Max.	Unit	Figure
Rise time	$t_{FR}$	4	20	ns	Figure 58.76
Fall time	$t_{FF}$	4	20	ns	
Rise/fall time lag	$t_{FR}/t_{FF}$	90	111.11	%	



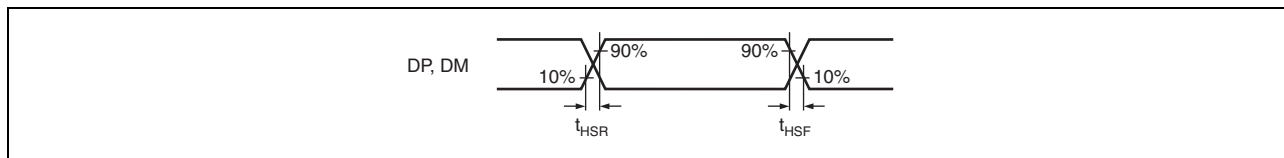
**Figure 58.76 DP1, DP0, DM1, and DM0 Output Timing (Full-Speed)**



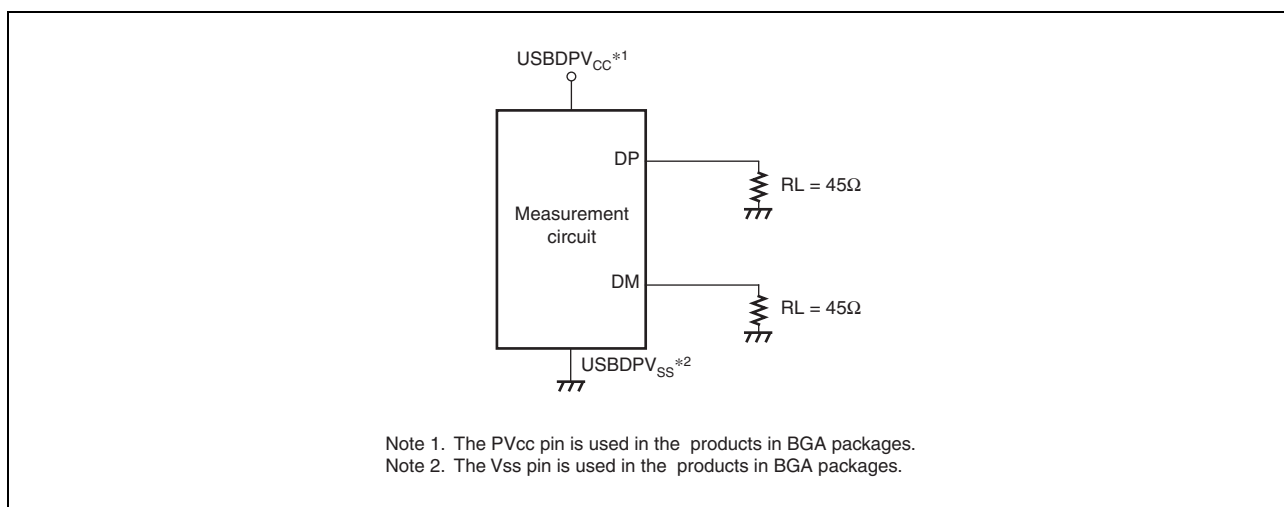
**Figure 58.77 Measurement Circuit (Full-Speed)**

**Table 58.24 USB Transceiver Timing (High-Speed)**

Item	Symbol	Min.	Max.	Unit	Figure
Rise time	$t_{HSR}$	500	—	ps	Figure 58.78
Fall time	$t_{HSF}$	500	—	ps	
Output driver resistance	$Z_{HSDRV}$	40.5	49.5	$\Omega$	



**Figure 58.78 DP1, DP0, DM1, and DM0 Output Timing (High-Speed)**



**Figure 58.79 Measurement Circuit (High-Speed)**

## 58.4.19 Video Display Controller 5 Timing

Table 58.25 Video Display Controller 5 Timing

Item	Symbol	Min.	Max.	Unit	Figure
DV1_CLK and DV0_CLK input clock frequency	$t_{Dcyc}$	—	87.00	MHz	Figure 58.80
DV1_CLK and DV0_CLK input clock low pulse width	$t_{WL}$	0.4	—	$t_{Dcyc}$	
DV1_CLK and DV0_CLK input clock high pulse width	$t_{WH}$	0.4	—		
LCD1_EXTCLK and LCD0_EXTCLK input clock frequency	$t_{Ecyc}$	—	87.00	MHz	
LCD1_EXTCLK and LCD0_EXTCLK input clock low pulse width	$t_{WL}$	0.4	—	$t_{Ecyc}$	
LCD1_EXTCLK and LCD0_EXTCLK input clock high pulse width	$t_{WH}$	0.4	—		
LCD1_CLK and LCD0_CLK output clock frequency	$t_{Lcyc}$	—	87.00	MHz	Figure 58.81
LCD1_CLK and LCD0_CLK clock output low pulse width	$t_{LOL}$	0.45	0.55	$t_{Lcyc}$	
LCD1_CLK and LCD0_CLK clock output high pulse width	$t_{LOH}$	0.45	0.55	$t_{Lcyc}$	
LCD1_CLK and LCD0_CLK clock output rise time	$t_{LOR}$	—	3	ns	
LCD1_CLK and LCD0_CLK clock output fall time	$t_{LOF}$	—	3	ns	
Input data setup time	$t_{vs}$	2.5	—	ns	Figure 58.82
Input data hold time	$t_{vH}$	3.5	—	ns	
Output data delay time	$t_{dD}$	-3	3	ns	Figure 58.83

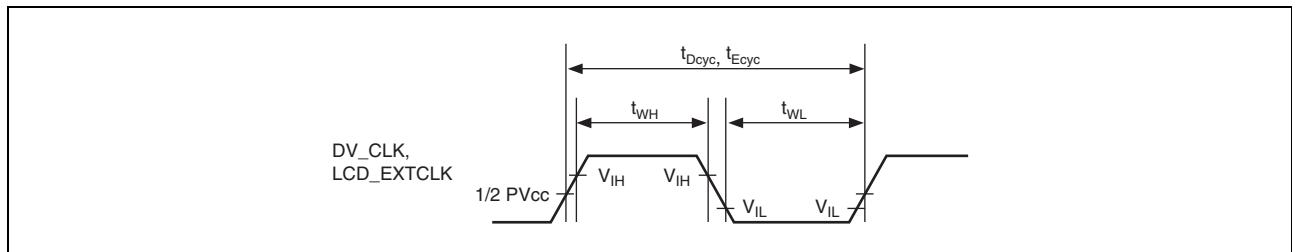


Figure 58.80 DV1\_CLK, DV0\_CLK, LCD1\_EXTCLK, and LCD0\_EXTCLK Clock Input Timing

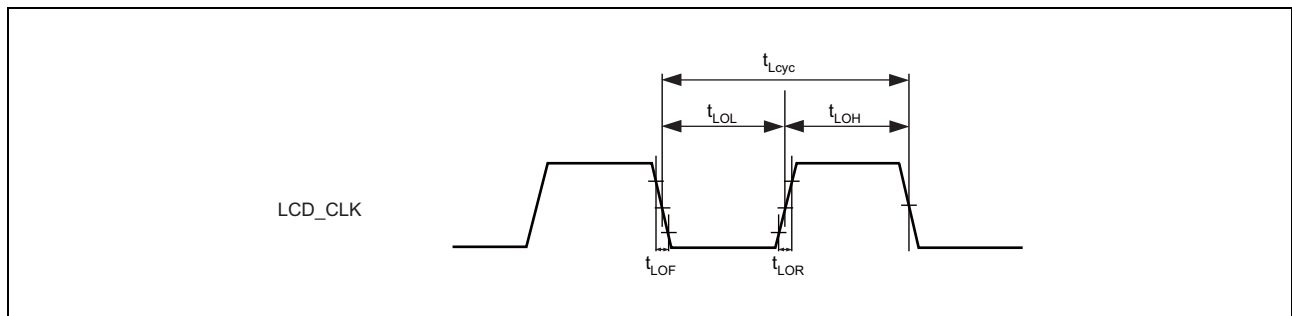


Figure 58.81 LCD1\_CLK and LCD0\_CLK Clock Output Timing

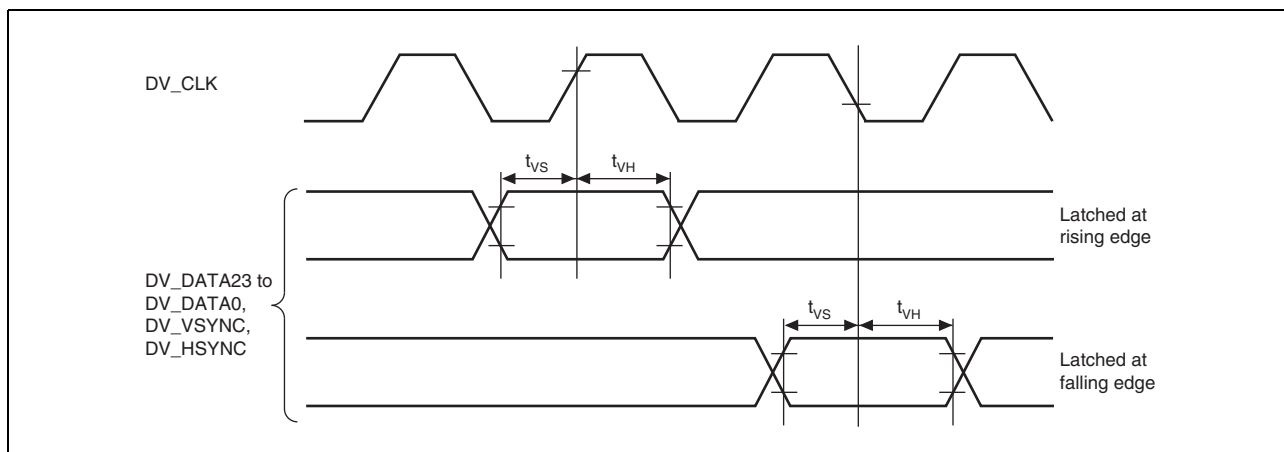


Figure 58.82 Video Input Timing

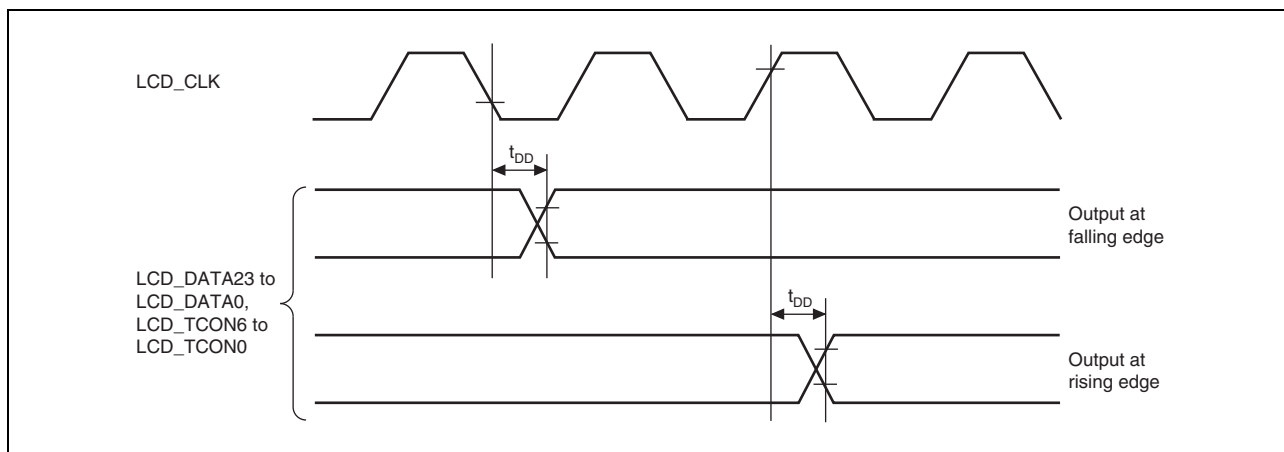


Figure 58.83 Display Output Timing



58.4.20 LVDS Timing

Table 58.26 LVDS Timing

Item	Symbol	Min.	Typ.	Max.	Unit	Figure
Rise time	LLHT	—	—	1.5	ns	Figure 58.84,
Fall time	LHLT	—	—	1.5	ns	Figure 58.85,
Transmitter Output Pulse Position for Bit1	TPPos1	-0.20	0	0.20	ns	and
Transmitter Output Pulse Position for Bit0	TPPos0	$T/7 - 0.20$	$T/7$	$T/7 + 0.20$	ns	Figure 58.86
Transmitter Output Pulse Position for Bit6	TPPos6	$T \times 2/7 - 0.20$	$T \times 2/7$	$T \times 2/7 + 0.20$	ns	
Transmitter Output Pulse Position for Bit5	TPPos5	$T \times 3/7 - 0.20$	$T \times 3/7$	$T \times 3/7 + 0.20$	ns	
Transmitter Output Pulse Position for Bit4	TPPos4	$T \times 4/7 - 0.20$	$T \times 4/7$	$T \times 4/7 + 0.20$	ns	
Transmitter Output Pulse Position for Bit3	TPPos3	$T \times 5/7 - 0.20$	$T \times 5/7$	$T \times 5/7 + 0.20$	ns	
Transmitter Output Pulse Position for Bit2	TPPos2	$T \times 6/7 - 0.20$	$T \times 6/7$	$T \times 6/7 + 0.20$	ns	
TXOUT Inter Channel skew	TCCS	—	—	200	ps	

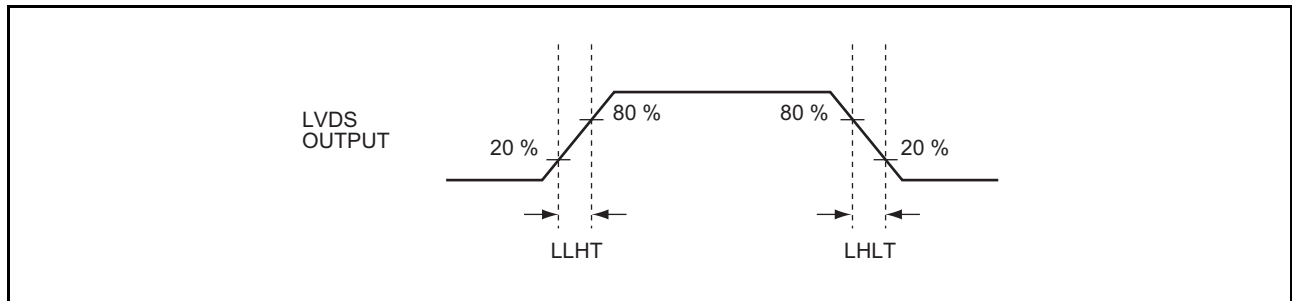


Figure 58.84 Transmitter LVDS Output Transition Time

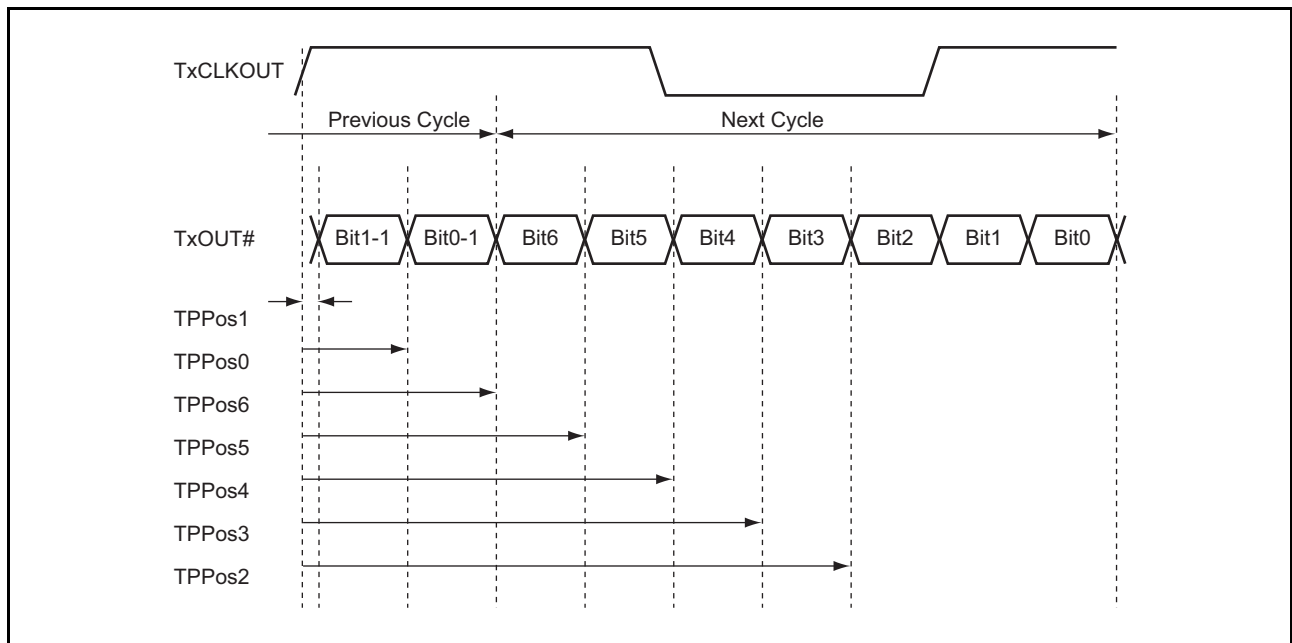


Figure 58.85 Transmitter LVDS Output Pulse Position Measurement

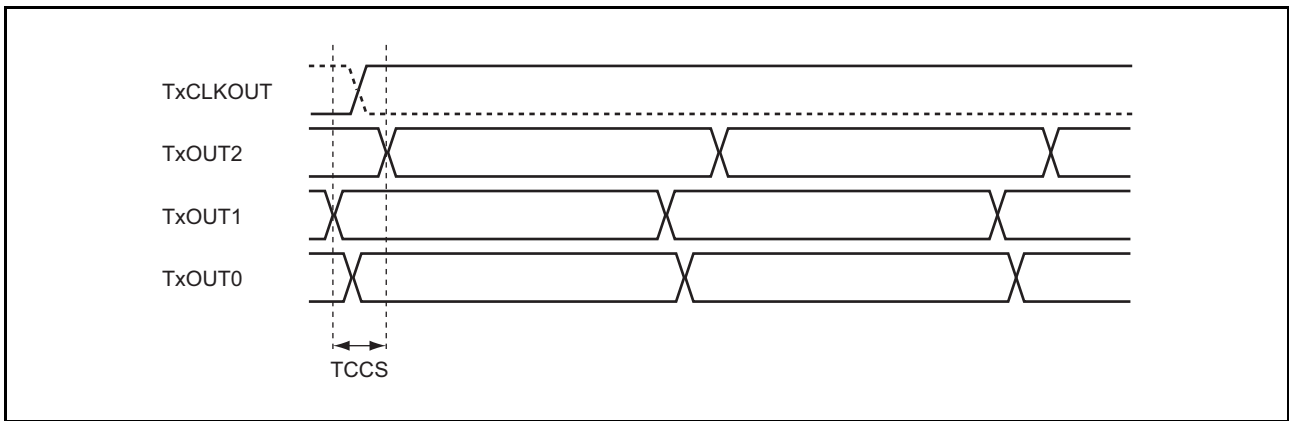


Figure 58.86 Transmitter Inter Channel Skew

## 58.4.21 Capture Engine Unit Module Signal Timing

Table 58.27 Capture Engine Unit Module Signal Timing

Item	Symbol	Min.	Max.	Unit	Figure
Vertical sync (VIO_VD) setup time	$t_{VDS}$	2	—	ns	Figure 58.87
Vertical sync (VIO_VD) hold time	$t_{VDH}$	3.5	—	ns	
Horizontal sync (VIO_VD) setup time	$t_{VHDS}$	2	—	ns	
Horizontal sync (VIO_VD) hold time	$t_{VHDH}$	3.5	—	ns	
Capture image data (VIO_D) setup time	$t_{VDS}$	2	—	ns	
Capture image data (VIO_D) hold time	$t_{VDH}$	3.5	—	ns	
Camera clock cycle	$t_{VCYC}$	—	87	MHz	
Camera clock high level	$t_{VHW}$	$0.4 \times t_{VCYC}$	—	ns	
Camera clock low level	$t_{VLW}$	$0.4 \times t_{VCYC}$	—	ns	
Field identification signal (VIO_FLD) setup time	$t_{VDS}$	2	—	ns	
Field identification signal (VIO_FLD) hold time	$t_{VDH}$	3.5	—	ns	

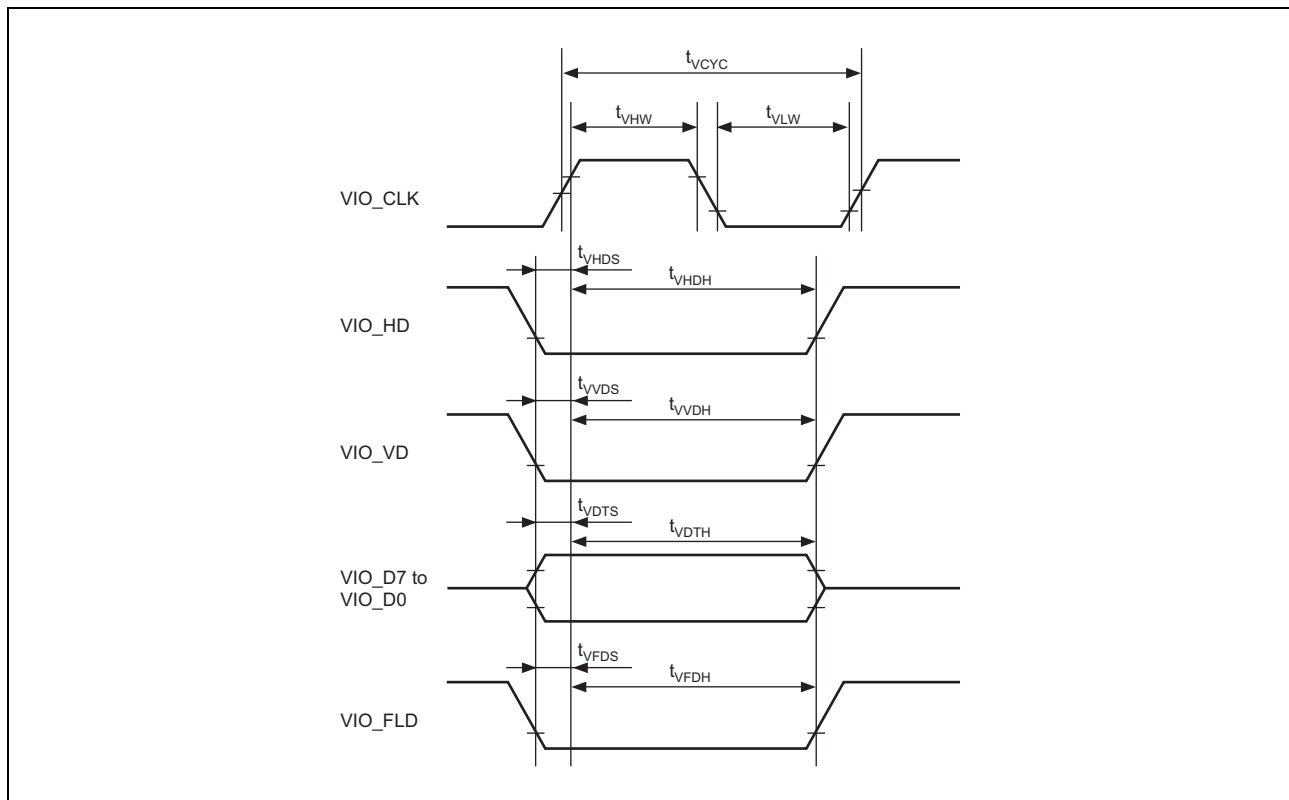


Figure 58.87 Capture Engine Unit Module Signal Timing

## 58.4.22 MMC Host Interface Timing

Table 58.28 MMC Host Interface Timing

Item	Symbol	Min.	Max.	Unit	Figure
MMC_CLK clock cycle	$t_{MMCPP}$	$2 \times t_{p1cyc}$	—	ns	Figure 58.88
MMC_CLK clock high level width	$t_{MMCWH}$	6.5	—	ns	
MMC_CLK clock low level width	$t_{MMCWL}$	6.5	—	ns	
MMC_CLK clock rise time	$t_{MMCCLH}$	—	3	ns	
MMC_CLK clock fall time	$t_{MMCCHL}$	—	3	ns	
MMC_CMD, MMC_D7 to MMC_D0 output data delay time (data transfer mode)	$t_{MMCODLY}$	-6.5	6.5	ns	
MMC_CMD, MMC_D7 to MMC_D0 input data setup time	$t_{MMCSISU}$	6.5	—	ns	
MMC_CMD, MMC_D7 to MMC_D0 input data hold time	$t_{MMCSIH}$	2	—	ns	

Note:  $t_{p1cyc}$  indicates peripheral clock 1 (P1 $\phi$ ) cycle.

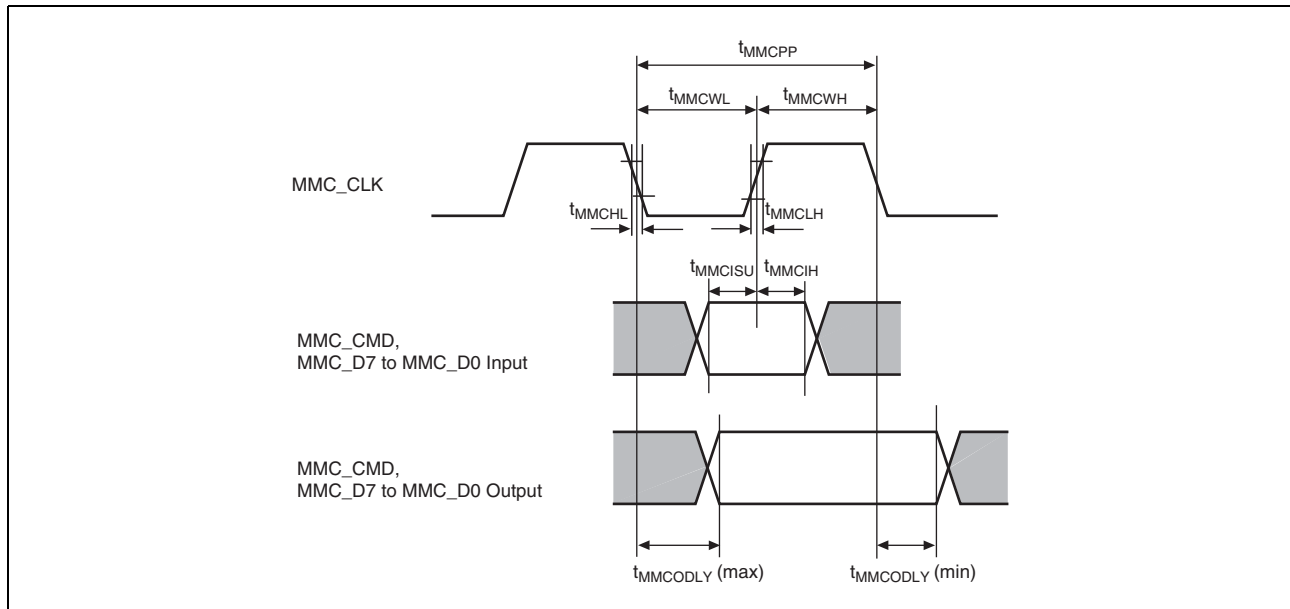


Figure 58.88 MMC Interface

## 58.4.23 General Purpose I/O Ports Timing

Table 58.29 General Purpose I/O Ports Timing

Item	Symbol	Min.	Max.	Unit	Figure
Output data delay time	$t_{\text{PORTD}}$	—	100	ns	Figure 58.89
Input data setup time	$t_{\text{PORTS}}$	100	—		
Input data hold time	$t_{\text{PORTH}}$	100	—		

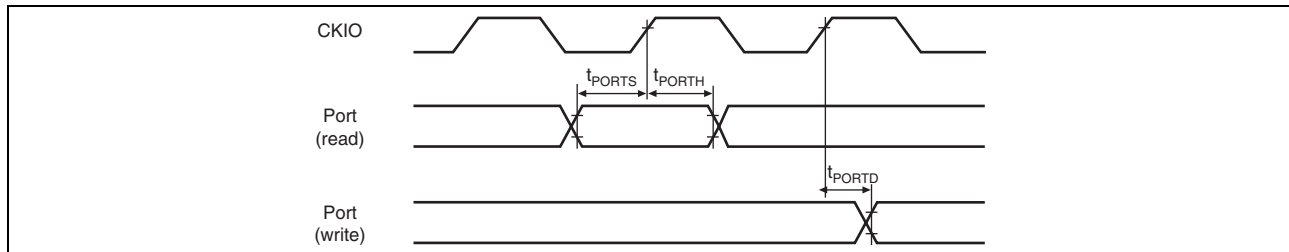


Figure 58.89 General I/O Ports Timing

## 58.4.24 Debugger Interface Timing

Table 58.30 Debugger Interface Timing

Item	Symbol	Min.	Max.	Unit	Figure
TCK cycle time	$t_{\text{TCKcyc}}$	50*1	—	ns	Figure 58.90
TCK high pulse width	$t_{\text{TCKH}}$	0.4	0.6	$t_{\text{TCKcyc}}$	
TCK low pulse width	$t_{\text{TCKL}}$	0.4	0.6	$t_{\text{TCKcyc}}$	
TDI setup time	$t_{\text{TDIS}}$	10	—	ns	Figure 58.91
TDI hold time	$t_{\text{TDIH}}$	10	—	ns	
TMS/SWDIO setup time	$t_{\text{TMSS}}$	10	—	ns	
TMS/SWDIO hold time	$t_{\text{TMSH}}$	10	—	ns	
SWDIO delay time	$t_{\text{SWDO}}$	—	16	ns	
TDO delay time	$t_{\text{TDOD}}$	—	16	ns	
Capture register setup time	$t_{\text{CAPTS}}$	10	—	ns	Figure 58.92
Capture register hold time	$t_{\text{CAPTH}}$	10	—	ns	
Update register delay time	$t_{\text{UPDATED}}$	—	20	ns	
Trace clock cycle	$t_{\text{TCYC}}$	30*2	—	ns	Figure 58.93
Trace clock high level	$t_{\text{THC}}$	12	—	ns	Output load: 15 pF
Trace clock low level	$t_{\text{TLC}}$	12	—	ns	
Trace data delay time	$t_{\text{TDI}}$	3	$0.3 \times t_{\text{TCYC}} + 3$	ns	

Note 1. Should be greater than the peripheral clock 0 (P0 $\phi$ ) cycle time.

Note 2. Generated by dividing the frequency of the peripheral clock (P1 $\phi$ ) by 2.

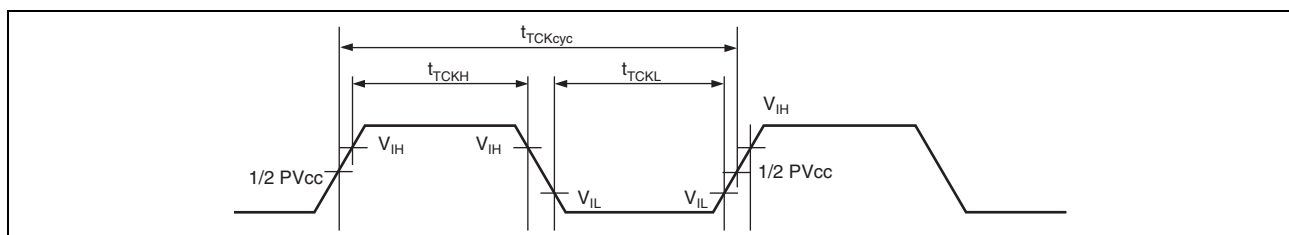


Figure 58.90 TCK Input Timing

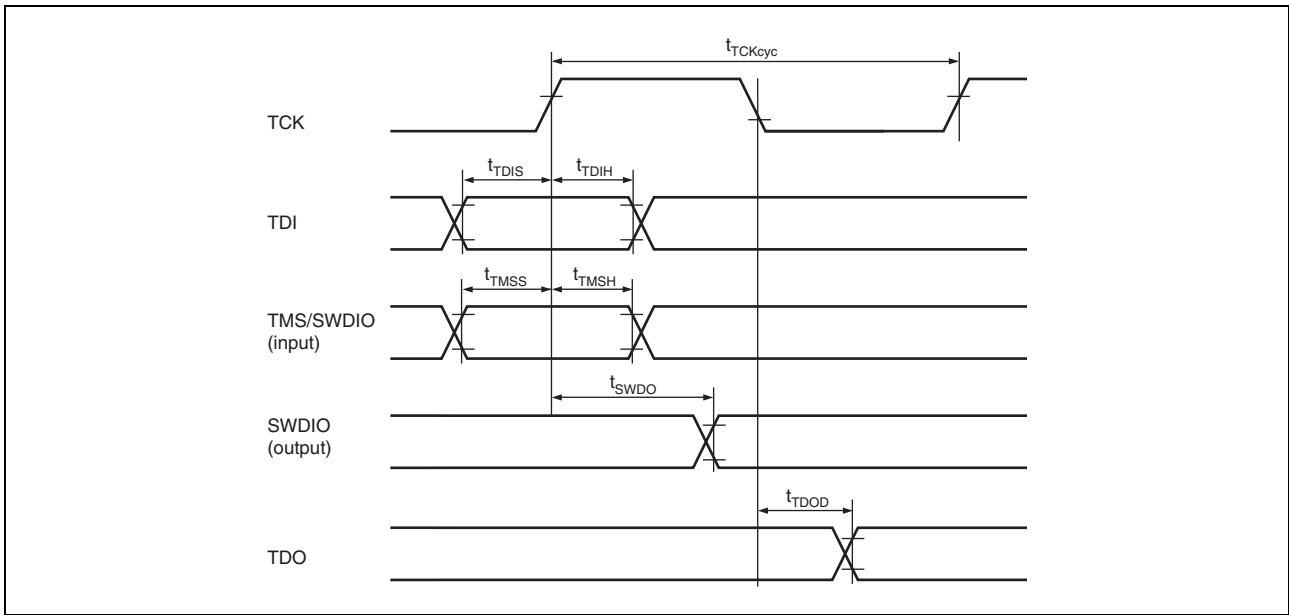


Figure 58.91 Data Transfer Timing

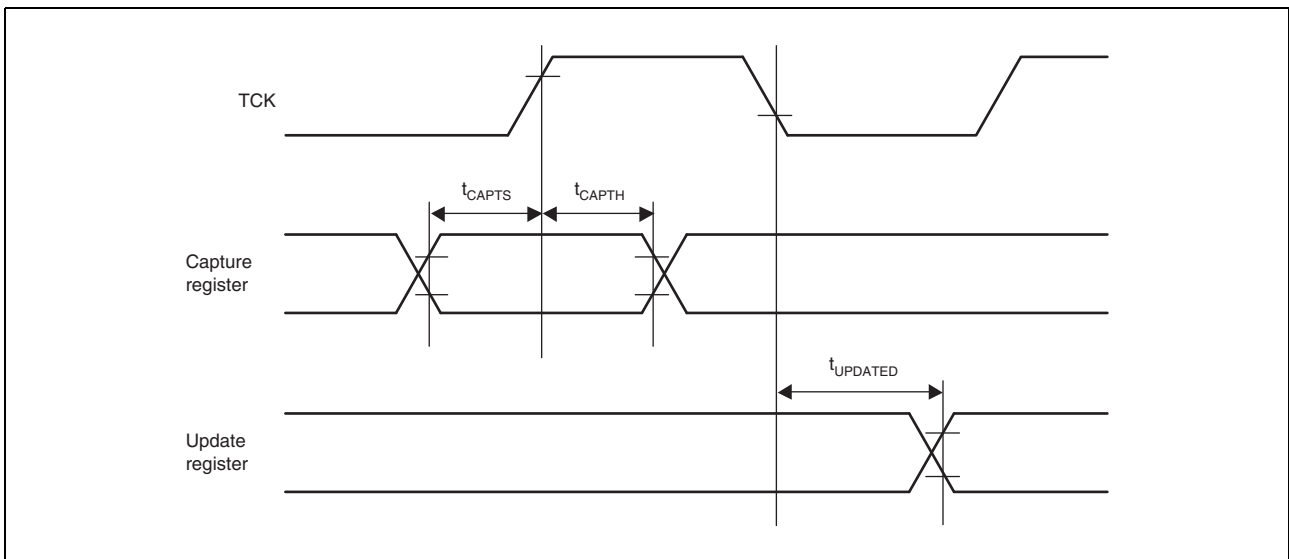


Figure 58.92 Boundary Scan Input/Output I/O Timing

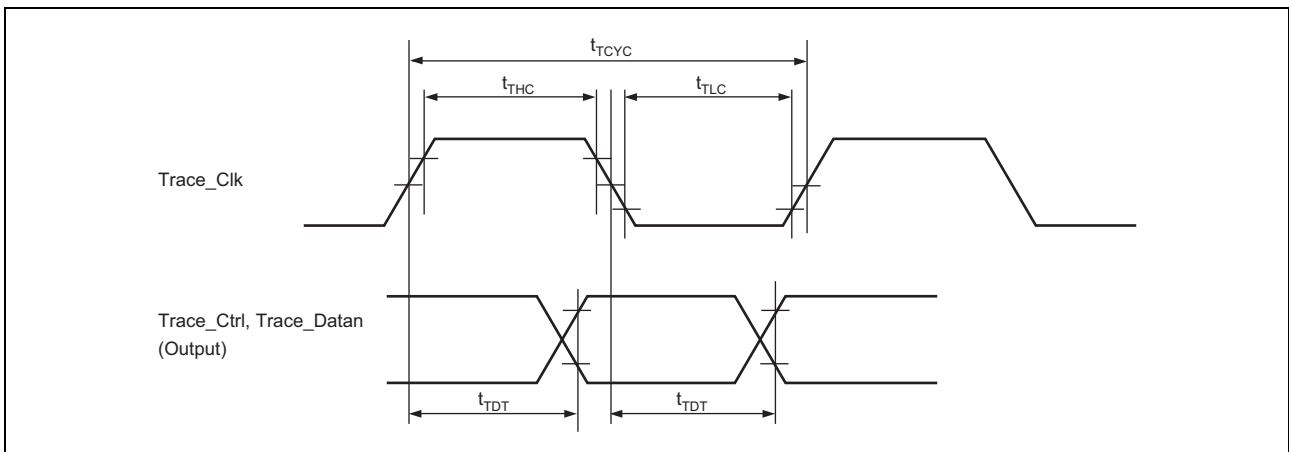


Figure 58.93 Trace Interface Timing



### 58.5 A/D Converter Characteristics

Conditions:  $V_{CC} = USBDV_{CC} = USBUV_{CC} = 1.10$  to  $1.26$  V,  $PV_{CC} = USBDPV_{CC} = 3.0$  to  $3.6$  V,  
 $PLL_{VCC} = 1.10$  to  $1.26$  V,  $AV_{CC} = 3.0$  to  $3.6$  V,  $USBAPV_{CC} = 3.0$  to  $3.6$  V,  $USBAV_{CC} = 1.10$  to  $1.26$  V,  
 $VDAV_{CC} = 3.0$  to  $3.6$  V,  $LVDSAPV_{CC} = 3.0$  to  $3.6$  V,  $LVDSPLL_{VCC} = 1.10$  to  $1.26$  V,  
 $V_{SS} = AV_{SS} = USBDV_{SS} = USBAV_{SS} = USBDPV_{SS} = USBAPV_{SS} = USBUV_{SS} = VDAV_{SS} = LVDSAPV_{SS} =$   
 $0$  V,  
 $T_a = -40$  to  $85$  °C,  $T_j = -40$  to  $125$  °C

Note: Products in BGA packages do not have  $USBDV_{CC}$ ,  $USBUV_{CC}$ ,  $USBDPV_{CC}$ ,  $USBDV_{SS}$ ,  $USBAV_{SS}$ ,  $USBDPV_{SS}$ ,  $USBAPV_{SS}$ ,  $USBUV_{SS}$ , and  $LVDSAPV_{SS}$  pins.

**Table 58.31 A/D Converter Characteristics**

Item		Min.	Typ.	Max.	Unit
Resolution		12	12	12	bits
Conversion time	12-bit	5	—	—	μs
	10-bit				
Analog input capacitance		—	—	20	pF
Permissible signal-source impedance		—	—	3	kΩ
DNL	12-bit	—	—	±1.0	LSB
	10-bit	—	—	±1.0	LSB
INL	12-bit	—	—	±4.0	LSB
	10-bit	—	—	±4.0	LSB
Offset error	12-bit	—	—	±8.0	LSB
	10-bit	—	—	±2.0	LSB
Full-scale error	12-bit	—	—	±8.0	LSB
	10-bit	—	—	±2.0	LSB
Absolute accuracy	12-bit	—	—	±11.0	LSB
	10-bit	—	—	±5.0	LSB



## 58.6 Video Characteristics of A/D Converter for the Input of Video Signals

Conditions:  $V_{CC} = USBDV_{CC} = USBUV_{CC} = 1.10$  to  $1.26$  V,  $PV_{CC} = USBDPV_{CC} = 3.0$  to  $3.6$  V,  $PLL_{VCC} = 1.10$  to  $1.26$  V,  
 $AV_{CC} = 3.0$  to  $3.6$  V,  $USBAPV_{CC} = 3.0$  to  $3.6$  V,  $USBAV_{CC} = 1.10$  to  $1.26$  V,  $VDAV_{CC} = 3.0$  to  $3.6$  V,  
 $LVDSAPV_{CC} = 3.0$  to  $3.6$  V,  $LVDSPLL_{VCC} = 1.10$  to  $1.26$  V,  
 $V_{SS} = AV_{SS} = USBDV_{SS} = USBAV_{SS} = USBDPV_{SS} = USBAPV_{SS} = USBUV_{SS} = VDAV_{SS} = LVDSAPV_{SS} = 0$  V,  
 $T_a = -40$  to  $85$  °C,  $T_j = -40$  to  $125$  °C

Note: Products in BGA packages do not have  $USBDV_{CC}$ ,  $USBUV_{CC}$ ,  $USBDPV_{CC}$ ,  $USBDV_{SS}$ ,  $USBAV_{SS}$ ,  $USBDPV_{SS}$ ,  $USBAPV_{SS}$ ,  $USBUV_{SS}$ , and  $LVDSAPV_{SS}$  pins.

**Table 58.32 Characteristics of A/D Converter for the Input of Video Signals (Reference Voltage)**

Item	Min.	Typ.	Max.	Unit	Test Conditions
Reference voltage (VRP)	—	1.7	—	V	
Reference voltage (VRM)	—	0.9	—	V	

**Table 58.33 Characteristics of A/D Converter for the Input of Video Signals (Clamping)**

Item	Min.	Typ.	Max.	Unit	Test Conditions
Clamping voltage level	—	0.6	—	V	
Sink current	—	4.0	—	μA	
Source current	—	0.2	—	mA	

**Table 58.34 Characteristics of A/D Converter for the Input of Video Signals (LPF)**

Item	Min.	Typ.	Max.	Unit	Test Conditions
Cutoff frequency	- 3	—	—	dB	5.1 MHz

**Table 58.35 Characteristics of A/D Converter for the Input of Video Signals (PGA)**

Item	Min.	Typ.	Max.	Unit	Test Conditions
Number of gain steps	—	64	—	step	
Gain step width	—	0.1	—	dB	
Minimum gain	—	0	—	dB	
Maximum gain	—	6.02	—	dB	

**Table 58.36 Characteristics of A/D Converter for the Input of Video Signals (ADC)**

Item	Min.	Typ.	Max.	Unit	Test Conditions
Resolution	—	10	—	bit	
A/D conversion range	—	1.6	—	V <sub>pp</sub>	(VRP – VRM) × 2
Integral linearity error	—	—	± 5.0	LSB	ADC + PGA f <sub>s</sub> = 27 MHz
Differential linearity error	—	—	± 2.0	LSB	ADC + PGA f <sub>s</sub> = 27 MHz
S/N	—	54*	—	dB	f <sub>in</sub> = 1 MHz, f <sub>s</sub> = 27 MHz PGA_GAIN = 000000
S/(N + D)	—	51*	—	dB	f <sub>in</sub> = 1 MHz, f <sub>s</sub> = 27 MHz PGA_GAIN = 000000
Time for return from power saving mode	—	1	—	ms	

Note: \* Reference value.

## 59. States and Handling of Pins

This section describes pin states in each operating mode and how to handle pins.

### 59.1 Pin States

Table 59.1 shows the pin states in each operating mode.

As for the input/output functions, input buffers are listed on the upper column and output buffers on the lower column.

**Table 59.1 Pin States**

Pin Function				Pin State						
Type	Pin Name			Normal State (Other than States at Right)	Power-On Reset <sup>1</sup>	Pin State Retained <sup>2</sup>		Power-Down State		
						EBUSKEEPE <sup>3</sup> (Other than States at Right)		Power-On Reset <sup>4</sup>	Deep Standby Mode	Software Standby Mode
						0	1			
Clock	EXTAL <sup>6</sup>			I	I	I		I/Z <sup>5</sup>	I	
	XTAL <sup>6</sup>			O	O	O		O/Z <sup>5</sup>	O/Z <sup>5</sup>	
	CKIO	Boot mode	0, 1	O/Z <sup>7</sup>	O	O	O/Z <sup>7</sup>		O/Z <sup>7</sup>	O/Z <sup>7</sup>
			Other than above	O/Z <sup>7</sup>	O	O/Z <sup>7</sup>		O/Z <sup>7</sup>	O/Z <sup>7</sup>	
	AUDIO_CLK			I	-	-		I/Z <sup>12</sup>	I	
	AUDIO_X1 <sup>6</sup>			I	I	I		Z	Z	
	AUDIO_X2 <sup>6</sup>			O	O	O		L	L	
	AUDIO_XOUT (P5_5)			O	-	Z		Z	L/Z <sup>9</sup>	
AUDIO_XOUT (other than P5_5), AUDIO_XOUT2, AUDIO_XOUT3			O	-	O/Z <sup>9*15</sup>		O/Z <sup>9*15</sup>	L/Z <sup>9</sup>		
System control	RES			I	I	I		I	I	
Operating mode control	MD_BOOT2, MD_BOOT1, MD_BOOT0, MD_CLK, MD_CLKS			-	I	-		-	-	
	BSCANP			I	I	I		I	I	
Interrupts	NMI			I	I	I		I	I	
	IRQ0 (P8_2, P9_1), IRQ1 (P2_15, P7_8), IRQ2 (P5_9), IRQ3 (P6_4), IRQ4 (P3_3), IRQ5 (P8_7), IRQ6 (P2_12, P3_1, P3_9), IRQ7 (P6_2)			I	-	-		I/Z <sup>12</sup>	I	
	IRQ0 (other than P8_2 and P9_1), IRQ1 (other than P2_15 and P7_8), IRQ2 (other than P5_9), IRQ3 (other than P6_4), IRQ4 (other than P3_3), IRQ5 (other than P8_7), IRQ6 (other than P2_12, P3_1, and P3_9), IRQ7 (other than P6_2)			I	-	-		Z	I	

Table 59.1 Pin States

Type	Pin Function			Pin State						
				Normal State (Other than States at Right)	Power-On Reset <sup>*1</sup>	Pin State Retained <sup>*2</sup>		Power-Down State		
						EBUSKEEPE <sup>*3</sup> (Other than States at Right)		Power-On Reset <sup>*4</sup>	Deep Standby Mode	Software Standby Mode
Pin Name		0	1							
Bus state controller	A25 to A21, A0			O	-	O/Z <sup>*10</sup>		O/Z <sup>*10</sup>	O/Z <sup>*10</sup>	
	A20 to A2	Boot mode	0, 1	O	Z	O	O/Z <sup>*10</sup>	O/Z <sup>*10</sup>	O/Z <sup>*10</sup>	
			Other than above	O	-	O/Z <sup>*10</sup>		O/Z <sup>*10</sup>	O/Z <sup>*10</sup>	
	A1	Boot mode	0	O	Z	O	O/Z <sup>*10</sup>	O/Z <sup>*10</sup>	O/Z <sup>*10</sup>	
			Other than above	O	-	O/Z <sup>*10</sup>		O/Z <sup>*10</sup>	O/Z <sup>*10</sup>	
	D0, D1, D3, D5 to D15	Boot mode	0, 1	I/Z	Z	I/Z		Z	Z	Z
				O/Z	Z	O/Z	Z	Z	Z	
			Other than above	I/Z	-	-		Z	Z	
				O/Z	-	Z		Z	Z	
	D2, D4	Boot mode	0, 1	I/Z	Z	I/Z		Z	I/Z <sup>*12</sup>	I
				O/Z	Z	O/Z	Z	Z	Z	
			Other than above	I/Z	-	-		I/Z <sup>*12</sup>	I	
				O/Z	-	Z		Z	Z	
	D16 to D27, D29, D30	Boot mode	1	I/Z	Z	I/Z		Z	Z	Z
				O/Z	Z	O/Z	Z	Z	Z	
			Other than above	I/Z	-	-		Z	Z	
				O/Z	-	Z		Z	Z	
D28, D31	Boot mode	1	I/Z	Z	I/Z		Z	I/Z <sup>*12</sup>	I	
			O/Z	Z	O/Z	Z	Z	Z		
		Other than above	I/Z	-	-		I/Z <sup>*12</sup>	I		
			O/Z	-	Z		Z	Z		
CS0, RD	Boot mode	0, 1	O	Z	O	H/Z <sup>*10</sup>	H/Z <sup>*10</sup>	H/Z <sup>*10</sup>		
		Other than above	O	-	H/Z <sup>*10</sup>		H/Z <sup>*10</sup>	H/Z <sup>*10</sup>		
CS5 to CS1, RD/WR, BS, WE3/AH/DQMUU, WE2/DQMUL, WE1/DQMLU/WE, WE0/DQMLL			O	-	H/Z <sup>*10</sup>		H/Z <sup>*10</sup>	H/Z <sup>*10</sup>		
WAIT			I	-	-		Z	Z		
RAS, CAS, CKE			O	-	O/Z <sup>*11</sup>		O/Z <sup>*11</sup>	O/Z <sup>*11</sup>		
Direct memory access controller	DREQ0			I	-	-		Z	Z	
	DACK0, TEND0			O	-	O/Z <sup>*9</sup>		O/Z <sup>*9</sup>	O/Z <sup>*9</sup>	
Multi-function timer pulse unit 2	TCLKA (P6_2)			I	-	-		I/Z <sup>*12</sup>	I	
	TCLKA (other than P6_2), TCLKB, TCLKC, TCLKD			I	-	-		Z	Z	
	TIOC1B (P2_12), TIOC2A (P6_2), TIOC3A (P7_8), TIOC4B (P3_9)	I			-	-		I/Z <sup>*12</sup>	I	
		O/Z			-	O/Z <sup>*9</sup>		O/Z <sup>*9</sup>	O/Z <sup>*9</sup>	
	TIOC0A (P5_0), TIOC0B (P5_1), TIOC0C (P5_5), TIOC0D (P5_7), TIOC1B (P5_2), TIOC3C (P5_3), TIOC3D (P5_4)	I			-	-		Z	Z	
		O/Z			-	Z		Z	O/Z <sup>*9</sup>	
TIOC0A (other than P5_0), TIOC0B (other than P2_12 and P5_1), TIOC0C (other than P5_5), TIOC0D (other than P5_7), TIOC1A, TIOC1B (other than P5_2), TIOC2A, TIOC2B, TIOC3A (other than P7_8), TIOC3B, TIOC3C (other than P5_3), TIOC3D (other than P5_4), TIOC4A, TIOC4B (other than P3_9), TIOC4C, TIOC4D			I	-	-		Z	Z		
O/Z			-	O/Z <sup>*9</sup>		O/Z <sup>*9</sup>	O/Z <sup>*9</sup>			
Watchdog timer	WDTOVF			O	-	H		H	H	

Table 59.1 Pin States

Pin Function				Pin State						
Type	Pin Name			Normal State (Other than States at Right)	Power-On Reset <sup>1</sup>	Pin State Retained <sup>2</sup>		Power-Down State		
						EBUSKEEPE <sup>3</sup> (Other than States at Right)		Power-On Reset <sup>4</sup>	Deep Standby Mode	Software Standby Mode
						0	1			
Realtime clock	RTC_X1 <sup>6</sup>			I/Z <sup>13</sup>	I	I/Z <sup>13</sup>		I/Z <sup>13</sup>	I/Z <sup>13</sup>	
	RTC_X2 <sup>6</sup>			O/Z <sup>13</sup>	O	O/Z <sup>13</sup>		O/Z <sup>13</sup>	O/Z <sup>13</sup>	
	RTC_X3			I/Z <sup>13</sup>	-	I/Z <sup>13</sup>		I/Z <sup>13</sup>	I/Z <sup>13</sup>	
	RTC_X4			O/Z <sup>13</sup>	-	O/Z <sup>13</sup>		O/Z <sup>13</sup>	O/Z <sup>13</sup>	
Serial communication interface with FIFO	TxD3 (P5_3), TxD4 (P5_0), TxD6 (P5_6)			O/Z	-	Z		Z	O/Z <sup>9</sup>	
	TxD0 to TxD2, TxD3 (other than P5_3), TxD4 (other than P5_0), TxD5, TxD6 (other than P5_6), TxD7			O/Z	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>	
	RxD0 (P2_15), RxD2 (P6_2), RxD5 (P8_2)			I	-	-		I/Z <sup>12</sup>	I	
	RxD0 (other than P2_15), RxD1, RxD2 (other than P6_2), RxD3, RxD4, RxD5 (other than P8_2), RxD6, RxD7			I	-	-		Z	Z	
	RTS5 (P6_4)			I	-	-		I/Z <sup>12</sup>	I	
				O/Z	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>	
	SCK3 (P5_2)			I	-	-		Z	Z	
				O/Z	-	Z		Z	O/Z <sup>9</sup>	
SCK0 to SCK2, SCK3 (other than P5_2), SCK4 to SCK7, RTS1, RTS5 (other than P6_4), RTS7, CTS1, CTS5, CTS7			I	-	-		Z	Z		
			O/Z	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>		
Serial communications interface	SCI_TXD0, SCI_TXD1			O/Z	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>	
	SCI_RXD0, SCI_RXD1			I	-	-		Z	Z	
	SCI_CTS1/RTS1			I	-	-		I/Z <sup>12</sup>	I	
				O/Z	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>	
	SCI_SCK0, SCI_SCK1, SCI_CTS0/RTS0			I	-	-		Z	Z	
O/Z				-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>		
Renesas serial peripheral interface	RSPCK0 (P2_12), RSPCK1 (P6_4), SSL30 (P3_1), MISO0 (P2_15, P8_2), MISO2 (P9_1), MISO3 (P3_3)			I	-	-		I/Z <sup>12</sup>	I	
				O/Z	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>	
	RSPCK3 (P5_0), SSL30 (P5_1), MOSI3 (P5_2), MISO3 (P5_3)			I	-	-		Z	Z	
				O/Z	-	Z		Z	O/Z <sup>9</sup>	
	RSPCK0 (other than P2_12), RSPCK1 (other than P6_4), RSPCK2, RSPCK3 (other than P5_0), RSPCK4, SSL00, SSL10, SSL20, SSL40, MOSI0 to MOSI2, MOSI3 (other than P5_2), MOSI4, MISO0 (other than P2_15), MISO1, MISO2 (other than P9_1), MISO4			I	-	-		Z	Z	
O/Z				-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>		
SPI multi I/O bus controller	SPBCLK_0, SPBSSL_0	Boot mode	2, 3	O/Z	-	-	O/Z <sup>9</sup>	O/Z <sup>9</sup>	O/Z <sup>9</sup>	
			Other than above	O/Z	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>	
	SPBMO0_0/SPBIO00_0, SPBMO0_0/SPBIO10_0			I	-	-		Z	Z	
		Boot mode	2, 3	O/Z	-	-	O/Z <sup>9</sup>	O/Z <sup>9</sup>	O/Z <sup>9</sup>	
			Other than above	O/Z	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>	
	SPBMO1_0/SPBIO01_0 (P2_12), SPBIO31_0			I	-	-		I/Z <sup>12</sup>	I	
				O/Z	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>	
	SPBIO20_0, SPBIO30_0, SPBMO1_0/SPBIO01_0 (other than P2_12), SPBMO1_0/SPBIO11_0, SPBIO21_0			I	-	-		Z	Z	
				O/Z	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>	
	SPBCLK_1, SPBSSL_1			O/Z	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>	
SPBMO0_1/SPBIO00_1, SPBMO1_1/SPBIO01_1, SPBIO20_1, SPBIO30_1, SPBMO1_1/SPBIO01_1, SPBMO1_1/SPBIO11_1, SPBIO21_1, SPBIO31_1			I	-	-		Z	Z		
			O/Z	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>		
I <sup>2</sup> C bus interface	RIIC0SCL to RIIC3SCL, RIIC0SDA to RIIC3SDA			I	-	-		Z	Z	
				L/Z	-	Z		Z	Z	

Table 59.1 Pin States

Pin Function		Pin State						
Type	Pin Name	Normal State (Other than States at Right)	Power-On Reset <sup>*1</sup>	Pin State Retained <sup>*2</sup>		Power-Down State		
				EBUSKEEPE <sup>*3</sup> (Other than States at Right)		Power-On Reset <sup>*4</sup>	Deep Standby Mode	Software Standby Mode
				0	1			
Serial sound interface	SSITxD0, SSITxD1, SSITxD3, SSITxD5	O	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>	
	SSIRxD0, SSIRxD1, SSIRxD3, SSIRxD5	I	-	-		Z	Z	
	SSIDATA2, SSIDATA4	I	-	-		Z	Z	
		O/Z	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>	
	SSISCK3 (P7_8)	I	-	-		I/Z <sup>12</sup>	I	
		O/Z	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>	
SSISCK0 to SSISCK2, SSISCK3 (other than P7_8), SSISCK4, SSISCK5, SSIWS0 to SSIWS5	I	-	-		Z	Z		
	O/Z	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>		
Media local bus	MLB_CLK	O/Z	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>	
	MLB_DAT, MLB_SIG	I	-	-		Z	Z	
O/Z		-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>		
CAN interface	CAN0TX to CAN4TX	O	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>	
	CAN0RX, CAN1RX (P5_9), CAN2RX (P6_4), CAN3RX (P2_12), CAN_CLK (P2_15)	I	-	-		I/Z <sup>12</sup>	I	
		I	-	-		Z	Z	
IEBus controller	IETxD	O	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>	
	IERxD (P5_9)	I	-	-		I/Z <sup>12</sup>	I	
	IERxD (other than P5_9)	I	-	-		Z	Z	
Renesas SPDIF interface	SPDIF_OUT (P5_7)	O	-	Z		Z	O/Z <sup>9</sup>	
	SPDIF_OUT (other than P5_7)	O	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>	
	SPDIF_IN	I	-	-		Z	Z	
LIN interface	RLIN30TX, RLIN31TX	O	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>	
	RLIN30RX	I	-	-		Z	Z	
	RLIN31RX	I	-	-		I/Z <sup>12</sup>	I	
Ethernet controller	ET_TXER, ET_TXEN, ET_TXD3 to ET_TXD0, ET_MDC	O	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>	
	ET_TXCLK, ET_RXCLK, ET_RXDV, ET_RXER, ET_RXD3 to ET_RXD0, ET_CRSD, ET_COL (other than P8_7)	I	-	-		Z	Z	
	ET_COL (P8_7)	I	-	-		I/Z <sup>12</sup>	I	
	ET_MDIO (P3_3)	I	-	-		I/Z <sup>12</sup>	I	
		O	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>	
	ET_MDIO (other than P3_3)	I	-	-		Z	Z	
O		-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>		
A/D converter	AN7 to AN0	I	-	-		Z	Z	
	ADTRG	I	-	-		Z	Z	
NAND flash memory controller	FRB	I	-	-		Z	Z	
	FCE	O	-	Z		Z	O/Z <sup>9</sup>	
	FALE, F $\overline{R}$ E, FCLE, F $\overline{W}$ E	O	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>	
	NAF1	I	-	-		I/Z <sup>12</sup>	I	
		O/Z	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>	
	NAF0, NAF2 to NAF7	I	-	-		Z	Z	
O/Z		-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>		

Table 59.1 Pin States

Pin Function		Pin State						
Type	Pin Name	Normal State (Other than States at Right)	Power-On Reset <sup>1</sup>	Pin State Retained <sup>2</sup>		Power-Down State		
				EBUSKEEPER <sup>3</sup> (Other than States at Right)		Power-On Reset <sup>4</sup>	Deep Standby Mode	Software Standby Mode
				0	1			
USB2.0 host/function module	DP0, DP1, DM0, DM1	I/Z	Z	I/Z		Z	I/Z	
		O/Z	Z	O/Z		Z	O/Z	
	VBUS0, VBUS1	I	I	I		I	I	
	REFRIN	I	I	I		I	I	
	USB_X1 <sup>6</sup>	I	I	I		Z	Z	
	USB_X2 <sup>6</sup>	O	O	O		L	L	
Digital video decoder	VIDEO_X1 <sup>6</sup>	I	I	I		Z	Z	
	VIDEO_X2 <sup>6</sup>	O	O	O		L	L	
	VIN1A, VIN2A, VIN1B, VIN2B	I	I	I		I	I	
Video display controller 5	LCD0_CLK, LCD1_CLK	O	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>	
	LCD0_DATA16 (P5_0), LCD0_DATA17 (P5_1), LCD0_DATA18 (P5_2), LCD0_DATA19 (P5_3), LCD0_DATA20 (P5_4), LCD0_DATA21 (P5_5), LCD0_DATA22 (P5_6), LCD0_DATA23 (P5_7), LCD1_DATA0 to LCD1_DATA7	O	-	Z		Z	O/Z <sup>9</sup>	
	LCD0_DATA0 to LCD0_DATA15, LCD0_DATA16 (other than P5_0), LCD0_DATA17 (other than P5_1), LCD0_DATA18 (other than P5_2), LCD0_DATA19 (other than P5_3), LCD0_DATA20 (other than P5_4), LCD0_DATA21 (other than P5_5), LCD0_DATA22 (other than P5_6), LCD0_DATA23 (other than P5_7), LCD1_DATA8 to LCD1_DATA23, LCD0_TCON0 to LCD0_TCON6, LCD1_TCON0 to LCD1_TCON6	O	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>	
	LCD0_EXTCLK, LCD1_EXTCLK	I	-	-		Z	Z	
	DV0_DATA12 (P2_12), DV0_DATA15 (P2_15), DV0_DATA18 (P6_2), DV0_DATA20 (P6_4), DV1_DATA4, DV0_VSYNC (P5_9)	I	-	-		I/Z <sup>12</sup>	I	
	DV0_CLK, DV1_CLK, DV0_DATA0 to DV0_DATA11, DV0_DATA12 (other than P2_12), DV0_DATA13, DV0_DATA14, DV0_DATA15 (other than P2_15), DV0_DATA16, DV0_DATA17, DV0_DATA18 (other than P6_2), DV0_DATA19, DV0_DATA20 (other than P6_4), DV0_DATA21 to DV0_DATA23, DV1_DATA0 to DV1_DATA3, DV1_DATA5 to DV1_DATA23, DV0_VSYNC (other than P5_9), DV1_VSYNC, DV0_HSYNC, DV1_HSYNC	I	-	-		Z	Z	
LVDS output interface	TXOUT0M, TXOUT0P, TXOUT1M, TXOUT1P, TXOUT2M, TXOUT2P, TXCLKOUTM, TXCLKOUTP	O	-	Z		Z	O/Z <sup>9</sup>	
Capture engine unit	VIO_CLK, VIO_VD, VIO_HD, VIO_FLD, VIO_D0 to VIO_D15	I	-	-		Z	Z	
Sound Generator	SGOUT_0 to SGOUT_3	O	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>	

Table 59.1 Pin States

Pin Function				Pin State						
Type	Pin Name			Normal State (Other than States at Right)	Power-On Reset <sup>1</sup>	Pin State Retained <sup>2</sup>		Power-Down State		
						EBUSKEEPE <sup>3</sup> (Other than States at Right)		Power-On Reset <sup>4</sup>	Deep Standby Mode	Software Standby Mode
						0	1			
SD host interface	SD_CLK_0 (P4_12)	Boot mode	4	O	-	-	O/Z <sup>9</sup>	O/Z <sup>9</sup>	O/Z <sup>9</sup>	
			Other than above	O	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>	
	SD_CLK_0 (other than P4_12)			O	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>	
	SD_CMD_0 (P4_13), SD_D0_0 (P4_11), SD_D1_0 (P4_10), SD_D2_0 (P4_15), SD_D3_0 (P4_14)			I	-	-		Z	Z	
		Boot mode	4	O	-	-	O/Z <sup>9</sup>	O/Z <sup>9</sup>	O/Z <sup>9</sup>	
			Other than above	O/Z	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>	
	SD_CMD_0 (other than P4_13), SD_D0_0 (other than P4_11), SD_D1_0 (other than P4_10), SD_D2_0 (other than P4_15), SD_D3_0 (other than P4_14)			I	-	-		Z	Z	
				O/Z	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>	
	SD_CLK_1			O/Z	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>	
	SD_CMD_1, SD_D0_1 to SD_D3_1			I	-	-		Z	Z	
			O/Z	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>		
SD_CD_0, SD_CD_1, SD_WP_0			I	-	-		Z	Z		
SD_WP_1			I	-	-		I/Z <sup>12</sup>	I		
MMC host interface	MMC_CLK (P3_12)	Boot mode	5	O	-	-	O/Z <sup>9</sup>	O/Z <sup>9</sup>	O/Z <sup>9</sup>	
			Other than above	O	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>	
	MMC_CLK (other than P3_12)			O	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>	
	MMC_CMD (P3_13), MMC_D0 (P3_11), MMC_D1 (P3_10), MMC_D2 (P3_15), MMC_D3 (P3_14)			I	-	-		Z	Z	
		Boot mode	5	O/Z	-	-	O/Z <sup>9</sup>	O/Z <sup>9</sup>	O/Z <sup>9</sup>	
			Other than above	O/Z	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>	
	MMC_CMD (other than P3_13), MMC_D0 (other than P3_11), MMC_D1 (other than P3_10), MMC_D2 (other than P3_15), MMC_D3 (other than P3_14), MMC_D4 to MMC_D7			I	-	-		Z	Z	
			O/Z	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>		
MMC_CD			I	-	-		Z	Z		
Motor control PWM timer	PWM1A to PWM1H, PWM2A to PWM2H			O	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>	
Ports	JP0_0, JP0_1, P0_0 to P0_3			I/Z	-	-		Z	Z	
	P0_4, P0_5, P1_8 to P1_15			I/Z	Z	Z		Z	Z	
	P1_0 to P1_7			I/Z <sup>8</sup>	Z	Z		Z	Z	
				L/Z	Z	Z		Z	Z	
	P2_12, P2_15, P3_1, P3_3, P3_9, P5_9, P6_2, P6_4, P7_8, P8_2, P8_7, P9_1			I	Z	Z		I/Z <sup>12</sup>	I	
				O/Z	Z	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>	
	P5_0 to P5_7			I/Z <sup>8</sup>	Z	Z		Z	Z	
				O/Z	Z	Z		Z	Z	
Other than above			I/Z <sup>8</sup>	Z	Z		Z	Z		
			O/Z	Z	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>		
Debugger interface	TDI			I	I	I		Z	I	
	TDO			O/Z <sup>14</sup>	O/Z <sup>14</sup>	O/Z <sup>14</sup>		Z	O/Z <sup>14</sup>	
	TMS/SWDIO			I	I	I		Z	I	
				O/Z	O/Z	O/Z		Z	O/Z	
	TCK/SWDCLK			I	I	I		Z	I	
	TRST			I	I	I		Z	I	
TRACEDATA3 to TRACEDATA0, TRACECLK, TRACECTL			O	-	O/Z <sup>9</sup>		O/Z <sup>9</sup>	O/Z <sup>9</sup>		

## [Legend]

I:	Input
O:	Output
H:	High-level output
L:	Low-level output
Z:	High-impedance

- Note 1. Indicates the power-on reset by low-level input to the  $\overline{\text{RES}}$  pin. The pin states after a power-on reset by the watchdog timer overflow is the same as the initial pin states at normal operation (see section 54, Ports).
- Note 2. After the chip has been released from deep standby mode by the input on pins for canceling the standby mode such as NMI, the pins retain the state until the IOKEEP bit in the deep standby cancel source flag register (DSFR) is cleared (see section 55, Power-Down Modes).
- Note 3. The EBUSKEEPE bit in deep standby control register (DSTCR) (see section 55, Power-Down Modes).
- Note 4. This LSI enters the power-on reset state for a certain period after recovery from deep standby control mode (see section 55, Power-Down Modes).
- Note 5. Depends on the setting of the RCKSEL bit in the realtime clock control register 5 (RCR5) (see section 13, Realtime Clock).
- Note 6. When pins for the connection with a crystal resonator are not used, the input pins (EXTAL, RTC\_X1, RTC\_X3, AUDIO\_X1, USB\_X1, and VIDEO\_X1) must be fixed (pull-up/down resistor, power supply, or ground.) and the output pins (XTAL, RTC\_X2, RTC\_X4, AUDIO\_X2, USB\_X2, and VIDEO\_X2) must be open.
- Note 7. Depends on the setting of the CKOEN bit in the frequency control register (FRQCR) of the clock pulse generator (see section 6, Clock Pulse Generator).
- Note 8. Depends on the setting of the PIPCnm bit in the port IP control register (PIPCn) of the general I/O ports.
- Note 9. Depends on the setting of the HIZ bit in the standby control register 2 (STBCR2) (see section 55, Power-Down Modes).
- Note 10. Depends on the setting of the HIZMEM bit in the common control register (CMNCR) of the bus state controller (see section 8, Bus State Controller).
- Note 11. Depends on the setting of the HIZCNT bit in the common control register (CMNCR) of the bus state controller (see section 8, Bus State Controller).
- Note 12. Depends on the setting of the corresponding bit in the deep standby cancel source select register (DSSSR) and the RCKSEL bit in the realtime clock control register 5 (RCR5) (see section 55, Power-Down Modes).
- Note 13. Depends on the setting of the RTCEN bit in the realtime clock control register 2 (RCR2) and the RCKSEL bit in the realtime clock control register 5 (RCR5) (see section 13, Realtime Clock).
- Note 14. Z when the TAP controller is neither the Shift-DR nor Shift-IR state.
- Note 15. When this is an output, the output is fixed to either the High or Low level. There is no oscillation.



## 59.2 Treatment of Unused Pins

How unused pins are to be handled is indicated below.

**Table 59.2 Handling of Unused Pins (Except for Debugger Interface Pins)**

Pin Name	Handling
NMI	Fix this pin at a high level (pull up or connect to a power supply).
DP1, DP0, DM1, DM0, VBUS1, VBUS0	- Connect these pins to USBDPVss (QFP package). - Connect them to Vss (BGA package).
REFRIN	Connect this pin, via a $5.6\text{ k}\Omega \pm 20\%$ resistor, to USBAPVcc.
1.18-V power dedicated to the USB (USBAPVcc, USBDVcc, USBUVcc)	Supply power at 1.18 V. Note: Products in BGA packages do not have pins USBDVcc and USBUVcc.
3.3-V power dedicated to the USB (USBAPVcc, USBDPVcc)	Supply power at 3.3 V. Note: Products in BGA packages do not have the USBDPVcc pin.
Dedicated USB ground (USBAPVss, USBDPVss, USBAPVss, USBDVss, USBUVss)	Connect to ground. Note: Products in BGA packages do not have the dedicated USB ground pins.
AVref	Connect this pin to AVcc.
Dedicated A/D power (AVcc)	Supply power at 3.3 V.
Dedicated A/D ground (AVss)	Connect to ground.
VIN1A, VIN2A, VIN1B, VIN2B, VRP, VRM, REXT	Open-circuit
Dedicated A/D power for input of video signals (VDAVcc)	Supply power at 3.3 V.
Dedicated A/D ground for input of video signals (VDAVss)	Connect to ground.
LVDSREFRIN	Open-circuit
LVDS analog power supply (LVDSAPVcc)	Supply power at 3.3 V.
LVDS PLL power supply (LVDSAPVss)	Supply power at 1.18 V.
LVDS analog ground (LVDSAPVss)	Connect to ground.
Dedicated input pins other than those listed above	Fix the level on the pins (pull them up or down, or connect them to the power supply or ground level).
Input/output pins other than those listed above	Make the input-pin settings and then fix the level (pull them up or down); alternatively, make the output-pin settings and leave the pins open-circuit.
Dedicated output pins	Open-circuit

Note: We recommend that the values of pull-up or pull-down resistors are in the range from  $4.7\text{ k}\Omega$  to  $100\text{ k}\Omega$ .

**Table 59.3 Handling of Debugger Interface Pins (when Emulator is not Used)**

Pin	Handling
BSCANP	Fix this pin at a low level (pull down or connect to the ground level).
TRST	Fix this pin at a low level (pull down or connect to the ground level). Or connect to another pin which operates in the same manner as the RES pin
TCK, TMS, TDI	Fix the level on the pins (pull them up or down, or connect them to the power supply or ground level).
TDO	Open-circuit

Note 1. When using the emulator, handle these pins as described in the manual for the emulator.

Note 2. We recommend that the values of pull-up or pull-down resistors are in the range from  $4.7\text{ k}\Omega$  to  $100\text{ k}\Omega$ .

### 59.3 Handling of Pins in Deep Standby Mode

How pins are to be handled in deep standby mode is indicated below.

For the states of pins in deep standby mode, refer to the corresponding items under section 59.1, Pin States. Handling of unused pins as described under section 59.2, Treatment of Unused Pins, also applies in deep standby mode.

**Table 59.4 Handling of Pins in Deep Standby Mode**

Pin	Handling
1.2-V power (Vcc, USBDVcc, USBUVcc, USBAVcc, PLLVcc, LVDSPLLvcc)	Supply power at 1.2 V. Note: Products in BGA packages do not have pins USBDVcc and USBUVcc.
3.3-V power (PVcc, AVcc, USBDPVcc, USBAPVcc, VDAVcc, LVDSAPVcc)	Supply power at 3.3 V. Note: Products in BGA packages do not have the USBDPVcc pin.
Ground (Vss, USBDVss, USBUVss, USBAVss, AVss USBDPVss, USBAPVss, VDAVss, LVDSAPVss)	Connect to ground. Note: Products in BGA packages do not have the USBDVss, USBAVss, USBUVss, USBDPVss, USBAPVss, and LVDSAPVss pins.
VBUS1, VBUS0	Fix the level on this pin (pull it up or down, or connect it to the power supply or ground level) or open circuit. However, note that current as indicated in table 58.2, DC Characteristics (2) [Current Consumption] will be drawn by the pin fixed to the high level.
REFRIN	Connect this pin to USBAPVss via 5.6 kΩ ± 1% resistor (QFP package). Connect this pin to Vss via 5.6 kΩ ± 1% resistor (BGA package).
DP1, DP0, DM1, DM0	Fix the level on the pins (pull them up or down, or connect them to the power supply or ground level) or open circuit.
AVref	Fix the level on this pin (from 3.0 V to AVcc)
REXT	Connect this pin to VDAVss via 22 kΩ ± 1% resistor.
VRP, VRM	Connect these pins to VDAVss via 0.1-μF resistor.
LVDSREFRIN	Connect this pin to LVDSAPVss via 5.6 kΩ ± 1% resistor (QFP package). Connect this pin to Vss via 5.6 kΩ ± 1% resistor (BGA package).
VIN1A, VIN1B, VIN2A, VIN2B	Fix the level on the pins (pull them up or down, or connect them to the power supply or ground level) or open circuit.
EXTAL, RTC_X1, AUDIO_X1, USB_X1, VIDEO_X1	Connect the pins to the crystal oscillator or the clock-input signal, or to a fixed level (pull them up or down, or connect them to the power supply or ground level)
XTAL, RTC_X2, AUDIO_X2, USB_X2, VIDEO_X2	Connect the pins to the crystal oscillator or open circuit.
Dedicated input pins other than those listed above	Fix the level on the pins (pull them up or down, or connect them to the power supply or ground level).
Input/output pins (other than those listed above) in the input state	Fix the level on the pins (pull them up or down).
Input/output pins (other than those listed above) in the high-impedance state	Fix the level on the pins (pull them up or down) or open circuit.
Input/output pins (other than those listed above) in the output state	Open-circuit
Dedicated output pins other than those listed above	Open-circuit

Note: We recommend that the values of pull-up or pull-down resistors are in the range from 4.7 kΩ to 100 kΩ.

### 59.4 Recommended Combination of Bypass Capacitor

Mount a multilayer ceramic capacitor between a pair of the power supply pins as a bypass capacitor. These capacitors must be placed as close as the power supply pins of the LSI. The capacitance of the capacitors should be used 0.1  $\mu\text{F}$  to 0.33  $\mu\text{F}$  (recommended values). For details of the capacitor related to the crystal resonator, see section 6, Clock Pulse Generator.

Figure 59.1 is an example of the externally allocated capacitor in the products in 256-pin QFP packages.

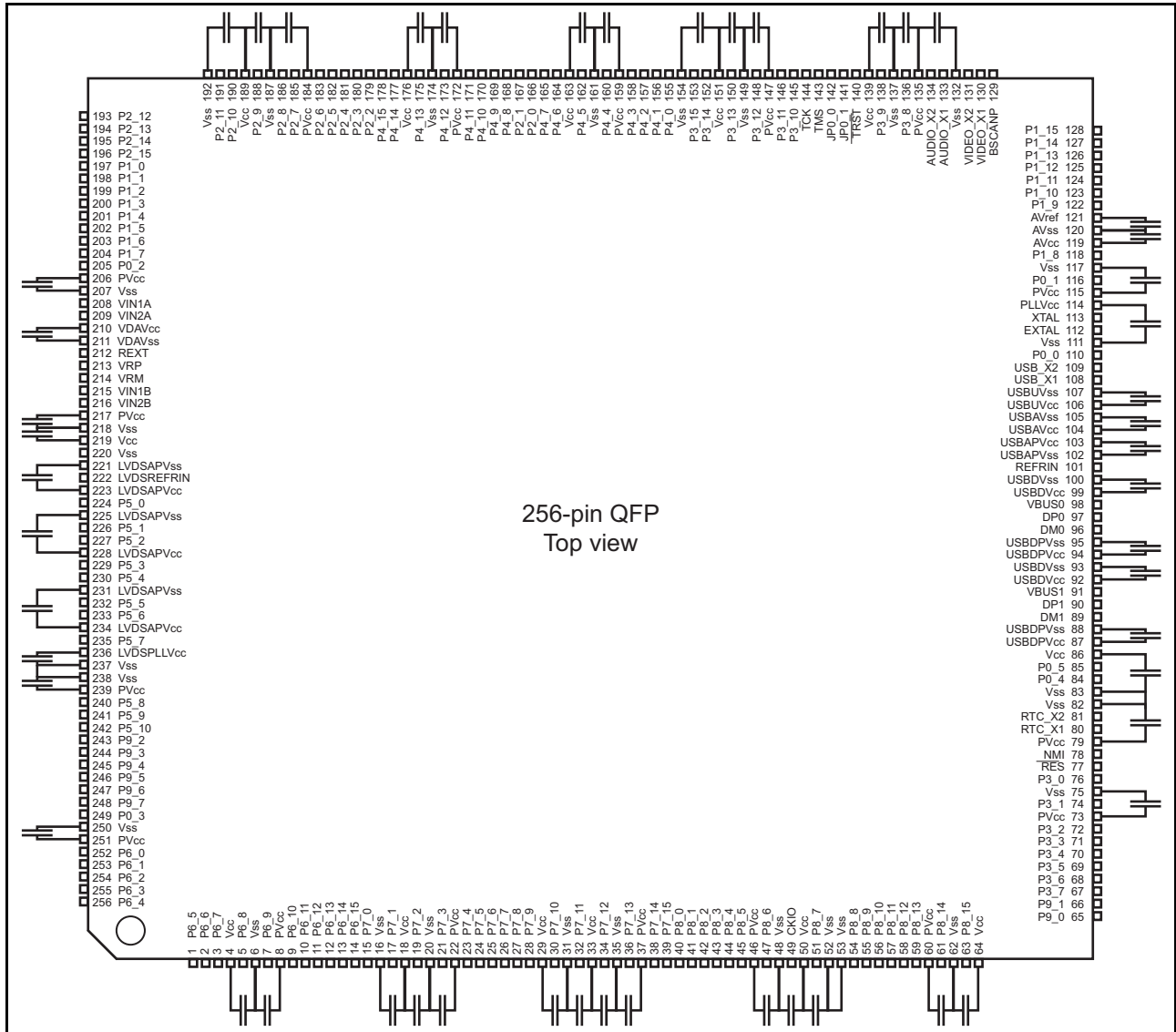


Figure 59.1 Example of Externally Allocated Capacitors in the Products in 256-Pin QFP Packages

# Package

## A. Package Dimensions

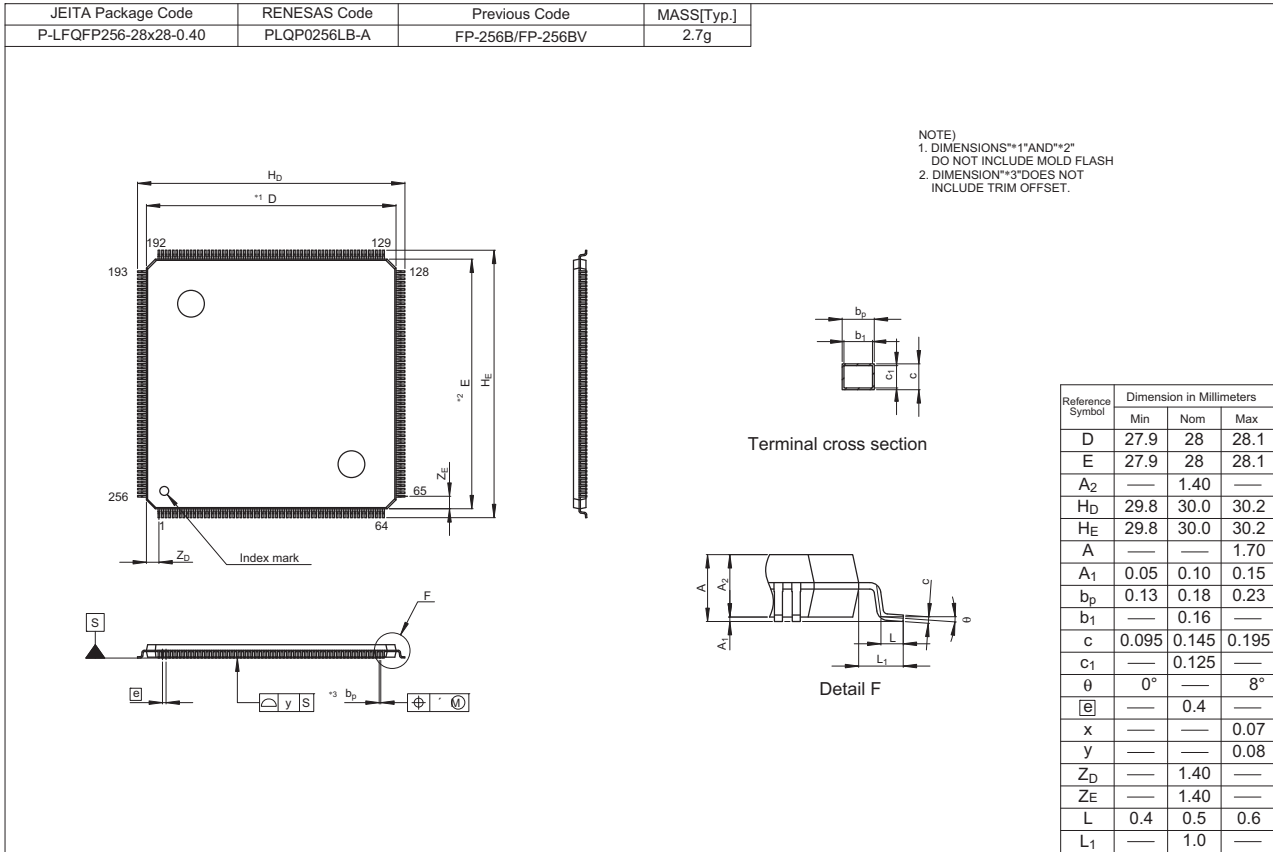


Figure A.1 Dimensions of 256-Pin QFP Package

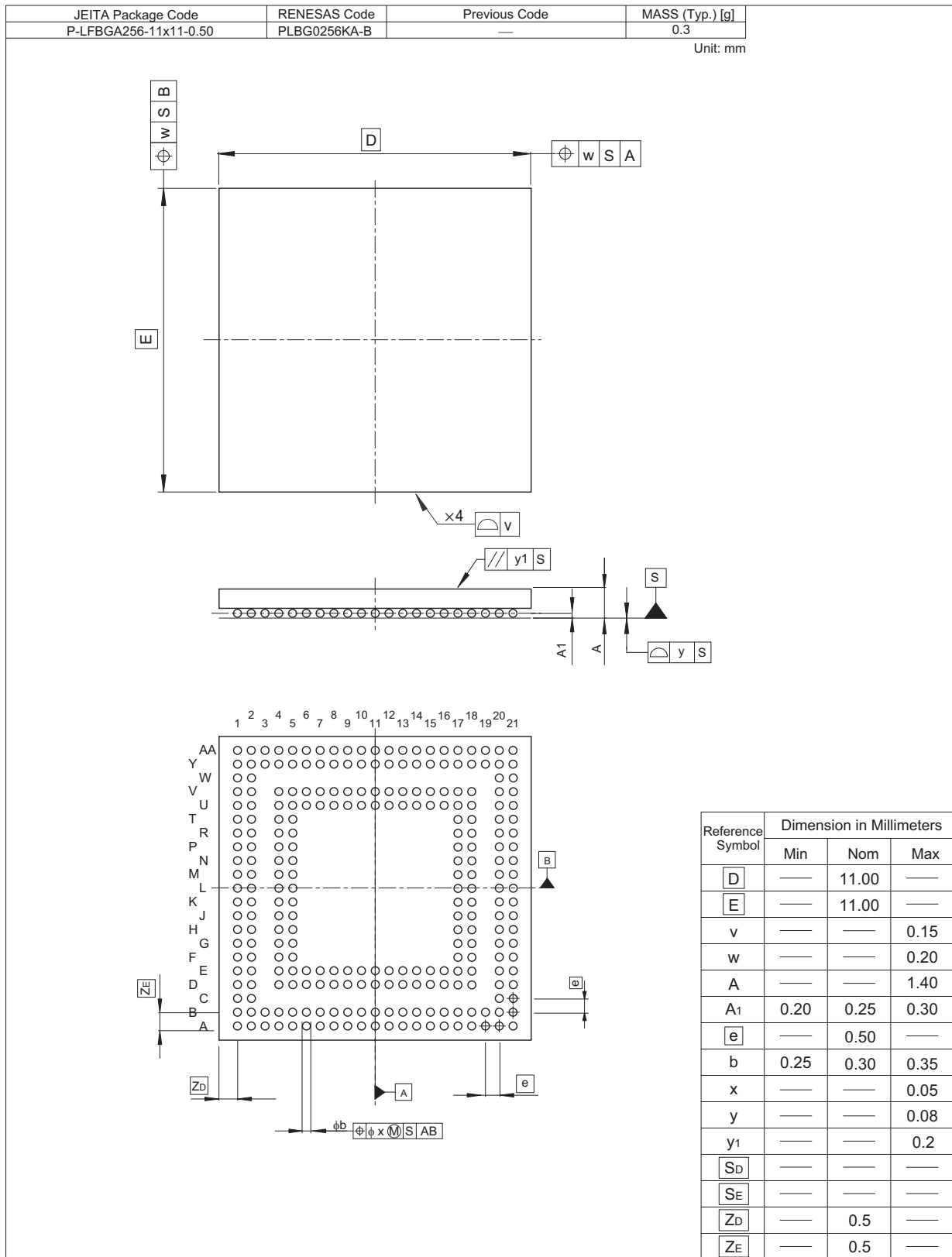


Figure A.2 Dimensions of 256-Pin BGA Package

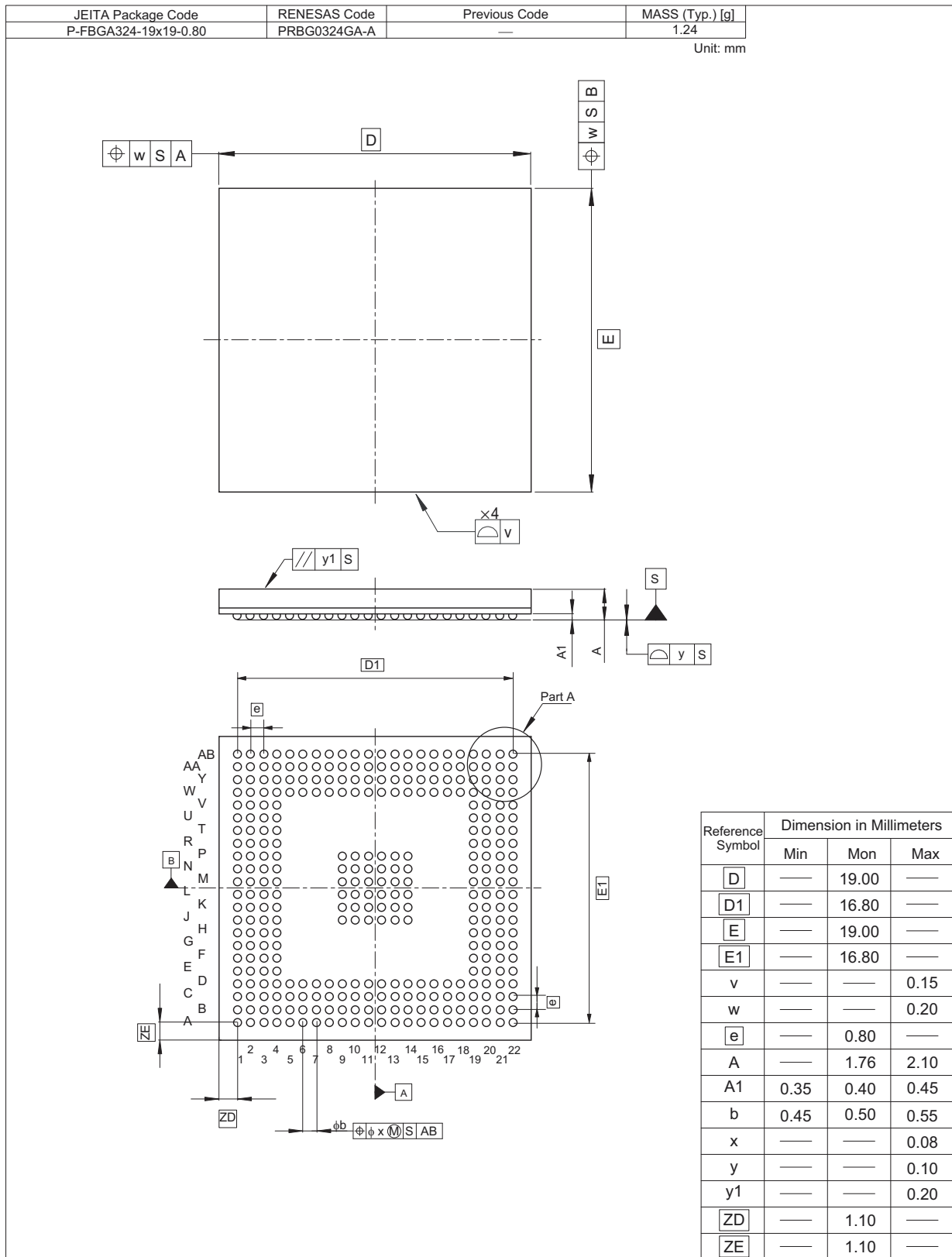


Figure A.3 Dimensions of 324-Pin BGA Package

## B. Thermal Characteristics

The junction temperature ( $T_j$ ) of an average sample (typ. value) can be calculated from either of the following expressions.

Calculation by using  $\theta_{ja}$

$T_j = T_a + (P_D \times \theta_{ja})$ , where

- $T_j$ : junction temperature;
- $T_a$ : ambient temperature;
- $P_D$ : power consumption of the entire LSI; and
- $\theta_{ja}$ : thermal resistance between  $T_j$  and  $T_a$ .

Calculation by using  $\Psi_{jt}$

$T_j = T_t + (P_D \times \Psi_{jt})$ , where

- $T_t$ : temperature at the center of the top of the package; and
- $\Psi_{jt}$ : thermal characteristics parameter representing the difference between  $T_j$  and  $T_t$ .

Since the value of  $\theta_{ja}$  varies greatly with the usage environment (e.g. the board and casing), the junction temperature ( $T_j$ ) calculated by using  $\theta_{ja}$  is less accurate. Accordingly, when calculating  $T_j$  by using  $\theta_{ja}$ , allow a sufficient margin in designing the system. The value of  $\Psi_{jt}$  varies little with differences in the usage environment, but pay particular attention to measuring the temperature at the center of the top of the package ( $T_t$ ) if you use this approach. For the values of  $\theta_{ja}$  and  $\Psi_{jt}$ , contact your local sales representative.

---

RZ/A1H Group User's Manual: Hardware

Publication Date: Rev.1.00 Jun 10, 2014

Published by: Renesas Electronics Corporation

---



**SALES OFFICES****Renesas Electronics Corporation**<http://www.renesas.com>Refer to "<http://www.renesas.com/>" for the latest and detailed information.**Renesas Electronics America Inc.**2801 Scott Boulevard Santa Clara, CA 95050-2549, U.S.A.  
Tel: +1-408-588-6000, Fax: +1-408-588-6130**Renesas Electronics Canada Limited**1101 Nicholson Road, Newmarket, Ontario L3Y 9C3, Canada  
Tel: +1-905-898-5441, Fax: +1-905-898-3220**Renesas Electronics Europe Limited**Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K  
Tel: +44-1628-585-100, Fax: +44-1628-585-900**Renesas Electronics Europe GmbH**Arcadiastrasse 10, 40472 Düsseldorf, Germany  
Tel: +49-211-6503-0, Fax: +49-211-6503-1327**Renesas Electronics (China) Co., Ltd.**Room 1709, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100191, P.R.China  
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679**Renesas Electronics (Shanghai) Co., Ltd.**Unit 301, Tower A, Central Towers, 555 Langao Road, Putuo District, Shanghai, P. R. China 200333  
Tel: +86-21-2226-0888, Fax: +86-21-2226-0999**Renesas Electronics Hong Kong Limited**Unit 1601-1613, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong  
Tel: +852-2265-6688, Fax: +852 2886-9022/9044**Renesas Electronics Taiwan Co., Ltd.**13F, No. 363, Fu Shing North Road, Taipei 10543, Taiwan  
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670**Renesas Electronics Singapore Pte. Ltd.**80 Bendemeer Road, Unit #06-02 Hyflux Innovation Centre, Singapore 339949  
Tel: +65-6213-0200, Fax: +65-6213-0300**Renesas Electronics Malaysia Sdn.Bhd.**Unit 906, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia  
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510**Renesas Electronics Korea Co., Ltd.**12F., 234 Teheran-ro, Gangnam-Ku, Seoul, 135-920, Korea  
Tel: +82-2-558-3737, Fax: +82-2-558-5141

RZ/A1H Group