User's Manual

Multimedia Processor for Mobile Applications

Audio/Voice and PWM Interfaces

EMMA Mobile1

Document No. R19UH0027EJ0300 (3rd edition)
(S19253EJ3V0UM00)
Date Published June 2010

[MEMO]

Notice

1. All information included in this document is current as of the date this document is issued. Such information, however, is subject to change without any prior notice. Before purchasing or using any Renesas Electronics products listed herein, please confirm the latest product information with a Renesas Electronics sales office. Also, please pay regular and careful attention to additional and different information to be disclosed by Renesas Electronics such as that disclosed through our website.
2. Renesas Electronics does not assume any liability for infringement of patents, copyrights, or other intellectual property rights of third parties by or arising from the use of Renesas Electronics products or technical information described in this document. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
3. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part.
4. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
5. When exporting the products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations. You should not use Renesas Electronics products or the technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. Renesas Electronics products and technology may not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations.
6. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
7. Renesas Electronics products are classified according to the following three quality grades: "Standard", "High Quality", and "Specific". The recommended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below. You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application categorized as "Specific" without the prior written consent of Renesas Electronics. Further, you may not use any Renesas Electronics product for any application for which it is not intended without the prior written consent of Renesas Electronics. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for an application categorized as "Specific" or for which the product is not intended where you have failed to obtain the prior written consent of Renesas Electronics. The quality grade of each Renesas Electronics product is "Standard" unless otherwise expressly specified in a Renesas Electronics data sheets or data books, etc.

   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots.

   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; safety equipment; and medical equipment not specifically designed for life support.

   "Specific": Aircraft; aerospace equipment; submersible repeaters; nuclear reactor control systems; medical equipment or systems for life support (e.g. artificial life support devices or systems), surgical implantations, or healthcare intervention (e.g. excision, etc.), and any other applications or purposes that pose a direct threat to human life.
8. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
9. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or system manufactured by you.
10. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
11. This document may not be reproduced or duplicated, in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

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

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

General Precautions in the Handling of MPU/MCU Products

The following usage notes are applicable to all MPU/MCU products from Renesas. For detailed usage notes on the products covered by this manual, refer to the relevant sections of the manual. If the descriptions under General Precautions in the Handling of MPU/MCU Products and in the body of the manual differ from each other, the description in the body of the manual takes precedence.

1. Handling of Unused Pins

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

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

2. Processing at Power-on

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

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

   In a finished product where the reset signal is applied to the external reset pin, the states of pins are not guaranteed from the moment when power is supplied until the reset process is completed.

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

3. Prohibition of Access to Reserved Addresses

   Access to reserved addresses is prohibited.

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

4. Clock Signals

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

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

5. Differences between Products

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

   — The characteristics of MPU/MCU in the same group but having different part numbers may differ because of the differences in internal memory capacity and layout pattern. When changing to products of different part numbers, implement a system-evaluation test for each of the products.

PREFACE

Readers

This manual is intended for hardware/software application system designers who wish to understand and use the audio/voice and PWM interface functions of EMMA Mobile1 (EM1), a multimedia processor for mobile applications.

Purpose

This manual is intended to explain to users the hardware and software functions of the audio/voice and PWM interfaces of EM1, and be used as a reference material for developing hardware and software for systems that use EM1.

Organization

This manual consists of the following chapters.

- Chapter 1 Overview
- Chapter 2 Pin functions
- Chapter 3 Registers
- Chapter 4 Description of functions
- Chapter 5 Usage
- Appendix A Transmit/receive operations

How to Read This Manual

It is assumed that the readers of this manual have general knowledge of electricity, logic circuits, and microcontrollers.

To understand the functions of the audio/voice and PWM interfaces of EM1 in detail
→ Read this manual according to the **CONTENTS**.

To understand the other functions of EM1
→ Refer to the user's manual of the respective module.

To understand the electrical specifications of EM1
→ Refer to the Data Sheet.

Conventions

Data significance:	Higher digits on the left and lower digits on the right
Note:	Footnote for item marked with **Note** in the text
Caution:	Information requiring particular attention
Remark:	Supplementary information
Numeric representation:	Binary ... xxxx or xxxxB or xxxb
	Decimal ... xxxx
	Hexadecimal ... xxxxH
Data type:	Word … 32 bits
	Halfword … 16 bits
	Byte … 8 bits

Related Documents

The related documents indicated in this publication may include preliminary versions. However, preliminary versions are not marked as such.

Document Name		Document No.
MC-10118A Data sheet		R19DS0008EJ (S19657E)
μPD77630A Data sheet		S19686E
User's manual	Audio/Voice and PWM Interfaces	This manual
	DDR SDRAM Interface	R19UH0028EJ (S19254E)
	DMA Controller	S19255E
	I^2C Interface	S19256E
	ITU-R BT.656 Interface	S19257E
	LCD Controller	S19258E
	MICROWIRE	S19259E
	NAND Flash Interface	S19260E
	SPI	S19261E
	UART Interface	S19262E
	Image Composer	S19263E
	Image Processor Unit	S19264E
	System Control/General-Purpose I/O Interface	R19UH0029EJ (S19265E)
	Timer	S19266E
	Terrestrial Digital TV Interface	S19267E
	Camera Interface	S19285E
	USB Interface	S19359E
	SD Memory Card Interface	S19361E
	PDMA	S19373E
	One Chip (MC-10118A)	R19UH0030EJ (S19598E)
	One Chip (μPD77630A)	R19UH0031EJ (S19687E)

( ) : old number

Caution The related documents listed above are subject to change without notice. Be sure to use the latest version of each document when designing.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1 OVERVIEW

1.1 Features

The major features of the audio/voice and PWM interfaces are listed below.

1.1.1 Audio/voice interface

○ Interface signals (2 channels; x = 0 or 1)
- Frame synchronization signal: PMx_SEN
- Clock signal: PMx_CLK
- Serial data output: PMx_SO
- Serial data input: PMx_SI

○ Serial interface for the voice/audio codec
○ Seven operation modes enabling the use of various serial interfaces
○ One 32-bit × 32-word FIFO for each of transmission and reception

1.1.2 PWM interface

The PWM interface outputs pulse-width-modulated signals. It is mainly used for the beep sound generator, motor control, and LCD modulation control.

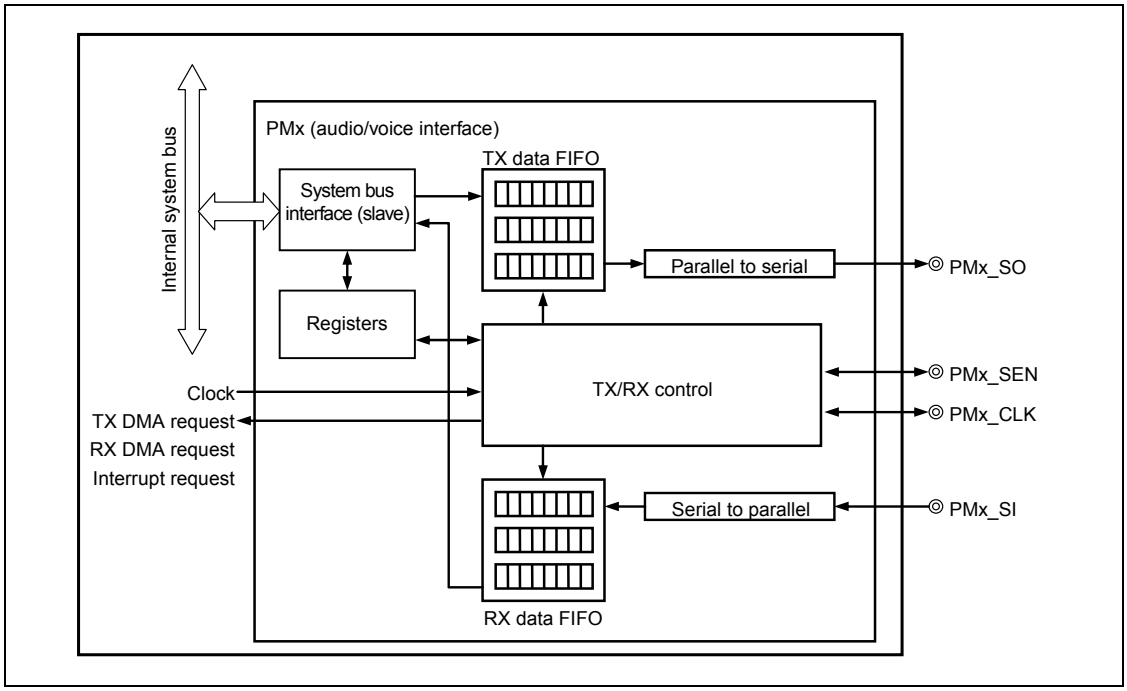
○ 2-channel PWM output
○ Waveform generation
- Three 32-bit counters per channel
- Delay (DELAY), leading edge (LEDGE), trailing edge (TEDGE), and total cycle (TOTAL) can be defined separately.
- Modulation by logic operators between counters (AND, OR, XOR) and for each counter (NOT)

○ The number of repetitions can be specified per counter (managed by 16-bit counters).
○ Independent counter clock (clock for generating PWM waveform)

1.2 Function Block Diagrams

1.2.1 Audio/voice interface

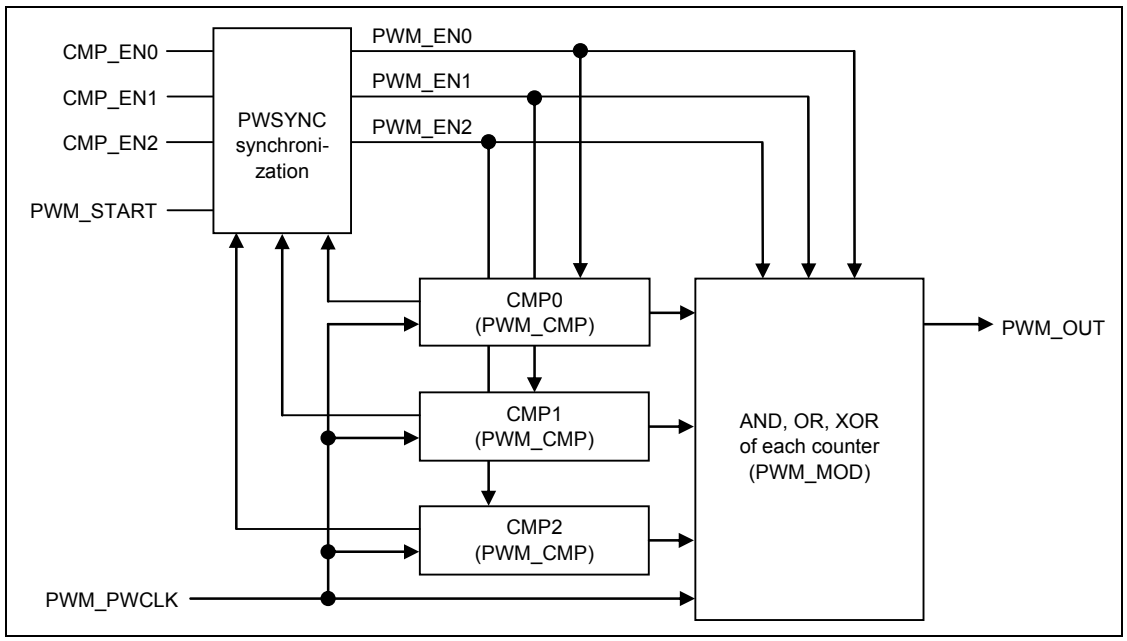
Figure 1-1. Audio/Voice Interface Block Diagram



Remark x = 0 or 1

1.2.2 PWM interface

Figure 1-2. PWM Interface Block Diagram (One Channel)



CHAPTER 2 PIN FUNCTIONS

2.1 Audio/Voice Interface Pins

Pin Name	I/O	After Reset	Function	Alternate Pin Function
PM0_CLK	I/O	0	PCM0 clock (default input)	–
PM0_SEN	I/O	0	PCM0 frame synchronization (default input)	–
PM0_SI	Input	–	PCM0 data	GIO_P87
PM0_SO	Output	0	PCM0 data	–
PM1_CLK	I/O	0	PCM1 clock (default input)	GIO_P72 NTS_CLK
PM1_SEN	I/O	0	PCM1 frame synchronization (default input)	GIO_P80 NTS_DATA5 SP1_CS4
PM1_SI	Input	–	PCM1 data	GIO_P81 NTS_DATA6 SP1_CS5
PM1_SO	Output	0	PCM1 data	GIO_P82 NTS_DATA7

2.2 PWM Interface Pins

Pin Name	I/O	After Reset	Function	Alternate Pin Function
PWM0	Output	0	PWM output	GIO_P94
PWM1	Output	0	PWM output	GIO_P95

CHAPTER 3 REGISTERS

3.1 Registers

The registers of the audio/voice and PWM interfaces allow word access only.
Operation is not guaranteed if an attempt is made to execute halfword or byte access.
Do not access the reserved registers.

3.1.1 Audio/voice interface

Base address: Audio/voice serial 0 C001_0000H
Audio/voice serial 1 400D_0000H

Address	Register Name	Register Symbol	R/W	After Reset
0000H	Operation mode setting register[Note 1]	PMx_FUNC_SEL	R/W	0000_0000H
0004H	Data transfer enable set register	PMx_TXRX_EN	W	–
0008H	Data transfer enable clear register	PMx_TXRX_DIS	W	–
000CH	Data transfer cycle setting register[Note 1]	PMx_CYCLE	R/W	0000_0000H
0010H	Interrupt raw status register	PMx_RAW	R	0000_0000H
0014H	Interrupt status register	PMx_STATUS	R	0000_0000H
0018H	Interrupt enable set register	PMx_ENSET	W	–
001CH	Interrupt enable clear register	PMx_ENCLR	W	–
0020H	Interrupt clear register	PMx_CLEAR	W	–
0024H	Transmit data register	PMx_TXQ	R/W	0000_0000H
0028H	Receive data register	PMx_RXQ	R	0000_0000H
002CH	Reserved	–	–	–
0030H	Data transfer cycle setting register 2[Note 1, 2]	PMx_CYCLE2	R/W	0000_0000H

Notes 1. The settings of the following registers must not be updated while transmission and reception are enabled, transmission and reception are stopped, and during one frame period immediately after transmission and reception are disabled (during one word period in phase 1 or phase 2, whichever is specified as longer, in mode 5 or 6).

- PMx_FUNC_SEL: Operation mode setting register
- PMx_CYCLE: Data transfer cycle setting register
- PMx_CYCLE2: Data transfer cycle setting register 2

2. The PMx_CYCLE2 register is valid only in modes 5 and 6.

Remark x: 0 or 1

3.1.2 PWM interface

Base address: 4010_0000H

Table 3-1. PWM Channel 0 Registers

Address	Register Name	Register Symbol	R/W	After Reset
0000H	PWM operation start/stop register	PWM_CH0_CTRL	R/W	0000_0000H
0004H	PWM0 mode control register	PWM_CH0_MODE	R/W	0000_0000H
0010H	Channel 0 counter 0 delay setting register	PWM_CH0_DELAY0	R/W	0000_0000H
0014H	Channel 0 counter 0 leading edge setting register	PWM_CH0_LEDGE0	R/W	0000_0000H
0018H	Channel 0 counter 0 trailing edge setting register	PWM_CH0_TEDGE0	R/W	0000_0000H**Note**
001CH	Channel 0 counter 0 total cycle setting register	PWM_CH0_TOTAL0	R/W	0000_0000H**Note**
0020H	Channel 0 counter 0 loop count setting register	PWM_CH0_LOOP0	R/W	0000_0000H**Note**
0040H	Channel 0 counter 1 delay setting register	PWM_CH0_DELAY1	R/W	0000_0000H
0044H	Channel 0 counter 1 leading edge setting register	PWM_CH0_LEDGE1	R/W	0000_0000H
0048H	Channel 0 counter 1 trailing edge setting register	PWM_CH0_TEDGE1	R/W	0000_0000H**Note**
004CH	Channel 0 counter 1 total cycle setting register	PWM_CH0_TOTAL1	R/W	0000_0000H**Note**
0050H	Channel 0 counter 1 loop count setting register	PWM_CH0_LOOP1	R/W	0000_0000H**Note**
0080H	Channel 0 counter 2 delay setting register	PWM_CH0_DELAY2	R/W	0000_0000H
0084H	Channel 0 counter 2 leading edge setting register	PWM_CH0_LEDGE2	R/W	0000_0000H
0088H	Channel 0 counter 2 trailing edge setting register	PWM_CH0_TEDGE2	R/W	0000_0000H**Note**
008CH	Channel 0 counter 2 total cycle setting register	PWM_CH0_TOTAL2	R/W	0000_0000H**Note**
0090H	Channel 0 counter 2 loop count setting register	PWM_CH0_LOOP2	R/W	0000_0000H**Note**

Note Do not set PWM_CH0_TEDGEn, PWM_CH0_TOTALn, and PWM_CH0_LOOPn to the After Reset value (0000_0000H). Be sure to use the updated values.

Remark The addresses in the above table are relative values from the start address of the areas assigned to PWM control.

Table 3-2. PWM Channel 1 Registers

Address	Register Name	Register Symbol	R/W	After Reset
0100H	PWM operation start/stop register	PWM_CH1_CTRL	R/W	0000_0000H
0104H	PWM0 mode control register	PWM_CH1_MODE	R/W	0000_0000H
0110H	Channel 1 counter 0 delay setting register	PWM_CH1_DELAY0	R/W	0000_0000H
0114H	Channel 1 counter 0 leading edge setting register	PWM_CH1_LEDGE0	R/W	0000_0000H
0118H	Channel 1 counter 0 trailing edge setting register	PWM_CH1_TEDGE0	R/W	0000_0000H**Note**
011CH	Channel 1 counter 0 total cycle setting register	PWM_CH1_TOTAL0	R/W	0000_0000H**Note**
0120H	Channel 1 counter 0 loop count setting register	PWM_CH1_LOOP0	R/W	0000_0000H**Note**
0140H	Channel 1 counter 1 delay setting register	PWM_CH1_DELAY1	R/W	0000_0000H
0144H	Channel 1 counter 1 leading edge setting register	PWM_CH1_LEDGE1	R/W	0000_0000H
0148H	Channel 1 counter 1 trailing edge setting register	PWM_CH1_TEDGE1	R/W	0000_0000H**Note**
014CH	Channel 1 counter 1 total cycle setting register	PWM_CH1_TOTAL1	R/W	0000_0000H**Note**
0150H	Channel 1 counter 1 loop count setting register	PWM_CH1_LOOP1	R/W	0000_0000H**Note**
0180H	Channel 1 counter 2 delay setting register	PWM_CH1_DELAY2	R/W	0000_0000H
0184H	Channel 1 counter 2 leading edge setting register	PWM_CH1_LEDGE2	R/W	0000_0000H
0188H	Channel 1 counter 2 trailing edge setting register	PWM_CH1_TEDGE2	R/W	0000_0000H**Note**
018CH	Channel 1 counter 2 total cycle setting register	PWM_CH1_TOTAL2	R/W	0000_0000H**Note**
0190H	Channel 1 counter 2 loop count setting register	PWM_CH1_LOOP2	R/W	0000_0000H**Note**

Note Do not set PWM_CH1_TEDGEn, PWM_CH1_TOTALn, and PWM_CH1_LOOPn to the After Reset value (0000_0000H). Be sure to use the updated values.

Table 3-3. PWM Interrupt Registers

Address	Register Name	Register Symbol	R/W	After Reset
0400H	PWM interrupt status register	PWM_INTSTATUS	R	0000_0000H
0404H	PWM interrupt raw status register	PWM_INTRAWSTATUS	R	0000_0000H
0408H	PWM interrupt enable set register	PWM_INTENSET	R/W	0000_0000H
040CH	PWM interrupt enable clear register	PWM_INTENCLR	W	0000_0000H
0410H	PWM interrupt source clear register	PWM_INTFFCLR	W	0000_0000H

3.2 Register Functions (Audio/Voice Interface)

3.2.1 Operation mode setting registers

These registers (PM0_FUNC_SEL: C001_0000H, PM1_FUNC_SEL: 400D_0000H) specify the operation mode, select the serial interface timing, master mode/slave mode, and transmission start timing, and specify whether to enable DMA transfer.

Be sure to set up these registers before transmission and reception are enabled, one frame period after transmission and reception are disabled (during one word period in phase 1 or phase 2, whichever is specified as longer, in mode 5 or 6), or after a transmission/reception re-enable interrupt occurs.

31	30	29	28	27	26	25	24
Reserved							

23	22	21	20	19	18	17	16
Reserved							

15	14	13	12	11	10	9	8
Reserved						CLK_INV	SEN_INV

7	6	5	4	3	2	1	0
Reserved	Tx_TIM		M_S		MODE_SEL		

(1/2)

Name	R/W	Bit	After Reset	Function
Reserved	R	31:10	0	Reserved. When these bits are read, 0 is returned for each bit.
CLK_INV	R/W	9	0	Selects whether to invert the serial clock polarity. 0: Does not invert (default). 1: Inverts.
SEN_INV	R/W	8	0	Selects whether to invert the serial data synchronization signal polarity. 0: Does not invert (default). 1: Inverts
Reserved	R	7	0	Reserved. When this bit is read, 0 is returned.
Tx_TIM	R/W	6:5	0	Selects the amount of valid data in the TX data FIFO required to trigger serial transmission. 00b: 30 words (default) 01b: 16 words 10b: 8 words 11b: 4 words
M_S	R/W	4:3	0	Selects the master mode/slave mode. 00b: Stop (default) 01b: Master mode 10b: Slave mode 11b: Setting prohibited

(2/2)

Name	R/W	Bit	After Reset	Function
MODE_SEL	R/W	2:0	0	Selects the serial interface operation mode. For the timing in each mode, see **4.1.2 Serial interface timing.** 000b: Mode 0 (default) 001b: Mode 1 010b: Mode 2 (I^2S format) 011b: Mode 3 (MSB justified) 100b: Mode 4 (LSB justified) 101b: Mode 5 (multi-channel mode) 110b: Mode 6 (multi-channel mode)

3.2.2 Data transfer enable set registers

These registers (PM0_TXRX_EN: C001_0004H, PM1_TXRX_EN: 400D_0004H) specify the reception enable bit and transmission enable bit.

The reception enable bit and the transmission enable bit are used to specify whether to enable serial transfer. If these bits are set to 1, serial transfer is performed. The status can be checked by reading this register.

If 1 is read in the RX_EN bit, reception is enabled. If 1 is read in the TX_EN bit, transmission is enabled. If 0 is read, transmission and reception are disabled. To clear these bits, use the data transfer enable clear register (PMx_TXRX_DIS).

31	30	29	28	27	26	25	24
Reserved							

23	22	21	20	19	18	17	16
Reserved							

15	14	13	12	11	10	9	8
Reserved							

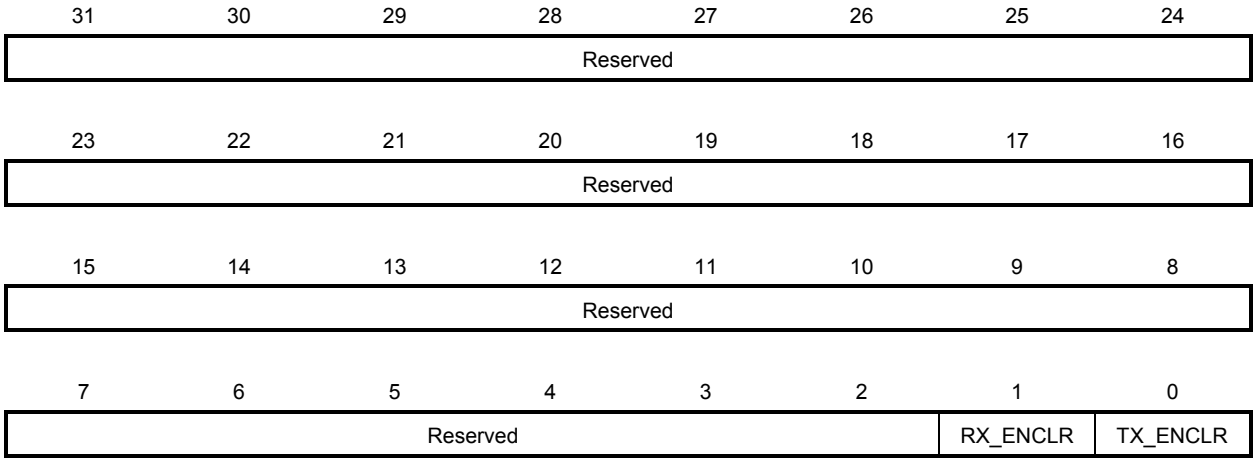
7	6	5	4	3	2	1	0
Reserved						RX_EN	TX_EN

Name	R/W	Bit	After Reset	Function
Reserved	R	31:2	0	Reserved. When these bits are read, 0 is returned for each bit.
RX_EN	W	1	0	Sets the reception enable bit[Note]. 0: The current status is retained. 1: Sets the reception enable bit to 1.
TX_EN	W	0	0	Sets the transmission enable bit[Note]. 0: The current status is retained. 1: Sets the transmission enable bit to 1.

Note When enabling transmission/reception that has been stopped, wait for at least one frame (one word period in phase 1 or 2, whichever is specified as longer, in modes 5 and 6) after transmission/reception was stopped or a transmission/reception re-enable interrupt occurs, so as to assure the period in which the PCM block operation stops completely.

3.2.3 Data transfer enable clear registers

These registers (PM0_TXRX_DIS: C001_0008H, PM1_TXRX_DIS: 400D_0008H) clear the reception enable bit and the transmission enable bit. If 1 is written to the RX_ENCLR bit, the reception enable bit is cleared. If 1 is written to the TX_ENCLR bit, the transmission enable bit is cleared. If 0 is written, the current status is retained.



Name	R/W	Bit	After Reset	Function
Reserved	–	31:2	0	Reserved.
RX_ENCLR	W	1	0	Clears the reception enable bit. 0: The current status is retained. 1: Clears the reception enable bit to 0.
TX_ENCLR	W	0	0	Clears the transmission enable bit. 0: The current status is retained. 1: Clears the transmission enable bit to 0.

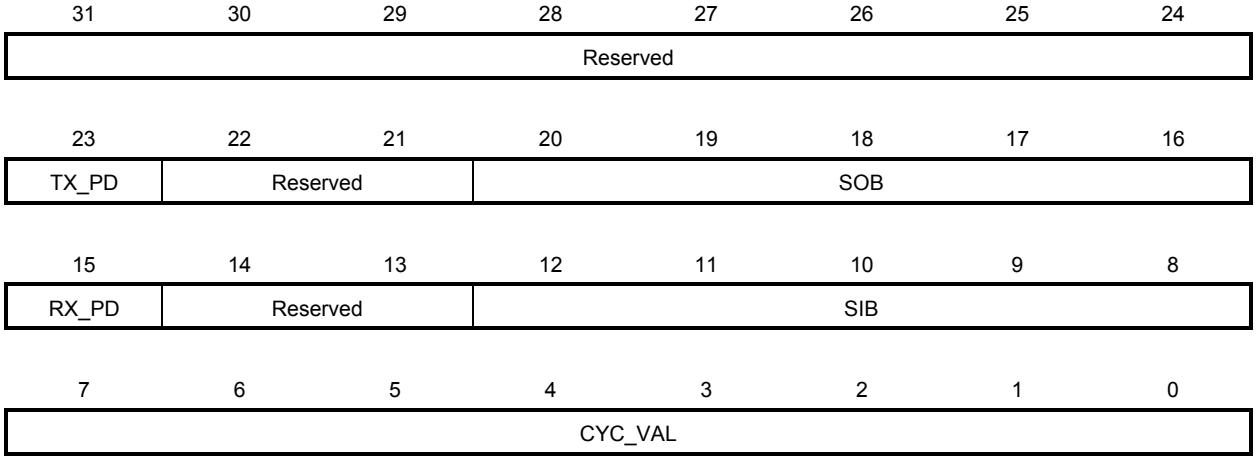
Remark For details of the statuses in a transmit operation, see **A.1.1 Status transitions in transmission block**. For details of the statuses in a receive operation, see **A.1.3 Status transitions in reception block**.

1) PMx_TXRX_EN (write register): Writing 1 sets the corresponding bit to 1.
2) PMx_TXRX_DIS (write register): Writing 1 sets the corresponding bit to 0.
3) PMx_TXRX_MON (read register): The current status is read out.

Example The execution result varies depending on the order of writing to the registers.
Set PMx_TXRX_EN to 11B → Bits 1 and 0 are set to ENABLE.
Set PMx_TXRX_DIS to 01B → Bit 0 is set to DISABLE.
Read PMx_TXRX_MON = 10B → Bit 1 remains ENABLE.

3.2.4 Data transfer cycle setting registers

These registers (PM0_CYCLE: C001_000CH, PM1_CYCLE: 400D_000CH) specify the frame length, the number of valid receive bits and the number of valid transmit bits, and whether to enable padding when receiving and transmitting data. Be sure to set up these registers before transmission and reception are enabled, and one frame period after transmission and reception are disabled (one word period in phase 1 or phase 2, whichever is specified as longer, in mode 5 or 6), or after a transmission/reception re-enable interrupt occurs. The settings of these registers apply to phase 1 in modes 5 and 6.



(1/2)

Name	R/W	Bit	After Reset	Function
Reserved	R	31:24	0	Reserved. When these bits are read, 0 is returned for each bit.
TX_PD	R/W	23	0	Specifies whether to enable padding when transmitting data. 0: Disable (default) 1: Enable To enable padding, the number of valid transmit bits must be set to 8 (SOB = 07H) or 16 (SOB = 0FH). The enable setting is prohibited in modes 5 and 6. Be sure to set this bit to 0.
Reserved	R	22:21	0	Reserved. When these bits are read, 0 is returned for each bit.
SOB	R/W	20:16	0	Specifies the number of valid transmit bits (PMx_SO signal) according to the number of PMx_CLK clock cycles. The number of valid transmit bits is (set value + 1). In modes 0 to 4, CYC_VAL ≥ SOB (the number of valid transmit bits) must be satisfied. Setting range: 07H to 1FH (8 to 32 bits)[Note 1]
RX_PD	R/W	15	0	Specifies whether to enable padding when receiving data. 0: Disable (default) 1: Enable To enable padding, the number of valid receive bits must be set to 8 (SIB = 07H) or 16 (SIB = 0FH). The enable setting is prohibited in modes 5 and 6. Be sure to set this bit to 0.
Reserved	R	14:13	0	Reserved. When these bits are read, 0 is returned for each bit.

(2/2)

Name	R/W	Bit	After Reset	Function
SIB	R/W	12: 8	0	Specifies the number of valid receive bits (PMx_SI signal) according to the number of PMx_CLK clock cycles. The number of valid receive bits is (set value + 1). In modes 0 to 4, CYC_VAL ≥ SIB (the number of valid receive bits) must be satisfied. Setting range: 07H to 1FH (8 to 32 bits)[Note 1]
CYC_VAL	R/W	7:0	0	In modes 0 to 4, the frame length is specified according to the number of PMx_CLK clock cycles. In mode 5 or 6, the frame length is specified by the number of words of phase 1. The frame length is (set value + 1). In modes 0 to 4, CYC_VAL ≥ SOB (the number of valid transmit bits) and CYC_VAL ≥ SIB (the number of valid receive bits) must be satisfied. The specifiable number of clock cycles depends on the mode selected. Modes 0 to 4: 07H to 3FH (8 to 64 clock cycles) Mode 5 or 6: 00H to 80H (0 to 128 clock cycles)[Note 2]

Notes 1. In mode 5 or 6, set the SOB and SIB bits to the same values for a simultaneous transmit/receive operation. A value different from phase 2 (SOB2 and SIB2 bits of PMx_CYCLE2 register) can be specified. (The same values must be specified for SOB and SIB. Setting different values to SOB and SOB2 or SIB and SIB2 is allowed.)

2. In mode 5 or 6, setting the CYC_VAL bit of the PMx_CYCLE register to 00H is prohibited if the CYC_VAL2 bit of the PMx_CYCLE2 register is set to 00H. If the CYC_VAL bit is set to 00H, the value set to the PMx_CYCLE2 register becomes valid and the single-phase operation is performed. A value different from phase 2 (SOB2 and SIB2 bits of PMx_CYCLE2 register) can be specified. (Setting different values to CYC_VAL and CYC_VAL2 is allowed.)

3.2.5 Interrupt raw status registers

These registers (PM0_RAW: C001_0010H, PM1_RAW: 400D_0010H) indicate the status of the interrupt sources.

31	30	29	28	27	26	25	24
Reserved							

23	22	21	20	19	18	17	16
Reserved							

15	14	13	12	11	10	9	8
Reserved			RX_STPRAW	Reserved			TX_STPRAW

7	6	5	4	3	2	1	0
RX_RENRAW	RX_ORERAW	RX_URERAW	RX_FRERAW	TX_WENRAW	TX_ORERAW	TX_URERAW	TX_FRERAW

(1/2)

Name	R/W	Bit	After Reset	Function
Reserved	R	31:13	0	Reserved. When these bits are read, 0 is returned for each bit.
RX_STPRAW	R	12	0	Indicates the raw status of the reception re-enable setting after reception has been stopped. 0: Before reception, or reception in progress (default). 1: Reception is re-enabled[Note 8].
Reserved	R	11:9	0	Reserved. When these bits are read, 0 is returned for each bit.
TX_STPRAW	R	8	0	Indicates the raw status of the transmission re-enable setting after transmission has been stopped. 0: Before transmission, or transmission in progress (default). 1: Transmission is re-enabled[Note 9].
RX_RENRAW	R	7	0	Indicates the raw status of the RX data FIFO. 0: There is no valid receive data (disabling read operation) (default). 1: There is valid receive data (enabling read operation)[Note 1].
RX_ORERAW	R	6	0	Indicates the raw status of a receive overrun error. 0: Normal (default) 1: An overrun error is detected[Note 1].
RX_URERAW	R	5	0	Indicates the raw status of a receive underrun error. 0: Normal (default) 1: An underrun error is detected[Note 1].
RX_FRERAW	R	4	0	Indicates the raw status of a receive frame synchronization error[Note 6]. 0: Normal (default) 1: A frame synchronization error is detected (only in slave operation)[Note 2].
TX_WENRAW	R	3	0	Indicates the raw status of the TX data FIFO[Note 7]. 0: The TX data FIFO has no free space (disabling write operation) (default) 1: The TX data FIFO has free space (enabling write operation)[Note 3].

(2/2)

Name	R/W	Bit	After Reset	Function
TX_ORERAW	R	2	0	Indicates the raw status of a transmit overrun error. 0: Normal (default) 1: An overrun error is detected[Note 3].
TX_URERAW	R	1	0	Indicates the raw status of a transmit underrun error. 0: Normal (default) 1: An underrun error is detected[Note 4].
TX_FRERAW	R	0	0	Indicates the raw status of a transmit frame synchronization error[Note 6]. 0: Normal (default) 1: A frame synchronization error is detected (only in slave operation)[Note 5].

Notes 1. This setting is reset by disabling reception (bit 1 of PMx_TXRX_DIS = 1).

2. This setting is reset by disabling reception (bit 1 of PMx_TXRX_DIS = 1) or clearing the interrupt source (bit 4 of PMx_CLEAR = 1).

3. This setting is reset by disabling transmission (bit 0 of PMx_TXRX_DIS = 1).

4. This setting is reset by disabling transmission (bit 0 of PMx_TXRX_DIS = 1) or clearing the interrupt source (bit 1 of PMx_CLEAR = 1).

5. This setting is reset by disabling transmission (bit 0 of PMx_TXRX_DIS = 1) or clearing the interrupt source (bit 0 of PMx_CLEAR = 1).

6. For synchronization errors, see **Note** in **3.2.6 Interrupt status registers**.

7. This bit is set to 0 during reset, and is set to 1 when 1 PCLK is input after reset release.

8. This setting is reset by clearing the interrupt source (bit 12 of PMx_CLEAR = 1).

9. This setting is reset by clearing the interrupt source (bit 8 of PMx_CLEAR = 1).

3.2.6 Interrupt status registers

These registers (PM0_STATUS: C001_0014H, PM1_STATUS: 400D_0014H) indicate the status of the interrupt sources.

The result of masking the interrupt raw status register with values set to the interrupt enable set register (PMx_ENSET) is shown.

31	30	29	28	27	26	25	24
Reserved							

23	22	21	20	19	18	17	16
Reserved							

15	14	13	12	11	10	9	8
Reserved			RX_STP	Reserved			TX_STP

7	6	5	4	3	2	1	0
RX_REN	RX_ORE	RX_URE	RX_FRE	TX_WEN	TX_ORE	TX_URE	TX_FRE

Name	R/W	Bit	After Reset	Function
Reserved	R	31:13	0	Reserved. When these bits are read, 0 is returned for each bit.
RX_STP	R	12	0	Indicates the status of the reception re-enable interrupt after reception stoppage is requested.
Reserved	R	11:9	0	Reserved. When these bits are read, 0 is returned for each bit.
TX_STP	R	8	0	Indicates the status of the transmission re-enable interrupt after transmission stoppage is requested.
RX_REN	R	7	0	Indicates the status of the RX data FIFO interrupt.
RX_ORE	R	6	0	Indicates the status of the receive overrun error interrupt.
RX_URE	R	5	0	Indicates the status of the receive underrun error interrupt.
RX_FRE	R	4	0	Indicates the status of the receive frame synchronization error interrupt.
TX_WEN	R	3	0	Indicates the status of the TX data FIFO interrupt[Note].
TX_ORE	R	2	0	Indicates the status of the transmit overrun error interrupt.
TX_URE	R	1	0	Indicates the status of the transmit underrun error interrupt.
TX_FRE	R	0	0	Indicates the status of the transmit frame synchronization error interrupt.

Note This bit is set to 0 during reset, and is set to 1 when 1 PCLK cycle has been input after reset release.

Remark Each interrupt source is set or reset at the following timing.

- RX_STP: Indicates whether reception can be re-enabled after reception stoppage is requested.
  Because this flag is set after the transfer being executed is stopped, the time taken until this bit is set varies depending on conditions such as the timing at which reception stoppage is requested, and whether the device is a master or a slave.

  Set (1): When reception can be re-enabled after reception is requested.
  Reset (0): Reception has not started or reception is in progress.

- TX_STP: Indicates whether transmission can be re-enabled after transmission stoppage is requested.
  Because this flag is set after the transfer being executed is stopped, the time taken until this bit is set varies depending on conditions such as the timing at which transmission stoppage is requested, and whether the device is a master or a slave.

  Set (1): When transmission can be re-enabled after transmission is requested.
  Reset (0): When transmission has not started or transmission is in progress.

- RX_REN: Indicates that receive data can be read.
  This flag and the PMx_DMA_RDMARQ (DMA receive request) signal are set to on or off simultaneously.

  Set (1): When there is valid data of at least one word in the receive FIFO (there is data that has not been read by the DMA controller).
  Reset (0): When all data in the receive FIFO has been read or receive is disabled (bit 1 of PMx_TXRX_DIS = 1).

- RX_ORE: Indicates that an overrun error occurred in the receive FIFO.

  Set (1): When the receive FIFO is not read by the DMA controller in time (serial data is received from the PMx_SI pin and is written to the receive FIFO while the receive FIFO is full).
  Reset (0): When reception is disabled (bit 1 of PMx_TXRX_DIS = 1).

- RX_URE: Indicates that an underrun error occurred in the receive FIFO.

  Set (1): When the receive FIFO is read when there is no valid data in the receive FIFO (RE_REN = 0) (non-existent receive data is read).
  Reset (0): When reception is disabled (bit 1 of PMx_TXRX_DIS = 1).

- RX_FRE: Indicates that a serial interface synchronization error[Note] occurred in the receive controller.

  Set (1): When deviation of the PMx_SEN signal is detected (only in the case of slave operation).
  Reset (0): When reception is disabled (bit 1 of PMx_TXRX_DIS = 1) after the interrupt source is cleared (bit 4 of PMx_CLEAR = 1).

- TX_WEN: Indicates that writing transmit data is enabled.

  This flag and the PMx_DMA_TDMARQ (DMA transmit request) signal are set to on or off simultaneously.

  Set (1): When the transmit FIFO is not full (the transmit FIFO has free space and data can be written).

  Reset (0): When the transmit FIFO is full and data can no longer be written, or transmission is disabled (bit 0 of PMx_TXRX_DIS = 1).

- TX_ORE: Indicates that an overrun error occurred in the transmit FIFO.

  Set (1): When data is written to the transmit FIFO when the transmit FIFO is full (TX_WEN = 0).

  Reset (0): When transmission is disabled (bit 0 of PMx_TXRX_DIS = 1).

- TX_URE: Indicates that an underrun error occurred in the transmit FIFO.

  Set (1): When all transmit data has been output from the PMx_SO pin and the transmit FIFO is empty (all-data transmit completion has been reported, or the serial data transmit speed is faster than writing to the transmit FIFO).

  Reset (0): When transmission is disabled (bit 0 of PMx_TXRX_DIS = 1) after the interrupt source is cleared (bit 1 of PMx_CLEAR = 1).

- TX_FRE: Indicates that a serial interface synchronization error[Note] occurred in the transmit controller.

  Set (1): When deviation of the PMx_SEN signal is detected (only in the case of slave operation).

  Reset (0): When transmission is disabled (bit 0 of PMx_TXRX_DIS = 1) after the interrupt source is cleared (bit 0 of PMx_CLEAR = 1).

Note A synchronization error occurs when the PMx_SEN signal does not change for the “number of clock cycles per frame” specified by CYC_VAL (modes 0 to 4), or CYC_VAL, SIB, SOB, CYC_VAL2, SIB2, and SOB2 (modes 5 and 6) in the data transfer cycle setting registers (PMx_CYCLE, PMx_CYCLE2) during communication (the change depends on the format of modes 0 to 6). A synchronization error also occurs if the transmit module and the receive module fall out of synchronization.

RX_FRE and TX_FRE are set to on respectively when a synchronization error is detected in the receiver or transmitter. If both reception and transmission are enabled, the flag for which the error occurred first is set (actually these errors occur almost at the same time).

3.2.7 Interrupt enable set registers

These registers (PM0_ENSET: C001_0018H, PM1_ENSET: 400D_0018H) specify whether to enable issuance of the interrupt request for each interrupt source. The status of the interrupt enable bits can be checked by reading this register.

Be sure to set up these registers before transmission and reception are enabled, and one frame period after transmission and reception are disabled (one word period in phase 1 or phase 2, whichever is specified as longer, in mode 5 or 6), or after a transmission/reception re-enable interrupt occurs.

31	30	29	28	27	26	25	24
Reserved							

23	22	21	20	19	18	17	16
Reserved							

15	14	13	12	11	10	9	8
Reserved			RX_STP_EN	Reserved			TX_STP_EN

7	6	5	4	3	2	1	0
RX_REN_EN	RX_ORE_EN	RX_URE_EN	RX_FRE_EN	TX_WEN_EN	TX_ORE_EN	TX_URE_EN	TX_FRE_EN

(1/2)

Name	R/W	Bit	After Reset	Function
Reserved	R	31:13	0	Reserved. When these bits are read, 0 is returned for each bit.
RX_STP_EN	W	12	0	Enables the reception re-enable interrupt. 1: Sets the corresponding interrupt enable bit. 0: Retains the setting of the interrupt enable bit.
Reserved	R	11:9	0	Reserved. When these bits are read, 0 is returned for each bit.
TX_STP_EN	W	8	0	Enables the transmission re-enable interrupt. 1: Sets the corresponding interrupt enable bit. 0: Retains the setting of the interrupt enable bit.
RX_REN_EN	W	7	0	Enables the RX data FIFO interrupt. 1: Sets the corresponding interrupt enable bit. 0: Retains the setting of an interrupt enable bit.
RX_ORE_EN	W	6	0	Enables the receive overrun error interrupt. 1: Sets the corresponding interrupt enable bit. 0: Retains the setting of the interrupt enable bit.
RX_URE_EN	W	5	0	Enables the receive underrun error interrupt. 1: Sets the corresponding interrupt enable bit. 0: Retains the setting of the interrupt enable bit.
RX_FRE_EN	W	4	0	Enables the receive frame synchronization error interrupt. 1: Sets the corresponding interrupt enable bit. 0: Retains the setting of the interrupt enable bit.

(2/2)

Name	R/W	Bit	After Reset	Function
TX_WEN_EN	W	3	0	Enables the TX data FIFO interrupt. 1: Sets the corresponding interrupt enable bit. 0: Retains the setting of the interrupt enable bit.
TX_ORE_EN	W	2	0	Enables the transmit overrun error interrupt. 1: Sets the corresponding interrupt enable bit. 0: Retains the setting of the interrupt enable bit.
TX_URE_EN	W	1	0	Enables the transmit underrun error interrupt. 1: Sets the corresponding interrupt enable bit. 0: Retains the setting of the interrupt enable bit.
TX_FRE_EN	W	0	0	Enables the transmit frame synchronization error interrupt. 1: Sets the corresponding interrupt enable bit. 0: Retains the setting of the interrupt enable bit.

Cautions 1. When performing DMA transfer, it is recommended to use bit 7 (RX_REN_EN) and bit 3 (TX_WEN_EN) with their interrupt mask settings. Interrupts corresponding to bits 7 and 3 are unnecessary because they can be checked by using the DMA request signal.

2. Interrupts are set using the three registers shown below. Each interrupt can be enabled by writing 1 to the corresponding bit.

Example The execution result varies depending on the order of writing to the registers.

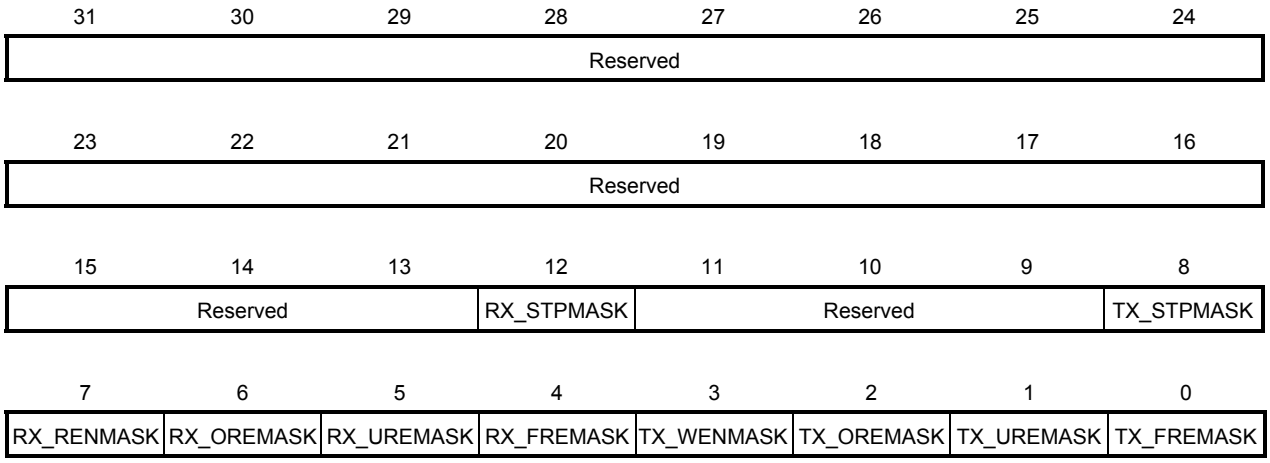
Set PMx_ENSET to 0101_0101B → Interrupts corresponding to bits 6, 4, 2 and 0 are enabled.

Set PMx_ENCLR to 0001_0001B → Interrupts corresponding to bits 4 and 0 are disabled.

Read PMx_ENSET = 0100_0100B → Interrupts corresponding to bits 6 and 2 remain enabled.

3.2.8 Interrupt enable clear registers

These registers (PM0_ENCLR: C001_001CH, PM1_ENCLR: 400D_001CH) clear the interrupt request issuance enable bit (interrupt enable bit) for each interrupt source.



(1/2)

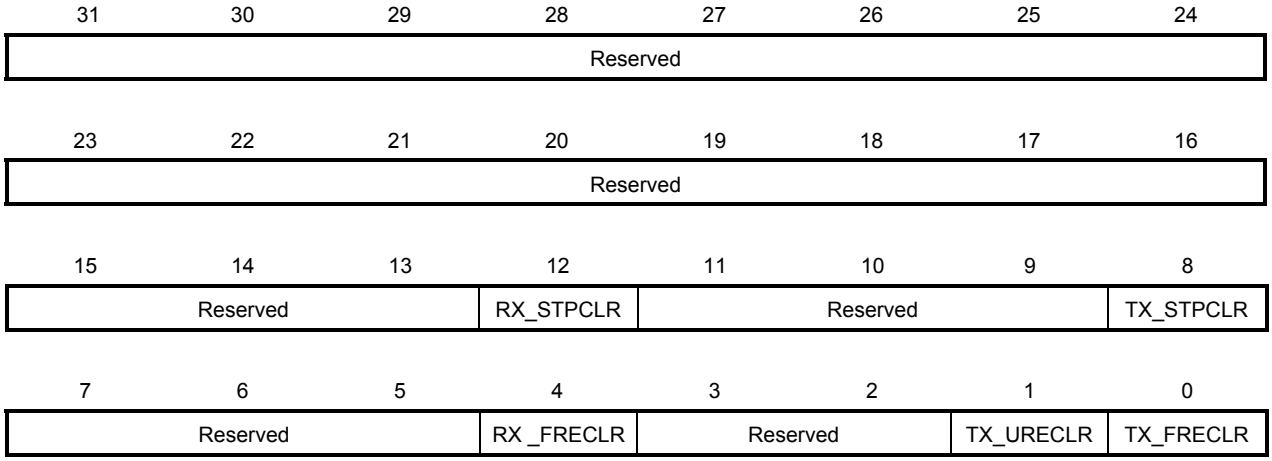
Name	R/W	Bit	After Reset	Function
Reserved	–	31:13	–	Reserved.
RX_STP_MASK	W	12	–	Clears the reception re-enable interrupt enable bit. 1: Clears the corresponding interrupt enable bit. 0: Retains the setting of the interrupt enable bit.
Reserved	–	11:9	–	Reserved.
TX_STP_MASK	W	8	–	Clears the transmission re-enable interrupt enable bit. 1: Clears the corresponding interrupt enable bit. 0: Retains the setting of the interrupt enable bit.
RX_RENMASK	W	7	–	Clears the RX data FIFO interrupt enable bit. 1: Clears the corresponding interrupt enable bit. 0: Retains the setting of the interrupt enable bit.
RX_OREMASK	W	6	–	Clears the receive overrun error interrupt enable bit. 1: Clears the corresponding interrupt enable bit. 0: Retains the setting of the interrupt enable bit.
RX_UREMASK	W	5	–	Clears the receive underrun error interrupt enable bit. 1: Clears the corresponding interrupt enable bit. 0: Retains the setting of the interrupt enable bit.
RX_FREMASK	W	4	–	Clears the receive frame synchronization error interrupt enable bit. 1: Clears the corresponding interrupt enable bit. 0: Retains the setting of the interrupt enable bit.
TX_WENMASK	W	3	–	Clears the TX data FIFO interrupt enable bit. 1: Clears the corresponding interrupt enable bit. 0: Retains the setting of the interrupt enable bit.
TX_OREMASK	W	2	–	Clears the transmit overrun error interrupt enable bit. 1: Clears the corresponding interrupt enable bit. 0: Retains the setting of the interrupt enable bit.

(2/2)

Name	R/W	Bit	After Reset	Function
TX_UREMASK	W	1	–	Clears the transmit underrun error interrupt enable bit. 1: Clears the corresponding interrupt enable bit. 0: Retains the setting of an interrupt enable bit.
TX_FREMASK	W	0	–	Clears the transmit frame synchronization error interrupt enable bit. 1: Clears the corresponding interrupt enable bit. 0: Retains the setting of the interrupt enable bit.

3.2.9 Interrupt clear registers

These registers (PM0_CLEAR: C001_0020H, PM1_CLEAR: 400D_0020H) clear the interrupt sources.



Name	R/W	Bit	After Reset	Function
Reserved	–	31:13	–	Reserved.
RX_STPCLR	W	12	–	Clears the RX_STP source (reception re-enable) bit. 1: Clears the corresponding interrupt source. 0: Retains the interrupt source.
Reserved	–	11:9	–	Reserved.
TX_STPCLR	W	8	–	Clears the TX_STP source (transmission re-enable) bit. 1: Clears the corresponding interrupt source. 0: Retains the interrupt source.
Reserved	–	7:5	–	Reserved.
RX_FRECLR	W	4	–	Clears the RX_FRE source (receive synchronization error) bit. 1: Clears the corresponding interrupt source. 0: Retains the interrupt source.
Reserved	–	3:2	–	Reserved.
TX_URECLR	W	1	–	Clears the TX_URE source (transmit underrun error) bit. 1: Clears the corresponding interrupt source. 0: Retains the interrupt source.
TX_FRECLR	W	0	–	Clears the TX_FRE source (transmit synchronization error) bit. 1: Clears the corresponding interrupt source. 0: Retains the interrupt source.

Remark Other interrupt sources are reset by disabling transmission (bit 0 of PMx_TXRX_DIS = 1) or disabling reception (bit 1 of PMx_TXRX_DIS = 1).

3.2.10 Transmit data registers

These registers (PM0_TXQ: C001_0024H, PM1_TXQ: 400D_0024H) store the transmit data to be written to the TX data FIFO. The last data written to the TX data FIFO can be read by reading this register.

Be sure to write to this register in 32-bit units. Operation is not guaranteed if this register is accessed in 8-bit or 16-bit units.

31	30	29	28	27	26	25	24
TXQ							

23	22	21	20	19	18	17	16
TXQ							

15	14	13	12	11	10	9	8
TXQ							

7	6	5	4	3	2	1	0
TXQ							

Name	R/W	Bit	After Reset	Function
TXQ	R/W	31:0	0	Controls TX data FIFO write data.

Remark The data written to PMx_TXQ is written to the entry pointed to by the write pointer in the 32-bit × 32-word transmit FIFO. The write pointer is automatically controlled by hardware. If PMx_TXQ is read, the last written data is read.

An example of writing data to the transmit FIFO is provided below.
Data longer than the transmit data bit length is ignored (×: Don't Care).

Example Transmit data bit length = 10 bits (SOB = 09H in PMx_CYCLE register)

xxxx_x123H → Write data (PMx_TXQ)
xxxx_x245H → Write data (PMx_TXQ)
xxxx_x067H → Write data (PMx_TXQ)
:

31 TX-FIFO[0] 24							
×	×	×	×	×	×	×	×
×	×	×	×	×	×	×	×
×	×	×	×	×	×	0	1
0	0	1	0	0	0	1	1
7							0

31 TX-FIFO[1] 24							
×	×	×	×	×	×	×	×
×	×	×	×	×	×	×	×
×	×	×	×	×	×	1	0
0	1	0	0	0	1	0	1
7							0

31 TX-FIFO[2] 24							
×	×	×	×	×	×	×	×
×	×	×	×	×	×	×	×
×	×	×	×	×	×	0	0
0	1	1	0	0	1	1	1
7							0

Caution The figure above shows an example of writing with no data padding. If writing is performed with data padding (even if the transmit data bit length is 8 or 16 bits), 32-bit data can be padded (no Don't Care bits are used). For details, see 4.1.3 Data padding.

3.2.11 Receive data registers

These registers (PM0_RXQ: C001_0028H, PM1_RXQ: 400D_0028H) are used to read receive data from the RX data FIFO. If a receive operation is stopped, the value of this register is set to 0000_0000H.

Be sure to read this register in 32-bit units. Operation is not guaranteed if this register is accessed in 8-bit or 16-bit units.

A value of 0 is read from this register if reception is disabled (bit 1 of PMx_TXRX_DIS register = 1).

31	30	29	28	27	26	25	24
RXQ							

23	22	21	20	19	18	17	16
RXQ							

15	14	13	12	11	10	9	8
RXQ							

7	6	5	4	3	2	1	0
RXQ							

Name	R/W	Bit	After Reset	Function
RXQ	R	31:0	0	Indicates RX data FIFO read data.

Remark Data is read from the entry pointed to by the read pointer in the 32-bit × 32-word receive FIFO. The read pointer is automatically controlled by hardware. No data can be written to PMx_RXQ.

An example of reading data from the receive FIFO is provided below.
Data beyond the receive data bit length is set to 0.

Example Receive data bit length = 28 bits (SIB = 1BH in PMx_CYCLE register)
Read data (PMx_RXQ) ← 0123_4567H (RXFIFO[0])
Read data (PMx_RXQ) ← 089A_BCDEH (RXFIFO[1])
Read data (PMx_RXQ) ← 0F24_68ACH (RXFIFO[2])

31 RX-FIFO[0] 24

0	0	0	0	0	0	0	1
0	0	1	0	0	0	1	1
0	1	0	0	0	1	0	1
0	1	1	0	0	1	1	1

7 0

31 RX-FIFO[1] 24

0	0	0	0	1	0	0	0
1	0	0	1	1	0	1	0
1	0	1	1	1	1	0	0
1	1	0	1	1	1	1	0

7 0

31 RX-FIFO[2] 24

0	0	0	0	1	1	1	1
0	0	1	0	0	1	0	0
0	1	1	0	1	0	0	0
1	0	1	0	1	1	0	0

7 0

Caution The figure above shows an example of reading with no data padding. When reading is performed with data padding (even if the receive data bit length is 8 or 16 bits), 32-bit data can be padded. For details, see 4.1.3 Data padding.

3.2.12 Data transfer cycle setting register 2

These registers (PM0_CYCLE2: C001_0030H, PM1_CYCLE2: 400D_0030H) specify the settings for phase 2 in modes 5 and 6. These settings are ignored in other modes.

Set up this register before enabling transmission/reception and at least one frame (one word period in phase 1 or 2, whichever is specified as longer, in modes 5 and 6) after disabling transmission/reception.

31	30	29	28	27	26	25	24
Reserved							

23	22	21	20	19	18	17	16
Reserved			SOB2				

15	14	13	12	11	10	9	8
Reserved			SIB2				

7	6	5	4	3	2	1	0
CYC_VAL2							

Name	R/W	Bit	After Reset	Function
Reserved	R	31:21	0	Reserved. When these bits are read, 0 is returned for each bit.
SOB2	R/W	20:16	0	Specifies the number of valid transmit bits (PMx_SO signal) according to the number of PMx_CLK clock cycles. The number of valid transmit bits is (set value + 1). Setting range: 07H to 1FH (8 to 32 bits)[Note 1]
Reserved	R	15:13	0	Reserved. When these bits are read, 0 is returned for each bit.
SIB2	R/W	12:8	0	Specifies the number of valid receive bits (PMx_SI signal) according to the number of PMx_CLK clock cycles. The number of valid receive bits is (set value + 1). Setting range: 07H to 1FH (8 to 32 bits)[Note 1]
CYC_VAL2	R/W	7:0	0	Specifies the number of words of phase 2 per frame. Modes 5 and 6: 00H to 80H (0 to 128 words)[Note 2]

Notes 1. In mode 5 or 6, set the SOB2 and SIB2 bits to the same values for a simultaneous transmit/receive operation. A value different from phase 1 (SOB and SIB bits of PMx_CYCLE register) can be specified. (The same values must be specified for SOB2 and SIB2. Setting different values to SOB and SOB2 or SIB and SIB2 is allowed.)

2. Setting the CYC_VAL2 bit of the PMx_CYCLE2 register to 00H is prohibited in modes 5 and 6 if the CYC_VAL bit of the PMx_CYCLE register is set to 00H. If CYC_VAL2 is set to 00H, the value set to PMx_CYCLE becomes valid and the single-phase operation is performed. A value different from the setting for phase 1 (CYC_VAL bit of the PMx_CYCLE register) can be specified. (Setting different values to CYC_VAL and CYC_VAL2 is allowed)

3.3 Register Functions (PWM Interface)

In the register descriptions, *x* indicates a channel number and *n* indicates a counter number. Here, x = 0 or 1 and n = 0 to 2.

3.3.1 PWM control registers

(1) PWM operation start/stop register

These registers (PWM_CH0_CTRL: 4010_0000H, PWM_CH1_CTRL: 4010_0100H) starts PWM.

The counter operation starts if the PWM_START bit is set to 1 after the CMP_ENn bit (CMP enable) of the PWM_CHx_MODE register is set to 1. The counter does not start if the PWM_START bit is set to 0 even if the CMP_ENn bit of the PWM_CHx_MODE register is set to 1.

If the PWM_START bit is set to 0 during counter operation, the counter stops after synchronization with PWM_PWCLKx and PWM_OUT becomes 0.

The operation status of each counter can be read by using the PWM_ENn bit.

31	30	29	28	27	26	25	24
Reserved							

23	22	21	20	19	18	17	16
Reserved							

15	14	13	12	11	10	9	8
Reserved							

7	6	5	4	3	2	1	0
Reserved	PWM_EN2	PWM_EN1	PWM_EN0	Reserved			PWM_START

Name	R/W	Bit	After Reset	Function
Reserved	R	31:7	0	Reserved. When these bits are read, 0 is returned for each bit.
PWM_EN2[Note 1]	R	6	0	Status of CMP2 counter 0: Counter has stopped, 1: Counter is running
PWM_EN1[Note 1]	R	5	0	Status of CMP1 counter 0: Counter has stopped, 1: Counter is running
PWM_EN0[Note 1]	R	4	0	Status of CMP0 counter 0: Counter has stopped, 1: Counter is running
Reserved	R	3:1	0	Reserved. When these bits are read, 0 is returned.
PWM_START[Note 2]	R/W	0	0	0: Counter stops, 1: Counter starts

Notes 1. Since the PWM_ENn bit does not synchronize with the system clock, the value of each status may deviate at the point the signals change.

2. If restarting the counter, set the PWM_START bit to 0 and then wait for at least three PWM_PWCLKx clocks.

3.3.2 PWM setting registers

(1) PWM mode control register

These registers (PWM_CH0_MODE: 4010_0004H, PWM_CH1_MODE: 4010_0104H) specify the PWM operation mode.

By setting the PWM_START bit of the PWM_CTRL register to 1 after enabling the CMP_ENn bit, the corresponding CMPn starts operating.

Caution Setting this register is prohibited while the counter is running. The operation is not guaranteed if this register is set when the PWM_START bit of the PWM_Chx_CTRL register is set to 1.

31	30	29	28	27	26	25	24
Reserved							

23	22	21	20	19	18	17	16
Reserved						MOD	

15	14	13	12	11	10	9	8
Reserved					CMP_INV2	CMP_ATST2	CMP_EN2

7	6	5	4	3	2	1	0
Reserved	CMP_INV1	CMP_ATST1	CMP_EN1	Reserved	CMP_INV0	CMP_ATST0	CMP_EN0

(1/2)

Name	R/W	Bit	After Reset	Function
Reserved	R	31:18	0	Reserved. When these bits are read, 0 is returned for each bit.
MOD	R/W	17:16	00B	00B: OR for compare value signals of CMP0 to CMP2 01B: AND for compare value signals of CMP0 to CMP2 10B: XOR for compare value signals of CMP0 to CMP2
Reserved	R	15:11	0	Reserved. When these bits are read, 0 is returned for each bit.
CMP_INV2	R/W	10	0	0: Output of CMP2 compare result 1: Inverted output of CMP2 compare result
CMP_ATST2	R/W	9	0	0: Infinite loop of CMP2 (does not terminate automatically) 1: CMP2 automatically terminates after looping the number of times set in PWM_CHx_LOOP2.
CMP_EN2	R/W	8	0	0: CMP2 disable, 1: CMP2 operation enable If CMP_ATST2 is set to 1, the value automatically becomes 0 after looping.
Reserved	R	7	0	Reserved. When this bit is read, 0 is returned.
CMP_INV1	R/W	6	0	0: Output of CMP1 compare result 1: Inverted output of CMP1 compare result
CMP_ATST1	R/W	5	0	0: Infinite loop of CMP1 (does not terminate automatically) 1: CMP1 automatically terminates after looping the number of times set in PWM_CHx_LOOP1.

(2/2)

Name	R/W	Bit	After Reset	Function
CMP_EN1	R/W	4	0	0: CMP1 disable, 1: CMP1 operation enable If CMP_ATST1 is set to 1, the value automatically becomes 0 after looping.
Reserved	R	3	0	Reserved. When this bit is read, 0 is returned.
CMP_INV0	R/W	2	0	0: Output of CMP0 compare result 1: Inverted output of CMP0 compare result
CMP_ATST0	R/W	1	0	0: Infinite loop of CMP0 (does not terminate automatically) 1: CMP0 automatically terminates after looping the number of times set in PWM_CHx_LOOP0.
CMP_EN0	R/W	0	0	0: CMP0 disable, 1: CMP0 operation enable If CMP_ATST0 is set to 1, the value automatically becomes 0 after looping.

The PWM output is as follows, depending on the setting of the MOD bit of the PWM_CHx_MODE register and the CMP0_ENn bit of the PWM_CHx_MODE register.

	OR Specification MOD = 00B	AND Specification MOD = 01B	XOR Specification MOD = 10B	Setting Prohibited MOD = 11B
CMP0 to CMP2 disable CMP_EN = 000B	0	0	0	–
Only CMP0 enable CMP_EN = 001B	CMP0	CMP0	CMP0	–
CMP0 and CMP1 enable CMP_EN = 011B	CMP0 | CMP1	CMP0 & CMP1	CMP0 ^ CMP1	–
CMP0 to CMP2 enable CMP_EN = 111B	CMP0 | CMP1 | CMP2	CMP0 & CMP1 & CMP2	CMP0 ^ CMP1 ^ CMP2	–

(2) Channel *x* counter *n* delay setting registers

These registers specify the delay after CMPn starts operating. The starting time of the counter can be delayed for PWM_PWCLKn cycles × DELAY.

- PWM_CH0_DELAY0: 4010_0010H
- PWM_CH0_DELAY1: 4010_0040H
- PWM_CH0_DELAY2: 4010_0080H
- PWM_CH1_DELAY0: 4010_0110H
- PWM_CH1_DELAY1: 4010_0140H
- PWM_CH1_DELAY2: 4010_0180H

31	30	29	28	27	26	25	24
DELAY							

23	22	21	20	19	18	17	16
DELAY							

15	14	13	12	11	10	9	8
DELAY							

7	6	5	4	3	2	1	0
DELAY							

Name	R/W	Bit	After Reset	Function
DELAY	R/W	31:0	0	Delay value setting bit 0000_0000H: No delay. No delay period is set. 0000_0001H: Delays PWM_PWCLK × 1 cycle from the reference time (delays for 83.3 ns). 0000_0002H: Delays PWM_PWCLK × 2 cycles from the reference time (delays for 166.6 ns). : 0000_FFFFH: Delays PWM_PWCLK × 66,535 cycles from the reference time (delays for 5.5423 ms). : FFFF_FFFFH: Delays PWM_PWCLK × 4,294,967,295 cycles from the reference time (delays for 357.7707757 s).

Remark The values in parentheses refer to when the frequency of PWM_PWCLKx is 12 MHz.

(3) Channel *x* counter *n* leading edge setting registers

These registers specify the leading edge of counter n. A low level (high level if inverted) is output from CMPn for the period of PWM_PWCLKn cycles × LEDGE from the next cycle after the delay period has finished.
See **Figure 4-10 Timing When Only Counter 0 Runs.**

- PWM_CH0_LEDGE0: 4010_0014H
- PWM_CH0_LEDGE1: 4010_0044H
- PWM_CH0_LEDGE2: 4010_0084H
- PWM_CH1_LEDGE0: 4010_0114H
- PWM_CH1_LEDGE1: 4010_0144H
- PWM_CH1_LEDGE2: 4010_0184H

31	30	29	28	27	26	25	24
LEDGE							

23	22	21	20	19	18	17	16
LEDGE							

15	14	13	12	11	10	9	8
LEDGE							

7	6	5	4	3	2	1	0
LEDGE							

Name	R/W	Bit	After Reset	Function
LEDGE[Note]	R/W	31:0	0	Leading edge period setting bit 0000_0000H: No LEDGE period when this value is set. 0000_0001H: Outputs low level for PWM_PWCLK × 1 cycle (83.3 ns). 0000_0002H: Outputs low level for PWM_PWCLK × 2 cycles (166.6 ns). : 0000_FFFFH: Outputs low level for PWM_PWCLK × 66,535 cycles (5.5423 ms). : FFFF_FFFFH: Outputs low level for PWM_PWCLK × 4,294,967,295 cycles (357.7707757 s).

Note The following restriction applies: PWM_CHx_LEDGEn < PWM_CHx_TEDGEn ≤ PWM_CHx_TOTALn.

Remark The values in the parentheses refer to when the frequency of PWM_PWCLKx is 12 MHz.

(4) Channel *x* counter *n* trailing edge setting registers

These registers specify the trailing edge of counter n. A high level (low level if inverted) is output from CMPn for the period specified by TEDGE from the next cycle after the LEDGE period has finished.

The pulse width is the period of the PWM_PWCLK cycles × (TEDGE − LEDGE). See **Figure 4-10 Timing When Only Counter 0 Runs**.

- PWM_CH0_TEDGE0: 4010_0018H
- PWM_CH0_TEDGE1: 4010_0048H
- PWM_CH0_TEDGE2: 4010_0088H
- PWM_CH1_TEDGE0: 4010_0118H
- PWM_CH1_TEDGE1: 4010_0148H
- PWM_CH1_TEDGE2: 4010_0188H

31	30	29	28	27	26	25	24
TEDGE							

23	22	21	20	19	18	17	16
TEDGE							

15	14	13	12	11	10	9	8
TEDGE							

7	6	5	4	3	2	1	0
TEDGE							

Name	R/W	Bit	After Reset	Function
TEDGE[Note 1]	R/W	31:0	0[Note 2]	Trailing edge period setting bit The following is output when LEDGE = 0. 0000_0000H: Setting prohibited[Note 2] 0000_0001H: Outputs high level for PWM_PWCLK × 1 cycle (83.3 ns). 0000_0002H: Outputs high level for PWM_PWCLK × 2 cycles (166.6 ns). : 0000_FFFFH: Outputs high level for PWM_PWCLK × 66,535 cycles (5.5423 ms). : FFFF_FFFFH: Outputs high level for PWM_PWCLK × 4,294,967,295 cycles (357.7707757 s).

Notes 1. The following restriction applies: PWM_CHx_LEDGEn < PWM_CHx_TEDGEn ≤ PWM_CHx_TOTALn.

2. Setting of After Reset value (0000_0000H) is prohibited if PWM_CHx_LEDGEn is set to 0. Be sure to use the updated value.

Remark The values in the parentheses refer to when the frequency of PWM_PWCLKx is 12 MHz.

(5) Channel *x* counter *n* total cycle setting registers

These registers specify one cycle of counter n. The length of one cycle is PWM_PWCLK cycles × TOTAL. Setting 0000_0000H is prohibited. The operation is not guaranteed if this value is set.

- PWM_CH0_TOTAL0: 4010_001CH
- PWM_CH0_TOTAL1: 4010_004CH
- PWM_CH0_TOTAL2: 4010_008CH
- PWM_CH1_TOTAL0: 4010_011CH
- PWM_CH1_TOTAL1: 4010_014CH
- PWM_CH1_TOTAL2: 4010_018CH

31	30	29	28	27	26	25	24
TOTAL							

23	22	21	20	19	18	17	16
TOTAL							

15	14	13	12	11	10	9	8
TOTAL							

7	6	5	4	3	2	1	0
TOTAL							

Name	R/W	Bit	After Reset	Function
TOTAL[Note 1]	R/W	31:0	0[Note 2]	Total cycle setting bit 0000_0000H: Setting prohibited[Note 2] 0000_0001H: The cycle is PWM_PWCLK × 1 cycle (83.3 ns) 0000_0002H: The cycle is PWM_PWCLK × 2 cycles (166.6 ns) : 0000_FFFFH: The cycle is PWM_PWCLK × 66,535 cycles (5.5423 ms) : FFFF_FFFFH: The cycle is PWM_PWCLK × 4,294,967,295 cycles (357.7707757 s)

Notes 1. The following restriction applies: PWM_CHx_LEDGEn < PWM_CHx_TEDGEn ≤ PWM_CHx_TOTALn.

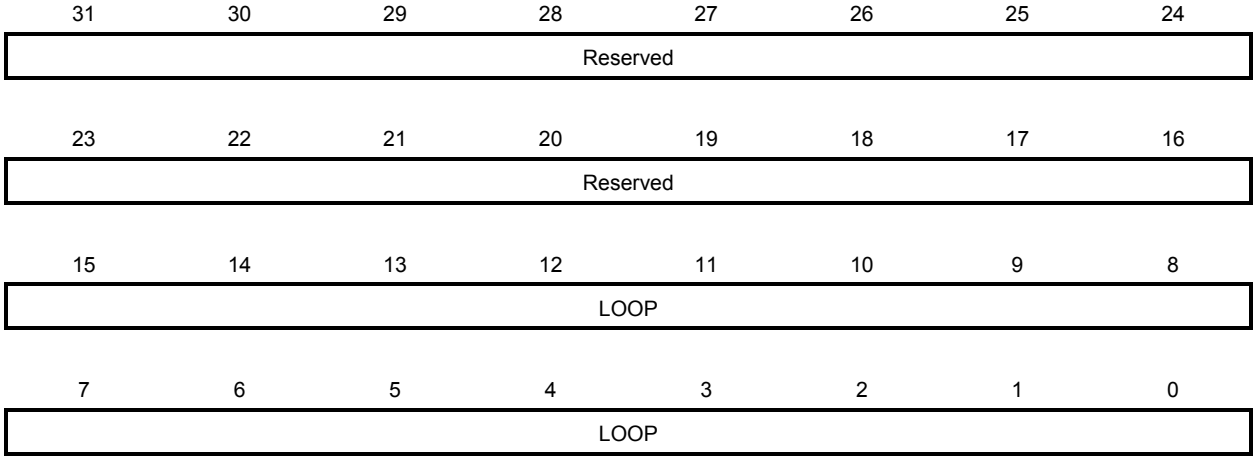
2. Setting of After Reset value (0000_0000H) is prohibited. Be sure to use the updated value.

Remark The values in the parentheses refer to when the frequency of PWM_PWCLKx is 12 MHz.

(6) Channel *x* counter *n* loop count setting register

These registers specify the number of cycles to be repeated. The counter automatically stops after repetition is over.

- PWM_CH0_LOOP0: 4010_0020H
- PWM_CH0_LOOP1: 4010_0050H
- PWM_CH0_LOOP2: 4010_0090H
- PWM_CH1_LOOP0: 4010_0120H
- PWM_CH1_LOOP1: 4010_0150H
- PWM_CH1_LOOP2: 4010_0190H



Name	R/W	Bit	After Reset	Function
Reserved	R/W	31:16	0	Reserved. When these bits are read, 0 is returned for each bit.
LOOP[Note 1]	R/W	15:0	0[Note 2]	0000H: Setting prohibited[Note 2] 0001H: The counter stops after one count. 0002H: The counter stops after two repetitions. : FFFFH: The cycle counter stops after 66,535 repetitions.

Notes 1. The counter does not automatically stop if the CMP_ATSTn bit of the PWM_CHx_MODE register is set to 0 even if a value is set. If the CMP_ATSTn bit is set to 1, setting 0000H is prohibited. Be sure to set a value of 1 or more.

2. Setting of the value after reset (0000_0000H) is prohibited. Be sure to use the updated value.

3.3.3 PWM interrupt registers

The PWM interface has two interrupts for each counter.

- Count completion interrupt: Generated for each count until the counter reaches the total cycle.
- Loop completion interrupt: Generated when the specified number of loops are complete when loops are specified.

The PWM_INT pin outputs each ORed interrupt status (PWM_INTSTATUS) signal.

(1) PWM interrupt status register

This register (PWM_INTSTATUS: 4010_0400H) indicates the status of the interrupt sources that are enabled in the PWM_INTENSET register.

If a source is disabled, 0 is read.

31	30	29	28	27	26	25	24
Reserved							

23	22	21	20	19	18	17	16
Reserved							

15	14	13	12	11	10	9	8
Reserved		Ch1_CMP2_ LENDSTATUS	Ch1_CMP2_ CENDSTATUS	Ch1_CMP1_ LENDSTATUS	Ch1_CMP1_ CENDSTATUS	Ch1_CMP0_ LENDSTATUS	Ch1_CMP0_ CENDSTATUS

7	6	5	4	3	2	1	0
Reserved		Ch0_CMP2_ LENDSTATUS	Ch0_CMP2_ CENDSTATUS	Ch0_CMP1_ LENDSTATUS	Ch0_CMP1_ CENDSTAUS	Ch0_CMP0_ LENDSTATUS	Ch0_CMP0_ CENDSTATUS

(1/2)

Name	R/W	Bit	After Reset	Function
Reserved	R	31:14	0	Reserved. When these bits are read, 0 is returned for each bit.
Ch1_CMP2_LENDSTATUS	R	13	0	Indicates the status of the loop completion interrupt for CMP2 on channel 1.
Ch1_CMP2_CENDSTATUS	R	12	0	Indicates the status of the count completion interrupt for CMP2 on channel 1.
Ch1_CMP1_LENDSTATUS	R	11	0	Indicates the status of the loop completion interrupt for CMP1 on channel 1.
Ch1_CMP1_CENDSTATUS	R	10	0	Indicates the status of the count completion interrupt for CMP1 on channel 1.
Ch1_CMP0_LENDSTATUS	R	9	0	Indicates the status of the loop completion interrupt for CMP0 on channel 1.
Ch1_CMP0_CENDSTATUS	R	8	0	Indicates the status of the count completion interrupt for CMP0 on channel 1.
Reserved	R	7:6	0	Reserved. When these bits are read, 0 is returned for each bit.
Ch0_CMP2_LENDSTATUS	R	5	0	Indicates the status of the loop completion interrupt for CMP2 on channel 0.

(2/2)

Name	R/W	Bit	After Reset	Function
Ch0_CMP2_CENDSTATUS	R	4	0	Indicates the status of the count completion interrupt for CMP2 on channel 0.
Ch0_CMP1_LENDSTATUS	R	3	0	Indicates the status of the loop completion interrupt for CMP1 on channel 0.
Ch0_CMP1_CENDSTAUS	R	2	0	Indicates the status of the count completion interrupt for CMP1 on channel 0.
Ch0_CMP0_LENDSTATUS	R	1	0	Indicates the status of the loop completion interrupt for CMP0 on channel 0.
Ch0_CMP0_CENDSTATUS	R	0	0	Indicates the status of the count completion interrupt for CMP0 on channel 0.

Remark 0: No interrupt request was issued.
1: An interrupt request was issued.

(2) PWM interrupt raw status register

This register (PWM_INTRAWSTATUS: 4010_0404H) indicates the interrupt source, regardless of whether the interrupt source is enabled in PWM_INTENSET.

A pending interrupt source can be monitored by reading this register.

31	30	29	28	27	26	25	24
Reserved							

23	22	21	20	19	18	17	16
Reserved							

15	14	13	12	11	10	9	8
Reserved		CH1_CMP2_LEND_RAWSTATUS	CH1_CMP2_CEND_RAWSTATUS	CH1_CMP1_LEND_RAWSTATUS	CH1_CMP1_CEND_RAWSTATUS	CH1_CMP0_LEND_RAWSTATUS	CH1_CMP0_CEND_RAWSTATUS

7	6	5	4	3	2	1	0
Reserved		CH0_CMP2_LEND_RAWSTATUS	CH0_CMP2_CEND_RAWSTATUS	CH0_CMP1_LEND_RAWSTATUS	CH0_CMP1_CEND_RAWSTATUS	CH0_CMP0_LEND_RAWSTATUS	CH0_CMP0_CEND_RAWSTATUS

(1/2)

Name	R/W	Bit	After Reset	Function
Reserved	R	31:14	0	Reserved. When these bits are read, 0 is returned for each bit.
CH1_CMP2_LEND_RAWSTATUS	R	13	0	Indicates the raw status of the loop completion interrupt for CMP2 on channel 1.
CH1_CMP2_CEND_RAWSTATUS	R	12	0	Indicates the raw status of the count completion interrupt for CMP2 on channel 1.
CH1_CMP1_LEND_RAWSTATUS	R	11	0	Indicates the raw status of the loop completion interrupt for CMP1 on channel 1.
CH1_CMP1_CEND_RAWSTATUS	R	10	0	Indicates the raw status of the count completion interrupt for CMP1 on channel 1.
CH1_CMP0_LEND_RAWSTATUS	R	9	0	Indicates the raw status of the loop completion interrupt for CMP0 on channel 1.
CH1_CMP0_CEND_RAWSTATUS	R	8	0	Indicates the raw status of the count completion interrupt for CMP0 on channel 1.
Reserved	R	7:6	0	Reserved. When these bits are read, 0 is returned for each bit.
CH0_CMP2_LEND_RAWSTATUS	R	5	0	Indicates the raw status of the loop completion interrupt for CMP2 on channel 0.
CH0_CMP2_CEND_RAWSTATUS	R	4	0	Indicates the raw status of the count completion interrupt for CMP2 on channel 0.
CH0_CMP1_LEND_RAWSTATUS	R	3	0	Indicates the raw status of the loop completion interrupt for CMP1 on channel 0.
CH0_CMP1_CEND_RAWSTATUS	R	2	0	Indicates the raw status of the count completion interrupt for CMP1 on channel 0.

(2/2)

Name	R/W	Bit	After Reset	Function
CH0_CMP0_LEND_RAWSTATUS	R	1	0	Indicates the raw status of the loop completion interrupt for CMP0 on channel 0.
CH0_CMP0_CEND_RAWSTATUS	R	0	0	Indicates the raw status of the count completion interrupt for CMP0 on channel 0.

Caution If an interrupt source is set and cleared at the same time, setting takes precedence. However, if the setting interval for the same source is "PWM_PCLK cycle × 4 + PWM_PWCLKx cycle × 4" or shorter, the source may not be set after the source is cleared. In this case, the source is set at the next source setting timing.

Remark 0: No interrupt request was issued.
1: An interrupt request was issued.

(3) PWM interrupt enable set register

This register (PWM_INTENSET: 4010_0408H) enables the interrupts. The register values are updated only for the bits to which 1 is set. The current enable/disable status can be checked by reading this register.

31	30	29	28	27	26	25	24
Reserved							

23	22	21	20	19	18	17	16
Reserved							

15	14	13	12	11	10	9	8
Reserved		CH1_CMP2_ LENDSET	CH1_CMP2_ CENDSET	CH1_CMP1_ LENDSET	CH1_CMP1_ CENDSET	CH1_CMP0_ LENDSET	CH1_CMP0_ CENDSET

7	6	5	4	3	2	1	0
Reserved		CH0_CMP2_ LENDSET	CH0_CMP2_ CENDSET	CH0_CMP1_ LENDSET	CH0_CMP1_ CENDSET	CH0_CMP0_ LENDSET	CH0_CMP0_ CENDSET

(1/2)

Name	R/W	Bit	After Reset	Function
Reserved	R	31:14	0	Reserved. When these bits are read, 0 is returned for each bit.
CH1_CMP2_LENDSET	R/W	13	0	Controls whether to enable the loop completion interrupt for CMP2 on channel 1.
CH1_CMP2_CENDSET	R/W	12	0	Controls whether to enable the count completion interrupt for CMP2 on channel 1.
CH1_CMP1_LENDSET	R/W	11	0	Controls whether to enable the loop completion interrupt for CMP1 on channel 1.
CH1_CMP1_CENDSET	R/W	10	0	Controls whether to enable the count completion interrupt for CMP1 on channel 1.
CH1_CMP0_LENDSET	R/W	9	0	Controls whether to enable the loop completion interrupt for CMP0 on channel 1.
CH1_CMP0_CENDSET	R/W	8	0	Controls whether to enable the count completion interrupt for CMP0 on channel 1.
Reserved	R	7:6	0	Reserved. When these bits are read, 0 is returned for each bit.
CH0_CMP2_LENDSET	R/W	5	0	Controls whether to enable the loop completion interrupt for CMP2 on channel 0.
CH0_CMP2_CENDSET	R/W	4	0	Controls whether to enable the count completion interrupt for CMP2 on channel 0.
CH0_CMP1_LENDSET	R/W	3	0	Controls whether to enable the loop completion interrupt for CMP1 on channel 0.
CH0_CMP1_CENDSET	R/W	2	0	Controls whether to enable the count completion interrupt for CMP1 on channel 0.
CH0_CMP0_LENDSET	R/W	1	0	Controls whether to enable the loop completion interrupt for CMP0 on channel 0.

(2/2)

Name	R/W	Bit	After Reset	Function
CH0_CMP0_CENDSET	R/W	0	0	Controls whether to enable the count completion interrupt for CMP0 on channel 0.

Remark When written: The interrupt enable bit is set or retained.

1: The interrupt enable bit is set.

0: The interrupt enable bit is retained.

When read: The status of the interrupt enable bit is indicated.

1: Interrupt request issuance is enabled.

0: Interrupt request issuance is disabled.

(4) PWM interrupt enable clear register

This register (PWM_INTENCLR: 4010_040CH) clears the interrupt source enable bits. The register values are updated only for the bits to which 1 is set. 0 is returned when this register is read since it is write-only.

31	30	29	28	27	26	25	24
Reserved							

23	22	21	20	19	18	17	16
Reserved							

15	14	13	12	11	10	9	8
Reserved		CH1_CMP2_ LEND_ENCLR	CH1_CMP2_ CEND_ENCLR	CH1_CMP1_ LEND_ENCLR	CH1_CMP1_ CEND_ENCLR	CH1_CMP0_ LEND_ENCLR	CH1_CMP0_ CEND_ENCLR

7	6	5	4	3	2	1	0
Reserved		CH0_CMP2_ LEND_ENCLR	CH0_CMP2_ CEND_ENCLR	CH0_CMP1_ LEND_ENCLR	CH0_CMP1_ CEND_ENCLR	CH0_CMP0_ LEND_ENCLR	CH0_CMP0_ CEND_ENCLR

Name	R/W	Bit	After Reset	Function
Reserved	R	31:14	0	Reserved. When these bits are read, 0 is returned for each bit.
CH1_CMP2_LEND_ENCLR	W	13	0	Clears the loop completion interrupt enable bit for CMP2 on channel 1.
CH1_CMP2_CEND_ENCLR	W	12	0	Clears the count completion interrupt enable bit for CMP2 on channel 1.
CH1_CMP1_LEND_ENCLR	W	11	0	Clears the loop completion interrupt enable bit for CMP1 on channel 1.
CH1_CMP1_CEND_ENCLR	W	10	0	Clears the count completion interrupt enable bit for CMP1 on channel 1.
CH1_CMP0_LEND_ENCLR	W	9	0	Clears the loop completion interrupt enable bit for CMP0 on channel 1.
CH1_CMP0_CEND_ENCLR	W	8	0	Clears the count completion interrupt enable bit for CMP0 on channel 1.
Reserved	R	7:6	0	Reserved. When these bits are read, 0 is returned for each bit.
CH0_CMP2_LEND_ENCLR	W	5	0	Clears the loop completion interrupt enable bit for CMP2 on channel 0.
CH0_CMP2_CEND_ENCLR	W	4	0	Clears the count completion interrupt enable bit for CMP2 on channel 0.
CH0_CMP1_LEND_ENCLR	W	3	0	Clears the loop completion interrupt enable bit for CMP1 on channel 0.
CH0_CMP1_CEND_ENCLR	W	2	0	Clears the count completion interrupt enable bit for CMP1 on channel 0.
CH0_CMP0_LEND_ENCLR	W	1	0	Clears the loop completion interrupt enable bit for CMP0 on channel 0.
CH0_CMP0_CEND_ENCLR	W	0	0	Clears the count completion interrupt enable bit for CMP0 on channel 0.

Remark 1: The interrupt enable bit is cleared (interrupt masked).
0: The interrupt enable bit is retained.

(5) PWM interrupt source clear register

This register (PWM_INTFFCLR: 4010_0410H) clears the generated interrupt sources. The register values are updated only for the bits to which 1 is set. 0 is returned when this register is read since it is write-only.

31	30	29	28	27	26	25	24
Reserved							

23	22	21	20	19	18	17	16
Reserved							

15	14	13	12	11	10	9	8
Reserved		CH1_CMP2_ LEND_FFCLR	CH1_CMP2_ CEND_FFCLR	CH1_CMP1_ LEND_FFCLR	CH1_CMP1_ CEND_FFCLR	CH1_CMP0_ LEND_FFCLR	CH1_CMP0_ CEND_FFCLR

7	6	5	4	3	2	1	0
Reserved		CH0_CMP2_ LEND_FFCLR	CH0_CMP2_ CEND_FFCLR	CH0_CMP1_ LEND_FFCLR	CH0_CMP1_ CEND_FFCLR	CH0_CMP0_ LEND_FFCLR	CH0_CMP0_ CEND_FFCLR

Name	R/W	Bit	After Reset	Function
Reserved	R	31:14	0	Reserved. When these bits are read, 0 is returned for each bit.
CH1_CMP2_LEND_FFCLR	W	13	0	Clears the loop completion interrupt source bit for CMP2 on channel 1.
CH1_CMP2_CEND_FFCLR	W	12	0	Clears the count completion interrupt source bit for CMP2 on channel 1.
CH1_CMP1_LEND_FFCLR	W	11	0	Clears the loop completion interrupt source bit for CMP1 on channel 1.
CH1_CMP1_CEND_FFCLR	W	10	0	Clears the count completion interrupt source bit for CMP1 on channel 1.
CH1_CMP0_LEND_FFCLR	W	9	0	Clears the loop completion interrupt source bit for CMP0 on channel 1.
CH1_CMP0_CEND_FFCLR	W	8	0	Clears the count completion interrupt source bit for CMP0 on channel 1.
Reserved	R	7:6	0	Reserved. When these bits are read, 0 is returned for each bit.
CH0_CMP2_LEND_FFCLR	W	5	0	Clears the loop completion interrupt source bit for CMP2 on channel 0.
CH0_CMP2_CEND_FFCLR	W	4	0	Clears the count completion interrupt source bit for CMP2 on channel 0.
CH0_CMP1_LEND_FFCLR	W	3	0	Clears the loop completion interrupt source bit for CMP1 on channel 0.
CH0_CMP1_CEND_FFCLR	W	2	0	Clears the count completion interrupt source bit for CMP1 on channel 0.
CH0_CMP0_LEND_FFCLR	W	1	0	Clears the loop completion interrupt source bit for CMP0 on channel 0.
CH0_CMP0_CEND_FFCLR	W	0	0	Clears the count completion interrupt source bit for CMP0 on channel 0.

Remark 1: The interrupt source is cleared.
0: The interrupt source is retained.

CHAPTER 4 DESCRIPTION OF FUNCTIONS

4.1 Audio/Voice Interface

The audio/voice interface is a serial interface used to connect with an external voice codec device or audio codec device. The audio/voice interface can operate in two modes. One is slave mode, in which the audio/voice interface receives a serial clock and frame synchronization signal from an external device. The other is master mode, in which the audio/voice interface generates a serial clock and frame synchronization signal. The audio/voice interface supports seven serial interface timing patterns.

The audio/voice interface has two FIFO buffers, one for transmission and one for reception, and an interface (DMA request issuance function) for transferring data to and from the DMA controller.

4.1.1 Master mode/slave mode setting

Either the master mode or slave mode can be selected for the serial clock (PMx_CLK) and frame synchronization signal (PMx_SEN). Use the M_S bit of the operation mode setting register (PMx_FUNC_SEL) to select the master/slave mode. In master mode, a serial clock and frame synchronization signal are generated and output by the audio/voice interface. In slave mode, a serial clock and frame synchronization signal are received from an external device.

In the initial status after reset release, neither the master mode nor the slave mode is selected. When the audio/voice interface is used, one of the modes needs to be selected using the M_S bit of the PMx_FUNC_SEL register.

4.1.2 Serial interface timing

One of seven serial interface modes (modes 0 to 6) is selected to perform serial data transmission/reception by using the interface timing pattern corresponding to the selected mode. Use the MODE_SEL bit of the PMx_FUNC_SEL register also to select the serial interface mode. The frame synchronization signal period, the number of valid receive bits, and the number of valid transmit bits need to be set.

- Frame synchronization signal period setting:
  Set the frame period by using the CYC_VAL bits of the data transfer cycle setting register (PMx_CYCLE).
  The value of the frame synchronization signal period is CYC_VAL plus 1.
- Number of valid receive bits (for modes 0 to 4):
  Set the number of valid receive bits by using the SIB bits of the PMx_CYCLE register.
  The number of valid receive bits is SIB plus 1. A value ranging from 8 to 32 can be set as the number of valid receive bits.
  CYC_VAL ≥ SIB must be satisfied.
- Number of valid receive bits (for modes 5 and 6):
  For phase 1, the number of valid receive bits is set by using the SIB bits of the PMx_CYCLE register. For phase 2, the number of valid receive bits is set by using the SIB2 bits of the PMx_CYCLE2 register.
  The number of valid receive bits is SIB or SIB2 plus 1. A value ranging from 8 to 32 can be set as the number of valid receive bits.
  The same value must be set for SOB and SIB, and the same value must be set for SOB2 and SIB2 (SOB = SIB, SOB2 = SIB2).
- Number of valid transmit bits (for modes 0 to 4):
  Set the number of valid transmit bits by using the SOB bits of the PMx_CYCLE register.
  The number of valid transmit bits is SOB plus 1. A value ranging from 8 to 32 can be set as the number of valid transmit bits. CYC_VAL ≥ SOB must be satisfied.

- Number of valid transmit bits (for modes 5 and 6):
  For phase 1, the number of valid transmit bits is set by using the SOB bits of the PMx_CYCLE register. For phase 2, the number of valid transmit bits is set by using the SOB2 bits of the PMx_CYCLE2 register.
  The number of valid transmit bits is SOB or SOB2 plus 1. A value ranging from 8 to 32 can be set as the number of valid transmit bits.
  The same value must be set for SOB and SIB, and the same value must be set for SOB2 and SIB2 (SOB = SIB, SOB2 = SIB2).

In all modes, serial I/O data is transferred MSB-first. In the initial status after reset release, mode 0 is set.
Table 4-1 shows the signal output timing and signal sampling timing in each mode.

Table 4-1. Signal Output Timing and Signal Sampling Timing

PMx_FUNC_SEL.CLK_INV = 0

Mode	PMx_SEN (Master Mode) Output Timing	PMx_SO Output Timing	PMx_SEN (Slave Mode)/ PMx_SI Sampling Timing
Mode 0	Rising edge of PMx_CLK		Falling edge of PMx_CLK
Mode 1	Falling edge of PMx_CLK	Rising edge of PMx_CLK	Rising edge of PMx_CLK
Mode 2	Falling edge of PMx_CLK		Rising edge of PMx_CLK
Mode 3			
Mode 4			
Mode 5	Rising edge of PMx_CLK		Falling edge of PMx_CLK
Mode 6	Falling edge of PMx_CLK	Rising edge of PMx_CLK	Rising edge of PMx_CLK

PMx_FUNC_SEL.CLK_INV = 1

Mode	PMx_SEN (Master Mode) Output Timing	PMx_SO Output Timing	PMx_SEN (Slave Mode)/ PMx_SI Sampling Timing
Mode 0	Falling edge of PMx_CLK		Rising edge of PMx_CLK
Mode 1	Rising edge of PMx_CLK	Falling edge of PMx_CLK	Falling edge of PMx_CLK
Mode 2	Rising edge of PMx_CLK		Falling edge of PMx_CLK
Mode 3			
Mode 4			
Mode 5	Falling edge of PMx_CLK		Rising edge of PMx_CLK
Mode 6	Rising edge of PMx_CLK	Falling edge of PMx_CLK	Falling edge of PMx_CLK

(1) Mode 0 serial interface timing

Figure 4-1 shows the serial interface timing applicable when mode 0 is selected.

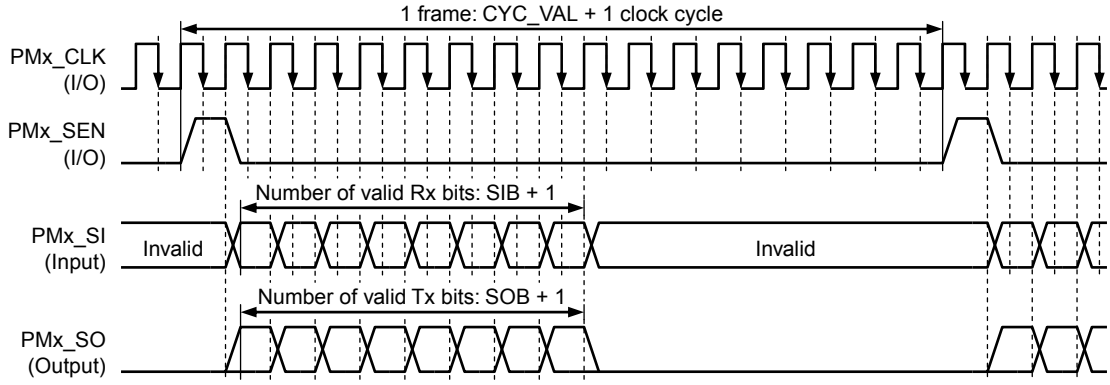
(a-1) Frame synchronization signal (PMx_SEN) output timing in master mode
The PMx_SEN signal is output at the rising edge of the serial clock signal (PMx_CLK).

(a-2) PMx_SEN signal sampling timing in slave mode
The PMx_SEN signal is sampled at the falling edge of the PMx_CLK signal.

(b) Transmit data (PMx_SO) output timing
Transmit data (PMx_SO) is output at the rising edge of the PMx_CLK signal.

(c) Receive data (PMx_SI) sampling timing
Receive data (PMx_SI) is sampled at the falling edge of the PMx_CLK signal.

Figure 4-1. Serial Interface Timing When Mode 0 Is Selected



(2) Mode 1 serial interface timing

Figure 4-2 shows the serial interface timing applicable when mode 1 is selected.

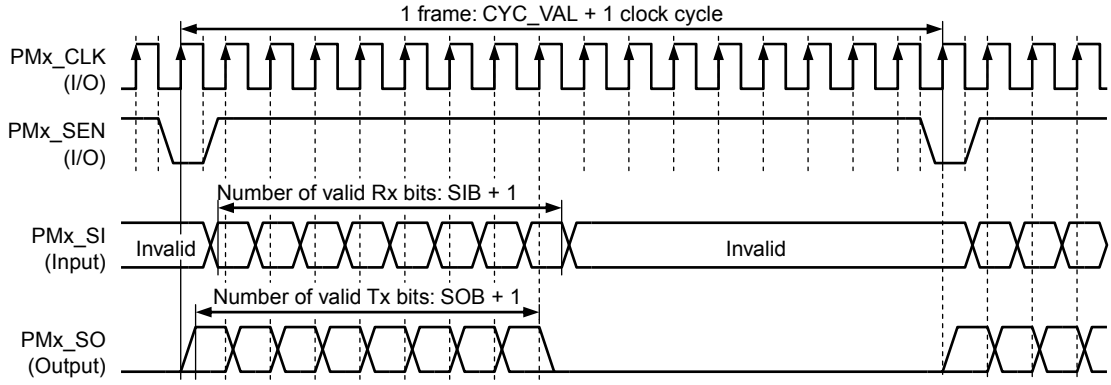
(a-1) Frame synchronization signal (PMx_SEN) output timing in master mode
The PMx_SEN signal is output at the falling edge of the serial clock signal (PMx_CLK).

(a-2) PMx_SEN signal sampling timing in slave mode
The PMx_SEN signal is sampled at the rising edge of the PMx_CLK signal.

(b) Transmit data (PMx_SO) output timing
Transmit data (PMx_SO) is output at the rising edge of the PMx_CLK signal.

(c) Receive data (PMx_SI) sampling timing
Receive data (PMx_SI) is sampled at the rising edge of the PMx_CLK signal.

Figure 4-2. Serial Interface Timing When Mode 1 Is Selected



(3) Mode 2 serial interface timing

Figure 4-3 shows the serial interface timing applicable when mode 2 is selected. The L-ch signal is transferred while PMx_SEN is low. The R-ch signal is transferred while PMx_SEN is high.

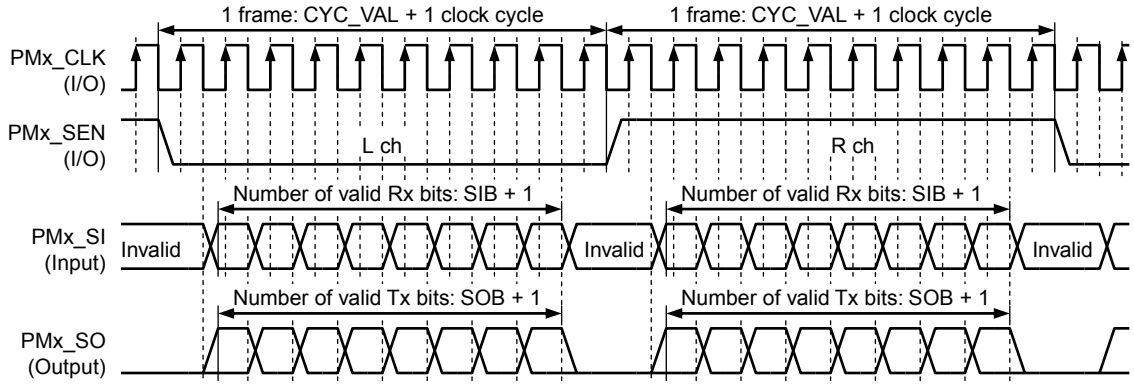
(a-1) Frame synchronization signal (PMx_SEN) output timing in master mode
The PMx_SEN signal is output at the falling edge of the serial clock signal (PMx_CLK).

(a-2) PMx_SEN sampling timing in slave mode
The PMx_SEN signal is sampled at the rising edge of the PMx_CLK signal.

(b) Transmit data (PMx_SO) output timing
Transmit data (PMx_SO) is output at the falling edge of the PMx_CLK signal.

(c) Receive data (PMx_SI) sampling timing
Receive data (PMx_SI) is sampled at the rising edge of the PMx_CLK signal.

Figure 4-3. Serial Interface Timing When Mode 2 Is Selected



(4) Mode 3 serial interface timing

Figure 4-4 shows the serial interface timing applicable when mode 3 is selected. The L-ch signal is transferred while PMx_SEN is high. The R-ch signal is transferred while PMx_SEN is low.

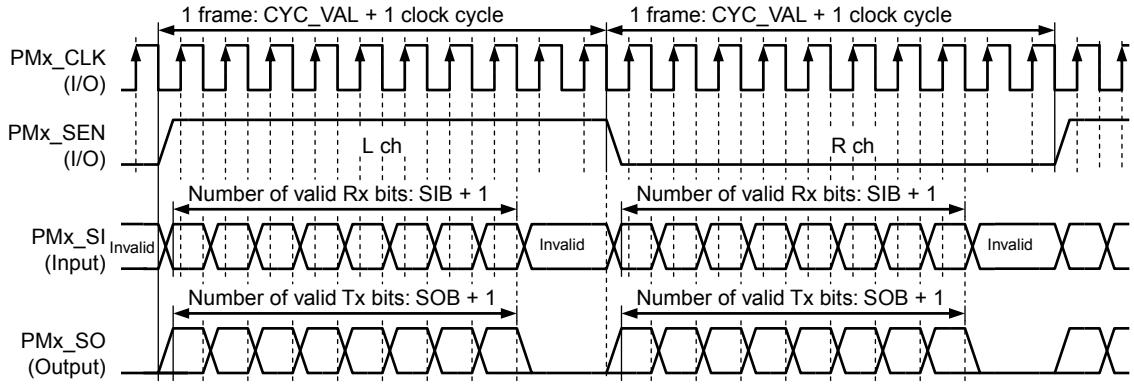
(a-1) Frame synchronization signal (PMx_SEN) output timing in master mode
The PMx_SEN signal is output at the falling edge of the serial clock signal (PMx_CLK).

(a-2) PMx_SEN signal sampling timing in slave mode
The PMx_SEN signal is sampled at the rising edge of the PMx_CLK signal.

(b) Transmit data (PMx_SO) output timing
Transmit data (PMx_SO) is output at the falling edge of the PMx_CLK signal.

(c) Receive data (PMx_SI) sampling timing
Receive data (PMx_SI) is sampled at the rising edge of the PMx_CLK signal.

Figure 4-4. Serial Interface Timing When Mode 3 Is Selected



(5) Mode 4 serial interface timing

Figure 4-5 shows the serial interface timing applicable when mode 4 is selected. The L-ch signal is transferred while PMx_SEN is high. The R-ch signal is transferred while PMx_SEN is low.

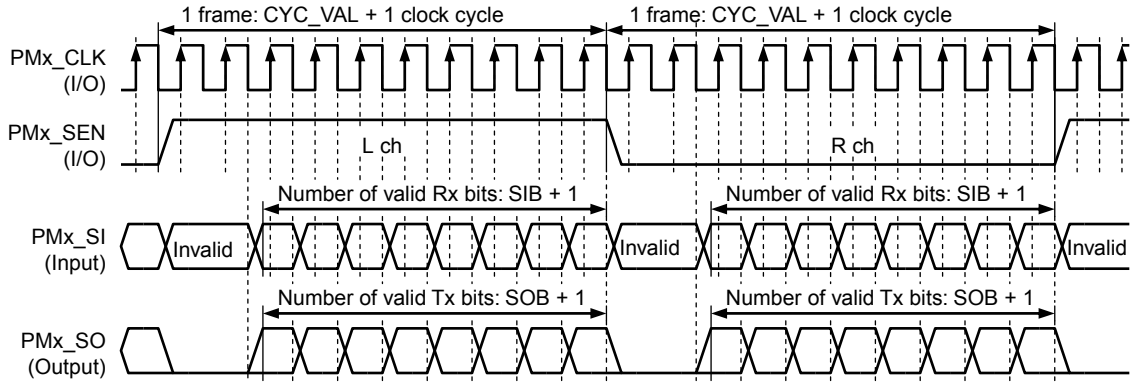
(a-1) Frame synchronization signal (PMx_SEN) output timing in master mode
The PMx_SEN signal is output at the falling edge of the serial clock signal (PMx_CLK).

(a-2) PMx_SEN signal sampling timing in slave mode
The PMx_SEN signal is sampled at the rising edge of the PMx_CLK signal.

(b) Transmit data (PMx_SO) output timing
Transmit data (PMx_SO) is output on the falling edge of the PMx_CLK signal.

(c) Receive data (PMx_SI) sampling timing
Receive data (PMx_SI) is sampled at the rising edge of the PMx_CLK signal.

Figure 4-5. Serial Interface Timing When Mode 4 Is Selected



(6) Mode 5 serial interface timing

Figure 4-6 shows the serial interface timing applicable when mode 5 is selected.

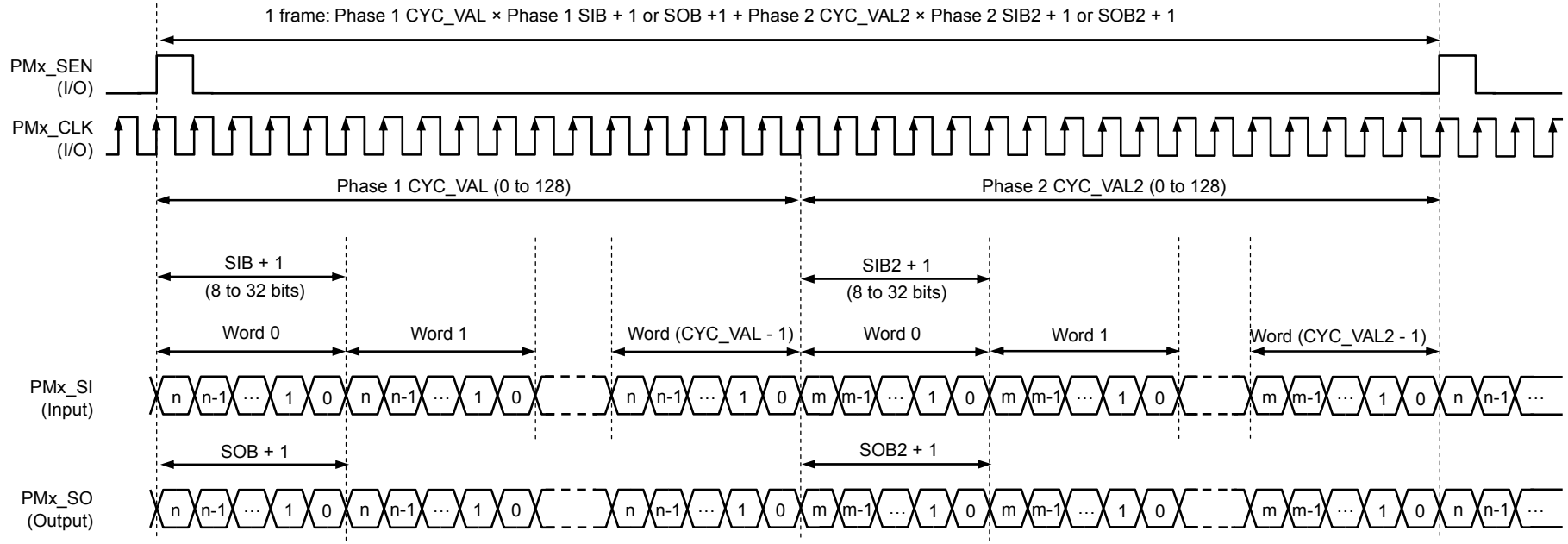
When Word_count (number of words) is set to 0 for phase 1 or 2, a single-phase operation is performed.

It is prohibited to set Word_count for both phase 1 and phase 2 to 0.

In simultaneous transmit/receive operation, set the same number of bits/words for transmission/reception in each phase (SIB = SOB, SIB2 = SOB2).

Word_size for phase 1 may differ from that for phase 2, and Word_count for phase 1 may differ from that for phase 2 (SIB ≠ SIB2, SOB ≠ SOB2, and CYC_VAL ≠ CYC_VAL2 may be set).

Figure 4-6. Serial Interface Timing When Mode 5 Is Selected



(7) Mode 6 serial interface timing

Figure 4-7 shows the serial interface timing applicable when mode 6 is selected.

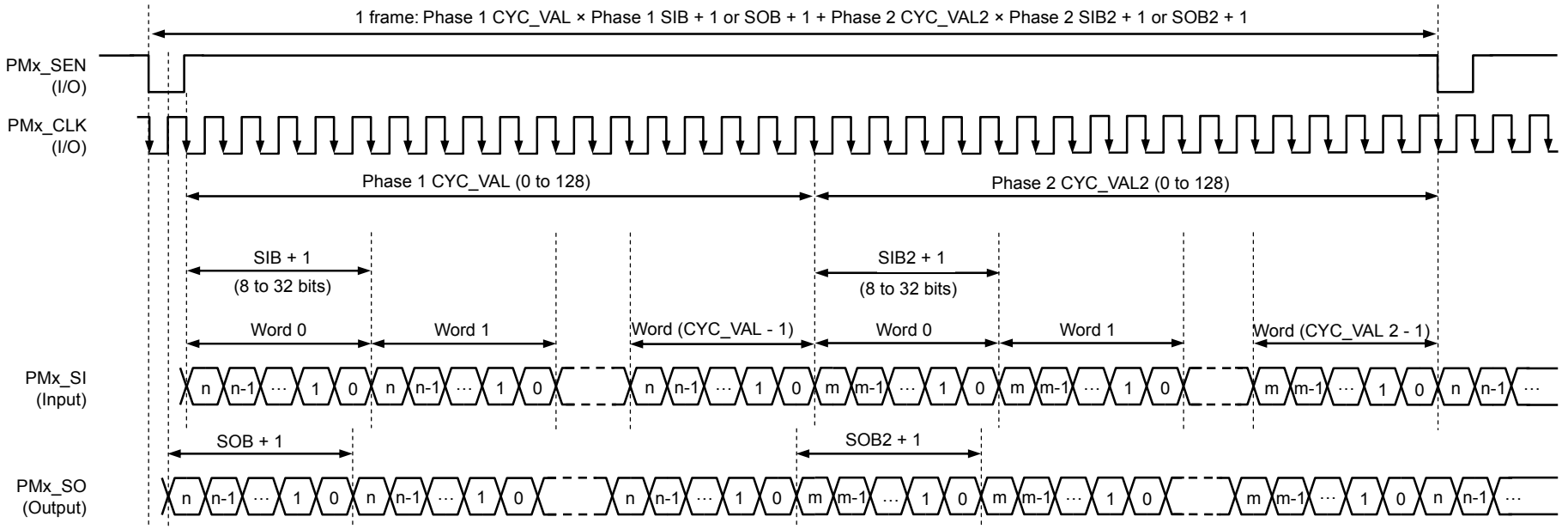
When Word_count (number of words) is set to 0 for phase 1 or 2, a single-phase operation is performed.

It is prohibited to set Word_count for both phase 1 and phase 2 to 0.

In simultaneous transmit/receive operation, set the same number of bits/words for transmission/reception in each phase (SIB = SOB, SIB2 = SOB2).

Word_size for phase 1 may differ from that for phase 2, and Word_count for phase 1 may differ from that for phase 2 (SIB ≠ SIB2, SOB ≠ SOB2, and CYC_VAL ≠ CYC_VAL2 may be set).

Figure 4-7. Serial Interface Timing When Mode 6 Is Selected



4.1.3 Data padding

Data padding can be performed when the data bit length is 8 or 16 bits. Data padding refers to an operation by which four 8-bit units or two 16-bit units are padded into one word (32 bits) since the FIFO bit width is 32 bits.

Data padding can be enabled or disabled by using the PMx_CYCLE register.

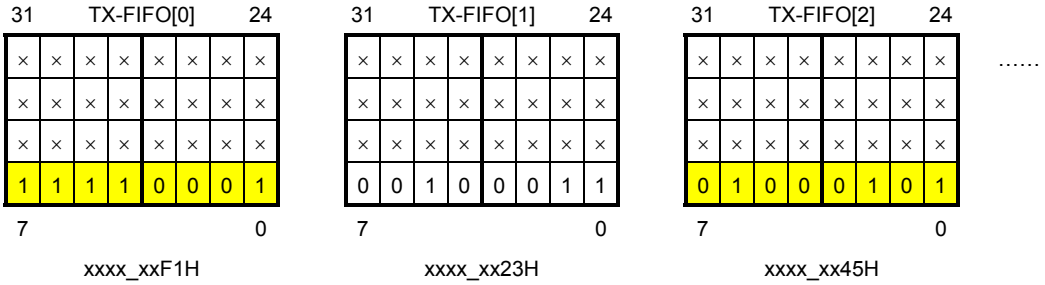
Data padding is prohibited in mode 5 or 6.

(1) Data padding on transmit side

(a) Without data padding

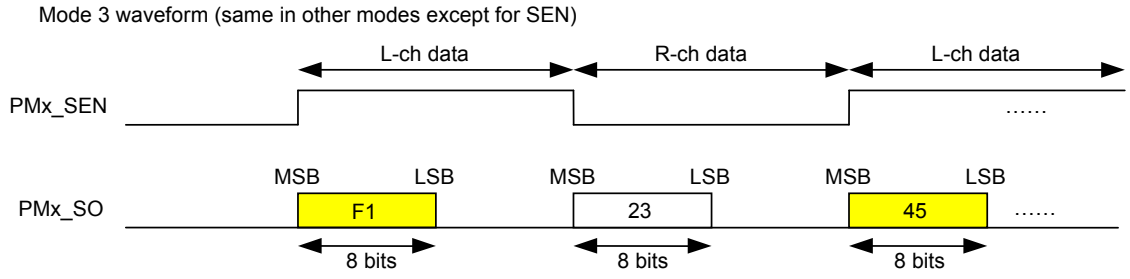
Example 1 Transmit data bit length = 8 bits (SOB = 07H)
Data padding on the transmit side is disabled
xxxx_xxF1H → Write data (PMx_TXQ)
xxxx_xx23H → Write data (PMx_TXQ)
xxxx_xx45H → Write data (PMx_TXQ)
xxxx_xx67H → Write data (PMx_TXQ)
:

Bits 20 to 16 of PMx_CYCLE
Bit 23 of PMx_CYCLE = 0



Data longer than the transmit data bit length is ignored (×: Don't care).

Figure 4-8. Transmit Waveform in Mode 3



(b) With data padding

Example 2 Transmit data bit length = 8 bits (SOB = 07H) — Bits 20 to 16 of PMx_CYCLE
Data padding on the transmit side is enabled — Bit 23 of PMx_CYCLE = 1
F123_4567H → Write data (PMx_TXQ)
89AB_CDEFH → Write data (PMx_TXQ)
2468_ACE0H → Write data (PMx_TXQ)

31 TX-FIFO[0] 24							
1	1	1	1	0	0	0	1
0	0	1	0	0	0	1	1
0	1	0	0	0	1	0	1
0	1	1	0	0	1	1	1
7	F123_4567H						0

31 TX-FIFO[1] 24							
1	0	0	0	1	0	0	1
1	0	1	0	1	0	1	1
1	1	0	0	1	1	0	1
1	1	1	0	1	1	1	1
7	89AB_CDEFH						0

31 TX-FIFO[2] 24							
0	0	1	0	0	1	0	0
0	1	1	0	1	0	0	0
1	0	1	0	1	1	0	0
1	1	1	0	0	0	0	0
7	2468_ACE0H						0

……

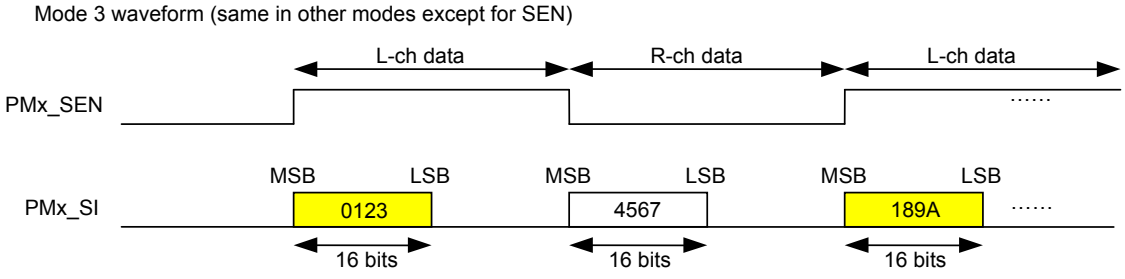
Remark The signal waveform is the same as in **Figure 4-8 Transmit Waveform in Mode 3**.

(2) Data padding at receive side

(a) Without data padding

Example 3 Receive data bit length = 16 bits (SIB = 0FH) — Bits 12 to 8 of PMx_CYCLE
Data padding on the receive side is disabled — Bit 15 of PMx_CYCLE = 0

Figure 4-9. Receive Waveform in Mode 3



31 RX-FIFO[0] 24							
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1
0	0	1	0	0	0	1	1
7	0000_0123H						0

31 RX-FIFO[1] 24							
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	1	0	0	0	1	0	1
0	1	1	0	0	1	1	1
7	0000_4567H						0

31 RX-FIFO[2] 24							
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	1	1	0	0	0
1	0	0	1	1	0	1	0
7	0000_189AH						0

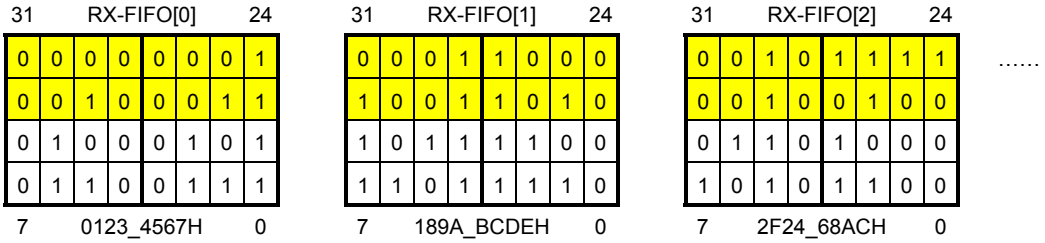
……

Data exceeding the receive data bit length is padded with zeros.

Read data (PMx_RXQ) ← 0000_0123H (RXFIFO[0])
Read data (PMx_RXQ) ← 0000_4567H (RXFIFO[1])
Read data (PMx_RXQ) ← 0000_189AH (RXFIFO[2])
:

(b) With data padding

Example 4 Receive data bit length = 16 bits (SIB = 0FH) Bits 12 to 8 of PMx_CYCLE
Data padding on the receive side is enabled Bit 15 of PMx_CYCLE = 1



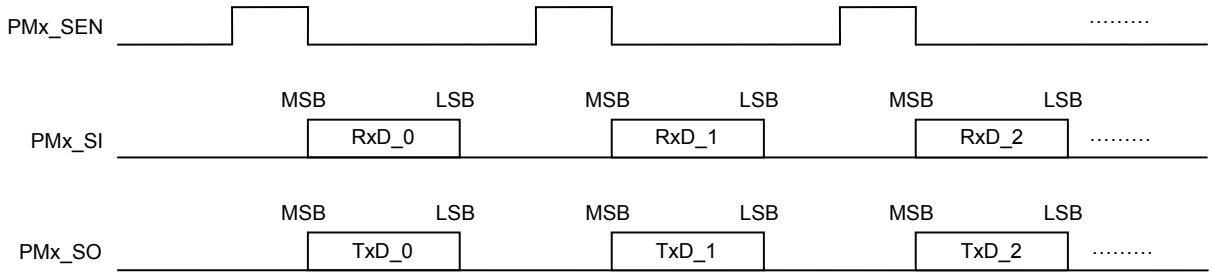
Read data (PMx_RXQ) ← 0123_4567H (RXFIFO[0])
Read data (PMx_RXQ) ← 189A_BCDEH (RXFIFO[1])
Read data (PMx_RXQ) ← 2F24_68ACH (RXFIFO[2])
:

Caution If 32 bits of the data received from the PMx_SI pin does not amount to 32 bit in RXFIFO, the data is not considered valid and the last data cannot be read.

Remark The signal waveform is the same as in **Figure 4-9 Receive Waveform in Mode 3**.

4.1.4 FIFO operations

(1) Modes 0 and 1

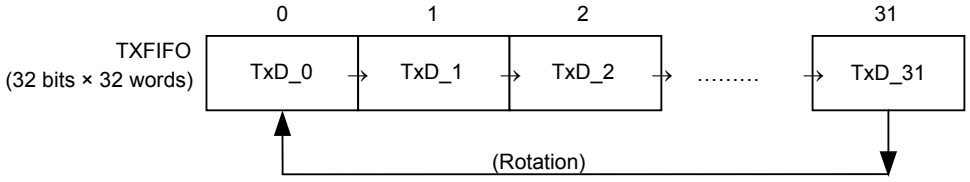


(a) TXFIFO

<1> Data is stored in TXFIFO in the order that the data is written to the transmit data write register (PMx_TXQ) from the host.

<2> One word of data is taken out of TXFIFO.
Based on the value *n* set to the SOB bits of the data transfer cycle setting register (PMx_CYCLE), serial data is output from the PMx_SO pin in order from bit *n* (MSB) to bit 0 (LSB).

<3> After the data is output up to bit 0 (LSB), TXFIFO is incremented by 1. The same processing as <2> is repeated when the next frame is transmitted. The count range is 0 to 31 (1FH) and the counter value returns to 0 when it has counted up to 31.

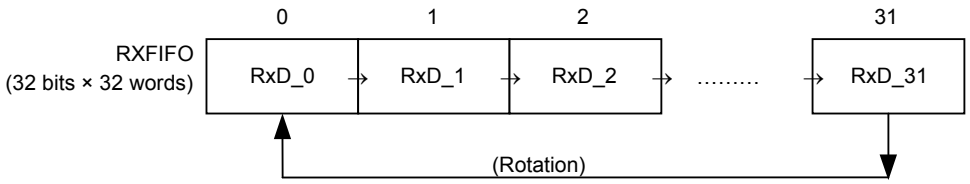


(b) RXFIFO

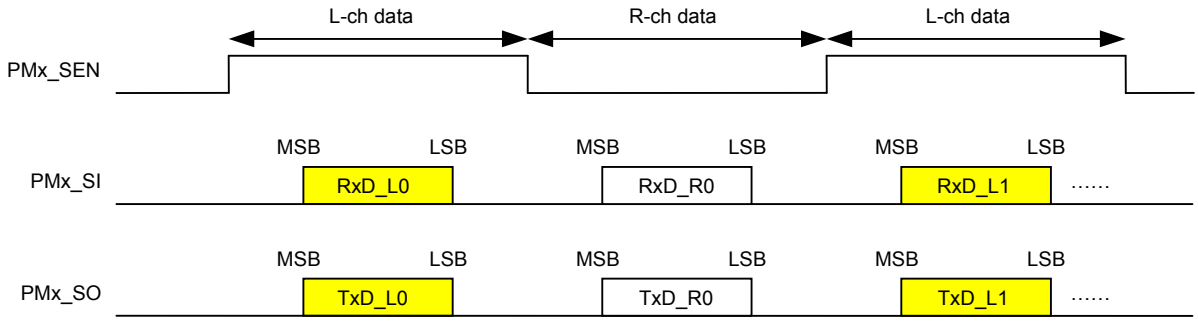
<1> RXFIFO receives serial data from the PMx_SI pin.

<2> Based on the value *m* set to the SIB bits in the data transfer cycle setting register (PMx_CYCLE), the receive bits are stored in RXFIFO in word units in order from bit *m* (MSB) to bit 0 (LSB). Bits higher than bit *m* are padded with zeros.

<3> After the data is stored up to bit 0 (LSB), RXFIFO is incremented by 1. The same processing as <2> is repeated when RXFIFO receives the next frame. The count range is 0 to 31 (1FH) and the counter value returns to 0 when it has counted up to 31.

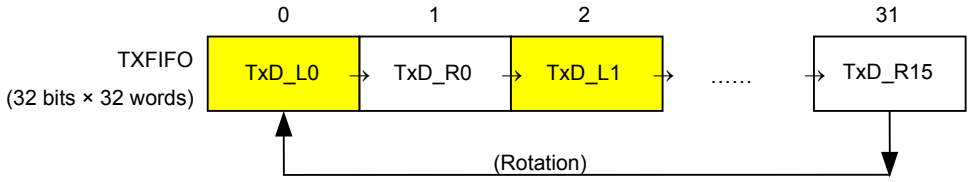


(2) Modes 2 to 4

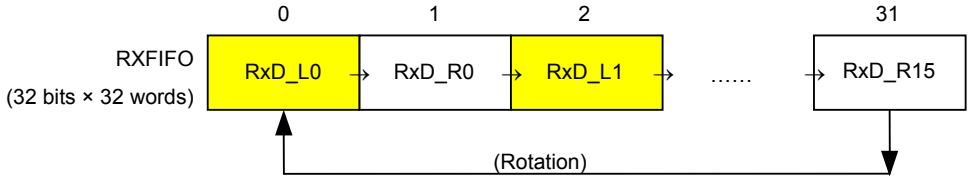


In modes 2 to 4, L-ch data and R-ch data are alternately stored in RXFIFO and TXFIFO, because the L-ch data and the R-ch data are alternately transmitted and received.
Therefore, the L-ch data and the R-ch data need to be alternately written for transmission. For reception, the L-ch data is read first, then the R-ch data.
For any other operation, the FIFO operation is the same as that in modes 0 and 1.

(a) TXFIFO

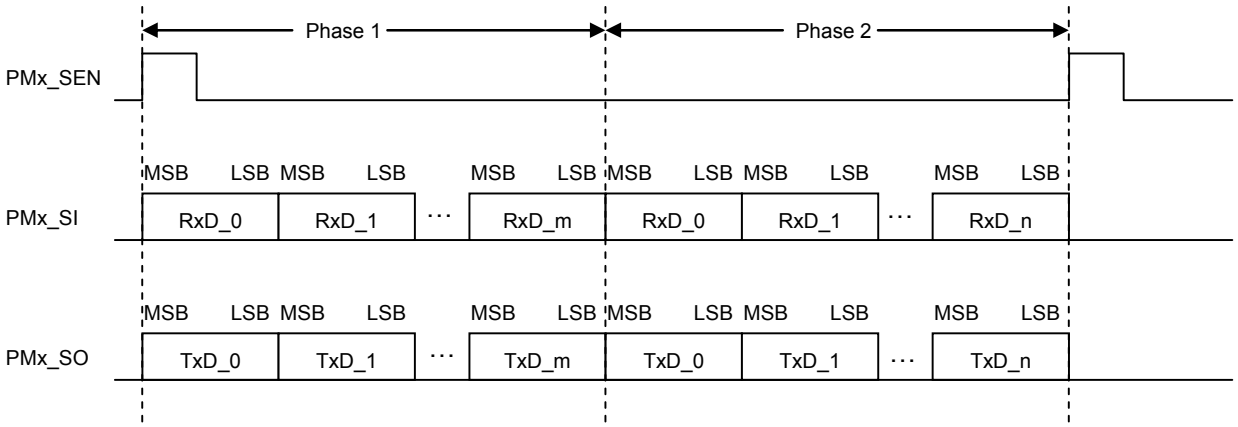


(b) RXFIFO



Remark For storing padded data in FIFO, see **4.1.3 Data padding**.

(3) Modes 5 and 6

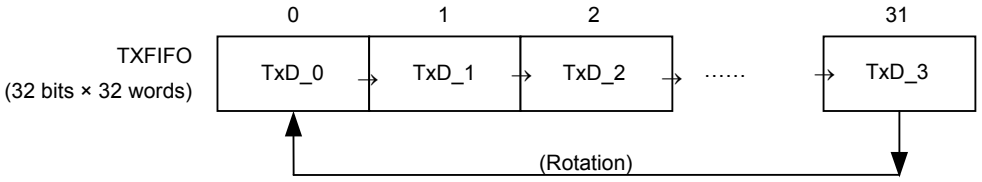


(a) TXFIFO

<1> Data is stored in TXFIFO in the order that the data is written to the transmit data write register (PMx_TXQ) from the host.

<2> Data corresponding to one word is taken out of TXFIFO.
Based on the values *n* and *m* set to the SOB and SOB2 bits in the data transfer cycle setting registers (PMx_CYCLE, PMx_CYCLE2), serial data is output from the PMx_SO pin as the phase 1 transmit data corresponding to the words set to the CYC_VAL bits in the PMx_CYCLE register in order from bit m (MSB) to bit 0 (LSB), and subsequently the phase 2 transmit data corresponding to the words set to the CYC_VAL2 bits in the PMx_CYCLE2 register in the order from bit *n* (MSB) to bit 0 (LSB).

<3> When one word of data is output up to bit 0 (LSB), TXFIFO is incremented by 1. Then, the data corresponding to the next word is output up to bit 0 (LSB) and TXFIFO is incremented again. The same processing is then repeated. The TXFIFO count range is 0 to 31 (1FH) and the counter value returns to 0 when it has counted up to 31.

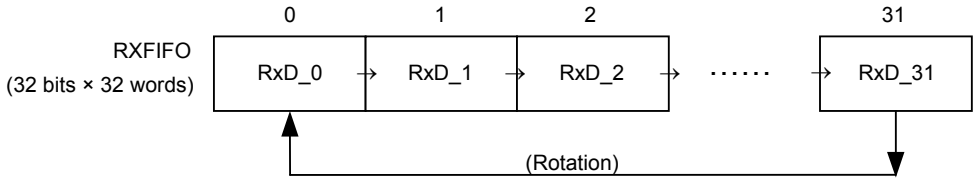


(b) RXFIFO

<1> RXFIFO receives serial data from the PMx_SI pin.

<2> Based on the values *m* and *n* set in the SIB and SOB2 bits in the data transfer cycle setting registers (PMx_CYCLE, PMx_CYCLE2), the phase 1 receive data corresponding to the words set to the CYC_VAL bits in the PMx_CYCLE register is stored in RXFIFO in word units in order from bit *m* (MSB) to bit 0 (LSB), and subsequently the phase 2 receive data corresponding to the words set to the CYC_VAL2 bits in the PMx_CYCLE2 register is stored in RXFIFO in word units in order from bit *n* (MSB) to bit 0 (LSB). Bits higher than *m* and *n* are padded with zeros if the size of the receive data is smaller than 32 bits.

<3> When data corresponding to one word is stored up to bit 0 (LSB), RXFIFO is incremented by 1. Then the data corresponding to the next word is received up to bit 0 (LSB) and RXFIFO is incremented again. The same processing is then repeated. The RXFIFO count range is 0 to 31 (1FH) and the counter returns to 0 when it has counted up to 31.



4.1.5 Data transfer status (status/error/interrupt source)

Table 4-2 lists the data transfer statuses. The data transfer status can be checked by reading the interrupt raw status register (PMx_RAW). If an error is detected, the transmit or receive operation in which the error occurred stops. If operation is performed in master mode, however, output of the PMx_CLK and PMx_SEN signals continues.

Table 4-2. Data Transfer Statuses (Status/Error/Interrupt Source)

(1/2)

Status	Status/Error Details and Operation
RX_STP	Indicates whether reception can be re-enabled after reception stoppage is requested. Because this flag is set after the transfer being executed is stopped, the time taken until this bit is set varies depending on conditions such as the timing at which reception stoppage is requested, and whether the device is a master or a slave. 1: When reception can be re-enabled after reception is requested. 0: When reception has not started or reception is in progress.
TX_STP	Indicates whether transmission can be re-enabled after transmission stoppage is requested. Because this flag is set after the transfer being executed is stopped, the time taken until this bit is set varies depending on conditions such as the timing at which transmission stoppage is requested, and whether the device is a master or a slave. 1: When transmission can be re-enabled after transmission is requested. 0: When transmission has not started or transmission is in progress.
RX_REN	Indicates that receive data can be read. This flag and the PMx_DMA_RDMARQ (DMA receive request) signal are set to on or off simultaneously. 1: When there is valid data of at least one word in the receive FIFO (there is data that has not been read by the DMA controller). 0: When all data in the receive FIFO has been read or reception is disabled (bit 1 of PMx_TXRX_DIS = 1).
RX_ORE	Indicates that an overrun error occurred in the receive FIFO. 1: When the receive FIFO is not read by the DMA controller in time (serial data is received from the PMx_SI pin and is written to the receive FIFO while the receive FIFO is full). 0: When reception is disabled (bit 1 of PMx_TXRX_DIS = 1).
RX_URE	Indicates that an underrun error occurred in the receive FIFO. 1: When the receive FIFO is read when there is no valid data in the receive FIFO (RE_REN = 0) (non-existent receive data is read). 0: When reception is disabled (bit 1 of PMx_TXRX_DIS = 1).
RX_FRE	Indicates that a serial interface synchronization error[Note 1] occurred in the receive controller. 1: When deviation of the PMx_SEN signal is detected (only in the case of slave operation). 0: When reception is disabled (bit 1 of PMx_TXRX_DIS = 1) after the interrupt source is cleared (bit 4 of PMx_CLEAR = 1).
TX_WEN	Indicates that writing transmit data is enabled.[Note 2] This flag and the PMx_DMA_TDMARQ (DMA transmit request) signal are set to on or off simultaneously. 1: When the transmit FIFO is not full (the transmit FIFO has free space and data can be written). 0: When the transmit FIFO is full and data can no longer be written, or transmission is disabled (bit 0 of PMx_TXRX_DIS = 1).

(2/2)

Status	Status/Error Details and Operation
TX_ORE	Indicates that an overrun error occurred in the transmit FIFO. 1: When data is written to the transmit FIFO when the transmit FIFO is full (TX_WEN = 0). 0: When transmission is disabled (bit 0 of PMx_TXRX_DIS = 1).
TX_URE	Indicates that an underrun error occurred in the transmit FIFO. 1: When all transmit data has been output from the PMx_SO pin and the transmit FIFO is empty (all-data transmit completion has been reported, or the serial data transmit speed is faster than writing to the transmit FIFO). 0: When transmission is disabled (bit 0 of PMx_TXRX_DIS = 1) after the interrupt source is cleared (bit 1 of PMx_CLEAR = 1).
TX_FRE	Indicates that a serial interface synchronization error[Note 1] occurred in the transmit controller. 1: When deviation of the PMx_SEN signal is detected (only in the case of slave operation). 0: When transmission is disabled (bit 0 of PMx_TXRX_DIS = 1) after the source is cleared (bit 0 of PMx_CLEAR = 1).

Notes 1. A synchronization error occurs when the PMx_SEN signal does not change for the “number of clock cycles per frame” specified by CYC_VAL (modes 0 to 4), or CYC_VAL, SIB, SOB, CYC_VAL2, SIB2, and SOB2 (modes 5 and 6) in the data transfer cycle setting registers (PMx_CYCLE, PMx_CYCLE2) during communication (the change depends on the format of modes 0 to 6). A synchronization error also occurs if the transmit module and the receive module fall out of synchronization

2. This bit is set to 0 during reset, and is set to 1 when 1 PCLK cycle has been input after reset release.

Remark RX_FRE and TX_FRE are set to on respectively when a synchronization error is detected in the receiver or transmitter. If both reception and transmission are enabled, the flag for which the error occurred first is set (actually these errors occur almost at the same time).

4.1.6 Error recovery

(1) Recovery from receive errors

When a receive error (RX_ORE, RX_URE, or RX_FRE) is detected, the receive operation is stopped. At this time, be sure to disable receive operations, wait for error recovery, and then restart receive operations.
When a receive error is detected, perform recovery processing according to the procedure below.

(a) Set the reception enable bit (RX_ENCLR) of the PMx_TXRX_DIS register to 1 to disable reception.
(b) Wait for all receive errors (RX_ORE, RX_URE, and RX_FRE) to be cleared to 0 and confirm error recovery.
(c) Set the reception enable bit (RX_EN) of the PMx_TXRX_EN register to 1 to restart receive operations.

(2) Recovery from transmit errors

When a transmit error (TX_ORE, TX_URE, or TX_FRE) is detected, the transmit operation is stopped. At this time, be sure to disable transmit operations, wait for error recovery, and then restart transmit operations.
When a transmit error is detected, perform recovery processing according to the procedure below.

(a) Set the transmission enable bit (TX_ENCLR) of the PMx_TXRX_DIS register to 1 to disable transmission.
(b) Wait for all transmit errors (TX_ORE, TX_URE, and TX_FRE) to be cleared to 0 and confirm error recovery.
(c) Set the transmission enable bit (TX_EN) of the PMx_TXRX_EN register to 1 to restart transmit operations.

4.2 PWM Interface

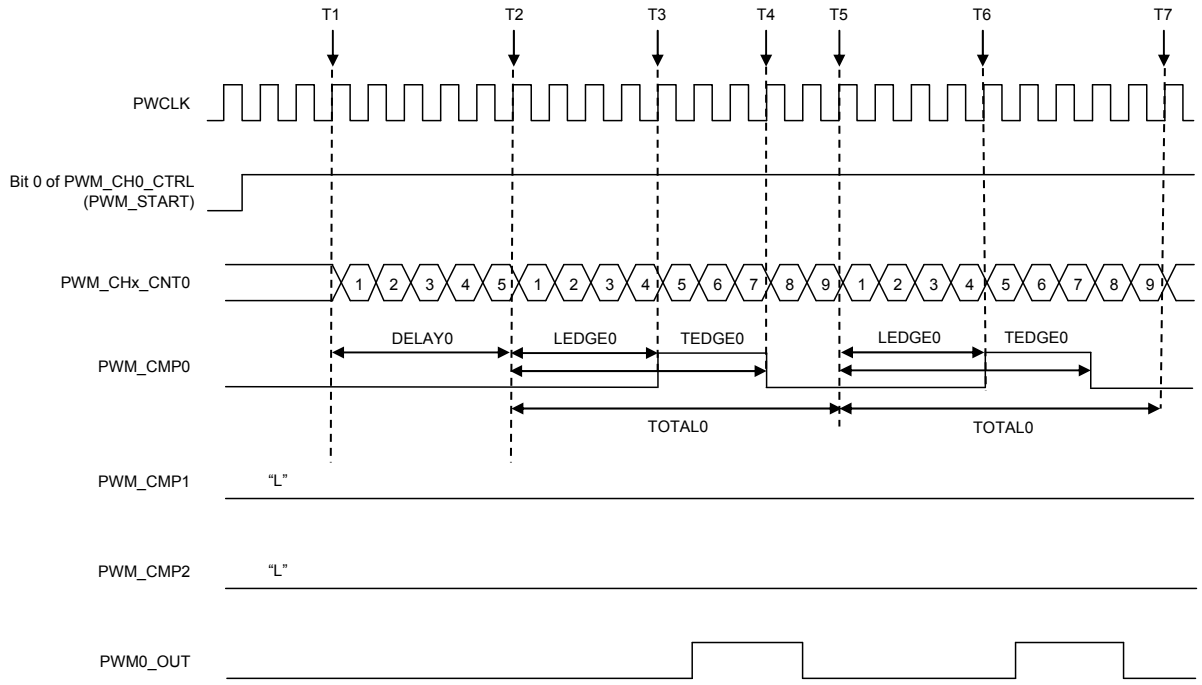
4.2.1 PWM operation

(1) When one counter is started

The following shows an example of when only counter 0 of channel 0 (counter 0; PWM_CH0_CNT0) is started.

- PWM_CH0_TOTAL0 = 0000_0009H: Total cycle: PWM_PWCLK × 9 cycles
- PWM_CH0_DELAY0 = 0000_0005H: Delay value: PWM_PWCLK × 5 cycles
- PWM_CH0_LEDGE0 = 0000_0004H: Leading edge setting: 4 cycles
- PWM_CH0_TEDGE0 = 0000_0007H: Trailing edge setting: 7 cycles
- PWM_CH0_MODE = 0000_0001H: Automatic stop disabled (repeat mode set) and inversion of PWM_CMP0 not specified

Figure 4-10. Timing When Only Counter 0 Runs



<1> Counter 0 starts at the rising edge of PWCLK three cycles after the CMP_EN0 bit of the PWM_CH0_MODE register is set to 1 and the PWM_START bit of the PWM_CTRL register is set to 1.

<2> If a delay value is set in the PWM_CH0_DELAY0 register, counter 0 continues counting up to that value. The output from PWM_CMP0 remains low level.

<3> When the counter value matches the value of PWM_CH0_DELAY0, counter 0 is set to 1 in the next cycle (T2 timing).

<4> Counter 0 starts counting at T2 again. When the counter value matches the PWM_CH0_LEDGE0 value, PWM_CMP0 outputs a high level at the next cycle.

<5> In the next cycle after the value of counter 0 matches TEDGE0 (T4 timing), the PWM_CMP0 output becomes low level. PWM_CMP0 continues outputting a low level until the values of TOTAL0 and counter 0 match.

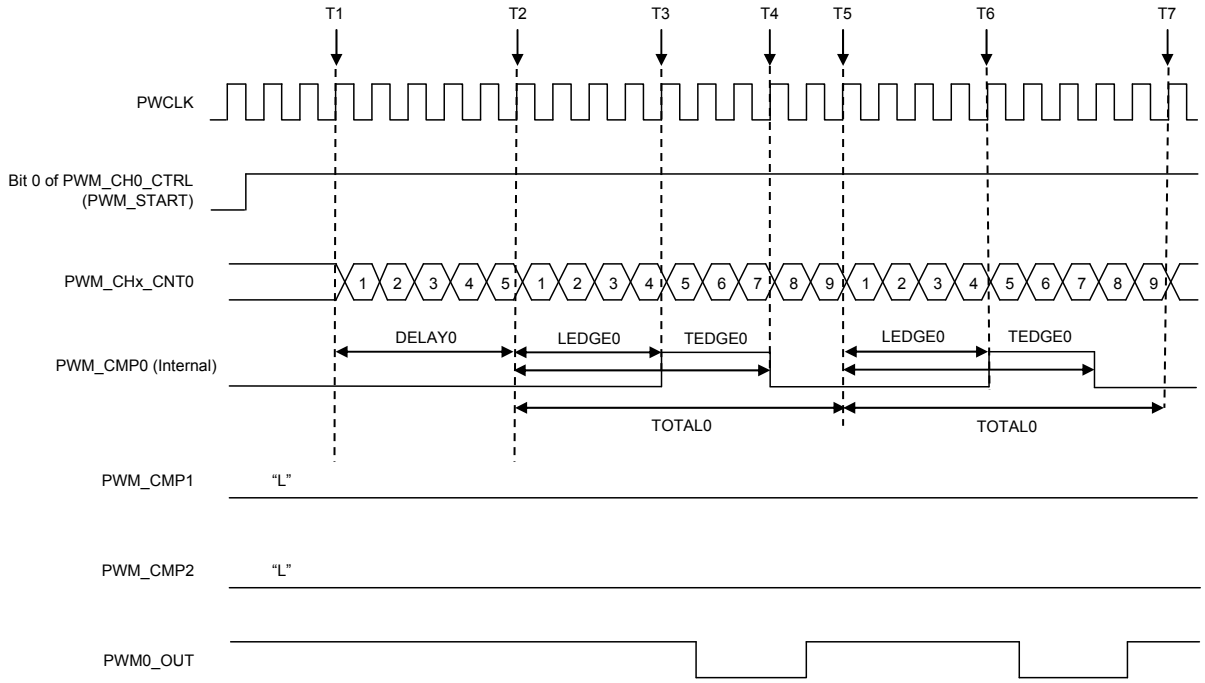
<6> If the values of counter 0 and TOTAL0 match, counter 0 is set to 1 in the next cycle (T5 timing).

<7> After the T5 timing, the operation of T2 to T5 is repeated.

(2) When one counter is started and output is inverted at PWM_CNT0

- PWM_CH0_TOTAL0 = 0000_0009H: Total cycle: PWM_PWCLK × 9 cycles
- PWM_CH0_DELAY0 = 0000_0005H: Delay value: PWM_PWCLK × 5 cycles
- PWM_CH0_LEDGE0 = 0000_0004H: Leading edge setting: 4 cycles
- PWM_CH0_TEDGE0 = 0000_0007H: Trailing edge setting: 7 cycles
- PWM_CH0_MODE = 0000_0005H: Automatic stop disabled (repeat mode set) and inversion of PWM_CMP0 specified

Figure 4-11. Only Counter 0 Runs and PWM_CMP0 Value Is Inverted



The waveform generated in CMP0 (CMP0 internal waveform) is inverted when PWM_CMP0 is output if inversion is specified.

(3) When looping is enabled

The following describes the operation when looping is specified for counter 0 of channel 0.

If looping is specified, counter 0 automatically stops after loop repeats the specified number of times.

This example applies when counter 0 of channel 0 is set as follows.

- PWM_CH0_TOTAL0 = 0000_0009H
- PWM_CH0_DELAY0 = 0000_0006H
- PWM_CH0_LEDGE0 = 0000_0004H
- PWM_CH0_TEDGE0 = 0000_0007H
- PWM_CH0_MODE = 0000_0003H: Automatic stop enabled and counter 0 enabled

Figure 4-12. Start Operation When Looping Is Enabled

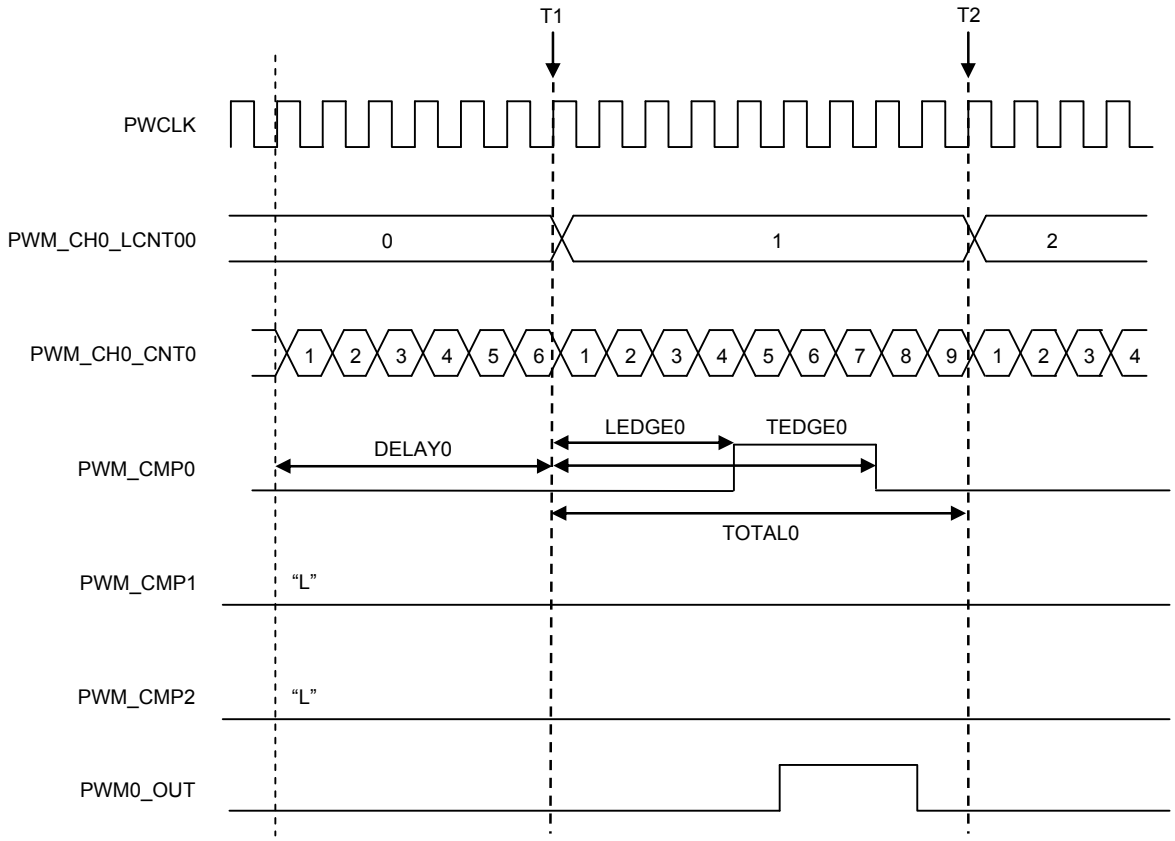
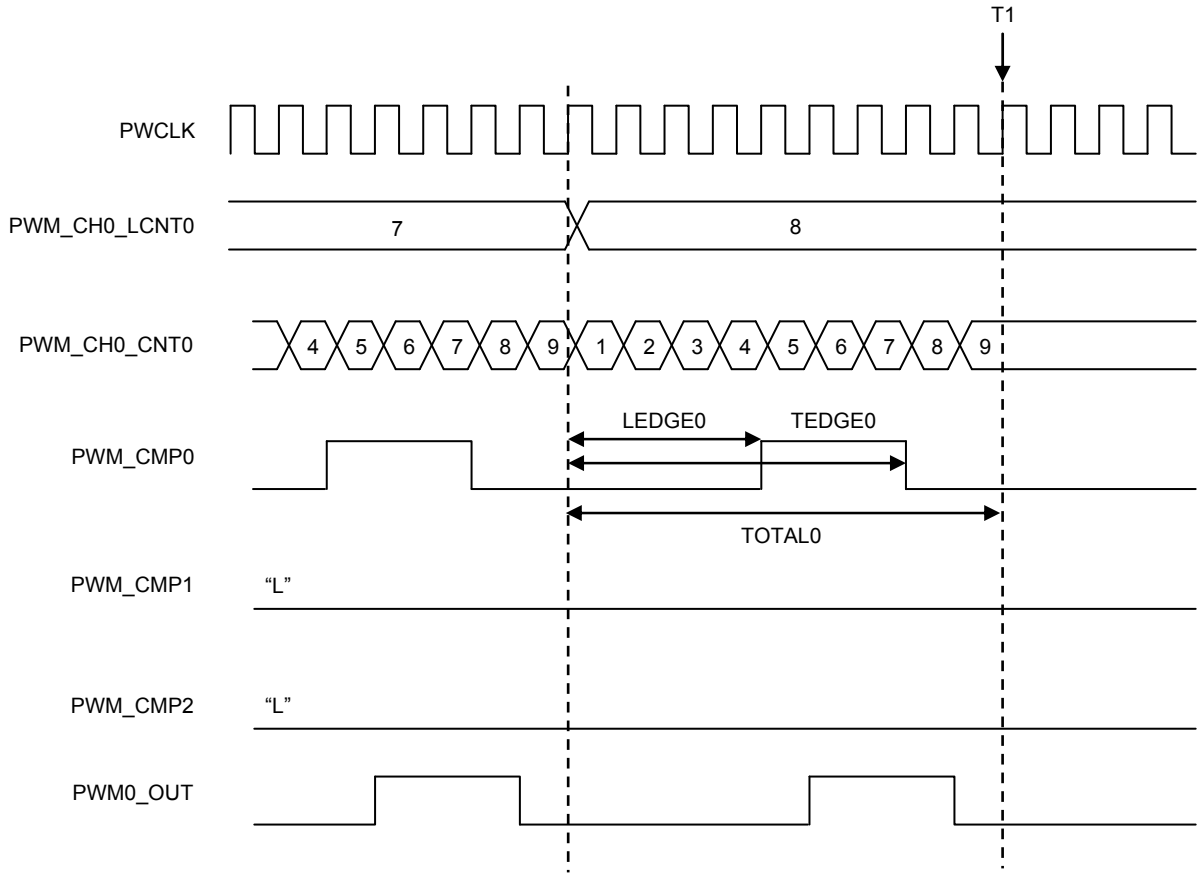


Figure 4-13. Termination Operation When Looping Is Enabled



PWM_CH0_CNT0 and PWM_CH0_LCNT0 stop at the next cycle after loop counter 0 matches the specified value. The PWM0_OUT output remains low level.

(4) When three counters are started (AND specified)

The following shows an example when counters 0 to 2 are set and started as follows.

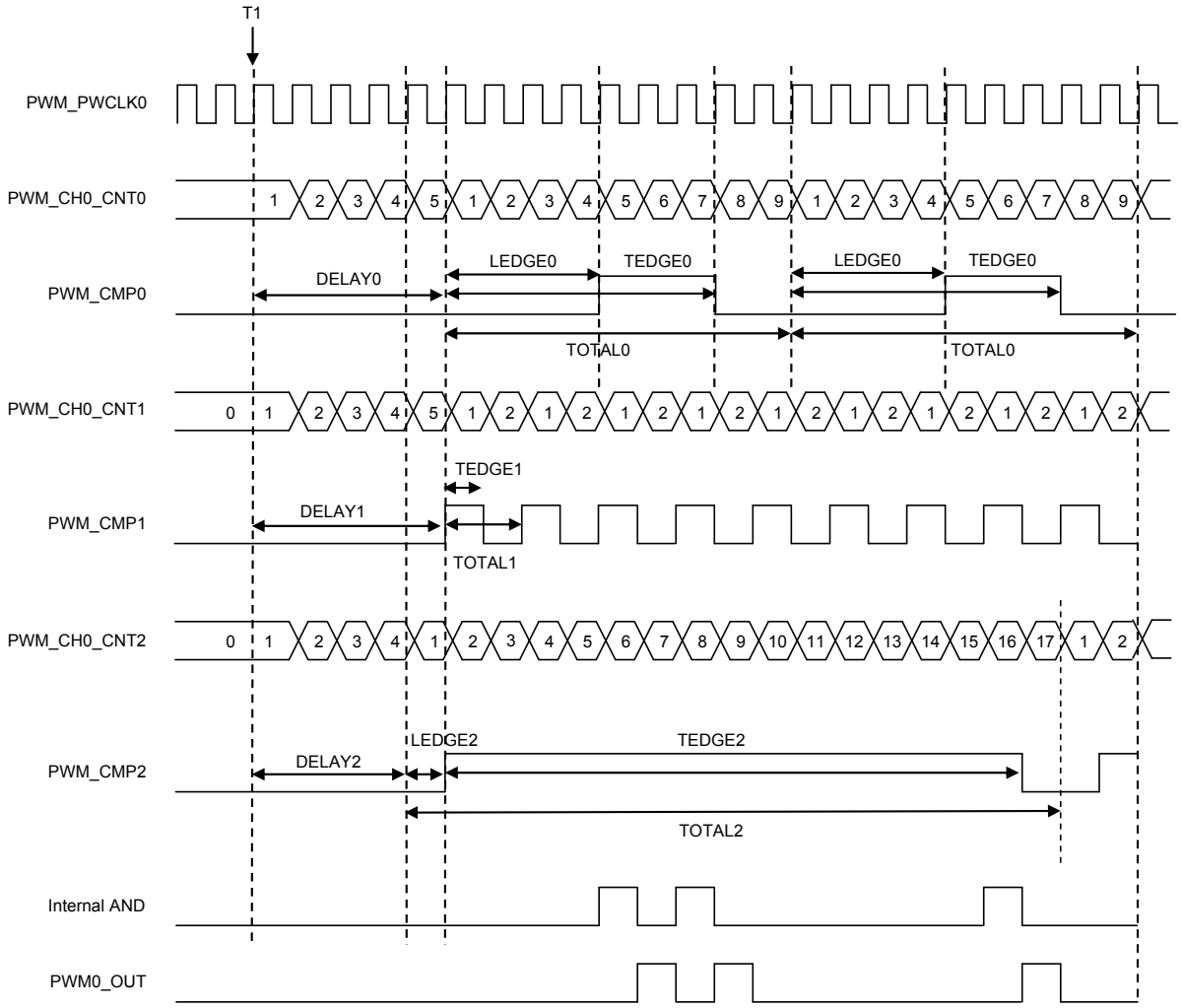
- PWM_CH0_TOTAL0 = 0000_0009H: Counter 0 total cycle: PWM_PWCLK × 9 cycles
- PWM_CH0_DELAY0 = 0000_0005H: Counter 0 delay value: PWM_PWCLK × 5 cycles
- PWM_CH0_LEDGE0 = 0000_0004H: Counter 0 leading edge setting: 4 cycles
- PWM_CH0_TEDGE0 = 0000_0003H: Counter 0 trailing edge setting: 3 cycles

- PWM_CH0_TOTAL1 = 0000_0002H: Counter 1 total cycle: PWM_PWCLK × 2 cycles
- PWM_CH0_DELAY1 = 0000_0005H: Counter 1 delay value: PWM_PWCLK × 5 cycles
- PWM_CH0_LEDGE1 = 0000_0000H: Counter 1 leading edge setting: 0 cycles
- PWM_CH0_TEDGE1 = 0000_0001H: Counter 1 trailing edge setting: 1 cycle

- PWM_CH0_TOTAL2 = 0000_0011H: Counter 2 total cycle: PWM_PWCLK × 17 cycles
- PWM_CH0_DELAY2 = 0000_0004H: Counter 2 delay value: PWM_PWCLK × 4 cycles
- PWM_CH0_LEDGE2 = 0000_0001H: Counter 2 leading edge setting: 1 cycle
- PWM_CH0_TEDGE2 = 0000_0010H: Counter 2 trailing edge setting: 16 cycles

- PWM_CH0_MODE = 0001_0111H: The operation of CMP_EN0 to CMP_EN2 is enabled.
  The output of CMP0 to CMP2 is not inverted.
  Automatic stop is not enabled.
  ANDing of each counter output is specified by the MOD bit.

Figure 4-14. Operation of Counters 0 to 2: With AND Specified



The values generated by PWM_CMP0 to PWM_CMP2 are internally ANDed and output from PWM0_OUT.

4.2.2 Interrupt timing

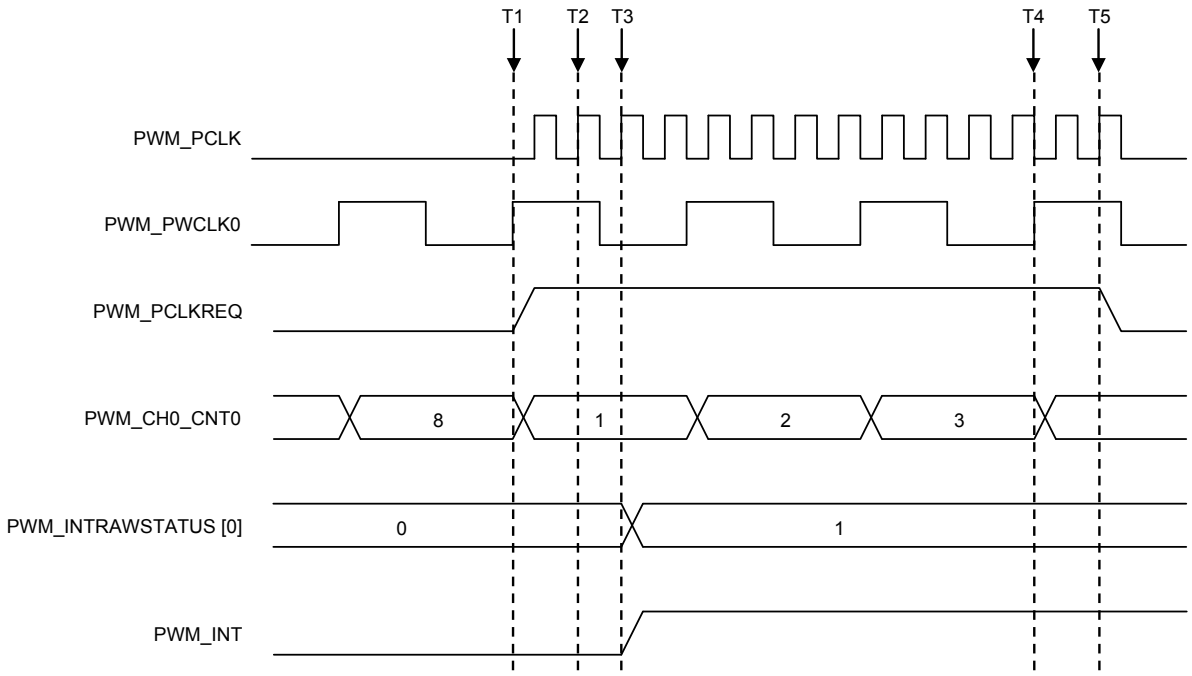
There are interrupts for each counter in the PWM interface and an interrupt source is generated when the counter reaches the total cycle value. An interrupt source is also generated when the loop count terminates.

When an enabled interrupt source is generated, the source is set to PWM_INTSTATUS and the ORed interrupt signal is output. The interrupt source is cleared by using the PWM_INTFFCLR register.

An example is shown below.

- Only CMP0 of channel 0 operates
- Interrupt output of CMP0 of channel 0 is enabled by setting INT_ENSET to 0000_0001H
- PWM_CH0_TOTAL0 = 8

Figure 4-15. Interrupt Timing (Clock Control)



If the value of counter 0 matches the specified value (T1 in Figure 4-15), the internal signal becomes active in the second PWM_PCLK cycle (T2 in Figure 4-15) and the interrupt source is reflected in the PWM_INTRAWSTATUS register in the next PWM_PCLK cycle (T3 in Figure 4-15). If an interrupt source is enabled, the interrupt output signal PWM_INT also becomes active.

The internal signal becomes inactive in the third PWM_PWCLK cycle or later after T3 (T4 in Figure 4-15) and in the second PWM_PCLK cycle after that (T5 in Figure 4-15).

4.2.3 Clock control timing

As shown in Figure 4-15, PCLKREQ must be asserted while the internal signal for setting an interrupt source is active so that PWM_PCLK will be supplied.

4.3 Clock/Reset

The clocks used for the audio/voice interface are shown below.

(1) PMx_SCLK (serial internal clock)

Used for the AHB bus clock and internal operation of the audio/voice interface.

(2) PM0_PCLK (APB clock)

Used for APB bus access (register access).
Common to audio/voice serial 0 and 1.

The clocks used for the PWM interface are shown below.

(1) PWM_PCLK

Used for APB bus access.

(2) PWM_PWCLK0, PWM_PWCLK1

Used for internal operation of the PWM interface.

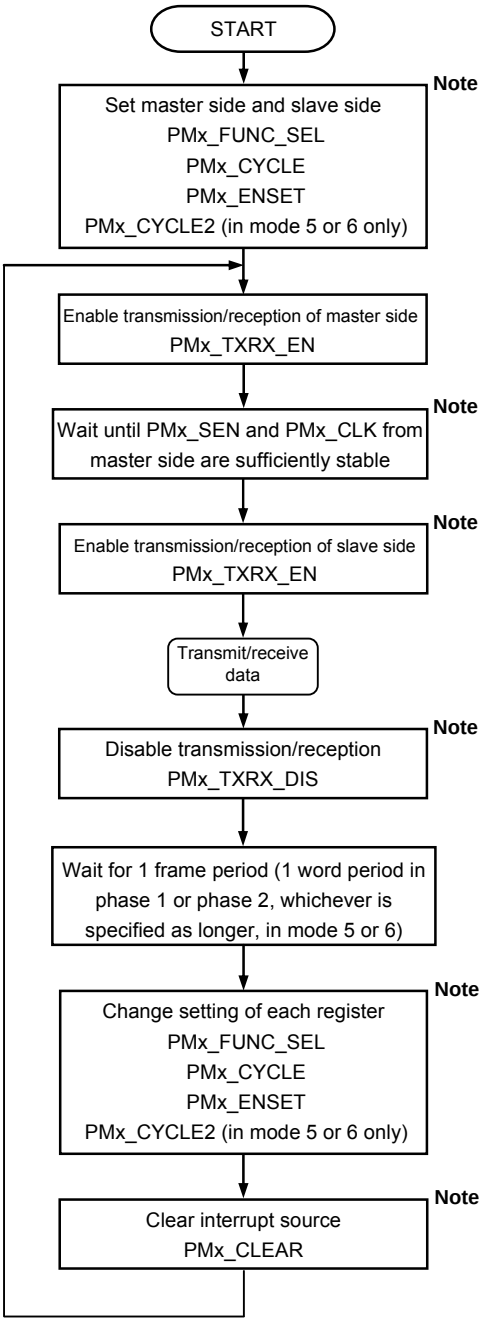
For details about clock/reset settings, see the **Multimedia Processor for Mobile Applications - System Control/General-Purpose I/O Interface User's Manual (S19265E)**.

CHAPTER 5 USAGE

5.1 Procedure for Setting Audio/Voice Interface Registers

The following figure shows the register setting procedure.

Figure 5-1. Register Setting Flow



Note In single-unit operation, no slave side setting is required.

5.1.1 Example of setting communication start

Base address PM0: C001_0000H

PM1: 400D_0000H

(1) PMx_FUNC_SEL (xxxx_0000H): Operation mode setting

- MODE_SEL[2:0] → 011: Mode 3
- M_S[1:0] → 10: Slave mode
- Tx_TIM[1:0] → 00: Transmission starts when 30 words are stored in the transmit FIFO.

(2) PMx_CYCLE (xxxx_000CH): Setting of the number of clocks and transmit/receive data bit length

- CYC_VAL[5:0] → 01_1111: Number of clocks per frame is 32
- SIB[4:0] → 1_0111: PMx_SI data bit length is 24 bits
- RX_PD → 0: No data padding on receive side
- SOB[4:0] → 1_0111: PMx_SO data bit length is 24 bits
- TX_PD → 0: No data padding on transmit side

(3) PMx_ENSET (xxxx_0018H): Interrupt mask cancellation

- RX_STP → 1: Source of reception re-enable (interrupt mask cancellation)
- TX_STP → 1: Source of transmission re-enable (interrupt mask cancellation)
- RM_REN → 0: Source of receive data read enable (interrupt mask setting)
- RM_ORE → 1: Source of receive overrun error (interrupt mask cancellation)
- RM_URE → 1: Source of receive underrun error (interrupt mask cancellation)
- RM_FRE → 1: Source of receive synchronization error (interrupt mask cancellation)
- TM_WEN → 0: Source of transmit data write enable (interrupt mask setting)
- TM_ORE → 1: Source of transmit overrun error (interrupt mask cancellation)
- TM_URE → 1: Source of transmit underrun error (interrupt mask cancellation)
- TM_FRE → 1: Source of transmit synchronization error (interrupt mask cancellation)

(4) PMx_TXRX_EN (xxxx_0004H): Transmission/reception enabled

- TX_EN → 1: Transmission enabled
- RX_EN → 1: Reception enabled

For the slave, enable transmission or reception after the PMx_SEN and PMx_CLK signals from the master are sufficiently stable.

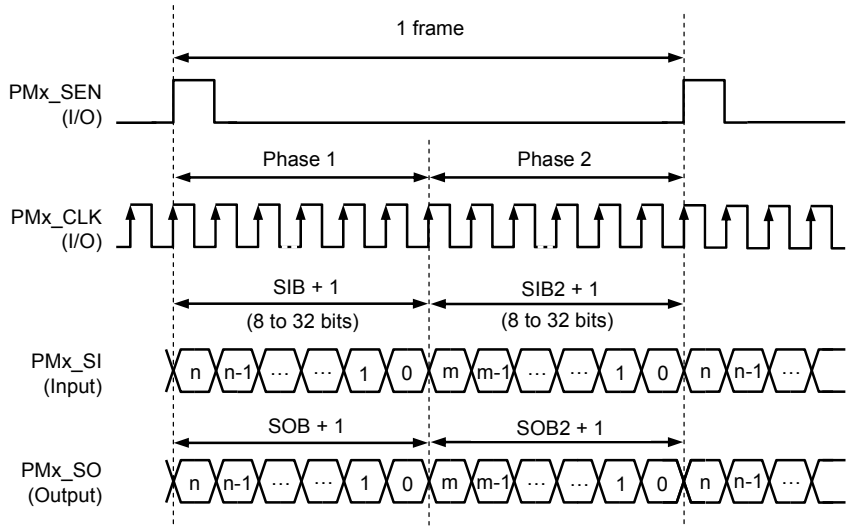
5.1.2 Examples of serial interface operations in modes 5 and 6

(1) Examples of operation in mode 5

(a) Double-phase operation

Figure 5-2. Example of Double-Phase Operation (Mode 5)

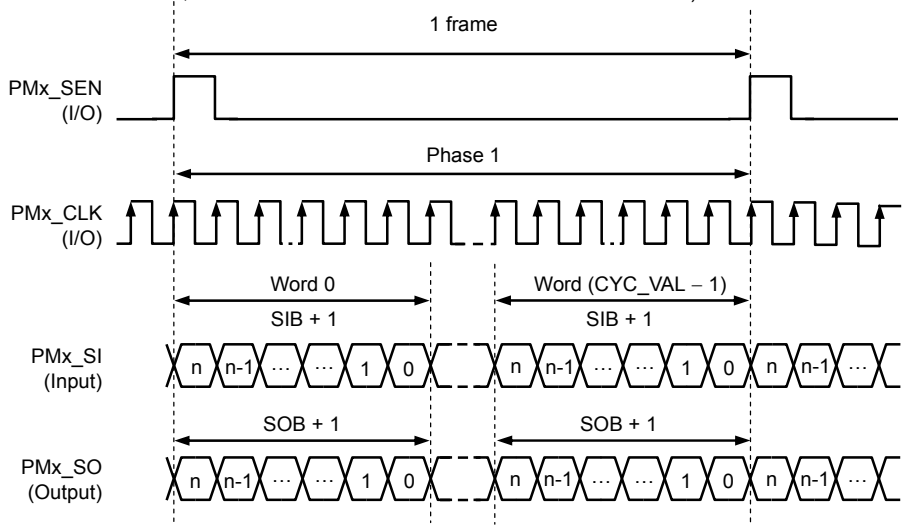
Example When PMx_CYCLE_VAL = 01H
PMx_CYCLE2_VAL2 = 01H



(b) Single-phase operation

Figure 5-3. Example of Single-Phase Operation (Mode 5)

Example When PMx_CYCLE2_VAL2 = 00H (single-phase operation)
A value from 1 to 128 is specified as the number of words for phase 1.
(When PMx_CYCLE_VAL = 00H, the number of words for phase 2 is set to between 1 and 128, and the number of valid bits to SIB2 + 1 and SOB2 + 1.)

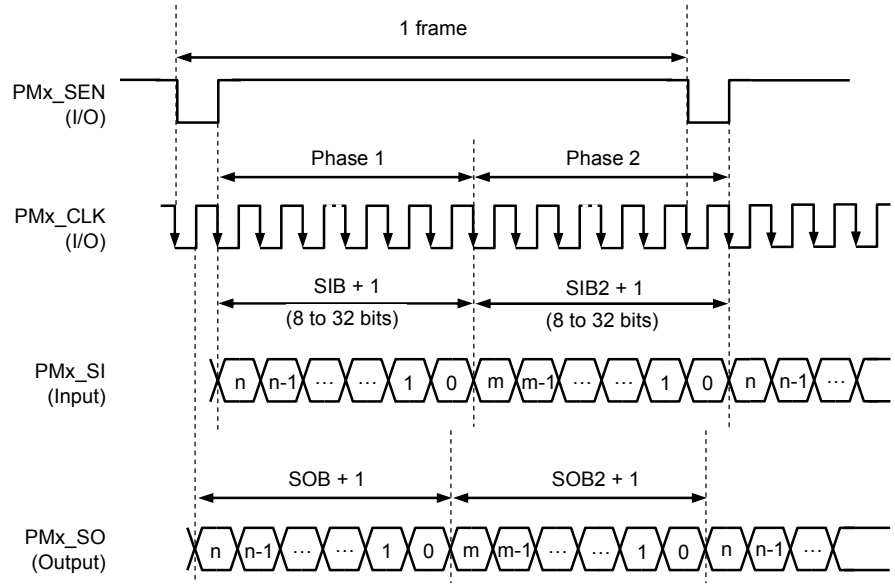


(2) Examples of operation in mode 6

(a) Double-phase operation

Figure 5-4. Example of Double-Phase Operation (Mode 6)

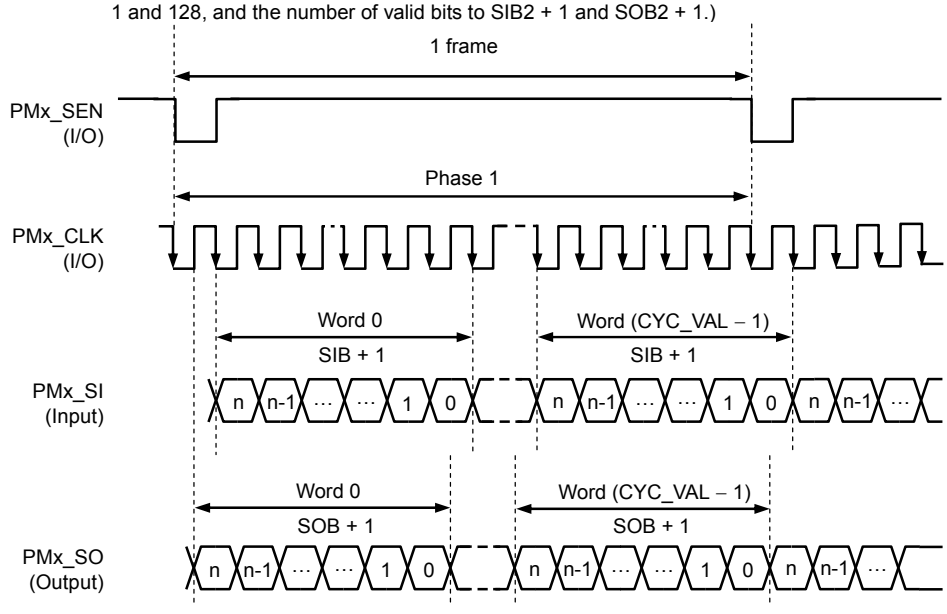
Example When PMx_CYCLE_VAL = 01H
PMx_CYCLE2_VAL2 = 01H



(b) Single-phase operation

Figure 5-5. Example of Single-Phase Operation (Mode 6)

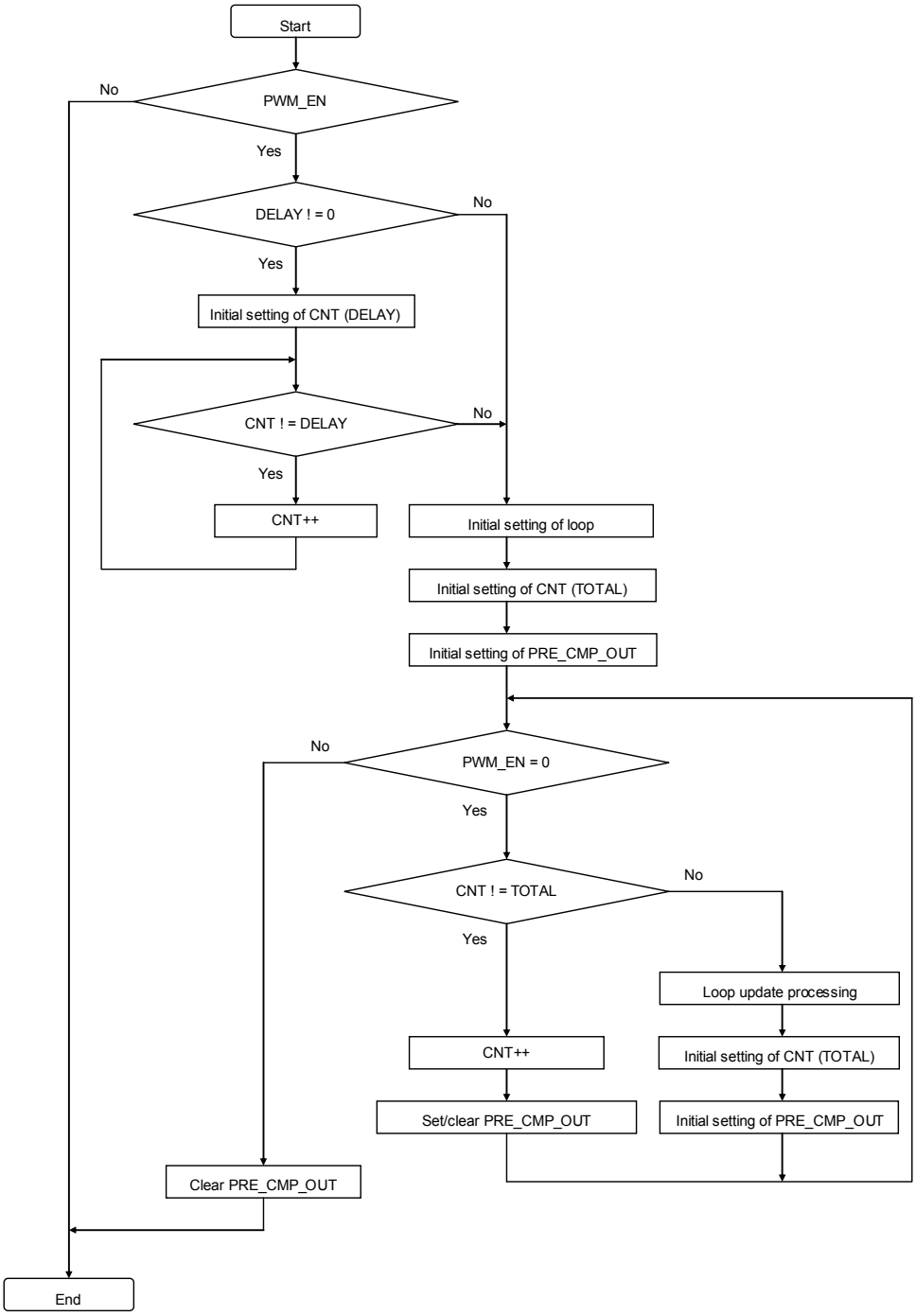
Example When PMx_CYCLE2_VAL2 = 00H (single-phase operation)
A value from 1 to 128 is specified as the number of words for phase 1.
(When PMx_CYCLE_VAL = 00H, the number of words for phase 2 is set to between 1 and 128, and the number of valid bits to SIB2 + 1 and SOB2 + 1.)



5.2 Procedure for Generating Waveform in PWM Interface

The following figure shows the processing flow of PWM_CHx_CMPn waveform generation.

Figure 5-6. Waveform Generation Flow



5.3 Cautions and Restrictions

5.3.1 Cautions on PWM interface

<1> The PWM interface uses only rising edges. It does not use falling edges.

<2> PWM_PCLK is not synchronized with PWM_PWCLK0 or PWM_PWCLK1.

<3> The PWM interface is initialized by PWM_RSTZ. Since PWM_RSTZ is synchronized with the system clock, the APB interface is synchronously reset and PWM channels 0 and 1 are asynchronously reset.

<4> Though one channel has three counters, starting these counters separately (for example, starting counter 1 after starting counter 0) is prohibited. If they are started separately, the phase between the counters cannot be guaranteed. Change the settings after making PWM_START inactive.

<5> Due to the PWM interface circuit configuration, the following restrictions apply to the settings of PWM_CHx_LEDGEn (leading edge), PWM_Chx_TEDGEn (trailing edge), and PWM_CHx_TOTALn (total).

- Keep the following relationship for the setting values: LEDGE < TEDGE ≤ TOTAL
- Setting PWM_CHx_TOTALn to 0000_0000H is prohibited.
- The counter remains at 1 if PWM_CHx_TOTALn is set to 0000_0001H.
- No LEDGE period is given if PWM_CHx_LEDGEn is set to 0000_0000H.
- Setting PWM_CHx_TEDGEn to 0000_0000H is prohibited.

<6> If the setting interval for the same source is “PWM_PCLK cycle × 4 + PWM_PWCLKx cycle × 4” or shorter, the source may not be set correctly after the source is cleared. In this case, the source is set at the next source setting timing.

<7> When restarting a counter, set the PWM_START bit to 0 and wait for at least three PWM_PWCLKx cycles.

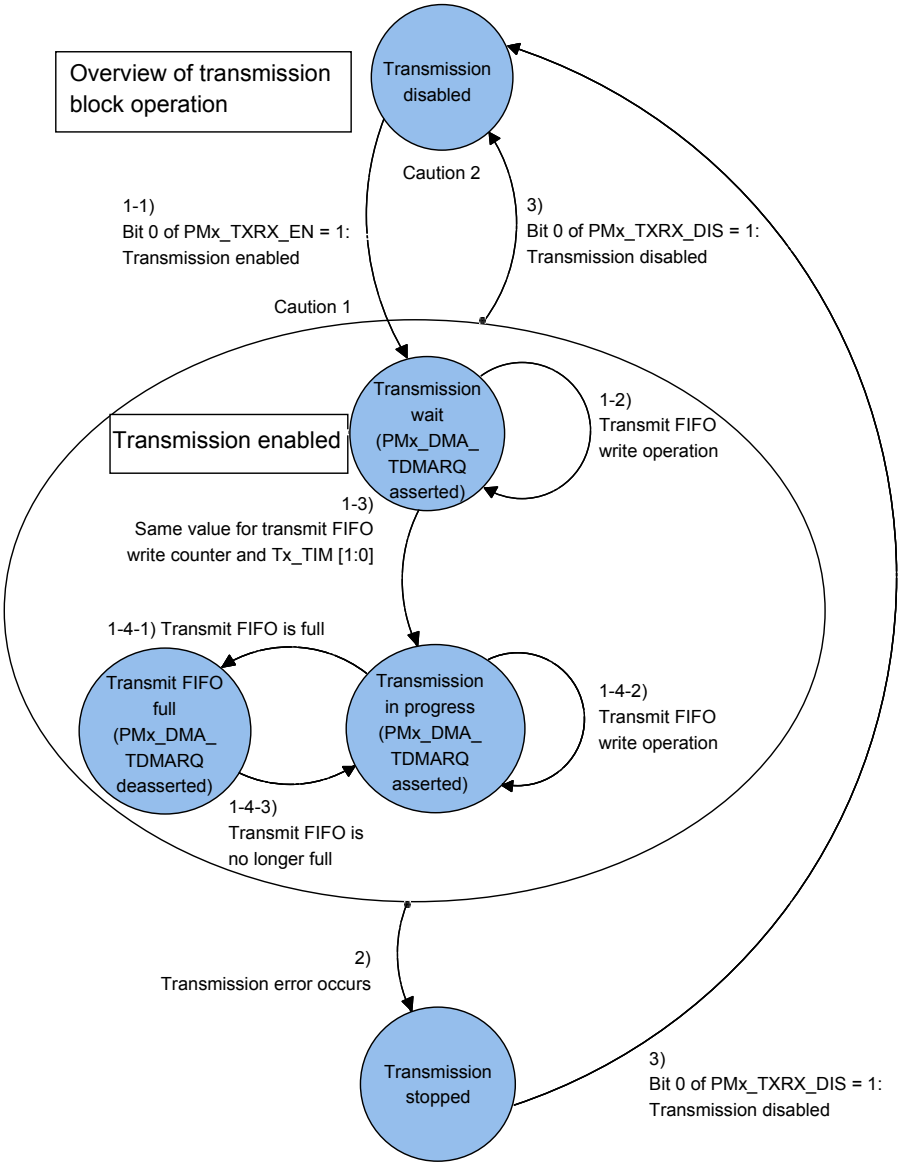
Remark x = 0 or 1
n = 0 to 2

APPENDIX A TRANSMIT/RECEIVE OPERATIONS

A.1 Status Transitions of Audio/Voice Interface

A.1.1 Transmission block status transition diagram

Figure A-1. Status Transitions in Transmission Block



Cautions 1. To enable transmission again while transmission is disabled, wait for one frame period (one word period in phase 1 or phase 2, whichever is specified as longer, in mode 5 or 6) after transmission is disabled or after a transmission re-enable interrupt occurs.

However, waiting is not required after the operation in Caution 2 is executed (because waiting has already been performed). For the slave, enable transmission after PMx_SEN and PMx_CLK from the master are sufficiently stable.

2. To change settings after transmission is stopped, wait for one frame period (one word period in phase 1 or phase 2, whichever is specified as longer, in mode 5 or 6) after transmission is disabled or after a transmission re-enable interrupt occurs.

Operation Source	Description	State		
		Transmission Disabled	Transmission Enabled	Transmission Stopped
PMx_DMA_TDMARQ pin	DMA transmission request	Fixed to OFF	ON or OFF	Fixed to OFF
PMx_SO pin	Serial data transmission	Fixed to 0	Data transmitted	Fixed to 0
PMx_INT pin	INT interrupt request	**Note 1**	With assertion	**Note 2**
PMx_SEN pin (in master)	Serial data synchronization signal	Fixed to 0	Data transmitted	Data transmitted
TX_WEN flag	Transmit data write enable flag	0 (OFF)	ON or OFF	ON/OFF retained
TX_ORE flag	Transmit overrun error detected	0 (OFF)	ON or OFF	ON/OFF retained
TX_URE flag	Transmit underrun error detected	0 (OFF)	ON or OFF	ON/OFF retained
TX_FRE flag	Transmit synchronization error detected	0 (OFF)	ON or OFF	ON or OFF
TxFIFO	Transmit FIFO	Write disabled	Write enabled	Write disabled[Note 3]
TX_W_CONT	Write counter of transmit FIFO	0	Count value incremented	Count value retained
TX_R_CONT	Read counter of transmit FIFO	0	Count value incremented	Count value retained
TX_WP_NUM[Note 4]	Word number corresponding to FIFO pointed to by write pointer in transmit FIFO	0	Shows word number	Word number retained
TX_PHASE[Note 4]	Write pointer in transmit FIFO points to either phase 1 or 2	0	Shows phase	Phase retained

Notes 1. The PMx_INT pin status depends on the receive interrupt source.

2. Asserted if the transmit interrupt source is unmasked.

3. Writing to the transmit FIFO while transmission is disabled may cause a transmit overrun error.

4. Fixed to 0 in modes other than modes 5 and 6.

A.1.2 Status transitions in transmission block

1) Bit 0 of PMx_TXRX_EN = 1 → Transmission enabled
   - For the slave, enable transmission after PMx_SEN and PMx_CLK from the master are sufficiently stable.

   1-1) When the TX_EN bit of the PMx_TXRX_EN register is set to 1, transmission is enabled.
   - The TX_WEN flag is set to 1 (transmit data write enabled).
   - The PMx_DMA_TDMARQ (DMA transmit request) signal is asserted.
   - TX_WP_NUM and TX_PHASE are initialized.

   1-2) Data is written to the transmit data register (PMx_TXQ) via the DMA controller.

   Remark The data written to PMx_TXQ is written to the 32-bit × 32-word transmit FIFO. The write pointer, TX_WP_NUM, and TX_PHASE are automatically controlled by hardware.

   1-3) When the number of words specified for the transmission start threshold (bits 6 and 5 of PMx_FUNC_SEL) are stored in the transmit FIFO, data transmission starts via the PMx_SO pin (serial data transmission), according to the specified operation mode.

   1-4) Transmit FIFO full

   1-4-1) When the transmit FIFO becomes full, the following occurs:
   - The TX_WEN flag is set to 0 (transmit data write disabled).
   - The PMx_DMA_TDMARQ (DMA transmit request) signal is deasserted.

   1-4-2) When the data in the transmit FIFO is transmitted via the PMx_SO pin and the transmit FIFO is no longer full, the transmission status returns to “in progress”, and the following occurs:
   - The TX_WEN flag is set to 1 (transmit data write enabled).
   - The PMx_DMA_TDMARQ (DMA transmit request) signal is asserted again.

   Writing to the transmit FIFO is then enabled again.

   1-4-3) An overrun error occurs if data is written to the transmit FIFO while the transmit FIFO is full. In this case, transmission is stopped.

2) Transmission error occurs → Transmission stopped

   2-1) When a transmission error[Note] occurs, transmission is stopped. If masking of the interrupt for this error has been cleared, the interrupt is issued (PMx_INT is asserted).

   Note Transmission errors include an overrun error (TX_ORE), underrun error (TX_URE), and synchronization error (TX_FRE).

   Caution When all data is successfully transmitted from the PMx_SO pin (normal completion), the transmit FIFO becomes empty and an underrun error occurs. In this case, transmission is stopped.

The operation while transmission is stopped is as follows.

- After the last data in the transmit FIFO is transmitted (before occurrence of a transmission error), PMx_SO is fixed to 0.
- The transmission error source is retained.
- The values of the transmit FIFO write counter and read counter are retained.
- Writing to the transmit FIFO is disabled.
- TX_WP_NUM and TX_PHASE are retained.

2-2) Clear the source by setting bit 0 of PMx_CLEAR to 1 if a synchronization error (TX_FRE) occurs, or by setting bit 1 of PMx_CLEAR to 1 if an underrun error (TX_URE) occurs.

3) Bit 0 of PMx_TXRX_DIS = 1 → Transmission disabled

When the TX_ENCLR bit of the PMx_TXRX_DIS register is set to 1, transmission is disabled. All transmission functions stop.

- The TX_WEN flag is set to 0 (transmit data write disabled).
- The TX_ORE (transmit overrun) and TX_URE (transmit underrun) flags are cleared.
- The values of the transmit FIFO write counter and read counter are reset to 0.

Caution To resume transmission after a transmit underrun or other error occurs, disable transmission once, then enable again as described below.

- **Bit 0 of PMx_TXRX_DIS = 1 → Transmission disabled**
- **Wait for one frame period (one word period in phase 1 or phase 2, whichever is specified as longer, in mode 5 or 6) after transmission is disabled.**
- **Bit 0 of PMx_TXRX_EN = 1 → Transmission enabled**

This is required because the data in the transmit FIFO or the counter value may have become incorrect due to an error, or the data in the transmit FIFOand its order, and the order of L-ch and R-ch data may have been changed in modes 2 to 4.

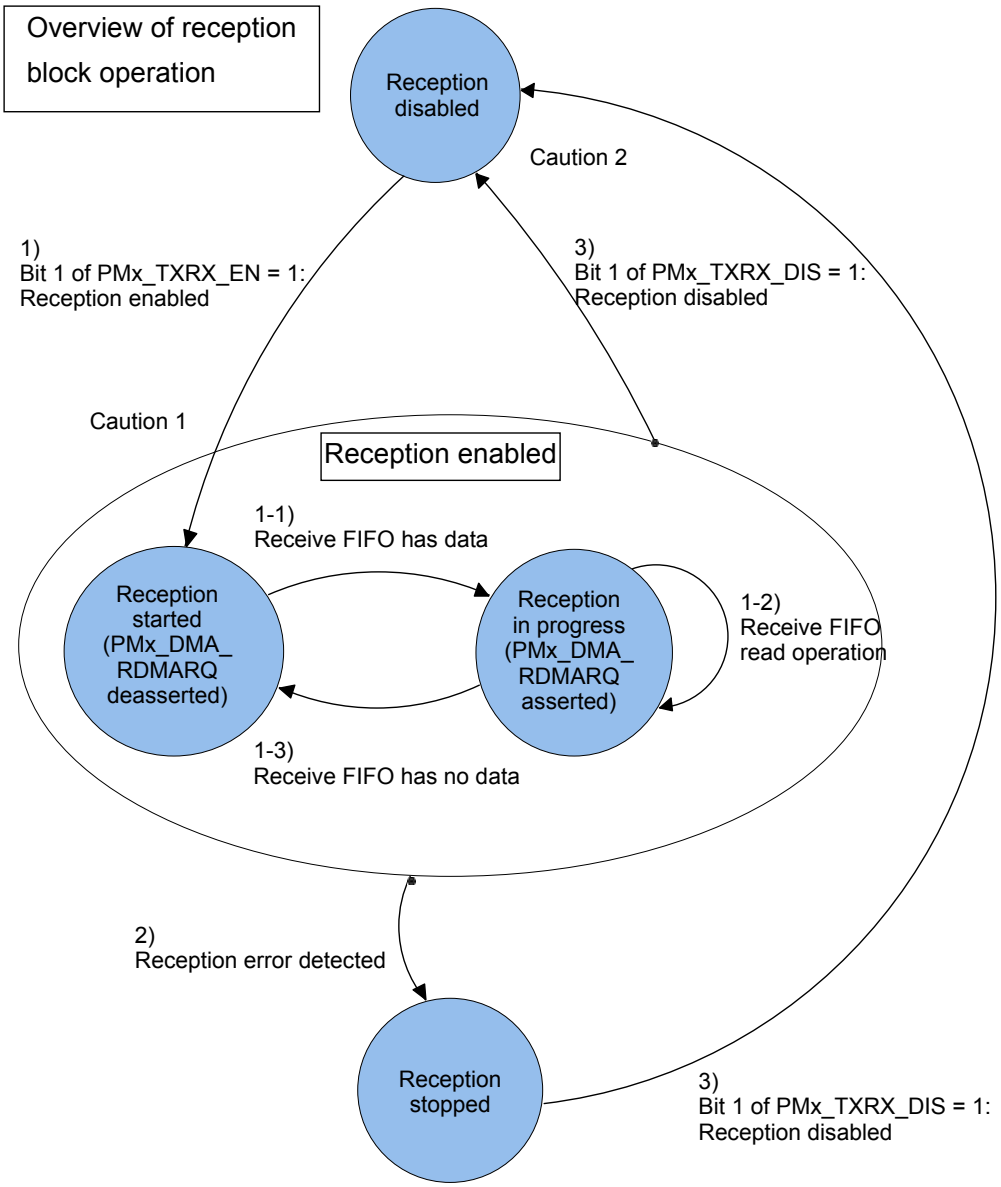
If transmission has been enabled and then disabled or a transmission re-enable interrupt occurs, also wait for one frame period (one word period in phase 1 or phase 2, whichever is specified as longer, in mode 5 or 6) before enabling and resuming transmission.

Wait for this period also when a setting needs to be changed.

After a setting is changed, an additional one-frame wait is not required before enabling transmission.

A.1.3 Reception block status transition diagram

Figure A-2. Status Transitions in Reception Block



Cautions 1. To enable reception again while reception is disabled, wait for one frame period (one word period in phase 1 or phase 2, whichever is specified as longer, in mode 5 or 6) after reception is disabled or after a reception re-enable interrupt occurs.

However, waiting is not required after the operation in Caution 2 is executed (because waiting has already been performed).

For the slave, enable reception after PMx_SEN and PMx_CLK from the master are sufficiently stable.

2. To change settings after reception is stopped, wait for one frame period (one word period in phase 1 or phase 2, whichever is specified as longer, in mode 5 or 6) after reception is disabled or after a reception re-enable interrupt occur.

Operation Source	Description	State		
		Reception Disabled	Reception Enabled	Reception Stopped
PMx_DMA_RDMARQ pin	DMA reception request	Fixed to OFF	With assertion	Fixed to OFF
PMx_SI pin	Serial data reception	Data not received	Data received	Data not received
PMx_INT pin	INT interrupt request	**Note 1**	With assertion	**Note 2**
RX_REN flag	Receive data write enable flag	0 (OFF)	ON or OFF	ON/OFF retained
RX_ORE flag	Receive overrun error detected	0 (OFF)	ON or OFF	ON/OFF retained
RX_URE flag	Receive underrun error detected	0 (OFF)	ON or OFF	ON/OFF retained
RX_FRE flag	Receive synchronization error detected	0 (OFF)	ON or OFF	ON/OFF retained
RXFIFO	Receive FIFO	Read disabled	Read enabled	Read disabled[Note 3]
RX_W_CONT	Write counter of receive FIFO	0	Count value incremented	Count value retained
RX_R_CONT	Read counter of receive FIFO	0	Count value incremented	Count value retained
RX_RP_NUM[Note 4]	Word number corresponding to FIFO pointed to by read pointer in receive FIFO	0	Shows word number	Word number retained
RX_PHASE[Note 4]	Read pointer in receive FIFO points to either phase 1 or 2	0	Shows phase	Phase retained

Notes 1. The PMx_INT pin status depends on the transmit interrupt source.

2. Asserted if the receive interrupt source is unmasked.

3. Reading from the receive FIFO while reception is disabled may cause a receive underrun error.

4. Fixed to 0 in modes other than modes 5 and 6.

A.1.4 Status transitions in reception block

1) Bit 1 of PMx_TXRX_EN = 1 → Reception enabled
   - For the slave, enable reception after PMx_SEN and PMx_CLK from the master are sufficiently stable.
     When the RX_EN bit of the PMx_TXRX_EN register is set to 1, reception is enabled.

   1-1) Serial data is received via the PMx_SI pin. When data of one word or more is received, it is transferred to the receive FIFO.
   - The RX_REN flag is set to 1 (valid data exists in the receive FIFO).
   - The PMx_DMA_RDMARQ (DMA receive request) signal is asserted.
   - RX_RP_NUM and RX_PHASE are initialized.

   1-2) The DMA controller reads the data stored in the receive FIFO from the receive data register (PMx_RXQ).

   Remark The receive FIFO consists of 32 bits × 32 words.
   The read pointer, RX_RP_NUM, and RX_PHASE are automatically controlled by hardware.

   1-3) When the PMx_RXQ register is read and the receive FIFO has no more valid data, the following occurs:
   - The RX_REN flag is set to 0 (valid data does not exist in the receive FIFO).
   - The PMx_DMA_RDMARQ (DMA receive request) signal is deasserted.

   Processing returns to 1-1) when serial data is received via the PMx_SI pin and valid data is stored in the receive
   FIFO again.

2) Reception error occurs → Reception stopped

   2-1) When a reception error[Note] occurs, reception is stopped. If masking of the interrupt for this error has been cleared, the interrupt is issued (PMx_INT is asserted).

   Note Reception errors include an overrun error (RX_ORE), underrun error (RX_URE), and synchronization error (RX_FRE).

   The operation while reception is stopped is as follows.
   - Data reception via the PMx_SI pin is stopped.
   - The reception error source is retained.
   - The values of the receive FIFO write counter and read counter are retained.
   - Reading from the receive FIFO in this status is prohibited.
   - RX_RP_NUM and RX_PHASE are retained.

   2-2) Clear the source by setting bit 4 of PMx_CLEAR to 1 if a synchronization error (RX_FRE) occurs.

3) Bit 1 of PMx_TXRX_DIS = 1 → Reception disabled

When the RX_ENCLR bit of the PMx_TXRX_DIS register is set to 1, reception is disabled. All reception functions stop.

- The RX_REN flag is set to 0 (receive FIFO read disabled).
- The source RX_ORE (receive overrun) and RX_URE (receive underrun) flags are cleared.
- The values of the receive FIFO write counter and read counter are reset to 0.

Caution To resume reception after a receive underrun or other error occurs, disable reception once, then enable it again as described below.

- **Bit 1 of PMx_TXRX_DIS = 1 → Reception disabled**
- **Wait for one frame period (one word period in phase 1 or phase 2, whichever is specified as longer, in mode 5 or 6) after reception is disabled.**
- **Bit 1 of PMx_TXRX_EN = 1 → Reception enabled**

This is required to initialize the data in the receive FIFO and the counter value if they have become incorrect due to a reception error. This processing corrects a change in the data in the receive FIFO and its order, and the order of L-ch and R-ch data in modes 2 to 4.

If reception has been enabled and then disabled or a reception re-enable interrupt occurs, also wait for one frame period (one word period in phase 1 or phase 2, whichever is specified as longer, in mode 5 or 6) before enabling and resuming reception.

Wait for this period also when a setting needs to be changed.

After a setting is changed, an additional one-frame wait is not required before enabling reception.

Revision History

Date	Revision	Comments
February 10, 2009	1.0	–
April 27, 2009	2.0	Incremental update from comments to the 1.0..
June 30, 2010	3.0	Incremental update from comments to the 2.0.

SALES OFFICES Renesas Electronics Corporation http://www.renesas.com

Refer to "http://www.renesas.com/" for the latest and detailed information.

Renesas Electronics America Inc.
2880 Scott Boulevard Santa Clara, CA 95050-2554, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130

Renesas Electronics Canada Limited
1101 Nicholson Road, Newmarket, Ontario L3Y 9C3, Canada
Tel: +1-905-898-5441, Fax: +1-905-898-3220

Renesas Electronics Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-585-100, Fax: +44-1628-585-900

Renesas Electronics Europe GmbH
Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-65030, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.
7th Floor, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100083, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.
Unit 204, 205, AZIA Center, No.1233 Lujiazui Ring Rd., Pudong District, Shanghai 200120, China
Tel: +86-21-5877-1818, Fax: +86-21-6887-7858 / -7898

Renesas Electronics Hong Kong Limited
Unit 1601-1613, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2886-9318, Fax: +852 2886-9022/9044

Renesas Electronics Taiwan Co., Ltd.
7F, No. 363 Fu Shing North Road Taipei, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.
1 harbourFront Avenue, #06-10, keppel Bay Tower, Singapore 098632
Tel: +65-6213-0200, Fax: +65-6278-8001

Renesas Electronics Malaysia Sdn.Bhd.
Unit 906, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

Renesas Electronics Korea Co., Ltd.
11F., Samik Lavied' or Bldg., 720-2 Yeoksam-Dong, Kangnam-Ku, Seoul 135-080, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141

Colophon 1.0