To our customers,

Old Company Name in Catalogs and Other Documents

On April 1st, 2010, NEC Electronics Corporation merged with Renesas Technology Corporation, and Renesas Electronics Corporation took over all the business of both companies. Therefore, although the old company name remains in this document, it is a valid Renesas Electronics document. We appreciate your understanding.

Renesas Electronics website: http://www.renesas.com

April 1st, 2010
Renesas Electronics Corporation

Issued by: Renesas Electronics Corporation (http://www.renesas.com)

Send any inquiries to http://www.renesas.com/inquiry.

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.

User's Manual

16/
32

M32C/86 Group (M32C/86, M32C/86T)

Hardware Manual

RENESAS 16/32-BIT SINGLE-CHIP
MICROCOMPUTER
M16C FAMILY / M32C/80 SERIES

Renesas Electronics
www.renesas.com

Rev.1.00 2005.09

Keep safety first in your circuit designs!

1. Renesas Technology Corp. puts the maximum effort into making semiconductor products better and more reliable, but there is always the possibility that trouble may occur with them. Trouble with semiconductors may lead to personal injury, fire or property damage. Remember to give due consideration to safety when making your circuit designs, with appropriate measures such as (i) placement of substitutive, auxiliary circuits, (ii) use of nonflammable material or (iii) prevention against any malfunction or mishap.

Notes regarding these materials

1. These materials are intended as a reference to assist our customers in the selection of the Renesas Technology Corp. product best suited to the customer's application; they do not convey any license under any intellectual property rights, or any other rights, belonging to Renesas Technology Corp. or a third party.
2. Renesas Technology Corp. assumes no responsibility for any damage, or infringement of any third-party's rights, originating in the use of any product data, diagrams, charts, programs, algorithms, or circuit application examples contained in these materials.
3. All information contained in these materials, including product data, diagrams, charts, programs and algorithms represents information on products at the time of publication of these materials, and are subject to change by Renesas Technology Corp. without notice due to product improvements or other reasons. It is therefore recommended that customers contact Renesas Technology Corp. or an authorized Renesas Technology Corp. product distributor for the latest product information before purchasing a product listed herein.
   The information described here may contain technical inaccuracies or typographical errors. Renesas Technology Corp. assumes no responsibility for any damage, liability, or other loss rising from these inaccuracies or errors.
   Please also pay attention to information published by Renesas Technology Corp. by various means, including the Renesas Technology Corp. Semiconductor home page (http://www.renesas.com).
4. When using any or all of the information contained in these materials, including product data, diagrams, charts, programs, and algorithms, please be sure to evaluate all information as a total system before making a final decision on the applicability of the information and products. Renesas Technology Corp. assumes no responsibility for any damage, liability or other loss resulting from the information contained herein.
5. Renesas Technology Corp. semiconductors are not designed or manufactured for use in a device or system that is used under circumstances in which human life is potentially at stake. Please contact Renesas Technology Corp. or an authorized Renesas Technology Corp. product distributor when considering the use of a product contained herein for any specific purposes, such as apparatus or systems for transportation, vehicular, medical, aerospace, nuclear, or undersea repeater use.
6. The prior written approval of Renesas Technology Corp. is necessary to reprint or reproduce in whole or in part these materials.
7. If these products or technologies are subject to the Japanese export control restrictions, they must be exported under a license from the Japanese government and cannot be imported into a country other than the approved destination.
   Any diversion or reexport contrary to the export control laws and regulations of Japan and/or the country of destination is prohibited.
8. Please contact Renesas Technology Corp. for further details on these materials or the products contained therein.

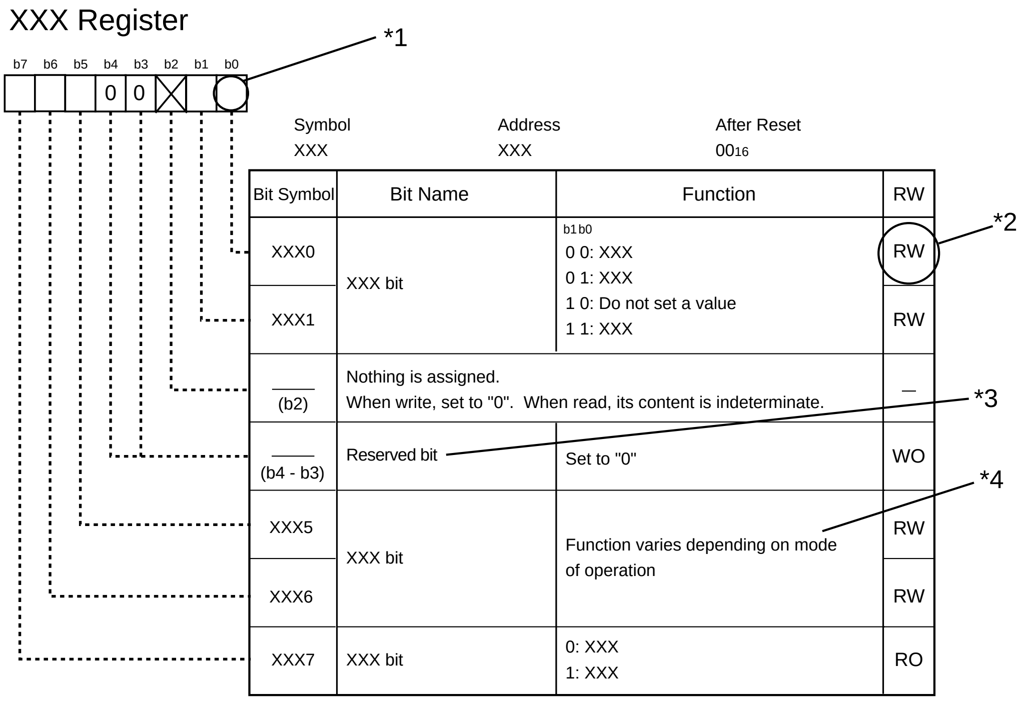
How to Use This Manual

1. Introduction

This hardware manual provides detailed information on the M32C/86 group (M32C/86, M32C/86T) microcomputers. Users are expected to have basic knowledge of electric circuits, logical circuits and microcomputers.

2. Register Diagram

The symbols, and descriptions, used for bit function in each register are shown below.



Symbol XXX
Address XXX
After Reset 00_{16}

Bit Symbol	Bit Name	Function	RW
XXX0	XXX bit	b1 b0 0 0: XXX 0 1: XXX 1 0: Do not set a value 1 1: XXX	RW
XXX1			RW
(b2)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		–
(b4 - b3)	Reserved bit	Set to "0"	WO
XXX5	XXX bit	Function varies depending on mode of operation	RW
XXX6			RW
XXX7	XXX bit	0: XXX 1: XXX	RO

*1

Blank: Set to "0" or "1" according to the application
0: Set to "0"
1: Set to "1"
X: Nothing is assigned

*2

RW: Read and write
RO: Read only
WO: Write only
–: Nothing is assigned

*3

- Reserved bit
  Reserved bit. Set to specified value.

*4

- Nothing is assigned
  Nothing is assigned to the bit concerned. As the bit may be use for future functions, set to "0" when writing to this bit.
- Do not set a value
  The operation is not guaranteed when a value is set.
- Function varies depending on mode of operation
  Bit function varies depending on peripheral function mode.
  Refer to respective register for each mode.

3. M16C Family Documents

The following documents were prepared for the M16C family. [1]

Document	Contents
Short Sheet	Hardware overview
Data Sheet	Hardware overview and electrical characteristics
Hardware Manual	Hardware specifications (pin assignments, memory maps, peripheral specifications, electrical characteristics, timing charts)
Software Manual	Detailed description of assembly instructions and microcomputer performance of each instruction
Application Note	• Application examples of peripheral functions • Sample programs • Introduction to the basic functions in the M16C family • Programming method with Assembly and C languages
RENESAS TECHNICAL UPDATE	Preliminary report about the specification of a product, a document, etc.

NOTES :

1. Before using this material, please visit the our website to verify that this is the most current document available.

Table of Contents

##

Quick Reference by Address

Address	Register	Page
0000$_{16}$		
0001$_{16}$		
0002$_{16}$		
0003$_{16}$		
0004$_{16}$	Processor Mode Register 0 (PM0)	51
0005$_{16}$	Processor Mode Register 1 (PM1)	62
0006$_{16}$	System Clock Control Register 0 (CM0)	73
0007$_{16}$	System Clock Control Register 1 (CM1)	74
0008$_{16}$		
0009$_{16}$	Address Match Interrupt Enable Register (AIER)	116
000A$_{16}$	Protect Register (PRCR)	96
000B$_{16}$	External Data Bus Width Control Register (DS)	54
000C$_{16}$	Main Clock Division Register (MCD)	75
000D$_{16}$	Oscillation Stop Detection Register (CM2)	76
000E$_{16}$	Watchdog Timer Start Register (WDTS)	122
000F$_{16}$	Watchdog Timer Control Register (WDC)	
0010$_{16}$	Address Match Interrupt Register 0 (RMAD0)	116
0011$_{16}$		
0012$_{16}$		
0013$_{16}$	Processor Mode Register 2 (PM2)	79
0014$_{16}$	Address Match Interrupt Register 1 (RMAD1)	116
0015$_{16}$		
0016$_{16}$		
0017$_{16}$	Voltage Detection Register 2 (VCR2)	44
0018$_{16}$	Address Match Interrupt Register 2 (RMAD2)	116
0019$_{16}$		
001A$_{16}$		
001B$_{16}$	Voltage Detection Register 1 (VCR1)	44
001C$_{16}$	Address Match Interrupt Register 3 (RMAD3)	116
001D$_{16}$		
001E$_{16}$		
001F$_{16}$		
0020$_{16}$		
0021$_{16}$		
0022$_{16}$		
0023$_{16}$		
0024$_{16}$		
0025$_{16}$		
0026$_{16}$	PLL Control Register 0 (PLC0)	78
0027$_{16}$	PLL Control Register 1 (PLC1)	
0028$_{16}$	Address Match Interrupt Register 4 (RMAD4)	116
0029$_{16}$		
002A$_{16}$		
002B$_{16}$		
002C$_{16}$	Address Match Interrupt Register 5 (RMAD5)	116
002D$_{16}$		
002E$_{16}$		
002F$_{16}$	Low Voltage Detection Interrupt Register (D4INT)	45

Address	Register	Page
0030$_{16}$		
0031$_{16}$		
0032$_{16}$		
0033$_{16}$		
0034$_{16}$		
0035$_{16}$		
0036$_{16}$		
0037$_{16}$		
0038$_{16}$	Address Match Interrupt Register 6 (RMAD6)	116
0039$_{16}$		
003A$_{16}$		
003B$_{16}$		
003C$_{16}$	Address Match Interrupt Register 7 (RMAD7)	116
003D$_{16}$		
003E$_{16}$		
003F$_{16}$		
0040$_{16}$		
0041$_{16}$		
0042$_{16}$		
0043$_{16}$		
0044$_{16}$		
0045$_{16}$		
0046$_{16}$		
0047$_{16}$		
0048$_{16}$	External Space Wait Control Register 0 (EWCR0)	60
0049$_{16}$	External Space Wait Control Register 1 (EWCR1)	
004A$_{16}$	External Space Wait Control Register 2 (EWCR2)	
004B$_{16}$	External Space Wait Control Register 3 (EWCR3)	
004C$_{16}$		
004D$_{16}$		
004E$_{16}$		
004F$_{16}$		
0050$_{16}$		
0051$_{16}$		
0052$_{16}$		
0053$_{16}$		
0054$_{16}$		
0055$_{16}$	Flash Memory Control Register 1 (FMR1)	399
0056$_{16}$		
0057$_{16}$	Flash Memory Control Register 0 (FMR0)	398
0058$_{16}$		
0059$_{16}$		
005A$_{16}$		
005B$_{16}$		
005C$_{16}$		
005D$_{16}$		
005E$_{16}$		
005F$_{16}$		

Blank spaces are reserved. No access is allowed.

Quick Reference by Address

Address	Register	Page
0060_{16}		
0061_{16}		
0062_{16}		
0063_{16}		
0064_{16}		
0065_{16}		
0066_{16}		
0067_{16}		
0068_{16}	DMA0 Interrupt Control Register (DM0IC)	105
0069_{16}	Timer B5 Interrupt Control Register (TB5IC)	
$006A_{16}$	DMA2 Interrupt Control Register (DM2IC)	
$006B_{16}$	UART2 Receive /ACK Interrupt Control Register (S2RIC)	
$006C_{16}$	Timer A0 Interrupt Control Register (TA0IC)	
$006D_{16}$	UART3 Receive /ACK Interrupt Control Register (S3RIC)	
$006E_{16}$	Timer A2 Interrupt Control Register (TA2IC)	
$006F_{16}$	UART4 Receive /ACK Interrupt Control Register (S4RIC)	
0070_{16}	Timer A4 Interrupt Control Register (TA4IC)	
0071_{16}	UART0 Bus Conflict Detect Interrupt Control Register (BCN0IC)/ UART3 Bus Conflict Detect Interrupt Control Register (BCN3IC)	
0072_{16}	UART0 Receive/ACK Interrupt Control Register (S0RIC)	
0073_{16}	A/D0 Conversion Interrupt Control Register (AD0IC)	
0074_{16}	UART1 Receive/ACK Interrupt Control Register (S1RIC)	
0075_{16}	Intelligent I/O Interrupt Control Register 0 (IIO0IC)/ CAN Interrupt 3 Control Register (CAN3IC)	
0076_{16}	Timer B1 Interrupt Control Register (TB1IC)	
0077_{16}	Intelligent I/O Interrupt Control Register 2 (IIO2IC)	
0078_{16}	Timer B3 Interrupt Control Register (TB3IC)	
0079_{16}	Intelligent I/O Interrupt Control Register 4 (IIO4IC)	
$007A_{16}$	INT5 Interrupt Control Register (INT5IC)	106
$007B_{16}$		
$007C_{16}$	INT3 Interrupt Control Register (INT3IC)	106
$007D_{16}$	Intelligent I/O Interrupt Control Register 8 (IIO8IC)	105
$007E_{16}$	INT1 Interrupt Control Register (INT1IC)	106
$007F_{16}$	Intelligent I/O Interrupt Control Register 10 (IIO10IC)/ CAN Interrupt 1 Control Register (CAN1IC)	105
0080_{16}		
0081_{16}	CAN Interrupt 2 Control Register (CAN2IC)	105
0082_{16}		
0083_{16}		
0084_{16}		
0085_{16}		
0086_{16}		
0087_{16}		
0088_{16}	DMA1 Interrupt Control Register (DM1IC)	105
0089_{16}	UART2 Transmit /NACK Interrupt Control Register (S2TIC)	
$008A_{16}$	DMA3 Interrupt Control Register (DM3IC)	
$008B_{16}$	UART3 Transmit /NACK Interrupt Control Register (S3TIC)	
$008C_{16}$	Timer A1 Interrupt Control Register (TA1IC)	
$008D_{16}$	UART4 Transmit /NACK Interrupt Control Register (S4TIC)	
$008E_{16}$	Timer A3 Interrupt Control Register (TA3IC)	
$008F_{16}$	UART2 Bus Conflict Detect Interrupt Control Register (BCN2IC)	

Address	Register	Page
0090_{16}	UART0 Transmit /NACK Interrupt Control Register (S0TIC)	105
0091_{16}	UART1 Bus Conflict Detect Interrupt Control Register (BCN1IC)/ UART4 Bus Conflict Detect Interrupt Control Register (BCN4IC)	
0092_{16}	UART1 Transmit/NACK Interrupt Control Register (S1TIC)	
0093_{16}	Key Input Interrupt Control Register (KUPIC)	
0094_{16}	Timer B0 Interrupt Control Register (TB0IC)	
0095_{16}	Intelligent I/O Interrupt Control Register 1 (IIO1IC)/ CAN Interrupt 4 Control Register (CAN4IC)	
0096_{16}	Timer B2 Interrupt Control Register (TB2IC)	
0097_{16}	Intelligent I/O Interrupt Control Register 3 (IIO3IC)	
0098_{16}	Timer B4 Interrupt Control Register (TB4IC)	
0099_{16}	CAN Interrupt 5 Control Register (CAN5IC)	
$009A_{16}$	INT4 Interrupt Control Register (INT4IC)	106
$009B_{16}$		
$009C_{16}$	INT2 Interrupt Control Register (INT2IC)	106
$009D_{16}$	Intelligent I/O Interrupt Control Register 9 (IIO9IC)/ CAN Interrupt 0 Control Register (CAN0IC)	105
$009E_{16}$	INT0 Interrupt Control Register (INT0IC)	106
$009F_{16}$	Exit Priority Control Register (RLVL)	107
$00A0_{16}$	Interrupt Request Register 0 (IIO0IR)	119
$00A1_{16}$	Interrupt Request Register 1 (IIO1IR)	
$00A2_{16}$	Interrupt Request Register 2 (IIO2IR)	
$00A3_{16}$	Interrupt Request Register 3 (IIO3IR)	
$00A4_{16}$	Interrupt Request Register 4 (IIO4IR)	
$00A5_{16}$	Interrupt Request Register 5 (IIO5IR)	
$00A6_{16}$		
$00A7_{16}$		
$00A8_{16}$	Interrupt Request Register 8 (IIO8IR)	119
$00A9_{16}$	Interrupt Request Register 9 (IIO9IR)	
$00AA_{16}$	Interrupt Request Register 10 (IIO10IR)	
$00AB_{16}$	Interrupt Request Register 11 (IIO11IR)	
$00AC_{16}$		
$00AD_{16}$		
$00AE_{16}$		
$00AF_{16}$		
$00B0_{16}$	Interrupt Enable Register 0 (IIO0IE)	120
$00B1_{16}$	Interrupt Enable Register 1 (IIO1IE)	
$00B2_{16}$	Interrupt Enable Register 2 (IIO2IE)	
$00B3_{16}$	Interrupt Enable Register 3 (IIO3IE)	
$00B4_{16}$	Interrupt Enable Register 4 (IIO4IE)	
$00B5_{16}$	Interrupt Enable Register 5 (IIO5IE)	
$00B6_{16}$		
$00B7_{16}$		
$00B8_{16}$	Interrupt Enable Register 8 (IIO8IE)	120
$00B9_{16}$	Interrupt Enable Register 9 (IIO9IE)	
$00BA_{16}$	Interrupt Enable Register 10 (IIO10IE)	
$00BB_{16}$	Interrupt Enable Register 11 (IIO11IE)	
$00BC_{16}$		
$00BD_{16}$		
$00BE_{16}$		
$00BF_{16}$		

Blank spaces are reserved. No access is allowed.

Quick Reference by Address

Address	Register	Page
00C0$_{16}$		
00C1$_{16}$		
00C2$_{16}$		
00C3$_{16}$		
00C4$_{16}$		
00C5$_{16}$		
00C6$_{16}$		
00C7$_{16}$		
00C8$_{16}$		
00C9$_{16}$		
00CA$_{16}$		
00CB$_{16}$		
00CC$_{16}$		
00CD$_{16}$		
00CE$_{16}$		
00CF$_{16}$		
00D0$_{16}$		
00D1$_{16}$		
00D2$_{16}$		
00D3$_{16}$		
00D4$_{16}$		
00D5$_{16}$		
00D6$_{16}$		
00D7$_{16}$		
00D8$_{16}$		
00D9$_{16}$		
00DA$_{16}$		
00DB$_{16}$		
00DC$_{16}$		
00DD$_{16}$		
00DE$_{16}$		
00DF$_{16}$		
00E0$_{16}$		
00E1$_{16}$		
00E2$_{16}$		
00E3$_{16}$		
00E4$_{16}$		
00E5$_{16}$		
00E6$_{16}$		
00E7$_{16}$		
00E8$_{16}$ 00E9$_{16}$	SI/O Receive Buffer Register0 (G0RB)	290
00EA$_{16}$	Transmit Buffer/Receive Data Register 0 (G0TB/G0DR)	296
00EB$_{16}$		
00EC$_{16}$	Receive Input Register 0 (G0RI)	289
00ED$_{16}$	SI/O Communication Mode Register 0 (G0MR)	291
00EE$_{16}$	Transmit Output Register 0 (G0TO)	289
00EF$_{16}$	SI/O Communication Control Register 0 (G0CR)	290

Address	Register	Page
00F0$_{16}$	Data Compare Register 00 (G0CMP0)	297
00F1$_{16}$	Data Compare Register 01 (G0CMP1)	
00F2$_{16}$	Data Compare Register 02 (G0CMP2)	
00F3$_{16}$	Data Compare Register 03 (G0CMP3)	
00F4$_{16}$	Data Mask Register 00 (G0MSK0)	
00F5$_{16}$	Data Mask Register 01 (G0MSK1)	
00F6$_{16}$	Communication Clock Select Register (CCS)	298
00F7$_{16}$		
00F8$_{16}$ 00F9$_{16}$	Receive CRC Code Register 0 (G0RCRC)	297
00FA$_{16}$ 00FB$_{16}$	Tramsmit CRC Code Register 0 (G0TCRC)	
00FC$_{16}$	SI/O Extended Mode Register 0 (G0EMR)	292
00FD$_{16}$	SI/O Extended Receive Control Register 0 (G0ERC)	294
00FE$_{16}$	SI/O Special Communication Interrupt Detect Register 0 (G0IRF)	295
00FF$_{16}$	SI/O Extended Transmit Control Register 0 (G0ETC)	293
0100$_{16}$ 0101$_{16}$	Time Measurement Register 10 (G1TM0)/ Waveform Generating Register 10 (G1PO0)	268/ 269
0102$_{16}$ 0103$_{16}$	Time Measurement Register 11 (G1TM1)/ Waveform Generating Register 11 (G1PO1)	
0104$_{16}$ 0105$_{16}$	Time Measurement Register 12 (G1TM2)/ Waveform Generating Register 12 (G1PO2)	
0106$_{16}$ 0107$_{16}$	Time Measurement Register 13 (G1TM3)/ Waveform Generating Register 13 (G1PO3)	
0108$_{16}$ 0109$_{16}$	Time Measurement Register 14 (G1TM4)/ Waveform Generating Register 14 (G1PO4)	
010A$_{16}$ 010B$_{16}$	Time Measurement Register 15 (G1TM5)/ Waveform Generating Register 16 (G1PO5)	
010C$_{16}$ 010D$_{16}$	Time Measurement Register 16 (G1TM6)/ Waveform Generating Register 16 (G1PO6)	
010E$_{16}$ 010F$_{16}$	Time Measurement Register 17 (G1TM7)/ Waveform Generating Register 17 (G1PO7)	
0110$_{16}$	Waveform Generating Control Register 10 (G1POCR0)	268
0111$_{16}$	Waveform Generating Control Register 11 (G1POCR1)	
0112$_{16}$	Waveform Generating Control Register 12 (G1POCR2)	
0113$_{16}$	Waveform Generating Control Register 13 (G1POCR3)	
0114$_{16}$	Waveform Generating Control Register 14 (G1POCR4)	
0115$_{16}$	Waveform Generating Control Register 15 (G1POCR5)	
0116$_{16}$	Waveform Generating Control Register 16 (G1POCR6)	
0117$_{16}$	Waveform Generating Control Register 17 (G1POCR7)	
0118$_{16}$	Time Measurement Control Register 10 (G1TMCR0)	267
0119$_{16}$	Time Measurement Control Register 11 (G1TMCR1)	
011A$_{16}$	Time Measurement Control Register 12 (G1TMCR2)	
011B$_{16}$	Time Measurement Control Register 13 (G1TMCR3)	
011C$_{16}$	Time Measurement Control Register 14 (G1TMCR4)	
011D$_{16}$	Time Measurement Control Register 15 (G1TMCR5)	
011E$_{16}$	Time Measurement Control Register 16 (G1TMCR6)	
011F$_{16}$	Time Measurement Control Register 17 (G1TMCR7)	

Blank spaces are reserved. No access is allowed.

Quick Reference by Address

Address	Register	Page
0120_{16}	Base Timer Register1 (G1BT)	265
0121_{16}		
0122_{16}	Base Timer Control Register 10 (G1BCR0)	
0123_{16}	Base Timer Control Register 11 (G1BCR1)	266
0124_{16}	Time Measurement Prescaler Register 16 (G1TPR6)	267
0125_{16}	Time Measurement Prescaler Register 17 (G1TPR7)	
0126_{16}	Function Enable Register 1 (G1FE)	270
0127_{16}	Function Select Register 1 (G1FS)	269
0128_{16}	SI/O Receive Buffer Register 1 (G1RB)	290
0129_{16}		
$012A_{16}$	Transmit Buffer/Receive Data Register 1 (G1TB/G1DR)	296
$012B_{16}$		
$012C_{16}$	Receive Input Register 1 (G1RI)	289
$012D_{16}$	SI/O Communication Mode Register 1 (G1MR)	291
$012E_{16}$	Transmit Output Register 1 (G1TO)	289
$012F_{16}$	SI/O Communication Control Register 1 (G1CR)	290
0130_{16}	Data Compare Register 10 (G1CMP0)	297
0131_{16}	Data Compare Register 11 (G1CMP1)	
0132_{16}	Data Compare Register 12 (G1CMP2)	
0133_{16}	Data Compare Register 13 (G1CMP3)	
0134_{16}	Data Mask Register 10 (G1MSK0)	
0135_{16}	Data Mask Register 11 (G1MSK1)	
0136_{16}		
0137_{16}		
0138_{16}	Receive CRC Code Register1 (G1RCRC)	297
0139_{16}		
$013A_{16}$	Transmit CRC Code Register1 (G1TCRC)	
$013B_{16}$		
$013C_{16}$	SI/O Extended Mode Register 1 (G1EMR)	292
$013D_{16}$	SI/O Extended Receive Control Register 1 (G1ERC)	294
$013E_{16}$	SI/O Special Communication Interrupt Detect Register 1 (G1IRF)	296
$013F_{16}$	SI/O Extended Transmit Control Register 1 (G1ETC)	293
0140_{16}		
0141_{16}		
0142_{16}		
0143_{16}		
0144_{16}		
0145_{16}		
0146_{16}		
0147_{16}		
0148_{16}		
0149_{16}		
$014A_{16}$		
$014B_{16}$		
$014C_{16}$		
$014D_{16}$		
$014E_{16}$		
$014F_{16}$		

Address	Register	Page
0150_{16}		
0151_{16}		
0152_{16}		
0153_{16}		
0154_{16}		
0155_{16}		
0156_{16}		
0157_{16}		
0158_{16}		
0159_{16}		
$015A_{16}$		
$015B_{16}$		
$015C_{16}$		
$015D_{16}$		
$015E_{16}$		
$015F_{16}$		
0160_{16}		
0161_{16}		
0162_{16}		
0163_{16}		
0164_{16}		
0165_{16}		
0166_{16}		
0167_{16}		
0168_{16}		
0169_{16}		
$016A_{16}$		
$016B_{16}$		
$016C_{16}$		
$016D_{16}$		
$016E_{16}$		
$016F_{16}$		
0170_{16}		
0171_{16}		
0172_{16}		
0173_{16}		
0174_{16}		
0175_{16}		
0176_{16}		
0177_{16}		
0178_{16}	Input Function Select Register (IPS)	385
0179_{16}	Input Function Select Register A (IPSA)	386
$017A_{16}$		
$017B_{16}$		
$017C_{16}$		
$017D_{16}$ to $01DF_{16}$		

Blank spaces are reserved. No access is allowed.

Quick Reference by Address

Address	Register	Page
01E0$_{16}$	CAN0 Message Slot Buffer 0 Standard ID0 (C0SLOT0_0)	354
01E1$_{16}$	CAN0 Message Slot Buffer 0 Standard ID1 (C0SLOT0_1)	
01E2$_{16}$	CAN0 Message Slot Buffer 0 Extended ID0 (C0SLOT0_2)	355
01E3$_{16}$	CAN0 Message Slot Buffer 0 Extended ID1 (C0SLOT0_3)	
01E4$_{16}$	CAN0 Message Slot Buffer 0 Extended ID2 (C0SLOT0_4)	356
01E5$_{16}$	CAN0 Message Slot Buffer 0 Data Length Code (C0SLOT0_5)	
01E6$_{16}$	CAN0 Message Slot Buffer 0 Data 0 (C0SLOT0_6)	357
01E7$_{16}$	CAN0 Message Slot Buffer 0 Data 1 (C0SLOT0_7)	
01E8$_{16}$	CAN0 Message Slot Buffer 0 Data 2 (C0SLOT0_8)	
01E9$_{16}$	CAN0 Message Slot Buffer 0 Data 3 (C0SLOT0_9)	
01EA$_{16}$	CAN0 Message Slot Buffer 0 Data 4 (C0SLOT0_10)	
01EB$_{16}$	CAN0 Message Slot Buffer 0 Data 5 (C0SLOT0_11)	
01EC$_{16}$	CAN0 Message Slot Buffer 0 Data 6 (C0SLOT0_12)	
01ED$_{16}$	CAN0 Message Slot Buffer 0 Data 7 (C0SLOT0_13)	
01EE$_{16}$	CAN0 Message Slot Buffer 0 Time Stamp High-Order (C0SLOT0_14)	
01EF$_{16}$	CAN0 Message Slot Buffer 0 Time Stamp Low-Order (C0SLOT0_15)	
01F0$_{16}$	CAN0 Message Slot Buffer 1 Standard ID0 (C0SLOT1_0)	354
01F1$_{16}$	CAN0 Message Slot Buffer 1 Standard ID1 (C0SLOT1_1)	
01F2$_{16}$	CAN0 Message Slot Buffer 1 Extended ID0 (C0SLOT1_2)	355
01F3$_{16}$	CAN0 Message Slot Buffer 1 Extended ID1 (C0SLOT1_3)	
01F4$_{16}$	CAN0 Message Slot Buffer 1 Extended ID2 (C0SLOT1_4)	356
01F5$_{16}$	CAN0 Message Slot Buffer 1 Data Length Code (C0SLOT1_5)	
01F6$_{16}$	CAN0 Message Slot Buffer 1 Data 0 (C0SLOT1_6)	357
01F7$_{16}$	CAN0 Message Slot Buffer 1 Data 1 (C0SLOT1_7)	
01F8$_{16}$	CAN0 Message Slot Buffer 1 Data 2 (C0SLOT1_8)	
01F9$_{16}$	CAN0 Message Slot Buffer 1 Data 3 (C0SLOT1_9)	
01FA$_{16}$	CAN0 Message Slot Buffer 1 Data 4 (C0SLOT1_10)	
01FB$_{16}$	CAN0 Message Slot Buffer 1 Data 5 (C0SLOT1_11)	
01FC$_{16}$	CAN0 Message Slot Buffer 1 Data 6 (C0SLOT1_12)	
01FD$_{16}$	CAN0 Message Slot Buffer 1 Data 7 (C0SLOT1_13)	
01FE$_{16}$	CAN0 Message Slot Buffer 1 Time Stamp High-Order (C0SLOT1_14)	
01FF$_{16}$	CAN0 Message Slot Buffer 1 Time Stamp Low-Order (C0SLOT1_15)	
0200$_{16}$ 0201$_{16}$	CAN0 Control Register0 (C0CTLR0)	318
0202$_{16}$ 0203$_{16}$	CAN0 Status Register (C0STR)	323
0204$_{16}$ 0205$_{16}$	CAN0 Extended ID Register (C0IDR)	326
0206$_{16}$ 0207$_{16}$	CAN0 Configuration Register (C0CONR)	327
0208$_{16}$ 0209$_{16}$	CAN0 Time Stamp Register (C0TSR)	330
020A$_{16}$	CAN0 Transmit Error Count Register (C0TEC)	331
020B$_{16}$	CAN0 Receive Error Count Register (C0REC)	
020C$_{16}$ 020D$_{16}$	CAN0 Slot Interrupt Status Register (C0SISTR)	332
020E$_{16}$		
020F$_{16}$		

Address	Register	Page
0210$_{16}$ 0211$_{16}$	CAN0 Slot Interrupt Mask Register (C0SIMKR)	334
0212$_{16}$		
0213$_{16}$		
0214$_{16}$	CAN0 Error Interrupt Mask Register (C0EIMKR)	335
0215$_{16}$	CAN0 Error Interrupt Status Register (C0EISTR)	336
0216$_{16}$	CAN0 Error Cause Register (C0EFR)	337
0217$_{16}$	CAN0 Baud Rate Prescaler (C0BRP)	329
0218$_{16}$		
0219$_{16}$	CAN0 Mode Register (C0MDR)	338
021A$_{16}$		
021B$_{16}$		
021C$_{16}$		
021D$_{16}$		
021E$_{16}$		
021F$_{16}$		
0220$_{16}$ 0221$_{16}$	CAN0 Single Shot Control Register (C0SSCTLR)	340
0222$_{16}$		
0223$_{16}$		
0224$_{16}$ 0225$_{16}$	CAN0 Single Shot Status Register (C0SSSTR)	341
0226$_{16}$		
0227$_{16}$		
0228$_{16}$	CAN0 Global Mask Register Standard ID0 (C0GMR0)	342
0229$_{16}$	CAN0 Global Mask Register Standard ID1 (C0GMR1)	343
022A$_{16}$	CAN0 Global Mask Register Extended ID0 (C0GMR2)	344
022B$_{16}$	CAN0 Global Mask Register Extended ID1 (C0GMR3)	345
022C$_{16}$	CAN0 Global Mask Register Extended ID2 (C0GMR4)	356
022D$_{16}$		
022E$_{16}$		
022F$_{16}$		
0230$_{16}$	CAN0 Message Slot 0 Control Register (C0MCTL0)/ CAN0 Local Mask Register A Standard ID0 (C0LMAR0)	349/ 342
0231$_{16}$	CAN0 Message Slot 1 Control Register (C0MCTL1)/ CAN0Local Mask Register A Standard ID1 (C0LMAR1)	349/ 343
0232$_{16}$	CAN0 Message Slot 2 Control Register (C0MCTL2)/ CAN0 Local Mask Register A Extended ID0 (C0LMAR2)	349/ 344
0233$_{16}$	CAN0 Message Slot 3 Control Register (C0MCTL3)/ CAN0 Local Mask Register A Extended ID1 (C0LMAR3)	349/ 345
0234$_{16}$	CAN0 Message Slot 4 Control Register (C0MCTL4)/ CAN0 Local Mask Register A Extended ID2 (C0LMAR4)	349/ 346
0235$_{16}$	CAN0 Message Slot 5 Control Register (C0MCTL5)	349
0236$_{16}$	CAN0 Message Sot 6 Control Register (C0MCTL6)	
0237$_{16}$	CAN0 Message Slot 7 Control Register (C0MCTL7)	
0238$_{16}$	CAN0 Message Slot 8 Control register (C0MCTL8)/ CAN0 Local Mask Register B Standard ID0 (C0LMBR0)	349/ 342

Blank spaces are reserved . No access is allowed.

Quick Reference by Address

Address	Register	Page
0239$_{16}$	CAN0 Message Slot 9 Control Register (C0MCTL9)/ CAN0 Local Mask Register B Standard ID1 (C0LMBR1)	349/ 343
023A$_{16}$	CAN0 Message Slot 10 Control Register (C0MCTL10)/ CAN0 Local Mask Register B Extended ID0 (C0LMBR2)	349/ 344
023B$_{16}$	CAN0 Message Slot 11 Control Register (C0MCTL11)/ CAN0 Local Mask Register B Extended ID1 (C0LMBR3)	349/ 345
023C$_{16}$	CAN0 Message Slot 12 Control Register (C0MCTL12)/ CAN0 Local Mask Register B Extended ID2 (C0LMBR4)	349/ 346
023D$_{16}$	CAN0 Message Slot 13 Control Register (C0MCTL13)	349
023E$_{16}$	CAN0 Message Slot 14 Control Register (C0MCTL14)	
023F$_{16}$	CAN0 Message Slot 15 Control Register(C0MCTL15)	
0240$_{16}$	CAN0 Slot Buffer Select Register (C0SBS)	353
0241$_{16}$	CAN0 Control Register 1 (C0CTLR1)	321
0242$_{16}$	CAN0 Sleep Control Register (C0SLPR)	322
0243$_{16}$		
0244$_{16}$ 0245$_{16}$	CAN0 Acceptance Filter Support Register (C0AFS)	358
0246$_{16}$		
0247$_{16}$		
0248$_{16}$		
0249$_{16}$		
024A$_{16}$		
024B$_{16}$		
024C$_{16}$		
024D$_{16}$		
024E$_{16}$		
024F$_{16}$		
0250$_{16}$	CAN1 Slot Buffer Select Register (C1SBS)	353
0251$_{16}$	CAN1 Control Register 1 (C1CTLR1)	321
0252$_{16}$	CAN1 Sleep Control Register (C1SLPR)	322
0253$_{16}$		
0254$_{16}$ 0255$_{16}$	CAN1 Acceptance Filter Support Register (C1AFS)	358
0256$_{16}$		
0257$_{16}$		
0258$_{16}$		
0259$_{16}$		
025A$_{16}$		
025B$_{16}$		
025C$_{16}$		
025D$_{16}$		
025E$_{16}$		
025F$_{16}$		

Address	Register	Page
0260$_{16}$	CAN1 Message Slot Buffer 0 Standard ID0 (C1SLOT0_0)	354
0261$_{16}$	CAN1 Message Slot Buffer 0 Standard ID1 (C1SLOT0_1)	
0262$_{16}$	CAN1 Message Slot Buffer 0 Extended ID0 (C1SLOT0_2)	355
0263$_{16}$	CAN1 Message Slot Buffer 0 Extended ID1 (C1SLOT0_3)	
0264$_{16}$	CAN1 Message Slot Buffer 0 Extended ID2 (C1SLOT0_4)	356
0265$_{16}$	CAN1 Message Slot Buffer 0 Data Length Code (C1SLOT0_5)	
0266$_{16}$	CAN1 Message Slot Buffer 0 Data 0 (C1SLOT0_6)	357
0267$_{16}$	CAN1 Message Slot Buffer 0 Data 1 (C1SLOT0_7)	
0268$_{16}$	CAN1 Message Slot Buffer 0 Data 2 (C1SLOT0_8)	
0269$_{16}$	CAN1 Message Slot Buffer 0 Data 3 (C1SLOT0_9)	
026A$_{16}$	CAN1 Message Slot Buffer 0 Data 4 (C1SLOT0_10)	
026B$_{16}$	CAN1 Message Slot Buffer 0 Data 5 (C1SLOT0_11)	
026C$_{16}$	CAN1 Message Slot Buffer 0 Data 6 (C1SLOT0_12)	
026D$_{16}$	CAN1 message Slot Buffer 0 Data 7 (C1SLOT0_13)	
026E$_{16}$	CAN1 Message Slot Buffer 0 Time Stamp High-Order (C1SLOT0_14)	
026F$_{16}$	CAN1 Message Slot Buffer 0 Time Stamp Low-Order (C1SLOT0_15)	
0270$_{16}$	CAN1 Message Slot Buffer 1 Standard ID0 (C1SLOT1_0)	354
0271$_{16}$	CAN1 Message Slot Buffer 1 Standard ID1 (C1SLOT1_1)	
0272$_{16}$	CAN1 Message Slot Buffer 1 Extended ID0 (C1SLOT1_2)	355
0273$_{16}$	CAN1 Message Slot Buffer 1 Extended ID1 (C1SLOT1_3)	
0274$_{16}$	CAN1 Message Slot Buffer 1 Extended ID2 (C1SLOT1_4)	356
0275$_{16}$	CAN1 Message Slot Buffer 1 Data Length Code (C1SLOT1_5)	
0276$_{16}$	CAN1 Message Slot Buffer 1 Data 0 (C1SLOT1_6)	357
0277$_{16}$	CAN1 Message Slot Buffer 1 Data 1 (C1SLOT1_7)	
0278$_{16}$	CAN1 Message Slot Buffer 1 Data 2 (C1SLOT1_8)	
0279$_{16}$	CAN1 Message Slot Buffer 1 Data 3 (C1SLOT1_9)	
027A$_{16}$	CAN1 Message Slot Buffer 1 Data 4 (C1SLOT1_10)	
027B$_{16}$	CAN1 Message Slot Buffer 1 Data 5 (C1SLOT1_11)	
027C$_{16}$	CAN1 Message Slot Buffer 1 Data 6 (C1SLOT1_12)	
027D$_{16}$	CAN1 Message Slot Buffer 1 Data 7 (C1SLOT1_13)	
027E$_{16}$	CAN1 Message Slot Buffer 1 Time Stamp High-Order (C1SLOT1_14)	
027F$_{16}$	CAN1 Message Slot Buffer 1 Time Stamp Low-Order (C1SLOT1_15)	
0280$_{16}$ 0281$_{16}$	CAN1 Control Register0 (C1CTLR0)	318
0282$_{16}$ 0283$_{16}$	CAN1 Status Register (C1STR)	323
0284$_{16}$ 0285$_{16}$	CAN1 Extended ID Register (C1IDR)	326
0286$_{16}$ 0287$_{16}$	CAN1 Configuration Register (C1CONR)	327
0288$_{16}$ 0289$_{16}$	CAN1 Time Stamp Register (C1TSR)	330
028A$_{16}$	CAN1 Transmit Error Count Register (C1TEC)	331
028B$_{16}$	CAN1 Receive Error Count Register (C1REC)	
028C$_{16}$ 028D$_{16}$	CAN1 Slot Interrupt Control Register (C1SISTR)	332
028E$_{16}$		
028F$_{16}$		

Blank spaces are reserved. No access is allowed.

Quick Reference by Address

Address	Register	Page
0290$_{16}$ 0291$_{16}$	CAN1 Slot Interrupt Mask Register (C1SIMKR)	334
0292$_{16}$		
0293$_{16}$		
0294$_{16}$	CAN1 Error Interrupt Mask Register (C1EIMKR)	335
0295$_{16}$	CAN1 Error Interrupt Status Register (C1EISTR)	336
0296$_{16}$	CAN1 Error Factor Register (C1EFR)	337
0297$_{16}$	CAN1 Baud Rate Prescaler (C1BRP)	329
0298$_{16}$		
0299$_{16}$	CAN1 Mode Register (C1MDR)	338
029A$_{16}$		
029B$_{16}$		
029C$_{16}$		
029D$_{16}$		
029E$_{16}$		
029F$_{16}$		
02A0$_{16}$ 02A1$_{16}$	CAN1 Single Shot Control Register (C1SSCTLR)	340
02A2$_{16}$		
02A3$_{16}$		
02A4$_{16}$ 02A5$_{16}$	CAN1 Single Shot Status Register (C1SSSTR)	341
02A6$_{16}$		
02A7$_{16}$		
02A8$_{16}$	CAN1 Global Mask Register Standard ID0 (C1GMR0)	342
02A9$_{16}$	CAN1 Global Mask Register Standard ID1 (C1GMR1)	343
02AA$_{16}$	CAN1 Global Mask Register Extended ID0 (C1GMR2)	344
02AB$_{16}$	CAN1 Global Mask Register Extended ID1 (C1GMR3)	345
02AC$_{16}$	CAN1 Global Mask Register Extended ID2 (C1GMR4)	346
02AD$_{16}$		
02AE$_{16}$		
02AF$_{16}$		
02B0$_{16}$	CAN1 Message Slot 0 Control Register (C1MCTL0)/ CAN1 Local Mask Register A Standard ID0 (C1LMAR0)	349/ 342
02B1$_{16}$	CAN1 Message Slot 1 Control Register (C1MCTL1)/ CAN1 Local Mask Register A Standard ID1 (C1LMAR1)	349/ 343
02B2$_{16}$	CAN1 Message Slot 2 Control Register (C1MCTL2)/ CAN1 Local Mask Register A Extended ID0 (C1LMAR2)	349/ 344
02B3$_{16}$	CAN1 Message Slot 3 Control Register (C1MCTL3)/ CAN1 Local Mask Register A Extended ID1 (C1LMAR3)	349/ 345
02B4$_{16}$	CAN1 Message Slot 4 Control Register (C1MCTL4)/ CAN1 Local Mask Register A Extended ID2 (C1LMAR4)	349/ 346
02B5$_{16}$	CAN1 Message Slot 5 Control Register (C1MCTL5)	349
02B6$_{16}$	CAN1 Message Slot 6 Control Register (C1MCTL6)	
02B7$_{16}$	CAN1 Message Slot 7 Control Register (C1MCTL7)	
02B8$_{16}$	CAN1 Message Slot 8 Control Register (C1MCTL8)/ CAN1 Local Mask Register B Standard ID0 (C1LMBR0)	349/ 342
02B9$_{16}$	CAN1 Message Slot 9 Control Register (C1MCTL9)/ CAN1 Local Mask Register B Standard ID1 (C1LMBR1)	349/ 343

Address	Register	Page
02BA$_{16}$	CAN1 Message Slot 10 Control Register (C1MCTL10)/ CAN1 Local Mask Register B Extended ID0 (C1LMBR2)	349/ 344
02BB$_{16}$	CAN1 Message Slot 11 Control Register (C1MCTL11)/ CAN1 Local Mask Register B Extended ID1 (C1LMBR3)	349/ 345
02BC$_{16}$	CAN1 Message Slot 12 Control Register (C1MCTL12)/ CAN1 Local Mask Register B Extended ID2 (C1LMBR4)	349/ 346
02BD$_{16}$	CAN1 Message Slot 13 Control Register (C1MCTL13)	349
02BE$_{16}$	CAN1 Message Slot 14 Control Register (C1MCTL14)	
02BF$_{16}$	CAN1 Message Slot 15 Control Register (C1MCTL15)	
02C0$_{16}$ 02C1$_{16}$	X0 Register Y0 Register (X0R,Y0R)	259
02C2$_{16}$ 02C3$_{16}$	X1 Register Y1 Register (X1R,Y1R)	
02C4$_{16}$ 02C5$_{16}$	X2 Register Y2 Register (X2R,Y2R)	
02C6$_{16}$ 02C7$_{16}$	X3 Register Y3 Register (X3R,Y3R)	
02C8$_{16}$ 02C9$_{16}$	X4 Register Y4 Register (X4R,Y4R)	
02CA$_{16}$ 02CB$_{16}$	X5 Register Y5 Register (X5R,Y5R)	
02CC$_{16}$ 02CD$_{16}$	X6 Register Y6 Register (X6R,Y6R)	
02CE$_{16}$ 02CF$_{16}$	X7 Register Y7 Register (X7R,Y7R)	
02D0$_{16}$ 02D1$_{16}$	X8 Register Y8 Register (X8R,Y8R)	
02D2$_{16}$ 02D3$_{16}$	X9 Register Y9 Register (X9R,Y9R)	
02D4$_{16}$ 02D5$_{16}$	X10 Register Y10 Register (X10R,Y10R)	
02D6$_{16}$ 02D7$_{16}$	X11 Register Y11 Register (X11R,Y11R)	
02D8$_{16}$ 02D9$_{16}$	X12 Register Y12 Register (X12R,Y12R)	
02DA$_{16}$ 02DB$_{16}$	X13 Register Y13 Register (X13R,Y13R)	
02DC$_{16}$ 02DD$_{16}$	X14 Register Y14 Register (X14R,Y14R)	
02DE$_{16}$ 02DF$_{16}$	X15 Register Y15 Register (X15R,Y15R)	

Blank spaces are reserved. No access is allowed.

Quick Reference by Address

Address	Register	Page
02E0$_{16}$	X/Y Control Register (XYC)	259
02E1$_{16}$		
02E2$_{16}$		
02E3$_{16}$		
02E4$_{16}$	UART1 Special Mode Register 4 (U1SMR4)	191
02E5$_{16}$	UART1 Special Mode Register 3 (U1SMR3)	190
02E6$_{16}$	UART1 Special Mode Register 2 (U1SMR2)	189
02E7$_{16}$	UART1 Special Mode Register (U1SMR)	188
02E8$_{16}$	UART1 Transmit/Receive Mode Register (U1MR)	186
02E9$_{16}$	UART1 Bit Rate Register (U1BRG)	
02EA$_{16}$	UART1 Transmit Buffer Register (U1TB)	185
02EB$_{16}$		
02EC$_{16}$	UART1 Transmit/Receive Control Register 0 (U1C0)	187
02ED$_{16}$	UART1 Transmit/Receive Control Register 1 (U1C1)	188
02EE$_{16}$	UART1 Receive Buffer Register (U1RB)	185
02EF$_{16}$		
02F0$_{16}$		
02F1$_{16}$		
02F2$_{16}$		
02F3$_{16}$		
02F4$_{16}$	UART4 Special Mode Register 4 (U4SMR4)	191
02F5$_{16}$	UART4 Special Mode Register 3 (U4SMR3)	190
02F6$_{16}$	UART4 Special Mode Register 2 (U4SMR2)	189
02F7$_{16}$	UART4 Special Mode Register (U4SMR)	188
02F8$_{16}$	UART4 Transmit/Receive Mode Register (U4MR)	186
02F9$_{16}$	UART4 Bit Rate Register (U4BRG)	
02FA$_{16}$	UART4 Transmit Buffer Register (U4TB)	185
02FB$_{16}$		
02FC$_{16}$	UART4 Transmit/Receive Control Register 0 (U4C0)	187
02FD$_{16}$	UART4 Transmit/Receive Control Register 1 (U4C1)	188
02FE$_{16}$	UART4 Receive Buffer Register (U4RB)	185
02FF$_{16}$		
0300$_{16}$	Timer B3,B4,B5 Count Start Flag (TBSR)	165
0301$_{16}$		
0302$_{16}$	Timer A1-1 Register (TA11)	178
0303$_{16}$		
0304$_{16}$	Timer A2-1 Register (TA21)	
0305$_{16}$		
0306$_{16}$	Timer A4-1 Register (TA41)	
0307$_{16}$		
0308$_{16}$	Three-Phase PWM Control Register 0 (INVC0)	175
0309$_{16}$	Three-Phase PWM Control Register 1 (INVC1)	176
030A$_{16}$	Three-Phase Output Buffer Register 0 (IDB0)	177
030B$_{16}$	Three-Phase Output Buffer Register 1 (IDB1)	
030C$_{16}$	Dead Time Timer (DTT)	177
030D$_{16}$	Timer B2 Interrupt Generating Frequency Set Counter (ICTB2)	178
030E$_{16}$		
030F$_{16}$		

Address	Register	Page
0310$_{16}$	Timer B3 Register (TB3)	163
0311$_{16}$		
0312$_{16}$	Timer B4 Register (TB4)	
0313$_{16}$		
0314$_{16}$	Timer B5 Register (TB5)	
0315$_{16}$		
0316$_{16}$		
0317$_{16}$		
0318$_{16}$		
0319$_{16}$		
031A$_{16}$		
031B$_{16}$	Timer B3 Mode Register (TB3MR)	164
031C$_{16}$	Timer B4 Mode Register (TB4MR)	
031D$_{16}$	Timer B5 Mode Register (TB5MR)	
031E$_{16}$		
031F$_{16}$	External Interrupt Request Source Select Register (IFSR)	114
0320$_{16}$		
0321$_{16}$		
0322$_{16}$		
0323$_{16}$		
0324$_{16}$	UART3 Special Mode Register 4 (U3SMR4)	191
0325$_{16}$	UART3 Special Mode Register 3 (U3SMR3)	190
0326$_{16}$	UART3 Special Mode Register 2 (U3SMR2)	189
0327$_{16}$	UART3 Special Mode Register (U3SMR)	188
0328$_{16}$	UART3 Transmit/Receive Mode Register (U3MR)	186
0329$_{16}$	UART3 Bit Rate Register (U3BRG)	
032A$_{16}$	UART3 Transmit Buffer Register (U3TB)	185
032B$_{16}$		
032C$_{16}$	UART3 Transmit/Receive Control Register 0 (U3C0)	187
032D$_{16}$	UART3 Transmit/Receive Control Register 1 (U3C1)	188
032E$_{16}$	UART3 Receive Buffer Register (U3RB)	185
032F$_{16}$		
0330$_{16}$		
0331$_{16}$		
0332$_{16}$		
0333$_{16}$		
0334$_{16}$	UART2 Special Mode Register 4 (U2SMR4)	191
0335$_{16}$	UART2 Special Mode Register 3 (U2SMR3)	190
0336$_{16}$	UART2 Special Mode Register 2 (U2SMR2)	189
0337$_{16}$	UART2 Special Mode Register (U2SMR)	188
0338$_{16}$	UART2 Transmit/Receive Mode Register (U2MR)	186
0339$_{16}$	UART2 Bit Rate Register (U2BRG)	
033A$_{16}$	UART2 Transmit Buffer Register (U2TB)	185
033B$_{16}$		
033C$_{16}$	UART2 Transmit/Receive Control Register 0 (U2C0)	187
033D$_{16}$	UART2 Transmit/Receive Control Register 1 (U2C1)	188
033E$_{16}$	UART2 Receive Buffer Register (U2RB)	185
033F$_{16}$		

Blank spaces are reserved. No access is allowed.

Quick Reference by Address

Address	Register	Page
0340$_{16}$	Count Start Flag (TABSR)	148
0341$_{16}$	Clock Prescaler Reset Flag (CPSRF)	77
0342$_{16}$	One-Shot Start Flag (ONSF)	149
0343$_{16}$	Trigger Select Register (TRGSR)	150
0344$_{16}$	Up/Down Flag (UDF)	149
0345$_{16}$		
0346$_{16}$	Timer A0 Register (TA0)	147
0347$_{16}$		
0348$_{16}$	Timer A1 Register (TA1)	
0349$_{16}$		
034A$_{16}$	Timer A2 Register (TA2)	
034B$_{16}$		
034C$_{16}$	Timer A3 Register (TA3)	
034D$_{16}$		
034E$_{16}$	Timer A4 Register (TA4)	
034F$_{16}$		
0350$_{16}$	Timer B0 Register (TB0)	163
0351$_{16}$		
0352$_{16}$	Timer B1 Register (TB1)	
0353$_{16}$		
0354$_{16}$	Timer B2 Register (TB2)	
0355$_{16}$		
0356$_{16}$	Timer A0 Mode Register (TA0MR)	148
0357$_{16}$	Timer A1 Mode Register (TA1MR)	
0358$_{16}$	Timer A2 Mode Register (TA2MR)	
0359$_{16}$	Timer A3 Mode Register (TA3MR)	
035A$_{16}$	Timer A4 Mode Register (TA4MR)	
035B$_{16}$	Timer B0 Mode Register (TB0MR)	164
035C$_{16}$	Timer B1 Mode Register (TB1MR)	
035D$_{16}$	Timer B2 Mode Register (TB2MR)	
035E$_{16}$	Timer B2 Special Mode Register (TB2SC)	178
035F$_{16}$	Count Source Prescaler Register (TCSPR)	77
0360$_{16}$		
0361$_{16}$		
0362$_{16}$		
0363$_{16}$		
0364$_{16}$	UART0 Special Mode Register 4 (U0SMR4)	191
0365$_{16}$	UART0 Special Mode Register 3 (U0SMR3)	190
0366$_{16}$	UART0 Special Mode Register 2 (U0SMR2)	189
0367$_{16}$	UART0 Special Mode Register (U0SMR)	188
0368$_{16}$	UART0 Transmit/Receive Mode Register (U0MR)	186
0369$_{16}$	UART0 Bit Rate Register (U0BRG)	
036A$_{16}$	UART0 Transmit Buffer Register (U0TB)	185
036B$_{16}$		
036C$_{16}$	UART0 Transmit/Receive Control Register 0 (U0C0)	187
036D$_{16}$	UART0 Transmit/Receive Control Register 1 (U0C1)	188
036E$_{16}$	UART0 Receive Buffer Register (U0RB)	185
036F$_{16}$		

Address	Register	Page
0370$_{16}$		
0371$_{16}$		
0372$_{16}$		
0373$_{16}$		
0374$_{16}$		
0375$_{16}$		
0376$_{16}$		
0377$_{16}$		
0378$_{16}$	DMA0 Request Source Select Register (DM0SL)	127
0379$_{16}$	DMA1 Request Source Select Register (DM1SL)	
037A$_{16}$	DMA2 Request Source Select Register (DM2SL)	
037B$_{16}$	DMA3 Request Source Select Register (DM3SL)	
037C$_{16}$	CRC Data Register (CRCD)	257
037D$_{16}$		
037E$_{16}$	CRC Input Register (CRCIN)	
037F$_{16}$		
0380$_{16}$	A/D0 Register0 (AD00)	242
0381$_{16}$		
0382$_{16}$	A/D0 Register1 (AD01)	
0383$_{16}$		
0384$_{16}$	A/D0 Register2 (AD02)	
0385$_{16}$		
0386$_{16}$	A/D0 Register3 (AD03)	
0387$_{16}$		
0388$_{16}$	A/D0 Register4 (AD04)	
0389$_{16}$		
038A$_{16}$	A/D0 Register5 (AD05)	
038B$_{16}$		
038C$_{16}$	A/D0 Register6 (AD06)	
038D$_{16}$		
038E$_{16}$	A/D0 Register7 (AD07)	
038F$_{16}$		
0390$_{16}$		
0391$_{16}$		
0392$_{16}$	A/D0 Control Register 4 (AD0CON4)	242
0393$_{16}$		
0394$_{16}$	A/D0 Control Register 2 (AD0CON2)	240
0395$_{16}$	A/D0 Control Register 3 (AD0CON3)	241
0396$_{16}$	A/D0 Control Register 0 (AD0CON0)	238
0397$_{16}$	A/D0 Control Register 1 (AD0CON1)	239
0398$_{16}$	D/A Register 0 (DA0)	256
0399$_{16}$		
039A$_{16}$	D/A Register 1 (DA1)	256
039B$_{16}$		
039C$_{16}$	D/A Control Register (DACON)	256
039D$_{16}$		
039E$_{16}$		
039F$_{16}$		

Blank spaces are reserved. No access is allowed.

Quick Reference by Address

Address	Register	Page
03A0$_{16}$	Function Select Register A8 (PS8)	376
03A1$_{16}$	Function Select Register A9 (PS9)	377
03A2$_{16}$		
03A3$_{16}$		
03A4$_{16}$		
03A5$_{16}$		
03A6$_{16}$		
03A7$_{16}$	Function Select Register D1 (PSD1)	382
03A8$_{16}$		
03A9$_{16}$		
03AA$_{16}$		
03AB$_{16}$		
03AC$_{16}$	Function Select Register C2 (PSC2)	381
03AD$_{16}$	Function Select Register C3 (PSC3)	382
03AE$_{16}$		
03AF$_{16}$	Function Select Register C (PSC)	381
03B0$_{16}$	Function Select Register A0 (PS0)	373
03B1$_{16}$	Function Select Register A1 (PS1)	
03B2$_{16}$	Function Select Register B0 (PSL0)	378
03B3$_{16}$	Function Select Register B1 (PSL1)	
03B4$_{16}$	Function Select Register A2 (PS2)	374
03B5$_{16}$	Function Select Register A3 (PS3)	
03B6$_{16}$	Function Select Register B2 (PSL2)	379
03B7$_{16}$	Function Select Register B3 (PSL3)	
03B8$_{16}$		
03B9$_{16}$	Function Select Register A5 (PS5)	375
03BA$_{16}$		
03BB$_{16}$		
03BC$_{16}$	Function Select Register A6 (PS6)	375
03B5$_{16}$	Function Select Register A7 (PS7)	376
03B6$_{16}$	Function Select Register B6 (PSL6)	380
03B7$_{16}$	Function Select Register B7 (PSL7)	
03C0$_{16}$	Port P6 Register (P6)	372
03C1$_{16}$	Port P7 Register (P7)	
03C2$_{16}$	Port P6 Direction Register (PD6)	371
03C3$_{16}$	Port P7 Direction Register (PD7)	
03C4$_{16}$	Port P8 Register (P8)	372
03C5$_{16}$	Port P9 Register (P9)	
03C6$_{16}$	Port P8 Direction Register (PD8)	371
03C7$_{16}$	Port P9 Direction Register (PD9)	
03C8$_{16}$	Port P10 Register (P10)	372
03C9$_{16}$	Port P11 Register (P11)	
03CA$_{16}$	Port P10 Direction Register (PD10)	371
03CB$_{16}$	Port P11 Direction Register(PD11)	
03CC$_{16}$	Port P12 Register (P12)	372
03CD$_{16}$	Port P13 Register (P13)	
03CE$_{16}$	Port P12 Direction Register (PD12)	371
03CF$_{16}$	Port P13 Direction Register (PD13)	

Address	Register	Page
03D0$_{16}$	Port P14 Register (P14)	372
03D1$_{16}$	Port P15 Register (P15)	
03D2$_{16}$	Port P14 Direction Register (PD14)	371
03D3$_{16}$	Port P15 Direction Register (PD15)	
03D4$_{16}$		
03D5$_{16}$		
03D6$_{16}$		
03D7$_{16}$		
03D8$_{16}$		
03D9$_{16}$		
03DA$_{16}$	Pull-Up Control Register 2 (PUR2)	383
03DB$_{16}$	Pull-Up Control Register 3 (PUR3)	384
03DC$_{16}$	Pull-Up Control Register 4 (PUR4)	
03DD$_{16}$		
03DE$_{16}$	Output Port Select Register (OPS)	311
03DF$_{16}$		
03E0$_{16}$	Port P14 Register (P0)	372
03E1$_{16}$	Port P14 Register (P1)	
03E2$_{16}$	Port P14 Direction Register (PD0)	371
03E3$_{16}$	Port P14 Direction Register (PD1)	
03E4$_{16}$	Port P14 Register (P2)	372
03E5$_{16}$	Port P14 Register (P3)	
03E6$_{16}$	Port P14 Direction Register (PD2)	371
03E7$_{16}$	Port P14 Direction Register (PD3)	
03E8$_{16}$	Port P14 Register (P4)	372
03E9$_{16}$	Port P14 Register (P5)	
03EA$_{16}$	Port P14 Direction Register (PD4)	371
03EB$_{16}$	Port P14 Direction Register (PD5)	
03EC$_{16}$		
03ED$_{16}$		
03EE$_{16}$		
03EF$_{16}$		
03F0$_{16}$	Pull-up Control Register 0 (PUR0)	383
03F1$_{16}$	Pull-up Control Register 1 (PUR1)	
03F2$_{16}$		
03F3$_{16}$		
03F4$_{16}$		
03F5$_{16}$		
03F6$_{16}$		
03F7$_{16}$		
03F8$_{16}$		
03F9$_{16}$		
03FA$_{16}$		
03FB$_{16}$		
03FC$_{16}$		
03FD$_{16}$		
03FE$_{16}$		
03FF$_{16}$	Port Control Register (PCR)	385

Blank spaces are reserved. No access is allowed.

1. Overview

The M32C/86 group (M32C/86, M32C/86T) microcomputer is a single-chip control unit that utilizes high-performance silicon gate CMOS technology with the M32C/80 series CPU core. The M32C/86 group (M32C/86, M32C/86T) is available in 144-pin plastic molded LQFP package.

With a 16-Mbyte address space, this microcomputer combines advanced instruction manipulation capabilities to process complex instructions by less bytes and execute instructions at higher speed.

It includes a multiplier and DMAC adequate for office automation, communication devices and industrial equipments, and other high-speed processing applications.

1.1 Applications

Automobiles, audio, cameras, office equipment, communications equipment, portable equipment, etc.

1.2 Performance Overview

Table 1.1 lists performance overview of the M32C/86 group (M32C/86, M32C/86T).

Table 1.1 M32C/86 Group (M32C/86, M32C/86T) Performance

<table>
<tr><th colspan="2" rowspan="2">Characteristic</th><th colspan="2">Performance</th></tr>
<tr><th>M32C/86</th><th>M32C/86T</th></tr>
<tr><td rowspan="5">CPU</td><td>Basic Instructions</td><td colspan="2">108 instructions</td></tr>
<tr><td>Minimum Instruction Execution Time</td><td colspan="2">31.3 ns (f(BCLK)=32 MHz, V_{CC}=4.2 V to 5.5 V)</td></tr>
<tr><td>Operating Mode</td><td>Single-chip mode, Memory expansion mode and Microprocessor mode</td><td>Single-chip mode</td></tr>
<tr><td>Address Space</td><td colspan="2">16 Mbytes</td></tr>
<tr><td>Memory Capacity</td><td colspan="2">See Table 1.2</td></tr>
<tr><td rowspan="18">Peripheral Function</td><td>I/O Port</td><td colspan="2">123 I/O pins and 1 input pin</td></tr>
<tr><td>Multifunction Timer</td><td colspan="2">Timer A: 16 bits x 5 channels, Timer B: 16 bits x 6 channels
Three-phase motor control circuit</td></tr>
<tr><td>Intelligent I/O</td><td colspan="2">Time measurement function or Waveform generating function:
16 bits x 8 channels
Communication function (Clock synchronous serial I/O, Clock asynchronous serial I/O, HDLC data processing)
Stepping Motor Control Function</td></tr>
<tr><td>Serial I/O</td><td colspan="2">5 Channels
Clock synchronous serial I/O, Clock asynchronous serial I/O, IEBus[1], I^2C bus[2]</td></tr>
<tr><td>CAN Module</td><td colspan="2">2 channels Supporting CAN 2.0B specification</td></tr>
<tr><td>A/D Converter</td><td colspan="2">10-bit A/D converter: 1 circuit, 34 channels</td></tr>
<tr><td>D/A Converter</td><td colspan="2">8 bits x 2 channels</td></tr>
<tr><td>DMAC</td><td colspan="2">4 channels</td></tr>
<tr><td>DMAC II</td><td colspan="2">Can be activated by all peripheral function interrupt sources
Immediate transfer, Calculation transfer and Chain transfer functions</td></tr>
<tr><td>CRC Calculation Circuit</td><td colspan="2">CRC-CCITT</td></tr>
<tr><td>X/Y Converter</td><td colspan="2">16 bits x 16 bits</td></tr>
<tr><td>Watchdog Timer</td><td colspan="2">15 bits x 1 channel (with prescaler)</td></tr>
<tr><td>Interrupt</td><td colspan="2">39 internal and 8 external sources, 5 software sources
Interrupt priority level: 7</td></tr>
<tr><td>Clock Generation Circuit</td><td colspan="2">4 circuits
Main clock oscillation circuit(*), Sub clock oscillation circuit(*), On-chip oscillator, PLL frequency synthesizer
(*)Equipped with a built-in feedback resistor. Ceramic resonator or crystal oscillator must be connected externally</td></tr>
<tr><td>Oscillation Stop Detect Function</td><td colspan="2">Main clock oscillation stop detect function</td></tr>
<tr><td>Voltage Detection Circuit</td><td>Available (optional)</td><td>Not available[3]</td></tr>
<tr><td rowspan="2">Electrical Characteristics</td><td>Supply Voltage</td><td colspan="2">V_{CC}=4.2 V to 5.5 V (f(BCLK)=32 MHz)</td></tr>
<tr><td>Power Consumption</td><td colspan="2">28 mA (V_{CC}=5 V, f(BCLK)=32 MHz)
10 µA (V_{CC}=5 V, f(BCLK)=32 kHz, in wait mode)</td></tr>
<tr><td rowspan="2">Flash Memory</td><td>Program/Erase Supply Voltage</td><td colspan="2">5.0 V ± 0.5 V</td></tr>
<tr><td>Program and Erase Endurance</td><td colspan="2">100 times (all space)</td></tr>
<tr><td colspan="2">Operating Ambient Temperature</td><td>–20 to 85°C
–40 to 85°C (optional)</td><td>–40 to 85°C (T version)</td></tr>
<tr><td colspan="2">Package</td><td colspan="2">144-pin plastic molded LQFP</td></tr>
</table>

NOTES:

1. IEBus is a trademark of NEC Electronics Corporation.
2. I^2C bus is a trademark of Koninklijke Philips Electronics N. V.
3. The cold start-up/warm start-up determine function is available only at the user's option.

All options are on a request basis.

1.3 Block Diagram

Figure 1.1 shows a block diagram of the M32C/86 group (M32C/86, M32C/86T) microcomputer.

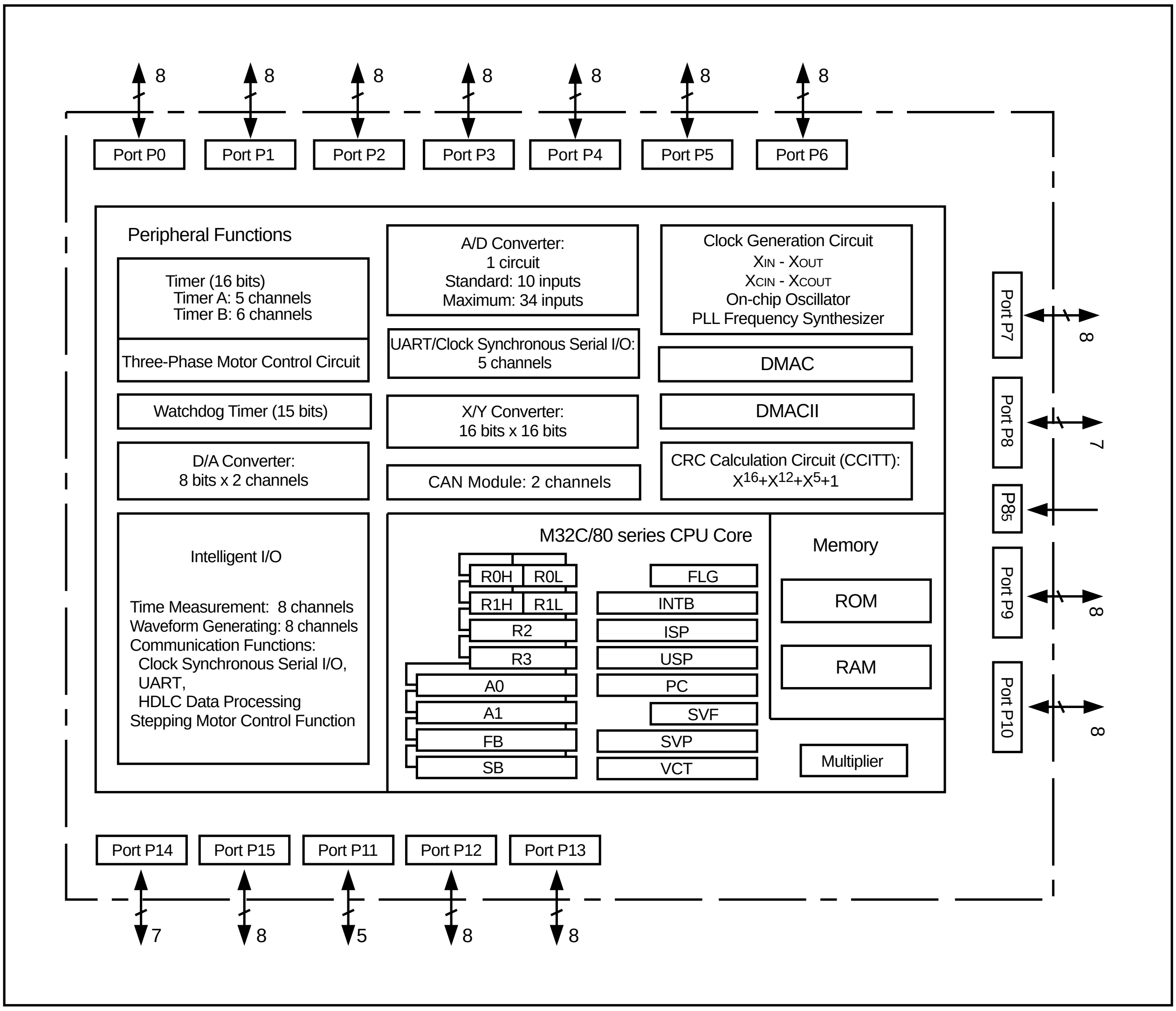


Figure 1.1 M32C/86 Group (M32C/86, M32C/86T) Block Diagram

1.4 Product Information

Table 1.2 lists the product information. Figure 1.2 shows the product numbering system.

Table 1.2 M32C/86 Group (1) (M32C/86) **As of September, 2005**

Type Number	Package Type	ROM Capacity	RAM Capacity	Remarks
M30865FJGP	PLQP0144KA-A (144P6Q-A)	512K+4K	24K	Flash Memory

Table 1.2 M32C/86 Group (2) (T Version, M32C/86T) **As of September, 2005**

Type Number	Package Type	ROM Capacity	RAM Capacity	Remarks
M30865FJTGP	PLQP0144KA-A (144P6Q-A)	512K+4K	24K	Flash Memory

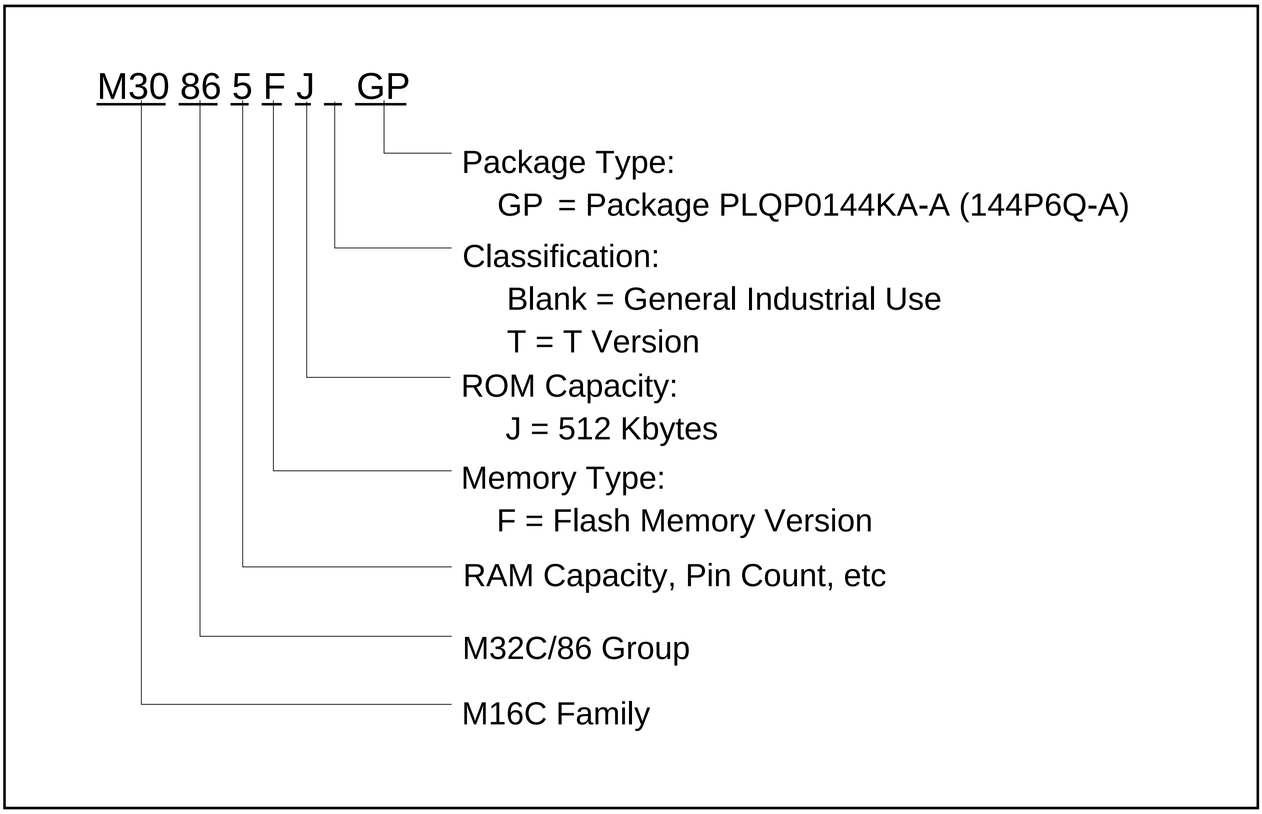
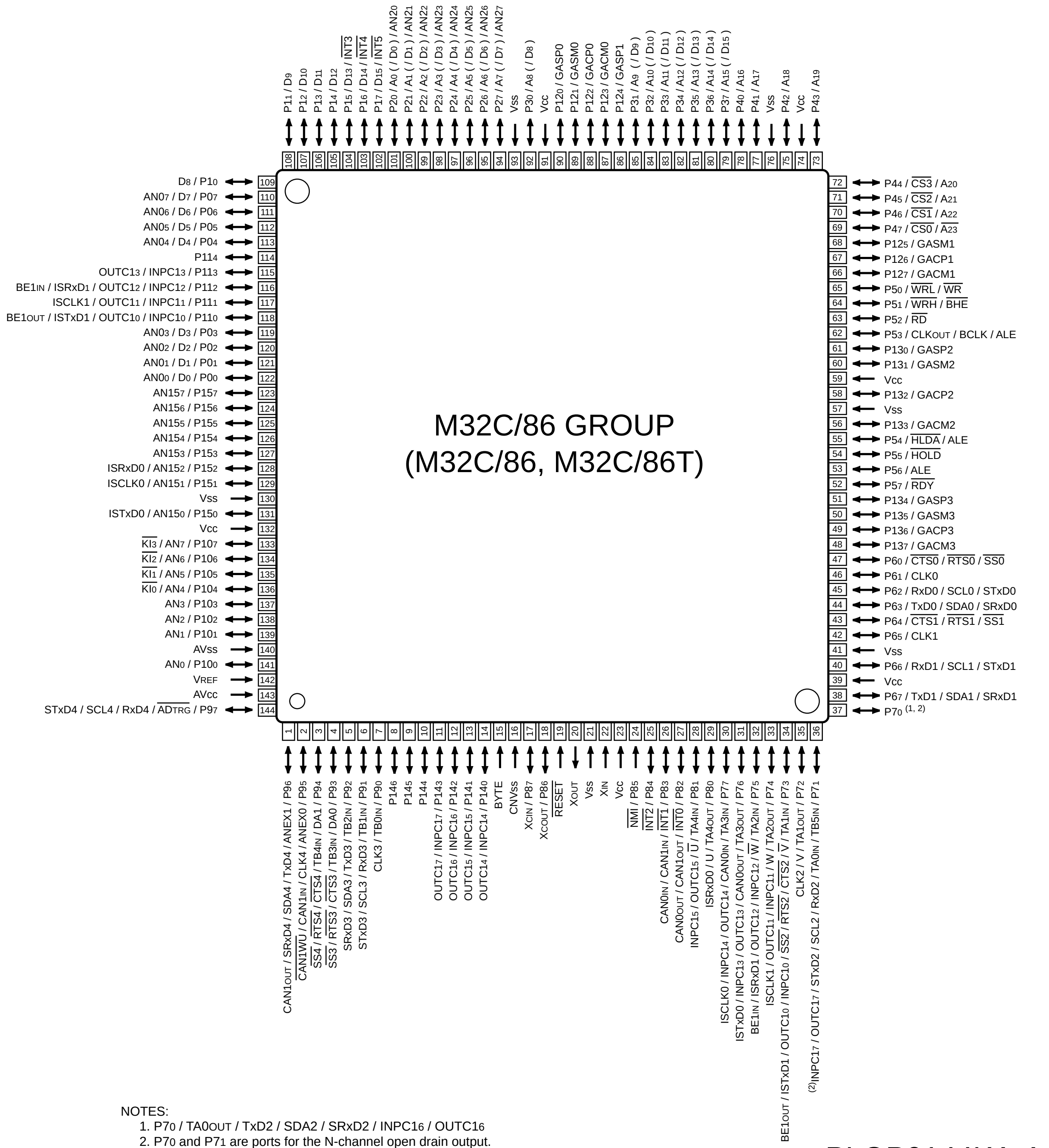


Figure 1.2 Product Numbering System

1.5 Pin Assignment

Figures 1.3 shows the pin assignment (top view).


108 P11 / D9
107 P12 / D10
106 P13 / D11
105 P14 / D12
104 P15 / D13 / INT3
103 P16 / D14 / INT4
102 P17 / D15 / INT5
101 P20 / A0 (/ D0) / AN20
100 P21 / A1 (/ D1) / AN21
99 P22 / A2 (/ D2) / AN22
98 P23 / A3 (/ D3) / AN23
97 P24 / A4 (/ D4) / AN24
96 P25 / A5 (/ D5) / AN25
95 P26 / A6 (/ D6) / AN26
94 P27 / A7 (/ D7) / AN27
93 Vss
92 P30 / A8 (/ D8)
91 Vcc
90 P120 / GASP0
89 P121 / GASM0
88 P122 / GACP0
87 P123 / GACM0
86 P124 / GASP1
85 P31 / A9 (/ D9)
84 P32 / A10 (/ D10)
83 P33 / A11 (/ D11)
82 P34 / A12 (/ D12)
81 P35 / A13 (/ D13)
80 P36 / A14 (/ D14)
79 P37 / A15 (/ D15)
78 P40 / A16
77 P41 / A17
76 Vss
75 P42 / A18
74 Vcc
73 P43 / A19
D8 / P10 109
AN07 / D7 / P07 110
AN06 / D6 / P06 111
AN05 / D5 / P05 112
AN04 / D4 / P04 113
P114 114
OUTC13 / INPC13 / P113 115
BE1IN / ISRxD1 / OUTC12 / INPC12 / P112 116
ISCLK1 / OUTC11 / INPC11 / P111 117
BE1OUT / ISTxD1 / OUTC10 / INPC10 / P110 118
AN03 / D3 / P03 119
AN02 / D2 / P02 120
AN01 / D1 / P01 121
AN00 / D0 / P00 122
AN157 / P157 123
AN156 / P156 124
AN155 / P155 125
AN154 / P154 126
AN153 / P153 127
ISRxD0 / AN152 / P152 128
ISCLK0 / AN151 / P151 129
Vss 130
ISTxD0 / AN150 / P150 131
Vcc 132
KI3 / AN7 / P107 133
KI2 / AN6 / P106 134
KI1 / AN5 / P105 135
KI0 / AN4 / P104 136
AN3 / P103 137
AN2 / P102 138
AN1 / P101 139
AVss 140
AN0 / P100 141
VREF 142
AVcc 143
STxD4 / SCL4 / RxD4 / ADTRG / P97 144
M32C/86 GROUP
(M32C/86, M32C/86T)
72 P44 / CS3 / A20
71 P45 / CS2 / A21
70 P46 / CS1 / A22
69 P47 / CS0 / A23
68 P125 / GASM1
67 P126 / GACP1
66 P127 / GACM1
65 P50 / WRL / WR
64 P51 / WRH / BHE
63 P52 / RD
62 P53 / CLKOUT / BCLK / ALE
61 P130 / GASP2
60 P131 / GASM2
59 Vcc
58 P132 / GACP2
57 Vss
56 P133 / GACM2
55 P54 / HLDA / ALE
54 P55 / HOLD
53 P56 / ALE
52 P57 / RDY
51 P134 / GASP3
50 P135 / GASM3
49 P136 / GACP3
48 P137 / GACM3
47 P60 / CTS0 / RTS0 / SS0
46 P61 / CLK0
45 P62 / RxD0 / SCL0 / STxD0
44 P63 / TxD0 / SDA0 / SRxD0
43 P64 / CTS1 / RTS1 / SS1
42 P65 / CLK1
41 Vss
40 P66 / RxD1 / SCL1 / STxD1
39 Vcc
38 P67 / TxD1 / SDA1 / SRxD1
37 P70 (1, 2)
1 CAN1OUT / SRxD4 / SDA4 / TxD4 / ANEX1 / P96
2 CAN1WU / CAN1IN / CLK4 / ANEX0 / P95
3 SS4 / RTS4 / CTS4 / TB4IN / DA1 / P94
4 SS3 / RTS3 / CTS3 / TB3IN / DA0 / P93
5 SRxD3 / SDA3 / TxD3 / TB2IN / P92
6 STxD3 / SCL3 / RxD3 / TB1IN / P91
7 CLK3 / TB0IN / P90
8 P146
9 P145
10 P144
11 OUTC17 / INPC17 / P143
12 OUTC16 / INPC16 / P142
13 OUTC15 / INPC15 / P141
14 OUTC14 / INPC14 / P140
15 BYTE
16 CNVss
17 XCIN / P87
18 XCOUT / P86
19 RESET
20 XOUT
21 Vss
22 XIN
23 Vcc
24 NMI / P85
25 INT2 / P84
26 CAN0IN / CAN1IN / INT1 / P83
27 CAN0OUT / CAN1OUT / INT0 / P82
28 INPC15 / OUTC15 / U / TA4IN / P81
29 ISRxD0 / U / TA4OUT / P80
30 ISCLK0 / INPC14 / OUTC14 / CAN0IN / TA3IN / P77
31 ISTxD0 / INPC13 / OUTC13 / CAN0OUT / TA3OUT / P76
32 BE1IN / ISRxD1 / OUTC12 / INPC12 / W / TA2IN / P75
33 ISCLK1 / OUTC11 / INPC11 / W / TA2OUT / P74
34 BE1OUT / ISTxD1 / OUTC10 / INPC10 / SS2 / RTS2 / CTS2 / V / TA1IN / P73
35 CLK2 / V / TA1OUT / P72
36 (2)INPC17 / OUTC17 / STxD2 / SCL2 / RxD2 / TA0IN / TB5IN / P71
PLQP0144KA-A
(144P6Q-A)

NOTES:
1. P70 / TA0OUT / TxD2 / SDA2 / SRxD2 / INPC16 / OUTC16
2. P70 and P71 are ports for the N-channel open drain output.

Figure 1.3 Pin Assignment

Table 1.3 Pin Characteristics

Pin No.	Control Pin	Port	Interrupt Pin	Timer Pin	UART/CAN Pin	Intelligent I/O Pin	Analog Pin	Bus Control Pin(1)
1		P9$_6$			TxD4/SDA4/SRxD4/CAN1$_{OUT}$		ANEX1	
2		P9$_5$			CLK4/CAN1$_{IN}$/$\overline{\text{CAN1WU}}$		ANEX0	
3		P9$_4$		TB4$_{IN}$	$\overline{\text{CTS4}}$/$\overline{\text{RTS4}}$/$\overline{\text{SS4}}$		DA1	
4		P9$_3$		TB3$_{IN}$	$\overline{\text{CTS3}}$/$\overline{\text{RTS3}}$/$\overline{\text{SS3}}$		DA0	
5		P9$_2$		TB2$_{IN}$	TxD3/SDA3/SRxD3			
6		P9$_1$		TB1$_{IN}$	RxD3/SCL3/STxD3			
7		P9$_0$		TB0$_{IN}$	CLK3			
8		P14$_6$						
9		P14$_5$						
10		P14$_4$						
11		P14$_3$				INPC1$_7$/OUTC1$_7$		
12		P14$_2$				INPC1$_6$/OUTC1$_6$		
13		P14$_1$				INPC1$_5$/OUTC1$_5$		
14		P14$_0$				INPC1$_4$/OUTC1$_4$		
15	BYTE							
16	CNV$_{SS}$							
17	X$_{CIN}$	P8$_7$						
18	X$_{COUT}$	P8$_6$						
19	$\overline{\text{RESET}}$							
20	X$_{OUT}$							
21	V$_{SS}$							
22	X$_{IN}$							
23	V$_{CC}$							
24		P8$_5$	$\overline{\text{NMI}}$					
25		P8$_4$	$\overline{\text{INT2}}$					
26		P8$_3$	$\overline{\text{INT1}}$		CAN0$_{IN}$/CAN1$_{IN}$			
27		P8$_2$	$\overline{\text{INT0}}$		CAN0$_{OUT}$/CAN1$_{OUT}$			
28		P8$_1$		TA4$_{IN}$/$\overline{\text{U}}$		INPC1$_5$/OUTC1$_5$		
29		P8$_0$		TA4$_{OUT}$/U		ISRxD0		
30		P7$_7$		TA3$_{IN}$	CAN0$_{IN}$	INPC1$_4$/OUTC1$_4$/ISCLK0		
31		P7$_6$		TA3$_{OUT}$	CAN0$_{OUT}$	INPC1$_3$/OUTC1$_3$/ISTxD0		
32		P7$_5$		TA2$_{IN}$/$\overline{\text{W}}$		INPC1$_2$/OUTC1$_2$/ISRxD1/BE1$_{IN}$		
33		P7$_4$		TA2$_{OUT}$/W		INPC1$_1$/OUTC1$_1$/ISCLK1		
34		P7$_3$		TA1$_{IN}$/$\overline{\text{V}}$	$\overline{\text{CTS2}}$/$\overline{\text{RTS2}}$/$\overline{\text{SS2}}$	INPC1$_0$/OUTC1$_0$/ISTxD1/BE1$_{OUT}$		
35		P7$_2$		TA1$_{OUT}$/V	CLK2			
36		P7$_1$		TB5$_{IN}$/TA0$_{IN}$	RxD2/SCL2/STxD2	INPC1$_7$/OUTC1$_7$		
37		P7$_0$		TA0$_{OUT}$	TxD2/SDA2/SRxD2	INPC1$_6$/OUTC1$_6$		
38		P6$_7$			TxD1/SDA1/SRxD1			
39	V$_{CC}$							
40		P6$_6$			RxD1/SCL1/STxD1			
41	V$_{SS}$							
42		P6$_5$			CLK1			
43		P6$_4$			$\overline{\text{CTS1}}$/$\overline{\text{RTS1}}$/$\overline{\text{SS1}}$			
44		P6$_3$			TxD0/SDA0/SRxD0			
45		P6$_2$			RxD0/SCL0/STxD0			
46		P6$_1$			CLK0			
47		P6$_0$			$\overline{\text{CTS0}}$/$\overline{\text{RTS0}}$/$\overline{\text{SS0}}$			
48		P13$_7$				GACM3		

NOTES:

1. Bus control pins in M32C/86T cannot be used.

Table 1.3 Pin Characteristics (Continued)

Pin No.	Control Pin	Port	Interrupt Pin	Timer Pin	UART/CAN Pin	Intelligent I/O Pin	Analog Pin	Bus Control Pin[1]
49		P13$_6$				GACP3		
50		P13$_5$				GASM3		
51		P13$_4$				GASP3		
52		P5$_7$						$\overline{\text{RDY}}$
53		P5$_6$						ALE
54		P5$_5$						$\overline{\text{HOLD}}$
55		P5$_4$						$\overline{\text{HLDA}}$/ALE
56		P13$_3$				GACM2		
57	V_{SS}							
58		P13$_2$				GACP2		
59	V_{CC}							
60		P13$_1$				GASM2		
61		P13$_0$				GASP2		
62		P5$_3$						CLK$_{OUT}$/BCLK/ALE
63		P5$_2$						$\overline{\text{RD}}$
64		P5$_1$						$\overline{\text{WRH}}$/$\overline{\text{BHE}}$
65		P5$_0$						$\overline{\text{WRL}}$/$\overline{\text{WR}}$
66		P12$_7$				GACM1		
67		P12$_6$				GACP1		
68		P12$_5$				GASM1		
69		P4$_7$						$\overline{\text{CS0}}$/$\overline{\text{A}_{23}}$
70		P4$_6$						$\overline{\text{CS1}}$/A$_{22}$
71		P4$_5$						$\overline{\text{CS2}}$/A$_{21}$
72		P4$_4$						$\overline{\text{CS3}}$/A$_{20}$
73		P4$_3$						A$_{19}$
74	V_{CC}							
75		P4$_2$						A$_{18}$
76	V_{SS}							
77		P4$_1$						A$_{17}$
78		P4$_0$						A$_{16}$
79		P3$_7$						A$_{15}$(/D$_{15}$)
80		P3$_6$						A$_{14}$(/D$_{14}$)
81		P3$_5$						A$_{13}$(/D$_{13}$)
82		P3$_4$						A$_{12}$(/D$_{12}$)
83		P3$_3$						A$_{11}$(/D$_{11}$)
84		P3$_2$						A$_{10}$(/D$_{10}$)
85		P3$_1$						A$_9$(/D$_9$)
86		P12$_4$				GASP1		
87		P12$_3$				GACM0		
88		P12$_2$				GACP0		
89		P12$_1$				GASM0		
90		P12$_0$				GASP0		
91	V_{CC}							
92		P3$_0$						A$_8$(/D$_8$)
93	V_{SS}							
94		P2$_7$					AN2$_7$	A$_7$(/D$_7$)
95		P2$_6$					AN2$_6$	A$_6$(/D$_6$)
96		P2$_5$					AN2$_5$	A$_5$(/D$_5$)

NOTES:

1. Bus control pins in M32C/86T cannot be used.

Table 1.3 Pin Characteristics (Continued)

Pin No.	Control Pin	Port	Interrupt Pin	Timer Pin	UART/CAN Pin	Intelligent I/O Pin	Analog Pin	Bus Control Pin(1)
97		$P2_4$					$AN2_4$	A_4(/D_4)
98		$P2_3$					$AN2_3$	A_3(/D_3)
99		$P2_2$					$AN2_2$	A_2(/D_2)
100		$P2_1$					$AN2_1$	A_1(/D_1)
101		$P2_0$					$AN2_0$	A_0(/D_0)
102		$P1_7$	$\overline{INT5}$					D_{15}
103		$P1_6$	$\overline{INT4}$					D_{14}
104		$P1_5$	$\overline{INT3}$					D_{13}
105		$P1_4$						D_{12}
106		$P1_3$						D_{11}
107		$P1_2$						D_{10}
108		$P1_1$						D_9
109		$P1_0$						D_8
110		$P0_7$					$AN0_7$	D_7
111		$P0_6$					$AN0_6$	D_6
112		$P0_5$					$AN0_5$	D_5
113		$P0_4$					$AN0_4$	D_4
114		$P11_4$						
115		$P11_3$				$INPC1_3$/$OUTC1_3$		
116		$P11_2$				$INPC1_2$/$OUTC1_2$/ISRxD1/$BE1_{IN}$		
117		$P11_1$				$INPC1_1$/$OUTC1_1$/ISCLK1		
118		$P11_0$				$INPC1_0$/$OUTC1_0$/ISTxD1/$BE1_{OUT}$		
119		$P0_3$					$AN0_3$	D_3
120		$P0_2$					$AN0_2$	D_2
121		$P0_1$					$AN0_1$	D_1
122		$P0_0$					$AN0_0$	D_0
123		$P15_7$					$AN15_7$	
124		$P15_6$					$AN15_6$	
125		$P15_5$					$AN15_5$	
126		$P15_4$					$AN15_4$	
127		$P15_3$					$AN15_3$	
128		$P15_2$				ISRxD0	$AN15_2$	
129		$P15_1$				ISCLK0	$AN15_1$	
130	V_{SS}							
131		$P15_0$				ISTxD0	$AN15_0$	
132	V_{CC}							
133		$P10_7$	$\overline{KI_3}$				AN_7	
134		$P10_6$	$\overline{KI_2}$				AN_6	
135		$P10_5$	$\overline{KI_1}$				AN_5	
136		$P10_4$	$\overline{KI_0}$				AN_4	
137		$P10_3$					AN_3	
138		$P10_2$					AN_2	
139		$P10_1$					AN_1	
140	AV_{SS}							
141		$P10_0$					AN_0	
142	V_{REF}							
143	AV_{CC}							
144		$P9_7$			RxD4/SCL4/STxD4		$\overline{AD_{TRG}}$	

NOTES:

1. Bus control pins in M32C/86T cannot be used,

1.6 Pin Description

Table 1.4 Pin Description

Classsfication	Symbol	I/O Type	Function
Power Supply	V_{CC} V_{SS}	I	Apply 4.2 V to 5.5 V to the V_{CC} pin. Apply 0V to the V_{SS} pin
Analog Power Supply	AV_{CC} AV_{SS}	I	Supplies power to the A/D converter. Connect the AV_{CC} pin to V_{CC} and the AV_{SS} pin to V_{SS}
Reset Input	$\overline{\text{RESET}}$	I	The microcomputer is in a reset state when "L" is applied to the $\overline{\text{RESET}}$ pin
CNV_{SS}	CNV_{SS}	I	Switches processor mode. Connect the CNV_{SS} pin to V_{SS} to start up in single-chip mode or to V_{CC} to start up in microprocessor mode
Input to Switch External Data Bus Width[1]	BYTE	I	Switches data bus width in external memory space 3. The data bus is 16 bits wide when the BYTE pin is held "L" and 8 bits wide when it is held "H". Set to either. Connect the BYTE pin to V_{SS} to use the microcomputer in single-chip mode
Bus Control Pins[1]	D_0 to D_7	I/O	Inputs and outputs data (D_0 to D_7) while accessing an external memory space with separate bus
	D_8 to D_{15}	I/O	Inputs and outputs data (D_8 to D_{15}) while accessing an external memory space with 16-bit separate bus
	A_0 to A_{22}	O	Outputs address bits A_0 to A_{22}
	$\overline{A_{23}}$	O	Outputs inversed address bit A_{23}
	A_0/D_0 to A_7/D_7	I/O	Inputs and outputs data (D_0 to D_7) and outputs 8 low-order address bits (A_0 to A_7) by time-sharing while accessing an external memory space with multiplexed bus
	A_8/D_8 to A_{15}/D_{15}	I/O	Inputs and outputs data (D_8 to D_{15}) and outputs 8 middle-order address bits (A_8 to A_{15}) by time-sharing while accessing an external memory space with 16-bit multiplexed bus
	$\overline{\text{CS0}}$ to $\overline{\text{CS3}}$	O	Outputs $\overline{\text{CS0}}$ to $\overline{\text{CS3}}$ that are chip-select signals specifying an external space
	$\overline{\text{WRL}}$ / $\overline{\text{WR}}$ $\overline{\text{WRH}}$ / $\overline{\text{BHE}}$ $\overline{\text{RD}}$	O	Outputs $\overline{\text{WRL}}$, $\overline{\text{WRH}}$, ($\overline{\text{WR}}$, $\overline{\text{BHE}}$) and $\overline{\text{RD}}$ signals. $\overline{\text{WRL}}$ and $\overline{\text{WRH}}$ can be switched with $\overline{\text{WR}}$ and $\overline{\text{BHE}}$ by program ■ $\overline{\text{WRL}}$, $\overline{\text{WRH}}$ and $\overline{\text{RD}}$ selected: If external data bus is 16 bits wide, data is written to an even address in external memory space when $\overline{\text{WRL}}$ is held "L". Data is written to an odd address when $\overline{\text{WRH}}$ is held "L". Data is read when $\overline{\text{RD}}$ is held "L". ■ $\overline{\text{WR}}$, $\overline{\text{BHE}}$ and $\overline{\text{RD}}$ selected: Data is written to external memory space when $\overline{\text{WR}}$ is held "L". Data in an external memory space is read when $\overline{\text{RD}}$ is held "L". An odd address is accessed when $\overline{\text{BHE}}$ is held "L". Select $\overline{\text{WR}}$, $\overline{\text{BHE}}$ and $\overline{\text{RD}}$ for external 8-bit data bus.
	ALE	O	ALE is a signal latching the address
	$\overline{\text{HOLD}}$	I	The microcomputer is placed in a hold state while the $\overline{\text{HOLD}}$ pin is held "L"
	$\overline{\text{HLDA}}$	O	Outputs an "L" signal while the microcomputer is placed in a hold state
	$\overline{\text{RDY}}$	I	Bus is placed in a wait state while the $\overline{\text{RDY}}$ pin is held "L"

I : Input O : Output I/O : Input and output

NOTES:

1. Bus control pins in M32C/86T cannot be used.

Table 1.4 Pin Description (Continued)

Classsfication	Symbol	I/O Type	Function
Main Clock Input	XIN	I	I/O pins for the main clock oscillation circuit. Connect a ceramic resonator or crystal oscillator between XIN and XOUT. To apply external clock, apply it to XIN and leave XOUT open
Main Clock Output	XOUT	O	
Sub Clock Input	XCIN	I	I/O pins for the sub clock oscillation circuit. Connect a crystal oscillator between XCIN and XCOUT. To apply external clock, apply it to XCIN and leave XCOUT open
Sub Clock Output	XCOUT	O	
BCLK Output[1]	BCLK	O	Outputs BCLK signal
Clock Output	CLKOUT	O	Outputs the clock having the same frequency as fC, f8 or f32
$\overline{\mathrm{INT}}$ Interrupt Input	$\overline{\mathrm{INT0}}$ to $\overline{\mathrm{INT2}}$	I	Input pins for the $\overline{\mathrm{INT}}$ interrupt
	$\overline{\mathrm{INT3}}$ to $\overline{\mathrm{INT5}}$		
$\overline{\mathrm{NMI}}$ Interrupt Input	$\overline{\mathrm{NMI}}$	I	Input pin for the $\overline{\mathrm{NMI}}$ interrupt
Key Input Interrupt	$\overline{\mathrm{KI0}}$ to $\overline{\mathrm{KI3}}$	I	Input pins for the key input interrupt
Timer A	TA0OUT to TA4OUT	I/O	I/O pins for the timer A0 to A4 (TA0OUT is a pin for the N-channel open drain output.)
	TA0IN to TA4IN	I	Input pins for the timer A0 to A4
Timer B	TB0IN to TB5IN	I	Input pins for the timer B0 to B5
Three-phase Motor Control Timer Output	U, $\overline{\mathrm{U}}$, V, $\overline{\mathrm{V}}$, W, $\overline{\mathrm{W}}$	O	Output pins for the three-phase motor control timer
Serial I/O	$\overline{\mathrm{CTS0}}$ to $\overline{\mathrm{CTS4}}$	I	Input pins for data transmission control
	$\overline{\mathrm{RTS0}}$ to $\overline{\mathrm{RTS4}}$	O	Output pins for data reception control
	CLK0 to CLK4	I/O	Inputs and outputs the transfer clock
	RxD0 to RxD4	I	Inputs serial data
	TxD0 to TxD4	O	Outputs serial data (TxD2 is a pin for the N-channel open drain output.)
I^{2}C Mode	SDA0 to SDA4	I/O	Inputs and outputs serial data (SDA2 is a pin for the N-channel open drain output.)
	SCL0 to SCL4		Inputs and outputs the transfer clock (SCL2 is a pin for the N-channel open drain output.)
Serial I/O Special Function	STxD0 to STxD4	O	Outputs serial data when slave mode is selected (STxD2 is a pin for the N-channel open drain output.)
	SRxD0 to SRxD4	I	Inputs serial data when slave mode is selected
	$\overline{\mathrm{SS0}}$ to $\overline{\mathrm{SS4}}$	I	Input pins to control serial I/O special function

I : Input O : Output I/O : Input and output

NOTES:

1. Bus control pins in M32C/86T cannot be used.

Table 1.4 Pin Description (Continued)

Classsfication	Symbol	I/O Type	Function
Reference Voltage Input	VREF	I	Applies reference voltage to the A/D converter and D/A converter
A/D Converter	AN0 to AN7 AN00 to AN07 AN20 to AN27	I	Analog input pins for the A/D converter
	$\overline{\text{ADTRG}}$	I	Input pin for an external A/D trigger
	ANEX0	I/O	Extended analog input pin for the A/D converter and output pin in external op-amp connection mode
	ANEX1	I	Extended analog input pin for the A/D converter
D/A Converter	DA0, DA1	O	Output pin for the D/A converter
Intelligent I/O	INPC10 to INPC13	I	Input pins for the time measurement function
	INPC14 to INPC17		
	OUTC10 to OUTC13	O	Output pins for the waveform generating function (OUTC16 and OUTC17 assigned to P70 and P71 are pins for the N-channel open drain output.)
	OUTC14 to OUTC17		
	ISCLK0	I/O	Inputs and outputs the clock for the intelligent I/O communication function
	ISCLK1		
	ISRxD0	I	Inputs data for the intelligent I/O communication function
	ISRxD1		
	ISTxD0	O	Outputs data for the intelligent I/O communication function
	ISTxD1		
	BE1IN	I	Inputs data for the intelligent I/O communication function
	BE1OUT	O	Outputs data for the intelligent I/O communication function
	GASP0 to GASP3	O	Output pins for the stepping motor control function
	GASM0 to GASM3		
	GACP0 to GACP3		
	GACM0 to GACM3		
CAN	CAN0IN	I	Input pins for the CAN communication function
	CAN1IN		
	CAN0OUT	O	Output pins for the CAN communication function
	CAN1OUT		
	$\overline{\text{CAN1WU}}$	I	Input pin for the CAN1 wake-up interrupt
I/O Ports	P00 to P07 P10 to P17 P20 to P27 P30 to P37 P40 to P47 P50 to P57	I/O	I/O ports for CMOS. Each port can be programmed for input or output under the control of the direction register. An input port can be set, by program, for a pull-up resistor available or for no pull-up resister available in 4-bit units
	P60 to P67 P70 to P77 P90 to P97 P100 to P107	I/O	I/O ports having equivalent functions to P0 (P70 and P71 are ports for the N-channel open drain output.)
	P80 to P84 P86, P87	I/O	I/O ports having equivalent functions to P0
Input Port	P85	I	Shares a pin with $\overline{\text{NMI}}$. $\overline{\text{NMI}}$ input state can be got by reading P85

I : Input O : Output I/O : Input and output

Table 1.4 Pin Description (Continued)

Classsfication	Symbol	I/O Type	Function
A/D Converter	AN15$_0$ to AN15$_7$	I	Analog input pins for the A/D converter
I/O Ports	P11$_0$ to P11$_4$ P12$_0$ to P12$_7$ P13$_0$ to P13$_7$	I/O	I/O ports having equivalent functions to P0
	P14$_0$ to P14$_6$ P15$_0$ to P15$_7$	I/O	I/O ports having equivalent functions to P0

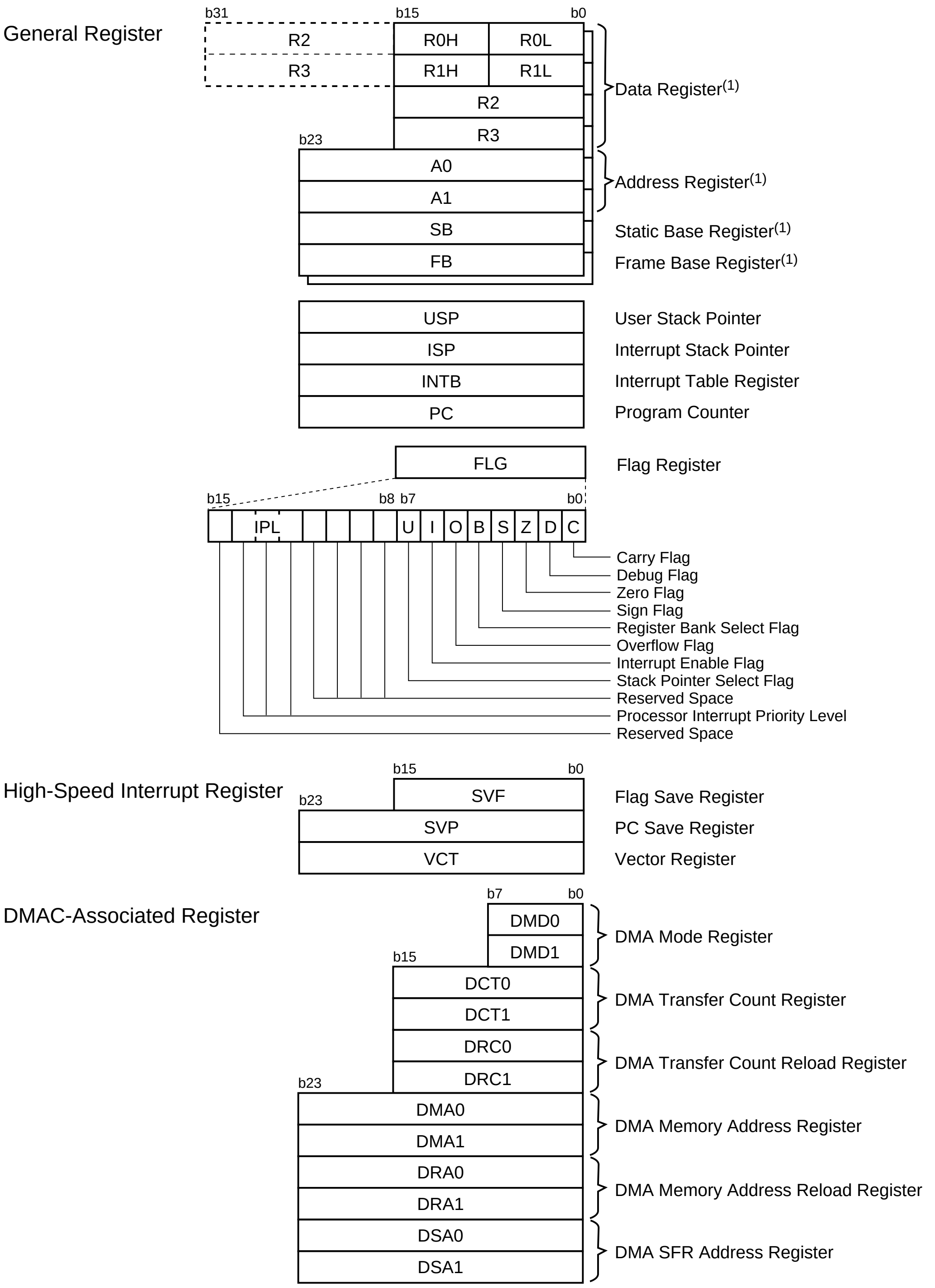
I : Input O : Output I/O : Input and output

2. Central Processing Unit (CPU)

Figure 2.1 shows the CPU registers.

The register bank is comprised of 8 registers (R0, R1, R2, R3, A0, A1, SB and FB) out of 28 CPU registers.

Two sets of register banks are provided.



NOTES:
1. The register bank is comprised of these registers. Two sets of register banks are provided.

Figure 2.1 CPU Register

2.1 General Registers

2.1.1 Data Registers (R0, R1, R2 and R3)

R0, R1, R2 and R3 are 16-bit registers for transfer, arithmetic and logic operations. R0 and R1 can be split into high-order bits (R0H) and low-order bits (R0L) to be used separately as 8-bit data registers. R0 can be combined with R2 to be used as a 32-bit data register (R2R0). The same applies to R1 and R3.

2.1.2 Address Registers (A0 and A1)

A0 and A1 are 24-bit registers for A0-/A1-indirect addressing, A0-/A1-relative addressing, transfer, arithmetic and logic operations.

2.1.3 Static Base Register (SB)

SB is a 24-bit register for SB-relative addressing.

2.1.4 Frame Base Register (FB)

FB is a 24-bit register for FB-relative addressing.

2.1.5 Program Counter (PC)

PC, 24 bits wide, indicates the address of an instruction to be executed.

2.1.6 Interrupt Table Register (INTB)

INTB is a 24-bit register indicating the starting address of an relocatable interrupt vector table.

2.1.7 User Stack Pointer (USP), Interrupt Stack Pointer (ISP)

The stack pointers (SP), USP and ISP, are 24 bits wide each. The U flag is used to switch between USP and ISP. Refer to **2.1.8 Flag Register (FLG)** for details on the U flag. Set USP and ISP to even addresses to execute an interrupt sequence efficiently.

2.1.8 Flag Register (FLG)

FLG is a 16-bit register indicating a CPU state.

2.1.8.1 Carry Flag (C)

The C flag indicates whether carry or borrow has occurred after executing an instruction.

2.1.8.2 Debug Flag (D)

The D flag is for debug only. Set to "0".

2.1.8.3 Zero Flag (Z)

The Z flag is set to "1" when the value of zero is obtained from an arithmetic operation; otherwise "0".

2.1.8.4 Sign Flag (S)

The S flag is set to "1" when a negative value is obtained from an arithmetic operation; otherwise "0".

2.1.8.5 Register Bank Select Flag (B)

The register bank 0 is selected when the B flag is set to "0". The register bank 1 is selected when this flag is set to "1".

2.1.8.6 Overflow Flag (O)

The O flag is set to "1" when the result of an arithmetic operation overflows; otherwise "0".

2.1.8.7 Interrupt Enable Flag (I)

The I flag enables a maskable interrupt.

Interrupt is disabled when the I flag is set to "0" and enabled when the I flag is set to "1". The I flag is set to "0" when an interrupt is acknowledged.

2.1.8.8 Stack Pointer Select Flag (U)

ISP is selected when the U flag is set to "0". USP is selected when this flag is set to "1".

The U flag is set to "0" when a hardware interrupt is acknowledged or the INT instruction of software interrupt numbers 0 to 31 is executed.

2.1.8.9 Processor Interrupt Priority Level (IPL)

IPL, 3 bits wide, assigns processor interrupt priority levels from level 0 to level 7.

If a requested interrupt has greater priority than IPL, the interrupt is enabled.

2.1.8.10 Reserved Space

When writing to a reserved space, set to "0". When reading, its content is indeterminate.

2.2 High-Speed Interrupt Registers

Registers associated with the high-speed interrupt are as follows:

- Flag save register (SVF)
- PC save register (SVP)
- Vector register (VCT)

Refer to **11.4 High-Speed Interrupt** for details.

2.3 DMAC-Associated Registers

Registers associated with DMAC are as follows:

- DMA mode register (DMD0, DMD1)
- DMA transfer count register (DCT0, DCT1)
- DMA transfer count reload register (DRC0, DRC1)
- DMA memory address register (DMA0, DMA1)
- DMA SFR address register (DSA0, DSA1)
- DMA memory address reload register (DRA0, DRA1)

Refer to **13. DMAC** for details.

3. Memory

Figure 3.1 shows a memory map of the M32C/86 group (M32C/86, M32C/86T).

The M32C/86 group (M32C/86, M32C/86T) provides 16-Mbyte address space addressed from 000000_{16} to $FFFFFF_{16}$.

The internal ROM is allocated from address $FFFFFF_{16}$ to lower. For example, a 64-Kbyte internal ROM is addressed from $FF0000_{16}$ to $FFFFFF_{16}$.

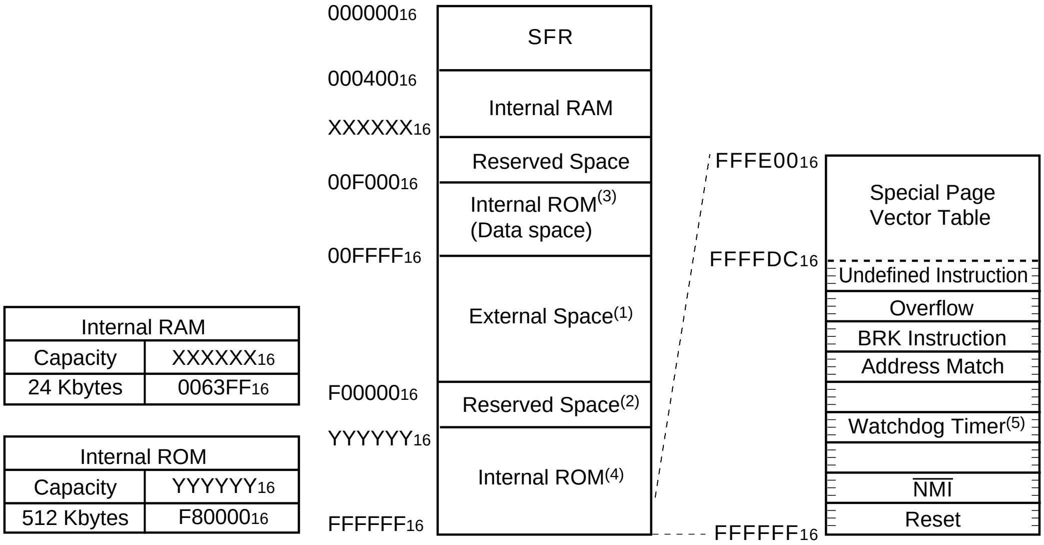
The fixed interrupt vectors are allocated from address $FFFFDC_{16}$ to $FFFFFF_{16}$. It stores the starting address of each interrupt routine.

The internal RAM is allocated from address 000400_{16} to higher. For example, a 10-Kbyte internal RAM is allocated from address 000400_{16} to $002BFF_{16}$. Besides storing data, it becomes stacks when the subroutine is called or an interrupt is acknowledged.

SFR, consisting of control registers for peripheral functions such as I/O port, A/D converter, serial I/O, timers, is allocated from address 000000_{16} to $0003FF_{16}$. All blank spaces within SFR are reserved and cannot be accessed by users.

The special page vectors are addressed from $FFFE00_{16}$ to $FFFFDB_{16}$. It is used for the JMPS instruction and JSRS instruction. Refer to the Renesas publication **M32C/80 Series Software Manual** for details.

In memory expansion mode and microprocessor mode, some spaces are reserved and cannot be accessed by users.



Internal RAM	
Capacity	$XXXXXX_{16}$
24 Kbytes	$0063FF_{16}$

Internal ROM	
Capacity	$YYYYYY_{16}$
512 Kbytes	$F80000_{16}$

NOTES:
1. In memory expansion and microprocessor modes.
2. In memory expansion mode.
   This space becomes external space in microprocessor mode.
3. Additional 4-Kbyte space is provided for storing data.
   This space can be used in single-chip mode and memory expansion mode.
   This space becomes reserved space in microprocessor mode.
4. This space can be used in single-chip mode and memory expansion mode.
   This space becomes external space in microprocessor mode.
5. Watchdog timer interrupt, oscillation stop detection interrupt, and low voltage detection interrupt share vectors.

Figure 3.1 Memory Map

4. Special Function Registers (SFR)

Address	Register	Symbol	Value after RESET
0000_{16}			
0001_{16}			
0002_{16}			
0003_{16}			
0004_{16}	Processor Mode Register(1)	PM0	1000 0000_2(CNVss pin ="L") 0000 0011_2(CNVss pin ="H")
0005_{16}	Processor Mode Register 1	PM1	00_{16}
0006_{16}	System Clock Control Register 0	CM0	0000 1000_2
0007_{16}	System Clock Control Register 1	CM1	0010 0000_2
0008_{16}			
0009_{16}	Address Match Interrupt Enable Register	AIER	00_{16}
$000A_{16}$	Protect Register	PRCR	XXXX 0000_2
$000B_{16}$	External Data Bus Width Control Register(2)	DS	XXXX 1000_2(BYTE pin ="L") XXXX 0000_2(BYTE pin ="H")
$000C_{16}$	Main Clock Division Register	MCD	XXX0 1000_2
$000D_{16}$	Oscillation Stop Detection Register	CM2	00_{16}
$000E_{16}$	Watchdog Timer Start Register	WDTS	XX_{16}
$000F_{16}$	Watchdog Timer Control Register	WDC	000X $XXXX_2$
0010_{16} 0011_{16} 0012_{16}	Address Match Interrupt Register 0	RMAD0	000000_{16}
0013_{16}	Processor Mode Register 2	PM2	00_{16}
0014_{16} 0015_{16} 0016_{16}	Address Match Interrupt Register 1	RMAD1	000000_{16}
0017_{16}	Voltage Detection Register 2(2)	VCR2	00_{16}
0018_{16} 0019_{16} $001A_{16}$	Address Match Interrupt Register 2	RMAD2	000000_{16}
$001B_{16}$	Voltage Detection Register 1(2)	VCR1	0000 1000_2
$001C_{16}$ $001D_{16}$ $001E_{16}$	Address Match Interrupt Register 3	RMAD3	000000_{16}
$001F_{16}$			
0020_{16}			
0021_{16}			
0022_{16}			
0023_{16}			
0024_{16}			
0025_{16}			
0026_{16}	PLL Control Register 0	PLC0	0001 $X010_2$
0027_{16}	PLL Control Register 1	PLC1	000X 0000_2
0028_{16} 0029_{16} $002A_{16}$	Address Match Interrupt Register 4	RMAD4	000000_{16}
$002B_{16}$			
$002C_{16}$ $002D_{16}$ $002E_{16}$	Address Match Interrupt Register 5	RMAD5	000000_{16}
$002F_{16}$	Low Voltage Detection Interrupt Register(2)	D4INT	00_{16}

X: Indeterminate

Blank spaces are reserved. No access is allowed.

NOTES:

1. The PM01 and PM00 bits in the PM0 register maintain values set before reset, even after software reset or watchdog timer reset has been performed.
2. These registers in M32C/86T cannot be used.

Address	Register	Symbol	Value after RESET
0030$_{16}$			
0031$_{16}$			
0032$_{16}$			
0033$_{16}$			
0034$_{16}$			
0035$_{16}$			
0036$_{16}$			
0037$_{16}$			
0038$_{16}$	Address Match Interrupt Register 6	RMAD6	000000$_{16}$
0039$_{16}$			
003A$_{16}$			
003B$_{16}$			
003C$_{16}$	Address Match Interrupt Register 7	RMAD7	000000$_{16}$
003D$_{16}$			
003E$_{16}$			
003F$_{16}$			
0040$_{16}$			
0041$_{16}$			
0042$_{16}$			
0043$_{16}$			
0044$_{16}$			
0045$_{16}$			
0046$_{16}$			
0047$_{16}$			
0048$_{16}$	External Space Wait Control Register 0[1]	EWCR0	X0X0 0011$_2$
0049$_{16}$	External Space Wait Control Register 1[1]	EWCR1	X0X0 0011$_2$
004A$_{16}$	External Space Wait Control Register 2[1]	EWCR2	X0X0 0011$_2$
004B$_{16}$	External Space Wait Control Register 3[1]	EWCR3	X0X0 0011$_2$
004C$_{16}$			
004D$_{16}$			
004E$_{16}$			
004F$_{16}$			
0050$_{16}$			
0051$_{16}$			
0052$_{16}$			
0053$_{16}$			
0054$_{16}$			
0055$_{16}$	Flash Memory Control Register 1	FMR1	0000 0101$_2$
0056$_{16}$			
0057$_{16}$	Flash Memory Control Register 0	FMR0	0000 0001$_2$
0058$_{16}$			
0059$_{16}$			
005A$_{16}$			
005B$_{16}$			
005C$_{16}$			
005D$_{16}$			
005E$_{16}$			
005F$_{16}$			

X: Indeterminate

Blank spaces are reserved. No access is allowed.

NOTES:

1. These registers cannot be used in M32C/86T.

Address	Register	Symbol	Value after RESET
0060_{16}			
0061_{16}			
0062_{16}			
0063_{16}			
0064_{16}			
0065_{16}			
0066_{16}			
0067_{16}			
0068_{16}	DMA0 Interrupt Control Register	DM0IC	XXXX $X000_2$
0069_{16}	Timer B5 Interrupt Control Register	TB5IC	XXXX $X000_2$
$006A_{16}$	DMA2 Interrupt Control Register	DM2IC	XXXX $X000_2$
$006B_{16}$	UART2 Receive /ACK Interrupt Control Register	S2RIC	XXXX $X000_2$
$006C_{16}$	Timer A0 Interrupt Control Register	TA0IC	XXXX $X000_2$
$006D_{16}$	UART3 Receive /ACK Interrupt Control Register	S3RIC	XXXX $X000_2$
$006E_{16}$	Timer A2 Interrupt Control Register	TA2IC	XXXX $X000_2$
$006F_{16}$	UART4 Receive /ACK Interrupt Control Register	S4RIC	XXXX $X000_2$
0070_{16}	Timer A4 Interrupt Control Register	TA4IC	XXXX $X000_2$
0071_{16}	UART0/UART3 Bus Conflict Detect Interrupt Control Register	BCN0IC/BCN3IC	XXXX $X000_2$
0072_{16}	UART0 Receive/ACK Interrupt Control Register	S0RIC	XXXX $X000_2$
0073_{16}	A/D0 Conversion Interrupt Control Register	AD0IC	XXXX $X000_2$
0074_{16}	UART1 Receive/ACK Interrupt Control Register	S1RIC	XXXX $X000_2$
0075_{16}	Intelligent I/O Interrupt Control Register 0/ CAN Interrupt 3 Control Register	IIO0IC/ CAN3IC	XXXX $X000_2$
0076_{16}	Timer B1 Interrupt Control Register	TB1IC	XXXX $X000_2$
0077_{16}	Intelligent I/O Interrupt Control Register 2	IIO2IC	XXXX $X000_2$
0078_{16}	Timer B3 Interrupt Control Register	TB3IC	XXXX $X000_2$
0079_{16}	Intelligent I/O Interrupt Control Register 4	IIO4IC	XXXX $X000_2$
$007A_{16}$	INT5 Interrupt Control Register	INT5IC	XX00 $X000_2$
$007B_{16}$			
$007C_{16}$	INT3 Interrupt Control Register	INT3IC	XX00 $X000_2$
$007D_{16}$	Intelligent I/O Interrupt Control Register 8	IIO8IC	XXXX $X000_2$
$007E_{16}$	INT1 Interrupt Control Register	INT1IC	XX00 $X000_2$
$007F_{16}$	Intelligent I/O Interrupt Control Register 10/ CAN Interrupt 1 Control Register	IIO10IC/ CAN1IC	XXXX $X000_2$
0080_{16}			
0081_{16}	CAN Interrupt 2 Control Register	CAN2IC	XXXX $X000_2$
0082_{16}			
0083_{16}			
0084_{16}			
0085_{16}			
0086_{16}			
0087_{16}			
0088_{16}	DMA1 Interrupt Control Register	DM1IC	XXXX $X000_2$
0089_{16}	UART2 Transmit /NACK Interrupt Control Register	S2TIC	XXXX $X000_2$
$008A_{16}$	DMA3 Interrupt Control Register	DM3IC	XXXX $X000_2$
$008B_{16}$	UART3 Transmit /NACK Interrupt Control Register	S3TIC	XXXX $X000_2$
$008C_{16}$	Timer A1 Interrupt Control Register	TA1IC	XXXX $X000_2$
$008D_{16}$	UART4 Transmit /NACK Interrupt Control Register	S4TIC	XXXX $X000_2$
$008E_{16}$	Timer A3 Interrupt Control Register	TA3IC	XXXX $X000_2$
$008F_{16}$	UART2 Bus Conflict Detect Interrupt Control Register	BCN2IC	XXXX $X000_2$

X: Indeterminate

Blank spaces are reserved. No access is allowed.

Address	Register	Symbol	Value after RESET
0090$_{16}$	UART0 Transmit /NACK Interrupt Control Register	S0TIC	XXXX X000$_2$
0091$_{16}$	UART1/UART4 Bus Conflict Detect Interrupt Control Register	BCN1IC/BCN4IC	XXXX X000$_2$
0092$_{16}$	UART1 Transmit/NACK Interrupt Control Register	S1TIC	XXXX X000$_2$
0093$_{16}$	Key Input Interrupt Control Register	KUPIC	XXXX X000$_2$
0094$_{16}$	Timer B0 Interrupt Control Register	TB0IC	XXXX X000$_2$
0095$_{16}$	Intelligent I/O Interrupt Control Register 1/ CAN Interrupt 4 Control Register	IIO1IC/ CAN4IC	XXXX X000$_2$
0096$_{16}$	Timer B2 Interrupt Control Register	TB2IC	XXXX X000$_2$
0097$_{16}$	Intelligent I/O Interrupt Control Register 3	IIO3IC	XXXX X000$_2$
0098$_{16}$	Timer B4 Interrupt Control Register	TB4IC	XXXX X000$_2$
0099$_{16}$	CAN Interrupt 5 Control Register	CAN5IC	XXXX X000$_2$
009A$_{16}$	INT4 Interrupt Control Register	INT4IC	XX00 X000$_2$
009B$_{16}$			
009C$_{16}$	INT2 Interrupt Control Register	INT2IC	XX00 X000$_2$
009D$_{16}$	Intelligent I/O Interrupt Control Register 9/ CAN Interrupt 0 Control Register	IIO9IC/ CAN0IC	XXXX X000$_2$
009E$_{16}$	INT0 Interrupt Control Register	INT0IC	XX00 X000$_2$
009F$_{16}$	Exit Priority Control Register	RLVL	XXXX 0000$_2$
00A0$_{16}$	Interrupt Request Register 0	IIO0IR	0000 000X$_2$
00A1$_{16}$	Interrupt Request Register 1	IIO1IR	0000 000X$_2$
00A2$_{16}$	Interrupt Request Register 2	IIO2IR	0000 000X$_2$
00A3$_{16}$	Interrupt Request Register 3	IIO3IR	0000 000X$_2$
00A4$_{16}$	Interrupt Request Register 4	IIO4IR	0000 000X$_2$
00A5$_{16}$	Interrupt Request Register 5	IIO5IR	0000 000X$_2$
00A6$_{16}$			
00A7$_{16}$			
00A8$_{16}$	Interrupt Request Register 8	IIO8IR	0000 000X$_2$
00A9$_{16}$	Interrupt Request Register 9	IIO9IR	0000 000X$_2$
00AA$_{16}$	Interrupt Request Register 10	IIO10IR	0000 000X$_2$
00AB$_{16}$	Interrupt Request Register 11	IIO11IR	0000 000X$_2$
00AC$_{16}$			
00AD$_{16}$			
00AE$_{16}$			
00AF$_{16}$			
00B0$_{16}$	Interrupt Enable Register 0	IIO0IE	00$_{16}$
00B1$_{16}$	Interrupt Enable Register 1	IIO1IE	00$_{16}$
00B2$_{16}$	Interrupt Enable Register 2	IIO2IE	00$_{16}$
00B3$_{16}$	Interrupt Enable Register 3	IIO3IE	00$_{16}$
00B4$_{16}$	Interrupt Enable Register 4	IIO4IE	00$_{16}$
00B5$_{16}$	Interrupt Enable Register 5	IIO5IE	00$_{16}$
00B6$_{16}$			
00B7$_{16}$			
00B8$_{16}$	Interrupt Enable Register 8	IIO8IE	00$_{16}$
00B9$_{16}$	Interrupt Enable Register 9	IIO9IE	00$_{16}$
00BA$_{16}$	Interrupt Enable Register 10	IIO10IE	00$_{16}$
00BB$_{16}$	Interrupt Enable Register 11	IIO11IE	00$_{16}$
00BC$_{16}$			
00BD$_{16}$			
00BE$_{16}$			
00BF$_{16}$			

X: Indeterminate

Blank spaces are reserved. No access is allowed.

Address	Register	Symbol	Value after RESET
00C0$_{16}$			
00C1$_{16}$			
00C2$_{16}$			
00C3$_{16}$			
00C4$_{16}$			
00C5$_{16}$			
00C6$_{16}$			
00C7$_{16}$			
00C8$_{16}$			
00C9$_{16}$			
00CA$_{16}$			
00CB$_{16}$			
00CC$_{16}$			
00CD$_{16}$			
00CE$_{16}$			
00CF$_{16}$			
00D0$_{16}$			
00D1$_{16}$			
00D2$_{16}$			
00D3$_{16}$			
00D4$_{16}$			
00D5$_{16}$			
00D6$_{16}$			
00D7$_{16}$			
00D8$_{16}$			
00D9$_{16}$			
00DA$_{16}$			
00DB$_{16}$			
00DC$_{16}$			
00DD$_{16}$			
00DE$_{16}$			
00DF$_{16}$			
00E0$_{16}$			
00E1$_{16}$			
00E2$_{16}$			
00E3$_{16}$			
00E4$_{16}$			
00E5$_{16}$			
00E6$_{16}$			
00E7$_{16}$			
00E8$_{16}$	SI/O Receive Buffer Register 0	G0RB	XXXX XXXX$_{2}$
00E9$_{16}$			XXX0 XXXX$_{2}$
00EA$_{16}$	Transmit Buffer/Receive Data Register 0	G0TB/G0DR	XX$_{16}$
00EB$_{16}$			
00EC$_{16}$	Receive Input Register 0	G0RI	XX$_{16}$
00ED$_{16}$	SI/O Communication Mode Register 0	G0MR	00$_{16}$
00EE$_{16}$	Transmit Output Register 0	G0TO	XX$_{16}$
00EF$_{16}$	SI/O Communication Control Register 0	G0CR	0000 X011$_{2}$

X: Indeterminate

Blank spaces are reserved. No access is allowed.

Address	Register	Symbol	Value after RESET
00F0$_{16}$	Data Compare Register 00	G0CMP0	XX$_{16}$
00F1$_{16}$	Data Compare Register 01	G0CMP1	XX$_{16}$
00F2$_{16}$	Data Compare Register 02	G0CMP2	XX$_{16}$
00F3$_{16}$	Data Compare Register 03	G0CMP3	XX$_{16}$
00F4$_{16}$	Data Mask Register 00	G0MSK0	XX$_{16}$
00F5$_{16}$	Data Mask Register 01	G0MSK1	XX$_{16}$
00F6$_{16}$	Communication Clock Select Register	CCS	XXXX 0000$_{2}$
00F7$_{16}$			
00F8$_{16}$ 00F9$_{16}$	Receive CRC Code Register 0	G0RCRC	XX$_{16}$ XX$_{16}$
00FA$_{16}$ 00FB$_{16}$	Transmit CRC Code Register 0	G0TCRC	00$_{16}$ 00$_{16}$
00FC$_{16}$	SI/O Expansion Mode Register 0	G0EMR	00$_{16}$
00FD$_{16}$	SI/O Expansion Receive Control Register 0	G0ERC	00$_{16}$
00FE$_{16}$	SI/O Special Communication Interrupt Detect Register 0	G0IRF	00$_{16}$
00FF$_{16}$	SI/O Expansion Transmit Control Register 0	G0ETC	0000 0XXX$_{2}$
0100$_{16}$ 0101$_{16}$	Time Measurement/Waveform Generating Register 10	G1TM0/G1PO0	XX$_{16}$ XX$_{16}$
0102$_{16}$ 0103$_{16}$	Time Measurement/Waveform Generating Register 11	G1TM1/G1PO1	XX$_{16}$ XX$_{16}$
0104$_{16}$ 0105$_{16}$	Time Measurement/Waveform Generating Register 12	G1TM2/G1PO2	XX$_{16}$ XX$_{16}$
0106$_{16}$ 0107$_{16}$	Time Measurement/Waveform Generating Register 13	G1TM3/G1PO3	XX$_{16}$ XX$_{16}$
0108$_{16}$ 0109$_{16}$	Time Measurement/Waveform Generating Register 14	G1TM4/G1PO4	XX$_{16}$ XX$_{16}$
010A$_{16}$ 010B$_{16}$	Time Measurement/Waveform Generating Register 15	G1TM5/G1PO5	XX$_{16}$ XX$_{16}$
010C$_{16}$ 010D$_{16}$	Time Measurement/Waveform Generating Register 16	G1TM6/G1PO6	XX$_{16}$ XX$_{16}$
010E$_{16}$ 010F$_{16}$	Time Measurement/Waveform Generating Register 17	G1TM7/G1PO7	XX$_{16}$ XX$_{16}$
0110$_{16}$	Waveform Generating Control Register 10	G1POCR0	0000 X000$_{2}$
0111$_{16}$	Waveform Generating Control Register 11	G1POCR1	0X00 X000$_{2}$
0112$_{16}$	Waveform Generating Control Register 12	G1POCR2	0X00 X000$_{2}$
0113$_{16}$	Waveform Generating Control Register 13	G1POCR3	0X00 X000$_{2}$
0114$_{16}$	Waveform Generating Control Register 14	G1POCR4	0X00 X000$_{2}$
0115$_{16}$	Waveform Generating Control Register 15	G1POCR5	0X00 X000$_{2}$
0116$_{16}$	Waveform Generating Control Register 16	G1POCR6	0X00 X000$_{2}$
0117$_{16}$	Waveform Generating Control Register 17	G1POCR7	0X00 X000$_{2}$
0118$_{16}$	Time Measurement Control Register 10	G1TMCR0	00$_{16}$
0119$_{16}$	Time Measurement Control Register 11	G1TMCR1	00$_{16}$
011A$_{16}$	Time Measurement Control Register 12	G1TMCR2	00$_{16}$
011B$_{16}$	Time Measurement Control Register 13	G1TMCR3	00$_{16}$
011C$_{16}$	Time Measurement Control Register 14	G1TMCR4	00$_{16}$
011D$_{16}$	Time Measurement Control Register 15	G1TMCR5	00$_{16}$
011E$_{16}$	Time Measurement Control Register 16	G1TMCR6	00$_{16}$
011F$_{16}$	Time Measurement Control Register 17	G1TMCR7	00$_{16}$

X: Indeterminate

Blank spaces are reserved. No access is allowed.

Address	Register	Symbol	Value after RESET
0120$_{16}$	Base Timer Register 1	G1BT	XX$_{16}$
0121$_{16}$			XX$_{16}$
0122$_{16}$	Base Timer Control Register 10	G1BCR0	00$_{16}$
0123$_{16}$	Base Timer Control Register 11	G1BCR1	X000 000X$_{2}$
0124$_{16}$	Time Measurement Prescaler Register 16	G1TPR6	00$_{16}$
0125$_{16}$	Time Measurement Prescaler Register 17	G1TPR7	00$_{16}$
0126$_{16}$	Function Enable Register 1	G1FE	00$_{16}$
0127$_{16}$	Function Select Register 1	G1FS	00$_{16}$
0128$_{16}$	SI/O Receive Buffer Register 1	G1RB	XXXX XXXX$_{2}$
0129$_{16}$			X000 XXXX$_{2}$
012A$_{16}$	Transmit Buffer/Receive Data Register 1	G1TB/G1DR	XX$_{16}$
012B$_{16}$			
012C$_{16}$	Receive Input Register 1	G1RI	XX$_{16}$
012D$_{16}$	SI/O Communication Mode Register 1	G1MR	00$_{16}$
012E$_{16}$	Transmit Output Register 1	G1TO	XX$_{16}$
012F$_{16}$	SI/O Communication Control Register 1	G1CR	0000 X011$_{2}$
0130$_{16}$	Data Compare Register 10	G1CMP0	XX$_{16}$
0131$_{16}$	Data Compare Register 11	G1CMP1	XX$_{16}$
0132$_{16}$	Data Compare Register 12	G1CMP2	XX$_{16}$
0133$_{16}$	Data Compare Register 13	G1CMP3	XX$_{16}$
0134$_{16}$	Data Mask Register 10	G1MSK0	XX$_{16}$
0135$_{16}$	Data Mask Register 11	G1MSK1	XX$_{16}$
0136$_{16}$			
0137$_{16}$			
0138$_{16}$	Receive CRC Code Register 1	G1RCRC	XX$_{16}$
0139$_{16}$			XX$_{16}$
013A$_{16}$	Transmit CRC Code Register 1	G1TCRC	00$_{16}$
013B$_{16}$			00$_{16}$
013C$_{16}$	SI/O Expansion Mode Register 1	G1EMR	00$_{16}$
013D$_{16}$	SI/O Expansion Receive Control Register 1	G1ERC	00$_{16}$
013E$_{16}$	SI/O Special Communication Interrupt Detection Register 1	G1IRF	00$_{16}$
013F$_{16}$	SI/O Expansion Transmit Control Register 1	G1ETC	0000 0XXX$_{2}$
0140$_{16}$			
0141$_{16}$			
0142$_{16}$			
0143$_{16}$			
0144$_{16}$			
0145$_{16}$			
0146$_{16}$			
0147$_{16}$			
0148$_{16}$			
0149$_{16}$			
014A$_{16}$			
014B$_{16}$			
014C$_{16}$			
014D$_{16}$			
014E$_{16}$			
014F$_{16}$			

X: Indeterminate

Blank spaces are reserved. No access is allowed.

Address	Register	Symbol	Value after RESET
0150_{16}			
0151_{16}			
0152_{16}			
0153_{16}			
0154_{16}			
0155_{16}			
0156_{16}			
0157_{16}			
0158_{16}			
0159_{16}			
$015A_{16}$			
$015B_{16}$			
$015C_{16}$			
$015D_{16}$			
$015E_{16}$			
$015F_{16}$			
0160_{16}			
0161_{16}			
0162_{16}			
0163_{16}			
0164_{16}			
0165_{16}			
0166_{16}			
0167_{16}			
0168_{16}			
0169_{16}			
$016A_{16}$			
$016B_{16}$			
$016C_{16}$			
$016D_{16}$			
$016E_{16}$			
$016F_{16}$			
0170_{16}			
0171_{16}			
0172_{16}			
0173_{16}			
0174_{16}			
0175_{16}			
0176_{16}			
0177_{16}			
0178_{16}	Input Function Select Register	IPS	00_{16}
0179_{16}	Input Function Select Register A	IPSA	00_{16}
$017A_{16}$			
$017B_{16}$			
$017C_{16}$			
$017D_{16}$ to $01DF_{16}$			

X: Indeterminate

Blank spaces are reserved. No access is allowed.

Address	Register	Symbol	Value after RESET
01E0$_{16}$	CAN0 Message Slot Buffer 0 Standard ID0	C0SLOT0_0	XX$_{16}$
01E1$_{16}$	CAN0 Message Slot Buffer 0 Standard ID1	C0SLOT0_1	XX$_{16}$
01E2$_{16}$	CAN0 Message Slot Buffer 0 Extended ID0	C0SLOT0_2	XX$_{16}$
01E3$_{16}$	CAN0 Message Slot Buffer 0 Extended ID1	C0SLOT0_3	XX$_{16}$
01E4$_{16}$	CAN0 Message Slot Buffer 0 Extended ID2	C0SLOT0_4	XX$_{16}$
01E5$_{16}$	CAN0 Message Slot Buffer 0 Data Length Code	C0SLOT0_5	XX$_{16}$
01E6$_{16}$	CAN0 Message Slot Buffer 0 Data 0	C0SLOT0_6	XX$_{16}$
01E7$_{16}$	CAN0 Message Slot Buffer 0 Data 1	C0SLOT0_7	XX$_{16}$
01E8$_{16}$	CAN0 Message Slot Buffer 0 Data 2	C0SLOT0_8	XX$_{16}$
01E9$_{16}$	CAN0 Message Slot Buffer 0 Data 3	C0SLOT0_9	XX$_{16}$
01EA$_{16}$	CAN0 Message Slot Buffer 0 Data 4	C0SLOT0_10	XX$_{16}$
01EB$_{16}$	CAN0 Message Slot Buffer 0 Data 5	C0SLOT0_11	XX$_{16}$
01EC$_{16}$	CAN0 Message Slot Buffer 0 Data 6	C0SLOT0_12	XX$_{16}$
01ED$_{16}$	CAN0 Message Slot Buffer 0 Data 7	C0SLOT0_13	XX$_{16}$
01EE$_{16}$	CAN0 Message Slot Buffer 0 Time Stamp High-Order	C0SLOT0_14	XX$_{16}$
01EF$_{16}$	CAN0 Message Slot Buffer 0 Time Stamp Low-Order	C0SLOT0_15	XX$_{16}$
01F0$_{16}$	CAN0 Message Slot Buffer 1 Standard ID0	C0SLOT1_0	XX$_{16}$
01F1$_{16}$	CAN0 Message Slot Buffer 1 Standard ID1	C0SLOT1_1	XX$_{16}$
01F2$_{16}$	CAN0 Message Slot Buffer 1 Extended ID0	C0SLOT1_2	XX$_{16}$
01F3$_{16}$	CAN0 Message Slot Buffer 1 Extended ID1	C0SLOT1_3	XX$_{16}$
01F4$_{16}$	CAN0 Message Slot Buffer 1 Extended ID2	C0SLOT1_4	XX$_{16}$
01F5$_{16}$	CAN0 Message Slot Buffer 1 Data Length Code	C0SLOT1_5	XX$_{16}$
01F6$_{16}$	CAN0 Message Slot Buffer 1 Data 0	C0SLOT1_6	XX$_{16}$
01F7$_{16}$	CAN0 Message Slot Buffer 1 Data 1	C0SLOT1_7	XX$_{16}$
01F8$_{16}$	CAN0 Message Slot Buffer 1 Data 2	C0SLOT1_8	XX$_{16}$
01F9$_{16}$	CAN0 Message Slot Buffer 1 Data 3	C0SLOT1_9	XX$_{16}$
01FA$_{16}$	CAN0 Message Slot Buffer 1 Data 4	C0SLOT1_10	XX$_{16}$
01FB$_{16}$	CAN0 Message Slot Buffer 1 Data 5	C0SLOT1_11	XX$_{16}$
01FC$_{16}$	CAN0 Message Slot Buffer 1 Data 6	C0SLOT1_12	XX$_{16}$
01FD$_{16}$	CAN0 Message Slot Buffer 1 Data 7	C0SLOT1_13	XX$_{16}$
01FE$_{16}$	CAN0 Message Slot Buffer 1 Time Stamp High-Order	C0SLOT1_14	XX$_{16}$
01FF$_{16}$	CAN0 Message Slot Buffer 1 Time Stamp Low-Order	C0SLOT1_15	XX$_{16}$
0200$_{16}$ 0201$_{16}$	CAN0 Control Register 0	C0CTLR0	XX01 0X01$_2$(1) XXXX 0000$_2$(1)
0202$_{16}$ 0203$_{16}$	CAN0 Status Register	C0STR	0000 0000$_2$(1) X000 0X01$_2$(1)
0204$_{16}$ 0205$_{16}$	CAN0 Extended ID Register	C0IDR	00$_{16}$(1) 00$_{16}$(1)
0206$_{16}$ 0207$_{16}$	CAN0 Configuration Register	C0CONR	0000 XXXX$_2$(1) 0000 0000$_2$(1)
0208$_{16}$ 0209$_{16}$	CAN0 Time Stamp Register	C0TSR	00$_{16}$(1) 00$_{16}$(1)
020A$_{16}$	CAN0 Transmit Error Count Register	C0TEC	00$_{16}$(1)
020B$_{16}$	CAN0 Receive Error Count Register	C0REC	00$_{16}$(1)
020C$_{16}$ 020D$_{16}$	CAN0 Slot Interrupt Status Register	C0SISTR	00$_{16}$(1) 00$_{16}$(1)
020E$_{16}$			
020F$_{16}$			

X: Indeterminate

Blank spaces are reserved. No access is allowed.

NOTES:

1. Values are obtained by setting the SLEEP bit in the C0SLPR register to "1" (sleep mode exited) after reset and applying the clock to the CAN module.

Address	Register	Symbol	Value after RESET	
0210$_{16}$	CAN0 Slot Interrupt Mask Register	C0SIMKR	00$_{16}$[2]	
0211$_{16}$			00$_{16}$[2]	
0212$_{16}$				
0213$_{16}$				
0214$_{16}$	CAN0 Error Interrupt Mask Register	C0EIMKR	XXXX X000$_{2}$[2]	
0215$_{16}$	CAN0 Error Interrupt Status Register	C0EISTR	XXXX X000$_{2}$[2]	
0216$_{16}$	CAN0 Error Cause Register	C0EFR	00$_{16}$[2]	
0217$_{16}$	CAN0 Baud Rate Prescaler	C0BRP	0000 0001$_{2}$[2]	
0218$_{16}$				
0219$_{16}$	CAN0 Mode Register	C0MDR	XXXX XX00$_{2}$[2]	
021A$_{16}$				
021B$_{16}$				
021C$_{16}$				
021D$_{16}$				
021E$_{16}$				
021F$_{16}$				
0220$_{16}$	CAN0 Single Shot Control Register	C0SSCTLR	00$_{16}$[2]	(Note 1)
0221$_{16}$			00$_{16}$[2]	
0222$_{16}$				
0223$_{16}$				
0224$_{16}$	CAN0 Single Shot Status Register	C0SSSTR	00$_{16}$[2]	
0225$_{16}$			00$_{16}$[2]	
0226$_{16}$				
0227$_{16}$				
0228$_{16}$	CAN0 Global Mask Register Standard ID0	C0GMR0	XXX0 0000$_{2}$[2]	
0229$_{16}$	CAN0 Global Mask Register Standard ID1	C0GMR1	XX00 0000$_{2}$[2]	
022A$_{16}$	CAN0 Global Mask Register Extended ID0	C0GMR2	XXXX 0000$_{2}$[2]	
022B$_{16}$	CAN0 Global Mask Register Extended ID1	C0GMR3	00$_{16}$[2]	
022C$_{16}$	CAN0 Global Mask Register Extended ID2	C0GMR4	XX00 0000$_{2}$[2]	
022D$_{16}$				
022E$_{16}$				
022F$_{16}$				
0230$_{16}$	CAN0 Message Slot 0 Control Register / CAN0 Local Mask Register A Standard ID0	C0MCTL0/ C0LMAR0	0000 0000$_{2}$[2] XXX0 0000$_{2}$[2]	
0231$_{16}$	CAN0 Message Slot 1 Control Register / CAN0 Local Mask Register A Standard ID1	C0MCTL1/ C0LMAR1	0000 0000$_{2}$[2] XX00 0000$_{2}$[2]	
0232$_{16}$	CAN0 Message Slot 2 Control Register / CAN0 Local Mask Register A Extended ID0	C0MCTL2/ C0LMAR2	0000 0000$_{2}$[2] XXXX 0000$_{2}$[2]	
0233$_{16}$	CAN0 Message Slot 3 Control Register / CAN0 local Mask Register A Extended ID1	C0MCTL3/ C0LMAR3	00$_{16}$[2] 00$_{16}$[2]	
0234$_{16}$	CAN0 Message Slot 4 Control Register / CAN0 Local Mask Register A Extended ID2	C0MCTL4/ C0LMAR4	0000 0000$_{2}$[2] XX00 0000$_{2}$[2]	
0235$_{16}$	CAN0 Message Slot 5 Control Register	C0MCTL5	00$_{16}$[2]	
0236$_{16}$	CAN0 Message Slot 6 Control Register	C0MCTL6	00$_{16}$[2]	
0237$_{16}$	CAN0 Message Slot 7 Control Register	C0MCTL7	00$_{16}$[2]	
0238$_{16}$	CAN0 Message Slot 8 Control Register / CAN0 Local Mask Register B Standard ID0	C0MCTL8/ C0LMBR0	0000 0000$_{2}$[2] XXX0 0000$_{2}$[2]	
0239$_{16}$	CAN0 Message Slot 9 Control Register / CAN0 Local Mask Register B Standard ID1	C0MCTL9/ C0LMBR1	0000 0000$_{2}$[2] XX00 0000$_{2}$[2]	

X: Indeterminate

Blank spaces are reserved. No access is allowed.

NOTES:

1. The BANKSEL bit in the C0CTLR1 register switches functions for addresses 0220$_{16}$ to 023F$_{16}$.
2. Values are obtained by setting the SLEEP bit in the C0SLPR register to "1" (sleep mode exited) after reset and applying the clock to the CAN module.

Address	Register	Symbol	Value after RESET	
$023A_{16}$	CAN0 Message Slot 10 Control Register / CAN0 Local Mask Register B Extended ID0	C0MCTL10/ C0LMBR2	$0000\ 0000_2$(2) $XXXX\ 0000_2$(2)	(Note 1)
$023B_{16}$	CAN0 Message Slot 11 Control Register / CAN0 Local Mask Register B Extended ID1	C0MCTL11/ C0LMBR3	00_{16}(2) 00_{16}(2)	
$023C_{16}$	CAN0 Message Slot 12 Control Register / CAN0 Local Mask Register B Extended ID2	C0MCTL12/ C0LMBR4	$0000\ 0000_2$(2) $XX00\ 0000_2$(2)	
$023D_{16}$	CAN0 Message Slot 13 Control Register	C0MCTL13	00_{16}(2)	
$023E_{16}$	CAN0 Message Slot 14 Control Register	C0MCTL14	00_{16}(2)	
$023F_{16}$	CAN0 Message Slot 15 Control Register	C0MCTL15	00_{16}(2)	
0240_{16}	CAN0 Slot Buffer Select Register	C0SBS	00_{16}(2)	
0241_{16}	CAN0 Control Register 1	C0CTLR1	$X000\ 00XX_2$(2)	
0242_{16}	CAN0 Sleep Control Register	C0SLPR	$XXXX\ XXX0_2$	
0243_{16}				
0244_{16} 0245_{16}	CAN0 Acceptance Filter Support Register	C0AFS	00_{16}(2) 01_{16}(2)	
0246_{16}				
0247_{16}				
0248_{16}				
0249_{16}				
$024A_{16}$				
$024B_{16}$				
$024C_{16}$				
$024D_{16}$				
$024E_{16}$				
$024F_{16}$				
0250_{16}	CAN1 Slot Buffer Select Register	C1SBS	00_{16}(3)	
0251_{16}	CAN1 Control Register 1	C1CTLR1	$X000\ 00XX_2$(3)	
0252_{16}	CAN1 Sleep Control Register	C1SLPR	$XXXX\ XXX0_2$	
0253_{16}				
0254_{16} 0255_{16}	CAN1 Acceptance Filter Support Register	C1AFS	00_{16}(3) 01_{16}(3)	
0256_{16}				
0257_{16}				
0258_{16}				
0259_{16}				
$025A_{16}$				
$025B_{16}$				
$025C_{16}$				
$025D_{16}$				
$025E_{16}$				
$025F_{16}$				

X: Indeterminate

Blank spaces are reserved. No access is allowed.

NOTES:

1. The BANKSEL bit in the C0CTLR1 register switches functions for addresses 0220_{16} to $023F_{16}$.
2. Values are obtained by setting the SLEEP bit in the C0SLPR register to "1" (sleep mode exited) after reset and applying the clock to the CAN module.
3. Values are obtained by setting the SLEEP bit in the C1SLPR register to "1" (sleep mode exited) after reset and applying the clock to the CAN module.

Address	Register	Symbol	Value after RESET
0260$_{16}$	CAN1 Message Slot Buffer 0 Standard ID0	C1SLOT0_0	XX$_{16}$
0261$_{16}$	CAN1 Message Slot Buffer 0 Standard ID1	C1SLOT0_1	XX$_{16}$
0262$_{16}$	CAN1 Message Slot Buffer 0 Extended ID0	C1SLOT0_2	XX$_{16}$
0263$_{16}$	CAN1 Message Slot Buffer 0 Extended ID1	C1SLOT0_3	XX$_{16}$
0264$_{16}$	CAN1 Message Slot Buffer 0 Extended ID2	C1SLOT0_4	XX$_{16}$
0265$_{16}$	CAN1 Message Slot Buffer 0 Data Length Code	C1SLOT0_5	XX$_{16}$
0266$_{16}$	CAN1 Message Slot Buffer 0 Data 0	C1SLOT0_6	XX$_{16}$
0267$_{16}$	CAN1 Message Slot Buffer 0 Data 1	C1SLOT0_7	XX$_{16}$
0268$_{16}$	CAN1 Message Slot Buffer 0 Data 2	C1SLOT0_8	XX$_{16}$
0269$_{16}$	CAN1 Message Slot Buffer 0 Data 3	C1SLOT0_9	XX$_{16}$
026A$_{16}$	CAN1 Message Slot Buffer 0 Data 4	C1SLOT0_10	XX$_{16}$
026B$_{16}$	CAN1 Message Slot Buffer 0 Data 5	C1SLOT0_11	XX$_{16}$
026C$_{16}$	CAN1 Message Slot Buffer 0 Data 6	C1SLOT0_12	XX$_{16}$
026D$_{16}$	CAN1 Message Slot Buffer 0 Data 7	C1SLOT0_13	XX$_{16}$
026E$_{16}$	CAN1 Message Slot Buffer 0 Time Stamp High-Order	C1SLOT0_14	XX$_{16}$
026F$_{16}$	CAN1 Message Slot Buffer 0 Time Stamp Low-Order	C1SLOT0_15	XX$_{16}$
0270$_{16}$	CAN1 Message Slot Buffer 1 Standard ID0	C1SLOT1_0	XX$_{16}$
0271$_{16}$	CAN1 Message Slot Buffer 1 Standard ID1	C1SLOT1_1	XX$_{16}$
0272$_{16}$	CAN1 Message Slot Buffer 1 Extended ID0	C1SLOT1_2	XX$_{16}$
0273$_{16}$	CAN1 Message Slot Buffer 1 Extended ID1	C1SLOT1_3	XX$_{16}$
0274$_{16}$	CAN1 Message Slot Buffer 1 Extended ID2	C1SLOT1_4	XX$_{16}$
0275$_{16}$	CAN1 Message Slot Buffer 1 Data Length Code	C1SLOT1_5	XX$_{16}$
0276$_{16}$	CAN1 Message Slot Buffer 1 Data 0	C1SLOT1_6	XX$_{16}$
0277$_{16}$	CAN1 Message Slot Buffer 1 Data 1	C1SLOT1_7	XX$_{16}$
0278$_{16}$	CAN1 Message Slot Buffer 1 Data 2	C1SLOT1_8	XX$_{16}$
0279$_{16}$	CAN1 Message Slot Buffer 1 Data 3	C1SLOT1_9	XX$_{16}$
027A$_{16}$	CAN1 Message Slot Buffer 1 Data 4	C1SLOT1_10	XX$_{16}$
027B$_{16}$	CAN1 Message Slot Buffer 1 Data 5	C1SLOT1_11	XX$_{16}$
027C$_{16}$	CAN1 Message Slot Buffer 1 Data 6	C1SLOT1_12	XX$_{16}$
027D$_{16}$	CAN1 Message Slot Buffer 1 Data 7	C1SLOT1_13	XX$_{16}$
027E$_{16}$	CAN1 Message Slot Buffer 1 Time Stamp High-Order	C1SLOT1_14	XX$_{16}$
027F$_{16}$	CAN1 Message Slot Buffer 1 Time Stamp Low-Order	C1SLOT1_15	XX$_{16}$
0280$_{16}$ 0281$_{16}$	CAN1 Control Register 0	C1CTLR0	XX01 0X01$_2$(1) XXXX 0000$_2$(1)
0282$_{16}$ 0283$_{16}$	CAN1 Status Register	C1STR	0000 0000$_2$(1) X000 0X01$_2$(1)
0284$_{16}$ 0285$_{16}$	CAN1 Extended ID Register	C1IDR	00$_{16}$(1) 00$_{16}$(1)
0286$_{16}$ 0287$_{16}$	CAN1 Configuration Register	C1CONR	0000 XXXX$_2$(1) 0000 0000$_2$(1)
0288$_{16}$ 0289$_{16}$	CAN1 Time Stamp Register	C1TSR	00$_{16}$(1) 00$_{16}$(1)
028A$_{16}$	CAN1 Transmit Error Count Register	C1TEC	00$_{16}$(1)
028B$_{16}$	CAN1 Receive Error Count Register	C1REC	00$_{16}$(1)
028C$_{16}$ 028D$_{16}$	CAN1 Slot Interrupt Status Register	C1SISTR	00$_{16}$(1) 00$_{16}$(1)
028E$_{16}$			
028F$_{16}$			

X: Indeterminate

Blank spaces are reserved. No access is allowed.

NOTES:

1. Values are obtained by setting the SLEEP bit in the C1SLPR register to "1" (sleep mode exited) after reset and supplying the clock to the CAN module.

Address	Register	Symbol	Value after RESET	
0290_{16} 0291_{16}	CAN1 Slot Interrupt Mask Register	C1SIMKR	00_{16} 00_{16}	
0292_{16}				
0293_{16}				
0294_{16}	CAN1 Error Interrupt Mask Register	C1EIMKR	XXXX X000$_2$[2]	
0295_{16}	CAN1 Error Interrupt Status Register	C1EISTR	XXXX X000$_2$[2]	
0296_{16}	CAN1 Error Factor Register	C1EFR	00_{16}[2]	
0297_{16}	CAN1 Baud Rate Prescaler	C1BRP	0000 0001$_2$[2]	
0298_{16}				
0299_{16}	CAN1 Mode Register	C1MDR	XXXX XX00$_2$[2]	
$029A_{16}$				
$029B_{16}$				
$029C_{16}$				
$029D_{16}$				
$029E_{16}$				
$029F_{16}$				
$02A0_{16}$ $02A1_{16}$	CAN1 Single Shot Control Register	C1SSCTLR	00_{16}[2] 00_{16}[2]	
$02A2_{16}$				
$02A3_{16}$				
$02A4_{16}$ $02A5_{16}$	CAN1 Single Shot Status Register	C1SSSTR	00_{16}[2] 00_{16}[2]	
$02A6_{16}$				
$02A7_{16}$				
$02A8_{16}$	CAN1 Global Mask Register Standard ID0	C1GMR0	XXX0 0000$_2$[2]	
$02A9_{16}$	CAN1 Global Mask Register Standard ID1	C1GMR1	XX00 0000$_2$[2]	
$02AA_{16}$	CAN1 Global Mask Register Extended ID0	C1GMR2	XXXX 0000$_2$[2]	
$02AB_{16}$	CAN1 Global Mask Register Extended ID1	C1GMR3	00_{16}[2]	
$02AC_{16}$	CAN1 Global Mask Register Extended ID2	C1GMR4	XX00 0000$_2$[2]	
$02AD_{16}$				
$02AE_{16}$				
$02AF_{16}$				(Note 1)
$02B0_{16}$	CAN1 Message Slot 0 Control Register / CAN1 Local Mask Register A Standard ID0	C1MCTL0/ C1LMAR0	0000 0000$_2$[2] XXX0 0000$_2$[2]	
$02B1_{16}$	CAN1 Message Slot 1 Control Register / CAN1 Local Mask Register A Standard ID1	C1MCTL1/ C1LMAR1	0000 0000$_2$[2] XX00 0000$_2$[2]	
$02B2_{16}$	CAN1 Message Slot 2 Control Register / CAN1 Local Mask Register A Extended ID0	C1MCTL2/ C1LMAR2	0000 0000$_2$[2] XXXX 0000$_2$[2]	
$02B3_{16}$	CAN1 Message Slot 3 Control Register / CAN1 Local Mask Register A Extended ID1	C1MCTL3/ C1LMAR3	00_{16}[2] 00_{16}[2]	
$02B4_{16}$	CAN1 Message Slot 4 Control Register / CAN1 Local Mask Register A Extended ID2	C1MCTL4/ C1LMAR4	0000 0000$_2$[2] XX00 0000$_2$[2]	
$02B5_{16}$	CAN1 Message Slot 5 Control Register	C1MCTL5	00_{16}[2]	
$02B6_{16}$	CAN1 Message Slot 6 Control Register	C1MCTL6	00_{16}[2]	
$02B7_{16}$	CAN1 Message Slot 7 Control Register	C1MCTL7	00_{16}[2]	
$02B8_{16}$	CAN1 Message Slot 8 Control Register / CAN1 Local Mask Register B Standard ID0	C1MCTL8/ C1LMBR0	0000 0000$_2$[2] XXX0 0000$_2$[2]	
$02B9_{16}$	CAN1 Message Slot 9 Control Register / CAN1 Local Mask Register B Standard ID1	C1MCTL9/ C1LMBR1	0000 0000$_2$[2] XX00 0000$_2$[2]	

X: Indeterminate

Blank spaces are reserved. No access is allowed.

NOTES:

1. The BANKSEL bit in the C1CTLR1 register switches functions for addresses $02A0_{16}$ to $02BF_{16}$.
2. Values are obtained by setting the SLEEP bit in the C1SLPR register to "1" (sleep mode exited) after reset and applying the clock to the CAN module.

Address	Register	Symbol	Value after RESET	
02BA$_{16}$	CAN1 Message Slot 10 Control Register / CAN1 Local Mask Register B Extended ID0	C1MCTL10/ C1LMBR2	0000 0000$_2$[2] XXXX 0000$_2$[2]	(Note 1)
02BB$_{16}$	CAN1 Message Slot 11 Control Register / CAN1 Local Mask Register B Extended ID1	C1MCTL11/ C1LMBR3	00$_{16}$[2] 00$_{16}$[2]	
02BC$_{16}$	CAN1 Message Slot 12 Control Register / CAN1 Local Mask Register B Extended ID2	C1MCTL12/ C1LMBR4	0000 0000$_2$[2] XX00 0000$_2$[2]	
02BD$_{16}$	CAN1 Message Slot 13 Control Register	C1MCTL13	00$_{16}$[2]	
02BE$_{16}$	CAN1 Message Slot 14 Control Register	C1MCTL14	00$_{16}$[2]	
02BF$_{16}$	CAN1 Message Slot 15 Control Register	C1MCTL15	00$_{16}$[2]	
02C0$_{16}$ 02C1$_{16}$	X0 Register Y0 Register	X0R,Y0R	XX$_{16}$ XX$_{16}$	
02C2$_{16}$ 02C3$_{16}$	X1 Register Y1 Register	X1R,Y1R	XX$_{16}$ XX$_{16}$	
02C4$_{16}$ 02C5$_{16}$	X2 Register Y2 Register	X2R,Y2R	XX$_{16}$ XX$_{16}$	
02C6$_{16}$ 02C7$_{16}$	X3 Register Y3 Register	X3R,Y3R	XX$_{16}$ XX$_{16}$	
02C8$_{16}$ 02C9$_{16}$	X4 Register Y4 Register	X4R,Y4R	XX$_{16}$ XX$_{16}$	
02CA$_{16}$ 02CB$_{16}$	X5 Register Y5 Register	X5R,Y5R	XX$_{16}$ XX$_{16}$	
02CC$_{16}$ 02CD$_{16}$	X6 Register Y6 Register	X6R,Y6R	XX$_{16}$ XX$_{16}$	
02CE$_{16}$ 02CF$_{16}$	X7 Register Y7 Register	X7R,Y7R	XX$_{16}$ XX$_{16}$	
02D0$_{16}$ 02D1$_{16}$	X8 Register Y8 Register	X8R,Y8R	XX$_{16}$ XX$_{16}$	
02D2$_{16}$ 02D3$_{16}$	X9 Register Y9 Register	X9R,Y9R	XX$_{16}$ XX$_{16}$	
02D4$_{16}$ 02D5$_{16}$	X10 Register Y10 Register	X10R,Y10R	XX$_{16}$ XX$_{16}$	
02D6$_{16}$ 02D7$_{16}$	X11 Register Y11 Register	X11R,Y11R	XX$_{16}$ XX$_{16}$	
02D8$_{16}$ 02D9$_{16}$	X12 Register Y12 Register	X12R,Y12R	XX$_{16}$ XX$_{16}$	
02DA$_{16}$ 02DB$_{16}$	X13 Register Y13 Register	X13R,Y13R	XX$_{16}$ XX$_{16}$	
02DC$_{16}$ 02DD$_{16}$	X14 Register Y14 Register	X14R,Y14R	XX$_{16}$ XX$_{16}$	
02DE$_{16}$ 02DF$_{16}$	X15 Register Y15 Register	X15R,Y15R	XX$_{16}$ XX$_{16}$	

X: Indeterminate

Blank spaces are reserved. No access is allowed.

NOTES:

1. The BANKSEL bit in the C1CTLR1 register switches functions for addresses 02A0$_{16}$ to 02BF$_{16}$.
2. Values are obtained by setting the SLEEP bit in the C1SLPR register to "1" (sleep mode exited) after reset and applying the clock to the CAN module.

Address	Register	Symbol	Value after RESET
02E0$_{16}$	X/Y Control Register	XYC	XXXX XX00$_{2}$
02E1$_{16}$			
02E2$_{16}$			
02E3$_{16}$			
02E4$_{16}$	UART1 Special Mode Register 4	U1SMR4	00$_{16}$
02E5$_{16}$	UART1 Special Mode Register 3	U1SMR3	00$_{16}$
02E6$_{16}$	UART1 Special Mode Register 2	U1SMR2	00$_{16}$
02E7$_{16}$	UART1 Special Mode Register	U1SMR	00$_{16}$
02E8$_{16}$	UART1 Transmit/Receive Mode Register	U1MR	00$_{16}$
02E9$_{16}$	UART1 Bit Rate Register	U1BRG	XX$_{16}$
02EA$_{16}$	UART1 Transmit Buffer Register	U1TB	XX$_{16}$
02EB$_{16}$			XX$_{16}$
02EC$_{16}$	UART1 Transmit/Receive Control Register 0	U1C0	0000 1000$_{2}$
02ED$_{16}$	UART1 Transmit/Receive Control Register 1	U1C1	0000 0010$_{2}$
02EE$_{16}$	UART1 Receive Buffer Register	U1RB	XX$_{16}$
02EF$_{16}$			XX$_{16}$
02F0$_{16}$			
02F1$_{16}$			
02F2$_{16}$			
02F3$_{16}$			
02F4$_{16}$	UART4 Special Mode Register 4	U4SMR4	00$_{16}$
02F5$_{16}$	UART4 Special Mode Register 3	U4SMR3	00$_{16}$
02F6$_{16}$	UART4 Special Mode Register 2	U4SMR2	00$_{16}$
02F7$_{16}$	UART4 Special Mode Register	U4SMR	00$_{16}$
02F8$_{16}$	UART4 Transmit/Receive Mode Register	U4MR	00$_{16}$
02F9$_{16}$	UART4 Bit Rate Register	U4BRG	XX$_{16}$
02FA$_{16}$	UART4 Transmit Buffer Register	U4TB	XX$_{16}$
02FB$_{16}$			XX$_{16}$
02FC$_{16}$	UART4 Transmit/Receive Control Register 0	U4C0	0000 1000$_{2}$
02FD$_{16}$	UART4 Transmit/Receive Control Register 1	U4C1	0000 0010$_{2}$
02FE$_{16}$	UART4 Receive Buffer Register	U4RB	XX$_{16}$
02FF$_{16}$			XX$_{16}$
0300$_{16}$	Timer B3, B4, B5 Count Start Flag	TBSR	000X XXXX$_{2}$
0301$_{16}$			
0302$_{16}$	Timer A1-1 Register	TA11	XX$_{16}$
0303$_{16}$			XX$_{16}$
0304$_{16}$	Timer A2-1 Register	TA21	XX$_{16}$
0305$_{16}$			XX$_{16}$
0306$_{16}$	Timer A4-1 Register	TA41	XX$_{16}$
0307$_{16}$			XX$_{16}$
0308$_{16}$	Three-Phase PWM Control Register 0	INVC0	00$_{16}$
0309$_{16}$	Three-Phase PWM Control Register 1	INVC1	00$_{16}$
030A$_{16}$	Three-Phase Output Buffer Register 0	IDB0	XX11 1111$_{2}$
030B$_{16}$	Three-Phase Output Buffer Register 1	IDB1	XX11 1111$_{2}$
030C$_{16}$	Dead Time Timer	DTT	XX$_{16}$
030D$_{16}$	Timer B2 Interrupt Generation Frequency Set Counter	ICTB2	XX$_{16}$
030E$_{16}$			
030F$_{16}$			

X: Indeterminate

Blank spaces are reserved. No access is allowed.

Address	Register	Symbol	Value after RESET
0310$_{16}$	Timer B3 Register	TB3	XX$_{16}$
0311$_{16}$			XX$_{16}$
0312$_{16}$	Timer B4 Register	TB4	XX$_{16}$
0313$_{16}$			XX$_{16}$
0314$_{16}$	Timer B5 Register	TB5	XX$_{16}$
0315$_{16}$			XX$_{16}$
0316$_{16}$			
0317$_{16}$			
0318$_{16}$			
0319$_{16}$			
031A$_{16}$			
031B$_{16}$	Timer B3 Mode Register	TB3MR	00XX 0000$_2$
031C$_{16}$	Timer B4 Mode Register	TB4MR	00XX 0000$_2$
031D$_{16}$	Timer B5 Mode Register	TB5MR	00XX 0000$_2$
031E$_{16}$			
031F$_{16}$	External Interrupt Request Source Select Register	IFSR	00$_{16}$
0320$_{16}$			
0321$_{16}$			
0322$_{16}$			
0323$_{16}$			
0324$_{16}$	UART3 Special Mode Register 4	U3SMR4	00$_{16}$
0325$_{16}$	UART3 Special Mode Register 3	U3SMR3	00$_{16}$
0326$_{16}$	UART3 Special Mode Register 2	U3SMR2	00$_{16}$
0327$_{16}$	UART3 Special Mode Register	U3SMR	00$_{16}$
0328$_{16}$	UART3 Transmit/Receive Mode Register	U3MR	00$_{16}$
0329$_{16}$	UART3 Bit Rate Register	U3BRG	XX$_{16}$
032A$_{16}$	UART3 Transmit Buffer Register	U3TB	XX$_{16}$
032B$_{16}$			XX$_{16}$
032C$_{16}$	UART3 Transmit/Receive Control Register 0	U3C0	0000 1000$_2$
032D$_{16}$	UART3 Transmit/Receive Control Register 1	U3C1	0000 0010$_2$
032E$_{16}$	UART3 Receive Buffer Register	U3RB	XX$_{16}$
032F$_{16}$			XX$_{16}$
0330$_{16}$			
0331$_{16}$			
0332$_{16}$			
0333$_{16}$			
0334$_{16}$	UART2 Special Mode Register 4	U2SMR4	00$_{16}$
0335$_{16}$	UART2 Special Mode Register 3	U2SMR3	00$_{16}$
0336$_{16}$	UART2 Special Mode Register 2	U2SMR2	00$_{16}$
0337$_{16}$	UART2 Special Mode Register	U2SMR	00$_{16}$
0338$_{16}$	UART2 Transmit/Receive Mode Register	U2MR	00$_{16}$
0339$_{16}$	UART2 Bit Rate Register	U2BRG	XX$_{16}$
033A$_{16}$	UART2 Transmit Buffer Register	U2TB	XX$_{16}$
033B$_{16}$			XX$_{16}$
033C$_{16}$	UART2 Transmit/Receive Control Register 0	U2C0	0000 1000$_2$
033D$_{16}$	UART2 Transmit/Receive Control Register 1	U2C1	0000 0010$_2$
033E$_{16}$	UART2 Receive Buffer Register	U2RB	XX$_{16}$
033F$_{16}$			XX$_{16}$

X: Indeterminate

Blank spaces are reserved. No access is allowed.

Address	Register	Symbol	Value after RESET
0340$_{16}$	Count Start Flag	TABSR	00$_{16}$
0341$_{16}$	Clock Prescaler Reset Flag	CPSRF	0XXX XXXX$_{2}$
0342$_{16}$	One-Shot Start Flag	ONSF	00$_{16}$
0343$_{16}$	Trigger Select Register	TRGSR	00$_{16}$
0344$_{16}$	Up/Down Flag	UDF	00$_{16}$
0345$_{16}$			
0346$_{16}$ 0347$_{16}$	Timer A0 Register	TA0	XX$_{16}$ XX$_{16}$
0348$_{16}$ 0349$_{16}$	Timer A1 Register	TA1	XX$_{16}$ XX$_{16}$
034A$_{16}$ 034B$_{16}$	Timer A2 Register	TA2	XX$_{16}$ XX$_{16}$
034C$_{16}$ 034D$_{16}$	Timer A3 Register	TA3	XX$_{16}$ XX$_{16}$
034E$_{16}$ 034F$_{16}$	Timer A4 Register	TA4	XX$_{16}$ XX$_{16}$
0350$_{16}$ 0351$_{16}$	Timer B0 Register	TB0	XX$_{16}$ XX$_{16}$
0352$_{16}$ 0353$_{16}$	Timer B1 Register	TB1	XX$_{16}$ XX$_{16}$
0354$_{16}$ 0355$_{16}$	Timer B2 Register	TB2	XX$_{16}$ XX$_{16}$
0356$_{16}$	Timer A0 Mode Register	TA0MR	00$_{16}$
0357$_{16}$	Timer A1 Mode Register	TA1MR	00$_{16}$
0358$_{16}$	Timer A2 Mode Register	TA2MR	00$_{16}$
0359$_{16}$	Timer A3 Mode Register	TA3MR	00$_{16}$
035A$_{16}$	Timer A4 Mode Register	TA4MR	00$_{16}$
035B$_{16}$	Timer B0 Mode Register	TB0MR	00XX 0000$_{2}$
035C$_{16}$	Timer B1 Mode Register	TB1MR	00XX 0000$_{2}$
035D$_{16}$	Timer B2 Mode Register	TB2MR	00XX 0000$_{2}$
035E$_{16}$	Timer B2 Special Mode Register	TB2SC	XXXX XXX0$_{2}$
035F$_{16}$	Count Source Prescaler Register(1)	TCSPR	0XXX 0000$_{2}$
0360$_{16}$			
0361$_{16}$			
0362$_{16}$			
0363$_{16}$			
0364$_{16}$	UART0 Special Mode Register 4	U0SMR4	00$_{16}$
0365$_{16}$	UART0 Special Mode Register 3	U0SMR3	00$_{16}$
0366$_{16}$	UART0 Special Mode Register 2	U0SMR2	00$_{16}$
0367$_{16}$	UART0 Special Mode Register	U0SMR	00$_{16}$
0368$_{16}$	UART0 Transmit/Receive Mode Register	U0MR	00$_{16}$
0369$_{16}$	UART0 Bit Rate Register	U0BRG	XX$_{16}$
036A$_{16}$ 036B$_{16}$	UART0 Transmit Buffer Register	U0TB	XX$_{16}$ XX$_{16}$
036C$_{16}$	UART0 Transmit/Receive Control Register 0	U0C0	0000 1000$_{2}$
036D$_{16}$	UART0 Transmit/Receive Control Register 1	U0C1	0000 0010$_{2}$
036E$_{16}$ 036F$_{16}$	UART0 Receive Buffer Register	U0RB	XX$_{16}$ XX$_{16}$

X: Indeterminate

Blank spaces are reserved. No access is allowed.

NOTES:

1. The TCSPR register maintains values set before reset, even after software reset or watchdog timer reset has been performed.

Address	Register	Symbol	Value after RESET
0370$_{16}$			
0371$_{16}$			
0372$_{16}$			
0373$_{16}$			
0374$_{16}$			
0375$_{16}$			
0376$_{16}$			
0377$_{16}$			
0378$_{16}$	DMA0 Request Source Select Register	DM0SL	0X00 0000$_2$
0379$_{16}$	DMA1 Request Source Select Register	DM1SL	0X00 0000$_2$
037A$_{16}$	DMA2 Request Source Select Register	DM2SL	0X00 0000$_2$
037B$_{16}$	DMA3 Request Source Select Register	DM3SL	0X00 0000$_2$
037C$_{16}$	CRC Data Register	CRCD	XX$_{16}$
037D$_{16}$			XX$_{16}$
037E$_{16}$	CRC Input Register	CRCIN	XX$_{16}$
037F$_{16}$			
0380$_{16}$	A/D0 Register 0	AD00	XXXX XXXX$_2$
0381$_{16}$			0000 0000$_2$
0382$_{16}$	A/D0 Register 1	AD01	XX$_{16}$
0383$_{16}$			XX$_{16}$
0384$_{16}$	A/D0 Register 2	AD02	XX$_{16}$
0385$_{16}$			XX$_{16}$
0386$_{16}$	A/D0 Register 3	AD03	XX$_{16}$
0387$_{16}$			XX$_{16}$
0388$_{16}$	A/D0 Register 4	AD04	XX$_{16}$
0389$_{16}$			XX$_{16}$
038A$_{16}$	A/D0 Register 5	AD05	XX$_{16}$
038B$_{16}$			XX$_{16}$
038C$_{16}$	A/D0 Register 6	AD06	XX$_{16}$
038D$_{16}$			XX$_{16}$
038E$_{16}$	A/D0 Register 7	AD07	XX$_{16}$
038F$_{16}$			XX$_{16}$
0390$_{16}$			
0391$_{16}$			
0392$_{16}$	A/D0 Control Register 4	AD0CON4	XXXX 00XX$_2$
0393$_{16}$			
0394$_{16}$	A/D0 Control Register 2	AD0CON2	XX0X X000$_2$
0395$_{16}$	A/D0 Control Register 3	AD0CON3	XXXX X000$_2$
0396$_{16}$	A/D0 Control Register 0	AD0CON0	00$_{16}$
0397$_{16}$	A/D0 Control Register 1	AD0CON1	00$_{16}$
0398$_{16}$	D/A Register 0	DA0	XX$_{16}$
0399$_{16}$			
039A$_{16}$	D/A Register 1	DA1	XX$_{16}$
039B$_{16}$			
039C$_{16}$	D/A Control Register	DACON	XXXX XX00$_2$
039D$_{16}$			
039E$_{16}$			
039F$_{16}$			

X: Indeterminate

Blank spaces are reserved. No access is allowed.

Address	Register	Symbol	Value after RESET
03A0$_{16}$	Function Select Register A8	PS8	X000 0000$_{2}$
03A1$_{16}$	Function Select Register A9	PS9	00$_{16}$
03A2$_{16}$			
03A3$_{16}$			
03A4$_{16}$			
03A5$_{16}$			
03A6$_{16}$			
03A7$_{16}$	Function Select Register D1	PSD1	X0XX XX00$_{2}$
03A8$_{16}$			
03A9$_{16}$			
03AA$_{16}$			
03AB$_{16}$			
03AC$_{16}$	Function Select Register C2	PSC2	XXXX X00X$_{2}$
03AD$_{16}$	Function Select Register C3	PSC3	X0XX XXXX$_{2}$
03AE$_{16}$			
03AF$_{16}$	Function Select Register C	PSC	00X0 0000$_{2}$
03B0$_{16}$	Function Select Register A0	PS0	00$_{16}$
03B1$_{16}$	Function Select Register A1	PS1	00$_{16}$
03B2$_{16}$	Function Select Register B0	PSL0	00$_{16}$
03B3$_{16}$	Function Select Register B1	PSL1	00$_{16}$
03B4$_{16}$	Function Select Register A2	PS2	00X0 0000$_{2}$
03B5$_{16}$	Function Select Register A3	PS3	00$_{16}$
03B6$_{16}$	Function Select Register B2	PSL2	00X0 0000$_{2}$
03B7$_{16}$	Function Select Register B3	PSL3	00$_{16}$
03B8$_{16}$			
03B9$_{16}$	Function Select Register A5	PS5	XXX0 0000$_{2}$
03BA$_{16}$			
03BB$_{16}$			
03BC$_{16}$	Function Select Register A6	PS6	00$_{16}$
03BD$_{16}$	Function Select Register A7	PS7	00$_{16}$
03BE$_{16}$	Function Select Register B6	PSL6	00$_{16}$
03BF$_{16}$	Function Select Register B7	PSL7	00$_{16}$
03C0$_{16}$	Port P6 Register	P6	XX$_{16}$
03C1$_{16}$	Port P7 Register	P7	XX$_{16}$
03C2$_{16}$	Port P6 Direction Register	PD6	00$_{16}$
03C3$_{16}$	Port P7 Direction Register	PD7	00$_{16}$
03C4$_{16}$	Port P8 Register	P8	XX$_{16}$
03C5$_{16}$	Port P9 Register	P9	XX$_{16}$
03C6$_{16}$	Port P8 Direction Register	PD8	00X0 0000$_{2}$
03C7$_{16}$	Port P9 Direction Register	PD9	00$_{16}$
03C8$_{16}$	Port P10 Register	P10	XX$_{16}$
03C9$_{16}$	Port P11 Register	P11	XX$_{16}$
03CA$_{16}$	Port P10 Direction Register	PD10	00$_{16}$
03CB$_{16}$	Port P11 Direction Register	PD11	XXX0 0000$_{2}$
03CC$_{16}$	Port P12 Register	P12	XX$_{16}$
03CD$_{16}$	Port P13 Register	P13	XX$_{16}$
03CE$_{16}$	Port P12 Direction Register	PD12	00$_{16}$
03CF$_{16}$	Port P13 Direction Register	PD13	00$_{16}$

X: Indeterminate

Blank spaces are reserved. No access is allowed.

Address	Register	Symbol	Value after RESET
03D0$_{16}$	Port P14 Register	P14	XX$_{16}$
03D1$_{16}$	Port P15 Register	P15	XX$_{16}$
03D2$_{16}$	Port P14 Direction Register	PD14	X000 0000$_{2}$
03D3$_{16}$	Port P15 Direction Register	PD15	00$_{16}$
03D4$_{16}$			
03D5$_{16}$			
03D6$_{16}$			
03D7$_{16}$			
03D8$_{16}$			
03D9$_{16}$			
03DA$_{16}$	Pull-Up Control Register 2	PUR2	00$_{16}$
03DB$_{16}$	Pull-Up Control Register 3	PUR3	00$_{16}$
03DC$_{16}$	Pull-Up Control Register 4	PUR4	XXXX 0000$_{2}$
03DD$_{16}$			
03DE$_{16}$	Output Port Switch Register	OPS	00$_{16}$
03DF$_{16}$			
03E0$_{16}$	Port P0 Register	P0	XX$_{16}$
03E1$_{16}$	Port P1 Register	P1	XX$_{16}$
03E2$_{16}$	Port P0 Direction Register	PD0	00$_{16}$
03E3$_{16}$	Port P1 Direction Register	PD1	00$_{16}$
03E4$_{16}$	Port P2 Register	P2	XX$_{16}$
03E5$_{16}$	Port P3 Register	P3	XX$_{16}$
03E6$_{16}$	Port P2 Direction Register	PD2	00$_{16}$
03E7$_{16}$	Port P3 Direction Register	PD3	00$_{16}$
03E8$_{16}$	Port P4 Register	P4	XX$_{16}$
03E9$_{16}$	Port P5 Register	P5	XX$_{16}$
03EA$_{16}$	Port P4 Direction Register	PD4	00$_{16}$
03EB$_{16}$	Port P5 Direction Register	PD5	00$_{16}$
03EC$_{16}$			
03ED$_{16}$			
03EE$_{16}$			
03EF$_{16}$			
03F0$_{16}$	Pull-Up Control Register 0	PUR0	00$_{16}$
03F1$_{16}$	Pull-Up Control Register 1	PUR1	XXXX 0000$_{2}$
03F2$_{16}$			
03F3$_{16}$			
03F4$_{16}$			
03F5$_{16}$			
03F6$_{16}$			
03F7$_{16}$			
03F8$_{16}$			
03F9$_{16}$			
03FA$_{16}$			
03FB$_{16}$			
03FC$_{16}$			
03FD$_{16}$			
03FE$_{16}$			
03FF$_{16}$	Port Control Register	PCR	XXXX XXX0$_{2}$

X: Indeterminate

Blank spaces are reserved. No access is allowed.

5. Reset

Hardware reset 1, brown-out detection reset (hardware reset 2), software reset and watchdog timer reset are available to reset the microcomputer.

5.1 Hardware Reset 1

Pins, the CPU and SFR are reset by setting the $\overline{\text{RESET}}$ pin. If the supply voltage meets the recommended operating conditions, all pins are reset when a low-level ("L") signal is applied to the $\overline{\text{RESET}}$ pin (see **Table 5.1**). The oscillation circuit is also reset and the main clock starts oscillating. The CPU and SFR are reset when the signal applied to the $\overline{\text{RESET}}$ pin changes "L" to high level ("H"). The microcomputer executes the program in an address indicated by the reset vector. The internal RAM is not reset. When an "L" signal is applied to the $\overline{\text{RESET}}$ pin while writing data to the internal RAM, the internal RAM is in an indeterminate state.

Figure 5.1 shows an example of the reset circuit. Figure 5.2 shows a reset sequence. Table 5.1 lists pin states while the $\overline{\text{RESET}}$ pin is held "L".

5.1.1 Reset on a Stable Supply Voltage

(1) Apply an "L" signal to the $\overline{\text{RESET}}$ pin
(2) Provide 20 or more clock cycle inputs into the XIN pin
(3) Apply an "H" signal to the $\overline{\text{RESET}}$ pin

5.1.2 Power-on Reset

(1) Apply an "L" signal to the $\overline{\text{RESET}}$ pin
(2) Raise the supply voltage to the recommended operating level
(3) Insert *td(P-R)* ms as wait time for the internal voltage to stabilize
(4) Provide 20 or more clock cycle inputs into the XIN pin
(5) Apply an "H" signal to the $\overline{\text{RESET}}$ pin

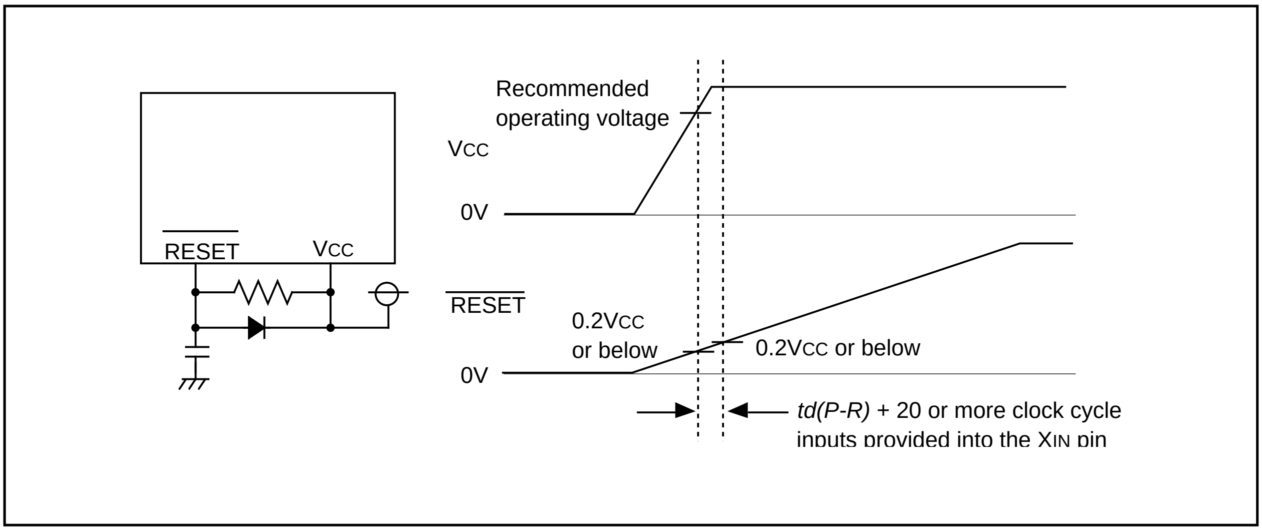
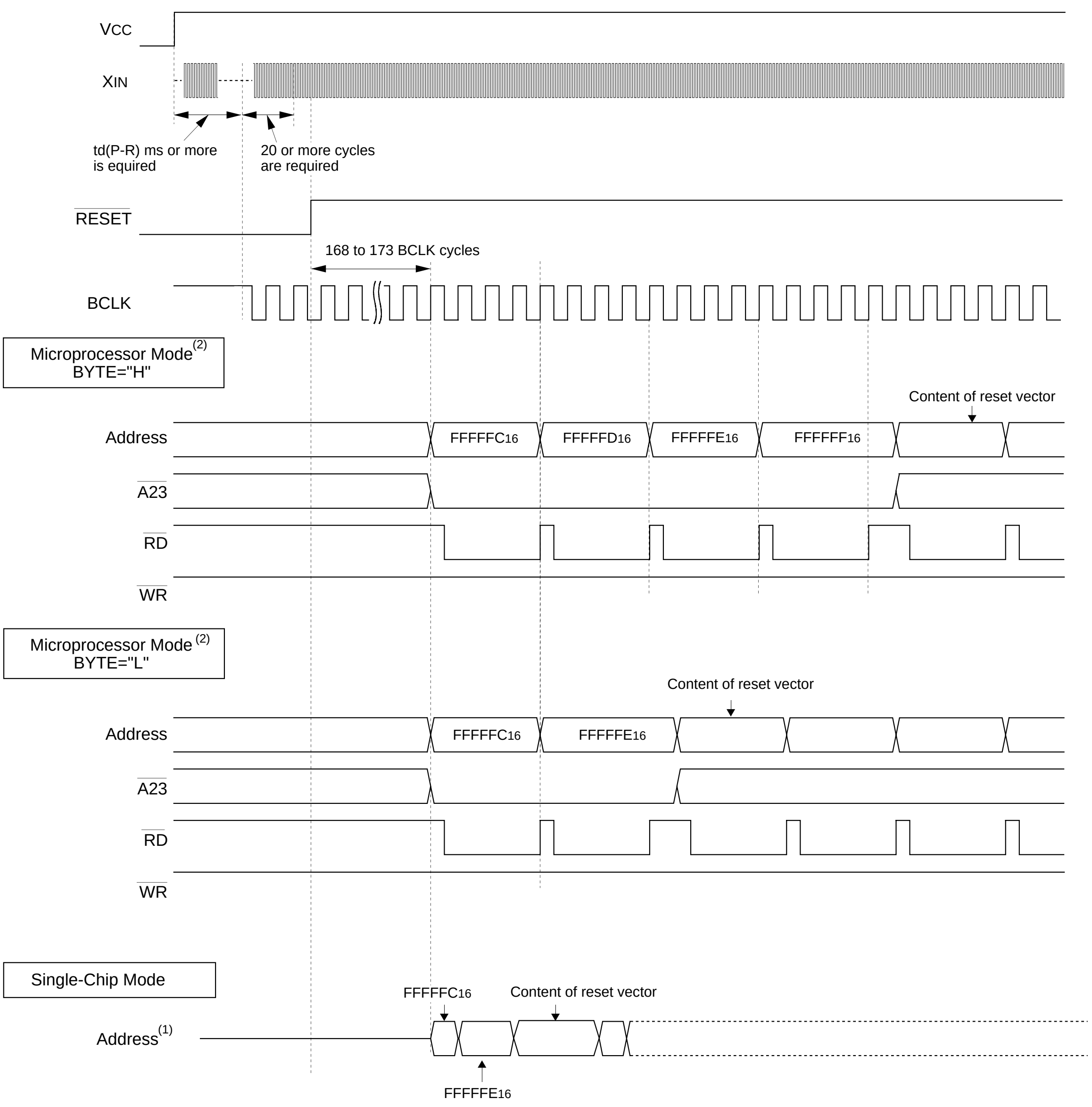

$\overline{\text{RESET}}$
VCC
Recommended operating voltage
VCC
0V
$\overline{\text{RESET}}$
0.2VCC or below
0.2VCC or below
0V
td(P-R) + 20 or more clock cycle inputs provided into the XIN pin

Figure 5.1 Reset Circuit



NOTES:
1. Address data is not output from pins in single-chip mode.
2. M32C/86T cannot be used in memory expansion mode and microprocessor mode.

Figure 5.2 Reset Sequence

Table 5.1 Pin States while $\overline{\text{RESET}}$ Pin is Held "L"

Pin Name	Pin States[2]		
	$\text{CNV}_{\text{SS}}\text{=V}_{\text{SS}}$	$\text{CNV}_{\text{SS}}\text{=V}_{\text{CC}}$	
		$\text{BYTE=V}_{\text{SS}}$	$\text{BYTE=V}_{\text{CC}}$
P0	Input port (high-impedance)	Inputs data (high-impedance)	
P1	Input port (high-impedance)	Inputs data (high-impedance)	Input port (high-impedance)
P2, P3, P4	Input port (high-impedance)	Output addresses (indeterminate)	
P5_0	Input port (high-impedance)	Outputs the $\overline{\text{WR}}$ signal ("H")[2]	
P5_1	Input port (high-impedance)	Outputs the $\overline{\text{BHE}}$ signal (indeterminate)	
P5_2	Input port (high-impedance)	Outputs the $\overline{\text{RD}}$ signal ("H")[2]	
P5_3	Input port (high-impedance)	Outputs the BCLK[2]	
P5_4	Input port (high-impedance)	Outputs the $\overline{\text{HLDA}}$ signal (Output signal depends on an input signal to the $\overline{\text{HOLD}}$ pin.)[2]	
P5_5	Input port (high-impedance)	Inputs the $\overline{\text{HOLD}}$ signal (high-impedance)	
P5_6	Input port (high-impedance)	Outputs an "H" signal[2]	
P5_7	Input port (high-impedance)	Inputs the $\overline{\text{RDY}}$ signal (high-impedance)	
P6 to P15[1]	Input port (high-impedance)	Input port (high-impedance)	

NOTES:

1. The availability of pull-up resistors is indeterminate until internal supply voltage stabilizes.
2. Each port is in this state after power is on and internal supply voltage stabilizes, but in an indeterminate state until internal supply voltage stabilizes.

5.2 Brown-Out Detection Reset (Hardware Reset 2)

Pins, the CPU and SFR are reset by using the built-in voltage detection circuit, which monitors the voltage applied to the V_{CC} pin.

When the VC26 bit in the VCR2 register is set to "1" (reset level detection circuit enabled), pins, the CPU and SFR are reset as soon as the voltage applied to the V_{CC} pin drops to Vdet3 or below.

Then, pins, the CPU and SFR are reset as soon as the voltage applied to the V_{CC} pin reaches Vdet3r or above. The microcomputer executes the program in an address determined by the reset vector.

The microcomputer executes the program after detecting Vdet3r and waiting *td(S-R)* ms . The same pins and registers are reset by the hardware reset 1 and brown-out detection reset, and are also placed in the same reset state.

The microcomputer cannot exit stop mode by brown-out detection reset.

Figure 5.3 shows an example of brown-out detection reset operation.

NOTES:

1. Brown-out detection reset cannot be used in M32C/86T.

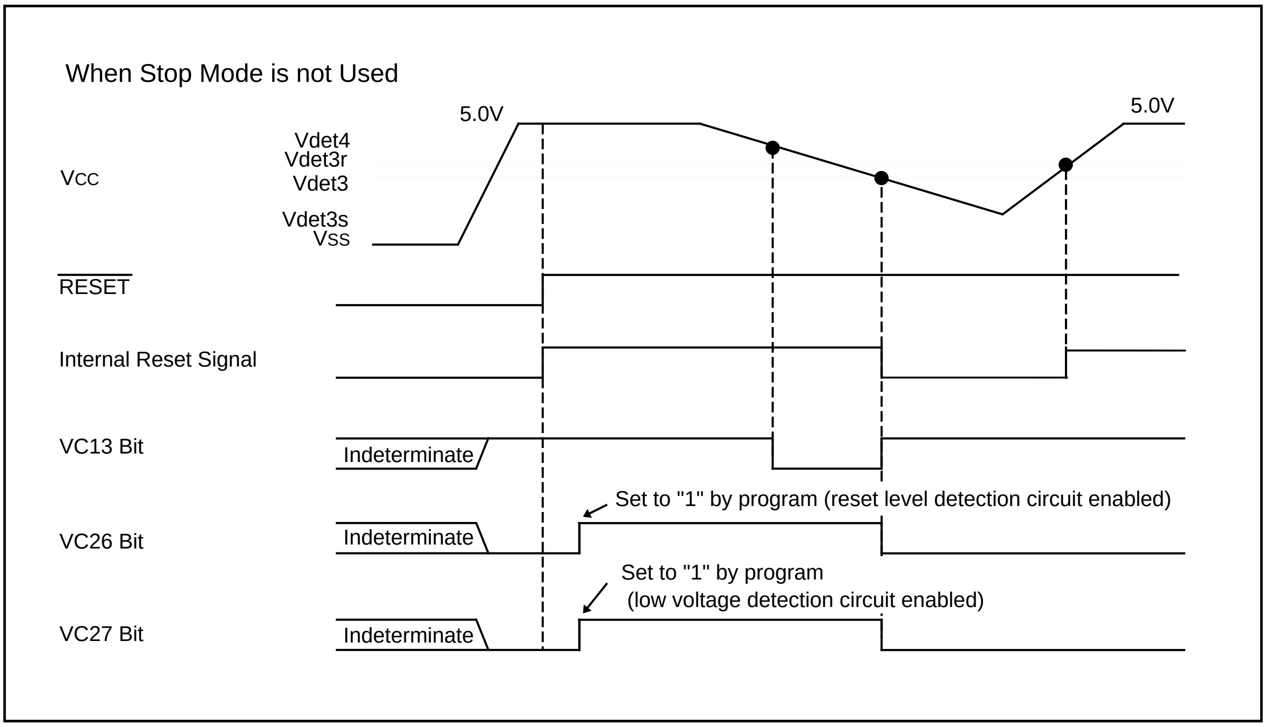


Figure 5.3 Brown-out Detection Reset (Hardware Reset 2)

5.3 Software Reset

Pins, the CPU and SFR are reset when the PM03 bit in the PM0 register is set to "1" (microcomputer reset). Then the microcomputer executes the program in an address determined by the reset vector.

Set the PM03 bit to "1" while the main clock is selected as the CPU clock and the main clock oscillation is stable.

In the software reset, the microcomputer does not reset a part of the SFR. Refer to **4. SFR** for details.

Processor mode remains unchanged since the PM01 and PM00 bits in the PM0 register are not reset.

5.4 Watchdog Timer Reset

Pins, the CPU and SFR are reset when the CM06 bit in the CM0 register is set to "1" (reset) and the watchdog timer underflows. Then the microcomputer executes the program in an address determined by the reset vector.

In the watchdog timer reset, the microcomputer does not reset a part of the SFR. Refer to **4. SFR** for details.

Processor mode remains unchanged since the PM01 and PM00 bits in the PM0 register are not reset.

5.5 Internal Space

Figure 5.4 shows CPU register states after reset. Refer to **4. SFR** for SFR states after reset.

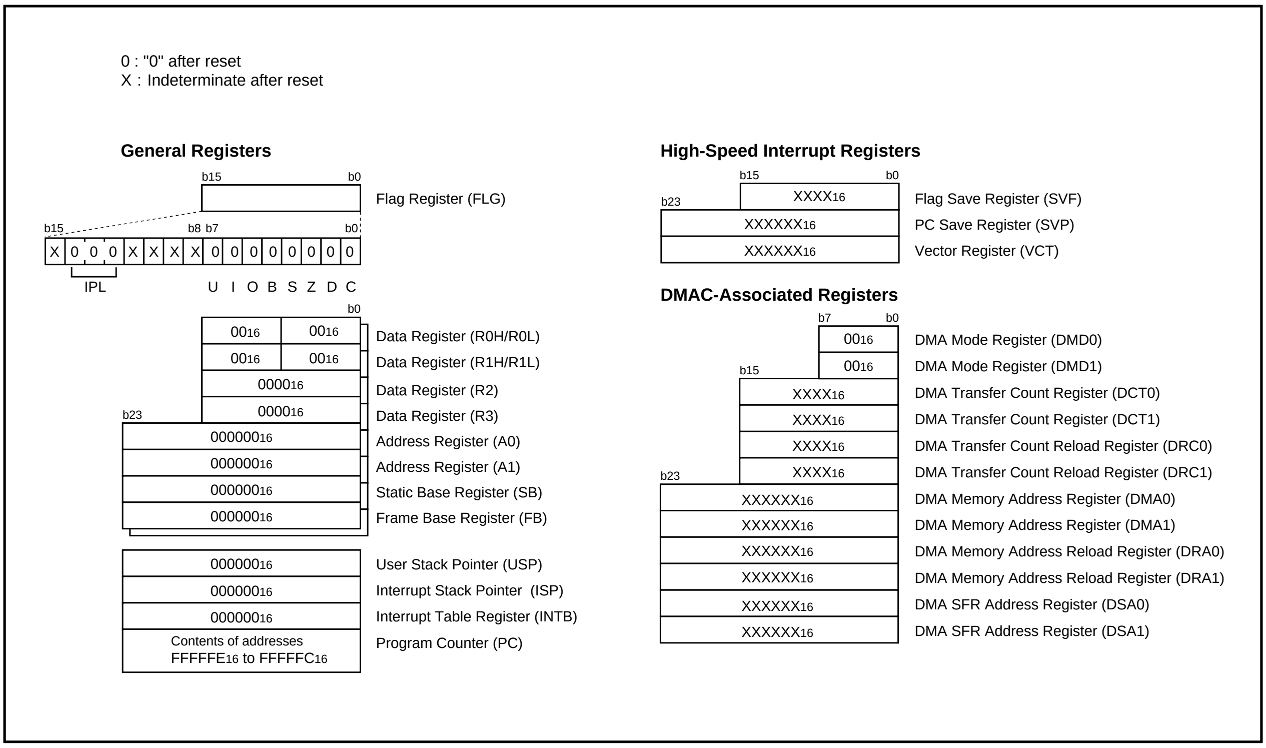


Figure 5.4 CPU Register States after Reset

6. Voltage Detection Circuit

NOTE

The voltage detection circuit in M32C/86T cannot be used.
However, the cold start-up/warm start-up determine function is available.

The voltage detection circuit consists of the reset level detection circuit and the low voltage detection circuit. The reset level detection circuit monitors the voltage applied to the VCC pin. The microcomputer is reset if the reset level detection circuit detects VCC is Vdet3 or below. This circuit is disabled when the microcomputer is in stop mode.

The voltage detection circuit also monitors the voltage applied to the VCC pin. The low voltage detection signal is generated when the low voltage detection circuit detects VCC is above or below Vdet4. This signal generates the low voltage detection interrupt. The VC13 bit in the VCR1 register determines whether VCC is above or below Vdet4.

The voltage detection circuit is available when VCC=4.2V to 5.5V.

Figure 6.1 shows a block diagram of the voltage detection circuit.

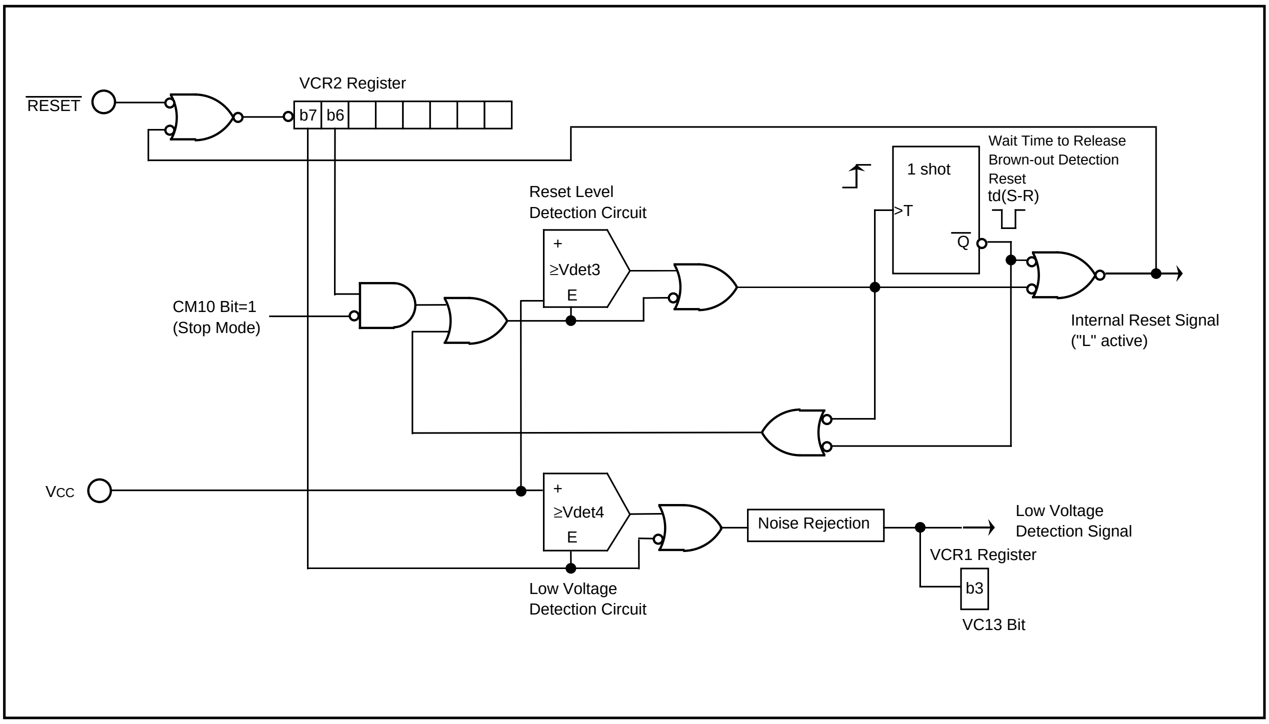
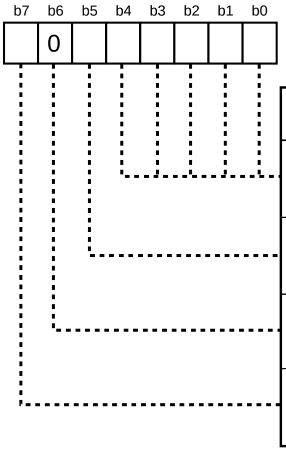


Figure 6.1 Voltage Detection Circuit Block Diagram

Watchdog Timer Control Register



Symbol	Address	After Reset
WDC	$000F_{16}$	$000X\ XXXX_2$

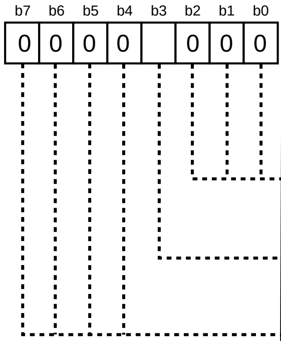
Bit Symbol	Bit Name	Function	RW
(b4 - b0)	High-Order Bit of the Watchdog Timer		RO
WDC5	Cold Start-up/ Warm Start-up Determine Flag(1,2, 3)	0 : Cold start-up 1 : Warm start-up	RW
(b6)	Reserved Bit	Set to "0"	RW
WDC7	Prescaler Select Bit	0 : Divide-by-16 1 : Divide-by-128	RW

NOTES:

1. The WDC5 bit remains set to "1", regardless of setting to "1" or "0".
2. The WDC5 bit is set to "0" when power is turned on and can be set to "1" by program only.
3. The WDC5 bit maintains a value set before reset, even after reset has been performed.

Figure 6.2 WDC Register

Voltage Detection Register 1[2]



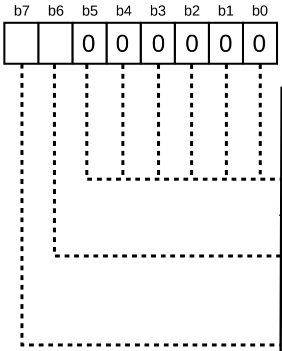
Symbol	Address	After Reset
VCR1	$001B_{16}$	$0000\ 1000_2$

Bit Symbol	Bit Name	Function	RW
——— (b2 - b0)	Reserved Bit	Set to "0"	RW
VC13	Low Voltage Monitor Flag[1]	0 : VCC < Vdet4 1 : VCC ≥ Vdet4	RO
——— (b7 - b4)	Reserved Bit	Set to "0"	RW

NOTES:

1. The VC13 bit setting is enabled when the VC27 bit in the VCR2 register is set to "1" (low voltage detection circuit enabled). The VC13 bit is set to "1" when the VC27 bit is set to "0" (low voltage detection circuit disabled).
2. The VCR1 register in M32C/86T cannot be used.

Voltage Detection Register 2[1, 5]



Symbol	Address	After Reset
VCR2	0017_{16}	00_{16}[2]

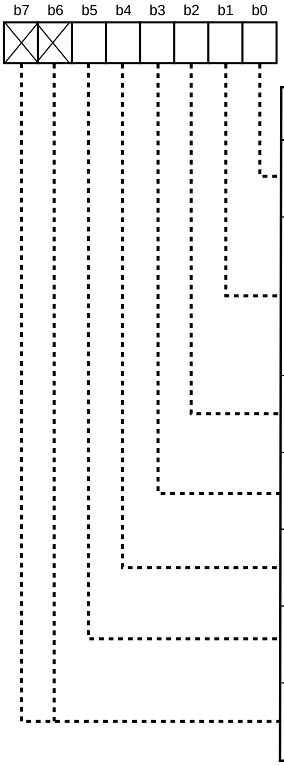
Bit Symbol	Bit Name	Function	RW
——— (b5 - b0)	Reserved Bit	Set to "0"	RW
VC26	Reset Level Monitor Bit[2, 4, 6]	0 : Disables reset level detection circuit 1 : Enables reset level detection circuit	RW
VC27	Low Voltage Monitor Bit[3, 4]	0 : Disables low voltage detection circuit 1 : Enables low voltage detection circuit	RW

NOTES:

1. Set the VCR2 register after the PRC3 bit in the PRCR register is set to "1" (write enable).
2. To use the brown-out detection reset (hardware reset 2), set the VC26 bit to "1".
3. Set the VC27 bit to "1" to set the VC13 bit in the VCR1 register and the D42 bit in the D4INT register, or to set the D40 bit to "1" (low voltage detect interrupt enabled).
4. The reset level detection circuit and low voltage detection circuit start operating *td(E-A)* ms after the VC26 or VC27 bit is set to "1".
5. The VCR2 register in M32C/86T cannot be used.
6. The VC26 bit setting is disabled when the microcomputer is in stop mode. Its setting is not reset even if the voltage applied to the VCC pin drops below Vdet3.

Figure 6.3 VCR1 and VCR2 Registers

Low Voltage Detection Interrupt Register(1,6)



Symbol	Address	After Reset
D4INT	002F$_{16}$	00$_{16}$

Bit Symbol	Bit Name	Function	RW
D40	Low Voltage Detection Interrupt Enable Bit(2)	0: Disables the interrupt 1: Enables the interrupt	RW
D41	Stop/Wait Mode Deactivation Control Bit(3)	0: Disabled (cannot use the low voltage detection interrupt to exit stop/wait mode) 1: Enabled (can use the low voltage detection interrupt to exit stop/wait mode)	RW
D42	Voltage Change Detect Flag(4, 5)	0: Not detected 1: Detects above or below Vdet4	RW
D43	WDT Overflow Detect Flag(5)	0: Not detected 1: Detected	RW
DF0	Sampling Clock Select Bit	b5 b4 0 0 : CPU clock divided by 8 0 1 : CPU clock divided by 16 1 0 : CPU clock divided by 32 1 1 : CPU clock divided by 64	RW
DF1			RW
—— (b7 - b6)	Reserved Bit	When read, its content is indeterminate	RO

NOTES:

1. Set the D4INT registers after the PRC3 bit in the PRCR register is set to "1" (write enable).
2. The D40 bit setting is enabled when the VC27 bit in the VCR2 register is set to "1" (low voltage detection circuit enabled).
   Use the following procedure to set the D40 bit to "1":
   (1) Set the VC27 bit to "1"
   (2) Wait *td(E-A)* ms to start operating the voltage detection circuit
   (3) Wait required sampling time (see Table 6.2)
   (4) Set the D40 bit to "1"
3. When exiting stop mode using the low voltage detection circuit again after having already done so, set the D41 bit to "1" after setting it to "0".
4. The D42 bit setting is enabled when the VC27 bit in the VCR2 register is set to "1" (low voltage detection circuit enabled). The D42 bit is set to "0" when the VC27 bit is set to "0" (low voltage detection circuit disabled).
5. The bit is set to "0" by a program. (It remains unchanged even if it is set to "1".)
6. The D4INT register in M32C/86T cannot be used.

Figure 6.4 D4INT Register

6.1 Low Voltage Detection Interrupt

If the D40 bit in the D4INT register is set to "1" (low voltage detection interrupt enabled), low voltage detection interrupt request is generated when the voltage applied to the VCC pin rises above or drops below Vdet4. The low voltage detection interrupt shares the same interrupt vector with the watchdog timer interrupt and oscillation stop detection interrupt. The D42 bit in the D4INT register determines whether the low voltage detection interrupt has been generated. Read the D42 bit using an interrupt routine when using the low voltage detection interrupt at the same time as the watchdog timer interrupt and oscillation stop detection interrupt.

Set the D41 bit in the D4INT register to "1" (enabled) to use the low voltage detection interrupt to exit stop mode or wait mode.

The D42 bit is set to "1" (more or less than Vdet4 detected) as soon as the voltage applied to the VCC pin reaches Vdet4 due to the voltage rise and voltage drop. When the D42 bit setting changes "0" to "1", low voltage detection interrupt request is generated. Set the D42 bit to "0" (not detected) by program. However, when the D41 bit is set to "1" and the microcomputer is in stop mode or wait mode, low voltage detection interrupt request is generated, regardless of the D42 bit setting, if the voltage applied to the VCC pin is detected to be higher than Vdet4. The microcomputer then exits stop mode or wait mode.

Table 6.1 shows how a low voltage detection interrupt request is generated.

The DF1 and DF0 bits in the D4INT register determine sampling period that detects the voltage applied to the VCC pin rises above or drops below Vdet4. Table 6.2 shows the sampling periods.

Table 6.1 Conditions to Generate Low Voltage Detection Interrupt Request

Operating Mode	VC27 Bit	D40 Bit	D41 Bit	D42 Bit[4]	VC13 Bit[3]
Normal Operating Mode[1]	1	1	"0" or "1"	"0" to"1"	"0" to"1" "1" to"0"
Wait Mode[2], Stop Mode[2]			1	-	"0" to"1"

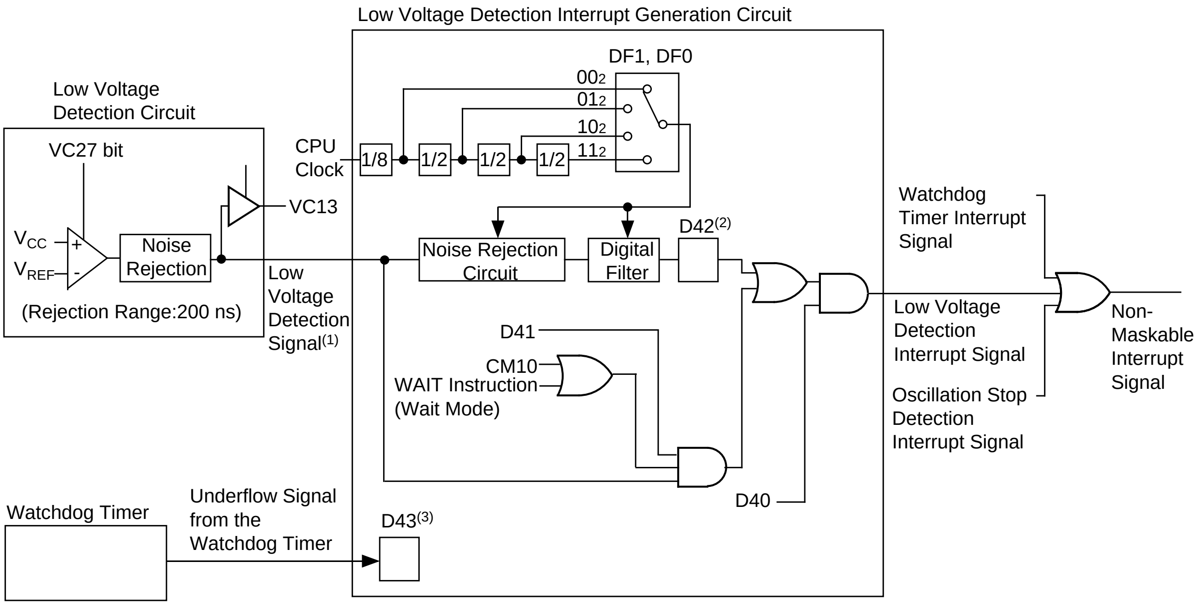
- : "0" or "1"

NOTES:

1. All states excluding wait mode and stop mode are handled as normal operating mode. (**Refer to 9. Clock Generation Circuit.**)
2. Refer to **6.1.1 Limitations for Exiting Stop/Wait Mode**.
3. Sampling begins after the VC13 bit setting changes. An interrupt request is generated after sampling is completed. See Figure 6.6 for details.
4. Set to "0" by program before generating an interrupt.

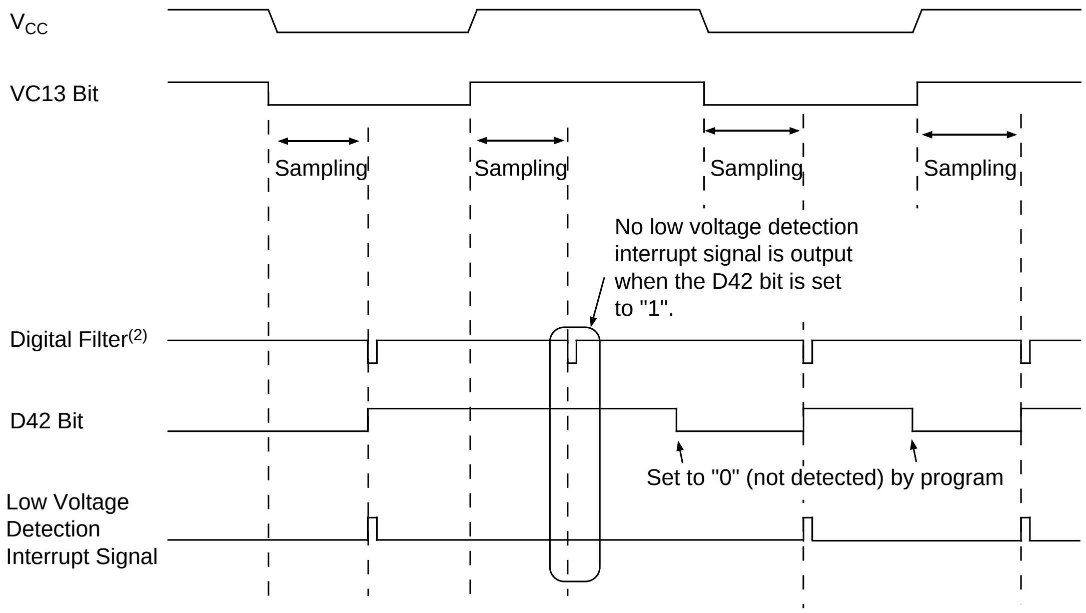
Table 6.2 Sampling Periods

CPU Clock (MHz)	Sampling Clock (µs)			
	Divide-by-8	Divide-by-16	Divide-by-32	Divide-by-64
16	3.0	6.0	12.0	24.0
32	1.5	3.0	6.0	12.0


Low Voltage Detection Interrupt Generation Circuit
DF1, DF0
00_2
01_2
10_2
11_2
Low Voltage Detection Circuit
VC27 bit
CPU Clock
1/8
1/2
1/2
1/2
VC13
V_{CC}
V_{REF}
Noise Rejection
(Rejection Range:200 ns)
Low Voltage Detection Signal(1)
Noise Rejection Circuit
Digital Filter
D42(2)
Watchdog Timer Interrupt Signal
Low Voltage Detection Interrupt Signal
Oscillation Stop Detection Interrupt Signal
Non-Maskable Interrupt Signal
D41
CM10
WAIT Instruction
(Wait Mode)
D40
Watchdog Timer
Underflow Signal from the Watchdog Timer
D43(3)

NOTES:
1. Low voltage detection signal becomes "H" when the VC27 bit in the VCR2 register is set to "0" (disabled).
2. The D42 bit in the D4INT register is set to "0" (not detected) by program. The D42 bit is set to "0" when the VC27 bit is set to "0" (low voltage detection circuit disabled).
3. The D43 bit is set to “0”(not detected) by program.

Figure 6.5 Low Voltage Detection Interrupt Generation Circuit


V_{CC}
VC13 Bit
Sampling
Sampling
Sampling
Sampling
No low voltage detection interrupt signal is output when the D42 bit is set to "1".
Digital Filter(2)
D42 Bit
Set to "0" (not detected) by program
Low Voltage Detection Interrupt Signal

NOTES:
1. This example applies to an operation of the low voltage detection interrupt generation circuit when the D40 bit in the D4INT register is set to "1" (low voltage detection interrupt enabled).
2. Output from the digital filter shown in Figure 6.5.

Figure 6.6 Low Voltage Detection Interrupt Generation Circuit Operation Example

6.1.1 Limitations on Exiting Stop/Wait Mode

The low voltage detection interrupt is generated and the microcomputer exits stop mode as soon as the CM10 bit in the CM1 register is set to "1" (all clocks stopped) under the conditions below. Additionally, if WAIT instruction is executed under these same conditions, the low voltage detection interrupt is immediately generated and the microcomputer exits wait mode.

- the VC27 bit in the VCR2 register is set to "1" (low voltage detection circuit enabled),
- the D40 bit in the D4INT register is set to "1" (low voltage detection interrupt enabled),
- the D41 bit in the D4INT register is set to "1" (low voltage detection interrupt is used to exit stop/wait mode), and
- the voltage applied to the VCC pin is higher than Vdet4 (the VC13 bit in the VCR1 register is set to "1")

Set the CM10 bit to "1" when the VC13 bit is "0" (VCC < Vdet4), if the microcomputer is set to enter stop/wait mode when the voltage applied to the VCC pin drops below Vdet4 and to exit stop/wait mode when the voltage applied rises to Vdet4 or above.

6.2 Cold Start-up / Warm Start-up Determine Function

The WDC5 bit in the WDC register determines either cold start-up, power-on reset, or warm start-up, reset during the microcomputer running. Default value of the WDC5 bit is "0" (cold start-up) when power-on. It is set to "1" (warm start-up) by writing desired values to the WDC register. The WDC5 bit is not reset, regardless of a software reset or reset signal input.

Figure 6.7 shows a block diagram of the cold start-up/warm start-up determine function. Figure 6.8 shows its operation exmaple.

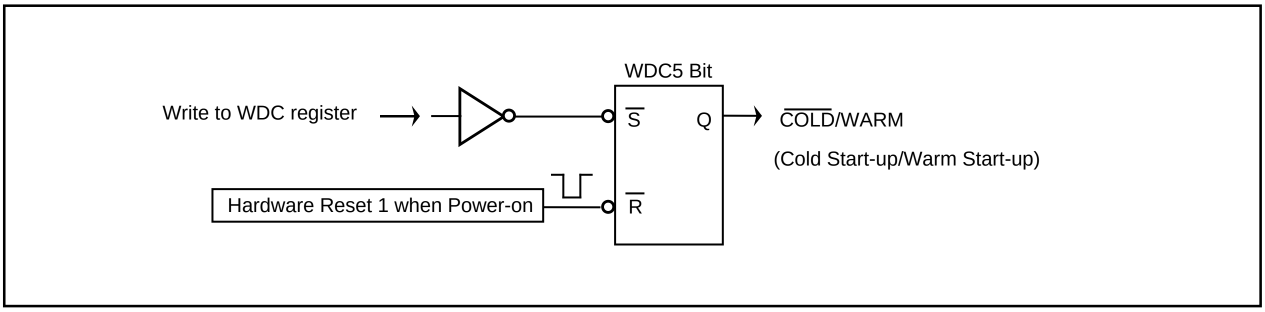
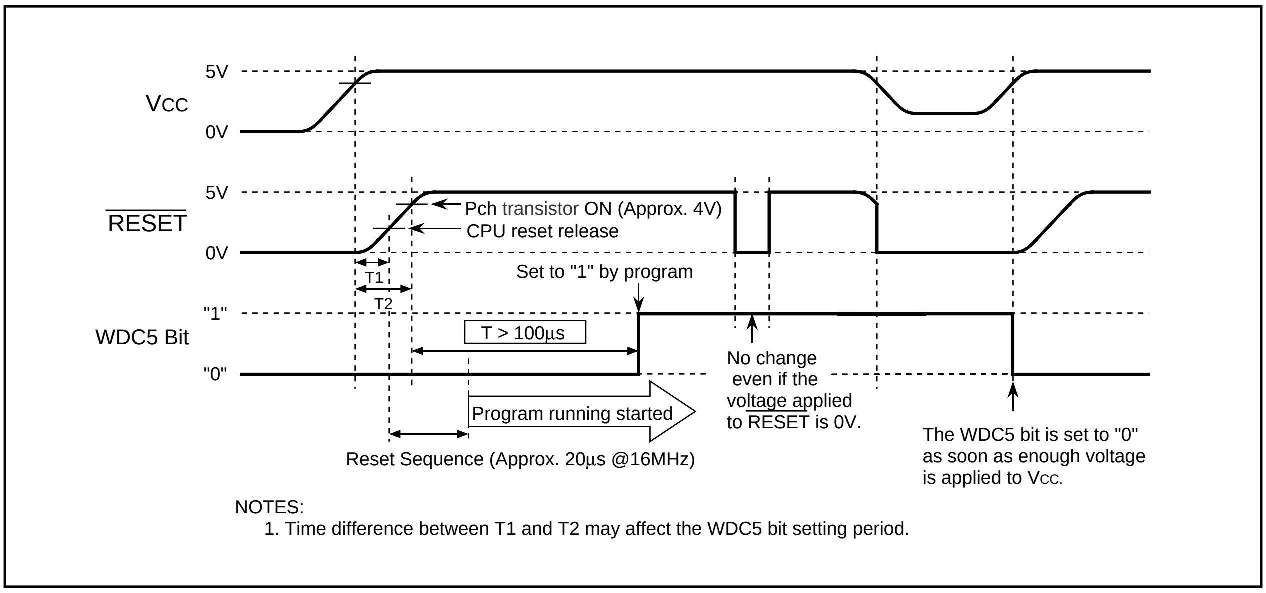


Figure 6.7 Cold Start-up/Warm Start-up Determine Function Block Diagram


5V
VCC
0V
5V
RESET
0V
Pch transistor ON (Approx. 4V)
CPU reset release
T1
T2
Set to "1" by program
"1"
WDC5 Bit
"0"
T > 100µs
No change even if the voltage applied to RESET is 0V.
Program running started
Reset Sequence (Approx. 20µs @16MHz)
The WDC5 bit is set to "0" as soon as enough voltage is applied to VCC.

NOTES:
1. Time difference between T1 and T2 may affect the WDC5 bit setting period.

Figure 6.8 Cold Start-up/Warm Start-up Determine Function Operation

7. Processor Mode

NOTE

Use M32C/86T in single-chip mode only.
M32C/86T cannot be used in memory expansion mode and microprocessor mode.

7.1 Types of Processor Mode

Single-chip mode, memory expansion mode or microprocessor mode can be selected as a processor mode. Table 7.1 lists a feature of the processor mode.

Table 7.1 Processor Mode Feature

Processor Mode	Accessable Space	Pin Status as I/O Ports
Single-chip Mode	SFR, Internal RAM, Internal ROM	All pins assigned to I/O ports or to I/O pins for the peripheral functions
Memory Expansion Mode	SFR, Internal RAM, Internal ROM, External Space(1)	Some pins assigned to bus control pins(1)
Microprocessor Mode	SFR, Internal RAM, External Space(1)	Some pins assigned to bus control pins(1)

NOTES:
1. Refer to **8. Bus** for details.

7.2 Setting of Processor Mode

The CNVSS pin state and the PM01 and PM00 bit settings in the PM0 register determine which processor mode is selected. Table 7.2 lists processor mode after hardware reset. Table 7.3 lists processor mode selected by PM01 and PM00 bit settings.

Table 7.2 Processor Mode after Hardware Reset

Input Level into the CNVSS pin	Processor Mode
VSS	Single-chip Mode
VCC(1, 2)	Microprocessor Mode

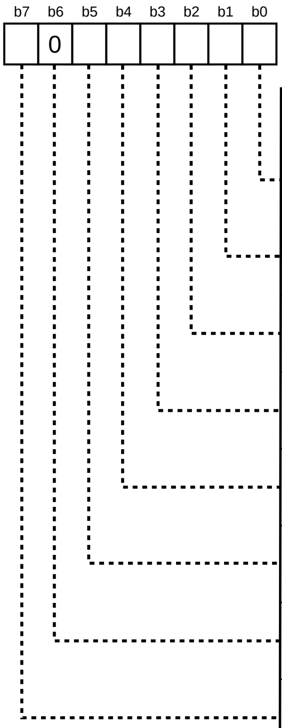
NOTES:
1. The internal ROM cannot be accessed, regardless of PM01 and PM00 bit settings, when applying VCC to the CNVSS pin and generating the hardware reset (hardware reset 1 or brown-out detection reset).
2. Multiplexed bus cannot be assigned to all $\overline{CS}$ areas.

Table 7.3 Processor Mode Selected by the PM01 and PM00 bit Settings

PM01 and PM00 Bits	Processor Mode
00_2	Single-chip Mode
01_2	Memory Expansion Mode
10_2	Do not set to this value
11_2	Microprocessor Mode

If the PM01 and PM00 bits are rewritten, the mode corresponding to the PM01 and PM00 bits is selected regardless of CNVSS pin level.
Do not change the PM01 and PM00 bits to "01_2" (memory expansion mode) or "11_2" (microprocessor mode) when the PM07 to PM02 bits in the PM0 register are being rewritten.
Do not enter microprocessor mode while the CPU is executing a program in the internal ROM.
Do not enter single-chip mode or memory expansion mode from microprocessor mode while the CPU is executing a program in an external memory space, the same address assigned for the internal ROM.
The internal ROM cannot be accessed, regardless of PM01 and PM00 bit settings, when applying VCC to the CNVSS pin and generating the hardware reset (hardware reset 1 or brown-out detection reset).
Figures 7.1 and 7.2 show the PM0 register and PM1 register. Figure 7.3 shows a memory map in each processor mode.

Processor Mode Register 0[1]



Symbol	Address	After Reset
PM0	0004_{16}	$1000\ 0000_2$ (CNVss = "L") $0000\ 0011_2$ (CNVss = "H")

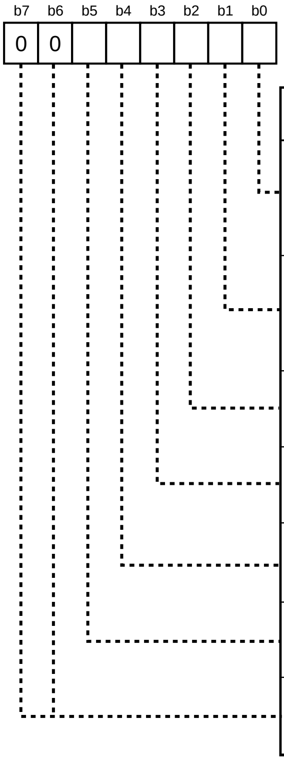
Bit Symbol	Bit Name	Function	RW
PM00	Processor Mode Bit[2, 3]	b1 b0 0 0: Single-chip mode 0 1: Memory expansion mode[8] 1 0: Do not set to this value 1 1: Microprocessor mode[8]	RW
PM01			RW
PM02	R/W Mode Select Bit	0: $\overline{RD}$ / $\overline{BHE}$ / $\overline{WR}$ 1: $\overline{RD}$ / $\overline{WRH}$ / $\overline{WRL}$	RW
PM03	Software Reset Bit	The microcomputer is reset when this bit is set to "1". When read, its content is "0".	RW
PM04	Multiplexed Bus Space Select Bit[4]	b5 b4 0 0 : Multiplexed bus is not used 0 1 : Access the $\overline{CS2}$ area using the bus 0 1 : Access the $\overline{CS1}$ area using the bus 1 1 : Access all $\overline{CS}$ areas using the bus[5]	RW
PM05			RW
——— (b6)	Reserved Bit	Set to "0"	RW
PM07	BCLK Output Disable Bit[6]	0 : BCLK is output[7] 1 : BCLK is not output The CM01 and CM00 bits in the CM0 register determine pin functions	RW

NOTES:

1. Rewrite the PM0 register after the PRC1 bit in the PRCR register is set to "1"(write enable).
2. The PM01 and PM00 bits maintain values set before reset, even after software reset or watchdog timer reset has performed.
3. Set the PM01 and PM00 bits to "01_2" or "11_2" separately. Rewrite other bits before rewriting the PM01 and PM00 bits.
4. The PM04 and PM05 bits are available in memory expansion mode or microprocessor mode.
   - Set the PM05 and PM04 bits to "00_2" in mode 0.
   - Do not set the PM05 and PM04 bits to "01_2" in mode 2.
5. The PM05 and PM04 bits cannot be set to "11_2" in microprocessor mode since the microcomputer starts up with the separate bus after reset.
   When the PM05 and PM04 bits are set to "11_2" in memory expansion mode, the microcomputer can access each 64-Kbyte chip-select-assigned address space. The multiplexed bus is not available in mode 0. The microcomputer accesses the $\overline{CS0}$ to $\overline{CS2}$ in mode 1, $\overline{CS0}$ and $\overline{CS1}$ in mode 2 and $\overline{CS0}$ to $\overline{CS3}$ in mode 3.
6. No BCLK is output in single-chip mode even if the PM07 bit is set to "0". When a clock output is terminated in microprocessor mode or memory expansion mode, set the PM07 bit to "1" and the CM01 and CM00 bits in the CM0 register to "00_2" (I/O port $P5_3$). $P5_3$ outputs "L".
7. When the PM07 bit is set to "0" (BCLK output), set the CM01 and CM00 bits to "00_2".
8. M32C/86T cannot be used in memory expansion mode and microprocessor mode.

Figure 7.1 PM0 Register

Processor Mode Register 1(1)



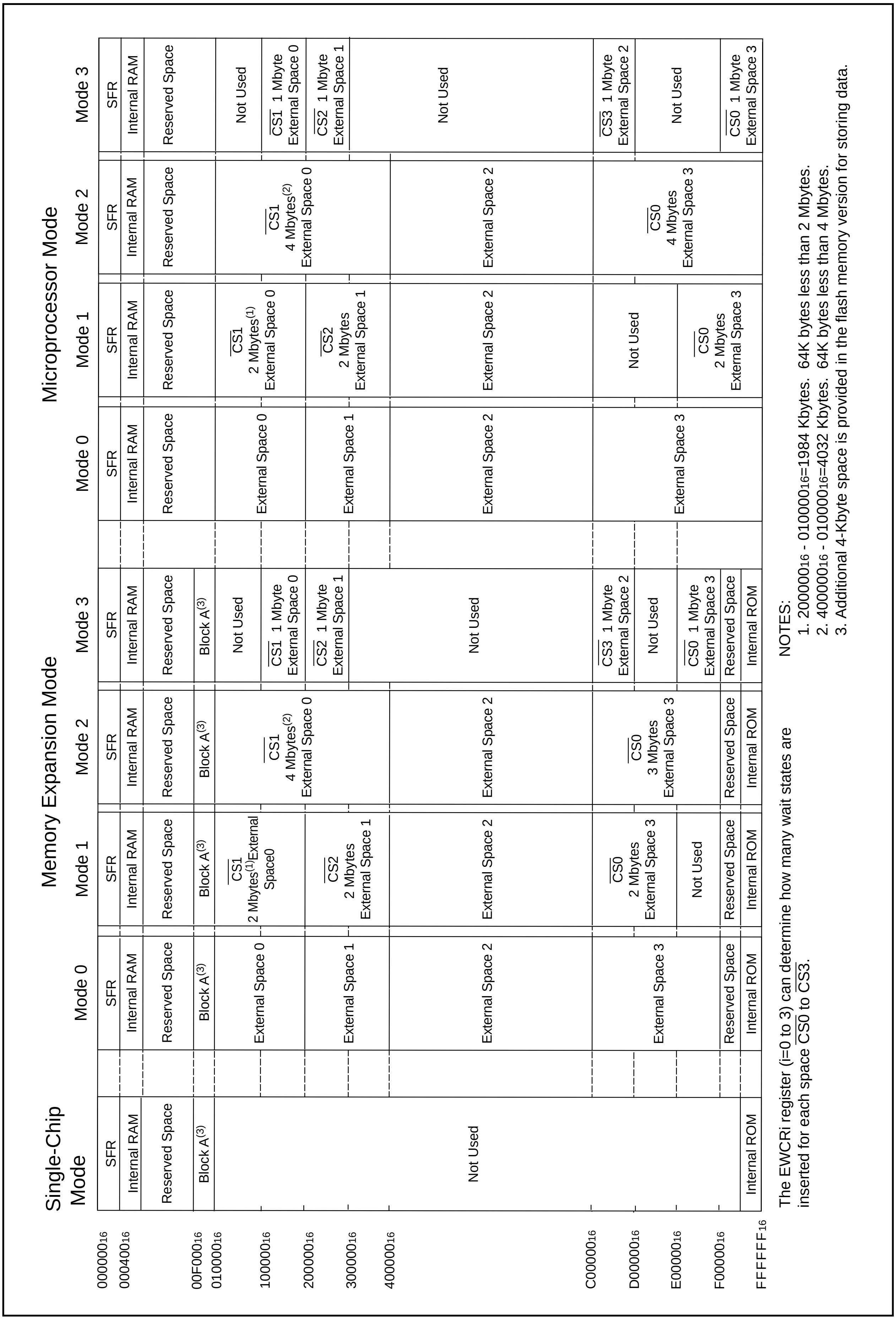
Symbol	Address	After Reset
PM1	0005_{16}	00_{16}

Bit Symbol	Bit Name	Function	RW
PM10	External Memory Space Mode Bit(2, 4)	b1 b0 0 0 : Mode 0 (A_{20} to $\overline{A_{23}}$ for $P4_4$ to $P4_7$) 0 1 : Mode 1 (A_{20} for $P4_4$, $\overline{CS2}$ to $\overline{CS0}$ for $P4_5$ to $P4_7$) 1 0 : Mode 2 (A_{20}, A_{21} for $P4_4$, $P4_5$, $\overline{CS1}$, $\overline{CS0}$ for $P4_6$, $P4_7$) 1 1 : Mode 3 ($\overline{CS3}$ to $\overline{CS0}$ for $P4_4$ to $P4_7$)	RW
PM11			RW
PM12	Internal Memory Wait Bit	0 : No wait state 1 : Wait state	RW
PM13	SFR Area Wait Bit	0 : 1 wait state 1 : 2 Wait states	RW
PM14	ALE Pin Select Bit(2, 4)	b5 b4 0 0 : No ALE 0 1 : $P5_3$/BCLK(3) 1 0 : $P5_6$ 1 1 : $P5_4$/$\overline{HLDA}$	RW
PM15			RW
— (b7-b6)	Reserved Bit	Set to "0"	RW

NOTES:

1. Rewrite the PM1 register after the PRC1 bit in the PRCR register is set to "1" (write enable).
2. The PM15 and PM14 bit setting, PM11 and PM10 bit setting are available in memory expansion mode or microprocessor mode.
3. Set the CM01 and CM00 bits in the CM0 register to "00_2" (I/O port $P5_3$) when the PM15 and PM14 bits are set to "01_2" ($P5_3$/BCLK select).
4. M32C/86T cannot be used in memory expansion mode and microprocessor mode.

Figure 7.2 PM1 Register


Single-Chip Mode
Memory Expansion Mode
Microprocessor Mode
Mode 0
Mode 1
Mode 2
Mode 3
00000016
00040016
00F00016
01000016
10000016
20000016
30000016
40000016
C0000016
D0000016
E0000016
F0000016
FFFFFF16
SFR
Internal RAM
Reserved Space
Block A(3)
Not Used
Internal ROM
External Space 0
External Space 1
External Space 2
External Space 3
CS1 2 Mbytes(1) External Space0
CS2 2 Mbytes External Space 1
CS0 2 Mbytes External Space 3
CS1 4 Mbytes(2) External Space 0
CS0 3 Mbytes External Space 3
CS1 1 Mbyte External Space 0
CS2 1 Mbyte External Space 1
CS3 1 Mbyte External Space 2
CS0 1 Mbyte External Space 3
CS1 2 Mbytes(1) External Space 0
CS0 4 Mbytes External Space 3

The EWCRi register (i=0 to 3) can determine how many wait states are inserted for each space CS0 to CS3.

NOTES:
1. 20000016 - 01000016=1984 Kbytes. 64K bytes less than 2 Mbytes.
2. 40000016 - 01000016=4032 Kbytes. 64K bytes less than 4 Mbytes.
3. Additional 4-Kbyte space is provided in the flash memory version for storing data.

Figure 7.3 Memory Map in Each Processor Mode

8. Bus

In memory expansion mode or microprocessor mode, some pins function as bus control pins to control the address bus and data bus. A_0 to A_{22}, $\overline{A_{23}}$, D_0 to D_{15}, $\overline{CS0}$ to $\overline{CS3}$, $\overline{WRL}/\overline{WR}$, $\overline{WRH}/\overline{BHE}$, $\overline{RD}$, BCLK/ALE, $\overline{HLDA}$/ALE, $\overline{HOLD}$, ALE, $\overline{RDY}$ are used as bus control pins.

NOTE

Bus control pins in M32C/86T cannot be used.

8.1 Bus Settings

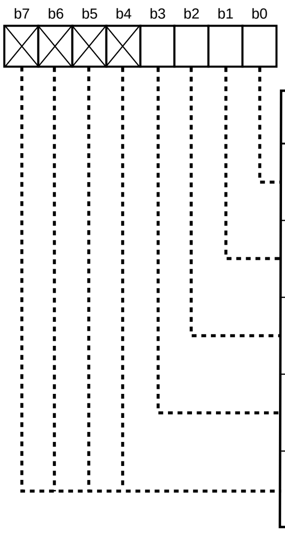
The BYTE pin, the DS register, the PM05 and PM04 bits in the PM0 register and the PM11 and PM10 bits in the PM1 register determine bus settings.

Table 8.1 lists how to change bus settings. Figure 8.1 shows the DS register.

Table 8.1 Bus Settings

Bus Setting	Changed By
Selecting External Address Bus Width	DS register
Setting Bus Width after Reset	BYTE pin (external space 3 only)
Selecting Between Separate Bus or Multiplexed Bus	PM05 and PM04 bits in PM0 register
Number of Chip-Select	PM11 and PM10 bits in PM1 register

External Data Bus Width Control Register



Symbol	Address	After Reset
DS	$000B_{16}$	XXXX 1000_2 (BYTE pin = "L") XXXX 0000_2 (BYTE pin = "H")

Bit Symbol	Bit Name	Function	RW
DS0	External Space 0 Data Bus Width Select Bit	0 : 8 bits wide 1 : 16 bits wide	RW
DS1	External Space 1 Data Bus Width Select Bit	0 : 8 bits wide 1 : 16 bits wide	RW
DS2	External Space 2 Data Bus Width Select Bit	0 : 8 bits wide 1 : 16 bits wide	RW
DS3	External Space 3 Data Bus Width Select Bit(1)	0 : 8 bits wide 1 : 16 bits wide	RW
—— (b7 - b4)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		—

NOTES:
1. The DS register in M32C/86T cannot be used.

Figure 8.1 DS Register

8.1.1 Selecting External Address Bus

The number of externally-output address buses, the number of chip-select signals and chip-select-assigned address space ($\overline{CS}$ area) vary depending on each external space mode. The PM11 and PM10 bits in the PM1 register determine the external space mode.

8.1.2 Selecting External Data Bus

The DS register selects either external 8-bit or 16-bit data bus per external space. The data bus in the external space 3, after reset, becomes 16 bits wide when a low-level ("L") signal is applied to the BYTE pin and 8 bits wide when a high-level ("H") signal is applied. Keep the BYTE pin input level while the microcomputer is operating. Internal bus is always 16 bits wide.

8.1.3 Selecting Separate/Multiplexed Bus

The PM05 and PM04 bits in the PM0 register determine either separate or multiplexed bus as bus format.

8.1.3.1 Separate Bus

The separate bus is a bus format which allows the microcomputer to input and output data and address separatelly. The DS register selects 8-bit or 16-bit data bus as the external data bus per external space. If all DSi bits in the DS register (i=0 to 3) are set to "0" (8-bit data bus), port P0 becomes the data bus and port P1, the programmable I/O port. If one of the DSi bits is set to "1" (16-bit data bus), ports P0 and P1 become the data bus. Port P1 is indeterminate when the microcomputer accesses a space where the DSi bit is set to "0".

The EWCRi register (i=0 to 3) determines the number of software wait states inserted, when the microcomputer accesses space using the separate bus.

8.1.3.2 Multiplexed Bus

The multiplexed bus is a bus format which allow the microcomputer to input and output data and address by timesharing. D_0 to D_7 are multiplexed with A_0 to A_7 in space accessed by the 8-bit data bus. D_0 to D_{15} are multiplexed with A_0 to A_{15} in space accessed by the 16-bit data bus. The DSi bit controls the data bus width. The EWCRi register (i=0 to 3) controls the number of software wait states inserted, when the microcomputer accesses a space using the multiplexed bus. Refer to **8.2.4 Bus Timing** for details.

The multiplexed bus can be assigned to access the $\overline{CS1}$ area, $\overline{CS2}$ area or all $\overline{CS}$ areas. However, because the microcomputer starts operation using the separate bus after reset, the multiplexed bus cannot be assigned to access all $\overline{CS}$ areas in microprocessor mode. When the PM05 and PM04 bits in the PM0 register are set to "11_2" (access all $\overline{CS}$ areas with the bus), 16 low-order bits, from A_0 to A_{15}, of an address are output. See **Table 8.2** for details.

Table 8.2 Processor Mode and Port Function

Processor Mode	Single-Chip Mode	Memory Expansion Mode/ Microprocessor Mode				Memory Expansion Mode	
PM05 to PM04 Bits in PM0 Register		"01_2", "10_2" (Access $\overline{CS1}$ or $\overline{CS2}$ using the Multiplexed Bus Access All Other $\overline{CS}$ Areas using the Separate Bus)		"00_2" (Access all $\overline{CS}$ Areas using the Separate Bus)		"11_2"(1) (Access all $\overline{CS}$ Areas using the Multiplexed Bus)	
Data Bus Width		Access all external space with 8-bit data bus	Access one or more external space with 16-bit data bus	Access all external space with 8-bit data bus	Access one or more external space with 16-bit data bus	Access all external space with 8-bit data bus	Access one or more external space with 16-bit data bus
$P0_0$ to $P0_7$	I/O port	Data bus D_0 to D_7	Data bus D_0 to D_7	Data bus D_0 to D_7	Data bus D_0 to D_7	I/O port	I/O port
$P1_0$ to $P1_7$	I/O port	I/O port	Data bus D_8 to D_{15}	I/O port	Data bus D_8 to D_{15}	I/O port	I/O port
$P2_0$ to $P2_7$	I/O port	Address bus Data bus(2) A_0/D_0 to A_7/D_7	Address bus Data bus(2) A_0/D_0 to A_7/D_7	Address bus A_0 to A_7	Address bus A_0 to A_7	Address bus Data bus A_0/D_0 to A_7/D_7	Address bus Data bus A_0/D_0 to A_7/D_7
$P3_0$ to $P3_7$	I/O port	Address bus A_8 to A_{15}	Address bus/ Data bus(2) A_8/D_8 to A_{15}/D_{15}	Address bus A_8 to A_{15}	Address bus A_8 to A_{15}	Address bus A_8 to A_{15}	Address bus/ Data bus A_8/D_8 to A_{15}/D_{15}
$P4_0$ to $P4_3$	I/O port	Address bus A_{16} to A_{19}	Address bus A_{16} to A_{19}	Address bus A_{16} to A_{19}	Address bus A_{16} to A_{19}	I/O port	I/O port
$P4_4$ to $P4_6$	I/O port	$\overline{CS}$ (Chip-select signal) or Address bus (A_{20} to A_{22}) (Refer to 8.2 Bus Control for details)(4)					
$P4_7$	I/O port	$\overline{CS}$ (Chip-select signal) or Address bus ($\overline{A_{23}}$) (Refer to 8.2 Bus Control for details)(4)					
$P5_0$ to $P5_3$	I/O port	Outputs $\overline{RD}$, $\overline{WRL}$, $\overline{WRH}$ and BCLK or outputs $\overline{RD}$, $\overline{BHE}$, $\overline{WR}$ and BCLK (Refer to 8.2 Bus Control for details)(3)					
$P5_4$	I/O port	$\overline{HDLA}$ (3)	$\overline{HDLA}$ (3)	$\overline{HDLA}$ (3)	$\overline{HDLA}$ (3)	$\overline{HDLA}$ (3)	$\overline{HDLA}$ (3)
$P5_5$	I/O port	$\overline{HOLD}$	$\overline{HOLD}$	$\overline{HOLD}$	$\overline{HOLD}$	$\overline{HOLD}$	$\overline{HOLD}$
$P5_6$	I/O port	ALE (3)	ALE (3)	ALE (3)	ALE (3)	ALE (3)	ALE (3)
$P5_7$	I/O port	$\overline{RDY}$	$\overline{RDY}$	$\overline{RDY}$	$\overline{RDY}$	$\overline{RDY}$	$\overline{RDY}$

NOTES:

1. The PM05 and PM04 bits cannot be set to "11_2" (access all $\overline{CS}$ areas using multiplexed bus) in microprocessor mode because the microcomputer starts operation using the separate bus after reset.
   When the PM05 and PM04 bits are set to "11_2" in memory expansion mode, the microcomputer accesses 64-Kbyte memory space per chip-select using the address bus .
2. These ports become address buses when accessing space using the separate bus.
3. The PM15 and PM14 bits in the PM1 register determines which pin outputs the ALE signal. The PM02 bit in the PM0 register selects either "$\overline{WRL}$,$\overline{WRH}$" or "$\overline{BHE}$,$\overline{WR}$" combination.
   $P5_6$ provides an indeterminate output when the PM15 and PM14 bits to "00_2" (no ALE). It cannot be used as an I/O port.
4. The PM11 and PM10 bits in the PM1 register determine the $\overline{CS}$ signal and address bus.

8.2 Bus Control

Signals, required to access external devices, are provided and software wait states are inserted as follows. The signals are available in memory expansion mode and microprocessor mode only.

8.2.1 Address Bus and Data Bus

Address bus is a signal accessing 16-Mbyte space and uses 24 control pins; A_0 to A_{22} and $\overline{A_{23}}$. $\overline{A_{23}}$ is the inversed output signal of the highest-order address bit.

Data bus is a signal for data input and output. The DS register selects an 8-bit data bus from D_0 to D_7 or a 16-bit data bus from D_0 to D_{15} for each external space. When applying a high-level ("H") signal to the BYTE pin, the data bus accessing the external memory space 3 becomes an 8-bit data bus after reset. When applying a low-level ("L") signal to the BYTE pin, the data bus accessing the external memory space 3 becomes the 16-bit data bus.

When changing single-chip mode to memory expansion mode, the address bus is in an indeterminate state until the microcomputer accesses an external memory space.

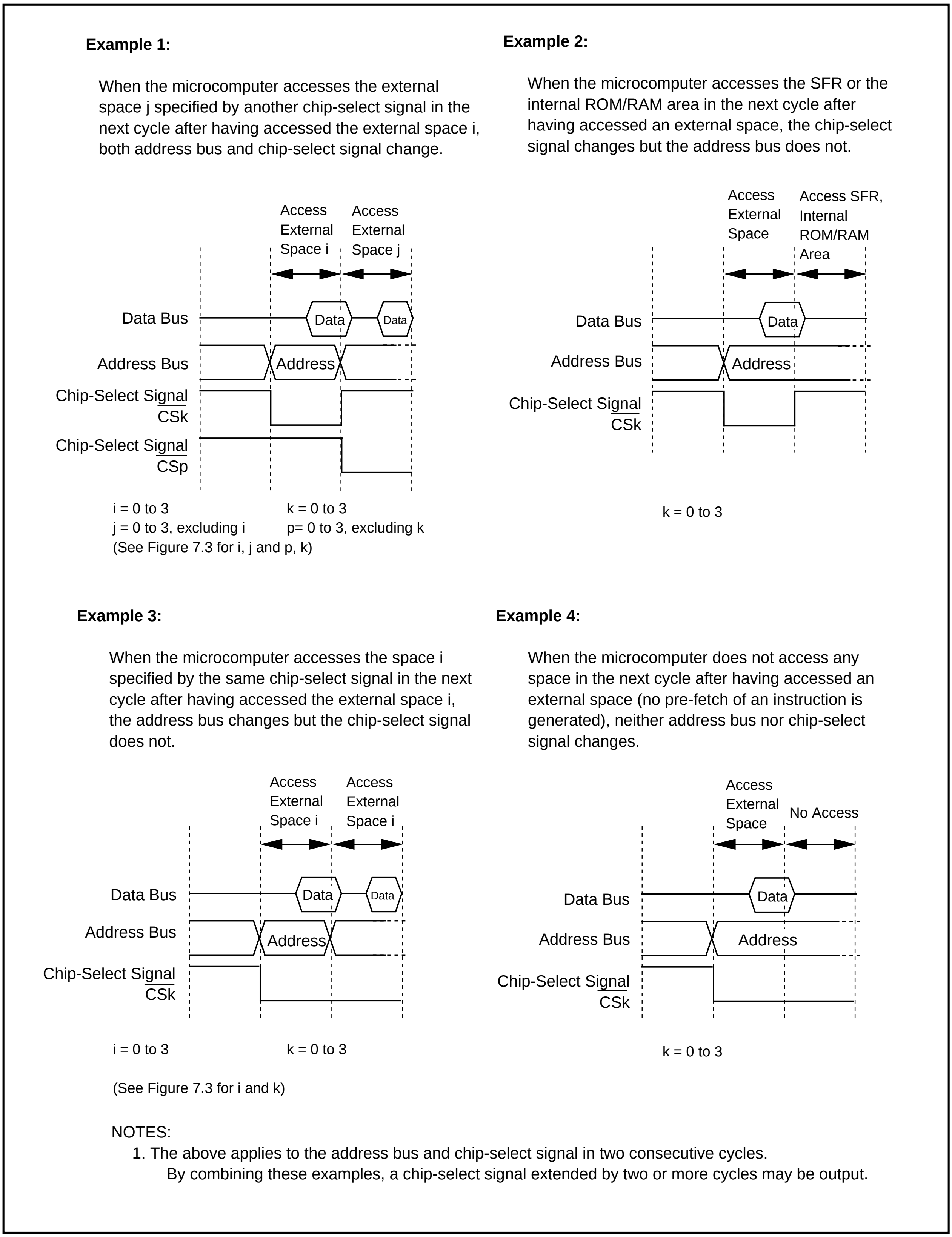
8.2.2 Chip-Select Signal

Chip-select signal shares pins with A_{20} to A_{22} and $\overline{A_{23}}$. The PM11 and PM10 bits in the PM1 register determine which $\overline{CS}$ area is accessed and how many chip-select signals are output. A maximum of four chip-select signals can be output.

In microprocessor mode, no chip-select signal, aside from $\overline{A_{23}}$ which can perform as a chip-select signal, is output after reset.

The chip-select signal becomes "L" while the microcomputer is accessing the external $\overline{CSi}$ area (i=0 to 3). It becomes "H" while the microcomputer is accessing other external memory space.

Figure 8.2 shows an example of the address bus and chip-select signal output.



Example 1:

When the microcomputer accesses the external space j specified by another chip-select signal in the next cycle after having accessed the external space i, both address bus and chip-select signal change.

i = 0 to 3
j = 0 to 3, excluding i
(See Figure 7.3 for i, j and p, k)
k = 0 to 3
p= 0 to 3, excluding k

Example 2:

When the microcomputer accesses the SFR or the internal ROM/RAM area in the next cycle after having accessed an external space, the chip-select signal changes but the address bus does not.

k = 0 to 3

Example 3:

When the microcomputer accesses the space i specified by the same chip-select signal in the next cycle after having accessed the external space i, the address bus changes but the chip-select signal does not.

i = 0 to 3
k = 0 to 3

(See Figure 7.3 for i and k)

Example 4:

When the microcomputer does not access any space in the next cycle after having accessed an external space (no pre-fetch of an instruction is generated), neither address bus nor chip-select signal changes.

k = 0 to 3

NOTES:
1. The above applies to the address bus and chip-select signal in two consecutive cycles.
By combining these examples, a chip-select signal extended by two or more cycles may be output.

Figure 8.2 Address Bus and Chip-Select Signal Outputs (Separate Bus)

8.2.3 Read and Write Signals

When using a16-bit data bus, the PM02 bit in the PM0 register selects a combination of the "$\overline{\text{RD}}$, $\overline{\text{WR}}$ and $\overline{\text{BHE}}$" signals or the "$\overline{\text{RD}}$, $\overline{\text{WRL}}$ and $\overline{\text{WRH}}$" signals to determine the read or write signal. When the DS3 to DS0 bits in the DS register are set to "0" (8-bit data bus), set the PM02 bit to "0" ($\overline{\text{RD}}$/$\overline{\text{WR}}$/$\overline{\text{BHE}}$). When any of the DS3 to DS0 bits are set to "1" (16-bit data bus) to access an 8-bit space, the combination of "$\overline{\text{RD}}$, $\overline{\text{WR}}$ and $\overline{\text{BHE}}$" is automatically selected regardless of the PM02 bit setting. Tables 8.3 and 8.4 list each signal operation.

The $\overline{\text{RD}}$, $\overline{\text{WR}}$ and $\overline{\text{BHE}}$ signals are combined for the read or write signal after reset.

When changing the combination of "$\overline{\text{RD}}$, $\overline{\text{WRL}}$ and $\overline{\text{WRH}}$", set the PM02 bit first to write data to an external memory.

Table 8.3 $\overline{\text{RD}}$, $\overline{\text{WRL}}$ and $\overline{\text{WRH}}$ Signals

Data Bus	$\overline{\text{RD}}$	$\overline{\text{WRL}}$	$\overline{\text{WRH}}$	Status of External Data Bus
16 Bits	L	H	H	Read data
	H	L	H	Write 1-byte data to even address
	H	H	L	Write 1-byte data to odd address
	H	L	L	Write data to both even and odd addresses
8 Bits	H	L(1)	Not used	Write 1-byte data
	L	H(1)	Not used	Read 1-byte data

NOTES:

1. The $\overline{\text{WR}}$ signal is used instead of the $\overline{\text{WRL}}$ signal.

Table 8.4 $\overline{\text{RD}}$, $\overline{\text{WR}}$ and $\overline{\text{BHE}}$ Signals

Data Bus	$\overline{\text{RD}}$	$\overline{\text{WR}}$	$\overline{\text{BHE}}$	A0	Status of External Data Bus
16 Bits	H	L	L	H	Write 1-byte data to odd address
	L	H	L	H	Read 1-byte data from odd address
	H	L	H	L	Write 1-byte data to even address
	L	H	H	L	Read 1-byte data from even address
	H	L	L	L	Write data to both even and odd addresses
	L	H	L	L	Read data from both even and odd addresses
8 Bits	H	L	Not used	H / L	Write 1-byte data
	L	H	Not used	H / L	Read 1-byte data

8.2.4 Bus Timing

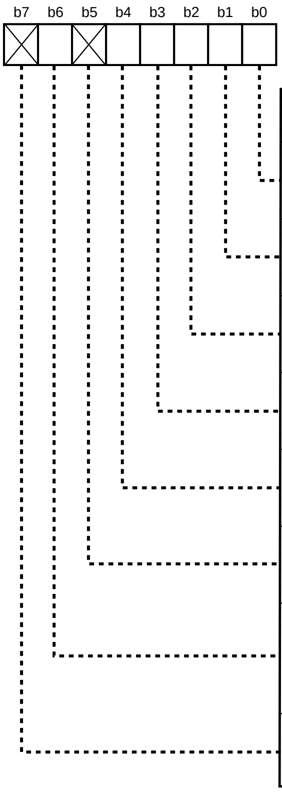
Bus cycle for the internal ROM and internal RAM is basically one BCLK cycle. When the PM12 bit in the PM1 register is set to "1" (wait state), the bus cycles are two BCLK cycles.

Bus cycles for the SFR are basically two BCLK cycles. When the PM13 bit in the PM1 register is set to "1" (2 wait states), the bus cycles are three BCLK cycles.

Basic bus cycle for an external space is 2ø (1ø+1ø) to read and to write. Bus cycle is selected by the EWCRi register (i=0 to 3) from 12 types of separate bus settings and 7 types of multiplexed bus settings. If the EWCRi04 to EWCRi00 bits are set to "00011_2" (1ø+3ø), bus cycles are four BCLK cycles.

Figure 8.3 shows the EWCRi register. Figures 8.4 to 8.8 show bus timing in an external space.

External Space Wait Control Register i (i=0 to 3)[3]



Symbol	Address	After Reset
EWCR0 to EWCR3	0048_{16}, 0049_{16}, $004A_{16}$, $004B_{16}$	$X0X0\ 0011_2$

Bit Symbol	Bit Name	Function	RW
EWCRi00	Bus Cycle Select Bit	b4 b3 b2 b1 b0 (1) (2) 0 0 0 0 1: 1ϕ + 1ϕ 0 0 0 1 0: 1ϕ + 2ϕ 0 0 0 1 1: 1ϕ + 3ϕ 0 0 1 0 0: 1ϕ + 4ϕ 0 0 1 0 1: 1ϕ + 5ϕ 0 0 1 1 0: 1ϕ + 6ϕ 0 1 0 1 0: 2ϕ + 2ϕ 0 1 0 1 1: 2ϕ + 3ϕ 0 1 1 0 0: 2ϕ + 4ϕ 0 1 1 0 1: 2ϕ + 5ϕ 1 0 0 1 1: 3ϕ + 3ϕ 1 0 1 0 0: 3ϕ + 4ϕ 1 0 1 0 1: 3ϕ + 5ϕ 1 0 1 1 0: 3ϕ + 6ϕ Do not set values other than the above	RW
EWCRi01			RW
EWCRi02			RW
EWCRi03			RW
EWCRi04			RW
(b5)	Nothing is assigned. When read, its content is indeterminate.		–
EWCRi06	Recovery Cycle Addition Select Bit	0 : Adds no recovery cycle when accessing external space i 1 : Adds a recovery cycle when accessing external space i	RW
(b7)	Nothing is assigned. When read, its content is indeterminate.		–

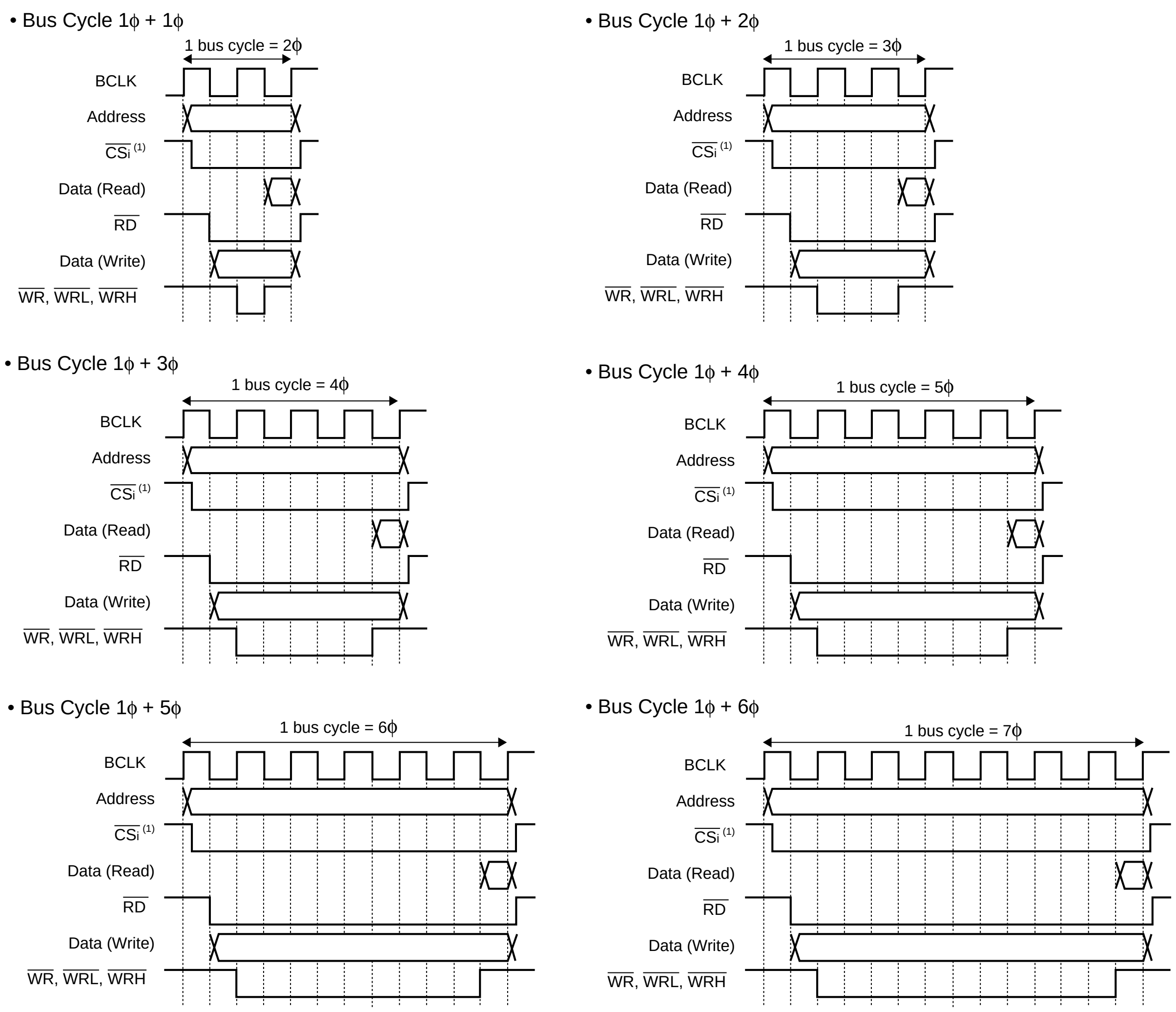
NOTES:

1. The number of bus cycles from "when bus access begins" to "when RD or WR signal becomes "L".
2. The number of bus cycles from "when RD or WR signal becomes "L" to "when it becomes "H".
3. The EWCR0 to EWCR3 registers in M32C/86T cannot be used.

Figure 8.3 EWCR0 to EWCR3 Registers

Table 8.5 Software Wait State and Bus Cycle

Space	External Bus Status	PM1 Register		EWCRi Register (i=0 to 3)	Bus Cycles
		PM13 Bit	PM12 Bit	EWCRi04 to EWCRi00 Bits	
SFR	---	0	---	---	2 BCLK cycles
		1			3 BCLK cycles
Internal ROM/RAM	---	---	0	--	1 BCLK cycles
			1		2 BCLK cycles
External Memory	Separate Bus	---	---	00001_2	2 BCLK cycles
				00010_2	3 BCLK cycles
				00011_2	4 BCLK cycles
				00100_2	5 BCLK cycles
				00101_2	6 BCLK cycles
				00110_2	7 BCLK cycles
				01010_2	4 BCLK cycles
				01011_2	5 BCLK cycles
				01100_2	6 BCLK cycles
				10011_2	6 BCLK cycles
				10100_2	7 BCLK cycles
				10110_2	9 BCLK cycles
	Multiplexed Bus	---	---	01010_2	4 BCLK cycles
				01011_2	5 BCLK cycles
				01101_2	7 BCLK cycles
				10011_2	6 BCLK cycles
				10100_2	7 BCLK cycles
				10101_2	8 BCLK cycles
				10110_2	9 BCLK cycles

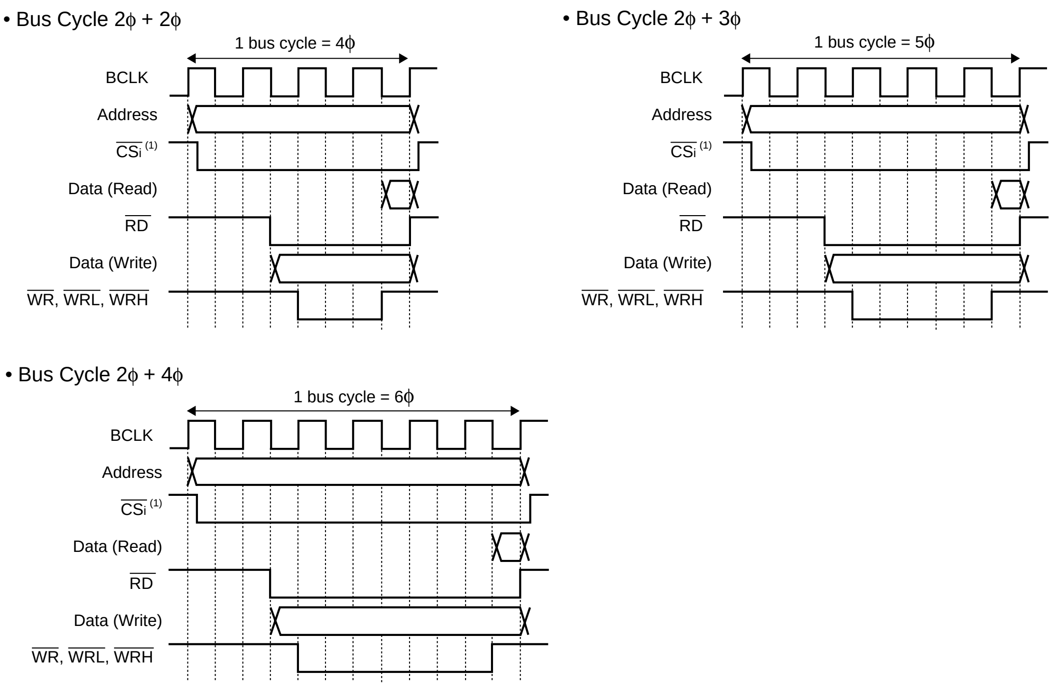


i=0 to 3

NOTES:

1. When the microcomputer continuously accesses the same $\overline{\text{CS}}$ area, the $\overline{\text{CS}}$i pin provides an "L" signal continuously.

Figure 8.4 Bus Cycle with Separate Bus (1)

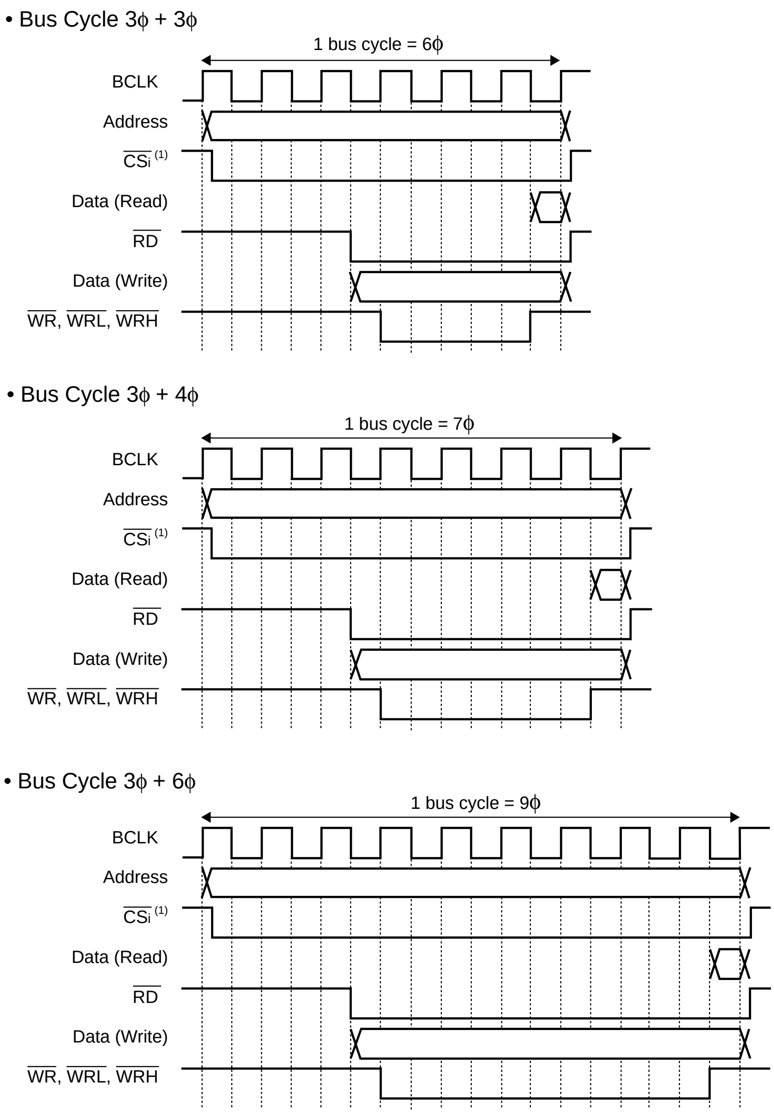


i=0 to 3

NOTES:

1. When the microcomputer continuously accesses the same $\overline{\text{CS}}$ area, the $\overline{\text{CSi}}$ pin provides an "L" signal continuously.

Figure 8.5 Bus Cycle with Separate Bus (2)

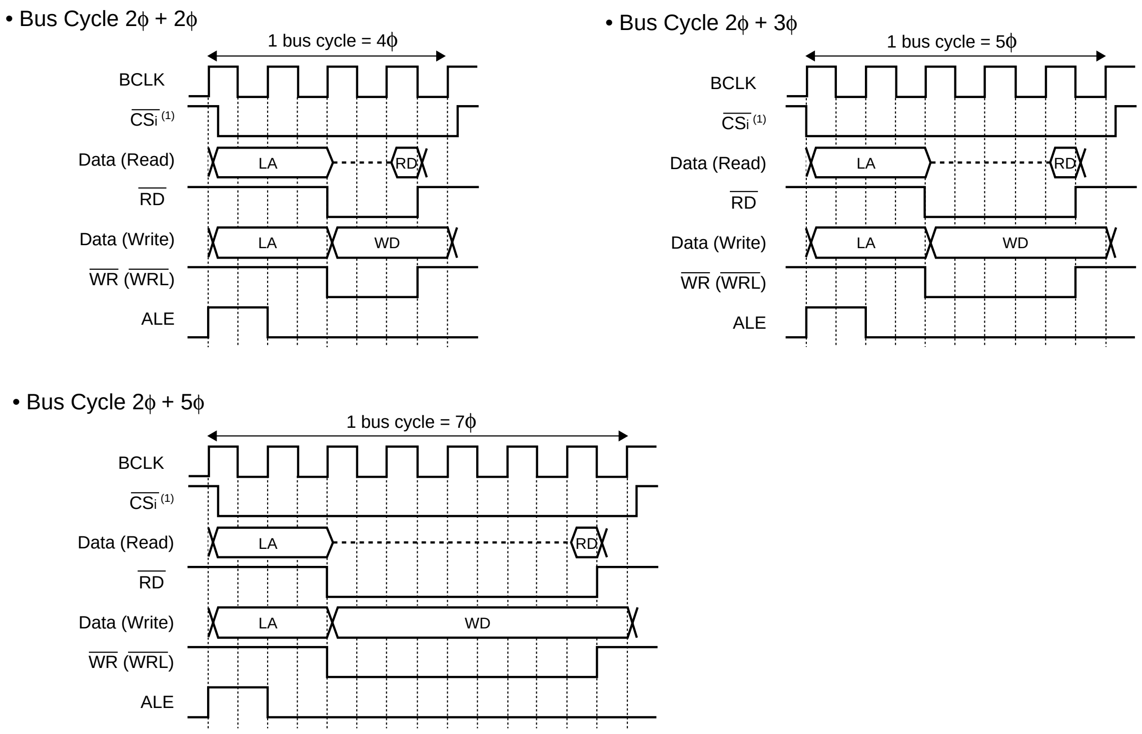

• Bus Cycle 3ϕ + 3ϕ
1 bus cycle = 6ϕ
BCLK
Address
$\overline{\text{CSi}}$ (1)
Data (Read)
$\overline{\text{RD}}$
Data (Write)
$\overline{\text{WR}}$, $\overline{\text{WRL}}$, $\overline{\text{WRH}}$
• Bus Cycle 3ϕ + 4ϕ
1 bus cycle = 7ϕ
BCLK
Address
$\overline{\text{CSi}}$ (1)
Data (Read)
$\overline{\text{RD}}$
Data (Write)
$\overline{\text{WR}}$, $\overline{\text{WRL}}$, $\overline{\text{WRH}}$
• Bus Cycle 3ϕ + 6ϕ
1 bus cycle = 9ϕ
BCLK
Address
$\overline{\text{CSi}}$ (1)
Data (Read)
$\overline{\text{RD}}$
Data (Write)
$\overline{\text{WR}}$, $\overline{\text{WRL}}$, $\overline{\text{WRH}}$

i=0 to 3

NOTES:

1. When the microcomputer continuously accesses the same $\overline{\text{CS}}$ area, the $\overline{\text{CSi}}$ pin provides an "L" signal continuously.

Figure 8.6 Bus Cycle with Separate Bus (3)



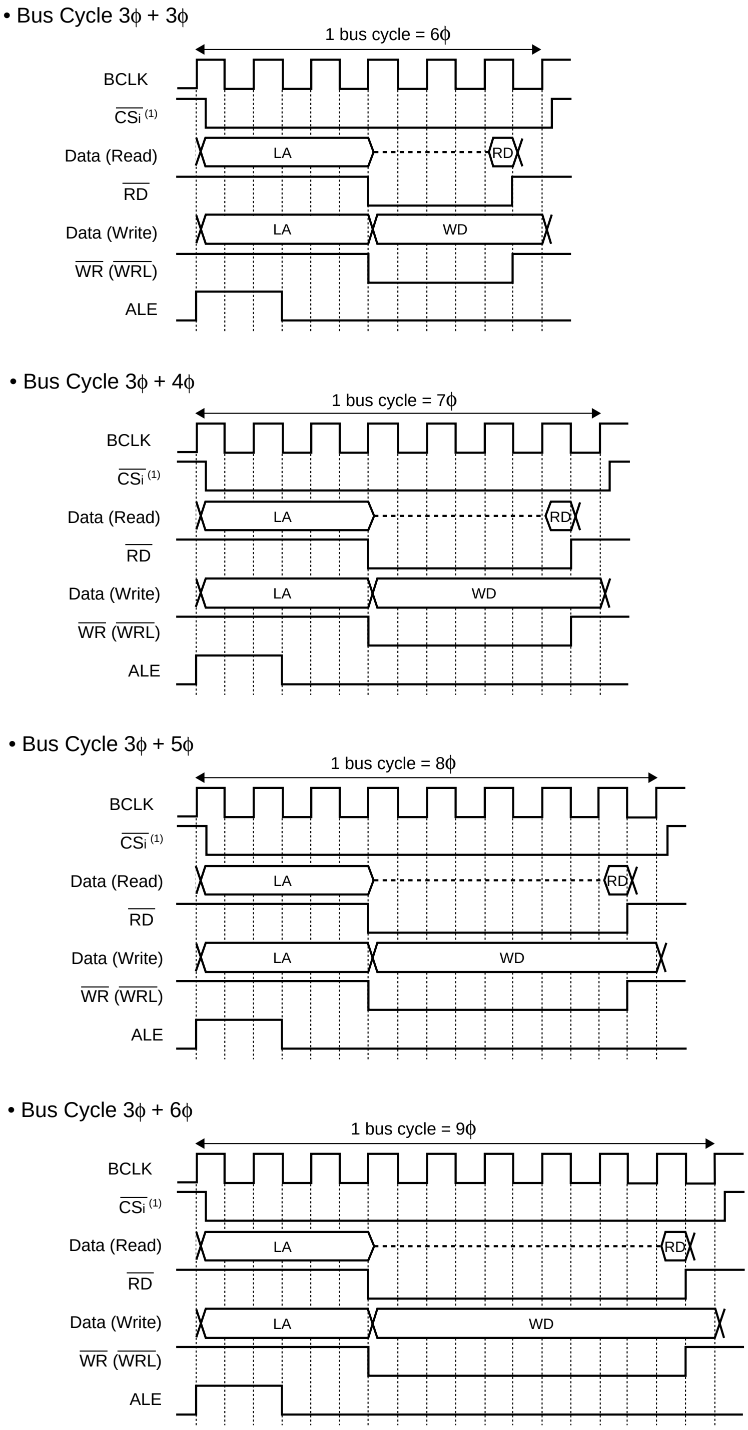
LA : Latch Address RD : ReadData WD : Write Data

i=0 to 3

NOTES:

1. When the microcomputer continuously accesses the same $\overline{\text{CS}}$ area, the $\overline{\text{CSi}}$ pin provides an "L" signal continuously.

Figure 8.7 Bus Cycle with Multiplexed Bus (1)



LA : Latch Address RD : Read Data WD : Write Data

i=0 to 3

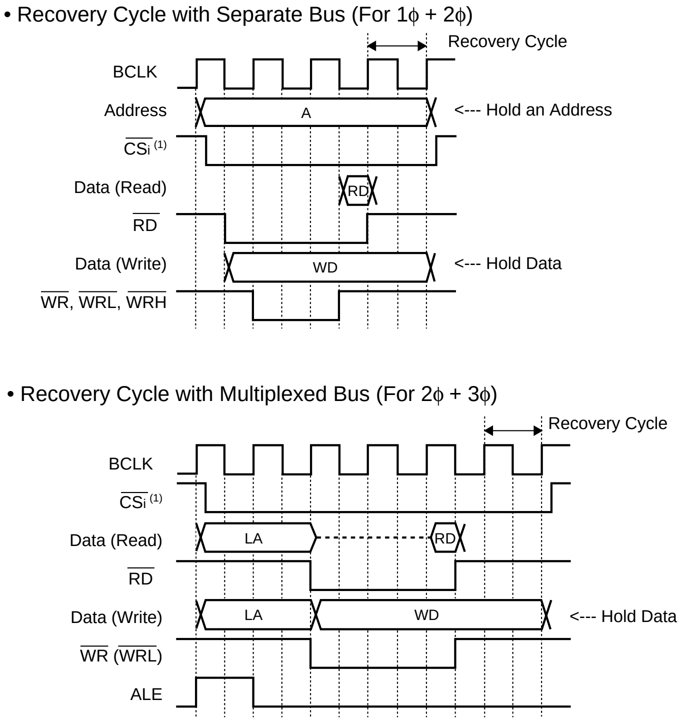
NOTES:

1. When the microcomputer continuously accesses the same $\overline{\text{CS}}$ area, the $\overline{\text{CSi}}$ pin provides an "L" signal continuously.

Figure 8.8 Bus Cycle with Multiplexed Bus (2)

8.2.4.1 Bus Cycle with Recovery Cycle Added

The EWCRi06 bit in the EWCRi register (i=0 to 3) determines whether the recovery cycle is added or not. In the recovery cycle, addresses and wrie data outputs are provided continuously (using the separate bus only). Devices, which take longer address hold time and data hold time to write data, are connectable.



A : Address LA : Latch Address RD : Read Data WD : Write Data

i=0 to 3

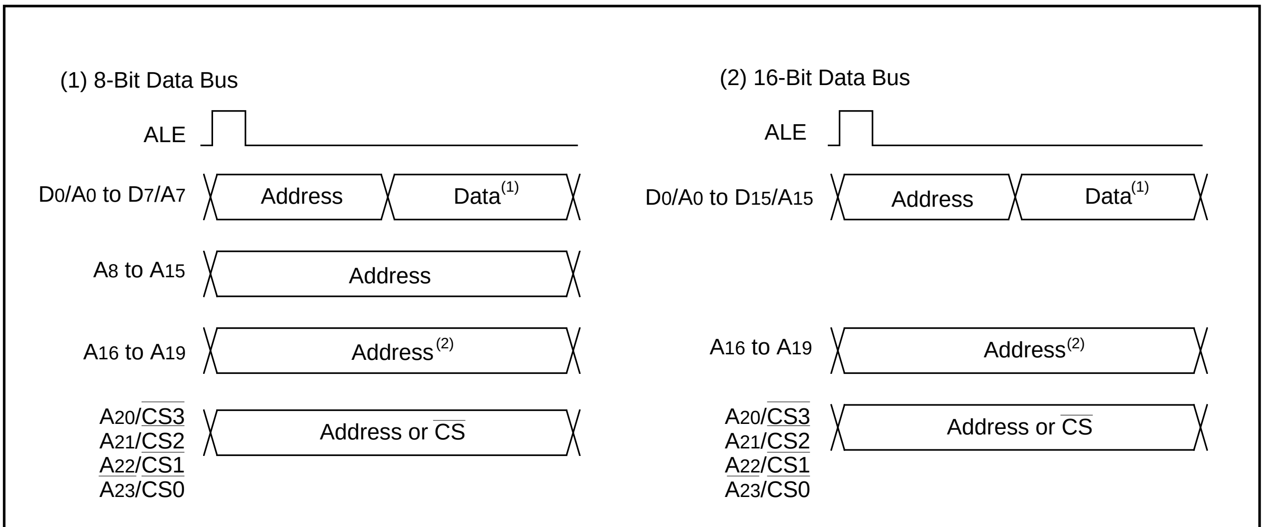
NOTES:

1. When the microcomputer continuously accesses the same CS area, the CSi pin provides an "L" signal continuously.

Figure 8.9 Recovery Cycle

8.2.5 ALE Signal

The ALE signal latches an address of the multiplexed bus. Latch an address on the falling edge of the ALE signal. The PM15 and PM14 bits in the PM1 register determine the output pin for the ALE signal. The ALE signal is output to internal space and external space.



NOTES:
1. D0/A0 to D7/A7 are placed in high-impedance states when read.
2. When the multiplexed bus is selected for all CS areas, the address bus becomes an I/O port.

Figure 8.10 ALE Signal and Address/Data Bus

8.2.6 RDY Signal

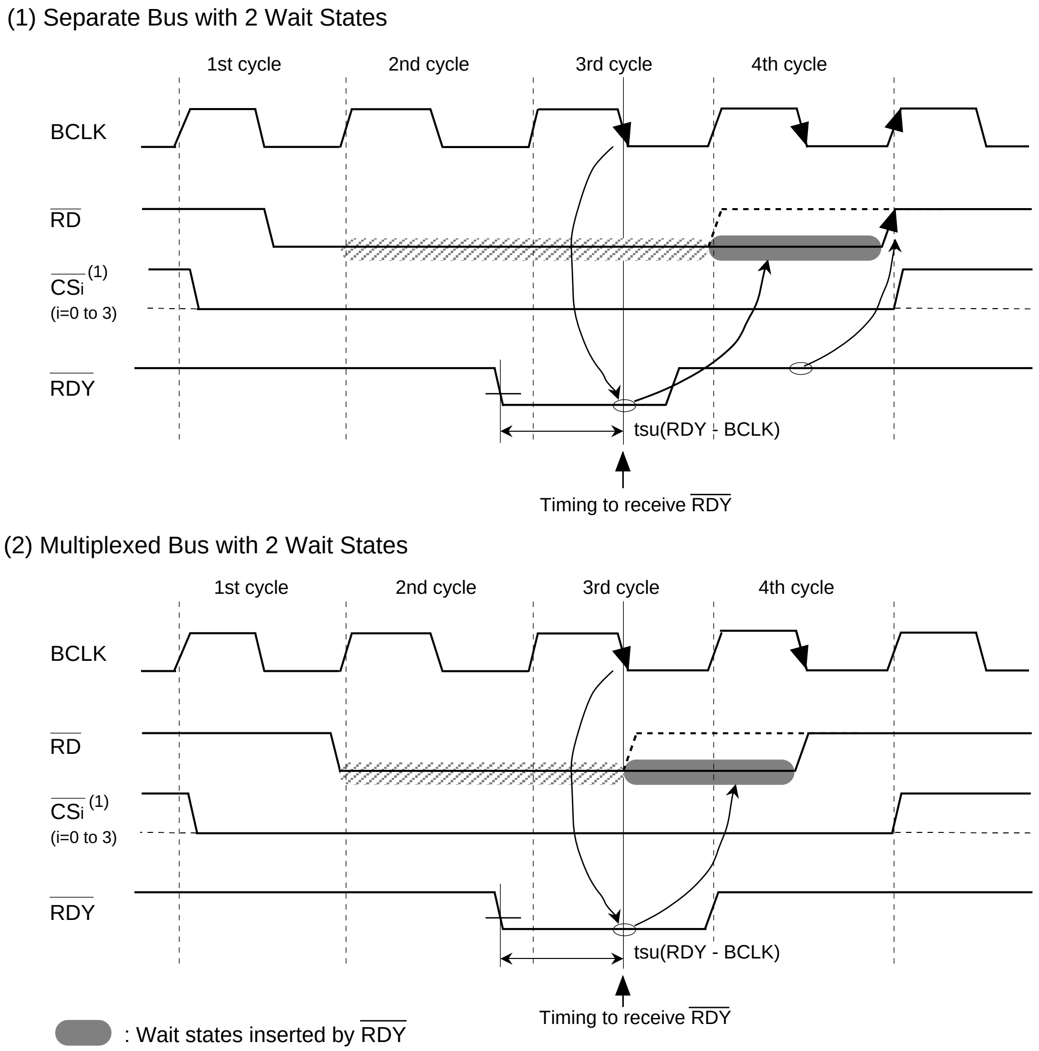
The RDY signal facilitates access to external devices requiring longer access time. When a low-level ("L") signal is applied to the RDY pin on the falling edge of the last BCLK of the bus cycle, wait states are inserted into the bus cycle. When a high-level ("H") signal is applied to the RDY pin on the falling edge of BCLK, the bus cycle starts running again.

Table 8.6 lists microcomputer states when the RDY signal inserts wait states into the bus cycle. Figure 8.11 shows an example of the RD signal that is extended by the RDY signal.

Table 8.6 Microcomputer States in Wait State[1]

Item	State
Oscillation	On
RD Signal, WR Signal, Address Bus, Data Bus, CS, ALE Signal, HLDA, Programmable I/O Ports	Maintains the same state as when RDY signal was received
Internal Peripheral Circuits	On

NOTES:
1. The RDY signal cannot be accepted immediately before software wait states are inserted.



● : Wait states inserted by $\overline{\text{RDY}}$

▨ : Wait states inserted by program

tsu(RDY-BCLK): Setup time for $\overline{\text{RDY}}$ input
Timing to receive $\overline{\text{RDY}}$ for j wait(s): j+1 cycles (j = 1 to 3)

NOTES:
1. The chip-select signal ($\overline{\text{CSi}}$) may be output longer depending on CPU state such as the instruction queue buffer.

Figure 8.11 $\overline{\text{RD}}$ Signal Output Extended by $\overline{\text{RDY}}$ Signal

8.2.7 $\overline{\text{HOLD}}$ Signal

The $\overline{\text{HOLD}}$ signal transfers bus privileges from the CPU to external circuits. When a low-level ("L") signal is applied to the $\overline{\text{HOLD}}$ pin, the microcomputer enters a hold state after bus access is completed. While the $\overline{\text{HOLD}}$ pin is held "L", the microcomputer is in a hold state and the $\overline{\text{HLDA}}$ pin outputs an "L" signal.

Table 8.7 shows the microcomputer status in a hold state.

Bus is used in the following priority order: $\overline{\text{HOLD}}$, DMAC, CPU.

$\overline{\text{HOLD}}$ > DMAC > CPU

Figure 8.12 Bus Priority Order

Table 8.7 Microcomputer Status in Hold State

Item	Status
Oscillation	On
$\overline{\text{RD}}$ Signal, $\overline{\text{WR}}$ Signal, Address Bus, Data Bus, $\overline{\text{CS}}$, $\overline{\text{BHE}}$	High-impedance
Programmable I/O Ports: P0 to P15	Maintains the same state as when $\overline{\text{HOLD}}$ was received
$\overline{\text{HLDA}}$	Outputs "L"
Internal Peripheral Circuits	On (excluding the watchdog timer)
ALE Signal	Outputs "L"

8.2.8 External Bus Status when Accessing Internal Space

Table 8.8 shows external bus states when an internal space is accessed.

Table 8.8 External Bus States when Accessing Internal Space

Item		State when Accessing SFR, Internal ROM, and Internal RAM
Address Bus		Holds address of external space last accessed
Data Bus	When Read	High-impedance
	When Write	High-impedance
$\overline{\text{RD}}$, $\overline{\text{WR}}$, $\overline{\text{WRL}}$, $\overline{\text{WRH}}$		Outputs "H"
$\overline{\text{BHE}}$		Holds state of external space last accessed
$\overline{\text{CS}}$		Outputs "H"
ALE		Outputs ALE

8.2.9 BCLK Output

The CPU clock operates the CPU. P5$_3$ outputs the CPU clock signal as BCLK when the PM07 bit in the PM0 register is set to "0" (BCLK) and the CM01 and CM00 bits in the CM0 register are set to "00$_2$" (I/O port P5$_3$).

No BCLK is output in single-chip mode. Refer to **9. Clock Generation Circuit** for details.

9. Clock Generation Circuit

9.1 Types of the Clock Generation Circuit

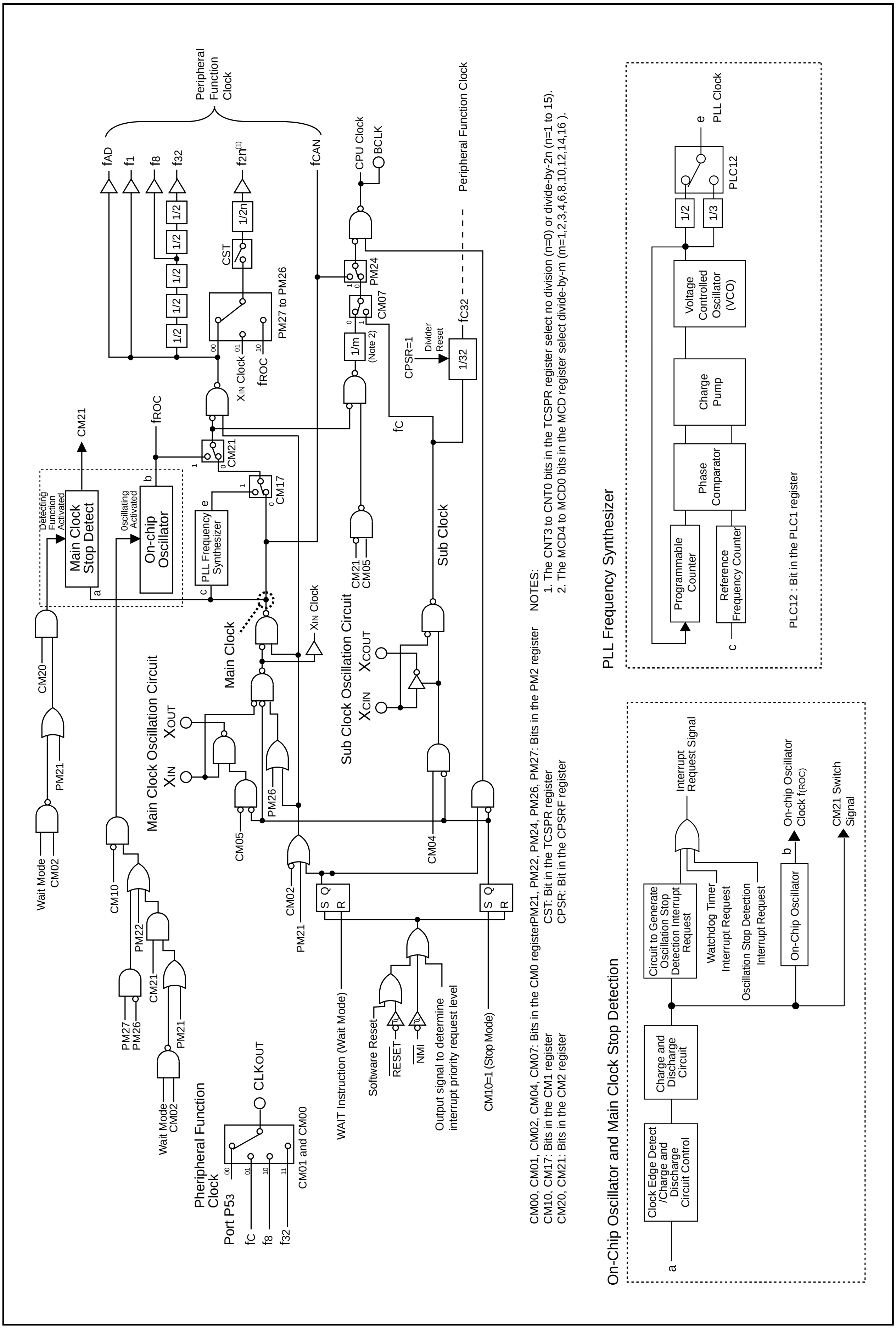
Four circuits are included to generate the system clock signal:

- Main clock oscillation circuit
- Sub clock oscillation circuit
- On-chip oscillator
- PLL frequency synthesizer

Table 9.1 lists specifications of the clock generation circuit. Figure 9.1 shows a block diagram of the clock generation circuit. Figures 9.2 to 9.8 show registers controlling the clock.

Table 9.1 Clock Generation Circuit Specifications

Item	Main Clock Oscillation Circuit	Sub Clock Oscillation Circuit	On-chip Oscillator	PLL Frequency Synthesizer
Use	CPU clock source, Peripheral function clock source	CPU clock source, Timer A and B clock source	CPU clock source, Peripheral function clock source	CPU clock source, Peripheral function clock source
Clock Frequency	Up to 32 MHz	32.768 kHz	Approx. 1 MHz	Up to 32 MHz (See Table 9.3)
Connectable Osillator or Additional Circuit	Ceramic resonator Crystal oscillator	Crystal oscillator	---	---
Pins for Oscillator or for Additional Circuit	XIN, XOUT	XCIN, XCOUT	---	---
Oscillation Stop / Restart Function	Available	Available	Available	Available
Oscillator State after Reset	Oscillating	Stopped	Stopped	Stopped
Other	Externally generated clock can be applied.	Externally generated clock can be applied.	When the main clock stops oscillating, the on-chip oscillator starts oscillating automatically and becomes clock source for the CPU and peripheral function.	---

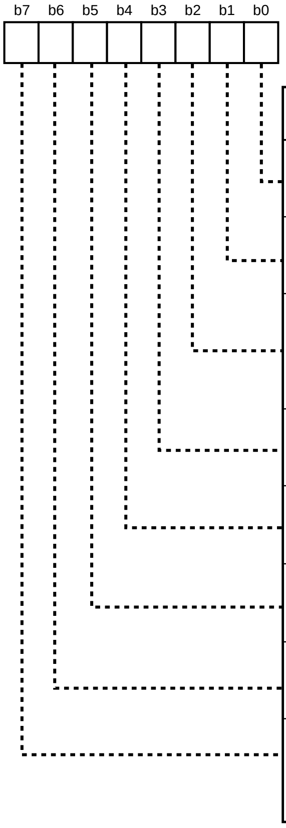

Peripheral Function Clock
fAD
f1
f8
f32
f2n(1)
fCAN
1/2
1/2
1/2
1/2
1/2
1/2n
CST
PM27 to PM26
00
01
10
XIN Clock
fROC
CPU Clock
BCLK
PM24
CM07
1/m
(Note 2)
CPSR=1
Divider Reset
1/32
fC32
Peripheral Function Clock
fC
CM21
fROC
Detecting Function Activated
Main Clock Stop Detect
Oscillating Activated
On-chip Oscillator
a
b
c
e
PLL Frequency Synthesizer
CM17
Main Clock
XIN Clock
Main Clock Oscillation Circuit
XIN
XOUT
Sub Clock Oscillation Circuit
XCIN
XCOUT
CM21
CM05
Sub Clock
CM20
PM21
Wait Mode
CM02
CM10
PM22
PM27
PM26
CM21
PM21
Wait Mode
CM02
CM05
PM26
CM04
CM02
S Q
R
PM21
S Q
R
WAIT Instruction (Wait Mode)
Software Reset
RESET
NMI
Output signal to determine interrupt priority level
CM10=1 (Stop Mode)
Pheripheral Function Clock
Port P53
fC
f8
f32
CLKOUT
00
01
10
11
CM01 and CM00
PLL Frequency Synthesizer
PLL Clock
e
PLC12
1/2
1/3
Voltage Controlled Oscillator (VCO)
Charge Pump
Phase Comparator
Programmable Counter
Reference Frequency Counter
c
PLC12 : Bit in the PLC1 register
On-Chip Oscillator and Main Clock Stop Detection
a
Clock Edge Detect /Charge and Discharge Circuit Control
Charge and Discharge Circuit
Circuit to Generate Oscillation Stop Detection Interrupt Request
Watchdog Timer Interrupt Request
Oscillation Stop Detection Interrupt Request
Interrupt Request Signal
On-Chip Oscillator
b
On-chip Oscillator Clock f(ROC)
CM21 Switch Signal

CM00, CM01, CM02, CM04, CM05, CM07: Bits in the CM0 register
CM10, CM17: Bits in the CM1 register
CM20, CM21: Bits in the CM2 register
PM21, PM22, PM24, PM26, PM27: Bits in the PM2 register
CST: Bit in the TCSPR register
CPSR: Bit in the CPSRF register

NOTES:
1. The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2n (n=1 to 15).
2. The MCD4 to MCD0 bits in the MCD register select divide-by-m (m=1,2,3,4,6,8,10,12,14,16).

Figure 9.1 Clock Generation Circuit

System Clock Control Register 0(1)



Symbol	Address	After Reset
CM0	0006_{16}	$0000\ 1000_2$

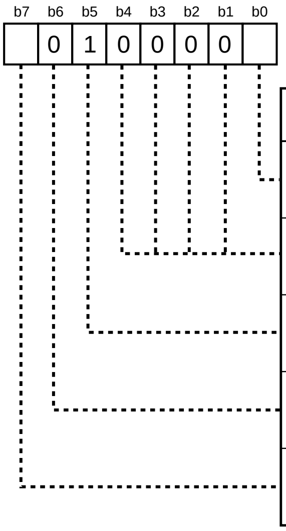
Bit Symbol	Bit Name	Function	RW
CM00	Clock Output Function Select Bit(2)	b1 b0 0 0 : I/O port $P5_3$ 0 1 : Outputs fC 1 0 : Outputs f_8 1 1 : Outputs f_{32}	RW
CM01			RW
CM02	In Wait Mode, Peripheral Function Clock Stop Bit(9)	0 : Peripheral clock does not stop in wait mode 1 : Peripheral clock stops in wait mode(3)	RW
CM03	XCIN-XCOUT Drive Capacity Select Bit(11)	0 : Low 1 : High	RW
CM04	Port XC Switch Bit	0 : I/O port function 1 : XCIN-XCOUT oscillation function(4)	RW
CM05	Main Clock (XIN-XOUT) Stop Bit(5, 9)	0 : Main clock oscillates 1 : Main clock stops(6)	RW
CM06	Watchdog Timer Function Select Bit	0 : Watchdog timer interrupt 1 : Reset(7)	RW
CM07	CPU Clock Select Bit 0(8, 9, 10)	0: Clock selected by the CM21 bit divided by MCD register setting 1: Sub clock	RW

NOTES:

1. Rewrite the CM0 register after the PRC0 bit in the PRCR register is set to "1" (write enable).
2. When the PM07 bit in the PM0 register is set to "0" (BCLK output), set the CM01 and CM00 bits to "00_2". When the PM15 and PM14 bits in the PM1 register are set to "01_2" (ALE output to $P5_3$), set the CM01 and CM00 bits to "00_2". When the PM07 bit is set to "1" (function selected in the CM01 and CM00 bits) in microprocessor or memory expansion mode, and the CM01 and CM00 bits are set to "00_2", an "L" signal is output from port $P5_3$ (port $P5_3$ does not function as an I/O port).
3. f_{C32} does not stop running. When the CM02 bit is set to "1", the PLL clock cannot be used in wait mode.
4. When setting the CM04 bit is set to "1", set the PD8_7 and PD8_6 bits in the PD8 register to "00_2" (port $P8_7$ and $P8_6$ in input mode) and the PU25 bit in the PUR2 register to "0" (no pull-up).
5. When entering low-power consumption mode or on-chip oscillator low-power consumption mode, the CM05 bit stops running the main clock. The CM05 bit cannot detect whether the main clock stops or not. To stop running the main clock, set the CM05 bit to "1" after the CM07 bit is set to "1" with a stable sub clock oscillation or after the CM21 bit in the CM2 register is set to "1" (on-chip oscillator clock). When the CM05 bit is set to "1", the clock applied to XOUT becomes "H". The built-in feedback resistor remains ON. XIN is pulled up to XOUT ("H" level) via the feedback resistor.
6. When the CM05 bit is set to "1", the MCD4 to MCD0 bits in the MCD register are set to "01000_2" (divide-by-8 mode). In on-chip oscillation mode, the MCD4 to MCD0 bits are not set to "01000_2" even if the CM05 bit terminates XIN-XOUT.
7. Once the CM06 bit is set to "1", it cannot be set to "0" by program.
8. After the CM04 bit is set to "1" with a stable sub clock oscillation, set the CM07 bit to "1" from "0". After the CM05 bit is set to "0" with a stable main clock oscillation, set the CM07 bit to "0" from "1". Do not set the CM07 bit and CM04 or CM05 bit simultaneously.
9. When the PM21 bit in the PM2 register is set to "1" (clock change disable), the CM02, CM05 and CM07 bits do not change even when written.
10. After the CM07 bit is set to "0", set the PM21 bit to "1".
11. When stop mode is entered, the CM03 bit is set to "1".

Figure 9.2 CM0 Register

System Clock Control Register 1(1)



Symbol	Address	After Reset
CM1	0007_{16}	$0010\ 0000_2$

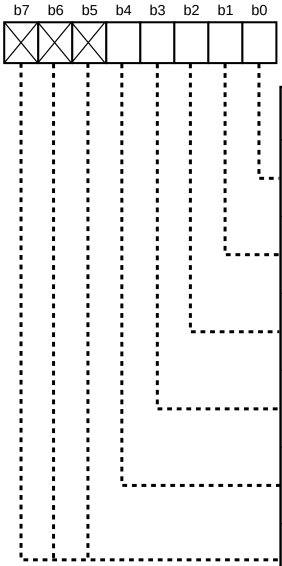
Bit Symbol	Bit Name	Function	RW
CM10	All Clock Stop Control Bit(2, 5)	0 : Clock oscillates 1 : All clocks stop (stop mode)(3)	RW
—— (b4 - b1)	Reserved Bit	Set to "0"	RW
—— (b5)	Reserved Bit	Set to "1"	RW
—— (b6)	Reserved Bit	Set to "0"	RW
CM17	CPU Clock Select Bit 1(4,5)	0 : Main clock 1 : PLL clock	RW

NOTES:

1. Rewrite the CM1 register after the PRC0 bit in the PRCR register is set to "1" (write enable).
2. When the CM10 bit is set to "1", the clock applied to XOUT becomes "H" and the built-in feedback resistor is disabled. XIN, XCIN and XCOUT are placed in high-impedance states.
3. When the CM10 bit is set to "1", the MCD4 to MCD0 bits in the MCD register are set to "01000_2" (divide-by-8 mode). When the CM20 bit is set to "1" (oscillation stop detect function enabled) or the CM21 bit to "1" (on-chip oscillator selected), do not set the CM10 bit to "1".
4. The CM17 bit setting is enabled only when the CM21 bit in the CM2 register is set to "0". Use the procedure shown in Figure 9.12 to set the CM17 bit to "1".
5. If the PM21 bit in the PM2 register is set to "1" (clock change disable), the CM10 and CM17 bits do not change when written.
   If the PM22 bit in the PM2 register is set to "1" (on-chip oscillator clock as watchdog timer count source), the CM10 bit setting does not change when written.

Figure 9.3 CM1 Register

Main Clock Division Register[1]



Symbol	Address	After Reset
MCD	$000C_{16}$	XXX0 1000_2

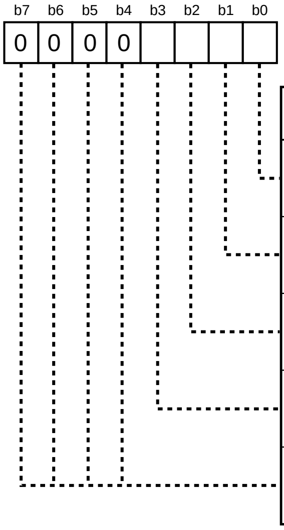
Bit Symbol	Bit Name	Function	RW
MCD0	Main Clock Division Select Bit[2, 4]	b4 b3 b2 b1 b0 1 0 0 1 0 : Divide-by-1(no division) mode 0 0 0 1 0 : Divide-by-2 mode 0 0 0 1 1 : Divide-by-3 mode 0 0 1 0 0 : Divide-by-4 mode 0 0 1 1 0 : Divide-by-6 mode 0 1 0 0 0 : Divide-by-8 mode 0 1 0 1 0 : Divide-by-10 mode 0 1 1 0 0 : Divide-by-12 mode 0 1 1 1 0 : Divide-by-14 mode 0 0 0 0 0 : Divide-by-16 mode (Note 3)	RW
MCD1			RW
MCD2			RW
MCD3			RW
MCD4			RW
—— (b7 - b5)	Reserved Bit	When read, its content is indeterminate	RO

NOTES:

1. Rewrite the MCD register after the PRC0 bit in the PRCR register is set to "1" (write enable).
2. When the microcomputer enters stop mode or low-power consumption mode, the MCD4 to MCD0 bits are set to "01000_2".
   The MCD4 to MCD0 bits are not set to "01000_2" even if the CM05 bit in the CM0 register is set to "1" (XIN-XOUT stopped) in on-chip oscillator mode.
3. Bit combinations cannot be set not listed above.
4. Access CAN-associated register addresses after setting the MCD4 to MCD0 bits are set to "10010_2", when the PM24 bit in the PM2 register is set to "0" (clock selected by the CM07 bit).

Figure 9.4 MCD Register

Oscillation Stop Detection Register[1]



Symbol	Address	After Reset
CM2	000D$_{16}$	00$_{16}$

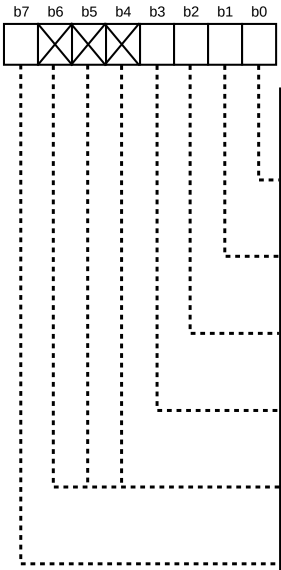
Bit Symbol	Bit Name	Function	RW
CM20	Oscillation Stop Detection Enable Bit[2]	0: Disables oscillation stop detect function 1: Enables oscillation stop detect function	RW
CM21	CPU Clock Select Bit 2[3, 4]	0: Clock selected by the CM17 bit 1: On-chip oscillator clock	RW
CM22	Oscillation Stop Detection Flag[5]	0: Main clock does not stop 1: Detects a main clock stop	RW
CM23	Main Clock Monitor Flag[6]	0: Main clock oscillates 1: Main clock stops	RO
—— (b7 - b4)	Reserved Bit	Set to "0"	RW

NOTES:

1. Rewrite the CM2 register after the PRC0 bit in the PRCR register is set to "1" (write enable).
2. If the PM21 bit in the PM2 register is set to "1" (clock change disable), the CM20 bit setting does not change when written.
3. When a main clock oscillation stop is detected while the CM20 bit is set to "1", the CM21 bit is set to "1". Although the main clock starts oscillating, the CM21 bit is not set to "0". If the main clock is used as a CPU clock source after the main clock resumes oscillating, set the CM21 bit to "0" by program.
4. When the CM20 bit is set to "1" and the CM22 bit is set to "1", do not set the CM21 bit to "0".
5. When a main clock stop is detected, the CM22 bit is set to "1". The CM22 bit can only be set to "0", not "1", by program.
   If the CM22 bit is set to "0" by program while the main clock stops, the CM22 bit cannot be set to "1" until the next main clock stop is detected.
6. Determine the main clock state by reading the CM23 bit several times after the oscillation stop detection interrupt is generated.

Figure 9.5 CM2 Register

Count Source Prescaler Register



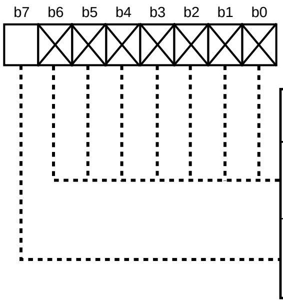
Symbol	Address	After Reset[2]
TCSPR	$035F_{16}$	$0XXX\ 0000_2$

Bit Symbol	Bit Name	Function	RW
CNT0	Division Rate Select Bit[1]	If setting value is *n*, *f2n* is the main clock, on-chip oscillator clock or PLL clock divided by *2n*. When *n* is set to "0", no division is selected.	RW
CNT1			RW
CNT2			RW
CNT3			RW
——— (b6 - b4)	Reserved Bit	When read, its content is indeterminate	RO
CST	Operation Enable Bit	0: Divider stops 1: Divider starts	RW

NOTES:

1. Rewrite the CNT3 to CNT0 bits after the CST bit is set to "0".
2. Value of the TCSPR register is not reset by software reset or watchdog timer reset.

Clock Prescaler Reset Flag



Symbol	Address	After Reset
CPSRF	0341_{16}	$0XXX\ XXXX_2$

Bit Symbol	Bit Name	Function	RW
——— (b6 - b0)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		—
CPSR	Clock Prescaler Reset Flag	When the CPSR bit is set to "1", *fc* divided by 32 is reset. When read, its content is "0".	RW

Figure 9.6 TCSPR and CPSRF Registers

PLL Control Register 0[1, 2, 5]

b7	b6	b5	b4	b3	b2	b1	b0
	1	0	1	☒			

Symbol	Address	After Reset
PLC0	0026_{16}	$0001\ X010_2$

Bit Symbol	Bit Name	Function	RW
PLC00	Programmable Counter Select Bit[3]	b2 b1 b0 0 1 1 : Multiply-by-6 1 0 0 : Multiply-by-8 Do not set to values other than the above	RW
PLC01			RW
PLC02			RW
(b3)	Reserved Bit	When read, its content is indeterminate	RO
(b4)	Reserved Bit	Set to "1"	RW
(b5)	Reserved Bit	Set to "0"	RW
(b6)	Reserved Bit	Set to "1"	RW
PLC07	Operation Enable Bit[4]	0: PLL is Off 1: PLL is On	RW

NOTES:
1. Rewrite the PLC0 register after the PRC0 bit in the PRCR register is set to "1" (write enable).
2. If the PM21 bit in the PM2 register is set to "1" (clock change disable), the PLC0 register setting does not change when written.
3. Set the PLC02 to PLC00 bits when the PLC07 bit is set to "0". Once these bits are set, they cannot be changed.
4. Set the CM17 bit in the CM1 register to "0" (main clock as CPU clock source) and the PLC07 bit to "0" before entering wait or stop mode.
5. Set the PLC0 and PLC1 registers simultaneously in 16-bit units.

PLL Control Register 1[1, 2, 3, 4]

b7	b6	b5	b4	b3	b2	b1	b0
0	0	0	☒	0		1	0

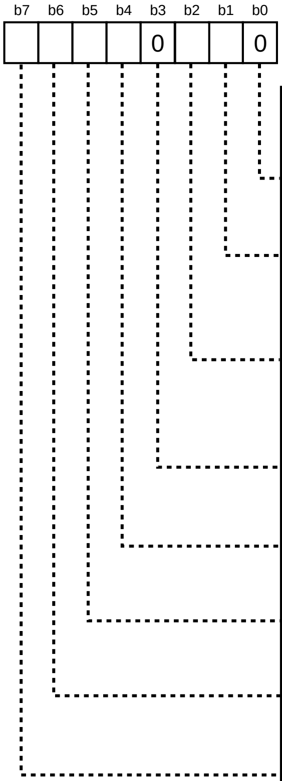
Symbol	Address	After Reset
PLC1	0027_{16}	$000X\ 0000_2$

Bit Symbol	Bit Name	Function	RW
(b0)	Reserved Bit	Set to "0"	RW
(b1)	Reserved Bit	Set to "1"	RW
PLC12	PLL Clock Division Switch Bit	0 : Divide-by-2 1 : Divide-by-3	RW
(b3)	Reserved Bit	Set to "0"	RW
(b4)	Reserved Bit	When read, its content is indeterminate	RO
(b7 - b5)	Reserved Bit	Set to "0"	RW

NOTES:
1. Rewrite the PLC1 register after the PRC0 bit in the PRCR register is set to "1" (write enable).
2. If the PM21 bit in the PM2 register is set to "1" (clock change disable), the PLC1 register does not change when written.
3. Set the PLC1 register when the PLC07 bit is set to "0" (PLL off).
4. Set the PLC0 and PLC1 registers simultaneously in 16-bit units.

Figure 9.7 PLC0 and PLC1 Registers

Processor Mode Register 2[1]



Symbol	Address	After Reset
PM2	0013_{16}	00_{16}

Bit Symbol	Bit Name	Function	RW
(b0)	Reserved Bit	Set to "0"	RW
PM21	System Clock Protect Bit[2, 3]	0 : Protects the clock by a PRCR register setting 1 : Disables a clock change	RW
PM22	WDT Count Source Protect Bit[2, 4]	0 : Selects BCLK as count source of the watchdog timer 1 : Selects the on-chip oscillator clock as count source of the watchdog timer	RW
(b3)	Reserved Bit	Set to "0"	RW
PM24	CPU Clock Select Bit 3	0 : Clock selected by the CM07 bit 1 : Main Clock	RW
PM25	CAN Clock Select Bit	0 : f_1 1 : Main Clock	RW
PM26	f_{2n} Count source Select Bit	b7 b6 0 0 : Peripheral function clock 0 1 : X_{IN} clock 1 0 : On-chip oscillator clock 1 1 : Do not set to this value	RW
PM27			RW

NOTES:

1. Rewrite the PM2 register after the PRC1 bit in the PRCR register is set to "1" (write enable).
2. Once the PM22 and PM21 bits are set to "1", they can not be set to "0" by program.
3. When the PM21 bit is set to "1",
   the CPU clock keeps running when the WAIT instruction is executed;
   nothing is changed even if following bits are set to either "0" or "1".
   - the CM02 bit in the CM0 register (the peripheral function clock is not stopped in wait mode.)
   - the CM05 bit in the CM0 register (the main clock is not stopped.)
   - the CM07 bit in the CM0 register (a CPU clock source is not changed.)
   - the CM10 bit in the CM1 register (the microcomputer does not enter stop mode.)
   - the CM17 bit in the CM1 register (a CPU clock source is not changed.)
   - the CM20 bit in the CM2 register (oscillation stop detect function settings are not changed.)
   - all bits in the PLC0 and PLC1 registers (PLL frequency synthesizer function settings are not changed.)
4. When the PM22 bit is set to "1",
   the on-chip oscillator clock becomes a count source of the watchdog timer after the on-chip oscillator starts;
   write to the CM10 bit is disabled (the microcomputer does not enter stop mode.);
   the watchdog timer keeps running when the microcomputer is in wait mode and hold state.

Figure 9.8 PM2 Register

9.1.1 Main Clock

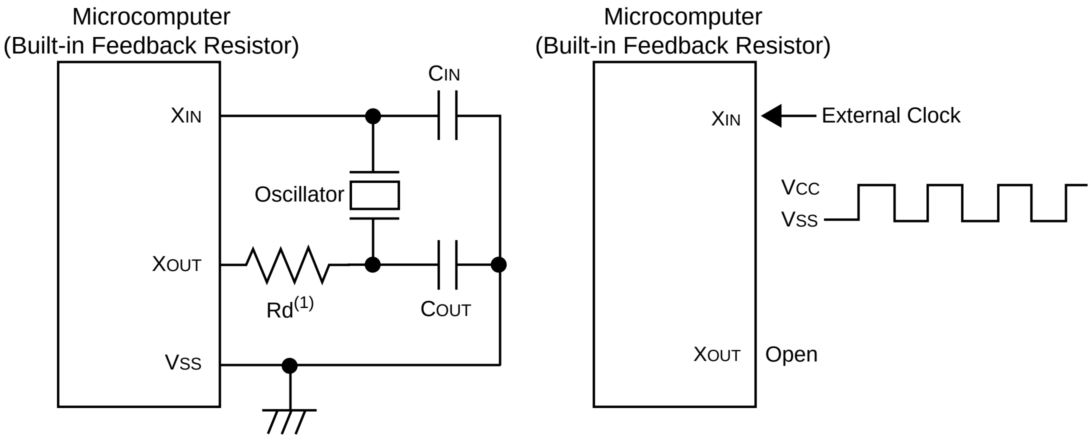
Main clock oscillation circuit generates the main clock. The main clock becomes clock source of the CPU clock and peripheral function clock.

The main clock oscillation circuit is configured by connecting an oscillator or resonator between the XIN and XOUT pins. The circuit has a built-in feedback resistor. The feedback resistor is separated from the oscillation circuit in stop mode to reduce power consumption. An external clock can be applied to the XIN pin in the main clock oscillation circuit. Figure 9.9 shows an example of a main clock circuit connection. Circuit constants vary depending on each oscillator. Use the circuit constant recommended by each oscillator manufacturer.

The main clock divided-by-eight becomes a CPU clock source after reset.

To reduce power consumption, set the CM05 bit in the CM0 register to "1" (main clock stopped) after switching the CPU clock source to the sub clock or on-chip oscillator clock. In this case, the clock applied to XOUT becomes high ("H"). XIN is pulled up by XOUT via the feedback resistor which remains on. When an external clock is applied to the XIN pin, do not set the CM05 bit to "1".

All clocks, including the main clock, stop in stop mode. Refer to **9.5 Power Consumption Control** for details.



NOTES:

1. Place a damping resistor if required. Resistance values vary depending on the oscillator setting. Use values recommended by each oscillator manufacturer.
   Place a feedback resistor between XIN and XOUT if the oscillator manufacturer recommends placing the resistor externally.

Figure 9.9 Main Clock Circuit Connection

9.1.2 Sub Clock

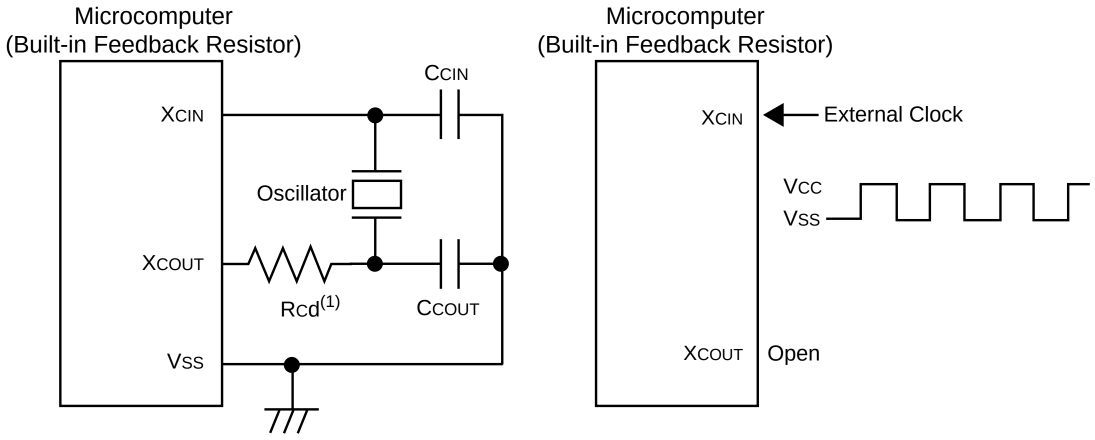
Sub clock oscillation circuit generates the sub clock. The sub clock becomes clock source of the CPU clock and for the timers A and B. The same frequency, fc, as the sub clock can be output from the CLKOUT pin.

The sub clock oscillation circuit is configured by connecting a crystal oscillator between the XCIN and XCOUT pins. The circuit has a built-in feedback resistor. The feedback resistor is separated from the oscillation circuit in stop mode to reduce power consumption. An external clock can be applied to the XCIN pin. Figure 9.10 shows an example of a sub clock circuit connection. Circuit constants vary depending on each oscillator. Use the circuit constant recommended by each oscillator manufacturer.

The sub clock stops after reset. The feedback resistor is separated from the oscillation circuit. When the PD8_6 and PD8_7 bits in the PD8 register are set to "0" (input mode) and the PU25 bit in the PUR2 register is set to "0" (no pull-up), set the CM04 bit in the CM0 register to "1" (XCIN-XCOUT oscillation function). The sub clock oscillation circuit starts oscillating. To apply an external clock to the XCIN pin, set the CM04 bit to "1" when the PD8_7 bit is set to "0" and the PU25 bit to "0". The clock applied to the XCIN pin becomes a clock source of the sub clock.

When the CM07 bit in the CM0 register is set to "1" (sub clock) after the sub clock oscillation has stabilized, the sub clock becomes a CPU clock source.

All clocks, including the sub clock, stop in stop mode. Refer to **9.5 Power Consumption Control** for details.



NOTES:
1. Place a damping resistor if required. Resistance values vary depending on the oscillator setting. Use values recommended by each oscillator manufacturer.
Place a feedback resistor between XCIN and XCOUT if the oscillator manufacturer recommends placing the resistor externally.

Figure 9.10 Sub Clock Circuit Connection

9.1.3 On-Chip Oscillator Clock

On-chip oscillator generates the on-chip oscillator clock. The 1-MHz on-chip oscillator clock becomes a clock source of the CPU clock and peripheral function clock.

The on-chip oscillator clock stops after reset. When the CM21 bit in the CM2 register is set to "1" (on-chip oscillator clock), the on-chip oscillator starts oscillating. Instead of the main clock, the on-chip oscillator clock becomes clock source of the CPU clock and peripheral function clock.

Table 9.2 shows bit settings for on-chip oscillator start condition.

Table 9.2 Bit Settings for On-Chip Oscillator Start Condition

CM2 Register	PM2 Register		Used as
CM21 Bit	PM22 Bit	PM27 and PM26 Bits	
1	0	0 0	CPU clock source or peripheral function clock source
0	1	0 0	Watchdog timer operating clock source (The clock keeps running when entering stop mode.)
0	0	0 1	f_{2n} count source

9.1.3.1 Oscillation Stop Detect Function

When the main clock is terminated by external source, the on-chip oscillator automatically starts oscillating to generate another clock.

When the CM 20 bit in the CM2 registser is set to "1" (oscillation stop detect function enabled), an oscillation stop detection interrupt request is generated as soon as the main clock stops. Simultaneously, the on-chip oscillator starts oscillating. Instead of the main clock, the on-chip oscillator clock becomes clock source for the CPU clock and peripheral function clock. Associated bits are set as follows:

- The CM21 bit is set to "1" (on-chip oscillator clock becomes a clock source of the CPU clock.)
- The CM22 bit is set to "1" (main clock stop is detected.)
- The CM23 bit is set to "1" (main clock stops.) (See **Figure 9.14**)

9.1.3.2 How to Use Oscillation Stop Detect Function

- The oscillation stop detection interrupt shares vectors with the watchdog timer interrupt and the low voltage detection interrupt. When these interrupts are used simultaneously, read the CM22 bit with an interrupt routine to determine if an oscillation stop detection interrupt request has been generated.
- When the main clock resumes running after an oscillation stop is detected, set the main clock as clock source of the CPU clock and peripheral function clock. Figure 9.11 shows the procedure to switch the on-chip oscillator clock to the main clock.
- In low-speed mode, when the main clock is stopped by setting the CM20 bit to "1", the oscillation stop detection interrupt request is generated. Simultaneously, the on-chip oscillator starts oscillating. The sub clock remains the CPU clock source. The on-chip oscillator clock becomes a clock source for the peripheral function clock.
- When the peripheral function clock stops running, the oscillation stop detect function is also disabled. To enter wait mode while the oscillation stop detect function is in use, set the CM02 bit in the CM0 register to "0" (peripheral clock does not stop in wait mode).
- The oscillation stop detect function is provided to handle main clock stop caused by external source. Set the CM20 bit to "0" (oscillation stop detect function disabled) when the main clock is terminated by program, i.e., entering stop mode or setting the CM05 bit to "1" (main clock oscillation stop).
- When the main clock frequency is 2MHz or less, the oscillation stop detect function is not available. Set the CM20 bit to "0".

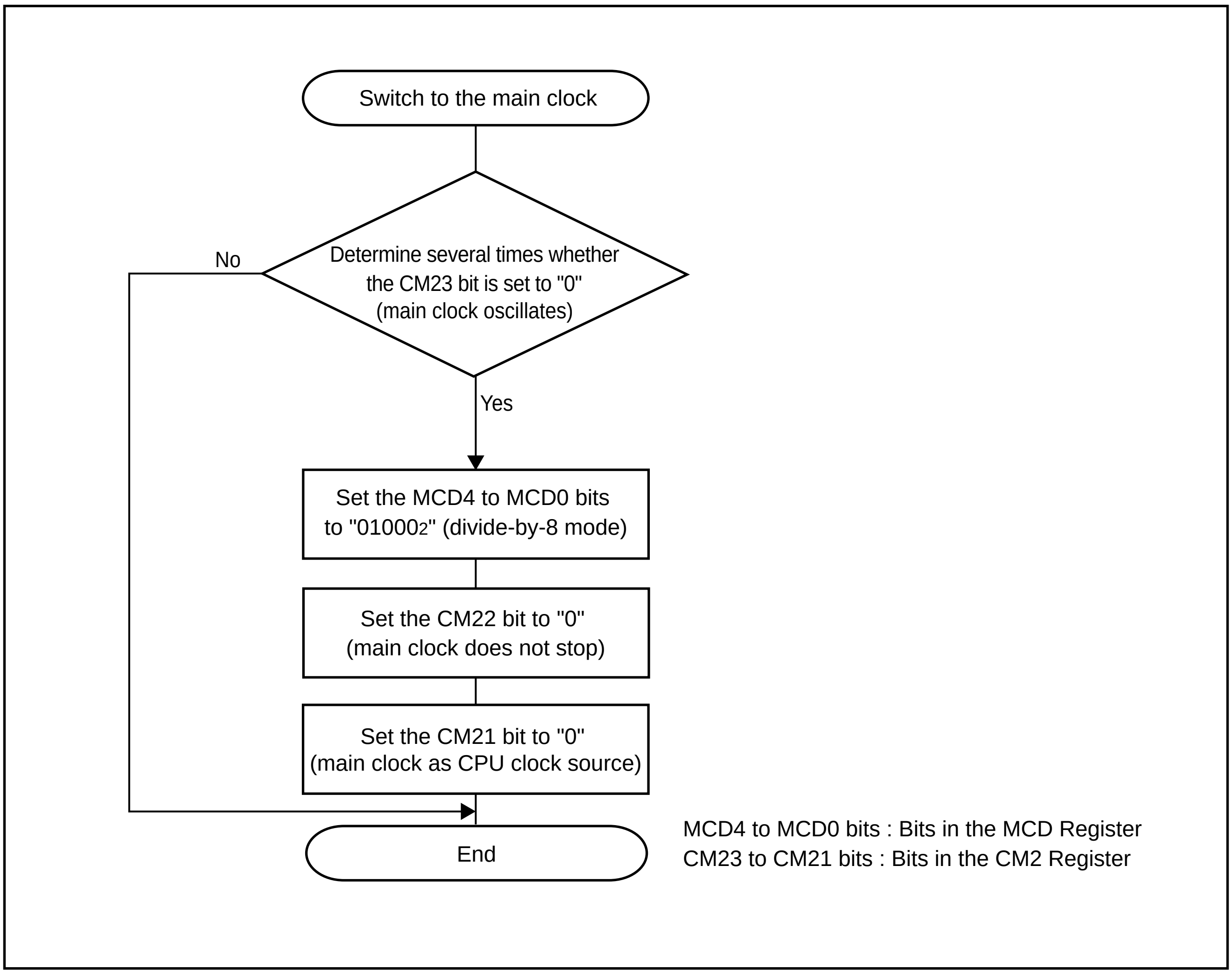

Switch to the main clock
Determine several times whether the CM23 bit is set to "0" (main clock oscillates)
No
Yes
Set the MCD4 to MCD0 bits to "01000₂" (divide-by-8 mode)
Set the CM22 bit to "0" (main clock does not stop)
Set the CM21 bit to "0" (main clock as CPU clock source)
End
MCD4 to MCD0 bits : Bits in the MCD Register
CM23 to CM21 bits : Bits in the CM2 Register

Figure 9.11 Switching Procedure from On-chip Oscillator Clock to Main Clock

9.1.4 PLL Clock

The PLL frequency synthesizer generates the PLL clock based on the main clock. The PLL clock can be used as clock source for the CPU clock and peripheral function clock.

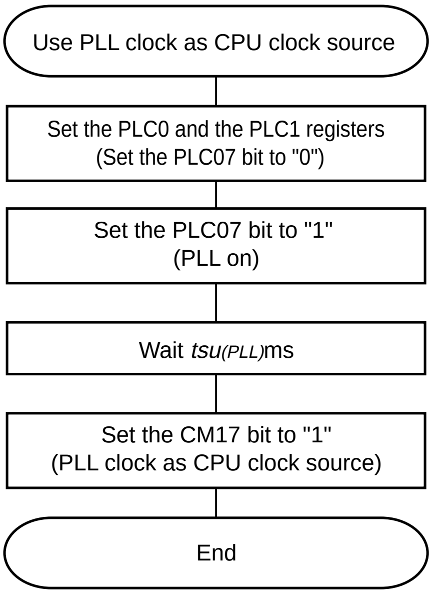
The PLL frequency synthesizer stops after reset. When the PLC07 bit is set to "1" (PLL on), the PLL frequency synthesizer starts operating. Wait $tsu_{(PLL)}$ ms for the PLL clock to stabilize.

The PLL clock can either be the clock output from the voltage controlled oscillator (VCO) divided-by-2 or divided-by-3. When the PLL clock is used as a clock source for the CPU clock or peripheral function clock, set each bit as is shown in Table 9.3. Figure 9.12 shows the procedure to use the PLL clock as the CPU clock source.

To enter wait or stop mode, set the CM17 bit to "0" (main clock as CPU clock source), set the PLC07 bit in the PLC0 register to "0" (PLL off) and then enter wait or stop mode.

Table 9.3 Bit Settings to Use PLL Clock as CPU Clock Source

f(XIN)	PLC0 Register			PLC1 Register	PLL Clock
	PLC02 Bit	PLC01 Bit	PLC00 Bit	CM21 Bit	
10 MHz	0	1	1	0	30 MHz
				1	20 MHz
8 MHz	1	0	0	0	32 MHz
				1	21.3 MHz


Use PLL clock as CPU clock source
Set the PLC0 and the PLC1 registers (Set the PLC07 bit to "0")
Set the PLC07 bit to "1" (PLL on)
Wait tsu(PLL)ms
Set the CM17 bit to "1" (PLL clock as CPU clock source)
End

PLC07 bit : Bit in the PLC0 Register
CM17 bit : Bit in the CM1 Register

Figure 9.12 Procedure to Use PLL Clock as CPU Clock Source

9.2 CPU Clock and BCLK

The CPU operating clock is referred to as the CPU clock. The CPU clock is also a count source for the watchdog timer. After reset, the CPU clock is the main clock divided-by-8 . In memory expansion or microprocessor mode, the clock having the same frequency as the CPU clock can be output from the BCLK pin as BCLK. Refer to **9.4 Clock Output Function** for details.

The main clock, sub clock, on-chip oscillator clock or PLL clock can be selected as a clock source for the CPU clock. Table 9.4 shows CPU clock source and bit settings.

When the main clock, on-chip oscillator clock or PLL clock is selected as a clock source of the CPU clock, the selected clock divided-by-1 (no division), -2, -3, -4, -6, -8, -10, -12, -14 or -16 becomes the CPU clock. The MCD4 to MCD0 bits in the MCD register select the clock division.

When the microcomputer enters stop mode or low-power consumption mode (except when the on-chip oscillator clock is the CPU clock), the MCD4 to MCD0 bits are set to "01000_2" (divide-by-8 mode). Therefore, when the main clock starts running, the CPU clock enters medium-speed mode (divide-by-8).

Table 9.4 CPU Clock Source and Bit Settings

CPU Clock Source	CM0 Register	CM1 Register	CM2 Register	PM2 Register
	CM07 Bit	CM17 Bit	CM21 Bit	PM24 Bit
Main Clock	0	0	0	0
Main Clock (Main Clock Direct Mode)(1)	0	0	0	1
Sub Clock	1	0	0	0
On-Chip Oscillator Clock	0	0	1	0
PLL Clock	0	1	0	0

NOTES:

1. Refer to **23.2 CAN Clock** for details.

9.3 Peripheral Function Clock

The peripheral function clock becomes an operating clock or count source for peripheral functions excluding the watchdog timer.

9.3.1 f_1, f_8, f_{32} and f_{2n}

f_1, f_8 and f_{32} are the peripheral function clock, selected by the CM21 bit, divided-by-1, -8, or -32. The PM27 and PM26 bits in the PM2 register selects a f_{2n} count source from the peripheral clock, X_{IN} clock, and the on-chip oscillator clock. The CNT3 to CNT0 bits in the TCSPR register selects a f_{2n} division. (n=0 to 15. No division when n=0.)

f_1, f_8, f_{32} and f_{2n} stop when the CM02 bit in the CM0 register to "1" (peripheral function stops in wait mode) to enter wait mode or when in low-power consumption mode.

f_1, f_8 and f_{2n} are used as an operating clock of the serial I/O and count source of the timers A and B. f_1 is also used as an operating clock for the intelligent I/O.

The CLK_{OUT} pin outputs f_8 and f_{32} . Refer to **9.4 Clock Output Function** for details.

9.3.2 f_{AD}

f_{AD} is an operating clock for the A/D converter and has the same frequency as either the main clock(1) or the on-chip oscillator clock. The CM21 bit determines which clock is selected.

If the CM02 bit is set to "1" (peripheral function stop in wait mode) to enter wait mode, f_{AD} stops. f_{AD} also stops in low-power consumption mode.

NOTES:

1. The PLL clock, instead of the main clock, when the CM17 bit is set to "1" (PLL clock).

9.3.3 f_{C32}

f_{C32} is the sub clock divided by 32. f_{C32} is used as a count source for the timers A and B. f_{C32} is available when the sub clock is running.

9.3.4 f_{CAN}

f_{CAN} has the same frequency as the main clock. It is a clock for the CAN module only.

9.4 Clock Output Function

The CLKOUT pin outputs f_C, f_8 or f_{32}.

In memory expansion mode or microprocessor mode, a clock having the same frequency as the CPU clock can be output from the BCLK pin as BCLK.

Table 9.5 lists CLKOUT pin function in single-chip mode. Table 9.6 lists CLKOUT pin function in memory expansion mode and microprocessor mode.

Table 9.5 CLKOUT Pin in Single-Chip Mode

PM0 Register [1]	CM0 Register [2]		CLKOUT Pin Function
PM07 Bit	CM01 Bit	CM00 Bit	
—	0	0	P5$_3$ I/O port
1	0	1	Outputs fc
1	1	0	Outputs f_8
1	1	1	Outputs f_{32}

- : Can be set to either "0" or "1"

NOTES:

1. Rewrite the PM0 register after the PRC1 bit in the PRCR register is set to "1" (write enable).
2. Rewrite the CM0 register after the PRC0 bit in the PRCR register is set to "1" (write enable).

Table 9.6 CLKOUT Pin in Memory Expansion Mode and Microprocessor Mode

PM1 Register[1]		PM0 Register[1]	CM0 Register[2]		CLKOUT Pin Function
PM15 Bit	PM14 Bit	PM07 Bit	CM01 Bit	CM00 Bit	
00_2, 10_2, 11_2,		0	0 [3]	0 [3]	Outputs BCLK
		1	0	0	Outputs "L" (not P5$_3$)
		1	0	1	Outputs fc
		1	1	0	Outputs f_8
		1	1	1	Outputs f_{32}
0	1	—	0 [3]	0 [3]	Outputs ALE

- : Can be set to either "0" or "1"

NOTES:

1. Rewrite the PM1 and PM0 registers after the PRC1 bit in the PRCR register is set to "1" (write enable).
2. Rewrite the CM0 register after the PRC0 bit in the PRCR register is set to "1" (write enable).
3. When the PM07 bit is set to "0" (selected in the CM01 and CM00 bits) or the PM15 and PM14 bits are set to "01_2" (P5$_3$/BCLK), set the CM01 and CM00 bits to "00_2" (I/O port P5$_3$).
4. M32C/86T cannot be used in memory expansion mode and microprocessor mode.

9.5 Power Consumption Control

Normal operating mode, wait mode and stop mode are provided as the power consumption control.
All mode states, except wait mode and stop mode, are called normal operating mode in this section. Figure 9.13 shows a block diagram of status transition in wait mode and stop mode. Figure 9.14 shows a block diagram of status transition in all modes.

9.5.1 Normal Operating Mode

The normal operating mode is further separated into six modes.
In normal operating mode, the CPU clock and peripheral function clock are supplied to operate the CPU and peripheral function. The power consumption control is enabled by controlling a CPU clock frequency. The higher the CPU clock frequency is, the more processing power increases. The lower the CPU clock frequency is, the more power consumption decreases. When unnecessary oscillation circuit stops, power consumption is further reduced.

9.5.1.1 High-Speed Mode

The main clock[1] becomes the CPU clock and a clock source of the peripheral function clock. When the sub clock runs, f_{C32} can be used as a count source for the timers A and B.

9.5.1.2 Medium-Speed Mode

The main clock[1] divided-by-2, -3, -4, -6, -8, -10, -12, -14, or -16 becomes the CPU clock. The main clock[1] is a clock source for the peripheral function clock. When the sub clock runs, f_{C32} can be used as a count source for the timers A and B.

9.5.1.3 Low-Speed Mode

The sub clock becomes the CPU clock . The main clock[1] is a clock source for the peripheral function clock. f_{C32} can be used as a count source for the timers A and B.

9.5.1.4 Low-Power Consumption Mode

The microcomputer enters low-power consumption mode when the main clock stops in low-speed mode. The sub clock becomes the CPU clock. Only f_{C32} can be used as a count source for the timers A and B and the peripheral function clock. In low-power consumption mode, the MCD4 to MCD0 bits in the MCD register are set to "01000_2" (divide-by-8 mode). Therefore, when the main clock resumes running, the microcomputer is in midium-speed mode (divide-by-8 mode).

9.5.1.5 On-Chip Oscillator Mode

The on-chip oscillator clock divided-by-1 (no division), -2, -3, 4-, -6, -8, -10, -12, -14, or -16 becomes the CPU clock. The on-chip oscillator clock is a clock source for the peripheral function clock. When the sub clock runs, f_{C32} can be used as a count source for the timers A and B.

9.5.1.6 On-Chip Oscillator Low-Power Consumption Mode

The microcomputer enters on-chip oscillator low-power consumption mode when the main clock stops in on-chip oscillator mode . The on-chip oscillator clock divided-by-1 (no division), -2, -3, -4, -6, -8, -10, -12, -14, or -16 becomes the CPU clock. The on-chip oscillator clock is a clock source for the peripheral function clock. When the sub clock runs, f_{C32} can be used as a count source for the timers A and B.

NOTES:

[1] The PLL clock, instead of the main clock, when the CM17 bit is set to "1" (PLL clock).

Switch the CPU clock after the clock to be switched to stabilize. Sub clock oscillation will take longer[2] to stabilize. Wait, by program, until the clock stabilizes directly after turning the microcomputer on or exiting stop mode.
To switch the on-chip oscillator clock to the main clock, enter medium-speed mode (divide-by-8) after the main clock is divided by eight in on-chip oscillator mode (the MCD4 to MCD0 bits in the MCD register are set to "01000_2").
Do not enter on-chip oscillator mode or on-chip oscillator low-power consumption mode from low-speed mode or low-power consumption mode and vice versa.

NOTES:
[2] Contact your oscillator manufacturer for oscillation stabilization time.

9.5.2 Wait Mode

In wait mode, the CPU clock stops running. The CPU and watchdog timer, operated by the CPU clock, also stop. When the PM22 bit in the PM2 register is set to "1" (on-chip oscillator clock as watchdog timer count source), the watchdog timer continues operating. Because the main clock, sub clock and on-chip oscillator clock continue running, peripheral functions using these clocks also continue operating.

9.5.2.1 Peripheral Function Clock Stop Function

If the CM02 bit in the CM0 register is set to "1" (peripheral function clock stops in wait mode), f_1, f_8, f_{32}, f_{2n} (when peripheral clock is selected as a count source), and f_{AD} stop in wait mode. Power consumption can be reduced. f_{2n}, when X_{IN} clock or on-chip oscillator clock is selected as a count source, and f_{C32} do not stop running.

9.5.2.2 Entering Wait Mode

If wait mode is entered after setting the CM02 bit to "1", set the MCD4 to MCD0 bits in the MCD register to be the 10-MHz or less CPU clock flequency after dividing the main clock.
Enter wait mode after setting the followings.

- Initial Setting
  Set each interrupt priority level after setting the exit priority level required to exit wait mode, controlled by the RLVL2 to RLVL0 bits in the RLVL register, to "7".

- Before Entering Wait Mode
  (1) Set the I flag to "0"
  (2) Set the interrupt priority level of the interrupt being used to exit wait mode
  (3) Set the interrupt priority levels of the interrupts, not being used to exit wait mode, to "0"
  (4) Set IPL in the FLG register. Then set the exit priority level to the same level as IPL
  Interrupt priority level of the interrupt used to exit wait mode > IPL = the exit priority level
  (5) Set the PRC0 bit in the PRCR register to "1"
  (6) If the CPU clock source is the PLL clock, set the CM17 bit in the CM1 register to "0" (main clock) and PLC07 bit in the PLC0 register to "0" (PLL off)
  (7) Set the I flag to "1"
  (8) Execute the WAIT instruction

- After Exiting Wait Mode
  Set the exit priority level to "7" as soon as exiting wait mode.

9.5.2.3 Pin Status in Wait Mode

Table 9.7 lists pin states in wait mode.

Table 9.7 Pin States in Wait Mode

Pin		Memory Expansion Mode[1] Microprocessor Mode[1]	Single-Chip Mode
Address Bus, Data Bus, $\overline{CS0}$ to $\overline{CS3}$, $\overline{BHE}$		Maintains state immediately before entering wait mode	
$\overline{RD}$, $\overline{WR}$, $\overline{WRL}$, $\overline{WRH}$		"H"	
$\overline{HLDA}$, BCLK		"H"	
ALE		"L"	
Ports		Maintains state immediately before entering wait mode	
CLKOUT	When f_C is selected	Outputs clock	
	When f_8, f_{32} are selected	Outputs the clock when the CM02 bit in the CM0 register is set to "0" (peripheral function clock does not stop in wait mode). Maintains state immediately before entering wait mode when the CM02 bit is set to "1" (peripheral function clock stops in wait mode).	

NOTES:

1. M32C/86T cannot be used in memory expansion mode and microprocessor mode.

9.5.2.4 Exiting Wait Mode

Wait mode is exited by the hardware reset, $\overline{NMI}$ interrupt or peripheral function interrupts.

When the hardware reset or $\overline{NMI}$ interrupt, but not the peripheral function interrupts, is used to exit wait mode, set the ILVL2 to ILVL0 bits for the peripheral function interrupts to "000_2" (interrupt disabled) before executing the WAIT instruction.

CM02 bit setting affects the peripheral function interrupts. When the CM02 bit in the CM0 register is set to "0" (peripheral function clock does not stop in wait mode), all peripheral function interrupts can be used to exit wait mode. When the CM02 bit is set to "1" (peripheral function clock stops in wait mode), peripheral functions using the peripheral function clock stop. Therefore, the peripheral function interrupts cannot be used to exit wait mode. However, the peripheral function interrupts caused by an external clock, f_{C32}, or f_{2n} whose count source is the X_{IN} clock or on-chip oscillator clock, can be used to exit wait mode.

The CPU clock used when exiting wait mode by the peripheral function interrupts or $\overline{NMI}$ interrupt is the same CPU clock used when the WAIT instruction is executed.

Table 9.8 shows interrupts to be used to exit wait mode and usage conditions.

Table 9.8 Interrupts to Exit Wait Mode

Interrupt	When CM02=0	When CM02=1
NMI Interrupt	Available	Available
Serial I/O Interrupt	Available when the internal and external clocks are used	Available when the external clock or f_{2n} (when X_{IN} clock or on-chip oscillator is selected) is used
Key Input Interrupt	Available	Available
A/D Conversion Interrupt	Available in single or single-sweep mode	Do not use
Timer A Interrupt Timer B Interrupt	Available in all modes	Available in event counter mode or when count source is f_{C32} or f_{2n} (when X_{IN} clock or on-chip oscillator is selected)
INT Interrupt	Available	Available
Low Voltage Detection Interrupt	Available	Available
CAN Interrupt	Available	Do not use
Intelligent I/O Interrupt	Available	Do not use

9.5.3 Stop Mode

In stop mode, all oscillators and resonators stop. The CPU clock and peripheral function clock, as well as the CPU and peripheral functions operated by these clocks, also stop. The least power required to operate the microcomputer is in stop mode. The internal RAM holds its data when the voltage applied to the V_{CC} pin is more than or equal to V_{RAM}. If the voltage applied to the V_{CC} pin is 2.7V or less, the voltage must be $Vcc \geq V_{RAM}$.

The following interrupts can be used to exit stop mode:

- $\overline{\text{NMI}}$ interrupt
- Key Input Interrupt
- $\overline{\text{INT}}$ interrupt
- Timer A and B interrupt (Available when the timer counts external pulse, having its 100Hz or less frequency, in event counter mode)
- Low voltage detection interrupt (Refer to **6.1 Low Voltage Detection Interrupt** for usage conditions)

9.5.3.1 Entering Stop Mode

Stop mode is entered when setting the CM10 bit in the CM10 register to "1" (all clocks stops). The MCD4 to MCD0 bits in the MCD register become set to "01000_2" (divide-by-8 mode).
Enter stop mode after setting the followings.

- Initial Setting

  Set each interrupt priority level after setting the exit priority level required to exit stop mode, controlled by the RLVL2 to RLVL0 bits in the RLVL register, to "7".

- Before Entering stop mode

  (1) Set the I flag to "0"
  (2) Set the interrupt priority level of the interrupt being used to exit stop mode
  (3) Set the interrupt priority levels of the interrupts, not being used to exit stop mode, to "0"
  (4) Set IPL in the FLG register. Then set the exit priority level to the same level as IPL
  Interrupt priority level of the interrupt used to exit stop mode > IPL = the exit priority level
  (5) Set the PRC0 bit in the PRCR register to "1" (write enable)
  (6) Select the main clock as the CPU clock

  - When the CPU clock source is the sub clock,
    (a) set the CM05 bit in the CM0 register to "0" (main clock oscillates)
    (b) set the CM07 bit in the CM0 register to "0" (clock selected by the CM21 bit divided by MCD register setting)
  - When the CPU clock source is the PLL clock,
    (a) set the CM17 bit in the CM1 register to "0" (main clock)
    (b) set the PLC07 bit in the PLC0 register to "0" (PLL off)
  - When main clock direct mode is used,
    (a) set the PRC1 bit in the PRCR register to "1" (write enable)
    (b) set the PM24 bit in the PM2 register to "0" (clock selected by the CM07 bit)
  - When the CPU clock source is the on-chip oscillator clock,
    (a) set MCD4 to MCD0 bits to "01000_2" (divide-by-8 mode)
    (b) set the CM05 bit to "0" (main clock oscillates)
    (c) set the CM21 bit in the CM2 register to "0" (clock selected by the CM17 bit)

  (7) The oscillation stop detect function is used, set the CM20 bit in the CM2 register to "0" (oscillation stop detect fucntion disabled)
  (8) Set the I flag to "1"
  (9) Set the CM10 bit to "1" (all clocks stops)

- After Exiting Stop Mode

  Set the exit priority level to "7" as soon as exiting stop mode.

9.5.3.2 Exiting Stop Mode

Stop mode is exited by the hardware reset, $\overline{\text{NMI}}$ interrupt or peripheral function interrupts (key input interrupt and $\overline{\text{INT}}$ interrupt).

When the hardware reset or $\overline{\text{NMI}}$ interrupt, but not the peripheral function interrupts, is used to exit wait mode, set all ILVL2 to ILVL0 bits in the interrupt control registers for the peripheral function interrupt to "000_2" (interrupt disabled) before setting the CM10 bit to "1" (all clocks stops).

9.5.3.3 Pin Status in Stop Mode

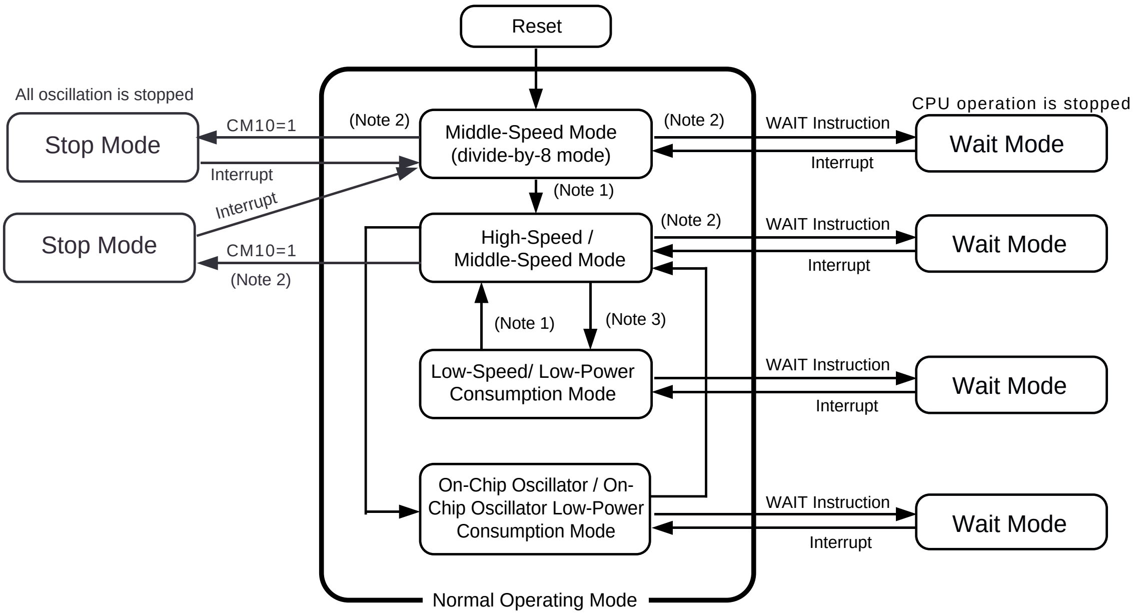
Table 9.9 lists pin status in stop mode.

Table 9.9 Pin Status in Stop Mode

Pin		Memory Expansion Mode[1] Microprocessor Mode[1]	Single-Chip Mode
Address Bus, Data Bus, $\overline{\text{CS0}}$ to $\overline{\text{CS3}}$, $\overline{\text{BHE}}$		Maintains state immediately before entering stop mode	
$\overline{\text{RD}}$, $\overline{\text{WR}}$, $\overline{\text{WRL}}$, $\overline{\text{WRH}}$		"H"	
$\overline{\text{HLDA}}$, BCLK		"H"	
ALE		"H"	
Ports		Maintains state immediately before entering stop mode	
CLKOUT	When f_C selected	"H"	
	When f_8, f_{32} selected	Maintains state immediately before entering stop mode	
XIN		Placed in a high-impedance state	
XOUT		"H"	
XCIN, XCOUT		Placed in a high-impedance state	

NOTES:

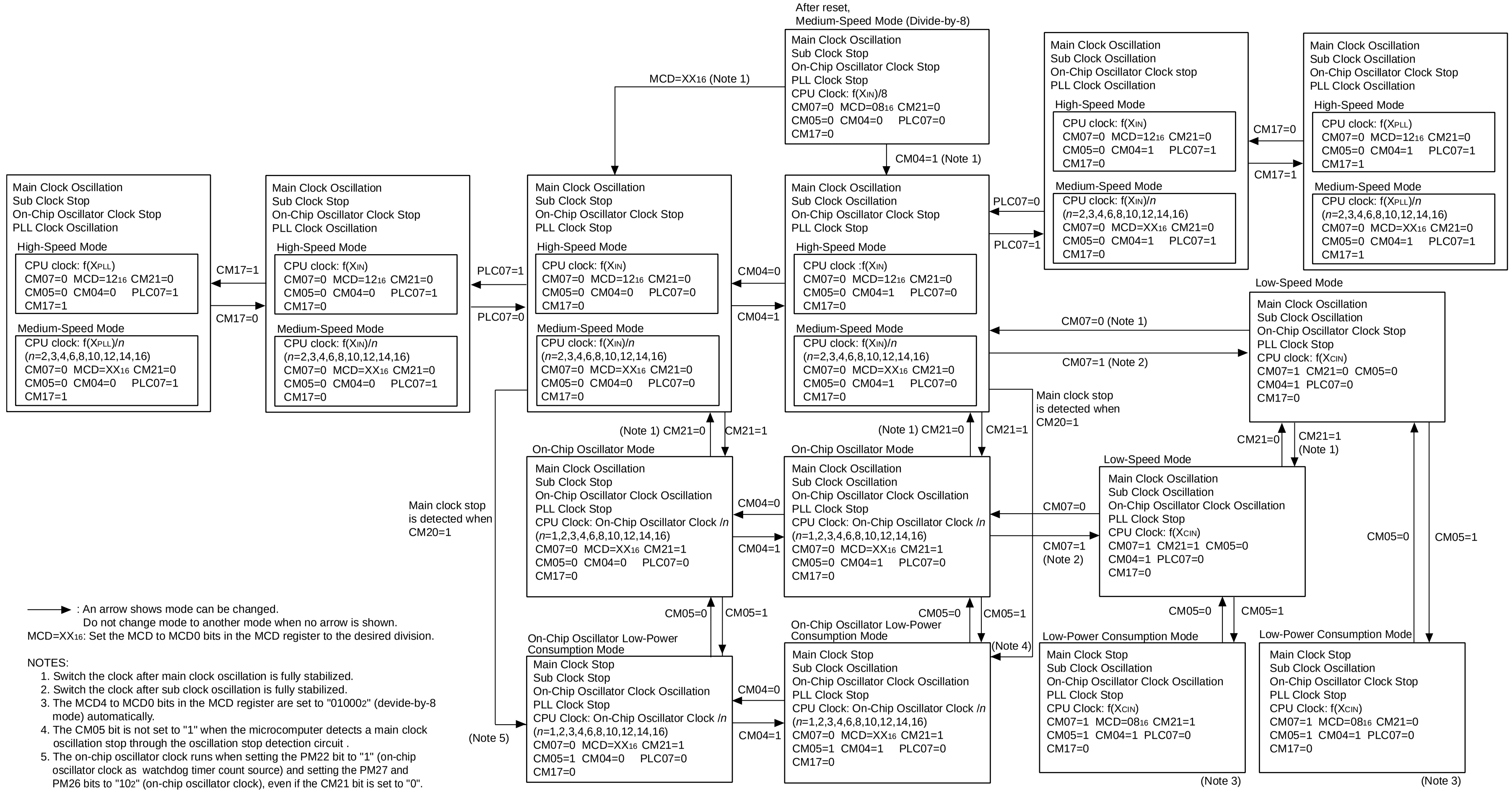
1. M32C/86T cannot be used in memory expansion mode and microprocessor mode.



NOTES:

1. See Figure 9.14.
2. When the CM17 bit is set to "1" (PLL clock as CPU clock source), set the CM17 bit to "0"(main clock as CPU clock source) and the PLC07 bit is set to "0" (PLL off). Then enter wait mode or stop mode.
3. When the CM17 bit is set to "1" (PLL clock as CPU clock source), set the CM17 bit to "0"(main clock as CPU clock source) and the PLC07 bit is set to "0" (PLL off). Then enter low-speed or low-power consumption mode.

Figure 9.13 Status Transition in Wait Mode and Stop Mode



→ : An arrow shows mode can be changed.
Do not change mode to another mode when no arrow is shown.

MCD=XX_{16}: Set the MCD to MCD0 bits in the MCD register to the desired division.

NOTES:
1. Switch the clock after main clock oscillation is fully stabilized.
2. Switch the clock after sub clock oscillation is fully stabilized.
3. The MCD4 to MCD0 bits in the MCD register are set to "01000_2" (devide-by-8 mode) automatically.
4. The CM05 bit is not set to "1" when the microcomputer detects a main clock oscillation stop through the oscillation stop detection circuit .
5. The on-chip oscillator clock runs when setting the PM22 bit to "1" (on-chip oscillator clock as watchdog timer count source) and setting the PM27 and PM26 bits to "10_2" (on-chip oscillator clock), even if the CM21 bit is set to "0".

Figure 9.14 Status Transition

9.6 System Clock Protect Function

The system clock protect function prohibits the CPU clock from changing clock sources when the main clock is selected as the CPU clock source. This prevents the CPU clock from stopping the program crash. When the PM21 bit in the PM2 register is set to "1" (clock change disable), the following bits cannot be written to:

- The CM02 bit, CM05 bit and CM07 bit in the CM0 register
- The CM10 bit and CM17 bit in the CM1 register
- The CM20 bit in the CM2 register
- All bits in the PLC0 and PLC1 registers

The CPU clock continues running when the WAIT instruction is executed.

To use the system clock protect function, set the CM05 bit in the CM0 register to "0" (main clock oscillation) and CM07 bit to "0" (main clock as BCLK clock source) and follow the procedure below.

(1) Set the PRC1 bit in the PRCR register to "1" (write enable).
(2) Set the PM21 bit in the PM2 register to "1" (protects the clock).
(3) Set the PRC1 bit in the PRCR register to "0" (write disable).

When the PM21 bit is set to "1", do not execute the WAIT instruction.

10. Protection

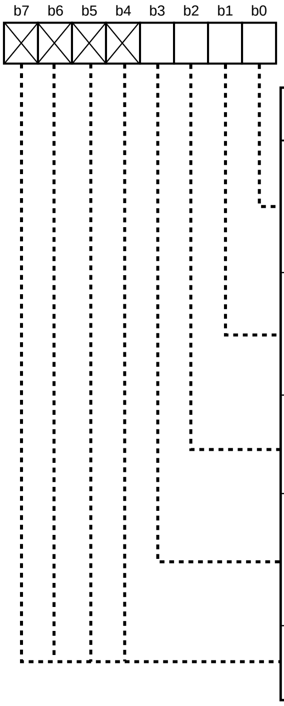
The protection function protects important registers from being easily overwritten when a program runs out of control.

Figure 10.1 shows the PRCR register. Each bit in the PRCR register protects the following registers:

- The PRC0 bit protects the CM0, CM1, CM2, MCD, PLC0 and PLC1 registers;
- The PRC1 bit protects the PM0, PM1, PM2, INVC0 and INVC1 registers;
- The PRC2 bit protects the PD9 and PS3 registers;
- The PRC3 bit protects the VCR2 and D4INT registers.

The PRC2 bit is set to "0" (write disable) when data is written to a desired address after setting the PRC2 bit to "1" (write enable). Set the PD9 and PS3 registers immediately after setting the PRC2 bit in the PRCR register to "1" (write enable). Do not generate an interrupt or a DMA transfer between the instruction to set to the PRC2 bit to "1" and the following instruction. The PRC0, PRC1 and PRC3 bits are not set to "0" even if data is written to desired addresses. Set the PRC0, PRC1 and PRC3 bits to "0" by program.

Protect Register



Symbol	Address	After Reset
PRCR	$000A_{16}$	$XXXX\ 0000_2$

Bit Symbol	Bit Name	Function	RW
PRC0	Protect Bit 0	Enables writing to CM0, CM1, CM2, MCD, PLC0, PLC1 registers 0 : Write disable 1 : Write enable	RW
PRC1	Protect Bit 1	Enables writing to PM0, PM1, PM2, INVC0, INVC1 registers 0 : Write disable 1 : Write enable	RW
PRC2	Protect Bit 2[1]	Enables writing to PD9, PS3 registers 0 : Write disable 1 : Write enable	RW
PRC3	Protect Bit 3	Enables writing to VCR2, D4INT registers 0 : Write disable 1 : Write enable	RW
—— (b7 - b4)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		—

NOTES:

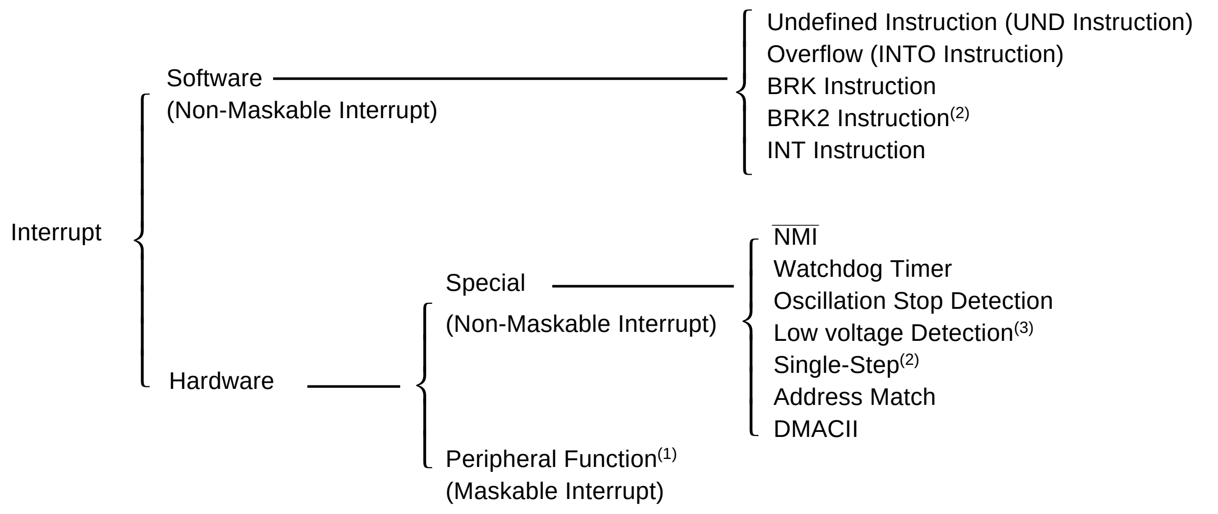
1. The PRC2 bit is set to "0" by writing into a desired address after the PRC2 bit is set to "1".
   The PRC0, PRC1 and PRC3 bits are not automatically set to "0". Set them to "0" by program.

Figure 10.1 PRCR Register

11. Interrupts

11.1 Types of Interrupts

Figure 11.1 shows types of interrupts.



NOTES:

1. The peripheral functions in the microcomputer are used to generate the peripheral interrupt.
2. Do not use this interrupt. For development support tools only.
3. Low voltage detection interrupt cannot be used in M32C/86T.

Figure 11.1 Interrupts

- Maskable Interrupt

  The I flag enables or disables an interrupt.

  The interrupt priority order based on interrupt priority level **can be changed**.

- Non-Maskable Interrupt

  The I flag does not enable nor disable an interrupt .

  The interrupt priority order based on interrupt priority level **cannot be changed**.

11.2 Software Interrupts

Software interrupt occurs when an instruction is executed. The software interrupts are non-maskable interrupts.

11.2.1 Undefined Instruction Interrupt

The undefined instruction interrupt occurs when the UND instruction is executed.

11.2.2 Overflow Interrupt

The overflow interrupt occurs when the O flag in the FLG register is set to "1" (overflow of arithmetic operation) and the INTO instruction is executed.
Instructions to set the O flag are :
ABS, ADC, ADCF, ADD, ADDX, CMP, CMPX, DIV, DIVU, DIVX, NEG, RMPA, SBB, SCMPU, SHA, SUB, SUBX

11.2.3 BRK Interrupt

The BRK interrupt occurs when the BRK instruction is executed.

11.2.4 BRK2 Interrupt

The BRK2 interrupt occurs when the BRK2 instruction is executed.
Do not use this interrupt. For development support tools only.

11.2.5 INT Instruction Interrupt

The INT instruction interrupt occurs when the INT instruction is executed. The INT instruction can select software interrupt numbers 0 to 63. Software interrupt numbers 8 to 49, 52 to 54 and 57 are assigned to the vector table used for the peripheral function interrupt. Therefore, the microcomputer executes the same interrupt routine when the INT instruction is executed as when a peripheral function interrupt occurs.
When the INT instruction is executed, the FLG register and PC are saved to the stack. PC also stores the relocatable vector of specified software interrupt numbers. Where the stack is saved varies depending on a software interrupt number. ISP is selected as the stack for software interrupt numbers 0 to 31 (setting the U flag to "0"). SP, which is set before the INT instruction is executed, is selected as the stack for software interrupt numbers 32 to 63 (the U flag is not changed).
With the peripheral function interrupt, the FLG register is saved and the U flag is set to "0" (ISP select) when an interrupt request is acknowledged. With software interrupt numbers 32 to 49, 52 to 54 and 57, SP to be used varies depending on whether the interrupt is generated by the peripheral function interrupt request or by the INT instruction.

11.3 Hardware Interrupts

Special interrupts and peripheral function interrupts are available as hardware interrupts.

11.3.1 Special Interrupts

Special interrupts are non-maskable interrupts.

11.3.1.1 $\overline{\text{NMI}}$ Interrupt

The $\overline{\text{NMI}}$ interrupt occurs when a signal applied to the $\overline{\text{NMI}}$ pin changes from a high-level ("H") signal to a low-level ("L") signal. Refer to **11.8 $\overline{\text{NMI}}$ Interrupt** for details.

11.3.1.2 Watchdog Timer Interrupt

The watchdog timer interrupt occurs when a count source of the watchdog timer underflows. Refer to **12. Watchdog Timer** for details.

11.3.1.3 Oscillation Stop Detection Interrupt

The oscillation stop detection interrupt occurs when the microcomputer detects a main clock oscillation stop. Refer to **9. Clock Generation Circuit** for details.

11.3.1.4 Low Voltage Detection Interrupt

The low voltage detection interrupt occurs when the voltage applied to VCC is above or below Vdet4. Refer to **6. Voltage Detection Circuit** for details.

NOTES:

1. Low voltage detection interrupt cannot be used in M32C/86T.

11.3.1.5 Single-Step Interrupt

Do not use the single-step interrupt. For development support tool only.

11.3.1.6 Address Match Interrupt

The address match interrupt occurs immediately before executing an instruction that is stored into an address indicated by the RMADi register (i=0 to 7) when the AIERi bit in the AIER register is set to "1" (address match interrupt enabled). Set the starting address of the instruction in the RMADi register. The address match interrupt does not occur when a table data or addresses of the instruction other than the starting address, if the instruction has multiple addresses, is set. Refer to **11.10 Address Match Interrupt** for details.

11.3.2 Peripheral Function Interrupt

The peripheral function interrupt occurs when a request from the peripheral functions in the microcomputer is acknowledged. The peripheral function interrupts and software interrupt numbers 8 to 49, 52 to 54 and 57 for the INT instruction use the same interrupt vector table. The peripheral function interrupt is a maskable interrupt.

See **Table 11.2** about how the peripheral function interrupt occurs. Refer to the descriptions of each function for details.

11.4 High-Speed Interrupt

The high-speed interrupt executes an interrupt sequence in five cycles and returns from the interrupt in three cycles.

When the FSIT bit in the RLVL register is set to "1" (interrupt priority level 7 available for the high-speed interrupt), the ILVL2 to ILVL0 bits in the interrupt control registers can be set to "111_2" (level 7) to use the high-speed interrupt.

Only one interrupt can be set as the high-speed interrupt. When using the high-speed interrupt, do not set multiple interrupts to interrupt priority level 7. Set the DMAII bit in the RLVL register to "0" (interrupt priority level 7 available for interrupts).

Set the starting address of the high-speed interrupt routine in the VCT register.

When the high-speed interrupt is acknowledged, the FLG register is saved into the SVF register and PC is saved into the SVP register. The program is executed from an address indicated by the VCT register.

Execute the FREIT instruction to return from the high-speed interrupt routine.

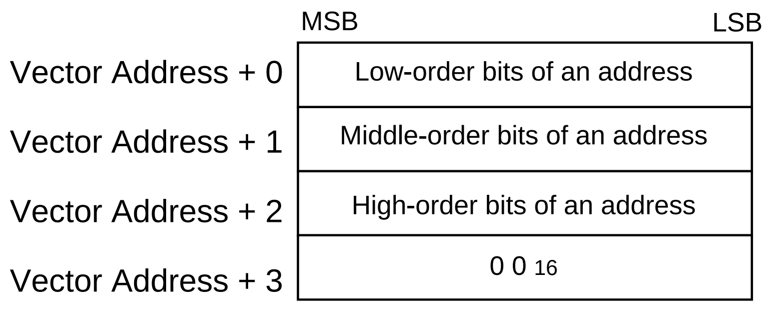
The values saved into the SVF and SVP registers are restored to the FLG register and PC by executing the FREIT instruction.

The high-speed interrupt and the DMA2 and DMA3 use the same register. When using the high-speed interrupt, neither DMA2 nor DMA3 is available. DMA0 and DMA1 can be used.

11.5 Interrupts and Interrupt Vectors

There are four bytes in one vector. Set the starting address of interrupt routine in each vector table. When an interrupt request is acknowledged, the interrupt routine is executed from the address set in the interrupt vectors.

Figure 11.2 shows the interrupt vector.



	MSB LSB
Vector Address + 0	Low-order bits of an address
Vector Address + 1	Middle-order bits of an address
Vector Address + 2	High-order bits of an address
Vector Address + 3	0 0$_{16}$

Figure 11.2 Interrupt Vector

11.5.1 Fixed Vector Tables

The fixed vector tables are allocated addresses FFFFDC$_{16}$ to FFFFFF$_{16}$. Table 11.1 lists the fixed vector tables. Refer to **25.2 Functions to Prevent Flash Memory from Rewriting** for fixed vectors of flash memory.

Table 11.1 Fixed Vector Table

Interrupt Generated by	Vector Addresses Address (L) to Address (H)	Remarks	Reference
Undefined Instruction	FFFFDC$_{16}$ to FFFFDF$_{16}$		M32C/80 Series Software Manual
Overflow	FFFFE0$_{16}$ to FFFFE3$_{16}$		
BRK Instruction	FFFFE4$_{16}$ to FFFFE7$_{16}$	If the content of address FFFFE7$_{16}$ is FF$_{16}$, a program is executed from the address stored into software interrupt number 0 in the relocatable vector table	
Address Match	FFFFE8$_{16}$ to FFFFEB$_{16}$		
-	FFFFEC$_{16}$ to FFFFEF$_{16}$	Reserved space	
Watchdog Timer	FFFFF0$_{16}$ to FFFFF3$_{16}$	These addresses are used for the watchdog timer interrupt, oscillation stop detection interrupt, and low voltage detection interrupt[1]	Reset, Clock Generation Circuit, Watchdog Timer
-	FFFFF4$_{16}$ to FFFFF7$_{16}$	Reserved space	
$\overline{\text{NMI}}$	FFFFF8$_{16}$ to FFFFFB$_{16}$		
Reset	FFFFFC$_{16}$ to FFFFFF$_{16}$		Reset

NOTES:

1. Low voltage detection interrupt cannot be used in M32C/86T.

11.5.2 Relocatable Vector Tables

The relocatable vector tables occupy 256 bytes from the starting address set in the INTB register. Table 11.2 lists the relocatable vector tables.

Set an even address as the starting address of the vector table set in the INTB register to increase interrupt sequence execution rate.

Table 11.2 Relocatable Vector Tables

Interrupt Generated by	Vector Table Address Address(L) to Address(H)[1]	Software Interrupt Number	Reference
BRK Instruction[2]	+0 to +3 (0000$_{16}$ to 0003$_{16}$)	0	M32C/80 Series Software Manual
Reserved Space	+4 to +31 (0004$_{16}$ to 001F$_{16}$)	1 to 7	
DMA0	+32 to +35 (0020$_{16}$ to 0023$_{16}$)	8	DMAC
DMA1	+36 to +39 (0024$_{16}$ to 0027$_{16}$)	9	
DMA2	+40 to +43 (0028$_{16}$ to 002B$_{16}$)	10	
DMA3	+44 to +47 (002C$_{16}$ to 002F$_{16}$)	11	
Timer A0	+48 to +51 (0030$_{16}$ to 0033$_{16}$)	12	Timer A
Timer A1	+52 to +55 (0034$_{16}$ to 0037$_{16}$)	13	
Timer A2	+56 to +59 (0038$_{16}$ to 003B$_{16}$)	14	
Timer A3	+60 to +63 (003C$_{16}$ to 003F$_{16}$)	15	
Timer A4	+64 to +67 (0040$_{16}$ to 0043$_{16}$)	16	
UART0 Transmission, NACK[3]	+68 to +71 (0044$_{16}$ to 0047$_{16}$)	17	Serial I/O
UART0 Reception, ACK[3]	+72 to +75 (0048$_{16}$ to 004B$_{16}$)	18	
UART1 Transmission, NACK[3]	+76 to +79 (004C$_{16}$ to 004F$_{16}$)	19	
UART1 Reception, ACK[3]	+80 to +83 (0050$_{16}$ to 0053$_{16}$)	20	
Timer B0	+84 to +87 (0054$_{16}$ to 0057$_{16}$)	21	Timer B
Timer B1	+88 to +91 (0058$_{16}$ to 005B$_{16}$)	22	
Timer B2	+92 to +95 (005C$_{16}$ to 005F$_{16}$)	23	
Timer B3	+96 to +99 (0060$_{16}$ to 0063$_{16}$)	24	
Timer B4	+100 to +103 (0064$_{16}$ to 0067$_{16}$)	25	
$\overline{\text{INT5}}$	+104 to +107 (0068$_{16}$ to 006B$_{16}$)	26	Interrupt
$\overline{\text{INT4}}$	+108 to +111 (006C$_{16}$ to 006F$_{16}$)	27	
$\overline{\text{INT3}}$	+112 to +115 (0070$_{16}$ to 0073$_{16}$)	28	
$\overline{\text{INT2}}$	+116 to +119 (0074$_{16}$ to 0077$_{16}$)	29	
$\overline{\text{INT1}}$	+120 to +123 (0078$_{16}$ to 007B$_{16}$)	30	
$\overline{\text{INT0}}$	+124 to +127 (007C$_{16}$ to 007F$_{16}$)	31	
Timer B5	+128 to +131 (0080$_{16}$ to 0083$_{16}$)	32	Timer B
UART2 Transmission, NACK[3]	+132 to +135 (0084$_{16}$ to 0087$_{16}$)	33	Serial I/O
UART2 Reception, ACK[3]	+136 to +139 (0088$_{16}$ to 008B$_{16}$)	34	
UART3 Transmission, NACK[3]	+140 to +143 (008C$_{16}$ to 008F$_{16}$)	35	
UART3 Reception, ACK[3]	+144 to +147 (0090$_{16}$ to 0093$_{16}$)	36	
UART4 Transmission, NACK[3]	+148 to +151 (0094$_{16}$ to 0097$_{16}$)	37	
UART4 Reception, ACK[3]	+152 to +155 (0098$_{16}$ to 009B$_{16}$)	38	

Table 11.2 Relocatable Vector Tables (Continued)

Interrupt Generated by	Vector Table Address Address(L) to Address(H)[1]	Software Interrupt Number	Reference
Bus Conflict Detect, Start Condition Detect, Stop Condition Detect (UART2)[3],	+156 to +159 (009C$_{16}$ to 009F$_{16}$)	39	Serial I/O
Bus Conflict Detect, Start Condition Detect, Stop Condition Detect (UART3/UART0)[4]	+160 to +163 (00A0$_{16}$ to 00A3$_{16}$)	40	
Bus Conflict Detect, Start Condition Detect, Stop Condition Detect (UART4/UART1)[4]	+164 to +167 (00A4$_{16}$ to 00A7$_{16}$)	41	
A/D0	+168 to +171 (00A8$_{16}$ to 00AB$_{16}$)	42	A/D Converter
Key Input	+172 to +175 (00AC$_{16}$ to 00AF$_{16}$)	43	Interrupts
Intelligent I/O Interrupt 0, CAN 3	+176 to +179 (00B0$_{16}$ to 00B3$_{16}$)	44	Intelligent I/O CAN
Intelligent I/O Interrupt 1, CAN 4	+180 to +183 (00B4$_{16}$ to 00B7$_{16}$)	45	
Intelligent I/O Interrupt 2	+184 to +187 (00B8$_{16}$ to 00BB$_{16}$)	46	
Intelligent I/O Interrupt 3	+188 to +191 (00BC$_{16}$ to 00BF$_{16}$)	47	
Intelligent I/O Interrupt 4	+192 to +195 (00C0$_{16}$ to 00C3$_{16}$)	48	
CAN 5	+196 to +199 (00C4$_{16}$ to 00C7$_{16}$)	49	CAN
Reserved Space	+200 to +207 (00C8$_{16}$ to 00CF$_{16}$)	50, 51	—
Intelligent I/O Interrupt 8	+208 to +211 (00D0$_{16}$ to 00D3$_{16}$)	52	Intelligent I/O CAN
Intelligent I/O Interrupt 9, CAN 0	+212 to +215 (00D4$_{16}$ to 00D7$_{16}$)	53	
Intelligent I/O Interrupt 10, CAN 1	+216 to +219 (00D8$_{16}$ to 00DB$_{16}$)	54	
Reserved Space	+220 to +227 (00DC$_{16}$ to 00E3$_{16}$)	55, 56	—
CAN 2	+228 to +231 (00E4$_{16}$ to 00E7$_{16}$)	57	CAN
Reserved Space	+232 to +255 (00E8$_{16}$ to 00FF$_{16}$)	58 to 63	—
INT Instruction[2]	+0 to +3 (0000$_{16}$ to 0003$_{16}$) to +252 to +255 (00FC$_{16}$ to 00FF$_{16}$)	0 to 63	Interrupts

NOTES:

1. These addresses are relative to those in the INTB register.
2. The I flag does not disable interrupts.
3. In I^{2}C mode, NACK, ACK or start/stop condition detection causes interrupts to be generated.
4. The IFSR6 bit in the IFSR register determines whether these addresses are used for an interrupt in UART0 or in UART3.
   The IFSR7 bit in the IFSR register determines whether these addresses are used for an interrupt in UART1 or in UART4.

11.6 Interrupt Request Acknowledgement

Software interrupts and special interrupts occur when conditions to generate an interrupt are met.
The peripheral function interrupts are acknowledged when all conditions below are met.

- I flag = "1"
- IR bit = "1"
- ILVL2 to ILVL0 bits > IPL

The I flag, IPL, IR bit and ILVL2 to ILVL0 bits are independent of each other. The I flag and IPL are in the FLG register. The IR bit and ILVL2 to ILVL0 bits are in the interrupt control register.

11.6.1 I Flag and IPL

The I flag enables or disables maskable interrupts. When the I flag is set to "1" (enable), all maskable interrupts are enabled; when the I flag is set to "0" (disable), they are disabled. The I flag is automatically set to "0" after reset.
IPL, consisting of three bits, indicates the interrupt priority level from level 0 to level 7.
If a requested interrupt has higher priority level than indicated by IPL, the interrupt is acknowledged.
Table 11.3 lists interrupt priority levels associated with IPL.

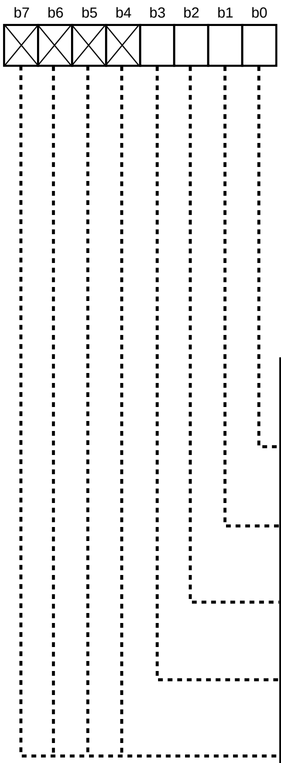
Table 11.3 Interrupt Priority Levels

IPL2	IPL1	IPL0	Interrupt Priority Levels
0	0	0	Level 1 and above
0	0	1	Level 2 and above
0	1	0	Level 3 and above
0	1	1	Level 4 and above
1	0	0	Level 5 and above
1	0	1	Level 6 and above
1	1	0	Level 7 and above
1	1	1	All maskable interrupts are disabled

11.6.2 Interrupt Control Register and RLVL Register

The peripheral function interrupts use interrupt control registers to control each interrupt. Figures 11.3 and 11.4 show the interrupt control register. Figure 11.5 shows the RLVL register.

Interrupt Control Register



Symbol	Address	After Reset
TA0IC to TA4IC	006C$_{16}$, 008C$_{16}$, 006E$_{16}$, 008E$_{16}$, 0070$_{16}$	XXXX X000$_2$
TB0IC to TB5IC	0094$_{16}$, 0076$_{16}$, 0096$_{16}$, 0078$_{16}$, 0098$_{16}$, 0069$_{16}$	XXXX X000$_2$
S0TIC to S4TIC	0090$_{16}$, 0092$_{16}$, 0089$_{16}$, 008B$_{16}$, 008D$_{16}$	XXXX X000$_2$
S0RIC to S4RIC	0072$_{16}$, 0074$_{16}$, 006B$_{16}$, 006D$_{16}$, 006F$_{16}$	XXXX X000$_2$
BCN0IC to BCN4IC	0071$_{16}$, 0091$_{16}$, 008F$_{16}$, 0071$_{16}$(1), 0091$_{16}$(2)	XXXX X000$_2$
DM0IC to DM3IC	0068$_{16}$, 0088$_{16}$, 006A$_{16}$, 008A$_{16}$	XXXX X000$_2$
AD0IC	0073$_{16}$	XXXX X000$_2$
KUPIC	0093$_{16}$	XXXX X000$_2$
IIO0IC to IIO4IC	0075$_{16}$, 0095$_{16}$, 0077$_{16}$, 0097$_{16}$, 0079$_{16}$	XXXX X000$_2$
IIO8IC to IIO10IC	007D$_{16}$, 009D$_{16}$, 007F$_{16}$	XXXX X000$_2$
CAN0IC to CAN2IC	009D$_{16}$, 007F$_{16}$, 0081$_{16}$(3)	XXXX X000$_2$
CAN3IC to CAN5IC	0075$_{16}$, 0095$_{16}$, 0099$_{16}$(3)	XXXX X000$_2$

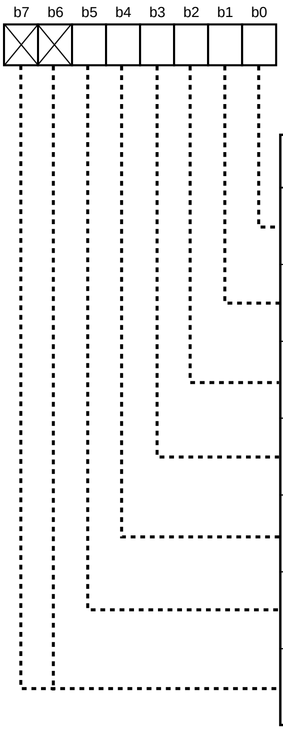
Bit Symbol	Bit Name	Function	RW
ILVL0	Interrupt Priority Level Select Bit	b2 b1 b0 0 0 0 : Level 0 (interrupt disabled) 0 0 1 : Level 1 0 1 0 : Level 2 0 1 1 : Level 3 1 0 0 : Level 4 1 0 1 : Level 5 1 1 0 : Level 6 1 1 1 : Level 7	RW
ILVL1			RW
ILVL2			RW
IR	Interrupt Request Bit	0 : No interrupt requested 1 : Interrupt requested(4)	RW
____ (b7 - b4)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		_

NOTES:

1. The BCN0IC register shares an address with the BCN3IC register.
2. The BCN1IC register shares an address with the BCN4IC register.
3. The IIO9IC register shares an address with the CAN0IC register.
   The IIO10IC register shares an address with the CAN1IC register.
   The IIO0IC register shares an address with the CAN3IC register.
   The IIO1IC register shares an address with the CAN4IC register.
4. The IR bit can be set to "0" only (do not set to "1").

Figure 11.3 Interrupt Control Register (1)

Interrupt Control Register



Symbol	Address	After Reset
INT0IC to INT2IC	009E$_{16}$, 007E$_{16}$, 009C$_{16}$	XX00 X000$_2$
INT3IC to INT5IC[1]	007C$_{16}$, 009A$_{16}$, 007A$_{16}$	XX00 X000$_2$

Bit Symbol	Bit Name	Function	RW
ILVL0	Interrupt Priority Level Select Bit	b2 b1 b0 0 0 0 : Level 0 (interrupt disabled) 0 0 1 : Level 1 0 1 0 : Level 2 0 1 1 : Level 3 1 0 0 : Level 4 1 0 1 : Level 5 1 1 0 : Level 6 1 1 1 : Level 7	RW
ILVL1			RW
ILVL2			RW
IR	Interrupt Request Bit	0 : Requests no interrupt 1 : Requests an interrupt[2]	RW
POL	Polarity Switch Bit	0 : Selects falling edge or "L"[3] 1 : Selects rising edge or "H"	RW
LVS	Level Sensitive/Edge Sensitive Switch Bit	0 : Edge sensitive 1 : Level sensitive[4]	RW
(b7 - b6)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		—

NOTES:

1. When a 16-bit data bus is used in microprocessor or memory expansion mode, each $\overline{\text{INT3}}$ to $\overline{\text{INT5}}$ pin is used as the data bus. Set the ILVL2 to ILVL0 bits in the INT3IC, INT4IC and INT5IC registers to "000$_2$".
2. The IR bit can be set to "0" only (do not set to "1").
3. Set the POL bit to "0" when a corresponding bit in the IFSR register is set to "1" (both edges).
4. When setting the LVS bit to "1" , set a corresponding bit in the IFSR register to "0" (one edge).

Figure 11.4 Interrupt Control Register (2)

11.6.2.1 ILVL2 to ILVL0 Bits

The ILVL2 to ILVL0 bits determines an interrupt priority level. The higher the interrupt priority level is, the higher interrupt priority is.

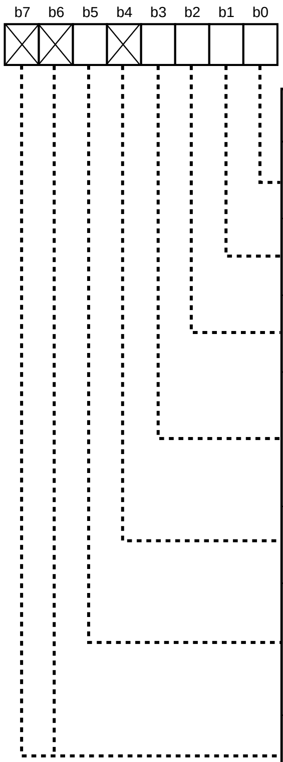
When an interrupt request is generated, its interrupt priority level is compared to IPL. This interrupt is acknowledged only when its interrupt priority level is higher than IPL. When the ILVL2 to ILVL0 bits are set to "000$_2$" (level 0), its interrupt is ignored.

11.6.2.2 IR Bit

The IR bit is automatically set to "1" (interrupt requested) when an interrupt request is generated. The IR bit is automatically set to "0" (no interrupt requested) after an interrupt request is acknowledged and an interrupt routine in the corresponding interrupt vector is executed.

The IR bit can be set to "0" by program. Do not set to "1".

Exit Priority Register



Symbol	Address	After Reset
RLVL	009F$_{16}$	XXXX 0000$_2$

Bit Symbol	Bit Name	Function	RW
RLVL0	Stop/Wait Mode Exit Minimum Interrupt Priority Level Control Bit(1)	b2 b1 b0 0 0 0 : Level 0 0 0 1 : Level 1 0 1 0 : Level 2 0 1 1 : Level 3 1 0 0 : Level 4 1 0 1 : Level 5 1 1 0 : Level 6 1 1 1 : Level 7	RW
RLVL1			RW
RLVL2			RW
FSIT	High-Speed Interrupt Set Bit(2)	0: Interrupt priority level 7 is used for normal interrupt 1: Interrupt priority level 7 is used for high-speed interrupt	RW
____ (b4)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		–
DMAII	DMA II Select Bit(4)	0: Interrupt priority level 7 is used for interrupt 1: Interrupt priority level 7 is used for DMA II transfer(3)	RW
____ (b7 - b6)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		–

NOTES:

1. The microcomputer exits stop or wait mode when the requested interrupt priority level is higher than the level set in the RLVL2 to RLVL0 bits. Set the RLVL2 to RLVL0 bits to the same value as IPL in the FLG register.
2. When the FSIT bit is set to "1", an interrupt having the interrupt priority level 7 becomes the high-speed interrupt. In this case, set only one interrupt to the interrupt priority level 7 and the DMAII bit to "0".
3. Set the ILVL2 to ILVL0 bits in the interrupt control register after setting the DMAII bit to "1". Do not change the DMAII bit setting to "0" after setting the DMAII bit to "1". Set the FSIT bit to "0" when the DMAII bit to "1".
4. The DMAII bit becomes indeterminate after reset. To use the DMAII bit for an interrupt setting, set it to "0" before setting the interrupt control register.

Figure 11.5 RLVL Register

11.6.2.3 RLVL2 to RLVL0 Bits

When using an interrupt to exit stop or wait mode, refer to **9.5.2 Wait Mode** and **9.5.3 Stop Mode** for details.

11.6.3 Interrupt Sequence

The interrupt sequence is performed between an interrupt request acknowledgment and interrupt routine execution.

When an interrupt request is generated while an instruction is executed, the CPU determines its interrupt priority level after the instruction is completed. The CPU starts the interrupt sequence from the following cycle. However, in regards to the SCMPU, SIN, SMOVB, SMOVF, SMOVU, SSTR, SOUT or RMPA instruction, if an interrupt request is generated while executing the instruction, the microcomputer suspends the instruction to start the interrupt sequence.

The interrupt sequence is performed as follows:

(1) The CPU obtains interrupt information (interrupt number and interrupt request level) by reading address 000000_{16} (address 000002_{16} for the high-speed interrupt). Then, the IR bit applicable to the interrupt information is set to "0" (interrupt requested).

(2) The FLG register, prior to an interrupt sequence, is saved to a temporary register(1) within the CPU.

(3) Each bit in the FLG register is set as follows:

- The I flag is set to "0" (interrupt disabled)
- The D flag is set to "0" (single-step disabled)
- The U flag is set to "0" (ISP selected)

(4) A temporary register within the CPU is saved to the stack; or to the SVF register for the high-speed interrupt.

(5) PC is saved to the stack; or to the SVP register for the high-speed interrupt.

(6) The interrupt priority level of the acknowledged interrupt is set in IPL .

(7) A relocatable vector corresponding to the acknowledged interrupt is stored into PC.

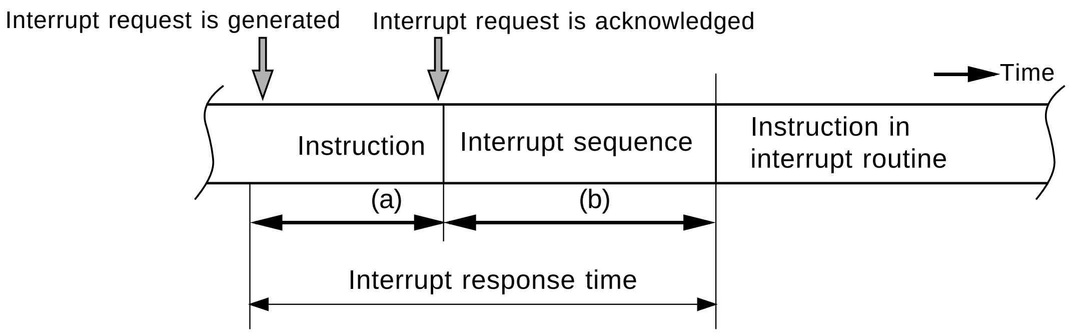
After the interrupt sequence is completed, an instruction is executed from the starting address of the interrupt routine.

NOTES:

1. Temporary register cannot be modified by users.

11.6.4 Interrupt Response Time

Figure 11.6 shows an interrupt response time. Interrupt response time is the period between an interrupt generation and the execution of the first instruction in an interrupt routine. Interrupt response time includes the period between an interrupt request generation and the completed execution of an instruction ((a) on Figure 11.6) and the period required to perform an interrupt sequence ((b) on Figure 11.6).


Interrupt request is generated
Interrupt request is acknowledged
Time
Instruction
Interrupt sequence
Instruction in interrupt routine
(a)
(b)
Interrupt response time

(a) Period between an interrupt request generation and the completed execution of an instruction.
(b) Period required to perform an interrupt sequence.

Figure 11.6 Interrupt Response Time

Time (a) varies depending on an instruction being executed. The DIV, DIVX and DIVU instructions require the longest time (a); 42 cycles when an immediate value or register is set as the divisor.
When the divisor is a value in the memory, the following value is added.

• Normal addressing	: 2 + X
• Index addressing	: 3 + X
• Indirect addressing	: 5 + X + 2Y
• Indirect index addressing	: 6 + X + 2Y

X is the number of wait states for a divisor space. Y is the number of wait states for the space that stores indirect addresses. If X and Y are in an odd address or in 8-bit bus space, the X and Y value must be doubled.
Table 11.4 lists time (b), shown Figure 11.6.

Table 11.4 Interrupt Sequence Execution Time

Interrupt	Interrupt Vector Address	16-Bit Bus	8-Bit Bus
Peripheral Function	Even address Odd address[(1)]	14 cycles 16 cycles	16 cycles 16 cycles
INT Instruction	Even address Odd address[(1)]	12 cycles 14 cycles	14 cycles 14 cycles
$\overline{\text{NMI}}$ Watchdog Timer Undefined Instruction Address Match	Even address[(2)]	13 cycles	15 cycles
Overflow	Even address[(2)]	14 cycles	16 cycles
BRK Instruction (relocatable vector table)	Even address Odd address[(1)]	17 cycles 19 cycles	19 cycles 19 cycles
BRK Instruction (fixed vector table)	Even address[(2)]	19 cycles	21 cycles
High-Speed Interrupt	Vector table is internal register	5 cycles	

NOTES:

1. Allocate interrupt vectors in even addresses.
2. Vectors are fixed to even addresses.

11.6.5 IPL Change when Interrupt Request is Acknowledged

When a peripheral function interrupt request is acknowledged, IPL sets the priority level for the acknowledged interrupt.

Software interrupts and special interrupts have no interrupt priority level. If an interrupt request that has no interrupt priority level is acknowledged, the value shown in Table 11.5 is set in IPL as the interrupt priority level.

Table 11.5 Interrupts without Interrupt Priority Levels and IPL

Interrupt Source	Level Set to IPL
Watchdog Timer, $\overline{\text{NMI}}$, Oscillation Stop Detection, Low Voltage Detection	7
Reset	0
Software, Address Match	Not changed

NOTES:

1. Low voltage detection interrupt cannot be used in M32C/86T.

11.6.6 Saving a Register

In the interrupt sequence, the FLG register and PC are saved to the stack.

After the FLG register is saved to the stack, 16 high-order bits and 16 low-order bits of PC, extended to 32 bits, are saved to the stack. Figure 11.7 shows stack states before and after an interrupt request is acknowledged.

Other important registers are saved by program at the beginning of an interrupt routine. The PUSHM instruction can save several registers[1] in the register bank used.

Refer to **11.4 High-Speed Interrupt** for the high-speed interrupt.

NOTES:

1. Can be selected from the R0, R1, R2, R3, A0, A1, SB and FB registers.

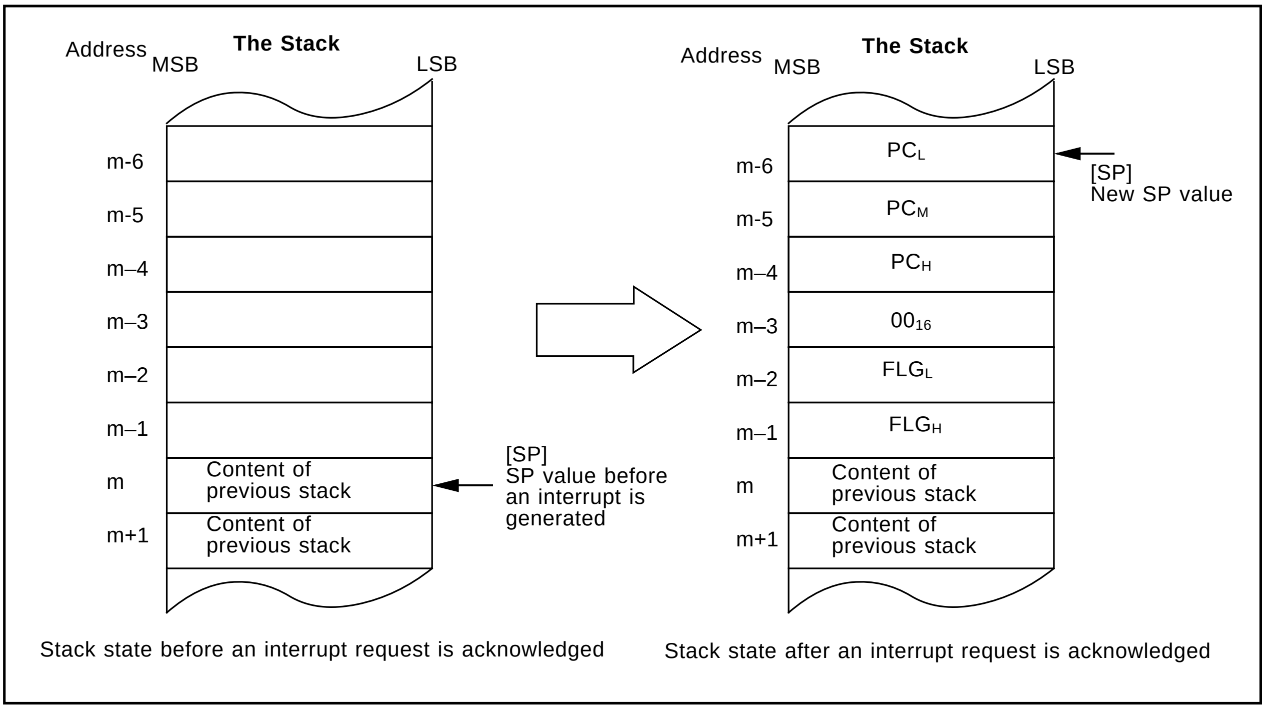

Address
The Stack
MSB
LSB
m-6
m-5
m–4
m–3
m–2
m–1
m
m+1
Content of previous stack
Content of previous stack
[SP]
SP value before an interrupt is generated
Address
The Stack
MSB
LSB
PC$_L$
PC$_M$
PC$_H$
00$_{16}$
FLG$_L$
FLG$_H$
[SP]
New SP value
Stack state before an interrupt request is acknowledged
Stack state after an interrupt request is acknowledged

Figure 11.7 Stack States

11.6.7 Restoration from Interrupt Routine

When the REIT instruction is executed at the end of an interrupt routine, the FLG register and PC before the interrupt sequence is performed, which have been saved to the stack, are automatically restored. The program, executed before an interrupt request was acknowledged, starts running again. Refer to **11.4 High-Speed Interrupt** for the high-speed interrupt.

Restore registers saved by program in an interrupt routine by the POPM instruction or others before the REIT and FREIT instructions. Register bank is switched back to the bank used prior to the interrupt sequence by the REIT or FREIT instruction.

11.6.8 Interrupt Priority

If two or more interrupt requests are sampled at the same sampling points (a timing to detect whether an interrupt request is generated or not), the interrupt with the highest priority is acknowledged.

Set the ILVL2 to ILVL0 bits to select the desired priority level for maskable interrupts (peripheral function interrupt).

Priority levels of special interrupts such as reset (reset has the highest priority) and watchdog timer are set by hardware. Figure 11.8 shows priority levels of hardware interrupts.

The interrupt priority does not affect software interrupts. Executing instruction causes the microcomputer to execute an interrupt routine.

Oscillation Stop Detection
Reset > $\overline{\text{NMI}}$ > Watchdog > Peripheral Function > Address Match
Low voltage Detection[1]

NOTES:

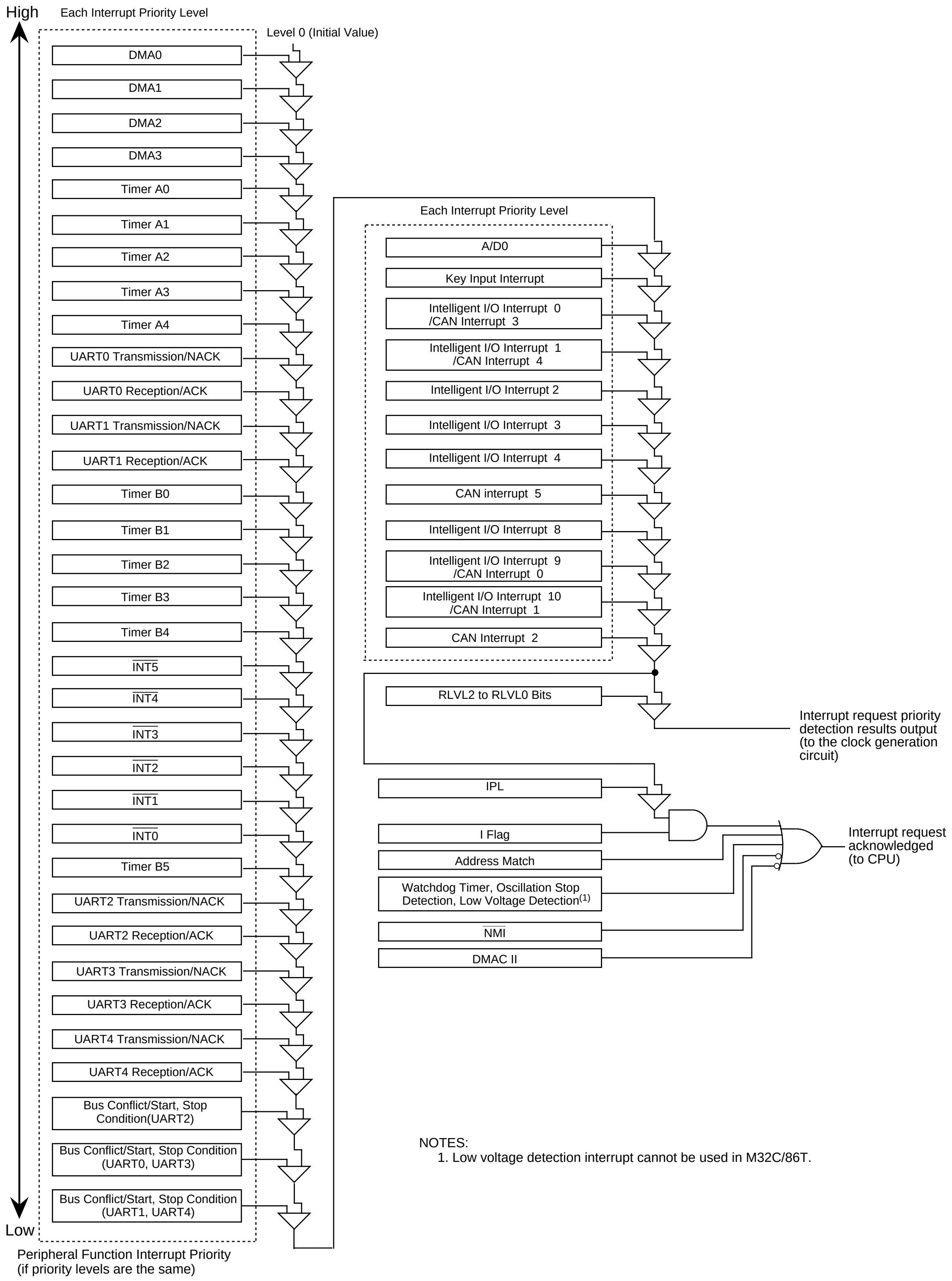
1. Low voltage detection interrupt cannot be used in M32C/86T.

Figure 11.8 Interrupt Priority

11.6.9 Interrupt Priority Level Select Circuit

The interrupt priority level select circuit selects the highest priority interrupt when two or more interrupt requests are sampled at the same sampling point.

Figure 11.9 shows the interrupt priority level select circuit.


High
Each Interrupt Priority Level
Level 0 (Initial Value)
DMA0
DMA1
DMA2
DMA3
Timer A0
Timer A1
Timer A2
Timer A3
Timer A4
UART0 Transmission/NACK
UART0 Reception/ACK
UART1 Transmission/NACK
UART1 Reception/ACK
Timer B0
Timer B1
Timer B2
Timer B3
Timer B4
INT5
INT4
INT3
INT2
INT1
INT0
Timer B5
UART2 Transmission/NACK
UART2 Reception/ACK
UART3 Transmission/NACK
UART3 Reception/ACK
UART4 Transmission/NACK
UART4 Reception/ACK
Bus Conflict/Start, Stop Condition(UART2)
Bus Conflict/Start, Stop Condition (UART0, UART3)
Bus Conflict/Start, Stop Condition (UART1, UART4)
Low
Peripheral Function Interrupt Priority (if priority levels are the same)
Each Interrupt Priority Level
A/D0
Key Input Interrupt
Intelligent I/O Interrupt 0 /CAN Interrupt 3
Intelligent I/O Interrupt 1 /CAN Interrupt 4
Intelligent I/O Interrupt 2
Intelligent I/O Interrupt 3
Intelligent I/O Interrupt 4
CAN interrupt 5
Intelligent I/O Interrupt 8
Intelligent I/O Interrupt 9 /CAN Interrupt 0
Intelligent I/O Interrupt 10 /CAN Interrupt 1
CAN Interrupt 2
RLVL2 to RLVL0 Bits
Interrupt request priority detection results output (to the clock generation circuit)
IPL
I Flag
Address Match
Watchdog Timer, Oscillation Stop Detection, Low Voltage Detection(1)
NMI
DMAC II
Interrupt request acknowledged (to CPU)

NOTES:
1. Low voltage detection interrupt cannot be used in M32C/86T.

Figure 11.9 Interrupt Priority Level Select Circuit

11.7 $\overline{\text{INT}}$ Interrupt

External input generates the $\overline{\text{INTi}}$ interrupt (i = 0 to 5). The LVS bit in the INTiIC register selects either edge sensitive triggering to generate an interrupt on any edge or level sensitive triggering to generate an interrupt at an applied signal level. The POL bit in the INTiIC register determines the polarity.

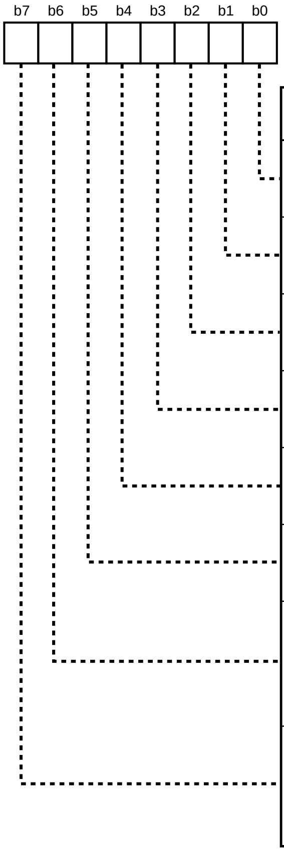
For edge sensitive, when the IFSRi bit in the IFSR register is set to "1", an interrupt occurs on both rising and falling edges of the external input. If the IFSRi bit is set to "1", set the POL bit in the corresponding register to "0" (falling edge).

For level sensitive, set the IFSRi bit to "0" (single edge). When the $\overline{\text{INTi}}$ pin input level reaches the level set in the POL bit, the IR bit in the INTiIC register is set to "1". The IR bit remains unchanged even if the $\overline{\text{INTi}}$ pin level is changed. The IR bit is set to "0" when the $\overline{\text{INTi}}$ interrupt is acknowledged or when the IR bit is written to "0" by program.

Figure 11.10 shows the IFSR register.

External Interrupt Request Source Select Register

Symbol	Address	After Reset
IFSR	031F$_{16}$	00$_{16}$



Bit Symbol	Bit Name	Function	RW
IFSR0	INT0 Interrupt Polarity Select Bit(1)	0 : One edge 1 : Both edges	RW
IFSR1	INT1 Interrupt Polarity Select Bit(1)	0 : One edge 1 : Both edges	RW
IFSR2	INT2 Interrupt Polarity Select Bit(1)	0 : One edge 1 : Both edges	RW
IFSR3	INT3 Interrupt Polarity Select Bit(1)	0 : One edge 1 : Both edges	RW
IFSR4	INT4 Interrupt Polarity select bit(1)	0 : One edge 1 : Both edges	RW
IFSR5	INT5 Interrupt Polarity Select Bit(1)	0 : One edge 1 : Both edges	RW
IFSR6	UART0, UART3 Interrupt Source Select Bit	0 : UART3 bus conflict, start condition detect, stop condition detect 1 : UART0 bus conflict, start condition detect, stop condition detect	RW
IFSR7	UART1, UART4 Interrupt Source Select Bit	0 : UART4 bus conflict, start condition detect, stop condition detect 1 : UART1 bus conflict, start condition detect, stop condition detect	RW

NOTES:
1. Set this bit to "0" to select a level-sensitive triggering.
When setting this bit to "1", set the POL bit in the INTiIC register (i = 0 to 5) to "0" (falling edge).

Figure 11.10 IFSR Register

11.8 $\overline{\text{NMI}}$ Interrupt[1]

The $\overline{\text{NMI}}$ interrupt occurs when a signal applied to the $\overline{\text{NMI}}$ pin changes from a high-level ("H") signal to a low-level ("L") signal. The $\overline{\text{NMI}}$ interrupt is a non-maskable interrupt. Although the P8$_5$/$\overline{\text{NMI}}$ pin is used as the $\overline{\text{NMI}}$ interrupt input pin, the P8_5 bit in the P8 register indicates the input level for this pin.

NOTES:

1. When the $\overline{\text{NMI}}$ interrupt is not used, connect the $\overline{\text{NMI}}$ pin to V$_{CC}$ via a resistor. Because the $\overline{\text{NMI}}$ interrupt cannot be ignored, the pin must be connected.

11.9 Key Input Interrupt

Key input interrupt request is generated when one of the signals applied to the P10$_4$ to P10$_7$ pins in input mode is on the falling edge. The key input interrupt can be also used as key-on wake-up function to exit wait or stop mode. To use the key input interrupt, do not use P10$_4$ to P10$_7$ as A/D input ports. Figure 11.11 shows a block diagram of the key input interrupt. When an "L" signal is applied to any pins in input mode, signals applied to other pins are not detected as an interrupt request signal.

When the PSC_7 bit in the PSC register[2] is set to "1" (key input interrupt disabled), no key input interrupt occurs regardless of interrupt control register settings. When the PSC_7 bit is set to "1", no input from a port pin is available even when in input mode.

NOTES:

2. Refer to **24. Programmable I/O Ports** about the PSC register.

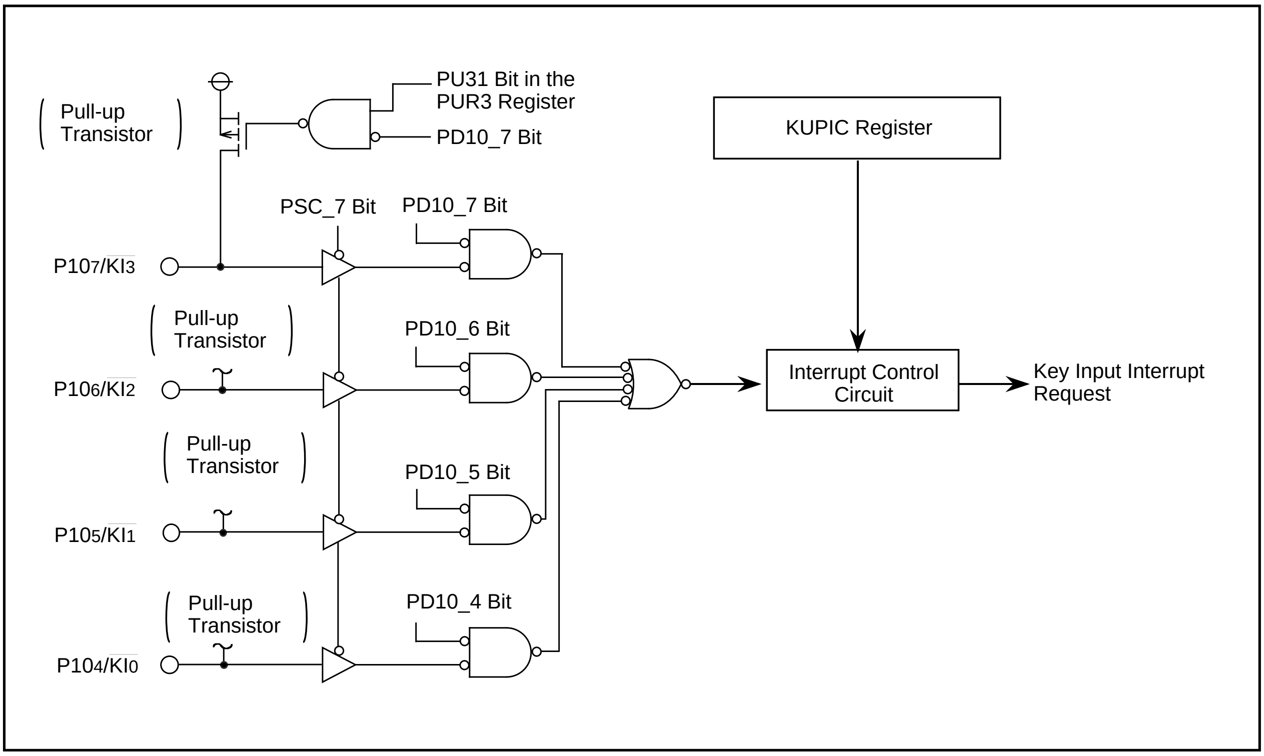

Pull-up Transistor
PU31 Bit in the PUR3 Register
PD10_7 Bit
KUPIC Register
PSC_7 Bit
PD10_7 Bit
P10$_7$/$\overline{\text{KI}_3}$
Pull-up Transistor
PD10_6 Bit
P10$_6$/$\overline{\text{KI}_2}$
Interrupt Control Circuit
Key Input Interrupt Request
Pull-up Transistor
PD10_5 Bit
P10$_5$/$\overline{\text{KI}_1}$
Pull-up Transistor
PD10_4 Bit
P10$_4$/$\overline{\text{KI}_0}$

Figure 11.11 Key Input Interrupt

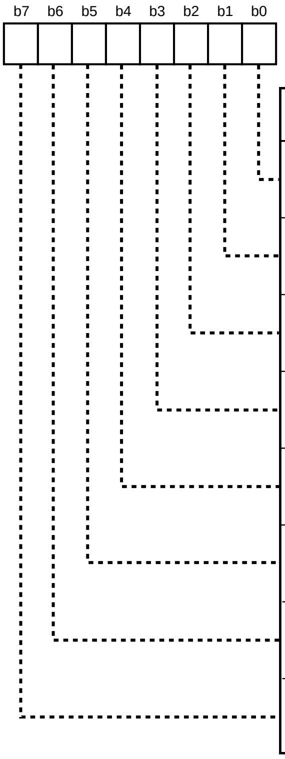
11.10 Address Match Interrupt

The address match interrupt occurs immediately before executing an instruction that is stored into an address indicated by the RMADi register (i=0 to 7). The address match interrupt can be set in eight addresses. The AIERi bit in the AIER register determines whether the interrupt is enabled or disabled. The I flag and IPL do not affect the address match interrupt.

Figure 11.12 shows registers associated with the address match interrupt.

The starting address of an instruction must be set in the RMADi register. The address match interrupt does not occur when a table data or addresses other than the starting address of the instruction is set.

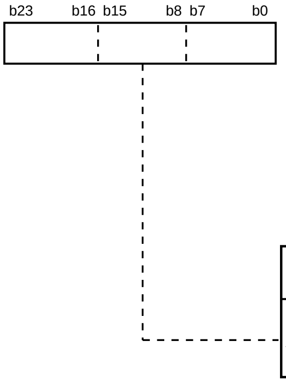
Address Match Interrupt Enable Register



Symbol	Address	After Reset
AIER	0009_{16}	$0000\ 0000_2$

Bit Symbol	Bit Name	Function	RW
AIER0	Address Match Interrupt 0 Enable Bit	0 : Disables the interrupt 1 : Enables the interrupt	RW
AIER1	Address Match Interrupt 1 Enable Bit	0 : Disables the interrupt 1 : Enables the interrupt	RW
AIER2	Address Match Interrupt 2 Enable Bit	0 : Disables the interrupt 1 : Enables the interrupt	RW
AIER3	Address Match Interrupt 3 Enable Bit	0 : Disables the interrupt 1 : Enables the interrupt	RW
AIER4	Address Match Interrupt 4 Enable Bit	0 : Disables the interrupt 1 : Enables the interrupt	RW
AIER5	Address Match Interrupt 5 Enable Bit	0 : Disables the interrupt 1 : Enables the interrupt	RW
AIER6	Address Match Interrupt 6 Enable Bit	0 : Disables the interrupt 1 : Enables the interrupt	RW
AIER7	Address Match Interrupt 7 Enable Bit	0 : Disables the interrupt 1 : Enables the interrupt	RW

Address Match Interrupt Register i (i=0 to 7)



Symbol	Address	After Reset
RMAD0	0012_{16} - 0010_{16}	000000_{16}
RMAD1	0016_{16} - 0014_{16}	000000_{16}
RMAD2	$001A_{16}$ - 0018_{16}	000000_{16}
RMAD3	$001E_{16}$ - $001C_{16}$	000000_{16}
RMAD4	$002A_{16}$ - 0028_{16}	000000_{16}
RMAD5	$002E_{16}$ - $002C_{16}$	000000_{16}
RMAD6	$003A_{16}$ - 0038_{16}	000000_{16}
RMAD7	$003E_{16}$ - $003C_{16}$	000000_{16}

Function	Setting Range	RW
Addressing Register for the Address Match Interrupt	000000_{16} to $FFFFFF_{16}$	RW

Figure 11.12 AIER Register and RMAD0 to RMAD7 Registers

11.11 Intelligent I/O Interrupt and CAN Interrupt

The intelligent I/O interrupt and CAN interrupt are assigned to software interrupt numbers 44 to 49, 52 to 54, and 57.

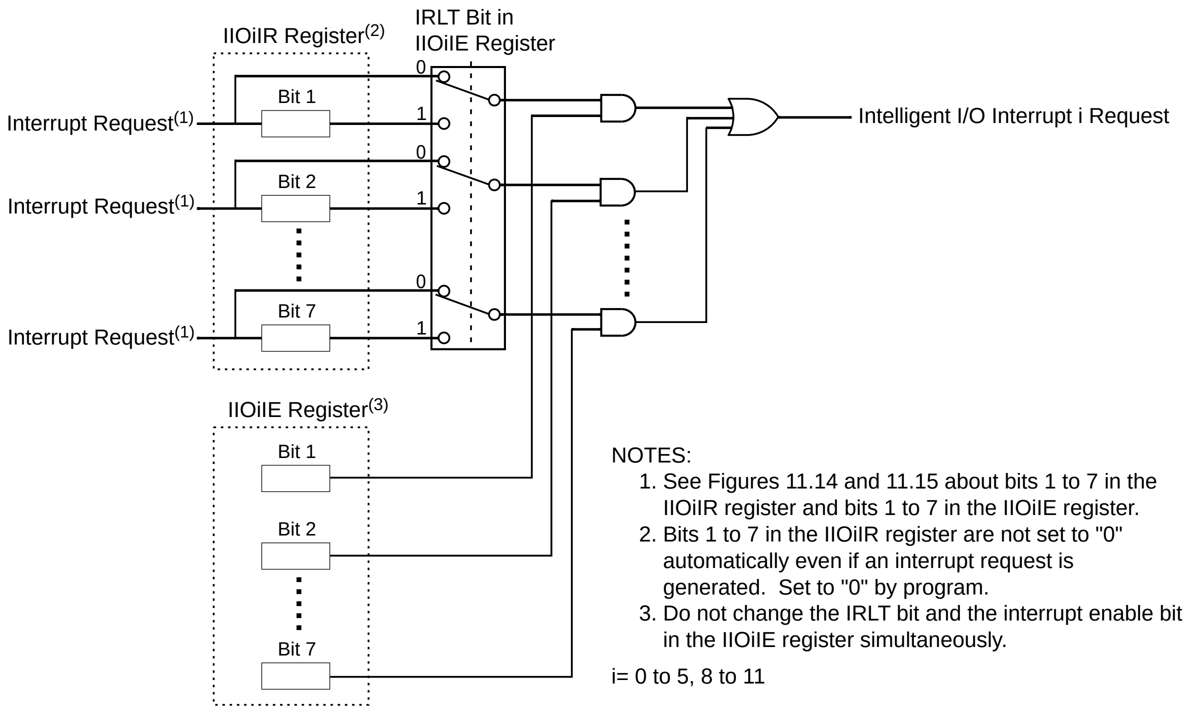
When using the intelligent I/O interrupt or CAN interrupt, set the IRLT bit in the IIOiIE register (i = 0 to 5, 8 to 11) to "1" (interrupt request for interrupt used).

Various interrupt requests cause the intelligent I/O interrupt to occur. When an interrupt request is generated with each intelligent I/O or CAN functions, the corresponding bit in the IIOiIR register is set to "1" (interrupt requested). When the corresponding bit in the IIOiIE register is set to "1" (interrupt enabled), the IR bit in the corresponding IIOiIC register is set to "1" (interrupt requested).

After the IR bit setting changes "0" to "1", the IR bit remains set to "1" when a bit in the IIOiIR register is set to "1" by another interrupt request and the corresponding bit in the IIOiIE register is set to "1".

Bits in the IIOiIR register are not set to "0" automatically, even if an interrupt is acknowledged. Set each bit to "0" by program. If these bit settings are left "1", all generated interrupt requests are ignored.

Figure 11.13 shows a block diagram of the intelligent I/O interrupt and CAN interrupt. Figure 11.14 shows the IIOiIR register. Figure 11.15 shows the IIOiIE register.



NOTES:
1. See Figures 11.14 and 11.15 about bits 1 to 7 in the IIOiIR register and bits 1 to 7 in the IIOiIE register.
2. Bits 1 to 7 in the IIOiIR register are not set to "0" automatically even if an interrupt request is generated. Set to "0" by program.
3. Do not change the IRLT bit and the interrupt enable bit in the IIOiIE register simultaneously.

i= 0 to 5, 8 to 11

Figure 11.13 Intelligent I/O Interrupt and CAN Interrupt

The CANjk (j=0 to 1, k=0 to 2) interrupt and CAN1 wake-up interrupt are provided as the CAN interrupt. The following registers are required for the CAN interrupts:

- Bits 7 in the IIO9IR to IIO11IR registers and Bits 7 in the IIO9IE to IIO11IE registers for the CAN00 to CAN02 interrupts.
- Bits 7 in the IIO0IR, IIO1IR and IIO5IR registers and Bits 7 in the IIO0IE, IIO1IE and IIO5IE registers for the CAN10 to CAN12 interrupts.
- Bit 6 in the IIO5IR register and Bit 6 in the IIO5IE register for the CAN1 wake-up interrupt.

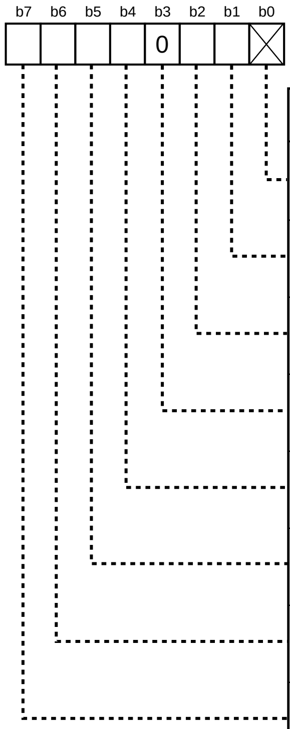
The CAN0IC, CAN1IC, CAN3IC and CAN4IC registers share addresses with the following registers:

- The CAN0IC register shares an address with the IIO9IC register.
- The CAN1IC register shares an address with the IIO10IC register.
- The CAN3IC register shares an address with the IIO0IC register.
- The CAN4IC register shares an address with the IIO1IC register.

Refer to **23.4 CAN Interrupt** for details.

When using the intelligent I/O interrupt or CAN interrupt to activate DMAC II, set the IRLT bit in the IIOiIE register to "0" (interrupt used for DMAC, DMAC II) to enable the interrupt request that the IIOiIE register requires.

Interrupt Request Register



Symbol	Address	After Reset
IIO0IR to IIO5IR, IIO8IR to IIO11IR	See below	0000 000X2

Bit Symbol	Function	RW
(b0)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.	—
(Note 1)	0 : Requests no interrupt 1 : Requests an interrupt(2)	RW
(Note 1)	0 : Requests no interrupt 1 : Requests an interrupt(2)	RW
(b3)	Reserved bit. Set to "0". When read, its content is indeterminate.	RW
(Note 1)	0 : Requests no interrupt 1 : Requests an interrupt(2)	RW
(Note 1)	0 : Requests no interrupt 1 : Requests an interrupt(2)	RW
(Note 1)	0 : Requests no interrupt 1 : Requests an interrupt(2)	RW
(Note 1)	0 : Requests no interrupt 1 : Requests an interrupt(2)	RW

NOTES:
1. See table below for bit symbols.
2. Only "0" can be set (nothing is changed even if "1" is set).

Bit Symbols for the Interrupt Request Register

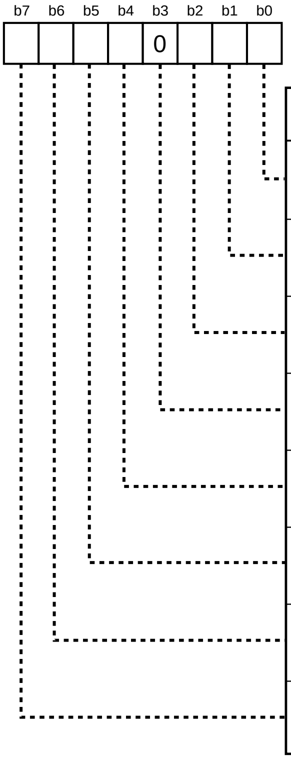
Symbol	Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IIO0IR	00A016	CAN10R	-	SIO0RR	G0RIR	-	TM13R/PO13R	-	-
IIO1IR	00A116	CAN11R	-	SIO0TR	G0TOR	-	TM14R/PO14R	-	-
IIO2IR	00A216	-	-	SIO1RR	G1RIR	-	TM12R/PO12R	-	-
IIO3IR	00A316	-	-	SIO1TR	G1TOR	-	TM10R/PO10R	-	-
IIO4IR	00A416	SRT0R	SRT1R	-	BT1R	-	TM17R/PO17R	-	-
IIO5IR	00A516	CAN12R	CAN1WUR	-	-	-	-	-	-
IIO8IR	00A816	-	-	-	-	-	-	TM11R/PO11R	-
IIO9IR	00A916	CAN00R	-	-	-	-	-	TM15R/PO15R	-
IIO10IR	00AA16	CAN01R	-	-	-	-	-	TM16R/PO16R	-
IIO11IR	00AB16	CAN02R	-	-	-	-	-	-	-

BT1R	: Intelligent I/O Base Timer Interrupt Request
TM1jR	: Intelligent I/O Time Measurement j Interrupt Request
PO1jR	: Intelligent I/O Waveform Generating Function j Interrupt Request
SIOiRR	: Intelligent I/O Communication Unit i Receive Interrupt Request
SIOiTR	: Intelligent I/O Communication Unit i Transmit Interrupt Request
GiTOR	: Intelligent I/O Communication Unit i HDLC Data Processing Function Interrupt Request (TO: Output to Transmit)
GiRIR	: Intelligent I/O Communication Unit i HDLC Data Processing Function Interrupt Request (RI: Input to Receive)
SRTiR	: Intelligent I/O Special Communication Function Interrupt Request
CAN0kR	: CAN0 Communication Function Interrupt Request (k = 0 to 2)
CAN1mR	: CAN1 Communication Function Interrupt Request (m = 0 to 2)
CAN1WUR	: CAN1 Wake-up Interrupt Request
-	: Reserved Bit. Set to "0".

i = 0, 1
j = 0 to 7

Figure 11.14 IIO0IR to IIO5IR, IIO8IR to IIO11IR Registers

Interrupt Enable Register



Symbol	Address	After Reset
IIO0IE to IIO5IE, IIO8IE to IIO11IE	See below	0000 0000$_2$

Bit Symbol	Bit Name	Function	RW
IRLT	Interrupt Request Select Bit[2]	0 : Interrupt request is used for DMAC, DMAC II 1 : Interrupt request is used for interrupt	RW
(Note 1)	———	0 : Disables an interrupt by bit 1 in IIOiIR register 1 : Enables an interrupt by bit 1 in IIOiIR register	RW
(Note 1)	———	0 : Disables an interrupt by bit 2 in IIOiIR register 1 : Enables an interrupt by bit 2 in IIOiIR register	RW
——— (b3)	Reserved Bit	Set to "0"	RW
(Note 1)	———	0 : Disables an interrupt by bit 4 in IIOiIR register 1 : Enables an interrupt by bit 4 in IIOiIR register	RW
(Note 1)	———	0 : Disables an interrupt by bit 5 in IIOiIR register 1 : Enables an interrupt by bit 5 in IIOiIR register	RW
(Note 1)	———	0 : Disables an interrupt by bit 6 in IIOiIR register 1 : Enables an interrupt by bit 6 in IIOiIR register	RW
(Note 1)	———	0 : Disables an interrupt by bit 7 in IIOiIR register 1 : Enables an interrupt by bit 7 in IIOiIR register	RW

NOTES:

1. See table below for bit symbols.
2. If an interrupt request is used for interrupt, set bit 1, 2, 4 to 7 to "1" after the IRLT bit is set to "1".

Bit Symbols for the Interrupt Enable Register

Symbol	Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IIO0IE	00B0$_{16}$	CAN10E	-	SIO0RE	G0RIE	-	TM13E/PO13E	-	IRLT
IIO1IE	00B1$_{16}$	CAN11E	-	SIO0TE	G0TOE	-	TM14E/PO14E	-	IRLT
IIO2IE	00B2$_{16}$	-	-	SIO1RE	G1RIE	-	TM12E/PO12E	-	IRLT
IIO3IE	00B3$_{16}$	-	-	SIO1TE	G1TOE	-	TM10E/PO10E	-	IRLT
IIO4IE	00B4$_{16}$	SRT0E	SRT1E	-	BT1E	-	TM17E/PO17E	-	IRLT
IIO5IE	00B5$_{16}$	CAN12E	CAN1WUE	-	-	-	-	-	IRLT
IIO8IE	00B8$_{16}$	-	-	-	-	-	-	TM11E/PO11E	IRLT
IIO9IE	00B9$_{16}$	CAN00E	-	-	-	-	-	TM15E/PO15E	IRLT
IIO10IE	00BA$_{16}$	CAN01E	-	-	-	-	-	TM16E/PO16E	IRLT
IIO11IE	00BB$_{16}$	CAN02E	-	-	-	-	-	-	IRLT

BT1E	: Intelligent I/O Base Timer Interrupt Enabled
TM1jE	: Intelligent I/O Time Measurement j Interrupt Enabled
PO1jE	: Intelligent I/O Waveform Generating Function j Interrupt Enabled
SIOiRE	: Intelligent I/O Communication Unit i Receive Interrupt Enabled
SIOiTE	: Intelligent I/O Communication Unit i Transmit Interrupt Enabled
GiTOE	: Intelligent I/O Communication Unit i HDLC Data Processing Function Interrupt Enabled (TO: Output to Transmit)
GiRIE	: Intelligent I/O Communication Unit i HDLC Data Processing Function Interrupt Enabled (RI: Input to Receive)
SRTiE	: Intelligent I/O Special Communication Function Interrupt Enabled
CAN0kE	: CAN0 Communication Function Interrupt Enabled (k = 0 to 2)
CAN1mE	: CAN1 Communication Function Interrupt Enabled (m = 0 to 2)
CAN1WUE	: CAN1 Wake-up Interrupt Enabled
-	: Reserved Bit. Set to "0".

i = 0, 1
j = 0 to 7

Figure 11.15 IIO0IE to IIO5IE, IIO8IE to IIO11IE Registers

12. Watchdog Timer

The watchdog timer monitors the program executions and detects defective program. It allows the microcomputer to trigger a reset or to generate an interrupt if the program error occurs. The watchdog timer contains a 15-bit counter, which is decremented by the CPU clock that the prescaler divides. The CM06 bit in the CM0 register determines whether a watchdog timer interrupt request or reset is generated if the watchdog timer underflows. The CM06 bit can only be set to "1" (reset). Once the CM06 bit is set to "1", it cannot be changed to "0" (watchdog timer interrupt) by program. The CM06 bit is set to "0" only after reset. When the main clock, on-chip oscillator clock, or PLL clock runs as the CPU clock, the WDC7 bit in the WDC register determine whether the prescaler divides the clock by 16 or by 128. When the sub clock runs as the CPU clock, the prescaler divides the clock by 2 regardless of the WDC7 bit setting. Watchdog timer cycle is calculated as follows. Marginal errors, due to the prescaler, may occur in watchdog timer cycle.

When the main clock, on-chip oscillator clock, or PLL clock is selected as the CPU clock,

$$\text{Watchdog timer cycle} = \frac{\text{Divide-by-16 or -128 prescaler x counter value of watchdog timer (32768)}}{\text{CPU clock}}$$

When the sub clock is selected as the CPU clock,

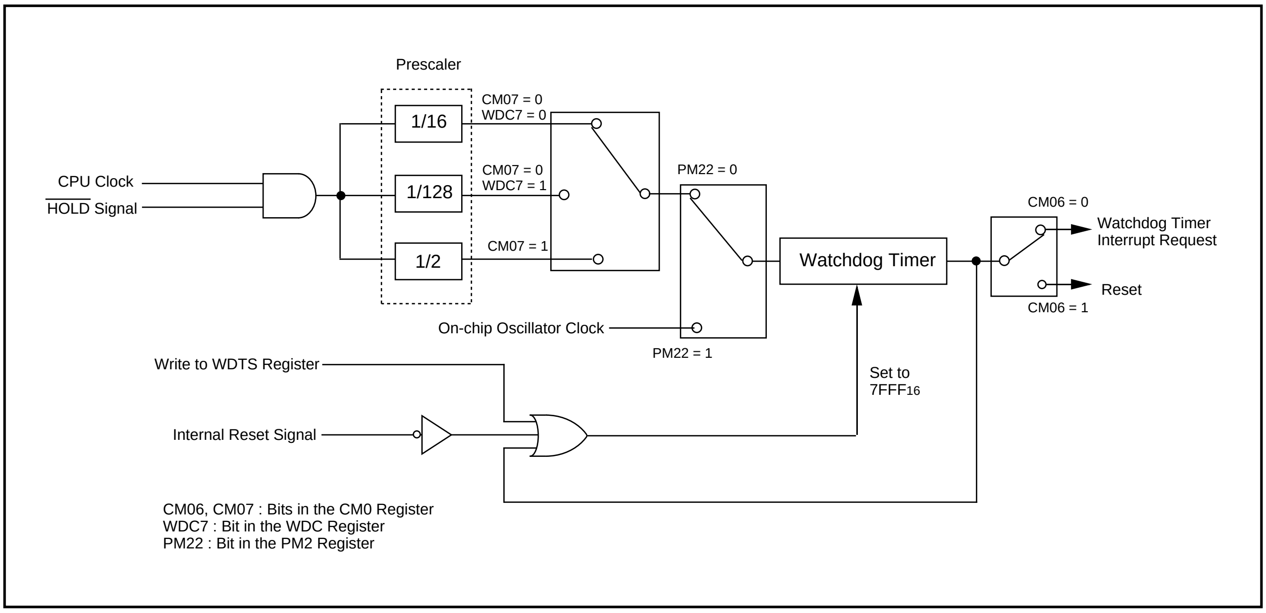
$$\text{Watchdog timer cycle} = \frac{\text{Divide-by-2 prescaler x counter value of watchdog timer (32768)}}{\text{CPU clock}}$$

For example, if the CPU clock frequency is 30MHz and the prescaler divides it by 16, the watchdog timer cycle is approximately 17.5 ms.

The watchdog timer is reset when the WDTS register is set and when a watchdog timer interrupt request is generated. The prescaler is reset only when the microcomputer is reset. Both watchdog timer and prescaler stop after reset. They begin counting when the WDTS register is set.

The watchdog timer and prescaler stop in stop mode, wait mode and hold state. They resume counting from the value held when the mode or state is exited.

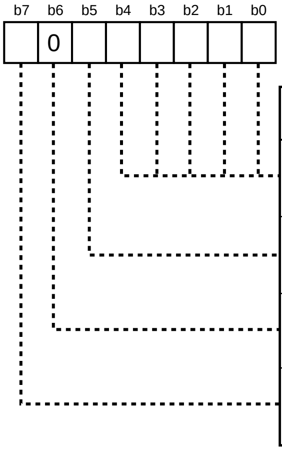
Figure 12.1 shows a block diagram of the watchdog timer. Figure 12.2 shows registers associated with the watchdog timer.



CM06, CM07 : Bits in the CM0 Register
WDC7 : Bit in the WDC Register
PM22 : Bit in the PM2 Register

Figure 12.1 Watchdog Timer Block Diagram

Watchdog Timer Control Register

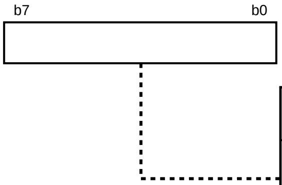


Symbol	Address	After Reset
WDC	000F$_{16}$	000X XXXX$_2$

Bit Symbol	Bit Name	Function	RW
— (b4 - b0)	High-Order Bit of the Watchdog Timer		RO
WDC5	Cold Start-up/ Warm Start-up Determine Flag[1,2, 3]	0 : Cold start-up 1 : Warm start-up	RW
— (b6)	Reserved Bit	Set to "0"	RW
WDC7	Prescaler Select Bit	0 : Divide-by-16 1 : Divide-by-128	RW

NOTES:
1. The WDC5 bit remains set to "1", regardless of setting to "1" or "0".
2. The WDC5 bit is set to "0" when power is turned on and can be set to "1" by program only.
3. The WDC5 bit maintains a value set before reset, even after reset has been performed.

Watchdog Timer Start Register[1]



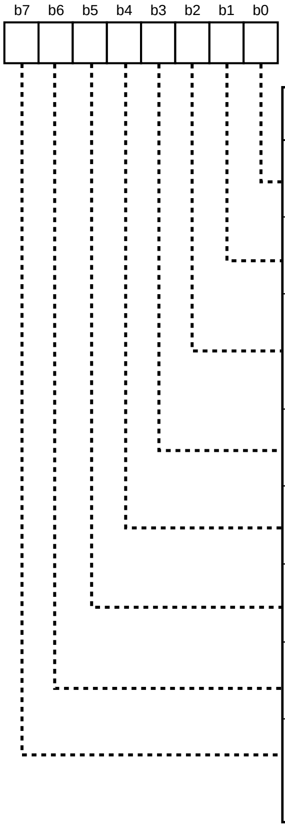
Symbol	Address	After Reset
WDTS	000E$_{16}$	Indeterminate

Function	RW
The watchdog timer is reset to start counting by a write instruction to the WDTS register. Default value of the watchdog timer is always set to "7FFF$_{16}$" regardless of the value written.	WO

NOTES:
1. Write the WDTS register after the watchdog timer interrupt is generated.

Figure 12.2 WDC Register and WDTS Register

System Clock Control Register 0(1)



Symbol	Address	After Reset
CM0	0006_{16}	$0000\ 1000_2$

Bit Symbol	Bit Name	Function	RW
CM00	Clock Output Function Select Bit(2)	b1 b0 0 0 : I/O port $P5_3$ 0 1 : Outputs f_C 1 0 : Outputs f_8 1 1 : Outputs f_{32}	RW
CM01			RW
CM02	In Wait Mode, Peripheral Function Clock Stop Bit(9)	0 : Peripheral clock does not stop in wait mode 1 : Peripheral clock stops in wait mode(3)	RW
CM03	XCIN-XCOUT Drive Capacity Select Bit(11)	0 : Low 1 : High	RW
CM04	Port XC Switch Bit	0 : I/O port function 1 : XCIN-XCOUT oscillation function(4)	RW
CM05	Main Clock (XIN-XOUT) Stop Bit(5, 9)	0 : Main clock oscillates 1 : Main clock stops(6)	RW
CM06	Watchdog Timer Function Select Bit	0 : Watchdog timer interrupt 1 : Reset(7)	RW
CM07	CPU Clock Select Bit 0(8, 9, 10)	0: Clock selected by the CM21 bit divided by MCD register setting 1: Sub clock	RW

NOTES:

1. Rewrite the CM0 register after the PRC0 bit in the PRCR register is set to "1" (write enable).
2. When the PM07 bit in the PM0 register is set to "0" (BCLK output), set the CM01 and CM00 bits to "00_2". When the PM15 and PM14 bits in the PM1 register are set to "01_2" (ALE output to $P5_3$), set the CM01 and CM00 bits to "00_2". When the PM07 bit is set to "1" (function selected in the CM01 and CM00 bits) in microprocessor or memory expansion mode, and the CM01 and CM00 bits are set to "00_2", an "L" signal is output from port $P5_3$ (port $P5_3$ does not function as an I/O port).
3. f_{C32} does not stop running. When the CM02 bit is set to "1", the PLL clock cannot be used in wait mode.
4. When setting the CM04 bit is set to "1", set the PD8_7 and PD8_6 bits in the PD8 register to "00_2" (port $P8_7$ and $P8_6$ in input mode) and the PU25 bit in the PUR2 register to "0" (no pull-up).
5. When entering low-power consumption mode or on-chip oscillator low-power consumption mode, the CM05 bit stops running the main clock. The CM05 bit cannot detect whether the main clock stops or not. To stop running the main clock, set the CM05 bit to "1" after the CM07 bit is set to "1" with a stable sub clock oscillation or after the CM21 bit in the CM2 register is set to "1" (on-chip oscillator clock). When the CM05 bit is set to "1", the clock applied to XOUT becomes "H". The built-in feedback resistor remains ON. XIN is pulled up to XOUT ("H" level) via the feedback resistor.
6. When the CM05 bit is set to "1", the MCD4 to MCD0 bits in the MCD register are set to "01000_2" (divide-by-8 mode). In on-chip oscillation mode, the MCD4 to MCD0 bits are not set to "01000_2" even if the CM05 bit terminates XIN-XOUT.
7. Once the CM06 bit is set to "1", it cannot be set to "0" by program.
8. After the CM04 bit is set to "1" with a stable sub clock oscillation, set the CM07 bit to "1" from "0". After the CM05 bit is set to "0" with a stable main clock oscillation, set the CM07 bit to "0" from "1". Do not set the CM07 bit and CM04 or CM05 bit simultaneously.
9. When the PM21 bit in the PM2 register is set to "1" (clock change disable), the CM02, CM05 and CM07 bits do not change even when written.
10. After the CM07 bit is set to "0", set the PM21 bit to "1".
11. When stop mode is entered, the CM03 bit is set to "1".

Figure 12.3 CM0 Register

12.1 Count Source Protection Mode

In count source protection mode, the on-chip oscillator clock is used as a count source for the watchdog timer. The count source protection mode allows the on-chip oscillator clock to run continuously, maintaining watchdog timer operation even if the program error occurs and the CPU clock stops running.

Follow the procedures below when using this mode.

(1) Set the PRC0 bit in the PRCR register to "1" (write to CM0 register enabled)

(2) Set the PRC1 bit in the PRCR register to "1" (write to PM2 register enabled)

(3) Set the CM06 bit in the CM0 register to "1" (reset when the watchdog timer overflows)

(4) Set the PM22 bit in the PM2 register to "1" (the on-chip oscillator clock as a count source of the watchdog timer)

(5) Set the PRC0 bit to "0" (write to CM0 register disabled)

(6) Set the PRC1 bit to "0" (write to PM2 register disabled)

(7) Write to the WDTS register (the watchdog timer starts counting)

The followings will occur when the PM22 bit is set to "1".

- The on-chip oscillator starts oscillating and the on-chip oscillator clock becomes a count source for the watchdog timer.

$$\text{Watchdog timer cycle} = \frac{\text{Counter value of watchdog timer (32768)}}{\text{On-chip oscillator clock}}$$

- Write to the CM10 bit in the CM1 register is disabled. (The bit setting remains unchanged even if set it to "1". The microcomputer does not enter stop mode.)
- In wait mode or hold state, the watchdog timer continues running. However, the watchdog timer interrupt cannot be used to exit wait mode.

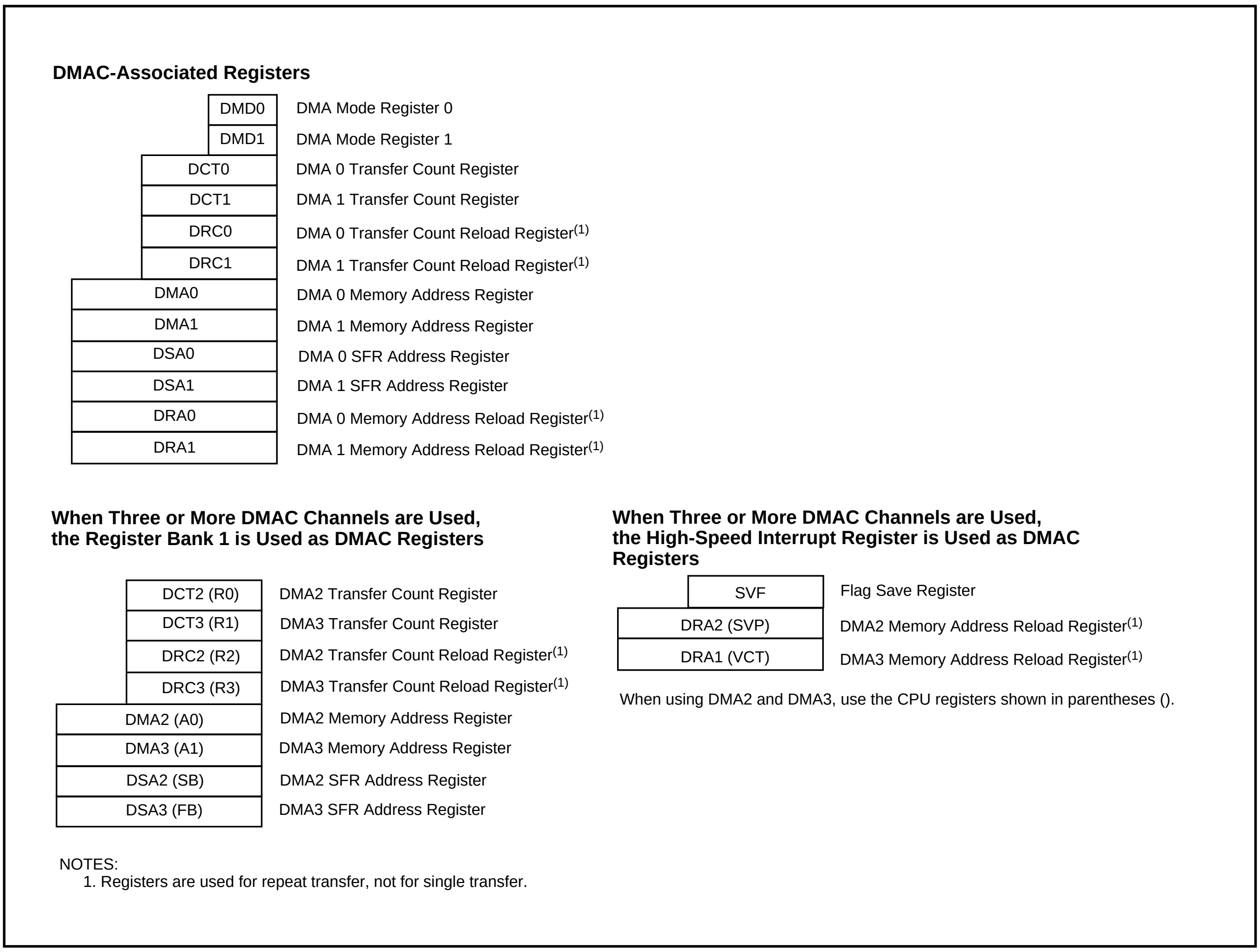
13. DMAC

This microcomputer contains four DMAC (direct memory access controller) channels that allow data to be sent to memory without using the CPU. DMAC transmits a 8- or 16-bit data from a source address to a destination address whenever a transmit request occurs. DMA0 and DMA1 must be prioritized if using DMAC. DMA2 and DMA3 share registers required for high-speed interrupts. High-speed interrupts cannot be used when using three or more DMAC channels.

The CPU and DMAC use the same data bus, but DMAC has a higher bus access privilege than the CPU. The cycle-steal method employed on DMAC enables high-speed operation between a transfer request and the complete transmission of 16-bit (word) or 8-bit (byte) data. Figure 13.1 shows a mapping of registers to be used for DMAC. Table 13.1 lists specifications of DMAC. Figures 13.2 to 13.5 show registers associated with DMAC.

Because the registers shown in Figure 13.1 are allocated in the CPU, use the LDC instruction to write to the registers. To set the DCT2, DCT3, DRC2, DRC3, DMA2 and DMA3 registers, set the B flag to "1" (register bank 1) and set the R0 to R3, A0, A1 registers with the MOV instruction.

To set the DSA2 and DSA3 registers, set the B flag to "1" and set the SB and FB registers with the LDC instruction. To set the DRA2 and DRA3 registers, set the SVP and VCT registers with the LDC instruction.



DMAC-Associated Registers

DMD0	DMA Mode Register 0
DMD1	DMA Mode Register 1
DCT0	DMA 0 Transfer Count Register
DCT1	DMA 1 Transfer Count Register
DRC0	DMA 0 Transfer Count Reload Register[1]
DRC1	DMA 1 Transfer Count Reload Register[1]
DMA0	DMA 0 Memory Address Register
DMA1	DMA 1 Memory Address Register
DSA0	DMA 0 SFR Address Register
DSA1	DMA 1 SFR Address Register
DRA0	DMA 0 Memory Address Reload Register[1]
DRA1	DMA 1 Memory Address Reload Register[1]

When Three or More DMAC Channels are Used, the Register Bank 1 is Used as DMAC Registers

DCT2 (R0)	DMA2 Transfer Count Register
DCT3 (R1)	DMA3 Transfer Count Register
DRC2 (R2)	DMA2 Transfer Count Reload Register[1]
DRC3 (R3)	DMA3 Transfer Count Reload Register[1]
DMA2 (A0)	DMA2 Memory Address Register
DMA3 (A1)	DMA3 Memory Address Register
DSA2 (SB)	DMA2 SFR Address Register
DSA3 (FB)	DMA3 SFR Address Register

When Three or More DMAC Channels are Used, the High-Speed Interrupt Register is Used as DMAC Registers

SVF	Flag Save Register
DRA2 (SVP)	DMA2 Memory Address Reload Register[1]
DRA1 (VCT)	DMA3 Memory Address Reload Register[1]

When using DMA2 and DMA3, use the CPU registers shown in parentheses ().

NOTES:
1. Registers are used for repeat transfer, not for single transfer.

Figure 13.1 Register Mapping for DMAC

DMAC starts a data transfer by setting the DSR bit in the DMiSL register (i=0 to 3) or by using an interrupt request, generated by the functions determined by the DSEL 4 to DSEL0 bits in the DMiSL register, as a DMA request. Unlike interrupt requests, the I flag and interrupt control register do not affect DMA. Therefore, a DMA request can be acknowledged even if an interrupt is disabled and cannot be acknowledged. In addition, the IR bit in the interrupt control register does not change when a DMA request is acknowledged.

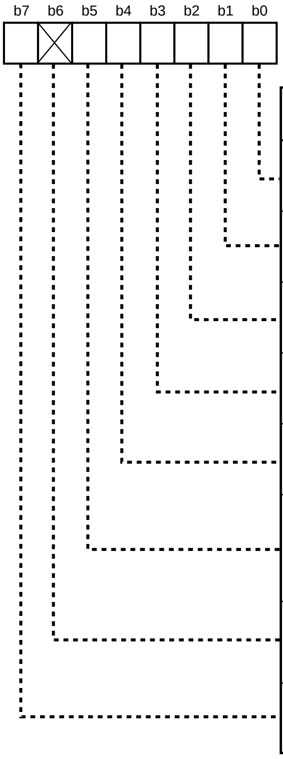
Table 13.1 DMAC Specifications

Item		Specification
Channels		4 channels (cycle-steal method)
Transfer Memory Space		• From a desired address in a 16-Mbyte space to a fixed address in a 16-Mbyte space • From a fixed address in a 16-Mbyte space to a desired address in a 16-Mbyte space
Maximum Bytes Transferred		128 Kbytes (when a 16-bit data is transferred) or 64 Kbytes (with an 8-bit data is transferred)
DMA Request Source[1]		Falling edge or both edges of signals applied to the $\overline{\text{INT0}}$ to $\overline{\text{INT3}}$ pins Timers A0 to A4 interrupt requests Timers B0 to B5 interrupt requests UART0 to UART4 transmit and receive interrupt requests A/D0 conversion interrupt request Intelligent I/O interrupt request CAN interrupt request Software trigger
Channel Priority		DMA0 > DMA1 > DMA2 > DMA3 (DMA0 has highest priority)
Transfer Unit		8 bits, 16 bits
Destination Address		Forward/fixed (forward and fixed directions cannot be specified when specifying source and destination addresses simultaneously)
Transfer Mode	Single Transfer	Transfer is completed when the DCTi register (i = 0 to 3) is set to "0000_{16}"
	Repeat Transfer	When the DCTi register is set to "0000_{16}", the value of the DRCi register is reloaded into the DCTi register and the DMA transfer is continued
DMA Interrupt Request Generation Timing		When the DCTi register changes "0001_{16}" to "0000_{16}"
DMA Startup	Single Transfer	DMA starts when a DMA request is generated after the DCTi register is set to "0001_{16}" or more and the MDi1 and MD0 bits in the DMDj register (j = 0,1) are set to "01_2" (single transfer)
	Repeat Transfer	DMA starts when a DMA request is generated after the DCTi register is set to "0001_{16}" or more and the MDi1 and MDi0 bits are set to "11_2" (repeat transfer)
DMA Stop	Single Transfer	DMA stops when the MDi1 and MDi0 bits are set to "00_2" (DMA disabled) and the DCTi register is set to "0000_{16}" (0 DMA transfer) by DMA transfer or write
	Repeat Transfer	DMA stops when the MDi1 and MDi0 bits are set to "00_2" and the DCTi register is set to "0000_{16}" and the DRCi register set to "0000_{16}"
Reload Timing to the DCTi or DMAi Register		When the DCTi register is set to "0000_{16}" from "0001_{16}" in repeat transfer mode
DMA Transfer Cycles		Minimum 3 cycles between SFR and internal RAM

NOTES:

1. The IR bit in the interrupt control register does not change when a DMA request is acknowledged.

DMAi Request Source Select Register (i=0 to 3)



Symbol	Address	After Reset
DM0SL to DM3SL	0378$_{16}$, 0379$_{16}$, 037A$_{16}$, 037B$_{16}$	0X00 0000$_2$

Bit Symbol	Bit Name	Function	RW
DSEL0	DMA Request Source Select Bit[(1)]	See Table 13.2 for the DMiSL register (i = 0 to 3) function	RW
DSEL1			RW
DSEL2			RW
DSEL3			RW
DSEL4			RW
DSR	Software DMA Request Bit[(2)]	When a software trigger is selected, a DMA request is generated by setting this bit to "1" (When read, its content is always "0")	RW
—— (b6)	Reserved Bit	When read, its content is indeterminate	RO
DRQ	DMA Request Bit[(2, 3)]	0 : Not requested 1 : Requested	RW

NOTES:

1. Change the DSEL4 to DSEL0 bit settings while the MDi1 and MDi0 bits in the DMD0 and DMD1 registers are set to "00$_2$" (DMA disabled). Also, set the DRQ bit to "1" simultaneously when the DSEL4 to DSEL0 bit settings are changed.
   e.g., MOV.B #083h, DMiSL ; Set timer A0
2. When the DSR bit is set to "1", set the DRQ bit to "1" simultaneously.
   e.g., OR.B #0A0h, DMiSL
3. Do not set the DRQ bit to "0".

Figure 13.2 DM0SL to DM3SL Registers

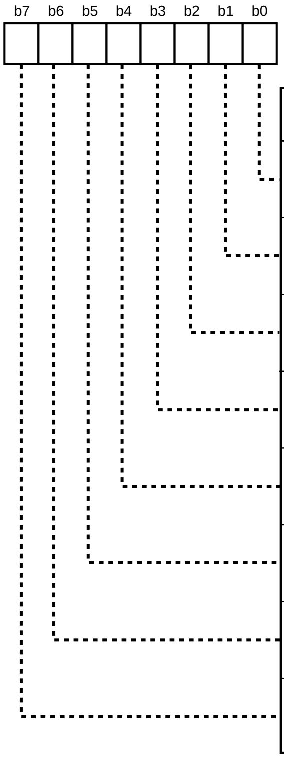
Table 13.2 DMiSL Register (i = 0 to 3) Function

Setting Value	DMA Request Source			
b4 b3 b2 b1 b0	DMA0	DMA1	DMA2	DMA3
0 0 0 0 0	Software trigger			
0 0 0 0 1	Falling Edge of INT0	Falling Edge of INT1	Falling Edge of INT2	Falling Edge of INT3(1) (Note 2)
0 0 0 1 0	Both Edges of INT0	Both Edges of INT1	Both Edges of INT2	Both Edges of INT3(1) (Note 2)
0 0 0 1 1	Timer A0 Interrupt Request			
0 0 1 0 0	Timer A1 Interrupt Request			
0 0 1 0 1	Timer A2 Interrupt Request			
0 0 1 1 0	Timer A3 Interrupt Request			
0 0 1 1 1	Timer A4 Interrupt Request			
0 1 0 0 0	Timer B0 Interrupt Request			
0 1 0 0 1	Timer B1 Interrupt Request			
0 1 0 1 0	Timer B2 Interrupt Request			
0 1 0 1 1	Timer B3 Interrupt Request			
0 1 1 0 0	Timer B4 Interrupt Request			
0 1 1 0 1	Timer B5 Interrupt Request			
0 1 1 1 0	UART0 Transmit Interrupt Request			
0 1 1 1 1	UART0 Receive or ACK Interrupt Request(3)			
1 0 0 0 0	UART1 Transmit Interrupt Request			
1 0 0 0 1	UART1 Receive or ACK Interrupt Request(3)			
1 0 0 1 0	UART2 Transmit Interrupt Request			
1 0 0 1 1	UART2 Receive or ACK Interrupt Request(3)			
1 0 1 0 0	UART3 Transmit Interrupt Request			
1 0 1 0 1	UART3 Receive or ACK Interrupt Request(3)			
1 0 1 1 0	UART4 Transmit Interrupt Request			
1 0 1 1 1	UART4 Receive or ACK Interrupt Request(3)			
1 1 0 0 0	A/D0 Interrupt Request			
1 1 0 0 1	Intelligent I/O Interrupt 0 Request(6)	—	Intelligent I/O Interrupt 2 Request	Intelligent I/O Interrupt 9 Request(4)
1 1 0 1 0	Intelligent I/O Interrupt 1 Request(7)	Intelligent I/O Interrupt 8 Request	Intelligent I/O Interrupt 3 Request	Intelligent I/O Interrupt 10 Request(5)
1 1 0 1 1	Intelligent I/O Interrupt 2 Request	Intelligent I/O Interrupt 9 Request(4)	Intelligent I/O Interrupt 4 Request	CAN Interrupt 2 Request
1 1 1 0 0	Intelligent I/O Interrupt 3 Request	Intelligent I/O Interrupt 10 Request(5)	CAN Interrupt 5 Request	Intelligent I/O Interrupt 0 Request(6)
1 1 1 0 1	Intelligent I/O Interrupt 4 Request	CAN Interrupt 2 Request	—	Intelligent I/O Interrupt 1 Request(7)
1 1 1 1 0	CAN Interrupt 5 Request	Intelligent I/O Interrupt 0 Request(6)	—	Intelligent I/O Interrupt 2 Request
1 1 1 1 1	—	Intelligent I/O Interrupt 1 Request(7)	Intelligent I/O Interrupt 8 Request	Intelligent I/O Interrupt 3 Request

NOTES:

1. If the $\overline{\text{INT3}}$ pin is used for data bus in memory expansion mode or microprocessor mode, a DMA3 interrupt request cannot be generated by a signal applied to the $\overline{\text{INT3}}$ pin.
2. The falling edge and both edges of signals applied to the $\overline{\text{INTj}}$ pin (j=0 to 3) cause a DMA request generation. The $\overline{\text{INT}}$ interrupt (the POL bit in the INTjIC register, the LVS bit, the IFSR register) is not affected and vice versa.
3. Use the UkSMR register and UkSMR2 register (k=0 to 4) to switch between the UARTk receive and ACK interrupt as a DMA request source.
   To use the ACK interrupt for a DMA reqest, set the IICM bit in the UkSMR register to "1" and the IICM2 bit in the UkSMR2 register to "0".
4. The same setting is used to generate an intelligent I/O interrupt 9 request and a CAN interrupt 0 request.
5. The same setting is used to generate an intelligent I/O interrupt 10 request and a CAN interrupt 1 request.
6. The same setting is used to generate an intelligent I/O interrupt 0 request and a CAN interrupt 3 request.
7. The same setting is used to generate an intelligent I/O interrupt 1 request and a CAN interrupt 4 request.

DMA Mode Register 0[1]



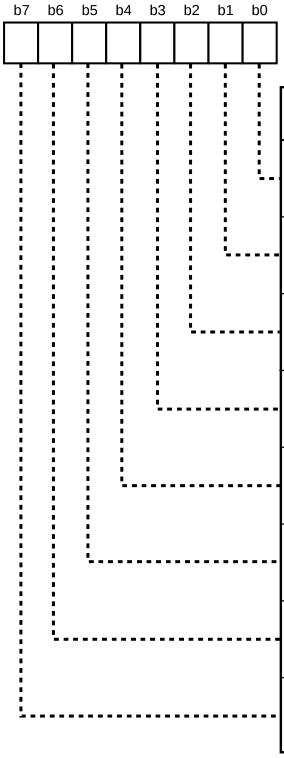
Symbol	Address	After Reset
DMD0	(CPU Internal Register)	00_{16}

Bit Symbol	Bit Name	Function	RW
MD00	Channel 0 Transfer Mode Select Bit	b1 b0 0 0 : DMA disabled 0 1 : Single transfer 1 0 : Do not set to this value 1 1 : Repeat transfer	RW
MD01			RW
BW0	Channel 0 Transfer Unit Select Bit	0 : 8 bits 1 : 16 bits	RW
RW0	Channel 0 Transfer Direction Select Bit	0 : Fixed address to memory (forward direction) 1 : Memory (forward direction) to fixed address	RW
MD10	Channel 1 Transfer Mode Select Bit	b5 b4 0 0 : DMA disabled 0 1 : Single transfer 1 0 : Do not set to this value 1 1 : Repeat transfer	RW
MD11			RW
BW1	Channel 1 Transfer Unit Select Bit	0 : 8 bits 1 : 16 bits	RW
RW1	Channel 1 Transfer Direction Select Bit	0 : Fixed address to memory (forward direction) 1 : Memory (forward direction) to fixed address	RW

NOTES:

1. Use the LDC instruction to set the DMD0 register.

DMA Mode Register 1[1]



Symbol	Address	After Reset
DMD1	(CPU internal register)	00_{16}

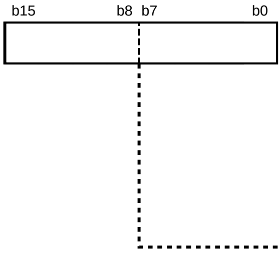
Bit Symbol	Bit Name	Function	RW
MD20	Channel 2 Transfer Mode Select Bit	b1 b0 0 0 : DMA disabled 0 1 : Single transfer 1 0 : Do not set to this value 1 1 : Repeat transfer	RW
MD21			RW
BW2	Channel 2 Transfer Unit Select Bit	0 : 8 bits 1 : 16 bits	RW
RW2	Channel 2 Transfer Direction Select Bit	0 : Fixed address to memory (forward direction) 1 : Memory (forward direction) to fixed address	RW
MD30	Channel 3 Transfer Mode Select Bit	b5 b4 0 0 : DMA disabled 0 1 : Single transfer 1 0 : Do not set to this value 1 1 : Repeat transfer	RW
MD31			RW
BW3	Channel 3 Transfer Unit Select Bit	0 : 8 bits 1 : 16 bits	RW
RW3	Channel 3 Transfer Direction Select Bit	0 : Fixed address to memory (forward direction) 1 : Memory (forward direction) to fixed address	RW

NOTES:

1. Use the LDC instruction to set the DMD1 register.

Figure 13.3 DMD0 and DMD1 Registers

DMAi Transfer Count Register (i=0 to 3)



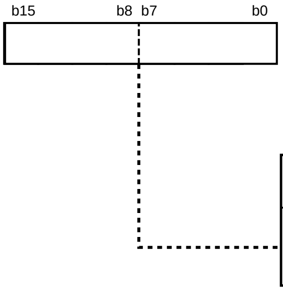
Symbol	Address	After Reset
DCT0(2)	(CPU Internal Register)	$XXXX_{16}$
DCT1(2)	(CPU Internal Register)	$XXXX_{16}$
DCT2(bank1;R0)(3)	(CPU Internal Register)	0000_{16}
DCT3(bank1;R1)(4)	(CPU Internal Register)	0000_{16}

Function	Setting Range	RW
Set the number of transfers	0000_{16} to $FFFF_{16}$(1)	RW

NOTES:

1. When the DCTi register is set to "0000_{16}", no data transfer occurs regardless of a DMA request.
2. Use the LDC instruction to set the DCT0 and DCT1 registers.
3. To set the DCT2 register, set the B flag in the FLG register to "1" (register bank 1) and set the R0 register. Use the MOV instruction to set the R0 register.
4. To set the DCT3 register, set the B flag to "1" and set R1 register. Use the MOV instruction to set the R1 register.

DMAi Transfer Count Reload Register (i=0 to 3)



Symbol	Address	After Reset
DRC0(1)	(CPU Internal Register)	$XXXX_{16}$
DRC1(1)	(CPU Internal Register)	$XXXX_{16}$
DRC2(bank1;R2)(2)	(CPU Internal Register)	0000_{16}
DRC3(bank1;R3)(3)	(CPU Internal Register)	0000_{16}

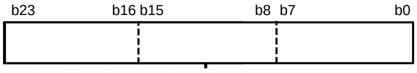
Function	Setting Range	RW
Set the number of transfers	0000_{16} to $FFFF_{16}$	RW

NOTES:

1. Use the LDC instruction to set the DRC0 and DRC1 registers.
2. To set the DRC2 register, set the B flag in the FLG register to "1" (register bank 1) and set the R2 register. Use the MOV instruction to set the R2 register.
3. To set the DRC3 register, set the B flag to "1" and set R3 register. Use the MOV instruction to set the R3 register.

Figure 13.4 DCT0 to DCT3 Registers and DRC0 to DRC3 Registers

DMAi Memory Address Register (i=0 to 3)



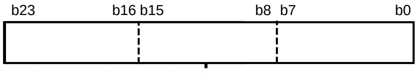
Symbol	Address	After Reset
DMA0[2]	(CPU Internal Register)	XXXXXX$_{16}$
DMA1[2]	(CPU Internal Register)	XXXXXX$_{16}$
DMA2(bank1;A0)[3]	(CPU Internal Register)	000000$_{16}$
DMA3(bank1;A1)[4]	(CPU Internal Register)	000000$_{16}$

Function	Setting Range	RW
Set a source memory address or destination memory address[1]	000000$_{16}$ to FFFFFF$_{16}$ (16-Mbyte space)	RW

NOTES:

1. When the RWk bit (k=0 to 3) in the DMDj register (j=0, 1)is set to "0" (fixed address to memory), a destination address is selected. When the RWk bit is set to "1" (memory to fixed address), a source address is selected.
2. Use the LDC instruction to set the DMA0 and DMA1 registers.
3. To set the DMA2 register, set the B flag in the FLG register to "1" (register bank 1) and set the A0 register. Use the MOV instruction to set the A0 register.
4. To set the DMA3 register, set the B flag to "1" and set the A1 register. Use the MOV instruction to set the A1 register.

DMAi SFR Address Register (i=0 to 3)



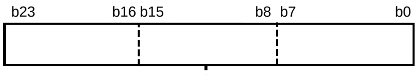
Symbol	Address	After Reset
DSA0[2]	(CPU Internal Register)	XXXXXX$_{16}$
DSA1[2]	(CPU Internal Register)	XXXXXX$_{16}$
DSA2(bank1;SB)[3]	(CPU Internal Register)	000000$_{16}$
DSA3(bank1;FB)[4]	(CPU Internal Register)	000000$_{16}$

Function	Setting Range	RW
Set a source fixed address or destination fixed address[1]	000000$_{16}$ to FFFFFF$_{16}$ (16-Mbyte space)	RW

NOTES:

1. When the RWk bit (k=0 to 3) in the DMDj register (j=0, 1)is set to "0" (fixed address to memory), a source address is selected. When the RWk bit is set to "1" (memory to fixed address), a destination address is selected.
2. Use the LDC instruction to set the DSA0 and DSA1 registers.
3. To set the DSA2 register, set the B flag in the FLG register to "1" (register bank 1) and the set the SB register. Use the LDC instruction to set the SB register.
4. To set the DSA3 register, set the B flag to "1" and set the FB register. Use the LDC instruction to set the PB register.

DMAi Memory Address Reload Register[1] (i=0 to 3)



Symbol	Address	After Reset
DRA0	(CPU Internal Register)	XXXXXX$_{16}$
DRA1	(CPU Internal Register)	XXXXXX$_{16}$
DRA2(SVP)[2]	(CPU Internal Register)	XXXXXX$_{16}$
DRA3(VCT)[3]	(CPU Internal Register)	XXXXXX$_{16}$

Function	Setting Range	RW
Set a source memory address or destination memory address[1]	000000$_{16}$ to FFFFFF$_{16}$ (16-Mbyte space)	RW

NOTES:

1. Use the LDC instruction to set the DRA0 and DRA1 registers.
2. To set the DRA2 register, set the SVP register.
3. To set the DRA3 register, set the VCT register.

Figure 13.5 DMA0 to DMA3 Registers, DSA0 to DSA3 Registers and DRA0 to DRA3 Registers

13.1 Transfer Cycle

Transfer cycle contains a bus cycle to read data from a memory or the SFR area (source read) and a bus cycle to write data to a memory space or the SFR area (destination write). The number of read and write bus cycles depends on source and destination addresses. In memory expansion mode and microprocessor mode, the number of read and write bus cycles also depends on DS register setting. Software wait state insertion and the $\overline{\text{RDY}}$ signal make a bus cycle longer.

13.1.1 Effect of Source and Destination Addresses

When a 16-bit data is transferred with a 16-bit data bus and a source address starting with an odd address, source read cycle is incremented by one bus cycle, compared to a source address starting with an even address.

When a 16-bit data is transferred with a 16-bit data bus and a destination address starting with an odd address, a destination write cycle is incremented by one bus cycle, compared to a destination address starting with an even address.

13.1.2 Effect of the DS Register

In an external space in memory expansion or microprocessor mode, transfer cycle varies depending on the data bus used at the source and destination addresses. See **Figure 8.1** for details about the DS register.

- When an 8-bit data bus (the DSi bit in the DS register is set to "0" (i=0 to 3)), accessing both source address and destination address, is used to transfer a 16-bit data, 8-bit data is transferred twice. Therefore, two bus cycles are required to read the data and another two bus cycles to write the data.
- When an 8-bit data bus (the DSi bit in the DS register is set to "0" (i=0 to 3)), accessing source address, and a 16-bit data bus, accessing destination address, are used to transfer a 16-bit data, 8-bit data is read twice but is written once as 16-bit data. Therefore, two bus cycles are required for reading and one bus cycle is for writing.
- When a 16-bit data bus, accessing source address, and an 8-bit data bus, accessing destination address, are used to transfer a 16-bit data, 16-bit data is read once and 8-bit data is written twice. Therefore, one bus cycle is required for reading and two bus cycles is for writing.

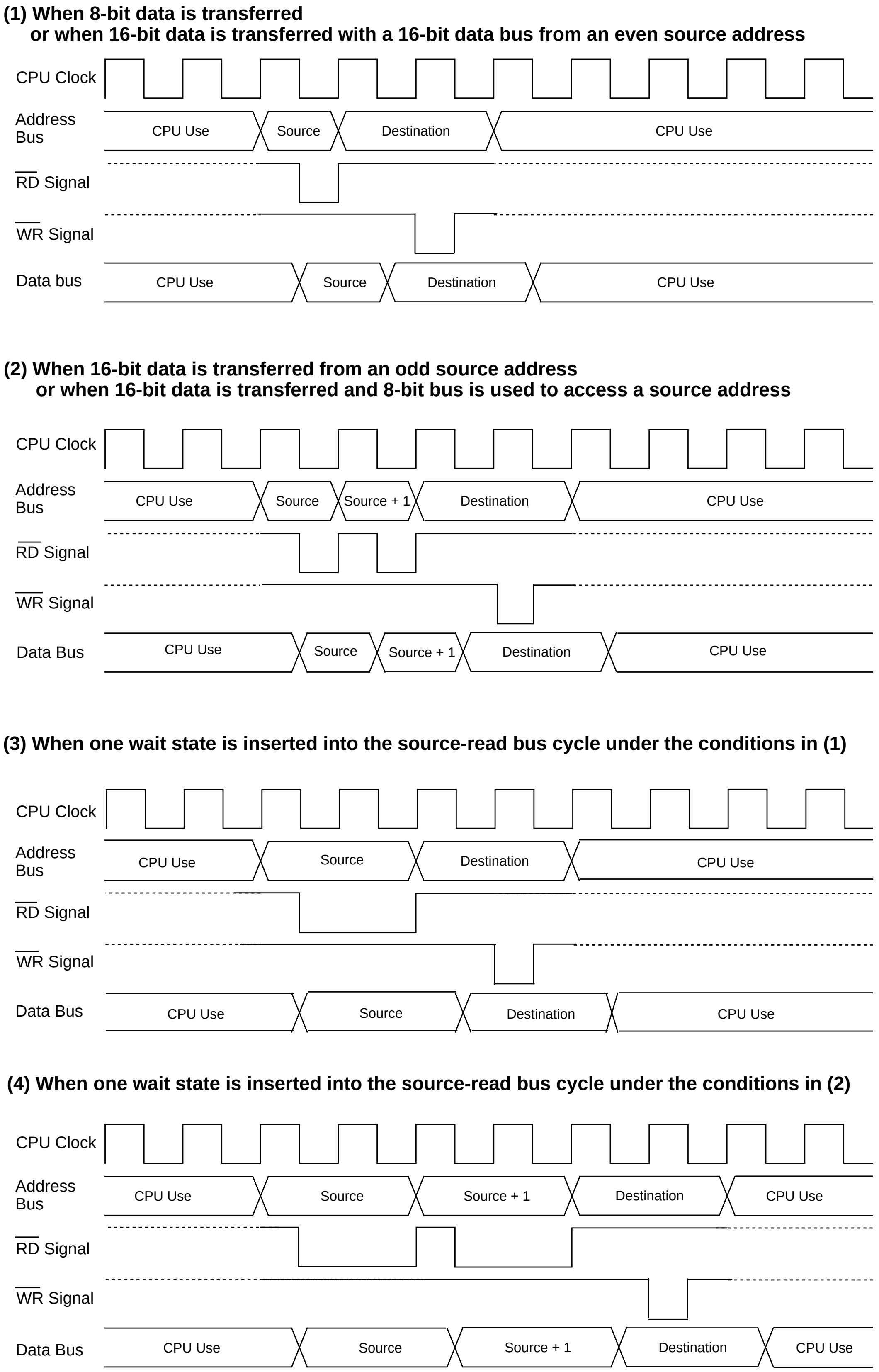
13.1.3 Effect of Software Wait State

When the SFR area or memory space with software wait states is accessed, the number of CPU clock cycles is incremented by software wait states.

Figure 13.6 shows an example of a transfer cycle for the source-read bus cycle. In Figure 13.6, the number of source-read bus cycles is illustrated under different conditions, provided that the destination address is an address of an external space with the destination-write cycle as two CPU clock cycles (=one bus cycle). In effect, the destination-write bus cycle is also affected by each condition and the transfer cycles change accordingly. To calculate a transfer cycle, apply respective conditions to both destination-write bus cycle and source-read bus cycle. As shown in example (2) of Figure 13.6, when an 8-bit data bus, accessing both source and destination addresses, is used to transfer a 16-bit data, two bus cycles each are required for the source-read bus cycle and destination-write bus cycle.

13.1.4 Effect of $\overline{\text{RDY}}$ Signal

In memory expansion or microprocessor mode, the $\overline{\text{RDY}}$ signal affects a bus cycle if a source address or destination address is allocated address in an external space. Refer to **8.2.6 $\overline{\text{RDY}}$ Signal** for details.



(1) When 8-bit data is transferred or when 16-bit data is transferred with a 16-bit data bus from an even source address

(2) When 16-bit data is transferred from an odd source address or when 16-bit data is transferred and 8-bit bus is used to access a source address

(3) When one wait state is inserted into the source-read bus cycle under the conditions in (1)

(4) When one wait state is inserted into the source-read bus cycle under the conditions in (2)

NOTES:

1. The above applies when the destination-write bus cycle is 2 CPU clock cycles (=1 bus cycle). However, if the destination-write bus cycle is pleaced under these conditions, it will change to the same timing as the source-read cycle illustrated above.

Figure 13.6 Transfer Cycle Examples with the Source-Read Bus Cycle

13.2 DMAC Transfer Cycle

The number of DMAC transfer cycle can be calculated as follows.

Any combination of even or odd transfer read and write addresses are possible. Table 13.3 lists the number of DMAC transfer cycles. Table 13.4 lists coefficient j, k.

Transfer cycles per transfer = Number of read cycle x j + Number of write cycle x k

Table 13.3 DMAC Transfer Cycles

Transfer Unit	Bus Width	Access Address	Single-Chip Mode		Memory Expansion Mode Microprocessor Mode	
			Read Cycle	Write Cycle	Read Cycle	Write Cycle
8-bit transfers (BWi bit in the DMDp register = 0)	16-bit	Even	1	1	1	1
		Odd	1	1	1	1
	8-bit	Even	—	—	1	1
		Odd	—	—	1	1
16-bit transfers (BWi bit = 1)	16-bit	Even	1	1	1	1
		Odd	2	2	2	2
	8-bit	Even	—	—	2	2
		Odd	—	—	2	2

i= 0 to 3, p = 0, 1

Table 13.4 Coefficient j, k

Internal Space			External Space
Internal ROM or internal RAM	Internal ROM or internal RAM	SFR area	j and k BCLK cycles shown in Table 8.5. Add one cycle to j or k cycles when inserting a recovery cycle.

with no wait state
j=1
k=1
	with a wait state
	j=2
	k=2
		j=2
		k=2	

j, k=2 to 9

13.3 Channel Priority and DMA Transfer Timing

When multiple DMA requests are generated in the same sampling period, between the falling edge of the CPU clock and the next falling edge, the DRQ bit in the DMiSL register (i = 0 to 3) is set to "1" (requested) simultaneously. Channel priority in this case is : DMA0 > DMA1 > DMA2 > DMA3.

Figure 13.7 shows an example of the DMA transfer by external source.

In Figure 13.7, the DMA0 request having highest priority is received first to start a transfer when a DMA0 request and DMA1 request are generated simultaneously. After one DMA0 transfer is completed, the bus privilege is returned to the CPU. When the CPU has completed one bus access, the DMA1 transfer starts. After one DMA1 transfer is completed, the privilege is again returned to the CPU.

In addition, DMA requests cannot be counted up since each channel has one DRQ bit. Therefore, when DMA requests, as DMA1 in Figure 13.7, occur more than once before receiving bus privilege, the DRQ bit is set to "0" as soon as privilege is acquired. The bus privilege is returned to the CPU when one transfer is completed.

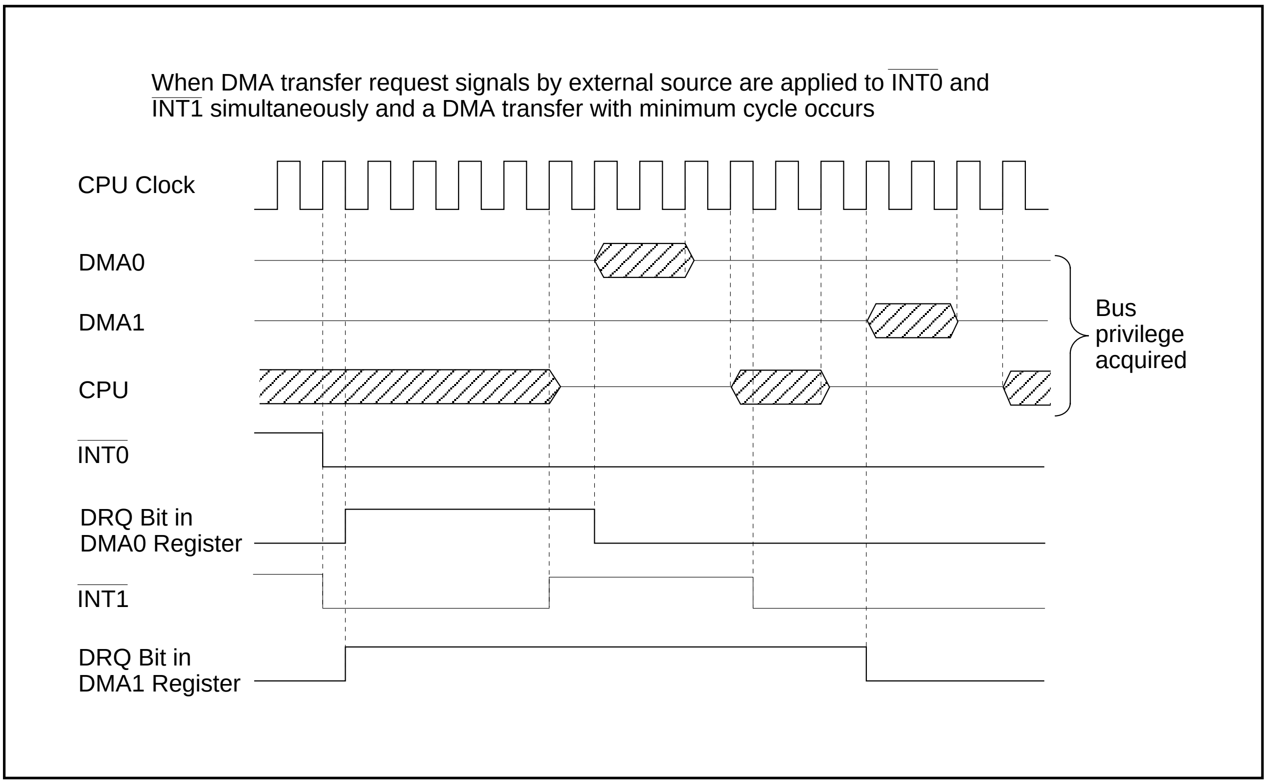

When DMA transfer request signals by external source are applied to $\overline{\text{INT0}}$ and $\overline{\text{INT1}}$ simultaneously and a DMA transfer with minimum cycle occurs
CPU Clock
DMA0
DMA1
CPU
$\overline{\text{INT0}}$
DRQ Bit in DMA0 Register
$\overline{\text{INT1}}$
DRQ Bit in DMA1 Register
Bus privilege acquired

Figure 13.7 DMA Transfer by External Source

14. DMAC II

DMAC II performs memory-to-memory transfer, immediate data transfer and calculation transfer, which transfers the sum of two data added by an interrupt request from any peripheral functions.
Table 14.1 lists specifications of DMAC II.

Table 14.1 DMAC II Specifications

Item	Specification
DMAC II Request Source	Interrupt requests generated by all peripheral functions when the ILVL2 to ILVL0 bits are set to "111_2"
Transfer Data	• Data in memory is transferred to memory (memory-to-memory transfer) • Immediate data is transferred to memory (immediate data transfer) • Data in memory (or immediate data) + data in memory are transferred to memory (calculation transfer)
Transfer Block	8 bits or 16 bits
Transfer Space	64-Kbyte space in addresses 00000_{16} to $0FFFF_{16}$(1, 2)
Transfer Direction	Fixed or forward address Selected separately for each source address and destination address
Transfer Mode	Single transfer, burst transfer
Chained Transfer Function	Parameters (transfer count, transfer address and other information) are switched when transfer counter reaches zero
End-of-Transfer Interrupt	Interrupt occurs when a transfer counter reaches zero
Multiple Transfer Function	Multiple data can be transferred by a generated request for one DMAC II transfer

NOTES:

1. When transferring a 16-bit data to destination address $0FFFF_{16}$, it is transferred to $0FFFF_{16}$ and 10000_{16}. The same transfer occurs when the source address is $0FFFF_{16}$.
2. The actual space where transfer can occurs is limited due to internal RAM capacity.

14.1 DMAC II Settings

DMAC II can be made available by setting up the following registers and tables.

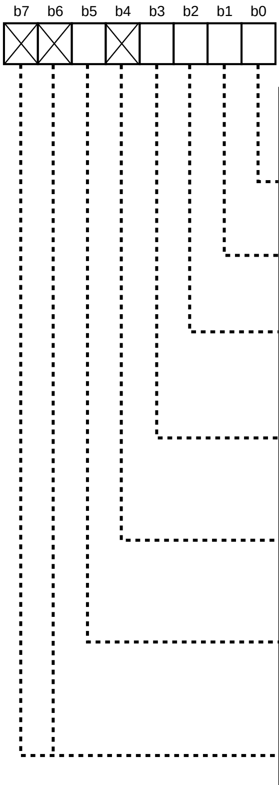
- RLVL register
- DMAC II Index
- Interrupt control register of the peripheral function causing a DMAC II request
- The relocatable vector table of the peripheral function causing a DMAC II request
- IRLT bit in the IIOiIE register (i = 0 to 5, 8 to 11) if using the intelligent I/O or CAN interrupt

Refer to **11. Interrupts** for details on the IIOiIE register.

14.1.1 RLVL Register

When the DMAII bit is set to "1" (DMAC II transfer) and the FSIT bit to "0" (normal interrupt), DMAC II is activated by an interrupt request from any peripheral function with the ILVL2 to ILVL0 bits in the interrupt control register set to "111_2" (level 7).
Figure 14.1 shows the RLVL register.

Exit Priority Register



Symbol	Address	After Reset
RLVL	009F$_{16}$	XXXX 0000$_2$

Bit Symbol	Bit Name	Function	RW
RLVL0	Stop/Wait Mode Exit Minimum Interrupt Priority Level Control Bit(1)	b2 b1 b0 0 0 0 : Level 0 0 0 1 : Level 1 0 1 0 : Level 2 0 1 1 : Level 3 1 0 0 : Level 4 1 0 1 : Level 5 1 1 0 : Level 6 1 1 1 : Level 7	RW
RLVL1			RW
RLVL2			RW
FSIT	High-Speed Interrupt Set Bit(2)	0: Interrupt priority level 7 is used for normal interrupt 1: Interrupt priority level 7 is used for high-speed interrupt	RW
____ (b4)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		–
DMAII	DMA II Select Bit(4)	0: Interrupt priority level 7 is used for interrupt 1: Interrupt priority level 7 is used for DMA II transfer(3)	RW
____ (b7 - b6)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		–

NOTES:

1. The microcomputer exits stop or wait mode when the requested interrupt priority level is higher than the level set in the RLVL2 to RLVL0 bits. Set the RLVL2 to RLVL0 bits to the same value as IPL in the FLG register.
2. When the FSIT bit is set to "1", an interrupt having the interrupt priority level 7 becomes the high-speed interrupt. In this case, set only one interrupt to the interrupt priority level 7 and the DMAII bit to "0".
3. Set the ILVL2 to ILVL0 bits in the interrupt control register after setting the DMAII bit to "1". Do not change the DMAII bit setting to "0" after setting the DMAII bit to "1". Set the FSIT bit to "0" when the DMAII bit to "1".
4. The DMAII bit becomes indeterminate after reset. To use the DMAII bit for an interrupt setting, set it to "0" before setting the interrupt control register.

Figure 14.1 RLVL Register

14.1.2 DMAC II Index

The DMAC II index is a data table which comprises 8 to 18 bytes (maximum 32 bytes when the multiple transfer function is selected). The DMAC II index stores parameters for transfer mode, transfer counter, source address (or immediate data), operation address as an address to be calculated, destination address, chained transfer address, and end-of-transfer interrupt address.

This DMAC II index must be located on the RAM area.

Figure 14.2 shows a configuration of the DMAC II index. Table 14.2 lists a configuration of the DMAC II index in transfer mode.

Memory-to-Memory Transfer, Immediate Transfer, Calculation Transfer

16 bits

DMAC II Index Starting Address (BASE)		
	Transfer Mode	(MOD)
BASE + 2	Transfer Counter	(COUNT)
BASE + 4	Transfer Source Address (or immediate data)	(SADR)
BASE + 6	Operation Address(1)	(OADR)
BASE + 8	Transfer Destination Address	(DADR)
BASE + 10	Chained Transfer Address(2)	(CADR0)
BASE + 12	Chained Transfer Address(2)	(CADR1)
BASE + 14	End-of-Transfer Interrupt Address(3)	(IADR0)
BASE + 16	End-of-Transfer Interrupt Address(3)	(IADR1)

Multiple Transfer

16 bits

BASE	Transfer Mode	(MOD)
BASE + 2	Transfer Counter	(COUNT)
BASE + 4	Transfer Source Address	(SADR1)
BASE + 6	Transfer Destination Address	(DADR1)
BASE + 8	Transfer Source Address	(SADR2)
BASE + 10	Transfer Destination Address	(DADR2)
	⁓	
BASE + 28	Transfer Source Address	(SADR7)
BASE + 30	Transfer Destination Address	(DADR7)

NOTES:
1. This data is not required when not using the calculation transfer function.
2. This data is not required when not using the chained transfer function.
3. This data is not required when not using the end-of-transfer interrupt.

The DMAC II index must be located on the RAM. Necessary data is set front-aligned. For example, if not using a calculation transfer function, set destination address to BASE+6. (See Table 14.2)
Starting address of the DMAC II index must be set in the interrupt vector for the peripheral function interrupt causing a DMAC II request.

Figure 14.2 DMAC II Index

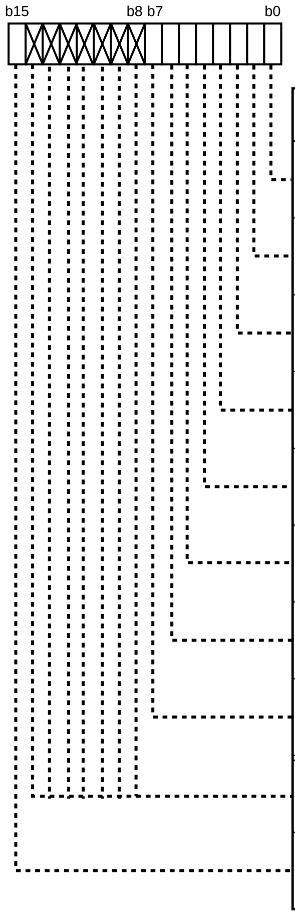
The followings are details of the DMAC II index. Set these parameters in the specified order listed in Table 14.2, according to DMAC II transfer mode.

- **Transfer mode (MOD)**
  Two-byte data is required to set transfer mode. Figure 14.3 shows a configuration for transfer mode.
- **Transfer counter (COUNT)**
  Two-byte data is required to set the number of transfer.
- **Transfer source address (SADR)**
  Two-byte data is required to set the source memory address or immediate data.
- **Operation address (OADR)**
  Two-byte data is required to set a memory address to be calculated. Set this data only when using the calculation transfer function.
- **Transfer destination address (DADR)**
  Two-byte data is required to set the destination memory address.
- **Chained transfer address (CADR)**
  Four-byte data is required to set the starting address of the DMAC II index for the next transfer. Set this data only when using the chained transfer function.
- **End-of-transfer interrupt address (IADR)**
  Four-byte data is required to set a jump address for end-of-transfer interrupt processing. Set this data only when using the end-of-transfer interrupt.

Table 14.2 DMAC II Index Configuration in Transfer Mode

Transfer Data	Memory-to-Memory Transfer /Immediate Data Transfer				Calculation Transfer				Multiple Transfer
Chained Transfer	Not Used	Used	Not Used	Used	Not Used	Used	Not Used	Used	Not Available
End-of-Transfer Interrupt	Not Used	Not Used	Used	Used	Not Used	Not Used	Used	Used	Not Available
DMAC II Index	MOD COUNT SADR DADR 8 bytes	MOD COUNT SADR DADR CADR0 CADR1 12 bytes	MOD COUNT SADR DADR IADR0 IADR1 12 bytes	MOD COUNT SADR DADR CADR0 CADR1 IADR0 IADR1 16 bytes	MOD COUNT SADR OADR DADR 10 bytes	MOD COUNT SADR OADR DADR CADR0 CADR1 14 bytes	MOD COUNT SADR OADR DADR IADR0 IADR1 14 bytes	MOD COUNT SADR OADR DADR CADR0 CADR1 IADR0 IADR1 18 bytes	MOD COUNT SADR1 DADR1 SADRi DADRi i=1 to 7 max. 32 bytes (when i=7)

Transfer Mode (MOD)[1]



Bit Symbol	Bit Name	Function (MULT=0)	Function (MULT=1)	RW
SIZE	Transfer Unit Select Bit	0: 8 bits 1: 16 bits		RW
IMM	Transfer Data Select Bit	0: Immediate data 1: Memory	Set to "1"	RW
UPDS	Transfer Source Direction Select Bit	0: Fixed address 1: Forward address		RW
UPDD	Transfer Destination Direction Select Bit	0: Fixed address 1: Forward address		RW
OPER/ CNT0[2]	Calculation Transfer Function Select Bit	0: Not used 1: Used	b6 b5 b4 0 0 0: Do not set to this value 0 0 1: Once 0 1 0: Twice : : 1 1 0: 6 times 1 1 1: 7 times	RW
BRST/ CNT1[2]	Burst Transfer Select Bit	0: Single transfer 1: Burst transfer		RW
INTE/ CNT2[2]	End-of-Transfer Interrupt Select Bit	0: Interrupt not used 1: Use interrupt		RW
CHAIN	Chained Transfer Select Bit	0: Chained transfer not used 1: Use chained transfer	Set to "0"	RW
——— (b14 - b8)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.			–
MULT	Multiple Transfer Select Bit	0: Multiple transfer not used	1: Use multiple transfer	RW

NOTES:
1. MOD must be located on the RAM.
2. When the MULT bit is set to "0" (no multiple transfer), bits 6 to 4 becomes the INTE, OPER and BRST bits. When the MULT bit is set to "1" (multiple transfer), bits 6 to 4 becomes the CNT2 to CNT0 bits.

Figure 14.3 MOD

14.1.3 Interrupt Control Register for the Peripheral Function

For the peripheral function interrupt activating DMAC II, set the ILVL2 to ILVL0 bits to "111_2" (level 7).

14.1.4 Relocatable Vector Table for the Peripheral Function

Set the starting address of the DMAC II index in the interrupt vector for the peripheral function interrupt activating DMAC II.

When using the chained transfer, the relocatable vector table must be located in the RAM.

14.1.5 IRLT Bit in the IIOiIE Register (i=0 to 5, 8 to 11)

When the intelligent I/O interrupt or CAN interrupt is used to activate DMAC II, set the IRLT bit in the IIOiIE register of the interrupt to "0".

14.2 DMAC II Performance

Function to activate DMAC II is selected by setting the DMA II bit to "1" (DMAC II transfer). DMAC II is activated by all peripheral function interrupts with the ILVL2 to ILVL0 bits set to "111_2" (level 7). These peripheral function interrupt request signals become DMAC II transfer request signals and the peripheral function interrupt cannot be used.

When an interrupt request is generated by setting the ILVL2 to ILVL0 bits to "111_2" (level 7), DMAC II is activated regardless of what state the I flag and IPL are in.

14.3 Transfer Data

DMAC II transfers 8-bit or 16-bit data.

- Memory-to-memory transfer : Data is transferred from a desired memory location in a 64-Kbyte space (Addresses 00000_{16} to $0FFFF_{16}$) to another desired memory location in the same space.
- Immediate data transfer : Immediate data is transferred to a desired memory location in a 64-Kbyte space.
- Calculation transfer : Two 8-bit or16-bit data are added together and the result is transferred to a desired memory location in a 64-Kbyte space.

When a 16-bit data is transferred to the destination address $0FFFF_{16}$, it is transferred to $0FFFF_{16}$ and 10000_{16}. The same transfer occurs when the source address is $0FFFF_{16}$. Actual transferable space varies depending on the internal RAM capacity.

14.3.1 Memory-to-memory Transfer

Data transfer between any two memory locations can be:

- a transfer from a fixed address to another fixed address
- a transfer from a fixed address to a relocatable address
- a transfer from a relocatable address to a fixed address
- a transfer from a relocatable address to another relocatable address

When a relocatable address is selected, the address is incremented, after a transfer, for the next transfer. In a 8-bit transfer, the transfer address is incremented by one. In a 16-bit transfer, the transfer address is incremented by two.

When a source or destination address exceeds address $0FFFF_{16}$ as a result of address incrementation, the source or destination address returns to address 00000_{16} and continues incrementation. Maintain source and destination address at address $0FFFF_{16}$ or below.

14.3.2 Immediate Data Transfer

DMAC II transfers immediate data to any memory location. A fixed or relocatable address can be selected as the destination address. Store the immediate data into SADR. To transfer an 8-bit immediate data, write the data in the low-order byte of SADR (high-order byte is ignored).

14.3.3 Calculation Transfer

After two memory data or an immediate data and memory data are added together, DMAC II transfers calculated result to any memory location. SADR must have one memory location address to be calculated or immediate data and OADR must have the other memory location address to be calculated. Fixed or relocatable address can be selected as source and destination addresses when using a memory + memory calculation transfer. If the transfer source address is relocatable, the operation address also becomes relocatable. Fixed or relocatable address can be selected as the transfer destination address when using an immediate data + memory calculation transfer.

14.4 Transfer Modes

Single and burst transfers are available. The BRST bit in MOD selects transfer method, either single transfer or burst transfer. COUNT determines how many transfers occur. No transfer occurs when COUNT is set to "0000_{16}".

14.4.1 Single Transfer

For every transfer request source, DMAC II transfers one transfer unit of 8-bit or 16-bit data once. When the source or destination address is relocatable, the address is incremented, after a transfer, for the next transfer.

COUNT is decremented every time a transfer occurs. When using the end-of-transfer interrupt, the interrupt is acknowledged when COUNT reaches "0".

14.4.2 Burst Transfer

For every transfer request source, DMAC II continuously transfers data the number of times determined by COUNT. COUNT is decremented every time a transfer occurs. The burst transfer ends when COUNT reaches "0". The end-of-transfer interrupt is acknowledged when the burst transfer ends if using the end-of-transfer interrupt. All interrupts are ignored while the burst transfer is in progress.

14.5 Multiple Transfer

The MULT bit in MOD selects the multiple transfer. When using the multiple transfer, select the memory-to-memory transfer. One transfer request source initiates multiple transfers. The CNT2 to CNT0 bits in MOD selects the number of transfers from "001_2" (once) to "111_2" (7 times). Do not set the CNT2 to CNT0 bits to "000_2".

The transfer source and destination addresses for each transfer must be allocated alternately in addresses following MOD and COUNT. When the multiple transfer is selected, the calculation transfer, burst transfer, end-of-transfer interrupt and chained transfer cannot be used.

14.6 Chained Transfer

The CHAIN bit in MOD selects the chained transfer.

The following process initiates the chained transfer.

(1) Transfer, caused by a transfer request source, occurs according to the content of the DMAC II index. The vectors of the request source indicates where the DMAC II index is allocated. For each request, the BRST bit selects either single or burst transfer.

(2) When COUNT reaches "0", the contents of CADR1 and CADR0 are written to the vector of the request source. When the INTE bit in MOD is set to "1", the end-of-transfer interrupt is generated simultaneously.

(3) When the next DMAC II transfer request is generated, transfer occurs according to the contents of the DMAC II index indicated by the peripheral function interrupt vector rewritten in (2).

Figure 14.4 shows the relocatable vector and DMACII index when the chained transfer is in progress. For the chained transfer, the relocatable vector table must be located in the RAM.

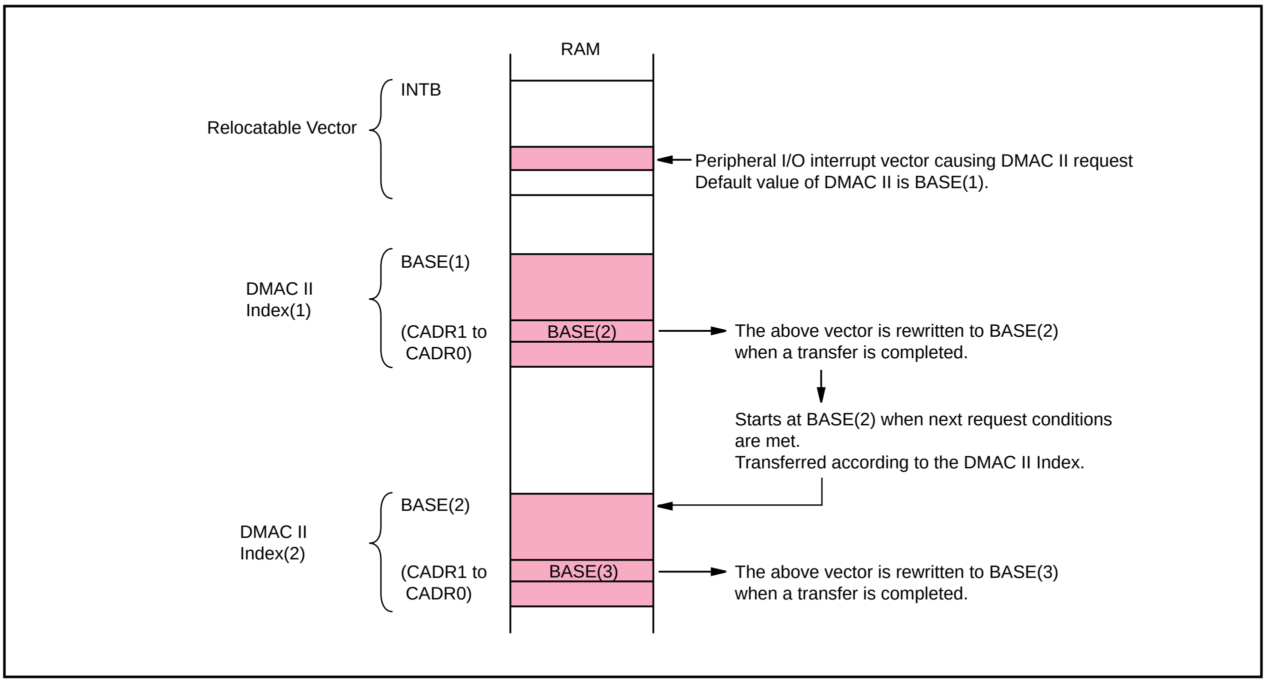

RAM
Relocatable Vector
INTB
Peripheral I/O interrupt vector causing DMAC II request
Default value of DMAC II is BASE(1).
DMAC II
Index(1)
BASE(1)
(CADR1 to
CADR0)
BASE(2)
The above vector is rewritten to BASE(2)
when a transfer is completed.
Starts at BASE(2) when next request conditions
are met.
Transferred according to the DMAC II Index.
DMAC II
Index(2)
BASE(2)
(CADR1 to
CADR0)
BASE(3)
The above vector is rewritten to BASE(3)
when a transfer is completed.

Figure 14.4 Relocatable Vector and DMAC II Index

14.7 End-of-Transfer Interrupt

The INTE bit in MOD selects the end-of-transfer interrupt. Set the starting address of the end-of-transfer interrupt routine in IADR1 and IADR0. The end-of-transfer interrupt is generated when COUNT reaches "0."

14.8 Execution Time

DMAC II execution cycle is calculated by the following equations:

Multiple transfers: t = 21+ (11 + b + c) x k cycles
Other than multiple transfers: t = 6 + (26 + a + b + c + d) x m + (4 + e) x n cycles

a: If IMM = 0 (source of transfer is immediate data), a = 0;
if IMM = 1 (source of transfer is memory), a = –1
b: If UPDS = 1 (source transfer address is a relocatable address), b = 0;
if UPDS = 0 (source transfer address is a fixed address), b = 1
c: If UPDD = 1 (destination transfer address is a relocatable address), c = 0;
if UPDD = 0 (destination transfer address is a fixed address), c = 1
d: If OPER = 0 (calculation function is not selected), d = 0;
if OPER = 1 (calculation function is selected) and UPDS = 0 (source of transfer is immediate data or fixed address memory), d = 7;
if OPER = 1 (calculation function is selected) and UPDS = 1 (source of transfer is relocatable address memory), d = 8
e: If CHAIN = 0 (chained transfer is not selected), e = 0; if CHAIN = 1 (chained transfer is selected), e = 4
m: BRST = 0 (single transfer), m = 1; BRST = 1 (burst transfer), m = the value set in transfer counter
n: If COUNT = 1, n = 0; if COUNT = 2 or more, n = 1
k: Number of transfers set in the CNT2 to CNT0 bits

The equations above are approximations. The number of cycles may vary depending on CPU state, bus wait state, and DMAC II index allocation.

The first instruction from the end-of-transfer interrupt routine is executed in the eighth cycle after the DMAC II transfer is completed.

If the end-of-transfer interrupt (transfer counter = 2) occurs with no chained transfer function after a memory-to-memory transfer occurs with a relocatable source address, fixed destination address, single transfer and double transfer:

a=-1 b=0 c=1 d=0 e=0 m=1

First DMAC II transfer	t=6+26x1+4x1=36 cycles
Second DMAC II transfer	t=6+26x1+4x0=32 cycles

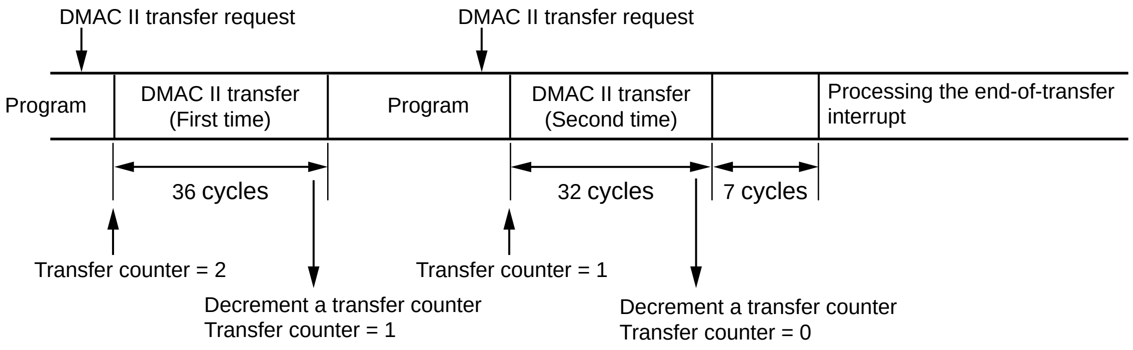
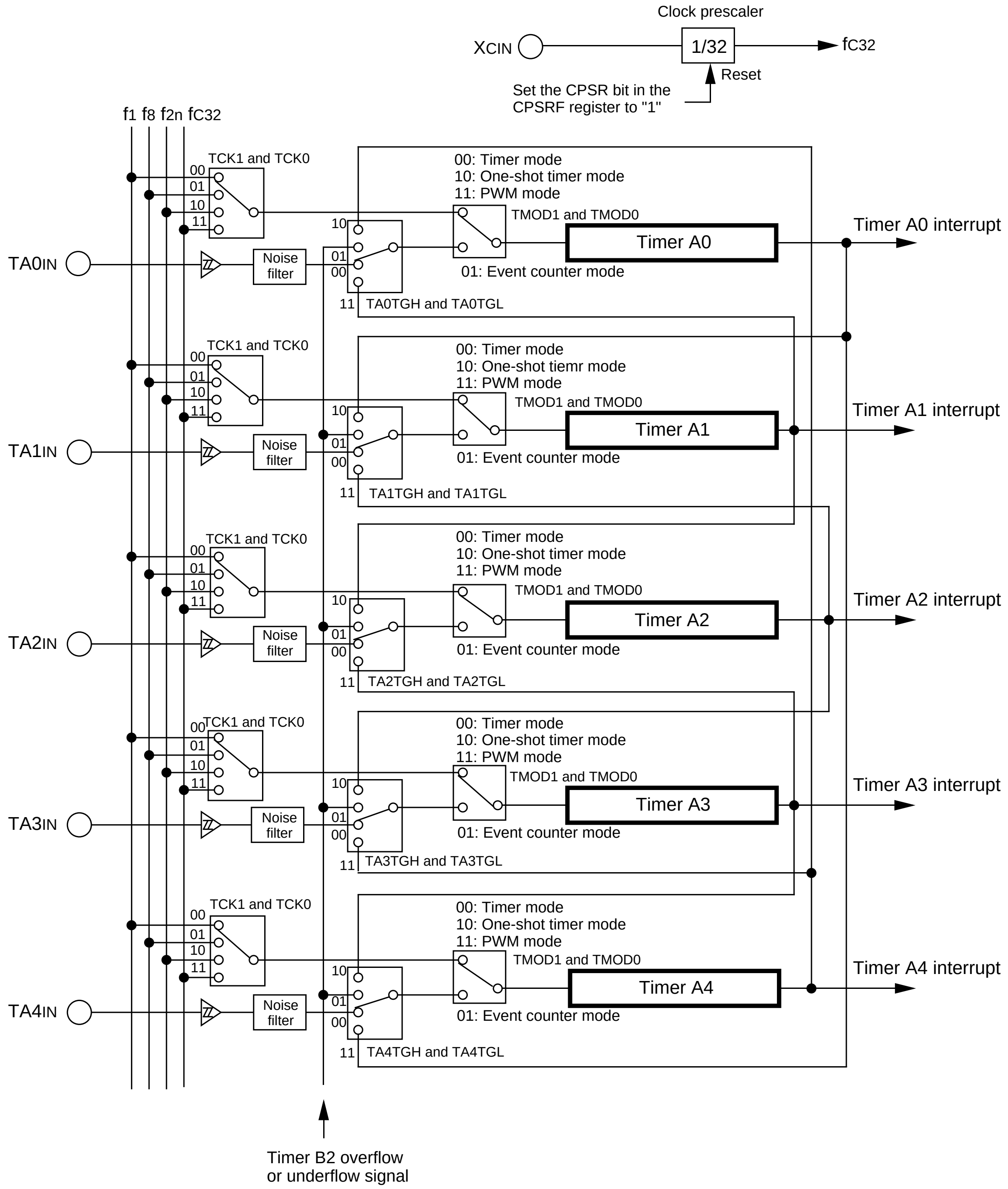


Figure 14.5 Transfer Cycle

When an interrupt request as a DMAC II transfer request source and another interrupt request with higher priority (e.g., $\overline{\text{NMI}}$ or watchdog timer) are generated simultaneously, the interrupt with higher priority takes precedence over the DMAC II transfer. The pending DMAC II transfer starts after the interrupt sequence has been completed.

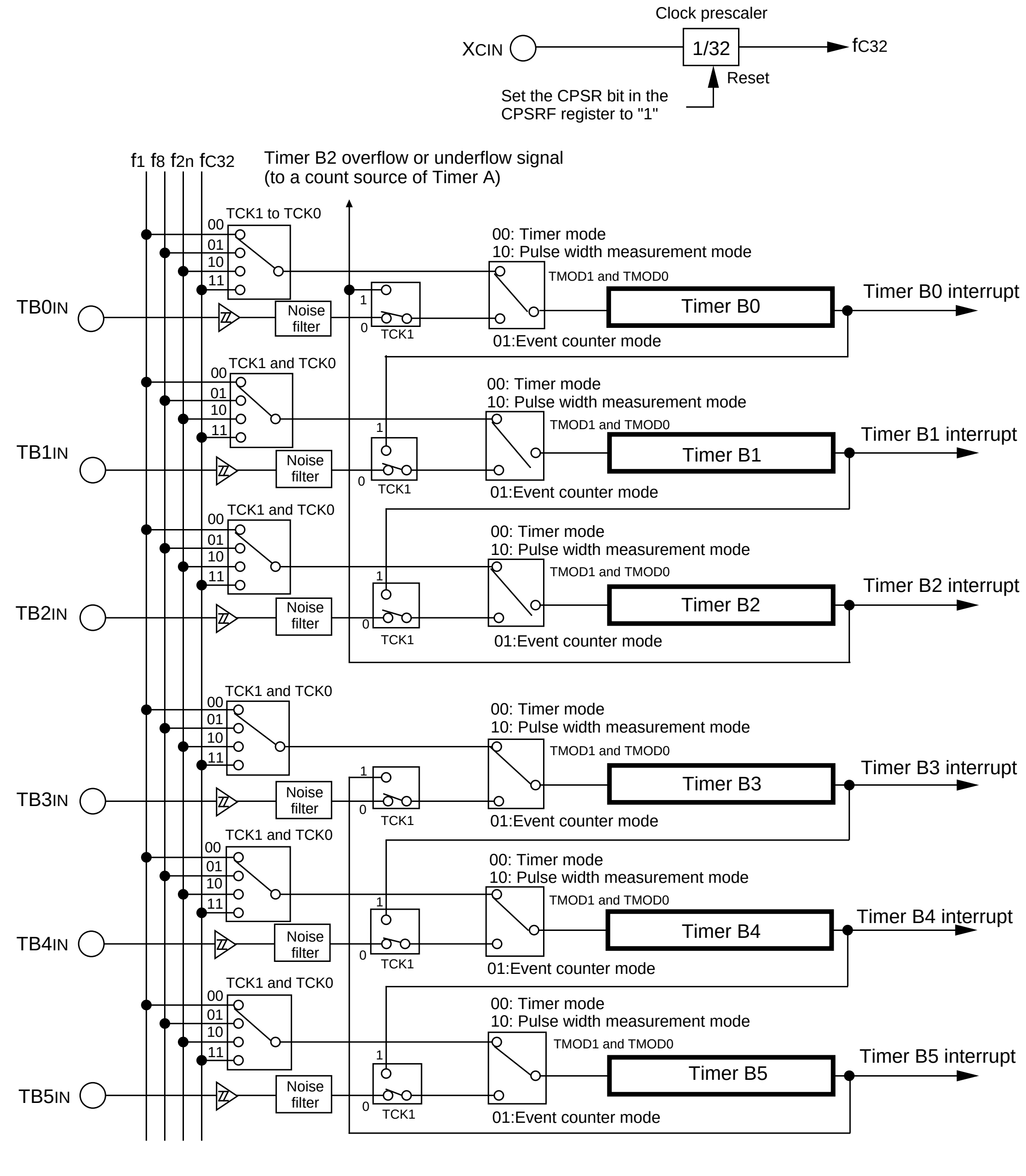
15. Timer

The microcomputer has eleven 16-bit timers. Five timers A and six timers B have different functions. Each timer functions independently. The count source for each timer becomes the clock for timer operations including counting and reloading, etc. Figures 15.1 and 15.2 show block diagrams of timer A and timer B configuration.



CST: Bit in the TCSPR Register
TCK1 and TCK0, TMOD1 and TMOD0 : Bits in the TAiMR Register (i=0 to 4)
TAiTGH and TAiTGL: Bits in the ONSF Register or TRGSR Register

Figure 15.1 Timer A Configuration



CST : Bit in the TCSPR Register
TCK1 and TCK0, TMOD1 and TMOD0 : Bits in the TBiMR Register (i=0 to 5)

Figure 15.2 Timer B Configuration

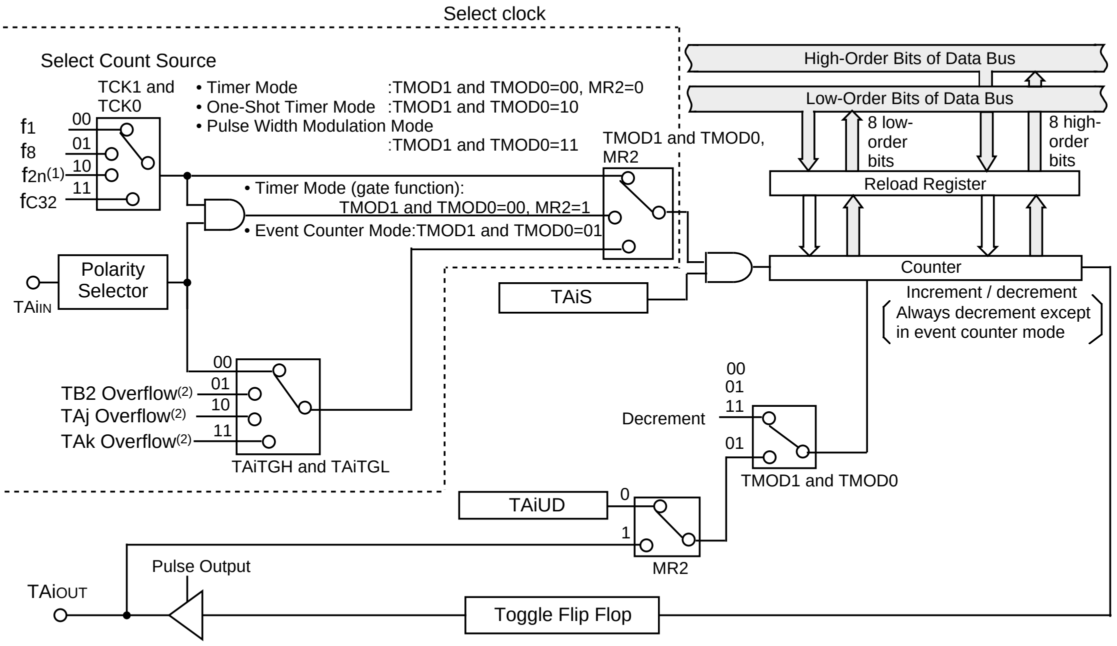
15.1 Timer A

Figure 15.3 shows a block diagram of the timer A. Figures 15.4 to 15.7 show registers associated with the timer A.

The timer A supports the following four modes. Except in event counter mode, all timers A0 to A4 have the same function. The TMOD1 and TMOD0 bits in the TAiMR register (i=0 to 4) determine which mode is used.

- Timer mode: The timer counts an internal count source.
- Event counter mode: The timer counts an external pulse or an overflow and underflow of other timers.
- One-shot timer mode: The timer outputs one valid pulse until a counter value reaches "0000$_{16}$".
- Pulse width modulation mode: The timer continuously outputs desired pulse widths.

Table 15.1 lists TAiOUT pin settings when used as an output. Table 15.2 lists TAiIN and TAiOUT pin settings when used as an input.



i=0 to 4
j=i-1, except j=4 if i=0
k=i+1, except k=0 if i=4

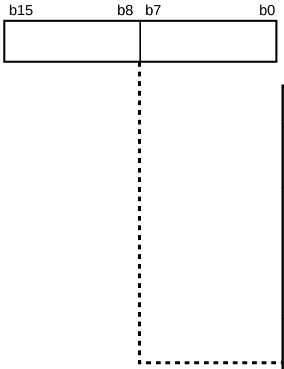
NOTES:
1. The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2n (n=1 to 15).
2. Overflow or underflow signal

TAi	Addresses		TAj	TAk
Timer A0	0347$_{16}$	0346$_{16}$	Timer A4	Timer A1
Timer A1	0349$_{16}$	0348$_{16}$	Timer A0	Timer A2
Timer A2	034B$_{16}$	034A$_{16}$	Timer A1	Timer A3
Timer A3	034D$_{16}$	034C$_{16}$	Timer A2	Timer A4
Timer A4	034F$_{16}$	034E$_{16}$	Timer A3	Timer A0

TCK1 and TCK0, TMOD1 and TMOD0, MR2 and MR1 : Bits in the TAiMR register
TAiTGH and TAiTGL : Bits in the ONSF register if i=0 or bits in the TRGSR register if i=1 to 4
TAiS : Bits in the TABSR register
TAiUD : Bits in the UDF register

Figure 15.3 Timer A Block Diagram

Timer Ai Register (i=0 to 4)[1]



Symbol	Address	After Reset
TA0 to TA2	0347_{16}-0346_{16}, 0349_{16}-0348_{16}, $034B_{16}$-$034A_{16}$	Indeterminate
TA3, TA4	$034D_{16}$-$034C_{16}$, $034F_{16}$-$034E_{16}$	Indeterminate

Mode	Function	Setting Range	RW
Timer Mode	If setting value is *n*, count source is divided by *n+1*.	0000_{16} to $FFFF_{16}$	RW
Event Counter Mode[2]	If setting value is *n*, count source is divided by *FFFF16 - n+1* when the counter is incremented and by *n+1* when the counter is decremented.	0000_{16} to $FFFF_{16}$	RW
One-Shot Timer Mode[4]	If setting value is *n*, count source is divided by *n*, then stops.	0000_{16} to $FFFF_{16}$[3]	WO
Pulse Width Modulation Mode[5] (16-Bit PWM)	If count source frequency is *fj* and setting value of the TAi register is *n*, PWM cycle: $(2^{16}-1) / fj$ "H" width of PWM pulse: n / fj	0000_{16} to $FFFE_{16}$[3]	WO
Pulse Width Modulation Mode[5] (8-Bit PWM)	If count source frequency is *fj*, setting value of high-order bits in the TAi register is *n* and setting value of low-order bits in the TAi register is *m*, PWM cycle: $(2^{8}-1)x(m+1) / fj$ "H" width of PWM pulse: $(m+1)n / fj$	00_{16} to FE_{16}[3] (High-order address bits) 00_{16} to FF_{16}[3] (Low-order address bits)	WO

fj : f_1, f_8, f_{2n}, f_{C32}

NOTES:

1. Use 16-bit data for reading and writing.
2. The TAi register counts how many pulse inputs are provided externally or how many times another timer counter overflows and underflows.
3. Use the MOV instruction to set the TAi register.
4. When the TAi register is set to "0000_{16}", the timer counter does not start and the timer Ai interrupt request is not generated.
5. When the TAi register is set to "0000_{16}", the pulse width modulator does not operate and the TAi_{OUT} pin is held "L". The TAi interrupt request is also not generated. The same situation occurs in 8-bit pulse width modulator mode if the 8 high-order bits in the TAi register are set to "00_{16}".

Figure 15.4 TA0 to TA4 Registers

Timer Ai Mode Register (i=0 to 4)

b7	b6	b5	b4	b3	b2	b1	b0
					0		

Symbol	Address	After Reset
TA0MR to TA4MR	0356_{16}, 0357_{16}, 0358_{16}, 0359_{16}, $035A_{16}$	00_{16}

Bit Symbol	Bit Name	Function	RW
TMOD0	Operating Mode Select Bit	b1b0 0 0 : Timer mode 0 1 : Event counter mode 1 0 : One-shot timer mode 1 1 : Pulse width modulation (PWM) mode	RW
TMOD1			RW
——— (b3)	Reserved Bit	Set to "0"	RW
MR1	———	Function varies depending on operating mode	RW
MR2			RW
MR3			RW
TCK0	Count Source Select Bit	Function varies depending on operating mode	RW
TCK1			RW

Count Start Flag

b7	b6	b5	b4	b3	b2	b1	b0

Symbol	Address	After Reset
TABSR	0340_{16}	00_{16}

Bit Symbol	Bit Name	Function	RW
TA0S	Timer A0 Count Start Flag	0 : Stops counting 1 : Starts counting	RW
TA1S	Timer A1 Count Start Flag	0 : Stops counting 1 : Starts counting	RW
TA2S	Timer A2 Count Start Flag	0 : Stops counting 1 : Starts counting	RW
TA3S	Timer A3 Count Start Flag	0 : Stops counting 1 : Starts counting	RW
TA4S	Timer A4 Count Start Flag	0 : Stops counting 1 : Starts counting	RW
TB0S	Timer B0 Count Start Flag	0 : Stops counting 1 : Starts counting	RW
TB1S	Timer B1 Count Start Flag	0 : Stops counting 1 : Starts counting	RW
TB2S	Timer B2 Count Start Flag	0 : Stops counting 1 : Starts counting	RW

Figure 15.5 TA0MR to TA4MR Registers and TABSR Register

Up/Down Flag(1)

b7 b6 b5 b4 b3 b2 b1 b0

Symbol	Address	After Reset
UDF	0344_{16}	00_{16}

Bit Symbol	Bit Name	Function	RW
TA0UD	Timer A0 Up/Down Flag(2)	0 : Decrement 1 : Increment	RW
TA1UD	Timer A1 Up/Down Flag(2)	0 : Decrement 1 : Increment	RW
TA2UD	Timer A2 Up/Down Flag(2)	0 : Decrement 1 : Increment	RW
TA3UD	Timer A3 Up/Down Flag(2)	0 : Decrement 1 : Increment	RW
TA4UD	Timer A4 Up/Down Flag(2)	0 : Decrement 1 : Increment	RW
TA2P	Timer A2 Two-Phase Pulse Signal Processing Function Select Bit(3)	0 : Disables two-phase pulse signal processing function 1 : Enables two-phase pulse signal processing function	WO
TA3P	Timer A3 Two-Phase Pulse Signal Processing Function Select Bit(3)	0 : Disables two-phase pulse signal processing function 1 : Enables two-phase pulse signal processing function	WO
TA4P	Timer A4 Two-Phase Pulse Signal Processing Function Select Bit(3)	0 : Disables two-phase pulse signal processing function 1 : Enables two-phase pulse signal processing function	WO

NOTES:

1. Use the MOV instruction to set the UDF register.
2. This bit is enabled when the MR2 bit in the TAiMR register (i=0 to 4) is set to "0" (the UDF register causes increment/decrement switching) in event counter mode.
3. Set this bit to "0" when not using the two-phase pulse signal processing function.

One-Shot Start Flag

b7 b6 b5 b4 b3 b2 b1 b0

Symbol	Address	After Reset
ONSF	0342_{16}	00_{16}

Bit Symbol	Bit Name	Function	RW
TA0OS	Timer A0 One-Shot Start Flag(1)	0 : In an idle state 1 : Starts the timer	RW
TA1OS	Timer A1 One-Shot Start Flag(1)	0 : In an idle state 1 : Starts the timer	RW
TA2OS	Timer A2 One-Shot Start Flag(1)	0 : In an idle state 1 : Starts the timer	RW
TA3OS	Timer A3 One-Shot Start Flag(1)	0 : In an idle state 1 : Starts the timer	RW
TA4OS	Timer A4 One-Shot Start Flag(1)	0 : In an idle state 1 : Starts the timer	RW
TAZIE	Z-Phase Input Enable Bit	0 : Disables Z-phase input 1 : Enables Z-phase input	RW
TA0TGL	Timer A0 Event/Trigger Select Bit	b7b6 0 0 : Selects an input to the TA0$_{IN}$ pin 0 1 : Selects the TB2 overflows(2) 1 0 : Selects the TA4 overflows(2) 1 1 : Selects the TA1 overflows(2)	RW
TA0TGH			RW

NOTES:

1. When read, this bit is set to "0".
2. Overflow or underflow.

Figure 15.6 UDF Register and ONSF Register

Trigger Select Register

b7 b6 b5 b4 b3 b2 b1 b0

Symbol	Address	After Reset
TRGSR	0343_{16}	00_{16}

Bit Symbol	Bit Name	Function	RW
TA1TGL	Timer A1 Event/Trigger Select Bit	b1 b0 0 0 : Selects an input to the $TA1_{IN}$ pin 0 1 : Selects the TB2 overflows(1) 1 0 : Selects the TA0 overflows(1) 1 1 : Selects the TA2 overflows(1)	RW
TA1TGH			RW
TA2TGL	Timer A2 Event/Trigger Select Bit	b3 b2 0 0 : Selects an input to the $TA2_{IN}$ pin 0 1 : Selects the TB2 overflows(1) 1 0 : Selects the TA1 overflows(1) 1 1 : Selects the TA3 overflows(1)	RW
TA2TGH			RW
TA3TGL	Timer A3 Event/Trigger Select Bit	b5 b4 0 0 : Selects an input to the $TA3_{IN}$ pin 0 1 : Selects the TB2 overflows(1) 1 0 : Selects the TA2 overflows(1) 1 1 : Selects the TA4 overflows(1)	RW
TA3TGH			RW
TA4TGL	Timer A4 Event/Trigger Select Bit	b7 b6 0 0 : Selects an input to the $TA4_{IN}$ pin 0 1 : Selects the TB2 overflows(1) 1 0 : Selects the TA3 overflows(1) 1 1 : Selects the TA0 overflows(1)	RW
TA4TGH			RW

NOTES:
1. Overflow or underflow

Count Source Prescaler Register

b7 b6 b5 b4 b3 b2 b1 b0

Symbol	Address	After Reset(2)
TCSPR	$035F_{16}$	$0XXX\ 0000_2$

Bit Symbol	Bit Name	Function	RW
CNT0	Divide Ratio Select Bit(1)	If setting value is *n*, $f2n$ is the main clock, on-chip oscillator or PLL clock divided by *2n*. Not divided if *n*=0.	RW
CNT1			RW
CNT2			RW
CNT3			RW
——— (b6 - b4)	Reserved Bit	When read, its content is indeterminate	RO
CST	Operation Enable Bit	0 : Stops a divider 1 : Starts a divider	RW

NOTES:
1. Set the CST bit to "0" before the CNT3 to CNT0 bits are rewritten.
2. The TCSPR register maintains values set before reset, even after software reset or watchdog timer reset has performed.

Figure 15.7 TRGSR Register and TCSPR Register

Table 15.1 Pin Settings for Output from TAi$_{OUT}$ Pin (i=0 to 4)

Pin	Setting		
	PS1, PS2 Registers	PSL1, PSL2 Registers	PSC Register
P7$_0$/TA0$_{OUT}$$^{(1)}$	PS1_0= 1	PSL1_0=1	PSC_0= 0
P7$_2$/TA1$_{OUT}$	PS1_2= 1	PSL1_2=1	PSC_2= 0
P7$_4$/TA2$_{OUT}$	PS1_4= 1	PSL1_4=0	PSC_4= 0
P7$_6$/TA3$_{OUT}$	PS1_6= 1	PSL1_6=1	PSC_6= 0
P8$_0$/TA4$_{OUT}$	PS2_0= 1	PSL2_0=0	–

NOTES:

1. P7$_0$/TA0$_{OUT}$ is a port for the N-channel open drain output.

Table 15.2 Pin Settings for Input to TAi$_{IN}$ and TAi$_{OUT}$ Pins (i=0 to 4)

Pin	Setting	
	PS1, PS2 Registers	PD7, PD8 Registers
P7$_0$/TA0$_{OUT}$	PS1_0=0	PD7_0=0
P7$_1$/TA0$_{IN}$	PS1_1=0	PD7_1=0
P7$_2$/TA1$_{OUT}$	PS1_2=0	PD7_2=0
P7$_3$/TA1$_{IN}$	PS1_3=0	PD7_3=0
P7$_4$TA2$_{OUT}$	PS1_4=0	PD7_4=0
P7$_5$/TA2$_{IN}$	PS1_5=0	PD7_5=0
P7$_6$TA3$_{OUT}$	PS1_6=0	PD7_6=0
P7$_7$/TA3$_{IN}$	PS1_7=0	PD7_7=0
P8$_0$/TA4$_{OUT}$	PS2_0=0	PD8_0=0
P8$_1$/TA4$_{IN}$	PS2_1=0	PD8_1=0

15.1.1 Timer Mode

In timer mode, the timer counts an internally generated count source (see **Table 15.3**). Figure 15.8 shows the TAiMR register (i=0 to 4) in timer mode.

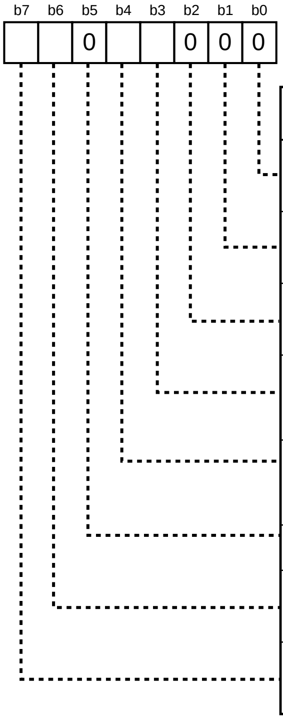
Table 15.3 Timer Mode Specifications

Item	Specification
Count Source	f_1, f_8, f_{2n}(1), f_{C32}
Counting Operation	• The timer decrements a counter value When the timer counter underflows, content of the reload register is reloaded into the count register and counting resumes.
Divide Ratio	1/(n+1) n: setting value of the TAi register (i=0 to 4) 0000_{16} to $FFFF_{16}$
Counter Start Condition	The TAiS bit in the TABSR register is set to "1" (starts counting)
Counter Stop Condition	The TAiS bit is set to "0" (stops counting)
Interrupt Request Generation Timing	The timer counter underflows
TAi_{IN} Pin Function	Programmable I/O port or gate input
TAi_{OUT} Pin Function	Programmable I/O port or pulse output
Read from Timer	The TAi register indicates counter value
Write to Timer	• While the timer counter stops, the value written to the TAi register is also written to both reload register and counter • While counting, the value written to the TAi register is written to the reload register (It is transferred to the counter at the next reload timing)
Selectable Function	• Gate function Input signal to the TAi_{IN} pin determines whether the timer counter starts or stops counting • Pulse output function The polarity of the TAi_{OUT} pin is inversed whenever the timer counter underflows

NOTES:

1. The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2n (n=1 to 15).

Timer Ai Mode Register (i=0 to 4) (Timer Mode)



Symbol	Address	After Reset
TA0MR to TA4MR	0356_{16}, 0357_{16}, 0358_{16}, 0359_{16}, $035A_{16}$	00_{16}

Bit Symbol	Bit Name	Function	RW
TMOD0	Operating Mode Select Bit	b1b0 0 0 : Timer mode	RW
TMOD1			RW
—— (b2)	Reserved Bit	Set to "0"	RW
MR1	Gate Function Select Bit	b4b3 0 X : Gate function disabled[1] (TAi$_{IN}$ pin is a programmable I/O pin) 1 0 : Timer counts only while the TAi$_{IN}$ pin is held "L" 1 1 : Timer counts only while the TAi$_{IN}$ pin is held "H"	RW
MR2			RW
MR3	Set to "0" in timer mode		RW
TCK0	Count Source Select Bit	b7b6 0 0 : f_1 0 1 : f_8 1 0 : f_{2n}[2] 1 1 : f_{C32}	RW
TCK1			RW

NOTES:

1. X can be set to either "0" or "1".
2. The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2n (n=1 to 15).

Figure 15.8 TA0MR to TA4MR Registers

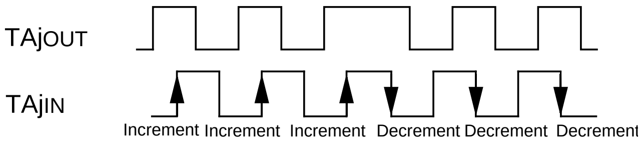
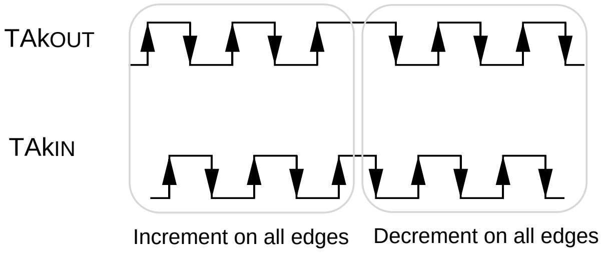
15.1.2 Event Counter Mode

In event counter mode, the timer counts how many external signals are applied or how many times another timer counter overflows and underflows. The timers A2, A3 and A4 can count externally generated two-phase signals. Table 15.4 lists specifications in event counter mode (when not handling a two-phase pulse signal). Table 15.5 lists specifications in event counter mode (when handling a two-phase pulse signal with the timers A2, A3 and A4). Figure 15.9 shows the TAiMR register (i=0 to 4) in event counter mode.

Table 15.4 Event Counter Mode Specifications (When Not Processing Two-phase Pulse Signal)

Item	Specification
Count Source	• External signal applied to the TAiIN pin (i = 0 to 4) (valid edge can be selected by program) • Timer B2 overflow or underflow signal, timer Aj overflow or underflow signal (j=i-1, except j=4 if i=0) and timer Ak overflow or underflow signal (k=i+1, except k=0 if i=4)
Counting Operation	• External signal and program can determine whether the timer increments or decrements a counter value • When the timer counter underflows or overflows, content of the reload register is reloaded into the count register and counting resumes. When the free-running count function is selected, the timer counter continues running without reloading.
Divide Ratio	• *1/(FFFF16 - n + 1)* for counter increment • *1/(n + 1)* for counter decrement *n* : setting value of the TAi register 0000_{16} to $FFFF_{16}$
Counter Start Condition	The TAiS bit in the TABSR register is set to "1" (starts counting)
Counter Stop Condition	The TAiS bit is set to "0" (stops counting)
Interrupt Request Generation Timing	The timer counter overflows or underflows
TAiIN Pin Function	Programmable I/O port or count source input
TAiOUT Pin Function	Programmable I/O port, pulse output or input selecting a counter increment or decrement
Read from Timer	The TAi register indicates counter value
Write to Timer	• When the timer counter stops, the value written to the TAi register is also written to both reload register and counter • While counting, the value written to the TAi register is written to the reload register (It is transferred to the counter at the next reload timing)
Selectable Function	• Free-running count function Content of the reload register is not reloaded even if the timer counter overflows or underflows • Pulse output function The polarity of the TAiOUT pin is inversed whenever the timer counter overflows or underflows

Table 15.5 Event Counter Mode Specifications (When Processing Two-phase Pulse Signal on Timer A2, A3 and A4)

Item	Specification
Count Source	Two-phase pulse signal applied to the TAiIN and TAiOUT pins (i = 2 to 4)
Counting Operation	• Two-phase pulse signal determines whether the timer increments or decrements a counter value • When the timer counter overflows or underflows, content of the reload register is reloaded into the count register and counting resumes. With the free-running count function, the timer counter continues running without reloading.
Divide Ratio	• $1/(FFFF_{16} - n + 1)$ for counter increment • $1/(n + 1)$ for counter decrement n : setting value of the TAi register 0000_{16} to $FFFF_{16}$
Counter Start Condition	The TAiS bit in the TABSR register is set to "1" (starts counting)
Counter Stop Condition	The TAiS bit is set to "0" (stops counting)
Interrupt Request Generation Timing	The timer counter overflows or underflows
TAiIN Pin Function	Two-phase pulse signal is applied
TAiOUT Pin Function	Two-phase pulse signal is applied
Read from Timer	The TAi register indicates the counter value
Write to Timer	• When the timer counter stops, the value written to the TAi register is also written to both reload register and counter • While counting, the value written to the TAi register is written to the reload register (It is transferred to the counter at the next reload timing)
Selectable Function[1]	• Normal processing operation (the timer A2 and timer A3) While a high-level ("H") signal is applied to the TAjOUT pin (j = 2 or 3), the timer increments a counter value on the rising edge of the TAjIN pin or decrements a counter on the falling edge.  TAjOUT TAjIN Increment Increment Increment Decrement Decrement Decrement • Multiply-by-4 processing operation (the timer A3 and timer A4) While an "H" signal is applied to the TAkOUT pin (k = 3 or 4) on the rising edge of the TAkIN pin, the timer increments a counter value on the rising and falling edges of the TAkOUT and TAkIN pins. While an "H" signal is applied to the TAkOUT pin on the falling edge of the TAkIN pin, the timer decrements a counter value on the rising and falling edges of the TAkOUT and TAkIN pins.  TAkOUT TAkIN Increment on all edges Decrement on all edges

NOTES:

1. Only timer A3 operation can be selected. The timer A2 is for the normal processing operation. The timer A4 is for the multiply-by-4 operation.

Timer Ai Mode Register (i=0 to 4) (Event Counter Mode)

b7	b6	b5	b4	b3	b2	b1	b0
		0			0	0	1

Symbol	Address	After Reset
TA0MR to TA4MR	0356_{16}, 0357_{16}, 0358_{16}, 0359_{16}, $035A_{16}$	00_{16}

Bit Symbol	Bit Name	Function (When not processing two-phase pulse signal)	Function (When processing two-phase pulse signal)	RW
TMOD0	Operating Mode Select Bit	b1b0 0 1 : Event counter mode(1)		RW
TMOD1				RW
—— (b2)	Reserved Bit	Set to "0"		RW
MR1	Count Polarity Select Bit(2)	0 : Counts falling edges of an external signal 1 : Counts rising edges of an external signal	Set to "0"	RW
MR2	Increment/Decrement Switching Source Select Bit	0 : UDF registser setting 1 : Input signal to TAiOUT pin(3)	Set to "1"	RW
MR3	Set to "0" in event counter mode			RW
TCK0	Count Operation Type Select Bit	0 : Reloading 1 : Free running		RW
TCK1	Two-Phase Pulse Signal Processing Operation Select Bit(4,5)	Set to "0"	0 : Normal processing operation 1 : Multiply-by-4 processing operation	RW

NOTES:

1. The TAiTGH and TAiTGL bits in the ONSF or TRGSR register determine the count source in the event counter mode.
2. MR1 bit setting is enabled only when counting how many times external signals are applied.
3. The timer decrements a counter value when an "L" signal is applied to the TAiOUT pin and the timer increments a counter value when an "H" signal is applied to the TAiOUT pin.
4. The TCK1 bit is enabled only in the TA3MR register.
5. For two-phase pulse signal processing, set the TAjP bit in the UDF register (j=2 to 4) to "1" (two-phase pulse signal processing function enabled). Also, set the TAiTGH and TAiTGL bits to "00_2" (input to the TAjIN pin).

Figure 15.9 TA0MR to TA4MR Registers

15.1.2.1 Counter Reset by Two-Phase Pulse Signal Processing

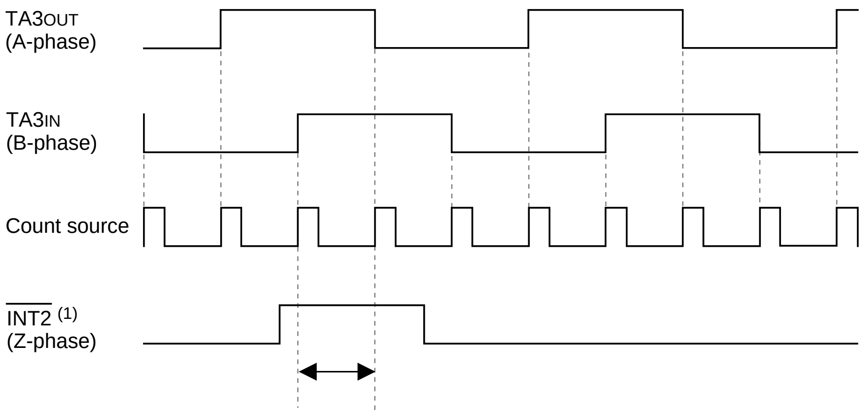
Z-phase input resets the timer counter when processing a two-phase pulse signal.

This function can be used in timer A3 event counter mode, two-phase pulse signal processing, free-running count operation type or multiply-by-4 processing. The Z-phase signal is applied to the $\overline{\text{INT2}}$ pin. When the TAZIE bit in the ONSF register is set to "1" (Z-phase input enabled), Z-phase input can reset the timer counter. To reset the counter by a Z-phase input, set the TA3 register to "0000$_{16}$" beforehand.

Z-phase input is enabled when the edge of the signal applied to the $\overline{\text{INT2}}$ pin is detected. The POL bit in the INT2IC register can determine edge polarity. The Z-phase must have a pulse width of one timer A3 count source cycle or more . Figure 15.10 shows two-phase pulses (A-phase and B-phase) and the Z-phase.

Z-phase input resets the timer counter in the next count source following Z-phase input. Figure 15.11 shows the counter reset timing.

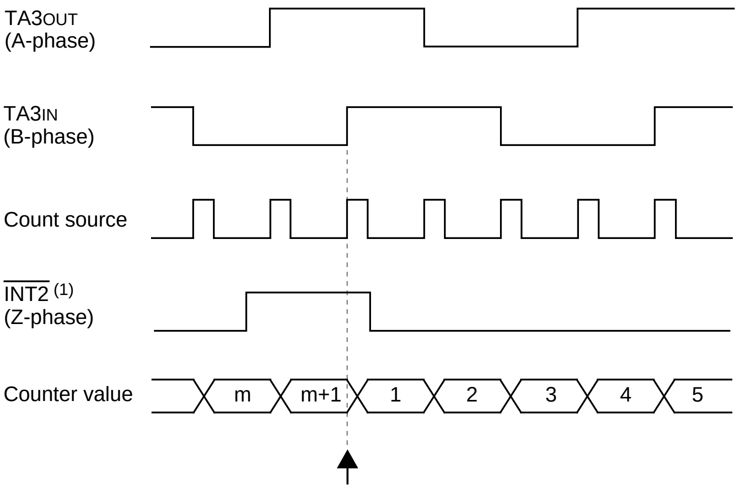
Timer A3 interrupt request is generated twice continuously when a timer A3 overflow or underflow, and a counter reset by $\overline{\text{INT2}}$ input occur at the same time. Do not use the timer A3 interrupt request when this function is used.



Pulse width of one count source cycle or more is required

NOTES:
1. When the rising edge of $\overline{\text{INT2}}$ is selected.

Figure 15.10 Two-Phase Pulse (A-phase and B-phase) and Z-phase



Timer counter is reset at this timing

NOTES:
1. When the rising edge of $\overline{\text{INT2}}$ is selected.

Figure 15.11 Counter Reset Timing

15.1.3 One-Shot Timer Mode

In one-shot timer mode, the timer operates only once for each trigger (see **Table 15.6**). Once a trigger occurs, the timer starts and continues operating for a desired period. Figure 15.12 shows the TAiMR register (i=0 to 4) in one-shot timer mode.

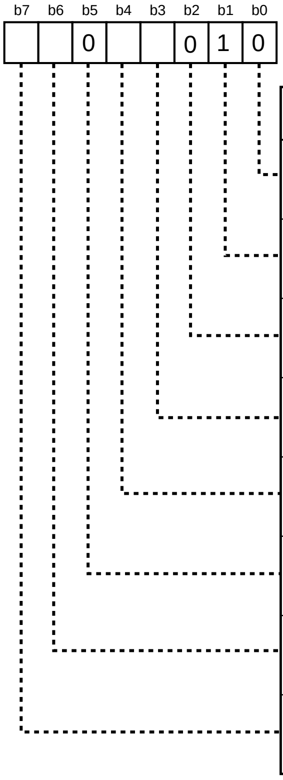
Table 15.6 One-Shot Timer Mode Specifications

Item	Specification
Count Source	f_1, f_8, f_{2n}(1), f_{C32}
Counting Operation	• The timer decrements a counter value When the timer counter reaches "0000_{16}", it stops counting after reloading. If a trigger occurs while counting, content of the reload register is reloaded into the count register and counting resumes.
Divide Ratio	*1/n* *n* : setting value of the TAi register (i=0 to 4) 0000_{16} to $FFFF_{16}$, but the timer counter does not run if n=0000_{16}
Counter Start Condition	The TAiS bit in the TABSR register is set to "1" (starts counting) and following triggers occur: • External trigger input is provided • Timer counter overflows or underflows • The TAiOS bit in the ONSF register is set to "1" (timer started)
Counter Stop Condition	• After the timer counter has reached "0000_{16}" and is reloaded • When the TAiS bit is set to "0" (stops counting)
Interrupt Request Generation Timing	The timer counter reaches "0000_{16}"
TAiIN Pin Function	Programmable I/O port or trigger input
TAiOUT Pin Function	Programmable I/O port or pulse output
Read from Timer	The value in the TAi register is indeterminate when read
Write to Timer	• When the timer counter stops, the value written to the TAi register is also written to both reload register and counter • While counting, the value written to the TAi register is written to the reload register (It is transferred to the counter at the next reload timing)

NOTES:

1. The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2n (n=1 to 15).

Timer Ai Mode Register (i=0 to 4) (One-Shot Timer Mode)



Symbol	Address	After Reset
TA0MR to TA4MR	0356_{16}, 0357_{16}, 0358_{16}, 0359_{16}, $035A_{16}$	00_{16}

Bit Symbol	Bit Name	Function	RW
TMOD0	Operating Mode Select Bit	b1b0 1 0 : One-shot timer mode	RW
TMOD1			RW
(b2)	Reserved Bit	Set to "0"	—
MR1	External Trigger Select Bit[1]	0 : Falling edge of input signal to TAiIN pin 1 : Rising edge of input signal to TAiIN pin	RW
MR2	Trigger Select Bit	0 : The TAiOS bit is enabled 1 : Selected by the TAiTGH and TAiTGL bits	RW
MR3	Set to "0" in the one-shot timer mode		RW
TCK0	Count Source Select Bit	b7b6 0 0 : f_1 0 1 : f_8 1 0 : f_{2n}[2] 1 1 : f_{C32}	RW
TCK1			RW

NOTES:

1. The MR1 bit setting is enabled only when the TAiTGH and TAiTGL bits in the TRGSR register are set to "00_2" (input to the TAiIN pin). The MR1 bit can be set to either "0" or "1" when the TAiTGH and TAiTGL bits are set to "01_2" (TB2 overflow and underflow), "10_2" (TAi overflow and underflow) or "11_2" (TAi overflow and underflow).
2. The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2n (n=1 to 15).

Figure 15.12 TA0MR to TA4MR Registers

15.1.4 Pulse Width Modulation Mode

In pulse width modulation mode, the timer outputs pulse of desired width continuously (see **Table 15.7**). The timer counter functions as either 16-bit pulse width modulator or 8-bit pulse width modulator. Figure 15.13 shows the TAiMR register (i=0 to 4) in pulse width modulation mode. Figures 15.14 and 15.15 show examples of how a 16-bit pulse width modulator operates and of how an 8-bit pulse width modulator operates.

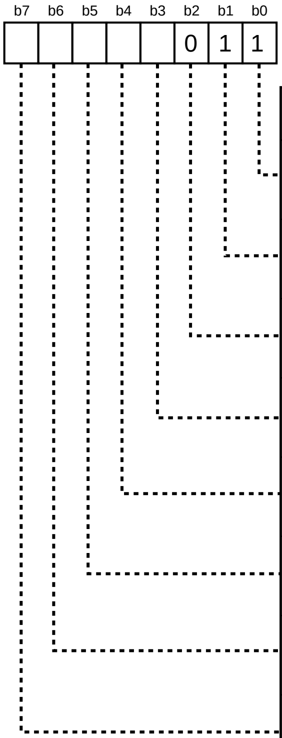
Table 15.7 Pulse Width Modulation Mode Specifications

Item	Specification
Count Source	f_1, f_8, f_{2n}(1), f_{C32}
Counting Operation	• The timer decrements a counter value (The counter functions as an 8-bit or a 16-bit pulse width modulator) Content of the reload register is reloaded on the rising edge of PWM pulse and counting continues. The timer is not affected by a trigger that is generated during counting.
16-Bit PWM	• "H" width = n / f_j n : setting value of the TAi register 0000_{16} to $FFFE_{16}$ f_j : count source frequency • Cycle = $(2^{16}-1) / f_j$ fixed
8-Bit PWM	• "H" width $= n \times (m+1) / f_j$ • Cycles = $(2^8-1) \times (m+1) / f_j$ m : setting value of low-order bit address of the TAi register 00_{16} to FF_{16} n : setting value of high-order bit address of the TAi register 00_{16} to FE_{16}
Counter Start Condition	• External trigger input is provided • Timer counter overflows or underflows • The TAiS bit in the TABSR register is set to "1" (starts counting)
Counter Stop Condition	The TAiS bit is set to "0" (stops counting)
Interrupt Request Generation Timing	On the falling edge of the PWM pulse
TAiIN Pin Function	Programmable I/O port or trigger input
TAiOUT Pin Function	Pulse output
Read from Timer	The value in the TAi register is indeterminate when read
Write to Timer	• When the timer counter stops, the value written to the TAi register is also written to both reload register and counter • While counting, the value written to the TAi register is written to the reload register (It is transferred to the counter at the next reload timing)

NOTES:

1. The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2n (n=1 to 15).

Timer Ai Mode Register (i=0 to 4) (Pulse Width Modulator Mode)



Symbol	Address	After Reset
TA0MR to TA4MR	0356_{16}, 0357_{16}, 0358_{16}, 0359_{16}, $035A_{16}$	00_{16}

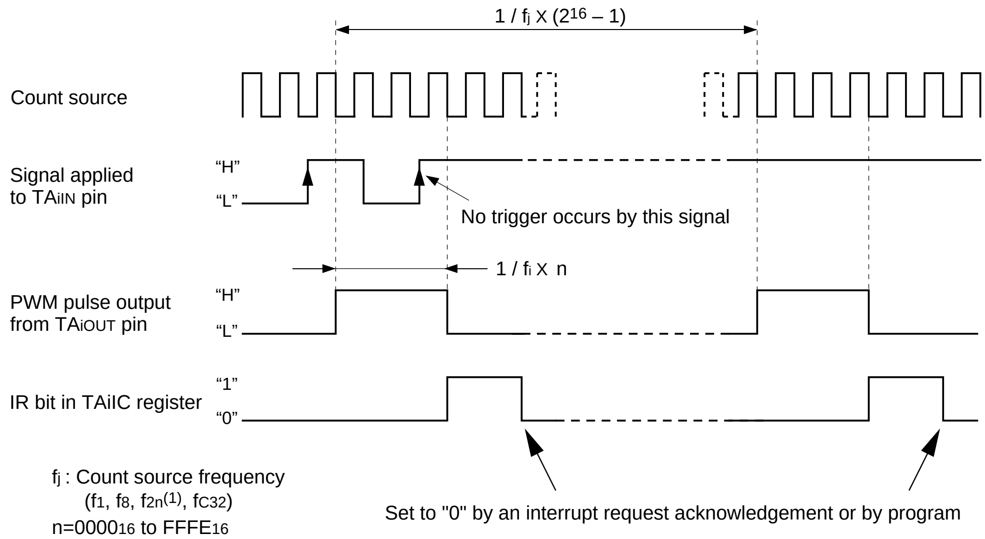
Bit Symbol	Bit Name	Function	RW
TMOD0	Operating Mode Select Bit	b1b0 1 1 : Pulse width modulation (PWM) mode	RW
TMOD1			RW
(b2)	Reserved Bit	Set to "0"	RW
MR1	External Trigger Select Bit[1]	0 : Falling edge of input signal to TAi$_{IN}$ pin 1 : Rising edge of input signal to TAi$_{IN}$ pin	RW
MR2	Trigger Select Bit	0 : The TAiS bit is enabled 1 : Selected by the TAiTGH and TAiTGL bits	RW
MR3	16/8-Bit PWM Mode Select Bit	0: Functions as a 16-bit pulse width modulator 1: Functions as an 8-bit pulse width modulator	RW
TCK0	Count Source Select Bit	b7b6 0 0 : f_1 0 1 : f_8 1 0 : f_{2n}[2] 1 1 : f_{C32}	RW
TCK1			RW

NOTES:

1. MR1 bit setting is enabled only when the TAiTGH and TAiTGL bits in the TRGSR register are set to "00_2" (input to the TAi$_{IN}$ pin). The MR1 bit can be set to either "0" or "1" when the TAiTGH and TAiTGL bits are set to "01_2" (TB2 overflow and underflow), "10_2" (TAi overflow and underflow) or "11_2" (TAi overflow and underflow).
2. The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2n (n=1 to 15).

Figure 15.13 TA0MR to TA4MR Registers

When the reload register is set to "0003_{16}" and an external trigger (on rising edge of a signal applied to the TAiIN pin) is selected

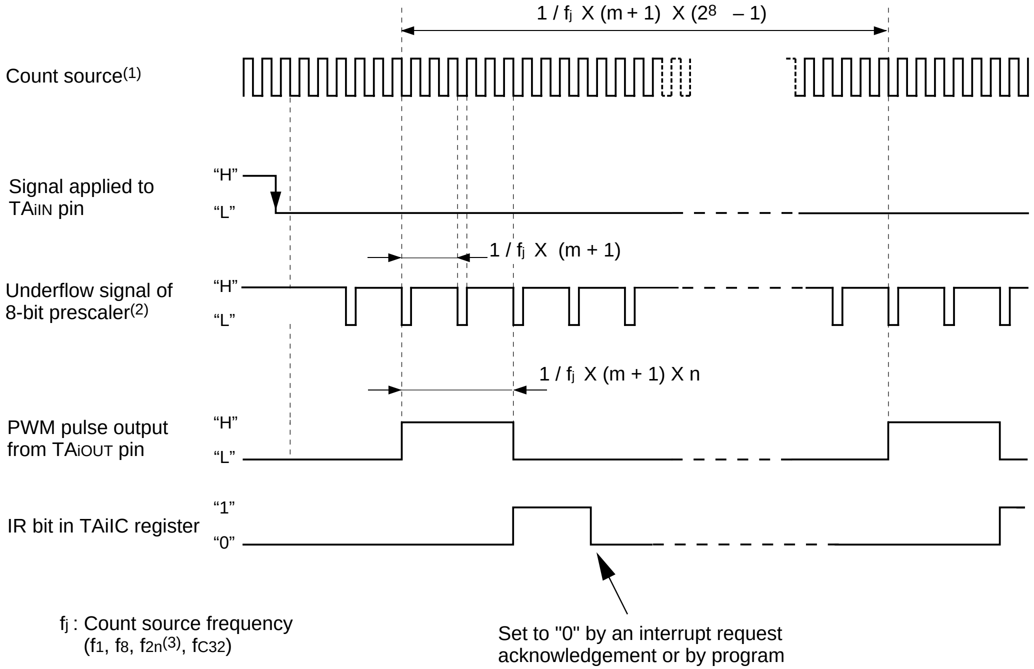

1 / fj X (2^16 – 1)
Count source
Signal applied to TAiIN pin
"H"
"L"
No trigger occurs by this signal
1 / fi X n
PWM pulse output from TAiOUT pin
"H"
"L"
IR bit in TAiIC register
"1"
"0"
Set to "0" by an interrupt request acknowledgement or by program

f_j : Count source frequency
(f_1, f_8, f_{2n}[1], f_{C32})
n=0000_{16} to $FFFE_{16}$
i=0 to 4

NOTES:
1. The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2n (n=1 to 15).

Figure 15.14 16-bit Pulse Width Modulator Operation

When 8 high-order bits of the reload register are set to "02_{16}", 8 low-order bits of the reload register are set to "02_{16}" and an external trigger (on falling edge of a signal applied to the TAiIN pin) is selected


1 / fj X (m + 1) X (2^8 – 1)
Count source(1)
Signal applied to TAiIN pin
"H"
"L"
1 / fj X (m + 1)
Underflow signal of 8-bit prescaler(2)
"H"
"L"
1 / fj X (m + 1) X n
PWM pulse output from TAiOUT pin
"H"
"L"
IR bit in TAiIC register
"1"
"0"
Set to "0" by an interrupt request acknowledgement or by program

f_j : Count source frequency
(f_1, f_8, f_{2n}[3], f_{C32})
m=00_{16} to FF_{16}, n=00_{16} to FE_{16}
i=0 to 4

NOTES:
1. 8-bit prescaler counts a count source.
2. 8-bit pulse width modulator counts underflow signals of the 8-bit prescaler.
3. The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2n (n=1 to 15).

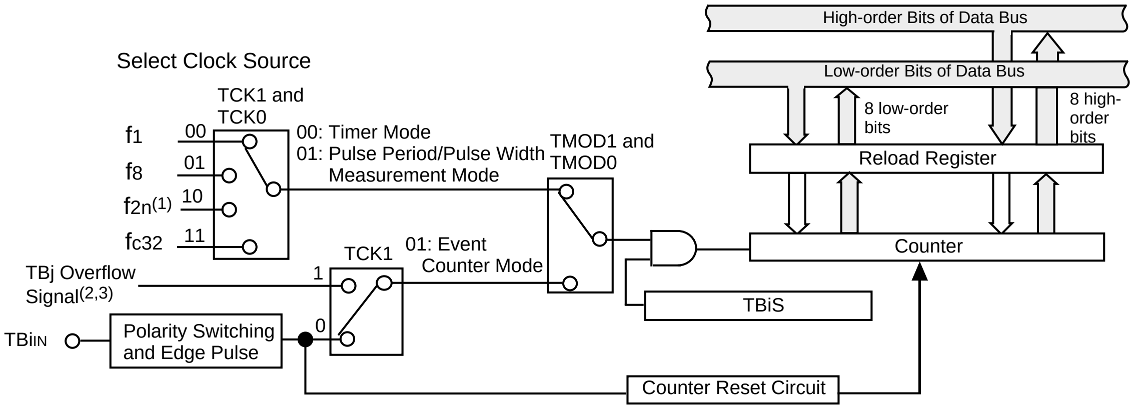
Figure 15.15 8-bit Pulse Width Modulator Operation

15.2 Timer B

Figure 15.16 shows a block diagram of the timer B. Figures 15.17 to 15.19 show registers associated with the timer B. The timer B supports the following three modes. The TMOD1 and TMOD0 bits in the TBiMR register (i=0 to 5) determine which mode is used.

- Timer mode : The timer counts an internal count source.
- Event counter mode : The timer counts pulses from an external source or overflow and underflow of another timer.
- Pulse period/pulse width measurement mode : The timer measures pulse period or pulse width of an external signal.

Table 15.8 lists TBiIN pin settings.



i=0 to 5

NOTES:

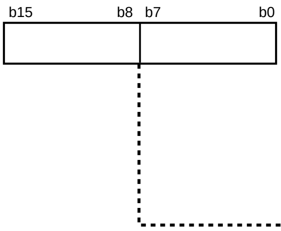
1. The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2n (n=1 to 15).
2. Overflow signal or underflow signal.
3. j=i-1, except j=2 when i=0 j=5 when i=3

TCK1 and TCK0, TMOD1 and TMOD0 : Bits in the TBiMR Register
TBiS : Bits in the TABSR and the TBSR Register

TBi	Address		TBj
Timer B0	0351_{16}	0350_{16}	Timer B2
Timer B1	0353_{16}	0352_{16}	Timer B0
Timer B2	0355_{16}	0354_{16}	Timer B1
Timer B3	0311_{16}	0310_{16}	Timer B5
Timer B4	0313_{16}	0312_{16}	Timer B3
Timer B5	0315_{16}	0314_{16}	Timer B4

Figure 15.16 Timer B Block Diagram

Timer Bi Register(1) (i=0 to 5)



Symbol	Address	After Reset
TB0 to TB2	0351_{16} - 0350_{16}, 0353_{16} - 0352_{16}, 0355_{16} - 0354_{16}	Indeterminate
TB3 to TB5	0311_{16} - 0310_{16}, 0313_{16} - 0312_{16}, 0315_{16} - 0314_{16}	Indeterminate

Mode	Function	Setting Range	RW
Timer Mode	If setting value is *n*, a count source is divided by *n+1*	0000_{16} to $FFFF_{16}$	RW
Event Counter Mode	If setting value is *n*, a count source is divided by *n+1*(2)	0000_{16} to $FFFF_{16}$	RW
Pulse Period/Pulse Width Measurement Mode	A count source is incremented between one valid edge and another valid edge of TBiIN pulse	————	RO

NOTES:

1. Use 16-bit data for reading and writing.
2. The TBi register counts how many pulse inputs are provided externally or how many times another timer counter overflows and underflows.

Figure 15.17 TB0 to TB5 Registers

Timer Bi Mode Register (i=0 to 5)

b7 b6 b5 b4 b3 b2 b1 b0

Symbol	Address	After Reset
TB0MR to TB5MR	$035B_{16}$, $035C_{16}$, $035D_{16}$, $031B_{16}$, $031C_{16}$, $031D_{16}$	$00XX\ 0000_2$

Bit Symbol	Bit Name	Function	RW
TMOD0	Operating Mode Select Bit	b1b0 0 0 : Timer mode 0 1 : Event counter mode 1 0 : Pulse period measurement mode, pulse width measurement mode 1 1 : Do not set to this value	RW
TMOD1			RW
MR0	________	Function varies depending on operating mode (1, 2)	RW
MR1			RW
MR2			RW
MR3			RW
TCK0	Count Source Select Bit	Function varies depending on operating mode	RW
TCK1			RW

NOTES:

1. Only MR2 bits in the TB0MR and TB3MR registers are enabled.
2. Nothing is assigned in the MR2 bit in the TB1MR, TB2MR, TB4MR and TB5MR registers. When write, set to "0". When read, its content is indeterminate.

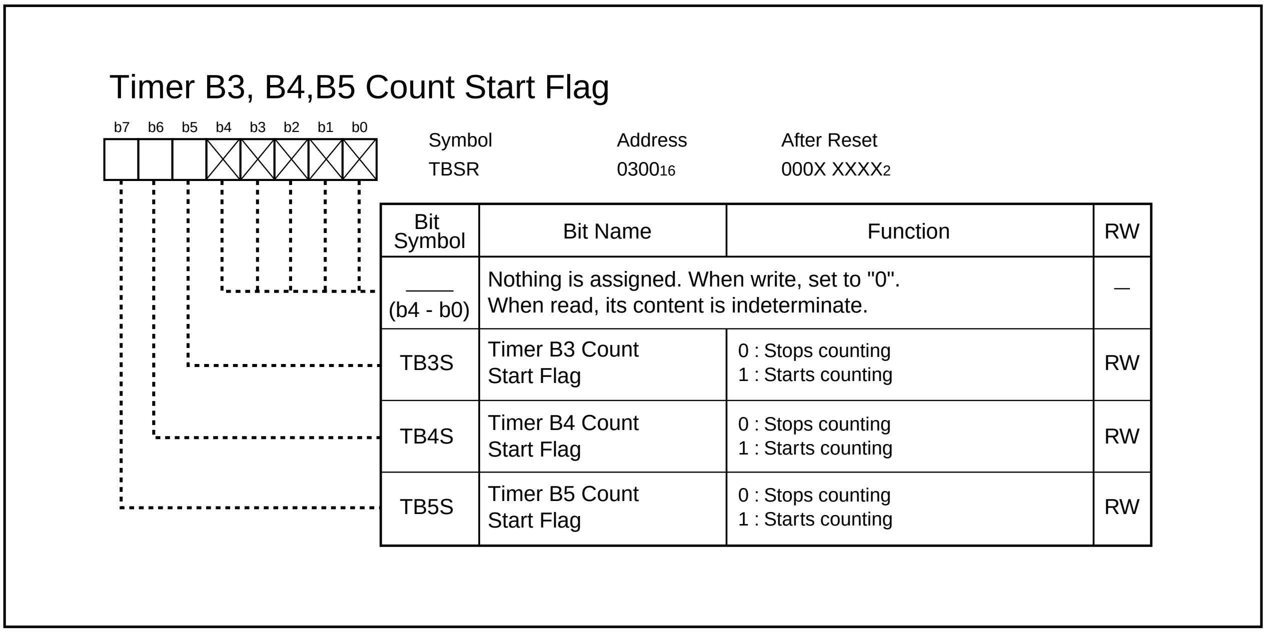
Count Start Flag

b7 b6 b5 b4 b3 b2 b1 b0

Symbol	Address	After Reset
TABSR	0340_{16}	00_{16}

Bit Symbol	Bit Name	Function	RW
TA0S	Timer A0 Count Start Flag	0 : Stops counting 1 : Starts counting	RW
TA1S	Timer A1 Count Start Flag	0 : Stops counting 1 : Starts counting	RW
TA2S	Timer A2 Count Start Flag	0 : Stops counting 1 : Starts counting	RW
TA3S	Timer A3 Count Start Flag	0 : Stops counting 1 : Starts counting	RW
TA4S	Timer A4 Count Start Flag	0 : Stops counting 1 : Starts counting	RW
TB0S	Timer B0 Count Start Flag	0 : Stops counting 1 : Starts counting	RW
TB1S	Timer B1 Count Start Flag	0 : Stops counting 1 : Starts counting	RW
TB2S	Timer B2 Count Start Flag	0 : Stops counting 1 : Starts counting	RW

Figure 15.18 TB0MR to TB5MR Registers, TABSR Register



Symbol	Address	After Reset
TBSR	0300$_{16}$	000X XXXX$_2$

Bit Symbol	Bit Name	Function	RW
____ (b4 - b0)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		—
TB3S	Timer B3 Count Start Flag	0 : Stops counting 1 : Starts counting	RW
TB4S	Timer B4 Count Start Flag	0 : Stops counting 1 : Starts counting	RW
TB5S	Timer B5 Count Start Flag	0 : Stops counting 1 : Starts counting	RW

Figure 15.19 TBSR Register

Table 15.8 Settings for the TBi$_{IN}$ Pins (i=0 to 5)

Port Name	Function	Setting	
		PS1, PS3(1) Registers	PD7, PD9(1) Registers
P9$_0$	TB0$_{IN}$	PS3_0=0	PD9_0=0
P9$_1$	TB1$_{IN}$	PS3_1=0	PD9_1=0
P9$_2$	TB2$_{IN}$	PS3_2=0	PD9_2=0
P9$_3$	TB3$_{IN}$	PS3_3=0	PD9_3=0
P9$_4$	TB4$_{IN}$	PS3_4=0	PD9_4=0
P7$_1$	TB5$_{IN}$	PS1_1=0	PD7_1=0

NOTES:

1. Set the PD9 and PS3 registers immediately after the PRC2 bit in the PRCR register is set to "1" (write enable). Do not generate an interrupt or a DMA transfer between the instruction to set the PRC2 bit to "1" and the instruction to set the PD9 and PS3 registers.

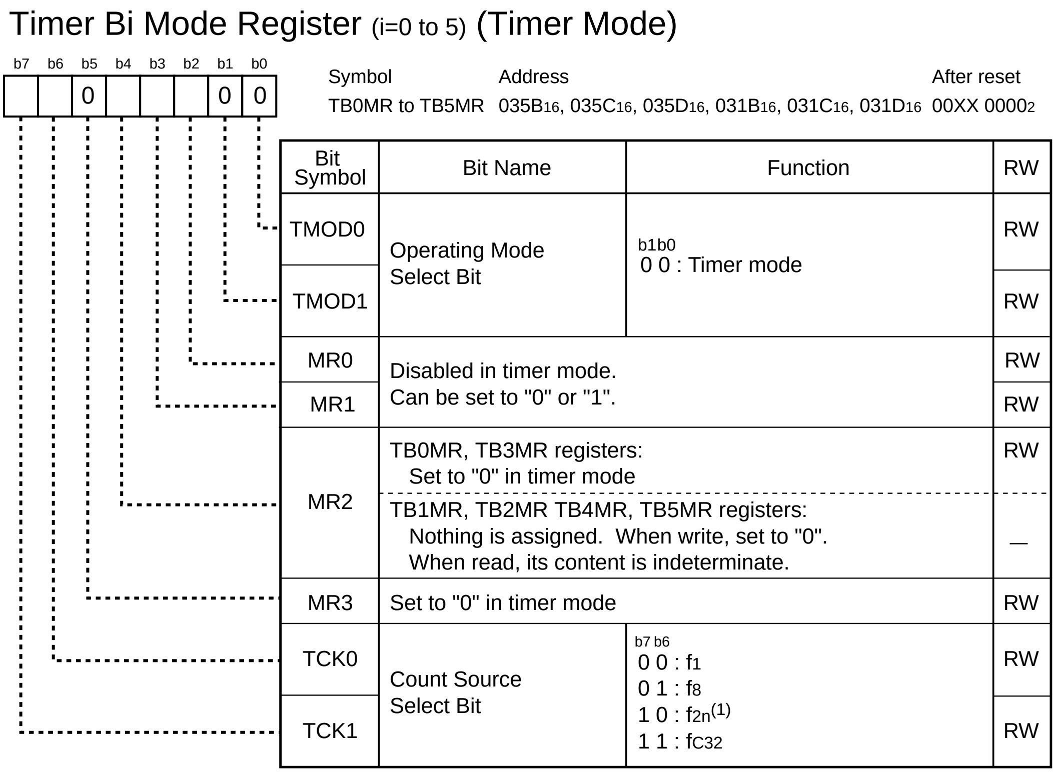
15.2.1 Timer Mode

In timer mode, the timer counts an internally generated count source (see **Table 15.9**). Figure 15.20 shows the TBiMR register (i=0 to 5) in timer mode.

Table 15.9 Timer Mode Specifications

Item	Specification
Count Source	f_1, f_8, f_{2n}(1), f_{C32}
Counting Operation	• The timer decrements a counter value When the timer counter underflows, content of the reload register is reloaded into the count register and counting resumes
Divide Ratio	$1/(n+1)$ n: setting value of the TBi register (i=0 to 5) 0000_{16} to $FFFF_{16}$
Counter Start Condition	The TBiS bits in the TABSR and TBSR registers are set to "1" (starts counting)
Counter Stop Condition	The TBiS bit is set to "0" (stops counting)
Interrupt Request Generation Timing	Timer counter underflows
TBi$_{IN}$ Pin Function	Programmable I/O port
Read from Timer	The TBi register indicates counter value
Write to Timer	• When the timer counter stops, the value written to the TBi register is also written to both reload register and counter • While counting, the value written to the TBi register is written to the reload register (It is transferred to the counter at the next reload timing)

NOTES:
1. The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2n (n=1 to 15).



Timer Bi Mode Register (i=0 to 5) (Timer Mode)

b7	b6	b5	b4	b3	b2	b1	b0
		0				0	0

Symbol	Address	After reset
TB0MR to TB5MR	$035B_{16}$, $035C_{16}$, $035D_{16}$, $031B_{16}$, $031C_{16}$, $031D_{16}$	00XX 0000_2

Bit Symbol	Bit Name	Function	RW
TMOD0	Operating Mode Select Bit	b1b0 0 0 : Timer mode	RW
TMOD1			RW
MR0	Disabled in timer mode. Can be set to "0" or "1".		RW
MR1			RW
MR2	TB0MR, TB3MR registers: Set to "0" in timer mode		RW
	TB1MR, TB2MR TB4MR, TB5MR registers: Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		—
MR3	Set to "0" in timer mode		RW
TCK0	Count Source Select Bit	b7 b6 0 0 : f_1 0 1 : f_8 1 0 : f_{2n}(1) 1 1 : f_{C32}	RW
TCK1			RW

NOTES:
1. The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2n (n=1 to 15).

Figure 15.20 TB0MR to TB5MR Registers

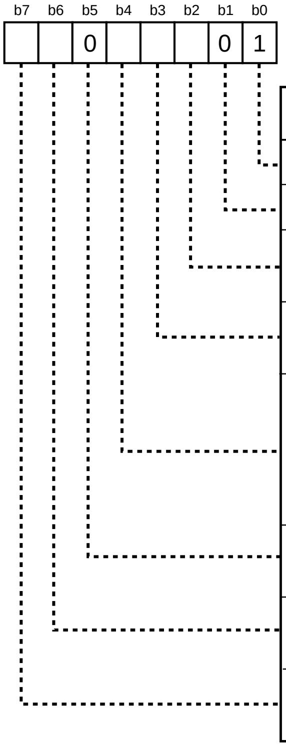
15.2.2 Event Counter Mode

In event counter mode, the timer counts how many external signals are applied or how many times another timer overflows and underflows. (See **Table 15.10**) Figure 15.21 shows the TBiMR register (i=0 to 5) in event counter mode.

Table 15.10 Event Counter Mode Specifications

Item	Specification
Count Source	• External signal applied to the TBiIN pin (i = 0 to 5) (valid edge can be selected by program) • TBj overflow or underflow signal (j=i-1, except j=2 when i=0, j=5 when i=3)
Counting Operation	• The timer decrements a counter value When the timer counter underflows, content of the reload register is reloaded into the count register to continue counting
Divide Ratio	*1/(n+1)* *n* : setting value of the TBi register 0000_{16} to $FFFF_{16}$
Counter Start Condition	The TBiS bits in the TABSR and TBSR register are set to "1" (starts counting)
Counter Stop Condition	The TBiS bit is set to "0" (stops counting)
Interrupt Request Generation Timing	The timer counter underflows
TBiIN Pin Function	Programmable I/O port or count source input
Read from Timer	The TBi register indicates counter value
Write to Timer	• When the timer counter stops, the value written to the TBi register is also written to both reload register and counter • While counting, the value written to the TBi register is written to the reload register (It is transferred to the counter at the next reload timing)

Timer Bi Mode Register (i=0 to 5) (Event Counter Mode)



Symbol	Address	After reset
TB0MR to TB5MR	035B$_{16}$, 035C$_{16}$, 035D$_{16}$, 031B$_{16}$, 031C$_{16}$, 031D$_{16}$	00XX 0000$_2$

Bit Symbol	Bit Name	Function	RW
TMOD0	Operating Mode Select Bit	b1b0 0 1 : Event counter mode	RW
TMOD1			RW
MR0	Count Polarity Select Bit[1]	b3b2 0 0 : Counts falling edges of external signal 0 1 : Counts rising edges of external signal 1 0 : Counts falling and rising edges of external signal 1 1 : Do not set to this value	RW
MR1			RW
MR2	TB0MR and TB3MR registers: Set to "0" in event counter mode		RW
	TB1MR, TB2MR, TB4MR and TB5MR registers: Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		—
MR3	Disabled in event counter mode. When write, set to "0". When read, its content is indeterminate.		—
TCK0	Disabled in event counter mode. Can be set to "0" or "1".		RW
TCK1	Event Clock Select Bit	0 : Input signal from the TBiIN pin 1 : TBj overflows or underflows[2]	RW

NOTES:

1. MR0 and MR1 bit settings are enabled when the TCK1 bit is set to "0". The MR1 bit can be set to either "0" or "1", when the TCK1 bit is set to "1".
2. j = i-1, except j = 2 when i = 0 and j = 5 when i = 3.

Figure 15.21 TB0MR to TB5MR Registers

15.2.3 Pulse Period/Pulse Width Measurement Mode

In pulse period/pulse width measurement mode, the timer measures pulse period or pulse width of an external signal. (See **Table 15.11**) Figure 15.22 shows the TBiMR register (i=0 to 5) in pulse period/pulse width measurement mode. Figure 15.23 shows an operation example in pulse period measurement mode. Figure 15.24 shows an operation example in the pulse width measurement mode.

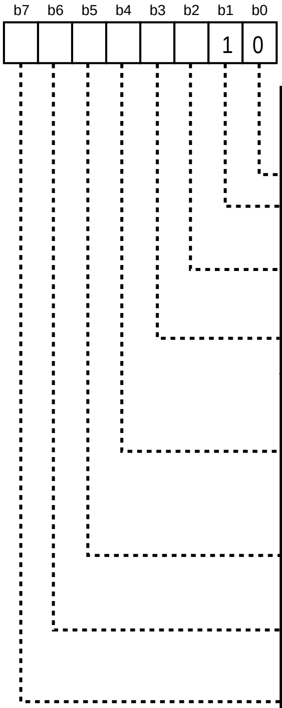
Table 15.11 Pulse Period/Pulse Width Measurement Mode Specifications

Item	Specification
Count Source	f_1, f_8, f_{2n}(3), f_{C32}
Counting Operation	• The timer increments a counter value Counter value is transferred to the reload register on the valid edge of a pulse to be measured. It is set to "0000$_{16}$" and the timer continues counting
Counter Start Condition	The TBiS bits (i=0 to 5) in the TABSR and TBSR register are set to "1" (starts counting)
Counter Stop Condition	The TBiS bit is set to "0" (stops counting)
Interrupt Request Generation Timing	• On the valid edge of a pulse to be measured(1) • The timer counter overflows The MR3 bit in the TBiMR register is set to "1" (overflow) simultaneously. When the TBiS bit is set to "1" (start counting) and the next count source is counted after setting the MR3 bit to "1" (overflow), the MR3 bit can be set to "0" (no overflow) by writing to the TBiMR register.
TBiIN Pin Function	Input for a pulse to be measured
Read from Timer	The TBi register indicates reload register values (measurement results)(2)
Write to Timer	Value written to the TBi register can be written to neither reload register nor counter

NOTES:

1. No interrupt request is generated when the pulse to be measured is on the first valid edge after the timer has started counting.
2. The TBi register is in an indeterminate state until the pulse to be measured is on the second valid edge after the timer has started counting.
3. The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2n (n=1 to 15).

Timer Bi Mode Register (i=0 to 5)
(Pulse Period / Pulse Width Measurement Mode)



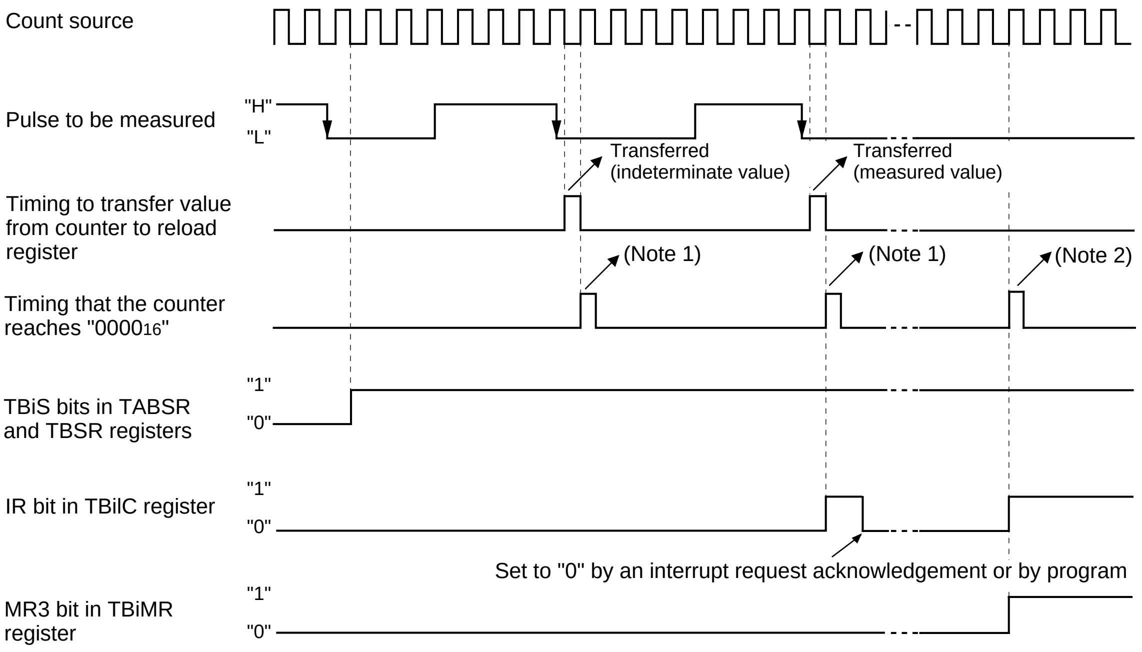
Symbol	Address	After reset
TB0MR to TB5MR	035B$_{16}$, 035C$_{16}$, 035D$_{16}$, 031B$_{16}$, 031C$_{16}$, 031D$_{16}$	00XX 0000$_2$

Bit Symbol	Bit Name	Function	RW
TMOD0	Operating Mode Select Bit	b1b0 1 0 : Pulse period measurement mode, Pulse width measurement mode	RW
TMOD1			RW
MR0	Measurement Mode Select Bit[1]	b3b2 0 0 : Pulse period measurement 1 0 1 : Pulse period measurement 2 1 0 : Pulse width measurement 1 1 : Do not set to this value	RW
MR1			RW
MR2	TB0MR, TB3MR registers: Set to "0" in pulse period/pulse width measurement mode		RW
	TB1MR, TB2MR TB4MR, TB5MR registers: Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		—
MR3	Timer Bi Overflow Flag[2]	0 : No overflow 1 : Overflow	RO
TCK0	Count Source Select Bit	b7b6 0 0 : f_1 0 1 : f_8 1 0 : f_{2n}[3] 1 1 : f_{C32}	RW
TCK1			RW

NOTES:

1. The MR1 and MR0 bits selects the following measurements.
   Pulse period measurement 1 (the MR1 and MR0 bits are set to "00$_2$") :
   Measures between the falling edge and the next falling edge of a pulse to be measured
   Pulse period measurement 2 (the MR1 and MR0 bits are set to "01$_2$") :
   Measures between the rising edge and the next rising edge of a pulse to be measured
   Pulse width measurement (the MR1 and MR0 bits are set to "10$_2$") :
   Measures between a falling edge and the next rising edge of a pulse to be measured and between the rising edge and the next falling edge of a pulse to be measured
2. The MR3 bit is indeterminate when reset.
   To set the MR3 bit to "0", se the TBiMR register after the MR3 bit is set to "1" and one or more cycles of the count source are counted, while the TBiS bits in the TABSR and TBSR registers are set to "1" (starts counting).
   The MR3 bit cannot be set to "1" by program.
3. The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2n (n=1 to 15).

Figure 15.22 TB0MR to TB5MR Registers

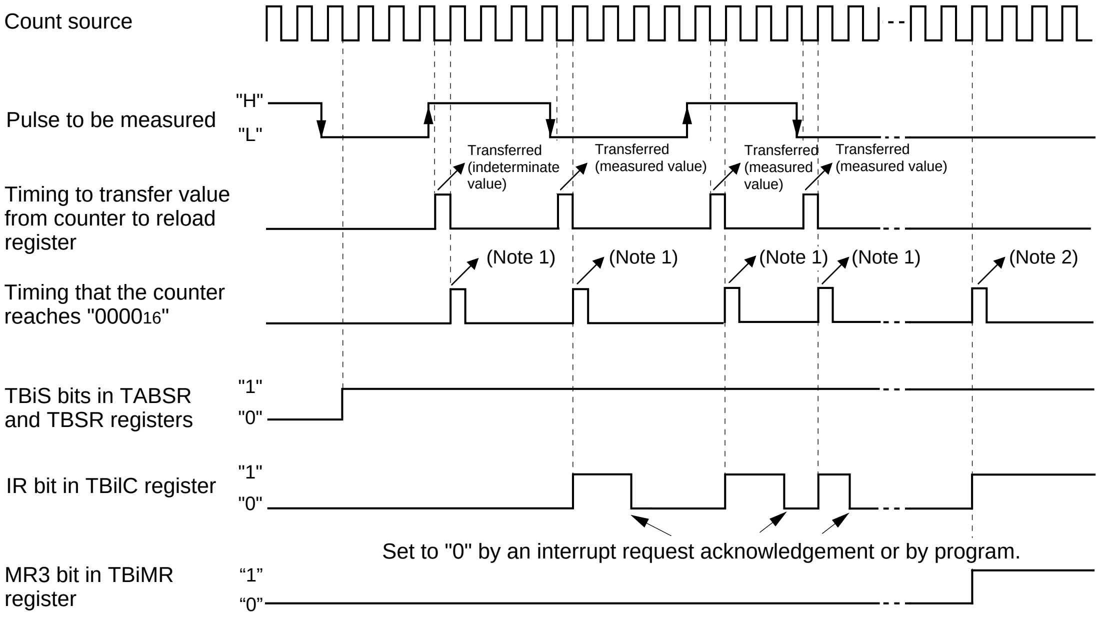

Count source
Pulse to be measured
"H"
"L"
Transferred
(indeterminate value)
Transferred
(measured value)
Timing to transfer value from counter to reload register
(Note 1)
(Note 1)
(Note 2)
Timing that the counter reaches "000016"
"1"
"0"
TBiS bits in TABSR and TBSR registers
"1"
"0"
IR bit in TBiIC register
Set to "0" by an interrupt request acknowledgement or by program
"1"
"0"
MR3 bit in TBiMR register

i=0 to 5

NOTES:

1. The counter is reset when a measurement is completed.
2. The timer counter overflows.

Figure 15.23 Operation Example in Pulse Period Measurement Mode



i=0 to 5

NOTES:

1. The counter is reset when a measurement is completed.
2. The timer counter overflows.

Figure 15.24 Operation Example in Pulse Width Measurement Mode

16. Three-Phase Motor Control Timer Functions

Three-phase motor driving waveform can be output by using the timers A1, A2, A4 and B2. Table 16.1 lists specifications of the three-phase motor control timer functions. Table 16.2 lists pin settings. Figure 16.1 shows a block diagram. Figures 16.2 to 16.7 show registers associated with the three-phase motor control timer functions.

Table 16.1 Three-Phase Motor Control Timer Functions Specification

Item	Specification
Three-Phase Waveform Output Pin	Six pins (U, $\overline{U}$, V, $\overline{V}$, W, $\overline{W}$)
Forced Cutoff[1]	Apply a low-level ("L") signal to the $\overline{NMI}$ pin
Timers to be Used	Timer A4, A1, A2 (used in one-shot timer mode): Timer A4: U- and $\overline{U}$-phase waveform control Timer A1: V- and $\overline{V}$-phase waveform control Timer A2: W- and $\overline{W}$-phase waveform control Timer B2 (used in timer mode): Carrier wave cycle control Dead time timer (three 8-bit timers share reload register): Dead time control
Output Waveform	Triangular wave modulation, Sawtooth wave modulation Can output a high-level waveform or a low-level waveform for one cycle; Can set positive-phase level and negative-phase level separately
Carrier Wave Cycle	Triangular wave modulation: *count source* x *(m+1)* x *2* Sawtooth wave modulation: *count source* x *(m+1)* *m*: setting value of the TB2 register, 0000_{16} to $FFFF_{16}$ Count source: f_1, f_8, f_{2n}[2], f_{C32}
Three-Phase PWM Output Width	Triangular wave modulation: *count source* x *n* x *2* Sawtooth wave modulation: *count source* x *n* *n*: setting value of the TA4, TA1 and TA2 register (of the TA4, TA41, TA1, TA11, TA2 and TA21 registers when setting the INV11 bit to "1"), 0001_{16} to $FFFF_{16}$ Count source: f_1, f_8, f_{2n}[2], f_{C32}
Dead Time	*Count source* x *p*, or no dead time *p*: setting value of the DTT register, 01_{16} to FF_{16} Count source: f_1, or f_1 divided by 2
Active Level	Selected from a high level ("H") or low level ("L")
Positive- and Negative-Phase Concurrent Active Disable Function	Positive and negative-phases concurrent active disable function Positive and negative-phases concurrent active detect function
Interrupt Frequency	For the timer B2 interrupt, one carrier wave cycle-to-cycle basis through 15 time- carrier wave cycle-to-cycle basis can be selected

NOTES:

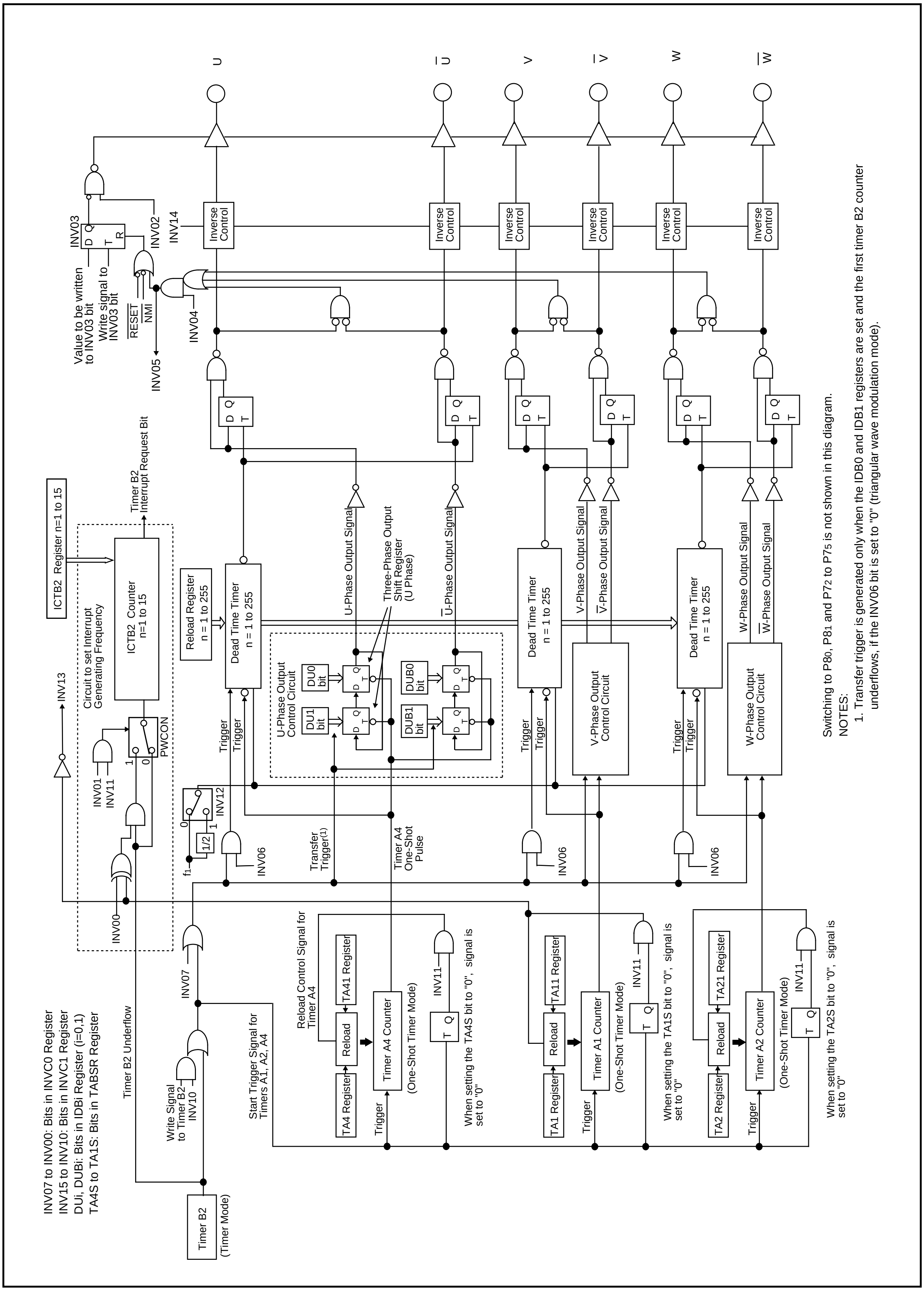
1. Forced cutoff by the signal applied to the $\overline{NMI}$ pin is available when the INV02 bit is set to "1" (three-phase motor control timer functions) and the INV03 bit is set to "1" (three-phase motor control timer output enabled).
2. The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2n (n=1 to 15).

Table 16.2 Pin Settings

Pin	Setting		
	PS1, PS2 Registers[(1)]	PSL1, PSL2 Registers	PSC Register
P7$_2$/V	PS1_2 =1	PSL1_2 =0	PSC_2 =1
P7$_3$/$\overline{V}$	PS1_3 =1	PSL1_3 =1	PSC_3 =0
P7$_4$/W	PS1_4 =1	PSL1_4 =1	PSC_4 =0
P7$_5$/$\overline{W}$	PS1_5 =1	PSL1_5 =0	——
P8$_0$/U	PS2_0 =1	PSL2_0 =1	——
P8$_1$/$\overline{U}$	PS2_1 =1	PSL2_1 =0	——

NOTES:

1. Set the PS1_5 to PS1_2 bits and PS2_1 and PS2_0 bits in the PS1 and PS2 registers to "1" after the INV02 bit is set to "1".


U
U̅
V
V̅
W
W̅
Inverse Control
INV03
D Q
T R
Value to be written to INV03 bit
Write signal to INV03 bit
RESET
NMI
INV02
INV14
INV04
INV05
Timer B2 Interrupt Request Bit
ICTB2 Register n=1 to 15
Circuit to set Interrupt Generating Frequency
ICTB2 Counter n=1 to 15
INV13
INV01
INV11
PWCON
1
0
INV00
INV12
f1
1/2
INV06
Reload Register n = 1 to 255
Dead Time Timer n = 1 to 255
Trigger
U-Phase Output Signal
U̅-Phase Output Signal
Three-Phase Output Shift Register (U Phase)
U-Phase Output Control Circuit
DU0 bit
DU1 bit
DUB0 bit
DUB1 bit
Transfer Trigger(1)
Timer A4 One-Shot Pulse
V-Phase Output Signal
V̅-Phase Output Signal
V-Phase Output Control Circuit
W-Phase Output Signal
W̅-Phase Output Signal
W-Phase Output Control Circuit
INV07
Timer B2 Underflow
Write Signal to Timer B2
INV10
Timer B2
(Timer Mode)
Start Trigger Signal for Timers A1, A2, A4
Reload Control Signal for Timer A4
TA41 Register
Reload
TA4 Register
Timer A4 Counter
(One-Shot Timer Mode)
INV11
T Q
When setting the TA4S bit to "0", signal is set to "0"
TA11 Register
TA1 Register
Timer A1 Counter
When setting the TA1S bit to "0", signal is set to "0"
TA21 Register
TA2 Register
Timer A2 Counter
When setting the TA2S bit to "0", signal is set to "0"
INV07 to INV00: Bits in INVC0 Register
INV15 to INV10: Bits in INVC1 Register
DUi, DUBi: Bits in IDBi Register (i=0,1)
TA4S to TA1S: Bits in TABSR Register

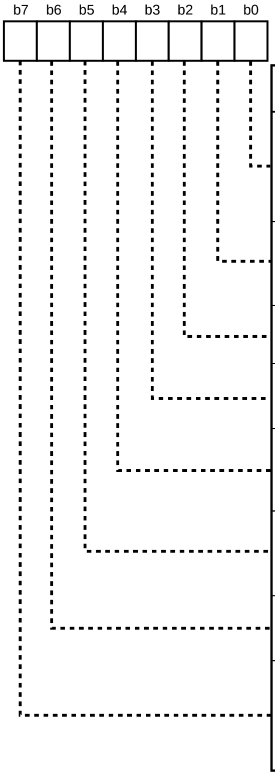
Switching to P80, P81 and P72 to P75 is not shown in this diagram.

NOTES:

1. Transfer trigger is generated only when the IDB0 and IDB1 registers are set and the first timer B2 counter underflows, if the INV06 bit is set to "0" (triangular wave modulation mode).

Figure 16.1 Three-Phase Motor Control Timer Functions Block Diagram

Three-Phase PWM Control Register 0[1]



Symbol	Address	After Reset
INVC0	0308_{16}	00_{16}

Bit Symbol	Bit Name	Function	RW
INV00	Interrupt Enable Output Polarity Select Bit[3]	0: The ICTB2 counter is incremented by one on the rising edge of the timer A1 reload control signal 1: The ICTB2 counter is incremented by one on the falling edge of the timer A1 reload control signal	RW
INV01	Interrupt Enable Output Specification Bit[2, 3]	0: ICTB2 counter is incremented by one when timer B2 counter underflows 1: Selected by the INV00 bit	RW
INV02	Mode Select Bit[4, 5, 6]	0: No three-phase control timer function 1: Three-phase control timer function	RW
INV03	Output Control Bit[6, 7]	0: Disables three-phase control timer output 1: Enables three-phase control timer output	RW
INV04	Positive and Negative-Phases Concurrent Active Disable Function Enable Bit	0: Enables concurrent active output 1: Disables concurrent active output	RW
INV05	Positive and Negative-Phases Concurrent Active Output Detect Flag[8]	0: Not detected 1: Detected	RW
INV06	Modulation Mode Select[9, 10]	0: Triangular wave modulation mode 1: Sawtooth wave modulation mode	RW
INV07	Software Trigger Select	Transfer trigger is generated when the INV07 bit is set to "1". Trigger to the dead time timer is also generated when setting the INV06 bit to "1". Its value is "0" when read.	RW

NOTES:

1. Set the INVC0 register after the PRC1 bit in the PRCR register is set to "1" (write enable). Rewrite the INV02 to INV00 and INV06 bits when the timers A1,A2, A4 and B2 stop.
2. Set the INV01 bit to "1" after setting the ICTB2 register.
3. The INV01 and INV00 bit settings are enabled only when the INV11 bit in the INVC1 register is set to "1" (three-phase mode 1). The ICTB2 counter is incremented by one every time the timer B2 counter underflows, regardless of INV01 and INV00bit settings, when the INV11 bit is set to "0" (three-phase mode). When setting the INV01 bit to "1", set the timer A1 count start flag before the first timer B2 counter underflows. When the INV00 bit is set to "1", the first interrupt is generated when the timer B2 counter underflows *n-1* times, if *n* is the value set in the ICTB2 counter. Subsequent interrupts are generated every *n* times the timer B2 counter underflows.
4. Set the INV02 bit to "1" to operate the dead time timer, U-, V-and W-phase output control circuits and ICTB2 counter.
5. Set pins after the INV02 bit is set to "1". See Table 16.2 for pin settings.
6. When the INV02 bit is set to "1" and the INV03 bit to "0", the U, $\overline{U}$, V, $\overline{V}$, W and $\overline{W}$ pins, including pins shared with other output functions, are all placed in high-impedance states.
7. The INV03 bit is set to "0" when the followings occurs :
   - Reset
   - A concurrent active state occurs while the INV04 bit is set to "1"
   - The INV03 bit is set to "0" by program
   - An "H" signal applied to the $\overline{\text{NMI}}$ pin changes to an "L" signal
8. The INV05 bit can not be set to "1" by program. Set the INV04 bit to "0", as well, when setting the INV05 bit to "0".
9. The following table describes how the INV06 bit setting works.

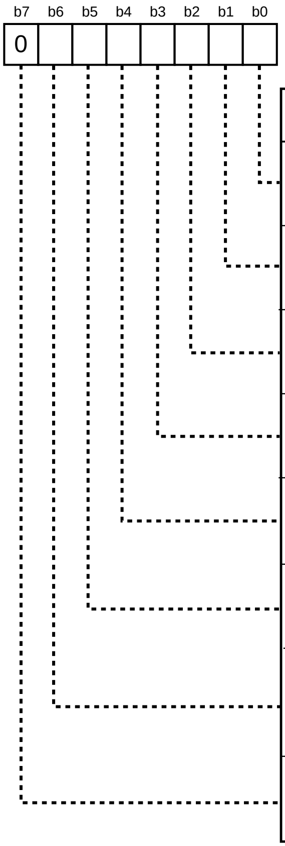
Item	INV06 = 0	INV06 = 1
Mode	Triangular wave modulation mode	Sawtooth wave modulation mode
Timing to Transfer from the IDB0 and IDB1 Registers to Three-Phase Output Shift Register	Transferred once by generating a transfer trigger after setting the IDB0 and IDB1 registers	Transferred every time a transfer trigger is generated
Timing to Trigger the Dead Time Timer when the INV16 Bit=0	On the falling edge of a one-shot pulse of the timer A1, A2 or A4	By a transfer trigger, or the falling edge of a one-shot pulse of the timer A1, A2 or A4
INV13 Bit	Enabled when the INV11 bit=1 and the INV06 bit=0	Disabled

Transfer trigger : Timer B2 counter underflows and write to the INV07 bit, or write to the TB2 register when INV10 = 1

10. When the INV06 bit is set to "1", set the INV11 bit to "0" (three-phase mode 0) and the PWCON bit in the TB2SC register to "0" (timer B2 counter underflows).

Figure 16.2 INVC0 Register

Three-Phase PWM Control Register 1[1]



Symbol	Address	After Reset
INVC1	0309_{16}	00_{16}

Bit Symbol	Bit Name	Function	RW
INV10	Timer A1, A2 and A4 Start Trigger Select Bit	0: Timer B2 counter underflows 1: Timer B2 counter underflows and write to the TB2 register	RW
INV11	Timer A1-1, A2-1 and A4-1 Control Bit[2, 3]	0: Three-phase mode 0 1: Three-phase mode 1	RW
INV12	Dead Time Timer Count Source Select Bit	0 : f_1 1 : f_1 divided-by-2	RW
INV13	Carrier Wave Detect Flag[4]	0: Timer A1 reload control signal is "0" 1: Timer A1 reload control signal is "1"	RO
INV14	Output Polarity Control Bit	0 : Active "L" of an output waveform 1 : Active "H" of an output waveform	RW
INV15	Dead Time Disable Bit	0: Enables dead time 1: Disables dead time	RW
INV16	Dead Time Timer Trigger Select Bit	0: Falling edge of a one-shot pulse of the timer A1, A2 and A4[5] 1: Rising edge of the three-phase output shift register (U-, V-, W-phase)	RW
—— (b7)	Reserved Bit	Set to "0"	RW

NOTES:

1. Rewrite the INVC1 register after the PRC1 bit in the PRCR register is set to "1" (write enable). The timers A1, A2, A4, and B2 must be stopped during rewrite.
2. The following table lists how the INV11 bit setting works.

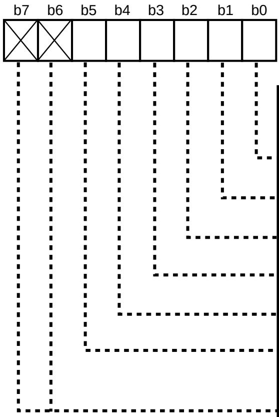
Item	INV11 = 0	INV11 = 1
Mode	Three-phase mode 0	Three-phase mode 1
TA11, TA21 and TA41 Registers	Not used	Used
INV01 and INV00 Bit in the INVC0 Register	Disabled. The ICTB2 counter is incremented whenever the timer B2 counter underflows	Enabled
INV13 Bit	Disabled	Enabled when INV11=1 and INV06=0

3. When the INV06 bit in the INVC0 registser is set to "1" (sawtooth wave modulation mode), set the INV11 bit to "0". Also, when the INV11 bit is set to "0", set the PWCON bit in the TB2SC register to "0" (Timer B2 counter underflows).
4. The INV13 bit setting is enabled only when the INV06 bit is set to "0" (Triangular wave modulation mode) and the INV11 bit to "1".
5. If the following conditions are all met, set the INV16 bit to "1".
   - The INV15 bit is set to "0"
   - The Dij bit (i=U, V or W, j=0, 1) and DiBj bit always have different values when the INV03 bit in the INVC0 register is set to "1". (The positive-phase and negative-phase outputs always provide opposite level signals.)

   If the above conditions are not met, set the INV16 bit to "0".

Figure 16.3 INVC1 Register

Three-Phase Output Buffer Register i[1] (i=0, 1)



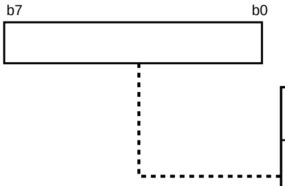
Symbol	Address	After Reset
IDB0, IDB1	$030A_{16}$, $030B_{16}$	XX11 1111_2

Bit Symbol	Bit Name	Function	RW
DUi	U-Phase Output Buffer i	Write output level 0: Active level 1: Inactive level When read, the value of the three-phase shift register is read.	RW
DUBi	$\overline{U}$-Phase Output Buffer i		RW
DVi	V-Phase Output Buffer i		RW
DVBi	$\overline{V}$-Phase Output Buffer i		RW
DWi	W-Phase Output Buffer i		RW
DWBi	$\overline{W}$-Phase Output Buffer i		RW
—— (b7 - b6)	Reserved Bit	When read, its content is indeterminate	RO

NOTES:

1. Values of the IDB0 and IDB1 registers are transferred to the three-phase output shift register by a transfer trigger.
   After the transfer trigger occurs, the values written in the IDB0 register determine each phase output signal level first. Then the value written in the IDB1 register on the falling edge of the timers A1, A2 and A4 one-shot pulse determines each phase output signal level.

Dead Time Timer[1, 2]



Symbol	Address	After Reset
DTT	$030C_{16}$	Indeterminate

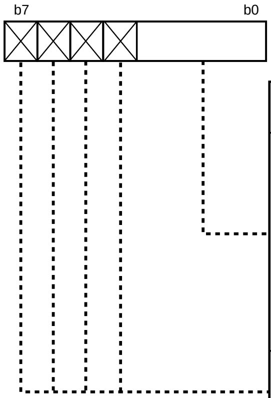
Function	Setting Range	RW
If setting value is *n*, the timer stops when counting *n* times a count source selected by the INV12 bit after start trigger occurs. Positive or negative phase, which changes from inactive level to active level, shifts when the dead time timer stops.	1 to 255	WO

NOTES:

1. Use the MOV instruction to set the DTT register.
2. The DTT register setting is enabled when the INV15 bit in the INVC1 register is set to "0" (dead time enabled). No dead time can be set when the INV15 bit is set to "1" (dead time disabled). The INV06 bit in the INVC0 register determines start trigger of the DTT register.

Figure 16.4 IDB0 and IDB1 registers, DTT Register

Timer B2 Interrupt Generation Frequency Set Counter[1, 2, 3]

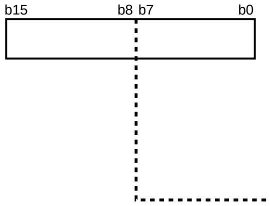


Symbol	Address	After Reset
ICTB2	$030D_{16}$	Indeterminate

Function	Setting Range	RW
When the INV01 bit is set to "0" (the ICTB2 counter increments whenever the timer B2 counter underflows) and the setting value is *n*, the timer B2 interrupt is generated every *n*th time timer B2 counter underflow occurs. When the INV01 bit is set to "1" (the INV00 bit selects count timing of the ICTB2 counter) and setting value is *n*, the timer B2 interrupt is generated every *n*th time timer B2 counter underflow meeting the condition selected in the INV00 bit occurs.	1 to 15	WO
Nothing is assigned. When write, set to "0".		—

NOTES:
1. Use the MOV instruction to set the ICTB2 register.
2. If the INV01 bit in the INVC0 register is set to "1", set the ICTB2 register in the TABSR register when the TB2S bit is set to "0" (timer B2 counter stopped).
   If the INV01 bit is set to "0" and the TB2S bit to "1" (timer B2 counter start), do not set the ICTB2 register when the timer B2 counter underflows.
3. If the INV00 bit in the INVC0 register is set to "1", the first interrupt is generated when the timer B2 counter underflows *n-1* times, *n* being the value set in the ICTB2 counter. Subsequent interrupts are generated every *n* times the timer B2 counter underflows.

Timer Ai, Ai-1 Register (i=1, 2, 4)[1, 2, 3, 4, 5, 6]

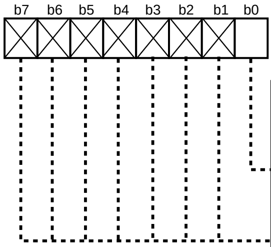


Symbol	Address	After Reset
TA1, TA2, TA4	0349_{16} - 0348_{16}, $034B_{16}$ - $034A_{16}$, $034F_{16}$ - $034E_{16}$	Indeterminate
TA11, TA21, TA41	0303_{16} - 0302_{16}, 0305_{16} - 0304_{16}, 0307_{16} - 0306_{16}	Indeterminate

Function	Setting Range	RW
If setting value is *n*, the timer stops when the *n*th count source is counted after a start trigger is generated. Positive phase changes to negative phase, and vice versa, when the timers A1, A2 and A4 stop.	0000_{16} to $FFFF_{16}$	WO

NOTES:
1. Use a 16-bit data for read and write.
2. If the TAi or TAi1 register is set to "0000_{16}", no counter starts and no timer Ai interrupt is generated.
3. Use the MOV instruction to set the TAi and TAi1 registers.
4. When the INV15 bit in the INVC1 register is set to "0" (dead timer enabled), phase switches from an inactive level to an active level when the dead time timer stops.
5. When the INV11 bit in the INVC1 register is set to "0" (three-phase mode 0), the value of the TAi register is transferred to the reload register by a timer Ai start trigger.
   When the INV11 bit is set to "1" (three-phase mode 1), the value of the TAi1 register is first transferred to the reload register by a timer Ai start trigger. Then, the value of the TAi register is transferred by the next trigger. The values of the TAi1 and TAi registers are transferred alternately to the reload register with every timer Ai start trigger.
6. Do not write to these registers when the timer B2 counter underflows.

Timer B2 Special Mode Register



Symbol	Address	After Reset
TB2SC	$035E_{16}$	XXXX $XXX0_2$

Bit Symbol	Bit Name	Function	RW
PWCON	Timer B2 Reload Timing Switching Bit[1]	0 : Timer B2 counter underflows 1 : Timer A output in odd-number times	RW
Nothing is assigned. When write, set to "0". When read, its content is "0."			—

NOTES:
1. Set the PWCON bit to "0" when setting the INV11 bit to "0" (three-phase mode 0) or the INV06 bit to "1" (sawtooth wave modulation mode).

Figure 16.5 ICTB2 Register, TA1, TA2, TA4, TA11, TA21 and TA41 Registers, TB2SC Register

Timer B2 Register[1]

b15 b8 b7 b0

Symbol	Address	After Reset
TB2	0355_{16} - 0354_{16}	Indeterminate

Function	Setting Range	RW
If setting value is *n*, count source is divided by *n*+1. The timers A1, A2 and A4 start every time an underflow occurs.	0000_{16} to $FFFF_{16}$	RW

NOTES:
1. Use a 16-bit data for read and write.

Trigger Select Register

b7 b6 b5 b4 b3 b2 b1 b0

Symbol	Address	After Reset
TRGSR	0343_{16}	00_{16}

Bit Symbol	Bit Name	Function	RW
TA1TGL	Timer A1 Event/Trigger Select Bit	Set to "01_2" (TB2 underflow) before using a V-phase output control circuit	RW
TA1TGH			RW
TA2TGL	Timer A2 Event/Trigger Select Bit	Set to "01_2" (TB2 underflow) before using a W-phase output control circuit	RW
TA2TGH			RW
TA3TGL	Timer A3 Event/Trigger Select Bit	b5 b4 0 0 : Selects an input to the TA3IN pin 0 1 : Selects TB2 overflow[1] 1 0 : Selects TA2 overflow[1] 1 1 : Selects TA4 overflow[1]	RW
TA3TGH			RW
TA4TGL	Timer A4 Event/Trigger Select Bit	Set to "01_2" (TB2 underflow) before using a U-phase output control circuit	RW
TA4TGH			RW

NOTES:
1. Overflow or underflow

Count Start Flag

b7 b6 b5 b4 b3 b2 b1 b0

Symbol	Address	After Reset
TABSR	0340_{16}	00_{16}

Bit Symbol	Bit Name	Function	RW
TA0S	Timer A0 Count Start Flag	0 : Stops counting 1 : Starts counting	RW
TA1S	Timer A1 Count Start Flag	0 : Stops counting 1 : Starts counting	RW
TA2S	Timer A2 Count Start Flag	0 : Stops counting 1 : Starts counting	RW
TA3S	Timer A3 Count Start Flag	0 : Stops counting 1 : Starts counting	RW
TA4S	Timer A4 Count Start Flag	0 : Stops counting 1 : Starts counting	RW
TB0S	Timer B0 Count Start Flag	0 : Stops counting 1 : Starts counting	RW
TB1S	Timer B1 Count Start Flag	0 : Stops counting 1 : Starts counting	RW
TB2S	Timer B2 Count Start Flag	0 : Stops counting 1 : Starts counting	RW

Figure 16.6 TB2, TRGSR and TABSR Registers

Timer Ai Mode Register (i=1, 2, 4)

b7	b6	b5	b4	b3	b2	b1	b0
		0	1	0	0	1	0

Symbol	Address	After Reset
TA1MR, TA2MR, TA4MR	0357_{16}, 0358_{16}, $035A_{16}$	00_{16}

Bit Symbol	Bit Name	Function	RW
TMOD0	Operating Mode Select Bit	Set to "10_2" (one-shot timer mode) when using the three-phase motor control timer function	RW
TMOD1			
MR0	Reserved Bit	Set to "0"	RW
MR1	External Trigger Select Bit	Set to "0" when using the three-phase motor control timer function	RW
MR2	Trigger Select Bit	Set to "1" (selected by the TRGSR register) when using the three-phase motor control timer function	RW
MR3	Set to "0" with the three-phase motor control timer function		RW
TCK0	Count Source Select Bit	b7 b6 0 0 : f_1 0 1 : f_8 1 0 : f_{2n}(1) 1 1 : f_{C32}	RW
TCK1			RW

NOTES:
1. The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2n (n=1 to 15).

Timer B2 Mode Register

b7	b6	b5	b4	b3	b2	b1	b0
		☒	0	0	0	0	0

Symbol	Address	After Reset
TB2MR	$035D_{16}$	$00XX\ 0000_2$

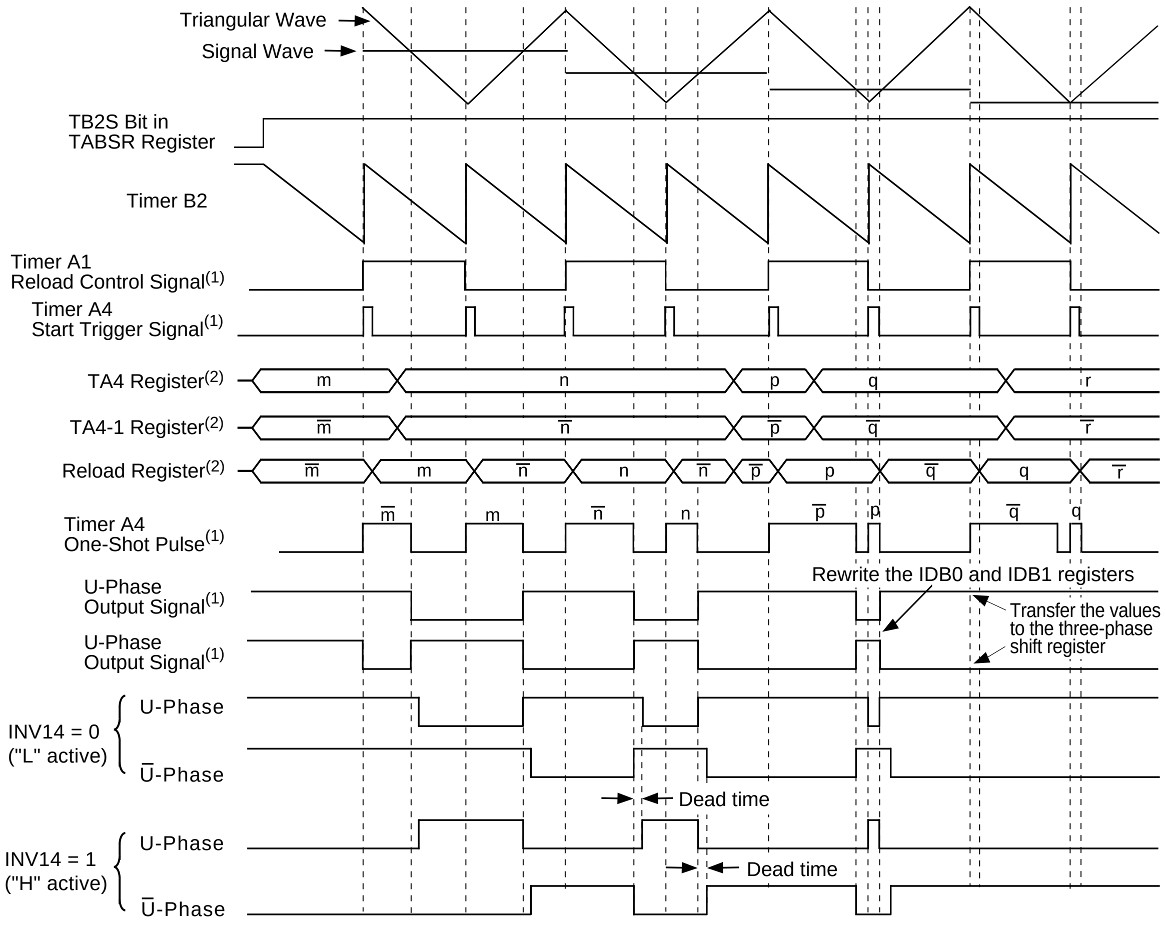
Bit Symbol	Bit Name	Function	RW
TMOD0	Operating Mode Select Bit	Set to "00_2" (timer mode) when using the three-phase motor control timer function	RW
TMOD1			
MR0	Disabled when using the three-phase motor control timer function. When write, set to "0". When read, its content is indeterminate.		—
MR1			
MR2	Set to "0" when using three-phase motor control timer function		RW
MR3	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		RW
TCK0	Count Source Select Bit	b7 b6 0 0 : f_1 0 1 : f_8 1 0 : f_{2n}(1) 1 1 : f_{C32}	RW
TCK1			RW

NOTES:
1. The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2n (n=1 to 15).

Figure 16.7 TA1MR, TA2MR and TA4MR Registers, TB2MR Register

The three-phase motor control timer function is available by setting the INV02 bit in the INVC0 register to "1". The timer B2 is used for carrier wave control and the timers A1, A2, A4 for three-phase PWM output (U, $\overline{U}$, V, $\overline{V}$, W, $\overline{W}$) control. An exclusive dead time timer controls dead time. Figure 16.8 shows an example of the triangular modulation waveform. Figure 16.9 shows an example of the sawtooth modulation waveform.

Triangular waveform as a Carrier Wave



INV00, INV01: Bits in INVC0 register
INV11, INV14: Bits in INVC1 register

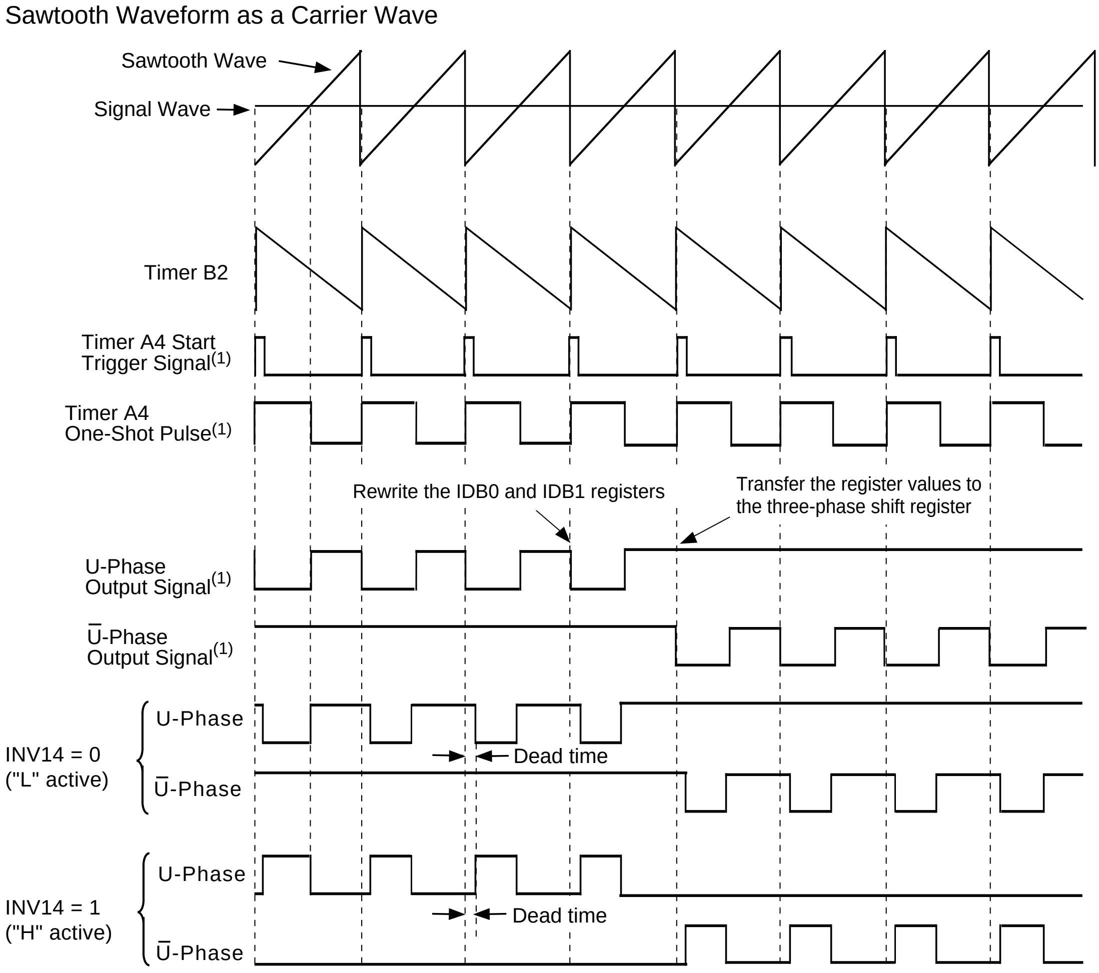
NOTES:
1. Internal signals. See Figure 16.1.
2. Applies only when the INV11 bit is set to "1" (three-phase mode).

The above applies to INVC0 = 00XX11XX$_2$ and INVC1 = 010XXXX0$_2$ (X varies depending on each system.)
Examples of PWM output change are

(a) When INV11=1 (three-phase mode 1)
- INV01=0 and ICTB2=2$_{16}$ (The timer B2 interrupt is generated with every second timer B2 underflow) or INV01=1, INV00=1and ICTB2=1$_{16}$ (The timer B2 interrupt is generated on the falling edge of the timer A reload control signal)
- Default value of the timer: TA41=$\overline{m}$, TA4=m
  The TA4 and TA41 registers are changed whenever the timer B2 interrupt is generated.
  First time: TA41=$\overline{n}$, TA4:=n.
  Second time: TA41=$\overline{p}$, TA4=p.
- Default value of the IDB0 and IDB1 registers
  DU0=1, DUB0=0, DU1=0, DUB1=1
  They are changed to DU0=1, DUB0=0, DU1=1, DUB1=0 by the third timer B2 interrupt.

(b) When INV11=0 (three-phase mode 0)
- INV01=0, ICTB2=1$_{16}$ (The timer B2 interrupt is generated whenever the timer B2 underflows)
- Default value of the timer: TA4=$\overline{m}$
  The TA4 register is changed whenever the timer B2 interrupt is generated.
  First time: TA4=m. Second time: TA4=$\overline{n}$.
  Third time: TA4=n. Fourth time: TA=$\overline{p}$.
  Fifth time: TA4=p.
- Default value of the IDB0 and IDB1 registers:
  DU0=1, DUB0=0, DU1=0, DUB1=1
  They are changed to DU0=1, DUB0=0, DU1=1, DUB1=0 by the sixth timer B2 interrupt.

Figure 16.8 Triangular Wave Modulation Operation


Sawtooth Waveform as a Carrier Wave
Sawtooth Wave
Signal Wave
Timer B2
Timer A4 Start Trigger Signal(1)
Timer A4 One-Shot Pulse(1)
Rewrite the IDB0 and IDB1 registers
Transfer the register values to the three-phase shift register
U-Phase Output Signal(1)
Ū-Phase Output Signal(1)
INV14 = 0 ("L" active)
U-Phase
Ū-Phase
Dead time
INV14 = 1 ("H" active)
U-Phase
Ū-Phase
Dead time

INV14: Bits in INVC1 register

NOTES:

1. Internal signals. See Figure 16.1.

The above applies to INVC0 = $01XX110X_2$ and INVC1 = $000XXX00_2$ (X varies depending on each system.)
The examples of PWM output change are
- Default value of the IDB0 and IDB1 registers: DU0=0, DUB0=1, DU1=1, DUB1=1
  They are changed to DU0=1, DUB0=0, DU1=1, DUB1=1 by the timer B2 interrupt.

Figure 16.9 Sawtooth Wave Modulation Operation

17. Serial I/O

Serial I/O consists of five channels (UART0 to UART4).

Each UARTi (i=0 to 4) has an exclusive timer to generate the transfer clock and operates independently.

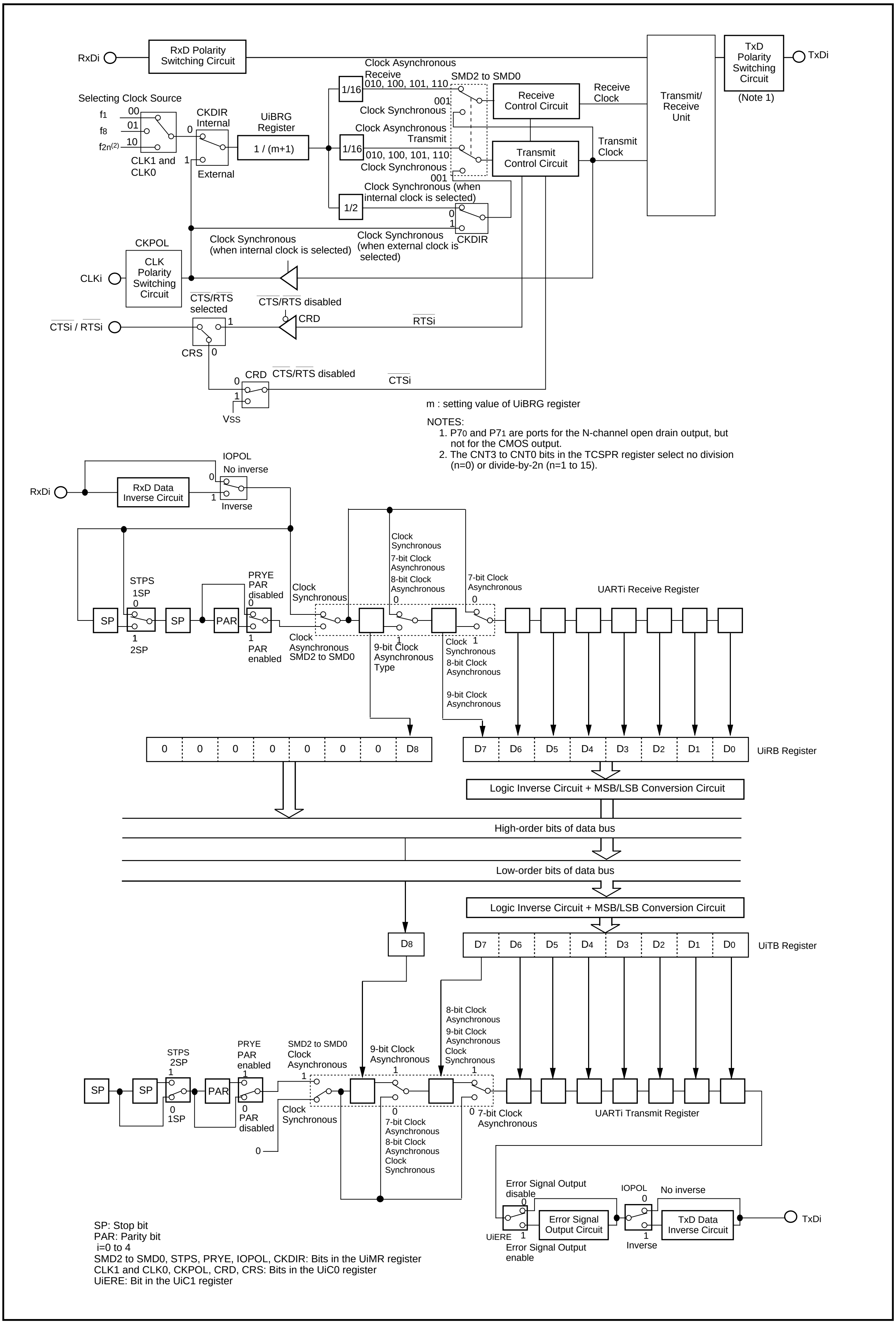
Figure 17.1 shows a UARTi block diagram.

UARTi supports the following modes :

- Clock synchronous serial I/O mode
- Clock asynchronous serial I/O mode (UART mode)
- Special mode 1 (I^2C mode)
- Special mode 2
- Special mode 3 (Clock-divided synchronous function, GCI mode)
- Special mode 4 (Bus conflict detect function, IE mode)
- Special mode 5 (SIM mode)

Figures 17.2 to 17.9 show registers associated with UARTi.

Refer to the tables listing each mode for register and pin settings.



m : setting value of UiBRG register

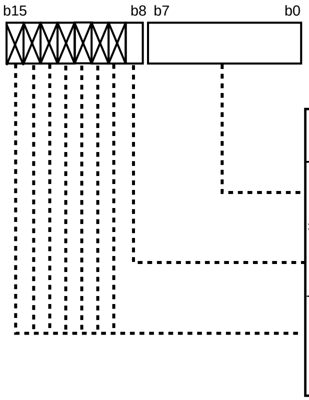
NOTES:

1. P70 and P71 are ports for the N-channel open drain output, but not for the CMOS output.
2. The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2n (n=1 to 15).

SP: Stop bit
PAR: Parity bit
i=0 to 4
SMD2 to SMD0, STPS, PRYE, IOPOL, CKDIR: Bits in the UiMR register
CLK1 and CLK0, CKPOL, CRD, CRS: Bits in the UiC0 register
UiERE: Bit in the UiC1 register

Figure 17.1 UARTi Block Diagram

UARTi Transmit Buffer Register (i=0 to 4)[1]

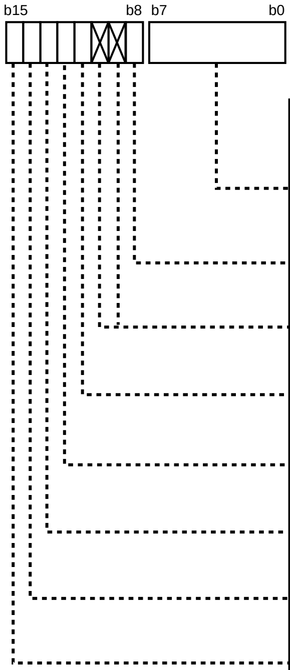


Symbol	Address	After Reset
U0TB to U2TB	$036B_{16}$-$036A_{16}$, $02EB_{16}$-$02EA_{16}$, $033B_{16}$-$033A_{16}$	Indeterminate
U3TB, U4TB	$032B_{16}$-$032A_{16}$, $02FB_{16}$-$02FA_{16}$	Indeterminate

Bit Symbol	Function	RW
—— (b7 - b0)	Transmit data (D_7 to D_0)	WO
—— (b8)	Transmit data (D_8)	WO
—— (b15 - b9)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.	—

NOTES:
1. Use the MOV instruction to set the UiTB register.

UARTi Receive Buffer Register (i=0 to 4)



Symbol	Address	After Reset
U0RB to U2RB	$036F_{16}$ - $036E_{16}$, $02EF_{16}$ - $02EE_{16}$, $033F_{16}$ - $033E_{16}$	Indeterminate
U3RB, U4RB	$032F_{16}$ - $032E_{16}$, $02FF_{16}$ - $02FE_{16}$	Indeterminate

Bit Symbol	Bit Name	Function	RW
—— (b7 - b0)	——	Received data (D_7 to D_0)	RO
—— (b8)	——	Received data (D_8)	RO
—— (b10 - b9)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		—
ABT	Arbitration Lost Detect Flag[1]	0: Not detected (win) 1: Detected (lose)	RW
OER	Overrun Error Flag[2]	0: No overrun error occurs 1: Overrun error occurs	RO
FER	Framing Error Flag[2, 3]	0: No framing error occurs 1: Framing error occurs	RO
PER	Parity Error Flag[2, 3]	0: No parity error occurs 1: Parity error occurs	RO
SUM	Error Sum Flag[2, 3]	0: No error occurs 1: Error occurs	RO

NOTES:
1. The ABT bit can be set to "0" only.
2. When the SMD2 to SMD0 bits in the UiMR register are set to "000_2" (serial I/O disable) or the RE bit in the UiC1 register is set to "0" (receive disable), the OER, FER, PER and SUM bits are set to "0". When all OER, FER and PER bits are set to "0", the SUM bit is set to "0". Also, the FER and PER bits are set to "0" by reading low-order bits in the UiRB register.
3. These error flags are disabled when the SMD2 to SMD0 bits are set to "001_2" (clock synchronous serial I/O mode) or to "010_2" (I^2C mode). When read, the contents are indeterminate.

Figure 17.2 U0TB to U4TB Registers and U0RB to U4RB Registers

UARTi Bit Rate Register (i=0 to 4)(1, 2)

b7 b0

Symbol	Address	After Reset
U0BRG to U4BRG	0369_{16}, $02E9_{16}$, 0339_{16}, 0329_{16}, $02F9_{16}$	Indeterminate

Function	Setting Range	RW
If the setting value is *m*, the UiBRG register divides a count source by *m+1*	00_{16} to FF_{16}	WO

NOTES:

1. Use the MOV instruction to set the UiBRG register.
2. Set the UiBRG register while no data transfer occurs.

UARTi Transmit/Receive Mode Register (i=0 to 4)

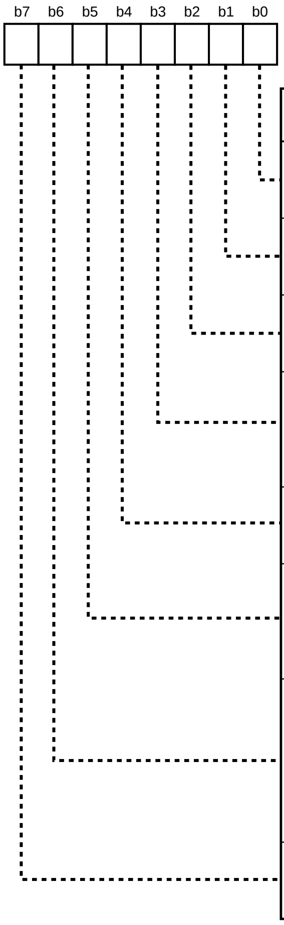
b7 b6 b5 b4 b3 b2 b1 b0

Symbol	Address	After Reset
U0MR to U4MR	0368_{16}, $02E8_{16}$, 0338_{16}, 0328_{16}, $02F8_{16}$	00_{16}

Bit Symbol	Bit Name	Function	RW
SMD0	Serial I/O Mode Select Bit	b2 b1 b0 0 0 0: Serial I/O disabled 0 0 1: Clock synchronous serial I/O mode 0 1 0: I^2C mode 1 0 0: UART mode, 7-bit transfer data 1 0 1: UART mode, 8-bit transfer data 1 1 0: UART mode, 9-bit transfer data Do not set value other than the above	RW
SMD1			RW
SMD2			RW
CKDIR	Internal/External Clock Select Bit	0 : Internal clock 1 : External clock	RW
STPS	Stop Bit Length Select Bit	0 : 1 stop bit 1 : 2 stop bits	RW
PRY	Odd/Even Parity Select Bit	Enables when PRYE = 1 0 : Odd parity 1 : Even parity	RW
PRYE	Parity Enable Bit	0 : Disables a parity 1 : Enables a parity	RW
IOPOL	TxD,RxD Input/Output Polarity Switch Bit	0: Not inversed 1: Inverse	RW

Figure 17.3 U0BRG to U4BRG Registers and U0MR to U4MR Registers

UARTi Transmit/Receive Control Register 0 (i=0 to 4)



Symbol	Address	After Reset
U0C0 to U4C0	$036C_{16}$, $02EC_{16}$, $033C_{16}$, $032C_{16}$, $02FC_{16}$	$0000\ 1000_2$

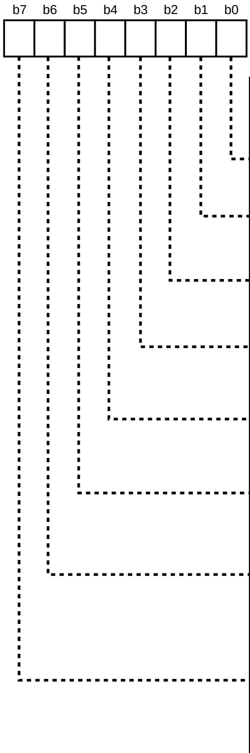
Bit Symbol	Bit Name	Function	RW
CLK0	UiBRG Count Source Select Bit	b1 b0 0 0: Selects f_1 0 1: Selects f_8 1 0: Selects f_{2n}(2) 1 1: Do not set to this value	RW
CLK1			RW
CRS	$\overline{\text{CST}}$/$\overline{\text{RTS}}$ Function Select Bit	Enabled when CRD=0 0 : Selects $\overline{\text{CTS}}$ function 1 : Selects $\overline{\text{RTS}}$ function	RW
TXEPT	Transmit Register Empty Flag	0 : Data in the transmit register (during transmission) 1 : No data in the transmit register (transmission is completed)	RO
CRD	$\overline{\text{CTS}}$/$\overline{\text{RTS}}$ Disable Bit	0 : Enables $\overline{\text{CTS}}$/$\overline{\text{RTS}}$ function 1 : Disables $\overline{\text{CTS}}$/$\overline{\text{RTS}}$ function	RW
NCH	Data Output Select Bit(1)	0 : TxDi/SDAi and SCLi are ports for the CMOS output 1 : TxDi/SDAi and SCLi are ports for the N-channel open drain output	RW
CKPOL	CLK Polarity Select Bit	0 : Data is transmitted on the falling edge of the transfer clock and data is received on the rising edge 1 : Data is transmitted on the rising edge of the transfer clock and data is received on the falling edge	RW
UFORM	Transfer Format Select Bit(3)	0 : LSB first 1 : MSB first	RW

NOTES:

1. $P7_0$/TxD2 and $P7_1$/SCL2 are ports for the N-channel open drain output, but not for the CMOS output.
2. The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2n (n=1 to 15).
3. The UFORM bit setting is enabled when the SMD2 to SMD0 bits in the UiMR register are set to "001_2" (clock syncronous serial I/O mode) or "101_2" (UART mode, 8-bit transfer data).
   Set the UFORM bit to "1" when setting the SMD2 to SMD0 bits to"010_2" (I^2C mode), or to "0" when setting them to "100_2" (UART mode, 7-bit transfer data) or "110_2" (UART mode, 9-bit transfer data).

Figure 17.4 U0C0 to U4C0 Registers

UARTi Transmit/Receive Control Register 1 (i=0 to 4)



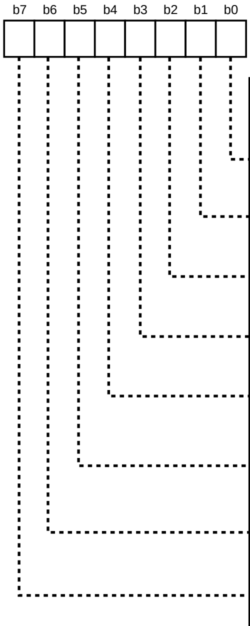
Symbol	Address	After Reset
U0C1 to U4C1	$036D_{16}$, $02ED_{16}$, $033D_{16}$, $032D_{16}$, $02FD_{16}$	$0000\ 0010_2$

Bit Symbol	Bit Name	Function	RW
TE	Transmit Enable Bit	0: Transmit disable 1: Transmit enable	RW
TI	Transmit Buffer Empty Flag	0: Data in the UiTB register 1: No data in the UiTB register	RO
RE	Receive Enable Bit	0: Receive disable 1: Receive enable	RW
RI	Receive Complete Flag	0: No data in the UiRB register 1: Data in the UiRB register	RO
UiIRS	UARTi Transmit Interrupt Cause Select Bit	0: No data in the UiTB register (TI = 1) 1: Transmission is completed (TXEPT = 1)	RW
UiRRM	UARTi Continuous Receive Mode Enable Bit	0: Disables continuous receive mode to be entered 1: Enables continuous receive mode to be entered	RW
UiLCH	Data Logic Select Bit(2)	0: Not inversed 1: Inverse	RW
SCLKSTPB /UiERE	Clock-Divided Synchronous Stop Bit / Error Signal Output Enable Bit(1)	Clock-divided synchronous stop bit (special mode 3) 0: Stops synchronizing 1: Starts synchronizing Error signal output enable bit (special mode 5) 0: Not output 1: Output	RW

NOTES:

1. Set the SCLKSTPB/UiERE bit after setting the SMD2 to SMD0 bits in the UiMR register.
2. The UiLCH bit setting is enabled when setting the SMD2 to SMD0 bits to "001_2" (clock syncronous serial I/O mode), "100_2" (UART mode, 7-bit transfer data) or "101_2" (UART mode, 8-bit transfer data). Set the UiLCH bit to "0" when setting the SMD2 to SMD0 bits to"010_2" (I^2C mode) or "110_2" (UART mode, 9-bit transfer data).

UARTi Special Mode Register (i=0 to 4)



Symbol	Address	After Reset
U0SMR to U4SMR	0367_{16}, $02E7_{16}$, 0337_{16}, 0327_{16}, $02F7_{16}$	00_{16}

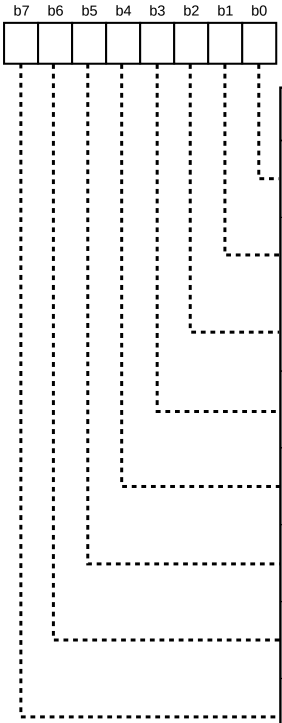
Bit Symbol	Bit Name	Function	RW
IICM	I^2C Mode Select Bit	0: Except I^2C mode 1: I^2C mode	RW
ABC	Arbitration Lost Detect Flag Control Bit	0: Update per bit 1: Update per byte	RW
BBS	Bus Busy Flag	0: Stop condition detected 1: Start condition detected (Busy)	RW(1)
LSYN	SCLL Sync Output Enable Bit	0: Disabled 1: Enabled	RW
ABSCS	Bus Conflict Detect Sampling Clock Select Bit	0: Rising edge of transfer clock 1: Timer Aj underflow(j=0 to 4)(2)	RW
ACSE	Auto Clear Function Select Bit for Transmit Enable Bit	0: No auto clear function 1: Auto clear at bus conflict	RW
SSS	Transmit Start Condition Select Bit	0: Not related to RxDi 1: Synchronized with RxDi	RW
SCLKDIV	Clock Divide Synchronous Bit	(Note 3)	RW

NOTES:

1. The BBS bit is set to "0" by program. It is unchanged if set to "1".
2. UART0: timer A3 underflow signal, UART1: timer A4 underflow signal, UART2: timer A0 underflow signal, UART3: timer A3 underflow signal, UART4: timer A4 underflow signal.
3. Refer to notes for the SU1HIM bit in the UiSMR2 register.

Figure 17.5 U0C1 to U4C1 Registers and U0SMR to U4SMR Registers

UARTi Special Mode Register 2 (i=0 to 4)



Symbol	Address	After Reset
U0SMR2 to U4SMR2	0366_{16}, $02E6_{16}$, 0336_{16}, 0326_{16}, $02F6_{16}$	00_{16}

Bit Symbol	Bit Name	Function	RW
IICM2	I^2C Mode Select Bit 2	(Note 1)	RW
CSC	Clock Synchronous Bit	0: Disabled 1: Enabled	RW
SWC	SCL Wait Output Bit	0: Disabled 1: Enabled	RW
ALS	SDA Output Stop Bit	0: Output 1: No output	RW
STC	UARTi Initialize Bit	0: Disabled 1: Enabled	RW
SWC2	SCL Wait Output Bit 2	0: Transfer clock 1: "L" output	RW
SDHI	SDA Output Inhibit Bit	0: Output 1: No output (high-impedance)	RW
SU1HIM	External Clock Synchronous Enable Bit	(Note 2)	RW

NOTES:

1. Refer to **Table 17.14**.
2. The external clock synchronous function can be selected by combining the SU1HIM bit and the SCLKDIV bit in the UiSMR register.

SCLKDIV bit in the UiSMR Register	SU1HIM bit in the UiSMR2 Register	External Clock Synchronous Function Selection
0	0	No synchronization
0	1	Same division as the external clock
1	0 or 1	External clock divided by 2

Figure 17.6 U0SMR2 to U4SMR2 Registers

UARTi Special Mode Register 3 (i=0 to 4)

Symbol	Address	After Reset
U0SMR3 to U4SMR3	0365_{16}, $02E5_{16}$, 0335_{16}, 0325_{16}, $02F5_{16}$	00_{16}

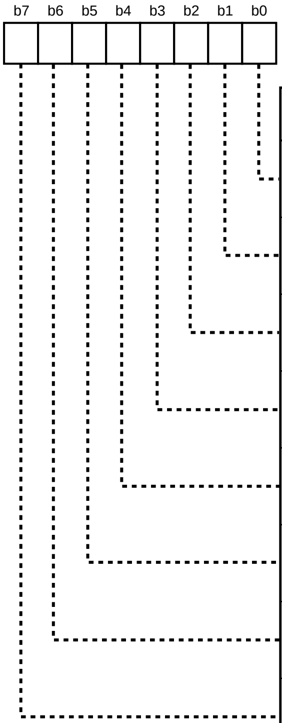
Bit Symbol	Bit Name	Function	RW
SSE	$\overline{SS}$ Pin Function Enable Bit[1]	0: Disables $\overline{SS}$ pin function 1: Enables $\overline{SS}$ pin function	RW
CKPH	Clock Phase Set Bit	0: No clock delay 1: Clock delay	RW
DINC	Serial Input Port Set Bit	0: Selects the TxDi and RxDi pins (master mode) 1: Selects the STxDi and SRxDi pins (slave mode)	RW
NODC	Clock Output Select Bit	0: CMOS output 1: N-channel open drain output	RW
ERR	Fault Error Flag[2]	0: No error 1: Error	RW
DL0	SDAi Digital Delay Time Set Bit[3, 4]	b7 b6 b5 000 : No delay 001 : 1-to-2 cycles of BRG count source 010 : 2-to-3 cycles of BRG count source 011 : 3-to-4 cycles of BRG count source 100 : 4-to-5 cycles of BRG count source 101 : 5-to-6 cycles of BRG count source 110 : 6-to-7 cycles of BRG count source 111 : 7-to-8 cycles of BRG count source	RW
DL1			RW
DL2			RW

NOTES:

1. Set the $\overline{SS}$ pin after the CRD bit in the UiC0 register is set to "1" ($\overline{CTS}/\overline{RTS}$ function disabled).
2. The ERR bit is set to "0" by program. It is unchanged if set to "1".
3. Digital delay is generated from a SDAi output by the DL2 to DL0 bits in I^2C mode. Set these bits to "000_2" (no delay) except in the I^2C mode.
4. When the external clock is selected, approximately 100ns delay is added.

Figure 17.7 U0SMR3 to U4SMR3 Registers

UARTi Special Mode Register 4 (i=0 to 4)



Symbol	Address	After Reset
U0SMR4 to U4SMR4	0364_{16}, $02E4_{16}$, 0334_{16}, 0324_{16}, $02F4_{16}$	00_{16}

Bit Symbol	Bit Name	Function	RW
STAREQ	Start Condition Generate Bit[1]	0: Clear 1: Start	RW
RSTAREQ	Restart Condition Generate Bit[1]	0: Clear 1: Start	RW
STPREQ	Stop Condition Generate Bit[1]	0: Clear 1: Start	RW
STSPSEL	SCL, SDA Output Select Bit	0: Selects the serial I/O circuit 1: Selects the start/stop condition generating circuit	RW
ACKD	ACK Data Bit	0: ACK 1: NACK	RW
ACKC	ACK Data Output Enable Bit	0: Serial I/O data output 1: ACK data output	RW
SCLHI	SCL Output Stop Enable Bit	0: Disabled 1: Enabled	RW
SWC9	SCL Wait Output Bit 3	0: SCL "L" hold disabled 1: SCL "L" hold enabled	RW

NOTES:

1. When each condition is generated, the STAREQ, RSTAREQ or STPREQ bit is set to "0". When a condition generation is incomplete, the bit remains unchanged as "1".

Figure 17.8 U0SMR4 to U4SMR4 Registers

External Interrupt Request Source Select Register

b7 b6 b5 b4 b3 b2 b1 b0

Symbol	Address	After Reset
IFSR	031F$_{16}$	00$_{16}$

Bit Symbol	Bit Name	Function	RW
IFSR0	INT0 Interrupt Polarity Select Bit[1]	0 : One edge 1 : Both edges	RW
IFSR1	INT1 Interrupt Polarity Select Bit[1]	0 : One edge 1 : Both edges	RW
IFSR2	INT2 Interrupt Polarity Select Bit[1]	0 : One edge 1 : Both edges	RW
IFSR3	INT3 Interrupt Polarity Select Bit[1]	0 : One edge 1 : Both edges	RW
IFSR4	INT4 Interrupt Polarity select bit[1]	0 : One edge 1 : Both edges	RW
IFSR5	INT5 Interrupt Polarity Select Bit[1]	0 : One edge 1 : Both edges	RW
IFSR6	UART0, UART3 Interrupt Source Select Bit	0 : UART3 bus conflict, start condition detect, stop condition detect 1 : UART0 bus conflict, start condition detect, stop condition detect	RW
IFSR7	UART1, UART4 Interrupt Source Select Bit	0 : UART4 bus conflict, start condition detect, stop condition detect 1 : UART1 bus conflict, start condition detect, stop condition detect	RW

NOTES:

1. Set this bit to "0" to select a level-sensitive triggering.
   When setting this bit to "1", set the POL bit in the INTiIC register (i = 0 to 5) to "0" (falling edge).

Figure 17.9 IFSR Register

17.1 Clock Synchronous Serial I/O Mode

In clock synchronous serial I/O mode, data is transmitted and received with the transfer clock. Table 17.1 lists specifications of clock synchronous serial I/O mode. Table 17.2 lists register settings. Tables 17.3 to 17.5 list pin settings. When UARTi (i=0 to 4) operating mode is selected, the TxDi pin outputs a high-level ("H") signal before transfer starts (the TxDi pin is in a high-impedance state when the N-channel open drain output is selected). Figure 17.10 shows transmit and receive timings in clock synchronous serial I/O mode.

Table 17.1 Clock Synchronous Serial I/O Mode Specifications

Item	Specification
Transfer Data Format	Transfer data : 8 bits long
Transfer Clock	• The CKDIR bit in the UiMR register (i=0 to 4) is set to "0" (internal clock selected): $\frac{fj}{2(m+1)}$ fj=f1, f8, f2n(1) m:setting value of the UiBRG register, 00_{16} to FF_{16} • The CKDIR bit is set to "1" (external clock selected) : an input from the CLKi pin
Transmit/Receive Control	Selected from the $\overline{\text{CTS}}$ function, $\overline{\text{RTS}}$ function or $\overline{\text{CTS}}$/$\overline{\text{RTS}}$ function disabled
Transmit Start Condition	To start transmitting, the following requirements must be met(2): - Set the TE bit in the UiC1 register to "1" (transmit enable) - Set the TI bit in the UiC1 register to "0" (data in the UiTB register) - Apply a low-level ("L") signal to the $\overline{\text{CTSi}}$ pin when the $\overline{\text{CTS}}$ function is selected
Receive Start Condition	To start receiving, the following requirements must be met(2): - Set the RE bit in the UiC1 register to "1" (receive enable) - Set the TE bit to "1" (transmit enable) - Set the TI bit to "0" (data in the UiTB register)
Interrupt Request Generation Timing	• While transmitting, the following conditions can be selected: - The UiIRS bit in the UiC1 register is set to "0" (no data in the transmit buffer): when data is transferred from the UiTB register to the UARTi transmit register (transfer started) - The UiIRS bit is set to "1" (transmission completed): when a data transfer from the UARTi transmit register is completed • While receiving When data is transferred from the UARTi receive register to the UiRB register (reception completed)
Error Detect	Overrun error(3) This error occurs when the seventh bit of the next received data is read before reading the UiRB register
Selectable Function	• CLK polarity Transferred data output and input are provided on either the rising edge or falling edge of the transfer clock • LSB first or MSB first Data is transmitted or received in either bit 0 or in bit 7 • Continuous receive mode Data can be received simultaneously by reading the UiRB register • Serial data logic inverse This function inverses transmitted/received data logically

NOTES:

1. The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2n (n=1 to 15).
2. To start transmission/reception when selecting the external clock, these conditions must be met after the CKPOL bit in the UiC0 register is set to "0" (data is transmitted on the falling edge of the transfer clock and data is received on the rising edge) and the CLKi pin is held "H", or when the CKPOL bit is set to "1" (data is transmitted on the rising edge of the transfer clock and data is received on the falling edge) and the CLKi pin is held "L".
3. If an overrun error occurs, the UiRB register is indeterminate. The IR bit in the SiRIC register does not change to "1" (interrupt requested).

Table 17.2 Register Settings in Clock Synchronous Serial I/O Mode

Register	Bit	Function
UiTB	7 to 0	Set transmit data
UiRB	7 to 0	Received data can be read
	OER	Overrun error flag
UiBRG	7 to 0	Set bit rate
UiMR	SMD2 to SMD0	Set to "001_2"
	CKDIR	Select the internal clock or external clock
	IOPOL	Set to "0"
UiC0	CLK1, CLK0	Select count source for the UiBRG register
	CRS	Select $\overline{\text{CTS}}$ or $\overline{\text{RTS}}$ when using either
	TXEPT	Transmit register empty flag
	CRD	Enables or disables the $\overline{\text{CTS}}$ or $\overline{\text{RTS}}$ function
	NCH	Select output format of the TxDi pin
	CKPOL	Select transmit clock polarity
	UFORM	Select either LSB first or MSB first
UiC1	TE	Set to "1" to enable data transmission and reception
	TI	Transmit buffer empty flag
	RE	Set to "1" to enable data reception
	RI	Reception complete flag
	UiIRS	Select what causes the UARTi transmit interrupt to be generated
	UiRRM	Set to "1" when using continuous receive mode
	UiLCH	Set to "1" when using data logic inverse
	SCLKSTPB	Set to "0"
UiSMR	7 to 0	Set to "00_{16}"
UiSMR2	7 to 0	Set to "00_{16}"
UiSMR3	2 to 0	Set to "000_2"
	NODC	Select clock output format
	7 to 4	Set to "0000_2"
UiSMR4	7 to 0	Set to "00_{16}"

i=0 to 4

Table 17.3 Pin Settings in Clock Synchronous Serial I/O Mode (1)

Port	Function	Setting		
		PS0 Register	PSL0 Register	PD6 Register
P6$_0$	$\overline{\text{CTS0}}$ input	PS0_0=0	-	PD6_0=0
	$\overline{\text{RTS0}}$ output	PS0_0=1	-	-
P6$_1$	CLK0 input	PS0_1=0	-	PD6_1=0
	CLK0 output	PS0_1=1	-	-
P6$_2$	RxD0 input	PS0_2=0	-	PD6_2=0
P6$_3$	TxD0 output	PS0_3=1	-	-
P6$_4$	$\overline{\text{CTS1}}$ input	PS0_4=0	-	PD6_4=0
	$\overline{\text{RTS1}}$ output	PS0_4=1	PSL0_4=0	-
P6$_5$	CLK1 input	PS0_5=0	-	PD6_5=0
	CLK1 output	PS0_5=1	-	-
P6$_6$	RxD1 input	PS0_6=0	-	PD6_6=0
P6$_7$	TxD1 output	PS0_7=1	-	-

Table 17.4 Pin Settings (2)

Port	Function	Setting			
		PS1 Register	PSL1 Register	PSC Register	PD7 Register
P7$_0$[1]	TxD2 output	PS1_0=1	PSL1_0=0	PSC_0=0	-
P7$_1$[1]	RxD2 input	PS1_1=0	-	-	PD7_1=0
P7$_2$	CLK2 input	PS1_2=0	-	-	PD7_2=0
	CLK2 output	PS1_2=1	PSL1_2=0	PSC_2=0	-
P7$_3$	$\overline{\text{CTS2}}$ input	PS1_3=0	-	-	PD7_3=0
	$\overline{\text{RTS2}}$ output	PS1_3=1	PSL1_3=0	PSC_3=0	-

NOTES:

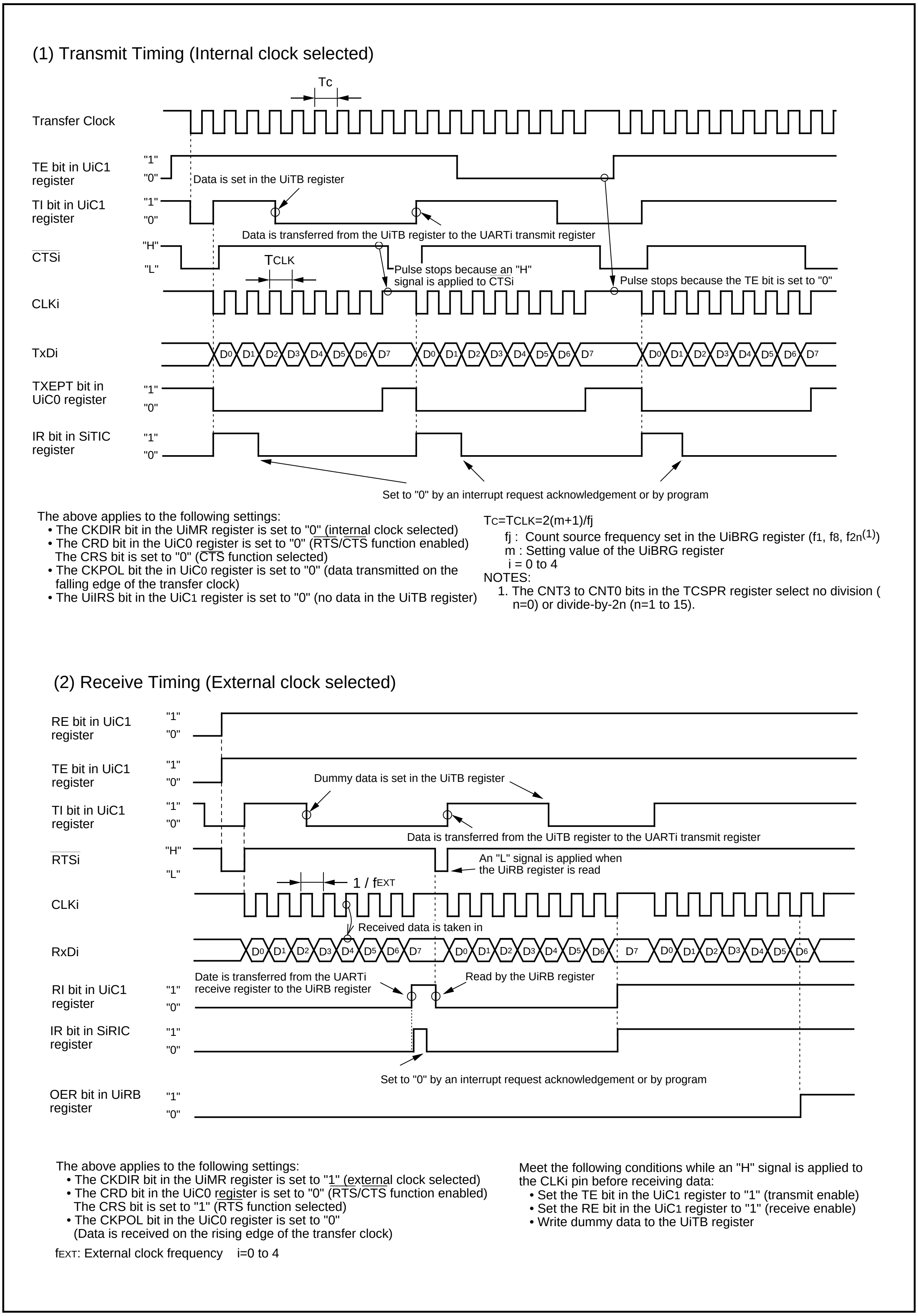
1. P7$_0$ and P7$_1$ are ports for the N-channel open drain output.

Table 17.5 Pin Settings (3)

Port	Function	Setting			
		PS3 Register[1]	PSL3 Register	PSC3 Register	PD9 Register[1]
P9$_0$	CLK3 input	PS3_0=0	-	-	PD9_0=0
	CLK3 output	PS3_0=1	-	-	-
P9$_1$	RxD3 input	PS3_1=0	-	-	PD9_1=0
P9$_2$	TxD3 output	PS3_2=1	PSL3_2=0	-	-
P9$_3$	$\overline{\text{CTS3}}$ input	PS3_3=0	PSL3_3=0	-	PD9_3=0
	$\overline{\text{RTS3}}$ output	PS3_3=1	-	-	-
P9$_4$	$\overline{\text{CTS4}}$ input	PS3_4=0	PSL3_4=0	-	PD9_4=0
	$\overline{\text{RTS4}}$ output	PS3_4=1	-	-	-
P9$_5$	CLK4 input	PS3_5=0	PSL3_5=0	-	PD9_5=0
	CLK4 output	PS3_5=1	-	-	-
P9$_6$	TxD4 output	PS3_6=1	-	PSC3_6=0	-
P9$_7$	RxD4 input	PS3_7=0	-	-	PD9_7=0

NOTES:

1. Set the PD9 and PS3 registers immediately after the PRC2 bit in the PRCR register is set to "1" (write enable). Do not generate an interrupt or a DMA transfer between the instruction to set the PRC2 bit to "1" and the instruction to set the PD9 and PS3 registers.


(1) Transmit Timing (Internal clock selected)
Tc
Transfer Clock
TE bit in UiC1 register
"1"
"0"
Data is set in the UiTB register
TI bit in UiC1 register
"1"
"0"
Data is transferred from the UiTB register to the UARTi transmit register
CTSi
"H"
"L"
TCLK
Pulse stops because an "H" signal is applied to CTSi
Pulse stops because the TE bit is set to "0"
CLKi
TxDi
D0 D1 D2 D3 D4 D5 D6 D7
TXEPT bit in UiC0 register
"1"
"0"
IR bit in SiTIC register
"1"
"0"
Set to "0" by an interrupt request acknowledgement or by program

The above applies to the following settings:

- The CKDIR bit in the UiMR register is set to "0" (internal clock selected)
- The CRD bit in the UiC0 register is set to "0" (RTS/CTS function enabled) The CRS bit is set to "0" (CTS function selected)
- The CKPOL bit the in UiC0 register is set to "0" (data transmitted on the falling edge of the transfer clock)
- The UiIRS bit in the UiC1 register is set to "0" (no data in the UiTB register)

TC=TCLK=2(m+1)/fj

fj : Count source frequency set in the UiBRG register (f1, f8, f2n[1])
m : Setting value of the UiBRG register
i = 0 to 4

NOTES:

1. The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2n (n=1 to 15).

(2) Receive Timing (External clock selected)
RE bit in UiC1 register
"1"
"0"
TE bit in UiC1 register
"1"
"0"
Dummy data is set in the UiTB register
TI bit in UiC1 register
"1"
"0"
Data is transferred from the UiTB register to the UARTi transmit register
RTSi
"H"
"L"
An "L" signal is applied when the UiRB register is read
1 / fEXT
CLKi
Received data is taken in
RxDi
D0 D1 D2 D3 D4 D5 D6 D7
Date is transferred from the UARTi receive register to the UiRB register
Read by the UiRB register
RI bit in UiC1 register
"1"
"0"
IR bit in SiRIC register
"1"
"0"
Set to "0" by an interrupt request acknowledgement or by program
OER bit in UiRB register
"1"
"0"

The above applies to the following settings:

- The CKDIR bit in the UiMR register is set to "1" (external clock selected)
- The CRD bit in the UiC0 register is set to "0" (RTS/CTS function enabled) The CRS bit is set to "1" (RTS function selected)
- The CKPOL bit in the UiC0 register is set to "0" (Data is received on the rising edge of the transfer clock)

fEXT: External clock frequency i=0 to 4

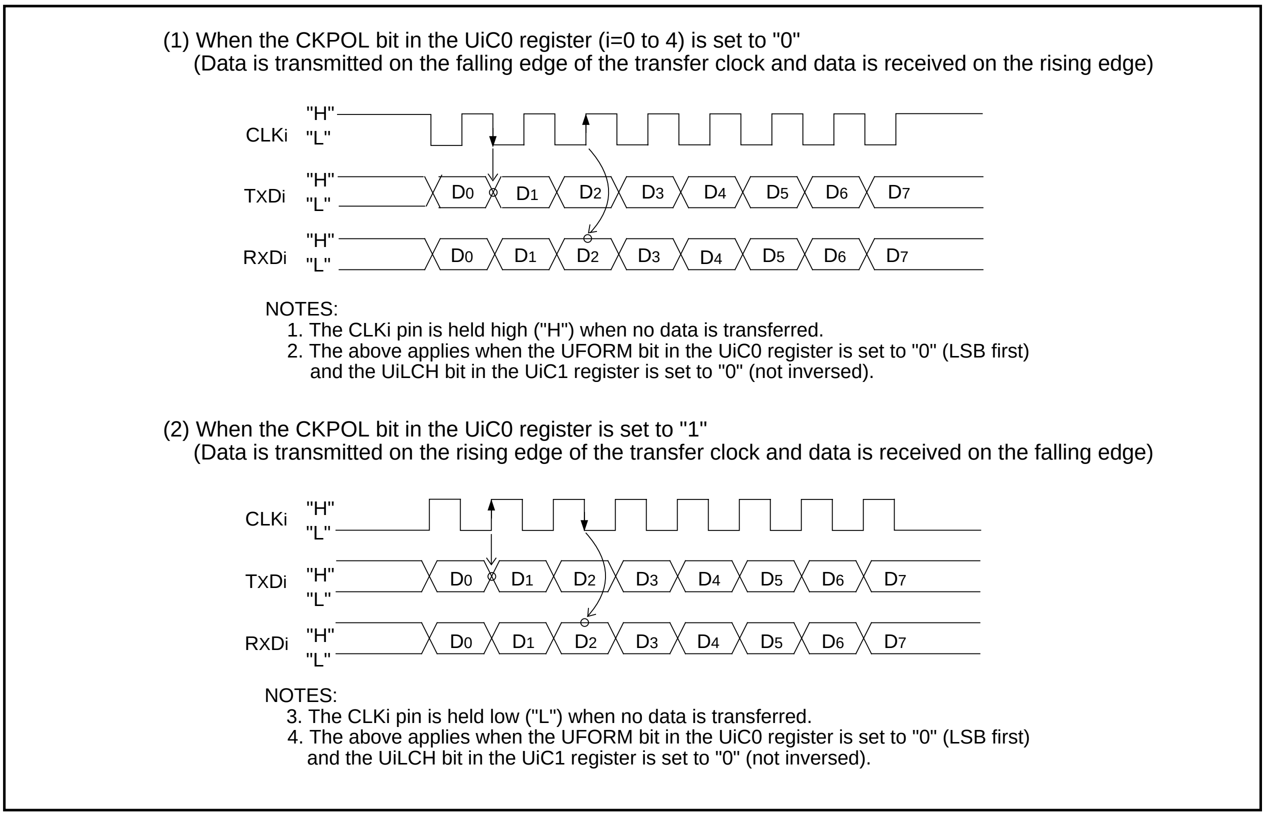
Meet the following conditions while an "H" signal is applied to the CLKi pin before receiving data:

- Set the TE bit in the UiC1 register to "1" (transmit enable)
- Set the RE bit in the UiC1 register to "1" (receive enable)
- Write dummy data to the UiTB register

Figure 17.10 Transmit and Receive Operation

17.1.1 Selecting CLK Polarity Selecting

As shown in Figure 17.11, the CKPOL bit in the UiC0 register (i=0 to 4) determines the polarity of the transfer clock.



(1) When the CKPOL bit in the UiC0 register (i=0 to 4) is set to "0"
(Data is transmitted on the falling edge of the transfer clock and data is received on the rising edge)

NOTES:
1. The CLKi pin is held high ("H") when no data is transferred.
2. The above applies when the UFORM bit in the UiC0 register is set to "0" (LSB first) and the UiLCH bit in the UiC1 register is set to "0" (not inversed).

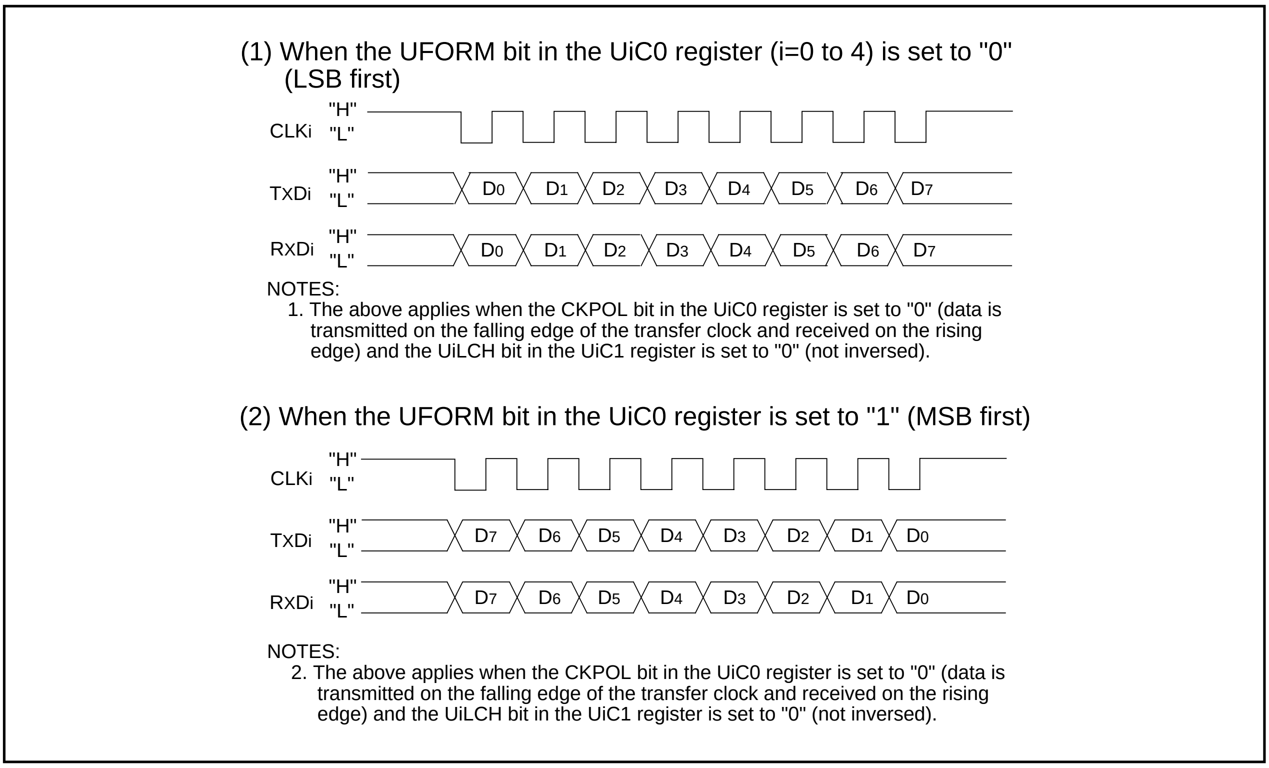
(2) When the CKPOL bit in the UiC0 register is set to "1"
(Data is transmitted on the rising edge of the transfer clock and data is received on the falling edge)

NOTES:
3. The CLKi pin is held low ("L") when no data is transferred.
4. The above applies when the UFORM bit in the UiC0 register is set to "0" (LSB first) and the UiLCH bit in the UiC1 register is set to "0" (not inversed).

Figure 17.11 Transfer Clock Polarity

17.1.2 Selecting LSB First or MSB First

As shown in Figure 17.12, the UFORM bit in the UiC0 register (i=0 to 4) determines a data transfer format.



(1) When the UFORM bit in the UiC0 register (i=0 to 4) is set to "0"
(LSB first)

NOTES:
1. The above applies when the CKPOL bit in the UiC0 register is set to "0" (data is transmitted on the falling edge of the transfer clock and received on the rising edge) and the UiLCH bit in the UiC1 register is set to "0" (not inversed).

(2) When the UFORM bit in the UiC0 register is set to "1" (MSB first)

NOTES:
2. The above applies when the CKPOL bit in the UiC0 register is set to "0" (data is transmitted on the falling edge of the transfer clock and received on the rising edge) and the UiLCH bit in the UiC1 register is set to "0" (not inversed).

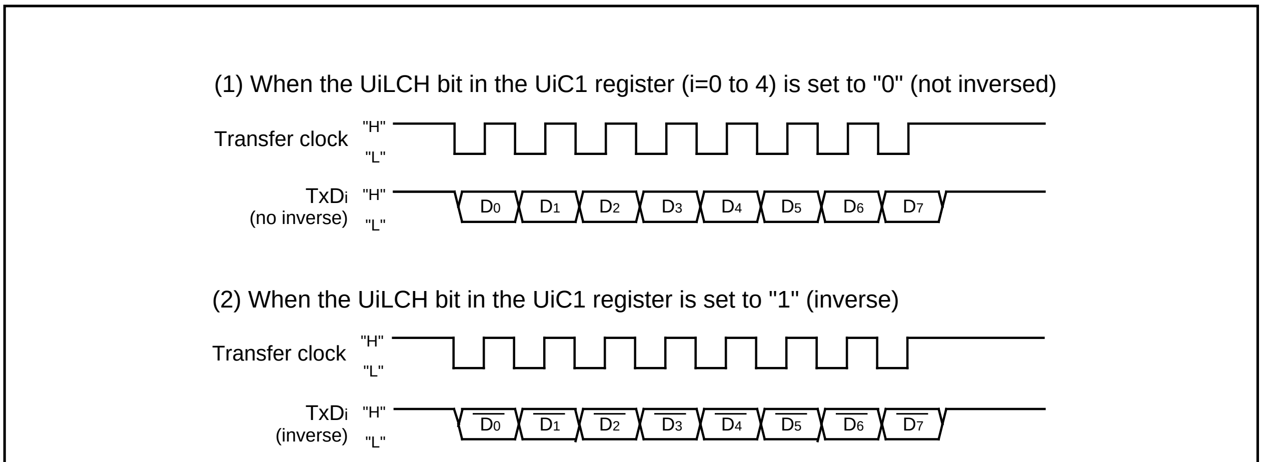
Figure 17.12 Transfer Format

17.1.3 Continuous Receive Mode

When the UiRRM bit in the UiC1 register (i=0 to 4) is set to "1" (continuous receive mode), the TI bit is set to "0" (data in the UiTB register) by reading the UiRB register. When the UiRRM bit is set to "1", do not set dummy data in the UiTB register by program.

17.1.4 Serial Data Logic Inverse

When the UiLCH bit (i=0 to 4) in the UiC1 register is set to "1" (inverse), data logic written in the UiTB register is inversed when transmitted. The inversed receive data logic can be read by reading the UiRB register. Figure 17.13 shows a switching example of the serial data logic.


(1) When the UiLCH bit in the UiC1 register (i=0 to 4) is set to "0" (not inversed)
Transfer clock
"H"
"L"
TxDi
(no inverse)
"H"
"L"
D0 D1 D2 D3 D4 D5 D6 D7
(2) When the UiLCH bit in the UiC1 register is set to "1" (inverse)
Transfer clock
"H"
"L"
TxDi
(inverse)
"H"
"L"
$\overline{D0}$ $\overline{D1}$ $\overline{D2}$ $\overline{D3}$ $\overline{D4}$ $\overline{D5}$ $\overline{D6}$ $\overline{D7}$

NOTES:
1. The above applies when the CKPOL bit in the UiC0 register is set to "0" (data is transmitted on the falling edge) and the UFORM bit in the UiC0 register is set to "0" (LSB first).

Figure 17.13 Serial Data Logic Inverse

17.2 Clock Asynchronous Serial I/O (UART) Mode

In UART mode, data is transmitted and received after setting a desired bit rate and data transfer format. Table 17.6 lists specifications of UART mode.

Table 17.6 UART Mode Specifications

Item	Specification
Transfer Data Format	• Character bit (transfer data) : selected from 7 bits, 8 bits, or 9 bits long • Start bit: 1 bit long • Parity bit: selected from odd, even, or none • Stop bit: selected from 1 bit or 2 bits long
Transfer Clock	• The CKDIR bit in the UiMR register is set to "0" (internal clock selected): *fj/16(m+1)* *fj* = f_1, f_8, f_{2n}(1) *m*: setting value of the UiBRG register , 00_{16} to FF_{16} • The CKDIR bit is set to "1" (external clock selected): *fEXT/16(m+1)* *fEXT*: clock applied to the CLKi pin
Transmit/Receive Control	Select from $\overline{\mathrm{CTS}}$ function, $\overline{\mathrm{RTS}}$ function or $\overline{\mathrm{CTS}}$/$\overline{\mathrm{RTS}}$ function disabled
Transmit Start Condition	To start transmitting, the following requirements must be met: - Set the TE bit in the UiC1 register to "1" (transmit enable) - Set the TI bit in the UiC1 register to "0" (data in the UiTB register) - Apply a low-velel ("L") signal to the $\overline{\mathrm{CTS}}$i pin when the $\overline{\mathrm{CTS}}$ function is selected
Receive Start Condition	To start receiving, the following requirements must be met: - Set the RE bit in the UiC1 register to "1" (receive enable) - The start bit is detected
Interrupt Request Generation Timing	While transmitting, the following condition can be selected: - The UiIRS bit in the UiC1 register is set to "0" (no data in the UiTB register): when data is transferred from the UiTB register to the UARTi transmit register (transfer started) - The UiIRS bit is set to "1" (transmission completed): when data transmission from the UARTi transfer register is completed While receiving when data is transferred from the UARTi receive register to the UiRB register (reception completed)
Error Detect	• Overrun error(2) This error occurs when the bit before the last stop bit of the next received data is read prior to reading the UiRB register (the first stop bit when selecting 2 stop bits) • Framing error This error occurs when the number of stop bits set is not detected • Parity error When parity is enabled, this error occurs when the number of "1" in parity and character bits does not match the number of "1" set • Error sum flag This flag is set to "1" when any of an overrun, framing or parity errors occur
Selectable Function	• LSB first or MSB first Data is transmitted or received in either bit 0 or in bit 7 •Serial data logic inverse Logic values of data to be transmitted and received data are inversed. The start bit and stop bit are not inversed •TxD and RxD I/O polarity Inverse TxD pin output and RxD pin input are inversed. All I/O data levels are also inversed

NOTES:

1. The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2n (n=1 to 15).
2. If an overrun error occurs, the UiRB register is indeterminate. The IR bit in the SiRIC register remains unchanged as "1" (interrupt requested).

Table 17.7 lists register settings. Tables 17.8 to 17.10 list pin settings. When UARTi (i=0 to 4) operating mode is selected, the TxDi pin outputs a high-level ("H") signal before transfer is started (the TxDi pin is in a high-impedance state when the N-channel open drain output is selected). Figure 17.14 shows an example of a transmit operation in UART mode. Figure 17.15 shows an example of a receive operation in UART mode.

Table 17.7 Register Settings in UART Mode

Register	Bit	Function
UiTB	8 to 0	Set transmit data(1)
UiRB	8 to 0	Received data can be read(1)
	OER, FER, PER, SUM	Error flags
UiBRG	7 to 0	Set bit rate
UiMR	SMD2 to SMD0	Set to "100_2" when transfer data is 7 bits long Set to "101_2" when transfer data is 8 bits long Set to "110_2" when transfer data is 9 bits long
	CKDIR	Select the internal clock or external clock
	STPS	Select stop bit length
	PRY, PRYE	Select parity enable or disable, odd or even
	IOPOL	Select TxD and RxD I/O polarity
UiC0	CLK1, CLK0	Select count source for the UiBRG register
	CRS	Select either $\overline{\text{CTS}}$ or $\overline{\text{RTS}}$ when using either
	TXEPT	Transfer register empty flag
	CRD	Select the $\overline{\text{CTS}}$ or $\overline{\text{RTS}}$ function enabled or disabled
	NCH	Select output format of the TxDi pin
	CKPOL	Set to "0"
	UFORM	Select the LSB first or MSB first when a transfer data is 8 bits long Set to "0" when transfer data is 7 bits or 9 bits long
UiC1	TE	Set to "1" to enable data transmission
	TI	Transfer buffer empty flag
	RE	Set to "1" to enable data reception
	RI	Reception complete flag
	UiIRS	Select what causes the UARTi transmit interrupt to be generated
	UiRRM	Set to "0"
	UiLCH	Select whether data logic is inversed or not inversed when a transfer data is 7 bits or 8 bits long. Set to "0" when transfer data is 9 bits long
	UiERE	Set to either "0" or "1"
UiSMR	7 to 0	Set to "00_{16}"
UiSMR2	7 to 0	Set to "00_{16}"
UiSMR3	7 to 0	Set to "00_{16}"
UiSMR4	7 to 0	Set to "00_{16}"

NOTES:

1. Use bits 0 to 6 when transfer data is 7 bits long, bits 0 to 7 when 8 bits long, bits 0 to 8 when 9 bits long.

Table 17.8 Pin Settings in UART Mode (1)

Port	Function	Setting		
		PS0 Register	PSL0 Register	PD6 Register
P60	$\overline{\text{CTS0}}$ input	PS0_0=0	–	PD6_0=0
	$\overline{\text{RTS0}}$ output	PS0_0=1	–	–
P61	CLK0 input	PS0_1=0	–	PD6_1=0
P62	RxD0 input	PS0_2=0	–	PD6_2=0
P63	TxD0 output	PS0_3=1	–	–
P64	$\overline{\text{CTS1}}$ input	PS0_4=0	–	PD6_4=0
	$\overline{\text{RTS1}}$ output	PS0_4=1	PSL0_4=0	–
P65	CLK1 input	PS0_5=0	–	PD6_5=0
P66	RxD1 input	PS0_6=0	–	PD6_6=0
P67	TxD1 output	PS0_7=1	–	–

Table 17.9 Pin Settings (2)

Port	Function	Setting			
		PS1 Register	PSL1 Register	PSC Register	PD7 Register
P70[1]	TxD2 output	PS1_0=1	PSL1_0=0	PSC_0=0	–
P71[1]	RxD2 input	PS1_1=0	–	–	PD7_1=0
P72	CLK2 input	PS1_2=0	–	–	PD7_2=0
P73	$\overline{\text{CTS2}}$ input	PS1_3=0	–	–	PD7_3=0
	$\overline{\text{RTS2}}$ output	PS1_3=1	PSL1_3=0	PSC_3=0	–

NOTES:

1. P70 and P71 are ports for the N-channel open drain output.

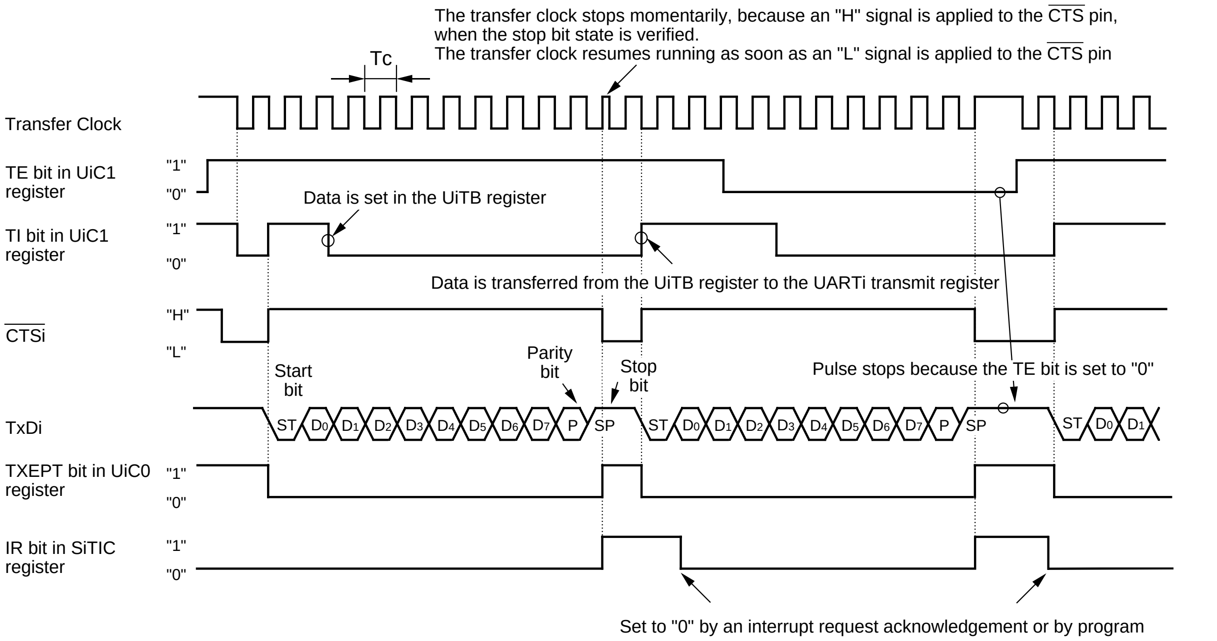
Table 17.10 Pin Settings (3)

Port	Function	Setting			
		PS3 Register[1]	PSL3 Register	PSC3 Register	PD9 Register[1]
P90	CLK3 input	PS3_0=0	–	–	PD9_0=0
P91	RxD3 input	PS3_1=0	–	–	PD9_1=0
P92	TxD3 output	PS3_2=1	PSL3_2=0	–	–
P93	$\overline{\text{CTS3}}$ input	PS3_3=0	PSL3_3=0	–	PD9_3=0
	$\overline{\text{RTS3}}$ output	PS3_3=1	–	–	–
P94	$\overline{\text{CTS4}}$ input	PS3_4=0	PSL3_4=0	–	PD9_4=0
	$\overline{\text{RTS4}}$ output	PS3_4=1	–	–	–
P95	CLK4 input	PS3_5=0	PSL3_5=0	–	PD9_5=0
P96	TxD4 output	PS3_6=1	–	PSC3_6=0	–
P97	RxD4 input	PS3_7=0	–	–	PD9_7=0

NOTES:

1. Set the PD9 and PS3 registers set immediately after the PRC2 bit in the PRCR register is set to "1" (write enable). Do not generate an interrupt or a DMA transfer between the instruction to set to the PRC2 bit to "1" and the instruction to set the PD9 and PS3 registers.

(1) 8-bit Data Transmit Timing (with a parity and 1 stop bit)

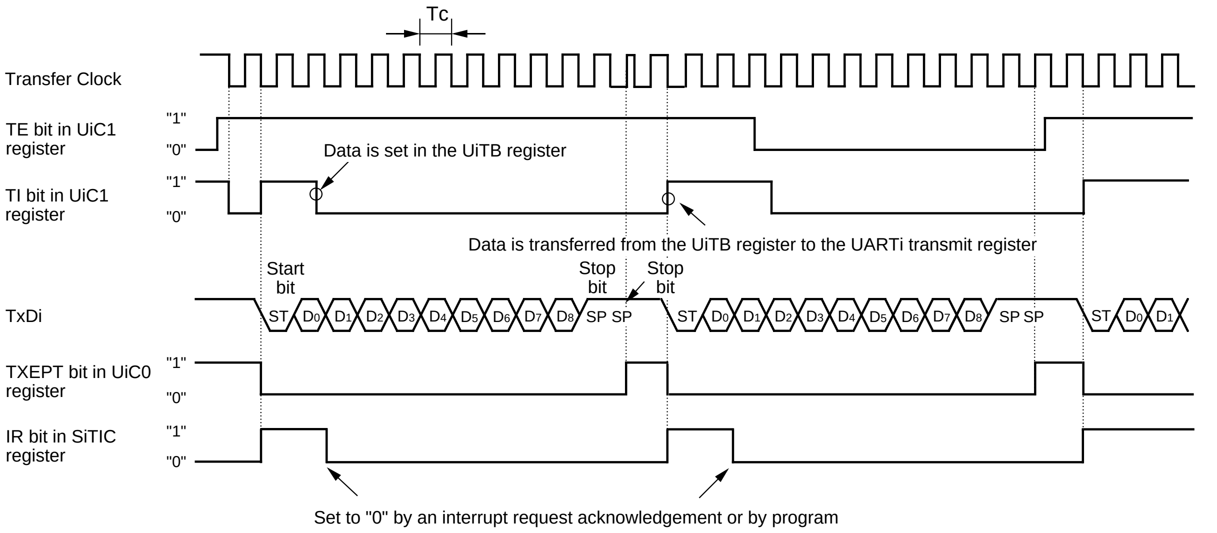

The transfer clock stops momentarily, because an "H" signal is applied to the $\overline{CTS}$ pin, when the stop bit state is verified.
The transfer clock resumes running as soon as an "L" signal is applied to the $\overline{CTS}$ pin
Tc
Transfer Clock
TE bit in UiC1 register
"1"
"0"
TI bit in UiC1 register
"1"
"0"
Data is set in the UiTB register
Data is transferred from the UiTB register to the UARTi transmit register
$\overline{CTSi}$
"H"
"L"
Start bit
Parity bit
Stop bit
Pulse stops because the TE bit is set to "0"
TxDi
ST D0 D1 D2 D3 D4 D5 D6 D7 P SP ST D0 D1 D2 D3 D4 D5 D6 D7 P SP ST D0 D1
TXEPT bit in UiC0 register
"1"
"0"
IR bit in SiTIC register
"1"
"0"
Set to "0" by an interrupt request acknowledgement or by program

i=0 to 4
The above timing applies to the following settings :
- The PRYE bit in the UiMR register is set to "1" (parity enabled)
- The STPS bit in the UiMR register is set to "0" (1 stop bit)
- The CRD bit in the UiC0 register is set to "0" and the CRS bit is set to "0" (CTS function selected)
- The UiIRS bit in the UiC1 register is set to "1" (transmission completed)

Tc = 16 (m + 1) / fj or 16 (m + 1) / fEXT
fj : count source frequency set in the UiBRG register (f1, f8, f2n(1))
fEXT : count source frequency set in the UiBRG register (external clock)
m : setting value of the UiBRG register

NOTES:
1. The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2n (n=1 to 15).

(2) 9-bit Data Transmit Timing (with no parity and 2 stop bits)

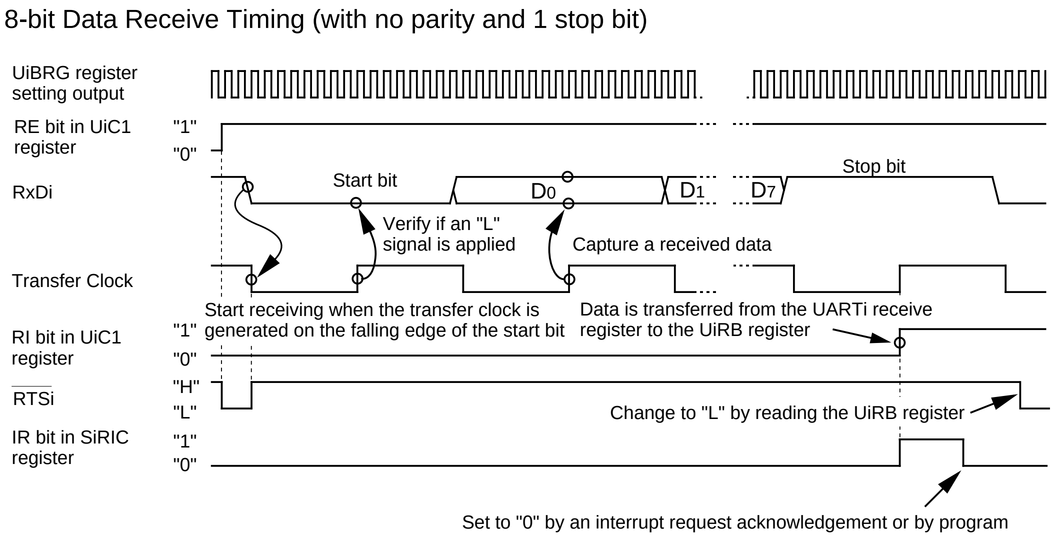

Tc
Transfer Clock
TE bit in UiC1 register
"1"
"0"
Data is set in the UiTB register
TI bit in UiC1 register
"1"
"0"
Data is transferred from the UiTB register to the UARTi transmit register
Start bit
Stop bit
Stop bit
TxDi
ST D0 D1 D2 D3 D4 D5 D6 D7 D8 SP SP ST D0 D1 D2 D3 D4 D5 D6 D7 D8 SP SP ST D0 D1
TXEPT bit in UiC0 register
"1"
"0"
IR bit in SiTIC register
"1"
"0"
Set to "0" by an interrupt request acknowledgement or by program

i=0 to 4
The above timing applies to the following settings :
- The PRYE bit in the UiMR register is set to "0" (parity disabled)
- The STPS bit in the UiMR register is set to "1" (2 stop bits)
- The CRD bit in the UiC0 register is set to "1" ($\overline{CTS}$ function disabled)
- The UiIRS bit in the UiC1 register is set to "0" (no data in the transmit buffer)

Tc = 16 (m + 1) / fj or 16 (m + 1) / fEXT
fj : count source frequency set in the UiBRG register (f1, f8, f2n(1))
fEXT : count source frequency set in the UiBRG register (external clock)
m : setting value of the UiBRG register

NOTES:
1. The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2n (n=1 to 15).

Figure 17.14 Transmit Operation


8-bit Data Receive Timing (with no parity and 1 stop bit)
UiBRG register setting output
RE bit in UiC1 register
"1"
"0"
RxDi
Start bit
D0
D1
D7
Stop bit
Verify if an "L" signal is applied
Capture a received data
Transfer Clock
Start receiving when the transfer clock is generated on the falling edge of the start bit
Data is transferred from the UARTi receive register to the UiRB register
RI bit in UiC1 register
"1"
"0"
RTSi
"H"
"L"
Change to "L" by reading the UiRB register
IR bit in SiRIC register
"1"
"0"
Set to "0" by an interrupt request acknowledgement or by program

i=0 to 4
NOTES:
1. The above applies when the PRYE bit in the UiMR register is set to "0" (parity disabled), the STPS bit in the UiMR register is set to "0" (1 stop bit) and the CRS bit in the UiC0 register is set to "1" (RTS function selected).

Figure 17.15 Receive Operation

17.2.1 Bit Rate

In UART mode, bit rate is clock frequency which is divided by a setting value of the UiBRG (i=0 to 4) register and again divided by 16. Table 17.11 lists an example of bit rate setting.

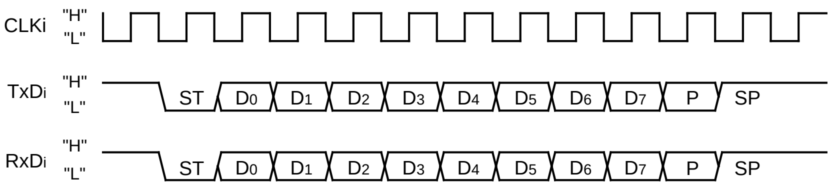
Table 17.11 Bit Rate

Bit Rate (bps)	Count Source of UiBRG	Peripheral Function Clock: 16MHz		Peripheral Function Clock: 24MHz		Peripheral Function Clock: 32MHz	
		Setting Value of UiBRG: n	Actual Bit Rate (bps)	Setting Value of UiBRG: n	Actual Bit Rate (bps)	Setting Value of UiBRG: n	Actual Bit Rate (bps)
1200	f8	103 (67h)	1202	155 (96h)	1202	207 (CFh)	1202
2400	f8	51 (33h)	2404	77 (46h)	2404	103 (67h)	2404
4800	f8	25 (19h)	4808	38 (26h)	4808	51 (33h)	4808
9600	f1	103 (67h)	9615	155 (96h)	9615	207 (CFh)	9615
14400	f1	68 (44h)	14493	103 (67h)	14423	138 (8Ah)	14388
19200	f1	51 (33h)	19231	77 (46h)	19231	103 (67h)	19231
28800	f1	34 (22h)	28571	51 (33h)	28846	68 (44h)	28986
31250	f1	31 (1Fh)	31250	47 (2Fh)	31250	63 (3Fh)	31250
38400	f1	25 (19h)	38462	38 (26h)	38462	51 (33h)	38462
51200	f1	19 (13h)	50000	28 (1Ch)	51724	38 (26h)	51282

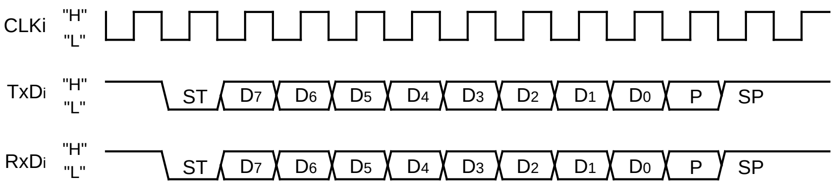
17.2.2 Selecting LSB First or MSB First

As shown in Figure 17.16, the UFORM bit in the UiC0 register (i=0 to 4) determines data transfer format. This function is available for 8-bit transfer data.

(1) When the UFORM Bit in the UiC0 Register (i=0 to 4) is set to "0" (LSB first)



(2) When the UFORM Bit in the UiC0 Register is set to "1" (MSB first)



NOTES:
1. The above applies when the CKPOL bit in the UiC0 register is set to "0" (data is transmitted on the falling edge of the transfer clock and received on the rising edge) and the UiLCH bit in the UiC1 register is set to "0" (no inverse).

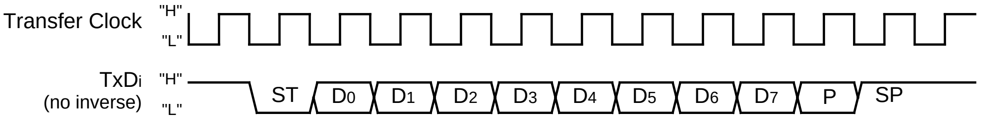
ST : Start bit
P : Parity bit
SP : Stop bit

Figure 17.16 Transfer Format

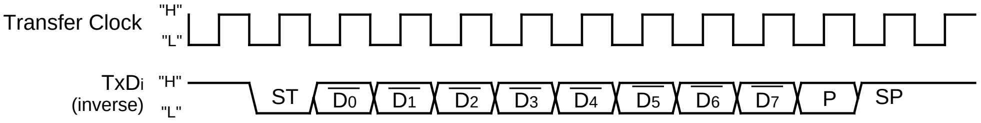
17.2.3 Serial Data Logic Inverse

When the UiLCH bit (i=0 to 4) in the UiC1 register is set to "1" (inverse), data logic written in the UiTB register is inversed when transmitted. The inversed receive data logic can be read by reading the UiRB register. Figure 17.17 shows a switching example of the serial data logic.

(1) When the UiLCH bit in the UiC1 register (i=0 to 4) is set to "0" (no inverse)



(2) When the UiLCH bit in the UiC1 register is set to "1" (inverse)



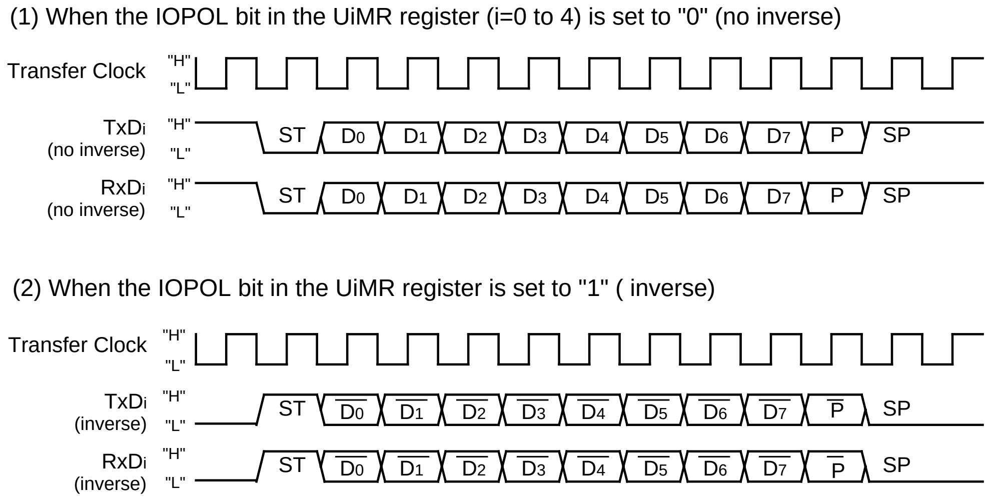
NOTES:
1. The above applies to when the UFORM bit in the UiC0 register is set to "0" (LSB first), the STPS bit in the UiMR register is set to "0" (1 stop bit) and the PRYE bit is set to "1" (parity enabled).

ST : Start bit
P : Parity bit
SP : Stop bit

Figure 17.17 Serial Data Logic Inverse

17.2.4 TxD and RxD I/O Polarity Inverse

TxD pin output and RxD pin input are inversed. All I/O data level, including the start bit, stop bit and parity bit, are inversed. Figure 17.18 shows TxD and RxD I/O polarity inverse.


(1) When the IOPOL bit in the UiMR register (i=0 to 4) is set to "0" (no inverse)
Transfer Clock
"H"
"L"
TxDi
(no inverse)
RxDi
(no inverse)
ST
D0
D1
D2
D3
D4
D5
D6
D7
P
SP
(2) When the IOPOL bit in the UiMR register is set to "1" (inverse)
Transfer Clock
TxDi
(inverse)
RxDi
(inverse)

NOTES:

1. The above applies when the UFORM bit in the UiC0 register is set to "0" (LSB first), the STPS bit in the UiMR bit is set to "0" (1 stop bit) and the PRYE bit is set to "1" (parity enabled).

ST : Start bit
P : Even parity
SP : Stop bit

Figure 17.18 TxD and RxD I/O Polarity Inverse

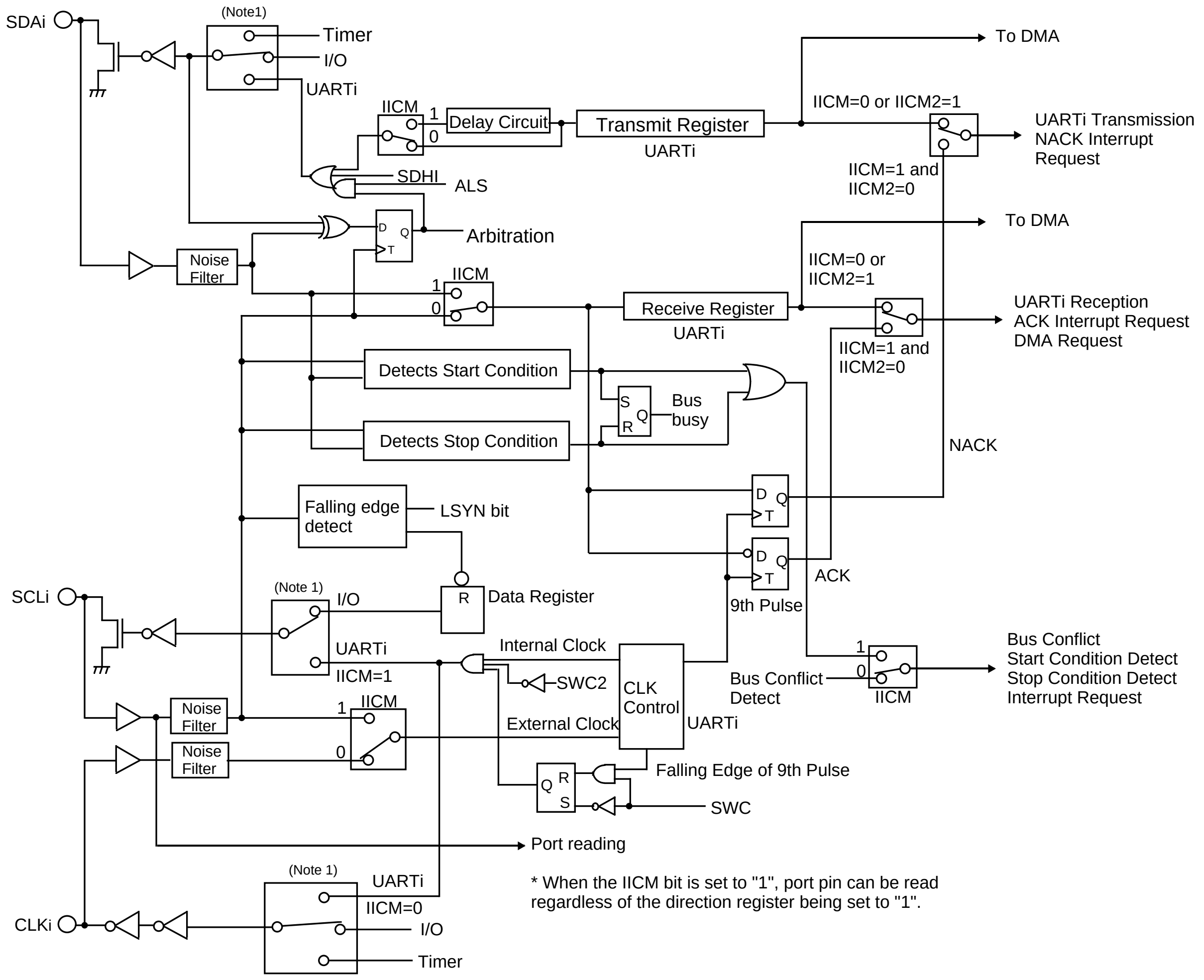
17.3 Special Mode 1 (I^2C Mode)

I^2C mode is a mode to communicate with external devices with a simplified I^2C. Table 17.12 lists specifications of I^2C mode. Table 17.13 lists register settings, Table 17.14 lists each function. Figure 17.19 shows a block diagram of I^2C mode. Figure 17.20 shows timings for transfer to the UiRB register (i=0 to 4) and interrupts. Tables 17.15 to 17.17 list pin settings.

As shown in Table 17.12, I^2C mode is entered when the SMD2 to SMD0 bits in the UiMR register is set to "010_2" and the IICM bit in the UiSMR register is set to "1". Output signal from the SDAi pin changes after the SCLi pin level becomes low ("L") and stabilizes due to a SDAi transmit output via the delay circuit.

Table 17.12 I^2C Mode Specifications

Item	Specifications
Interrupt	Start condition detect, stop condition detect, no acknowledgment detect, acknowledgment detect
Selectable Function	• Arbitration lost The update timing of the ABT bit in the UiRB register can be selected. Refer to **17.3.3 Arbitration** • SDAi digital delay Selected from no digital delay or 2 to 8 cycle delay of the count source of the UiBRG register. Refer to **17.3.5 SDA Output** • Clock phase setting Selected from clock delay or no clock delay. Refer to **17.3.4 Transfer clock**



i=0 to 4

NOTES:

1. Set the PSj (j=0,1,3), PSLj or PSC register to determine.

IICM : Bit in the UiSMR register
IICM2 : Bit in the UiSMR2 register

Figure 17.19 I²C Mode Block Diagram

Table 17.13 Register Settings in I²C Mode

Register	Bit	Function	
		Master	Slave
UiTB	7 to 0	Set transmit data	
UiRB	7 to 0	Received data can be read	
	8	ACK or NACK bit can be read	
	ABT	Arbitration lost detect flag	Disabled
	OER	Overrun error flag	
UiBRG	7 to 0	Set bit rate	Disabled
UiMR	SMD2 to SMD0	Set to "010_2"	
	CKDIR	Set to "0"	Set to "1"
	IOPOL	Set to "0"	
UiC0	CLK1, CLK0	Select count source of the UiBRG register	Disabled
	CRS	Disabled because the CRD bit is set to "1"	
	TXEPT	Transfer register empty flag	
	CRD, NCH	Set to "1"	
	CKPOL	Set to "0"	
	UFORM	Set to "1"	
UiC1	TE	Set to "1" to enable data transmission	
	TI	Transfer buffer empty flag	
	RE	Set to "1" to enable data reception	
	RI	Reception complete flag	
	UiRRM, UiLCH, UiERE	Set to "0"	
UiSMR	IICM	Set to "1"	
	ABC	Select an arbitration lost detect timing	Disabled
	BBS	Bus busy flag	
	7 to 3	Set to "00000_2"	
UiSMR2	IICM2	See Table 17.14	
	CSC	Set to "1" to enable clock synchronization	Set to "0"
	SWC	Set to "1" to fix an "L" signal output from SCLi on the falling edge of the ninth bit of the transfer clock	
	ALS	Set to "1" to terminate SDAi output when detecting the arbitration lost	Not used. Set to "0"
	STC	Not used. Set to "0"	Set to "1" to reset UARTi by detecting the start condition
	SWC2	Set to "1" for an "L" signal output from SCL forcibly	
	SDHI	Set to "1" to disable SDA output	
	SU1HIM	Set to "0"	
UiSMR3	SSE	Set to "0"	
	CKPH	See Table 17.14	
	DINC, NODC, ERR	Set to "0"	
	DL2 to DL0	Set digital delay value	
UiSMR4	STAREQ	Set to "1" when generating a start condition	Not used. Set to "0"
	RSTAREQ	Set to "1" when generating a restart condition	
	STPREQ	Set to "1" when generating a stop condition	
	STSPSEL	Set to "1" when using a condition generating function	
	ACKD	Select ACK or NACK	
	ACKC	Set to "1" for ACK data output	
	SCLHI	Set to "1" to enable SCL output stop when detecting stop condition	Not used. Set to "0"
	SWC9	Not used. Set to "0"	Set to "1" to fix an "L" signal output from SCLi on the falling edge of the ninth bit of the transfer clock
IFSR	IFSR6, IFSR7	Set to "1"	

i=0 to 4

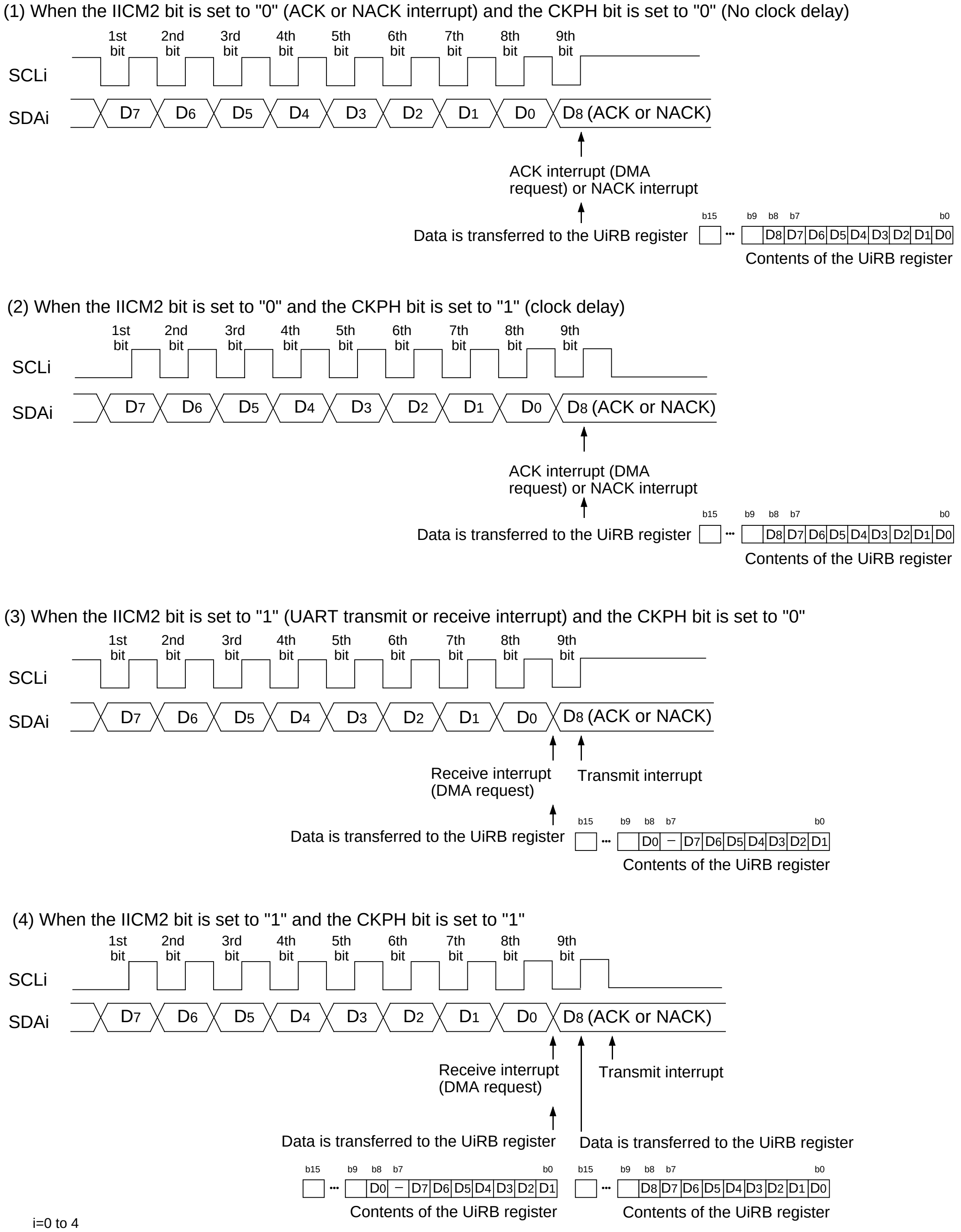
Table 17.14 I^2C Mode Functions

Function	Clock Synchronous Serial I/O Mode (SMD2 to SMD0=001_2, IICM=0)	I^2C Mode (SMD2 to SMD0=010_2, IICM=1)			
		IICM2=0 (NACK/ACK interrupt)		IICM2=1 (UART transmit / UART receive interrupt)	
		CKPH=0 (No clock delay)	CKPH=1 (Clock delay)	CKPH=0 (No clock delay)	CKPH=1 (Clock delay)
Interrupt Numbers 39 to 41 Generated[1] (See Figure 17.20)	-	Start condition or stop condition detect (See Table 17.18)			
Interrupt Number 17, 19, 33, 35 and 37 Generated[1] (See Figure 17.20)	UARTi Transmission - Transmission started or completed (selected by the UiIRS register)	No Acknowledgement Detection (NACK) - Rising edge of 9th bit of SCLi		UARTi Transmission - Rising edge of 9th bit of SCLi	UARTi Transmission - Next falling edge after the 9th bit of SCLi
Interrupt Numbers 18, 20, 34, 36 and 38 Generated[1] (See Figure 17.20)	UARTi Reception - Receiving at 8th bit CKPOL=0(rising edge) CKPOL=1(falling edge)	Acknowledgement Detection (ACK) - Rising edge of 9th bit of SCLi		UARTi Reception - Falling edge of 9th bit of SCLi	
Data Transfer Timing from the UART Receive Shift Register to the UiRB Register	CKPOL=0(rising edge) CKPOL=1(falling edge)	Rising edge of 9th bit of SCLi		Falling edge of 9th bit of SCLi	Falling edge and rising edge of 9th bit of SCLi
UARTi Transmit Output Delay	No delay	Delay			
$P6_3$, $P6_7$, $P7_0$, $P9_2$, $P9_6$ Pin Functions	TxDi output	SDAi input and output			
$P6_2$, $P6_6$, $P7_1$, $P9_1$, $P9_7$ Pin Functions	RxDi input	SCLi input and output			
$P6_1$, $P6_5$, $P7_2$, $P9_0$, $P9_5$ Pin Functions	Select CLKi input or output	– (Not used in I^2C mode)			
Noise Filter Width	15ns	200ns			
Reading RxDi and SCLi Pin Levels	Can be read if port direction bit is set to "0"	Can be read regardless of the port direction bit			
Default Value of TxDi, SDAi Output	CKPOL=0 (H) CKPOL=1 (L)	Values set in the port register before entering I^2C mode[2]			
SCLi Default and End Value	–	H	L	H	L
DMA Generated (See Figure 17.20)	UARTi reception	Acknowledgement detection (ACK)		UARTi Reception - Falling edge of 9th bit of SCLi	
Store Received Data	1st to 8th bits of the received data are stored into bits 7 to 0 in the UiRB register	1st to 8th bits of the received data are stored into bits 7 to 0 in the UiRB register		1st to 7th bits of the received data are stored into bits 6 to 0 in the UiRB register. 8th bit is stored into bit 8 in the UiRB register.	1st to 8th bits are stored into bits 7 to 0 in the UiRB register[3]
Reading Received Data	The UiRB register status is read				Bits 6 to 0 in the UiRB registerts[4] are read as bit 7 to 1. Bit 8 in the UiRB register is read as bit 0

i=0 to 4

NOTES:

1. Use the following procedure to change what causes an interrupt to be generated.
   (a) Disable interrupt of corresponding interrupt number.
   (b) Change what causes an interrupt to be generated.
   (c) Set the IR bit of a corresponding interrupt number to "0" (no interrupt requested).
   (d) Set the ILVL2 to ILVL0 bits of a corresponding interrupt number.
2. Set default value of the SDAi output when the SMD2 to SMD0 bits in the UiMR register are set to "000_2" (serial I/O disabled).
3. Second data transfer to the UiRB register (on the rising edge of the ninth bit of SCLi).
4. First data transfer to the UiRB register (on the falling edge of the ninth bit of SCLi).


(1) When the IICM2 bit is set to "0" (ACK or NACK interrupt) and the CKPH bit is set to "0" (No clock delay)

(2) When the IICM2 bit is set to "0" and the CKPH bit is set to "1" (clock delay)

(3) When the IICM2 bit is set to "1" (UART transmit or receive interrupt) and the CKPH bit is set to "0"

(4) When the IICM2 bit is set to "1" and the CKPH bit is set to "1"

1st bit
2nd bit
3rd bit
4th bit
5th bit
6th bit
7th bit
8th bit
9th bit
SCLi
SDAi
D7
D6
D5
D4
D3
D2
D1
D0
D8 (ACK or NACK)
ACK interrupt (DMA request) or NACK interrupt
Receive interrupt (DMA request)
Transmit interrupt
Data is transferred to the UiRB register
b15
b9
b8
b7
b0
D8 D7 D6 D5 D4 D3 D2 D1 D0
D0 − D7 D6 D5 D4 D3 D2 D1
Contents of the UiRB register

i=0 to 4
IICM2 : Bit in the UiSMR2 register
CKPH : Bit in the UiSMR3 regiser

The above timing applies to the following setting :
- The CKDIR bit in the UiMR register is set to "1" (slave)

Figure 17.20 SCLi Timing

Table 17.15 Pin Settings in I²C Mode (1)

Port	Function	Setting		
		PS0 Register	PSL0 Register	PD6 Register
P62	SCL0 output	PS0_2=1	PSL0_2=0	-
	SCL0 input	PS0_2=0	-	PD6_2=0
P63	SDA0 output	PS0_3=1	-	-
	SDA0 input	PS0_3=0	-	PD6_3=0
P66	SCL1 output	PS0_6=1	PSL0_6=0	-
	SCL1 input	PS0_6=0	-	PD6_6=0
P67	SDA1 output	PS0_7=1	-	-
	SDA1 input	PS0_7=0	-	PD6_7=0

Table 17.16 Pin Settings (2)

Port	Function	Setting			
		PS1 Register	PSL1 Register	PSC Register	PD7 Register
P70[1]	SDA2 output	PS1_0=1	PSL1_0=0	PSC_0=0	–
	SDA2 input	PS1_0=0	–	–	PD7_0=0
P71[1]	SCL2 output	PS1_1=1	PSL1_1=1	PSC_1=0	–
	SCL2 input	PS1_1=0	–	–	PD7_1=0

NOTES:

1. P70 and P71 are ports for the N-channel open drain output.

Table 17.17 Pin Settings (3)

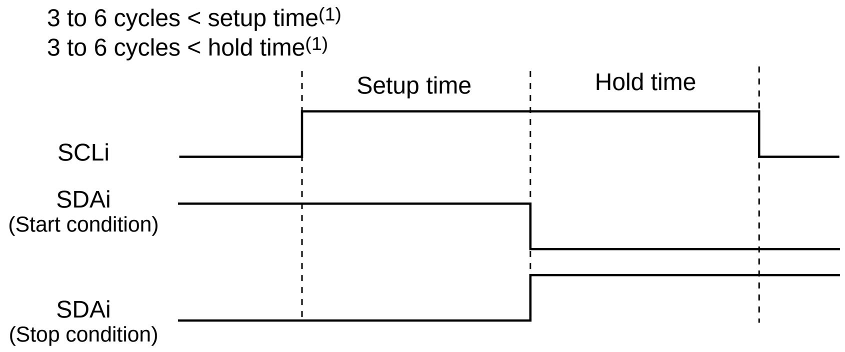
Port	Function	Setting			
		PS3 Register[1]	PSL3 Register	PSC3 Register	PD9 Register[1]
P91	SCL3 output	PS3_1=1	PSL3_1=0	-	-
	SCL3 input	PS3_1=0	-	-	PD9_1=0
P92	SDA3 output	PS3_2=1	PSL3_2=0	-	-
	SDA3 input	PS3_2=0	-	-	PD9_2=0
P96	SDA4 output	PS3_6=1	-	PSC3_6=0	-
	SDA4 input	PS3_6=0	-	-	PD9_6=0
P97	SCL4 output	PS3_7=1	PSL3_7=0	-	-
	SCL4 input	PS3_7=0	-	-	PD9_7=0

NOTES:

1. Set the PD9 and PS3 registers immediately after the PRC2 bit in the PRCR register is set to "1" (write enable). Do not generate an interrupt or a DMA transfer between the instruction to set to the PRC2 bit to "1" and the instruction to set the PD9 and PS3 registers.

17.3.1 Detecting Start Condition and Stop Condition

The microcomputer detects either a start condition or stop condition. The start condition detect interrupt is generated when the SCLi (i=0 to 4) pin level is held high ("H") and the SDAi pin level changes "H" to low ("L"). The stop condition detect interrupt is generated when the SCLi pin level is held "H" and the SDAi pin level changes "L" to "H". The start condition detect interrupt shares interrupt control registers and vectors with the stop condition detect interrupt. The BBS bit in the UiSMR register determines which interrupt is requested.



i=0 to 4
NOTES:
1. These cycles are main clock generation frequency cycles (XIN).

Figure 17.21 Start Condition or Stop Condition Detecting

17.3.2 Start Condition or Stop Condition Output

The start condition is generated when the STAREQ bit in the UiSMR4 register (i=0 to 4) is set to "1" (start). The restart condition is generated when the RSTAREQ bit in the UiSMR4 register is set to "1" (start). The stop condition is generated the STPREQ bit in the UiSMR4 is set to "1" (start).

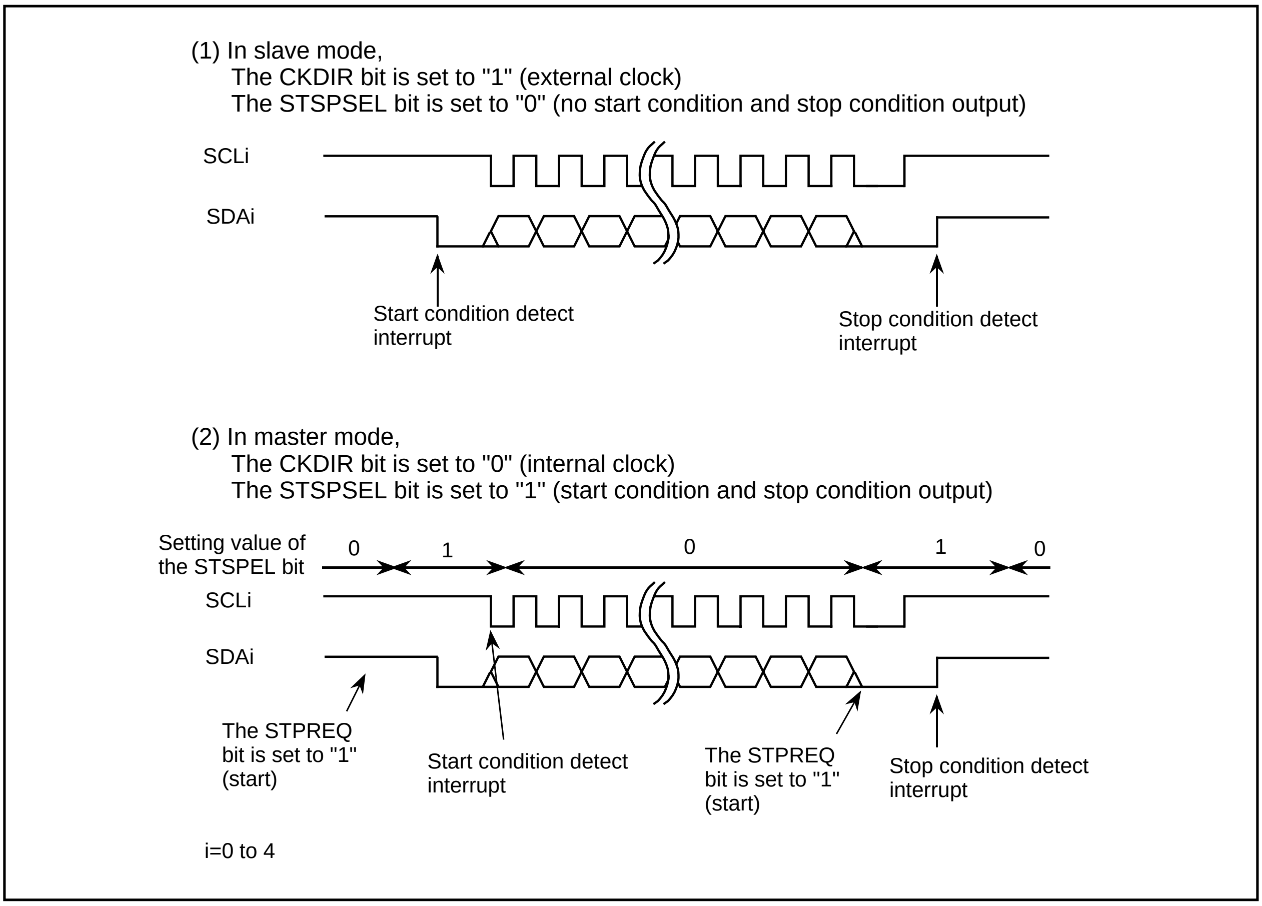
The start condition is output when the STAREQ bit is set to "1" and the STSPSEL bit in the UiSMR4 register is set to "1" (start or stop condition generating circuit selected). The restart condition output is provided when the RSTAREQ bit and STSPSEL bit are set to "1". The stop condition output is provided when the STPREQ bit and the STSPSEL bit are set to "1".

When the start condition, stop condition or restart condition is output, do not generate an interrupt between the instruction to set the STAREQ bit, STPREQ bit or RSTAREQ bit to "1" and the instruction to set the STSPSEL bit to "1". When the start condition is output, set the STAREQ bit to "1" before the STSPSEL bit is set to "1".

Table 17.18 lists function of the STSPSEL bit. Figure 17.22 shows functions of the STSPSEL bit.

Table 17.18 STSPSEL Bit Function

Function	STSPSEL = 0	STSPSEL = 1
Start condition and stop condition output	Program with ports determines how the start condition or stop condition output is provided	The STAREQ bit, RSTAREQ bit and STPREQ bit determine how the start condition or stop condition output is provided
Timing to generate start condition and stop condition interrupt requests	The start condition and stop condition are detected	Start condition and stop condition generation are completed



(1) In slave mode,
The CKDIR bit is set to "1" (external clock)
The STSPSEL bit is set to "0" (no start condition and stop condition output)
SCLi
SDAi
Start condition detect interrupt
Stop condition detect interrupt
(2) In master mode,
The CKDIR bit is set to "0" (internal clock)
The STSPSEL bit is set to "1" (start condition and stop condition output)
Setting value of the STSPEL bit
0
1
0
1
0
SCLi
SDAi
The STPREQ bit is set to "1" (start)
Start condition detect interrupt
The STPREQ bit is set to "1" (start)
Stop condition detect interrupt
i=0 to 4

Figure 17.22 STSPSEL Bit Function

17.3.3 Arbitration

The ABC bit in the UiSMR register (i=0 to 4) determines an update timing for the ABT bit in the UiRB register. On the rising edge of the SCLi pin, the microcomputer determines whether a transmit data matches data input to the SDAi pin.

When the ABC bit is set to "0" (update per bit), the ABT bit is set to "1" (detected-arbitration is lost) as soon as a data discrepancy is detected. The ABT bit is set to "0" (not detected-arbitration is won) if not detected. When the ABC bit is set to "1" (update per byte), the ABT bit is set to "1" on the falling edge of the ninth bit of the transfer clock if any discrepancy is detected. When the ABT bit is updated per byte, set the ABT bit to "0" between an ACK detection in the first byte data and the next byte data to be transferred.

When the ALS bit in the UiSMR2 register is set to "1" (SDA output stop enabled), the arbitration lost occurs. As soon as the ABT bit is set to "1", the SDAi pin is placed in a high-impedance state.

17.3.4 Transfer Clock

The transfer clock transmits and receives data as is shown in Figure 17.20.

The CSC bit in the UiSMR2 register (i=0 to 4) synchronizes an internally generated clock (internal SCLi) with the external clock applied to the SCLi pin. When the CSC bit is set to "1" (clock synchronous enabled) and the internal SCLi is held high ("H"), the internal SCLi become low ("L") if signal applied to the SCLi pin is on the falling edge. Value of the UiBRG register is reloaded to start counting for low level. A counter stops when the SCLi pin is held "L" and then the internal SCLi changes "L" to "H". Counting is resumed when the SCLi pin become "H". The transfer clock of UARTi is equivalent to the AND for signals from the internal SCLi and the SCLi pin.

The transfer clock is synchronized between a half cycle before the falling edge of first bit of the internal SCLi and the rising edge of the ninth bit. Select the internal clock as the transfer clock while the CSC bit is set to "1".

The SWC bit in the UiSMR2 register determines whether the SCLi pin is fixed to be an "L" signal output on the falling edge of the ninth cycle of the transfer clock or not.

When the SCLHI bit in the UiSMR4 register is set to "1" (enabled), a SCLi output stops when a stop condition is detected (high-impedance).

When the SWC2 bit in the UiSMR2 register is set to "1" (0 output), the SCLi pin focibly outputs an "L" signal while transmitting and receiving. The fixed "L" signal applied to the SCLi pin is cancelled by setting the SWC2 bit to "0" (transfer clock) and the transfer clock input to and output from the SCLi pin are provided.

When the CKPH bit in the UiSMR3 register is set to "1" and the SWC9 bit in the UiSMR4 register is set to "1" (SCL "L" hold enabled), the SCLi pin is fixed to be an "L" signal output on the next falling edge after the ninth bit of the clock. The fixed "L" signal applied to the SCLi pin is cancelled by setting the SWC9 bit to "0" (SCL "L" hold disabled).

17.3.5 SDA Output

Values output set in bits 7 to 0 (D_7 to D_0) in the UiTB register (i=0 to 4) are provided in descending order from D_7. The ninth bit (D_8) is ACK or NACK.

Set the default value of SDAi transmit output when the IICM bit is set to "1" (I^2C mode) and the SMD2 to SMD0 bits in the UiMR register are set to "000_2" (serial I/O disabled).

The DL2 to DL0 bits in the UiSMR3 register determine no delay in the SDAi output or a delay of 2 to 8 UiBRG register count source cycles.

When the SDHI bit in the UiSMR2 register is set to "1" (SDA output disabled), the SDAi pin is forcibly placed in a high-impedance state. Do not set the SDHI bit on the rising edge of the UARTi transfer clock. The ABT bit in the UiRB register may be set to "1" (detected).

17.3.6 SDA Input

When the IICM2 bit in the UiSMR2 register (i=0 to 4) is set to "0", the first eight bits of received data are stored into bits 7 to 0 (D_7 to D_0) in the UiRB register. The ninth bit (D_8) is ACK or NACK.

When the IICM2 bit is set to "1", the first seven bits (D_7 to D_1) of received data are stored into bits 6 to 0 in the UiRB register. Store the eighth bit (D_0) into bit 8 in the UiRB register.

If the IICM2 bit is set to "1" and the CKPH bit in the UiSMR3 register is set to "1", the same data as that of when setting the IICM2 bit to "0" can be read. To read the data, read the UiRB register after the rising edge of the ninth bit of the transfer clock.

17.3.7 ACK, NACK

When the STSPSEL bit in the UiSMR4 register (i=0 to 4) is set to "0" (serial I/O circuit selected) and the ACKC bit in the UiSMR4 register is set to "1" (ACK data output), the SDAi pin provides the value output set in the ACKD bit in the UiSMR4 register.

If the IICM2 bit is set to "0", the NACK interrupt request is generated when the SDAi pin is held high ("H") on the rising edge of the ninth bit of the transfer clock. The ACK interrupt request is generated when the SDAi pin is held low ("L") on the rising edge of the ninth bit of the transfer clock.

When ACK is selected to generate a DMA request, the DMA transfer is activated by an ACK detection.

17.3.8 Transmit and Receive Reset

When the STC bit in the UiSMR2 register (i=0 to 4) is set to "1" (UARTi initialization enabled) and a start condition is detected,

- the transmit shift register is reset and the content of the UiTB register is transferred to the transmit shift register. The first bit starts transmitting when the next clock is input. UARTi output value remains unchanged between when the clock is applied and when the first bit data output is provided. The value remains the same as when start condition was detected.
- the receive shift register is reset and the first bit start receiving when the next clock is applied.
- the SWC bit is set to "1" (SCL wait output enabled). The SCLi pin becomes "L" on the falling edge of the ninth bit of the transfer clock.

If UARTi transmission and reception are started with this function, the TI bit in the UiC1 register remains unchanged. Select the external clock as the transfer clock when using this function.

17.4 Special Mode 2

In special mode 2, serial communication between one or multiple masters and multiple slaves is available. The $\overline{\text{SSi}}$ input pin (i=0 to 4) controls the serial bus communication. Table 17.19 lists specifications of special mode 2. Table 17.20 lists register settings. Tables 17.21 to 17.23 list pin settings.

Table 17.19 Special Mode 2 Specifications

Item	Specification
Transfer Data Format	Transfer data : 8 bits long
Transfer Clock	• The CKDIR bit in the UiMR register (i=0 to 4) is set to "0" (internal clock selected): $fj/2(m+1)$ fj = f_1, f_8, f_{2n}(1) m : setting value of the UiBRG register, 00_{16} to FF_{16} • The CKDIR bit to "1" (external clock selected) : input from the CLKi pin
Transmit/Receive Control	$\overline{\text{SSi}}$ input pin function
Transmit Start Condition	To start transmitting, the following requirements must be met(2): - Set the TE bit in the UiC1 register to "1" (transmit enable) - Set the TI bit in the UiC1 register to "0" (data in the UiTB register)
Receive Start Condition	To start receiving, the following requirement must be met(2): - Set the RE bit in the UiC1 register to "1" (receive enable) - Set the TE bit in the UiC1 register to "1" (transmit enable) - Set the TI bit in the UiC1 register to "0" (data in the UiTB register)
Interrupt Request Generation Timing	• While transmitting, the following conditions can be selected: - The UiIRS bit in the UiC1 register is set to "0" (no data in a transmit buffer) : when data is transferred from the UiTB register to the UARTi transmit register (transmission started) - The UiIRS register is set to "1" (transmission completed): when data transmission from UARTi transfer register is completed • While receiving When data is transferred from the UARTi receive register to the UiRB register (reception completed)
Error Detection	• Overrun error(3) This error occurs when the seventh bit of the next received data is read before reading the UiRB register • Fault error In master mode, the fault error occurs an "L" signal is applied to the $\overline{\text{SSi}}$ pin
Selectable Function	• CLK polarity Select from the rising edge or falling edge of the transfer clock when transferred data is output and input are provided • LSB first or MSB first Data is transmitted or received in either bit 0 or in bit 7 • Continuous receive mode Reception is enabled simultaneously by reading the UiRB register • Serial data logic inverse This function inverses transmitted or received data logically • TxD and RxD I/O polarity inverse TxD pin output and RxD pin input are inversed. All I/O data levels are also inversed • Clock phase Select from one of 4 combinations of transfer data polarity and phases • $\overline{\text{SSi}}$ input pin function Output pin is placed in a high-impedance state to avoid data conflict between master and other masters or slaves

NOTES:

1. The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2n (n=1 to 15).
2. To start transmission/reception when selecting the external clock, these conditions must be met after the CKPOL bit in the UiC0 register is set to "0" (data is transmitted on the falling edge of the transfer clock and data is received on the rising edge) and the CLKi pin is held high ("H"), or when the CKPOL bit is set to "1" (Data is transmitted on the rising edge of the transfer clock and data is received on the falling edge) and the CLKi pin is held low ("L").
3. If an overrun error occurs, the UiRB register is in an indeterminate state. The IR bit in the SiRIC register does not change to "1" (interrupt requested).

Table 17.20 Register Settings in Special Mode 2

Register	Bit	Function
UiTB	7 to 0	Set transmit data
UiRB	7 to 0	Received data can be read
	OER	Overrun error flag
UiBRG	7 to 0	Set bit rate
UiMR	SMD2 to SMD0	Set to "001_2"
	CKDIR	Set to "0" in master mode or "1" in slave mode
	IOPOL	Set to "0"
UiC0	CLK1, CLK0	Select count source for the UiBRG register
	CRS	Disabled because the CRD bit is set to "1"
	TXEPT	Transfer register empty flag
	CRD	Set to "1"
	NCH	Select the output format of the TxDi pin
	CKPOL	Clock phase can be set by the combination of the CKPOL bit and the CKPH bit in the UiSMR3 register
	UFORM	Select either LSB first or MSB first
UiC1	TE	Set to "1" to enable data transmission and reception
	TI	Transfer buffer empty flag
	RE	Set to "1" to enable data reception
	RI	Reception complete flag
	UiIRS	Select what causes the UARTi transmit interrupt to be generated
	UiRRM	Set to "1" to enable continuous receive mode
	UiLCH, SCLKSTPB	Set to "0"
UiSMR	7 to 0	Set to "00_{16}"
UiSMR2	7 to 0	Set to "00_{16}"
UiSMR3	SSE	Set to "1"
	CKPH	Clock phase can be set by the combination of the CKPH bit and the CKPOL bit in the UiC0 register
	DINC	Set to "0" in master mode or "1" in slave mode
	NODC	Set to "0"
	ERR	Fault error flag
	7 to 5	Set to "000_2"
UiSMR4	7 to 0	Set to "00_{16}"

i=0 to 4

Table 17.21 Pin Settings in Special Mode 2 (1)

Port	Function	Setting		
		PS0 Register	PSL0 Register	PD6 Register
P6$_0$	$\overline{\text{SS0}}$ input	PS0_0=0	–	PD6_0=0
P6$_1$	CLK0 input (slave)	PS0_1=0	–	PD6_1=0
	CLK0 output (master)	PS0_1=1	–	–
P6$_2$	RxD0 input (master)	PS0_2=0	–	PD6_2=0
	STxD0 output (slave)	PS0_2=1	PSL0_2=1	–
P6$_3$	TxD0 output (master)	PS0_3=1	–	–
	SRxD0 input (slave)	PS0_3=0	–	PD6_3=0
P6$_4$	$\overline{\text{SS1}}$ input	PS0_4=0	–	PD6_4=0
P6$_5$	CLK1 input (slave)	PS0_5=0	–	PD6_5=0
	CLK1 output (master)	PS0_5=1	–	–
P6$_6$	RxD1 input (master)	PS0_6=0	–	PD6_6=0
	STxD1 output (slave)	PS0_6=1	PSL0_6=1	–
P6$_7$	TxD1 output (master)	PS0_7=1	–	–
	SRxD1 input (slave)	PS0_7=0	–	PD6_7=0

Table 17.22 Pin Settings (2)

Port	Function	Setting			
		PS1 Register	PSL1 Register	PSC Register	PD7 Register
P7$_0$(1)	TxD2 output (master)	PS1_0=1	PSL1_0=0	PSC_0=0	–
	SRxD2 input (slave)	PS1_0=0	–	–	PD7_0=0
P7$_1$(1)	RxD2 input (master)	PS1_1=0	–	–	PD7_1=0
	STxD2 output (slave)	PS1_1=1	PSL1_1=1	PSC_1=0	–
P7$_2$	CLK2 input (slave)	PS1_2=0	–	–	PD7_2=0
	CLK2 output (master)	PS1_2=1	PSL1_2=0	PSC_2=0	–
P7$_3$	$\overline{\text{SS2}}$ input	PS1_3=0	–	–	PD7_3=0

NOTES:

1. P7$_0$ and P7$_1$ are ports for the N-channel open drain output.

Table 17.23 Pin Settings (3)

Port	Function	Setting			
		PS3 Register(1)	PSL3 Register	PSC3 Register	PD9 Register(1)
P9$_0$	CLK3 input (slave)	PS3_0=0	–	–	PD9_0=0
	CLK3 output (master)	PS3_0=1	–	–	–
P9$_1$	RxD3 input (master)	PS3_1=0	–	–	PD9_1=0
	STxD3 output (slave)	PS3_1=1	PSL3_1=1	–	–
P9$_2$	TxD3 output (master)	PS3_2=1	PSL3_2=0	–	–
	SRxD3 input (slave)	PS3_2=0	–	–	PD9_2=0
P9$_3$	$\overline{\text{SS3}}$ input	PS3_3=0	PSL3_3=0	–	PD9_3=0
P9$_4$	$\overline{\text{SS4}}$ input	PS3_4=0	PSL3_4=0	–	PD9_4=0
P9$_5$	CLK4 input (slave)	PS3_5=0	PSL3_5=0	–	PD9_5=0
	CLK4 output (master)	PS3_5=1	–	–	–
P9$_6$	TxD4 output (master)	PS3_6=1	–	PSC3_6=0	–
	SRxD4 input (slave)	PS3_6=0	PSL3_6=0	–	PD9_6=0
P9$_7$	RxD4 input (master)	PS3_7=0	–	–	PD9_7=0
	STxD4 output (slave)	PS3_7=1	PSL3_7=1	–	–

NOTES:

1. Set the PD9 and PS3 registers immediately after the PRC2 bit in the PRCR register is set to "1" (write enable). Do not generate an interrupt or a DMA transfer between the instruction to set to the PRC2 bit to "1" and the instruction to set the PD9 and PS3 registers.

17.4.1 $\overline{\text{SSi}}$ Input Pin Function (i=0 to 4)

When the SSE bit in the UiSMR3 register is set to "1" ($\overline{\text{SS}}$ function enabled), the special mode 2 is selected, activating the pin function.

The DINC bit in the UiSMR3 register determines which microcomputer performs as master or slave. When multiple microcomputers perform as the masters (multi-master system), the $\overline{\text{SSi}}$ pin setting determines which master microcomputer is active and when.

17.4.1.1 When Setting the DINC Bit to "1" (Slave Mode)

When a high-level ("H") signal is applied to the $\overline{\text{SSi}}$ pin, the STxDi and SRxDi pins are placed in a high-impedance state and the transfer clock applied to the CLKi pin is ignored. When a low-level ("L") signal is applied to the $\overline{\text{SSi}}$ input pin, the transfer clock input is valid and serial communication is enabled.

17.4.1.2 When Setting the DINC Bit to "0" (Master Mode)

When using the $\overline{\text{SSi}}$ pin functin in master mode, set the UiIRS bit in the UiC1 register to "1" (transmission completed).

When an "H" signal is applied to the $\overline{\text{SSi}}$ pin, serial communication is available due to transmission privilege. The master provides the transfer clock output. When an "L" signal is applied to the $\overline{\text{SSi}}$ pin, it indicates that another master is active. The TxDi and CLKi pins are placed in high-impedance states and the ERR bit in the UiSMR3 register is set to "1" (fault error) Use the transmit complete interrupt routine to verify the ERR bit state.

To resume the serial communication after the fault error occurs, set the ERR bit to "0" while applying the "H" signal to the $\overline{\text{SSi}}$ pin. The TxDi and CLKi pins become ready for signal outputs.

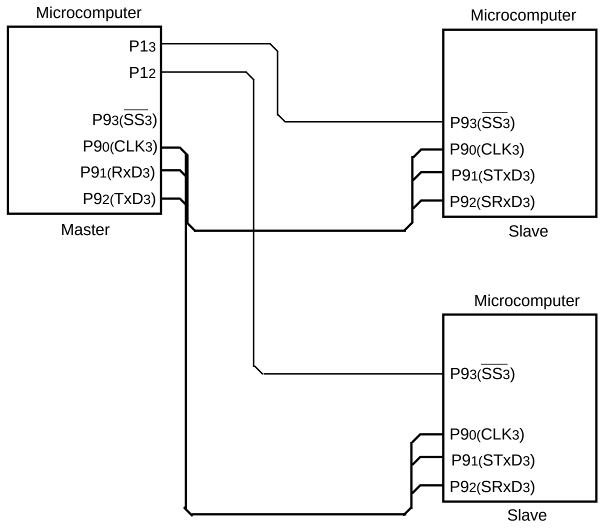


Figure 17.23 Serial Bus Communication Control with $\overline{\text{SS}}$ Pin

17.4.2 Clock Phase Setting Function

The CKPH bit in the UiSMR3 register (i=0 to 4) and the CKPOL bit in the UiC0 register select one of four combinations of transfer clock polarity and phases.

The transfer clock phase and polarity must be the same between the master and the slave involved in the transfer.

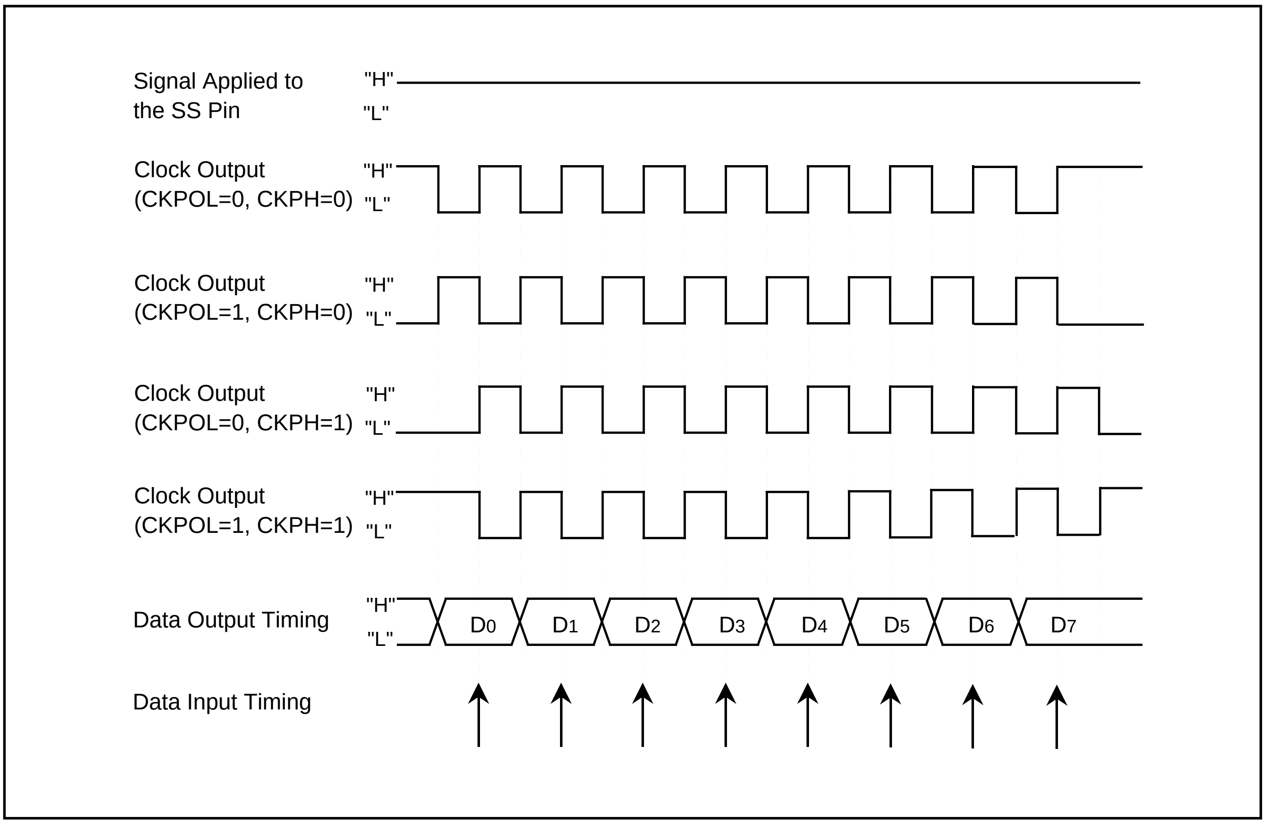
17.4.2.1 When setting the DINC Bit to "0" (Master (Internal Clock))

Figure 17.24 shows transmit and receive timing.

17.4.2.2 When Setting the DINC Bit to "1" (Slave (External Clock))

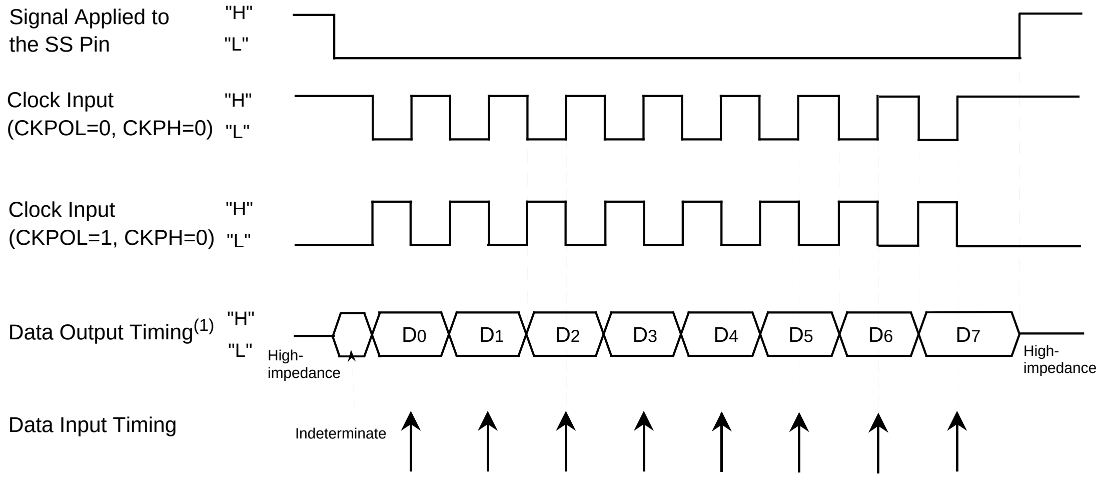
When the CKPH bit is set to "0" (no clock delay) and the $\overline{\text{SSi}}$ input pin is held high ("H"), the STxDi pin is placed in a high-impedance state. When the $\overline{\text{SSi}}$ input pin becomes low ("L"), conditions to start a serial transfer are met, but output is indeterminate. The serial transmission is synchronized with the transfer clock. Figure 17.25 shows the transmit and receive timing.

When the CKPH bit is set to "1" (clock delay) and the $\overline{\text{SSi}}$ input pin is held high, the STxDi pin is placed in a high-impedance state. When the $\overline{\text{SSi}}$ pin becomes low, the first data is output. The serial transmission is synchronized with the transfer clock. Figure 17.26 shows the transmit and receive timing.



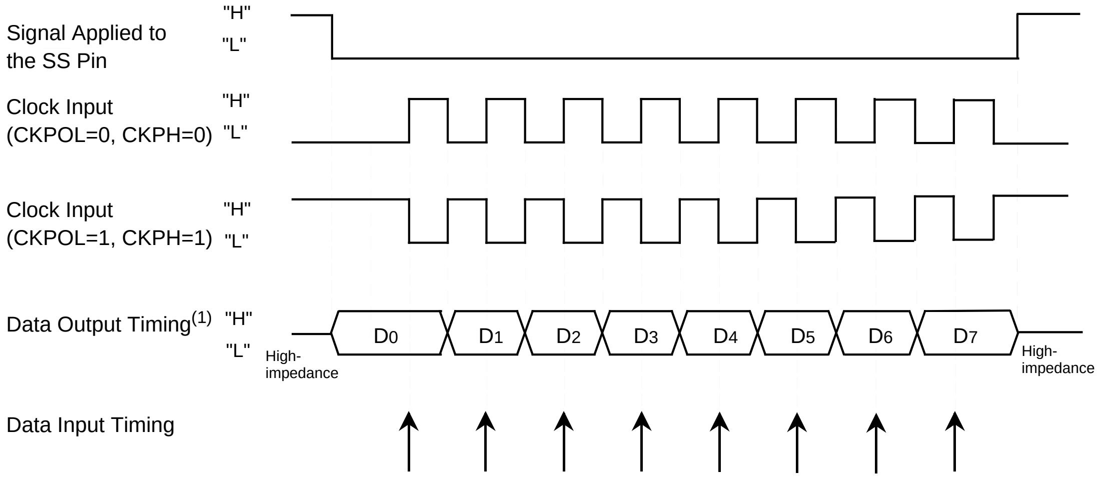
Signal Applied to the SS Pin
"H"
"L"
Clock Output (CKPOL=0, CKPH=0)
"H"
"L"
Clock Output (CKPOL=1, CKPH=0)
"H"
"L"
Clock Output (CKPOL=0, CKPH=1)
"H"
"L"
Clock Output (CKPOL=1, CKPH=1)
"H"
"L"
Data Output Timing
"H"
"L"
D0
D1
D2
D3
D4
D5
D6
D7
Data Input Timing

Figure 17.24 Transmit and Receive Timing in Master Mode (Internal Clock)



NOTES:
1. P70 and P71 is ports for the N-channel open drain output and must be pulled up externally for data output.

Figure 17.25 Transmit and Receive Timing in Slave Mode (External Clock) (CKPH=0)



NOTES:
1. P70 and P71 are ports for the N-channel open drain output and must be pulled up externally for data output.

Figure 17.26 Transmit and Receive Timing in Slave Mode (External Clock) (CKPH=1)

17.5 Special Mode 3 (GCI Mode)

In GCI mode, the external clock is synchronized with the transfer clock used in the clock synchronous serial I/O mode.

Table 17.24 lists specifications of GCI mode. Table 17.25 lists registers settings. Tables 17.26 to 17.28 list pin settings.

Table17.24 GCI Mode Specifications

Item	Specification
Transfer Data Format	Transfer data : 8 bits long
Transfer Clock	The CKDIR bit in the UiMR register (i=0 to 4) is set to "1" (external clock selected): input from the CLKi pin
Clock Synchronization Function	Trigger signal input from the $\overline{\text{CTSi}}$ pin
Transmit/Receive Start Condition	To start data transmission and reception, meet the following conditions and then apply a trigger signal to the $\overline{\text{CTSi}}$ pin: - Set the TE bit in the UiC1 register to "1" (transmit enable) - Set the RE bit in the UiC1 register to "1" (receive enable) - Set the TI bit in the UiC1 register to "0" (Data in the UiTB register)
Interrupt Request Generation Timing	• While transmitting, the following condition can be selected: - The UiIRS bit in the UiC1 register is set to "0" (UiTB register empty): when data is transferred from the UiTB register to the UARTi transmit register (transmission started) - The UiIRS bit is set to "1" (Transmit completed): when a data transmission from the UARTi transfer register is completed • While receiving, when data is transferred from the UARTi receive register to the UiRB register (reception completed)
Error Detection	Overrun error[1] This error occurs when the seventh bit of the next received data is read before reading the UiRB register.

NOTES:

1. If an overrun error occurs, the UiRB register is indeterminate. The IR bit in the SiRIC register does not change to "1" (interrupt requested).

Table 17.25 Register Settings in GCI Mode

Register	Bit	Function
UiTB	7 to 0	Set transmit data
UiRB	7 to 0	Received data
	OER	Overrun error flag
UiBRG	7 to 0	Set to "00_{16}"
UiMR	SMD2 to SMD0	Set to "001_{2}"
	CKDIR	Set to "1"
	IOPOL	Set to "0"
UiC0	CLK1, CLK0	Set to "00_{2}"
	CRS	Disabled because the CRD bit is set to "1"
	TXEPT	Transfer register empty flag
	CRD	Set to "1"
	NCH	Select the output format of the TxDi pin
	CKPOL	Set to "0"
	UFORM	Set to "0"
UiC1	TE	Set to "1" to enable data transmission and reception
	TI	Transfer buffer empty flag
	RE	Set to "1" to enable data reception
	RI	Reception complete flag
	UiIRS	Select what causes the UARTi transmit interrupt to be generated
	UiRRM, UiLCH	Set to "0"
	SCLKSTPB	Set to "0"
UiSMR	6 to 0	Set to "0000000_{2}"
	SCLKDIV	See Table 17.29
UiSMR2	6 to 0	Set to "0000000_{2}"
	SU1HIM	See Table 17.29
UiSMR3	2 to 0	Set to "000_{2}"
	NODC	Set to "0"
	7 to 4	Set to "0000_{2}"
UiSMR4	7 to 0	Set to "00_{16}"

i=0 to 4

Table 17.26 Pin Settings in GCI Mode (1)

Port	Function	Setting	
		PS0 Register	PD6 Register
P6$_0$	$\overline{\text{CTS0}}$ input(1)	PS0_0=0	PD6_0=0
P6$_1$	CLK0 input	PS0_1=0	PD6_1=0
P6$_2$	RxD0 input	PS0_2=0	PD6_2=0
P6$_3$	TxD0 output	PS0_3=1	–
P6$_4$	$\overline{\text{CTS1}}$ input(1)	PS0_4=0	PD6_4=0
P6$_5$	CLK1 input	PS0_5=0	PD6_5=0
P6$_6$	RxD1 input	PS0_6=0	PD6_6=0
P6$_7$	TxD1 output	PS0_7=1	–

NOTES:

1. $\overline{\text{CTS}}$ input is used as a trigger siganl input.

Table 17.27 Pin Settings (2)

Port	Function	Setting			
		PS1 Register	PSL1 Register	PSC Register	PD7 Register
P7$_0$(1)	TxD2 output	PS1_0=1	PSL1_0=0	PSC_0=0	–
P7$_1$(1)	RxD2 input	PS1_1=0	–	–	PD7_1=0
P7$_2$	CLK2 input	PS1_2=0	–	–	PD7_2=0
P7$_3$	$\overline{\text{CTS2}}$ input(2)	PS1_3=0	–	–	PD7_3=0

NOTES:

1. P7$_0$ and P7$_1$ are ports for the N-channel open drain output.
2. $\overline{\text{CTS}}$ input is used as a trigger siganl input.

Table 17.28 Pin Settings (3)

Port	Function	Setting			
		PS3 Register(1)	PSL3 Register	PSC3 Register	PD9 Register(1)
P9$_0$	CLK3 input	PS3_0=0	–	–	PD9_0=0
P9$_1$	RxD3 input	PS3_1=0	–	–	PD9_1=0
P9$_2$	TxD3 output	PS3_2=1	PSL3_2=0	–	–
P9$_3$	$\overline{\text{CTS3}}$ input(2)	PS3_3=0	PSL3_3=0	–	PD9_3=0
P9$_4$	$\overline{\text{CTS4}}$ input(2)	PS3_4=0	PSL3_4=0	–	PD9_4=0
P9$_5$	CLK4 input	PS3_5=0	PSL3_5=0	–	PD9_5=0
P9$_6$	TxD4 output	PS3_6=1	PSL3_6=0	PSC3_6=0	–
P9$_7$	RxD4 input	PS3_7=0	–	–	PD9_7=0

NOTES:

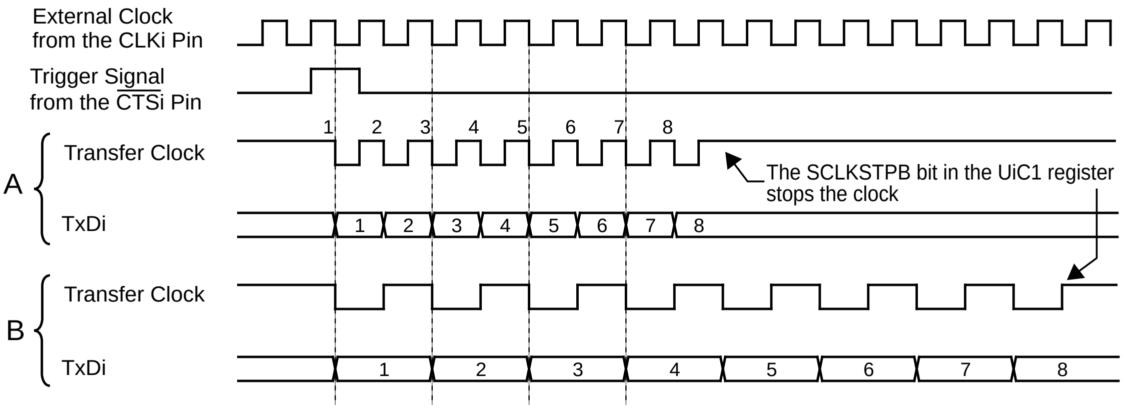
1. Set the PD9 and PS3 registers immediately after the PRC2 bit in the PRCR register is set to "1" (write enable). Do not generate an interrupt or a DMA transfer between the instruction to set to the PRC2 bit to "1" and the instruction to set the PD9 and PS3 registers.
2. $\overline{\text{CTS}}$ input is used for a trigger siganl input.

To generate the internal clock synchronized with the external clock, set the SU1HIM bit in the UiSMR2 register (i=0 to 4) and the SCLKDIV bit in the UiSMR register to values shown in Table 17.29. Then apply a trigger signal to the $\overline{\text{CTSi}}$ pin. Either the same clock cycle as the external clock or external clock divided by two can be selected as the transfer clock. The SCLKSTPB bit in the UiC1 register controls the transfer clock. Set the SCLKSTPB bit accordingly, to start or stop the transfer clock during an external clock operation. Figure 17.27 shows an example of the clock-divided synchronous function.

Table 17.29 Clock-Divided Synchronous Function Select

SCLKDIV Bit in UiSMR Register	SU1HIM Bit in UiSMR2 Register	Clock-Divided Synchronous Function	Example of Waveform
0	0	Not synchronized	-
0	1	Same division as the external clock	A in Figure 17.27
1	0 or 1	Same division as the external clock divided by 2	B in Figure 17.27

i=0 to 4



i=0 to 4
A, B : See Table 17.29.

Figure 17.27 Clock-Divided Synchronous Function

17.6 Special Mode 4 (IE Mode)

In IE mode, devices connected with the IEBus can communicate in UART mode.

Table 17.30 lists register settings. Tables 17.31 to 17.33 list pin settings.

Table 17.30 Register Settings in IE Mode

Register	Bit	Function
UiTB	8 to 0	Set transmit data
UiRB	8 to 0	Received data can be read
	OER, FER, PER, SUM	Error flags
UiBRG	7 to 0	Set bit rate
UiMR	SMD2 to SMD0	Set to "110_2"
	CKDIR	Select the internal clock or external clock
	STPS	Set to "0"
	PRY	Disabled because the PRYE bit is set to "0"
	PRYE	Set to "0"
	IOPOL	Select TxD and RxD I/O polarity
UiC0	CLK1, CLK0	Select count source for the UiBRG register
	CRS	Disabled because the CRD bit is set to "1"
	TXEPT	Transfer register empty flag
	CRD	Set to "1"
	NCH	Select output format of the TxDi pin
	CKPOL	Set to "0"
	UFORM	Set to "0"
UiC1	TE	Set to "1" to enable data transmission
	TI	Transfer buffer empty flag
	RE	Set to "1" te enable data reception
	RI	Reception complete flag
	UiIRS	Select what causes the UARTi transmit interrupt to be generated
	UiRRM, UiLCH, SCLKSTPB	Set to "0"
UiSMR	3 to 0	Set to "0000_2"
	ABSCS	Select bus conflict detect sampling timing
	ACSE	Set to "1" to automatically clear the transmit enable bit
	SSS	Select transmit start condition
	SCLKDIV	Set to "0"
UiSMR2	7 to 0	Set to "00_{16}"
UiSMR3	7 to 0	Set to "00_{16}"
UiSMR4	7 to 0	Set to "00_{16}"
IFSR	IFSR6, IFSR7	Select how the bus conflict interrupt occurs

i=0 to 4

Table 17.31 Pin Settings in IE Mode (1)

Port	Function	Setting		
		PS0 Register	PSL0 Register	PD6 Register
$P6_1$	CLK0 input	PS0_1=0	–	PD6_1=0
	CLK0 output	PS0_1=1	–	–
$P6_2$	RxD0 input	PS0_2=0	–	PD6_2=0
$P6_3$	TxD0 output	PS0_3=1	–	–
$P6_5$	CLK1 input	PS0_5=0	–	PD6_5=0
	CLK1 output	PS0_5=1	–	–
$P6_6$	RxD1 input	PS0_6=0	–	PD6_6=0
$P6_7$	TxD1 output	PS0_7=1	–	–

Table 17.32 Pin Settings (2)

Port	Function	Setting			
		PS1 Register	PSL1 Register	PSC Register	PD7 Register
$P7_0$(1)	TxD2 output	PS1_0=1	PSL1_0=0	PSC_0=0	–
$P7_1$(1)	RxD2 input	PS1_1=0	–	–	PD7_1=0
$P7_2$	CLK2 input	PS1_2=0	–	–	PD7_2=0
	CLK2 output	PS1_2=1	PSL1_2=0	PSC_2=0	–

NOTES:

1. $P7_0$ and $P7_1$ are ports for the N-channel open drain output.

Table 17.33 Pin Settings (3)

Port	Function	Setting			
		PS3 Register(1)	PSL3 Register	PSC3 Register	PD9 Register(1)
$P9_0$	CLK3 input	PS3_0=0	–	–	PD9_0=0
	CLK3 output	PS3_0=1	–	–	–
$P9_1$	RxD3 input	PS3_1=0	–	–	PD9_1=0
$P9_2$	TxD3 output	PS3_2=1	PSL3_2=0	–	–
$P9_5$	CLK4 input	PS3_5=0	PSL3_5=0	–	PD9_5=0
	CLK4 output	PS3_5=1	–	–	–
$P9_6$	TxD4 output	PS3_6=1	–	PSC3_6=0	–
$P9_7$	RxD4 input	PS3_7=0	–	–	PD9_7=0

NOTES:

1. Set the PD9 and PS3 registers immediately after the PRC2 bit in the PRCR register is set to "1" (write enable). Do not generate an interrupt or a DMA transfer between the instruction to set to the PRC2 bit to "1" and the instruction to set the PD9 and PS3 registers.

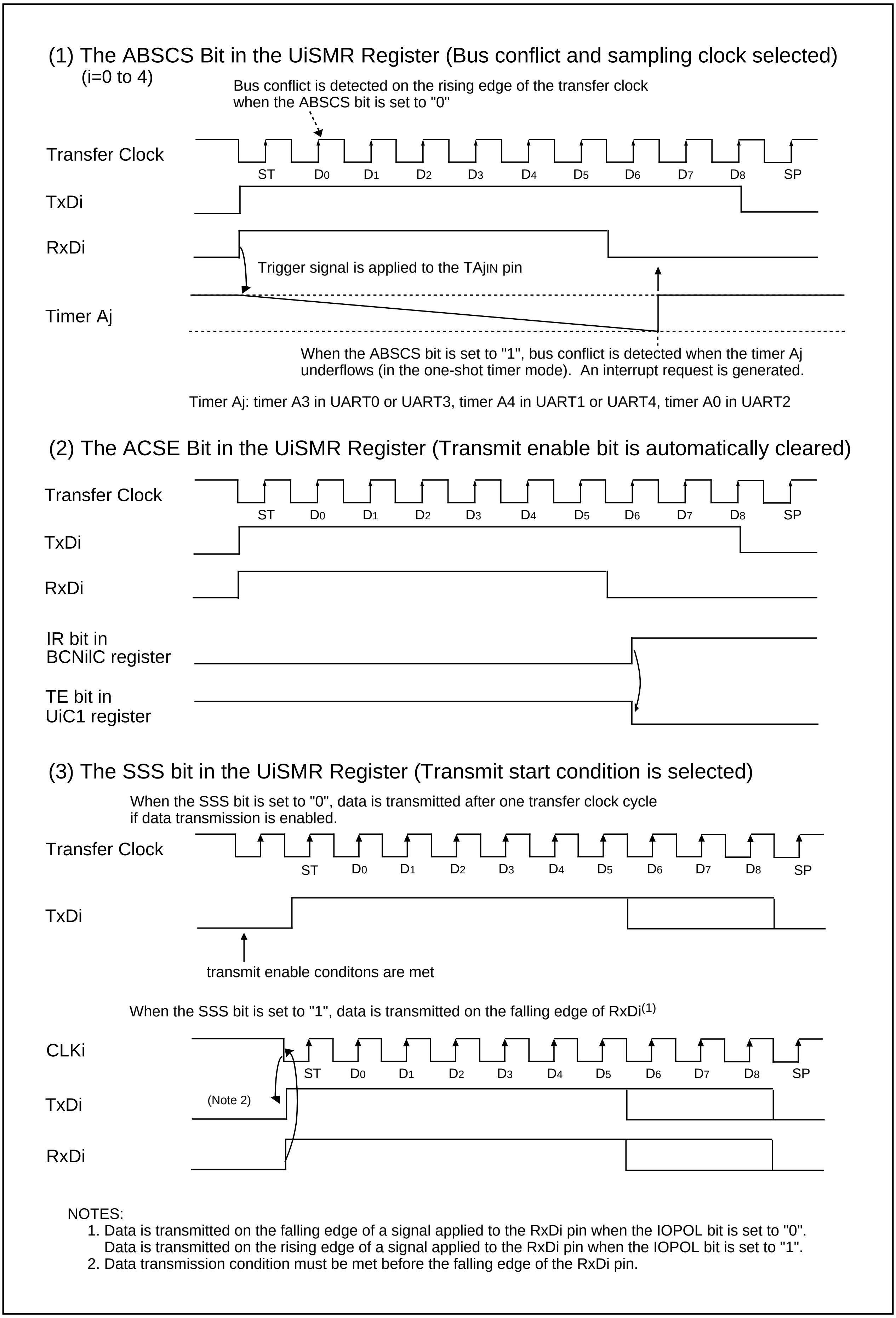
If the output signal level of the TxDi pin (i=0 to 4) differs from the input signal level of the RxDi pin, an interrupt request is generated.

UART0 and UART3 are assigned software interrupt number 40. UART1 and UART4 are assigned number 41. When using the bus conflict detect function of UART0 or UART3, of UART1 or UART4, set the IFSR6 bit and the IFSR7 bit in the IFSR register accordingly.

When the ABSCS bit in the UiSMR register is set to "0" (rising edge of the transfer clock), it is determined, on the rising edge of the transfer clock, if the output level of the TxD pin and the input level of the RxD pin match. When the ABSCS bit is set to "1" (timer Aj underflow), it is determined when the timer Aj (timer A3 in UART0, timer A4 in UART1, timer A0 in UART2, timer A3 in UART3, the timer A4 in UART4) counter overflows. Use the timer Aj in one-shot timer mode.

When the ACSE bit in the UiSMR register is set to "1" (automatic clear at bus conflict) and the IR bit in the BCNiIC register to "1" (discrepancy detected), the TE bit in the UiC1 register is set to "0" (transmit disable).

When the SSS bit in the UiSMR register is set to "1" (synchronized with RxDi), data is transmitted from the TxDi pin on the falling edge of the RxDi pin. Figure 17.28 shows bits associated with the bus conflict detect function.



(1) The ABSCS Bit in the UiSMR Register (Bus conflict and sampling clock selected)
(i=0 to 4)

Bus conflict is detected on the rising edge of the transfer clock when the ABSCS bit is set to "0"

Trigger signal is applied to the TAjIN pin

When the ABSCS bit is set to "1", bus conflict is detected when the timer Aj underflows (in the one-shot timer mode). An interrupt request is generated.

Timer Aj: timer A3 in UART0 or UART3, timer A4 in UART1 or UART4, timer A0 in UART2

(2) The ACSE Bit in the UiSMR Register (Transmit enable bit is automatically cleared)

(3) The SSS bit in the UiSMR Register (Transmit start condition is selected)

When the SSS bit is set to "0", data is transmitted after one transfer clock cycle if data transmission is enabled.

transmit enable conditons are met

When the SSS bit is set to "1", data is transmitted on the falling edge of RxDi(1)

(Note 2)

NOTES:
1. Data is transmitted on the falling edge of a signal applied to the RxDi pin when the IOPOL bit is set to "0". Data is transmitted on the rising edge of a signal applied to the RxDi pin when the IOPOL bit is set to "1".
2. Data transmission condition must be met before the falling edge of the RxDi pin.

Figure 17.28 Bit Function Related Bus Conflict Detection

17.7 Special Mode 5 (SIM Mode)

In SIM mode, SIM interface devices can communicate in UART mode. Both direct and inverse formats are available and a low-level ("L") signal output can be provided from the TxDi pin (i=0 to 4) when a parity error is detected.

Table 17.34 lists specifications of SIM mode. Table 17.35 lists register settings. Tables 17.36 to 17.38 list pin settings.

Table 17.34 SIM Mode Specifications

Item	Specification
Transfer Data Format	• Transfer data: 8-bit UART mode • One stop bit • In direct format: Parity: Even; Data logic: Direct; Transfer format: LSB first • In inverse format: Parity: Odd; Data logic: Inverse; Transfer format: MSB first
Transfer Clock	• The CKDIR bit in the UiMR register (i=0 to 4) is "0" (internal clock selected): $fj/16(m+1)$[1] fj = f_1, f_8, f_{2n}[2] m : setting value of the UiBRG register, 00_{16} to FF_{16} Do not set the CKDIR bit to "1" (external clock selected)
Transmit/Receive Control	The CRD bit in the UiC0 register is set to "1" ($\overline{CTS}$, $\overline{RTS}$ function disabled)
Other Setting Items	The UiIRS bit in the UiC1 register is set to "1" (transmission completed)
Transmit Start Condition	To start transmitting, the following requirements must be met: - Set the TE bit in the UiC1 register to "1" (transmit enable) - Set the TI bit in the UiC1 register to "0" (data in the UiTB register)
Receive Start Condition	To start receiving, the following requirements must be met: - Set the RE bit in the UiC1 register to "1" (receive enable) - Detect the start bit
Interrupt Request Generation Timing	• While transmitting, -The UiIRS bit is set to "1" (transmission completed): when data transmission from the UARTi transfer register is completed • While receiving, when data is transferred from the UARTi receive register to the UiRB register (reception completed)
Error Detection	• Overrun error[1] This error occurs when the eighth bit of the next data is received before reading the UiRB register • Flaming error This error occurs when the number of the stop bit set is not detected • Parity error This error occurs when the number of "1" in parity bit and character bits differs from the number set • Error sum flag The SUM bit is set to "1" when an overrun error, framing error or parity error occurs

NOTES:

1. If an overrun error occurs, the UiRB register is indeterminate. The IR bit in the SiRIC register does not change to "1" (interrupt requested).
2. The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2n (n=1 to 15).

Table 17.35 Register Settings in SIM Mode

Register	Bit	Function
UiTB	7 to 0	Set transmit data
UiRB	7 to 0	Received data can be read
	OER, FER, PER, SUM	Error flags
UiBRG	7 to 0	Set bit rate
UiMR	SMD2 to SMD0	Set to "101_2"
	CKDIR	Set to "0"
	STPS	Set to "0"
	PRY	Set to "1" for direct format or "0" for inverse format
	PRYE	Set to "1"
	IOPOL	Set to "0"
UiC0	CLK1, CLK0	Select count source for the UiBRG register
	CRS	Disabled because the CRD bit is set to "1"
	TXEPT	Transfer register empty flag
	CRD	Set to "1"
	NCH	Set to "1"
	CKPOL	Set to "0"
	UFORM	Set to "0" for direct format or "1" for inverse format
UiC1	TE	Set to "1" to enable data transmission
	TI	Transfer buffer empty flag
	RE	Set to "1" to enable data reception
	RI	Reception complete flag
	UiIRS	Set to "1"
	UiRRM	Set to "0"
	UiLCH	Set to "0" for direct format or "1" for inverse format
	UiERE	Set to "1"
UiSMR	7 to 0	Set to "00_{16}"
UiSMR2	7 to 0	Set to "00_{16}"
UiSMR3	7 to 0	Set to "00_{16}"
UiSMR4	7 to 0	Set to "00_{16}"

i=0 to 4

Table 17.36 Pin Settings in SIM Mode (1)

Port	Function	Setting	
		PS0 Register	PD6 Register
P6$_2$	RxD0 input	PS0_2=0	PD6_2=0
P6$_3$	TxD0 output	PS0_3=1	–
P6$_6$	RxD1 input	PS0_6=0	PD6_6=0
P6$_7$	TxD1 output	PS0_7=1	–

Table 17.37 Pin Settings (2)

Port	Function	Setting			
		PS1 Register	PSL1 Register	PSC Register	PD7 Register
P7$_0$(1)	TxD2 output	PS1_0=1	PSL1_0=0	PSC_0=0	–
P7$_1$(1)	RxD2 input	PS1_1=0	–	–	PD7_1=0

NOTES:

1. P7$_0$ and P7$_1$ are ports for the N-channel open drain output.

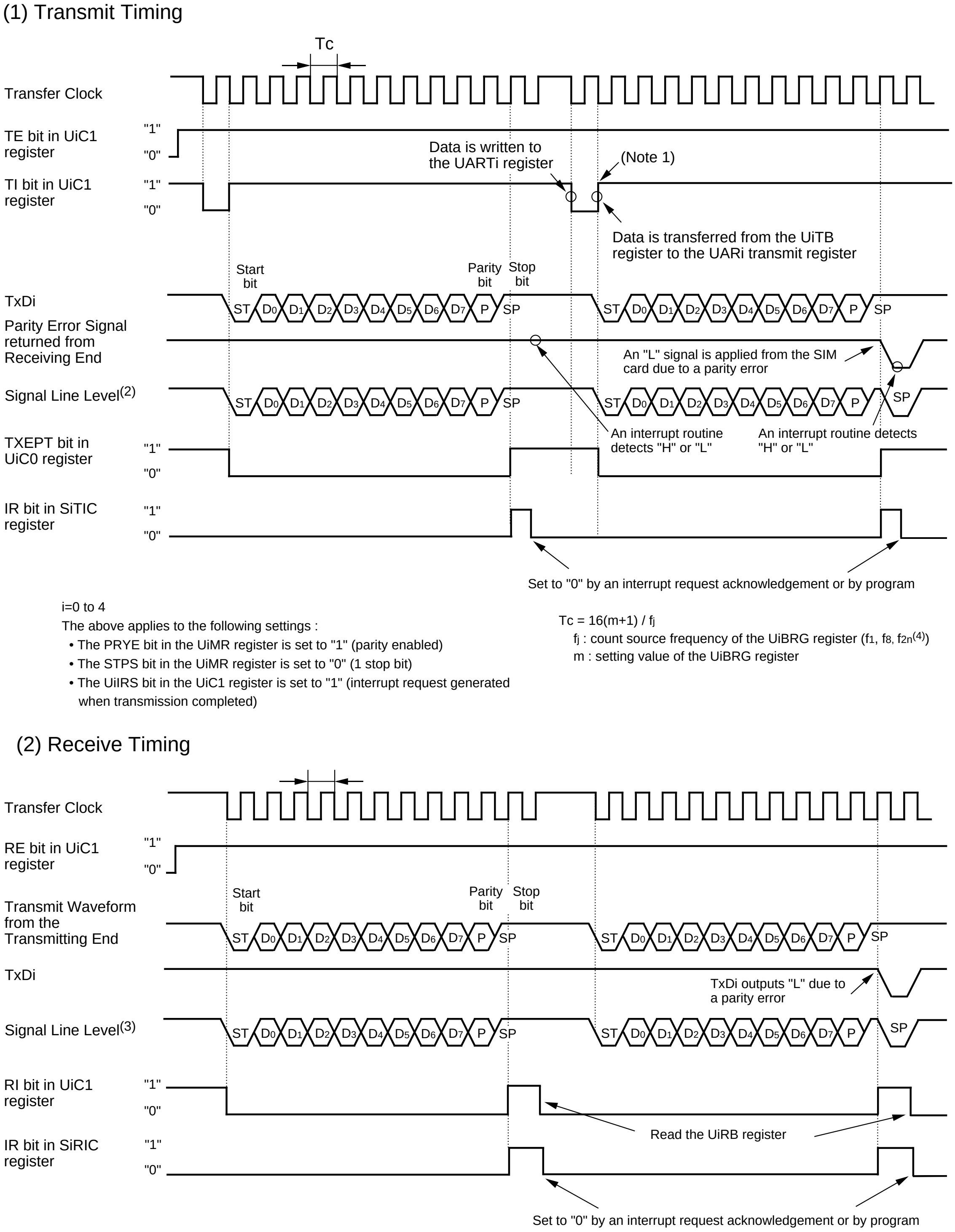
Table 17.38 Pin Settings (3)

Port	Function	Setting			
		PS3 Register(1)	PSL3 Register	PSC3 Register	PD9 Register(1)
P9$_1$	RxD3 input	PS3_1=0	–		PD9_1=0
P9$_2$	TxD3 output	PS3_2=1	PSL3_2=0		–
P9$_6$	TxD4 output	PS3_6=1	–	PSC3_6=0	–
P9$_7$	RxD4 input	PS3_7=0	–		PD9_7=0

NOTES:

1. Set the PD9 and PS3 registers immediately after the PRC2 bit in the PRCR register is set to "1" (write enable). Do not generate an interrupt or a DMA transfer between the instruction to set to the PRC2 bit to "1" and the instruction to set the PD9 and PS3 registers.

Figure 17.29 shows an example of a SIM interface operation. Figure 17.30 shows an example of a SIM interface connection. Connect the TxDi pin to the RxDi pin for a pull-up.


(1) Transmit Timing
Tc
Transfer Clock
TE bit in UiC1 register
"1"
"0"
TI bit in UiC1 register
"1"
"0"
Data is written to the UARTi register
(Note 1)
Data is transferred from the UiTB register to the UARi transmit register
TxDi
Start bit
Parity bit
Stop bit
ST D0 D1 D2 D3 D4 D5 D6 D7 P SP
Parity Error Signal returned from Receiving End
An "L" signal is applied from the SIM card due to a parity error
Signal Line Level(2)
An interrupt routine detects "H" or "L"
An interrupt routine detects "H" or "L"
TXEPT bit in UiC0 register
IR bit in SiTIC register
Set to "0" by an interrupt request acknowledgement or by program
(2) Receive Timing
Transfer Clock
RE bit in UiC1 register
Transmit Waveform from the Transmitting End
TxDi
TxDi outputs "L" due to a parity error
Signal Line Level(3)
RI bit in UiC1 register
Read the UiRB register
IR bit in SiRIC register
Set to "0" by an interrupt request acknowledgement or by program

i=0 to 4

The above applies to the following settings :

- The PRYE bit in the UiMR register is set to "1" (parity enabled)
- The STPS bit in the UiMR register is set to "0" (1 stop bit)
- The UiIRS bit in the UiC1 register is set to "1" (interrupt request generated when transmission completed)

Tc = 16(m+1) / f_j

f_j : count source frequency of the UiBRG register (f_1, f_8, f_{2n}(4))

m : setting value of the UiBRG register

i=0 to 4

The above applies to the following settings :

- The PRYE bit in the UiMR register is set to "1" (parity enabled)
- The STPS bit in the UiMR register is set to "0" (1 stop bit)

Tc = 16(m+1) / f_j

f_j : count source frequency of the UiBRG register (f_1, f_8, f_{2n}(4))

m : setting value of the UiBRG register

NOTES:

1. Data transmission starts when BRG overflows after a value is set to the UiTB register on the rising edge of the TI bit.
2. Because the TxDi and RxDi pins are connected, a composite waveform, consisting of transmit waveform from the TxDi pin and parity error signal from the receiving end, is generated.
3. Because the TxDi and RxDi pins are connected, a composite waveform, consisting of transmit waveform from the transmitting end and parity error signal from the TxDi pin, is generated.
4. The CNT3 to CNT0 bits in the TCSPR register selects no division (n=0) or divide-by-2n (n=1 to 15).

Figure 17.29 SIM Interface Operation

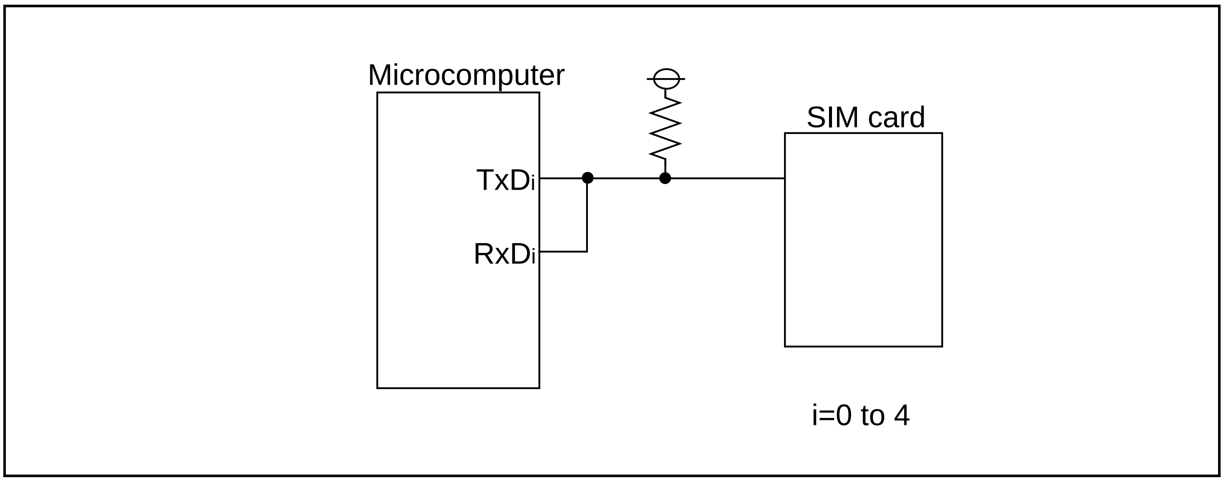


Figure 17.30 SIM Interface Connection

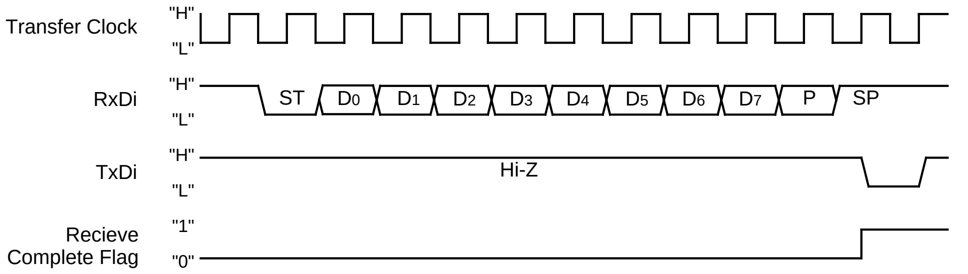
17.7.1 Parity Error Signal

17.7.1.1 Parity Error Signal Output Function

When the UiERE bit in the UiC1 register (i=0 to 4) is set to "1" (output), the parity error signal output can be provided. The parity error signal output is provided when a parity error is detected upon receiving data. A low-level ("L") signal output is provided from the TxDi pin in the timing shown in Figure 17.31. When reading the UiRB register during a parity error output, the PER bit in the UiRB register is set to "0" (no error occurs) and a high-level ("H") signal output is again provided simultaneously.

17.7.1.2 Parity Error Signal

To determine whether the parity error signal is output, the port that shares a pin with the RxDi pin is read by using an end-of-transmit interrupt routine.



NOTES:
1. The above applies to direct format conditions.
(The PRY bit is set to "1", the UFORM bit is set to "0",
and the UiLCH bit is set to "0").

ST : Start bit
P : Even parity
SP : Stop bit
i=0 to 4

Figure 17.31 Parity Error Signal Output Timing (LSB First)

17.7.2 Format

17.7.2.1 Direct Format

Set the PRYE bit in the UiMR register (i=0 to 4) to "1" (parity enabled), the PRY bit to "1" (even parity), the UFORM bit in the UiC0 register to "0" (LSB first) and the UiLCH bit in the UiC1 register to "0" (not inversed). When data are transmitted, data set in the UiTB register are transmitted with the even-numbered parity, starting from D_0. When data are received, received data are stored in the UiRB register, starting from D_0. The even-numbered parity determines whether a parity error occurs.

17.7.2.2 Inverse Format

Set the PRYE bit to "1", the PRY bit to "0" (odd parity), the UFORM bit to "1" (MSB first) and the UiLCH bit to "1" (inversed). When data are transmitted, values set in the UiTB register are logically inversed and are transmitted with the odd-numbered parity, starting from D_7. When data are received, received data are logically inversed to be stored in the UiRB register, starting from D_7. The odd-numbered parity determines whether a parity error occurs.

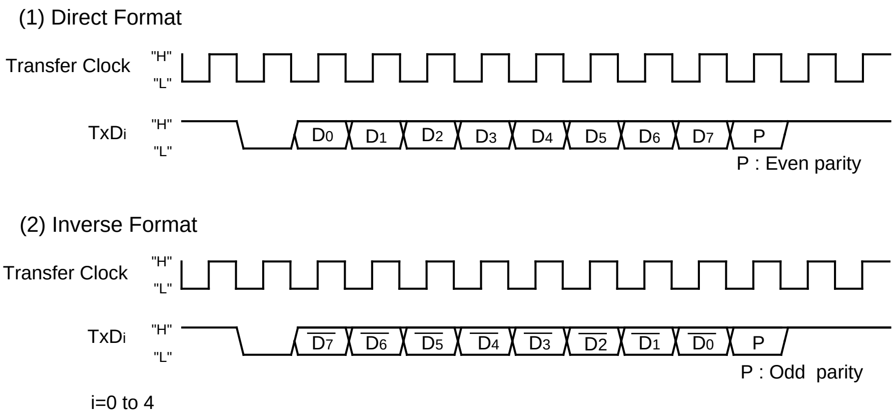


Figure 17.32 SIM Interface Format

18. A/D Converter

The A/D converter consists of one 10-bit successive approximation A/D converter with a capacitive coupling amplifier.

The result of an A/D conversion is stored into the A/D registers corresponding to selected pins. It is stored into the AD00 register only when DMAC operating mode is entered.

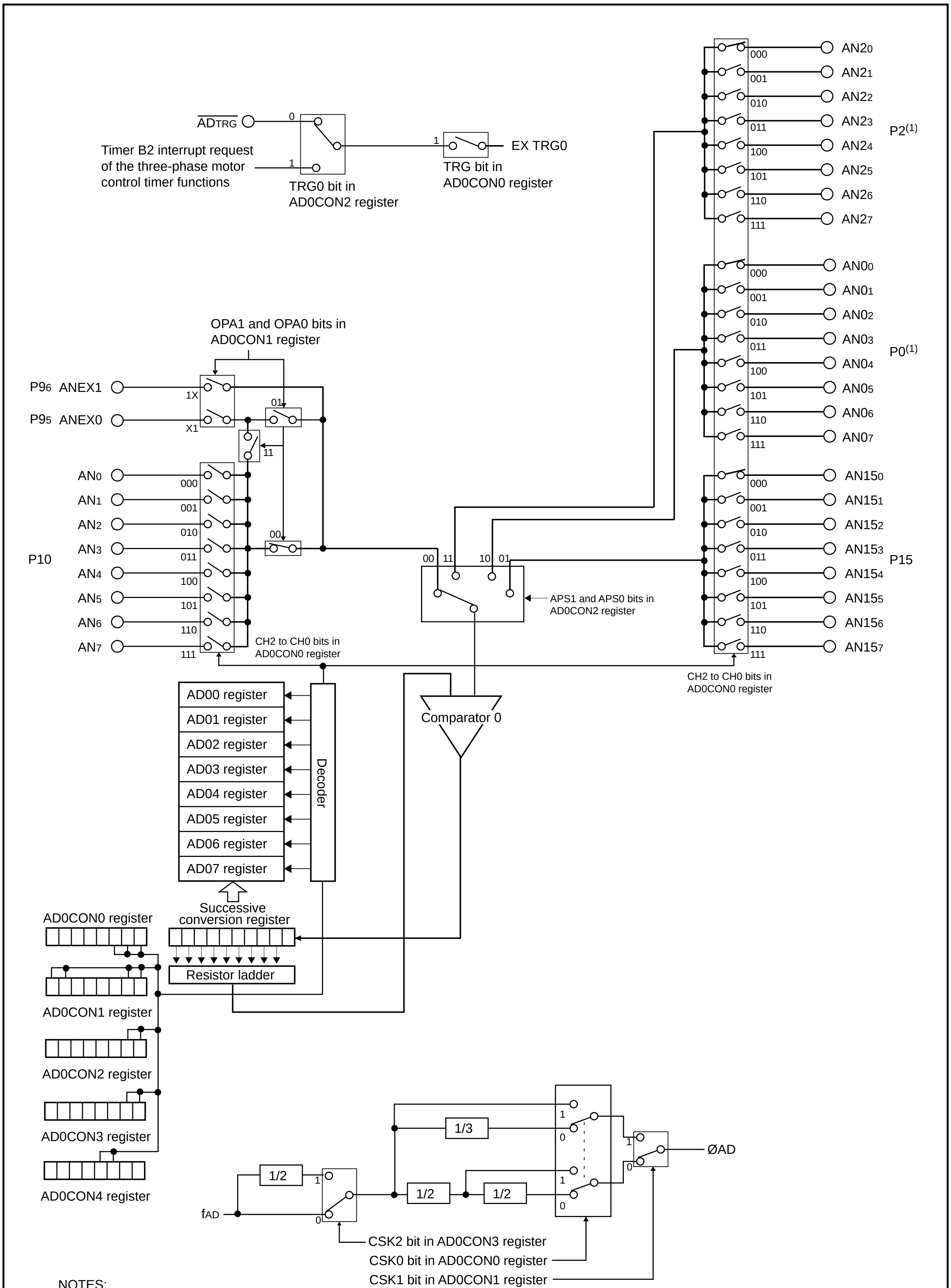
Table 18.1 lists specifications of the A/D converter. Figure 18.1 shows a block diagram of the A/D converter. Figures 18.2 to 18.6 show registers associated with the A/D converter.

Table 18.1 A/D Converter Specifications

Item	Specification
A/D Conversion Method	Successive approximation (with a capacitive coupling amplifier)
Analog Input Voltage(1)	0V to AV$_{CC}$ (V$_{CC}$)
Operating Clock, Ø$_{AD}$(2)	f_{AD}, $f_{AD}/2$, $f_{AD}/3$, $f_{AD}/4$, $f_{AD}/6$, $f_{AD}/8$
Resolution	Select from 8 bits or 10 bits
Operating Mode	One-shot mode, repeat mode, single sweep mode, repeat sweep mode 0, repeat sweep mode 1, multi-port single sweep mode, multi-port repeat sweep mode 0
Analog Input Pins(3)	34 pins 8 pins each for AN (AN$_0$ to AN$_7$), AN0 (AN0$_0$ to AN0$_7$), AN2 (AN2$_0$ to AN2$_7$), AN15 (AN15$_0$ to AN15$_7$) 2 extended input pins (ANEX0 and ANEX1)
A/D Conversion Start Condition	• Software trigger The ADST bit in the AD0CON0 register is set to "1" (A/D conversion started) by program • External trigger (re-trigger is enabled) When a falling edge is applied to the $\overline{\text{AD}_{\text{TRG}}}$ pin after the ADST bit is set to "1" by program • Hardware trigger (re-trigger is enabled) The timer B2 interrupt request of the three-phase motor control timer functions (after the ICTB2 counter completes counting) is generated after the ADST bit is set to "1" by program
Conversion Rate Per Pin	• Without the sample and hold function 8-bit resolution : 49 Ø$_{AD}$ cycles 10-bit resolution : 59 Ø$_{AD}$ cycles • With the sample and hold function 8-bit resolution : 28 Ø$_{AD}$ cycles 10-bit resolution : 33 Ø$_{AD}$ cycles

NOTES:

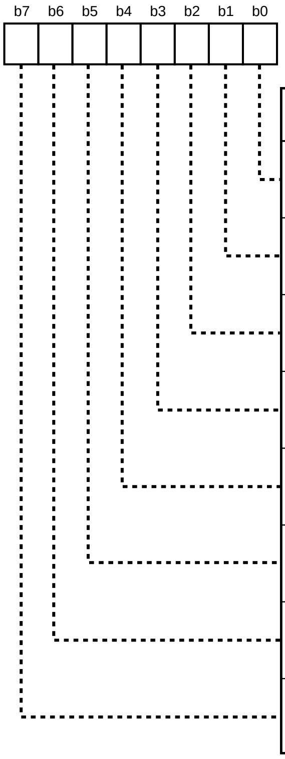
1. Analog input voltage is not affected by the sample and hold function status.
2. Ø$_{AD}$ frequency must be under 16 MHz when V$_{CC}$=5V.
   Without the sample and hold function, the Ø$_{AD}$ frequency is 250 kHz or more.
   With the sample and hold function, the Ø$_{AD}$ frequency is 1 MHz or more.
3. AV$_{CC}$ = V$_{REF}$ = V$_{CC}$, A/D input voltage (for AN$_0$ to AN$_7$, AN0$_0$ to AN0$_7$, AN2$_0$ to AN2$_7$, AN15$_0$ to AN15$_7$, ANEX0 and ANEX1) $\leq$ V$_{CC}$.



NOTES:
1. These pins are available in single-chip mode.

Figure 18.1 A/D Converter Block Diagram

A/D0 Control Register 0[1]



Symbol	Address	After Reset
AD0CON0	0396_{16}	00_{16}

Bit Symbol	Bit Name	Function	RW
CH0	Analog Input Pin Select Bit[2, 3, 8, 9]	b2 b1 b0 0 0 0 : ANi$_0$ 0 0 1 : ANi$_1$ 0 1 0 : ANi$_2$ 0 1 1 : ANi$_3$ 1 0 0 : ANi$_4$ 1 0 1 : ANi$_5$ 1 1 0 : ANi$_6$ 1 1 1 : ANi$_7$ (i=none, 0, 2, 15)	RW
CH1			RW
CH2			RW
MD0	A/D Operating Mode Select Bit 0[2, 6, 7]	b4 b3 0 0 : One-shot mode 0 1 : Repeat mode 1 0 : Single sweep mode 1 1 : Repeat sweep mode 0 or 1	RW
MD1			RW
TRG	Trigger Select Bit	0 : Software trigger 1 : External trigger, hardware trigger[4]	RW
ADST	A/D Conversion Start Flag	0 : A/D conversion stops 1 : A/D conversion starts[4]	RW
CKS0	Frequency Select Bit	(Note 5)	RW

NOTES:

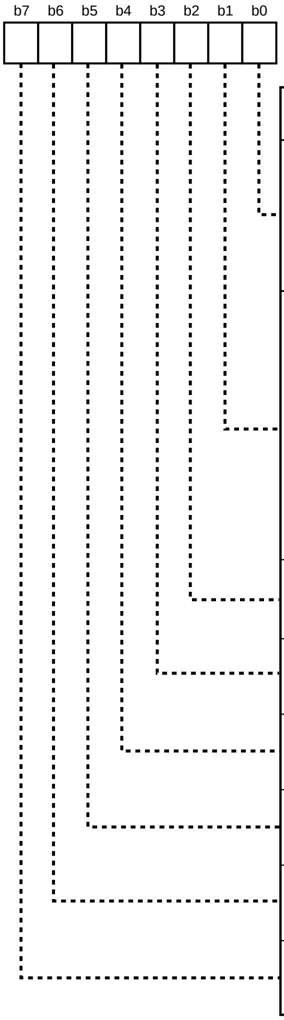
1. When the AD0CON0 register is rewritten during the A/D conversion, the conversion result is indeterminate.
2. Analog input pins must be set again after changing an A/D operating mode.
3. The CH2 to CH0 bit settings are enabled in one-shot mode and repeat mode.
4. To set the TRG bit to "1", select the cause of trigger by setting the TRG0 bit in the AD0CON2 register. Then set the ADST bit to "1" after the TRG bit is set to "1".
5. ϕ_{AD} frequency must be under 16 MHz when V_{CC}=5V.
   Combination of the CKS0, CKS1 and CKS2 bits selects ϕ_{AD}.

The CKS2 Bit in the AD0CON3 Register	The CKS0 Bit in the AD0CON0 Register	The CKS1 Bit in the AD0CON1 Register	ϕ_{AD}
0	0	0	f_{AD} divided by 4
		1	f_{AD} divided by 3
	1	0	f_{AD} divided by 2
		1	f_{AD}
1	0	0	f_{AD} divided by 8
		1	f_{AD} divided by 6

6. When the MSS bit in the AD0CON3 register is set to "1" (multi-port sweep mode enabled), set the MD1 and MD0 bits to "10_2" to enter multi-port single sweep mode and to "11_2" to enter multi-port repeat sweep mode 0.
7. When the MSS bit is set to "1", the MD1 and MD0 bits cannot be set to "00_2" or "01_2".
8. $AV_{CC}=V_{REF}=V_{CC}$, AD input voltage (for AN$_0$ to AN$_7$, AN0$_0$ to AN0$_7$, AN2$_0$ to AM2$_7$, AN15$_0$ to AN15$_7$, ANEX0, ANEX1) $\leq V_{CC}$.
9. Set the PSC_7 bit in the PSC register to "1" to use the P10 pin as an analog input pin.

Figure 18.2 AD0CON0 Register

A/D0 Control Register 1[1]



Symbol	Address	After Reset
AD0CON1	0397_{16}	00_{16}

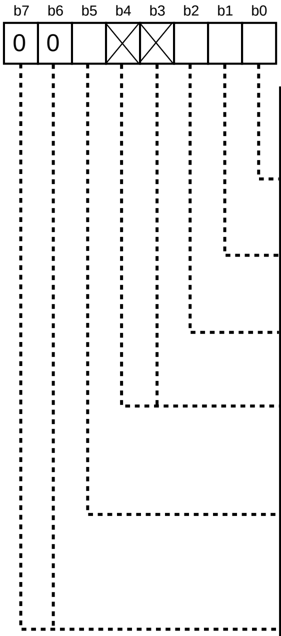
Bit Symbol	Bit Name	Function	RW
SCAN0	A/D Sweep Pin Select Bit[2, 10]	Single sweep mode and repeat sweep mode 0 b1 b0 0 0 : ANi$_0$, ANi$_1$ 0 1 : ANi$_0$ to ANi$_3$ 1 0 : ANi$_0$ to ANi$_5$ 1 1 : ANi$_0$ to ANi$_7$ Repeat sweep mode 1[3] b1b0 0 0 : ANi$_0$ 0 1 : ANi$_0$, ANi$_1$ 1 0 : ANi$_0$ to ANi$_2$ 1 1 : ANi$_0$ to ANi$_3$ (i=none, 0, 2, 15) Multi-port single sweep mode and multi-port repeat sweep mode 0[4] b1b0 1 1 : ANi$_0$ to ANi$_7$	RW
SCAN1			RW
MD2	A/D Operating Mode Select Bit 1	0 : Any mode other than repeat sweep mode 1 1 : Repeat sweep mode 1[5]	RW
BITS	8/10-Bit Mode Select Bit	0 : 8-bit mode 1 : 10-bit mode	RW
CKS1	Frequency Select Bit	(Note 6)	RW
VCUT	VREF Connection Bit	0 : No VREF connection[11] 1 : VREF connection	RW
OPA0	External Op-Amp Connection Mode Bit[7, 9]	b7 b6 0 0 : ANEX0 and ANEX1 are not used[8] 0 1 : Signal into ANEX0 is A/D converted 1 0 : Signal into ANEX1 is A/D converted 1 1 : External op-amp connection mode	RW
OPA1			RW

NOTES:

1. When the AD0CON1 register is rewritten during the A/D conversion, the conversion result is indeterminate.
2. The SCAN1 and SCAN0 bit settings are disabled in single sweep mode, repeat sweep mode 0, repeat sweep mode 1, mutli-port single sweep mode and multi-port repeat sweep mode 0.
3. This pin is commonly used in the A/D conversion when the MD2 bit is set to "1".
4. In multi-port single sweep mode or multi-port repeat sweep mode 0, do not set the SCAN1 and SCAN0 bits to any setting other than "11_2".
5. When the MSS bit in the AD0CON3 register is set to "1" (multi-port sweep mode enabled), set the MD2 bit to "0".
6. Refer to the note for the CKS0 bit in the AD0CON0 register.
7. In one-shot mode and repeat mode, the OPA1 and OPA0 bits can be set to "01_2" or "10_2" only. Do not set the OPA0 and OPA1 bits to "01_2" or "10_2" in other modes.
8. To set the OPA1 and OPA0 bits to "00_2", set the PSL3_5 bit in PSL3 register to "0" (other than ANEX0) and the PSL3_6 bit to "0" (other than ANEX1).
9. When the MSS bit is set to "1", set the OPA1 and OPA0 bits to "00_2".
10. $AV_{CC}=V_{REF}=V_{CC}$, AD input voltage (for AN$_0$ to AN$_7$, AN0$_0$ to AN0$_7$, AN2$_0$ to AM2$_7$, AN15$_0$ to AN15$_7$, ANEX0, ANEX1) ≤ V_{CC}.
11. Do not set the VCUT bit to "0" during the A/D conversion.
    VREF is a reference voltage for AD0 only. The VCUT bit setting does not affect the VREF performance of the D/A converter.

Figure 18.3 AD0CON1 Register

A/D0 Control Register 2[1]



Symbol	Address	After Reset
AD0CON2	0394_{16}	XX0X X000$_2$

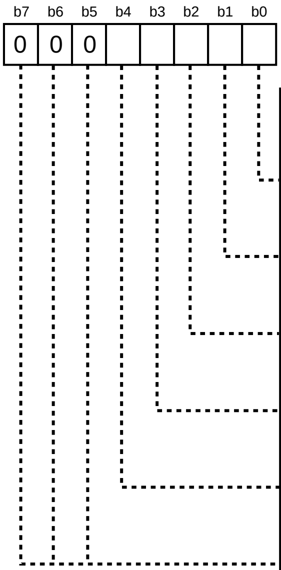
Bit Symbol	Bit Name	Function	RW
SMP	A/D Conversion Method Select Bit	0 : Without the sample and hold funtion 1 : With the sample and hold function	RW
APS0	Analog Input Port Select Bit[2, 3]	b2b1 0 0 : AN_0 to AN_7, ANEX0, ANEX1 0 1 : $AN15_0$ to $AN15_7$ 1 0 : $AN0_0$ to $AN0_7$ 1 1 : $AN2_0$ to $AN2_7$	RW
APS1			RW
(b4 - b3)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		–
TRG0	External Trigger Request Cause Select Bit	0 : Selects $\overline{\text{ADTRG}}$ 1 : Selects a timer B2 interrupt request of the three-phase motor control timer functions (after the ICTB2 counter completes counting)	RW
(b7 - b6)	Reserved Bit	Set to "0". When read, its content is indeterminate.	RW

NOTES:

1. When the AD0CON2 register is rewritten during the A/D conversion, the conversion result is indeterminate.
2. When the MSS bit in the AD0CON3 register is set to "1" (multi-port sweep mode enabled), set the APS1 and APS0 bits to "01$_2$".
3. The APS1 and APS0 bits can be set to "10$_2$" or "11$_2$" in single-chip mode only.

Figure 18.4 AD0CON2 Register

A/D0 Control Register 3(1, 2)



Symbol	Address	After Reset
AD0CON3	0395_{16}	$XXXX\ X000_2$

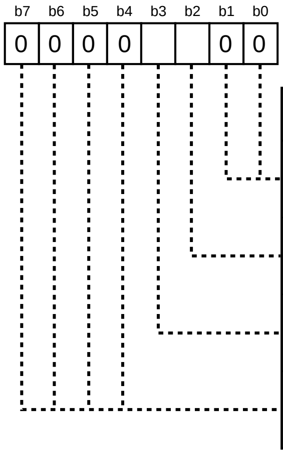
Bit Symbol	Bit Name	Function	RW
DUS	DMAC Operation Select Bit(3)	0 : Disables DMAC operating mode 1 : Enables DMAC operating mode(4, 5)	RW
MSS	Multi-Port Sweep Mode Select Bit	0 : Disables multi-port sweep mode 1 : Enables multi-port sweep mode(3, 6)	RW
CKS2	Frequency Select Bit	(Note 7)	RW
MSF0	Multi-Port Sweep Status Flag(8)	b4 b3 0 0 : AN$_0$ to AN$_7$ 0 1 : AN15$_0$ to AN15$_7$ 1 0 : AN0$_0$ to AN0$_7$ 1 1 : AN2$_0$ to AN2$_7$	RO
MSF1			RO
—— (b7 - b5)	Reserved Bit	Set to "0". When read, its content is indeterminate.	RW

NOTES:

1. When the AD0CON3 register is rewritten during the A/D conversion, the conversion result is indeterminate.
2. The AD0CON3 may be read uncorrectly during the A/D conversion. It must be read or written after the A/D converter stops operating.
3. When the MSS bit is set to "1", set the DUS bit to "1".
4. When the DUS bit is set to "1", the AD00 register stores all A/D conversion results.
5. When the DUS bit is set to "1", set the DMAC.
6. When the MSS bit is set to "1", set the MD2 bit in the AD0CON1 register to "0" (other than repeat sweep mode 1), the APS1 and APS0 bits in the AD0CON2 register to "01$_2$" (AN15$_0$ to AN15$_7$) and the OPA1 and OPA0 bits in the AD0CON1 register to "00$_2$" (ANEX0 and ANEX1 not used).
7. Refer to the note for the CKS0 bit in the AD0CON0 register.
8. The MSF1 and MSF0 bit settings are enabled when the MSS bit is set to "1". Value in the bit is indeterminate when the MSS bit is set to "0".

Figure 18.5 AD0CON3 Register

A/D0 Control Register 4[1]

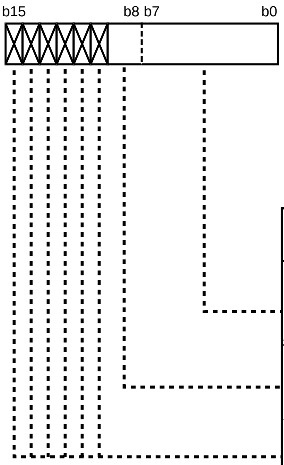


Symbol	Address	After Reset
AD0CON4	0392$_{16}$	XXXX 00XX$_2$

Bit Symbol	Bit Name	Function	RW
(b1 - b0)	Reserved Bit	Set to "0". When read, its content is indeterminate.	RW
MPS10	Multi-Port Sweep Port Select Bit[2]	b3 b2 0 0 : (Note 3) 0 1 : AN$_0$ to AN$_7$, AN15$_0$ to AN15$_7$ 1 0 : AN$_0$ to AN$_7$, AN0$_0$ to AN0$_7$ 1 1 : AN$_0$ to AN$_7$, AN2$_0$ to AN2$_7$	RW
MPS11			RW
(b7 - b4)	Reserved Bit	Set to "0". When read, its content is indeterminate.	RW

NOTES:
1. When the AD0CON4 register is rewritten during the A/D conversion, the conversion result is indeterminate.
2. The MPS11 and MPS10 bits can be set to "10$_2$" or "11$_2$" in single-chip mode only.
3. When the MSS bit in the AD0CON3 regsiter is set to "0" (multi-port sweep mode disabled), set the MPS11 and MPS10 bits to "00$_2$".
   When the MSS bit is set to "1" (multi-port sweep mode enabled), set the MPS11 and MPS10 bits to "01$_2$", "10$_2$" or "11$_2$".

A/D0 Register i (i =0 to 7)[1, 2, 3, 4, 5]



Symbol	Address	After Reset
AD00	0381$_{16}$ - 0380$_{16}$	00000000 XXXXXXXX$_2$
AD01 to AD03	0383$_{16}$ - 0382$_{16}$, 0385$_{16}$ - 0384$_{16}$, 0387$_{16}$ - 0386$_{16}$	Indeterminate
AD04 to AD06	0389$_{16}$ - 0388$_{16}$, 038B$_{16}$ - 038A$_{16}$, 038D$_{16}$ - 038C$_{16}$	Indeterminate
AD07	038F$_{16}$ - 038E$_{16}$	Indeterminate

Function	RW
8 low-order bits in an A/D conversion result	RO
In 10-bit mode : 2 high-order bits in an A/D conversion result In 8-bit mode : When read, its content is indeterminate.	RO
When read, its content is indeterminate.	RO

NOTES:
1. In DMAC operating mode, register value read by program is indeterminate.
2. Register value is indeterminate when written while the A/D conversion is stopped.
3. Register value is indeterminate if the next A/D conversion result is stored before reading the register.
4. The AD00 register is available in DMAC operating mode. Other registers are indeterminate.
5. In DMAC operating mode and 10-bit mode, set DMAC for a 16-bit transfer.

Figure 18.6 AD0CON4 Register and AD00 to AD07 Registers

18.1 Mode Description

18.1.1 One-shot Mode

In one-shot mode, analog voltage applied to a selected pin is converted to a digital code once. Table 18.2 lists specifications of one-shot mode.

Table 18.2 One-shot Mode Specifications

Item	Specification
Function	The CH2 to CH0 bits in the AD0CON0 register, the OPA1 and OPA0 bits in the AD0CON1 register and the APS1 and APS0 bits in the AD0CON2 register select a pin. Analog voltage applied to the pin is converted to a digital code once
Start Condition	• When the TRG bit in the AD0CON0 register is set to "0" (software trigger), the ADST bit in the AD0CON0 register is set to "1" (A/D conversion starts) by program • When the TRG bit is set to "1" (external trigger, hardware trigger): - a falling edge is applied to the $\overline{\text{ADTRG}}$ pin after the ADST bit is set to "1" by program - The timer B2 interrupt request of three-phase motor control timer functions (after the ICTB2 register counter completes counting) is generated after the ADST bit is set to "1" by program
Stop Condition	• A/D conversion is completed (the ADST bit is set to "0" when the software trigger is selected) • The ADST bit is set to "0" (A/D conversion stopped) by program
Interrupt Request Generation Timing	A/D conversion is completed
Analog Voltage Input Pins	Select one pin from ANi0 to ANi7 (i=none, 0, 2, 15), ANEX0 or ANEX1
Reading of A/D Conversion Result	• When the DUS bit in the AD0CON3 register is set to "0" (DMAC operating mode disabled), the microcomputer reads the AD0j register (j=0 to 7) corresponding to selected pin • When the DUS bit is set to "1" (DMAC operating mode enabled), do not read the AD00 register. A/D conversion result is stored in the AD00 register after the A/D conversion is completed. DMAC transfers the conversion result to any memory space. Refer to **13. DMAC** for DMAC settings

18.1.2 Repeat Mode

In repeat mode, analog voltage applied to a selected pin is repeatedly converted to a digital code. Table 18.3 lists specifications of repeat mode.

Table 18.3 Repeat Mode Specifications

Item	Specification
Function	The CH2 to CH0 bits in the AD0CON0 register, the OPA1 and OPA0 bits in the AD0CON1 register and the APS1 and APS0 bits in the AD0CON2 register select a pin. Analog voltage applied to the pin is repeatedly converted to a digital code
Start Condition	Same as one-shot mode
Stop Condition	The ADST bit in the AD0CON0 register is set to "0" (A/D conversion stopped) by program
Interrupt Request Generation Timing	• When the DUS bit in the AD0CON3 register is set to "0" (DMAC operating mode disabled), no interrupt request is generated. • When DUS bit is set to "1" (DMAC operating mode enabled), an interrupt request is generated every time an A/D conversion is completed.
Analog Voltage Input Pins	Select one pin from ANi$_0$ to ANi$_7$ (i=none, 0, 2, 15), ANEX0 or ANEX1
Reading of A/D Conversion Result	• When the DUS bit is set to "0", the microcomputer reads the AD0j register (j=0 to 7) corresponding to the selected pin. • When DUS bit is set to "1", do not read the AD00 register. A/D conversion result is stored in the AD00 register after the A/D conversion is completed. DMAC transfers the conversion result to any memory space. Refer to **13. DMAC** for DMAC settings

18.1.3 Single Sweep Mode

In single sweep mode, analog voltage that is applied to selected pins is converted one-by-one to a digital code. Table 18.4 lists specifications of single sweep mode.

Table 18.4 Single Sweep Mode Specifications

Item	Specification
Function	The SCAN1 and SCAN0 bits in the AD0CON1 register and the APS1 and APS0 bits in the AD0CON2 register select pins. Analog voltage applied to the pin is converted one-by-one to a digital code
Start Condition	Same as one-shot mode
Stop Condition	Same as one-shot mode
Interrupt Request Generation Timing	• When the DUS bit in the AD0CON3 register is set to "0" (DMAC operating mode disabled), an interrupt request is generated after a sweep is completed. • When DUS bit is set to "1" (DMAC operating mode enabled), an interrupt request is generated every time an A/D conversion is completed
Analog Voltage Input Pins	Select from ANi_0 and ANi_1 (2 pins) (i=none, 0, 2, 15), ANi_0 to ANi_3 (4 pins), ANi_0 to ANi_5 (6 pins) or ANi_0 to ANi_7 (8 pins)
Reading of A/D Conversion Result	• When the DUS bit is set to "0", the microcomputer reads the AD0j register corresponding to selected pins • When DUS bit is set to "1", do not read the AD00 register. A/D conversion result is stored in the AD00 register after the A/D conversion is completed. DMAC transfers the conversion result to any memory space. Refer to **13. DMAC** for DMAC settings

18.1.4 Repeat Sweep Mode 0

In repeat sweep mode 0, analog voltage applied to selected pins is repeatedly converted to a digital code. Table 18.5 lists specifications of repeat sweep mode 0.

Table 18.5 Repeat Sweep Mode 0 Specifications

Item	Specification
Function	The SCAN1 and SCAN0 bits in the AD0CON1 register and the APS1 and APS0 bits in the AD0CON2 register select pins. Analog voltage applied to the pins is repeatedly converted to a digital code
Start Condition	Same as one-shot mode
Stop Condition	The ADST bit in the AD0CON0 register is set to "0" (A/D conversion stopped) by program
Interrupt Request Generation Timing	• When the DUS bit in the AD0CON3 register is set to "0" (DMAC operating mode disabled), no interrupt request is generated • When DUS bit is set to "1" (DMAC operating mode enabled), an interrupt request is generated every time an A/D conversion is completed
Analog Voltage Input Pins	Select from ANi$_0$ and ANi1 (2 pins) (i=none, 0, 2, 15), ANi$_0$ to ANi$_3$ (4 pins), ANi$_0$ to ANi$_5$ (6 pins) or ANi$_0$ to ANi$_7$ (8 pins)
Reading of A/D Conversion Result	• When the DUS bit is set to "0", the microcomputer reads the AD0j register (j=0 to 7) corresponding to selected pins • When the DUS bit is set to "1", do not read the AD00 register. A/D conversion result is stored in the AD00 register after the A/D conversion is completed. DMAC transfers the conversion result to any memory space. Refer to **13. DMAC** for DMAC settings

18.1.5 Repeat Sweep Mode 1

In repeat sweep mode 1, analog voltage selectively applied to eight pins is repeatedly converted to a digital code. Table 18.6 lists specifications of repeat sweep mode 1.

Table 18.6 Repeat Sweep Mode 1 Specifications

Item	Specification
Function	The SCAN1 and SCAN0 bits in the AD0CON1 register and the APS1 and APS0 bits in the AD0CON2 register select 8 pins. Analog voltage selectively applied to 8 pins is repeatedly converted to a digital code e.g., When ANi$_0$ is selected (i =none, 0, 2, 15), analog voltage is converted to a digital code in the following order: ANi$_0$ → ANi$_1$ → ANi$_0$ → ANi$_2$→ ANi$_0$ → ANi$_3$ .... etc.
Start Condition	Same as one-shot mode (Any trigger generated during an A/D conversion is invalid)
Stop Condition	The ADST bit is set to "0" (A/D conversion stopped) by program
Interrupt Request Generation Timing	• When the DUS bit in the AD0CON3 register is set to "0" (DMAC operating mode disabled), no interrupt request is generated • When DUS bit is set to "1" (DMAC operating mode enabled), an interrupt request is generated every time an A/D conversion is completed
Analog Voltage Input Pins	ANi$_0$ to ANi$_7$ (8 pins)
Prioritized Pins	ANi$_0$ (1 pin), ANi$_0$ and ANi$_1$ (2 pins), ANi$_0$ to ANi$_2$ (3 pins) or ANi$_0$ to ANi$_3$ (4 pins)
Reading of A/D Conversion Result	• When the DUS bit is set to "0", the microcomputer reads the AD0j register (j=0 to 7) corresponding to selected pins • When the DUS bit is set to "1", do not read the AD00 register. A/D conversion result is stored in the AD00 register after the A/D conversion is completed. DMAC transfers the conversion result to any memory space. Refer to **13. DMAC** for DMAC settings

18.1.6 Multi-Port Single Sweep Mode

In multi-port single sweep mode, analog voltage applied to 16 selected pins is converted one-by-one to a digital code. Set the DUS bit in the AD0CON3 register to "1" (DMAC operating mode enabled). Table 18.7 lists specifications of multi-port single sweep mode.

Table 18.7 Multi-Port Single Sweep Mode Specifications

Item	Specification
Function	The MPS11 and MPS10 bits in the AD0CON4 register select 16 pins. Analog voltage applied to 16 pins is converted one-by-one to a digital code in the following order: AN_0 to AN_7 → ANi_0 to ANi_7 (i=0, 2, 15) e.g., When the MPS11 and MPS10 bits are set to "10_2" (AN_0 to AN_7, $AN0_0$ to $AN0_7$), analog voltage is converted to a digital code in the following order: AN_0 → AN_1 → AN_2 → AN_3 → AN_4 → AN_5 → AN_6 → AN_7 → $AN0_0$ → $AN0_1$ → .... → $AN0_6$ → $AN0_7$
Start Condition	Same as one-shot mode
Stop Condition	The ADST bit in the AD0CON0 register is set to "0" (A/D conversion stopped) by program
Interrupt Request Generation Timing	An interrupt request is generated every time A/D conversion is completed (Set the DUS bit to "1")
Analog Voltage Input Pins	Select from AN_0 to AN_7 → $AN15_0$ to $AN15_7$, AN_0 to AN_7 → $AN0_0$ to $AN0_7$ or AN_0 to AN_7 → $AN2_0$ to $AN2_7$
Reading of A/D Conversion Result	Do not read the AD00 register. A/D conversion result is stored in the AD00 register after the A/D conversion is completed. DMAC transfers the conversion result to any memory space. Refer to **13. DMAC** for DMAC settings (Set the DUS bit to "1")

18.1.7 Multi-Port Repeat Sweep Mode 0

In multi-port repeat sweep mode 0, analog voltage that is applied to 16 selected pins is repeatedly converted to a digital code. Set the DUS bit in the AD0CON3 register to "1" (DMAC operating mode enabled). Table 18.8 lists specifications of multi-port repeat sweep mode 0.

Table 18.8 Multi-Port Repeat Sweep Mode 0 Specifications

Item	Specification
Function	The MPS11 and MPS10 bits in the AD0CON4 register select 16 pins. Analog voltage applied to the 16 pins is repeatedly converted to a digital code in the following order: AN$_0$ to AN$_7$ → ANi_0 to ANi_7 (i=0, 2, 15) e.g., When the MPS11 and MPS10 bits are set to "10$_2$" (AN$_0$ to AN$_7$, AN0$_0$ to AN0$_7$), analog voltage is repeatedly converted to a digital code in the following order: AN$_0$ → AN$_1$ → AN$_2$ → AN$_3$ → AN$_4$ → AN$_5$ → AN$_6$ → AN$_7$ → AN0$_0$ → AN0$_1$ → .... → AN0$_6$ → AN0$_7$
Start Condition	Same as one-shot mode
Stop Condition	The ADST bit is set to "0" (A/D conversion stopped) by program
Interrupt Request Generation Timing	An interrupt request is generated after each A/D conversion is completed (Set the DUS bit to "1")
Analog Voltage Input Pins	Selectable from AN$_0$ to AN$_7$ → AN15$_0$ to AN15$_7$, AN$_0$ to AN$_7$ → AN0$_0$ to AN0$_7$ or AN$_0$ to AN$_7$→AN2$_0$ to AN2$_7$
Reading of A/D Conversion Result	Do not read the AD00 register. A/D conversion result is stored in the AD00 register after the A/D conversion is completed. DMAC transfers the conversion result to any memory space. Refer to **13. DMAC** for DMAC settings (Set the DUS bit to "1")

18.2 Functions

18.2.1 Resolution Select Function

The BITS bit in the AD0CON1 register determines the resolution. When the BITS bit is set to "1" (10-bit precision), the A/D conversion result is stored into bits 9 to 0 in the AD0j register (j = 0 to 7). When the BITS bit is set to "0" (8-bit precision), the A/D conversion result is stored into bits 7 to 0 in the AD0j register.

18.2.2 Sample and Hold Function

When the SMP bit in the AD0CON2 register is set to "1" (with the sample and hold function), A/D conversion rate per pin increases to 28 $Ø_{AD}$ cycles for 8-bit resolution and 33 $Ø_{AD}$ cycles for 10-bit resolution. The sample and hold function is available in all operating modes. Start the A/D conversion after selecting whether the sample and hold function is to be used or not.

18.2.3 Trigger Select Function

The TRG bit in the AD0CON0 register and the TRG0 bit in the AD0CON2 register select the trigger to start the A/D conversion. Table 18.9 lists settings of the trigger select function.

Table 18.9 Trigger Select Function Settings

Bit and Setting		Trigger
AD0CON0 Register	AD0CON2 Register	
TRG = 0	-	Software trigger The A/D0 starts the A/D conversion when the ADST bit in the AD0CON0 register is set to "1"
TRG = 1[1]	TRG0 = 0	External trigger[2] Falling edge of a signal applied to $\overline{\text{ADTRG}}$
	TRG0 = 1	Hardware trigger[2] The timer B2 interrupt request of three-phase motor control timer functions (after the ICTB2 counter completes counting)

NOTES:

1. A/D0 starts the A/D conversion when the ADST bit is set to "1" (A/D conversion started) and a trigger is generated.
2. The A/D conversion is restarted if an external trigger or a hardware trigger is inserted during the A/D conversion. (The A/D conversion in process is aborted.)

18.2.4 DMAC Operating Mode

DMAC operating mode is available with all operating modes. When the A/D converter is in multi-port single sweep mode or multi-port repeat sweep mode 0, the DMAC operating mode must be used. When the DUS bit in the AD0CON3 register is set to "1" (DMAC operating mode enabled), all A/D conversion results are stored into the AD00 register. DMAC transfers data from the AD00 register to any memory space every time an A/D conversion is completed in each pin. 8-bit DMA transfer must be selected for 8-bit resolution and 16-bit DMA transfer for 10-bit resolution. Refer to **13. DMAC** for instructions.

18.2.5 Extended Analog Input Pins

In one-shot mode and repeat mode, the ANEX0 and ANEX1 pins can be used as analog input pins. The OPA1 and OPA0 bits in the AD0CON1 register select which pins to use as analog input pins. An A/D conversion result for the ANEX0 pin is stored into the AD00 register. The result for the ANEX1 pin is stored into the AD01 register, but is stored into the AD00 register when the DUS bit in the AD0CON3 register is set to "1" (DMAC operating mode enabled).

Set the APS1 and APS0 bits in the AD0CON2 register to "00_2" (AN_0 to AN_7, ANEX0, ANEX1) and the MSS bit in the AD0CON3 register to "0" (multi-port sweep mode disabled).

18.2.6 External Operating Amplifier (Op-Amp) Connection Mode

In external op-amp connection mode, multiple analog voltage can be amplified by one external op-amp using extended analog input pins ANEX0 and ANEX1.

When the OPA1 and OPA0 bits in the AD0CON1 register are set to "11_2" (external op-amp connection), voltage applied to the AN_0 to AN_7 pins are output from ANEX0. Amplify this output signal by an external op-amp and apply it to ANEX1.

Analog voltage applied to ANEX1 is converted to a digital code and the A/D conversion result is stored into the corresponding AD0j register (j=0 to 7). A/D conversion rate varies depending on the response of the external op-amp. The ANEX0 pin cannot be connected to the ANEX1 pin directly.

Set the APS1 and APS0 bits in the AD0CON2 register to "00_2" (AN_0 to AN_7, ANEX0, ANEX1).

Figure 18.7 shows an example of an external op-amp connection.

Table 18.10 Extended Analog Input Pin Settings

AD0CON1 Register		ANEX0 Function	ANEX1 Function
OPA1 Bit	OPA0 Bit		
0	0	Not used	Not used
0	1	$P9_5$ as an analog input	Not used
1	0	Not used	$P9_6$ as an analog input
1	1	Output to an external op-amp	Input from an external op-amp

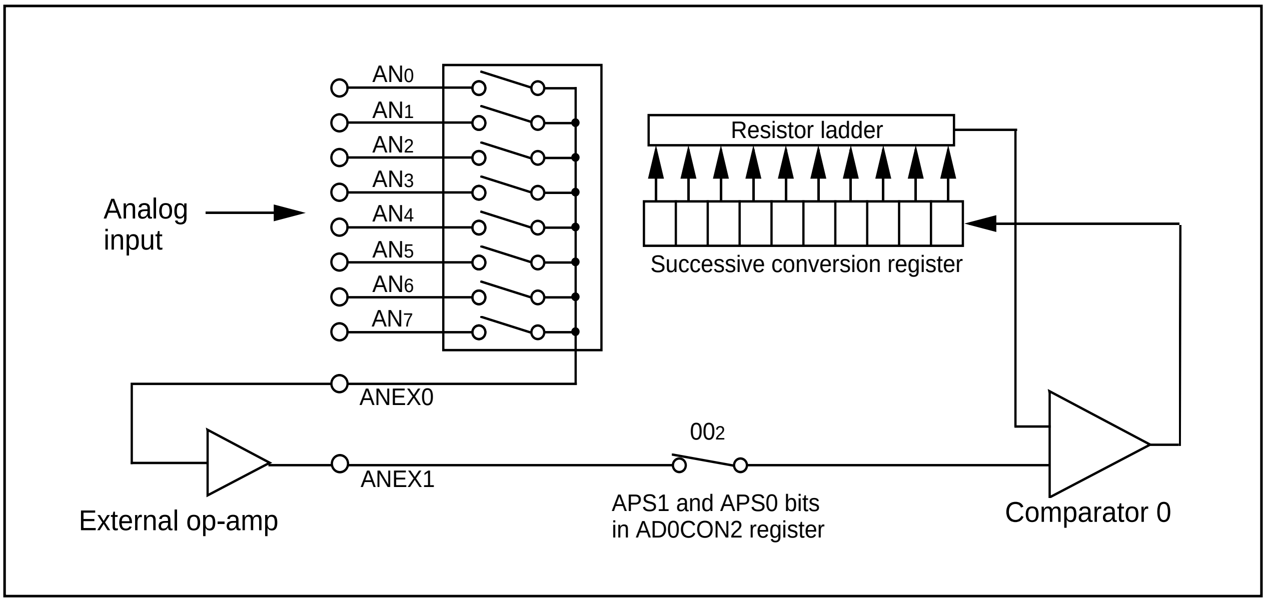


Figure 18.7 External Op-Amp Connection

18.2.7 Power Consumption Reducing Function

When the A/D converter is not used, the VCUT bit in the AD0CON1 register isolates the resistor ladder of the A/D converter from the reference voltage input pin (VREF). Power consumption is reduced by shutting off any current flow into the resistor ladder from the VREF pin.

When using the A/D converter, set the VCUT bit to "1" (VREF connection) before setting the ADST bit in the AD0CON0 register to "1" (A/D conversion started). Do not set the ADST bit and VCUT bit to "1" simultaneously, nor set the VCUT bit to "0" (no VREF connection) during the A/D conversion. The VCUT bit does not affect the VREF performance of the D/A converter.

18.2.8 Output Impedance of Sensor Equivalent Circuit under A/D Conversion

For perfect A/D converter performance, complete internal capacitor (C) charging, shown in Figure 18.8, for the specified period (T) as sampling time. Output Impedance of the sensor equivalent circuit (R0) is determined by the following equations:

$$V_C = V_{IN}\{1 - e^{-\frac{1}{C(R0+R)}t}\}$$

$$\text{When } t = T,\quad V_C = V_{IN} - \frac{X}{Y}V_{IN} = V_{IN}(1 - \frac{X}{Y})$$

$$e^{-\frac{1}{C(R0+R)}T} = \frac{X}{Y}$$

$$-\frac{1}{C(R_0+R)}T = \ln\frac{X}{Y}$$

$$R0 = -\frac{T}{C \bullet \ln\frac{X}{Y}} - R$$

where:

V_C = Voltage between pins

R = Internal resistance of the microcomputer

X = Precision (error) of the A/D converter

Y = Resolution of the A/D converter (1024 in 10-bit mode, and 256 in 8-bit mode)

Figure 18.8 shows analog input pin and external sensor equivalent circuit. The impedance (R_0) can be obtained if the voltage between pins (V_C) changes from 0 to V_{IN}-(0.1/1024) V_{IN} in the time (T), when the difference between V_{IN} and V_C becomes 0.1LSB.

(0.1/1024) means that A/D precision drop, due to insufficient capacitor charge, is held to 0.1LSB at time of A/D conversion in the 10-bit mode. Actual error, however, is the value of absolute precision added to 0.1LSB.

When ØAD = 10 MHz, T = 0.3 µs in the A/D conversion mode with the sample and hold function. Output impedance (R_0) for sufficiently charging capacitor (C) in the time (T) is determined by the following equation:

Using T = 0.3 µs, R = 7.8 kΩ, C = 1.5 pF, X = 0.1, Y = 1024,

$$R0 = -\frac{0.3 \text{ X } 10^{-6}}{1.5 \text{ X } 10^{-12} \bullet \ln\frac{0.1}{1024}} - 7.8 \text{ X}10^{3} = 13.9 \text{ X } 10^{3}$$

Thus, the allowable output impedance of the sensor equivalent circuit, making the precision (error) 0.1LSB or less, is approximately 13.9 kΩ maximum.

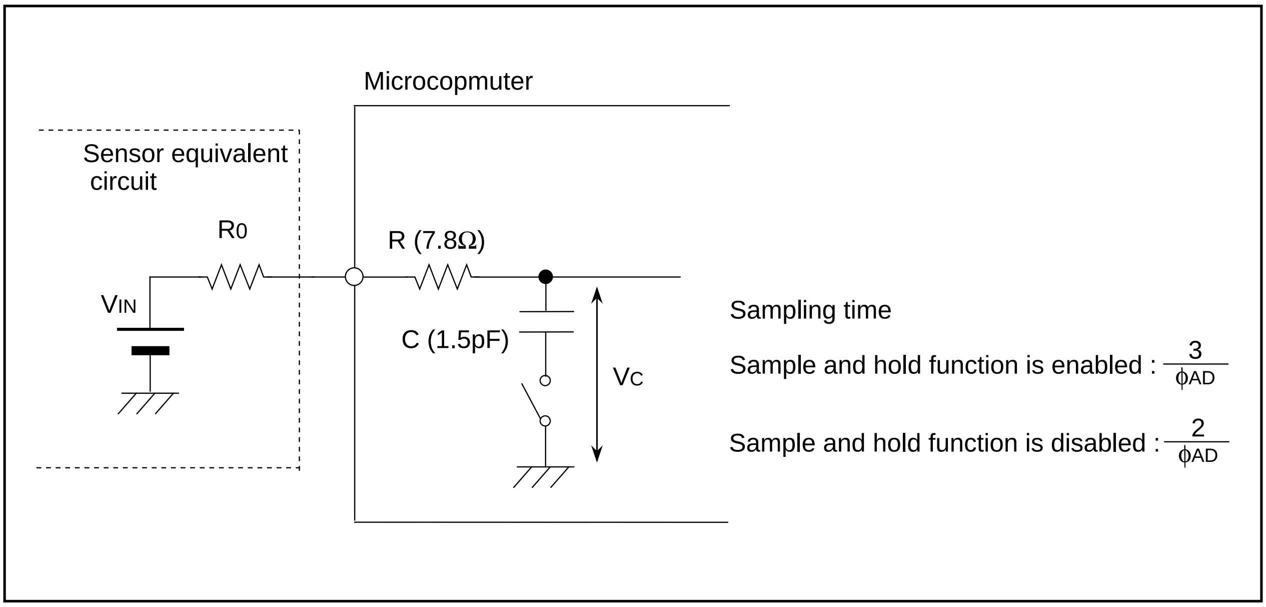

Microcopmuter
Sensor equivalent circuit
R0
VIN
R (7.8Ω)
C (1.5pF)
VC
Sampling time
Sample and hold function is enabled : 3/φAD
Sample and hold function is disabled : 2/φAD

Figure 18.8 Analog Input Pin and External Sensor Equivalent Circuit

19. D/A Converter

The D/A converter consists of two separate 8-bit R-2R ladder D/A converters.

Digital code is converted to an analog voltage when a value is written to the corresponding DAi registers (i=0,1). The DAiE bit in the DACON register determines whether the D/A conversion result output is provided or not. Set the DAiE bit to "1" (output enabled) to disable a pull-up of a corresponding port.

Output analog voltage (V) is calculated from value n (n=decimal) set in the DAi register.

$$V = \frac{V_{REF} \text{ x } n}{256} \quad (n = 0 \text{ to } 255)$$

V_{REF} : reference voltage (not related to VCUT bit setting in the AD0CON1 register)

Table 19.1 lists specifications of the D/A converter. Table 19.2 lists the DA0 and DA1 pin settings. Figure 19.1 shows a block diagram of the D/A converter. Figure 19.2 shows the D/A control register. Figure 19.3 shows a D/A converter equivalent circuit.

When the D/A converter is not used, set the DAi register to "00_{16}" and the DAiE bit to "0" (output disabled).

Table 19.1 D/A Converter Specifications

Item	Specification
D/A Conversion Method	R-2R
Resolution	8 bits
Analog Output Pin	2 channels

Table 19.2 Pin Settings

Port	Function	Bit and Setting		
		PD9 Register(1)	PS3 Register(1)	PSL3 Register
$P9_3$	DA_0 output	PD9_3=0	PS3_3=0	PSL3_3=1
$P9_4$	DA_1 output	PD9_4=0	PS3_4=0	PSL3_4=1

NOTES:

1. Set the PD9 and PS3 registers immediately after the PRC2 bit in the PRCR register is set to "1" (write enable). Do not generate an interrupt or a DMA transfer between the instruction to set the PRC2 bit to "1" and the instruction to set the PD9 and PS3 registers.

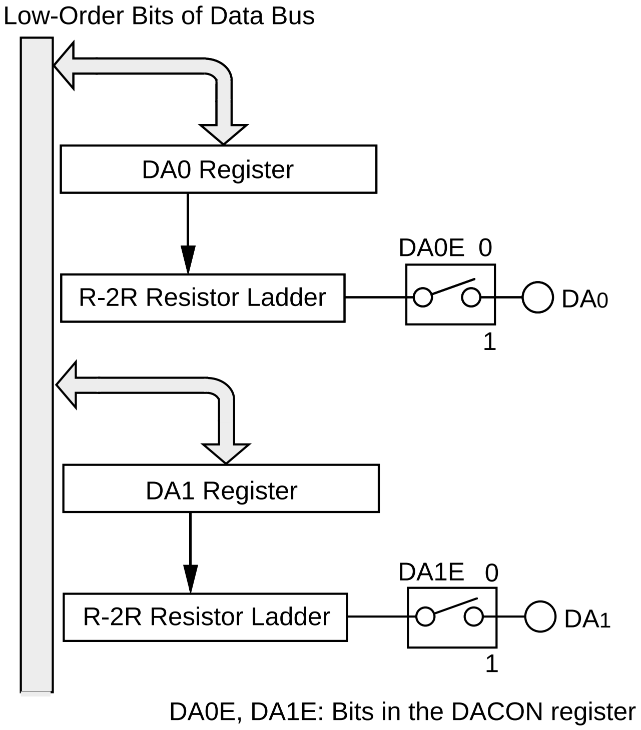
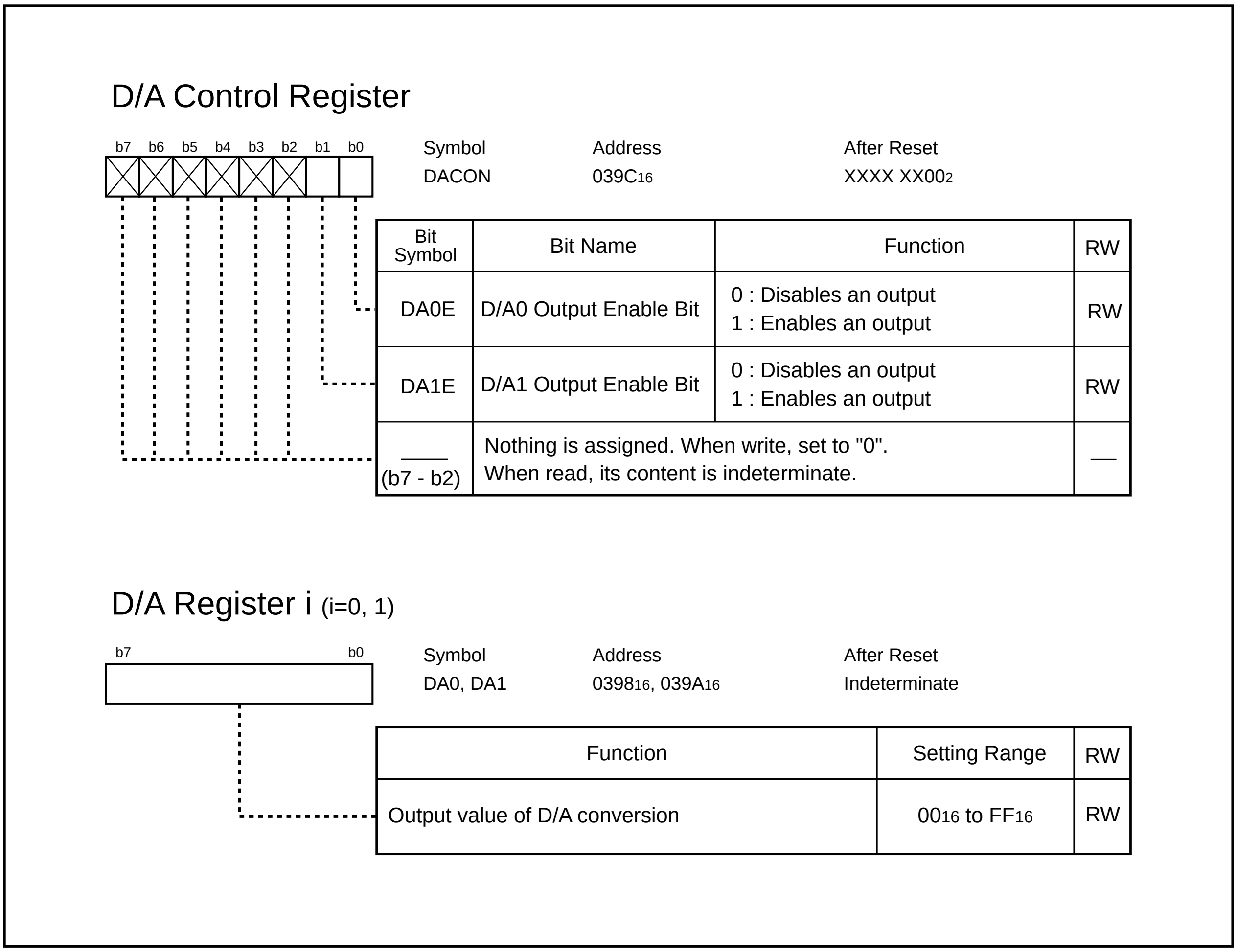


Figure 19.1 D/A Converter



D/A Control Register

Symbol	Address	After Reset
DACON	039C$_{16}$	XXXX XX00$_2$

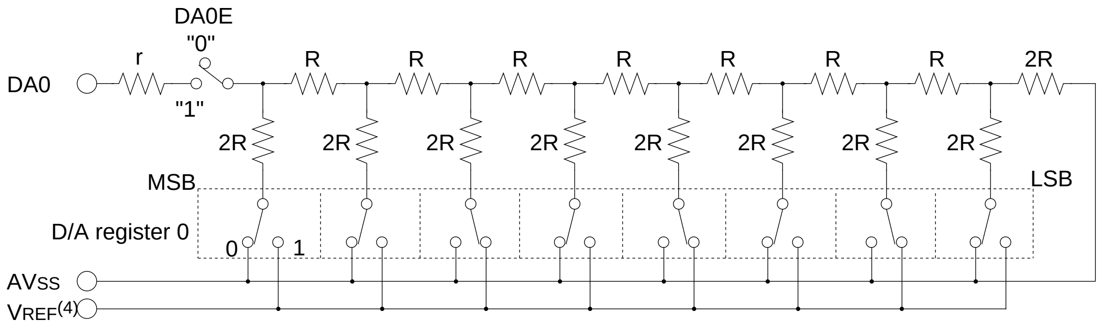
Bit Symbol	Bit Name	Function	RW
DA0E	D/A0 Output Enable Bit	0 : Disables an output 1 : Enables an output	RW
DA1E	D/A1 Output Enable Bit	0 : Disables an output 1 : Enables an output	RW
—— (b7 - b2)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		—

D/A Register i (i=0, 1)

Symbol	Address	After Reset
DA0, DA1	0398$_{16}$, 039A$_{16}$	Indeterminate

Function	Setting Range	RW
Output value of D/A conversion	00$_{16}$ to FF$_{16}$	RW

Figure 19.2 DACON Register, DA0 and DA1 Registers



NOTES:
1. The above applies when the DA0 register is set to "2A$_{16}$".
2. This circuitry is the same for D/A1.
3. To reduce power consumption when the D/A converter is not used, set the DAiE bit (i=0, 1) to "0" (output disabled) and the DAi register to "00$_{16}$" to stop current from flowing into the R-2R resistor.
4. VREF is not related to VCUT bit setting in the AD0CON1 register.

Figure 19.3 D/A Converter Equivalent Circuit

20. CRC Calculation

The CRC (Cyclic Redundancy Check) calculation detects an error in data blocks. A generator polynomial of CRC_CCITT ($X^{16} + X^{12} + X^{5} + 1$) generates CRC code.

The CRC code is a 16-bit code generated for a block of data of desired length. This block of data is in 8-bit units.The CRC code is set in the CRCD register every time one-byte data is transferred to the CRCIN register after a default value is written to the CRCD register. CRC code generation for one-byte data is completed in two cycles.

Figure 20.1 shows a block diagram of a CRC circuit. Figure 20.2 shows CRC-associated registers. Figure 20.3 shows an example of the CRC calculation.

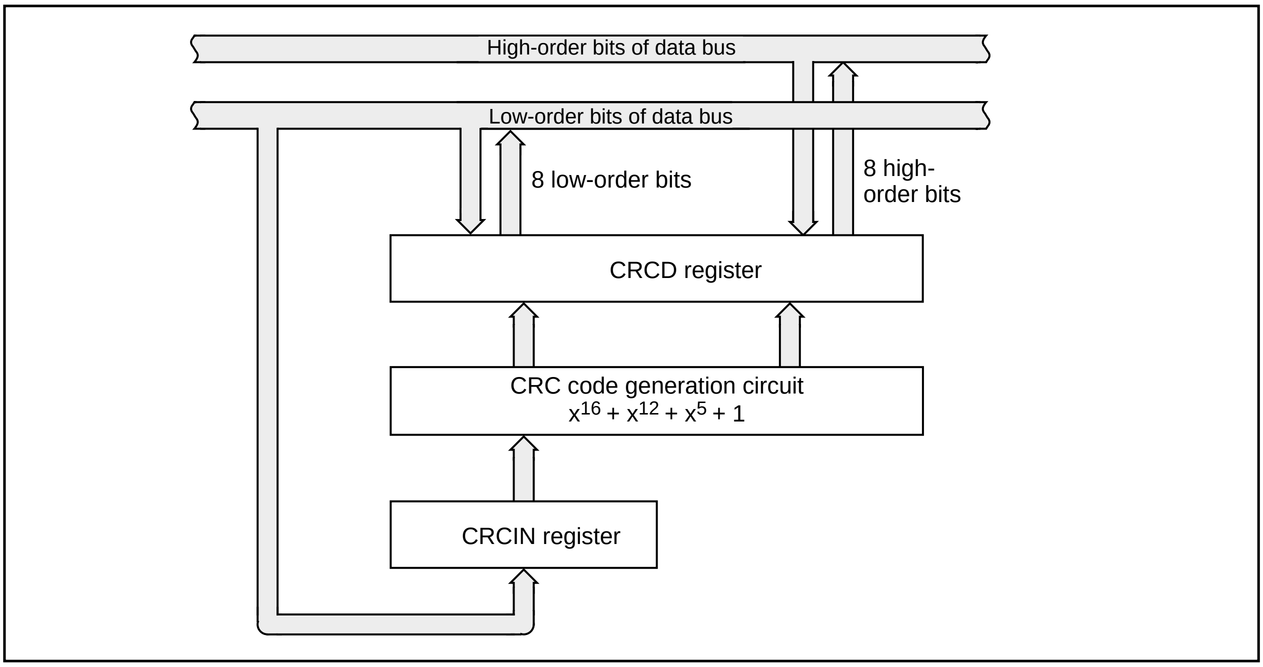
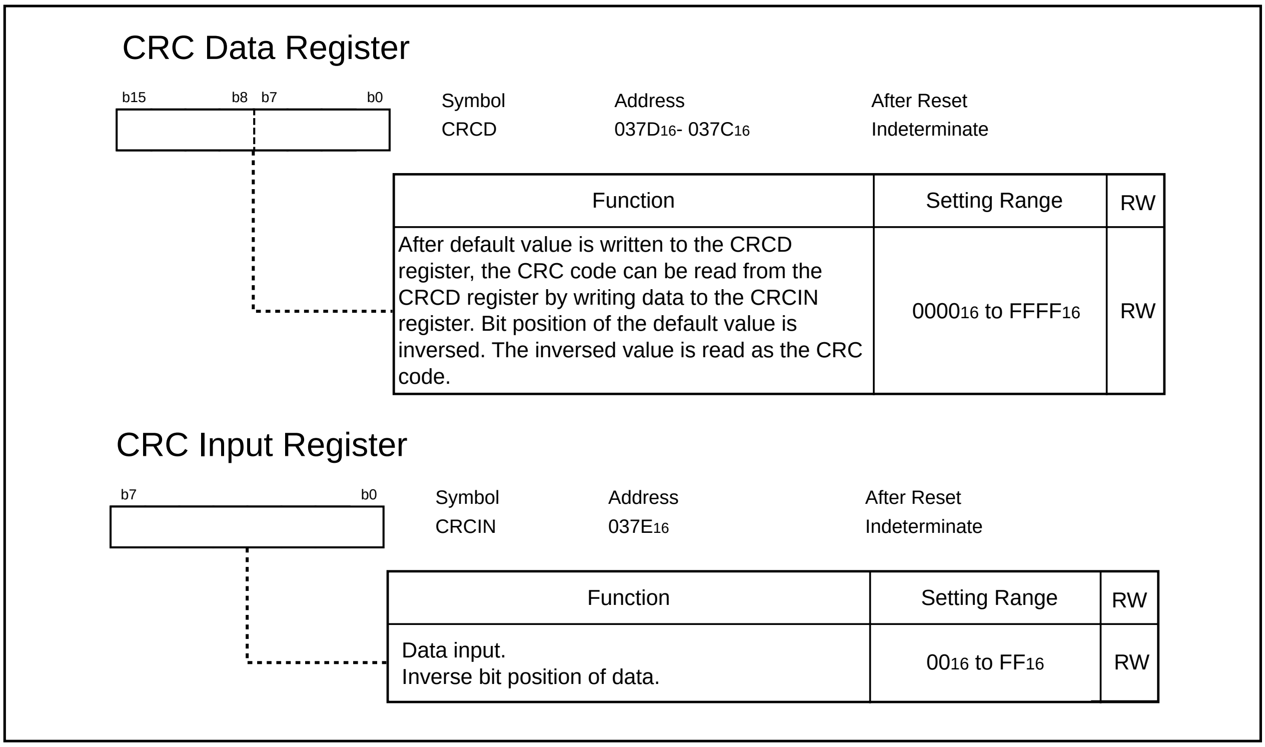

High-order bits of data bus
Low-order bits of data bus
8 low-order bits
8 high-order bits
CRCD register
CRC code generation circuit
$x^{16} + x^{12} + x^{5} + 1$
CRCIN register

Figure 20.1 CRC Calculation Block Diagram



CRC Data Register

Symbol	Address	After Reset
CRCD	$037D_{16}$- $037C_{16}$	Indeterminate

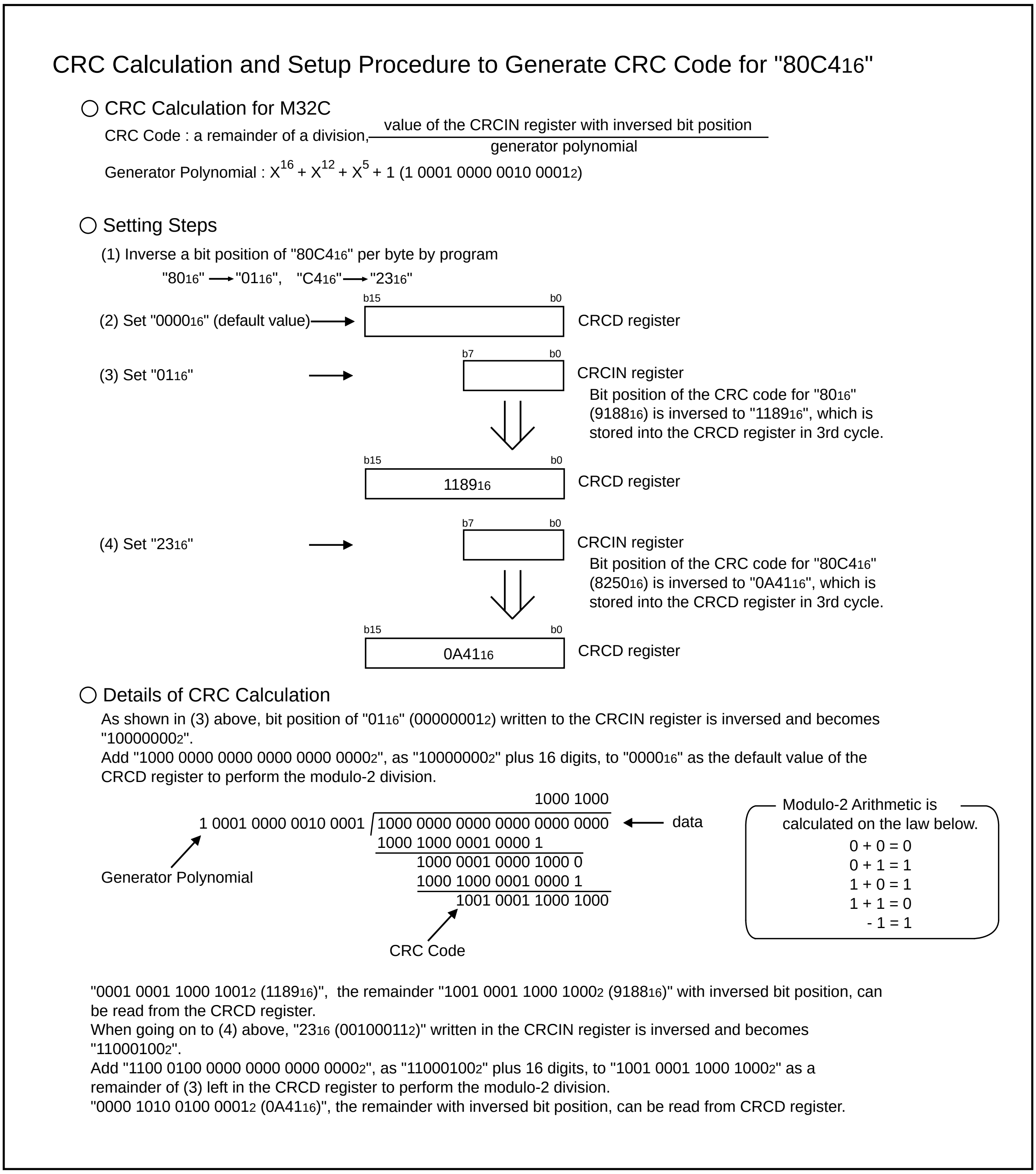
Function	Setting Range	RW
After default value is written to the CRCD register, the CRC code can be read from the CRCD register by writing data to the CRCIN register. Bit position of the default value is inversed. The inversed value is read as the CRC code.	0000_{16} to $FFFF_{16}$	RW

CRC Input Register

Symbol	Address	After Reset
CRCIN	$037E_{16}$	Indeterminate

Function	Setting Range	RW
Data input. Inverse bit position of data.	00_{16} to FF_{16}	RW

Figure 20.2 CRCD Register and CRCIN Register



CRC Calculation and Setup Procedure to Generate CRC Code for "80C4₁₆"

○ CRC Calculation for M32C

CRC Code : a remainder of a division, $\frac{\text{value of the CRCIN register with inversed bit position}}{\text{generator polynomial}}$

Generator Polynomial : $X^{16} + X^{12} + X^{5} + 1$ (1 0001 0000 0010 0001$_2$)

○ Setting Steps

(1) Inverse a bit position of "80C4$_{16}$" per byte by program

"80$_{16}$" → "01$_{16}$", "C4$_{16}$" → "23$_{16}$"

(2) Set "0000$_{16}$" (default value) → CRCD register

(3) Set "01$_{16}$" → CRCIN register

Bit position of the CRC code for "80$_{16}$" (9188$_{16}$) is inversed to "1189$_{16}$", which is stored into the CRCD register in 3rd cycle.

(4) Set "23$_{16}$" → CRCIN register

Bit position of the CRC code for "80C4$_{16}$" (8250$_{16}$) is inversed to "0A41$_{16}$", which is stored into the CRCD register in 3rd cycle.

○ Details of CRC Calculation

As shown in (3) above, bit position of "01$_{16}$" (00000001$_2$) written to the CRCIN register is inversed and becomes "10000000$_2$".
Add "1000 0000 0000 0000 0000 0000$_2$", as "10000000$_2$" plus 16 digits, to "0000$_{16}$" as the default value of the CRCD register to perform the modulo-2 division.

```
                                    1000 1000
1 0001 0000 0010 0001 ) 1000 0000 0000 0000 0000 0000   ← data
                        1000 1000 0001 0000 1
                              1000 0001 0000 1000 0
                              1000 1000 0001 0000 1
                                   1001 0001 1000 1000
```

Modulo-2 Arithmetic is calculated on the law below.
0 + 0 = 0
0 + 1 = 1
1 + 0 = 1
1 + 1 = 0
- 1 = 1

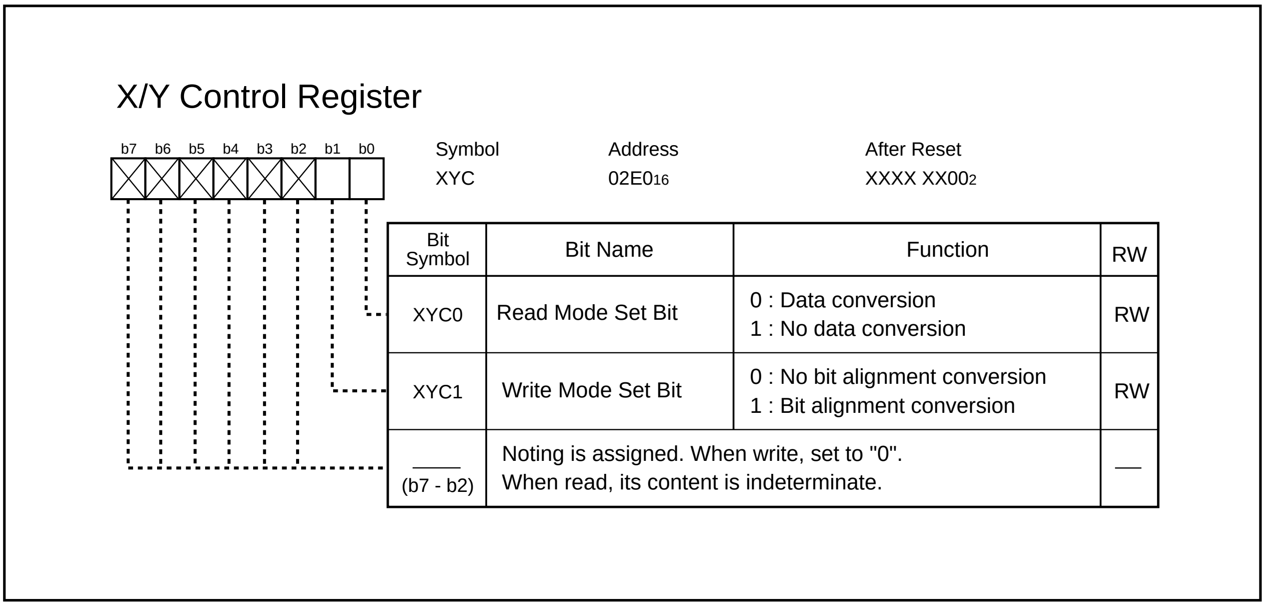
"0001 0001 1000 1001$_2$ (1189$_{16}$)", the remainder "1001 0001 1000 1000$_2$ (9188$_{16}$)" with inversed bit position, can be read from the CRCD register.
When going on to (4) above, "23$_{16}$ (00100011$_2$)" written in the CRCIN register is inversed and becomes "11000100$_2$".
Add "1100 0100 0000 0000 0000 0000$_2$", as "11000100$_2$" plus 16 digits, to "1001 0001 1000 1000$_2$" as a remainder of (3) left in the CRCD register to perform the modulo-2 division.
"0000 1010 0100 0001$_2$ (0A41$_{16}$)", the remainder with inversed bit position, can be read from CRCD register.

Figure 20.3 CRC Calculation

21. X/Y Conversion

The X/Y conversion rotates a 16 x 16 matrix data by 90 degrees and inverses high-order bits and low-order bits of a 16-bit data. Figure 21.1 shows the XYC register.

The 16-bit XiR register (i=0 to 15) and 16-bit YjR register (j=0 to 15) are allocated to the same address. The XiR register is a write-only register, while the YjR register is a read-only register. Access the XiR and YjR registers from an even address in 16-bit units. Performance cannot be guaranteed if the XiR and YiR registers are accessed in 8-bit units.



X/Y Control Register

Symbol	Address	After Reset
XYC	$02E0_{16}$	XXXX XX00_2

Bit Symbol	Bit Name	Function	RW
XYC0	Read Mode Set Bit	0 : Data conversion 1 : No data conversion	RW
XYC1	Write Mode Set Bit	0 : No bit alignment conversion 1 : Bit alignment conversion	RW
—— (b7 - b2)	Noting is assigned. When write, set to "0". When read, its content is indeterminate.		—

Figure 21.1 XYC Register

The XYC0 bit in the XYC register determines how to read the YjR register.

By reading the YjR register when the XYC0 bit is set to "0" (data conversion), bit j in the X0R to X15R registers can be read simultaneously.

For example, bit 0 in the X0R register can be read if reading bit 0 in the Y0R register, bit 0 in the X1R register if reading bit 1 in the Y0R register..., bit 0 in the X14R register if reading bit 14 in the Y0R register and bit 0 in the X15R register if reading bit 15 in the Y0R register.

Figure 21.2 shows the conversion table when the XYC0 bit is set to "0". Figure 21.3 shows an example of the X/Y conversion.

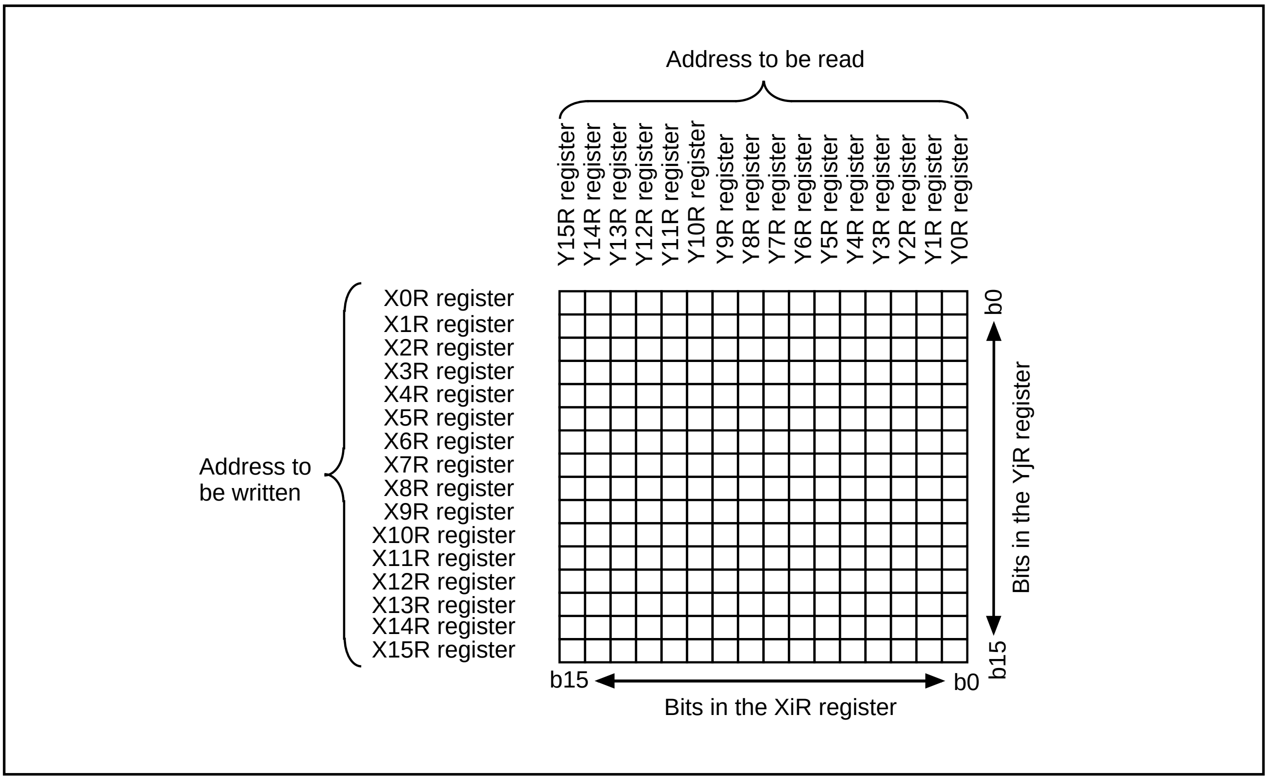


Figure 21.2 Conversion Table when Setting the XYC0 Bit to "0"

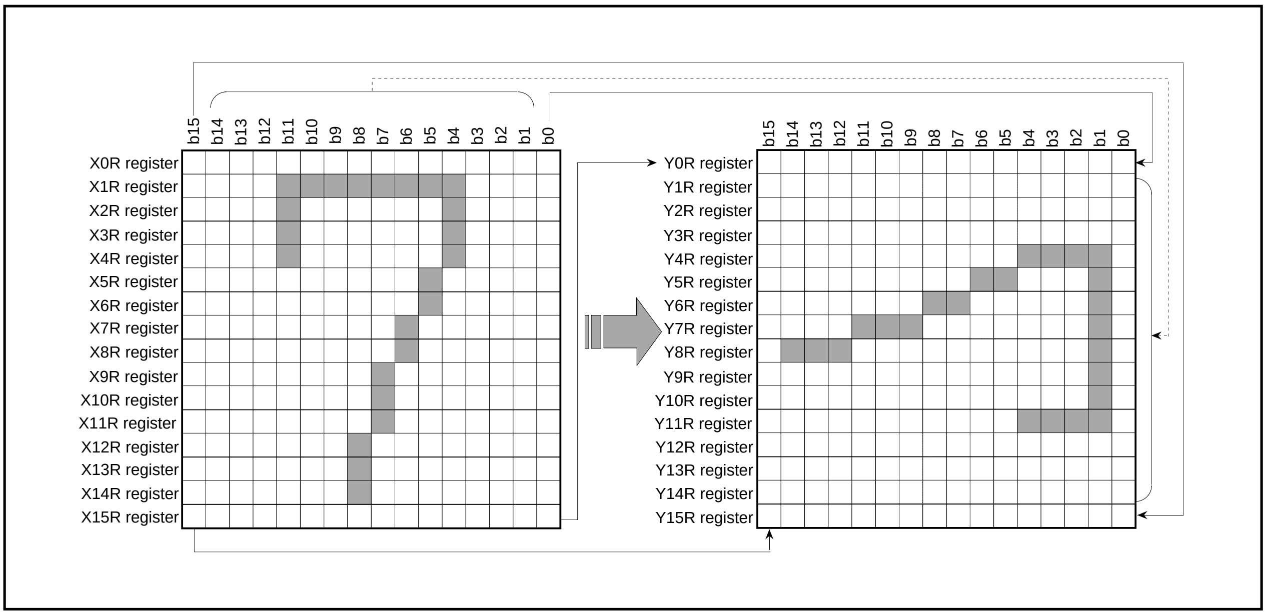


Figure 21.3 X/Y Conversion

By reading the YjR register when the XYC0 bit in the XYC register is set to "1" (no data conversion), the value written to the XiR register can be read directly. Figure 21.4 shows the conversion table when the XYC0 bit is set to "1."

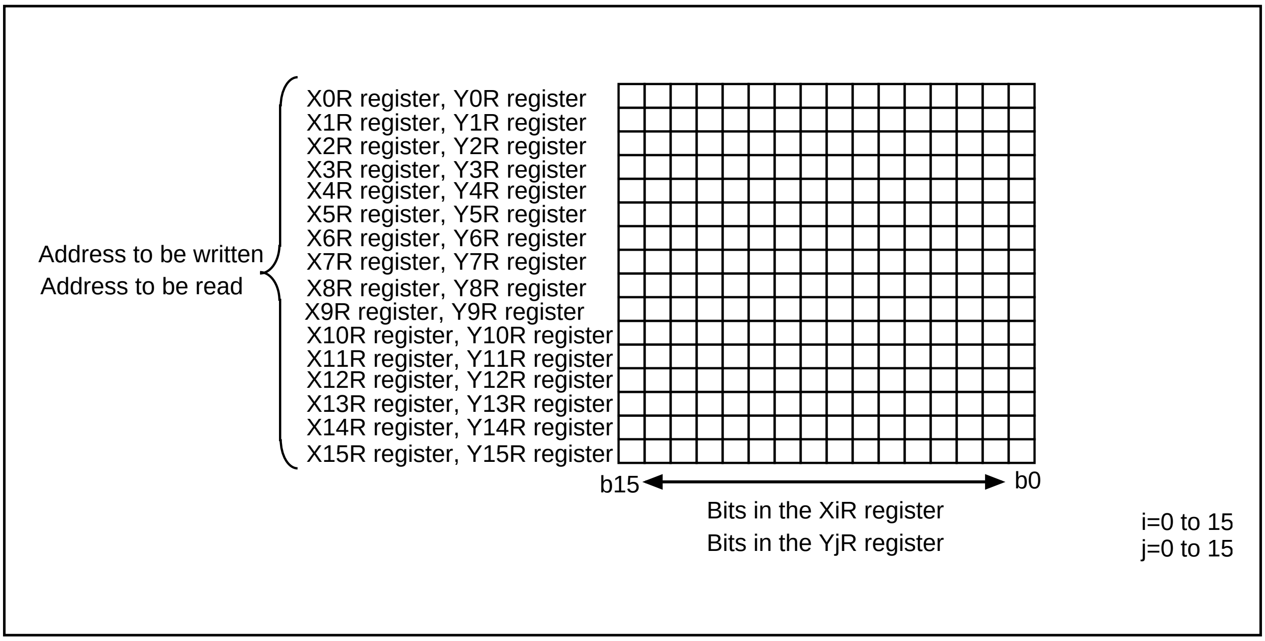

Address to be written
Address to be read
X0R register, Y0R register
X1R register, Y1R register
X2R register, Y2R register
X3R register, Y3R register
X4R register, Y4R register
X5R register, Y5R register
X6R register, Y6R register
X7R register, Y7R register
X8R register, Y8R register
X9R register, Y9R register
X10R register, Y10R register
X11R register, Y11R register
X12R register, Y12R register
X13R register, Y13R register
X14R register, Y14R register
X15R register, Y15R register
b15
b0
Bits in the XiR register
Bits in the YjR register
i=0 to 15
j=0 to 15

Figure 21.4 Conversion Table when Setting the XYC0 Bit to "1"

The XYC1 bit in the XYC register selects bit alignment of the value in the XiR register.
By writing to the XiR register while the XYC1 bit is set to "0" (no bit alignment conversion), bit alignment is written as is. By writing to the XiR register while the XYC1 bit is set to "1" (bit sequence replaced), bit alignment is written inversed.
Figure 21.5 shows the conversion table when the XYC1 bit is set to "1".

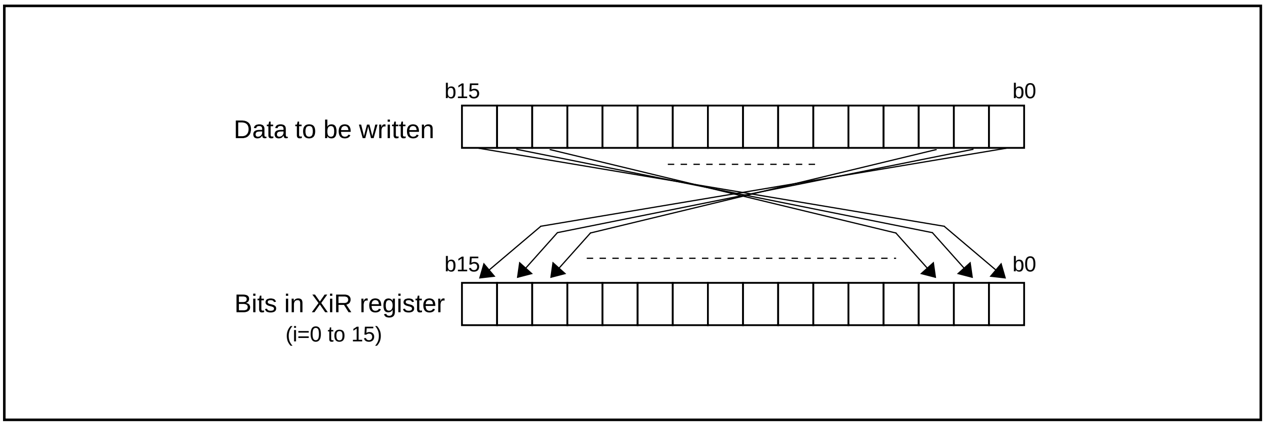


Figure 21.5 Conversion Table when Setting the XYC1 Bit to "1"

22. Intelligent I/O

The intelligent I/O is a multifunctional I/O port for time measurement, waveform generating, clock synchronous serial I/O, clock asynchronous serial I/O (UART), HDLC data processing and more.

The intelligent I/O has one 16-bit base timer for free-running operation, eight 16-bit registers for time measurement and waveform generating and two sets of two 8-bit shift registers for communications.

Table 22.1 lists functions and channels of the intelligent I/O.

Table 22.1 Intelligent I/O Functions and Channels

Function	Description	
Time Measurement[1]	8 channels	
Digital Filter	8 channels	
Trigger Input Prescaler	2 channels (channel 6 and channel 7)	
Trigger Input Gate	2 channels (channel 6 and channel 7)	
Waveform Generating[1]	8 channels	
Single-Phase Waveform Output Mode	8 channels	
Phase-Delayed Waveform Output Mode	8 channels	
SR Waveform Output Mode	8 channels	
Communication	Communication unit 0	Communication unit 1
Clock Synchronous Serial I/O Mode	Available	Available
UART Mode	Not Available	
HDLC Data Processing Mode	Available	
Stepping Motor Control	4 outputs x 4 sets	

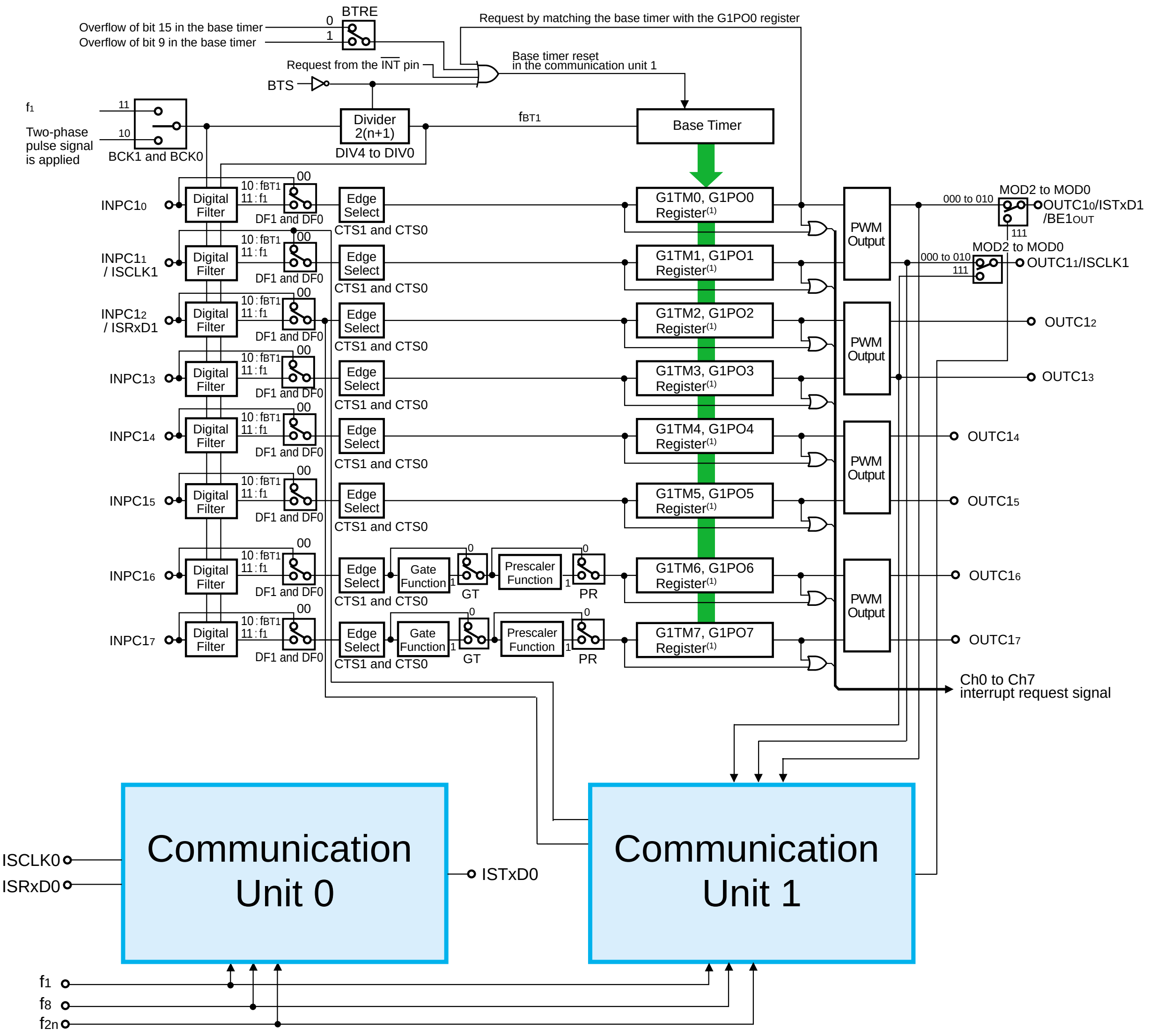
NOTES:

1. The time measurement function and the waveform generating function share a pin.

The time measurement function and waveform generating function can be selected for each channel.

The communication function is available by a combining multiple channels.

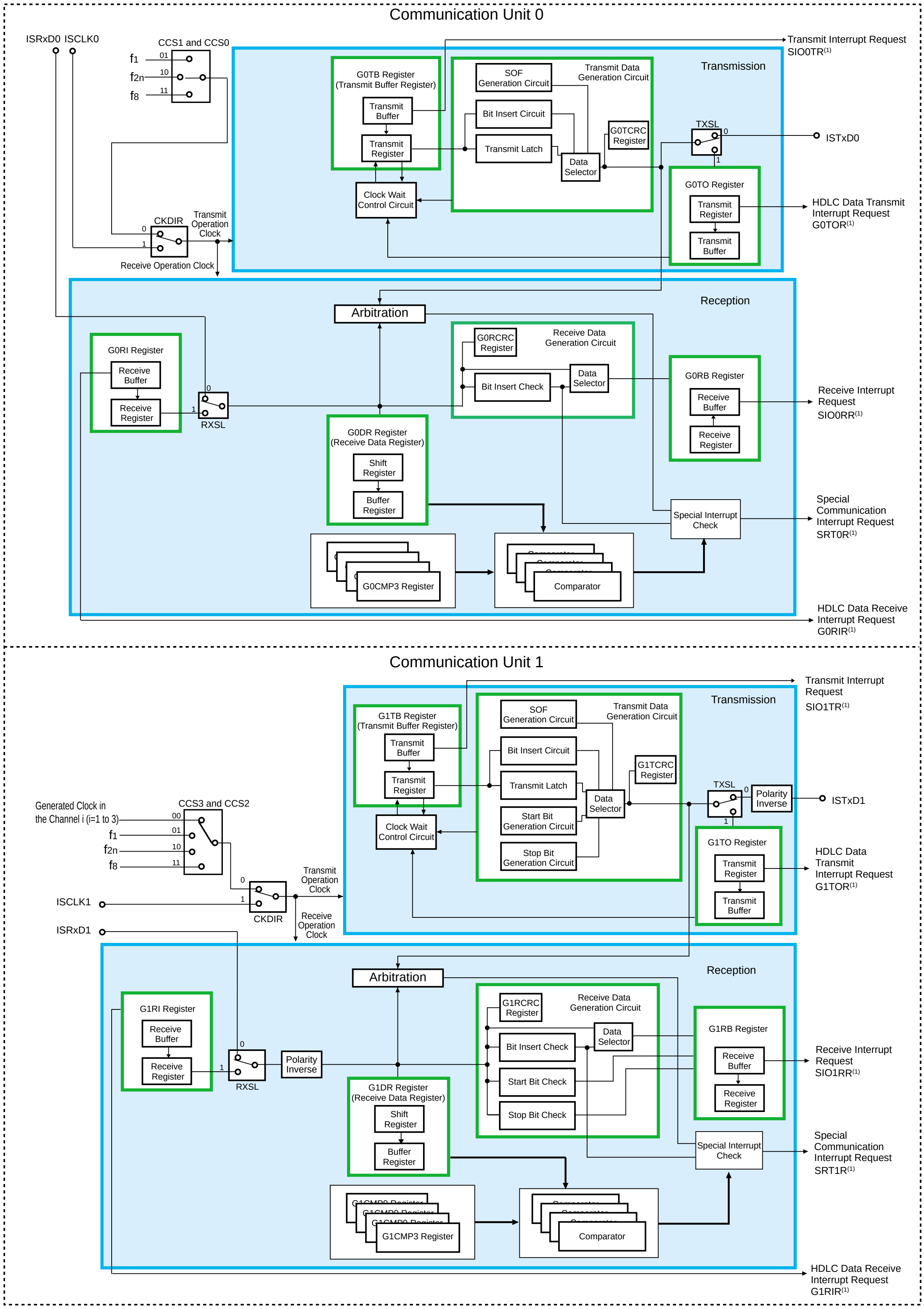
Figures 22.1 shows a block diagram of the intelligent I/O. Figure 22.2 shows a block diagram of the intelligent I/O communication.



NOTES:
1. Each register is placed in a reset state after the G1BCR0 register supplies the clock.

DIV4 to DIV0 bits, BCK1 and BCK0 bits : Bits in the G1BCR0 Register
BTS : Bit in the G1BCR1 Register
CTS1 and CTS0, DF1 and DF0, GT, PR : Bits in the G1TMCRj Register (j = 0 to 7)
MOD2 to MOD0 : Bits in the G1POCRj Register
BTRE : Bit in the G1POCR0 Register

Figure 22.1 Intelligent I/O Block Diagram



NOTES:
1. See Figure 11.14.

CKDIR : Bit in the GiMR Register (i=0,1)
TXSL, RXSL : Bits in the GiEMR Register
CCS1 and CCS0 : Bits in the CCS Register

Figure 22.2 Intelligent I/O Communication Block Diagram

Figures 22.3 to 22.8 show registers associated with the intelligent I/O base timer, the time measurement function and waveform generating function. (For registers associated with the communication function, see Figures 22.19 to 22.28.)

Base Timer Register 1[(2)]

b15 b8 b7 b0

Symbol	Address	After Reset
G1BT	0121_{16} - 0120_{16}	Indeterminate

Function	Setting Range	RW
• When the base timer is counting: When read, the value of the base timer can be read. When write, the counter starts counting from the value written. When the base timer is reset, the G1BT register is set to "0000_{16}"[(1)]. • When the base timer is reset: The G1BT register is set to "0000_{16}" but value is indeterminate. No value is written[(1)].	0000_{16} to $FFFF_{16}$	RW

NOTES:
1. The base timer stops only when the BCK1 and BCK0 bits in the G1BCR0 register are set to "00_2" (clock stopped). The base timer counts when the BCK1 and BCK0 bits are set to a value other than "00_2". When the BTS bit in the G1BCR1 register is set to "0", the base timer is reset continually, remaining set to "0000_{16}". This, in effect, places the base timer in a "no counting" state. When the BTS bit is set to "1", this state is cleared and counting starts.
2. The G1BT register reflects the base timer value after one half f_{BT1} cycle.

Base Timer Control Register 10

b7 b6 b5 b4 b3 b2 b1 b0

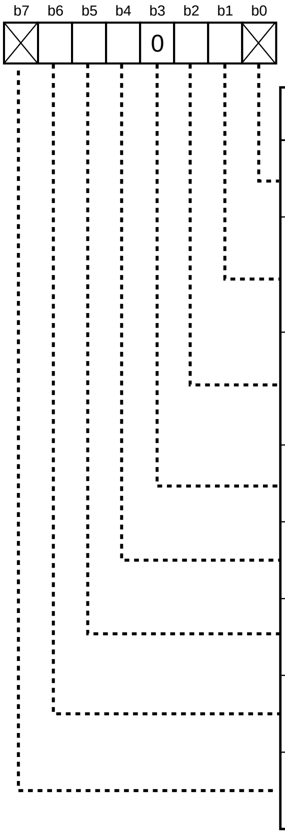
Symbol	Address	After Reset
G1BCR0	0122_{16}	00_{16}

Bit Symbol	Bit Name	Function	RW
BCK0	Count Source Select Bit	b1b0 0 0: Clock stops 0 1: Do not set to this value 1 0: Two-phase pulse signal is applied[(1)] 1 1: f_1	RW
BCK1			RW
DIV0	Count Source Divide Ratio Select Bit	If setting value is n (n = 0 to 31), count source is divided by 2(n + 1). No division if n=31. b6 b5 b4 b3 b2 (n=0) 0 0 0 0 0 : Divide-by-2 (n=1) 0 0 0 0 1 : Divide-by-4 (n=2) 0 0 0 1 0 : Divide-by-6 ⋮ (n=30) 1 1 1 1 0 : Divide-by-62 (n=31) 1 1 1 1 1 : No division	RW
DIV1			RW
DIV2			RW
DIV3			RW
DIV4			RW
IT	Base Timer Interrupt Select Bit	0 : Bit 15 overflows 1 : Bit 14 overflows	RW

NOTES:
1. This setting can be used only when the UD1 and UD0 bits in the G1BCR1 register are set to "10_2" (two-phase signal processing mode). Do not set the BCK1 and BCK0 bits to "10_2" when setting the UD1 and UD0 bits to "00_2" (counter increment mode) or "01_2" (Counter increment/decrement mode).

Figure 22.3 G1BT Register and G1BCR0 Register

Base Timer Control Register 11



Symbol	Address	After Reset
G1BCR1	0123_{16}	$X000\ 000X_2$

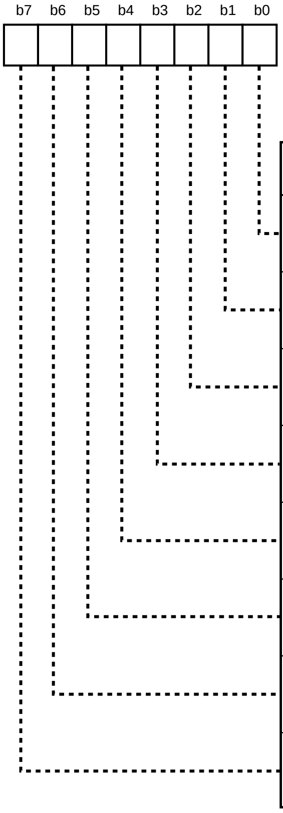
Bit Symbol	Bit Name	Function	RW
(b0)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		–
RST1	Base Timer Reset Cause Select Bit 1	0: The base timer is not reset by matching with the G1PO0 register 1: The base timer is reset by matching with the G1PO0 register[1]	RW
RST2	Base Timer Reset Cause Select Bit 2	0: The base timer is not reset by applying "L" to the $\overline{INT0}$ or $\overline{INT1}$ pin 1: The base timer is reset by applying "L" to the $\overline{INT0}$ or $\overline{INT1}$ pin[2]	RW
(b3)	Reserved Bit	Set to "0"	RW
BTS	Base Timer Start Bit	0: Base timer is reset 1: Base timer starts counting	RW
UD0	Counter Increment/ Decrement Control Bit	b6b5 0 0 : Counter increment mode 0 1 : Counter increment/decrement mode 1 0 : Two-phase pulse signal processing mode[3] 1 1 : Do not set to this value	RW
UD1			RW
(b7)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		–

NOTES:

1. The base timer is reset after two f_{BT1} clock cycles when the base timer value matches the G1PO0 register setting. (See **Figure 22.7** for details on the G1PO0 register.) When the RST1 bit is set to "1", the G1POj register (j=1 to 7) for the waveform generating function and communication function must be set to a value smaller than the G1PO0 register.
2. The IPSA_0 bit in the IPSA register can select the $\overline{INT0}$ or $\overline{INT1}$ pin.
3. In two-phase pulse signal processing mode, the base timer is not reset, even though the RST1 bit is set to "1", if the counter is decremented after two clock cycles when the base timer value matches the G1PO0 register setting.

Figure 22.4 G1BCR1 Register

Time Measurement Control Register 1j (j=0 to 7)



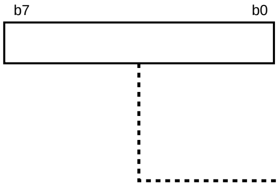
Symbol	Address	After Reset
G1TMCR0 to G1TMCR3	0118_{16}, 0119_{16}, $011A_{16}$, $011B_{16}$	00_{16}
G1TMCR4 to G1TMCR7	$011C_{16}$, $011D_{16}$, $011E_{16}$, $011F_{16}$	00_{16}

Bit Symbol	Bit Name	Function	RW
CTS0	Time Measurement Trigger Select Bit	b1b0 0 0 : No time measurement 0 1 : Rising edge 1 0 : Falling edge 1 1 : Both edges	RW
CTS1			RW
DF0	Digital Filter Function Select Bit	b3b2 0 0 : No digital filter 0 1 : Do not set to this value 1 0 : f_{BT1} 1 1 : f_1	RW
DF1			RW
GT	Gate Function Select Bit(1)	0 : Gate function is not used 1 : Gate function is used	RW
GOC	Gate Function Clear Select Bit(1, 2, 3)	0 : Not cleared 1 : The gate is cleared when the base timer matches the GiPOk register	RW
GSC	Gate Function Clear Bit(1, 2)	The gate is cleared by setting the GSC bit to "1"	RW
PR	Prescaler Function Select Bit(1)	0 : Not used 1 : Used	RW

NOTES:

1. The GT, GOC, GSC and PR bits in the G1TMCR6 and G1TMCR7 registers can be used to select these functions.
   Set all bits 7 to 4 in the G1TMCR0 to G1TMCR5 registers to "0".
2. The GOC and GSC bits are enabled only when the GT bit is set to "1".
3. The GOC bit is set to "0" after the gate function is cleared. See **Figure 22.7** about the G1POk register (k=4 when j=6 and k=5 when j=7).

Time Measurement Prescaler Register 1j (j=6,7)



Symbol	Address	After Reset
G1TPR6, G1TPR7	0124_{16}, 0125_{16}	00_{16}

Function	Setting Range	RW
If the setting value is *n*, the base timer value is stored into G1TMj register whenever a trigger input is counted by *n+1*(1)	00_{16} to FF_{16}	RW

NOTES:

1. The first prescaler, after the PR bit setting in the G1TMCRj register is changed from "0" (not used) to "1" (used), may be divided by *n* rather than *n+1*. The subsequent prescaler is divided by *n+1*.

Figure 22.5 G1TMCR0 to G1TMCR7 Registers, G1TPR6 and G1TPR7 Registers

Time Measurement Register 1j (j=0 to 7)

b15 b8 b7 b0

Symbol	Address	After Reset
G1TM0 to G1TM2	0101_{16} - 0100_{16}, 0103_{16} - 0102_{16}, 0105_{16} - 0104_{16}	Indeterminate
G1TM3 to G1TM5	0107_{16} - 0106_{16}, 0109_{16} - 0108_{16}, $010B_{16}$ - $010A_{16}$	Indeterminate
G1TM6, G1TM7	$010D_{16}$ - $010C_{16}$, $010F_{16}$ - $010E_{16}$	Indeterminate

Function	Setting Range	RW
The base timer value is stored every measurement timing	———	RO

Waveform Generating Control Register 1j (j=0 to 7)

b7 b6 b5 b4 b3 b2 b1 b0

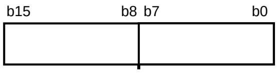
Symbol	Address	After Reset
G1POCR0	0110_{16}	$0000\ X000_2$
G1POCR1 to G1POCR3	0111_{16}, 0112_{16}, 0113_{16}	$0X00\ X000_2$
G1POCR4 to G1POCR7	0114_{16}, 0115_{16}, 0116_{16}, 0117_{16}	$0X00\ X000_2$

Bit Symbol	Bit Name	Function	RW
MOD0	Operating Mode Select Bit	b2b1b0 0 0 0: Single waveform output mode 0 0 1: SR waveform output mode(1) 0 1 0: Phase-delayed waveform output mode 0 1 1: Do not set to this value 1 0 0: Do not set to this value 1 0 1: Do not set to this value 1 1 0: Do not set to this value(2) 1 1 1: Use communication function output(3)	RW
MOD1			RW
MOD2			RW
—— (b3)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		—
IVL	Output Initial Value Select Bit(6)	0: "L" output as default value 1: "H" output as default value	RW
RLD	G1POj Register Value Reload Timing Select Bit	0: Reloads the G1POj register when value is written 1: Reloads the G1POj register when the base timer is reset	RW
BTRE	Base Timer Reset Enable Bit(4)	0: Disables base timer reset when bit 15 in the base timer overflows 1: Enables base timer reset when bit 9 in the base timer overflows(7)	RW
INV	Inverse Output Function Select Bit(5)	0: Output is not inversed 1: Output is inversed	RW

NOTES:
1. This setting is enabled only for even channels. In SR waveform output mode, values written to the corresponding odd channel (next channel after an even channel) are ignored. Even channels provides waveform output. Odd channels provides no waveform output.
2. To receive data in UART mode, set the G1POCR2 register to "$0000\ 0110_2$".
3. This setting is enabled only for channels 0 and 1. To use the ISTxD1 pin, set the MOD2 to MOD0 bits in the G1POCR0 register to "111_2". To use the ISCLK1 pin for an output, set the MOD2 to MOD0 bits in the G1POCR1 register to"111_2". Do not set the MOD2 to MOD0 bits to "111_2" except in channels 0 and 1 and for the communication function.
4. The BTRE bit is provided in the G1POCR0 register only. Set each bit 6 in the G1POCR1 to G1POCR7 registers to "0".
5. The inverse output function is the final step in waveform generating process. When the INV bit is set to "1", an "H" signal is provided a default output by setting the IVL bit to "0"; and an "L" signal is provided by setting it to "1".
6. To provide either "H" or "L" signal output set in the IVL bit, set the FSCj bit in the G1FS register to "0" (waveform generating function selected) and IFEj bit in the G1FE register to "1" (channel j function enabled). Then set the IVL bit to "0" or "1".
7. When the BTRE bit is set to "1", set the BCK1 and BCK0 bits in the G1BCR0 register to "11_2" (f_1) and the UD1 and UD0 bits in the G1BCR1 register to "00_2" (counter increment mode).

Figure 22.6 G1TM0 to G1TM7 Registers and G1POCR0 to G1POCR7 Registers

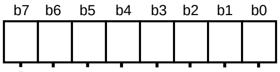
Waveform Generating Register 1j (j=0 to 7)



Symbol	Address	After Reset
G1PO0 to G1PO2	0101$_{16}$-0100$_{16}$, 0103$_{16}$-0102$_{16}$, 0105$_{16}$-0104$_{16}$	Indeterminate
G1PO3 to G1PO5	0107$_{16}$-0106$_{16}$, 0109$_{16}$-0108$_{16}$, 010B$_{16}$-010A$_{16}$	Indeterminate
G1PO6 to G1PO7	010D$_{16}$-010C$_{16}$, 010F$_{16}$-010E$_{16}$	Indeterminate

Function	Setting Range	RW
• When the RLD bit in the G1POCRj register is set to "0", value is reloaded into the G1POj register for output as soon as written, for example, a waveform output, reflecting the value. • When the RLD bit is set to "1", the value is reloaded when the base timer is reset. The value written can be read until reloading.	0000$_{16}$ to FFFF$_{16}$	RW

Function Select Register 1

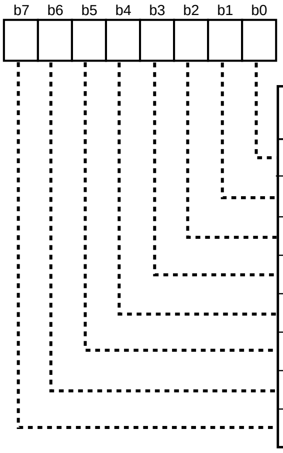


Symbol	Address	After Reset
G1FS	0127$_{16}$	00$_{16}$

Bit Symbol	Bit Name	Function	RW
FSC0	Channel 0 Time Measurement/Waveform Generating Function Select Bit	0 : Selects the waveform generating function 1 : Selects the time measurement function	RW
FSC1	Channel 1 Time Measurement/Waveform Generating Function Select Bit		RW
FSC2	Channel 2 Time Measurement/Waveform Generating Function Select Bit		RW
FSC3	Channel 3 Time Measurement/Waveform Generating Function Select Bit		RW
FSC4	Channel 4 Time Measurement/Waveform Generating Function Select Bit		RW
FSC5	Channel 5 Time Measurement/Waveform Generating Function Select Bit		RW
FSC6	Channel 6 Time Measurement/Waveform Generating Function Select Bit		RW
FSC7	Channel 7 Time Measurement/Waveform Generating Function Select Bit		RW

Figure 22.7 G1PO0 to G1PO7 Registers and G1FS Register

Function Enable Register 1



Symbol	Address	After Reset
G1FE	0126_{16}	00_{16}

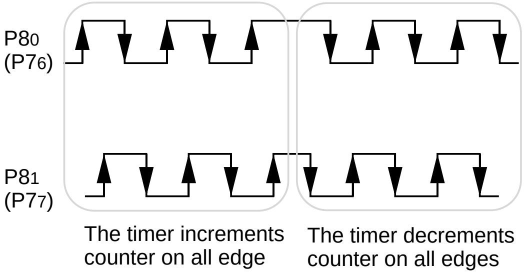
Bit Symbol	Bit Name	Function	RW
IFE0	Channel 0 Function Enable Bit	0 : Disables functions for channel j 1 : Enables functions for channel j (j=0 to 7)	RW
IFE1	Channel 1 Function Enable Bit		RW
IFE2	Channel 2 Function Enable Bit		RW
IFE3	Channel 3 Function Enable Bit		RW
IFE4	Channel 4 Function Enable Bit		RW
IFE5	Channel 5 Function Enable Bit		RW
IFE6	Channel 6 Function Enable Bit		RW
IFE7	Channel 7 Function Enable Bit		RW

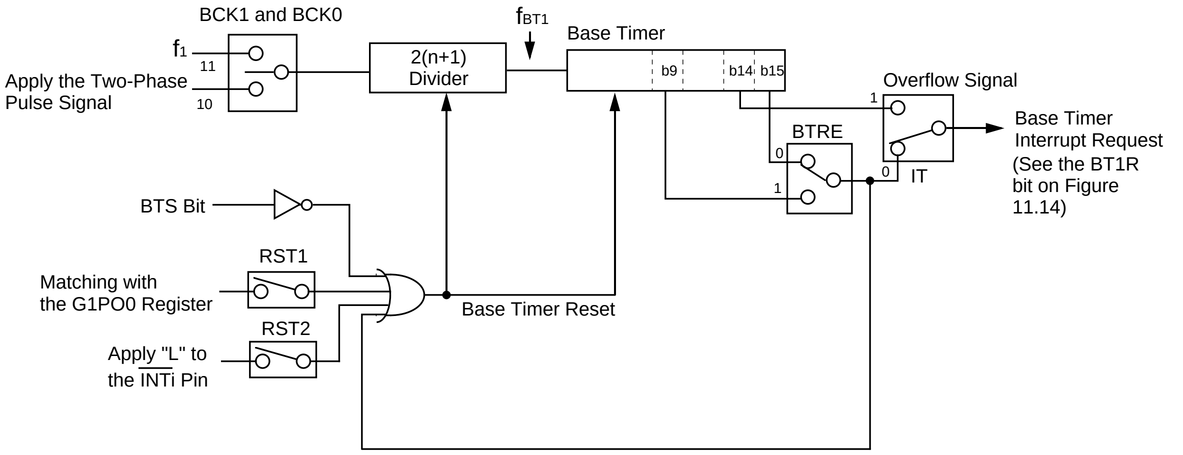
Figure 22.8 G1FE Register

22.1 Base Timer

The base timer is a free-running counter that counts an internally generated count source.
Table 22.2 lists specifications of the base timer. Figures 22.3 and 22.4 show registers associated with the base timer. Figure 22.9 shows a block diagram of the base timer. Figure 22.10 shows an example of the base timer in counter increment mode. Figure 22.11 shows an example of the base timer in counter increment/decrement mode. Figure 22.12 shows an example of two-phase pulse signal processing mode.

Table 22.2 Base Timer Specifications

Item	Specification
Count Source (f_{BT1})	f_1 divided by $2(n+1)$, two-phase pulse input divided by $2(n+1)$ n: determined by the DIV4 to DIV0 bits in the G1BCR0 register n=0 to 31; however no division when n=31
Counting Operation	The base timer increments the counter value The base timer increments and decrements the counter value Two-phase pulse signal processing
Counter Start Condition	The BTS bit in the G1BCR1 register is set to "1" (base timer starts counting)
Counter Stop Condition	The BTS bit in the G1BCR1 register is set to "0" (base timer reset)
Base Timer Reset Condition	• The value of the base timer matches the value of the G1PO0 register • An low-level ("L") signal is applied to the $\overline{INT0}$ or $\overline{INT1}$ pin • Bit 15 or bit 9 in the base timer overflows
Value when the Base Timer is Reset	"0000_{16}"
Interrupt Request	The BT1R bit in the IIO4IR register is set to "1" (interrupt requested) when bit 9, bit 14 or bit 15 in the base timer overflows (See Figure 11.14.)
Read from Base Timer	• The G1BT register indicates the counter value while the base timer is running • The G1BT register is indeterminate when the base timer is reset
Write to Base Timer	When a value is written while the base timer is running, the timer counter immediately starts counting from this value. No value can be written while the base timer is reset
Selectable Function	• Counter increment/decrement mode The base timer starts counting when the BTS bit is set to "1". After reaching to "$FFFF_{16}$", the timer counter is then decremented back to "0000_{16}". If the RST1 bit in the G1BCR1 register is set to "1" (the base timer is reset by matching with the G1PO0 register), the timer counter starts decrementing in two counts after the base timer matches the G1PO0 register. The base timer increments the counter value again when the timer counter reaches "0000_{16}." (See **Figure 22.11**.) • Two-phase pulse processing mode Two-phase pulse signals from $P7_6$ and $P7_7$ pins or $P8_0$ and $P8_1$ pins are counted as well. (See Figure 22.12.) The IPSA_0 bit in the IPSA register controls input pin selection. (Refer to **24. Programmable I/O Ports**)  $P8_0$ ($P7_6$) $P8_1$ ($P7_7$) The timer increments counter on all edge The timer decrements counter on all edges


BCK1 and BCK0
f1
11
10
Apply the Two-Phase Pulse Signal
2(n+1) Divider
fBT1
Base Timer
b9
b14
b15
Overflow Signal
1
0
BTRE
0
1
IT
Base Timer Interrupt Request (See the BT1R bit on Figure 11.14)
BTS Bit
RST1
Matching with the G1PO0 Register
RST2
Apply "L" to the INTi Pin
Base Timer Reset

(i=0,1)

BCK1 and BCK0, IT : Bits in the G1BCR0 register
RST2 to RST0, BTS : Bits in the G1BCR1 register
BTRE : Bit in the G1POCR0 register

Figure 22.9 Base Timer Block Diagram

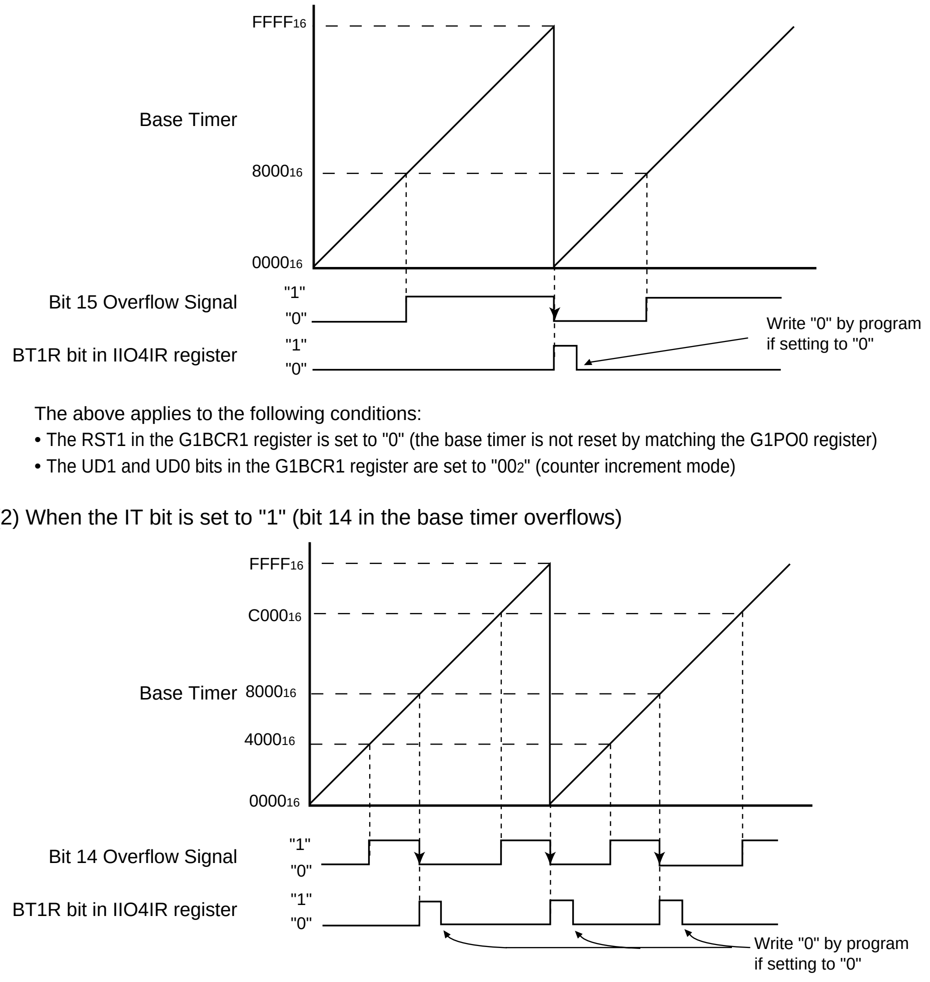
Table 22.3 Base Timer Associated Register Settings

(Also applies when using time measurement function, waveform generating function and communication function)

Register	Bit	Function
G1BCR0	BCK1, BCK0	Select count source
	DIV4 to DIV0	Select divide ratio of count source
	IT	Select the base timer interrupt
G1BCR1	RST2, RST1	Select source for a base timer reset
	BTS	Used to start the base timer independently
	UD1, UD0	Select how to count
G1POCR0	BTRE	Select source for a base timer reset
G1BT	-	Read or write base timer value

Set the following registers to set the RST1 bit to "1" (base timer reset by matching the base timer with the G1PO0 register).

G1POCR0	MOD2 to MOD0	Set to "000$_2$" (single-phase waveform output mode)
G1PO0	-	Set reset cycle
G1FS	FSC0	Set to "0" (waveform generating function)
G1FE	IFE0	Set to "1" (channel operation start)



(1) When the IT bit in the G1BCR0 register is set to "0" (bit 15 in the base timer overflows)

The above applies to the following conditions:

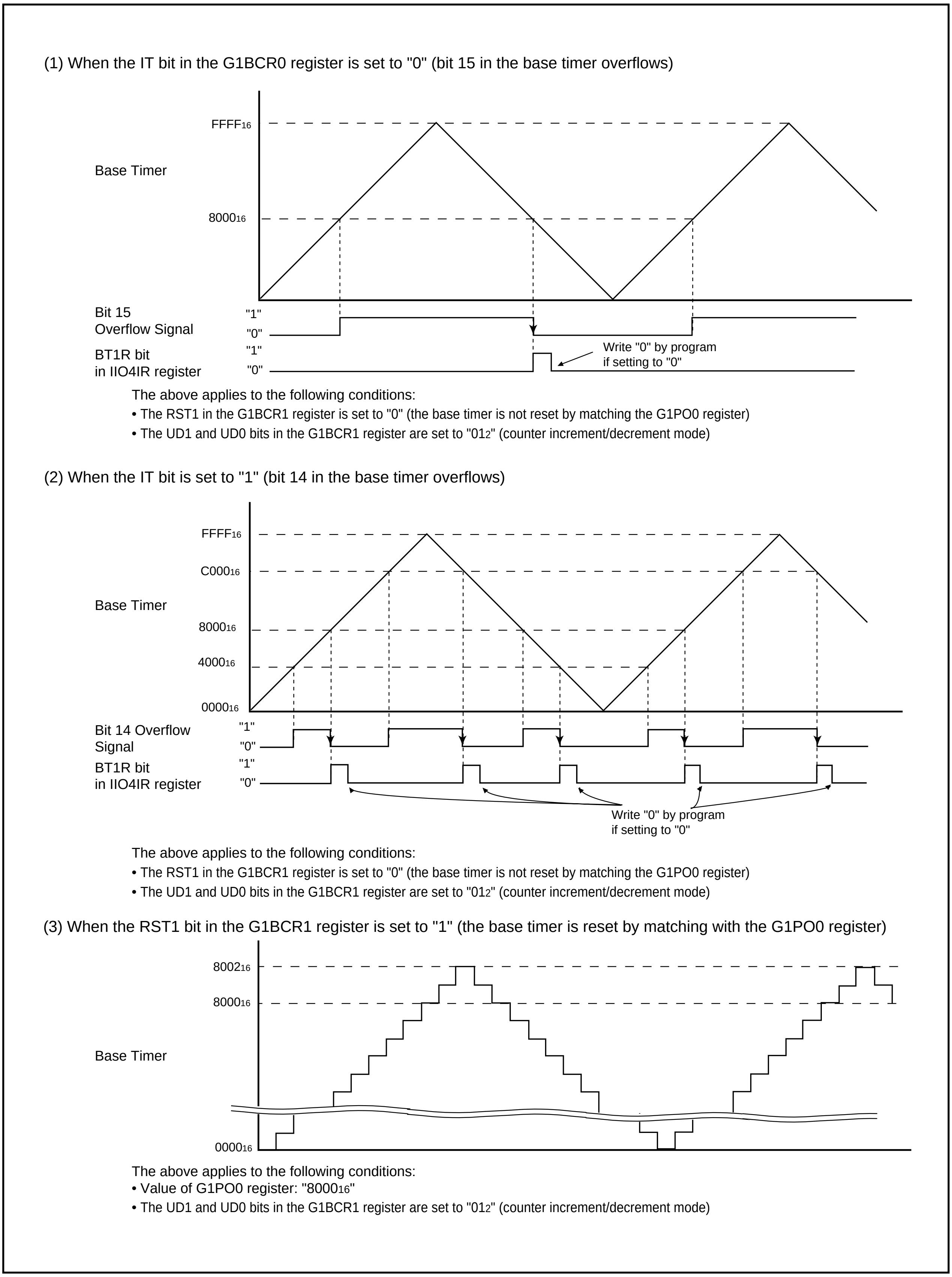
- The RST1 in the G1BCR1 register is set to "0" (the base timer is not reset by matching the G1PO0 register)
- The UD1 and UD0 bits in the G1BCR1 register are set to "00_2" (counter increment mode)

(2) When the IT bit is set to "1" (bit 14 in the base timer overflows)

The above applies to the following conditions:

- The RST1 in the G1BCR1 register is set to "0" (the base timer is not reset by matching the G1PO0 register)
- The UD1 and UD0 bits in the G1BCR1 register are set to "00_2" (counter increment mode)

Figure 22.10 Counter Increment Mode



(1) When the IT bit in the G1BCR0 register is set to "0" (bit 15 in the base timer overflows)

The above applies to the following conditions:

- The RST1 in the G1BCR1 register is set to "0" (the base timer is not reset by matching the G1PO0 register)
- The UD1 and UD0 bits in the G1BCR1 register are set to "01_2" (counter increment/decrement mode)

(2) When the IT bit is set to "1" (bit 14 in the base timer overflows)

The above applies to the following conditions:

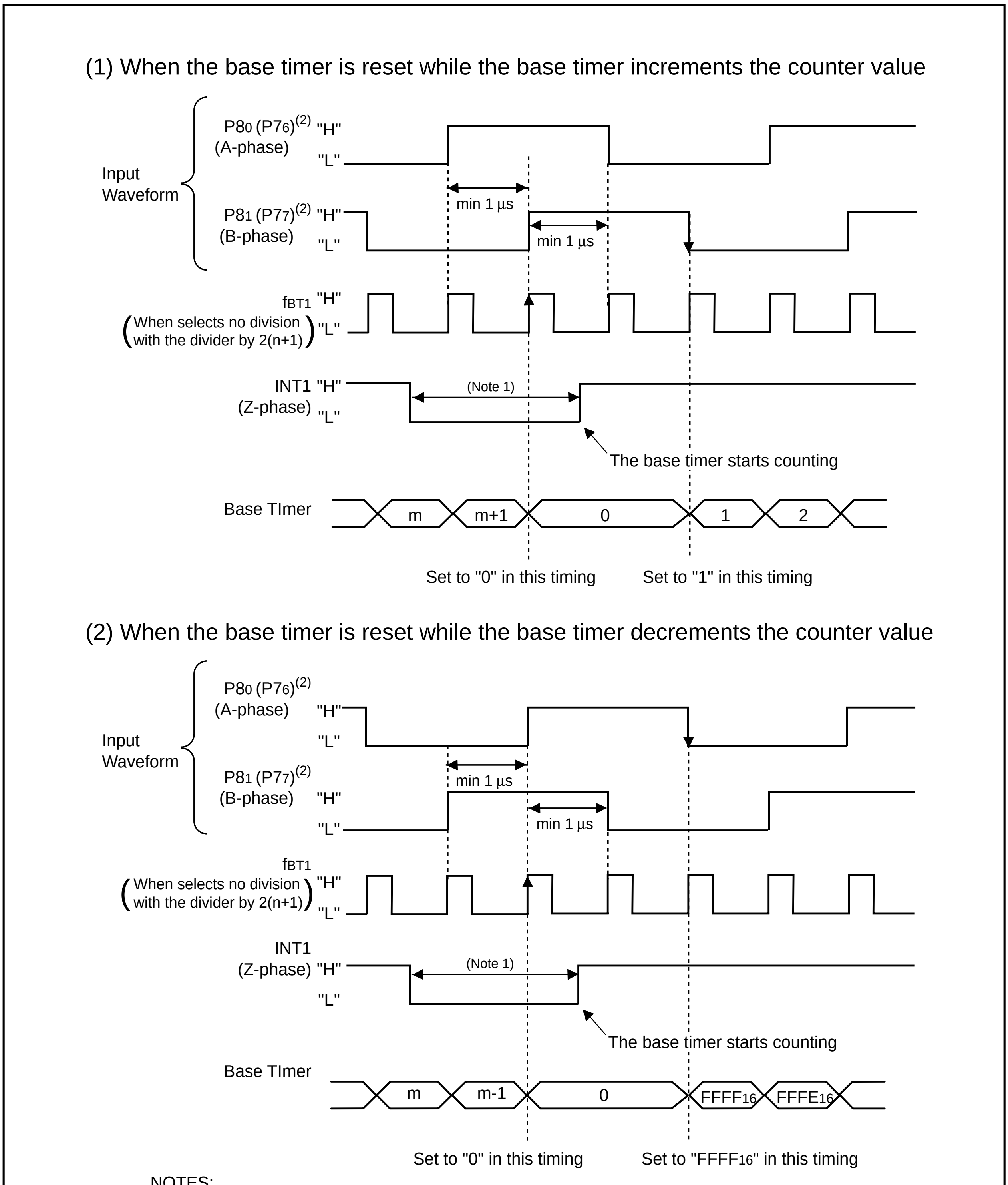
- The RST1 in the G1BCR1 register is set to "0" (the base timer is not reset by matching the G1PO0 register)
- The UD1 and UD0 bits in the G1BCR1 register are set to "01_2" (counter increment/decrement mode)

(3) When the RST1 bit in the G1BCR1 register is set to "1" (the base timer is reset by matching with the G1PO0 register)

The above applies to the following conditions:

- Value of G1PO0 register: "8000_{16}"
- The UD1 and UD0 bits in the G1BCR1 register are set to "01_2" (counter increment/decrement mode)

Figure 22.11 Counter Increment/Decrement Mode


(1) When the base timer is reset while the base timer increments the counter value
Input Waveform
P80 (P76)(2) (A-phase) "H" "L"
P81 (P77)(2) (B-phase) "H" "L"
min 1 µs
min 1 µs
fBT1 (When selects no division with the divider by 2(n+1)) "H" "L"
INT1 (Z-phase) "H" "L"
(Note 1)
The base timer starts counting
Base TImer
m
m+1
0
1
2
Set to "0" in this timing
Set to "1" in this timing
(2) When the base timer is reset while the base timer decrements the counter value
Input Waveform
P80 (P76)(2) (A-phase) "H" "L"
P81 (P77)(2) (B-phase) "H" "L"
min 1 µs
min 1 µs
fBT1 (When selects no division with the divider by 2(n+1)) "H" "L"
INT1 (Z-phase) "H" "L"
(Note 1)
The base timer starts counting
Base TImer
m
m-1
0
FFFF16
FFFE16
Set to "0" in this timing
Set to "FFFF16" in this timing

NOTES:

1. 1.5 fBT1 clock cycles or more are required.
2. Select either port by setting the IPSA_0 bit in the IPSA register.

Figure 22.12 Base Timer Operation in Two-phase Pulse Signal Processing Mode

22.2 Time Measurement Function

When external trigger is applied, the base timer value is stored into the G1TMj register (j=0 to 7). Table 22.4 shows specifications of the time measurement function. Tables 22.5 and 22.6 list pin settings of the time measurement function. Figures 22.13 and 22.14 show operation examples of the time measurement function. Figure 22.15 shows an operation example of the prescaler function and gate function.

Table 22.4 Time Measurement Function Specifications

Item	Specification
Measurement Channel	Channels 0 to 7
Trigger Input Polarity	Rising edge, falling edge and both edges of the INPC1j pin
Measurement Start Condition	The IFEj bit in the G1FE register is set to "1" (channel j function enabled) while the FSCj bit (j=0 to 7) in the G1FS register is set to "1" (time measurement function selected)
Measurement Stop Condition	The IFEj bit is set to "0" (channel j function disabled)
Time Measurement Timing	• No prescaler: every time a trigger signal is applied • Prescaler (for channel 6 and channel 7): every *G1TPRk register (k=6,7) value +1* times a trigger signal is applied
Interrupt Request Generating Timing	The TM1jR bit in the interrupt request register (See Figure 11.14) is set to "1" (interrupt requested) at time measurement timing
INPC1j Pin Function	Trigger input pin
Selectable Function	• Digital filter function The digital filter samples a trigger input signal level every f_1 or f_{BT1} cycles and passes pulse signals, matching trigger input signal level three times • Prescaler function (for channel 6 and channel 7) Time measurement is executed every *G1TPRk register value +1* times a trigger signal is applied • Gate function (for channel 6 and channel 7) After time measurement by the first trigger input, trigger input cannot be accepted. However, while the GOC bit in the G1TMCRk register is set to "1" (gate cleared by matching the base timer with the G1POp register (p=4 when k=6, p=5 when k=7), trigger input can be accepted again by matching the base timer value with the G1POp register setting or by setting the GSC bit in the G1TMCRk register is set to "1"

Table 22.5 Pin Settings for Time Measurement Function

Pin	Bit and Setting		
	PS1, PS2, PS5, PS8 Registers	PD7, PD8, PD11, PD14 Registers	IPS Register
P7$_0$/INPC1$_6$	PS1_0 = 0	PD7_0 = 0	IPS1 = 0
P7$_1$/INPC1$_7$	PS1_1 = 0	PD7_1 = 0	
P7$_3$/INPC1$_0$	PS1_3 = 0	PD7_3 = 0	
P7$_4$/INPC1$_1$	PS1_4 = 0	PD7_4 = 0	
P7$_5$/INPC1$_2$	PS1_5 = 0	PD7_5 = 0	
P7$_6$/INPC1$_3$	PS1_6 = 0	PD7_6 = 0	
P7$_7$/INPC1$_4$	PS1_7 = 0	PD7_7 = 0	
P8$_1$/INPC1$_5$	PS2_1 = 0	PD8_1 = 0	
P11$_0$/INPC1$_0$	PS5_0 = 0	PD11_0 = 0	IPS1 = 1
P11$_1$/INPC1$_1$	PS5_1 = 0	PD11_1 = 0	
P11$_2$/INPC1$_2$	PS5_2 = 0	PD11_2 = 0	
P11$_3$/INPC1$_3$	PS5_3 = 0	PD11_3 = 0	
P14$_0$/INPC1$_4$	PS8_0 = 0	PD14_0 = 0	
P14$_1$/INPC1$_5$	PS8_1 = 0	PD14_1 = 0	
P14$_2$/INPC1$_6$	PS8_2 = 0	PD14_2 = 0	
P14$_3$/INPC1$_7$	PS8_3 = 0	PD14_3 = 0	

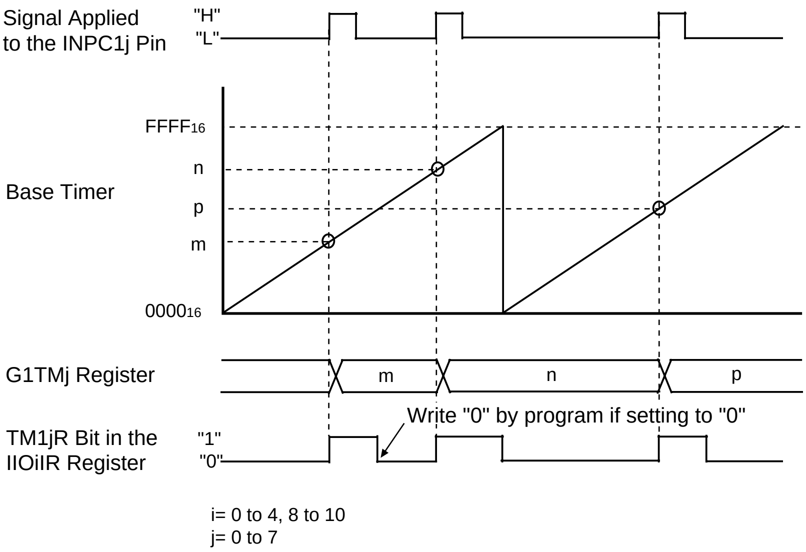
Table 22.6 Time Measurement Function Associated Register Settings

Register	Bit	Function
G1TMCRj	CTS1, CTS0	Select a time measurement trigger
	DF1, DF0	Select the digital filter function
	GT, GOC, GSC	Select the gate function
	PR	Select the prescaler function
G1TPRk	-	Setting value of the prescaler
G1FS	FSCj	Set to "1" (time measurement function)
G1FE	IFEj	Set to "1" (channel j function enabled)

j = 0 to 7 k = 6, 7

Bit configurations and functions vary with channels used.

Registers associated with the time measurement function must be set after setting registers associated with the base timer.

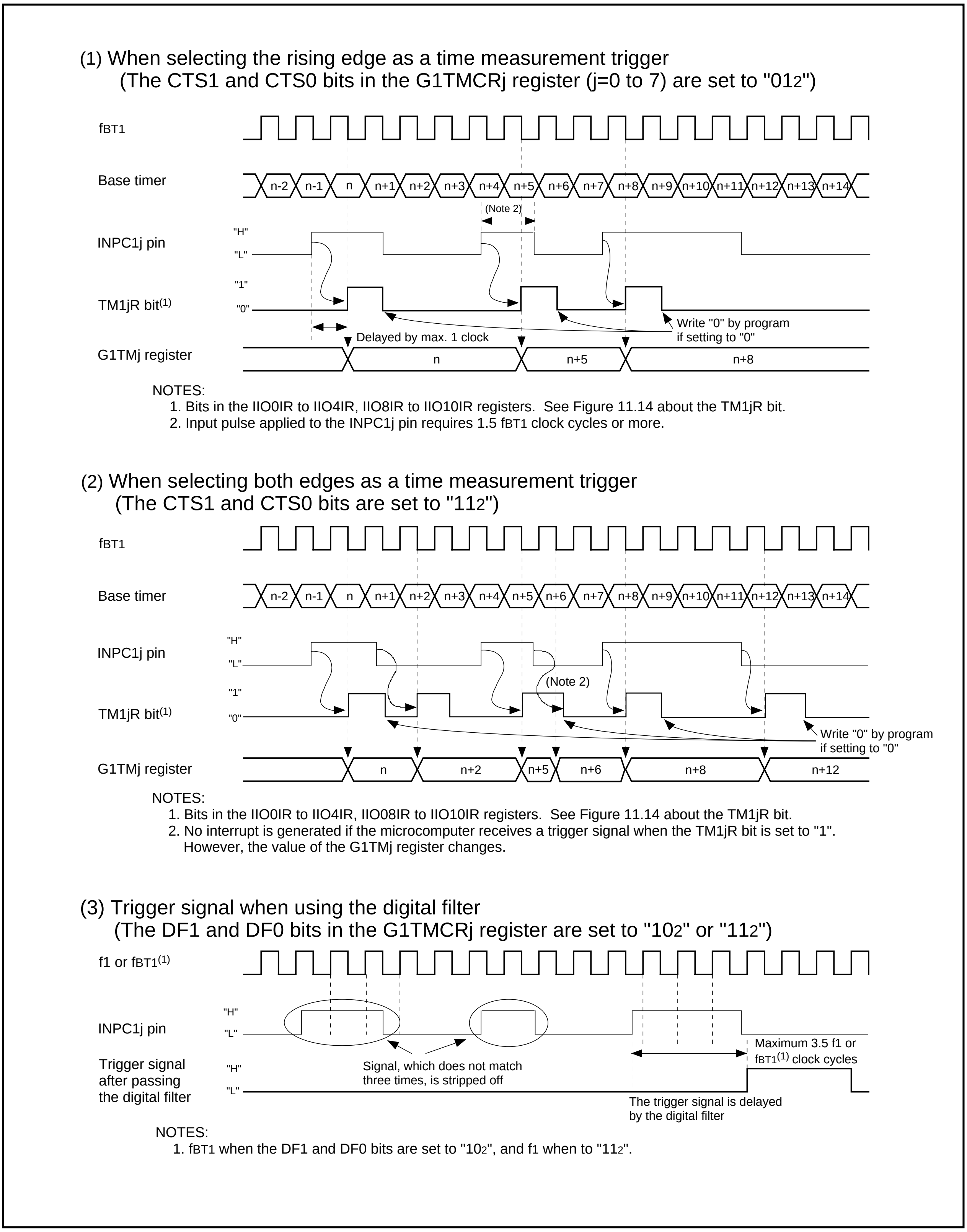

Signal Applied to the INPC1j Pin
"H"
"L"
Base Timer
$FFFF_{16}$
n
p
m
0000_{16}
G1TMj Register
m
n
p
Write "0" by program if setting to "0"
TM1jR Bit in the IIOiIR Register
"1"
"0"
i= 0 to 4, 8 to 10
j= 0 to 7

The above applies to the following conditions:

- The CTS1 and CTS0 bits in the G1TMCRj registers are set to "01_2" (rising edge). The PR bit is set to "0" (no prescaler used) and the GT bit is set to "0" (no gate function used).
- The RST2 and RST1 bits in the G1BCR1 register are set to "00_2" (no base timer reset). The UD1 and UD0 bits are set to "00_2" (counter increment mode).

To set the base timer to "0000_{16}" (setting the RST1 bit to "1" and the RST2 bit to "0") when the base timer value matches the G1PO0 register setting, the base timer is set to "0000_{16}" after it reaches *the G1PO0 register value +2*.

Figure 22.13 Time Measurement Function (1)



(1) When selecting the rising edge as a time measurement trigger
(The CTS1 and CTS0 bits in the G1TMCRj register (j=0 to 7) are set to "01$_2$")

fBT1
Base timer
n-2 n-1 n n+1 n+2 n+3 n+4 n+5 n+6 n+7 n+8 n+9 n+10 n+11 n+12 n+13 n+14
(Note 2)
INPC1j pin
"H"
"L"
TM1jR bit(1)
"1"
"0"
Write "0" by program if setting to "0"
Delayed by max. 1 clock
G1TMj register
n
n+5
n+8

NOTES:
1. Bits in the IIO0IR to IIO4IR, IIO8IR to IIO10IR registers. See Figure 11.14 about the TM1jR bit.
2. Input pulse applied to the INPC1j pin requires 1.5 fBT1 clock cycles or more.

(2) When selecting both edges as a time measurement trigger
(The CTS1 and CTS0 bits are set to "11$_2$")

fBT1
Base timer
n-2 n-1 n n+1 n+2 n+3 n+4 n+5 n+6 n+7 n+8 n+9 n+10 n+11 n+12 n+13 n+14
INPC1j pin
"H"
"L"
(Note 2)
TM1jR bit(1)
"1"
"0"
Write "0" by program if setting to "0"
G1TMj register
n
n+2
n+5
n+6
n+8
n+12

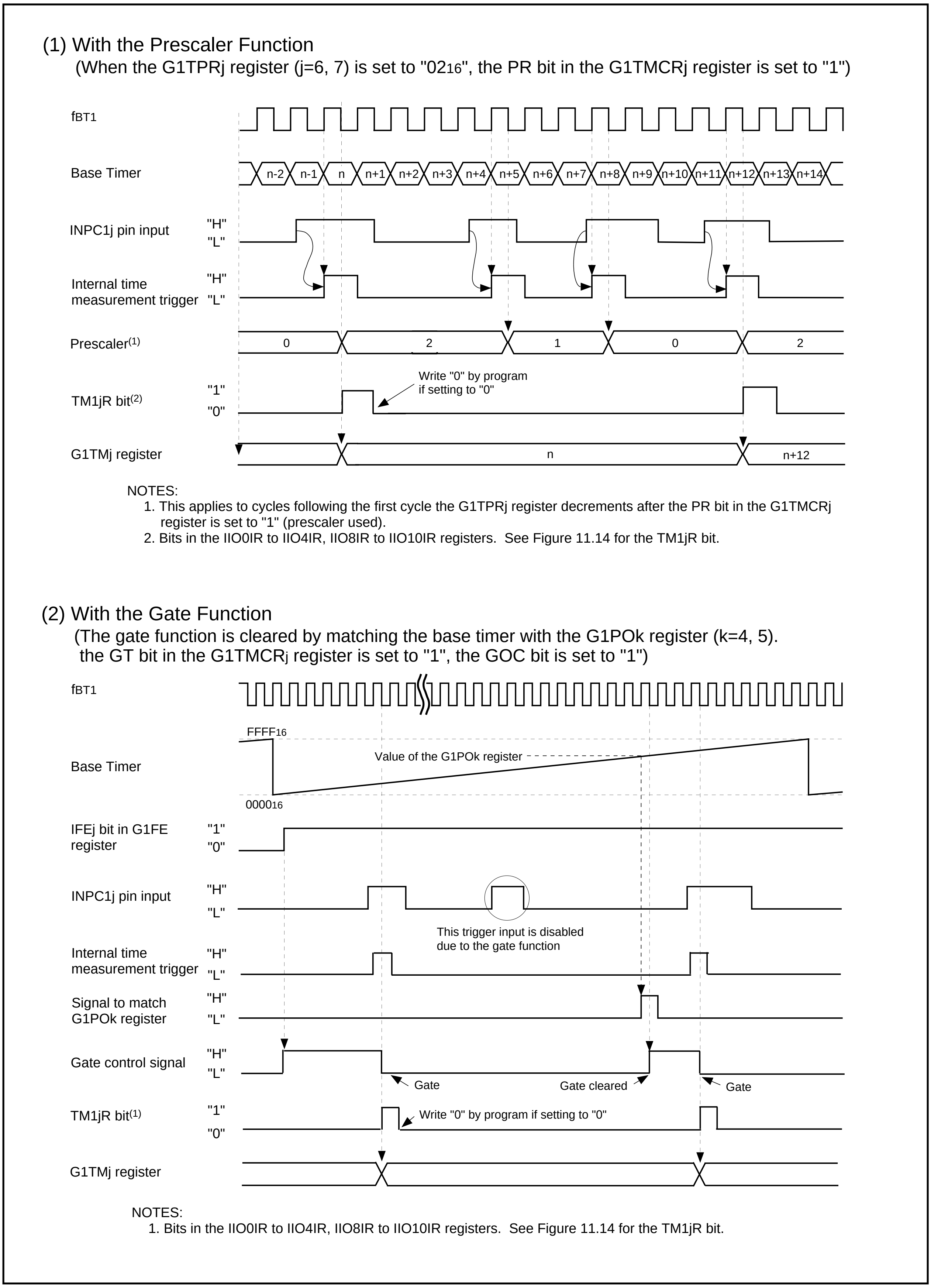
NOTES:
1. Bits in the IIO0IR to IIO4IR, IIO08IR to IIO10IR registers. See Figure 11.14 about the TM1jR bit.
2. No interrupt is generated if the microcomputer receives a trigger signal when the TM1jR bit is set to "1". However, the value of the G1TMj register changes.

(3) Trigger signal when using the digital filter
(The DF1 and DF0 bits in the G1TMCRj register are set to "10$_2$" or "11$_2$")

f1 or fBT1(1)
INPC1j pin
"H"
"L"
Trigger signal after passing the digital filter
"H"
"L"
Signal, which does not match three times, is stripped off
Maximum 3.5 f1 or fBT1(1) clock cycles
The trigger signal is delayed by the digital filter

NOTES:
1. fBT1 when the DF1 and DF0 bits are set to "10$_2$", and f1 when to "11$_2$".

Figure 22.14 Time Measurement Function (2)



(1) With the Prescaler Function
(When the G1TPRj register (j=6, 7) is set to "02_{16}", the PR bit in the G1TMCRj register is set to "1")

NOTES:
1. This applies to cycles following the first cycle the G1TPRj register decrements after the PR bit in the G1TMCRj register is set to "1" (prescaler used).
2. Bits in the IIO0IR to IIO4IR, IIO8IR to IIO10IR registers. See Figure 11.14 for the TM1jR bit.

(2) With the Gate Function
(The gate function is cleared by matching the base timer with the G1POk register (k=4, 5).
the GT bit in the G1TMCRj register is set to "1", the GOC bit is set to "1")

NOTES:
1. Bits in the IIO0IR to IIO4IR, IIO8IR to IIO10IR registers. See Figure 11.14 for the TM1jR bit.

Figure 22.15 Prescaler Function and Gate Function

22.3 Waveform Generating Function

Waveforms are generated when the value of the base timer matches that of the G1POj register (j=0 to 7). The waveform generating function has the following three modes :

- Single-phase waveform output mode
- Phase-delayed waveform output mode
- Set/Reset waveform output (SR waveform output) mode

Table 22.7 lists pin settings of the waveform generating function. Table 22.8 lists registers associated with the waveform generating function.

Table 22.7 Pin Settings for Waveform Generating Function

Pin	Bit and Setting			
	PS1, PS2, PS5 to PS8 Registers	PSL1, PSL2 Registers	PSC, PSC2 Registers	PSD1 Register
P7$_0$/OUTC1$_6$	PS1_0 = 1	PSL1_0 = 0	PSC_0 = 1	PSD1_0=1
P7$_1$/OUTC1$_7$	PS1_1 = 1	PSL1_1 = 0	PSC_1 = 1	PSD1_1=1
P7$_3$/OUTC1$_0$	PS1_3 = 1	PSL1_3 = 0	PSC_3 = 1	-
P7$_4$/OUTC1$_1$	PS1_4 = 1	PSL1_4 = 0	PSC_4 = 1	-
P7$_5$/OUTC1$_2$	PS1_5 = 1	PSL1_5 = 1	-	-
P7$_6$/OUTC1$_3$	PS1_6 = 1	PSL1_6 = 0	PSC_6 = 0	PSD1_6=1
P7$_7$/OUTC1$_4$	PS1_7 = 1	PSL1_7 = 1	-	-
P8$_1$/OUTC1$_5$	PS2_1 = 1	PSL2_1 = 1	PSC2_1=1	-
P11$_0$/OUTC1$_0$	PS5_0 = 1	-	-	-
P11$_1$/OUTC1$_1$	PS5_1 = 1			
P11$_2$/OUTC1$_2$	PS5_2 = 1			
P11$_3$/OUTC1$_3$	PS5_3 = 1			
P14$_0$/OUTC1$_4$	PS8_0 = 1			
P14$_1$/OUTC1$_5$	PS8_1 = 1			
P14$_2$/OUTC1$_6$	PS8_2 = 1			
P14$_3$/OUTC1$_7$	PS8_3 = 1			

Table 22.8 Waveform Generating Function Associated Register Settings

Register	Bit	Function
G1POCRj	MOD2 to MOD0	Select waveform output mode
	IVL	Select default output value
	RLD	Select a timing to reload the value of the G1POj register
	INV	Select if output level is inversed
G1POj	-	Select when output waveform is inversed
G1FS	FSCj	Set to "0" (waveform generating function)
G1FE	IFEj	Set to "1" (channel j function enabled)

j = 0 to 7

Bit configurations and functions vary with channels used.

Registers associated with the waveform generating measurement function must be set after setting registers associated with the base timer.

22.3.1 Single-Phase Waveform Output Mode

Output signal level of the OUTC1j pin becomes high ("H") when the base timer value matches the G1POj register (j=0 to 7) setting. The "H" signal switches to a low-level ("L") signal when the base timer reaches "0000_{16}". If the IVL bit in the G1POCRj register is set to "1" ("H" output as default value), an "H" signal output is provided when waveform output starts. If the INV bit is set to "1" (output inversed), the level of the waveform output is inversed. See Figure 22.16 for details on single-phase waveform output mode operation. Table 22.9 lists specifications of single-phase waveform output mode.

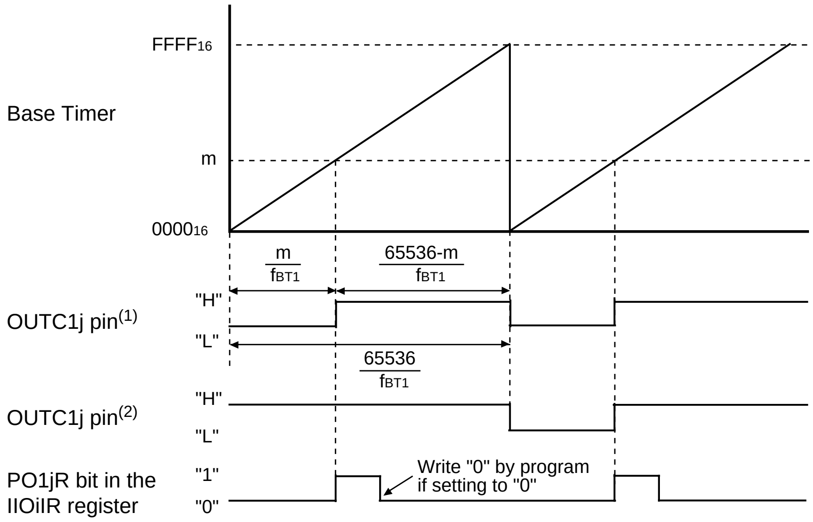
Table 22.9 Single-Phase Waveform Output Mode Specifications

Item	Specification
Output Waveform[2]	• Free-running operation (the RST2 and RST1 bits in the G1BCR1 register are set to "00_2") Cycle : $\frac{65536}{f_{BT1}}$ "L" width : $\frac{m}{f_{BT1}}$ "H" width : $\frac{65536-m}{f_{BT1}}$ m : setting value of the G1POj register (j=0 to 7), 0000_{16} to $FFFF_{16}$ • The base timer is cleared to "0000_{16}" by matching the base timer with the G1PO0 register (the RST1 bit is set to "1" and the RST2 bit is set to "0") Cycle : $\frac{n+2}{f_{BT1}}$ "L" width : $\frac{m}{f_{BT1}}$ "H" width : $\frac{n+2-m}{f_{BT1}}$ m : setting value of the G1POj register (j=1 to 7), 0000_{16} to $FFFF_{16}$ n : setting value of the G1PO0 register, 0001_{16} to $FFFD_{16}$ If m ≥ n+2, the output level is fixed to "L"
Waveform Output Start Condition[1]	The IFEj bit in the G1FE register is set to "1" (channel j function enabled)
Waveform Output Stop Condition	The IFEj bit is set to "0" (channel j function disabled)
Interrupt Request	The PO1jR bit in the interrupt request register is set to "1" (interrupt requested) when the base timer value matches the G1POj register setting. (See Figure 11.14)
OUTC1j Pin	Pulse signal output pin
Selectable Function	• Default value set function: Set starting waveform output level • Inversed output function: Waveform output signal is inversed and provided from the OUTC1j pin

NOTES:

1. Set the FSCj bit in the G1FS register to "0" (waveform generating function selected).
2. When the INV bit in the G1POCRj register is set to "1" (output inversed), the "L" width and "H" width are inversed.

(1) Free-Running Operation
(The RST2 and RST1 bits in the G1BCR1 register are set to "00_2")



i=0 to 4, 8 to 10; j=0 to 7
m : Setting value of the G1POj register, 0000_{16} to $FFFF_{16}$

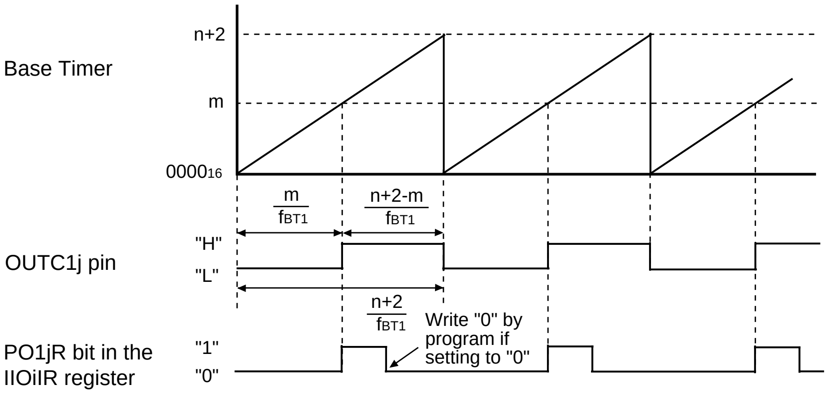
NOTES:

1. Waveform output when the INV bit in the G1POCRj register is set to "0" (not inversed) and the IVL bit in the G1POCRi register is set to "0" (output "L" as default value).
2. Waveform output when the INV bit is set to "0" (not inversed) and the IVL bit is set to "1" ("H" output as default value).

The above applies to the following condition:

- The RST2 and RST1 bits in the G1BCR1 register are set to "00_2" (no base timer reset and the UD1 and UD0 bits in the G1BCR1 register to "00_2" (counter increment mode).

(2) The Base Timer is Reset by Matching the Base Timer with the G1PO0 Regist
(The RST1 bit is set to "1" and the RST2 bit is set to "0")



i=0 to 4, 8 to 10; j=1 to 7
m : Setting value of the G1POj register, 0000_{16} to $FFFF_{16}$
n: Setting value of the G1PO0 register, 0001_{16} to $FFFD_{16}$

The above applies to the following conditions:

- The IVL bit in the G1POCRj register is set to "0" ("L" output as default value) and the INV bit is set to "0" (not inversed).
- The UD1 and UD0 bits are set to "00_2" (counter increment mode).
- $m<n+2$

Figure 22.16 Single-Phase Waveform Output Mode

22.3.2 Phase-Delayed Waveform Output Mode

Output signal level of the OUTC1j pin is inversed every time the base timer value matches the G1POj register (j=0 to 7) setting. Table 22.10 lists specifications of phase-delayed waveform output mode. Figure 22.17 lists an example of phase-delayed waveform output mode operation.

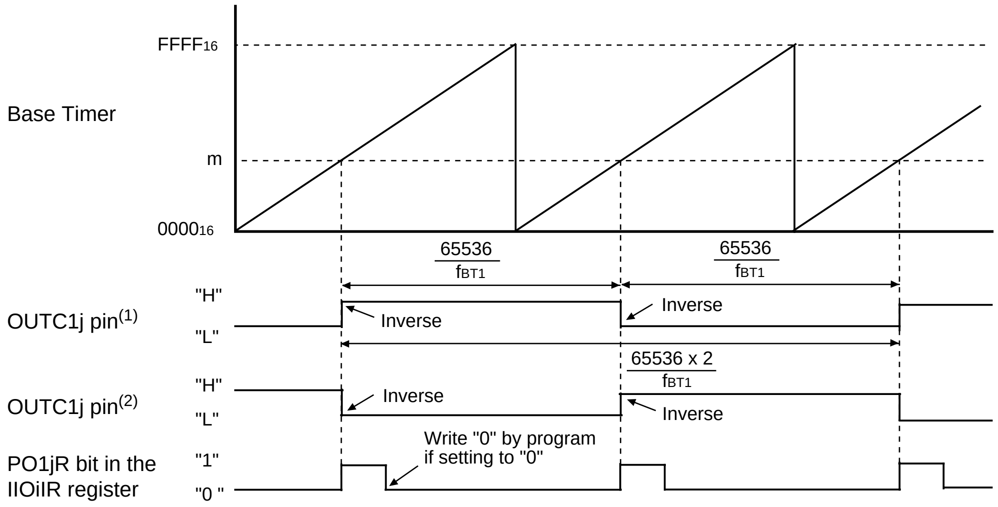
Table 22.10 Phase-Delayed Waveform Output Mode Specifications

Item	Specification
Output Waveform	• Free-running operation (the RST2 and RST1 bits in the G1BCR1 register are set to "00_2") Cycle : $\frac{65536 \times 2}{f_{BT1}}$ "H" and "L" widths : $\frac{65536}{f_{BT1}}$ Setting value of the G1POj (j=0 to 7) register is 0000_{16} to $FFFF_{16}$ • The base timer is cleared to "0000_{16}" by matching the base timer with the G1PO0 register (the RST1 bit is set to "1" and the RST2 bit is set to "0") Cycle : $\frac{2(n+2)}{f_{BT1}}$ "H" and "L" widths : $\frac{n+2}{f_{BT1}}$ n : setting value of the G1PO0 register, 0001_{16} to $FFFD_{16}$ Setting value of the G1POj (j=1 to 7) register is 0000_{16} to $FFFF_{16}$ If G1POj register ≥ n+2, the output level is not inversed
Waveform Output Start Condition[1]	The IFEj bit (j=0 to 7) in the G1FE register is set to "1" (channel j function enabled)
Waveform Output Stop Condition	The IFEj bit is set to "0" (channel j function disabled)
Interrupt Request	The PO1jR bit in the interrupt request register is set to "1" (interrupt requested) when the base timer vslur matches the G1POj register setting. (See Figure 11.14)
OUTC1j Pin	Pulse signal output pin
Selectable Function	• Default value set function: Set starting waveform output level • Inversed output function Waveform output level is inversed to output a waveform from the OUTC1j pin

NOTES:

1. Set the FSCj bit in the G1FS register to "0" (waveform generating function selected).

(1) Free-Running Operation
(The RST2 and RST1 bits in the G1BCR1 register are set to "00_2")

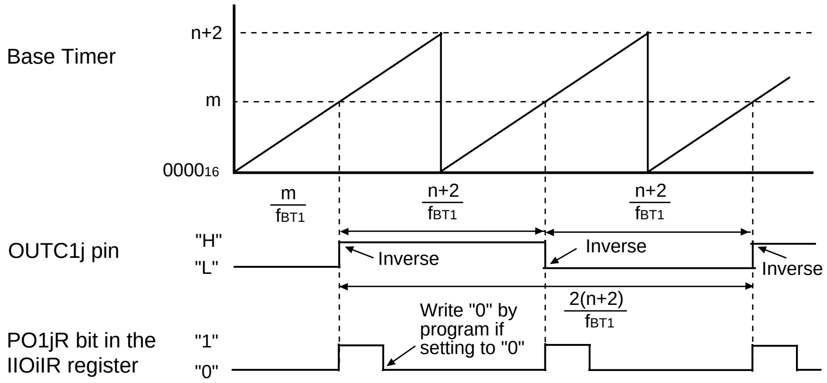


i=0 to 4, 8 to 10; j=0 to 7
m : Setting value of the G1POj register, 0000_{16} to $FFFF_{16}$

NOTES:
1. Waveform output when the INV bit in the G1POCRj register is set to "0" (not inversed) and the IVL bit in the G1POCRj register is set to "0" ("L" output as default value).
2. Waveform output when the INV bit is set to "0" (not inversed) and the IVL bit is set to "1" ("H" output as default value).

The above applies to the following condition:
- The RST2 and RST1 bits in the G1BCR1 register are set to "00_2" (no base timer reset) and the UD1 and UD0 bits in the G1BCR1 register to "00_2" (counter increment mode).

(2) The Base Timer is Reset when the Base Timer Matches the G1PO0 Register
(The RST1 bit is set to "1" and the RST2 bit is set to "0")



i=0 to 4, 8 to 10; j=1 to 7
m : Setting value of the G1POj register, 0000_{16} to $FFFF_{16}$
n: Setting value of the G1PO0 register, 0001_{16} to $FFFD_{16}$

The above applies to the following conditions:
- The IVL bit in the G1POCRj register is set to "0" ("L" output as default value) and the INV bit is set to "0" (not inversed).
- The UD1 and UD0 bits are set to "00_2" (counter increment mode).
- m<n+2

Figure 22.17 Phase-delayed Waveform Output Mode

22.3.3 Set/Reset Waveform Output (SR Waveform Output) Mode

Output signal level of the OUTC1j pin becomes high ("H") when the base timer value matches the G1POj register (j=0, 2, 4, 6) setting. The "H" signal switches to a low-level ("L") signal when the base timer value matches the G1POk register (k=j+1) setting or when the base timer is set to "0000_{16}". If the IVL bit in the G1POCRj register is set to "1" ("H" output as default value), an "H" signal output is provided when waveform output starts. If the INV bit is set to "1" (output inversed), the level of the output waveform is inversed. Table 22.11 lists specifications of SR waveform output mode. Figure 22.18 shows an example of a SR waveform output mode operation.

Table 22.11 SR Waveform Output Mode Specifications

Item	Specification
Output Waveform[2]	• Free-running operation (the RST2 and RST1 bits in the G1BCR1 register are set to "00_2") (1) $m < n$ "H" width : $\frac{n\text{-}m}{f_{BT1}}$ "L" width : $\frac{m}{f_{BT1}}$ [3] + $\frac{65536 - n}{f_{BT1}}$ [4] (2) $m \geq n$ "H" width : $\frac{65536 - m}{f_{BT1}}$ "L" width : $\frac{m}{f_{BT1}}$ m : setting value of the G1POj register (j=0, 2, 4, 6) n : setting value of the G1POk register (k=j+1) • The base timer is cleared to "0000_{16}" by matching the base timer with the G1PO0 register[1] (the RST1 bit is set to "1" and the RST2 bit is set to "0") (1) $m < n < p+2$ "H" width : $\frac{n\text{-}m}{f_{BT1}}$ "L" width : $\frac{m}{f_{BT1}}$ [3] + $\frac{p + 2 - n}{f_{BT1}}$ [4] (2) $m < p+2 \leq n$ "H" width : $\frac{p + 2 - m}{f_{BT1}}$ "L" width : $\frac{m}{f_{BT1}}$ (3) If $m \geq p+2$, the output level is fixed to "L" m : setting value of the G1POj register (j=2, 4, 6), 0000_{16} to $FFFF_{16}$ n : setting value of the G1POk register (k=j+1), 0000_{16} to $FFFF_{16}$ p : setting value of the G1PO0 register, 0001_{16} to $FFFD_{16}$

NOTES:

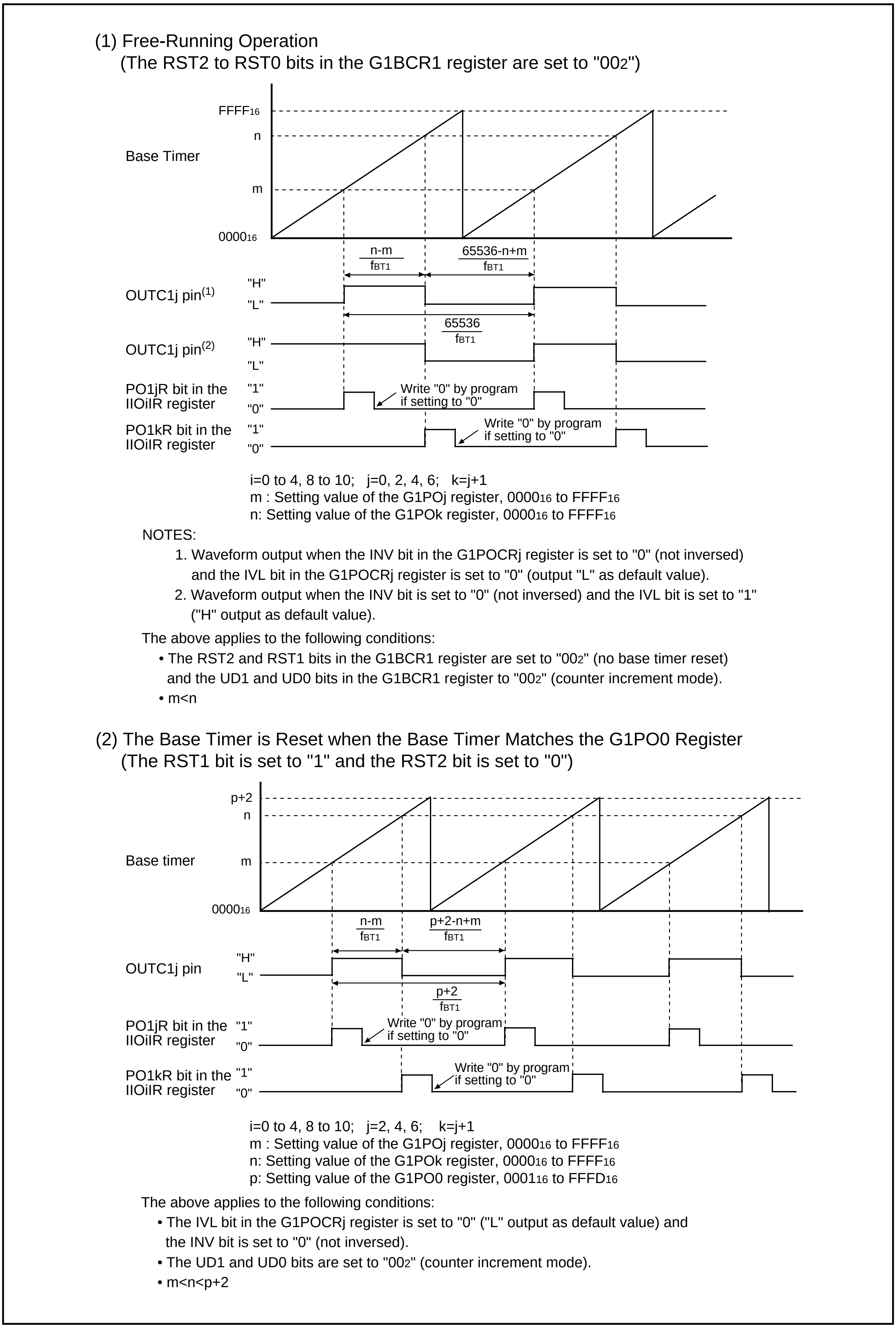
1. When the G1PO0 register resets the base timer, the channel 0 and 1 SR waveform generating functions are not available.
2. When the INV bit in the G1POCRj register is set to "1" (output inversed), the "L" width and "H" width are inversed.
3. Waveform from base timer reset until when output level becomes "H".
4. Waveform from when output level becomes "L" until base timer reset.

Table 22.11 SR Waveform Output Mode Specifications (Continued)

Item	Specification
Waveform Output Start Condition[5]	The IFEq bit (q=0 to 7) in the G1FE register is set to "1" (channel q function enabled)
Waveform Output Stop Condition	The IFEq bit is set to "0" (channel q function disabled)
Interrupt Request	The PO1jR bit in the interrupt request register is set to "1" (interrupt requested) when the value of the base timer matches that of the G1POj register. The PO1kR bit in the interrupt request register is set to "1" (imterrupt requested) when the value of the base timer matches that of the G1POk register. (See Figure 11.14)
OUTC1j Pin	Pulse signal output pin
Selectable Function	• Default value set function: Set starting waveform output level • Inversed output function Waveform output level is inversed to provide a waveform from the OUTC1j pin

NOTES:

5. Set the FSCj bit in the G1FS register to "0" (waveform generating function selected).


FFFF16
n
m
Base Timer
000016
n-m / fBT1
65536-n+m / fBT1
OUTC1j pin(1)
"H"
"L"
65536 / fBT1
OUTC1j pin(2)
"H"
"L"
PO1jR bit in the IIOiIR register
"1"
"0"
Write "0" by program if setting to "0"
PO1kR bit in the IIOiIR register
"1"
"0"
Write "0" by program if setting to "0"
p+2
n
m
Base timer
000016
n-m / fBT1
p+2-n+m / fBT1
OUTC1j pin
"H"
"L"
p+2 / fBT1
PO1jR bit in the IIOiIR register
"1"
"0"
Write "0" by program if setting to "0"
PO1kR bit in the IIOiIR register
"1"
"0"
Write "0" by program if setting to "0"

(1) Free-Running Operation
(The RST2 to RST0 bits in the G1BCR1 register are set to "00_2")

i=0 to 4, 8 to 10; j=0, 2, 4, 6; k=j+1
m : Setting value of the G1POj register, 0000_{16} to $FFFF_{16}$
n: Setting value of the G1POk register, 0000_{16} to $FFFF_{16}$

NOTES:
1. Waveform output when the INV bit in the G1POCRj register is set to "0" (not inversed) and the IVL bit in the G1POCRj register is set to "0" (output "L" as default value).
2. Waveform output when the INV bit is set to "0" (not inversed) and the IVL bit is set to "1" ("H" output as default value).

The above applies to the following conditions:
- The RST2 and RST1 bits in the G1BCR1 register are set to "00_2" (no base timer reset) and the UD1 and UD0 bits in the G1BCR1 register to "00_2" (counter increment mode).
- m<n

(2) The Base Timer is Reset when the Base Timer Matches the G1PO0 Register
(The RST1 bit is set to "1" and the RST2 bit is set to "0")

i=0 to 4, 8 to 10; j=2, 4, 6; k=j+1
m : Setting value of the G1POj register, 0000_{16} to $FFFF_{16}$
n: Setting value of the G1POk register, 0000_{16} to $FFFF_{16}$
p: Setting value of the G1PO0 register, 0001_{16} to $FFFD_{16}$

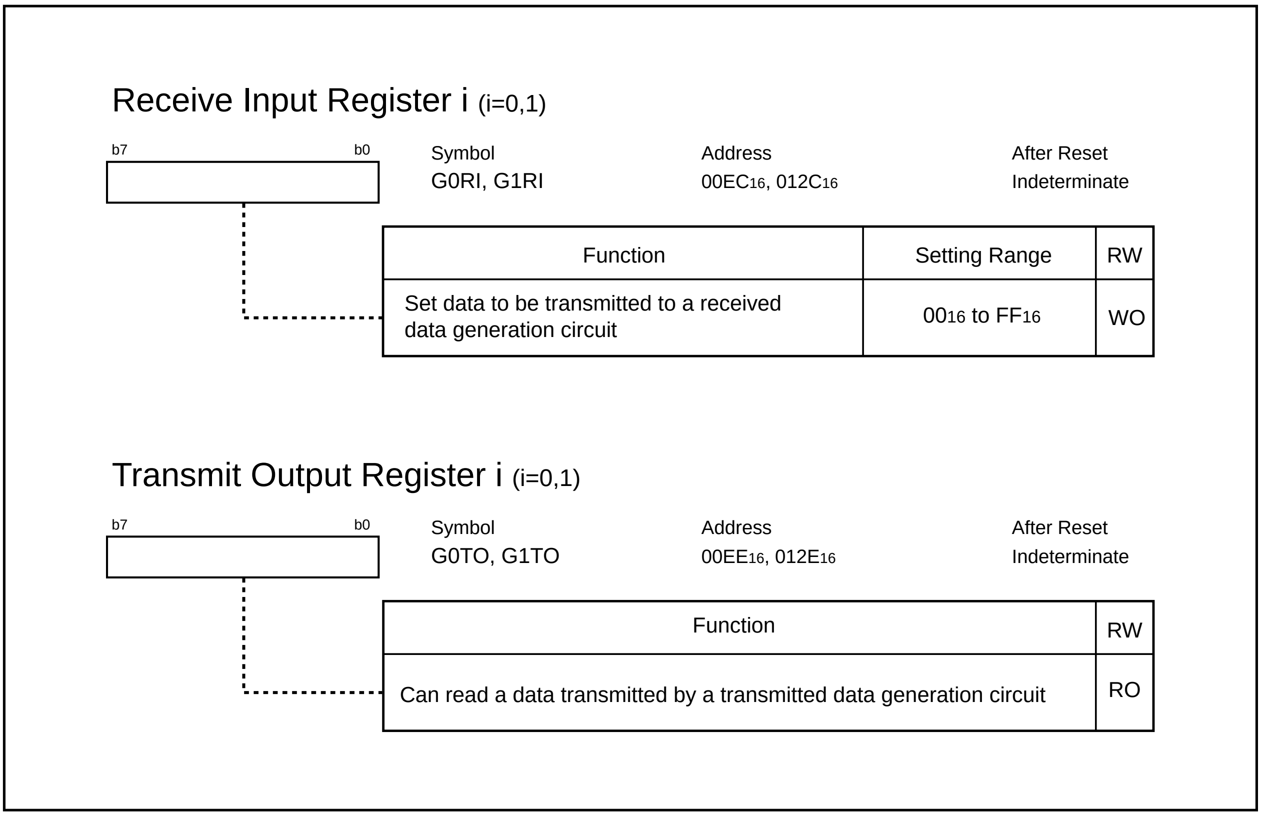
The above applies to the following conditions:
- The IVL bit in the G1POCRj register is set to "0" ("L" output as default value) and the INV bit is set to "0" (not inversed).
- The UD1 and UD0 bits are set to "00_2" (counter increment mode).
- m<n<p+2

Figure 22.18 SR Waveform Output Mode

22.4 Communication Unit 0 and 1 Communication Function

In the intelligent I/O communication unit 1, 8-bit clock synchronous serial I/O, 8-bit clock asynchronous serial I/O (UART) or HDLC data processing is available. In the communication unit 0, 8-bit clock synchronous serial I/O or HDLC data processing is available.

Figures 22.19 to 22.28 show registers associated with the communication function.



Receive Input Register i (i=0,1)

Symbol	Address	After Reset
G0RI, G1RI	00EC$_{16}$, 012C$_{16}$	Indeterminate

Function	Setting Range	RW
Set data to be transmitted to a received data generation circuit	00$_{16}$ to FF$_{16}$	WO

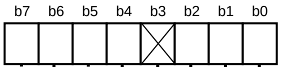
Transmit Output Register i (i=0,1)

Symbol	Address	After Reset
G0TO, G1TO	00EE$_{16}$, 012E$_{16}$	Indeterminate

Function	RW
Can read a data transmitted by a transmitted data generation circuit	RO

Figure 22.19 G0RI and G1RI Registers, G0TO and G1TO Registers

SI/O Communication Control Register i (i=0, 1)



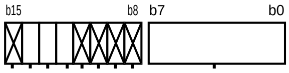
Symbol	Address	After Reset
G0CR, G1CR	00EF$_{16}$, 012F$_{16}$	0000 X011$_2$

Bit Symbol	Bit Name	Function	RW
TI	Transmit Buffer Empty Flag	0 : Data in the GiTB register 1 : No data in the GiTB register	RO
TXEPT	Transmit Register Empty Flag	0 : Data in the transmit register (during transmission) 1 : No data in the transmit register (transmit completed)	RO
RI	Receive Complete Flag	0 : No data in the GiRB register 1 : Data in the GiRB register	RO
(b3)	Nothing is assigned. When write, set to "0". When read, its contents is indeterminate.		–
TE	Transmit Enable Bit	0 : Transmit disable 1 : Transmit enable	RW
RE	Receive Enable Bit	0 : Receive disable 1 : Receive enable	RW
IPOL	ISRxD Input Polarity Switch Bit	0 : No inverse 1 : Inverse[1]	RW
OPOL	ISTxD Output Polarity Switch Bit	0 : No inverse 1 : Inverse[1]	RW

NOTES:

1. Set this bit to "1" when using UART mode.

SI/O Receive Buffer Register i (i=0, 1)



Symbol	Address	After Reset
G0RB, G1RB	00E9$_{16}$-00E8$_{16}$, 0129$_{16}$-0128$_{16}$	X000 XXXX XXXX XXXX$_2$

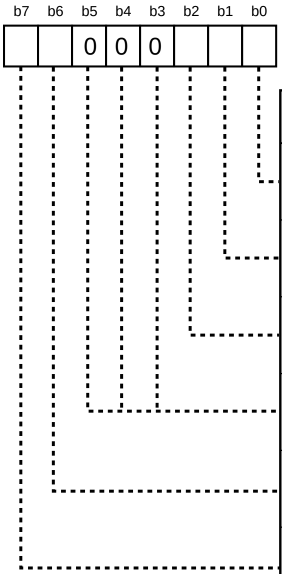
Bit Symbol	Bit Name	Function	RW
(b7 - b0)	————	Received data	RW
(b11 - b8)	Nothing is assigned. When read, its content is indeterminate.		–
OER	Overrun Error Flag	0 : No overrun error 1 : Overrun error found	RO
FER	Framing Error Flag[1]	0 : No framing error 1 : Framing error found	RO
PER	Parity Error Flag[1]	0 : No parity error 1 : Parity error found	RO
(b15)	Nothing is assigned. When read, its content is indeterminate.		–

NOTES:

1. Nothing is assigned in the FER and PER bits in the G0RB register. When read, its content is indeterminate.

Figure 22.20 G0CR and G1CR Registers, G0RB and G1RB Registers

SI/O Communication Mode Register 0



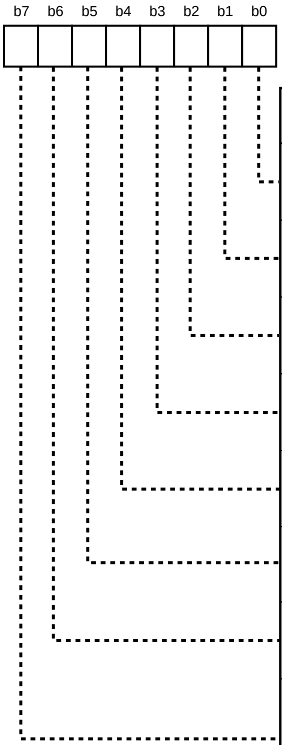
Symbol	Address	After Reset
G0MR	$00ED_{16}$	00_{16}

Bit Symbol	Bit Name	Function	RW
GMD0	Communication Mode Select Bit	b1 b0 0 1 : Clock synchronous serial I/O mode 1 1 : HDLC data processing mode[1]	RW
GMD1			RW
CKDIR	Internal/External Clock Select Bit	0 : Internal clock 1 : External clock	RW
—— (b5 - b3)	Reserved Bit	Set to "0"	RW
UFORM	Transfer Format Select Bit	0 : LSB first 1 : MSB first	RW
IRS	Transmit Interrupt Cause Select Bit	0 : No data in the G0TB register (TI=1) 1 : Transmission is completed (TXEPT=1)	RW

NOTES:

1. Do not set to any bit combinations except the above.

SI/O Communication Mode Register 1



Symbol	Address	After Reset
G1MR	$012D_{16}$	00_{16}

Bit Symbol	Bit Name	Function	RW
GMD0	Communication Mode Select Bit	b1 b0 0 0 : UART mode 0 1 : Clock synchronous serial I/O mode 1 0 : Special communication mode[1] 1 1 : HDLC data processing mode	RW
GMD1			RW
CKDIR	Internal/External Clock Select Bit	0 : Internal clock 1 : External clock	RW
STPS	Stop Bit Length Select Bit	0 : 1 stop bit 1 : 2 stop bits	RW
PRY	Parity Odd/Even Select Bit	0 : Odd parity 1 : Even parity	RW
PRYE	Parity Enable Select Bit	0 : Parity disabled 1 : Parity enabled	RW
UFORM	Transfer Format Select Bit	0 : LSB first 1 : MSB first	RW
IRS	Transmit Interrupt Cause Select Bit	0 : No data in the G1TB register (TI=1) 1 : Transmission is completed (TXEPT=1)	RW

NOTES:

1. In M32C/86, do not set the GMD1 and GMD0 bits to "10_2" except when using in motor vehicles.

Figure 22.22 G0MR and G1MR Registers

SI/O Expansion Mode Register 0[1]

b7 b6 b5 b4 b3 b2 b1 b0: b0 = 0

Symbol	Address	After Reset
G0EMR	$00FC_{16}$	00_{16}

Bit Symbol	Bit Name	Function	RW
(b0)	Reserved Bit	Set to "0"	RW
CRCV	CRC Default Value Select Bit	0 : Set to "0000_{16}" 1 : Set to "$FFFF_{16}$"	RW
ACRC	CRC Reset Select Bit	0 : Not reset 1 : Reset[2]	RW
BSINT	Bit Stuffing Error Interrupt Select Bit	0 : Not used 1 : Used	RW
RXSL	Receive Source Switch Bit	0 : ISRxD0 pin 1 : G0RI register	RW
TXSL	Transmit Source Switch Bit	0 : ISTxD0 pin 1 : G0TO register	RW
CRC0	CRC Generation Polynomial Select Bit	b7 b6 0 0 : X^8+X^4+X+1 0 1 : Do not set to this value 1 0 : $X^{16}+X^{15}+X^2+1$ 1 1 : $X^{16}+X^{12}+X^5+1$	RW
CRC1			RW

NOTES:
1. The G0EMR register is used in HDLC data processing mode. It must be in a reset state or set to "00_{16}" in clock synchronous serial I/O mode.
2. CRC is reset when data in the G0CMP3 register matches received data.

SI/O Expansion Mode Register 1[1]

b7 b6 b5 b4 b3 b2 b1 b0

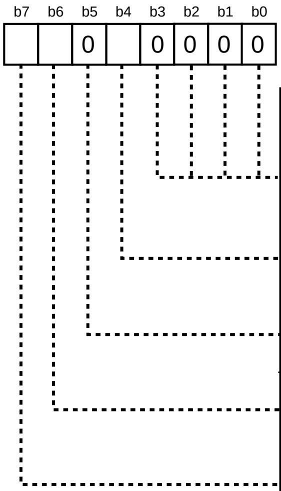
Symbol	Address	After Reset
G1EMR	$013C_{16}$	00_{16}

Bit Symbol	Bit Name	Function	RW
SMODE	Synchronous Mode Select Bit	0 : Re-synchronous mode not used 1 : Re-synchronous mode	RW
CRCV	CRC Default Value Select Bit	0 : Set to "0000_{16}" 1 : Set to "$FFFF_{16}$"	RW
ACRC	CRC Reset Select Bit	0 : Not reset 1 : Reset[2]	RW
BSINT	Bit Stuffing Error Interrupt Select Bit	0 : Not used 1 : Used	RW
RXSL	Receive Source Switch Bit	0 : ISRxD1 pin 1 : G1RI register	RW
TXSL	Transmit Source Switch Bit	0 : ISTxD1 pin 1 : G1TO register	RW
CRC0	CRC Generation Polynomial Select bit	b7 b6 0 0 : X^8+X^4+X+1 0 1 : Do not set to this value 1 0 : $X^{16}+X^{15}+X^2+1$ 1 1 : $X^{16}+X^{12}+X^5+1$	RW
CRC1			RW

NOTES:
1. The G1EMR register is used in special communication mode or HDLC data processing mode. It must be in a reset state or be set to "00_{16}" in clock synchronous serial I/O mode or UART mode.
2. CRC is reset when data in the G1CMP3 register matches received data.

Figure 22.23 G0EMR and G1EMR Registers

SI/O Expansion Transmit Control Register 0[1]



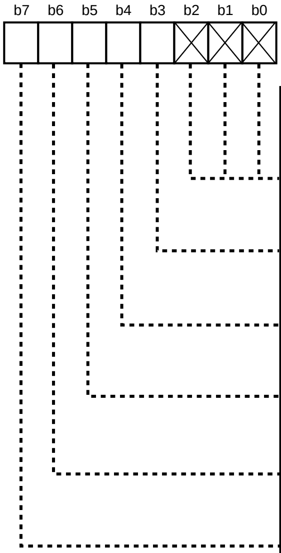
Symbol G0ETC   Address $00FF_{16}$ After Reset $0000\ 0XXX_2$

Bit Symbol	Bit Name	Function	RW
—— (b3 - b0)	Reserved Bit	Set to "0"	—
TCRCE	Transmit CRC Enable Bit	0 : Not used 1 : Used	RW
—— (b5)	Reserved Bit	Set to "0"	RW
TBSF0	Transmit Bit Stuffing "1" Insert Select Bit	0 : "1" is not inserted 1 : "1" is inserted	RW
TBSF1	Transmit Bit Stuffing "0" Insert Select Bit	0 : "0" is not inserted 1 : "0" is inserted	RW

NOTES:

1. The G0ETC register is used in HDLC data processing mode. It must be in a reset state or set to "00_{16}" in clock synchronous serial I/O mode.

SI/O Expansion Transmit Control Register 1[1]



Symbol G1ETC   Address $013F_{16}$ After Reset $0000\ 0XXX_2$

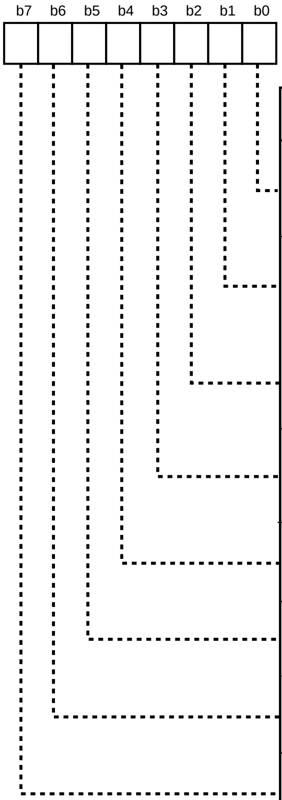
Bit Symbol	Bit Name	Function	RW
—— (b2 - b0)	Reserved Bit	When read, its content is indeterminate	RO
SOF	SOF Transmit Request Bit	0 : No request to transmit SOF 1 : Request to transmit SOF	RW
TCRCE	Transmit CRC Enable Bit	0 : Not used 1 : Used	RW
ABTE	Arbitration Enable Bit	0 : Not used 1 : Used	RW
TBSF0	Transmit Bit Stuffing "1" Insert Select Bit	0 : "1" is not inserted 1 : "1" is inserted	RW
TBSF1	Transmit Bit Stuffing "0" Insert Select Bit	0 : "0" is not inserted 1 : "0" is inserted	RW

NOTES:

1. The G1ETC register is used in special communication mode or HDLC data processing mode. It must be in a reset state or set to "00_{16}" in clock synchronous serial I/O mode or UART mode.

Figure 22.24 G0ETC and G1ETC Registers

SI/O Expansion Receive Control Register i (i=0,1)[1]



Symbol	Address	After Reset
G0ERC, G1ERC	$00FD_{16}$, $013D_{16}$	00_{16}

Bit Symbol	Bit Name	Function	RW
CMP0E	Data Compare Function 0 Select Bit	0 : The GiDR register (receive data register) is not compared with the GiCMP0 register 1 : The GiDR register is compared with the GiCMP0 register	RW
CMP1E	Data Compare Function 1 Select Bit	0 : The GiDR register (receive data register) is not compared with the GiCMP1 register 1 : The GiDR register is compared with the GiCMP1 register	RW
CMP2E	Data Compare Function 2 Select Bit	0 : The GiDR register (receive data register) is not compared with the GiCMP2 register 1 : The GiDR register is compared with the GiCMP2 register	RW
CMP3E	Data Compare Function 3 Select Bit	0 : The GiDR register (receive data register) is not compared with the GiCMP3 register 1 : The GiDR register is compared with the GiCMP3 register[2]	RW
RCRCE	Receive CRC Enable Bit	0 : Not used 1 : Used	RW
RSHTE	Receive Shift Operation Enable Bit	0 : Receive shift operation disabled 1 : Receive shift operation enabled	RW
RBSF0	Receive Bit Stuffing "1" Delete Select Bit	0 : "1" is not deleted 1 : "1" is deleted	RW
RBSF1	Receive Bit Stuffing "0" Delete Select Bit	0 : "0" is not deleted 1 : "0" is deleted	RW

NOTES:

1. The GiERC register is used in special communication mode or HDLC data processing mode.
   It must be set to "0010 0000_2" in clock synchronous serial I/O mode.
   It must be in a reset state or be set to "00_{16}" in UART mode.
2. When the ACRC bit in the GiEMR register is set to "1" (CRC reset function used), set the CMP3E bit to "1".

Figure 22.25 G0ERC and G1ERC Registers

SI/O Special Communication Interrupt Detect Register 0 (1, 2)

b7	b6	b5	b4	b3	b2	b1	b0
				0		0	0

Symbol	Address	After Reset
G0IRF	$00FE_{16}$	00_{16}

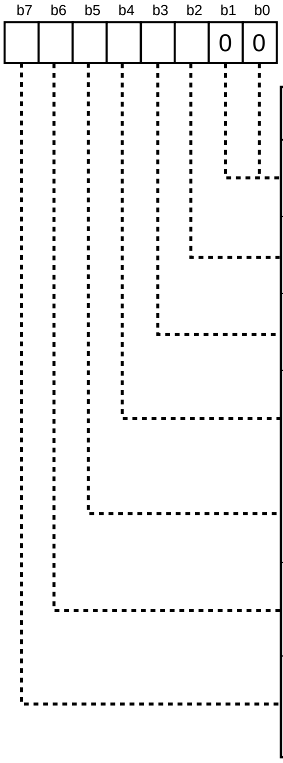
Bit Symbol	Bit Name	Function	RW
(b1 - b0)	Reserved Bit	Set to "0"	RW
BSERR	Bit Stuffing Error Detect Flag	0 : Not detected 1 : Detected	RW
(b3)	Reserved Bit	Set to "0"	RW
IRF0	Interrupt Cause Determination Flag 0	0 : The G0DR register (receive data register) does not match the G0CMP0 register 1 : The G0DR register matches the G0CMP0 register	RW
IRF1	Interrupt Cause Determination Flag 1	0 : The G0DR register (receive data register) does not match the G0CMP1 register 1 : The G0DR register matches the G0CMP1 register	RW
IRF2	Interrupt Cause Determination Flag 2	0 : The G0DR register (receive data register) does not match the G0CMP2 register 1 : The G0DR register matches the G0CMP2 register	RW
IRF3	Interrupt Cause Determination Flag 3	0 : The G0DR register (receive data register) does not match the G0CMP3 register 1 : The G0DR register matches the G0CMP3 register	RW

NOTES:

1. The G0IRF register is used in HDLC data processing mode. Do not use in clock synchronous serial I/O mode.
2. The SRT0R bit in the IIO4IR register is set to "1" if the BSERR or IRF0 to IRF3 bit is set to "1".

Figure 22.26 G0IRF Register

SI/O Special Communication Interrupt Detect Register 1[1,2]



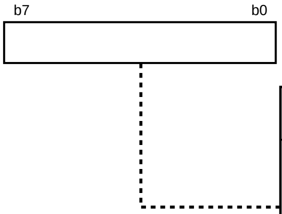
Symbol	Address	After Reset
G1IRF	$013E_{16}$	00_{16}

Bit Symbol	Bit Name	Function	RW
—— (b1 - b0)	Reserved Bit	Set to "0"	RW
BSERR	Bit Stuffing Error Detect Flag	0 : Not detected 1 : Detected	RW
ABT	Arbitration Lost Detect Flag	0 : Not detected 1 : Detected	RW
IRF0	Interrupt Cause Determination Flag 0	0 : The G1DR register (receive data register) does not match the G1CMP0 register 1 : The G1DR register (receive data register) matches the G1CMP0 register	RW
IRF1	Interrupt Cause Determination Flag 1	0 : The G1DR register (receive data register) does not match the G1CMP1 register 1 : The G1DR register (receive data register) matches the G1CMP1 register	RW
IRF2	Interrupt Cause Determination Flag 2	0 : The G1DR register (receive data register) does not match the G1CMP2 register 1 : The G1DR register (receive data register) matches the G1CMP2 register	RW
IRF3	Interrupt Cause Determination Flag 3	0 : The G1DR register (receive data register) does not match the G1CMP3 register 1 : The G1DR register (receive data register) matches the G1CMP3 register	RW

NOTES:

1. The G1IRF register is used in special communication mode or HDLC data processing mode. It must be in a reset state or set to "00_{16}" in clock synchronous serial I/O mode or UART mode.
2. The SRT1R bit in the IIO4IR register is also set to "1" if the BSERR, ABT or IRF0 to IRF3 bit is set to "1".

Transmit Buffer (Receive Data) Register (i=0,1)



Symbol	Address	After Reset
G0TB, G0DR	$00EA_{16}$	Indeterminate
G1TB, G1DR	$012A_{16}$	Indeterminate

Function	RW
Set data to be transmitted. In HDLC data processing mode, the receive data register is read by reading the GiTB register. Value is written to the transmit buffer register by writing it to the GiTB register. In HDLC data processing mode, the value set in the GiRI register is transferred to the GiDR register.	RW

Figure 22.27 G1IRF Register, G0TB and G1TB / G0DR and G1DR Registers

Data Compare Register ij (i=0,1, j=0 to 3)

b7 b0

Symbol	Address	After Reset
G0CMP0 to G0CMP3	00F0$_{16}$, 00F1$_{16}$, 00F2$_{16}$, 00F3$_{16}$	Indeterminate
G1CMP0 to G1CMP3	0130$_{16}$, 0131$_{16}$, 0132$_{16}$, 0133$_{16}$	Indeterminate

Function	Setting Range	RW
Data to be compared	00$_{16}$ to FF$_{16}$	RW

NOTES:

1. Set the GiMSK0 register to use the GiCMP0 register.
   Set the GiMSK1 register to use the GiCMP1 register.

Data Mask Register ij (i=0,1, j=0,1)

b7 b0

Symbol	Address	After Reset
G0MSK0, G0MSK1	00F4$_{16}$, 00F5$_{16}$	Indeterminate
G1MSK0, G1MSK1	0134$_{16}$, 0135$_{16}$	Indeterminate

Function	Setting Range	RW
Masked data for received data Set incomparable bit to "1"	00$_{16}$ to FF$_{16}$	RW

Transmit CRC Code Register i (i=0,1)

b15 b8 b7 b0

Symbol	Address	After Reset
G0TCRC, G1TCRC	00FB$_{16}$-00FA$_{16}$, 013B$_{16}$-013A$_{16}$	0000$_{16}$

Function	RW
Result of the transmit CRC calculation(1, 2)	RO

NOTES:

1. The calculated result is reset by setting the TE bit in the GiCR register to "0" (transmit disabled). The CRCV bit in the GiEMR register selects a default value.
2. Transmit CRC calculation is performed with each bit of data transmitted while the TCRCE bit in the GiETC register is set to "1" (used).

Receive CRC Code Register i (i=0,1)

b15 b8 b7 b0

Symbol	Address	After Reset
G0RCRC, G1RCRC	00F9$_{16}$-00F8$_{16}$, 0139$_{16}$-0138$_{16}$	Indeterminate

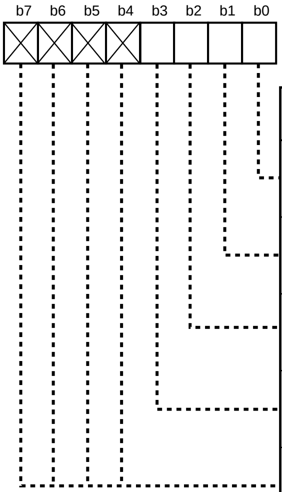
Function	RW
Result of the receive CRC calculation(1, 2, 3)	RO

NOTES:

1. The calculated result is reset by setting the RCRCE bit in the GiERC register to "0" (not used). If the ACRC bit in the GiEMR register is set to "1" (reset), the result is reset by matching data in the GiCMPj register (j=0 to 3) with the received data.
2. The result is reset to the default value selected by the CRCV bit in the GiEMR register before reception starts.
3. Receive CRC calculation is performed with every bit of data received while the RCRCE bit in the GiERC register is set to "1" (used).

Figure 22.28 G0CMP0 to G0CMP3 Registers and G1CMP0 to G1CMP3 Registers G0MSK0 and G0MSK1 Registers, G1MSK0 and G1MSK1 Registers G0TCRC and G1TCRC Registers, G0RCRC and G1RCRC Registers

Communication Clock Select Register



Symbol	Address	After Reset
CCS	$00F6_{16}$	XXXX 0000_2

Bit Symbol	Bit Name	Function	RW
CCS0	Communication Unit 0 Clock Select Bit	b1b0 0 0 : Do not set to this value 0 1 : f_1[(1)] 1 0 : f_{2n} 1 1 : f_8	RW
CCS1			RW
CCS2	Communication Unit 1 Clock Select Bit	b3b2 0 0 : Clock output from the channel i (i=1 to 3) 0 1 : f_1[(1)] 1 0 : f_{2n} 1 1 : f_8	RW
CCS3			RW
(b7 - b4)	Nothing is assigned. When write, set to "0". When read, its contents is indeterminate.		–

NOTES:

1. This setting is enabled in HDLC data processing mode. Do not set the CCS1 and CC0 bits or CCS3 and CCS2 bits to "01_2" in other modes.

Figure 22.29 CCS Register

22.4.1 Clock Synchronous Serial I/O Mode (Communication Units 0 and 1)

In clock synchronous serial I/O mode, data is transmitted and received with the transfer clock. f_8 or f_{2n} can be selected as the communication unit 0 transfer clock. f_8, f_{2n} or the clock generated in channels 0 and 3 can be selected as the communication unit 1 transfer clock.

Table 22.12 lists specifications of clock synchronous serial I/O mode for the communication units 0 and 1. Tables 22.13 and 22.14 list clock settings. Table 22.15 lists register settings. Tables 22.16 to 22.19 list pin settings. Figure 22.29 shows an example of transmit and receive operation.

Table 22.12 Clock Synchronous Serial I/O Mode Specifications (Communication Units 0 and 1)

Item	Specification
Transfer Data Format	Transfer data : 8 bits long
Transfer Clock[1]	See Tables 22.13 and 22.14
Transmit Start Condition	Set registers associated with the waveform generating function, the GiMR and GiERC registers (i=0,1). Then, set as is written below after at least one transfer clock cycle. • Set the TE bit in the GiCR register to "1" (transmit enable) • Set the TI bit in the GiCR register to "0" (data in the GiTB register)
Receive Start Condition	Set registers associated with the waveform generating function, the GiMR and GiERC registers. Then, set as is written below after at least one transfer clock cycle. • Set the RE bit in the GiCR register to "1" (receive enable) • Set the TE bit to "1" (transmit enable) • Set the TI bit to "0" (data in the GiTB register)
Interrupt Request	• While transmitting, one of the following conditions can be selected to set the SIOiTR bit to "1" (interrupt requested) (see Figure 11.14) : – The IRS bit in the GiMR register is set to "0" (no data in the GiTB register) and data is transferred to the transmit register from the GiTB register – The IRS bit is set to "1" (transmission completed) and data transfer from the transmit register is completed • While receiving, the following condition can be selected to set SIOiRR bit is set to "1" (data reception is completed): Data is transferred from the receive register to the GiRB register
Error Detection	Overrun error[2] This error occurs, when the next data reception is started and the 8th bit of the next data is received before reading the GiRB register
Selectable Function	• LSB first or MSB first Select either bit 0 or bit 7 to transmit or receive data • ISTxDi and ISRxDi I/O polarity inverse ISTxDi pin output level and ISRxDi pin input level are inversed

NOTES:
1. In clock synchronous serial I/O mode, set the RSHTE bit in the GiERC register (i=0, 1) to "1" (receive shift operation enabled).
2. When an overrun error occurs, the GiRB register is indeterminate.

When the OPOL bit in the GiCR register is set to "0" (ISTxD output polarity not inversed), the ISTxDi pin puts in a high-level ("H") signal output after selecting operating mode until transfer starts. When the OPOL bit is set to "1" (ISTxD output polarity inversed), the ISTxDi pin puts in a low-level ("L") signal output.

Table 22.13 Clock Settings (Communication Unit 0)

Transfer Clock	G0MR Register	CCS Register	
	CKDIR Bit	CCS0 Bit	CCS1 Bit
f_8	0	1	1
f_{2n}[1]	0	0	1
Input from ISCLK0	1	-	-

NOTES:
1. The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2n (n=1 to 15).

Table 22.14 Clock Settings (Communication Unit 1)

Transfer Clock(3)	G1MR Register	CCS Register	
	CKDIR Bit	CCS2 Bit	CCS3 Bit
$\frac{f_{BT1}}{2(n+2)}$(1)	0	0	0
f_8	0	1	1
f_{2n}(2)	0	0	1
Input from ISCLK1	1	-	-

n: Setting value of the G1PO0 register, 0001_{16} to $FFFD_{16}$

NOTES:

1. The transfer clock is generated in phase-delayed waveform output mode of the channel 3 waveform generating function.
2. The CNT3 to CNT0 bits in the TCSPR register select no division (*n*=0) or divide-by-2*n* (*n*=1 to 15).
3. The transfer clock must be f_{BT1} divided by six or more.

Table 22.15 Register Settings in Clock Synchronous Serial I/O Mode (Communication Units 0 and 1)

Register	Bit	Function	
		Communication Unit 1	Communication Unit 0
CCS	CCS1, CCS0	Setting not required when using the communication unit 1 only	Select transfer clock
	CCS3, CSS2	Select transfer clock	Setting not required when using the communication unit 0 only
G1BCR0(2)	BCK1, BCK0	Set to "11$_2$" (f1)	
	DIV4 to DIV0	Select divide ratio of count source	
	IT	Set to "0"	
G1BCR1(2)	7 to 0	Set to "0001 0010$_2$"	
G1POCR0(2)	7 to 0	Set to "0000 0111$_2$"	
G1POCR1(2)	7 to 0	Set to "0000 0111$_2$"	
G1POCR3(2)	MOD2 to MOD0	Set to "010$_2$"(1)	
	IVL	Select default ISCLKi output value(1)	
	RLD	Set to "0"	
	INV	Select whether ISCLKi puts in an inversed signal or not(1)	
G1PO0(2)	15 to 0	Set bit rate $\frac{f_{BT1}}{2 \times (\text{setting value} + 2)}$ = transfer clock frequency	
G1PO3(2)	15 to 0	Set to a value smaller than the G1PO0 register(1)	
G1FS(2)	FSC3,FSC1,FSC0	Set to "0"(1)	
G1FE(2)	IFE3,IFE1,IFE0	Set to "1"(1)	
GiERC	7 to 0	Set to "0010 0000$_2$"	
GiMR	GMD1, GMD0	Set to "01$_2$"	
	CKDIR	Select the internal clock or external clock	
	STPS	Set to "0"	
	UFORM	Select either LSB first or MSB first	
	IRS	Select what cause the transmit interrupt to be generated	
GiCR	TI	Transmit buffer empty flag	
	TXEPT	Transmit register empty flag	
	RI	Receive complete flag	
	TE	Set to "1" to enable transmission and reception	
	RE	Set to "1" to enable reception	
	IPOL	Select ISRxDi input polarity (usually set to "0")	
	OPOL	Select ISTxDi output polarity (usually set to "0")	
GiTB	–	Write data to be transmitted	
GiRB	–	Received data and error flag are stored	

i = 0 to 1

NOTES:

1. The CKDIR bit in the GiMR register is set to "0" (internal clock).
2. These registers must be set, when f_8 or f_{2n} is selected as transfer clock source notwithstanding.

Table 22.16 Pin Settings in Clock Synchronous Serial I/O Mode (Communication Units 0 and 1)(1)

Port Name	Function	Setting						Register (1)
		PS1 Register	PSL1 Register	PSC Register	PSD1 Register	PD7 Register	IPS Register	
$P7_3$	ISTxD1 Output	PS1_3=1	PSL1_3=0	PSC_3=1	-	-	-	G1POCR0
$P7_4$	ISCLK1 Input	PS1_4=0	-	-	-	PD7_4=0	IPS1=0	-
	ISCLK1 Output	PS1_4=1	PSL1_4=0	PSC_4=1	-	-	-	G1POCR1
$P7_5$	ISRxD1 Input	PS1_5=0	-	-	-	PD7_5=0	IPS1=0	-
$p7_6$	ISTxD0 Output	PS1_6=1	PSL1_6=0	PSC_6=0	PSD1_6=0	-	-	-
$p7_7$	ISCLK0 Input	PS1_7=0	-	-	-	PD7_7=0	IPS0=0	-
	ISCLK0 Output	PS1_7=1	PSL1_7=0	-	-	-	-	-

NOTES:

1. Set the MOD2 to MOD0 bits in the corresponding register to "111_2" (output from the communication function used).

Table 22.17 Pin Settings (2)

Port Name	Function	Setting		
		PS2 Register	PD8 Register	IPS Register
$P8_0$	ISRxD0 input	PS2_0 = 0	PD8_0 = 0	IPS0 = 0

Table 22.18 Pin Settings (3)

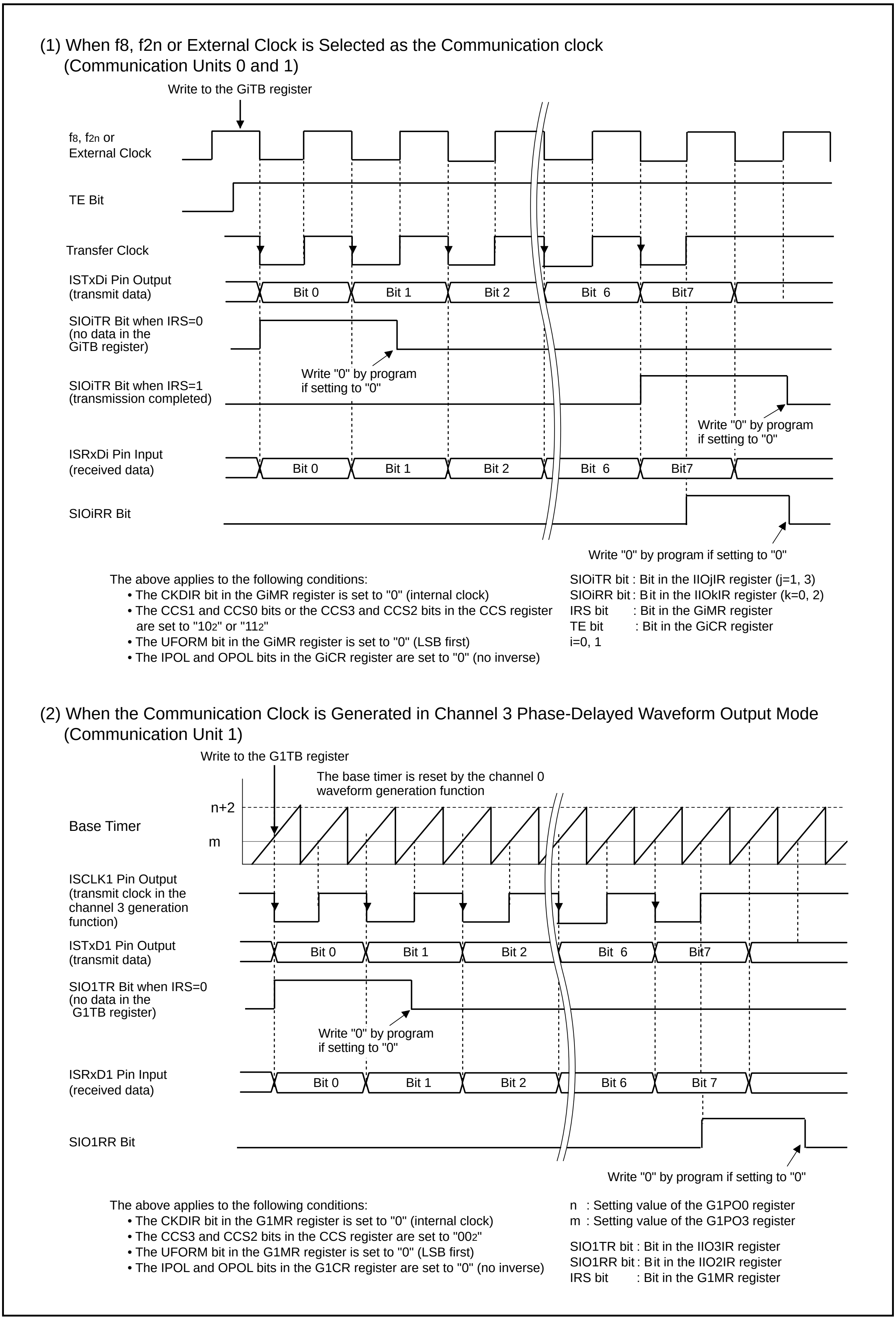
Port Name	Function	Setting			Register(1)
		PS5 Register	PD11 Register	IPS Register	
$P11_0$	ISTxD1 output	PS5_0 = 1	-	-	G1POCR0
$P11_1$	ISCLK1 input	PS5_1 = 0	PD11_1 = 0	IPS1 = 1	-
	ISCLK1 output	PS5_1 = 1	-	-	G1POCR1
$P11_2$	ISRxD1 input	PS5_2 = 0	PD11_2 = 0	IPS1 = 1	-

NOTES:

1. Set the MOD2 to MOD0 bits in the corresponding register to "111_2" (output from communication function used).

Table 22.19 Pin Settings (4)

Port Name	Function	Setting		
		PS9 Register	PD15 Register	IPS Register
$P15_0$	ISTxD0 output	PS9_0 = 1	-	-
$P15_1$	ISCLK0 input	PS9_1 = 0	PD15_2 = 0	IPS0 = 1
	ISCLK0 output	PS9_1 = 1	-	-
$P15_2$	ISRxD0 input	-	PD15_2 = 0	IPS0 = 1



(1) When f8, f2n or External Clock is Selected as the Communication clock (Communication Units 0 and 1)

Write to the GiTB register
f8, f2n or External Clock
TE Bit
Transfer Clock
ISTxDi Pin Output (transmit data)
Bit 0
Bit 1
Bit 2
Bit 6
Bit7
SIOiTR Bit when IRS=0 (no data in the GiTB register)
Write "0" by program if setting to "0"
SIOiTR Bit when IRS=1 (transmission completed)
Write "0" by program if setting to "0"
ISRxDi Pin Input (received data)
Bit 0
Bit 1
Bit 2
Bit 6
Bit7
SIOiRR Bit
Write "0" by program if setting to "0"

The above applies to the following conditions:

- The CKDIR bit in the GiMR register is set to "0" (internal clock)
- The CCS1 and CCS0 bits or the CCS3 and CCS2 bits in the CCS register are set to "10$_2$" or "11$_2$"
- The UFORM bit in the GiMR register is set to "0" (LSB first)
- The IPOL and OPOL bits in the GiCR register are set to "0" (no inverse)

SIOiTR bit : Bit in the IIOjIR register (j=1, 3)
SIOiRR bit : Bit in the IIOkIR register (k=0, 2)
IRS bit : Bit in the GiMR register
TE bit : Bit in the GiCR register
i=0, 1

(2) When the Communication Clock is Generated in Channel 3 Phase-Delayed Waveform Output Mode (Communication Unit 1)

Write to the G1TB register
The base timer is reset by the channel 0 waveform generation function
Base Timer
n+2
m
ISCLK1 Pin Output (transmit clock in the channel 3 generation function)
ISTxD1 Pin Output (transmit data)
Bit 0
Bit 1
Bit 2
Bit 6
Bit7
SIO1TR Bit when IRS=0 (no data in the G1TB register)
Write "0" by program if setting to "0"
ISRxD1 Pin Input (received data)
Bit 0
Bit 1
Bit 2
Bit 6
Bit 7
SIO1RR Bit
Write "0" by program if setting to "0"

The above applies to the following conditions:

- The CKDIR bit in the G1MR register is set to "0" (internal clock)
- The CCS3 and CCS2 bits in the CCS register are set to "00$_2$"
- The UFORM bit in the G1MR register is set to "0" (LSB first)
- The IPOL and OPOL bits in the G1CR register are set to "0" (no inverse)

n : Setting value of the G1PO0 register
m : Setting value of the G1PO3 register

SIO1TR bit : Bit in the IIO3IR register
SIO1RR bit : Bit in the IIO2IR register
IRS bit : Bit in the G1MR register

Figure 22.30 Transmit and Receive Operation

22.4.2 Clock Asynchronous Serial I/O (UART) Mode (Communication Unit 1)

In clock asynchronous serial I/O (UART) mode, data is transmitted at a desired bit rate and in a desired transfer data format. Table 22.20 lists specifications of UART mode in the communication unit 1. Table 22.21 lists clock settings. Table 22.22 lists register settings. Tables 22.23 and 22.24 list pin settings. Figure 22.30 shows an example of transmit operation. Figure 22.31 shows an example of receive operation.

Table 22.20 UART Mode Specifications (Communication Unit 1)

Item	Specification
Transfer Data Format	• Character bit (transfer data) : 8 bits long • Start bit : 1 bit long • Parity bit: selected from odd, even, or none • Stop bit : selected length from 1 bit or 2 bits
Transfer Clock[1]	See Table 22.21
Transmit Start Condition	Set registers associated with the waveform generating function, the G1MR and G1ERC registers. Then, set as is written below after at least one transfer clock cycle: • Set the TE bit in the G1CR register to "1" (transmit enable) • Set the TI bit in the G1CR register to "0" (data written to the G1TB register)
Receive Start Condition	Set registers associated with the waveform generating function, the G1MR and G1ERC registers. Then, set as is written below after at least one transfer clock cycle: • Set the RE bit in the G1CR register to "1" (receive enable) • Detect the start bit
Interrupt Request	• While transmitting, one of the following conditions can be selected to set the SIO1TR bit to "1" (interrupt requested) (See Figure 11.14.) : – The IRS bit in the G1MR register is set to "0" (no data in the G1TB register) and data is transferred to the transmit register from the G1TB register. – The IRS bit is set to "1" (transmission completed) and data transfer from the transmit register is completed • While receiving, the following condition can be selected to set the SIO1RR bit is set to "1": Data is transferred from the receive register to the G1RB register (data reception is completed)
Error Detection	• Overrun error[2] This error occurs, when the next data reception is started and the final stop bit of the next data is received before reading the G1RB register • Parity error While parity is enabled, this error occurs when the number of "1" in parity and character bits does not match the number of "1" set • Framing error This error occurs when the number of the stop bits set is not detected
Selectable Function	• Stop bit length The length of the stop bit is selected from 1 bit or 2 bits • LSB first or MSB first Select either bit 0 or bit 7 to transmit or receive data

NOTES:

1. The transfer clock must be f_{BT1} divided by six or more.
2. When an overrun error occurs, the G1RB register is indeterminate.

Table 22.21 Clock Settings (Communication Unit 1)

Transfer Clock[3]	G1MR Register	CCS Register	
	CKDIR Bit	CCS2 Bit	CCS3 Bit
$\frac{f_{BT1}}{2(n+2)}$ [1, 2]	0	0	0

n: Setting value of the G1PO0 register 0001_{16} to $FFFD_{16}$

NOTES:

1. Transmit clock is generated in phase-delayed waveform output mode of the channel 3 waveform generating function.
2. Received clock is generated when phase-delayed waveform mode of the channel 2 waveform generating function and the channel 2 time measurement function is simultaneously performed.
3. The transfer clock must be f_{BT1} divided by six or more.

Table 22.22 Register Settings in UART Mode (Communication Unit 1)

Register	Bit	Function
G1BCR0	BCK1, BCK0	Set to "11_2" (f_1)
	DIV4 to DIV0	Select divide ratio of count source
	IT	Set to "0"
G1BCR1	7 to 0	Set to "0001 0010_2"
G1POCR0	7 to 0	Set to "0000 0111_2"
G1POCR2	7 to 0	Set to "0000 0110_2"
G1POCR3	7 to 0	Set to "0000 0010_2"
G1TMCR2	7 to 0	Set to "0000 0010_2"
G1PO0	15 to 0	Set bit rate $\frac{f_{BT1}}{2 \text{ x (setting value + 2)}}$ = transfer clock frequency
G1PO3	15 to 0	Set to a value smaller than the G1PO0 register
G1FS	FSC3 to FSC0	Set to "0100_2"
G1FE	IFE3 to IFE0	Set to "1101_2"
G1MR	GMD1, GMD0	Set to "00_2"
	CKDIR	Set to "0"
	STPS	Select stop bit length
	PRY, PRYE	Select either parity enabled or disabled and either odd parity or even parity
	UFORM	Select either the LSB first or MSB first
	IRS	Select what causes the receive interrupt to be generated
G1CR	TI	Transmit buffer empty flag
	TXEPT	Transmit register empty flag
	RI	Receive complete flag
	TE	Set to "1" to enable transmission and reception
	RE	Set to "1" to enable reception
	IPOL	Set to "1"
	OPOL	Set to "1"
G1TB	7 to 0	Write data to be transmitted
G1RB	15 to 0	Received data and error flag are stored
CCS	CCS3, CCS2	Set to "00_2"

Table 22.23 Pin Settings in UART Mode

Port Name	Function	Setting					Register[1]
		PS1 Register	PSL1 Register	PSC Register	PD7 Register	IPS Register	
$P7_3$	ISTxD1 output	PS1_3 = 1	PSL1_3 = 0	PSC_3 = 1	-	-	G1POCR0
$P7_5$	ISRxD1 input	PS1_5 = 0	-	-	PD7_5 = 0	IPS1 = 0	-

NOTES:

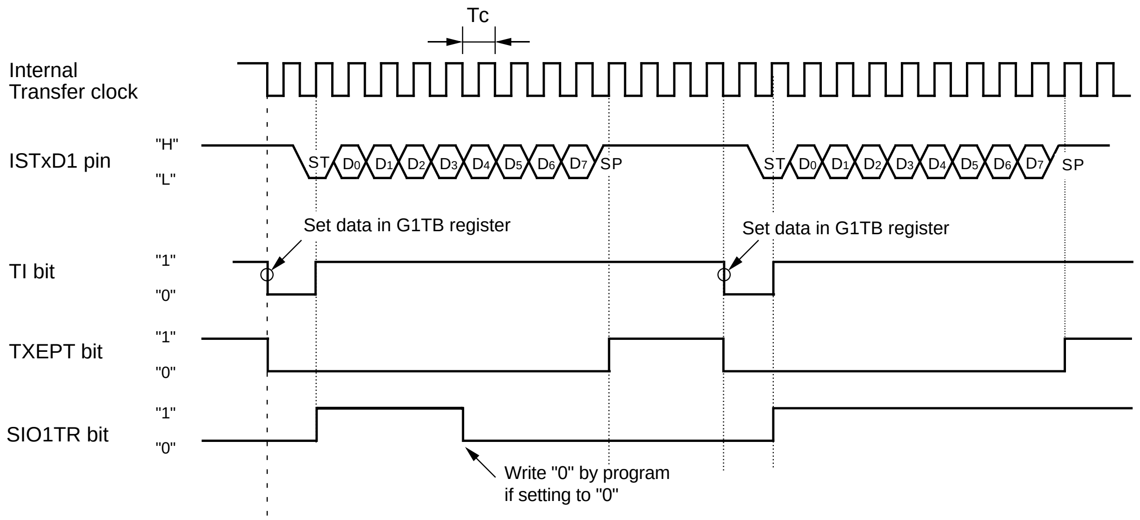
1. Set the MOD2 to MOD0 bits in the corresponding register to "111_2" (output from communication function used).

Table 22.24 Pin Settings (Continued)

Port Name	Function	Setting			Register[1]
		PS5 Register	PD11 Register	IPS Register	
P11$_0$	ISTxD1 output	PS5_0 = 1	-	-	G1POCR0
P11$_2$	ISRxD1 input	PS5_2 = 0	PD11_2 = 0	IPS1 = 1	-

NOTES:

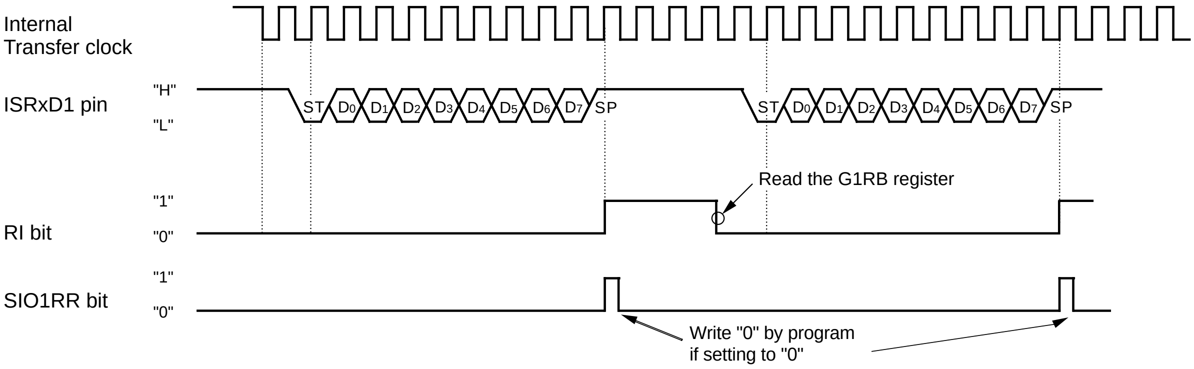
1. Set the MOD2 to MOD0 bits in the corresponding register to "111$_2$" (output from the communication function used).



The above applies to the following conditions:
- The STPS bit in the G1MR register is set to "0" (1 stop bit)
- The PRYE bit in the G1MR register is set to "0" (parity disabled)
- The UFORM bit in the G1MR register is set to "0" (LSB first)
- The INV bits in the G1POCR0 to G1POCR7 registers are set to "0" (no inverse)
- The IRS bit in the G1MR register is set to "0" (no data in the G1TB register)

TI, TXEPT bit : Bits in the G1CR register
SIO1TR bit : Bit in the IIO3IR register

Figure 22.31 Transmit Operation



The above applies to the following conditions:
- The STPS bit in the G1MR register is set to "0" (1 stop bit)
- The PRYE bit in the G1MR register is set to "0" (parity disabled)
- The UFORM bit in the G1MR register is set to "0" (LSB first)
- The INV bits in the G1POCR0 to G1POCR7 registers are set to "0" (no inverse)

SIO1RR bit : Bit in the IIO2IR register
RI bit : Bit in the G1CR register

Figure 22.32 Receive Operation

22.4.3 HDLC Data Processing Mode (Communication Units 0 and 1)

In HDLC data processing mode, bit stuffing, flag detection, abort detection and CRC calculation are available for HDLC control. f_1, f_8 or f_{2n} can be selected as the communication unit 0 transfer clock. f_1, f_8, f_{2n} or clock, generated in the channel 0 or 1, can be selected as the communication unit 1 transfer clock. No pin is used. To convert data, data to be transmitted is written to the GiTB register (i=0,1) and the data conversion result is restored after data conversion. If any data are in the GiTO register after data conversion, the conversion is terminated. If no data is in the GiTO register, bit stuffing processing is executed regardless of no data available in the transmit output buffer. A CRC value is calculated every time one bit is converted. If no data is in the GiRI register, received data conversion is terminated.

Table 22.25 list specifications of the HDLC data processing mode. Tables 22.26 and 22.27 list clock settings. Table 22.28 lists register settings.

Table 22.25 HDLC Processing Mode Specifications (Communication Units 0 and 1)

Item	Specification
Input Data Format	8-bit data fixed, bit alignment is optional
Output Data Format	8-bit data fixed
Transfer Clock	See Tables 22.26 and 22.27
I/O Method	• During transmit data processing, value set in the GiTB register is converted in HDLC data processing mode and transferred to the GiTO register. • During received data processing, value set in the GiRI register is converted in HDLC data processing mode and transferred to the GiRB register. The value in the GiRI register is also transferred to the GiTB register (received data register).
Bit Stuffing	During transmit data processing, "0" following five continuous "1" is inserted. During received data processing, "0" following five continuous "1" is deleted.
Flag Detection	Write the flag data "$7E_{16}$" to the GiCMPj register (j=0 to 3) to use the special communication interrupt (the SRTiR bit in the IIO4IR register)
Abort Detection	Write the masked data "01_{16}" to the GiMSKj register
CRC	The CRC1 and CRC0 bits are set to "11_2" ($X^{16}+X^{12}+X^5+1$). The CRCV bit is set to "1" (set to "$FFFF_{16}$"). • During transmit data processing, CRC calculation result is stored into the GiTCRC register. The TCRCE bit in the GiETC register is set to "1" (transmit CRC used). The CRC calculation result is reset when the TE bit in the GiCR register is set to "0" (transmit disabled). • During received data processing, CRC calculation result is stored into the GiRCRC register. The RCRCE bit in the GiERC register is set to "1" (receive CRC used). The CRC calculation result is reset by comparing the flag data "$7E_{16}$" and matching the result with the value in the GiCMP3 register. The ACRC bit in the GiEMR register is set to "1" (CRC reset).
Data Processing Start Condition	The following conditions are required to start transmit data processing: • The TE bit in the GiCR register is set to "1" (transmit enable) • Data is written to the GiTB register The following conditions are required to start receive data processing: • The RE bit in the GiCR register is set to "1" (receive enable) • Data is written to the GiRI register

Table 22.25 HDLC Processing Mode Specifications (Continued)

Item	Specification
Interrupt Request[1]	During transmit data processing, • One of the following conditions can be selected to set the GiTOR bit in the interrupt request register to "1" (interrupt request) (see Figure 11.14). – When the IRS bit in the GiMR register is set to "0" (no data in the GiTB register) and data is transferred from the GiTB register to the transmit register (transmit start). – When the IRS bit is set to "1" (transmission completed) and data transfer from the transmit register to the GiTO register is completed. • When data, which is already converted to HDLC data, is transferred from the receive register of the GiTO register to the transmit buffer, the GiTOR bit is set to "1" During received data processing, • When data is transferred from the GiRI register to the GiRB register (reception completed), the GiRIR bit is set to "1" (See Figure 11.14). • When received data is transferred from the receive buffer of the GiRI register to the receive register, the GiRIR bit is set to "1". • When the GiTB register is compared to the GiCMPj register (j=0 to 3), the SRTiR bit is set to "1".

NOTES:

1. See **Figure 11.14** for details on the GiTOR bit, GiRIR bit and SRTiR bit.

Table 22.26 Clock Settings (Communication Unit 0)

Transfer Clock[1]	CCS Register	
	CCS0 Bit	CCS1 Bit
f_1	1	0
f_8	1	1
f_{2n}[2]	0	1

NOTES:

1. The transfer clock for reception is generated when the RSHTE bit in the G0ERC register is set to "1" (receive shift operation enabled).
2. The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2n (n=1 to 15).

Table 22.27 Clock Settings (Communication Unit 1)

Transfer Clock[1]	CCS Register	
	CCS2 Bit	CCS3 Bit
$\frac{f_{BT1}}{2x(n+2)}$ [2]	0	0
f_1	1	0
f_8	1	1
f_{2n}[3]	0	1

n: Setting value of the G1PO0 register, 0001_{16} to $FFFD_{16}$

NOTES:

1. The transfer clock for reception is generated when the RSHTE bit in the G1ERC register is set to "1" (receive shift operation enabled).
2. The transfer clock is generated in single-phase waveform output mode of the channel 1.
3. The CNT3 to CNT0 bits in the TCSPR register select no division (n=0) or divide-by-2n (n=1 to 15).

Table 22.28 Register Settings in HDLC Processing Mode (Communication Units 0 and 1)

Register	Bit	Function
G1BCR0	BCK1, BCK0	Select count source
	DIV4 to DIV0	Select divide ratio of count source
	IT	Select the base timer interrupt
G1BCR1(1)	7 to 0	Set to "0001 0010$_2$"
G1POCR0(1)	7 to 0	Set to "0000 0000$_2$"
G1POCR1(1)	7 to 0	Set to "0000 0000$_2$"
G1PO0(1)	15 to 0	Set bit rate
G1PO1(1)	15 to 0	Set the timing of the rising edge of the transfer clock. Timing of the falling edge ("H" width of the transfer clock) is fixed. Setting value of the G1PO1 register ≤ Setting value of the G1PO0 register
G1FS(1)	FSC1, FSC0	Set to "00$_2$"
G1FE(1)	IFE1, IFE0	Set to "11$_2$"
GiMR	GMD1, GMD0	Set to "11$_2$"
	CKDIR	Set to "0"
	UFORM	Set to "0"
	IRS	Select what causes the transmit interrupt to be generated
GiEMR	7 to 0	Set to "1111 0110$_2$"
GiCR	TI	Transmit buffer empty flag
	TXEPT	Transmit register empty flag
	RI	Receive complete flag
	TE	Transmit enable bit
	RE	Receive enable bit
GiETC	SOF	Set to "0"
	TCRCE	Select whether transmit CRC is used or not
	ABTE	Set to "0"
	TBSF1, TBSF0	Transmit bit stuffing
GiERC	CMP2E to CMP0E	Select whether received data is compared or not
	CMP3E	Set to "1"
	RCRCE	Select whether receive CRC is used or not
	RSHTE	Set to "1" to use it in the receiver
	RBSF1, RBSF0	Receive bit stuffing
GiIRF	BSERR, ABT	Set to "0"
	IRF3 to IRF0	Select what causes an interrupt to be generated
GiCMP0, GiCMP1	7 to 0	Write "FE$_{16}$" to abort processing
GiCMP2	7 to 0	Data to be compared
GiCMP3	7 to 0	Write "7E$_{16}$"
GiMSK0, GiMSK1	7 to 0	Write "01$_{16}$" to abort processing
GiTCRC	15 to 0	Transmit CRC calculation result can be read
GiRCRC	15 to 0	Receive CRC calculation result can be read
GiTO	7 to 0	Data, which is output from a transmit data generation circuit, can be read
GiRI	7 to 0	Set data input to a receive data generation circuit
GiRB	7 to 0	Received data is stored
GiTB	7 to 0	For transmission: write data to be transmitted For reception : received data for comparison is stored
CCS	CCS1, CCS0	Select the HDLC processing clock
	CCS3, CCS2	Select the HDLC processing clock

i=0, 1

NOTES:

1. These register settings are required when the CCS3 and CCS2 bit in the CCS register are set to "00$_2$" (clock output from channel j (j=1 to 3)).

22.5 Stepping Motor Control Function

The stepping motor control function controls up to four sets of stepping motors by microstepping.

Table 22.29 lists pin structure. Table 22.30 lists specifications of the stepping motor control function.

Table 22.29 Pin Structure

PWM Set	Output Pin	Function
Set 0	GASP0	A-phase positive terminal for Set 0
	GASM0	A-phase negative terminal for Set 0
	GACP0	B-phase positive terminal for Set 0
	GACM0	B-phase negative terminal for Set 0
Set 1	GASP1	A-phase positve terminal for Set 1
	GASM1	A-phase negative terminal for Set 1
	GACP1	B-phase positive terminal for Set 1
	GACM1	B-phase negative terminal for Set 1
Set 2	GASP2	A-phase positive terminal for Set 2
	GASM2	A-phase negative terminal for Set 2
	GACP2	B-phase positive terminal for Set 2
	GACM2	B-phase negative terminal for Set 2
Set 3	GASP3	A-phase positive terminal for Set 3
	GASM3	A-phase negative terminal for Set 3
	GACP3	B-phase positive terminal for Set 3
	GACM3	B-phase negative terminal for Set 3

Table 22.30 Stepping Motor Control Function Specifications

Item	Specification
PWM Output	4 sets (4 outputs per set)
Count Source	f_{BT1}
Carrier Wave Cycle	$\frac{1026}{f_{BTi}}$ Set the BTRE bit in the G1POCRi register (i=0 to 7) to "1" (base timer reset enabled when bit 10 in the base timer overflows)
PWM Output Waveform	Primary output width : $\frac{m}{f_{BTi}}$ Inversed level signal width : $\frac{1026- m}{f_{BTi}}$ m: setting value of the G1POi register
EMI Countermeasure	Delaying PWM output from each set reduces noise generated by output buffer switching

Figure 22.32 shows a block diagram of the stepping motor function.

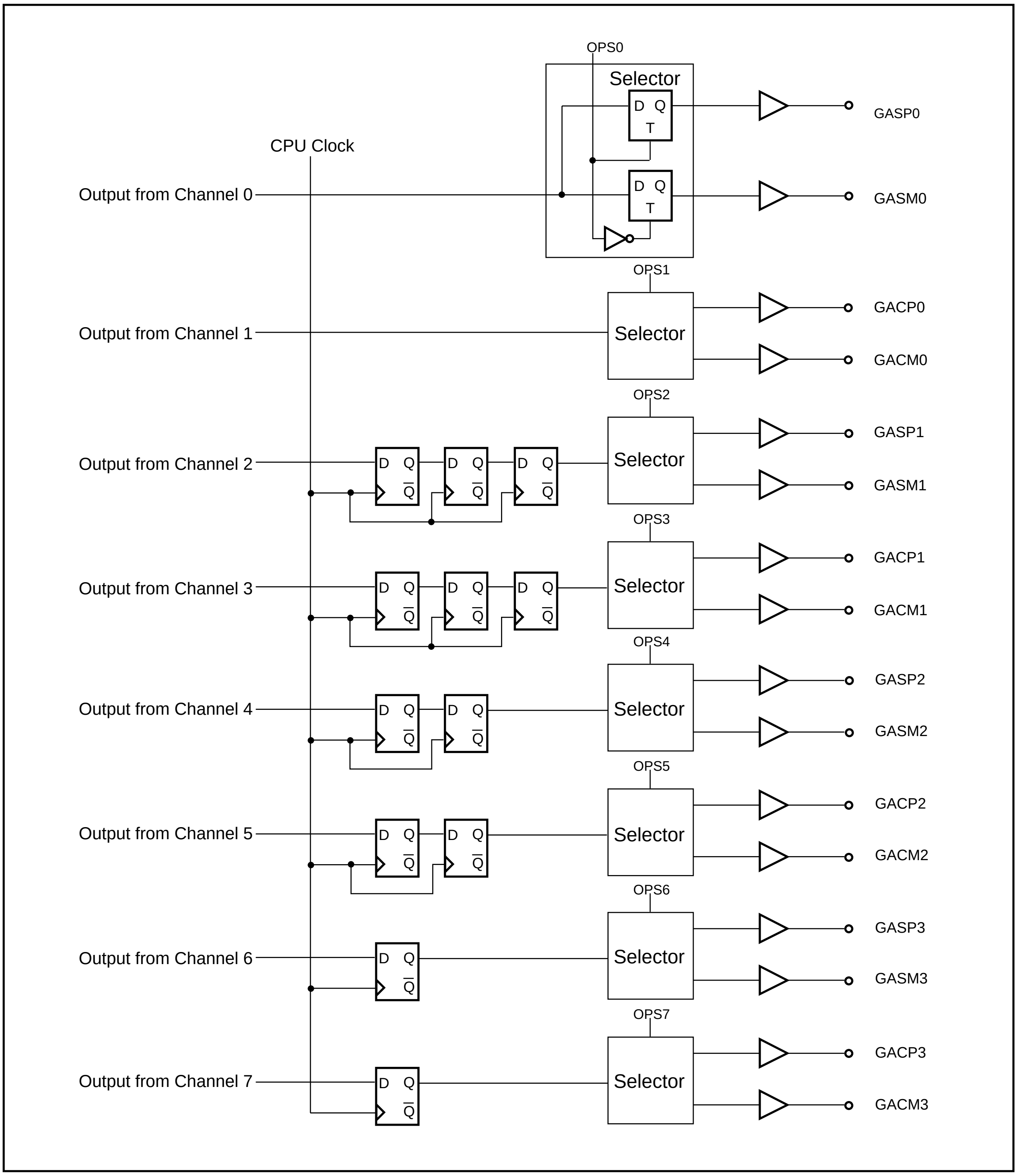
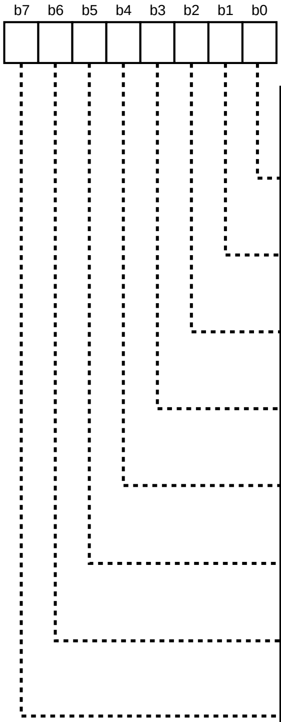


Figure 22.32 Stepping Motor Control Function Block Diagram

Figure 22.33 shows the OPS register. Figure 22.34 shows an example usage for the OPS register in the communication unit 0 while using the stepping motor control function. Table 22.31 lists stepping motor control function-associated register settings. Table 22.32 lists pin settings for when using the stepping motor control function. Table 22.28 shows a count source and PWM output frequency.

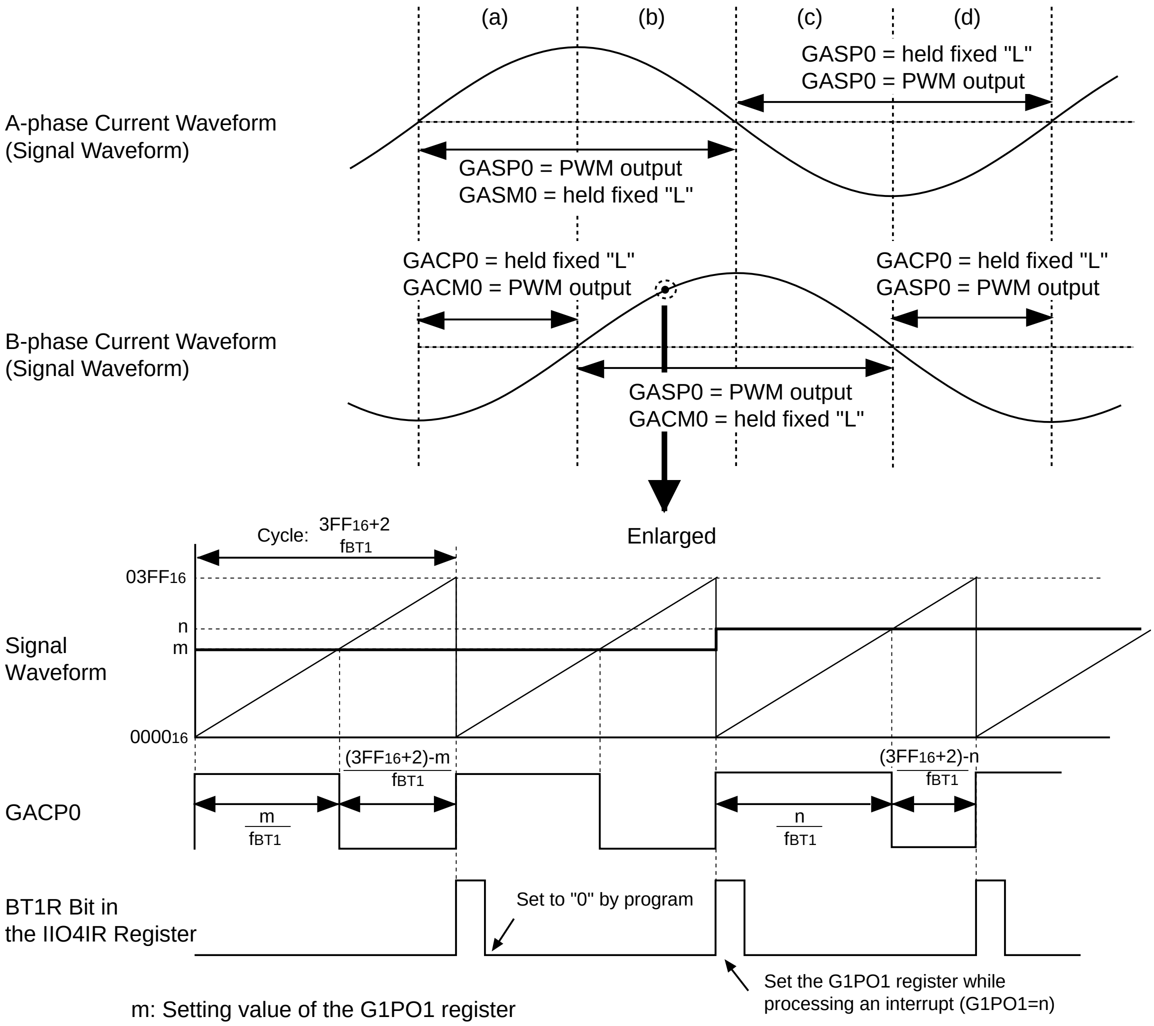
Output Port Switch Register



Symbol	Address	After Reset
OPS	$03DE_{16}$	$0000\ 0000_2$

Bit Symbol	Bit Name	Function	RW
OPS0	Port Switch Bit 0	0: PWM generated in channel 0 is output to GASP0 1: Output to GASM0	RW
OPS1	Port Switch Bit 1	0: PWM generated in channel 1 is output to GACP0 1: Output to GACM0	RW
OPS2	Port Switch Bit 2	0: PWM generated in channel 2 is output to GASP1 1: Output to GASM1	RW
OPS3	Port Switch Bit 3	0: PWM generated in channel 3 is output to GACP1 1: Output to GACM1	RW
OPS4	Port Switch Bit 4	0: PWM generated in channel 4 is output to GASP2 1: Output to GASM2	RW
OPS5	Port Switch Bit 5	0: PWM generated in channel 5 is output to GACP2 1: Output to GACM2	RW
OPS6	Port Switch Bit 6	0: PWM generated in channel 6 is output to GASP3 1: Output to GASM3	RW
OPS7	Port Switch Bit 7	0: PWM generated in channel 7 is output to GACP3 1: Output to GACM3	RW

Figure 22.35 OPS Register



Settings for the OPS0 to OPS1 bits in the OPS register during periods (a) to (d)

Periods	(a)	(b)	(c)	(d)
OPS0 bit	0	0	1	1
OPS1 bit	1	0	0	1

Pin status remains unchanged when the OPSi bit (i=0,1) is set to "0".

Figure 22.36 Usage of the OSP Register in Communication Unit 0

Table 22.26 Stepping Motor Control Function-associated Register Settings

Register	Bit	Function
G1BCR0	BCK1, BCK0	Set to "11$_2$" (f$_1$)
	DIV4 to DIV0	Select divide ratio of count source
	IT	Set to "0"
G1BCR1	-	Set to "0001 0000$_2$"
G1POCRj	MOD2 to MOD0	Set to "000$_2$"
	IVL	Set to "0"
	RLD	Set to "1"
	INV	Select output inversed or not inversed
G1PO0j	-	Select when the PWM output waveform is inversed
G1FS	FSCj	Set to "0"
G1FE(1)	IFEj	Set to "1"
OPS	OPSj	Set to "1" when the PWM waveform output is selected

i=0 to 7

Table 22.27 Pin Settings

Pins	Settings	
	PS6 and PS7 Registers	PSL6 and PSL7 Registers
P12$_0$/GASP0	PS6_0 = 1	PSL6_0 = 1
P12$_1$/GASM0	PS6_1 = 1	PSL6_1 = 1
P12$_2$/GACP0	PS6_1 = 1	PSL6_2 = 1
P12$_3$/GACM0	PS6_3 = 1	PSL6_3 = 1
P12$_4$/GASP1	PS6_4 = 1	PSL6_4 = 1
P12$_5$/GASP1	PS6_5 = 1	PSL6_5 = 1
P12$_6$/GACP1	PS6_6 = 1	PSL6_6 = 1
P12$_7$/GACM1	PS6_7 = 1	PSL6_7 = 1
P13$_0$/GASP2	PS7_0 = 1	PSL7_0 = 1
P13$_1$/GASM2	PS7_1 = 1	PSL7_1 = 1
P13$_2$/GACP2	PS7_2 = 1	PSL7_2 = 1
P13$_3$/GACM2	PS7_3 = 1	PSL7_3 = 1
P13$_4$/GASP3	PS7_4 = 1	PSL7_4 = 1
P13$_5$/GASM3	PS7_5 = 1	PSL7_5 = 1
P13$_6$/GACP3	PS7_6 = 1	PSL7_6 = 1
P13$_7$/GACM3	PS7_7 = 1	PSL7_7 = 1

Table 22.28 Count Source and PWM Output Frequency

f(X$_{IN}$)	DIV4 to DIV0 Bits in the G1BCR0 Register	Count Source (ns)	Frequency (kHz)	Cycles (µs)
30MHz	11111$_2$ (no division)	33.3	29.2	34.2
	00000$_2$ (divide-by-2)	66.7	14.6	68.4
20MHz	11111$_2$ (no division)	50	19.5	51.3
	00000$_2$ (divide-by-2)	100	9.7	102.6
16MHz	11111$_2$ (no division)	62.5	15.6	64.1

23. CAN Module

The CAN (Controller Area Network) module included in the M32C/86 group (M32C/86, M32C/86T) is a Full CAN module, compatible with CAN Specification 2.0 Part B. Two channels, CAN0 and CAN1, can be used. Table 23.1 lists specifications of the CAN module.

Table 23.1 CAN Module Specifications

Item	Specification
Protocol	CAN Specification 2.0 Part B
Message Slots	16 slots
Polarity	Dominant: "L" Recessive: "H"
Acceptance Filter	Global mask: 1 (for message slots 0 to 13) Local mask: 2 (for message slots 14 and 15 respectively)
Baud Rate	Baud rate = $\frac{1}{\text{Tq clock cycle x Tq per bit}}$ --- Max. 1 Mbps Tq clock cycle = $\frac{\text{BRP + 1}}{\text{CAN clock}}$ Tq per bit = SS + PTS +PBS1+PBS2 Tq: Time quantum BRP: Setting value of the C0BRP and C1BRP registers, 1-255 SS: Synchronization Segment; 1 Tq PTS: Propagation Time Segment; 1 to 8 Tq PBS1: Phase Buffer Segment 1; 2 to 8 Tq PBS2: Phase Buffer Segment 2 ; 2 to 8 Tq
Remote Frame Automatic Answering Function	Message slot that receives the remote frame transmits the data frame automatically
Time Stamp Function	Time stamp function with a 16-bit counter. Count source can be selected from the CAN bus bit clock divided by 1, 2, 3 or 4 CAN bus bit clock = $\frac{1}{\text{CAN bit time}}$
BasicCAN Mode	BasicCAN function can be used with the CANi message slots 14 and 15
Transmit Abort Function	Transmit request is aborted
Loopback Function	Frame transmitted by the CAN module is received by the same CAN module
Forcible Error Active Transition Function	The CAN module is forced into an error active state by resetting an error counter.
Single-Shot Transmit Function	The CAN module does not transmit data again even if arbitration lost or transmission error causes a transmission failure
Self-Test Function	The CAN module communicates internally and diagnoses its CAN module state

NOTES:

1. Use an oscillator with maximum 1.58% oscillator tolerance.

Figure 23.1 shows a block diagram of the CAN module. Figure 23.2 shows CANi message slot (the message slot) j (j = 0 to 15) and CANi message slot buffer (i=0, 1). Table 23.2 lists pin settings of the CAN module.

The message slot cannot be accessed directly from the CPU. Allocate the message slot j to be used to the message slot buffer 0 or 1. The message slot j is accessed via the message slot buffer address. The CiSBS register selects the message slot j to be allocated. Figure 23.2 shows the 16-byte message slot buffer and message slot.

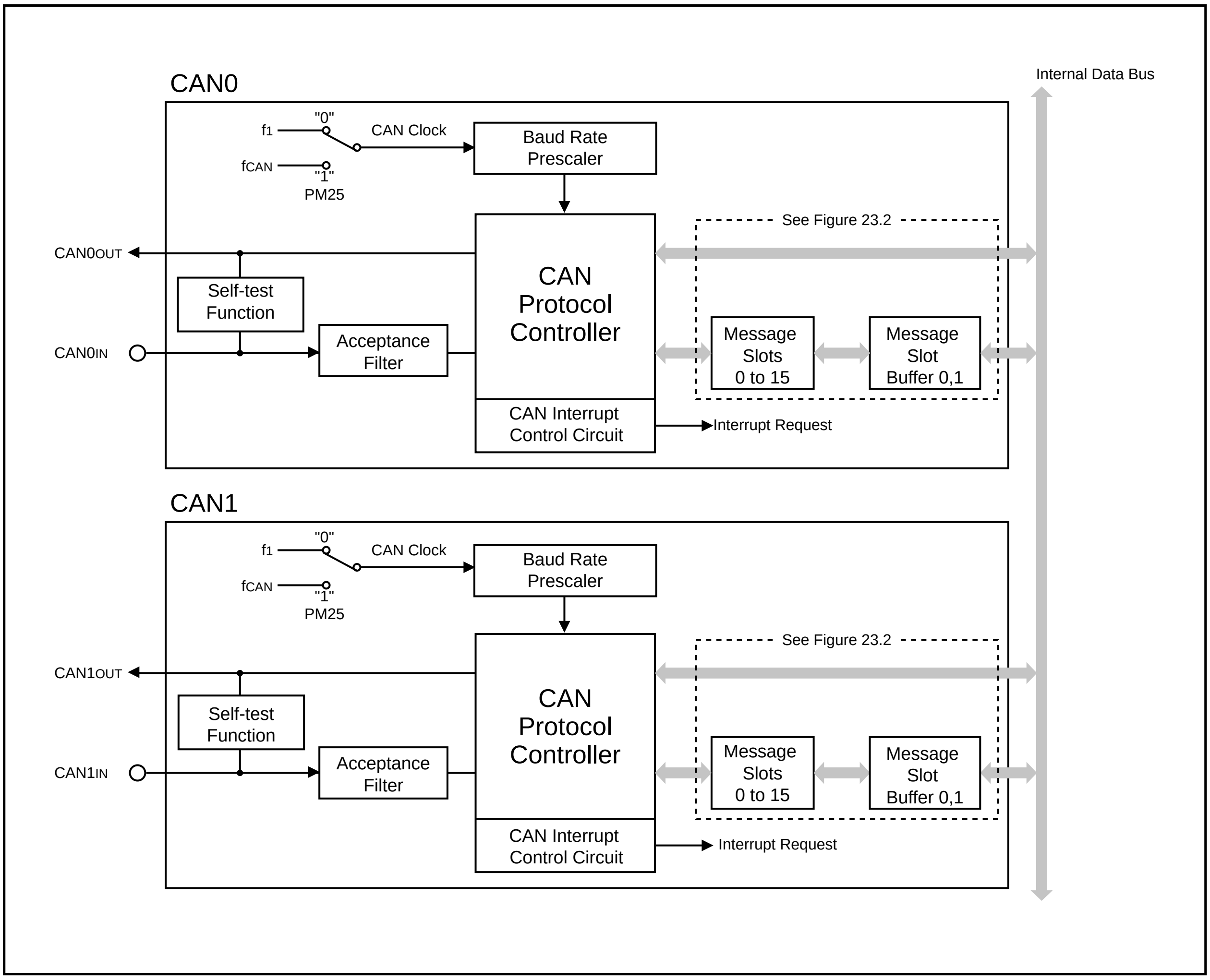


Figure 23.1 CAN Module Block Diagram

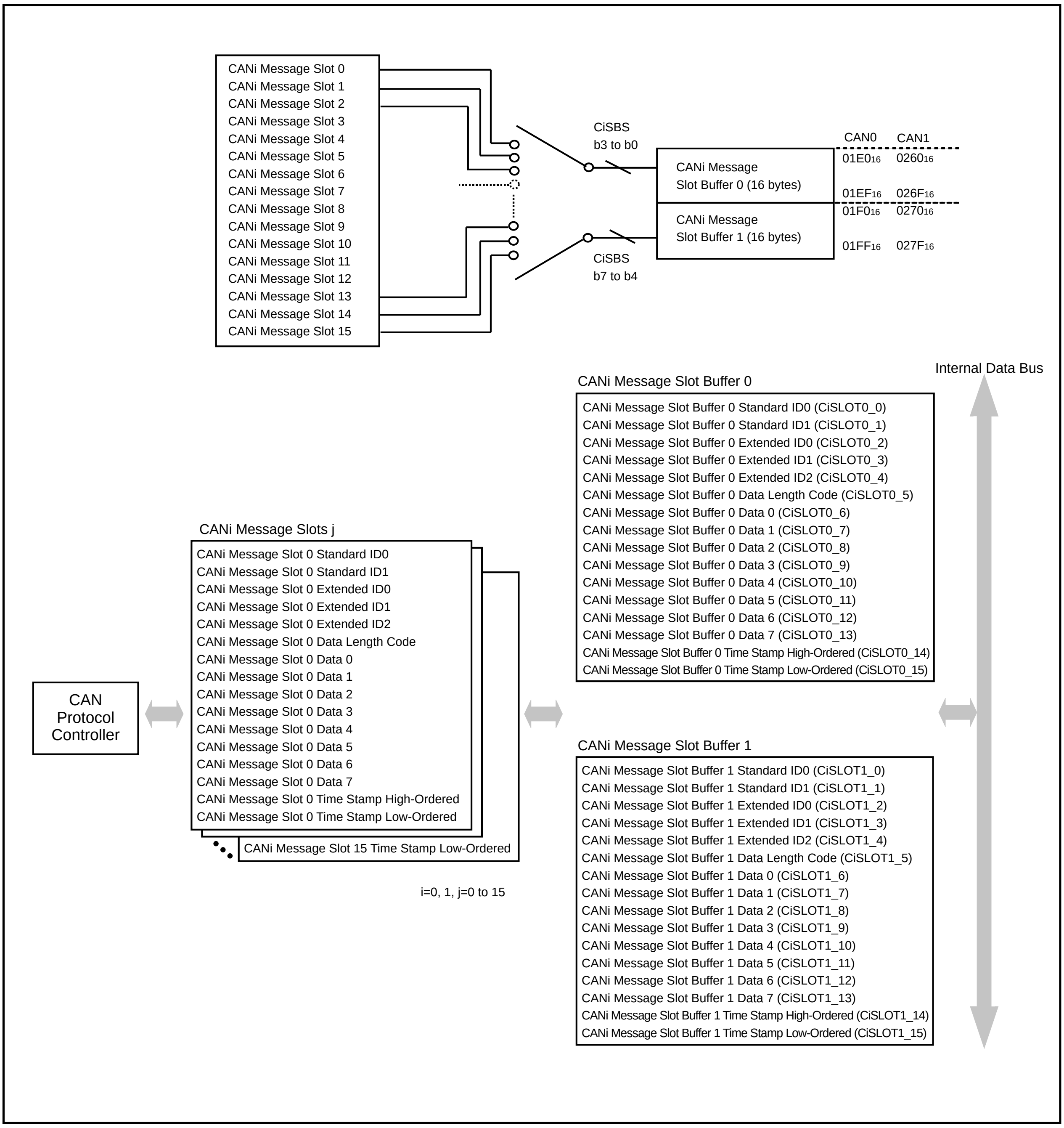

CANi Message Slot 0
CANi Message Slot 1
CANi Message Slot 2
CANi Message Slot 3
CANi Message Slot 4
CANi Message Slot 5
CANi Message Slot 6
CANi Message Slot 7
CANi Message Slot 8
CANi Message Slot 9
CANi Message Slot 10
CANi Message Slot 11
CANi Message Slot 12
CANi Message Slot 13
CANi Message Slot 14
CANi Message Slot 15
CiSBS
b3 to b0
CiSBS
b7 to b4
CANi Message
Slot Buffer 0 (16 bytes)
CANi Message
Slot Buffer 1 (16 bytes)
CAN0
CAN1
01E0₁₆
0260₁₆
01EF₁₆
026F₁₆
01F0₁₆
0270₁₆
01FF₁₆
027F₁₆
Internal Data Bus
CANi Message Slot Buffer 0
CANi Message Slot Buffer 0 Standard ID0 (CiSLOT0_0)
CANi Message Slot Buffer 0 Standard ID1 (CiSLOT0_1)
CANi Message Slot Buffer 0 Extended ID0 (CiSLOT0_2)
CANi Message Slot Buffer 0 Extended ID1 (CiSLOT0_3)
CANi Message Slot Buffer 0 Extended ID2 (CiSLOT0_4)
CANi Message Slot Buffer 0 Data Length Code (CiSLOT0_5)
CANi Message Slot Buffer 0 Data 0 (CiSLOT0_6)
CANi Message Slot Buffer 0 Data 1 (CiSLOT0_7)
CANi Message Slot Buffer 0 Data 2 (CiSLOT0_8)
CANi Message Slot Buffer 0 Data 3 (CiSLOT0_9)
CANi Message Slot Buffer 0 Data 4 (CiSLOT0_10)
CANi Message Slot Buffer 0 Data 5 (CiSLOT0_11)
CANi Message Slot Buffer 0 Data 6 (CiSLOT0_12)
CANi Message Slot Buffer 0 Data 7 (CiSLOT0_13)
CANi Message Slot Buffer 0 Time Stamp High-Ordered (CiSLOT0_14)
CANi Message Slot Buffer 0 Time Stamp Low-Ordered (CiSLOT0_15)
CANi Message Slots j
CANi Message Slot 0 Standard ID0
CANi Message Slot 0 Standard ID1
CANi Message Slot 0 Extended ID0
CANi Message Slot 0 Extended ID1
CANi Message Slot 0 Extended ID2
CANi Message Slot 0 Data Length Code
CANi Message Slot 0 Data 0
CANi Message Slot 0 Data 1
CANi Message Slot 0 Data 2
CANi Message Slot 0 Data 3
CANi Message Slot 0 Data 4
CANi Message Slot 0 Data 5
CANi Message Slot 0 Data 6
CANi Message Slot 0 Data 7
CANi Message Slot 0 Time Stamp High-Ordered
CANi Message Slot 0 Time Stamp Low-Ordered
CANi Message Slot 15 Time Stamp Low-Ordered
CAN Protocol Controller
i=0, 1, j=0 to 15
CANi Message Slot Buffer 1
CANi Message Slot Buffer 1 Standard ID0 (CiSLOT1_0)
CANi Message Slot Buffer 1 Standard ID1 (CiSLOT1_1)
CANi Message Slot Buffer 1 Extended ID0 (CiSLOT1_2)
CANi Message Slot Buffer 1 Extended ID1 (CiSLOT1_3)
CANi Message Slot Buffer 1 Extended ID2 (CiSLOT1_4)
CANi Message Slot Buffer 1 Data Length Code (CiSLOT1_5)
CANi Message Slot Buffer 1 Data 0 (CiSLOT1_6)
CANi Message Slot Buffer 1 Data 1 (CiSLOT1_7)
CANi Message Slot Buffer 1 Data 2 (CiSLOT1_8)
CANi Message Slot Buffer 1 Data 3 (CiSLOT1_9)
CANi Message Slot Buffer 1 Data 4 (CiSLOT1_10)
CANi Message Slot Buffer 1 Data 5 (CiSLOT1_11)
CANi Message Slot Buffer 1 Data 6 (CiSLOT1_12)
CANi Message Slot Buffer 1 Data 7 (CiSLOT1_13)
CANi Message Slot Buffer 1 Time Stamp High-Ordered (CiSLOT1_14)
CANi Message Slot Buffer 1 Time Stamp Low-Ordered (CiSLOT1_15)

Figure 23.2 CANi Message Slot and CANi Message Slot Buffer

Table 23.2 Pin Settings

Port	Function	Bit and Setting				
		IPS, IPSA Registers	PS1, PS2, PS3 Registers(1)	PSL1, PSL2, PSL3 Registers	PSC, PSC2, PSC3 Registers	PD7, PD8, PD9(1) Regsiters
P7$_6$	CAN0$_{OUT}$	–	PS1_6=1	PSL1_6=0	PSC_6=1	–
P7$_7$	CAN0$_{IN}$	IPS3=0	PS1_7=0	–	–	PD7_7=0
P8$_2$	CAN0$_{OUT}$	–	PS2_2=1	PSL2_2=1	PSC2_2=0	–
	CAN1$_{OUT}$	–	PS2_2=1	PSL2_2=1	PSC2_2=1	–
P8$_3$	CAN0$_{IN}$	IPS3=1	–	–	–	PD8_3=0
	CAN1$_{IN}$	IPSA_3=1	–	–	–	PD8_3=0
P9$_5$	CAN1$_{IN}$	IPSA_3=0	PS3_5=0	PSL3_5=0	–	PD9_5=0
P9$_6$	CAN1$_{OUT}$	–	PS3_6=1	–	PSC3_6=1	–

NOTES:

1. Set the PD9 and PS3 registers immediately after the PRC2 bit in the PRCR register is set to "1" (write enable). Do not generate an interrupt or a DMA transfer between the instruction to set to the PRC2 bit to "1" and the instruction to set the PD9 and PS3 registers.

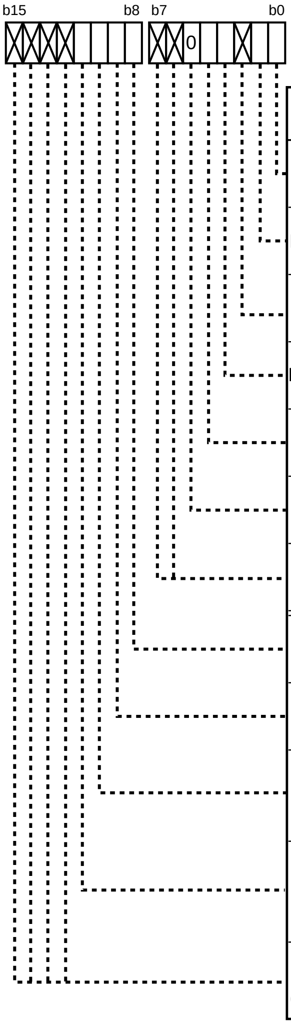
23.1 CAN-Associated Registers

Figures 23.3 to 23.18, and Figures 23.20 to 23.33 show registers associated with CAN. To access the CAN-associated registers, set the CM21 bit in the CM2 register to "0" (main clock or PLL clock as CPU clock) and the MCD4 to MCD0 bits in the MCD register to "10010_2" (no division mode). Or, set the PM24 bit in the PM2 register to "1" (main clock direct mode) and the PM25 bit in the PM2 regiseter to "1" (CAN clock). Two wait states are added into the bus cycle.

Refer to **7. Processor Mode** and **9. Clock Generation Circuit**.

23.1.1 CANi Control Register 0 (CiCTLR0 Register) (i=0, 1)

CANi Control Register 0 (i=0, 1)



Symbol	Address	After Reset[1]
C0CTLR0	0201_{16} - 0200_{16}	$XXXX 0000 XX01 0X01_2$
C1CTLR0	0281_{16} - 0280_{16}	$XXXX 0000 XX01 0X01_2$

Bit Symbol	Bit Name	Function	RW
RESET0	CAN Reset Bit 0	0: CAN module reset exited 1: CAN module is reset[2]	RW
LOOPBACK	Loop Back Mode Select Bit	0: Disables loop back function 1: Enables loop back function	RW
(b2)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		–
BASICCAN	BasicCAN Mode Select Bit	0: Disables BasicCAN mode function 1: Enables BasicCAN mode function	RW
RESET1	CAN Reset Bit 1	0: CAN module reset exited 1: CAN module is reset[2]	RW
(b5)	Reserved Bit	Set to "0"	RW
(b7 - b6)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		–
TSPRE0	Time Stamp Prescaler Select Bit	b9 b8 0 0: Selects the CAN bus bit clock 0 1: Selects the CAN bus bit clock divided by 2 1 0: Selects the CAN bus bit clock divided by 3 1 1: Selects the CAN bus bit clock divided by 4	RW
TSPRE1			RW
TSRESET	Time Stamp Counter Reset Bit	0: Nothing is occurred 1: This bit is automatically set to "0" after the CiTSR register is set to "0000_{16}"[3]	RW
ECRESET	Error Counter Reset Bit	0: Nothing is occurred 1: This bit is automatically set to "0" after the CiTEC and CiREC registers are set to "00_{16}"[3]	RW
(b15 - b12)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		–

NOTES:

1. Value is obtained by setting the SLEEP bit in the CiSLPR register to "1" (sleep mode exited) after reset and supplying the clock to the CAN module.
2. Set the RESET1 and RESET0 bits to the same value simultaneously.
3. These bits can only be set to "1", not "0", by program.

Figure 23.3 C0CTLR0 and C1CTLR0 Registers

23.1.1.1 RESET1 and RESET0 Bits

When both RESET1 and RESET0 bits are set to "1" (CAN module reset), the CAN module is immediately initialized regardless of ongoing CAN communication.

After the RESET1 and RESET0 bits are set to "1" and the CAN module reset is completed, the CiTSR register (i=0, 1) is set to "0000_{16}". The CiTEC and CiREC registers are set to "00_{16}" and the STATE_ERRPAS and STATE_BUSOFF bits in the CiSTR register are set to "0" as well.

When both RESET1 and RESET0 bit settings are changed "1" to "0", the CiTSR register starts counting. CAN communication is available after 11 continuous recessive bits are detected.

NOTES:

1. Set the same value in both RESET1 and RESET0 bits simultaneously.
2. Confirm that the STATE_RESET bit in the CiSTR register is set to "1" (CAN module reset completed) after setting the RESET1 and RESET0 bits to "1".
3. The CANOUT pin puts in a high-level ("H") signal as soon as the RESET1 and RESET0 bits are set to "1". CAN bus error may occur when the RESET1 and RESET0 bits are set to "1" while the CAN frame is transmitting.
4. For CAN communication, set the PS1, PS2, PS3, PSL1, PSL2, PSL3, PSC, PSC2, PSC3, IPS, IPSA, PD7, PD8 and PD9 registers when the STATE_RESET bit is set to "1" (CAN module reset completed).

23.1.1.2 LOOPBACK Bit

When the LOOPBACK bit is set to "1" (loopback function enabled) and the receive message slot has a matched ID and frame format with a transmitted frame, the transmitted frame is stored to the receive message slot.

NOTES:

1. No ACK for the transmitted frame is returned.
2. Change the LOOPBACK bit setting only when the STATE_RESET bit is set to "1" (CAN module reset completed).

23.1.1.3 BASICCAN Bit

When the BASICCAN bit is set to "1", the message slots 14 and 15 enter BasicCAN mode.

In BasicCAN mode, the message slots 14 and 15 are used as dual-structured buffers. The message slots 14 and 15 alternately store a received frame having matched ID detected by acceptance filtering. ID in the message slot 14 and the CiLMAR0 to CiLMAR4 registers are used for acceptance filtering when the message slot 14 is active (the next received frame is to be stored in the message slot 14). ID in the message slot 15 and the CiLMBR0 to CiLMBR4 registers are used when the message slot 15 is active. Both data frame and remote frame can be received.

Use the following procedure to enter BasicCAN mode.

(1) Set the BASICCAN bit to "1".

(2) Set the same value into IDs in the message slots 14 and 15.

(3) Set the same value in the CiLMAR0 to CiLMAR4 registers and CiLMBR0 to CiLMBR4 registers.

(4) Set the IDE14 and IDE15 bits in the CiIDR register to select a frame format (standard or extended) for the message slots 14 and 15. (Set to the same format.)

(5) Set the CiMCTL14 and CiMCTL15 registers in the message slots 14 and 15 to receive the data frame.

NOTES:

1. Change the BASICCAN bit setting only when the STATE_RESET bit is set to "1" (CAN module reset completed).
2. The message slot 14 is the first slot to become active after the RESET1 and RESET0 bits are set to "0".
3. The message slots 0 to 13 are not affected by entering BasicCAN mode.

23.1.1.4 TSPRE1, TSPRE0 Bits

The TSPRE1 and TSPRE0 bits determine which count source is used for the time stamp counter.

NOTES:

1. Change the TSPRE1 and TSPRE0 bit settings only when the STATE_RESET bit is set to "1" (CAN module reset completed).

23.1.1.5 TSRESET Bit

When the TSRESET bit is set to "1", the CiTSR register is set to "0000_{16}". The TSRESET bit is automatically set to "0" after the CiTSR register is set to "0000_{16}".

23.1.1.6 ECRESET Bit

When the ECRESET bit is set to "1", the CiTEC and CiREC registers are set to "00_{16}". The CAN module forcibly goes into an error active state.

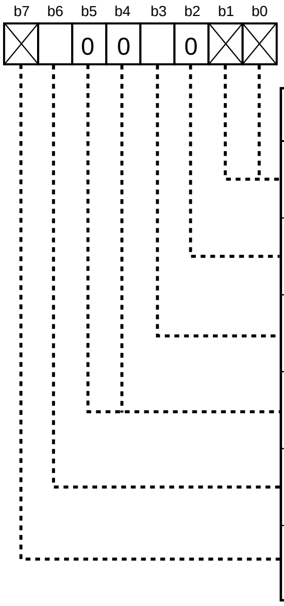
The ECRESET bit is automatically set to "0" after the CAN module enters an error active state.

NOTES:

1. In an error active state, the CAN module is ready to communicate when 11 continuous recessive bits are detected on the CAN bus.
2. The CAN$_{iOUT}$ pin provides an "H" signal output as soon as the ECRESET bit is set to "1". The CAN bus error may occur when setting the ECRESET bit to "1" during CAN frame transmission.

23.1.2 CANi Control Register 1 (CiCTLR1 Register) (i=0, 1)

CANi Control Register 1 (i=0, 1)



Symbol	Address	After Reset[1]
C0CTLR1	0241_{16}	$X000\ 00XX_2$
C1CTLR1	0251_{16}	$X000\ 00XX_2$

Bit Symbol	Bit Name	Function	RW
(b1 - b0)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		–
(b2)	Reserved Bit	Set to "0"	RW
BANKSEL	CANi Bank Switch Bit	0: Selects the message slot control register and single-shot register 1: Selects the mask register	RW
(b5 - b4)	Reserved Bit	Set to "0"	RW
INTSEL	CANi Interrupt Mode Select Bit	0: Outputs 3 types of interrupts via OR 1: Outputs 3 types of interrupts separately	RW
(b7)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		–

NOTES:

1. Value is obtained by setting the SLEEP bit in the CiSLPR register to "1" (sleep mode exited) after reset and supplying the clock to the CAN module.

Figure 23.4 C0CTLR1 and C1CTLR1 Registers

23.1.2.1 BANKSEL Bit

The BANKSEL bit in the C0CTLR1 register selects the registers allocated to addresses 0220_{16} to $023F_{16}$. The BANKSEL bit in the C1CTLR1 register selects registers allocated to addresses $02A0_{16}$ to $02BF_{16}$.

The CiSSCTLR register, CiSSSTR register and the CiMCTL0 to CiMCTL15 registers can be accessed by setting the BANKSEL bit to "0". The CiGMR0 to CiGMR4 registers, CiLMAR0 to CiLMAR4 registers and CiLMBR0 to CiLMBR4 registers can be accessed by setting the BANKSEL bit to "1".

23.1.2.2 INTSEL Bit

The INTSEL bit determines whether the three types of interrupt outputs (CANi transmit interrupt, CANi receive interrupt and CANi error interrupt) are provided via OR or is separately.

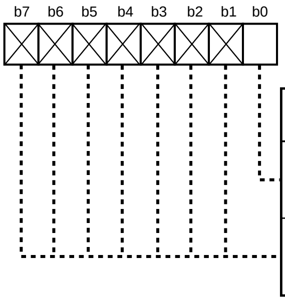
Refer to **23.4 CAN Interrupts** for details.

NOTES:

1. Change the INTSEL bit setting when the STATE_RESET bit is set to "1" (CAN module reset completed).

23.1.3 CANi Sleep Control Register (CiSLPR Register) (i=0, 1)

CANi Sleep Control Register (i=0, 1)



Symbol	Address	After Reset
C0SLPR	0242_{16}	XXXX $XXX0_2$
C1SLPR	0252_{16}	XXXX $XXX0_2$

Bit Symbol	Bit Name	Function	RW
SLEEP	Sleep Mode Control Bit	0 : Enters sleep mode 1 : Exits sleep mode(1)	RW
____ (b7 - b1)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		–

NOTES:

1. Set up the initial setting for the CAN module after CAN sleep mode is exited. While the CAN0 module is in sleep mode, no SFR (addresses $01E0_{16}$ to 0245_{16}) for CAN0, except the C0SLPR register, can be accessed. While the CAN1 module is in sleep mode, no SFR (addresses 0250_{16} to $02BF_{16}$) for CAN1, except the C1SLPR register, can be accessed.

Figure 23.5 C0SLPR and C1SLPR Registers

23.1.3.1 SLEEP Bit

When the SLEEP bit is set to "0", the clock supplied to the CAN module stops running and the CAN module enters sleep mode.

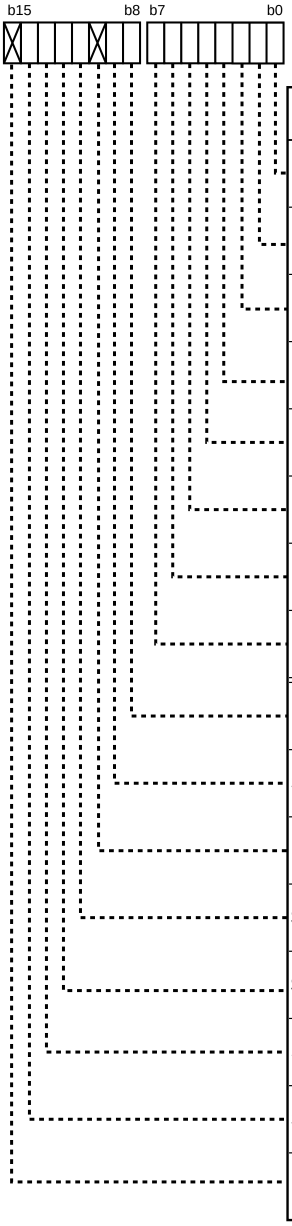
When the SLEEP bit is set to "1", the clock supplied to the CAN module starts running and the CAN module exits sleep mode.

NOTES:

1. Enter sleep mode after the STATE_RESET bit in the CiSTR register is set to "1" (CAN module reset completed).

23.1.4 CANi Status Register (CiSTR Register) (i=0, 1)

CANi Status Register (i=0, 1)



Symbol	Address	After Reset[1]
C0STR	0203_{16} - 0202_{16}	X000 0X01 0000 0000_2
C1STR	0283_{16} - 0282_{16}	X000 0X01 0000 0000_2

Bit Symbol	Bit Name	Function	RW
MBOX0	Active Slot Determination Bit	b3 b2 b1 b0 0 0 0 0 : Message slot 0 0 0 0 1 : Message slot 1 0 0 1 0 : Message slot 2 0 0 1 1 : Message slot 3 ⋮ ⋮ 1 1 0 1 : Message slot 13 1 1 1 0 : Message slot 14 1 1 1 1 : Message slot 15	RO
MBOX1			RO
MBOX2			RO
MBOX3			RO
TRMSUCC	Transmit Complete State Flag	0: Transmission is not completed 1: Transmission is completed	RO
RECSUCC	Receive Complete State Flag	0: Reception is not completed 1: Reception is completed	RO
TRMSTATE	Transmit State Flag	0: Not transmitting 1: During transmission	RO
RECSTATE	Receive State Flag	0: Not receiving 1: During reception	RO
STATE_RESET	CAN Reset State Flag	0: CAN module is operating 1: CAN module reset is completed	RO
STATE_LOOPBACK	Loop Back State Flag	0: Mode except Loop back mode 1: Loop back mode	RO
____ (b10)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		–
STATE_BASICCAN	BasicCAN State Flag	0: Mode except BasicCAN mode 1: BasicCAN mode	RO
STATE_BUSERROR	CAN Bus Error State Flag	0: No error occurs 1: Error occurs	RO
STATE_ERRPAS	Error Passive State Flag	0: No error passive state 1: Error passive state	RO
STATE_BUSOFF	Bus-Off State Flag	0: No bus-off state 1: Bus-off state	RO
____ (b15)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		–

NOTES:

1. Value is obtained by setting the SLEEP bit in the CiSLPR register to "1" (sleep mode exited) after reset and supplying the clock to the CAN module.

Figure 23.6 C0STR and C1STR Registers

23.1.4.1 MBOX3 to MBOX0 Bits

The MBOX3 to MBOX0 bits store relevant slot numbers when the CAN module has completed transmitting data or storing received data.

23.1.4.2 TRMSUCC Bit

The TRMSUCC bit is set to "1" when the CAN module has transmitted data as expected.

The TRMSUCC bit is set to "0" when the CAN module has received data as expected.

23.1.4.3 RECSUCC Bit

The RECSUCC bit is set to "1" when the CAN module has received data as expected. (Whether received message has been stored in the message slot or not is irrelevant.) If the received message is transmitted in loopback mode, the TRMSUCC bit is set to "1" and the RECSUCC bit is set to "0". The RECSUCC bit is set to "0" when the CAN module has transmitted data as expected.

23.1.4.4 TRMSTATE Bit

The TRMSTATE bit is set to "1" when the CAN module is performing as a transmit node.
The TRMSTATE bit is set to "0" when the CAN module is in a bus-idle state or starts performing as a receive node.

23.1.4.5 RECSTATE Bit

The RECSTATE bit is set to "1" when the CAN module is performing as a receive node.
The RECSTATE bit is set to "0" when the CAN module is in a bus-idle state or starts performing as a transmit node.

23.1.4.6 STATE_RESET Bit

After both RESET1 and RESET0 bits are set to "1" (CAN module reset), the STATE_RESET bit is set to "1" as soon as the CAN module is initialized.
The STATE_RESET bit is set to "0" when the RESET1 and RESET0 bits are set to "0".

23.1.4.7 STATE_LOOPBACK Bit

The STATE_ LOOPBACK bit is set to "1" when the CAN module is in loopback mode.
The STATE_LOOPBACK bit is set to "1" when the LOOPBACK bit in the CiCTLR0 register is set to "1" (loop back function enabled).
The STATE_LOOPBACK bit is set to "0" when the LOOPBACK bit is set to "0" (loop back function disabled).

23.1.4.8 STATE_BASICCAN Bit

The STATE_BASICCAN bit is set to "1" when the CAN module is in BasicCAN mode.
Refer to **23.1.1.3 BASICCAN bit** for BasicCAN mode.
The STATE_BASICCAN bit is set to "0" when the BASICCAN bit is set to "0" (BasicCAN mode function disabled).
The STATE_BASICCAN bit is set to "1" when the BASICCAN bit is set to "1" (BasicCAN mode function enabled), the REMACTIVE bits in the CiMCTL14 and CiMCTL15 registers in the message slots 14 and 15 are set to "0" (data frame received).

23.1.4.9 STATE_BUSERROR Bit

The STATE_BUSERROR bit is set to "1" when an CAN communication error is detected.
The STATE_BUSERROR bit is set to "0" when the CAN module has transmitted or received data as expected. Whether a received message has been stored into the message slot or not is irrelevant.

NOTES:

1. When the STATE_BUSERROR bit is set to "1", the STATE_BUSERROR bit remains unchanged even if both RESET1 and RESET0 bits are set to "1" (CAN module reset).

23.1.4.10 STATE_ERRPAS Bit

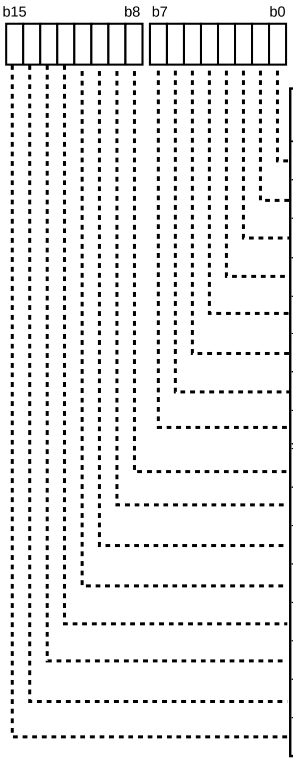
The STATE_ERRPAS bit is set to "1" when the value of the CiTEC or CiREC register (i=0, 1) exceeds 127 and the CAN module is placed in an error-passive state.
The STATE_ERRPAS bit is set to "0" when the CAN module in an error-passive state is placed in another error state.
The STATE_ERRPAS bit is set to "0" when both RESET1 and RESET0 bits are set to "1" (CAN module is reset).

23.1.4.11 STATE_BUSOFF Bit

The STATE_BUSOFF bit is set to "1" when the value of the CiTEC register exceeds 255 and the CAN module is placed in a bus-off state.
The STATE_BUSOFF bit is set to "0" when the CAN module in a bus-off state is placed in an error-active state.
The STATE_BUSOFF bit is set to "0" when both RESET1 and RESET0 bits are set to "1" (CAN module reset).

23.1.5 CANi Extended ID Register (CiIDR Register) (i=0, 1)

CANi Extended ID Register[1] (i=0, 1)



Symbol	Address	After Reset[2]
C0IDR	0205_{16} - 0204_{16}	0000_{16}
C1IDR	0285_{16} - 0284_{16}	0000_{16}

Bit Symbol	Bit Name	Function	RW
IDE15	Extended ID15 (Message Slot 15)	Standard or extended format is set by the corresponding message slot 0: Standard format 1: Extended format	RW
IDE14	Extended ID14 (Message Slot 14)		RW
IDE13	Extended ID13 (Message Slot 13)		RW
IDE12	Extended ID12 (Message Slot 12)		RW
IDE11	Extended ID11 (Message Slot 11)		RW
IDE10	Extended ID10 (Message Slot 10)		RW
IDE9	Extended ID9 (Message Slot 9)		RW
IDE8	Extended ID8 (Message Slot 8)		RW
IDE7	Extended ID7 (Message Slot 7)		RW
IDE6	Extended ID6 (Message Slot 6)		RW
IDE5	Extended ID5 (Message Slot 5)		RW
IDE4	Extended ID4 (Message Slot 4)		RW
IDE3	Extended ID3 (Message Slot 3)		RW
IDE2	Extended ID2 (Message Slot 2)		RW
IDE1	Extended ID1 (Message Slot 1)		RW
IDE0	Extended ID0 (Message Slot 0)		RW

NOTES:

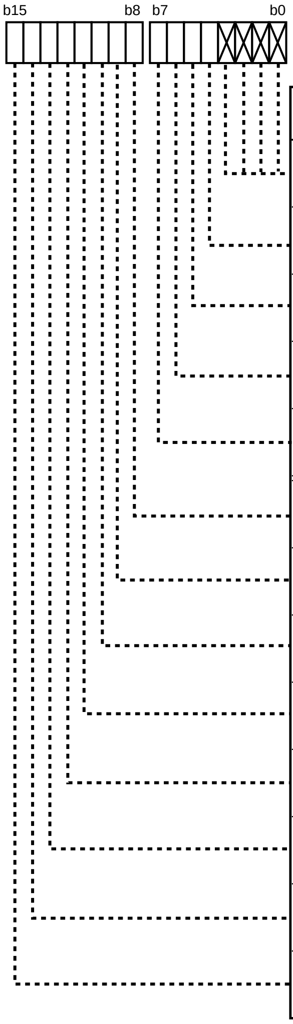
1. Change the CiIDR register setting while the CiMCTLj (j=0to 15) register, corresponding to bits to be changed, is set to "00_{16}".
2. Value is obtained by setting the SLEEP bit in the CiSLPR register to "1" (sleep mode exited) after reset and supplying the clock to the CAN module.

Figure 23.7 C0IDR and C1IDR Registers

Bits in the CiIDR register determine the frame format in the message slot corresponding to each bit.
The standard format is selected when the bit is set to "0".
The extended format is selected when the bit is to set "1".

23.1.6 CANi Configuration Register (CiCONR Register) (i=0, 1)

CANi Configuration Register (i=0, 1)



Symbol	Address	After Reset[1]
C0CONR	0207_{16} - 0206_{16}	0000 0000 0000 XXXX$_2$
C1CONR	0287_{16} - 0286_{16}	0000 0000 0000 XXXX$_2$

Bit Symbol	Bit Name	Function	RW
_____ (b3 - b0)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		—
SAM	Sampling Number	0: Sampled once 1: Sampled three times	RW
PTS0	Propagation Time Segment	b7 b6 b5 0 0 0: 1Tq 0 0 1: 2Tq 0 1 0: 3Tq 0 1 1: 4Tq 1 0 0: 5Tq 1 0 1: 6Tq 1 1 0: 7Tq 1 1 1: 8Tq	RW
PTS1			RW
PTS2			RW
PBS10	Phase Buffer Segment 1	b10 b9 b8 0 0 0: Do not set to this value 0 0 1: 2Tq 0 1 0: 3Tq 0 1 1: 4Tq 1 0 0: 5Tq 1 0 1: 6Tq 1 1 0: 7Tq 1 1 1: 8Tq	RW
PBS11			RW
PBS12			RW
PBS20	Phase Buffer Segment 2	b13 b12 b11 0 0 0: Do not set to this value 0 0 1: 2Tq 0 1 0: 3Tq 0 1 1: 4Tq 1 0 0: 5Tq 1 0 1: 6Tq 1 1 0: 7Tq 1 1 1: 8Tq	RW
PBS21			RW
PBS22			RW
SJW0	reSynchronization Jump Width	b15 b14 0 0: 1Tq 0 1: 2Tq 1 0: 3Tq 1 1: 4Tq	RW
SJW1			RW

NOTES:

1. Value is obtained by setting the SLEEP bit in the CiSLPR register to "1" (sleep mode exited) after reset and supplying the clock to the CAN module.

Figure 23.8 C0CONR and C1CONR Registers

23.1.6.1 SAM Bit

The SAM bit determines the number of sample points to be taken per bit.

When the SAM bit is set to "0", only one sample is taken per bit at the end of the Phase Buffer Segment 1 (PBS1) to determine the value of the bit.

When the SAM bit is set to "1", three samples per bit are taken; one time quantum and two time quanta before the end of PBS1, and at the end of PBS1. The sample result value which is detected more than twice becomes the value of the bit sampled.

23.1.6.2 PTS2 to PTS0 Bits

The PTS2 to PTS0 bits determine PTS width.

23.1.6.3 PBS12 to PBS10 Bits

The PBS12 to PBS10 bits determine PBS1 width. Set the PBS12 to 10 bits to "001_2" or more.

23.1.6.4 PBS22 to PBS20 Bits

The PBS22 to PBS20 bits determine PBS2 width. Set the PBS22 to PBS20 bits to "001_2" or more.

23.1.6.5 SJW1 and SJW0 Bits

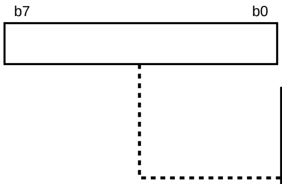
The SJW1 and SJW0 bits determine SJW width. Set the SJW1 and SJW0 bits to values less than or equal to the PBS12 to PBS10 bit settings and the PBS22 to PBS20 bit settings.

Table 23.3 Bit Timing when CPU Clock = 30 MHz

Baud Rate	BRP	Tq Clock Cycles (ns)	Tq Per Bit	PTS+PBS1	PBS2	Sample Point
1Mbps	1	66.7	15	12	2	87%
	1	66.7	15	11	3	80%
	1	66.7	15	10	4	73%
	2	100	10	7	2	80%
	2	100	10	6	3	70%
	2	100	10	5	4	60%
500Kbps	2	100	20	16	3	85%
	2	100	20	15	4	80%
	2	100	20	14	5	75%
	3	133.3	15	12	2	87%
	3	133.3	15	11	3	80%
	3	133.3	15	10	4	73%
	4	166.7	12	9	2	83%
	4	166.7	12	8	3	75%
	4	166.7	12	7	4	67%
	5	200	10	7	2	80%
	5	200	10	6	3	70%
	5	200	10	5	4	60%

23.1.7 CANi Baud Rate Prescaler (CiBRP Register) (i=0, 1)

CANi Baud Rate Prescaler (i=0, 1)



Symbol	Address	After Reset[1]
C0BRP	0217_{16}	$0000\ 0001_{16}$
C1BRP	0297_{16}	$0000\ 0001_{16}$

Function	Setting Range	RW
If setting value is *n*, the CPU clock is divided by (*n*+1).	01_{16} to FF_{16}[2]	RW

NOTES:

1. Value is obtained by setting the SLEEP bit in the CiSLPR register to "1" (sleep mode exited) after reset and supplying the clock to the CAN module.
2. Do not set to "00_{16}" (divide-by-1).

Figure 23.9 C0BRP and C1BRP Registers

The CiBRP register determines the Tq clock cycle of the CAN bit time. The baud rate is obtained from Tq clock cycle x Tq per bit.

Tq clock cycle = (BRP+1) / CAN clock

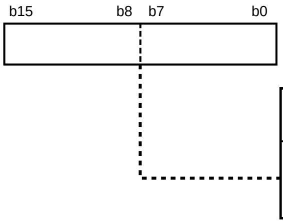
$$\text{Baud rate} = \frac{1}{\text{Tq clcok cycle x Tq per bit}}$$

Tq per bit = SS + PTS + PBS1 + PBS2

Tq: Time quantum	BRP: Setting value of the CiBPR register; 1-255
SS: Synchronization Segment; 1 Tq	PTS: Propagation Time Segment; 1 to 8 Tq
PBS1: Phase Buffer Segment 1; 2 to 8 Tq	PBS2: Phase Buffer Segment 2; 2 to 8 Tq

23.1.8 CANi Time Stamp Register (CiTSR Register) (i=0, 1)

CANi Time Stamp Register (i=0, 1)



Symbol	Address	After Reset[1]
C0TSR	0209_{16} - 0208_{16}	0000_{16}
C1TSR	0289_{16} - 0288_{16}	0000_{16}

Function	RW
Value of time stamp	RO

NOTES:

1. Value is obtained by setting the SLEEP bit in the CiSLPR register to "1" (sleep mode exited) after reset and supplying the clock to the CAN module.

Figure 23.10 C0TSR and C1TSR Registers

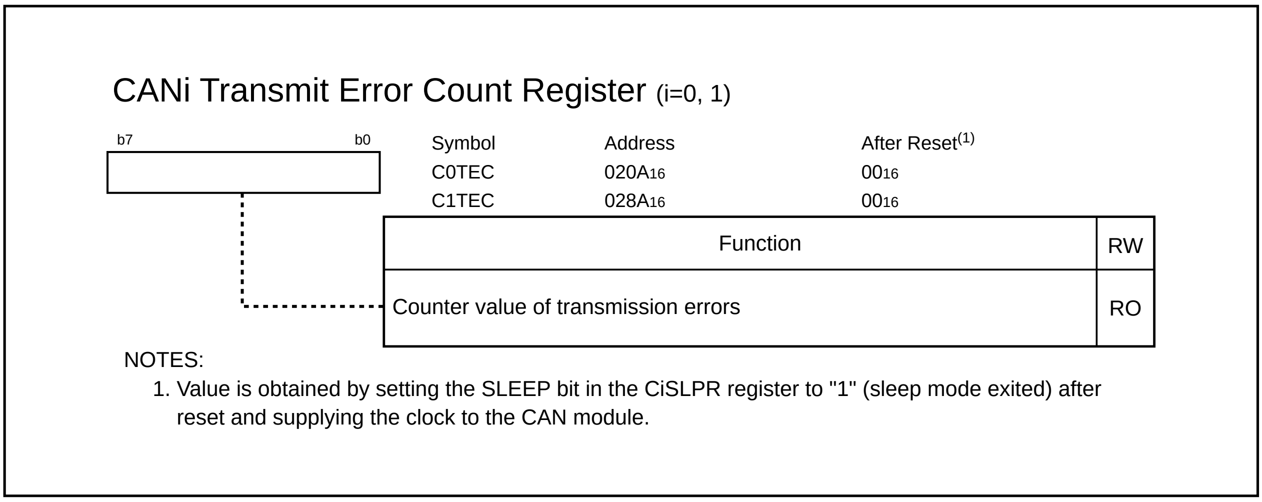
The CiTSR register is a 16-bit counter. The TSPRE1 and TSPRE0 bits in the CiCTLR0 register select the CAN bus bit clock divided by 1, 2, 3 or 4 as the count source for the CiTSR register.
When data transmission or reception is completed, the value of the CiTSR register is automatically stored into the message slot.
In loopback mode, when either data frame receive message slot or remote frame receive message slot is available to store the message, the value of the CiTSR register is also stored into the message slot when data reception is completed. The value of the CiTSR register is not stored when data transmission is completed.

The CiTSR register starts a counter increment when the RESET1 and RESET0 bits in the CiCTLR0 register are set to "0".
The CiTSR register is set to "0000_{16}":

- at the next count timing after the CiTSR register is set to "$FFFF_{16}$";
- when the RESET1 and RESET0 bits are set to "1" (CAN module reset) by program; or
- when the TSRESET bit is set to "1" (CiTSR register reset) by program.

$$\text{CAN bus bit clock} = \frac{1}{\text{CAN bit time}}$$

23.1.9 CANi Transmit Error Count Register (CiTEC Register) (i=0, 1)



CANi Transmit Error Count Register (i=0, 1)

Symbol	Address	After Reset[1]
C0TEC	020A16	0016
C1TEC	028A16	0016

Function	RW
Counter value of transmission errors	RO

NOTES:

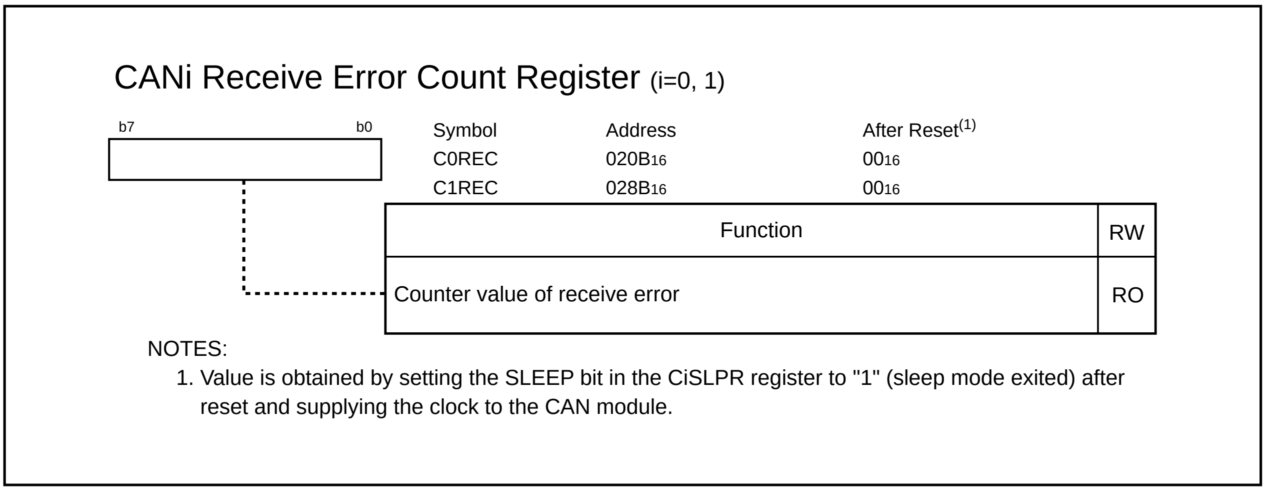
1. Value is obtained by setting the SLEEP bit in the CiSLPR register to "1" (sleep mode exited) after reset and supplying the clock to the CAN module.

Figure 23.11 C0TEC and C1TEC Registers

In an error active or an error passive state, the counting value of a transmission error is stored into the CiTEC register. The counter is decremented when the CAN module has transmitted data as expected or is incremented when an transmit error occurs.

In a bus-off state, an indeterminate value is stored into the CiTEC register. The CiTEC register is set to "0016" when the CAN module is placed in an error active state again.

23.1.10 CANi Receive Error Count Register (CiREC Register) (i=0, 1)



CANi Receive Error Count Register (i=0, 1)

Symbol	Address	After Reset[1]
C0REC	020B16	0016
C1REC	028B16	0016

Function	RW
Counter value of receive error	RO

NOTES:

1. Value is obtained by setting the SLEEP bit in the CiSLPR register to "1" (sleep mode exited) after reset and supplying the clock to the CAN module.

Figure 23.12 C0REC and C1REC Registers

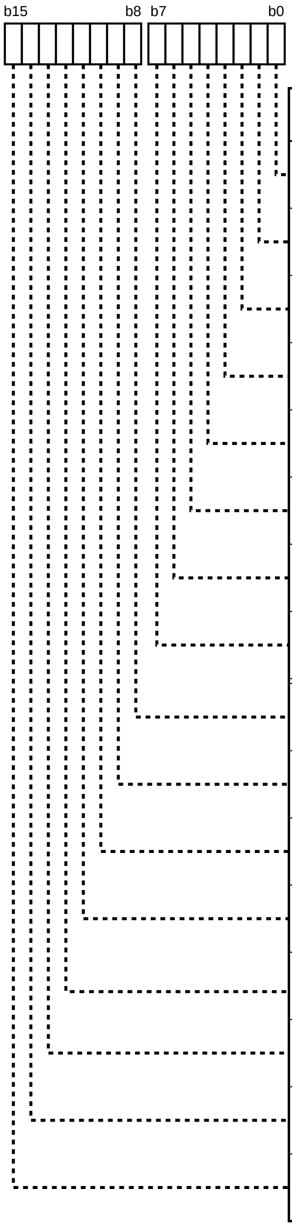
In an error active or an error passive state, a counting value of the reception error is stored into the CiREC register. The counter is decremented when the CAN module has received data as expected or it is incremented when a receive error occurs.

The CiREC register is set to 127 when the CiREC register is 128 (error passive state) or more and the CAN module has received as expected.

In a bus-off state, an indeterminate value is stored into the CiREC register. The CiREC register is set to "0016" when the CAN module is placed in an error active state again.

23.1.11 CANi Slot Interrupt Status Register (CiSISTR Register) (i=0, 1)

CANi Slot Interrupt Status Register (i=0, 1)



Symbol	Address	After Reset[1]
C0SISTR	$020D_{16}$ - $020C_{16}$	0000_{16}
C1SISTR	$028D_{16}$ - $028C_{16}$	0000_{16}

Bit Symbol	Bit Name	Function	RW
SIS15	Message Slot 15 Interrupt Request Status Bit	Determines whether an interrupt of a corresponding message slot is requested or not. 0: Requests no interrupt [2] 1: Requests an interrupt [2]	RW
SIS14	Message Slot 14 Interrupt Request Status Bit		RW
SIS13	Message Slot 13 Interrupt Request Status Bit		RW
SIS12	Message Slot 12 Interrupt Request Status Bit		RW
SIS11	Message Slot 11 Interrupt Request Status Bit		RW
SIS10	Message Slot 10 Interrupt Request Status Bit		RW
SIS9	Message Slot 9 Interrupt Request Status Bit		RW
SIS8	Message Slot 8 Interrupt Request Status Bit		RW
SIS7	Message Slot 7 Interrupt Request Status Bit		RW
SIS6	Message Slot 6 Interrupt Request Status Bit		RW
SIS5	Message Slot 5 Interrupt Request Status Bit		RW
SIS4	Message Slot 4 Interrupt Request Status Bit		RW
SIS3	Message Slot 3 Interrupt Request Status Bit		RW
SIS2	Message Slot 2 Interrupt Request Status Bit		RW
SIS1	Message Slot 1 Interrupt Request Status Bit		RW
SIS0	Message Slot 0 Interrupt Request Status Bit		RW

NOTES:

1. Value is obtained by setting the SLEEP bit in the CiSLPR register to "1" (sleep mode exited) after reset and supplying the clock to the CAN module.
2. Set to "0" by program. If it is set to "1", the value before setting to "1" remains.

Figure 23.13 C0SISTR and C1SISTR Registers

When using the CAN interrupt, the CiSISTR register (i=0, 1) indicates which message slot is requesting an interrupt. The SISj bits (j=0 to 15) are not automatically set to "0" (no interrupt requested) when an interrupt is acknowledged. Set the SISj bits to "0" by program.

Use the MOV instruction, instead of the bit clear instruction, to set the SISj bits to "0". The SISj bits, which are not being changed to "0", must be set to "1".

For example: To set the SIS0 bit to "0"

```
        Assembly language:      mov.w       #07FFFh, C0SISTR
        C language: c0sistr = 0x7FFF;
```

Refer to **23.4 CAN Interrupt** for details.

23.1.11.1 Message Slot for Transmission

The SISj bit is set to "1" (interrupt requested) when the CiTSR register is stored into the message slot j after data transmission is completed.

23.1.11.2 Message Slot for Reception

The SISj bit is set to "1" (interrupt requested) when the received message is stored in the message slot j after data reception is completed.

NOTES:

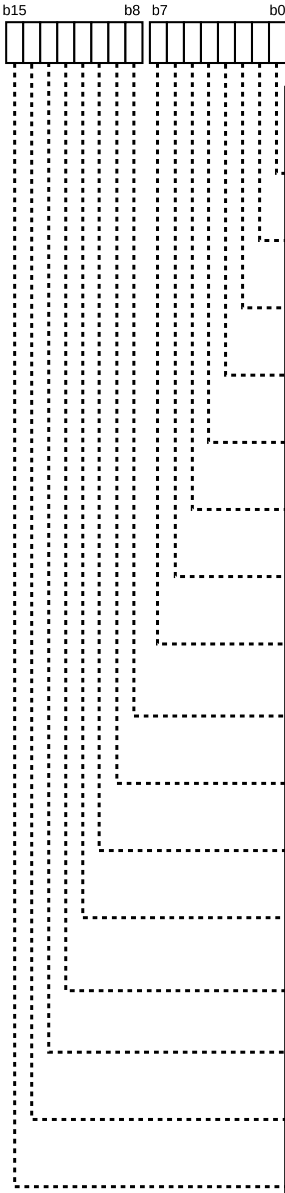
1.If the automatic answering function is enabled in the remote frame receive message slot, the SISj bit is set to "1" after the remote frame is received and the data frame is transmitted.

2.In the remote frame transmit message slot, the SISj bit is set to "1" after the remote frame is transmitted and the data frame is received.

3.The SISj bit is set to "1" if the SISj bit is set to "1" by an interrupt request and "0" by program simultaneously.

23.1.12 CANi Slot Interrupt Mask Register (CiSIMKR Register) (i=0, 1)

CANi Slot Interrupt Mask Register(1) (i=0, 1)



Symbol	Address	After Reset(2)
C0SIMKR	0211_{16} - 0210_{16}	0000_{16}
C1SIMKR	0291_{16} - 0290_{16}	0000_{16}

Bit Symbol	Bit Name	Function	RW
SIM15	Slot 15 Interrupt Request Mask Bit	Controls whether the interrupt request of the corresponding message slot is enabled or masked. 0: Masks (disables) an interrupt request 1: Enables an interrupt request	RW
SIM14	Slot 14 Interrupt Request Mask Bit		RW
SIM13	Slot 13 Interrupt Request Mask Bit		RW
SIM12	Slot 12 Interrupt Request Mask Bit		RW
SIM11	Slot 11 Interrupt Request Mask Bit		RW
SIM10	Slot 10 Interrupt Request Mask Bit		RW
SIM9	Slot 9 Interrupt Request Mask Bit		RW
SIM8	Slot 8 Interrupt Request Mask Bit		RW
SIM7	Slot 7 Interrupt Request Mask Bit		RW
SIM6	Slot 6 Interrupt Request Mask Bit		RW
SIM5	Slot 5 Interrupt Request Mask Bit		RW
SIM4	Slot 4 Interrupt Request Mask Bit		RW
SIM3	Slot 3 Interrupt Request Mask Bit		RW
SIM2	Slot 2 Interrupt Request Mask Bit		RW
SIM1	Slot 1 Interrupt Request Mask Bit		RW
SIM0	Slot 0 Interrupt Request Mask Bit		RW

NOTES:

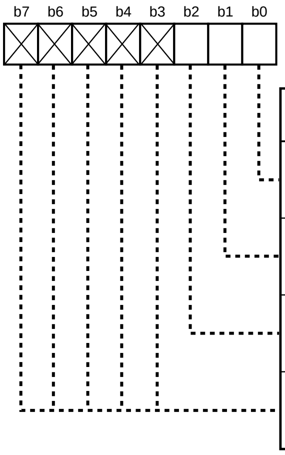
1. Change the CiSIMKR register setting while the CiMCTLj (j=0to 15) register, corresponding to the bit to be changed, is set to "00_{16}".
2. Value is obtained by setting the SLEEP bit in the CiSLPR register to "1" (sleep mode exited) after reset and supplying the clock to the CAN module.

Figure 23.14 C0SIMKR and C1SIMKR Registers

The CiSIMKR register determines whether an interrupt request, generated by a data transmission or reception in the corresponding message slot is enabled or disabled. When the SIMj bit (j=0 to 15) is set to "1" (no interrupt requested), an interrupt request generated by a data transmission or reception in the corresponding message slot is enabled. Refer to **23.4 CAN Interrupt** for details.

23.1.13 CANi Error Interrupt Mask Register (CiEIMKR Register) (i=0, 1)

CANi Error Interrupt Mask Register (i=0, 1)



Symbol	Address	After Reset[1]
C0EIMKR	0214_{16}	$XXXX X000_2$
C1EIMKR	0294_{16}	$XXXX X000_2$

Bit Symbol	Bit Name	Function	RW
BOIM	Bus-Off Interrupt Mask Bit	0: Masks (disables) an interrupt request 1: Enables an interrupt request	RW
EPIM	Error-Passive Interrupt Mask Bit	0: Masks (disables) an interrupt request 1: Enables an interrupt request	RW
BEIM	CAN Bus-Error Interrupt Mask Bit	0: Masks (disables) an interrupt request 1: Enables an interrupt request	RW
(b7 - b3)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		–

NOTES:

1. Value is obtained by setting the SLEEP bit in the CiSLPR register to "1" (sleep mode exited) after reset and supplying the clock to the CAN module.

Figure 23.15 C0EIMKR and C1EIMKR Registers

Refer to **23.4 CAN Interrupt** for details.

23.1.13.1 BOIM Bit

The BOIM bit determines whether an interrupt request is enabled or disabled when the CAN module is placed in a bus-off state. When the BOIM bit is set to "1", the bus-off interrupt request is enabled.

23.1.13.2 EPIM Bit

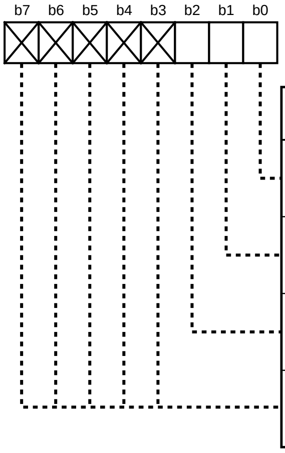
The EPIM bit determines whether an interrupt request is enabled or disabled when the CAN module is placed in an error passive state. When the EPIM bit is set to "1", the error passive interrupt request is enabled.

23.1.13.3 BEIM Bit

The BEIM bit determines whether an interrupt request is enabled or disabled when a CAN bus error occurs. When the BEIM bit is set to "1", the CAN bus error interrupt request is enabled.

23.1.14 CANi Error Interrupt Status Register (CiEISTR Register) (i=0, 1)

CANi Error Interrupt Status Register (i=0, 1)



Symbol	Address	After Reset[1]
C0EISTR	0215_{16}	$XXXX X000_2$
C1EISTR	0295_{16}	$XXXX X000_2$

Bit Symbol	Bit Name	Function	RW
BOIS	Bus-Off Interrupt Status Bit[2]	0: No interrupt is requested 1: Interrupt is requested	RW
EPIS	Error-Passive Interrupt Status Bit[2]	0: No interrupt is requested 1: Interrupt is requested	RW
BEIS	CAN Bus-Error Interrupt Status Bit[2]	0: No interrupt is requested 1: Interrupt is requested	RW
____ (b7 - b3)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		—

NOTES:

1. Value is obtained by setting the SLEEP bit in the CiSLPR register to "1" (sleep mode exited) after reset and supplying the clock to the CAN module.
2. Set to "0" by program. When it is set to "1", the value before setting to "1" remains.

Figure 23.16 C0EISTR and C1EISTR Registers

When using the CAN interrupt, the CiEISTR register indicates the source of the generated error interrupt. The BOIS, EPIS and BEIS bits are not automatically set to "0" (no interrupt requested) even if an interrupt is acknowledged. Set these bits to "0" by program.

Use the MOV instruction, instead of the bit clear instruction, to set each bit in the CiEISTR register to "0".

Bits not being changed to "0" must be set to "1".

For example: To set the BOIS bit for CAN0 to "0"

```
Assembly language:      mov.b#006h, C0EISTR
C language:             c0eistr = 0x06;
```

Refer to **23.4 CAN Interrupt** for details.

23.1.14.1 BOIS Bit

The BOIS bit is set to "1" when the CAN module is placed in a bus-off state.

23.1.14.2 EPIS Bit

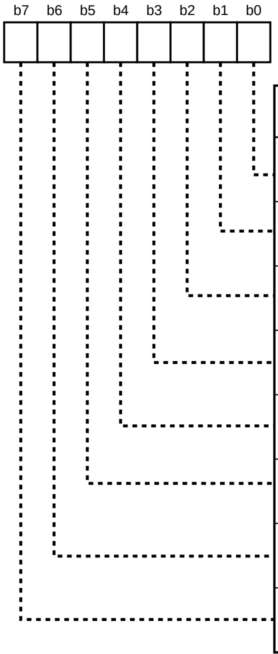
The EPIS bit is set to "1" when the CAN module is placed in an error passive state.

23.1.14.3 BEIS Bit

The BEIS bit is set to "1" when a CAN bus error is detected.

23.1.15 CANi Error Factor Register (CiEFR Register) (i=0, 1)

CANi Error Factor Register (i=0, 1)



Symbol	Address	After Reset[1]
C0EFR	0216_{16}	00_{16}
C1EFR	0296_{16}	00_{16}

Bit Symbol	Bit Name	Function	RW
ACKE	ACK Error Detect Bit[2]	0: Detects no ACK error 1: Detects an ACK error	RW
CRCE	CRC Error Detect Bit[2]	0: Detects no CRC error 1: Detects a CRC error	RW
FORME	FORM Error Detect Bit[2]	0: Detects no form error 1: Detects a form error	RW
STFE	Stuff Error Detect Bit[2]	0: Detects no stuff error 1: Detects a stuff error	RW
BITE0	Bit Error Detect Bit 0[2]	0: Detects no bit error while transmitting "H" 1: Detects a bit error while transmitting "H"	RW
BITE1	Bit Error Detect Bit 1[2]	0: Detects no bit error while transmitting "L" 1: Detects a bit error while transmitting "L"	RW
RCVE	Receive Error Detect Bit[2]	0: Detects no error while receiving data 1: Detects an error while receiving data	RW
TRE	Transmit Error Detect Bit[2]	0: Detects no error while transmitting data 1: Detects an error while transmitting data	RW

NOTES:
1. Value is obtained by setting the SLEEP bit in the CiSLPR register to "1" (sleep mode exited) after reset and supplying the clock to the CAN module.
2. Set to "0" by program. If it is set to "1", the value before setting to "1" remains.

Figure 23.17 C0EFR and C1EFR Registers

The CiEFR register indicates the cause of error when a communication error is detected. Set the following bits to "0" by program because they are not changed "1" to "0" automatically.

Use the MOV instruction, instead of the bit clear instruction, to set each bit in the CiEFR register to "0".
Bits not being changed to "0" must be set to "1".
For example: To set the ACKE bit for CAN0 to "0"

```
Assembly language:      mov.b#0FEh, C0EFR
C language:             c0efr = 0xFE;
```

23.1.15.1 ACKE Bit

The ACKE bit is set to "1" when an ACK error is detected.

23.1.15.2 CRCE Bit

The CRC bit is set to "1" when a CRC error is detected.

23.1.15.3 FORME Bit

The FORME bit is set to "1" when a form error is detected.

23.1.15.4 STFE Bit

The STFE bit is set to "1" when a stuff error is detected.

23.1.15.5 BITE0 Bit

The BITE0 bit is set to "1" when a bit error is detected while transmitting recessive "H".

23.1.15.6 BITE1 Bit

The BITE1 bit is set to "1" when a bit error is detected while transmitting dominant "L".

23.1.15.7 RCVE Bit

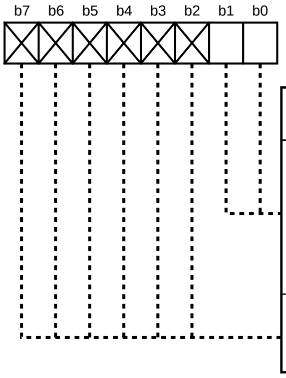
The RCVE bit is set to "1" when an error is detected while receiving data.

23.1.15.8 TRE Bit

The TRE bit is set to "1" when an error is detected while transmitting data.

23.1.16 CANi Mode Register (CiMDR Register) (i=0, 1)

CANi Mode Register (i=0, 1)[1]



Symbol	Address	After Reset[2]
C0MDR	0219_{16}	XXXX $XX00_2$
C1MDR	0299_{16}	XXXX $XX00_2$

Bit Symbol	Bit Name	Function	RW
CMOD	CAN Operating Mode Select Bit	b1 b0 0 0: Normal operating mode 0 1: Bus monitoring mode 1 0: Self-test mode 1 1: Do not set to this value	RW RW
____ (b7 - b2)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		–

NOTES:

1. Set the CiMDR register when the STATE_RESET bit in the CiSTR register is set to "1" (CAN module reset completed).
2. Value is obtained by setting the SLEEP bit in the CiSLPR register to "1" (sleep mode exited) after reset and supplying the clock to the CAN module.

Figure 23.18 C0MDR and C1MDR Registers

23.1.16.1 CMOD Bit

The CMOD bit selects a CAN operating mode.

- Normal operating mode: The CAN module transmits and receives data as expected.
- Bus monitoring mode[1]: The CAN module receives data. Output signal from the CANiOUT pin is fixed as a high-level ("H") signal in bus monitoring mode. The CAN mod ule transmits neither ACK nor error frame.
- Self-test mode: The CAN module connects the CANiOUT pin to the CANiIN pin internally. The CAN module can communicate without additional device in loop back mode. Output signal from the CANiOUT pin is fixed as an "H" signal in self-test mode while transmitting data. Figure 23.19 shows an image diagram in self-test mode.

NOTES:

1. Do not generate a transmit request in bus monitoring mode.
   The CAN module assumes the ACK bit is set to dominant "L" regardless of the ACK bit setting. Therefore, when the CRC delimiter is received as expected, the CAN module determines the data is received with no error regardless of the ACK bit setting.

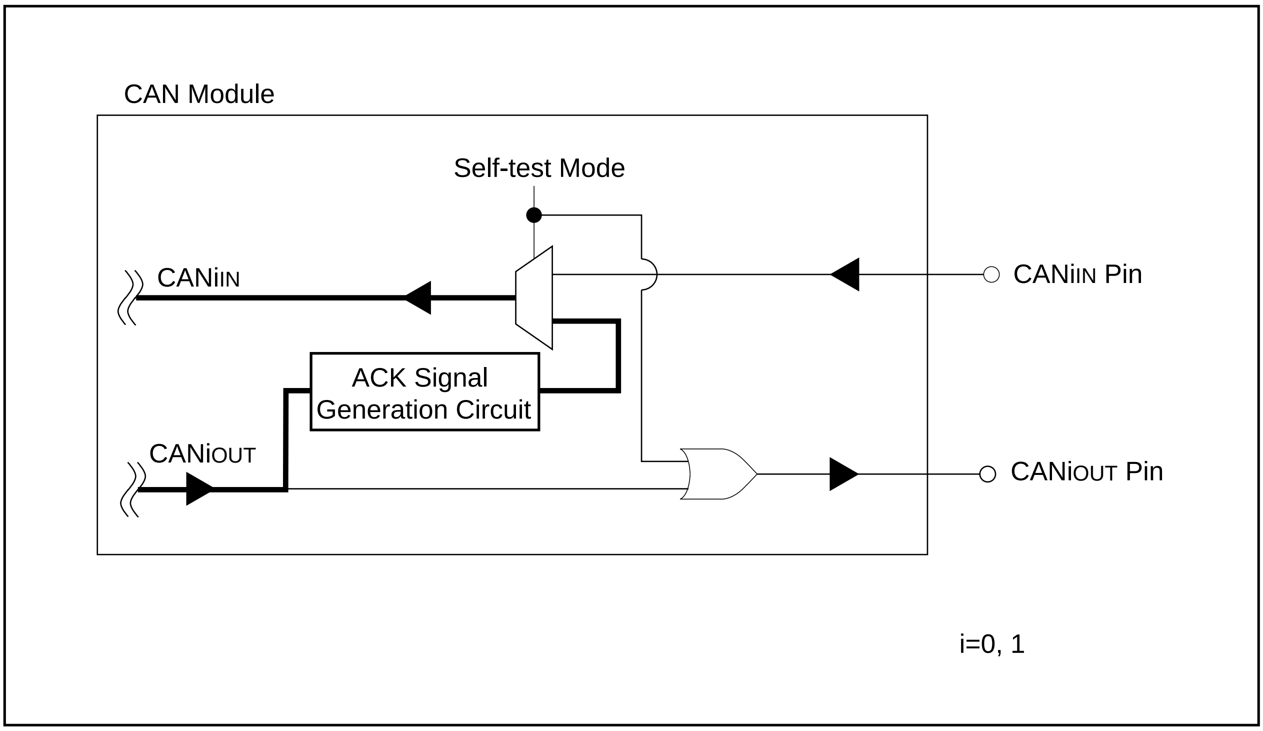
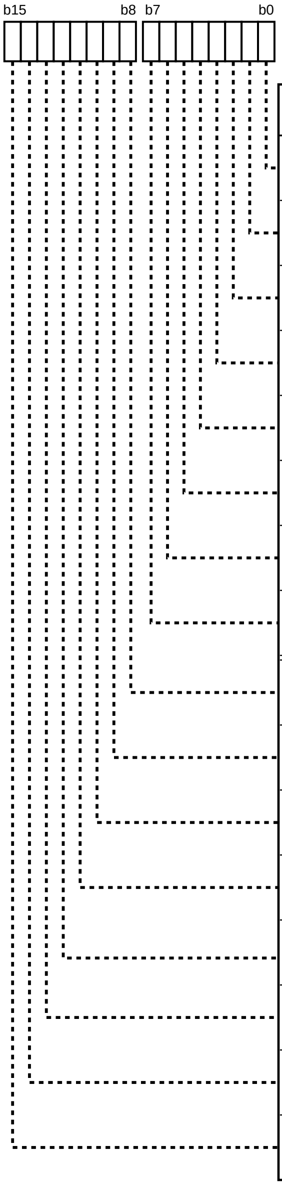


Figure 23.19 Self-Test Mode

23.1.17 CANi Single-Shot Control Register (CiSSCTLR Register) (i=0, 1)

CANi Single-Shot Control Register (i=0, 1)(1, 2)



Symbol	Address	After Reset(3)
C0SSCTLR	0221_{16} - 0220_{16}	0000_{16}
C1SSCTLR	$02A1_{16}$ - $02A0_{16}$	0000_{16}

Bit Symbol	Bit Name	Function	RW
SSC15	Message Slot 15 Single-Shot Control Bit	0: Single-shot mode not used 1: Use single-shot mode	RW
SSC14	Message Slot 14 Single-Shot Control Bit		RW
SSC13	Message Slot 13 Single-Shot Control Bit		RW
SSC12	Message Slot 12 Single-Shot Control Bit		RW
SSC11	Message Slot 11 Single-Shot Control Bit		RW
SSC10	Message Slot 10 Single-Shot Control Bit		RW
SSC9	Message Slot 9 Single-Shot Control Bit		RW
SSC8	Message Slot 8 Single-Shot Control Bit		RW
SSC7	Message Slot 7 Single-Shot Control Bit		RW
SSC6	Message Slot 6 Single-Shot Control Bit		RW
SSC5	Message Slot 5 Single-Shot Control Bit		RW
SSC4	Message Slot 4 Single-Shot Control Bit		RW
SSC3	Message Slot 3 Single-Shot Control Bit		RW
SSC2	Message Slot 2 Single-Shot Control Bit		RW
SSC1	Message Slot 1 Single-Shot Control Bit		RW
SSC0	Message Slot 0 Single-Shot Control Bit		RW

NOTES:

1. Set the CiSSCTLR register after the CiMCTLj register (j=0 to 15) in a slot, corresponding to the bit to be changed, is set to "00_{16}".
2. The CiSSCTLR register can be accessed only when the BANKSEL bit in the CiCTLR1 register is set to "0" (message slot control register and single-shot register selected).
3. Value is obtained by setting the SLEEP bit in the CiSLPR register to "1" (sleep mode exited) after reset, supplying the clock to the CAN module, and setting the BANKSEL bit to "0".

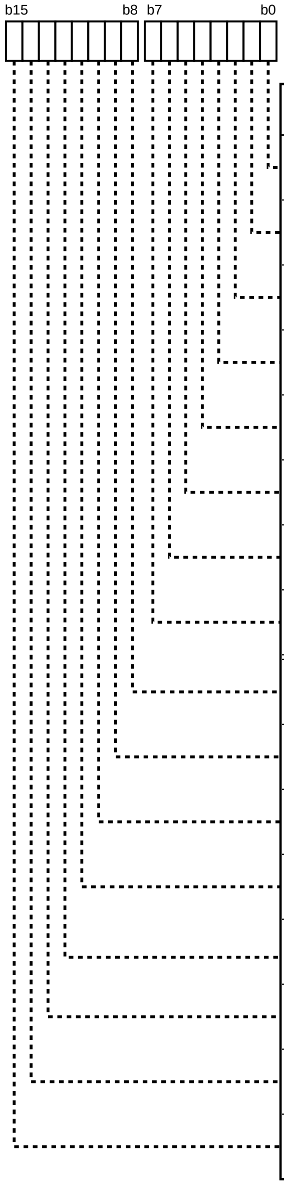
Figure 23.20 C0SSCTLR and C1SSCTLR Registers

According to the CAN Specification 2.0 Part B, if the arbitration lost or transmission error causes a transmit failure, the microcomputer continues transmitting data until the transmission is completed. The CiSSCTLR register determines whether or not, and from which slot, data is re-transmitted.

In single-shot mode, if the arbitration lost or transmission error causes a transmission failure, data is not transmitted again. When the SSCj bit (j=0 to 15) is set to "1", the corresponding message slot j is in single-shot mode.

23.1.18 CANi Single-Shot Status Register (CiSSSTR Register) (i=0, 1)

CANi Single-Shot Status Register (i=0, 1)[1]



Symbol	Address	After Reset[2]
C0SSSTR	0225_{16} - 0224_{16}	0000_{16}
C1SSSTR	$02A5_{16}$ - $02A4_{16}$	0000_{16}

Bit Symbol	Bit Name	Function	RW
SSS15	Message Slot 15 Single-Shot Status Bit	0: No arbitration is lost, or no transmit error occurs 1: Arbitration is lost, or transmit error occurs (Note 3)	RW
SSS14	Message Slot 14 Single-Shot Status Bit		RW
SSS13	Message Slot 13 Single-Shot Status Bit		RW
SSS12	Message Slot 12 Single-Shot Status Bit		RW
SSS11	Message Slot 11 Single-Shot Status Bit		RW
SSS10	Message Slot 10 Single-Shot Status Bit		RW
SSS9	Message Slot 9 Single-Shot Status Bit		RW
SSS8	Message Slot 8 Single-Shot Status Bit		RW
SSS7	Message Slot 7 Single-Shot Status Bit		RW
SSS6	Message Slot 6 Single-Shot Status Bit		RW
SSS5	Message Slot 5 Single-Shot Status Bit		RW
SSS4	Message Slot 4 Single-Shot Status Bit		RW
SSS3	Message Slot 3 Single-Shot Status Bit		RW
SSS2	Message Slot 2 Single-Shot Status Bit		RW
SSS1	Message Slot 1 Single-Shot Status Bit		RW
SSS0	Message Slot 0 Single-Shot Status Bit		RW

NOTES:

1. The CiSSSTR register can be accessed only when the BANKSEL bit in the CiCTLR1 is set to "0" (message slot control register and single-shot register selected).
2. Value is obtained by setting the SLEEP bit in the CiSLPR register to "1" (sleep mode exited) after reset and supplying the clock to the CAN module.
3. Set to "0" by program. When it is set it to "1", the value before setting to "1" remains.

Figure 23.21 C0SSSTR and C1SSSTR Registers

If the arbitration lost or transmission error causes a transmission failure, the bit corresponding to message slot j (j=0 to 15) is set to "1". The SSSj bit is set to "0" by program because it is not set to "0" automatically.

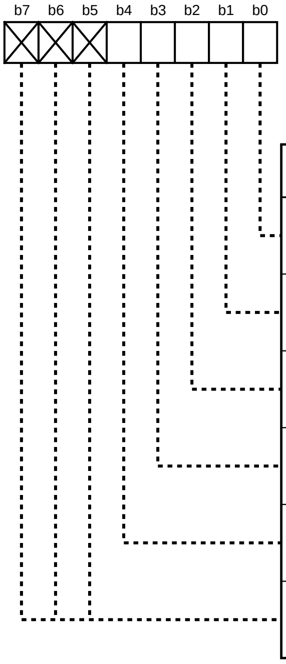
Use the MOV instruction, instead of the bit clear instruction, to set the SSSj bit to "0". Bits not being changed to "0" must be set to "1".

For example: To set the SSS0 bit for CAN0 to "0"

```
Assembly language:    mov.w #07FFFh, C0SSSTR
C language:           c0ssstr = 0x7FFF;
```

23.1.19 CANi Global Mask Register, CANi Local Mask Register A and CANi Local Mask Register B (CiGMRk, CiLMARk and CiLMBRk Registers) (i=0,1, k=0 to 4)

CANi Global Mask Register Standard ID0[1]
CANi Local Mask Register A Standard ID0[1]
CANi Local Mask Register B Standard ID0[1] (i=0, 1)



Symbol	Address	After Reset[2]
C0GMR0, C1GMR0	0228_{16}, $02A8_{16}$	XXX0 0000_2
C0LMAR0, C1LMAR0	0230_{16}[3], $02B0_{16}$[4]	XXX0 0000_2
C0LMBR0, C1LMBR0	0238_{16}[5], $02B8_{16}$[6]	XXX0 0000_2

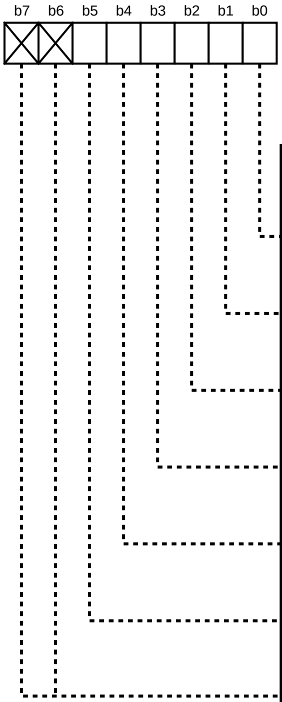
Bit Symbol	Bit Name	Function	RW
SID6M	Standard ID6	0: No ID is verified 1: ID is verified	RW
SID7M	Standard ID7		RW
SID8M	Standard ID8		RW
SID9M	Standard ID9		RW
SID10M	Standard ID10		RW
____ (b7 - b5)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		—

NOTES:

1. The CiGMR0, CiLMAR0 and CiLMBR0 registers can be accessed only when the BANKSEL bit in the CiCTLR1 register is set to "1" (mask register selected).
2. Value is obtained by setting the SLEEP bit in the CiSLPR register to "1" (sleep mode exited) after reset, supplying the clock to the CAN module, and setting the BANKSEL bit to "1".
3. The C0LMAR0 register shares the same address with the C0MCTL0 register.
4. The C1LMAR0 register shares the same address with the C1MCTL0 register.
5. The C0LMBR0 register shares the same address with the C0MCTL8 register.
6. The C1LMBR0 register shares the same address with the C1MCTL8 register.

Figure 23.22 C0GMR0, C0LMAR0 and C0LMBR0 Registers C1GMR0, C1LMAR0 and C1LMBR0 Registers

CANi Global Mask Register Standard ID1[1]
CANi Local Mask Register A Standard ID1[1]
CANi Local Mask Register B Standard ID1[1] (i=0, 1)



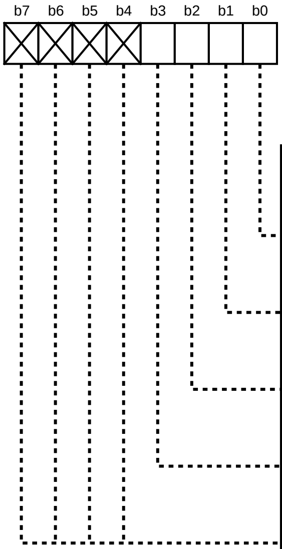
Symbol	Address	After Reset[2]
C0GMR1, C1GMR1	0229_{16}, $02A9_{16}$	$XX00\ 0000_2$
C0LMAR1, C1LMAR1	0231_{16}[3], $02B1_{16}$[4]	$XX00\ 0000_2$
C0LMBR1, C1LMBR1	0239_{16}[5], $02B9_{16}$[6]	$XX00\ 0000_2$

Bit Symbol	Bit Name	Function	RW
SID0M	Standard ID0	0: No ID is verified 1: ID is verified	RW
SID1M	Standard ID1		RW
SID2M	Standard ID2		RW
SID3M	Standard ID3		RW
SID4M	Standard ID4		RW
SID5M	Standard ID5		RW
____ (b7 - b6)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		–

NOTES:
1. The CiGMR0, CiLMAR0 and CiLMBR0 registers can be accessed only when the BANKSEL bit in the CiCTLR1 register is set to "1" (mask register selected).
2. Value is obtained by setting the SLEEP bit in the CiSLPR register to "1" (sleep mode exited) after reset, supplying the clock to the CAN module, and setting the BANKSEL bit to "0".
3. The C0LMAR1 register shares the same address with the C0MCTL1 register.
4. The C1LMAR1 register shares the same address with the C1MCTL1 register.
5. The C0LMBR1 register shares the same address with the C0MCTL9 register.
6. The C1LMBR1 register shares the same address with the C1MCTL9 register.

Figure 23.23 C0GMR1, C0LMAR1 and C0LMBR1 Registers C1GMR1, C1LMAR1 and C1LMBR1 Registers

CANi Global Mask Register Extended ID0[1]
CANi Local Mask Register A Extended ID0[1]
CANi Local Mask Register B Extended ID0[1] (i=0, 1)



Symbol	Address	After Reset[2]
C0GMR2, C1GMR2	$022A_{16}$, $02AA_{16}$	XXXX 0000_2
C0LMAR2, C1LMAR2	0232_{16}[3], $02B2_{16}$[4]	XXXX 0000_2
C0LMBR2, C1LMBR2	$023A_{16}$[5], $02BA_{16}$[6]	XXXX 0000_2

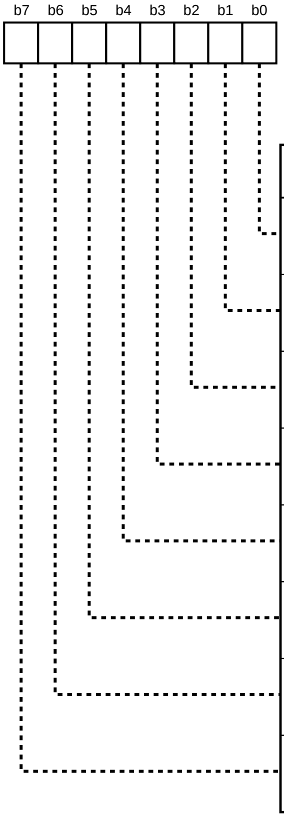
Bit Symbol	Bit Name	Function	RW
EID14M	Extended ID14	0: No ID is verified 1: ID is verified	RW
EID15M	Extended ID15		RW
EID16M	Extended ID16		RW
EID17M	Extended ID17		RW
____ (b7 - b4)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		–

NOTES:

1. The CiGMR2, CiLMAR2 and CiLMBR2 registers can be accessed only when the BANKSEL bit in the CiCTLR1 register is set to "1" (mask register selected).
2. Value is obtained by setting the SLEEP bit in the CiSLPR register to "1" (sleep mode exited) after reset, supplying the clock to the CAN module, and setting the BANKSEL bit to "0".
3. The C0LMAR2 register shares the same address with the C0MCTL2 register.
4. The C1LMAR2 register shares the same address with the C1MCTL2 register.
5. The C0LMBR2 register shares the same address with the C0MCTL10 register.
6. The C1LMBR2 register shares the same address with the C1MCTL10 register.

Figure 23.24 C0GMR2, C0LMAR2 and C0LMBR2 Registers
C1GMR2, C1LMAR2 and C1LMBR2 Registers

CANi Global Mask Register Extended ID1[1]
CANi Local Mask Register A Extended ID1[1]
CANi Local Mask Register B Extended ID1[1] (i=0,1)



Symbol	Address	After Reset[2]
C0GMR3, C1GMR3	022B$_{16}$, 02AB$_{16}$	00$_{16}$
C0LMAR3, C1LMAR3	0233$_{16}$[3], 02B3$_{16}$[4]	00$_{16}$
C0LMBR3, C1LMBR3	023B$_{16}$[5], 02BB$_{16}$[6]	00$_{16}$

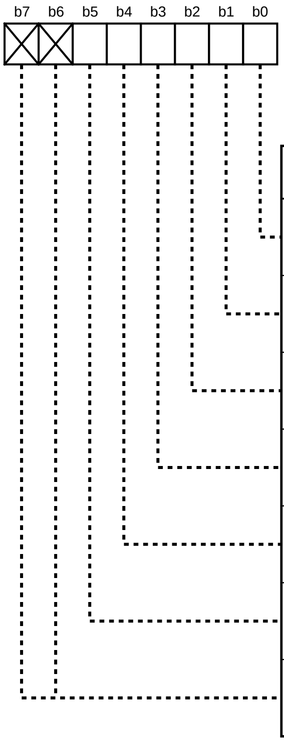
Bit Symbol	Bit Name	Function	RW
EID6M	Extended ID6	0: No ID is verified 1: ID is verified	RW
EID7M	Extended ID7		RW
EID8M	Extended ID8		RW
EID9M	Extended ID9		RW
EID10M	Extended ID10		RW
EID11M	Extended ID11		RW
EID12M	Extended ID12		RW
EID13M	Extended ID13		RW

NOTES:

1. The CiGMR3, CiLMAR3 and CiLMBR3 registers can be accessed only when the BANKSEL bit in the CiCTLR1 register is set to "1" (mask register selected).
2. Value is obtained by setting the SLEEP bit in the CiSLPR register to "1" (sleep mode exited) after reset, supplying the clock to the CAN module, and setting the BANKSEL bit to "0".
3. The C0LMAR3 register shares the same address with the C0MCTL3 register.
4. The C1LMAR3 register shares the same address with the C1MCTL3 register.
5. The C0LMBR3 register shares the same address with the C0MCTL11 register.
6. The C1LMBR3 register shares the same address with the C1MCTL11 register.

Figure 23.25 C0GMR3, C0LMAR3 and C0LMBR3 Registers
C1GMR3, C1LMAR3 and C1LMBR3 Registers

CANi Global Mask Register Extended ID2[1]
CANi Local Mask Register A Extended ID2[1]
CANi Local Mask Register B Extended ID2[1] (i=0, 1)



Symbol	Address	After Reset[2]
C0GMR4, C1GMR4	$022C_{16}$, $02AC_{16}$	$XX00\ 0000_2$
C0LMAR4, C1LMAR4	0234_{16}[3], $02B4_{16}$[4]	$XX00\ 0000_2$
C0LMBR4, C1LMBR4	$023C_{16}$[5], $02BC_{16}$[6]	$XX00\ 0000_2$

Bit Symbol	Bit Name	Function	RW
EID0M	Extended ID0	0: No ID is verified 1: ID is verified	RW
EID1M	Extended ID1		RW
EID2M	Extended ID2		RW
EID3M	Extended ID3		RW
EID4M	Extended ID4		RW
EID5M	Extended ID5		RW
____ (b7 - b6)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		—

NOTES:
1. The CiGMR4, CiLMAR4 and CiLMBR4 registers can be accessed only when the BANKSEL bit in the CiCTLR1 register is set to "1" (mask register selected).
2. Value is obtained by setting the SLEEP bit in the CiSLPR register to "1" (sleep mode exited) after reset, supplying the clock to the CAN module, and setting the BANKSEL bit to "0".
3. The C0LMAR4 register shares the same address with the C0MCTL4 register.
4. The C1LMAR4 register shares the same address with the C1MCTL4 register.
5. The C0LMBR4 register shares the same address with the C0MCTL12 register.
6. The C1LMBR4 register shares the same address with the C1MCTL12 register.

**Figure 23.26 C0GMR4, C0LMAR4 and C0LMBR4 Registers
C1GMR4, C1LMAR4 and C1LMBR4 Registers**

The CiGMRk, CiLMARk and CiLMBRk registers are used for acceptance filtering. The users can select and receive user-desired messages.
The CiGMRk register determines whether IDs in the message slots 0 to 13 are verified. The CiLMARk register determines whether ID in the message slot 14 is verified. The CiLMBRk register determines whether ID in the message slot 15 is verified.

- When bits in these registers are set to "0", each standard ID0 and standard ID1 bits (ID bit) and extended ID0 to extended ID2 bits in the CANi message slots j (j=0 to 15) corresponding to the bits in the above registers, is masked while acceptance filtering. (The corresponding bits are assumed to have matching IDs.)
- When bits in these registers are set to "1", corresponding ID bits are compared with received IDs while acceptance filtering. If the received ID matches the ID in the message slot j, the received data having the matched ID is stored into that message slot.

NOTES:

1. Change the CiGMRk register setting only when the message slots 0 to 13 have no receive request.
2. Change the CiLMARk register setting only when the message slot 14 has no receive request.
3. Change the CiLMBRk register setting only when the message slot 15 has no receive request.
4. More than two message slots are able to store a receive message ID, the ID is stored into the message slot, having the smallest slot number.

Figure 23.27 shows each mask register and corresponding message slot. Figure 23.28 shows the acceptance filtering.

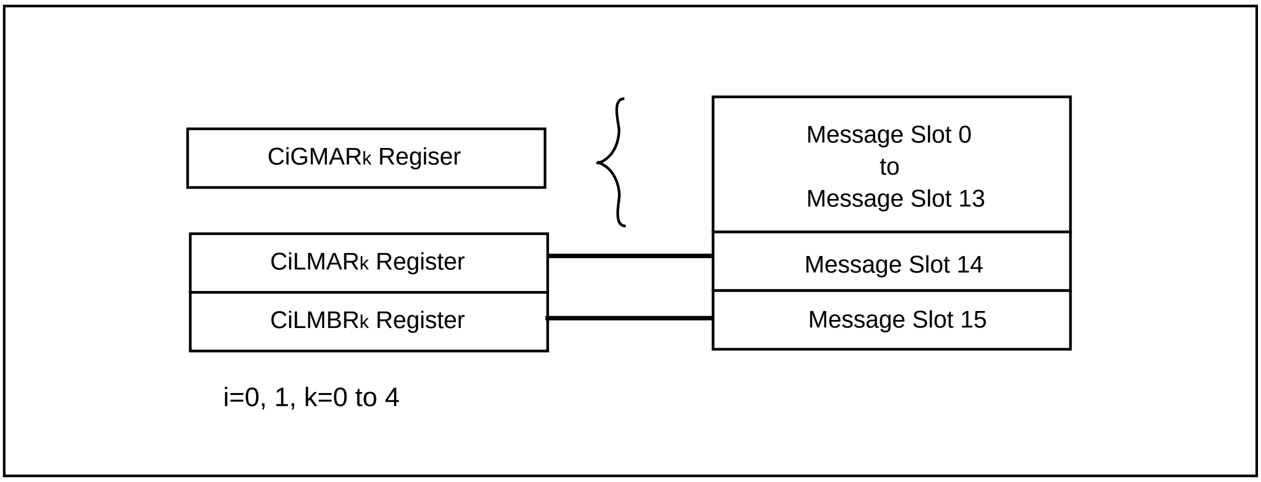


Figure 23.27 Mask Registers and Message Slots

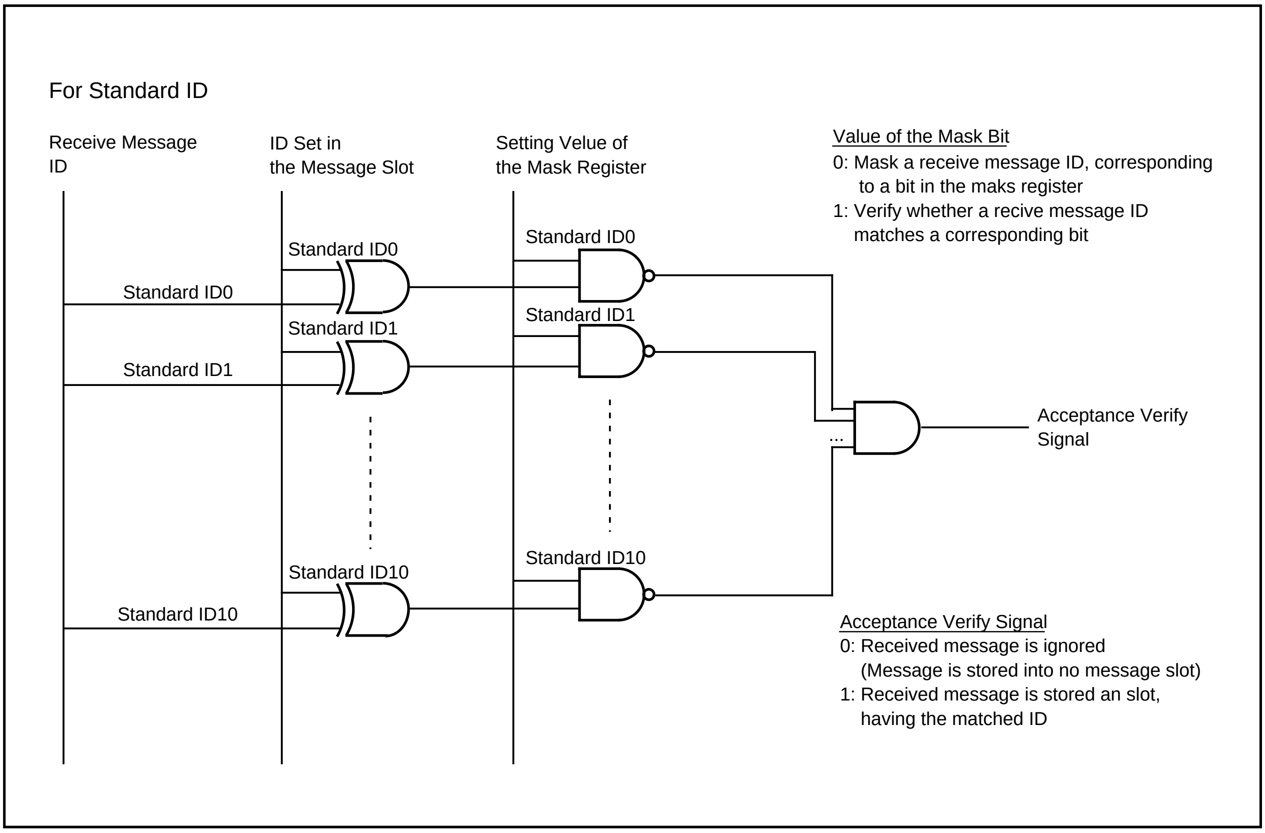
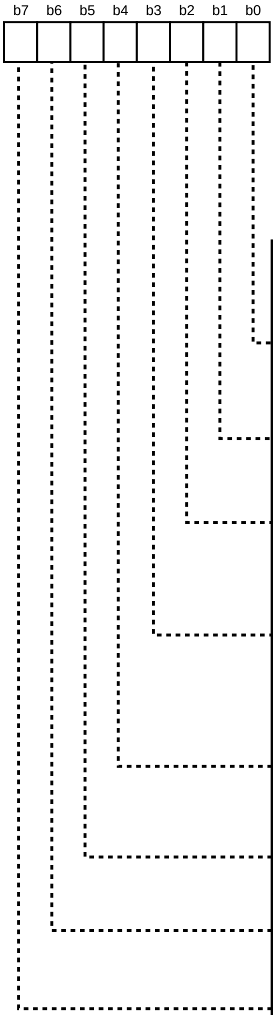

For Standard ID
Receive Message ID
ID Set in the Message Slot
Setting Velue of the Mask Register
Value of the Mask Bit
0: Mask a receive message ID, corresponding to a bit in the maks register
1: Verify whether a recive message ID matches a corresponding bit
Standard ID0
Standard ID1
Standard ID10
Standard ID0
Standard ID1
Standard ID10
Standard ID0
Standard ID1
Standard ID10
...
Acceptance Verify Signal
Acceptance Verify Signal
0: Received message is ignored (Message is stored into no message slot)
1: Received message is stored an slot, having the matched ID

Figure 23.28 Acceptance Filtering

23.1.20 CANi Message Slot j Control Register (CiMCTLj Register) (i=0,1, j=0 to 15)

CANi Message Slot j Control Register (i=0,1, j=0 to 15)[1]



Symbol	Address	After Reset[2]
C0MCTL0 to C0MCTL3	0230_{16}[3], 0231_{16}[3], 0232_{16}[3], 0233_{16}[3]	00_{16}
C0MCTL4 to C0MCTL7	0234_{16}[3], 0235_{16}, 0236_{16}, 0237_{16}	00_{16}
C0MCTL8 to C0MCTL11	0238_{16}[4], 0239_{16}[4], $023A_{16}$[4], $023B_{16}$[4]	00_{16}
C0MCTL12 to C0MCTL15	$023C_{16}$[4], $023D_{16}$, $023E_{16}$, $023F_{16}$	00_{16}
C1MCTL0 to C1MCTL3	$02B0_{16}$[5], $02B1_{16}$[5], $02B2_{16}$[5], $02B3_{16}$[5]	00_{16}
C1MCTL4 to C1MCTL7	$02B4_{16}$[5], $02B5_{16}$, $02B6_{16}$, $02B7_{16}$	00_{16}
C1MCTL8 to C1MCTL11	$02B8_{16}$[6], $02B9_{16}$[6], $02BA_{16}$[6], $02BB_{16}$[6]	00_{16}
C1MCTL12 to C1MCTL15	$02BC_{16}$[6], $02BD_{16}$, $02BE_{16}$, $02BF_{16}$	00_{16}

Bit Symbol	Bit Name	Function		RW
When receive, NEWDATA When transmit, SENTDATA	Receive Complete Flag Transmit Complete Flag	When transmitting 0: Not transmitted[4] 1: Transmit complete	When receiving 0: Not received[4] 1: Receive complete	RW
When receive, INVALDATA When transmit, TRMACTIVE	Receiving Flag Transmitting Flag	When transmitting 0: Except transmitting 1: Transmittting	When receiving 0: Except storing received data 1: Stores received data	RO
MSGLOST	Overwrite Flag[7]	0: No overrun error occurs 1: Overrun error occurs		RW
REMACTIVE	Remote Frame Transmit/Receive Status Flag	In modes other than BasicCan mode 0: Data frame 1: Remote frame In BasicCan mode 0: Receives the data frame (status) 1: Receives the remote frame (status)		RO
RSPLOCK	Automatic Answering Disable Mode Select Bit	0: Enables automatic answering of the remote frame 1: Disables automatic answering of the remote frame		RW
REMOTE	Remote Frame Set Bit	0: Transmits/receives the data frame 1: Transmits/receives the remote frame		RW
RECREQ	Receive Request Bit	0: No request to receive the frame 1: Request to receive the frame		RW
TRMREQ	Transmit Request Bit	0: No request to transmit the frame 1: Request to transmit the frame		RW

NOTES:

1. The CiMCTLj register can be accessed only when the BANKSEL bit in the CiCTLR1 register is set to "0" (message slot control register and single-shot register selected).
2. Value is obtained by setting the SLEEP bit in the CiSLPR register to "1" (sleep mode exited) after reset, supplying the clock to the CAN module, and setting the BANKSEL bit to "0".
3. The C0MCTL0 to C0MCTL4 registers each share addresses with the C0LMAR0 to C0LMAR4 registers.
4. The C0MCTL8 to C0MCTL12 registers each share addresses with the C0LMBR0 to C0LMBR4 registers.
5. The C1MCTL0 to C1MCTL4 registers each share addresses with the C1LMAR0 to C1LMAR4 registers.
6. The C1MCTL8 to C1MCTL12 registers each share addresses with the C1LMBR0 to C1LMBR4 registers.
7. Set to "0" by program. If it is set to "1", the value before setting to "1" remains.

Figure 23.29 C0MCTL0 to C0MCTL15 Registers and C1MCTL0 to C1MCTL15 Registers

Table 23.4 CiMCTLj register(i=0,1, j= 0 to 15) Settings and Transmit/Receive Mode

Settings for the CiMCTLj Register								Transmit/Receive Mode
TRMREQ	RECREQ	REMOTE	RSPLOCK	REMACTIVE	MSGLOST	TRMACTIVE INVALDATA	SENTDATA NEWDATA	
0	0	0	0	0	0	0	0	No frame is transmitted or received
0	1	0	0	0	0	0	0	Data frame is received
0	1	1	1 or 0	0	0	0	0	Remote frame is received (The data frame is transmitted after receiving the remote frame.)
1	0	0	0	0	0	0	0	Data frame is transmitted
1	0	1	0	0	0	0	0	Remote frame is transmitted (The data frame is received after transmitting the remote frame)

23.1.20.1 SENTDATA/NEWDATA Bit

The SENTDATA/NEWDATA bit indicates that the CAN module has transmitted or received the CAN message. Set the SENTDATA/NEWDATA bit to "0 " (not transmitted or not received) by program before data transmission and reception is started. The SENTDATA/NEWDATA bit is not set to "0" automatically. When the TRMACTIVE/INVALDATA bit is set to "1" (during transmission or storing received data), the SENTDATA/NEWDATA bit cannot be set to "0".

SENTDATA : The SENTDATA bit is set to "1" (transmit complete) when data transmission is completed in the transmit message slot.

NEWDATA : The NEWDATA bit is set to "1" (receive complete) when the message to be stored into the message slot j (j=0 to 15) is received in the receive message slot as expected.

NOTES:

1. To read a received data from the message slot j, set the NEWDATA bit to "0" before reading. If the NEWDATA bit is set to "1" immediately after reading, this indicates that new received data has been stored into the message slot while reading and the read data contains an indeterminate value. In this case, discard the data with indeterminate value and then read the message slot again after the NEWDATA bit is set to "0".
2. When the remote frame is transmitted or received, the SENTDATA/NEWDATA bit remains unchanged after the remote frame transmission or reception is completed. The SENTDATA/NEWDATA bit is set to "1" when a subsequent data frame transmission or reception is completed.

23.1.20.2 TRMACTIVE/INVALDATA Bit

The TRMACTIVE/INVALDATA bit indicates that the CAN protcol controller is transmitting or receiving a message and accessing the message slot j. The TRMACTIVE/INVALDATA bit is set to "1" when the CAN module is accessing the message slot and to "0 " when not accessing the message slot.

TRMACTIVE : The TRMACTIVE bit is set to "1" (except transmitting) when a data transmission is started in the message slot. If the CAN module loses in bus arbitration, the TRMACTIVE bit is set to "0" (stops transmitting) when a CAN bus error occurs or when a data transmission is completed.

INVALDATA : The INVALDATA bit is set to "1" (storing received data) when receiving a received message into the messaqe slot j, after a message reception is completed. Then the INVALDATA bit is set to "0" after a message storage is completed. Data, if read from the message slot j while this bit is set to "1", is indeterminate.

23.1.20.3 MSGLOST Bit

The MSGLOST bit is enabled only when the message slot is set for reception. The MSGLOST bit is set to "1" (overrun error occurred) when the message slot j is overwritten by a new received message while the NEWDATA bit set to "1" (already received).

The MSGLOST bit is not automatically set to "0". Set to "0" (no overrun error occurred) by program.

23.1.20.4 REMACTIVE Bit

The CiMCTL0 to CiMCTL15 registers all have the same function when the STATE_BASICCAN bit is set to "0" (other than BasicCAN mode).

The REMACTIVE bit is set to "1" (remote frame) when the message slot j is set to transmit or receive the remote frame. The REMACTIVE bit is set to "0" (data frame) after the remote frame has been transmitted or received.

The functions of the CiMCTL14 and CiMCTL15 registers change when the STATE_BASICCAN bit is set to "1" (BasicCAN mode). When the REMACTIVE bit is set to "0", this indicates that a message stored into the message slot is the data frame. When the REMACTIVE bit is set to "1", this indicates a message stored into the message slot is the remote frame.

23.1.20.5 RSPLOCK Bit

The RSPLOCK bit is enabled only when remote frame reception shown in Table 23.4 is selected. The RSPLOCK bit determines whether the received remote frame is processed or not.

When the RSPLOCK bit is set to "0" (automatic answering of the remote frame enabled), the slot automatically changes to a transmit slot after the remote frame is received and the message stored into the message slot is automatically transmitted as the data frame.

When the RSPLOCK bit is set to "1" (automatic answering of the remote frame disabled), message is not automatically transmitted upon receiving the remote frame.

Set the RSPLOCK bit to "0" to select any transmit/receive mode other than the remote frame reception.

23.1.20.6 REMOTE Bit

The REMOTE bit selects transmit/receive mode shown in Table 23.4. Set the REMOTE bit to "0" to transmit or receive data frame. Set to "1" to transmit or receive remote frame.

The followings occur during remote frame transmission or reception.

- Transmitting the remote frame

  A message stored into the message slot j (j=0 to 15) is transmitted as the remote frame. After transmission, the slot automatically becomes ready to receive data frame.

  If the data frame is received before the remote frame is transmitted, the data frame is stored into the message slot j. The remote frame is not transmitted.

- Receiving the remote frame

  The message slot receives the remote frame. The RSPLOCK bit determines whether or not to process the received remote frame.

23.1.20.7 RECREQ Bit

The RECREQ bit selects transmit/receive mode shown in Table 23.4. Set the RECREQ bit to "1" (receive requested) when data frame or remote frame is received. Set the RECREQ bit to "0" (no receive requested) when data frame or remote frame is transmitted.

When a data frame is automatically transmitted after a remote frame is received, the RECREQ bit remains set to "1". Set the RECREQ bit to "0" to transmit a remote frame. After a remote frame is transmitted, a data frame is automatically received while the RECREQ bit remains set to "0".

When setting the TRMREQ bit to "1" (transmit requested), do not set the RECREQ bit to "1" (receive requested).

23.1.20.8 TRMREQ Bit

The TRMREQ bit selects transmit/receive mode shown in Table 23.4. Set the TRMREQ bit to "1" (transmit requested) when data frame or remote frame is transmitted.

Set the TRMREQ bit to "0" (no request to transmit the frame) when data frame or remote frame is received.

When the data frame is automatically received after the remote frame is transmitted, the TRMREQ bit remains set to "1". Set the TRMREQ bit to "0" to receive the remote frame. After the remote frame is received, data frame is automatically transmitted while the TRMREQ bit remains set to "0".

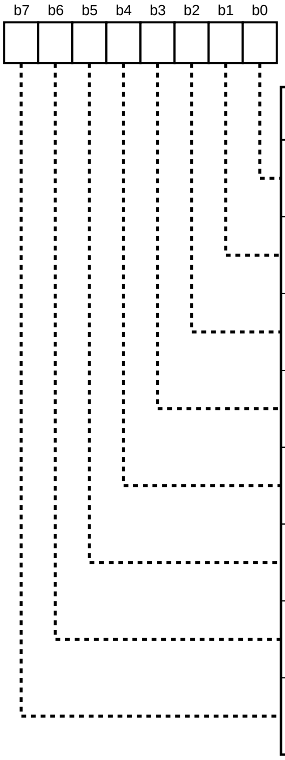
If the RECREQ bit is set to "1" (request to receive the frame), do not set the TRMREQ bit to "1" (request to transmit the frame).

NOTES:

1. If some message slots are requested to transmit the data frame or remote frame, the message slot, having the smallest slot number starts transmitting.
2. In single-shot mode, the CiMCTLj register is set to "00_{16}" when data transmission is failed, due to the arbitration lost or transmission error.

23.1.21 CANi Slot Buffer Select Register (CiSBS Register) (i=0,1)

CANi Slot Buffer Select Register (i=0,1)



Symbol	Address	After Reset[2]
C0SBS	0240_{16}	00_{16}
C1SBS	0250_{16}	00_{16}

Bit Symbol	Bit Name	Function	RW
SBS00	CANi Message Slot Buffer 0 Number Select Bit	b3 b2 b1 b0 0 0 0 0 : Message slot 0 0 0 0 1 : Message slot 1 0 0 1 0 : Message slot 2 0 0 1 1 : Message slot 3 ⋮ ⋮ (Note 1) 1 1 0 0 : Message slot 12 1 1 0 1 : Message slot 13 1 1 1 0 : Message slot 14 1 1 1 1 : Message slot 15	RW
SBS01			RW
SBS02			RW
SBS03			RW
SBS10	CANi Message Slot Buffer 1 Number Select Bit	b3 b2 b1 b0 0 0 0 0 : Message slot 0 0 0 0 1 : Message slot 1 0 0 1 0 : Message slot 2 0 0 1 1 : Message slot 3 ⋮ ⋮ (Note 1) 1 1 0 0 : Message slot 12 1 1 0 1 : Message slot 13 1 1 1 0 : Message slot 14 1 1 1 1 : Message slot 15	RW
SBS11			RW
SBS12			RW
SBS13			RW

NOTES:

1. 16 CANi message slots are provided. Each message slot can be selected as a transmit or a receive slot.
2. Value is obtained by setting the SLEEP bit in the CiSLPR register to "1" (sleep mode exited) after reset and supplying the clock to the CAN module.

Figure 23.30 C0SBS and C1SBS Registers

23.1.21.1 SBS03 to SBS00 Bits

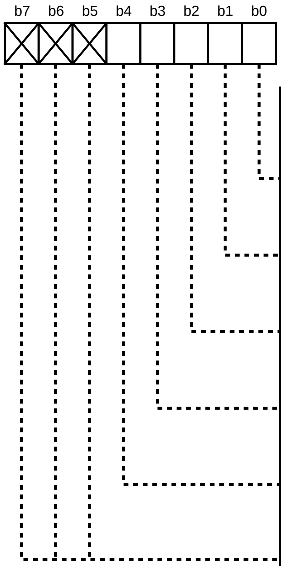
If the SBS03 to SBS00 bits select a number j (j=0 to 15), the message slot j is allocated to the CANi message slot buffer 0. The message slot j can be accessed via addresses $01E0_{16}$ to $01EF_{16}$, and 0260_{16} to $026F_{16}$.

23.1.21.2 SBS13 to SBS10 Bits

If the SBS13 to SBS10 bits select a number j, the message slot j is allocated to the CANi message slot buffer 1. The message slot j can be accessed via addresses $01F0_{16}$ to $01FF_{16}$, and 0270_{16} to $027F_{16}$.

23.1.22 CANi Message Slot Buffer j (i=0,1, j=0,1)

CANi Message Slot Buffer j Standard ID0 (i=0,1, j=0,1)(1)

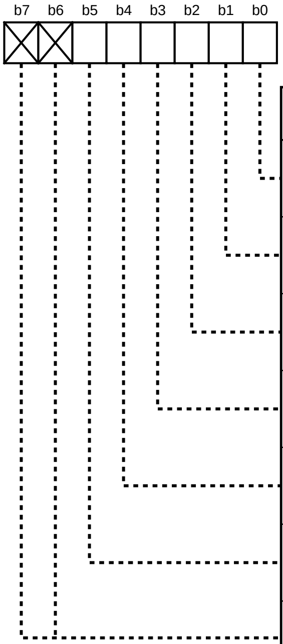


Symbol	Address	After Reset
C0SLOT0_0, C0SLOT1_0	$01E0_{16}$, $01F0_{16}$	Indeterminate
C1SLOT0_0, C1SLOT1_0	0260_{16}, 0270_{16}	Indeterminate

Bit Symbol	Bit Name	Function	RW
SID6	Standard ID6	Read or write the standard ID6 in the message slot k (k=0 to 15)	RW
SID7	Standard ID7	Read or write the standard ID7 in the message slot k	RW
SID8	Standard ID8	Read or write the standard ID8 in the message slot k	RW
SID9	Standard ID9	Read or write the standard ID9 in the message slot k	RW
SID10	Standard ID10	Read or write the standard ID10 in the message slot k	RW
____ (b7 - b5)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		—

NOTES:
1. Select, by setting the CiSBS register, the message slot k to be accessed by the CiSLOTj_0 register.

CANi Message Slot Buffer j Standard ID1 (i=0,1, j=0,1)(1)



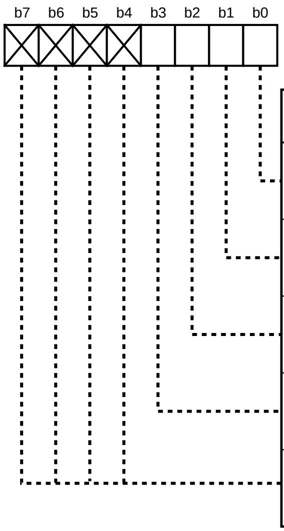
Symbol	Address	After Reset
C0SLOT0_1, C0SLOT1_1	$01E1_{16}$, $01F1_{16}$	Indeterminate
C1SLOT0_1, C1SLOT1_1	0261_{16}, 0271_{16}	Indeterminate

Bit Symbol	Bit Name	Function	RW
SID0	Standard ID0	Read or write the standard ID0 in the message slot k (k=0 to 15)	RW
SID1	Standard ID1	Read or write the standard ID1 in the message slot k	RW
SID2	Standard ID2	Read or write the standard ID2 in the message slot k	RW
SID3	Standard ID3	Read or write the standard ID3 in the message slot k	RW
SID4	Standard ID4	Read or write the standard ID4 in the message slot k	RW
SID5	Standard ID5	Read or write the standard ID5 in the message slot k	RW
____ (b7 - b6)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		—

NOTES:
1. Select, by setting the CiSBS register, the message slot k to be accessed by the CiSLOTj_1 register.

Figure 23.31 C0SLOT0_0, C0SLOT1_0, C0SLOT0_1 and C0SLOT1_1 Registers C1SLOT0_0, C1SLOT1_0, C1SLOT0_1 and C1SLOT1_1 Registers

CANi Message Slot Buffer j Extended ID0 (i=0,1, j=0,1)(1, 2)

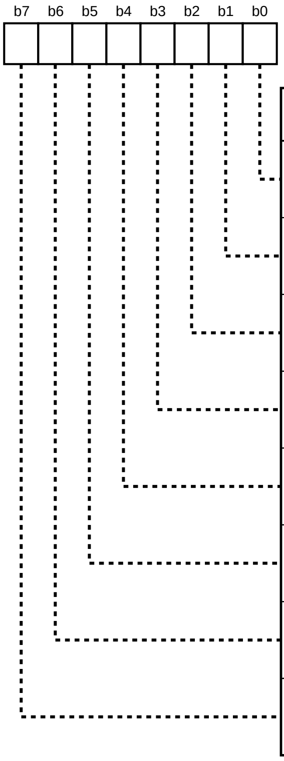


Symbol	Address	After Reset
C0SLOT0_2, C0SLOT1_2	$01E2_{16}$, $01F2_{16}$	Indeterminate
C1SLOT0_2, C1SLOT1_2	0262_{16}, 0272_{16}	Indeterminate

Bit Symbol	Bit Name	Function	RW
EID14	Extended ID14	Read or write the extended ID14 in the message slot k (k=0 to 15)	RW
EID15	Extended ID15	Read or write the extended ID15 in the message slot k	RW
EID16	Extended ID16	Read or write the extended ID16 in the message slot k	RW
EID17	Extended ID17	Read or write the extended ID17 in the message slot k	RW
____ (b7 - b4)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		_

NOTES:
1. If the receive slot is standard ID formatted, the EID17 to EID14 bits are indeterminate when received data is stored.
2. Select, by setting the CiSBS register, the message slot k to be accessed by the CiSLOTj_2 register.

CANi Message Slot Buffer j Extended ID1 (i=0,1, j=0,1)(1, 2)



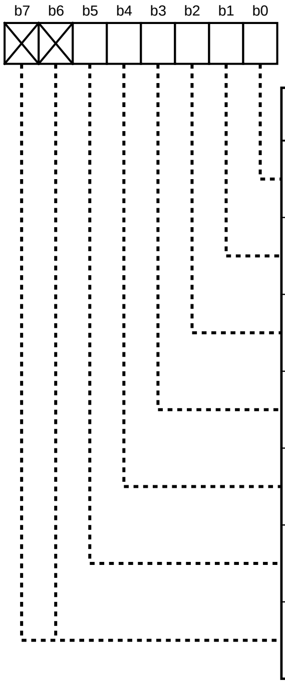
Symbol	Address	After Reset
C0SLOT0_3, C0SLOT1_3	$01E3_{16}$, $01F3_{16}$	Indeterminate
C1SLOT0_3, C1SLOT1_3	0263_{16}, 0273_{16}	Indeterminate

Bit Symbol	Bit Name	Function	RW
EID6	Extended ID6	Read or write the extended ID6 in the message slot k (k=0 to 15)	RW
EID7	Extended ID7	Read or write the extended ID7 in the message slot k	RW
EID8	Extended ID8	Read or write the extended ID8 in the message slot k	RW
EID9	Extended ID9	Read or write the extended ID9 in the message slot k	RW
EID10	Extended ID10	Read or write the extended ID10 in the message slot k	RW
EID11	Extended ID11	Read or write the extended ID11 in the message slot k	RW
EID12	Extended ID12	Read or write the extended ID12 in the message slot k	RW
EID13	Extended ID13	Read or write the extended ID13 in the message slot k	RW

NOTES:
1. If the receive slot is standard ID formatted, the EID13 to EID6 bits are indeterminate when received data is stored.
2. Select, by setting the CiSBS register, the message slot k to be accessed by the CiSLOTj_3 register.

Figure 23.32 C0SLOT0_2, C0SLOT1_2, C0SLOT0_3 and C0SLOT1_3 Registers C1SLOT0_2, C1SLOT1_2, C1SLOT0_3 and C1SLOT1_3 Registers

CANi Message Slot Buffer j Extended ID2 (i=0,1, j=0,1)[1, 2]



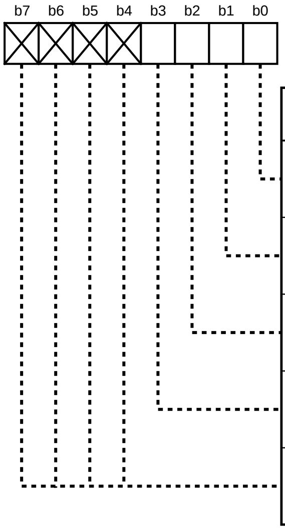
Symbol	Address	After Reset
C0SLOT0_4, C0SLOT1_4	$01E4_{16}$, $01F4_{16}$	Indeterminate
C1SLOT0_4, C1SLOT1_4	0264_{16}, 0274_{16}	Indeterminate

Bit Symbol	Bit Name	Function	RW
EID0	Extended ID0	Read or write the extended ID0 in the message slot k (k=0 to 15)	RW
EID1	Extended ID1	Read or write the extended ID1 in the message slot k	RW
EID2	Extended ID2	Read or write the extended ID2 in the message slot k	RW
EID3	Extended ID3	Read or write the extended ID3 in the message slot k	RW
EID4	Extended ID4	Read or write the extended ID4 in the message slot k	RW
EID5	Extended ID5	Read or write the extended ID5 in the message slot k	RW
____ (b7 - b6)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		_

NOTES:

1. If the receive slot is standard ID formatted, the EID5 to EID0 bits are indeterminate when received data is stored.
2. Select, by setting the CiSBS register, the message slot k to be accessed by the CiSLOTj_4 register.

CANi Message Slot Buffer j Data Length Code (i=0,1, j=0,1)[1]



Symbol	Address	After Reset
C0SLOT0_5, C0SLOT1_5	$01E5_{16}$, $01F5_{16}$	Indeterminate
C1SLOT0_5, C1SLOT1_5	0265_{16}, 0275_{16}	Indeterminate

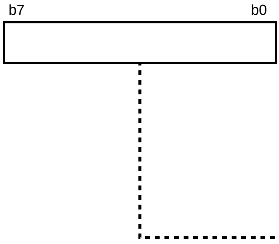
Bit Symbol	Bit Name	Function	RW
DLC0	Data Length Set Bit	Read or write the data length set bit in the message slot k (k=0 to 15)	RW
DLC1			RW
DLC2			RW
DLC3			RW
____ (b7 - b4)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		_

NOTES:

1. Select, by setting the CiSBS register, the message slot k to be accessed by the CiSLOTj_5 register.

**Figure 23.33 C0SLOT0_4, C0SLOT1_4, C0SLOT0_5 and C0SLOT1_5 Registers
C1SLOT0_4, C1SLOT1_4, C1SLOT0_5 and C1SLOT1_5 Registers**

CANi Message Slot Buffer j Data m (i=0,1, j=0,1)(1, 2)



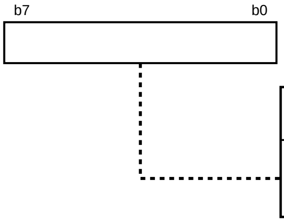
Symbol	Address	After Reset
C0SLOT0_6 to C0SLOT0_13	$01E6_{16}$ - $01ED_{16}$	Indeterminate
C0SLOT1_6 to C0SLOT1_13	$01F6_{16}$ - $01FD_{16}$	Indeterminate
C1SLOT0_6 to C1SLOT0_13	0266_{16} - $026D_{16}$	Indeterminate
C1SLOT1_6 to C1SLOT1_13	0276_{16} - $027D_{16}$	Indeterminate

Function	Setting Range	RW
Read or write data m in the message slot k (k=0 to 15, m=0 to 7)	00_{16} to FF_{16}	RW

NOTES:

1. Select, by setting the CiSBS register, the data *m* in the message slot *k* to be accessed by the CiSLOTj_6 to CiSLOTj_13 registers.
2. When the data frame is received, data with less than the data length selected by the CiSLOTj_5 register is indeterminate.

CANi Message Slot Buffer j Time Stamp High-Ordered (i=0,1, j=0,1)(1)



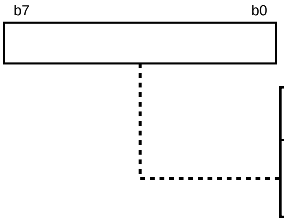
Symbol	Address	After Reset
C0SLOT0_14, C0SLOT1_14	$01EE_{16}$, $01FE_{16}$	Indeterminate
C1SLOT0_14, C1SLOT1_14	$026E_{16}$, $027E_{16}$	Indeterminate

Function	Setting Range	RW
Read or write the time stamp high-ordered in the message slot k (k=0 to 15)	00_{16} to FF_{16}	RW

NOTES:

1. Select, by setting the CiSBS register, the time stamp high-ordered in the message slot k to be accessed by the CiSLOTj_14 register.

CANi Message Slot Buffer j Time Stamp Low-Ordered (i=0,1, j=0,1)(1)



Symbol	Address	After Reset
C0SLOT0_15, C0SLOT1_15	$01EF_{16}$, $01FF_{16}$	Indeterminate
C1SLOT0_15, C1SLOT1_15	$026F_{16}$, $027F_{16}$	Indeterminate

Function	Setting Range	RW
Read or write the time stamp low-ordered in the message slot k (k=0 to 15)	00_{16} to FF_{16}	RW

NOTES:

1. Select, by setting the CiSBS register, the time stamp low-ordered in the message slot k to be accessed by the CiSLOTj_15 register.

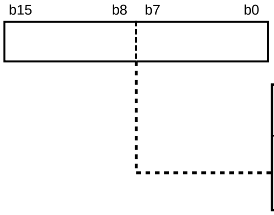
Figure 23.34 C0SLOT0_6 to C0SLOT0_13, C0SLOT1_6 to C0SLOT1_13, C0SLOT0_14, C0SLOT1_14, C0SLOT0_15 and C0SLOT1_15 Registers C1SLOT0_6 to C1SLOT0_13, C1SLOT1_6 to C1SLOT1_13, C1SLOT0_14, C1SLOT1_14, C1SLOT0_15 and C1SLOT1_15 Registers

The message slot, selected by setting the CiSBS register, is read by reading the message slot buffer. A message can be written in the message slot selected by the CiSBS register if the message is written to the message slot buffer.

Write to the message slot k (k=0 to 15) while the corresponding CiMCTLk register is set to "00_{16}".

23.1.23 CANi Acceptance Filter Support Register (CiAFS Register) (i=0,1)

CANi Acceptance Filter Support Register (i=0,1)

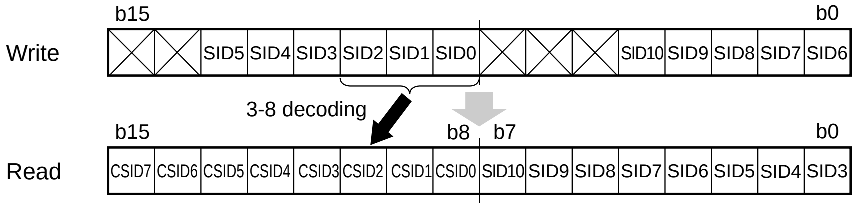


Symbol	Address	After Reset[1]
C0AFS	0245$_{16}$ - 0244$_{16}$	0100$_{16}$
C1AFS	0255$_{16}$ - 0254$_{16}$	0100$_{16}$

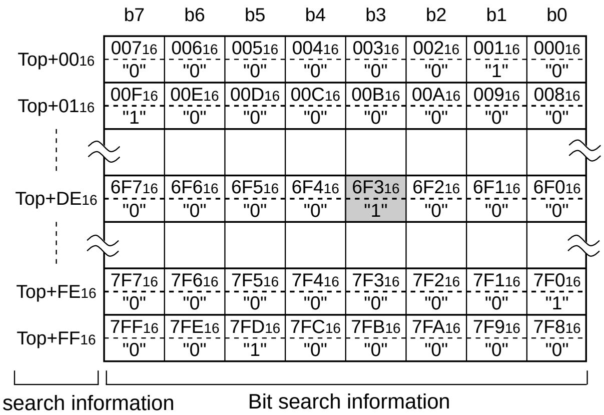
Function	Setting Range	RW
Generates data to determine a received ID	0000$_{16}$ to FFFF$_{16}$	RW

NOTES:

1. Value is obtained by setting the SLEEP bit in the CiSLPR register to "1" (sleep mode exited) after reset and supplying the clock to the CAN module.

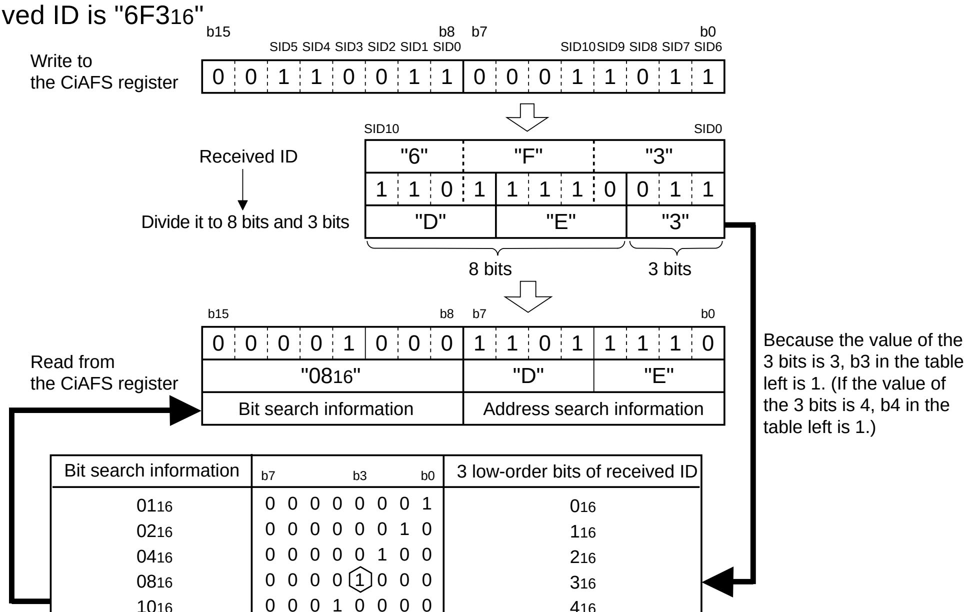


Data used to search a data table is generated from a received ID in standard format. The table search with this data determines whether or not a received ID is valid.



	b7	b6	b5	b4	b3	b2	b1	b0
Top+00$_{16}$	007$_{16}$ "0"	006$_{16}$ "0"	005$_{16}$ "0"	004$_{16}$ "0"	003$_{16}$ "0"	002$_{16}$ "0"	001$_{16}$ "1"	000$_{16}$ "0"
Top+01$_{16}$	00F$_{16}$ "1"	00E$_{16}$ "0"	00D$_{16}$ "0"	00C$_{16}$ "0"	00B$_{16}$ "0"	00A$_{16}$ "0"	009$_{16}$ "0"	008$_{16}$ "0"
⋮								
Top+DE$_{16}$	6F7$_{16}$ "0"	6F6$_{16}$ "0"	6F5$_{16}$ "0"	6F4$_{16}$ "0"	6F3$_{16}$ "1"	6F2$_{16}$ "0"	6F1$_{16}$ "0"	6F0$_{16}$ "0"
⋮								
Top+FE$_{16}$	7F7$_{16}$ "0"	7F6$_{16}$ "0"	7F5$_{16}$ "0"	7F4$_{16}$ "0"	7F3$_{16}$ "0"	7F2$_{16}$ "0"	7F1$_{16}$ "0"	7F0$_{16}$ "1"
Top+FF$_{16}$	7FF$_{16}$ "0"	7FE$_{16}$ "0"	7FD$_{16}$ "1"	7FC$_{16}$ "0"	7FB$_{16}$ "0"	7FA$_{16}$ "0"	7F9$_{16}$ "0"	7F8$_{16}$ "0"

When a received ID is "6F3$_{16}$"



Because the value of the 3 bits is 3, b3 in the table left is 1. (If the value of the 3 bits is 4, b4 in the table left is 1.)

Bit search information	b7 b3 b0	3 low-order bits of received ID
01$_{16}$	0 0 0 0 0 0 0 1	0$_{16}$
02$_{16}$	0 0 0 0 0 0 1 0	1$_{16}$
04$_{16}$	0 0 0 0 0 1 0 0	2$_{16}$
08$_{16}$	0 0 0 0 1 0 0 0	3$_{16}$
10$_{16}$	0 0 0 1 0 0 0 0	4$_{16}$
20$_{16}$	0 0 1 0 0 0 0 0	5$_{16}$
40$_{16}$	0 1 0 0 0 0 0 0	6$_{16}$
80$_{16}$	1 0 0 0 0 0 0 0	7$_{16}$

i = 0, 1

Figure 23.35 C0AFS Register and C1AFS Register

The CiAFS register enables prompt performance of the table search to determine the varidity of a received ID. This function is for standard-formatted ID only.

23.2 CAN Clock

The CAN clock is the operating clock for the CAN module. f_1 or f_{CAN} can be selected as the CAN clock. f_{CAN} has the same frequency as the main clock. The PM25 bit in the PM2 register determines the CAN clock. Refer to **9. Clock Generation Circuit** for details.

23.2.1 Main Clock Direct Mode

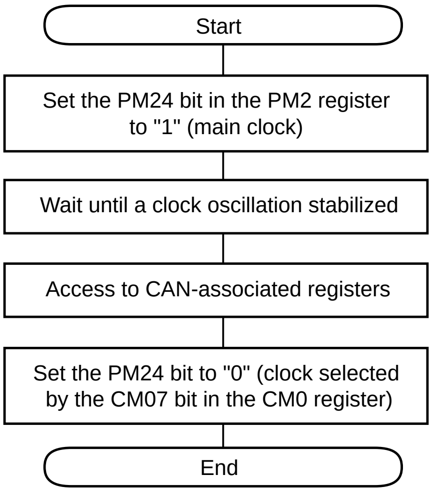
f_{CAN} becomes the CAN clock in main clock direct mode. The CAN module must enter main clock direct mode while the PM25 bit is set to "1" (main clock). Set the PM25 bit in CAN sleep mode.

Set the PM24 bit in the PM2 register to "1" (main clock) before accessing CAN-associated registers in main clock direct mode. Do not enter wait mode or stop mode when the PM24 bit is set to "1".

Table 23.5 lists CAN clock settings. Figure 23.36 shows a flow chart of accessing procedure for CAN-associated registers.

Table 23.5 CAN Clock Settings

CAN Clock	Clock Source	CM0 Register	CM1 Register	CM2 Register	PM2 Register		MCD Register
		CM07 Bit	CM17 Bit	CM21 Bit	PM24 Bit	PM25 Bit	MCD4 to MCD0 bits
f_{CAN}	Main Clock (Main Clock Direct Mode)	0	1	0	1	1	---
f_1	Main Clock	0	0	0	0	0	10010_2
	PLL Clock	0	1	0	0	0	10010_2


Start
Set the PM24 bit in the PM2 register to "1" (main clock)
Wait until a clock oscillation stabilized
Access to CAN-associated registers
Set the PM24 bit to "0" (clock selected by the CM07 bit in the CM0 register)
End

NOTES:

1. Waiting time varies depending on the CPU clock frequency before or after PM24 bit setting is changed.
   - High Frequency : Higher Frequency compared "before PM24 bit setting changes" with "after PM24 bit setting changes"
   - Low Frequency : Lower Frequency compared "before PM24 bit setting changes" with "after PM24 bit setting changes"

$$\text{Waiting Time} \geq \frac{2 \text{ x High Frequency}}{\text{Low Frequency}} \text{ Cycles}$$

Figure 23.36 Accessing Procedure for CAN-Associated Registers

23.3 Timing with CAN-Associated Registers

23.3.1 CAN Module Reset Timing

Figure 23.37 shows an operation example of when the CAN module is reset.

(1) The CAN module can be reset when the STATE_RESET bit in the CiSTR register (i=0,1) is set to "1" (CAN module reset completed) after the RESET1 and RESET0 bits in the CiCTLR0 register are set to "1" (CAN module reset).

(2) Set necessary CAN-associated registers.

(3) CAN communication can be established after the STATE_RESET bit is set to "0" (resetting) after the RESET1 and RESET0 bits are set to "0" (CAN module reset exited) .

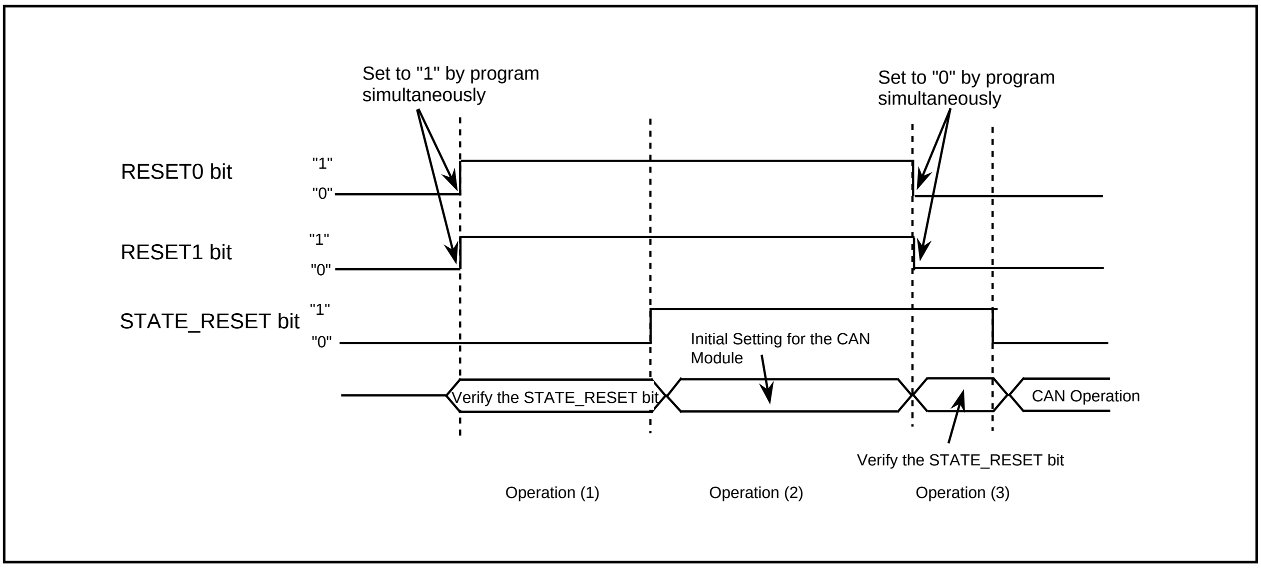

Set to "1" by program simultaneously
Set to "0" by program simultaneously
RESET0 bit
"1"
"0"
RESET1 bit
"1"
"0"
STATE_RESET bit
"1"
"0"
Initial Setting for the CAN Module
Verify the STATE_RESET bit
CAN Operation
Verify the STATE_RESET bit
Operation (1)
Operation (2)
Operation (3)

Figure 23.37 Example of CAN Module Reset Operation

23.3.2 CAN Transmit Timing

Figure 23.38 shows an operation example of when the CAN transmits a frame.

(1) When the TRMREQ bit in the CiMCTLj register (j=0 to 15) is set to "1" (request to transmit the data frame) while the CAN bus is in an idle state, the TRMACTIVE bit in the CiMCTLj register is set to "1" (during transmission) and the TRMSTATE bit in the CiSTR register is set to "1" (during transmission). The CAN starts transmitting the frame.

(2) After a CAN frame transmission is completed, the SENTDATA bit in the CiMCTLj register is set to "1" (already transmitted), the TRMSUCC bit in the CiSTR register to "1" (transmission completed) and the SISj bit in the CiSISTR register to "1" (interrupt requested). The MBOX3 to MBOX0 bits in the CiSTR register store transmitted message slot numbers.

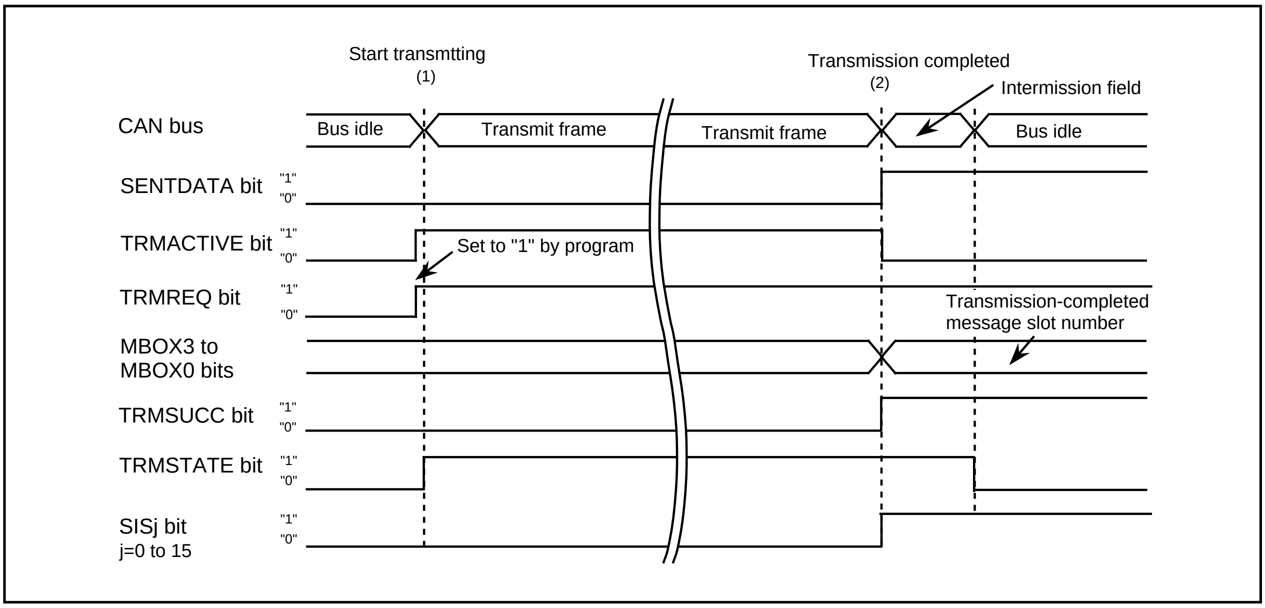


Figure 23.38 Example of CAN Data Frame Transmit Operation

23.3.3 CAN Receive Timing

Figure 23.39 shows an operation example of when the CAN receives a frame.

(1) When the RECREQ bit in the CiMCTLj register (i=0,1, j= 0 to 15) is set to "1" (receive requested), the CAN is ready to receive the frame at anytime.

(2) When the CAN starts receiving the frame, the RECSTATE bit in the CiSTR register is set to "1" (during reception).

(3) After the CAN frame reception is completed, the INVALDATA bit in the CiMCTLj register is set to "1" (storing received data), the NEWDATA bit in the CiMCTLj register is set to "1" (receive complete) and the RECSUCC bit in the CiSTR register is set to "1" (reception completed).

(4) After data is written to the message slot, the INVALDATA bit is set to "0" (storing receiving data) and the SISj bit in the CiSISTR register is set to "1" (interrupt requested). The MBOX3 to MBOX0 bits in the CiSTR register store received message slot numbers.

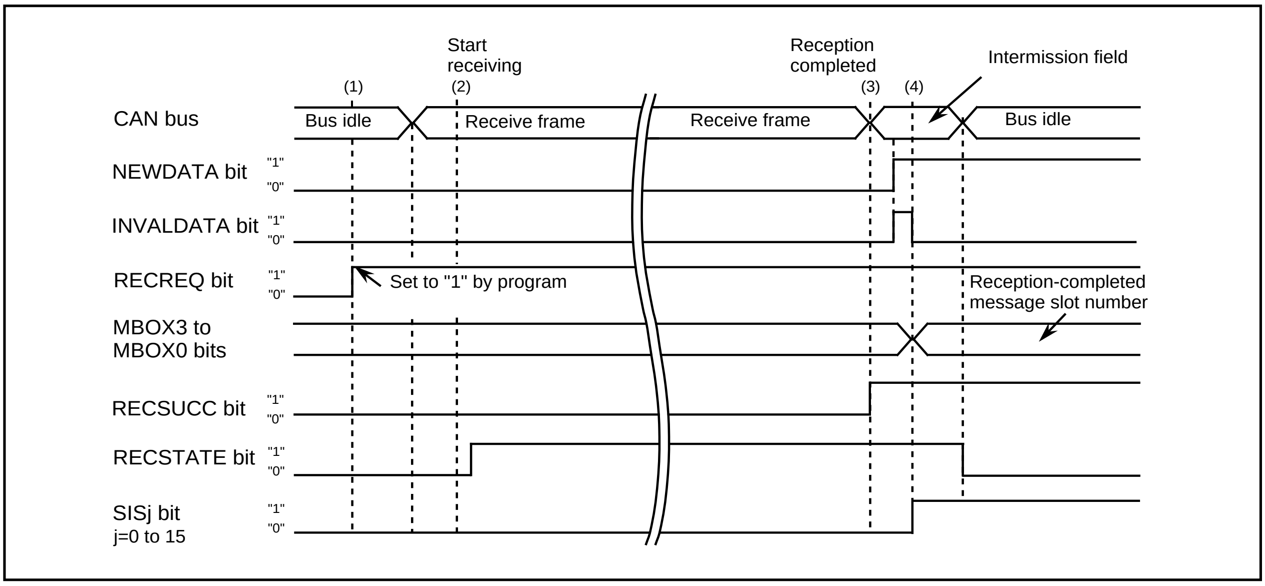


Figure 23.39 Example of CAN Data Frame Receive Operation

23.3.4 CAN Bus Error Timing

Figure 23.40 shows an operation example of when a CAN bus error occurs.

(1) When a CAN bus error is detected, the STATE_BUSERROR bit in the CiSTR register is set to "1", (error occurred) and the BEIS bit in the CiEISTR register is set to "1" (interrupt requested). The CAN starts transmitting the error frame.

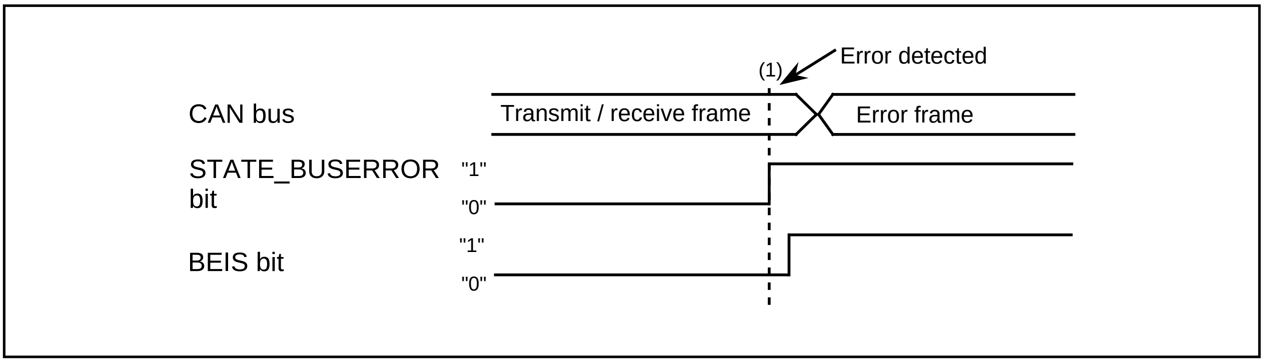


Figure 23.40 Operation Timing when CAN Bus Error Occurs

23.4 CAN Interrupts

The CAN1 wake-up interrupt and CANij interrupts (i=0,1,j=0 to 2) are provided as the CAN interrupt.

23.4.1 CAN1 Wake-Up Interrupt

When a signal applied to the $\overline{\text{CAN1WU}}$ pin is on the falling edge, the CAN1WUR bit in the IIO5IR register is set to "1" (interrupt requested). At this time, the IR bit in the CAN5IC register is set to "1" (interrupt requested) if the CAN1WUE bit in the IIO5IE register is set to "1" (interrupt enabled).

If $P7_7$ ($CAN0_{IN}$) is used as a CAN0 input port, the CAN0 wake-up interrupt is available by using event counter mode of Timer A3 ($TA3_{IN}$) that shares a pin with CAN0.

If $P8_3$ ($CAN0_{IN}$/$CAN1_{IN}$) is used as a CAN input port, the CAN0 and CAN1 wake-up interrupts are available by using $\overline{\text{INT1}}$ that shares a pin with $CAN0_{IN}$/$CAN1_{IN}$.

23.4.2 CANij Interrupts

Figure 23.41 shows a block diagram of the CANij interrupts. The followings cause the CAN-associated interrupt request to be generated.

- The CANi slot k (k=0 to 15) completes a transmission
- The CANi slot k completes a reception
- The CANi module detects a bus error
- The CANi module moves into an error-passive state
- The CANi module moves into a bus-off state

The INTSEL bit in the CiCTLR1 register determines how an interrupt request is generated. When the INTSEL bit is set to "0", one of the above CANi interrupt request source causes the CANij interrupts to be generated by the OR circuit. When the INTSEL bit is set to "1", CANi transmission completed, CANi reception completed and CANi errors (CANi bus error detection, CANi module into error-passive state and CANi module into bus-off state) cause the CANij interrupt corresponding to each source to be generated.

23.4.2.1 When the INTSEL Bit is Set to "0"

If the CAN-associated interrupt is generated by one of the interrupt request source listed in **23.4.2 CANij Interrupts**, the corresponding bit in the CiSISTR register (i=0,1) is set to "1" (interrupt requested) when the CANi slot k completes a transmission or a reception. The corresponding bit in the CiEISTR register is set to "1" (interrupt requested) when the CANi module detects a bus error, moves into an error-passive state, or moves into a bus-off state.

The CANi interrupt request signal is set to "1" when the corresponding bit in the CiSISTR or CiEISTR is set to "1" and the corresponding bit in the CiSIMKR or CiEIMKR is set to "1"

When the CAN0 interrupt request signal changes "0" to "1", all CAN0jR bits (j=0 to 2) in the IIO9IR to IIO11IR registers are set to "1" (interrupt requested).

If at least one of the CAN0jE bits in the IIO9IE to IIO11IE registers is set to "1" (interrupt enabled), the IR bits in the corresponding CAN0IC to CAN2IC registers are set to "1" (interrupt requested). The CAN0 interrupt request signal remains set to "1" if another interrupt request source causes a corresponding bit in the C0SISTR or C0EISTR to be set to "1" and the corresponding bit in the C0SIMKR or C0EIMKR to be set to "1" after the CAN0 interrupt request signal changes "0" to "1". The CAN0jR and IR bits also remain unchanged.

When the CAN1 interrupt request signal changes "0" to "1", all three CAN1jR bits in the IIO0IR to IIO1IR and IIO5IR registers are set to "1" (interrupt requested).

If at least one of the CAN1jE bits in the IIO0IE to IIO1IE and IIO5IE registers is set to "1", the IR bits in the corresponding CAN3IC to CAN5IC registers are set to "1". The CAN0 interrupt request signal remains set to "1" if another interrupt request causes the corresponding bit in the C1SISTR or C1EISTR to be set to "1" and the corresponding bit in the C1SIMKR or C1EIMKR to be set to "1" after the CAN1 interrupt request signal changes "0" to "1". The CAN1jR and IR bits also remain unchanged.

Bits in the CiSISTR or CiEISTR register and CANijR bits (i=0,1, j=0 to 2) in the IIO0IR to IIO1IR, IIO5IR or IIO9IR to IIO11IR registers are not set to "0" automatically, interrupt acknowledgment notwithstanding. Set these bits to "0" by program.

The CANi interrupts are acknowledged when the CANijR bit in the IIO0IR to IIO1IR, IIO5IR or IIO9IR to IIO11IR register and the corresponding bit in the CiSISTR or CiEISTR register are set to "0". If these bits remain set to "1", all CAN-associated interrupt request source become invalid.

23.4.2.2 When the INTSEL Bit is Set to "1"

If the CAN-associated interrupt is generated by one of the interrupt request source listed in **23.3.2 CANij Interrupts**, the corresponding bit in the CiSISTR register (i=0,1) is set to "1" (interrupt requested) when the CANi slot k completes a transmission or a reception. The corresponding bit in the CiEISTR register is set to "1" (interrupt requested) when the CANi module detects a bus error, goes into an error-passive state, or goes into a bus-off state.

The CANi receive interrupt request signal is set to "1" if the corresponding bit in the CiSIMKR register is set to "1" (interrupt request enabled) and the corresponding bit in the CiSISTR register is set to "1" when the CANi module completes a reception.
The CANi transmit interrupt request signal is set to "1" if the corresponding bit in the CiSIMKR register is set to "1" and the corresponding bit in the CiSISTR register is set to "1" when the CANi module completes a transmission.
The CANi error interrupt request signal is set to "1" if corresponding bits in the CiEIMKR register are set to "1" and the corresponding bit in the CiEISTR register is set to "1" when the CANi module detects a bus error, goes into an error-passive state, or goes into a bus-off state.

When the CANi receive interrupt request signal changes "0" to "1", the CAN00R bit in the IIO9IR register and the CAN10R bit in the IIO0IR registers are set to "1" (interrupt requested). If the CAN00E in the IIO9IE register is set to "1" (interrupt enabled), the IR bit in the CAN0IC register is set to "1" (interrupt requested). If the CAN10E bit in the IIO0IE register is set to "1" (interrupt enabled), the IR bit in the CAN3IC register is set to "1" (interrupt requested).
When the CANi transmit interrupt request signal changes "0" to "1", the CAN01R bit in the IIO10IR register and the CAN11R bit in the IIO1IR registers are set to "1" (interrupt requested). If the CAN01E in the IIO10IE register is set to "1" (interrupt enabled), the IR bit in the CAN1IC register is set to "1" (interrupt requested). If the CAN11E bit in the IIO1IE register is set to "1" (interrupt enabled), the IR bit in the CAN4IC register is set to "1" (interrupt requested).
When the CANi error interrupt request signal changes "0" to "1", the CAN02R bit in the IIO11IR register and CAN12R bit in the IIO5IR register are set to "1" (interrupt requested). If the CAN02E in the IIO11IE register is set to "1" (interrupt enabled), the IR bit in the CAN2IC register is set to "1" (interrupt requested). If the CAN12E bit in the IIO5IE register is set to "1" (interrupt enabled), the IR bit in the CAN5IC register is set to "1" (interrupt requested).

The CANi error interrupt request signal remains set to "1" if another interrupt request causes the corresponding bit in the CiEIMKR register is set to "1" and the corresponding bit in the CiEISTR to be set to "1" after the CANi error interrupt request signal changes "0" to "1". The CAN02R, CAN12R and IR bits also remain unchanged.

Bits in the CiSISTR or CiEISTR register and CANijR bits (i=0,1, j=0 to 2) in the IIO0IR to IIO1IR, IIO5IR or IIO9IR to IIO11IR registers are not set to "0" automatically, interrupt acknowledgment notwithstanding. Set these bits to "0" by program.
The CANi receive interrupt and CANi transmit interrupt are acknowledged when the CAN00R bit in the IIO9IR register, the CAN01R bit in the IIO10IR register, the CAN10R bit in the IIO0IR register and the CAN11R bit in the IIO1IR register are set to "0". Corresponding bits in the CiSISTR register can be set to either "0" or "1".
The CANi error interrupt is acknowledged when the CAN02R bit in the IIO11IR register, the CAN12R bit in the IIO5IR register and corresponding bits in the CiEISTR register are set to "0".
If these bits remain set to "1", all CAN-associated interrupt request source become invalid.

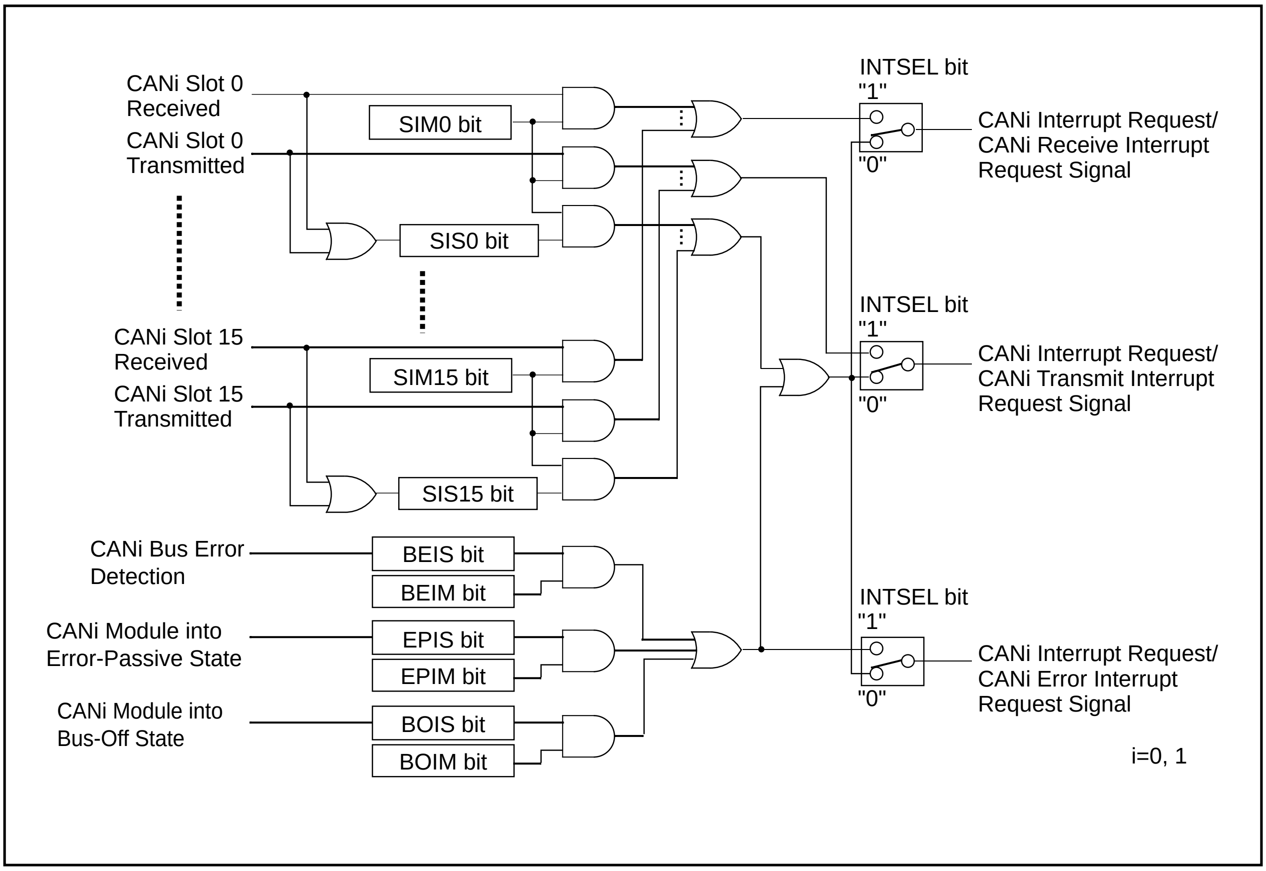

CANi Slot 0 Received
CANi Slot 0 Transmitted
SIM0 bit
SIS0 bit
CANi Slot 15 Received
CANi Slot 15 Transmitted
SIM15 bit
SIS15 bit
CANi Bus Error Detection
BEIS bit
BEIM bit
CANi Module into Error-Passive State
EPIS bit
EPIM bit
CANi Module into Bus-Off State
BOIS bit
BOIM bit
INTSEL bit
"1"
"0"
CANi Interrupt Request/ CANi Receive Interrupt Request Signal
INTSEL bit
"1"
"0"
CANi Interrupt Request/ CANi Transmit Interrupt Request Signal
INTSEL bit
"1"
"0"
CANi Interrupt Request/ CANi Error Interrupt Request Signal
i=0, 1

Figure 23.41 CAN Interrupts

24. Programmable I/O Ports

123 programmable I/O ports from P0 to P15 (excluding $P8_5$) are available. The direction registers determine each port status, input or output. The pull-up control registers determine whether the ports, divided into groups of four ports, are pulled up or not. $P8_5$ is an input port and no pull-up for this port is allowed. The P8_5 bit in the P8 register indicates an $\overline{\text{NMI}}$ input level since $P8_5$ shares pins with $\overline{\text{NMI}}$.

Figures 24.1 to 24.4 show programmable I/O port configurations.

Each pin functions as the programmable I/O port, an I/O pin for internal peripheral functions or the bus control pin.

To use the pins as input or output pins for internal peripheral functions, refer to the explanations for each fuction. Refer to **8. Bus** when used as the bus control pin.

The registers associated with the programmable I/O ports are as follows.

24.1 Port Pi Direction Register (PDi Register, i=0 to 15)

Figure 24.5 shows the PDi register.

The PDi register selects input or output status of a programmable I/O port. Each bit in the PDi register corresponds to a port.

In memory expansion and microprocessor mode, the PDi register cannot control pins being used as bus control pins (A_0 to A_{22}, $\overline{A_{23}}$, D_0 to D_{15}, $\overline{\text{CS0}}$ to $\overline{\text{CS3}}$, $\overline{\text{WRL}}/\overline{\text{WR}}$, $\overline{\text{WRH}}/\overline{\text{BHE}}$, $\overline{\text{RD}}$, BCLK/ALE/CLK_{OUT}, $\overline{\text{HLDA}}$/ALE, $\overline{\text{HOLD}}$, ALE and $\overline{\text{RDY}}$). No bit controlling $P8_5$ is provided in the direction registers.

24.2 Port Pi Register (Pi Register, i=0 to 15)

Figure 24.6 shows the Pi register.

The Pi register writes and reads data to communicate with external devices. The Pi register consists of a port latch to hold output data and a circuit to read pin states. Each bit in the Pi register corresponds to a port.

In memory expansion and microprocessor mode, the Pi register cannot control pins being used as bus control pins (A_0 to A_{22}, $\overline{A_{23}}$, D_0 to D_{15}, $\overline{\text{CS0}}$ to $\overline{\text{CS3}}$, $\overline{\text{WRL}}/\overline{\text{WR}}$, $\overline{\text{WRH}}/\overline{\text{BHE}}$, $\overline{\text{RD}}$, BCLK/ALE/CLK_{OUT}, $\overline{\text{HLDA}}$/ALE, $\overline{\text{HOLD}}$, ALE and $\overline{\text{RDY}}$).

24.3 Function Select Register Aj (PSj Register) (j=0 to 3, 5 to 9)

Figures 24.7 to 24.11 show the PSj registers.

The PSj register selects either I/O port or peripheral function output if an I/O port shares pins with a peripheral function output (excluding DA0 and DA1.)

When multiple peripheral function outputs are assigned to a pin, set the PSL0 to PSL3, PSL6, PSL7, PSC, PSC2, PSC3 and PSD1 registers to select which function is used.

Tables 24.3 to 24.12 list peripheral function output control settings for each pin.

24.4 Function Select Register Bk (PSLk Registers) (k=0 to 3, 6, 7)

Figures 24.12 to 24.14 show the PSLk registers.

When multiple peripheral function outputs are assigned to a pin, the PSLk registers select which peripheral function output is used.

Refer to **24.10 Analog Input and Other Peripheral Function Input** for the PSL3_6 to PSL3_3 bits in the PSL3 register.

24.5 Function Select Register C, C2, C3 (PSC, PSC2, PSC3 Registers)

Figures 24.15 and 24.16 show the PSC, PSC2 and PSC3 registers.

When multiple peripheral function outputs are assigned to a pin, the PSC, PSC2 and PSC3 registers select which peripheral function output is used.

Refer to **24.10 Analog Input and Other Peripheral Function Input** for the PSC_7 bit in the PSC register.

24.6 Function Select Register D (PSD1 Register)

Figure 24.16 shows the PSD1 register.

When multiple peripheral function outputs are assigned to a pin, the PSD1 register selects which peripheral function output is used.

24.7 Pull-up Control Register 0 to 4 (PUR0 to PUR4 Registers)

Figures 24.17 and 24.18 show the PUR0 to PUR4 registers.

The PUR0 to PUR4 registers select whether the ports, divided into groups of four ports, are pulled up or not. Ports with bits in the PUR0 to PUR4 registers set to "1" (pull-up) and the direction registers set to "0" (input mode) are pulled up.

Set bits in the PUR0 and PUR1 registers in P0 to P5, running as bus, to "0" (no pull-up) in memory expansion mode and microprocessor mode. P0, P1 and $P4_0$ to $P4_3$ can be pulled up when they are used as input ports in memory expansion mode and microprocessor mode.

24.8 Port Control Register (PCR Register)

Figure 24.19 shows the PCR register.

The PCR register selects either CMOS output or N-channel open drain output as the P1 output format. If the PCR0 bit is set to "1", N-channel open drain output is selected because the P-channel in the CMOS port is turned off. This is, however, not a perfect open drain. Therefore, the absolute maximum rating of the input voltage is between -0.3V and V_{CC} + 0.3V.

If P1 is used as the data bus in memory expansion mode and microprocessor mode, set the PCR0 bit to "0". If P1 is used as a port in memory expansion mode and microprocessor mode, the PCR0 bit determines the output format.

24.9 Input Function Select Register (IPS and IPSA Registers)

Figures 24.19 and 24.20 show the IPS and IPSA registers.

The IPS3, IPS1 and IPS0 bits in the IPS register and the IPSA_3 and IPSA_0 bits in the IPSA register select which pin is assigned for the intelligent I/O or CAN input functions.

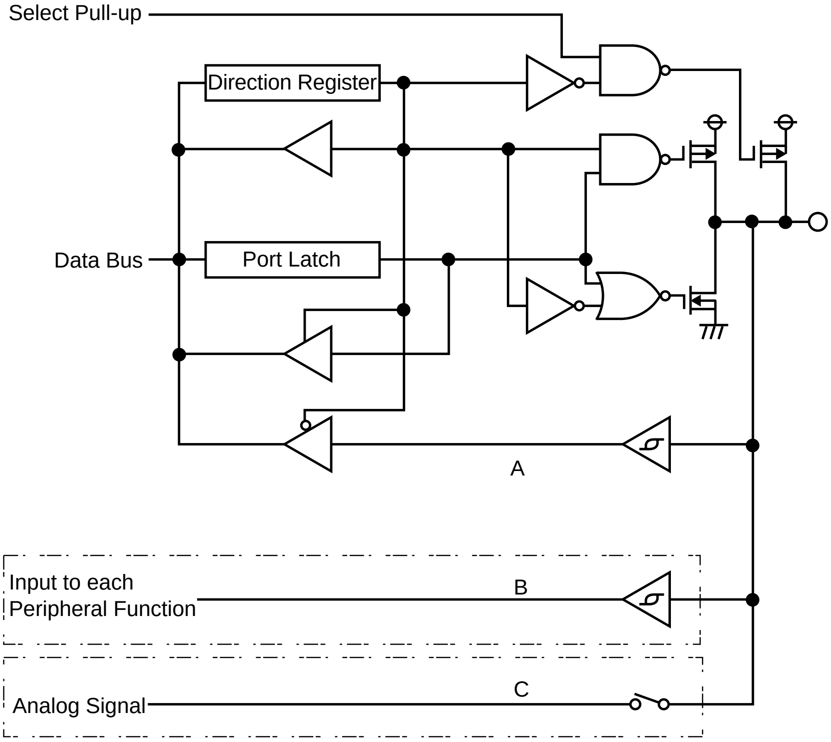
Refer to **24.10 Analog Input and Other Peripheral Function Input** for the IPS2 bit.

24.10 Analog Input and Other Peripheral Function Input

The PSL3_6 to PSL3_3 bits in the PSL3 register, the PSC_7 bit in the PSC register and the IPS2 bit in the IPS register each separate analog I/O ports from other peripheral functions. Setting the corresponding bit to "1" (analog I/O) to use the analog I/O port (DA0, DA1, ANEX0, ANEX1, AN_4 to AN_7 or $AN15_0$ to $AN15_7$) prevents an intermediate potential from being impressed to other peripheral functions. The impressed intermediate potential may cause increase in power consumption.

Set the corresponding bit to "0" (except analog I/O) when analog I/O is not used. All peripheral function inputs except the analog I/O port are available when the corresponding bit is set to "0". These inputs are indeterminate when the bit is set to "1". When the PSC_7 bit is set to "1", key input interrupt request remains unchanged regardless of $\overline{KI_0}$ to $\overline{KI_3}$ pin input level change.

Programmable I/O Ports

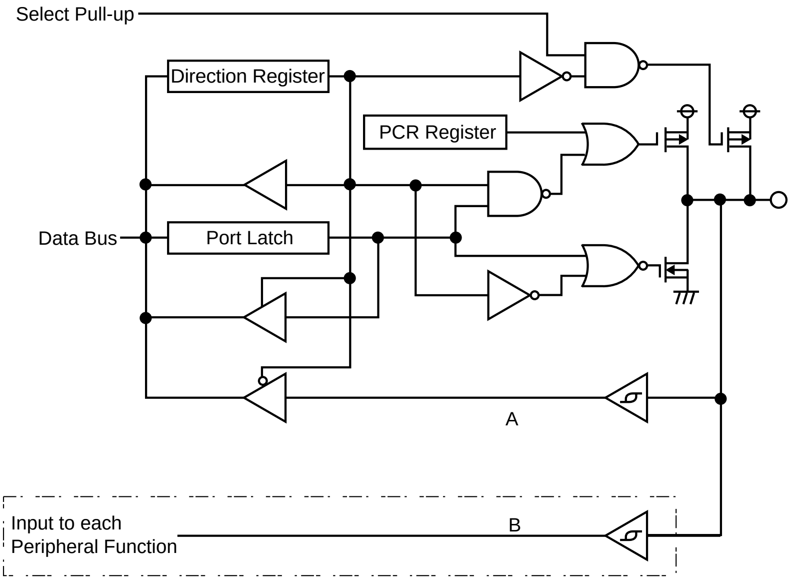


Option / Port	(A) Hysteresis	Circuit (B) Peripheral Function Input	Circuit (C) Analog I/F
$P0_0$ to $P0_7$ $P2_0$ to $P2_7$	–	–	○
$P3_0$ to $P3_7$ $P4_0$ to $P4_7$ $P5_0$ to $P5_2$ $P5_4$	–	–	–
$P5_5$	–	○	–
$P5_6$	–	–	–
$P5_7$	–	○	–
$P8_3$, $P8_4$	○	○	–
$P8_6$	–	–	–
$P8_7$	–	○	○
$P10_0$ to $P10_3$	–	–	○
$P10_4$ to $P10_7$	○	○	○
$P11_4$ $P14_4$ to $P14_6$	–	–	–
$P15_2$ to $P15_7$	–	–	○

○ : Available –: Not Available

Figure 24.1 Programmable I/O Ports (1)

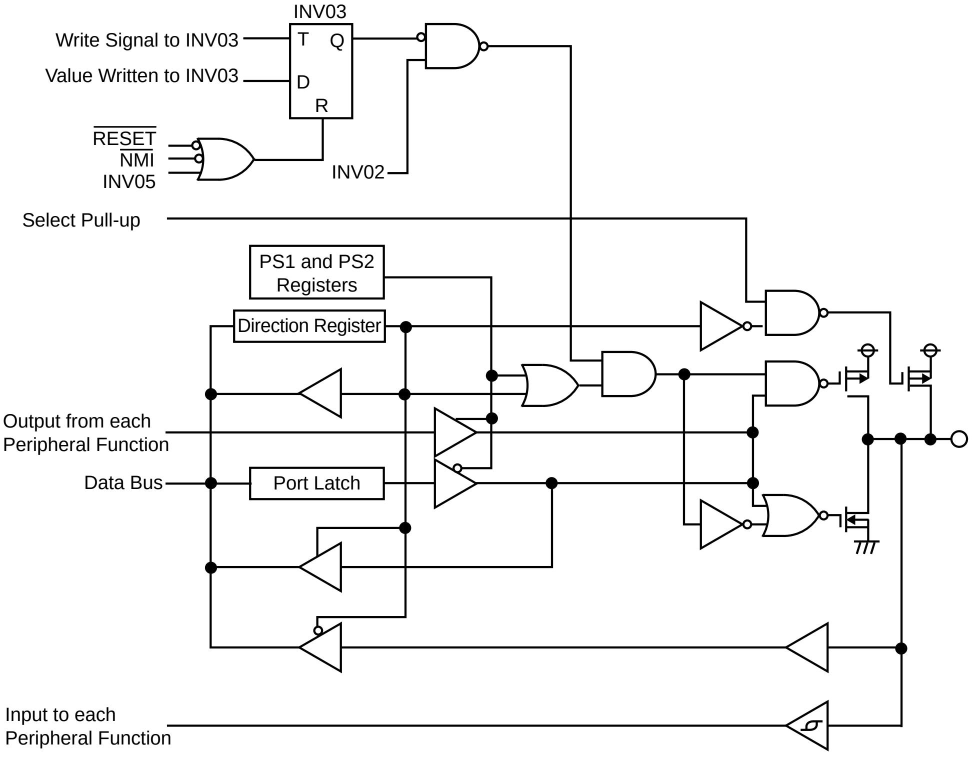
Programmable I/O Ports with the Port Control Register



Option / Port	(A) Hysteresis	Circuit (B) Peripheral Function Input
P1$_0$ to P1$_4$	–	–
P1$_5$ to P1$_7$	○	○

○ : Available –: Not Available

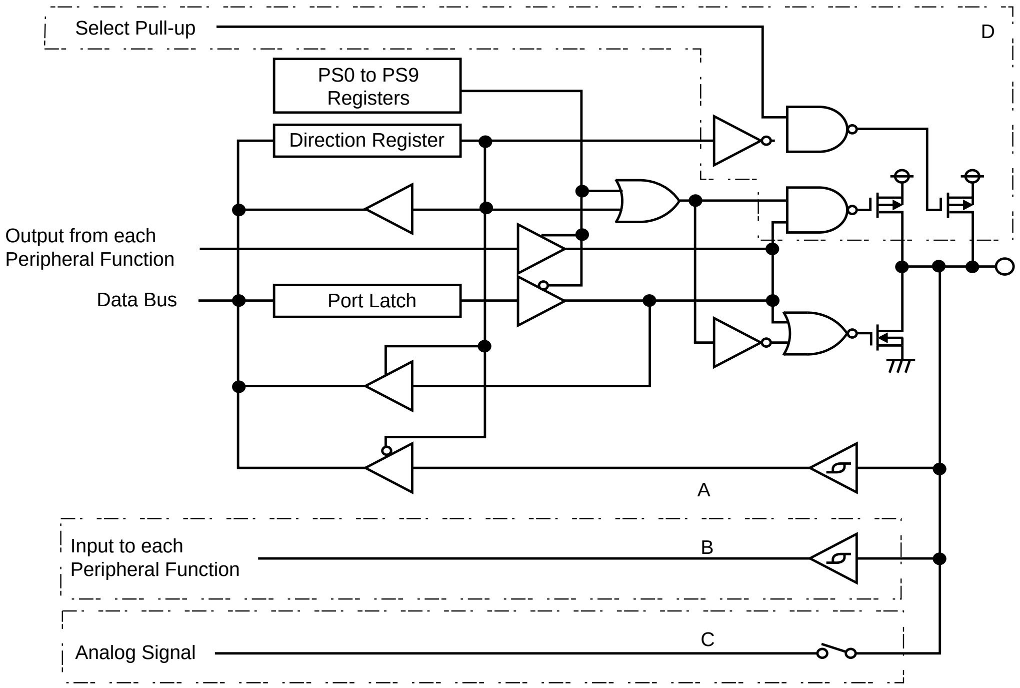
Programmable I/O Ports with the Function Select Register



Port : P7$_2$, P7$_3$, P7$_4$, P7$_5$, P8$_0$, P8$_1$

Figure 24.2 Programmable I/O Ports (2)

Programmable I/O ports with the Function Select Register



Option / Port	(A) Hysteresis	Circuit (B) Peripheral Function Input	Circuit (C) Analog I/F	Circuit (D)
$P5_3$	—	—	—	○
$P6_0$ to $P6_7$	—	○	—	○
$P7_0$, $P7_1$ (1)	—	○	—	—
$P7_6$, $P7_7$	—	○	—	○
$P8_2$	○	○	—	○
$P9_0$ to $P9_2$	—	○	—	○
$P9_3$ to $P9_6$	—	○	○	○
$P9_7$	—	○	—	○
$P11_0$ to $P11_3$	—	○	—	○
$P12_0$ to $P12_7$ $P13_0$ to $P13_7$	—	—	—	○
$P14_0$ to $P14_3$	—	○	—	○
$P15_0$	—	—	○	○
$P15_1$	—	○	○	○

○ : Available —: Not Available

NOTES:

1. $P7_0$ and $P7_1$ are ports for the N-channel open drain output.

Figure 24.3 Programmable I/O Ports (3)

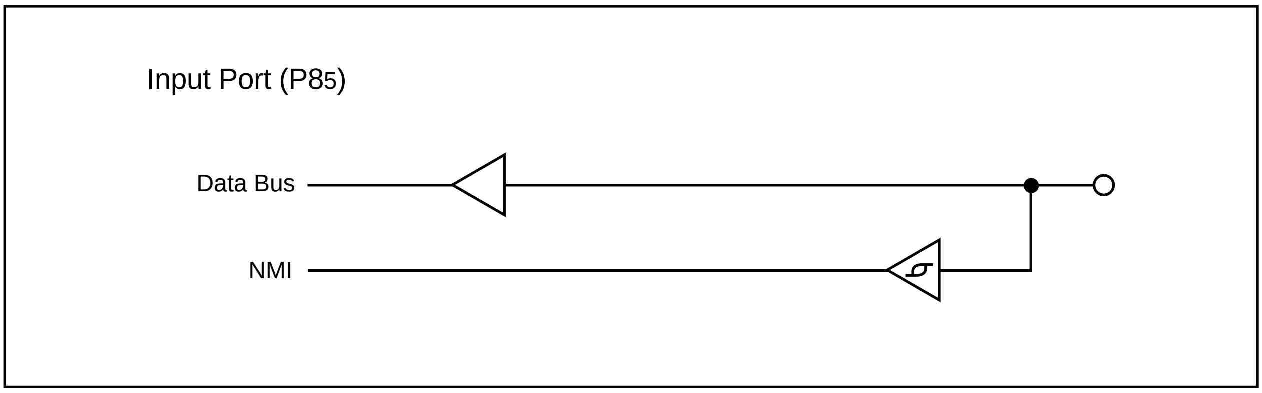
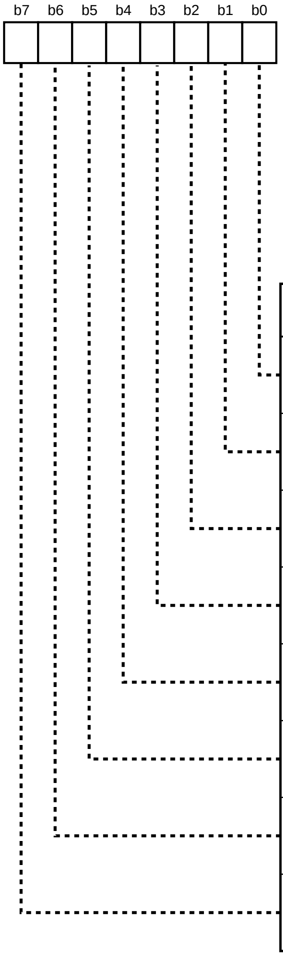


Figure 24.4 Programmable I/O Ports (4)

Port Pi Direction Register (i=0 to 15) (2)



Symbol	Address	After Reset
PD0 to PD3	$03E2_{16}$, $03E3_{16}$, $03E6_{16}$, $03E7_{16}$	00_{16}
PD4 to PD7	$03EA_{16}$, $03EB_{16}$, $03C2_{16}$, $03C3_{16}$	00_{16}
PD8	$03C6_{16}$(3)	00X0 0000_2
PD9 to PD10	$03C7_{16}$(1), $03CA_{16}$	00_{16}
PD11	$03CB_{16}$(3)	XXX0 0000_2
PD12 to PD13	03CE, $03CF_{16}$	00_{16}
PD14	$03D2_{16}$(3)	X000 0000_2
PD15	$03D3_{16}$	00_{16}

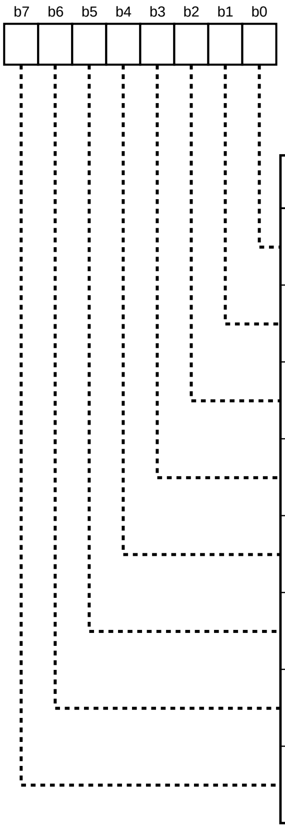
Bit Symbol	Bit Name	Function	RW
PDi_0	Port Pi$_0$ Direction Bit	0 : Input mode (Functions as input port) 1 : Output mode (Functions as output port)	RW
PDi_1	Port Pi$_1$ Direction Bit	0 : Input mode (Functions as input port) 1 : Output mode (Functions as output port)	RW
PDi_2	Port Pi$_2$ Direction Bit	0 : Input mode (Functions as input port) 1 : Output mode (Functions as output port)	RW
PDi_3	Port Pi$_3$ Direction Bit	0 : Input mode (Functions as input port) 1 : Output mode (Functions as output port)	RW
PDi_4	Port Pi$_4$ Direction Bit	0 : Input mode (Functions as input port) 1 : Output mode (Functions as output port)	RW
PDi_5	Port Pi$_5$ Direction Bit	0 : Input mode (Functions as input port) 1 : Output mode (Functions as output port)	RW
PDi_6	Port Pi$_6$ Direction Bit	0 : Input mode (Functions as input port) 1 : Output mode (Functions as output port)	RW
PDi_7	Port Pi$_7$ Direction Bit	0 : Input mode (Functions as input port) 1 : Output mode (Functions as output port)	RW

NOTES:

1. Set the PD9 register immediately after the PRC2 bit in the PRCR register is set to "1" (write enable). Do not generate an interrupt or a DMA transfer between the instruction to set the PRC2 bit to "1" and the instruction to set the PD9 register.
2. In memory expansion mode and microprocessor mode, the PDi register cannot control pins being used as bus control pins (A_0 to A_{22}, $\overline{A_{23}}$, D_0 to D_{15}, $\overline{CS0}$ to $\overline{CS3}$, $\overline{WRL}/\overline{WR}$, $\overline{WRH}/\overline{BHE}$, BCLK/ALE/$CLK_{OUT}$, $\overline{RD}$, $\overline{HLDA}$/ALE, $\overline{HOLD}$, ALE and $\overline{RDY}$).
   M32C/86T cannot be used in memory expansion mode and microprocessor mode.
3. Nothing is assigned in the PD8_5 bit in the PD8 register, the PD11_7 to PD11_5 bits in the PD11 register and the P14_7 bit in the PD14 register. If write, set these bits to "0". When read, their contents are indeterminate.

Figure 24.5 PD0 to PD15 Registers

Port Pi Register (i=0 to 15)(1, 2)



Symbol	Address	After Reset
P0 to P5	$03E0_{16}$, $03E1_{16}$, $03E4_{16}$, $03E5_{16}$, $03E8_{16}$, $03E9_{16}$	Indeterminate
P6 to P10	$03C0_{16}$, $03C1_{16}$(3), $03C4_{16}$(4), $03C5_{16}$, $03C8_{16}$	Indeterminate
P11 to P15	$03C9_{16}$(5), $03CC_{16}$, $03CD_{16}$, $03D0_{16}$(5), $03D1_{16}$	Indeterminate

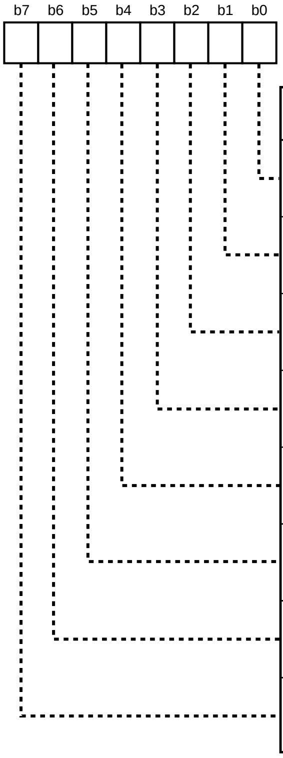
Bit Symbol	Bit Name	Function	RW
Pi_0	Port Pi_0 Bit	Pin levels can be read by reading bits corresponding to programmable ports in input mode. Pin levels can be controlled by writing to bits corresponding to programmable ports in output mode. 0: "L" level 1: "H" level	RW
Pi_1	Port Pi_1 Bit		RW
Pi_2	Port Pi_2 Bit		RW
Pi_3	Port Pi_3 Bit		RW
Pi_4	Port Pi_4 Bit		RW
Pi_5	Port Pi_5 Bit		RW
Pi_6	Port Pi_6 Bit		RW
Pi_7	Port Pi_7 Bit		RW

NOTES:

1. In memory expansion mode and microprocessor mode, the Pi register cannot control pins being used as bus control pins (A_0 to A_{22}, $\overline{A_{23}}$, D_0 to D_{15}, $\overline{CS0}$ to $\overline{CS3}$, $\overline{WRL}/\overline{WR}$, $\overline{WRH}/\overline{BHE}$, $\overline{RD}$, BCLK/ALE/CLK_{OUT}, $\overline{HLDA}$/ALE, $\overline{HOLD}$, ALE and $\overline{RDY}$).
   M32C/86T cannot be used in memory expansion mode and microprocessor mode.
2. $P7_0$ and $P7_1$ are ports for the N-channel open drain output. The pins go into high-impedance states when $P7_0$ and $P7_1$ output "H" signal.
3. The P8_5 bit is for read only.
4. Nothing is assigned in the P11_7 to P11_5 bits in the P11 register and the P14_7 bit in the P14 register. If write, set these bits to "0". When read, their contents are indeterminate.

Figure 24.6 P0 to P15 Registers

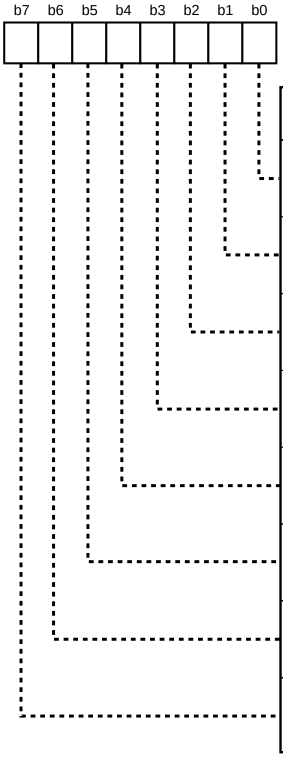
Function Select Register A0



Symbol PS0 Address $03B0_{16}$ After Reset 00_{16}

Bit Symbol	Bit Name	Function	RW
PS0_0	Port $P6_0$ Output Function Select Bit	0 : I/O port 1 : $\overline{\mathrm{RTS0}}$	RW
PS0_1	Port $P6_1$ Output Function Select Bit	0 : I/O port 1 : CLK0 output	RW
PS0_2	Port $P6_2$ Output Function Select Bit	0 : I/O port 1 : Selected by the PSL0_2 bit	RW
PS0_3	Port $P6_3$ Output Function Select Bit	0 : I/O port 1 : TxD0/SDA0 output	RW
PS0_4	Port $P6_4$ Output Function Select Bit	0 : I/O port 1 : Selected by the PSL0_4 bit	RW
PS0_5	Port $P6_5$ Output Function Select Bit	0 : I/O port 1 : CLK1 output	RW
PS0_6	Port $P6_6$ Output Function Select Bit	0 : I/O port 1 : Selected by the PSL0_6 bit	RW
PS0_7	Port $P6_7$ Output Function Select Bit	0 : I/O port 1 : TxD1/SDA1 output	RW

Function Select Register A1

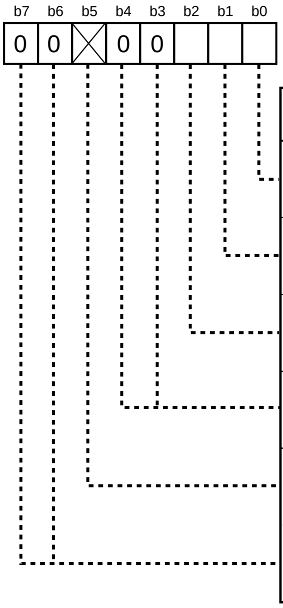


Symbol PS1 Address $03B1_{16}$ After Reset 00_{16}

Bit Symbol	Bit Name	Function	RW
PS1_0	Port $P7_0$ Output Function Select Bit	0 : I/O port 1 : Selected by the PSL1_0 bit	RW
PS1_1	Port $P7_1$ Output Function Select Bit	0 : I/O port 1 : Selected by the PSL1_1 bit	RW
PS1_2	Port $P7_2$ Output Function Select Bit	0 : I/O port 1 : Selected by the PSL1_2 bit	RW
PS1_3	Port $P7_3$ Output Function Select Bit	0 : I/O port 1 : Selected by the PSL1_3 bit	RW
PS1_4	Port $P7_4$ Output Function Select Bit	0 : I/O port 1 : Selected by the PSL1_4 bit	RW
PS1_5	Port $P7_5$ Output Function Select Bit	0 : I/O port 1 : Selected by the PSL1_5 bit	RW
PS1_6	Port $P7_6$ Output Function Select Bit	0 : I/O port 1 : Selected by the PSL1_6 bit	RW
PS1_7	Port $P7_7$ Output Function Select Bit	0 : I/O port 1 : Selected by the PSL1_7 bit	RW

Figure 24.7 PS0 Register and PS1 Register

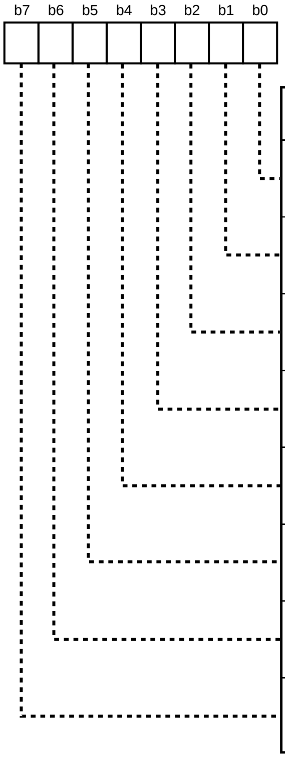
Function Select Register A2



Symbol	Address	After Reset
PS2	03B4$_{16}$	00X0 0000$_2$

Bit Symbol	Bit Name	Function	RW
PS2_0	Port P8$_0$ Output Function Select Bit	0 : I/O port 1 : Selected by the PSL2_0 bit	RW
PS2_1	Port P8$_1$ Output Function Select Bit	0 : I/O port 1 : Selected by the PSL2_1 bit	RW
PS2_2	Port P8$_2$ Output Function Select Bit	0 : I/O port 1 : Selected by the PSL2_2 bit	RW
— (b4 - b3)	Reserved Bit	Set to "0"	RW
— (b5)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		—
— (b7 - b6)	Reserved Bit	Set to "0"	RW

Function Select Register A3(1)



Symbol	Address	After Reset
PS3	03B5$_{16}$	00$_{16}$

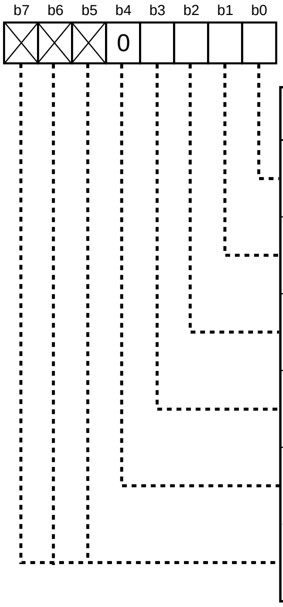
Bit Symbol	Bit Name	Function	RW
PS3_0	Port P9$_0$ Output Function Select Bit	0 : I/O port 1 : CLK3 output	RW
PS3_1	Port P9$_1$ Output Function Select Bit	0 : I/O port 1 : Selected by the PSL3_1 bit	RW
PS3_2	Port P9$_2$ Output Function Select Bit	0 : I/O port 1 : Selected by the PSL3_2 bit	RW
PS3_3	Port P9$_3$ Output Function Select Bit	0 : I/O port 1 : $\overline{\text{RTS3}}$	RW
PS3_4	Port P9$_4$ Output Function Select Bit	0 : I/O port 1 : $\overline{\text{RTS4}}$	RW
PS3_5	Port P9$_5$ Output Function Select Bit	0 : I/O port 1 : CLK4 output	RW
PS3_6	Port P9$_6$ Output Function Select Bit	0 : I/O port 1 : Selected by the PSC3_6 bit	RW
PS3_7	Port P9$_7$ Output Function Select Bit	0 : I/O port 1 : Selected by the PSL3_7 bit	RW

NOTES:

1. Set the PS3 register immediately after the PRC2 bit in the PRCR register is set to "1" (write enable). Do not generate an interrupt or a DMA transfer between the instruction to set the PRC2 bit to "1" and the instruction to set the PS3 register.

Figure 24.8 PS2 Register and PS3 Register

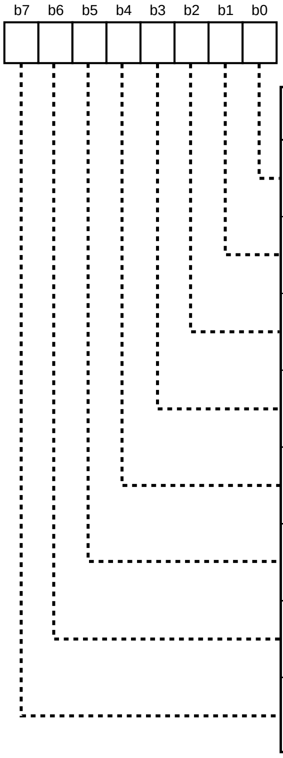
Function Select Register A5



Symbol	Address	After Reset
PS5	03B9$_{16}$	XXX0 0000$_2$

Bit Symbol	Bit Name	Function	RW
PS5_0	Port P11$_0$ Output Function Select Bit	0 : I/O port 1 : OUTC1$_0$/ IST$_X$D1/BE1$_{OUT}$	RW
PS5_1	Port P11$_1$ Output Function Select Bit	0 : I/O port 1 : OUTC1$_1$/ ISCLK1 output	RW
PS5_2	Port P11$_2$ Output Function Select Bit	0 : I/O port 1 : OUTC1$_2$	RW
PS5_3	Port P11$_3$ Output Function Select Bit	0 : I/O port 1 : OUTC1$_3$	RW
—— (b4)	Reserved Bit	Set to "0"	RW
—— (b7 - b5)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		—

Function Select Register A6

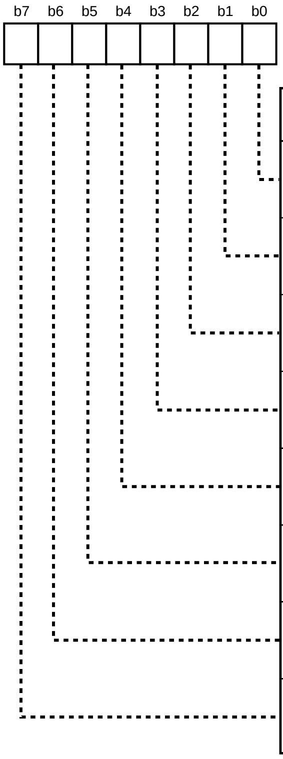


Symbol	Address	After Reset
PS6	03BC$_{16}$	00$_{16}$

Bit Symbol	Bit Name	Function	RW
PS6_0	Port P12$_0$ Output Function Select Bit	0 : I/O port 1 : Selected by the PSL6_0 bit	RW
PS6_1	Port P12$_1$ Output Function Select Bit	0 : I/O port 1 : Selected by the PSL6_1 bit	RW
PS6_2	Port P12$_2$ Output Function Select Bit	0 : I/O port 1 : Selected by the PSL6_2 bit	RW
PS6_3	Port P12$_3$ Output Function Select Bit	0 : I/O port 1 : Selected by the PSL6_3 bit	RW
PS6_4	Port P12$_4$ Output Function Select Bit	0 : I/O port 1 : Selected by the PSL6_4 bit	RW
PS6_5	Port P12$_5$ Output Function Select Bit	0 : I/O port 1 : Selected by the PSL6_5 bit	RW
PS6_6	Port P12$_6$ Output Function Select Bit	0 : I/O port 1 : Selected by the PSL6_6 bit	RW
PS6_7	Port P12$_7$ Output Function Select Bit	0 : I/O port 1 : Selected by the PSL6_7 bit	RW

Figure 24.9 PS5 Register and PS6 Register

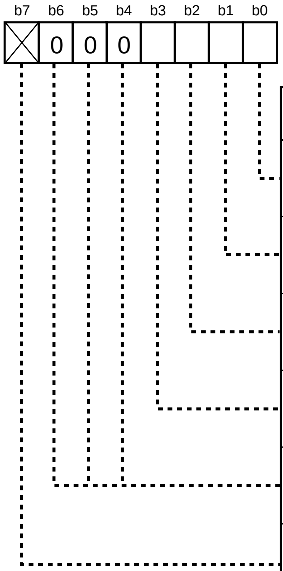
Function Select Register A7



Symbol	Address	After Reset
PS7	03BD$_{16}$	00$_{16}$

Bit Symbol	Bit Name	Function	RW
PS7_0	Port P13$_0$ Output Function Select Bit	0 : I/O port 1 : Selected by the PSL7_0 bit	RW
PS7_1	Port P13$_1$ Output Function Select Bit	0 : I/O port 1 : Selected by the PSL7_1 bit	RW
PS7_2	Port P13$_2$ Output Function Select Bit	0 : I/O port 1 : Selected by the PSL7_2 bit	RW
PS7_3	Port P13$_3$ Output Function Select Bit	0 : I/O port 1 : Selected by the PSL7_3 bit	RW
PS7_4	Port P13$_4$ Output Function Select Bit	0 : I/O port 1 : Selected by the PSL7_4 bit	RW
PS7_5	Port P13$_5$ Output Function Select Bit	0 : I/O port 1 : Selected by the PSL7_5 bit	RW
PS7_6	Port P13$_6$ Output Function Select Bit	0 : I/O port 1 : Selected by the PSL7_6 bit	RW
PS7_7	Port P13$_7$ Output Function Select Bit	0 : I/O port 1 : Selected by the PSL7_7 bit	RW

Function Select Register A8

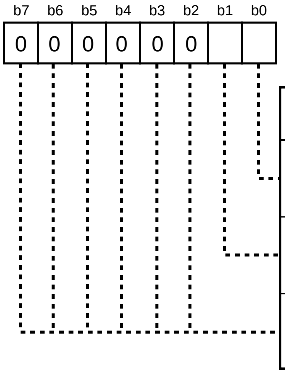


Symbol	Address	After Reset
PS8	03A0$_{16}$	X000 0000$_2$

Bit Symbol	Bit Name	Function	RW
PS8_0	Port P14$_0$ output function select bit	0 : I/O port 1 : OUTC1$_4$	RW
PS8_1	Port P14$_1$ output function select bit	0 : I/O port 1 : OUTC1$_5$	RW
PS8_2	Port P14$_2$ output function select bit	0 : I/O port 1 : OUTC1$_6$	RW
PS8_3	Port P14$_3$ output function select bit	0 : I/O port 1 : OUTC1$_7$	RW
(b6 - b4)	Reserved bit	Set to "0"	RW
(b7)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		—

Figure 24.10 PS7 Register and PS8 Register

Function Select Register A9

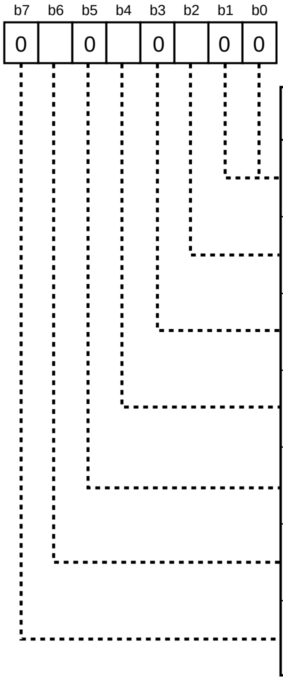


Symbol	Address	After Reset
PS9	03A1$_{16}$	00$_{16}$

Bit Symbol	Bit Name	Function	RW
PS9_0	Port P15$_0$ Output Function Select Bit	0 : I/O port 1 : ISTxD0	RW
PS9_1	Port P15$_1$ Output Function Select Bit	0 : I/O port 1 : ISCLK0 output	RW
—— (b7 - b2)	Reserved Bit	Set to "0"	RW

Figure 24.11 PS9 Register

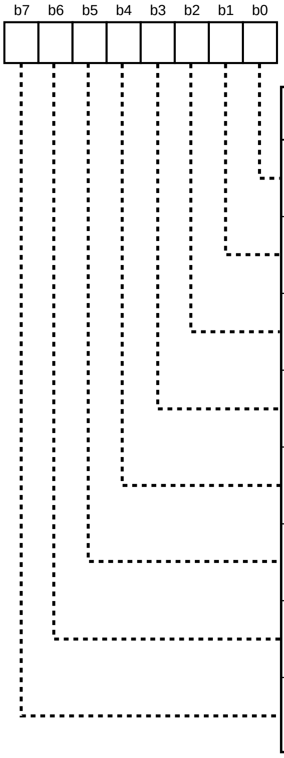
Function Select Register B0



Symbol	Address	After Reset
PSL0	03B2$_{16}$	00$_{16}$

Bit Symbol	Bit Name	Function	RW
(b1 - b0)	Reserved Bit	Set to "0"	RW
PSL0_2	Port P6$_2$ Output Peripheral Function Select Bit	0 : SCL0 output 1 : STxD0	RW
(b3)	Reserved Bit	Set to "0"	RW
PSL0_4	Port P6$_4$ Output Peripheral Function Select Bit	0 : $\overline{\text{RTS1}}$ 1 : Do not set to this value	RW
(b5)	Reserved Bit	Set to "0"	RW
PSL0_6	Port P6$_6$ Output Peripheral Function Select Bit	0 : SCL1 output 1 : STxD1	RW
(b7)	Reserved Bit	Set to "0"	RW

Function Select Register B1



Symbol	Address	After Reset
PSL1	03B3$_{16}$	00$_{16}$

Bit Symbol	Bit Name	Function	RW
PSL1_0	Port P7$_0$ Output Peripheral Function Select Bit	0 : Selected by the PSC_0 bit 1 : TA0OUT output(1)	RW
PSL1_1	Port P7$_1$ Output Peripheral Function Select Bit	0 : Selected by the PSC_1 bit 1 : STxD2(1)	RW
PSL1_2	Port P7$_2$ Output Peripheral Function Select Bit	0 : Selected by the PSC_2 bit 1 : TA1OUT output(1)	RW
PSL1_3	Port P7$_3$ Output Peripheral Function Select Bit	0 : Selected by the PSC_3 bit 1 : $\overline{\text{V}}$(1)	RW
PSL1_4	Port P7$_4$ Output Peripheral Function Select Bit	0 : Selected by the PSC_4 bit 1 : W(1)	RW
PSL1_5	Port P7$_5$ Output Peripheral Function Select Bit	0 : $\overline{\text{W}}$ 1 : OUTC1$_2$	RW
PSL1_6	Port P7$_6$ Output Peripheral Function Select Bit	0 : Selected by the PSC_6 bit 1 : TA3OUT output(1)	RW
PSL1_7	Port P7$_7$ Output Peripheral Function Select Bit	0 : ISCLK0 output 1 : OUTC1$_4$	RW

NOTES:

1. When setting the PSL1_i (i = 0 to 4, 6) bit to "1", set the corresponding PSC_i bit in the PSC register to "0".

Figure 24.12 PSL0 Register and PSL1 Register

Function Select Register B2

b7	b6	b5	b4	b3	b2	b1	b0
0	0	☒	0	0			

Symbol	Address	After Reset
PSL2	$03B6_{16}$	$00X0\ 0000_2$

Bit Symbol	Bit Name	Function	RW
PSL2_0	Port P8$_0$ Output Peripheral Function Select Bit	0 : TA4$_{OUT}$ output 1 : U	RW
PSL2_1	Port P8$_1$ Output Peripheral Function Select Bit	0 : $\overline{\text{U}}$ 1 : Selected by the PSC2_1 bit	RW
PSL2_2	Port P8$_2$ Output Peripheral Function Select Bit	0 : Do not set to this value 1 : Selected by the PSC2_2 bit	RW
— (b4 - b3)	Reserved Bit	Set to "0"	RW
— (b5)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		—
— (b7 - b6)	Reserved Bit	Set to "0"	RW

Function Select Register B3

b7	b6	b5	b4	b3	b2	b1	b0
							0

Symbol	Address	After Reset
PSL3	$03B7_{16}$	00_{16}

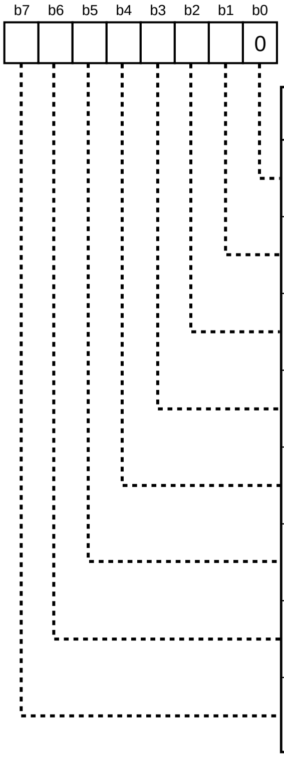
Bit Symbol	Bit Name	Function	RW
— (b0)	Reserved Bit	Set to "0"	RW
PSL3_1	Port P9$_1$ Output Peripheral Function Select Bit	0 : SCL3 output 1 : STxD3	RW
PSL3_2	Port P9$_2$ Output Peripheral Function Select Bit	0 : TxD3/SDA3 output 1 : Do not set to this value	RW
PSL3_3	Port P9$_3$ Output Peripheral Function Select Bit	0 : Except DA0 1 : DA0(1)	RW
PSL3_4	Port P9$_4$ Output Peripheral Function Select Bit	0 : Except DA1 1 : DA1(1)	RW
PSL3_5	Port P9$_5$ Output Peripheral Function Select Bit	0 : Except ANEX0 1 : ANEX0(1)	RW
PSL3_6	Port P9$_6$ Output Peripheral Function Select Bit	0 : Except ANEX1 1 : ANEX1(1)	RW
PSL3_7	Port P9$_7$ Output Peripheral Function Select Bit	0 : SCL4 output 1 : STxD4	RW

NOTES:

1. Although DA0, DA1, ANEX0 and ANEX1 can be used when this bit is set to "0", power consumption may increase.

Figure 24.13 PSL2 Register and PSL3 Register

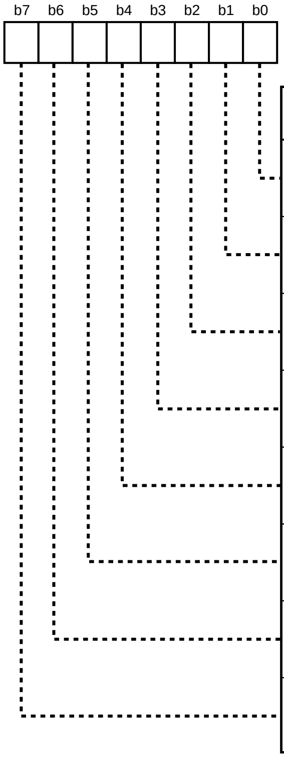
Function Select Register B6



Symbol	Address	After Reset
PSL6	03BE$_{16}$	00$_{16}$

Bit Symbol	Bit Name	Function	RW
PSL6_0	Port P12$_0$ Output Peripheral Function Select Bit	0 : Do not set to this value 1 : GASP0 output	RW
PSL6_1	Port P12$_1$ Output Peripheral Function Select Bit	0 : Do not set to this value 1 : GASM0 output	RW
PSL6_2	Port P12$_2$ Output Peripheral Function Select Bit	0 : Do not set to this value 1 : GACP0 output	RW
PSL6_3	Port P12$_3$ Output Peripheral Function Select Bit	0 : Do not set to this value 1 : GACM0 output	RW
PSL6_4	Port P12$_4$ Output Peripheral Function Select Bit	0 : Do not set to this value 1 : GASP1 output	RW
PSL6_5	Port P12$_5$ Output Peripheral Function Select Bit	0 : Do not set to this value 1 : GASM1 output	RW
PSL6_6	Port P12$_6$ Output Peripheral Function Select Bit	0 : Do not set to this value 1 : GACP1 output	RW
PSL6_7	Port P12$_7$ Output Peripheral Function Select Bit	0 : Do not set to this value 1 : GACM1 output	RW

Function Select Register B7

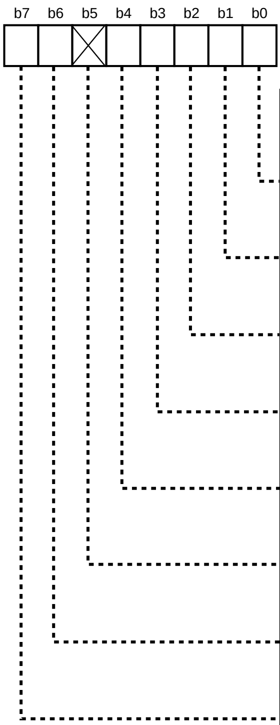


Symbol	Address	After Reset
PSL7	03BF$_{16}$	00$_{16}$

Bit Symbol	Bit Name	Function	RW
PSL7_0	Port P13$_0$ Output Peripheral Function Select Bit	0 : Do not set to this value 1 : GASP2 output	RW
PSL7_1	Port P13$_1$ Output Peripheral Function Select Bit	0 : Do not set to this value 1 : GASM2 output	RW
PSL7_2	Port P13$_2$ Output Peripheral Function Select Bit	0 : Do not set to this value 1 : GACP2 output	RW
PSL7_3	Port P13$_3$ Output Peripheral Function Select Bit	0 : Do not set to this value 1 : GACM2 output	RW
PSL7_4	Port P13$_4$ Output Peripheral Function Select Bit	0 : Do not set to this value 1 : GASP3 output	RW
PSL7_5	Port P13$_5$ Output Peripheral Function Select Bit	0 : Do not set to this value 1 : GASM3 output	RW
PSL7_6	Port P13$_6$ Output Peripheral Function Select Bit	0 : Do not set to this value 1 : GACP3 output	RW
PSL7_7	Port P13$_7$ Output Peripheral Function Select Bit	0 : Do not set to this value 1 : GACM3 output	RW

Figure 24.14 PSL6 Register and PSL7 Register

Function Select Register C



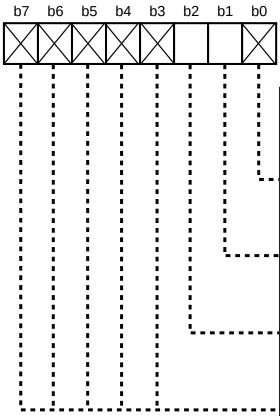
Symbol	Address	After Reset
PSC	$03AF_{16}$	$00X0\ 0000_2$

Bit Symbol	Bit Name	Function	RW
PSC_0	Port $P7_0$ Output Peripheral Function Select Bit	0 : TxD2/SDA2 output 1 : Selected by the PSD1_0 bit	RW
PSC_1	Port $P7_1$ Output Peripheral Function Select Bit	0 : SCL2 output 1 : Selected by the PSD1_1 bit	RW
PSC_2	Port $P7_2$ Output Peripheral Function Select Bit	0 : CLK2 output 1 : V	RW
PSC_3	Port $P7_3$ Output Peripheral Function Select Bit	0 : $\overline{RTS2}$ 1 : $OUTC1_0$/$ISTxD1$/$BE1_{OUT}$	RW
PSC_4	Port $P7_4$ Output Peripheral Function Select Bit	0 : $TA2_{OUT}$ output 1 : $OUTC1_1$/ISCLK1	RW
(b5)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		—
PSC_6	Port $P7_6$ Output Peripheral Function Select Bit	0 : Selected by the PSD1_6 bit 1 : $CAN0_{OUT}$	RW
PSC_7	Key Input Interrupt Disabled Select Bit	0 : $P10_4$ to $P10_7$ or $\overline{KI_0}$ to $\overline{KI_3}$ 1 : AN_4 to AN_7(1)	RW

NOTES:

1. Set the ILVL2 to ILVL0 bits in the the KUPIC register to "000_2" (interrupt disabled) when changing the PSC_7 bit setting.
   Although AN_4 to AN_7 can be used when this bit is set to "0", power consumption may increase.

Function Select Register C2

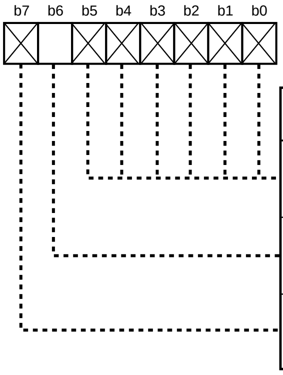


Symbol	Address	After Reset
PSC2	$03AC_{16}$	$XXXX\ X00X_2$

Bit Symbol	Bit Name	Function	RW
(b0)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		—
PSC2_1	Port $P8_1$ Output Peripheral Function Select Bit	0 : Do not set to this value 1 : $OUTC1_5$	RW
PSC2_2	Port $P8_2$ Output Peripheral Function Select Bit	0 : $CAN0_{OUT}$ 1 : $CAN1_{OUT}$	RW
(b7 - b3)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		—

Figure 24.15 PSC Register and PSC2 Register

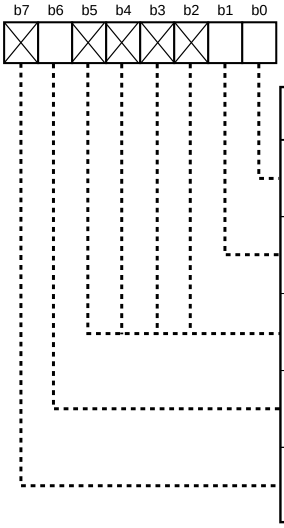
Function Select Register C3



Symbol	Address	After Reset
PSC3	$03AD_{16}$	$X0XX\ XXXX_2$

Bit Symbol	Bit Name	Function	RW
—— (b5 - b0)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		—
PSC3_6	Port $P9_6$ Output Peripheral Function Select Bit	0 : TxD4/SDA4 output 1 : $CAN1_{OUT}$	RW
—— (b7)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		—

Function Select Register D1



Symbol	Address	After Reset
PSD1	$03A7_{16}$	$X0XX\ XX00_2$

Bit Symbol	Bit Name	Function	RW
PSD1_0	Port $P7_0$ Output Peripheral Function Select Bit	0 : Do not set to this value 1 : $OUTC1_6$	RW
PSD1_1	Port $P7_1$ Output Peripheral Function Select Bit	0 : Do not set to this value 1 : $OUTC1_7$	RW
—— (b5 - b2)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		—
PSD1_6	Port $P7_6$ Output Peripheral Function Select Bit	0 : ISTxD0 1 : $OUTC1_3$	RW
—— (b7)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		—

Figure 24.16 PSC3 Register and PSD1 Register

Pull-Up Control Register 0[1]

b7 b6 b5 b4 b3 b2 b1 b0

Symbol	Address	After Reset
PUR0	$03F0_{16}$	00_{16}

Bit Symbol	Bit Name	Function	RW
PU00	$P0_0$ to $P0_3$ Pull-Up	Pull-up setting for corresponding port 0 : Not pulled up 1 : Pulled up	RW
PU01	$P0_4$ to $P0_7$ Pull-Up		RW
PU02	$P1_0$ to $P1_3$ Pull-Up		RW
PU03	$P1_4$ to $P1_7$ Pull-Up		RW
PU04	$P2_0$ to $P2_3$ Pull-Up		RW
PU05	$P2_4$ to $P2_7$ Pull-Up		RW
PU06	$P3_0$ to $P3_3$ Pull-Up		RW
PU07	$P3_4$ to $P3_7$ Pull-Up		RW

NOTES:

1. Set each bit in the PUR0 register to "0" when ports P0 to P5 become bus control pins in memory expansion mode and microprocessor mode. When using the ports as I/O ports, pull-up or no pull-up setting can be selected.
   M32C/86T cannot be used in memory expansion mode and microprocessor mode.

Pull-Up Control Register 1[1]

b7 b6 b5 b4 b3 b2 b1 b0

Symbol	Address	After Reset
PUR1	$03F1_{16}$	$\text{XXXX }0000_2$

Bit Symbol	Bit Name	Function	RW
PU10	$P4_0$ to $P4_3$ Pull-Up	Pull-up setting for corresponding port 0 : Not pulled up 1 : Pulled up	RW
PU11	$P4_4$ to $P4_7$ Pull-Up		RW
PU12	$P5_0$ to $P5_3$ Pull-Up		RW
PU13	$P5_4$ to $P5_7$ Pull-Up		RW
____ (b7 - b4)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		___

NOTES:

1. Set each bit in the PUR1 register to "0" when ports P0 to P5 become bus control pins in memory expansion mode and microprocessor mode. When using the ports as I/O ports, pull-up or no pull-up setting can be selected.
   M32C/86T cannot be used in memory expansion mode and microprocessor mode.

Pull-Up Control Register 2

b7 b6 b5 b4 b3 b2 b1 b0

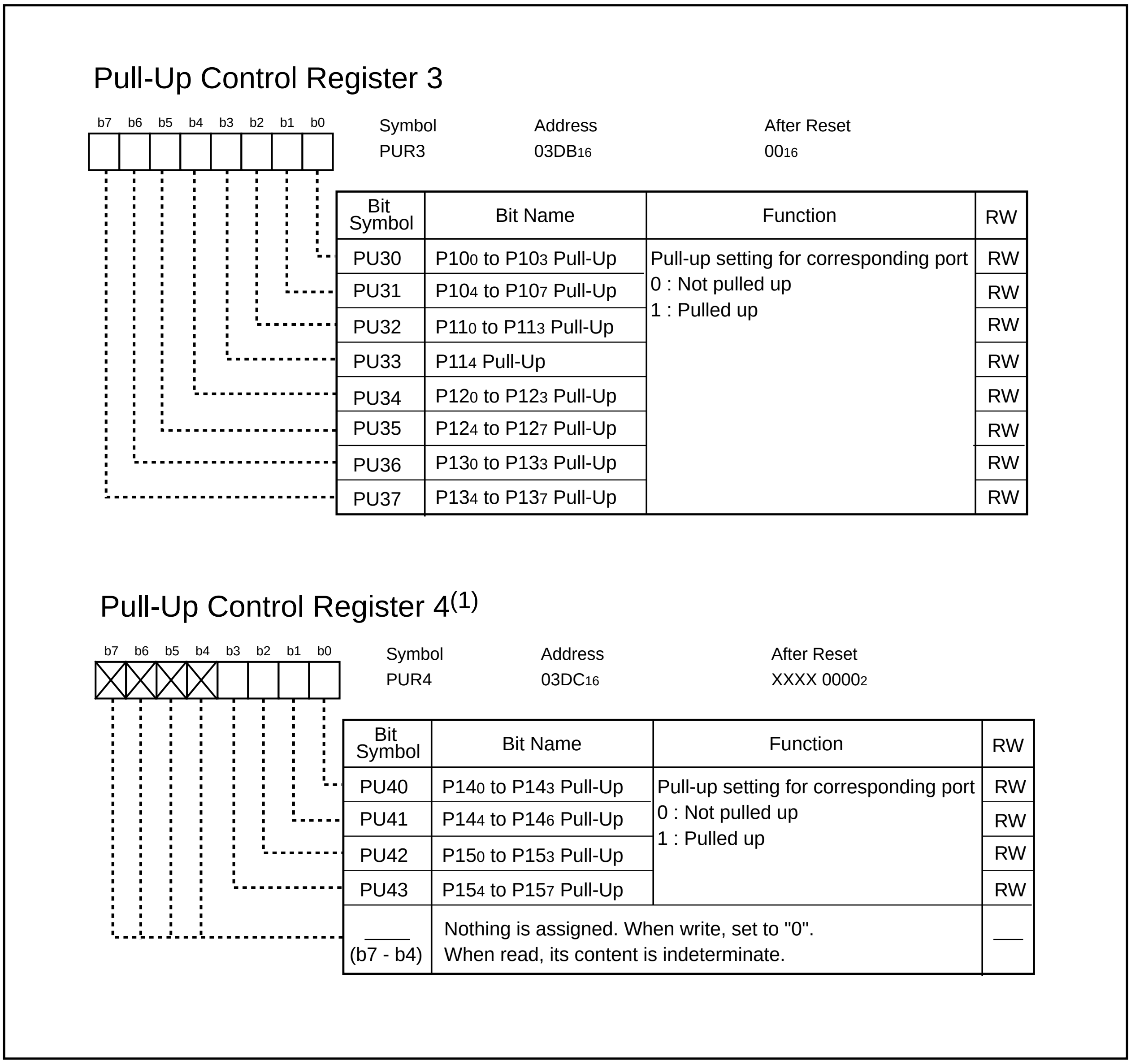
Symbol	Address	After Reset
PUR2	$03DA_{16}$	00_{16}

Bit Symbol	Bit Name	Function	RW
PU20	$P6_0$ to $P6_3$ Pull-Up	Pull-up setting for corresponding port 0 : Not pulled up 1 : Pulled up	RW
PU21	$P6_4$ to $P6_7$ Pull-Up		RW
PU22	$P7_2$ to $P7_3$ Pull-Up[1]		RW
PU23	$P7_4$ to $P7_7$ Pull-Up		RW
PU24	$P8_0$ to $P8_3$ Pull-Up		RW
PU25	$P8_4$ to $P8_7$ Pull-Up[2]		RW
PU26	$P9_0$ to $P9_3$ Pull-Up		RW
PU27	$P9_4$ to $P9_7$ Pull-Up		RW

NOTES:

1. $P7_0$ and $P7_1$ cannot be pulled up.
2. $P8_5$ cannot be pulled up.

Figure 24.17 PUR0 Register, PUR1 Register and PUR2 Register



Pull-Up Control Register 3

Symbol	Address	After Reset
PUR3	03DB$_{16}$	00$_{16}$

Bit Symbol	Bit Name	Function	RW
PU30	P10$_0$ to P10$_3$ Pull-Up	Pull-up setting for corresponding port 0 : Not pulled up 1 : Pulled up	RW
PU31	P10$_4$ to P10$_7$ Pull-Up		RW
PU32	P11$_0$ to P11$_3$ Pull-Up		RW
PU33	P11$_4$ Pull-Up		RW
PU34	P12$_0$ to P12$_3$ Pull-Up		RW
PU35	P12$_4$ to P12$_7$ Pull-Up		RW
PU36	P13$_0$ to P13$_3$ Pull-Up		RW
PU37	P13$_4$ to P13$_7$ Pull-Up		RW

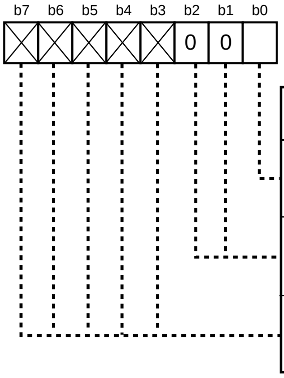
Pull-Up Control Register 4(1)

Symbol	Address	After Reset
PUR4	03DC$_{16}$	XXXX 0000$_2$

Bit Symbol	Bit Name	Function	RW
PU40	P14$_0$ to P14$_3$ Pull-Up	Pull-up setting for corresponding port 0 : Not pulled up 1 : Pulled up	RW
PU41	P14$_4$ to P14$_6$ Pull-Up		RW
PU42	P15$_0$ to P15$_3$ Pull-Up		RW
PU43	P15$_4$ to P15$_7$ Pull-Up		RW
—— (b7 - b4)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		—

Figure 24.18 PUR3 Register and PUR4 Register

Port Control Register[1]



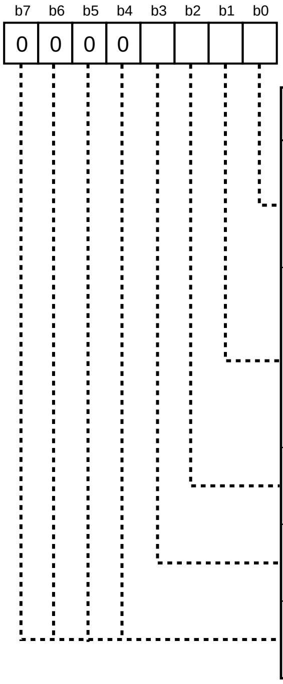
Symbol	Address	After Reset
PCR	$03FF_{16}$	$XXXX\ XXX0_2$

Bit Symbol	Bit Name	Function	RW
PCR0	Port P1 Control Bit	0 : CMOS output 1 : N-channel open drain output[2]	RW
—— (b2 - b1)	Reserved Bit	Set to "0"	RW
—— (b7 - b3)	Nothing is assigned. When write, set to "0". When read, its content is indeterminate.		—

NOTES:

1. Set the PCR0 bit to "0" when P1 operates as a data bus in memory expansion mode and microprocessor mode. When using the ports as I/O ports, CMOS port or N-channel open drain output port can be selected.
   M32C/86T cannot be used in memory expansion mode and microprocessor mode.
2. This function is designed not to make port P1 a full open drain but to turn off the P channel in the CMOS port.
   Absolute maximum rating of the input voltage is between -0.3V and V_{CC} + 0.3V.

Input Function Select Register



Symbol	Address	After Reset
IPS	0178_{16}	00_{16}

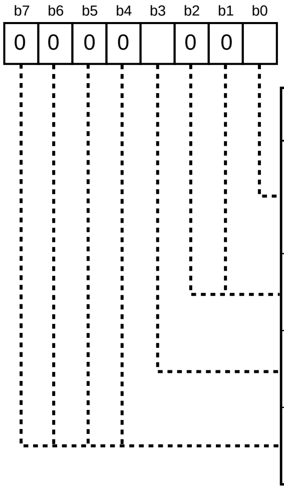
Bit Symbol	Bit Name	Function	RW
IPS0	Communication Unit 0 Input Pin Select Bit 0	Assigns each function of ISCLK0 and ISRxD0 to the following ports. 0 : $P7_7$, $P8_0$ 1 : $P15_1$, $P15_2$	RW
IPS1	Communication Unit 1 Input Pin Select Bit 1	Assigns each function of $INPC1_0$, $INPC1_1$/ISCLK1, $INPC1_2$/ISRxD1/$BE1_{IN}$, $INPC1_3$, $INPC1_4$, $INPC1_5$, $INPC1_6$ and $INPC1_7$ to the following ports. 0 : $P7_3$, $P7_4$, $P7_5$, $P7_6$, $P7_7$, $P8_1$, $P7_0$, $P7_1$ 1 : $P11_0$, $P11_1$, $P11_2$, $P11_3$, $P14_0$, $P14_1$, $P14_2$, $P14_3$	RW
IPS2	Port P15 Input Peripheral Function Select Bit	0 : Except AN15[1] 1 : AN15	RW
IPS3	$CAN0_{IN}$ Function Pin Select Bit	0 : $P7_7$ 1 : $P8_3$	RW
—— (b7 - b4)	Reserved Bit	Set to "0"	RW

NOTES:

1. Although $AN15_0$ to $AN15_7$ can be used when the IPS2 bit is set to "0", power consumption may increase.

Figure 24.19 PCR Register and IPS Register

Input Function Select Register A



Symbol	Address	After Reset
IPSA	0179_{16}	00_{16}

Bit Symbol	Bit Name	Function	RW
IPSA_0	Intelligent I/O Two-Phase Pulse Input Pin and Base Timer Reset Pin Switch Bit	0 : $P8_0$, $P8_1$ used as two-phase pulse input pin, $\overline{\text{INT1}}$ as base timer reset 1 : $P7_6$, $P7_7$ used as two-phase pulse input pin, $\overline{\text{INT0}}$ as base timer reset	RW
(b2 - b1)	Reserved Bit	Set to "0"	RW
IPSA_3	CAN1$_{IN}$ Function Pin Select Bit	0 : $P9_5$ 1 : $P8_3$	RW
(b7 - b4)	Reserved Bit	Set to "0"	RW

Figure 24.20 IPSA Register

Table 24.1 Unassigned Pin Settings in Single-Chip Mode

Pin Name	Setting
P0 to P15 (excluding P85)(1,2,3)	Enter input mode and connect each pin to VSS via a resistor (pull-down); or enter output mode and leave the pins open
XOUT(5)	Leave pin open
$\overline{\text{NMI}}$(P85)	Connect pin to VCC via a resistor (pull-up)
AVCC	Connect pin to VCC
AVSS, VREF, BYTE	Connect pins to VSS

NOTES:

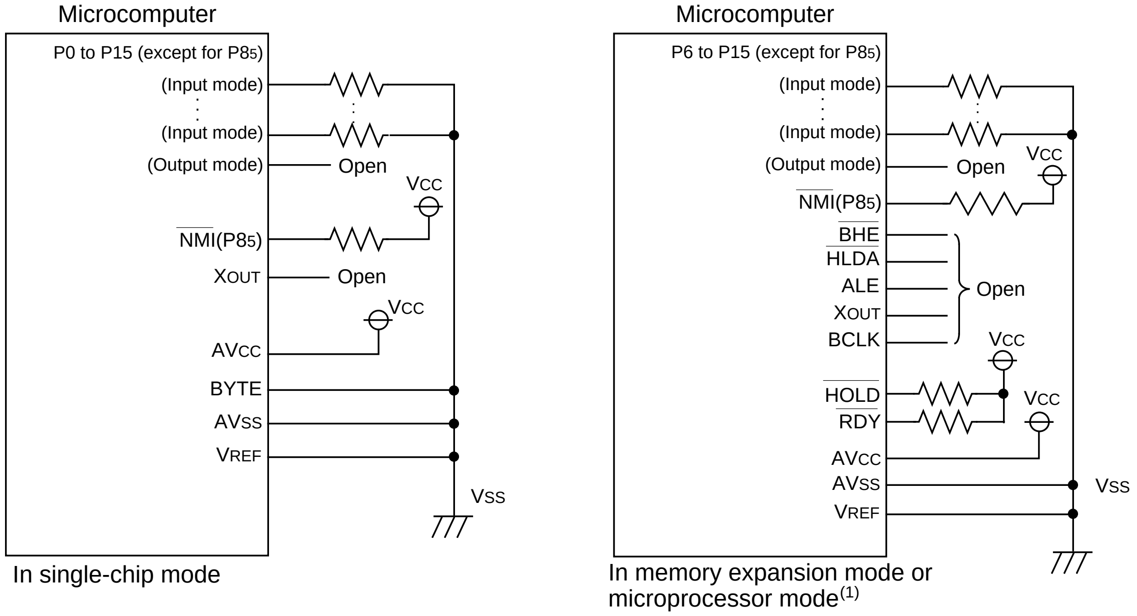
1. If the port enters output mode and is left open, it is in input mode before output mode is entered by program after reset. While the port is in input mode, voltage level on the pins is indeterminate and power consumption may increase.
   Direction register settings may be changed by noise or failure caused by noise. Configure direction register settings regulary to increase the reliability of the program.
2. Use the shortest possible wiring to connect the microcomputer pins to unassigned pins (within 2 cm).
3. P70 and P71 must put in low-level ("L") signal outputs if they are in output mode. They are ports for the N-channel open-drain output.
4. When the external clock is applied to the XIN pin, set the pin as written above.

Table 24.2 Unassigned Pin Setting in Memory Expansion Mode and Microprocessor Mode

Pin Name	Setting
P6 to P15 (excluding P85)(1,2,3)	Enter input mode and connect each pin to VSS via a resistor (pull-down); or enter output mode and leave the pins open
BHE, ALE, $\overline{\text{HLDA}}$, XOUT(5), BCLK	Leave pin open
$\overline{\text{RDY}}$, $\overline{\text{HOLD}}$, $\overline{\text{NMI}}$(P85)	Connect pins to VCC via a resistor (pull-up)
AVCC	Connect pin to VCC
AVSS, VREF	Connect pins to VSS

NOTES:

1. If the port enters output mode and is left open, it is in input mode before output mode is entered by program after reset. While the port is in input mode, voltage level on the pins is indeterminate and power consumption may increase.
   Direction register settings may be changed by noise or failure caused by noise. Configure direction register settings regulary to increase the reliability of the program.
2. Use the shortest possible wiring to connect the microcomputer pins to unassigned pins (within 2 cm).
3. P70 and P71 must put in low-level ("L") signal outputs if they are in output mode. They are ports for the N-channel open-drain output.
4. When the external clock is applied to the XIN pin, set the pin as written above.



NOTES:
1. M32C/86T cannot be used in memory expansion mode and microprocessor mode.

Figure 24.21 Unassigned Pin Handling

Table 24.3 Port P6 Peripheral Function Output Control

	PS0 Register	PSL0 Register
Bit 0	0: P6$_0$/$\overline{\text{CTS0}}$/$\overline{\text{SS0}}$ 1: $\overline{\text{RTS0}}$	Set to "0"
Bit 1	0: P6$_1$/CLK0(input) 1: CLK0(output)	Set to "0"
Bit 2	0: P6$_2$/RxD0/SCL0(input) 1: Selected by the PSL0 register	0: SCL0(output) 1: STxD0
Bit 3	0: P6$_3$/SRxD0/SDA0 (input) 1: TxD0/SDA0 (output)	Set to "0"
Bit 4	0: P6$_4$/$\overline{\text{CTS1}}$/$\overline{\text{SS1}}$ 1: Selected by the PSL0 register	0: $\overline{\text{RTS1}}$ 1: Do not set this value
Bit 5	0: P6$_5$/CLK1(input) 1: CLK1(output)	Set to "0"
Bit 6	0: P6$_6$/RxD1/SCL1(input) 1: Selected by the PSL0 register	0: SCL1(output) 1: STxD1
Bit 7	0: P6$_7$/SRxD1/SDA1 (input) 1: TxD1/SDA1 (output)	Set to "0"

Table 24.4 Port P7 Peripheral Function Output Control

	PS1 Register	PSL1 Register	PSC Register[1]	PSD1 Register
Bit 0	0: P7$_0$/TA0$_{OUT}$(input)/SRxD2 INPC1$_6$/SDA2 (input) 1: Selected by the PSL1 register	0: Selected by the PSC register 1: TA0$_{OUT}$(output)	0: TxD2/SDA2(output) 1: Selected by the PSD1 register	0: Do not set to this value 1: OUTC1$_6$
Bit 1	0: P7$_1$/TB5$_{IN}$/TA0$_{IN}$/RxD2/ INPC1$_7$/SCL2 (input) 1: Selected by the PSL1 register	0: Selected by the PSC register 1: STxD2	0: SCL2(output) 1: Selected by the PSD1 register	0: Do not set to this value 1: OUTC1$_7$
Bit 2	0: P7$_2$/TA1$_{OUT}$(input)/ CLK2(input) 1: Selected by the PSL1 register	0: Selected by the PSC register 1: TA1$_{OUT}$(output)	0: CLK2(output) 1: V	Set to "0"
Bit 3	0: P7$_3$/TA1$_{IN}$/$\overline{\text{CTS2}}$/$\overline{\text{SS2}}$/ INPC1$_0$ 1: Selected by the PSL1 register	0: Selected by the PSC register 1: $\overline{\text{V}}$	0: $\overline{\text{RTS2}}$ 1: OUTC1$_0$/ISTxD1/BE1$_{OUT}$	Set to "0"
Bit 4	0: P7$_4$/INPC1$_1$/ISCLK1(input)/ TA2$_{OUT}$(input) 1: Selected by the PSL1 register	0: Selected by the PSC register 1: W	0: TA2$_{OUT}$(output) 1: OUTC1$_1$/ISCLK1(output)	Set to "0"
Bit 5	0: P7$_5$/TA2$_{IN}$/INPC1$_2$/ ISRxD1/BE1$_{IN}$ 1: Selected by the PSL1 register	0: $\overline{\text{W}}$ 1: OUTC1$_2$	Set to "0"	Set to "0"
Bit 6	0: P7$_6$/INPC1$_3$/TA3$_{OUT}$(input) 1: Selected by the PSL1 register	0: Selected by the PSC register 1: TA3$_{OUT}$(output)	0: Selected by the PSD1 register 1: CAN0$_{OUT}$	0: ISTxD0 1: OUTC1$_3$
Bit 7	0: P7$_7$/TA3$_{IN}$/CAN0$_{IN}$/ ISCLK0(input)/INPC1$_4$ 1: Selected by the PSL1 register	0: ISCLK0(output) 1: OUTC1$_4$	0: P10$_4$ to P10$_7$ or KI0 to KI3 1: AN4 to AN$_7$ (No relation to P7$_7$)	Set to "0"

NOTES:

1. When setting the PSL1_i bit (i=0 to 4, 6) to "1", set the corresponding PSC_i bit to "0".

Table 24.5 Port P8 Peripheral Function Output Control

	PS2 Register	PSL2 Register	PSC2 Register
Bit 0	0: P8$_0$/ISRxD0/TA4$_{OUT}$(input) 1: Selected by the PSL2 register	0: TA4$_{OUT}$(output) 1: U	Set to "0"
Bit 1	0: P8$_1$/TA4$_{IN}$/INPC1$_5$ 1: Selected by the PSL2 register	0: $\overline{\text{U}}$ 1: Selected by the PSC2 register	0: Do not set to this value 1: OUTC1$_5$
Bit 2	0: P8$_2$/$\overline{\text{INT0}}$ 1: Selected by the PSL2 register	0: Do not set to this value 1: Selected by the PSC2 register	0: CAN0$_{OUT}$ 1: CAN1$_{OUT}$
Bit 3 to 7	Set to "00000$_2$"		

Table 24.6 Port P9 Peripheral Function Output Control

	PS3 Register	PSL3 Register	PSC3 Register
Bit 0	0: P9$_0$/TB0$_{IN}$/CLK3(input) 1: CLK3(output)	Set to "0"	Set to "0"
Bit 1	0: P9$_1$/TB1$_{IN}$/RxD3/SCL3(input) 1: Selected by the PSL3 register	0: SCL3(output) 1: STxD3	Set to "0"
Bit 2	0: P9$_2$/TB2$_{IN}$/SRxD3/SDA3(input) 1: Selected by the PSL3 register	0: TxD3/SDA3(output) 1: Do not set to this value	Set to "0"
Bit 3	0: P9$_3$/TB3$_{IN}$/$\overline{\text{CTS3}}$/$\overline{\text{SS3}}$/DA0(output) 1: $\overline{\text{RTS3}}$	0: Except DA0 1: DA0	Set to "0"
Bit 4	0: P9$_4$/TB4$_{IN}$/$\overline{\text{CTS4}}$/$\overline{\text{SS4}}$/DA1(output) 1: $\overline{\text{RTS4}}$	0: Except DA1 1: DA1	Set to "0"
Bit 5	0: P9$_5$/ANEX0/CLK4(input)/CAN1$_{IN}$/ $\overline{\text{CAN1WU}}$ 1: CLK4(output)	0: Except ANEX0 1: ANEX0	Set to "0"
Bit 6	0: P9$_6$/SRxD4/ANEX1/SDA4(input) 1: Selected by the PSC3 register	0: Except ANEX1 1: ANEX1	0: TxD4/SDA4 1: CAN1$_{OUT}$
Bit 7	0: P9$_7$/RxD4/$\overline{\text{AD}_{\text{TRG}}}$/SCL4(input) 1: Selected by the PSL3 register	0: SCL4(output) 1: STxD4	Set to "0"

Table 24.7 Port P10 Peripheral Function Output Control

	PSC Register
Bit 7	0: P10$_4$ to P10$_7$ or $\overline{\text{KI0}}$ to $\overline{\text{KI3}}$ 1: AN$_4$ to AN$_7$

Table 24.8 Port P11 Peripheral Function Output Control

	PS5 Register
Bit 0	0: P11$_0$/INPC1$_0$ 1: OUTC1$_0$/ISTxD1/BE1OUT
Bit 1	0: P11$_1$/INPC1$_1$/ISCLK1(input) 1: OUTC1$_1$/ISCLK1(output)
Bit 2	0: P11$_2$/INPC1$_2$/ISRxD1/BE1$_{IN}$ 1: OUTC1$_2$
Bit 3	0: P11$_3$/INPC1$_3$ 1: OUTC1$_3$
Bit 4 to 7	Set to "0000$_2$"

Table 24.9 Port P12 Peripheral Function Output Control

	PS6 Register	PSL6 Register
Bit 0	0: P12$_0$ 1: Selected by the PSL6 register	0: Do not set this value 1: GASP0
Bit 1	0: P12$_1$ 1: Selected by the PSL6 register	0: Do not set this value 1: GASM0
Bit 2	0: P12$_2$ 1: Selected by the PSL6 register	0: Do not set this value 1: GACP0
Bit 3	0: P12$_3$ 1: Selected by the PSL6 register	0: Do not set this value 1: GACM0
Bit 4	0: P12$_4$ 1: Selected by the PSL6 register	0: Do not set this value 1: GASP1
Bit 5	0: P12$_5$ 1: Selected by the PSL6 register	0: Do not set this value 1: GASM1
Bit 6	0: P12$_6$ 1: Selected by the PSL6 register	0: Do not set this value 1: GACP1
Bit 7	0: P12$_7$ 1: Selected by the PSL6 register	0: Do not set this value 1: GACM1

Table 24.10 Port P13 Peripheral Function Output Control

	PS7 Register	PSL7 Register
Bit 0	0: P13$_0$ 1: Selected by the PSL7 register	0: Do not set this value 1: GASP2
Bit 1	0: P13$_1$ 1: Selected by the PSL7 register	0: Do not set this value 1: GASM2
Bit 2	0: P13$_2$ 1: Selected by the PSL7 register	0: Do not set this value 1: GACP2
Bit 3	0: P13$_3$ 1: Selected by the PSL7 register	0: Do not set this value 1: GACM2
Bit 4	0: P13$_4$ 1: Selected by the PSL7 register	0: Do not set this value 1: GASP3
Bit 5	0: P13$_5$ 1: Selected by the PSL7 register	0: Do not set this value 1: GASM3
Bit 6	0: P13$_6$ 1: Selected by the PSL7 register	0: Do not set this value 1: GACP3
Bit 7	0: P13$_7$ 1: Selected by the PSL7 register	0: Do not set this value 1: GACM3

Table 24.11 Port P14 Peripheral Function Output Control

	PS8 Register
Bit 0	0: P14$_0$/INPC1$_4$ 1: OUTC1$_4$
Bit 1	0: P14$_1$/INPC1$_5$ 1: OUTC1$_5$
Bit 2	0: P14$_2$/INPC1$_6$ 1: OUTC1$_6$
Bit 3	0: P14$_3$/INPC1$_7$ 1: OUTC1$_7$
Bit 4 to 7	Set to "0000$_2$"

Table 24.12 Port P15 Peripheral Function Output Control

	PS9 Register
Bit 0	0: P15$_0$/AN15$_0$ 1: ISTxD0
Bit 1	0: P15$_1$/AN15$_1$/ISCLK0(input) 1: ISCLK0(output)
Bit 2 to 7	Set to "000000$_2$"

25. Flash Memory Version

Aside from the built-in flash memory, the flash memory version microcomputer has the same functions as the masked ROM version.

In the flash memory version, rewrite operation to the flash memory can be performed in three modes: CPU rewrite mode, standard serial I/O mode and parallel I/O mode.

Table 25.1 lists specifications of the flash memory version. See **Tables 1.1 and 1.2** for the items not listed in Table 25.1.

Table 25.1 Flash Memory Version Specifications

Item		Specification
Flash Memory Operating Mode		3 modes (CPU rewrite, standard serial I/O, parallel I/O)
Erase Block	User ROM Area	See Figure 25.1
	Boot ROM Area	1 block (4 Kbytes)(1)
Program Method		Per word (16 bytes), per byte (8 bits)(2)
Erase Method		All block erase, erase per block
Program and Erase Control Method		Software commands control programming and erasing on the flash memory
Protect Method		The lock bit protects each block in the flash memory
Number of Commands		8 commands
Program and Erase Endurance		100 times(3)
Data Retention		10 years
ROM Code Protection		Standard serial I/O mode and parallel I/O mode supported

NOTES:

1. The rewrite control program for standard serial I/O mode is stored in the boot ROM area before shipment. This space can be rewritten in parallel I/O mode only.
2. Programming per byte is available in parallel I/O mode only.
3. Program and erase endurance refers to the number of times a block erase can be performed. Every block erase performed after writing data of one word or more counts as one program and erase operation.

Table 25.2 Flash Memory Rewrite Mode Overview

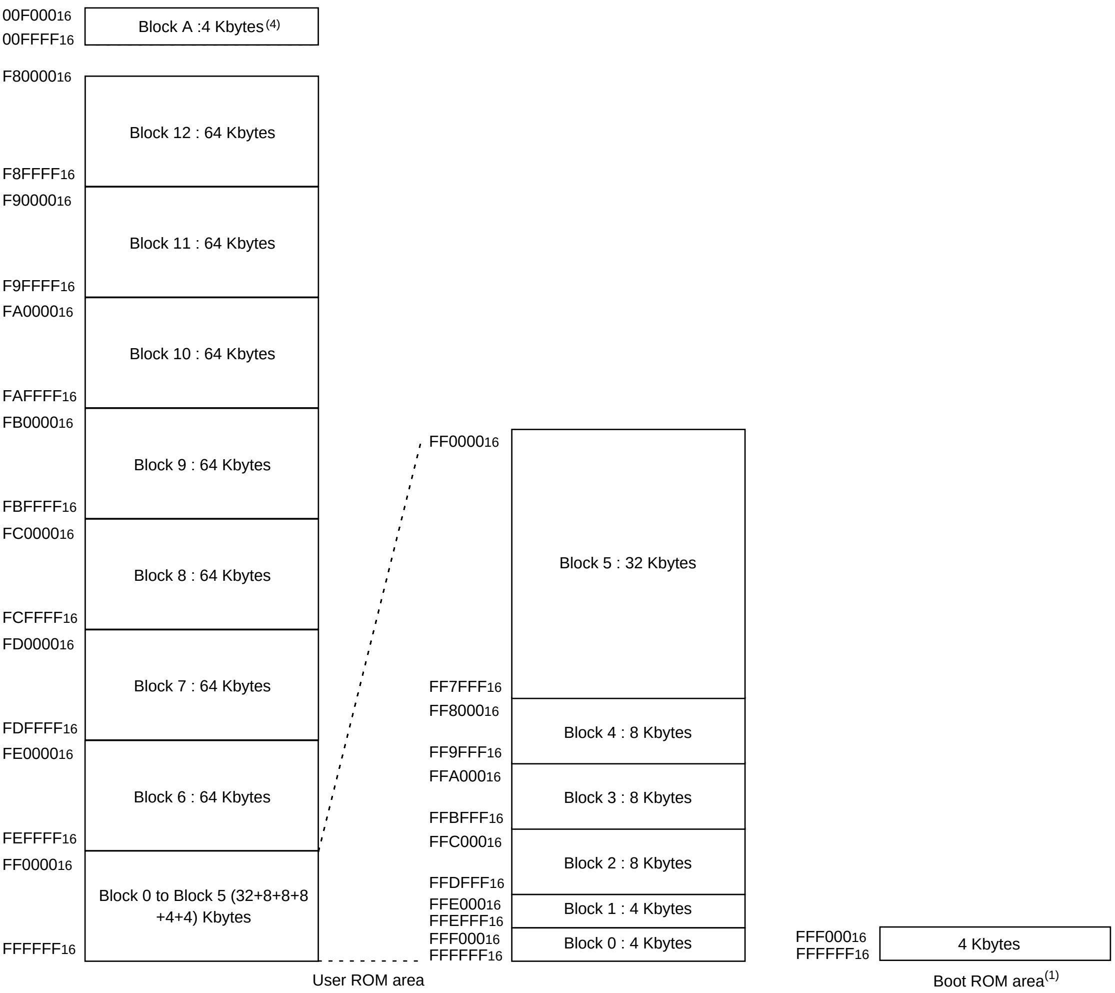
Flash Memory Rewrite Mode	CPU Rewrite Mode	Standard Serial I/O Mode	Parallel I/O Mode
Function	Software command execution by CPU rewrites the user ROM area. EW mode 0: Rewritable in areas other than flash memory EW mode 1: Rewritable in flash memory	A dedicated serial programmer rewrites the user ROM area. Standard serial I/O mode 1: Clock synchronous serial I/O Standard serial I/O mode 2: UART Standard serial I/O mode 3: CAN	A dedicated parallel programmer rewrites the boot ROM area and user ROM area.
Rewritable Space	User ROM area	User ROM area	User ROM area Boot ROM area
Operating Mode	Single-chip mode Memory expansion mode (EW mode 0) Boot mode (EW mode 0)	Boot mode	Parallel I/O mode
Programmer	None	Serial programmer	Parallel programmer

25.1 Memory Map

The flash memory includes the user ROM area and the boot ROM area. The user ROM area has space to store the microcomputer operating programs in single-chip mode or memory expansion mode, and a separate 4-kbyte space as the block A. Figure 25.1 shows a block diagram of the flash memory.

The user ROM area is divided into several blocks, each of which can be protected (locked) from program or erase. The user ROM area can be rewritten in CPU rewrite mode, standard serial I/O mode and parallel I/O mode.

The boot ROM area is located at the same addresses as the user ROM area. It can only be rewritten in parallel I/O mode. A program in the boot ROM area is executed after a hardware reset occurs while a high-level ("H") signal is applied to the CNVSS and P50 pins and a low-level ("L") signal is applied to the P55 pin. A program in the user ROM area is executed after a hardware reset occurs while an "L" signal is applied to the CNVSS pin. Consequently, the boot ROM area cannot be read.



NOTES:
1. The boot ROM area can be rewritten in parallel I/O mode only.
2. When specifying a block, use an even address in the block to be specified.
3. Shown here is a flash memory block diagram in single-chip mode.
4. The block A cannot be erased by the all erase unlocked block command. Use the block erase command to erase.

Figure 25.1 Flash Memory Block Diagram

25.1.1 Boot Mode

The microcomputer enters boot mode when a hardware reset is performed while a high-level ("H") signal is applied to the CNVSS and $P5_0$ pins and a low-level ("L") signal is applied to the $P5_5$ pin. A program in the boot ROM area is executed.

In boot mode, the FMR05 bit in the FMR0 register selects access to either the boot ROM area or the user ROM area.

In the factory setting, the rewrite control program for standard serial I/O mode is stored into the boot ROM area.

The boot ROM area can be rewritten in parallel I/O mode only. If any rewrite control program using erase-write mode 0 (EW mode 0) is written in the boot ROM area, the flash memory can be rewritten according to the system implemented.

25.2 Functions to Prevent the Flash Memory from Rewriting

The flash memory has the ROM code protect function for parallel I/O mode and the ID code verify function for standard I/O mode to prevent the flash memory from reading or rewriting.

25.2.1 ROM Code Protect Function

The ROM code protect function prevents the flash memory from reading and rewriting in parallel I/O mode.

Figure 25.2 shows the ROMCP register. The ROMCP register is located in the user ROM area.

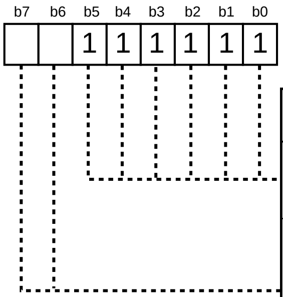
The ROM code protect function is enabled when the ROMCP1 bit is set to "00_2", "01_2" or "10_2".

25.2.2 ID Code Verify Function

Use the ID code verify function in standard serial I/O mode. The ID code sent from the serial programmer is compared with the ID code written in the flash memory for a match. If the ID codes do not match, commands sent from the serial programmer are not accepted. However, if the four bytes of the reset vector are "$FFFFFFFF_{16}$", ID codes are not compared, allowing all commands to be accepted.

The ID codes are 7-byte data stored consecutively, starting with the first byte, into addresses $0FFFFDF_{16}$, $0FFFFE3_{16}$, $0FFFFEB_{16}$, $0FFFFEF_{16}$, $0FFFFF3_{16}$, $0FFFFF7_{16}$ and $0FFFFFB_{16}$. The flash memory must have a program with the ID codes set in these addresses.

ROM Code Protect Control Address[5]



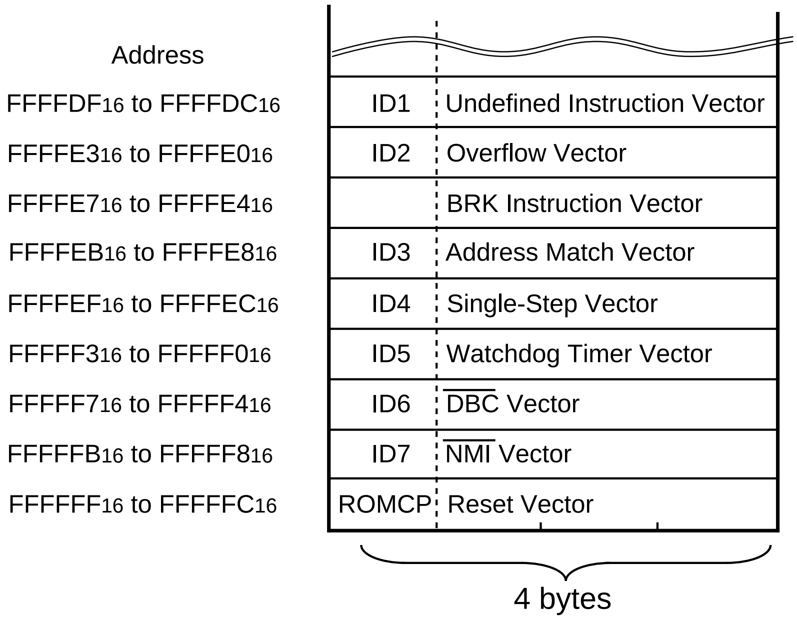
Symbol	Address	Factory Setting
ROMCP	$FFFFFF_{16}$	FF_{16}[4]

Bit Symbol	Bit Name	Function	RW
—— (b5 - b0)	Reserved Bit	Set to "1"	RW
ROMCP1	ROM Code Protect Level 1 Set Bit[1, 2, 3]	b7 b6 0 0 : ROM code protection active 0 1 : ROM code protection active 1 0 : ROM code protection active 1 1 : ROM code protection inactive	RW

NOTES:

1. When the ROM code protection is active by the ROMCP1 bit setting, the flash memory is protected against reading or rewriting in parallel I/O mode.
2. Set the bit 5 to bit 0 to "111111_2" when the ROMCP1 bit is set to a value other than "11_2". If the bit 5 to bit 0 are set to values other than "111111_2", the ROM code protection may not become active by setting the ROMCP1 bit to a value other than "11_2".
3. To make the ROM code protection inactive, erase a block including the ROMCP address in standard serial I/O mode or CPU rewrite mode.
4. The ROMCP address is set to "FF_{16}" when a block, including the ROMCP address, is erased.
5. When a value of the ROMCP address is "00_{16}" or "FF_{16}", the ROM code protect function is disabled.

Figure 25.2 ROMCP Address



Address		
$FFFFDF_{16}$ to $FFFFDC_{16}$	ID1	Undefined Instruction Vector
$FFFFE3_{16}$ to $FFFFE0_{16}$	ID2	Overflow Vector
$FFFFE7_{16}$ to $FFFFE4_{16}$		BRK Instruction Vector
$FFFFEB_{16}$ to $FFFFE8_{16}$	ID3	Address Match Vector
$FFFFEF_{16}$ to $FFFFEC_{16}$	ID4	Single-Step Vector
$FFFFF3_{16}$ to $FFFFF0_{16}$	ID5	Watchdog Timer Vector
$FFFFF7_{16}$ to $FFFFF4_{16}$	ID6	$\overline{\text{DBC}}$ Vector
$FFFFFB_{16}$ to $FFFFF8_{16}$	ID7	$\overline{\text{NMI}}$ Vector
$FFFFFF_{16}$ to $FFFFFC_{16}$	ROMCP	Reset Vector

Figure 25.3 Address for ID Code Stored

25.3 CPU Rewrite Mode

In CPU rewrite mode, the user ROM area can be rewritten when the CPU executes software commands. The user ROM area can be rewritten with the microcomputer mounted on a board without using a parallel or serial programmer.

In CPU rewrite mode, only the user ROM area shown in Figure 25.1 can be rewritten. The boot ROM area cannot be rewritten. The program and block erase commands are executed only for each block in the user ROM area.

Erase-write (EW) mode 0 and erase-write mode 1 are provided as CPU rewrite mode. Table 25.3 lists differences between EW mode 0 and EW mode 1.

Table 25.3 EW Mode 0 and EW Mode 1

Item	EW mode 0	EW mode 1
Operating Mode	• Single-chip mode • Memory expansion mode • Boot mode	Single-chip mode
Space where the rewrite control program can be placed	• User ROM area • Boot ROM area	User ROM area
Space where the rewrite control program can be executed	The rewrite control program must be transferred to any space other than the flash memory (e.g.,RAM) before being executed	The rewrite control program can be executed in the user ROM area
Space which can be rewritten	User ROM area	User ROM area However, this excludes blocks with the rewrite control program
Software Command Restriction	None	• Program and block erase commands cannot be executed in a block having the rewrite control program. • Erase all unlocked block command cannot be executed when the lock bit in a block having the rewrite control program is set to "1"(unlocked) or when the FMR02 bit in the FMR0 register is set to "1"(lock bit disabled). • Read status register command cannot be used.
Mode after Programming or Erasing	Read status register mode	Read array mode
CPU State during Auto Program and Erase Operation	Operating	In a hold state (I/O ports maintains the state before the command was executed)(1)
Flash Memory State Detection	• Read the FMR00, FMR06 and FMR07 bits in the FMR0 register by program • Execute the read status register command to read the SR7, SR5 and SR4 bits in the SRD register	Read the FMR00, FMR06 and FMR07 bits in the FMR0 register by program

NOTES:

1. Do not generate an interrupt (except $\overline{\text{NMI}}$ interrupt) or a DMA transfer.

25.3.1 EW Mode 0

The microcomputer enters CPU rewrite mode by setting the FMR01 bit in the FMR0 register to "1" (CPU rewrite mode enabled) and is ready to accept commands. EW mode 0 is selected by setting the FMR11 bit in the FMR1 register to "0". To set the FMR01 bit to "1", set to "1" after first writing "0".

The software commands control programming and erasing. The FMR0 register or the SRD register indicates whether a program or erase operation is completed as expected or not.

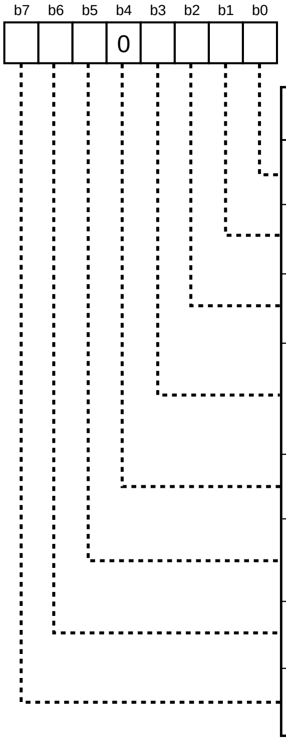
25.3.2 EW Mode 1

EW mode 1 is selected by setting the FMR11 bit to "1" after the FMR01 bit is set to "1". (Both bits must be set to "0" first before setting to "1".)

The FMR0 register indicates whether or not a program or erase operation has been completed as expected. The SRD register cannot be read in EW mode 1.

25.3.3 Flash Memory Control Register (FMR0 Register and FMR1 Register)

Flash Memory Control Register 0



Symbol	Address	After Reset
FMR0	0057_{16}	$0000\ 0001_2$

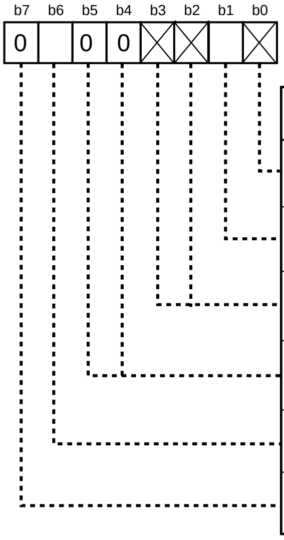
Bit Symbol	Bit Name	Function	RW
FMR00	RY/$\overline{\text{BY}}$ Status Flag	0 : BUSY (programming or erasing)(6) 1 : READY	RO
FMR01	CPU Rewrite Mode Select Bit(1, 7)	0 : Disables CPU rewrite mode 1 : Enables CPU rewrite mode	RW
FMR02	Lock Bit Disable Select Bit(2)	0 : Enables the lock bit 1 : Disables the lock bit	RW
FMSTP	Flash Memory Stop Bit(3, 5)	0 : Starts the flash memory 1 : Stops the flash memory (Enters low power consumption state and flash memory is reset)	RW
(b4)	Reserved Bit	Set to "0"	RW
FMR05	User ROM Area Select Bit(3) (Available in boot mode only)	0 : Boot ROM area is accessed 1 : User ROM area is accessed	RW
FMR06	Program Status Flag(4)	0 : Successfully completed 1 : Terminated by error	RO
FMR07	Erase Status Flag(4)	0 : Successfully completed 1 : Terminated by error	RO

NOTES:

1. Set the FMR01 bit while the $\overline{\text{NMI}}$ pin is held "H". Set it by program in a space other than the flash memory in EW mode 0.
2. Set the FMR02 bit to "1" in 8-bit unit immediately after setting it first to "0" while the FMR01 bit is set to "1". Do not generate an interrupt or a DMA transfer between setting the FMR02 bit to "0" and setting it to "1".
3. Set the FMSTP and FMR05 bits by program in a space other than the flash memory.
4. The FMR07 and FMR06 bits is set to "0" by executing the clear status command.
5. FMSTP bit setting is enabled when the FMR01 bit is set to "1" (CPU rewrite mode enabled). The FMSTP bit can be set to "1" when the FMR01 bit is set to "0", but the flash memory does not enter low-power consumption state nor is reset.
6. Write and read operations by the lock bit program command and read lock bit status command are included.
7. **To change a FMR01 bit setting from "0" to "1"**, set the FMR01 bit to "1" immediately after setting it first to "0" in 8-bit unit. Do not generate an interrupt or a DMA transfer between setting the FMR01 bit to "0" and setting it to "1".
   To change a FMR01 bit setting from "1" to "0", enter read array mode to write to addresses 0057_{16} in 16-bit unit. Write "00_{16}" into 8 high-order bits.
   e. g., to change a FMR01 bit setting from "1" to "0";
   Assembly language: mov.w #0000h, 0057h

Figure 25.4 FMR0 Register

Flash Memory Control Register 1



Symbol	Address	After Reset
FMR1	0055_{16}	$0000\ 0101_2$

Bit Symbol	Bit Name	Function	RW
(b0)	Reserved Bit	When read, its content is indeterminate	RO
FMR11	EW Mode Select Bit(1)	0 : EW mode 0 1 : EW mode 1	RW
(b3 - b2)	Reserved Bit	When read, its content is indeterminate	RO
(b5 - b4)	Reserved Bit	Set to "0"	RW
FMR16	Lock Bit Status Flag	0 : Locked 1 : Unlocked	RO
(b7)	Reserved Bit	Set to "0"	RW

NOTES:

1. Set the FMR11 bit to "1" in 8-bit unit immediately after setting it first to "0" while the FMR01 bit is set to "1". Do not generate an interrupt or a DMA transfer between setting the FMR11 bit to "0" and setting it to "1". Set it while the $\overline{\text{NMI}}$ pin is held "H". If the FMR01 bit is set to "0", the FMR01 bit and FMR11 bit are both set to "0".

Figure 25.5 FMR1 Register

25.3.3.1 FMR00 Bit

The FMR00 bit indicates the flash memory operating state. It is set to "0" while the program, block erase, erase all unlocked block, lock bit program, or read lock bit status command is being executed; otherwise, it is set to "1".

25.3.3.2 FMR01 Bit

The microcomputer can accept commands when the FMR01 bit is set to "1" (CPU rewrite mode). Set the FMR05 bit to "1" (user ROM area access) as well if in boot mode.

25.3.3.3 FMR02 Bit

The lock bit is invalid by setting the FMR02 bit to "1" (lock bit disabled). (Refer to **25.3.6 Data Protect Function**.) The lock bit is valid by setting the FMR02 bit to "0" (lock bit enabled).

The FMR02 bit does not change the lock bit status but disables the lock bit function. If the block erase or erase all unlocked block command is executed when the FMR02 bit is set to "1", the lock bit status changes "0" (locked) to "1" (unlocked) after command execution is completed.

25.3.3.4 FMSTP Bit

The FMSTP bit initializes the flash memory control circuits and minimizes power consumption in the flash memory. Access to the flash memory is disabled when the FMSTP bit is set to "1". Set the FMSTP bit by program in a space other than the flash memory.
Set the FMSTP bit to "1" if one of the followings occurs:

- A flash memory access error occurs while erasing or programming in EW mode 0 (FMR00 bit does not switch back to "1" (ready)).
- Low-power consumption mode or on-chip low-power consumption mode is entered.

Use the following the procedure to change the FMSTP bit setting.
(1) Set the FMSTP bit to "1"
(2) Set tps (the wait time to stabilize flash memory circuit)
(3) Set the FMSTP bit to "0"
(4) Set tps (the wait time to stabilize flash memory circuit)

Figure 25.8 shows a flow chart illustrating how to start and stop the flash memory before and after entering low power mode. Follow the procedure on this flow chart.
When entering stop or wait mode, the flash memory is automatically turned off. When exiting stop or wait mode, the flash memory is turned back on. The FMR0 register does not need to be set.

25.3.3.5 FMR05 Bit

The FMR05 bit selects the boot ROM or user ROM area in boot mode. Set to "0" to access (read) the boot ROM area or to "1" (user ROM access) to access (read, write or erase) the user ROM area.

25.3.3.6 FMR06 Bit

The FMR06 bit is a read-only bit indicating an auto program operation state. The FMR06 bit is set to "1" when a program error occurs; otherwise, it is set to "0". Refer to **25.3.8 Full Status Check**.

25.3.3.7 FMR07 Bit

The FM07 bit is a read-only bit indicating the auto erase operation state. The FMR07 bit is set to "1" when an erase error occurs; otherwise, it is set to “0”. For details, refer to **25.3.8 Full Status Check**.

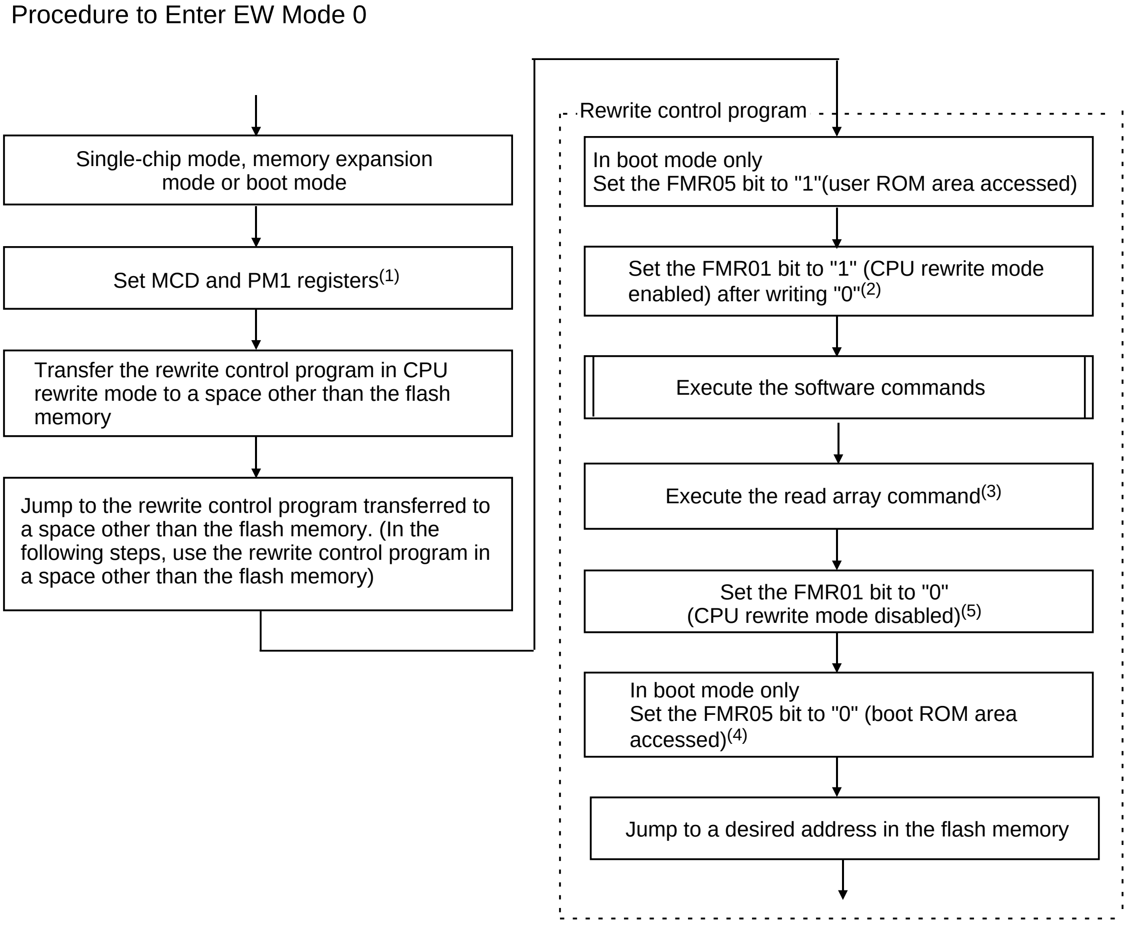
Figure 25.6 shows how to enter and exit EW mode 0. Figure 25.7 shows how to enter and exit EW mode 1.

25.3.3.8 FMR11 Bit

EW mode 0 is entered by setting the FMR11 bit to "0" (EW mode 0).
EW mode 1 is entered by setting the FMR11 bit to "1" (EW mode 1).

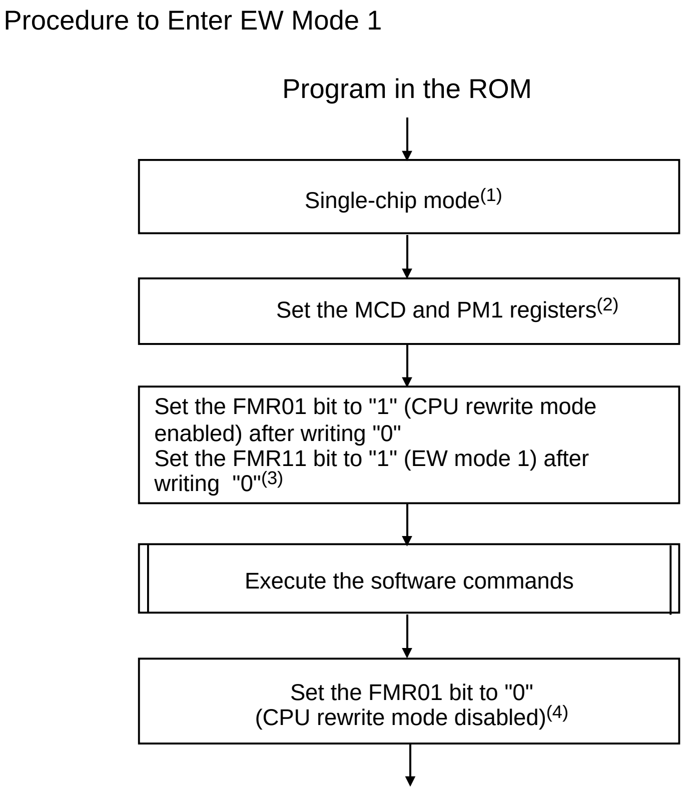
25.3.3.9 FMR16 Bit

The FMR16 bit is a read-only bit indicating the execution result of the read lock bit status command.
When the block, where the read lock bit status command is executed, is locked, the FMR16 bit is set to "0".
When the block, where the read lock bit status command is executed, is unlocked, the FMR16 bit is set to "1".


Procedure to Enter EW Mode 0
Single-chip mode, memory expansion mode or boot mode
Set MCD and PM1 registers(1)
Transfer the rewrite control program in CPU rewrite mode to a space other than the flash memory
Jump to the rewrite control program transferred to a space other than the flash memory. (In the following steps, use the rewrite control program in a space other than the flash memory)
Rewrite control program
In boot mode only
Set the FMR05 bit to "1"(user ROM area accessed)
Set the FMR01 bit to "1" (CPU rewrite mode enabled) after writing "0"(2)
Execute the software commands
Execute the read array command(3)
Set the FMR01 bit to "0"
(CPU rewrite mode disabled)(5)
In boot mode only
Set the FMR05 bit to "0" (boot ROM area accessed)(4)
Jump to a desired address in the flash memory

NOTES:
1. In CPU rewrite mode, set the MCD register to be the 10-MHz CPU clock frequency or less and set the PM12 bit in the PM1 register to "1" (internal access wait).
2. To set the FMR01 bit to "1", set it to "1" in 8-bit unit immediately after setting it first to "0". Do not generate an interrupt or a DMA transfer between setting the bit to "0" and setting it to "1". Set the FMR01 bit in a space other than flash memory. Set the FMR01 bit while the $\overline{\text{NMI}}$ pin is held "H".
3. Exit CPU rewrite mode after executing the read array command.
4. When the FMR05 bit is set to "1", the user ROM area can be accessed.
5. To change the FMR01 bit setting from "1" to "0", enter read array mode to write to addresses 0057_{16} in 16-bit unit. Write "00_{16}" into 8 high-order bits.

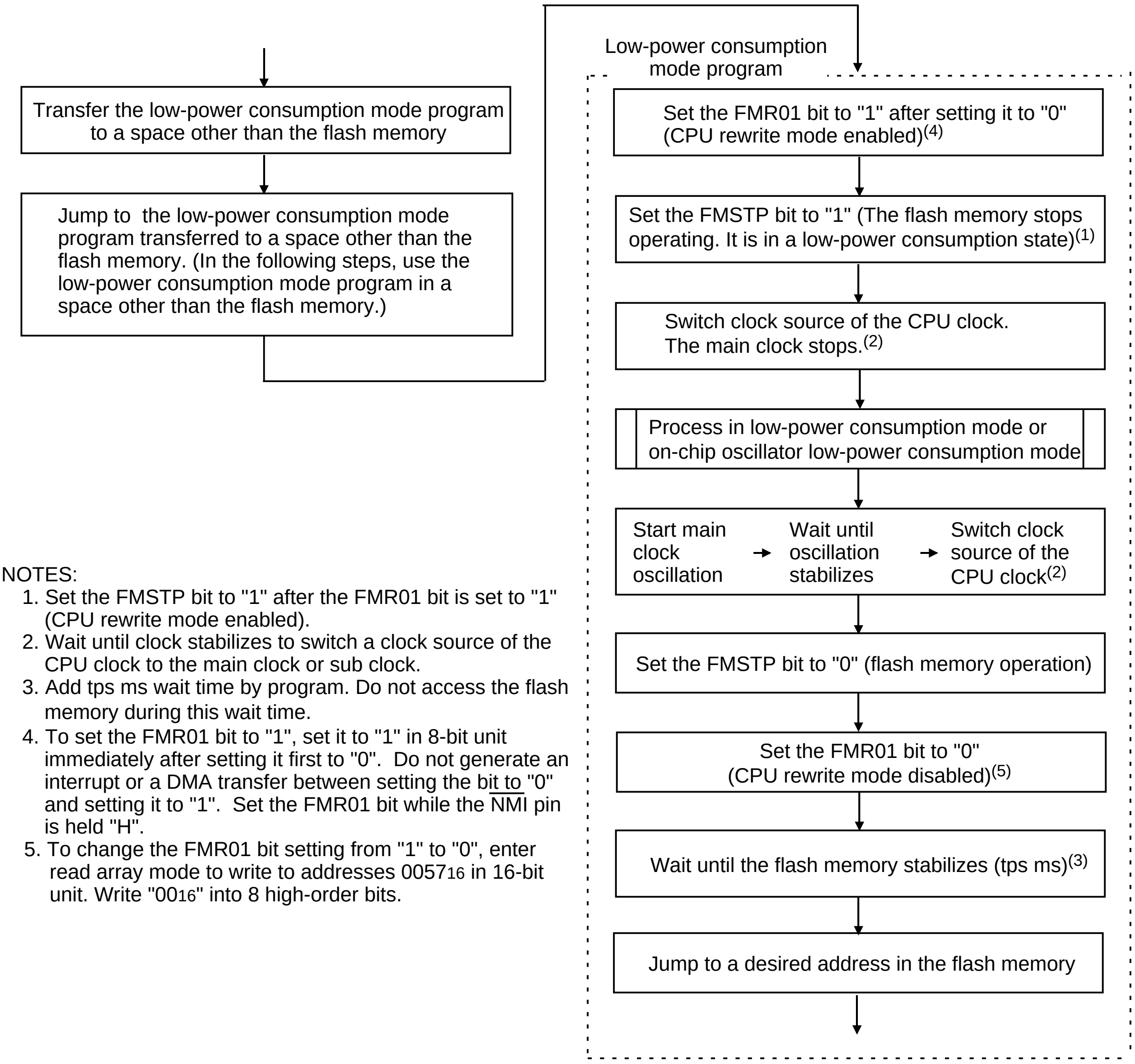
Figure 25.6 How to Enter and Exit EW Mode 0


Procedure to Enter EW Mode 1
Program in the ROM
Single-chip mode(1)
Set the MCD and PM1 registers(2)
Set the FMR01 bit to "1" (CPU rewrite mode enabled) after writing "0"
Set the FMR11 bit to "1" (EW mode 1) after writing "0"(3)
Execute the software commands
Set the FMR01 bit to "0"
(CPU rewrite mode disabled)(4)

NOTES:

1. In EW mode 1, do not enter memory expansion or boot mode.
2. In CPU rewrite mode, set the MCD register to be the 10-MHz CPU clock frequency or less. Set the PM12 bit in the PM1 register to "1" (internal access wait).
3. To set the FMR01 bit to "1", set it to "1" in 8-bit unit immediately after setting it first to "0". Do not generate an interrupt or a DMA transfer between setting the FMR01 bit to "0" and setting it to "1". To set the FMR11 bit to "1", set it to "1" in 8-bit unit immediately after setting it first to "0". Do not generate an interrupt or a DMA transfer between setting the FMR11 bit to "0" and setting it to "1". Set the FMR01 and FMR11 bits while the $\overline{\text{NMI}}$ pin is held "H".
4. To change the FMR01 bit setting from "1" to "0", enter read array mode to write to addresses 0057_{16} in 16-bit unit. Write "00_{16}" into 8 high-order bits.

Figure 25.7 How to Enter and Exit EW Mode 1


Transfer the low-power consumption mode program to a space other than the flash memory
Jump to the low-power consumption mode program transferred to a space other than the flash memory. (In the following steps, use the low-power consumption mode program in a space other than the flash memory.)
Low-power consumption mode program
Set the FMR01 bit to "1" after setting it to "0" (CPU rewrite mode enabled)(4)
Set the FMSTP bit to "1" (The flash memory stops operating. It is in a low-power consumption state)(1)
Switch clock source of the CPU clock. The main clock stops.(2)
Process in low-power consumption mode or on-chip oscillator low-power consumption mode
Start main clock oscillation
Wait until oscillation stabilizes
Switch clock source of the CPU clock(2)
Set the FMSTP bit to "0" (flash memory operation)
Set the FMR01 bit to "0" (CPU rewrite mode disabled)(5)
Wait until the flash memory stabilizes (tps ms)(3)
Jump to a desired address in the flash memory

NOTES:

1. Set the FMSTP bit to "1" after the FMR01 bit is set to "1" (CPU rewrite mode enabled).
2. Wait until clock stabilizes to switch a clock source of the CPU clock to the main clock or sub clock.
3. Add tps ms wait time by program. Do not access the flash memory during this wait time.
4. To set the FMR01 bit to "1", set it to "1" in 8-bit unit immediately after setting it first to "0". Do not generate an interrupt or a DMA transfer between setting the bit to "0" and setting it to "1". Set the FMR01 bit while the NMI pin is held "H".
5. To change the FMR01 bit setting from "1" to "0", enter read array mode to write to addresses 0057$_{16}$ in 16-bit unit. Write "00$_{16}$" into 8 high-order bits.

Figure 25.8 Handling Before and After Low Power Consumption Mode

25.3.4 Precautions in CPU Rewrite Mode

25.3.4.1 Operating Speed

Set the MCD4 to MCD0 bits in the MCD register to CPU clock frequency of 10 MHz or less before entering CPU rewrite mode (EW mode 0 or EW mode 1). Also, set the PM12 bit in the PM1 register to "1" (wait state).

25.3.4.2 Prohibited Instructions

The following instructions cannot be used in EW mode 0 because the CPU tries to read data in the flash memory: the UND instruction, INTO instruction, JMPS instruction, JSRS instruction, and BRK instruction.

25.3.4.3 Interrupts (EW Mode 0)

- To use interrupts having vectors in a relocatable vector table, the vectors must be relocated to the RAM area.
- The $\overline{\text{NMI}}$ and watchdog timer interrupts are available since the FMR0 and FMR1 registers are forcibly reset when either interrupt occurs. Allocate the forward addresses for each interrupt routine to the fixed vector table. Flash memory rewrite operation is aborted when the $\overline{\text{NMI}}$ or watchdog timer interrupt occurs. Execute the rewrite program again after exiting the interrupt routine.
- The address match interrupt is not available since the CPU tries to read data in the flash memory.

25.3.4.4 Interrupts (EW Mode 1)

- Do not acknowledge any interrupts with vectors in the relocatable vector table or address match interrupt during the auto program or auto erase period.
- Do not use the watchdog timer interrupt.
- The $\overline{\text{NMI}}$ interrupt is available since the FMR0 and FMR1 registers are forcibly reset when either interrupt occurs. Allocate the forward address for the interrupt routine to the fixed vector table. Flash memory rewrite operation is aborted when the $\overline{\text{NMI}}$ interrupt occurs. Execute the rewrite program again after exiting the interrupt routine.

25.3.4.5 How to Access

To set the FMR01, FMR02 in the FMR0 register or FMR11 bit in the FMR1 register to "1", set to "1" in 8-bit units immediately after setting to "0". Do not generate an interrupt or a DMA transfer between the instruction to set the bit to "0" and the instruction to set the bit to "1". Set the bit while a high-level ("H") signal is applied to the $\overline{\text{NMI}}$ pin.

To change the FMR01 bit from "1" to "0", enter read array mode first, and write into address 0057_{16} in 16-bit units. Eight high-order bits must be set to "00_{16}".

25.3.4.6 Rewriting in the User ROM Area (EW Mode 0)

If the supply voltage drops while rewriting the block where the rewrite control program is stored, the flash memory cannot be rewritten because the rewrite control program is not rewritten as expected. If this error occurs, rewrite the user ROM area while in standard serial I/O mode or parallel I/O mode.

25.3.4.7 Rewriting in the User ROM Area (EW Mode 1)

Do not rewrite the block where the rewrite control program is stored.

25.3.4.8 DMA Transfer

In EW mode 1, do not generate a DMA transfer while the FMR00 bit in the FMR0 register is set to "0" (busy-programming or erasing).

25.2.4.9 Writing Command and Data

Write commands and data to even addresses in the user ROM area.

25.3.4.10 Wait Mode

When entering wait mode, set the FMR01 bit in the FMR0 register to "0" (CPU rewrite mode disabled) before executing the WAIT instruction.

25.3.4.11 Stop Mode

When entering stop mode, the following settings are required:

- Set the FMR01 bit to "0" (CPU rewrite mode disabled). Disable a DMA transfer before setting the CM10 bit to "1" (stop mode).
- Execute the instruction to set the CM10 bit to "1" (stop mode) and then the JMP.B instruction.

```
e.g.,                  BSET           0, CM1         ; Stop mode
                       JMP.B          L1
               L1:
                       Program after exiting stop mode
```

25.3.4.12 Low-Power Consumption Mode and On-Chip Oscillator Low-Power Consumption Mode

If the CM05 bit is set to "1" (main clock stopped), do not execute the following commands:

- Program
- Block erase
- Erase all unlocked blocks
- Lock bit program
- Read lock bit status

25.3.5 Software Commands

Read or write 16-bit commands and data from or to even addresses in the user ROM area, in 16-bit units. When writing a command code, 8 high-order bits (D_{15} to D_8) are ignored.

Table 25.4 Software Commands

Command	First Bus Cycle			Second Bus Cycle		
	Mode	Address	Data (D_{15} to D_0)	Mode	Address	Data (D_{15} to D_0)
Read Array	Write	X	$xxFF_{16}$			
Read Status Register	Write	X	$xx70_{16}$	Read	X	SRD
Clear Status Register	Write	X	$xx50_{16}$			
Program	Write	WA	$xx40_{16}$	Write	WA	WD
Block Erase	Write	X	$xx20_{16}$	Write	BA	$xxD0_{16}$
Erase All Unlocked Block[1]	Write	X	$xxA7_{16}$	Write	X	$xxD0_{16}$
Lock Bit Program	Write	BA	$xx77_{16}$	Write	BA	$xxD0_{16}$
Read Lock Bit Status	Write	X	$xx71_{16}$	Write	BA	$xxD0_{16}$

NOTES:

1. Blocks 0 to 12 can be erased by the erase all unlocked block command.
   Block A cannot be erased. The block erase command must be used to erase the block A.

SRD: Data in the SRD register (D_7 to D_0)
WA: Address to be written (The address specified in the the first bus cycle is the same even address as the address specified in the second bus cycle.)
WD: 16-bit write data
BA: Highest-order block address (must be an even address)
X: Any even address in the user ROM space
xx: 8 high-order bits of command code (ignored)

25.3.5.1 Read Array Command

The read array command reads the flash memory.

Read array mode is entered by writing command code "$xxFF_{16}$" in the first bus cycle. Content of a specified address can be read in 16-bit units after the next bus cycle.

The microcomputer remains in read array mode until another command is written. Therefore, contents from multiple addresses can be read consecutively.

25.3.5.2 Read Status Register Command

The read status register command reads the SRD register (refer to **25.3.7 Status Register** for detail).

By writing command code "$xx70_{16}$" in the first bus cycle, the SRD register can be read in the second bus cycle. Read an even address in the user ROM area.

Do not execute this command in EW mode 1.

25.3.5.3 Clear Status Register Command

The clear status register command clears the SRD register. By writing "$xx50_{16}$" in the first bus cycle, the FMR07 and FMR06 bits in the FMR0 register are set to "00_2" and the SR5 and SR4 bits in the SRD register are set to "00_2".

25.3.5.4 Program Command

The program command writes 1-word, or 2-byte, data to the flash memory.

Auto program operation (data program and verify) will start by writing command code "xx40_{16}" in the first bus cycle and data to the write address in the second bus cycle. The address value specified in the first bus cycle must be the same even address as the write address specified in the second bus cycle.

The FMR00 bit in the FMR0 register indicates whether or not an auto program operation has been completed. The FMR00 bit is set to "0" (busy) during auto program and to "1" (ready) when the auto program operation is completed.

After the completion of auto program operation, the FMR06 bit in the FMR0 register indicates whether or not the auto program operation has been completed as expected. (Refer to **25.3.8 Full Status Check**.)

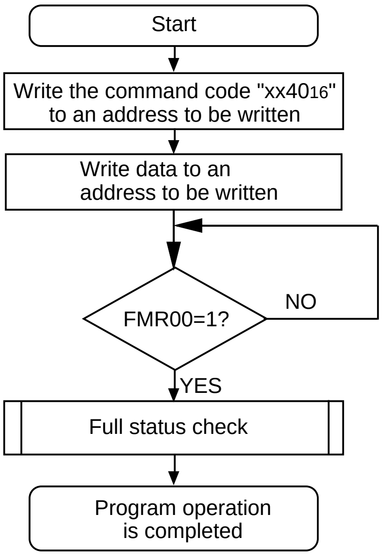
An address that is already written cannot be altered or rewritten.

Figure 25.9 shows a flow chart of the program command programming.

The lock bit can protect each block from being programmed inadvertently. (Refer to **25.3.6 Data Protect Function**.)

In EW mode 1, do not execute this command on the block where the rewrite control program is allocated.

In EW mode 0, the microcomputer enters read status register mode as soon as an auto program operation starts. The SRD register can be read. The SR7 bit in the SRD register is set to "0" at the same time an auto program operation starts. It is set to "1" when an auto program operation is completed. The microcomputer remains in read status register mode until the read array command is written. After completion of an auto program operation, the SRD register indicates whether or not the auto program operation has been completed as expected.


Start
Write the command code "xx4016" to an address to be written
Write data to an address to be written
FMR00=1?
NO
YES
Full status check
Program operation is completed

NOTES:
1. Write the command code and data to even addresses.

Figure 25.9 Program Command

25.3.5.5 Block Erase Command

The block erase command erases each block.

Auto erase operation (erase and verify) will start in the specified block by writing command code "xx20$_{16}$" in the first bus cycle and "xxD0$_{16}$" to the highest-order even address of a block in the second bus cycle.

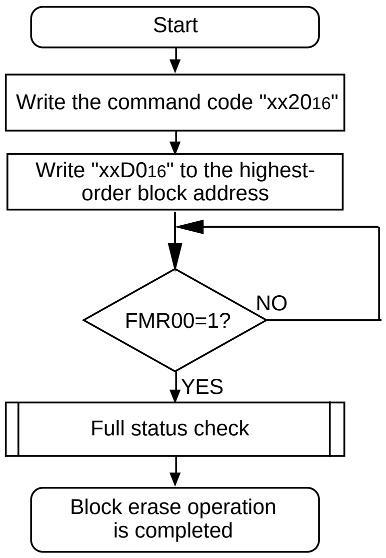
The FMR00 bit in the FMR0 register indicates whether or not an auto erase operation has been completed. The FMR00 bit is set to "0" (busy) during auto erase and to "1" (ready) when the auto erase operation is completed.

After the completion of an auto erase operation, the FMR07 bit in the FMR0 register indicates whether or not the auto erase operation has been completed as expected. (Refer to **25.3.8 Full Status Check**.)

Figure 25.10 shows a flow chart of the block erase command programming.

The lock bit can protect each block from being programmed inadvertently. (Refer to **25.3.6 Data Protect Function**.)

In EW mode 1, do not execute this command on the block where the rewrite control program is allocated.
In EW mode 0, the microcomputer enters read status register mode as soon as an auto erase operation starts. The SRD register can be read. The SR7 bit in the SRD register is set to "0" at the same time an auto erase operation starts. It is set to "1" when an auto erase operation is completed. The microcomputer remains in read status register mode until the read array command or read lock bit status command is written.


Start
Write the command code "xx20$_{16}$"
Write "xxD0$_{16}$" to the highest-order block address
FMR00=1?
NO
YES
Full status check
Block erase operation is completed

NOTES:
1. Write the command code and data to even addresses.

Figure 25.10 Block Erase Command

25.3.5.6 Erase All Unlocked Block Command

The erase all unlocked block command erases all blocks except the block A.

By writing command code "xxA7$_{16}$" in the first bus cycle and "xxD0$_{16}$" in the second bus cycle, auto erase (erase and verify) operation will run continuously in all blocks except the block A.

The FMR00 bit in the FMR0 register indicates whether or not an auto erase operation has been completed.

After the completion of an auto erase operation, the FMR07 bit in the FMR0 register indicates whether or not the auto erase operation has been completed as expected.

The lock bit can protect each block from being programmed inadvertently. (Refer to **25.3.6 Data Protect Function**.)

In EW mode 1, do not execute this command when the lock bit for any block storing the rewrite control program is set to "1" (unlocked) or when the FMR02 bit in the FMR0 register is set to "1" (lock bit disabled).

In EW mode 0, the microcomputer enters read status register mode as soon as an auto erase operation starts. The SRD register can be read. The SR7 bit in the SRD register is set to "0" (busy) at the same time an auto erase operation starts. It is set to "1" (ready) when an auto erase operation is completed. The microcomputer remains in read status register mode until the read array command or read lock bit status command is written.

Only blocks 0 to 12 can be erased by the erase all unlocked block command. The block A cannot be erased. Use the block erase command to erase the block A.

25.3.5.7 Lock Bit Program Command

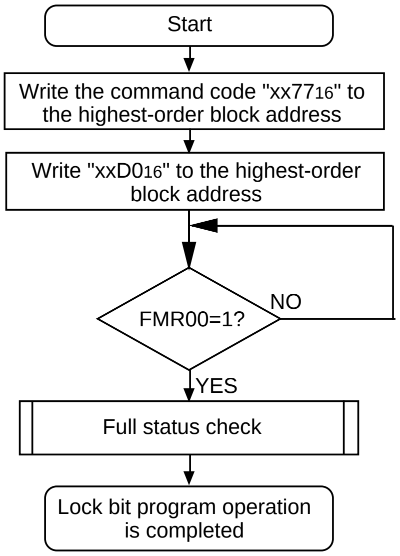
The lock bit program command sets the lock bit for a specified block to "0" (locked).

By writing command code "xx77_{16}" in the first bus cycle and "xx$D0_{16}$" to the highest-order even address of a block in the second bus cycle, the lock bit for the specified block is set to "0". The address value specified in the first bus cycle must be the same highest-order even address of a block specified in the second bus cycle.

Figure 25.11 shows a flow chart of the lock bit program command programming. Execute read lock bit status command to read lock bit state (lock bit data).

The FMR00 bit in the FMR0 register indicates whether a lock bit program operation is completed.

Refer to **25.3.6 Data Protect Function** for details on lock bit functions and how to set it to "1" (unlocked).


Start
Write the command code "xx7716" to the highest-order block address
Write "xxD016" to the highest-order block address
FMR00=1?
NO
YES
Full status check
Lock bit program operation is completed

NOTES:
1. Write the command code and data to even addresses.

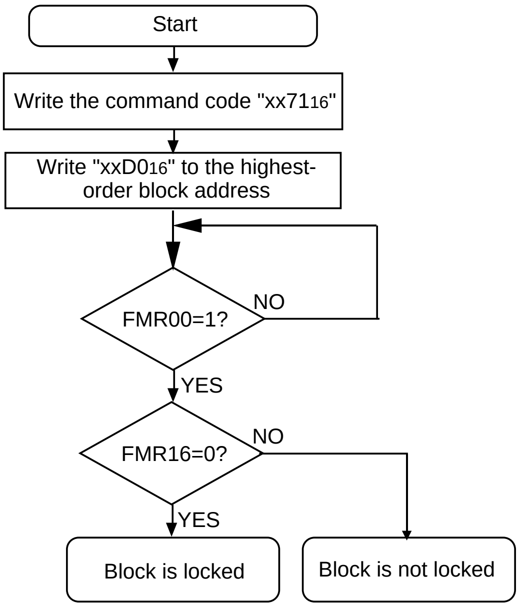
Figure 25.11 Lock Bit Program Command

25.3.5.8 Read Lock Bit Status Command

The read lock bit status command reads the lock bit state (the lock bit data) of a specified block.

By writing command code "xx71$_{16}$" in the first bus cycle and "xxD0$_{16}$" to the highest-order even address of a block in the second bus cycle, the FMR16 bit in the FMR1 register stores information on whether or not the lock bit of a specified block is locked. Read the FMR16 bit after the FMR00 bit in the FMR0 register is set to "1" (ready).

Figure 25.12 shows a flow chart of the read lock bit status command programming.


Start
Write the command code "xx71$_{16}$"
Write "xxD0$_{16}$" to the highest-order block address
FMR00=1?
NO
YES
FMR16=0?
NO
YES
Block is locked
Block is not locked

NOTES:
1. Write the command code and data to even addresses.

Figure 25.12 Read Lock Bit Status Command

25.3.6 Data Protect Function

Each block in the flash memory has a nonvolatile lock bit. The lock bit is enabled by setting the FMR02 bit to "0" (lock bit enabled). The lock bit individually protects (locks) each block against program and erase. This prevents data from being inadvertently written to or erased from the flash memory.

- When the lock bit status is set to "0", the block is locked (block is protected against program and erase).
- When the lock bit status is set to "1", the block is not locked (block can be programmed or erased).

The lock bit status is set to "0" (locked) by executing the lock bit program command and to "1" (unlocked) by erasing the block. The lock bit status cannot be set to "1" by any commands.
The lock bit status can be read by the read lock bit status command.

The lock bit function is disabled by setting the FMR02 bit to "1". All blocks are unlocked. However, individual lock bit status remains unchanged. The lock bit function is enabled by setting the FMR02 bit to "0". Lock bit status is retained.
If the block erase or erase all unlocked block command is executed while the FMR02 bit is set to "1", the target block or all blocks are erased regardless of lock bit status. The lock bit status of each block are set to "1" after an erase operation is completed.
Refer to **25.3.5 Software Commands** for details on each command.

25.3.7 Status Register (SRD Register)

The SRD register indicates the flash memory operating state and whether or not an erase or program operation is completed as expected. The FMR00, FMR06 and FMR07 bits in the FMR0 register indicate SRD register states.
Table 25.5 shows the SRD register.
In EW mode 0, the SRD register can be read when the followings occur.

- Any even address in the user ROM area is read after writing the read status register command
- Any even address in the user ROM area is read from when the program, block erase, erase all unlocked block, or lock bit program command is executed until when the read array command is executed.

25.3.7.1 Sequencer Status (SR7 and FMR00 Bits)

The sequencer status indicates the flash memory operating state. It is set to "0" while the program, block erase, erase all unlocked block, lock bit program, or read lock bit status command is being executed; otherwise, it is set to "1".

25.3.7.2 Erase Status (SR5 and FMR07 Bits)

Refer to **25.3.8 Full Status Check**.

25.3.7.3 Program Status (SR4 and FMR06 Bits)

Refer to **25.3.8 Full Status Check**.

Table 25.5 Status Register

Bits in SRD register	Bits in FMR0 Register	Status Name	Definition		Value after Reset
			"0"	"1"	
SR7 (D7)	FMR00	Sequencer status	BUSY	READY	1
SR6 (D6)	—	Reserved bit	-	-	-
SR5 (D5)	FMR07[1]	Erase status	Successfully completed	Error	0
SR4 (D4)	FMR06[1]	Program status	Successfully completed	Error	0
SR3 (D3)	—	Reserved bit	-	-	-
SR2 (D2)	—	Reserved bit	-	-	-
SR1 (D1)	—	Reserved bit	-	-	-
SR0 (D0)	—	Reserved bit	-	-	-

D0 to D7: These data buses are read when the read status register command is executed.

NOTES:

1. The FMR07 (SR5) and FMR06 (SR4) bits are set to "0" by executing the clear status register command. When the FMR07 (SR5) or FMR06 (SR4) bit is set to "1", the program, block erase, erase all unlocked block and lock bit program commands are not accepted.

25.3.8 Full Status Check

If an error occurs when a program or erase operation is completed, the FMR07 and FMR06 bits in the FMR0 register are set to "1", indicating a specific error. Therefore, execution results can be confirmed by verifying these bits (full status check).

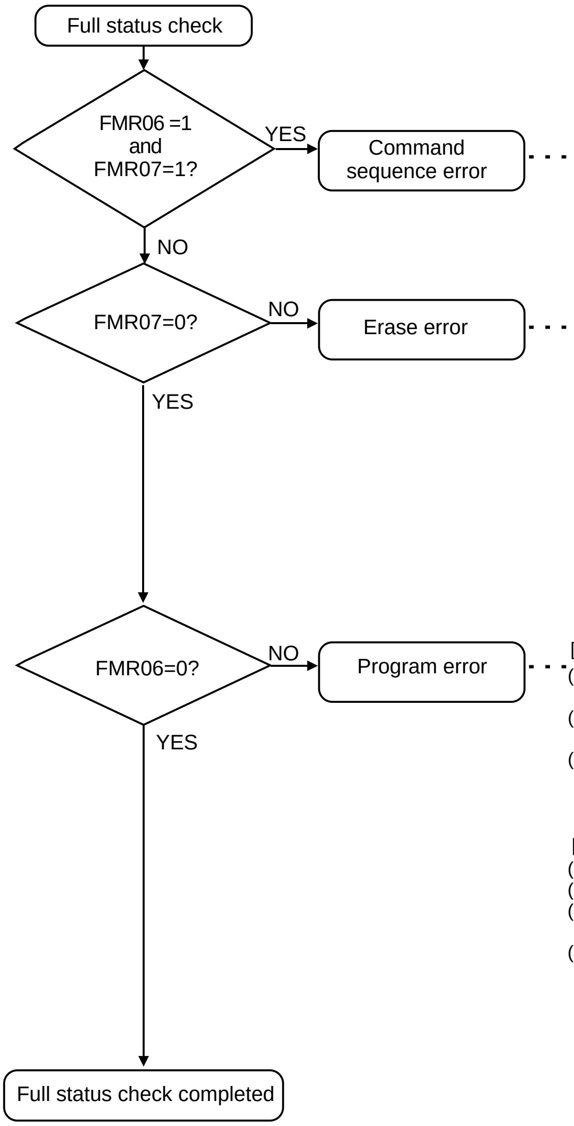
Table 25.6 lists errors and FMR0 register state. Figure 25.13 shows a flow chart of the full status check and handling procedure for each error.

Table 25.6 Errors and FMR0 Register State

FMR0 Register (SRD Register) State		Error	Error Occurrence Conditions
FMR07 (SR5)	FMR06 (SR4)		
1	1	Command sequence error	• An incorrect command is written • A value other than "xxD0$_{16}$" or "xxFF$_{16}$" is written in the second bus cycle of the lock bit program, block erase or erase all unlocked block command[1]
1	0	Erase error	• The block erase command is executed on a locked block[2] • The block erase or erase all unlocked block command is executed on an unlock block, but the erase operation is not successfully completed
0	1	Program error	• The program command is executed on locked blocks[2] • The program command is executed on an unlocked block, but the program operation is not completed as expected • The lock bit program command is executed but the program operation is not successfully completed

NOTES:

1. The flash memory enters read array mode when command code "xxFF$_{16}$" is written in the second bus cycle of these commands. The command code written in the first bus cycle is ignored.
2. When the FMR02 bit is set to "1" (lock bit disabled), no error occurs even under the conditions above.



(1) Execute the clear status register command and set the SR5 and SR4 bits to "0" (successfully completed) .
(2) Execute the correct commands again.

(1) Execute the clear status register command and set the SR5 bit to "0".
(2) Execute the lock bit read status command. Set the FMR02 bit to "1" (lock bit disabled) if the lock bit in the block where the error occurred is set to "0" (locked).
(3) Execute the block erase or erase all unlocked block command again.
NOTE: If similar error occurs, that block cannot be used.
If the lock bit is set to "1" (unlocked) in (2) above, that block cannot be used.

[When a program operation is executed]
(1) Execute the clear status register command and set the SR4 bit to "0"(successfully completed) .
(2) Execute the read lock bit status command and set the FMR02 bit to "1" if the lock bit in the block where the error occurred is set to "0".
(3) Execute the program command again.
NOTE: If a similar error occurs, that block cannot be used.
If the lock bit is set to "1" in (2) above, that block cannot be used.

[When a lock bit program operation is executed]
(1) Execute the clear status register command and set the SR4 bit to "0".
(2) Set the FMR02 bit in the FMR0 register to "1".
(3) Execute the block erase command to erase the block where the error occurred.
(4) Execute the lock bit program command again.
NOTE: If similar error occurs, that block cannot be used.

NOTE: When either FMR06 or FMR07 bit is set to "1" (terminated by error) , the program, block erase, erase all unlocked block, lock bit program and read lock bit status commands cannot be accepted.
Execute the clear status register command before each command.

Figure 25.13 Full Status Check and Handling Procedure for Each Error

25.4 Standard Serial I/O Mode

In standard serial I/O mode, the serial programmer supporting the M32C/86 group (M32C/86, M32C/86T) can be used to rewrite the flash memory user ROM area, while the microcomputer is mounted on a board. For more information about the serial programmer, contact your serial programmer manufacturer. Refer to the user's manual included with your serial programmer for instructions.

Table 25.7 lists pin descriptions (flash memory standard serial I/O mode). Figures 25.14 to 25.16 show pin connections in serial I/O mode.

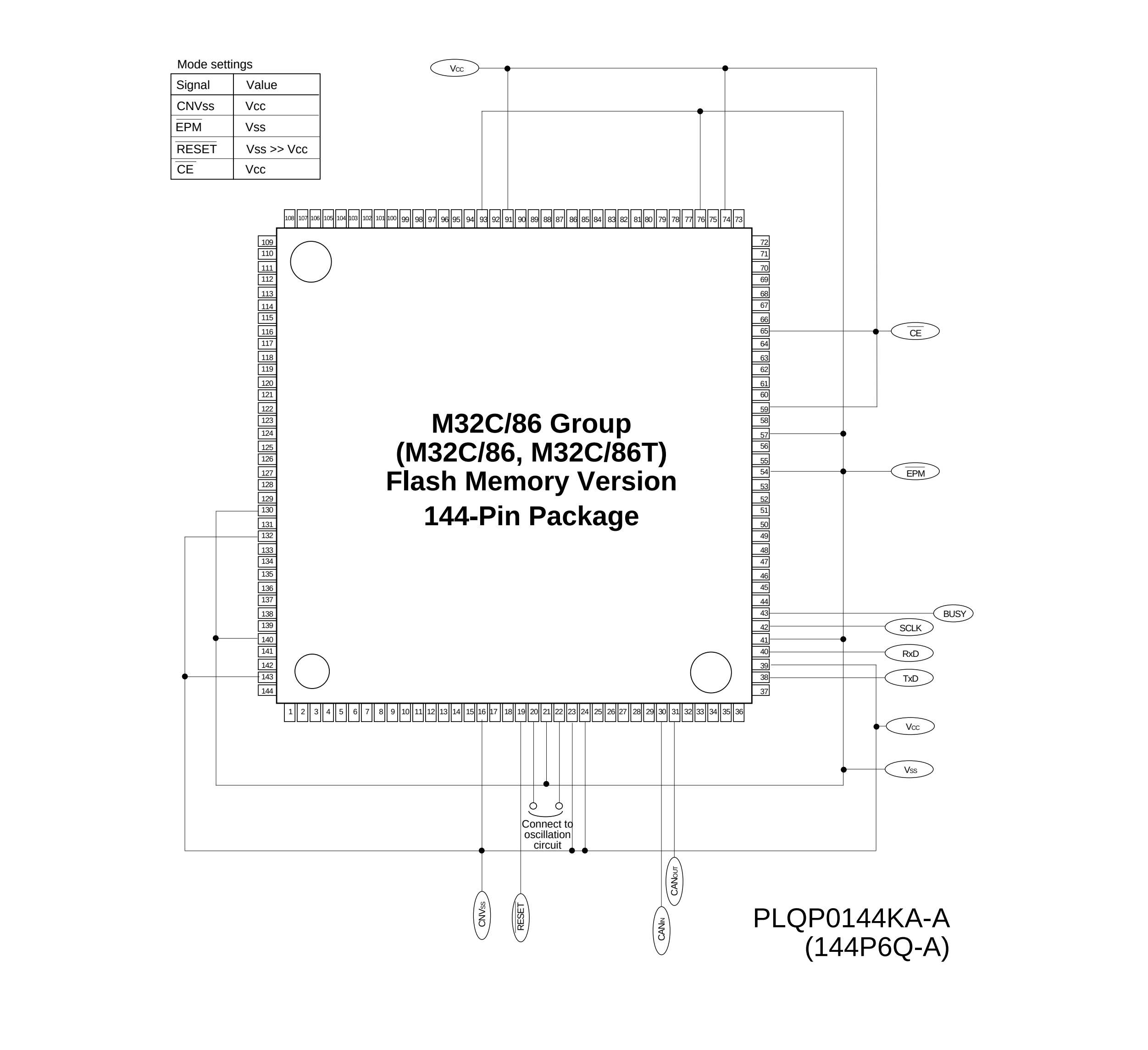
25.4.1 ID Code Verify Function

The ID code verify function determines whether or not the ID codes sent from the serial programmer matches those written in the flash memory. (Refer to **25.2 Functions to Prevent Flash Memory from Rewriting**.)

Table 25.7 Pin Description (Flash Memory Standard Serial I/O Mode)

Symbol	Function	I/O	Function
V_{CC} V_{SS}	Power supply input	I	Apply the guaranteed program/erase supply voltage to the V_{CC} pin. Apply 0 V to the V_{SS} pin
CNV_{SS}	CNV_{SS}	I	Connect this pin to V_{CC}
$\overline{RESET}$	Reset input	I	Reset input pin. Apply 20 or more clock cycles to the X_{IN} pin while an "L" signal is applied to the $\overline{RESET}$ pin
X_{IN}	Clock input	I	Connect a ceramic resonator or crystal oscillator between X_{IN} and X_{OUT}
X_{OUT}	Clock output	O	To use the external clock, input the clock from X_{IN} and leave X_{OUT} open
BYTE	BYTE input	I	Connect this pin to V_{SS} or V_{CC}
AV_{CC} AV_{SS}	Analog power supply input	I	Connect AV_{CC} to V_{CC} Connect AV_{SS} to V_{SS}
V_{REF}	Reference voltage input	I	Reference voltage input pin for the A/D converter
$P0_0$ to $P0_7$	Input port P0	I	Apply an "H" or "L" signal to this pin, or leave open
$P1_0$ to $P1_7$	Input port P1	I	Apply an "H" or "L" signal to this pin, or leave open
$P2_0$ to $P2_7$	Input port P2	I	Apply an "H" or "L" signal to this pin, or leave open
$P3_0$ to $P3_7$	Input port P3	I	Apply an "H" or "L" signal to this pin, or leave open
$P4_0$ to $P4_7$	Input port P4	I	Apply an "H" or "L" siganlto this pin, or leave open
$P5_0$	$\overline{CE}$ input	I	Apply "H" to this pin
$P5_5$	$\overline{EPM}$ input	I	Apply "L" to this pin
$P5_1$ to $P5_4$ $P5_6$, $P5_7$	Input port P5	I	Apply "H" or "L" to this pin, or leave open
$P6_0$ to $P6_3$	Input port P6	I	Apply an "H" or "L" signal to this pin, or leave open
$P6_4$	BUSY output	O	Standard serial I/O mode 1: BUSY signal output pin Standard serial I/O mode 2: Program running verify monitor Standard serial I/O mode 3: Leave open
$P6_5$	SCLK input	I	Standard serial I/O mode 1: Serial clock input pin Standard serial I/O mode 2, 3: Apply "L" to this pin
$P6_6$	RxD Data input	I	Standard serial I/O mode 1, 2: Serial data input pin Standard serial I/O mode 3: Apply "H" to this pin
$P6_7$	TxD Data output	O	Standard serial I/O mode 1, 2: Serial data output pin Standard serial I/O mode 3: Leave open
$P7_0$ to $P7_5$	Input port P7	I	Apply "H" or "L" to this pin, or leave open
$P7_6$	CAN output	O	Standard serial I/O mode 1, 2: Apply an "H" or "L" signal to this pin, or leave open Standard serial I/O mode 3: CAN output pin
$P7_7$	CAN input	I	Standard serial I/O mode 1, 2: Apply an "H" or "L" signal to this pin, or leave open Standard serial I/O mode 3: CAN input pin
$P8_0$ to $P8_4$ $P8_6$, $P8_7$	Input port P8	I	Apply an "H" or "L" signal to this pin, or leave open
$P8_5$	$\overline{NMI}$ input	I	Connect this pin to V_{CC}
$P9_0$ to $P9_7$	Input port P9	I	Apply an "H" or "L" signal to this pin, or leave open
$P10_0$ to $P10_7$	Input port P10	I	Apply an "H" or "L" signal to this pin, or leave open
$P11_0$ to $P11_4$	Input port P11	I	Apply an "H" or "L" signal to this pin, or leave open
$P12_0$ to $P12_7$	Input port P12	I	Apply an "H" or "L" signal to this pin, or leave open
$P13_0$ to $P13_7$	Input port P13	I	Apply an "H" or "L" signal to this pin, or leave open
$P14_0$ to $P14_6$	Input port P14	I	Apply an "H" or "L" signal to this pin, or leave open
$P15_0$ to $P15_7$	Input port P15	I	Apply an "H" or "L" signal to this pin, or leave open

I : Input O : Output I/O : Input and output



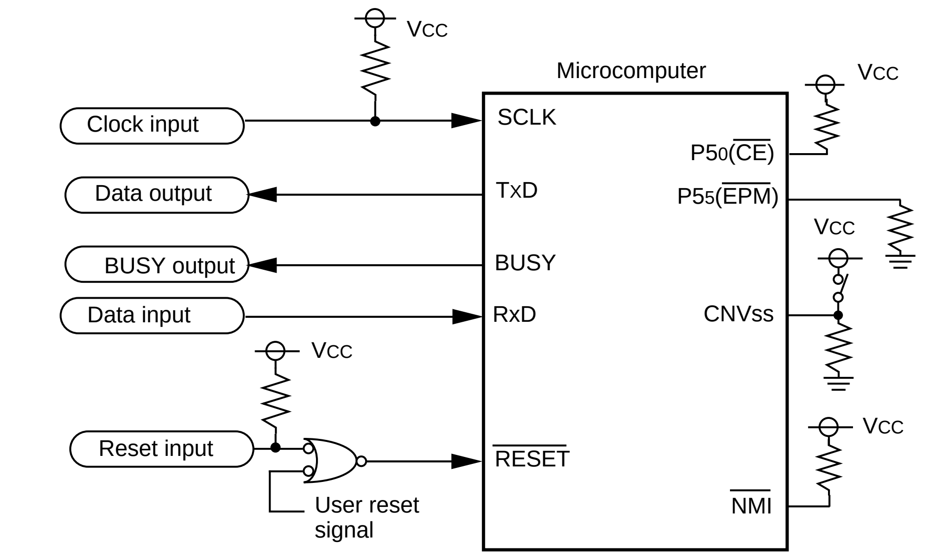
Mode settings

Signal	Value
CNVss	Vcc
EPM	Vss
RESET	Vss >> Vcc
CE	Vcc

Figure 25.14 Pin Connections in Standard Serial I/O Mode

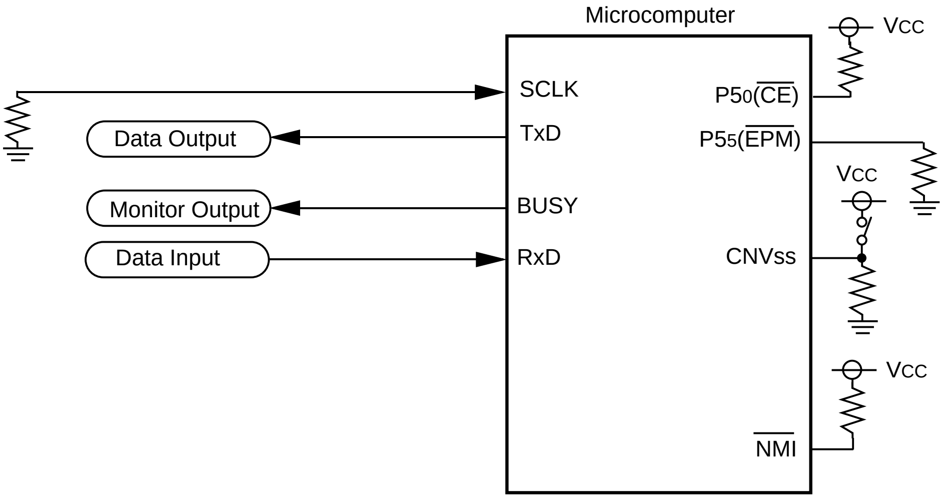
25.4.2 Circuit Application in Standard Serial I/O Mode

Figure 25.17 shows an example of a circuit application in standard serial I/O mode 1. Figure 25.18 shows an example of a circuit application serial I/O mode 2. Figure 25.19 shows an example of a circuit application serial I/O mode 3. Refer to the user's manual of your serial programmer to handle pins controlled by the serial programmer.



NOTES:
1. Control pins and external circuitry vary with the serial programmer. Refer to the user's manual included with the serial programmer.
2. In this example, a selector controls the voltage applied to the CNVss pin to switch between in single-chip mode and in standard serial I/O mode.
3. In standard serial I/O mode 1, if the user reset signal becomes "L" while the microcomputer is communicating with the serial programmer, break the connection between the user reset signal and the RESET pin by, for example, a jumper selector.

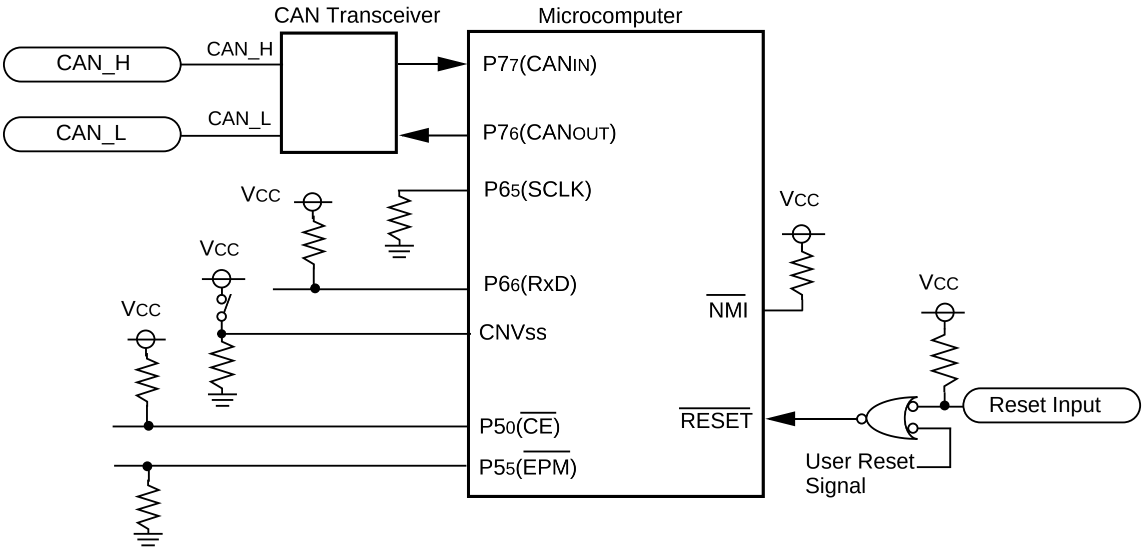
Figure 25.15 Circuit Application in Standard Serial I/O Mode 1



NOTES:

1. In this example, a selector controls the voltage applied to the CNVss pin to switch between in single-chip mode and in standard serial I/O mode.

Figure 25.16 Circuit Application in Standard Serial I/O Mode 2



NOTES:

1. Control pins and external circuitry vary with the serial programmer. Refer to the user's manual included with the serial programmer.
2. In this example, a selector controls the voltage applied to the CNVss pin to switch between in single-chip mode and in standard serial I/O mode.

Figure 25.17 Circuit Application in Standard Serial I/O Mode 3

25.5 Parallel I/O Mode

In parallel I/O mode, the user ROM area and the boot ROM area can be rewritten by a parallel programmer supporting the M32C/86 Group (M32C/86, M32C/86T). Contact your parallel programmer manufacturer for more information on the parallel programmer. Refer to the user's manual included with your parallel programmer for instructions.

25.5.1 Boot ROM Area

An erase block operation in the boot ROM area is applied to only one 4-Kbyte block. The rewrite control program in standard serial I/O mode is written in the boot ROM area before shipment. Do not rewrite the boot ROM area if using the serial programmer.

In parallel I/O mode, the boot ROM area is located in addresses $FFF000_{16}$ to $FFFFFF_{16}$. Rewrite this address range only if rewriting the boot ROM area. (Do not access addresses other than addresses $FFF000_{16}$ to $FFFFFF_{16}$.)

25.5.2 ROM Code Protect Function

The ROM code protect function prevents the flash memory from being read and rewritten in parallel I/O mode. (Refer to **25.2 Functions to Prevent Flash Memory from Rewriting**.)

26. Electrical Characteristics

26.1 Electrical Characteristics (M32C/86)

Table 26.1 Absolute Maximum Ratings

Symbol	Parameter		Condition	Value	Unit
V_{CC}	Supply Voltage		V_{CC}=AV_{CC}	-0.3 to 6.0	V
AV_{CC}	Analog Supply Voltage		V_{CC}=AV_{CC}	-0.3 to 6.0	V
V_I	Input Voltage	$P0_{0}$-$P0_{7}$, $P1_{0}$-$P1_{7}$, $P2_{0}$-$P2_{7}$, $P3_{0}$-$P3_{7}$, $P4_{0}$-$P4_{7}$, $P5_{0}$-$P5_{7}$, $P6_{0}$-$P6_{7}$, $P7_{2}$-$P7_{7}$, $P8_{0}$-$P8_{7}$, $P9_{0}$-$P9_{7}$, $P10_{0}$-$P10_{7}$, $P11_{0}$-$P11_{4}$, $P12_{0}$-$P12_{7}$, $P13_{0}$-$P13_{7}$, $P14_{0}$-$P14_{6}$, $P15_{0}$-$P15_{7}$, V_{REF}, X_{IN}, $\overline{RESET}$, CNV_{SS}, BYTE		-0.3 to V_{CC}+0.3	V
		$P7_{0}$, $P7_{1}$		-0.3 to 6.0	V
V_O	Output Voltage	$P0_{0}$-$P0_{7}$, $P1_{0}$-$P1_{7}$, $P2_{0}$-$P2_{7}$, $P3_{0}$-$P3_{7}$, $P4_{0}$-$P4_{7}$, $P5_{0}$-$P5_{7}$, $P6_{0}$-$P6_{7}$, $P7_{2}$-$P7_{7}$, $P8_{0}$-$P8_{7}$, $P9_{0}$-$P9_{7}$, $P10_{0}$-$P10_{7}$, $P11_{0}$-$P11_{4}$, $P12_{0}$-$P12_{7}$, $P13_{0}$-$P13_{7}$, $P14_{0}$-$P14_{6}$, $P15_{0}$-$P15_{7}$, X_{OUT}		-0.3 to V_{CC}+0.3	V
		$P7_{0}$, $P7_{1}$		-0.3 to 6.0	
Pd	Power Dissipation		Topr=25° C	500	mW
Topr	Operating Ambient Temperature	during CPU operation		-20 to 85/ -40 to 85[1]	° C
		during flash memory program and erase operation		0 to 60	
Tstg	Storage Temperature			-65 to 150	° C

NOTES:

1. Contact our sales office if temperature range of -40 to 85° C is required.

Table 26.2 Recommended Operating Conditions (V$_{CC}$=4.2 V to 5.5 V at Topr=– 20 to 85^{o}C unless otherwise specified)

Symbol	Parameter		Standard			Unit
			Min.	Typ.	Max.	
V$_{CC}$	Supply Voltage		4.2	5.0	5.5	V
AV$_{CC}$	Analog Supply Voltage			V$_{CC}$		V
V$_{SS}$	Supply Voltage			0		V
AV$_{SS}$	Analog Supply Voltage			0		V
V$_{IH}$	Input High ("H") Voltage	P2$_0$-P2$_7$, P3$_0$-P3$_7$, P4$_0$-P4$_7$, P5$_0$-P5$_7$, P6$_0$-P6$_7$, P7$_2$-P7$_7$, P8$_0$-P8$_7$[3], P9$_0$-P9$_7$, P10$_0$-P10$_7$, P11$_0$-P11$_4$, P12$_0$-P12$_7$, P13$_0$-P13$_7$, P14$_0$-P14$_6$, P15$_0$-P15$_7$, X$_{IN}$, $\overline{\text{RESET}}$, CNV$_{SS}$, BYTE	0.8V$_{CC}$		V$_{CC}$	V
		P7$_0$, P7$_1$	0.8V$_{CC}$		6.0	
		P0$_0$-P0$_7$, P1$_0$-P1$_7$ (in single-chip mode)	0.8V$_{CC}$		V$_{CC}$	
		P0$_0$-P0$_7$, P1$_0$-P1$_7$ (in memory expansion mode and microprocesor mode)	0.5V$_{CC}$		V$_{CC}$	
V$_{IL}$	Input Low ("L") Voltage	P2$_0$-P2$_7$, P3$_0$-P3$_7$, P4$_0$-P4$_7$, P5$_0$-P5$_7$, P6$_0$-P6$_7$, P7$_2$-P7$_7$, P8$_0$-P8$_7$[3], P9$_0$-P9$_7$, P10$_0$-P10$_7$, P11$_0$-P11$_4$, P12$_0$-P12$_7$, P13$_0$-P13$_7$, P14$_0$-P14$_6$, P15$_0$-P15$_7$, X$_{IN}$, $\overline{\text{RESET}}$, CNV$_{SS}$, BYTE	0		0.2V$_{CC}$	V
		P0$_0$-P0$_7$, P1$_0$-P1$_7$ (in single-chip mode)	0		0.2V$_{CC}$	
		P0$_0$-P0$_7$, P1$_0$-P1$_7$ (in memory expansion mode and microprocesor mode)	0		0.16V$_{CC}$	
I$_{OH(peak)}$	Peak Output High ("H") Current[2]	P0$_0$-P0$_7$, P1$_0$-P1$_7$, P2$_0$-P2$_7$, P3$_0$-P3$_7$, P4$_0$-P4$_7$, P5$_0$-P5$_7$, P6$_0$-P6$_7$, P7$_2$-P7$_7$, P8$_0$-P8$_4$, P8$_6$, P8$_7$, P9$_0$-P9$_7$, P10$_0$-P10$_7$, P11$_0$-P11$_4$, P12$_0$-P12$_7$, P13$_0$-P13$_7$, P14$_0$-P14$_6$, P15$_0$-P15$_7$			-10.0	mA
		P12$_0$-P12$_7$, P13$_0$-P13$_7$[4]			-20.0	
I$_{OH(avg)}$	Average Output High ("H") Current[1]	P0$_0$-P0$_7$, P1$_0$-P1$_7$, P2$_0$-P2$_7$, P3$_0$-P3$_7$, P4$_0$-P4$_7$, P5$_0$-P5$_7$, P6$_0$-P6$_7$, P7$_2$-P7$_7$, P8$_0$-P8$_4$, P8$_6$, P8$_7$, P9$_0$-P9$_7$, P10$_0$-P10$_7$, P11$_0$-P11$_4$, P12$_0$-P12$_7$, P13$_0$-P13$_7$, P14$_0$-P14$_6$, P15$_0$-P15$_7$			-5.0	mA
		P12$_0$-P12$_7$, P13$_0$-P13$_7$[4]			-10.0	
I$_{OL(peak)}$	Peak Output Low ("L") Current[2]	P0$_0$-P0$_7$, P1$_0$-P1$_7$, P2$_0$-P2$_7$, P3$_0$-P3$_7$, P4$_0$-P4$_7$, P5$_0$-P5$_7$, P6$_0$-P6$_7$, P7$_0$-P7$_7$, P8$_0$-P8$_4$, P8$_6$, P8$_7$, P9$_0$-P9$_7$, P10$_0$-P10$_7$, P11$_0$-P11$_4$, P12$_0$-P12$_7$, P13$_0$-P13$_7$, P14$_0$-P14$_6$, P15$_0$-P15$_7$			10.0	mA
		P12$_0$-P12$_7$, P13$_0$-P13$_7$[4]			20.0	
I$_{OL(avg)}$	Average Output Low ("L") Current[1]	P0$_0$-P0$_7$, P1$_0$-P1$_7$, P2$_0$-P2$_7$, P3$_0$-P3$_7$, P4$_0$-P4$_7$, P5$_0$-P5$_7$, P6$_0$-P6$_7$, P7$_0$-P7$_7$, P8$_0$-P8$_4$, P8$_6$, P8$_7$, P9$_0$-P9$_7$, P10$_0$-P10$_7$, P11$_0$-P11$_4$, P12$_0$-P12$_7$, P13$_0$-P13$_7$, P14$_0$-P14$_6$, P15$_0$-P15$_7$			5.0	mA
		P12$_0$-P12$_7$, P13$_0$-P13$_7$[4]			10.0	

NOTES:

1. Typical values when average output current is 100ms.
2. Total I$_{OL(peak)}$ for P0, P1, P2, P8$_6$, P8$_7$, P9, P10, P11, P14 and P15 must be 80 mA or less.
   Total I$_{OL(peak)}$ for P3, P4, P5, P6, P7, P8$_0$ to P8$_4$, P12 and P13 must be 80 mA or less.
   Total I$_{OH(peak)}$ for P0, P1, P2, and P11 must be -40 mA or less.
   Total I$_{OH(peak)}$ for P8$_6$, P8$_7$, P9, P10, P14 and P15 must be -40 mA or less.
   Total I$_{OH(peak)}$ for P3, P4, P5, P12 and P13 must be -40 mA or less.
   Total I$_{OH(peak)}$ for P6, P7, and P8$_0$ to P8$_4$ must be -40 mA or less.
3. The V$_{IH}$ and V$_{IL}$ reference for P8$_7$ applies when P8$_7$ is used as a programmable input port.
   It does not apply when P8$_7$ is used as X$_{CIN}$.
4. This reference applies when the stepping motor control function is used.
   I$_{OL(avg)}$ must be 80 mA or less. I$_{OL(peak)}$ must be 120 mA or less.
   I$_{OH(avg)}$ must be -80 mA or less. I$_{OH(peak)}$ must be -120 mA or less.

Table 26.2 Recommended Operating Conditions (Continued) (VCC=4.2 V to 5.5 V at Topr=–20 to 85°C unless otherwise specified)

Symbol	Parameter		Standard			Unit
			Min.	Typ.	Max.	
$f_{(BCLK)}$	CPU Clock Frequency	V_{CC}=4.2 to 5.5V	0		32	MHz
$f_{(X_{IN})}$	Main Clock Input Frequency	V_{CC}=4.2 to 5.5V	0		32	MHz
$f_{(X_{CIN})}$	Sub Clock Frequency			32.768	50	kHz
$f_{(Ring)}$	On-chip Oscillator Frequency (V_{CC}=5.0V, Topr=25° C)		0.5	1	2	MHz
$f_{(PLL)}$	PLL Clock Frequency	V_{CC}=4.2 to 5.5V	10		32	MHz
$t_{SU(PLL)}$	Wait Time to Stabilize PLL Frequency Synthesizer	V_{CC}=5.0V			5	ms

V_{CC}=5V

Table 26.3 Electrical Characteristics

(V_{CC}=4.2 V to 5.5 V, V_{SS}=0 V at Topr= –20 to 85°C, f(BCLK)=32MHz unless otherwise specified)

Symbol	Parameter			Condition	Standard			Unit
					Min.	Typ.	Max.	
V_{OH}	Output High ("H") Voltage	$P0_0$-$P0_7$, $P1_0$-$P1_7$, $P2_0$-$P2_7$, $P3_0$-$P3_7$, $P4_0$-$P4_7$, $P5_0$-$P5_7$, $P6_0$-$P6_7$, $P7_2$-$P7_7$, $P8_0$-$P8_4$, $P8_6$, $P8_7$, $P9_0$-$P9_7$, $P10_0$-$P10_7$, $P11_0$-$P11_4$, $P12_0$-$P12_7$, $P13_0$-$P13_7$, $P14_0$-$P14_6$, $P15_0$-$P15_7$		I_{OH}=-5 mA	V_{CC}-2.0		V_{CC}	V
		$P0_0$-$P0_7$, $P1_0$-$P1_7$, $P2_0$-$P2_7$, $P3_0$-$P3_7$, $P4_0$-$P4_7$, $P5_0$-$P5_7$, $P6_0$-$P6_7$, $P7_2$-$P7_7$, $P8_0$-$P8_4$, $P8_6$, $P8_7$, $P9_0$-$P9_7$, $P10_0$-$P10_7$, $P11_0$-$P11_4$, $P12_0$-$P12_7$, $P13_0$-$P13_7$, $P14_0$-$P14_6$, $P15_0$-$P15_7$		I_{OH}=-200 μA	V_{CC}-0.3		V_{CC}	V
		$P12_0$-$P12_7$, $P13_0$-$P13_7$[1]		I_{OH}=-10 mA	V_{CC}-2.0			V
		X_{OUT}		I_{OH}=-1 mA	3.0		V_{CC}	V
		X_{COUT}	High Power	No load applied		2.5		V
			Low Power	No load applied		1.6		
V_{OL}	Output Low ("L") Voltage	$P0_0$-$P0_7$, $P1_0$-$P1_7$, $P2_0$-$P2_7$, $P3_0$-$P3_7$, $P4_0$-$P4_7$, $P5_0$-$P5_7$, $P6_0$-$P6_7$, $P7_0$-$P7_7$, $P8_0$-$P8_4$, $P8_6$, $P8_7$, $P9_0$-$P9_7$, $P10_0$-$P10_7$, $P11_0$-$P11_4$, $P12_0$-$P12_7$, $P13_0$-$P13_7$, $P14_0$-$P14_6$, $P15_0$-$P15_7$		I_{OL}=5 mA			2.0	V
		$P0_0$-$P0_7$, $P1_0$-$P1_7$, $P2_0$-$P2_7$, $P3_0$-$P3_7$, $P4_0$-$P4_7$, $P5_0$-$P5_7$, $P6_0$-$P6_7$, $P7_0$-$P7_7$, $P8_0$-$P8_4$, $P8_6$, $P8_7$, $P9_0$-$P9_7$, $P10_0$-$P10_7$, $P11_0$-$P11_4$, $P12_0$-$P12_7$, $P13_0$-$P13_7$, $P14_0$-$P14_6$, $P15_0$-$P15_7$		I_{OL}=200 μA			0.45	V
		$P12_0$-$P12_7$, $P13_0$-$P13_7$[1]		I_{OH}=-10 mA			2.0	V
		X_{OUT}		I_{OL}=1 mA			2.0	V
		X_{COUT}	High Power	No load applied		0		V
			Low Power	No load applied		0		
V_{T+}-V_{T-}	Hysteresis	$\overline{HOLD}$, $\overline{RDY}$, $TA0_{IN}$-$TA4_{IN}$, $TB0_{IN}$-$TB5_{IN}$, $\overline{INT0}$-$\overline{INT5}$, $\overline{AD_{TRG}}$, $\overline{CTS0}$-$\overline{CTS4}$, CLK0-CLK4, $TA0_{OUT}$-$TA4_{OUT}$, $\overline{NMI}$, $\overline{KI0}$-$\overline{KI3}$, RxD0-RxD4, SCL0-SCL4, SDA0-SDA4			0.2		1.0	V
		$\overline{RESET}$			0.2		1.8	V
I_{IH}	Input High ("H") Current	$P0_0$-$P0_7$, $P1_0$-$P1_7$, $P2_0$-$P2_7$, $P3_0$-$P3_7$, $P4_0$-$P4_7$, $P5_0$-$P5_7$, $P6_0$-$P6_7$, $P7_0$-$P7_7$, $P8_0$-$P8_7$, $P9_0$-$P9_7$, $P10_0$-$P10_7$, $P11_0$-$P11_4$, $P12_0$-$P12_7$, $P13_0$-$P13_7$, $P14_0$-$P14_6$, $P15_0$-$P15_7$, X_{IN}, $\overline{RESET}$, CNV_{SS}, BYTE		V_I=5V			5.0	μA
I_{IL}	Input Low ("L") Current	$P0_0$-$P0_7$, $P1_0$-$P1_7$, $P2_0$-$P2_7$, $P3_0$-$P3_7$, $P4_0$-$P4_7$, $P5_0$-$P5_7$, $P6_0$-$P6_7$, $P7_0$-$P7_7$, $P8_0$-$P8_7$, $P9_0$-$P9_7$, $P10_0$-$P10_7$, $P11_0$-$P11_4$, $P12_0$-$P12_7$, $P13_0$-$P13_7$, $P14_0$-$P14_6$, $P15_0$-$P15_7$, X_{IN}, $\overline{RESET}$, CNV_{SS}, BYTE		V_I=0V			-5.0	μA
R_{PULLUP}	Pull-up Resistance	$P0_0$-$P0_7$, $P1_0$-$P1_7$, $P2_0$-$P2_7$, $P3_0$-$P3_7$, $P4_0$-$P4_7$, $P5_0$-$P5_7$, $P6_0$-$P6_7$, $P7_2$-$P7_7$, $P8_0$-$P8_4$, $P8_6$, $P8_7$, $P9_0$-$P9_7$, $P10_0$-$P10_7$, $P11_0$-$P11_4$, $P12_0$-$P12_7$, $P13_0$-$P13_7$, $P14_0$-$P14_6$, $P15_0$-$P15_7$		V_I=0V	30	50	167	kΩ

NOTES:

1. This reference applies when the stepping motor control function is used.

VCC=5V

Table 26.3 Electrical Characteristics (Continued)

(VCC=4.2 V to 5.5 V, VSS=0 V at Topr= –20 to 85°C, f(BCLK)=32MHz unless otherwise specified)

Symbol	Parameter	Measurement Condition		Standard			Unit
				Min.	Typ.	Max.	
Rf_{XIN}	Feedback Resistance	X_{IN}			1.5		MΩ
Rf_{XCIN}	Feedback Resistance	X_{CIN}			10		MΩ
V_{RAM}	RAM Standby Voltage	In stop mode		2.0			V
I_{CC}	Power Supply Current	In single-chip mode, output pins are left open and other pins are connected to V_{SS}.	f(BCLK)=32 MHz, Square wave, No division		28	45	mA
			f(BCLK)=32 kHz, In low-power consumption mode, Program running on ROM		430		µA
			f(BCLK)=32 kHz, In low-power consumption mode, Program running on RAM(1)		25		µA
			f(BCLK)=32 kHz, In wait mode, Topr=25° C		10		µA
			While clock stops, Topr=25° C		0.8	5	µA
			While clock stops, Topr=85° C			50	µA

NOTES:

1. Value is obtained when setting the FMSTP bit in the FMR0 register to "1" (flash memory stopped).

V_{CC}=5V

Table 26.4 A/D Conversion Characteristics (V_{CC}=AV_{CC}=V_{REF}=4.2 to 5.5 V, Vss= AV_{SS} = 0 V at Topr=–20 to 85^{o}C, f(BCLK) = 32MHz unless otherwise specified)

Symbol	Parameter	Measurement Condition		Standard			Unit
				Min.	Typ.	Max.	
-	Resolution	V_{REF}=V_{CC}				10	Bits
INL	Integral Nonlinearity Error	V_{REF}=V_{CC}=5V	AN_0 to AN_7, $AN0_0$ to $AN0_7$, $AN2_0$ to $AN2_7$, $AN15_0$ to $AN15_7$, ANEX0, ANEX1			±3	LSB LSB
			External op-amp connection mode			±7	LSB LSB
DNL	Differential Nonlinearity Error					±1	LSB
-	Offset Error					±3	LSB
-	Gain Error					±3	LSB
R_{LADDER}	Resistor Ladder	V_{REF}=V_{CC}		8		40	kΩ
t_{CONV}	10-bit Conversion Time(1, 2)			2.06			µs
t_{CONV}	8-bit Conversion Time(1, 2)			1.75			µs
t_{SAMP}	Sampling Time(1)			0.188			µs
V_{REF}	Reference Voltage			2		V_{CC}	V
V_{IA}	Analog Input Voltage			0		V_{REF}	V

NOTES:

1. Divide f(X_{IN}), if exceeding 16 MHz, to keep ϕAD frequency at 16 MHz or less.
2. With using the sample and hold function.

Table 26.5 D/A Conversion Characteristics (V_{CC}=V_{REF}=4.2 to 5.5 V, V_{SS}=AV_{SS}=0 V at Topr=–20 to 85^{o}C, f(BCLK) = 32MHz unless otherwise specified)

Symbol	Parameter	Measurement Condition	Standard			Unit
			Min.	Typ.	Max.	
-	Resolution				8	Bits
-	Absolute Accuracy				1.0	%
t_{SU}	Setup Time				3	µs
R_O	Output Resistance		4	10	20	kΩ
I_{VREF}	Reference Power Supply Input Current	(Note 1)			1.5	mA

NOTES:

1. Measurement when using one D/A converter. The DAi register (i=0, 1) of the D/A converter, not being used, is set to "00_{16}". The resistor ladder in the A/D converter is excluded.
   I_{VREF} flows even if the VCUT bit in the AD0CON1 register is set to "0" (no V_{REF} connection).

VCC=5V

Table 26.6 Flash Memory Version Electrical Characteristics (VCC=4.5 to 5.5 V at Topr=0 to 60°C unless otherwise specified)

Symbol	Parameter		Standard			Unit
			Min.	Typ.	Max.	
-	Program and Erase Endurance[2]		100			cycles
-	Word Program Time (VCC=5.0V, Topr=25° C)			25	200	µs
-	Lock Bit Program Time			25	200	µs
-	Block Erase Time (VCC=5.0V, Topr=25° C)	4-Kbyte Block		0.3	4	s
		8-Kbyte Block		0.3	4	s
		32-Kbyte Block		0.5	4	s
		64-Kbyte Block		0.8	4	s
-	All-Unlocked-Block Erase Time[1]				4 x *n*	s
t_{PS}	Wait Time to Stabilize Flash Memory Circuit				15	µs
-	Data Hold Time (Topr=-40 to 85 ° C)		10			years

NOTES:

1. *n* denotes the number of block to be erased.
2. Number of program-erase cycles per block.
   If Program and Erase Endurance is *n* cycle (*n*=100), each block can be erased and programmed *n* cycles.
   For example, if a 4-Kbyte block A is erased after programming a word data 2,048 times, each to a different address, this counts as one program and erase endurance. Data can not be programmed to the same address more than once without erasing the block. (rewrite prohibited).

Table 26.7 Voltage Detection Circuit Electrical Characteristics (VCC=4.2 V to 5.5 V, Vss=0 V at Topr=25°C unless otherwise specified)

Symbol	Parameter	Measurement Condition	Standard			Unit
			Min.	Typ.	Max.	
Vdet4	Low Voltage Detection Voltage[1]	VCC=4.2 to 5.5V		3.8		V
Vdet3	Reset Space Detection Voltage[1]			3.0		V
Vdet3s	Low Voltage Reset Hold Voltage		2.0			V
Vdet3r	Low Voltage Reset Release Voltage[2]			3.1		V

NOTES:

1. Vdet4 >Vdet3
2. Vdet3r >Vdet3 is not guaranteed.

Table 26.8 Power Supply Timing

Symbol	Parameter	Measurement Condition	Standard			Unit
			Min.	Typ.	Max.	
td(P-R)	Wait Time to Stabilize Internal Supply Voltage when Power-on	VCC=4.2 to 5.5V			2	ms
td(S-R)	Wait Time to Release Brown-out Detection Reset	VCC=Vdet3r to 5.5V		6[1]	20	ms
td(E-A)	Start-up Time for Low Voltage Detection Circuit Operation	VCC=4.2 to 5.5V			20	µs

NOTES:

1. VCC=5V

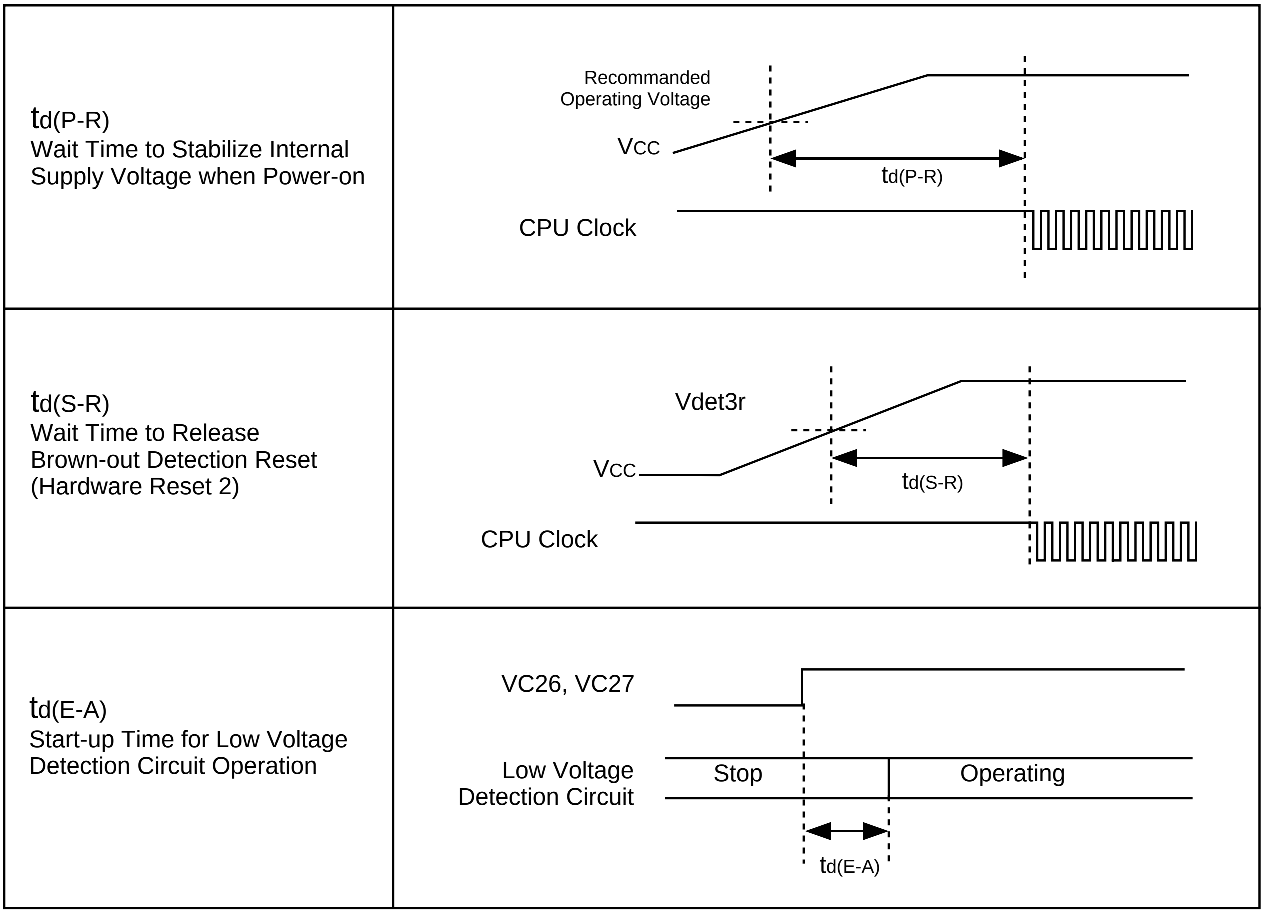

td(P-R)
Wait Time to Stabilize Internal Supply Voltage when Power-on
Recommanded Operating Voltage
VCC
td(P-R)
CPU Clock
td(S-R)
Wait Time to Release Brown-out Detection Reset (Hardware Reset 2)
Vdet3r
VCC
td(S-R)
CPU Clock
td(E-A)
Start-up Time for Low Voltage Detection Circuit Operation
VC26, VC27
Low Voltage Detection Circuit
Stop
Operating
td(E-A)

Figure 26.1 Power Supply Timing Diagram

VCC=5V

Timing Requirements

(VCC=4.2 V to 5.5 V, VSS=0 V at Topr=–20 to 85°C unless otherwise specified)

Table 26.9 External Clock Input

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t_c	External Clock Input Cycle Time	31.25		ns
$t_{w(H)}$	External Clock Input High ("H") Width	13.75		ns
$t_{w(L)}$	External Clock Input Low ("L") Width	13.75		ns
t_r	External Clock Rise Time		5	ns
t_f	External Clock Fall Time		5	ns

Table 26.10 Memory Expansion Mode and Microprocessor Mode

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{ac1(RD-DB)}$	Data Input Access Time (RD standard)		(Note 1)	ns
$t_{ac1(AD-DB)}$	Data Input Access Time (AD standard, CS standard)		(Note 1)	ns
$t_{ac2(RD-DB)}$	Data Input Access Time (RD standard, when accessing a space with the multiplexrd bus)		(Note 1)	ns
$t_{ac2(AD-DB)}$	Data Input Access Time (AD standard, when accessing a space with the multiplexed bus)		(Note 1)	ns
$t_{su(DB-BCLK)}$	Data Input Setup Time	26		ns
$t_{su(RDY-BCLK)}$	$\overline{\text{RDY}}$ Input Setup Time	26		ns
$t_{su(HOLD-BCLK)}$	$\overline{\text{HOLD}}$ Input Setup Time	30		ns
$t_{h(RD-DB)}$	Data Input Hold Time	0		ns
$t_{h(BCLK-RDY)}$	$\overline{\text{RDY}}$ Input Hold Time	0		ns
$t_{h(BCLK-HOLD)}$	$\overline{\text{HOLD}}$ Input Hold Time	0		ns
$t_{d(BCLK-HLDA)}$	$\overline{\text{HLDA}}$ Output Delay Time		25	ns

NOTES:

1. Values can be obtained from the following equations, according to BCLK frequecncy and external bus cycles. Insert a wait state or lower the operation frequency, f(BCLK), if the calculated value is negative.

$$t_{ac1(RD-DB)} = \frac{10^9 \text{ X } m}{f(BCLK) \text{ X } 2} - 35 \quad \text{[ns] (if external bus cycle is } a\phi + b\phi\text{, } m=(bx2)+1\text{)}$$

$$t_{ac1(AD-DB)} = \frac{10^9 \text{ X } n}{f(BCLK)} - 35 \quad \text{[ns] (if external bus cycle is } a\phi + b\phi\text{, } n=a+b\text{)}$$

$$t_{ac2(RD-DB)} = \frac{10^9 \text{ X } m}{f(BCLK) \text{ X } 2} - 35 \quad \text{[ns] (if external bus cycle is } a\phi + b\phi\text{, } m=(bx2)-1\text{)}$$

$$t_{ac2(AD-DB)} = \frac{10^9 \text{ X } p}{f(BCLK) \text{ X } 2} - 35 \quad \text{[ns] (if external bus cycle is } a\phi + b\phi\text{, } p=\{(a+b-1)x2\}+1\text{)}$$

VCC=5V

Timing Requirements

(VCC=4.2 V to 5.5 V, VSS=0 V at Topr=–20 to 85°C unless otherwise specified)

Table 26.11 Timer A Input (Count Source Input in Event Counter Mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TA)}$	TAiIN Input Cycle Time	100		ns
$t_{w(TAH)}$	TAiIN Input High ("H") Width	40		ns
$t_{w(TAL)}$	TAiIN Input Low ("L") Width	40		ns

Table 26.12 Timer A Input (Gate Input in Timer Mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TA)}$	TAiIN Input Cycle Time	400		ns
$t_{w(TAH)}$	TAiIN Input High ("H") Width	200		ns
$t_{w(TAL)}$	TAiIN Input Low ("L") Width	200		ns

Table 26.13 Timer A Input (External Trigger Input in One-Shot Timer Mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TA)}$	TAiIN Input Cycle Time	200		ns
$t_{w(TAH)}$	TAiIN Input High ("H") Width	100		ns
$t_{w(TAL)}$	TAiIN Input Low ("L") Width	100		ns

Table 26.14 Timer A Input (External Trigger Input in Pulse Width Modulation Mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{w(TAH)}$	TAiIN Input High ("H") Width	100		ns
$t_{w(TAL)}$	TAiIN Input Low ("L") Width	100		ns

Table 26.15 Timer A Input (Counter Increment/Decrement Input in Event Counter Mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(UP)}$	TAiOUT Input Cycle Time	2000		ns
$t_{w(UPH)}$	TAiOUT Input High ("H") Width	1000		ns
$t_{w(UPL)}$	TAiOUT Input Low ("L") Width	1000		ns
$t_{su(UP-TIN)}$	TAiOUT Input Setup Time	400		ns
$t_{h(TIN-UP)}$	TAiOUT Input Hold Time	400		ns

V_{CC}=5V

Timing Requirements

(V_{CC}=4.2 V to 5.5 V, V_{SS} = 0 V at Topr = –20 to 85°C unless otherwise specified)

Table 26.16 Timer B Input (Count Source Input in Event Counter Mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TB)}$	TBi_{IN} Input Cycle Time (counted on one edge)	100		ns
$t_{w(TBH)}$	TBi_{IN} Input High ("H") Width (counted on one edge)	40		ns
$t_{w(TBL)}$	TBi_{IN} Input Low ("L") Width (counted on one edge)	40		ns
$t_{c(TB)}$	TBi_{IN} Input Cycle Time (counted on both edges)	200		ns
$t_{w(TBH)}$	TBi_{IN} Input High ("H") Width (counted on both edges)	80		ns
$t_{w(TBL)}$	TBi_{IN} Input Low ("L") Width (counted on both edges)	80		ns

Table 26.17 Timer B Input (Pulse Period Measurement Mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TB)}$	TBi_{IN} Input Cycle Time	400		ns
$t_{w(TBH)}$	TBi_{IN} Input High ("H") Width	200		ns
$t_{w(TBL)}$	TBi_{IN} Input Low ("L") Width	200		ns

Table 26.18 Timer B Input (Pulse Width Measurement Mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TB)}$	TBi_{IN} Input Cycle Time	400		ns
$t_{w(TBH)}$	TBi_{IN} Input High ("H") Width	200		ns
$t_{w(TBL)}$	TBi_{IN} Input Low ("L") Width	200		ns

Table 26.19 A/D Trigger Input

Symbol	Parameter	Standard		Unit
		Min.	Max	
$t_{c(AD)}$	$\overline{AD_{TRG}}$ Input Cycle Time (required for trigger)	1000		ns
$t_{w(ADL)}$	$\overline{AD_{TRG}}$ Input Low ("L") Width	125		ns

Table 26.20 Serial I/O

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(CK)}$	CLKi Input Cycle Time	200		ns
$t_{w(CKH)}$	CLKi Input High ("H") Width	100		ns
$t_{w(CKL)}$	CLKi Input Low ("L") Width	100		ns
$t_{d(C-Q)}$	TxDi Output Delay Time		80	ns
$t_{h(C-Q)}$	TxDi Hold Time	0		ns
$t_{su(D-C)}$	RxDi Input Setup Time	30		ns
$t_{h(C-Q)}$	RxDi Input Hold Time	90		ns

Table 26.21 External Interrupt $\overline{\text{INTi}}$ Input

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{w(INH)}$	$\overline{\text{INTi}}$ Input High ("H") Width	250		ns
$t_{w(INL)}$	$\overline{\text{INTi}}$ Input Low ("L") Width	250		ns

VCC=5V

Switching Characteristics

(VCC=4.2 V to 5.5 V, VSS = 0 V at Topr = –20 to 85°C unless otherwise specified)

Table 26.22 Memory Expansion Mode and Microprocessor Mode (when accessing external memory space)

Symbol	Parameter	Measurement Condition	Standard		Unit
			Min.	Max.	
$t_{d(BCLK-AD)}$	Address Output Delay Time	See Figure 26.2		18	ns
$t_{h(BCLK-AD)}$	Address Output Hold Time (BCLK standard)		-3		ns
$t_{h(RD-AD)}$	Address Output Hold Time (RD standard)(3)		0		ns
$t_{h(WR-AD)}$	Address Output Hold Time (WR standard)(3)		(Note 1)		ns
$t_{d(BCLK-CS)}$	Chip-Select Signal Output Delay Time			18	ns
$t_{h(BCLK-CS)}$	Chip-Select Signal Output Hold Time (BCLK standard)		-3		ns
$t_{h(RD-CS)}$	Chip-Select Signal Output Hold Time (RD standard)(3)		0		ns
$t_{h(WR-CS)}$	Chip-Select Signal Output Hold Time (WR standard)(3)		(Note 1)		ns
$t_{d(BCLK-RD)}$	RD Signal Output Delay Time			18	ns
$t_{h(BCLK-RD)}$	RD Signal Output Hold Time		-5		ns
$t_{d(BCLK-WR)}$	WR Signal Output Delay Time			18	ns
$t_{h(BCLK-WR)}$	WR Signal Output Hold Time		-5		ns
$t_{d(DB-WR)}$	Data Output Delay Time (WR standard)		(Note 2)		ns
$t_{h(WR-DB)}$	Data Output Hold Time (WR standard)(3)		(Note 1)		ns
$t_{w(WR)}$	WR Output Width		(Note 2)		ns

NOTES:

1. Values can be obtained from the following equations, according to BCLK frequency.

$$t_{h(WR-DB)} = \frac{10^9}{f_{(BCLK)} \times 2} - 10 \quad [\text{ns}]$$

$$t_{h(WR-AD)} = \frac{10^9}{f_{(BCLK)} \times 2} - 10 \quad [\text{ns}]$$

$$t_{h(WR-CS)} = \frac{10^9}{f_{(BCLK)} \times 2} - 10 \quad [\text{ns}]$$

2. Values can be obtained from the following equations, according to BCLK frequency and external bus cycles.

$$t_{w(WR)} = \frac{10^9 \times n}{f_{(BCLK)} \times 2} - 15 \quad [\text{ns}]$$ (if external bus cycle is aϕ + bϕ, n=(bx2)-1)

$$t_{d(DB-WR)} = \frac{10^9 \times m}{f_{(BCLK)}} - 20 \quad [\text{ns}]$$ (if external bus cycle is aϕ + bϕ, m= b)

3. *tc* ns is added when recovery cycle is inserted.

Switching Characteristics

(VCC=4.2 V to 5.5 V, VSS = 0 V at Topr = –20 to 85°C unless otherwise specified)

Table 26.23 Memory Expansion Mode and Microprocessor Mode (when accessing an external memory space with the multiplexed bus)

Symbol	Parameter	Measurement Condition	Standard		Unit
			Min.	Max.	
$t_{d(BCLK-AD)}$	Address Output Delay Time	See Figure 26.2		18	ns
$t_{h(BCLK-AD)}$	Address Output Hold Time (BCLK standard)		-3		ns
$t_{h(RD-AD)}$	Address Output Hold Time (RD standard)(5)		(Note 1)		ns
$t_{h(WR-AD)}$	Address Output Hold Time (WR standard)(5)		(Note 1)		ns
$t_{d(BCLK-CS)}$	Chip-Select Signal Output Delay Time			18	ns
$t_{h(BCLK-CS)}$	Chip-Select Signal Output Hold Time (BCLK standard)		-3		ns
$t_{h(RD-CS)}$	Chip-Select Signal Output Hold Time (RD standard)(5)		(Note 1)		ns
$t_{h(WR-CS)}$	Chip-Select Signal Output Hold Time (WR standard)(5)		(Note 1)		ns
$t_{d(BCLK-RD)}$	RD Signal Output Delay Time			18	ns
$t_{h(BCLK-RD)}$	RD Signal Output Hold Time		-5		ns
$t_{d(BCLK-WR)}$	WR Signal Output Delay Time			18	ns
$t_{h(BCLK-WR)}$	WR Signal Output Hold Time		-5		ns
$t_{d(DB-WR)}$	Data Output Delay Time (WR standard)		(Note 2)		ns
$t_{h(WR-DB)}$	Data Output Hold Time (WR standard)(5)		(Note 1)		ns
$t_{d(BCLK-ALE)}$	ALE Signal Output Delay Time (BCLK standard)			18	ns
$t_{h(BCLK-ALE)}$	ALE Signal Output Hold Time (BCLK standard)		-2		ns
$t_{d(AD-ALE)}$	ALE Signal Output Delay Time (address standard)		(Note 3)		ns
$t_{h(ALE-AD)}$	ALE Signal Output Hold Time (address standard)		(Note 4)		ns
$t_{dz(RD-AD)}$	Address Output Float Start Time			8	ns

NOTES:

1. Values can be obtained from the following equations, according to BCLK frequency.

$$t_{h(RD-AD)} = \frac{10^{9}}{f_{(BCLK)} \times 2} - 10 \quad [ns]$$

$$t_{h(WR-AD)} = \frac{10^{9}}{f_{(BCLK)} \times 2} - 10 \quad [ns]$$

$$t_{h(RD-CS)} = \frac{10^{9}}{f_{(BCLK)} \times 2} - 10 \quad [ns]$$

$$t_{h(WR-CS)} = \frac{10^{9}}{f_{(BCLK)} \times 2} - 10 \quad [ns]$$

$$t_{h(WR-DB)} = \frac{10^{9}}{f_{(BCLK)} \times 2} - 10 \quad [ns]$$

2. Values can be obtained from the following equations, according to BCLK frequency and external bus cycle.

$$t_{d(DB-WR)} = \frac{10^{9} \times m}{f_{(BCLK)} \times 2} - 25 \quad [ns] \quad \text{(if external bus cycle is } a\phi + b\phi,\ m = (b \times 2) - 1)$$

3. Values can be obtained from the following equations, according to BCLK frequency and external bus cycle.

$$t_{d(AD-ALE)} = \frac{10^{9} \times n}{f_{(BCLK)} \times 2} - 20 \quad [ns] \quad \text{(if external bus cycle is } a\phi + b\phi,\ n = a)$$

4. Values can be obtained from the following equations, according to BCLK frequency and external bus cycle.

$$t_{h(ALE-AD)} = \frac{10^{9} \times n}{f_{(BCLK)} \times 2} - 10 \quad [ns] \quad \text{(if external bus cycle is } a\phi + b\phi,\ n = a)$$

5. *tc* ns is added when recovery cycle is inserted.

VCC=5V

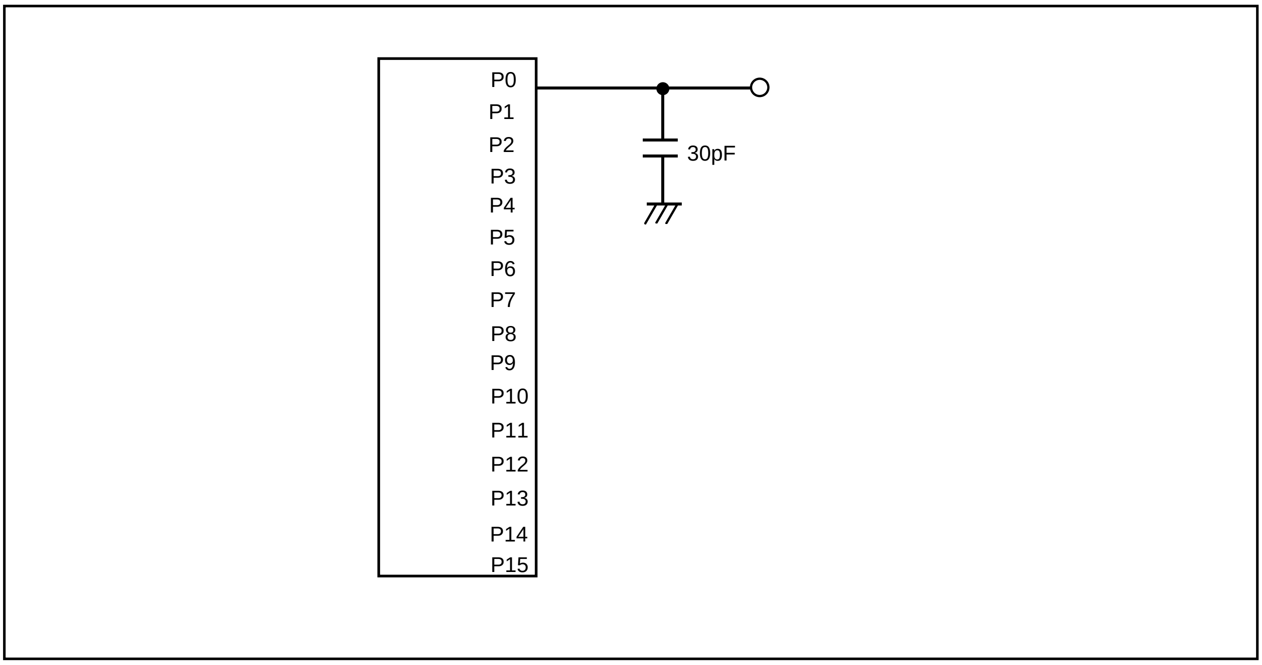


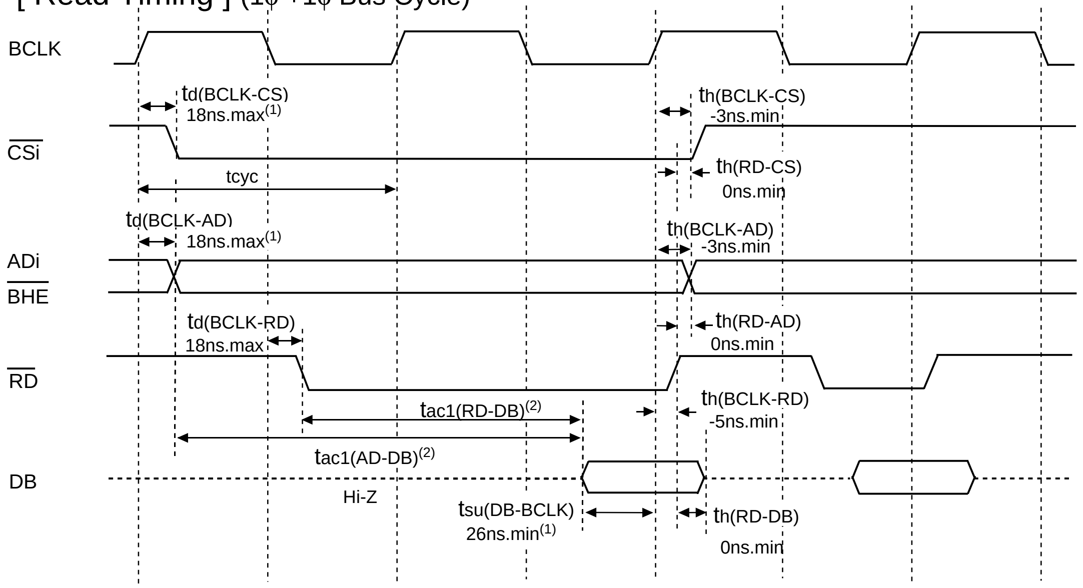
Figure 26.2 P0 to P15 Measurement Circuit

Vcc=5V

Memory Expansion Mode and Microprocessor Mode

(when accessing an external memory space)

[Read Timing] (1ϕ +1ϕ Bus Cycle)



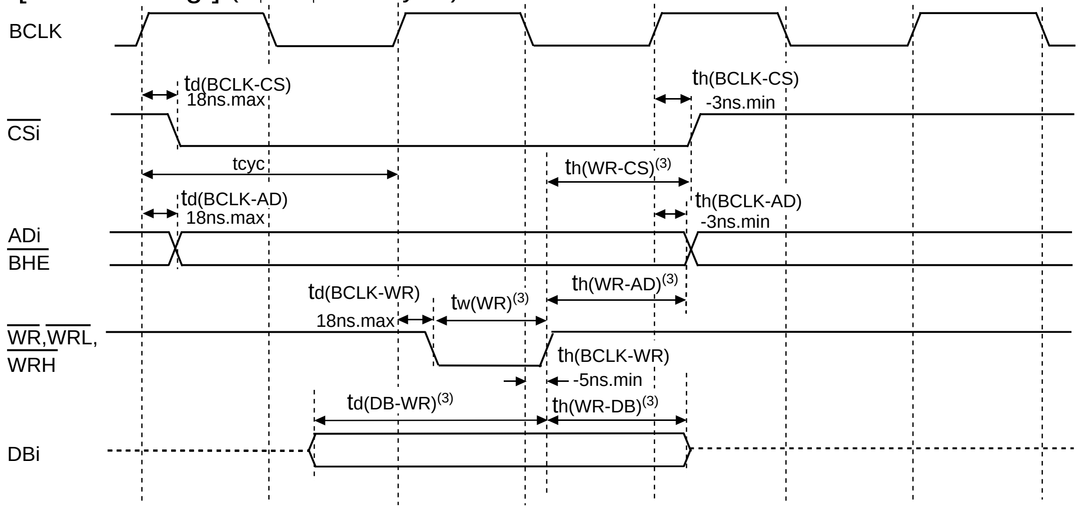
NOTES:

1. Values guaranteed only when the microcomputer is used independently.
   A maximum of 35ns is guaranteed for $t_{d(BCLK-AD)}+t_{su(DB-BCLK)}$.
2. Varies with operation frequency:

   $t_{ac1(RD-DB)}$=(tcyc/2 x m-35)ns.max (if external bus cycle is aϕ + bϕ, m=(b x 2)+1)

   $t_{ac1(AD-DB)}$=(tcyc x n-35)ns.max (if external bus cycle is aϕ + bϕ, n=a+b)

[Write timing] (1ϕ +1ϕ Bus Cycle)



NOTES:

3. Varies with operation frequency:
   $t_{d(DB-WR)}$=(tcyc x m-20)ns.min
   (if external bus cycle is aϕ+bϕ, m=b)
   $t_{h(WR-DB)}$=(tcyc/2-10)ns.min
   $t_{h(WR-AD)}$=(tcyc/2-10)ns.min
   $t_{h(WR-CS)}$=(tcyc/2-10)ns.min
   $t_{w(WR)}$=(tcyc/2 x n-15)ns.min
   (if external bus cycle is aϕ+bϕ , n=(bx2)-1)

Measurement Conditions:

- V_{CC}=4.2 to 5.5V
- Input high and low voltage: V_{IH}=2.5V, V_{IL}=0.8V
- Output high and low voltage: V_{OH}=2.0V, V_{OL}=0.8V

$$tcyc= \frac{10^9}{f_{(BCLK)}}$$

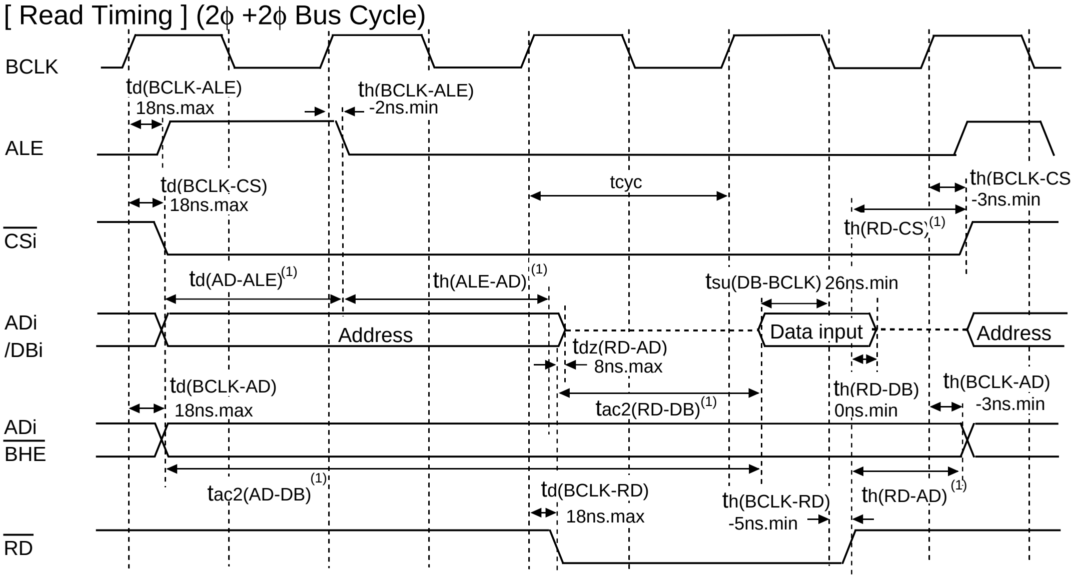
Figure 26.3 Vcc=5V Timing Diagram (1)

Vcc=5V

Memory Expansion Mode and Microprocessor Mode

(when accessing an external memory space with the multiplexed bus)

[Read Timing] (2ϕ +2ϕ Bus Cycle)



NOTES:

1. Varies with operation frequency:

$t_{d(AD-ALE)}$=(tcyc/2 x n-20)ns.min (if external bus cycle is aϕ + bϕ, n=a)

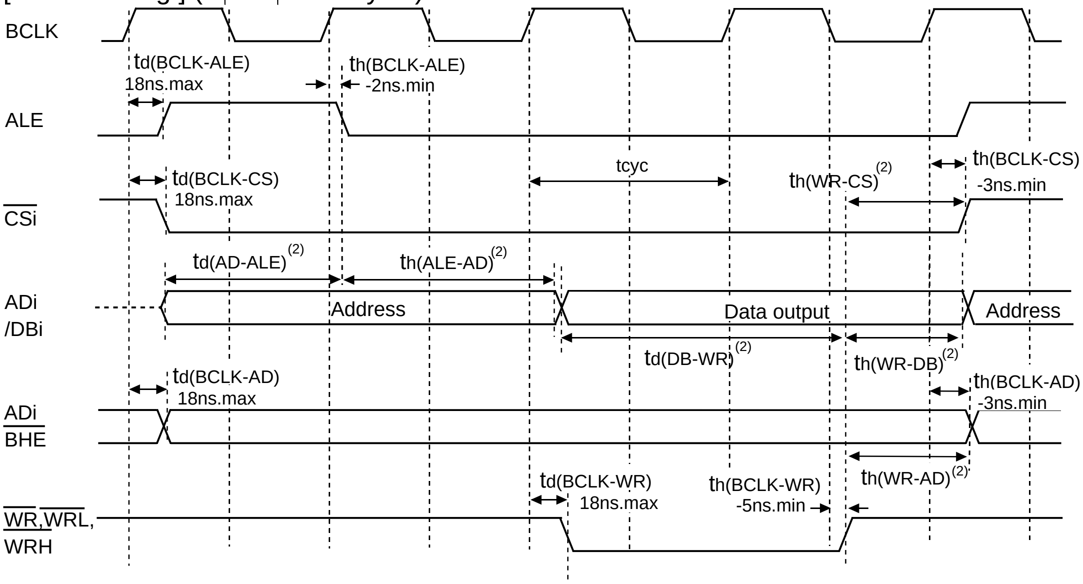
$t_{h(ALE-AD)}$=(tcyc/2 x n-10)ns.min (if external bus cycle is aϕ + bϕ, n=a)

$t_{h(RD-AD)}$=(tcyc/2-10)ns.min, $t_{h(RD-CS)}$=(tcyc/2-10)ns.min

$t_{ac2(RD-DB)}$=(tcyc/2 x m-35)ns.max (if external bus cycle is aϕ + bϕ, m=(b x 2)-1)

$t_{ac2(AD-DB)}$=(tcyc/2 x p-35)ns.max (if external bus cycle is aϕ + bϕ, p={(a+b-1) x 2}+1)

[Write Timing] (2ϕ +2ϕ Bus Cycle)



NOTES:

2. Varies with operation frequency:

$t_{d(AD-ALE)}$=(tcyc/2 x n - 20)ns.min

(if external bus cycle is aϕ + bϕ, n=a)

$t_{h(ALE-AD)}$=(tcyc/2 x n -10)ns.min

(if external bus cycle is aϕ + bϕ, n=a)

$t_{h(WR-AD)}$=(tcyc/2-10)ns.min,

$t_{h(WR-CS)}$=(tcyc/2-10)ns.min, $t_{h(WR-DB)}$=(tcyc/2-10)ns.min

$t_{d(DB-WR)}$=(tcyc/2 x m-25)ns.min

(if external bus cycle is aϕ + bϕ, m=(b x 2)-1)

Measurement Conditions:

- V_{CC}=4.2 to 5.5V
- Input high and low voltage:
  V_{IH}=2.5V, V_{IL}=0.8V
- Output high and low voltage:
  V_{OH}=2.0V, V_{OL}=0.8V

$$t_{cyc}=\frac{10^9}{f(BCLK)}$$

Figure 26.4 VCC=5V Timing Diagram (2)

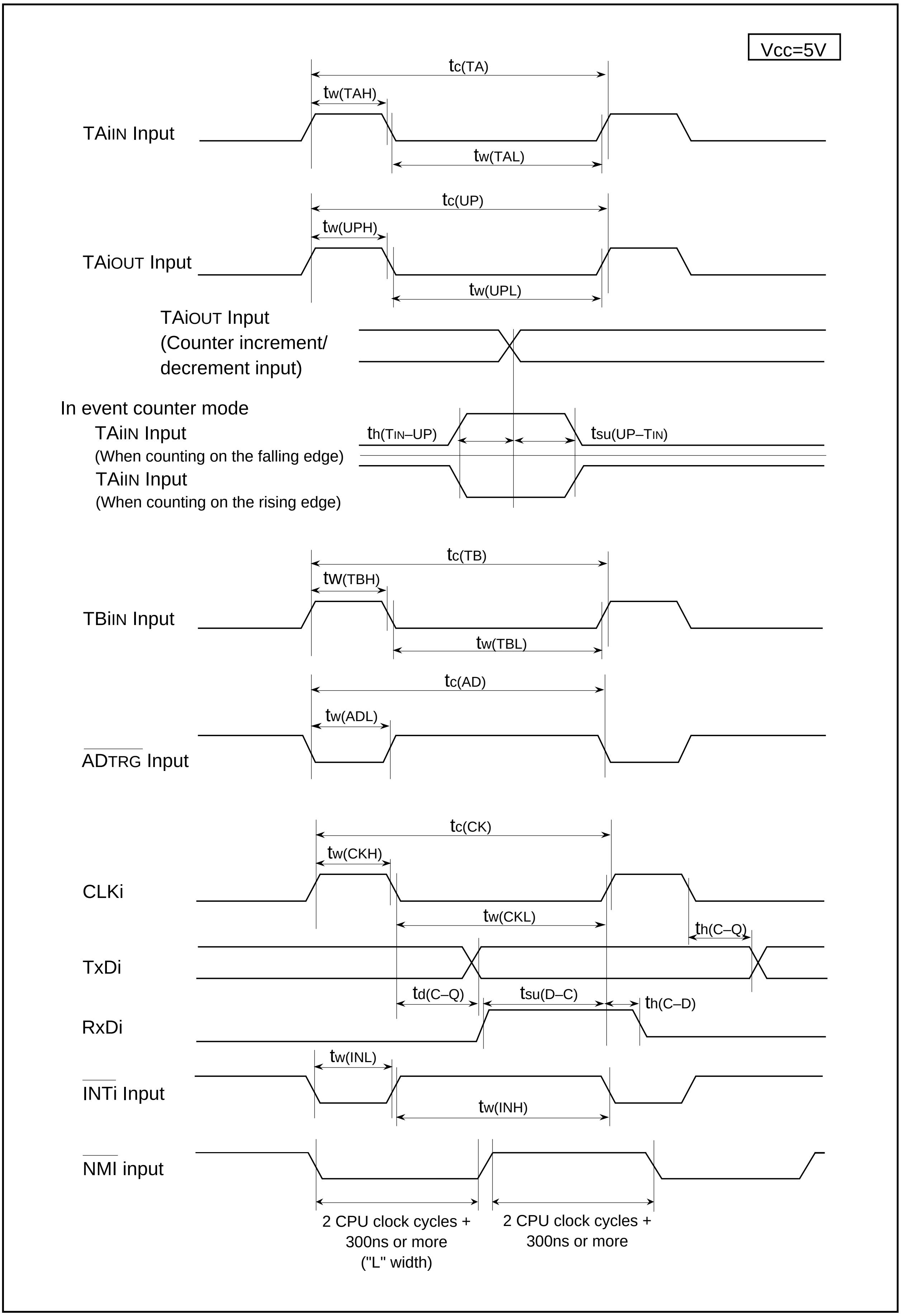
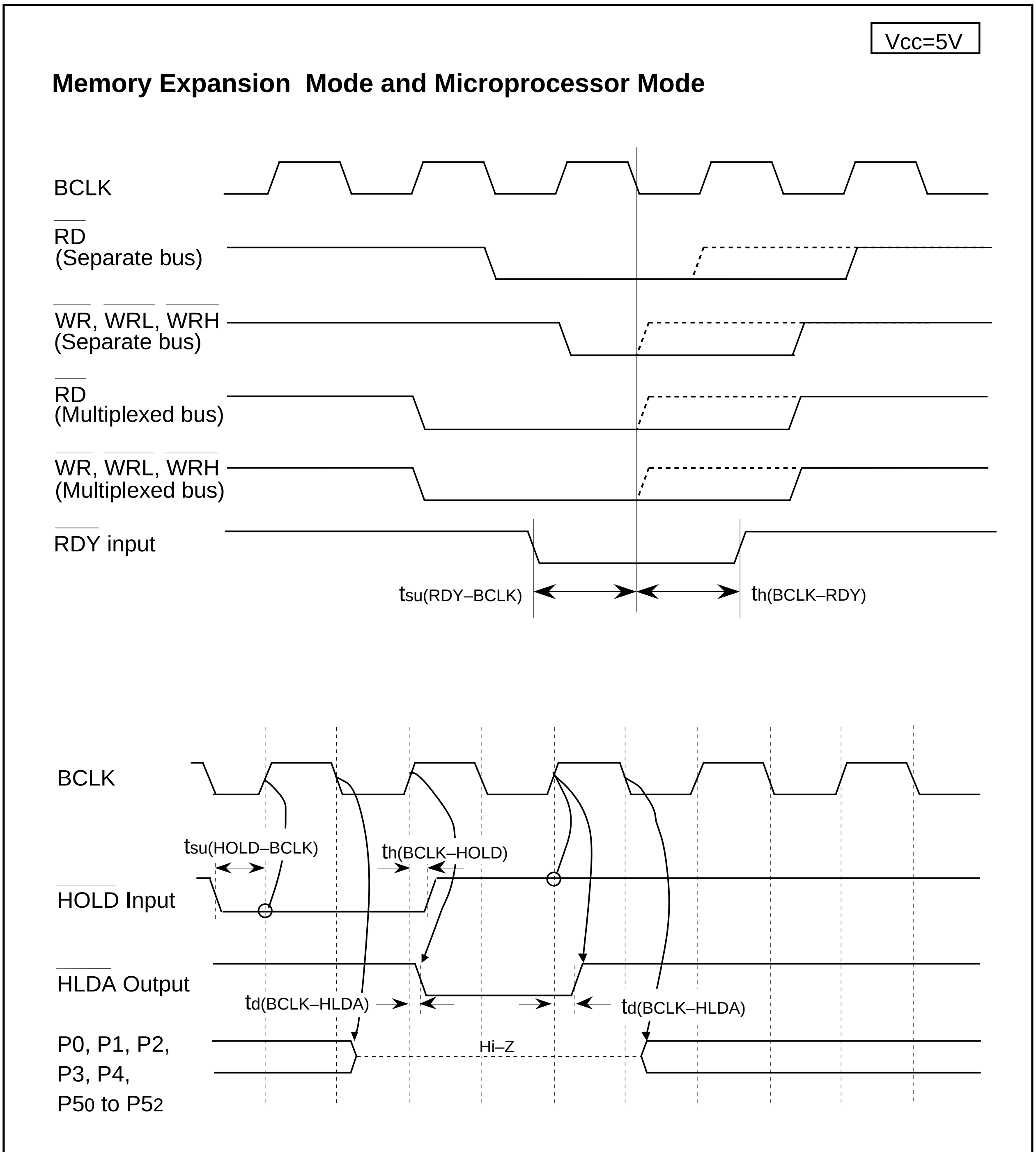


Figure 26.5 Vcc=5V Timing Diagram (3)



Measurement Conditions
- V_{CC}=4.2 to 5.5V
- Input high and low voltage: V_{IH}=4.0V, V_{IL}=1.0V
- Output high and low voltage: V_{OH}=2.5V, V_{OL}=2.5V

Figure 26.6 Vcc=5V Timing Diagram (4)

26.2 Electrical Characteristics (M32C/86T)

Table 26.24 Absolute Maximum Ratings

Symbol	Parameter		Condition	Value	Unit
V$_{CC}$	Supply Voltage		V$_{CC}$=AV$_{CC}$	-0.3 to 6.0	V
AV$_{CC}$	Analog Supply Voltage		V$_{CC}$=AV$_{CC}$	-0.3 to 6.0	V
V$_I$	Input Voltage	P0$_0$-P0$_7$, P1$_0$-P1$_7$, P2$_0$-P2$_7$, P3$_0$-P3$_7$, P4$_0$-P4$_7$, P5$_0$-P5$_7$, P6$_0$-P6$_7$, P7$_2$-P7$_7$, P8$_0$-P8$_7$, P9$_0$-P9$_7$, P10$_0$-P10$_7$, P11$_0$-P11$_4$, P12$_0$-P12$_7$, P13$_0$-P13$_7$, P14$_0$-P14$_6$, P15$_0$-P15$_7$, V$_{REF}$, X$_{IN}$, $\overline{\text{RESET}}$, CNV$_{SS}$, BYTE		-0.3 to V$_{CC}$+0.3	V
		P7$_0$, P7$_1$		-0.3 to 6.0	V
V$_O$	Output Voltage	P0$_0$-P0$_7$, P1$_0$-P1$_7$, P2$_0$-P2$_7$, P3$_0$-P3$_7$, P4$_0$-P4$_7$, P5$_0$-P5$_7$, P6$_0$-P6$_7$, P7$_2$-P7$_7$, P8$_0$-P8$_7$, P9$_0$-P9$_7$, P10$_0$-P10$_7$, P11$_0$-P11$_4$, P12$_0$-P12$_7$, P13$_0$-P13$_7$, P14$_0$-P14$_6$, P15$_0$-P15$_7$, X$_{OUT}$		-0.3 to V$_{CC}$+0.3	V
		P7$_0$, P7$_1$		-0.3 to 6.0	
Pd	Power Dissipation		Topr=25° C	500	mW
Topr	Operating Ambient Temperature	during CPU operation	T version	-40 to 85	° C
		during flash memory program and erase operation		0 to 60	
Tstg	Storage Temperature			-65 to 150	° C

Table 26.25 Recommended Operating Conditions

(V$_{CC}$=4.2 to 5.5 V, V$_{SS}$=0 V at Topr = -40 to 85^{o}C (T version) unless otherwise specified)

Symbol	Parameter		Standard			Unit
			Min.	Typ.	Max.	
V$_{CC}$	Supply Voltage		4.2	5.0	5.5	V
AV$_{CC}$	Analog Supply Voltage			V$_{CC}$		V
V$_{SS}$	Supply Voltage			0		V
AV$_{SS}$	Analog Supply Voltage			0		V
V$_{IH}$	Input High ("H") Voltage	P2$_0$-P2$_7$, P3$_0$-P3$_7$, P4$_0$-P4$_7$, P5$_0$-P5$_7$, P6$_0$-P6$_7$, P7$_2$-P7$_7$, P8$_0$-P8$_7$$^{(3)}$, P9$_0$-P9$_7$, P10$_0$-P10$_7$, P11$_0$-P11$_4$, P12$_0$-P12$_7$, P13$_0$-P13$_7$, P14$_0$-P14$_6$, P15$_0$-P15$_7$, X$_{IN}$, $\overline{\text{RESET}}$, CNV$_{SS}$, BYTE	0.8V$_{CC}$		V$_{CC}$	V
		P7$_0$, P7$_1$	0.8V$_{CC}$		6.0	
V$_{IL}$	Input Low ("L") Voltage	P2$_0$-P2$_7$, P3$_0$-P3$_7$, P4$_0$-P4$_7$, P5$_0$-P5$_7$, P6$_0$-P6$_7$, P7$_2$-P7$_7$, P8$_0$-P8$_7$$^{(3)}$, P9$_0$-P9$_7$, P10$_0$-P10$_7$, P11$_0$-P11$_4$, P12$_0$-P12$_7$, P13$_0$-P13$_7$, P14$_0$-P14$_6$, P15$_0$-P15$_7$, X$_{IN}$, $\overline{\text{RESET}}$, CNV$_{SS}$, BYTE	0		0.2V$_{CC}$	V
I$_{OH(peak)}$	Peak Output High ("H") Current$^{(2)}$	P0$_0$-P0$_7$, P1$_0$-P1$_7$, P2$_0$-P2$_7$, P3$_0$-P3$_7$, P4$_0$-P4$_7$, P5$_0$-P5$_7$, P6$_0$-P6$_7$, P7$_2$-P7$_7$, P8$_0$-P8$_4$, P8$_6$, P8$_7$, P9$_0$-P9$_7$, P10$_0$-P10$_7$, P11$_0$-P11$_4$, P12$_0$-P12$_7$, P13$_0$-P13$_7$, P14$_0$-P14$_6$, P15$_0$-P15$_7$			-10.0	mA
		P12$_0$-P12$_7$, P13$_0$-P13$_7$$^{(4)}$			-20.0	
I$_{OH(avg)}$	Average Output High ("H") Current$^{(1)}$	P0$_0$-P0$_7$, P1$_0$-P1$_7$, P2$_0$-P2$_7$, P3$_0$-P3$_7$, P4$_0$-P4$_7$, P5$_0$-P5$_7$, P6$_0$-P6$_7$, P7$_2$-P7$_7$, P8$_0$-P8$_4$, P8$_6$, P8$_7$, P9$_0$-P9$_7$, P10$_0$-P10$_7$, P11$_0$-P11$_4$, P12$_0$-P12$_7$, P13$_0$-P13$_7$, P14$_0$-P14$_6$, P15$_0$-P15$_7$			-5.0	mA
		P12$_0$-P12$_7$, P13$_0$-P13$_7$$^{(4)}$			-10.0	
I$_{OL(peak)}$	Peak Output Low ("L") Current$^{(2)}$	P0$_0$-P0$_7$, P1$_0$-P1$_7$, P2$_0$-P2$_7$, P3$_0$-P3$_7$, P4$_0$-P4$_7$, P5$_0$-P5$_7$, P6$_0$-P6$_7$, P7$_0$-P7$_7$, P8$_0$-P8$_4$, P8$_6$, P8$_7$, P9$_0$-P9$_7$, P10$_0$-P10$_7$, P11$_0$-P11$_4$, P12$_0$-P12$_7$, P13$_0$-P13$_7$, P14$_0$-P14$_6$, P15$_0$-P15$_7$			10.0	mA
		P12$_0$-P12$_7$, P13$_0$-P13$_7$$^{(4)}$			20.0	
I$_{OL(avg)}$	Average Output Low ("L") Current$^{(1)}$	P0$_0$-P0$_7$, P1$_0$-P1$_7$, P2$_0$-P2$_7$, P3$_0$-P3$_7$, P4$_0$-P4$_7$, P5$_0$-P5$_7$, P6$_0$-P6$_7$, P7$_0$-P7$_7$, P8$_0$-P8$_4$, P8$_6$, P8$_7$, P9$_0$-P9$_7$, P10$_0$-P10$_7$, P11$_0$-P11$_4$, P12$_0$-P12$_7$, P13$_0$-P13$_7$, P14$_0$-P14$_6$, P15$_0$-P15$_7$			5.0	mA
		P12$_0$-P12$_7$, P13$_0$-P13$_7$$^{(4)}$			10.0	

NOTES:

1. Typical values when average output current is 100ms.
2. Total I$_{OL(peak)}$ for P0, P1, P2, P8$_6$, P8$_7$, P9, P10, P11, P14 and P15 must be 80 mA or less.
   Total I$_{OL(peak)}$ for P3, P4, P5, P6, P7, P8$_0$ to P8$_4$, P12 and P13 must be 80 mA or less.
   Total I$_{OH(peak)}$ for P0, P1, P2, and P11 must be -40 mA or less.
   Total I$_{OH(peak)}$ for P8$_6$, P8$_7$, P9, P10, P14 and P15 must be -40 mA or less.
   Total I$_{OH(peak)}$ for P3, P4, P5, P12 and P13 must be -40 mA or less.
   Total I$_{OH(peak)}$ for P6, P7, and P8$_0$ to P8$_4$ must be -40 mA or less.
3. The V$_{IH}$ and V$_{IL}$ reference for P8$_7$ applies when P8$_7$ is used as a programmable input port.
   It does not apply when P8$_7$ is used as X$_{CIN}$.
4. This reference applies when the stepping motor control function is used.
   I$_{OL(avg)}$ must be 80 mA or less. I$_{OL(peak)}$ must be 120 mA or less.
   I$_{OH(avg)}$ must be -80 mA or less. I$_{OH(peak)}$ must be -120 mA or less.

Table 26.26 Recommended Operating Conditions (Continued)

(V_{CC}=4.2 to 5.5 V, V_{SS}=0 V at Topr = -40 to 85^{o}C (T version) unless otherwise specified)

Symbol	Parameter		Standard			Unit
			Min.	Typ.	Max.	
$f_{(BCLK)}$	CPU Clock Frequency	V_{CC}=4.2 to 5.5V	0		32	MHz
$f_{(X_{IN})}$	Main Clock Input Frequency	V_{CC}=4.2 to 5.5V	0		32	MHz
$f_{(X_{CIN})}$	Sub Clock Frequency			32.768	50	kHz
$f_{(Ring)}$	On-chip Oscillator Frequency (V_{CC}=5.0V, Topr=25° C)		0.5	1	2	MHz
$f_{(PLL)}$	PLL Clock Frequency	V_{CC}=4.2 to 5.5V	10		32	MHz
$t_{SU(PLL)}$	Wait Time to Stabilize PLL Frequency Synthesizer	V_{CC}=5.0V			5	ms

V_{CC}=5V

Table 26.27 Electrical Characteristics
(V_{CC}=4.2 to 5.5 V, V_{SS}=0 V at Topr = -40 to 85°C (T version), f(BCLK)=32MHz unless otherwise specified)

Symbol	Parameter			Condition	Standard			Unit
					Min.	Typ.	Max.	
V_{OH}	Output High ("H") Voltage	$P0_0$-$P0_7$, $P1_0$-$P1_7$, $P2_0$-$P2_7$, $P3_0$-$P3_7$, $P4_0$-$P4_7$, $P5_0$-$P5_7$, $P6_0$-$P6_7$, $P7_2$-$P7_7$, $P8_0$-$P8_4$, $P8_6$, $P8_7$, $P9_0$-$P9_7$, $P10_0$-$P10_7$, $P11_0$-$P11_4$, $P12_0$-$P12_7$, $P13_0$-$P13_7$, $P14_0$-$P14_6$, $P15_0$-$P15_7$		I_{OH}=-5 mA	V_{CC}-2.0		V_{CC}	V
		$P0_0$-$P0_7$, $P1_0$-$P1_7$, $P2_0$-$P2_7$, $P3_0$-$P3_7$, $P4_0$-$P4_7$, $P5_0$-$P5_7$, $P6_0$-$P6_7$, $P7_2$-$P7_7$, $P8_0$-$P8_4$, $P8_6$, $P8_7$, $P9_0$-$P9_7$, $P10_0$-$P10_7$, $P11_0$-$P11_4$, $P12_0$-$P12_7$, $P13_0$-$P13_7$, $P14_0$-$P14_6$, $P15_0$-$P15_7$		I_{OH}=-200 µA	V_{CC}-0.3		V_{CC}	V
		$P12_0$-$P12_7$, $P13_0$-$P13_7$(1)		I_{OH}=-10 mA	V_{CC}-2.0			V
		X_{OUT}		I_{OH}=-1 mA	3.0			V
		X_{COUT}	High Power	No load applied		2.5		V
			Low Power	No load applied		1.6		
V_{OL}	Output Low ("L") Voltage	$P0_0$-$P0_7$, $P1_0$-$P1_7$, $P2_0$-$P2_7$, $P3_0$-$P3_7$, $P4_0$-$P4_7$, $P5_0$-$P5_7$, $P6_0$-$P6_7$, $P7_0$-$P7_7$, $P8_0$-$P8_4$, $P8_6$, $P8_7$, $P9_0$-$P9_7$, $P10_0$-$P10_7$, $P11_0$-$P11_4$, $P12_0$-$P12_7$, $P13_0$-$P13_7$, $P14_0$-$P14_6$, $P15_0$-$P15_7$		I_{OL}=5 mA			2.0	V
		$P0_0$-$P0_7$, $P1_0$-$P1_7$, $P2_0$-$P2_7$, $P3_0$-$P3_7$, $P4_0$-$P4_7$, $P5_0$-$P5_7$, $P6_0$-$P6_7$, $P7_0$-$P7_7$, $P8_0$-$P8_4$, $P8_6$, $P8_7$, $P9_0$-$P9_7$, $P10_0$-$P10_7$, $P11_0$-$P11_4$, $P12_0$-$P12_7$, $P13_0$-$P13_7$, $P14_0$-$P14_6$, $P15_0$-$P15_7$		I_{OL}=200 µA			0.45	V
		$P12_0$-$P12_7$, $P13_0$-$P13_7$(1)		I_{OH}=-10 mA			2.0	V
		X_{OUT}		I_{OL}=1 mA			2.0	V
		X_{COUT}	High Power	No load applied		0		V
			Low Power	No load applied		0		
V_{T+}-V_{T-}	Hysteresis	$\overline{HOLD}$, $\overline{RDY}$, $TA0_{IN}$-$TA4_{IN}$, $TB0_{IN}$-$TB5_{IN}$, $\overline{INT0}$-$\overline{INT5}$, $\overline{AD_{TRG}}$, $\overline{CTS0}$-$\overline{CTS4}$, CLK0-CLK4, $TA0_{OUT}$-$TA4_{OUT}$, $\overline{NMI}$, $\overline{KI0}$-$\overline{KI3}$, RxD0-RxD4, SCL0-SCL4, SDA0-SDA4			0.2		1.0	V
		$\overline{RESET}$			0.2		1.8	V
I_{IH}	Input High ("H") Current	$P0_0$-$P0_7$, $P1_0$-$P1_7$, $P2_0$-$P2_7$, $P3_0$-$P3_7$, $P4_0$-$P4_7$, $P5_0$-$P5_7$, $P6_0$-$P6_7$, $P7_0$-$P7_7$, $P8_0$-$P8_7$, $P9_0$-$P9_7$, $P10_0$-$P10_7$, $P11_0$-$P11_4$, $P12_0$-$P12_7$, $P13_0$-$P13_7$, $P14_0$-$P14_6$, $P15_0$-$P15_7$, X_{IN}, $\overline{RESET}$, CNV_{SS}, BYTE		V_I=5V			5.0	µA
I_{IL}	Input Low ("L") Current	$P0_0$-$P0_7$, $P1_0$-$P1_7$, $P2_0$-$P2_7$, $P3_0$-$P3_7$, $P4_0$-$P4_7$, $P5_0$-$P5_7$, $P6_0$-$P6_7$, $P7_0$-$P7_7$, $P8_0$-$P8_7$, $P9_0$-$P9_7$, $P10_0$-$P10_7$, $P11_0$-$P11_4$, $P12_0$-$P12_7$, $P13_0$-$P13_7$, $P14_0$-$P14_6$, $P15_0$-$P15_7$, X_{IN}, $\overline{RESET}$, CNV_{SS}, BYTE		V_I=0V			-5.0	µA
R_{PULLUP}	Pull-up Resistance	$P0_0$-$P0_7$, $P1_0$-$P1_7$, $P2_0$-$P2_7$, $P3_0$-$P3_7$, $P4_0$-$P4_7$, $P5_0$-$P5_7$, $P6_0$-$P6_7$, $P7_2$-$P7_7$, $P8_0$-$P8_4$, $P8_6$, $P8_7$, $P9_0$-$P9_7$, $P10_0$-$P10_7$, $P11_0$-$P11_4$, $P12_0$-$P12_7$, $P13_0$-$P13_7$, $P14_0$-$P14_6$, $P15_0$-$P15_7$		V_I=0V	30	50	167	kΩ

NOTES:

1. This reference applies when the stepping motor control function is used.

V_{CC}=5V

Table 26.28 Electrical Characteristics (Continued)
(V_{CC}=4.2 to 5.5 V, V_{SS}=0 V at Topr = -40 to 85°C (T version), f(BCLK)=32MHz unless otherwise specified)

Symbol	Parameter	Measurement Condition		Standard			Unit
				Min.	Typ.	Max.	
Rf_{XIN}	Feedback Resistance	X_{IN}			1.5		MΩ
Rf_{XCIN}	Feedback Resistance	X_{CIN}			10		MΩ
V_{RAM}	RAM Standby Voltage	In stop mode		2.0			V
I_{CC}	Power Supply Current	In single-chip mode, output pins are left open and other pins are connected to V_{SS}.	f(BCLK)=32 MHz, Square wave, No division		28	50	mA
			f(BCLK)=32 kHz, In low-power consumption mode, Program running on ROM		430		µA
			f(BCLK)=32 kHz, In low-power consumption mode, Program running on RAM(1)		25		µA
			f(BCLK)=32 kHz, In wait mode, Topr=25° C		10		µA
			While clock stops, Topr=25° C		0.8	5	µA
			While clock stops, Topr=85° C			50	µA

NOTES:

1. Value is obtained when setting the FMSTP bit in the FMR0 register to "1" (flash memory stopped).

V_{CC}=5V

Table 26.29 A/D Conversion Characteristics (V_{CC}=4.2 to 5.5 V, V_{SS}=0 V at Topr= -40 to 85^{o}C (T version), f(BCLK)=32MHz unless otherwise specified)

Symbol	Parameter	Measurement Condition		Standard			Unit
				Min.	Typ.	Max.	
-	Resolution	V_{REF}=V_{CC}				10	Bits
INL	Integral Nonlinearity Error	V_{REF}=V_{CC}=5V	AN_0 to AN_7, $AN0_0$ to $AN0_7$, $AN2_0$ to $AN2_7$, $AN15_0$ to $AN15_7$, ANEX0, ANEX1			±3	LSB
							LSB
			External op-amp connection mode			±7	LSB
							LSB
DNL	Differential Nonlinearity Error					±1	LSB
-	Offset Error					±3	LSB
-	Gain Error					±3	LSB
R_{LADDER}	Resistor Ladder	V_{REF}=V_{CC}		8		40	kΩ
t_{CONV}	10-bit Conversion Time[1, 2]			2.06			µs
t_{CONV}	8-bit Conversion Time[1, 2]			1.75			µs
t_{SAMP}	Sampling Time[1]			0.188			µs
V_{REF}	Reference Voltage			2		V_{CC}	V
V_{IA}	Analog Input Voltage			0		V_{REF}	V

NOTES:

1. Divide f(X_{IN}), if exceeding 16 MHz, to keep ϕAD frequency at 16 MHz or less.
2. With using the sample and hold function.

Table 26.30 D/A Conversion Characteristics (V_{CC}=4.2 to 5.5 V, V_{SS}=0 V at Topr= -40 to 85^{o}C (T version), f(BCLK)=32MHz unless otherwise specified)

Symbol	Parameter	Measurement Condition	Standard			Unit
			Min.	Typ.	Max.	
-	Resolution				8	Bits
-	Absolute Accuracy				1.0	%
t_{SU}	Setup Time				3	µs
R_O	Output Resistance		4	10	20	kΩ
I_{VREF}	Reference Power Supply Input Current	(Note 1)			1.5	mA

NOTES:

1. Measurement when using one D/A converter. The DAi register (i=0, 1) of the D/A converter, not being used, is set to "00_{16}". The resistor ladder in the A/D converter is excluded.
   I_{VREF} flows even if the VCUT bit in the AD0CON1 register is set to "0" (no V_{REF} connection).

VCC=5V

Table 26.31 Flash Memory Version Electrical Characteristics
(VCC=4.5 to 5.5 V at Topr= 0 to 60°C unless otherwise specified)

Symbol	Parameter		Standard			Unit
			Min.	Typ.	Max.	
-	Program and Erase Endurance[2]		100			cycles
-	Word Program Time (VCC=5.0V, Topr=25° C)			25	200	µs
-	Lock Bit Program Time			25	200	µs
-	Block Erase Time (VCC=5.0V, Topr=25° C)	4-Kbyte Block		0.3	4	s
		8-Kbyte Block		0.3	4	s
		32-Kbyte Block		0.5	4	s
		64-Kbyte Block		0.8	4	s
-	All-Unlocked-Block Erase Time[1]				4 x *n*	s
t_{PS}	Wait Time to Stabilize Flash Memory Circuit				15	µs
-	Data Hold Time (Topr=-40 to 85 ° C)		10			years

NOTES:

1. *n* denotes the number of block to be erased.
2. Number of program-erase cycles per block.
   If Program and Erase Endurance is *n* cycle (*n*=100), each block can be erased and programmed *n* cycles.
   For example, if a 4-Kbyte block A is erased after programming a word data 2,048 times, each to a different address, this counts as one program and erase endurance. Data can not be programmed to the same address more than once without erasing the block. (rewrite prohibited).

Table 26.32 Power Supply Timing

Symbol	Parameter	Measurement Condition	Standard			Unit
			Min.	Typ.	Max.	
td(P-R)	Wait Time to Stabilize Internal Supply Voltage when Power-on	VCC=4.2 to 5.5 V			2	ms

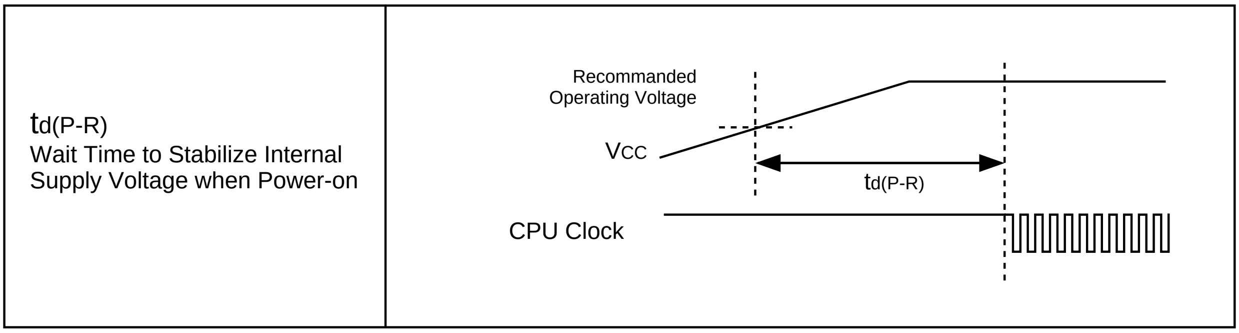

td(P-R)
Wait Time to Stabilize Internal Supply Voltage when Power-on
Recommanded Operating Voltage
VCC
td(P-R)
CPU Clock

Figure 26.8 Power Supply Timing Diagram

VCC=5V

Timing Requirements

(VCC=4.2 to 5.5 V, VSS=0 V at Topr= -40 to 85°C (T version) unless otherwise specified)

Table 26.33 External Clock Input

Symbol	Parameter	Standard		Unit
		Min.	Max.	
tc	External Clock Input Cycle Time	31.25		ns
$t_{w(H)}$	External Clock Input High ("H") Width	13.75		ns
$t_{w(L)}$	External Clock Input Low ("L") Width	13.75		ns
tr	External Clock Rise Time		5	ns
tf	External Clock Fall Time		5	ns

VCC=5V

Timing Requirements

(VCC=4.2 to 5.5 V, VSS=0 V at Topr= -40 to 85°C (T version) unless otherwise specified)

Table 26.34 Timer A Input (Count Source Input in Event Counter Mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
tc(TA)	TAiIN Input Cycle Time	100		ns
tw(TAH)	TAiIN Input High ("H") Width	40		ns
tw(TAL)	TAiIN Input Low ("L") Width	40		ns

Table 26.35 Timer A Input (Gate Input in Timer Mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
tc(TA)	TAiIN Input Cycle Time	400		ns
tw(TAH)	TAiIN Input High ("H") Width	200		ns
tw(TAL)	TAiIN Input Low ("L") Width	200		ns

Table 26.36 Timer A Input (External Trigger Input in One-Shot Timer Mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
tc(TA)	TAiIN Input Cycle Time	200		ns
tw(TAH)	TAiIN Input High ("H") Width	100		ns
tw(TAL)	TAiIN Input Low ("L") Width	100		ns

Table 26.37 Timer A Input (External Trigger Input in Pulse Width Modulation Mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
tw(TAH)	TAiIN Input High ("H") Width	100		ns
tw(TAL)	TAiIN Input Low ("L") Width	100		ns

Table 26.38 Timer A Input (Counter Increment/Decrement Input in Event Counter Mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
tc(UP)	TAiOUT Input Cycle Time	2000		ns
tw(UPH)	TAiOUT Input High ("H") Width	1000		ns
tw(UPL)	TAiOUT Input Low ("L") Width	1000		ns
tsu(UP-TIN)	TAiOUT Input Setup Time	400		ns
th(TIN-UP)	TAiOUT Input Hold Time	400		ns

V_{CC}=5V

Timing Requirements

(V_{CC}=4.2 to 5.5 V, V_{SS}=0 V at Topr= -40 to 85°C (T version) unless otherwise specified)

Table 26.39 Timer B Input (Count Source Input in Event Counter Mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TB)}$	TBi$_{IN}$ Input Cycle Time (counted on one edge)	100		ns
$t_{w(TBH)}$	TBi$_{IN}$ Input High ("H") Width (counted on one edge)	40		ns
$t_{w(TBL)}$	TBi$_{IN}$ Input Low ("L") Width (counted on one edge)	40		ns
$t_{c(TB)}$	TBi$_{IN}$ Input Cycle Time (counted on both edges)	200		ns
$t_{w(TBH)}$	TBi$_{IN}$ Input High ("H") Width (counted on both edges)	80		ns
$t_{w(TBL)}$	TBi$_{IN}$ Input Low ("L") Width (counted on both edges)	80		ns

Table 26.40 Timer B Input (Pulse Period Measurement Mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TB)}$	TBi$_{IN}$ Input Cycle Time	400		ns
$t_{w(TBH)}$	TBi$_{IN}$ Input High ("H") Width	200		ns
$t_{w(TBL)}$	TBi$_{IN}$ Input Low ("L") Width	200		ns

Table 26.41 Timer B Input (Pulse Width Measurement Mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TB)}$	TBi$_{IN}$ Input Cycle Time	400		ns
$t_{w(TBH)}$	TBi$_{IN}$ Input High ("H") Width	200		ns
$t_{w(TBL)}$	TBi$_{IN}$ Input Low ("L") Width	200		ns

Table 26.42 A/D Trigger Input

Symbol	Parameter	Standard		Unit
		Min.	Max	
$t_{c(AD)}$	$\overline{\text{AD}}_{\text{TRG}}$ Input Cycle Time (required for trigger)	1000		ns
$t_{w(ADL)}$	$\overline{\text{AD}}_{\text{TRG}}$ Input Low ("L") Pulse Width	125		ns

Table 26.43 Serial I/O

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(CK)}$	CLKi Input Cycle Time	200		ns
$t_{w(CKH)}$	CLKi Input High ("H") Width	100		ns
$t_{w(CKL)}$	CLKi Input Low ("L") Width	100		ns
$t_{d(C-Q)}$	TxDi Output Delay Time		80	ns
$t_{h(C-Q)}$	TxDi Hold Time	0		ns
$t_{su(D-C)}$	RxDi Input Setup Time	30		ns
$t_{h(C-Q)}$	RxDi Input Hold Time	90		ns

Table 26.44 External Interrupt $\overline{\text{INTi}}$ Input

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{w(INH)}$	$\overline{\text{INTi}}$ Input High ("H") Width	250		ns
$t_{w(INL)}$	$\overline{\text{INTi}}$ Input Low ("L") Width	250		ns

VCC=5V

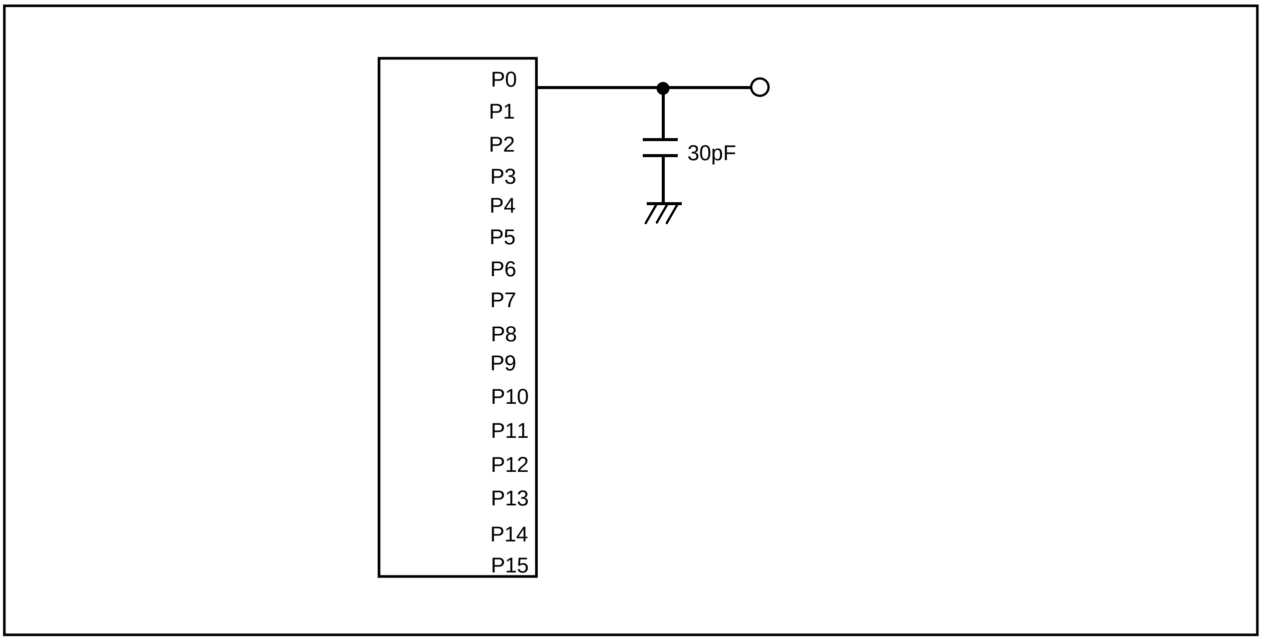


Figure 26.9 P0 to P15 Measurement Circuit

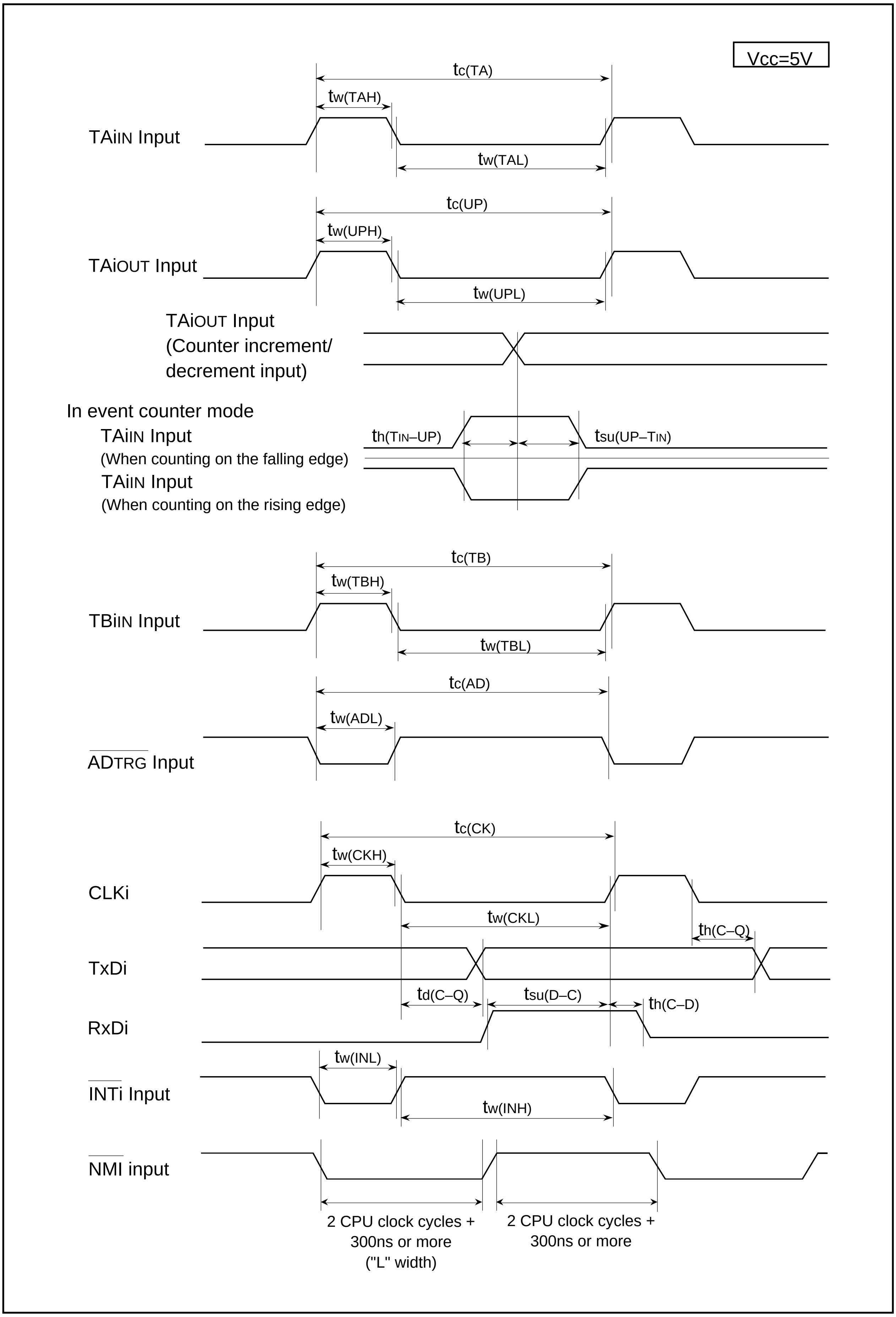


Figure 26.10 Vcc=5V Timing Diagram

27. Precautions

27.1 Restrictions to Use M32C/86T (High-Reliability Version)

The M32C/86T microcomputer (high-reliability version) has the following usage restrictions:

- M32C/86T must be used in single-chip mode only.
  M32C/86T cannot be used in memory expansion mode and microprocessor mode.

- Bus control pins (A0 to A22, A23, D0 to D15, $\overline{\text{CS0}}$ to $\overline{\text{CS3}}$, $\overline{\text{WRL}}$/$\overline{\text{WR}}$, $\overline{\text{WRH}}$/$\overline{\text{BHE}}$, $\overline{\text{RD}}$, BCLK/ALE, $\overline{\text{HLDA}}$/ALE, $\overline{\text{HOLD}}$, ALE, $\overline{\text{RDY}}$) and BCLK pins in M32C/86T cannot be used.

- The voltage detection circuit in M32C/86T cannot be used. Low voltage detection interrupt and brown-out detection reset cannot also be used.

- The DS register, VCR1 register, VCR2 register, D4INT register and EWCR0 to EWCR3 registers in M32C/86T cannot be used.

27.2 Reset

Voltage applied to the VCC pin must meet the SVCC standard.

Table 27.1 Power Supply Increasing Slope

Symbol	Parameter	Standard			Unit
		Min.	Typ.	Max.	
SVCC	Power Supply Increasing Slope (VCC)	0.05			V/ms

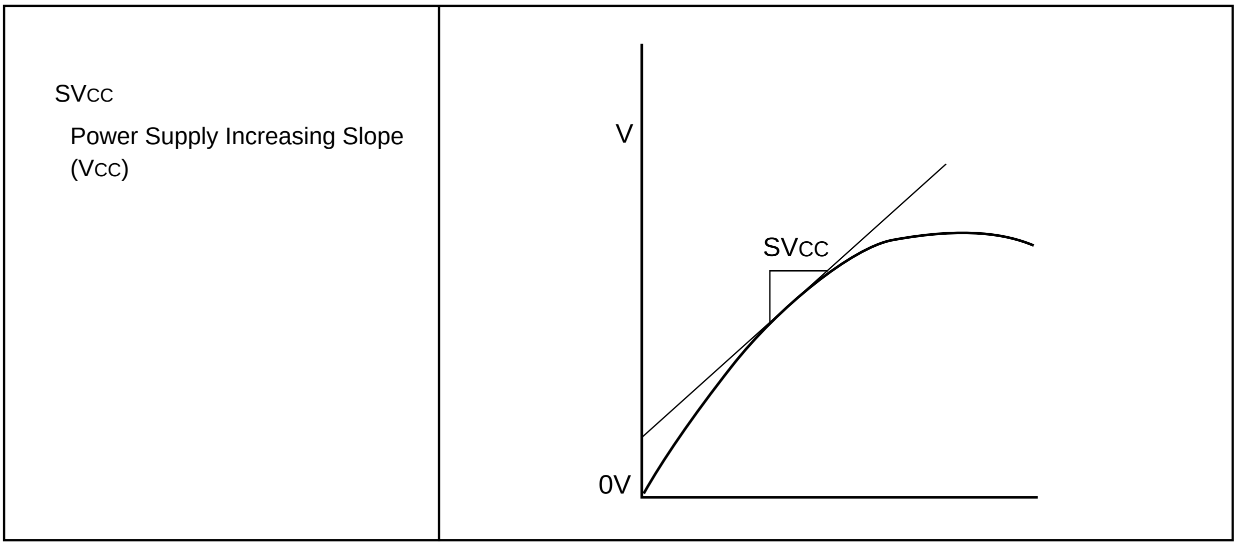


Figure 27.1 SVCC Timing

27.3 Bus

27.3.1 $\overline{\text{HOLD}}$ Signal

When entering microprocessor mode or memory expansion mode from single-chip mode and using $\overline{\text{HOLD}}$ input, set the PM01 and PM00 bits to "11_2" (microprocessor mode) or to "01_2" (memory expansion mode) after setting the PD4_7 to PD4_0 bits in the PD4 register and the PD5_2 to PD5_0 bits in the PD5 register to "0" (input mode).

$P4_0$ to $P4_7$ (A_{16} to A_{22}, $\overline{A_{23}}$, $\overline{\text{CS0}}$ to $\overline{\text{CS3}}$, MA8 to MA12) and $P5_0$ to $P5_2$ ($\overline{\text{RD}}/\overline{\text{WR}}/\overline{\text{BHE}}$, $\overline{\text{RD}}/\overline{\text{WRL}}/\overline{\text{WRH}}$) are not placed in high-impedance states even when a low-level ("L") signal is applied to the $\overline{\text{HOLD}}$ pin, if the PM01 and PM00 bits are set to "11_2" (microprocessor mode) or to "01_2" (memory expansion mode) after setting the PD4_7 to PD4_0 bits in the PD4 register and the PD5_2 to PD5_0 bits in the PD5 register to "1" (output mode) in single-chip mode.

27.3.2 External Bus

The internal ROM cannot be read when a high-level ("H") signal is applied to the CNV$_{SS}$ pin and the hardware reset (hardware reset 1 or brown-out detection reset) occurs.

27.4 SFR

27.4.1 Register Settings

Table 27.2 lists registers containing bits which can only be written to. Set these registers with immediate values. When establishing the next value by altering the present value, write the present value to the RAM as well as to the register. Transfer the next value to the register after making changes in the RAM.

Table 27.2 Registers with Write-only Bits

Register	Address		Register	Address
WDTS Register	$000E_{16}$		U3BRG Register	0329_{16}
G0RI Register	$00EC_{16}$		U3TB Register	$032B_{16}$, $032A_{16}$
G1RI Register	$012C_{16}$		U2BRG Register	0339_{16}
U1BRG Register	$02E9_{16}$		U2TB Register	$033B_{16}$, $033A_{16}$
U1TB Register	$02EB_{16}$, $02EA_{16}$		UDF Register	0344_{16}
U4BRG Register	$02F9_{16}$		TA0 Register(1)	0347_{16}, 0346_{16}
U4TB Register	$02FB_{16}$, $02FA_{16}$		TA1 Register(1)	0349_{16}, 0348_{16}
TA11 Register	0303_{16}, 0302_{16}		TA2 Register(1)	$034B_{16}$, $034A_{16}$
TA21 Register	0305_{16}, 0304_{16}		TA3 Register(1)	$034D_{16}$, $034C_{16}$
TA41 Register	0307_{16}, 0306_{16}		TA4 Register(1)	$034F_{16}$, $034E_{16}$
DTT Register	$030C_{16}$		U0BRG Register	0369_{16}
ICTB2 Register	$030D_{16}$		U0TB Register	$036B_{16}$, $36A_{16}$

NOTES :

1. In one-shot timer mode and pulse width modulation mode only.

27.5 Clock Generation Circuit

27.5.1 CPU Clock

- When the CPU operating frequency is 24 MHz or more, use the following procedure for better EMC (Electromagnetic Compatibility) performance.
  1) Oscillator connected between the XIN and XOUT pins, or external clock applied to the XIN pin, has less than 24 MHz frequency.
  2) Use the PLL frequency synthesizer to multiply the main clock.

- In M32C/86T, the main clock frequency must be 24 MHz or less.

27.5.2 Sub Clock

Set the CM03 bit to "0" (XCIN-XCOUT drive capacity "LOW") when selecting the sub clock (XCIN-XCOUT) as the CPU clock, or Timer A or Timer B count source (fC32).

27.5.2.1 Sub Clock Oscillation

When oscillating the sub clock, set the CM04 bit in the CM0 register to "1" (XCIN-XCOUT oscillation function) after setting the CM07 bit in the CM0 register to "0" (clock other than sub clock) and the CM03 bit to "1" (XCIN-XCOUT drive capacity "HIGH"). Set the CM03 bit to "0" after sub clock oscillation stabilizes.

Set the sub clock as the CPU clock, or Timer A or Timer B count source (fC32) after the above settings are completed.

27.5.2.2 Using Stop Mode

When the microcomputer enters stop mode, the CM03 bit is automatically set to "1" (XCIN-XCOUT drive capacity "HIGH"). Use the following procedure to select the main clock as the CPU clock when entering stop mode.

1) Set the CM17 bit in the CM1 register to "0" (main clock).
2) Set the CM21 bit in the CM2 register to "0" (clock selected by the CM17 bit).
3) Set the CM07 bit in the CM0 register to "0" (clock selected by the CM21 bit divided by the MCD register setting).

After exiting stop mode, wait for the sub clock oscillation to stabilize. Then set the CM03 bit to "0" and the CM07 bit to "1" (sub clock).

27.5.2.3 Oscillation Parameter Matching

If the sub slock oscillation parameters have only been evaluated with the drive capacity "HIGH", the parameters should be reevaluated for drive capacity "LOW".

Contact your oscillator manufacturer for details on matching parameters.

27.5.3 PLL Frequency Synthesizer

Stabilize supply voltage to meet the power supply standard when using the PLL frequency synthesizer.

Table 27.3 Power Supply Ripple

Symbol	Parameter		Standard			Unit
			Min.	Typ.	Max.	
$f_{(ripple)}$	Power Supply Ripple Tolerable Frequency (V_{CC})	V_{CC1}=5V			10	kHz
$V_{P-P(ripple)}$	Power Supply Ripple Voltage Fluctuation Range	V_{CC1}=5V			0.5	V
$V_{CC(\lvert\Delta V/\Delta T\rvert)}$	Power Supply Ripple Voltage Fluctuation Rate	V_{CC1}=5V			1	V/ms

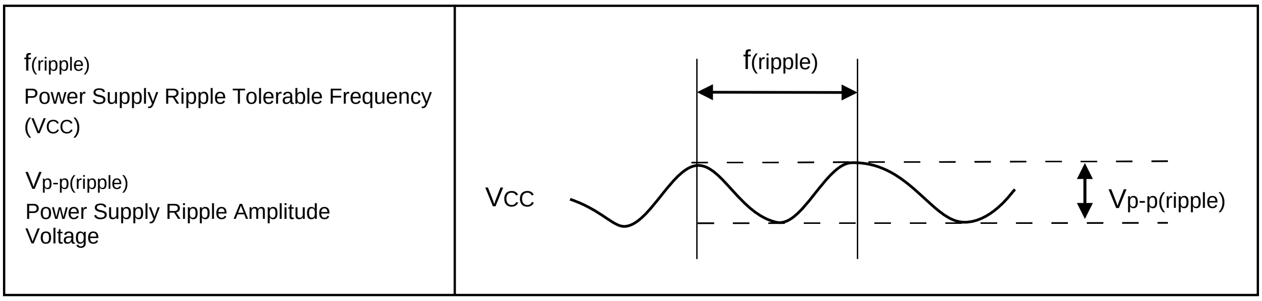

f(ripple)
Power Supply Ripple Tolerable Frequency (VCC)
Vp-p(ripple)
Power Supply Ripple Amplitude Voltage
f(ripple)
VCC
Vp-p(ripple)

Figure 27.2 Power Supply Fluctuation Timing

27.5.4 External Clock

Do not stop an external clock running if the main clock is selected as the CPU clock while the external clock is applied to the X_{IN} pin.

Do not set the CM05 bit in the CM0 register to "1" (main clock stopped) while the external clock input is used for the CPU clock.

27.5.5 Clock Divide Ratio

Set the PM12 bit in the PM1 register to "0" (no wait state) when changing the MCD4 to MCD0 bit settings in the MCD register.

27.5.6 Power Consumption Control

Stabilize the main clock, sub clock or PLL clock to switch the CPU clock source to each clock.

27.5.6.1 Wait Mode

When entering wait mode while the CM02 bit in the CM0 register is set to "1" (peripheral function stop in wait mode), set the MCD4 to MCD0 bits in the MCD register to maintain the 10-MHz CPU clock frequency or less.

When entering wait mode, the instruction queue reads ahead to instructions following the WAIT instruction, and the program stops. Write at least 4 NOP instructions after the WAIT instruction.

27.5.6.2 Stop Mode

- Use the following procedure to select the main clock as the CPU clock when entering stop mode.
  1) Set the CM17 bit in the CM1 register to "0" (main clock).
  2) Set the CM21 bit in the CM2 register to "0" (clock selected by the CM17 bit).
  3) Set the CM07 bit in the CM0 register to "0" (clock selected by the CM21 bit divided by the MCD register setting).

  If the PLL clock is selected as the CPU clock source, set the CM17 bit to "0" (main clock) and the PLC07 bit in the PLC0 register to "0" (PLL off) before entering stop mode.

- The microcomputer cannot enter stop mode if a low-level signal ("L") is applied to the $\overline{\text{NMI}}$ pin. Apply a high-level ("H") signal instead.

- If stop mode is exited by any reset, apply an "L" signal to the $\overline{\text{RESET}}$ pin until a main clock oscillation is stabilized enough.

- If using the $\overline{\text{NMI}}$ interrupt to exit stop mode, use the following procedure to set the CM10 bit in the CM1 register (all clocks stopped).
  1) Exit stop mode with using the $\overline{\text{NMI}}$ interrupt.
  2) Generate a dummy interrupt.
  3) Set the CM10 bit to "1".

```
 e.g.,          int     #63                  ; dummy interrupt
                bset    cm1                  ; all clocks stopped

                /* dummy interrupt handling */
            dummy
               reit
```

- When entering stop mode, the instruction queue reads ahead to instructions following the instruction setting the CM10 bit in the CM1 register to "1" (all clocks stopped), and the program stops. When the microcomputer exits stop mode, the instruction lined in the instruction queue is executed before the interrupt routine for recovery is done.

  Write the JMP.B instruction, as follows, after the instruction setting the CM10 bit in the CM1 register to "1" (all clocks stopped).

```
 e.g.,          bset 0, prcr                 ; protection removed
                bset 0, cm1                  ; all clocks stopped
                jmp.b LABEL_001              ; JMP.B instruction executed (no instuction between JMP.B
                                             ; and LABEL.)
        LABEL_001:
                nop                          ; NOP (1)
                nop                          ; NOP (2)
                nop                          ; NOP (3)
                nop                          ; NOP (4)
                mov.b #0, prcr               ; Protection set
                  •
                  •
                  •
```

27.5.6.3 Suggestions for Reducing Power Consumption

The followings are suggestions for reducing power consumption when programming or designing systems.

Ports: I/O ports maintains the same state despite the microcomputer entering wait mode or stop mode. Current flows through active output ports. Feedthrough current flows through input ports in a high-impedance state. Set unassigned ports as input ports and stabilize electrical potential before entering wait mode or stop mode.

A/D Converter: If the A/D conversion is not performed, set the VCUT bit in the AD0CON1 register to "0" (no V_{REF} connection). Set the VCUT bit to "1" (V_{REF} connection) and wait at least 1µs before starting the A/D conversion.

D/A Converter: Set the DAi bit (i=0, 1) in the DACON register to "0" (output disabled) and set the DAi register to "00_{16}" when the D/A conversion is not performed.

Peripheral Function Stop: Set the CM02 bit in the CM0 register while in wait mode to stop unnecessary peripheral functions. However, this does not reduce power consumption because the peripheral function clock (fc32) generating from the sub clock does not stop. When in low-speed mode and low-power consumption mode, do not enter wait mode when the CM02 bit is set to "1" (peripheral clock stops in wait mode).

27.6 Protection

The PRC2 bit setting in the PRCR register is changed to "0" (write disable) when an instruction is written to any address after the PRC2 bit is set to "1" (write enable). Write instruction immediately after setting the PRC2 bit to "1" to change registers protected by the PRC2 bit. Do not generate an interrupt or a DMA transfer between the instruction to set the PRC2 bit to "1" and the following instruction.

27.7 Interrupts

27.7.1 ISP Setting

After reset, the ISP is set to "000000_{16}". The program runs out of control if an interrupt is acknowledged before the ISP is set. Therefore, the ISP must be set before an interrupt request is generated. Set the ISP to an even address, which allows interrupt sequences to be executed at a higher speed.
To use NMI interrupt, set the ISP at the beginning of the program. The NMI interrupt can be acknowledged after the first instruction has been executed after reset.

27.7.2 NMI Interrupt

- NMI interrupt cannot be denied. Connect the NMI pin to V_{CC} via a resistor (pull-up) when not in use.

- The P8_5 bit in the P8 register indicates the NMI pin value. Read the P8_5 bit only to determine the pin level after a NMI interrupt occurs.

- "H" and "L" signals applied to the NMI pin must be over 2 CPU clock cycles + 300 ns wide.

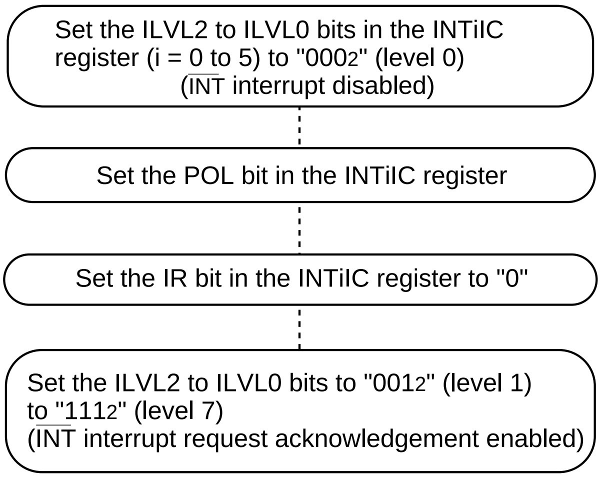
- NMI interrupt request may not be acknowledged if this and other interrupt requests are generated simultaneously.

27.7.3 INT Interrupt

- Edge Sensitive
  "H" and "L" signals applied to the INT0 to INT5 pins must be at least 250 ns wide, regardless of the CPU clock.

- Level Sensitive
  "H" and "L" signals applied to the INT0 to INT5 pins must be at least 1 CPU clock cycle + 200 ns wide. For example, "H" and "L" must be at least 234ns wide if X_{IN}=30MHz with no division.

- The IR bit setting may change to "1" (interrupt requested) when switching the polarity of the INT0 to INT5 pins. Set the IR bit to "0" (no interrupt requested) after selecting the polarity. Figure 27.3 shows an example of the switching procedure for the INT interrupt.



Set the ILVL2 to ILVL0 bits in the INTiIC register (i = 0 to 5) to "000_2" (level 0)
(INT interrupt disabled)

Set the POL bit in the INTiIC register

Set the IR bit in the INTiIC register to "0"

Set the ILVL2 to ILVL0 bits to "001_2" (level 1)
to "111_2" (level 7)
(INT interrupt request acknowledgement enabled)

Figure 27.3 Switching Procedure for INT Interrupt

27.7.4 Watchdog Timer Interrupt

Reset the watchdog timer after a watchdog timer interrupt occurs.

27.7.5 Changing Interrupt Control Register

To change the interrupt control register while the interrupt request is denied, follow the instructions below.

Changing IR bit

The IR bit setting may not change to "0" (no interrupt requested) depending on the instructions written. If this is a problem, use the following instruction to change the register: MOV

Changing Bits Except IR Bit

When an interrupt request is generated while executing an instruction, the IR bit may not be set to "1" (interrupt requested) and the interrupt may be ignored. If this is a problem, use the following instructions to change the register: AND, OR, BCLR, BSET

27.7.6 Changing IIOiIR Register (i = 0 to 5, 8 to 11)

Use the following instructions to set bits 1 to 7 in the IIOiIR register to "0" (no interrupt requested): AND, BCLR

27.7.7 Changing RLVL Register

The DMAII bit is indeterminate after reset. When using the DMAII bit to generate an interrupt, set the interrupt control register after setting the DMAII bit to "0" (interrupt priority level 7 available for interrupts).

27.8 DMAC

- Set DMAC-associated registers while the MDi1 and MDi0 bits (i=0 to 3) in the channel to be used are set to "00_2" (DMA disabled). Set the MDi1 and MDi0 bits to "01_2" (single transfer) or "11_2" (repeat transfer) at the end of setup procedure to start DMA requests.

- Do not set the DRQ bit in the DMiSL register to "0" (no request).
  If a DMA request is generated but the receiving channel is not ready to receive[1], the DMA transfer does not occur and the DRQ bit is set to "0".

NOTES:

1. The MDi1 and MDi0 bits are set to "00_2" or the DCTi register is set to "0000_{16}" (transferred 0 times).

- To start a DMA transfer by a software trigger, set the DSR bit and DRQ bit in the DMiSL register to "1" simultaneously.
  e.g.,

```
                OR.B #0A0h,DMiSL                ; Set the DSR and DRQ bits to "1" simultaneously
```

- Do not generate a channel i DMA request when setting the MDi1 and MDi0 bits in the DMDj register (j=0,1) corresponding to channel i to "01_2" (single transfer) or "11_2" (repeat transfer), if the DCTi register of channel i is set to "1".

- Select the peripheral function which causes the DMA request after setting the DMA-associated registers. If none of the conditions above (setting $\overline{\text{INT}}$ interrupt as DMA request source) apply, do not write "1" to the DCTi register.

- Enable DMA[2] after setting the DMiSL register (i=0 to 3) and waiting six BCLK cycles or more by program.

NOTES:

2. DMA is enabled when the values set in the MDi1 and MDi0 bits in the DMDj register are changed from "00_2" (DMA disabled) to "01_2" (single transfer) or "11_2" (repeat transfer).

27.9 Timer

27.9.1 Timers A and B

Timers stop after reset. Set the TAiS(i=0 to 4) bit or TBjS(j=0 to 5) bit in the TABSR register or TBSR register to "1" (starts counting) after setting operating mode, count source and counter.
The following registers and bits must be set while the TAiS bit or TBjS bit is set to "0" (stops counting).

- TAiMR, TBjMR register
- TAi, TBj register
- UDF register
- TAZIE, TA0TGL, TA0TGH bits in the ONSF register
- TRGSR register

27.9.2 Timer A

The TA1$_{OUT}$, TA2$_{OUT}$ and TA4$_{OUT}$ pins are placed in high-impedance states when a low-level ("L") signal is applied to the $\overline{\text{NMI}}$ pin while the INV03 and INV02 bits in the INVC0 register are set to "11$_2$" (forced cutoff of the three-phase output by an "L" signal applied to the $\overline{\text{NMI}}$ pin).

27.9.2.1 Timer A (Timer Mode)

- The TAiS bit (i=0 to 4) in the TABSR register is set to "0" (stops counting) after reset. Set the TAiS bit to "1" (starts counting) after selecting an operating mode and setting the TAi register.

- The TAi register indicates the counter value during counting at any given time. However, the counter is "FFFF$_{16}$" when reloading. The setting value can be read after setting the TAi register while the counter stops and before the counter starts counting.

27.9.2.2 Timer A (Event Counter Mode)

- The TAiS (i=0 to 4) bit in the TABSR register is set to "0" (stops counting) after reset. Set the TAiS bit to "1" (starts counting) after selecting an operating mode and setting the TAi register.

- The TAi register indicates the counter values during counting at any given time. However, the counter will be "FFFF$_{16}$" during underflow and "0000$_{16}$" during overflow, when reloading. The setting value can be read after setting the TAi register while the counter stops and before the counter starts counting.

27.9.2.3 Timer A (One-shot Timer Mode)

- The TAiS (i=0 to 4) bit in the TABSR register is set to "0" (stops counting) after reset. Set the TAiS bit to "1" (starts counting) after selecting an operating mode and setting the TAi register.

- The followings occur when the TABSR register is set to "0" (stops counting) while counting:
  - The counter stops counting and the microcomputer reloads contents of the reload register.
  - The TAi$_{OUT}$ pin becomes low ("L").
  - The IR bit in the TAiIC register is set to "1" (interrupt requested) after one CPU clock cycle.

- The output of the one-shot timer is synchronized with an internal count source. When set to an external trigger, there is a delay of one count source cycle maximum, from trigger input to the TAi$_{IN}$ pin to the one-shot timer output.

- The IR bit is set to "1" when the following procedures are performed to set timer mode:
  - selecting one-shot timer mode after reset.
  - switching from timer mode to one-shot timer mode.
  - switching from event counter mode to one-shot timer mode.

  Therefore, set the IR bit to "0" to generate a timer Ai interrupt (IR bit) after performing these procedures.

- When a trigger is generated while counting, the reload register reloads and continues counting after the counter has decremented once following a re-trigger. To generate a trigger while counting, wait at least 1 count source cycle after the previous trigger has been generated and generate a re-trigger.

- If an external trigger input is selected to start counting in timer A one-shot timer mode, do not provide another external trigger input again for 300 ns before the timer A counter value reaches "0000$_{16}$". One-shot timer may stop counting.

27.9.2.4 Timer A (Pulse Width Modulation Mode)

- The TAiS(i=0 to 4) bit in the TABSR register is set to "0" (stops counting) after reset. Set the TAiS bit to "1" (starts counting) after selecting an operating mode and setting the TAi register.

- The IR bit is set to "1" when the following procedures are performed to set timer mode:
  - Selecting PWM mode after reset
  - Switching from timer mode to PWM mode
  - Switching from event counter mode to PWM mode

  Therefore, set the IR bit to "0" by program to generate a timer Ai interrupt (IR bit) after performing these procedures.

- The followings occur when the TAiS bit is set to "0" (stops counting) while PWM pulse is output:
  - The counter stops counting
  - Output level changes to low ("L") and the IR bit changes to "1" when the TAi$_{OUT}$ pin is held high ("H")
  - The IR bit and the output level remain unchanged when TAi$_{OUT}$ pin is held "L"

27.9.3 Timer B

27.9.3.1 Timer B (Timer Mode, Event Counter Mode)

- The TBiS (i=0 to 5) bit is set to "0" (stops counting) after reset. Set the TBiS bit to "1" (starts counting) after selecting an operating mode and setting TBi register.
  The TB2S to TB0S bits are bits 7 to 5 in the TABSR register. The TB5S to TB3S bits are bits 7 to 5 in the TBSR register.

- The TBi register indicates the counter value during counting at any given time. However, the counter is "FFFF$_{16}$" when reloading. The setting value can be read after setting the TBi register while the counter stops and before the counter starts counting.

27.9.3.2 Timer B (Pulse Period/Pulse Width Measurement Mode)

- The IR bit in the TBiIC (i=0 to 5) register is set to "1" (interrupt requested) when the valid edge of a pulse to be measured is input and when the timer Bi counter overflows. The MR3 bit in the TBiMR register determines the interrupt source within an interrupt routine.

- Use another timer to count how often the timer counter overflows when an interrupt source cannot be determined by the MR3 bit, such as when a pulse to be measured is input at the same time the timer counter overflows.

- To set the MR3 bit in the TBiMR register to "0" (no overflow), set the TBiMR register after the MR3 bit is set to "1" (overflow) and one or more cycles of the count source are counted, while the TBiS bits in the TABSR and TBSR registers are set to "1" (starts counting).

- The IR bit in the TBiIC register is used to detect overflow only. Use the MR3 bit only to determine interrupt source within an interrupt routine.

- Indeterminate values are transferred to the reload register during the first valid edge input after counting is started. Timer Bi interrupt request is not generated at this time.

- The counter value is indeterminate when counting is started. Therefore, the MR3 bit setting may change to "1" (overflow) and causes timer Bi interrupt requests to be generated until a valid edge is input after counting is started.

- The IR bit may be set to "1" (interrupt requested) if the MR1 and MR0 bits in the TBiMR register are set to a different value after a count begins. If the MR1 and MR0 bits are rewritten, but to the same value as before, the IR bit remains unchanged.

- Pulse width measurement measures pulse width continuously. Use program to determine whether measurement results are high ("'H") or low ("L").

27.10 Serial I/O

27.10.1 Clock Synchronous Serial I/O Mode

The $\overline{\text{RTS}}$2 and CLK2 pins are placed in high-impedance states when a low-level ("L") signal is applied to the $\overline{\text{NMI}}$ pin while the INV03 to INV02 bits in the INVC0 register are set to "11_2" (forced cutoff of the three-phase output by an "L" signal applied to the $\overline{\text{NMI}}$ pin).

27.10.1.1 Transmission /Reception

When the $\overline{\text{RTS}}$ function is used while an external clock is selected, the output level of the $\overline{\text{RTSi}}$ pin is held "L" indicating that the microcomputer is ready for reception. The transmitting microcomputer is notified that reception is possible. The output level of the $\overline{\text{RTSi}}$ pin becomes high ("H") when reception begins. Therefore, connecting the $\overline{\text{RTSi}}$ pin to the $\overline{\text{CTSi}}$ pin of the transmitting microcomputer synchronizes transmission and reception. The $\overline{\text{RTS}}$ function is disabled if an internal clock is selected.

27.10.1.2 Transmission

When an external clock is selected while the CKPOL bit in the UiC0 (i=0 to 4) register is set to "0" (data is transmitted on the falling edge of the transfer clock and received on the rising edge) and the external clock is held "H", or when the CKPOL bit is set to "1" (data is transmitted on the rising edge of the transfer clock and received on the falling edge) and the external clock is held "L", meet the following conditions:

- Set the TE bit in the UiC1 register to "1" (receive enabled)
- Set the TI bit in the UiC1 register to "0" (data in the UiTB register)
- Apply "L" signal to the $\overline{\text{CTSi}}$ pin if the $\overline{\text{CTS}}$ function is selected

27.10.1.3 Reception

Activating the transmitter in clock synchronous serial I/O mode generates the shift clock. Therefore, set for transmission even if the microcomputer is used for reception only. Dummy data is output from the TxDi pin while receiving.

If an internal clock is selected, the shift clock is generated when the TE bit in the UiC1 registers is set to "1" (receive enabled) and dummy data is set in the UiTB register. If an external clock is selected, the shift clock is generated when the external clock is input into CLKi pin while the TE bit is set to "1" (receive enabled) and dummy data is set in the UiTB register.

When receiving data consecutively while the RE bit in the UiC1 register is set to "1" (data in the UiRB register) and the next data is received by the UARTi reception register, an overrun error occurs and the OER bit in the UiRB register is set to "1" (overrun error). In this case, the UiRB register is indeterminate. When overrun error occurs, program both reception and transmission registers to retransmit earlier data. The IR bit in the SiRIC does not change when an overrun error occurs.

When receiving data consecutively, feed dummy data to the low-order byte in the UiTB register every time a reception is made.

When an external clock is selected while the CKPOL bit in the UiC0 register is set to "0" (data is transmitted on the falling edge of the transfer clock and received on the rising edge) and the external clock is held "H" or when the CKPOL bit is set to "1" (data is transmitted on the rising edge of the transfer clock and received on the falling edge) and the external clock is held "L", meet the following conditions:

- Set the RE bit in the UiC1 register to "1" (receive enabled)
- Set the TE bit in the UiC1 register to "1" (transmit enabled)
- Set the TI bit in the UiC1 register to "0" (data in the UiTB register)

27.10.2 UART Mode

Set the UiERE bit (i=0 to 4) in the UiC1 register after setting the UiMR register.

27.10.3 Special Mode 1 (I^2C Mode)

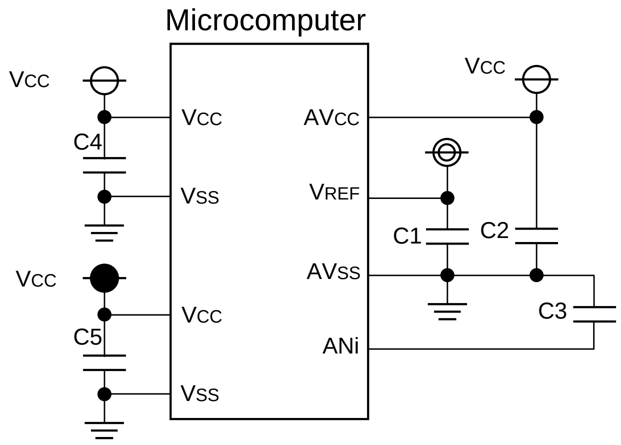
To generate the start condition, stop condition or restart condition, set the STSPSEL bit in the UiSMR4 register to "0" first. Then, change each condition generating bit (the STAREQ bit, STPREQ bit or RSTAREQ bit) setting from "0" to "1" after going through a half cycle of the transfer clock.

27.11 A/D Converter

- Set the AD0CON0 (bit 6 excluded), AD0CON1, AD0CON2, AD0CON3, and AD0CON4 registers while the A/D conversion is stopped (before a trigger is generated).

- Wait a minimum of 1µs before starting the A/D conversion when changing the VCUT bit setting in the AD0CON1 register from "0" (VREF no connection) to "1" (VREF connection).
  Change the VCUT bit setting from "1" to "0" after the A/D conversion is completed.

- Insert capacitors between the AVCC pin, VREF pin, analog input pin ANij (i=none, 0, 2, 15; j=0 to 7) and AVSS pin to prevent latch-ups and malfunctions due to noise, and to minimize conversion errors. The same applies to the VCC and VSS pins. Figure 27.4 shows the use of capacitors to reduce noise.



ANi: ANi, AN0i, AN15i and AN2i (i=0 to 7)

NOTES:
1. C1≥0.47µF, C2≥0.47µF, C3≥100pF, C4≥0.1µF, C5≥0.1µF (reference)
2. Use thick and shortest possible wiring to connect capacitors.

Figure 27.4 Use of Capacitors to Reduce Noise

- Set the bit in the port direction register, which corresponds to the pin being used as the analog input, to "0" (input mode). Set the bit in the port direction register, which corresponds to the $\overline{\text{ADTRG}}$ pin, to "0" (input mode) if the TRG bit in the AD0CON0 register is set to "1" (external trigger).

- When generating a key input interrupt, do not use the AN4 to AN7 pins as analog input pins (key input interrupt request is generated when the A/D input voltage becomes "L").

- The φAD frequency must be 16MHz or less. When the sample and hold function is not activated, the φAD frequency must be 250 kHz or more. If the sample and hold function is activated, the φAD frequency must be 1MHz or more.

- Set the CH2 to CH0 bits in the AD0CON0 register or the SCAN1 and SCAN0 bits in the AD0CON1 register to re-select analog input pins when changing A/D conversion mode.

- $AV_{CC} = V_{REF} = V_{CC}$, A/D input voltage (for AN_0 to AN_7, $AN0_0$ to $AN0_7$, $AN2_0$ to $AN2_7$, $AN15_0$ to $AN15_7$, ANEX0, and ANEX1) $\leq V_{CC}$.

- Wrong values are stored in the AD0i register (i=0 to 7) if the CPU reads the AD0i register while the AD0i register stores results from a completed A/D conversion. This occurs when the CPU clock is set to a divided main clock or a sub clock.
  In one-shot mode or single sweep mode, read the corresponding AD0i register after verifying that the A/D conversion has been completed. The IR bit in the AD0IC register determines the completion of the A/D conversion.
  In repeat mode, repeat sweep mode 0, repeat sweep mode 1, multi-port single sweep mode, and multi-port repeat sweep mode 0, use an undivided main clock as the CPU clock.

- Conversion results of the A/D converter are indeterminate if the ADST bit in the AD0CON0 register is set to "0" (A/D conversion stopped) and the conversion is forcibly terminated by program during the A/D conversion. The AD0i register not performing the A/D conversion may also be indeterminate.
  If the ADST bit is changed to "0" by program, during the A/D conversion, do not use any values obtained from the AD0i registers.

- External triggers cannot be used in DMAC operating mode. Do not read the AD00 register by program.

- Do not perform the A/D conversion in wait mode.

- Set the MCD4 to MCD0 bits in the MCD register to "10010_2" (no division) if using the sample and hold function.

- Do not acknowledge any interrupt requests, even if generated, before setting the ADST bit, if the A/D conversion is terminated by setting the ADST bit in the AD0CON0 register to "0" (A/D conversion stopped) while the microcomputer is A/D converting in single sweep mode.

27.12 Intelligent I/O

27.12.1 Register Setting

Operations, controlled by the values written to the G1BT, G1BCR1, G1TMCR0 to G1TMCR7, G1TPR6, G1TPR7, G1TM0 to G1TM7, G1POCR0 to G1POCR7, G1PO0 to G1PO7, G1FS and G1FE registers, are affected by the count source (f_{BT1}) set in the BCK1 and BCK0 bits in the G1BCR0 register.
Set the BCK1 and BCK0 bits before setting the G1BT, G1BCR1, G1TMCR0 to G1TMCR7, G1TPR6, G1TPR7, G1TM0 to G1TM7, G1POCR0 to G1POCR7, G1PO0 to G1PO7, G1FS and G1FE registers.

Operations, controlled by the values written to the G0RI and G1RI, G0TO and G1TO, G0CR and G1CR, G0RB and G1RB, G0MR and G1MR, G0EMR and G1EMR, G0ETC and G1ETC, G0ERC and G1ERC, G0IRF, G1IRF, G0TB and G1TB, G0CMP0 to G0CMP3, G1CMP0 to G1CMP3, G0MSK0 and G0MSK1, G1MSK0 and G1MSK1, G0TCRC and G1TCRC, G0RCRC and G1RCRC registers are affected by the transfer clock.
Set trasfer clock before setting the G0RI and G1RI, G0TO and G1TO, G0CR and G1CR, G0RB and G1RB, G0MR and G1MR, G0EMR and G1EMR, G0ETC and G1ECT, G0ERC and G1ERC, G0IRF and G1IRF, G0TB and G1TB, G0CMP0 to G0CMP3, G1CMP0 to G1CMP3, G0MSK0 and G0MSK1, G1MSK0 and G1MSK1, G0TCRC and G1TCRC, G0RCRC and G1RCRC registers.

27.13 Programmable I/O Ports

- Because ports $P7_2$ to $P7_5$, $P8_0$, and $P8_1$ have three-phase PWM output forced cutoff function, they are affected by the three-phase motor control timer function and the $\overline{\text{NMI}}$ pin when these ports are set for output functions (port output, timer output, three-phase PWM output, serial I/O output, intelligent I/O output).

  Table 27.4 shows the INVC0 register setting, the $\overline{\text{NMI}}$ pin input level and the state of output ports.

Table 27.4 INVC0 Register and the $\overline{\text{NMI}}$ Pin

Setting Value of the INVC0 Register		Signal level Applied to the NMI Pin	$P7_2$ to $P7_5$, $P8_0$, $P8_1$ Pin States (When Setting Them as Output Pins)
INV02 Bit	INV03 Bit		
0 (Not Using the Three-Phase Motor Control Timer Functions)	-	-	Provides functions selected by the PS1, PSL1, PSC, PS2, PSL2 registers
1 (Using the Three-Phase Motor Control Timer Functions)	0 (Three-Phase Motor Control Timer Output Disabled)	-	High-impedance state
	1 (Three-Phase Motor Control Timer Output Enabled)$^{(1)}$	H	Provides functions selected by the PS1, PSL1, PSC, PS2, PSL2 registers
		L (Forcibly Terminated)	High-impedance state

NOTES :

1. The INV03 bit is set to "0" after a low-level ("L") signal is applied to the $\overline{\text{NMI}}$ pin.

- The availability of pull-up resistors is indeterminate until internal power voltage stabilizes, if the RESET pin is held "L".

- The input threshold voltage varies between programmable I/O ports and peripheral functions. Therefore, if the lelvel of the voltage applied to a pin shared by both programmable I/O ports and peripheral functions is not within the recommended operating condition, V_{IH} and V_{IL} (neither "H" nor "L"), the level may vary depending on the programmable ports and peripheral functions.

27.14 Flash Memory Version

27.14.1 Boot Mode

I/O pins may not be placed in high-impedance states until internal voltage stabilizes, when power is turned on in boot mode. Use the following procedure to turn on power in boot mode.

1) Apply an "L" signal to the $\overline{\text{RESET}}$ and the CNV$_{SS}$ pin

2) Wait a minimum of 2ms after V$_{CC}$ reaches 2.7V or above (until internal voltage stabilizes)

3) Apply an "H" signal to the CNV$_{SS}$ pin

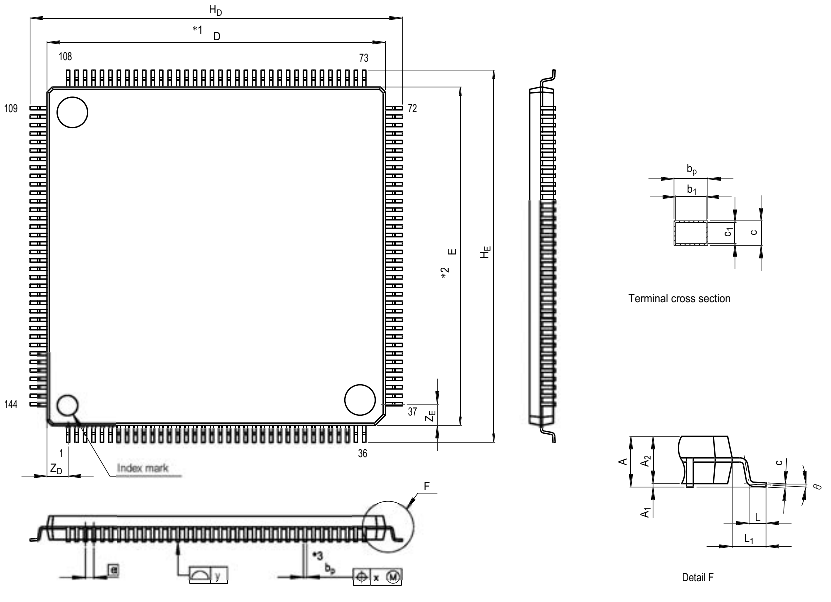
4) Apply an "H" signal to the $\overline{\text{RESET}}$ pin (reset exited)

27.15 Noise

Connect a bypass capacitor (0.1µF or more) between VCC and VSS by shortest path, using thick wires.

Package Dimensions

JEITA Package Code	RENESAS Code	Previous Code	MASS[Typ.]
P-LQFP144-20x20-0.50	PLQP0144KA-A	144P6Q-A / FP-144L / FP-144LV	1.2g



Terminal cross section

Detail F

NOTE)
1. DIMENSIONS "*1" AND "*2" DO NOT INCLUDE MOLD FLASH.
2. DIMENSION "*3" DOES NOT INCLUDE TRIM OFFSET.

Reference Symbol	Dimension in Millimeters		
	Min	Nom	Max
D	19.9	20.0	20.1
E	19.9	20.0	20.1
A_2	——	1.4	——
H_D	21.8	22.0	22.2
H_E	21.8	22.0	22.2
A	——	——	1.7
A_1	0.05	0.1	0.15
b_p	0.17	0.22	0.27
b_1	——	0.20	——
c	0.09	0.145	0.20
c_1	——	0.125	——
θ	0∞	——	8∞
$\boxed{e}$	——	0.5	——
x	——	——	0.08
y	——	——	0.10
Z_D	——	1.25	——
Z_E	——	1.25	——
L	0.35	0.5	0.65
L_1	——	1.0	——

Register Index

REVISION HISTORY	M32C/86 Group(M32C/86, M32C/86T) Hardware Manual

Rev.	Date	Description	
		Page	Summary
1.00	**Sep.08, 05**	–	New Document

RENESAS 16/32-BIT SINGLE-CHIP MICROCOMPUTER
HARDWARE MANUAL
M32C/86 Group (M32C/86, M32C/86T)

Publication Data : Rev.1.00 Sep. 08, 2005

Published by : Sales Strategic Planning Div.
Renesas Technology Corp.

M32C/86 Group (M32C/86, M32C/86T)
Hardware Manual

Renesas Electronics Corporation
1753, Shimonumabe, Nakahara-ku, Kawasaki-shi, Kanagawa 211-8668 Japan