To all our customers

Regarding the change of names mentioned in the document, such as Mitsubishi Electric and Mitsubishi XX, to Renesas Technology Corp.

The semiconductor operations of Hitachi and Mitsubishi Electric were transferred to Renesas Technology Corporation on April 1st 2003. These operations include microcomputer, logic, analog and discrete devices, and memory chips other than DRAMs (flash memory, SRAMs etc.) Accordingly, although Mitsubishi Electric, Mitsubishi Electric Corporation, Mitsubishi Semiconductors, and other Mitsubishi brand names are mentioned in the document, these names have in fact all been changed to Renesas Technology Corp. Thank you for your understanding. Except for our corporate trademark, logo and corporate statement, no changes whatsoever have been made to the contents of the document, and these changes do not constitute any alteration to the contents of the document itself.

Note : Mitsubishi Electric will continue the business operations of high frequency & optical devices and power devices.

Renesas Technology Corp.
Customer Support Dept.
April 1, 2003

MITSUBISHI 16-BIT SINGLE-CHIP MICROCOMPUTER
M16C FAMILY / M16C/60 SERIES

M16C/62

(M16C/62A, M16C/62M)

Group

User's Manual

http://www.infomicom.maec.co.jp/indexe.htm

Before using this material, please visit the above website to confirm that this is the most current document available.

RENESAS
Renesas Technology Corp.

Rev. 1.0
Revision date: June 18, 2001

Keep safety first in your circuit designs!

- Mitsubishi Electric Corporation 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 non-flammable material or (iii) prevention against any malfunction or mishap.

Notes regarding these materials

- These materials are intended as a reference to assist our customers in the selection of the Mitsubishi semiconductor 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 Mitsubishi Electric Corporation or a third party.
- Mitsubishi Electric Corporation 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.
- 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 Mitsubishi Electric Corporation without notice due to product improvements or other reasons. It is therefore recommended that customers contact Mitsubishi Electric Corporation or an authorized Mitsubishi Semiconductor product distributor for the latest product information before purchasing a product listed herein.
  The information described here may contain technical inaccuracies or typographical errors. Mitsubishi Electric Corporation assumes no responsibility for any damage, liability, or other loss rising from these inaccuracies or errors.
  Please also pay attention to information published by Mitsubishi Electric Corporation by various means, including the Mitsubishi Semiconductor home page (http://www.mitsubishichips.com).
- 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. Mitsubishi Electric Corporation assumes no responsibility for any damage, liability or other loss resulting from the information contained herein.
- Mitsubishi Electric Corporation 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 Mitsubishi Electric Corporation or an authorized Mitsubishi Semiconductor 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.
- The prior written approval of Mitsubishi Electric Corporation is necessary to reprint or reproduce in whole or in part these materials.
- 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.
- Please contact Mitsubishi Electric Corporation or an authorized Mitsubishi Semicon ductor product distributor for further details on these materials or the products con tained therein.

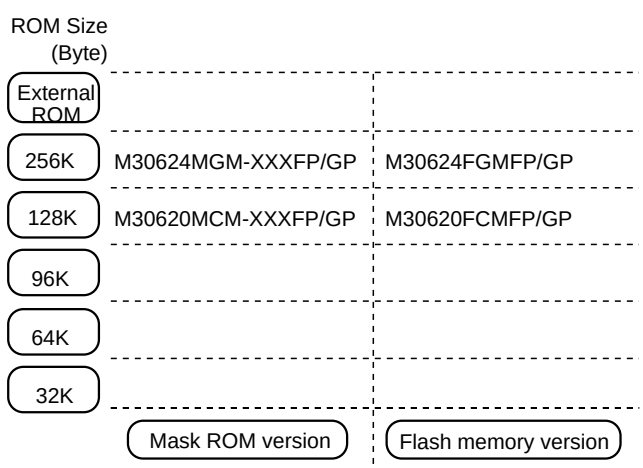
How to Use This Manual

This user's manual is written for the M16C/62A group.
The reader of this manual is expected to have the basic knowledge of electric and logic circuits and microcomputers.
This manual explains a function of the following kind.

- M30620M8A-XXXFP/GP
- M30620MAA-XXXFP/GP
- M30620MCA-XXXFP/GP
- M30622M4A-XXXFP/GP
- M30622M8A-XXXFP/GP
- M30622MAA-XXXFP/GP
- M30622MCA-XXXFP/GP
- M30624MGA-XXXFP/GP
- M30620SAFP/GP
- M30622SAFP/GP
- M30620FCAFP/GP
- M30624FGAFP/GP

- M30620MCM-XXXFP/GP
- M30624MGM-XXXFP/GP
- M30620FCMFP/GP
- M30624FGMFP/GP

These products have similar features except for the memories, which differ from one product to another. This manual gives descriptions of M30622MCA-XXXFP. An electric characteristic refer to data sheet responded to. Memories built-in are as shown below. Be careful when writing a program, as the memories have different capacities.



This manual comprises of five chapters. Use the suggested chapters as a reference for the following topics:

* To understand hardware specifications ..... Chapter 1 Hardware

* To understand the basic way of using peripheral features and the operation timing ..... Chapter 2 Peripheral Functions Usage

* To observe applications of peripheral features ..... Chapter 3 Examples of Peripheral Functions Applications

* To understand how to use external buses ..... Chapter 4 External Buses

* To know the difference between the mask ROM Version and external ROM Version ..... Chapter 5 External ROM Version

This manual includes a quick reference immediately following the Table of Contents and Table of Contents Classified by Function, indicate the page of the topic to be pursued. An index is also described at the end of this manual.

* To find a page describing a peripheral function ..... Table of Contents Classified by Function

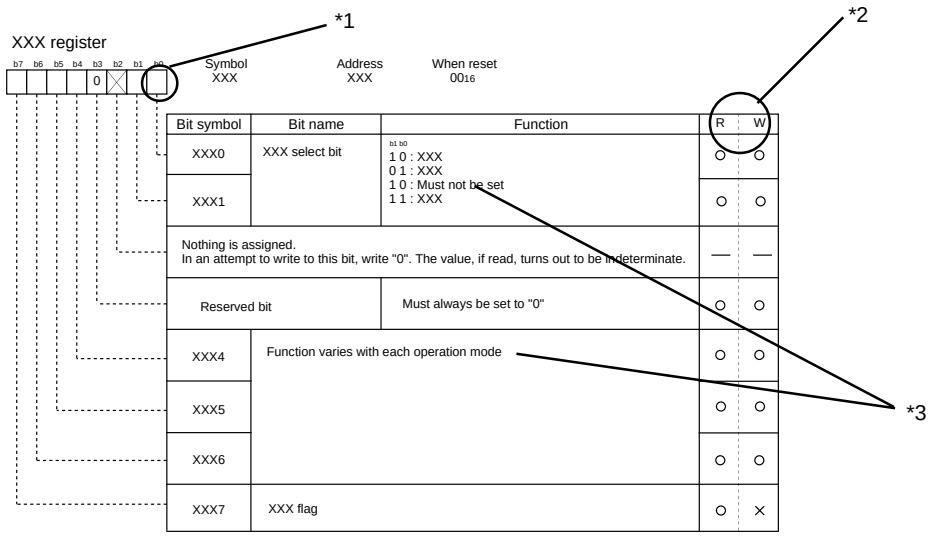
* To find a page describing a specific register by the register address ..... Quick Reference to Pages Classified by Address

Extra application note explains follows, and please refer to each application note in addition to above.

* I^2C BUS ..... M16C/62 Group SIMPLE I^2C BUS

* Three-phase motor control timer function ... M16C/62 Group THREE-PHASE MOTOR CONTROL

The figure of each register configuration describes its functions and attributes as follows :



*1

Blank: Set to "0" or "1" according to intended use

0: Set to "0"

1: Set to "1"

X: Nothing is assigned

*2

R: Read

O.....Possible to read

X.....Impossible to read

–.....Nothing is assigned

W: Write

O.....Possible to write

X.....Written value is invalid

When write, value can be "0" or "1"

–.....Nothing is assigned

*3

Terms to use here are explained as follows.

• Nothing is assigned

Nothing is assigned to the bit concerned. When write, set "0" for new function in future plan.

• Must not be set

Not select. The operation at having selected is not guaranteed.

• Reserved bit

Reserved bit. Set the specified value.

• Function varies with each operation mode

Bit function changes according to the mode of peripheral functions.

• Invalid in A mode

The bit concerned has no function in A mode. Set the specified value.

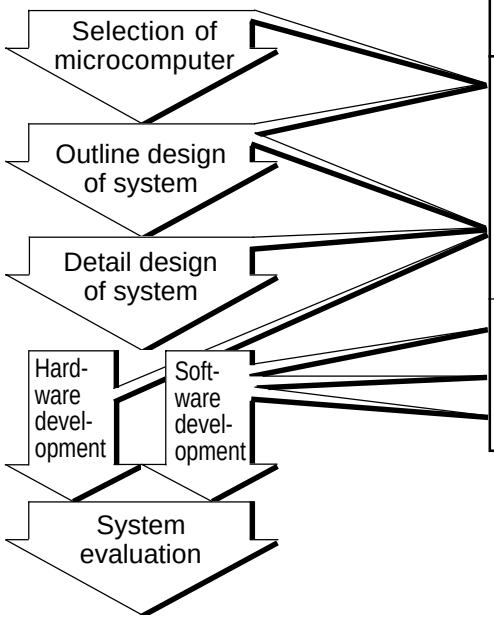
• Valid when bit A="0"

When bit A is "1", the bit concerned has no function. When bit A is "0", the bit concerned has function.

M16C Family-related document list

Usages

(Microcomputer development flow)



Type of document		Contents
Hardware	Data sheet and data book	Hardware specifications (pin assignment, memory map, specifications of peripheral functions, electrical characteristics, timing charts)
	User's manual	Detailed description about hardware specifications, operation, and application examples (connection with peripherals, relationship with software)
Software	Programming manual	Method for creating programs using assembly and C languages
	Software manual	Detailed description about operation of each instruction (assembly language)

M16C Family Line-up

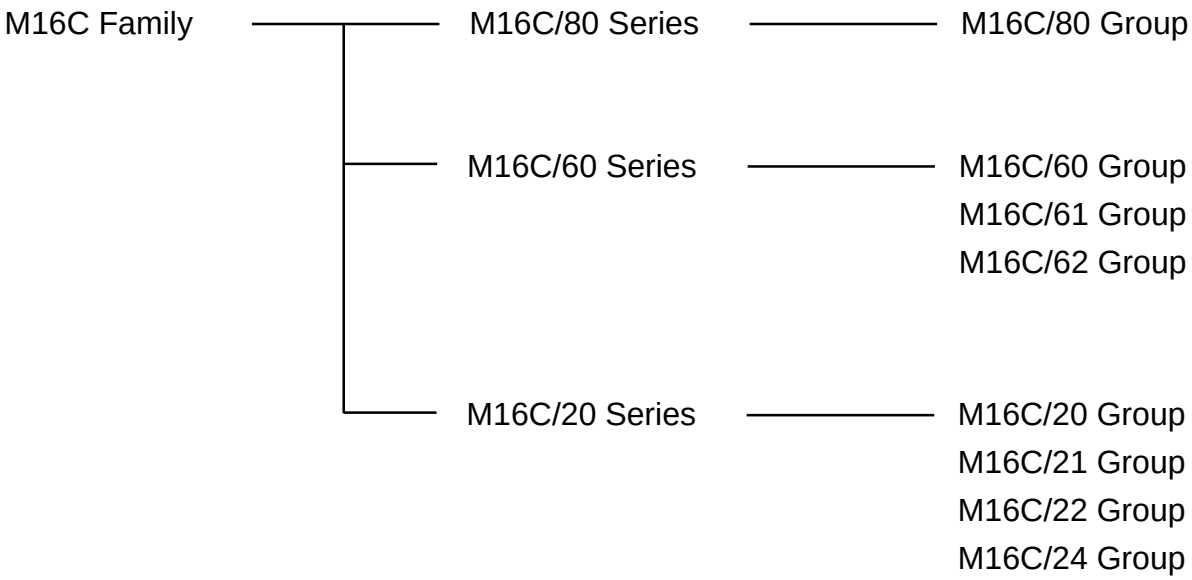


Table of Contents

Chapter 1 Hardware

Description 1-2
Memory 1-11
Central Processing Unit (CPU) 1-12
Reset 1-15
Processor Mode 1-22
Bus Settings 1-26
Bus Control 1-28
Clock Generating Circuit 1-36
Protection 1-45
Overview of Interrupt 1-46
Watchdog Timer 1-66
DMAC 1-68
Timer 1-78
Timer A 1-80
Timer B 1-90
Timers' functions for three-phase motor control 1-96
Serial I/O 1-108
A-D Converter 1-149
D-A Converter 1-159
CRC Calculation Circuit 1-161
Programmable I/O Ports 1-163
Flash Memory Version 1-217

Chapter 2 Peripheral Functions Usage

2.1 Protect Usage 2-2
2.1.1 Overview of the protect usage 2-2
2.1.2 Protect Operation 2-4
2.1.3 Precaution for Protect 2-5
2.2 Timer A Usage 2-6
2.2.1 Overview of the timer A usage 2-6
2.2.2 Operation of Timer A (timer mode) 2-12

2.2.3 Operation of Timer A (timer mode, gate function selected) 2-14
2.2.4 Operation of Timer A (timer mode, pulse output function selected) 2-16
2.2.5 Operation of Timer A (event counter mode, reload type selected) 2-18
2.2.6 Operation of Timer A (event counter mode, free run type selected) 2-20
2.2.7 Operation of timer A (2-phase pulse signal process in event counter mode, normal mode selected) 2-22
2.2.8 Operation of timer A (2-phase pulse signal process in event counter mode, multiply-by-4 mode selected) 2-24
2.2.9 Operation of Timer A (one-shot timer mode) 2-26
2.2.10 Operation of Timer A (one-shot timer mode, external trigger selected) 2-28
2.2.11 Operation of Timer A (pulse width modulation mode, 16-bit PWM mode selected) 2-30
2.2.12 Operation of Timer A (pulse width modulation mode, 8-bit PWM mode selected) 2-32
2.2.13 Precautions for Timer A (timer mode) 2-34
2.2.14 Precautions for Timer A (event counter mode) 2-35
2.2.15 Precautions for Timer A (one-shot timer mode) 2-36
2.2.16 Precautions for Timer A (pulse width modulation mode) 2-37
2.3 Timer B Usage 2-38
2.3.1 Overview of the timer B usage 2-38
2.3.2 Operation of Timer B (timer mode) 2-42
2.3.3 Operation of Timer B (event counter mode) 2-44
2.3.4 Operation of Timer B (pulse period measurement mode) 2-46
2.3.5 Operation of Timer B (pulse width measurement mode) 2-48
2.3.6 Precautions for Timer B (timer mode, event counter mode) 2-50
2.3.7 Precautions for Timer B (pulse period/pulse width measurement mode) 2-51
2.4 Clock-Synchronous Serial I/O Usage 2-52
2.4.1 Overview of the clock-synchronous serial I/O usage 2-52
2.4.2 Operation of Serial I/O (transmission in clock-synchronous serial I/O mode) 2-60
2.4.3 Operation of the Serial I/O (transmission in clock-synchronous serial I/O mode, transfer clock output from multiple pins function selected) 2-64
2.4.4 Operation of Serial I/O (reception in clock-synchronous serial I/O mode) 2-68
2.4.5 Precautions for Serial I/O (in clock-synchronous serial I/O) 2-72
2.5 Clock-Asynchronous Serial I/O (UART) Usage 2-74
2.5.1 Overview of the clock-asynchronous serial I/O usage 2-74
2.5.2 Operation of Serial I/O (transmission in UART mode) 2-84
2.5.3 Operation of Serial I/O (reception in UART mode) 2-88

2.5.4 Operation of Serial I/O (transmission used for SIM interface) 2-92
2.5.5 Operation of Serial I/O (reception used for SIM interface) 2-96
2.5.6 Clock Signals in used for the SIM Interface 2-100
2.5.7 Error-permitted range range of transfer baud 2-104
2.6 SI/O3, 4 Usage 2-106
2.6.1 Overview of the SI/O3,4 usage 2-106
2.6.2 Operation of SI/O3,4 2-108
2.7 A-D Converter Usage 2-110
2.7.1 Overview of the A-D converter usage 2-110
2.7.2 Operation of A-D converter (one-shot mode) 2-116
2.7.3 Operation of A-D Converter (in one-shot mode, an external trigger selected) 2-118
2.7.4 Operation of A-D Converter (in one-shot mode, expanded analog input pin selected) ... 2-120
2.7.5 Operation of A-D Converter (in one-shot mode, external op-amp connection mode selected) .
..... 2-122
2.7.6 Operation of A-D Converter (in repeat mode) 2-124
2.7.7 Operation of A-D Converter (in single sweep mode) 2-126
2.7.8 Operation of A-D Converter (in repeat sweep mode 0) 2-128
2.7.9 Operation of A-D Converter (in repeat sweep mode 1) 2-130
2.7.10 Precautions for A-D Converter 2-132
2.7.11 Method of A-D Conversion (10-bit mode) 2-133
2.7.12 Method of A-D Conversion (8-bit mode) 2-135
2.7.13 Absolute Accuracy and Differential Non-Linearity Error 2-137
2.7.14 Internal Equivalent Circuit of Analog Input 2-139
2.7.15 Sensor’s Output Impedance under A-D Conversion 2-140
2.8 D-A Converter Usage 2-142
2.8.1 Overview of the D-A converter usage 2-142
2.8.2 D-A Converter Operation 2-143
2.9 DMAC Usage 2-144
2.9.1 Overview of the DMAC usage 2-144
2.9.2 Operation of DMAC (one-shot transfer mode) 2-148
2.9.3 Operation of DMAC (repeated transfer mode) 2-150
2.10 CRC Calculation Circuit Usage 2-152
2.10.1 Overview of the CRC calculation circuit usage 2-152
2.10.2 Operation of CRC Calculation Circuit 2-153
2.11 Watchdog Timer Usage 2-154

2.11.1 Overview of the watchdog timer usage 2-154
2.11.2 Operation of Watchdog Timer 2-156
2.12 Address Match Interrupt Usage 2-158
2.12.1 Overview of the address match interrupt usage 2-158
2.12.2 Operation of Address Match Interrupt 2-160
2.13 Key-Input Interrupt Usage 2-162
2.13.1 Overview of the key-input interrupt usage 2-162
2.13.2 Operation of Key-Input Interrupt 2-164
2.14 Multiple interrupts usage 2-166
2.14.1 Overview of the Multiple interrupts usage 2-166
2.14.2 Multiple Interrupts operation 2-171
2.15 Power Control Usage 2-173
2.15.1 Overview of the power control usage 2-173
2.15.2 Stop Mode Set-Up 2-178
2.15.3 Wait Mode Set-Up 2-179
2.15.4 Precautions in Power Control 2-180
2.16 Programmable I/O Ports Usage 2-181
2.16.1 Overview of the programmable I/O ports usage 2-181

Chapter 3 Examples of Peripheral functions Applications

3.1 Long-Period Timers 2-192
3.2 Variable-Period Variable-Duty PWM Output 2-196
3.3 Delayed One-Shot Output 2-200
3.4 Buzzer Output 2-204
3.5 Solution for External Interrupt Pins Shortage 2-206
3.6 Memory to Memory DMA Transfer 2-208
3.7 Controlling Power Using Stop Mode 2-212
3.8 Controling Power Using Wait Mode 2-216

Chapter 4 External Buses

4.1 Overview of External Buses 2-222
4.2 Data Access 2-223
4.2.1 Data Bus Width 2-223
4.2.2 Chip Selects and Address Bus 2-224
4.2.3 Bus Types 2-225
4.2.4 R/W Modes 2-225

4.3 Connection Examples 2-226
4.3.1 16-bit Memory to 16-bit Width Data Bus Connection Example 2-226
4.3.2 8-bit Memory to 16-bit Width Data Bus Connection Example 2-227
4.3.3 8-bit Memory to 8-bit Width Data Bus Connection Example 2-229
4.3.4 Two 8-bit and 16-Bit Memory to 16-Bit Width Data Bus Connection Example 2-230
4.3.5 Chip Selects and Address Bus 2-231
4.4 Connectable Memories 2-232
4.4.1 Operation Frequency and Access Time 2-232
4.4.2 Connecting Low-Speed Memory 2-236
4.4.3 Connectable Memories 2-240
4.5 Releasing an External Bus ($\overline{\text{HOLD}}$ input and $\overline{\text{HLDA}}$ output) 2-241
4.6 Precautions for External Bus 2-242

Chapter 5 External ROM Version

5.1 Pin Configuration 2-245
5.2 Pin Description 2-247
5.3 Memory Map 2-249
5.4 Processor Mode 2-253

Table of Contents Classified by Function

Protection 1-45

2.1.1 Overview of the protect usage 2-2

2.1.2 Protect Operation 2-4

2.1.3 Precaution for Protect 2-5

Timer A 1-80

2.2.1 Overview of the timer A usage 2-6

2.2.2 Operation of Timer A (timer mode) 2-12

2.2.3 Operation of Timer A (timer mode, gate function selected) 2-14

2.2.4 Operation of Timer A (timer mode, pulse output function selected) 2-16

2.2.5 Operation of Timer A (event counter mode, reload type selected) 2-18

2.2.6 Operation of Timer A (event counter mode, free run type selected) 2-20

2.2.7 Operation of timer A (2-phase pulse signal process in event counter mode, normal mode selected) 2-22

2.2.8 Operation of timer A (2-phase pulse signal process in event counter mode, multiply-by-4 mode selected) 2-24

2.2.9 Operation of Timer A (one-shot timer mode) 2-26

2.2.10 Operation of Timer A (one-shot timer mode, external trigger selected) 2-28

2.2.11 Operation of Timer A (pulse width modulation mode, 16-bit PWM mode selected) 2-30

2.2.12 Operation of Timer A (pulse width modulation mode, 8-bit PWM mode selected) 2-32

2.2.13 Precautions for Timer A (timer mode) 2-34

2.2.14 Precautions for Timer A (event counter mode) 2-35

2.2.15 Precautions for Timer A (one-shot timer mode) 2-36

2.2.16 Precautions for Timer A (pulse width modulation mode) 2-37

Timer B 1-90

2.3.1 Overview of the timer B usage 2-38

2.3.2 Operation of Timer B (timer mode) 2-42

2.3.3 Operation of Timer B (event counter mode) 2-44

2.3.4 Operation of Timer B (pulse period measurement mode) 2-46

2.3.5 Operation of Timer B (pulse width measurement mode) 2-48

2.3.6 Precautions for Timer B (timer mode, event counter mode) 2-50

2.3.7 Precautions for Timer B (pulse period/pulse width measurement mode) 2-51

Serial I/O 1-108
2.4.1 Overview of the clock-synchronous serial I/O usage 2-52
2.4.2 Operation of Serial I/O (transmission in clock-synchronous serial I/O mode) 2-60
2.4.3 Operation of the Serial I/O (transmission in clock-synchronous serial I/O mode, transfer clock output from multiple pins function selected) 2-64
2.4.4 Operation of Serial I/O (reception in clock-synchronous serial I/O mode) 2-68
2.4.5 Precautions for Serial I/O (in clock-synchronous serial I/O) 2-72
2.5.1 Overview of the clock-asynchronous serial I/O usage 2-74
2.5.2 Operation of Serial I/O (transmission in UART mode) 2-84
2.5.3 Operation of Serial I/O (reception in UART mode) 2-88
2.5.4 Operation of Serial I/O (transmission used for SIM interface) 2-92
2.5.5 Operation of Serial I/O (reception used for SIM interface) 2-96
2.5.6 Clock Signals in used for the SIM Interface 2-100
2.5.7 Error-permitted range range of transfer baud 2-104
2.6.1 Overview of the SI/O3,4 usage 2-106
2.6.2 Operation of SI/O3,4 2-108
A-D Converter 1-149
2.7.1 Overview of the A-D converter usage 2-110
2.7.2 Operation of A-D converter (one-shot mode) 2-116
2.7.3 Operation of A-D Converter (in one-shot mode, an external trigger selected) 2-118
2.7.4 Operation of A-D Converter (in one-shot mode, expanded analog input pin selected) ... 2-120
2.7.5 Operation of A-D Converter (in one-shot mode, external op-amp connection mode selected) 2-122
2.7.6 Operation of A-D Converter (in repeat mode) 2-124
2.7.7 Operation of A-D Converter (in single sweep mode) 2-126
2.7.8 Operation of A-D Converter (in repeat sweep mode 0) 2-128
2.7.9 Operation of A-D Converter (in repeat sweep mode 1) 2-130
2.7.10 Precautions for A-D Converter 2-132
2.7.11 Method of A-D Conversion (10-bit mode) 2-133
2.7.12 Method of A-D Conversion (8-bit mode) 2-135
2.7.13 Absolute Accuracy and Differential Non-Linearity Error 2-137
2.7.14 Internal Equivalent Circuit of Analog Input 2-139
2.7.15 Sensor’s Output Impedance under A-D Conversion 2-140
D-A Converter 1-159
2.8.1 Overview of the D-A converter usage 2-142

2.8.2 D-A Converter Operation 2-143
DMAC 1-68
2.9.1 Overview of the DMAC usage 2-144
2.9.2 Operation of DMAC (one-shot transfer mode) 2-148
2.9.3 Operation of DMAC (repeated transfer mode) 2-150
CRC Calculation Circuit 1-161
2.10.1 Overview of the CRC calculation circuit usage 2-152
2.10.2 Operation of CRC Calculation Circuit 2-153
Watchdog Timer 1-66
2.11.1 Overview of the watchdog timer usage 2-154
2.11.2 Operation of Watchdog Timer 2-156
Overview of Interrupt 1-46
2.12.1 Overview of the address match interrupt usage 2-158
2.12.2 Operation of Address Match Interrupt 2-160
2.13.1 Overview of the key-input interrupt usage 2-162
2.13.2 Operation of Key-Input Interrupt 2-164
2.14.1 Overview of the Multiple interrupts usage 2-166
2.14.2 Multiple Interrupts operation 2-171
Power Control 1-43
2.15.1 Overview of the power control usage 2-173
2.15.2 Stop Mode Set-Up 2-178
2.15.3 Wait Mode Set-Up 2-179
2.15.4 Precautions in Power Control 2-180
Programmable I/O Ports 1-163
2.16.1 Overview of the programmable I/O ports usage 2-181

Quick Reference to Pages Classified by Address

Address	Register	Page
0000₁₆		
0001₁₆		
0002₁₆		
0003₁₆		
0004₁₆	Processor mode register 0 (PM0)	1-23
0005₁₆	Processor mode register 1 (PM1)	
0006₁₆	System clock control register 0 (CM0)	1-39
0007₁₆	System clock control register 1 (CM1)	
0008₁₆	Chip select control register (CSR)	1-29
0009₁₆	Address match interrupt enable register (AIER)	1-63
000A₁₆	Protect register (PRCR)	1-38
000B₁₆		
000C₁₆		
000D₁₆		
000E₁₆	Watchdog timer start register (WDTS)	1-67
000F₁₆	Watchdog timer control register (WDC)	
0010₁₆ 0011₁₆ 0012₁₆	Address match interrupt register 0 (RMAD0)	1-63
0013₁₆		
0014₁₆ 0015₁₆ 0016₁₆	Address match interrupt register 1 (RMAD1)	1-63
0017₁₆		
0018₁₆		
0019₁₆		
001A₁₆		
001B₁₆		
001C₁₆		
001D₁₆		
001E₁₆		
001F₁₆		
0020₁₆ 0021₁₆ 0022₁₆	DMA0 source pointer (SAR0)	1-72
0023₁₆		
0024₁₆ 0025₁₆ 0026₁₆	DMA0 destination pointer (DAR0)	1-72
0027₁₆		
0028₁₆ 0029₁₆	DMA0 transfer counter (TCR0)	1-72
002A₁₆		
002B₁₆		
002C₁₆	DMA0 control register (DM0CON)	1-71
002D₁₆		
002E₁₆		
002F₁₆		
0030₁₆ 0031₁₆ 0032₁₆	DMA1 source pointer (SAR1)	1-72
0033₁₆		
0034₁₆ 0035₁₆ 0036₁₆	DMA1 destination pointer (DAR1)	1-72
0037₁₆		
0038₁₆ 0039₁₆	DMA1 transfer counter (TCR1)	1-72
003A₁₆		
003B₁₆		
003C₁₆	DMA1 control register (DM1CON)	1-71
003D₁₆		
003E₁₆		
003F₁₆		

Address	Register	Page
0040₁₆		
0041₁₆		
0042₁₆		
0043₁₆		
0044₁₆	INT3 interrupt control register (INT3IC)	1-52
0045₁₆	Timer B5 interrupt control register (TB5IC)	
0046₁₆	Timer B4 interrupt control register (TB4IC)	
0047₁₆	Timer B3 interrupt control register (TB3IC)	
0048₁₆	SI/O4 interrupt control register (S4IC) INT5 interrupt control register (INT5IC)	
0049₁₆	SI/O3 interrupt control register (S3IC) INT4 interrupt control register (INT4IC)	
004A₁₆	Bus collision detection interrupt control register (BCNIC)	
004B₁₆	DMA0 interrupt control register (DM0IC)	
004C₁₆	DMA1 interrupt control register (DM1IC)	
004D₁₆	Key input interrupt control register (KUPIC)	
004E₁₆	A-D conversion interrupt control register (ADIC)	
004F₁₆	UART2 transmit interrupt control register (S2TIC)	
0050₁₆	UART2 receive interrupt control register (S2RIC)	
0051₁₆	UART0 transmit interrupt control register (S0TIC)	
0052₁₆	UART0 receive interrupt control register (S0RIC)	
0053₁₆	UART1 transmit interrupt control register (S1TIC)	
0054₁₆	UART1 receive interrupt control register (S1RIC)	
0055₁₆	Timer A0 interrupt control register (TA0IC)	
0056₁₆	Timer A1 interrupt control register (TA1IC)	
0057₁₆	Timer A2 interrupt control register (TA2IC)	
0058₁₆	Timer A3 interrupt control register (TA3IC)	
0059₁₆	Timer A4 interrupt control register (TA4IC)	
005A₁₆	Timer B0 interrupt control register (TB0IC)	
005B₁₆	Timer B1 interrupt control register (TB1IC)	
005C₁₆	Timer B2 interrupt control register (TB2IC)	
005D₁₆	INT0 interrupt control register (INT0IC)	
005E₁₆	INT1 interrupt control register (INT1IC)	
005F₁₆	INT2 interrupt control register (INT2IC)	
0060₁₆		
0061₁₆		
0062₁₆		
0063₁₆		
0064₁₆		
0065₁₆		
≈	≈	
032A₁₆		
032B₁₆		
032C₁₆		
032D₁₆		
032E₁₆		
032F₁₆		
0330₁₆		
0331₁₆		
0332₁₆		
0333₁₆		
0334₁₆		
0335₁₆		
0336₁₆		
0337₁₆		
0338₁₆		
0339₁₆		
033A₁₆		
033B₁₆		
033C₁₆		
033D₁₆		
033E₁₆		
033F₁₆		

Note : Locations in the SFR area where nothing is allocated are reserved areas. Do not access these areas for read or write.

Quick Reference to Pages Classified by Address

Address	Register	Page
0340₁₆	Timer B3, 4, 5 count start flag (TBSR)	1-91
0341₁₆		
0342₁₆	Timer A1-1 register (TA11)	1-98
0343₁₆		
0344₁₆	Timer A2-1 register (TA21)	
0345₁₆		
0346₁₆	Timer A4-1 register (TA41)	
0347₁₆		
0348₁₆	Three-phase PWM control register 0(INVC0)	1-96
0349₁₆	Three-phase PWM control register 1(INVC1)	
034A₁₆	Three-phase output buffer register 0(IDB0)	1-97
034B₁₆	Three-phase output buffer register 1(IDB1)	
034C₁₆	Dead time timer(DTT)	
034D₁₆	Timer B2 interrupt occurrence frequency set counter(ICTB2)	
034E₁₆		
034F₁₆		
0350₁₆	Timer B3 register (TB3)	1-91
0351₁₆		
0352₁₆	Timer B4 register (TB4)	
0353₁₆		
0354₁₆	Timer B5 register (TB5)	
0355₁₆		
0356₁₆		
0357₁₆		
0358₁₆		
0359₁₆		
035A₁₆		
035B₁₆	Timer B3 mode register (TB3MR)	1-90
035C₁₆	Timer B4 mode register (TB4MR)	
035D₁₆	Timer B5 mode register (TB5MR)	
035E₁₆		
035F₁₆	Interrupt request cause select register (IFSR)	1-61
0360₁₆	SI/O3 transmit/receive register (S3TRR)	1-146
0361₁₆		
0362₁₆	SI/O3 control register (S3C)	
0363₁₆	SI/O3 bit rate generator (S3BRG)	
0364₁₆	SI/O4 transmit/receive register (S4TRR)	
0365₁₆		
0366₁₆	SI/O4 control register (S4C)	
0367₁₆	SI/O4 bit rate generator (S4BRG)	
0368₁₆		
0369₁₆		
036A₁₆		
036B₁₆		
036C₁₆		
036D₁₆		
036E₁₆		
036F₁₆		
0370₁₆		
0371₁₆		
0372₁₆		
0373₁₆		
0374₁₆		
0375₁₆	UART2 special mode register 3 (U2SMR3)	1-137
0376₁₆	UART2 special mode register 2 (U2SMR2)	1-141
0377₁₆	UART2 special mode register (U2SMR)	1-137
0378₁₆	UART2 transmit/receive mode register (U2MR)	1-113
0379₁₆	UART2 bit rate generator (U2BRG)	1-112
037A₁₆	UART2 transmit buffer register (U2TB)	
037B₁₆		
037C₁₆	UART2 transmit/receive control register 0 (U2C0)	1-114
037D₁₆	UART2 transmit/receive control register 1 (U2C1)	1-115
037E₁₆	UART2 receive buffer register (U2RB)	1-112
037F₁₆		

Address	Register	Page
0380₁₆	Count start flag (TABSR)	1-81
0381₁₆	Clock prescaler reset flag (CPSRF)	1-82
0382₁₆	One-shot start flag (ONSF)	
0383₁₆	Trigger select register (TRGSR)	
0384₁₆	Up-down flag (UDF)	1-81
0385₁₆		
0386₁₆	Timer A0 register (TA0)	1-81
0387₁₆		
0388₁₆	Timer A1 register (TA1)	
0389₁₆		
038A₁₆	Timer A2 register (TA2)	
038B₁₆		
038C₁₆	Timer A3 register (TA3)	
038D₁₆		
038E₁₆	Timer A4 register (TA4)	
038F₁₆		
0390₁₆	Timer B0 register (TB0)	1-91
0391₁₆		
0392₁₆	Timer B1 register (TB1)	
0393₁₆		
0394₁₆	Timer B2 register (TB2)	
0395₁₆		
0396₁₆	Timer A0 mode register (TA0MR)	1-80
0397₁₆	Timer A1 mode register (TA1MR)	
0398₁₆	Timer A2 mode register (TA2MR)	
0399₁₆	Timer A3 mode register (TA3MR)	
039A₁₆	Timer A4 mode register (TA4MR)	
039B₁₆	Timer B0 mode register (TB0MR)	1-90
039C₁₆	Timer B1 mode register (TB1MR)	
039D₁₆	Timer B2 mode register (TB2MR)	
039E₁₆		
039F₁₆		
03A0₁₆	UART0 transmit/receive mode register (U0MR)	1-113
03A1₁₆	UART0 bit rate generator (U0BRG)	1-112
03A2₁₆	UART0 transmit buffer register (U0TB)	
03A3₁₆		
03A4₁₆	UART0 transmit/receive control register 0 (U0C0)	1-114
03A5₁₆	UART0 transmit/receive control register 1 (U0C1)	1-115
03A6₁₆	UART0 receive buffer register (U0RB)	1-112
03A7₁₆		
03A8₁₆	UART1 transmit/receive mode register (U1MR)	1-113
03A9₁₆	UART1 bit rate generator (U1BRG)	1-112
03AA₁₆	UART1 transmit buffer register (U1TB)	
03AB₁₆		
03AC₁₆	UART1 transmit/receive control register 0 (U1C0)	1-114
03AD₁₆	UART1 transmit/receive control register 1 (U1C1)	1-115
03AE₁₆	UART1 receive buffer register (U1RB)	1-112
03AF₁₆		
03B0₁₆	UART transmit/receive control register 2 (UCON)	1-116
03B1₁₆		
03B2₁₆		
03B3₁₆		
03B4₁₆		
03B5₁₆		
03B6₁₆	Flash memory control register 1 (FMR1) (Note 1)	1-221
03B7₁₆	Flash memory control register 0 (FMR0) (Note 1)	
03B8₁₆	DMA0 request cause select register (DM0SL)	1-70
03B9₁₆		
03BA₁₆	DMA1 request cause select register (DM1SL)	1-71
03BB₁₆		
03BC₁₆	CRC data register (CRCD)	1-161
03BD₁₆		
03BE₁₆	CRC input register (CRCIN)	
03BF₁₆		

Note 1 : This register is only exist in flash memory version.
Note 2 : Locations in the SFR area where nothing is allocated are reserved areas. Do not access these areas for read or write.

Quick Reference to Pages Classified by Address

Address	Register	Page
03C0₁₆ 03C1₁₆	A-D register 0 (AD0)	1-152
03C2₁₆ 03C3₁₆	A-D register 1 (AD1)	
03C4₁₆ 03C5₁₆	A-D register 2 (AD2)	
03C6₁₆ 03C7₁₆	A-D register 3 (AD3)	
03C8₁₆ 03C9₁₆	A-D register 4 (AD4)	
03CA₁₆ 03CB₁₆	A-D register 5 (AD5)	
03CC₁₆ 03CD₁₆	A-D register 6 (AD6)	
03CE₁₆ 03CF₁₆	A-D register 7 (AD7)	
03D0₁₆		
03D1₁₆		
03D2₁₆		
03D3₁₆		
03D4₁₆	A-D control register 2 (ADCON2)	1-152
03D5₁₆		
03D6₁₆	A-D control register 0 (ADCON0)	1-151
03D7₁₆	A-D control register 1 (ADCON1)	
03D8₁₆	D-A register 0 (DA0)	1-160
03D9₁₆		
03DA₁₆	D-A register 1 (DA1)	
03DB₁₆		
03DC₁₆	D-A control register (DACON)	
03DD₁₆		
03DE₁₆		
03DF₁₆		
03E0₁₆	Port P0 register (P0)	1-169
03E1₁₆	Port P1 register (P1)	
03E2₁₆	Port P0 direction register (PD0)	1-168
03E3₁₆	Port P1 direction register (PD1)	
03E4₁₆	Port P2 register (P2)	1-169
03E5₁₆	Port P3 register (P3)	
03E6₁₆	Port P2 direction register (PD2)	1-168
03E7₁₆	Port P3 direction register (PD3)	
03E8₁₆	Port P4 register (P4)	1-169
03E9₁₆	Port P5 register (P5)	
03EA₁₆	Port P4 direction register (PD4)	1-168
03EB₁₆	Port P5 direction register (PD5)	
03EC₁₆	Port P6 register (P6)	1-169
03ED₁₆	Port P7 register (P7)	
03EE₁₆	Port P6 direction register (PD6)	1-168
03EF₁₆	Port P7 direction register (PD7)	
03F0₁₆	Port P8 register (P8)	1-169
03F1₁₆	Port P9 register (P9)	
03F2₁₆	Port P8 direction register (PD8)	1-168
03F3₁₆	Port P9 direction register (PD9)	
03F4₁₆	Port P10 register (P10)	1-169
03F5₁₆		
03F6₁₆	Port P10 direction register (PD10)	1-168
03F7₁₆		
03F8₁₆		
03F9₁₆		
03FA₁₆		
03FB₁₆		
03FC₁₆	Pull-up control register 0 (PUR0)	1-170
03FD₁₆	Pull-up control register 1 (PUR1)	
03FE₁₆	Pull-up control register 2 (PUR2)	
03FF₁₆	Port control register (PCR)	1-171

Note : Locations in the SFR area where nothing is allocated are reserved areas. Do not access these areas for read or write.

Chapter 1

Hardware

Description

The M16C/62A group of single-chip microcomputers are built using the high-performance silicon gate CMOS process using a M16C/60 Series CPU core and are packaged in a 100-pin plastic molded QFP. These single-chip microcomputers operate using sophisticated instructions featuring a high level of instruction efficiency. With 1M bytes of address space, they are capable of executing instructions at high speed. They also feature a built-in multiplier and DMAC, making them ideal for controlling office, communications, industrial equipment, and other high-speed processing applications.

The M16C/62A group includes a wide range of products with different internal memory types and sizes and various package types.

Features

- Memory capacity ..... ROM (See Figure 1.1.4. ROM Expansion)
  RAM 3K to 20K bytes
- Shortest instruction execution time ..... 62.5ns (f(XIN)=16MHz, VCC=5V)
  100ns (f(XIN)=10MHz, VCC=3V, with software one-wait) : Mask ROM, flash memory 5V version
- Supply voltage ..... 4.2V to 5.5V (f(XIN)=16MHz, without software wait) : Mask ROM, flash memory 5V version
  2.7V to 5.5V (f(XIN)=10MHz with software one-wait) : Mask ROM, flash memory 5V version
- Low power consumption ..... 25.5mW (f(XIN)=10MHz, with software one-wait, VCC = 3V)
- Interrupts ..... 25 internal and 8 external interrupt sources, 4 software interrupt sources; 7 levels (including key input interrupt)
- Multifunction 16-bit timer ..... 5 output timers + 6 input timers
- Serial I/O ..... 5 channels (3 for UART or clock synchronous, 2 for clock synchronous)
- DMAC ..... 2 channels (trigger: 24 sources)
- A-D converter ..... 10 bits X 8 channels (Expandable up to 10 channels)
- D-A converter ..... 8 bits X 2 channels
- CRC calculation circuit ..... 1 circuit
- Watchdog timer ..... 1 line
- Programmable I/O ..... 87 lines
- Input port ..... 1 line (P85 shared with $\overline{\text{NMI}}$ pin)
- Memory expansion ..... Available (to a maximum of 1M bytes)
- Chip select output ..... 4 lines
- Clock generating circuit ..... 2 built-in clock generation circuits
  (built-in feedback resistor, and external ceramic or quartz oscillator)

Applications

Audio, cameras, office equipment, communications equipment, portable equipment

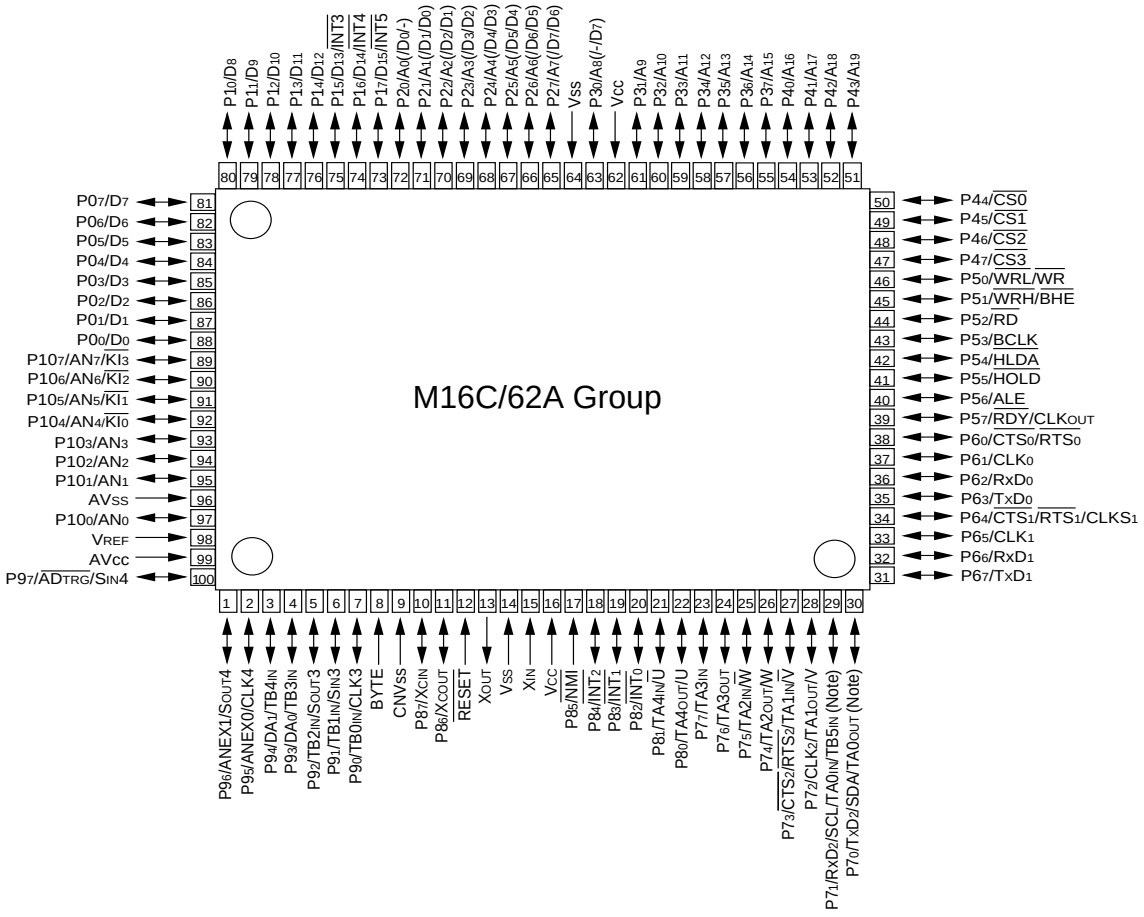
------Table of Contents------

Central Processing Unit (CPU) 1-12
Reset 1-15
Processor Mode 1-22
Clock Generating Circuit 1-36
Protection 1-46
Interrupts 1-46
Watchdog Timer 1-66
DMAC 1-68
Timer 1-78
Serial I/O 1-108
A-D Converter 1-149
D-A Converter 1-159
CRC Calculation Circuit 1-161
Programmable I/O Ports 1-163
Electrical characteristic 1-174
Flash memory version 1-217

Pin Configuration

Figures 1.1.1 and 1.1.2 show the pin configurations (top view).

PIN CONFIGURATION (top view)

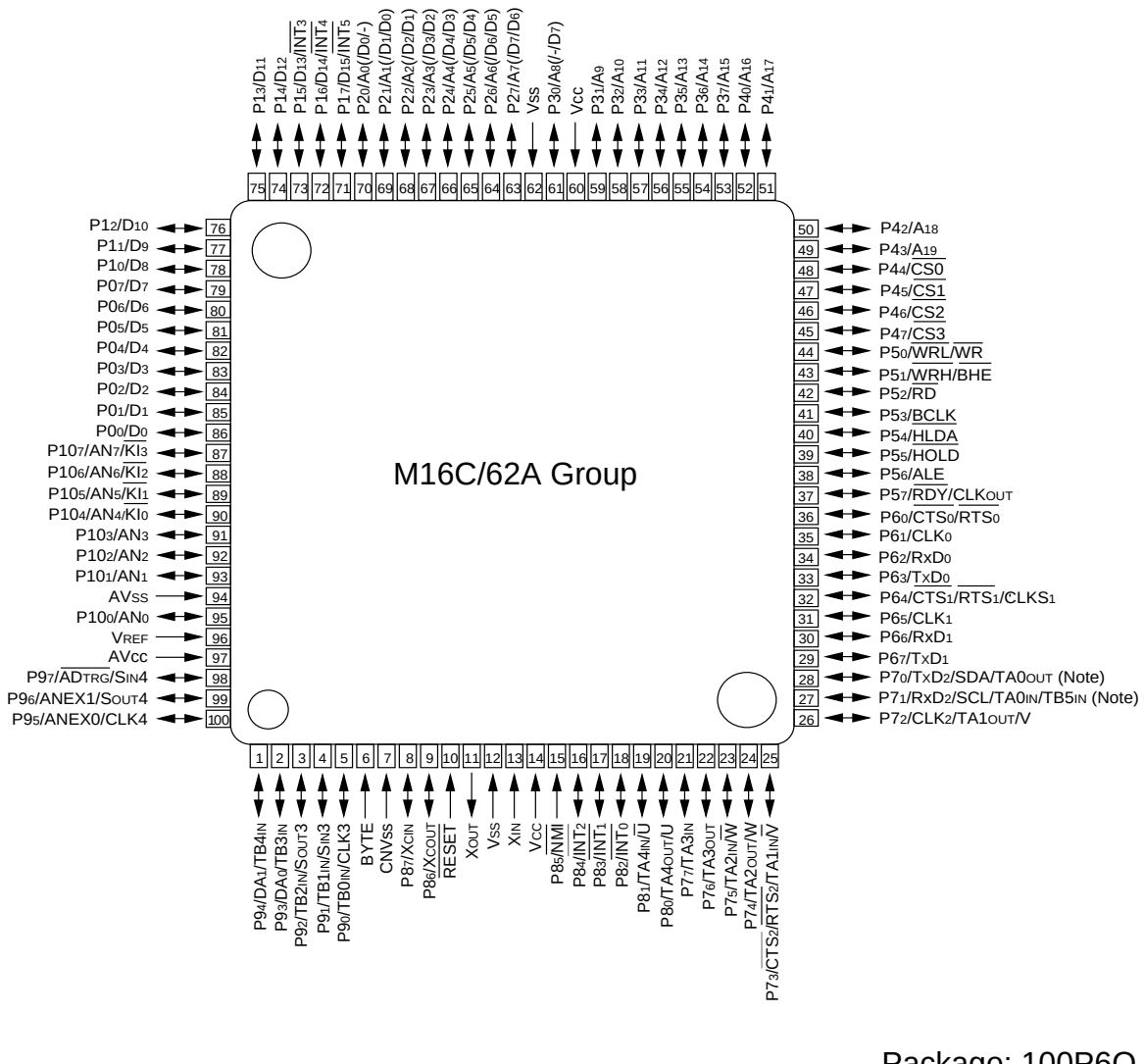


Note: P70 and P71 are N channel open-drain output pin.

Package: 100P6S-A

Figure 1.1.1. Pin configuration (top view)

PIN CONFIGURATION (top view)



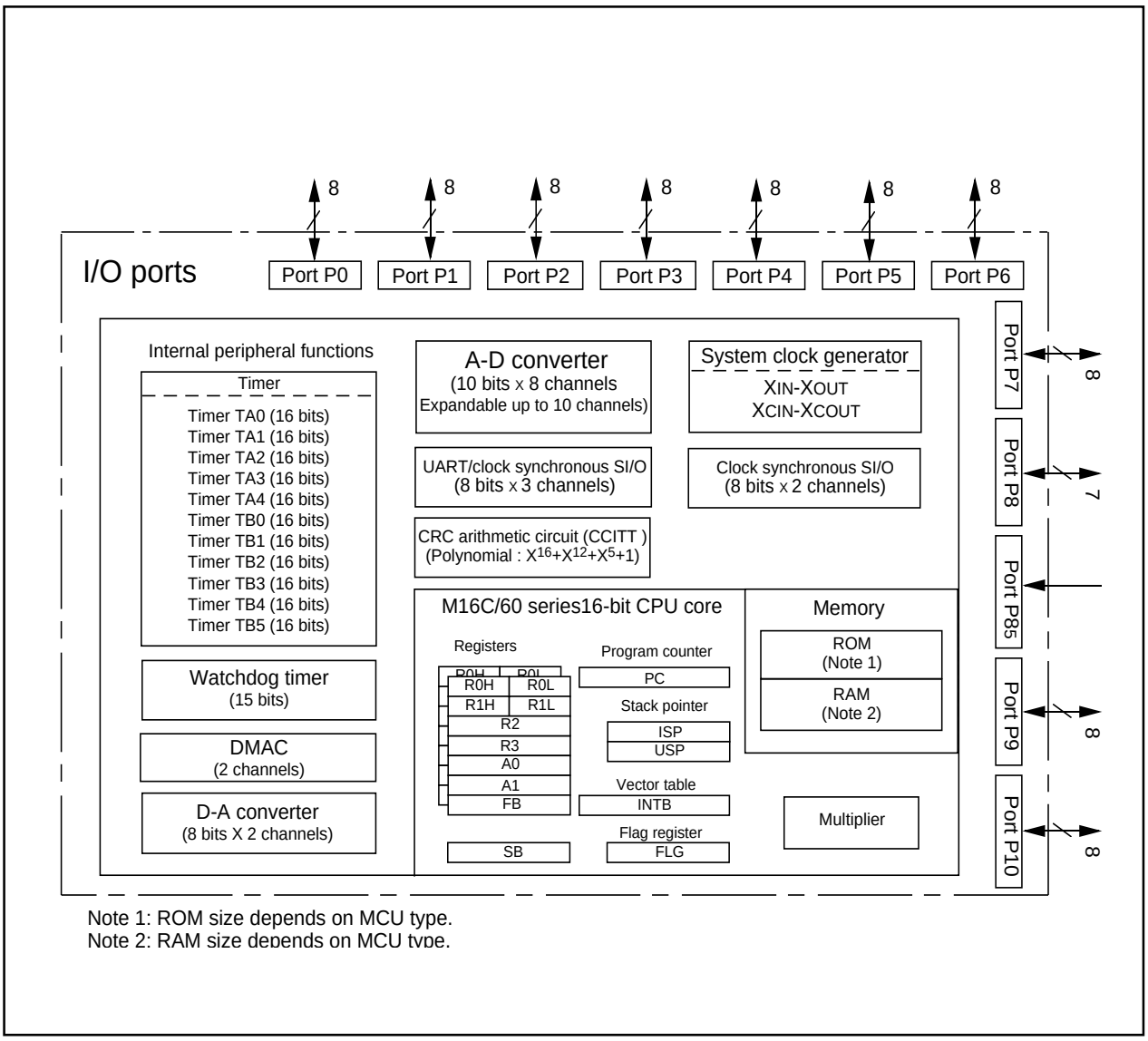
Package: 100P6Q-A

Note: P70 and P71 are N channel open-drain output pin.

Figure 1.1.2. Pin configuration (top view)

Block Diagram

Figure 1.1.3 is a block diagram of the M16C/62A group.



Note 1: ROM size depends on MCU type.
Note 2: RAM size depends on MCU type.

Figure 1.1.3. Block diagram of M16C/62A group

Performance Outline

Table 1.1.1 is a performance outline of M16C/62A group.

Table 1.1.1. Performance outline of M16C/62A group

Item		Performance
Number of basic instructions		91 instructions
Shortest instruction execution time		62.5ns(f(XIN)=16MHz, VCC=5V) 100ns (f(XIN)=10MHz, VCC=3V, with software one-wait) : Mask ROM, flash memory 5V version
Memory capacity	ROM	(See the figure 1.1.4. ROM Expansion)
	RAM	3K to 20K bytes
I/O port	P0 to P10 (except P85)	8 bits x 10, 7 bits x 1
Input port	P85	1 bit x 1
Multifunction timer	TA0, TA1, TA2, TA3, TA4	16 bits x 5
	TB0, TB1, TB2, TB3, TB4, TB5	16 bits x 6
Serial I/O	UART0, UART1, UART2	(UART or clock synchronous) x 3
	SI/O3, SI/O4	(Clock synchronous) x 2
A-D converter		10 bits x (8 + 2) channels
D-A converter		8 bits x 2
DMAC		2 channels (trigger: 24 sources)
CRC calculation circuit		CRC-CCITT
Watchdog timer		15 bits x 1 (with prescaler)
Interrupt		25 internal and 8 external sources, 4 software sources, 7 levels
Clock generating circuit		2 built-in clock generation circuits (built-in feedback resistor, and external ceramic or quartz oscillator)
Supply voltage		4.2V to 5.5V (f(XIN)=16MHz, without software wait) : Mask ROM, flash memory 5V version 2.7V to 5.5V (f(XIN)=10MHz with software one-wait) : Mask ROM, flash memory 5V version
Power consumption		25.5mW (f(XIN) = 10MHz, VCC=3V with software one-wait)
I/O characteristics	I/O withstand voltage	5V
	Output current	5mA
Memory expansion		Available (to a maximum of 1M bytes)
Device configuration		CMOS high performance silicon gate
Package		100-pin plastic mold QFP

Mitsubishi plans to release the following products in the M16C/62A group:

(1) Support for mask ROM version, external ROM version, and flash memory version

(2) ROM capacity

(3) Package

100P6S-A : Plastic molded QFP (mask ROM, and flash memory versions)

100P6Q-A : Plastic molded QFP(mask ROM, and flash memory versions)

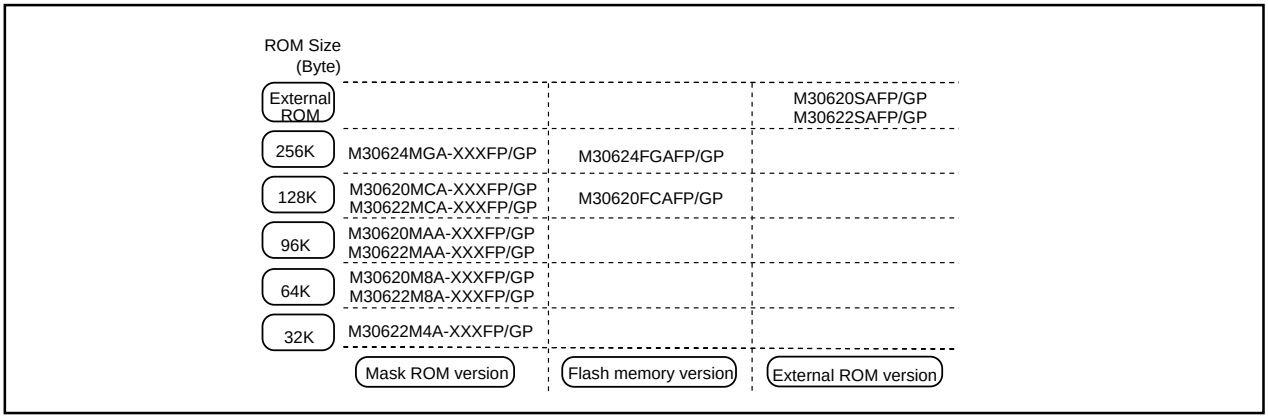


Figure 1.1.4. ROM expansion

The M16C/62A group products currently supported are listed in Table 1.1.2.

Table 1.1.2. M16C/62A group

June, 2001

Type No.	ROM capacity	RAM capacity	Package type	Remarks
M30622M4A-XXXFP	32K byte	3K byte	100P6S-A	Mask ROM version
M30622M4A-XXXGP			100P6Q-A	
M30620M8A-XXXFP	64K byte	10K byte	100P6S-A	
M30620M8A-XXXGP			100P6Q-A	
M30622M8A-XXXFP		4K byte	100P6S-A	
M30622M8A-XXXGP			100P6Q-A	
M30620MAA-XXXFP	96K byte	10K byte	100P6S-A	
M30620MAA-XXXGP			100P6Q-A	
M30622MAA-XXXFP		5K byte	100P6S-A	
M30622MAA-XXXGP			100P6Q-A	
M30620MCA-XXXFP	128K byte	10K byte	100P6S-A	
M30620MCA-XXXGP			100P6Q-A	
M30622MCA-XXXFP		5K byte	100P6S-A	
M30622MCA-XXXGP			100P6Q-A	
M30624MGA-XXXFP	256K byte	20K byte	100P6S-A	
M30624MGA-XXXGP			100P6Q-A	
M30620FCAFP	128K byte	10K byte	100P6S-A	Flash memory 5V version
M30620FCAGP			100P6Q-A	
M30624FGAFP	256K byte	20K byte	100P6S-A	
M30624FGAGP			100P6Q-A	
M30620SAFP	———	10K byte	100P6S-A	External ROM version
M30620SAGP			100P6Q-A	
M30622SAFP		3K byte	100P6S-A	
M30622SAGP			100P6Q-A	

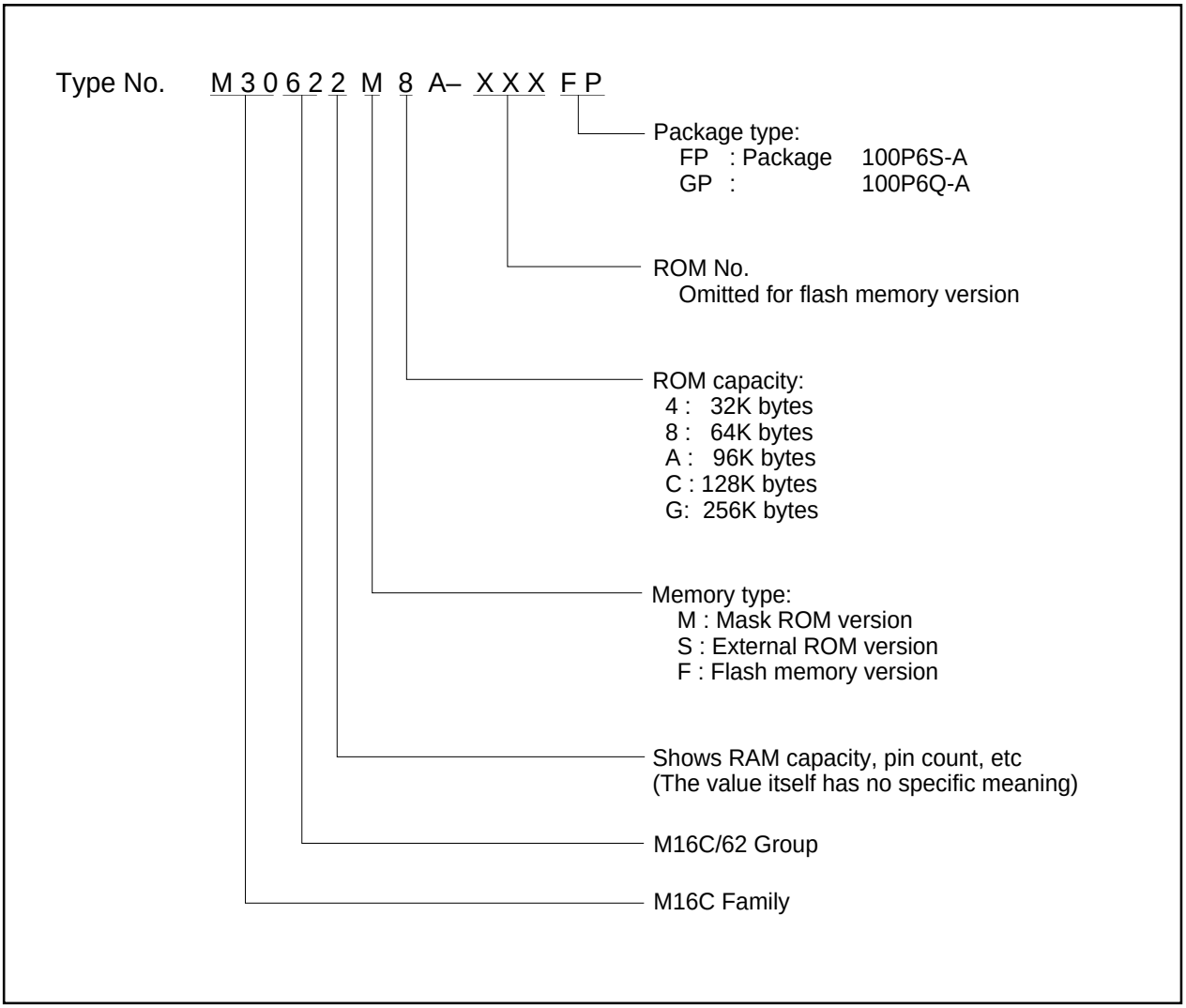


Figure 1.1.5. Type No., memory size, and package

Pin Description

Pin name	Signal name	I/O type	Function
V_{CC}, V_{SS}	Power supply input		Supply 2.7V to 5.5 V to the V_{CC} pin. Supply 0 V to the V_{SS} pin.
CNVSS	CNVSS	Input	This pin switches between processor modes. Connect this pin to the V_{SS} pin when after a reset you want to start operation in single-chip mode (memory expansion mode) or the V_{CC} pin when starting operation in microprocessor mode.
$\overline{\text{RESET}}$	Reset input	Input	A “L” on this input resets the microcomputer.
X_{IN} X_{OUT}	Clock input Clock output	Input Output	These pins are provided for the main clock generating circuit.Connect a ceramic resonator or crystal between the X_{IN} and the X_{OUT} pins. To use an externally derived clock, input it to the X_{IN} pin and leave the X_{OUT} pin open.
BYTE	External data bus width select input	Input	This pin selects the width of an external data bus. A 16-bit width is selected when this input is “L”; an 8-bit width is selected when this input is “H”. This input must be fixed to either “H” or “L”. Connect this pin to the V_{SS} pin when not using external data bus.
AVCC	Analog power supply input		This pin is a power supply input for the A-D converter. Connect this pin to V_{CC}.
AVSS	Analog power supply input		This pin is a power supply input for the A-D converter. Connect this pin to V_{SS}.
V_{REF}	Reference voltage input	Input	This pin is a reference voltage input for the A-D converter.
$P0_0$ to $P0_7$	I/O port P0	Input/output	This is an 8-bit CMOS I/O port. It has an input/output port direction register that allows the user to set each pin for input or output individually. When used for input in single-chip mode, the port can be set to have or not have a pull-up resistor in units of four bits by software. In memory expansion and microprocessor modes, selection of the internal pull-resistor is not available.
D_0 to D_7		Input/output	When set as a separate bus, these pins input and output data (D_0–D_7).
$P1_0$ to $P1_7$	I/O port P1	Input/output	This is an 8-bit I/O port equivalent to P0. $P1_5$ to $P1_7$ also function as external interrupt pins as selected by software.
D_8 to D_{15}		Input/output	When set as a separate bus, these pins input and output data (D_8–D_{15}).
$P2_0$ to $P2_7$	I/O port P2	Input/output	This is an 8-bit I/O port equivalent to P0.
A_0 to A_7		Output	These pins output 8 low-order address bits (A_0–A_7).
A_0/D_0 to A_7/D_7		Input/output	If the external bus is set as an 8-bit wide multiplexed bus, these pins input and output data (D_0–D_7) and output 8 low-order address bits (A_0–A_7) separated in time by multiplexing.
A_0 A_1/D_0 to A_7/D_6		Output Input/output	If the external bus is set as a 16-bit wide multiplexed bus, these pins input and output data (D_0–D_6) and output address (A_1–A_7) separated in time by multiplexing. They also output address (A_0).
$P3_0$ to $P3_7$	I/O port P3	Input/output	This is an 8-bit I/O port equivalent to P0.
A_8 to A_{15}		Output	These pins output 8 middle-order address bits (A_8–A_{15}).
A_8/D_7, A_9 to A_{15}		Input/output Output	If the external bus is set as a 16-bit wide multiplexed bus, these pins input and output data (D_7) and output address (A_8) separated in time by multiplexing. They also output address (A_9–A_{15}).
$P4_0$ to $P4_7$	I/O port P4	Input/output	This is an 8-bit I/O port equivalent to P0.
A_{16} to A_{19}, $\overline{CS_0}$ to $\overline{CS_3}$		Output Output	These pins output A_{16}–A_{19} and $\overline{CS_0}$–$\overline{CS_3}$ signals. A_{16}–A_{19} are 4 high-order address bits. $\overline{CS_0}$–$\overline{CS_3}$ are chip select signals used to specify an access space.

Pin Description

Pin name	Signal name	I/O type	Function
P5$_0$ to P5$_7$	I/O port P5	Input/output	This is an 8-bit I/O port equivalent to P0. In single-chip mode, P5$_7$ in this port outputs a divide-by-8 or divide-by-32 clock of X_{IN} or a clock of the same frequency as X_{CIN} as selected by software.
$\overline{WRL}$ / $\overline{WR}$, $\overline{WRH}$ / BHE, $\overline{RD}$, $\overline{BCLK}$, $\overline{HLDA}$, HOLD, ALE, $\overline{RDY}$		Output Output Output Output Output Input Output Input	Output $\overline{WRL}$, $\overline{WRH}$ ($\overline{WR}$ and $\overline{BHE}$), $\overline{RD}$, $\overline{BCLK}$, $\overline{HLDA}$, and ALE signals. $\overline{WRL}$ and $\overline{WRH}$, and $\overline{BHE}$ and $\overline{WR}$ can be switched using software control. ■ $\overline{WRL}$, $\overline{WRH}$, and $\overline{RD}$ selected With a 16-bit external data bus, data is written to even addresses when the $\overline{WRL}$ signal is "L" and to the odd addresses when the $\overline{WRH}$ signal is "L". Data is read when $\overline{RD}$ is "L". ■ $\overline{WR}$, $\overline{BHE}$, and $\overline{RD}$ selected Data is written when $\overline{WR}$ is "L". Data is read when $\overline{RD}$ is "L". Odd addresses are accessed when $\overline{BHE}$ is "L". Use this mode when using an 8-bit external data bus. While the input level at the $\overline{HOLD}$ pin is "L", the microcomputer is placed in the hold state. While in the hold state, $\overline{HLDA}$ outputs a "L" level. ALE is used to latch the address. While the input level of the $\overline{RDY}$ pin is "L", the microcomputer is in the ready state.
P6$_0$ to P6$_7$	I/O port P6	Input/output	This is an 8-bit I/O port equivalent to P0. When used for input in single-chip, memory expansion, and microprocessor modes, the port can be set to have or not have a pull-up resistor in units of four bits by software. Pins in this port also function as UART0 and UART1 I/O pins as selected by software.
P7$_0$ to P7$_7$	I/O port P7	Input/output	This is an 8-bit I/O port equivalent to P6 (P7$_0$ and P7$_1$ are N channel open-drain output). Pins in this port also function as timer A$_0$–A$_3$, timer B5 or UART2 I/O pins as selected by software.
P8$_0$ to P8$_4$, P8$_6$, P8$_7$, P8$_5$	I/O port P8 I/O port P8$_5$	Input/output Input/output Input/output Input	P8$_0$ to P8$_4$, P8$_6$, and P8$_7$ are I/O ports with the same functions as P6. Using software, they can be made to function as the I/O pins for timer A4 and the input pins for external interrupts. P8$_6$ and P8$_7$ can be set using software to function as the I/O pins for a sub clock generation circuit. In this case, connect a quartz oscillator between P8$_6$ (X_{COUT} pin) and P8$_7$ (X_{CIN} pin). P8$_5$ is an input-only port that also functions for $\overline{NMI}$. The $\overline{NMI}$ interrupt is generated when the input at this pin changes from "H" to "L". The $\overline{NMI}$ function cannot be cancelled using software. The pull-up cannot be set for this pin.
P9$_0$ to P9$_7$	I/O port P9	Input/output	This is an 8-bit I/O port equivalent to P6. Pins in this port also function as SI/O3, 4 I/O pins, Timer B0–B4 input pins, D-A converter output pins, A-D converter extended input pins, or A-D trigger input pins as selected by software.
P10$_0$ to P10$_7$	I/O port P10	Input/output	This is an 8-bit I/O port equivalent to P6. Pins in this port also function as A-D converter input pins as selected by software. Furthermore, P10$_4$ –P10$_7$ also function as input pins for the key input interrupt function.

Operation of Functional Blocks

The M16C/62A group accommodates certain units in a single chip. These units include ROM and RAM to store instructions and data and the central processing unit (CPU) to execute arithmetic/logic operations. Also included are peripheral units such as timers, serial I/O, D-A converter, DMAC, CRC calculation circuit, A-D converter, and I/O ports.

The following explains each unit.

Memory

Figure 1.3.1 is a memory map of the M16C/62A group. The address space extends the 1M bytes from address 00000_{16} to $FFFFF_{16}$. From $FFFFF_{16}$ down is ROM. For example, in the M30622MCA-XXXFP, there is 128K bytes of internal ROM from $E0000_{16}$ to $FFFFF_{16}$. The vector table for fixed interrupts such as the reset and $\overline{NMI}$ are mapped to $FFFDC_{16}$ to $FFFFF_{16}$. The starting address of the interrupt routine is stored here. The address of the vector table for timer interrupts, etc., can be set as desired using the internal register (INTB). See the section on interrupts for details.

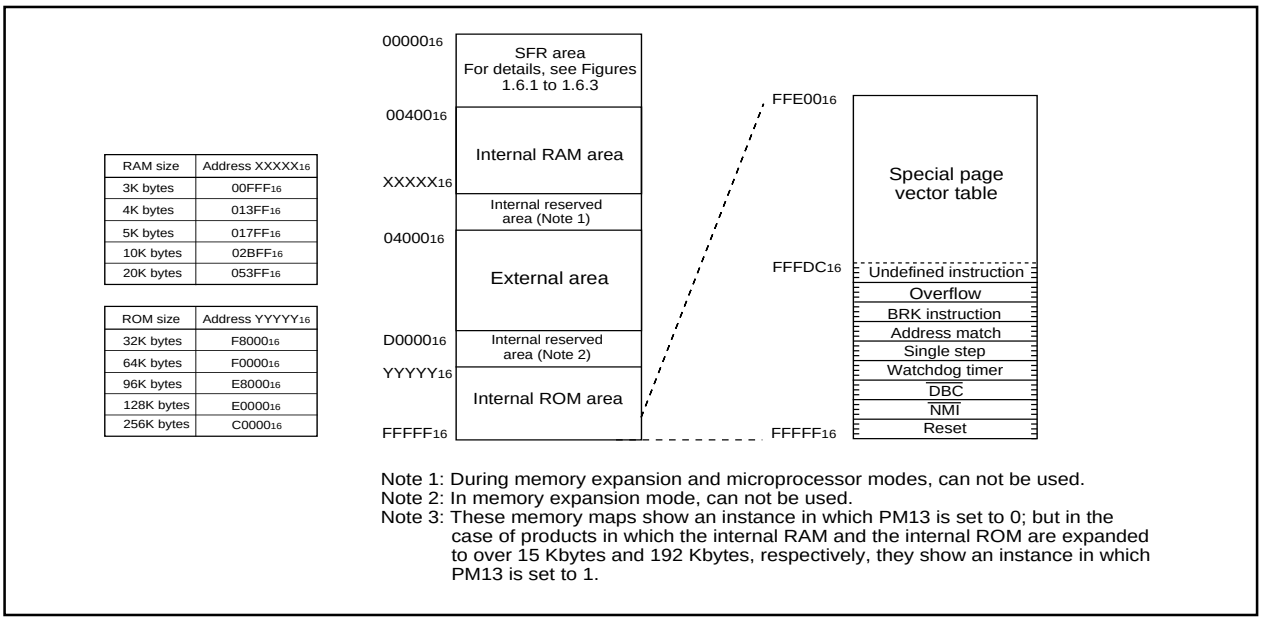
From 00400_{16} up is RAM. For example, in the M30622MCA-XXXFP, 5K bytes of internal RAM is mapped to the space from 00400_{16} to $017FF_{16}$. In addition to storing data, the RAM also stores the stack used when calling subroutines and when interrupts are generated.

The SFR area is mapped to 00000_{16} to $003FF_{16}$. This area accommodates the control registers for peripheral devices such as I/O ports, A-D converter, serial I/O, and timers, etc. Figures 1.6.1 to 1.6.3 are location of peripheral unit control registers. Any part of the SFR area that is not occupied is reserved and cannot be used for other purposes.

The special page vector table is mapped to $FFE00_{16}$ to $FFFDB_{16}$. If the starting addresses of subroutines or the destination addresses of jumps are stored here, subroutine call instructions and jump instructions can be used as 2-byte instructions, reducing the number of program steps.

In memory expansion mode and microprocessor mode, a part of the spaces are reserved and cannot be used. For example, in the M30622MCA-XXXFP, the following spaces cannot be used.

- The space between 01800_{16} and $03FFF_{16}$ (Memory expansion and microprocessor modes)
- The space between $D0000_{16}$ and $DFFFF_{16}$ (Memory expansion mode)



Note 1: During memory expansion and microprocessor modes, can not be used.
Note 2: In memory expansion mode, can not be used.
Note 3: These memory maps show an instance in which PM13 is set to 0; but in the case of products in which the internal RAM and the internal ROM are expanded to over 15 Kbytes and 192 Kbytes, respectively, they show an instance in which PM13 is set to 1.

Figure 1.3.1. Memory map

Central Processing Unit (CPU)

The CPU has a total of 13 registers shown in Figure 1.4.1. Seven of these registers (R0, R1, R2, R3, A0, A1, and FB) come in two sets; therefore, these have two register banks.

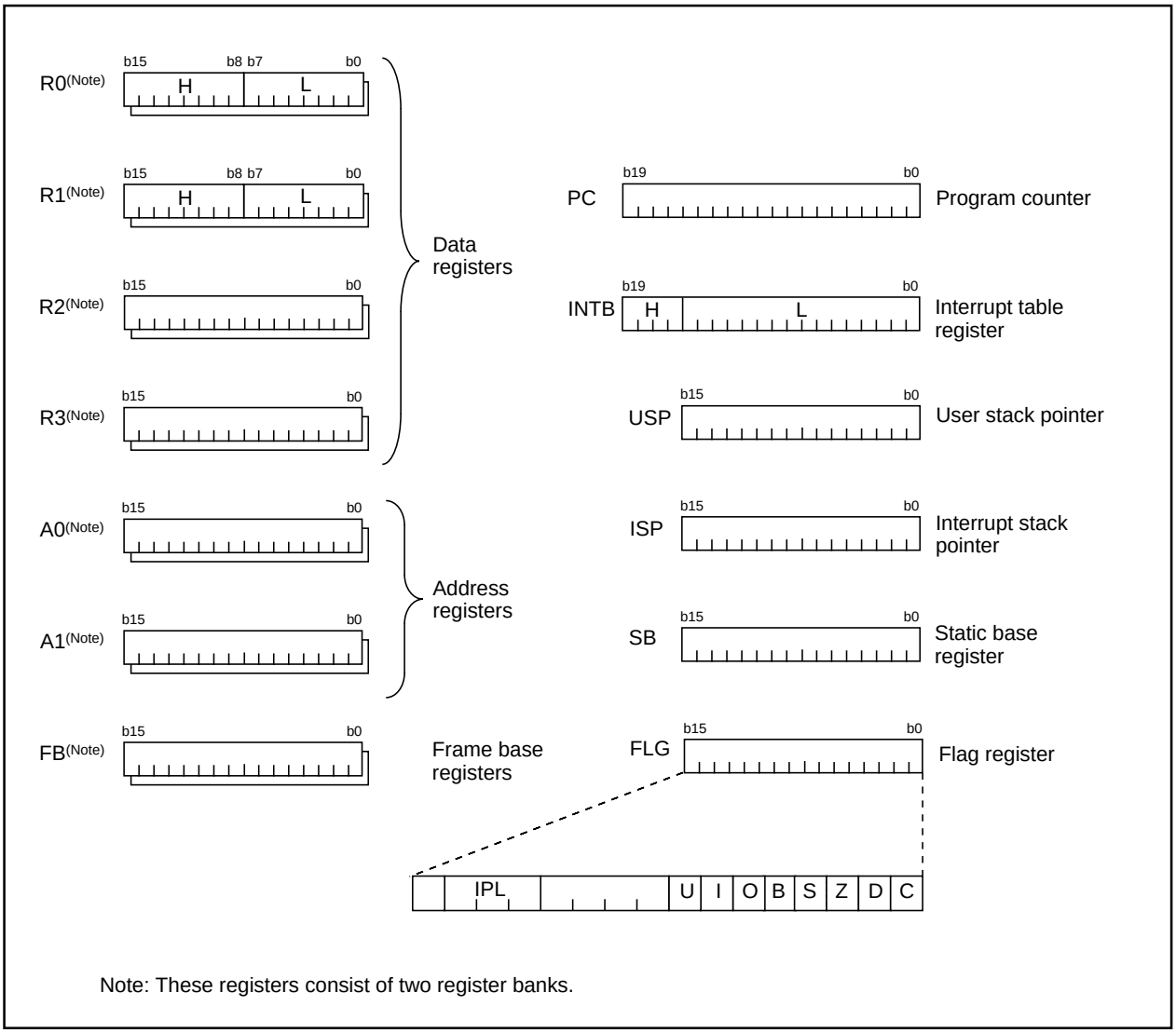


Figure 1.4.1. Central processing unit register

(1) Data registers (R0, R0H, R0L, R1, R1H, R1L, R2, and R3)

Data registers (R0, R1, R2, and R3) are configured with 16 bits, and are used primarily for transfer and arithmetic/logic operations.

Registers R0 and R1 each can be used as separate 8-bit data registers, high-order bits as (R0H/R1H), and low-order bits as (R0L/R1L). In some instructions, registers R2 and R0, as well as R3 and R1 can use as 32-bit data registers (R2R0/R3R1).

(2) Address registers (A0 and A1)

Address registers (A0 and A1) are configured with 16 bits, and have functions equivalent to those of data registers. These registers can also be used for address register indirect addressing and address register relative addressing.

In some instructions, registers A1 and A0 can be combined for use as a 32-bit address register (A1A0).

(3) Frame base register (FB)

Frame base register (FB) is configured with 16 bits, and is used for FB relative addressing.

(4) Program counter (PC)

Program counter (PC) is configured with 20 bits, indicating the address of an instruction to be executed.

(5) Interrupt table register (INTB)

Interrupt table register (INTB) is configured with 20 bits, indicating the start address of an interrupt vector table.

(6) Stack pointer (USP/ISP)

Stack pointer comes in two types: user stack pointer (USP) and interrupt stack pointer (ISP), each configured with 16 bits.

Your desired type of stack pointer (USP or ISP) can be selected by a stack pointer select flag (U flag). This flag is located at the position of bit 7 in the flag register (FLG).

(7) Static base register (SB)

Static base register (SB) is configured with 16 bits, and is used for SB relative addressing.

(8) Flag register (FLG)

Flag register (FLG) is configured with 11 bits, each bit is used as a flag. Figure 1.4.2 shows the flag register (FLG). The following explains the function of each flag:

- **Bit 0: Carry flag (C flag)**
  This flag retains a carry, borrow, or shift-out bit that has occurred in the arithmetic/logic unit.
- **Bit 1: Debug flag (D flag)**
  This flag enables a single-step interrupt.
  When this flag is “1”, a single-step interrupt is generated after instruction execution. This flag is cleared to “0” when the interrupt is acknowledged.
- **Bit 2: Zero flag (Z flag)**
  This flag is set to “1” when an arithmetic operation resulted in 0; otherwise, cleared to “0”.
- **Bit 3: Sign flag (S flag)**
  This flag is set to “1” when an arithmetic operation resulted in a negative value; otherwise, cleared to “0”.
- **Bit 4: Register bank select flag (B flag)**
  This flag chooses a register bank. Register bank 0 is selected when this flag is “0” ; register bank 1 is selected when this flag is “1”.
- **Bit 5: Overflow flag (O flag)**
  This flag is set to “1” when an arithmetic operation resulted in overflow; otherwise, cleared to “0”.
- **Bit 6: Interrupt enable flag (I flag)**
  This flag enables a maskable interrupt.
  An interrupt is disabled when this flag is “0”, and is enabled when this flag is “1”. This flag is cleared to “0” when the interrupt is acknowledged.

- **Bit 7: Stack pointer select flag (U flag)**
  Interrupt stack pointer (ISP) is selected when this flag is “0” ; user stack pointer (USP) is selected when this flag is “1”.
  This flag is cleared to “0” when a hardware interrupt is acknowledged or an INT instruction of software interrupt Nos. 0 to 31 is executed.
- **Bits 8 to 11: Reserved area**
- **Bits 12 to 14: Processor interrupt priority level (IPL)**
  Processor interrupt priority level (IPL) is configured with three bits, for specification of up to eight processor interrupt priority levels from level 0 to level 7.
  If a requested interrupt has priority greater than the processor interrupt priority level (IPL), the interrupt is enabled.
- **Bit 15: Reserved area**

The C, Z, S, and O flags are changed when instructions are executed. See the software manual for details.

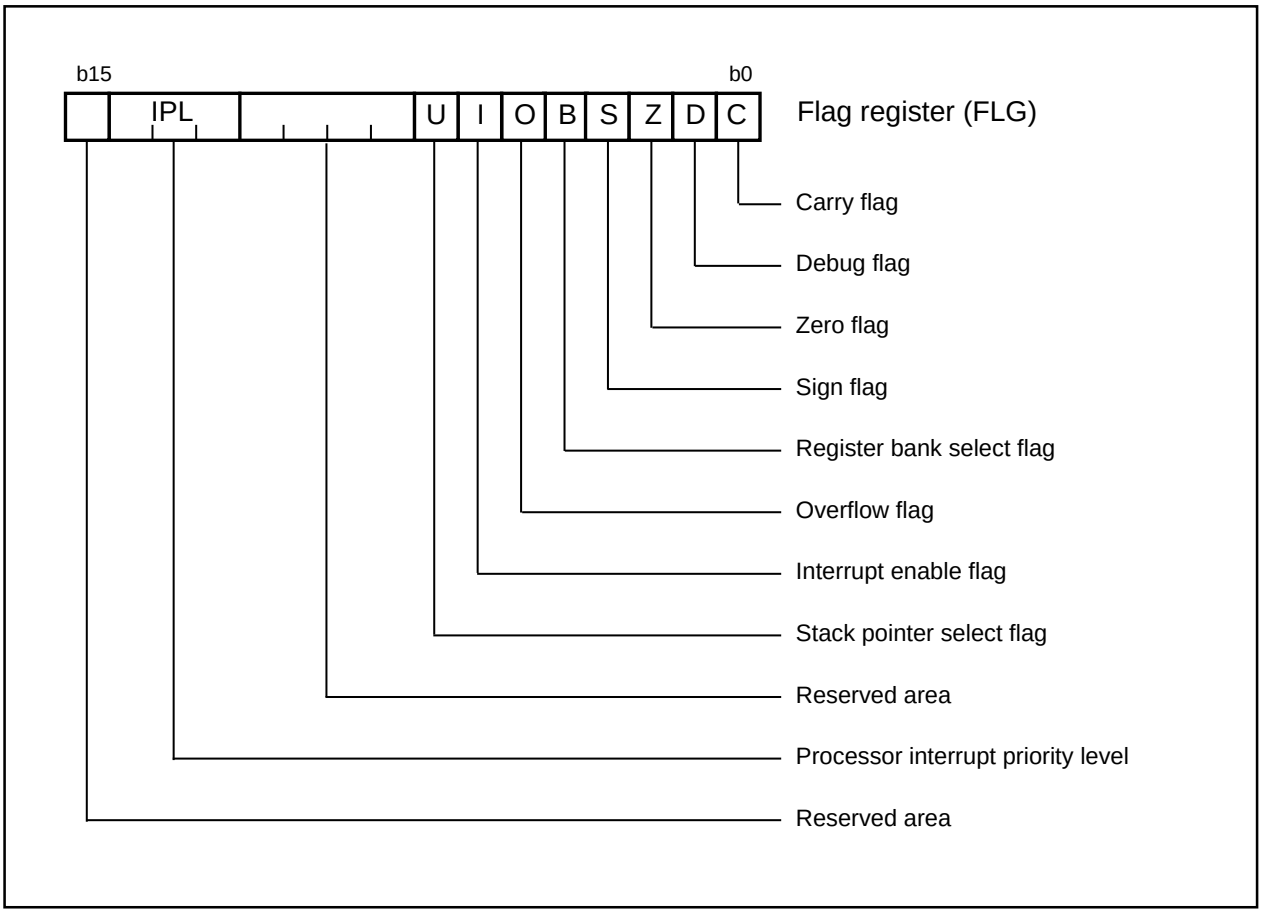


Figure 1.4.2. Flag register (FLG)

Reset

There are two kinds of resets; hardware and software. In both cases, operation is the same after the reset. (See “Software Reset” for details of software resets.) This section explains on hardware resets.

When the supply voltage is in the range where operation is guaranteed, a reset is effected by holding the reset pin level “L” (0.2Vcc max.) for at least 20 cycles. When the reset pin level is then returned to the “H” level while main clock is stable, the reset status is cancelled and program execution resumes from the address in the reset vector table.

Figure 1.5.1 shows the example reset circuit. Figure 1.5.2 shows the reset sequence.

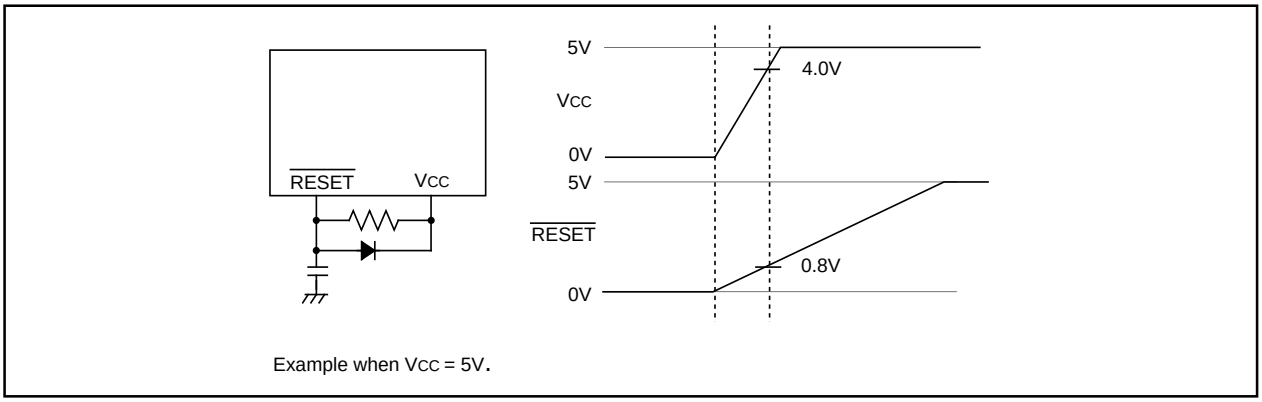


Figure 1.5.1. Example reset circuit

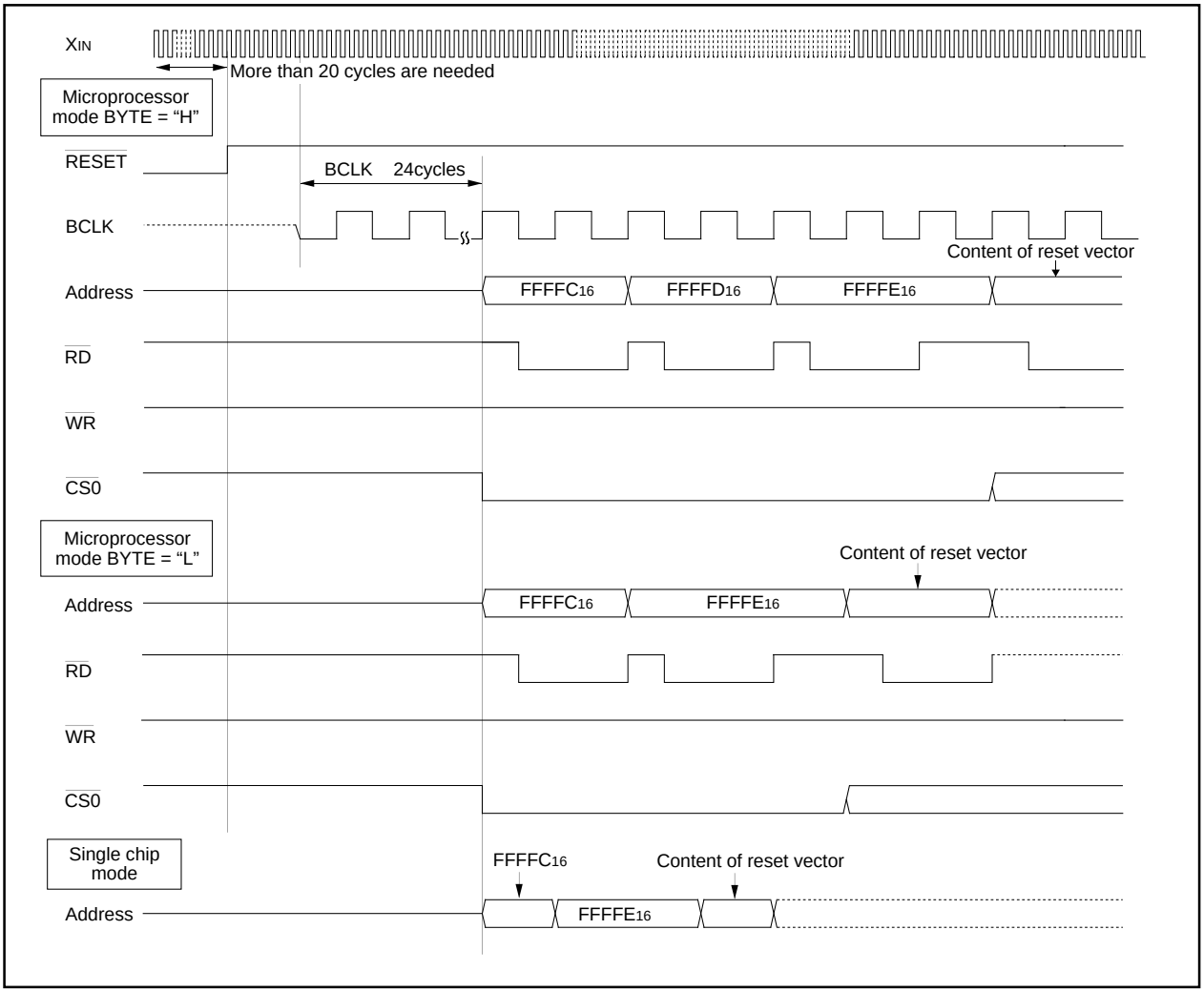


Figure 1.5.2. Reset sequence

Table 1.5.1 shows the statuses of the other pins while the $\overline{\text{RESET}}$ pin level is “L”. Figures 1.5.3 and 1.5.4 show the internal status of the microcomputer immediately after the reset is cancelled.

Table 1.5.1. Pin status when $\overline{\text{RESET}}$ pin level is “L”

Pin name	Status		
	CNVSS = VSS	CNVSS = VCC	
		BYTE = VSS	BYTE = VCC
P0	Input port (floating)	Data input (floating)	Data input (floating)
P1	Input port (floating)	Data input (floating)	Input port (floating)
P2, P3, P40 to P43	Input port (floating)	Address output (undefined)	Address output (undefined)
P44	Input port (floating)	$\overline{\text{CS0}}$ output (“H” level is output)	$\overline{\text{CS0}}$ output (“H” level is output)
P45 to P47	Input port (floating)	Input port (floating) (pull-up resistor is on)	Input port (floating) (pull-up resistor is on)
P50	Input port (floating)	$\overline{\text{WR}}$ output (“H” level is output)	$\overline{\text{WR}}$ output (“H” level is output)
P51	Input port (floating)	$\overline{\text{BHE}}$ output (undefined)	$\overline{\text{BHE}}$ output (undefined)
P52	Input port (floating)	$\overline{\text{RD}}$ output (“H” level is output)	$\overline{\text{RD}}$ output (“H” level is output)
P53	Input port (floating)	BCLK output	BCLK output
P54	Input port (floating)	$\overline{\text{HLDA}}$ output (The output value depends on the input to the $\overline{\text{HOLD}}$ pin)	$\overline{\text{HLDA}}$ output (The output value depends on the input to the $\overline{\text{HOLD}}$ pin)
P55	Input port (floating)	$\overline{\text{HOLD}}$ input (floating)	$\overline{\text{HOLD}}$ input (floating)
P56	Input port (floating)	ALE output (“L” level is output)	ALE output (“L” level is output)
P57	Input port (floating)	$\overline{\text{RDY}}$ input (floating)	$\overline{\text{RDY}}$ input (floating)
P6, P7, P80 to P84, P86, P87, P9, P10	Input port (floating)	Input port (floating)	Input port (floating)

(1) Processor mode register 0 (Note 1) (0004_{16}) ··· 00_{16}

(2) Processor mode register 1 (0005_{16}) ··· 0 0 0 0 0 x x 0

(3) System clock control register 0 (0006_{16}) ··· 0 1 0 0 1 0 0 0

(4) System clock control register 1 (0007_{16}) ··· 0 0 1 0 0 0 0 0

(5) Chip select control register (0008_{16}) ··· 0 0 0 0 0 0 0 1

(6) Address match interrupt enable register (0009_{16}) ··· x x x x x x 0 0

(7) Protect register ($000A_{16}$) ··· x x x x x 0 0 0

(8) Watchdog timer control register ($000F_{16}$) ··· 0 0 0 ? ? ? ? ?

(9) Address match interrupt register 0 (0010_{16}) ··· 00_{16}

(0011_{16}) ··· 00_{16}

(0012_{16}) ··· x x x x 0 0 0 0

(10) Address match interrupt register 1 (0014_{16}) ··· 00_{16}

(0015_{16}) ··· 00_{16}

(0016_{16}) ··· x x x x 0 0 0 0

(11) DMA0 control register ($002C_{16}$) ··· 0 0 0 0 0 ? 0 0

(12) DMA1 control register ($003C_{16}$) ··· 0 0 0 0 0 ? 0 0

(13) INT3 interrupt control register (0044_{16}) ··· x x 0 0 ? 0 0 0

(14) Timer B5 interrupt control register (0045_{16}) ··· x x x x ? 0 0 0

(15) Timer B4 interrupt control register (0046_{16}) ··· x x x x ? 0 0 0

(16) Timer B3 interrupt control register (0047_{16}) ··· x x x x ? 0 0 0

(17) SI/O4 interrupt control register (0048_{16}) ··· x x 0 0 ? 0 0 0

(18) SI/O3 interrupt control register (0049_{16}) ··· x x 0 0 ? 0 0 0

(19) Bus collision detection interrupt control register ($004A_{16}$) ··· x x x x ? 0 0 0

(20) DMA0 interrupt control register ($004B_{16}$) ··· x x x x ? 0 0 0

(21) DMA1 interrupt control register ($004C_{16}$) ··· x x x x ? 0 0 0

(22) Key input interrupt control register ($004D_{16}$) ··· x x x x ? 0 0 0

(23) A-D conversion interrupt control register ($004E_{16}$) ··· x x x x ? 0 0 0

(24) UART2 transmit interrupt control register ($004F_{16}$) ··· x x x x ? 0 0 0

(25) UART2 receive interrupt control register (0050_{16}) ··· x x x x ? 0 0 0

(26) UART0 transmit interrupt control register (0051_{16}) ··· x x x x ? 0 0 0

(27) UART0 receive interrupt control register (0052_{16}) ··· x x x x ? 0 0 0

(28) UART1 transmit interrupt control register (0053_{16}) ··· x x x x ? 0 0 0

(29) UART1 receive interrupt control register (0054_{16}) ··· x x x x ? 0 0 0

(30) Timer A0 interrupt control register (0055_{16}) ··· x x x x ? 0 0 0

(31) Timer A1 interrupt control register (0056_{16}) ··· x x x x ? 0 0 0

(32) Timer A2 interrupt control register (0057_{16}) ··· x x x x ? 0 0 0

(33) Timer A3 interrupt control register (0058_{16}) ··· x x x x ? 0 0 0

(34) Timer A4 interrupt control register (0059_{16}) ··· x x x x ? 0 0 0

(35) Timer B0 interrupt control register ($005A_{16}$) ··· x x x x ? 0 0 0

(36) Timer B1 interrupt control register ($005B_{16}$) ··· x x x x ? 0 0 0

(37) Timer B2 interrupt control register ($005C_{16}$) ··· x x x x ? 0 0 0

(38) INT0 interrupt control register ($005D_{16}$) ··· x x 0 0 ? 0 0 0

(39) INT1 interrupt control register ($005E_{16}$) ··· x x 0 0 ? 0 0 0

(40) INT2 interrupt control register ($005F_{16}$) ··· x x 0 0 ? 0 0 0

(41) Timer B3,4,5 count start flag (0340_{16}) ··· 0 0 0 x x x x x

(42) Three-phase PWM control register 0 (0348_{16}) ··· 00_{16}

(43) Three-phase PWM control register 1 (0349_{16}) ··· 00_{16}

(44) Three-phase output buffer register 0 ($034A_{16}$) ··· 00_{16}

(45) Three-phase output buffer register 1 ($034B_{16}$) ··· 00_{16}

(46) Timer B3 mode register ($035B_{16}$) ··· 0 0 ? x 0 0 0 0

(47) Timer B4 mode register ($035C_{16}$) ··· 0 0 ? x 0 0 0 0

(48) Timer B5 mode register ($035D_{16}$) ··· 0 0 ? x 0 0 0 0

(49) Interrupt cause select register ($035F_{16}$) ··· 00_{16}

(50) SI/O3 control register (0362_{16}) ··· 40_{16}

(51) SI/O4 control register (0366_{16}) ··· 40_{16}

(52) UART2 special mode register 3 (Note 2) (0375_{16}) ··· ?

(53) UART2 special mode register 2 (0376_{16}) ··· 00_{16}

(54) UART2 special mode register (0377_{16}) ··· 00_{16}

(55) UART2 transmit/receive mode register (0378_{16}) ··· 00_{16}

(56) UART2 transmit/receive control register 0 ($037C_{16}$) ··· 0 0 0 0 1 0 0 0

(57) UART2 transmit/receive control register 1 ($037D_{16}$) ··· 0 0 0 0 0 0 1 0

x : Nothing is mapped to this bit
? : Undefined

The content of other registers and RAM is undefined when the microcomputer is reset. The initial values must therefore be set.

Note 1: When the V_{CC} level is applied to the CNV_{SS} pin, it is 03_{16} at a reset.
Note 2: "00_{16}" is read out when set bit 7 (SDDS) of the UART2 special mode register (address 0377_{16}) to "1".

Figure 1.5.3. Device's internal status after a reset is cleared

(58) Count start flag	(0380$_{16}$)··· 00$_{16}$	(84) A-D control register 1	(03D7$_{16}$)··· 00$_{16}$
(59) Clock prescaler reset flag	(0381$_{16}$)··· 0 x x x x x x ?	(85) D-A control register	(03DC$_{16}$)··· 00$_{16}$
(60) One-shot start flag	(0382$_{16}$)··· 0 0 x 0 0 0 0 0	(86) Port P0 direction register	(03E2$_{16}$)··· 00$_{16}$
(61) Trigger select flag	(0383$_{16}$)··· 00$_{16}$	(87) Port P1 direction register	(03E3$_{16}$)··· 00$_{16}$
(62) Up-down flag	(0384$_{16}$)··· 00$_{16}$	(88) Port P2 direction register	(03E6$_{16}$)··· 00$_{16}$
(63) Timer A0 mode register	(0396$_{16}$)··· 00$_{16}$	(89) Port P3 direction register	(03E7$_{16}$)··· 00$_{16}$
(64) Timer A1 mode register	(0397$_{16}$)··· 00$_{16}$	(90) Port P4 direction register	(03EA$_{16}$)··· 00$_{16}$
(65) Timer A2 mode register	(0398$_{16}$)··· 00$_{16}$	(91) Port P5 direction register	(03EB$_{16}$)··· 00$_{16}$
(66) Timer A3 mode register	(0399$_{16}$)··· 00$_{16}$	(92) Port P6 direction register	(03EE$_{16}$)··· 00$_{16}$
(67) Timer A4 mode register	(039A$_{16}$)··· 00$_{16}$	(93) Port P7 direction register	(03EF$_{16}$)··· 00$_{16}$
(68) Timer B0 mode register	(039B$_{16}$)··· 0 0 ? x 0 0 0 0	(94) Port P8 direction register	(03F2$_{16}$)··· 0 0 x 0 0 0 0 0
(69) Timer B1 mode register	(039C$_{16}$)··· 0 0 ? x 0 0 0 0	(95) Port P9 direction register	(03F3$_{16}$)··· 00$_{16}$
(70) Timer B2 mode register	(039D$_{16}$)··· 0 0 ? x 0 0 0 0	(96) Port P10 direction register	(03F6$_{16}$)··· 00$_{16}$
(71) UART0 transmit/receive mode register	(03A0$_{16}$)··· 00$_{16}$	(97) Pull-up control register 0	(03FC$_{16}$)··· 00$_{16}$
(72) UART0 transmit/receive control register 0	(03A4$_{16}$)··· 0 0 0 0 1 0 0 0	(98) Pull-up control register 1(Note1)	(03FD$_{16}$)··· 00$_{16}$
(73) UART0 transmit/receive control register 1	(03A5$_{16}$)··· 0 0 0 0 0 0 1 0	(99) Pull-up control register 2	(03FE$_{16}$)··· 00$_{16}$
(74) UART1 transmit/receive mode register	(03A8$_{16}$)··· 00$_{16}$	(100) Port control register	(03FF$_{16}$)··· 00$_{16}$
(75) UART1 transmit/receive control register 0	(03AC$_{16}$)··· 0 0 0 0 1 0 0 0	(101) Data registers (R0/R1/R2/R3)	0000$_{16}$
(76) UART1 transmit/receive control register 1	(03AD$_{16}$)··· 0 0 0 0 0 0 1 0	(102) Address registers (A0/A1)	0000$_{16}$
(77) UART transmit/receive control register 2	(03B0$_{16}$)··· x 0 0 0 0 0 0 0	(103) Frame base register (FB)	0000$_{16}$
(78) Flash memory control register 1 (Note2)	(03B6$_{16}$)··· ? ? ? ? 0 ? ? ?	(104) Interrupt table register (INTB)	00000$_{16}$
(79) Flash memory control register 0 (Note2)	(03B7$_{16}$)··· x x 0 0 0 0 0 1	(105) User stack pointer (USP)	0000$_{16}$
(80) DMA0 cause select register	(03B8$_{16}$)··· 00$_{16}$	(106) Interrupt stack pointer (ISP)	0000$_{16}$
(81) DMA1 cause select register	(03BA$_{16}$)··· 00$_{16}$	(107) Static base register (SB)	0000$_{16}$
(82) A-D control register 2	(03D4$_{16}$)··· 0 0 0 0 x x x 0	(108) Flag register (FLG)	0000$_{16}$
(83) A-D control register 0	(03D6$_{16}$)··· 0 0 0 0 0 ? ? ?		

x : Nothing is mapped to this bit
? : Undefined

The content of other registers and RAM is undefined when the microcomputer is reset. The initial values must therefore be set.

Note1: When the V_{CC} level is applied to the CNV_{SS} pin, it is 02$_{16}$ at a reset.
Note2: This register is only exist in flash memory version.

Figure 1.5.4. Device's internal status after a reset is cleared

0000₁₆	
0001₁₆	
0002₁₆	
0003₁₆	
0004₁₆	Processor mode register 0 (PM0)
0005₁₆	Processor mode register 1(PM1)
0006₁₆	System clock control register 0 (CM0)
0007₁₆	System clock control register 1 (CM1)
0008₁₆	Chip select control register (CSR)
0009₁₆	Address match interrupt enable register (AIER)
000A₁₆	Protect register (PRCR)
000B₁₆	
000C₁₆	
000D₁₆	
000E₁₆	Watchdog timer start register (WDTS)
000F₁₆	Watchdog timer control register (WDC)
0010₁₆	Address match interrupt register 0 (RMAD0)
0011₁₆	
0012₁₆	
0013₁₆	
0014₁₆	Address match interrupt register 1 (RMAD1)
0015₁₆	
0016₁₆	
0017₁₆	
0018₁₆	
0019₁₆	
001A₁₆	
001B₁₆	
001C₁₆	
001D₁₆	
001E₁₆	
001F₁₆	
0020₁₆	DMA0 source pointer (SAR0)
0021₁₆	
0022₁₆	
0023₁₆	
0024₁₆	DMA0 destination pointer (DAR0)
0025₁₆	
0026₁₆	
0027₁₆	
0028₁₆	DMA0 transfer counter (TCR0)
0029₁₆	
002A₁₆	
002B₁₆	
002C₁₆	DMA0 control register (DM0CON)
002D₁₆	
002E₁₆	
002F₁₆	
0030₁₆	DMA1 source pointer (SAR1)
0031₁₆	
0032₁₆	
0033₁₆	
0034₁₆	DMA1 destination pointer (DAR1)
0035₁₆	
0036₁₆	
0037₁₆	
0038₁₆	DMA1 transfer counter (TCR1)
0039₁₆	
003A₁₆	
003B₁₆	
003C₁₆	DMA1 control register (DM1CON)
003D₁₆	
003E₁₆	
003F₁₆	

0040₁₆	
0041₁₆	
0042₁₆	
0043₁₆	
0044₁₆	INT3 interrupt control register (INT3IC)
0045₁₆	Timer B5 interrupt control register (TB5IC)
0046₁₆	Timer B4 interrupt control register (TB4IC)
0047₁₆	Timer B3 interrupt control register (TB3IC)
0048₁₆	SI/O4 interrupt control register (S4IC) INT5 interrupt control register (INT5IC)
0049₁₆	SI/O3 interrupt control register (S3IC) INT4 interrupt control register (INT4IC)
004A₁₆	Bus collision detection interrupt control register (BCNIC)
004B₁₆	DMA0 interrupt control register (DM0IC)
004C₁₆	DMA1 interrupt control register (DM1IC)
004D₁₆	Key input interrupt control register (KUPIC)
004E₁₆	A-D conversion interrupt control register (ADIC)
004F₁₆	UART2 transmit interrupt control register (S2TIC)
0050₁₆	UART2 receive interrupt control register (S2RIC)
0051₁₆	UART0 transmit interrupt control register (S0TIC)
0052₁₆	UART0 receive interrupt control register (S0RIC)
0053₁₆	UART1 transmit interrupt control register (S1TIC)
0054₁₆	UART1 receive interrupt control register (S1RIC)
0055₁₆	Timer A0 interrupt control register (TA0IC)
0056₁₆	Timer A1 interrupt control register (TA1IC)
0057₁₆	Timer A2 interrupt control register (TA2IC)
0058₁₆	Timer A3 interrupt control register (TA3IC)
0059₁₆	Timer A4 interrupt control register (TA4IC)
005A₁₆	Timer B0 interrupt control register (TB0IC)
005B₁₆	Timer B1 interrupt control register (TB1IC)
005C₁₆	Timer B2 interrupt control register (TB2IC)
005D₁₆	INT0 interrupt control register (INT0IC)
005E₁₆	INT1 interrupt control register (INT1IC)
005F₁₆	INT2 interrupt control register (INT2IC)
0060₁₆	
0061₁₆	
0062₁₆	
0063₁₆	
0064₁₆	
0065₁₆	
≈	≈
032A₁₆	
032B₁₆	
032C₁₆	
032D₁₆	
032E₁₆	
032F₁₆	
0330₁₆	
0331₁₆	
0332₁₆	
0333₁₆	
0334₁₆	
0335₁₆	
0336₁₆	
0337₁₆	
0338₁₆	
0339₁₆	
033A₁₆	
033B₁₆	
033C₁₆	
033D₁₆	
033E₁₆	
033F₁₆	

Note : Locations in the SFR area where nothing is allocated are reserved areas. Do not access these areas for read or write.

Figure 1.6.1. Location of peripheral unit control registers (1)

0340₁₆	Timer B3, 4, 5 count start flag (TBSR)
0341₁₆	
0342₁₆	Timer A1-1 register (TA11)
0343₁₆	
0344₁₆	Timer A2-1 register (TA21)
0345₁₆	
0346₁₆	Timer A4-1 register (TA41)
0347₁₆	
0348₁₆	Three-phase PWM control register 0(INVC0)
0349₁₆	Three-phase PWM control register 1(INVC1)
034A₁₆	Three-phase output buffer register 0(IDB0)
034B₁₆	Three-phase output buffer register 1(IDB1)
034C₁₆	Dead time timer(DTT)
034D₁₆	Timer B2 interrupt occurrence frequency set counter(ICTB2)
034E₁₆	
034F₁₆	
0350₁₆	Timer B3 register (TB3)
0351₁₆	
0352₁₆	Timer B4 register (TB4)
0353₁₆	
0354₁₆	Timer B5 register (TB5)
0355₁₆	
0356₁₆	
0357₁₆	
0358₁₆	
0359₁₆	
035A₁₆	
035B₁₆	Timer B3 mode register (TB3MR)
035C₁₆	Timer B4 mode register (TB4MR)
035D₁₆	Timer B5 mode register (TB5MR)
035E₁₆	
035F₁₆	Interrupt cause select register (IFSR)
0360₁₆	SI/O3 transmit/receive register (S3TRR)
0361₁₆	
0362₁₆	SI/O3 control register (S3C)
0363₁₆	SI/O3 bit rate generator (S3BRG)
0364₁₆	SI/O4 transmit/receive register (S4TRR)
0365₁₆	
0366₁₆	SI/O4 control register (S4C)
0367₁₆	SI/O4 bit rate generator (S4BRG)
0368₁₆	
0369₁₆	
036A₁₆	
036B₁₆	
036C₁₆	
036D₁₆	
036E₁₆	
036F₁₆	
0370₁₆	
0371₁₆	
0372₁₆	
0373₁₆	
0374₁₆	
0375₁₆	UART2 special mode register 3(U2SMR3)
0376₁₆	UART2 special mode register 2(U2SMR2)
0377₁₆	UART2 special mode register (U2SMR)
0378₁₆	UART2 transmit/receive mode register (U2MR)
0379₁₆	UART2 bit rate generator (U2BRG)
037A₁₆	UART2 transmit buffer register (U2TB)
037B₁₆	
037C₁₆	UART2 transmit/receive control register 0 (U2C0)
037D₁₆	UART2 transmit/receive control register 1 (U2C1)
037E₁₆	UART2 receive buffer register (U2RB)
037F₁₆	

0380₁₆	Count start flag (TABSR)
0381₁₆	Clock prescaler reset flag (CPSRF)
0382₁₆	One-shot start flag (ONSF)
0383₁₆	Trigger select register (TRGSR)
0384₁₆	Up-down flag (UDF)
0385₁₆	
0386₁₆	Timer A0 (TA0)
0387₁₆	
0388₁₆	Timer A1 (TA1)
0389₁₆	
038A₁₆	Timer A2 (TA2)
038B₁₆	
038C₁₆	Timer A3 (TA3)
038D₁₆	
038E₁₆	Timer A4 (TA4)
038F₁₆	
0390₁₆	Timer B0 (TB0)
0391₁₆	
0392₁₆	Timer B1 (TB1)
0393₁₆	
0394₁₆	Timer B2 (TB2)
0395₁₆	
0396₁₆	Timer A0 mode register (TA0MR)
0397₁₆	Timer A1 mode register (TA1MR)
0398₁₆	Timer A2 mode register (TA2MR)
0399₁₆	Timer A3 mode register (TA3MR)
039A₁₆	Timer A4 mode register (TA4MR)
039B₁₆	Timer B0 mode register (TB0MR)
039C₁₆	Timer B1 mode register (TB1MR)
039D₁₆	Timer B2 mode register (TB2MR)
039E₁₆	
039F₁₆	
03A0₁₆	UART0 transmit/receive mode register (U0MR)
03A1₁₆	UART0 bit rate generator (U0BRG)
03A2₁₆	UART0 transmit buffer register (U0TB)
03A3₁₆	
03A4₁₆	UART0 transmit/receive control register 0 (U0C0)
03A5₁₆	UART0 transmit/receive control register 1 (U0C1)
03A6₁₆	UART0 receive buffer register (U0RB)
03A7₁₆	
03A8₁₆	UART1 transmit/receive mode register (U1MR)
03A9₁₆	UART1 bit rate generator (U1BRG)
03AA₁₆	UART1 transmit buffer register (U1TB)
03AB₁₆	
03AC₁₆	UART1 transmit/receive control register 0 (U1C0)
03AD₁₆	UART1 transmit/receive control register 1 (U1C1)
03AE₁₆	UART1 receive buffer register (U1RB)
03AF₁₆	
03B0₁₆	UART transmit/receive control register 2 (UCON)
03B1₁₆	
03B2₁₆	
03B3₁₆	
03B4₁₆	
03B5₁₆	
03B6₁₆	Flash memory control register 1 (FMR1) (Note1)
03B7₁₆	Flash memory control register 0 (FMR0) (Note1)
03B8₁₆	DMA0 request cause select register (DM0SL)
03B9₁₆	
03BA₁₆	DMA1 request cause select register (DM1SL)
03BB₁₆	
03BC₁₆	CRC data register (CRCD)
03BD₁₆	
03BE₁₆	CRC input register (CRCIN)
03BF₁₆	

Note 1: This register is only exist in flash memory version.
Note 2: Locations in the SFR area where nothing is allocated are reserved areas. Do not access these areas for read or write.

Figure 1.6.2. Location of peripheral unit control registers (2)

03C0₁₆ 03C1₁₆	A-D register 0 (AD0)
03C2₁₆ 03C3₁₆	A-D register 1 (AD1)
03C4₁₆ 03C5₁₆	A-D register 2 (AD2)
03C6₁₆ 03C7₁₆	A-D register 3 (AD3)
03C8₁₆ 03C9₁₆	A-D register 4 (AD4)
03CA₁₆ 03CB₁₆	A-D register 5 (AD5)
03CC₁₆ 03CD₁₆	A-D register 6 (AD6)
03CE₁₆ 03CF₁₆	A-D register 7 (AD7)
03D0₁₆	
03D1₁₆	
03D2₁₆	
03D3₁₆	
03D4₁₆	A-D control register 2 (ADCON2)
03D5₁₆	
03D6₁₆	A-D control register 0 (ADCON0)
03D7₁₆	A-D control register 1 (ADCON1)
03D8₁₆	D-A register 0 (DA0)
03D9₁₆	
03DA₁₆	D-A register 1 (DA1)
03DB₁₆	
03DC₁₆	D-A control register (DACON)
03DD₁₆	
03DE₁₆	
03DF₁₆	
03E0₁₆	Port P0 (P0)
03E1₁₆	Port P1 (P1)
03E2₁₆	Port P0 direction register (PD0)
03E3₁₆	Port P1 direction register (PD1)
03E4₁₆	Port P2 (P2)
03E5₁₆	Port P3 (P3)
03E6₁₆	Port P2 direction register (PD2)
03E7₁₆	Port P3 direction register (PD3)
03E8₁₆	Port P4 (P4)
03E9₁₆	Port P5 (P5)
03EA₁₆	Port P4 direction register (PD4)
03EB₁₆	Port P5 direction register (PD5)
03EC₁₆	Port P6 (P6)
03ED₁₆	Port P7 (P7)
03EE₁₆	Port P6 direction register (PD6)
03EF₁₆	Port P7 direction register (PD7)
03F0₁₆	Port P8 (P8)
03F1₁₆	Port P9 (P9)
03F2₁₆	Port P8 direction register (PD8)
03F3₁₆	Port P9 direction register (PD9)
03F4₁₆	Port P10 (P10)
03F5₁₆	
03F6₁₆	Port P10 direction register (PD10)
03F7₁₆	
03F8₁₆	
03F9₁₆	
03FA₁₆	
03FB₁₆	
03FC₁₆	Pull-up control register 0 (PUR0)
03FD₁₆	Pull-up control register 1 (PUR1)
03FE₁₆	Pull-up control register 2 (PUR2)
03FF₁₆	Port control register (PCR)

Note : Locations in the SFR area where nothing is allocated are reserved areas. Do not access these areas for read or write.

Figure 1.6.3. Location of peripheral unit control registers (3)

Software Reset

Writing “1” to bit 3 of the processor mode register 0 (address 0004_{16}) applies a (software) reset to the microcomputer. A software reset has the same effect as a hardware reset. The contents of internal RAM are preserved.

Processor Mode

(1) Types of Processor Mode

One of three processor modes can be selected: single-chip mode, memory expansion mode, and microprocessor mode. The functions of some pins, the memory map, and the access space differ according to the selected processor mode.

- **Single-chip mode**

  In single-chip mode, only internal memory space (SFR, internal RAM, and internal ROM) can be accessed. However, after the reset has been released and the operation of shifting from the microprocessor mode has started (“H” applied to the CNVSS pin), the internal ROM area cannot be accessed even if the CPU shifts to the single-chip mode.

  Ports P0 to P10 can be used as programmable I/O ports or as I/O ports for the internal peripheral functions.

- **Memory expansion mode**

  In memory expansion mode, external memory can be accessed in addition to the internal memory space (SFR, internal RAM, and internal ROM). However, after the reset has been released and the operation of shifting from the microprocessor mode has started (“H” applied to the CNVSS pin), the internal ROM area cannot be accessed even if the CPU shifts to the memory expansion mode.

  In this mode, some of the pins function as the address bus, the data bus, and as control signals. The number of pins assigned to these functions depends on the bus and register settings. (See “Bus Settings” for details.)

- **Microprocessor mode**

  In microprocessor mode, the SFR, internal RAM, and external memory space can be accessed. The internal ROM area cannot be accessed.

  In this mode, some of the pins function as the address bus, the data bus, and as control signals. The number of pins assigned to these functions depends on the bus width and register settings. (See “Bus Settings” for details.)

(2) Setting Processor Modes

The processor mode is set using the CNVSS pin and the processor mode bits (bits 1 and 0 at address 0004_{16}). Do not set the processor mode bits to “10_2”.

Regardless of the level of the CNVSS pin, changing the processor mode bits selects the mode. Therefore, never change the processor mode bits when changing the contents of other bits. Do not change the processor mode bits simultaneously with other bits when changing the processor mode bits “01_2” or “11_2”. Change the processor mode bits after changing the other bits. Also do not attempt to shift to or from the microprocessor mode within the program stored in the internal ROM area.

- **Applying Vss to CNVss pin**

  The microcomputer begins operation in single-chip mode after being reset. Memory expansion mode is selected by writing “01_2” to the processor mode bits.

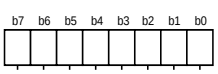
- **Applying Vcc to CNVss pin**

  The microcomputer starts to operate in microprocessor mode after being reset.

Figure 1.7.1 shows the processor mode register 0 and 1.

Figure 1.7.2 shows the memory maps applicable for each of the modes.

Processor mode register 0 (Note 1)

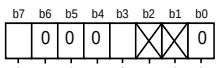


Symbol: PM0 | Address: 0004₁₆ | When reset: 00₁₆ (Note 2)

Bit symbol	Bit name	Function	R	W
PM00	Processor mode bit	b1 b0 0 0: Single-chip mode 0 1: Memory expansion mode 1 0: Do not set 1 1: Microprocessor mode	O	O
PM01			O	O
PM02	R/W mode select bit	0 : $\overline{RD}$,$\overline{BHE}$,$\overline{WR}$ 1 : RD,WRH,WRL	O	O
PM03	Software reset bit	The device is reset when this bit is set to "1". The value of this bit is "0" when read.	O	O
PM04	Multiplexed bus space select bit	b5 b4 0 0 : Multiplexed bus is not used 0 1 : Allocated to CS2 space 1 0 : Allocated to CS1 space 1 1 : Allocated to entire space (Note4)	O	O
PM05			O	O
PM06	Port P4₀ to P4₃ function select bit (Note 3)	0 : Address output 1 : Port function (Address is not output)	O	O
PM07	BCLK output disable bit	0 : BCLK is output 1 : BCLK is not output (Pin is left floating)	O	O

Note 1: Set bit 1 of the protect register (address 000A₁₆) to "1" when writing new values to this register.
Note 2: If the Vcc voltage is applied to the CNVss, the value of this register when reset is 03₁₆. (PM00 and PM01 both are set to "1".)
Note 3: Valid in microprocessor and memory expansion modes.
Note 4: If the entire space is of multiplexed bus in memory expansion mode, choose an 8-bit width.The processor operates using the separate bus after reset is revoked, so the entire space multiplexed bus cannot be chosen in microprocessor mode.
P3₁ to P3₇ become a port if the entire space multiplexed bus is chosen, so only 256 bytes can be used in each chip select.

Processor mode register 1 (Note 1)

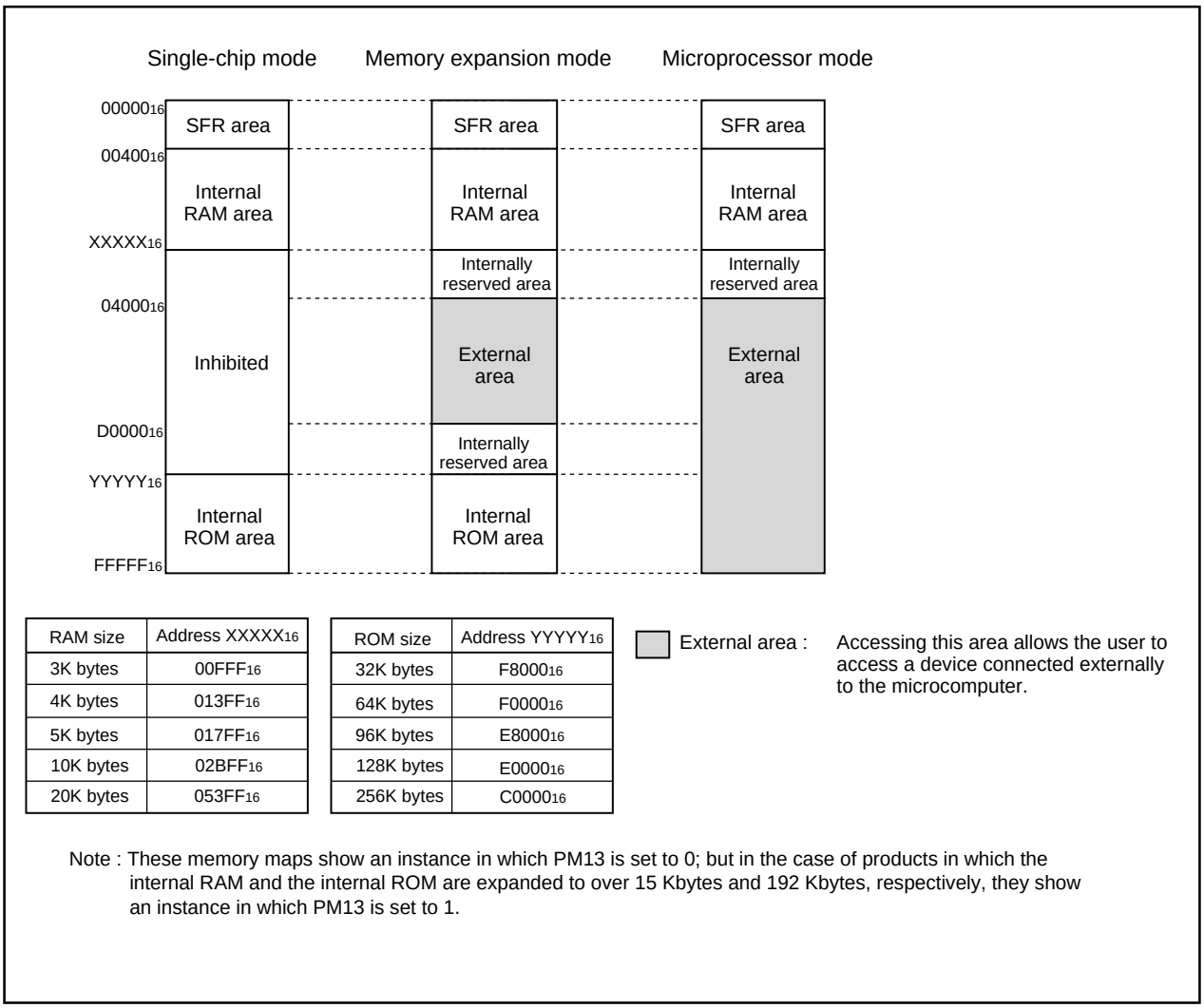


Symbol: PM1 | Address: 0005₁₆ | When reset: 00000XX0₂

Bit symbol	Bit name	Function	R	W
Reserved bit		Must always be set to "0"	O	O
Nothing is assigned. In an attempt to write to these bits, write "0". The value, if read, turns out to be indeterminate.			—	—
PM13	Internal reserved area expansion bit (Note 2)	0: The internal RAM area is 15 kbytes or less and the internal ROM area is 192 kbytes or less 1: Expands the internal RAM area and internal ROM area to over 15 kbytes and to over 192 kbytes respectively. (Note 2)	O	O
Reserved bit		Must always be set to "0"	O	O
Reserved bit		Must always be set to "0"	O	O
Reserved bit		Must always be set to "0"	O	O
PM17	Wait bit	0 : No wait state 1 : Wait state inserted	O	O

Note 1: Set bit 1 of the protect register (address 000A₁₆) to "1" when writing new values to this register.
Note 2: When the reset is revoked, this bit is set to "0". To expand the internal area, set this bit to "1" in user program. And the top of user program must be allocated to D0000₁₆ or subsequent address.

Figure 1.7.1. Processor mode register 0 and 1



RAM size	Address XXXXX$_{16}$
3K bytes	00FFF$_{16}$
4K bytes	013FF$_{16}$
5K bytes	017FF$_{16}$
10K bytes	02BFF$_{16}$
20K bytes	053FF$_{16}$

ROM size	Address YYYYY$_{16}$
32K bytes	F8000$_{16}$
64K bytes	F0000$_{16}$
96K bytes	E8000$_{16}$
128K bytes	E0000$_{16}$
256K bytes	C0000$_{16}$

Note : These memory maps show an instance in which PM13 is set to 0; but in the case of products in which the internal RAM and the internal ROM are expanded to over 15 Kbytes and 192 Kbytes, respectively, they show an instance in which PM13 is set to 1.

Figure 1.7.2. Memory maps in each processor mode (without memory area expansion, normal mode)

Internal Reserved Area Expansion Bit (PM13)

This bit expands the internal RAM area and the internal ROM area, and changes the chip select area. In M30624MGA/FGA, for example, to set this bit to “1” expands the internal RAM area and the internal ROM area to 20 Kbytes and 256 Kbytes respectively. Refer to Figure 1.7.3 for the chip select area. When the reset is revoked, this bit is set to “0”. To expand the internal area, set this bit to “1” in user program. And the top of user program must be allocated to D0000$_{16}$ or subsequent address.

In the case of the product in which the internal ROM is 192 Kbytes or less and the internal RAM is 15 Kbytes or less, set this bit to “0” when this product is used in the memory expansion mode or the microprocessor mode. When the product is used in the single chip mode, the internal area is not expanded and any action is not affected, even if this bit is set to “1”.

Figure 1.7.3 shows the memory maps and the chip selection areas effected by PM13 (the internal reserved area expansion bit) in each processor mode for the product having an internal RAM of more than 15K bytes and a ROM of more than 192K bytes.

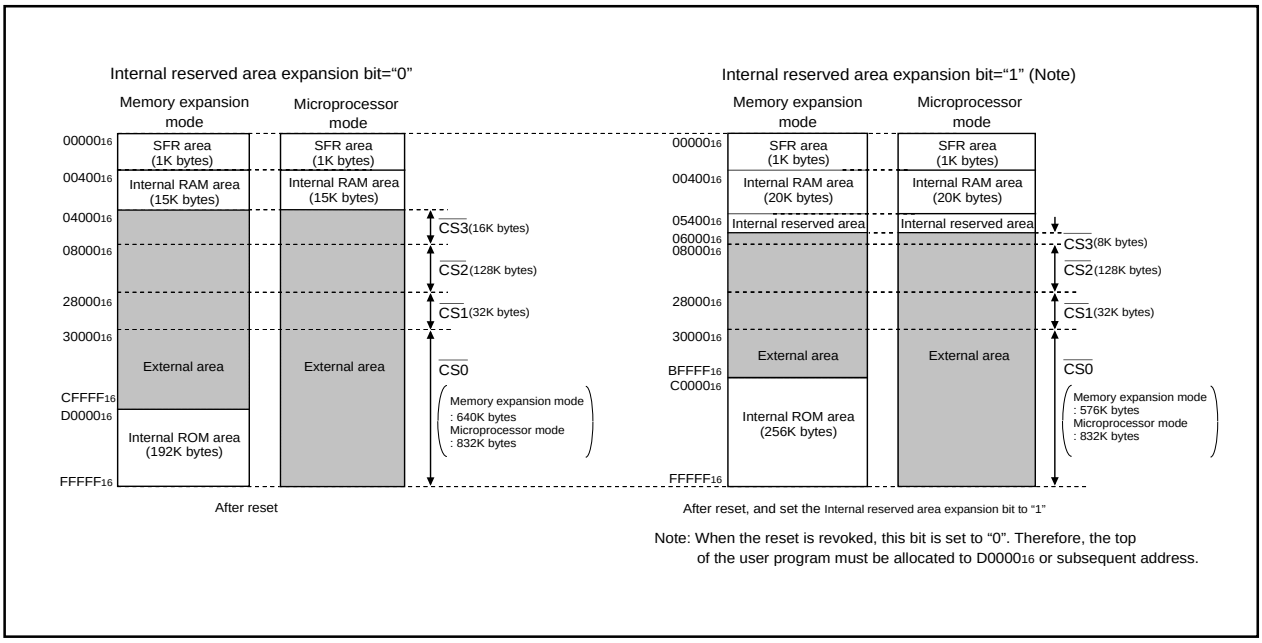


Figure 1.7.3. Memory location and chip select area in each processor mode

Bus Settings

The BYTE pin and bits 4 to 6 of the processor mode register 0 (address 0004_{16}) are used to change the bus settings. Table 1.8.1 shows the factors used to change the bus settings.

Table 1.8.1. Factors for switching bus settings

Bus setting	Switching factor
Switching external address bus width	Bit 6 of processor mode register 0
Switching external data bus width	BYTE pin
Switching between separate and multiplex bus	Bits 4 and 5 of processor mode register 0

(1) Selecting external address bus width

The address bus width for external output in the 1M bytes of address space can be set to 16 bits (64K bytes address space) or 20 bits (1M bytes address space). When bit 6 of the processor mode register 0 is set to “1”, the external address bus width is set to 16 bits, and P2 and P3 become part of the address bus. $P4_0$ to $P4_3$ can be used as programmable I/O ports. When bit 6 of processor mode register 0 is set to “0”, the external address bus width is set to 20 bits, and P2, P3, and $P4_0$ to $P4_3$ become part of the address bus.

(2) Selecting external data bus width

The external data bus width can be set to 8 or 16 bits. (Note, however, that only the separate bus can be set.) When the BYTE pin is “L”, the bus width is set to 16 bits; when “H”, it is set to 8 bits. (The internal bus width is permanently set to 16 bits.) While operating, fix the BYTE pin either to “H” or to “L”.

(3) Selecting separate/multiplex bus

The bus format can be set to multiplex or separate bus using bits 4 and 5 of the processor mode register 0.

- **Separate bus**

  In this mode, the data and address are input and output separately. The data bus can be set using the BYTE pin to be 8 or 16 bits. When the BYTE pin is “H”, the data bus is set to 8 bits and P0 functions as the data bus and P1 as a programmable I/O port. When the BYTE pin is “L”, the data bus is set to 16 bits and P0 and P1 are both used for the data bus.

  When the separate bus is used for access, a software wait can be selected.

- **Multiplex bus**

  In this mode, data and address I/O are time multiplexed. With the BYTE pin = “H”, the 8 bits from D_0 to D_7 are multiplexed with A_0 to A_7.

  With the BYTE pin = “L”, the 8 bits from D_0 to D_7 are multiplexed with A_1 to A_8. D_8 to D_{15} are not multiplexed. In this case, the external devices connected to the multiplexed bus are mapped to the microcomputer’s even addresses (every 2nd address). To access these external devices, access the even addresses as bytes.

  The ALE signal latches the address. It is output from $P5_6$.

  Before using the multiplex bus for access, be sure to insert a software wait.

  If the entire space is of multiplexed bus in memory expansion mode, choose an 8-bit width.

  The processor operates using the separate bus after reset is revoked, so the entire space multiplexed bus cannot be chosen in microprocessor mode.

  $P3_1$ to $P3_7$ become a port if the entire space multiplexed bus is chosen, so only 256 bytes can be used in each chip select.

Table 1.8.2. Pin functions for each processor mode

Processor mode	Single-chip mode	Memory expansion mode/microprocessor modes				Memory expansion mode
Multiplexed bus space select bit		"01", "10" Either CS1 or CS2 is for multiplexed bus and others are for separate bus		"00" (separate bus)		"11" (Note 1) multiplexed bus for the entire space
Data bus width BYTE pin level		8 bits "H"	16 bits "L"	8 bits "H"	16 bits "L"	8 bit "H"
$P0_0$ to $P0_7$	I/O port	Data bus	Data bus	Data bus	Data bus	I/O port
$P1_0$ to $P1_7$	I/O port	I/O port	Data bus	I/O port	Data bus	I/O port
$P2_0$	I/O port	Address bus /data bus(Note 2)	Address bus	Address bus	Address bus	Address bus /data bus
$P2_1$ to $P2_7$	I/O port	Address bus /data bus(Note 2)	Address bus /data bus(Note 2)	Address bus	Address bus	Address bus /data bus
$P3_0$	I/O port	Address bus	Address bus /data bus(Note 2)	Address bus	Address bus	A_8/D_7
$P3_1$ to $P3_7$	I/O port	Address bus	Address bus	Address bus	Address bus	I/O port
$P4_0$ to $P4_3$ Port $P4_0$ to $P4_3$ function select bit = 1	I/O port	I/O port	I/O port	I/O port	I/O port	I/O port
$P4_0$ to $P4_3$ Port $P4_0$ to $P4_3$ function select bit = 0	I/O port	Address bus	Address bus	Address bus	Address bus	I/O port
$P4_4$ to $P4_7$	I/O port	$\overline{CS}$ (chip select) or programmable I/O port (For details, refer to "Bus control")				
$P5_0$ to $P5_3$	I/O port	Outputs $\overline{RD}$, $\overline{WRL}$, $\overline{WRH}$, and BCLK or $\overline{RD}$, $\overline{BHE}$, $\overline{WR}$, and BCLK (For details, refer to "Bus control")				
$P5_4$	I/O port	$\overline{HLDA}$	$\overline{HLDA}$	$\overline{HLDA}$	$\overline{HLDA}$	$\overline{HLDA}$
$P5_5$	I/O port	$\overline{HOLD}$	$\overline{HOLD}$	$\overline{HOLD}$	$\overline{HOLD}$	$\overline{HOLD}$
$P5_6$	I/O port	ALE	ALE	ALE	ALE	ALE
$P5_7$	I/O port	$\overline{RDY}$	$\overline{RDY}$	$\overline{RDY}$	$\overline{RDY}$	$\overline{RDY}$

Note 1: If the entire space is of multiplexed bus in memory expansion mode, choose an 8-bit width.
The processor operates using the separate bus after reset is revoked, so the entire space multiplexed bus cannot be chosen in microprocessor mode.
$P3_1$ to $P3_7$ become a port if the entire space multiplexed bus is chosen, so only 256 bytes can be used in each chip select.
Note 2: Address bus when in separate bus mode.

Bus Control

The following explains the signals required for accessing external devices and software waits. The signals required for accessing the external devices are valid when the processor mode is set to memory expansion mode and microprocessor mode. The software waits are valid in all processor modes.

(1) Address bus/data bus

The address bus consists of the 20 pins A_0 to A_{19} for accessing the 1M bytes of address space.

The data bus consists of the pins for data I/O. When the BYTE pin is “H”, the 8 ports D_0 to D_7 function as the data bus. When BYTE is “L”, the 16 ports D_0 to D_{15} function as the data bus.

When a change is made from single-chip mode to memory expansion mode, the value of the address bus is undefined until external memory is accessed.

(2) Chip select signal

The chip select signal is output using the same pins as $P4_4$ to $P4_7$. Bits 0 to 3 of the chip select control register (address 0008_{16}) set each pin to function as a port or to output the chip select signal. The chip select control register is valid in memory expansion mode and microprocessor mode. In single-chip mode, $P4_4$ to $P4_7$ function as programmable I/O ports regardless of the value in the chip select control register.

In microprocessor mode, only $\overline{CS0}$ outputs the chip select signal after the reset state has been cancelled. $\overline{CS1}$ to $\overline{CS3}$ function as input ports. Figure 1.9.1 shows the chip select control register.

The chip select signal can be used to split the external area into as many as four blocks. Tables 1.9.1 and 1.9.2 show the external memory areas specified using the chip select signal.

Table 1.9.1. External areas specified by the chip select signals

(A product having an internal RAM equal to or less than 15K bytes and a ROM equal to or less than 192K bytes)(Note)

Processor mode	Chip select signal			
	$\overline{CS0}$	$\overline{CS1}$	$\overline{CS2}$	$\overline{CS3}$
Memory expansion mode	30000_{16} to $CFFFF_{16}$ (640K bytes)	28000_{16} to $2FFFF_{16}$ (32K bytes)	08000_{16} to $27FFF_{16}$ (128K bytes)	04000_{16} to $07FFF_{16}$ (16K bytes)
Microprocessor mode	30000_{16} to $FFFFF_{16}$ (832K bytes)			

Note :Be sure to set bit 3 (PM13) of processor mode register 1 to “0”.

Table 1.9.2. External areas specified by the chip select signals

(A product having an internal RAM of more than 15K bytes and a ROM of more than 192K bytes)

Processor mode	Chip select signal			
	$\overline{CS0}$	$\overline{CS1}$	$\overline{CS2}$	$\overline{CS3}$
Memory expansion mode	When PM13=0 30000_{16} to $CFFFF_{16}$ (640K bytes)	28000_{16} to $2FFFF_{16}$ (32K bytes)	08000_{16} to $27FFF_{16}$ (128K bytes)	When PM13=0 04000_{16} to $07FFF_{16}$ (16K bytes)
	When PM13=1 30000_{16} to $BFFFF_{16}$ (576K bytes)			When PM13=1 06000_{16} to $07FFF_{16}$ (8K bytes)
Microprocessor mode	03000_{16} to $FFFFF_{16}$ (816K bytes)			

Chip select control register

b7	b6	b5	b4	b3	b2	b1	b0

Symbol: CSR — Address: 0008_{16} — When reset: 01_{16}

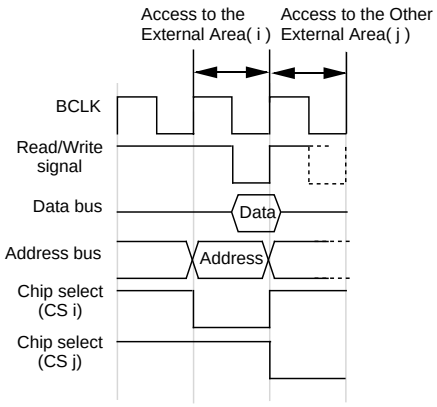
Bit symbol	Bit name	Function	R	W
CS0	$\overline{CS0}$ output enable bit	0 : Chip select output disabled (Normal port pin) 1 : Chip select output enabled	○	○
CS1	$\overline{CS1}$ output enable bit		○	○
CS2	$\overline{CS2}$ output enable bit		○	○
CS3	$\overline{CS3}$ output enable bit		○	○
CS0W	$\overline{CS0}$ wait bit	0 : Wait state inserted 1 : No wait state	○	○
CS1W	$\overline{CS1}$ wait bit		○	○
CS2W	$\overline{CS2}$ wait bit		○	○
CS3W	$\overline{CS3}$ wait bit		○	○

Figure 1.9.1. Chip select control register

The timing of the chip select signal changing to “L”(active) is synchronized with the address bus. But the timing of the chip select signal changing to “H” depends on the area which will be accessed in the next cycle. Figure 1.9.2 shows the output example of the address bus and chip select signal.

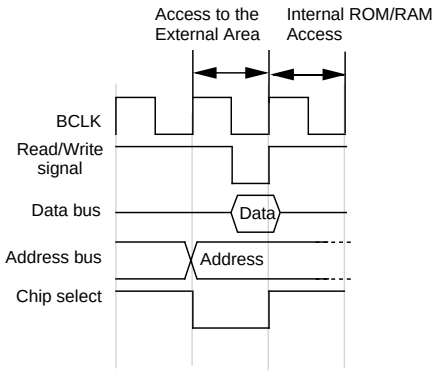
Example 1) After access the external area, both the address signal and the chip select signal change concurrently in the next cycle.

In this example, after access to the external area(i), an access to the area indicated by the other chip select signal(j) will occur in the next cycle. In this case, both the address bus and the chip select signal change between the two cycles.



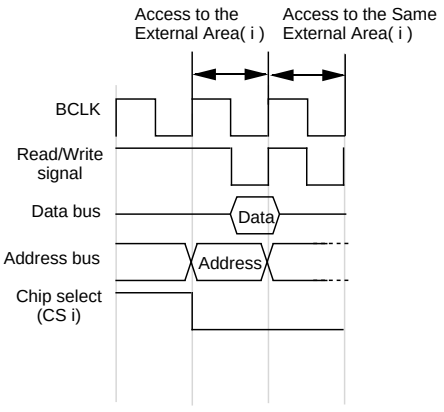
Example 2) After access the external area, only the chip select signal changes in the next cycle (the address bus does not change).

In this example, an access to the internal ROM or the internal RAM in the next cycle will occur, after access to the external area. In this case, the chip select signal changes between the two cycles, but the address does not change.



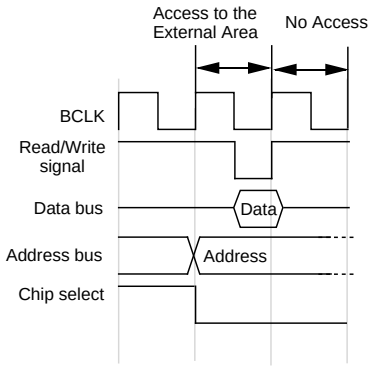
Example 3) After access the external area, only the address bus changes in the next cycle (the chip select signal does not change).

In this example, after access to the external area(i), an access to the area indicated by the same chip select signal(i) will occur in the next cycle. In this case, the address bus changes between the two cycles, but the chip select signal does not change.



Example 4) After access the external area, either the address signal and the chip select signal do not change in the next cycle.

In this example, any access to any area does not occur in the next cycle (either instruction prefetch does not occur). In this case,either the address bus and chip select signal do not change between the two cycles.



Note : These examples show the address bus and chip select signal within the successive two cycles.
According to the combination of these examples, the chip select can be elongated to over 2cycles.

Figure 1.9.2. Output Examples about Address Bus and Chip Select Signal (Separated Bus without Wait)

(3) Read/write signals

With a 16-bit data bus (BYTE pin ="L"), bit 2 of the processor mode register 0 (address 0004_{16}) select the combinations of $\overline{RD}$, $\overline{BHE}$, and $\overline{WR}$ signals or $\overline{RD}$, $\overline{WRL}$, and $\overline{WRH}$ signals. With an 8-bit data bus (BYTE pin = "H"), use the combination of $\overline{RD}$, $\overline{WR}$, and $\overline{BHE}$ signals. (Set bit 2 of the processor mode register 0 (address 0004_{16}) to "0".) Tables 1.9.3 and 1.9.4 show the operation of these signals.
After a reset has been cancelled, the combination of $\overline{RD}$, $\overline{WR}$, and $\overline{BHE}$ signals is automatically selected. When switching to the $\overline{RD}$, $\overline{WRL}$, and $\overline{WRH}$ combination, do not write to external memory until bit 2 of the processor mode register 0 (address 0004_{16}) has been set (Note).

Note: Before attempting to change the contents of the processor mode register 0, set bit 1 of the protect register (address $000A_{16}$) to "1".

Table 1.9.3. Operation of $\overline{RD}$, $\overline{WRL}$, and $\overline{WRH}$ signals

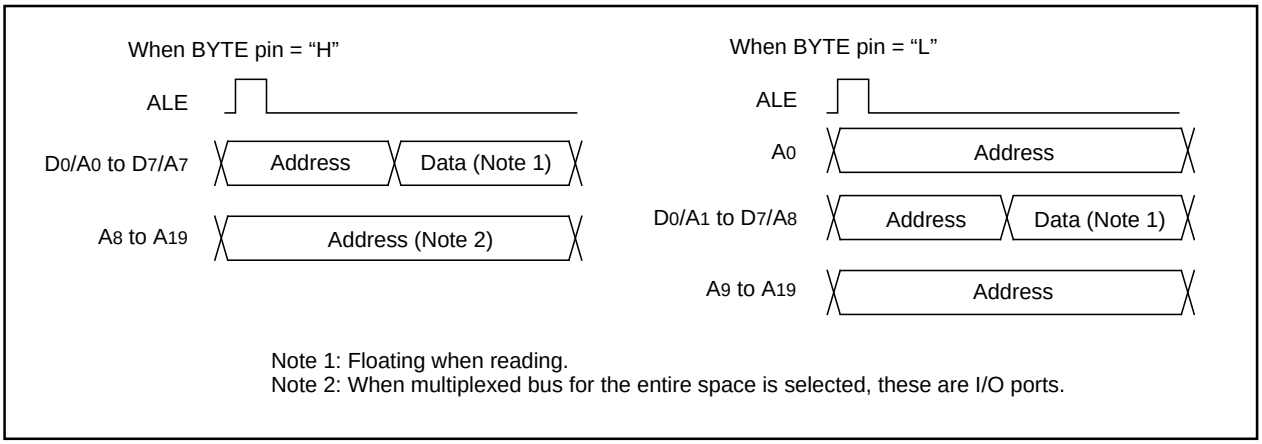
Data bus width	$\overline{RD}$	$\overline{WRL}$	$\overline{WRH}$	Status of external data bus
16-bit (BYTE = "L")	L	H	H	Read data
	H	L	H	Write 1 byte of data to even address
	H	H	L	Write 1 byte of data to odd address
	H	L	L	Write data to both even and odd addresses

Table 1.9.4. Operation of $\overline{RD}$, $\overline{WR}$, and $\overline{BHE}$ signals

Data bus width	$\overline{RD}$	$\overline{WR}$	$\overline{BHE}$	A0	Status of external data bus
16-bit (BYTE = "L")	H	L	L	H	Write 1 byte of data to odd address
	L	H	L	H	Read 1 byte of data from odd address
	H	L	H	L	Write 1 byte of data to even address
	L	H	H	L	Read 1 byte of 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-bit (BYTE = "H")	H	L	Not used	H / L	Write 1 byte of data
	L	H	Not used	H / L	Read 1 byte of data

(4) ALE signal

The ALE signal latches the address when accessing the multiplex bus space. Latch the address when the ALE signal falls.



Note 1: Floating when reading.
Note 2: When multiplexed bus for the entire space is selected, these are I/O ports.

Figure 1.9.3. ALE signal and address/data bus

(5) The $\overline{RDY}$ signal

$\overline{RDY}$ is a signal that facilitates access to an external device that requires long access time. As shown in Figure 1.9.4, if an “L” is being input to the $\overline{RDY}$ at the BCLK falling edge, the bus turns to the wait state. If an “H” is being input to the $\overline{RDY}$ pin at the BCLK falling edge, the bus cancels the wait state. Table 1.9.5 shows the state of the microcomputer with the bus in the wait state, and Figure 1.9.4 shows an example in which the $\overline{RD}$ signal is prolonged by the $\overline{RDY}$ signal.

The $\overline{RDY}$ signal is valid when accessing the external area during the bus cycle in which bits 4 to 7 of the chip select control register (address 0008_{16}) are set to “0”. The $\overline{RDY}$ signal is invalid when setting “1” to all bits 4 to 7 of the chip select control register (address 0008_{16}), but the $\overline{RDY}$ pin should be treated as properly as in non-using.

Table 1.9.5. Microcomputer status in wait state (Note)

Item	Status
Oscillation	On
R/$\overline{W}$ signal, address bus, data bus, $\overline{CS}$ ALE signal, $\overline{HLDA}$, programmable I/O ports	Maintain status when $\overline{RDY}$ signal received
Internal peripheral circuits	On

Note: The $\overline{RDY}$ signal cannot be received immediately prior to a software wait.

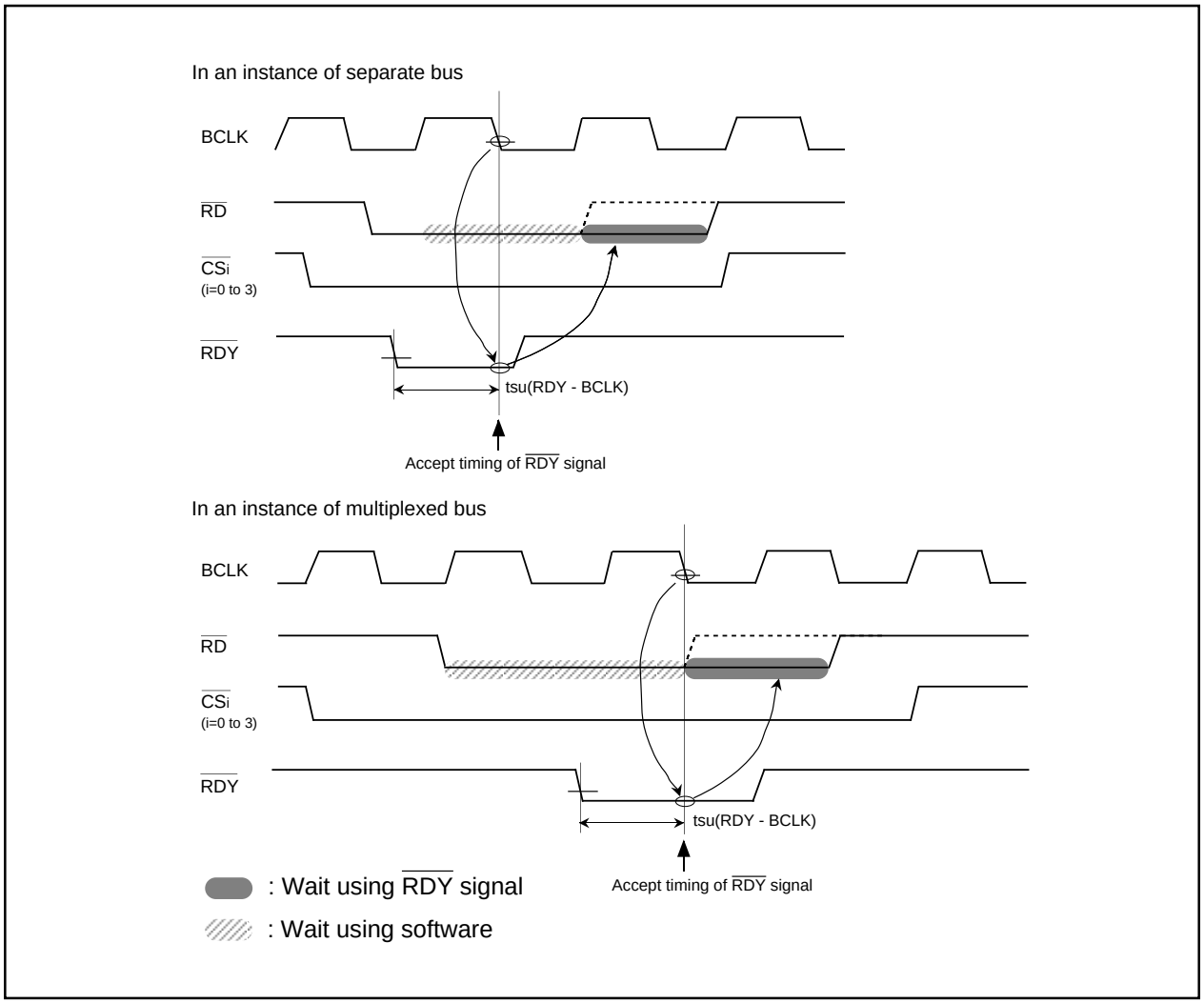


Figure 1.9.4. Example of $\overline{RD}$ signal extended by $\overline{RDY}$ signal

(6) Hold signal

The hold signal is used to transfer the bus privileges from the CPU to the external circuits. Inputting “L” to the $\overline{\text{HOLD}}$ pin places the microcomputer in the hold state at the end of the current bus access. This status is maintained and “L” is output from the $\overline{\text{HLDA}}$ pin as long as “L” is input to the $\overline{\text{HOLD}}$ pin. Table 1.9.6 shows the microcomputer status in the hold state.

Bus-using priorities are given to $\overline{\text{HOLD}}$, DMAC, and CPU in order of decreasing precedence.

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

Figure 1.9.5. Bus-using priorities

Table 1.9.6. Microcomputer status in hold state

Item		Status
Oscillation		ON
R/$\overline{\text{W}}$ signal, address bus, data bus, $\overline{\text{CS}}$, $\overline{\text{BHE}}$		Floating
Programmable I/O ports	P0, P1, P2, P3, P4, P5	Floating
	P6, P7, P8, P9, P10	Maintains status when hold signal is received
$\overline{\text{HLDA}}$		Output “L”
Internal peripheral circuits		ON (but watchdog timer stops)
ALE signal		Undefined

(7) External bus status when the internal area is accessed

Table 1.9.7 shows the external bus status when the internal area is accessed.

Table 1.9.7. External bus status when the internal area is accessed

Item		SFR accessed	Internal ROM/RAM accessed
Address bus		Address output	Maintain status before accessed address of external area
Data bus	When read	Floating	Floating
	When write	Output data	Undefined
$\overline{\text{RD}}$, $\overline{\text{WR}}$, $\overline{\text{WRL}}$, $\overline{\text{WRH}}$		$\overline{\text{RD}}$, $\overline{\text{WR}}$, $\overline{\text{WRL}}$, $\overline{\text{WRH}}$ output	Output “H”
$\overline{\text{BHE}}$		$\overline{\text{BHE}}$ output	Maintain status before accessed status of external area
$\overline{\text{CS}}$		Output “H”	Output “H”
ALE		Output “L”	Output “L”

(8) BCLK output

The user can choose the BCLK output by use of bit 7 of processor mode register 0 (0004_{16}) (Note). When set to “1”, the output floating.

Note: Before attempting to change the contents of the processor mode register 0, set bit 1 of the protect register (address $000A_{16}$) to “1”.

(9) Software wait

A software wait can be inserted by setting the wait bit (bit 7) of the processor mode register 1 (address 0005_{16}) (Note) and bits 4 to 7 of the chip select control register (address 0008_{16}).

A software wait is inserted in the internal ROM/RAM area and in the external memory area by setting the wait bit of the processor mode register 1. When set to “0”, each bus cycle is executed in one BCLK cycle. When set to “1”, each bus cycle is executed in two or three BCLK cycles. After the microcomputer has been reset, this bit defaults to “0”. When set to “1”, a wait is applied to all memory areas (two or three BCLK cycles), regardless of the contents of bits 4 to 7 of the chip select control register. Set this bit after referring to the recommended operating conditions (main clock input oscillation frequency) of the electric characteristics. However, when the user is using the $\overline{RDY}$ signal, the relevant bit in the chip select control register’s bits 4 to 7 must be set to “0”.

When the wait bit of the processor mode register 1 is “0”, software waits can be set independently for each of the 4 areas selected using the chip select signal. Bits 4 to 7 of the chip select control register correspond to chip selects $\overline{CS0}$ to $\overline{CS3}$. When one of these bits is set to “1”, the bus cycle is executed in one BCLK cycle. When set to “0”, the bus cycle is executed in two or three BCLK cycles. These bits default to “0” after the microcomputer has been reset.

The SFR area is always accessed in two BCLK cycles regardless of the setting of these control bits. Also, insert a software wait if using the multiplex bus to access the external memory area.

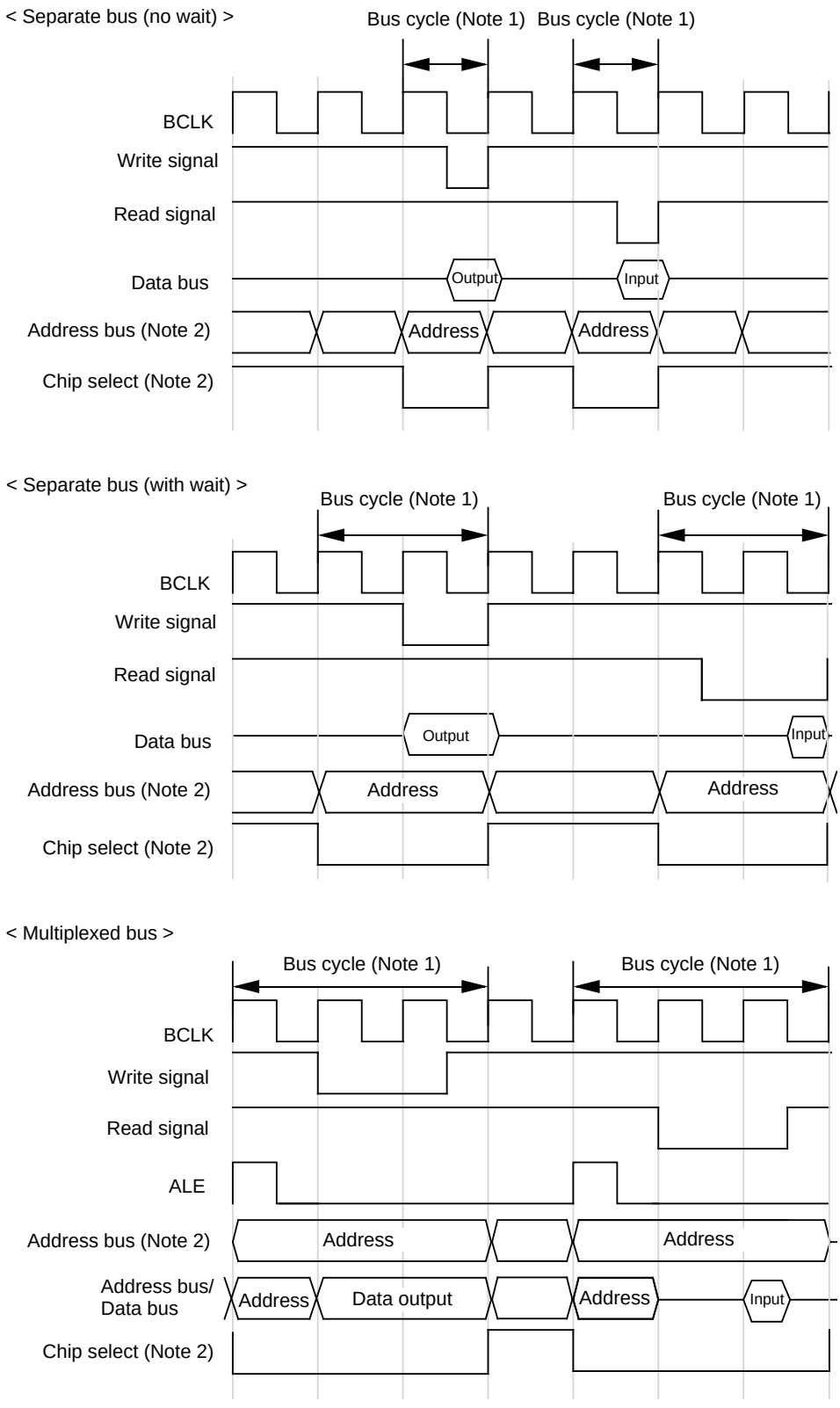
Table 1.9.8 shows the software wait and bus cycles. Figure 1.9.6 shows example of bus timing when using software waits.

Note: Before attempting to change the contents of the processor mode register 1, set bit 1 of the protect register (address $000A_{16}$) to “1”.

Table 1.9.8. Software waits and bus cycles

Area	Bus status	Wait bit	Bits 4 to 7 of chip select control register	Bus cycle
SFR	———	Invalid	Invalid	2 BCLK cycles
Internal ROM/RAM	———	0	Invalid	1 BCLK cycle
	———	1	Invalid	2 BCLK cycles
External memory area	Separate bus	0	1	1 BCLK cycle
	Separate bus	0	0	2 BCLK cycles
	Separate bus	1	0 (Note)	2 BCLK cycles
	Multiplex bus	0	0	3 BCLK cycles
	Multiplex bus	1	0 (Note)	3 BCLK cycles

Note: When using the $\overline{RDY}$ signal, always set to “0”.



Note 1: These example timing charts indicate bus cycle length.
After this bus cycle sometimes come read and write cycles in succession.
Note 2: The address bus and chip select may be extended depending on the CPU status such as that of the instruction queue buffer.

Figure 1.9.6. Typical bus timings using software wait

Clock Generating Circuit

The clock generating circuit contains two oscillator circuits that supply the operating clock sources to the CPU and internal peripheral units.

Table 1.10.1. Main clock and sub-clock generating circuits

	Main clock generating circuit	Sub-clock generating circuit
Use of clock	• CPU's operating clock source • Internal peripheral units' operating clock source	• CPU's operating clock source • Timer A/B's count clock source
Usable oscillator	Ceramic or crystal oscillator	Crystal oscillator
Pins to connect oscillator	XIN, XOUT	XCIN, XCOUT
Oscillation stop/restart function	Available	Available
Oscillator status immediately after reset	Oscillating	Stopped
Other	Externally derived clock can be input	

Example of oscillator circuit

Figure 1.10.1 shows some examples of the main clock circuit, one using an oscillator connected to the circuit, and the other one using an externally derived clock for input. Figure 1.10.2 shows some examples of sub-clock circuits, one using an oscillator connected to the circuit, and the other one using an externally derived clock for input. Circuit constants in Figures 1.10.1 and 1.10.2 vary with each oscillator used. Use the values recommended by the manufacturer of your oscillator.

Note: Insert a damping resistor if required. The resistance will vary depending on the oscillator and the oscillation drive capacity setting. Use the value recommended by the maker of the oscillator.
When the oscillation drive capacity is set to low, check that oscillation is stable. Also, if the oscillator manufacturer's data sheet specifies that a feedback resistor be added external to the chip, insert a feedback resistor between XIN and XOUT following the instruction.

Figure 1.10.1. Examples of main clock

Note: Insert a damping resistor if required. The resistance will vary depending on the oscillator and the oscillation drive capacity setting. Use the value recommended by the maker of the oscillator.
When the oscillation drive capacity is set to low, check that oscillation is stable. Also, if the oscillator manufacturer's data sheet specifies that a feedback resistor be added external to the chip, insert a feedback resistor between XCIN and XCOUT following the instruction.

Figure 1.10.2. Examples of sub-clock

Clock Control

Figure 1.10.3 shows the block diagram of the clock generating circuit.

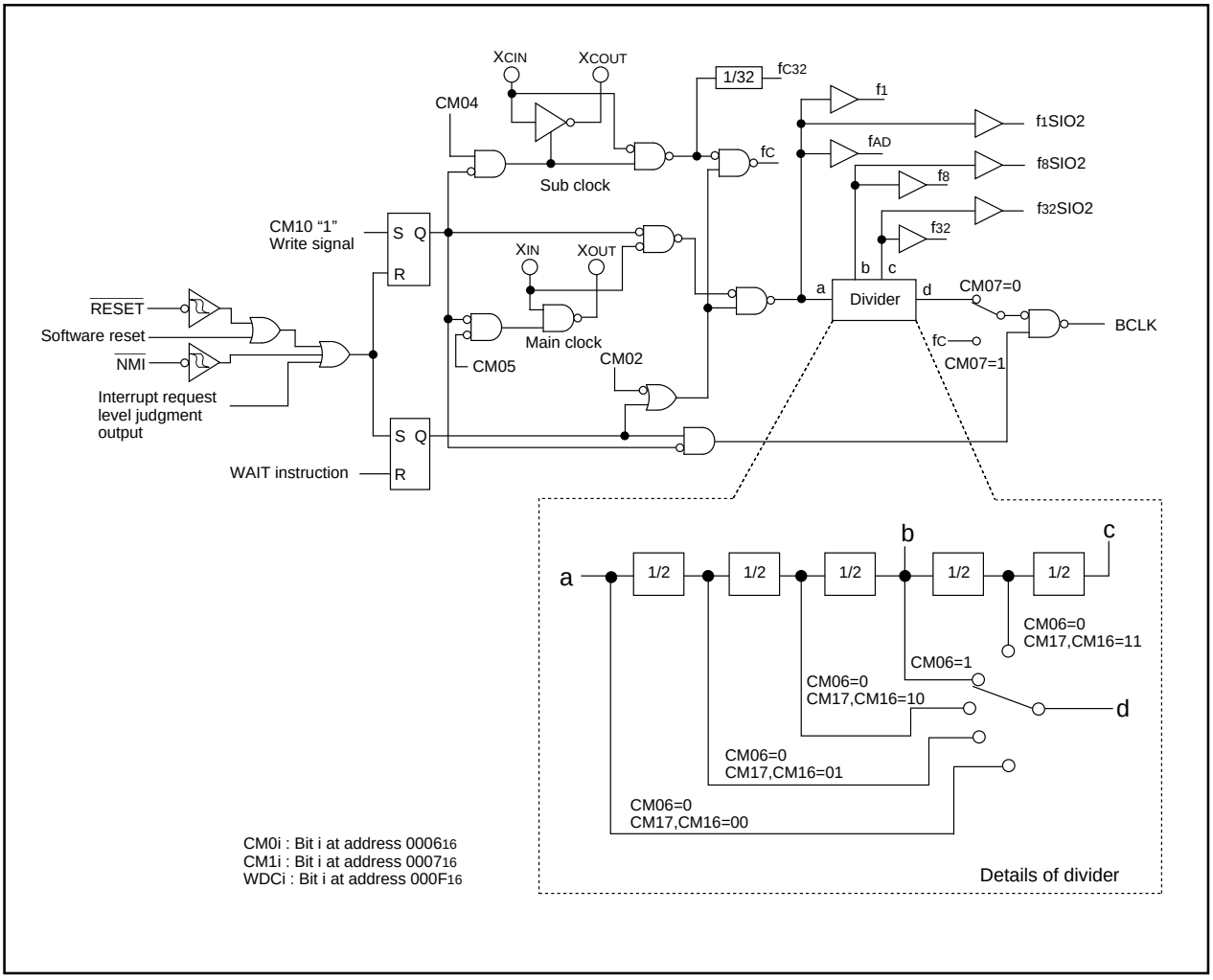


Figure 1.10.3. Clock generating circuit

The following paragraphs describes the clocks generated by the clock generating circuit.

(1) Main clock

The main clock is generated by the main clock oscillation circuit. After a reset, the clock is divided by 8 to the BCLK. The clock can be stopped using the main clock stop bit (bit 5 at address 0006_{16}). Stopping the clock, after switching the operating clock source of CPU to the sub-clock, reduces the power dissipation. After the oscillation of the main clock oscillation circuit has stabilized, the drive capacity of the main clock oscillation circuit can be reduced using the XIN-XOUT drive capacity select bit (bit 5 at address 0007_{16}). Reducing the drive capacity of the main clock oscillation circuit reduces the power dissipation. This bit changes to "1" when shifting from high-speed/medium-speed mode to stop mode and at a reset. When shifting from low-speed/low power dissipation mode to stop mode, the value before stop mode is retained.

(2) Sub-clock

The sub-clock is generated by the sub-clock oscillation circuit. No sub-clock is generated after a reset. After oscillation is started using the port XC select bit (bit 4 at address 0006_{16}), the sub-clock can be selected as the BCLK by using the system clock select bit (bit 7 at address 0006_{16}). However, be sure that the sub-clock oscillation has fully stabilized before switching.

After the oscillation of the sub-clock oscillation circuit has stabilized, the drive capacity of the sub-clock oscillation circuit can be reduced using the XCIN-XCOUT drive capacity select bit (bit 3 at address 0006_{16}). Reducing the drive capacity of the sub-clock oscillation circuit reduces the power dissipation. This bit changes to "1" when shifting to stop mode and at a reset.

When the XCIN/XCOUT is used, set ports $P8_6$ and $P8_7$ as the input ports without pull-up.

(3) BCLK

The BCLK is the clock that drives the CPU, and is fC or the clock is derived by dividing the main clock by 1, 2, 4, 8, or 16. The BCLK is derived by dividing the main clock by 8 after a reset. The BCLK signal can be output from BCLK pin by the BCLK output disable bit (bit 7 at address 0004_{16}) in the memory expansion and the microprocessor modes.

The main clock division select bit 0(bit 6 at address 0006_{16}) changes to "1" when shifting from high-speed/medium-speed to stop mode and at reset. When shifting from low-speed/low power dissipation mode to stop mode, the value before stop mode is retained.

(4) Peripheral function clock(f_1, f_8, f_{32}, f_{1SIO2}, f_{8SIO2},f_{32SIO2},f_{AD})

The clock for the peripheral devices is derived from the main clock or by dividing it by 1, 8, or 32. The peripheral function clock is stopped by stopping the main clock or by setting the WAIT peripheral function clock stop bit (bit 2 at 0006_{16}) to "1" and then executing a WAIT instruction.

(5) f_{C32}

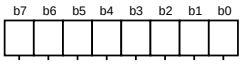
This clock is derived by dividing the sub-clock by 32. It is used for the timer A and timer B counts.

(6) fC

This clock has the same frequency as the sub-clock. It is used for the BCLK and for the watchdog timer.

Figure 1.10.4 shows the system clock control registers 0 and 1.

System clock control register 0 (Note 1)

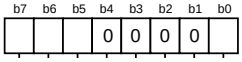


Symbol	Address	When reset
CM0	000616	4816

Bit symbol	Bit name	Function	R	W
CM00	Clock output function select bit (Valid only in single-chip mode)	b1 b0 0 0 : I/O port P57 0 1 : fC output 1 0 : f8 output 1 1 : f32 output	○	○
CM01			○	○
CM02	WAIT peripheral function clock stop bit	0 : Do not stop peripheral function clock in wait mode 1 : Stop peripheral function clock in wait mode (Note 8)	○	○
CM03	XCIN-XCOUT drive capacity select bit (Note 2)	0 : LOW 1 : HIGH	○	○
CM04	Port XC select bit	0 : I/O port 1 : XCIN-XCOUT generation (Note 9)	○	○
CM05	Main clock (XIN-XOUT) stop bit (Note 3, 4, 5)	0 : On 1 : Off	○	○
CM06	Main clock division select bit 0 (Note 7)	0 : CM16 and CM17 valid 1 : Division by 8 mode	○	○
CM07	System clock select bit (Note 6)	0 : XIN, XOUT 1 : XCIN, XCOUT	○	○

Note 1: Set bit 0 of the protect register (address 000A16) to “1” before writing to this register.
Note 2: Changes to “1” when shiffing to stop mode and at a reset.
Note 3: When entering low power dissipation mode, main clock stops by using this bit. To stop the main clock, when the sub clock oscillation is stable, set system clock select bit (CM07) to “1” before setting this bit to “1”.
Note 4: When inputting external clock, only clock oscillation buffer is stopped and clock input is acceptable.
Note 5: If this bit is set to “1”, XOUT turns “H”. The built-in feedback resistor remains being connected, so XIN turns pulled up to XOUT (“H”) via the feedback resistor.
Note 6: Set port XC select bit (CM04) to “1” and stabilize the sub-clock oscillating before setting this bit from “0” to “1”. Do not write to both bits at the same time. And also, set the main clock stop bit (CM05) to “0” and stabilize the main clock oscillating before setting this bit from “1” to “0”.
Note 7: This bit changes to “1” when shifting from high-speed/medium-speed mode to stop mode and at a reset. When shifting from low-speed/low power dissipation mode to stop mode, the value before stop mode is retained.
Note 8: fC32 is not included. Do not set to “1” when using low-speed or low power dissipation mode.
Note 9: When the XCIN/XCOUT is used, set ports P86 and P87 as the input ports without pull-up.

System clock control register 1 (Note 1)



Symbol	Address	When reset
CM1	000716	2016

Bit symbol	Bit name	Function	R	W
CM10	All clock stop control bit (Note4)	0 : Clock on 1 : All clocks off (stop mode)	○	○
Reserved bit		Must always be set to “0”	○	○
Reserved bit		Must always be set to “0”	○	○
Reserved bit		Must always be set to “0”	○	○
Reserved bit		Must always be set to “0”	○	○
CM15	XIN-XOUT drive capacity select bit (Note 2)	0 : LOW 1 : HIGH	○	○
CM16	Main clock division select bit 1 (Note 3)	b7 b6 0 0 : No division mode 0 1 : Division by 2 mode 1 0 : Division by 4 mode 1 1 : Division by 16 mode	○	○
CM17				

Note 1: Set bit 0 of the protect register (address 000A16) to “1” before writing to this register.
Note 2: This bit changes to “1” when shifting from high-speed/medium-speed mode to stop mode and at a reset. When shifting from low-speed/low power dissipation mode to stop mode, the value before stop mode is retained.
Note 3: Can be selected when bit 6 of the system clock control register 0 (address 000616) is “0”. If “1”, division mode is fixed at 8.
Note 4: If this bit is set to “1”, XOUT turns “H”, and the built-in feedback resistor is cut off. XCIN and XCOUT turn high-impedance state.

Figure 1.10.4. Clock control registers 0 and 1

Clock Output

In single-chip mode, the clock output function select bits (bits 0 and 1 at address 0006_{16}) enable f_8, f_{32}, or fc to be output from the P57/CLKOUT pin. When the WAIT peripheral function clock stop bit (bit 2 at address 0006_{16}) is set to “1”, the output of f_8 and f_{32} stops when a WAIT instruction is executed.

Stop Mode

Writing “1” to the all-clock stop control bit (bit 0 at address 0007_{16}) stops all oscillation and the microcomputer enters stop mode. In stop mode, the content of the internal RAM is retained provided that V_{CC} remains above 2V.

Because the oscillation , BCLK, f_1 to f_{32}, f_{1SIO2} to f_{32SIO2}, f_C, f_{C32}, and f_{AD} stops in stop mode, peripheral functions such as the A-D converter and watchdog timer do not function. However, timer A and timer B operate provided that the event counter mode is set to an external pulse, and UARTi(i = 0 to 2), SI/O3,4 functions provided an external clock is selected. Table 1.10.2 shows the status of the ports in stop mode.

Stop mode is cancelled by a hardware reset or an interrupt. If an interrupt is to be used to cancel stop mode, that interrupt must first have been enabled, and the priority level of the interrupt which is not used to cancel must have been changed to 0. If returning by an interrupt, that interrupt routine is executed. If only a hardware reset or an $\overline{NMI}$ interrupt is used to cancel stop mode, change the priority level of all interrupt to 0, then shift to stop mode.

When shifting from high-speed/medium-speed mode to stop mode and at a reset, the main clock division select bit 0 (bit 6 at address 0006_{16}) is set to “1”. When shifting from low-speed/low power dissipation mode to stop mode, the value before stop mode is retained.

Table 1.10.2. Port status during stop mode

Pin		Memory expansion mode Microprocessor mode	Single-chip mode
Address bus, data bus, $\overline{CS0}$ to $\overline{CS3}$, $\overline{BHE}$		Retains status before stop mode	
$\overline{RD}$, $\overline{WR}$, $\overline{WRL}$, $\overline{WRH}$		“H”	
$\overline{HLDA}$, BCLK		“H”	
ALE		“H”	
Port		Retains status before stop mode	Retains status before stop mode
CLKOUT	When fc selected	Valid only in single-chip mode	“H”
	When f_8, f_{32} selected	Valid only in single-chip mode	Retains status before stop mode

Wait Mode

When a WAIT instruction is executed, the BCLK stops and the microcomputer enters the wait mode. In this mode, oscillation continues but the BCLK and watchdog timer stop. Writing "1" to the WAIT peripheral function clock stop bit and executing a WAIT instruction stops the clock being supplied to the internal peripheral functions, allowing power dissipation to be reduced. However, peripheral function clock f_{C32} does not stop so that the peripherals using f_{C32} do not contribute to the power saving. When the MCU running in low-speed or low power dissipation mode, do not enter WAIT mode with this bit set to "1". Table 1.10.3 shows the status of the ports in wait mode.

Wait mode is cancelled by a hardware reset or an interrupt. If an interrupt is used to cancel wait mode, that interrupt must first have been enabled, and the priority level of the interrupt which is not used to cancel must have been changed to 0. If returning by an interrupt, the clock in which the WAIT instruction executed is set to BCLK by the microcomputer, and the action is resumed from the interrupt routine. If only a hardware reset or an $\overline{NMI}$ interrupt is used to cancel wait mode, change the priority level of all interrupt to 0,then shift to wait mode.

Table 1.10.3. Port status during wait mode

Pin		Memory expansion mode Microprocessor mode	Single-chip mode
Address bus, data bus, $\overline{CS0}$ to $\overline{CS3}$, $\overline{BHE}$		Retains status before wait mode	
$\overline{RD}$, $\overline{WR}$, $\overline{WRL}$, $\overline{WRH}$		"H"	
$\overline{HLDA}$,BCLK		"H"	
ALE		"H"	
Port		Retains status before wait mode	Retains status before wait mode
CLK_{OUT}	When f_C selected	Valid only in single-chip mode	Does not stop
	When f_8, f_{32} selected	Valid only in single-chip mode	Does not stop when the WAIT peripheral function clock stop bit is "0". When the WAIT peripheral function clock stop bit is "1", the status immediately prior to entering wait mode is maintained.

Status Transition of BCLK

Power dissipation can be reduced and low-voltage operation achieved by changing the count source for BCLK. Table 1.10.4 shows the operating modes corresponding to the settings of system clock control registers 0 and 1.

When reset, the device starts in division by 8 mode. The main clock division select bit 0(bit 6 at address 0006_{16}) changes to "1" when shifting from high-speed/medium-speed to stop mode and at a reset. When shifting from low-speed/low power dissipation mode to stop mode, the value before stop mode is retained. The following shows the operational modes of BCLK.

(1) Division by 2 mode

The main clock is divided by 2 to obtain the BCLK.

(2) Division by 4 mode

The main clock is divided by 4 to obtain the BCLK.

(3) Division by 8 mode

The main clock is divided by 8 to obtain the BCLK. When reset, the device starts operating from this mode. Before the user can go from this mode to no division mode, division by 2 mode, or division by 4 mode, the main clock must be oscillating stably. When going to low-speed or lower power consumption mode, make sure the sub-clock is oscillating stably.

(4) Division by 16 mode

The main clock is divided by 16 to obtain the BCLK.

(5) No-division mode

The main clock is divided by 1 to obtain the BCLK.

(6) Low-speed mode

f_C is used as the BCLK. Note that oscillation of both the main and sub-clocks must have stabilized before transferring from this mode to another or vice versa. At least 2 to 3 seconds are required after the sub-clock starts. Therefore, the program must be written to wait until this clock has stabilized immediately after powering up and after stop mode is cancelled.

(7) Low power dissipation mode

f_C is the BCLK and the main clock is stopped.

Note : Before the count source for BCLK can be changed from X_{IN} to X_{CIN} or vice versa, the clock to which the count source is going to be switched must be oscillating stably. Allow a wait time in software for the oscillation to stabilize before switching over the clock.

Table 1.10.4. Operating modes dictated by settings of system clock control registers 0 and 1

CM17	CM16	CM07	CM06	CM05	CM04	Operating mode of BCLK
0	1	0	0	0	Invalid	Division by 2 mode
1	0	0	0	0	Invalid	Division by 4 mode
Invalid	Invalid	0	1	0	Invalid	Division by 8 mode
1	1	0	0	0	Invalid	Division by 16 mode
0	0	0	0	0	Invalid	No-division mode
Invalid	Invalid	1	Invalid	0	1	Low-speed mode
Invalid	Invalid	1	Invalid	1	1	Low power dissipation mode

Power control

The following is a description of the three available power control modes:

Modes

Power control is available in three modes.

(a) Normal operation mode

- **High-speed mode**
  Divide-by-1 frequency of the main clock becomes the BCLK. The CPU operates with the BCLK. Each peripheral function operates according to its assigned clock.

- **Medium-speed mode**
  Divide-by-2, divide-by-4, divide-by-8, or divide-by-16 frequency of the main clock becomes the BCLK. The CPU operates with the BCLK. Each peripheral function operates according to its assigned clock.

- **Low-speed mode**
  fC becomes the BCLK. The CPU operates according to the fC clock. The fC clock is supplied by the subclock. Each peripheral function operates according to its assigned clock.

- **Low power dissipation mode**
  The main clock operating in low-speed mode is stopped. The CPU operates according to the fC clock. The fC clock is supplied by the subclock. The only peripheral functions that operate are those with the subclock selected as the count source.

(b) Wait mode

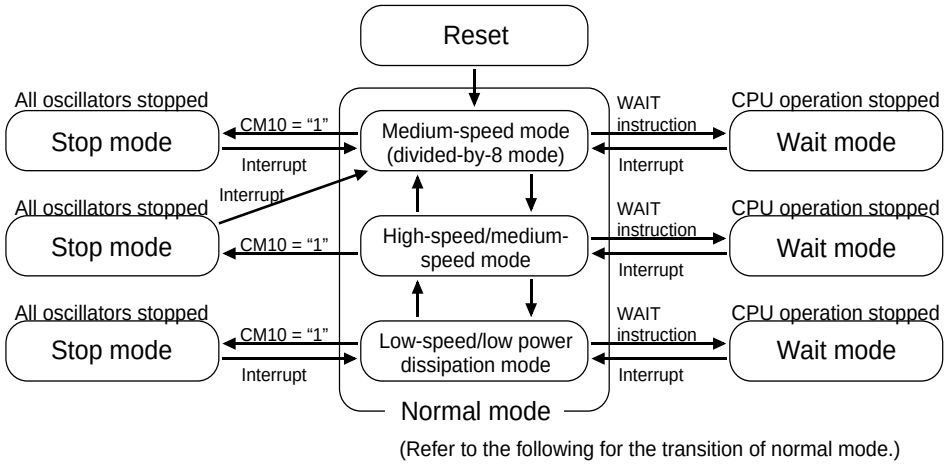
The CPU operation is stopped. The oscillators do not stop.

(c) Stop mode

All oscillators stop. The CPU and all built-in peripheral functions stop. This mode, among the three modes listed here, is the most effective in decreasing power consumption.

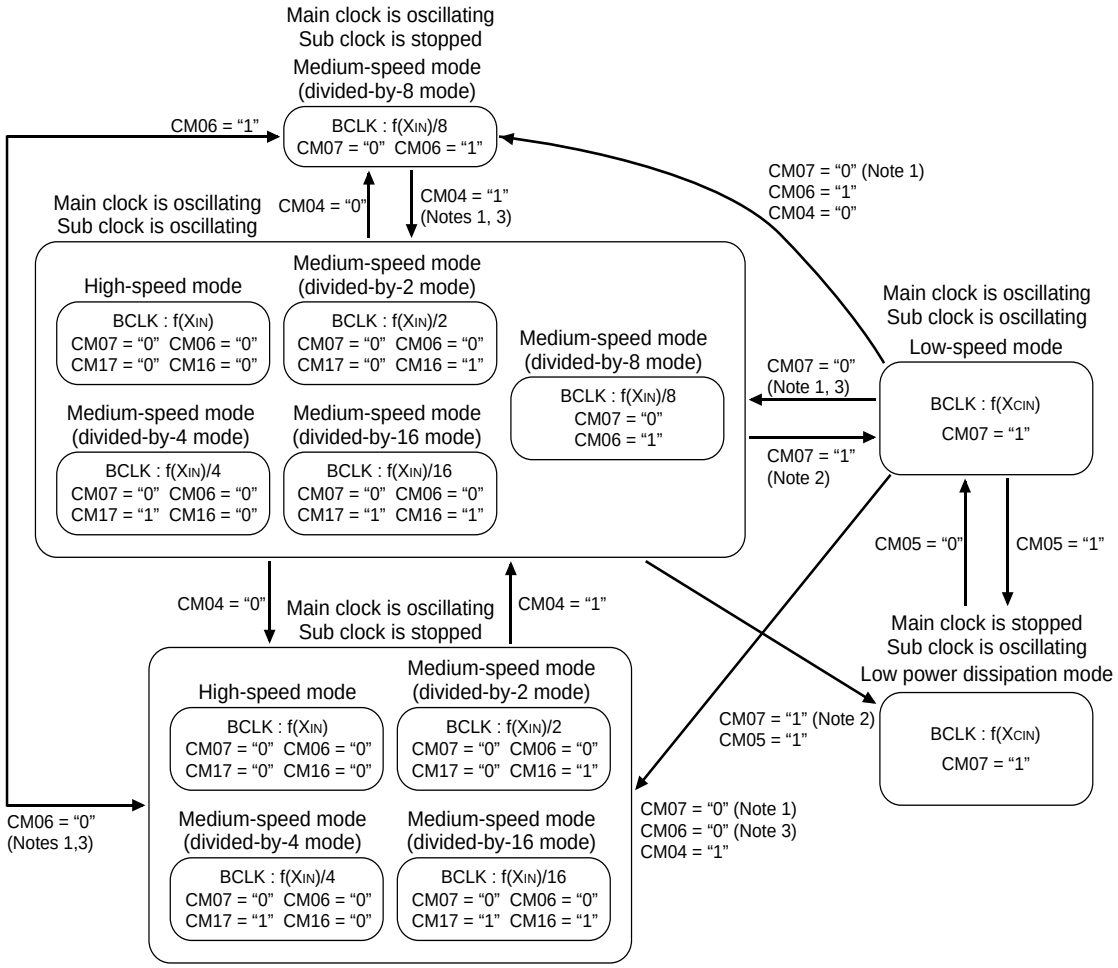
Figure 1.10.5 is the state transition diagram of the above modes.

Transition of stop mode, wait mode



(Refer to the following for the transition of normal mode.)

Transition of normal mode



Note 1: Switch clock after oscillation of main clock is sufficiently stable.
Note 2: Switch clock after oscillation of sub clock is sufficiently stable.
Note 3: Change CM06 after changing CM17 and CM16.
Note 4: Transit in accordance with arrow.

Figure 1.10.5. State transition diagram of Power control mode

Protection

The protection function is provided so that the values in important registers cannot be changed in the event that the program runs out of control. Figure 1.10.6 shows the protect register. The values in the processor mode register 0 (address 0004_{16}), processor mode register 1 (address 0005_{16}), system clock control register 0 (address 0006_{16}), system clock control register 1 (address 0007_{16}), port P9 direction register (address $03F3_{16}$) , SI/O3 control register (address 0362_{16}) and SI/O4 control register (address 0366_{16}) can only be changed when the respective bit in the protect register is set to “1”. Therefore, important outputs can be allocated to port P9.

If, after “1” (write-enabled) has been written to the port P9 direction register and SI/Oi control register (i=3,4) write-enable bit (bit 2 at address $000A_{16}$), a value is written to any address, the bit automatically reverts to “0” (write-inhibited). However, the system clock control registers 0 and 1 write-enable bit (bit 0 at $000A_{16}$) and processor mode register 0 and 1 write-enable bit (bit 1 at $000A_{16}$) do not automatically return to “0” after a value has been written to an address. The program must therefore be written to return these bits to “0”.

Protect register

b7	b6	b5	b4	b3	b2	b1	b0
✕	✕	✕	✕	✕			

Symbol: PRCR
Address: $000A_{16}$
When reset: $XXXXX000_2$

Bit symbol	Bit name	Function	R	W
PRC0	Enables writing to system clock control registers 0 and 1 (addresses 0006_{16} and 0007_{16})	0 : Write-inhibited 1 : Write-enabled	○	○
PRC1	Enables writing to processor mode registers 0 and 1 (addresses 0004_{16} and 0005_{16})	0 : Write-inhibited 1 : Write-enabled	○	○
PRC2	Enables writing to port P9 direction register (address $03F3_{16}$) and SI/Oi control register (i=3,4) (addresses 0362_{16} and 0366_{16}) (Note)	0 : Write-inhibited 1 : Write-enabled	○	○
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be indeterminate.			—	—

Note: Writing a value to an address after “1” is written to this bit returns the bit to “0” . Other bits do not automatically return to “0” and they must therefore be reset by the program.

Figure 1.10.6. Protect register

Overview of Interrupt

Type of Interrupts

Figure 1.11.1 lists the types of interrupts.

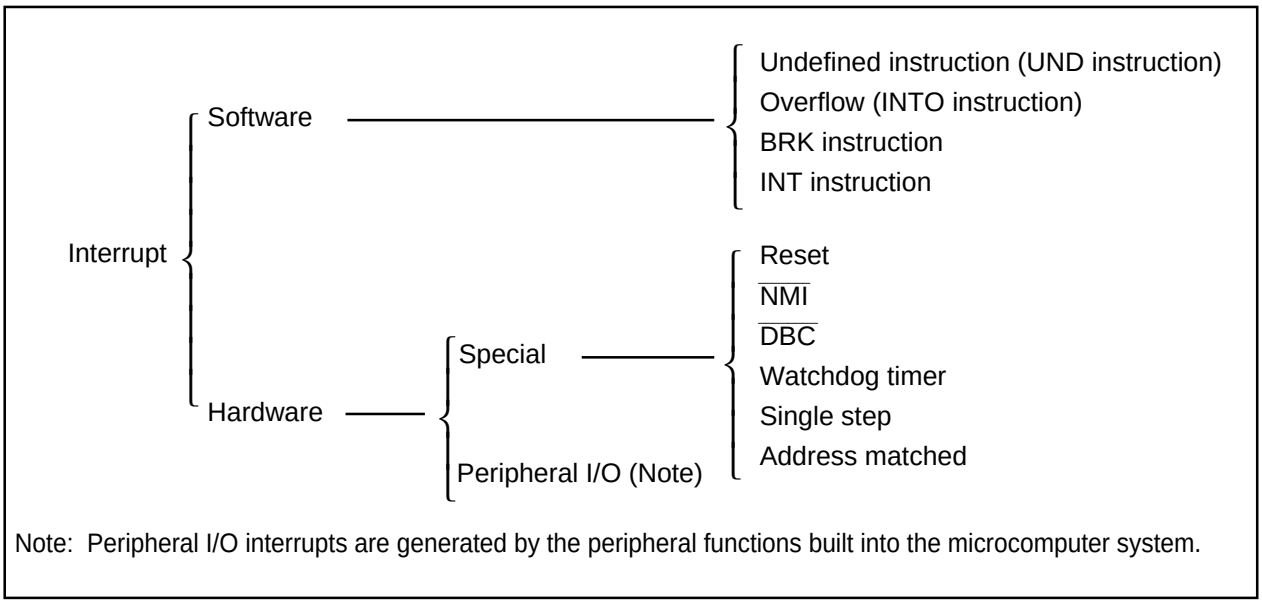


Figure 1.11.1. Classification of interrupts

- Maskable interrupt : An interrupt which can be enabled (disabled) by the interrupt enable flag (I flag) or whose interrupt priority **can be changed** by priority level.
- Non-maskable interrupt : An interrupt which cannot be enabled (disabled) by the interrupt enable flag (I flag) or whose interrupt priority **cannot be changed** by priority level.

Software Interrupts

A software interrupt occurs when executing certain instructions. Software interrupts are non-maskable interrupts.

- **Undefined instruction interrupt**
  An undefined instruction interrupt occurs when executing the UND instruction.
- **Overflow interrupt**
  An overflow interrupt occurs when executing the INTO instruction with the overflow flag (O flag) set to "1". The following are instructions whose O flag changes by arithmetic:
  ABS, ADC, ADCF, ADD, CMP, DIV, DIVU, DIVX, NEG, RMPA, SBB, SHA, SUB
- **BRK interrupt**
  A BRK interrupt occurs when executing the BRK instruction.
- **INT interrupt**
  An INT interrupt occurs when specifying one of software interrupt numbers 0 through 63 and executing the INT instruction. Software interrupt numbers 0 through 31 are assigned to peripheral I/O interrupts, so executing the INT instruction allows executing the same interrupt routine that a peripheral I/O interrupt does.
  The stack pointer (SP) used for the INT interrupt is dependent on which software interrupt number is involved.
  So far as software interrupt numbers 0 through 31 are concerned, the microcomputer saves the stack pointer assignment flag (U flag) when it accepts an interrupt request. If change the U flag to "0" and select the interrupt stack pointer (ISP), and then execute an interrupt sequence. When returning from the interrupt routine, the U flag is returned to the state it was before the acceptance of interrupt request. So far as software numbers 32 through 63 are concerned, the stack pointer does not make a shift.

Hardware Interrupts

Hardware interrupts are classified into two types — special interrupts and peripheral I/O interrupts.

(1) Special interrupts

Special interrupts are non-maskable interrupts.

- **Reset**

  Reset occurs if an “L” is input to the $\overline{\text{RESET}}$ pin.

- **$\overline{\text{NMI}}$ interrupt**

  An $\overline{\text{NMI}}$ interrupt occurs if an “L” is input to the $\overline{\text{NMI}}$ pin.

- **$\overline{\text{DBC}}$ interrupt**

  This interrupt is exclusively for the debugger, do not use it in other circumstances.

- **Watchdog timer interrupt**

  Generated by the watchdog timer.

- **Single-step interrupt**

  This interrupt is exclusively for the debugger, do not use it in other circumstances. With the debug flag (D flag) set to “1”, a single-step interrupt occurs after one instruction is executed.

- **Address match interrupt**

  An address match interrupt occurs immediately before the instruction held in the address indicated by the address match interrupt register is executed with the address match interrupt enable bit set to “1”. If an address other than the first address of the instruction in the address match interrupt register is set, no address match interrupt occurs.

(2) Peripheral I/O interrupts

A peripheral I/O interrupt is generated by one of built-in peripheral functions. Built-in peripheral functions are dependent on classes of products, so the interrupt factors too are dependent on classes of products. The interrupt vector table is the same as the one for software interrupt numbers 0 through 31 the INT instruction uses. Peripheral I/O interrupts are maskable interrupts.

- **Bus collision detection interrupt**

  This is an interrupt that the serial I/O bus collision detection generates.

- **DMA0 interrupt, DMA1 interrupt**

  These are interrupts that DMA generates.

- **Key-input interrupt**

  A key-input interrupt occurs if an “L” is input to the $\overline{\text{KI}}$ pin.

- **A-D conversion interrupt**

  This is an interrupt that the A-D converter generates.

- **UART0, UART1, UART2/NACK, SI/O3 and SI/O4 transmission interrupt**

  These are interrupts that the serial I/O transmission generates.

- **UART0, UART1, UART2/ACK, SI/O3 and SI/O4 reception interrupt**

  These are interrupts that the serial I/O reception generates.

- **Timer A0 interrupt through timer A4 interrupt**

  These are interrupts that timer A generates

- **Timer B0 interrupt through timer B5 interrupt**

  These are interrupts that timer B generates.

- **$\overline{\text{INT0}}$ interrupt through $\overline{\text{INT5}}$ interrupt**

  An $\overline{\text{INT}}$ interrupt occurs if either a rising edge or a falling edge or a both edge is input to the $\overline{\text{INT}}$ pin.

Interrupts and Interrupt Vector Tables

If an interrupt request is accepted, a program branches to the interrupt routine set in the interrupt vector table. Set the first address of the interrupt routine in each vector table. Figure 1.11.2 shows the format for specifying the address.

Two types of interrupt vector tables are available — fixed vector table in which addresses are fixed and variable vector table in which addresses can be varied by the setting.

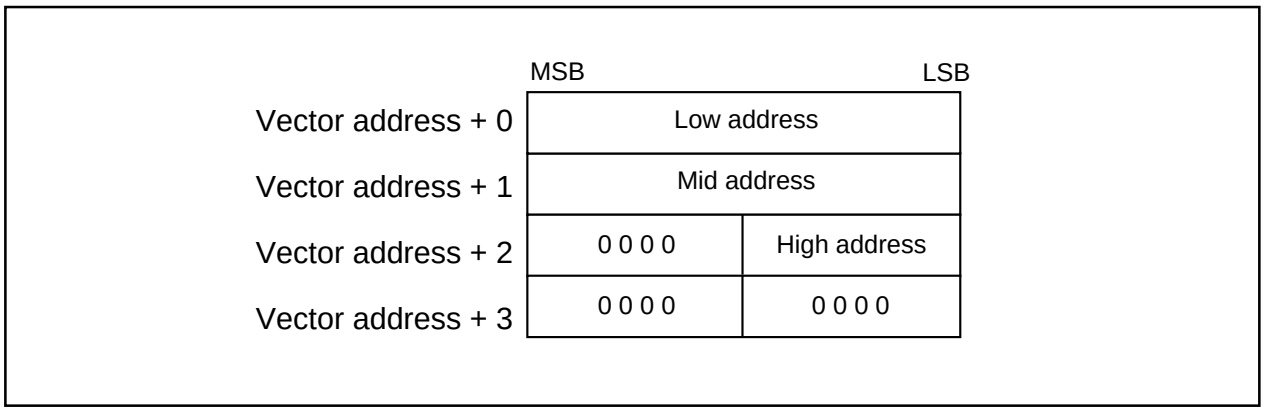


Figure 1.11.2. Format for specifying interrupt vector addresses

• Fixed vector tables

The fixed vector table is a table in which addresses are fixed. The vector tables are located in an area extending from FFFDC16 to FFFFF16. One vector table comprises four bytes. Set the first address of interrupt routine in each vector table. Table 1.11.1 shows the interrupts assigned to the fixed vector tables and addresses of vector tables.

Table 1.11.1. Interrupts assigned to the fixed vector tables and addresses of vector tables

Interrupt source	Vector table addresses Address (L) to address (H)	Remarks
Undefined instruction	FFFDC16 to FFFDF16	Interrupt on UND instruction
Overflow	FFFE016 to FFFE316	Interrupt on INTO instruction
BRK instruction	FFFE416 to FFFE716	If the vector contains FF16, program execution starts from the address shown by the vector in the variable vector table
Address match	FFFE816 to FFFEB16	There is an address-matching interrupt enable bit
Single step (Note)	FFFEC16 to FFFEF16	Do not use
Watchdog timer	FFFF016 to FFFF316	
$\overline{\text{DBC}}$ (Note)	FFFF416 to FFFF716	Do not use
$\overline{\text{NMI}}$	FFFF816 to FFFFB16	External interrupt by input to $\overline{\text{NMI}}$ pin
Reset	FFFFC16 to FFFFF16	

Note: Interrupts used for debugging purposes only.

• **Variable vector tables**

The addresses in the variable vector table can be modified, according to the user's settings. Indicate the first address using the interrupt table register (INTB). The 256-byte area subsequent to the address the INTB indicates becomes the area for the variable vector tables. One vector table comprises four bytes. Set the first address of the interrupt routine in each vector table. Table 1.11.2 shows the interrupts assigned to the variable vector tables and addresses of vector tables.

Table 1.11.2. Interrupts assigned to the variable vector tables and addresses of vector tables

Software interrupt number	Vector table address Address (L) to address (H)	Interrupt source	Remarks
Software interrupt number 0	+0 to +3 (Note 1)	BRK instruction	Cannot be masked I flag
Software interrupt number 4	+16 to +19 (Note 1)	$\overline{\text{INT3}}$	
Software interrupt number 5	+20 to +23 (Note 1)	Timer B5	
Software interrupt number 6	+24 to +27 (Note 1)	Timer B4	
Software interrupt number 7	+28 to +31 (Note 1)	Timer B3	
Software interrupt number 8	+32 to +35 (Note 1)	SI/O4/$\overline{\text{INT5}}$ (Note 2)	
Software interrupt number 9	+36 to +39 (Note 1)	SI/O3/$\overline{\text{INT4}}$ (Note 2)	
Software interrupt number 10	+40 to +43 (Note 1)	Bus collision detection	
Software interrupt number 11	+44 to +47 (Note 1)	DMA0	
Software interrupt number 12	+48 to +51 (Note 1)	DMA1	
Software interrupt number 13	+52 to +55 (Note 1)	Key input interrupt	
Software interrupt number 14	+56 to +59 (Note 1)	A-D	
Software interrupt number 15	+60 to +63 (Note 1)	UART2 transmit/NACK (Note 3)	
Software interrupt number 16	+64 to +67 (Note 1)	UART2 receive/ACK (Note 3)	
Software interrupt number 17	+68 to +71 (Note 1)	UART0 transmit	
Software interrupt number 18	+72 to +75 (Note 1)	UART0 receive	
Software interrupt number 19	+76 to +79 (Note 1)	UART1 transmit	
Software interrupt number 20	+80 to +83 (Note 1)	UART1 receive	
Software interrupt number 21	+84 to +87 (Note 1)	Timer A0	
Software interrupt number 22	+88 to +91 (Note 1)	Timer A1	
Software interrupt number 23	+92 to +95 (Note 1)	Timer A2	
Software interrupt number 24	+96 to +99 (Note 1)	Timer A3	
Software interrupt number 25	+100 to +103 (Note 1)	Timer A4	
Software interrupt number 26	+104 to +107 (Note 1)	Timer B0	
Software interrupt number 27	+108 to +111 (Note 1)	Timer B1	
Software interrupt number 28	+112 to +115 (Note 1)	Timer B2	
Software interrupt number 29	+116 to +119 (Note 1)	$\overline{\text{INT0}}$	
Software interrupt number 30	+120 to +123 (Note 1)	$\overline{\text{INT1}}$	
Software interrupt number 31	+124 to +127 (Note 1)	$\overline{\text{INT2}}$	
Software interrupt number 32 to Software interrupt number 63	+128 to +131 (Note 1) to +252 to +255 (Note 1)	Software interrupt	Cannot be masked I flag

Note 1: Address relative to address in interrupt table register (INTB).
Note 2: It is selected by interrupt request cause bit (bit 6, 7 in address 035F16).
Note 3: When IIC mode is selected, NACK and ACK interrupts are selected.

Interrupt Control

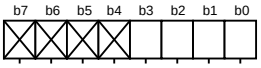
Descriptions are given here regarding how to enable or disable maskable interrupts and how to set the priority to be accepted. What is described here does not apply to non-maskable interrupts.

Enable or disable a maskable interrupt using the interrupt enable flag (I flag), interrupt priority level selection bit, or processor interrupt priority level (IPL). Whether an interrupt request is present or absent is indicated by the interrupt request bit. The interrupt request bit and the interrupt priority level selection bit are located in the interrupt control register of each interrupt. Also, the interrupt enable flag (I flag) and the IPL are located in the flag register (FLG).

Figure 1.11.3 shows the memory map of the interrupt control registers.

Interrupt control register (Note 2)

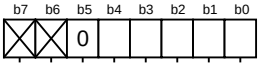
Symbol	Address	When reset
TBiIC(i=3 to 5)	0045_{16} to 0047_{16}	$XXXXX000_2$
BCNIC	$004A_{16}$	$XXXXX000_2$
DMiIC(i=0, 1)	$004B_{16}$, $004C_{16}$	$XXXXX000_2$
KUPIC	$004D_{16}$	$XXXXX000_2$
ADIC	$004E_{16}$	$XXXXX000_2$
SiTIC(i=0 to 2)	0051_{16}, 0053_{16}, $004F_{16}$	$XXXXX000_2$
SiRIC(i=0 to 2)	0052_{16}, 0054_{16}, 0050_{16}	$XXXXX000_2$
TAiIC(i=0 to 4)	0055_{16} to 0059_{16}	$XXXXX000_2$
TBiIC(i=0 to 2)	$005A_{16}$ to $005C_{16}$	$XXXXX000_2$



Bit symbol	Bit name	Function	R	W
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	○	○
ILVL1			○	○
ILVL2			○	○
IR	Interrupt request bit	0 : Interrupt not requested 1 : Interrupt requested	○	○ (Note 1)
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be indeterminate.			–	–

Note 1: This bit can only be accessed for reset (= 0), but cannot be accessed for set (= 1).
Note 2: To rewrite the interrupt control register, do so at a point that does not generate the interrupt request for that register. For details, see the precautions for interrupts.

Symbol		Address	When reset
INTiIC(i=3)		0044_{16}	$XX00X000_2$
SiIC/INTjIC	(i=4, 3)	0048_{16}, 0049_{16}	$XX00X000_2$
	(j=5, 4)	0048_{16}, 0049_{16}	$XX00X000_2$
INTiIC(i=0 to 2)		$005D_{16}$ to $005F_{16}$	$XX00X000_2$



Bit symbol	Bit name	Function	R	W
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	○	○
ILVL1			○	○
ILVL2			○	○
IR	Interrupt request bit	0: Interrupt not requested 1: Interrupt requested	○	○ (Note 1)
POL	Polarity select bit	0 : Selects falling edge 1 : Selects rising edge	○	○
Reserved bit		Must always be set to “0”	○	○
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be indeterminate.			–	–

Note 1: This bit can only be accessed for reset (= 0), but cannot be accessed for set (= 1).
Note 2: To rewrite the interrupt control register, do so at a point that does not generate the interrupt request for that register. For details, see the precautions for interrupts.

Figure 1.11.3. Interrupt control registers

Interrupt Enable Flag (I flag)

The interrupt enable flag (I flag) controls the enabling and disabling of maskable interrupts. Setting this flag to “1” enables all maskable interrupts; setting it to “0” disables all maskable interrupts. This flag is set to “0” after reset.

Interrupt Request Bit

The interrupt request bit is set to “1” by hardware when an interrupt is requested. After the interrupt is accepted and jumps to the corresponding interrupt vector, the request bit is set to “0” by hardware. The interrupt request bit can also be set to “0” by software. (Do not set this bit to “1”).

Interrupt Priority Level Select Bit and Processor Interrupt Priority Level (IPL)

Set the interrupt priority level using the interrupt priority level select bit, which is one of the component bits of the interrupt control register. When an interrupt request occurs, the interrupt priority level is compared with the IPL. The interrupt is enabled only when the priority level of the interrupt is higher than the IPL. Therefore, setting the interrupt priority level to “0” disables the interrupt.
Table 1.11.3 shows the settings of interrupt priority levels and Table 1.11.4 shows the interrupt levels enabled, according to the contents of the IPL.

The following are conditions under which an interrupt is accepted:
· interrupt enable flag (I flag) = “1”
· interrupt request bit = “1”
· interrupt priority level > IPL

The interrupt enable flag (I flag), the interrupt request bit, the interrupt priority select bit, and the IPL are independent, and they are not affected by one another.

Table 1.11.3. Settings of interrupt priority levels

Interrupt priority level select bit (b2 b1 b0)	Interrupt priority level	Priority order
0 0 0	Level 0 (interrupt disabled)	
0 0 1	Level 1	Low ↓
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	High

Table 1.11.4. Interrupt levels enabled according to the contents of the IPL

IPL (IPL_2 IPL_1 IPL_0)	Enabled interrupt priority levels
0 0 0	Interrupt levels 1 and above are enabled
0 0 1	Interrupt levels 2 and above are enabled
0 1 0	Interrupt levels 3 and above are enabled
0 1 1	Interrupt levels 4 and above are enabled
1 0 0	Interrupt levels 5 and above are enabled
1 0 1	Interrupt levels 6 and above are enabled
1 1 0	Interrupt levels 7 and above are enabled
1 1 1	All maskable interrupts are disabled

Rewrite the interrupt control register

To rewrite the interrupt control register, do so at a point that does not generate the interrupt request for that register. If there is possibility of the interrupt request occur, rewrite the interrupt control register after the interrupt is disabled. The program examples are described as follow:

Example 1:

```
INT_SWITCH1:
    FCLR    I             ; Disable interrupts.
    AND.B   #00h, 0055h   ; Clear TA0IC int. priority level and int. request bit.
    NOP                   ; Four NOP instructions are required when using HOLD function.
    NOP
    FSET    I             ; Enable interrupts.
```

Example 2:

```
INT_SWITCH2:
    FCLR    I             ; Disable interrupts.
    AND.B   #00h, 0055h   ; Clear TA0IC int. priority level and int. request bit.
    MOV.W   MEM, R0       ; Dummy read.
    FSET    I             ; Enable interrupts.
```

Example 3:

```
INT_SWITCH3:
    PUSHC   FLG           ; Push Flag register onto stack
    FCLR    I             ; Disable interrupts.
    AND.B   #00h, 0055h   ; Clear TA0IC int. priority level and int. request bit.
    POPC    FLG           ; Enable interrupts.
```

The reason why two NOP instructions (four when using the HOLD function) or dummy read are inserted before FSET I in Examples 1 and 2 is to prevent the interrupt enable flag I from being set before the interrupt control register is rewritten due to effects of the instruction queue.

When a instruction to rewrite the interrupt control register is executed but the interrupt is disabled, the interrupt request bit is not set sometimes even if the interrupt request for that register has been generated. This will depend on the instruction. If this creates problems, use the below instructions to change the register.

Instructions : AND, OR, BCLR, BSET

Interrupt Sequence

An interrupt sequence — what are performed over a period from the instant an interrupt is accepted to the instant the interrupt routine is executed — is described here.

If an interrupt occurs during execution of an instruction, the processor determines its priority when the execution of the instruction is completed, and transfers control to the interrupt sequence from the next cycle. If an interrupt occurs during execution of either the SMOVB, SMOVF, SSTR or RMPA instruction, the processor temporarily suspends the instruction being executed, and transfers control to the interrupt sequence.

In the interrupt sequence, the processor carries out the following in sequence given:

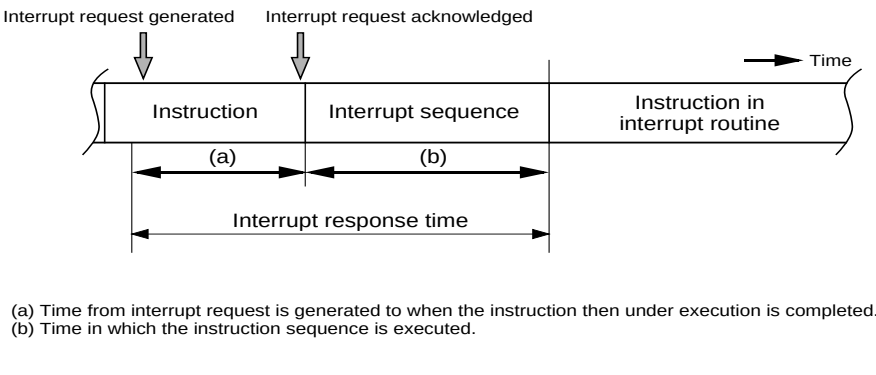
(1) CPU gets the interrupt information (the interrupt number and interrupt request level) by reading address 0000_{16}. After this, the corresponding interrupt request bit becomes “0”.
(2) Saves the content of the flag register (FLG) as it was immediately before the start of interrupt sequence in the temporary register (Note) within the CPU.
(3) Sets the interrupt enable flag (I flag), the debug flag (D flag), and the stack pointer select flag (U flag) to “0” (the U flag, however does not change if the INT instruction, in software interrupt numbers 32 through 63, is executed)
(4) Saves the content of the temporary register (Note) within the CPU in the stack area.
(5) Saves the content of the program counter (PC) in the stack area.
(6) Sets the interrupt priority level of the accepted instruction in the IPL.

After the interrupt sequence is completed, the processor resumes executing instructions from the first address of the interrupt routine.

Note: This register cannot be utilized by the user.

Interrupt Response Time

'Interrupt response time' is the period between the instant an interrupt occurs and the instant the first instruction within the interrupt routine has been executed. This time comprises the period from the occurrence of an interrupt to the completion of the instruction under execution at that moment (a) and the time required for executing the interrupt sequence (b). Figure 1.11.4 shows the interrupt response time.



(a) Time from interrupt request is generated to when the instruction then under execution is completed.
(b) Time in which the instruction sequence is executed.

Figure 1.11.4. Interrupt response time

Time (a) is dependent on the instruction under execution. Thirty cycles is the maximum required for the DIVX instruction (without wait).

Time (b) is as shown in Table 1.11.5.

Table 1.11.5. Time required for executing the interrupt sequence

Interrupt vector address	Stack pointer (SP) value	16-Bit bus, without wait	8-Bit bus, without wait
Even	Even	18 cycles (Note 1)	20 cycles (Note 1)
Even	Odd	19 cycles (Note 1)	20 cycles (Note 1)
Odd (Note 2)	Even	19 cycles (Note 1)	20 cycles (Note 1)
Odd (Note 2)	Odd	20 cycles (Note 1)	20 cycles (Note 1)

Note 1: Add 2 cycles in the case of a $\overline{\text{DBC}}$ interrupt; add 1 cycle in the case either of an address match interrupt or of a single-step interrupt.

Note 2: Locate an interrupt vector address in an even address, if possible.

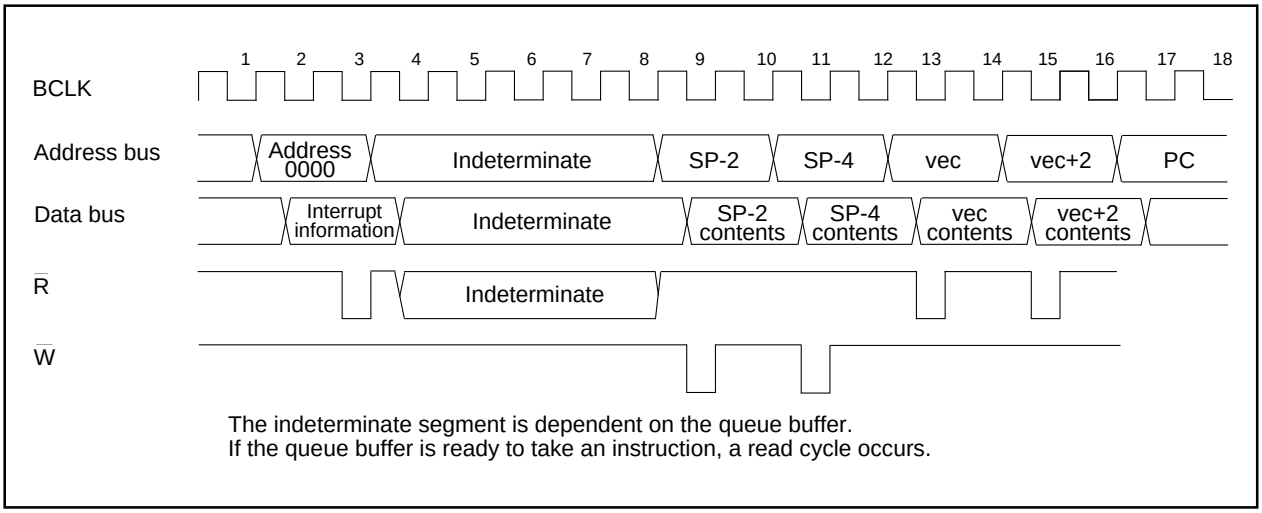


Figure 1.11.5. Time required for executing the interrupt sequence

Variation of IPL when Interrupt Request is Accepted

If an interrupt request is accepted, the interrupt priority level of the accepted interrupt is set in the IPL.

If an interrupt request, that does not have an interrupt priority level, is accepted, one of the values shown in Table 1.11.6 is set in the IPL.

Table 1.11.6. Relationship between interrupts without interrupt priority levels and IPL

Interrupt sources without priority levels	Value set in the IPL
Watchdog timer, $\overline{\text{NMI}}$	7
Reset	0
Other	Not changed

Saving Registers

In the interrupt sequence, only the contents of the flag register (FLG) and that of the program counter (PC) are saved in the stack area.

First, the processor saves the four higher-order bits of the program counter, and 4 upper-order bits and 8 lower-order bits of the FLG register, 16 bits in total, in the stack area, then saves 16 lower-order bits of the program counter. Figure 1.11.6 shows the state of the stack as it was before the acceptance of the interrupt request, and the state the stack after the acceptance of the interrupt request.

Save other necessary registers at the beginning of the interrupt routine using software. Using the PUSHM instruction alone can save all the registers except the stack pointer (SP).

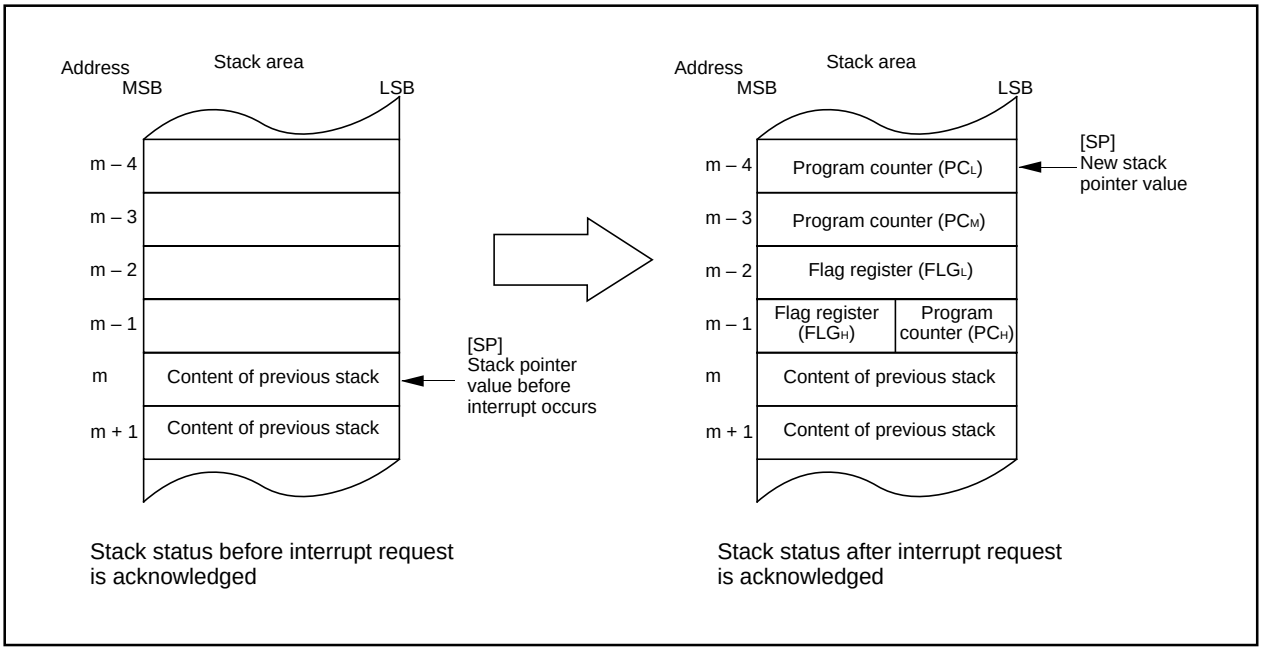


Figure 1.11.6. State of stack before and after acceptance of interrupt request

The operation of saving registers carried out in the interrupt sequence is dependent on whether the content of the stack pointer (Note) , at the time of acceptance of an interrupt request, is even or odd. If the content of the stack pointer (Note) is even, the content of the flag register (FLG) and the content of the program counter (PC) are saved, 16 bits at a time. If odd, their contents are saved in two steps, 8 bits at a time. Figure 1.11.7 shows the operation of the saving registers.

Note: When any INT instruction in software numbers 32 to 63 has been executed, this is the stack pointer indicated by the U flag. Otherwise, it is the interrupt stack pointer (ISP).

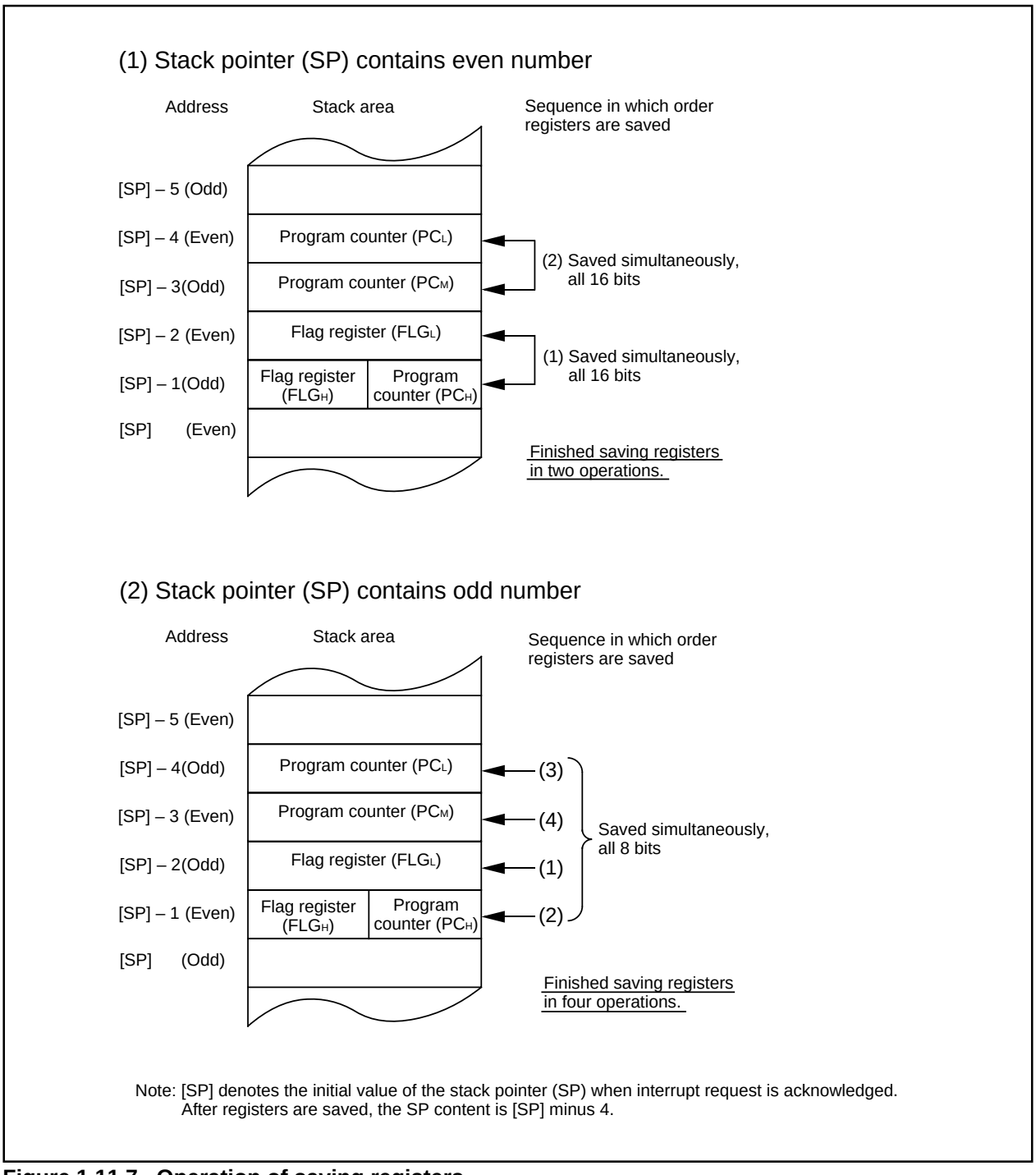


Figure 1.11.7. Operation of saving registers

Returning from an Interrupt Routine

Executing the REIT instruction at the end of an interrupt routine returns the contents of the flag register (FLG) as it was immediately before the start of interrupt sequence and the contents of the program counter (PC), both of which have been saved in the stack area. Then control returns to the program that was being executed before the acceptance of the interrupt request, so that the suspended process resumes.

Return the other registers saved by software within the interrupt routine using the POPM or similar instruction before executing the REIT instruction.

Interrupt Priority

If there are two or more interrupt requests occurring at a point in time within a single sampling (checking whether interrupt requests are made), the interrupt assigned a higher priority is accepted.

Assign an arbitrary priority to maskable interrupts (peripheral I/O interrupts) using the interrupt priority level select bit. If the same interrupt priority level is assigned, however, the interrupt assigned a higher hardware priority is accepted.

Priorities of the special interrupts, such as Reset (dealt with as an interrupt assigned the highest priority), watchdog timer interrupt, etc. are regulated by hardware.

Figure 1.11.8 shows the priorities of hardware interrupts.

Software interrupts are not affected by the interrupt priority. If an instruction is executed, control branches invariably to the interrupt routine.

Reset > $\overline{\text{NMI}}$ > $\overline{\text{DBC}}$ > Watchdog timer > Peripheral I/O > Single step > Address match

Figure 1.11.8. Hardware interrupts priorities

Interrupt resolution circuit

When two or more interrupts are generated simultaneously, this circuit selects the interrupt with the highest priority level. Figure 1.11.9 shows the circuit that judges the interrupt priority level.

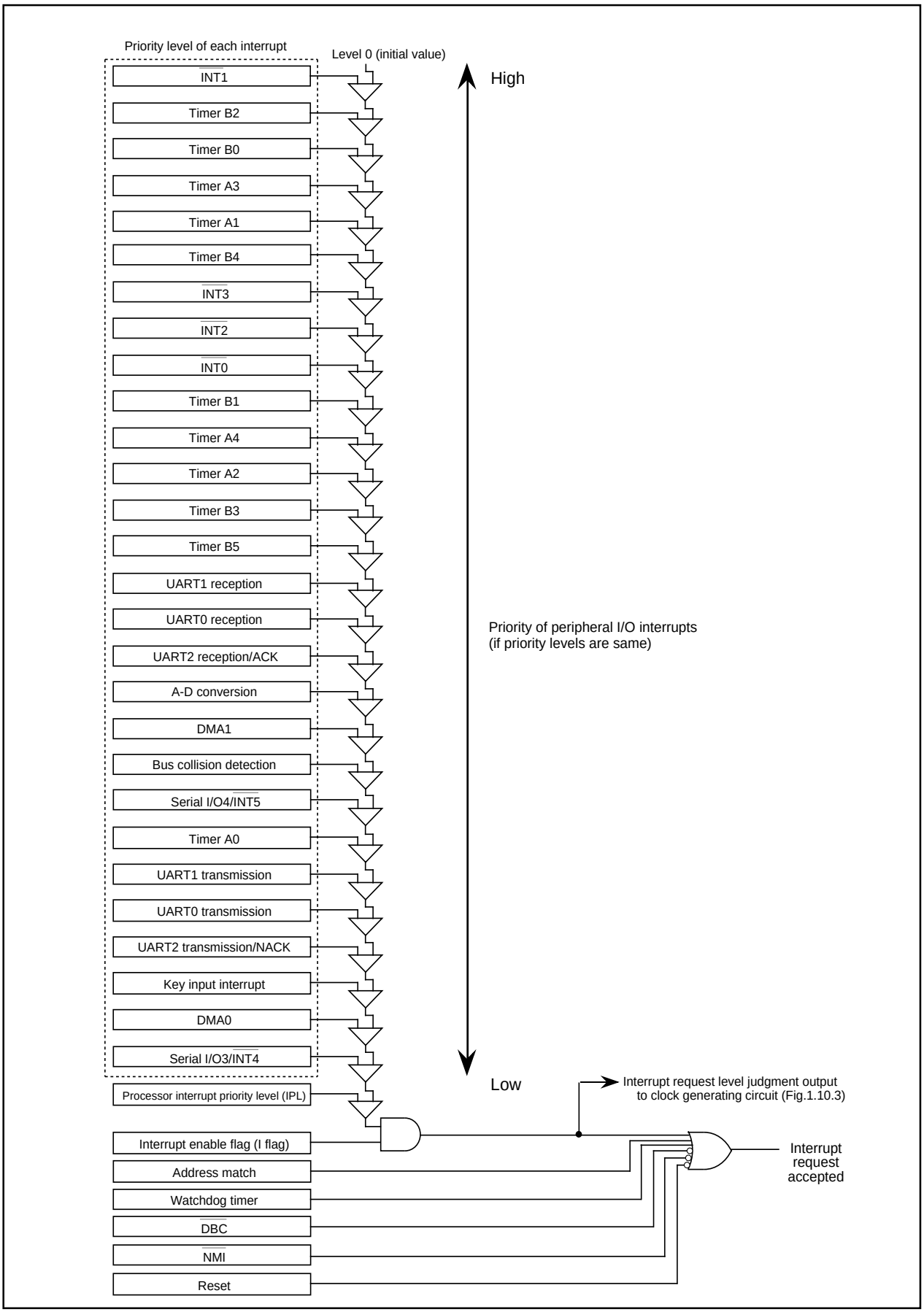


Figure 1.11.9. Maskable interrupts priorities (peripheral I/O interrupts)

INT Interrupt

INT0 to INT5 are triggered by the edges of external inputs. The edge polarity is selected using the polarity select bit.

Of interrupt control registers, 0048$_{16}$ is used both as serial I/O4 and external interrupt INT5 input control register, and 0049$_{16}$ is used both as serial I/O3 and as external interrupt INT4 input control register. Use the interrupt request cause select bits - bits 6 and 7 of the interrupt request cause select register (035F$_{16}$) - to specify which interrupt request cause to select. After having set an interrupt request cause, be sure to clear the corresponding interrupt request bit before enabling an interrupt.

Either of the interrupt control registers - 0048$_{16}$, 0049$_{16}$ - has the polarity-switching bit. Be sure to set this bit to “0” to select an serial I/O as the interrupt request cause.

As for external interrupt input, an interrupt can be generated both at the rising edge and at the falling edge by setting “1” in the INTi interrupt polarity switching bit of the interrupt request cause select register (035F$_{16}$). To select both edges, set the polarity switching bit of the corresponding interrupt control register to ‘falling edge’ (“0”).

Figure 1.11.10 shows the Interrupt request cause select register.

Interrupt request cause select register

Symbol: IFSR
Address: 035F$_{16}$
When reset: 00$_{16}$

Bit symbol	Bit name	Function	R	W
IFSR0	INT0 interrupt polarity switching bit	0 : One edge 1 : Two edges	○	○
IFSR1	INT1 interrupt polarity switching bit	0 : One edge 1 : Two edges	○	○
IFSR2	INT2 interrupt polarity switching bit	0 : One edge 1 : Two edges	○	○
IFSR3	INT3 interrupt polarity switching bit	0 : One edge 1 : Two edges	○	○
IFSR4	INT4 interrupt polarity switching bit	0 : One edge 1 : Two edges	○	○
IFSR5	INT5 interrupt polarity switching bit	0 : One edge 1 : Two edges	○	○
IFSR6	Interrupt request cause select bit	0 : SIO3 1 : INT4	○	○
IFSR7	Interrupt request cause select bit	0 : SIO4 1 : INT5	○	○

Figure 1.11.10. Interrupt request cause select register

NMI Interrupt

An NMI interrupt is generated when the input to the P85/NMI pin changes from “H” to “L”. The NMI interrupt is a non-maskable external interrupt. The pin level can be checked in the port P85 register (bit 5 at address 03F016).

This pin cannot be used as a normal port input.

Key Input Interrupt

If the direction register of any of P104 to P107 is set for input and a falling edge is input to that port, a key input interrupt is generated. A key input interrupt can also be used as a key-on wakeup function for cancelling the wait mode or stop mode. However, if you intend to use the key input interrupt, do not use P104 to P107 as A-D input ports. Figure 1.11.11 shows the block diagram of the key input interrupt. Note that if an “L” level is input to any pin that has not been disabled for input, inputs to the other pins are not detected as an interrupt.

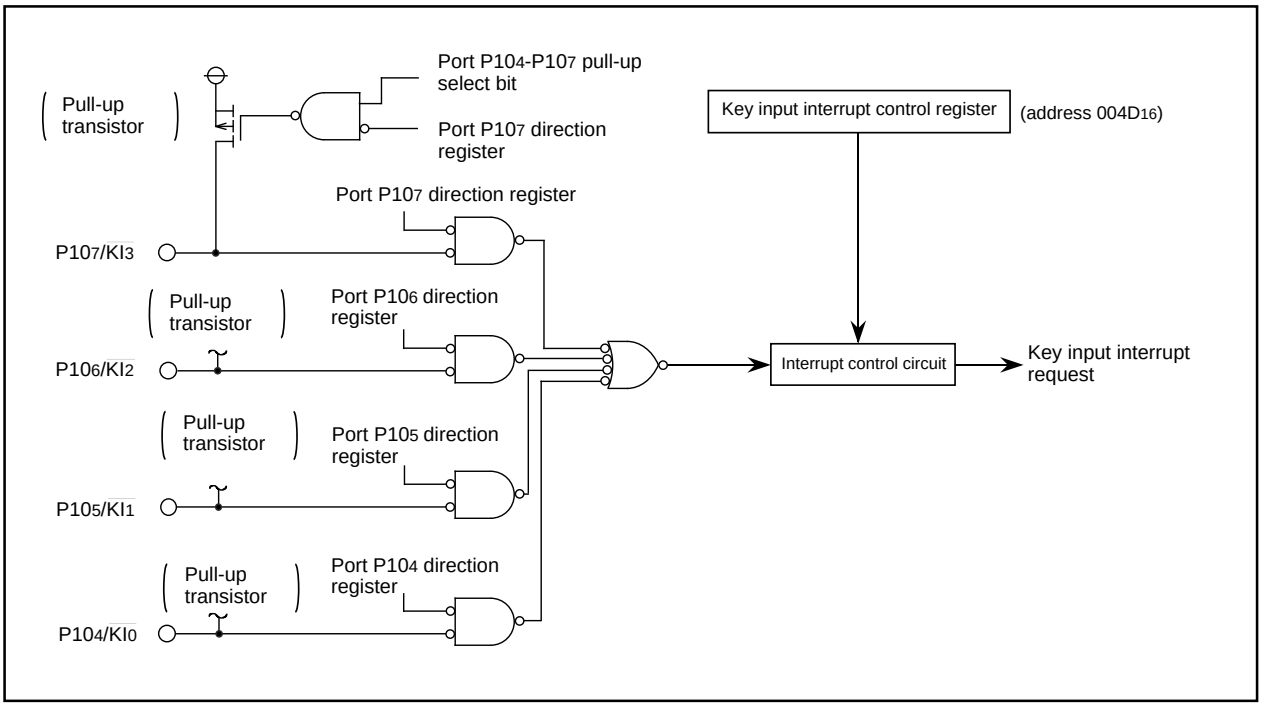


Figure 1.11.11. Block diagram of key input interrupt

Address Match Interrupt

An address match interrupt is generated when the address match interrupt address register contents match the program counter value. Two address match interrupts can be set, each of which can be enabled and disabled by an address match interrupt enable bit. Address match interrupts are not affected by the interrupt enable flag (I flag) and processor interrupt priority level (IPL). For an address match interrupt, the value of the program counter (PC) that is saved to the stack area varies depending on the instruction being executed. Note that when using the external data bus in width of 8 bits, the address match interrupt cannot be used for external area.

Figure 1.11.12 shows the address match interrupt-related registers.

Address match interrupt enable register

b7 b6 b5 b4 b3 b2 b1 b0

Symbol	Address	When reset
AIER	0009_{16}	$XXXXXX00_2$

Bit symbol	Bit name	Function	R	W
AIER0	Address match interrupt 0 enable bit	0 : Interrupt disabled 1 : Interrupt enabled	○	○
AIER1	Address match interrupt 1 enable bit	0 : Interrupt disabled 1 : Interrupt enabled	○	○
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be indeterminated.			—	—

Address match interrupt register i (i = 0, 1)

(b23) b7 (b19) b3 (b16)(b15) b0 b7 (b8) b0 b7 b0

Symbol	Address	When reset
RMAD0	0012_{16} to 0010_{16}	$X00000_{16}$
RMAD1	0016_{16} to 0014_{16}	$X00000_{16}$

Function	Values that can be set	R	W
Address setting register for address match interrupt	00000_{16} to $FFFFF_{16}$	○	○
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be indeterminated.		—	—

Figure 1.11.12. Address match interrupt-related registers

Precautions for Interrupts

(1) Reading address 00000$_{16}$

- When maskable interrupt is occurred, CPU reads the interrupt information (the interrupt number and interrupt request level) in the interrupt sequence.
  The interrupt request bit of the certain interrupt written in address 00000$_{16}$ will then be set to "0".
  Even if the address 00000$_{16}$ is read out by software, "0" is set to the enabled highest priority interrupt source request bit. Therefore interrupt can be canceled and unexpected interrupt can occur.
  Do not read address 00000$_{16}$ by software.

(2) Setting the stack pointer

- The value of the stack pointer immediately after reset is initialized to 0000$_{16}$. Accepting an interrupt before setting a value in the stack pointer may become a factor of runaway. Be sure to set a value in the stack pointer before accepting an interrupt. When using the $\overline{\text{NMI}}$ interrupt, initialize the stack pointer at the beginning of a program. Concerning the first instruction immediately after reset, generating any interrupts including the $\overline{\text{NMI}}$ interrupt is prohibited.

(3) The $\overline{\text{NMI}}$ interrupt

- The $\overline{\text{NMI}}$ interrupt can not be disabled. Be sure to connect $\overline{\text{NMI}}$ pin to Vcc via a pull-up resistor if unused. Be sure to work on it.
- The $\overline{\text{NMI}}$ pin also serves as P8$_5$, which is exclusively input. Reading the contents of the P8 register allows reading the pin value. Use the reading of this pin only for establishing the pin level at the time when the $\overline{\text{NMI}}$ interrupt is input.
- Do not reset the CPU with the input to the $\overline{\text{NMI}}$ pin being in the "L" state.
- Do not attempt to go into stop mode with the input to the $\overline{\text{NMI}}$ pin being in the "L" state. With the input to the $\overline{\text{NMI}}$ being in the "L" state, the CM10 is fixed to "0", so attempting to go into stop mode is turned down.
- Do not attempt to go into wait mode with the input to the $\overline{\text{NMI}}$ pin being in the "L" state. With the input to the $\overline{\text{NMI}}$ pin being in the "L" state, the CPU stops but the oscillation does not stop, so no power is saved. In this instance, the CPU is returned to the normal state by a later interrupt.
- Signals input to the $\overline{\text{NMI}}$ pin require an "L" level of 1 clock or more, from the operation clock of the CPU.

(4) External interrupt

- Either an "L" level or an "H" level of at least 250 ns width is necessary for the signal input to pins $\overline{\text{INT}_0}$ through $\overline{\text{INT}_5}$ regardless of the CPU operation clock.
- When the polarity of the $\overline{\text{INT}_0}$ to $\overline{\text{INT}_5}$ pins is changed, the interrupt request bit is sometimes set to "1". After changing the polarity, set the interrupt request bit to "0". Figure 1.11.13 shows the procedure for changing the $\overline{\text{INT}}$ interrupt generate factor.

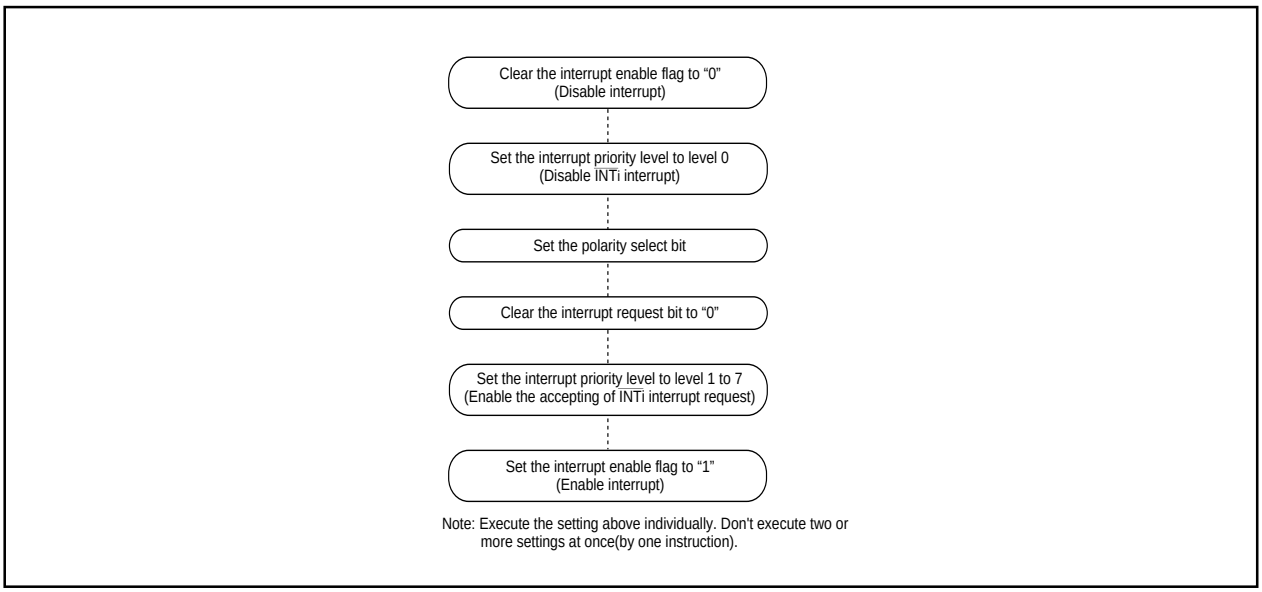


Figure 1.11.13. Switching condition of $\overline{\text{INT}}$ interrupt request

(5) Rewrite the interrupt control register

• To rewrite the interrupt control register, do so at a point that does not generate the interrupt request for that register. If there is possibility of the interrupt request occur, rewrite the interrupt control register after the interrupt is disabled. The program examples are described as follow:

Example 1:

```
INT_SWITCH1:
    FCLR    I               ; Disable interrupts.
    AND.B   #00h, 0055h     ; Clear TA0IC int. priority level and int. request bit.
    NOP                     ; Four NOP instructions are required when using HOLD function.
    NOP
    FSET    I               ; Enable interrupts.
```

Example 2:

```
INT_SWITCH2:
    FCLR    I               ; Disable interrupts.
    AND.B   #00h, 0055h     ; Clear TA0IC int. priority level and int. request bit.
    MOV.W   MEM, R0         ; Dummy read.
    FSET    I               ; Enable interrupts.
```

Example 3:

```
INT_SWITCH3:
    PUSHC   FLG             ; Push Flag register onto stack
    FCLR    I               ; Disable interrupts.
    AND.B   #00h, 0055h     ; Clear TA0IC int. priority level and int. request bit.
    POPC    FLG             ; Enable interrupts.
```

The reason why two NOP instructions (four when using the HOLD function) or dummy read are inserted before FSET I in Examples 1 and 2 is to prevent the interrupt enable flag I from being set before the interrupt control register is rewritten due to effects of the instruction queue.

• When a instruction to rewrite the interrupt control register is executed but the interrupt is disabled, the interrupt request bit is not set sometimes even if the interrupt request for that register has been generated. This will depend on the instruction. If this creates problems, use the below instructions to change the register.
Instructions : AND, OR, BCLR, BSET

Watchdog Timer

The watchdog timer has the function of detecting when the program is out of control. Therefore, we recommend using the watchdog timer to improve reliability of a system. The watchdog timer is a 15-bit counter which down-counts the clock derived by dividing the BCLK using the prescaler. A watchdog timer interrupt is generated when an underflow occurs in the watchdog timer. When X_{IN} is selected for the BCLK, bit 7 of the watchdog timer control register (address $000F_{16}$) selects the prescaler division ratio (by 16 or by 128). When X_{CIN} is selected as the BCLK, the prescaler is set for division by 2 regardless of bit 7 of the watchdog timer control register (address $000F_{16}$). Thus the watchdog timer's period can be calculated as given below. The watchdog timer's period is, however, subject to an error due to the prescaler.

With X_{IN} chosen for BCLK

$$\text{Watchdog timer period} = \frac{\text{prescaler dividing ratio (16 or 128) X watchdog timer count (32768)}}{\text{BCLK}}$$

With X_{CIN} chosen for BCLK

$$\text{Watchdog timer period} = \frac{\text{prescaler dividing ratio (2) X watchdog timer count (32768)}}{\text{BCLK}}$$

For example, suppose that BCLK runs at 16 MHz and that 16 has been chosen for the dividing ratio of the prescaler, then the watchdog timer's period becomes approximately 32.8 ms.

The watchdog timer is initialized by writing to the watchdog timer start register (address $000E_{16}$) and when a watchdog timer interrupt request is generated. The prescaler is initialized only when the microcomputer is reset. After a reset is cancelled, the watchdog timer and prescaler are both stopped. The count is started by writing to the watchdog timer start register (address $000E_{16}$). In stop mode, wait mode and hold state, the watchdog timer and prescaler are stopped. Counting is resumed from the held value when the modes or state are released.

Figure 1.12.1 shows the block diagram of the watchdog timer. Figure 1.12.2 shows the watchdog timer-related registers.

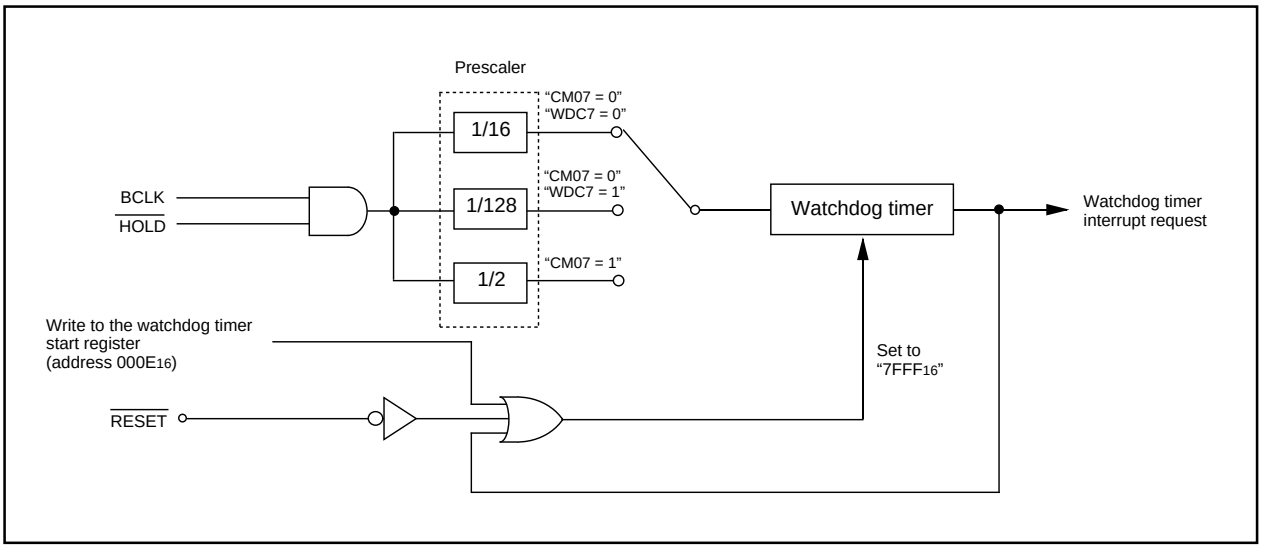


Figure 1.12.1. Block diagram of watchdog timer

Watchdog timer control register

b7	b6	b5	b4	b3	b2	b1	b0
	0	0					

Symbol	Address	When reset
WDC	$000F_{16}$	$000XXXXX_2$

Bit symbol	Bit name	Function	R	W
High-order bit of watchdog timer			○	×
Reserved bit		Must always be set to “0”	○	○
Reserved bit		Must always be set to “0”	○	○
WDC7	Prescaler select bit	0 : Divided by 16 1 : Divided by 128	○	○

Watchdog timer start register

b7	b0

Symbol	Address	When reset
WDTS	$000E_{16}$	Indeterminate

Function	R	W
The watchdog timer is initialized and starts counting after a write instruction to this register. The watchdog timer value is always initialized to “$7FFF_{16}$” regardless of whatever value is written.	×	○

Figure 1.12.2. Watchdog timer control and start registers

DMAC

This microcomputer has two DMAC (direct memory access controller) channels that allow data to be sent to memory without using the CPU. DMAC shares the same data bus with the CPU. The DMAC is given a higher right of using the bus than the CPU, which leads to working the cycle stealing method. On this account, the operation from the occurrence of DMA transfer request signal to the completion of 1-word (16-bit) or 1-byte (8-bit) data transfer can be performed at high speed. Figure 1.13.1 shows the block diagram of the DMAC. Table 1.13.1 shows the DMAC specifications. Figures 1.13.2 to 1.13.4 show the registers used by the DMAC.

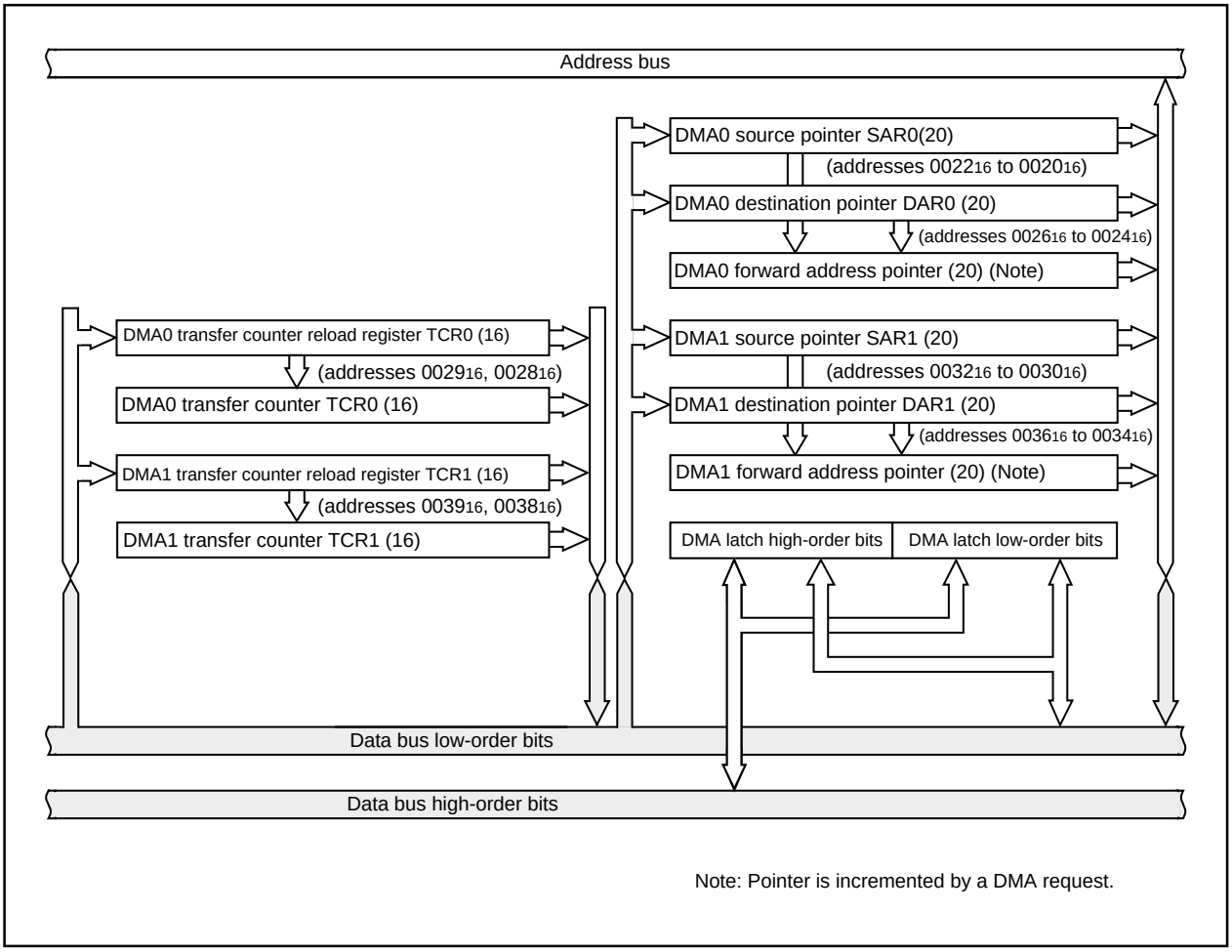


Figure 1.13.1. Block diagram of DMAC

Either a write signal to the software DMA request bit or an interrupt request signal is used as a DMA transfer request signal. But the DMA transfer is affected neither by the interrupt enable flag (I flag) nor by the interrupt priority level. The DMA transfer doesn't affect any interrupts either.

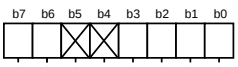
If the DMAC is active (the DMA enable bit is set to 1), data transfer starts every time a DMA transfer request signal occurs. If the cycle of the occurrences of DMA transfer request signals is higher than the DMA transfer cycle, there can be instances in which the number of transfer requests doesn't agree with the number of transfers. For details, see the description of the DMA request bit.

Table 1.13.1. DMAC specifications

Item	Specification
No. of channels	2 (cycle steal method)
Transfer memory space	• From any address in the 1M bytes space to a fixed address • From a fixed address to any address in the 1M bytes space • From a fixed address to a fixed address (Note that DMA-related registers [0020$_{16}$ to 003F$_{16}$] cannot be accessed)
Maximum No. of bytes transferred	128K bytes (with 16-bit transfers) or 64K bytes (with 8-bit transfers)
DMA request factors (Note)	Falling edge of $\overline{INT0}$ or $\overline{INT1}$ or both edge Timer A0 to timer A4 interrupt requests Timer B0 to timer B5 interrupt requests UART0 transfer and reception interrupt requests UART1 transfer and reception interrupt requests UART2 transfer and reception interrupt requests Serial I/O3, 4 interrpt requests A-D conversion interrupt requests Software triggers
Channel priority	DMA0 takes precedence if DMA0 and DMA1 requests are generated simultaneously
Transfer unit	8 bits or 16 bits
Transfer address direction	forward/fixed (forward direction cannot be specified for both source and destination simultaneously)
Transfer mode	• Single transfer mode After the transfer counter underflows, the DMA enable bit turns to "0", and the DMAC turns inactive • Repeat transfer mode After the transfer counter underflows, the value of the transfer counter reload register is reloaded to the transfer counter. The DMAC remains active unless a "0" is written to the DMA enable bit.
DMA interrupt request generation timing	When an underflow occurs in the transfer counter
Active	When the DMA enable bit is set to "1", the DMAC is active. When the DMAC is active, data transfer starts every time a DMA transfer request signal occurs.
Inactive	• When the DMA enable bit is set to "0", the DMAC is inactive. • After the transfer counter underflows in single transfer mode
Reload timing for forward address pointer and transfer counter	At the time of starting data transfer immediately after turning the DMAC active, the value of one of source pointer and destination pointer - the one specified for the forward direction - is reloaded to the forward direction address pointer, and the value of the transfer counter reload register is reloaded to the transfer counter.
Writing to register	Registers specified for forward direction transfer are always write enabled. Registers specified for fixed address transfer are write-enabled when the DMA enable bit is "0".
Reading the register	Can be read at any time. However, when the DMA enable bit is "1", reading the register set up as the forward register is the same as reading the value of the forward address pointer.

Note: DMA transfer is not effective to any interrupt. DMA transfer is affected neither by the interrupt enable flag (I flag) nor by the interrupt priority level.

DMA0 request cause select register



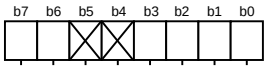
b7	b6	b5	b4	b3	b2	b1	b0
		X	X				

Symbol	Address	When reset
DM0SL	$03B8_{16}$	00_{16}

Bit symbol	Bit name	Function	R	W
DSEL0	DMA request cause select bit	b3 b2 b1 b0 0 0 0 0 : Falling edge of $\overline{\text{INT0}}$ pin 0 0 0 1 : Software trigger 0 0 1 0 : Timer A0 0 0 1 1 : Timer A1 0 1 0 0 : Timer A2 0 1 0 1 : Timer A3 0 1 1 0 : Timer A4 (DMS=0) /two edges of $\overline{\text{INT0}}$ pin (DMS=1) 0 1 1 1 : Timer B0 (DMS=0) Timer B3 (DMS=1) 1 0 0 0 : Timer B1 (DMS=0) Timer B4 (DMS=1) 1 0 0 1 : Timer B2 (DMS=0) Timer B5 (DMS=1) 1 0 1 0 : UART0 transmit 1 0 1 1 : UART0 receive 1 1 0 0 : UART2 transmit 1 1 0 1 : UART2 receive 1 1 1 0 : A-D conversion 1 1 1 1 : UART1 transmit	○	○
DSEL1			○	○
DSEL2			○	○
DSEL3			○	○
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be “0”.			–	–
DMS	DMA request cause expansion select bit	0 : Normal 1 : Expanded cause	○	○
DSR	Software DMA request bit	If software trigger is selected, a DMA request is generated by setting this bit to “1” (When read, the value of this bit is always “0”)	○	○

Figure 1.13.2. DMAC register (1)

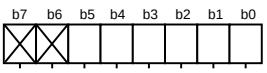
DMA1 request cause select register



Symbol	Address	When reset
DM1SL	03BA16	0016

Bit symbol	Bit name	Function	R	W
DSEL0	DMA request cause select bit	b3 b2 b1 b0 0 0 0 0 : Falling edge of INT1 pin 0 0 0 1 : Software trigger 0 0 1 0 : Timer A0 0 0 1 1 : Timer A1 0 1 0 0 : Timer A2 0 1 0 1 : Timer A3(DMS=0) /serial I/O3 (DMS=1) 0 1 1 0 : Timer A4 (DMS=0) /serial I/O4 (DMS=1) 0 1 1 1 : Timer B0 (DMS=0) /two edges of INT1 (DMS=1) 1 0 0 0 : Timer B1 1 0 0 1 : Timer B2 1 0 1 0 : UART0 transmit 1 0 1 1 : UART0 receive 1 1 0 0 : UART2 transmit 1 1 0 1 : UART2 receive 1 1 1 0 : A-D conversion 1 1 1 1 : UART1 receive	○	○
DSEL1			○	○
DSEL2			○	○
DSEL3			○	○
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be “0”.			–	–
DMS	DMA request cause expansion select bit	0 : Normal 1 : Expanded cause	○	○
DSR	Software DMA request bit	If software trigger is selected, a DMA request is generated by setting this bit to “1” (When read, the value of this bit is always “0”)	○	○

DMAi control register



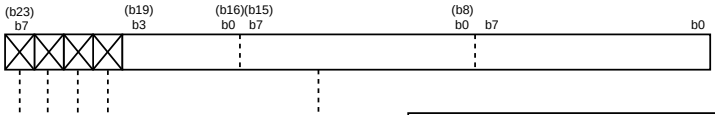
Symbol	Address	When reset
DMiCON(i=0,1)	002C16, 003C16	00000X002

Bit symbol	Bit name	Function	R	W
DMBIT	Transfer unit bit select bit	0 : 16 bits 1 : 8 bits	○	○
DMASL	Repeat transfer mode select bit	0 : Single transfer 1 : Repeat transfer	○	○
DMAS	DMA request bit (Note 1)	0 : DMA not requested 1 : DMA requested	○	○ (Note 2)
DMAE	DMA enable bit	0 : Disabled 1 : Enabled	○	○
DSD	Source address direction select bit (Note 3)	0 : Fixed 1 : Forward	○	○
DAD	Destination address direction select bit (Note 3)	0 : Fixed 1 : Forward	○	○
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be “0”.			–	–

Note 1: DMA request can be cleared by resetting the bit.
Note 2: This bit can only be set to “0”.
Note 3: Source address direction select bit and destination address direction select bit cannot be set to “1” simultaneously.

Figure 1.13.3. DMAC register (2)

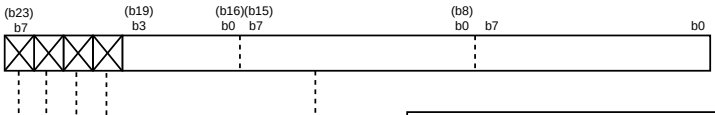
DMAi source pointer (i = 0, 1)



Symbol	Address	When reset
SAR0	0022_{16} to 0020_{16}	Indeterminate
SAR1	0032_{16} to 0030_{16}	Indeterminate

Function	Transfer address specification	R	W
• Source pointer Stores the source address	00000_{16} to $FFFFF_{16}$	○	○
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be “0”.		–	–

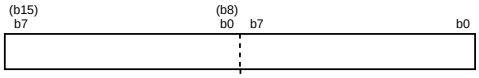
DMAi destination pointer (i = 0, 1)



Symbol	Address	When reset
DAR0	0026_{16} to 0024_{16}	Indeterminate
DAR1	0036_{16} to 0034_{16}	Indeterminate

Function	Transfer address specification	R	W
• Destination pointer Stores the destination address	00000_{16} to $FFFFF_{16}$	○	○
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be “0”.		–	–

DMAi transfer counter (i = 0, 1)



Symbol	Address	When reset
TCR0	0029_{16}, 0028_{16}	Indeterminate
TCR1	0039_{16}, 0038_{16}	Indeterminate

Function	Transfer count specification	R	W
• Transfer counter Set a value one less than the transfer count	0000_{16} to $FFFF_{16}$	○	○

Figure 1.13.4. DMAC register (3)

(1) Transfer cycle

The transfer cycle consists of the bus cycle in which data is read from memory or from the SFR area (source read) and the bus cycle in which the data is written to memory or to the SFR area (destination write). The number of read and write bus cycles depends on the source and destination addresses. In memory expansion mode and microprocessor mode, the number of read and write bus cycles also depends on the level of the BYTE pin. Also, the bus cycle itself is longer when software waits are inserted.

(a) Effect of source and destination addresses

When 16-bit data is transferred on a 16-bit data bus, and the source and destination both start at odd addresses, there are one more source read cycle and destination write cycle than when the source and destination both start at even addresses.

(b) Effect of BYTE pin level

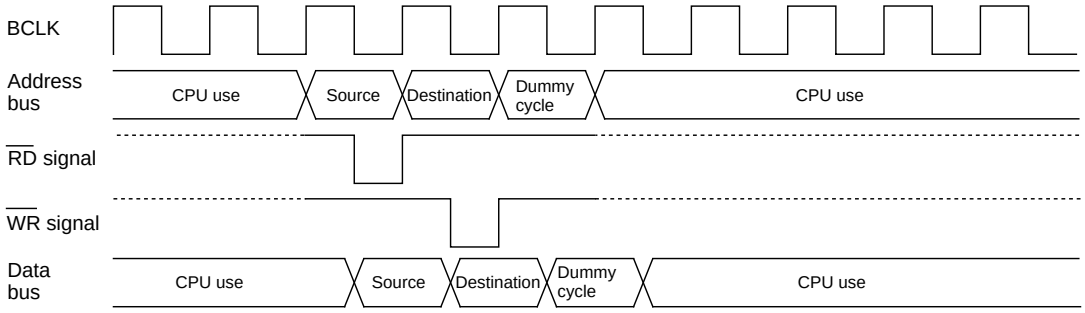
When transferring 16-bit data over an 8-bit data bus (BYTE pin = “H”) in memory expansion mode and microprocessor mode, the 16 bits of data are sent in two 8-bit blocks. Therefore, two bus cycles are required for reading the data and two are required for writing the data. Also, in contrast to when the CPU accesses internal memory, when the DMAC accesses internal memory (internal ROM, internal RAM, and SFR), these areas are accessed using the data size selected by the BYTE pin.

(c) Effect of software wait

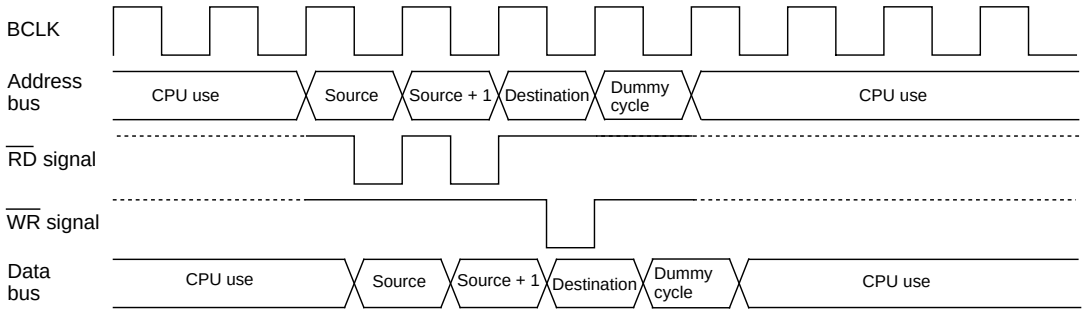
When the SFR area or a memory area with a software wait is accessed, the number of cycles is increased for the wait by 1 bus cycle. The length of the cycle is determined by BCLK.

Figure 1.13.5 shows the example of the transfer cycles for a source read. For convenience, the destination write cycle is shown as one cycle and the source read cycles for the different conditions are shown. In reality, the destination write cycle is subject to the same conditions as the source read cycle, with the transfer cycle changing accordingly. When calculating the transfer cycle, remember to apply the respective conditions to both the destination write cycle and the source read cycle. For example (2) in Figure 1.13.5, if data is being transferred in 16-bit units on an 8-bit bus, two bus cycles are required for both the source read cycle and the destination write cycle.

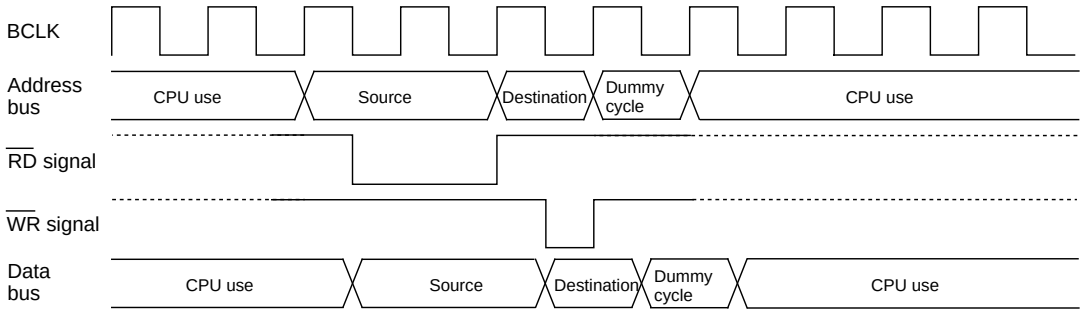
(1) 8-bit transfers
16-bit transfers and the source address is even.



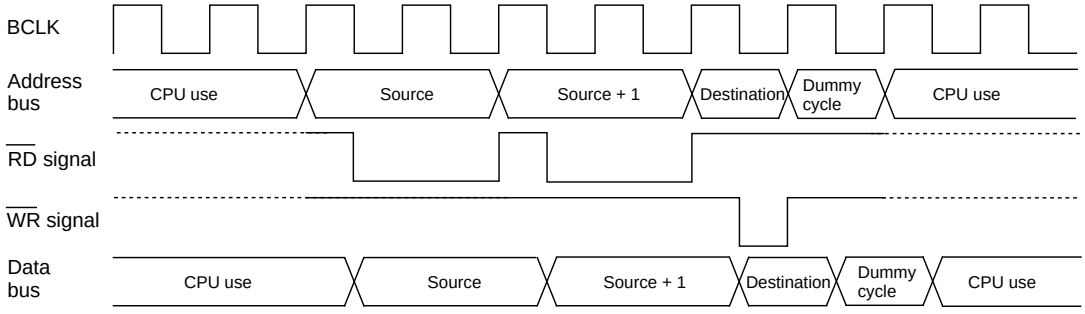
(2) 16-bit transfers and the source address is odd
Transferring 16-bit data on an 8-bit data bus (In this case, there are also two destination cycles).



(3) One wait is inserted into the source read under the conditions in (1)



(4) One wait is inserted into the source read under the conditions in (2)
(When 16-bit data is transferred on an 8-bit data bus, there are two destination cycles).



Note: The same timing changes occur with the respective conditions at the destination as at the source.

Figure 1.13.5. Example of the transfer cycles for a source read

(2) DMAC transfer cycles

Any combination of even or odd transfer read and write addresses is possible. Table 1.13.2 shows the number of DMAC transfer cycles.

The number of DMAC transfer cycles can be calculated as follows:

No. of transfer cycles per transfer unit = No. of read cycles x j + No. of write cycles x k

Table 1.13.2. No. of DMAC transfer cycles

Transfer unit	Bus width	Access address	Single-chip mode		Memory expansion mode Microprocessor mode	
			No. of read cycles	No. of write cycles	No. of read cycles	No. of write cycles
8-bit transfers (DMBIT= “1”)	16-bit (BYTE= “L”)	Even	1	1	1	1
		Odd	1	1	1	1
	8-bit (BYTE = “H”)	Even	—	—	1	1
		Odd	—	—	1	1
16-bit transfers (DMBIT= “0”)	16-bit (BYTE = “L”)	Even	1	1	1	1
		Odd	2	2	2	2
	8-bit (BYTE = “H”)	Even	—	—	2	2
		Odd	—	—	2	2

Coefficient j, k

Internal memory			External memory		
Internal ROM/RAM No wait	Internal ROM/RAM With wait	SFR area	Separate bus No wait	Separate bus With wait	Multiplex bus
1	2	2	1	2	3

DMA enable bit

Setting the DMA enable bit to “1” makes the DMAC active. The DMAC carries out the following operations at the time data transfer starts immediately after DMAC is turned active.

(1) Reloads the value of one of the source pointer and the destination pointer - the one specified for the forward direction - to the forward direction address pointer.

(2) Reloads the value of the transfer counter reload register to the transfer counter.

Thus overwriting “1” to the DMA enable bit with the DMAC being active carries out the operations given above, so the DMAC operates again from the initial state at the instant “1” is overwritten to the DMA enable bit.

DMA request bit

The DMAC can generate a DMA transfer request signal triggered by a factor chosen in advance out of DMA request factors for each channel.

DMA request factors include the following.

* Factors effected by using the interrupt request signals from the built-in peripheral functions and software DMA factors (internal factors) effected by a program.
* External factors effected by utilizing the input from external interrupt signals.

For the selection of DMA request factors, see the descriptions of the DMAi factor selection register.

The DMA request bit turns to “1” if the DMA transfer request signal occurs regardless of the DMAC's state (regardless of whether the DMA enable bit is set to “1” or “0”). It turns to “0” immediately before data transfer starts.

In addition, it can be set to “0” by use of a program, but cannot be set to “1”.

There can be instances in which a change in DMA request factor selection bit causes the DMA request bit to turn to “1”. So be sure to set the DMA request bit to “0” after the DMA request factor selection bit is changed.

The DMA request bit turns to “1” if a DMA transfer request signal occurs, and turns to “0” immediately before data transfer starts. If the DMAC is active, data transfer starts immediately, so the value of the DMA request bit, if read by use of a program, turns out to be “0” in most cases. To examine whether the DMAC is active, read the DMA enable bit.

Here follows the timing of changes in the DMA request bit.

(1) Internal factors

Except the DMA request factors triggered by software, the timing for the DMA request bit to turn to “1” due to an internal factor is the same as the timing for the interrupt request bit of the interrupt control register to turn to “1” due to several factors.

Turning the DMA request bit to “0” due to an internal factor is timed to be effected immediately before the transfer starts.

(2) External factors

An external factor is a factor caused to occur by the leading edge of input from the $\overline{\text{INTi}}$ pin (i depends on which DMAC channel is used).

Selecting the $\overline{\text{INTi}}$ pins as external factors using the DMA request factor selection bit causes input from these pins to become the DMA transfer request signals.

The timing for the DMA request bit to turn to “1” when an external factor is selected synchronizes with the signal's edge applicable to the function specified by the DMA request factor selection bit (synchronizes with the trailing edge of the input signal to each $\overline{\text{INTi}}$ pin, for example).

With an external factor selected, the DMA request bit is timed to turn to “0” immediately before data transfer starts similarly to the state in which an internal factor is selected.

(3) The priorities of channels and DMA transfer timing

If a DMA transfer request signal falls on a single sampling cycle (a sampling cycle means one period from the leading edge to the trailing edge of BCLK), the DMA request bits of applicable channels concurrently turn to “1”. If the channels are active at that moment, DMA0 is given a high priority to start data transfer. When DMA0 finishes data transfer, it gives the bus right to the CPU. When the CPU finishes single bus access, then DMA1 starts data transfer and gives the bus right to the CPU.

An example in which DMA transfer is carried out in minimum cycles at the time when DMA transfer request signals due to external factors concurrently occur.

Figure 1.13.6 An example of DMA transfer effected by external factors.

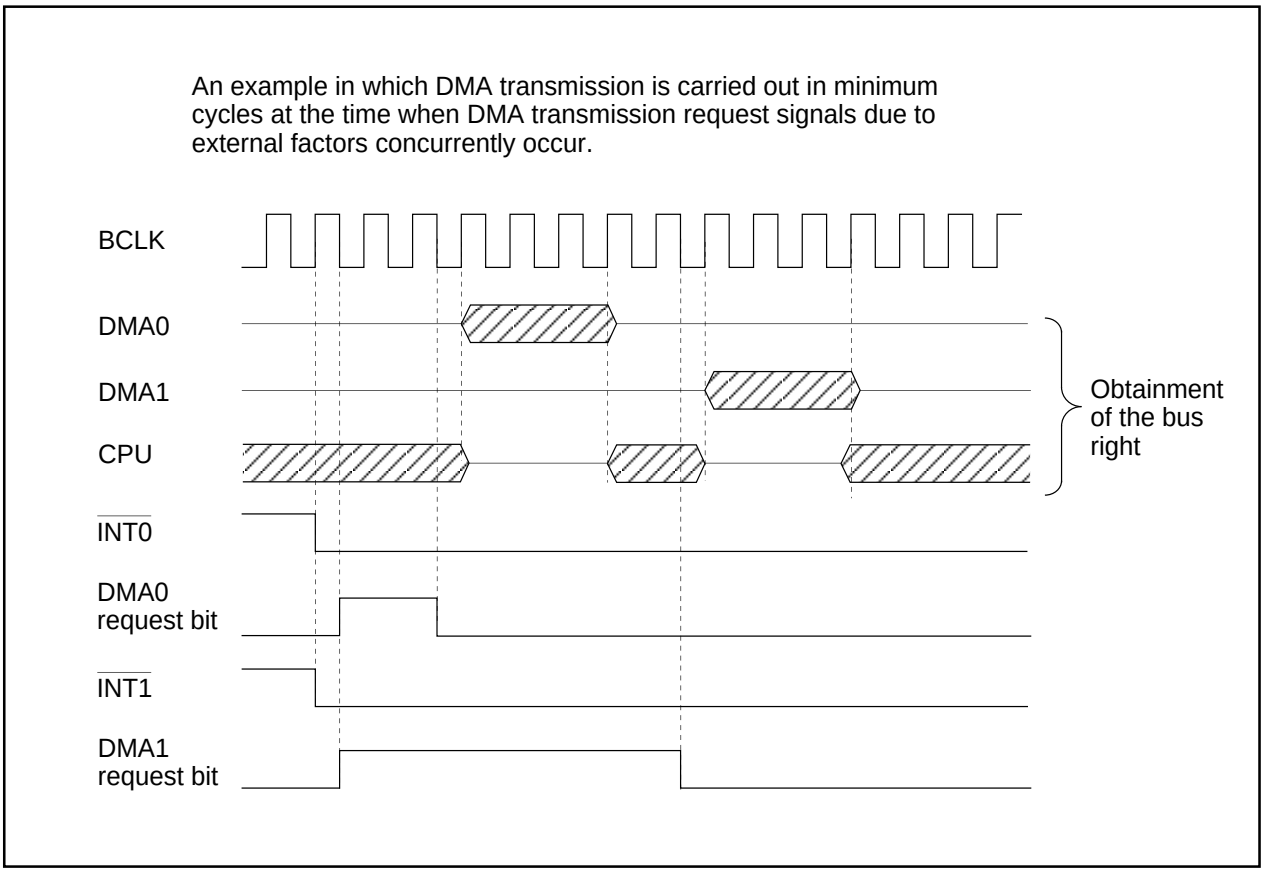
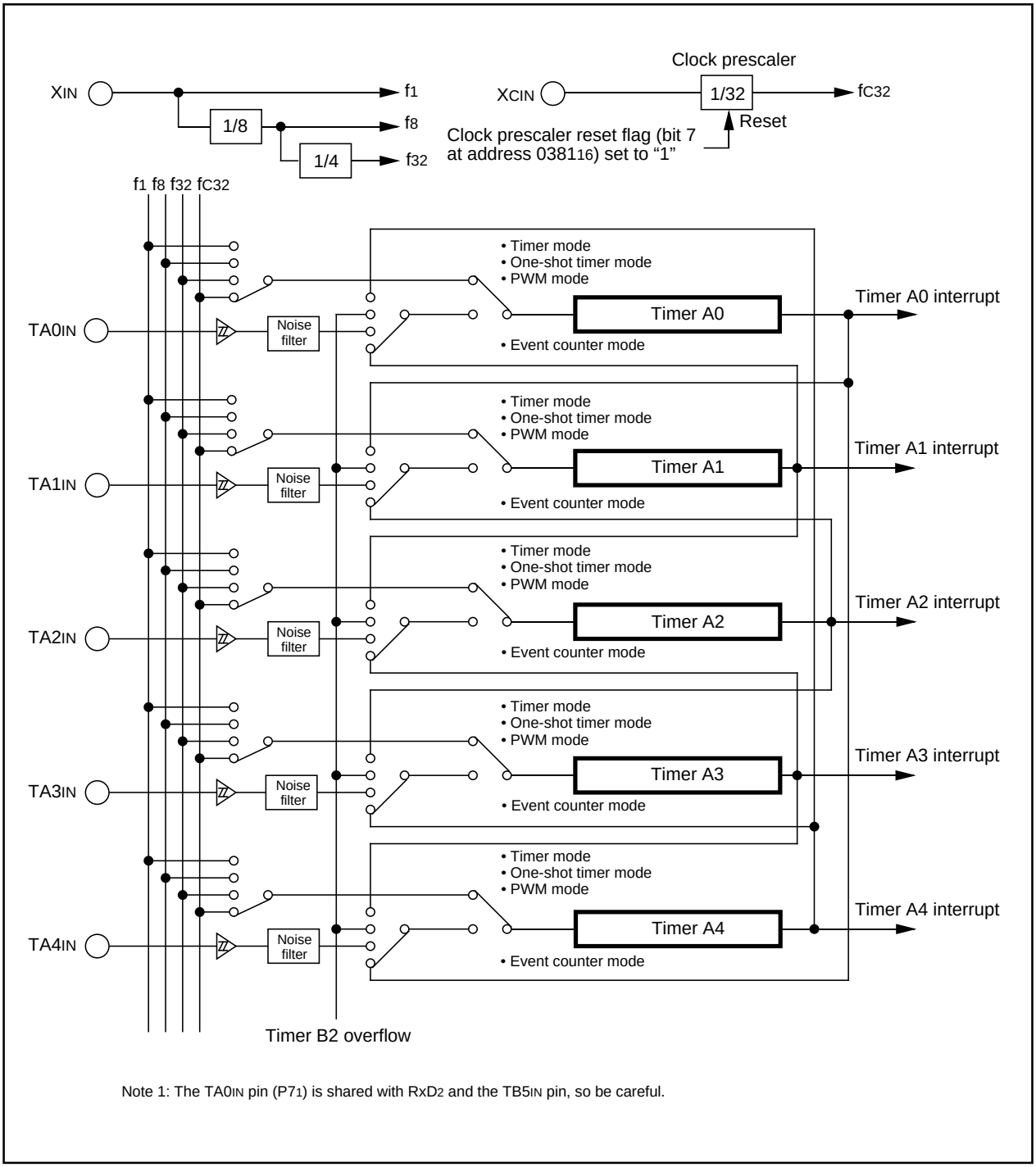


Figure 1.13.6. An example of DMA transfer effected by external factors

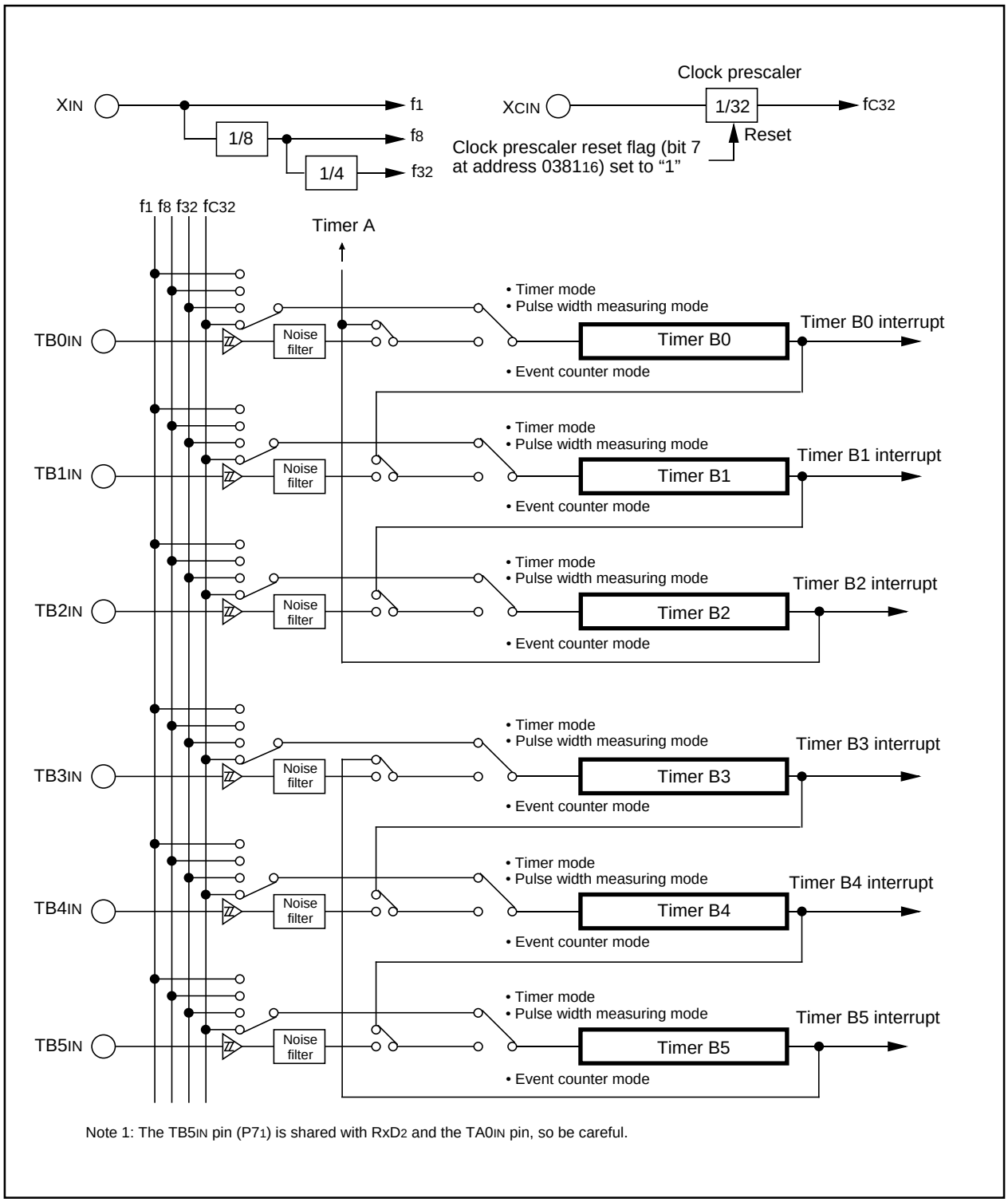
Timer

There are eleven 16-bit timers. These timers can be classified by function into timers A (five) and timers B (six). All these timers function independently. Figures 1.14.1 and 1.14.2 show the block diagram of timers.



Note 1: The TA0IN pin (P71) is shared with RxD2 and the TB5IN pin, so be careful.

Figure 1.14.1. Timer A block diagram



Note 1: The TB5IN pin (P71) is shared with RxD2 and the TA0IN pin, so be careful.

Figure 1.14.2. Timer B block diagram

Timer A

Figure 1.14.3 shows the block diagram of timer A. Figures 1.14.4 to 1.14.6 show the timer A-related registers.

Except in event counter mode, timers A0 through A4 all have the same function. Use the timer Ai mode register (i = 0 to 4) bits 0 and 1 to choose the desired mode.

Timer A has the four operation modes listed as follows:

- Timer mode: The timer counts an internal count source.
- Event counter mode: The timer counts pulses from an external source or a timer over flow.
- One-shot timer mode: The timer stops counting when the count reaches "0000$_{16}$".
- Pulse width modulation (PWM) mode: The timer outputs pulses of a given width.

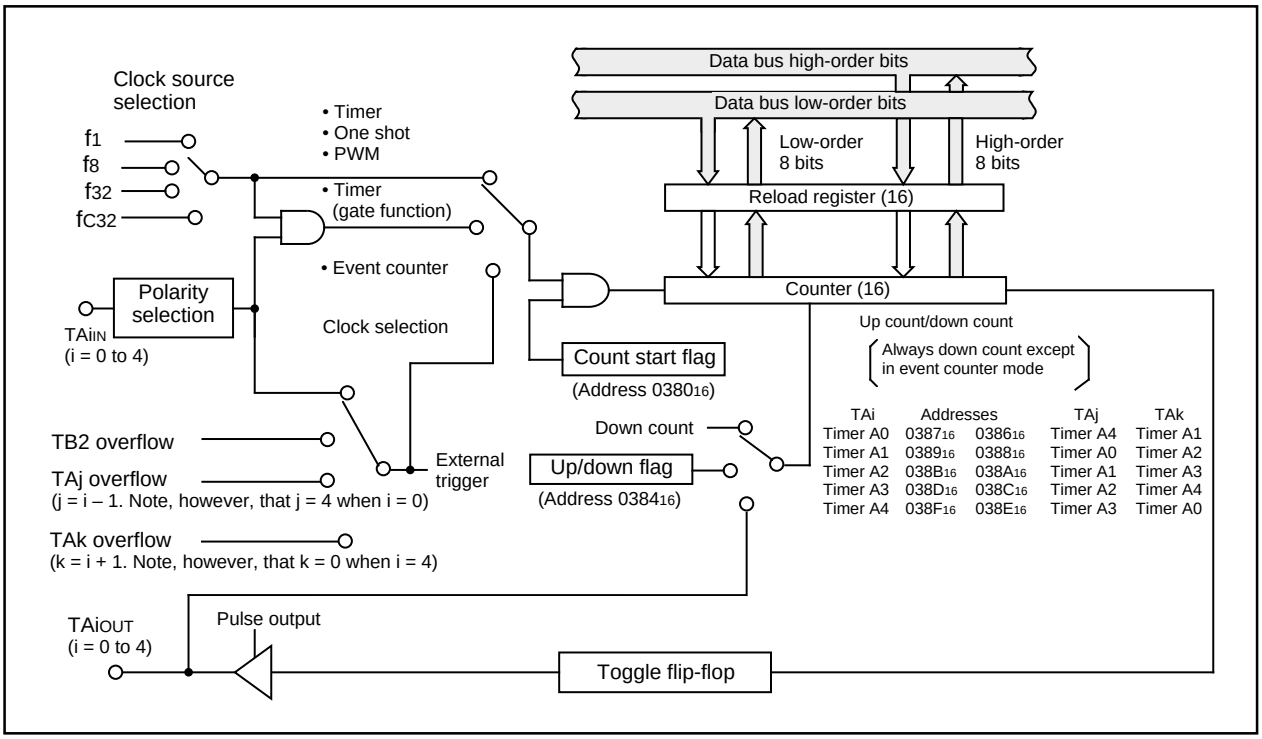


Figure 1.14.3. Block diagram of timer A

Timer Ai mode register

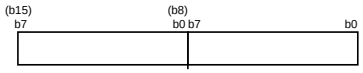
Symbol	Address	When reset
TAiMR(i=0 to 4)	0396$_{16}$ to 039A$_{16}$	00$_{16}$

b7	b6	b5	b4	b3	b2	b1	b0

Bit symbol	Bit name	Function	R	W
TMOD0	Operation mode select bit	b1 b0 0 0 : Timer mode 0 1 : Event counter mode 1 0 : One-shot timer mode 1 1 : Pulse width modulation (PWM) mode	○	○
TMOD1			○	○
MR0	Function varies with each operation mode		○	○
MR1			○	○
MR2			○	○
MR3			○	○
TCK0	Count source select bit (Function varies with each operation mode)		○	○
TCK1			○	○

Figure 1.14.4. Timer A-related registers (1)

Timer Ai register (Note 1)

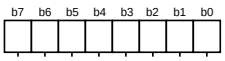


Symbol	Address	When reset
TA0	0387_{16},0386_{16}	Indeterminate
TA1	0389_{16},0388_{16}	Indeterminate
TA2	$038B_{16}$,$038A_{16}$	Indeterminate
TA3	$038D_{16}$,$038C_{16}$	Indeterminate
TA4	$038F_{16}$,$038E_{16}$	Indeterminate

Function	Values that can be set	R	W
• Timer mode Counts an internal count source	0000_{16} to $FFFF_{16}$	O	O
• Event counter mode Counts pulses from an external source or timer overflow	0000_{16} to $FFFF_{16}$	O	O
• One-shot timer mode Counts a one shot width	0000_{16} to $FFFF_{16}$ (Note 2,4)	X	O
• Pulse width modulation mode (16-bit PWM) Functions as a 16-bit pulse width modulator	0000_{16} to $FFFE_{16}$ (Note 3,4)	X	O
• Pulse width modulation mode (8-bit PWM) Timer low-order address functions as an 8-bit prescaler and high-order address functions as an 8-bit pulse width modulator	00_{16} to FE_{16} (High-order address) 00_{16} to FF_{16} (Low-order address) (Note 3,4)	X	O

Note 1: Read and write data in 16-bit units.
Note 2: When the timer Ai register is set to "0000_{16}", the counter does not operate and the timer Ai interrupt request is not generated. When the pulse is set to output, the pulse does not output from the TAiOUT pin.
Note 3: When the timer Ai register is set to "0000_{16}", the pulse width modulator does not operate and the output level of the TAiOUT pin remains "L" level, therefore the timer Ai interrupt request is not generated. This also occurs in the 8-bit pulse width modulator mode when the significant 8 high-order bits in the timer Ai register are set to "00_{16}".
Note 4: Use MOV instruction to write to this register.

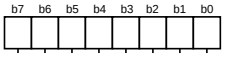
Count start flag



Symbol	Address	When reset
TABSR	0380_{16}	00_{16}

Bit symbol	Bit name	Function	R	W
TA0S	Timer A0 count start flag	0 : Stops counting 1 : Starts counting	O	O
TA1S	Timer A1 count start flag		O	O
TA2S	Timer A2 count start flag		O	O
TA3S	Timer A3 count start flag		O	O
TA4S	Timer A4 count start flag		O	O
TB0S	Timer B0 count start flag		O	O
TB1S	Timer B1 count start flag		O	O
TB2S	Timer B2 count start flag		O	O

Up/down flag (Note 1)



Symbol	Address	When reset
UDF	0384_{16}	00_{16}

Bit symbol	Bit name	Function	R	W
TA0UD	Timer A0 up/down flag	0 : Down count 1 : Up count This specification becomes valid when the up/down flag content is selected for up/down switching cause	O	O
TA1UD	Timer A1 up/down flag		O	O
TA2UD	Timer A2 up/down flag		O	O
TA3UD	Timer A3 up/down flag		O	O
TA4UD	Timer A4 up/down flag		O	O
TA2P	Timer A2 two-phase pulse signal processing select bit	0 : two-phase pulse signal processing disabled 1 : two-phase pulse signal processing enabled (Note 2) When not using the two-phase pulse signal processing function, set the select bit to "0"	X	O
TA3P	Timer A3 two-phase pulse signal processing select bit		X	O
TA4P	Timer A4 two-phase pulse signal processing select bit		X	O

Note 1: Use MOV instruction to write to this register.
Note 2: Set the TAiIN and TAiOUT pins correspondent port direction registers to "0".

Figure 1.14.5. Timer A-related registers (2)

One-shot start flag

Symbol	Address	When reset
ONSF	0382_{16}	$00X00000_2$

Bit symbol	Bit name	Function	R	W
TA0OS	Timer A0 one-shot start flag	1 : Timer start When read, the value is “0”	○	○
TA1OS	Timer A1 one-shot start flag		○	○
TA2OS	Timer A2 one-shot start flag		○	○
TA3OS	Timer A3 one-shot start flag		○	○
TA4OS	Timer A4 one-shot start flag		○	○
Nothing is assigned. In an attempt to write to this bit, write “0”. The value, if read, turns out to be indeterminate.			—	—
TA0TGL	Timer A0 event/trigger select bit	b7 b6 0 0 : Input on $TA0_{IN}$ is selected (Note) 0 1 : TB2 overflow is selected 1 0 : TA4 overflow is selected 1 1 : TA1 overflow is selected	○	○
TA0TGH			○	○

Note: Set the corresponding port direction register to “0”.

Trigger select register

Symbol	Address	When reset
TRGSR	0383_{16}	00_{16}

Bit symbol	Bit name	Function	R	W
TA1TGL	Timer A1 event/trigger select bit	b1 b0 0 0 : Input on $TA1_{IN}$ is selected (Note) 0 1 : TB2 overflow is selected 1 0 : TA0 overflow is selected 1 1 : TA2 overflow is selected	○	○
TA1TGH			○	○
TA2TGL	Timer A2 event/trigger select bit	b3 b2 0 0 : Input on $TA2_{IN}$ is selected (Note) 0 1 : TB2 overflow is selected 1 0 : TA1 overflow is selected 1 1 : TA3 overflow is selected	○	○
TA2TGH			○	○
TA3TGL	Timer A3 event/trigger select bit	b5 b4 0 0 : Input on $TA3_{IN}$ is selected (Note) 0 1 : TB2 overflow is selected 1 0 : TA2 overflow is selected 1 1 : TA4 overflow is selected	○	○
TA3TGH			○	○
TA4TGL	Timer A4 event/trigger select bit	b7 b6 0 0 : Input on $TA4_{IN}$ is selected (Note) 0 1 : TB2 overflow is selected 1 0 : TA3 overflow is selected 1 1 : TA0 overflow is selected	○	○
TA4TGH			○	○

Note: Set the corresponding port direction register to “0”.

Clock prescaler reset flag

Symbol	Address	When reset
CPSRF	0381_{16}	$0XXXXXXX_2$

Bit symbol	Bit name	Function	R	W
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be indeterminate.			—	—
CPSR	Clock prescaler reset flag	0 : No effect 1 : Prescaler is reset (When read, the value is “0”)	○	○

Figure 1.14.6. Timer A-related registers (3)

(1) Timer mode

In this mode, the timer counts an internally generated count source. (See Table 1.14.1.) Figure 1.14.7 shows the timer Ai mode register in timer mode.

Table 1.14.1. Specifications of timer mode

Item	Specification
Count source	f_1, f_8, f_{32}, f_{C32}
Count operation	• Down count • When the timer underflows, it reloads the reload register contents before continuing counting
Divide ratio	1/(n+1) n : Set value
Count start condition	Count start flag is set (= 1)
Count stop condition	Count start flag is reset (= 0)
Interrupt request generation timing	When the timer underflows
TAiIN pin function	Programmable I/O port or gate input
TAiOUT pin function	Programmable I/O port or pulse output
Read from timer	Count value can be read out by reading timer Ai register
Write to timer	• When counting stopped When a value is written to timer Ai register, it is written to both reload register and counter • When counting in progress When a value is written to timer Ai register, it is written to only reload register (Transferred to counter at next reload time)
Select function	• Gate function Counting can be started and stopped by the TAiIN pin's input signal • Pulse output function Each time the timer underflows, the TAiOUT pin's polarity is reversed

Timer Ai mode register

b7	b6	b5	b4	b3	b2	b1	b0
		0				0	0

Symbol: TAiMR(i=0 to 4) — Address: 039_{16} to $039A_{16}$ — When reset: 00_{16}

Bit symbol	Bit name	Function	R	W
TMOD0	Operation mode select bit	b1 b0 0 0 : Timer mode	○	○
TMOD1			○	○
MR0	Pulse output function select bit	0 : Pulse is not output (TAiOUT pin is a normal port pin) 1 : Pulse is output (Note 1) (TAiOUT pin is a pulse output pin)	○	○
MR1	Gate function select bit	b4 b3 0 X (Note 2): Gate function not available (TAiIN pin is a normal port pin) 1 0 : Timer counts only when TAiIN pin is held "L" (Note 3) 1 1 : Timer counts only when TAiIN pin is held "H" (Note 3)	○	○
MR2			○	○
MR3	0 (Must always be "0" in timer mode)		○	○
TCK0	Count source select bit	b7 b6 0 0 : f_1 0 1 : f_8 1 0 : f_{32} 1 1 : f_{C32}	○	○
TCK1			○	○

Note 1: The settings of the corresponding port register and port direction register are invalid.
Note 2: The bit can be "0" or "1".
Note 3: Set the corresponding port direction register to "0".

Figure 1.14.7. Timer Ai mode register in timer mode

(2) Event counter mode

In this mode, the timer counts an external signal or an internal timer's overflow. Timers A0 and A1 can count a single-phase external signal. Timers A2, A3, and A4 can count a single-phase and a two-phase external signal. Table 1.14.2 lists timer specifications when counting a single-phase external signal. Figure 1.14.8 shows the timer Ai mode register in event counter mode.

Table 1.14.3 lists timer specifications when counting a two-phase external signal. Figure 1.14.9 shows the timer Ai mode register in event counter mode.

Table 1.14.2. Timer specifications in event counter mode (when not processing two-phase pulse signal)

Item	Specification
Count source	• External signals input to TAiIN pin (effective edge can be selected by software) • TB2 overflow, TAj overflow
Count operation	• Up count or down count can be selected by external signal or software • When the timer overflows or underflows, it reloads the reload register con tents before continuing counting (Note)
Divide ratio	1/ ($FFFF_{16}$ - n + 1) for up count 1/ (n + 1) for down count n : Set value
Count start condition	Count start flag is set (= 1)
Count stop condition	Count start flag is reset (= 0)
Interrupt request generation timing	The timer overflows or underflows
TAiIN pin function	Programmable I/O port or count source input
TAiOUT pin function	Programmable I/O port, pulse output, or up/down count select input
Read from timer	Count value can be read out by reading timer Ai register
Write to timer	• When counting stopped When a value is written to timer Ai register, it is written to both reload register and counter • When counting in progress When a value is written to timer Ai register, it is written to only reload register (Transferred to counter at next reload time)
Select function	• Free-run count function Even when the timer overflows or underflows, the reload register content is not reloaded to it • Pulse output function Each time the timer overflows or underflows, the TAiOUT pin's polarity is reversed

Note: This does not apply when the free-run function is selected.

Timer Ai mode register
(When not using two-phase pulse signal processing)

b7	b6	b5	b4	b3	b2	b1	b0
		0				0	1

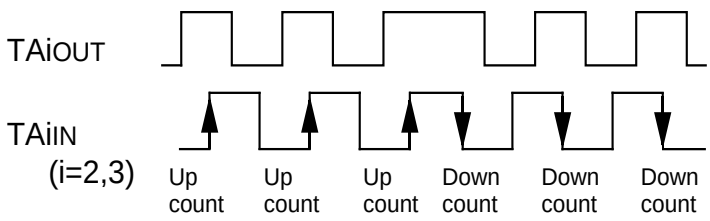
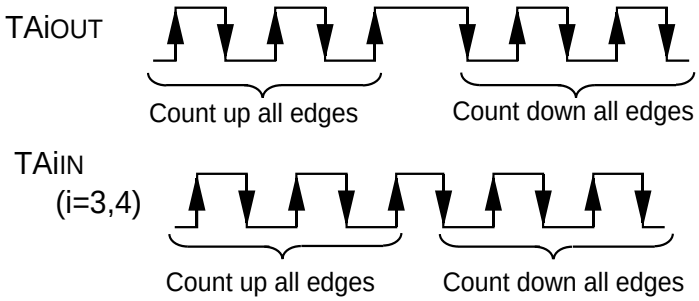
Symbol: TAiMR(i = 0 to 4) Address: 039_{16} to $039A_{16}$ When reset: 00_{16}

Bit symbol	Bit name	Function	R	W
TMOD0	Operation mode select bit	b1 b0 0 1 : Event counter mode (Note 1)	O	O
TMOD1			O	O
MR0	Pulse output function select bit	0 : Pulse is not output (TAiOUT pin is a normal port pin) 1 : Pulse is output (Note 2) (TAiOUT pin is a pulse output pin)	O	O
MR1	Count polarity select bit (Note 3)	0 : Counts external signal's falling edge 1 : Counts external signal's rising edge	O	O
MR2	Up/down switching cause select bit	0 : Up/down flag's content 1 : TAiOUT pin's input signal (Note 4)	O	O
MR3	0 (Must always be "0" in event counter mode)		O	O
TCK0	Count operation type select bit	0 : Reload type 1 : Free-run type	O	O
TCK1	Invalid when not using two-phase pulse signal processing Can be "0" or "1"		O	O

Note 1: In event counter mode, the count source is selected by the event / trigger select bit (addresses 0382_{16} and 0383_{16}).
Note 2: The settings of the corresponding port register and port direction register are invalid.
Note 3: Valid only when counting an external signal.
Note 4: When an "L" signal is input to the TAiOUT pin, the downcount is activated. When "H", the upcount is activated. Set the corresponding port direction register to "0".

Figure 1.14.8. Timer Ai mode register in event counter mode

Table 1.14.3. Timer specifications in event counter mode (when processing two-phase pulse signal with timers A2, A3, and A4)

Item	Specification
Count source	• Two-phase pulse signals input to TAiIN or TAiOUT pin
Count operation	• Up count or down count can be selected by two-phase pulse signal • When the timer overflows or underflows, the reload register content is reloaded and the timer starts over again (Note 1)
Divide ratio	1/ ($FFFF_{16}$ - n + 1) for up count 1/ (n + 1) for down count n : Set value
Count start condition	Count start flag is set (= 1)
Count stop condition	Count start flag is reset (= 0)
Interrupt request generation timing	Timer overflows or underflows
TAiIN pin function	Two-phase pulse input (Set the TAiIN pin correspondent port direction register to "0".)
TAiOUT pin function	Two-phase pulse input (Set the TAiOUT pin correspondent port direction register to "0".)
Read from timer	Count value can be read out by reading timer A2, A3, or A4 register
Write to timer	• When counting stopped When a value is written to timer A2, A3, or A4 register, it is written to both reload register and counter • When counting in progress When a value is written to timer A2, A3, or A4 register, it is written to only reload register. (Transferred to counter at next reload time.)
Select function (Note 2)	• Normal processing operation (timer A2 and timer A3) The timer counts up rising edges or counts down falling edges on the TAiIN pin when input signal on the TAiOUT pin is "H".  • Multiply-by-4 processing operation (timer A3 and timer A4) If the phase relationship is such that the TAiIN pin goes "H" when the input signal on the TAiOUT pin is "H", the timer counts up rising and falling edges on the TAiOUT and TAiIN pins. If the phase relationship is such that the TAiIN pin goes "L" when the input signal on the TAiOUT pin is "H", the timer counts down rising and falling edges on the TAiOUT and TAiIN pins. 

Note 1: This does not apply when the free-run function is selected.

Note 2: Timer A3 alone can be selected. Timer A2 is fixed to normal processing operation, and timer A4 is fixed to multiply-by-4 processing operation.

Timer Ai mode register
(When using two-phase pulse signal processing)

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

Symbol: TAiMR(i = 2 to 4) — Address: 0398_{16} to $039A_{16}$ — When reset: 00_{16}

	Bit name	Function	R	W
TMOD0	Operation mode select bit	b1 b0 0 1 : Event counter mode	○	○
TMOD1			○	○
MR0	0 (Must always be “0” when using two-phase pulse signal processing)		○	○
MR1	0 (Must always be “0” when using two-phase pulse signal processing)		○	○
MR2	1 (Must always be “1” when using two-phase pulse signal processing)		○	○
MR3	0 (Must always be “0” when using two-phase pulse signal processing)		○	○
TCK0	Count operation type select bit	0 : Reload type 1 : Free-run type	○	○
TCK1	Two-phase pulse processing operation select bit (Note 1)(Note 2)	0 : Normal processing operation 1 : Multiply-by-4 processing operation	○	○

Note 1: This bit is valid for timer A3 mode register. Timer A2 is fixed to normal processing operation, and timer A4 is fixed to multiply-by-4 processing operation.
Note 2: When performing two-phase pulse signal processing, make sure the two-phase pulse signal processing operation select bit (address 0384_{16}) is set to “1”. Also, always be sure to set the event/trigger select bit (addresses 0382_{16} and 0383_{16}) to “00”.

Figure 1.14.9. Timer Ai mode register in event counter mode

(3) One-shot timer mode

In this mode, the timer operates only once. (See Table 1.14.4.) When a trigger occurs, the timer starts up and continues operating for a given period. Figure 1.14.10 shows the timer Ai mode register in one-shot timer mode.

Table 1.14.4. Timer specifications in one-shot timer mode

Item	Specification
Count source	f_1, f_8, f_{32}, f_{C32}
Count operation	• The timer counts down • When the count reaches 0000_{16}, the timer stops counting after reloading a new count • If a trigger occurs when counting, the timer reloads a new count and restarts counting
Divide ratio	1/n n : Set value
Count start condition	• An external trigger is input • The timer overflows • The one-shot start flag is set (= 1)
Count stop condition	• A new count is reloaded after the count has reached 0000_{16} • The count start flag is reset (= 0)
Interrupt request generation timing	The count 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	When timer Ai register is read, it indicates an indeterminate value
Write to timer	• When counting stopped When a value is written to timer Ai register, it is written to both reload register and counter • When counting in progress When a value is written to timer Ai register, it is written to only reload register (Transferred to counter at next reload time)

Timer Ai mode register

b7	b6	b5	b4	b3	b2	b1	b0
		0				1	0

Symbol: TAiMR(i = 0 to 4) Address: 0396_{16} to $039A_{16}$ When reset: 00_{16}

Bit symbol	Bit name	Function	R	W
TMOD0	Operation mode select bit	b1 b0 1 0 : One-shot timer mode	○	○
TMOD1			○	○
MR0	Pulse output function select bit	0 : Pulse is not output (TAiOUT pin is a normal port pin) 1 : Pulse is output (Note 1) (TAiOUT pin is a pulse output pin)	○	○
MR1	External trigger select bit (Note 2)	0 : Falling edge of TAiIN pin's input signal (Note 3) 1 : Rising edge of TAiIN pin's input signal (Note 3)	○	○
MR2	Trigger select bit	0 : One-shot start flag is valid 1 : Selected by event/trigger select bits	○	○
MR3	0 (Must always be "0" in one-shot timer mode)		○	○
TCK0	Count source select bit	b7 b6 0 0 : f_1 0 1 : f_8 1 0 : f_{32} 1 1 : f_{C32}	○	○
TCK1			○	○

Note 1: The settings of the corresponding port register and port direction register are invalid.
Note 2: Valid only when the TAiIN pin is selected by the event/trigger select bit (addresses 0382_{16} and 0383_{16}). If timer overflow is selected, this bit can be "1" or "0".
Note 3: Set the corresponding port direction register to "0".

Figure 1.14.10. Timer Ai mode register in one-shot timer mode

(4) Pulse width modulation (PWM) mode

In this mode, the timer outputs pulses of a given width in succession. (See Table 1.14.5.) In this mode, the counter functions as either a 16-bit pulse width modulator or an 8-bit pulse width modulator. Figure 1.14.11 shows the timer Ai mode register in pulse width modulation mode. Figure 1.14.12 shows the example of how a 16-bit pulse width modulator operates. Figure 1.14.13 shows the example of how an 8-bit pulse width modulator operates.

Table 1.14.5. Timer specifications in pulse width modulation mode

Item	Specification
Count source	f_1, f_8, f_{32}, f_{C32}
Count operation	• The timer counts down (operating as an 8-bit or a 16-bit pulse width modulator) • The timer reloads a new count at a rising edge of PWM pulse and continues counting • The timer is not affected by a trigger that occurs when counting
16-bit PWM	• High level width n / fi n : Set value • Cycle time $(2^{16}-1)$ / fi fixed
8-bit PWM	• High level width n×(m+1) / fi n : values set to timer Ai register's high-order address • Cycle time $(2^{8}-1)$× (m+1) / fi m : values set to timer Ai register's low-order address
Count start condition	• External trigger is input • The timer overflows • The count start flag is set (= 1)
Count stop condition	• The count start flag is reset (= 0)
Interrupt request generation timing	PWM pulse goes "L"
TAiIN pin function	Programmable I/O port or trigger input
TAiOUT pin function	Pulse output
Read from timer	When timer Ai register is read, it indicates an indeterminate value
Write to timer	• When counting stopped When a value is written to timer Ai register, it is written to both reload register and counter • When counting in progress When a value is written to timer Ai register, it is written to only reload register (Transferred to counter at next reload time)

Timer Ai mode register

b7	b6	b5	b4	b3	b2	b1	b0
					1	1	1

Symbol	Address	When reset
TAiMR(i=0 to 4)	039_{16} to $039A_{16}$	00_{16}

Bit symbol	Bit name	Function	R	W
TMOD0	Operation mode select bit	b1 b0 1 1 : PWM mode	○	○
TMOD1			○	○
MR0	1 (Must always be "1" in PWM mode)		○	○
MR1	External trigger select bit (Note 1)	0: Falling edge of TAiIN pin's input signal (Note 2) 1: Rising edge of TAiIN pin's input signal (Note 2)	○	○
MR2	Trigger select bit	0: Count start flag is valid 1: Selected by event/trigger select bits	○	○
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	○	○
TCK0	Count source select bit	b7 b6 0 0 : f_1 0 1 : f_8 1 0 : f_{32} 1 1 : f_{C32}	○	○
TCK1			○	○

Note 1: Valid only when the TAiIN pin is selected by the event/trigger select bit (addresses 0382_{16} and 0383_{16}). If timer overflow is selected, this bit can be "1" or "0".
Note 2: Set the corresponding port direction register to "0".

Figure 1.14.11. Timer Ai mode register in pulse width modulation mode

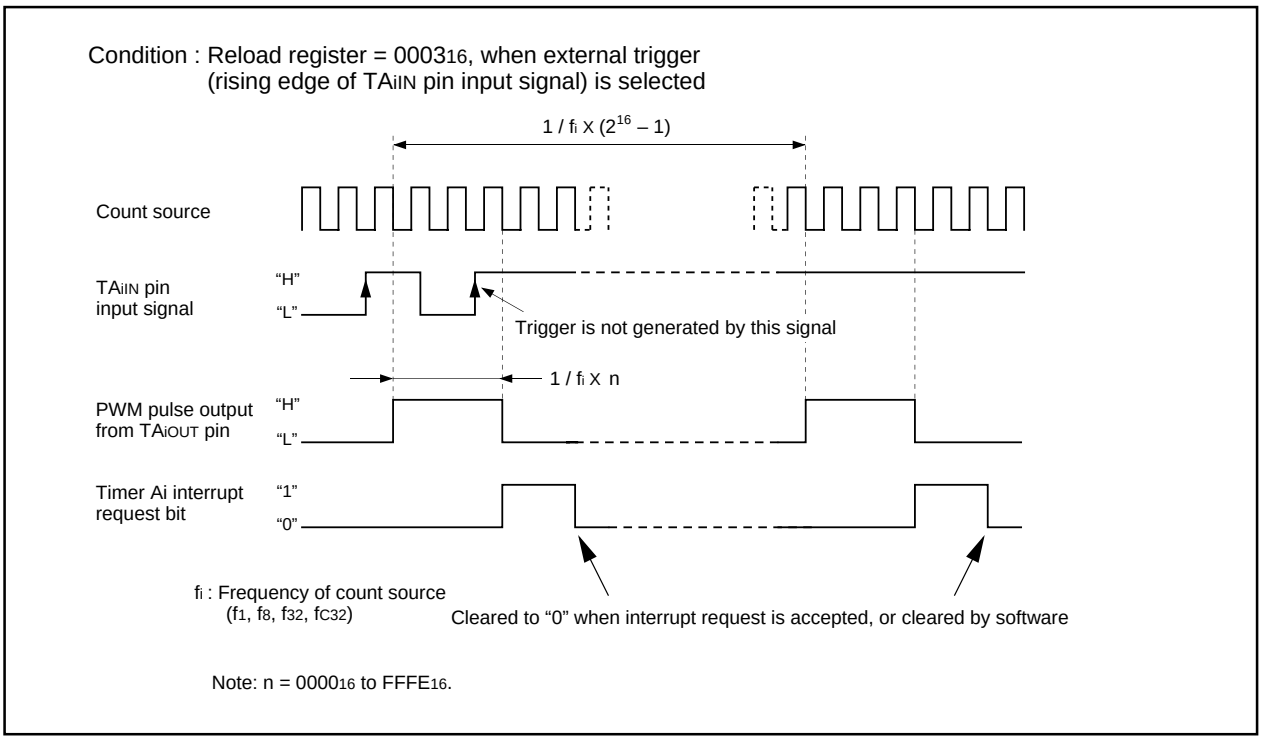


Figure 1.14.12. Example of how a 16-bit pulse width modulator operates

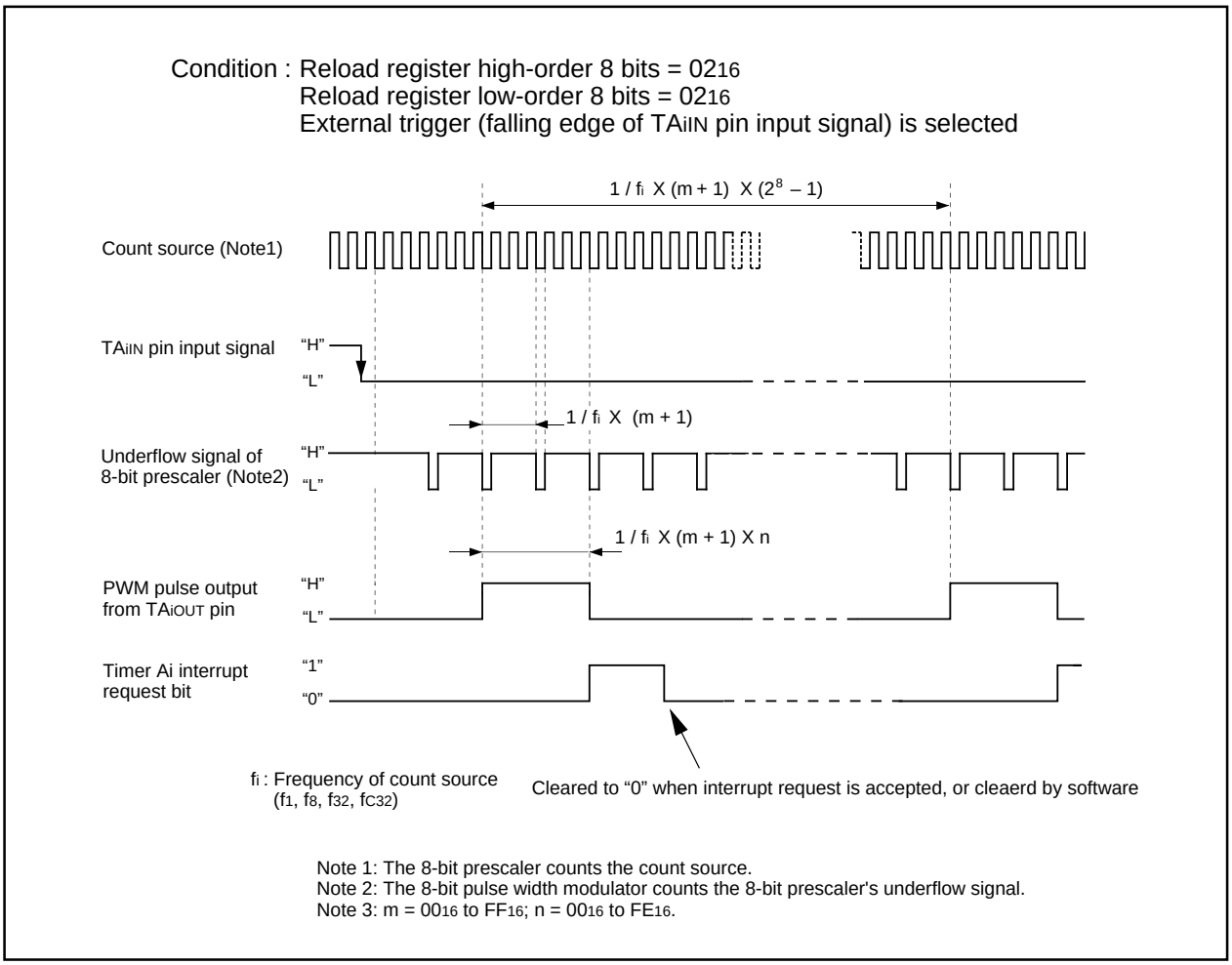


Figure 1.14.13. Example of how an 8-bit pulse width modulator operates

Timer B

Figure 1.14.14 shows the block diagram of timer B. Figures 1.14.15 and 1.14.16 show the timer B-related registers.

Use the timer Bi mode register (i = 0 to 5) bits 0 and 1 to choose the desired mode.

Timer B has three operation modes listed as follows:

- Timer mode: The timer counts an internal count source.
- Event counter mode: The timer counts pulses from an external source or a timer overflow.
- Pulse period/pulse width measuring mode: The timer measures an external signal's pulse period or pulse width.

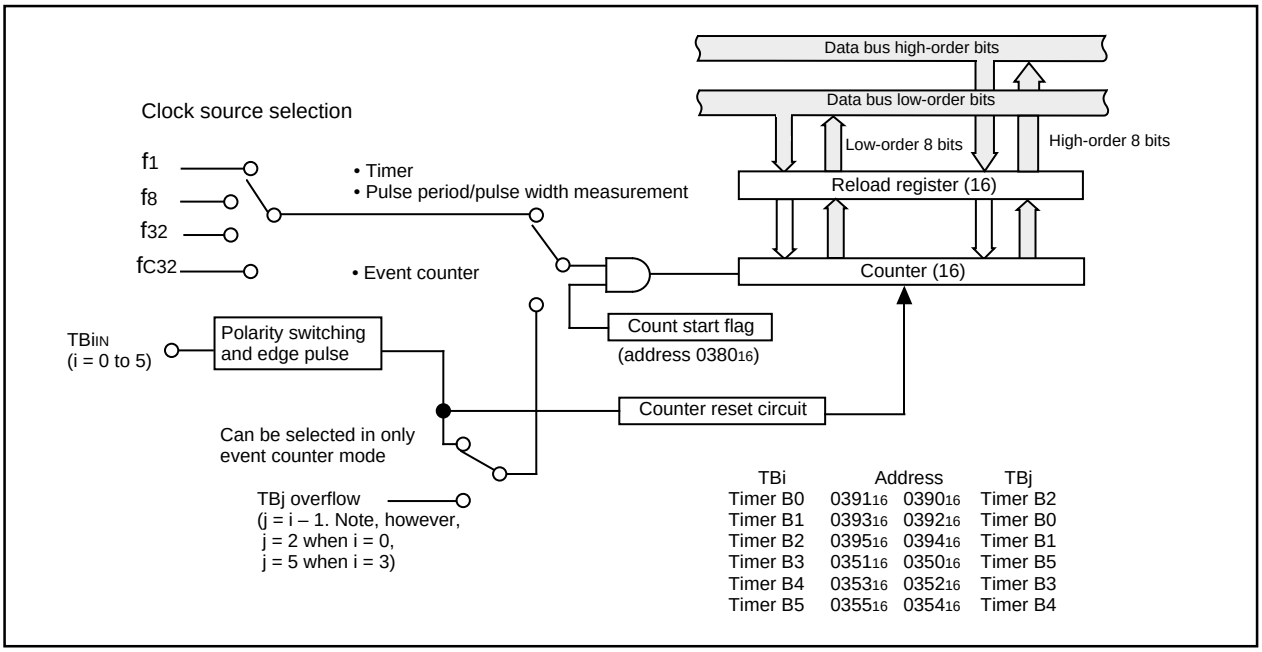


Figure 1.14.14. Block diagram of timer B

Timer Bi mode register

b7 b6 b5 b4 b3 b2 b1 b0

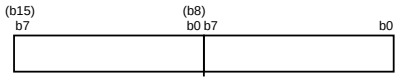
Symbol	Address	When reset
TBiMR(i = 0 to 5)	039B16 to 039D16	00XX00002
	035B16 to 035D16	00XX00002

Bit symbol	Bit name	Function	R	W
TMOD0	Operation mode select bit	b1 b0 0 0 : Timer mode 0 1 : Event counter mode 1 0 : Pulse period/pulse width measurement mode 1 1 : Must not be set.	○	○
TMOD1			○	○
MR0	Function varies with each operation mode		○	○
MR1			○	○
MR2			○ (Note 1)	○
			× (Note 2)	×
MR3			○	×
TCK0	Count source select bit (Function varies with each operation mode)		○	○
TCK1			○	○

Note 1: Timer B0, timer B3.
Note 2: Timer B1, timer B2, timer B4, timer B5.

Figure 1.14.15. Timer B-related registers (1)

Timer Bi register (Note)

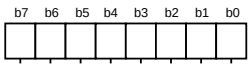


Symbol	Address	When reset
TB0	0391_{16}, 0390_{16}	Indeterminate
TB1	0393_{16}, 0392_{16}	Indeterminate
TB2	0395_{16}, 0394_{16}	Indeterminate
TB3	0351_{16}, 0350_{16}	Indeterminate
TB4	0353_{16}, 0352_{16}	Indeterminate
TB5	0355_{16}, 0354_{16}	Indeterminate

Function	Values that can be set	R	W
• Timer mode Counts the timer's period	0000_{16} to $FFFF_{16}$	○	○
• Event counter mode Counts external pulses input or a timer overflow	0000_{16} to $FFFF_{16}$	○	○
• Pulse period / pulse width measurement mode Measures a pulse period or width	——	○	×

Note: Read and write data in 16-bit units.

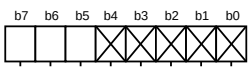
Count start flag



Symbol	Address	When reset
TABSR	0380_{16}	00_{16}

Bit symbol	Bit name	Function	R	W
TA0S	Timer A0 count start flag	0 : Stops counting 1 : Starts counting	○	○
TA1S	Timer A1 count start flag		○	○
TA2S	Timer A2 count start flag		○	○
TA3S	Timer A3 count start flag		○	○
TA4S	Timer A4 count start flag		○	○
TB0S	Timer B0 count start flag		○	○
TB1S	Timer B1 count start flag		○	○
TB2S	Timer B2 count start flag		○	○

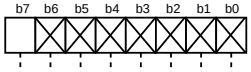
Timer B3, 4, 5 count start flag



Symbol	Address	When reset
TBSR	0340_{16}	$000XXXXX_2$

Bit symbol	Bit name	Function	R	W
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be indeterminate.			—	—
TB3S	Timer B3 count start flag	0 : Stops counting 1 : Starts counting	○	○
TB4S	Timer B4 count start flag		○	○
TB5S	Timer B5 count start flag		○	○

Clock prescaler reset flag



Symbol	Address	When reset
CPSRF	0381_{16}	$0XXXXXXX_2$

Bit symbol	Bit name	Function	R	W
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be indeterminate.			—	—
CPSR	Clock prescaler reset flag	0 : No effect 1 : Prescaler is reset (When read, the value is “0”)	○	○

Figure 1.14.16. Timer B-related registers (2)

(1) Timer mode

In this mode, the timer counts an internally generated count source. (See Table 1.14.6.) Figure 1.14.17 shows the timer Bi mode register in timer mode.

Table 1.14.6. Timer specifications in timer mode

Item	Specification
Count source	f_1, f_8, f_{32}, f_{C32}
Count operation	• Counts down • When the timer underflows, it reloads the reload register contents before continuing counting
Divide ratio	1/(n+1) n : Set value
Count start condition	Count start flag is set (= 1)
Count stop condition	Count start flag is reset (= 0)
Interrupt request generation timing	The timer underflows
TBi_{IN} pin function	Programmable I/O port
Read from timer	Count value is read out by reading timer Bi register
Write to timer	• When counting stopped When a value is written to timer Bi register, it is written to both reload register and counter • When counting in progress When a value is written to timer Bi register, it is written to only reload register (Transferred to counter at next reload time)

Timer Bi mode register

b7	b6	b5	b4	b3	b2	b1	b0
						0	0

Symbol	Address	When reset
TBiMR(i=0 to 5)	$039B_{16}$ to $039D_{16}$	$00XX0000_2$
	$035B_{16}$ to $035D_{16}$	$00XX0000_2$

Bit symbol	Bit name	Function	R	W
TMOD0	Operation mode select bit	b1 b0 0 0 : Timer mode	○	○
TMOD1			○	○
MR0	Invalid in timer mode Can be “0” or “1”		○	○
MR1			○	○
MR2	0 (Must always be “0” in timer mode ; i = 0, 3)		○ (Note 1)	○
	Nothing is assiigned (i = 1, 2, 4, 5). In an attempt to write to this bit, write “0”. The value, if read, turns out to be indeterminate.		× (Note 2)	×
MR3	Invalid in timer mode. In an attempt to write to this bit, write “0”. The value, if read in timer mode, turns out to be indeterminate.		○	×
TCK0	Count source select bit	b7 b6 0 0 : f_1 0 1 : f_8 1 0 : f_{32} 1 1 : f_{C32}	○	○
TCK1			○	○

Note 1: Timer B0, timer B3.
Note 2: Timer B1, timer B2, timer B4, timer B5.

Figure 1.14.17. Timer Bi mode register in timer mode

(2) Event counter mode

In this mode, the timer counts an external signal or an internal timer's overflow. (See Table 1.14.7.) Figure 1.14.18 shows the timer Bi mode register in event counter mode.

Table 1.14.7. Timer specifications in event counter mode

Item	Specification
Count source	• External signals input to TBiIN pin • Effective edge of count source can be a rising edge, a falling edge, or falling and rising edges as selected by software
Count operation	• Counts down • When the timer underflows, it reloads the reload register contents before continuing counting
Divide ratio	1/(n+1) n : Set value
Count start condition	Count start flag is set (= 1)
Count stop condition	Count start flag is reset (= 0)
Interrupt request generation timing	The timer underflows
TBiIN pin function	Count source input
Read from timer	Count value can be read out by reading timer Bi register
Write to timer	• When counting stopped When a value is written to timer Bi register, it is written to both reload register and counter • When counting in progress When a value is written to timer Bi register, it is written to only reload register (Transferred to counter at next reload time)

Timer Bi mode register

b7	b6	b5	b4	b3	b2	b1	b0
						0	1

Symbol	Address	When reset
TBiMR(i=0 to 5)	039B16 to 039D16 035B16 to 035D16	00XX00002 00XX00002

Bit symbol	Bit name	Function	R	W
TMOD0	Operation mode select bit	b1 b0 0 1 : Event counter mode	○	○
TMOD1			○	○
MR0	Count polarity select bit (Note 1)	b3 b2 0 0 : Counts external signal's falling edges 0 1 : Counts external signal's rising edges 1 0 : Counts external signal's falling and rising edges 1 1 : Must not be set.	○	○
MR1			○	○
MR2	0 (Must always be “0” in event counter mode; i = 0, 3)		○ (Note 2)	○
	Nothing is assigned (i = 1, 2, 4, 5). In an attempt to write to this bit, write “0”. The value, if read, turns out to be indeterminate.		× (Note 3)	×
MR3	Invalid in event counter mode. In an attempt to write to this bit, write “0”. The value, if read in event counter mode, turns out to be indeterminate.		○	×
TCK0	Invalid in event counter mode. Can be “0” or “1”.		○	○
TCK1	Event clock select	0 : Input from TBiIN pin (Note 4) 1 : TBj overflow (j = i – 1; however, j = 2 when i = 0, j = 5 when i = 3)	○	○

Note 1: Valid only when input from the TBiIN pin is selected as the event clock.
If timer's overflow is selected, this bit can be “0” or “1”.
Note 2: Timer B0, timer B3.
Note 3: Timer B1, timer B2, timer B4, timer B5.
Note 4: Set the corresponding port direction register to “0”.

Figure 1.14.18. Timer Bi mode register in event counter mode

(3) Pulse period/pulse width measurement mode

In this mode, the timer measures the pulse period or pulse width of an external signal. (See Table 1.14.8.) Figure 1.14.19 shows the timer Bi mode register in pulse period/pulse width measurement mode. Figure 1.14.20 shows the operation timing when measuring a pulse period. Figure 1.14.21 shows the operation timing when measuring a pulse width.

Table 1.14.8. Timer specifications in pulse period/pulse width measurement mode

Item	Specification
Count source	f_1, f_8, f_{32}, f_{C32}
Count operation	• Up count • Counter value "0000_{16}" is transferred to reload register at measurement pulse's effective edge and the timer continues counting
Count start condition	Count start flag is set (= 1)
Count stop condition	Count start flag is reset (= 0)
Interrupt request generation timing	• When measurement pulse's effective edge is input (Note 1) • When an overflow occurs. (Simultaneously, the timer Bi overflow flag changes to "1". The timer Bi overflow flag changes to "0" when the count start flag is "1" and a value is written to the timer Bi mode register.)
TBi_{IN} pin function	Measurement pulse input
Read from timer	When timer Bi register is read, it indicates the reload register's content (measurement result) (Note 2)
Write to timer	Cannot be written to

Note 1: An interrupt request is not generated when the first effective edge is input after the timer has started counting.
Note 2: The value read out from the timer Bi register is indeterminate until the second effective edge is input after the timer has started counting.

Timer Bi mode register

b7	b6	b5	b4	b3	b2	b1	b0
						1	0

Symbol	Address	When reset
TBiMR(i=0 to 5)	$039B_{16}$ to $039D_{16}$	$00XX0000_2$
	$035B_{16}$ to $035D_{16}$	$00XX0000_2$

Bit symbol	Bit name	Function	R	W
TMOD0	Operation mode select bit	b1 b0 1 0 : Pulse period / pulse width measurement mode	○	○
TMOD1			○	○
MR0	Measurement mode select bit	b3 b2 0 0 : Pulse period measurement (Interval between measurement pulse's falling edge to falling edge) 0 1 : Pulse period measurement (Interval between measurement pulse's rising edge to rising edge) 1 0 : Pulse width measurement (Interval between measurement pulse's falling edge to rising edge, and between rising edge to falling edge) 1 1 : Must not be set.	○	○
MR1			○	○
MR2	0 (Must always be "0" in pulse period/pulse width measurement mode; i = 0, 3)		○ (Note 2)	○
	Nothing is assigned (i = 1, 2, 4, 5). In an attempt to write to this bit, write "0". The value, if read, turns out to be indeterminate.		× (Note 3)	×
MR3	Timer Bi overflow flag (Note 1)	0 : Timer did not overflow 1 : Timer has overflowed	○	×
TCK0	Count source select bit	b7 b6 0 0 : f_1 0 1 : f_8 1 0 : f_{32} 1 1 : f_{C32}	○	○
TCK1			○	○

Note 1: It is indeterminate when reset. The timer Bi overflow flag changes to "0" when the count start flag is "1" and a value is written to the timer Bi mode register. This flag cannot be set to "1" by software.
Note 2: Timer B0, timer B3.
Note 3: Timer B1, timer B2, timer B4, timer B5.

Figure 1.14.19. Timer Bi mode register in pulse period/pulse width measurement mode

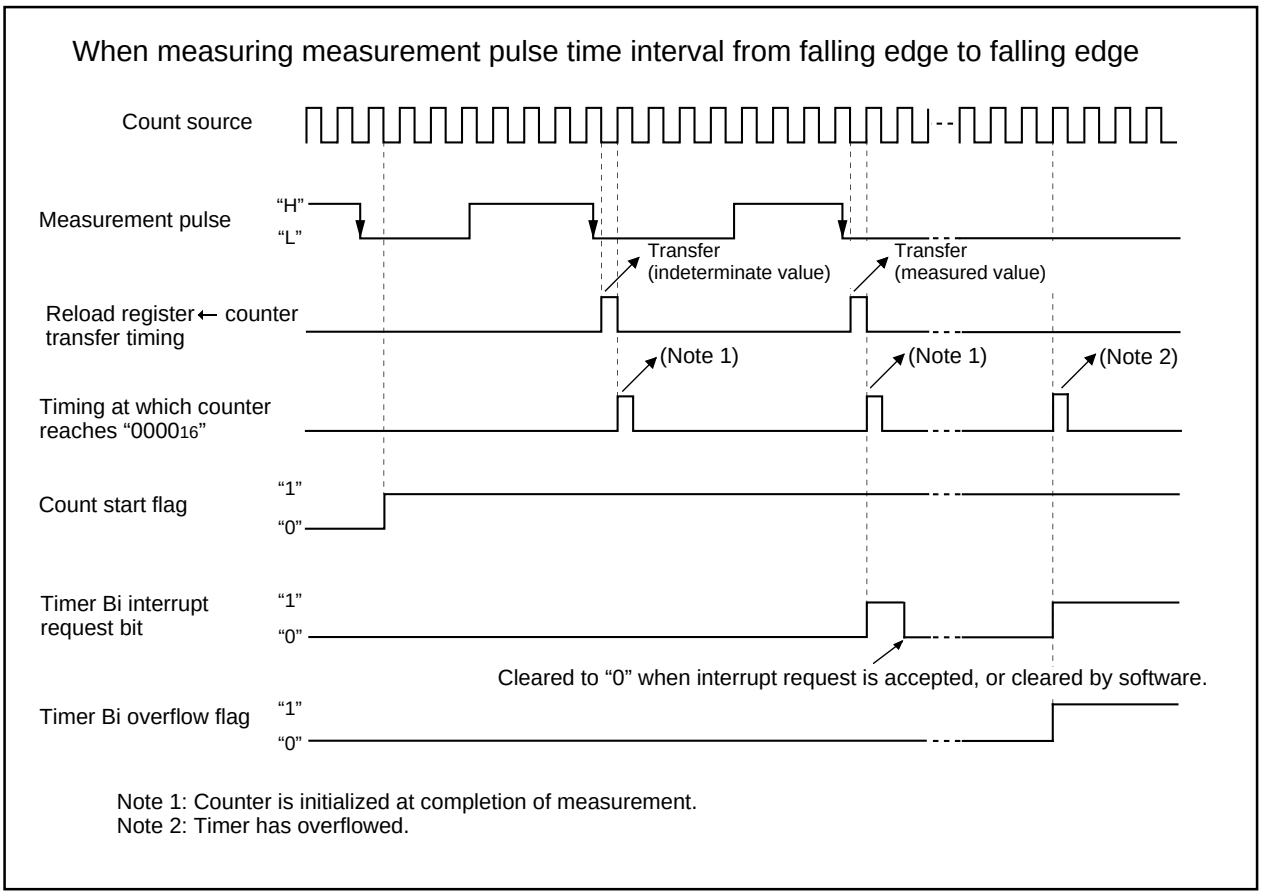


Figure 1.14.20. Operation timing when measuring a pulse period

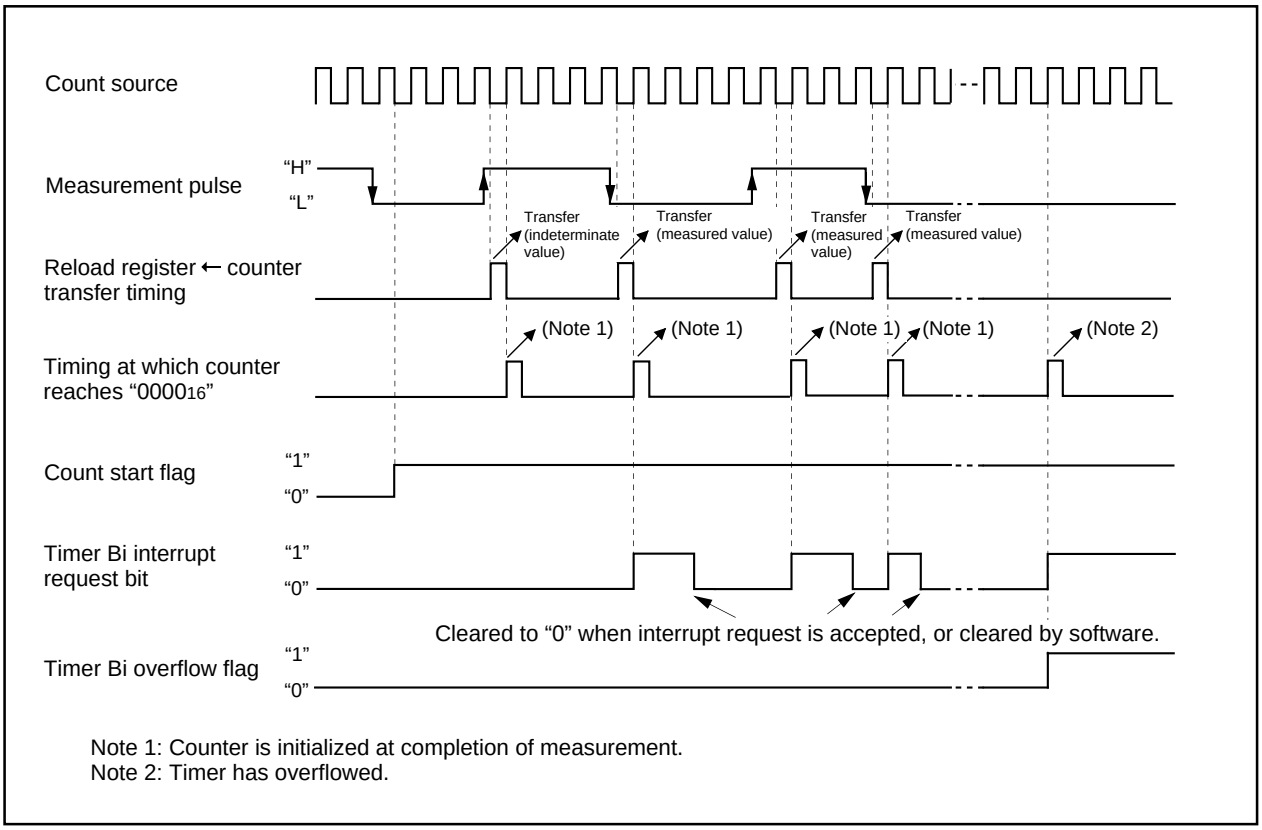


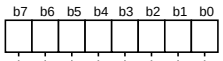
Figure 1.14.21. Operation timing when measuring a pulse width

Timers' functions for three-phase motor control

Use of more than one built-in timer A and timer B provides the means of outputting three-phase motor driving waveforms.

Figures 1.15.1 to 1.15.3 show registers related to timers for three-phase motor control.

Three-phase PWM control register 0

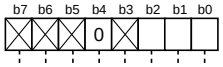


Symbol: INVC0 | Address: 034816 | When reset: 0016

Bit symbol	Bit name	Description	R	W
INV00	Effective interrupt output polarity select bit	0: A timer B2 interrupt occurs when the timer A1 reload control signal is "1". 1: A timer B2 interrupt occurs when the timer A1 reload control signal is "0". **Effective only in three-phase mode 1**	○	○
INV01	Effective interrupt output specification bit (Note 4)	0: Not specified. 1: Selected by the effective interrupt output polarity selection bit. **Effective only in three-phase mode 1**	○	○
INV02	Mode select bit (Note 2)	0: Normal mode 1: Three-phase PWM output mode	○	○
INV03	Output control bit	0: Output disabled 1: Output enabled	○	○
INV04	Positive and negative phases concurrent L output disable function enable bit	0: Feature disabled 1: Feature enabled	○	○
INV05	Positive and negative phases concurrent L output detect flag	0: Not detected yet 1: Already detected	○	○ (Note 1)
INV06	Modulation mode select bit (Note 3)	0: Triangular wave modulation mode 1: Sawtooth wave modulation mode	○	○
INV07	Software trigger bit	1: Trigger generated The value, when read, is "0".	○	○

Note 1: No value other than "0" can be written.
Note 2: Selecting three-phase PWM output mode causes $P8_0$, $P8_1$, and $P7_2$ through $P7_5$ to output U, $\overline{U}$, V, $\overline{V}$, W, and $\overline{W}$, and works the timer for setting short circuit prevention time, the U, V, W phase output control circuits, and the circuit for setting timer B2 interrupt frequency.
Note 3: **In triangular wave modulation mode:**
The short circuit prevention timer starts in synchronization with the falling edge of timer Ai output.
The data transfer from the three-phase buffer register to the three-phase output shift register is made only once in synchronization with the transfer trigger signal after writing to the three-phase output buffer register.
In sawtooth wave modulation mode:
The short circuit prevention timer starts in synchronization with the falling edge of timer A output and with the transfer trigger signal.
The data transfer from the three-phase output buffer register to the three-phase output shift register is made with respect to every transfer trigger.
Note 4: To write "1" to bit 1 (INV01) of the three-phase PWM control register 0, set in advance the content of the timer B2 interrupt occurrences frequency set counter.

Three-phase PWM control register 1



Symbol: INVC1 | Address: 034916 | When reset: 0016

Bit symbol	Bit name	Description	R	W
INV10	Timer Ai start trigger signal select bit	0: Timer B2 overflow signal 1: Timer B2 overflow signal, signal for writing to timer B2	○	○
INV11	Timer A1-1, A2-1, A4-1 control bit	0: Three-phase mode 0 1: Three-phase mode 1	○	○
INV12	Short circuit timer count source select bit	0 : Must not be set. 1 : $f_1/2$ (Note)	○	○
Noting is assigned. In an attempt to write to this bit, write "0". The value, if read, turns out to be "0".			—	—
Reserved bit		Must always be set to "0"	○	○
Noting is assigned. In an attempt to write to these bits, write "0". The value, if read, turns out to be "0".			—	—

Note : To use three-phase PWM output mode, write "1" to INV12.

Figure 1.15.1. Registers related to timers for three-phase motor control

Three-phase output buffer register 0

b7	b6	b5	b4	b3	b2	b1	b0
×	×						

Symbol: IDB0 — Address: 034A16 — When reset: 0016

Bit Symbol	Bit name	Function	R	W
DU0	U phase output buffer 0	Setting in U phase output buffer 0	(Note)	○
DUB0	$\overline{U}$ phase output buffer 0	Setting in $\overline{U}$ phase output buffer 0	(Note)	○
DV0	V phase output buffer 0	Setting in V phase output buffer 0	(Note)	○
DVB0	$\overline{V}$ phase output buffer 0	Setting in $\overline{V}$ phase output buffer 0	(Note)	○
DW0	W phase output buffer 0	Setting in W phase output buffer 0	(Note)	○
DWB0	$\overline{W}$ phase output buffer 0	Setting in $\overline{W}$ phase output buffer 0	(Note)	○
Nothing is assigned. In an attempt to write to these bits, write "0". The value, if read, turns out to be "0".			—	—

Note: When executing read instruction of this register, the contents of three-phase shift register is read out.

Three-phase output buffer register 1

b7	b6	b5	b4	b3	b2	b1	b0
×	×						

Symbol: IDB1 — Address: 034B16 — When reset: 0016

Bit Symbol	Bit name	Function	R	W
DU1	U phase output buffer 1	Setting in U phase output buffer 1	(Note)	○
DUB1	$\overline{U}$ phase output buffer 1	Setting in $\overline{U}$ phase output buffer 1	(Note)	○
DV1	V phase output buffer 1	Setting in V phase output buffer 1	(Note)	○
DVB1	$\overline{V}$ phase output buffer 1	Setting in $\overline{V}$ phase output buffer 1	(Note)	○
DW1	W phase output buffer 1	Setting in W phase output buffer 1	(Note)	○
DWB1	$\overline{W}$ phase output buffer 1	Setting in $\overline{W}$ phase output buffer 1	(Note)	○
Nothing is assigned. In an attempt to write to these bits, write "0". The value, if read, turns out to be "0".			—	—

Note: When executing read instruction of this register, the contents of three-phase shift register is read out.

Dead time timer (Note)

b7	b0

Symbol: DTT — Address: 034C16 — When reset: Indeterminate

Function	Values that can be set	R	W
Set dead time timer	1 to 255	—	○

Note: Use MOV instruction to write to this register.

Timer B2 interrupt occurrences frequency set counter (Note 1, 2, 3)

b3	b0

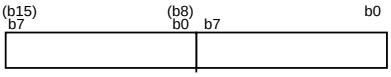
Symbol: ICTB2 — Address: 034D16 — When reset: Indeterminate

Function	Values that can be set	R	W
Set occurrence frequency of timer B2 interrupt request	1 to 15	—	○

Note 1: In setting 1 to bit 1 (INV01) - the effective interrupt output specification bit - of three-phase PWM control register 0, do not change the B2 interrupt occurrences frequency set counter to deal with the timer function for three-phase motor control.
Note 2: Do not write at the timing of an overflow occurrence in timer B2.
Note 3: Use MOV instruction to write to this register.

Figure 1.15.2. Registers related to timers for three-phase motor control

Timer Ai register (Note 1)

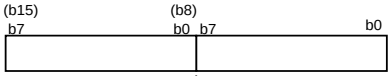


Symbol	Address	When reset
TA1	0389$_{16}$,0388$_{16}$	Indeterminate
TA2	038B$_{16}$,038A$_{16}$	Indeterminate
TA4	038F$_{16}$,038E$_{16}$	Indeterminate
TB2	0395$_{16}$,0394$_{16}$	Indeterminate

Function	Values that can be set	R	W
• Timer mode Counts an internal count source	0000$_{16}$ to FFFF$_{16}$	○	○
• One-shot timer mode Counts a one shot width	0000$_{16}$ to FFFF$_{16}$ (Note 2, 3)	×	○

Note 1: Read and write data in 16-bit units.
Note 2: When the timer Ai register is set to "0000$_{16}$", the counter does not operate and a timer Ai interrupt does not occur.
Note 3: Use MOV instruction to write to this register.

Timer Ai-1 register (Note)

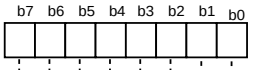


Symbol	Address	When reset
TA11	0343$_{16}$,0342$_{16}$	Indeterminate
TA21	0345$_{16}$,0344$_{16}$	Indeterminate
TA41	0347$_{16}$,0346$_{16}$	Indeterminate

Function	Values that can be set	R	W
Counts an internal count source	0000$_{16}$ to FFFF$_{16}$	○	○

Note: Read and write data in 16-bit units.

Trigger select register

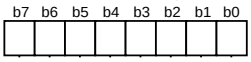


Symbol	Address	When reset
TRGSR	0383$_{16}$	00$_{16}$

Bit symbol	Bit name	Function	R	W
TA1TGL	Timer A1 event/trigger select bit	b1 b0 0 0 : Input on TA1$_{IN}$ is selected (Note) 0 1 : TB2 overflow is selected 1 0 : TA0 overflow is selected 1 1 : TA2 overflow is selected	○	○
TA1TGH			○	○
TA2TGL	Timer A2 event/trigger select bit	b3 b2 0 0 : Input on TA2$_{IN}$ is selected (Note) 0 1 : TB2 overflow is selected 1 0 : TA1 overflow is selected 1 1 : TA3 overflow is selected	○	○
TA2TGH			○	○
TA3TGL	Timer A3 event/trigger select bit	b5 b4 0 0 : Input on TA3$_{IN}$ is selected (Note) 0 1 : TB2 overflow is selected 1 0 : TA2 overflow is selected 1 1 : TA4 overflow is selected	○	○
TA3TGH			○	○
TA4TGL	Timer A4 event/trigger select bit	b7 b6 0 0 : Input on TA4$_{IN}$ is selected (Note) 0 1 : TB2 overflow is selected 1 0 : TA3 overflow is selected 1 1 : TA0 overflow is selected	○	○
TA4TGH			○	○

Note: Set the corresponding port direction register to "0".

Count start flag



Symbol	Address	When reset
TABSR	0380$_{16}$	00$_{16}$

Bit symbol	Bit name	Function	R	W
TA0S	Timer A0 count start flag	0 : Stops counting 1 : Starts counting	○	○
TA1S	Timer A1 count start flag		○	○
TA2S	Timer A2 count start flag		○	○
TA3S	Timer A3 count start flag		○	○
TA4S	Timer A4 count start flag		○	○
TB0S	Timer B0 count start flag		○	○
TB1S	Timer B1 count start flag		○	○
TB2S	Timer B2 count start flag		○	○

Figure 1.15.3. Registers related to timers for three-phase motor control

Three-phase motor driving waveform output mode (three-phase PWM output mode)

Setting "1" in the mode select bit (bit 2 at 0348_{16}) shown in Figure 1.15.1 - causes three-phase PWM output mode that uses four timers A1, A2, A4, and B2 to be selected. As shown in Figure 1.15.4, set timers A1, A2, and A4 in one-shot timer mode, set the trigger in timer B2, and set timer B2 in timer mode using the respective timer mode registers.

Timer Ai mode register

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

Symbol	Address	When reset
TA1MR	039716	0016
TA2MR	039816	0016
TA3MR	$039A16$	0016

Bit symbol	Bit name	Function	R	W
TMOD0	Operation mode select bit	b1 b0 1 0 : One-shot timer mode	○	○
TMOD1			○	○
MR0	Pulse output function select bit	0 (Must always be "0" in three-phase PWM output mode)	○	○
MR1	External trigger select bit	Invalid in three-phase PWM output mode	○	○
MR2	Trigger select bit	1 : Selected by event/trigger select register	○	○
MR3	0 (Must always be "0" in one-shot timer mode)		○	○
TCK0	Count source select bit	b7 b6 0 0 : f_1 0 1 : f_8 1 0 : f_{32} 1 1 : fc_{32}	○	○
TCK1			○	○

Timer B2 mode register

b7	b6	b5	b4	b3	b2	b1	b0
			0			0	0

Symbol	Address	When reset
TB2MR	$039D16$	$00XX0000_2$

Bit symbol	Bit name	Function	R	W
TMOD0	Operation mode select bit	b1 b0 0 0 : Timer mode	○	○
TMOD1			○	○
MR0	Invalid in timer mode Can be "0" or "1"		○	○
MR1			○	○
MR2	0 (Must always be "0" in timer mode)		○	○
MR3	Invalid in timer mode. In an attempt to write to this bit, write "0". When read in timer mode, its content is indeterminate.		○	×
TCK0	Count source select bit	b7 b6 0 0 : f_1 0 1 : f_8 1 0 : f_{32} 1 1 : fc_{32}	○	○
TCK1			○	○

Figure 1.15.4. Timer mode registers in three-phase PWM output mode

Figure 1.15.5 shows the block diagram for three-phase PWM output mode. In three-phase PWM output mode, the positive-phase waveforms (U phase, V phase, and W phase) and negative waveforms ($\overline{U}$ phase, $\overline{V}$ phase, and $\overline{W}$ phase), six waveforms in total, are output from $P8_0$, $P8_1$, $P7_2$, $P7_3$, $P7_4$, and $P7_5$ as active on the "L" level. Of the timers used in this mode, timer A4 controls the U phase and $\overline{U}$ phase, timer A1 controls the V phase and $\overline{V}$ phase, and timer A2 controls the W phase and $\overline{W}$ phase respectively; timer B2 controls the periods of one-shot pulse output from timers A4, A1, and A2.

In outputting a waveform, dead time can be set so as to cause the "L" level of the positive waveform output (U phase, V phase, and W phase) not to lap over the "L" level of the negative waveform output ($\overline{U}$ phase, $\overline{V}$ phase, and $\overline{W}$ phase).

To set short circuit time, use three 8-bit timers sharing the reload register for setting dead time. A value from 1 through 255 can be set as the count of the timer for setting dead time. The timer for setting dead time works as a one-shot timer. If a value is written to the dead time timer ($034C_{16}$), the value is written to the reload register shared by the three timers for setting dead time.

Any of the timers for setting dead time takes the value of the reload register into its counter, if a start trigger comes from its corresponding timer, and performs a down count in line with the clock source selected by the dead time timer count source select bit (bit 2 at 0349_{16}). The timer can receive another trigger again before the workings due to the previous trigger are completed. In this instance, the timer performs a down count from the reload register's content after its transfer, provoked by the trigger, to the timer for setting dead time.

Since the timer for setting dead time works as a one-shot timer, it starts outputting pulses if a trigger comes; it stops outputting pulses as soon as its content becomes 00_{16}, and waits for the next trigger to come.

The positive waveforms (U phase, V phase, and W phase) and the negative waveforms ($\overline{U}$ phase, $\overline{V}$ phase, and $\overline{W}$ phase) in three-phase PWM output mode are output from respective ports by means of setting "1" in the output control bit (bit 3 at 0348_{16}). Setting "0" in this bit causes the ports to be the state of set by port direction register. This bit can be set to "0" not only by use of the applicable instruction, but by entering a falling edge in the $\overline{NMI}$ terminal or by resetting. Also, if "1" is set in the positive and negative phases concurrent L output disable function enable bit (bit 4 at 0348_{16}) causes one of the pairs of U phase and $\overline{U}$ phase, V phase and $\overline{V}$ phase, and W phase and $\overline{W}$ phase concurrently go to "L", as a result, the port becomes the state of set by port direction register.

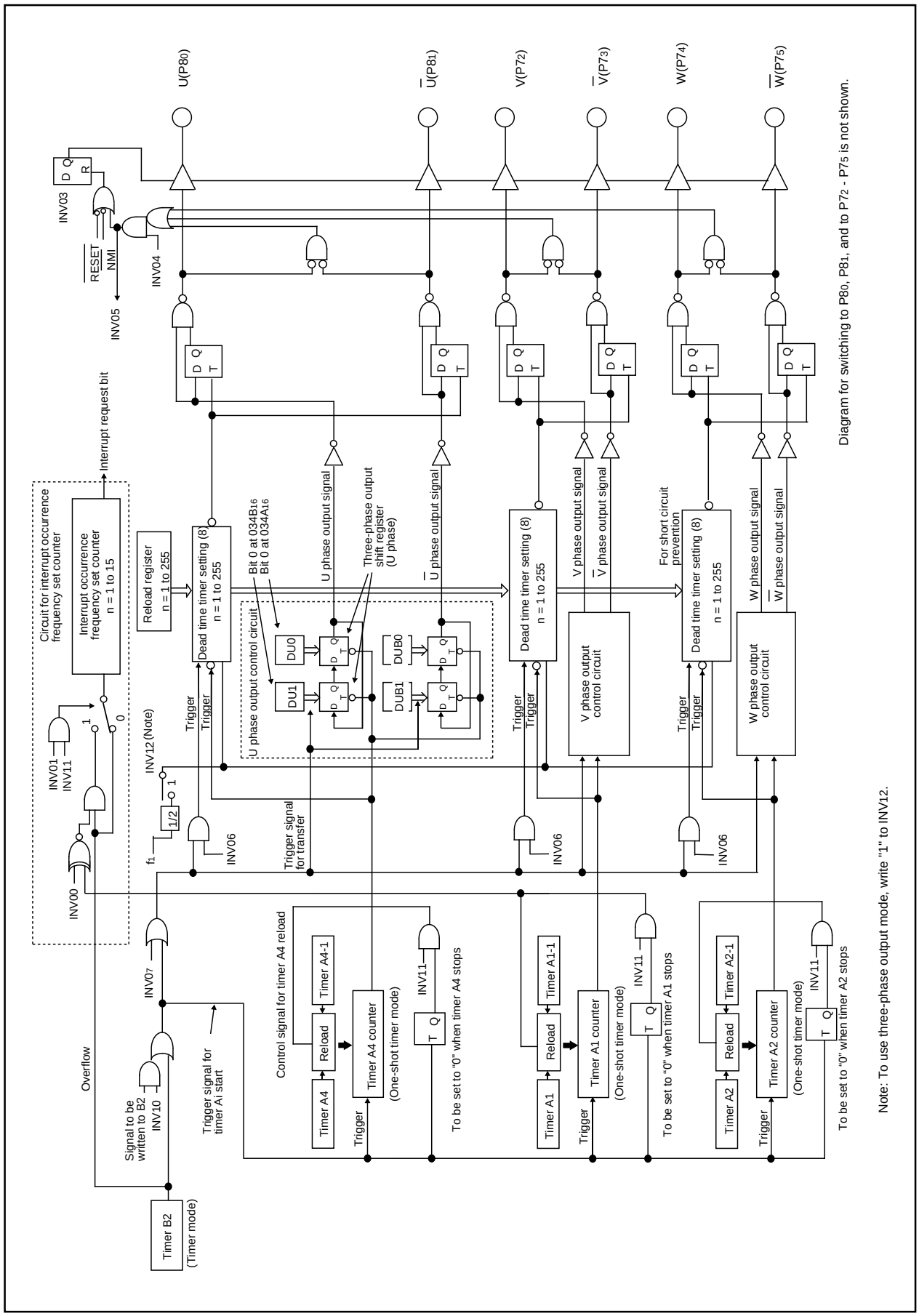


Figure 1.15.5. Block diagram for three-phase PWM output mode

Triangular wave modulation

To generate a PWM waveform of triangular wave modulation, set "0" in the modulation mode select bit (bit 6 at 0348_{16}). Also, set "1" in the timers A4-1, A1-1, A2-1 control bit (bit 1 at 0349_{16}). In this mode, each of timers A4, A1, and A2 has two timer registers, and alternately reloads the timer register's content to the counter every time timer B2 counter's content becomes 0000_{16}. If "0" is set to the effective interrupt output specification bit (bit 1 at 0348_{16}), the frequency of interrupt requests that occur every time the timer B2 counter's value becomes 0000_{16} can be set by use of the timer B2 counter ($034D_{16}$) for setting the frequency of interrupt occurrences. The frequency of occurrences is given by (setting; setting $\neq$ 0).

Setting "1" in the effective interrupt output specification bit (bit 1 at 0348_{16}) provides the means to choose which value of the timer A1 reload control signal to use, "0" or "1", to cause timer B2's interrupt request to occur. To make this selection, use the effective interrupt output polarity selection bit (bit 0 at 0348_{16}).

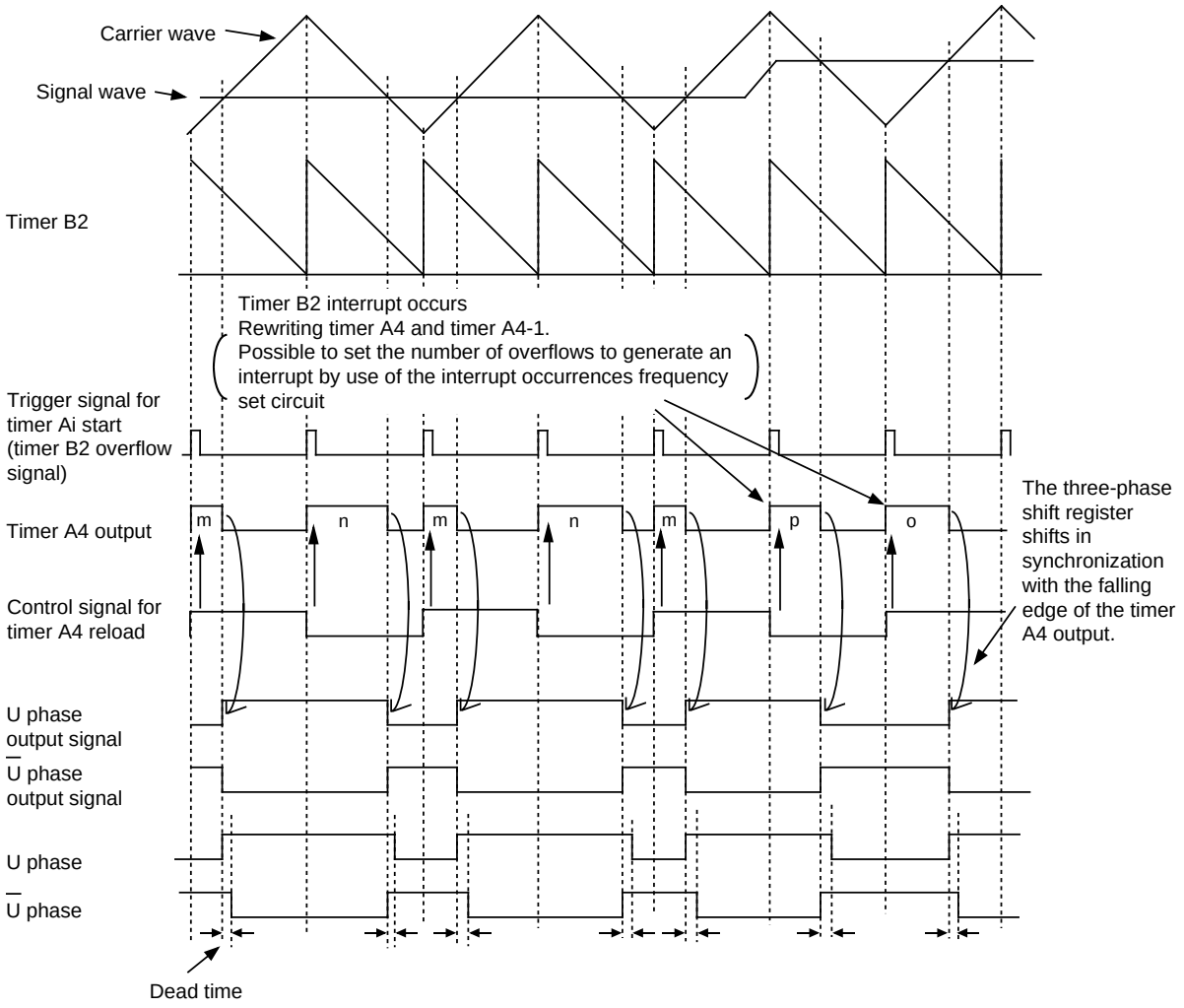
An example of U phase waveform is shown in Figure 1.15.6, and the description of waveform output workings is given below. Set "1" in DU0 (bit 0 at $034A_{16}$). And set "0" in DUB0 (bit 1 at $034A_{16}$). In addition, set "0" in DU1 (bit 0 at $034B_{16}$) and set "1" in DUB1 (bit 1 at $034B_{16}$). Also, set "0" in the effective interrupt output specification bit (bit 1 at 0348_{16}) to set a value in the timer B2 interrupt occurrence frequency set counter. By this setting, a timer B2 interrupt occurs when the timer B2 counter's content becomes 0000_{16} as many as (setting) times. Furthermore, set "1" in the effective interrupt output specification bit (bit 1 at 0348_{16}), set "0" in the effective interrupt output polarity select bit (bit 0 at 0348_{16}) and set "1" in the interrupt occurrence frequency set counter ($034D_{16}$). These settings cause a timer B2 interrupt to occur every other interval when the U phase output goes to "H".

When the timer B2 counter's content becomes 0000_{16}, timer A4 starts outputting one-shot pulses. In this instance, the content of DU1 (bit 0 at $034B_{16}$) and that of DU0 (bit 0 at $034A_{16}$) are set in the three-phase output shift register (U phase), the content of DUB1 (bit 1 at $034B_{16}$) and that of DUB0 (bit 1 at $034A_{16}$) are set in the three-phase output shift register ($\overline{U}$ phase). After triangular wave modulation mode is selected, however, no setting is made in the shift register even though the timer B2 counter's content becomes 0000_{16}.

The value of DU0 and that of DUB0 are output to the U terminal ($P8_0$) and to the $\overline{U}$ terminal ($P8_1$) respectively. When the timer A4 counter counts the value written to timer A4 ($038F_{16}$, $038E_{16}$) and when timer A4 finishes outputting one-shot pulses, the three-phase shift register's content is shifted one position, and the value of DU1 and that of DUB1 are output to the U phase output signal and to $\overline{U}$ phase output signal respectively. At this time, one-shot pulses are output from the timer for setting dead time used for setting the time over which the "L" level of the U phase waveform does not lap over the "L" level of the $\overline{U}$ phase waveform, which has the opposite phase of the former. The U phase waveform output that started from the "H" level keeps its level until the timer for setting dead time finishes outputting one-shot pulses even though the three-phase output shift register's content changes from "1" to "0" by the effect of the one-shot pulses. When the timer for setting dead time finishes outputting one-shot pulses, "0" already shifted in the three-phase shift register goes effective, and the U phase waveform changes to the "L" level. When the timer B2 counter's content becomes 0000_{16}, the timer A4 counter starts counting the value written to timer A4-1 (0347_{16}, 0346_{16}), and starts outputting one-shot pulses. When timer A4 finishes outputting one-shot pulses, the three-phase shift register's content is shifted one position, but if the three-phase output shift register's content changes from "0" to "1" as a result of the shift, the output level changes from "L" to "H" without waiting for the timer for setting dead time to finish outputting one-shot pulses. A U phase waveform is generated by these workings repeatedly. With the exception that the three-phase output shift register on the $\overline{U}$ phase side is used, the workings in generating a $\overline{U}$ phase waveform, which has the opposite phase of the U phase waveform, are the same as in generating a U

phase waveform. In this way, a waveform can be picked up from the applicable terminal in a manner in which the "L" level of the U phase waveform doesn't lap over that of the $\overline{U}$ phase waveform, which has the opposite phase of the U phase waveform. The width of the "L" level too can be adjusted by varying the values of timer B2, timer A4, and timer A4-1. In dealing with the V and W phases, and $\overline{V}$ and $\overline{W}$ phases, the latter are of opposite phase of the former, have the corresponding timers work similarly to dealing with the U and $\overline{U}$ phases to generate an intended waveform.

A carrier wave of triangular waveform



Note: Set to triangular wave modulation mode and to three-phase mode 1.

Figure 1.15.6. Timing chart of operation (1)

Assigning certain values to DU0 (bit 0 at 034A16) and DUB0 (bit 1 at 034A16), and to DU1 (bit 0 at 034B16) and DUB1 (bit 1 at 034B16) allows the user to output the waveforms as shown in Figure 1.15.7, that is, to output the U phase alone, to fix $\overline{U}$ phase to "H", to fix the U phase to "H," or to output the $\overline{U}$ phase alone.

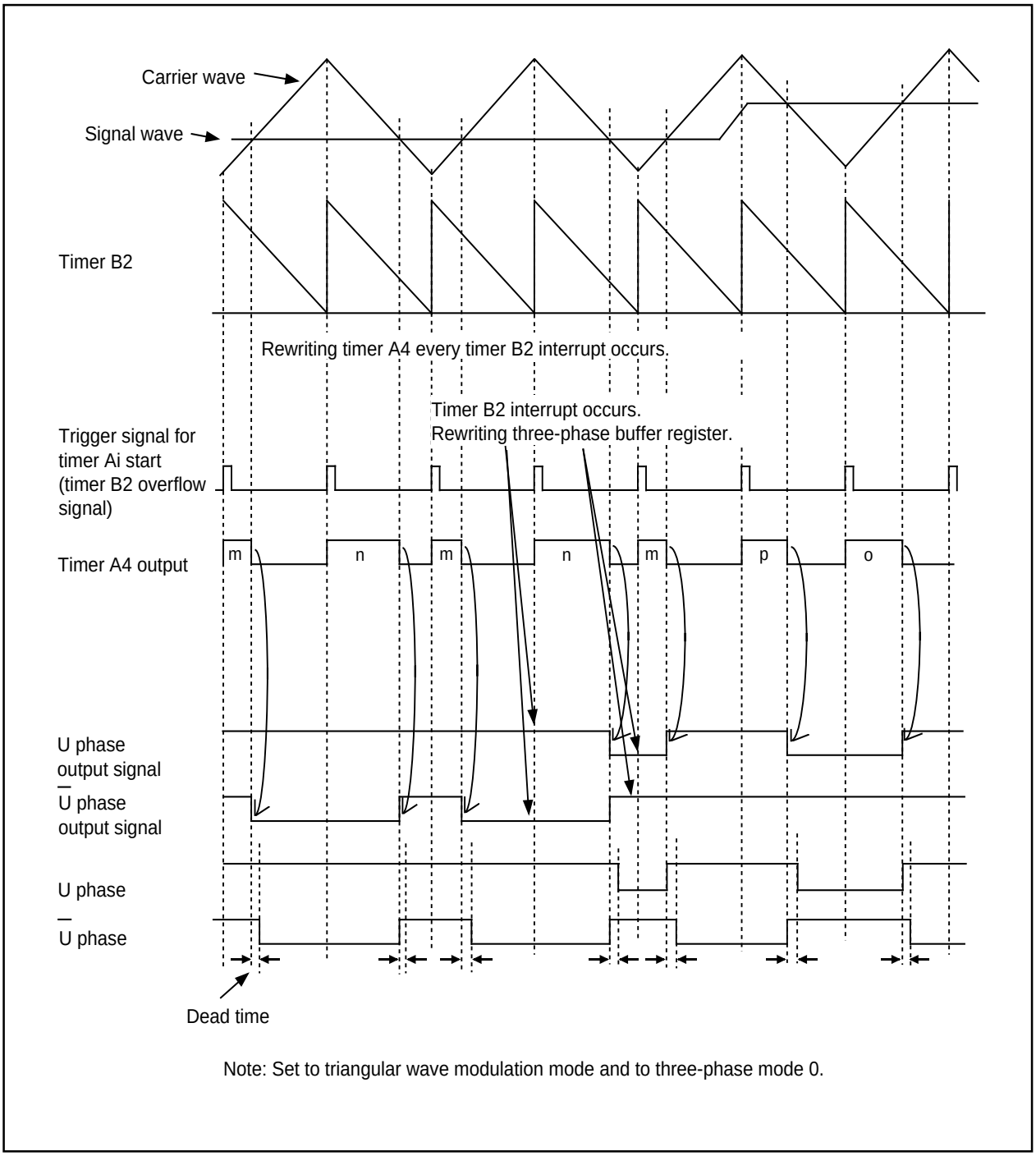


Figure 1.15.7. Timing chart of operation (2)

Sawtooth modulation

To generate a PWM waveform of sawtooth wave modulation, set "1" in the modulation mode select bit (bit 6 at 0348_{16}). Also, set "0" in the timers A4-1, A1-1, and A2-1 control bit (bit 1 at 0349_{16}). In this mode, the timer registers of timers A4, A1, and A2 comprise conventional timers A4, A1, and A2 alone, and reload the corresponding timer register's content to the counter every time the timer B2 counter's content becomes 0000_{16}. The effective interrupt output specification bit (bit 1 at 0348_{16}) and the effective interrupt output polarity select bit (bit 0 at 0348_{16}) go nullified.

An example of U phase waveform is shown in Figure 1.15.8, and the description of waveform output workings is given below. Set "1" in DU0 (bit 0 at $034A_{16}$), and set "0" in DUB0 (bit 1 at $034A_{16}$). In addition, set "0" in DU1 (bit 0 at $034A_{16}$) and set "1" in DUB1 (bit 1 at $034A_{16}$).

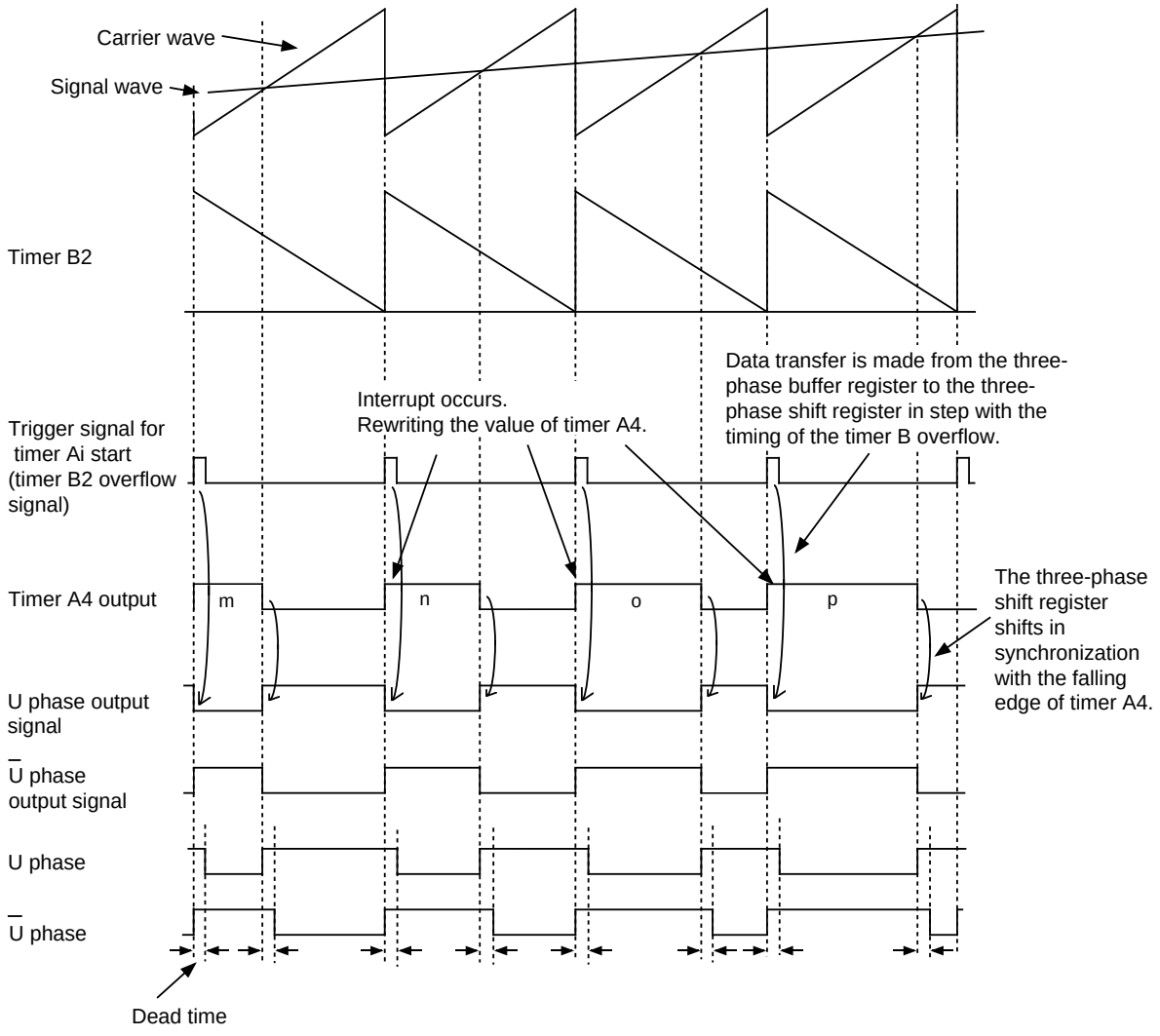
When the timber B2 counter's content becomes 0000_{16}, timer B2 generates an interrupt, and timer A4 starts outputting one-shot pulses at the same time. In this instance, the contents of the three-phase buffer registers DU1 and DU0 are set in the three-phase output shift register (U phase), and the contents of DUB1 and DUB0 are set in the three-phase output shift register ($\overline{U}$ phase). After this, the three-phase buffer register's content is set in the three-phase shift register every time the timer B2 counter's content becomes 0000_{16}.

The value of DU0 and that of DUB0 are output to the U terminal ($P8_0$) and to the $\overline{U}$ terminal ($P8_1$) respectively. When the timer A4 counter counts the value written to timer A4 ($038F_{16}$, $038E_{16}$) and when timer A4 finishes outputting one-shot pulses, the three-phase output shift register's content is shifted one position, and the value of DU1 and that of DUB1 are output to the U phase output signal and to the $\overline{U}$ output signal respectively. At this time, one-shot pulses are output from the timer for setting dead time used for setting the time over which the "L" level of the U phase waveform doesn't lap over the "L" level of the $\overline{U}$ phase waveform, which has the opposite phase of the former. The U phase waveform output that started from the "H" level keeps its level until the timer for setting dead time finishes outputting one-shot pulses even though the three-phase output shift register's content changes from "1" to "0 "by the effect of the one-shot pulses. When the timer for setting dead time finishes outputting one-shot pulses, 0 already shifted in the three-phase shift register goes effective, and the U phase waveform changes to the "L" level. When the timer B2 counter's content becomes 0000_{16}, the contents of the three-phase buffer registers DU1 and DU0 are set in the three-phase output shift register (U phase), and the contents of DUB1 and DUB0 are set in the three-phase output shift register ($\overline{U}$ phase) again.

A U phase waveform is generated by these workings repeatedly. With the exception that the three-phase output shift register on the $\overline{U}$ phase side is used, the workings in generating a $\overline{U}$ phase waveform, which has the opposite phase of the U phase waveform, are the same as in generating a U phase waveform. In this way, a waveform can be picked up from the applicable terminal in a manner in which the "L" level of the U phase waveform doesn't lap over that of the $\overline{U}$ phase waveform, which has the opposite phase of the U phase waveform. The width of the "L" level too can be adjusted by varying the values of timer B2 and timer A4. In dealing with the V and W phases, and $\overline{V}$ and $\overline{W}$ phases, the latter are of opposite phase of the former, have the corresponding timers work similarly to dealing with the U and $\overline{U}$ phases to generate an intended waveform.

Setting "1" both in DUB0 and in DUB1 provides a means to output the U phase alone and to fix the $\overline{U}$ phase output to "H" as shown in Figure 1.15.9.

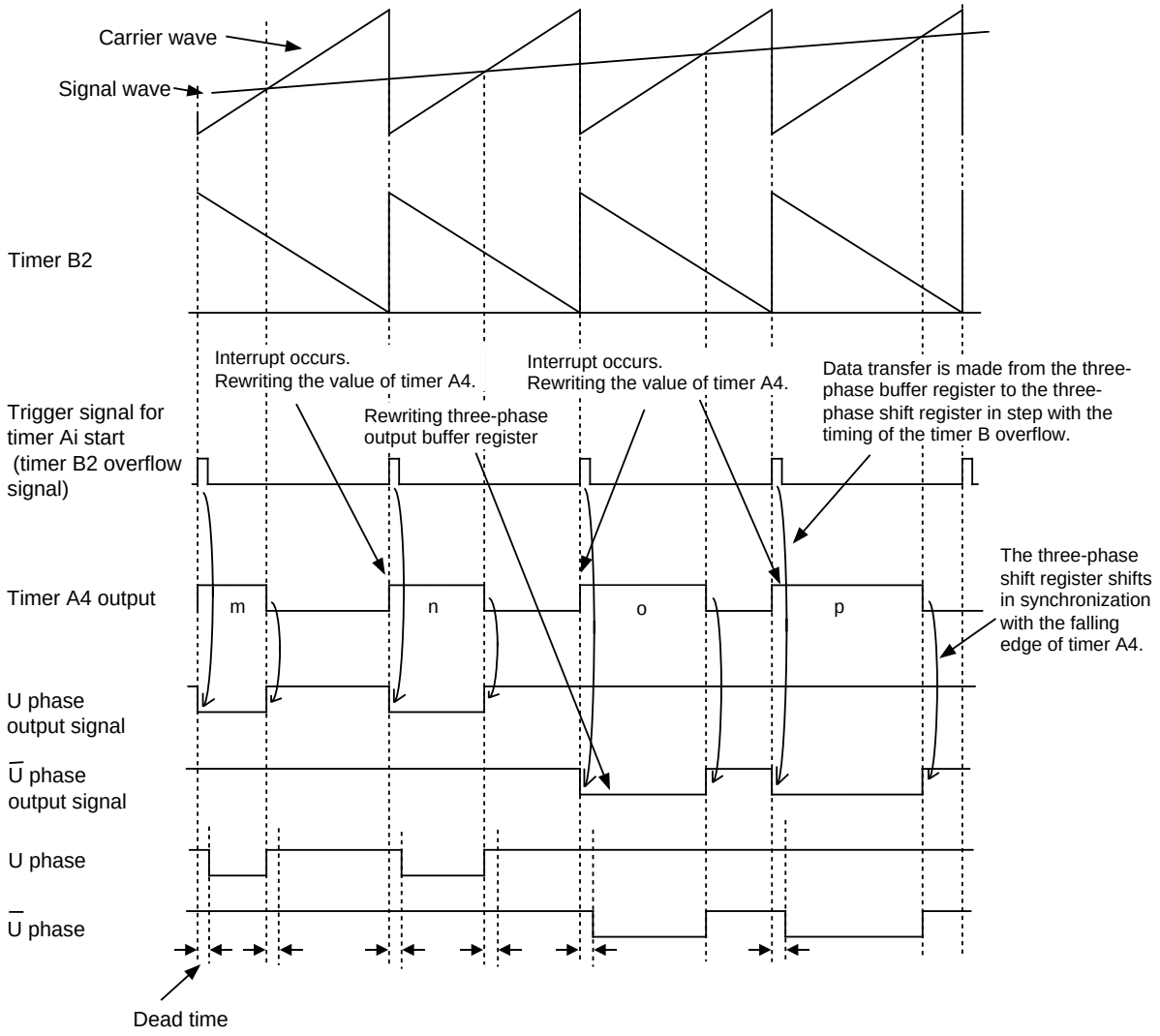
A carrier wave of sawtooth waveform



Note: Set to sawtooth modulation mode and to three-phase mode 0.

Figure 1.15.8. Timing chart of operation (3)

A carrier wave of sawtooth waveform



Note: Set to sawtooth modulation mode and to three-phase mode 0.

Figure 1.15.9. Timing chart of operation (4)

Serial I/O

Serial I/O is configured as five channels: UART0, UART1, UART2, S I/O3 and S I/O4.

UART0 to 2

UART0, UART1 and UART2 each have an exclusive timer to generate a transfer clock, so they operate independently of each other.

Figure 1.16.1 shows the block diagram of UART0, UART1 and UART2. Figures 1.16.2 and 1.16.3 show the block diagram of the transmit/receive unit.

UARTi (i = 0 to 2) has two operation modes: a clock synchronous serial I/O mode and a clock asynchronous serial I/O mode (UART mode). The contents of the serial I/O mode select bits (bits 0 to 2 at addresses $03A0_{16}$, $03A8_{16}$ and 0378_{16}) determine whether UARTi is used as a clock synchronous serial I/O or as a UART. Although a few functions are different, UART0, UART1 and UART2 have almost the same functions. UART2, in particular, is used for the SIM interface with some extra settings added in clock-asynchronous serial I/O mode (Note). It also has the bus collision detection function that generates an interrupt request if the TxD pin and the RxD pin are different in level.

Table 1.16.1 shows the comparison of functions of UART0 through UART2, and Figures 1.16.4 to 1.16.9 show the registers related to UARTi.

Note: SIM : Subscriber Identity Module

Table 1.16.1. Comparison of functions of UART0 through UART2

Function	UART0	UART1	UART2
CLK polarity selection	Possible (Note 1)	Possible (Note 1)	Possible (Note 1)
LSB first / MSB first selection	Possible (Note 1)	Possible (Note 1)	Possible (Note 2)
Continuous receive mode selection	Possible (Note 1)	Possible (Note 1)	Possible (Note 1)
Transfer clock output from multiple pins selection	Impossible	Possible (Note 1)	Impossible
Serial data logic switch	Impossible	Impossible	Possible (Note 4)
Sleep mode selection	Possible (Note 3)	Possible (Note 3)	Impossible
TxD, RxD I/O polarity switch	Impossible	Impossible	Possible
TxD, RxD port output format	CMOS output	CMOS output	N-channel open-drain output
Parity error signal output	Impossible	Impossible	Possible (Note 4)
Bus collision detection	Impossible	Impossible	Possible

Note 1: Only when clock synchronous serial I/O mode.
Note 2: Only when clock synchronous serial I/O mode and 8-bit UART mode.
Note 3: Only when UART mode.
Note 4: Using for SIM interface.

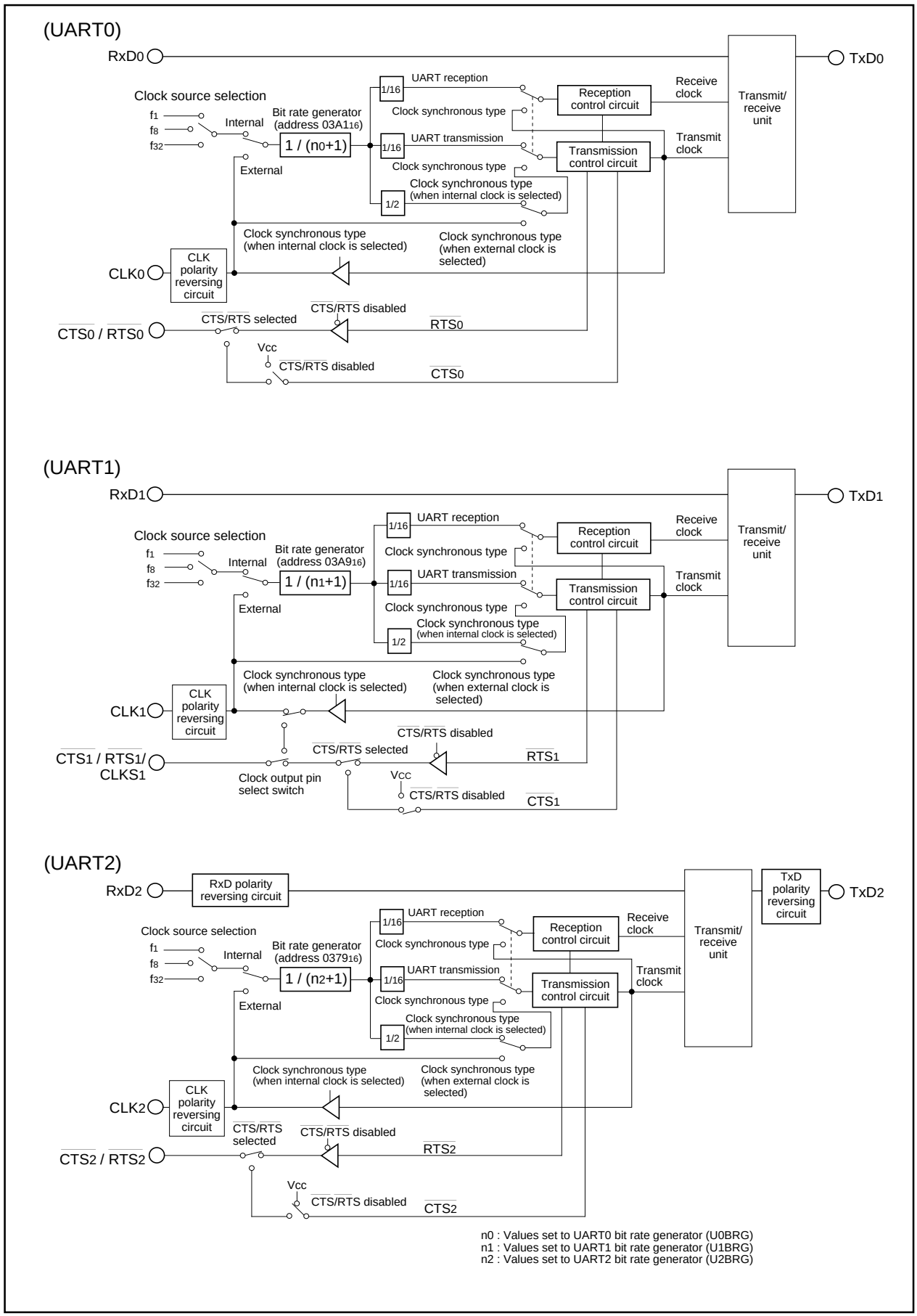


Figure 1.16.1. Block diagram of UARTi (i = 0 to 2)

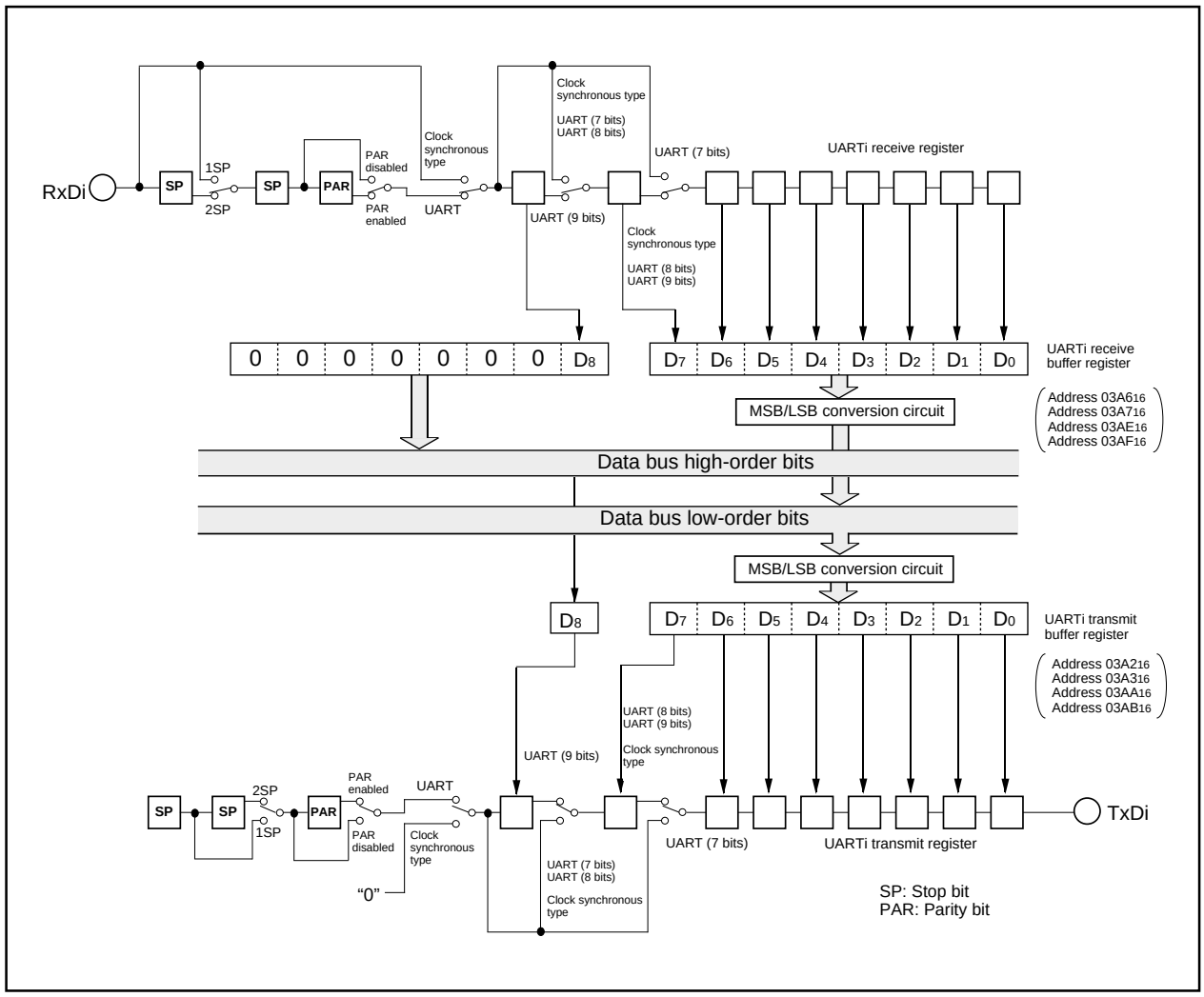


Figure 1.16.2. Block diagram of UARTi (i = 0, 1) transmit/receive unit

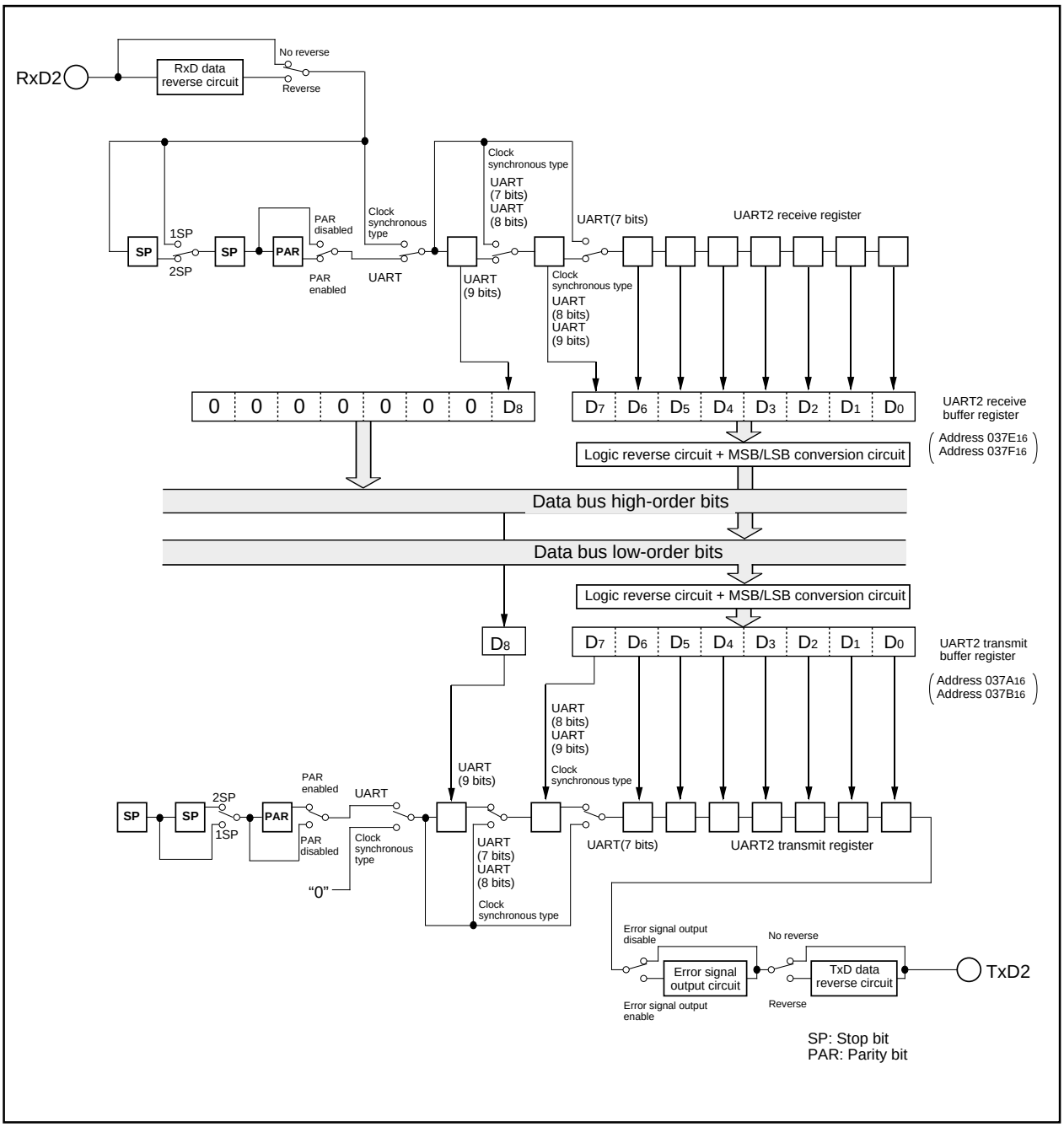
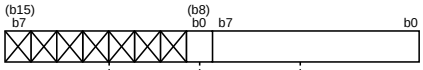


Figure 1.16.3. Block diagram of UART2 transmit/receive unit

UARTi transmit buffer register (Note)

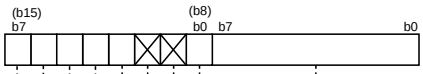


Symbol	Address	When reset
U0TB	$03A3_{16}$, $03A2_{16}$	Indeterminate
U1TB	$03AB_{16}$, $03AA_{16}$	Indeterminate
U2TB	$037B_{16}$, $037A_{16}$	Indeterminate

Function	R	W
Transmit data	×	○
Nothing is assigned. In an attempt to write to these bits, write "0". The value, if read, turns out to be indeterminate.	—	—

Note: Use MOV instruction to write to this register.

UARTi receive buffer register

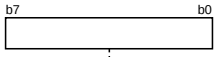


Symbol	Address	When reset
U0RB	$03A7_{16}$, $03A6_{16}$	Indeterminate
U1RB	$03AF_{16}$, $03AE_{16}$	Indeterminate
U2RB	$037F_{16}$, $037E_{16}$	Indeterminate

Bit symbol	Bit name	Function (During clock synchronous serial I/O mode)	Function (During UART mode)	R	W
—	———	Receive data	Receive data	○	×
Nothing is assigned. In an attempt to write to these bits, write "0". The value, if read, turns out to be "0".				—	—
ABT	Arbitration lost detecting flag (Note 2)	0 : Not detected 1 : Detected	Invalid	○	○
OER	Overrun error flag (Note 1)	0 : No overrun error 1 : Overrun error found	0 : No overrun error 1 : Overrun error found	○	×
FER	Framing error flag (Note 1)	Invalid	0 : No framing error 1 : Framing error found	○	×
PER	Parity error flag (Note 1)	Invalid	0 : No parity error 1 : Parity error found	○	×
SUM	Error sum flag (Note 1)	Invalid	0 : No error 1 : Error found	○	×

Note 1: Bits 15 through 12 are set to "0" when the serial I/O mode select bit (bits 2 to 0 at addresses $03A0_{16}$, $03A8_{16}$ and 0378_{16}) are set to "000_2" or the receive enable bit is set to "0".
(Bit 15 is set to "0" when bits 14 to 12 all are set to "0".) Bits 14 and 13 are also set to "0" when the lower byte of the UARTi receive buffer register (addresses $03A6_{16}$, $03AE_{16}$ and $037E_{16}$) is read out.
Note 2: Arbitration lost detecting flag is allocated to U2RB and noting but "0" may be written. Nothing is assigned in bit 11 of U0RB and U1RB. When write, set "0". The value, if read, turns out to be "0".

UARTi bit rate generator (Note 1, 2)



Symbol	Address	When reset
U0BRG	$03A1_{16}$	Indeterminate
U1BRG	$03A9_{16}$	Indeterminate
U2BRG	0379_{16}	Indeterminate

Function	Values that can be set	R	W
Assuming that set value = n, BRGi divides the count source by n + 1	00_{16} to FF_{16}	×	○

Note 1: Write a value to this register while transmit/receive halts.
Note 2: Use MOV instruction to write to this register.

Figure 1.16.4. Serial I/O-related registers (1)

UARTi transmit/receive mode register

b7	b6	b5	b4	b3	b2	b1	b0

Symbol: UiMR(i=0,1) — Address: 03A0$_{16}$, 03A8$_{16}$ — When reset: 00$_{16}$

Bit symbol	Bit name	Function (During clock synchronous serial I/O mode)	Function (During UART mode)	R	W
SMD0	Serial I/O mode select bit	Must be fixed to 001	b2 b1 b0 1 0 0 : Transfer data 7 bits long 1 0 1 : Transfer data 8 bits long 1 1 0 : Transfer data 9 bits long 0 0 0 : Serial I/O invalid 0 1 0 : Must not be set. 0 1 1 : Must not be set. 1 1 1 : Must not be set.	○	○
SMD1		b2 b1 b0 0 0 0 : Serial I/O invalid 0 1 0 : Must not be set. 0 1 1 : Must not be set. 1 1 1 : Must not be set.		○	○
SMD2				○	○
CKDIR	Internal/external clock select bit	0 : Internal clock 1 : External clock (Note)	0 : Internal clock 1 : External clock (Note)	○	○
STPS	Stop bit length select bit	Invalid	0 : One stop bit 1 : Two stop bits	○	○
PRY	Odd/even parity select bit	Invalid	Valid when bit 6 = “1” 0 : Odd parity 1 : Even parity	○	○
PRYE	Parity enable bit	Invalid	0 : Parity disabled 1 : Parity enabled	○	○
SLEP	Sleep select bit	Must always be “0”	0 : Sleep mode deselected 1 : Sleep mode selected	○	○

Note : Set the corresponding port direction register to “0”.

UART2 transmit/receive mode register

b7	b6	b5	b4	b3	b2	b1	b0

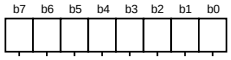
Symbol: U2MR — Address: 0378$_{16}$ — When reset: 00$_{16}$

Bit symbol	Bit name	Function (During clock synchronous serial I/O mode)	Function (During UART mode)	R	W
SMD0	Serial I/O mode select bit	Must be fixed to 001	b2 b1 b0 1 0 0 : Transfer data 7 bits long 1 0 1 : Transfer data 8 bits long 1 1 0 : Transfer data 9 bits long 0 0 0 : Serial I/O invalid 0 1 0 : Must not be set. 0 1 1 : Must not be set. 1 1 1 : Must not be set.	○	○
SMD1		b2 b1 b0 0 0 0 : Serial I/O invalid 0 1 0 : (Note 1) 0 1 1 : Must not be set. 1 1 1 : Must not be set.		○	○
SMD2				○	○
CKDIR	Internal/external clock select bit	0 : Internal clock 1 : External clock (Note 2)	Must always be “0”	○	○
STPS	Stop bit length select bit	Invalid	0 : One stop bit 1 : Two stop bits	○	○
PRY	Odd/even parity select bit	Invalid	Valid when bit 6 = “1” 0 : Odd parity 1 : Even parity	○	○
PRYE	Parity enable bit	Invalid	0 : Parity disabled 1 : Parity enabled	○	○
IOPOL	TxD, RxD I/O polarity reverse bit	0 : No reverse 1 : Reverse Usually set to “0”	0 : No reverse 1 : Reverse Usually set to “0”	○	○

Note 1: Bit 2 to bit 0 are set to “010$_2$” when I^{2}C mode is used.
Note 2: Set the corresponding port direction register to “0”.

Figure 1.16.5. Serial I/O-related registers (2)

UARTi transmit/receive control register 0

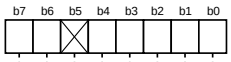


Symbol: UiC0(i=0,1) | Address: 03A4$_{16}$, 03AC$_{16}$ | When reset: 08$_{16}$

Bit symbol	Bit name	Function (During clock synchronous serial I/O mode)	Function (During UART mode)	R	W
CLK0	BRG count source select bit	b1 b0 0 0 : f_1 is selected 0 1 : f_8 is selected 1 0 : f_{32} is selected 1 1 : Must not be set.	b1 b0 0 0 : f_1 is selected 0 1 : f_8 is selected 1 0 : f_{32} is selected 1 1 : Must not be set.	○	○
CLK1				○	○
CRS	$\overline{CTS}/\overline{RTS}$ function select bit	Valid when bit 4 = "0" 0 : $\overline{CTS}$ function is selected (Note 1) 1 : $\overline{RTS}$ function is selected (Note 2)	Valid when bit 4 = "0" 0 : $\overline{CTS}$ function is selected (Note 1) 1 : $\overline{RTS}$ function is selected (Note 2)	○	○
TXEPT	Transmit register empty flag	0 : Data present in transmit register (during transmission) 1 : No data present in transmit register (transmission completed)	0 : Data present in transmit register (during transmission) 1 : No data present in transmit register (transmission completed)	○	×
CRD	$\overline{CTS}/\overline{RTS}$ disable bit	0 : $\overline{CTS}/\overline{RTS}$ function enabled 1 : $\overline{CTS}/\overline{RTS}$ function disabled ($P6_0$ and $P6_4$ function as programmable I/O port)	0 : $\overline{CTS}/\overline{RTS}$ function enabled 1 : $\overline{CTS}/\overline{RTS}$ function disabled ($P6_0$ and $P6_4$ function as programmable I/O port)	○	○
NCH	Data output select bit	0 : TxDi pin is CMOS output 1 : TxDi pin is N-channel open-drain output	0: TxDi pin is CMOS output 1: TxDi pin is N-channel open-drain output	○	○
CKPOL	CLK polarity select bit	0 : Transmit data is output at falling edge of transfer clock and receive data is input at rising edge 1 : Transmit data is output at rising edge of transfer clock and receive data is input at falling edge	Must always be "0"	○	○
UFORM	Transfer format select bit	0 : LSB first 1 : MSB first	Must always be "0"	○	○

Note 1: Set the corresponding port direction register to "0".
Note 2: The settings of the corresponding port register and port direction register are invalid.

UART2 transmit/receive control register 0



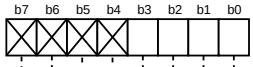
Symbol: U2C0 | Address: 037C$_{16}$ | When reset: 08$_{16}$

Bit symbol	Bit name	Function (During clock synchronous serial I/O mode)	Function (During UART mode)	R	W
CLK0	BRG count source select bit	b1 b0 0 0 : f_1 is selected 0 1 : f_8 is selected 1 0 : f_{32} is selected 1 1 : Must not be set.	b1 b0 0 0 : f_1 is selected 0 1 : f_8 is selected 1 0 : f_{32} is selected 1 1 : Must not be set.	○	○
CLK1				○	○
CRS	$\overline{CTS}/\overline{RTS}$ function select bit	Valid when bit 4 = "0" 0 : $\overline{CTS}$ function is selected (Note 1) 1 : $\overline{RTS}$ function is selected (Note 2)	Valid when bit 4 = "0" 0 : $\overline{CTS}$ function is selected (Note 1) 1 : $\overline{RTS}$ function is selected (Note 2)	○	○
TXEPT	Transmit register empty flag	0 : Data present in transmit register (during transmission) 1 : No data present in transmit register (transmission completed)	0 : Data present in transmit register (during transmission) 1 : No data present in transmit register (transmission completed)	○	×
CRD	$\overline{CTS}/\overline{RTS}$ disable bit	0 : $\overline{CTS}/\overline{RTS}$ function enabled 1 : $\overline{CTS}/\overline{RTS}$ function disabled ($P7_3$ functions programmable I/O port)	0 : $\overline{CTS}/\overline{RTS}$ function enabled 1 : $\overline{CTS}/\overline{RTS}$ function disabled ($P7_3$ functions programmable I/O port)	○	○
Nothing is assigned. In an attempt to write to this bit, write "0". The value, if read, turns out to be "0".				–	–
CKPOL	CLK polarity select bit	0 : Transmit data is output at falling edge of transfer clock and receive data is input at rising edge 1 : Transmit data is output at rising edge of transfer clock and receive data is input at falling edge	Must always be "0"	○	○
UFORM	Transfer format select bit (Note 3)	0 : LSB first 1 : MSB first	0 : LSB first 1 : MSB first	○	○

Note 1: Set the corresponding port direction register to "0".
Note 2: The settings of the corresponding port register and port direction register are invalid.
Note 3: Only clock synchronous serial I/O mode and 8-bit UART mode are valid.

Figure 1.16.6. Serial I/O-related registers (3)

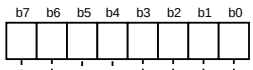
UARTi transmit/receive control register 1



Symbol: UiC1(i=0,1) | Address: 03A5$_{16}$,03AD$_{16}$ | When reset: 02$_{16}$

Bit symbol	Bit name	Function (During clock synchronous serial I/O mode)	Function (During UART mode)	R	W
TE	Transmit enable bit	0 : Transmission disabled 1 : Transmission enabled	0 : Transmission disabled 1 : Transmission enabled	○	○
TI	Transmit buffer empty flag	0 : Data present in transmit buffer register 1 : No data present in transmit buffer register	0 : Data present in transmit buffer register 1 : No data present in transmit buffer register	○	×
RE	Receive enable bit	0 : Reception disabled 1 : Reception enabled	0 : Reception disabled 1 : Reception enabled	○	○
RI	Receive complete flag	0 : No data present in receive buffer register 1 : Data present in receive buffer register	0 : No data present in receive buffer register 1 : Data present in receive buffer register	○	×
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be “0”.				—	—

UART2 transmit/receive control register 1



Symbol: U2C1 | Address: 037D$_{16}$ | When reset: 02$_{16}$

Bit symbol	Bit name	Function (During clock synchronous serial I/O mode)	Function (During UART mode)	R	W
TE	Transmit enable bit	0 : Transmission disabled 1 : Transmission enabled	0 : Transmission disabled 1 : Transmission enabled	○	○
TI	Transmit buffer empty flag	0 : Data present in transmit buffer register 1 : No data present in transmit buffer register	0 : Data present in transmit buffer register 1 : No data present in transmit buffer register	○	×
RE	Receive enable bit	0 : Reception disabled 1 : Reception enabled	0 : Reception disabled 1 : Reception enabled	○	○
RI	Receive complete flag	0 : No data present in receive buffer register 1 : Data present in receive buffer register	0 : No data present in receive buffer register 1 : Data present in receive buffer register	○	×
U2IRS	UART2 transmit interrupt cause select bit	0 : Transmit buffer empty (TI = 1) 1 : Transmit is completed (TXEPT = 1)	0 : Transmit buffer empty (TI = 1) 1 : Transmit is completed (TXEPT = 1)	○	○
U2RRM	UART2 continuous receive mode enable bit	0 : Continuous receive mode disabled 1 : Continuous receive mode enabled	Must always be "0"	○	○
U2LCH	Data logic select bit	0 : No reverse 1 : Reverse	0 : No reverse 1 : Reverse	○	○
U2ERE	Error signal output enable bit	Must be fixed to “0”	0 : Output disabled 1 : Output enabled	○	○

Figure 1.16.7. Serial I/O-related registers (4)

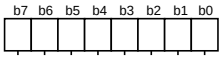
UART transmit/receive control register 2

Symbol: UCON | Address: 03B0$_{16}$ | When reset: X0000000$_2$

Bit symbol	Bit name	Function (During clock synchronous serial I/O mode)	Function (During UART mode)	R	W
U0IRS	UART0 transmit interrupt cause select bit	0 : Transmit buffer empty (TI = 1) 1 : Transmission completed (TXEPT = 1)	0 : Transmit buffer empty (TI = 1) 1 : Transmission completed (TXEPT = 1)	○	○
U1IRS	UART1 transmit interrupt cause select bit	0 : Transmit buffer empty (TI = 1) 1 : Transmission completed (TXEPT = 1)	0 : Transmit buffer empty (TI = 1) 1 : Transmission completed (TXEPT = 1)	○	○
U0RRM	UART0 continuous receive mode enable bit	0 : Continuous receive mode disabled 1 : Continuous receive mode enable	Must always be “0”	○	○
U1RRM	UART1 continuous receive mode enable bit	0 : Continuous receive mode disabled 1 : Continuous receive mode enabled	Must always be “0”	○	○
CLKMD0	CLK/CLKS select bit 0	Valid when bit 5 = “1” 0 : Clock output to CLK1 1 : Clock output to CLKS1	Invalid	○	○
CLKMD1	CLK/CLKS select bit 1 (Note)	0 : Normal mode (CLK output is CLK1 only) 1 : Transfer clock output from multiple pins function selected	Must always be “0”	○	○
Reserved bit		Must always be set to “0”		○	○
Nothing is assigned. In an attempt to write to this bit, write “0”. The value, if read, turns out to be indeterminate.				–	–

Note: When using multiple pins to output the transfer clock, the following requirements must be met:
• UART1 internal/external clock select bit (bit 3 at address 03A8$_{16}$) = “0”.

UART2 special mode register



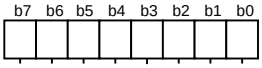
Symbol: U2SMR | Address: 0377$_{16}$ | When reset: 00$_{16}$

Bit symbol	Bit name	Function (During clock synchronous serial I/O mode)	Function (During UART mode)	R	W
IICM	I^{2}C mode select bit	0 : Normal mode 1 : I^{2}C mode	Must always be “0”	○	○
ABC	Arbitration lost detecting flag control bit	0 : Update per bit 1 : Update per byte	Must always be “0”	○	○
BBS	Bus busy flag	0 : STOP condition detected 1 : START condition detected	Must always be “0”	○	○ (Note1)
LSYN	SCLL sync output enable bit	0 : Disabled 1 : Enabled	Must always be “0”	○	○
ABSCS	Bus collision detect sampling clock select bit	Must always be “0”	0 : Rising edge of transfer clock 1 : Underflow signal of timer A0	○	○
ACSE	Auto clear function select bit of transmit enable bit	Must always be “0”	0 : No auto clear function 1 : Auto clear at occurrence of bus collision	○	○
SSS	Transmit start condition select bit	Must always be “0”	0 : Ordinary 1 : Falling edge of RxD$_2$	○	○
SDDS	SDA digital delay select bit (Note 2, Note 3)	0 : Analog delay output is selected 1 : Digital delay output is selected (must always be “0” when not using I^{2}C mode)	Must always be “0”	○	○

Note 1: Nothing but “0” may be written.
Note 2: When not in I^{2}C mode, do not set this bit by writing a “1”. During normal mode, fix it to “0”. When this bit = “0”, UART2 special mode register 3 (U2SMR3 at address 0375$_{16}$) bits 7 to 5 (DL2 to DL0 = SDA digital delay setup bits) are initialized to “000”, with the analog delay circuit selected. Also, when SDDS = “0”, the U2SMR3 register cannot be read or written to.
Note 3: When analog delay is selected, only the analog delay value is effective; when digital delay is selected, only the digital delay value is effective.

Figure 1.16.8. Serial I/O-related registers (5)

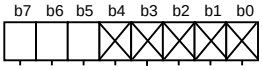
UART2 special mode register 2 (I^2C bus exclusive use register)



Symbol U2SMR2 | Address 037_{16} | When reset 00_{16}

Bit symbol	Bit name	Function (I^2C bus exclusive use)	R	W
IICM2	I^2C mode select bit 2	Refer to Table 1.16.11	○	○
CSC	Clock-synchronous bit	0 : Disabled 1 : Enabled	○	○
SWC	SCL wait output bit	0 : Disabled 1 : Enabled	○	○
ALS	SDA output stop bit	0 : Disabled 1 : Enabled	○	○
STAC	UART2 initialization bit	0 : Disabled 1 : Enabled	○	○
SWC2	SCL wait output bit 2	0: UART2 clock 1: 0 output	○	○
SDHI	SDA output disable bit	0: Enabled 1: Disabled (high impedance)	○	○
SHTC	Start/stop condition control bit	Set this bit to "1" in I^2C mode (refer to Table 1.16.12)	○	○

UART2 special mode register 3 (I^2C bus exclusive use register)



Symbol U2SMR3 | Address 037_{16} | When reset Indeterminate (However, when SDDS = "1", the initial value is "00_{16}")

Bit symbol	Bit name	Function (I^2C bus exclusive use register)	R	W
Nothing is assigned. In an attempt to write to these bits, write "0". The value, if read, turns out to be indeterminate. However, when SDDS = "1", the value "0" is read out (Note 1)			—	—
DL0	SDA digital delay setup bit (Note 1, Note 2, Note 3, Note 4)	b7 b6 b5 0 0 0 : Analog delay is selected 0 0 1 : 1 to 2 cycle(s) of 1/f(X_{IN}) 0 1 0 : 2 to 3 cycles of 1/f(X_{IN}) 0 1 1 : 3 to 4 cycles of 1/f(X_{IN}) 1 0 0 : 4 to 5 cycles of 1/f(X_{IN}) 1 0 1 : 5 to 6 cycles of 1/f(X_{IN}) 1 1 0 : 6 to 7 cycles of 1/f(X_{IN}) 1 1 1 : 7 to 8 cycles of 1/f(X_{IN}) } Digital delay is selected	○	○
DL1			○	○
DL2			○	○

Note 1: This bit can be read or written to when UART2 special mode register (U2SMR at address 037_{16}) bit 7 (SDDS: SDA digital delay select bit) = "1". When the initial value of UART2 special mode register 3 (U2SMR3) is read after setting SDDS = "1", the value is "00_{16}". When writing to UART2 special mode register 3 (U2SMR3) after setting SDDS = "1", be sure to write 0's to bits 0–4. When SDDS = "0", this register cannot be written to; when read, the value is indeterminate.

Note 2: These bits are initialized to "000" when SDDS = "0", with the analog delay circuit selected. After a reset, these bits are set to "000", with the analog delay circuit selected. However, because these bits can be read only when SDDS = "1", the value read from these bits when SDDS = "0" is indeterminate.

Note 3: When analog delay is selected, only the analog delay value is effective; when digital delay is selected, only the digital delay value is effective.

Note 4: The amount of delay varies with the load on SCL and SDA pins. Also, when using an external clock, the amount of delay increases by about 100 ns, so be sure to take this into account when using the device.

Figure 1.16.9. Serial I/O-related registers (6)

(1) Clock synchronous serial I/O mode

The clock synchronous serial I/O mode uses a transfer clock to transmit and receive data. Tables 1.16.2 and 1.16.3 list the specifications of the clock synchronous serial I/O mode. Figure 1.16.10 shows the UARTi transmit/receive mode register.

Table 1.16.2. Specifications of clock synchronous serial I/O mode (1)

Item	Specification
Transfer data format	• Transfer data length: 8 bits
Transfer clock	• When internal clock is selected (bit 3 at addresses 03A0$_{16}$, 03A8$_{16}$, 0378$_{16}$ = "0") : fi/ 2(n+1) (Note 1) fi = f_1, f_8, f_{32} • When external clock is selected (bit 3 at addresses 03A0$_{16}$, 03A8$_{16}$, 0378$_{16}$ = "1") : Input from CLKi pin
Transmission/reception control	• $\overline{CTS}$ function, $\overline{RTS}$ function, $\overline{CTS}$ and $\overline{RTS}$ function invalid: selectable
Transmission start condition	• To start transmission, the following requirements must be met: – Transmit enable bit (bit 0 at addresses 03A5$_{16}$, 03AD$_{16}$, 037D$_{16}$) = "1" – Transmit buffer empty flag (bit 1 at addresses 03A5$_{16}$, 03AD$_{16}$, 037D$_{16}$) = "0" – When $\overline{CTS}$ function selected, $\overline{CTS}$ input level = "L" • Furthermore, if external clock is selected, the following requirements must also be met: – CLKi polarity select bit (bit 6 at addresses 03A4$_{16}$, 03AC$_{16}$, 037C$_{16}$) = "0": CLKi input level = "H" – CLKi polarity select bit (bit 6 at addresses 03A4$_{16}$, 03AC$_{16}$, 037C$_{16}$) = "1": CLKi input level = "L"
Reception start condition	• To start reception, the following requirements must be met: – Receive enable bit (bit 2 at addresses 03A5$_{16}$, 03AD$_{16}$, 037D$_{16}$) = "1" – Transmit enable bit (bit 0 at addresses 03A5$_{16}$, 03AD$_{16}$, 037D$_{16}$) = "1" – Transmit buffer empty flag (bit 1 at addresses 03A5$_{16}$, 03AD$_{16}$, 037D$_{16}$) = "0" • Furthermore, if external clock is selected, the following requirements must also be met: – CLKi polarity select bit (bit 6 at addresses 03A4$_{16}$, 03AC$_{16}$, 037C$_{16}$) = "0": CLKi input level = "H" – CLKi polarity select bit (bit 6 at addresses 03A4$_{16}$, 03AC$_{16}$, 037C$_{16}$) = "1": CLKi input level = "L"
Interrupt request generation timing	• When transmitting – Transmit interrupt cause select bit (bits 0, 1 at address 03B0$_{16}$, bit 4 at address 037D$_{16}$) = "0": Interrupts requested when data transfer from UARTi transfer buffer register to UARTi transmit register is completed – Transmit interrupt cause select bit (bits 0, 1 at address 03B0$_{16}$, bit 4 at address 037D$_{16}$) = "1": Interrupts requested when data transmission from UARTi transfer register is completed • When receiving – Interrupts requested when data transfer from UARTi receive register to UARTi receive buffer register is completed
Error detection	• Overrun error (Note 2) This error occurs when the next data is ready before contents of UARTi receive buffer register are read out

Note 1: "n" denotes the value 00$_{16}$ to FF$_{16}$ that is set to the UART bit rate generator.

Note 2: If an overrun error occurs, the UARTi receive buffer will have the next data written in. Note also that the UARTi receive interrupt request bit does not change.

Table 1.16.3. Specifications of clock synchronous serial I/O mode (2)

Item	Specification
Select function	• CLK polarity selection Whether transmit data is output/input timing at the rising edge or falling edge of the transfer clock can be selected • LSB first/MSB first selection Whether transmission/reception begins with bit 0 or bit 7 can be selected • Continuous receive mode selection Reception is enabled simultaneously by a read from the receive buffer register • Transfer clock output from multiple pins selection (UART1) UART1 transfer clock can be chosen by software to be output from one of the two pins set • Switching serial data logic (UART2) Whether to reverse data in writing to the transmission buffer register or reading the reception buffer register can be selected. • TxD, RxD I/O polarity reverse (UART2) This function is reversing TxD port output and RxD port input. All I/O data level is reversed.

UARTi transmit/receive mode registers

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

Symbol: UiMR(i=0,1) — Address: $03A0_{16}$, $03A8_{16}$ — When reset: 00_{16}

Bit symbol	Bit name	Function	R	W
SMD0	Serial I/O mode select bit	b2 b1 b0 0 0 1 : Clock synchronous serial I/O mode	○	○
SMD1			○	○
SMD2			○	○
CKDIR	Internal/external clock select bit	0 : Internal clock 1 : External clock (Note)	○	○
STPS	Invalid in clock synchronous serial I/O mode		○	○
PRY			○	○
PRYE			○	○
SLEP	0 (Must always be “0” in clock synchronous serial I/O mode)		○	○

Note : Set the corresponding port direction register to “0”.

UART2 transmit/receive mode register

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

Symbol: U2MR — Address: 0378_{16} — When reset: 00_{16}

Bit symbol	Bit name	Function	R	W
SMD0	Serial I/O mode select bit	b2 b1 b0 0 0 1 : Clock synchronous serial I/O mode	○	○
SMD1			○	○
SMD2			○	○
CKDIR	Internal/external clock select bit	0 : Internal clock 1 : External clock (Note 2)	○	○
STPS	Invalid in clock synchronous serial I/O mode		○	○
PRY			○	○
PRYE			○	○
IOPOL	TxD, RxD I/O polarity reverse bit (Note 1)	0 : No reverse 1 : Reverse	○	○

Note 1: Usually set to “0”.
Note 2: Set the corresponding port direction register to “0”.

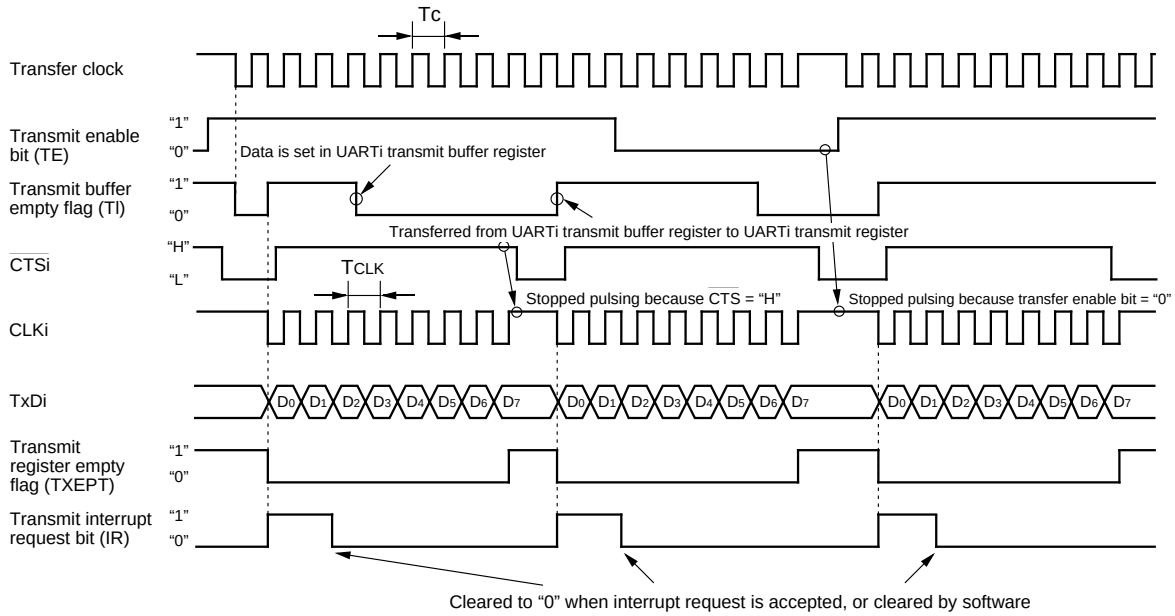
Figure 1.16.10. UARTi transmit/receive mode register in clock synchronous serial I/O mode

Table 1.16.4 lists the functions of the input/output pins during clock synchronous serial I/O mode. This table shows the pin functions when the transfer clock output from multiple pins function is not selected. Note that for a period from when the UARTi operation mode is selected to when transfer starts, the TxDi pin outputs an "H". (If the N-channel open-drain is selected, this pin is in floating state.)

Table 1.16.4. Input/output pin functions in clock synchronous serial I/O mode
(when transfer clock output from multiple pins is not selected)

Pin name	Function	Method of selection
TxDi ($P6_3$, $P6_7$, $P7_0$)	Serial data output	(Outputs dummy data when performing reception only)
RxDi ($P6_2$, $P6_6$, $P7_1$)	Serial data input	Port $P6_2$, $P6_6$ and $P7_1$ direction register (bits 2 and 6 at address $03EE_{16}$, bit 1 at address $03EF_{16}$)= "0" (Can be used as an input port when performing transmission only)
CLKi ($P6_1$, $P6_5$, $P7_2$)	Transfer clock output	Internal/external clock select bit (bit 3 at address $03A0_{16}$, $03A8_{16}$, 0378_{16}) = "0"
	Transfer clock input	Internal/external clock select bit (bit 3 at address $03A0_{16}$, $03A8_{16}$, 0378_{16}) = "1" Port $P6_1$, $P6_5$ and $P7_2$ direction register (bits 1 and 5 at address $03EE_{16}$, bit 2 at address $03EF_{16}$) = "0"
$\overline{CTS}i/\overline{RTS}i$ ($P6_0$, $P6_4$, $P7_3$)	$\overline{CTS}$ input	$\overline{CTS}/\overline{RTS}$ disable bit (bit 4 at address $03A4_{16}$, $03AC_{16}$, $037C_{16}$) ="0" $\overline{CTS}/\overline{RTS}$ function select bit (bit 2 at address $03A4_{16}$, $03AC_{16}$, $037C_{16}$) = "0" Port $P6_0$, $P6_4$ and $P7_3$ direction register (bits 0 and 4 at address $03EE_{16}$, bit 3 at address $03EF_{16}$) = "0"
	$\overline{RTS}$ output	$\overline{CTS}/\overline{RTS}$ disable bit (bit 4 at address $03A4_{16}$, $03AC_{16}$, $037C_{16}$) = "0" $\overline{CTS}/\overline{RTS}$ function select bit (bit 2 at address $03A4_{16}$, $03AC_{16}$, $037C_{16}$) = "1"
	Programmable I/O port	$\overline{CTS}/\overline{RTS}$ disable bit (bit 4 at address $03A4_{16}$, $03AC_{16}$, $037C_{16}$) = "1"

• Example of transmit timing (when internal clock is selected)

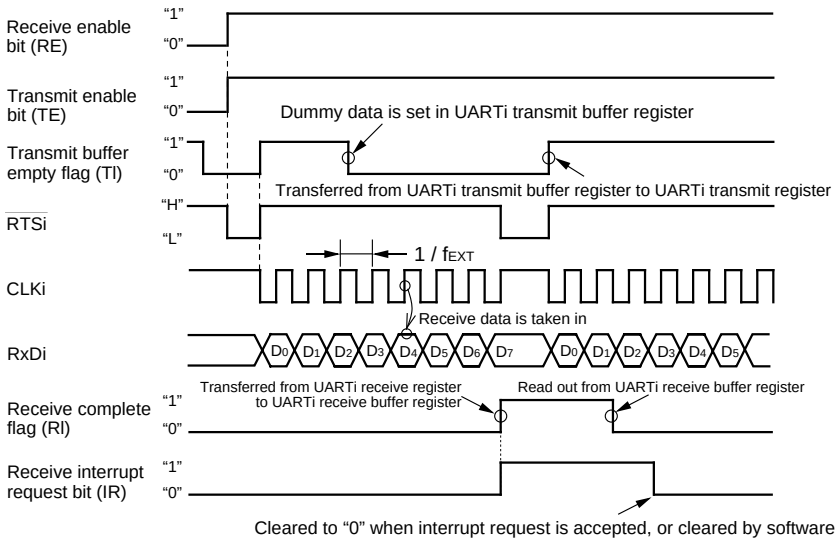


Shown in ( ) are bit symbols.

The above timing applies to the following settings:
- Internal clock is selected.
- CTS function is selected.
- CLK polarity select bit = "0".
- Transmit interrupt cause select bit = "0".

Tc = TCLK = 2(n + 1) / fi
fi: frequency of BRGi count source (f1, f8, f32)
n: value set to BRGi

• Example of receive timing (when external clock is selected)



Shown in ( ) are bit symbols.

The above timing applies to the following settings:
- External clock is selected.
- RTS function is selected.
- CLK polarity select bit = "0".

fEXT: frequency of external clock

Meet the following conditions are met when the CLK input before data reception = "H"
- Transmit enable bit → "1"
- Receive enable bit → "1"
- Dummy data write to UARTi transmit buffer register

Figure 1.16.11. Typical transmit/receive timings in clock synchronous serial I/O mode

(a) Polarity select function

As shown in Figure 1.16.12, the CLK polarity select bit (bit 6 at addresses 03A4$_{16}$, 03AC$_{16}$, 037C$_{16}$) allows selection of the polarity of the transfer clock.

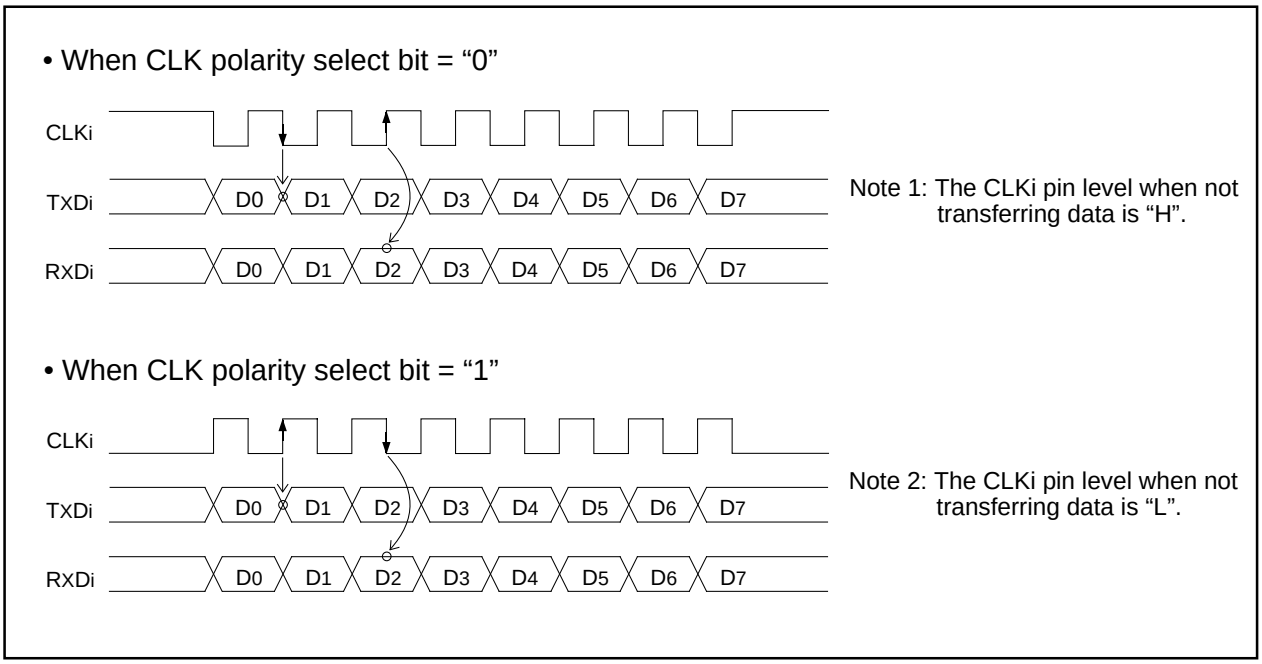


Figure 1.16.12. Polarity of transfer clock

(b) LSB first/MSB first select function

As shown in Figure 1.16.13, when the transfer format select bit (bit 7 at addresses 03A4$_{16}$, 03AC$_{16}$, 037C$_{16}$) = “0”, the transfer format is “LSB first”; when the bit = “1”, the transfer format is “MSB first”.

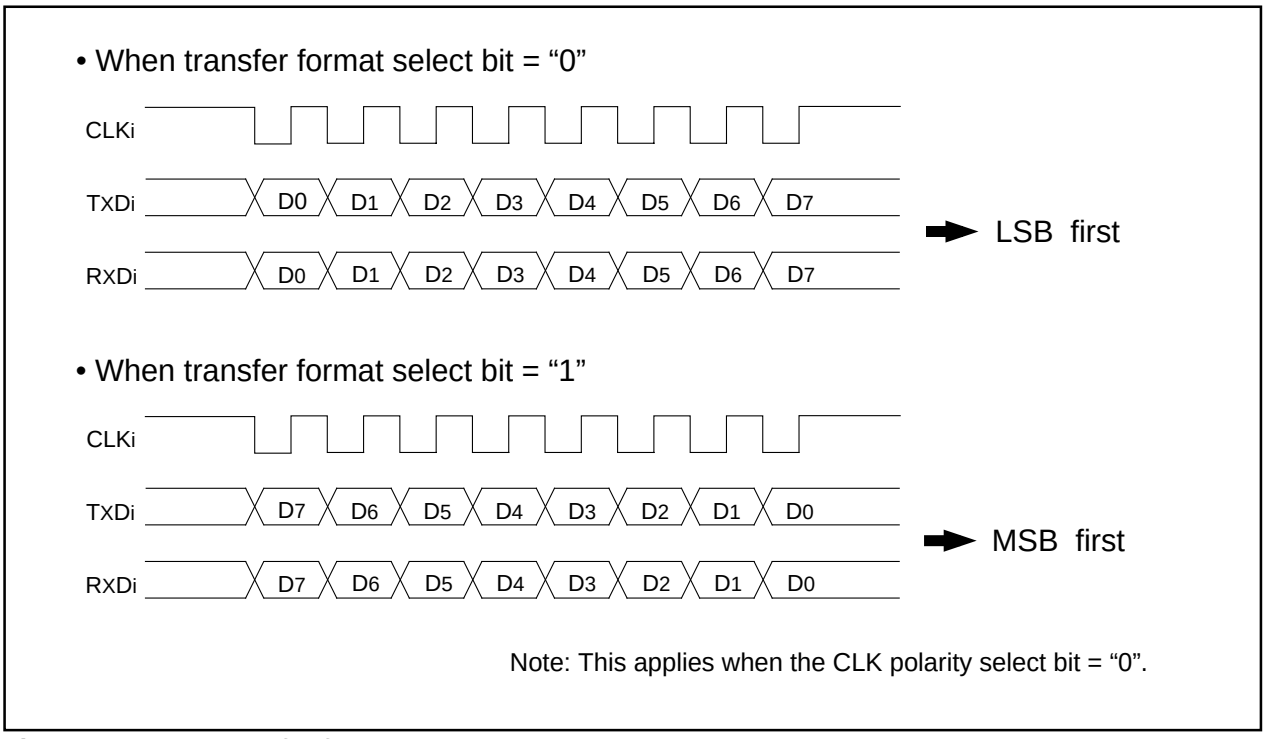


Figure 1.16.13. Transfer format

(c) Transfer clock output from multiple pins function (UART1)

This function allows the setting two transfer clock output pins and choosing one of the two to output a clock by using the CLK and CLKS select bit (bits 4 and 5 at address 03B0$_{16}$). (See Figure 1.16.14.) The multiple pins function is valid only when the internal clock is selected for UART1.

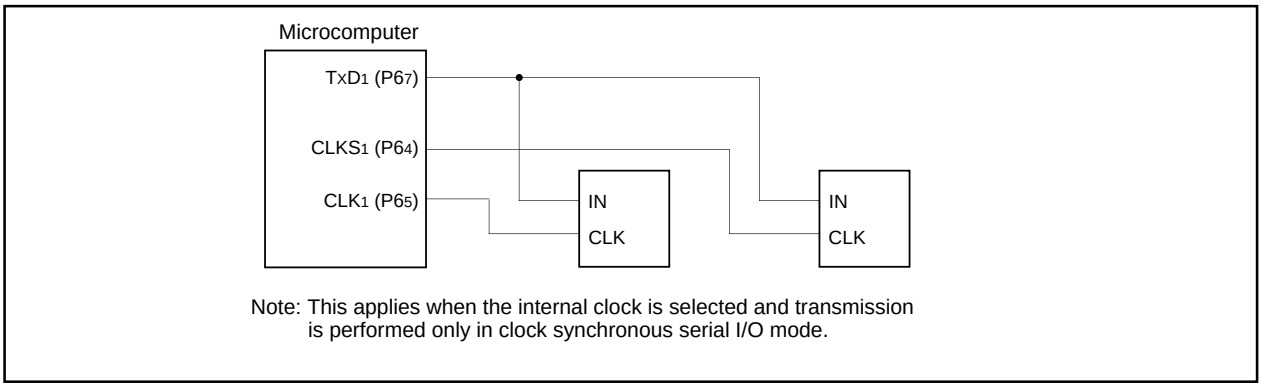


Figure 1.16.14. The transfer clock output from the multiple pins function usage

(d) Continuous receive mode

If the continuous receive mode enable bit (bits 2 and 3 at address 03B0$_{16}$, bit 5 at address 037D$_{16}$) is set to "1", the unit is placed in continuous receive mode. In this mode, when the receive buffer register is read out, the unit simultaneously goes to a receive enable state without having to set dummy data to the transmit buffer register back again.

(e) Serial data logic switch function (UART2)

When the data logic select bit (bit6 at address 037D$_{16}$) = "1", and writing to transmit buffer register or reading from receive buffer register, data is reversed. Figure 1.16.15 shows the example of serial data logic switch timing.

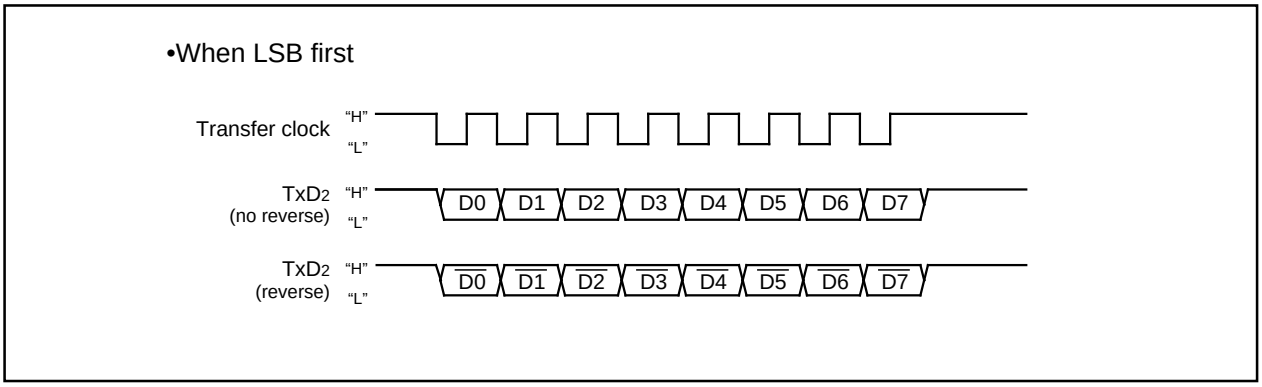


Figure 1.16.15. Serial data logic switch timing

(2) Clock asynchronous serial I/O (UART) mode

The UART mode allows transmitting and receiving data after setting the desired transfer rate and transfer data format. Tables 1.16.5 and 1.16.6 list the specifications of the UART mode. Figure 1.16.16 shows the UARTi transmit/receive mode register.

Table 1.16.5. Specifications of UART Mode (1)

Item	Specification
Transfer data format	• Character bit (transfer data): 7 bits, 8 bits, or 9 bits as selected • Start bit: 1 bit • Parity bit: Odd, even, or nothing as selected • Stop bit: 1 bit or 2 bits as selected
Transfer clock	• When internal clock is selected (bit 3 at addresses $03A0_{16}$, $03A8_{16}$, 0378_{16} = "0") : fi/16(n+1) (Note 1) fi = f_1, f_8, f_{32} • When external clock is selected (bit 3 at addresses $03A0_{16}$, $03A8_{16}$ ="1") : f_{EXT}/16(n+1) (Note 1) (Note 2) (Do not set external clock for UART2)
Transmission/reception control	• $\overline{CTS}$ function, $\overline{RTS}$ function, $\overline{CTS}$ and $\overline{RTS}$ function invalid: selectable
Transmission start condition	• To start transmission, the following requirements must be met: - Transmit enable bit (bit 0 at addresses $03A5_{16}$, $03AD_{16}$, $037D_{16}$) = "1" - Transmit buffer empty flag (bit 1 at addresses $03A5_{16}$, $03AD_{16}$, $037D_{16}$) = "0" - When $\overline{CTS}$ function selected, $\overline{CTS}$ input level = "L"
Reception start condition	• To start reception, the following requirements must be met: - Receive enable bit (bit 2 at addresses $03A5_{16}$, $03AD_{16}$, $037D_{16}$) = "1" - Start bit detection
Interrupt request generation timing	• When transmitting - Transmit interrupt cause select bits (bits 0,1 at address $03B0_{16}$, bit4 at address $037D_{16}$) = "0": Interrupts requested when data transfer from UARTi transfer buffer register to UARTi transmit register is completed - Transmit interrupt cause select bits (bits 0, 1 at address $03B0_{16}$, bit4 at address $037D_{16}$) = "1": Interrupts requested when data transmission from UARTi transfer register is completed • When receiving - Interrupts requested when data transfer from UARTi receive register to UARTi receive buffer register is completed
Error detection	• Overrun error (Note 3) This error occurs when the next data is ready before contents of UARTi receive buffer register are read out • Framing error This error occurs when the number of stop bits set is not detected • Parity error This error occurs when if parity is enabled, the number of 1's in parity and character bits does not match the number of 1's set • Error sum flag This flag is set (= 1) when any of the overrun, framing, and parity errors is encountered

Note 1: 'n' denotes the value 00_{16} to FF_{16} that is set to the UARTi bit rate generator.

Note 2: f_{EXT} is input from the CLKi pin.

Note 3: If an overrun error occurs, the UARTi receive buffer will have the next data written in. Note also that the UARTi receive interrupt request bit does not change.

Table 1.16.6. Specifications of UART Mode (2)

Item	Specification
Select function	• Sleep mode selection (UART0, UART1) This mode is used to transfer data to and from one of multiple slave microcomputers • Serial data logic switch (UART2) This function is reversing logic value of transferring data. Start bit, parity bit and stop bit are not reversed. • TxD, RxD I/O polarity switch (UART2) This function is reversing TxD port output and RxD port input. All I/O data level is reversed.

UARTi transmit / receive mode registers

b7	b6	b5	b4	b3	b2	b1	b0

Symbol: UiMR(i=0,1) — Address: 03A0$_{16}$, 03A8$_{16}$ — When reset: 00$_{16}$

Bit symbol	Bit name	Function	R	W
SMD0	Serial I/O mode select bit	b2 b1 b0 1 0 0 : Transfer data 7 bits long 1 0 1 : Transfer data 8 bits long 1 1 0 : Transfer data 9 bits long	○	○
SMD1			○	○
SMD2			○	○
CKDIR	Internal / external clock select bit	0 : Internal clock 1 : External clock (Note)	○	○
STPS	Stop bit length select bit	0 : One stop bit 1 : Two stop bits	○	○
PRY	Odd / even parity select bit	Valid when bit 6 = “1” 0 : Odd parity 1 : Even parity	○	○
PRYE	Parity enable bit	0 : Parity disabled 1 : Parity enabled	○	○
SLEP	Sleep select bit	0 : Sleep mode deselected 1 : Sleep mode selected	○	○

Note : Set the corresponding port direction register to “0”.

UART2 transmit / receive mode register

b7	b6	b5	b4	b3	b2	b1	b0

Symbol: U2MR — Address: 0378$_{16}$ — When reset: 00$_{16}$

Bit symbol	Bit name	Function	R	W
SMD0	Serial I/O mode select bit	b2 b1 b0 1 0 0 : Transfer data 7 bits long 1 0 1 : Transfer data 8 bits long 1 1 0 : Transfer data 9 bits long	○	○
SMD1			○	○
SMD2			○	○
CKDIR	Internal / external clock select bit	Must always be “0”	○	○
STPS	Stop bit length select bit	0 : One stop bit 1 : Two stop bits	○	○
PRY	Odd / even parity select bit	Valid when bit 6 = “1” 0 : Odd parity 1 : Even parity	○	○
PRYE	Parity enable bit	0 : Parity disabled 1 : Parity enabled	○	○
IOPOL	TxD, RxD I/O polarity reverse bit (Note)	0 : No reverse 1 : Reverse	○	○

Note: Usually set to “0”.

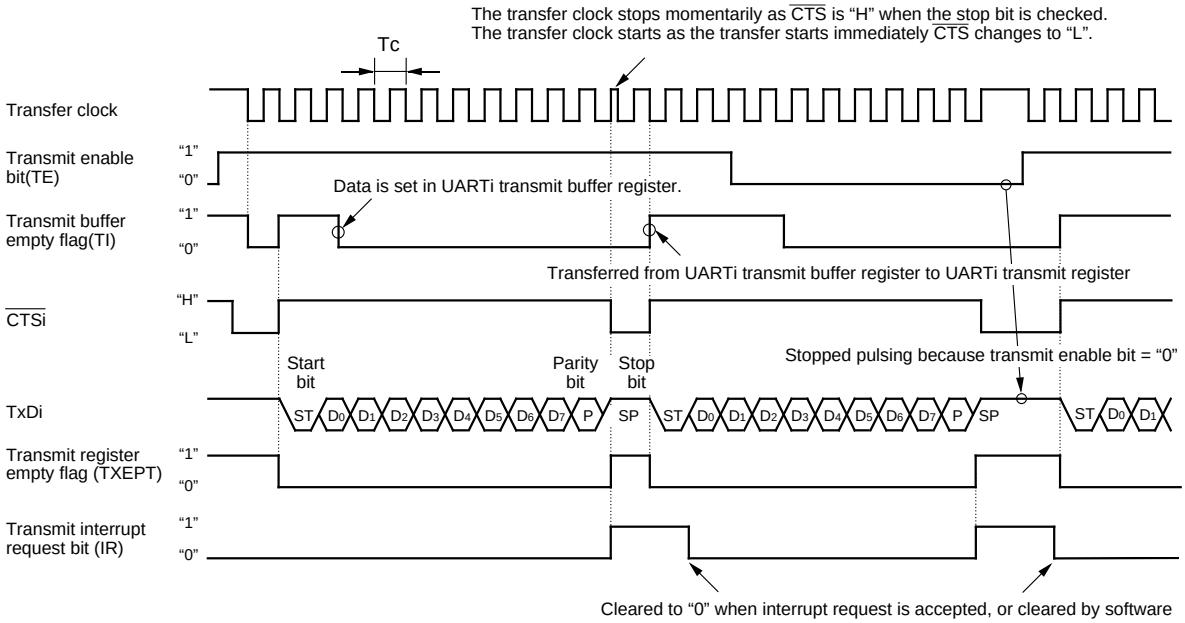
Figure 1.16.16. UARTi transmit/receive mode register in UART mode

Table 1.16.7 lists the functions of the input/output pins during UART mode. Note that for a period from when the UARTi operation mode is selected to when transfer starts, the TxDi pin outputs an “H”. (If the N-channel open-drain is selected, this pin is in floating state.)

Table 1.16.7. Input/output pin functions in UART mode

Pin name	Function	Method of selection
TxDi (P6$_3$, P6$_7$, P7$_0$)	Serial data output	
RxDi (P6$_2$, P6$_6$, P7$_1$)	Serial data input	Port P6$_2$, P6$_6$ and P7$_1$ direction register (bits 2 and 6 at address 03EE$_{16}$, bit 1 at address 03EF$_{16}$)= “0” (Can be used as an input port when performing transmission only)
CLKi (P6$_1$, P6$_5$, P7$_2$)	Programmable I/O port	Internal/external clock select bit (bit 3 at address 03A0$_{16}$, 03A8$_{16}$, 0378$_{16}$) = “0”
	Transfer clock input	Internal/external clock select bit (bit 3 at address 03A0$_{16}$, 03A8$_{16}$) = “1” Port P6$_1$, P6$_5$ direction register (bits 1 and 5 at address 03EE$_{16}$) = “0” (Do not set external clock for UART2)
$\overline{\text{CTS}}$i/$\overline{\text{RTS}}$i (P6$_0$, P6$_4$, P7$_3$)	$\overline{\text{CTS}}$ input	$\overline{\text{CTS}}$/$\overline{\text{RTS}}$ disable bit (bit 4 at address 03A4$_{16}$, 03AC$_{16}$, 037C$_{16}$) =“0” $\overline{\text{CTS}}$/$\overline{\text{RTS}}$ function select bit (bit 2 at address 03A4$_{16}$, 03AC$_{16}$, 037C$_{16}$) = “0” Port P6$_0$, P6$_4$ and P7$_3$ direction register (bits 0 and 4 at address 03EE$_{16}$, bit 3 at address 03EF$_{16}$) = “0”
	$\overline{\text{RTS}}$ output	$\overline{\text{CTS}}$/$\overline{\text{RTS}}$ disable bit (bit 4 at address 03A4$_{16}$, 03AC$_{16}$, 037C$_{16}$) = “0” $\overline{\text{CTS}}$/$\overline{\text{RTS}}$ function select bit (bit 2 at address 03A4$_{16}$, 03AC$_{16}$, 037C$_{16}$) = “1”
	Programmable I/O port	$\overline{\text{CTS}}$/$\overline{\text{RTS}}$ disable bit (bit 4 at address 03A4$_{16}$, 03AC$_{16}$, 037C$_{16}$) = “1”

• Example of transmit timing when transfer data is 8 bits long (parity enabled, one stop bit)

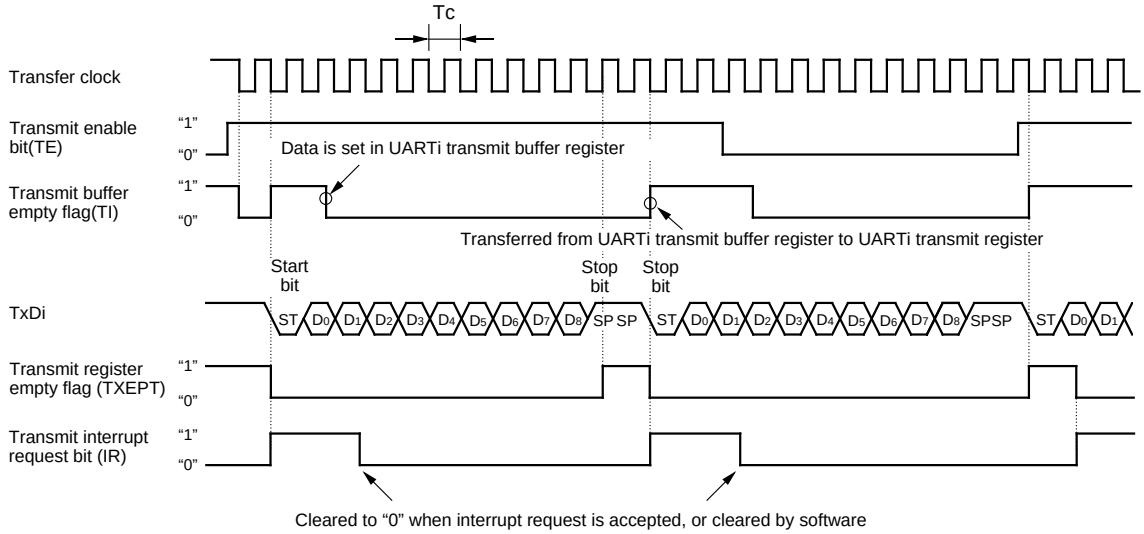


Shown in ( ) are bit symbols.

The above timing applies to the following settings :
- Parity is enabled.
- One stop bit.
- CTS function is selected.
- Transmit interrupt cause select bit = "1".

Tc = 16 (n + 1) / fi or 16 (n + 1) / fEXT
fi : frequency of BRGi count source (f1, f8, f32)
fEXT : frequency of BRGi count source (external clock)
n : value set to BRGi

• Example of transmit timing when transfer data is 9 bits long (parity disabled, two stop bits)



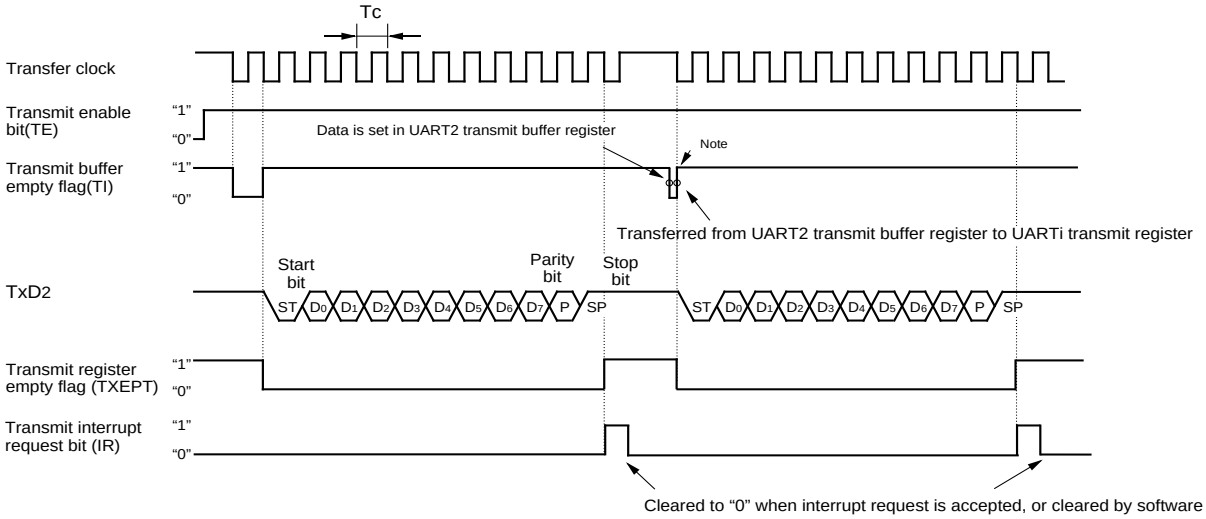
Shown in ( ) are bit symbols.

The above timing applies to the following settings :
- Parity is disabled.
- Two stop bits.
- CTS function is disabled.
- Transmit interrupt cause select bit = "0".

Tc = 16 (n + 1) / fi or 16 (n + 1) / fEXT
fi : frequency of BRGi count source (f1, f8, f32)
fEXT : frequency of BRGi count source (external clock)
n : value set to BRGi

Figure 1.16.17. Typical transmit timings in UART mode(UART0,UART1)

• Example of transmit timing when transfer data is 8 bits long (parity enabled, one stop bit)



Shown in ( ) are bit symbols.

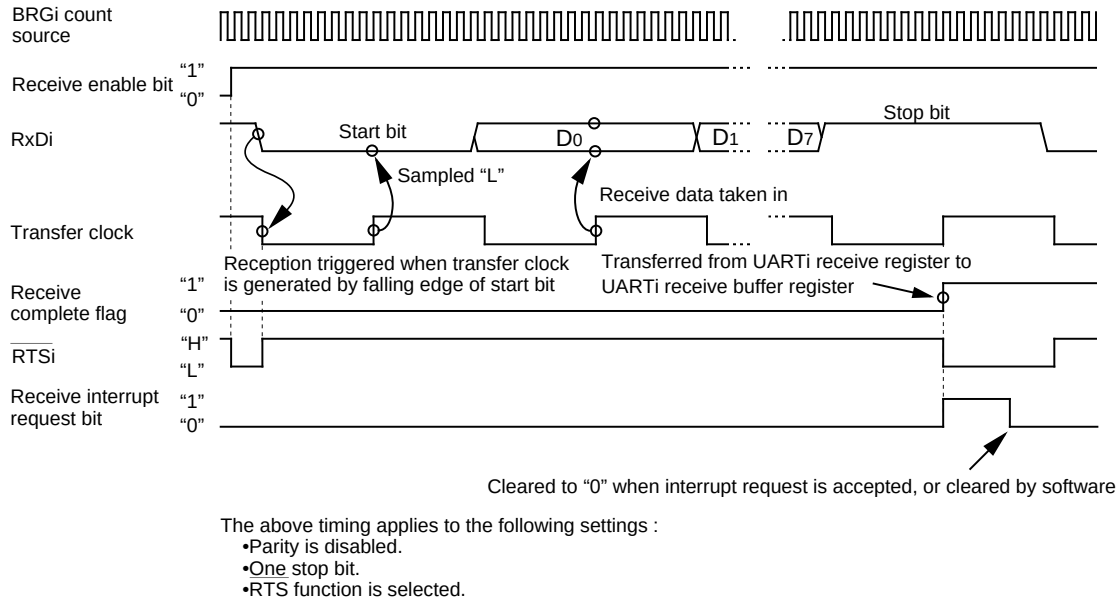
The above timing applies to the following settings :
- Parity is enabled.
- One stop bit.
- Transmit interrupt cause select bit = "1".

Tc = 16 (n + 1) / fi
fi : frequency of BRG2 count source (f_1, f_8, f_{32})
n : value set to BRG2

Note: The transmit is started with overflow timing of BRG after having written in a value at the transmit buffer in the above timing.

Figure 1.16.18. Typical transmit timings in UART mode(UART2)

• Example of receive timing when transfer data is 8 bits long (parity disabled, one stop bit)



The above timing applies to the following settings :
- Parity is disabled.
- One stop bit.
- RTS function is selected.

Figure 1.16.19. Typical receive timing in UART mode

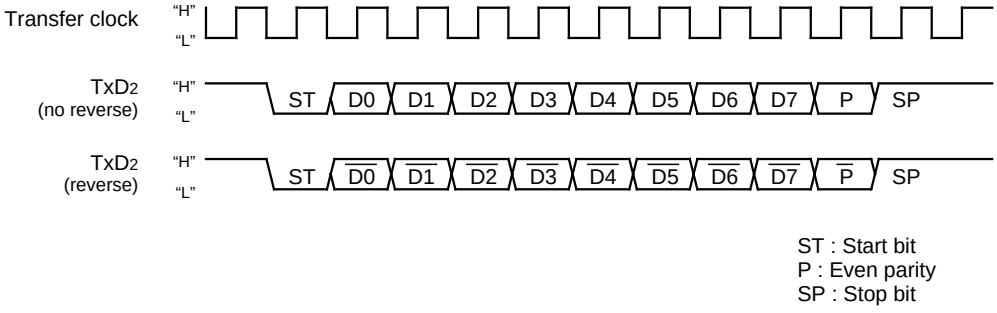
(a) Sleep mode (UART0, UART1)

This mode is used to transfer data between specific microcomputers among multiple microcomputers connected using UARTi. The sleep mode is selected when the sleep select bit (bit 7 at addresses $03A0_{16}$, $03A8_{16}$) is set to "1" during reception. In this mode, the unit performs receive operation when the MSB of the received data = "1" and does not perform receive operation when the MSB = "0".

(b) Function for switching serial data logic (UART2)

When the data logic select bit (bit 6 of address $037D_{16}$) is assigned 1, data is inverted in writing to the transmission buffer register or reading the reception buffer register. Figure 1.16.20 shows the example of timing for switching serial data logic.

• When LSB first, parity enabled, one stop bit



ST : Start bit
P : Even parity
SP : Stop bit

Figure 1.16.20. Timing for switching serial data logic

(c) TxD, RxD I/O polarity reverse function (UART2)

This function is to reverse TxD pin output and RxD pin input. The level of any data to be input or output (including the start bit, stop bit(s), and parity bit) is reversed. Set this function to “0” (not to reverse) for usual use.

(d) Bus collision detection function (UART2)

This function is to sample the output level of the TxD pin and the input level of the RxD pin at the rising edge of the transfer clock; if their values are different, then an interrupt request occurs. Figure 1.16.21 shows the example of detection timing of a bus collision (in UART mode).

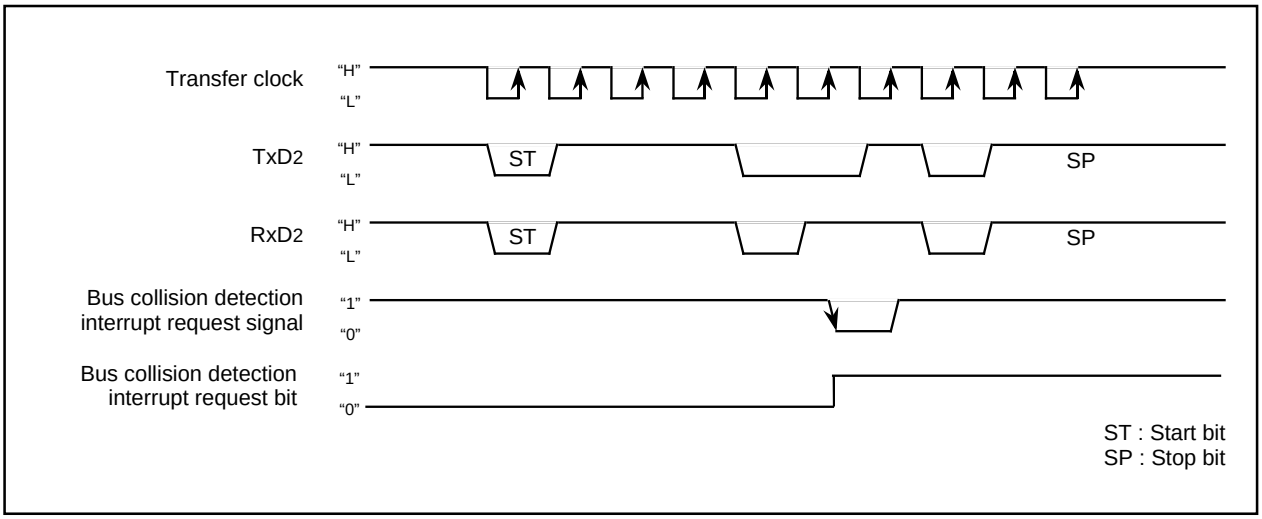


Figure 1.16.21. Detection timing of a bus collision (in UART mode)

(3) Clock-asynchronous serial I/O mode (used for the SIM interface)

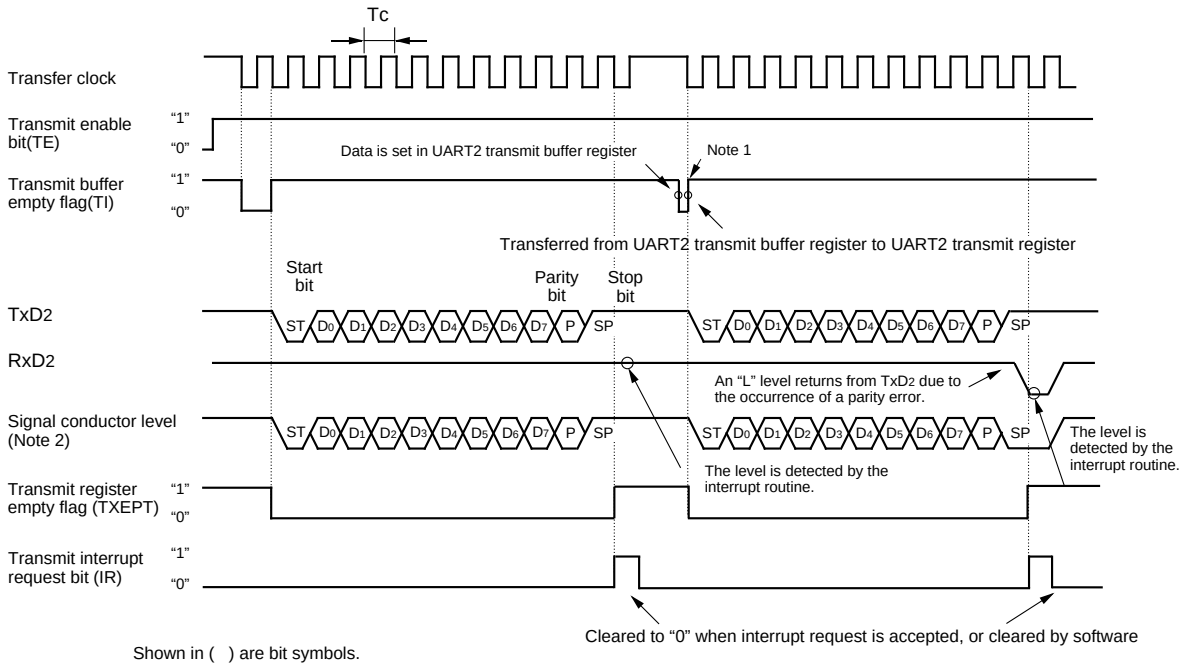
The SIM interface is used for connecting the microcomputer with a memory card or the like; adding some extra settings in UART2 clock-asynchronous serial I/O mode allows the user to effect this function. Table 1.16.8 shows the specifications of clock-asynchronous serial I/O mode (used for the SIM interface).

Table 1.16.8. Specifications of clock-asynchronous serial I/O mode (used for the SIM interface)

Item	Specification
Transfer data format	• Transfer data 8-bit UART mode (bit 2 through bit 0 of address 0378_{16} = "101_2") • One stop bit (bit 4 of address 0378_{16} = "0") • With the direct format chosen Set parity to "even" (bit 5 and bit 6 of address 0378_{16} = "1" and "1" respectively) Set data logic to "direct" (bit 6 of address $037D_{16}$ = "0"). Set transfer format to LSB (bit 7 of address $037C_{16}$ = "0"). • With the inverse format chosen Set parity to "odd" (bit 5 and bit 6 of address 0378_{16} = "0" and "1" respectively) Set data logic to "inverse" (bit 6 of address $037D_{16}$ = "1") Set transfer format to MSB (bit 7 of address $037C_{16}$ = "1")
Transfer clock	• With the internal clock chosen (bit 3 of address 0378_{16} = "0") : fi / 16 (n + 1) (Note 1) : fi=f_1, f_8, f_{32} (Do not set external clock)
Transmission / reception control	• Disable the $\overline{CTS}$ and $\overline{RTS}$ function (bit 4 of address $037C_{16}$ = "1")
Other settings	• The sleep mode select function is not available for UART2 • Set transmission interrupt factor to "transmission completed" (bit 4 of address $037D_{16}$ = "1")
Transmission start condition	• To start transmission, the following requirements must be met: - Transmit enable bit (bit 0 of address $037D_{16}$) = "1" - Transmit buffer empty flag (bit 1 of address $037D_{16}$) = "0"
Reception start condition	• To start reception, the following requirements must be met: - Reception enable bit (bit 2 of address $037D_{16}$) = "1" - Detection of a start bit
Interrupt request generation timing	• When transmitting When data transmission from the UART2 transmit register is completed (bit 4 of address $037D_{16}$ = "1") • When receiving When data transfer from the UART2 receive register to the UART2 receive buffer register is completed
Error detection	• Overrun error (see the specifications of clock-asynchronous serial I/O) (Note 2) • Framing error (see the specifications of clock-asynchronous serial I/O) • Parity error (see the specifications of clock-asynchronous serial I/O) - On the reception side, an "L" level is output from the TxD_2 pin by use of the parity error signal output function (bit 7 of address $037D_{16}$ = "1") when a parity error is detected - On the transmission side, a parity error is detected by the level of input to the RxD_2 pin when a transmission interrupt occurs • The error sum flag (see the specifications of clock-asynchronous serial I/O)

Note 1: 'n' denotes the value 00_{16} to FF_{16} that is set to the UART2 bit rate generator.

Note 2: If an overrun error occurs, the UART2 receive buffer will have the next data written in. Note also that the UART2 receive interrupt request bit does not change.

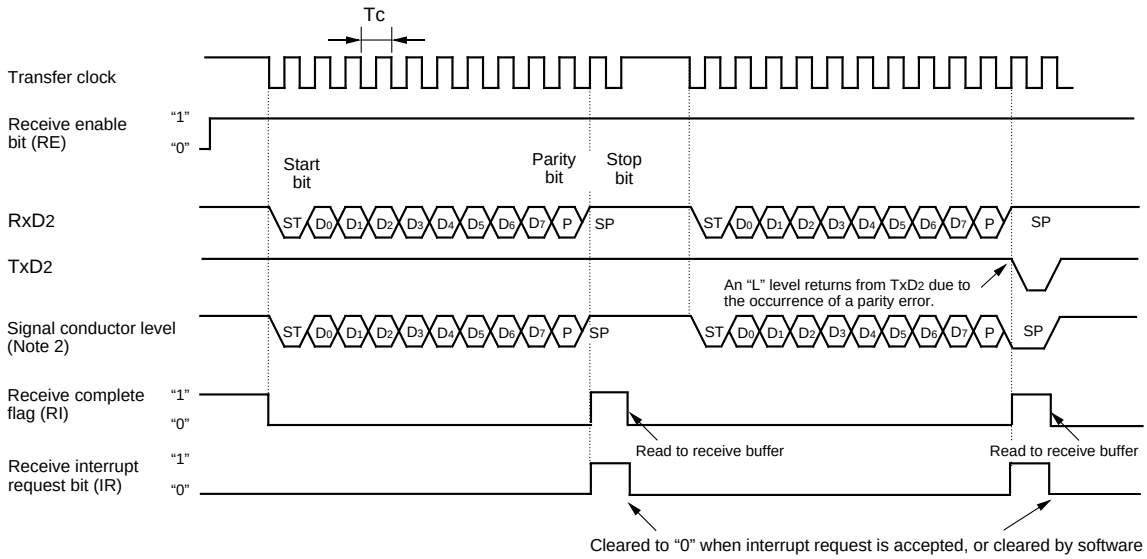


Shown in ( ) are bit symbols.

The above timing applies to the following settings :
- Parity is enabled.
- One stop bit.
- Transmit interrupt cause select bit = "1".

Tc = 16 (n + 1) / fi
fi : frequency of BRG2 count source (f1, f8, f32)
n : value set to BRG2

Note 1: The transmit is started with overflow timing of BRG after having written in a value at the transmit buffer in the above timing.



Shown in ( ) are bit symbols.

The above timing applies to the following settings :
- Parity is enabled.
- One stop bit.
- Transmit interrupt cause select bit = "0".

Tc = 16 (n + 1) / fi
fi : frequency of BRG2 count source (f1, f8, f32)
n : value set to BRG2

Note 2: Equal in waveform because TxD2 and RxD2 are connected.

Figure 1.16.22. Typical transmit/receive timing in UART mode (used for the SIM interface)

(a) Function for outputting a parity error signal

During reception, with the error signal output enable bit (bit 7 of address 037D16) assigned “1”, you can output an “L” level from the TxD2 pin when a parity error is detected. And during transmission, comparing with the case in which the error signal output enable bit (bit 7 of address 037D16) is assigned “0”, the transmission completion interrupt occurs in the half cycle later of the transfer clock. Therefore parity error signals can be detected by a transmission completion interrupt program. Figure 1.16.23 shows the output timing of the parity error signal.

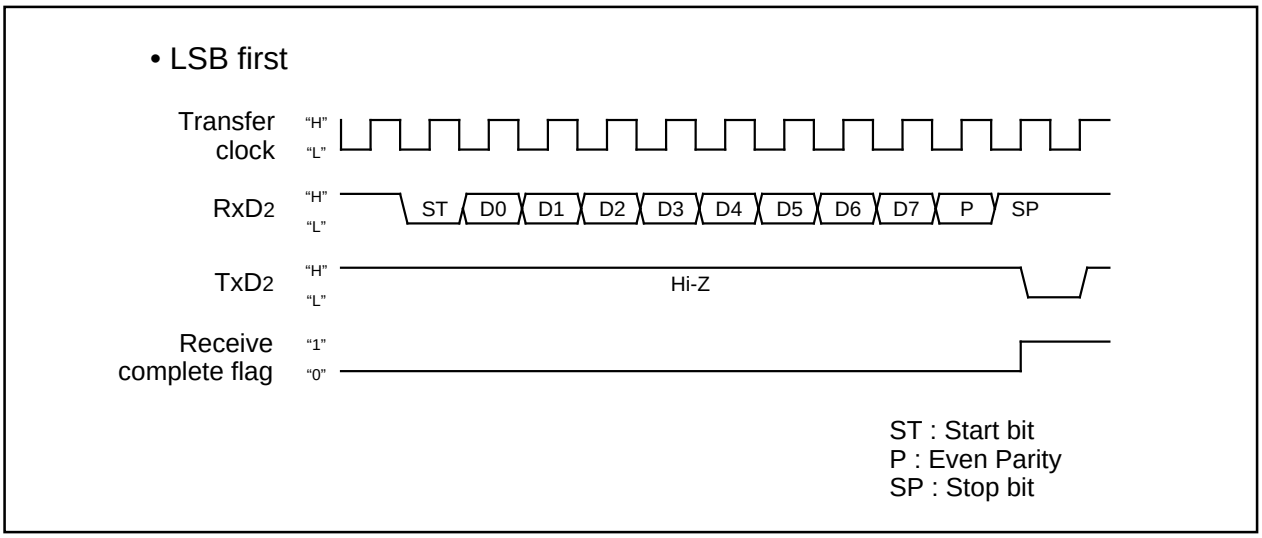


Figure 1.16.23. Output timing of the parity error signal

(b) Direct format/inverse format

Connecting the SIM card allows you to switch between direct format and inverse format. If you choose the direct format, D0 data is output from TxD2. If you choose the inverse format, D7 data is inverted and output from TxD2.

Figure 1.16.24 shows the SIM interface format.

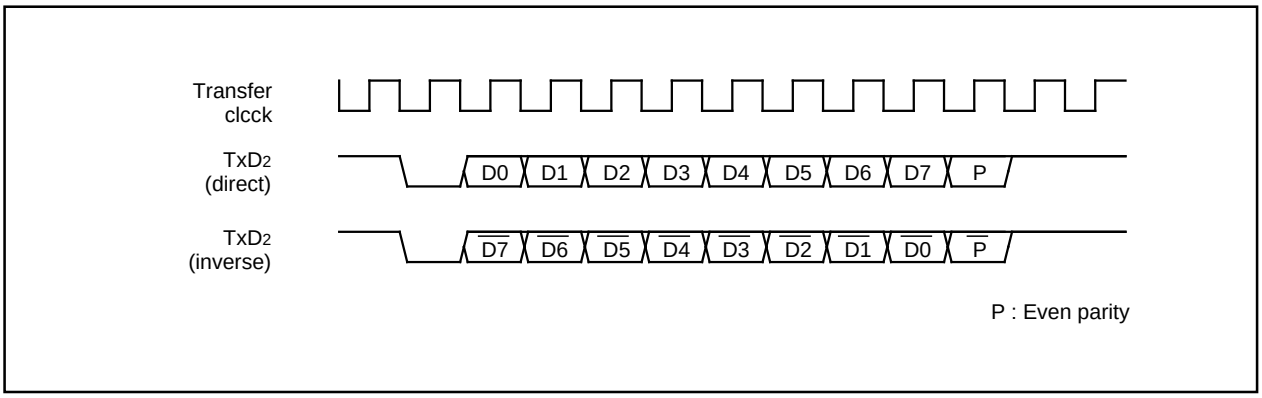


Figure 1.16.24. SIM interface format

Figure 1.16.25 shows the example of connecting the SIM interface. Connect TxD2 and RxD2 and apply pull-up.

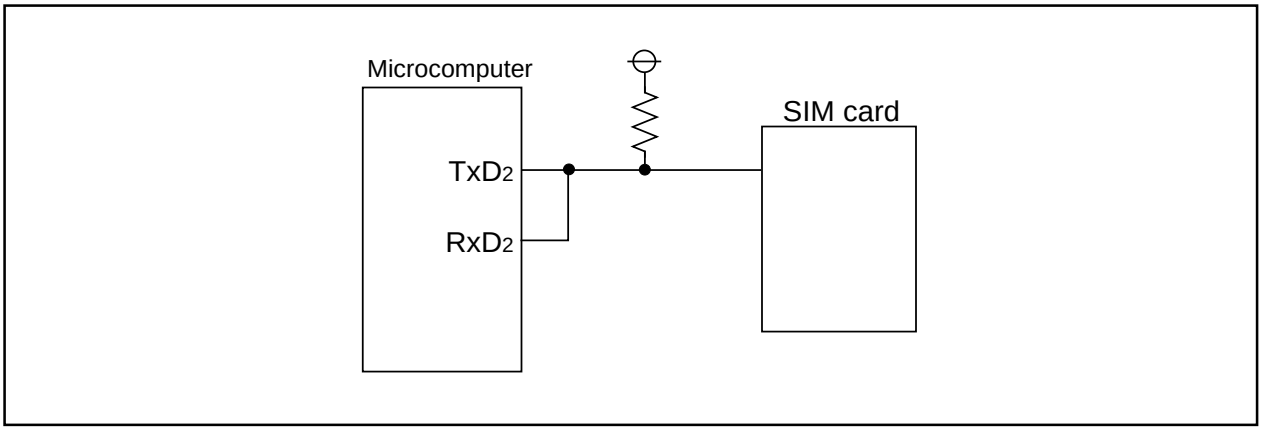


Figure 1.16.25. Connecting the SIM interface

UART2 Special Mode Register

The UART2 special mode register (address 037716) is used to control UART2 in various ways.

Figure 1.16.26 shows the UART2 special mode register.

Bit 0 of the UART2 special mode register (037716) is used as the I^2C mode select bit.

Setting “1” in the I^2C mode select bit (bit 0) goes the circuit to achieve the I^2C bus (simplified I^2C bus) interface effective.

Table 1.16.9 shows the relation between the I^2C mode select bit and respective control workings.

Since this function uses clock-synchronous serial I/O mode, set this bit to “0” in UART mode.

UART2 special mode register

b7 b6 b5 b4 b3 b2 b1 b0

Symbol: U2SMR — Address: 037716 — When reset: 0016

Bit symbol	Bit name	Function (During clock synchronous serial I/O mode)	Function (During UART mode)	R	W
IICM	I^2C mode select bit	0 : Normal mode 1 : I^2C mode	Must always be “0”	○	○
ABC	Arbitration lost detecting flag control bit	0 : Update per bit 1 : Update per byte	Must always be “0”	○	○
BBS	Bus busy flag	0 : STOP condition detected 1 : START condition detected	Must always be “0”	○	○ (Note1)
LSYN	SCL L sync output enable bit	0 : Disabled 1 : Enabled	Must always be “0”	○	○
ABSCS	Bus collision detect sampling clock select bit	Must always be “0”	0 : Rising edge of transfer clock 1 : Underflow signal of timer A0	○	○
ACSE	Auto clear function select bit of transmit enable bit	Must always be “0”	0 : No auto clear function 1 : Auto clear at occurrence of bus collision	○	○
SSS	Transmit start condition select bit	Must always be “0”	0 : Ordinary 1 : Falling edge of RxD_2	○	○
SDDS	SDA digital delay select bit (Note 2, Note 3)	0 : Analog delay output is selected 1 : Digital delay output is selected (must always be “0” when not using I^2C mode)	Must always be “0”	○	○

Note 1: Nothing but “0” may be written.
Note 2: When not in I^2C mode, do not set this bit by writing a “1”. During normal mode, fix it to “0”. When this bit = “0”, UART2 special mode register 3 (U2SMR3 at address 037516) bits 7 to 5 (DL2 to DL0 = SDA digital delay setup bits) are initialized to “000”, with the analog delay circuit selected. Also, when SDDS = “0”, the U2SMR3 register cannot be read or written to.
Note 3: When analog delay is selected, only the analog delay value is effective; when digital delay is selected, only the digital delay value is effective.

UART2 special mode register 3 (I^2C bus exclusive use register)

b7 b6 b5 b4 b3 b2 b1 b0

Symbol: U2SMR3 — Address: 037516 — When reset: Indeterminate (However, when SDDS = “1”, the initial value is “0016”)

Bit symbol	Bit name	Function (I^2C bus exclusive use register)	R	W
		Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be indeterminate. However, when SDDS = “1”, the value “0” is read out (Note 1)	—	—
DL0	SDA digital delay setup bit (Note 1, Note 2, Note 3, Note 4)	b7 b6 b5 0 0 0 : Analog delay is selected 0 0 1 : 1 to 2 cycle(s) of 1/f(X_{IN}) 0 1 0 : 2 to 3 cycles of 1/f(X_{IN}) 0 1 1 : 3 to 4 cycles of 1/f(X_{IN}) 1 0 0 : 4 to 5 cycles of 1/f(X_{IN}) 1 0 1 : 5 to 6 cycles of 1/f(X_{IN}) 1 1 0 : 6 to 7 cycles of 1/f(X_{IN}) 1 1 1 : 7 to 8 cycles of 1/f(X_{IN}) } Digital delay is selected	○	○
DL1			○	○
DL2			○	○

Note 1: This bit can be read or written to when UART2 special mode register (U2SMR at address 037716) bit 7 (SDDS: SDA digital delay select bit) = “1”. When the initial value of UART2 special mode register 3 (U2SMR3) is read after setting SDDS = “1”, the value is “0016”. When writing to UART2 special mode register 3 (U2SMR3) after setting SDDS = “1”, be sure to write 0's to bits 0–4. When SDDS = “0”, this register cannot be written to; when read, the value is indeterminate.
Note 2: These bits are initialized to “000” when SDDS = “0”, with the analog delay circuit selected. After a reset, these bits are set to “000”, with the analog delay circuit selected. However, because these bits can be read only when SDDS = “1”, the value read from these bits when SDDS = “0” is indeterminate.
Note 3: When analog delay is selected, only the analog delay value is effective; when digital delay is selected, only the digital delay value is effective.
Note 4: The amount of delay varies with the load on SCL and SDA pins. Also, when using an external clock, the amount of delay increases by about 100 ns, so be sure to take this into account when using the device.

Figure 1.16.26. UART2 special mode register

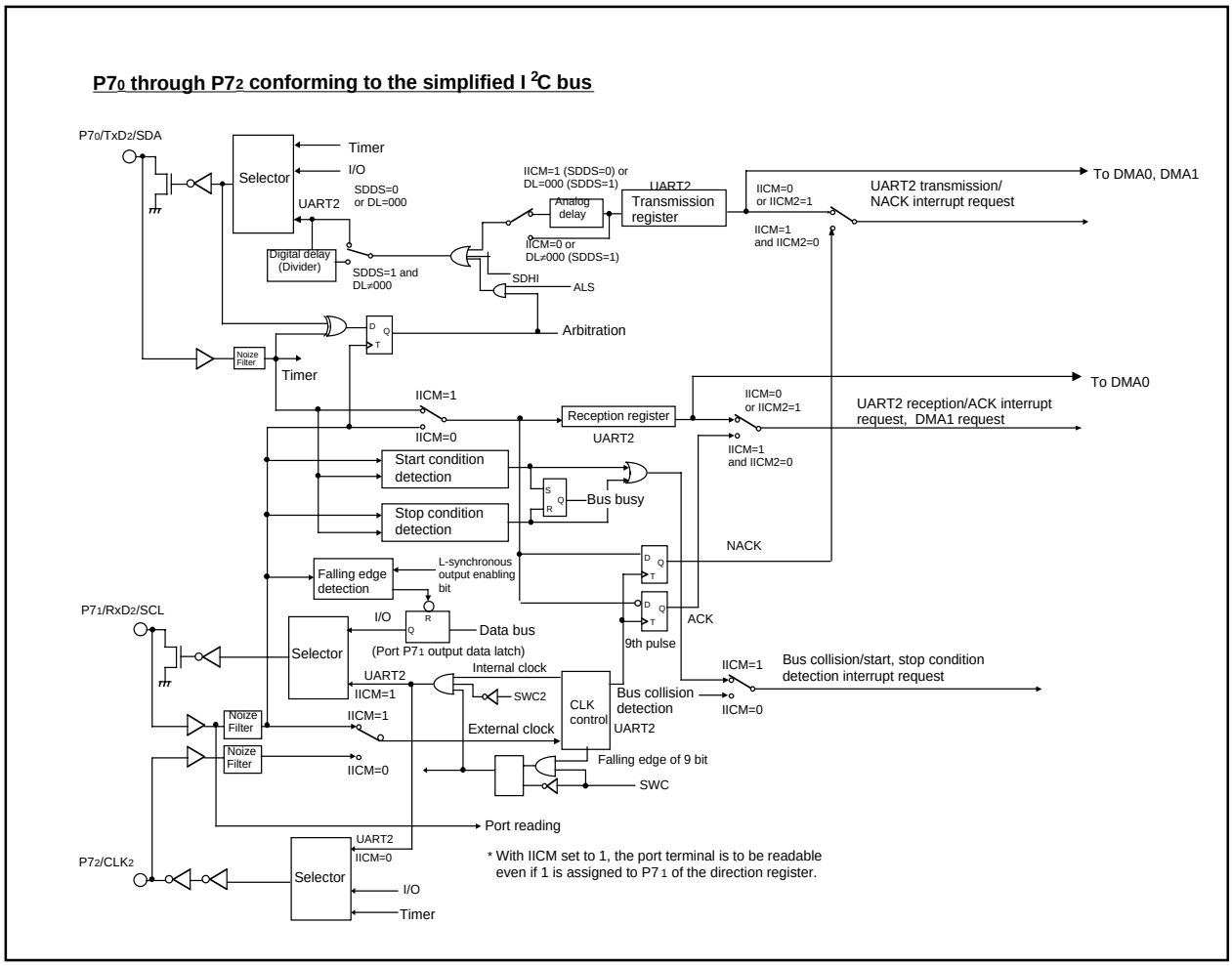


Figure 1.16.27. Functional block diagram for I²C mode

Table 1.16.9. Features in I²C mode

	Function	Normal mode	I²C mode (Note 1)
1	Factor of interrupt number 10 (Note 2)	Bus collision detection	Start condition detection or stop condition detection
2	Factor of interrupt number 15 (Note 2)	UART2 transmission	No acknowledgment detection (NACK)
3	Factor of interrupt number 16 (Note 2)	UART2 reception	Acknowledgment detection (ACK)
4	UART2 transmission output delay	Not delayed	Delayed
5	P7₀ at the time when UART2 is in use	TxD₂ (output)	SDA (input/output) (Note 3)
6	P7₁ at the time when UART2 is in use	RxD₂ (input)	SCL (input/output)
7	P7₂ at the time when UART2 is in use	CLK₂	P7₂
8	DMA1 factor at the time when 1 1 0 1 is assigned to the DMA request factor selection bits	UART2 reception	Acknowledgment detection (ACK)
9	Noise filter width	15ns	50ns
10	Reading P7₁	Reading the terminal when 0 is assigned to the direction register	Reading the terminal regardless of the value of the direction register
11	Initial value of UART2 output	H level (when 0 is assigned to the CLK polarity select bit)	The value set in latch P7₀ when the port is selected

Note 1: Make the settings given below when I²C mode is in use.
Set 0 1 0 in bits 2, 1, 0 of the UART2 transmission/reception mode register.
Disable the RTS/CTS function. Choose the MSB First function.
Note 2: Follow the steps given below to switch from a factor to another.
1. Disable the interrupt of the corresponding number.
2. Switch from a factor to another.
3. Reset the interrupt request flag of the corresponding number.
4. Set an interrupt level of the corresponding number.
Note 3: Set an initial value of SDA transmission output when serial I/O is invalid.

Figure 1.16.27 shows the functional block diagram for I^2C mode. Setting “1” in the I^2C mode select bit (IICM) causes ports $P7_0$, $P7_1$, and $P7_2$ to work as data transmission-reception terminal SDA, clock input-output terminal SCL, and port $P7_2$ respectively. A delay circuit is added to the SDA transmission output, so the SDA output changes after SCL fully goes to “L”. The SDA digital delay select bit (bit 7 at address 0377_{16}) can be used to select between analog delay and digital delay. When digital delay is selected, the amount of delay can be selected in the range of 2 cycles to 8 cycles of f1 using UART2 special mode register 3 (at address 0375_{16}). Delay circuit select conditions are shown in Table 1.16.10.

Table 1.16.10. Delay circuit select conditions

	Register value			Contents
	IICM	SDDS	DL	
Digital delay is selected	1	1	001 to 111	When digital delay is selected, no analog delay is added. Only digital delay is effective.
Analog delay is selected	1	1	000	When DL is set to “000”, analog delay is selected no matter what value is set in SDDS.
		0	(000)	When SDDS is set to “0”, DL is initialized, so that DL =“000”.
No delay	0	0	(000)	When IICM = “0”, no delay circuit is selected. When IICM = “0”, however, always make sure SDDS = “0”.

An attempt to read Port $P7_1$ (SCL) results in getting the terminal’s level regardless of the content of the port direction register. The initial value of SDA transmission output in this mode goes to the value set in port $P7_0$. The interrupt factors of the bus collision detection interrupt, UART2 transmission interrupt, and of UART2 reception interrupt turn to the start/stop condition detection interrupt, acknowledgment non-detection interrupt, and acknowledgment detection interrupt respectively.

The start condition detection interrupt refers to the interrupt that occurs when the falling edge of the SDA terminal ($P7_0$) is detected with the SCL terminal ($P7_1$) staying “H”. The stop condition detection interrupt refers to the interrupt that occurs when the rising edge of the SDA terminal ($P7_0$) is detected with the SCL terminal ($P7_1$) staying “H”. The bus busy flag (bit 2 of the UART2 special mode register) is set to “1” by the start condition detection, and set to “0” by the stop condition detection.

The acknowledgment non-detection interrupt refers to the interrupt that occurs when the SDA terminal level is detected still staying “H” at the rising edge of the 9th transmission clock. The acknowledgment detection interrupt refers to the interrupt that occurs when SDA terminal’s level is detected already went to “L” at the 9th transmission clock. Also, assigning 1 1 0 1 (UART2 reception) to the DMA1 request factor select bits provides the means to start up the DMA transfer by the effect of acknowledgment detection.

Bit 1 of the UART2 special mode register (0377_{16}) is used as the arbitration lost detecting flag control bit. Arbitration means the act of detecting the nonconformity between transmission data and SDA terminal data at the timing of the SCL rising edge. This detecting flag is located at bit 11 of the UART2 reception buffer register ($037F_{16}$, $037E_{16}$), and “1” is set in this flag when nonconformity is detected. Use the arbitration lost detecting flag control bit to choose which way to use to update the flag, bit by bit or byte by byte. When setting this bit to “1” and updated the flag byte by byte if nonconformity is detected, the arbitration lost detecting flag is set to “1” at the falling edge of the 9th transmission clock.

If update the flag byte by byte, must judge and clear (“0”) the arbitration lost detecting flag after completing the first byte acknowledge detect and before starting the next one byte transmission.

Bit 3 of the UART2 special mode register is used as SCL- and L-synchronous output enable bit. Setting this bit to “1” goes the $P7_1$ data register to “0” in synchronization with the SCL terminal level going to “L”.

Some other functions added are explained here. Figure 1.16.28 shows their workings.

Bit 4 of the UART2 special mode register is used as the bus collision detect sampling clock select bit. The bus collision detect interrupt occurs when the RxD2 level and TxD2 level do not match, but the nonconformity is detected in synchronization with the rising edge of the transfer clock signal if the bit is set to “0”. If this bit is set to “1”, the nonconformity is detected at the timing of the overflow of timer A0 rather than at the rising edge of the transfer clock.

Bit 5 of the UART2 special mode register is used as the auto clear function select bit of transmit enable bit. Setting this bit to “1” automatically resets the transmit enable bit to “0” when “1” is set in the bus collision detect interrupt request bit (nonconformity).

Bit 6 of the UART2 special mode register is used as the transmit start condition select bit. Setting this bit to “1” starts the TxD transmission in synchronization with the falling edge of the RxD terminal.

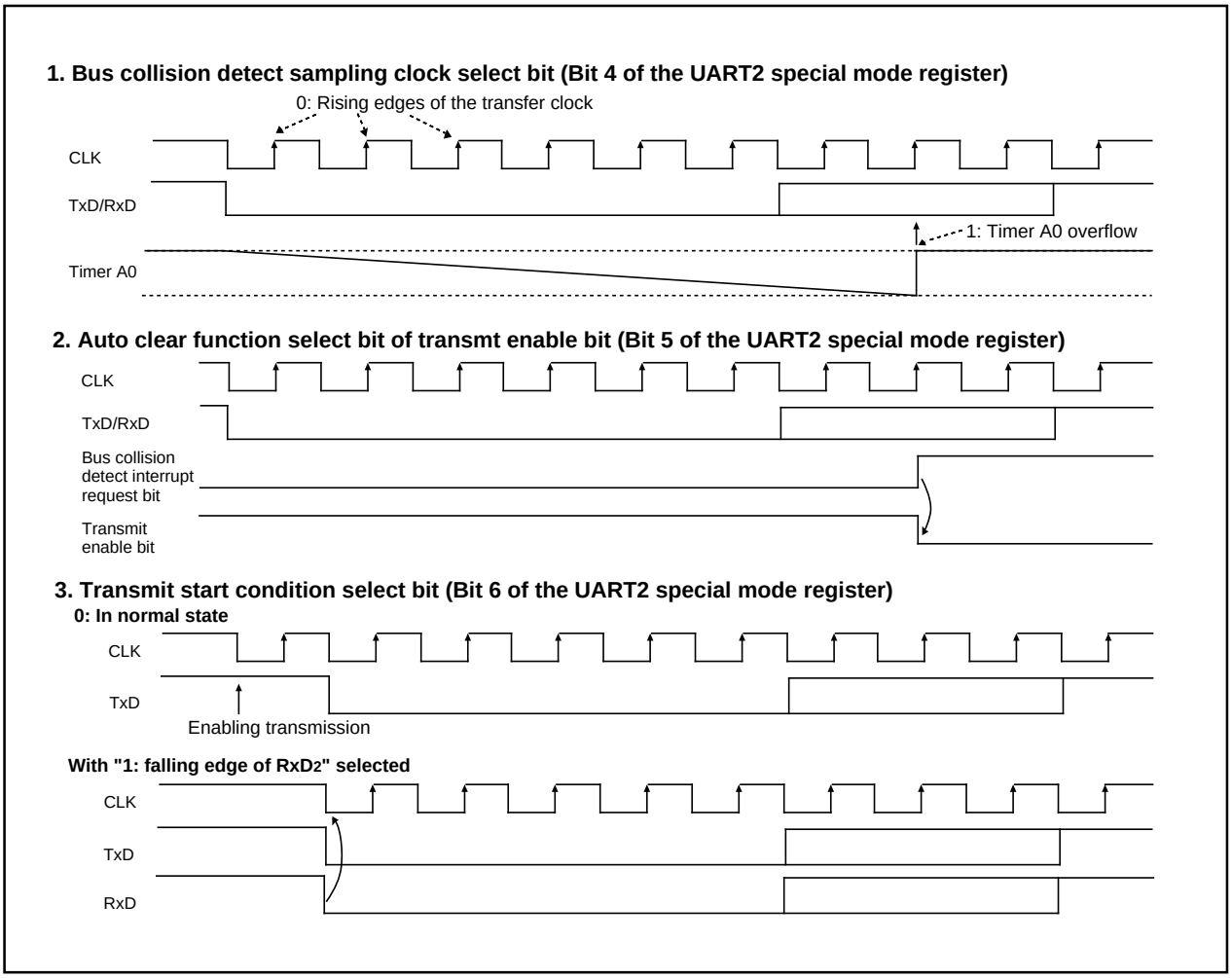


Figure 1.16.28. Some other functions added

UART2 Special Mode Register 2

UART2 special mode register 2 (address 037616) is used to further control UART2 in I2C mode. Figure 1.16.29 shows the UART2 special mode register 2.

UART2 special mode register 2 (I2C bus exclusive use register)

b7	b6	b5	b4	b3	b2	b1	b0

Symbol: U2SMR2 — Address: 037616 — When reset: 0016

Bit symbol	Bit name	Function	R	W
IICM2	I2C mode select bit 2	Refer to Table 1.16.11	○	○
CSC	Clock-synchronous bit	0 : Disabled 1 : Enabled	○	○
SWC	SCL wait output bit	0 : Disabled 1 : Enabled	○	○
ALS	SDA output stop bit	0 : Disabled 1 : Enabled	○	○
STAC	UART2 initialization bit	0 : Disabled 1 : Enabled	○	○
SWC2	SCL wait output bit 2	0: UART2 clock 1: 0 output	○	○
SDHI	SDA output disable bit	0: Enabled 1: Disabled (high impedance)	○	○
SHTC	Start/stop condition control bit	Set this bit to “1” in I2C mode (refer to Table 1.16.12)	○	○

Figure 1.16.29. UART2 special mode register 2

Bit 0 of the UART2 special mode register 2 (address 0376_{16}) is used as the I^2C mode select bit 2. Table 1.16.11 shows the types of control to be changed by I^2C mode select bit 2 when the I^2C mode select bit is set to "1". Table 1.16.12 shows the timing characteristics of detecting the start condition and the stop condition. Set the start/stop condition control bit (bit 7 of UART2 special mode register 2) to "1" in I^2C mode.

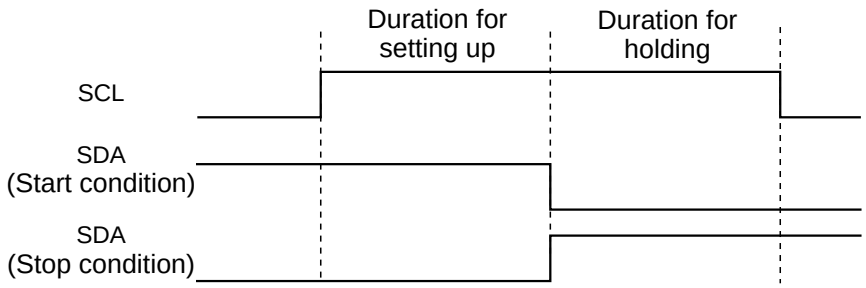
Table 1.16.11. Functions changed by I^2C mode select bit 2

	Function	IICM2 = 0	IICM2 = 1
1	Factor of interrupt number 15	No acknowledgment detection (NACK)	UART2 transmission (the rising edge of the final bit of the clock)
2	Factor of interrupt number 16	Acknowledgment detection (ACK)	UART2 reception (the falling edge of the final bit of the clock)
3	DMA1 factor at the time when 1 1 0 1 is assigned to the DMA request factor selection bits	Acknowledgment detection (ACK)	UART2 reception (the falling edge of the final bit of the clock)
4	Timing for transferring data from the UART2 reception shift register to the reception buffer.	The rising edge of the final bit of the reception clock	The falling edge of the final bit of the reception clock
5	Timing for generating a UART2 reception/ACK interrupt request	The rising edge of the final bit of the reception clock	The falling edge of the final bit of the reception clock

Table 1.16.12. Timing characteristics of detecting the start condition and the stop condition (Note 1)

3 to 6 cycles < duration for setting-up (Note 2)
3 to 6 cycles < duration for holding (Note 2)

Note 1 : When the start/stop condition control bit SHTC is "1" .
Note 2 : "Cycles" is in terms of the input oscillation frequency f(X_{IN}) of the main clock.



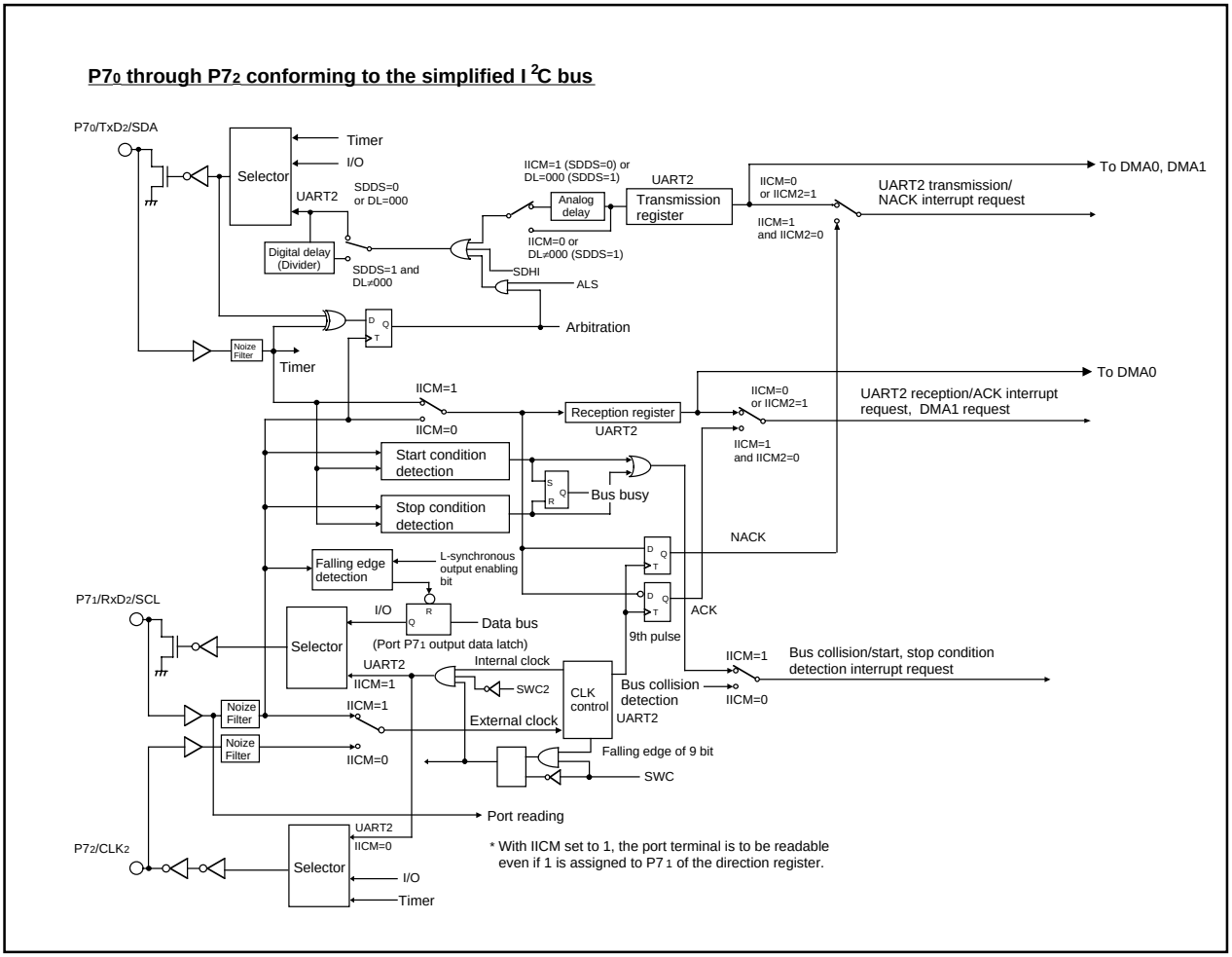


Figure 1.16.30. Functional block diagram for I²C mode

Functions available in I²C mode are shown in Figure 1.16.30 — a functional block diagram.

Bit 3 of the UART2 special mode register 2 (address 0376₁₆) is used as the SDA output stop bit. Setting this bit to “1” causes an arbitration loss to occur, and the SDA pin turns to high-impedance state at the instant when the arbitration lost detecting flag is set to “1”.

Bit 1 of the UART2 special mode register 2 (address 0376₁₆) is used as the clock synchronization bit. With this bit set to “1” at the time when the internal SCL is set to “H”, the internal SCL turns to “L” if the falling edge is found in the SCL pin; and the baud rate generator reloads the set value, and start counting within the “L” interval. When the internal SCL changes from “L” to “H” with the SCL pin set to “L”, stops counting the baud rate generator, and starts counting it again when the SCL pin turns to “H”. Due to this function, the UART2 transmission-reception clock becomes the logical product of the signal flowing through the internal SCL and that flowing through the SCL pin. This function operates over the period from the moment earlier by a half cycle than falling edge of the UART2 first clock to the rising edge of the ninth bit. To use this function, choose the internal clock for the transfer clock.

Bit 2 of the UART2 special mode register 2 (0376₁₆) is used as the SCL wait output bit. Setting this bit to “1” causes the SCL pin to be fixed to “L” at the falling edge of the ninth bit of the clock. Setting this bit to “0” frees the output fixed to “L”.

UART2 Special Mode Register 2

Bit 4 of the UART2 special mode register 2 (address 0376_{16}) is used as the UART2 initialization bit. Setting this bit to “1”, and when the start condition is detected, the microcomputer operates as follows.

(1) The transmission shift register is initialized, and the content of the transmission register is transferred to the transmission shift register. This starts transmission by dealing with the clock entered next as the first bit. The UART2 output value, however, doesn’t change until the first bit data is output after the entrance of the clock, and remains unchanged from the value at the moment when the microcomputer detected the start condition.

(2) The reception shift register is initialized, and the microcomputer starts reception by dealing with the clock entered next as the first bit.

(3) The SCL wait output bit turns to “1”. This turns the SCL pin to “L” at the falling edge of the ninth bit of the clock.

Starting to transmit/receive signals to/from UART2 using this function doesn’t change the value of the transmission buffer empty flag. To use this function, choose the external clock for the transfer clock.

Bit 5 of the UART2 special mode register 2 (0376_{16}) is used as the SCL pin wait output bit 2. Setting this bit to “1” with the serial I/O specified allows the user to forcibly output an “1” from the SCL pin even if UART2 is in operation. Setting this bit to “0” frees the “L” output from the SCL pin, and the UART2 clock is input/output.

Bit 6 of the UART2 special mode register 2 (0376_{16}) is used as the SDA output disable bit. Setting this bit to “1” forces the SDA pin to turn to the high-impedance state. Refrain from changing the value of this bit at the rising edge of the UART2 transfer clock. There can be instances in which arbitration lost detecting flag is turned on.

S I/O3, 4

S I/O3 and S I/O4 are exclusive clock-synchronous serial I/Os.

Figure 1.16.31 shows the S I/O3, 4 block diagram, and Figure 1.16.32 shows the S I/O3, 4 related register.

Table 1.16.13 shows the specifications of S I/O3, 4.

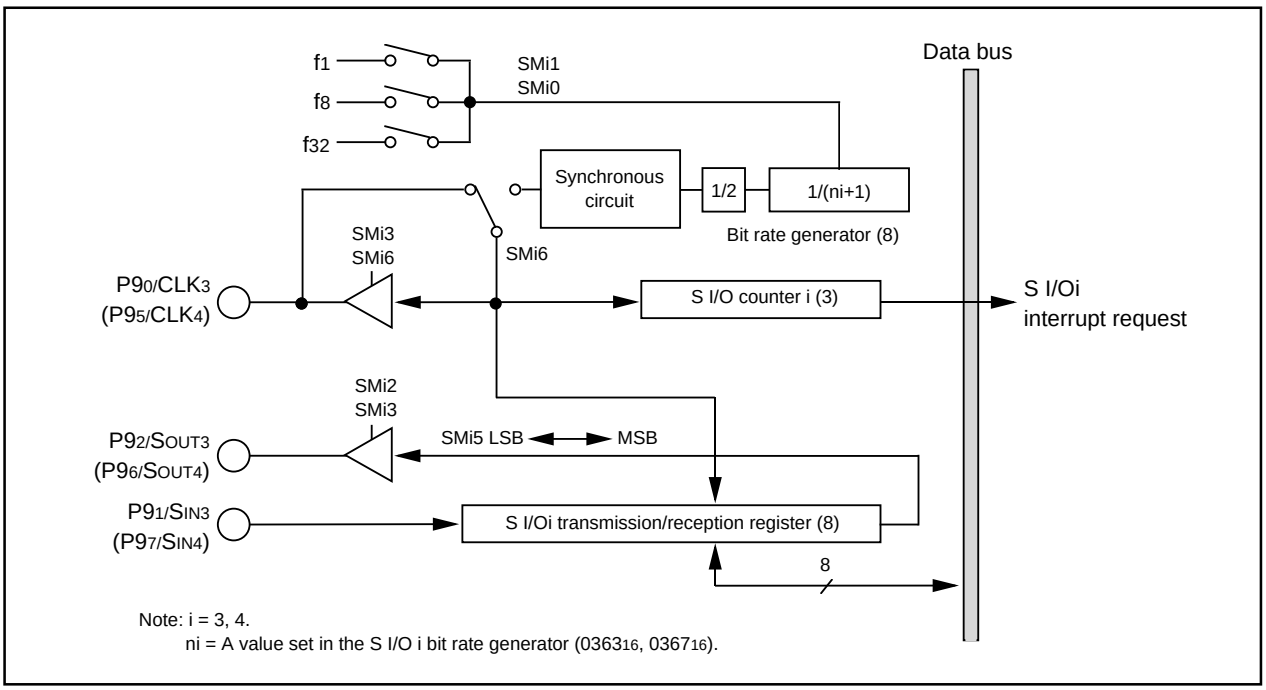
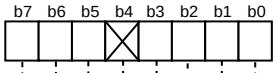


Figure 1.16.31. S I/O3, 4 block diagram

S I/Oi control register (i = 3, 4) (Note 1)

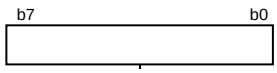


Symbol	Address	When reset
SiC	0362₁₆, 0366₁₆	40₁₆

Bit symbol	Bit name	Description	R	W
SMi0	Internal synchronous clock select bit	b1 b0 0 0 : Selecting f_1 0 1 : Selecting f_8 1 0 : Selecting f_{32} 1 1 : Must not be set.	○	○
SMi1			○	○
SMi2	SOUTi output disable bit	0 : SOUTi output 1 : SOUTi output disable(high impedance)	○	○
SMi3	S I/Oi port select bit (Note 2)	0 : Input-output port 1 : SOUTi output, CLK function	○	○
Nothing is assigned. In an attempt to write to this bit, write "0". The value, if read, turns out to be "0".			—	—
SMi5	Transfer direction select bit	0 : LSB first 1 : MSB first	○	○
SMi6	Synchronous clock select bit (Note 2)	0 : External clock 1 : Internal clock	○	○
SMi7	SOUTi initial value set bit	Effective when SMi3 = 0 0 : L output 1 : H output	○	○

Note 1: Set "1" in bit 2 of the protection register (000A₁₆) in advance to write to the S I/Oi control register (i = 3, 4).
Note 2: When using the port as an input/output port by setting the SI/Oi port select bit (i = 3, 4) to "0", be sure to set the sync clock select bit to "1".

SI/Oi bit rate generator (Note 1, 2)

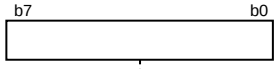


Symbol	Address	When reset
S3BRG	0363₁₆	Indeterminate
S4BRG	0367₁₆	Indeterminate

Indeterminate	Values that can be set	R	W
Assuming that set value = n, BRGi divides the count source by n + 1	00₁₆ to FF₁₆		○

Note 1: Write a value to this register while transmit/receive halts.
Note 2: Use MOV instruction to write to this register.

SI/Oi transmit/receive register (Note)



Symbol	Address	When reset
S3TRR	0360₁₆	Indeterminate
S4TRR	0364₁₆	Indeterminate

Indeterminate	R	W
Transmission/reception starts by writing data to this register. After transmission/reception finishes, reception data is input.	○	○

Note: Write a value to this register while transmit/receive halts.

Figure 1.16.32. S I/O3, 4 related register

Table 1.16.13. Specifications of S I/O3, 4

Item	Specifications
Transfer data format	• Transfer data length: 8 bits
Transfer clock	• With the internal clock selected (bit 6 of 0362_{16}, 0366_{16} = "1"): $f_1/2(ni+1)$, $f_8/2(ni+1)$, $f_{32}/2(ni+1)$ (Note 1) • With the external clock selected (bit 6 of 0362_{16}, 0366_{16} = 0):Input from the CLKi terminal (Note 2)
Conditions for transmission/ reception start	• To start transmit/reception, the following requirements must be met: - Select the synchronous clock (use bit 6 of 0362_{16}, 0366_{16}). Select a frequency dividing ratio if the internal clock has been selected (use bits 0 and 1 of 0362_{16}, 0366_{16}). - SOUTi initial value set bit (use bit 7 of 0362_{16}, 0366_{16})= 1. - S I/Oi port select bit (bit 3 of 0362_{16}, 0366_{16}) = 1. - Select the transfer direction (use bit 5 of 0362_{16}, 0366_{16}) -Write transfer data to SI/Oi transmit/receive register (0360_{16}, 0364_{16}) • To use S I/Oi interrupts, the following requirements must be met: - Clear the SI/Oi interrupt request bit before writing transfer data to the SI/Oi transmit/receive register (bit 3 of 0049_{16}, 0048_{16}) = 0.
Interrupt request generation timing	• Rising edge of the last transfer clock. (Note 3)
Select function	• LSB first or MSB first selection Whether transmission/reception begins with bit 0 (LSB) or bit 7 (MSB) can be selected. • Function for setting an SOUTi initial value selection When using an external clock for the transfer clock, the user can choose the SOUTi pin output level during a non-transfer time. For details on how to set, see Figure 1.16.33.
Precaution	• Unlike UART0–2, SI/Oi (i = 3, 4) is not divided for transfer register and buffer. Therefore, do not write the next transfer data to the SI/Oi transmit/receive register (addresses 0360_{16}, 0364_{16}) during a transfer. • When the internal clock is selected for the transfer clock, SOUTi holds the last data for a 1/2 transfer clock period after it finished transferring and then goes to a high-impedance state. However, if the transfer data is written to the SI/Oi transmit/ receive register (addresses 0360_{16}, 0364_{16}) during this time, SOUTi is placed in the high-impedance state immediately upon writing and the data hold time is thereby reduced.

Note 1: n is a value from 00_{16} through FF_{16} set in the S I/Oi bit rate generator (i = 3, 4).

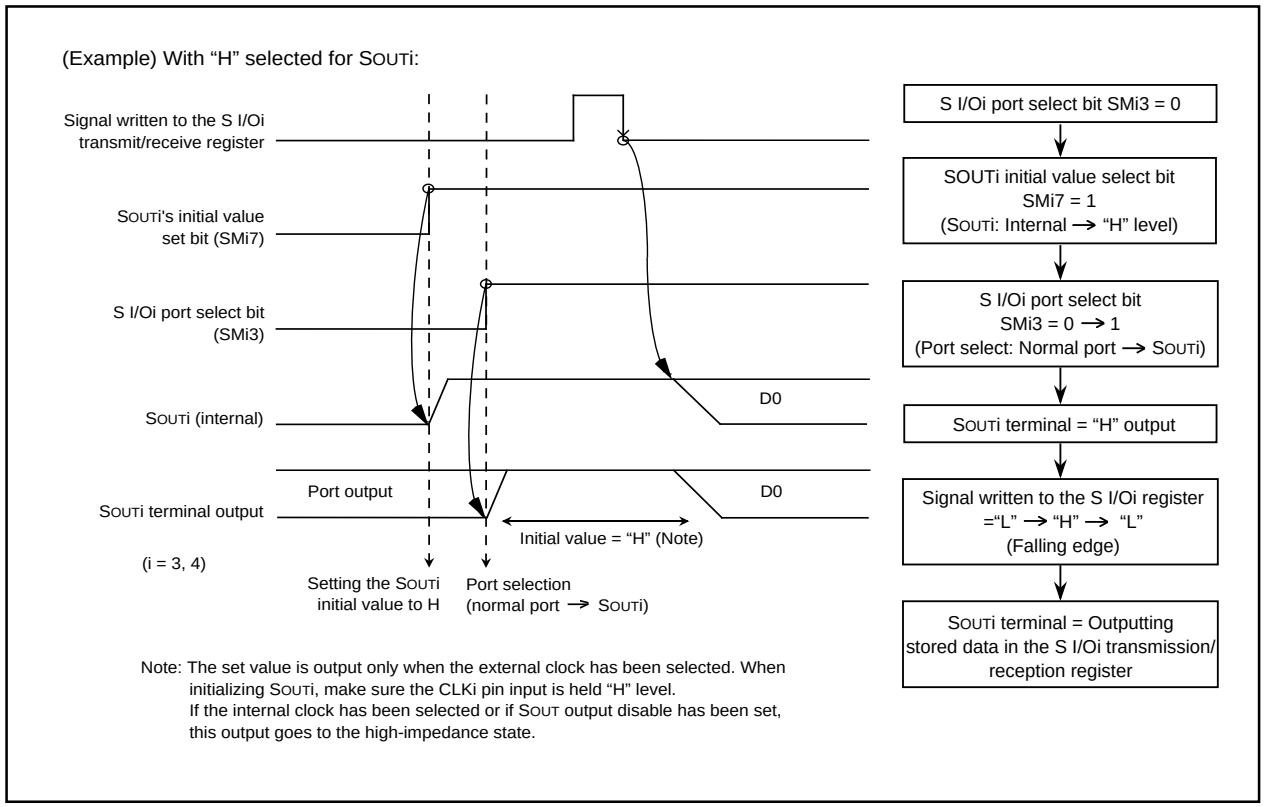
Note 2: With the external clock selected:

- Before data can be written to the SI/Oi transmit/receive register (addresses 0360_{16}, 0364_{16}), the CLKi pin input must be in the high state. Also, before rewriting the SI/Oi Control Register (addresses 0362_{16}, 0366_{16})'s bit 7 (SOUTi initial value set bit), make sure the CLKi pin input is held high.
- The S I/Oi circuit keeps on with the shift operation as long as the synchronous clock is entered in it, so stop the synchronous clock at the instant when it counts to eight. The internal clock, if selected, automatically stops.

Note 3: If the internal clock is used for the synchronous clock, the transfer clock signal stops at the "H" state.

■ Functions for setting an SOUTi initial value

When using an external clock for the transfer clock, the SOUTi pin output level during a non-transfer time can be set to the high or the low state. Figure 1.16.33 shows the timing chart for setting an SOUTi initial value and how to set it.

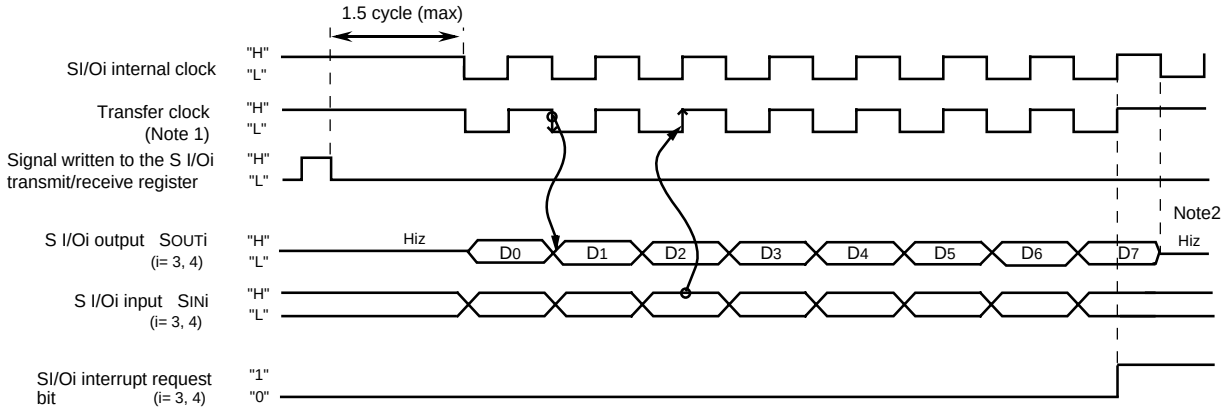


Note: The set value is output only when the external clock has been selected. When initializing SOUTi, make sure the CLKi pin input is held "H" level.
If the internal clock has been selected or if SOUT output disable has been set, this output goes to the high-impedance state.

Figure 1.16.33. Timing chart for setting SOUTi's initial value and how to set it

■ S I/Oi operation timing

Figure 1.16.34 shows the S I/Oi operation timing



Note 1: With the internal clock selected for the transfer clock, the frequency dividing ratio can be selected using bits 0 and 1 of the S I/Oi control register. (i=3,4) (No frequency division, 8-division frequency, 32-division frequency.)
Note 2: With the internal clock selected for the transfer clock, the SOUTi (i = 3, 4) pin becomes to the high-impedance state after the transfer finishes.
Note 3: Shown above is the case where the SOUTi (i = 3, 4) port select bit ="1".

Figure 1.16.34. S I/Oi operation timing chart

A-D Converter

The A-D converter consists of one 10-bit successive approximation A-D converter circuit with a capacitive coupling amplifier. Pins $P10_0$ to $P10_7$, $P9_5$, and $P9_6$ also function as the analog signal input pins. The direction registers of these pins for A-D conversion must therefore be set to input. The Vref connect bit (bit 5 at address $03D7_{16}$) can be used to isolate the resistance ladder of the A-D converter from the reference voltage input pin (V_{REF}) when the A-D converter is not used. Doing so stops any current flowing into the resistance ladder from V_{REF}, reducing the power dissipation. When using the A-D converter, start A-D conversion only after setting bit 5 of $03D7_{16}$ to connect V_{REF}. The result of A-D conversion is stored in the A-D registers of the selected pins. When set to 10-bit precision, the low 8 bits are stored in the even addresses and the high 2 bits in the odd addresses. When set to 8-bit precision, the low 8 bits are stored in the even addresses.

Table 1.17.1 shows the performance of the A-D converter. Figure 1.17.1 shows the block diagram of the A-D converter, and Figures 1.17.2 and 1.17.3 show the A-D converter-related registers.

Table 1.17.1. Performance of A-D converter

Item	Performance
Method of A-D conversion	Successive approximation (capacitive coupling amplifier)
Analog input voltage (Note 1)	0V to AV_{CC} (V_{CC})
Operating clock ϕ_{AD} (Note 2)	V_{CC} = 5V f_{AD}/divide-by-2 of f_{AD}/divide-by-4 of f_{AD}, $f_{AD}=f(X_{IN})$ V_{CC} = 3V divide-by-2 of f_{AD}/divide-by-4 of f_{AD}, $f_{AD}=f(X_{IN})$
Resolution	8-bit or 10-bit (selectable)
Absolute precision	V_{CC} = 5V • Without sample and hold function ±3LSB • With sample and hold function (8-bit resolution) ±2LSB • With sample and hold function (10-bit resolution) AN_0 to AN_7 input : ±3LSB ANEX0 and ANEX1 input (including mode in which external operation amp is connected) : ±7LSB V_{CC} = 3V • Without sample and hold function (8-bit resolution) ±2LSB
Operating modes	One-shot mode, repeat mode, single sweep mode, repeat sweep mode 0, and repeat sweep mode 1
Analog input pins	8pins (AN_0 to AN_7) + 2pins (ANEX0 and ANEX1)
A-D conversion start condition	• Software trigger A-D conversion starts when the A-D conversion start flag changes to “1” • External trigger (can be retriggered) A-D conversion starts when the A-D conversion start flag is “1” and the $\overline{ADTRG}$/$P9_7$ input changes from “H” to “L”
Conversion speed per pin	• Without sample and hold function 8-bit resolution: 49 ϕ_{AD} cycles, 10-bit resolution: 59 ϕ_{AD} cycles • With sample and hold function 8-bit resolution: 28 ϕ_{AD} cycles, 10-bit resolution: 33 ϕ_{AD} cycles

Note 1: Does not depend on use of sample and hold function.

Note 2: Divide the frequency if $f(X_{IN})$ exceeds 10MHz, and make ϕ_{AD} frequency equal to or less than 10MHz.
Without sample and hold function, set the ϕ_{AD} frequency to 250kHz min.
With the sample and hold function, set the ϕ_{AD} frequency to 1MHz min.

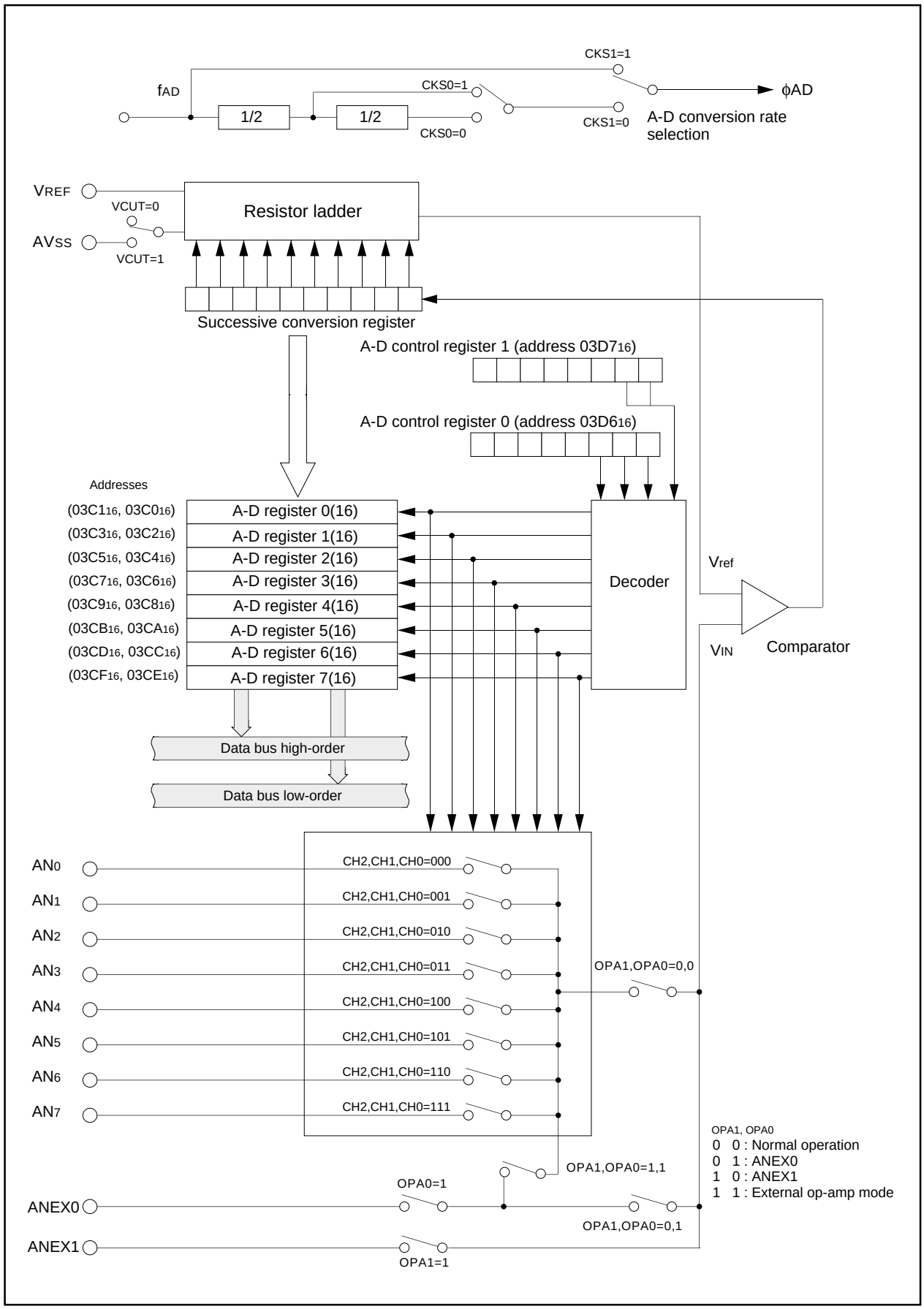
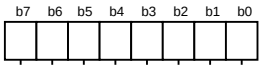


Figure 1.17.1. Block diagram of A-D converter

A-D control register 0 (Note 1)

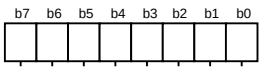


Symbol	Address	When reset
ADCON0	$03D6_{16}$	$00000XXX_2$

Bit symbol	Bit name	Function	R	W
CH0	Analog input pin select bit	b2 b1 b0 0 0 0 : AN_0 is selected 0 0 1 : AN_1 is selected 0 1 0 : AN_2 is selected 0 1 1 : AN_3 is selected 1 0 0 : AN_4 is selected 1 0 1 : AN_5 is selected 1 1 0 : AN_6 is selected 1 1 1 : AN_7 is selected (Note 2)	○	○
CH1				
CH2				
MD0	A-D operation mode select bit 0	b4 b3 0 0 : One-shot mode 0 1 : Repeat mode 1 0 : Single sweep mode 1 1 : Repeat sweep mode 0 Repeat sweep mode 1 (Note 2)	○	○
MD1				
TRG	Trigger select bit	0 : Software trigger 1 : AD_{TRG} trigger	○	○
ADST	A-D conversion start flag	0 : A-D conversion disabled 1 : A-D conversion started	○	○
CKS0	Frequency select bit 0	0 : $f_{AD}/4$ is selected 1 : $f_{AD}/2$ is selected	○	○

Note 1: If the A-D control register is rewritten during A-D conversion, the conversion result is indeterminate.
Note 2: When changing A-D operation mode, set analog input pin again.

A-D control register 1 (Note)



Symbol	Address	When reset
ADCON1	$03D7_{16}$	00_{16}

Bit symbol	Bit name	Function	R	W
SCAN0	A-D sweep pin select bit	When single sweep and repeat sweep mode 0 are selected b1 b0 0 0 : AN_0, AN_1 (2 pins) 0 1 : AN_0 to AN_3 (4 pins) 1 0 : AN_0 to AN_5 (6 pins) 1 1 : AN_0 to AN_7 (8 pins)	○	○
SCAN1		When repeat sweep mode 1 is selected b1 b0 0 0 : AN_0 (1 pin) 0 1 : AN_0, AN_1 (2 pins) 1 0 : AN_0 to AN_2 (3 pins) 1 1 : AN_0 to AN_3 (4 pins)	○	○
MD2	A-D operation mode select bit 1	0 : Any mode other than repeat sweep mode 1 1 : Repeat sweep mode 1	○	○
BITS	8/10-bit mode select bit	0 : 8-bit mode 1 : 10-bit mode	○	○
CKS1	Frequency select bit 1	0 : $f_{AD}/2$ or $f_{AD}/4$ is selected 1 : f_{AD} is selected	○	○
VCUT	Vref connect bit	0 : Vref not connected 1 : Vref connected	○	○
OPA0	External op-amp connection mode bit	b7 b6 0 0 : ANEX0 and ANEX1 are not used 0 1 : ANEX0 input is A-D converted 1 0 : ANEX1 input is A-D converted 1 1 : External op-amp connection mode	○	○
OPA1			○	○

Note: If the A-D control register is rewritten during A-D conversion, the conversion result is indeterminate.

Figure 1.17.2. A-D converter-related registers (1)

A-D control register 2 (Note)

Symbol	Address	When reset
ADCON2	$03D4_{16}$	$0000XXX0_2$

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

Bit symbol	Bit name	Function	R	W
SMP	A-D conversion method select bit	0 : Without sample and hold 1 : With sample and hold	○	○
Reserved bit		Must always be set to “0”	○	○
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be “0”.			–	–

Note: If the A-D control register is rewritten during A-D conversion, the conversion result is indeterminate.

A-D register i

Symbol	Address	When reset
ADi(i=0 to 7)	$03C0_{16}$ to $03CF_{16}$	Indeterminate

(b15) b7							(b8) b0	b7							b0
X	X	X	X	X	X	X									

Function	R	W
Eight low-order bits of A-D conversion result	○	×
• During 10-bit mode Two high-order bits of A-D conversion result	○	×
• During 8-bit mode When read, the content is indeterminate	×	×
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be “0”.	–	–

Figure 1.17.3. A-D converter-related registers (2)

(1) One-shot mode

In one-shot mode, the pin selected using the analog input pin select bit is used for one-shot A-D conversion. Table 1.17.2 shows the specifications of one-shot mode. Figure 1.17.4 shows the A-D control register in one-shot mode.

Table 1.17.2. One-shot mode specifications

Item	Specification
Function	The pin selected by the analog input pin select bit is used for one A-D conversion
Start condition	Writing "1" to A-D conversion start flag
Stop condition	• End of A-D conversion (A-D conversion start flag changes to "0", except when external trigger is selected) • Writing "0" to A-D conversion start flag
Interrupt request generation timing	End of A-D conversion
Input pin	One of AN_0 to AN_7, as selected
Reading of result of A-D converter	Read A-D register corresponding to selected pin

A-D control register 0 (Note 1)

b7	b6	b5	b4	b3	b2	b1	b0
			0	0			

Symbol: ADCON0 — Address: $03D6_{16}$ — When reset: $00000XXX_2$

Bit symbol	Bit name	Function	R	W
CH0	Analog input pin select bit	b2 b1 b0 0 0 0 : AN_0 is selected 0 0 1 : AN_1 is selected 0 1 0 : AN_2 is selected 0 1 1 : AN_3 is selected 1 0 0 : AN_4 is selected 1 0 1 : AN_5 is selected 1 1 0 : AN_6 is selected 1 1 1 : AN_7 is selected (Note 2)	○	○
CH1			○	○
CH2			○	○
MD0	A-D operation mode select bit 0	b4 b3 0 0 : One-shot mode (Note 2)	○	○
MD1			○	○
TRG	Trigger select bit	0 : Software trigger 1 : AD_{TRG} trigger	○	○
ADST	A-D conversion start flag	0 : A-D conversion disabled 1 : A-D conversion started	○	○
CKS0	Frequency select bit 0	0: $f_{AD}/4$ is selected 1: $f_{AD}/2$ is selected	○	○

Note 1: If the A-D control register is rewritten during A-D conversion, the conversion result is indeterminate.
Note 2: When changing A-D operation mode, set analog input pin again.

A-D control register 1 (Note)

b7	b6	b5	b4	b3	b2	b1	b0
		1			0		

Symbol: ADCON1 — Address: $03D7_{16}$ — When reset: 00_{16}

Bit symbol	Bit name	Function	R	W
SCAN0	A-D sweep pin select bit	Invalid in one-shot mode	○	○
SCAN1			○	○
MD2	A-D operation mode select bit 1	Set to "0" when this mode is selected	○	○
BITS	8/10-bit mode select bit	0 : 8-bit mode 1 : 10-bit mode	○	○
CKS1	Frequency select bit1	0 : $f_{AD}/2$ or $f_{AD}/4$ is selected 1 : f_{AD} is selected	○	○
VCUT	Vref connect bit	1 : Vref connected	○	○
OPA0	External op-amp connection mode bit	b7 b6 0 0 : ANEX0 and ANEX1 are not used 0 1 : ANEX0 input is A-D converted 1 0 : ANEX1 input is A-D converted 1 1 : External op-amp connection mode	○	○
OPA1			○	○

Note: If the A-D control register is rewritten during A-D conversion, the conversion result is indeterminate.

Figure 1.17.4. A-D conversion register in one-shot mode

(2) Repeat mode

In repeat mode, the pin selected using the analog input pin select bit is used for repeated A-D conversion. Table 1.17.3 shows the specifications of repeat mode. Figure 1.17.5 shows the A-D control register in repeat mode.

Table 1.17.3. Repeat mode specifications

Item	Specification
Function	The pin selected by the analog input pin select bit is used for repeated A-D conversion
Star condition	Writing “1” to A-D conversion start flag
Stop condition	Writing “0” to A-D conversion start flag
Interrupt request generation timing	None generated
Input pin	One of AN_0 to AN_7, as selected
Reading of result of A-D converter	Read A-D register corresponding to selected pin (at any time)

A-D control register 0 (Note 1)

b7	b6	b5	b4	b3	b2	b1	b0
			0	1			

Symbol ADCON0 — Address $03D6_{16}$ — When reset $00000XXX_2$

Bit symbol	Bit name	Function	R	W
CH0	Analog input pin select bit	b2 b1 b0 0 0 0 : AN_0 is selected 0 0 1 : AN_1 is selected 0 1 0 : AN_2 is selected 0 1 1 : AN_3 is selected 1 0 0 : AN_4 is selected 1 0 1 : AN_5 is selected 1 1 0 : AN_6 is selected 1 1 1 : AN_7 is selected (Note 2)	○	○
CH1			○	○
CH2			○	○
MD0	A-D operation mode select bit 0	b4 b3 0 1 : Repeat mode (Note 2)	○	○
MD1			○	○
TRG	Trigger select bit	0 : Software trigger 1 : AD_{TRG} trigger	○	○
ADST	A-D conversion start flag	0 : A-D conversion disabled 1 : A-D conversion started	○	○
CKS0	Frequency select bit 0	0 : $f_{AD}/4$ is selected 1 : $f_{AD}/2$ is selected	○	○

Note 1: If the A-D control register is rewritten during A-D conversion, the conversion result is indeterminate.
Note 2: When changing A-D operation mode, set analog input pin again.

A-D control register 1 (Note)

b7	b6	b5	b4	b3	b2	b1	b0
		1			0		

Symbol ADCON1 — Address $03D7_{16}$ — When reset 00_{16}

Bit symbol	Bit name	Function	R	W
SCAN0	A-D sweep pin select bit	Invalid in repeat mode	○	○
SCAN1			○	○
MD2	A-D operation mode select bit 1	Set to “0” when this mode is selected	○	○
BITS	8/10-bit mode select bit	0 : 8-bit mode 1 : 10-bit mode	○	○
CKS1	Frequency select bit 1	0 : $f_{AD}/2$ or $f_{AD}/4$ is selected 1 : f_{AD} is selected	○	○
VCUT	Vref connect bit	1 : Vref connected	○	○
OPA0	External op-amp connection mode bit	b7 b6 0 0 : ANEX0 and ANEX1 are not used 0 1 : ANEX0 input is A-D converted 1 0 : ANEX1 input is A-D converted 1 1 : External op-amp connection mode	○	○
OPA1			○	○

Note: If the A-D control register is rewritten during A-D conversion, the conversion result is indeterminate.

Figure 1.17.5. A-D conversion register in repeat mode

(3) Single sweep mode

In single sweep mode, the pins selected using the A-D sweep pin select bit are used for one-by-one A-D conversion. Table 1.17.4 shows the specifications of single sweep mode. Figure 1.17.6 shows the A-D control register in single sweep mode.

Table 1.17.4. Single sweep mode specifications

Item	Specification
Function	The pins selected by the A-D sweep pin select bit are used for one-by-one A-D conversion
Start condition	Writing "1" to A-D converter start flag
Stop condition	• End of A-D conversion (A-D conversion start flag changes to "0", except when external trigger is selected) • Writing "0" to A-D conversion start flag
Interrupt request generation timing	End of A-D conversion
Input pin	AN_0 and AN_1 (2 pins), AN_0 to AN_3 (4 pins), AN_0 to AN_5 (6 pins), or AN_0 to AN_7 (8 pins)
Reading of result of A-D converter	Read A-D register corresponding to selected pin

A-D control register 0 (Note)

b7	b6	b5	b4	b3	b2	b1	b0
			1	0			

Symbol	Address	When reset
ADCON0	$03D6_{16}$	$00000XXX_2$

Bit symbol	Bit name	Function	R	W
CH0	Analog input pin select bit	Invalid in single sweep mode	O	O
CH1			O	O
CH2			O	O
MD0	A-D operation mode select bit 0	b4 b3 1 0 : Single sweep mode	O	O
MD1			O	O
TRG	Trigger select bit	0 : Software trigger 1 : AD_{TRG} trigger	O	O
ADST	A-D conversion start flag	0 : A-D conversion disabled 1 : A-D conversion started	O	O
CKS0	Frequency select bit 0	0 : $f_{AD}/4$ is selected 1 : $f_{AD}/2$ is selected	O	O

Note: If the A-D control register is rewritten during A-D conversion, the conversion result is indeterminate.

A-D control register 1 (Note 1)

b7	b6	b5	b4	b3	b2	b1	b0
		1			0		

Symbol	Address	When reset
ADCON1	$03D7_{16}$	00_{16}

Bit symbol	Bit name	Function	R	W
SCAN0	A-D sweep pin select bit	When single sweep and repeat sweep mode 0 are selected b1 b0 0 0 : AN_0, AN_1 (2 pins) 0 1 : AN_0 to AN_3 (4 pins) 1 0 : AN_0 to AN_5 (6 pins) 1 1 : AN_0 to AN_7 (8 pins)	O	O
SCAN1			O	O
MD2	A-D operation mode select bit 1	Set to "0" when this mode is selected	O	O
BITS	8/10-bit mode select bit	0 : 8-bit mode 1 : 10-bit mode	O	O
CKS1	Frequency select bit 1	0 : $f_{AD}/2$ or $f_{AD}/4$ is selected 1 : f_{AD} is selected	O	O
VCUT	Vref connect bit	1 : Vref connected	O	O
OPA0	External op-amp connection mode bit (Note 2)	b7 b6 0 0 : ANEX0 and ANEX1 are not used 0 1 : ANEX0 input is A-D converted 1 0 : ANEX1 input is A-D converted 1 1 : External op-amp connection mode	O	O
OPA1			O	O

Note 1: If the A-D control register is rewritten during A-D conversion, the conversion result is indeterminate.
Note 2: Neither '01' nor '10' can be selected with the external op-amp connection mode bit.

Figure 1.17.6. A-D conversion register in single sweep mode

(4) Repeat sweep mode 0

In repeat sweep mode 0, the pins selected using the A-D sweep pin select bit are used for repeat sweep A-D conversion. Table 1.17.5 shows the specifications of repeat sweep mode 0. Figure 1.17.7 shows the A-D control register in repeat sweep mode 0.

Table 1.17.5. Repeat sweep mode 0 specifications

Item	Specification
Function	The pins selected by the A-D sweep pin select bit are used for repeat A-D conversion
Start condition	Writing "1" to A-D conversion start flag
Stop condition	Writing "0" to A-D conversion start flag
Interrupt request generation timing	None generated
Input pin	AN_0 and AN_1 (2 pins), AN_0 to AN_3 (4 pins), AN_0 to AN_5 (6 pins), or AN_0 to AN_7 (8 pins)
Reading of result of A-D converter	Read A-D register corresponding to selected pin (at any time)

A-D control register 0 (Note)

b7	b6	b5	b4	b3	b2	b1	b0
			1	1			

Symbol: ADCON0 — Address: $03D6_{16}$ — When reset: $00000XXX_2$

Bit symbol	Bit name	Function	R	W
CH0	Analog input pin select bit	Invalid in repeat sweep mode 0	O	O
CH1			O	O
CH2			O	O
MD0	A-D operation mode select bit 0	b4 b3 1 1 : Repeat sweep mode 0	O	O
MD1			O	O
TRG	Trigger select bit	0 : Software trigger 1 : AD_{TRG} trigger	O	O
ADST	A-D conversion start flag	0 : A-D conversion disabled 1 : A-D conversion started	O	O
CKS0	Frequency select bit 0	0 : $f_{AD}/4$ is selected 1 : $f_{AD}/2$ is selected	O	O

Note: If the A-D control register is rewritten during A-D conversion, the conversion result is indeterminate.

A-D control register 1 (Note 1)

b7	b6	b5	b4	b3	b2	b1	b0
		1			0		

Symbol: ADCON1 — Address: $03D7_{16}$ — When reset: 00_{16}

Bit symbol	Bit name	Function	R	W
SCAN0	A-D sweep pin select bit	When single sweep and repeat sweep mode 0 are selected b1 b0 0 0 : AN_0, AN_1 (2 pins) 0 1 : AN_0 to AN_3 (4 pins) 1 0 : AN_0 to AN_5 (6 pins) 1 1 : AN_0 to AN_7 (8 pins)	O	O
SCAN1			O	O
MD2	A-D operation mode select bit 1	Set to "0" when this mode is selected	O	O
BITS	8/10-bit mode select bit	0 : 8-bit mode 1 : 10-bit mode	O	O
CKS1	Frequency select bit 1	0 : $f_{AD}/2$ or $f_{AD}/4$ is selected 1 : f_{AD} is selected	O	O
VCUT	Vref connect bit	1 : Vref connected	O	O
OPA0	External op-amp connection mode bit (Note 2)	b7 b6 0 0 : ANEX0 and ANEX1 are not used 0 1 : ANEX0 input is A-D converted 1 0 : ANEX1 input is A-D converted 1 1 : External op-amp connection mode	O	O
OPA1			O	O

Note 1: If the A-D control register is rewritten during A-D conversion, the conversion result is indeterminate.
Note 2: Neither "01" nor "10" can be selected with the external op-amp connection mode bit.

Figure 1.17.7. A-D conversion register in repeat sweep mode 0

(5) Repeat sweep mode 1

In repeat sweep mode 1, all pins are used for A-D conversion with emphasis on the pin or pins selected using the A-D sweep pin select bit. Table 1.17.6 shows the specifications of repeat sweep mode 1. Figure 1.17.8 shows the A-D control register in repeat sweep mode 1.

Table 1.17.6. Repeat sweep mode 1 specifications

Item	Specification
Function	All pins perform repeat A-D conversion, with emphasis on the pin or pins selected by the A-D sweep pin select bit Example : AN_0 selected $AN_0 \rightarrow AN_1 \rightarrow AN_0 \rightarrow AN_2 \rightarrow AN_0 \rightarrow AN_3$, etc
Start condition	Writing "1" to A-D conversion start flag
Stop condition	Writing "0" to A-D conversion start flag
Interrupt request generation timing	None generated
Input pin	With emphasis on these pins ; AN_0 (1 pin), AN_0 and AN_1 (2 pins), AN_0 to AN_2 (3 pins), AN_0 to AN_3 (4 pins)
Reading of result of A-D converter	Read A-D register corresponding to selected pin (at any time)

A-D control register 0 (Note)

b7	b6	b5	b4	b3	b2	b1	b0
			1	1			

Symbol ADCON0 — Address $03D6_{16}$ — When reset $00000XXX_2$

Bit symbol	Bit name	Function	R	W
CH0	Analog input pin select bit	Invalid in repeat sweep mode 1	O	O
CH1			O	O
CH2			O	O
MD0	A-D operation mode select bit 0	b4 b3 1 1 : Repeat sweep mode 1	O	O
MD1			O	O
TRG	Trigger select bit	0 : Software trigger 1 : AD_{TRG} trigger	O	O
ADST	A-D conversion start flag	0 : A-D conversion disabled 1 : A-D conversion started	O	O
CKS0	Frequency select bit 0	0 : $f_{AD}/4$ is selected 1 : $f_{AD}/2$ is selected	O	O

Note: If the A-D control register is rewritten during A-D conversion, the conversion result is indeterminate.

A-D control register 1 (Note 1)

b7	b6	b5	b4	b3	b2	b1	b0
		1			1		

Symbol ADCON1 — Address $03D7_{16}$ — When reset 00_{16}

Bit symbol	Bit name	Function	R	W
SCAN0	A-D sweep pin select bit	When repeat sweep mode 1 is selected b1 b0 0 0 : AN_0 (1 pin) 0 1 : AN_0, AN_1 (2 pins) 1 0 : AN_0 to AN_2 (3 pins) 1 1 : AN_0 to AN_3 (4 pins)	O	O
SCAN1			O	O
MD2	A-D operation mode select bit 1	Set to "1" when this mode is selected	O	O
BITS	8/10-bit mode select bit	0 : 8-bit mode 1 : 10-bit mode	O	O
CKS1	Frequency select bit 1	0 : $f_{AD}/2$ or $f_{AD}/4$ is selected 1 : f_{AD} is selected	O	O
VCUT	Vref connect bit	1 : Vref connected	O	O
OPA0	External op-amp connection mode bit (Note 2)	b7 b6 0 0 : ANEX0 and ANEX1 are not used 0 1 : ANEX0 input is A-D converted 1 0 : ANEX1 input is A-D converted 1 1 : External op-amp connection mode	O	O
OPA1			O	O

Note 1: If the A-D control register is rewritten during A-D conversion, the conversion result is indeterminate.
Note 2: Neither '01' nor '10' can be selected with the external op-amp connection mode bit.

Figure 1.17.8. A-D conversion register in repeat sweep mode 1

(a) Sample and hold

Sample and hold is selected by setting bit 0 of the A-D control register 2 (address 03D4$_{16}$) to “1”. When sample and hold is selected, the rate of conversion of each pin increases. As a result, a 28 ϕ_{AD} cycle is achieved with 8-bit resolution and 33 ϕ_{AD} with 10-bit resolution. Sample and hold can be selected in all modes. However, in all modes, be sure to specify before starting A-D conversion whether sample and hold is to be used.

(b) Extended analog input pins

In one-shot mode and repeat mode, the input via the extended analog input pins ANEX0 and ANEX1 can also be converted from analog to digital.

When bit 6 of the A-D control register 1 (address 03D7$_{16}$) is “1” and bit 7 is “0”, input via ANEX0 is converted from analog to digital. The result of conversion is stored in A-D register 0.

When bit 6 of the A-D control register 1 (address 03D7$_{16}$) is “0” and bit 7 is “1”, input via ANEX1 is converted from analog to digital. The result of conversion is stored in A-D register 1.

(c) External operation amp connection mode

In this mode, multiple external analog inputs via the extended analog input pins, ANEX0 and ANEX1, can be amplified together by just one operation amp and used as the input for A-D conversion.

When bit 6 of the A-D control register 1 (address 03D7$_{16}$) is “1” and bit 7 is “1”, input via AN$_0$ to AN$_7$ is output from ANEX0. The input from ANEX1 is converted from analog to digital and the result stored in the corresponding A-D register. The speed of A-D conversion depends on the response of the external operation amp. Do not connect the ANEX0 and ANEX1 pins directly. Figure 1.17.9 is an example of how to connect the pins in external operation amp mode.

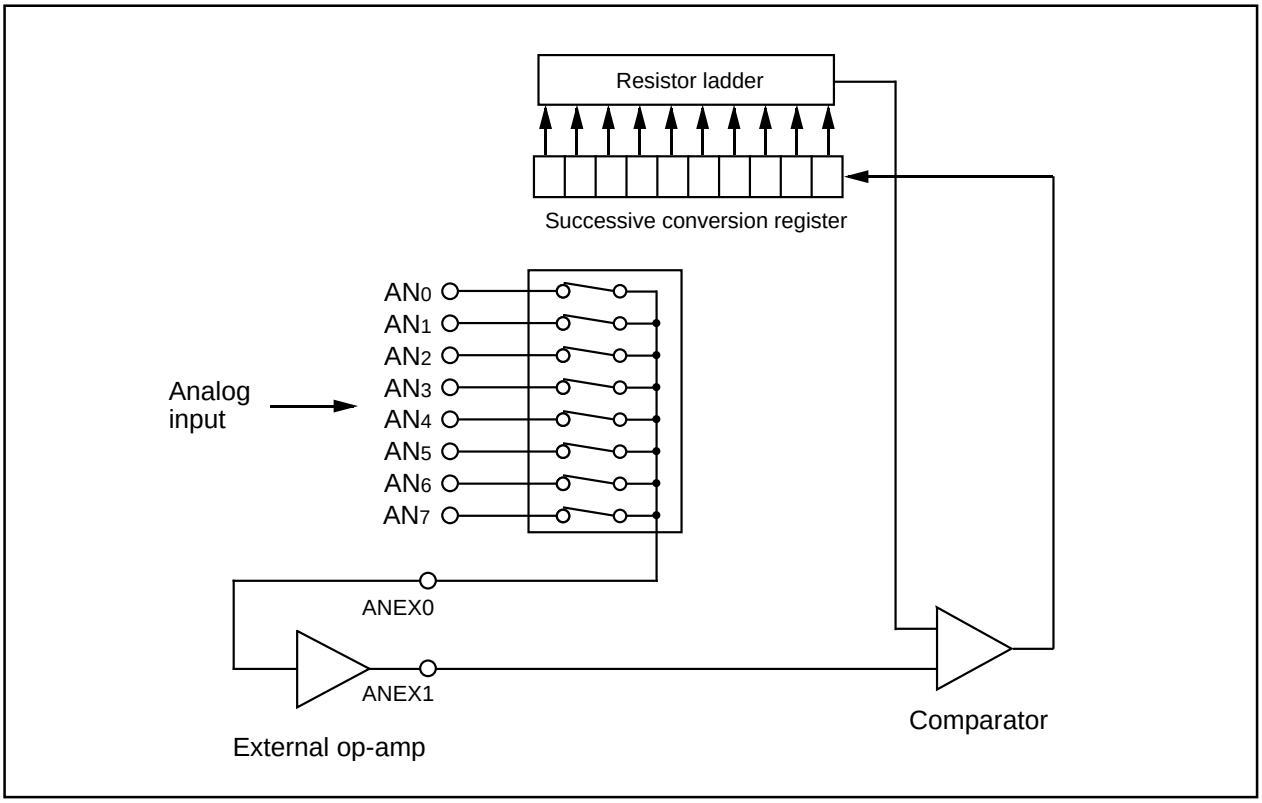


Figure 1.17.9. Example of external op-amp connection mode

D-A Converter

This is an 8-bit, R-2R type D-A converter. The microcomputer contains two independent D-A converters of this type.

D-A conversion is performed when a value is written to the corresponding D-A register. Bits 0 and 1 (D-A output enable bits) of the D-A control register decide if the result of conversion is to be output. Do not set the target port to output mode if D-A conversion is to be performed. When the D-A output is enabled, the pull-up function of the corresponding port is automatically disabled.

Output analog voltage (V) is determined by a set value (n : decimal) in the D-A register.

$$V = V_{REF} \times n / 256 \quad (n = 0 \text{ to } 255)$$

V_{REF} : reference voltage

Table 1.18.1 lists the performance of the D-A converter. Figure 1.18.1 shows the block diagram of the D-A converter. Figure 1.18.2 shows the D-A control register. Figure 1.18.3 shows the D-A converter equivalent circuit.

Table 1.18.1. Performance of D-A converter

Item	Performance
Conversion method	R-2R method
Resolution	8 bits
Analog output pin	2 channels

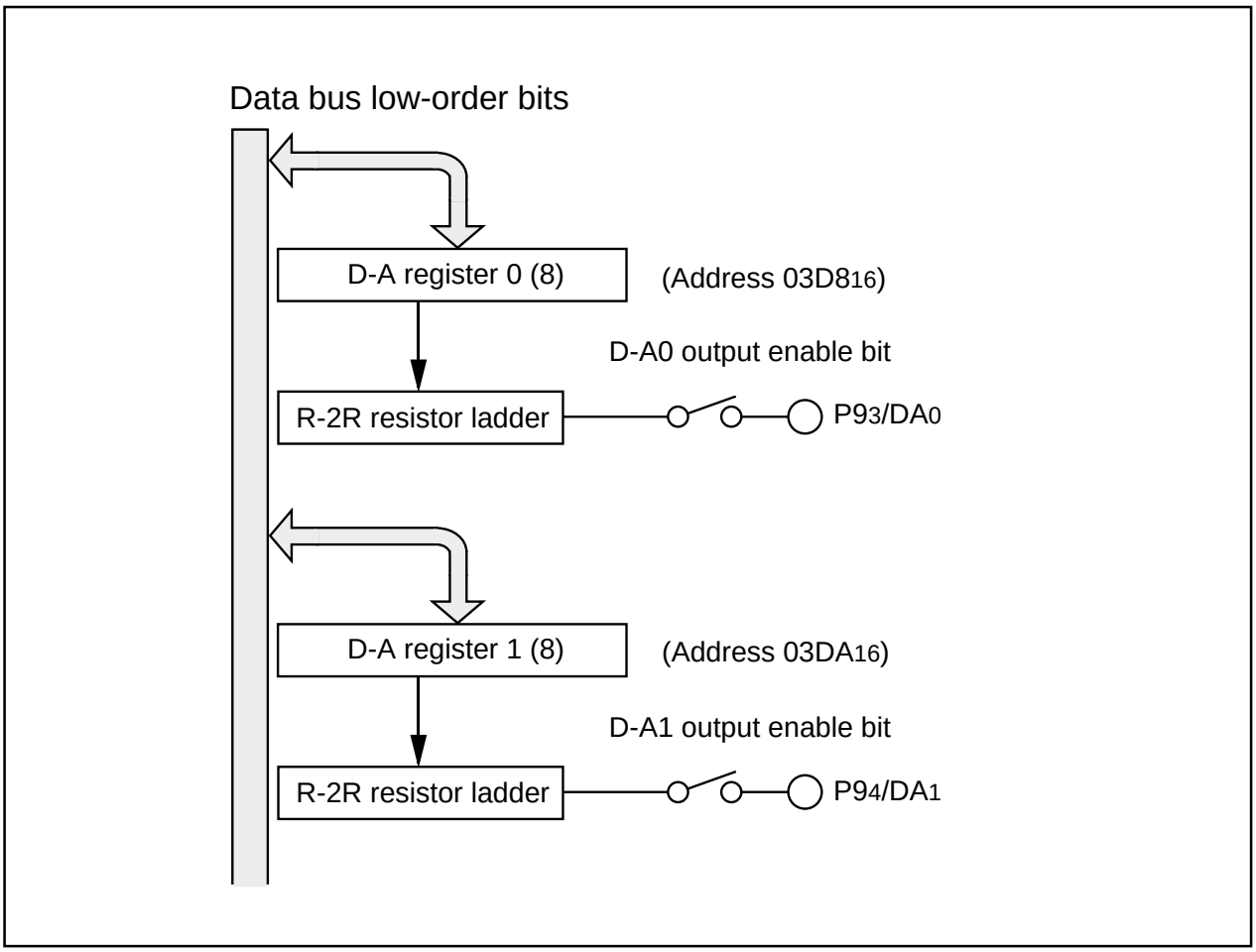


Figure 1.18.1. Block diagram of D-A converter

D-A control register

Symbol	Address	When reset
DACON	$03DC_{16}$	00_{16}

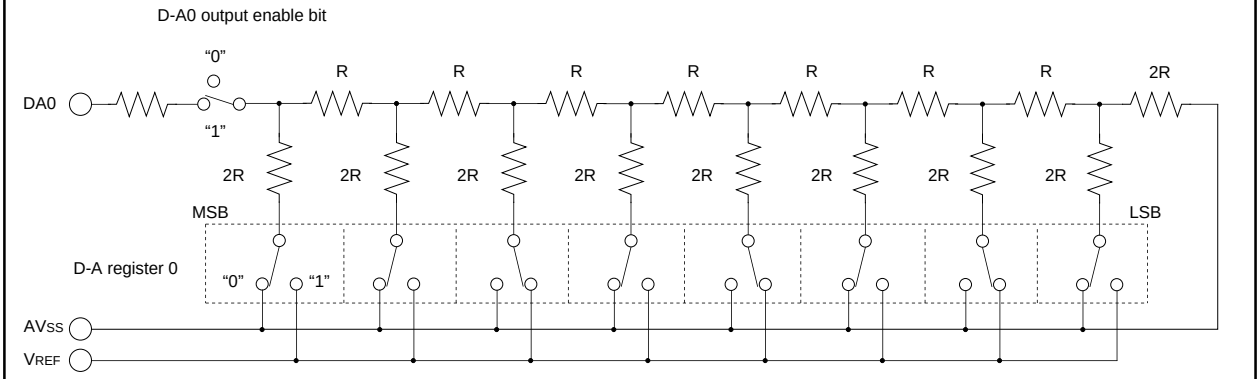
Bit symbol	Bit name	Function	R	W
DA0E	D-A0 output enable bit	0 : Output disabled 1 : Output enabled	○	○
DA1E	D-A1 output enable bit	0 : Output disabled 1 : Output enabled	○	○
Nothing is assigned. In an attempt to write to these bits, write "0". The value, if read, turns out to be "0"			–	–

D-A register

Symbol	Address	When reset
DAi (i = 0,1)	$03D8_{16}$, $03DA_{16}$	Indeterminate

Function	R	W
Output value of D-A conversion	○	○

Figure 1.18.2. D-A control register



Note 1: The above diagram shows an instance in which the D-A register is assigned $2A_{16}$.
Note 2: The same circuit as this is also used for D-A1.
Note 3: To reduce the current consumption when the D-A converter is not used, set the D-A output enable bit to 0 and set the D-A register to 00_{16} so that no current flows in the resistors Rs and 2Rs.

Figure 1.18.3. D-A converter equivalent circuit

CRC Calculation Circuit

The Cyclic Redundancy Check (CRC) calculation circuit detects an error in data blocks. The microcomputer uses a generator polynomial of CRC_CCITT ($X^{16} + X^{12} + X^5 + 1$) to generate CRC code.

The CRC code is a 16-bit code generated for a block of a given data length in multiples of 8 bits. The CRC code is set in a CRC data register each time one byte of data is transferred to a CRC input register after writing an initial value into the CRC data register. Generation of CRC code for one byte of data is completed in two machine cycles.

Figure 1.19.1 shows the block diagram of the CRC circuit. Figure 1.19.2 shows the CRC-related registers. Figure 1.19.3 shows the calculation example using the CRC calculation circuit.

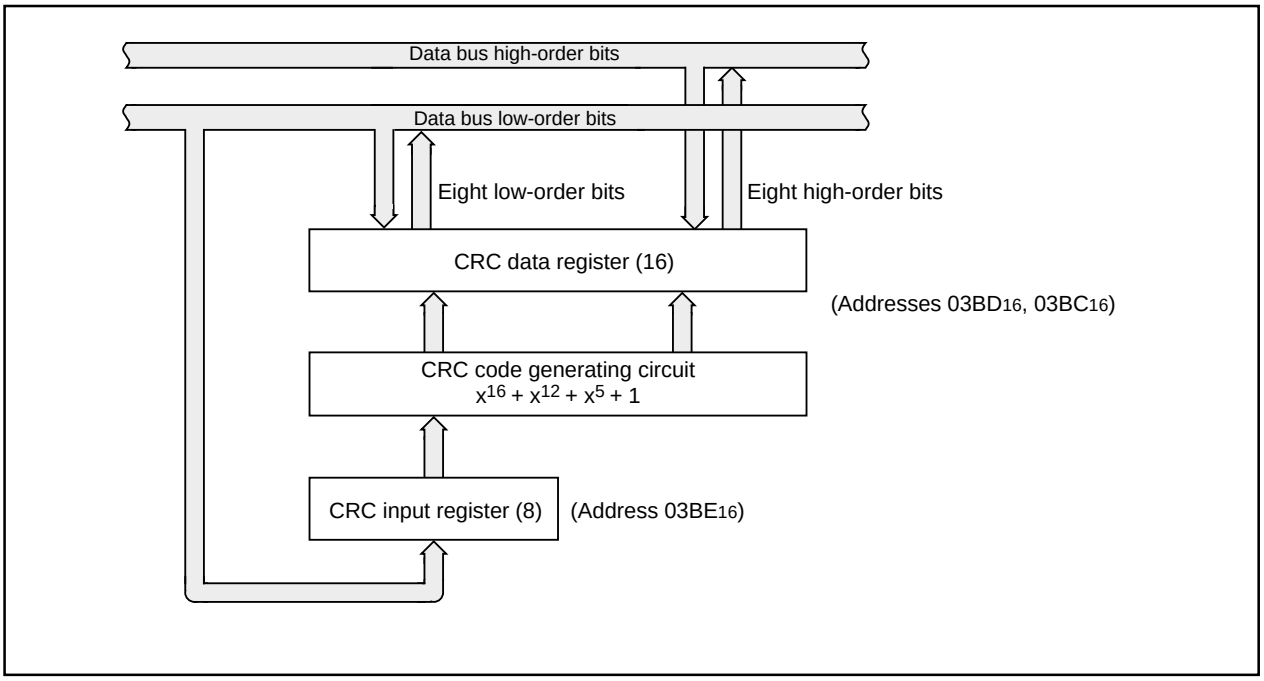


Figure 1.19.1. Block diagram of CRC circuit

CRC data register

(b15) b7 (b8) b0 b7 b0

Symbol	Address	When reset
CRCD	03BD16, 03BC16	Indeterminate

Function	Values that can be set	R	W
CRC calculation result output register	000016 to FFFF16	○	○

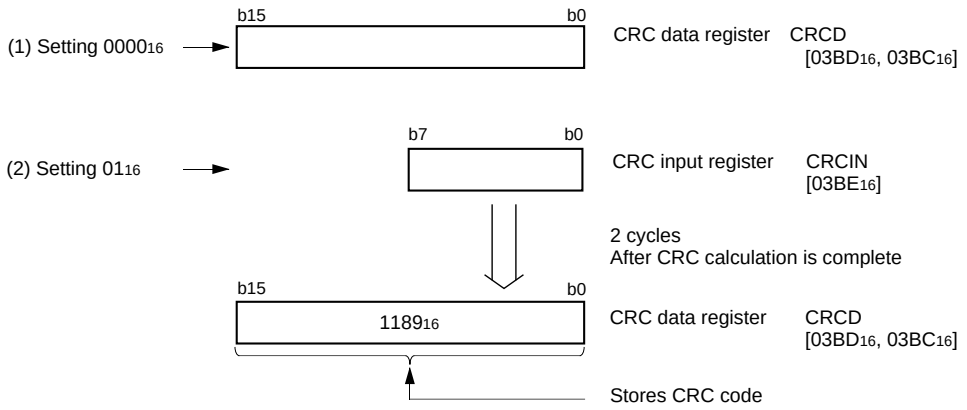
CRC input register

b7 b0

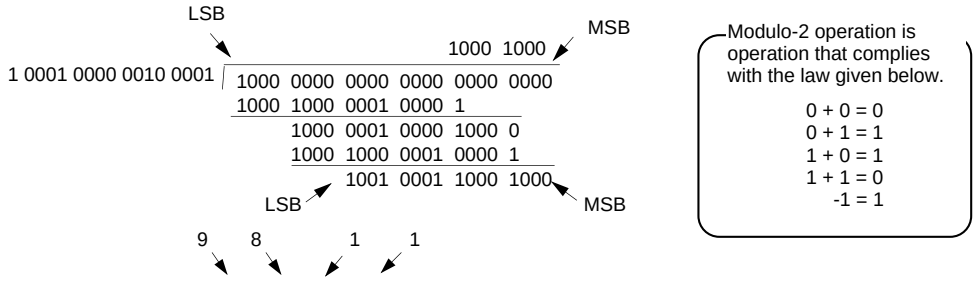
Symbo	Address	When reset
CRCIN	03BE16	Indeterminate

Function	Values that can be set	R	W
Data input register	0016 to FF16	○	○

Figure 1.19.2. CRC-related registers



The code resulting from sending 01_{16} in LSB first mode is (1000 0000). Thus the CRC code in the generating polynomial, $(X^{16} + X^{12} + X^5 + 1)$, becomes the remainder resulting from dividing (1000 0000) X^{16} by (1 0001 0000 0010 0001) in conformity with the modulo-2 operation.



Thus the CRC code becomes (1001 0001 1000 1000). Since the operation is in LSB first mode, the (1001 0001 1000 1000) corresponds to 1189_{16} in hexadecimal notation. If the CRC operation in MSB first mode is necessary in the CRC operation circuit built in the M16C, switch between the LSB side and the MSB side of the input-holding bits, and carry out the CRC operation. Also switch between the MSB and LSB of the result as stored in CRC data.

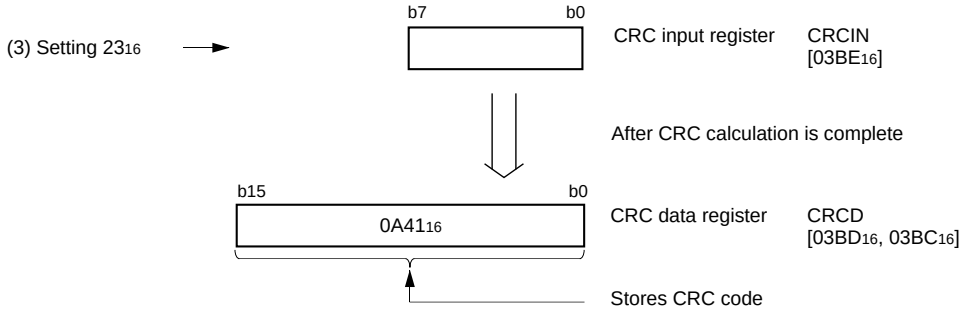


Figure 1.19.3. Calculation example using the CRC calculation circuit

Programmable I/O Ports

There are 87 programmable I/O ports: P0 to P10 (excluding $P8_5$). Each port can be set independently for input or output using the direction register. A pull-up resistance for each block of 4 ports can be set. $P8_5$ is an input-only port and has no built-in pull-up resistance.

Figures 1.20.1 to 1.20.4 show the programmable I/O ports. Figure 1.20.5 shows the I/O pins.

Each pin functions as a programmable I/O port and as the I/O for the built-in peripheral devices.

To use the pins as the inputs for the built-in peripheral devices, set the direction register of each pin to input mode. When the pins are used as the outputs for the built-in peripheral devices (other than the D-A converter), they function as outputs regardless of the contents of the direction registers. When pins are to be used as the outputs for the D-A converter, do not set the direction registers to output mode. See the descriptions of the respective functions for how to set up the built-in peripheral devices.

(1) Direction registers

Figure 1.20.6 shows the direction registers.

These registers are used to choose the direction of the programmable I/O ports. Each bit in these registers corresponds one for one to each I/O pin.

In memory expansion and microprocessor mode, the contents of corresponding direction register of pins A_0 to A_{19}, D_0 to D_{15}, $\overline{CS0}$ to $\overline{CS3}$, $\overline{RD}$, $\overline{WRL}/\overline{WR}$, $\overline{WRH}/\overline{BHE}$, ALE, $\overline{RDY}$, $\overline{HOLD}$, $\overline{HLDA}$ and BCLK cannot be modified.

Note: There is no direction register bit for $P8_5$.

(2) Port registers

Figure 1.20.7 shows the port registers.

These registers are used to write and read data for input and output to and from an external device. A port register consists of a port latch to hold output data and a circuit to read the status of a pin. Each bit in port registers corresponds one for one to each I/O pin.

In memory expansion and microprocessor mode, the contents of corresponding port register of pins A_0 to A_{19}, D_0 to D_{15}, $\overline{CS0}$ to $\overline{CS3}$, $\overline{RD}$, $\overline{WRL}/\overline{WR}$, $\overline{WRH}/\overline{BHE}$, ALE, $\overline{RDY}$, $\overline{HOLD}$, $\overline{HLDA}$ and BCLK cannot be modified.

(3) Pull-up control registers

Figure 1.20.8 shows the pull-up control registers.

The pull-up control register can be set to apply a pull-up resistance to each block of 4 ports. When ports are set to have a pull-up resistance, the pull-up resistance is connected only when the direction register is set for input.

However, in memory expansion mode and microprocessor mode, the pull-up control register of P0 to P3, $P4_0$ to $P4_3$, and P5 is invalid. The contents of register can be changed, but the pull-up resistance is not connected.

(4) Port control register

Figure 1.20.9 shows the port control register.

The bit 0 of port control register is used to read port P1 as follows:

0 : When port P1 is input port, port input level is read.
When port P1 is output port , the contents of port P1 register is read.

1 : The contents of port P1 register is read always.

This register is valid in the following:

- External bus width is 8 bits in microprocessor mode or memory expansion mode.
- Port P1 can be used as a port in multiplexed bus for the entire space.

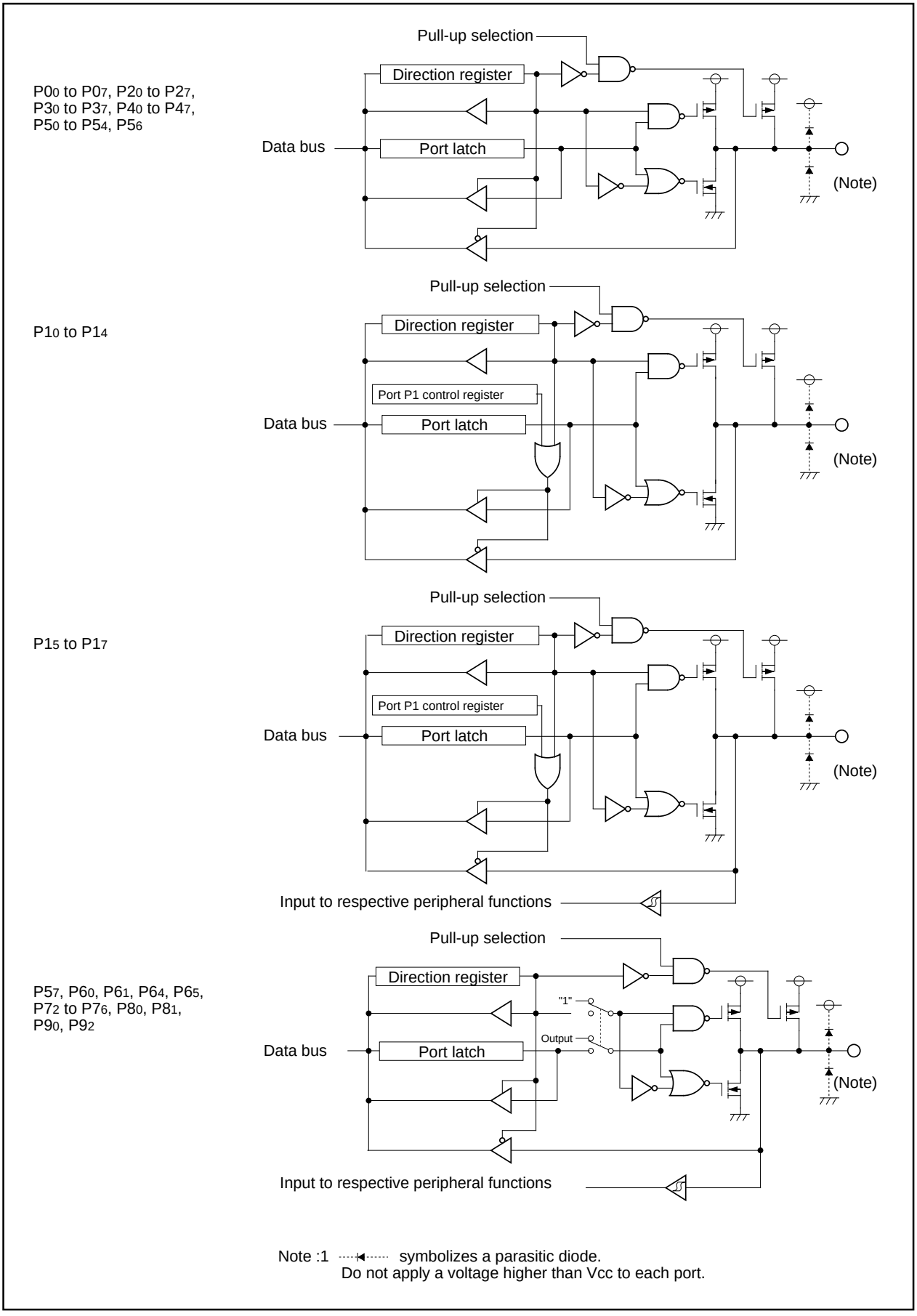


Figure 1.20.1. Programmable I/O ports (1)

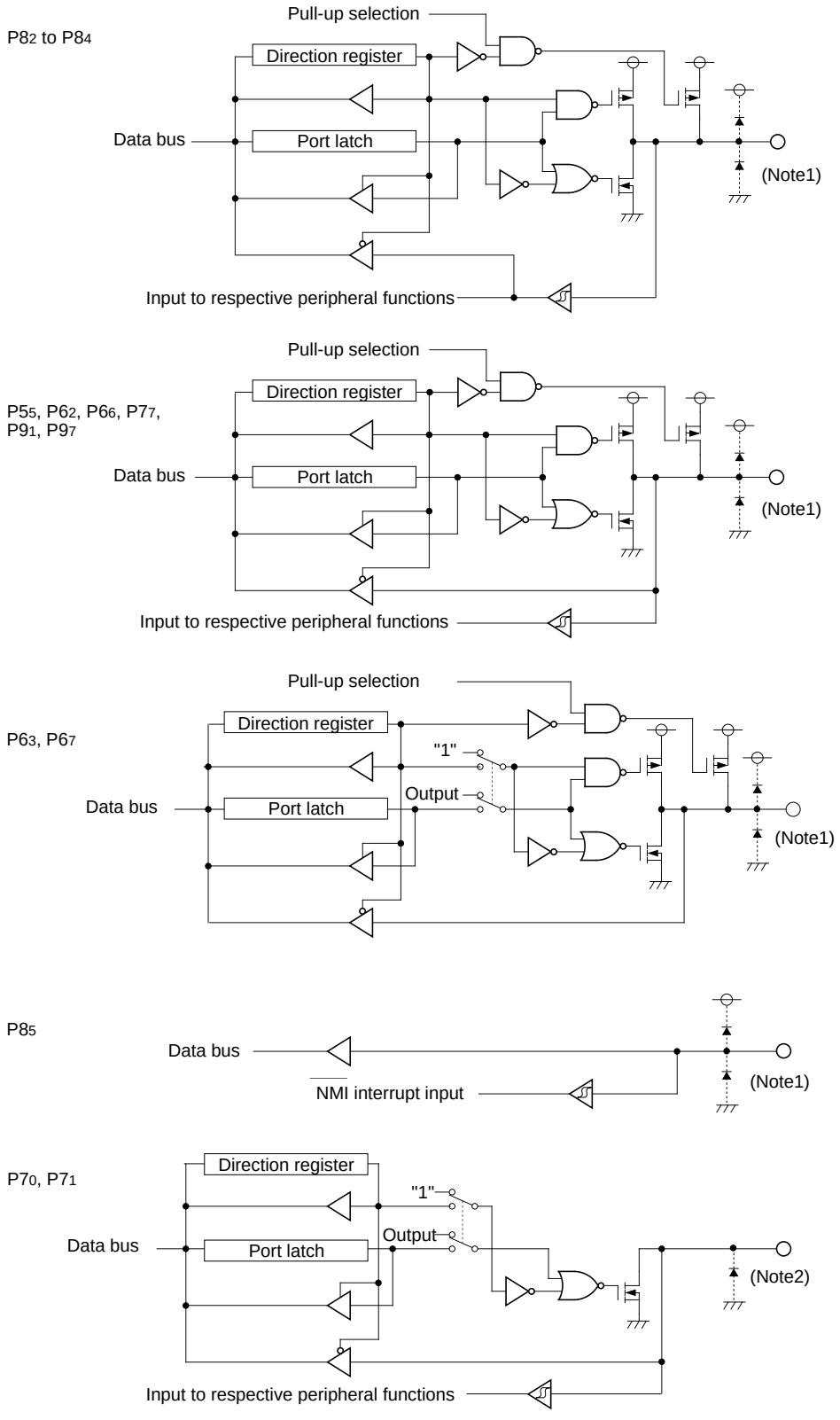


Figure 1.20.2. Programmable I/O ports (2)

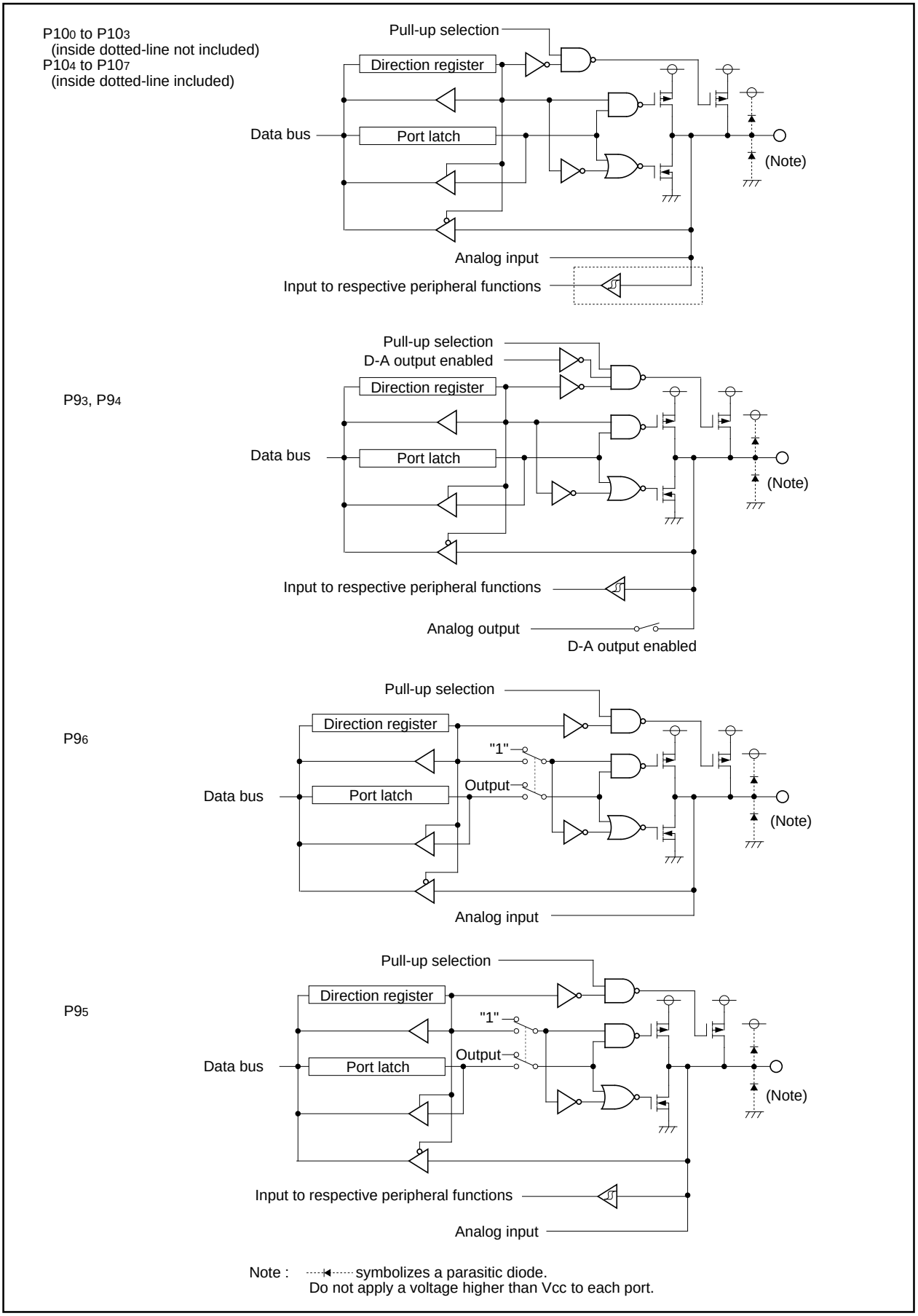


Figure 1.20.3. Programmable I/O ports (3)

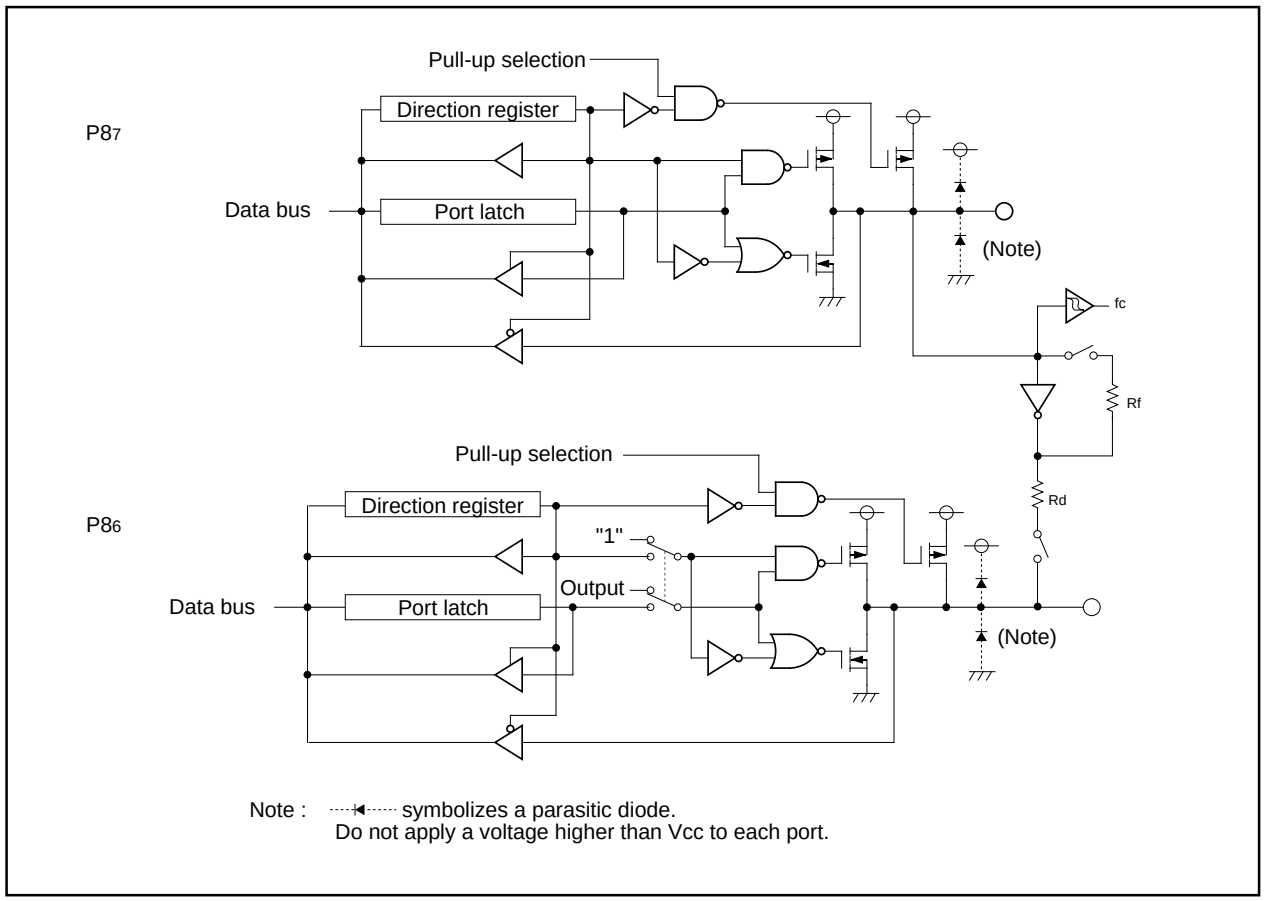


Figure 1.20.4. Programmable I/O ports (4)

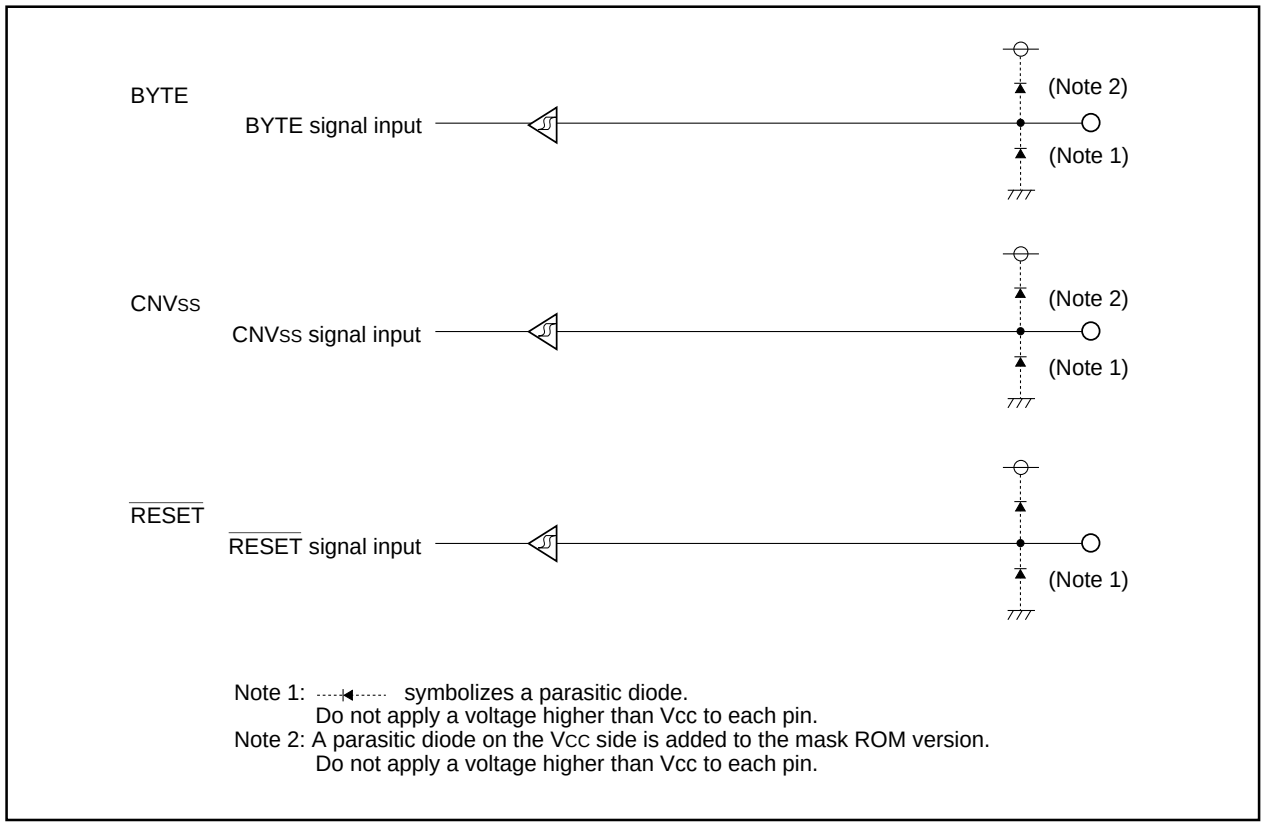


Figure 1.20.5. I/O pins

Port Pi direction register (Note 1, 2)

b7	b6	b5	b4	b3	b2	b1	b0

Symbol	Address	When reset
PDi (i = 0 to 10, except 8)	03E2$_{16}$, 03E3$_{16}$, 03E6$_{16}$, 03E7$_{16}$, 03EA$_{16}$ 03EB$_{16}$, 03EE$_{16}$, 03EF$_{16}$, 03F3$_{16}$, 03F6$_{16}$	00$_{16}$

Bit symbol	Bit name	Function	R	W
PDi_0	Port Pi$_0$ direction register	0 : Input mode (Functions as an input port) 1 : Output mode (Functions as an output port) (i = 0 to 10 except 8)	○	○
PDi_1	Port Pi$_1$ direction register		○	○
PDi_2	Port Pi$_2$ direction register		○	○
PDi_3	Port Pi$_3$ direction register		○	○
PDi_4	Port Pi$_4$ direction register		○	○
PDi_5	Port Pi$_5$ direction register		○	○
PDi_6	Port Pi$_6$ direction register		○	○
PDi_7	Port Pi$_7$ direction register		○	○

Note 1: Set bit 2 of protect register (address 000A$_{16}$) to “1” before rewriting to the port P9 direction register.
Note 2: In memory expansion and microprocessor mode, the contents of corresponding port Pi direction register of pins A$_0$ to A$_{19}$, D$_0$ to D$_{15}$, CS$_0$ to CS$_3$, RD, WRL/WR, WRH/BHE, ALE, RDY, HOLD, HLDA and BCLK cannot be modified.

Port P8 direction register

b7	b6	b5	b4	b3	b2	b1	b0
		☒					

Symbol	Address	When reset
PD8	03F2$_{16}$	00X00000$_2$

Bit symbol	Bit name	Function	R	W
PD8_0	Port P8$_0$ direction register	0 : Input mode (Functions as an input port) 1 : Output mode (Functions as an output port)	○	○
PD8_1	Port P8$_1$ direction register		○	○
PD8_2	Port P8$_2$ direction register		○	○
PD8_3	Port P8$_3$ direction register		○	○
PD8_4	Port P8$_4$ direction register		○	○
Nothing is assigned. In an attempt to write to this bit, write “0”. The value, if read, turns out to be indeterminate.			—	—
PD8_6	Port P8$_6$ direction register	0 : Input mode (Functions as an input port) 1 : Output mode (Functions as an output port)	○	○
PD8_7	Port P8$_7$ direction register		○	○

Figure 1.20.6. Direction register

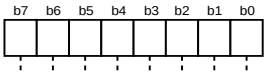
Port Pi register (Note 2)

Symbol	Address	When reset
Pi (i = 0 to 10, except 8)	03E0$_{16}$, 03E1$_{16}$, 03E4$_{16}$, 03E5$_{16}$, 03E8$_{16}$ 03E9$_{16}$, 03EC$_{16}$, 03ED$_{16}$, 03F1$_{16}$, 03F4$_{16}$	Indeterminate Indeterminate

Bit symbol	Bit name	Function	R	W
Pi_0	Port Pi$_0$ register	Data is input and output to and from each pin by reading and writing to and from each corresponding bit 0 : "L" level data 1 : "H" level data (Note 1) (i = 0 to 10 except 8)	○	○
Pi_1	Port Pi$_1$ register		○	○
Pi_2	Port Pi$_2$ register		○	○
Pi_3	Port Pi$_3$ register		○	○
Pi_4	Port Pi$_4$ register		○	○
Pi_5	Port Pi$_5$ register		○	○
Pi_6	Port Pi$_6$ register		○	○
Pi_7	Port Pi$_7$ register		○	○

Note 1: Since P7$_0$ and P7$_1$ are N-channel open drain ports, the data is high-impedance.
Note 2: In memory expansion and microprocessor mode, the contents of corresponding port Pi register of pins A$_0$ to A$_{19}$, D$_0$ to D$_{15}$, $\overline{CS_0}$ to $\overline{CS_3}$, $\overline{RD}$, WRL/$\overline{WR}$, WRH/$\overline{BHE}$, ALE, $\overline{RDY}$, $\overline{HOLD}$, $\overline{HLDA}$ and BCLK cannot be modified.

Port P8 register

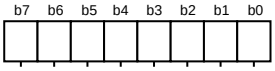


Symbol	Address	When reset
P8	03F0$_{16}$	Indeterminate

Bit symbol	Bit name	Function	R	W
P8_0	Port P8$_0$ register	Data is input and output to and from each pin by reading and writing to and from each corresponding bit (except for P8$_5$) 0 : "L" level data 1 : "H" level data	○	○
P8_1	Port P8$_1$ register		○	○
P8_2	Port P8$_2$ register		○	○
P8_3	Port P8$_3$ register		○	○
P8_4	Port P8$_4$ register		○	○
P8_5	Port P8$_5$ register		○	×
P8_6	Port P8$_6$ register		○	○
P8_7	Port P8$_7$ register		○	○

Figure 1.20.7. Port register

Pull-up control register 0 (Note)

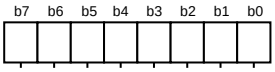


Symbol: PUR0 | Address: 03FC16 | When reset: 0016

Bit symbol	Bit name	Function	R	W
PU00	P00 to P03 pull-up	The corresponding port is pulled high with a pull-up resistor 0 : Not pulled high 1 : Pulled high	○	○
PU01	P04 to P07 pull-up		○	○
PU02	P10 to P13 pull-up		○	○
PU03	P14 to P17 pull-up		○	○
PU04	P20 to P23 pull-up		○	○
PU05	P24 to P27 pull-up		○	○
PU06	P30 to P33 pull-up		○	○
PU07	P34 to P37 pull-up		○	○

Note : In memory expansion and microprocessor mode, the content of this register can be changed, but the pull-up resistance is not connected.

Pull-up control register 1

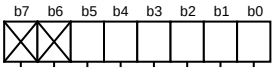


Symbol: PUR1 | Address: 03FD16 | When reset: 0016 (Note 2)

Bit symbol	Bit name	Function	R	W
PU10	P40 to P43 pull-up (Note 3)	The corresponding port is pulled high with a pull-up resistor 0 : Not pulled high 1 : Pulled high	○	○
PU11	P44 to P47 pull-up		○	○
PU12	P50 to P53 pull-up (Note 3)		○	○
PU13	P54 to P57 pull-up		○	○
PU14	P60 to P63 pull-up		○	○
PU15	P64 to P67 pull-up		○	○
PU16	P72 to P73 pull-up (Note 1)		○	○
PU17	P74 to P77 pull-up		○	○

Note 1: Since P70 and P71 are N-channel open drain ports, pull-up is not available for them.
Note 2: When the VCC level is being impressed to the CNVSS terminal, this register becomes to 0216 when reset (PU11 becomes to “1”).
Note 3: In memory expansion and microprocessor mode, the content of these bits can be changed, but the pull-up resistance is not connected.

Pull-up control register 2

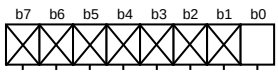


Symbol: PUR2 | Address: 03FE16 | When reset: 0016

Bit symbol	Bit name	Function	R	W
PU20	P80 to P83 pull-up	The corresponding port is pulled high with a pull-up resistor 0 : Not pulled high 1 : Pulled high	○	○
PU21	P84 to P87 pull-up (Except P85)		○	○
PU22	P90 to P93 pull-up		○	○
PU23	P94 to P97 pull-up		○	○
PU24	P100 to P103 pull-up		○	○
PU25	P104 to P107 pull-up		○	○
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be “0”.			—	—
			—	—

Figure 1.20.8. Pull-up control register

Port control register



Symbpl: PCR
Address: 03FF16
When reset: 0016

Bit symbol	Bit name	Function	R	W
PCR0	Port P1 control register	0 : When input port, read port input level. When output port, read the contents of port P1 register. 1 : Read the contents of port P1 register though input/output port.	○	○
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be “0”.			—	—

Figure 1.20.9. Port control register

Table 1.20.1. Example connection of unused pins in single-chip mode

Pin name	Connection
Ports P0 to P10 (excluding P8$_5$)	After setting for input mode, connect every pin to V_{SS} via a resistor (pull-down); or after setting for output mode, leave these pins open.
X_{OUT} (Note)	Open
$\overline{NMI}$	Connect via resistor to V_{CC} (pull-up)
AV_{CC}	Connect to V_{CC}
AV_{SS}, V_{REF}, BYTE	Connect to V_{SS}

Note: With external clock input to X_{IN} pin.

Table 1.20.2. Example connection of unused pins in memory expansion mode and microprocessor mode

Pin name	Connection
Ports P6 to P10 (excluding P8$_5$)	After setting for input mode, connect every pin to V_{SS} via a resistor (pull-down); or after setting for output mode, leave these pins open.
P4$_5$ / $\overline{CS1}$ to P4$_7$ / $\overline{CS3}$	Set ports to input mode, set output enable bits of $\overline{CS1}$ through $\overline{CS3}$ to 0, and connect to Vcc via resistors (pull-up).
$\overline{BHE}$, ALE, $\overline{HLDA}$, X_{OUT} (Note 1), BCLK (Note 2)	Open
$\overline{HOLD}$, $\overline{RDY}$, $\overline{NMI}$	Connect via resistor to V_{CC} (pull-up)
AV_{CC}	Connect to V_{CC}
AV_{SS}, V_{REF}	Connect to V_{SS}

Note 1: With external clock input to X_{IN} pin.
Note 2: When the BCLK output disable bit (bit 7 at address 0004$_{16}$) is set to “1”, connect to V_{CC} via a resistor (pull-up).

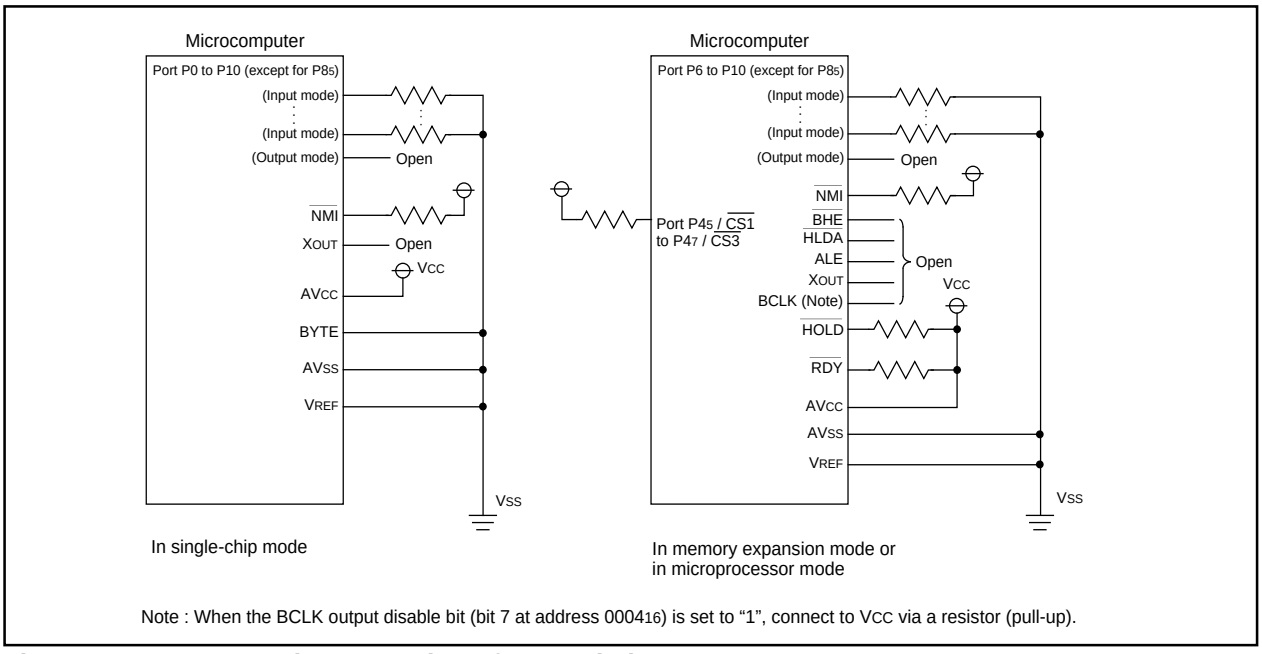


Figure 1.20.10. Example connection of unused pins

Items to be submitted when ordering masked ROM version

Please submit the following when ordering masked ROM products:

(1) Mask ROM confirmation form

(2) Mark specification sheet

(3) ROM data : Floppy disks *

*: 3.5-inch double-sided high-density disk (IBM format) is required per pattern.

Table 1.23.1. Absolute maximum ratings

Symbol	Parameter		Condition	Rated value	Unit
V_{CC}	Supply voltage		V_{CC}=AV$_{CC}$	-0.3 to 6.5	V
AVcc	Analog supply voltage		V_{CC}=AV$_{CC}$	-0.3 to 6.5	V
V_I	Input voltage	$\overline{RESET}$, CNV$_{SS}$, BYTE, P0$_0$ to P0$_7$, P1$_0$ to P1$_7$, P2$_0$ to P2$_7$, P3$_0$ to P3$_7$, P4$_0$ to P4$_7$, P5$_0$ to P5$_7$, P6$_0$ to P6$_7$, P7$_2$ to P7$_7$, P8$_0$ to P8$_7$, P9$_0$ to P9$_7$, P10$_0$ to P10$_7$, V_{REF}, X_{IN}		-0.3 to Vcc+0.3	V
		P7$_0$, P7$_1$		-0.3 to 6.5	V
V_O	Output voltage	P0$_0$ to P0$_7$, P1$_0$ to P1$_7$, P2$_0$ to P2$_7$, P3$_0$ to P3$_7$,P4$_0$ to P4$_7$, P5$_0$ to P5$_7$, P6$_0$ to P6$_7$,P7$_2$ to P7$_7$, P8$_0$ to P8$_4$, P8$_6$, P8$_7$, P9$_0$ to P9$_7$, P10$_0$ to P10$_7$, X_{OUT}		-0.3 to Vcc+0.3	V
		P7$_0$, P7$_1$		-0.3 to 6.5	V
P_d	Power dissipation		Topr=25°C	300	mW
T_{opr}	Operating ambient temperature			-20 to 85 / -40 to 85 (Note)	°C
T_{stg}	Storage temperature			-65 to 150	°C

Note : Specify a product of -40°C to 85°C to use it.

Table 1.23.2. Recommended operating conditions (referenced to V_{CC} = 2.7V to 5.5V at Topr = – 20°C to 85°C / – 40°C to 85°C (Note 3) unless otherwise specified)

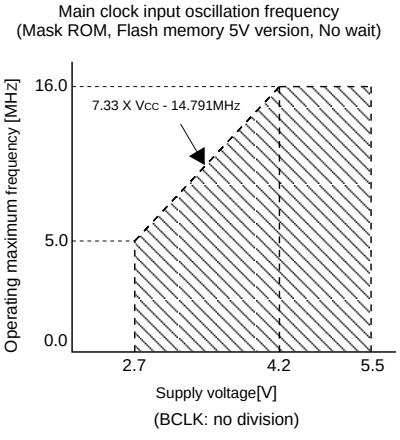
Symbol	Parameter				Standard Min	Standard Typ.	Standard Max.	Unit
Vcc	Supply voltage				2.7	5.0	5.5	V
AVcc	Analog supply voltage					Vcc		V
Vss	Supply voltage					0		V
AVss	Analog supply voltage					0		V
V_{IH}	HIGH input voltage	$P3_1$ to $P3_7$, $P4_0$ to $P4_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_2$ to $P7_7$, $P8_0$ to $P8_7$, $P9_0$ to $P9_7$, $P10_0$ to $P10_7$, X_{IN}, RESET, CNV_{SS}, BYTE			0.8Vcc		Vcc	V
		$P7_0$, $P7_1$			0.8Vcc		6.5	V
		$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ (during single-chip mode)			0.8Vcc		Vcc	V
		$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ (data input function during memory expansion and microprocessor modes)			0.5Vcc		Vcc	V
V_{IL}	LOW input voltage	$P3_1$ to $P3_7$, $P4_0$ to $P4_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_7$, $P9_0$ to $P9_7$, $P10_0$ to $P10_7$, X_{IN}, RESET, CNV_{SS}, BYTE			0		0.2Vcc	V
		$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ (during single-chip mode)			0		0.2Vcc	V
		$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ (data input function during memory expansion and microprocessor modes)			0		0.16Vcc	V
$I_{OH (peak)}$	HIGH peak output current	$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_2$ to $P7_7$, $P8_0$ to $P8_4$, $P8_6$, $P8_7$, $P9_0$ to $P9_7$, $P10_0$ to $P10_7$					-10.0	mA
$I_{OH (avg)}$	HIGH average output current	$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_2$ to $P7_7$, $P8_0$ to $P8_4$, $P8_6$, $P8_7$, $P9_0$ to $P9_7$, $P10_0$ to $P10_7$					-5.0	mA
$I_{OL (peak)}$	LOW peak output current	$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_4$, $P8_6$, $P8_7$, $P9_0$ to $P9_7$, $P10_0$ to $P10_7$					10.0	mA
$I_{OL (avg)}$	LOW average output current	$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_4$, $P8_6$, $P8_7$, $P9_0$ to $P9_7$, $P10_0$ to $P10_7$					5.0	mA
f (X_{IN})	Main clock input oscillation frequency	No wait	Mask ROM version, Flash memory 5V version (Note 5)	Vcc=4.2V to 5.5V	0		16	MHz
				Vcc=2.7V to 4.2V	0		7.33 X Vcc -14.791	MHz
		With wait	Mask ROM version, Flash memory 5V version (Note 5)	Vcc=4.2V to 5.5V	0		16	MHz
				Vcc=2.7V to 4.2V	0		4 X Vcc -0.8	MHz
f (X_{CIN})	Subclock oscillation frequency					32.768	50	kHz

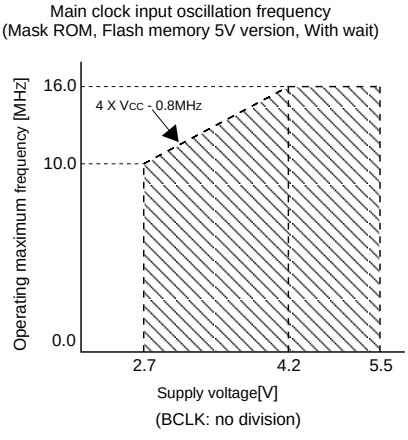
Note 1: The mean output current is the mean value within 100ms.

Note 2: The total I_{OL} (peak) for ports P0, P1, P2, $P8_6$, $P8_7$, P9, and P10 must be 80mA max. The total I_{OH} (peak) for ports P0, P1, P2, $P8_6$, $P8_7$, P9, and P10 must be 80mA max. The total I_{OL} (peak) for ports P3, P4, P5, P6, P7, and $P8_0$ to $P8_4$ must be 80mA max. The total I_{OH} (peak) for ports P3, P4, P5, P6, $P7_2$ to $P7_7$, and $P8_0$ to $P8_4$ must be 80mA max.

Note 3: Specify a product of -40°C to 85°C to use it.

Note 4: Relationship between main clock oscillation frequency and supply voltage.





Note 5: Execute case without wait, program / erase of flash memory by Vcc=4.2V to 5.5V and f(BCLK) ≤ 6.25 MHz. Execute case with wait, program / erase of flash memory by Vcc=4.2V to 5.5V and f(BCLK) ≤ 12.5 MHz.

Vcc = 5V

Table 1.23.3. A-D conversion characteristics (referenced to Vcc = AVcc = VREF = 2.7V to 5.5V, Vss = AVss = 0V at Topr = – 20°C to 85°C / – 40°C to 85°C (Note 4) unless otherwise specified)

Symbol	Parameter		Measuring condition		Standard			Unit
					Min.	Typ.	Max.	
–	Resolution		VREF = Vcc				10	Bits
–	Absolute accuracy	Sample & hold function not available	VREF = Vcc = 5V				±3	LSB
		Sample & hold function available(10bit)	VREF =Vcc = 5V	AN0 to AN7 input			±3	LSB
				ANEX0, ANEX1 input, External op-amp connection mode			±7	LSB
		Sample & hold function available(8bit)	VREF = Vcc = 5V				±2	LSB
		Sample & hold function not available(8bit)	VREF = Vcc = 3V, ØAD = fAD/2				±2	LSB
RLADDER	Ladder resistance		VREF = Vcc		10		40	kΩ
tCONV	Conversion time(10bit), Sample & hold function available		VREF = Vcc = 5V, ØAD =10MHz		3.3			µs
tCONV	Conversion time(8bit), Sample & hold function available		VREF = Vcc = 5V, ØAD =10MHz		2.8			µs
tCONV	Conversion time(8bit), Sample & hold function not available		VREF = Vcc = 3V, ØAD = fAD/2 = 5MHz		9.8			µs
tSAMP	Sampling time				0.3			µs
VREF	Reference voltage				2.7		Vcc	V
VIA	Analog input voltage				0		VREF	V

Note 1: Do f(XIN) in range of main clock input oscillation frequency prescribed with recommended operating conditions of table 1.23.2. Divide the f AD if f(XIN) exceeds 10MHz, and make AD operation clock frequency (ØAD) equal to or lower than 10MHz. And divide the f AD if Vcc is less than 4.2V, and make AD operation clock frequency (ØAD) equal to or lower than f AD/2.
Note 2: A case without sample & hold function turn AD operation clock frequency (ØAD) into 250 kHz or more in addition to a limit of Note 1.
A case with sample & hold function turn AD operation clock frequency (ØAD) into 1MHz or more in addition to a limit of Note 1.
Note 3: Connect AVcc pin to Vcc pin and apply the same electric potential.
Note 4: Specify a product of -40°C to 85°C to use it.

Table 1.23.4. D-A conversion characteristics (referenced to Vcc = VREF = 2.7V to 5.5V, Vss = AVss = 0V, at Topr = – 20°C to 85°C / – 40°C to 85°C (Note 2) unless otherwise specified)

Symbol	Parameter	Measuring condition	Standard			Unit
			Min.	Typ.	Max.	
–	Resolution				8	Bits
–	Absolute accuracy				1.0	%
tsu	Setup time				3	µs
RO	Output resistance		4	10	20	kΩ
IVREF	Reference power supply input current	(Note 1)			1.5	mA

Note 1: This applies when using one D-A converter, with the D-A register for the unused D-A converter set to "00$_{16}$".
The A-D converter's ladder resistance is not included.
Also, when D-A register contents are not "00 $_{16}$", the current IVREF always flows even though Vref may have been set to be unconnected by the A-D control register.
Note 2: Specify a product of -40°C to 85°C to use it.

Table 1.23.5. Flash memory version electrical characteristics (referenced to Vcc = 4.2V to 5.5V, at Topr =0 to 60°C unless otherwise specified)

Parameter	Standard			Unit
	Min.	Typ.	Max	
Page program time		6	120	ms
Block erase time		50	600	ms
Erase all unlocked blocks time		50 X n (Note)	600 X n (Note)	ms
Lock bit program time		6	120	ms

Note : n denotes the number of block erases.

VCC = 5V

Table 1.23.6. Electrical characteristics (referenced to VCC = 4.2V to 5.5V, VSS = 0V at Topr = – 20°C to 85°C / – 40°C to 85°C (Note 2), f(XIN) = 16MHz unless otherwise specified)

Symbol	Parameter		Measuring condition		Standard			Unit
					Min	Typ.	Max.	
V_{OH}	HIGH output voltage	$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_2$ to $P7_7$, $P8_0$ to $P8_4$, $P8_6$, $P8_7$, $P9_0$ to $P9_7$, $P10_0$ to $P10_7$	I_{OH} = -5mA		3.0			V
V_{OH}	HIGH output voltage	$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_2$ to $P7_7$, $P8_0$ to $P8_4$, $P8_6$, $P8_7$, $P9_0$ to $P9_7$, $P10_0$ to $P10_7$	I_{OH} = -200µA		4.7			V
V_{OH}	HIGH output voltage	X_{OUT}	HIGHPOWER	I_{OH} = -1mA	3.0			V
			LOWPOWER	I_{OH} = -0.5mA	3.0			
	HIGH output voltage	X_{COUT}	HIGHPOWER	With no load applied		3.0		V
			LOWPOWER	With no load applied		1.6		
V_{OL}	LOW output voltage	$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_4$, $P8_6$, $P8_7$, $P9_0$ to $P9_7$, $P10_0$ to $P10_7$	I_{OL} = 5mA				2.0	V
V_{OL}	LOW output voltage	$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_4$, $P8_6$, $P8_7$, $P9_0$ to $P9_7$, $P10_0$ to $P10_7$	I_{OL} = 200µA				0.45	V
V_{OL}	LOW output voltage	X_{OUT}	HIGHPOWER	I_{OL} = 1mA			2.0	V
			LOWPOWER	I_{OL} = 0.5mA			2.0	
	LOW output voltage	X_{COUT}	HIGHPOWER	With no load applied		0		V
			LOWPOWER	With no load applied		0		
V_{T+}-V_{T-}	Hysteresis	$\overline{HOLD}$, $\overline{RDY}$, $TA0_{IN}$ to $TA4_{IN}$, $TB0_{IN}$ to $TB5_{IN}$, $\overline{INT0}$ to $\overline{INT5}$, $\overline{NMI}$, $\overline{ADTRG}$, $\overline{CTS0}$ to $\overline{CTS2}$, SCL, SDA, $CLK0$ to $CLK4$, $TA2_{OUT}$ to $TA4_{OUT}$, $\overline{KI0}$ to $\overline{KI3}$, $RxD0$ to $RxD2$, $SIN3$, $SIN4$			0.2		1.0	V
V_{T+}-V_{T-}	Hysteresis	$\overline{RESET}$			0.2		1.8	V
I_{IH}	HIGH input current	$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_7$, $P9_0$ to $P9_7$, $P10_0$ to $P10_7$, X_{IN}, $\overline{RESET}$, CNVss, BYTE	V_I = 5V				5.0	µA
I_{IL}	LOW input current	$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_7$, $P9_0$ to $P9_7$, $P10_0$ to $P10_7$, X_{IN}, $\overline{RESET}$, CNVss, BYTE	V_I = 0V				-5.0	µA
R_{PULLUP}	Pull-up resistance	$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_2$ to $P7_7$, $P8_0$ to $P8_4$, $P8_6$, $P8_7$, $P9_0$ to $P9_7$, $P10_0$ to $P10_7$	V_I = 0V		30.0	50.0	167.0	kΩ
R_{fXIN}	Feedback resistance	X_{IN}				1.0		MΩ
R_{fXCIN}	Feedback resistance	X_{CIN}				6.0		MΩ
V_{RAM}	RAM retention voltage		When clock is stopped		2.0			V
Icc	Power supply current	In single-chip mode, the output pins are open and other pins are VSS	Mask ROM version	f(X_{IN}) = 16MHz Square wave, no division		30.0	50.0	mA
			Flash memory 5V version	f(X_{IN}) = 16MHz Square wave, no division		32.5	50.0	mA
			Mask ROM version	f(X_{CIN}) = 32kHz Square wave		90.0		µA
			Flash memory 5V version	f(X_{CIN}) = 32kHz Square wave, in RAM		90.0		µA
			Flash memory 5V version	f(X_{CIN}) = 32kHz Square wave, in flash memory		2.2		mA
			Flash memory 5V version, Program	f(X_{IN}) = 16MHz Square wave, Division by 4		25		mA
			Flash memory 5V version, Erase	f(X_{IN}) = 16MHz Square wave, Division by 4		28		mA
				f(XCIN) = 32kHz When a WAIT instruction is executed (Note 1)		4.0	10 (Topr = 25°C)	µA
				Topr = 25°C when clock is stopped			1.0	µA
				Topr = 85°C when clock is stopped			20.0	

Note 1: With one timer operated using f_{C32}.
Note 2: Specify a product of -40°C to 85°C to use it.

VCC = 5V

Timing requirements (referenced to VCC = 5V, VSS = 0V at Topr = – 20°C to 85°C / – 40°C to 85°C (*) unless otherwise specified)

* : Specify a product of -40°C to 85°C to use it.

Table 1.23.7. External clock input

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t_c	External clock input cycle time	62.5		ns
$t_{w(H)}$	External clock input HIGH pulse width	25		ns
$t_{w(L)}$	External clock input LOW pulse width	25		ns
t_r	External clock rise time		15	ns
t_f	External clock fall time		15	ns

Table 1.23.8. Memory expansion and microprocessor modes

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{ac1(RD-DB)}$	Data input access time (no wait)		(Note)	ns
$t_{ac2(RD-DB)}$	Data input access time (with wait)		(Note)	ns
$t_{ac3(RD-DB)}$	Data input access time (when accessing multiplex bus area)		(Note)	ns
$t_{su(DB-RD)}$	Data input setup time	40		ns
$t_{su(RDY-BCLK)}$	$\overline{RDY}$ input setup time	30		ns
$t_{su(HOLD-BCLK)}$	$\overline{HOLD}$ input setup time	40		ns
$t_{h(RD-DB)}$	Data input hold time	0		ns
$t_{h(BCLK-RDY)}$	$\overline{RDY}$ input hold time	0		ns
$t_{h(BCLK-HOLD)}$	$\overline{HOLD}$ input hold time	0		ns
$t_{d(BCLK-HLDA)}$	$\overline{HLDA}$ output delay time		40	ns

Note: Calculated according to the BCLK frequency as follows:

$$t_{ac1(RD-DB)} = \frac{10^9}{f(BCLK) \times 2} - 45 \quad [ns]$$

$$t_{ac2(RD-DB)} = \frac{3 \times 10^9}{f(BCLK) \times 2} - 45 \quad [ns]$$

$$t_{ac3(RD-DB)} = \frac{3 \times 10^9}{f(BCLK) \times 2} - 45 \quad [ns]$$

VCC = 5V

Timing requirements (referenced to VCC = 5V, VSS = 0V at Topr = – 20°C to 85°C / – 40°C to 85°C (*) unless otherwise specified)

* : Specify a product of -40°C to 85°C to use it.

Table 1.23.9. Timer A input (counter 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 pulse width	40		ns
$t_{w(TAL)}$	TAiIN input LOW pulse width	40		ns

Table 1.23.10. Timer A input (gating 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 pulse width	200		ns
$t_{w(TAL)}$	TAiIN input LOW pulse width	200		ns

Table 1.23.11. 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 pulse width	100		ns
$t_{w(TAL)}$	TAiIN input LOW pulse width	100		ns

Table 1.23.12. Timer A input (external trigger input in pulse width modulation mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{w(TAH)}$	TAiIN input HIGH pulse width	100		ns
$t_{w(TAL)}$	TAiIN input LOW pulse width	100		ns

Table 1.23.13. Timer A input (up/down 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 pulse width	1000		ns
$t_{w(UPL)}$	TAiOUT input LOW pulse width	1000		ns
$t_{su(UP-TIN)}$	TAiOUT input setup time	400		ns
$t_{h(TIN-UP)}$	TAiOUT input hold time	400		ns

VCC = 5V

Timing requirements (referenced to VCC = 5V, VSS = 0V at Topr = – 20°C to 85°C / – 40°C to 85°C (*) unless otherwise specified)

* : Specify a product of -40°C to 85°C to use it.

Table 1.23.14. Timer B input (counter input in event counter mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TB)}$	TBiIN input cycle time (counted on one edge)	100		ns
$t_{w(TBH)}$	TBiIN input HIGH pulse width (counted on one edge)	40		ns
$t_{w(TBL)}$	TBiIN input LOW pulse width (counted on one edge)	40		ns
$t_{c(TB)}$	TBiIN input cycle time (counted on both edges)	200		ns
$t_{w(TBH)}$	TBiIN input HIGH pulse width (counted on both edges)	80		ns
$t_{w(TBL)}$	TBiIN input LOW pulse width (counted on both edges)	80		ns

Table 1.23.15. Timer B input (pulse period measurement mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TB)}$	TBiIN input cycle time	400		ns
$t_{w(TBH)}$	TBiIN input HIGH pulse width	200		ns
$t_{w(TBL)}$	TBiIN input LOW pulse width	200		ns

Table 1.23.16. Timer B input (pulse width measurement mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TB)}$	TBiIN input cycle time	400		ns
$t_{w(TBH)}$	TBiIN input HIGH pulse width	200		ns
$t_{w(TBL)}$	TBiIN input LOW pulse width	200		ns

Table 1.23.17. A-D trigger input

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(AD)}$	$\overline{ADTRG}$ input cycle time (trigger able minimum)	1000		ns
$t_{w(ADL)}$	$\overline{ADTRG}$ input LOW pulse width	125		ns

Table 1.23.18. Serial I/O

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(CK)}$	CLKi input cycle time	200		ns
$t_{w(CKH)}$	CLKi input HIGH pulse width	100		ns
$t_{w(CKL)}$	CLKi input LOW pulse 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
$t_{h(C-D)}$	RxDi input hold time	90		ns

Table 1.23.19. External interrupt $\overline{INTi}$ inputs

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{w(INH)}$	$\overline{INTi}$ input HIGH pulse width	250		ns
$t_{w(INL)}$	$\overline{INTi}$ input LOW pulse width	250		ns

VCC = 5V

Switching characteristics (referenced to VCC = 5V, VSS = 0V at Topr = – 20°C to 85°C / – 40°C to 85°C (Note 3), CM15 = "1" unless otherwise specified)

Table 1.23.20. Memory expansion mode and microprocessor mode (no wait)

Symbol	Parameter	Measuring condition	Standard		Unit
			Min.	Max.	
$t_{d(BCLK-AD)}$	Address output delay time	Figure 1.23.1		25	ns
$t_{h(BCLK-AD)}$	Address output hold time (BCLK standard)		4		ns
$t_{h(RD-AD)}$	Address output hold time (RD standard)		0		ns
$t_{h(WR-AD)}$	Address output hold time (WR standard)		0		ns
$t_{d(BCLK-CS)}$	Chip select output delay time			25	ns
$t_{h(BCLK-CS)}$	Chip select output hold time (BCLK standard)		4		ns
$t_{d(BCLK-ALE)}$	ALE signal output delay time			25	ns
$t_{h(BCLK-ALE)}$	ALE signal output hold time		– 4		ns
$t_{d(BCLK-RD)}$	RD signal output delay time			25	ns
$t_{h(BCLK-RD)}$	RD signal output hold time		0		ns
$t_{d(BCLK-WR)}$	WR signal output delay time			25	ns
$t_{h(BCLK-WR)}$	WR signal output hold time		0		ns
$t_{d(BCLK-DB)}$	Data output delay time (BCLK standard)			40	ns
$t_{h(BCLK-DB)}$	Data output hold time (BCLK standard)		4		ns
$t_{d(DB-WR)}$	Data output delay time (WR standard)		(Note1)		ns
$t_{h(WR-DB)}$	Data output hold time (WR standard)(Note2)		0		ns

Note 1: Calculated according to the BCLK frequency as follows:

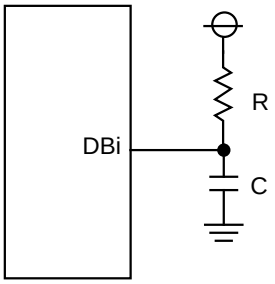
$$td(DB - WR) = \frac{10^9}{f(BCLK) \times 2} - 40 \quad [ns]$$

Note 2: This is standard value shows the timing when the output is off, and doesn't show hold time of data bus.
Hold time of data bus is different by capacitor volume and pull-up (pull-down) resistance value.
Hold time of data bus is expressed in

$$t = -CR \times \ln(1 - V_{OL} / V_{CC})$$

by a circuit of the right figure.
For example, when $V_{OL} = 0.2V_{CC}$, C = 30pF, R = 1kΩ, hold time of output "L" level is

$$t = -30pF \times 1k\Omega \times \ln(1 - 0.2V_{CC} / V_{CC})$$
$$= 6.7ns.$$



Note 3: Specify a product of -40°C to 85°C to use it.

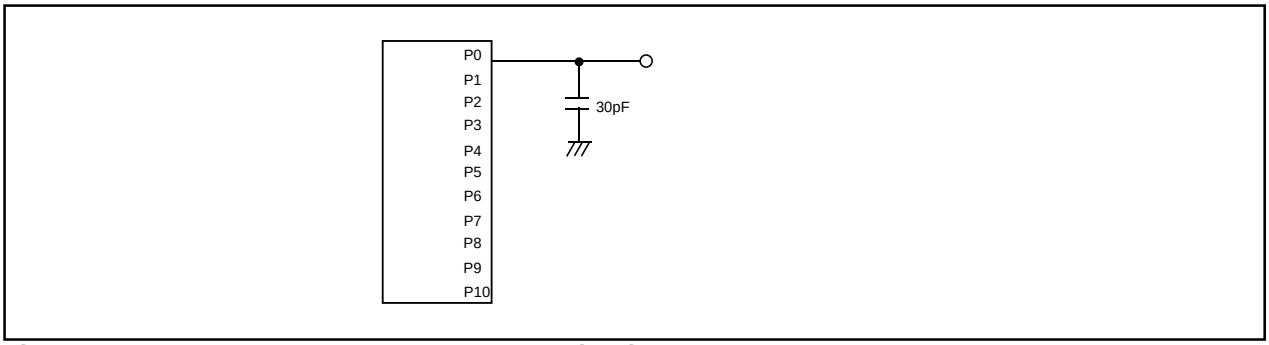


Figure 1.23.1. Port P0 to P10 measurement circuit

VCC = 5V

Switching characteristics (referenced to VCC = 5V, VSS = 0V at Topr = – 20°C to 85°C / – 40°C to 85°C (Note 3), CM15 = "1" unless otherwise specified)

Table 1.23.21. Memory expansion mode and microprocessor mode (with wait, accessing external memory)

Symbol	Parameter	Measuring condition	Standard		Unit
			Min.	Max.	
$t_{d(BCLK-AD)}$	Address output delay time	Figure 1.23.1		25	ns
$t_{h(BCLK-AD)}$	Address output hold time (BCLK standard)		4		ns
$t_{h(RD-AD)}$	Address output hold time (RD standard)		0		ns
$t_{h(WR-AD)}$	Address output hold time (WR standard)		0		ns
$t_{d(BCLK-CS)}$	Chip select output delay time			25	ns
$t_{h(BCLK-CS)}$	Chip select output hold time (BCLK standard)		4		ns
$t_{d(BCLK-ALE)}$	ALE signal output delay time			25	ns
$t_{h(BCLK-ALE)}$	ALE signal output hold time		– 4		ns
$t_{d(BCLK-RD)}$	RD signal output delay time			25	ns
$t_{h(BCLK-RD)}$	RD signal output hold time		0		ns
$t_{d(BCLK-WR)}$	WR signal output delay time			25	ns
$t_{h(BCLK-WR)}$	WR signal output hold time		0		ns
$t_{d(BCLK-DB)}$	Data output delay time (BCLK standard)			40	ns
$t_{h(BCLK-DB)}$	Data output hold time (BCLK standard)		4		ns
$t_{d(DB-WR)}$	Data output delay time (WR standard)		(Note1)		ns
$t_{h(WR-DB)}$	Data output hold time (WR standard)(Note2)		0		ns

Note 1: Calculated according to the BCLK frequency as follows:

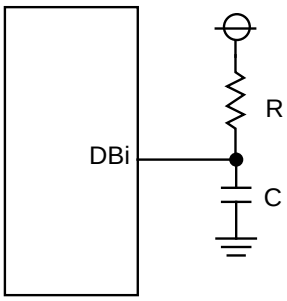
$$td(DB - WR) = \frac{10^9}{f(BCLK)} - 40 \quad [ns]$$

Note 2: This is standard value shows the timing when the output is off, and doesn't show hold time of data bus.
Hold time of data bus is different by capacitor volume and pull-up (pull-down) resistance value.
Hold time of data bus is expressed in

$t = -CR \times \ln(1 - V_{OL} / V_{CC})$

by a circuit of the right figure.
For example, when $V_{OL} = 0.2V_{CC}$, C = 30pF, R = 1kΩ, hold time of output "L" level is

$t = -30pF \times 1k\Omega \times \ln(1 - 0.2V_{CC} / V_{CC})$
$= 6.7ns.$



Note 3: Specify a product of -40°C to 85°C to use it.

VCC = 5V

Switching characteristics (referenced to VCC = 5V, VSS = 0V at Topr = – 20°C to 85°C / – 40°C to 85°C (Note 2), CM15 = "1" unless otherwise specified)

Table 1.23.22. Memory expansion mode and microprocessor mode (with wait, accessing external memory, multiplex bus area selected)

Symbol	Parameter	Measuring condition	Standard		Unit
			Min.	Max.	
$t_{d(BCLK-AD)}$	Address output delay time	Figure 1.23.1		25	ns
$t_{h(BCLK-AD)}$	Address output hold time (BCLK standard)		4		ns
$t_{h(RD-AD)}$	Address output hold time (RD standard)		(Note1)		ns
$t_{h(WR-AD)}$	Address output hold time (WR standard)		(Note1)		ns
$t_{d(BCLK-CS)}$	Chip select output delay time			25	ns
$t_{h(BCLK-CS)}$	Chip select output hold time (BCLK standard)		4		ns
$t_{h(RD-CS)}$	Chip select output hold time (RD standard)		(Note1)		ns
$t_{h(WR-CS)}$	Chip select output hold time (WR standard)		(Note1)		ns
$t_{d(BCLK-RD)}$	RD signal output delay time			25	ns
$t_{h(BCLK-RD)}$	RD signal output hold time		0		ns
$t_{d(BCLK-WR)}$	WR signal output delay time			25	ns
$t_{h(BCLK-WR)}$	WR signal output hold time		0		ns
$t_{d(BCLK-DB)}$	Data output delay time (BCLK standard)			40	ns
$t_{h(BCLK-DB)}$	Data output hold time (BCLK standard)		4		ns
$t_{d(DB-WR)}$	Data output delay time (WR standard)		(Note1)		ns
$t_{h(WR-DB)}$	Data output hold time (WR standard)		(Note1)		ns
$t_{d(BCLK-ALE)}$	ALE signal output delay time (BCLK standard)			25	ns
$t_{h(BCLK-ALE)}$	ALE signal output hold time (BCLK standard)		– 4		ns
$t_{d(AD-ALE)}$	ALE signal output delay time (Address standard)		(Note1)		ns
$t_{h(ALE-AD)}$	ALE signal output hold time (Adderss standard)		30		ns
$t_{d(AD-RD)}$	Post-address RD signal output delay time		0		ns
$t_{d(AD-WR)}$	Post-address WR signal output delay time		0		ns
$t_{dZ(RD-AD)}$	Address output floating start time			8	ns

Note 1: Calculated according to the BCLK frequency as follows:

$$th(RD - AD) = \frac{10^9}{f(BCLK) \times 2} \quad [ns]$$

$$th(WR - AD) = \frac{10^9}{f(BCLK) \times 2} \quad [ns]$$

$$th(RD - CS) = \frac{10^9}{f(BCLK) \times 2} \quad [ns]$$

$$th(WR - CS) = \frac{10^9}{f(BCLK) \times 2} \quad [ns]$$

$$td(DB - WR) = \frac{10^9 \times 3}{f(BCLK) \times 2} - 40 \quad [ns]$$

$$th(WR - DB) = \frac{10^9}{f(BCLK) \times 2} \quad [ns]$$

$$td(AD - ALE) = \frac{10^9}{f(BCLK) \times 2} - 25 \quad [ns]$$

Note 2: Specify a product of -40°C to 85°C to use it.

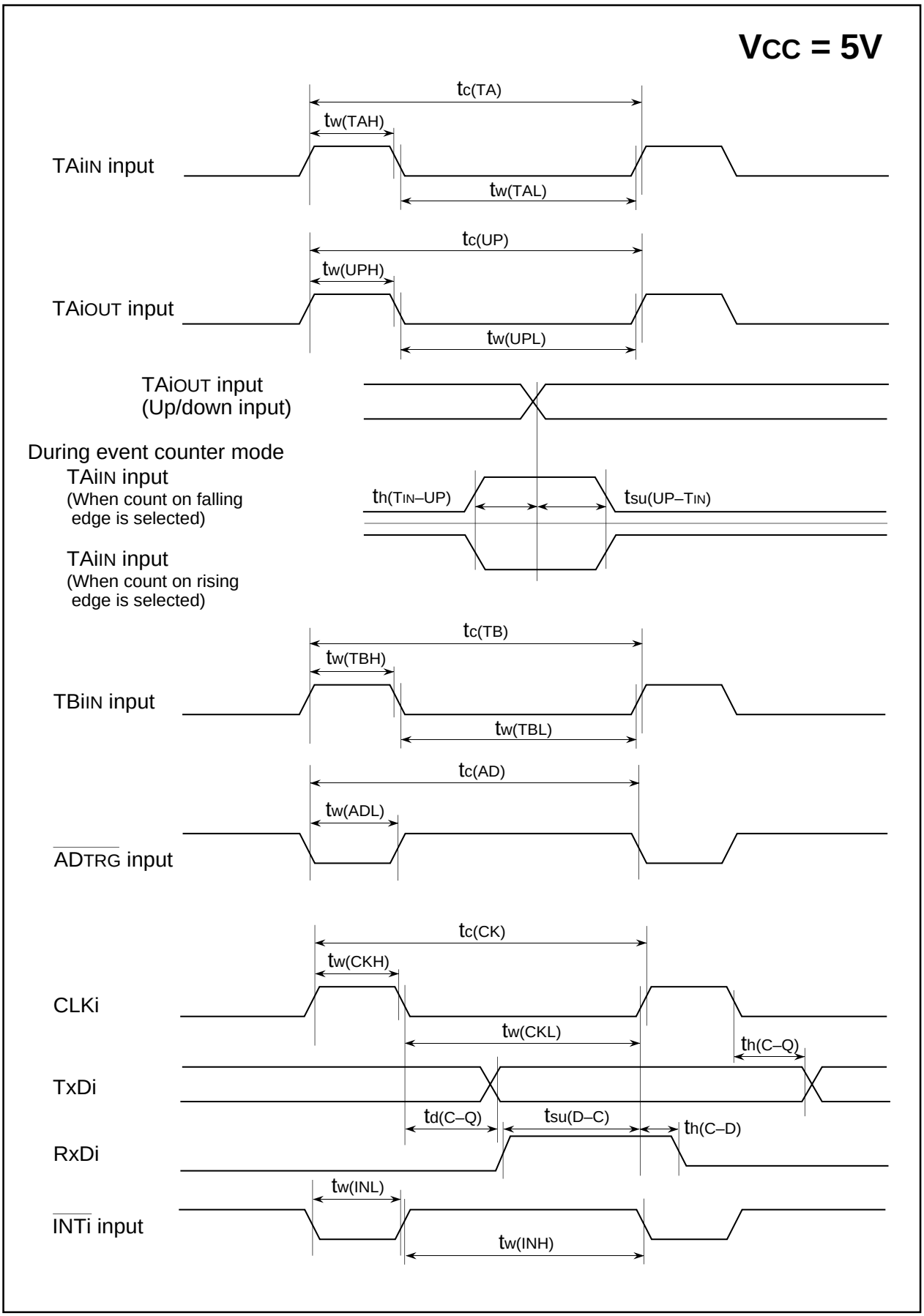
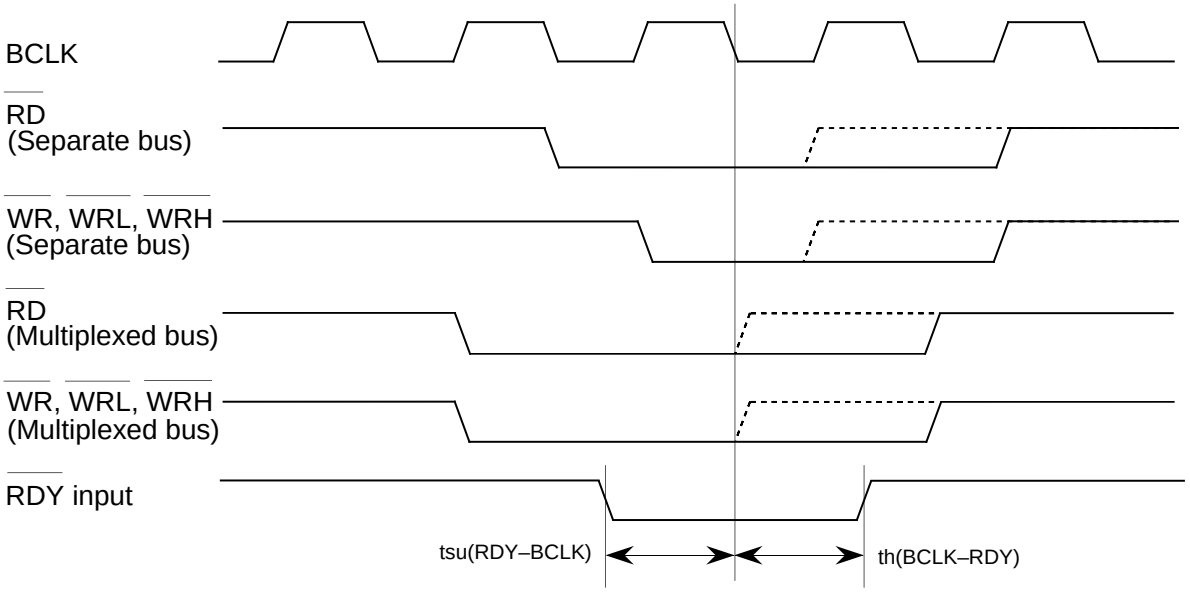


Figure 1.23.2. Vcc = 5V timing diagram (1)

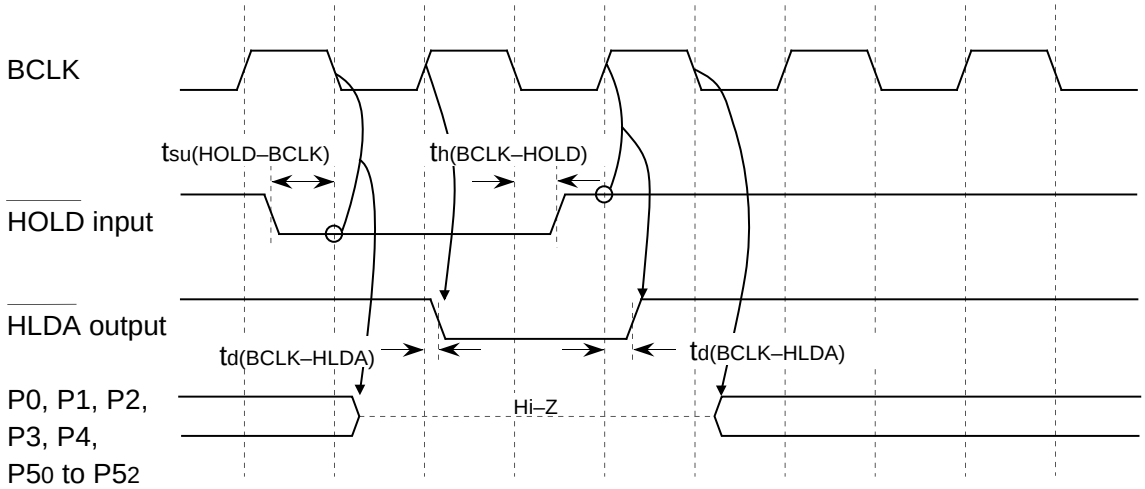
Vcc = 5V

Memory Expansion Mode and Microprocessor Mode

(Valid only with wait)



(Valid with or without wait)



Note: The above pins are set to high-impedance regardless of the input level of the BYTE pin and bit (PM06) of processor mode register 0 selects the function of ports P40 to P43.

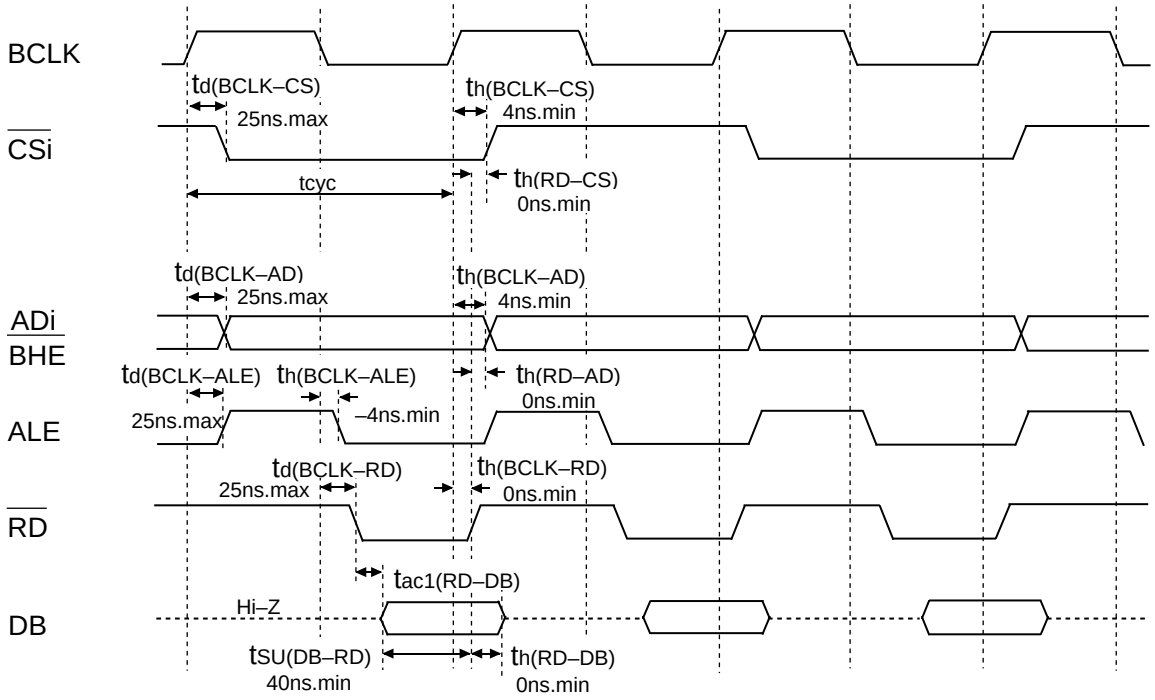
Measuring conditions :
- VCC=5V
- Input timing voltage : Determined with VIL=1.0V, VIH=4.0V
- Output timing voltage : Determined with VOL=2.5V, VOH=2.5V

Figure 1.23.3. Vcc = 5V timing diagram (2)

Memory Expansion Mode and Microprocessor Mode (With no wait)

Vcc = 5V

Read timing



Write timing

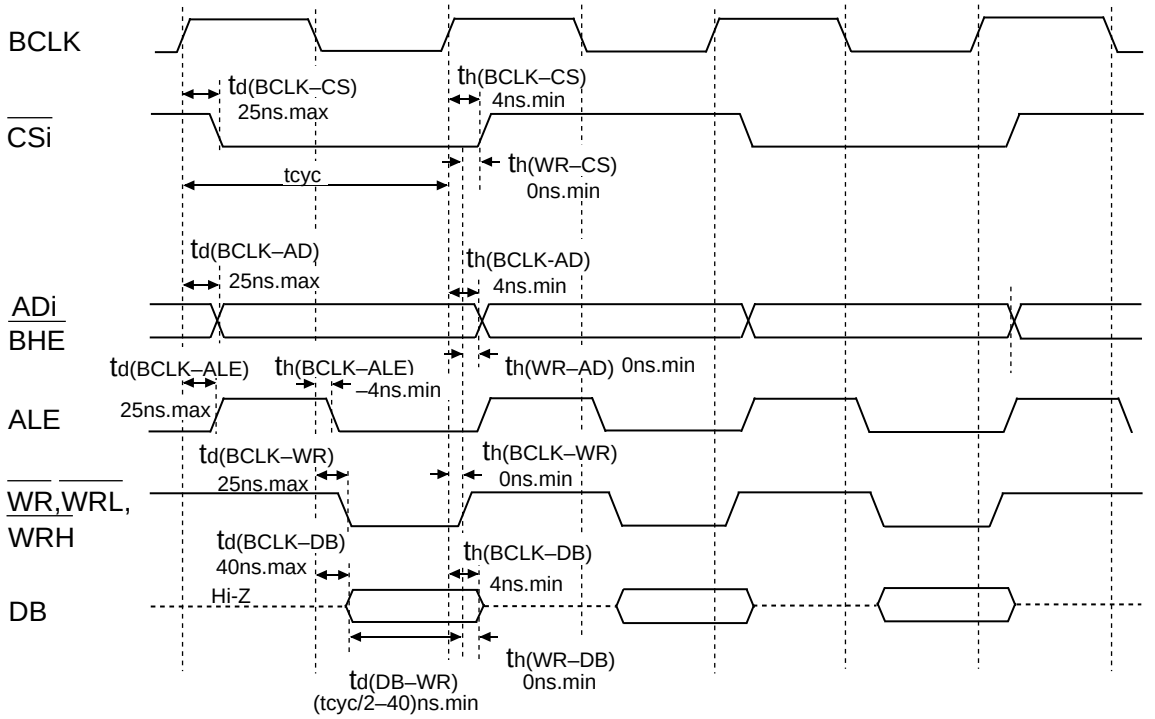


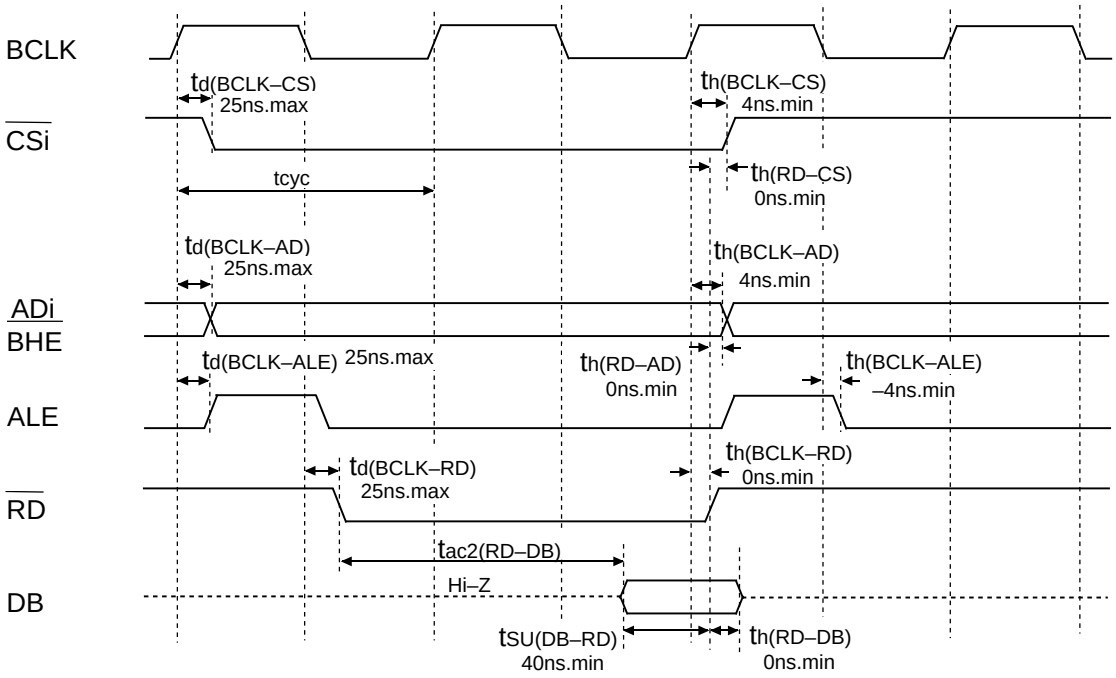
Figure 1.23.4. Vcc = 5V timing diagram (3)

Memory Expansion Mode and Microprocessor Mode

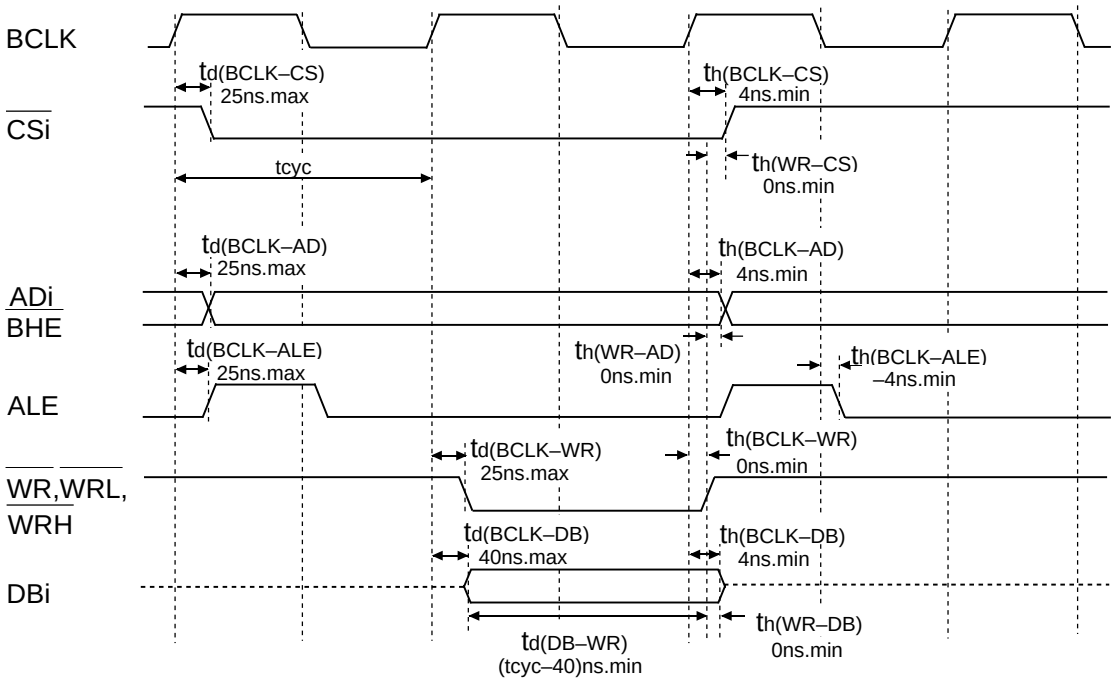
VCC = 5V

(When accessing external memory area with wait)

Read timing



Write timing



Measuring conditions :

- VCC=5V
- Input timing voltage : Determined with: VIL=0.8V, VIH=2.5V
- Output timing voltage : Determined with: VOL=0.8V, VOH=2.0V

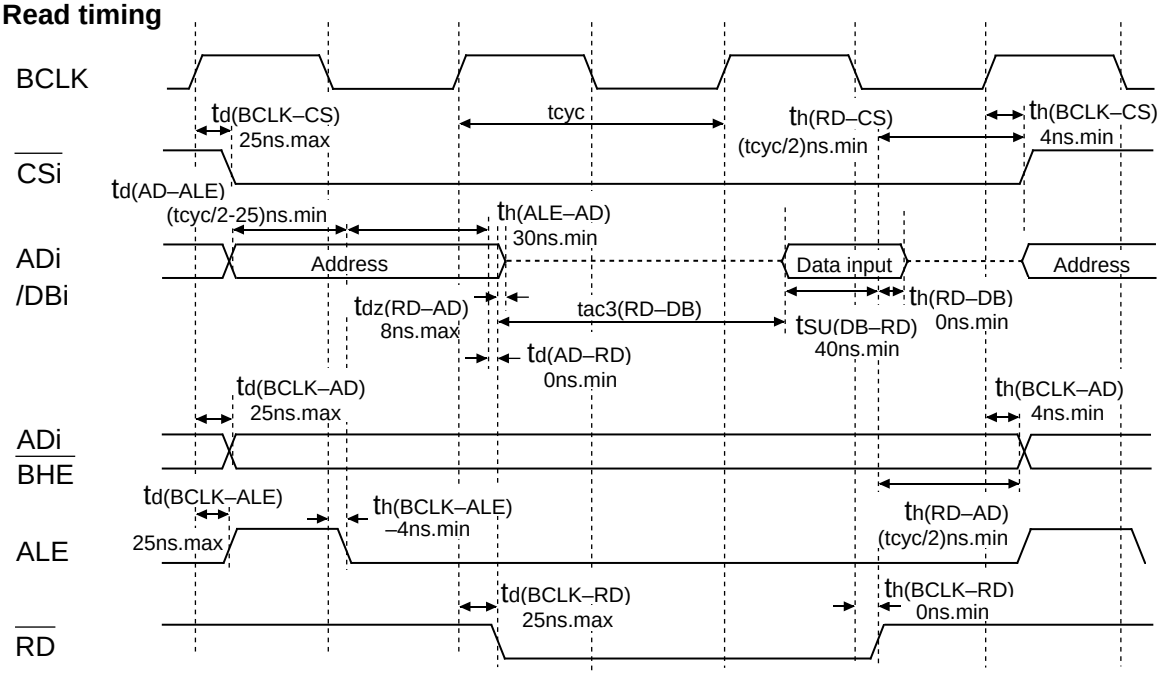
Figure 1.23.5. Vcc = 5V timing diagram (4)

Memory Expansion Mode and Microprocessor Mode

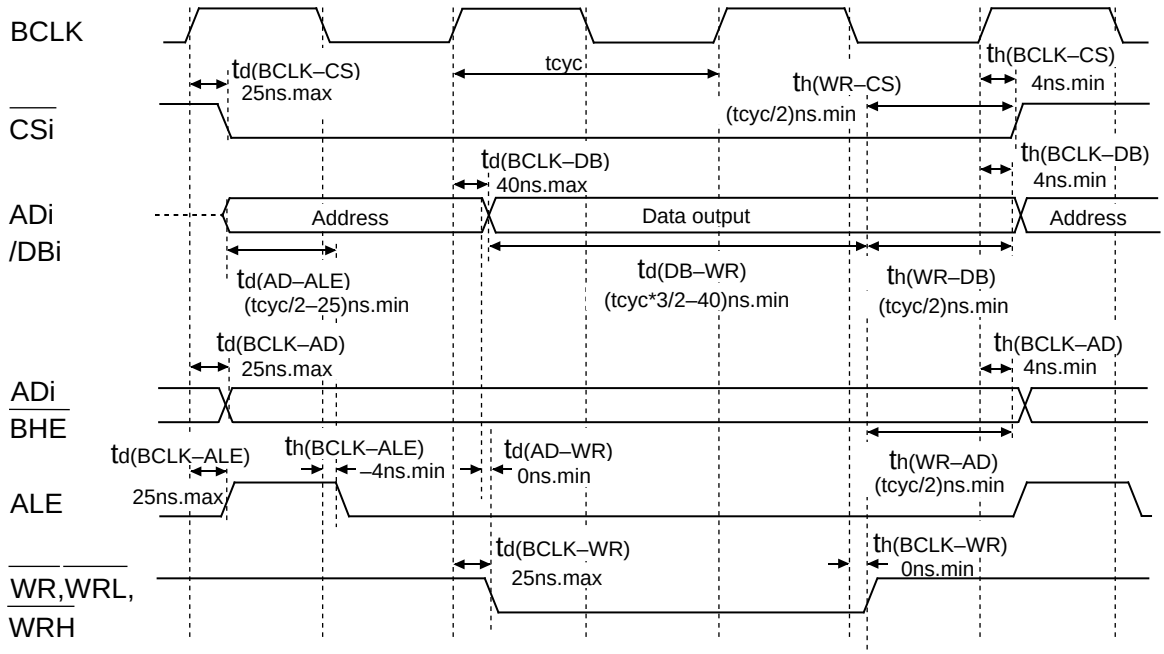
VCC = 5V

(When accessing external memory area with wait, and select multiplexed bus)

Read timing



Write timing



Measuring conditions :

- VCC=5V
- Input timing voltage : Determined with VIL=0.8V, VIH=2.5V
- Output timing voltage : Determined with VOL=0.8V, VOH=2.0V

Figure 1.23.6. Vcc = 5V timing diagram (5)

Vcc = 3V

Table 1.23.23. Electrical characteristics (referenced to Vcc = 2.7 to 3.3V, Vss = 0V at Topr = – 20°C to 85°C / – 40°C to 85°C (Note 1), f(XIN) = 10MHz (Note 2) with wait unless otherwise specified)

Symbol	Parameter		Measuring condition		Standard Min	Standard Typ.	Standard Max.	Unit
V_{OH}	HIGH output voltage $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_2$ to $P7_7$, $P8_0$ to $P8_4$, $P8_6$, $P8_7$, $P9_0$ to $P9_7$, $P10_0$ to $P10_7$		I_{OH} = -1mA		2.5			V
V_{OH}	HIGH output voltage X_{OUT}	HIGHPOWER	I_{OH} = -0.1mA		2.5			V
		LOWPOWER	I_{OH} = -50µA		2.5			
	HIGH output voltage X_{COUT}	HIGHPOWER	With no load applied			3.0		V
		LOWPOWER	With no load applied			1.6		
V_{OL}	LOW output voltage $P0_0$ to $P0_7$,$P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_4$, $P8_6$, $P8_7$, $P9_0$ to $P9_7$, $P10_0$ to $P10_7$		I_{OL} = 1mA				0.5	V
V_{OL}	LOW output voltage X_{OUT}	HIGHPOWER	I_{OL} = 0.1mA				0.5	V
		LOWPOWER	I_{OL} = 50µA				0.5	
	LOW output voltage X_{COUT}	HIGHPOWER	With no load applied			0		V
		LOWPOWER	With no load applied			0		
V_{T+}-V_{T-}	Hysteresis $\overline{HOLD}$, $\overline{RDY}$, $TA0_{IN}$ to $TA4_{IN}$, $TB0_{IN}$ to $TB5_{IN}$, $\overline{INT0}$ to $\overline{INT5}$, $\overline{NMI}$, $\overline{ADTRG}$, $\overline{CTS0}$ to $\overline{CTS2}$, SCL, SDA $CLK0$ to $CLK4$,$TA2_{OUT}$ to $TA4_{OUT}$, $\overline{KI0}$ to $\overline{KI3}$, $RxD0$ to $RxD2$, $SIN3$, $SIN4$				0.2		0.8	V
V_{T+}-V_{T-}	Hysteresis $\overline{RESET}$				0.2		1.8	V
I_{IH}	HIGH input current $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_7$,$P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_7$, $P9_0$ to $P9_7$, $P10_0$ to $P10_7$, X_{IN}, $\overline{RESET}$, CNVss, BYTE		V_I = 3V				4.0	µA
I_{IL}	LOW input current $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_7$, $P9_0$ to $P9_7$, $P10_0$ to $P10_7$, X_{IN}, $\overline{RESET}$, CNVss, BYTE		V_I = 0V				-4.0	µA
R_{PULLUP}	Pull-up resistance $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_2$ to $P7_7$, $P8_0$ to $P8_4$, $P8_6$, $P8_7$, $P9_0$ to $P9_7$,$P10_0$ to $P10_7$		V_I = 0V		66.0	120.0	500.0	kΩ
R_{fXIN}	Feedback resistance X_{IN}					3.0		MΩ
R_{fXCIN}	Feedback resistance X_{CIN}					10.0		MΩ
V_{RAM}	RAM retention voltage		When clock is stopped		2.0			V
Icc	Power supply current	In single-chip mode, the output pins are open and other pins are Vss	Mask ROM version	f(X_{IN}) = 10MHz Square wave, no division		8.5	21.25	mA
			Flash memory 5V version	f(X_{IN}) = 10MHz Square wave, no division		12.0	21.25	mA
			Mask ROM version	f(X_{CIN}) = 32kHz Square wave		40.0		µA
			Flash memory 5V version	f(X_{CIN}) = 32kHz Square wave, in RAM		40.0		µA
			Flash memory 5V version	f(X_{CIN}) = 32kHz Square wave, in flash memory		800		µA
				f(X_{CIN}) = 32kHz When a WAITinstruction is executed. Oscillation capacity High (Note 3)		2.8	10 (Topr = 25°C)	µA
				f(X_{CIN}) = 32kHz When a WAIT instruction is executed. Oscillation capacity Low (Note 3)		0.9	10 (Topr = 25°C)	µA
				Topr = 25°C when clock is stopped			1.0	µA
				Topr = 85°C when clock is stopped			20.0	

Note 1: Specify a product of -40°C to 85°C to use it.
Note 2: 10 MHz for the mask ROM version and flash memory 5V version.
Note 3: With one timer operated using f_{C32}.

VCC = 3V

Timing requirements (referenced to VCC = 3V, VSS = 0V at Topr = – 20°C to 85°C / – 40°C to 85°C (*) unless otherwise specified)

* : Specify a product of -40°C to 85°C to use it.

Table 1.23.24. External clock input

Symbol	Parameter		Standard		Unit
			Min.	Max.	
t_c	External clock input cycle time	Mask ROM, Flash memory 5V version	100		ns
$t_{w(H)}$	External clock input HIGH pulse width	Mask ROM, Flash memory 5V version	40		ns
$t_{w(L)}$	External clock input LOW pulse width	Mask ROM, Flash memory 5V version	40		ns
t_r	External clock rise time			18	ns
t_f	External clock fall time			18	ns

Table 1.23.25. Memory expansion and microprocessor modes

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{ac1(RD-DB)}$	Data input access time (no wait)		(Note)	ns
$t_{ac2(RD-DB)}$	Data input access time (with wait)		(Note)	ns
$t_{ac3(RD-DB)}$	Data input access time (when accessing multiplex bus area)		(Note)	ns
$t_{su(DB-RD)}$	Data input setup time	80		ns
$t_{su(RDY-BCLK)}$	$\overline{RDY}$ input setup time	60		ns
$t_{su(HOLD-BCLK)}$	$\overline{HOLD}$ input setup time	80		ns
$t_{h(RD-DB)}$	Data input hold time	0		ns
$t_{h(BCLK-RDY)}$	$\overline{RDY}$ input hold time	0		ns
$t_{h(BCLK-HOLD)}$	$\overline{HOLD}$ input hold time	0		ns
$t_{d(BCLK-HLDA)}$	$\overline{HLDA}$ output delay time		100	ns

Note: Calculated according to the BCLK frequency as follows:

$$t_{ac1(RD-DB)} = \frac{10^9}{f(BCLK) \times 2} - 90 \quad [ns]$$

$$t_{ac2(RD-DB)} = \frac{3 \times 10^9}{f(BCLK) \times 2} - 90 \quad [ns]$$

$$t_{ac3(RD-DB)} = \frac{3 \times 10^9}{f(BCLK) \times 2} - 90 \quad [ns]$$

VCC = 3V

Timing requirements (referenced to VCC = 3V, VSS = 0V at Topr = – 20°C to 85°C / – 40°C to 85°C (*) unless otherwise specified)

* : Specify a product of -40°C to 85°C to use it.

Table 1.23.26. Timer A input (counter input in event counter mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TA)}$	TAiIN input cycle time	150		ns
$t_{w(TAH)}$	TAiIN input HIGH pulse width	60		ns
$t_{w(TAL)}$	TAiIN input LOW pulse width	60		ns

Table 1.23.27. Timer A input (gating input in timer mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TA)}$	TAiIN input cycle time	600		ns
$t_{w(TAH)}$	TAiIN input HIGH pulse width	300		ns
$t_{w(TAL)}$	TAiIN input LOW pulse width	300		ns

Table 1.23.28. Timer A input (external trigger input in one-shot timer mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TA)}$	TAiIN input cycle time	300		ns
$t_{w(TAH)}$	TAiIN input HIGH pulse width	150		ns
$t_{w(TAL)}$	TAiIN input LOW pulse width	150		ns

Table 1.23.29. Timer A input (external trigger input in pulse width modulation mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{w(TAH)}$	TAiIN input HIGH pulse width	150		ns
$t_{w(TAL)}$	TAiIN input LOW pulse width	150		ns

Table 1.23.30. Timer A input (up/down input in event counter mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(UP)}$	TAiOUT input cycle time	3000		ns
$t_{w(UPH)}$	TAiOUT input HIGH pulse width	1500		ns
$t_{w(UPL)}$	TAiOUT input LOW pulse width	1500		ns
$t_{su(UP-TIN)}$	TAiOUT input setup time	600		ns
$t_{h(TIN-UP)}$	TAiOUT input hold time	600		ns

VCC = 3V

Timing requirements (referenced to VCC = 3V, VSS = 0V at Topr = – 20°C to 85°C / – 40°C to 85°C (*) unless otherwise specified)

* : Specify a product of -40°C to 85°C to use it.

Table 1.23.31. Timer B input (counter input in event counter mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TB)}$	TBiIN input cycle time (counted on one edge)	150		ns
$t_{w(TBH)}$	TBiIN input HIGH pulse width (counted on one edge)	60		ns
$t_{w(TBL)}$	TBiIN input LOW pulse width (counted on one edge)	60		ns
$t_{c(TB)}$	TBiIN input cycle time (counted on both edges)	300		ns
$t_{w(TBH)}$	TBiIN input HIGH pulse width (counted on both edges)	160		ns
$t_{w(TBL)}$	TBiIN input LOW pulse width (counted on both edges)	160		ns

Table 1.23.32. Timer B input (pulse period measurement mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TB)}$	TBiIN input cycle time	600		ns
$t_{w(TBH)}$	TBiIN input HIGH pulse width	300		ns
$t_{w(TBL)}$	TBiIN input LOW pulse width	300		ns

Table 1.23.33. Timer B input (pulse width measurement mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TB)}$	TBiIN input cycle time	600		ns
$t_{w(TBH)}$	TBiIN input HIGH pulse width	300		ns
$t_{w(TBL)}$	TBiIN input LOW pulse width	300		ns

Table 1.23.34. A-D trigger input

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(AD)}$	$\overline{ADTRG}$ input cycle time (trigger able minimum)	1500		ns
$t_{w(ADL)}$	$\overline{ADTRG}$ input LOW pulse width	200		ns

Table 1.23.35. Serial I/O

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(CK)}$	CLKi input cycle time	300		ns
$t_{w(CKH)}$	CLKi input HIGH pulse width	150		ns
$t_{w(CKL)}$	CLKi input LOW pulse width	150		ns
$t_{d(C-Q)}$	TxDi output delay time		160	ns
$t_{h(C-Q)}$	TxDi hold time	0		ns
$t_{su(D-C)}$	RxDi input setup time	50		ns
$t_{h(C-D)}$	RxDi input hold time	90		ns

Table 1.23.36. External interrupt $\overline{INTi}$ inputs

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{w(INH)}$	$\overline{INTi}$ input HIGH pulse width	380		ns
$t_{w(INL)}$	$\overline{INTi}$ input LOW pulse width	380		ns

VCC = 3V

Switching characteristics (referenced to VCC = 3V, VSS = 0V at Topr = – 20°C to 85°C / – 40°C to 85°C (Note 3), CM15="1" unless otherwise specified)

Table 1.23.37. Memory expansion and microprocessor modes (with no wait)

Symbol	Parameter	Measuring condition	Standard		Unit
			Min.	Max.	
$t_{d(BCLK-AD)}$	Address output delay time	Figure 1.23.7		60	ns
$t_{h(BCLK-AD)}$	Address output hold time (BCLK standard)		4		ns
$t_{h(RD-AD)}$	Address output hold time (RD standard)		0		ns
$t_{h(WR-AD)}$	Address output hold time (WR standard)		0		ns
$t_{d(BCLK-CS)}$	Chip select output delay time			60	ns
$t_{h(BCLK-CS)}$	Chip select output hold time (BCLK standard)		4		ns
$t_{d(BCLK-ALE)}$	ALE signal output delay time			60	ns
$t_{h(BCLK-ALE)}$	ALE signal output hold time		– 4		ns
$t_{d(BCLK-RD)}$	RD signal output delay time			60	ns
$t_{h(BCLK-RD)}$	RD signal output hold time		0		ns
$t_{d(BCLK-WR)}$	WR signal output delay time			60	ns
$t_{h(BCLK-WR)}$	WR signal output hold time		0		ns
$t_{d(BCLK-DB)}$	Data output delay time (BCLK standard)			80	ns
$t_{h(BCLK-DB)}$	Data output hold time (BCLK standard)		4		ns
$t_{d(DB-WR)}$	Data output delay time (WR standard)		(Note1)		ns
$t_{h(WR-DB)}$	Data output hold time (WR standard)(Note2)		0		ns

Note 1: Calculated according to the BCLK frequency as follows:

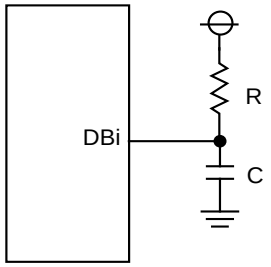
$$td(DB - WR) = \frac{10^9}{f(BCLK) \times 2} - 80 \quad [ns]$$

Note 2: This is standard value shows the timing when the output is off, and doesn't show hold time of data bus.
Hold time of data bus is different by capacitor volume and pull-up (pull-down) resistance value.
Hold time of data bus is expressed in

$$t = -CR \times \ln(1 - V_{OL} / V_{CC})$$

by a circuit of the right figure.
For example, when $V_{OL} = 0.2V_{CC}$, C = 30pF, R = 1kΩ, hold time of output "L" level is

$$t = -30pF \times 1k\Omega \times \ln(1 - 0.2V_{CC} / V_{CC}) = 6.7ns.$$



Note 3: Specify a product of -40°C to 85°C to use it.

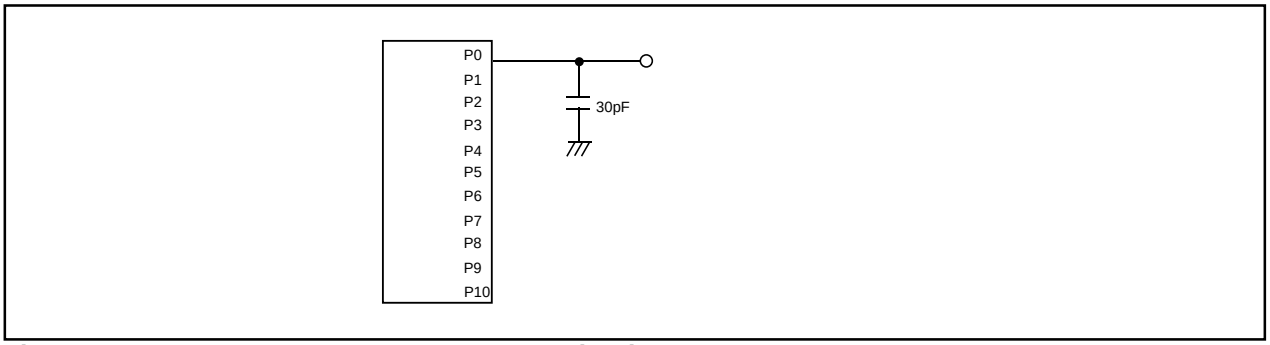


Figure 1.23.7. Port P0 to P10 measurement circuit

VCC = 3V

Switching characteristics (referenced to VCC = 3V, VSS = 0V at Topr = – 20°C to 85°C / – 40°C to 85°C (Note 3), CM15="1" unless otherwise specified)

Table 1.23.38. Memory expansion and microprocessor modes (when accessing external memory area with wait)

Symbol	Parameter	Measuring condition	Standard		Unit
			Min.	Max.	
$t_{d(BCLK-AD)}$	Address output delay time	Figure 1.23.7		60	ns
$t_{h(BCLK-AD)}$	Address output hold time (BCLK standard)		4		ns
$t_{h(RD-AD)}$	Address output hold time (RD standard)		0		ns
$t_{h(WR-AD)}$	Address output hold time (WR standard)		0		ns
$t_{d(BCLK-CS)}$	Chip select output delay time			60	ns
$t_{h(BCLK-CS)}$	Chip select output hold time (BCLK standard)		4		ns
$t_{d(BCLK-ALE)}$	ALE signal output delay time			60	ns
$t_{h(BCLK-ALE)}$	ALE signal output hold time		– 4		ns
$t_{d(BCLK-RD)}$	RD signal output delay time			60	ns
$t_{h(BCLK-RD)}$	RD signal output hold time		0		ns
$t_{d(BCLK-WR)}$	WR signal output delay time			60	ns
$t_{h(BCLK-WR)}$	WR signal output hold time		0		ns
$t_{d(BCLK-DB)}$	Data output delay time (BCLK standard)			80	ns
$t_{h(BCLK-DB)}$	Data output hold time (BCLK standard)		4		ns
$t_{d(DB-WR)}$	Data output delay time (WR standard)		(Note1)		ns
$t_{h(WR-DB)}$	Data output hold time (WR standard)(Note2)		0		ns

Note 1: Calculated according to the BCLK frequency as follows:

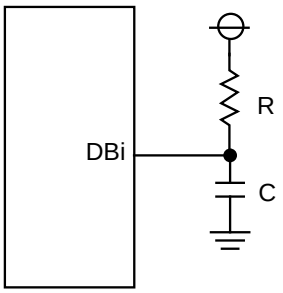
$$td(DB - WR) = \frac{10^9}{f(BCLK)} - 80 \quad [ns]$$

Note 2: This is standard value shows the timing when the output is off, and doesn't show hold time of data bus.
Hold time of data bus is different by capacitor volume and pull-up (pull-down) resistance value.
Hold time of data bus is expressed in

$t = -CR \times \ln(1 - V_{OL} / V_{CC})$

by a circuit of the right figure.
For example, when $V_{OL} = 0.2V_{CC}$, C = 30pF, R = 1kΩ, hold time of output "L" level is

$t = -30pF \times 1k\Omega \times \ln(1 - 0.2V_{CC} / V_{CC})$
$= 6.7ns.$



Note 3: Specify a product of -40°C to 85°C to use it.

VCC = 3V

Switching characteristics (referenced to VCC = 3V, VSS = 0V at Topr = – 20°C to 85°C / – 40°C to 85°C (Note 2), CM15="1" unless otherwise specified)

Table 1.23.39. Memory expansion and microprocessor modes (when accessing external memory area with wait, and select multiplexed bus)

Symbol	Parameter	Measuring condition	Standard		Unit
			Min.	Max.	
$t_{d(BCLK-AD)}$	Address output delay time	Figure 1.23.7		60	ns
$t_{h(BCLK-AD)}$	Address output hold time (BCLK standard)		4		ns
$t_{h(RD-AD)}$	Address output hold time (RD standard)		(Note1)		ns
$t_{h(WR-AD)}$	Address output hold time (WR standard)		(Note1)		ns
$t_{d(BCLK-CS)}$	Chip select output delay time			60	ns
$t_{h(BCLK-CS)}$	Chip select output hold time (BCLK standard)		4		ns
$t_{h(RD-CS)}$	Chip select output hold time (RD standard)		(Note1)		ns
$t_{h(WR-CS)}$	Chip select output hold time (WR standard)		(Note1)		ns
$t_{d(BCLK-RD)}$	RD signal output delay time			60	ns
$t_{h(BCLK-RD)}$	RD signal output hold time		0		ns
$t_{d(BCLK-WR)}$	WR signal output delay time			60	ns
$t_{h(BCLK-WR)}$	WR signal output hold time		0		ns
$t_{d(BCLK-DB)}$	Data output delay time (BCLK standard)			80	ns
$t_{h(BCLK-DB)}$	Data output hold time (BCLK standard)		4		ns
$t_{d(DB-WR)}$	Data output delay time (WR standard)		(Note1)		ns
$t_{h(WR-DB)}$	Data output hold time (WR standard)		(Note1)		ns
$t_{d(BCLK-ALE)}$	ALE signal output delay time (BCLK standard)			60	ns
$t_{h(BCLK-ALE)}$	ALE signal output hold time (BCLK standard)		– 4		ns
$t_{d(AD-ALE)}$	ALE signal output delay time (Address standard)		(Note1)		ns
$t_{h(ALE-AD)}$	ALE signal output hold time(Address standard)		50		ns
$t_{d(AD-RD)}$	Post-address RD signal output delay time		0		ns
$t_{d(AD-WR)}$	Post-address WR signal output delay time		0		ns
$t_{dZ(RD-AD)}$	Address output floating start time			8	ns

Note: Calculated according to the BCLK frequency as follows:

$$th(RD - AD) = \frac{10^9}{f(BCLK) \times 2} \quad [ns]$$

$$th(WR - AD) = \frac{10^9}{f(BCLK) \times 2} \quad [ns]$$

$$th(RD - CS) = \frac{10^9}{f(BCLK) \times 2} \quad [ns]$$

$$th(WR - CS) = \frac{10^9}{f(BCLK) \times 2} \quad [ns]$$

$$td(DB - WR) = \frac{10^9 \times 3}{f(BCLK) \times 2} - 80 \quad [ns]$$

$$th(WR - DB) = \frac{10^9}{f(BCLK) \times 2} \quad [ns]$$

$$td(AD - ALE) = \frac{10^9}{f(BCLK) \times 2} - 45 \quad [ns]$$

Note 2: Specify a product of -40°C to 85°C to use it.

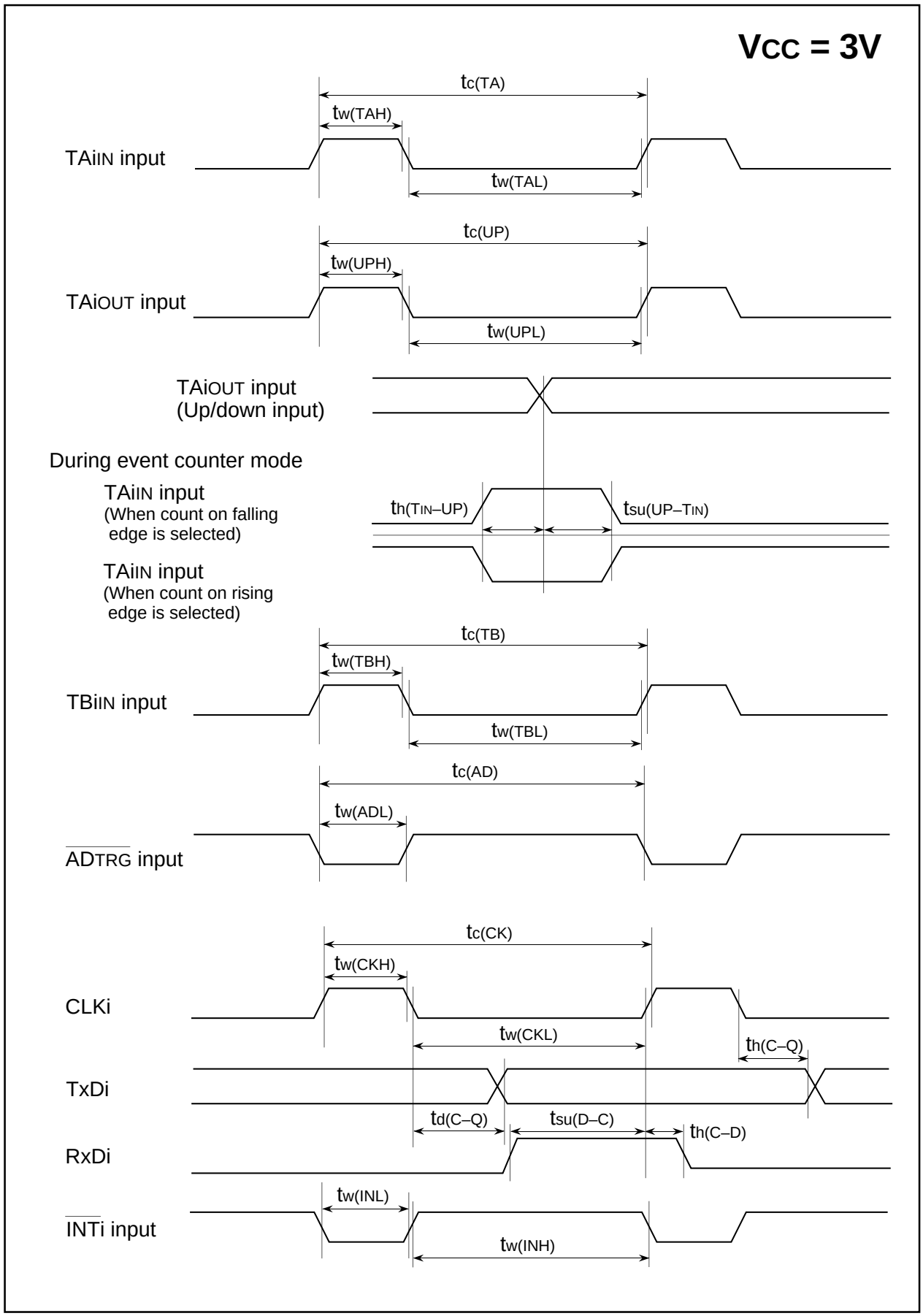
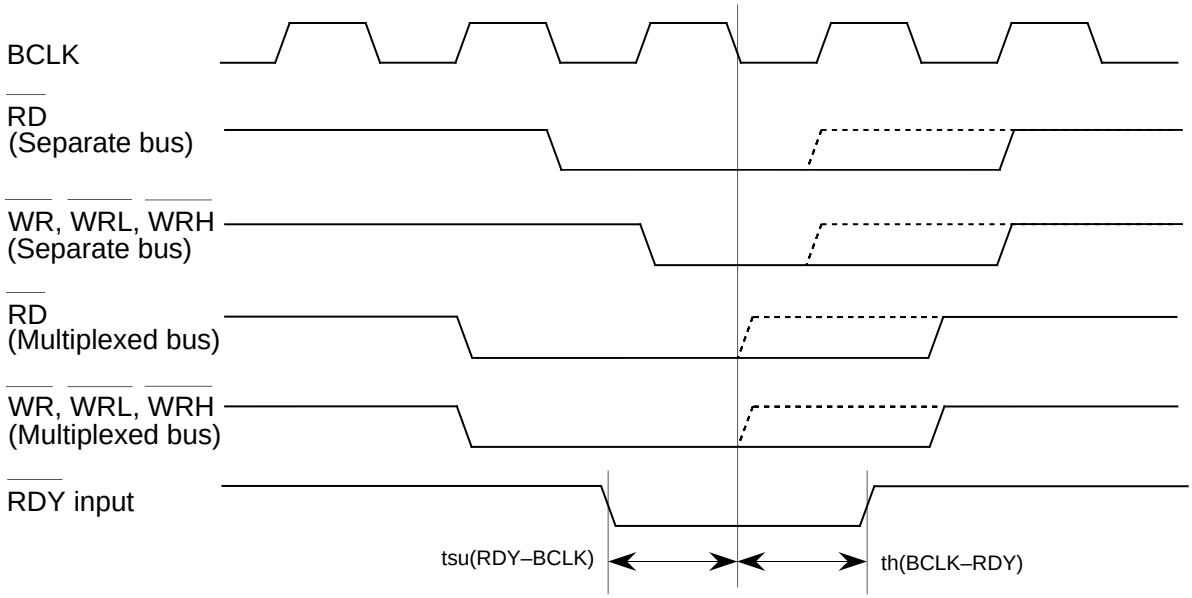


Figure 1.23.8. Vcc = 3V timing diagram (1)

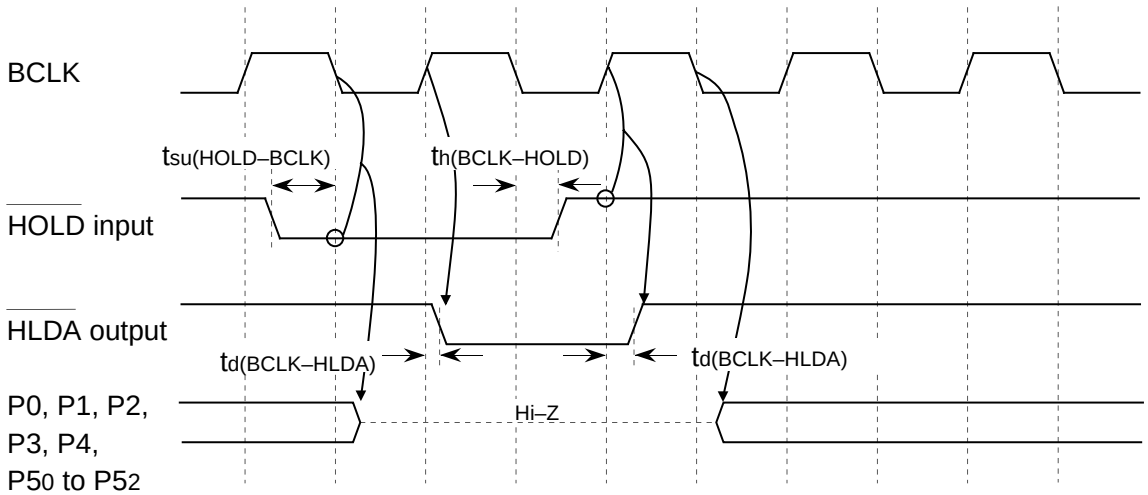
Vcc = 3V

Memory Expansion Mode and Microprocessor Mode

(Valid only with wait)



(Valid with or without wait)



Note: The above pins are set to high-impedance regardless of the input level of the BYTE pin and bit (PM06) of processor mode register 0 selects the function of ports P40 to P43.

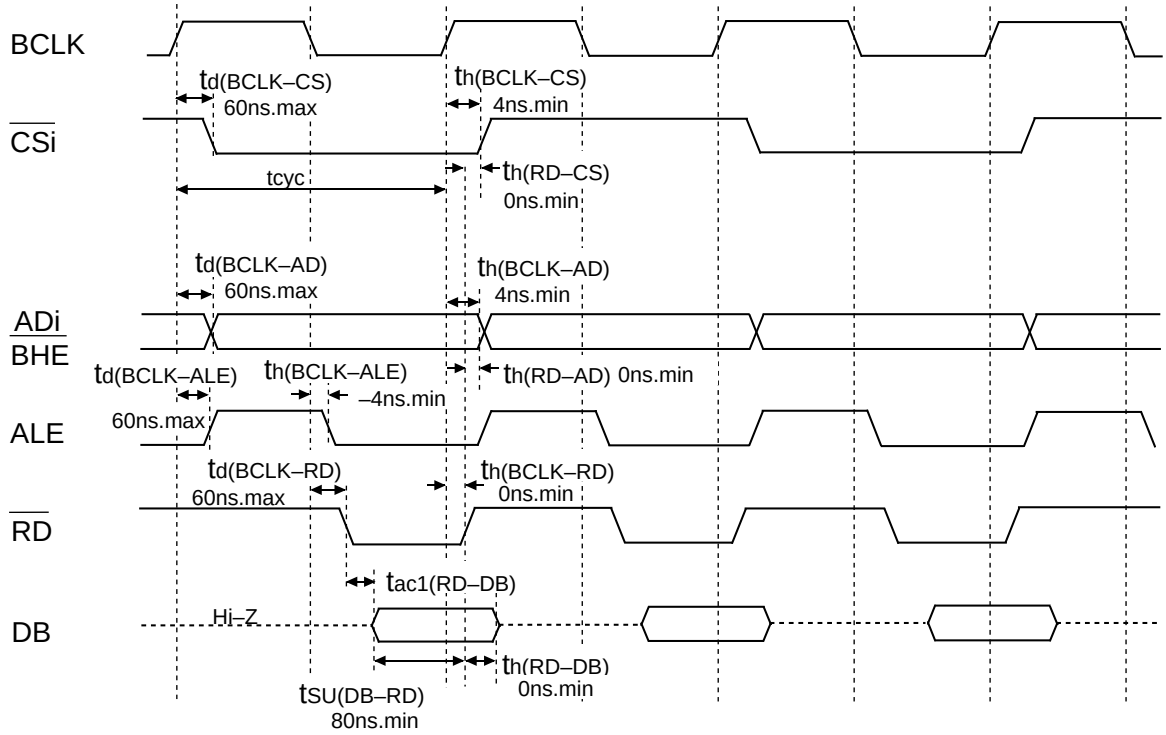
Measuring conditions :
- VCC=3V
- Input timing voltage : Determined with VIL=0.6V, VIH=2.4V
- Output timing voltage : Determined with VOL=1.5V, VOH=1.5V

Figure 1.23.9. Vcc = 3V timing diagram (2)

Memory Expansion Mode and Microprocessor Mode
(With no wait)

Vcc = 3V

Read timing



Write timing

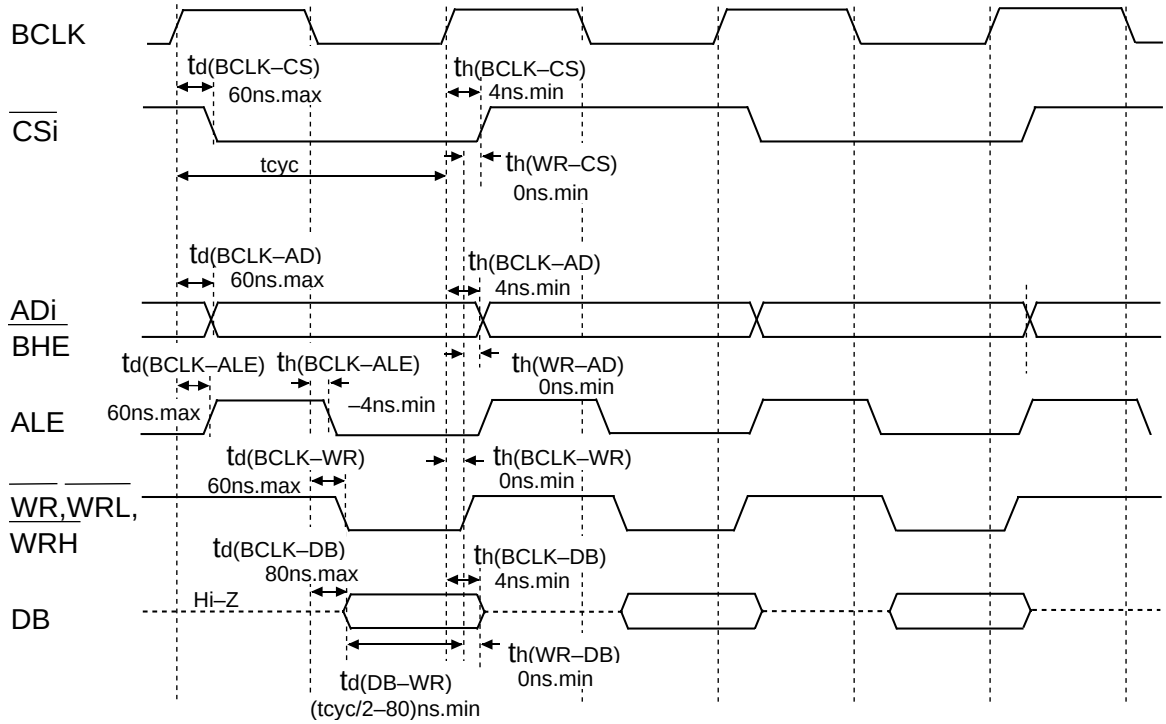


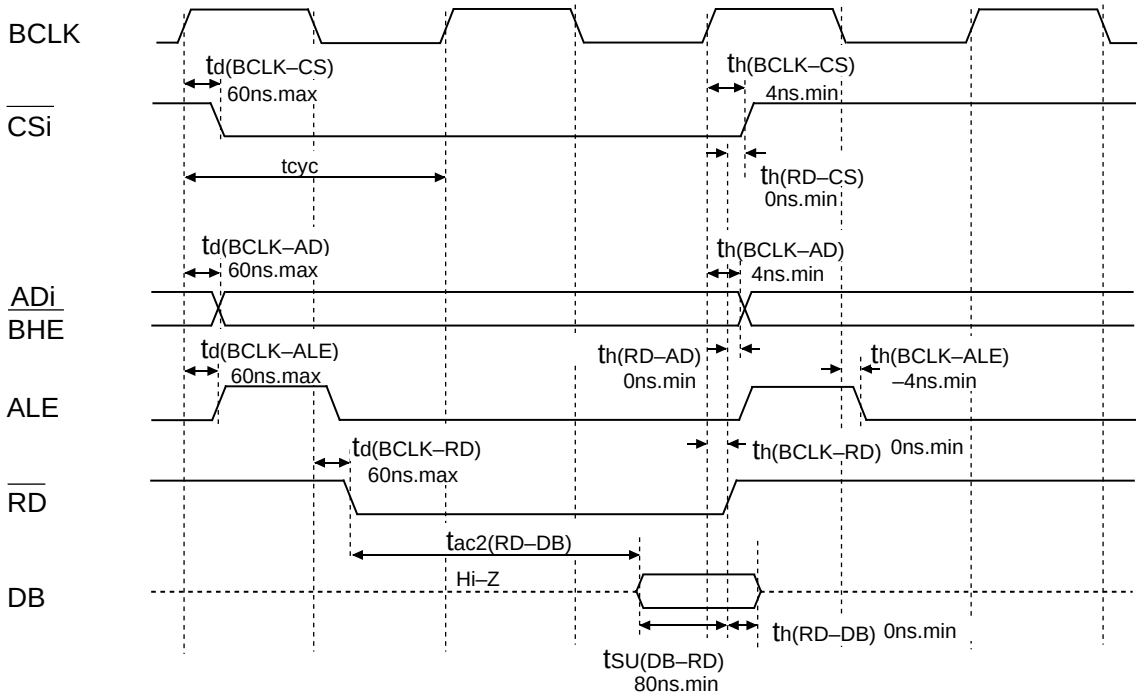
Figure 1.23.10. Vcc = 3V timing diagram (3)

Memory Expansion Mode and Microprocessor Mode

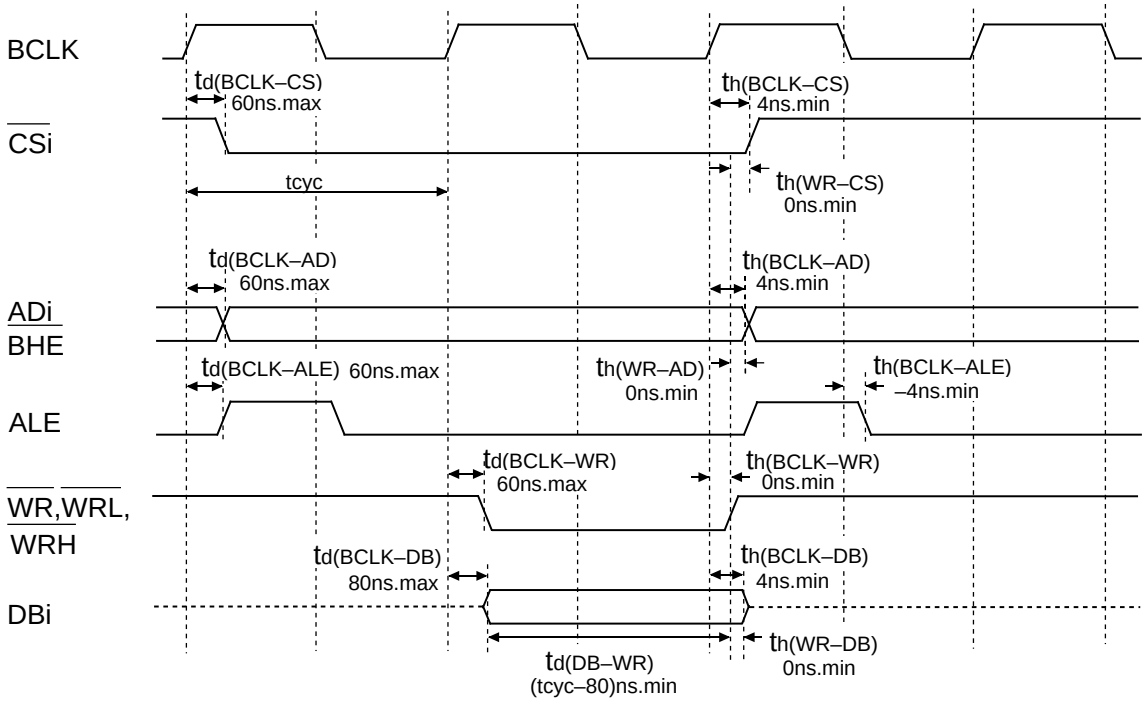
Vcc = 3V

(When accessing external memory area with wait)

Read timing



Write timing



Measuring conditions :

- Vcc=3V
- Input timing voltage : Determined with VIL=0.48V, VIH=1.5V
- Output timing voltage : Determined with VOL=1.5V, VOH=1.5V

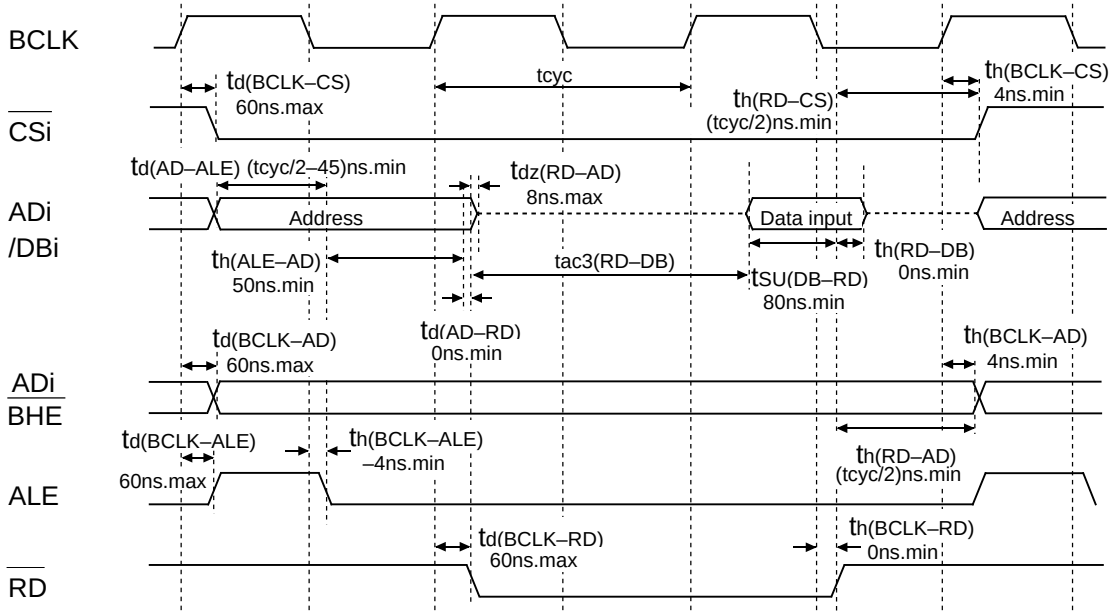
Figure 1.23.11. Vcc = 3V timing diagram (4)

Vcc = 3V

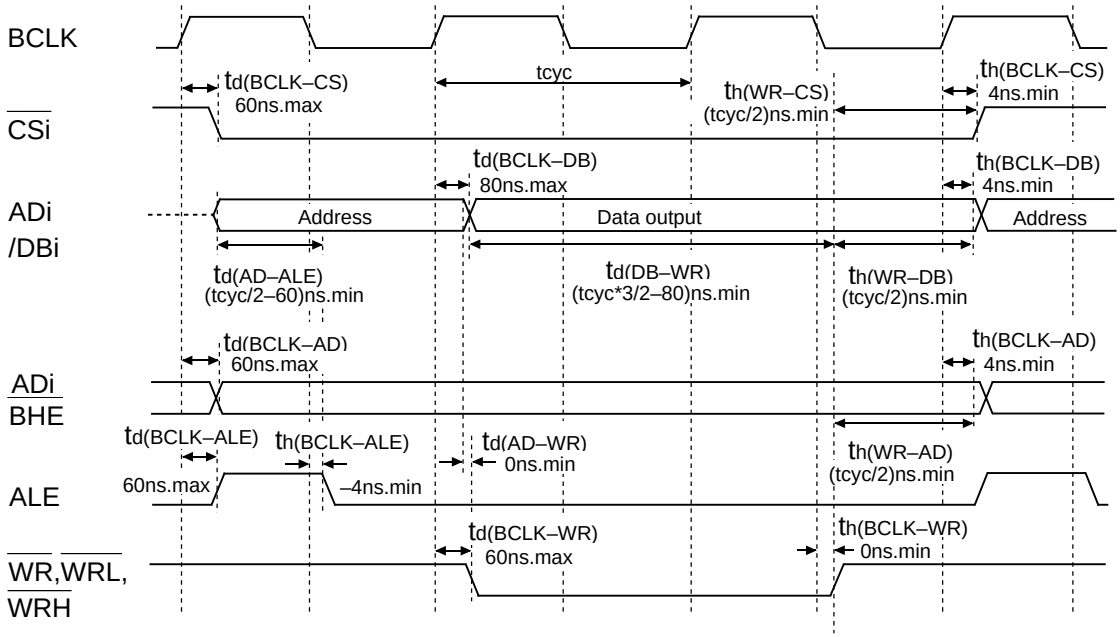
Memory Expansion Mode and Microprocessor Mode

(When accessing external memory area with wait, and select multiplexed bus)

Read timing



Write timing



Measuring conditions :

- Vcc=3V
- Input timing voltage : Determined with VIL=0.48V,VIH=1.5V
- Output timing voltage : Determined with VOL=1.5V,VOH=1.5V

Figure 1.23.12. Vcc = 3V timing diagram (5)

GZZ-SH13-36B<96A0>

Mask ROM number	

MITSUBISHI ELECTRIC-CHIP 16-BIT MICROCOMPUTER M30620M8A-XXXFP/GP MASK ROM CONFIRMATION FORM

Receipt	Date :	
	Section head signature	Supervisor signature

Note : Please complete all items marked ✵ .

✵	Customer	Company name	TEL ( )	Issuance signature	Submitted by	Supervisor
		Date issued	Date :			

✵1. Check sheet

Mitsubishi processes the mask files generated by the mask file generation utilities out of those held on the floppy disks you give in to us, and forms them into masks. Hence, we assume liability provided that there is any discrepancy between the contents of these mask files and the ROM data to be burned into products we produce. Check thoroughly the contents of the mask files you give in.
Prepare 3.5 inches 2HD (IBM format) floppy disks. And store only one mask file in a floppy disk.

Microcomputer type No. : ☐ M30620M8A-XXXFP ☐ M30620M8A-XXXGP

File code : ☐☐☐☐☐☐☐☐ (hex)

Mask file name : ☐☐☐☐☐☐☐☐ .MSK (alpha-numeric 8-digit)

✵2. Mark specification

The mark specification differs according to the type of package. After entering the mark specification on the separate mark specification sheet (for each package), attach that sheet to this masking check sheet for submission to Mitsubishi.

For the M30620M8A-XXXFP, submit the 100P6S mark specification sheet. For the M30620M8A-XXXGP, submit the 100P6Q mark specification sheet.

✵3. Usage Conditions

For our reference when of testing our products, please reply to the following questions about the usage of the products you ordered.

(1) Which kind of XIN-XOUT oscillation circuit is used?

☐ Ceramic resonator ☐ Quartz-crystal oscillator
☐ External clock input ☐ Other ( )

What frequency do not use?

f(XIN) = ☐ MHz

GZZ-SH13-36B<96A0>

Mask ROM number	

MITSUBISHI ELECTRIC-CHIP 16-BIT MICROCOMPUTER M30620M8A-XXXFP/GP MASK ROM CONFIRMATION FORM

(2) Which kind of XCIN-XCOUT oscillation circuit is used?

☐ Ceramic resonator ☐ Quartz-crystal oscillator

☐ External clock input ☐ Other ( )

What frequency do not use?

f(XCIN) = ☐ kHz

(3) Which operation mode do you use?

☐ Single-chip mode ☐ Memory expansion mode

☐ Microprocessor mode

(4) Which operating supply voltage do you use?

(Circle the operating voltage range of use)

2.4 2.7 3.0 3.3 3.5 3.8 4.0 4.2 4.5 4.7 5.0 5.3 5.5 5.7 (V)

(5) Which operating ambient temperature do you use?

(Circle the operating temperature range of use)

-50 -40 -30 -20 -10 0 10 20 30 40 50 60 70 80 90 (°C)

(6) Do you use I^2C (Inter IC) bus function?

☐ Not use ☐ Use

(7) Do you use IE (Inter Equipment) bus function?

☐ Not use ☐ Use

Thank you cooperation.

❋4. Special item (Indicate none if there is not specified item)

GZZ-SH13-37B<96A0>

Mask ROM number	

MITSUBISHI ELECTRIC-CHIP 16-BIT MICROCOMPUTER M30620MAA-XXXFP/GP MASK ROM CONFIRMATION FORM

Receipt	Date :	
	Section head signature	Supervisor signature

Note : Please complete all items marked ✵ .

✵	Customer	Company name	TEL ( )	Issuance signature	Submitted by	Supervisor
		Date issued	Date :			

✵1. Check sheet

Mitsubishi processes the mask files generated by the mask file generation utilities out of those held on the floppy disks you give in to us, and forms them into masks. Hence, we assume liability provided that there is any discrepancy between the contents of these mask files and the ROM data to be burned into products we produce. Check thoroughly the contents of the mask files you give in.
Prepare 3.5 inches 2HD (IBM format) floppy disks. And store only one mask file in a floppy disk.

Microcomputer type No. : ☐ M30620MAA-XXXFP ☐ M30620MAA-XXXGP

File code : ☐☐☐☐☐☐☐☐ (hex)

Mask file name : ☐☐☐☐☐☐☐☐ .MSK (alpha-numeric 8-digit)

✵2. Mark specification

The mark specification differs according to the type of package. After entering the mark specification on the separate mark specification sheet (for each package), attach that sheet to this masking check sheet for submission to Mitsubishi.

For the M30620MAA-XXXFP, submit the 100P6S mark specification sheet. For the M30620MAA-XXXGP, submit the 100P6Q mark specification sheet.

✵3. Usage Conditions

For our reference when of testing our products, please reply to the following questions about the usage of the products you ordered.

(1) Which kind of XIN-XOUT oscillation circuit is used?

☐ Ceramic resonator ☐ Quartz-crystal oscillator
☐ External clock input ☐ Other ( )

What frequency do not use?

f(XIN) = ☐ MHz

GZZ-SH13-37B<96A0>

Mask ROM number	

MITSUBISHI ELECTRIC-CHIP 16-BIT MICROCOMPUTER M30620MAA-XXXFP/GP MASK ROM CONFIRMATION FORM

(2) Which kind of XCIN-XCOUT oscillation circuit is used?

☐ Ceramic resonator ☐ Quartz-crystal oscillator

☐ External clock input ☐ Other ( )

What frequency do not use?

f(XCIN) = ☐ kHz

(3) Which operation mode do you use?

☐ Single-chip mode ☐ Memory expansion mode

☐ Microprocessor mode

(4) Which operating supply voltage do you use?

(Circle the operating voltage range of use)

2.4 2.7 3.0 3.3 3.5 3.8 4.0 4.2 4.5 4.7 5.0 5.3 5.5 5.7 (V)

(5) Which operating ambient temperature do you use?

(Circle the operating temperature range of use)

-50 -40 -30 -20 -10 0 10 20 30 40 50 60 70 80 90 (°C)

(6) Do you use I^2C (Inter IC) bus function?

☐ Not use ☐ Use

(7) Do you use IE (Inter Equipment) bus function?

☐ Not use ☐ Use

Thank you cooperation.

❋4. Special item (Indicate none if there is not specified item)

GZZ-SH13-28B<95A0>

Mask ROM number	

MITSUBISHI ELECTRIC-CHIP 16-BIT MICROCOMPUTER M30620MCA-XXXFP/GP MASK ROM CONFIRMATION FORM

Receipt	Date :	
	Section head signature	Supervisor signature

Note : Please complete all items marked ✵ .

✵	Customer	Company name	TEL ( )	Issuance signature	Submitted by	Supervisor
		Date issued	Date :			

✵1. Check sheet

Mitsubishi processes the mask files generated by the mask file generation utilities out of those held on the floppy disks you give in to us, and forms them into masks. Hence, we assume liability provided that there is any discrepancy between the contents of these mask files and the ROM data to be burned into products we produce. Check thoroughly the contents of the mask files you give in.
Prepare 3.5 inches 2HD (IBM format) floppy disks. And store only one mask file in a floppy disk.

Microcomputer type No. : ☐ M30620MCA-XXXFP ☐ M30620MCA-XXXGP

File code : ☐☐☐☐☐☐☐☐ (hex)

Mask file name : ☐☐☐☐☐☐☐☐ .MSK (alpha-numeric 8-digit)

✵2. Mark specification

The mark specification differs according to the type of package. After entering the mark specification on the separate mark specification sheet (for each package), attach that sheet to this masking check sheet for submission to Mitsubishi.

For the M30620MCA-XXXFP, submit the 100P6S mark specification sheet. For the M30620MCA-XXXGP, submit the 100P6Q mark specification sheet.

✵3. Usage Conditions

For our reference when of testing our products, please reply to the following questions about the usage of the products you ordered.

(1) Which kind of XIN-XOUT oscillation circuit is used?

☐ Ceramic resonator ☐ Quartz-crystal oscillator
☐ External clock input ☐ Other ( )

What frequency do not use?

f(XIN) = ☐ MHz

GZZ-SH13-28B<95A0>

Mask ROM number	

MITSUBISHI ELECTRIC-CHIP 16-BIT MICROCOMPUTER M30620MCA-XXXFP/GP MASK ROM CONFIRMATION FORM

(2) Which kind of XCIN-XCOUT oscillation circuit is used?

☐ Ceramic resonator ☐ Quartz-crystal oscillator

☐ External clock input ☐ Other ( )

What frequency do not use?

f(XCIN) = ☐ kHz

(3) Which operation mode do you use?

☐ Single-chip mode ☐ Memory expansion mode

☐ Microprocessor mode

(4) Which operating supply voltage do you use?

(Circle the operating voltage range of use)

2.4 2.7 3.0 3.3 3.5 3.8 4.0 4.2 4.5 4.7 5.0 5.3 5.5 5.7 (V)

(5) Which operating ambient temperature do you use?

(Circle the operating temperature range of use)

-50 -40 -30 -20 -10 0 10 20 30 40 50 60 70 80 90 (°C)

(6) Do you use I^2C (Inter IC) bus function?

☐ Not use ☐ Use

(7) Do you use IE (Inter Equipment) bus function?

☐ Not use ☐ Use

Thank you cooperation.

✵4. Special item (Indicate none if there is not specified item)

GZZ-SH13-40B<96A0>

Mask ROM number	

Receipt	Date :	
	Section head signature	Supervisor signature

MITSUBISHI ELECTRIC-CHIP 16-BIT MICROCOMPUTER M30622M4A-XXXFP/GP MASK ROM CONFIRMATION FORM

Note : Please complete all items marked ✵ .

				Issuance signature	Submitted by	Supervisor
✵	Customer	Company name	TEL ( )			
		Date issued	Date :			

✵1. Check sheet

Mitsubishi processes the mask files generated by the mask file generation utilities out of those held on the floppy disks you give in to us, and forms them into masks. Hence, we assume liability provided that there is any discrepancy between the contents of these mask files and the ROM data to be burned into products we produce. Check thoroughly the contents of the mask files you give in.
Prepare 3.5 inches 2HD (IBM format) floppy disks. And store only one mask file in a floppy disk.

Microcomputer type No. : ☐ M30622M4A-XXXFP ☐ M30622M4A-XXXGP

File code : ☐☐☐☐☐☐☐☐ (hex)

Mask file name : ☐☐☐☐☐☐☐☐ .MSK (alpha-numeric 8-digit)

✵2. Mark specification

The mark specification differs according to the type of package. After entering the mark specification on the separate mark specification sheet (for each package), attach that sheet to this masking check sheet for submission to Mitsubishi.

For the M30622M4A-XXXFP, submit the 100P6S mark specification sheet. For the M30622M4A-XXXGP, submit the 100P6Q mark specification sheet.

✵3. Usage Conditions

For our reference when of testing our products, please reply to the following questions about the usage of the products you ordered.

(1) Which kind of XIN-XOUT oscillation circuit is used?

☐ Ceramic resonator ☐ Quartz-crystal oscillator
☐ External clock input ☐ Other ( )

What frequency do not use?

f(XIN) = ☐ MHz

GZZ-SH13-40B<96A0>

Mask ROM number	

MITSUBISHI ELECTRIC-CHIP 16-BIT MICROCOMPUTER M30622M4A-XXXFP/GP MASK ROM CONFIRMATION FORM

(2) Which kind of XCIN-XCOUT oscillation circuit is used?

☐ Ceramic resonator ☐ Quartz-crystal oscillator

☐ External clock input ☐ Other ( )

What frequency do not use?

f(XCIN) = ☐ kHz

(3) Which operation mode do you use?

☐ Single-chip mode ☐ Memory expansion mode

☐ Microprocessor mode

(4) Which operating supply voltage do you use?

(Circle the operating voltage range of use)

2.4 2.7 3.0 3.3 3.5 3.8 4.0 4.2 4.5 4.7 5.0 5.3 5.5 5.7 (V)

(5) Which operating ambient temperature do you use?

(Circle the operating temperature range of use)

-50 -40 -30 -20 -10 0 10 20 30 40 50 60 70 80 90 (°C)

(6) Do you use I^2C (Inter IC) bus function?

☐ Not use ☐ Use

(7) Do you use IE (Inter Equipment) bus function?

☐ Not use ☐ Use

Thank you cooperation.

❋4. Special item (Indicate none if there is not specified item)

GZZ-SH13-38B<96A0>

Mask ROM number	

MITSUBISHI ELECTRIC-CHIP 16-BIT MICROCOMPUTER M30622M8A-XXXFP/GP MASK ROM CONFIRMATION FORM

Receipt	Date :	
	Section head signature	Supervisor signature

Note : Please complete all items marked ✵ .

✵	Customer	Company name	TEL ( )	Issuance signature	Submitted by	Supervisor
		Date issued	Date :			

✵1. Check sheet

Mitsubishi processes the mask files generated by the mask file generation utilities out of those held on the floppy disks you give in to us, and forms them into masks. Hence, we assume liability provided that there is any discrepancy between the contents of these mask files and the ROM data to be burned into products we produce. Check thoroughly the contents of the mask files you give in.
Prepare 3.5 inches 2HD (IBM format) floppy disks. And store only one mask file in a floppy disk.

Microcomputer type No. : ☐ M30622M8A-XXXFP ☐ M30622M8A-XXXGP

File code : ☐☐☐☐☐☐☐☐ (hex)

Mask file name : ☐☐☐☐☐☐☐☐ .MSK (alpha-numeric 8-digit)

✵2. Mark specification

The mark specification differs according to the type of package. After entering the mark specification on the separate mark specification sheet (for each package), attach that sheet to this masking check sheet for submission to Mitsubishi.

For the M30622M8A-XXXFP, submit the 100P6S mark specification sheet. For the M30622M8A-XXXGP, submit the 100P6Q mark specification sheet.

✵3. Usage Conditions

For our reference when of testing our products, please reply to the following questions about the usage of the products you ordered.

(1) Which kind of XIN-XOUT oscillation circuit is used?

☐ Ceramic resonator ☐ Quartz-crystal oscillator
☐ External clock input ☐ Other ( )

What frequency do not use?

f(XIN) = ☐ MHz

GZZ-SH13-38B<96A0>

Mask ROM number	

MITSUBISHI ELECTRIC-CHIP 16-BIT MICROCOMPUTER M30622M8A-XXXFP/GP MASK ROM CONFIRMATION FORM

(2) Which kind of XCIN-XCOUT oscillation circuit is used?

☐ Ceramic resonator ☐ Quartz-crystal oscillator

☐ External clock input ☐ Other ( )

What frequency do not use?

f(XCIN) = ☐ kHz

(3) Which operation mode do you use?

☐ Single-chip mode ☐ Memory expansion mode

☐ Microprocessor mode

(4) Which operating supply voltage do you use?

(Circle the operating voltage range of use)

2.4 2.7 3.0 3.3 3.5 3.8 4.0 4.2 4.5 4.7 5.0 5.3 5.5 5.7 (V)

(5) Which operating ambient temperature do you use?

(Circle the operating temperature range of use)

-50 -40 -30 -20 -10 0 10 20 30 40 50 60 70 80 90 (°C)

(6) Do you use I^2C (Inter IC) bus function?

☐ Not use ☐ Use

(7) Do you use IE (Inter Equipment) bus function?

☐ Not use ☐ Use

Thank you cooperation.

✵4. Special item (Indicate none if there is not specified item)

GZZ-SH13-34B<96A0>

Mask ROM number	

MITSUBISHI ELECTRIC-CHIP 16-BIT MICROCOMPUTER M30622MAA-XXXFP/GP MASK ROM CONFIRMATION FORM

Receipt	Date :	
	Section head signature	Supervisor signature

Note : Please complete all items marked ✵ .

✵	Customer	Company name	TEL ( )	Issuance signature	Submitted by	Supervisor
		Date issued	Date :			

✵1. Check sheet

Mitsubishi processes the mask files generated by the mask file generation utilities out of those held on the floppy disks you give in to us, and forms them into masks. Hence, we assume liability provided that there is any discrepancy between the contents of these mask files and the ROM data to be burned into products we produce. Check thoroughly the contents of the mask files you give in.
Prepare 3.5 inches 2HD (IBM format) floppy disks. And store only one mask file in a floppy disk.

Microcomputer type No. : ☐ M30622MAA-XXXFP ☐ M30622MAA-XXXGP

File code : ☐☐☐☐☐☐☐☐ (hex)

Mask file name : ☐☐☐☐☐☐☐☐ .MSK (alpha-numeric 8-digit)

✵2. Mark specification

The mark specification differs according to the type of package. After entering the mark specification on the separate mark specification sheet (for each package), attach that sheet to this masking check sheet for submission to Mitsubishi.

For the M30622MAA-XXXFP, submit the 100P6S mark specification sheet. For the M30622MAA-XXXGP, submit the 100P6Q mark specification sheet.

✵3. Usage Conditions

For our reference when of testing our products, please reply to the following questions about the usage of the products you ordered.

(1) Which kind of XIN-XOUT oscillation circuit is used?

☐ Ceramic resonator ☐ Quartz-crystal oscillator
☐ External clock input ☐ Other ( )

What frequency do not use?

f(XIN) = ☐ MHz

GZZ-SH13-34B<96A0>

Mask ROM number	

MITSUBISHI ELECTRIC-CHIP 16-BIT MICROCOMPUTER M30622MAA-XXXFP/GP MASK ROM CONFIRMATION FORM

(2) Which kind of XCIN-XCOUT oscillation circuit is used?

☐ Ceramic resonator ☐ Quartz-crystal oscillator

☐ External clock input ☐ Other ( )

What frequency do not use?

f(XCIN) = ☐ kHz

(3) Which operation mode do you use?

☐ Single-chip mode ☐ Memory expansion mode

☐ Microprocessor mode

(4) Which operating supply voltage do you use?

(Circle the operating voltage range of use)

2.4 2.7 3.0 3.3 3.5 3.8 4.0 4.2 4.5 4.7 5.0 5.3 5.5 5.7 (V)

(5) Which operating ambient temperature do you use?

(Circle the operating temperature range of use)

-50 -40 -30 -20 -10 0 10 20 30 40 50 60 70 80 90 (°C)

(6) Do you use I^2C (Inter IC) bus function?

☐ Not use ☐ Use

(7) Do you use IE (Inter Equipment) bus function?

☐ Not use ☐ Use

Thank you cooperation.

❋4. Special item (Indicate none if there is not specified item)

GZZ-SH13-39B<96A0>

Mask ROM number	

MITSUBISHI ELECTRIC-CHIP 16-BIT MICROCOMPUTER M30622MCA-XXXFP/GP MASK ROM CONFIRMATION FORM

Receipt	Date :	
	Section head signature	Supervisor signature

Note : Please complete all items marked ✵ .

✵	Customer	Company name	TEL ( )	Issuance signature	Submitted by	Supervisor
		Date issued	Date :			

✵1. Check sheet

Mitsubishi processes the mask files generated by the mask file generation utilities out of those held on the floppy disks you give in to us, and forms them into masks. Hence, we assume liability provided that there is any discrepancy between the contents of these mask files and the ROM data to be burned into products we produce. Check thoroughly the contents of the mask files you give in.
Prepare 3.5 inches 2HD (IBM format) floppy disks. And store only one mask file in a floppy disk.

Microcomputer type No. : ☐ M30622MCA-XXXFP ☐ M30622MCA-XXXGP

File code : ☐☐☐☐☐☐☐☐ (hex)

Mask file name : ☐☐☐☐☐☐☐☐ .MSK (alpha-numeric 8-digit)

✵2. Mark specification

The mark specification differs according to the type of package. After entering the mark specification on the separate mark specification sheet (for each package), attach that sheet to this masking check sheet for submission to Mitsubishi.

For the M30622MCA-XXXFP, submit the 100P6S mark specification sheet. For the M30622MCA-XXXGP, submit the 100P6Q mark specification sheet.

✵3. Usage Conditions

For our reference when of testing our products, please reply to the following questions about the usage of the products you ordered.

(1) Which kind of XIN-XOUT oscillation circuit is used?

☐ Ceramic resonator ☐ Quartz-crystal oscillator
☐ External clock input ☐ Other ( )

What frequency do not use?

f(XIN) = ☐ MHz

GZZ-SH13-39B<96A0>

Mask ROM number	

MITSUBISHI ELECTRIC-CHIP 16-BIT MICROCOMPUTER M30622MCA-XXXFP/GP MASK ROM CONFIRMATION FORM

(2) Which kind of XCIN-XCOUT oscillation circuit is used?

☐ Ceramic resonator ☐ Quartz-crystal oscillator

☐ External clock input ☐ Other ( )

What frequency do not use?

f(XCIN) = ☐ kHz

(3) Which operation mode do you use?

☐ Single-chip mode ☐ Memory expansion mode

☐ Microprocessor mode

(4) Which operating supply voltage do you use?

(Circle the operating voltage range of use)

2.4 2.7 3.0 3.3 3.5 3.8 4.0 4.2 4.5 4.7 5.0 5.3 5.5 5.7 (V)

(5) Which operating ambient temperature do you use?

(Circle the operating temperature range of use)

-50 -40 -30 -20 -10 0 10 20 30 40 50 60 70 80 90 (°C)

(6) Do you use I^2C (Inter IC) bus function?

☐ Not use ☐ Use

(7) Do you use IE (Inter Equipment) bus function?

☐ Not use ☐ Use

Thank you cooperation.

✵4. Special item (Indicate none if there is not specified item)

GZZ-SH13-30B<95A0>

Mask ROM number	

MITSUBISHI ELECTRIC-CHIP 16-BIT MICROCOMPUTER M30624MGA-XXXFP/GP MASK ROM CONFIRMATION FORM

Receipt	Date :	
	Section head signature	Supervisor signature

Note : Please complete all items marked ✵ .

✵	Customer	Company name	TEL ( )	Issuance signature	Submitted by	Supervisor
		Date issued	Date :			

✵1. Check sheet

Mitsubishi processes the mask files generated by the mask file generation utilities out of those held on the floppy disks you give in to us, and forms them into masks. Hence, we assume liability provided that there is any discrepancy between the contents of these mask files and the ROM data to be burned into products we produce. Check thoroughly the contents of the mask files you give in.
Prepare 3.5 inches 2HD (IBM format) floppy disks. And store only one mask file in a floppy disk.

Microcomputer type No. : ☐ M30624MGA-XXXFP ☐ M30624MGA-XXXGP

File code : ☐☐☐☐☐☐☐☐ (hex)

Mask file name : ☐☐☐☐☐☐☐☐ .MSK (alpha-numeric 8-digit)

✵2. Mark specification

The mark specification differs according to the type of package. After entering the mark specification on the separate mark specification sheet (for each package), attach that sheet to this masking check sheet for submission to Mitsubishi.

For the M30624MGA-XXXFP, submit the 100P6S mark specification sheet. For the M30624MGA-XXXGP, submit the 100P6Q mark specification sheet.

✵3. Usage Conditions

For our reference when of testing our products, please reply to the following questions about the usage of the products you ordered.

(1) Which kind of XIN-XOUT oscillation circuit is used?

☐ Ceramic resonator ☐ Quartz-crystal oscillator
☐ External clock input ☐ Other ( )

What frequency do not use?

f(XIN) = ☐ MHz

GZZ-SH13-30B<95A0>

Mask ROM number	

MITSUBISHI ELECTRIC-CHIP 16-BIT MICROCOMPUTER M30624MGA-XXXFP/GP MASK ROM CONFIRMATION FORM

(2) Which kind of XCIN-XCOUT oscillation circuit is used?

☐ Ceramic resonator ☐ Quartz-crystal oscillator

☐ External clock input ☐ Other ( )

What frequency do not use?

f(XCIN) = ☐ kHz

(3) Which operation mode do you use?

☐ Single-chip mode ☐ Memory expansion mode

☐ Microprocessor mode

(4) Which operating supply voltage do you use?

(Circle the operating voltage range of use)

2.4 2.7 3.0 3.3 3.5 3.8 4.0 4.2 4.5 4.7 5.0 5.3 5.5 5.7 (V)

(5) Which operating ambient temperature do you use?

(Circle the operating temperature range of use)

-50 -40 -30 -20 -10 0 10 20 30 40 50 60 70 80 90 (°C)

(6) Do you use I^2C (Inter IC) bus function?

☐ Not use ☐ Use

(7) Do you use IE (Inter Equipment) bus function?

☐ Not use ☐ Use

Thank you cooperation.

❋4. Special item (Indicate none if there is not specified item)

Outline Performance

Table 1.25.1 shows the outline performance of the M16C/62A (flash memory version).

Table 1.25.1. Outline performance of the M16C/62A (flash memory version)

Item		Performance
Flash memory operation mode		Three modes (parallel I/O, standard serial I/O, CPU rewrite)
Erase block division	User ROM area	See Figure 1.25.1
	Boot ROM area	One division (8 Kbytes) (Note)
Program method		In units of pages (in units of 256 bytes)
Erase method		Collective erase/block erase
Program/erase control method		Program/erase control by software command
Protect method		Protected for each block by lock bit
Number of commands		8 commands
Program/erase count		100 times
Data retantion		10 years
ROM code protect		Parallel I/O and standard serial I/O modes are supported.

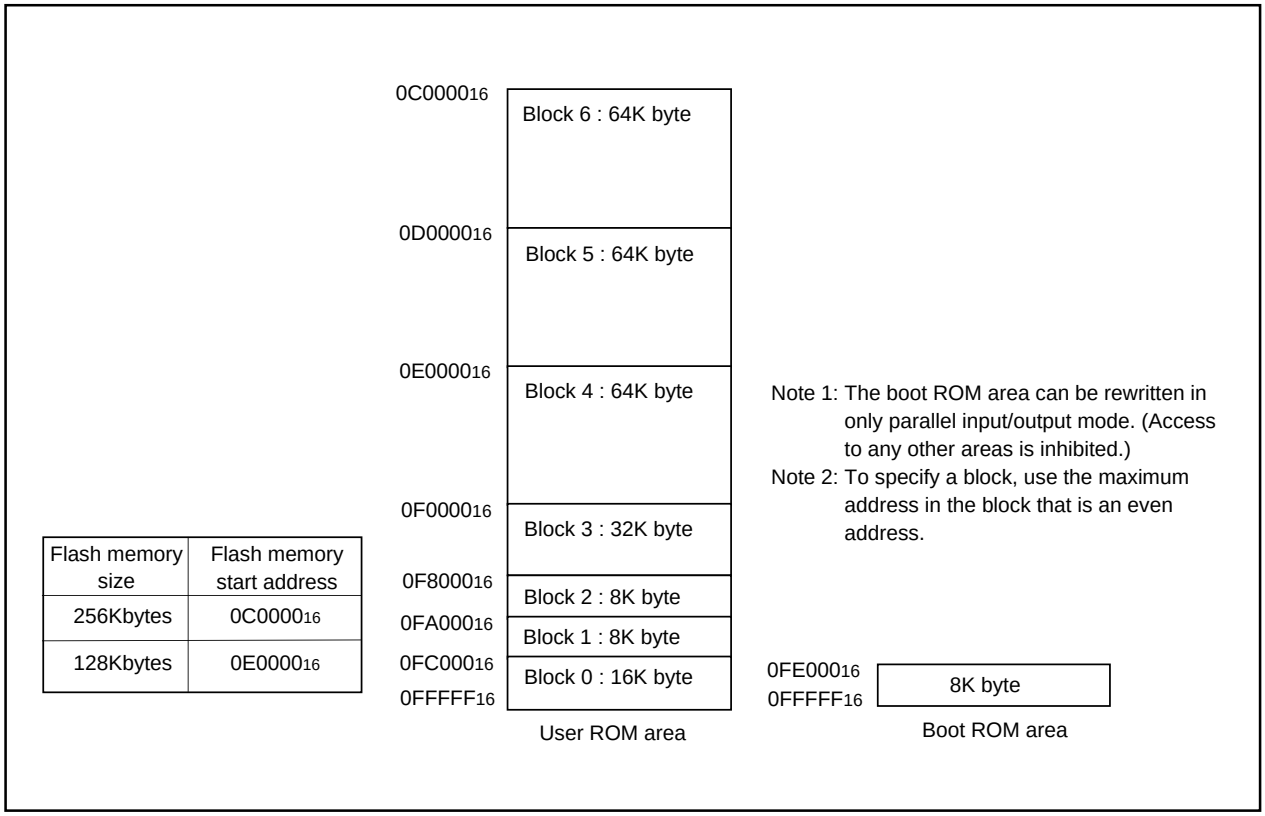
Note: The boot ROM area contains a standard serial I/O mode control program which is stored in it when shipped from the factory. This area can be erased and programmed in only parallel I/O mode.

Flash Memory

The M16C/62A (flash memory version) contains the flash memory that can be rewritten with a single voltage. For this flash memory, three flash memory modes are available in which to read, program, and erase: parallel I/O and standard serial I/O modes in which the flash memory can be manipulated using a programmer and a CPU rewrite mode in which the flash memory can be manipulated by the Central Processing Unit (CPU). Each mode is detailed in the pages to follow.

The flash memory is divided into several blocks as shown in Figure 1.25.1, so that memory can be erased one block at a time. Each block has a lock bit to enable or disable execution of an erase or program operation, allowing for data in each block to be protected.

In addition to the ordinary user ROM area to store a microcomputer operation control program, the flash memory has a boot ROM area that is used to store a program to control rewriting in CPU rewrite and standard serial I/O modes. This boot ROM area has had a standard serial I/O mode control program stored in it when shipped from the factory. However, the user can write a rewrite control program in this area that suits the user's application system. This boot ROM area can be rewritten in only parallel I/O mode.



Flash memory size	Flash memory start address
256Kbytes	0C000016
128Kbytes	0E000016

Figure 1.25.1. Block diagram of flash memory version

CPU Rewrite Mode

In CPU rewrite mode, the on-chip flash memory can be operated on (read, program, or erase) under control of the Central Processing Unit (CPU).

In CPU rewrite mode, only the user ROM area shown in Figure 1.25.1 can be rewritten; the boot ROM area cannot be rewritten. Make sure the program and block erase commands are issued for only the user ROM area and each block area.

The control program for CPU rewrite mode can be stored in either user ROM or boot ROM area. In the CPU rewrite mode, because the flash memory cannot be read from the CPU, the rewrite control program must be transferred to any area other than the internal flash memory before it can be executed.

Microcomputer Mode and Boot Mode

The control program for CPU rewrite mode must be written into the user ROM or boot ROM area in parallel I/O mode beforehand. (If the control program is written into the boot ROM area, the standard serial I/O mode becomes unusable.)

See Figure 1.25.1 for details about the boot ROM area.

Normal microcomputer mode is entered when the microcomputer is reset with pulling CNV_{SS} pin low. In this case, the CPU starts operating using the control program in the user ROM area.

When the microcomputer is reset by pulling the $P5_5$ pin low, the CNV_{SS} pin high, and the $P5_0$ pin high, the CPU starts operating using the control program in the boot ROM area. This mode is called the “boot” mode. The control program in the boot ROM area can also be used to rewrite the user ROM area.

Block Address

Block addresses refer to the maximum even address of each block. These addresses are used in the block erase command, lock bit program command, and read lock status command.

Outline Performance (CPU Rewrite Mode)

In the CPU rewrite mode, the CPU erases, programs and reads the internal flash memory as instructed by software commands. Operations must be executed from a memory other than the internal flash memory, such as the internal RAM.

When the CPU rewrite mode select bit (bit 1 at address $03B7_{16}$) is set to “1”, transition to CPU rewrite mode occurs and software commands can be accepted.

In the CPU rewrite mode, write to and read from software commands and data into even-numbered address (“0” for byte address A_0) in 16-bit units. Always write 8-bit software commands into even-numbered address. Commands are ignored with odd-numbered addresses.

Use software commands to control program and erase operations. Whether a program or erase operation has terminated normally or in error can be verified by reading the status register.

Figure 1.26.1 shows the flash memory control register 0 and the flash memory control register 1.

Bit 0 of the flash memory control register 0 is the RY/$\overline{BY}$ status flag used exclusively to read the operating status of the flash memory. During programming and erase operations, it is “0”. Otherwise, it is “1”.

Bit 1 of the flash memory control register 0 is the CPU rewrite mode select bit. The CPU rewrite mode is entered by setting this bit to “1”, so that software commands become acceptable. In CPU rewrite mode, the CPU becomes unable to access the internal flash memory directly. Therefore, write bit 1 in an area other than the internal flash memory. Also only when $\overline{NMI}$ pin is "H" level. To set this bit to “1”, it is necessary to write “0” and then write “1” in succession. The bit can be set to “0” by only writing a “0” .

Bit 2 of the flash memory control register 0 is a lock bit disable select bit. By setting this bit to “1”, it is possible to disable erase and write protect (block lock) effectuated by the lock bit data. The lock bit disable select bit only disables the lock bit function; it does not change the lock data bit value. However, if an erase operation is performed when this bit =“1”, the lock bit data that is “0” (locked) is set to “1” (unlocked) after erasure. To set this bit to “1”, it is necessary to write “0” and then write “1” in succession. This bit can be manipulated only when the CPU rewrite mode select bit = “1”.

Bit 3 of the flash memory control register 0 is the flash memory reset bit used to reset the control circuit of the internal flash memory. This bit is used when exiting CPU rewrite mode and when flash memory access has failed. When the CPU rewrite mode select bit is “1”, writing “1” for this bit resets the control circuit. To release the reset, it is necessary to set this bit to “0”.

Bit 5 of the flash memory control register 0 is a user ROM area select bit which is effective in only boot mode. If this bit is set to “1” in boot mode, the area to be accessed is switched from the boot ROM area to the user ROM area. When the CPU rewrite mode needs to be used in boot mode, set this bit to “1”. Note that if the microcomputer is booted from the user ROM area, it is always the user ROM area that can be accessed and this bit has no effect. When in boot mode, the function of this bit is effective regardless of whether the CPU rewrite mode is on or off. Write to this bit only when executing out of an area other than the internal flash memory.

Bit 3 of the flash memory control register 1 turns power supply to the internal flash memory on/off. When this bit is set to “1”, power is not supplied to the internal flash memory, thus power consumption can be reduced. However, in this state, the internal flash memory cannot be accessed. To set this bit to “1”, it is necessary to write “0” and then write “1” in succession. Use this bit mainly in the low speed mode (when X_{CIN} is the count source of BCLK).

When the CPU is shifted to the stop or wait modes, power to the internal flash memory is automatically shut off. It is reconnected automatically when CPU operation is restored. Therefore, it is not particularly necessary to set flash memory control register 1.

Figure 1.26.2 shows a flowchart for setting/releasing the CPU rewrite mode. Figure 1.26.3 shows a flowchart for shifting to the low speed mode. Always perform operation as indicated in these flowcharts.

Flash memory control register 0

b7	b6	b5	b4	b3	b2	b1	b0
			0				

Symbol	Address	When reset
FMR0	$03B7_{16}$	$XX000001_2$

Bit symbol	Bit name	Function	R	W
FMR00	RY/$\overline{BY}$ status flag	0: Busy (being written or erased) 1: Ready	○	×
FMR01	CPU rewrite mode select bit (Note 1)	0: Normal mode (Software commands invalid) 1: CPU rewrite mode (Software commands acceptable)	○	○
FMR02	Lock bit disable select bit (Note 2)	0: Block lock by lock bit data is enabled 1: Block lock by lock bit data is disabled	○	○
FMR03	Flash memory reset bit (Note 3)	0: Normal operation 1: Reset	○	○
Reserved bit		Must always be set to “0”	○	○
FMR05	User ROM area select bit (Note 4) (Effective in only boot mode)	0: Boot ROM area is accessed 1: User ROM area is accessed	○	○
Nothing is assigned. When write, set "0". When read, values are indeterminate.			—	—

Note 1: For this bit to be set to “1”, the user needs to write a “0” and then a “1” to it in succession. When it is not this procedure, it is not enacted in “1”. This is necessary to ensure that no interrupt or DMA transfer will be executed during the interval. Write to this bit only when executing out of an area other than the internal flash memory. Also only when NMI pin is “H” level. Clear this bit to “0” after read array command.
Note 2: For this bit to be set to “1”, the user needs to write a “0” and then a “1” to it in succession when the CPU rewrite mode select bit = “1”. When it is not this procedure, it is not enacted in “1”. This is necessary to ensure that no interrupt or DMA transfer will be executed during the interval.
Note 3: Effective only when the CPU rewrite mode select bit = 1. Set this bit to 0 subsequently after setting it to 1 (reset).
Note 4: Write to this bit only when executing out of an area other than the internal flash memory.

Flash memory control register 1

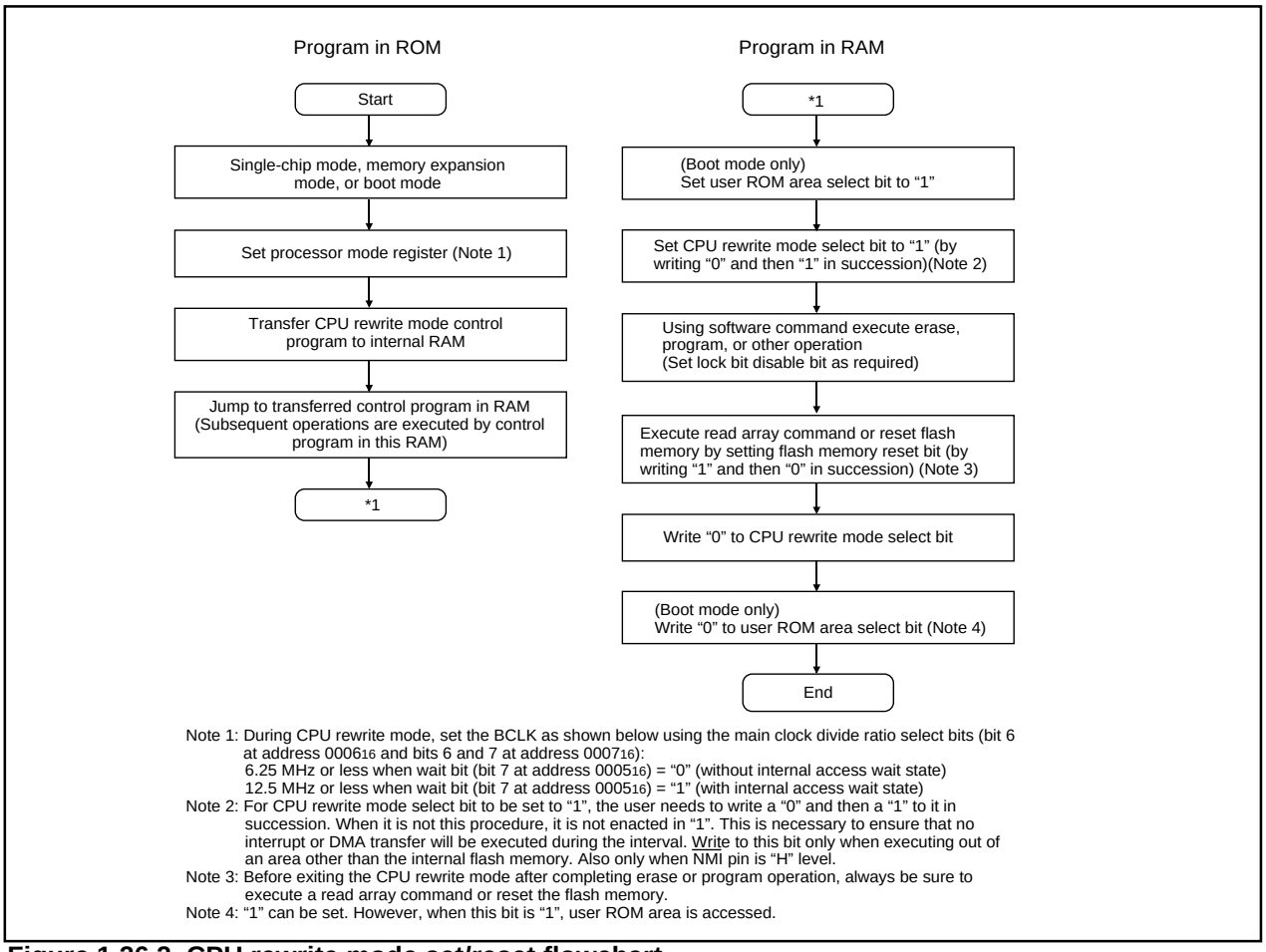
b7	b6	b5	b4	b3	b2	b1	b0
0	0	0	0		0	0	0

Symbol	Address	When reset
FMR1	$03B6_{16}$	$XXXX0XXX_2$

Bit symbol	Bit name	Function	R	W
Reserved bit		Must always be set to “0”	—	○
FMR13	Flash memory power supply-OFF bit (Note)	0: Flash memory power supply is connected 1: Flash memory power supply-off	○	○
Reserved bit		Must always be set to “0”	—	○

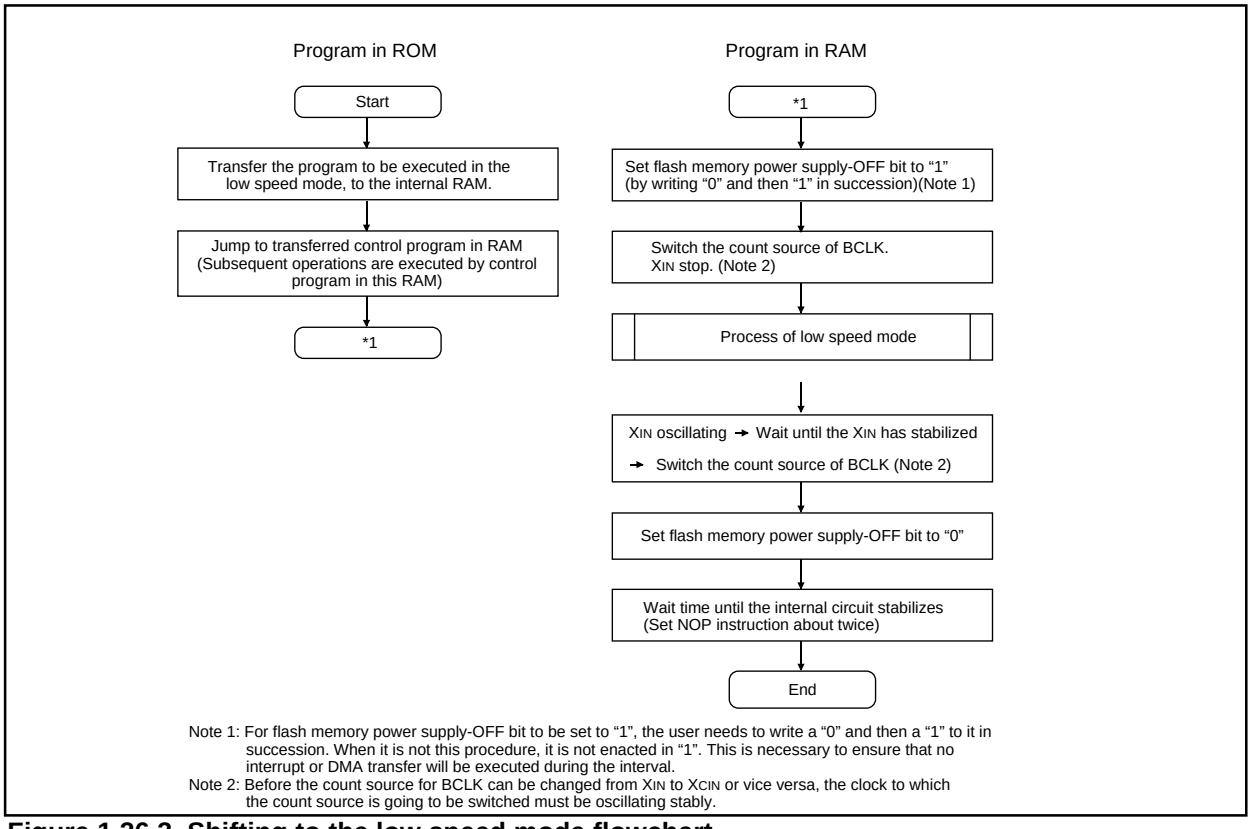
Note : For this bit to be set to “1”, the user needs to write a “0” and then a “1” to it in succession. When it is not this procedure, it is not enacted in “1”. This is necessary to ensure that no interrupt or DMA transfer will be executed during the interval. During parallel I/O mode,programming,erase or read of flash memory is not controlled by this bit,only by external pins. Write to this bit only when executing out of an area other than the internal flash memory.

Figure 1.26.1. Flash memory control registers



Note 1: During CPU rewrite mode, set the BCLK as shown below using the main clock divide ratio select bits (bit 6 at address 0006_{16} and bits 6 and 7 at address 0007_{16}):
6.25 MHz or less when wait bit (bit 7 at address 0005_{16}) = "0" (without internal access wait state)
12.5 MHz or less when wait bit (bit 7 at address 0005_{16}) = "1" (with internal access wait state)
Note 2: For CPU rewrite mode select bit to be set to "1", the user needs to write a "0" and then a "1" to it in succession. When it is not this procedure, it is not enacted in "1". This is necessary to ensure that no interrupt or DMA transfer will be executed during the interval. Write to this bit only when executing out of an area other than the internal flash memory. Also only when $\overline{NMI}$ pin is "H" level.
Note 3: Before exiting the CPU rewrite mode after completing erase or program operation, always be sure to execute a read array command or reset the flash memory.
Note 4: "1" can be set. However, when this bit is "1", user ROM area is accessed.

Figure 1.26.2. CPU rewrite mode set/reset flowchart



Note 1: For flash memory power supply-OFF bit to be set to "1", the user needs to write a "0" and then a "1" to it in succession. When it is not this procedure, it is not enacted in "1". This is necessary to ensure that no interrupt or DMA transfer will be executed during the interval.
Note 2: Before the count source for BCLK can be changed from X_{IN} to X_{CIN} or vice versa, the clock to which the count source is going to be switched must be oscillating stably.

Figure 1.26.3. Shifting to the low speed mode flowchart

Precautions on CPU Rewrite Mode

Described below are the precautions to be observed when rewriting the flash memory in CPU rewrite mode.

(1) Operation speed

During CPU rewrite mode, set the BCLK as shown below using the main clock divide ratio select bit (bit 6 at address 0006_{16} and bits 6 and 7 at address 0007_{16}):

6.25 MHz or less when wait bit (bit 7 at address 0005_{16}) = 0 (without internal access wait state)

12.5 MHz or less when wait bit (bit 7 at address 0005_{16}) = 1 (with internal access wait state)

(2) Instructions inhibited against use

The instructions listed below cannot be used during CPU rewrite mode because they refer to the internal data of the flash memory:

UND instruction, INTO instruction, JMPS instruction, JSRS instruction, and BRK instruction

(3) Interrupts inhibited against use

The address match interrupt cannot be used during CPU rewrite mode because they refer to the internal data of the flash memory. If interrupts have their vector in the variable vector table, they can be used by transferring the vector into the RAM area. The $\overline{\text{NMI}}$ and watchdog timer interrupts can be used because the flash memory conterol register 0 and 1 is forcibly initialized and return to normal mode when each interrupt occurs. But it is needed that the jump addresses for each interrupt are set in the fixed vector table and there is an interrupt program. Since the rewrite operation is halted when the $\overline{\text{NMI}}$ and watchdog timer interrupts occur, it is needed that CPU rewriting mode select bit is set to “1” and the erase/program operation is performed over again.

(4) Internal reserved area expansion bit (Bit 3 at address 0005_{16})

The reserved area of the internal memory can be changed by using the internal reserved area expansion bit (bit 3 at address 0005_{16}). However, if the CPU rewrite mode select bit (bit 1 at address $03B7_{16}$) is set to 1, the internal reserved area expansion bit (bit 3 at address 0005_{16}) also is set to 1 automatically. Similarly, if the CPU rewrite mode select bit (bit 1 at address $03B7_{16}$) is set to 0, the internal reserved area expansion bit (bit 3 at address 0005_{16}) also is set to 0 automatically.

The precautions above apply to the products which RAM size is over 15 Kbytes or flash memory size is over 192 Kbytes.

(5) Reset

Reset input is always accepted. After a reset, the addresses $0C0000_{16}$ through $0CFFFF_{16}$ are made a reserved area and cannot be accessed. Therefore, if your product has this area in the user ROM area, do not write any address of this area to the reset vector. This area is made accessible by changing the internal reserved area expansion bit (bit 3 at address 0005_{16}) in a program.

(6) Access disable

Write CPU rewrite mode select bit, flash memory power supply-OFF bit and user ROM area select bit only when executing out of an area other than the internal flash memory.

(7) How to access

For CPU rewrite mode select bit, lock bit disable select bit, and flash memory power supply-OFF bit to be set to “1”, the user needs to write a “0” and then a “1” to it in succession. When it is not this procedure, it is not enacted in “1”. This is necessary to ensure that no interrupt or DMA transfer will be executed during the interval.

Write CPU rewrite mode select bit only when executing out of an area other than the internal flash memory. Also only when $\overline{\text{NMI}}$ pin is “H” level.

(8) Writing in the user ROM area

If power is lost while rewriting blocks that contain the flash rewrite program with the CPU rewrite mode, those blocks may not be correctly rewritten and it is possible that the flash memory can no longer be rewritten after that. Therefore, it is recommended to use the standard serial I/O mode or parallel I/O mode to rewrite these blocks.

(9) Using the lock bit

To use the CPU rewrite mode, use a boot program that can set and cancel the lock command.

Software Commands

Table 1.26.1 lists the software commands available with the M16C/62A (flash memory version).

After setting the CPU rewrite mode select bit to 1, write a software command to specify an erase or program operation. Note that when entering a software command, the upper byte (D_8 to D_{15}) is ignored.

The content of each software command is explained below.

Table 1.26.1. List of software commands (CPU rewrite mode)

Command	First bus cycle			Second bus cycle			Third bus cycle		
	Mode	Address	Data (D_0 to D_7)	Mode	Address	Data (D_0 to D_7)	Mode	Address	Data (D_0 to D_7)
Read array	Write	X (Note 6)	FF_{16}						
Read status register	Write	X	70_{16}	Read	X	SRD (Note 2)			
Clear status register	Write	X	50_{16}						
Page program (Note 3)	Write	X	41_{16}	Write	WA0 (Note 3)	WD0 (Note 3)	Write	WA1	WD1
Block erase	Write	X	20_{16}	Write	BA (Note 4)	$D0_{16}$			
Erase all unlock block	Write	X	$A7_{16}$	Write	X	$D0_{16}$			
Lock bit program	Write	X	77_{16}	Write	BA	$D0_{16}$			
Read lock bit status	Write	X	71_{16}	Read	BA	D_6 (Note 5)			

Note 1: When a software command is input, the high-order byte of data (D_8 to D_{15}) is ignored.
Note 2: SRD = Status Register Data
Note 3: WA = Write Address, WD = Write Data
WA and WD must be set sequentially from 00_{16} to FE_{16} (byte address; however, an even address). The page size is 256 bytes.
Note 4: BA = Block Address (Enter the maximum address of each block that is an even address.)
Note 5: D_6 corresponds to the block lock status. Block not locked when D_6 = 1, block locked when D_6 = 0.
Note 6: X denotes a given address in the user ROM area (that is an even address).

Read Array Command (FF_{16})

The read array mode is entered by writing the command code "FF_{16}" in the first bus cycle. When an even address to be read is input in one of the bus cycles that follow, the content of the specified address is read out at the data bus (D_0–D_{15}), 16 bits at a time.

The read array mode is retained intact until another command is written.

Read Status Register Command (70_{16})

When the command code "70_{16}" is written in the first bus cycle, the content of the status register is read out at the data bus (D_0–D_7) by a read in the second bus cycle.

The status register is explained in the next section.

Clear Status Register Command (50_{16})

This command is used to clear the bits SR3 to 5 of the status register after they have been set. These bits indicate that operation has ended in an error. To use this command, write the command code "50_{16}" in the first bus cycle.

Page Program Command (41$_{16}$)

Page program allows for high-speed programming in units of 256 bytes. Page program operation starts when the command code "41$_{16}$" is written in the first bus cycle. In the second bus cycle through the 129th bus cycle, the write data is sequentially written 16 bits at a time. At this time, the addresses A$_0$-A$_7$ need to be incremented by 2 from "00$_{16}$" to "FE$_{16}$." When the system finishes loading the data, it starts an auto write operation (data program and verify operation).

Whether the auto write operation is completed can be confirmed by reading the status register or the flash memory control register 0. At the same time the auto write operation starts, the read status register mode is automatically entered, so the content of the status register can be read out. The status register bit 7 (SR7) is set to 0 at the same time the auto write operation starts and is returned to 1 upon completion of the auto write operation. In this case, the read status register mode remains active until the Read Array command (FF$_{16}$) or Read Lock Bit Status command (71$_{16}$) is written or the flash memory is reset using its reset bit.

The RY/$\overline{BY}$ status flag of the flash memory control register 0 is 0 during auto write operation and 1 when the auto write operation is completed as is the status register bit 7.

After the auto write operation is completed, the status register can be read out to know the result of the auto write operation. For details, refer to the section where the status register is detailed.

Figure 1.26.4 shows an example of a page program flowchart.

Each block of the flash memory can be write protected by using a lock bit. For details, refer to the section where the data protect function is detailed.

Additional writes to the already programmed pages are prohibited.

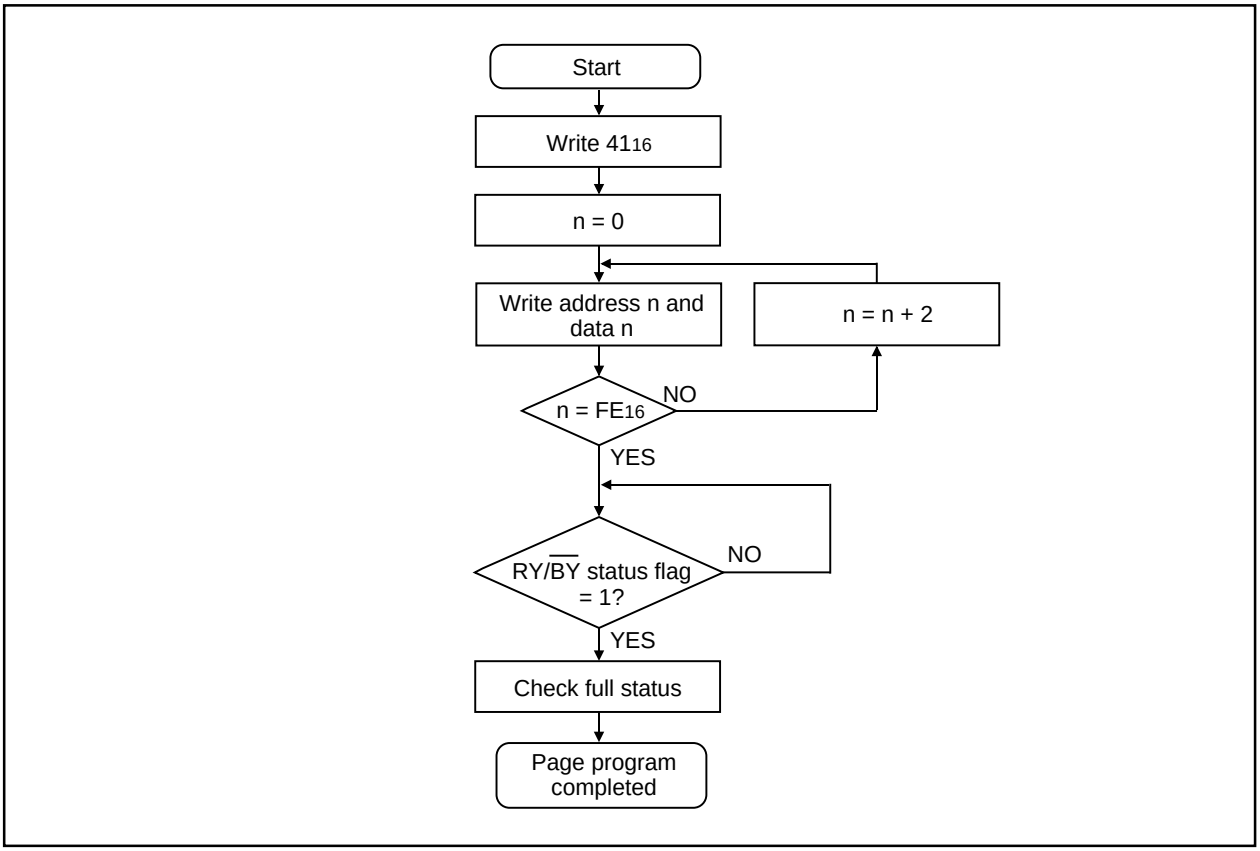


Figure 1.26.4. Page program flowchart

Block Erase Command (20_{16}/$D0_{16}$)

By writing the command code "20_{16}" in the first bus cycle and the confirmation command code "$D0_{16}$" in the second bus cycle that follows to the block address of a flash memory block, the system initiates an auto erase (erase and erase verify) operation.

Whether the auto erase operation is completed can be confirmed by reading the status register or the flash memory control register 0. At the same time the auto erase operation starts, the read status register mode is automatically entered, so the content of the status register can be read out. The status register bit 7 (SR7) is set to 0 at the same time the auto erase operation starts and is returned to 1 upon completion of the auto erase operation. In this case, the read status register mode remains active until the Read Array command (FF_{16}) or Read Lock Bit Status command (71_{16}) is written or the flash memory is reset using its reset bit.

The RY/$\overline{BY}$ status flag of the flash memory control register 0 is 0 during auto erase operation and 1 when the auto erase operation is completed as is the status register bit 7.

After the auto erase operation is completed, the status register can be read out to know the result of the auto erase operation. For details, refer to the section where the status register is detailed.

Figure 1.26.5 shows an example of a block erase flowchart.

Each block of the flash memory can be protected against erasure by using a lock bit. For details, refer to the section where the data protect function is detailed.

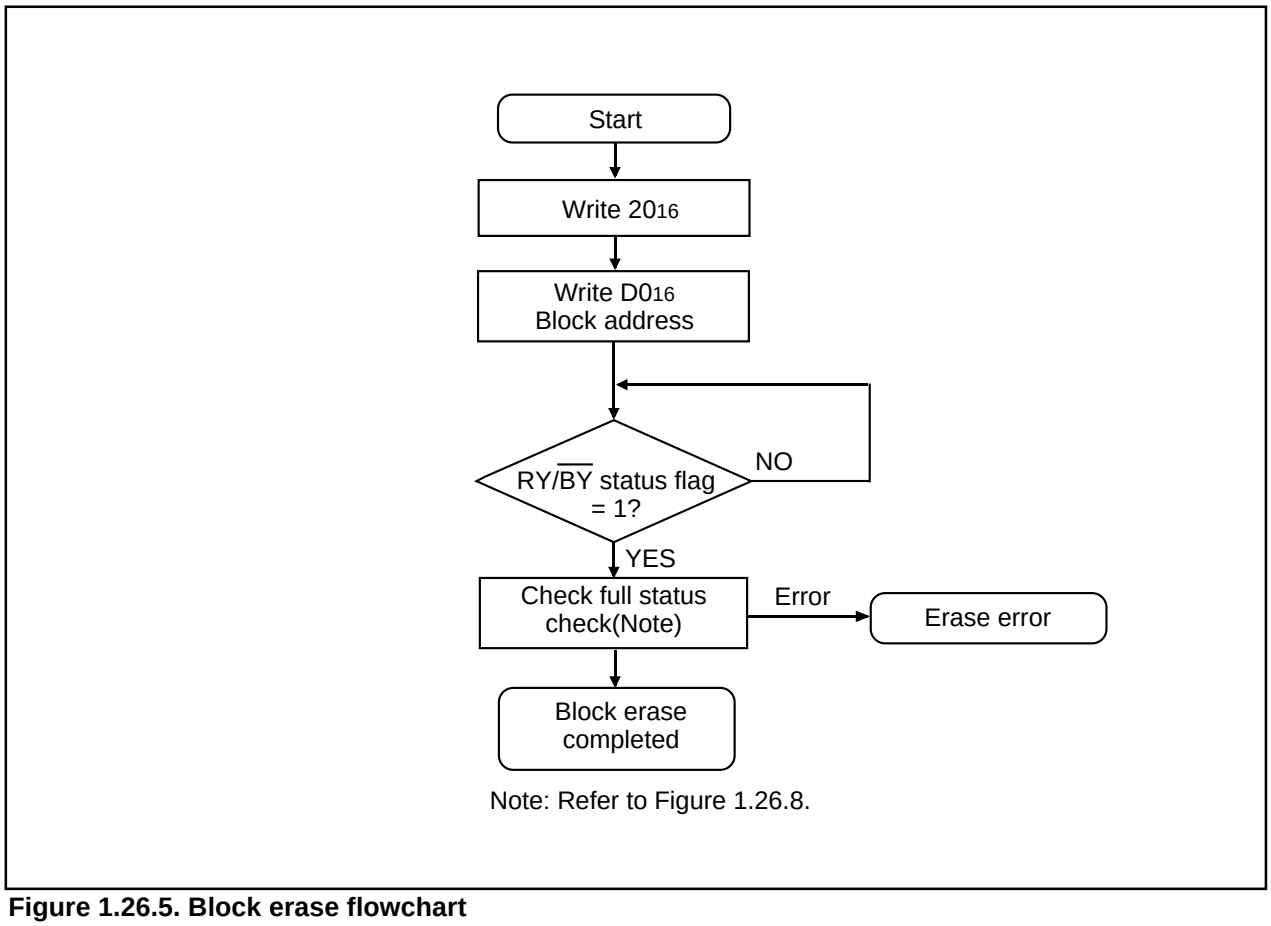


Figure 1.26.5. Block erase flowchart

Erase All Unlock Blocks Command ($A7_{16}$/$D0_{16}$)

By writing the command code "$A7_{16}$" in the first bus cycle and the confirmation command code "$D0_{16}$" in the second bus cycle that follows, the system starts erasing blocks successively.

Whether the erase all unlock blocks command is terminated can be confirmed by reading the status register or the flash memory control register 0, in the same way as for block erase. Also, the status register can be read out to know the result of the auto erase operation.

When the lock bit disable select bit of the flash memory control register 0 = 1, all blocks are erased no matter how the lock bit is set. On the other hand, when the lock bit disable select bit = 0, the function of the lock bit is effective and only nonlocked blocks (where lock bit data = 1) are erased.

Lock Bit Program Command (77_{16}/$D0_{16}$)

By writing the command code "77_{16}" in the first bus cycle and the confirmation command code "$D0_{16}$" in the second bus cycle that follows to the block address of a flash memory block, the system sets the lock bit for the specified block to 0 (locked).

Figure 1.26.6 shows an example of a lock bit program flowchart. The status of the lock bit (lock bit data) can be read out by a read lock bit status command.

Whether the lock bit program command is terminated can be confirmed by reading the status register or the flash memory control register 0, in the same way as for page program.

For details about the function of the lock bit and how to reset the lock bit, refer to the section where the data protect function is detailed.

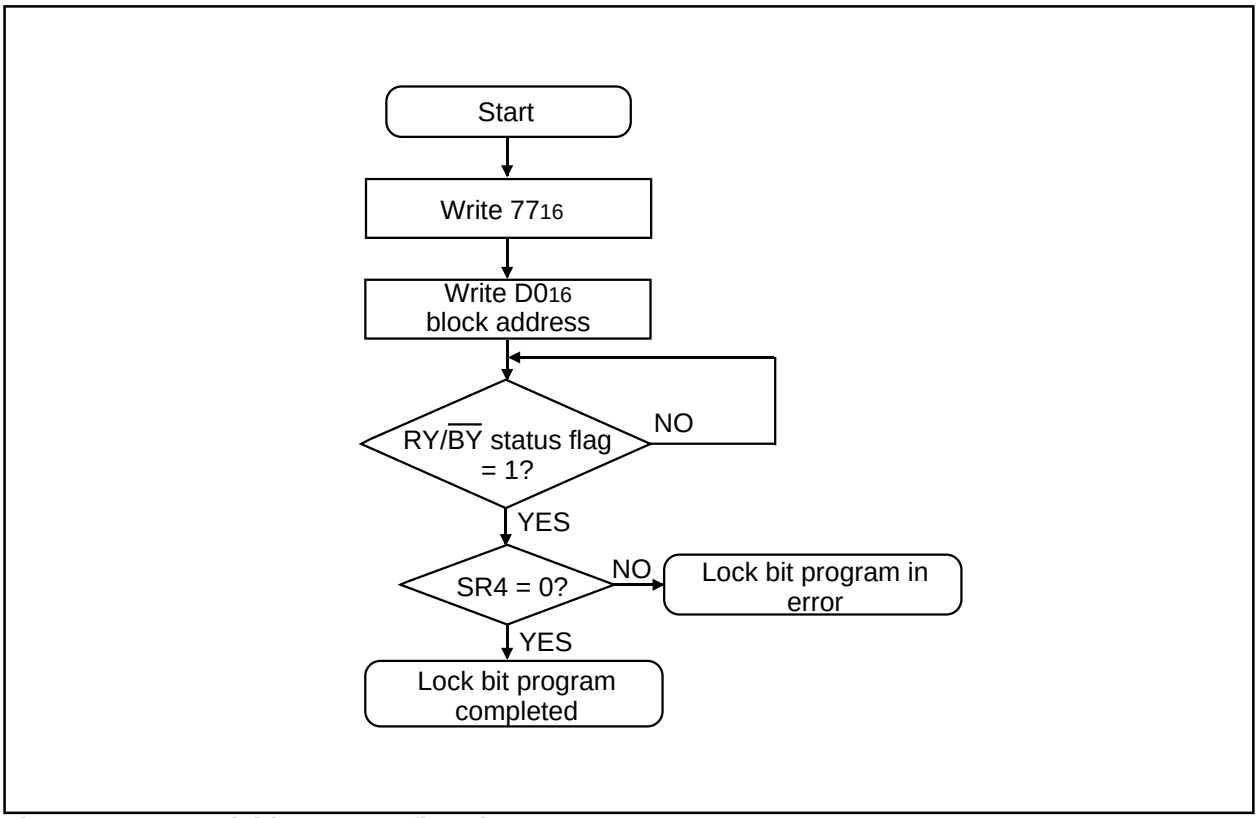


Figure 1.26.6. Lock bit program flowchart

Read Lock Bit Status Command (71_{16})

By writing the command code "71_{16}" in the first bus cycle and then the block address of a flash memory block in the second bus cycle that follows, the system reads out the status of the lock bit of the specified block on to the data bus(D_6).

Figure 1.26.7 shows an example of a read lock bit program flowchart.

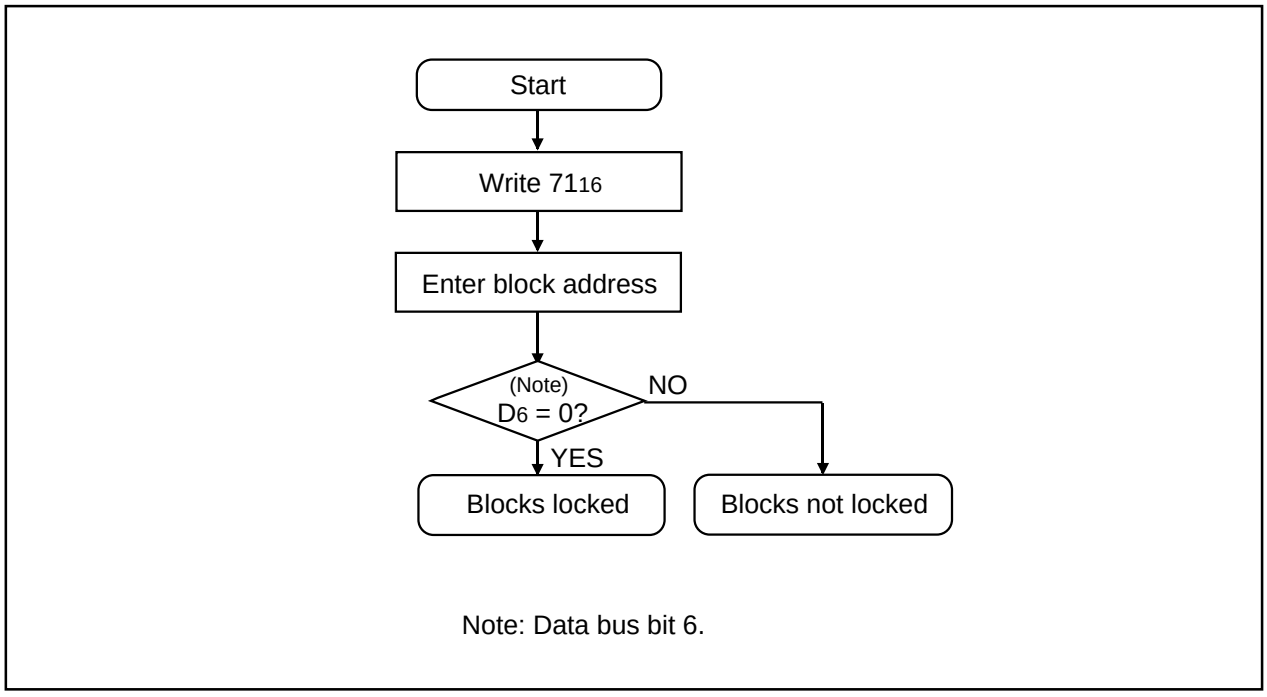


Figure 1.26.7. Read lock bit status flowchart

Data Protect Function (Block Lock)

Each block in Figure 1.25.1 has a nonvolatile lock bit to specify that the block be protected (locked) against erase/write. The lock bit program command is used to set the lock bit to 0 (locked). The lock bit of each block can be read out using the read lock bit status command.

Whether block lock is enabled or disabled is determined by the status of the lock bit and how the flash memory control register 0's lock bit disable select bit is set.

(1) When the lock bit disable select bit = "0", a specified block can be locked or unlocked by the lock bit status (lock bit data). Blocks whose lock bit data = "0" are locked, so they are disabled against erase/write. On the other hand, the blocks whose lock bit data = "1" are not locked, so they are enabled for erase/write.

(2) When the lock bit disable select bit = "1", all blocks are nonlocked regardless of the lock bit data, so they are enabled for erase/write. In this case, the lock bit data that is "0" (locked) is set to "1" (nonlocked) after erasure, so that the lock bit-actuated lock is removed.

Status Register

The status register indicates the operating status of the flash memory and whether an erase or program operation has terminated normally or in an error. The content of this register can be read out by only writing the read status register command (70_{16}). Table 1.26.2 details the status register.

The status register is cleared by writing the Clear Status Register command (50_{16}).

After a reset, the status register is set to "80_{16}."

Each bit in this register is explained below.

Write state machine (WSM) status (SR7)

After power-on, the write state machine (WSM) status is set to "1".

The write state machine (WSM) status indicates the operating status of the device, as for output on the RY/$\overline{BY}$ pin. This status bit is set to "0" during auto write or auto erase operation and is set to "1" upon completion of these operations.

Erase status (SR5)

The erase status informs the operating status of auto erase operation to the CPU. When an erase error occurs, it is set to "1".

The erase status is reset to "0" when cleared.

Program status (SR4)

The program status informs the operating status of auto write operation to the CPU. When a write error occurs, it is set to “1”.

The program status is reset to “0” when cleared.

When an erase command is in error (which occurs if the command entered after the block erase command (20_{16}) is not the confirmation command ($D0_{16}$), both the program status and erase status (SR5) are set to “1”.

When the program status or erase status = “1”, only the following flash commands will be accepted: Read Array, Read Status Register, and Clear Status Register.

Also, in one of the following cases, both SR4 and SR5 are set to “1” (command sequence error):

(1) When the valid command is not entered correctly

(2) When the data entered in the second bus cycle of lock bit program (77_{16}/$D0_{16}$), block erase (20_{16}/$D0_{16}$), or erase all unlock blocks ($A7_{16}$/$D0_{16}$) is not the $D0_{16}$ or FF_{16}. However, if FF_{16} is entered, read array is assumed and the command that has been set up in the first bus cycle is canceled.

Block status after program (SR3)

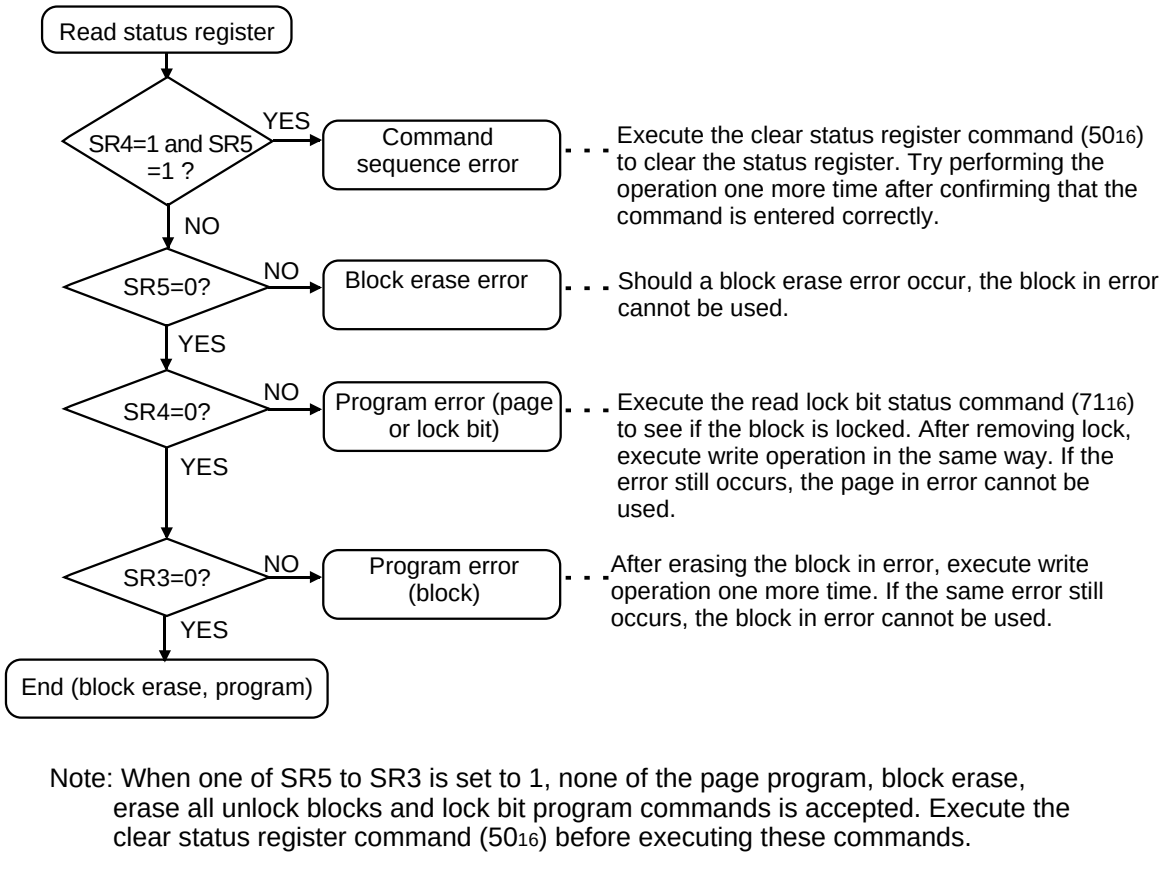
If excessive data is written (phenomenon whereby the memory cell becomes depressed which results in data not being read correctly), “1” is set for the program status after-program at the end of the page write operation. In other words, when writing ends successfully, “80_{16}” is output; when writing fails, “90_{16}” is output; and when excessive data is written, “88_{16}” is output.

Table 1.26.2. Definition of each bit in status register

Each bit of SRD	Status name	Definition	
		"1"	"0"
SR7 (bit7)	Write state machine (WSM) status	Ready	Busy
SR6 (bit6)	Reserved	-	-
SR5 (bit5)	Erase status	Terminated in error	Terminated normally
SR4 (bit4)	Program status	Terminated in error	Terminated normally
SR3 (bit3)	Block status after program	Terminated in error	Terminated normally
SR2 (bit2)	Reserved	-	-
SR1 (bit1)	Reserved	-	-
SR0 (bit0)	Reserved	-	-

Full Status Check

By performing full status check, it is possible to know the execution results of erase and program operations. Figure 1.26.8 shows a full status check flowchart and the action to be taken when each error occurs.



Note: When one of SR5 to SR3 is set to 1, none of the page program, block erase, erase all unlock blocks and lock bit program commands is accepted. Execute the clear status register command (50_{16}) before executing these commands.

Figure 1.26.8. Full status check flowchart and remedial procedure for errors

Functions To Inhibit Rewriting Flash Memory Version

To prevent the contents of the flash memory version from being read out or rewritten easily, the device incorporates a ROM code protect function for use in parallel I/O mode and an ID code check function for use in standard serial I/O mode.

ROM code protect function

The ROM code protect function is used to prohibit reading out or modifying the contents of the flash memory during parallel I/O mode and is set by using the ROM code protect control address register (0FFFFF16). Figure 1.27.1 shows the ROM code protect control address (0FFFFF16). (This address exists in the user ROM area.)

If one of the pair of ROM code protect bits is set to 0, ROM code protect is turned on, so that the contents of the flash memory version are protected against readout and modification. ROM code protect is implemented in two levels. If level 2 is selected, the flash memory is protected even against readout by a shipment inspection LSI tester, etc. When an attempt is made to select both level 1 and level 2, level 2 is selected by default.

If both of the two ROM code protect reset bits are set to “00,” ROM code protect is turned off, so that the contents of the flash memory version can be read out or modified. Once ROM code protect is turned on, the contents of the ROM code protect reset bits cannot be modified in parallel I/O mode. Use the serial I/O or some other mode to rewrite the contents of the ROM code protect reset bits.

ROM code protect control address

b7	b6	b5	b4	b3	b2	b1	b0
						1	1

Symbol: ROMCP
Address: 0FFFFF16
When reset: FF16

Bit symbol	Bit name	Function
Reserved bit		Always set this bit to 1.
ROMCP2	ROM code protect level 2 set bit (Note 1, 2)	b3 b2 00: Protect enabled 01: Protect enabled 10: Protect enabled 11: Protect disabled
ROMCR	ROM code protect reset bit (Note 3)	b5 b4 00: Protect removed 01: Protect set bit effective 10: Protect set bit effective 11: Protect set bit effective
ROMCP1	ROM code protect level 1 set bit (Note 1)	b7 b6 00: Protect enabled 01: Protect enabled 10: Protect enabled 11: Protect disabled

Note 1: When ROM code protect is turned on, the on-chip flash memory is protected against readout or modification in parallel input/output mode.
Note 2: When ROM code protect level 2 is turned on, ROM code readout by a shipment inspection LSI tester, etc. also is inhibited.
Note 3: The ROM code protect reset bits can be used to turn off ROM code protect level 1 and ROM code protect level 2. However, since these bits cannot be changed in parallel input/output mode, they need to be rewritten in serial input/output or some other mode.

Figure 1.27.1. ROM code protect control address

ID Code Check Function

Use this function in standard serial I/O mode. When the contents of the flash memory are not blank, the ID code sent from the peripheral unit is compared with the ID code written in the flash memory to see if they match. If the ID codes do not match, the commands sent from the peripheral unit are not accepted. The ID code consists of 8-bit data, the areas of which, beginning with the first byte, are 0FFFDF16, 0FFFE316, 0FFFEB16, 0FFFEF16, 0FFFF316, 0FFFF716, and 0FFFFB16. Write a program which has had the ID code preset at these addresses to the flash memory.

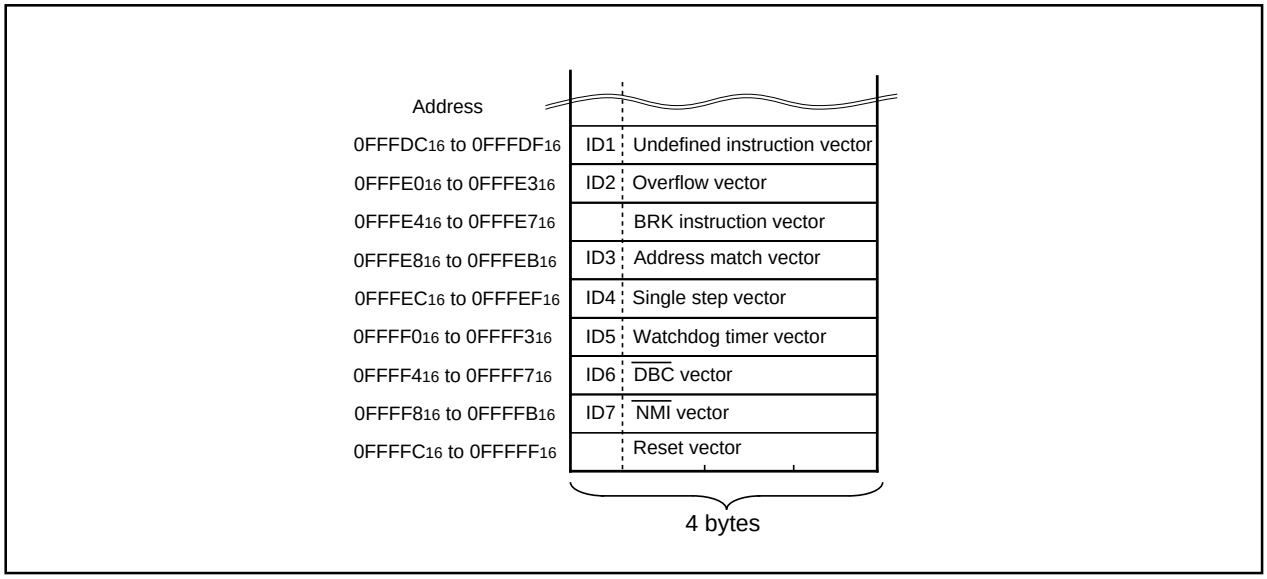


Figure 1.27.2. ID code store addresses

Parallel I/O Mode

The parallel I/O mode inputs and outputs the software commands, addresses and data needed to operate (read, program, erase, etc.) the internal flash memory. This I/O is parallel.
Use an exclusive programer supporting M16C/62A (flash memory version).
Refer to the instruction manual of each programer maker for the details of use.

User ROM and Boot ROM Areas

In parallel I/O mode, the user ROM and boot ROM areas shown in Figure 1.25.1 can be rewritten. Both areas of flash memory can be operated on in the same way.
Program and block erase operations can be performed in the user ROM area. The user ROM area and its blocks are shown in Figure 1.25.1.
The boot ROM area is 8 Kbytes in size. In parallel I/O mode, it is located at addresses $0FE000_{16}$ through $0FFFFF_{16}$. Make sure program and block erase operations are always performed within this address range. (Access to any location outside this address range is prohibited.)
In the boot ROM area, an erase block operation is applied to only one 8 Kbyte block. The boot ROM area has had a standard serial I/O mode control program stored in it when shipped from the Mitsubishi factory. Therefore, using the device in standard serial input/output mode, you do not need to write to the boot ROM area.

Pin functions (Flash memory standard serial I/O mode)

Pin	Name	I/O	Description
VCC,VSS	Power input		Apply program/erase protection voltage to Vcc pin and 0 V to Vss pin.
CNVSS	CNVSS	I	Connect to Vcc pin.
$\overline{\text{RESET}}$	Reset input	I	Reset input pin. While reset is "L" level, a 20 cycle or longer clock must be input to XIN pin.
XIN	Clock input	I	Connect a ceramic resonator or crystal oscillator between XIN and XOUT pins. To input an externally generated clock, input it to XIN pin and open XOUT pin.
XOUT	Clock output	O	
BYTE	BYTE	I	Connect this pin to VSS or VCC.
AVCC, AVSS	Analog power supply input		Connect AVSS to VSS and AVCC to VCC, respectively.
VREF	Reference voltage input	I	Enter the reference voltage for AD from this pin.
P00 to P07	Input port P0	I	Input "H" or "L" level signal or open.
P10 to P17	Input port P1	I	Input "H" or "L" level signal or open.
P20 to P27	Input port P2	I	Input "H" or "L" level signal or open.
P30 to P37	Input port P3	I	Input "H" or "L" level signal or open.
P40 to P47	Input port P4	I	Input "H" or "L" level signal or open.
P51 to P54, P56, P57	Input port P5	I	Input "H" or "L" level signal or open.
P50	$\overline{\text{CE}}$ input	I	Input "H" level signal.
P55	$\overline{\text{EPM}}$ input	I	Input "L" level signal.
P60 to P63	Input port P6	I	Input "H" or "L" level signal or open.
P64	BUSY output	O	Standard serial I/O mode 1: BUSY signal output pin Standard serial I/O mode 2: Monitors the boot program operation check signal output pin.
P65	SCLK input	I	Standard serial I/O mode 1: Serial clock input pin Standard serial I/O mode 2: Input "L".
P66	RxD input	I	Serial data input pin
P67	TxD output	O	Serial data output pin
P70 to P77	Input port P7	I	Input "H" or "L" level signal or open.
P80 to P84, P86, P87	Input port P8	I	Input "H" or "L" level signal or open.
P85	$\overline{\text{NMI}}$ input	I	Connect this pin to Vcc.
P90 to P97	Input port P9	I	Input "H" or "L" level signal or open.
P100 to P107	Input port P10	I	Input "H" or "L" level signal or open.

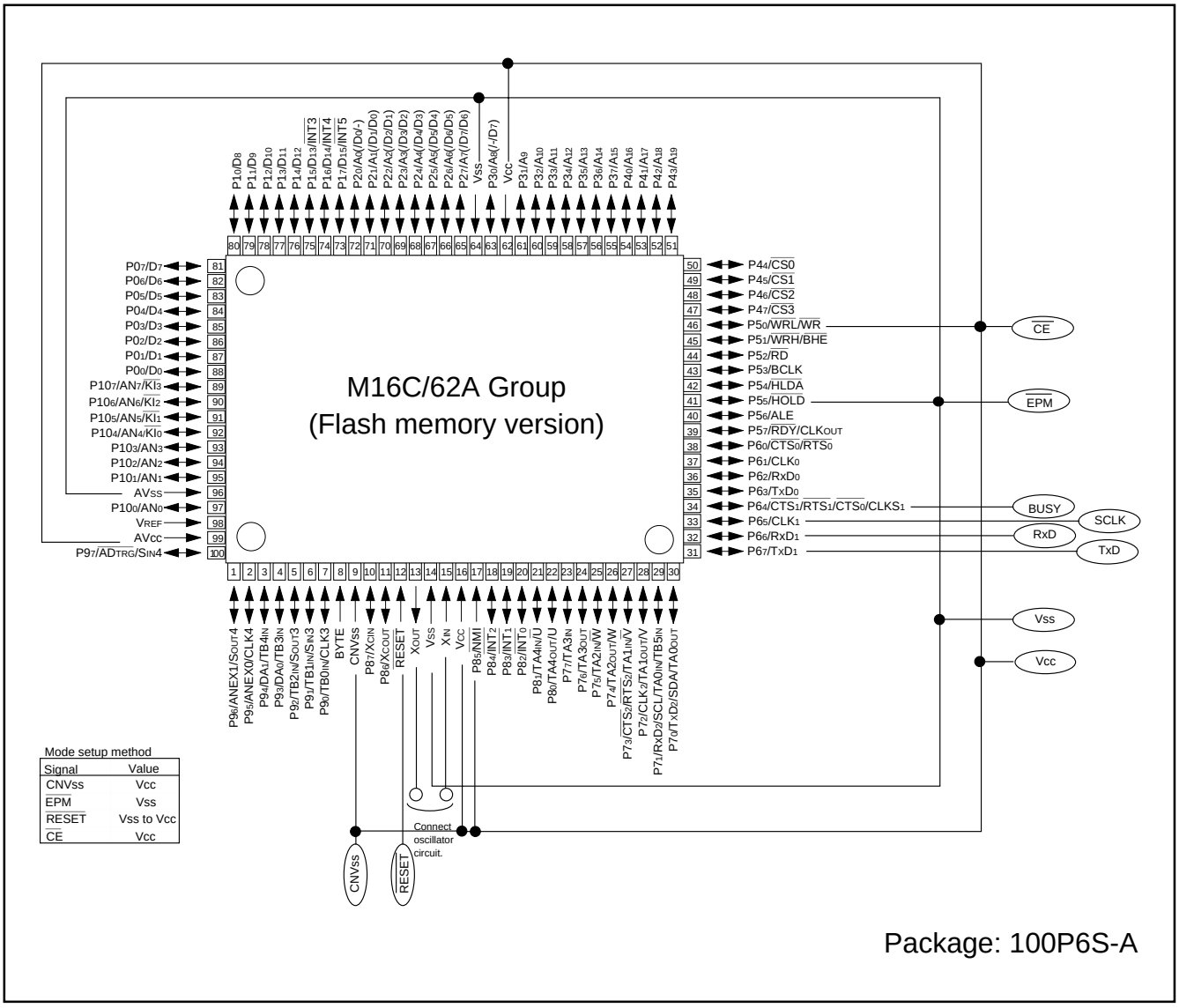


Figure 1.29.1. Pin connections for serial I/O mode (1)

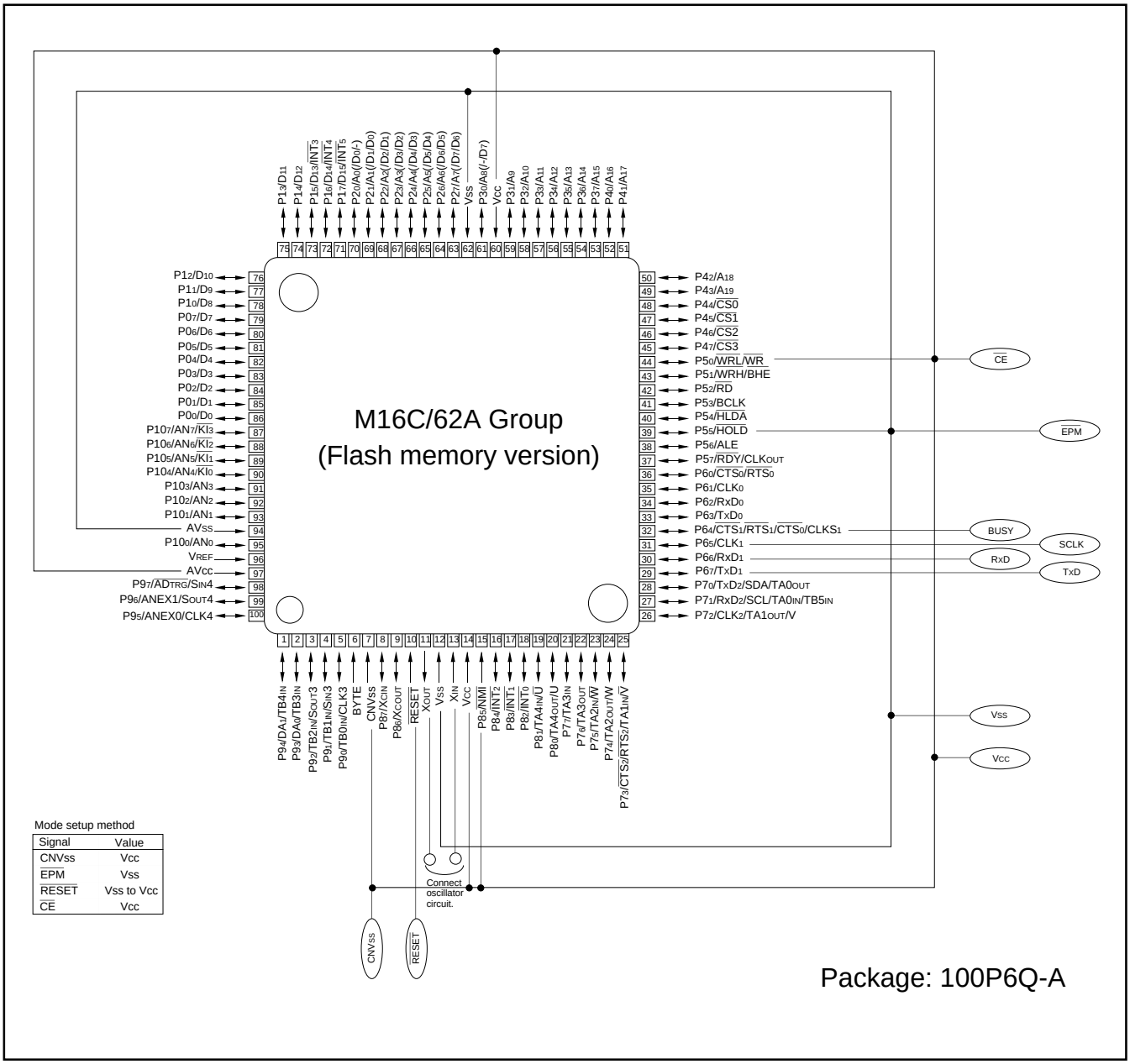


Figure 1.29.2. Pin connections for serial I/O mode (2)

Standard serial I/O mode

The standard serial I/O mode inputs and outputs the software commands, addresses and data needed to operate (read, program, erase, etc.) the internal flash memory. This I/O is serial. There are actually two standard serial I/O modes: mode 1, which is clock synchronized, and mode 2, which is asynchronized. Both modes require a purpose-specific peripheral unit.

The standard serial I/O mode is different from the parallel I/O mode in that the CPU controls flash memory rewrite (uses the CPU's rewrite mode), rewrite data input and so forth. It is started when the reset is released, which is done when the $P5_0$ ($\overline{CE}$) pin is "H" level, the $P5_5$ ($\overline{EPM}$) pin "L" level and the CNVss pin "H" level. (In the ordinary command mode, set CNVss pin to "L" level.)
This control program is written in the boot ROM area when the product is shipped from Mitsubishi. Accordingly, make note of the fact that the standard serial I/O mode cannot be used if the boot ROM area is rewritten in the parallel I/O mode. Figures 1.29.1 and 1.29.2 show the pin connections for the standard serial I/O mode. Serial data I/O uses UART1 and transfers the data serially in 8-bit units. Standard serial I/O switches between mode 1 (clock synchronized) and mode 2 (clock asynchronized) according to the level of CLK_1 pin when the reset is released.

To use standard serial I/O mode 1 (clock synchronized), set the CLK_1 pin to "H" level and release the reset. The operation uses the four UART1 pins CLK_1, RxD_1, TxD_1 and RTS_1 (BUSY). The CLK_1 pin is the transfer clock input pin through which an external transfer clock is input. The TxD_1 pin is for CMOS output. The RTS_1 (BUSY) pin outputs an "L" level when ready for reception and an "H" level when reception starts.

To use standard serial I/O mode 2 (clock asynchronized), set the CLK_1 pin to "L" level and release the reset. The operation uses the two UART1 pins RxD_1 and TxD_1.

In the standard serial I/O mode, only the user ROM area indicated in Figure 1.29.19 can be rewritten. The boot ROM cannot.
In the standard serial I/O mode, a 7-byte ID code is used. When there is data in the flash memory, commands sent from the peripheral unit are not accepted unless the ID code matches.

Overview of standard serial I/O mode 1 (clock synchronized)

In standard serial I/O mode 1, software commands, addresses and data are input and output between the MCU and peripheral units (serial programer, etc.) using 4-wire clock-synchronized serial I/O (UART1). Standard serial I/O mode 1 is engaged by releasing the reset with the $P6_5$ (CLK_1) pin "H" level.

In reception, software commands, addresses and program data are synchronized with the rise of the transfer clock that is input to the CLK_1 pin, and are then input to the MCU via the RxD_1 pin. In transmission, the read data and status are synchronized with the fall of the transfer clock, and output from the TxD_1 pin.

The TxD_1 pin is for CMOS output. Transfer is in 8-bit units with LSB first.

When busy, such as during transmission, reception, erasing or program execution, the RTS_1 (BUSY) pin is "H" level. Accordingly, always start the next transfer after the RTS_1 (BUSY) pin is "L" level.

Also, data and status registers in memory can be read after inputting software commands. Status, such as the operating state of the flash memory or whether a program or erase operation ended successfully or not, can be checked by reading the status register. Here following are explained software commands, status registers, etc.

Software Commands

Table 1.29.1 lists software commands. In the standard serial I/O mode 1, erase operations, programs and reading are controlled by transferring software commands via the RxD_1 pin. Software commands are explained here below.

Table 1.29.1. Software commands (Standard serial I/O mode 1)

	Control command	1st byte transfer	2nd byte	3rd byte	4th byte	5th byte	6th byte		When ID is not verified
1	Page read	FF_{16}	Address (middle)	Address (high)	Data output	Data output	Data output	Data output to 259th byte	Not acceptable
2	Page program	41_{16}	Address (middle)	Address (high)	Data input	Data input	Data input	Data input to 259th byte	Not acceptable
3	Block erase	20_{16}	Address (middle)	Address (high)	$D0_{16}$				Not acceptable
4	Erase all unlocked blocks	$A7_{16}$	$D0_{16}$						Not acceptable
5	Read status register	70_{16}	SRD output	SRD1 output					Acceptable
6	Clear status register	50_{16}							Not acceptable
7	Read lock bit status	71_{16}	Address (middle)	Address (high)	Lock bit data output				Not acceptable
8	Lock bit program	77_{16}	Address (middle)	Address (high)	$D0_{16}$				Not acceptable
9	Lock bit enable	$7A_{16}$							Not acceptable
10	Lock bit disable	75_{16}							Not acceptable
11	ID check function	$F5_{16}$	Address (low)	Address (middle)	Address (high)	ID size	ID1	To ID7	Acceptable
12	Download function	FA_{16}	Size (low)	Size (high)	Check-sum	Data input	To required number of times		Not acceptable
13	Version data output function	FB_{16}	Version data output	Version data output	Version data output	Version data output	Version data output	Version data output to 9th byte	Acceptable
14	Boot ROM area output function	FC_{16}	Address (middle)	Address (high)	Data output	Data output	Data output	Data output to 259th byte	Not acceptable
15	Read check data	FD_{16}	Check data (low)	Check data (high)					Not acceptable

Note 1: Shading indicates transfer from flash memory microcomputer to peripheral unit. All other data is transferred from the peripheral unit to the flash memory microcomputer.

Note 2: SRD refers to status register data. SRD1 refers to status register 1 data.

Note 3: All commands can be accepted when the flash memory is totally blank.

Page Read Command

This command reads the specified page (256 bytes) in the flash memory sequentially one byte at a time. Execute the page read command as explained here following.

(1) Transfer the "FF_{16}" command code with the 1st byte.

(2) Transfer addresses A_8 to A_{15} and A_{16} to A_{23} with the 2nd and 3rd bytes respectively.

(3) From the 4th byte onward, data (D_0–D_7) for the page (256 bytes) specified with addresses A_8 to A_{23} will be output sequentially from the smallest address first in sync with the fall of the clock.

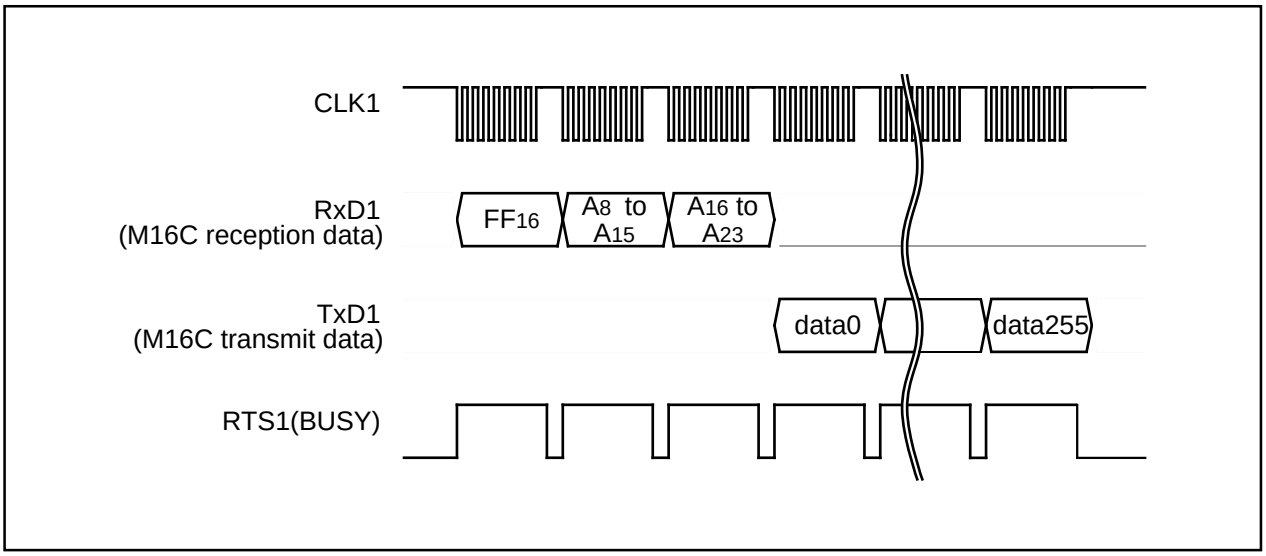


Figure 1.29.3. Timing for page read

Read Status Register Command

This command reads status information. When the "70_{16}" command code is sent with the 1st byte, the contents of the status register (SRD) specified with the 2nd byte and the contents of status register 1 (SRD1) specified with the 3rd byte are read.

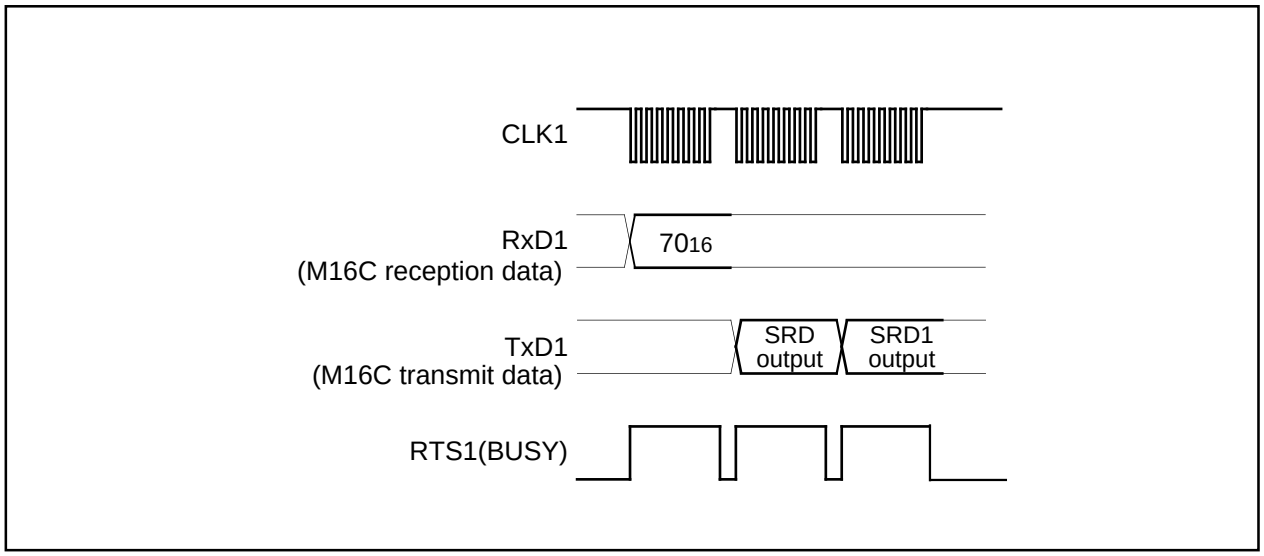


Figure 1.29.4. Timing for reading the status register

Clear Status Register Command

This command clears the bits (SR3–SR5) which are set when the status register operation ends in error. When the "50_{16}" command code is sent with the 1st byte, the aforementioned bits are cleared. When the clear status register operation ends, the RTS_1 (BUSY) signal changes from the "H" to the "L" level.

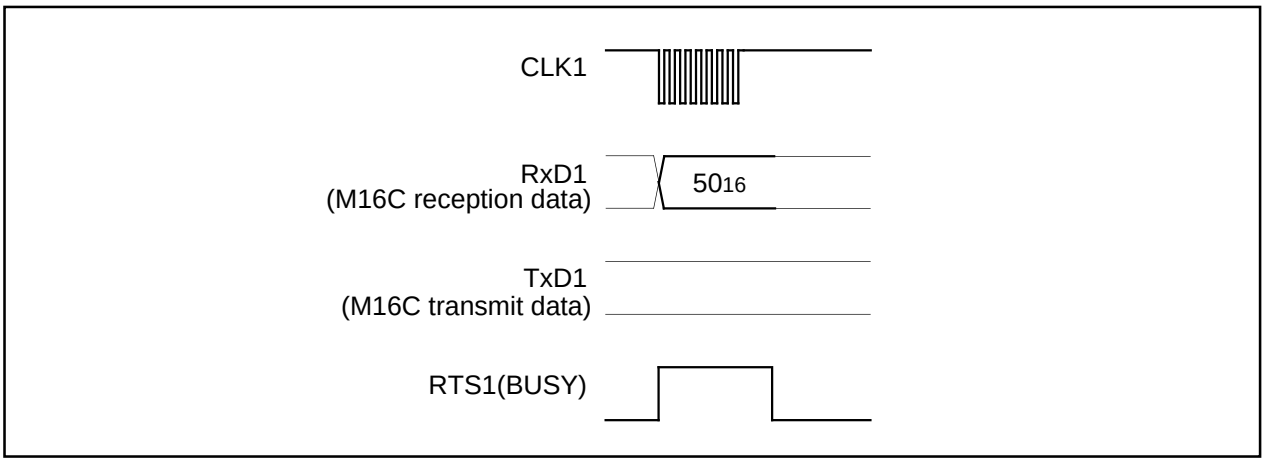


Figure 1.29.5. Timing for clearing the status register

Page Program Command

This command writes the specified page (256 bytes) in the flash memory sequentially one byte at a time. Execute the page program command as explained here following.

(1) Transfer the "41_{16}" command code with the 1st byte.
(2) Transfer addresses A_8 to A_{15} and A_{16} to A_{23} with the 2nd and 3rd bytes respectively.
(3) From the 4th byte onward, as write data (D_0–D_7) for the page (256 bytes) specified with addresses A_8 to A_{23} is input sequentially from the smallest address first, that page is automatically written.

When reception setup for the next 256 bytes ends, the RTS_1 (BUSY) signal changes from the "H" to the "L" level. The result of the page program can be known by reading the status register. For more information, see the section on the status register.

Each block can be write-protected with the lock bit. For more information, see the section on the data protection function. Additional writing is not allowed with already programmed pages.

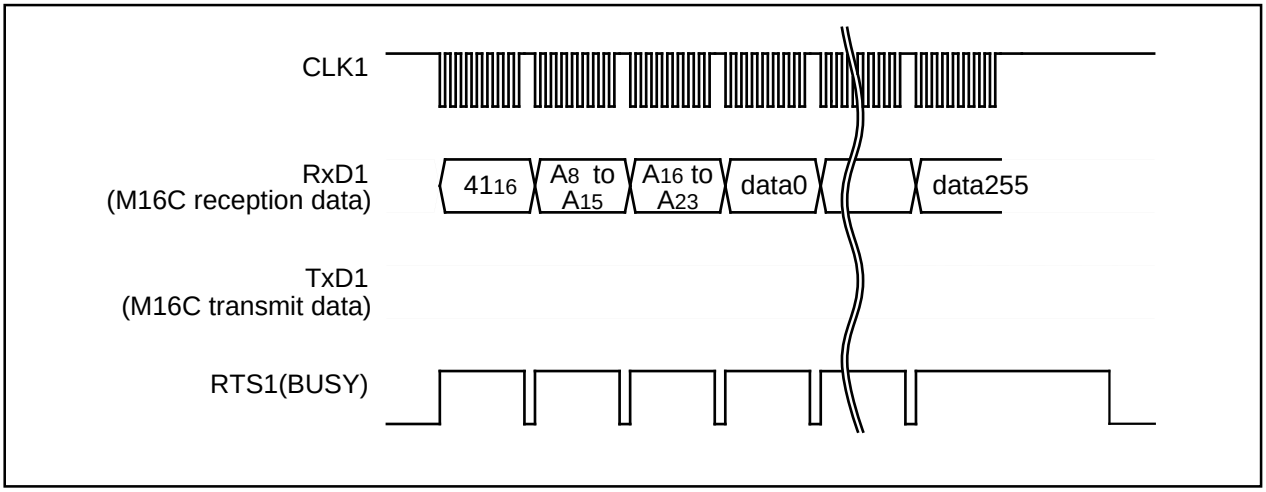


Figure 1.29.6. Timing for the page program

Block Erase Command

This command erases the data in the specified block. Execute the block erase command as explained here following.

(1) Transfer the "20_{16}" command code with the 1st byte.

(2) Transfer addresses A_8 to A_{15} and A_{16} to A_{23} with the 2nd and 3rd bytes respectively.

(3) Transfer the verify command code "$D0_{16}$" with the 4th byte. With the verify command code, the erase operation will start for the specified block in the flash memory. Write the highest address of the specified block for addresses A_8 to A_{23}.

When block erasing ends, the RTS_1 (BUSY) signal changes from the "H" to the "L" level. After block erase ends, the result of the block erase operation can be known by reading the status register. For more information, see the section on the status register.

Each block can be erase-protected with the lock bit. For more information, see the section on the data protection function.

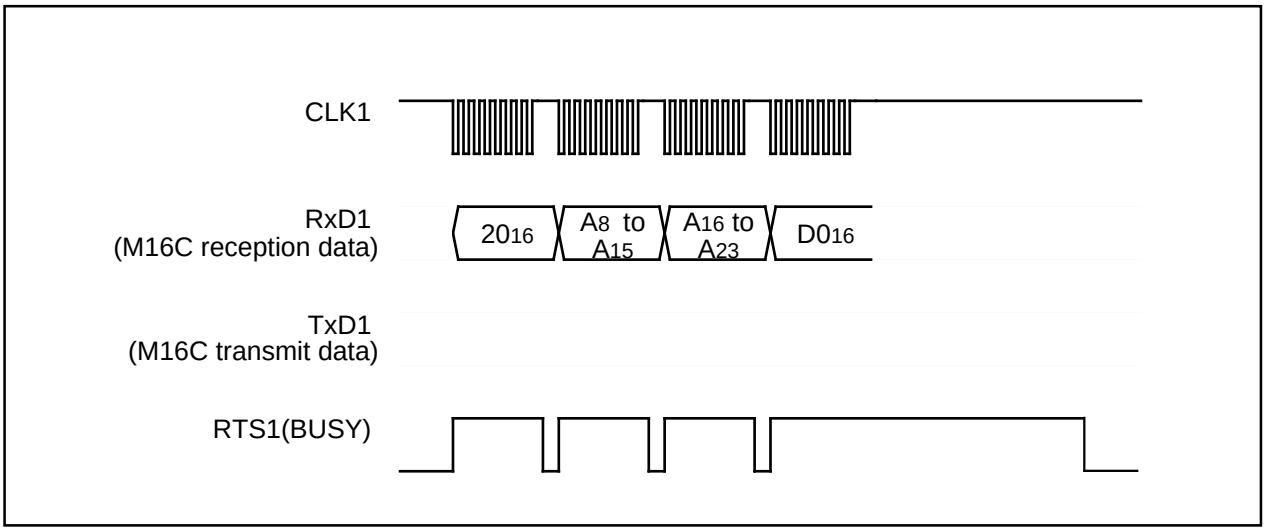


Figure 1.29.7. Timing for block erasing

Erase All Unlocked Blocks Command

This command erases the content of all blocks. Execute the erase all unlocked blocks command as explained here following.

(1) Transfer the "A7$_{16}$" command code with the 1st byte.

(2) Transfer the verify command code "D0$_{16}$" with the 2nd byte. With the verify command code, the erase operation will start and continue for all blocks in the flash memory.

When block erasing ends, the RTS$_1$ (BUSY) signal changes from the "H" to the "L" level. The result of the erase operation can be known by reading the status register. Each block can be erase-protected with the lock bit. For more information, see the section on the data protection function.

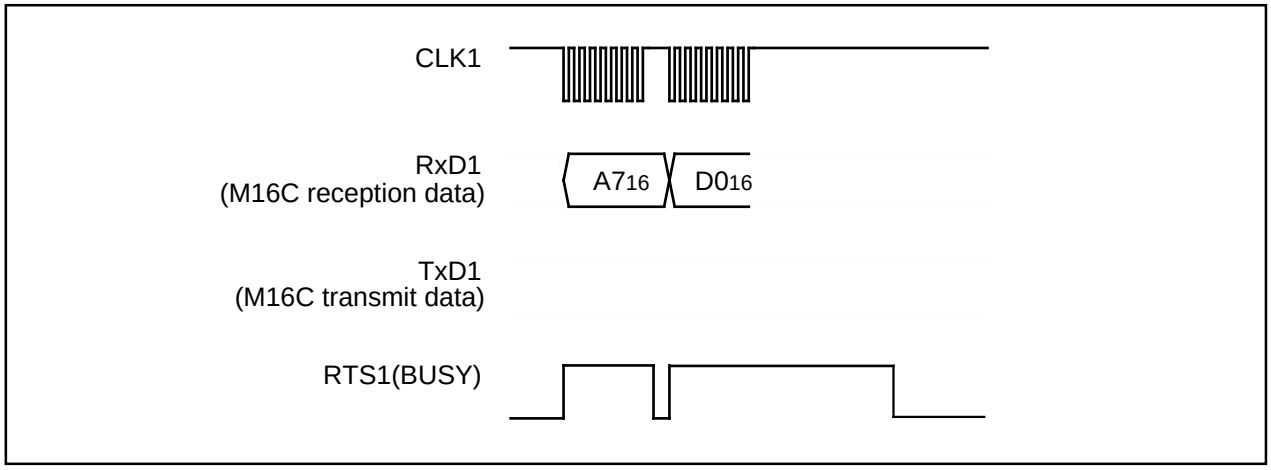


Figure 1.29.8. Timing for erasing all unlocked blocks

Lock Bit Program Command

This command writes "0" (lock) for the lock bit of the specified block. Execute the lock bit program command as explained here following.

(1) Transfer the "77$_{16}$" command code with the 1st byte.

(2) Transfer addresses A$_8$ to A$_{15}$ and A$_{16}$ to A$_{23}$ with the 2nd and 3rd bytes respectively.

(3) Transfer the verify command code "D0$_{16}$" with the 4th byte. With the verify command code, "0" is written for the lock bit of the specified block. Write the highest address of the specified block for addresses A$_8$ to A$_{23}$.

When writing ends, the RTS$_1$ (BUSY) signal changes from the "H" to the "L" level. Lock bit status can be read with the read lock bit status command. For information on the lock bit function, reset procedure and so on, see the section on the data protection function.

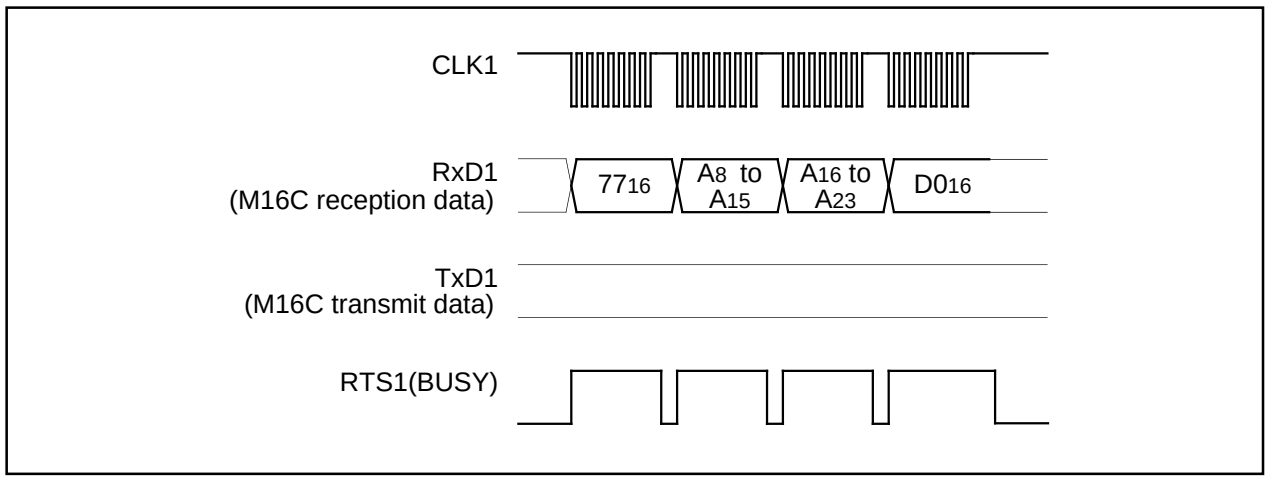


Figure 1.29.9. Timing for the lock bit program

Read Lock Bit Status Command

This command reads the lock bit status of the specified block. Execute the read lock bit status command as explained here following.

(1) Transfer the "71_{16}" command code with the 1st byte.

(2) Transfer addresses A_8 to A_{15} and A_{16} to A_{23} with the 2nd and 3rd bytes respectively.

(3) The lock bit data of the specified block is output with the 4th byte. The lock bit data is the 6th bit(D_6) of the output data. Write the highest address of the specified block for addresses A_8 to A_{23}.

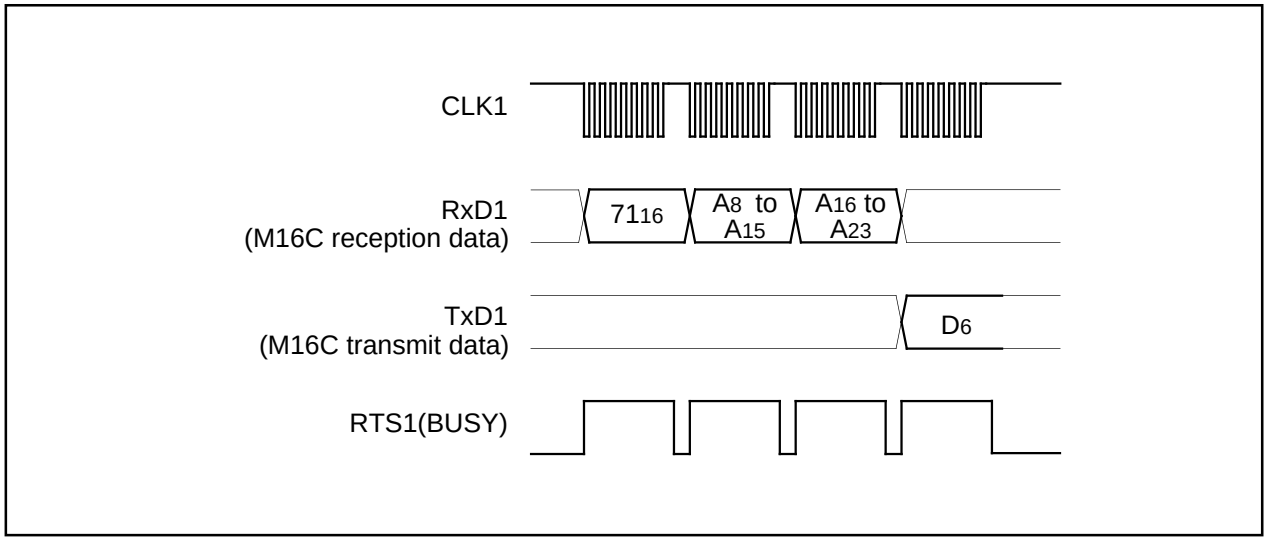


Figure 1.29.10. Timing for reading lock bit status

Lock Bit Enable Command

This command enables the lock bit in blocks whose bit was disabled with the lock bit disable command. The command code "$7A_{16}$" is sent with the 1st byte of the serial transmission. This command only enables the lock bit function; it does not set the lock bit itself.

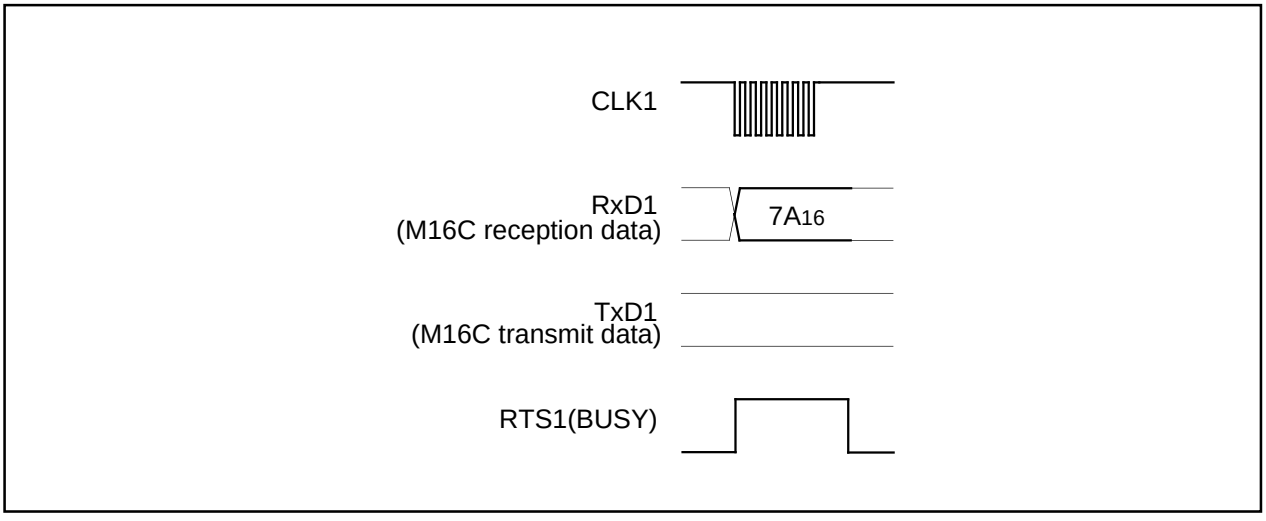


Figure 1.29.11. Timing for enabling the lock bit

Lock Bit Disable Command

This command disables the lock bit. The command code "75_{16}" is sent with the 1st byte of the serial transmission. This command only disables the lock bit function; it does not set the lock bit itself. However, if an erase command is executed after executing the lock bit disable command, "0" (locked) lock bit data is set to "1" (unlocked) after the erase operation ends. In any case, after the reset is cancelled, the lock bit is enabled.

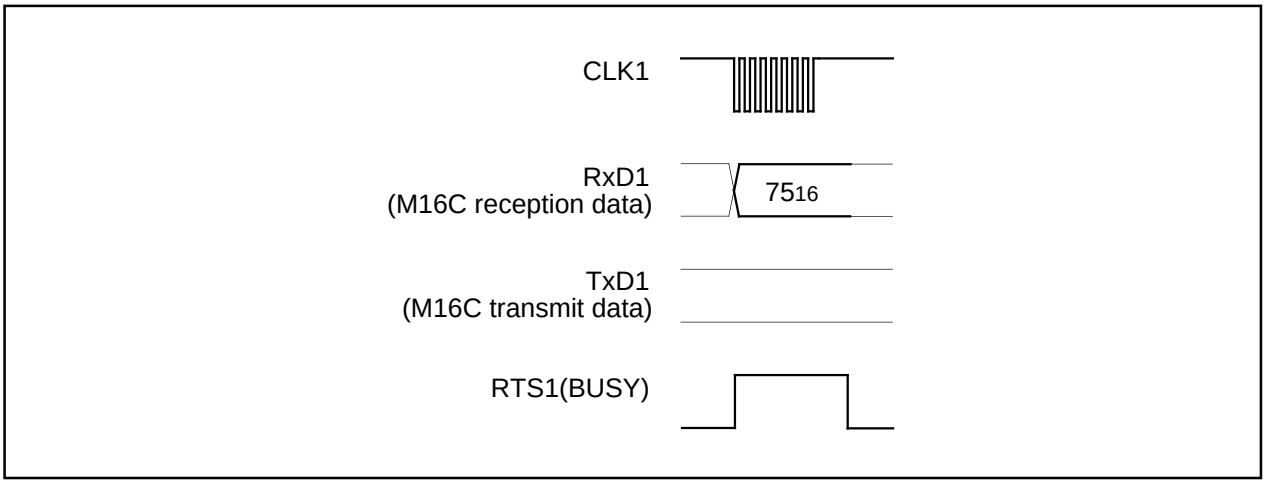


Figure 1.29.12. Timing for disabling the lock bit

Download Command

This command downloads a program to the RAM for execution. Execute the download command as explained here following.

(1) Transfer the "FA_{16}" command code with the 1st byte.
(2) Transfer the program size with the 2nd and 3rd bytes.
(3) Transfer the check sum with the 4th byte. The check sum is added to all data sent with the 5th byte onward.
(4) The program to execute is sent with the 5th byte onward.

When all data has been transmitted, if the check sum matches, the downloaded program is executed. The size of the program will vary according to the internal RAM.

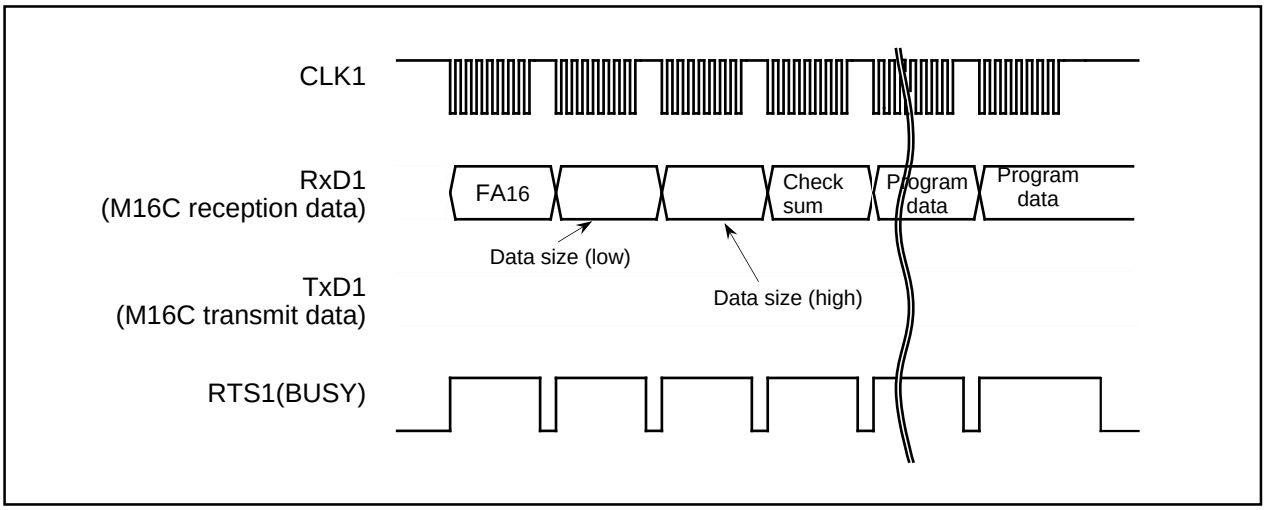


Figure 1.29.13. Timing for download

Version Information Output Command

This command outputs the version information of the control program stored in the boot area. Execute the version information output command as explained here following.

(1) Transfer the "FB_{16}" command code with the 1st byte.

(2) The version information will be output from the 2nd byte onward. This data is composed of 8 ASCII code characters.

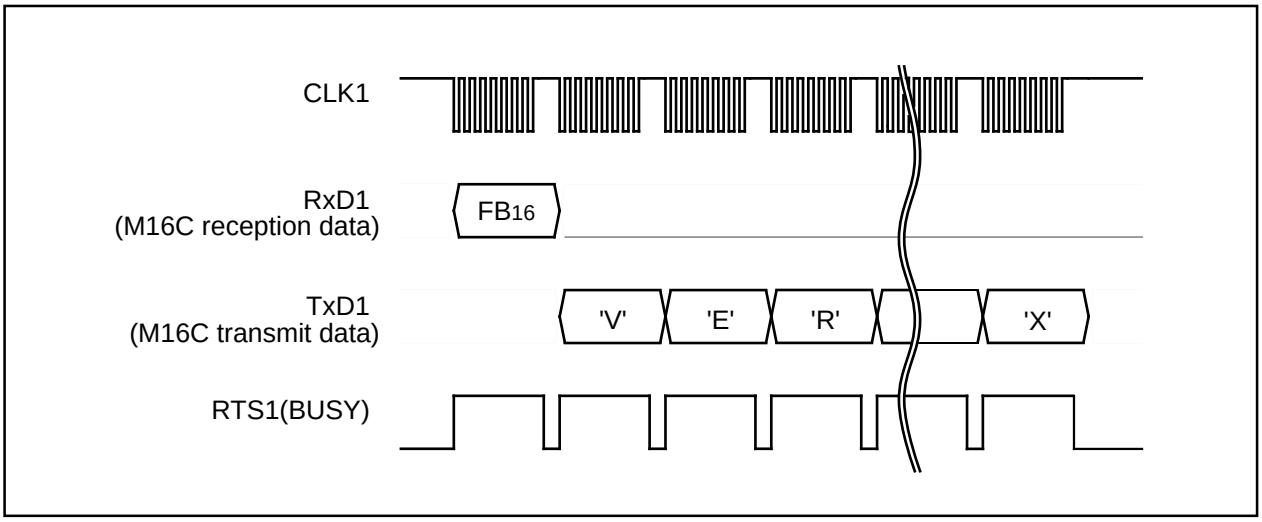


Figure 1.29.14. Timing for version information output

Boot ROM Area Output Command

This command outputs the control program stored in the boot ROM area in one page blocks (256 bytes). Execute the boot ROM area output command as explained here following.

(1) Transfer the "FC_{16}" command code with the 1st byte.

(2) Transfer addresses A_8 to A_{15} and A_{16} to A_{23} with the 2nd and 3rd bytes respectively.

(3) From the 4th byte onward, data (D_0–D_7) for the page (256 bytes) specified with addresses A_8 to A_{23} will be output sequentially from the smallest address first, in sync with the fall of the clock.

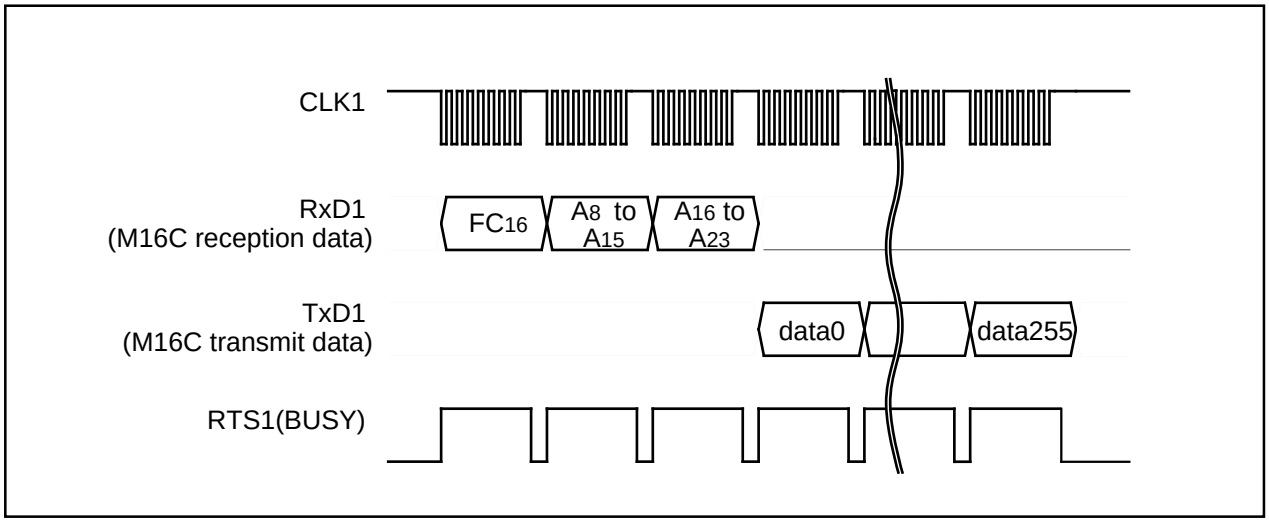


Figure 1.29.15. Timing for boot ROM area output

ID Check

This command checks the ID code. Execute the boot ID check command as explained here following.

(1) Transfer the "F5$_{16}$" command code with the 1st byte.

(2) Transfer addresses A$_0$ to A$_7$, A$_8$ to A$_{15}$ and A$_{16}$ to A$_{23}$ of the 1st byte of the ID code with the 2nd, 3rd and 4th bytes respectively.

(3) Transfer the number of data sets of the ID code with the 5th byte.

(4) The ID code is sent with the 6th byte onward, starting with the 1st byte of the code.

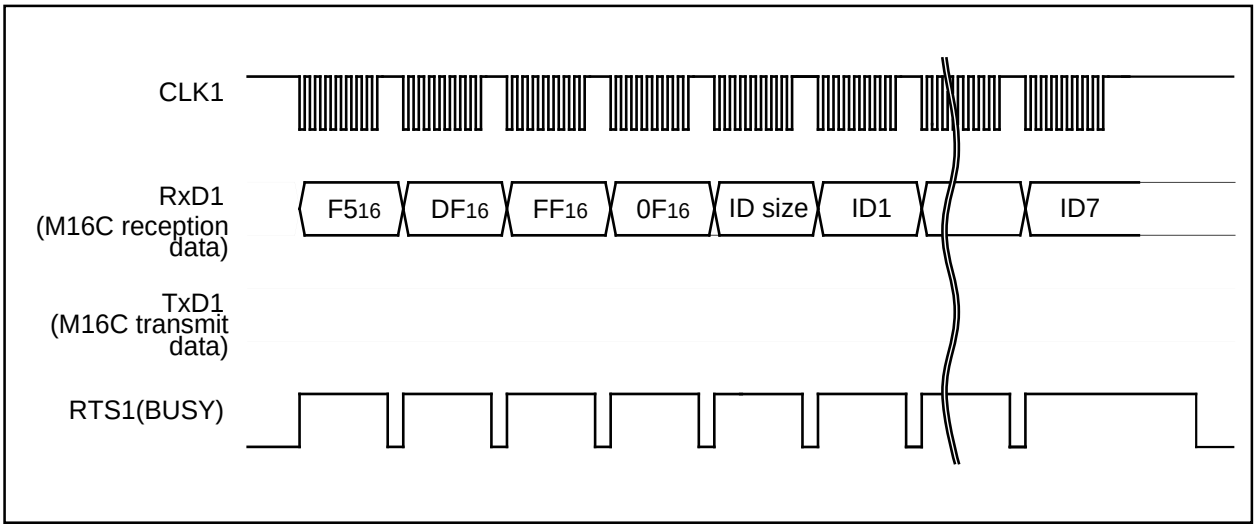


Figure 1.29.16. Timing for the ID check

ID Code

When the flash memory is not blank, the ID code sent from the peripheral units and the ID code written in the flash memory are compared to see if they match. If the codes do not match, the command sent from the peripheral units is not accepted. An ID code contains 8 bits of data. Area is, from the 1st byte, addresses 0FFFDF$_{16}$, 0FFFE3$_{16}$, 0FFFEB$_{16}$, 0FFFEF$_{16}$, 0FFFF3$_{16}$, 0FFFF7$_{16}$ and 0FFFFB$_{16}$. Write a program into the flash memory, which already has the ID code set for these addresses.

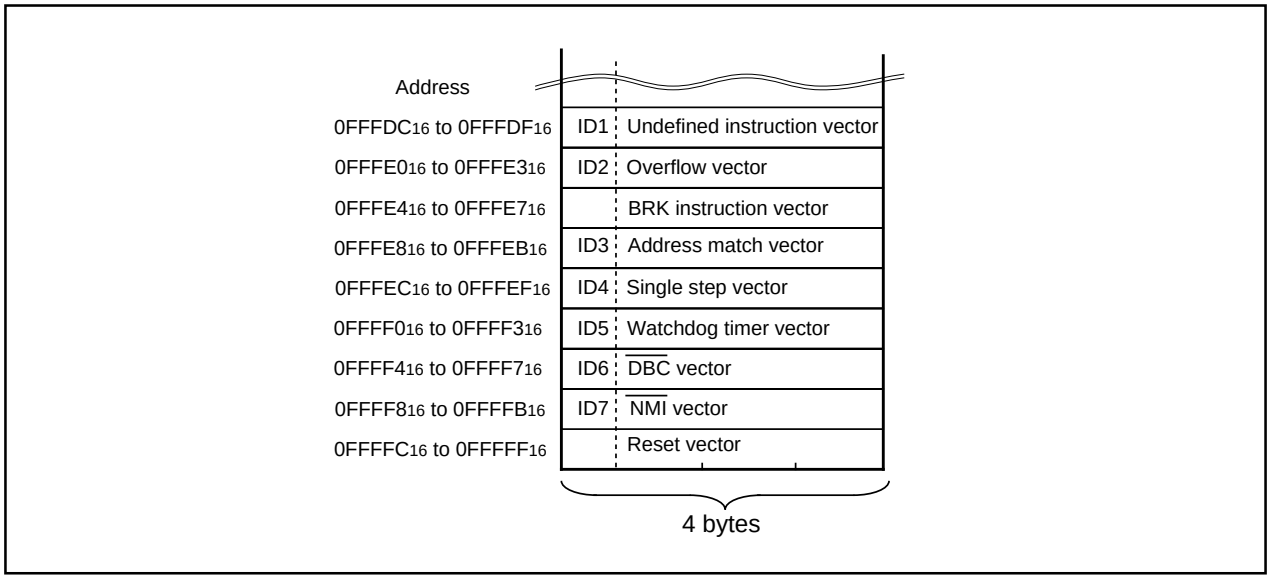


Figure 1.29.17. ID code storage addresses

Read Check Data

This command reads the check data that confirms that the write data, which was sent with the page program command, was successfully received.

(1) Transfer the "FD16" command code with the 1st byte.

(2) The check data (low) is received with the 2nd byte and the check data (high) with the 3rd.

To use this read check data command, first execute the command and then initialize the check data. Next, execute the page program command the required number of times. After that, when the read check command is executed again, the check data for all of the read data that was sent with the page program command during this time is read. The check data is the result of CRC operation of write data.

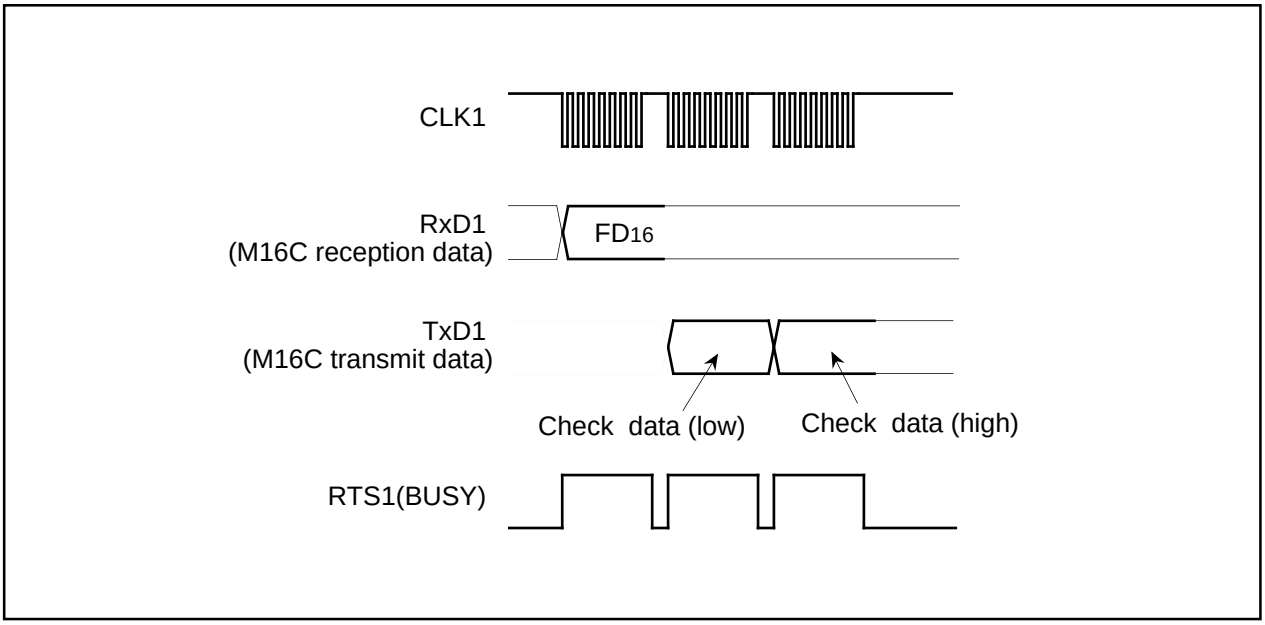


Figure 1.29.18. Timing for the read check data

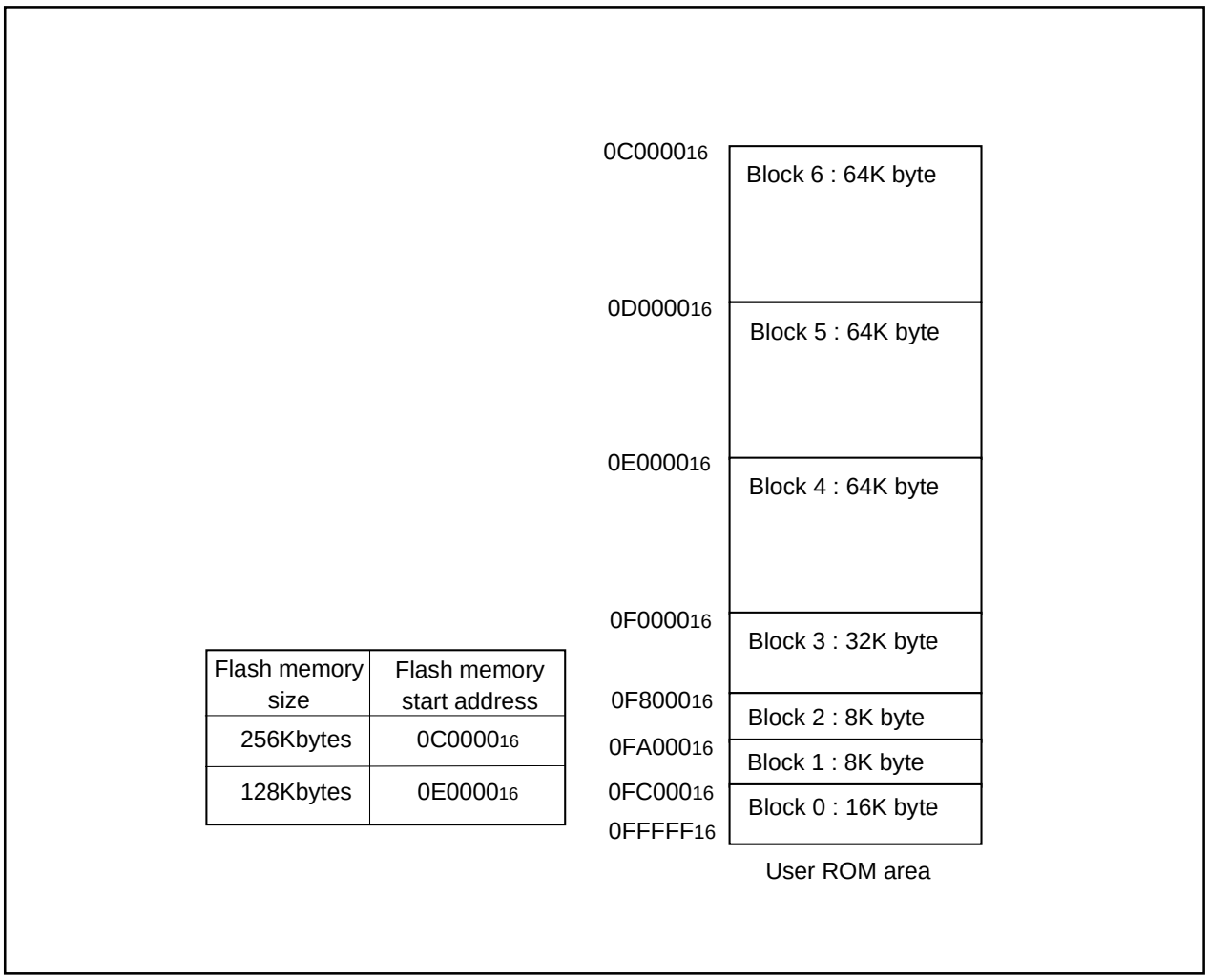
Data Protection (Block Lock)

Each of the blocks in Figure 1.29.19 have a nonvolatile lock bit that specifies protection (block lock) against erasing/writing. A block is locked (writing “0” for the lock bit) with the lock bit program command. Also, the lock bit of any block can be read with the read lock bit status command.

Block lock disable/enable is determined by the status of the lock bit itself and execution status of the lock bit disable and lock enable bit commands.

(1) After the reset has been cancelled and the lock bit enable command executed, the specified block can be locked/unlocked using the lock bit (lock bit data). Blocks with a “0” lock bit data are locked and cannot be erased or written in. On the other hand, blocks with a “1” lock bit data are unlocked and can be erased or written in.

(2) After the lock bit disable command has been executed, all blocks are unlocked regardless of lock bit data status and can be erased or written in. In this case, lock bit data that was “0” (locked) before the block was erased is set to “1” (unlocked) after erasing, therefore the block is actually unlocked with the lock bit.



Flash memory size	Flash memory start address
256Kbytes	0C000016
128Kbytes	0E000016

Figure 1.29.19. Blocks in the user area

Status Register (SRD)

The status register indicates operating status of the flash memory and status such as whether an erase operation or a program ended successfully or in error. It can be read by writing the read status register command (70_{16}). Also, the status register is cleared by writing the clear status register command (50_{16}). Table 1.29.2 gives the definition of each status register bit. After clearing the reset, the status register outputs "80_{16}".

Table 1.29.2. Status register (SRD)

SRD0 bits	Status name	Definition	
		"1"	"0"
SR7 (bit7)	Write state machine (WSM) status	Ready	Busy
SR6 (bit6)	Reserved	-	-
SR5 (bit5)	Erase status	Terminated in error	Terminated normally
SR4 (bit4)	Program status	Terminated in error	Terminated normally
SR3 (bit3)	Block status after program	Terminated in error	Terminated normally
SR2 (bit2)	Reserved	-	-
SR1 (bit1)	Reserved	-	-
SR0 (bit0)	Reserved	-	-

Write State Machine (WSM) Status (SR7)

The write state machine (WSM) status indicates the operating status of the flash memory. When power is turned on, "1" (ready) is set for it. The bit is set to "0" (busy) during an auto write or auto erase operation, but it is set back to "1" when the operation ends.

Erase Status (SR5)

The erase status reports the operating status of the auto erase operation. If an erase error occurs, it is set to "1". When the erase status is cleared, it is set to "0".

Program Status (SR4)

The program status reports the operating status of the auto write operation. If a write error occurs, it is set to "1". When the program status is cleared, it is set to "0".

Block Status After Program (SR3)

If excessive data is written (phenomenon whereby the memory cell becomes depressed which results in data not being read correctly), "1" is set for the block status after-program at the end of the page write operation. In other words, when writing ends successfully, "80_{16}" is output; when writing fails, "90_{16}" is output; and when excessive data is written, "88_{16}" is output.

If "1" is written for any of the SR5, SR4 or SR3 bits, the page program, block erase, erase all unlocked blocks and lock bit program commands are not accepted. Before executing these commands, execute the clear status register command (50_{16}) and clear the status register.

Status Register 1 (SRD1)

Status register 1 indicates the status of serial communications, results from ID checks and results from check sum comparisons. It can be read after the SRD by writing the read status register command (70_{16}). Also, status register 1 is cleared by writing the clear status register command (50_{16}).

Table 1.29.3 gives the definition of each status register 1 bit. "00_{16}" is output when power is turned ON and the flag status is maintained even after the reset.

Table 1.29.3. Status register 1 (SRD1)

SRD1 bits	Status name	Definition	
		"1"	"0"
SR15 (bit7)	Boot update completed bit	Update completed	Not update
SR14 (bit6)	Reserved	-	-
SR13 (bit5)	Reserved	-	-
SR12 (bit4)	Check sum match bit	Match	Mismatch
SR11 (bit3) SR10 (bit2)	ID check completed bits	00 Not verified 01 Verification mismatch 10 Reserved 11 Verified	
SR9 (bit1)	Data receive time out	Time out	Normal operation
SR8 (bit0)	Reserved	-	-

Boot Update Completed Bit (SR15)

This flag indicates whether the control program was downloaded to the RAM or not, using the download function.

Check Sum Match Bit (SR12)

This flag indicates whether the check sum matches or not when a program, is downloaded for execution using the download function.

ID Check Completed Bits (SR11 and SR10)

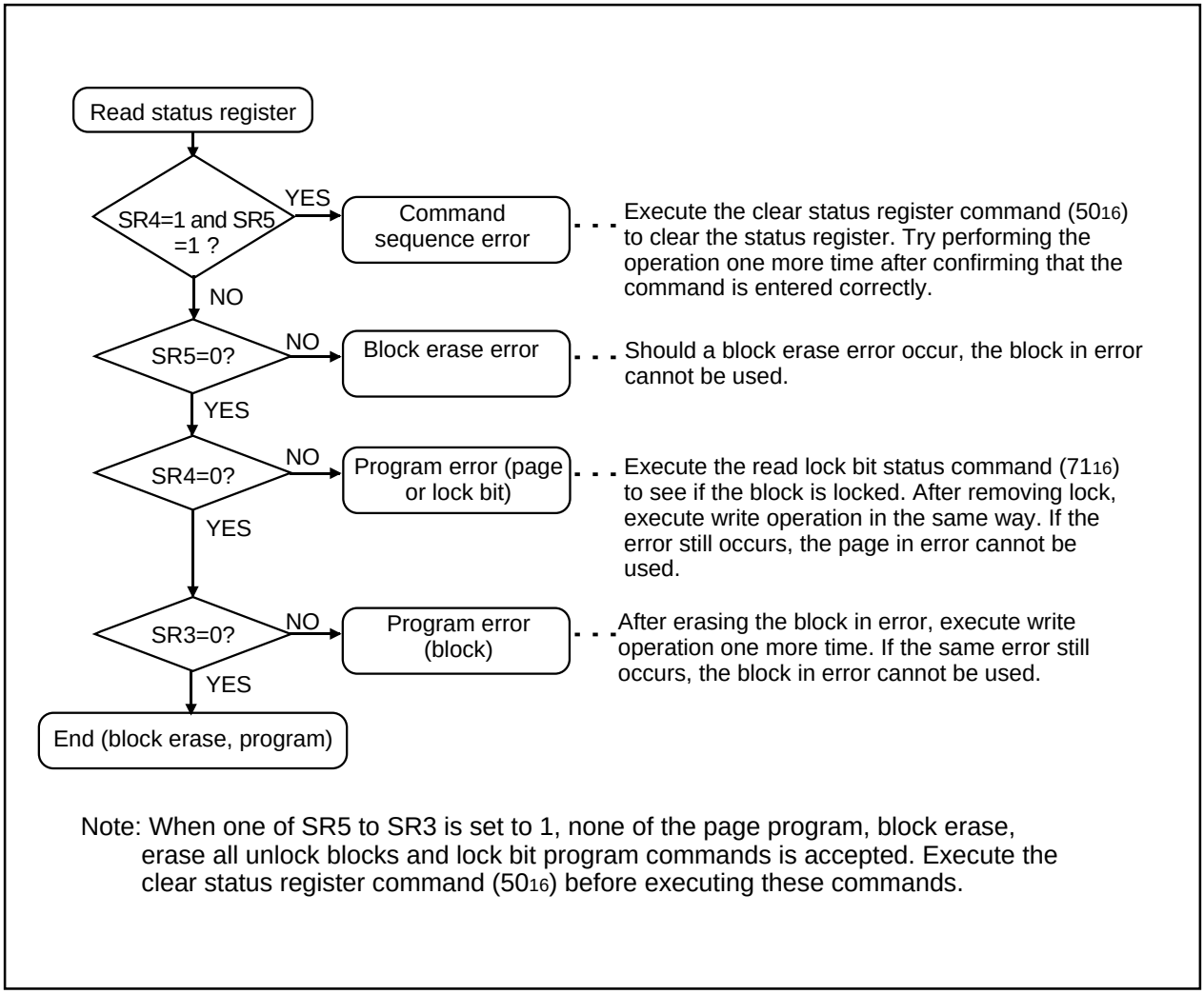
These flags indicate the result of ID checks. Some commands cannot be accepted without an ID check.

Data Receive Time Out (SR9)

This flag indicates when a time out error is generated during data reception. If this flag is attached during data reception, the received data is discarded and the microcomputer returns to the command wait state.

Full Status Check

Results from executed erase and program operations can be known by running a full status check. Figure 1.29.20 shows a flowchart of the full status check and explains how to remedy errors which occur.

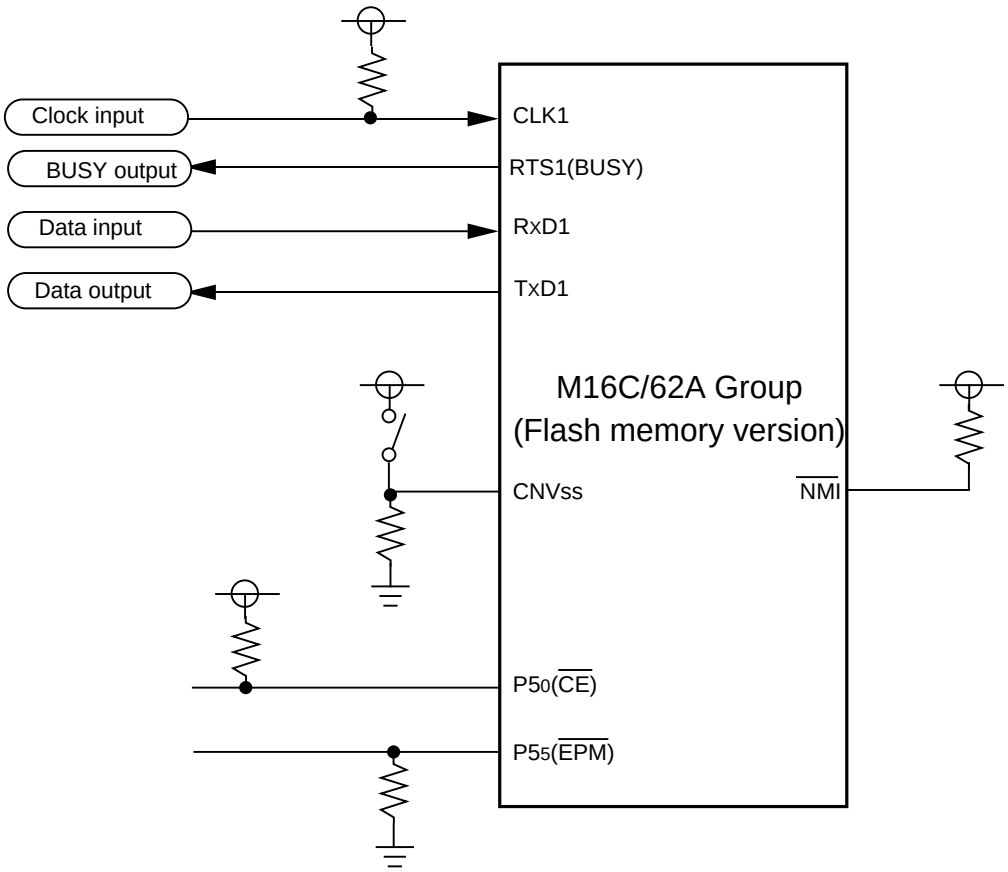


Note: When one of SR5 to SR3 is set to 1, none of the page program, block erase, erase all unlock blocks and lock bit program commands is accepted. Execute the clear status register command (5016) before executing these commands.

Figure 1.29.20. Full status check flowchart and remedial procedure for errors

Example Circuit Application for The Standard Serial I/O Mode 1

The below figure shows a circuit application for the standard serial I/O mode 1. Control pins will vary according to programmer, therefore see the peripheral unit manual for more information.



(1) Control pins and external circuitry will vary according to peripheral unit. For more information, see the peripheral unit manual.
(2) In this example, the microprocessor mode and standard serial I/O mode are switched via a switch.

Figure 1.29.21. Example circuit application for the standard serial I/O mode 1

Overview of standard serial I/O mode 2 (clock asynchronized)

In standard serial I/O mode 2, software commands, addresses and data are input and output between the MCU and peripheral units (serial programer, etc.) using 2-wire clock-asynchronized serial I/O (UART1). Standard serial I/O mode 2 is engaged by releasing the reset with the $P6_5$ (CLK_1) pin "L" level.

The TxD_1 pin is for CMOS output. Data transfer is in 8-bit units with LSB first, 1 stop bit and parity OFF.

After the reset is released, connections can be established at 9,600 bps when initial communications (Figure 1.29.22) are made with a peripheral unit. However, this requires a main clock with a minimum 2 MHz input oscillation frequency. Baud rate can also be changed from 9,600 bps to 19,200, 38,400 or 57,600 bps by executing software commands. However, communication errors may occur because of the oscillation frequency of the main clock. If errors occur, change the main clock's oscillation frequency and the baud rate.

After executing commands from a peripheral unit that requires time to erase and write data, as with erase and program commands, allow a sufficient time interval or execute the read status command and check how processing ended, before executing the next command.

Data and status registers in memory can be read after transmitting software commands. Status, such as the operating state of the flash memory or whether a program or erase operation ended successfully or not, can be checked by reading the status register. Here following are explained initial communications with peripheral units, how frequency is identified and software commands.

Initial communications with peripheral units

After the reset is released, the bit rate generator is adjusted to 9,600 bps to match the oscillation frequency of the main clock, by sending the code as prescribed by the protocol for initial communications with peripheral units (Figure 1.29.22).

(1) Transmit "$B0_{16}$" from a peripheral unit. If the oscillation frequency input by the main clock is 10 or 16 MHz, the MCU with internal flash memory outputs the "$B0_{16}$" check code. If the oscillation frequency is anything other than 10 or 16 MHz, the MCU does not output anything.

(2) Transmit "00_{16}" from a peripheral unit 16 times. (The MCU with internal flash memory sets the bit rate generator so that "00_{16}" can be successfully received.)

(3) The MCU with internal flash memory outputs the "$B0_{16}$" check code and initial communications end successfully *1. Initial communications must be transmitted at a speed of 9,600 bps and a transfer interval of a minimum 15 ms. Also, the baud rate at the end of initial communications is 9,600 bps.

*1. If the peripheral unit cannot receive "$B0_{16}$" successfully, change the oscillation frequency of the main clock.

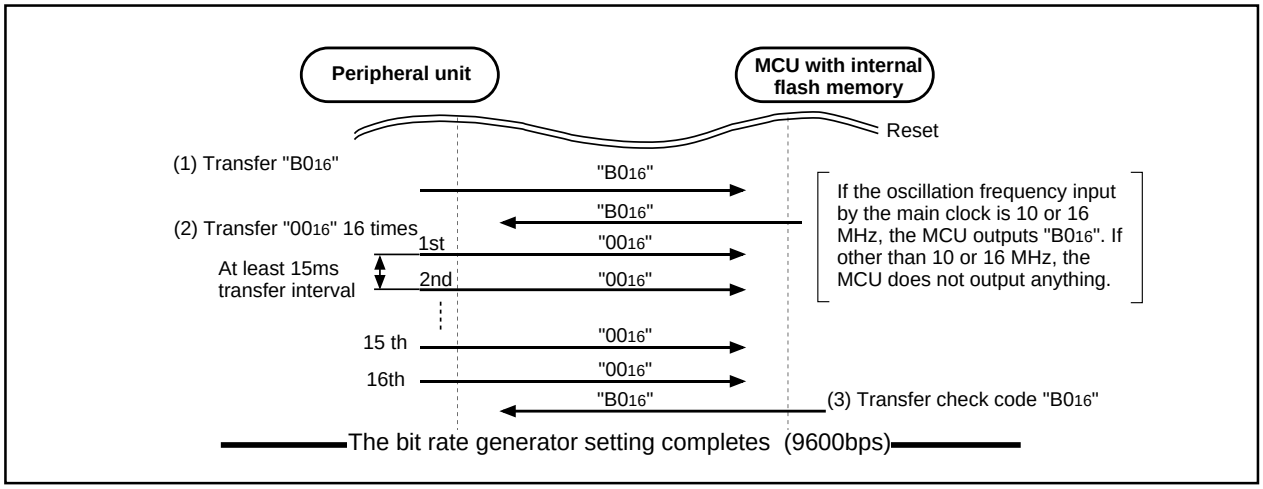


Figure 1.29.22. Peripheral unit and initial communication

How frequency is identified

When "00$_{16}$" data is received 16 times from a peripheral unit at a baud rate of 9,600 bps, the value of the bit rate generator is set to match the operating frequency (2 - 16 MHz). The highest speed is taken from the first 8 transmissions and the lowest from the last 8. These values are then used to calculate the bit rate generator value for a baud rate of 9,600 bps.

Baud rate cannot be attained with some operating frequencies. Table 1.29.4 gives the operation frequency and the baud rate that can be attained for.

Table 1.29.4 Operation frequency and the baud rate

Operation frequency (MHz)	Baud rate 9,600bps	Baud rate 19,200bps	Baud rate 38,400bps	Baud rate 57,600bps
16MHz	√	√	√	√
12MHz	√	√	√	–
11MHz	√	√	√	–
10MHz	√	√	–	√
8MHz	√	√	–	√
7.3728MHz	√	√	√	√
6MHz	√	√	√	–
5MHz	√	√	–	–
4.5MHz	√	√	–	√
4.194304MHz	√	√	√	–
4MHz	√	√	–	–
3.58MHz	√	√	√	√
3MHz	√	√	√	–
2MHz	√	–	–	–

√ : Communications possible
– : Communications not possible

Software Commands

Table 1.29.5 lists software commands. In the standard serial I/O mode 2, erase operations, programs and reading are controlled by transferring software commands via the RxD1 pin. Standard serial I/O mode 2 adds four transmission speed commands - 9,600, 19,200, 38,400 and 57,600 bps - to the software commands of standard serial I/O mode 1. Software commands are explained here below.

Table 1.29.5. Software commands (Standard serial I/O mode 2)

	Control command	1st byte transfer	2nd byte	3rd byte	4th byte	5th byte	6th byte		When ID is not verified
1	Page read	FF_{16}	Address (middle)	Address (high)	Data output	Data output	Data output	Data output to 259th byte	Not acceptable
2	Page program	41_{16}	Address (middle)	Address (high)	Data input	Data input	Data input	Data input to 259th byte	Not acceptable
3	Block erase	20_{16}	Address (middle)	Address (high)	$D0_{16}$				Not acceptable
4	Erase all unlocked blocks	$A7_{16}$	$D0_{16}$						Not acceptable
5	Read status register	70_{16}	SRD output	SRD1 output					Acceptable
6	Clear status register	50_{16}							Not acceptable
7	Read lock bit status	71_{16}	Address (middle)	Address (high)	Lock bit data output				Not acceptable
8	Lock bit program	77_{16}	Address (middle)	Address (high)	$D0_{16}$				Not acceptable
9	Lock bit enable	$7A_{16}$							Not acceptable
10	Lock bit disable	75_{16}							Not acceptable
11	ID check function	$F5_{16}$	Address (low)	Address (middle)	Address (high)	ID size	ID1	To ID7	Acceptable
12	Download function	FA_{16}	Size (low)	Size (high)	Check-sum	Data input	To required number of times		Not acceptable
13	Version data output function	FB_{16}	Version data output	Version data output	Version data output	Version data output	Version data output	Version data output to 9th byte	Acceptable
14	Boot ROM area output function	FC_{16}	Address (middle)	Address (high)	Data output	Data output	Data output	Data output to 259th byte	Not acceptable
15	Read check data	FD_{16}	Check data (low)	Check data (high)					Not acceptable
16	Baud rate 9600	$B0_{16}$	$B0_{16}$						Acceptable
17	Baud rate 19200	$B1_{16}$	$B1_{16}$						Acceptable
18	Baud rate 38400	$B2_{16}$	$B2_{16}$						Acceptable
19	Baud rate 57600	$B3_{16}$	$B3_{16}$						Acceptable

Note 1: Shading indicates transfer from flash memory microcomputer to peripheral unit. All other data is transferred from the peripheral unit to the flash memory microcomputer.

Note 2: SRD refers to status register data. SRD1 refers to status register 1 data.

Note 3: All commands can be accepted when the flash memory is totally blank.

Page Read Command

This command reads the specified page (256 bytes) in the flash memory sequentially one byte at a time. Execute the page read command as explained here following.

(1) Transfer the "FF_{16}" command code with the 1st byte.

(2) Transfer addresses A_8 to A_{15} and A_{16} to A_{23} with the 2nd and 3rd bytes respectively.

(3) From the 4th byte onward, data (D_0–D_7) for the page (256 bytes) specified with addresses A_8 to A_{23} will be output sequentially from the smallest address first.

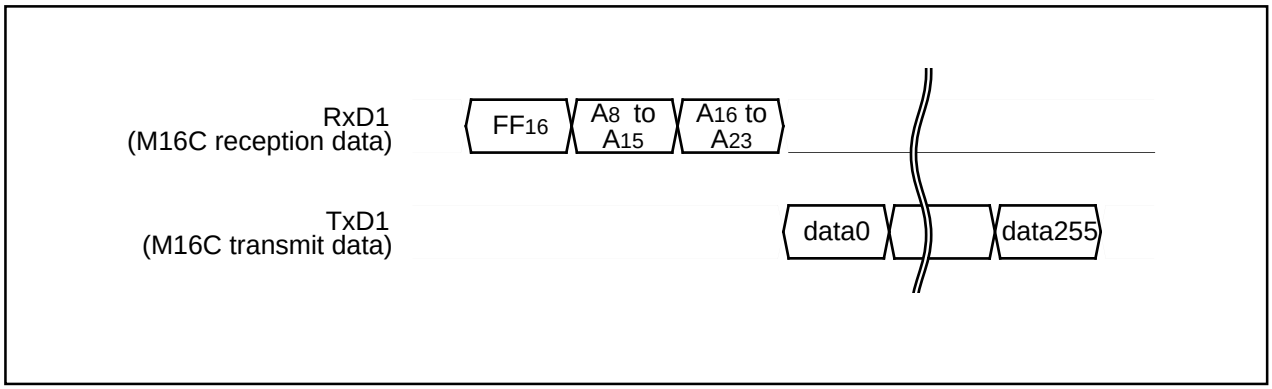


Figure 1.29.23. Timing for page read

Read Status Register Command

This command reads status information. When the "70_{16}" command code is sent with the 1st byte, the contents of the status register (SRD) specified with the 2nd byte and the contents of status register 1 (SRD1) specified with the 3rd byte are read.

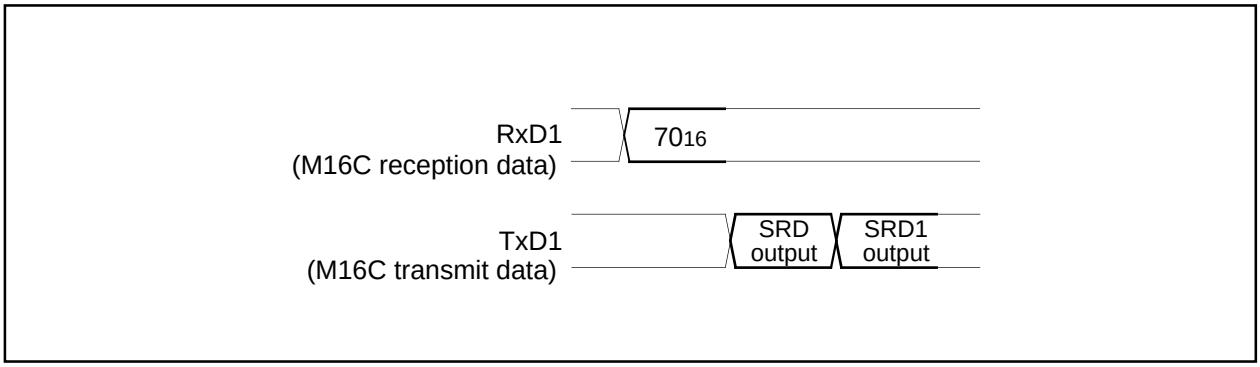


Figure 1.29.24. Timing for reading the status register

Clear Status Register Command

This command clears the bits (SR3–SR5) which are set when the status register operation ends in error. When the "50_{16}" command code is sent with the 1st byte, the aforementioned bits are cleared.

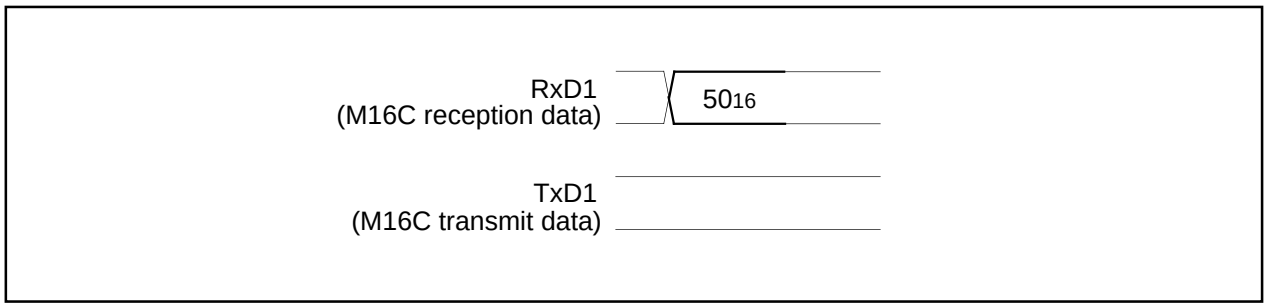


Figure 1.29.25. Timing for clearing the status register

Page Program Command

This command writes the specified page (256 bytes) in the flash memory sequentially one byte at a time. Execute the page program command as explained here following.

(1) Transfer the "41_{16}" command code with the 1st byte.
(2) Transfer addresses A_8 to A_{15} and A_{16} to A_{23} with the 2nd and 3rd bytes respectively.
(3) From the 4th byte onward, as write data (D_0–D_7) for the page (256 bytes) specified with addresses A_8 to A_{23} is input sequentially from the smallest address first, that page is automatically written.

The result of the page program can be known by reading the status register. For more information, see the section on the status register.

Each block can be write-protected with the lock bit. For more information, see the section on the data protection function. Additional writing is not allowed with already programmed pages.

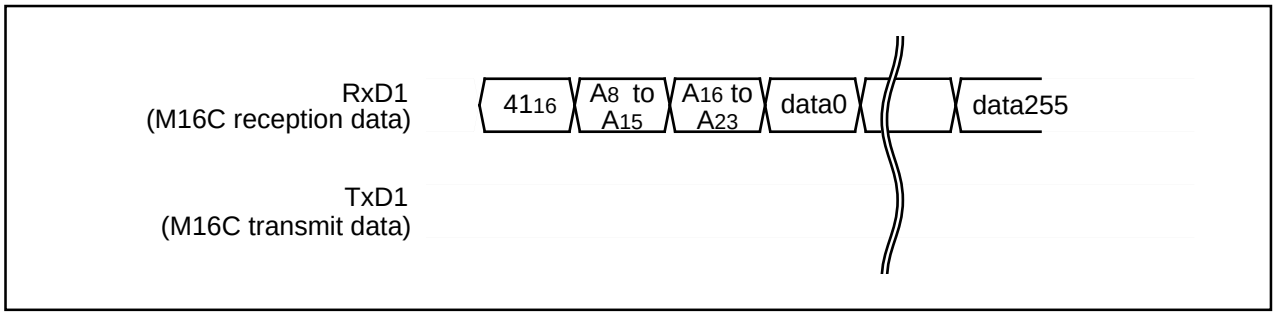


Figure 1.29.26. Timing for the page program

Block Erase Command

This command erases the data in the specified block. Execute the block erase command as explained here following.

(1) Transfer the "20_{16}" command code with the 1st byte.
(2) Transfer addresses A_8 to A_{15} and A_{16} to A_{23} with the 2nd and 3rd bytes respectively.
(3) Transfer the verify command code "$D0_{16}$" with the 4th byte. With the verify command code, the erase operation will start for the specified block in the flash memory. Write the highest address of the specified block for addresses A_8 to A_{23}.

After block erase ends, the result of the block erase operation can be known by reading the status register. For more information, see the section on the status register.
Each block can be erase-protected with the lock bit. For more information, see the section on the data protection function.

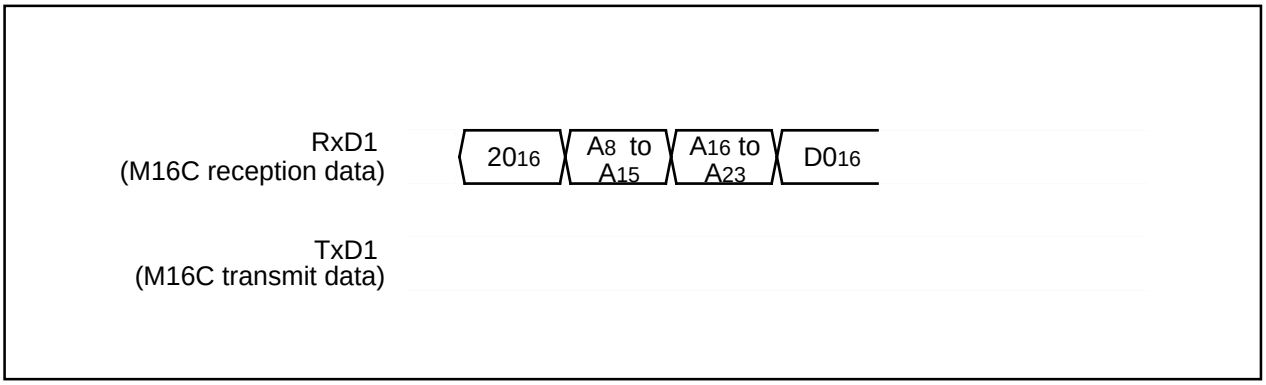


Figure 1.29.7. Timing for block erasing

Erase All Unlocked Blocks Command

This command erases the content of all blocks. Execute the erase all unlocked blocks command as explained here following.

(1) Transfer the "A7$_{16}$" command code with the 1st byte.

(2) Transfer the verify command code "D0$_{16}$" with the 2nd byte. With the verify command code, the erase operation will start and continue for all blocks in the flash memory.

The result of the erase operation can be known by reading the status register. Each block can be erase-protected with the lock bit. For more information, see the section on the data protection function.

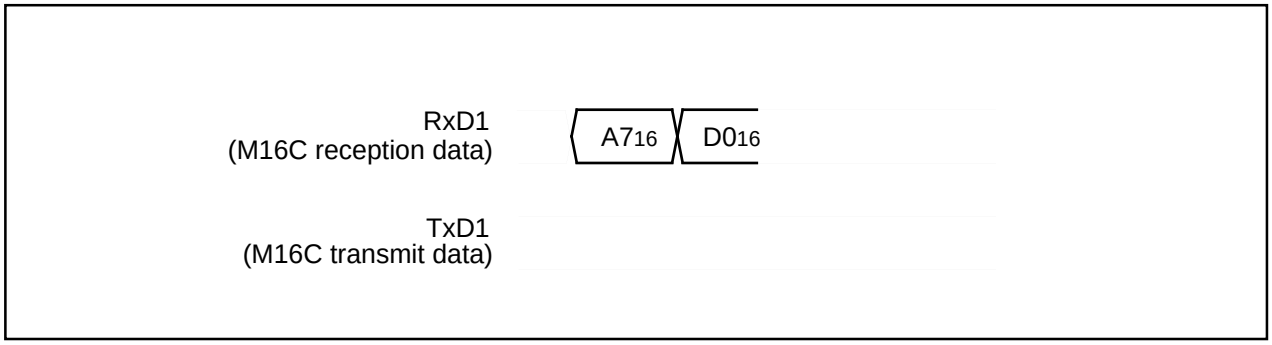


Figure 1.29.28. Timing for erasing all unlocked blocks

Lock Bit Program Command

This command writes "0" (lock) for the lock bit of the specified block. Execute the lock bit program command as explained here following.

(1) Transfer the "77$_{16}$" command code with the 1st byte.

(2) Transfer addresses A$_8$ to A$_{15}$ and A$_{16}$ to A$_{23}$ with the 2nd and 3rd bytes respectively.

(3) Transfer the verify command code "D0$_{16}$" with the 4th byte. With the verify command code, "0" is written for the lock bit of the specified block. Write the highest address of the specified block for addresses A$_8$ to A$_{23}$.

Lock bit status can be read with the read lock bit status command. For information on the lock bit function, reset procedure and so on, see the section on the data protection function.

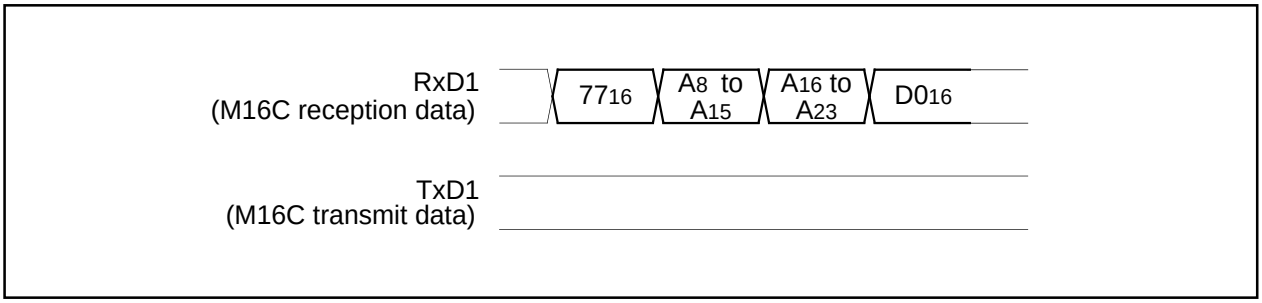


Figure 1.29.29. Timing for the lock bit program

Read Lock Bit Status Command

This command reads the lock bit status of the specified block. Execute the read lock bit status command as explained here following.

(1) Transfer the "71_{16}" command code with the 1st byte.

(2) Transfer addresses A_8 to A_{15} and A_{16} to A_{23} with the 2nd and 3rd bytes respectively.

(3) The lock bit data of the specified block is output with the 4th byte. The lock bit data is the 6th bit(D_6) of the output data. Write the highest address of the specified block for addresses A_8 to A_{23}.

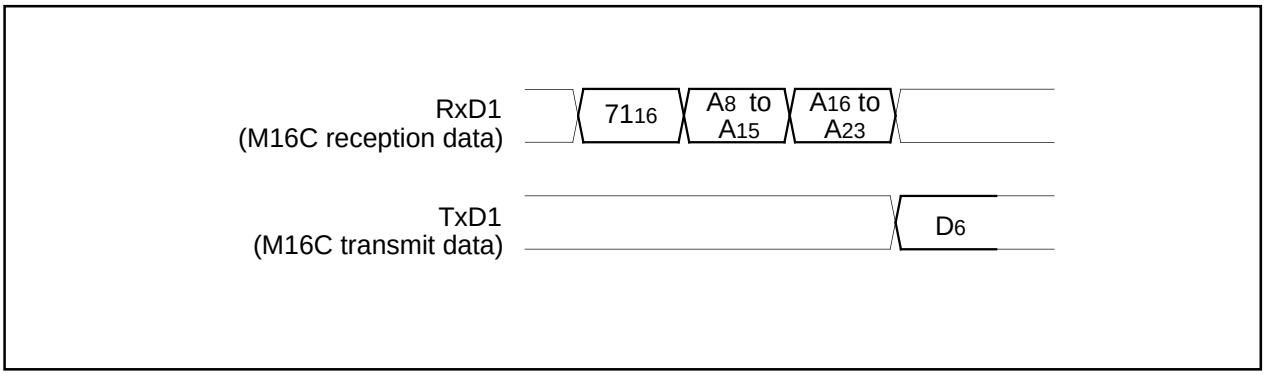


Figure 1.29.30. Timing for reading lock bit status

Lock Bit Enable Command

This command enables the lock bit in blocks whose bit was disabled with the lock bit disable command. The command code "$7A_{16}$" is sent with the 1st byte of the serial transmission. This command only enables the lock bit function; it does not set the lock bit itself.

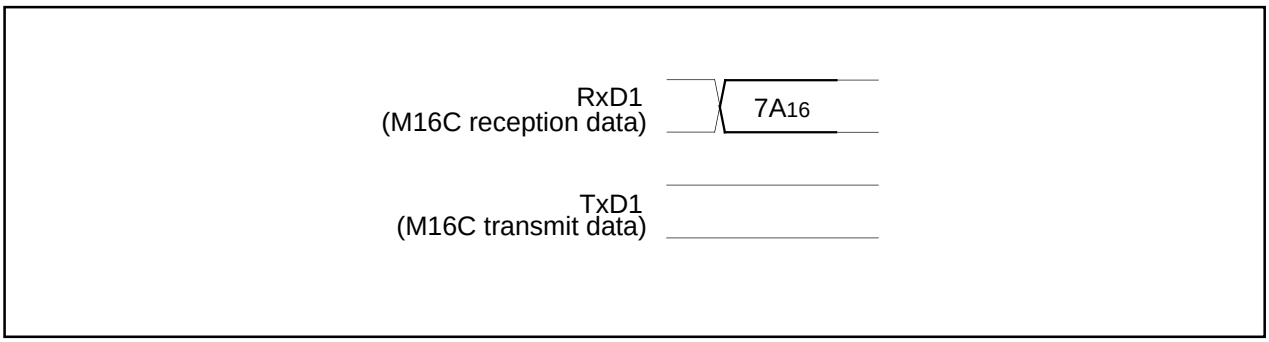


Figure 1.29.31. Timing for enabling the lock bit

Lock Bit Disable Command

This command disables the lock bit. The command code "75_{16}" is sent with the 1st byte of the serial transmission. This command only disables the lock bit function; it does not set the lock bit itself. However, if an erase command is executed after executing the lock bit disable command, "0" (locked) lock bit data is set to "1" (unlocked) after the erase operation ends. In any case, after the reset is cancelled, the lock bit is enabled.

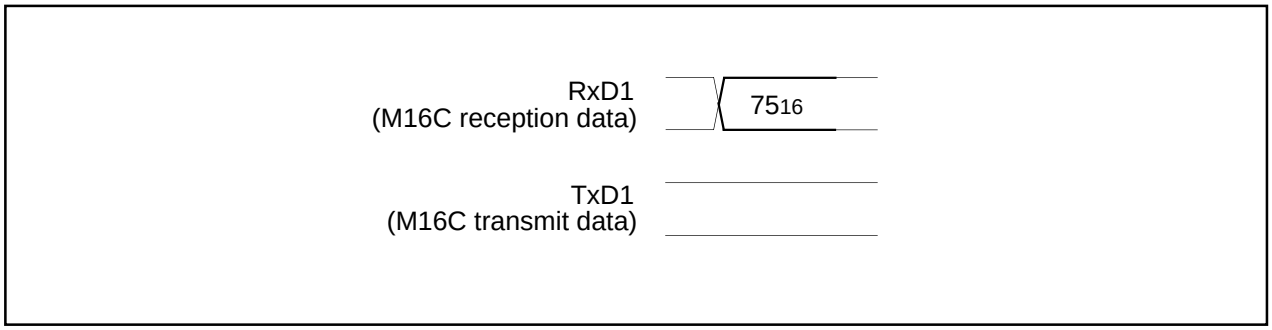


Figure 1.29.32. Timing for disabling the lock bit

Download Command

This command downloads a program to the RAM for execution. Execute the download command as explained here following.

(1) Transfer the "FA_{16}" command code with the 1st byte.
(2) Transfer the program size with the 2nd and 3rd bytes.
(3) Transfer the check sum with the 4th byte. The check sum is added to all data sent with the 5th byte onward.
(4) The program to execute is sent with the 5th byte onward.

When all data has been transmitted, if the check sum matches, the downloaded program is executed. The size of the program will vary according to the internal RAM.

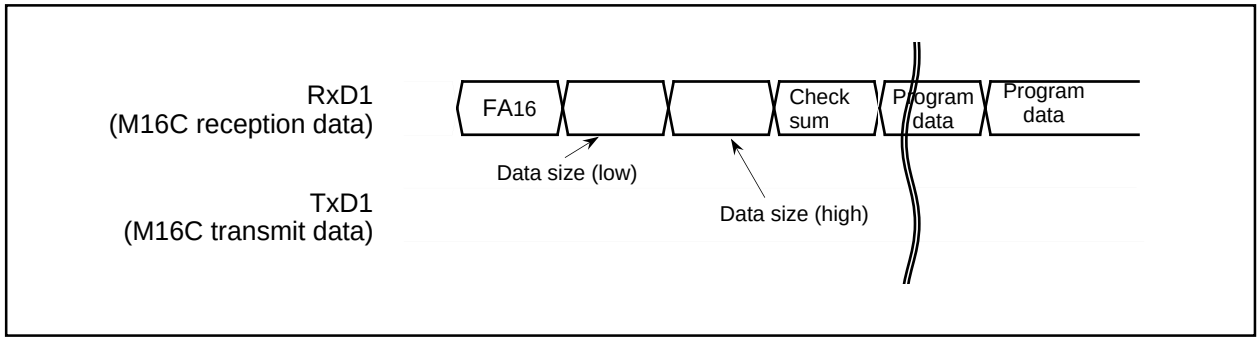


Figure 1.29.33. Timing for download

Version Information Output Command

This command outputs the version information of the control program stored in the boot area. Execute the version information output command as explained here following.

(1) Transfer the "FB16" command code with the 1st byte.

(2) The version information will be output from the 2nd byte onward. This data is composed of 8 ASCII code characters.

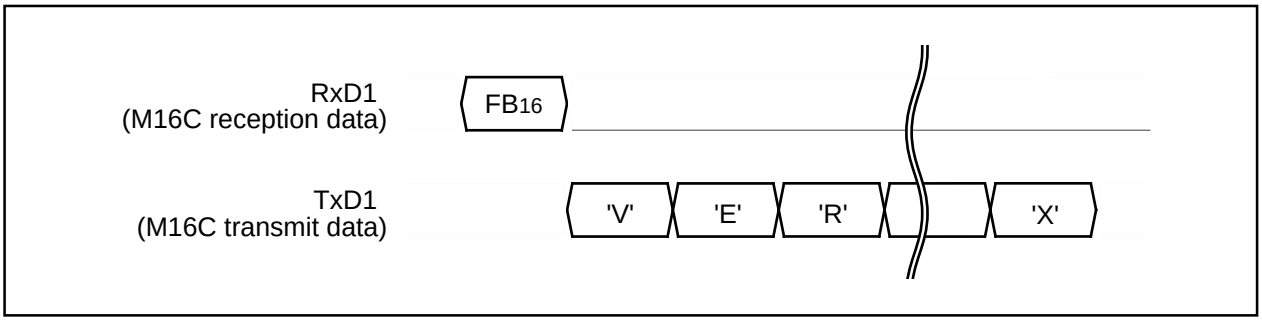


Figure 1.29.34. Timing for version information output

Boot ROM Area Output Command

This command outputs the control program stored in the boot ROM area in one page blocks (256 bytes). Execute the boot ROM area output command as explained here following.

(1) Transfer the "FC16" command code with the 1st byte.

(2) Transfer addresses A8 to A15 and A16 to A23 with the 2nd and 3rd bytes respectively.

(3) From the 4th byte onward, data (D0–D7) for the page (256 bytes) specified with addresses A8 to A23 will be output sequentially from the smallest address first.

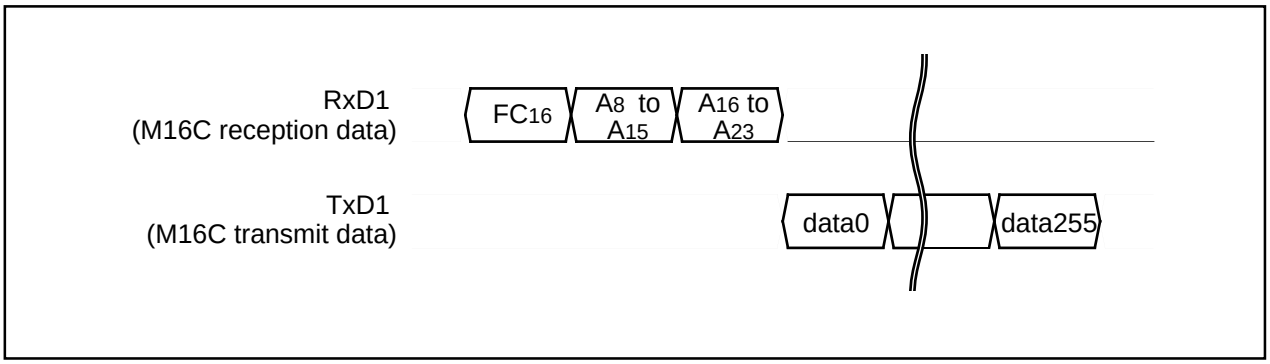


Figure 1.29.35. Timing for boot ROM area output

ID Check

This command checks the ID code. Execute the boot ID check command as explained here following.

(1) Transfer the "F5$_{16}$" command code with the 1st byte.

(2) Transfer addresses A$_0$ to A$_7$, A$_8$ to A$_{15}$ and A$_{16}$ to A$_{23}$ of the 1st byte of the ID code with the 2nd, 3rd and 4th bytes respectively.

(3) Transfer the number of data sets of the ID code with the 5th byte.

(4) The ID code is sent with the 6th byte onward, starting with the 1st byte of the code.

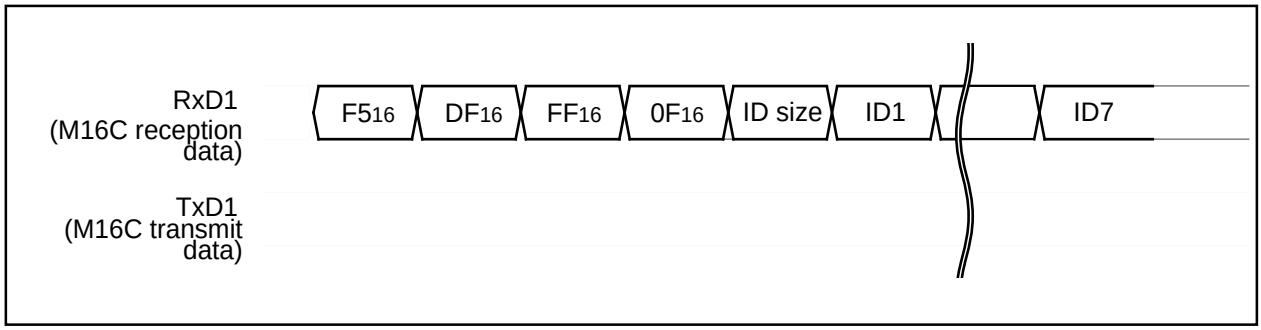


Figure 1.29.36. Timing for the ID check

ID Code

When the flash memory is not blank, the ID code sent from the peripheral units and the ID code written in the flash memory are compared to see if they match. If the codes do not match, the command sent from the peripheral units is not accepted. An ID code contains 8 bits of data. Area is, from the 1st byte, addresses 0FFFDF$_{16}$, 0FFFE3$_{16}$, 0FFFEB$_{16}$, 0FFFEF$_{16}$, 0FFFF3$_{16}$, 0FFFF7$_{16}$ and 0FFFFB$_{16}$. Write a program into the flash memory, which already has the ID code set for these addresses.

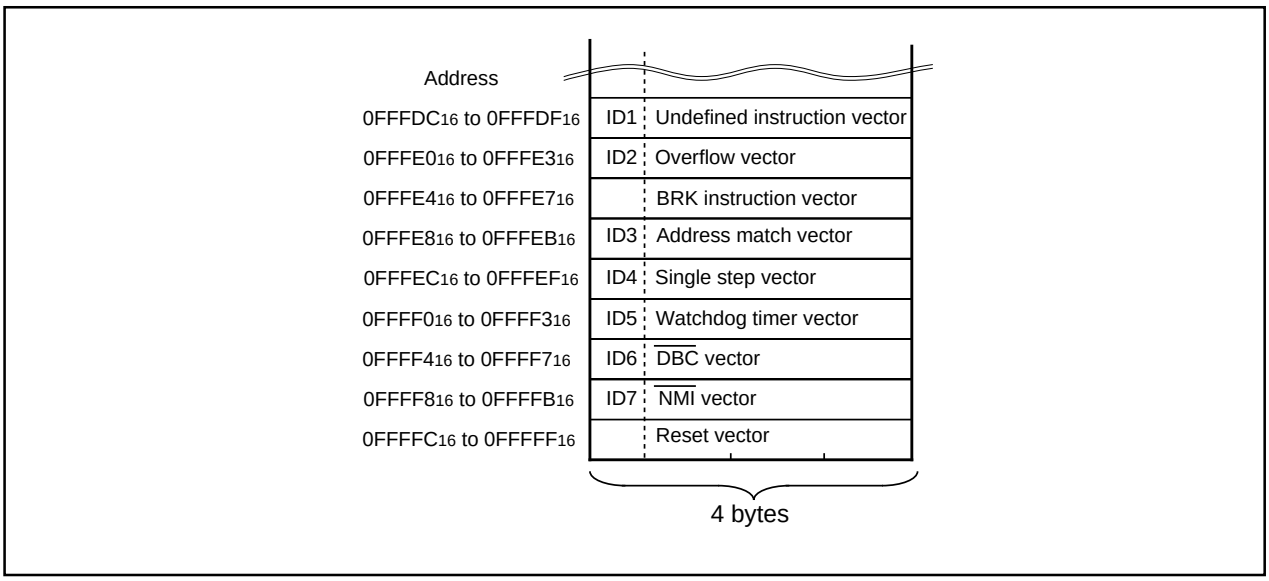


Figure 1.29.37. ID code storage addresses

Read Check Data

This command reads the check data that confirms that the write data, which was sent with the page program command, was successfully received.

(1) Transfer the "FD16" command code with the 1st byte.

(2) The check data (low) is received with the 2nd byte and the check data (high) with the 3rd.

To use this read check data command, first execute the command and then initialize the check data. Next, execute the page program command the required number of times. After that, when the read check command is executed again, the check data for all of the read data that was sent with the page program command during this time is read. The check data is the result of CRC operation of write data.

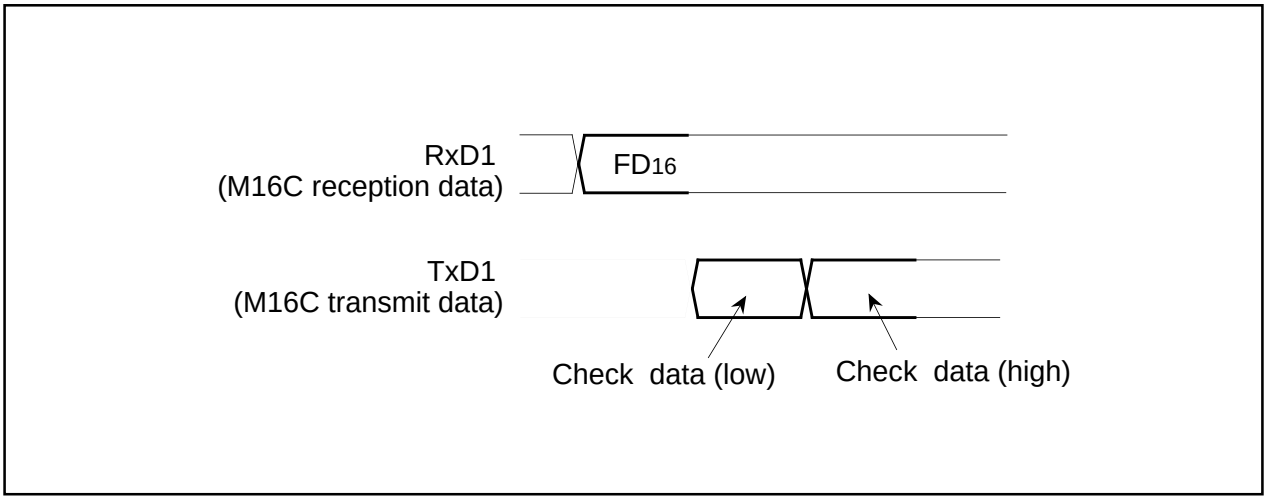


Figure 1.29.38. Timing for the read check data

Baud Rate 9600

This command changes baud rate to 9,600 bps. Execute it as follows.

(1) Transfer the "B016" command code with the 1st byte.

(2) After the "B016" check code is output with the 2nd byte, change the baud rate to 9,600 bps.

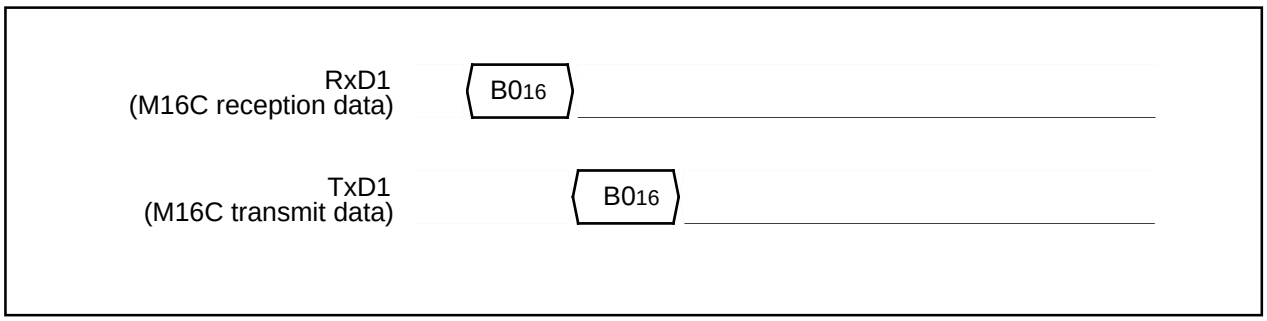


Figure 1.29.39. Timing of baud rate 9600

Baud Rate 19200

This command changes baud rate to 19,200 bps. Execute it as follows.

(1) Transfer the "B1$_{16}$" command code with the 1st byte.

(2) After the "B1$_{16}$" check code is output with the 2nd byte, change the baud rate to 19,200 bps.

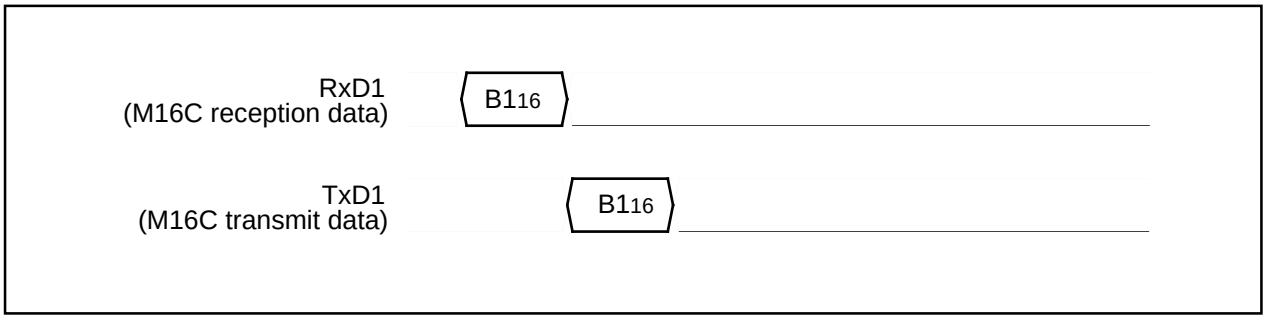


Figure 1.29.40. Timing of baud rate 19200

Baud Rate 38400

This command changes baud rate to 38,400 bps. Execute it as follows.

(1) Transfer the "B2$_{16}$" command code with the 1st byte.

(2) After the "B2$_{16}$" check code is output with the 2nd byte, change the baud rate to 38,400 bps.

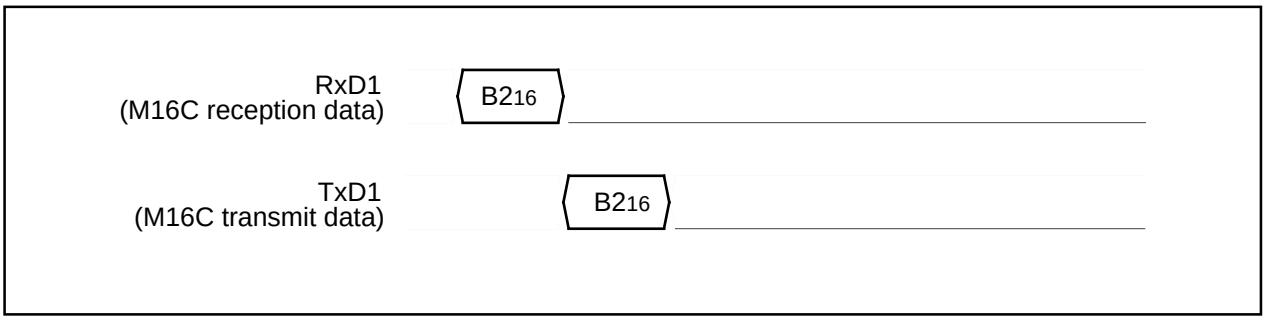


Figure 1.29.41. Timing of baud rate 38400

Baud Rate 57600

This command changes baud rate to 57,600 bps. Execute it as follows.

(1) Transfer the "B3$_{16}$" command code with the 1st byte.

(2) After the "B3$_{16}$" check code is output with the 2nd byte, change the baud rate to 57,600 bps.

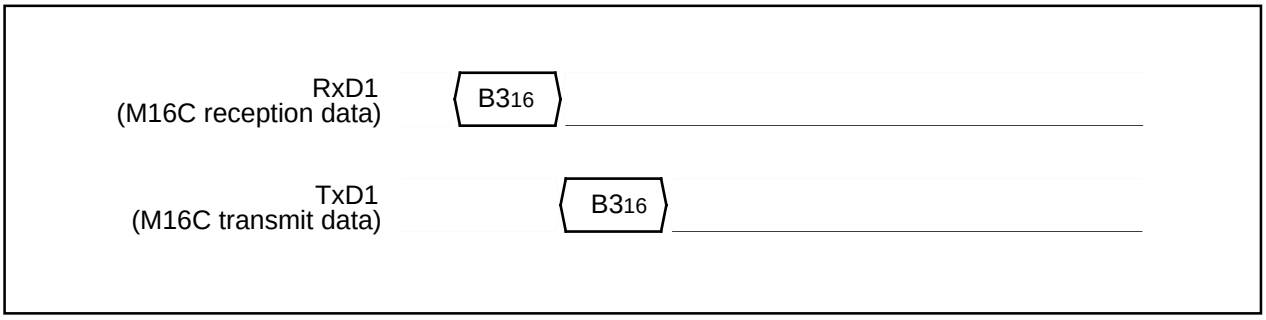
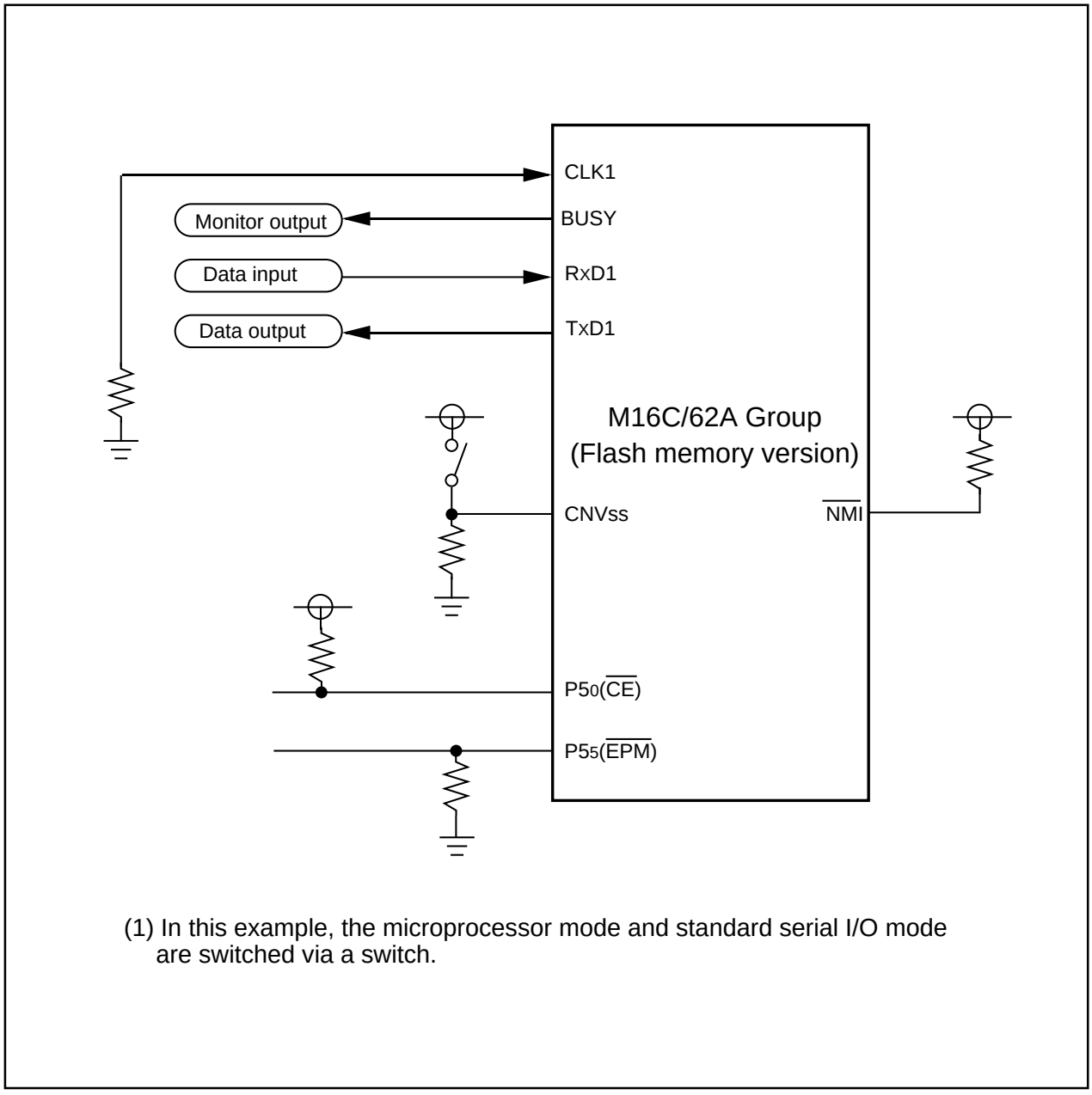


Figure 1.29.42. Timing of baud rate 57600

Example Circuit Application for The Standard Serial I/O Mode 2

The below figure shows a circuit application for the standard serial I/O mode 2.



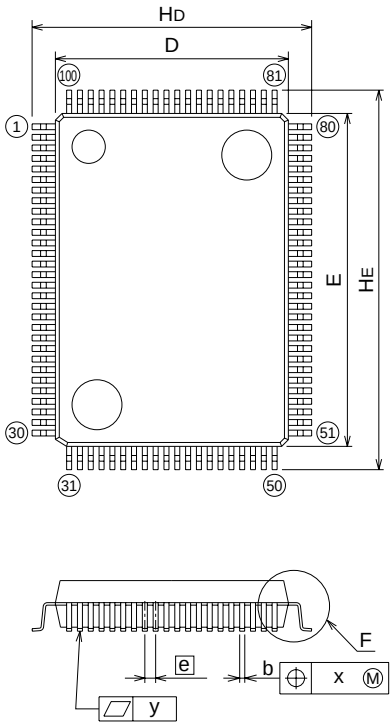
(1) In this example, the microprocessor mode and standard serial I/O mode are switched via a switch.

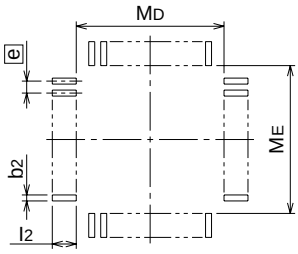
Figure 1.29.43. Example circuit application for the standard serial I/O mode 2

100P6S-A (MMP)

Plastic 100pin 14×20mm body QFP

EIAJ Package Code	JEDEC Code	Weight(g)	Lead Material
QFP100-P-1420-0.65	–	1.58	Alloy 42





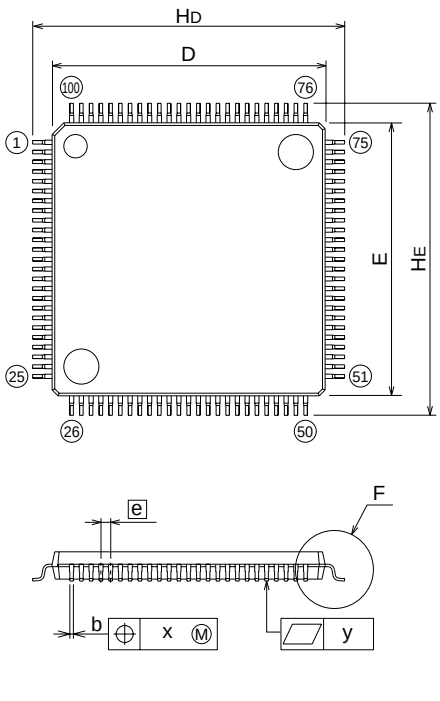
Recommended Mount Pad

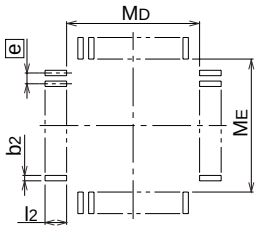
Symbol	Dimension in Millimeters		
	Min	Nom	Max
A	–	–	3.05
A1	0	0.1	0.2
A2	–	2.8	–
b	0.25	0.3	0.4
c	0.13	0.15	0.2
D	13.8	14.0	14.2
E	19.8	20.0	20.2
e	–	0.65	–
HD	16.5	16.8	17.1
HE	22.5	22.8	23.1
L	0.4	0.6	0.8
L1	–	1.4	–
x	–	–	0.13
y	–	–	0.1
θ	0°	–	10°
b2	–	0.35	–
l2	1.3	–	–
MD	–	14.6	–
ME	–	20.6	–

100P6Q-A (MMP)

Plastic 100pin 14×14mm body LQFP

EIAJ Package Code	JEDEC Code	Weight(g)	Lead Material
LQFP100-P-1414-0.50	–	0.63	Cu Alloy





Recommended Mount Pad

Symbol	Dimension in Millimeters		
	Min	Nom	Max
A	–	–	1.7
A1	0	0.1	0.2
A2	–	1.4	–
b	0.13	0.18	0.28
c	0.105	0.125	0.175
D	13.9	14.0	14.1
E	13.9	14.0	14.1
e	–	0.5	–
HD	15.8	16.0	16.2
HE	15.8	16.0	16.2
L	0.3	0.5	0.7
L1	–	1.0	–
Lp	0.45	0.6	0.75
A3	–	0.25	–
x	–	–	0.08
y	–	–	0.1
θ	0°	–	10°
b2	–	0.225	–
l2	0.9	–	–
MD	–	14.4	–
ME	–	14.4	–

Differences between M16C/62A and M16C/62

Item	M16C/62A	M16C/62
Memory area	1 Mbyte fixed	Memory expansion 1.2 Mbytes mode 4 Mbytes mode
Serial I/O	No CTS/RTS separate function	CTS/RTS separate function
IIC bus mode	Analog or digital delay is selected as SDA delay	Only analog delay is selected as SDA delay
EPROM / one time PROM version	None	Have
Flash memory version	Standard serial I/O mode (clock asynchronized) is supported	Clock synchronized only

Differences in SFR between M16C/62A and M16C/62

Address	Register name	M16C/62A	M16C/62
0005_{16}	Processor mode register 1 (PM1)	b5,b4 Reserved bits	b5,b4 Memory area expansion bits
$000B_{16}$	Data bank register (DBR)	Reserved register	Have
$03B0_{16}$	UART transmit/receive register 2 (UCON)	b6 Reserved bit	b6 CTS/RTS separation bit
0375_{16}	UART2 special mode register 3 (U2SMR3)	Have	None
0377_{16}	UART2 special mode register (U2SMR)	b7 SDA digital delay select bit	b7 Reserved bit

Description

The M16C/62M group of single-chip microcomputers are built using the high-performance silicon gate CMOS process using a M16C/60 Series CPU core and are packaged in a 100-pin plastic molded QFP. These single-chip microcomputers operate using sophisticated instructions featuring a high level of instruction efficiency. With 1M bytes of address space, low voltage (2.2V to 3.6V), they are capable of executing instructions at high speed. They also feature a built-in multiplier and DMAC, making them ideal for controlling office, communications, industrial equipment, and other high-speed processing applications.

The M16C/62M group includes a wide range of products with different internal memory types and sizes and various package types.

Features

- Memory capacity ..... ROM (See Figure 1.30.4. ROM Expansion)
  RAM 10K to 20K bytes
- Shortest instruction execution time ..... 100ns (f(X_{IN})=10MHz, V_{CC}=2.7V to 3.6V)
  142.9ns (f(X_{IN})=7MHz, V_{CC}=2.2V to 3.6V with software one-wait)
- Supply voltage ..... 2.7V to 3.6V (f(X_{IN})=10MHz, without software wait)
  2.4V to 2.7V (f(X_{IN})=7MHz, without software wait)
  2.2V to 2.4V (f(X_{IN})=7MHz with software one-wait)
- Low power consumption ..... 28.5mW (V_{CC} = 3V, f(X_{IN})=10MHz, without software wait)
- Interrupts ..... 25 internal and 8 external interrupt sources, 4 software interrupt sources; 7 levels (including key input interrupt)
- Multifunction 16-bit timer ..... 5 output timers + 6 input timers
- Serial I/O ..... 5 channels
  (3 for UART or clock synchronous, 2 for clock synchronous)
- DMAC ..... 2 channels (trigger: 24 sources)
- A-D converter ..... 10 bits X 8 channels (Expandable up to 10 channels)
- D-A converter ..... 8 bits X 2 channels
- CRC calculation circuit ..... 1 circuit
- Watchdog timer ..... 1 line
- Programmable I/O ..... 87 lines
- Input port ..... 1 line ($P8_5$ shared with $\overline{NMI}$ pin)
- Memory expansion ..... Available (to a maximum of 1M bytes)
- Chip select output ..... 4 lines
- Clock generating circuit ..... 2 built-in clock generation circuits
  (built-in feedback resistor, and external ceramic or quartz oscillator)

Applications

Audio, cameras, office equipment, communications equipment, portable equipment

Pin Configuration

Figures 1.30.1 and 1.30.2 show the pin configurations (top view).

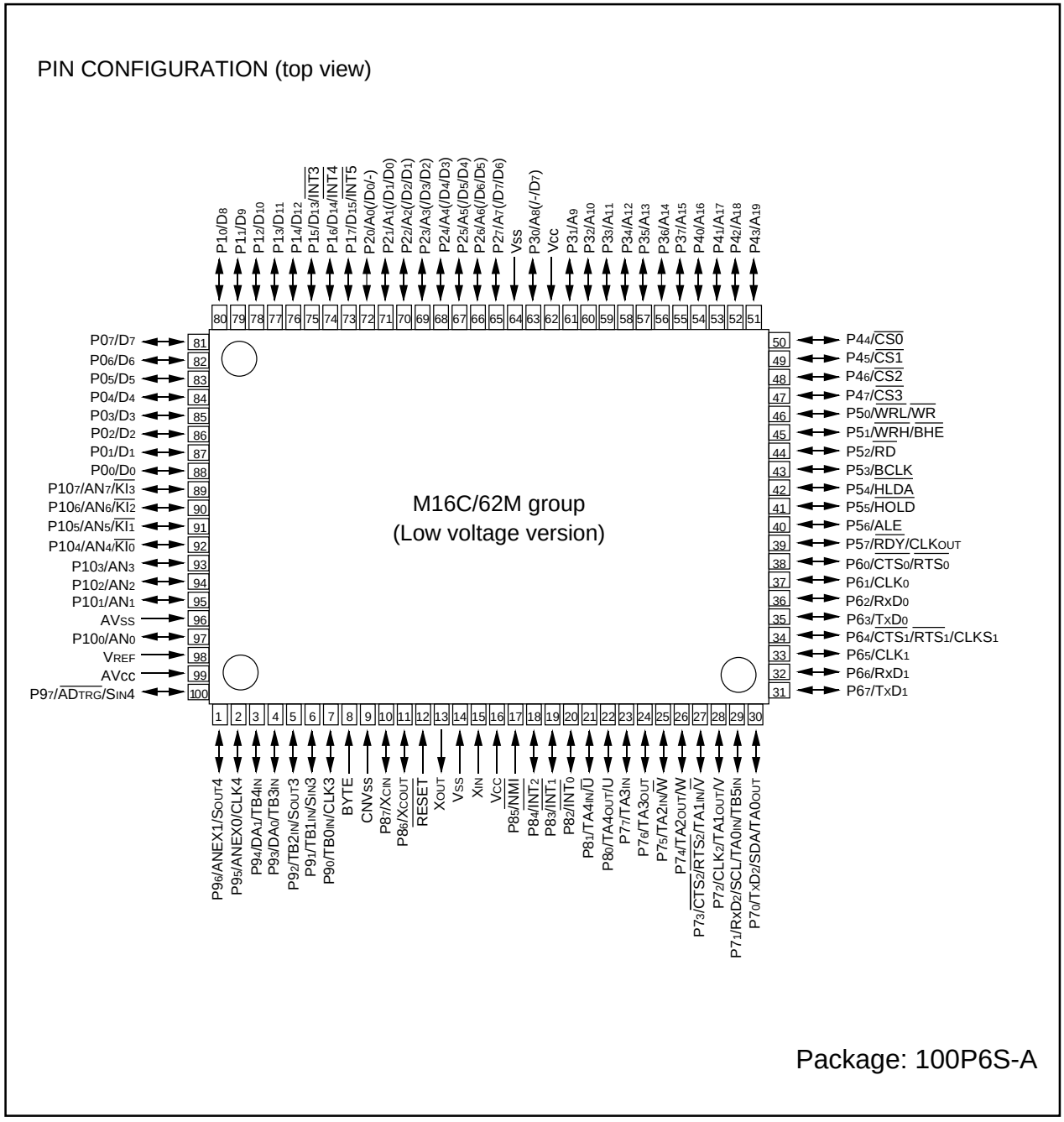


Figure 1.30.1. Pin configuration (top view)

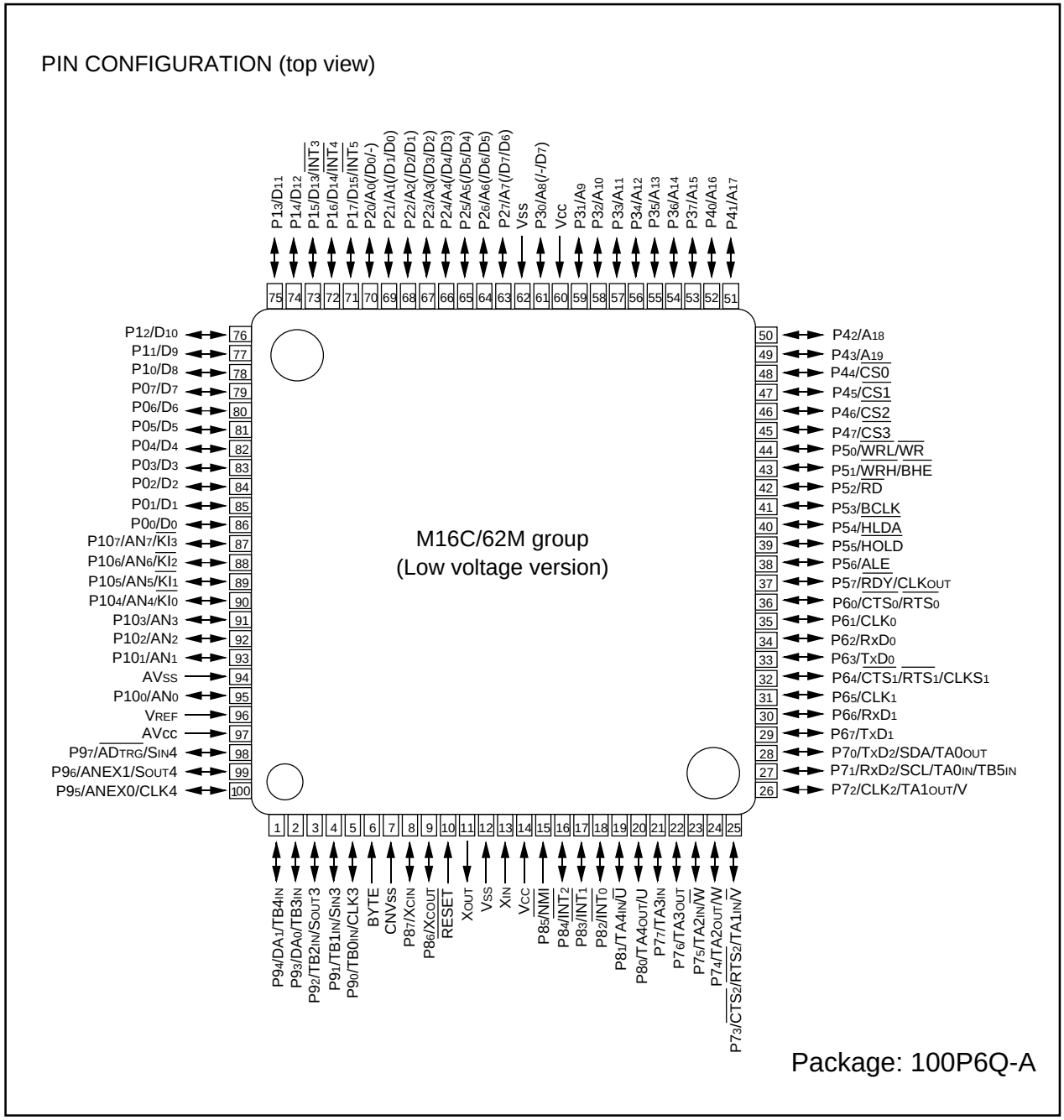
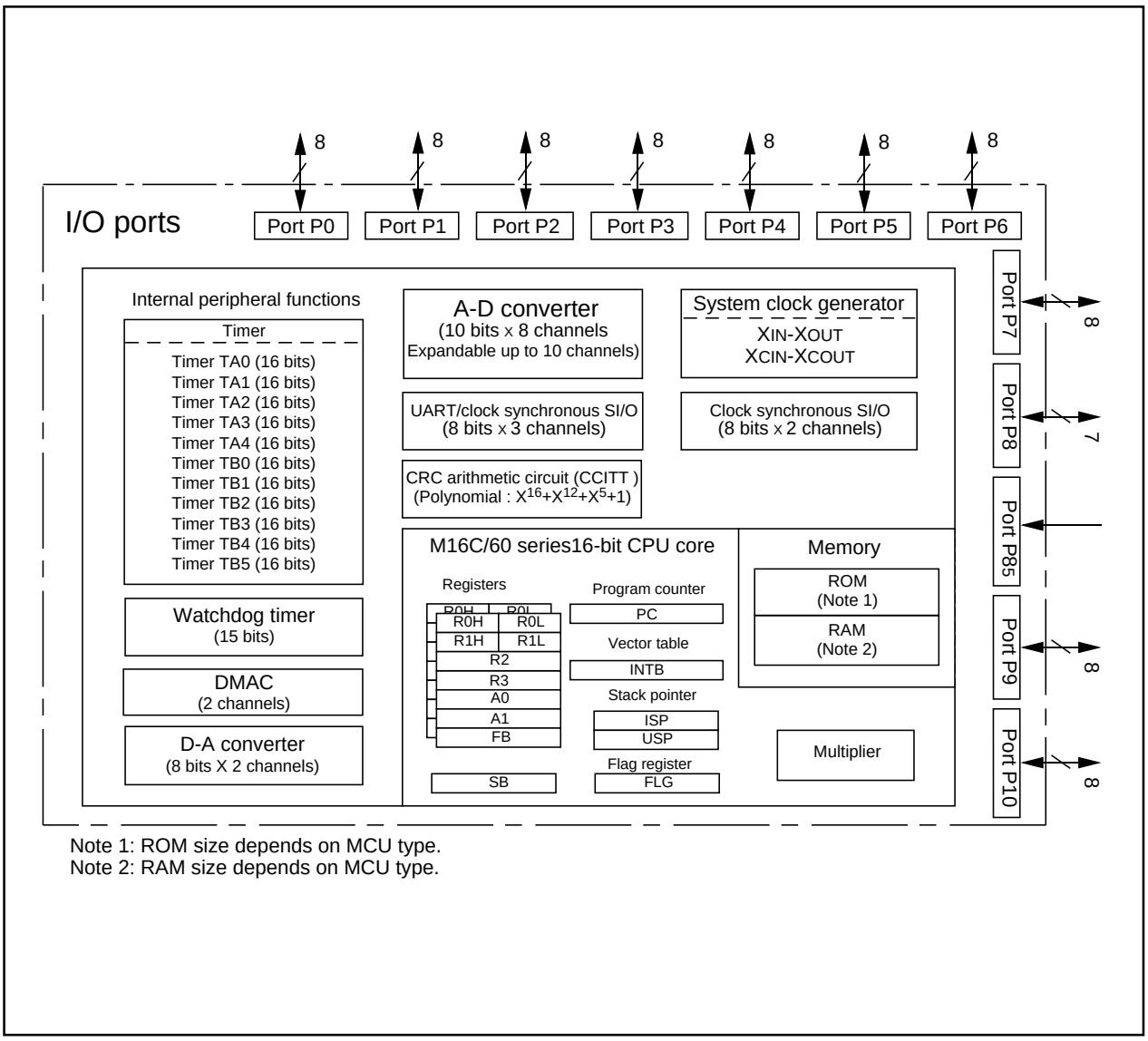


Figure 1.30.2. Pin configuration (top view)

Block Diagram

Figure 1.30.3 is a block diagram of the M16C/62M group.



Note 1: ROM size depends on MCU type.
Note 2: RAM size depends on MCU type.

Figure 1.30.3. Block diagram of M16C/62M group

Performance Outline

Table 1.30.1 is a performance outline of M16C/62M group.

Table 1.30.1. Performance outline of M16C/62M group

Item		Performance
Number of basic instructions		91 instructions
Shortest instruction execution time		100ns(f(XIN)=10MHz, VCC=2.7V to 3.6V) 142.9ns (f(XIN)=7MHz, VCC=2.2V to 3.6V with software one-wait)
Memory capacity	ROM	(See the figure 1.30.4. ROM Expansion)
	RAM	10K to 20K bytes
I/O port	P0 to P10 (except P85)	8 bits x 10, 7 bits x 1
Input port	P85	1 bit x 1
Multifunction timer	TA0, TA1, TA2, TA3, TA4	16 bits x 5
	TB0, TB1, TB2, TB3, TB4, TB5	16 bits x 6
Serial I/O	UART0, UART1, UART2	(UART or clock synchronous) x 3
	SI/O3, SI/O4	(Clock synchronous) x 2
A-D converter		10 bits x (8 + 2) channels
D-A converter		8 bits x 2
DMAC		2 channels (trigger: 24 sources)
CRC calculation circuit		CRC-CCITT
Watchdog timer		15 bits x 1 (with prescaler)
Interrupt		25 internal and 8 external sources, 4 software sources, 7 levels
Clock generating circuit		2 built-in clock generation circuits (built-in feedback resistor, and external ceramic or quartz oscillator)
Supply voltage		2.7V to 3.6V (f(XIN)=10MHz, without software wait) 2.4V to 2.7V (f(XIN)=7MHz, without software wait) 2.2V to 2.4V (f(XIN)=7MHz with software one-wait)
Power consumption		28.5mW (f(XIN) =10MHz, VCC=3V without software wait)
I/O characteristics	I/O withstand voltage	3V
	Output current	1mA
Memory expansion		Available (to a maximum of 1M bytes)
Device configuration		CMOS high performance silicon gate
Package		100-pin plastic mold QFP

Mitsubishi plans to release the following products in the M16C/62M group:

(1) Support for mask ROM version and Flash memory version

(2) ROM capacity

(3) Package

100P6S-A : Plastic molded QFP (mask ROM and flash memory versions)

100P6Q-A : Plastic molded QFP (mask ROM and flash memory versions)

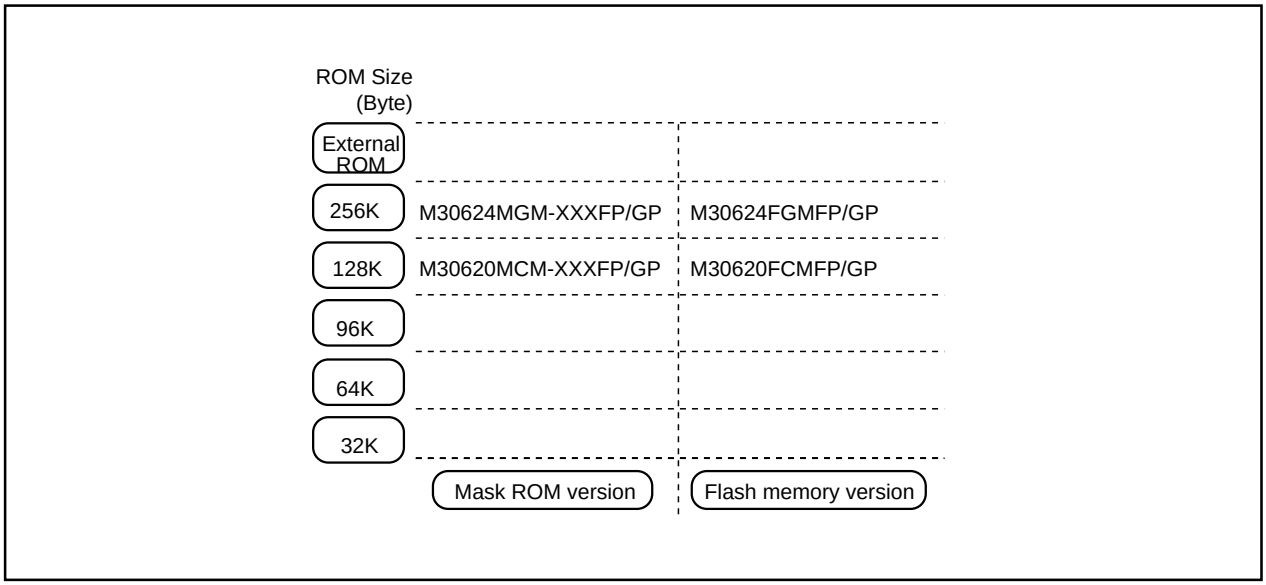


Figure 1.30.4. ROM expansion

The M16C/62M group products currently supported are listed in Table 1.30.2.

Table 1.30.2. M16C/62M group

June, 2001

Type No	ROM capacity	RAM capacity	Package type	Remarks
M30620MCM-XXXFP	128K byte	10K byte	100P6S-A	mask ROM version
M30620MCM-XXXGP			100P6Q-A	
M30624MGM-XXXFP	256K byte	20K byte	100P6S-A	
M30624MGM-XXXGP			100P6Q-A	
M30620FCMFP	128K byte	10K byte	100P6S-A	Flash memory 3V version
M30620FCMGP			100P6Q-A	
M30624FGMFP	256K byte	20K byte	100P6S-A	
M30624FGMGP			100P6Q-A	

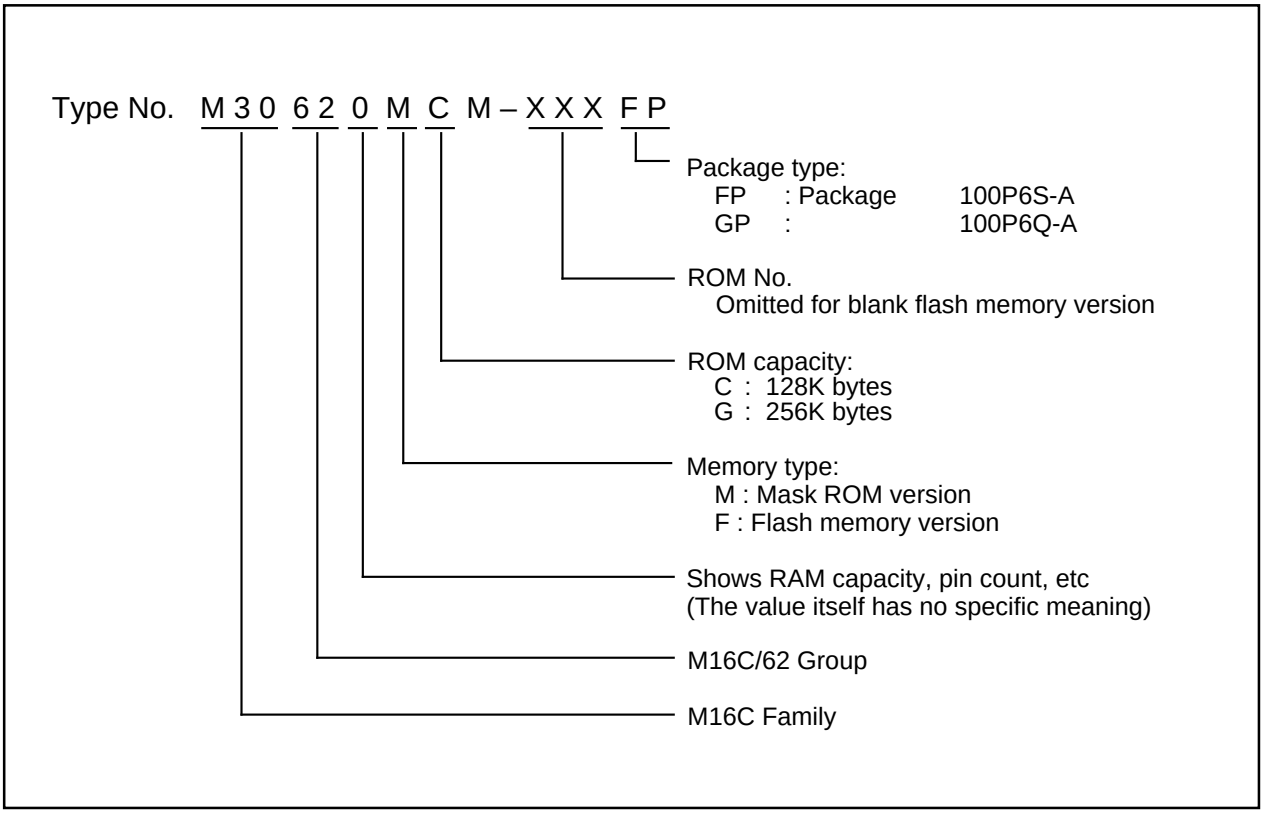


Figure 1.30.5. Type No., memory size, and package

Table 1.31.1. Absolute maximum ratings

Symbol	Parameter		Condition	Rated value	Unit
V_{CC}	Supply voltage		V_{CC}=AV$_{CC}$	- 0.3 to 4.6	V
AVcc	Analog supply voltage		V_{CC}=AV$_{CC}$	- 0.3 to 4.6	V
V_I	Input voltage	$\overline{RESET}$, CNV$_{SS}$, BYTE, P0$_0$ to P0$_7$, P1$_0$ to P1$_7$, P2$_0$ to P2$_7$, P3$_0$ to P3$_7$,P4$_0$ to P4$_7$, P5$_0$ to P5$_7$, P6$_0$ to P6$_7$, P7$_2$ to P7$_7$, P8$_0$ to P8$_7$, P9$_0$ to P9$_7$, P10$_0$ to P10$_7$, V_{REF}, X_{IN}		- 0.3 to Vcc + 0.3	V
		P7$_0$, P7$_1$		- 0.3 to 4.6	V
V_O	Output voltage	P0$_0$ to P0$_7$, P1$_0$ to P1$_7$, P2$_0$ to P2$_7$, P3$_0$ to P3$_7$, P4$_0$ to P4$_7$, P5$_0$ to P5$_7$, P6$_0$ to P6$_7$, P7$_2$ to P7$_7$, P8$_0$ to P8$_4$, P8$_6$, P8$_7$, P9$_0$ to P9$_7$, P10$_0$ to P10$_7$, X_{OUT}		- 0.3 to Vcc + 0.3	V
		P7$_0$, P7$_1$		- 0.3 to 4.6	V
P_d	Power dissipation		Topr=25 °C	300	mW
T_{opr}	Operating ambient temperature			- 20 to 85 / -40 to 85 (Note)	°C
T_{stg}	Storage temperature			- 65 to 150	°C

Note : Specify a product of -40°C to 85°C to use it.

Table 1.31.2. Recommended operating conditions (referenced to V_{CC} = 2.2V to 3.6V at Topr = –20°C to 85°C / – 40°C to 85°C (Note 3) unless otherwise specified)

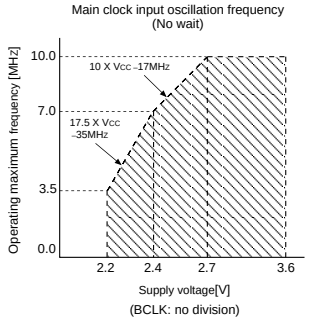
Symbol	Parameter			Standard			Unit
				Min.	Typ.	Max.	
Vcc	Supply voltage			2.2	3.0	3.6	V
AVcc	Analog supply voltage				Vcc		V
Vss	Supply voltage				0		V
AVss	Analog supply voltage				0		V
V_{IH}	HIGH input voltage	$P3_1$ to $P3_7$, $P4_0$ to $P4_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_2$ to $P7_7$, $P8_0$ to $P8_7$, $P9_0$ to $P9_7$, $P10_0$ to $P10_7$, X_{IN}, RESET, CNV_{SS}, BYTE		0.8Vcc		Vcc	V
		$P7_0$, $P7_1$		0.8Vcc		4.6	V
		$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ (during single-chip mode)		0.8Vcc		Vcc	V
		$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ (data input function during memory expansion and microprocessor modes)		0.5Vcc		Vcc	V
V_{IL}	LOW input voltage	$P3_1$ to $P3_7$, $P4_0$ to $P4_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_7$, $P9_0$ to $P9_7$, $P10_0$ to $P10_7$, X_{IN}, RESET, CNV_{SS}, BYTE		0		0.2Vcc	V
		$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ (during single-chip mode)		0		0.2Vcc	V
		$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ (data input function during memory expansion and microprocessor modes)		0		0.16Vcc	V
$I_{OH (peak)}$	HIGH peak output current	$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_2$ to $P7_7$, $P8_0$ to $P8_4$, $P8_6$, $P8_7$, $P9_0$ to $P9_7$, $P10_0$ to $P10_7$				- 10.0	mA
$I_{OH (avg)}$	HIGH average output current	$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_2$ to $P7_7$, $P8_0$ to $P8_4$, $P8_6$, $P8_7$, $P9_0$ to $P9_7$, $P10_0$ to $P10_7$				- 5.0	mA
$I_{OL (peak)}$	LOW peak output current	$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_4$, $P8_6$, $P8_7$, $P9_0$ to $P9_7$, $P10_0$ to $P10_7$				10.0	mA
$I_{OL (avg)}$	LOW average output current	$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_4$, $P8_6$, $P8_7$, $P9_0$ to $P9_7$, $P10_0$ to $P10_7$				5.0	mA
f (X_{IN})	Main clock input oscillation frequency	No wait	Vcc=2.7V to 3.6V	0		10	MHz
			Vcc=2.4V to 2.7V	0		10 X Vcc - 17	MHz
			Vcc=2.2V to 2.4V	0		17.5 X Vcc - 35	MHz
		with wait	Vcc=2.7V to 3.6V	0		10	MHz
			Vcc=2.2V to 2.7V	0		6 X Vcc - 6.2	MHz
f (X_{CIN})	Subclock oscillation frequency				32.768	50	kHz

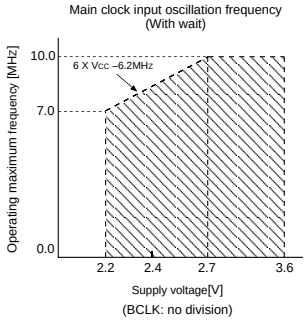
Note 1: The mean output current is the mean value within 100ms.

Note 2: The total I_{OL} (peak) for ports P0, P1, P2, $P8_6$, $P8_7$, P9, and P10 must be 80mA max. The total I_{OH} (peak) for ports P0, P1, P2, $P8_6$, $P8_7$, P9, and P10 must be 80mA max. The total I_{OL} (peak) for ports P3, P4, P5, P6, P7, and $P8_0$ to $P8_4$ must be 80mA max. The total I_{OH} (peak) for ports P3, P4, P5, P6, $P7_2$ to $P7_7$, and $P8_0$ to $P8_4$ must be 80mA max.

Note 3: Specify a product of -40°C to 85°C to use it.

Note 4: Relationship between main clock oscillation frequency and supply voltage.





Flash memory version program voltage and read operation voltage characteristics

Flash program voltage	Flash read operation voltage
Vcc=2.7V to 3.6V	Vcc=2.4V to 3.6V
Vcc=2.7V to 3.4V	Vcc=2.2V to 2.4V

Note 5: Execute case without wait, program / erase of flash memory by Vcc=2.7V to 3.6V and f(BCLK) ≤ 6.25 MHz. Execute case with wait, program / erase of flash memory by Vcc=2.7V to 3.6V and f(BCLK) ≤ 10.0 MHz.

Table 1.31.3. Electrical characteristics (referenced to V_{CC} = 2.7V to 3.6V, V_{SS} = 0V at Topr = – 20°C to 85°C / – 40°C to 85°C (Note 1), f(X_{IN}) = 10MHz without wait unless otherwise specified)

Symbol	Parameter			Measuring condition		Standard Min	Standard Typ.	Standard Max.	Unit
V_{OH}	HIGH output voltage	$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_2$ to $P7_7$, $P8_0$ to $P8_4$, $P8_6$,$P8_7$, $P9_0$ to $P9_7$, $P10_0$ to $P10_7$		I_{OH}=–1mA		2.5			V
V_{OH}	HIGH output voltage	X_{OUT}	HIGHPOWER	I_{OH}=–0.1mA		2.5			V
			LOWPOWER	I_{OH}=–50µA		2.5			
	HIGH output voltage	X_{COUT}	HIGHPOWER	With no load applied			3.0		V
			LOWPOWER	With no load applied			1.6		
V_{OL}	LOW output voltage	$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_4$, $P8_6$,$P8_7$, $P9_0$ to $P9_7$, $P10_0$ to $P10_7$		I_{OL}=1mA				0.5	V
V_{OL}	LOW output voltage	X_{OUT}	HIGHPOWER	I_{OL}=0.1mA				0.5	V
			LOWPOWER	I_{OL}=50µA				0.5	
	LOW output voltage	X_{COUT}	HIGHPOWER	With no load applied			0		V
			LOWPOWER	With no load applied			0		
V_{T+}–V_{T-}	Hysteresis	$\overline{HOLD}$, $\overline{RDY}$, $TA0_{IN}$ to $TA4_{IN}$, $TB0_{IN}$ to $TB5_{IN}$, $\overline{INT0}$ to $\overline{INT5}$, $\overline{NMI}$, $\overline{ADTRG}$, $\overline{CTS0}$ to $\overline{CTS2}$, SCL, SDA, CLK_0 to CLK_4, $TA2_{OUT}$ to $TA4_{OUT}$, $\overline{KI0}$ to $\overline{KI3}$, RxD_0 to RxD_2, SIN_3, SIN_4				0.2		0.8	V
V_{T+}–V_{T-}	Hysteresis	$\overline{RESET}$				0.2		1.8	V
I_{IH}	HIGH input current	$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_7$, $P9_0$ to $P9_7$, $P10_0$ to $P10_7$, X_{IN}, $\overline{RESET}$, CNVss, BYTE		V_I=3V				4.0	µA
I_{IL}	LOW input current	$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_7$, $P9_0$ to $P9_7$, $P10_0$ to $P10_7$, X_{IN}, $\overline{RESET}$, CNVss, BYTE		V_I=0V				–4.0	µA
R_{PULLUP}	Pull-up resistance	$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_2$ to $P7_7$, $P8_0$ to $P8_4$, $P8_6$,$P8_7$, $P9_0$ to $P9_7$, $P10_0$ to $P10_7$		V_I=0V		20	75	330	kΩ
R_{fXIN}	Feedback resistance	X_{IN}					3.0		MΩ
R_{fXCIN}	Feedback resistance	X_{CIN}					10.0		MΩ
V_{RAM}	RAM retention voltage			When clock is stopped		2.0			V
I_{CC}	Power supply current	In single-chip mode, the output pins are open and other pins are V_{SS}		Mask ROM version	f(X_{IN})=10MHz Square wave, no division		9.5	21.25	mA
				Flash memory 3V version	f(X_{IN})=10MHz Square wave, no division		12.0	21.25	mA
				Mask ROM version, flash memory 3V version	f(X_{CIN})=32kHz Square wave		45.0		µA
				Flash memory 3V version program	f(X_{IN})=10MHz Square wave, division by 2		14.0		mA
				Flash memory 3V version erase	f(X_{IN})=10MHz Square wave, division by 2		17.0		mA
				Mask ROM version, flash memory 3V version	f(X_{CIN})=32kHz When a WAIT instruction is executed. Oscillation capacity High (Note 2)		2.8		µA
					f(X_{CIN})=32kHz When a WAIT instruction is executed. Oscillation capacity Low (Note 2)		0.9		µA
					When clock is stopped Topr=25°C			1.0	µA
					When clock is stopped Topr=85°C			20.0	

Note 1: Specify a product of -40°C to 85°C to use it.

Note 2: With one timer operated using f_{C32}.

Table 1.31.4. A-D conversion characteristics (referenced to V_{CC} = AV_{CC} = V_{REF} = 2.4V to 3.6V, V_{SS} = AV_{SS} = 0V, at Topr = – 20°C to 85°C / – 40°C to 85°C (Note 2), f(X_{IN})=10MHz unless otherwise specified)

Symbol	Parameter		Measuring condition	Standard			Unit
				Min.	Typ.	Max	
–	Resolution		V_{REF} =V_{CC}			10	Bits
–	Absolute accuracy	Sample & hold function not available (8 bit)	V_{REF} =V_{CC}=3V, f_{AD}=f_{AD}/2			±2	LSB
R_{LADDER}	Ladder resistance		V_{REF} =V_{CC}	10		40	kΩ
t_{CONV}	Conversion time(8bit)			9.8			µs
V_{REF}	Reference voltage			2.4		V_{CC}	V
V_{IA}	Analog input voltage			0		V_{REF}	V

Note 1: Connect AV_{CC} pin to V_{CC} pin and apply the same electric potential.

Note 2: Specify a product of –40°C to 85°C to use it.

Table 1.31.5. D-A conversion characteristics (referenced to V_{CC} = 2.4V to 3.6V, V_{SS} = AV_{SS} = 0V, V_{REF}=3V, at Topr = – 20°C to 85°C / – 40°C to 85°C (Note 2), f(X_{IN})=10MHz unless otherwise specified)

Symbol	Parameter	Measuring 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	(Note1)			1.0	mA

Note 1: This applies when using one D-A converter, with the D-A register for the unused D-A converter set to "00_{16}". The A-D converter's ladder resistance is not included.

Also, when D-A register contents are not "00_{16}", the current I_{VREF} always flows even though Vref may have been set to be "unconnected" by the A-D control register.

Note 2: Specify a product of –40°C to 85°C to use it.

Table 1.31.6. Flash memory version electrical characteristics (referenced to V_{CC} = 2.7V to 3.6V, at Topr = 0°C to 60°C unless otherwise specified)

Parameter	Standard			Unit
	Min.	Typ.	Max	
Page program time		6	120	ms
Block erase time		50	600	ms
Erase all unlocked blocks time		50 X n (Note)	600 X n (Note)	ms
Lock bit program time		6	120	ms

Note : n denotes the number of block erases.

Table 1.31.7. Flash memory version program voltage and read operation voltage characteristics (Topr = 0°C to 60°C)

Flash program voltage	Flash read operation voltage
V_{CC}=2.7V to 3.6V	V_{CC}=2.4V to 3.6V
V_{CC}=2.7V to 3.4V	V_{CC}=2.2V to 2.4V

Timing requirements

(referenced to V_{CC} = 3V, V_{SS} = 0V, at Topr = – 20°C to 85°C / – 40°C to 85°C (*) unless otherwise specified)

* : Specify a product of -40°C to 85°C to use it.

Table 1.31.8. External clock input

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t_c	External clock input cycle time	100		ns
$t_{w(H)}$	External clock input HIGH pulse width	40		ns
$t_{w(L)}$	External clock input LOW pulse width	40		ns
t_r	External clock rise time		18	ns
t_f	External clock fall time		18	ns

Table 1.31.9. Memory expansion and microprocessor modes

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{ac1(RD-DB)}$	Data input access time (no wait)		(Note)	ns
$t_{ac2(RD-DB)}$	Data input access time (with wait)		(Note)	ns
$t_{ac3(RD-DB)}$	Data input access time (when accessing multiplex bus area)		(Note)	ns
$t_{su(DB-RD)}$	Data input setup time	80		ns
$t_{su(RDY-BCLK)}$	$\overline{RDY}$ input setup time	60		ns
$t_{su(HOLD-BCLK)}$	$\overline{HOLD}$ input setup time	80		ns
$t_{h(RD-DB)}$	Data input hold time	0		ns
$t_{h(BCLK-RDY)}$	$\overline{RDY}$ input hold time	0		ns
$t_{h(BCLK-HOLD)}$	$\overline{HOLD}$ input hold time	0		ns
$t_{d(BCLK-HLDA)}$	$\overline{HLDA}$ output delay time		100	ns

Note: Calculated according to the BCLK frequency as follows:

$$t_{ac1(RD-DB)} = \frac{10^9}{f(BCLK) \times 2} - 90 \quad [ns]$$

$$t_{ac2(RD-DB)} = \frac{3 \times 10^9}{f(BCLK) \times 2} - 90 \quad [ns]$$

$$t_{ac3(RD-DB)} = \frac{3 \times 10^9}{f(BCLK) \times 2} - 90 \quad [ns]$$

Timing requirements

(referenced to V_{CC} = 3V, V_{SS} = 0V, at Topr = – 20°C to 85°C / – 40°C to 85°C (*) unless otherwise specified)

* : Specify a product of –40°C to 85°C to use it.

Table 1.31.10. Timer A input (counter input in event counter mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TA)}$	TAiIN input cycle time	150		ns
$t_{w(TAH)}$	TAiIN input HIGH pulse width	60		ns
$t_{w(TAL)}$	TAiIN input LOW pulse width	60		ns

Table 1.31.11. Timer A input (gating input in timer mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TA)}$	TAiIN input cycle time	600		ns
$t_{w(TAH)}$	TAiIN input HIGH pulse width	300		ns
$t_{w(TAL)}$	TAiIN input LOW pulse width	300		ns

Table 1.31.12. Timer A input (external trigger input in one-shot timer mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TA)}$	TAiIN input cycle time	300		ns
$t_{w(TAH)}$	TAiIN input HIGH pulse width	150		ns
$t_{w(TAL)}$	TAiIN input LOW pulse width	150		ns

Table 1.31.13. Timer A input (external trigger input in pulse width modulation mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{w(TAH)}$	TAiIN input HIGH pulse width	150		ns
$t_{w(TAL)}$	TAiIN input LOW pulse width	150		ns

Table 1.31.14. Timer A input (up/down input in event counter mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(UP)}$	TAiOUT input cycle time	3000		ns
$t_{w(UPH)}$	TAiOUT input HIGH pulse width	1500		ns
$t_{w(UPL)}$	TAiOUT input LOW pulse width	1500		ns
$t_{su(UP-TIN)}$	TAiOUT input setup time	600		ns
$t_{h(TIN-UP)}$	TAiOUT input hold time	600		ns

Timing requirements

(referenced to V_{CC} = 3V, V_{SS} = 0V, at Topr = – 20°C to 85°C / – 40°C to 85°C (*) unless otherwise specified)

* : Specify a product of –40°C to 85°C to use it.

Table 1.31.15. Timer B input (counter input in event counter mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TB)}$	TBiIN input cycle time (counted on one edge)	150		ns
$t_{w(TBH)}$	TBiIN input HIGH pulse width (counted on one edge)	60		ns
$t_{w(TBL)}$	TBiIN input LOW pulse width (counted on one edge)	60		ns
$t_{c(TB)}$	TBiIN input cycle time (counted on both edges)	300		ns
$t_{w(TBH)}$	TBiIN input HIGH pulse width (counted on both edges)	160		ns
$t_{w(TBL)}$	TBiIN input LOW pulse width (counted on both edges)	160		ns

Table 1.31.16. Timer B input (pulse period measurement mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TB)}$	TBiIN input cycle time	600		ns
$t_{w(TBH)}$	TBiIN input HIGH pulse width	300		ns
$t_{w(TBL)}$	TBiIN input LOW pulse width	300		ns

Table 1.31.17. Timer B input (pulse width measurement mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TB)}$	TBiIN input cycle time	600		ns
$t_{w(TBH)}$	TBiIN input HIGH pulse width	300		ns
$t_{w(TBL)}$	TBiIN input LOW pulse width	300		ns

Table 1.31.18. A-D trigger input

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(AD)}$	$\overline{ADTRG}$ input cycle time (trigger able minimum)	1500		ns
$t_{w(ADL)}$	$\overline{ADTRG}$ input LOW pulse width	200		ns

Table 1.31.19. Serial I/O

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(CK)}$	CLKi input cycle time	300		ns
$t_{w(CKH)}$	CLKi input HIGH pulse width	150		ns
$t_{w(CKL)}$	CLKi input LOW pulse width	150		ns
$t_{d(C-Q)}$	TxDi output delay time		160	ns
$t_{h(C-Q)}$	TxDi hold time	0		ns
$t_{su(D-C)}$	RxDi input setup time	50		ns
$t_{h(C-D)}$	RxDi input hold time	90		ns

Table 1.31.20. External interrupt $\overline{INTi}$ inputs

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{w(INH)}$	$\overline{INTi}$ input HIGH pulse width	380		ns
$t_{w(INL)}$	$\overline{INTi}$ input LOW pulse width	380		ns

Switching characteristics (referenced to V_{CC} = 3V, V_{SS} = 0V at Topr = – 20°C to 85°C / – 40°C to 85°C (Note 3), CM15 = "1" unless otherwise specified)

Table 1.31.21. Memory expansion and microprocessor modes (with no wait)

Symbol	Parameter	Measuring condition	Standard		Unit
			Min.	Max.	
$t_{d(BCLK-AD)}$	Address output delay time	Figure 1.31.1		60	ns
$t_{h(BCLK-AD)}$	Address output hold time (BCLK standard)		4		ns
$t_{h(RD-AD)}$	Address output hold time (RD standard)		0		ns
$t_{h(WR-AD)}$	Address output hold time (WR standard)		0		ns
$t_{d(BCLK-CS)}$	Chip select output delay time			60	ns
$t_{h(BCLK-CS)}$	Chip select output hold time (BCLK standard)		4		ns
$t_{d(BCLK-ALE)}$	ALE signal output delay time			60	ns
$t_{h(BCLK-ALE)}$	ALE signal output hold time		– 4		ns
$t_{d(BCLK-RD)}$	RD signal output delay time			60	ns
$t_{h(BCLK-RD)}$	RD signal output hold time		0		ns
$t_{d(BCLK-WR)}$	WR signal output delay time			60	ns
$t_{h(BCLK-WR)}$	WR signal output hold time		0		ns
$t_{d(BCLK-DB)}$	Data output delay time (BCLK standard)			80	ns
$t_{h(BCLK-DB)}$	Data output hold time (BCLK standard)		4		ns
$t_{d(DB-WR)}$	Data output delay time (WR standard)		(Note1)		ns
$t_{h(WR-DB)}$	Data output hold time (WR standard)(Note2)		0		ns

Note 1: Calculated according to the BCLK frequency as follows:

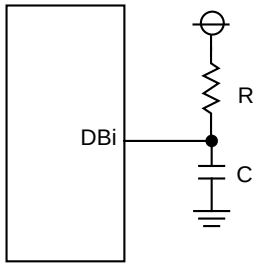
$$td(DB - WR) = \frac{10^9}{f(BCLK) \times 2} - 80 \quad [ns]$$

Note 2: This is standard value shows the timing when the output is off, and doesn't show hold time of data bus.
Hold time of data bus is different by capacitor volume and pull-up (pull-down) resistance value.
Hold time of data bus is expressed in

$$t = -CR \times \ln(1 - V_{OL} / V_{CC})$$

by a circuit of the right figure.
For example, when $V_{OL} = 0.2V_{CC}$, C = 30pF, R = 1kΩ, hold time of output "L" level is

$$t = -30pF \times 1k\Omega \times \ln(1 - 0.2V_{CC} / V_{CC}) = 6.7ns.$$



Note 3: Specify a product of –40°C to 85°C to use it.

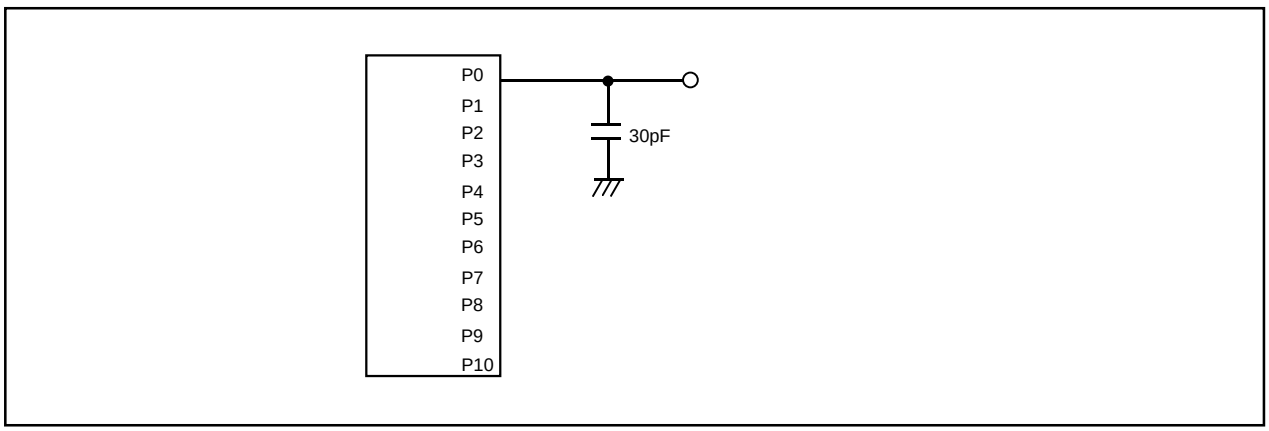


Figure 1.31.1. Port P0 to P10 measurement circuit

Switching characteristics (referenced to V_{CC} = 3V, V_{SS} = 0V at Topr = – 20°C to 85°C / – 40°C to 85°C (Note 3), CM15 = "1" unless otherwise specified)

Table 1.31.22. Memory expansion and microprocessor modes (when accessing external memory area with wait)

Symbol	Parameter	Measuring condition	Standard		Unit
			Min.	Max.	
$t_{d(BCLK-AD)}$	Address output delay time	Figure 1.31.1		60	ns
$t_{h(BCLK-AD)}$	Address output hold time (BCLK standard)		4		ns
$t_{h(RD-AD)}$	Address output hold time (RD standard)		0		ns
$t_{h(WR-AD)}$	Address output hold time (WR standard)		0		ns
$t_{d(BCLK-CS)}$	Chip select output delay time			60	ns
$t_{h(BCLK-CS)}$	Chip select output hold time (BCLK standard)		4		ns
$t_{d(BCLK-ALE)}$	ALE signal output delay time			60	ns
$t_{h(BCLK-ALE)}$	ALE signal output hold time		– 4		ns
$t_{d(BCLK-RD)}$	RD signal output delay time			60	ns
$t_{h(BCLK-RD)}$	RD signal output hold time		0		ns
$t_{d(BCLK-WR)}$	WR signal output delay time			60	ns
$t_{h(BCLK-WR)}$	WR signal output hold time		0		ns
$t_{d(BCLK-DB)}$	Data output delay time (BCLK standard)			80	ns
$t_{h(BCLK-DB)}$	Data output hold time (BCLK standard)		4		ns
$t_{d(DB-WR)}$	Data output delay time (WR standard)		(Note1)		ns
$t_{h(WR-DB)}$	Data output hold time (WR standard)(Note2)		0		ns

Note 1: Calculated according to the BCLK frequency as follows:

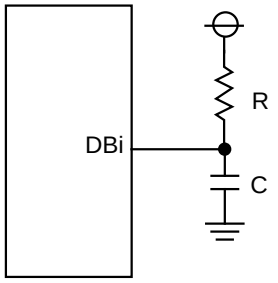
$$td(DB - WR) = \frac{10^9}{f(BCLK)} - 80 \quad [ns]$$

Note 2: This is standard value shows the timing when the output is off, and doesn't show hold time of data bus.
Hold time of data bus is different by capacitor volume and pull-up (pull-down) resistance value.
Hold time of data bus is expressed in

$t = –CR \times \ln (1 – V_{OL} / V_{CC})$

by a circuit of the right figure.
For example, when $V_{OL} = 0.2V_{CC}$, C = 30pF, R = 1kΩ, hold time of output "L" level is

$t = – 30pF \times 1k\Omega \times \ln (1 – 0.2V_{CC} / V_{CC})$
$= 6.7ns.$



Note 3: Specify a product of –40°C to 85°C to use it.

Switching characteristics (referenced to V_{CC} = 3V, V_{SS} = 0V at Topr = – 20°C to 85°C / – 40°C to 85°C (Note 2), CM15 = "1" unless otherwise specified)

Table 1.31.23. Memory expansion and microprocessor modes (when accessing external memory area with wait, and select multiplexed bus)

Symbol	Parameter	Measuring condition	Standard		Unit
			Min.	Max.	
$t_{d(BCLK-AD)}$	Address output delay time	Figure 1.31.1		60	ns
$t_{h(BCLK-AD)}$	Address output hold time (BCLK standard)		4		ns
$t_{h(RD-AD)}$	Address output hold time (RD standard)		(Note 1)		ns
$t_{h(WR-AD)}$	Address output hold time (WR standard)		(Note 1)		ns
$t_{d(BCLK-CS)}$	Chip select output delay time			60	ns
$t_{h(BCLK-CS)}$	Chip select output hold time (BCLK standard)		4		ns
$t_{h(RD-CS)}$	Chip select output hold time (RD standard)		(Note 1)		ns
$t_{h(WR-CS)}$	Chip select output hold time (WR standard)		(Note 1)		ns
$t_{d(BCLK-RD)}$	RD signal output delay time			60	ns
$t_{h(BCLK-RD)}$	RD signal output hold time		0		ns
$t_{d(BCLK-WR)}$	WR signal output delay time			60	ns
$t_{h(BCLK-WR)}$	WR signal output hold time		0		ns
$t_{d(BCLK-DB)}$	Data output delay time (BCLK standard)			80	ns
$t_{h(BCLK-DB)}$	Data output hold time (BCLK standard)		4		ns
$t_{d(DB-WR)}$	Data output delay time (WR standard)		(Note 1)		ns
$t_{h(WR-DB)}$	Data output hold time (WR standard)		(Note 1)		ns
$t_{d(BCLK-ALE)}$	ALE signal output delay time (BCLK standard)			60	ns
$t_{h(BCLK-ALE)}$	ALE signal output hold time (BCLK standard)		– 4		ns
$t_{d(AD-ALE)}$	ALE signal output delay time (Address standard)		(Note 1)		ns
$t_{h(ALE-AD)}$	ALE signal output hold time(Address standard)		40		ns
$t_{d(AD-RD)}$	Post-address RD signal output delay time		0		ns
$t_{d(AD-WR)}$	Post-address WR signal output delay time		0		ns
$t_{dZ(RD-AD)}$	Address output floating start time			8	ns

Note 1: Calculated according to the BCLK frequency as follows:

$$th(RD - AD) = \frac{10^9}{f(BCLK) \times 2} \quad [ns]$$

$$th(WR - AD) = \frac{10^9}{f(BCLK) \times 2} \quad [ns]$$

$$th(RD - CS) = \frac{10^9}{f(BCLK) \times 2} \quad [ns]$$

$$th(WR - CS) = \frac{10^9}{f(BCLK) \times 2} \quad [ns]$$

$$td(DB - WR) = \frac{10^9 \times 3}{f(BCLK) \times 2} - 80 \quad [ns]$$

$$th(WR - DB) = \frac{10^9}{f(BCLK) \times 2} \quad [ns]$$

$$td(AD - ALE) = \frac{10^9}{f(BCLK) \times 2} - 45 \quad [ns]$$

Note 2: Specify a product of –40°C to 85°C to use it.

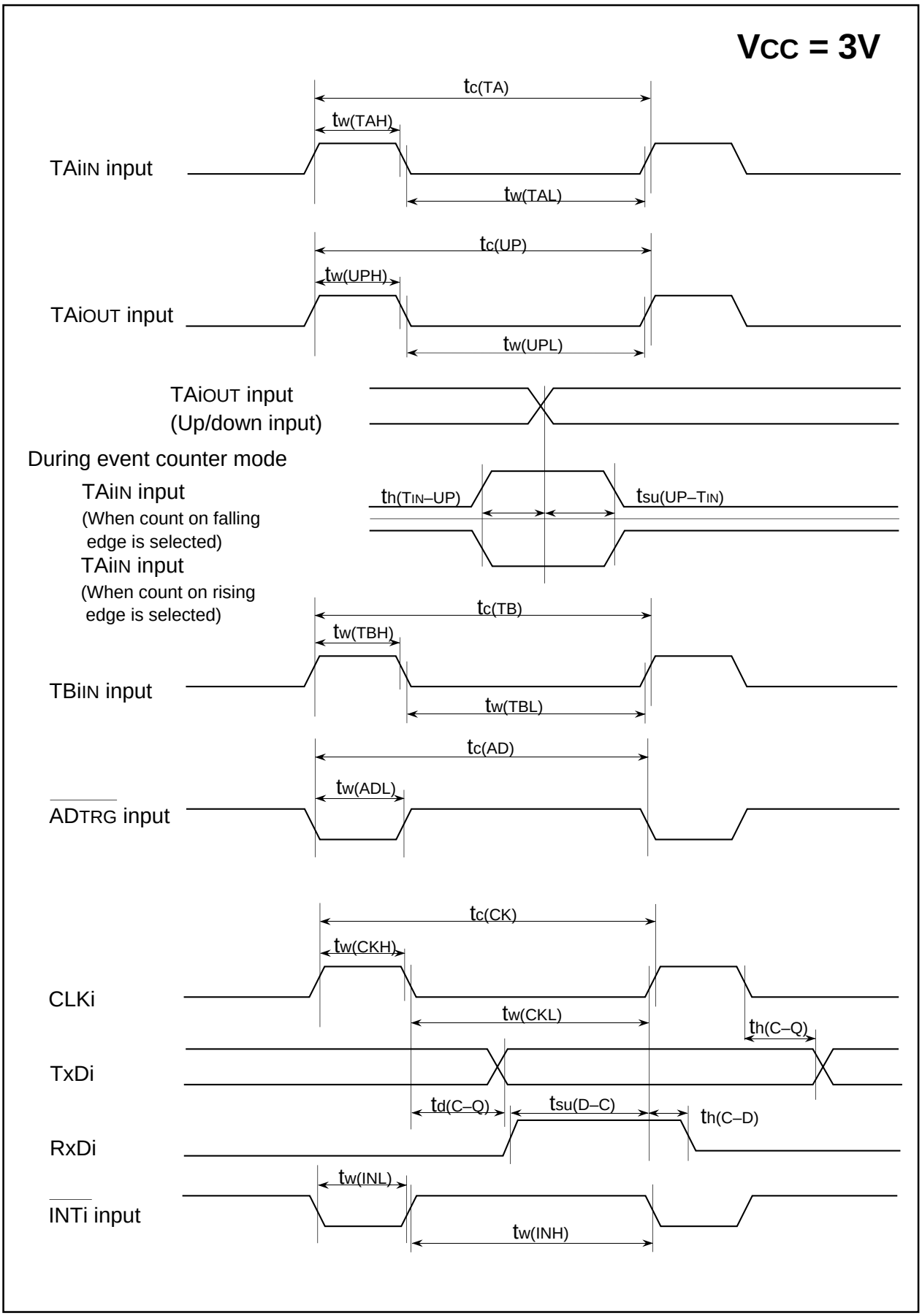
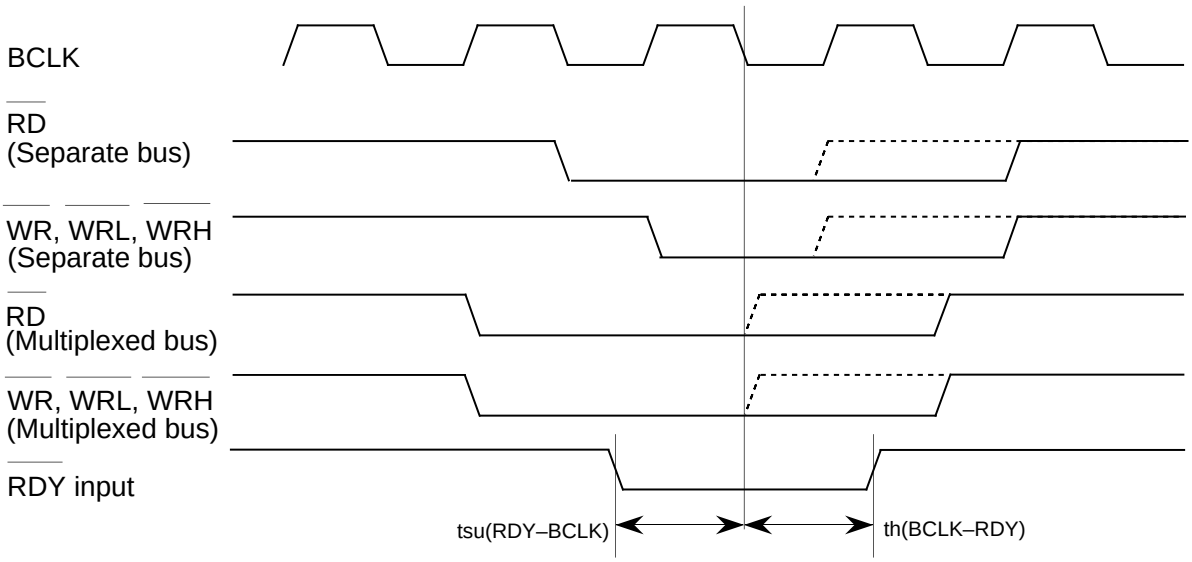


Figure 1.31.2. Vcc=3V timing diagram (1)

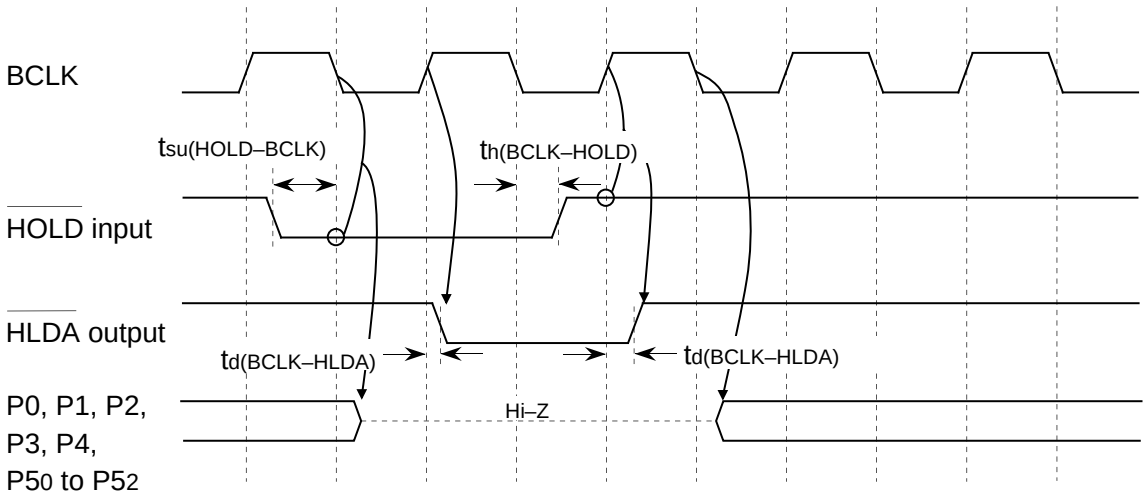
Vcc = 3V

Memory Expansion Mode and Microprocessor Mode

(Valid only with wait)



(Valid with or without wait)



Note: The above pins are set to high-impedance regardless of the input level of the BYTE pin and bit (PM06) of processor mode register 0 selects the function of ports P40 to P43.

Measuring conditions :

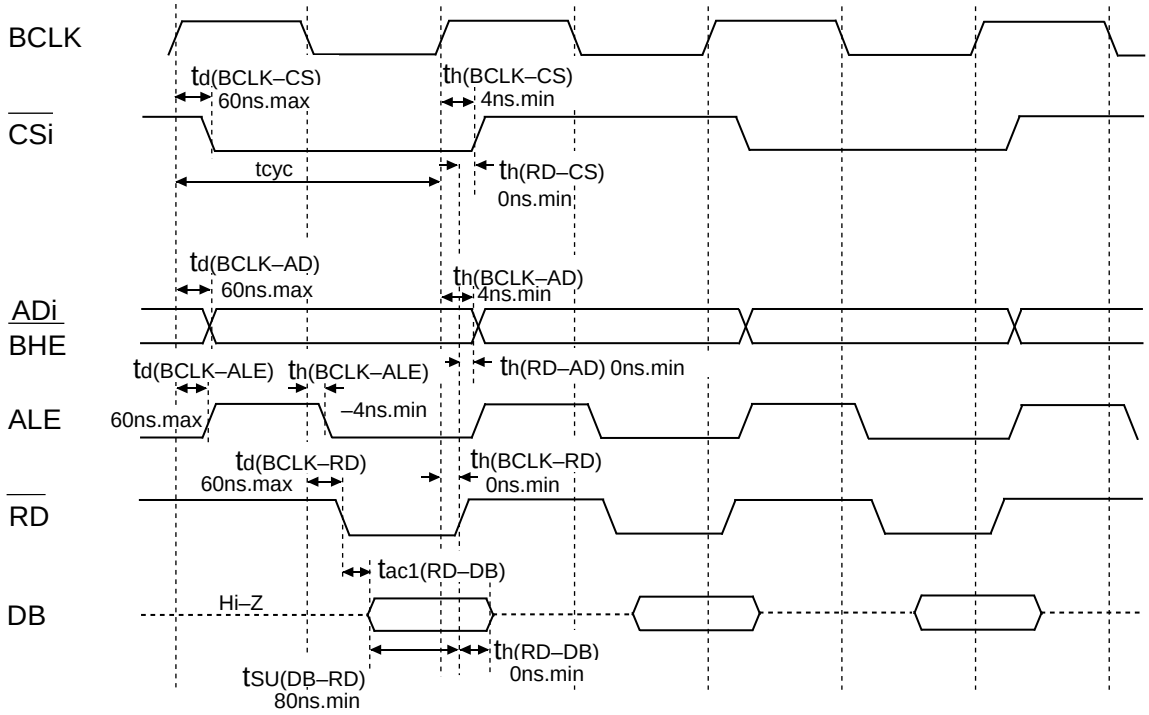
- Vcc=3V
- Input timing voltage : Determined with VIL=0.6V, VIH=2.4V
- Output timing voltage : Determined with VOL=1.5V, VOH=1.5V

Figure 1.31.3. Vcc=3V timing diagram (2)

Vcc = 3V

Memory Expansion Mode and Microprocessor Mode
(With no wait)

Read timing



Write timing

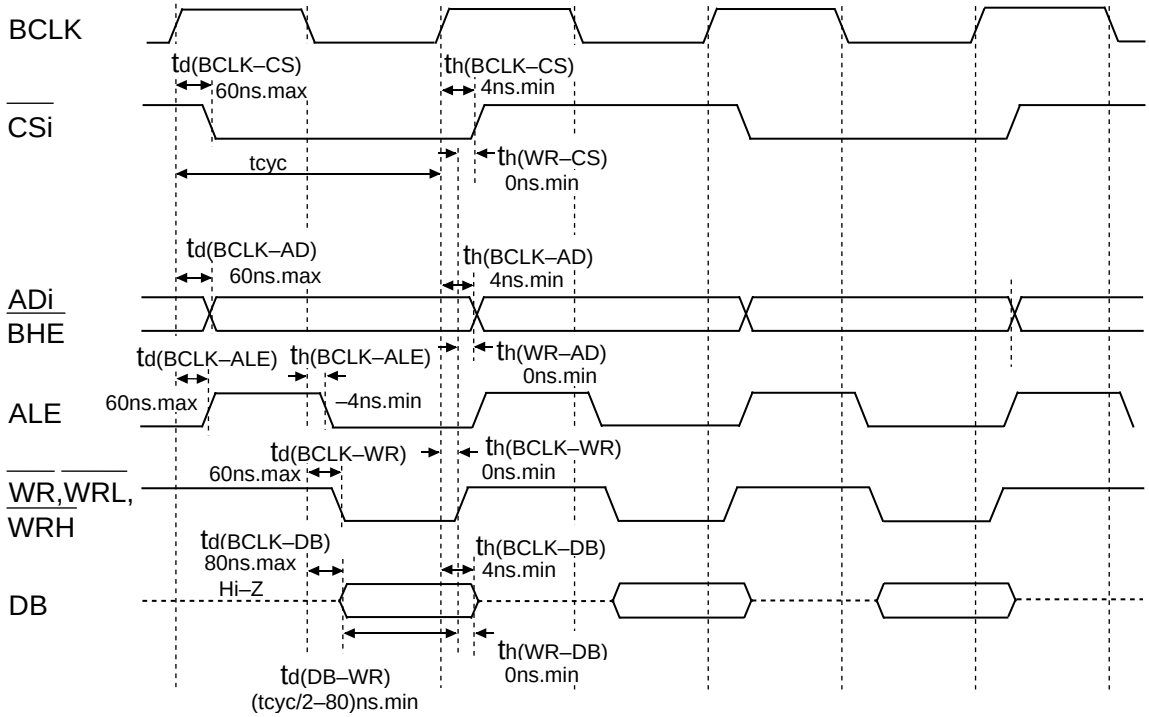


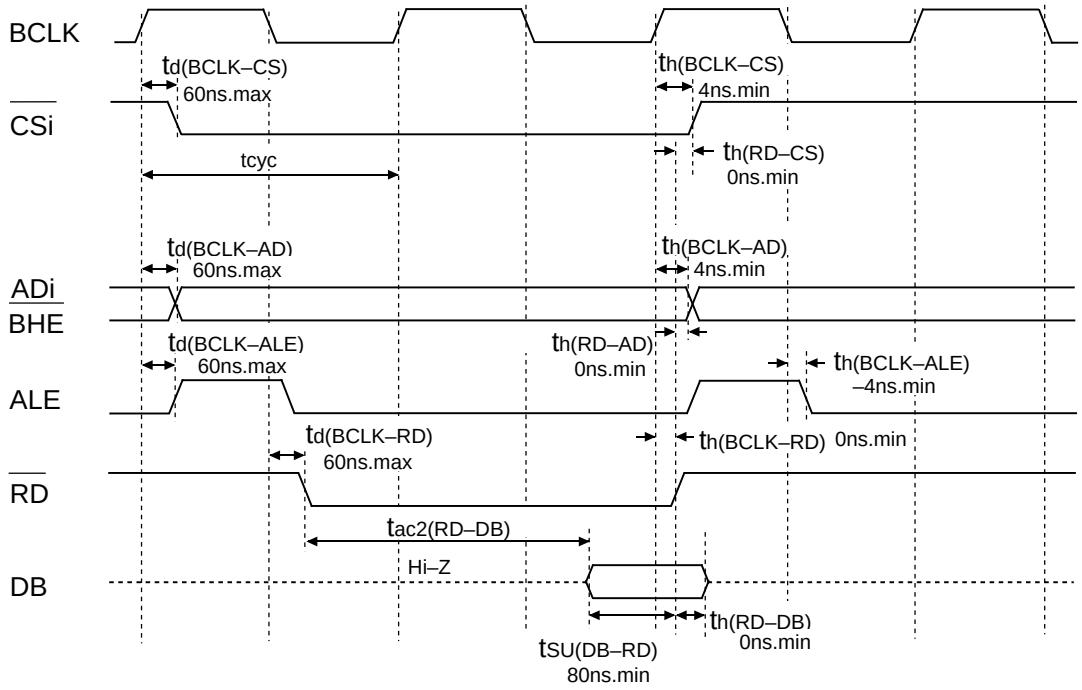
Figure 1.31.4. Vcc=3V timing diagram (3)

Vcc = 3V

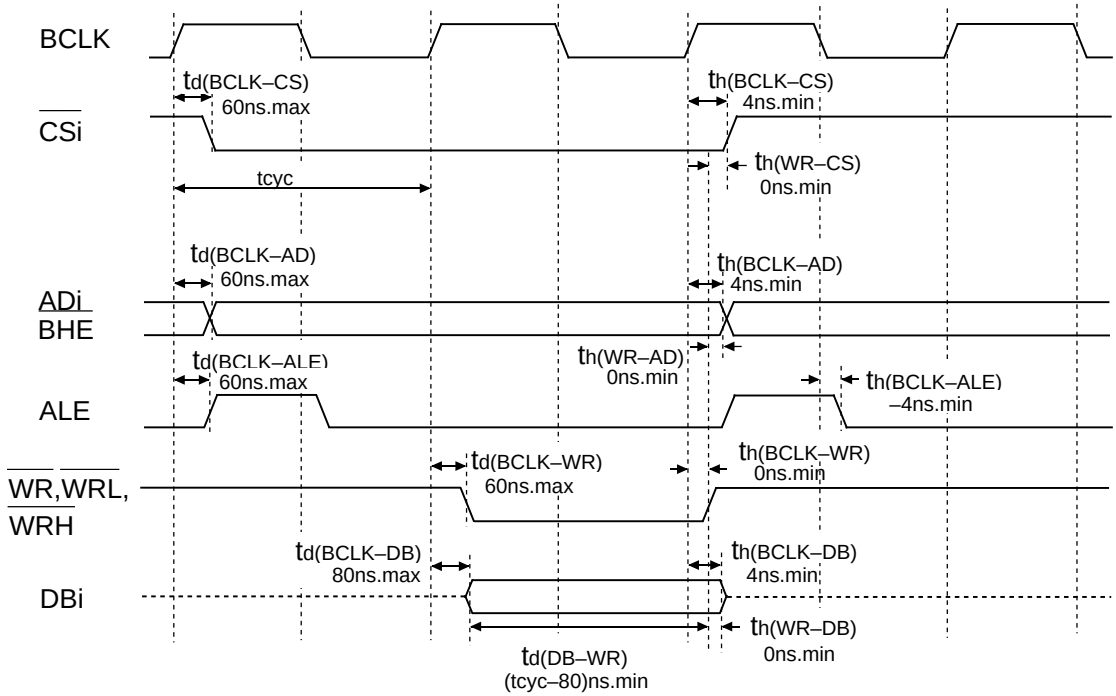
Memory Expansion Mode and Microprocessor Mode

(When accessing external memory area with wait)

Read timing



Write timing



Measuring conditions :

- V_{CC}=3V
- Input timing voltage : Determined with V_{IL}=0.48V, V_{IH}=1.5V
- Output timing voltage : Determined with V_{OL}=1.5V, V_{OH}=1.5V

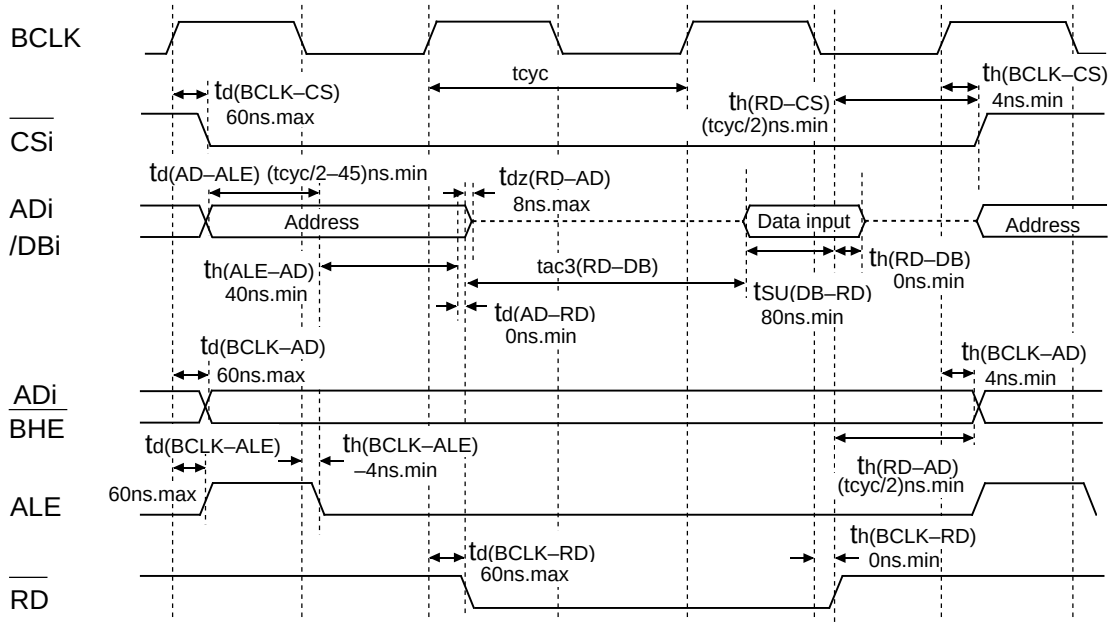
Figure 1.31.5. V_{CC}=3V timing diagram (4)

Vcc = 3V

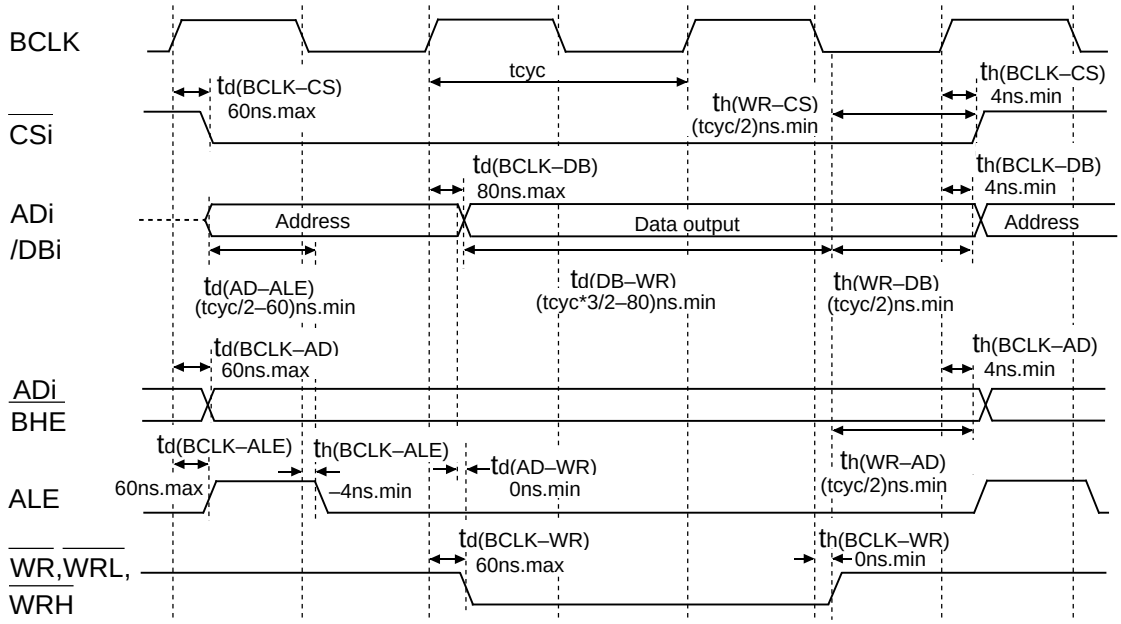
Memory Expansion Mode and Microprocessor Mode

(When accessing external memory area with wait, and select multiplexed bus)

Read timing



Write timing



Measuring conditions :
- Vcc=3V
- Input timing voltage : Determined with VIL=0.48V,VIH=1.5V
- Output timing voltage : Determined with VOL=1.5V,VOH=1.5V

Figure 1.31.6. Vcc=3V timing diagram (5)

GZZ-SH13-95B<02A0>

Mask ROM number	

MITSUBISHI ELECTRIC-CHIP 16-BIT MICROCOMPUTER M30620MCM-XXXFP/GP MASK ROM CONFIRMATION FORM

Receipt	Date :	
	Section head signature	Supervisor signature

Note : Please complete all items marked ✵ .

✵	Customer	Company name	TEL ( )	Issuance signature	Submitted by	Supervisor
		Date issued	Date :			

✵1. Check sheet

Mitsubishi processes the mask files generated by the mask file generation utilities out of those held on the floppy disks you give in to us, and forms them into masks. Hence, we assume liability provided that there is any discrepancy between the contents of these mask files and the ROM data to be burned into products we produce. Check thoroughly the contents of the mask files you give in.
Prepare 3.5 inches 2HD (IBM format) floppy disks. And store only one mask file in a floppy disk.

Microcomputer type No. : ☐ M30620MCM-XXXFP ☐ M30620MCM-XXXGP

File code : ☐☐☐☐☐☐☐☐ (hex)

Mask file name : ☐☐☐☐☐☐☐☐ .MSK (alpha-numeric 8-digit)

✵2. Mark specification

The mark specification differs according to the type of package. After entering the mark specification on the separate mark specification sheet (for each package), attach that sheet to this masking check sheet for submission to Mitsubishi.

For the M30620MCM-XXXFP, submit the 100P6S mark specification sheet. For the M30620MCM-XXXGP, submit the 100P6Q mark specification sheet.

✵3. Usage Conditions

For our reference when of testing our products, please reply to the following questions about the usage of the products you ordered.

(1) Which kind of XIN-XOUT oscillation circuit is used?

☐ Ceramic resonator ☐ Quartz-crystal oscillator
☐ External clock input ☐ Other ( )

What frequency do not use?

f(XIN) = ☐ MHz

GZZ-SH13-95B<02A0>

Mask ROM number	

MITSUBISHI ELECTRIC-CHIP 16-BIT MICROCOMPUTER M30620MCM-XXXFP/GP MASK ROM CONFIRMATION FORM

(2) Which kind of XCIN-XCOUT oscillation circuit is used?

☐ Ceramic resonator ☐ Quartz-crystal oscillator

☐ External clock input ☐ Other ( )

What frequency do not use?

f(XCIN) = ☐ kHz

(3) Which operation mode do you use?

☐ Single-chip mode ☐ Memory expansion mode

☐ Microprocessor mode

(4) Which operating supply voltage do you use?

(Circle the operating voltage range of use)

2.2 2.4 2.6 2.7 2.8 2.9 3.0 3.1 3.2 3.3 3.4 3.5 3.6 3.7 3.8 (V)

(5) Which operating ambient temperature do you use?

(Circle the operating temperature range of use)

-50 -40 -30 -20 -10 0 10 20 30 40 50 60 70 80 90 (°C)

(6) Do you use I^2C (Inter IC) bus function?

☐ Not use ☐ Use

(7) Do you use IE (Inter Equipment) bus function?

☐ Not use ☐ Use

Thank you cooperation.

✵4. Special item (Indicate none if there is not specified item)

GZZ-SH13-48B<98A1>

Mask ROM number	

MITSUBISHI ELECTRIC-CHIP 16-BIT MICROCOMPUTER M30624MGM-XXXFP/GP MASK ROM CONFIRMATION FORM

Receipt	Date :	
	Section head signature	Supervisor signature

Note : Please complete all items marked ✵ .

✵	Customer	Company name	TEL ( )	Issuance signature	Submitted by	Supervisor
		Date issued	Date :			

✵1. Check sheet

Mitsubishi processes the mask files generated by the mask file generation utilities out of those held on the floppy disks you give in to us, and forms them into masks. Hence, we assume liability provided that there is any discrepancy between the contents of these mask files and the ROM data to be burned into products we produce. Check thoroughly the contents of the mask files you give in.
Prepare 3.5 inches 2HD (IBM format) floppy disks. And store only one mask file in a floppy disk.

Microcomputer type No. : ☐ M30624MGM-XXXFP ☐ M30624MGM-XXXGP

File code : ☐☐☐☐☐☐☐☐ (hex)

Mask file name : ☐☐☐☐☐☐☐☐ .MSK (alpha-numeric 8-digit)

✵2. Mark specification

The mark specification differs according to the type of package. After entering the mark specification on the separate mark specification sheet (for each package), attach that sheet to this masking check sheet for submission to Mitsubishi.

For the M30624MGM-XXXFP, submit the 100P6S mark specification sheet. For the M30624MGM-XXXGP, submit the 100P6Q mark specification sheet.

✵3. Usage Conditions

For our reference when of testing our products, please reply to the following questions about the usage of the products you ordered.

(1) Which kind of XIN-XOUT oscillation circuit is used?

☐ Ceramic resonator ☐ Quartz-crystal oscillator
☐ External clock input ☐ Other ( )

What frequency do not use?

f(XIN) = ☐ MHz

GZZ-SH13-48B<98A1>

Mask ROM number	

MITSUBISHI ELECTRIC-CHIP 16-BIT MICROCOMPUTER M30624MGM-XXXFP/GP MASK ROM CONFIRMATION FORM

(2) Which kind of XCIN-XCOUT oscillation circuit is used?

☐ Ceramic resonator ☐ Quartz-crystal oscillator

☐ External clock input ☐ Other ( )

What frequency do not use?

f(XCIN) = ☐ kHz

(3) Which operation mode do you use?

☐ Single-chip mode ☐ Memory expansion mode

☐ Microprocessor mode

(4) Which operating supply voltage do you use?

(Circle the operating voltage range of use)

2.2 2.4 2.6 2.7 2.8 2.9 3.0 3.1 3.2 3.3 3.4 3.5 3.6 3.7 3.8 (V)

(5) Which operating ambient temperature do you use?

(Circle the operating temperature range of use)

-50 -40 -30 -20 -10 0 10 20 30 40 50 60 70 80 90 (°C)

(6) Do you use I^2C (Inter IC) bus function?

☐ Not use ☐ Use

(7) Do you use IE (Inter Equipment) bus function?

☐ Not use ☐ Use

Thank you cooperation.

❋4. Special item (Indicate none if there is not specified item)

[MEMO]

Chapter 2

Peripheral Functions Usage

2.1 Protect Usage

2.1.1 Overview of the protect usage

'Protect' is a function that causes a value held in a register to be unchanged even when a program runs away. The following is an overview of the protect usage:

(1) Registers affected by the protect function

The registers affected by the protect function are:

(a) System clock control registers 0, 1 (addresses 0006_{16} and 0007_{16})

(b) Processor mode registers 0, 1 (addresses 0004_{16} and 0005_{16})

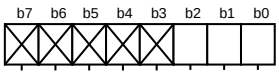
(c) Port P9 direction register (address $03F3_{16}$), SI/Oi control register (i=3,4)(addresses 0362_{16} and 0366_{16})

The values in registers (1) through (3) cannot be changed in write-protect state. To change values in the registers, put the individual registers in write-enabled state.

(2) Protect register

Figure 2.1.1 shows protect register.

Protect register



Symbol	Address	When reset
PRCR	$000A_{16}$	$XXXXX000_2$

Bit symbol	Bit name	Function	R	W
PRC0	Enables writing to system clock control registers 0 and 1 (addresses 0006_{16} and 0007_{16})	0 : Write-inhibited 1 : Write-enabled	○	○
PRC1	Enables writing to processor mode registers 0 and 1 (addresses 0004_{16} and 0005_{16})	0 : Write-inhibited 1 : Write-enabled	○	○
PRC2	Enables writing to port P9 direction register (address $03F3_{16}$) and SI/Oi control register (i=3,4) (addresses 0362_{16} and 0366_{16}) (Note)	0 : Write-inhibited 1 : Write-enabled	○	○
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be indeterminate.			–	–

Note: Writing a value to an address after “1” is written to this bit returns the bit to “0” . Other bits do not automatically return to “0” and they must therefore be reset by the program.

Figure 2.1.1. Protect register

2.1.2 Protect Operation

The following explains the protect operation. Figure 2.1.2 shows the set-up procedure.

Operation (1) Setting “1” in the write-enable bit of system clock control registers 0 and 1 causes system clock control register 0 and system clock control register 1 to be in write-enabled state.

(2) The contents of system clock control register 0 and that of system clock control register 1 are changed.

(3) Setting “0” in the write-enable bit of system control registers 0 and 1 causes system clock control register 0 and system control register 1 to be in write-inhibited state.

(4) To change the contents of processor mode register 0 and that of processor mode register 1, follow the same steps as in dealing with system clock control registers.

(5) The write-enable bit of port 9 direction register and SI/Oi control register (i=3,4) goes to “0” when the next write instruction is executed after write-enabled state is readied. Make changes in input/output and SI/Oi control register (i=3,4) immediately after the instruction that sets “1” in the write-enable bit of port P9 direction register and SI/Oi control register (i=3,4)(avoid causing an interrupt). Also take measures to prevent DMA transfer from being executed.

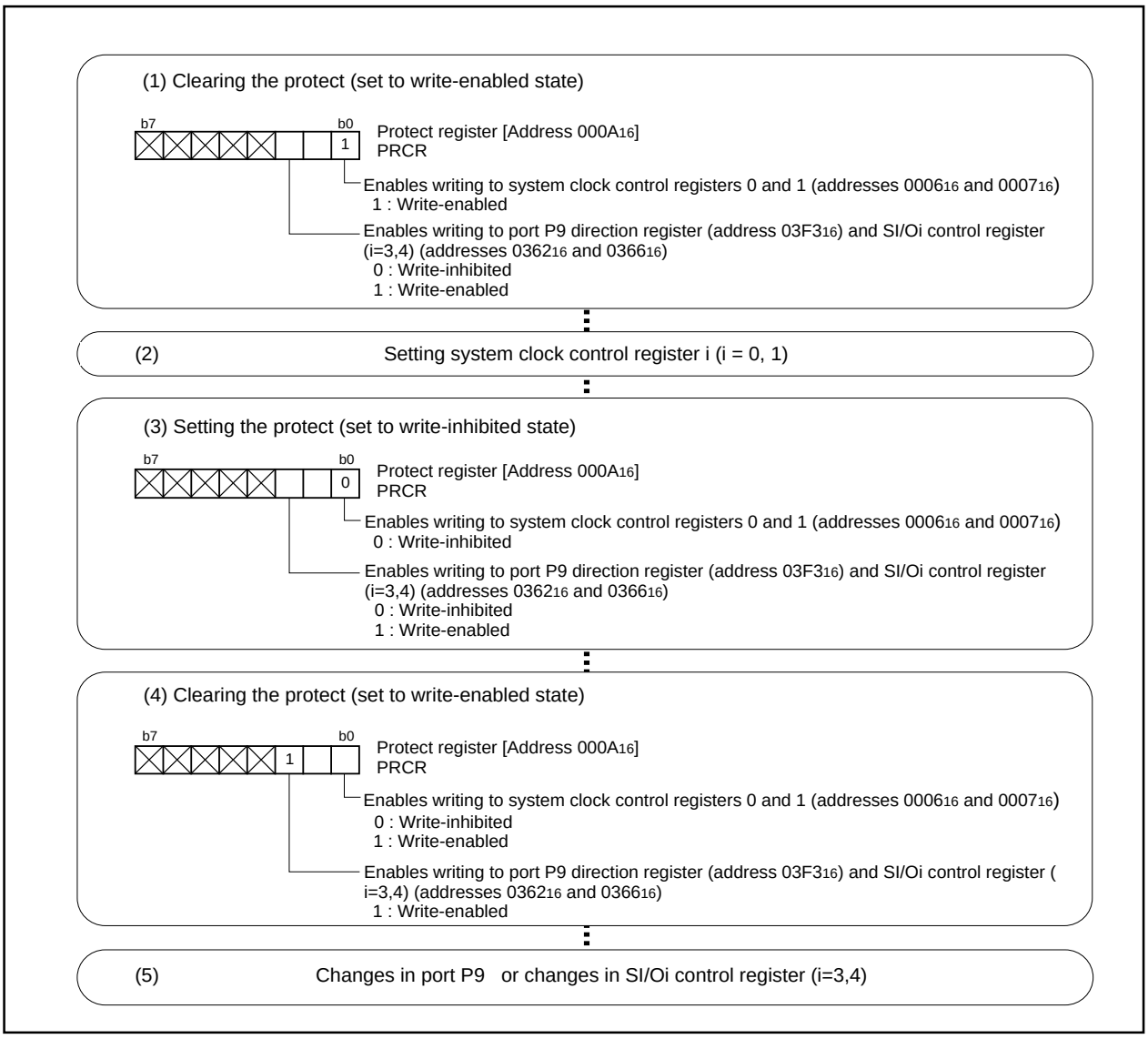


Figure 2.1.2. Set-up procedure for protect function

2.1.3 Precaution for Protect

(1) The write-enable bit of port 9 direction register and SI/Oi control register (i=3,4) goes to “0” when the next write instruction is executed after write-enabled state is readied. Make changes in input/output and SI/Oi control register (i=3,4) immediately after the instruction that sets “1” in the write-enable bit of port P9 direction register and SI/Oi control register (i=3,4)(avoid causing an interrupt). Also take measures to prevent DMA transfer from being executed.

2.2 Timer A Usage

2.2.1 Overview of the timer A usage

Timer A is a 16-bit timer. The following is an overview of the timer A usage.

(1) Mode

Timer A operates in one of the four modes:

(a) Timer mode

In this mode, the internal count source is counted. Two functions can be selected: the pulse output function that reverses output from a port every time an overflow occurs, or the gate function which controls the count start/stop according to the input signal from a port.

• Timer mode operation P2-12
• Timer mode, gate function operation P2-14
• Timer mode, pulse output function operation P2-16

(b) Event counter mode

This mode counts the pulses from the outside and the number of overflows in other timers. The free-run type, in which nothing is reloaded from the reload register, can be selected when an underflow occurs. The pulse output function can also be selected. Please refer to the timer mode explanation for details, as the operation is identical.

• Event counter mode operation P2-18
• Event counter mode, free run type operation P2-20

Furthermore, Timer A has a 2-phase pulse signal processing function which generates an up count or down count in the event counter mode, depending on the phase of the two input signals.

• Operation of the 2-phase pulse signal processing function in normal event counter mode P2-22
• Operation of the 2-phase pulse signal processing function in 4-multiplication mode P2-24

(c) One-shot timer mode

In this mode, the timer is started by the trigger and stops when the timer goes to “0”. The trigger can be selected from the following 3 types: an external input signal, an overflow of the timer, or a software trigger. The pulse output function can also be selected. Please refer to the timer mode explanation for details, as the operation is identical.

• Operation in one-shot timer mode effected by software P2-26
• Operation in one-shot timer mode effected by an external trigger P2-28

(d) Pulse width modulation (PWM) mode

In this mode, the arbitrary pulses are successively output. Either a 16-bit fixed-period PWM mode or 8-bit variable-period mode can be selected. The trigger for initiating output can also be selected. Please refer to the one-shot timer mode explanation for details, as the operation is identical.

• 16-bit PWM mode operation P2-30
• 8-bit PWM mode operation P2-32

(2) Count source

The internal count source can be selected from f1, f_8, f_{32}, and f_{C32}. Clocks f_1, f_8, and f_{32} are derived by dividing the CPU's main clock by 1, 8, and 32 respectively. Clock f_{C32} is derived by dividing the CPU's secondary clock by 32.

(3) Count value

In timer mode or pulse width modulation mode, [the value set in the timer register + 1] becomes the count value. In event counter mode, [the set value + 1] becomes the count value when a down count is performed, or [$FFFF_{16}$ - the set value + 1] becomes the count value when an up count is performed. In one-shot timer mode, the value set in the timer register becomes the count value.

The counter overflows (or underflows) when a count source equal to a count value is input, and an interrupt occurs. For the pulse output function, the output from the port varies (the value in the port register does not vary).

(4) Reading the timer

Either in timer mode or in event counter mode, reading the timer register takes out the count at that moment. Read it in 16-bit units. The data either in one-shot timer mode or in pulse width modulation mode is indeterminate.

(5) Writing to the timer

To write to the timer register when a count is in progress, the value is written only to the reload register. When writing to the timer register when a count is stopped, the value is written both to the reload register and to the counter. Write a value in 16-bit units.

(6) Relation between the input/output to/from the timer and the direction register

With the output function of the timer, pulses are output regardless of the direction register of the relevant port. To input an external signal to the timer, set the direction register of the relevant port to input.

(7) Pins related to timer A

(a) $TA0_{IN}$, $TA1_{IN}$, $TA2_{IN}$, $TA3_{IN}$, $TA4_{IN}$	Input pins to timer A.
(b) $TA0_{OUT}$, $TA1_{OUT}$, $TA2_{OUT}$, $TA3_{OUT}$, $TA4_{OUT}$	Output pins from timer A. They become input pins to timer A when event counter mode is active.

(8) Registers related to timer A

Figure 2.2.1 shows the memory map of timer A-related registers. Figures 2.2.2 through 2.2.5 show timer A-related registers.

0055₁₆	Timer A0 interrupt control register (TA0IC)
0056₁₆	Timer A1 interrupt control register (TA1IC)
0057₁₆	Timer A2 interrupt control register (TA2IC)
0058₁₆	Timer A3 interrupt control register (TA3IC)
0059₁₆	Timer A4 interrupt control register (TA4IC)
≈	≈
0380₁₆	Count start flag (TABSR)
0381₁₆	Clock prescaler reset flag (CPSRF)
0382₁₆	One-shot start flag (ONSF)
0383₁₆	Trigger select register (TRGSR)
0384₁₆	Up-down flag (UDF)
0385₁₆	
0386₁₆ 0387₁₆	Timer A0 register (TA0)
0388₁₆ 0389₁₆	Timer A1 register (TA1)
038A₁₆ 038B₁₆	Timer A2 register (TA2)
038C₁₆ 038D₁₆	Timer A3 register (TA3)
038E₁₆ 038F₁₆	Timer A4 register (TA4)
≈	≈
0396₁₆	Timer A0 mode register (TA0MR)
0397₁₆	Timer A1 mode register (TA1MR)
0398₁₆	Timer A2 mode register (TA2MR)
0399₁₆	Timer A3 mode register (TA3MR)
039A₁₆	Timer A4 mode register (TA4MR)

Figure 2.2.1. Memory map of timer A-related registers

Timer Ai mode register

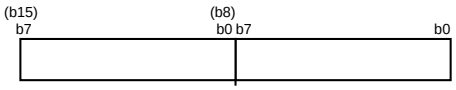
Symbol	Address	When reset
TAiMR(i=0 to 4)	0396₁₆ to 039A₁₆	00₁₆

b7	b6	b5	b4	b3	b2	b1	b0

Bit symbol	Bit name	Function	R	W
TMOD0	Operation mode select bit	b1 b0 0 0 : Timer mode 0 1 : Event counter mode 1 0 : One-shot timer mode 1 1 : Pulse width modulation (PWM) mode	○	○
TMOD1			○	○
MR0	Function varies with each operation mode		○	○
MR1			○	○
MR2			○	○
MR3			○	○
TCK0	Count source select bit (Function varies with each operation mode)		○	○
TCK1			○	○

Figure 2.2.2. Timer A-related registers (1)

Timer Ai register (Note 1)

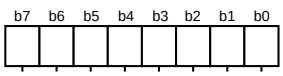


Symbol	Address	When reset
TA0	0387_{16},0386_{16}	Indeterminate
TA1	0389_{16},0388_{16}	Indeterminate
TA2	$038B_{16}$,$038A_{16}$	Indeterminate
TA3	$038D_{16}$,$038C_{16}$	Indeterminate
TA4	$038F_{16}$,$038E_{16}$	Indeterminate

Function	Values that can be set	R	W
• Timer mode Counts an internal count source	0000_{16} to $FFFF_{16}$	○	○
• Event counter mode Counts pulses from an external source or timer overflow	0000_{16} to $FFFF_{16}$	○	○
• One-shot timer mode Counts a one shot width	0000_{16} to $FFFF_{16}$ (Note 2,4)	×	○
• Pulse width modulation mode (16-bit PWM) Functions as a 16-bit pulse width modulator	0000_{16} to $FFFE_{16}$ (Note 3,4)	×	○
• Pulse width modulation mode (8-bit PWM) Timer low-order address functions as an 8-bit prescaler and high-order address functions as an 8-bit pulse width modulator	00_{16} to FE_{16} (High-order address) 00_{16} to FF_{16} (Low-order address) (Note 3,4)	×	○

Note 1: Read and write data in 16-bit units.
Note 2: When the timer Ai register is set to "0000_{16}", the counter does not operate and the timer Ai interrupt request is not generated. When the pulse is set to output, the pulse does not output from the TAi$_{OUT}$ pin.
Note 3: When the timer Ai register is set to "0000_{16}", the pulse width modulator does not operate and the output level of the TAi$_{OUT}$ pin remains "L" level, therefore the timer Ai interrupt request is not generated. This also occurs in the 8-bit pulse width modulator mode when the significant 8 high-order bits in the timer Ai register are set to "00_{16}".
Note 4: Use MOV instruction to write to this register.

Count start flag



Symbol	Address	When reset
TABSR	0380_{16}	00_{16}

Bit symbol	Bit name	Function	R	W
TA0S	Timer A0 count start flag	0 : Stops counting 1 : Starts counting	○	○
TA1S	Timer A1 count start flag		○	○
TA2S	Timer A2 count start flag		○	○
TA3S	Timer A3 count start flag		○	○
TA4S	Timer A4 count start flag		○	○
TB0S	Timer B0 count start flag		○	○
TB1S	Timer B1 count start flag		○	○
TB2S	Timer B2 count start flag		○	○

Figure 2.2.3. Timer A-related registers (2)

Up/down flag (Note 1)

b7	b6	b5	b4	b3	b2	b1	b0

Symbol	Address	When reset
UDF	0384_{16}	00_{16}

Bit symbol	Bit name	Function	R	W
TA0UD	Timer A0 up/down flag	0 : Down count 1 : Up count This specification becomes valid when the up/down flag content is selected for up/down switching cause	○	○
TA1UD	Timer A1 up/down flag		○	○
TA2UD	Timer A2 up/down flag		○	○
TA3UD	Timer A3 up/down flag		○	○
TA4UD	Timer A4 up/down flag		○	○
TA2P	Timer A2 two-phase pulse signal processing select bit	0 : two-phase pulse signal processing disabled 1 : two-phase pulse signal processing enabled (Note 2) When not using the two-phase pulse signal processing function, set the select bit to “0”	×	○
TA3P	Timer A3 two-phase pulse signal processing select bit		×	○
TA4P	Timer A4 two-phase pulse signal processing select bit		×	○

Note 1: Use MOV instruction to write to this register.
Note 2: Set the TAiIN and TAiOUT pins correspondent port direction registers to “0”.

One-shot start flag

b7	b6	b5	b4	b3	b2	b1	b0
		×					

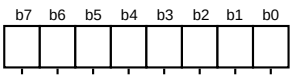
Symbol	Address	When reset
ONSF	0382_{16}	$00X00000_2$

Bit symbol	Bit name	Function	R	W
TA0OS	Timer A0 one-shot start flag	1 : Timer start When read, the value is “0”	○	○
TA1OS	Timer A1 one-shot start flag		○	○
TA2OS	Timer A2 one-shot start flag		○	○
TA3OS	Timer A3 one-shot start flag		○	○
TA4OS	Timer A4 one-shot start flag		○	○
Nothing is assigned. In an attempt to write to this bit, write “0”. The value, if read, turns out to be indeterminate.			—	—
TA0TGL	Timer A0 event/trigger select bit	b7 b6 0 0 : Input on TA0IN is selected (Note) 0 1 : TB2 overflow is selected 1 0 : TA4 overflow is selected 1 1 : TA1 overflow is selected	○	○
TA0TGH			○	○

Note: Set the corresponding port direction register to “0”.

Figure 2.2.4. Timer A-related registers (3)

Trigger select register

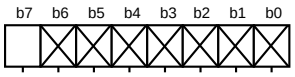


Symbol	Address	When reset
TRGSR	0383_{16}	00_{16}

Bit symbol	Bit name	Function	R	W
TA1TGL	Timer A1 event/trigger select bit	b1 b0 0 0 : Input on TA1IN is selected (Note) 0 1 : TB2 overflow is selected 1 0 : TA0 overflow is selected 1 1 : TA2 overflow is selected	○	○
TA1TGH			○	○
TA2TGL	Timer A2 event/trigger select bit	b3 b2 0 0 : Input on TA2IN is selected (Note) 0 1 : TB2 overflow is selected 1 0 : TA1 overflow is selected 1 1 : TA3 overflow is selected	○	○
TA2TGH			○	○
TA3TGL	Timer A3 event/trigger select bit	b5 b4 0 0 : Input on TA3IN is selected (Note) 0 1 : TB2 overflow is selected 1 0 : TA2 overflow is selected 1 1 : TA4 overflow is selected	○	○
TA3TGH			○	○
TA4TGL	Timer A4 event/trigger select bit	b7 b6 0 0 : Input on TA4IN is selected (Note) 0 1 : TB2 overflow is selected 1 0 : TA3 overflow is selected 1 1 : TA0 overflow is selected	○	○
TA4TGH			○	○

Note: Set the corresponding port direction register to “0”.

Clock prescaler reset flag



Symbol	Address	When reset
CPSRF	0381_{16}	$0XXXXXXX_2$

Bit symbol	Bit name	Function	R	W
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be indeterminate.			—	—
CPSR	Clock prescaler reset flag	0 : No effect 1 : Prescaler is reset (When read, the value is “0”)	○	○

Figure 2.2.5. Timer A-related registers (4)

2.2.2 Operation of Timer A (timer mode)

In timer mode, choose functions from those listed in Table 2.2.1. Operations of the circled items are described below. Figure 2.2.6 shows the operation timing, and Figure 2.2.7 shows the set-up procedure.

Table 2.2.1. Choosed functions

Item	Set-up	
Count source	**O**	Internal count source (f_1 / f_8 / f_{32} / fc_{32})
Pulse output function	**O**	No pulses output
		Pulses output
Gate function	**O**	No gate function
		Performs count only for the period in which the TAi_{IN} pin is at "L" level
		Performs count only for the period in which the TAi_{IN} pin is at "H" level

Operation (1) Setting the count start flag to "1" causes the counter to perform a down count on the count source.
(2) If an underflow occurs, the content of the reload register is reloaded, and the count continues. At this time, the timer Ai interrupt request bit goes to "1".
(3) Setting the count start flag to "0" causes the counter to hold its value and to stop.

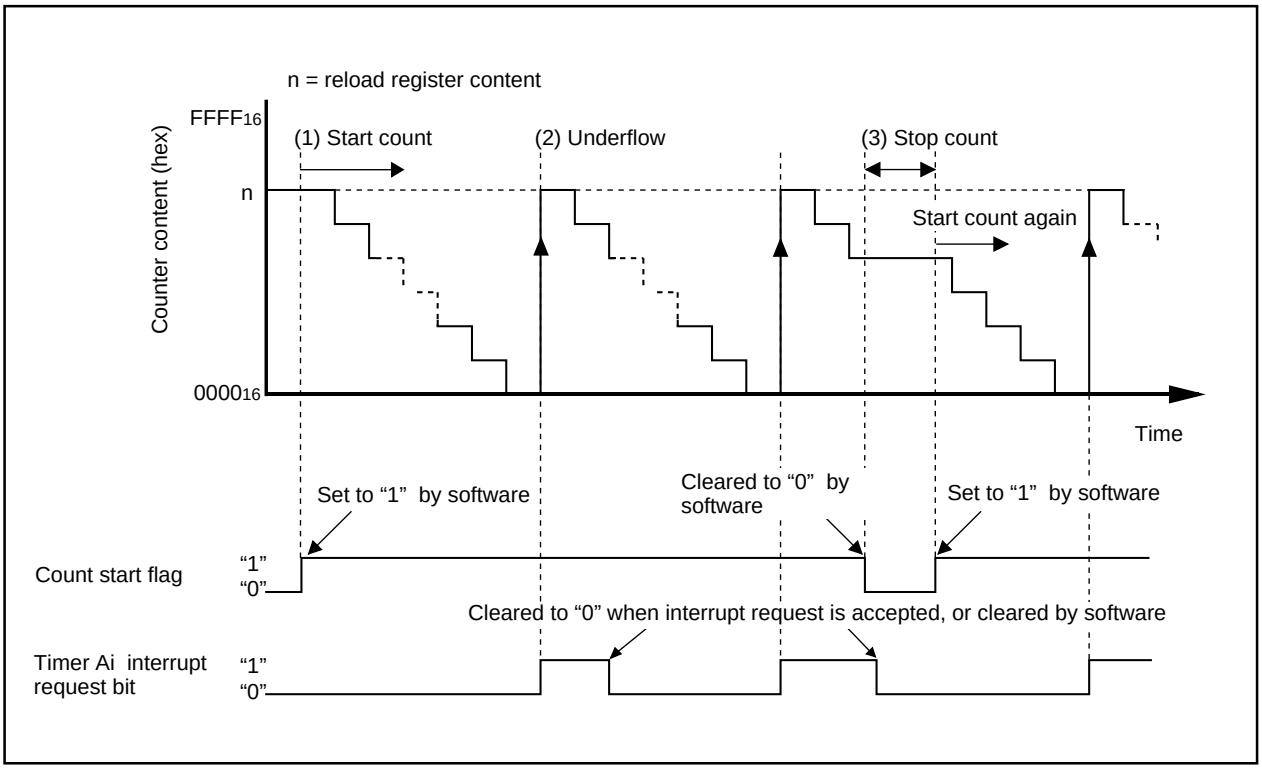


Figure 2.2.6. Operation timing of timer mode

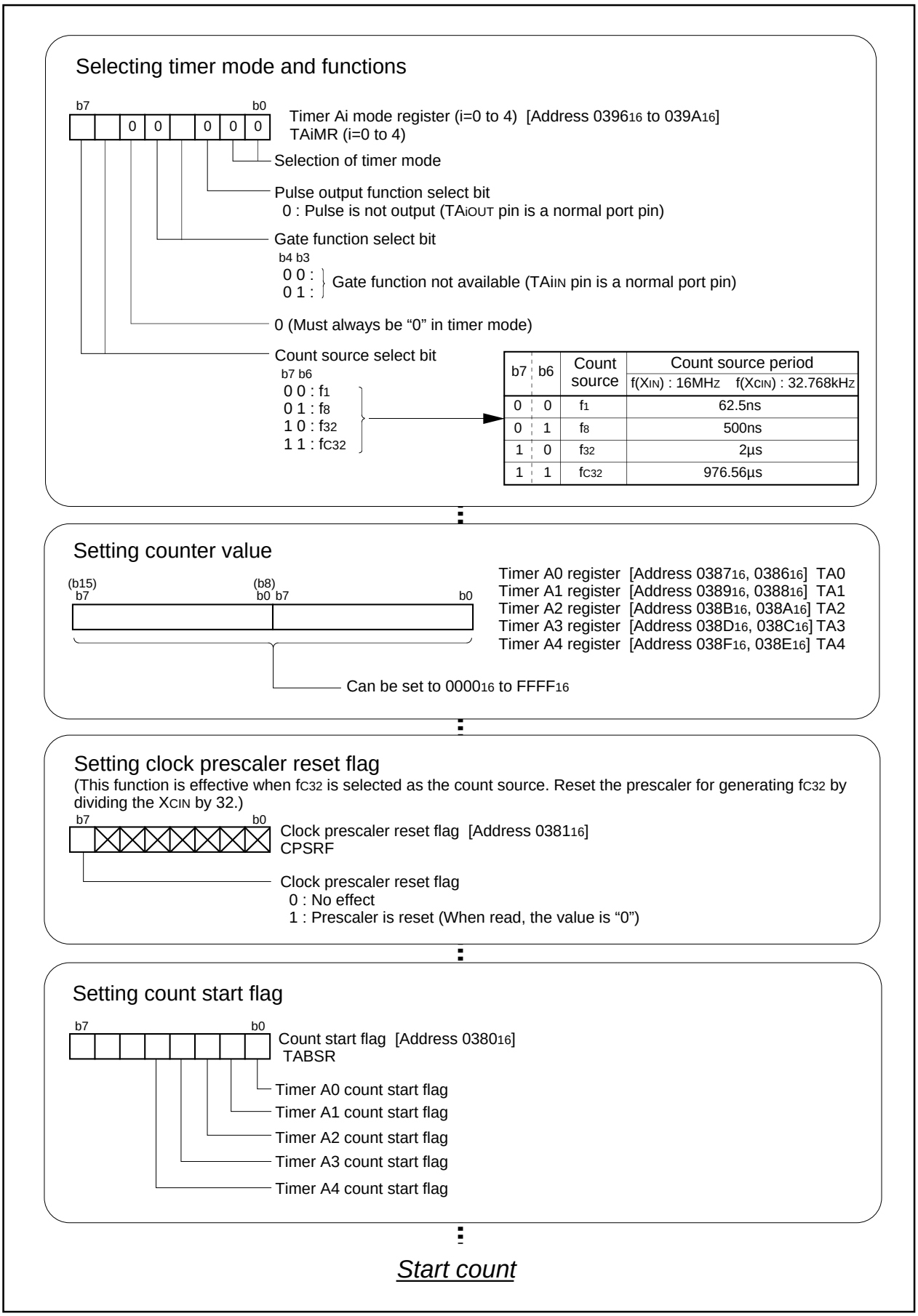


Figure 2.2.7. Set-up procedure of timer mode

2.2.3 Operation of Timer A (timer mode, gate function selected)

In timer mode, choose functions from those listed in Table 2.2.2. Operations of the circled items are described below. Figure 2.2.8 shows the operation timing, and Figure 2.2.9 shows the set-up procedure.

Table 2.2.2. Choosed functions

Item	Set-up	
Count source	**O**	Internal count source(f_1 / f_8 / f_{32} / fc_{32})
Pulse output function	**O**	No pulses output
		Pulses output
Gate function		No gate function
		Performs count only for the period in which the TAiIN pin is at "L" level
	O	Performs count only for the period in which the TAiIN pin is at "H" level

Operation (1) When the count start flag is set to "1" and the TAiIN pin inputs at "H" level, the counter performs a down count on the count source.
(2) When the TAiIN pin inputs at "L" level, the counter holds its value and stops.
(3) If an underflow occurs, the content of the reload register is reloaded and the count continues. At this time, the timer Ai interrupt request bit goes to "1".
(4) Setting the count start flag to "0" causes the counter to hold its value and to stop.

Note • Make the pulse width of the signal input to the TAiIN pin not less than two cycles of the count source.

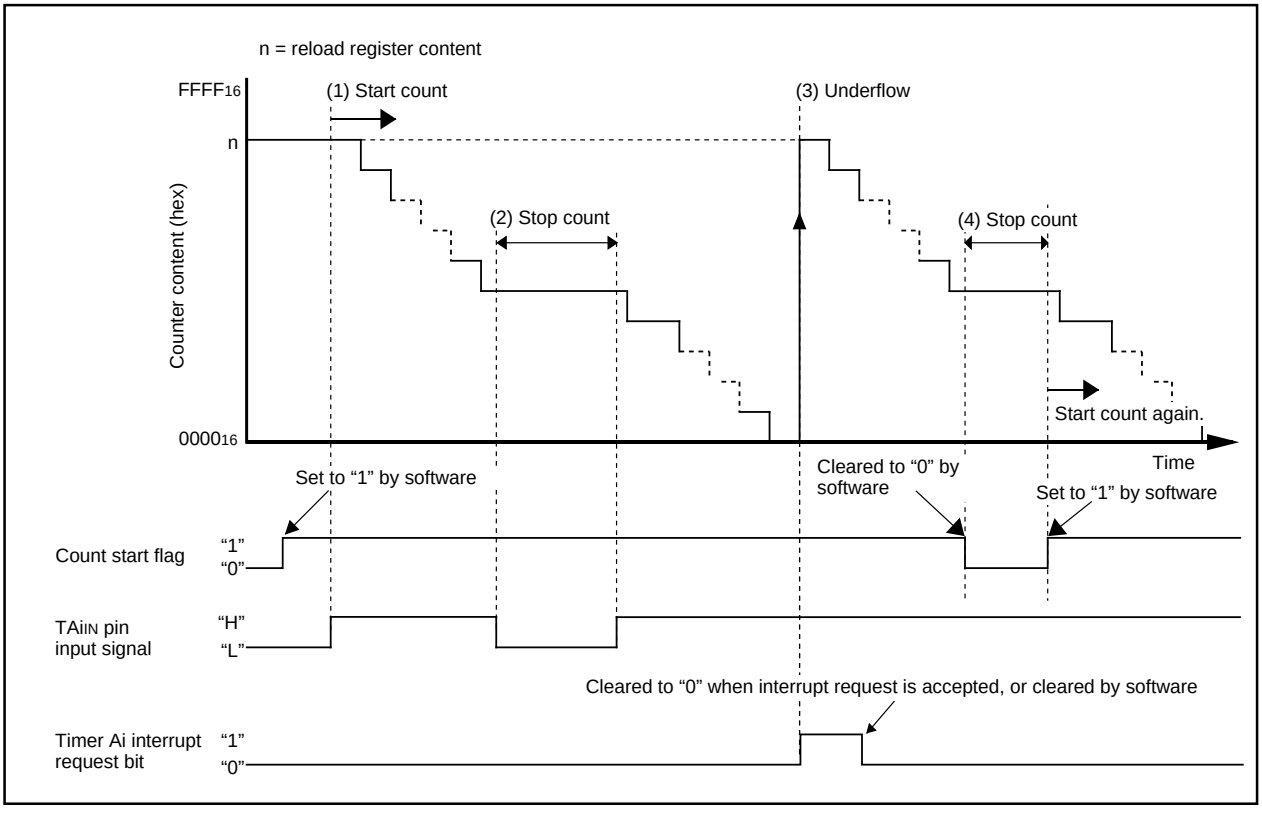


Figure 2.2.8. Operation timing of timer mode, gate function selected

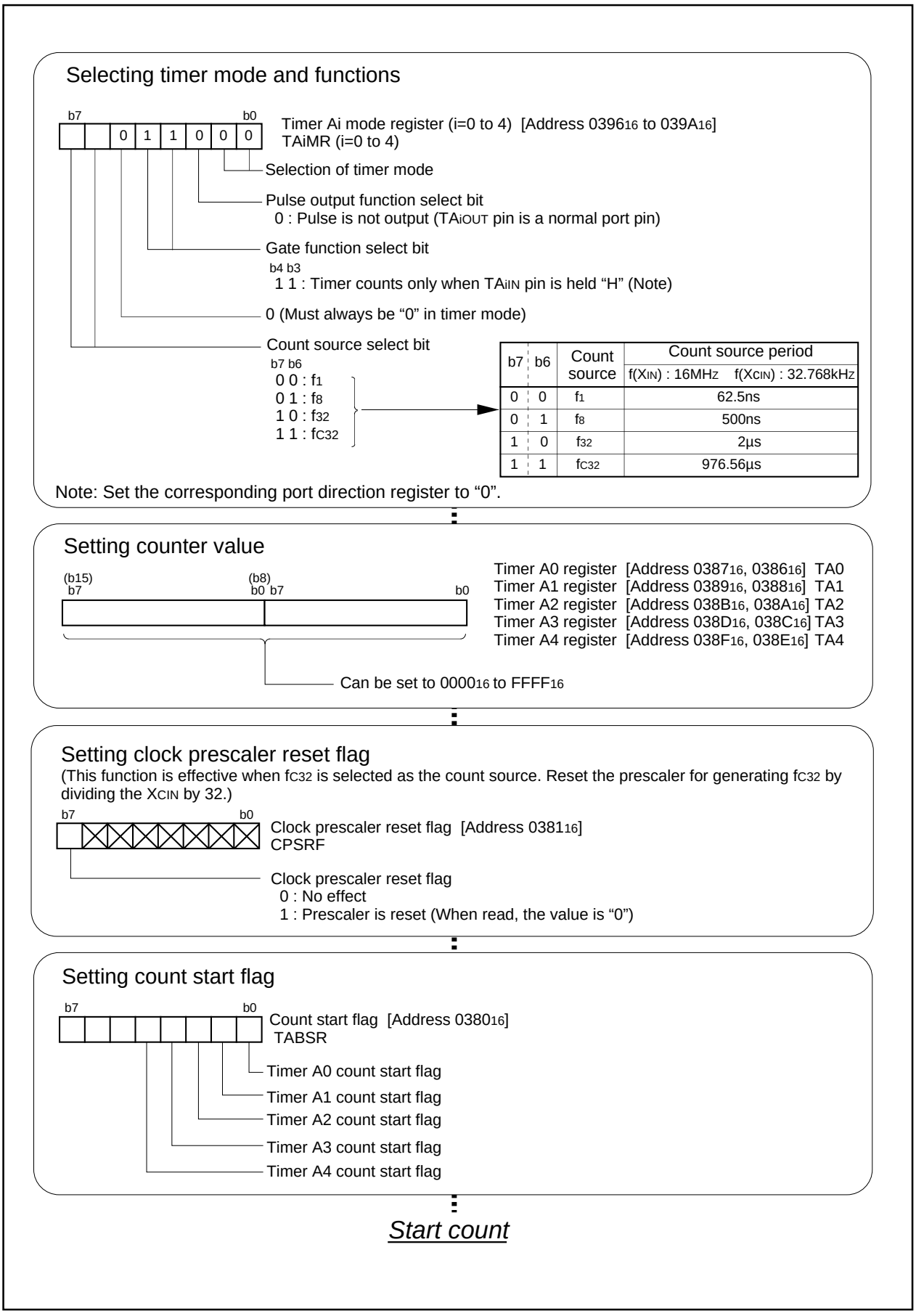


Figure 2.2.9. Set-up procedure of timer mode, gate function selected

2.2.4 Operation of Timer A (timer mode, pulse output function selected)

In timer mode, choose functions from those listed in Table 2.2.3. Operations of the circled items are described below. Figure 2.2.10 shows the operation timing, and Figure 2.2.11 shows the set-up procedure.

Table 2.2.3. Choosed functions

Item	Set-up	
Count source	O	Internal count source(f_1 / f_8 / f_{32} / fc_{32})
Pulse output function		No pulses output
	O	Pulses output
Gate function	O	No gate function
		Performs count only for the period in which the TAi_{IN} pin is at "L" level
		Performs count only for the period in which the TAi_{IN} pin is at "H" level

Operation (1) Setting the count start flag to "1" causes the counter to perform a down count on the count source.

(2) If an underflow occurs, the content of the reload register is reloaded and the count continues. At this time, the timer Ai interrupt request bit goes to "1". Also, the output polarity of the TAi_{OUT} pin reverses.

(3) Setting the count start flag to "0" causes the counter to hold its value and to stop. Also, the TAi_{OUT} pin outputs an "L" level.

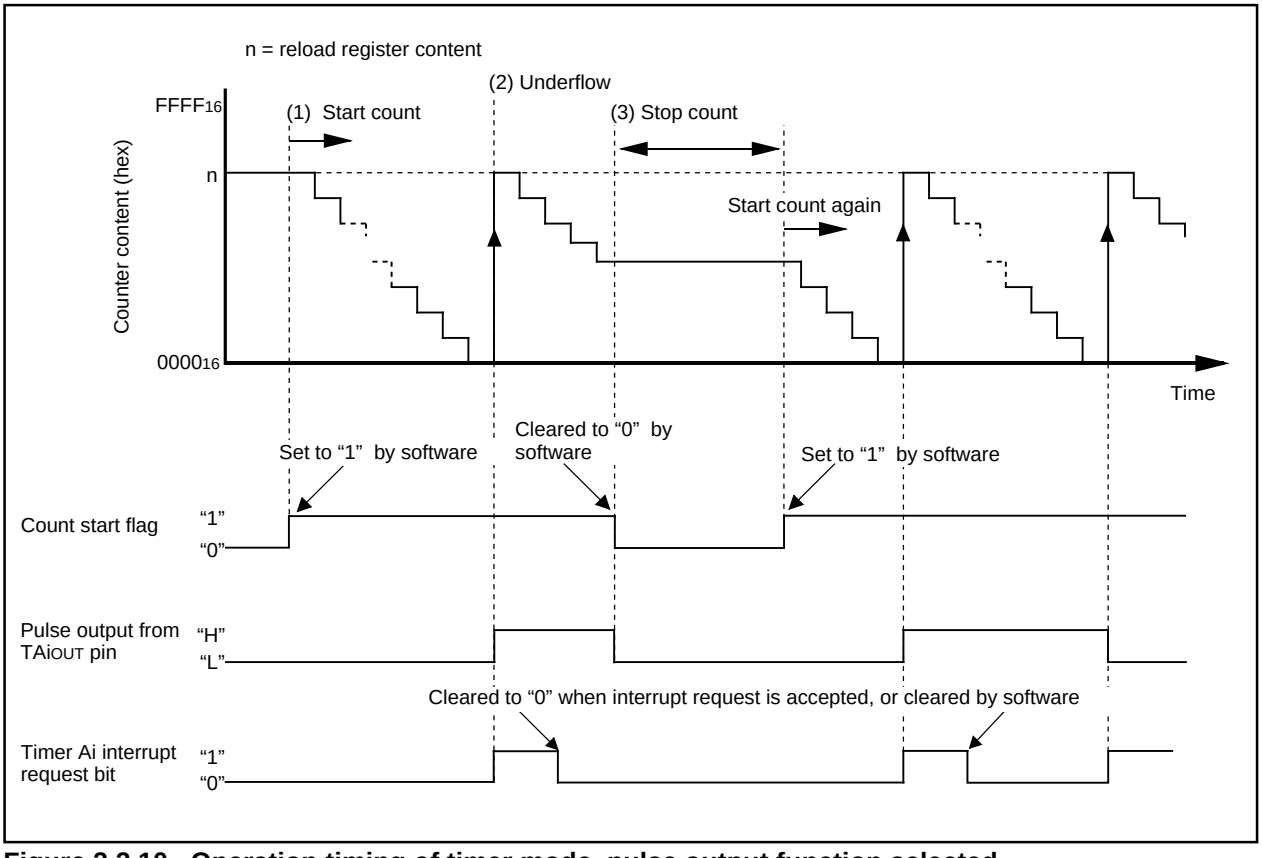


Figure 2.2.10. Operation timing of timer mode, pulse output function selected

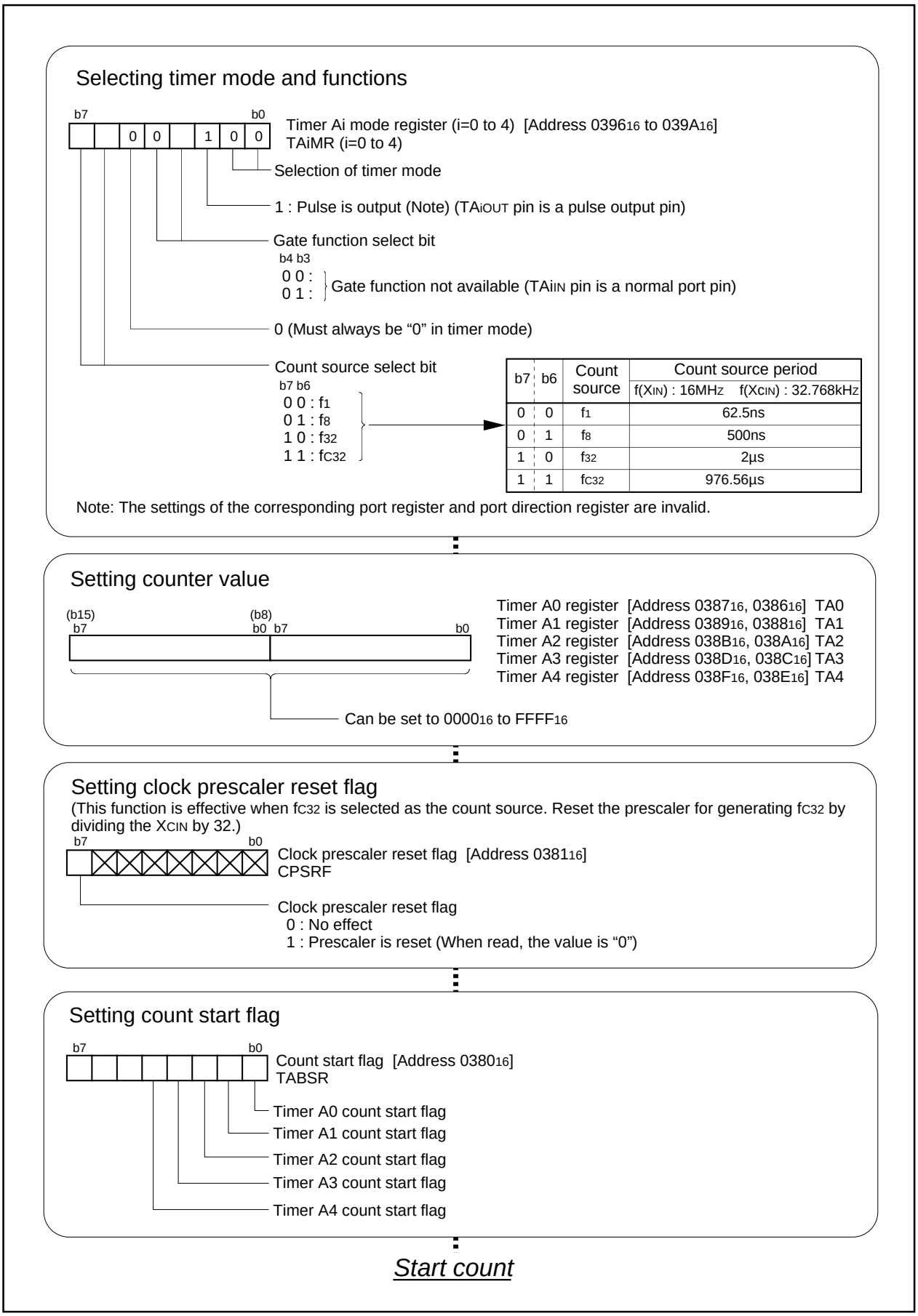


Figure 2.2.11. Set-up procedure of timer mode, pulse output function selected

2.2.5 Operation of Timer A (event counter mode, reload type selected)

In event counter mode, choose functions from those listed in Table 2.2.4. Operations of the circled items are described below. Figure 2.2.12 shows the operation timing, and Figure 2.2.13 shows the set-up procedure.

Table 2.2.4. Choosed functions

Item	Set-up		Item	Set-up	
Count source	O	Input signal to TAiIN (counting falling edges)	Pulse output function	O	No pulses output
					Pulses output
		Input signal to TAiIN (counting rising edges)	Count operation type	O	Reload type
					Free-run type
		Timer overflow (TB2/TAj overflow)	Factor for switching between up and down	O	Content of up/down flag
					Input signal to TAiOUT

Note: j = i – 1, but j = 4 when i = 0.

Operation
(1) Setting the count start flag to “1” causes the counter to count the falling edges of the count source.
(2) If an underflow occurs, the content of the reload register is reloaded, and the count continues. At this time, the timer Ai interrupt request bit goes to “1”.
(3) If switching from an up count to a down count or vice versa while a count is in progress, the switch takes effect from the next effective edge of the count source.
(4) Setting the count start flag to “0” causes the counter to hold its value and to stop.
(5) If an overflow occurs, the content of the reload register is reloaded, and the count continues. At this time, the timer Ai interrupt request bit goes to “1”.

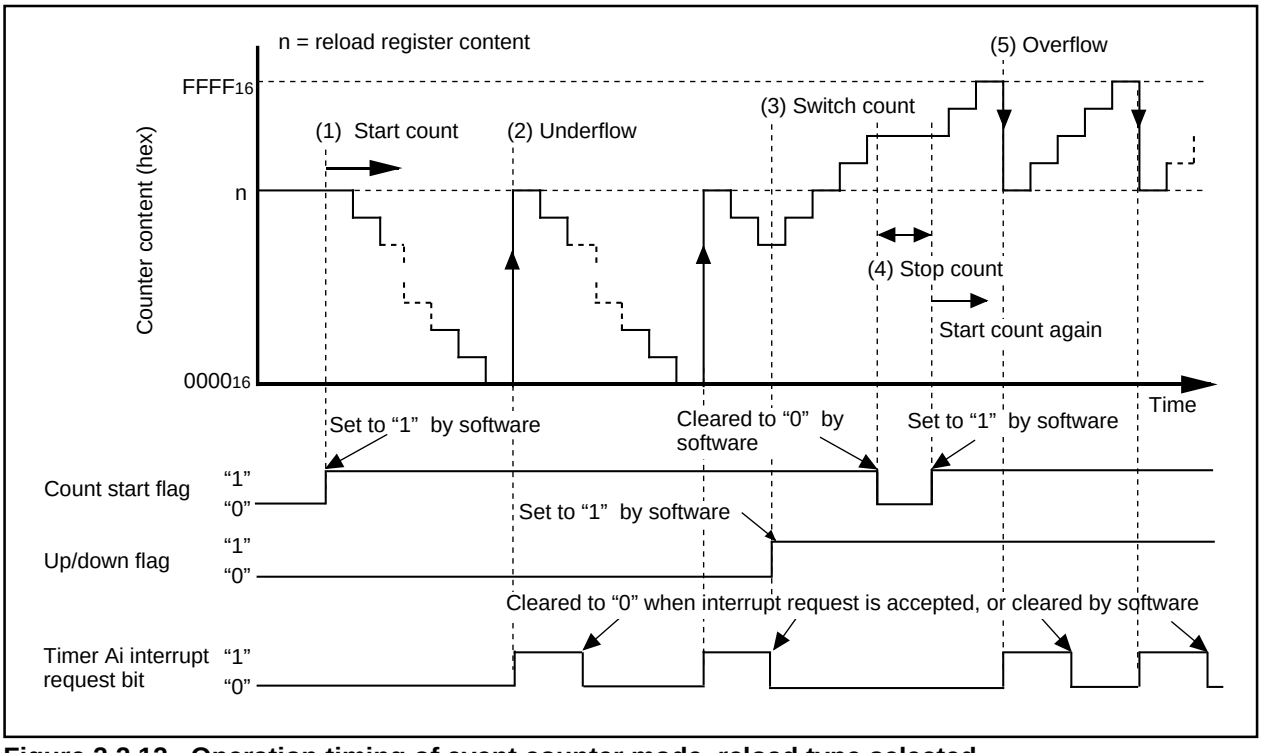


Figure 2.2.12. Operation timing of event counter mode, reload type selected

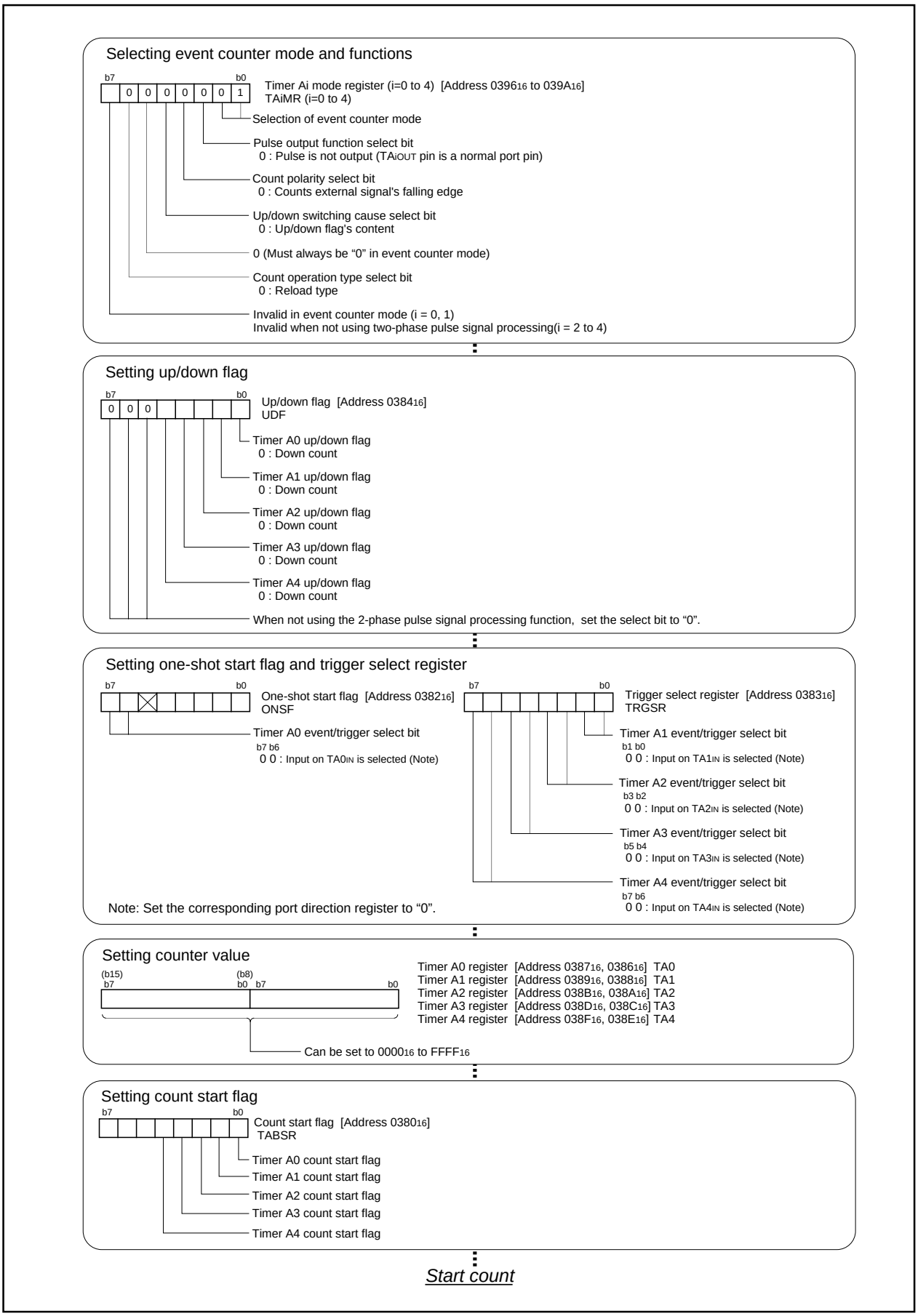


Figure 2.2.13. Set-up procedure of event counter mode, reload type selected

2.2.6 Operation of Timer A (event counter mode, free run type selected)

In event counter mode, choose functions from those listed in Table 2.2.5. Operations of the circled items are described below. Figure 2.2.14 shows the operation timing, and Figure 2.2.15 shows the set-up procedure.

Table 2.2.5. Choosed functions

Item	Set-up		Item	Set-up	
Count source	O	Input signal to TAiIN (counting falling edges)	Pulse output function	O	No pulses output
					Pulses output
		Input signal to TAiIN (counting rising edges)	Count operation type		Reload type
				O	Free-run type
		Timer overflow (TB2/TAj overflow)	Factor for switching between up and down	O	Content of up/down flag
					Input signal to TAiOUT

Note: j = i – 1, but j = 4 when i = 0

Operation (1) Setting the count start flag to “1” causes the counter to count the falling edges of the count source.

(2) Even if an underflow occurs, the content of the reload register is not reloaded, but the count continues. At this time, the timer Ai interrupt request bit goes to “1”.

(3) If switching from an up count to a down count or vice versa while a count is in progress, the switch takes effect from the next effective edge of the count source.

(4) Even if an overflow occurs, the content of the reload register is not reloaded, but the count continues. At this time, the timer Ai interrupt request bit goes to “1”.

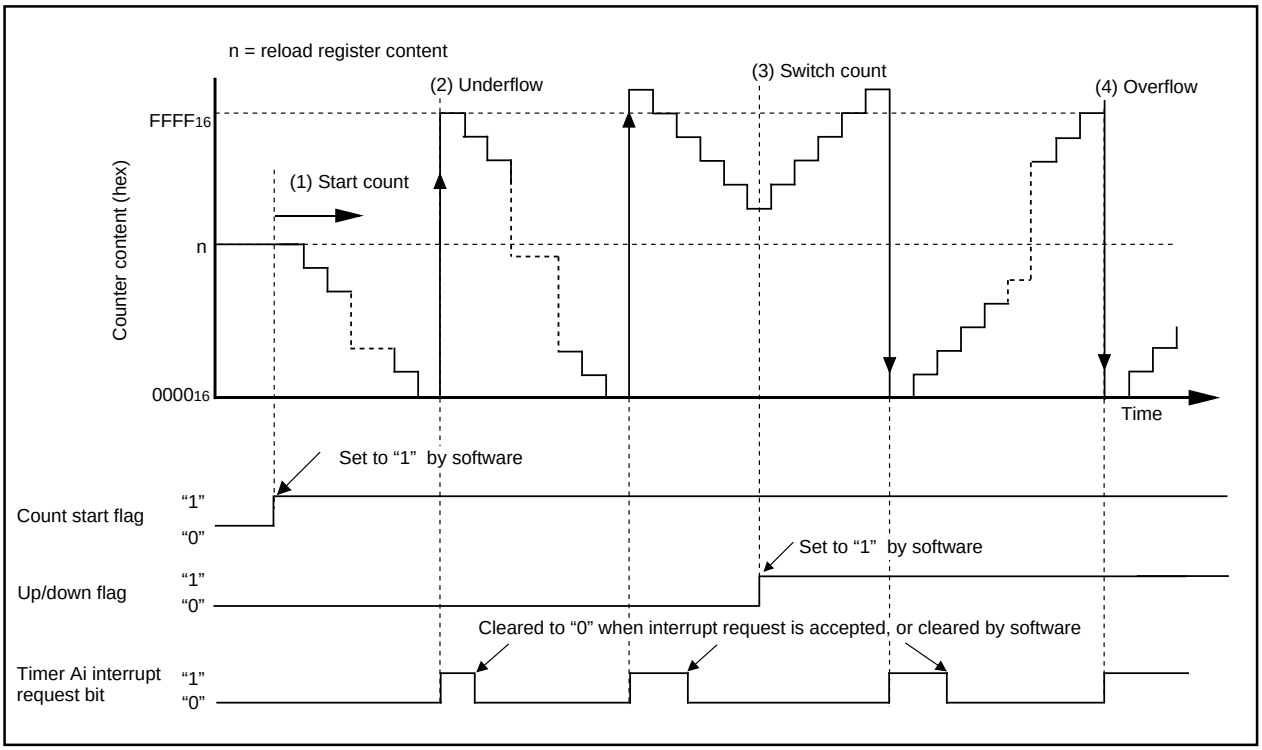


Figure 2.2.14. Operation timing of event counter mode, free run type selected

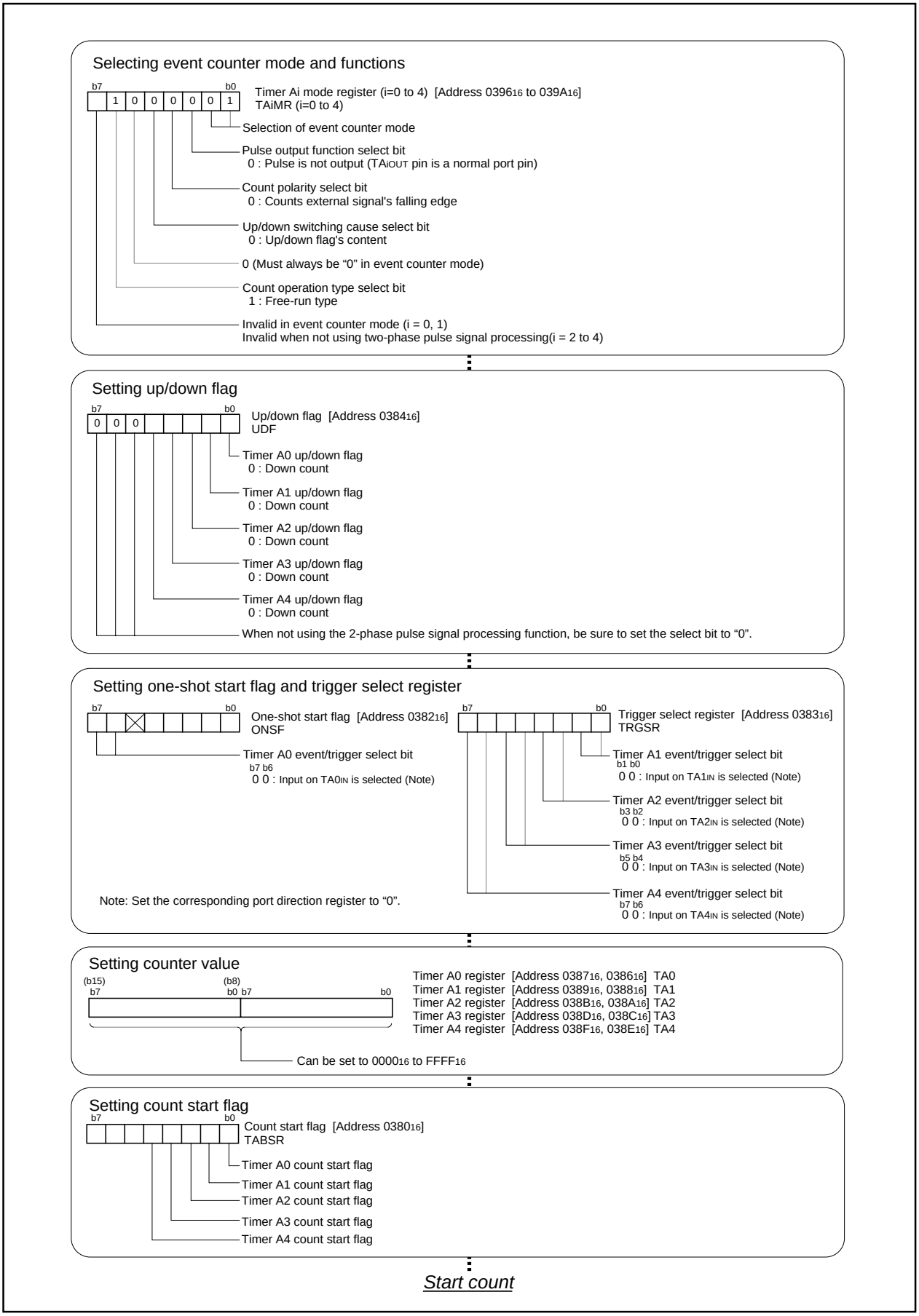


Figure 2.2.15. Set-up procedure of event counter mode, free run type selected

2.2.7 Operation of timer A (2-phase pulse signal process in event counter mode, normal mode selected)

In processing 2-phase pulse signals in event counter mode, choose functions from those listed in Table 2.2.6. Operations of the circled items are described below. Figure 2.2.16 shows the operation timing, and Figure 2.2.17 shows the set-up procedure.

Table 2.2.6. Choosed functions

Item	Set-up	
Count operation type		Reload type
	O	Free run type
2-phase pulses process (Note)	O	Normal processing
		4-multiplication processing

Note: Timer A3 alone can be selected. Timer A2 is solely used for normal processes, and timer A4 is solely used for 4 multiplication processes.

Operation (1) Setting the count start flag to “1” causes the counter to count effective edges of the count source.
(2) Even if an underflow occurs, the content of the reload register is not reloaded, but the count continues. At this time, the timer Ai interrupt request bit goes to “1”.
(3) Even if an overflow occurs, the content of the reload register is not reloaded, but the count continues. At this time, the timer Ai interrupt request bit goes to “1”.

Note • The up count or down count conditions are as follows:
If a rising edge is present at the TAiIN pin when the input signal level to the TAiOUT pin is “H”, an up count is performed.
If a falling edge is present at the TAiIN pin when the input signal level to the TAiOUT pin is “H”, a down count is performed.

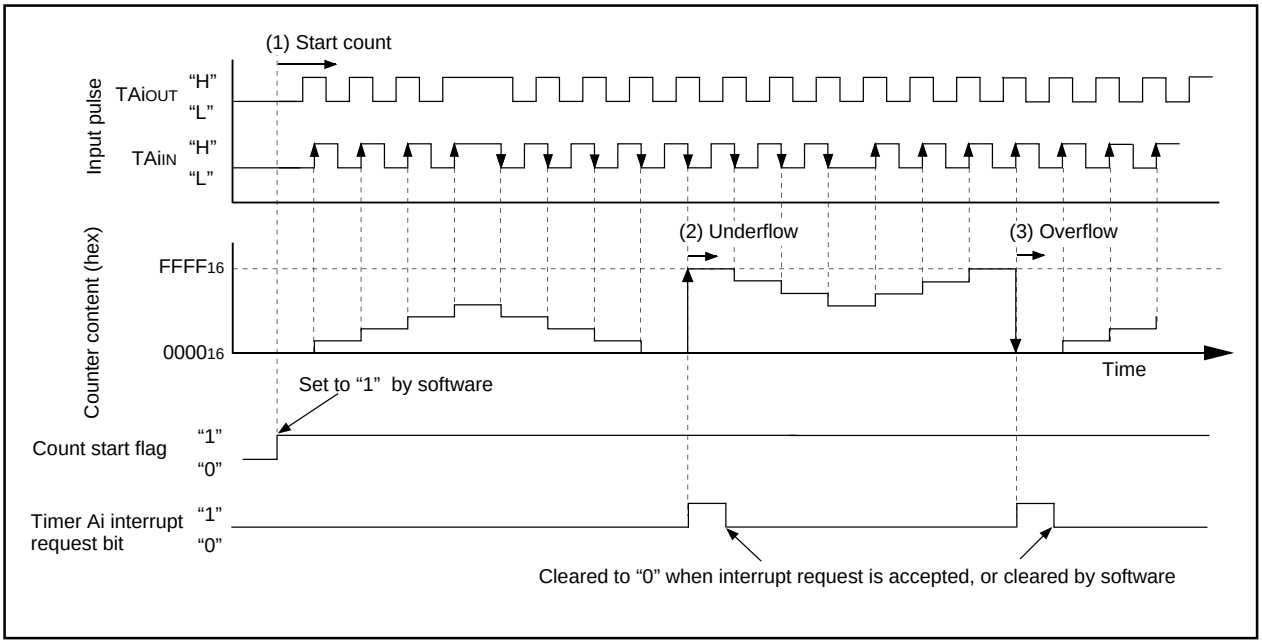


Figure 2.2.16. Operation timing of 2-phase pulse signal process in event counter mode, normal mode selected

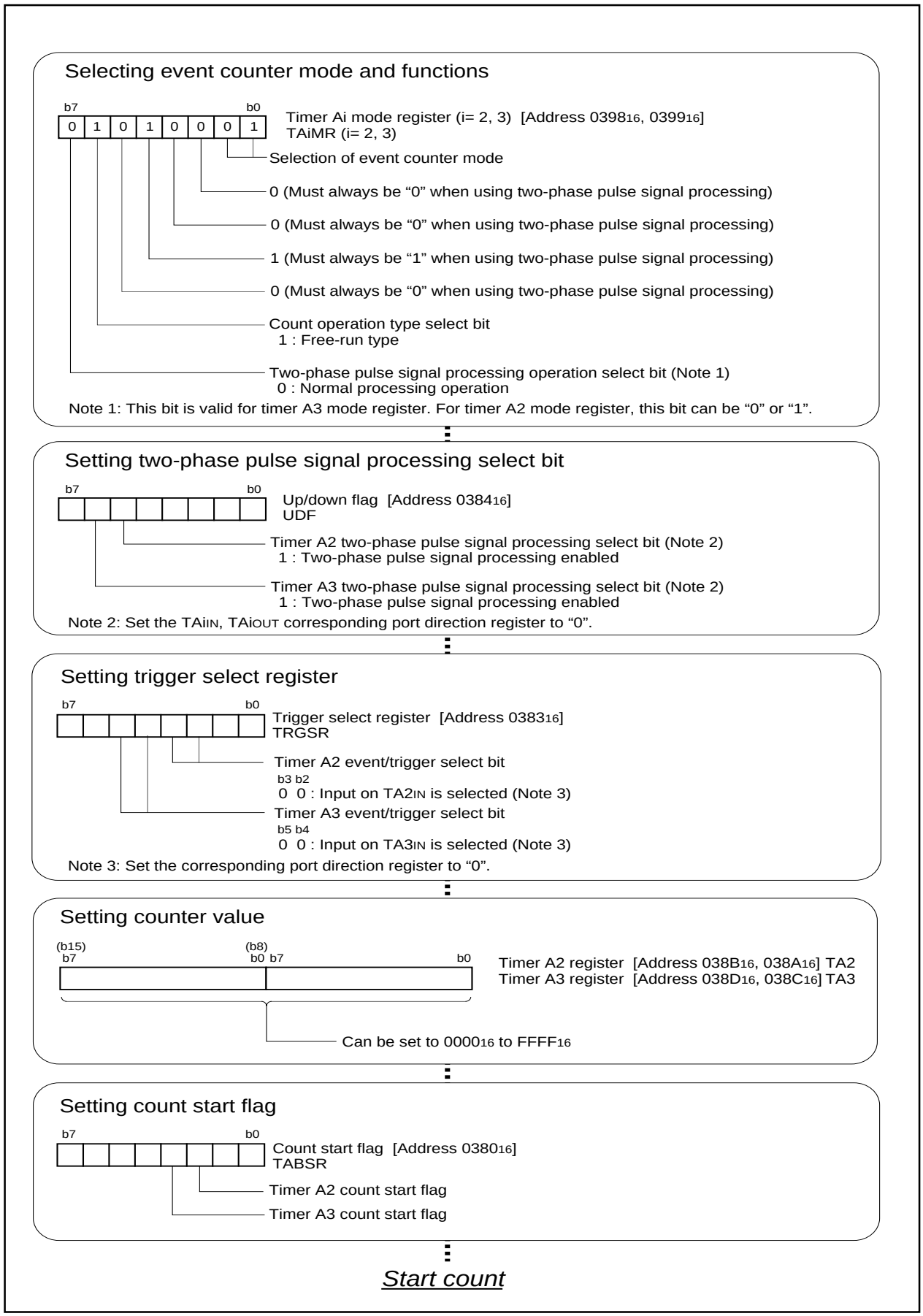


Figure 2.2.17. Set-up procedure of 2-phase pulse signal process in event counter mode, normal mode selected

2.2.8 Operation of timer A (2-phase pulse signal process in event counter mode, multiply-by-4 mode selected)

In processing 2-phase pulse signals in event counter mode, choose functions from those listed in Table 2.2.7. Operations of the circled items are described below. Figure 2.2.18 shows the operation timing, and Figure 2.2.19 shows the set-up procedure.

Table 2.2.7. Choosed functions

Item	Set-up		Item	Set-up	
Count operation type		Reload type	Processing 2 phase pulses (Note)		Normal processing
	O	Free run type		O	4-multiplication processing

Note: Timer A3 alone can be selected. Timer A2 is solely used for normal processes, and timer A4 is solely used for 4-multiplication processes.

Operation
(1) Setting the count start flag to “1” causes the counter to count effective edges of the count source.
(2) Even if an underflow occurs, the content of the reload register is not reloaded, but the count continues. At this time, the interrupt request bit goes to “1”.
(3) Even if an overflow occurs, the content of the reload register is not reloaded, but the count continues. At this time, the interrupt request bit goes to “1”.

Note
• The up count or down count conditions are as follows:

Table 2.2.8. The up count or down count conditions

	Input signal to the TAi_{OUT} pin	Input signal to the TAi_{IN} pin		Input signal to the TAi_{OUT} pin	Input signal to the TAi_{IN} pin
Up count	“H” level	Rising	Down count	“H” level	Falling
	“L” level	Falling		“L” level	Rising
	Rising	“L” level		Rising	“H” level
	Falling	“H” level		Falling	“L” level

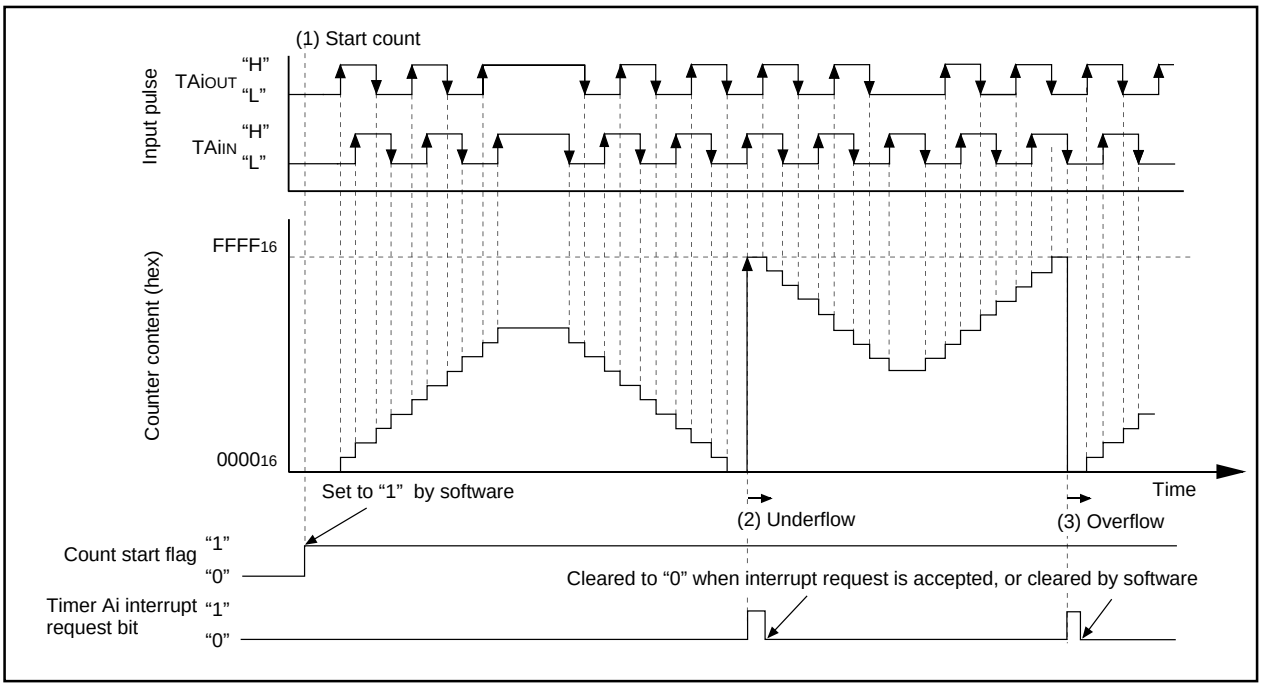


Figure 2.2.18. Operation timing of 2-phase pulse signal process in event counter mode, multiply-by-4 mode selected

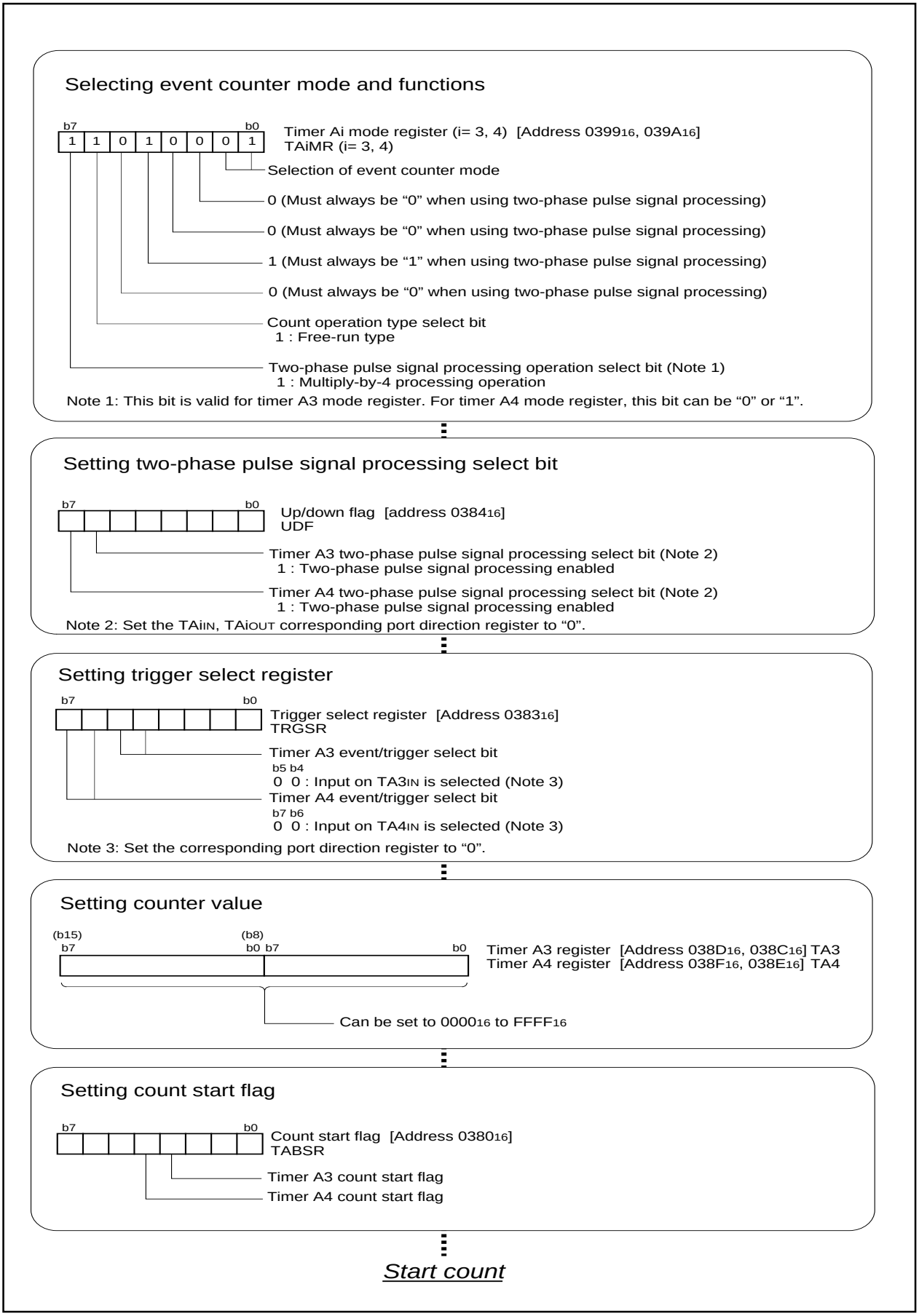


Figure 2.2.19. Set-up procedure of 2-phase pulse signal process in event counter mode, multiply-by-4 mode selected

2.2.9 Operation of Timer A (one-shot timer mode)

In one-shot timer mode, choose functions from those listed in Table 2.2.9. Operations of the circled items are described below. Figure 2.2.20 shows the operation timing, and Figure 2.2.21 shows the set-up procedure.

Table 2.2.9. Choosed functions

Item	Set-up	
Count source	**O**	Internal count source (f_1 / f_8 / f_{32} / fc_{32})
Pulse output function		No pulses output
	O	Pulses output
Count start condition		External trigger input (falling edge of input signal to the TAi$_{IN}$ pin)
		External trigger input (rising edge of input signal to the TAi$_{IN}$ pin)
		Timer overflow (TB2/TAj/TAk overflow)
	O	Writing “1” to the one-shot start flag

Note: j = i – 1, but j = 4 when i = 0; k = i + 1, but k = 0 when i = 4.

Operation
(1) Setting the one-shot start flag to “1” with the count start flag set to “1” causes the counter to perform a down count on the count source. At this time, the TAi$_{OUT}$ pin outputs an “H” level.
(2) The instant the value of the counter becomes “0000$_{16}$”, the TAi$_{OUT}$ pin outputs an “L” level, and the counter reloads the content of the reload register and stops counting. At this time, the timer Ai interrupt request bit goes to “1”.
(3) If a trigger occurs while a count is in progress, the counter reloads the value in the reload register again and continues counting. The reload timing is in step with the next count source input after the trigger.
(4) Setting the count start flag to “0” causes the counter to stop and to reload the content of the reload register. Also, the TAi$_{OUT}$ pin outputs an “L” level. At this time, the timer Ai interrupt request bit goes to “1”.

Note
• When the timer Ai register is set to “0000$_{16}$”, the counter does not operate and the timer Ai interrupt request is not generated. When the pulse is set to output, the pulse does not output from the TAi$_{OUT}$ pin.

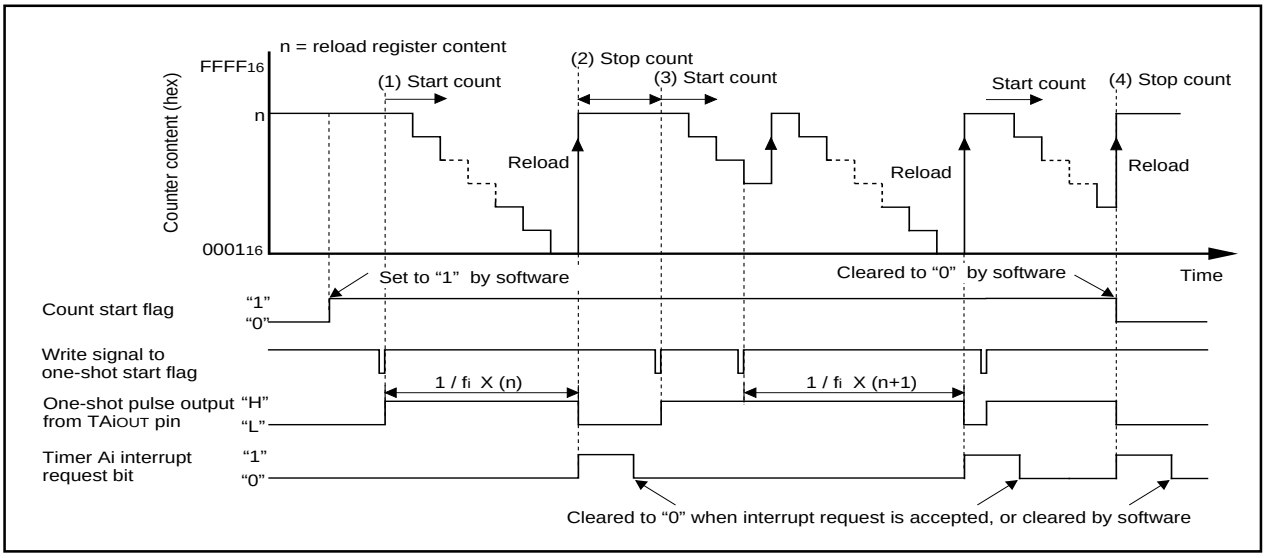


Figure 2.2.20. Operation timing of one-shot mode

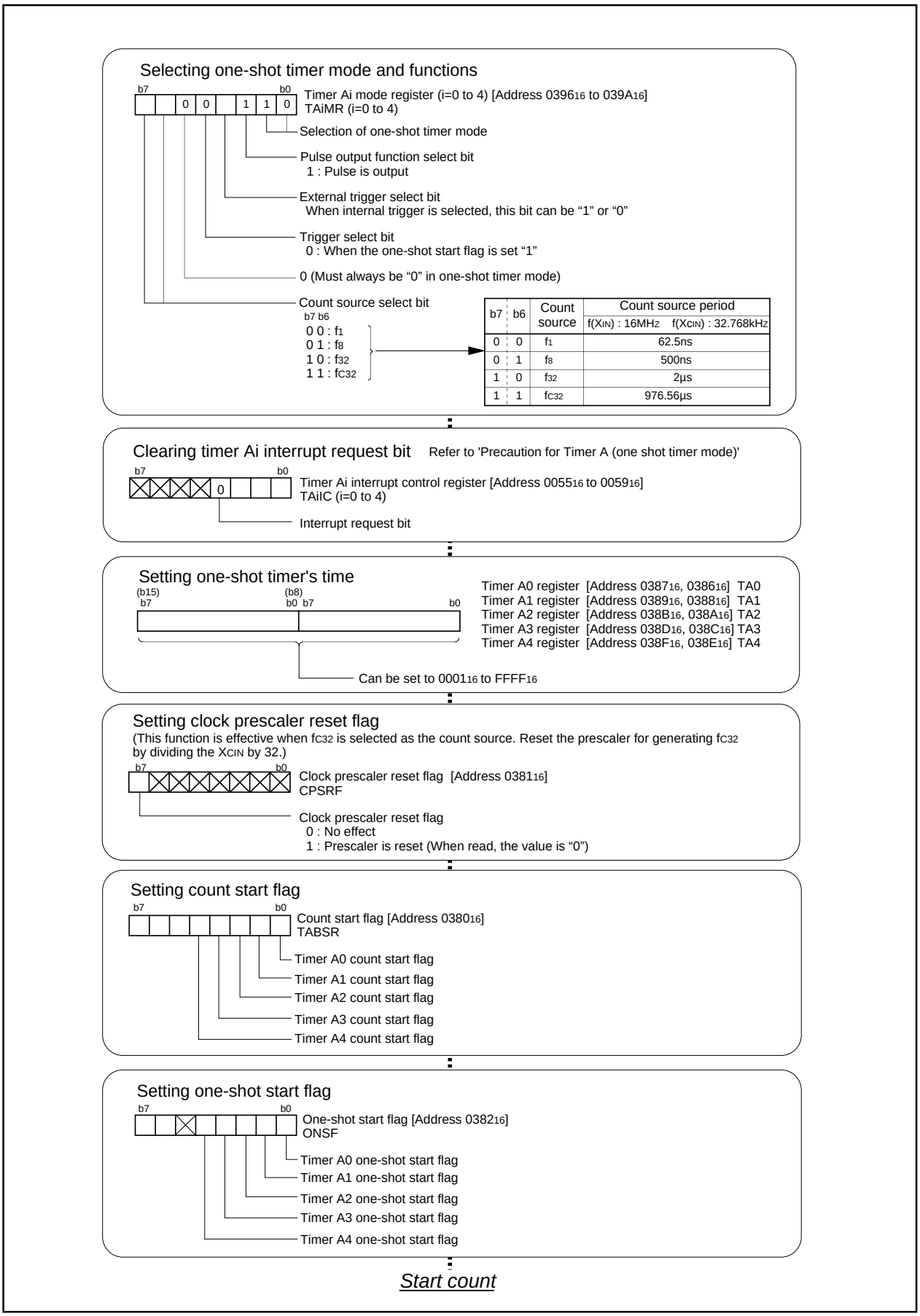


Figure 2.2.21. Set-up procedure of one-shot mode

2.2.10 Operation of Timer A (one-shot timer mode, external trigger selected)

In one-shot timer mode, choose functions from those listed in Table 2.2.10. Operations of the circled items are described below. Figure 2.2.22 shows the operation timing, and Figure 2.2.23 shows the set-up procedure.

Table 2.2.10. Choosed functions

Item	Set-up	
Count source	**O**	Internal count source (f_1 / f_8 / f_{32} / fc_{32})
Pulse output function		No pulses output
	O	Pulses output
Count start condition		External trigger input (falling edge of input signal to the TAiIN pin)
	O	External trigger input (rising edge of input signal to the TAiIN pin)
		Timer overflow (TB2/TAj/TAk overflow)
		Writing “1” to the one-shot start flag

Note: j = i – 1, but j = 4 when i = 0; k = i + 1, but k = 0 when i = 4.

Operation (1) If the TAiIN pin input level changes from “L” to “H” with the count start flag set to “1”, the counter performs a down count on the count source. At this time, the TAiOUT pin output level goes to “H” level.

(2) If the value of the counter becomes “0000_{16}”, the TAiOUT pin outputs an “L” level, and the counter reloads the content of the reload register and stops counting. At this time, the timer Ai interrupt request bit goes to “1”.

(3) If a trigger occurs while a count is in progress, the counter reloads the value of the reload register again and continues counting. The reload timing is in step with the next count source input after the trigger.

(4) Setting the count start flag to “0” causes the counter to stop and to reload the content of the reload register. Also, the TAiOUT pin outputs an “L” level. At this time, the timer Ai interrupt request bit goes to “1”.

Note • When the timer Ai register is set to “0000_{16}”, the counter does not operate and the timer Ai interrupt request is not generated. When the pulse is set to output, the pulse does not output from the TAiOUT pin.

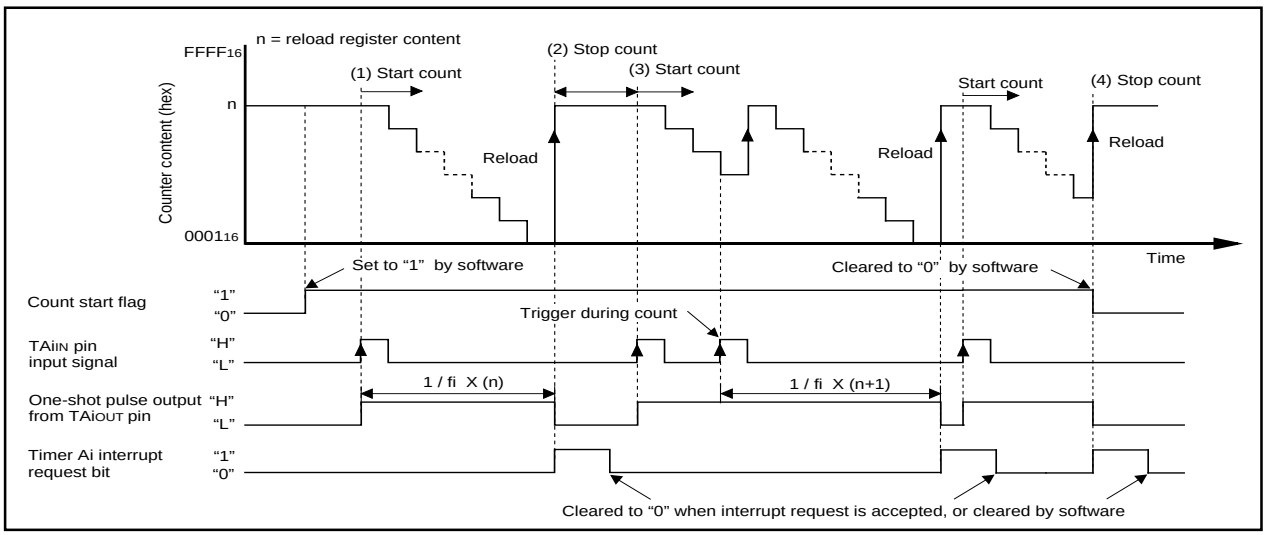


Figure 2.2.22. Operation timing of one-shot mode, external trigger selected

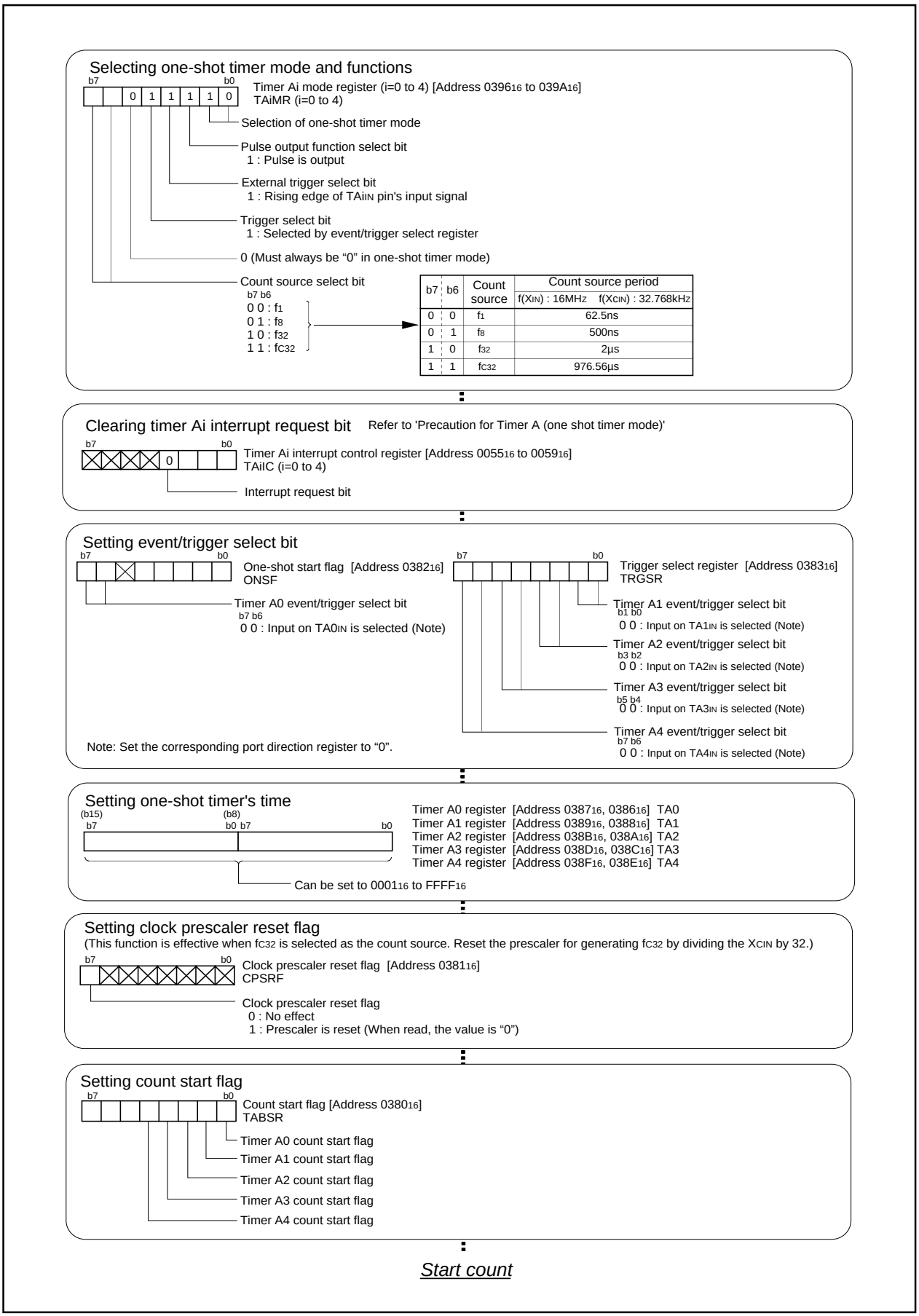


Figure 2.2.23. Set-up procedure of one-shot mode, external trigger selected

2.2.11 Operation of Timer A (pulse width modulation mode, 16-bit PWM mode selected)

In pulse width modulation mode, choose functions from those listed in Table 2.2.11. Operations of the circled items are described below. Figure 2.2.24 shows the operation timing, and Figure 2.2.25 shows the set-up procedure.

Table 2.2.11. Choosed functions

Item	Set-up	
Count source	**O**	Internal count source (f_1 / f_8 / f_{32} / fc_{32})
PWM mode	**O**	16-bit PWM
		8-bit PWM
Count start condition		External trigger input (falling edge of input signal to the TAi_{IN} pin)
	O	External trigger input (rising edge of input signal to the TAi_{IN} pin)
		Timer overflow (TB2/TAj/TAk overflow)

Note: j = i – 1, but j = 4 when i = 0; k = i + 1, but k = 0 when i = 4.

Operation
(1) If the TAi_{IN} pin input level changes from "L" to "H" with the count start flag set to "1", the counter performs a down count on the count source. Also, the TAi_{OUT} pin outputs an "H" level.
(2) The TAi_{OUT} pin output level changes from "H" to "L" when a set time period elapses. At this time, the timer Ai interrupt request bit goes to "1".
(3) The counter reloads the content of the reload register every time PWM pulses are output for one cycle, and continues counting.
(4) Setting the count start flag to "0" causes the counter to hold its value and to stop. Also, the TAi_{OUT} outputs an "L" level.

Note
• The period of PWM pulses becomes $(2^{16} – 1)/fi$, and the "H" level pulse width becomes n/fi. If the timer Ai register is set to "0000_{16}", the pulse width modulator does not work, and the TAi_{OUT} pin outputs "L" level, therefore the timer Ai interrupt request is not generated.
(fi : frequency of the count source f_1, f_8, f_{32}, fC_{32}; n : value of the timer)

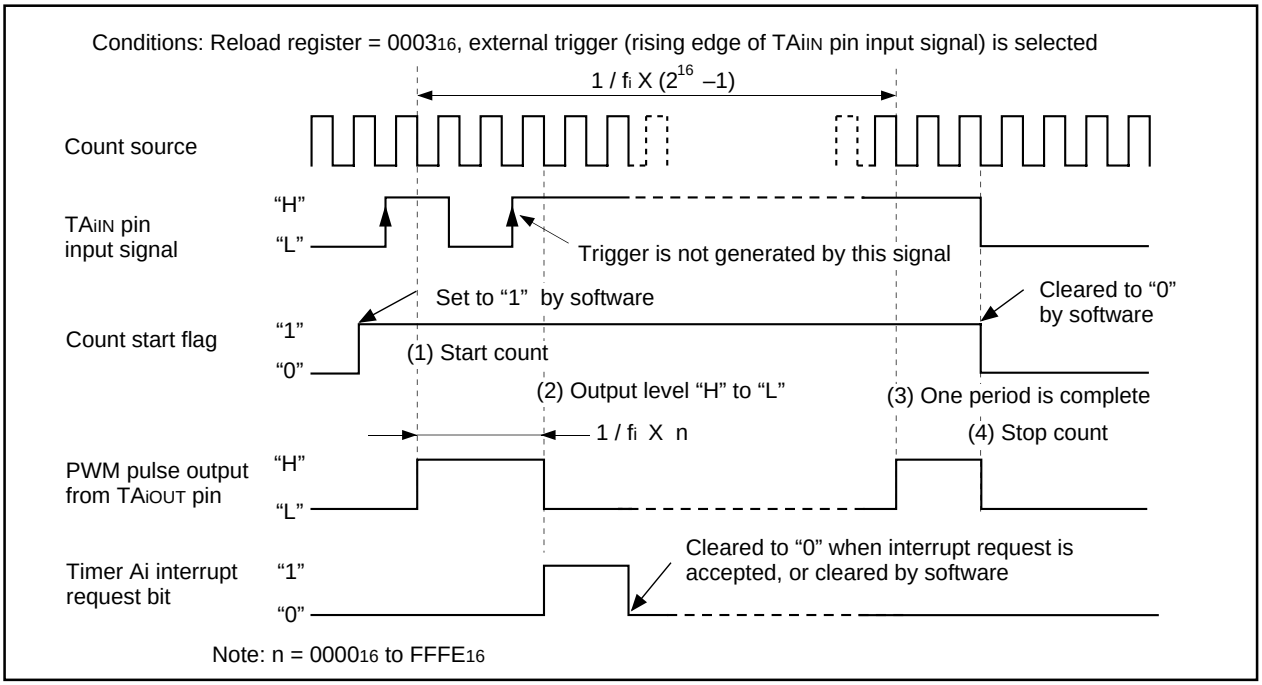


Figure 2.2.24. Operation timing of pulse width modulation mode, 16-bit PWM mode selected

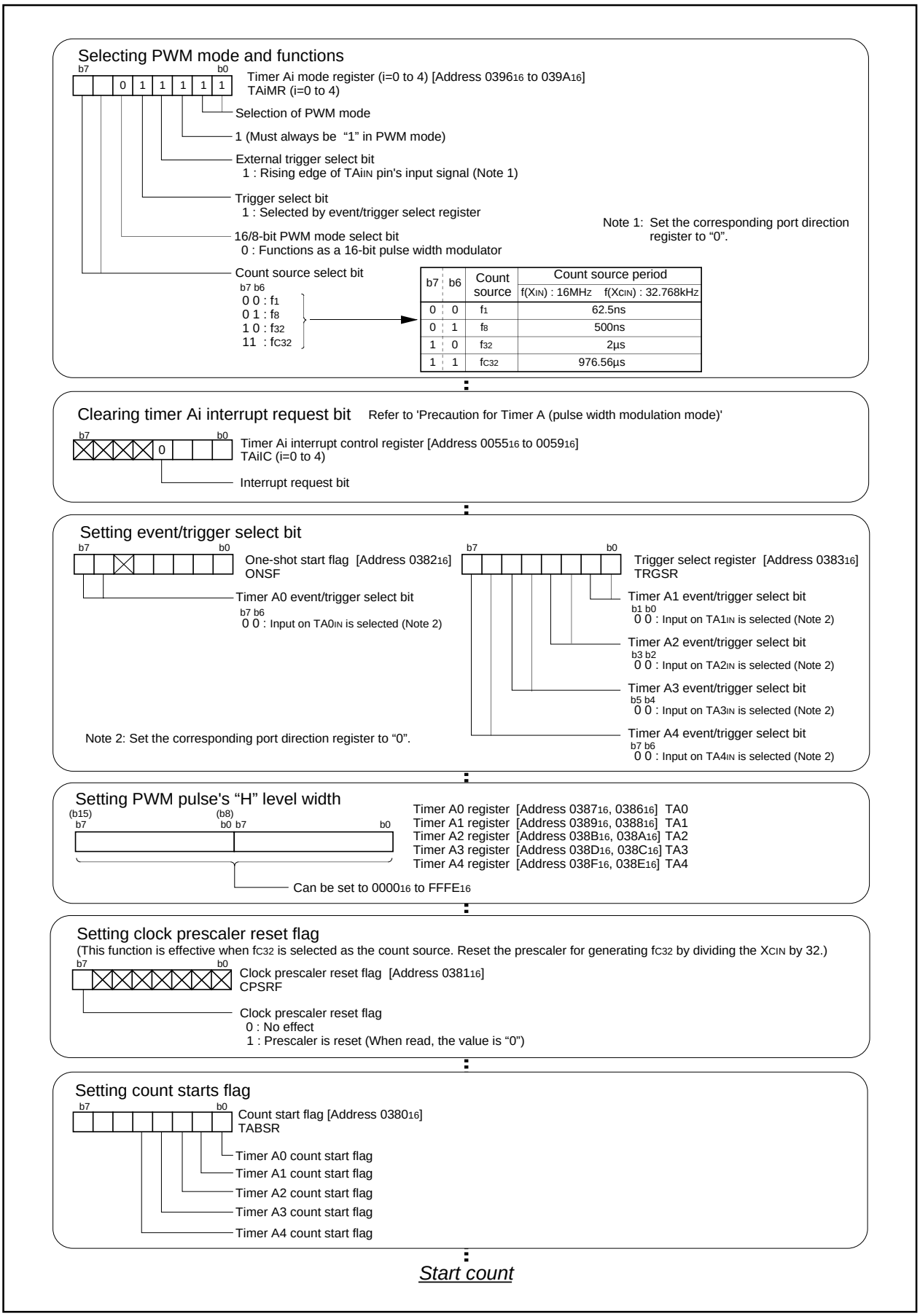


Figure 2.2.25. Set-up procedure of pulse width modulation mode, 16-bit PWM mode selected

2.2.12 Operation of Timer A (pulse width modulation mode, 8-bit PWM mode selected)

In pulse width modulation mode, choose functions from those listed in Table 2.2.12. Operations of the circled items are described below. Figure 2.2.26 shows the operation timing, and Figure 2.2.27 shows the set-up procedure.

Table 2.2.12. Choosed functions

Item	Set-up	
Count source	**O**	Internal count source (f_1 / f_8 / f_{32} / fc_{32})
PWM mode		16-bit PWM
	O	8-bit PWM
Count start condition	**O**	External trigger input (falling edge of input signal to the TAiIN pin)
		External trigger input (rising edge of input signal to the TAiIN pin)
		Timer overflow (TB2/TAj/TAk overflow)

Note: j = i – 1, but j = 4 when i = 0; k = i + 1, but k = 0 when i = 4.

Operation
(1) If the TAiIN pin input level changes from "H" to "L" with the count start flag set to "1", the counter performs a down count on the count source. Also, the TAiOUT pin outputs an "H" level.
(2) The TAiOUT pin output level changes from "H" to "L" when a set time period elapses. At this time, the timer Ai interrupt request bit goes to "1".
(3) The counter reloads the content of the reload register every time PWM pulses are output for one cycle, and continues counting.
(4) Setting the count start flag to "0" causes the counter to hold its value and to stop. Also, the TAiOUT pin outputs an "L" level.

Note
- The period of PWM pulses becomes (m + 1) X (2^8 – 1) / fi, and the "H" level pulse width becomes n X (m + 1) / fi. If "00_{16}" is set in the eight higher-order bits of the timer Ai register, the pulse width modulator does not work, and the TAiOUT pin outputs "L" level, therefore the timer Ai interrupt request is not generated.
  (fi : frequency of the count source f_1, f_8, f_{32}, fc_{32}; n : value of the timer)
- When a trigger is generated, the TAiout pin outputs "L" level of same amplitude as "H" level of the set PWM pulse, after which it starts PWM pulse output.

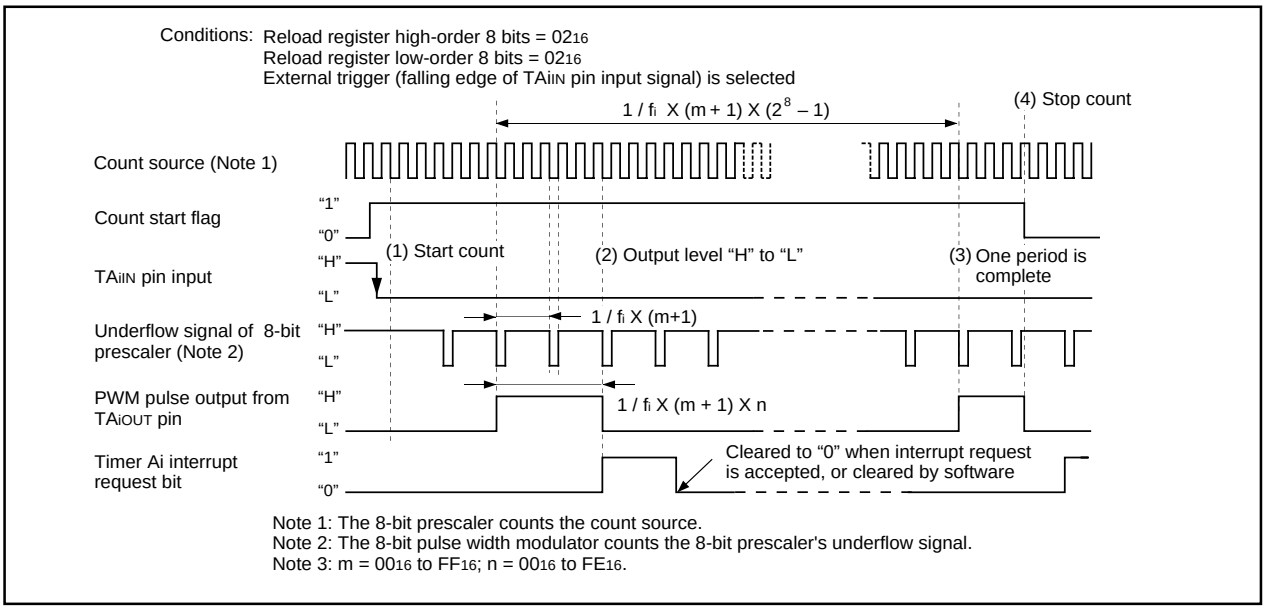


Figure 2.2.26. Operation timing of pulse width modulation mode, with 8-bit PWM mode selected

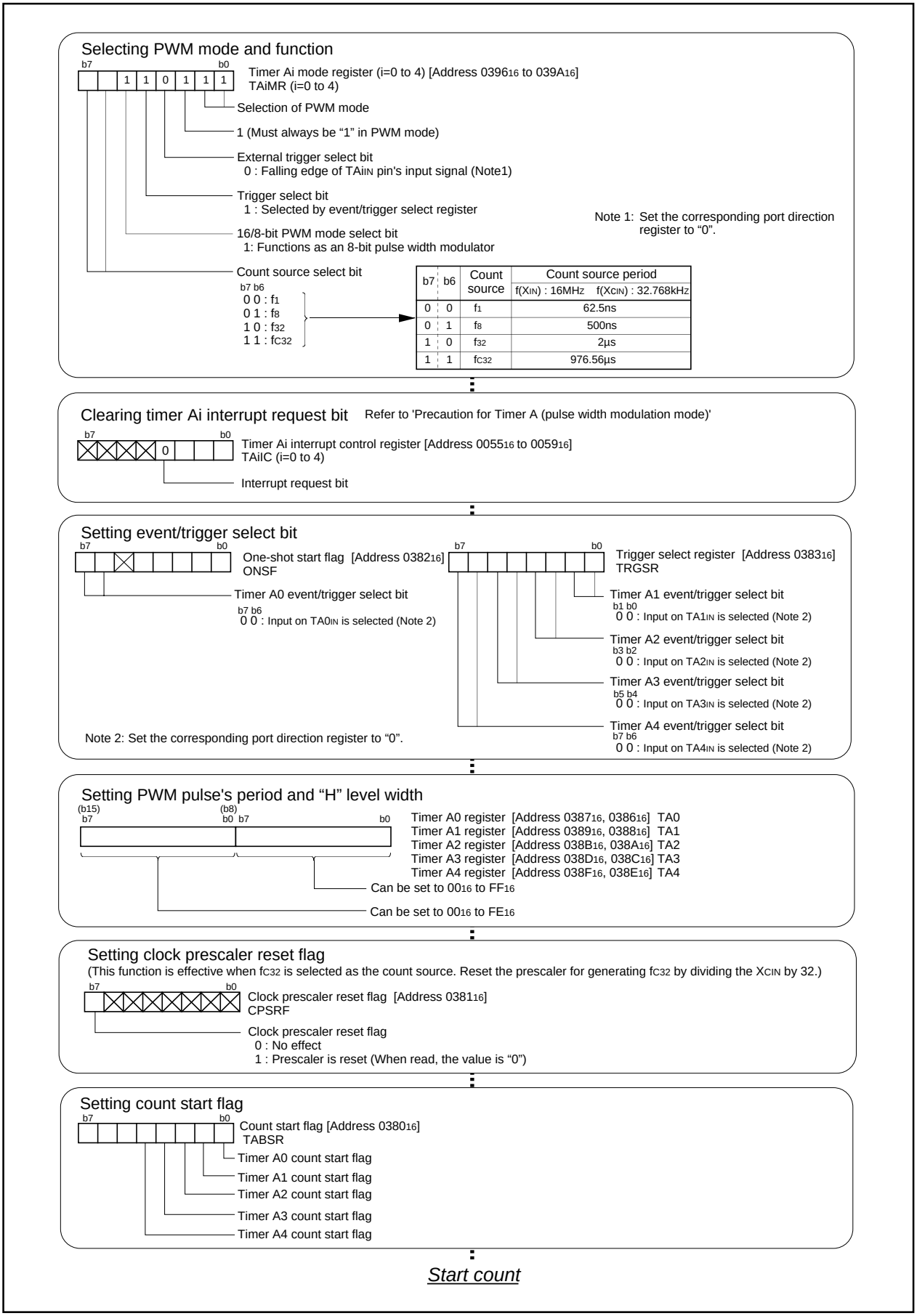


Figure 2.2.27. Set-up procedure of pulse width modulation mode, 8-bit PWM mode selected

2.2.13 Precautions for Timer A (timer mode)

(1) To clear reset, the count start flag is set to “0”. Set a value in the timer Ai register, then set the flag to “1”.

(2) Reading the timer Ai register while a count is in progress allows reading, with arbitrary timing, the value of the counter. Reading the timer Ai register with the reload timing shown in Figure 2.2.28 gets “FFFF16”. Reading the timer Ai register after setting a value in the timer Ai register with a count halted but before the counter starts counting gets a proper value.

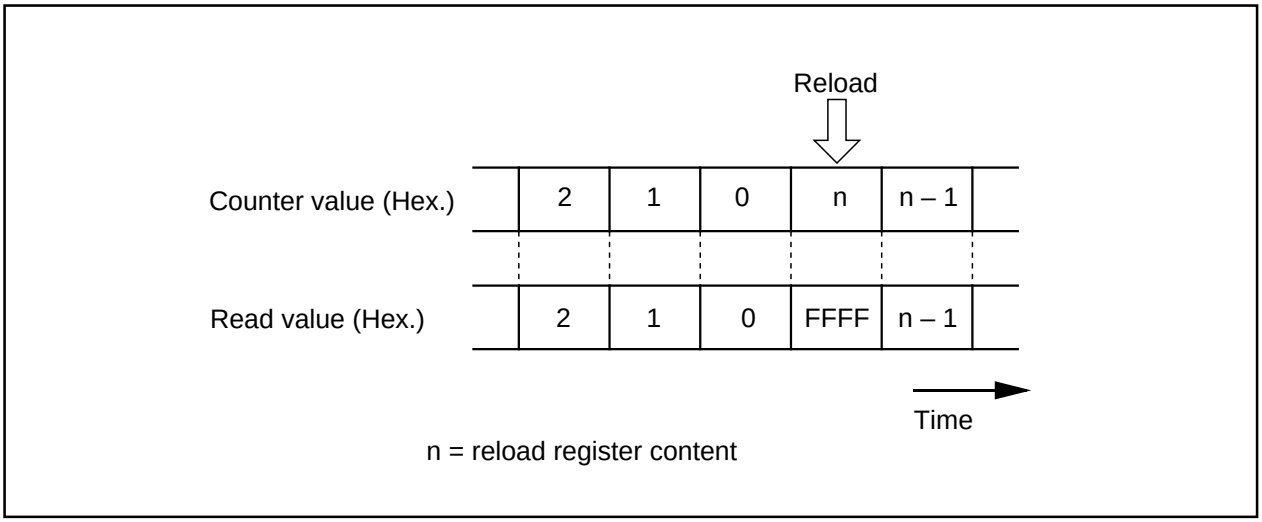


Figure 2.2.28. Reading timer Ai register

2.2.14 Precautions for Timer A (event counter mode)

(1) To clear reset, the count start flag is set to “0”. Set a value in the timer Ai register, then set the flag to “1”.

(2) Reading the timer Ai register while a count is in progress allows reading, with arbitrary timing, the value of the counter. Reading the timer Ai register with the reload timing shown in Figure 2.2.29 gets “FFFF16” by underflow or “000016” by overflow. Reading the timer Ai register after setting a value in the timer Ai register with a count halted but before the counter starts counting gets a proper value.

(3) Please note the standards for the differences between the 2 pulses used in the 2-phase pulse signals input signals to the TAiIN pin and TAiOUT pin (i = 2, 3, 4), as shown in Figure 2.2.30.

(4) When free run type is selected, if count is stopped, set a value in the timer Ai register again.

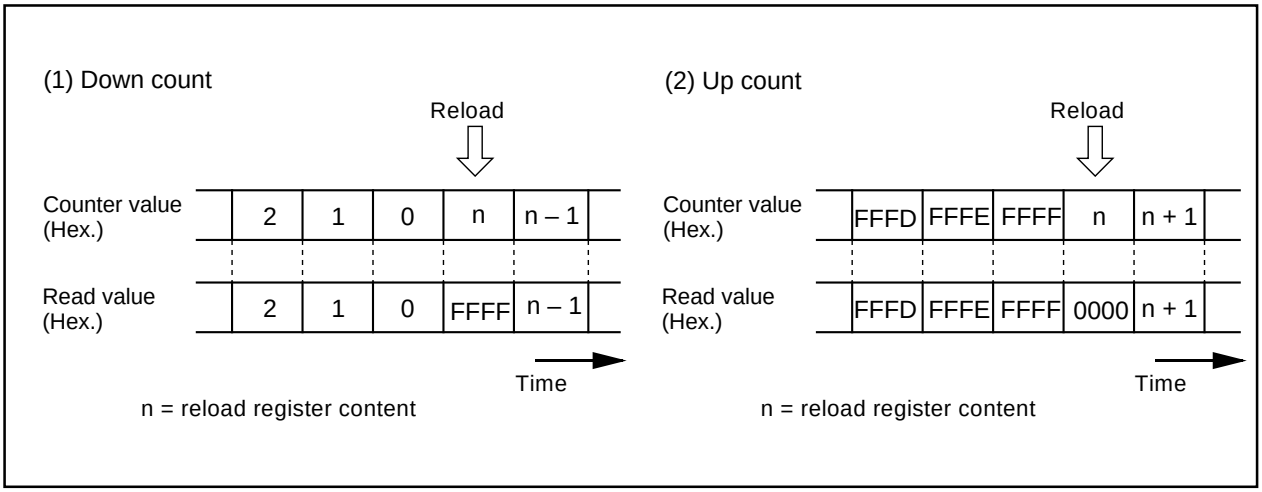


Figure 2.2.29. Reading timer Ai register

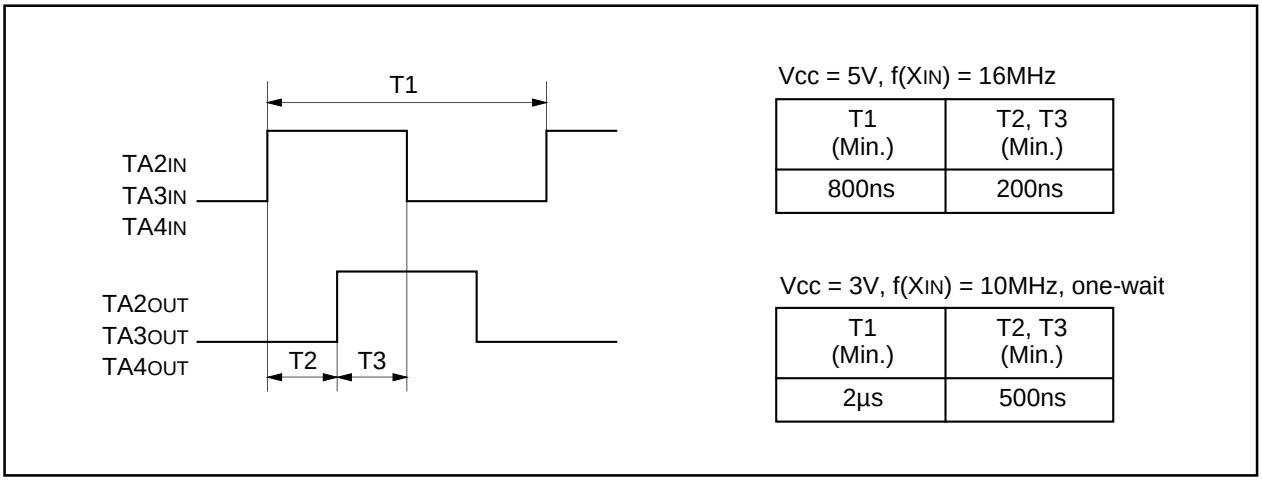


Figure 2.2.30. Standard of 2-phase pulses

2.2.15 Precautions for Timer A (one-shot timer mode)

(1) At reset, the count start flag is set to “0”. Set a value in the timer Ai register, then set the flag to “1”.

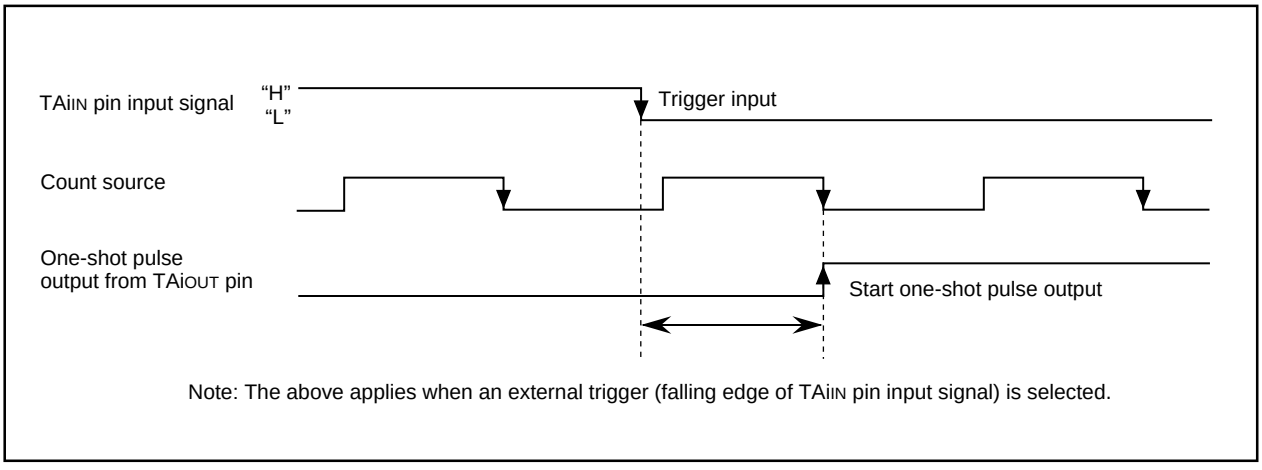
(2) Setting the count start flag to “0” while a count is in progress causes as follows:
- The counter stops counting and a content of reload register is reloaded.
- The TAiOUT pin outputs “L” level.
- The interrupt request generated and the timer Ai interrupt request bit goes to “1”.

(3) The output from the one-shot timer synchronizes with the count source generated internally. Therefore, when an external trigger has been selected, a delay of one cycle of the maximum count source occurs between the trigger input to the TAiIN pin and the one-shot timer output.

(4) The timer Ai interrupt request bit goes to “1” if the timer's operation mode is set using any of the following procedures:
- Selecting one-shot timer mode after reset.
- Changing operation mode from timer mode to one-shot timer mode.
- Changing operation mode from event counter mode to one-shot timer mode.

Therefore, to use timer Ai interrupt (interrupt request bit), set timer Ai interrupt request bit to “0” after the above listed changes have been made.

(5) If a trigger occurs while a count is in progress, after the counter performs one down count following the reoccurrence of a trigger, the reload register contents are reloaded, and the count continues. To generate a trigger while a count is in progress, generate the second trigger after an elapse longer than one cycle of the timer's count source after the previous trigger occurred.



Note: The above applies when an external trigger (falling edge of TAiIN pin input signal) is selected.

Figure 2.2.31. One-shot timer delay

2.2.16 Precautions for Timer A (pulse width modulation mode)

(1) To clear reset, the count start flag is set to “0”. Set a value in the timer Ai register, then set the flag to “1”.

(2) The timer Ai interrupt request bit becomes “1” if setting operation mode of the timer in compliance with any of the following procedures:
- Selecting PWM mode after reset.
- Changing operation mode from timer mode to PWM mode.
- Changing operation mode from event counter mode to PWM mode.

Therefore, to use timer Ai interrupt (interrupt request bit), set timer Ai interrupt request bit to “0” after the above listed changes have been made.

(3) Setting the count start flag to “0” while PWM pulses are being output causes the counter to stop counting. If the TAiOUT pin is outputting an “H” level in this instance, the output level goes to “L”, and the timer Ai interrupt request bit goes to “1”. If the TAiOUT pin is outputting an “L” level in this instance, the level does not change, and the timer Ai interrupt request bit does not becomes “1”.

2.3 Timer B Usage

2.3.1 Overview of the timer B usage

Timer B is a 16-bit timer. The following is an overview of the timer B usage.

(1) Mode

Timer B operates in one of three modes:

(a) Timer mode

The internal count source is counted.

• Operation in timer mode P2-42

(b) Event counter mode

The number of pulses coming from outside and the number of the timer overflows are counted.

• Operation in event counter mode P2-44

(c) Pulse period measurement/pulse width measurement mode

External pulse period or external pulse widths are measured. If pulse period measurement mode is selected, the periods of input pulses are continuously measured. If pulse width measurement mode is selected, widths of “H” level pulses and those of “L” level pulses are continuously measured.

• Operation in pulse period measurement mode P2-46
• Operation in pulse width measurement mode P2-48

(2) Count source

An internal count source can be selected from f_1, f_8, f_{32}, and f_{C32}. f_1, f_8, and f_{32} are clocks obtained by dividing the CPU main clock by 1, 8, and 32 respectively. f_{C32} is the clock obtained by dividing the CPU secondary clock by 32.

(3) Count value

The count value equals [the value set in the timer register + 1]. The counter underflows when a count source equal to a count value is input, and an interrupt request occurs.

(4) Reading the timer

In timer mode or event counter mode, the count value at the time of reading the timer register will be read. Read the register in 16-bit increments. In both the pulse period measurement mode and pulse width measurement mode, an indeterminate value is read until the second effective edge is input after a count is started, otherwise, the measurement results are read.

(5) Writing to the timer

When writing to the timer register while a count is in progress, the value is written only to the reload register. When writing to the timer register while a count has stopped, the value is written both to the reload register and the count. Write the value in 16-bit increments. The timer register cannot be written to in either the pulse period measurement mode or the pulse width measurement mode.

(6) Input to the timer and the direction register

To input an external signal to the timer, set the direction register of the relevant port to input.

(7) Pins related to timer B

(a) $TB0_{IN}$, $TB1_{IN}$, $TB2_{IN}$, $TB3_{IN}$, $TB4_{IN}$, $TB5_{IN}$:Input pins to timer B.

(8) Registers related to timer B

Figure 2.3.1 shows the memory map of timer B-related registers. Figures 2.3.2 and 2.3.3 show timer B-related registers.

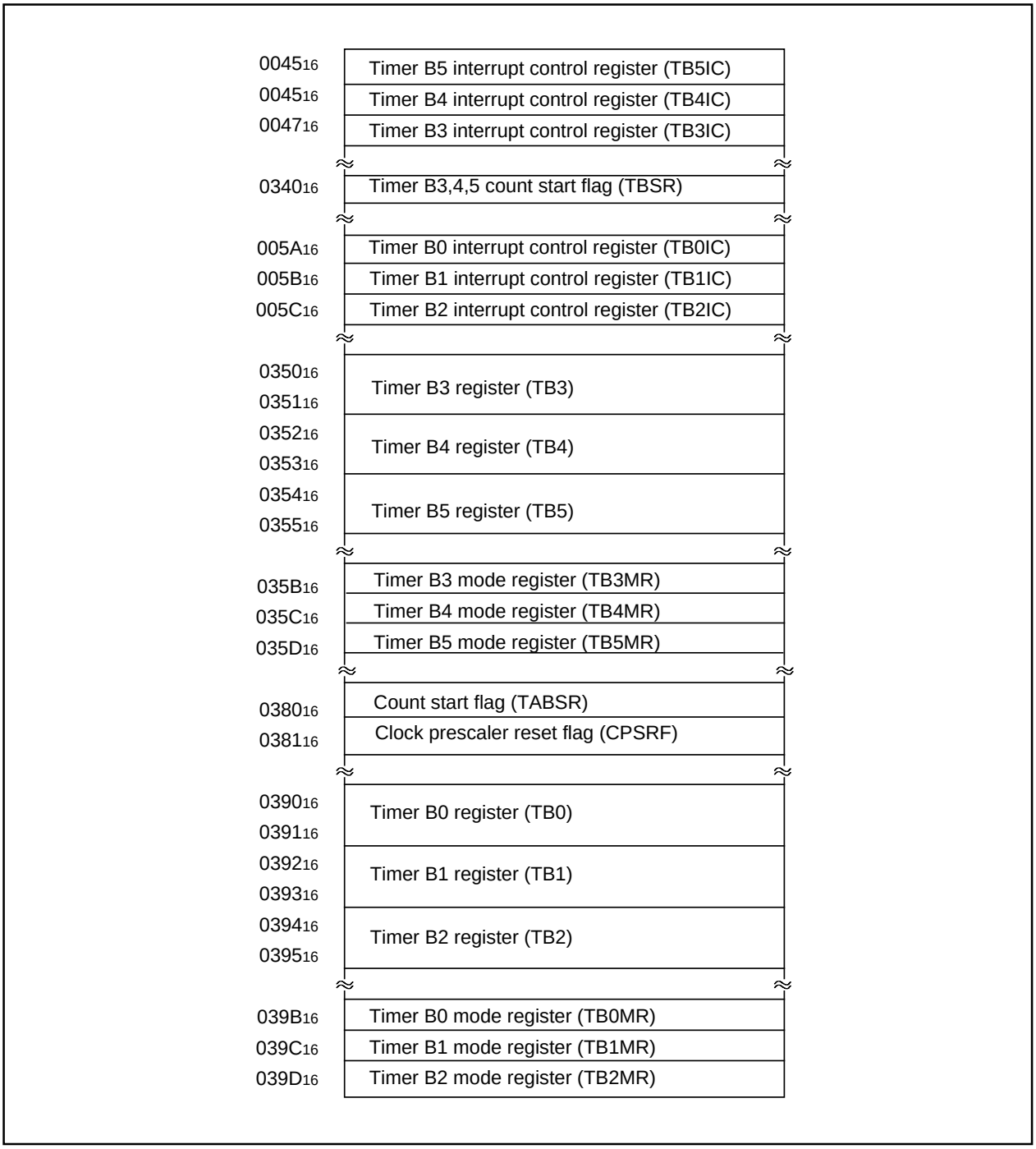
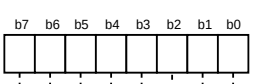


Figure 2.3.1. Memory map of timer B-related registers

Timer Bi mode register



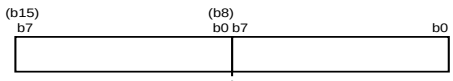
Symbol	Address	When reset
TBiMR(i = 0 to 5)	039B16 to 039D16	00XX00002
	035B16 to 035D16	00XX00002

Bit symbol	Bit name	Function	R	W
TMOD0	Operation mode select bit	b1 b0 0 0 : Timer mode 0 1 : Event counter mode 1 0 : Pulse period/pulse width measurement mode 1 1 : Must not be set.	○	○
TMOD1			○	○
MR0	Function varies with each operation mode		○	○
MR1			○	○
MR2			○ (Note 1)	○
			× (Note 2)	×
MR3			○	×
TCK0	Count source select bit (Function varies with each operation mode)		○	○
TCK1			○	○

Note 1: Timer B0, timer B3.
Note 2: Timer B1, timer B2, timer B4, timer B5.

Figure 2.3.2. Timer B-related registers (1)

Timer Bi register (Note)



Symbol	Address	When reset
TB0	0391_{16}, 0390_{16}	Indeterminate
TB1	0393_{16}, 0392_{16}	Indeterminate
TB2	0395_{16}, 0394_{16}	Indeterminate
TB3	0351_{16}, 0350_{16}	Indeterminate
TB4	0353_{16}, 0352_{16}	Indeterminate
TB5	0355_{16}, 0354_{16}	Indeterminate

Function	Values that can be set	R	W
• Timer mode Counts the timer's period	0000_{16} to $FFFF_{16}$	○	○
• Event counter mode Counts external pulses input or a timer overflow	0000_{16} to $FFFF_{16}$	○	○
• Pulse period / pulse width measurement mode Measures a pulse period or width	——	○	×

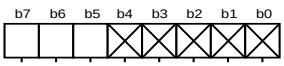
Note: Read and write data in 16-bit units.

Count start flag

Symbol	Address	When reset
TABSR	0380_{16}	00_{16}

Bit symbol	Bit name	Function	R	W
TA0S	Timer A0 count start flag	0 : Stops counting 1 : Starts counting	○	○
TA1S	Timer A1 count start flag		○	○
TA2S	Timer A2 count start flag		○	○
TA3S	Timer A3 count start flag		○	○
TA4S	Timer A4 count start flag		○	○
TB0S	Timer B0 count start flag		○	○
TB1S	Timer B1 count start flag		○	○
TB2S	Timer B2 count start flag		○	○

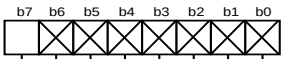
Timer B3, 4, 5 count start flag



Symbol	Address	When reset
TBSR	0340_{16}	$000XXXXX_2$

Bit symbol	Bit name	Function	R	W
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be indeterminate.			—	—
TB3S	Timer B3 count start flag	0 : Stops counting 1 : Starts counting	○	○
TB4S	Timer B4 count start flag		○	○
TB5S	Timer B5 count start flag		○	○

Clock prescaler reset flag



Symbol	Address	When reset
CPSRF	0381_{16}	$0XXXXXXX_2$

Bit symbol	Bit name	Function	R	W
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be indeterminate.			—	—
CPSR	Clock prescaler reset flag	0 : No effect 1 : Prescaler is reset (When read, the value is “0”)	○	○

Figure 2.3.3. Timer B-related registers (2)

2.3.2 Operation of Timer B (timer mode)

In timer mode, choose functions from those listed in Table 2.3.1. Operations of the circled items are described below. Figure 2.3.4 shows the operation timing, and Figure 2.3.5 shows the set-up procedure.

Table 2.3.1. Choosed functions

Item	Set-up	
Count source	**O**	Internal count source (f_1 / f_8 / f_{32} / fc_{32})

Operation (1) Setting the count start flag to “1” causes the counter to perform a down count on the count source.

(2) If an underflow occurs, the content of the reload register is reloaded, and the counter continues counting. At this time, the timer Bi interrupt request bit goes to “1”.

(3) Setting the count start flag to “0” causes the counter to hold its value and to stop.

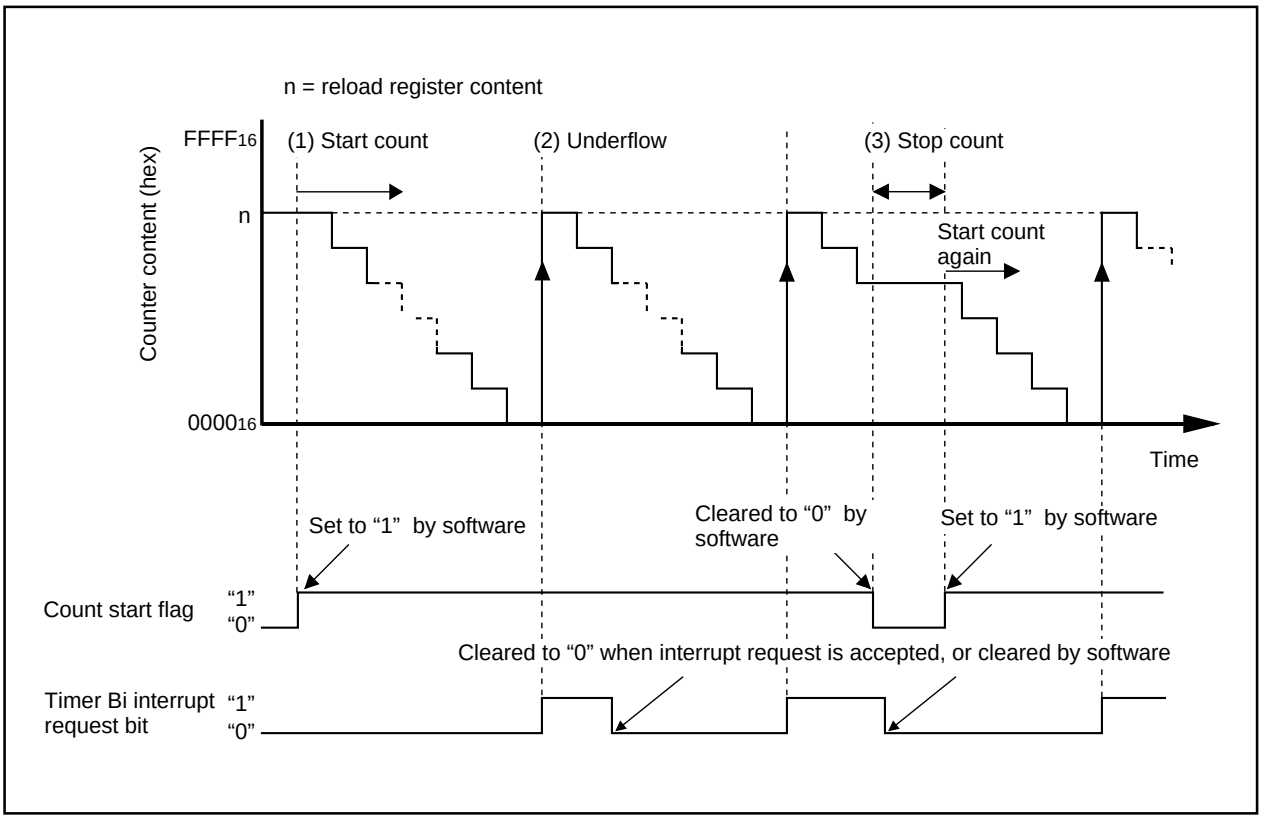


Figure 2.3.4. Operation timing of timer mode

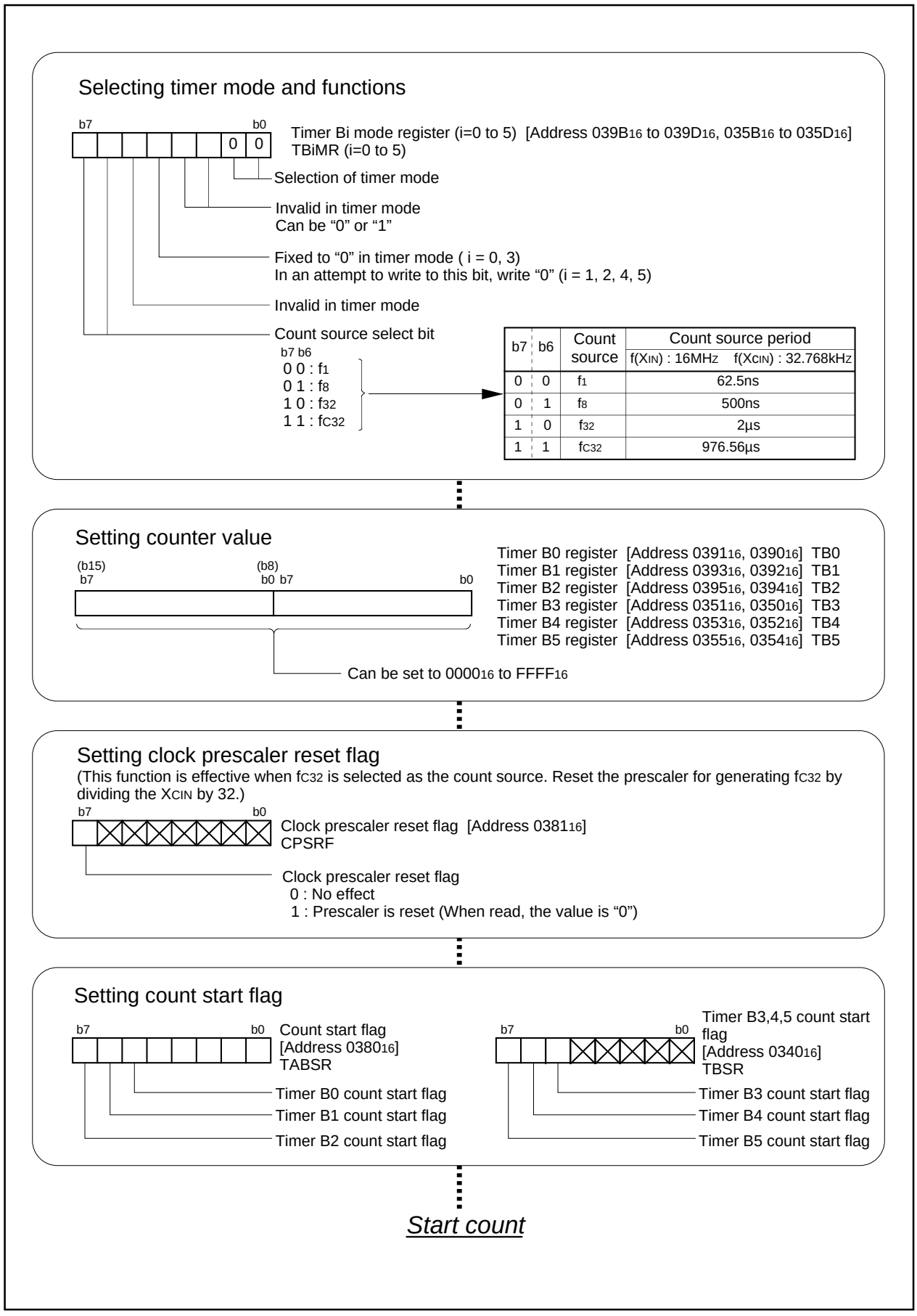


Figure 2.3.5. Set-up procedure of timer mode

2.3.3 Operation of Timer B (event counter mode)

In event counter mode, choose functions from those listed in Table 2.3.2. Operations of the circled items are described below. Figure 2.3.6 shows the operation timing, and Figure 2.3.7 shows the set-up procedure.

Table 2.3.2. Choosed functions

Item	Set-up	
Count source	**O**	Input signal to the TBiIN pin (counting falling edges)
		Input signal to the TBiIN pin (counting rising edges)
		Input signal to the TBiIN pin (counting rising edges and falling edges)
		Timer overflow(TBj overflow)

Note: j = i – 1, but j = 2 when i = 0, j = 5 when i = 3

Operation (1) Setting the count start flag to “1” causes the counter to count the falling edges of the count source.

(2) If an underflow occurs, the content of the reload register is reloaded, and the count continues. At this time, the timer Bi interrupt request bit goes to “1”.

(3) Setting the count start flag to “0” causes the counter to hold its value and to stop.

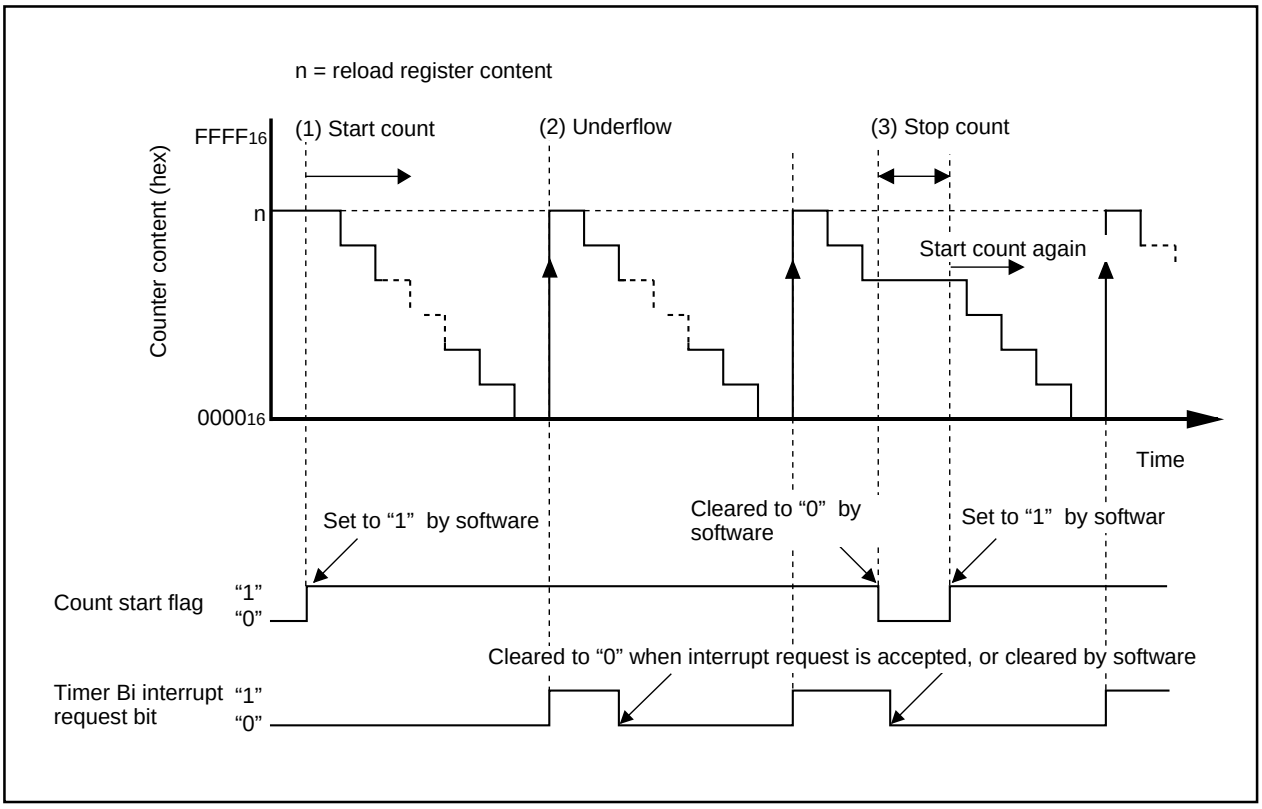


Figure 2.3.6. Operation timing of event counter mode

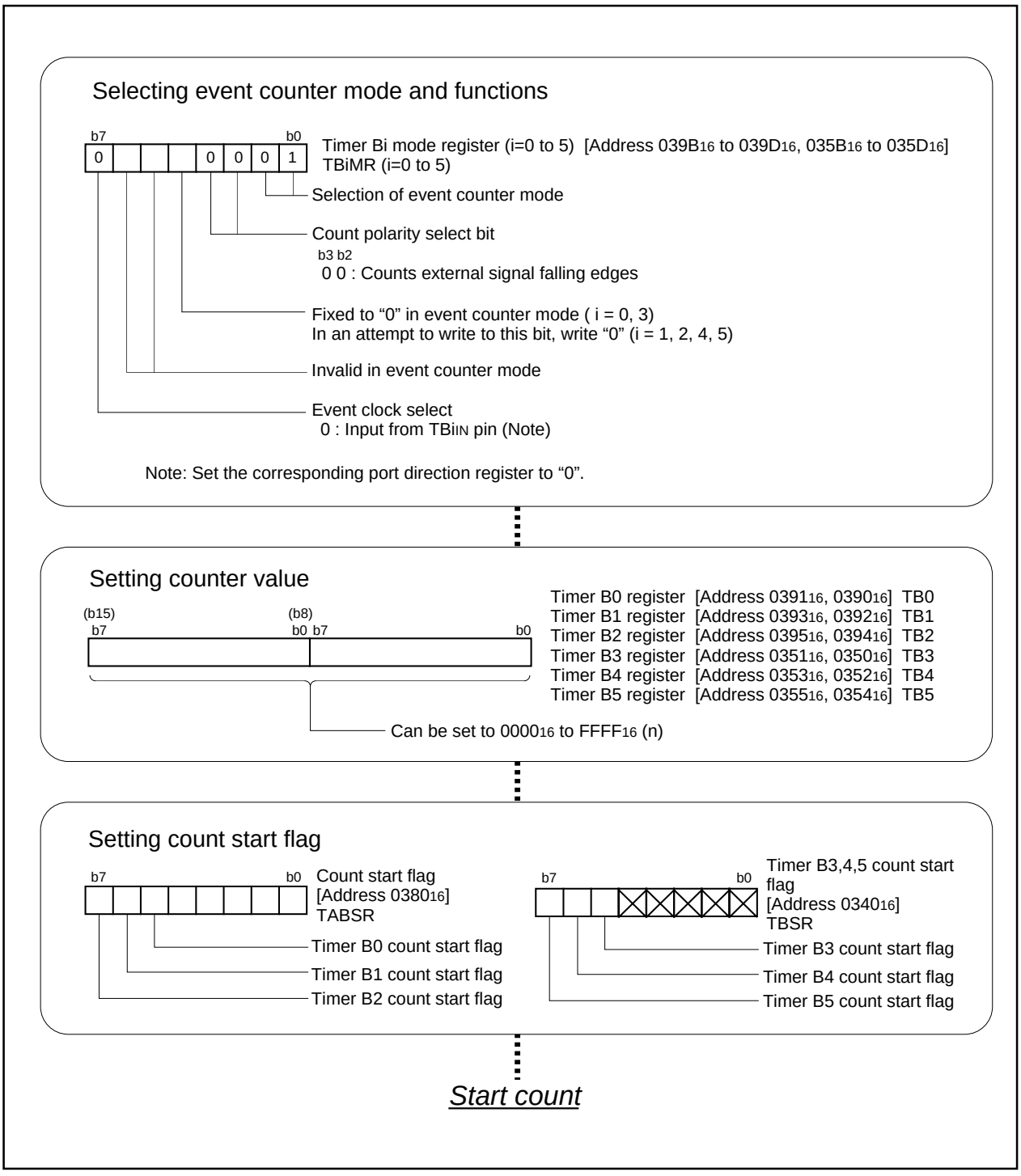


Figure 2.3.7. Set-up procedure of event counter mode

2.3.4 Operation of Timer B (pulse period measurement mode)

In pulse period/pulse width measurement mode, choose functions from those listed in Table 2.3.3. Operations of the circled items are described below. Figure 2.3.8 shows the operation timing, and Figure 2.3.9 shows the set-up procedure.

Table 2.3.3. Choosed functions

Item	Set-up	
Count source	**O**	Internal count source (f_1 / f_8 / f_{32} / fc_{32})
Measurement mode	**O**	Pulse period measurement (interval between measurement pulse falling edge to falling edge)
		Pulse period measurement (interval between measurement pulse rising edge to rising edge)
		Pulse width measurement (interval between measurement pulse falling edge to rising edge, and between rising edge to falling edge)

Operation (1) Setting the count start flag to “1” causes the counter to start counting the count source.

(2) If a measurement pulse changes from “H” to “L”, the value of the counter goes to “0000_{16}”, and measurement is started. In this instance, an indeterminate value is transferred to the reload register. The timer Bi interrupt request does not generate.

(3) If a measurement pulse changes from “H” to “L” again, the value of the counter is transferred to the reload register, and the timer Bi interrupt request bit goes to “1”. Then the value of the counter becomes “0000_{16}”, and the measurement is started again.

Note
- The timer Bi interrupt request bit goes to “1” when an effective edge of a measurement pulse is input or timer Bi is overflowed. The factor of interrupt request can be determined by use of the timer Bi overflow flag within the interrupt routine.
- The value of the counter at the beginning of a count is indeterminate. Therefore, the timer Bi overflow flag may go to “1” and timer Bi interrupt request may be generated during the interval between a count start and an effective edge input.
- The timer Bi overflow flag is indeterminate after reset. The timer Bi overflow flag goes to “0” if timer Bi mode register is written to when the count start flag is “1”. This flag cannot be set to “1” by software.

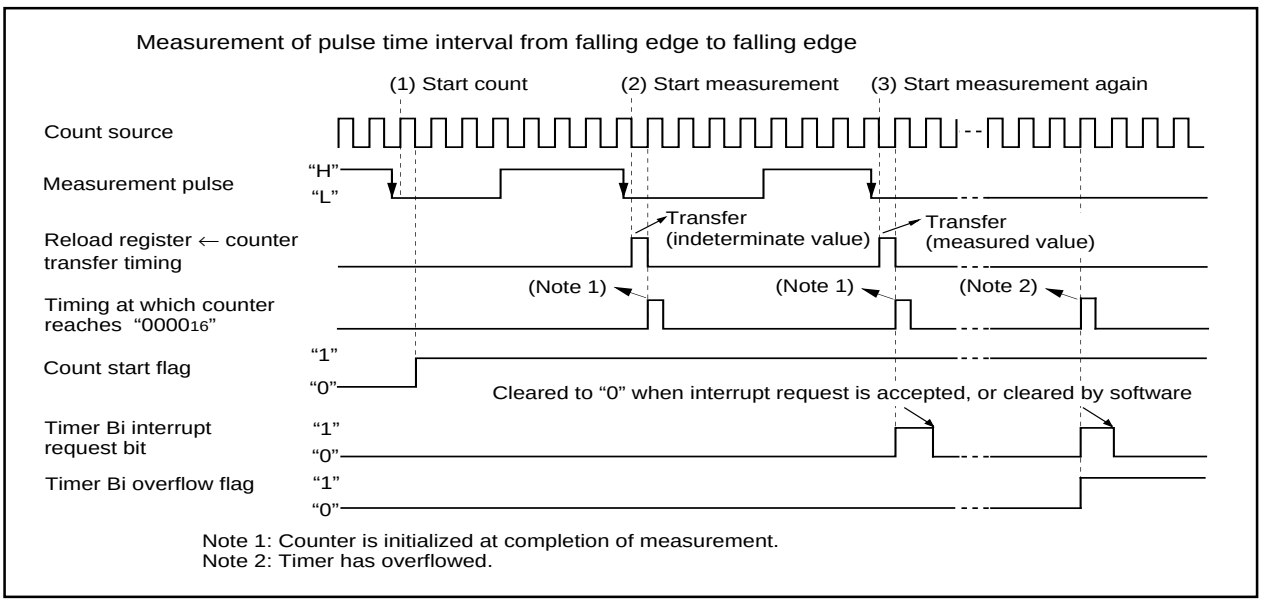


Figure 2.3.8. Operation timing of pulse period measurement mode

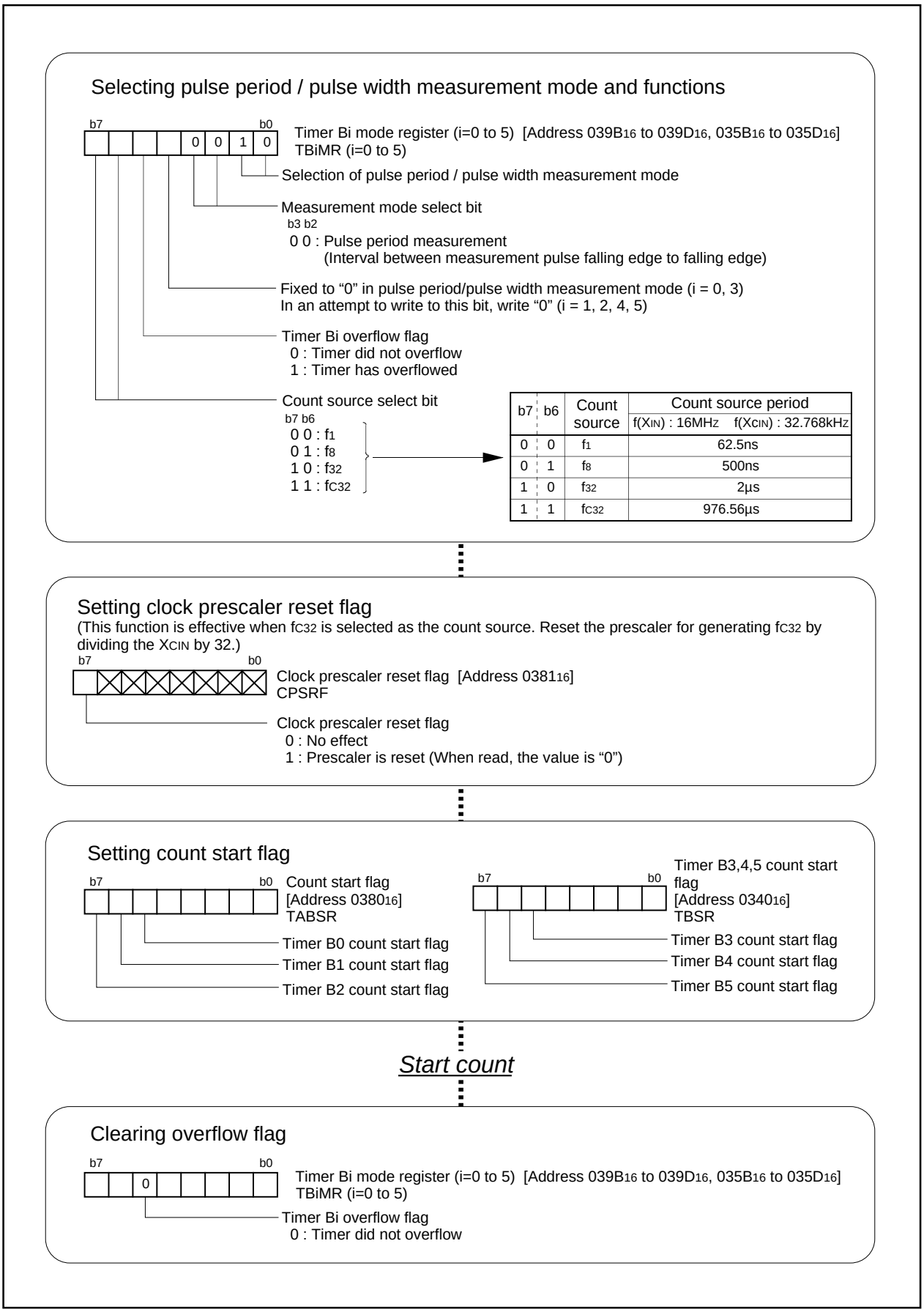


Figure 2.3.9. Set-up procedure of pulse period measurement mode

2.3.5 Operation of Timer B (pulse width measurement mode)

In pulse period/pulse width measurement mode, choose functions from those listed in Table 2.3.4. Operations of the circled items are described below. Figure 2.3.10 shows the operation timing, and Figure 2.3.11 shows the set-up procedure.

Table 2.3.4. Choosed functions

Item	Set-up	
Count source	O	Internal count source (f_1 / f_8 / f_{32} / fc_{32})
Measurement mode		Pulse period measurement (interval between measurement pulse falling edge to falling edge)
		Pulse period measurement (interval between measurement pulse rising edge to rising edge)
	O	Pulse width measurement (interval between measurement pulse falling edge to rising edge, and between rising edge to falling edge)

Operation (1) Setting the count start flag to “1” causes the counter to start counting the count source.

(2) If an effective edge of a pulse to be measured is input, the value of the counter goes to “0000_{16}”, and measurement is started. In this instance, an indeterminate value is transferred to the reload register. The timer Bi interrupt request does not generate.

(3) If an effective edge of a pulse to be measured is input again, the value of the counter is transferred to the reload register, and the timer Bi interrupt request bit goes to “1”. Then the value of the counter becomes “0000_{16}”, and measurement is started again.

Note
- The timer Bi interrupt request bit goes to “1” when an effective edge of a pulse to be measured is input or timer Bi is overflows. The factor of interrupt request can be determined by use of the timer Bi overflow flag within the interrupt routine.
- The value of the counter at the beginning of a count is indeterminate. Therefore, the timer Bi overflow flag may go to “1” and timer Bi interrupt request may be generated during the interval between a count start and an effective edge input.
- The timer Bi overflow flag is indeterminate after reset. The timer Bi overflow flag goes to “0” if timer Bi mode register is written to when the count start flag is “1”. This flag cannot be set to “1” by software.

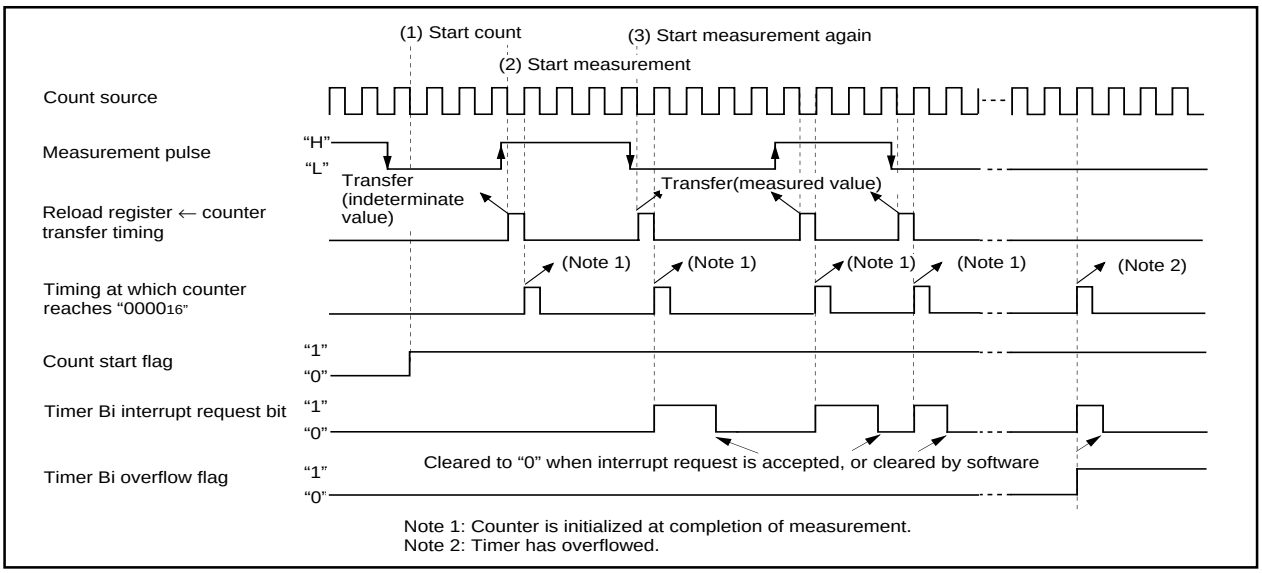


Figure 2.3.10. Operation timing of pulse width measurement mode

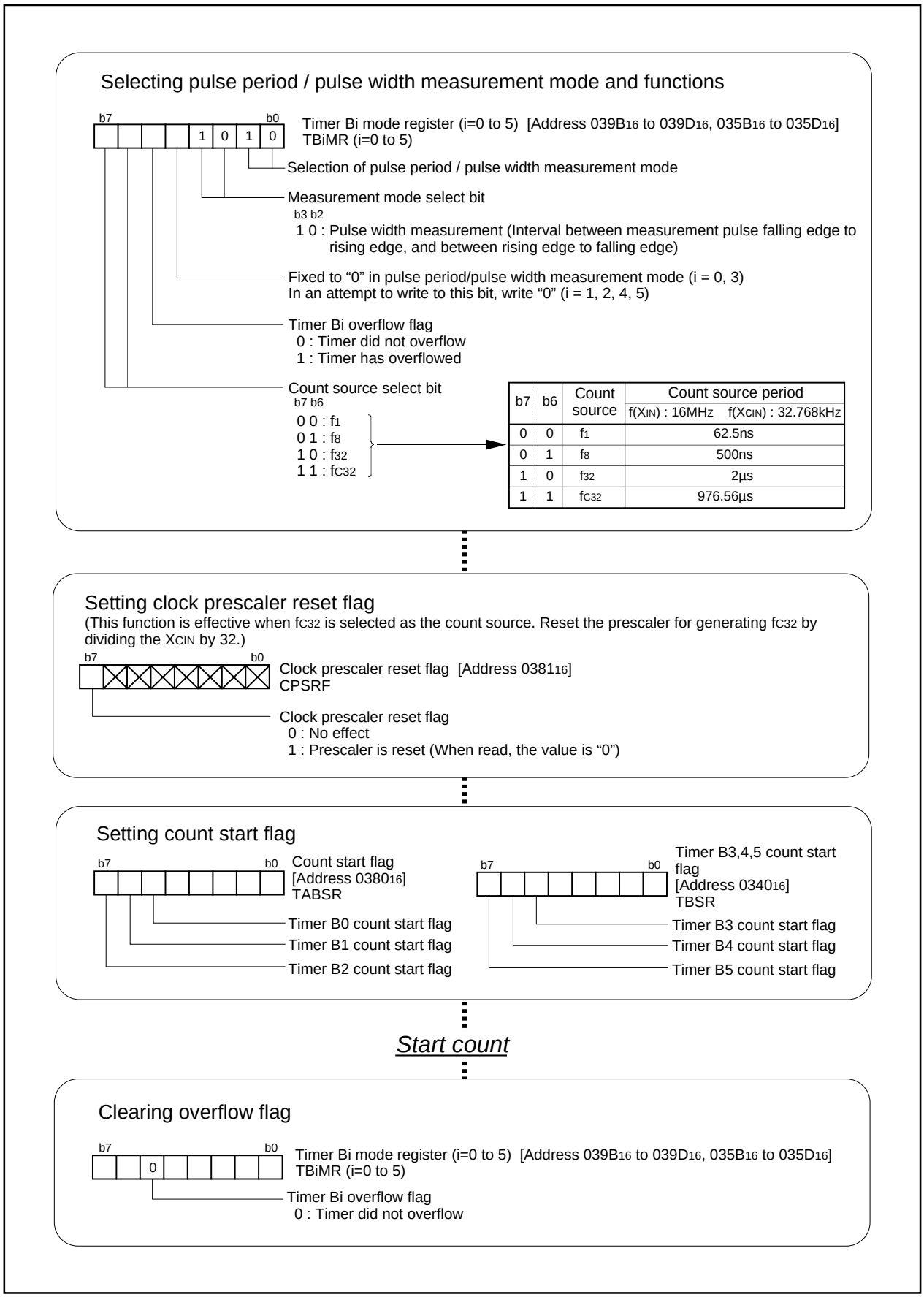


Figure 2.3.11. Set-up procedure of pulse width measurement mode

2.3.6 Precautions for Timer B (timer mode, event counter mode)

(1) To clear reset, the count start flag is set to “0”. Set a value in the timer Bi register, then set the flag to “1”.

(2) Reading the timer Bi register while a count is in progress allows reading, with arbitrary timing, the value of the counter. Reading the timer Bi register with the reload timing shown in Figure 2.3.12 gets “FFFF16”. Reading the timer Bi register after setting a value in the timer Bi register with a count halted but before the counter starts counting gets a proper value.

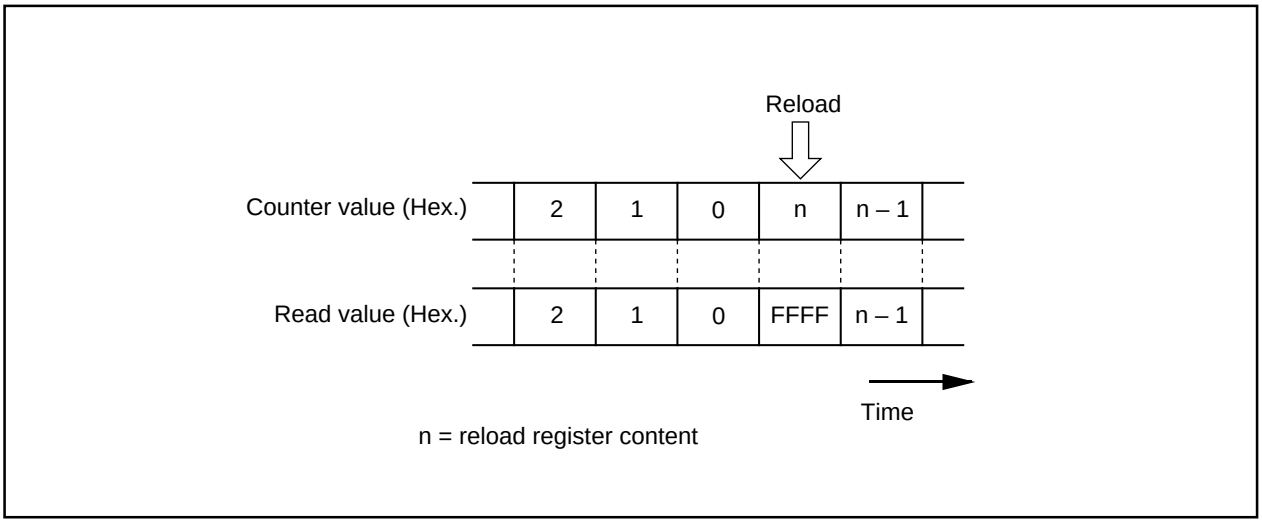


Figure 2.3.12. Reading timer Bi register

2.3.7 Precautions for Timer B (pulse period/pulse width measurement mode)

(1) The timer Bi interrupt request bit goes to “1” when an effective edge of a measurement pulse is input or timer Bi is overflowed. The factor of interrupt request can be determined by use of the timer Bi overflow flag within the interrupt routine.

(2) If the timer overflow occurs simultaneously with the input of a measurement pulse, and if the interrupt factor cannot be determined from the timer Bi overflow flag, connect the timers and count the number of overflows.

(3) When reset, the timer Bi overflow flag goes to “1”. This flag can be set to “0” by writing to the timer Bi mode register when the count start flag is “1”.

(4) Use the timer Bi interrupt request bit to detect only overflows. Use the timer Bi overflow flag only to determine the interrupt factor within the interrupt routine.

(5) When the first effective edge is input after a count is started, an indeterminate value is transferred to the reload register. At this time, timer Bi interrupt request is not generated.

(6) The value of the counter is indeterminate at the beginning of a count. Therefore, the timer Bi overflow flag may go to “1” and timer Bi interrupt request may be generated during the interval between a count start and an effective edge input.

(7) If changing the measurement mode select bits are set after a count is started, the timer Bi interrupt request bit goes to “1”. Note that the timer Bi interrupt request bit does not change if the same value as before is written to the measurement mode select bits.

(8) If the input signal to the TBiIN pin is affected by noise, precise measurement may not be performed in some cases. It is recommended to see that measurements fall within a specific range by use of software.

(9) For pulse width measurement, pulse widths are successively measured. Use software to check whether the measurement result is an “H” level width or an “L” level width.

2.4 Clock-Synchronous Serial I/O Usage

2.4.1 Overview of the clock-synchronous serial I/O usage

Clock-synchronous serial I/O carries out 8-bit data communications in synchronization with the clock. The following is an overview of the clock-synchronous serial I/O usage.

(1) Transmission/reception format

8-bit data

(2) Transfer rate

If the internal clock is selected as the transfer clock, the divide-by-2 frequency, resulting from the bit rate generator division, becomes the transfer rate. The bit rate generator count source can be selected from the following: f_1, f_8, and f_{32}. Clocks f_1, f_8, and f_{32} are derived by dividing the CPU's main clock by 1, 8, and 32 respectively.

Furthermore, if an external clock is selected as the transfer clock, the clock frequency input to the CLK pin becomes the transfer rate.

(3) Error detection

Only overrun error can be detected. Overrun error is an error that occurs when the next data is made ready before the reception buffer register is read.

(4) How to deal with an error

When receiving data, read an error flag and reception data simultaneously to determine which error has occurred. If the data read is erroneous, initialize the error flag and the UARTi receive buffer register, then receive the data again.

To initialize the UARTi receive buffer register

1. Set the receive enable bit to "0" (disable reception).
2. Set the serial I/O mode select bit to "000_2" (invalid serial I/O).
3. Set the serial I/O mode select bit.
4. Set the receive enable bit to "1" again (enable reception).

To transmit data again due to an error on the reception side, set the UARTi transmit buffer register again, then transmit the data again.

To set the UARTi transmit buffer register again

1. Set the serial I/O mode select bits to "000_2" (invalidate serial I/O).
2. Set the serial I/O mode select bits again.
3. Set the transmit enable bit to "1" (enable transmission), then set transmission data in the UARTi transmit buffer register.

(5) Function selection

For clock-synchronous serial I/O, the following functions can be selected:

(a) $\overline{\text{CTS}}/\overline{\text{RTS}}$ function

In the $\overline{\text{CTS}}$ function, an external IC can start transmission/reception by inputting an "H" level to the $\overline{\text{CTS}}$ pin. The $\overline{\text{CTS}}$ pin input level is detected when transmission/reception starts. Therefore, if the level is set to "L" during transmission/reception, it will stop from the next data.

The $\overline{\text{RTS}}$ function informs an external IC that $\overline{\text{RTS}}$ is reception-ready and has changed to "L". $\overline{\text{RTS}}$ goes to "H" at the falling edge of the transfer clock.

The clock-synchronous serial I/O has three types of $\overline{CTS}/\overline{RTS}$ functions to choose from:

- $\overline{CTS}/\overline{RTS}$ functions disabled — $\overline{CTS}/\overline{RTS}$ pin is a programmable I/O port.
- $\overline{CTS}$ function only enabled — $\overline{CTS}/\overline{RTS}$ pin performs the $\overline{CTS}$ function.
- $\overline{RTS}$ function only enabled — $\overline{CTS}/\overline{RTS}$ pin performs the $\overline{RTS}$ function.

(b) Function for choosing polarity

This function switches the polarity of the transfer clock. The following operations are available:

- Data is input at the falling edge of the transfer clock, and is output at the rising edge.
- Data is input at the rising edge of the transfer clock, and is output at the falling edge.

(c) Function for choosing which bit to transmit first

This function is to choose whether to transmit data from bit 0 or from bit 7. Choose either of the following:

- LSB first — Data is transmitted from bit 0.
- MSB first — Data is transmitted from bit 7.

(d) Function for choosing successive reception mode

Successive reception mode is a mode in which reading the receive buffer register makes the reception-enabled status ready. In this mode, there is no need to write dummy data to the transmit buffer register so as to make the reception-enabled status ready. But at the time of starting reception, read the receive buffer register into a dummy manner.

- Normal mode — Writing dummy data to the transmit buffer register makes the reception enabled status ready.
- Successive reception mode — Reading the reception buffer register makes the reception-enabled status ready.

(e) Function for outputting transfer clock to multiple pins

This function is to switch among pins to output the transfer clock. This function is effective only when selecting the internal clock. Switching among pins for outputting the transfer clock allows data transmission to two external ICs in a time-sharing manner.

(f) Data logic select function

This function is to reserve data when writing to transmit buffer register or reading from receive buffer register.

(g) Function for choosing a transmission interrupt factor

The timing to generate a transmission interrupt can be selected from the following: the instant the transmission buffer is emptied or the instant the transmission register is emptied. When transmission buffer empty timing is selected, an interrupt occurs when transmitted data is moved from the transmission buffer to the transmission register. Therefore, data can be transmitted in succession. When transmission register empty timing is selected, an interrupt occurs when data transmission is complete.

(h) TxD, RxD I/O polarity reverse function

This function is to reserve a polarity of TxD port output level and a polarity of RxD port input level.

Following are some examples in which various functions (a) through (g) are selected:

- Transmission Operation WITH: $\overline{CTS}$ function, transmission at falling edge of transfer clock, LSB First, interrupt at instant transmission buffer is emptied; WITHOUT transfer clock output to multiple pins function P2-60
- Transmission Operation WITH: $\overline{CTS}$/$\overline{RTS}$ function disabled, transmission at falling edge of transfer clock, LSB First, interrupt at instant transmission is completed; WITH transfer clock output to multiple pins function (UART0 selection available) P2-64
- Reception WITH: $\overline{RTS}$ function, reception at falling edge of transfer clock, LSB First, successive reception mode disabled; WITHOUT transfer clock output to multiple pins function P2-68

(6) Input to the serial I/O and the direction register

To input an external signal to the serial I/O, set the direction register of the relevant port to input.

(7) Pins related to the serial I/O

- $\overline{CTS0}$, $\overline{CTS1}$, $\overline{CTS2}$ pins — Input pins for the $\overline{CTS}$ function
- $\overline{RTS0}$, $\overline{RTS1}$, $\overline{RTS2}$ pins — Output pins for the $\overline{RTS}$ function
- CLK0, CLK1, CLK2 pins — Input/output pins for the transfer clock
- RxD0, RxD1, RxD2 pins — Input pins for data
- TxD0, TxD1, TxD2 pins — Output pins for data (Since TxD2 pin is N-channel open drain, this pin needs pull-up resistor.)
- CLKS1 pin — Output pin for transfer clock. Can be used as transfer clock output pin in the transfer clock output to multiple pins function.

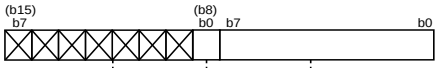
(8) Registers related to the serial I/O

Figure 2.4.1 shows the memory map of serial I/O-related registers, and Figures 2.4.2 to 2.4.6 show serial I/O-related registers.

004F16	UART2 transmit interrupt control register (S2TIC)
005016	UART2 receive interrupt control register (S2RIC)
005116	UART0 transmit interrupt control register (S0TIC)
005216	UART0 receive interrupt control register (S0RIC)
005316	UART1 transmit interrupt control regster(S1TIC)
005416	UART1 receive interrupt control register(S1RIC)
≈	≈
037816	UART2 transmit/receive mode register (U2MR)
037916	UART2 bit rate generator (U2BRG)
037A16 037B16	UART2 transmit buffer register (U2TB)
037C16	UART2 transmit/receive control register 0 (U2C0)
037D16	UART2 transmit/receive control register 1 (U2C1)
037E16 037F16	UART2 receive buffer register (U2RB)
≈	≈
03A016	UART0 transmit/receive mode register (U0MR)
03A116	UART0 bit rate generator (U0BRG)
03A216 03A316	UART0 transmit buffer register (U0TB)
03A416	UART0 transmit/receive control register 0 (U0C0)
03A516	UART0 transmit/receive control register 1 (U0C1)
03A616 03A716	UART0 receive buffer register (U0RB)
03A816	UART1 transmit/receive mode register (U1MR)
03A916	UART1 bit rate generator (U1BRG)
03AA16 03AB16	UART1 transmit buffer register (U1TB)
03AC16	UART1 transmit/receive control register 0 (U1C0)
03AD16	UART1 transmit/receive control register 1 (U1C1)
03AE16 03AF16	UART1 receive buffer register (U1RB)
03B016	UART transmit/receive control register 2 (UCON)
03B116	

Figure 2.4.1. Memory map of serial I/O-related registers

UARTi transmit buffer register (Note)

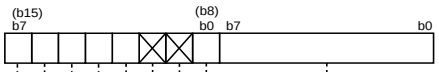


Symbol	Address	When reset
U0TB	$03A3_{16}$, $03A2_{16}$	Indeterminate
U1TB	$03AB_{16}$, $03AA_{16}$	Indeterminate
U2TB	$037B_{16}$, $037A_{16}$	Indeterminate

Function	R	W
Transmit data	×	○
Nothing is assigned. In an attempt to write to these bits, write "0". The value, if read, turns out to be indeterminate.	—	—

Note: Use MOV instruction to write to this register.

UARTi receive buffer register

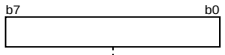


Symbol	Address	When reset
U0RB	$03A7_{16}$, $03A6_{16}$	Indeterminate
U1RB	$03AF_{16}$, $03AE_{16}$	Indeterminate
U2RB	$037F_{16}$, $037E_{16}$	Indeterminate

Bit symbol	Bit name	Function (During clock synchronous serial I/O mode)	Function (During UART mode)	R	W
—	———	Receive data	Receive data	○	×
Nothing is assigned. In an attempt to write to these bits, write "0". The value, if read, turns out to be "0".				—	—
ABT	Arbitration lost detecting flag (Note 2)	0 : Not detected 1 : Detected	Invalid	○	○
OER	Overrun error flag (Note 1)	0 : No overrun error 1 : Overrun error found	0 : No overrun error 1 : Overrun error found	○	×
FER	Framing error flag (Note 1)	Invalid	0 : No framing error 1 : Framing error found	○	×
PER	Parity error flag (Note 1)	Invalid	0 : No parity error 1 : Parity error found	○	×
SUM	Error sum flag (Note 1)	Invalid	0 : No error 1 : Error found	○	×

Note 1: Bits 15 through 12 are set to "0" when the serial I/O mode select bit (bits 2 to 0 at addresses $03A0_{16}$, $03A8_{16}$ and 0378_{16}) are set to "000_2" or the receive enable bit is set to "0".
(Bit 15 is set to "0" when bits 14 to 12 all are set to "0".) Bits 14 and 13 are also set to "0" when the lower byte of the UARTi receive buffer register (addresses $03A6_{16}$, $03AE_{16}$ and $037E_{16}$) is read out.
Note 2: Arbitration lost detecting flag is allocated to U2RB and noting but "0" may be written. Nothing is assigned in bit 11 of U0RB and U1RB. When write, set "0". The value, if read, turns out to be "0".

UARTi bit rate generator (Note 1, 2)



Symbol	Address	When reset
U0BRG	$03A1_{16}$	Indeterminate
U1BRG	$03A9_{16}$	Indeterminate
U2BRG	0379_{16}	Indeterminate

Function	Values that can be set	R	W
Assuming that set value = n, BRGi divides the count source by n + 1	00_{16} to FF_{16}	×	○

Note 1: Write a value to this register while transmit/receive halts.
Note 2: Use MOV instruction to write to this register.

Figure 2.4.2. Serial I/O-related registers (1)

UARTi transmit/receive mode register

b7 b6 b5 b4 b3 b2 b1 b0

Symbol: UiMR(i=0,1) | Address: $03A0_{16}$, $03A8_{16}$ | When reset: 00_{16}

Bit symbol	Bit name	Function (During clock synchronous serial I/O mode)	Function (During UART mode)	R	W
SMD0	Serial I/O mode select bit	Must be fixed to 001 b2 b1 b0 0 0 0 : Serial I/O invalid 0 1 0 : Must not be set. 0 1 1 : Must not be set. 1 1 1 : Must not be set.	b2 b1 b0 1 0 0 : Transfer data 7 bits long 1 0 1 : Transfer data 8 bits long 1 1 0 : Transfer data 9 bits long 0 0 0 : Serial I/O invalid 0 1 0 : Must not be set. 0 1 1 : Must not be set. 1 1 1 : Must not be set.	○	○
SMD1				○	○
SMD2				○	○
CKDIR	Internal/external clock select bit	0 : Internal clock 1 : External clock (Note)	0 : Internal clock 1 : External clock (Note)	○	○
STPS	Stop bit length select bit	Invalid	0 : One stop bit 1 : Two stop bits	○	○
PRY	Odd/even parity select bit	Invalid	Valid when bit 6 = “1” 0 : Odd parity 1 : Even parity	○	○
PRYE	Parity enable bit	Invalid	0 : Parity disabled 1 : Parity enabled	○	○
SLEP	Sleep select bit	Must always be “0”	0 : Sleep mode deselected 1 : Sleep mode selected	○	○

Note : Set the corresponding port direction register to “0”.

UART2 transmit/receive mode register

b7 b6 b5 b4 b3 b2 b1 b0

Symbol: U2MR | Address: 0378_{16} | When reset: 00_{16}

Bit symbol	Bit name	Function (During clock synchronous serial I/O mode)	Function (During UART mode)	R	W
SMD0	Serial I/O mode select bit	Must be fixed to 001 b2 b1 b0 0 0 0 : Serial I/O invalid 0 1 0 : (Note 1) 0 1 1 : Must not be set. 1 1 1 : Must not be set.	b2 b1 b0 1 0 0 : Transfer data 7 bits long 1 0 1 : Transfer data 8 bits long 1 1 0 : Transfer data 9 bits long 0 0 0 : Serial I/O invalid 0 1 0 : Must not be set. 0 1 1 : Must not be set. 1 1 1 : Must not be set.	○	○
SMD1				○	○
SMD2				○	○
CKDIR	Internal/external clock select bit	0 : Internal clock 1 : External clock (Note 2)	Must always be “0”	○	○
STPS	Stop bit length select bit	Invalid	0 : One stop bit 1 : Two stop bits	○	○
PRY	Odd/even parity select bit	Invalid	Valid when bit 6 = “1” 0 : Odd parity 1 : Even parity	○	○
PRYE	Parity enable bit	Invalid	0 : Parity disabled 1 : Parity enabled	○	○
IOPOL	TxD, RxD I/O polarity reverse bit	0 : No reverse 1 : Reverse Usually set to “0”	0 : No reverse 1 : Reverse Usually set to “0”	○	○

Note 1: Bit 2 to bit 0 are set to “010_2” when I^2C mode is used.
Note 2: Set the corresponding port direction register to “0”.

Figure 2.4.3. Serial I/O-related registers (2)

UARTi transmit/receive control register 0

b7	b6	b5	b4	b3	b2	b1	b0

Symbol: UiC0(i=0,1) — Address: 03A4$_{16}$, 03AC$_{16}$ — When reset: 08$_{16}$

Bit symbol	Bit name	Function (During clock synchronous serial I/O mode)	Function (During UART mode)	R	W
CLK0	BRG count source select bit	b1 b0 0 0 : f_1 is selected 0 1 : f_8 is selected 1 0 : f_{32} is selected 1 1 : Must not be set.	b1 b0 0 0 : f_1 is selected 0 1 : f_8 is selected 1 0 : f_{32} is selected 1 1 : Must not be set.	○	○
CLK1				○	○
CRS	$\overline{CTS}$/$\overline{RTS}$ function select bit	Valid when bit 4 = "0" 0 : $\overline{CTS}$ function is selected (Note 1) 1 : $\overline{RTS}$ function is selected (Note 2)	Valid when bit 4 = "0" 0 : $\overline{CTS}$ function is selected (Note 1) 1 : $\overline{RTS}$ function is selected (Note 2)	○	○
TXEPT	Transmit register empty flag	0 : Data present in transmit register (during transmission) 1 : No data present in transmit register (transmission completed)	0 : Data present in transmit register (during transmission) 1 : No data present in transmit register (transmission completed)	○	×
CRD	$\overline{CTS}$/$\overline{RTS}$ disable bit	0 : $\overline{CTS}$/$\overline{RTS}$ function enabled 1 : $\overline{CTS}$/$\overline{RTS}$ function disabled ($P6_0$ and $P6_4$ function as programmable I/O port)	0 : $\overline{CTS}$/$\overline{RTS}$ function enabled 1 : $\overline{CTS}$/$\overline{RTS}$ function disabled ($P6_0$ and $P6_4$ function as programmable I/O port)	○	○
NCH	Data output select bit	0 : TxDi pin is CMOS output 1 : TxDi pin is N-channel open-drain output	0: TxDi pin is CMOS output 1: TxDi pin is N-channel open-drain output	○	○
CKPOL	CLK polarity select bit	0 : Transmit data is output at falling edge of transfer clock and receive data is input at rising edge 1 : Transmit data is output at rising edge of transfer clock and receive data is input at falling edge	Must always be "0"	○	○
UFORM	Transfer format select bit	0 : LSB first 1 : MSB first	Must always be "0"	○	○

Note 1: Set the corresponding port direction register to "0".
Note 2: The settings of the corresponding port register and port direction register are invalid.

UART2 transmit/receive control register 0

b7	b6	b5	b4	b3	b2	b1	b0
		⊠					

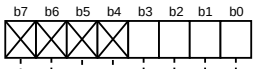
Symbol: U2C0 — Address: 037C$_{16}$ — When reset: 08$_{16}$

Bit symbol	Bit name	Function (During clock synchronous serial I/O mode)	Function (During UART mode)	R	W
CLK0	BRG count source select bit	b1 b0 0 0 : f_1 is selected 0 1 : f_8 is selected 1 0 : f_{32} is selected 1 1 : Must not be set.	b1 b0 0 0 : f_1 is selected 0 1 : f_8 is selected 1 0 : f_{32} is selected 1 1 : Must not be set.	○	○
CLK1				○	○
CRS	$\overline{CTS}$/$\overline{RTS}$ function select bit	Valid when bit 4 = "0" 0 : $\overline{CTS}$ function is selected (Note 1) 1 : $\overline{RTS}$ function is selected (Note 2)	Valid when bit 4 = "0" 0 : $\overline{CTS}$ function is selected (Note 1) 1 : $\overline{RTS}$ function is selected (Note 2)	○	○
TXEPT	Transmit register empty flag	0 : Data present in transmit register (during transmission) 1 : No data present in transmit register (transmission completed)	0 : Data present in transmit register (during transmission) 1 : No data present in transmit register (transmission completed)	○	×
CRD	$\overline{CTS}$/$\overline{RTS}$ disable bit	0 : $\overline{CTS}$/$\overline{RTS}$ function enabled 1 : CTS/RTS function disabled ($P7_3$ functions programmable I/O port)	0 : $\overline{CTS}$/$\overline{RTS}$ function enabled 1 : CTS/RTS function disabled ($P7_3$ functions programmable I/O port)	○	○
Nothing is assigned. In an attempt to write to this bit, write "0". The value, if read, turns out to be "0".				–	–
CKPOL	CLK polarity select bit	0 : Transmit data is output at falling edge of transfer clock and receive data is input at rising edge 1 : Transmit data is output at rising edge of transfer clock and receive data is input at falling edge	Must always be "0"	○	○
UFORM	Transfer format select bit (Note 3)	0 : LSB first 1 : MSB first	0 : LSB first 1 : MSB first	○	○

Note 1: Set the corresponding port direction register to "0".
Note 2: The settings of the corresponding port register and port direction register are invalid.
Note 3: Only clock synchronous serial I/O mode and 8-bit UART mode are valid.

Figure 2.4.4. Serial I/O-related registers (3)

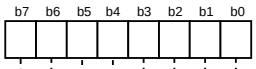
UARTi transmit/receive control register 1



Symbol: UiC1(i=0,1) | Address: 03A5$_{16}$,03AD$_{16}$ | When reset: 02$_{16}$

Bit symbol	Bit name	Function (During clock synchronous serial I/O mode)	Function (During UART mode)	R	W
TE	Transmit enable bit	0 : Transmission disabled 1 : Transmission enabled	0 : Transmission disabled 1 : Transmission enabled	O	O
TI	Transmit buffer empty flag	0 : Data present in transmit buffer register 1 : No data present in transmit buffer register	0 : Data present in transmit buffer register 1 : No data present in transmit buffer register	O	X
RE	Receive enable bit	0 : Reception disabled 1 : Reception enabled	0 : Reception disabled 1 : Reception enabled	O	O
RI	Receive complete flag	0 : No data present in receive buffer register 1 : Data present in receive buffer register	0 : No data present in receive buffer register 1 : Data present in receive buffer register	O	X
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be “0”.				–	–

UART2 transmit/receive control register 1

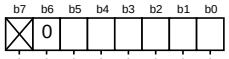


Symbol: U2C1 | Address: 037D$_{16}$ | When reset: 02$_{16}$

Bit symbol	Bit name	Function (During clock synchronous serial I/O mode)	Function (During UART mode)	R	W
TE	Transmit enable bit	0 : Transmission disabled 1 : Transmission enabled	0 : Transmission disabled 1 : Transmission enabled	O	O
TI	Transmit buffer empty flag	0 : Data present in transmit buffer register 1 : No data present in transmit buffer register	0 : Data present in transmit buffer register 1 : No data present in transmit buffer register	O	X
RE	Receive enable bit	0 : Reception disabled 1 : Reception enabled	0 : Reception disabled 1 : Reception enabled	O	O
RI	Receive complete flag	0 : No data present in receive buffer register 1 : Data present in receive buffer register	0 : No data present in receive buffer register 1 : Data present in receive buffer register	O	X
U2IRS	UART2 transmit interrupt cause select bit	0 : Transmit buffer empty (TI = 1) 1 : Transmit is completed (TXEPT = 1)	0 : Transmit buffer empty (TI = 1) 1 : Transmit is completed (TXEPT = 1)	O	O
U2RRM	UART2 continuous receive mode enable bit	0 : Continuous receive mode disabled 1 : Continuous receive mode enabled	Must always be "0"	O	O
U2LCH	Data logic select bit	0 : No reverse 1 : Reverse	0 : No reverse 1 : Reverse	O	O
U2ERE	Error signal output enable bit	Must be fixed to “0”	0 : Output disabled 1 : Output enabled	O	O

Figure 2.4.5. Serial I/O-related registers (4)

UART transmit/receive control register 2

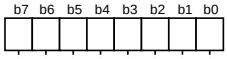


Symbol: UCON — Address: $03B0_{16}$ — When reset: $X0000000_2$

Bit symbol	Bit name	Function (During clock synchronous serial I/O mode)	Function (During UART mode)	R	W
U0IRS	UART0 transmit interrupt cause select bit	0 : Transmit buffer empty (TI = 1) 1 : Transmission completed (TXEPT = 1)	0 : Transmit buffer empty (TI = 1) 1 : Transmission completed (TXEPT = 1)	○	○
U1IRS	UART1 transmit interrupt cause select bit	0 : Transmit buffer empty (TI = 1) 1 : Transmission completed (TXEPT = 1)	0 : Transmit buffer empty (TI = 1) 1 : Transmission completed (TXEPT = 1)	○	○
U0RRM	UART0 continuous receive mode enable bit	0 : Continuous receive mode disabled 1 : Continuous receive mode enable	Must always be “0”	○	○
U1RRM	UART1 continuous receive mode enable bit	0 : Continuous receive mode disabled 1 : Continuous receive mode enabled	Must always be “0”	○	○
CLKMD0	CLK/CLKS select bit 0	Valid when bit 5 = “1” 0 : Clock output to CLK1 1 : Clock output to CLKS1	Invalid	○	○
CLKMD1	CLK/CLKS select bit 1 (Note)	0 : Normal mode (CLK output is CLK1 only) 1 : Transfer clock output from multiple pins function selected	Must always be “0”	○	○
Reserved bit		Must always be set to “0”		○	○
Nothing is assigned. In an attempt to write to this bit, write “0”. The value, if read, turns out to be indeterminate.				—	—

Note: When using multiple pins to output the transfer clock, the following requirements must be met:
• UART1 internal/external clock select bit (bit 3 at address $03A8_{16}$) = “0”.

UART2 special mode register



Symbol: U2SMR — Address: 0377_{16} — When reset: 00_{16}

Bit symbol	Bit name	Function (During clock synchronous serial I/O mode)	Function (During UART mode)	R	W
IICM	I^2C mode select bit	0 : Normal mode 1 : I^2C mode	Must always be “0”	○	○
ABC	Arbitration lost detecting flag control bit	0 : Update per bit 1 : Update per byte	Must always be “0”	○	○
BBS	Bus busy flag	0 : STOP condition detected 1 : START condition detected	Must always be “0”	○ (Note1)	○
LSYN	SCLL sync output enable bit	0 : Disabled 1 : Enabled	Must always be “0”	○	○
ABSCS	Bus collision detect sampling clock select bit	Must always be “0”	0 : Rising edge of transfer clock 1 : Underflow signal of timer A0	○	○
ACSE	Auto clear function select bit of transmit enable bit	Must always be “0”	0 : No auto clear function 1 : Auto clear at occurrence of bus collision	○	○
SSS	Transmit start condition select bit	Must always be “0”	0 : Ordinary 1 : Falling edge of RxD_2	○	○
SDDS	SDA digital delay select bit (Note 2, Note 3)	0 : Analog delay output is selected 1 : Digital delay output is selected (must always be “0” when not using I^2C mode)	Must always be “0”	○	○

Note 1: Nothing but “0” may be written.
Note 2: When not in I^2C mode, do not set this bit by writing a “1”. During normal mode, fix it to “0”. When this bit = “0”, UART2 special mode register 3 (U2SMR3 at address 0375_{16}) bits 7 to 5 (DL2 to DL0 = SDA digital delay setup bits) are initialized to “000”, with the analog delay circuit selected. Also, when SDDS = “0”, the U2SMR3 register cannot be read or written to.
Note 3: When analog delay is selected, only the analog delay value is effective; when digital delay is selected, only the digital delay value is effective.

Figure 2.4.6. Serial I/O-related registers (5)

2.4.2 Operation of Serial I/O (transmission in clock-synchronous serial I/O mode)

In transmitting data in clock-synchronous serial I/O mode, choose functions from those listed in Table 2.4.1. Operations of the circled items are described below. Figure 2.4.7 shows the operation timing, and Figures 2.4.8 and 2.4.9 show the set-up procedures.

Table 2.4.1. Choosed functions

Item	Set-up		Item	Set-up	
Transfer clock source	O	Internal clock (f_1 / f_8 / f_{32})	Transmission interrupt factor	O	Transmission buffer empty
		External clock (CLKi pin)			Transmission complete
$\overline{CTS}$ function	O	$\overline{CTS}$ function enabled	Output transfer clock to multiple pins (Note 1)	O	Not selected
		$\overline{CTS}$ function disabled			Selected
CLK polarity	O	Output transmission data at the falling edge of the transfer clock	Data logic select function (Note 2)	O	No reverse
					Reverse
		Output transmission data at the rising edge of the transfer clock	TxD, RxD I/O polarity reverse bit (Note 2)	O	No reverse
					Reverse
Transfer clock	O	LSB first			
		MSB first			

Note 1: This can be selected only when UART1 is used in combination with the internal clock. When this function is selected, UART1 $\overline{CTS}$/$\overline{RTS}$ function can not be utilized. Set the UART1 $\overline{CTS}$/$\overline{RTS}$ disable bit to "1".

Note 2: UART2 only.

Operation

(1) Setting the transmit enable bit to "1" and writing transmission data to the UARTi transmit buffer register makes data transmissible status ready.

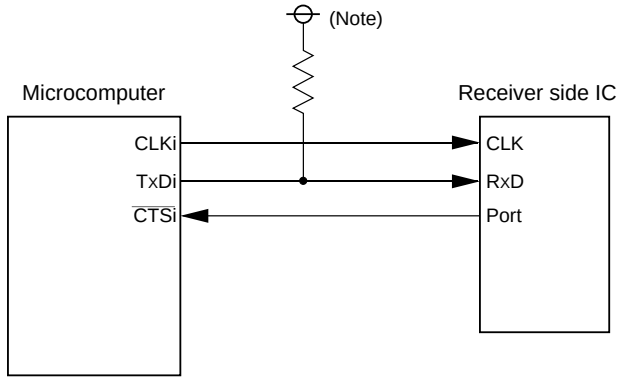
(2) When input to the $\overline{CTS}$i pin goes to "L" level, transmission starts (the $\overline{CTS}$i pin must be controlled on the reception side).

(3) In synchronization with the first falling edge of the transfer clock, transmission data held in the UARTi transmit buffer register is transmitted to the UARTi transmit register. At this time, the UARTi transmit interrupt request bit goes to "1". Also, the first bit of the transmission data is transmitted from the TxDi pin. Then the data is transmitted bit by bit from the lower order in synchronization with the falling edges.

(4) When transmission of 1-byte data is completed, the transmit register empty flag goes to "1", which indicates that transmission is completed. The transfer clock stops at "H" level.

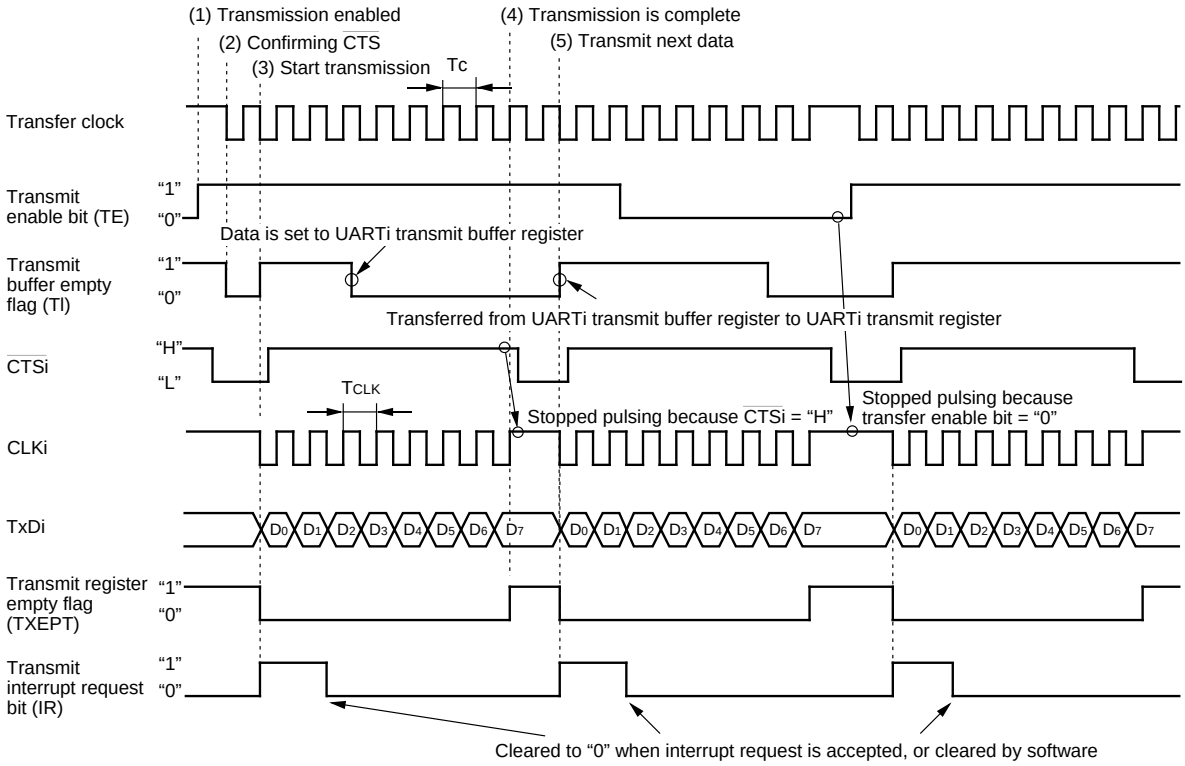
(5) If the next transmission data is set in the UARTi transmit buffer register while transmission is in progress (before the eighth bit has been transmitted), the data is transmitted in succession.

Example of wiring



Note : Since TxD2 pin is N-channel open drain, this pin needs pull-up resistor.

Example of operation



Shown in ( ) are bit symbols.
The above timing applies to the following settings:
- Internal clock is selected.
- CTS function is selected.
- CLK polarity select bit = "0".
- Transmit interrupt cause select bit = "0".

Tc = TCLK = 2(n + 1) / fi
fi: frequency of BRGi count source (f1, f8, f32)
n: value set to BRGi

Figure 2.4.7. Operation timing of transmission in clock-synchronous serial I/O mode

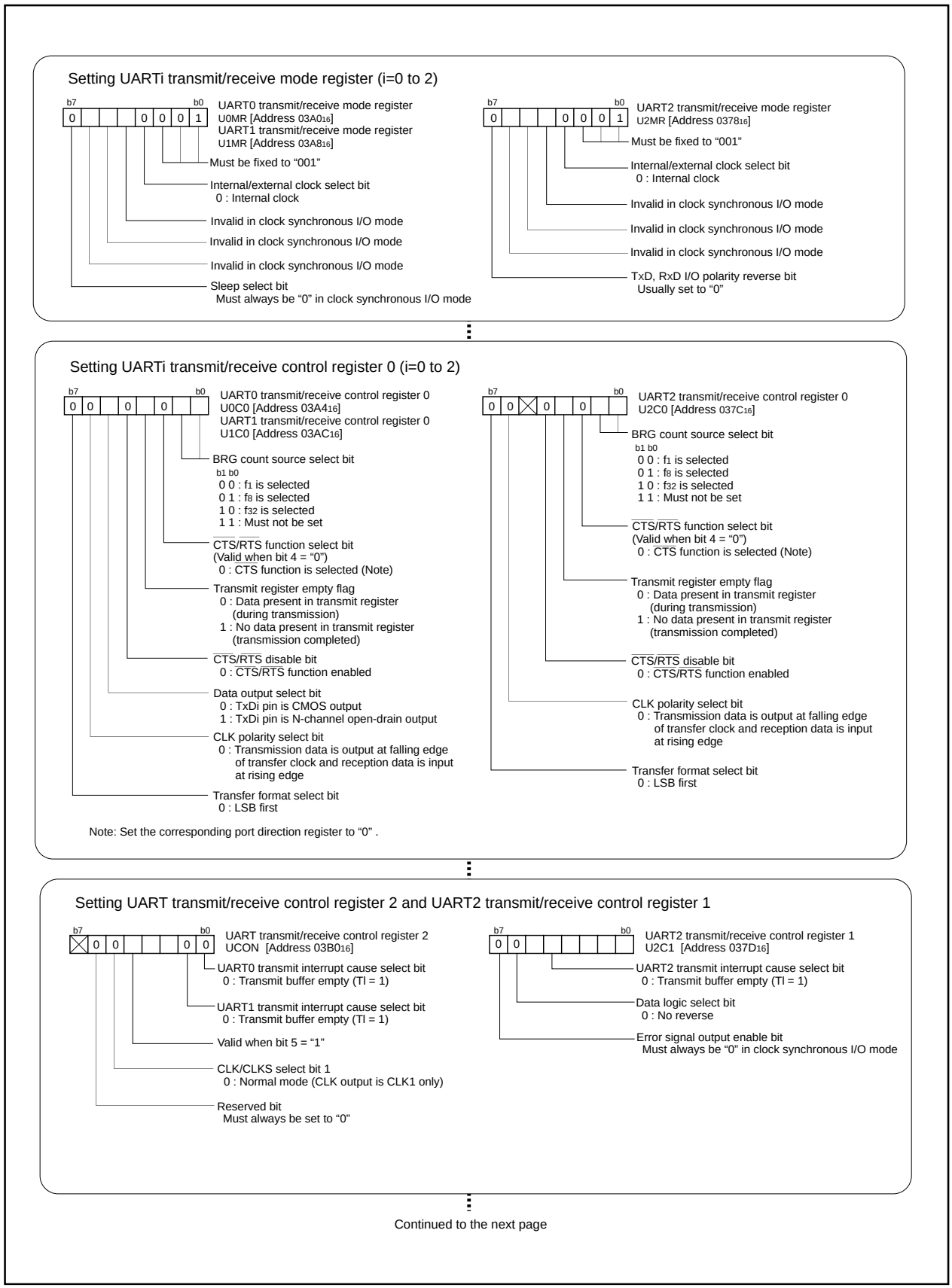


Figure 2.4.8. Set-up procedure of transmission in clock-synchronous serial I/O mode (1)

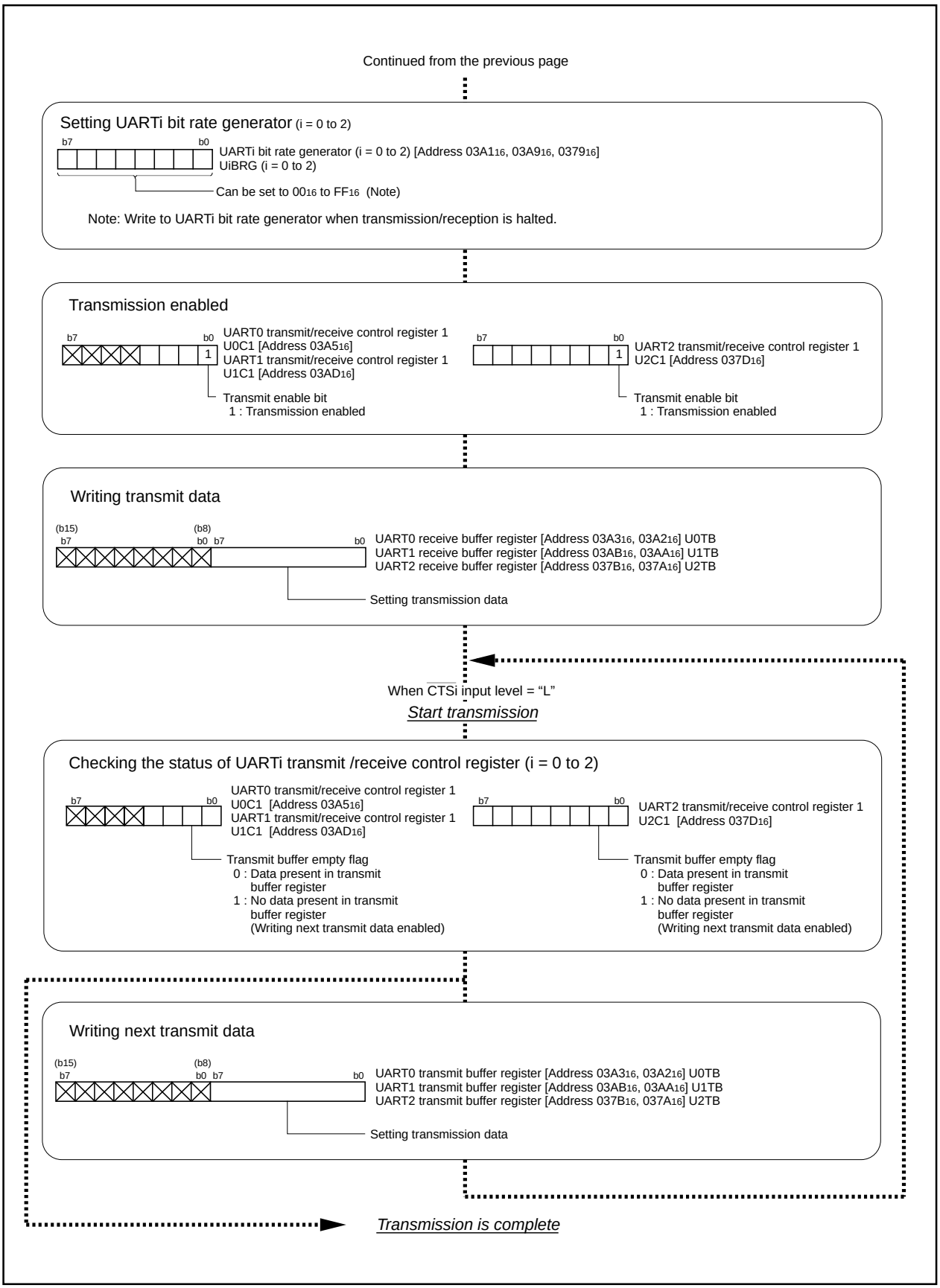


Figure 2.4.9. Set-up procedure of transmission in clock-synchronous serial I/O mode (2)

2.4.3 Operation of the Serial I/O (transmission in clock-synchronous serial I/O mode, transfer clock output from multiple pins function selected)

In transmitting data in clock-synchronous serial I/O mode, choose functions from those listed in Table 2.4.2. Operations of the circled items are described below. Figure 2.4.10 shows the operation timing, and Figures 2.4.11 and 2.4.12 show the set-up procedures.

Table 2.4.2. Choosed functions

Item	Set-up		Item	Set-up	
Transfer clock source	O	Internal clock (f_1 / f_8 / f_{32})	Transmission interrupt factor		Transmission buffer empty
		External clock (CLKi pin)		O	Transmission complete
$\overline{CTS}$ function		$\overline{CTS}$ function enabled	Output transfer clock to multiple pins (Note 1)		Not selected
	O	$\overline{CTS}$ function disabled		O	Selected
CLK polarity	O	Output transmission data at the falling edge of the transfer clock	Data logic select function (Note 2)	O	No reverse
					Reverse
		Output transmission data at the rising edge of the transfer clock	TxD, RxD I/O polarity reverse bit (Note 2)	O	No reverse
					Reverse
Transfer clock	O	LSB first			
		MSB first			

Note 1: This can be selected only when UART1 is used in combination with the internal clock. When this function is selected, UART1 $\overline{CTS}$/$\overline{RTS}$ function can not be utilized. Set the UART1 $\overline{CTS}$/$\overline{RTS}$ disable bit to "1".

Note 2: UART2 only.

Operation

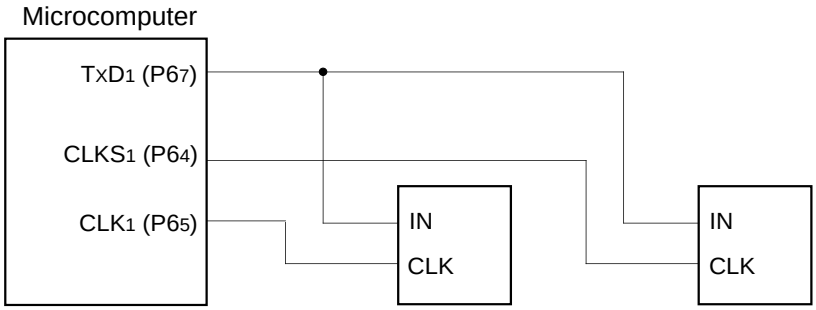
(1) Setting the transmit enable bit to "1" makes data transmissible status ready.

(2) When transmission data is written to the UART1 transmit buffer register, transmission data held in the UART1 transmit buffer register is transmitted to the UART1 transmit register in synchronization with the first falling edge of the transfer clock. At this time, the first bit of the transmission data is transmitted from the TxD1 pin. Then the data is transmitted bit by bit from the lower order in synchronization with the falling edges of the transfer clock.

(3) When transmission of 1-byte data is completed, the transmit register empty flag goes to "1", which indicates that the transmission is completed. The transfer clock stops at "H" level. At this time, the UART1 transmit interrupt request bit goes to "1".

(4) Setting CLK/CLKS select bit 1 to "1" and setting CLK/CLKS select bit 0 to "1" causes the CLKS1 pin to go to the transfer clock output pin. Change the transfer clock output pin when transmission is halted.

Example of wiring



Note: This applies when performing only transmission with an internal clock selected in the clock synchronous serial I/O mode.

Example of operation

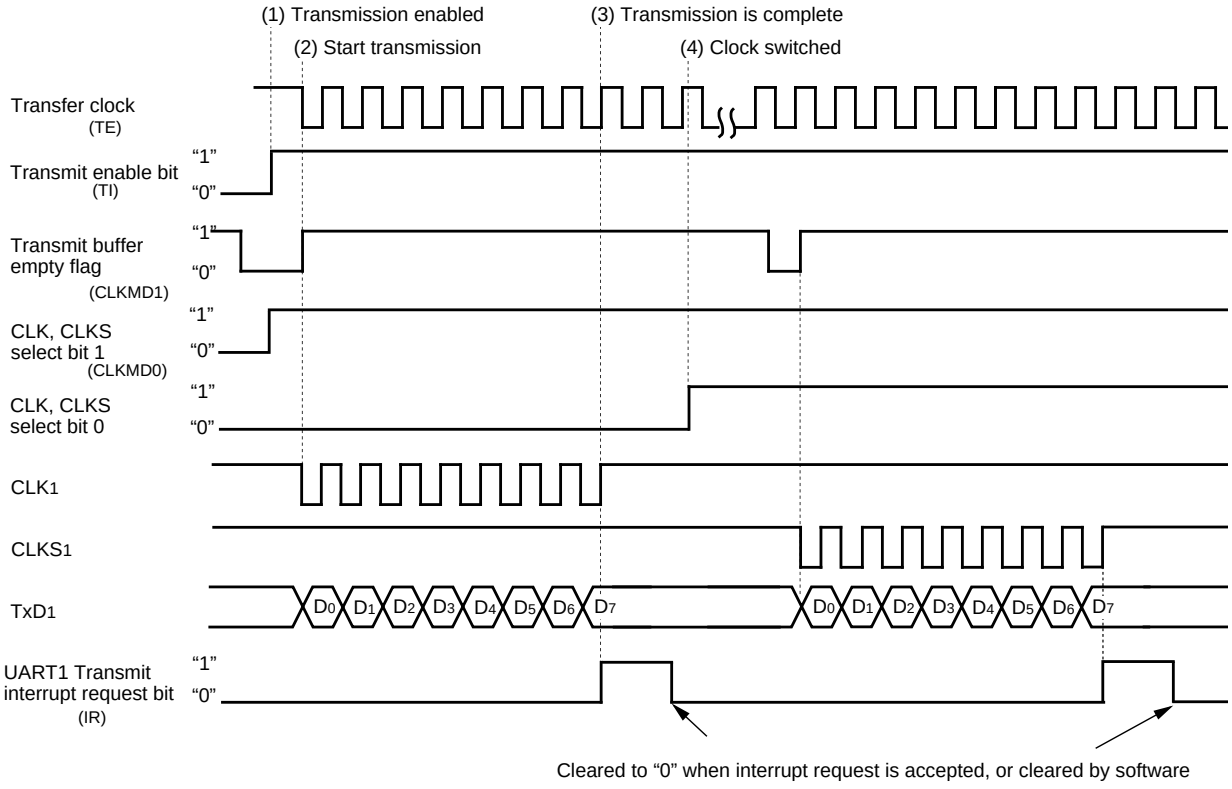


Figure 2.4.10. Operation timing of transmission in clock-synchronous serial I/O mode, transfer clock output from multiple pins function selected

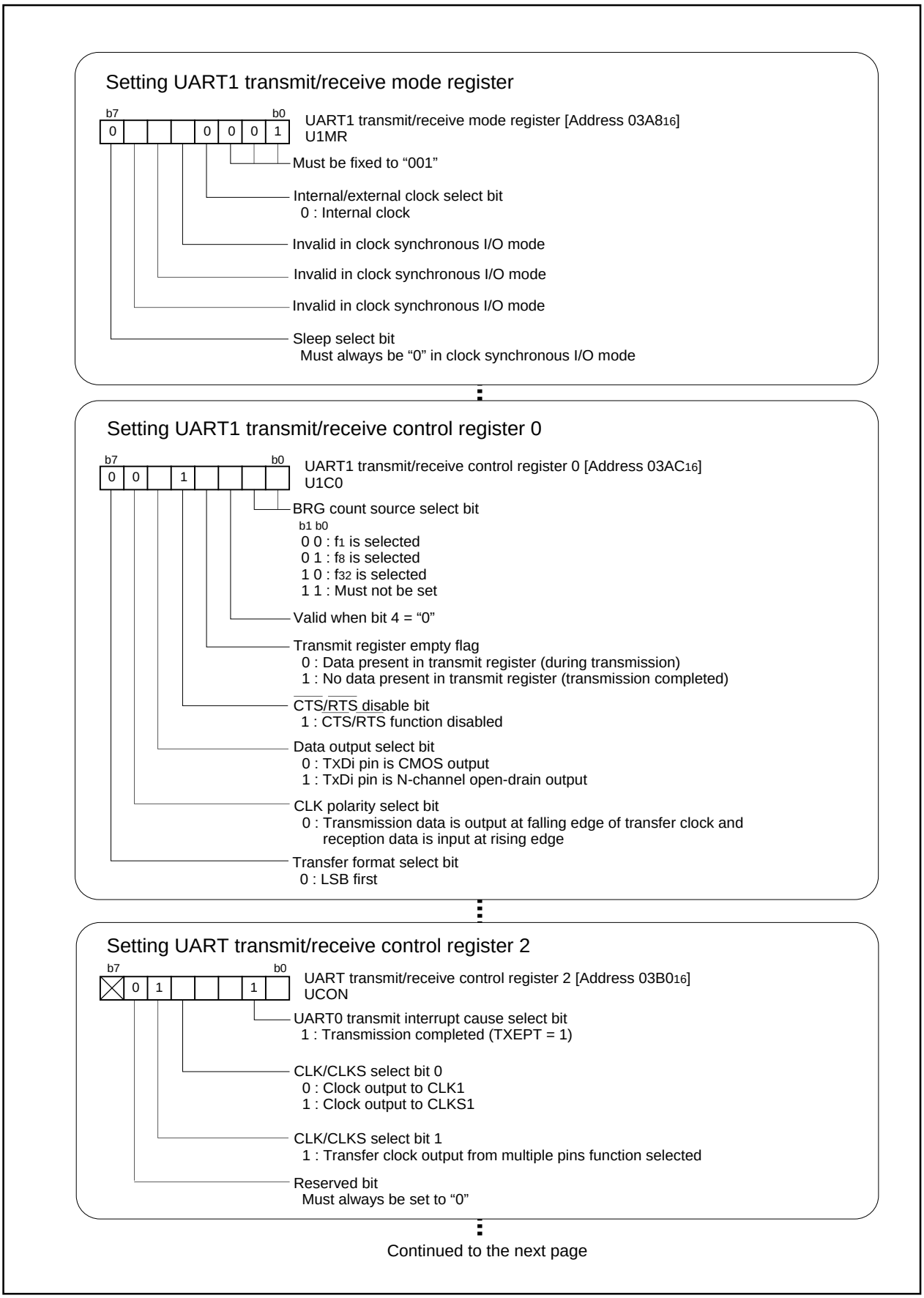


Figure 2.4.11. Set-up procedure of transmission in clock-synchronous serial I/O mode, transfer clock output from multiple pins function selected (1)

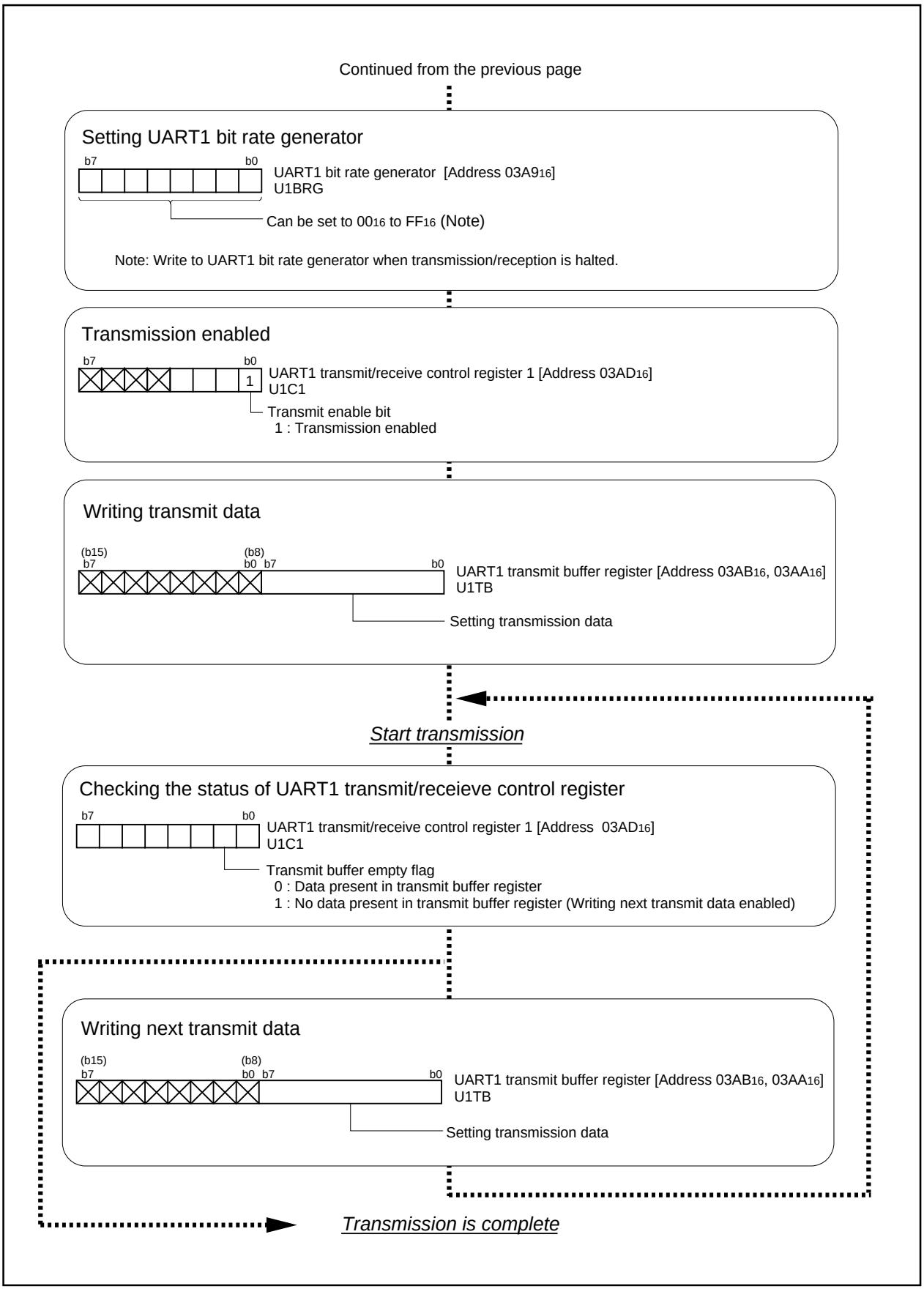


Figure 2.4.12. Set-up procedure of transmission in clock-synchronous serial I/O mode, transfer clock output from multiple pins function selected (2)

2.4.4 Operation of Serial I/O (reception in clock-synchronous serial I/O mode)

In receiving data in clock-synchronous serial I/O mode, choose functions from those listed in Table 2.4.3. Operations of the circled items are described below. Figure 2.4.13 shows the operation timing, and Figures 2.4.14 and 2.4.15 show the set-up procedures.

Table 2.4.3. Choosed functions

Item	Set-up		Item	Set-up	
Transfer clock source		Internal clock (f_1 / f_8 / f_{32})	Continuous receive mode	**O**	Disabled
	O	External clock (CLKi pin)			Enabled
$\overline{RTS}$ function	**O**	$\overline{RTS}$ function enabled	Output transfer clock to multiple pins (Note 1)	**O**	Not selected
		$\overline{RTS}$ function disabled			Selected
CLK polarity	**O**	Input reception data at the rising edge of the transfer clock	Data logic select function (Note 2)	**O**	No reverse
					Reverse
		Input reception data at the falling edge of the transfer clock	TxD, RxD I/O polarity reverse bit (Note 2)	**O**	No reverse
					Reverse
Transfer clock	**O**	LSB first			
		MSB first			

Note 1: This can be selected only when UART1 is used in combination with the internal clock. When this function is selected, UART1 $\overline{CTS}/\overline{RTS}$ function can not be utilized. Set the UART1 $\overline{CTS}/\overline{RTS}$ disable bit to "1".

Note 2: UART2 only.

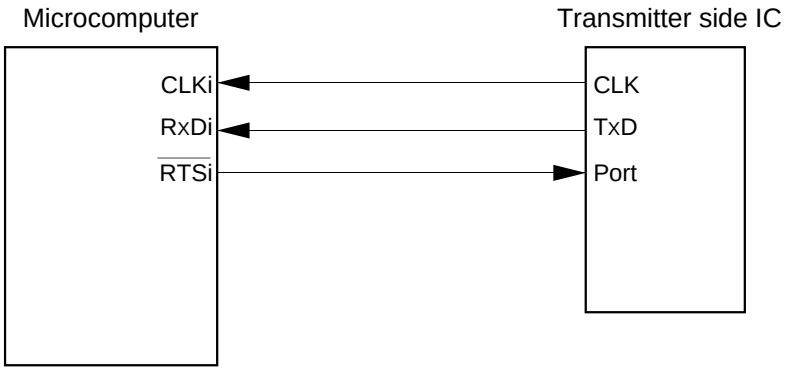
Operation (1) Writing dummy data to the UARTi transmit buffer register, setting the receive enable bit to "1", and the transmit enable bit to "1", makes the data receivable status ready. At this time, the output from the $\overline{RTSi}$ pin goes to "L" level, which informs the transmission side that the data receivable status is ready (output the transfer clock from the IC on the transmission side after checking that the $\overline{RTS}$ output has gone to "L" level).

(2) In synchronization with the first rising edge of the transfer clock, the input signal to the RxDi pin is stored in the highest bit of the UARTi receive register. Then, data is taken in by shifting right the content of the UARTi reception data in synchronization with the rising edges of the transfer clock.

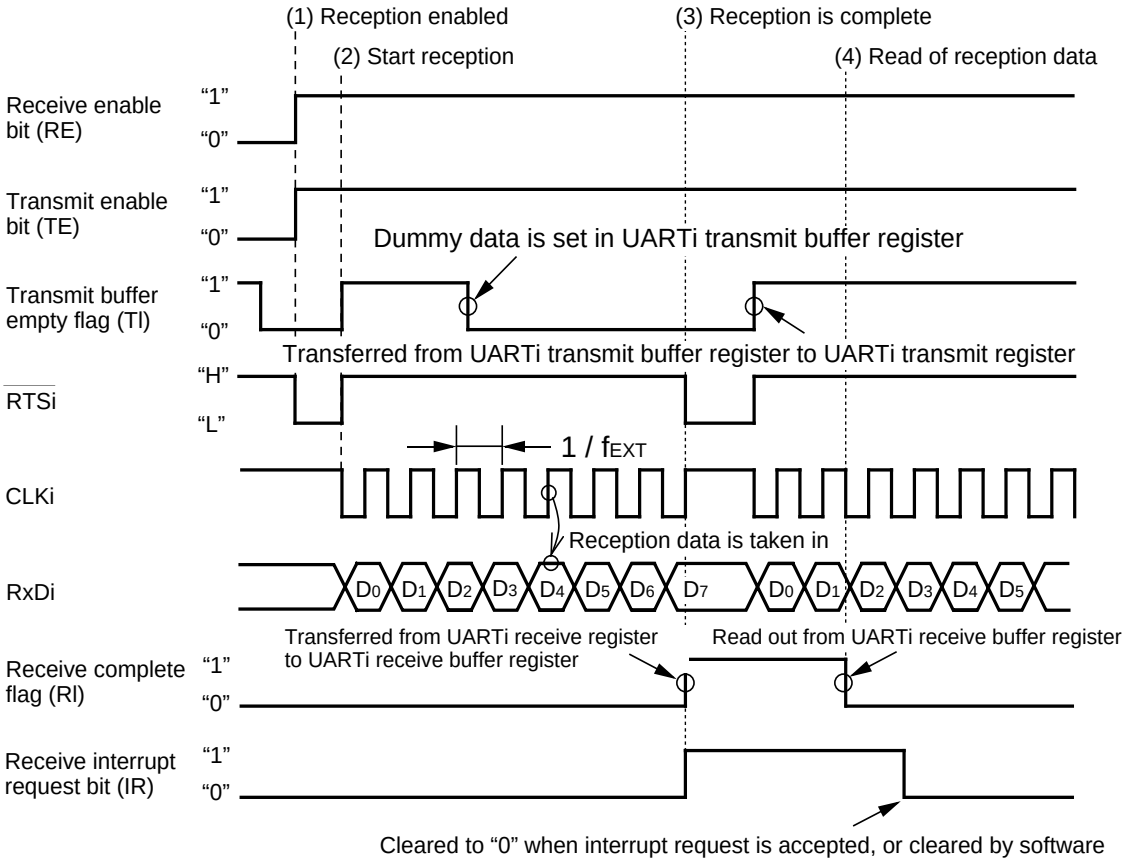
(3) When 1-byte data lines up in the UARTi receive register, the content of the UARTi receive register is transmitted to the UARTi receive buffer register. The transfer clock stops at "H" level. At this time, the receive complete flag and the UARTi receive interrupt request bit goes to "1".

(4) The receive complete flag goes to "0" when the lower-order byte of the UARTi buffer register is read.

Example of wiring



Example of operation



Shown in ( ) are bit symbols.

The above timing applies to the following settings:
- External clock is selected.
- RTS function is selected.
- CLK polarity select bit = "0".

fEXT: frequency of external clock

Make sure that the following conditions are met when the CLKi pin input ="H" before data reception
- Transmit enable bit → "1"
- Receive enable bit → "1"
- Dummy data write to UARTi transmit buffer register

Figure 2.4.13. Operation timing of reception in clock-synchronous serial I/O mode

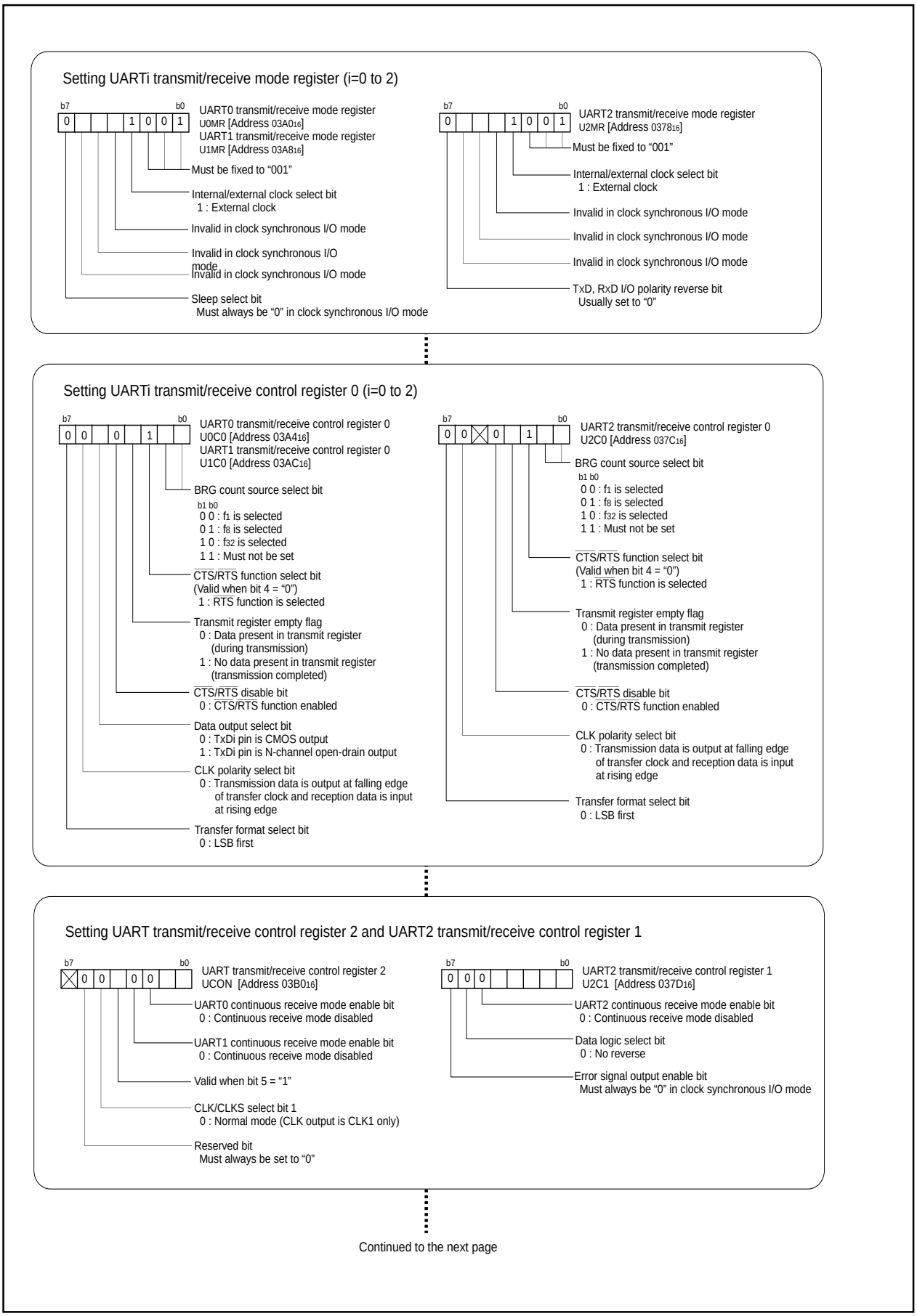


Figure 2.4.14. Set-up procedure of reception in clock-synchronous serial I/O mode (1)

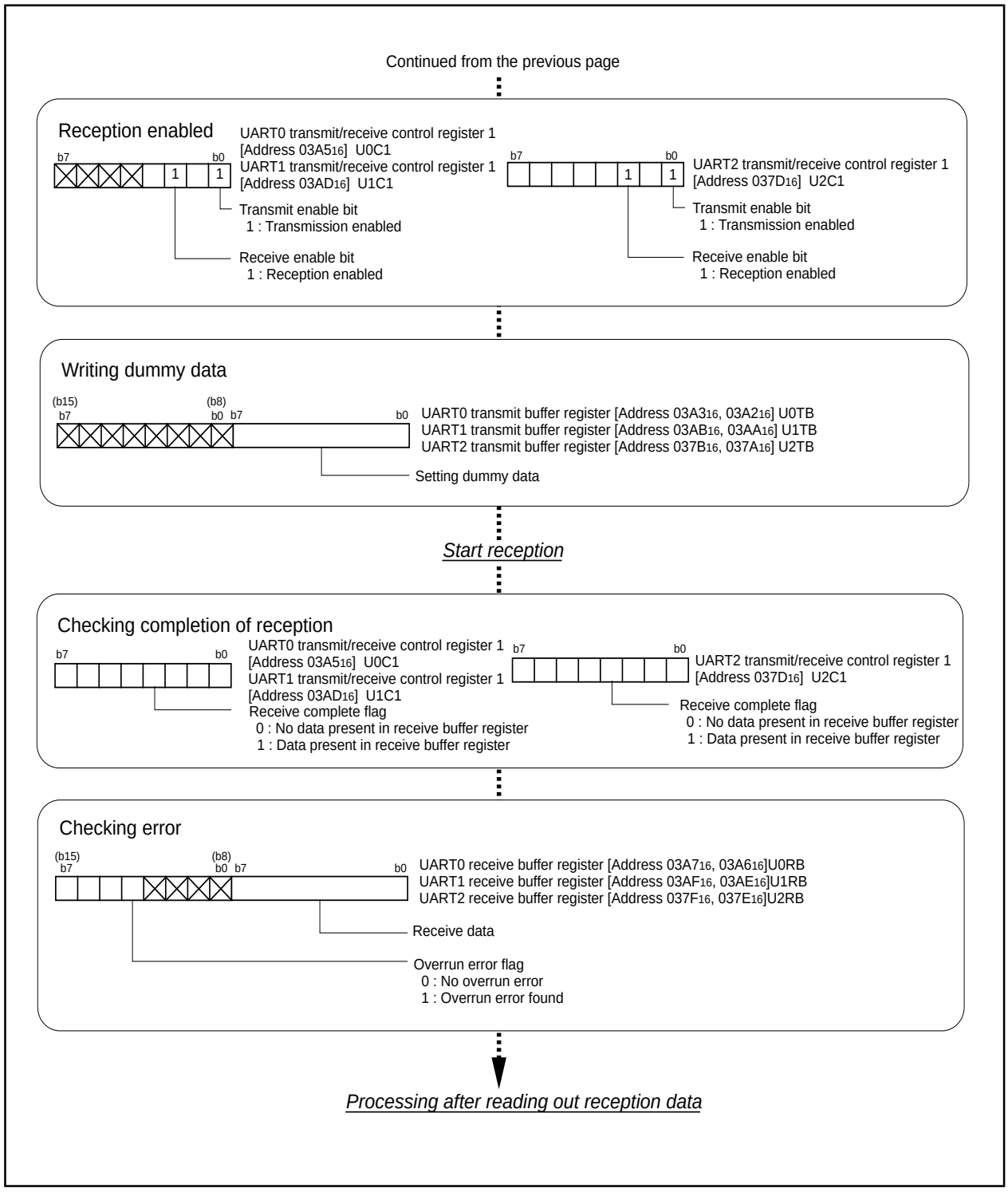


Figure 2.4.15. Set-up procedure of reception in clock-synchronous serial I/O mode (2)

2.4.5 Precautions for Serial I/O (in clock-synchronous serial I/O)

Transmission/reception

(1) With an external clock selected, and choosing the $\overline{RTS}$ function, the output level of the $\overline{RTSi}$ pin goes to “L” when the data-receivable status becomes ready, which informs the transmission side that the reception has become ready. The output level of the $\overline{RTSi}$ pin goes to “H” when reception starts. So if the $\overline{RTSi}$ pin is connected to the $\overline{CTSi}$ pin on the transmission side, the circuit can transmission and reception data with consistent timing. With the internal clock, the $\overline{RTS}$ function has no effect. Figure 2.4.16 shows an example of wiring.

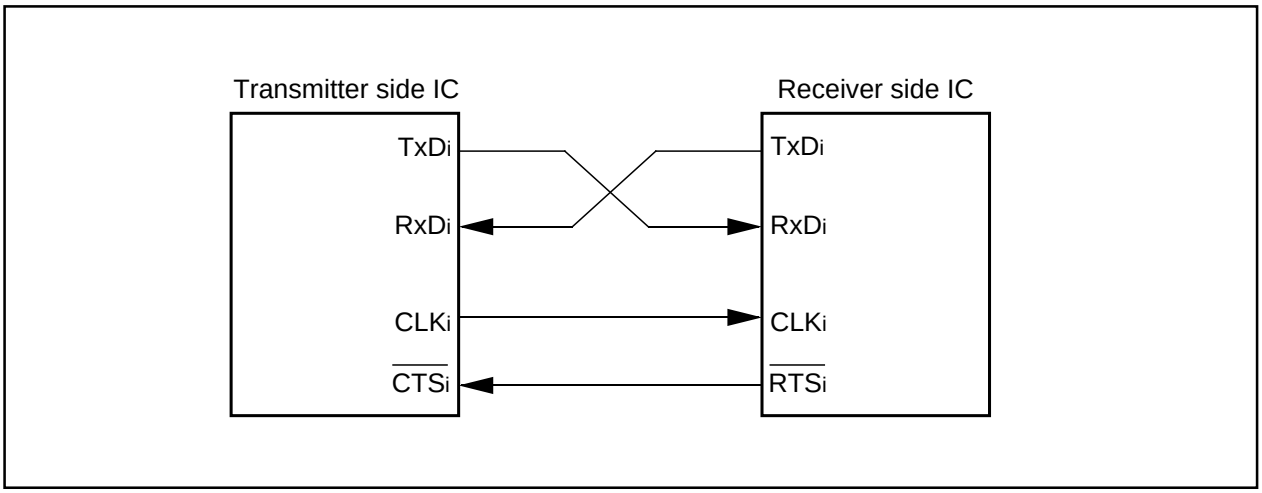


Figure 2.4.16. Example of wiring

Transmission

(1) With an external clock selected, perform the following set-up procedure with the CLKi pin input level = “H” if the CLK polarity select bit = “0” or with the CLKi pin input level = “L” if the CLK polarity select bit = “1”:
   1. Set the transmit enable bit (to “1”)
   2. Write transmission data to the UARTi transmit buffer register
   3. “L” level input to the $\overline{\text{CTSi}}$ pin (when the $\overline{\text{CTS}}$ function is selected)

Reception (1) In operating the clock-synchronous serial I/O, operating a transmitter generates a shift clock. Fix settings for transmission even when using the device only for reception. Dummy data is output to the outside from the TxDi pin (transmission pin) when receiving data.

(2) With the internal clock selected, setting the transmit enable bit to “1” (transmission-enabled status) and setting dummy data in the UARTi transmission buffer register generates a shift clock.
With the external clock selected, a shift clock is generated when the transmit enable bit is set to “1”, dummy data is set in the UARTi transmit buffer register, and the external clock is input to the CLKi pin.

(3) In receiving data in succession, an overrun error occurs when the next reception data is made ready in the UARTi receive register with the receive complete flag set to “1” (before the content of the UARTi receive buffer register is read), and overrun error flag is set to “1”. In this instance, the next data is written to the UARTi receive buffer register, so handle with this problem by writing programs on transmission side and reception side so that the previous data is transmitted again.
If an overrun error occurs, the UARTi receive interrupt request bit does not change.

(4) To receive data in succession, set dummy data in the lower-order byte of the UARTi transmit buffer register every time reception is made.

(5) With an external clock selected, perform the following set-up procedure with the CLKi pin input level = “H” if the CLK polarity select bit = “0” or with the CLKi pin input level = “L” if the CLK polarity select bit = “1”:
   1. Set receive enable bit (to “1”)
   2. Set transmit enable bit (to “1”)
   3. Write dummy data to the UARTi transmit buffer register

(6) Output from the $\overline{\text{RTS}}$ pin goes to “L” level as soon as the receive enable bit is set to “1”. This is not related to the content of the transmit buffer empty flag or the content of the transmit enable bit.
Output from the $\overline{\text{RTS}}$ pin goes to “H” level when reception starts, and goes to “L” level when reception is completed. This is not related to the content of the transmit buffer empty flag or the content of the receive complete flag.

2.5 Clock-Asynchronous Serial I/O (UART) Usage

2.5.1 Overview of the clock-asynchronous serial I/O usage

UART handles communications by means of character-by-character synchronization. The transmission side and the reception side are independent of each other, so full-duplex communication is possible. The following is an overview of the clock-asynchronous serial I/O usage.

(1) Transmission/reception format

Figure 2.5.1 shows the transmission/reception format, and Table 2.5.1 shows the names and functions of transmission data.

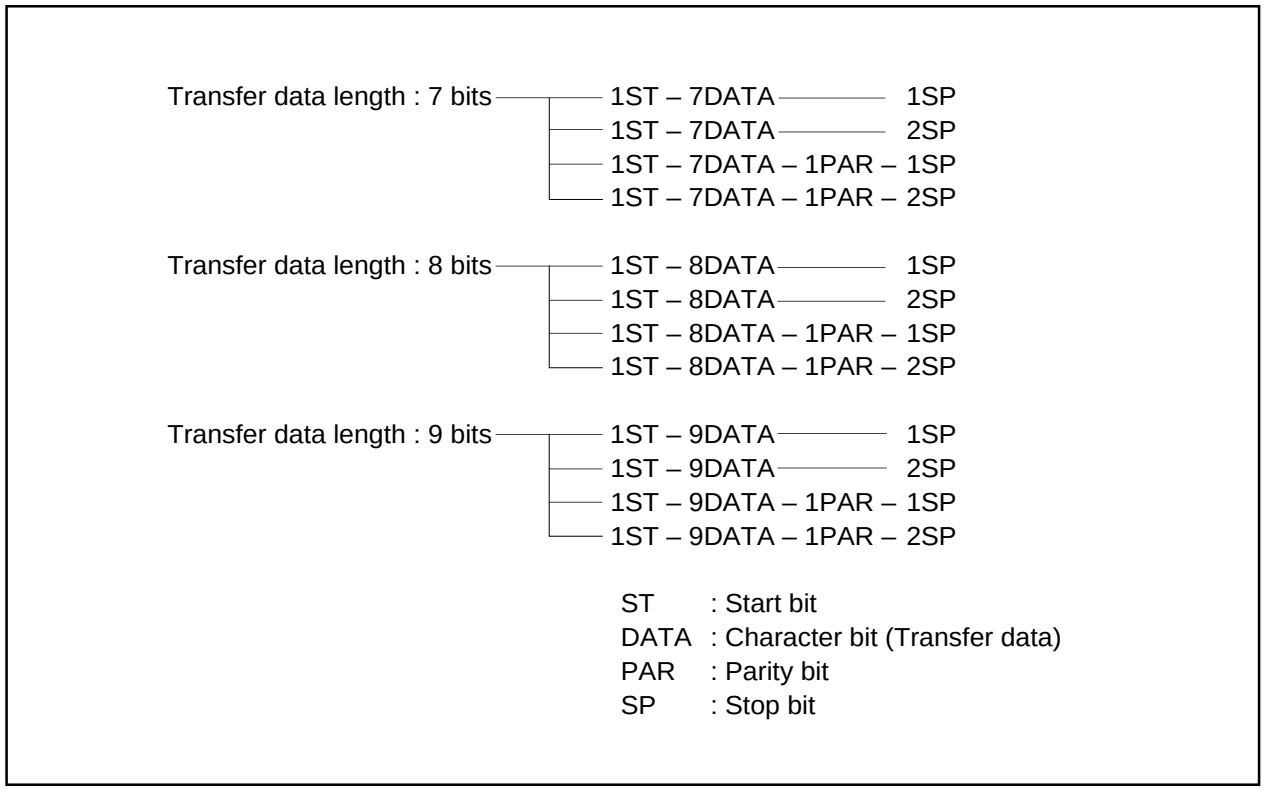


Figure 2.5.1. Transmission/reception format

Table 2.5.1. Transmission data names and functions

Name	Function
ST (start bit)	A 1-bit “L” signal to be added immediately before character bits. This bit signals the start of data transmission.
DATA (character bits)	Transmission data set in the UARTi transmit buffer register.
PAR (parity bit)	A signal to be added immediately after character bits so as to increase data reliability. The level of this signal so varies that the total number of 1's in character bits and this bit always becomes even or odd depending on which parity is chosen, even or odd.
SP (stop bit)	Either 1-bit or 2-bit “H” signal to be added immediately after character bits (after the parity bit if parity is checked). This / they signals the end of data transmission.

(2) Transfer rate

The divide-by-16 frequency, resulting from division in the bit rate generator (BRG), becomes the transfer rate. The count source for the transfer rate register can be selected from f_1, f_8, f_{32}, and the input from the CLK pin. Clocks f_1, f_8, f_{32} are derived by dividing the CPU's main clock by 1, 8, and 32 respectively.

Table 2.5.2. Example of baud rate setting

Baud rate (bps)	BRG's count source	System clock : 16MHz		System clock : 7.3728MHz	
		BRG's set value : n	Actual time (bps)	BRG's set value : n	Actual time (bps)
600	f_8	207 (CF_{16})	601	95 ($5F_{16}$)	600
1200	f_8	103 (67_{16})	1202	47 ($2F_{16}$)	1200
2400	f_8	51 (33_{16})	2404	23 (17_{16})	2400
4800	f_1	207 (CF_{16})	4808	95 ($5F_{16}$)	4800
9600	f_1	103 (67_{16})	9615	47 ($2F_{16}$)	9600
14400	f_1	68 (44_{16})	14493	31 ($1F_{16}$)	14400
19200	f_1	51 (33_{16})	19231	23 (17_{16})	19200
28800	f_1	34 (22_{16})	28571	15 (F_{16})	28800
31250	f_1	33 (21_{16})	31250		

(3) An error detection

In clock-asynchronous serial I/O mode, detect errors are shown in Table 2.5.3.

Table 2.5.3. Error detection

Type of error	Description	When the flag turns on	How to clear the flag
Overrun error	• This error occurs when the next data lines up before the content of the UARTi receive buffer register is read. • The next data is written to the UARTi receive buffer register. • The UARTi receive interrupt request bit does not change.	The error is detected when data is transferred from the UARTi receive register to the UARTi receive buffer register.	• Set the serial I/O mode select bits to "000_2". • Set the receive enable bit to "0".
Framing error	• This error occurs when the stop bit falls short of the set number of stop bits.		• Set the serial I/O mode select bits to "000_2". • Set the receive enable bit to "0". • Read the lower-order byte of the UARTi receive buffer register.
Parity error	• With parity enabled, this error occurs when the total number of 1's in character bits and the parity bit is different from the specified number.		
Error-sum flag	• This flag turns on when any error (overrun, framing, or parity) is detected.		• When all error (overrun, framing, and parity) are removed, the flag is cleared.

(4) How to deal with an error

When receiving data, read an error flag and reception data simultaneously to determine which error has occurred. If the data read is erroneous, initialize the error flag and the UARTi receive buffer register, then receive the data again.

To initialize the UARTi receive buffer register

1. Set the receive enable bit to “0” (disable reception).
2. Set the receive enable bit to “1” again (enable reception).

To transmit data again due to an error on the reception side, set the UARTi transmit buffer register again, then transmit the data again.

To set the UARTi transmit buffer register again

1. Set the serial I/O mode select bits to “000_2” (invalidate serial I/O).
2. Set the serial I/O mode select bits again.
3. Set the transmit enable bit to “1” (enable transmission), then set transmission data in the UARTi transmit buffer register.

(5) Functions selection

In operating UART, the following functions can be used:

(a) $\overline{\text{CTS}}/\overline{\text{RTS}}$ function

$\overline{\text{CTS}}$ function is a function in which an external IC can start transmission/reception by means of inputting an “L” level to the $\overline{\text{CTS}}$ pin. The $\overline{\text{CTS}}$ pin input level is detected when transmission/reception starts, so if the level is gone to“ H” while transmission/reception is in progress, transmission/reception stops at the next data.

$\overline{\text{RTS}}$ function is a function to inform an external IC that $\overline{\text{RTS}}$ pin output level has changed to “L” when reception is ready. $\overline{\text{RTS}}$ regoes to “H” at the falling edge of the transfer clock.

When using clock-asynchronous serial I/O, choose one of three types of $\overline{\text{CTS}}/\overline{\text{RTS}}$ functions.

- $\overline{\text{CTS}}/\overline{\text{RTS}}$ functions disabled — $\overline{\text{CTS}}/\overline{\text{RTS}}$ pin is a programmable I/O port.
- $\overline{\text{CTS}}$ function only enabled — $\overline{\text{CTS}}/\overline{\text{RTS}}$ pin performs the $\overline{\text{CTS}}$ function.
- $\overline{\text{RTS}}$ function only enabled — $\overline{\text{CTS}}/\overline{\text{RTS}}$ pin performs the $\overline{\text{RTS}}$ function.

(b) Sleep mode

Sleep mode is a mode in which data is transferred to a particular microcomputer among those connected by use of clock-asynchronous serial I/O devices.

(c) Data logic select function

This function is to reserve data when writing to transmit buffer register or reading from receive buffer register.

(d) TxD, RxD I/O polarity reverse function

This function receive a polarity of TxD port output level and a polarity of RxD port input level.

(e) Bus collision detection function

This function is to sample the output level of the TxD pin and the input level of the RxD pin at the rising edge of the transfer clock; if their values are different, then an interrupt request occurs.

The following are examples in which functions (a) to (e) are chosen:

• Transmission WITH: $\overline{CTS}$ function, WITHOUT: other functions P2-84
• Reception WITH: $\overline{RTS}$ function, WITHOUT: other functions P2-88

Also, the SIM interface is used by adding some extra settings in UART2's clock-asynchronous serial I/O mode. Direct or inverse format is selected by connecting SIM card.

• Transmission WITH: direct format P2-92
• Reception WITH: direct format P2-96

(6) Input to the serial I/O and the direction register

To input an external signal to the serial I/O, set the direction register of the relevant port to input.

(7) Pins related to the serial I/O

- $\overline{CTS_0}$, $\overline{CTS_1}$, $\overline{CTS_2}$ pins :Input pins for the $\overline{CTS}$ function
- $\overline{RTS_0}$, $\overline{RTS_1}$, $\overline{RTS_2}$ pins :Output pins for the $\overline{RTS}$ function
- CLK_0, CLK_1 pins :Input pins for the transfer clock
- RxD_0, RxD_1, RxD_2 pins :Input pins for data
- TxD_0, TxD_1, TxD_2 pins :Output pins for data

Since TxD_2 pin is N-channel open drain, this pin needs pull-up resistor.

(8) Registers related to the serial I/O

Figure 2.5.2 shows the memory map of serial I/O-related registers, and Figures 2.5.3 to 2.5.7 show UARTi-related registers.

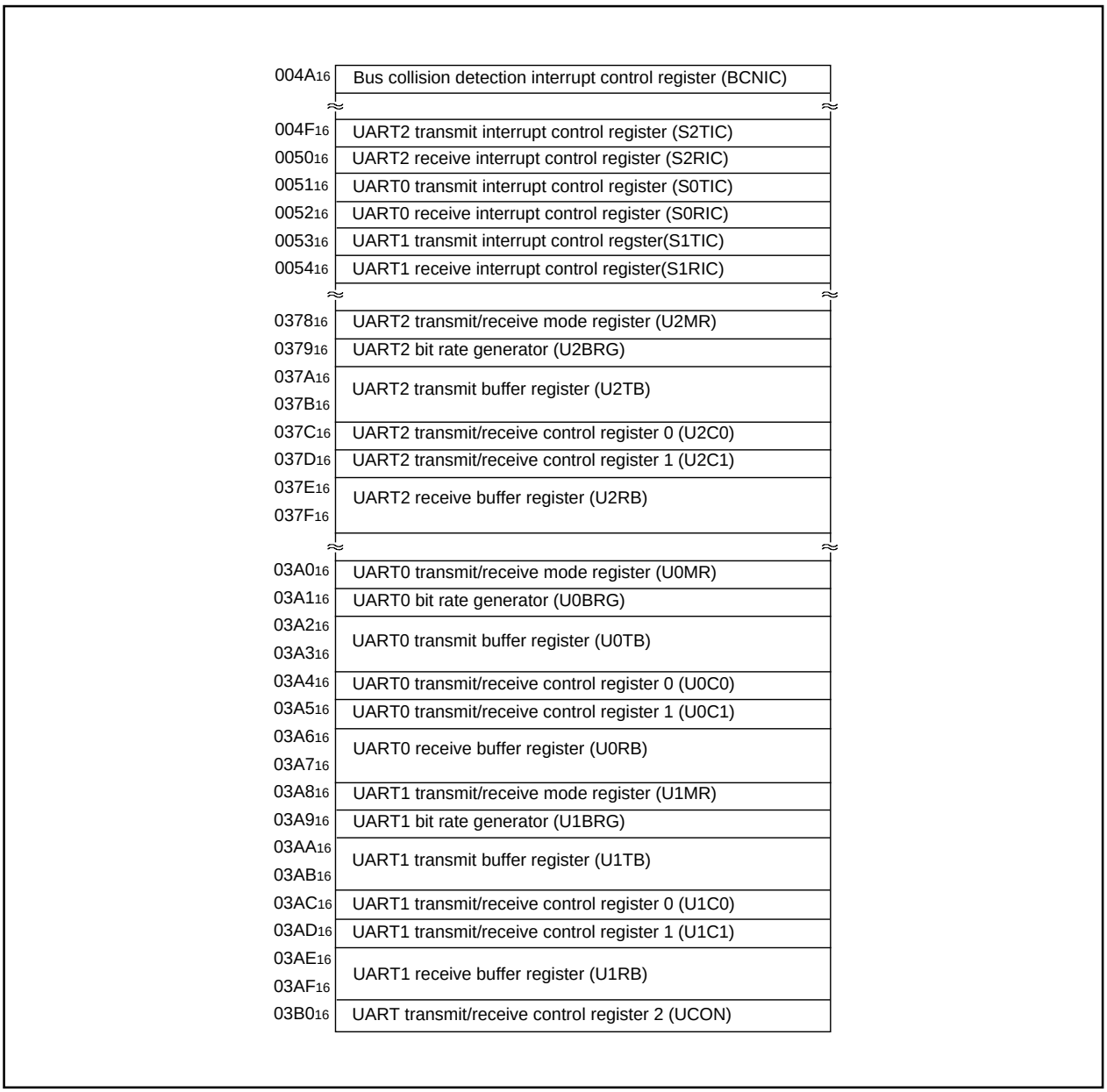
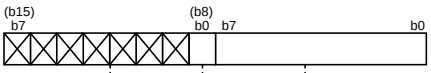


Figure 2.5.2. Memory map of UARTi-related registers

UARTi transmit buffer register (Note)

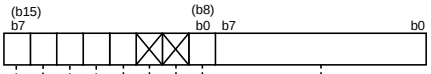


Symbol	Address	When reset
U0TB	$03A3_{16}$, $03A2_{16}$	Indeterminate
U1TB	$03AB_{16}$, $03AA_{16}$	Indeterminate
U2TB	$037B_{16}$, $037A_{16}$	Indeterminate

Function	R	W
Transmit data	×	○
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be indeterminate.	—	—

Note: Use MOV instruction to write to this register.

UARTi receive buffer register

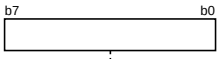


Symbol	Address	When reset
U0RB	$03A7_{16}$, $03A6_{16}$	Indeterminate
U1RB	$03AF_{16}$, $03AE_{16}$	Indeterminate
U2RB	$037F_{16}$, $037E_{16}$	Indeterminate

Bit symbol	Bit name	Function (During clock synchronous serial I/O mode)	Function (During UART mode)	R	W
—	———	Receive data	Receive data	○	×
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be “0”.				—	—
ABT	Arbitration lost detecting flag (Note 2)	0 : Not detected 1 : Detected	Invalid	○	○
OER	Overrun error flag (Note 1)	0 : No overrun error 1 : Overrun error found	0 : No overrun error 1 : Overrun error found	○	×
FER	Framing error flag (Note 1)	Invalid	0 : No framing error 1 : Framing error found	○	×
PER	Parity error flag (Note 1)	Invalid	0 : No parity error 1 : Parity error found	○	×
SUM	Error sum flag (Note 1)	Invalid	0 : No error 1 : Error found	○	×

Note 1: Bits 15 through 12 are set to “0” when the serial I/O mode select bit (bits 2 to 0 at addresses $03A0_{16}$, $03A8_{16}$ and 0378_{16}) are set to “000_2” or the receive enable bit is set to “0”.
(Bit 15 is set to “0” when bits 14 to 12 all are set to “0”.) Bits 14 and 13 are also set to “0” when the lower byte of the UARTi receive buffer register (addresses $03A6_{16}$, $03AE_{16}$ and $037E_{16}$) is read out.
Note 2: Arbitration lost detecting flag is allocated to U2RB and noting but “0” may be written. Nothing is assigned in bit 11 of U0RB and U1RB. When write, set “0”. The value, if read, turns out to be “0”.

UARTi bit rate generator (Note 1, 2)



Symbol	Address	When reset
U0BRG	$03A1_{16}$	Indeterminate
U1BRG	$03A9_{16}$	Indeterminate
U2BRG	0379_{16}	Indeterminate

Function	Values that can be set	R	W
Assuming that set value = n, BRGi divides the count source by n + 1	00_{16} to FF_{16}	×	○

Note 1: Write a value to this register while transmit/receive halts.
Note 2: Use MOV instruction to write to this register.

Figure 2.5.3. UARTi-related registers (1)

UARTi transmit/receive mode register

b7	b6	b5	b4	b3	b2	b1	b0

Symbol: UiMR(i=0,1) — Address: $03A0_{16}$, $03A8_{16}$ — When reset: 00_{16}

Bit symbol	Bit name	Function (During clock synchronous serial I/O mode)	Function (During UART mode)	R	W
SMD0	Serial I/O mode select bit	Must be fixed to 001 b2 b1 b0 0 0 0 : Serial I/O invalid 0 1 0 : Must not be set. 0 1 1 : Must not be set. 1 1 1 : Must not be set.	b2 b1 b0 1 0 0 : Transfer data 7 bits long 1 0 1 : Transfer data 8 bits long 1 1 0 : Transfer data 9 bits long 0 0 0 : Serial I/O invalid 0 1 0 : Must not be set. 0 1 1 : Must not be set. 1 1 1 : Must not be set.	○	○
SMD1				○	○
SMD2				○	○
CKDIR	Internal/external clock select bit	0 : Internal clock 1 : External clock (Note)	0 : Internal clock 1 : External clock (Note)	○	○
STPS	Stop bit length select bit	Invalid	0 : One stop bit 1 : Two stop bits	○	○
PRY	Odd/even parity select bit	Invalid	Valid when bit 6 = “1” 0 : Odd parity 1 : Even parity	○	○
PRYE	Parity enable bit	Invalid	0 : Parity disabled 1 : Parity enabled	○	○
SLEP	Sleep select bit	Must always be “0”	0 : Sleep mode deselected 1 : Sleep mode selected	○	○

Note : Set the corresponding port direction register to “0”.

UART2 transmit/receive mode register

b7	b6	b5	b4	b3	b2	b1	b0

Symbol: U2MR — Address: 0378_{16} — When reset: 00_{16}

Bit symbol	Bit name	Function (During clock synchronous serial I/O mode)	Function (During UART mode)	R	W
SMD0	Serial I/O mode select bit	Must be fixed to 001 b2 b1 b0 0 0 0 : Serial I/O invalid 0 1 0 : (Note 1) 0 1 1 : Must not be set. 1 1 1 : Must not be set.	b2 b1 b0 1 0 0 : Transfer data 7 bits long 1 0 1 : Transfer data 8 bits long 1 1 0 : Transfer data 9 bits long 0 0 0 : Serial I/O invalid 0 1 0 : Must not be set. 0 1 1 : Must not be set. 1 1 1 : Must not be set.	○	○
SMD1				○	○
SMD2				○	○
CKDIR	Internal/external clock select bit	0 : Internal clock 1 : External clock (Note 2)	Must always be “0”	○	○
STPS	Stop bit length select bit	Invalid	0 : One stop bit 1 : Two stop bits	○	○
PRY	Odd/even parity select bit	Invalid	Valid when bit 6 = “1” 0 : Odd parity 1 : Even parity	○	○
PRYE	Parity enable bit	Invalid	0 : Parity disabled 1 : Parity enabled	○	○
IOPOL	TxD, RxD I/O polarity reverse bit	0 : No reverse 1 : Reverse Usually set to “0”	0 : No reverse 1 : Reverse Usually set to “0”	○	○

Note 1: Bit 2 to bit 0 are set to “010_2” when I^2C mode is used.
Note 2: Set the corresponding port direction register to “0”.

Figure 2.5.4. UARTi-related registers (2)

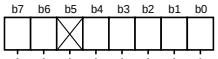
UARTi transmit/receive control register 0

Symbol: UiC0(i=0,1) — Address: 03A4$_{16}$, 03AC$_{16}$ — When reset: 08$_{16}$

Bit symbol	Bit name	Function (During clock synchronous serial I/O mode)	Function (During UART mode)	R	W
CLK0	BRG count source select bit	b1 b0 0 0 : f_1 is selected 0 1 : f_8 is selected 1 0 : f_{32} is selected 1 1 : Must not be set.	b1 b0 0 0 : f_1 is selected 0 1 : f_8 is selected 1 0 : f_{32} is selected 1 1 : Must not be set.	○	○
CLK1				○	○
CRS	$\overline{CTS}/\overline{RTS}$ function select bit	Valid when bit 4 = "0" 0 : $\overline{CTS}$ function is selected (Note 1) 1 : $\overline{RTS}$ function is selected (Note 2)	Valid when bit 4 = "0" 0 : $\overline{CTS}$ function is selected (Note 1) 1 : $\overline{RTS}$ function is selected (Note 2)	○	○
TXEPT	Transmit register empty flag	0 : Data present in transmit register (during transmission) 1 : No data present in transmit register (transmission completed)	0 : Data present in transmit register (during transmission) 1 : No data present in transmit register (transmission completed)	○	×
CRD	$\overline{CTS}/\overline{RTS}$ disable bit	0 : $\overline{CTS}/\overline{RTS}$ function enabled 1 : $\overline{CTS}/\overline{RTS}$ function disabled ($P6_0$ and $P6_4$ function as programmable I/O port)	0 : $\overline{CTS}/\overline{RTS}$ function enabled 1 : $\overline{CTS}/\overline{RTS}$ function disabled ($P6_0$ and $P6_4$ function as programmable I/O port)	○	○
NCH	Data output select bit	0 : TxDi pin is CMOS output 1 : TxDi pin is N-channel open-drain output	0: TxDi pin is CMOS output 1: TxDi pin is N-channel open-drain output	○	○
CKPOL	CLK polarity select bit	0 : Transmit data is output at falling edge of transfer clock and receive data is input at rising edge 1 : Transmit data is output at rising edge of transfer clock and receive data is input at falling edge	Must always be "0"	○	○
UFORM	Transfer format select bit	0 : LSB first 1 : MSB first	Must always be "0"	○	○

Note 1: Set the corresponding port direction register to "0".
Note 2: The settings of the corresponding port register and port direction register are invalid.

UART2 transmit/receive control register 0



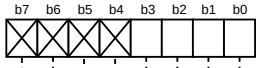
Symbol: U2C0 — Address: 037C$_{16}$ — When reset: 08$_{16}$

Bit symbol	Bit name	Function (During clock synchronous serial I/O mode)	Function (During UART mode)	R	W
CLK0	BRG count source select bit	b1 b0 0 0 : f_1 is selected 0 1 : f_8 is selected 1 0 : f_{32} is selected 1 1 : Must not be set.	b1 b0 0 0 : f_1 is selected 0 1 : f_8 is selected 1 0 : f_{32} is selected 1 1 : Must not be set.	○	○
CLK1				○	○
CRS	$\overline{CTS}/\overline{RTS}$ function select bit	Valid when bit 4 = "0" 0 : $\overline{CTS}$ function is selected (Note 1) 1 : $\overline{RTS}$ function is selected (Note 2)	Valid when bit 4 = "0" 0 : $\overline{CTS}$ function is selected (Note 1) 1 : $\overline{RTS}$ function is selected (Note 2)	○	○
TXEPT	Transmit register empty flag	0 : Data present in transmit register (during transmission) 1 : No data present in transmit register (transmission completed)	0 : Data present in transmit register (during transmission) 1 : No data present in transmit register (transmission completed)	○	×
CRD	$\overline{CTS}/\overline{RTS}$ disable bit	0 : $\overline{CTS}/\overline{RTS}$ function enabled 1 : $\overline{CTS}/\overline{RTS}$ function disabled ($P7_3$ functions programmable I/O port)	0 : $\overline{CTS}/\overline{RTS}$ function enabled 1 : $\overline{CTS}/\overline{RTS}$ function disabled ($P7_3$ functions programmable I/O port)	○	○
Nothing is assigned. In an attempt to write to this bit, write "0". The value, if read, turns out to be "0".				–	–
CKPOL	CLK polarity select bit	0 : Transmit data is output at falling edge of transfer clock and receive data is input at rising edge 1 : Transmit data is output at rising edge of transfer clock and receive data is input at falling edge	Must always be "0"	○	○
UFORM	Transfer format select bit (Note 3)	0 : LSB first 1 : MSB first	0 : LSB first 1 : MSB first	○	○

Note 1: Set the corresponding port direction register to "0".
Note 2: The settings of the corresponding port register and port direction register are invalid.
Note 3: Only clock synchronous serial I/O mode and 8-bit UART mode are valid.

Figure 2.5.5. UARTi-related registers (3)

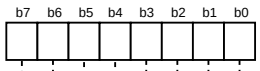
UARTi transmit/receive control register 1



Symbol: UiC1(i=0,1) | Address: 03A5$_{16}$,03AD$_{16}$ | When reset: 02$_{16}$

Bit symbol	Bit name	Function (During clock synchronous serial I/O mode)	Function (During UART mode)	R	W
TE	Transmit enable bit	0 : Transmission disabled 1 : Transmission enabled	0 : Transmission disabled 1 : Transmission enabled	○	○
TI	Transmit buffer empty flag	0 : Data present in transmit buffer register 1 : No data present in transmit buffer register	0 : Data present in transmit buffer register 1 : No data present in transmit buffer register	○	×
RE	Receive enable bit	0 : Reception disabled 1 : Reception enabled	0 : Reception disabled 1 : Reception enabled	○	○
RI	Receive complete flag	0 : No data present in receive buffer register 1 : Data present in receive buffer register	0 : No data present in receive buffer register 1 : Data present in receive buffer register	○	×
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be “0”.				–	–

UART2 transmit/receive control register 1

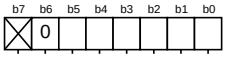


Symbol: U2C1 | Address: 037D$_{16}$ | When reset: 02$_{16}$

Bit symbol	Bit name	Function (During clock synchronous serial I/O mode)	Function (During UART mode)	R	W
TE	Transmit enable bit	0 : Transmission disabled 1 : Transmission enabled	0 : Transmission disabled 1 : Transmission enabled	○	○
TI	Transmit buffer empty flag	0 : Data present in transmit buffer register 1 : No data present in transmit buffer register	0 : Data present in transmit buffer register 1 : No data present in transmit buffer register	○	×
RE	Receive enable bit	0 : Reception disabled 1 : Reception enabled	0 : Reception disabled 1 : Reception enabled	○	○
RI	Receive complete flag	0 : No data present in receive buffer register 1 : Data present in receive buffer register	0 : No data present in receive buffer register 1 : Data present in receive buffer register	○	×
U2IRS	UART2 transmit interrupt cause select bit	0 : Transmit buffer empty (TI = 1) 1 : Transmit is completed (TXEPT = 1)	0 : Transmit buffer empty (TI = 1) 1 : Transmit is completed (TXEPT = 1)	○	○
U2RRM	UART2 continuous receive mode enable bit	0 : Continuous receive mode disabled 1 : Continuous receive mode enabled	Must always be "0"	○	○
U2LCH	Data logic select bit	0 : No reverse 1 : Reverse	0 : No reverse 1 : Reverse	○	○
U2ERE	Error signal output enable bit	Must be fixed to “0”	0 : Output disabled 1 : Output enabled	○	○

Figure 2.5.6. UARTi-related registers (4)

UART transmit/receive control register 2

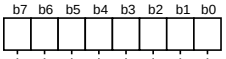


Symbol: UCON | Address: 03B0₁₆ | When reset: X0000000₂

Bit symbol	Bit name	Function (During clock synchronous serial I/O mode)	Function (During UART mode)	R	W
U0IRS	UART0 transmit interrupt cause select bit	0 : Transmit buffer empty (TI = 1) 1 : Transmission completed (TXEPT = 1)	0 : Transmit buffer empty (TI = 1) 1 : Transmission completed (TXEPT = 1)	○	○
U1IRS	UART1 transmit interrupt cause select bit	0 : Transmit buffer empty (TI = 1) 1 : Transmission completed (TXEPT = 1)	0 : Transmit buffer empty (TI = 1) 1 : Transmission completed (TXEPT = 1)	○	○
U0RRM	UART0 continuous receive mode enable bit	0 : Continuous receive mode disabled 1 : Continuous receive mode enable	Must always be “0”	○	○
U1RRM	UART1 continuous receive mode enable bit	0 : Continuous receive mode disabled 1 : Continuous receive mode enabled	Must always be “0”	○	○
CLKMD0	CLK/CLKS select bit 0	Valid when bit 5 = “1” 0 : Clock output to CLK1 1 : Clock output to CLKS1	Invalid	○	○
CLKMD1	CLK/CLKS select bit 1 (Note)	0 : Normal mode (CLK output is CLK1 only) 1 : Transfer clock output from multiple pins function selected	Must always be “0”	○	○
Reserved bit		Must always be set to “0”		○	○
Nothing is assigned. In an attempt to write to this bit, write “0”. The value, if read, turns out to be indeterminate.				—	—

Note: When using multiple pins to output the transfer clock, the following requirements must be met:
• UART1 internal/external clock select bit (bit 3 at address 03A8₁₆) = “0”.

UART2 special mode register



Symbol: U2SMR | Address: 0377₁₆ | When reset: 00₁₆

Bit symbol	Bit name	Function (During clock synchronous serial I/O mode)	Function (During UART mode)	R	W
IICM	I²C mode select bit	0 : Normal mode 1 : I²C mode	Must always be “0”	○	○
ABC	Arbitration lost detecting flag control bit	0 : Update per bit 1 : Update per byte	Must always be “0”	○	○
BBS	Bus busy flag	0 : STOP condition detected 1 : START condition detected	Must always be “0”	○ (Note1)	○
LSYN	SCLL sync output enable bit	0 : Disabled 1 : Enabled	Must always be “0”	○	○
ABSCS	Bus collision detect sampling clock select bit	Must always be “0”	0 : Rising edge of transfer clock 1 : Underflow signal of timer A0	○	○
ACSE	Auto clear function select bit of transmit enable bit	Must always be “0”	0 : No auto clear function 1 : Auto clear at occurrence of bus collision	○	○
SSS	Transmit start condition select bit	Must always be “0”	0 : Ordinary 1 : Falling edge of RxD2	○	○
SDDS	SDA digital delay select bit (Note 2, Note 3)	0 : Analog delay output is selected 1 : Digital delay output is selected (must always be “0” when not using I²C mode)	Must always be “0”	○	○

Note 1: Nothing but “0” may be written.
Note 2: When not in I²C mode, do not set this bit by writing a “1”. During normal mode, fix it to “0”. When this bit = “0”, UART2 special mode register 3 (U2SMR3 at address 0375₁₆) bits 7 to 5 (DL2 to DL0 = SDA digital delay setup bits) are initialized to “000”, with the analog delay circuit selected. Also, when SDDS = “0”, the U2SMR3 register cannot be read or written to.
Note 3: When analog delay is selected, only the analog delay value is effective; when digital delay is selected, only the digital delay value is effective.

Figure 2.5.7. UARTi-related registers (5)

2.5.2 Operation of Serial I/O (transmission in UART mode)

In transmitting data in UART mode, choose functions from those listed in Table 2.5.4. Operations of the circled items are described below. Figure 2.5.8 shows the operation timing, and Figures 2.5.9 and 2.5.10 show the set-up procedures.

Table 2.5.4. Choosed functions

Item	Set-up		Item	Set-up	
Transfer clock source (Note 1)	O	Internal clock (f_1 / f_8 / f_{32})	Sleep mode (Note 1)	O	Sleep mode off
		External clock (CLKi pin)			Sleep mode selected
$\overline{CTS}$ function	O	$\overline{CTS}$ function enabled	Data logic select function (Note 2)	O	No reverse
		$\overline{CTS}$ function disabled			Reverse
Transmission interrupt factor	O	Transmission buffer empty	TxD, RxD I/O polarity reverse bit (Note 2)	O	No reverse
		Transmission complete			Reverse
			Bus collision detection function (Note 2)	O	Not selected
					Selected

Note 1: UART0, UART1 only.
Note 2: UART2 only.

Operation (1) Setting the transmit enable bit to “1” and writing transmission data to the UARTi transmit buffer register readies the data transmissible status.

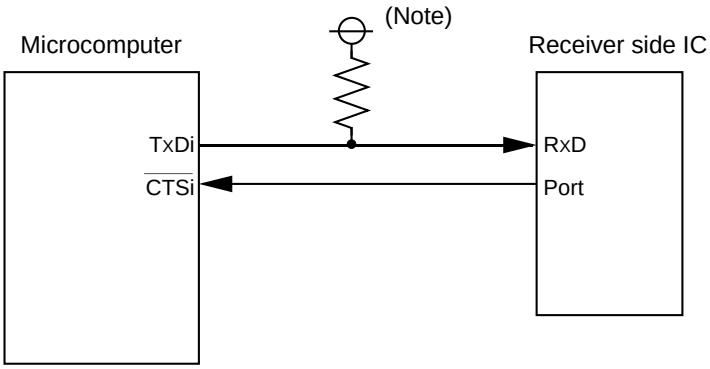
(2) When input to the $\overline{CTSi}$ pin goes to “L”, transmission starts (the $\overline{CTSi}$ pin needs to be controlled on the reception side).

(3) Transmission data held in the UARTi transmit buffer register is transmitted to the UARTi transmit register. At this time, the first bit (the start bit) of the transmission data is transmitted from the TxDi pin. Then, data is transmitted, bit by bit, in sequence: LSB, ···, MSB, parity bit, and stop bit(s).

(4) When the stop bit(s) is (are) transmitted, the transmit register empty flag goes to “1”, which indicates that transmission is completed. At this time, the UARTi transmit interrupt request bit goes to “1”. The transfer clock stops at “H” level.

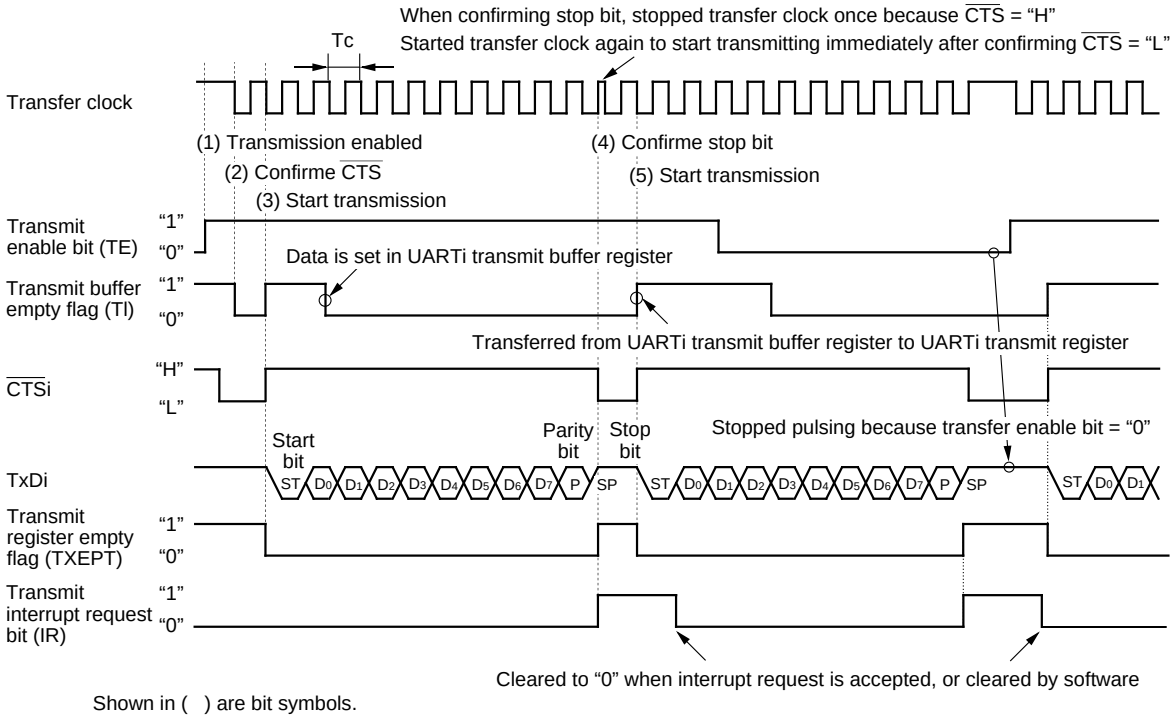
(5) If the transmission condition of the next data is ready when transmission is completed, a start bit is generated following to stop bit(s), and the next data is transmitted.

Example of wiring



Note: Since TxD2 pin is N-channel open drain, this pin needs pull-up resistor.

Example of operation



Shown in ( ) are bit symbols.

The above timing applies to the following settings :
- Parity is enabled.
- One stop bit.
- CTS function is selected.
- Transmit interrupt cause select bit = "1".

Tc = 16 (n + 1) / fi or 16 (n + 1) / fEXT
fi : frequency of BRGi count source (f1, f8, f32)
fEXT : frequency of BRGi count source (external clock)
n : value set to BRGi

Figure 2.5.8. Operation timing of transmission in UART mode

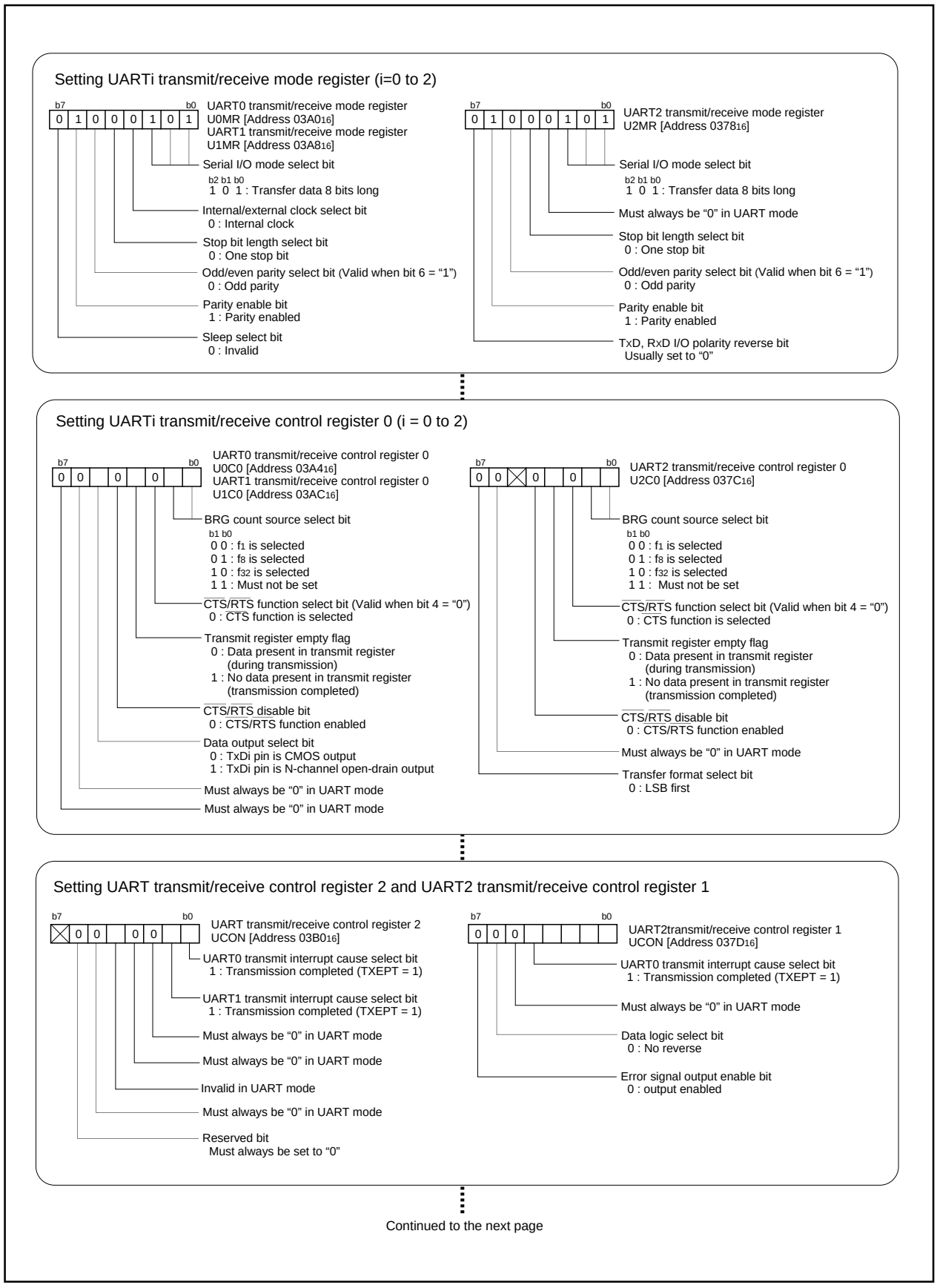


Figure 2.5.9. Set-up procedure of transmission in UART mode (1)

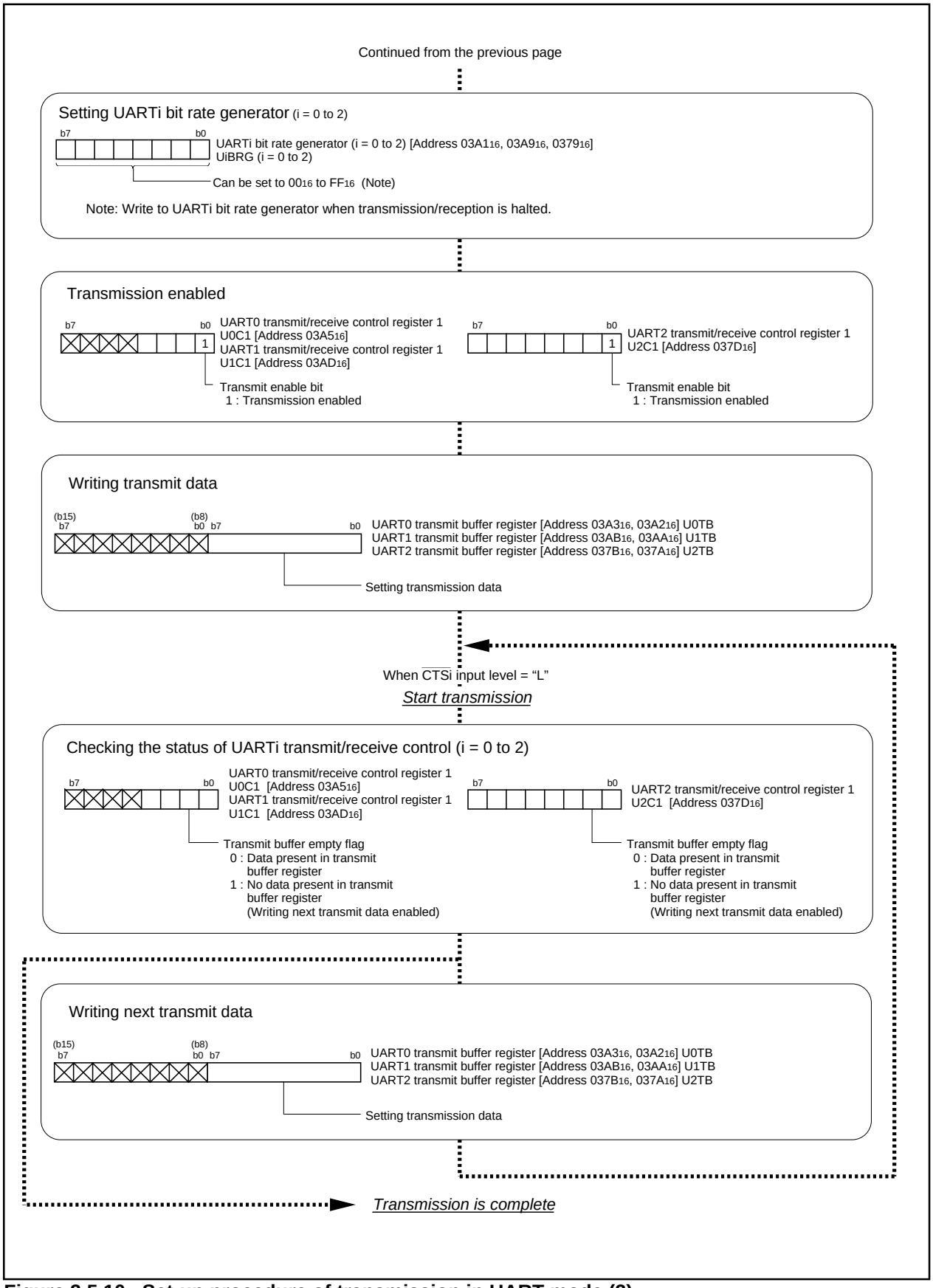


Figure 2.5.10. Set-up procedure of transmission in UART mode (2)

2.5.3 Operation of Serial I/O (reception in UART mode)

In receiving data in UART mode, choose functions from those listed in Table 2.5.5. Operations of the circled items are described below. Figure 2.5.11 shows the operation timing, and Figures 2.5.12 and 2.5.13 show the set-up procedures.

Table 2.5.5. Choosed functions

Item	Set-up		Item	Set-up	
Transfer clock source (Note 1)	O	Internal clock (f_1 / f_8 / f_{32})	Data logic select function (Note 2)	O	No reverse
		External clock (CLKi pin)			Reverse
$\overline{\text{RTS}}$ function	O	$\overline{\text{RTS}}$ function enabled	TxD, RxD I/O polarity reverse bit (Note 2)	O	No reverse
		$\overline{\text{RTS}}$ function disabled			Reverse
Sleep mode (Note 1)	O	Sleep mode off	Bus collision detection function (Note 2)	O	Not selected
		Sleep mode selected			Selected

Note 1: UART0, UART1 only.
Note 2: UART2 only.

Operation (1) Setting the receive enable bit to “1” readies data-receivable status. At this time, output from the $\overline{\text{RTSi}}$ pin goes to “L” level to inform the transmission side that the receivable status is ready.

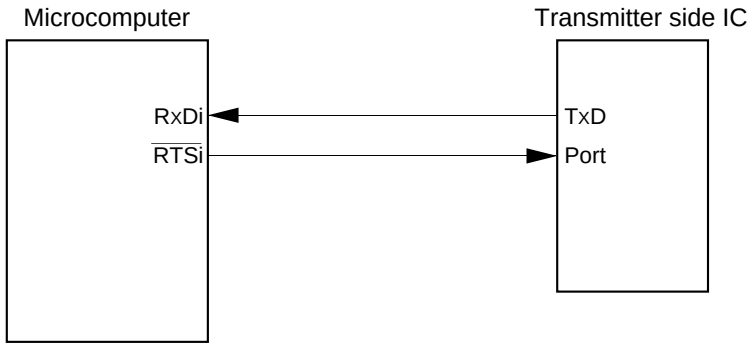
(2) When the first bit (the start bit) of reception data is received from the RxDi pin, output from the $\overline{\text{RTS}}$ goes to “H” level. Then, data is received, bit by bit, in sequence: LSB, ⋯, MSB, and stop bit(s).

(3) When the stop bit(s) is (are) received, the content of the UARTi receive register is transmitted to the UARTi receive buffer register.

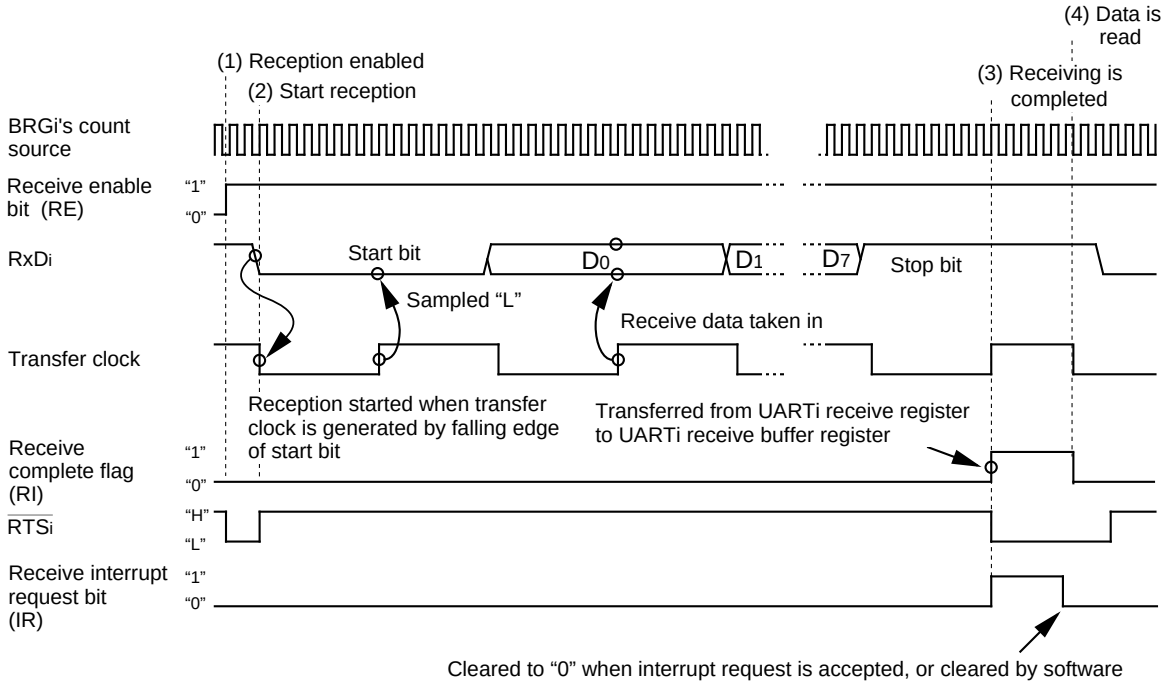
At this time, the receive complete flag goes to “1” to indicate that the reception is completed, the UARTi receive interrupt request bit goes to “1”, and output from the $\overline{\text{RTS}}$ pin goes to “H” level.

(4) The receive complete flag goes to “0” when the lower-order byte of the UARTi buffer register is read.

Example of wiring



Example of operation



Timing of transfer data 8 bits long applies to the following settings :
- Transfer data length is 8 bits.
- Parity is disabled.
- One stop bit
- RTS function is selected.

Figure 2.5.11. Operation timing of reception in UART mode

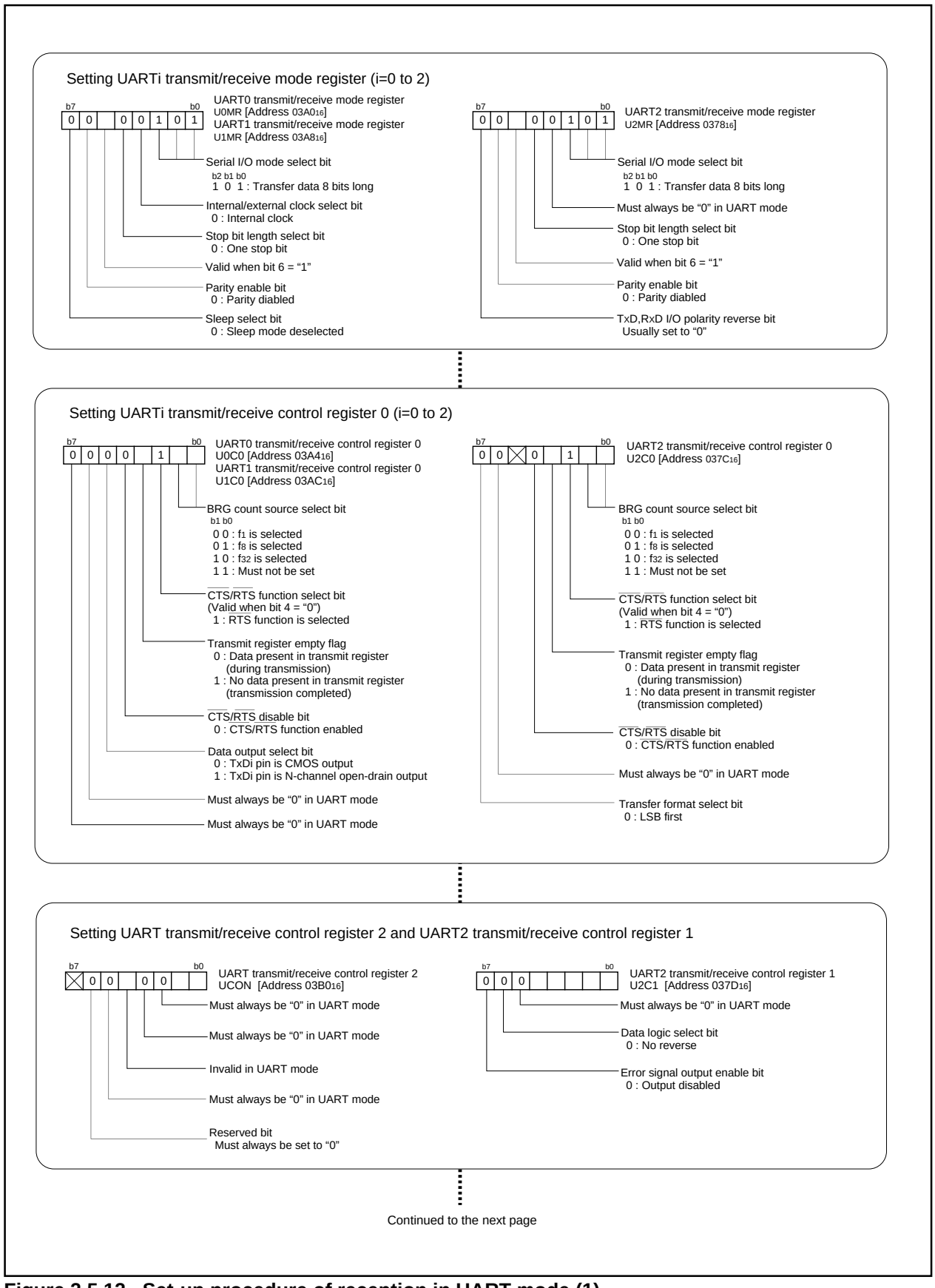


Figure 2.5.12. Set-up procedure of reception in UART mode (1)

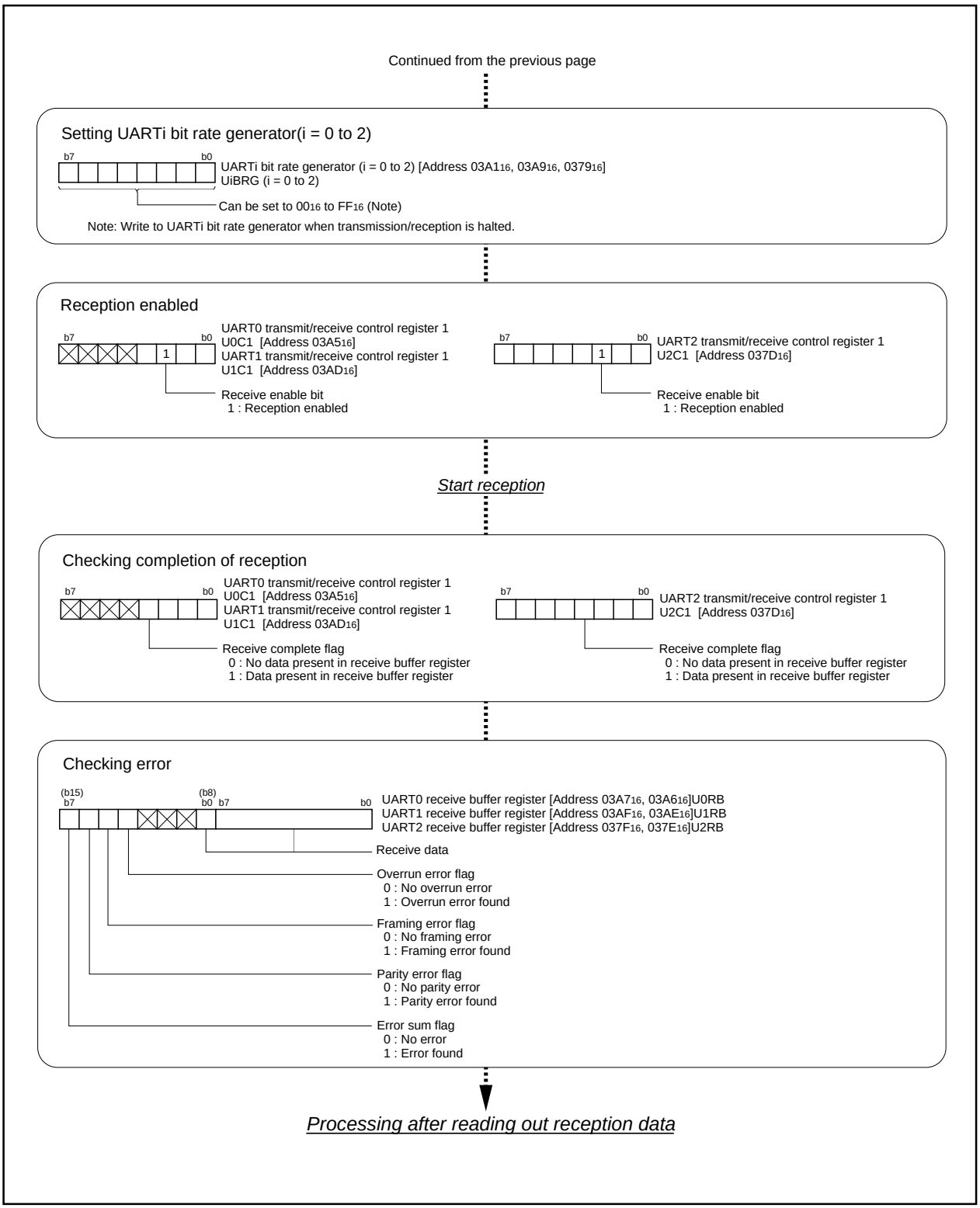


Figure 2.5.13. Set-up procedure of reception in UART mode (2)

2.5.4 Operation of Serial I/O (transmission used for SIM interface)

In transmitting data in UART mode (used for SIM interface), choose functions from those listed in Table 2.5.6. Operations of the circled items are described below. Figure 2.5.14 shows the operation timing, and Figures 2.5.15 and 2.5.16 show the set-up procedures.

Table 2.5.6. Choosed functions

Item	Set-up	
Transfer data format	O	Direct format
		Inverse format

Operation (1) Setting the transmit enable bit and receive enable bit to “1” and writing transmission data to the UART2 transmit buffer register readies the data transmissible status. Set UART2 transfer interrupt is enabled.

(2) Transmission data held in the UART2 transmit buffer register is transmitted to the UART2 transmit register. At this time, the first bit (the start bit) of the transmission data is transmitted from the TxD2 pin. Then, data is transmitted, bit by bit, in sequence: LSB, ⋯, MSB, parity bit, and stop bit(s).

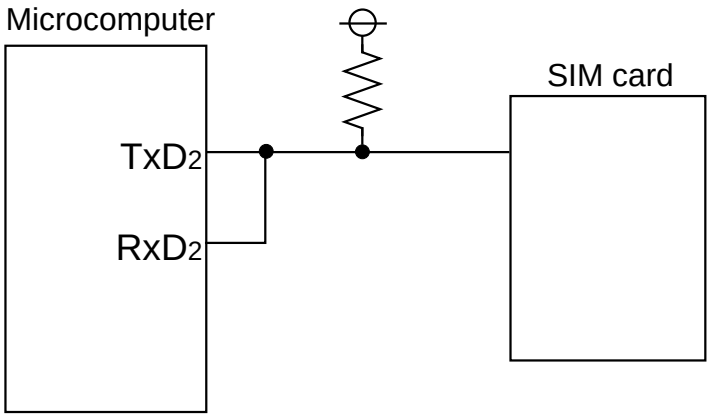
(3) When the stop bit(s) is (are) transmitted, the transmit register empty flag goes to “1”, which indicates that transmission is completed. At this time, the UART2 transmit interrupt request bit goes to “1”. The transfer clock stops at “H” level.

(4) If the transmission condition of the next data is ready when transmission is completed, a start bit is generated following to stop bit(s), and the next data is transmitted.

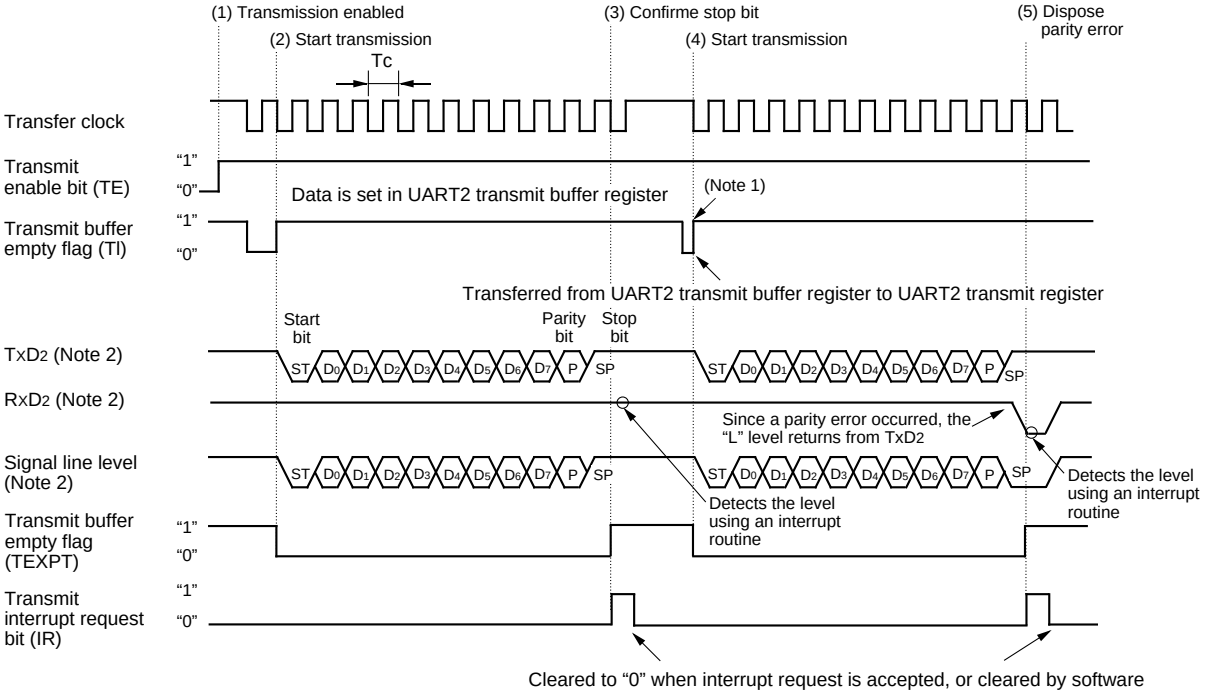
(5) If a parity error occurs, an L is output from the SIM card, and the RxD2 terminal turns to the "L" level. Check the RxD2 terminal's level within the UART2 transmission interrupt routine, and if it is found to be at the "L" level, then handle the error.

Note
- The parity error level is determined within a UART2 transmission interrupt. When a transmission interrupt request occurs, set the priority level of the transmission interrupt higher than those of other interrupts so that the interrupt routine can be immediately carried out. Either in the main routine or in an interrupt routine, the interrupt inhibition time has to be made as short as possible.
- Set the RxD2 terminal's direction register to input.

Example of wiring



Example of operation (when direct format)



Shown in ( ) are bit symbols.

The above timing applies to the following settings :
- Parity is enabled.
- One stop bit.
- Transmit interrupt cause select bit = "1".

Tc = 16 (n + 1) / fi
fi : frequency of BRG2 count source (f1, f8, f32)
n : value set to BRG2

Note 1: The transmit is started with overflow timing of BRG after having written in a value at the transmit buffer in the above timing.
Note 2: TxD2 and RxD2 are connected in the manner of wired OR as shown in the connection diagram. So TxD2 and RxD2 ought to become the same signal from the logical standpoint, but the output signals turn complex, so they are shown separately. Also, the signal level resulting from connecting TxD2 and RxD2 is shown as a signal line level.

Figure 2.5.14. Operation timing of transmission in UART mode (used for SIM interface)

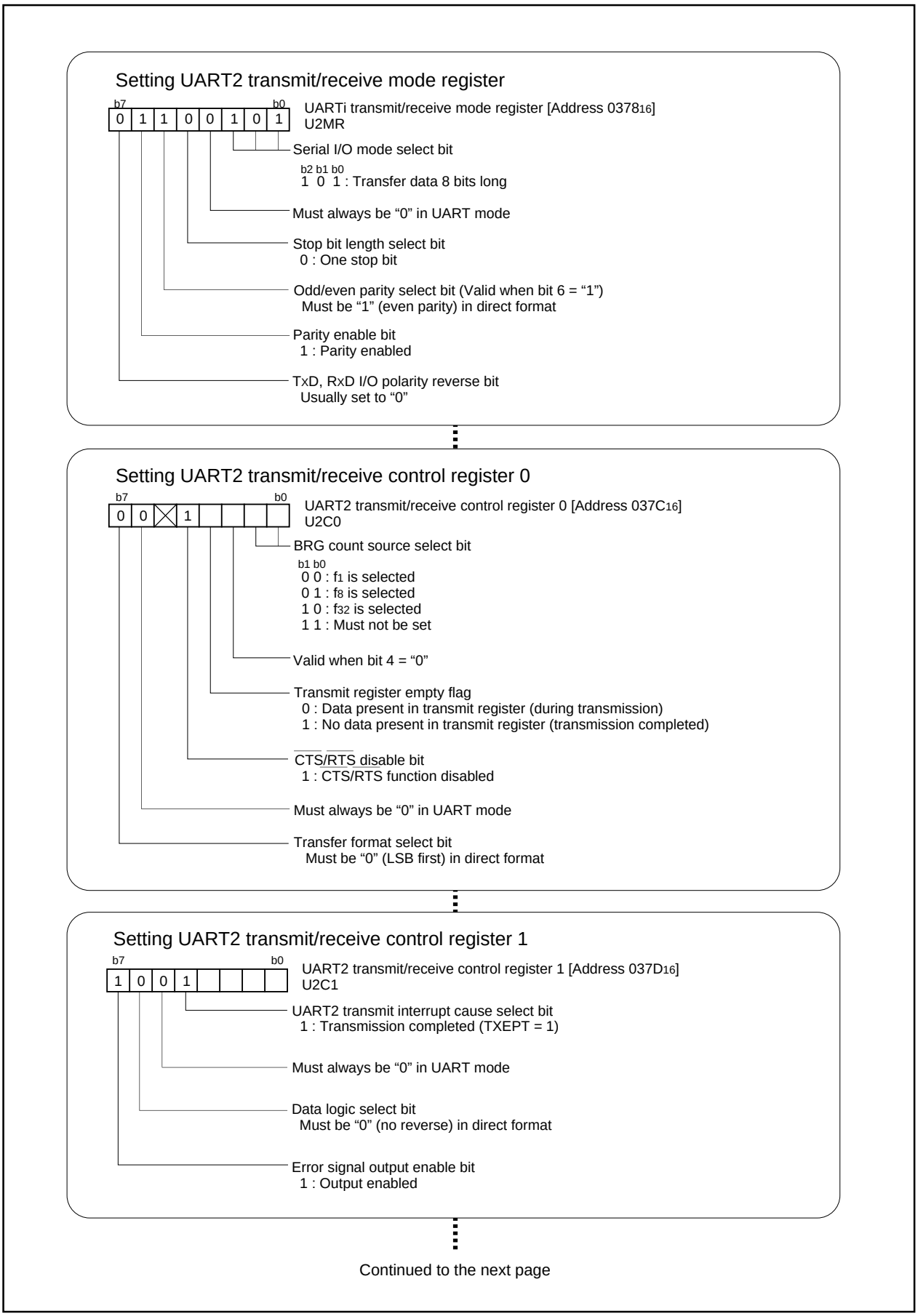


Figure 2.5.15. Set-up procedure of transmission in UART mode (used for SIM interface) (1)

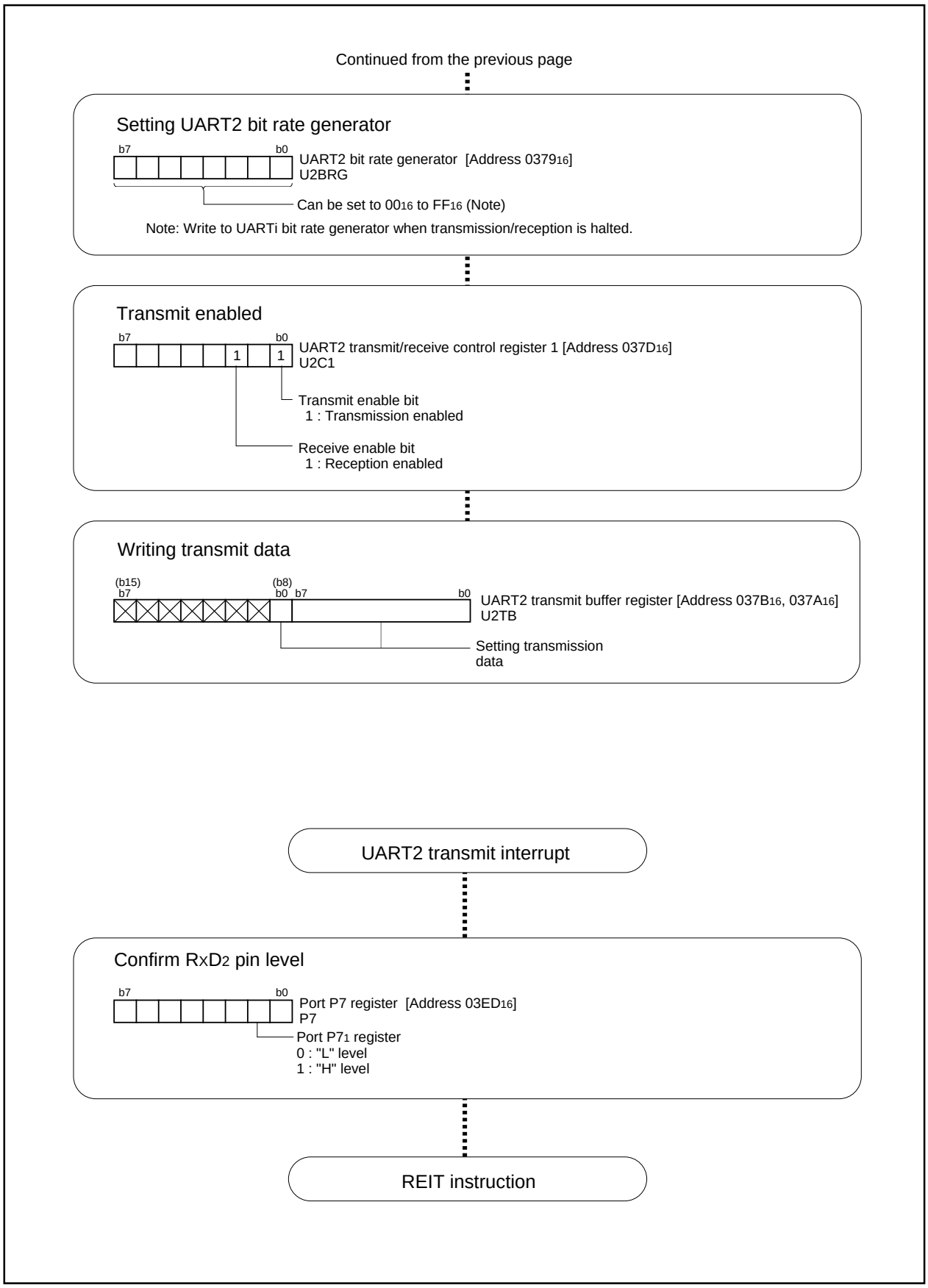


Figure 2.5.16. Set-up procedure of transmission in UART mode (used for SIM interface) (2)

2.5.5 Operation of Serial I/O (reception used for SIM interface)

In receiving data in UART mode (used for SIM interface), choose functions from those listed in Table 2.5.7. Operations of the circled items are described below. Figure 2.5.17 shows the operation timing, and Figures 2.5.18 and 2.5.19 show the set-up procedures.

Figure 2.5.7. Choosed functions

Item	Set-up	
Transfer data format	O	Direct format
		Inverse format

Operation (1) Setting the transmit enable bit and receive enable bit to “1” readies data-receivable status.

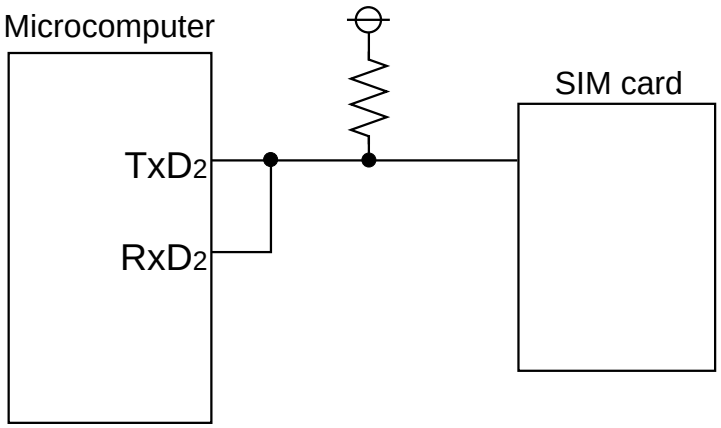
(2) When the first bit (the start bit) of reception data is received from the RxD2 pin, data is received, bit by bit, in sequence: LSB, ···, MSB, and stop bit(s).

(3) When the stop bit(s) is (are) received, the content of the UART2 receive register is transmitted to the UART2 receive buffer register.
At this time, the receive complete flag goes to “1” to indicate that the reception is completed, the UART2 receive interrupt request bit goes to “1”, and output from the $\overline{\text{RTS}}$ pin goes to “H” level.

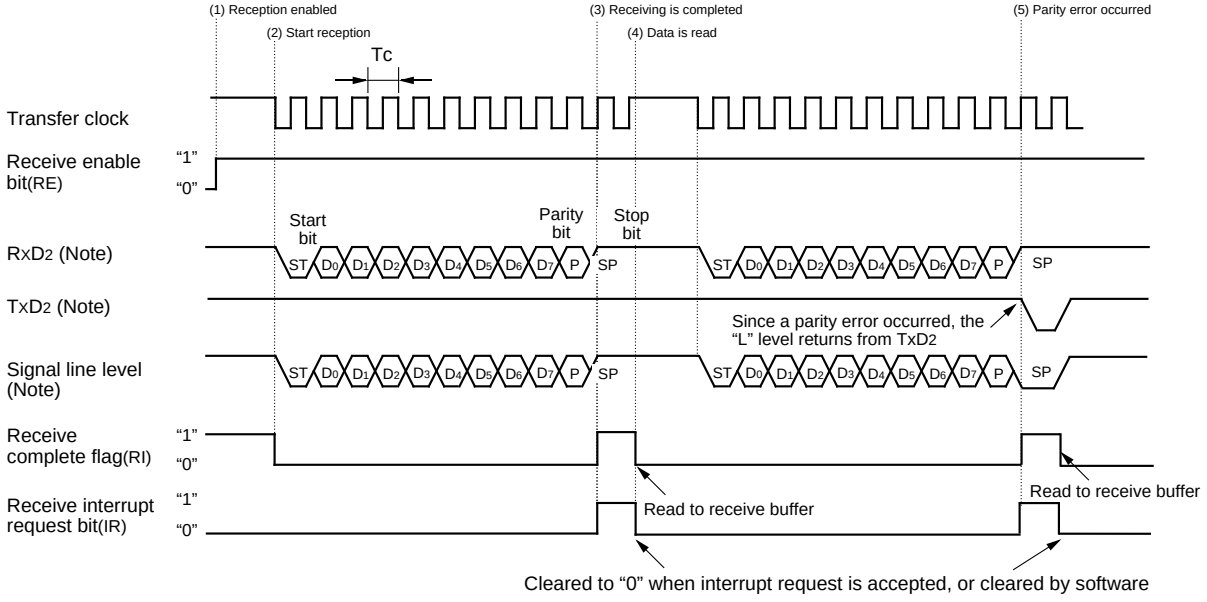
(4) The receive complete flag goes to “0” when the lower-order byte of the UART2 buffer register is read.

(5) When the parity error is occurred, TxD2 pin goes to “L” level.

Example of wiring



Example of operation (when direct format)



Shown in ( ) are bit symbols.

The above timing applies to the following settings :
- Parity is enabled.
- One stop bit.
- Transmit interrupt cause select bit = "1".

Tc = 16 (n + 1) / fi
fi : frequency of BRG2 count source (f1, f8, f32)
n : value set to BRG2

Note: TxD2 and RxD2 are connected in the manner of wired OR as shown in the connection diagram. So TxD2 and RxD2 ought to become the same signal from the logical standpoint, but the output signals turn complex, so they are shown separately. Also, the signal level resulting from connecting TxD2 and RxD2 is shown as a signal line level.

Figure 2.5.17. Operation timing of reception in UART mode (used for SIM interface)

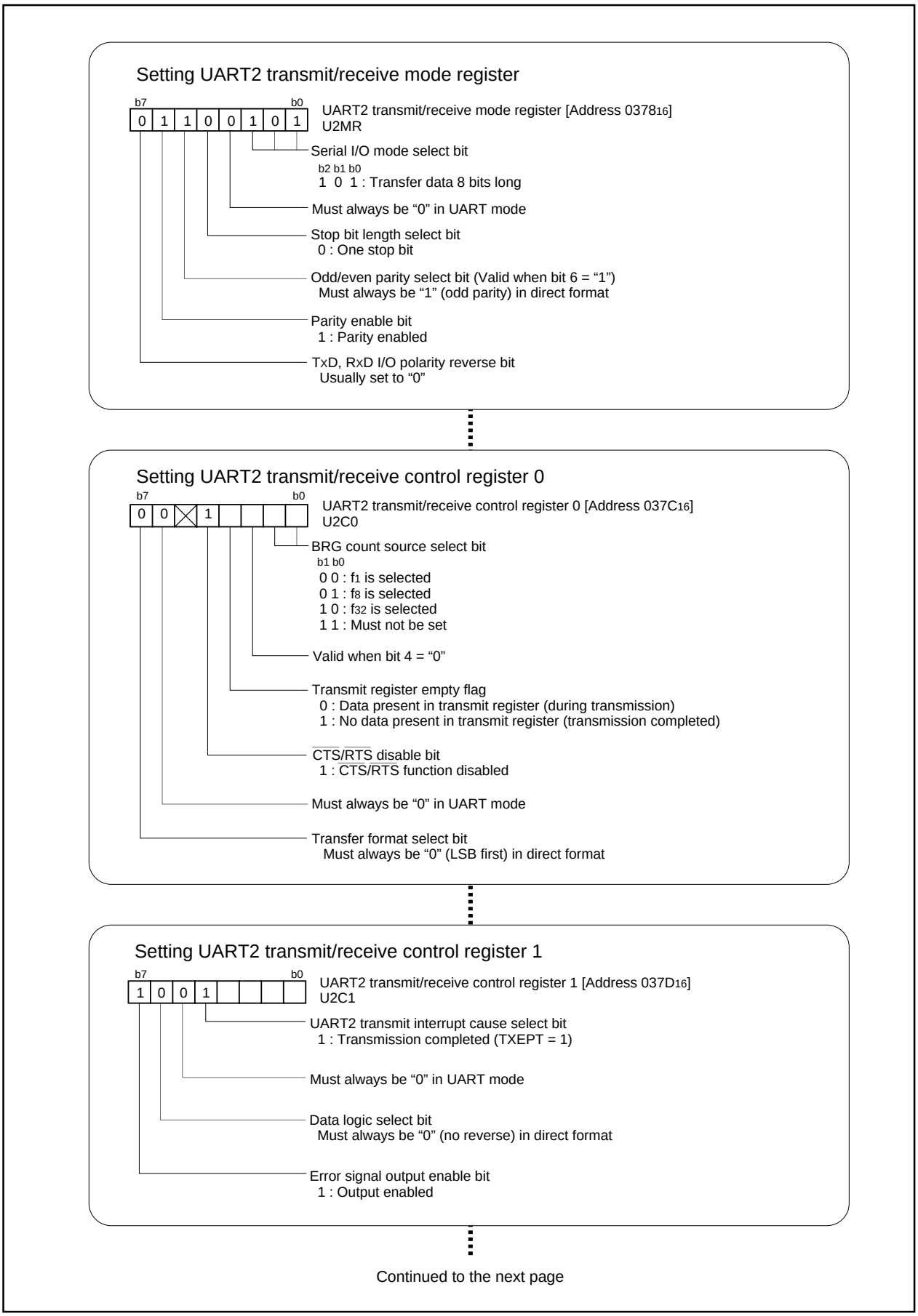


Figure 2.5.18. Set-up procedure of reception in UART mode (used for SIM interface) (1)

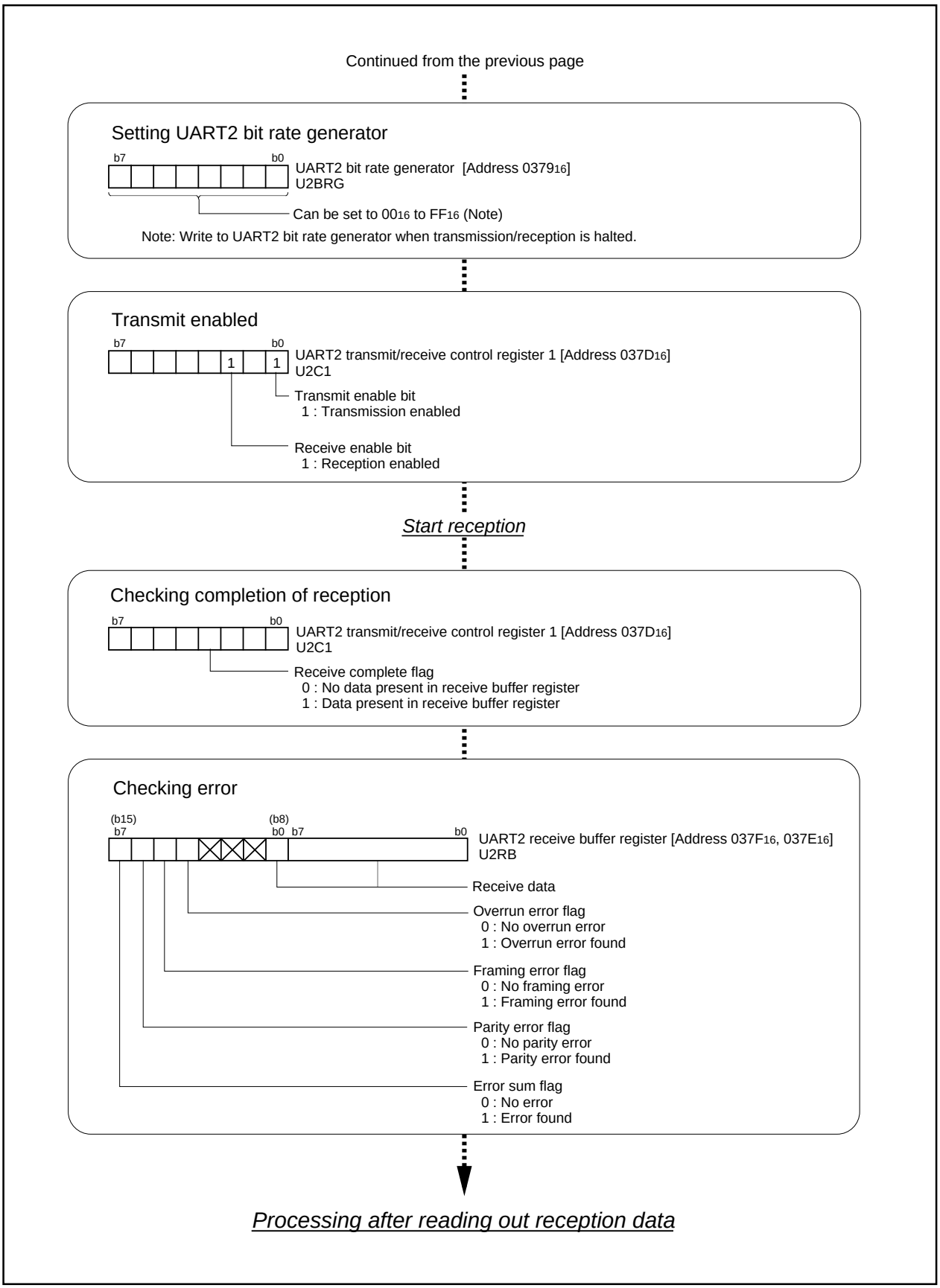


Figure 2.5.19. Set-up procedure of reception in UART mode (used for SIM interface)(2)

2.5.6 Clock Signals in used for the SIM Interface

In conforming to the SIM interface, the UART clock signal within the SIM card needs to conform to the UART2 clock signal within the microprocessor. Two examples are given here as means of generating a UART2 clock signal within the microprocessor.

* In the case of setting a value equal to or less than (1/256 X 1/16) in the division rate of UART2 clock
  Choose f_1 for the UART's source clock signal and set an optional value in the bit rate generator.
* In the case of setting a value equal to or greater than (1/256 X 1/16) in the division rate of UART2 clock
  Set the bit rate generator to "0", turn the source clock signal to timer output and set an optional value in the timer. In order to maintain the synchronization, the serial I/O mode select bits must be reset to "000_2", then the UART2 transmit/receive mode register is set back to the original setting at every one byte transmission.

Let F be the clock signal within the SIM card and D be the bit rate adjustment factor, then the formula for the UART clock signal becomes as follows. Figure 2.5.20 shows an example of connection.

• In the case of setting a value equal to or less than (1/256 X 1/16) in the division rate of UART2 clock
  UART2 clock signal within microprocessor = UART clock within SIM card

$$f_1 \times \frac{1}{\text{Bit rate generator} + 1} \times \frac{1}{16} = f_1 \times \frac{1}{\text{Timer Ai counter} + 1} \times \text{flip-flop} \times \frac{1}{F/D}$$

Let X_{IN} = 16 MHz, timer Ai counter = 1, F = 372, and D = 1, then the value to be set in the bit rate generator becomes

$$16 \times \frac{1}{\text{Bit rate generator} + 1} \times \frac{1}{16} = 16 \times \frac{1}{2} \times \frac{1}{2} \times \frac{1}{372/1}$$

Bit rate generator = 92

Table 2.5.8 shows an example of setting in the UART2 bit rate generator.

• In the case of setting a value equal to or greater than (1/256 X 1/16) in the division rate of UART2 clock
  UART2 clock signal within microprocessor = UART clock within SIM card

$$f_1 \times \frac{1}{\text{Timer Aj counter} + 1} \times \text{flip-flop} \times \frac{1}{\text{Bit rate generator} + 1} \times \frac{1}{16}$$

$$= f_1 \times \frac{1}{\text{Bit rate generator} + 1} \times \text{flip-flop} \times \frac{1}{F/D}$$

Let X_{IN}= 16 MHz, timer Ai counter = 3, bit rate generator = 0, F = 1860, and D = 1, then the value to be set in the timer Aj counter becomes

$$16 \times \frac{1}{\text{Timer Aj counter} + 1} \times \frac{1}{2} \times \frac{1}{0 + 1} \times \frac{1}{16} = 16 \times \frac{1}{3 + 1} \times \frac{1}{2} \times \frac{1}{1860/1}$$

Timer Aj counter = 464

Table 2.5.9 shows an example of setting in the timer Aj counter.

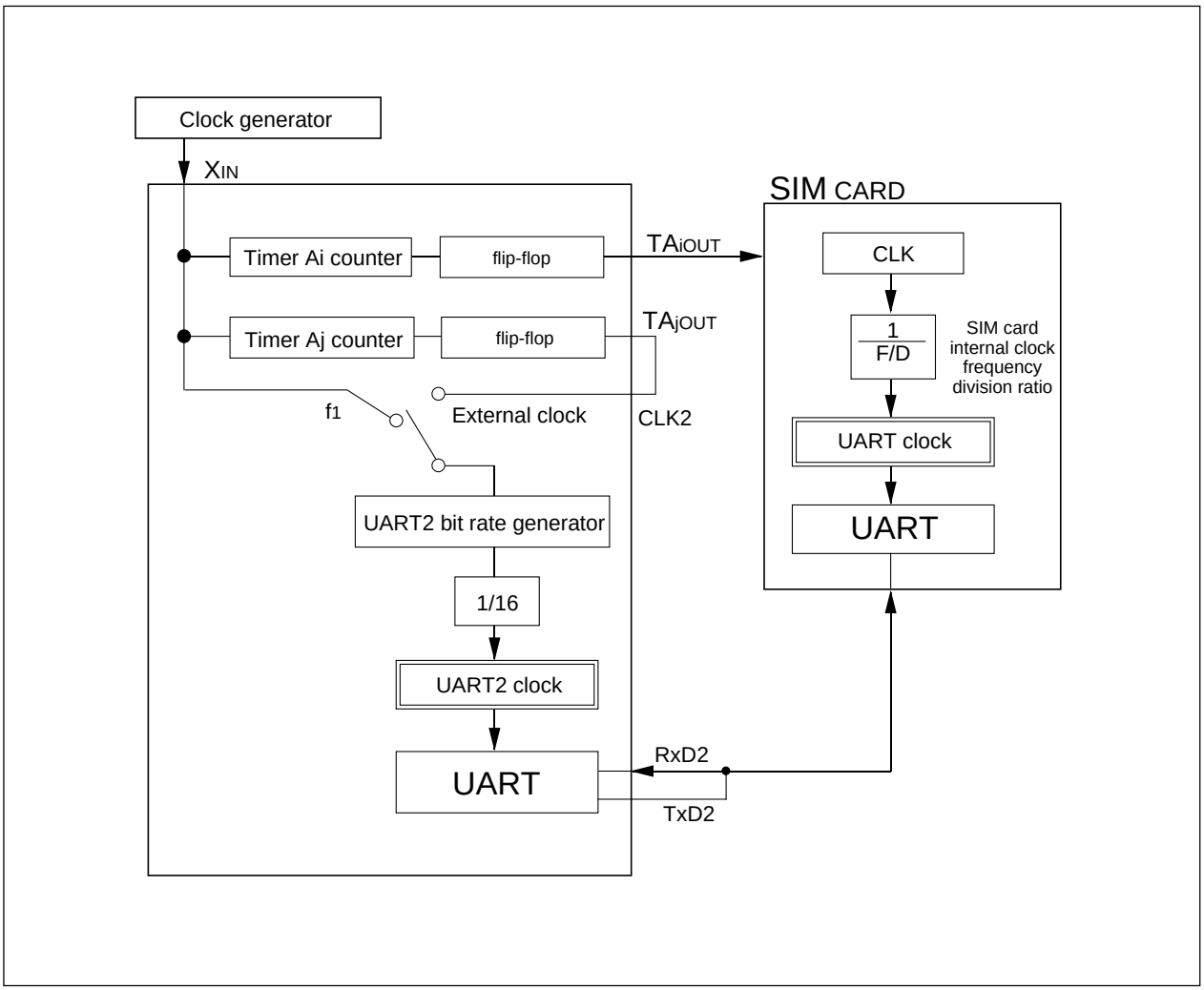


Figure 2.5.20. Example of connection

Table 2.5.8. UART2 bit rate adjustment factor

SIM card internal clock F(Hz)	Bit rate D	F/D	UART2 bit rate generator set value	SIM card internal clock F(Hz)	Bit rate D	F/D	UART2 bit rate generator set value
372	1	372	92	1116	1	1116	
	2	186			2	558	
	4	93			4	279	
	8				8		
	16				16		
	1/2	744	185		1/2	2232	
	1/4	1488			1/4	4464	
	1/8	2976			1/8	8928	
	1/16	5952			1/16	17856	
	1/32	11904			1/32	35712	
	1/64	23808			1/64	71424	
558	1	558		1488	1	1488	
	2	279			2	744	185
	4				4	372	92
	8				8	186	
	16				16	93	
	1/2	1116			1/2	2976	
	1/4	2232			1/4	5952	
	1/8	4464			1/8	11904	
	1/16	8928			1/16	23808	
	1/32	17856			1/32	47616	
	1/64	35712			1/64	95232	
744	1	744	185	1860	1	1860	
	2	372	92		2	930	
	4	186			4	465	
	8	93			8		
	16				16		
	1/2	1488			1/2	3720	
	1/4	2976			1/4	7440	
	1/8	5952			1/8	14880	
	1/16	11904			1/16	29760	
	1/32	23808			1/32	59520	
	1/64	47616			1/64	119040	

Combination impossible

Combination in which the F/D itself does not become an integer

Setting example under the following conditions.

f(XIN)=16MHz

Timer Ai counter set value = 1

Table 2.5.9. TimerAi register adjustment factor

SIM card internal clock F(Hz)	Bit rate D	F/D	Timer Ai value	SIM card internal clock F(Hz)	Bit rate D	F/D	Timer Aj value
372	1	372	92	1116	1	1116	278
	2	186			2	558	
	4	93			4	279	
	8				8		
	16				16		
	1/2	744	185		1/2	2232	557
	1/4	1488	371		1/4	4464	1115
	1/8	2976	743		1/8	8928	2231
	1/16	5952	1487		1/16	17856	4463
	1/32	11904	2975		1/32	35712	8927
	1/64	23808	5951		1/64	71424	17855
558	1	558		1488	1	1488	371
	2	279			2	744	185
	4				4	372	92
	8				8	186	
	16				16	93	
	1/2	1116	278		1/2	2976	743
	1/4	2232	557		1/4	5952	1487
	1/8	4464	1115		1/8	11904	2975
	1/16	8928	2231		1/16	23808	5951
	1/32	17856	4463		1/32	47616	11903
	1/64	35712	8927		1/64	95232	23807
744	1	744	185	1860	1	1860	464
	2	372	92		2	930	
	4	186			4	465	
	8	93			8		
	16				16		
	1/2	1488	371		1/2	3720	929
	1/4	2976	743		1/4	7440	1859
	1/8	5952	1487		1/8	14880	3719
	1/16	11904	2975		1/16	29760	7439
	1/32	23808	5951		1/32	59520	14879
	1/64	47616	11903		1/64	119040	29759

Combination impossible

Combination in which the F/D itself does not become an integer

Setting example under the following conditions.

f(XIN)=16MHz

Timer Ai counter set value = 3, UART2 bit rate generator set value = 0

2.5.7 Error-permitted range range of transfer baud

During reception, the receive data input to the RxDi pin is taken at the rising edge of the transfer clock. Accordingly, in order to receive data correctly, the stop bit must be input when the transfer clock of one-set receive data rises last. Figure 2.5.21 shows the relationship between the transfer clock and receive data.

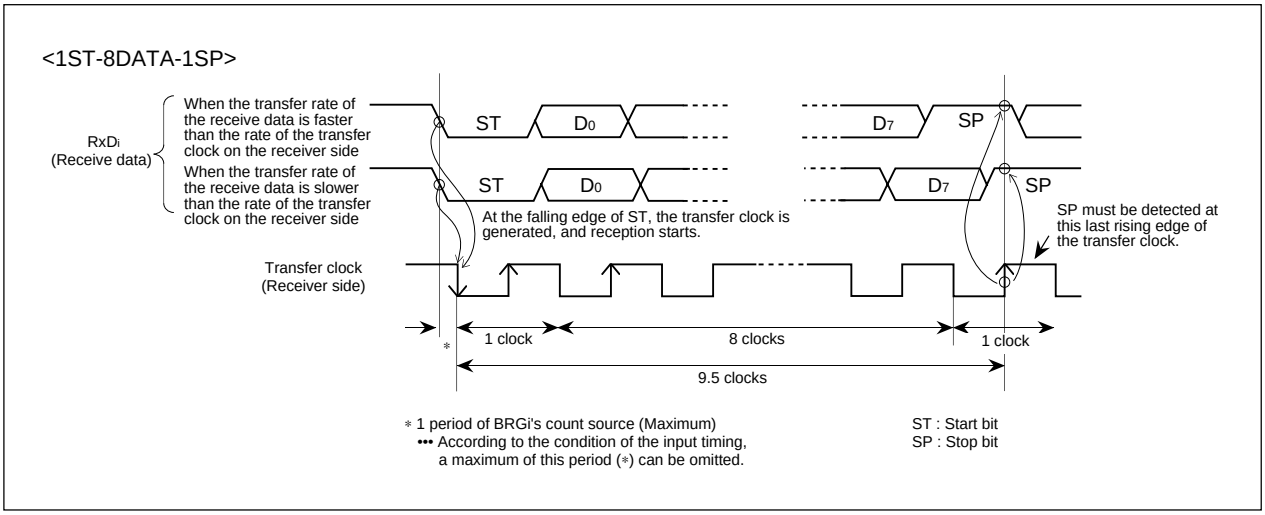


Figure 2.5.21 Relationship between transfer clock and receive data

Accordingly, the transfer rate of the receiver and transmitter sides must satisfy the following fomula in order to receive data correctly.

$$\left(\frac{1}{Bt} \times (b - 1) + \frac{1}{F}\right) < \left(\frac{1}{Br} \times (b - 0.5) + \frac{1}{F}\right) < \left(\frac{1}{Bt} \times b\right)$$

Br : Transfer rate on receiver side (bps)
Bt : Transfer rate on transmitter side (bps)
F : BRGi's count source frequency on receiver side (Hz)
b : Entire bit number of one-set data
(ex : 12 bits in the case of 1ST-8DATA-1PAR-2SP; See Figure 2.5.1)

Be sure to satisfy the above formula, and set the timing with enough margin. Also, the user shall make sufficient evaluation before actually using it.

2.6 SI/O3, 4 Usage

2.6.1 Overview of the SI/O3,4 usage

SI/O3, 4 carries out 8-bit data communications in synchronization with the clock. The following is an overview of the SI/O3, 4 usage.

(1) Transmission/reception format

8-bit data

(2) Transfer rate

If the internal clock is selected as the transfer clock, the divide-by-2 frequency, resulting from the bit rate generator division, becomes the transfer rate. The bit rate generator count source can be selected from the following: f_1, f_8, and f_{32}. Clocks f_1, f_8, and f_{32} are derived by dividing the CPU's main clock by 1, 8, and 32 respectively.

Furthermore, if an external clock is selected as the transfer clock, the clock frequency input to the CLK pin becomes the transfer rate.

(3) Function selection

For SI/O3, 4, the following functions can be selected:

(a) Function for choosing which bit to transmit first

This function is to choose whether to transmit data from bit 0 or from bit 7. Choose either of the following:

- LSB first Data is transmitted from bit 0.
- MSB first Data is transmitted from bit 7.

(b) Choosing output level when not transferring

- Internal clock High-impedance output.
- External clock "H" or "L" output level is selected.

(6) Input to the serial I/O and the direction register

To input an external signal to the serial I/O, set the direction register of the relevant port to input.

(7) Pins related to the SI/O3, 4

- CLK3, CLK4 pins Input/output pins for the transfer clock
- S_{IN3}, S_{IN4} pins Input pins for data
- S_{OUT3}, S_{OUT4} pins Output pins for data

(8) Registers related to the SI/O3, 4

Figure 2.6.1 shows the memory map of SI/O3, 4-related registers, and Figures 2.6.2 show SI/O3, 4-related registers.

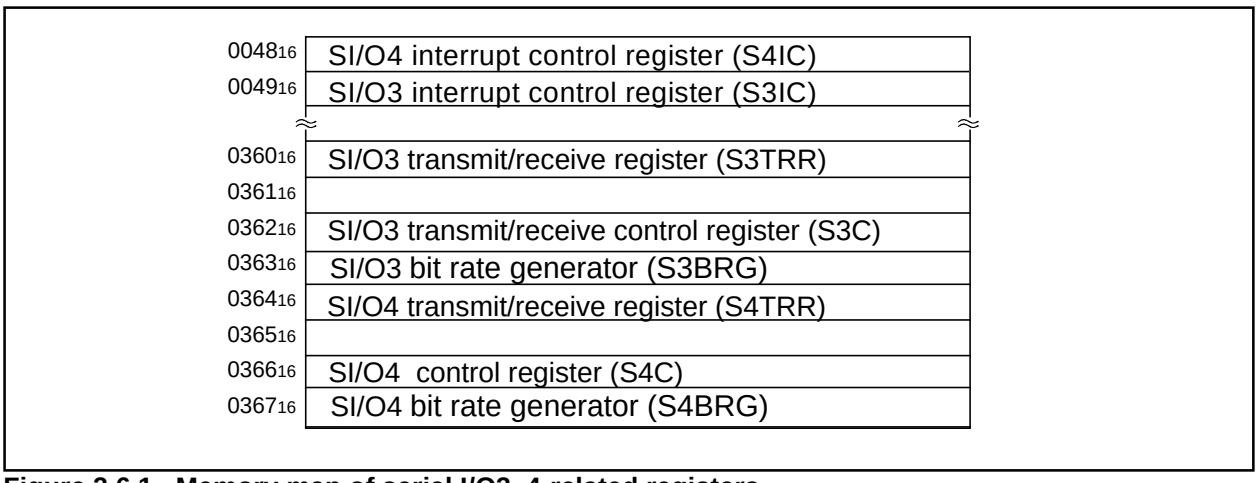
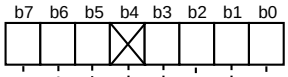


Figure 2.6.1. Memory map of serial I/O3, 4-related registers

S I/Oi control register (i = 3, 4) (Note 1)

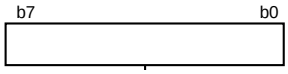


Symbol	Address	When reset
SiC	0362$_{16}$, 0366$_{16}$	40$_{16}$

Bit symbol	Bit name	Description	R	W
SMi0	Internal synchronous clock select bit	b1 b0 0 0 : Selecting f_1 0 1 : Selecting f_8 1 0 : Selecting f_{32} 1 1 : Must not be set.	○	○
SMi1			○	○
SMi2	SOUTi output disable bit	0 : SOUTi output 1 : SOUTi output disable(high impedance)	○	○
SMi3	S I/Oi port select bit (Note 2)	0 : Input-output port 1 : SOUTi output, CLK function	○	○
	Nothing is assigned. In an attempt to write to this bit, write "0". The value, if read, turns out to be "0".		—	—
SMi5	Transfer direction select bit	0 : LSB first 1 : MSB first	○	○
SMi6	Synchronous clock select bit (Note 2)	0 : External clock 1 : Internal clock	○	○
SMi7	SOUTi initial value set bit	Effective when SMi3 = 0 0 : L output 1 : H output	○	○

Note 1: Set "1" in bit 2 of the protection register (000A$_{16}$) in advance to write to the S I/Oi control register (i = 3, 4).
Note 2: When using the port as an input/output port by setting the SI/Oi port select bit (i = 3, 4) to "0", be sure to set the sync clock select bit to "1".

SI/Oi bit rate generator (Note 1, 2)

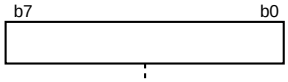


Symbol	Address	When reset
S3BRG	0363$_{16}$	Indeterminate
S4BRG	0367$_{16}$	Indeterminate

Indeterminate	Values that can be set	R	W
Assuming that set value = n, BRGi divides the count source by n + 1	00$_{16}$ to FF$_{16}$		○

Note 1: Write a value to this register while transmit/receive halts.
Note 2: Use MOV instruction to write to this register.

SI/Oi transmit/receive register (Note)



Symbol	Address	When reset
S3TRR	0360$_{16}$	Indeterminate
S4TRR	0364$_{16}$	Indeterminate

Indeterminate	R	W
Transmission/reception starts by writing data to this register. After transmission/reception finishes, reception data is input.	○	○

Note: Write a value to this register while transmit/receive halts.

Figure 2.6.2. Serial I/O3, 4-related registers

2.6.2 Operation of SI/O3,4

In transmitting data in this mode, choose functions from those listed in Table 2.6.1. Operations of the circled items are described below. Figure 2.6.3 shows the operation timing, and Figures 2.6.4 and 2.6.5 show the set-up procedures.

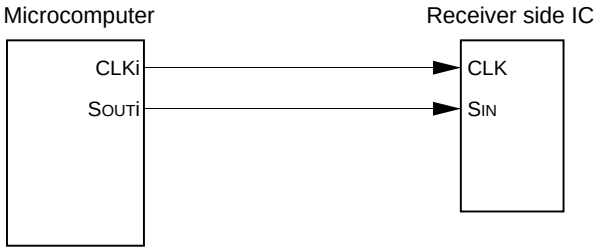
Table 2.6.1. Choosed functions

Item	Set-up		Item	Set-up	
Transfer clock source	O	Internal clock (f_1 / f_8 / f_{32})	SOUTi initial value set function	O	Not used
		External clock (CLKi pin)			Used
Transfer clock	O	LSB first			
		MSB first			

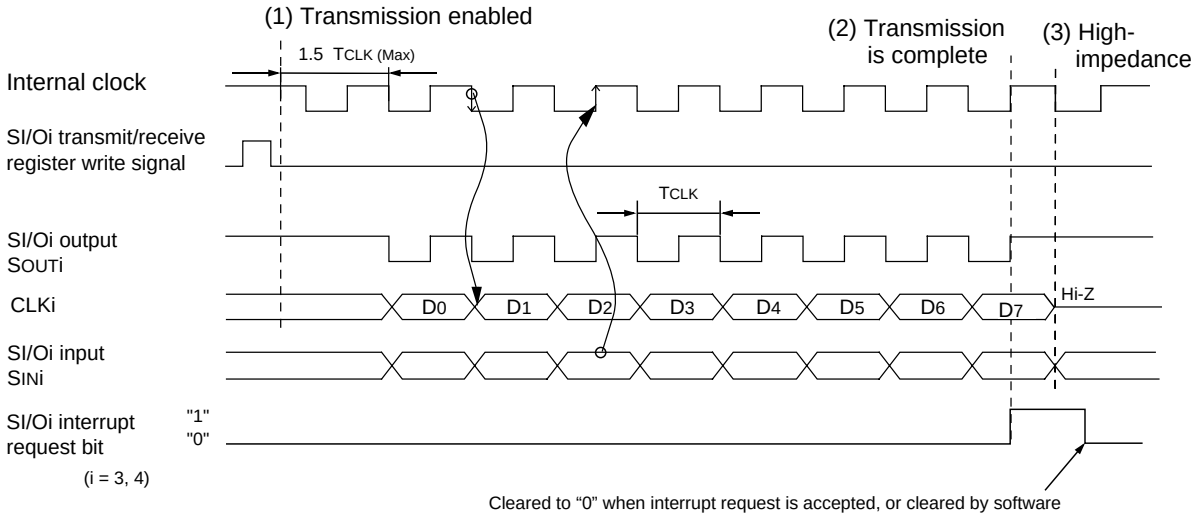
Operation (1) Transfer begins upon writing the SI/Oi transmit data.
The transmit data is sent out from the SOUTi pin synchronously with falling edges of the transfer clock.
(2) When SOUT finishes sending one byte of data, the interrupt request bit is set to 1.
(3) After the transfer is completed, SOUT holds the last data for a 1/2 transfer clock period before going to a high-impedance state.

Note
- Do not write data to the SI/Oi transmit/receive register (i = 3, 4; addresses 0360_{16}, 0364_{16}) during a transfer.
- Data can only be written to the SI/Oi transmit/receive register when the device is idle neither sending nor receiving data.

Example of wiring



Example of operation



TCLK = 2(n + 1) / fi
fi: frequency of BRGi count source (f_1, f_8, f_{32})
n: value set to SiBRG

Figure 2.6.3. Operation timing of transmission in SI/O3, 4 mode

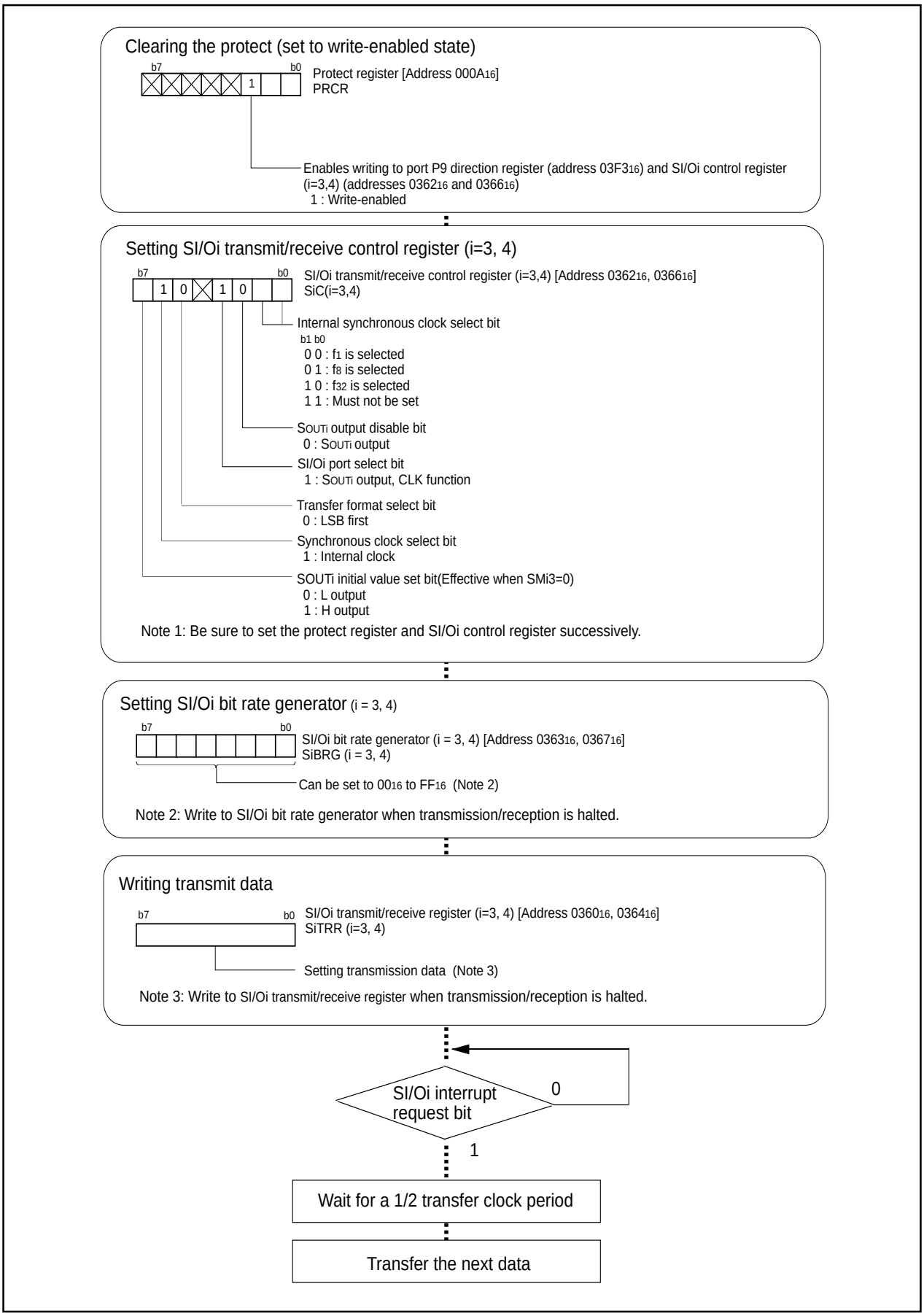


Figure 2.6.4. Set-up procedure of transmission in SI/O3, 4 mode

2.7 A-D Converter Usage

2.7.1 Overview of the A-D converter usage

The A-D converter used in the M16C/62A group operates on a successive conversion basis. The following is an overview of the A-D converter usage.

(1) Mode

The A-D converter operates in one of five modes:

(a) One-shot mode

Carries out A-D conversion on input level of one specified pin only once.

(b) Repetition mode

Repeatedly carries out A-D conversion on input level of one specified pin.

(c) One-shot sweep mode

Carries out A-D conversion on input level of two or more specified pins only once.

(d) Repeated sweep mode 0

Repeatedly carries out A-D conversion on input level of two or more pins.

(e) Repeated sweep mode 1

Repeatedly carries out A-D conversion on input level of two or more pins. This mode is different from the repeated sweep mode 0 in that weights can be assigned to specifing pins control the number of conversion times.

(2) Operation clock

The operation clock in 5 V operation can be selected from the following: f_{AD}, divide-by-2 f_{AD}, and divide-by-4 f_{AD}. In 3 V operation, the selection is divide-by-2 f_{AD} or divide-by-4. The f_{AD} frequency is equal to that of the CPU's main clock.

(3) Conversion time

Number of conversion for A-D convertor varies depending on resolution as given. Table 2.7.1 shows relation between the A-D converter operation clock and conversion time.

Sample & Hold function selected:

33 ϕAD cycles for 10-bit resolution, or 28 ϕAD cycles for 8-bit resolution

No Sample & Hold function:

59 ϕAD cycles for 10-bit resolution, or 49 ϕAD cycles for 8-bit resolution

Table 2.7.1. Conversion time every operation clock

Frequency selection bit 1		0		1
Frequency selection bit 0		0	1	Invalid
A-D converter's operation clock		$\phi AD = \frac{f_{AD}}{4}$	$\phi AD = \frac{f_{AD}}{2}$	$\phi AD = f_{AD}$
Min. conversion cycles (Note 1)	8-bit mode	28 X ϕAD		
	10-bit mode	33 X ϕAD		
Min. conversion time (Note 2)	8-bit mode	11.2µs	5.6µs	2.8µs
	10-bit mode	13.2µs	6.6µs	3.3µs

Note 1: The number of conversion cycles per one analog input pin.

Note 2: The conversion time per one analog input pin (when $f_{AD} = f(X_{IN}) = 10$ MHz)

(4) Functions selection

(a) Sample & Hold function

Sample & Hold function samples input voltage when A-D conversion starts and carries out A-D conversion on the voltage sampled. When A-D conversion starts, input voltage is sampled for 3 cycles of the operation clock. When the Sample & Hold function is selected, set the operation clock for A-D conversion to 1 MHz or higher.

(b) 8-bit A-D to 10-bit A-D switching function

Either 8-bit resolution or 10-bit resolution can be selected. When 8-bit resolution is selected, the 8 higher-order bits of the 10-bit A-D are subjected to A-D conversion. The equations for 10-bit resolution and 8-bit resolution are given below:

10-bit resolution $(Vref \times n / 2^{10}) - (Vref \times 0.5 / 2^{10})$ (n = 1 to 1023), 0 (n = 0)

8-bit resolution $(Vref \times n / 2^{8}) - (Vref \times 0.5 / 2^{10})$ (n = 1 to 255), 0 (n = 0)

(c) A-D conversion by external trigger

The user can select software or an external pin input to start A-D conversion.

(d) External operation amplifier connection function

The selected A-D convertor pin input voltage can be output from the ANEX0 pin. By connecting an operation amplifier between the ANEX1 pin and ANEX0 pin when using this function, the input voltage to all A-D conversion pins can be amplified with one operation amplifier.

(e) Expanded analog input pins function

A-D conversion can be done for voltage input from either the ANEX0 pin or the ANEX1 pin.

(f) Connecting or cutting Vref

Cutting Vref allows decrease of the current flowing into the A-D converter. To decrease the microcomputer's power consumption, cut Vref. To carry out A-D conversion, start A-D conversion 1 µs or longer after connecting Vref.

The following are exsamples in which functions (a) through (f) are selected:

• One-shot mode P2-116
• One-shot mode, trigger by $\overline{\text{ADTRG}}$ P2-118
• One-shot mode, software trigger, expanded analog input P2-120
• One-shot mode, software trigger, external operation amplifier connected P2-122
• Repeat mode, software trigger P2-124
• One-shot sweep mode, software trigger P2-126
• Repeated sweep mode 0, software trigger P2-128
• Repeated sweep mode 1, software trigger P2-130

(5) Input to A-D converter and direction register

To use the A-D converter, set the direction register of the relevant port to input.

(6) Pins related to A-D converter

(a) AN_0 pin through AN_7 pin	Input pins of the A-D converter
(b) AVcc pin	Power source pin of the analog section
(c) V_{REF} pin	Input pin of reference voltage
(d) AVss pin	GND pin of the analog section
(e) ANEX0 pin and ANEX1 pin	Expanded input pins of the A-D converter
(f) $\overline{ADTRG}$ pin	Trigger input pin of the A-D converter

(7) A-D converter and related registers

Figure 2.7.1 shows the memory map of A-D converter-related registers, and Figures 2.7.2 through 2.7.4 show A-D converter-related registers.

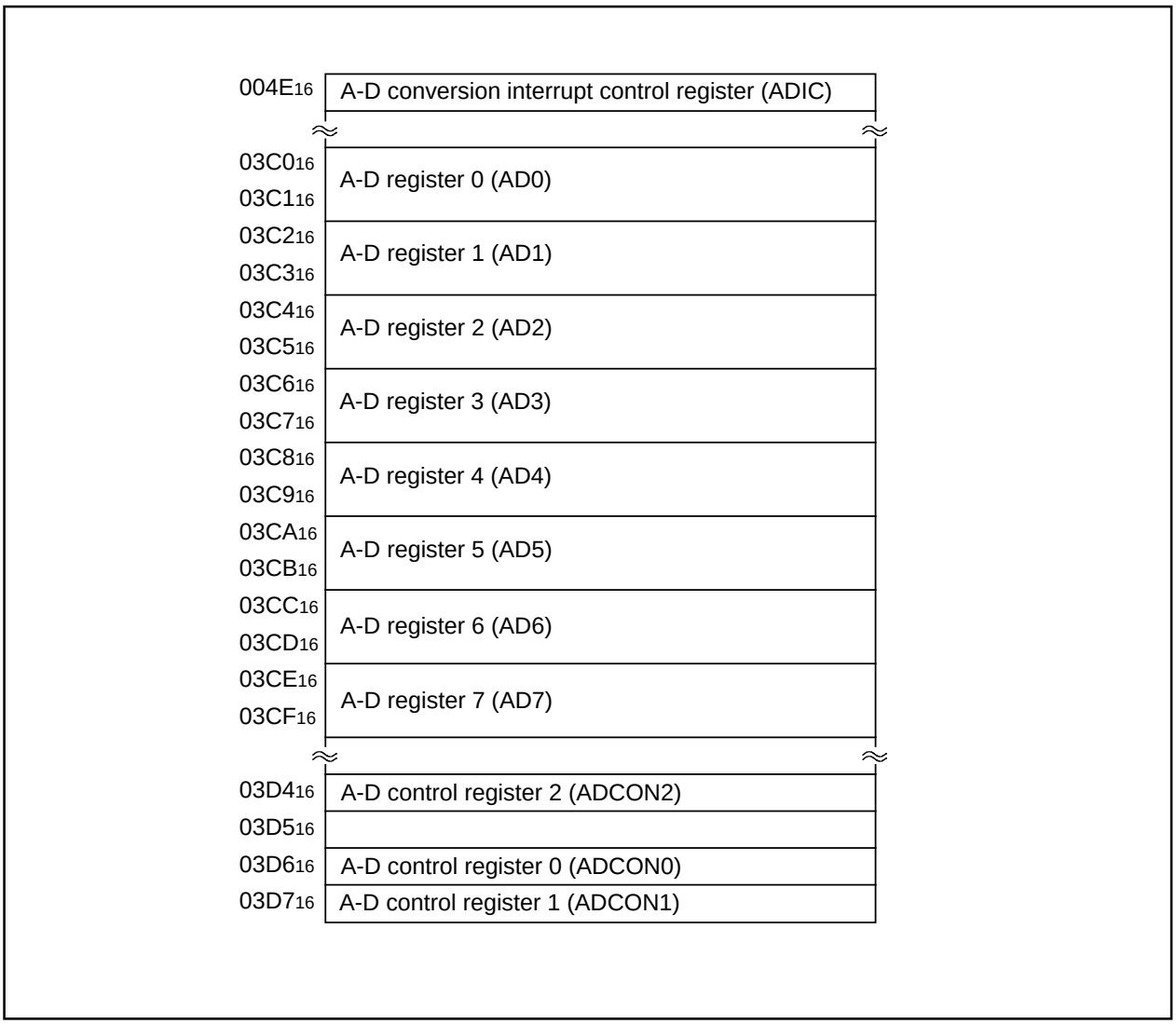


Figure 2.7.1. Memory map of A-D converter-related registers

A-D control register 0 (Note 1)

b7	b6	b5	b4	b3	b2	b1	b0

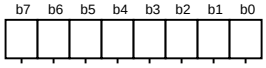
Symbol	Address	When reset
ADCON0	03D6$_{16}$	00000XXX$_2$

Bit symbol	Bit name	Function	R	W
CH0	Analog input pin select bit	b2 b1 b0 0 0 0 : AN$_0$ is selected 0 0 1 : AN$_1$ is selected 0 1 0 : AN$_2$ is selected 0 1 1 : AN$_3$ is selected 1 0 0 : AN$_4$ is selected 1 0 1 : AN$_5$ is selected 1 1 0 : AN$_6$ is selected 1 1 1 : AN$_7$ is selected (Note 2)	○	○
CH1			○	○
CH2			○	○
MD0	A-D operation mode select bit 0	b4 b3 0 0 : One-shot mode 0 1 : Repeat mode 1 0 : Single sweep mode 1 1 : Repeat sweep mode 0 Repeat sweep mode 1 (Note 2)	○	○
MD1			○	○
TRG	Trigger select bit	0 : Software trigger 1 : ADTRG trigger	○	○
ADST	A-D conversion start flag	0 : A-D conversion disabled 1 : A-D conversion started	○	○
CKS0	Frequency select bit 0	0 : f$_{AD}$/4 is selected 1 : f$_{AD}$/2 is selected	○	○

Note 1: If the A-D control register is rewritten during A-D conversion, the conversion result is indeterminate.
Note 2: When changing A-D operation mode, set analog input pin again.

Figure 2.7.2. A-D converter-related registers (1)

A-D control register 1 (Note)



Symbol: ADCON1 | Address: $03D7_{16}$ | When reset: 00_{16}

Bit symbol	Bit name	Function	R	W
SCAN0	A-D sweep pin select bit	When single sweep and repeat sweep mode 0 are selected b1 b0 0 0 : AN_0, AN_1 (2 pins) 0 1 : AN_0 to AN_3 (4 pins) 1 0 : AN_0 to AN_5 (6 pins) 1 1 : AN_0 to AN_7 (8 pins)	○	○
SCAN1		When repeat sweep mode 1 is selected b1 b0 0 0 : AN_0 (1 pin) 0 1 : AN_0, AN_1 (2 pins) 1 0 : AN_0 to AN_2 (3 pins) 1 1 : AN_0 to AN_3 (4 pins)	○	○
MD2	A-D operation mode select bit 1	0 : Any mode other than repeat sweep mode 1 1 : Repeat sweep mode 1	○	○
BITS	8/10-bit mode select bit	0 : 8-bit mode 1 : 10-bit mode	○	○
CKS1	Frequency select bit 1	0 : $f_{AD}/2$ or $f_{AD}/4$ is selected 1 : f_{AD} is selected	○	○
VCUT	Vref connect bit	0 : Vref not connected 1 : Vref connected	○	○
OPA0	External op-amp connection mode bit	b7 b6 0 0 : ANEX0 and ANEX1 are not used 0 1 : ANEX0 input is A-D converted 1 0 : ANEX1 input is A-D converted 1 1 : External op-amp connection mode	○	○
OPA1			○	○

Note: If the A-D control register is rewritten during A-D conversion, the conversion result is indeterminate.

Figure 2.7.3. A-D converter-related registers (2)

A-D control register 2 (Note)

Symbol	Address	When reset
ADCON2	$03D4_{16}$	$0000XXX0_2$

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

Bit symbol	Bit name	Function	R	W
SMP	A-D conversion method select bit	0 : Without sample and hold 1 : With sample and hold	○	○
Reserved bit		Must always be set to “0”	○	○
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be “0”.			–	–

Note: If the A-D control register is rewritten during A-D conversion, the conversion result is indeterminate.

A-D register i

Symbol	Address	When reset
ADi(i=0 to 7)	$03C0_{16}$ to $03CF_{16}$	Indeterminate

(b15) b7							(b8) b0	b7							b0
X	X	X	X	X	X	X									

Function	R	W
Eight low-order bits of A-D conversion result	○	×
• During 10-bit mode Two high-order bits of A-D conversion result	○	×
• During 8-bit mode When read, the content is indeterminate	×	×
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be “0”.	–	–

Figure 2.7.4. A-D converter-related registers (3)

2.7.2 Operation of A-D converter (one-shot mode)

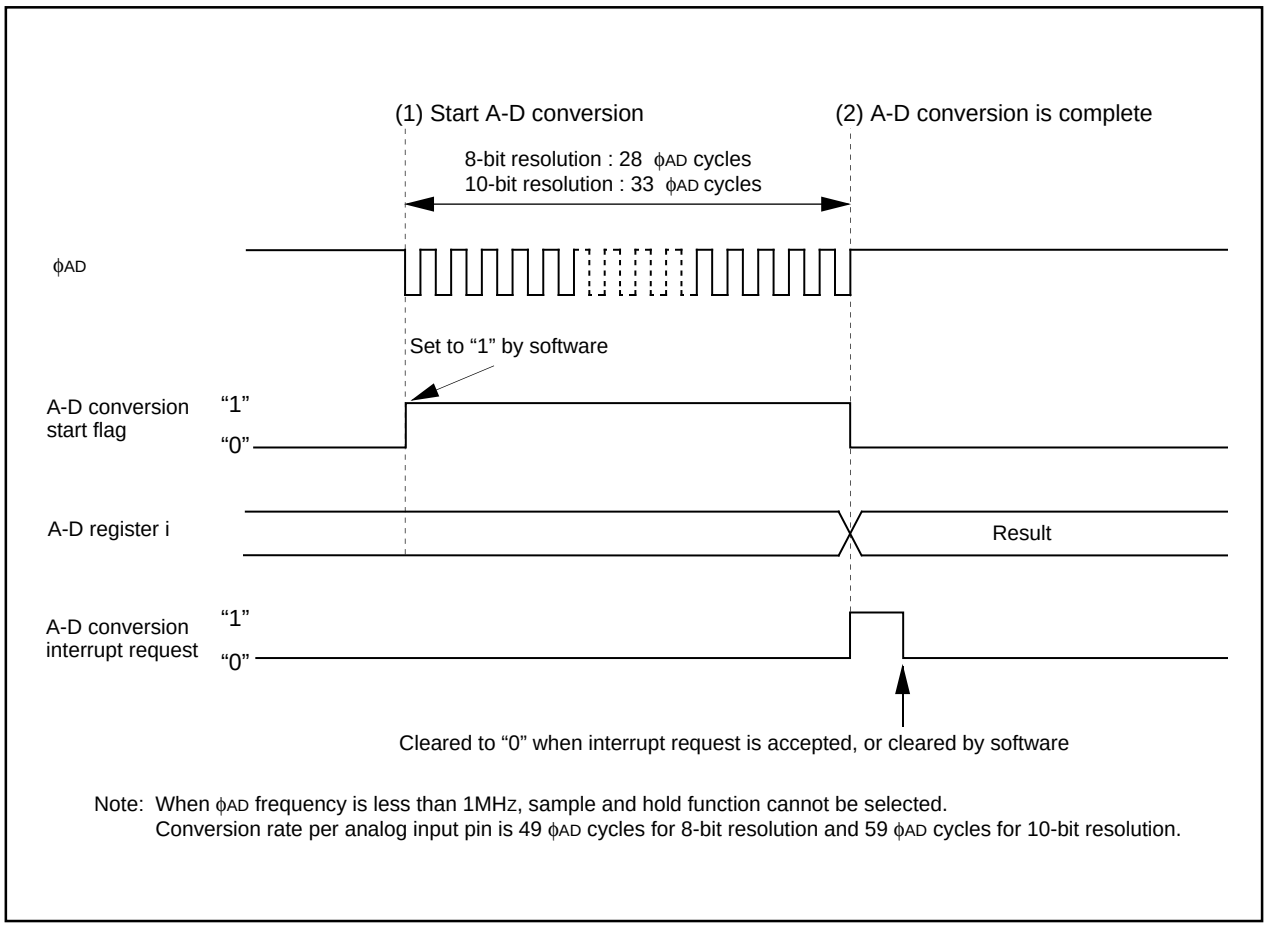
In one-shot mode, choose functions from those listed in Table 2.7.2. Operations of the circled items are described below. Figure 2.7.5 shows the operation timing, and Figure 2.7.6 shows the set-up procedure.

Table 2.7.2. Choosed functions

Item	Set-up		Item	Set-up	
Operation clock ϕ_{AD}	**O**	Divided-by-4 f_{AD} / divided-by-2 f_{AD} / f_{AD}	Expanded analog input pin	**O**	Not used
					Either ANEX0 pin or ANEX1 pin
Resolution	**O**	8-bit / 10-bit			External operation amplifier connection mode
Analog input pin	**O**	One of AN_0 pin to AN_7 pin			
Trigger for starting A-D conversion	**O**	Software trigger	Sample & Hold		Not activated
		Trigger by $\overline{ADTRG}$		**O**	Activated

Operation (1) Setting the A-D conversion start flag to “1” causes the A-D converter to begin operating.

(2) After A-D conversion is completed, the content of the successive comparison register (conversion result) is transmitted to A-D register i. At this time, the A-D conversion interrupt request bit goes to “1”. Also, the A-D conversion start flag goes to “0”, and the A-D converter stops operating.



Note: When ϕ_{AD} frequency is less than 1MHz, sample and hold function cannot be selected.
Conversion rate per analog input pin is 49 ϕ_{AD} cycles for 8-bit resolution and 59 ϕ_{AD} cycles for 10-bit resolution.

Figure 2.7.5. Operation timing of one-shot mode

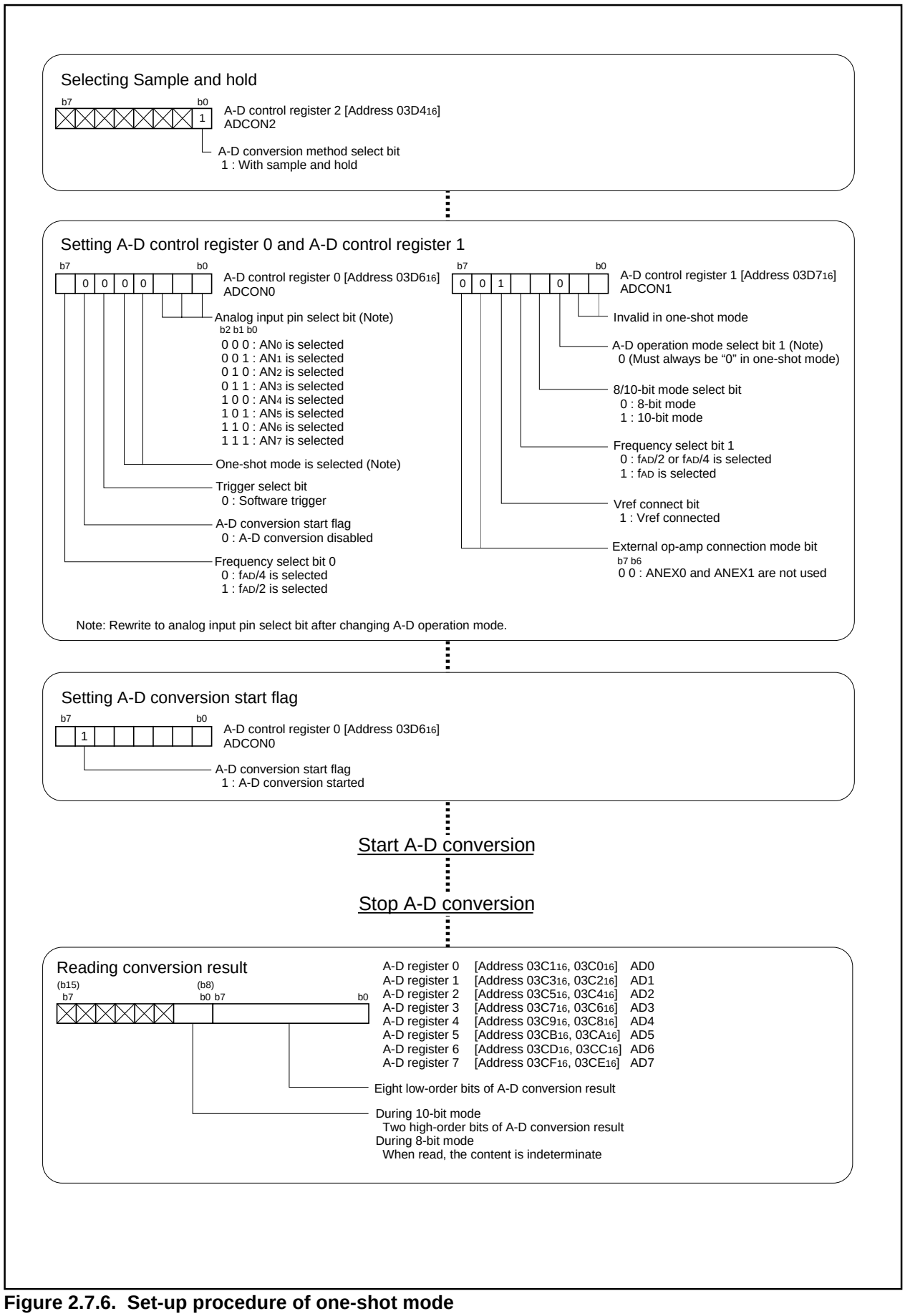


Figure 2.7.6. Set-up procedure of one-shot mode

2.7.3 Operation of A-D Converter (in one-shot mode, an external trigger selected)

In one-shot mode, choose functions from those listed in Table 2.7.3. Operations of the circled items are described below. Figure 2.7.7 shows timing chart, and Figure 2.7.8 shows the set-up procedure.

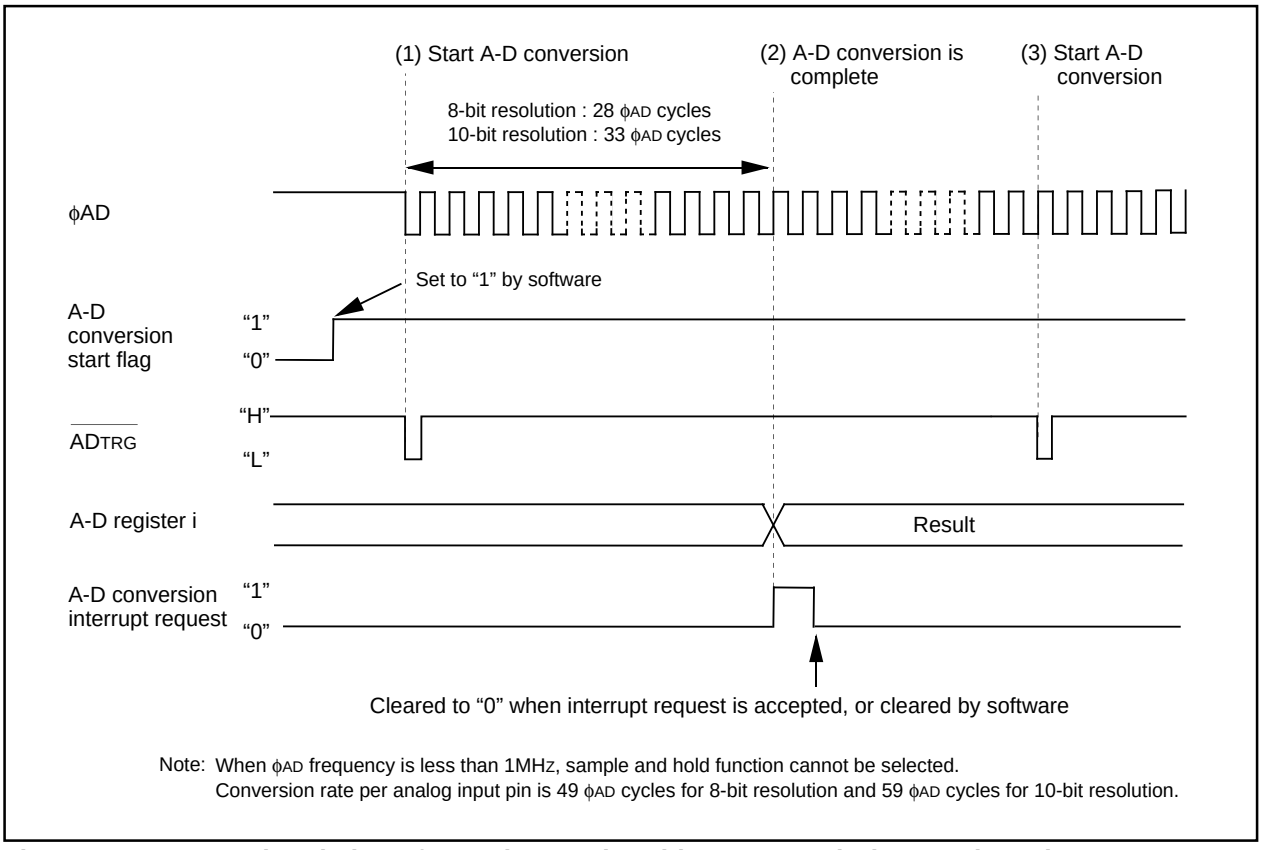
Table 2.7.3. Choosed functions

Item	Set-up		Item	Set-up	
Operation clock ϕ_{AD}	O	Divided-by-4 f_{AD} / divided-by-2 f_{AD} / f_{AD}	Expanded analog input pin	O	Not used
					Either ANEX0 pin or ANEX1 pin
Resolution	O	8-bit / 10-bit			External operation amplifier connection mode
Analog input pin	O	One of AN_0 pin to AN_7 pin			
Trigger for starting A-D conversion		Software trigger	Sample & Hold		Not activated
	O	Trigger by $\overline{ADTRG}$		O	Activated

Operation (1) If the level of the $\overline{ADTRG}$ changes from “H” to “L” with the A-D conversion start flag set to “1”, the A-D converter begins operating.

(2) After A-D conversion is completed, the content of the successive comparison register (conversion result) is transmitted to A-D register i. At this time, the A-D conversion interrupt request bit goes to “1”. Also the A-D converter stops operating.

(3) If the level of the $\overline{ADTRG}$ pin changes from “H” to “L”, the A-D converter carries out conversion from step (1) again. If the level of the $\overline{ADTRG}$ pin changes from “H” to “L” while conversion is in progress, the A-D converter stops the A-D conversion in process, and carries out conversion from step (1) again.



Note: When ϕ_{AD} frequency is less than 1MHz, sample and hold function cannot be selected.
Conversion rate per analog input pin is 49 ϕ_{AD} cycles for 8-bit resolution and 59 ϕ_{AD} cycles for 10-bit resolution.

Figure 2.7.7. Operation timing of one-shot mode, with an external trigger selected

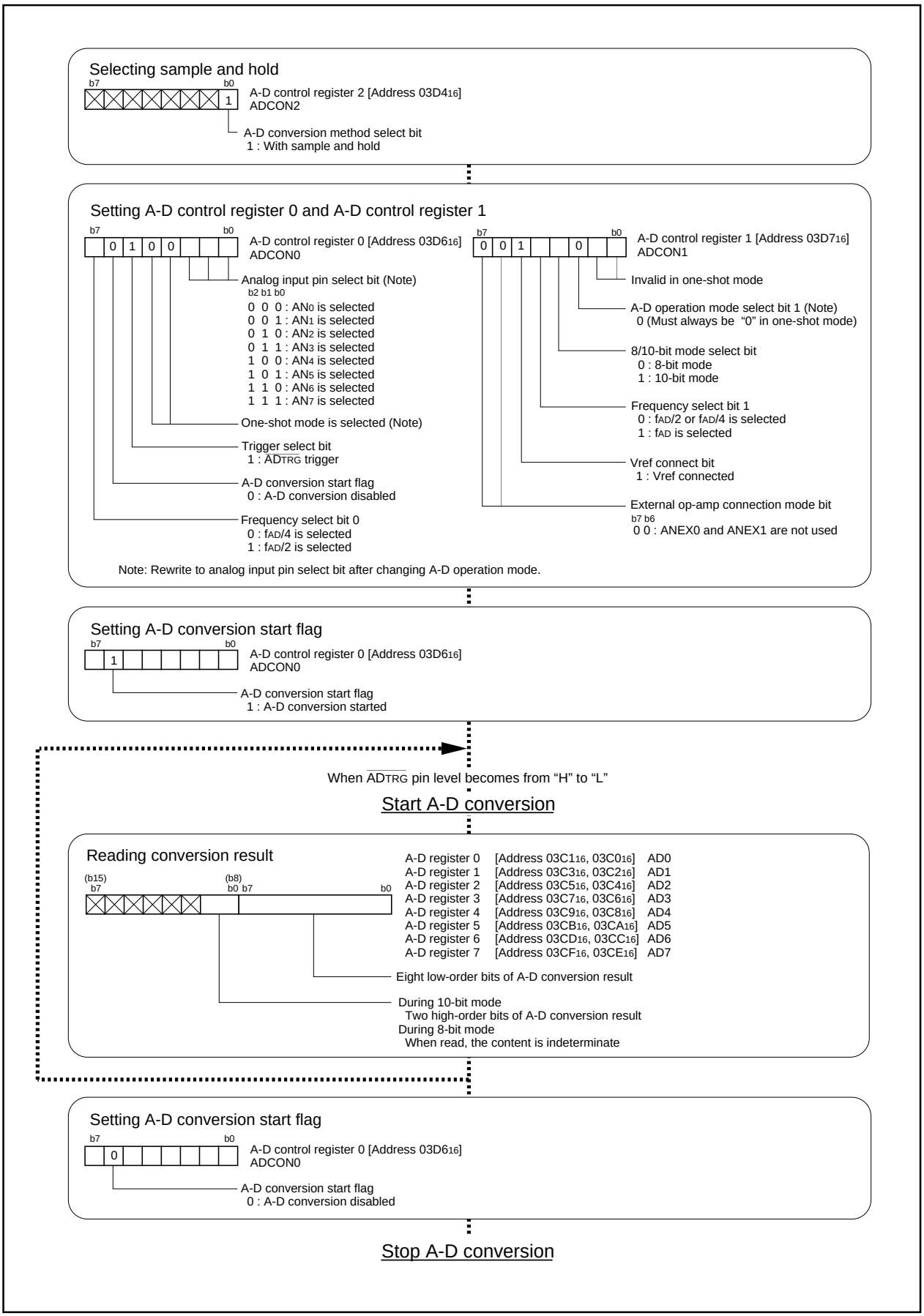


Figure 2.7.8. Set-up procedure of one-shot mode, with an external trigger selected

2.7.4 Operation of A-D Converter (in one-shot mode, expanded analog input pin selected)

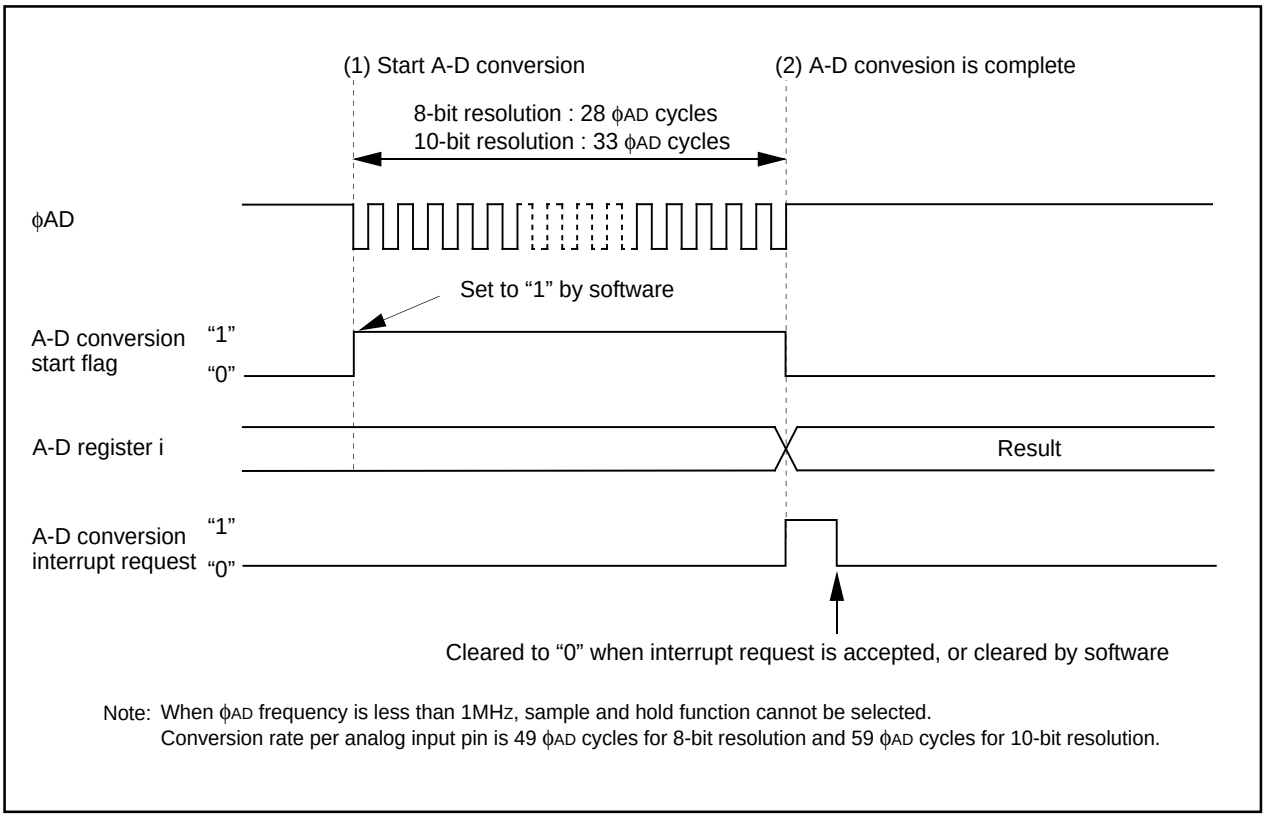
In one-shot mode, choose functions from those listed in Table 2.7.4. Operations of the circled items are described below. Figure 2.7.9 shows timing chart, and Figure 2.7.10 shows the set-up procedure.

Table 2.7.4. Choosed functions

Item	Set-up		Item	Set-up	
Operation clock ϕ_{AD}	O	Divided-by-4 f_{AD} / divided-by-2 f_{AD} / f_{AD}	Expanded analog input pin		Not used
				O	Either ANEX0 pin or ANEX1 pin
Resolution	O	8-bit / 10-bit			External operation amplifier connection mode
Analog input pin	O	One of AN_0 pin to AN_7 pin			
Trigger for starting A-D conversion	O	Software trigger	Sample & Hold		Not activated
		Trigger by $\overline{ADTRG}$		O	Activated

Operation (1) Setting the A-D conversion start flag to “1” causes the A-D converter to start the conversion on voltage input to the ANEXi pin.

(2) After the A-D conversion of voltage input to the ANEXi pin is completed, the content of the successive comparison register (conversion result) is transmitted to the A-D register. At the same time, the A-D conversion interrupt request bit goes to “1”. Also, the A-D conversion start flag goes to “0”, and the A-D converter stops operating.



Note: When ϕ_{AD} frequency is less than 1MHz, sample and hold function cannot be selected.
Conversion rate per analog input pin is 49 ϕ_{AD} cycles for 8-bit resolution and 59 ϕ_{AD} cycles for 10-bit resolution.

Figure 2.7.9. Operation timing of one-shot mode, with expanded analog input pin selected

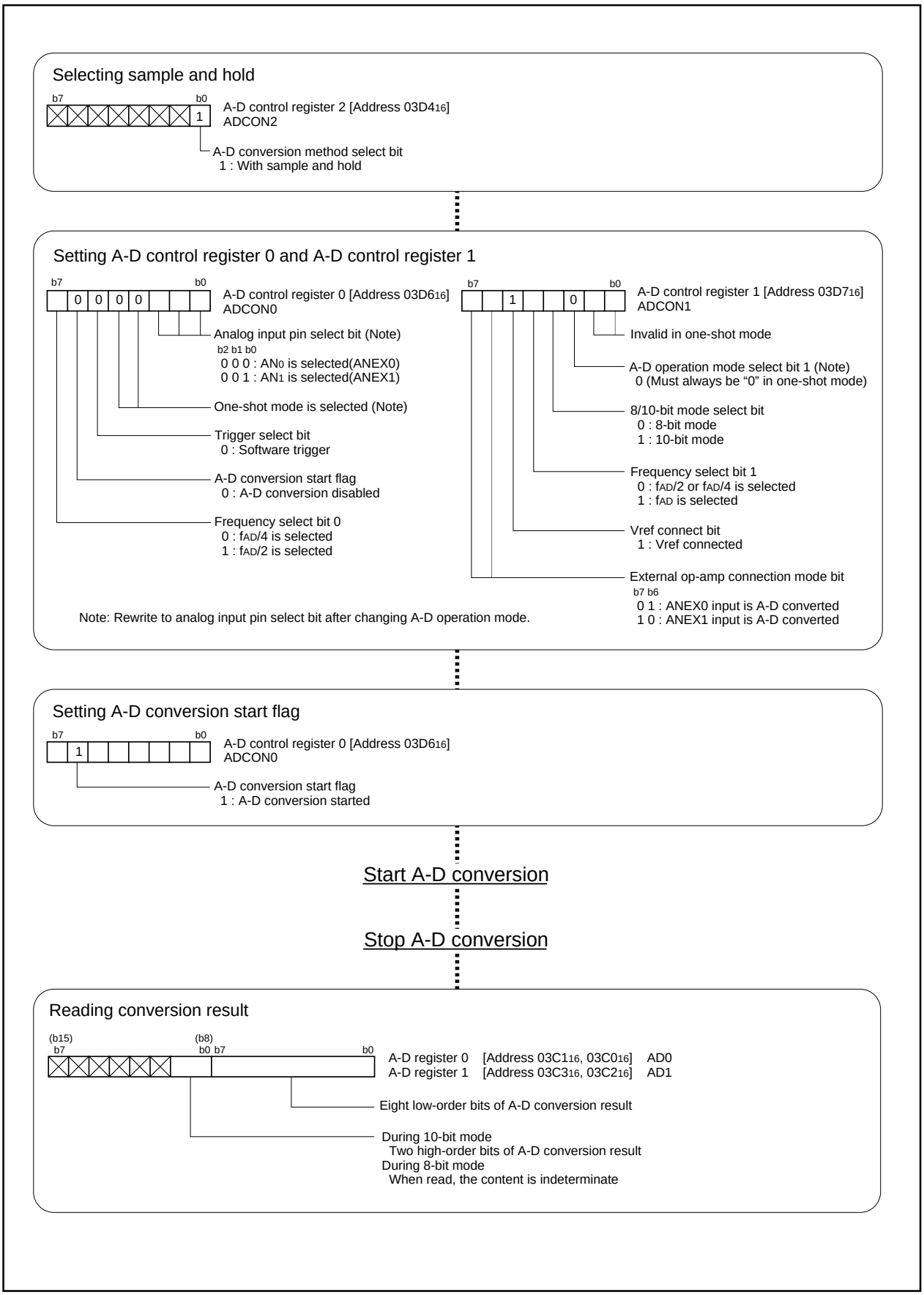


Figure 2.7.10. Set-up procedure of one-shot mode, with expanded analog input pin selected

2.7.5 Operation of A-D Converter (in one-shot mode, external op-amp connection mode selected)

In one-shot mode, choose functions from those listed in Table 2.7.5. Operations of the circled items are described below. Figure 2.7.11 shows timing chart, and Figure 2.7.12 shows the set-up procedure.

Table 2.7.5. Choosed functions

Item	Set-up		Item	Set-up	
Operation clock ϕ_{AD}	O	Divided-by-4 f_{AD} / divided-by-2 f_{AD} / f_{AD}	Expanded analog input pin		Not used
					Either ANEX0 pin or ANEX1 pin
Resolution	O	8-bit / 10-bit		O	External operation amplifier connection mode
Analog input pin	O	One of AN_0 pin to AN_7 pin			
Trigger for starting A-D conversion	O	Software trigger	Sample & Hold		Not activated
		Trigger by $\overline{ADTRG}$		O	Activated

Operation (1) Setting the A-D conversion start flag to “1” causes voltage input to the ANi pin to be output from the ANEX0 pin. The A-D conversion is carried out on voltage input to the ANEX1 pin (connect an operation amplifier between the ANEX0 pin and the ANEX1 pin).

(2) After the A-D conversion is completed, the content of the successive comparison register (conversion result) is transmitted to A-D register i corresponding to the ANi pin. At this time, the A-D conversion interrupt request bit goes to “1”.

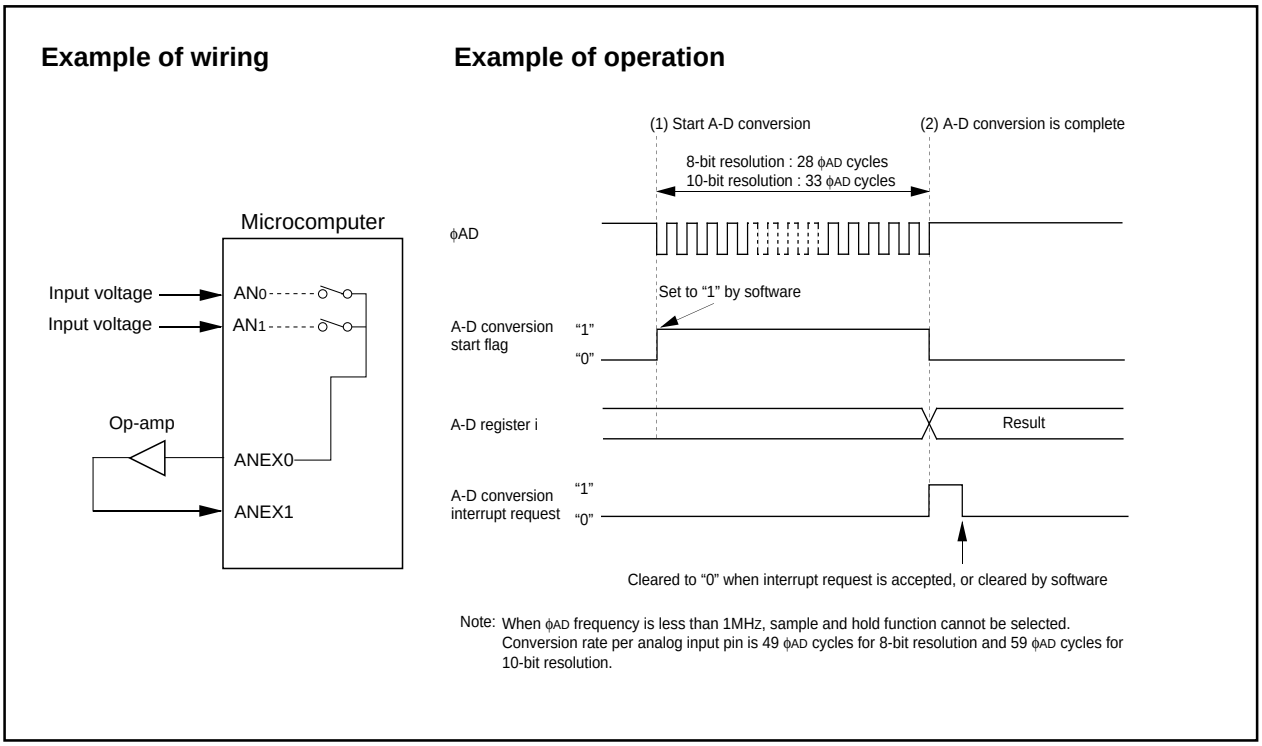


Figure 2.7.11. Operation timing of one-shot mode, with external op-amp connection mode selected

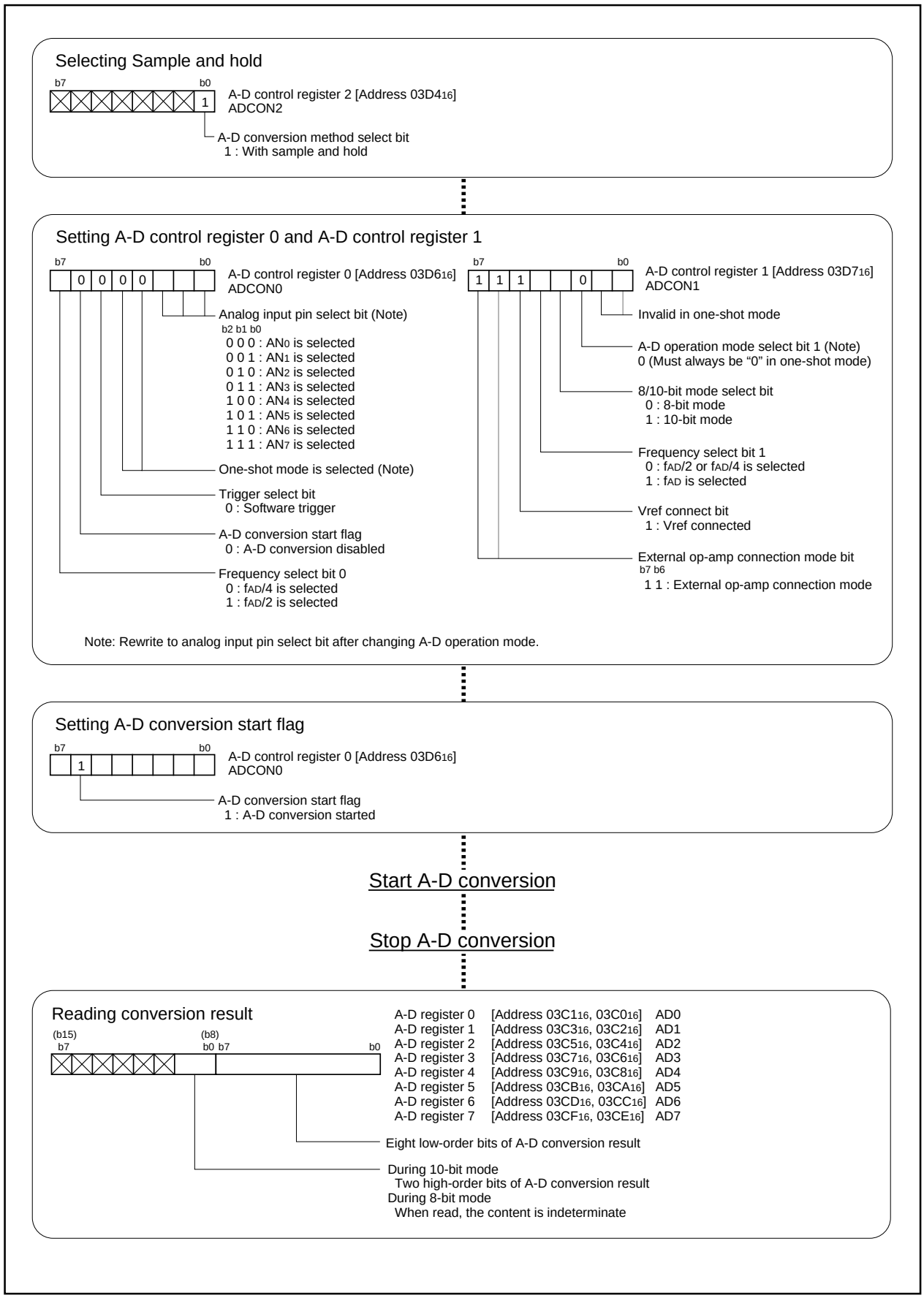


Figure 2.7.12. Set-up procedure of one-shot mode, with external op-amp connection mode selected

2.7.6 Operation of A-D Converter (in repeat mode)

In repeat mode, choose functions from those listed in Table 2.7.6. Operations of the circled items are described below. Figure 2.7.13 shows timing chart, and Figure 2.7.14 shows the set-up procedure.

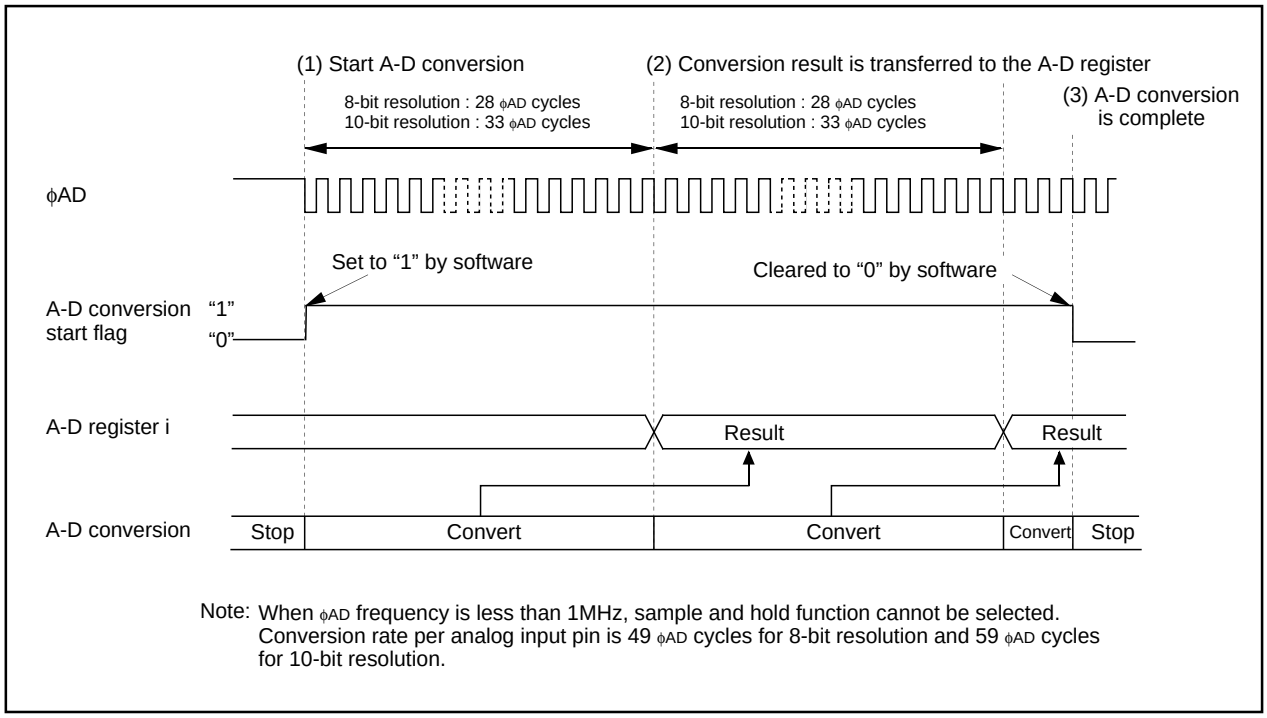
Table 2.7.6. Choosed functions

Item	Set-up		Item	Set-up	
Operation clock ϕ_{AD}	O	Divided-by-4 f_{AD} / divided-by-2 f_{AD} / f_{AD}	Expanded analog input pin	O	Not used
					Either ANEX0 pin or ANEX1 pin
Resolution	O	8-bit / 10-bit			External operation amplifier connection mode
Analog input pin	O	One of AN_0 pin to AN_7 pin			
Trigger for starting A-D conversion	O	Software trigger	Sample & Hold		Not activated
		Trigger by $\overline{ADTRG}$		O	Activated

Operation (1) Setting the A-D conversion start flag to “1” causes the A-D converter to start operating.

(2) After the first conversion is completed, the content of the successive comparison register (conversion result) is transmitted to A-D register i. The A-D conversion interrupt request bit does not go to “1”.

(3) The A-D converter continues operating until the A-D conversion start flag is set to “0” by software. The conversion result is transmitted to A-D register i every time a conversion is completed.



Note: When ϕ_{AD} frequency is less than 1MHz, sample and hold function cannot be selected. Conversion rate per analog input pin is 49 ϕ_{AD} cycles for 8-bit resolution and 59 ϕ_{AD} cycles for 10-bit resolution.

Figure 2.7.13. Operation timing of repeat mode

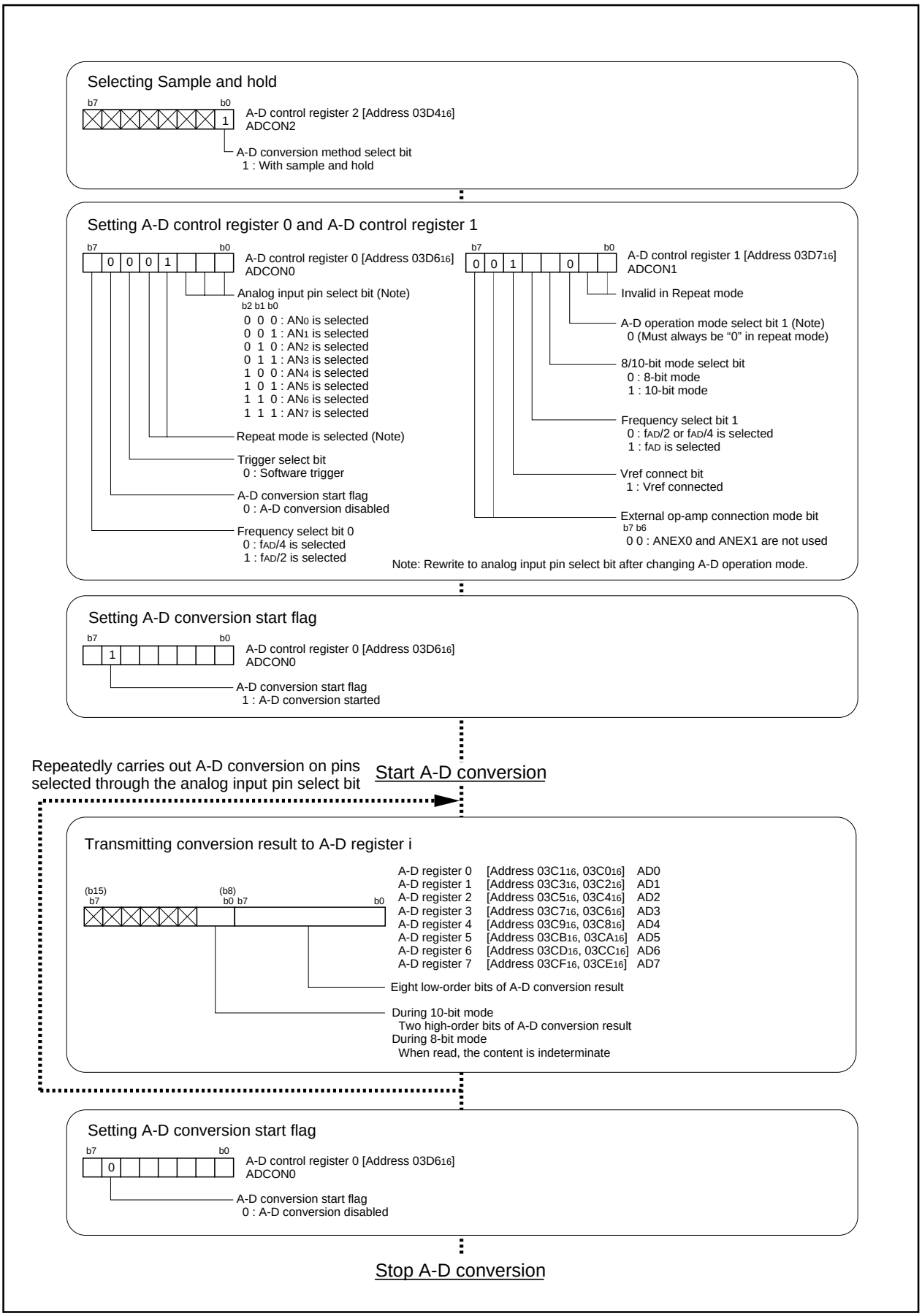


Figure 2.7.14. Set-up procedure of repeat mode

2.7.7 Operation of A-D Converter (in single sweep mode)

In single sweep mode, choose functions from those listed in Table 2.7.7. Operations of the circled items are described below. Figure 2.7.15 shows timing chart, and Figure 2.7.16 shows the set-up procedure.

Table 2.7.7. Choosed functions

Item	Set-up		Item	Set-up	
Operation clock f_{AD}	O	Divided-by-4 f_{AD} / divided-by-2 f_{AD} / f_{AD}	Trigger for starting A-D conversion	O	Software trigger
					Trigger by $\overline{ADTRG}$
Resolution	O	8-bit / 10-bit	Expanded analog input pin	O	Not used
Analog input pin	O	AN_0 and AN_1 (2 pins) / AN_0 to AN_3 (4 pins) / AN_0 to AN_5 (6 pins) / AN_0 to AN_7 (8 pins)			External ope-amp connection mode
			Sample & Hold		Not activated
				O	Activated

Operation (1) Setting the A-D conversion start flag to “1” causes the A-D converter to start the conversion on voltage input to the AN_0 pin.

(2) After the A-D conversion of voltage input to the AN_0 pin is completed, the content of the successive comparison register (conversion result) is transmitted to A-D register 0. The A-D converter converts all analog input pins selected by the user. The conversion result is transmitted to A-D register i corresponding to each pin, every time conversion on one pin is completed.

(3) When the A-D conversion on all the analog input pins selected is completed, the A-D conversion interrupt request bit goes to “1”. At this time, the A-D conversion start flag goes to “0”. The A-D converter stops operating.

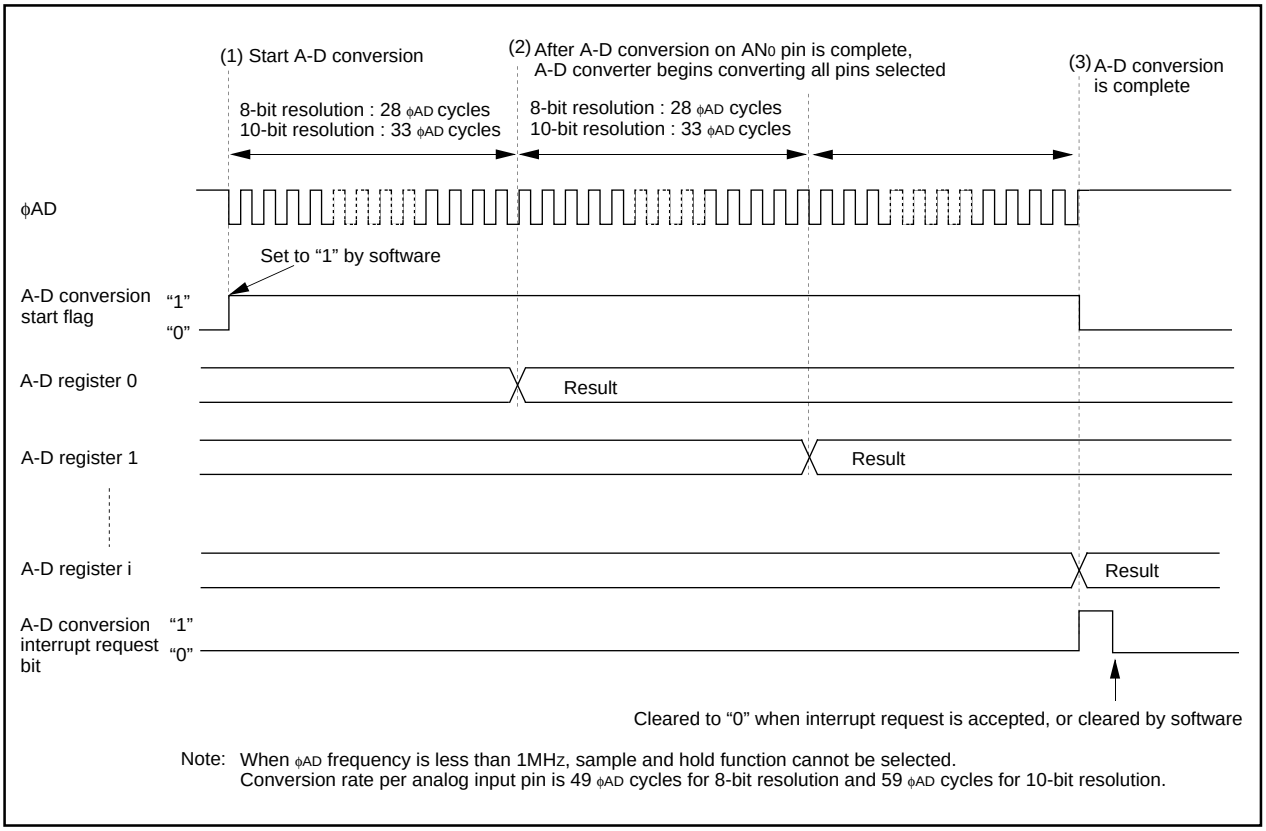


Figure 2.7.15. Operation timing of single sweep mode

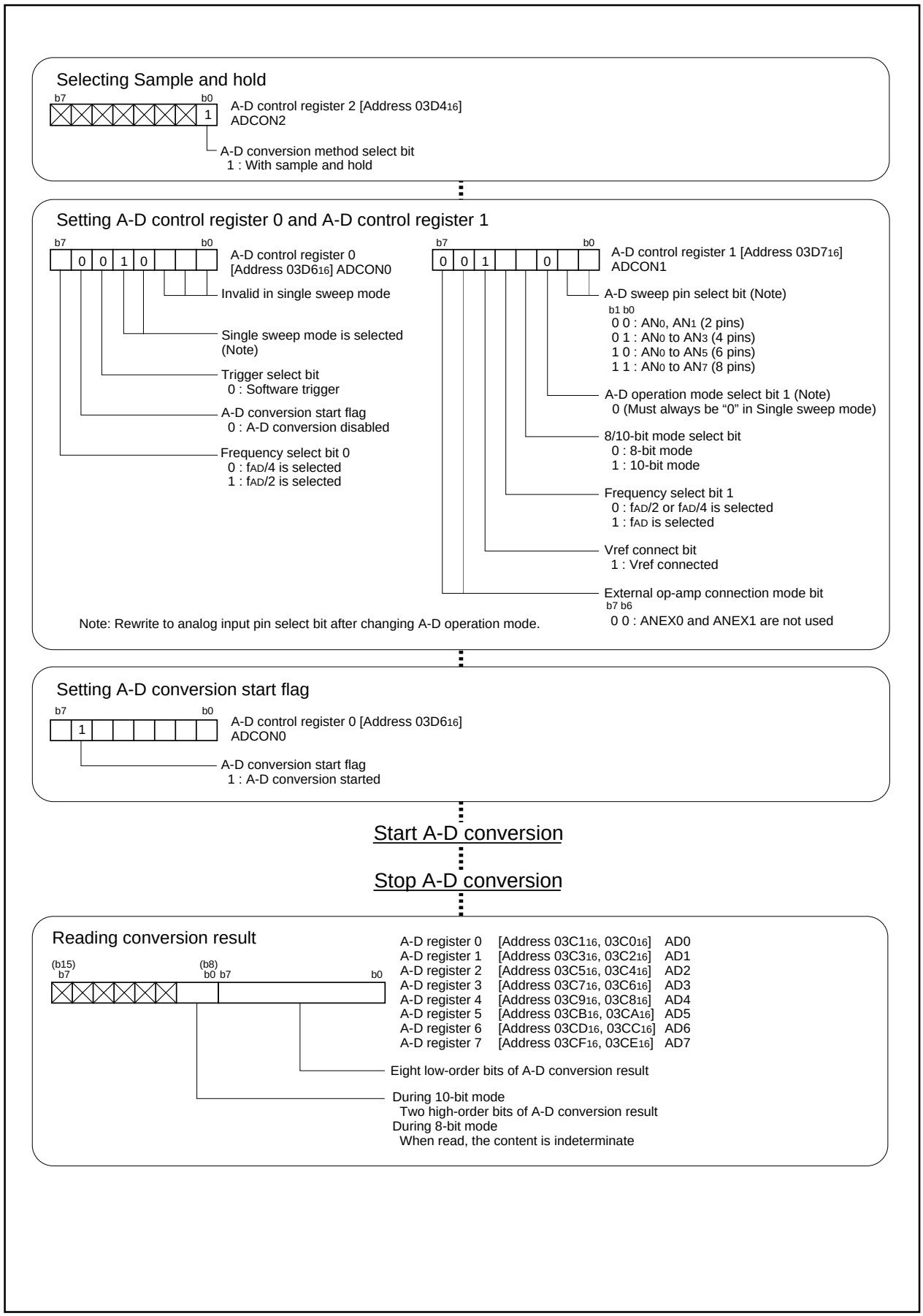


Figure 2.7.16. Set-up procedure of single sweep mode

2.7.8 Operation of A-D Converter (in repeat sweep mode 0)

In repeat sweep 0 mode, choose functions from those listed in Table 2.7.8. Operations of the circled items are described below. Figure 2.7.17 shows timing chart, and Figure 2.7.18 shows the set-up procedure.

Table 2.7.8. Choosed functions

Item	Set-up		Item	Set-up	
Operation clock f_{AD}	**O**	Divided-by-4 f_{AD} / divided-by-2 f_{AD} / f_{AD}	Trigger for starting A-D conversion	**O**	Software trigger
					Trigger by $\overline{ADTRG}$
Resolution	**O**	8-bit / 10-bit	Expanded analog input pin	**O**	Not used
Analog input pin	**O**	AN_0 and AN_1 (2 pins) / AN_0 to AN_3 (4 pins) / AN_0 to AN_5 (6 pins) / AN_0 to AN_7 (8 pins)			External ope-amp connection mode
			Sample & Hold		Not activated
				O	Activated

Operation (1) Setting the A-D conversion start flag to “1” causes the A-D converter to start the conversion on voltage input to the AN_0 pin.

(2) After the A-D conversion of voltage input to the AN_0 pin is completed, the content of the successive comparison register (conversion result) is transmitted to A-D register 0.

(3) The A-D converter converts all pins selected by the user. The conversion result is transmitted to A-D register i corresponding to each pin every time A-D conversion on the pin is completed. The A-D conversion interrupt request bit does not go to “1”.

(4) The A-D converter continues operating until the A-D conversion start flag is set to “0” by software.

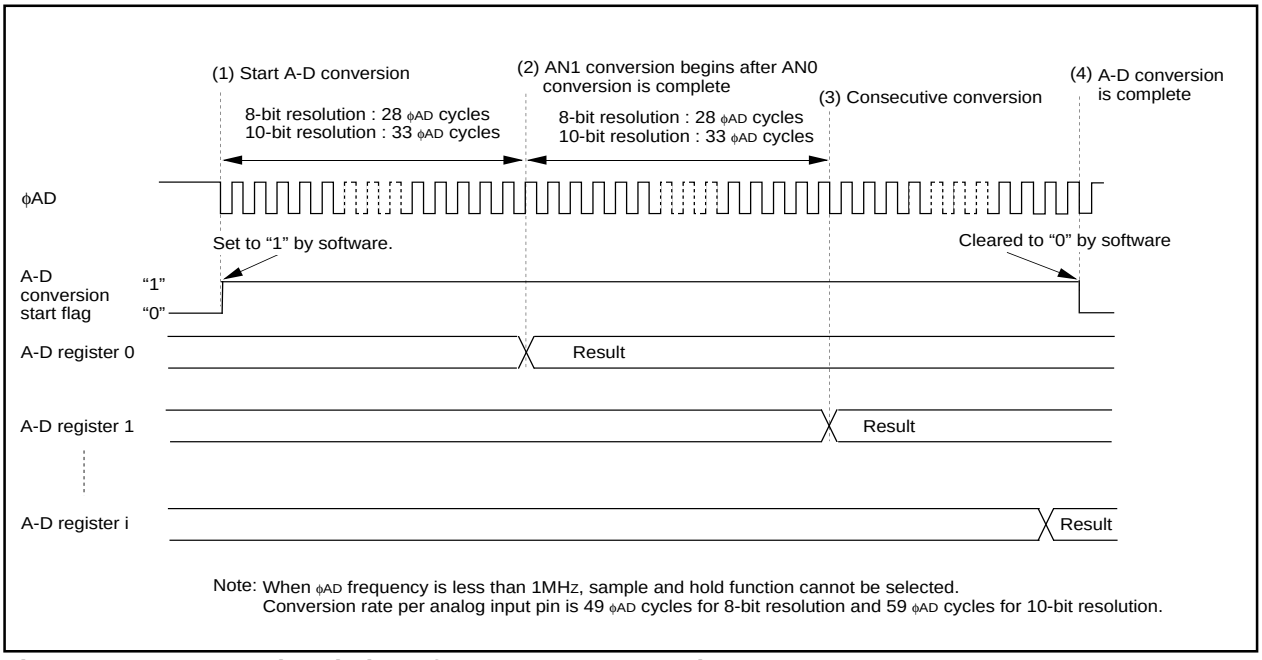


Figure 2.7.17. Operation timing of repeat sweep 0 mode

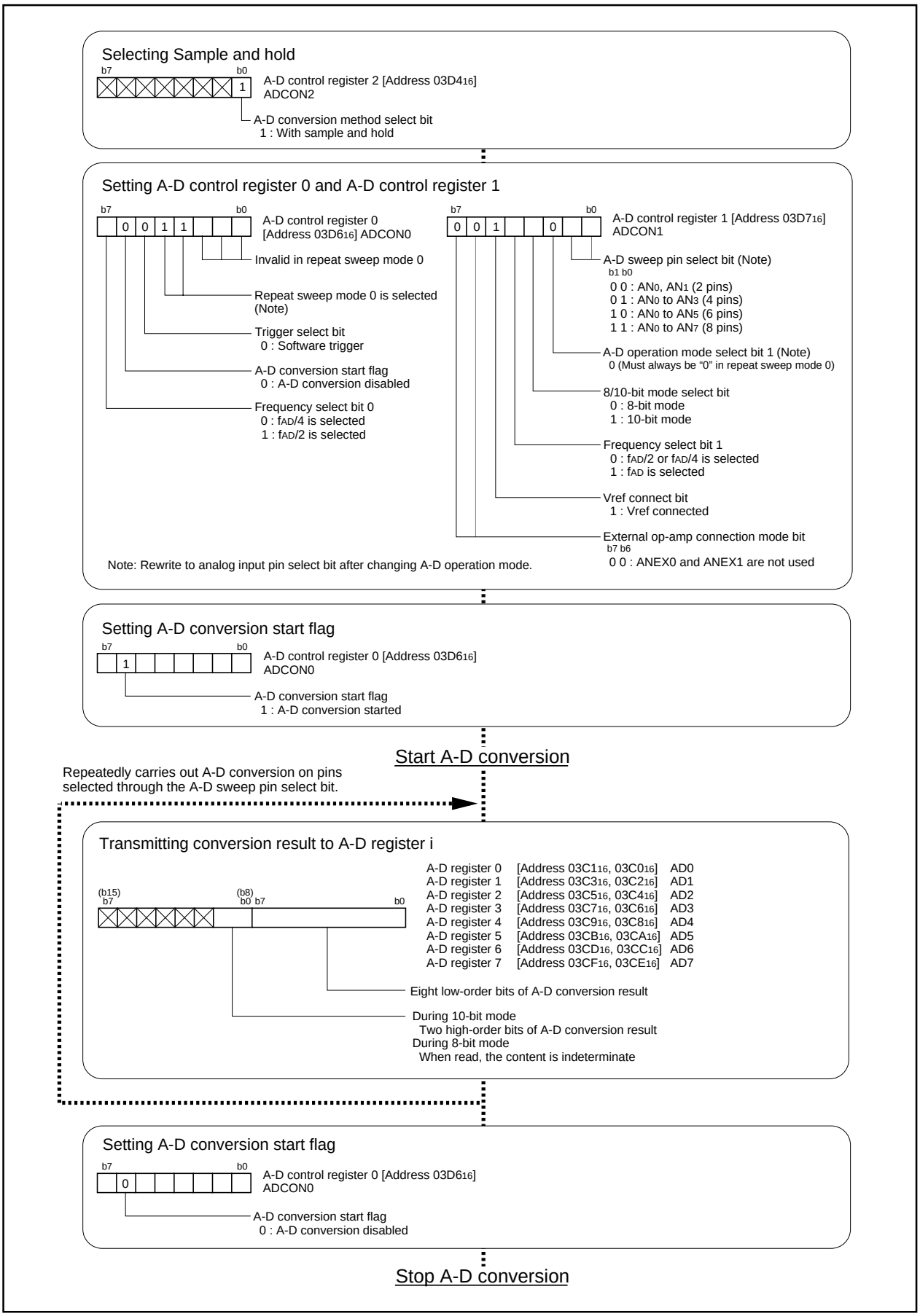


Figure 2.7.18. Set-up procedure of repeat sweep 0 mode

2.7.9 Operation of A-D Converter (in repeat sweep mode 1)

In repeat sweep 1 mode, choose functions from those listed in Table 2.7.9. Operations of the circled items are described below. Figure 2.7.19 shows ANi pin's sweep sequence, Figure 2.7.20 shows timing chart, and Figure 2.7.21 shows the set-up procedure.

Table 2.7.9. Choosed functions

Item	Set-up		Item	Set-up	
Operation clock ϕ_{AD}	O	Divided-by-4 f_{AD} / divided-by-2 f_{AD} / f_{AD}	Trigger for starting A-D conversion	O	Software trigger
					Trigger by $\overline{ADTRG}$
Resolution	O	8-bit / 10-bit	Expanded analog input pin	O	Not used
Analog input pin	O	AN_0 (1 pin) / AN_0 and AN_1 (2 pins) / AN_0 to AN_2 (3 pins) / AN_0 to AN_3 (4 pins)			External ope-amp connection mode
			Sample & Hold		Not activated
				O	Activated

Operation (1) Setting the A-D conversion start flag to “1” causes the A-D converter to start the conversion on voltage input to the AN_0 pin.

(2) After the A-D conversion on voltage input to the AN_0 pin is completed, the content of the successive comparison register (conversion result) is transmitted to A-D register 0.

(3) Every time the A-D converter carries out A-D conversion on a selected analog input pin, the A-D converter carries out A-D conversion on only one unselected pin, and then the A-D converter carries out A-D conversion from the AN0 pin again. (See Figure 2.7.19.) The conversion result is transmitted to A-D register i every time conversion on a pin is completed. The A-D conversion interrupt request bit does not go to “1”.

(4) The A-D converter continues operating until software goes the A-D conversion start flag to “0”.

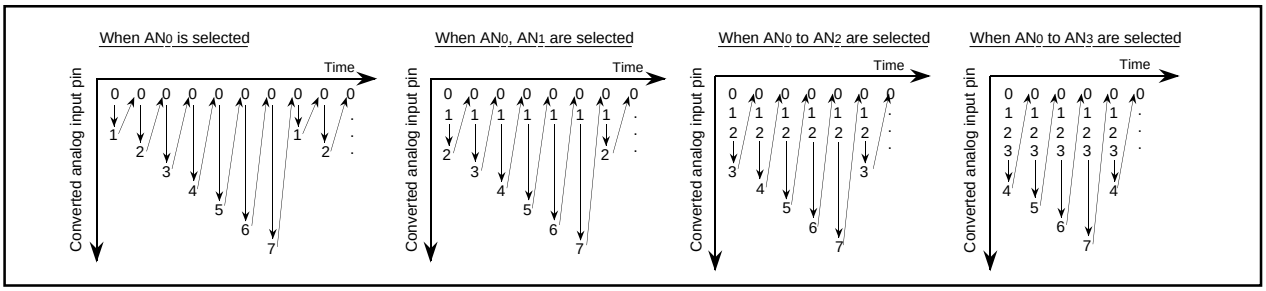


Figure 2.7.19. ANi pin's sweep sequence in repeat sweep mode

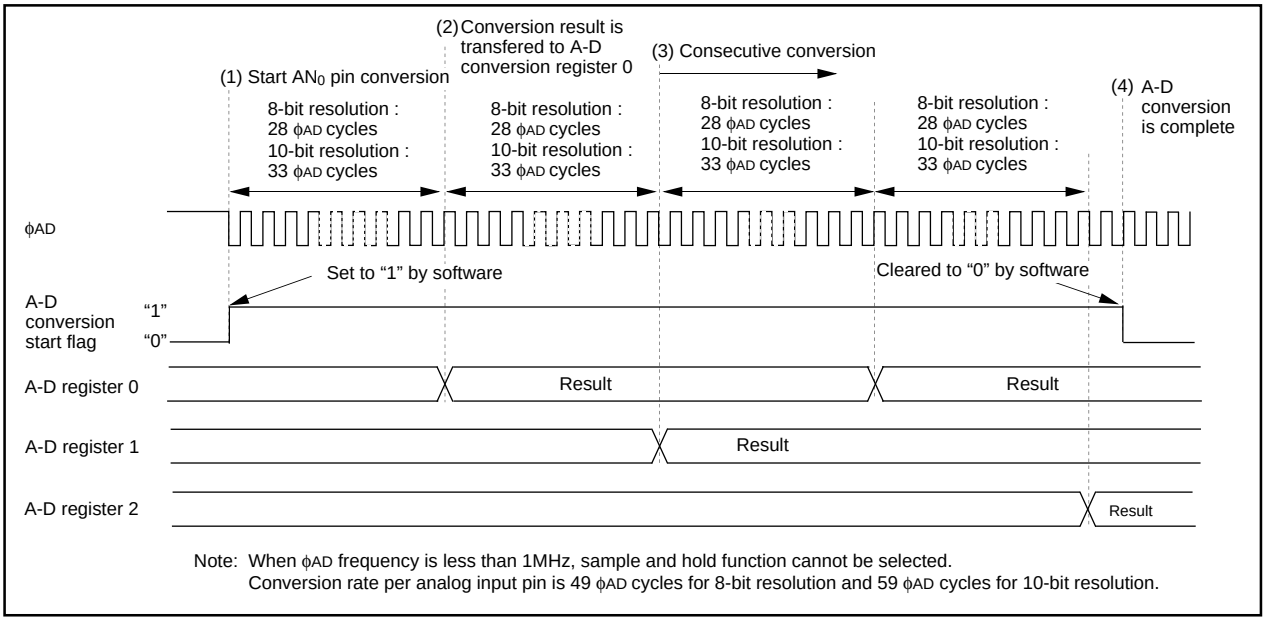


Figure 2.7.20. Operation timing of repeat sweep 1 mode

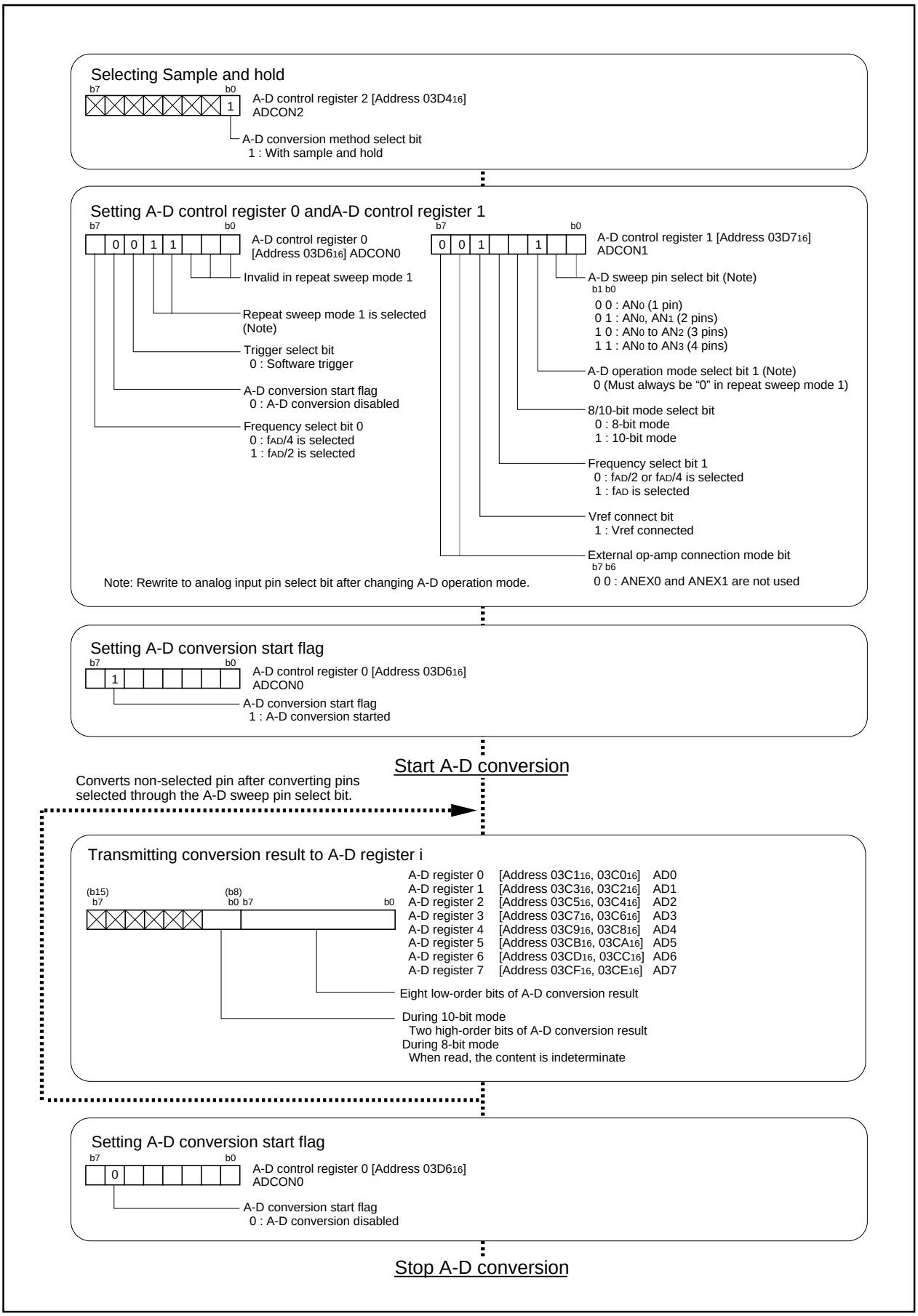


Figure 2.7.21. Set-up procedure of repeat sweep 1 mode

2.7.10 Precautions for A-D Converter

(1) Write to each bit (except bit 6) of A-D control register 0, to each bit of A-D control register 1, and to bit 0 of A-D control register 2 when A-D conversion is stopped (before a trigger occurs). In particular, when the Vref connection bit is changed from 0 to 1, start A-D conversion after an elapse of 1 μs or longer.

(2) To reduce conversion error due to noise, connect a voltage to the AVcc pin and to the Vref pin from an independent source. It is recommended to connect a capacitor between the AVss pin and the AVcc pin, between the AVss pin and the Vref pin, and between the AVss pin and the analog input pin (ANi). Figure 2.7.22 shows the an example of connecting the capacitors to these pins.

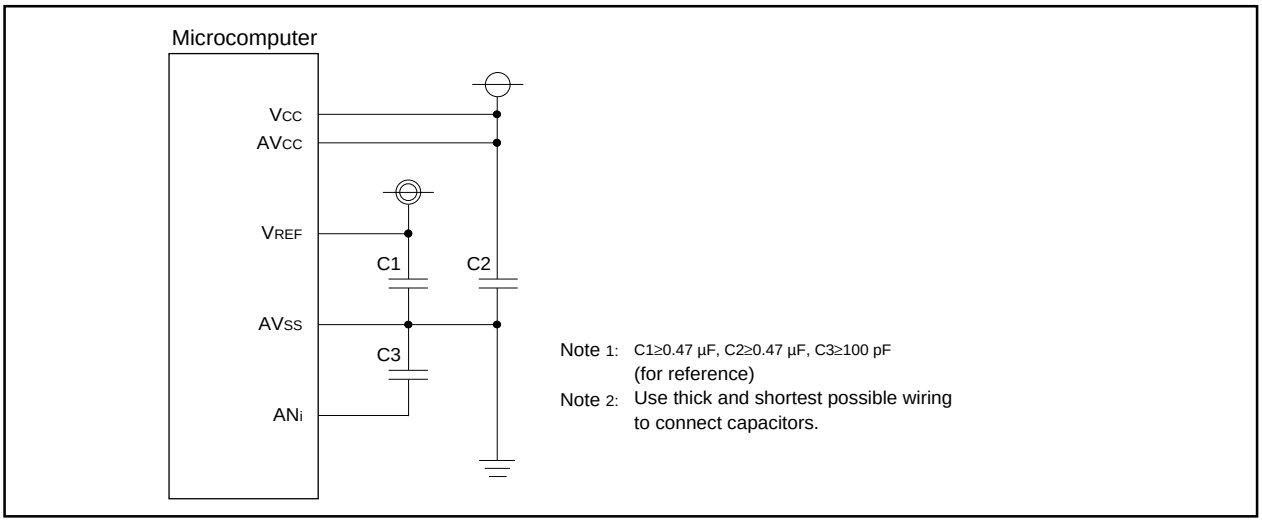


Figure 2.7.22. Use of capacitors to reduce noise

(3) Set the direction register of the following ports to input: the port corresponding to a pin to be used as an analog input pin and external trigger input pin (P97).

(4) In using a key-input interrupt, none of the 4 pins (AN4 through AN7) can be used as an A-D conversion port (if the A-D input voltage goes to “L” level, a key-input interrupt occurs).

(5) If using the A-D converter with Vcc = 2.7V to 4.0 V:
Use only a divided frequency for fAD (undivided fAD is not allowed).
Select without the Sample & Hold feature.
Select 8-bit mode.

(6) Rewrite to analog input pin select bits after changing A-D operation mode.

(7) When using the one-shot or single sweep mode
Confirm that A-D conversion is complete before reading the A-D register.
(Note: When A-D conversion interrupt request bit is set, it shows that A-D conversion is completed.)

(8) When using the repeat mode or repeat sweep mode 0 or 1
Use the undivided main clock as the internal CPU clock.

(9) Use ϕAD under 10 MHz. When XIN is over 10 MHz, divide it.

2.7.11 Method of A-D Conversion (10-bit mode)

(1) The A-D converter compares the reference voltage (Vref) generated internally based on the contents of the successive comparison register with the analog input voltage (V_{IN}) input from the analog input pin. Each bit of the comparison result is stored in the successive comparison register until analog-to-digital conversion (successive comparison method) is complete. If a trigger occurs, the A-D converter carries out the following:

1. Fixes bit 9 of the successive comparison register.
   Compares Vref with V_{IN}: [In this instance, the contents of the successive comparison register are "1000000000_2" (default).]
   Bit 9 of the successive comparison register varies depending on the comparison result as follows.
   If Vref < V_{IN}, then "1" is assigned to bit 9.
   If Vref > V_{IN}, then "0" is assigned to bit 9.

2. Fixes bit 8 of the successive comparison register.
   Sets bit 8 of the successive comparison register to "1", then compares Vref with V_{IN}.
   Bit 8 of the successive comparison register varies depending on the comparison result as follows:
   If Vref < V_{IN}, then "1" is assigned to bit 8.
   If Vref > V_{IN}, then "0" is assigned to bit 8.

3. Fixes bit 7 through bit 0 of the successive comparison register.
   Carries out step 2 above on bit 7 through bit 0.
   After bit 0 is fixed, the contents of the successive comparison register (conversion result) are transmitted to A-D register i.

Vref is generated based on the latest content of the successive comparison register. Table 2.7.10 shows the relationship of the successive comparison register contents and Vref. Table 2.7.11 shows how the successive comparison register and Vref vary while A-D conversion is in progress. Figure 2.7.23 shows theoretical A-D conversion characteristics.

Table 2.7.10. Relationship of the successive comparison register contents and Vref

Successive approximation register : n	Vref (V)
0	0
1 to1023	$\frac{V_{REF}}{1024} \times n - \frac{V_{REF}}{2048}$

Table 2.7.11. Variation of the successive comparison register and Vref while A-D conversion is in progress (10-bit mode)

	Successive approximation register	Vref change
A-D converter stopped	b9 b0: 1 0 0 0 0 0 0 0 0 0	$\frac{V_{REF}}{2}$ [V]
1st comparison	1 0 0 0 0 0 0 0 0 0	$\frac{V_{REF}}{2} - \frac{V_{REF}}{2048}$ [V]
2nd comparison	n9 1 0 0 0 0 0 0 0 0 (1st comparison result)	$\frac{V_{REF}}{2} \pm \frac{V_{REF}}{4} - \frac{V_{REF}}{2048}$ [V] ($n_9 = 1$: $+\frac{V_{REF}}{4}$; $n_9 = 0$: $-\frac{V_{REF}}{4}$)
3rd comparison	n9 n8 1 0 0 0 0 0 0 0 (2nd comparison result)	$\frac{V_{REF}}{2} \pm \frac{V_{REF}}{4} \pm \frac{V_{REF}}{8} - \frac{V_{REF}}{2048}$ [V] ($n_8 = 1$: $+\frac{V_{REF}}{8}$; $n_8 = 0$: $-\frac{V_{REF}}{8}$)
10th comparison	n9 n8 n7 n6 n5 n4 n3 n2 n1 0	$\frac{V_{REF}}{2} \pm \frac{V_{REF}}{4} \pm \frac{V_{REF}}{8} \pm \ldots\ldots \pm \frac{V_{REF}}{1024} - \frac{V_{REF}}{2048}$ [V]
Conversion complete	n9 n8 n7 n6 n5 n4 n3 n2 n1 n0 (This data transfers to the bit 0 to bit 9 of A-D register i.)	

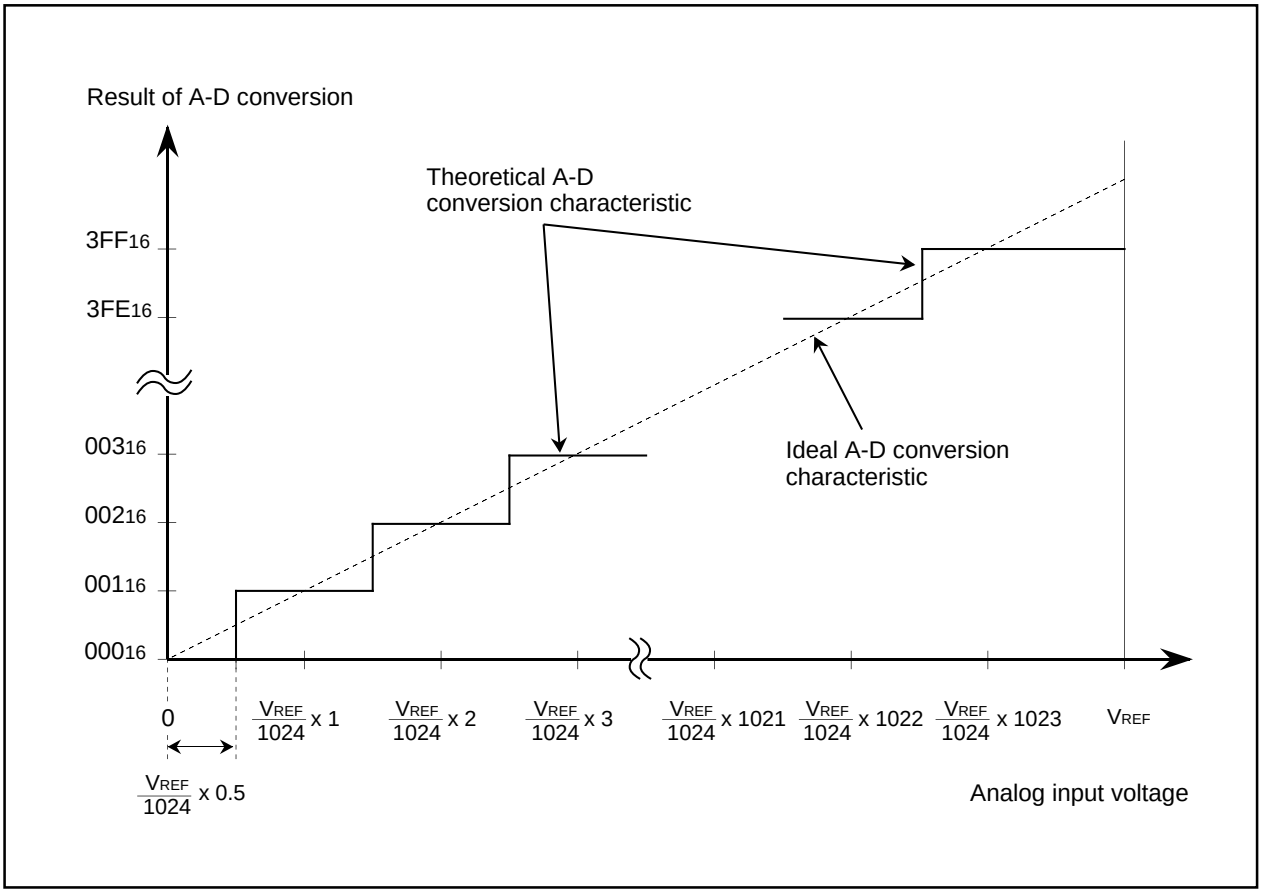


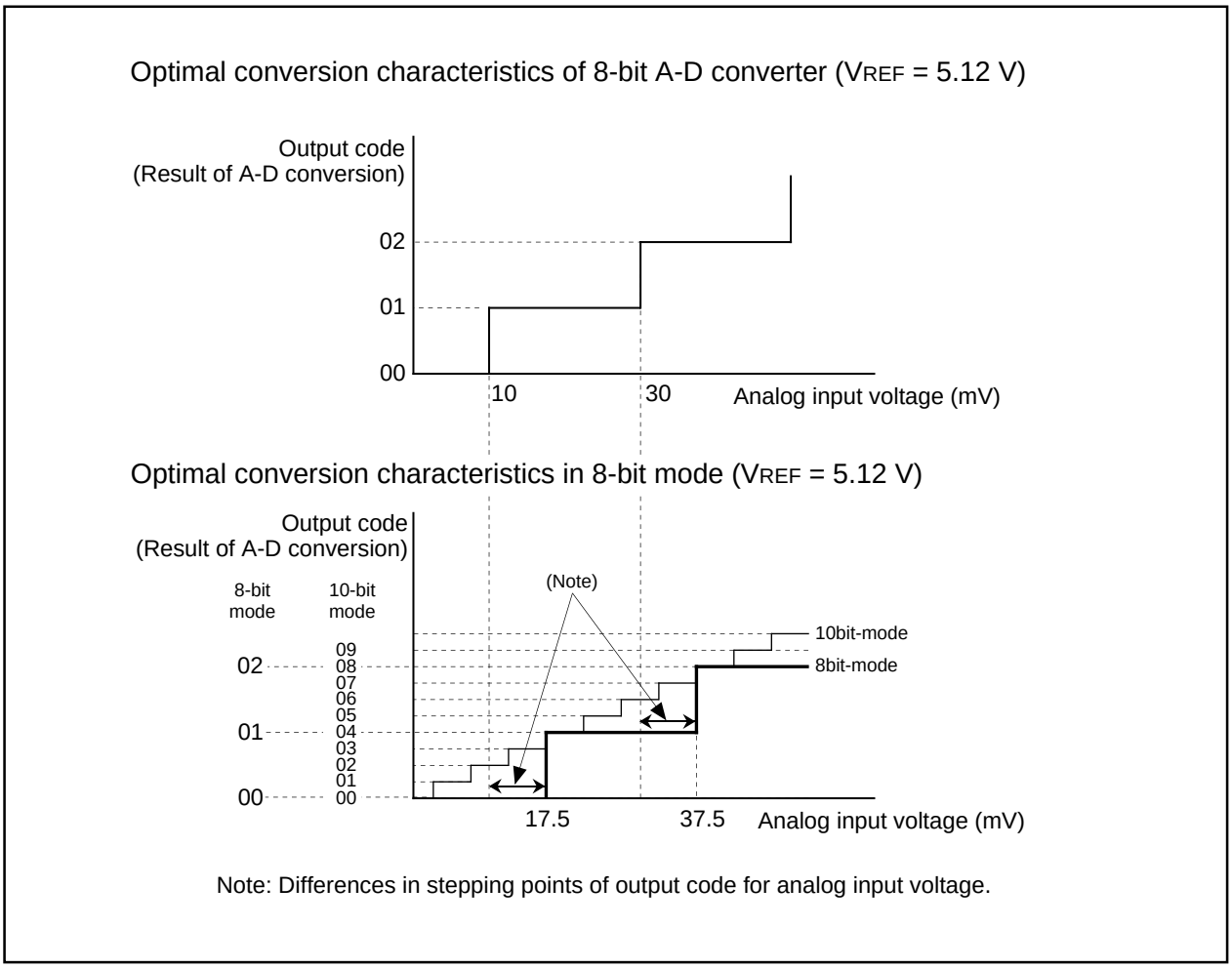
Figure 2.7.23. Theoretical A-D conversion characteristics (10-bit mode)

2.7.12 Method of A-D Conversion (8-bit mode)

(1) In 8-bit mode, 8 higher-order bits of the 10-bit successive comparison register becomes A-D conversion result. Hence, if compared to a result obtained by using an 8-bit A-D converter, the voltage compared is different by 3 VREF/2048 (see what are underscored in Table 2.7.12), and differences in stepping points of output codes occur as shown in Figure 2.7.24.

Table 2.7.12. The comparison voltage in 8-bit mode compared to 8-bit A-D converter

		8-bit mode	8-bit A-D converter
Comparison voltage Vref	n = 0	0	0
	n = 1 to 255	$\frac{VREF}{2^8} \times n - \underline{\frac{VREF}{2^{10}} \times 0.5}$	$\frac{VREF}{2^8} \times n - \underline{\frac{VREF}{2^8} \times 0.5}$



Note: Differences in stepping points of output code for analog input voltage.

Figure 2.7.24. The level conversion characteristics of 8-bit mode and 8-bit A-D converter

Table 2.7.13. Variation of the successive comparison register and Vref while A-D conversion is in progress (8-bit mode)

	Successive approximation register	Vref change
A-D converter stopped	b9 b0 1 0 0 0 0 0 0 0 0 0	$\frac{V_{REF}}{2}$ [V]
1st comparison	1 0 0 0 0 0 0 0 0 0	$\frac{V_{REF}}{2} - \frac{V_{REF}}{2048}$ [V]
2nd comparison	n9 1 0 0 0 0 0 0 0 0 1st comparison result	$\frac{V_{REF}}{2} \pm \frac{V_{REF}}{4} - \frac{V_{REF}}{2048}$ [V] ($n_9 = 1$: $+\frac{V_{REF}}{4}$; $n_9 = 0$: $-\frac{V_{REF}}{4}$)
3rd comparison	n9 n8 1 0 0 0 0 0 0 0 2nd comparison result	$\frac{V_{REF}}{2} \pm \frac{V_{REF}}{4} \pm \frac{V_{REF}}{8} - \frac{V_{REF}}{2048}$ [V] ($n_8 = 1$: $+\frac{V_{REF}}{8}$; $n_8 = 0$: $-\frac{V_{REF}}{8}$)
	⋮	⋮
8th comparison	n9 n8 n7 n6 n5 n4 n3 1 0 0	$\frac{V_{REF}}{2} \pm \frac{V_{REF}}{4} \pm \frac{V_{REF}}{8} \pm ..... \pm \frac{V_{REF}}{256} - \frac{V_{REF}}{2048}$ [V]
Conversion complete	n9 n8 n7 n6 n5 n4 n3 n2 0 0 This data transfers to bit 0 to bit 7 of A-D register i.	

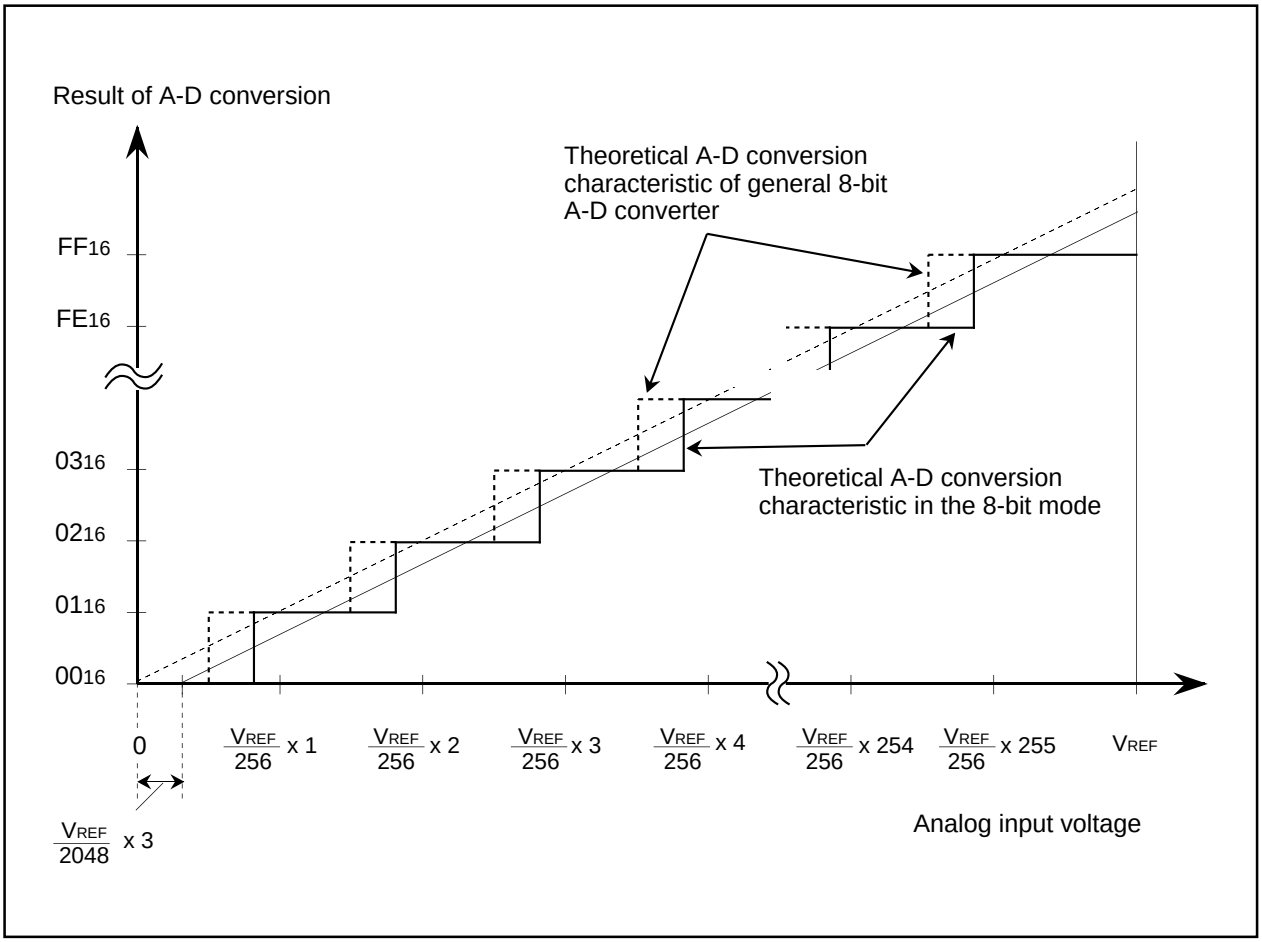


Figure 2.7.25. Theoretical A-D conversion characteristics (8-bit mode)

2.7.13 Absolute Accuracy and Differential Non-Linearity Error

• Absolute accuracy

Absolute accuracy is the difference between output code based on the theoretical A-D conversion characteristics, and actual A-D conversion result. When measuring absolute accuracy, the voltage at the middle point of the width of analog input voltage (1-LSB width), that can meet the expectation of outputting an equal code based on the theoretical A-D conversion characteristics, is used as an analog input voltage. For example, if 10-bit resolution is used and if V_{REF} (reference voltage) = 5.12 V, then 1-LSB width becomes 5 mV, and 0 mV, 5 mV, 10 mV, 15 mV, 20 mV, ····are used as analog input voltages. If analog input voltage is 25 mV, "absolute accuracy = ± 3LSB" refers to the fact that actual A-D conversion falls on a range from "002_{16}" to "008_{16}" though an output code, "005_{16}", can be expected from the theoretical A-D conversion characteristics. Zero error and full-scale error are included in absolute accuracy.

Also, all the output codes for analog input voltage between V_{REF} and AVcc becomes "$3FF_{16}$".

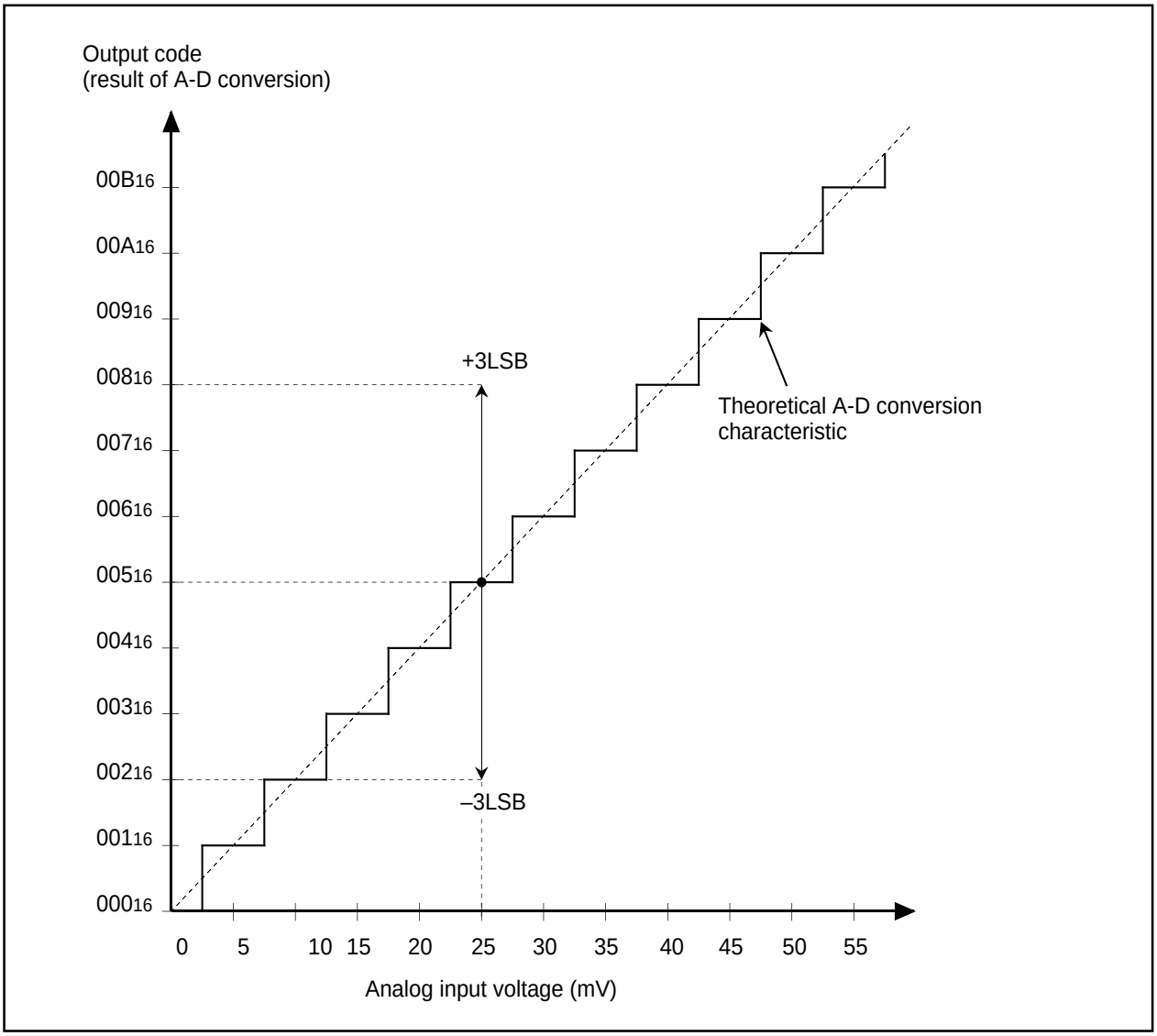


Figure 2.7.26. Absolute accuracy (10-bit resolution)

• Differential non-linearity error

Differential non-linearity error refers to the difference between 1-LSB width based on the theoretical A-D conversion characteristics (an analog input width that can meet the expectation of outputting an equal code) and an actually measured 1-LSB width (analog input voltage width that outputs an equal code). If 10-bit resolution is used and if VREF (reference voltage) = 5.12 V, “differential non-linearity error = ± 1LSB” refers to the fact that 1-LSB width actually measured falls on a range from 0 mV to 10 mV though 1-LSB width based on the theoretical A-D conversion characteristics is 5 mV.

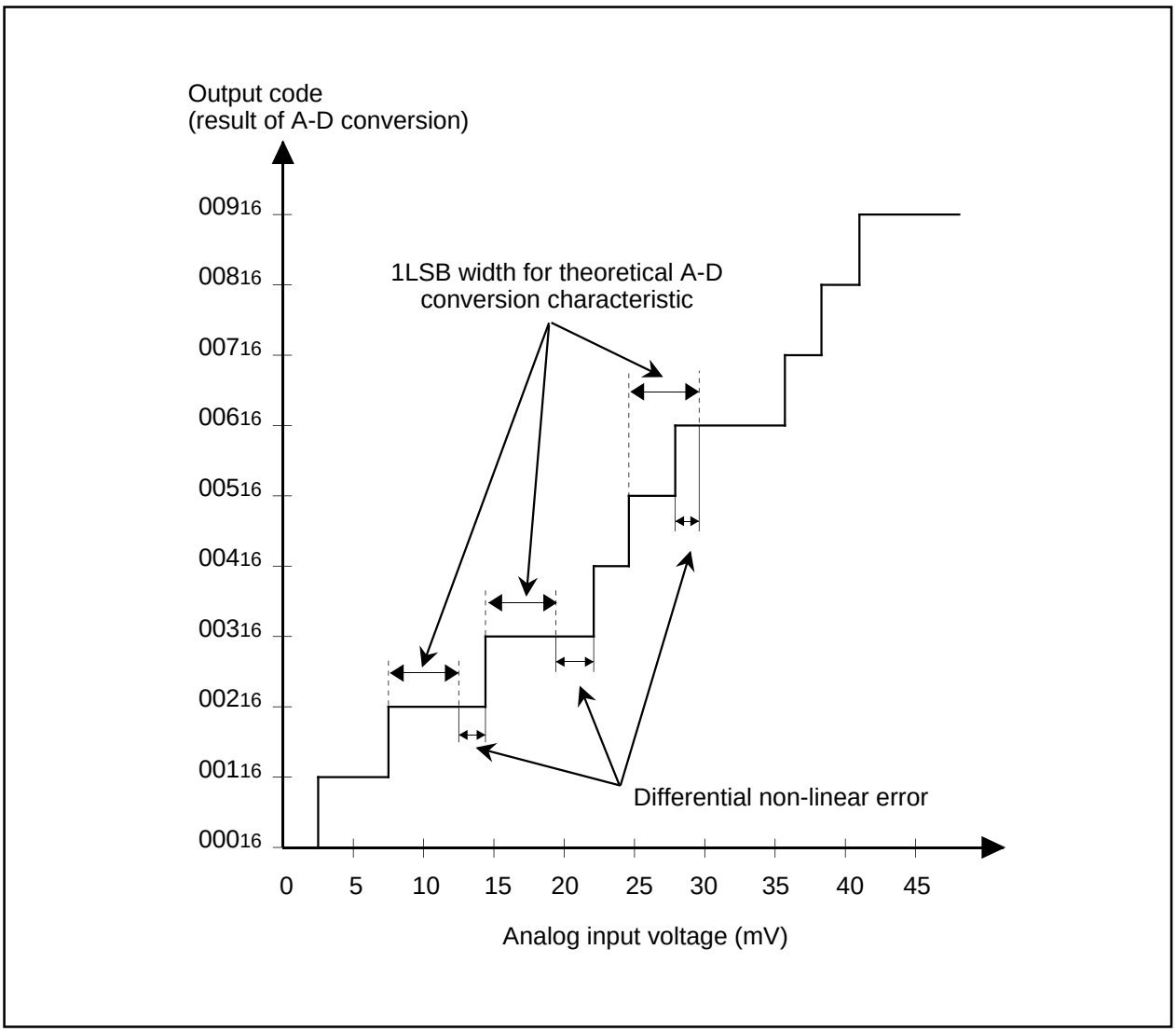
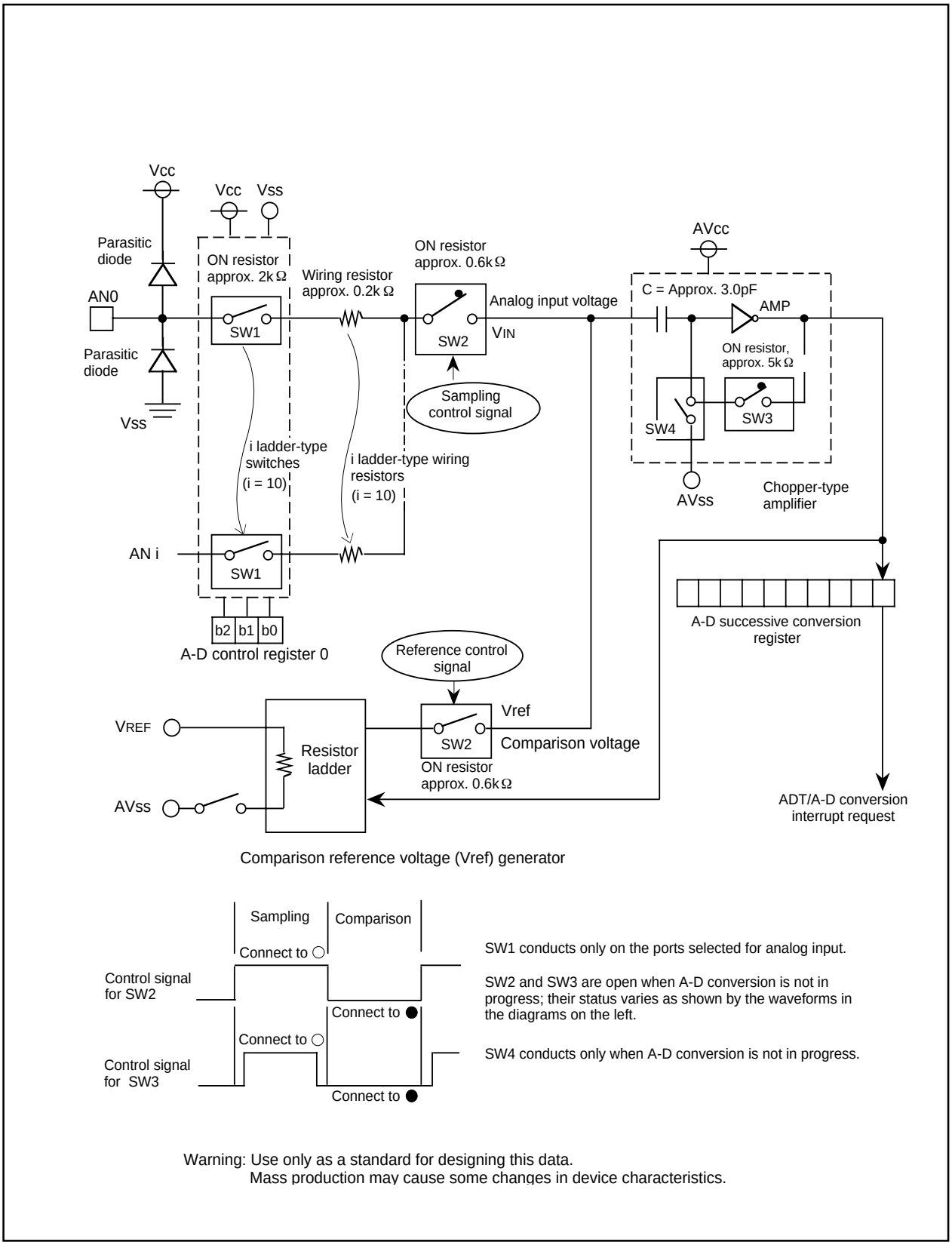


Figure 2.7.27. Differential non-linearity error (10-bit resolution)

2.7.14 Internal Equivalent Circuit of Analog Input

Figure 2.7.28 shows the internal equivalent circuit of analog input.



Warning: Use only as a standard for designing this data.
Mass production may cause some changes in device characteristics.

Figure 2.7.28. Internal equivalent circuit to analog input

2.7.15 Sensor's Output Impedance under A-D Conversion

To carry out A-D conversion properly, charging the internal capacitor C shown in Figure 2.7.29 has to be completed within a specified period of time. With T as the specified time, time T is the time that switches SW2 and SW3 are connected to O in Figure 2.7.28. Let output impedance of sensor equivalent circuit be R0, microcomputer's internal resistance be R, precision (error) of the A-D converter be X, and the A-D converter's resolution be Y (Y is 1024 in the 10-bit mode, and 256 in the 8-bit mode).

Vc is generally $V_C = V_{IN} \{1 - e^{-\frac{t}{C(R0 + R)}}\}$

And when t = T, $V_C = V_{IN} - \frac{X}{Y} V_{IN} = V_{IN}(1 - \frac{X}{Y})$

$$e^{-\frac{T}{C(R0 + R)}} = \frac{X}{Y}$$

$$-\frac{T}{C(R0 + R)} = \ln \frac{X}{Y}$$

Hence, $R0 = -\frac{T}{C \cdot \ln \frac{X}{Y}} - R$

With the model shown in Figure 2.7.29 as an example, when the difference between V_{IN} and V_C becomes 0.1LSB, we find impedance R0 when voltage between pins V_C changes from 0 to V_{IN}-(0.1/1024) V_{IN} in time T. (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 $f(X_{IN})$ = 10 MHz, T = 0.3 μs in the A-D conversion mode with sample & hold. Output impedance R0 for sufficiently charging capacitor C within time T is determined as follows.

T = 0.3 μs, R = 7.8 kΩ, C = 3 pF, X = 0.1, and Y = 1024 . Hence,

$$R0 = -\frac{0.3 \times 10^{-6}}{3.0 \times 10^{-12} \cdot \ln \frac{0.1}{1024}} - 7.8 \times 10^3 \fallingdotseq 3.0 \times 10^3$$

Thus, the allowable output impedance of the sensor circuit capable of thoroughly driving the A-D converter turns out to be approximately 3.0 kΩ. Tables 2.7.14 and 2.7.15 show output impedance values based on the LSB values.

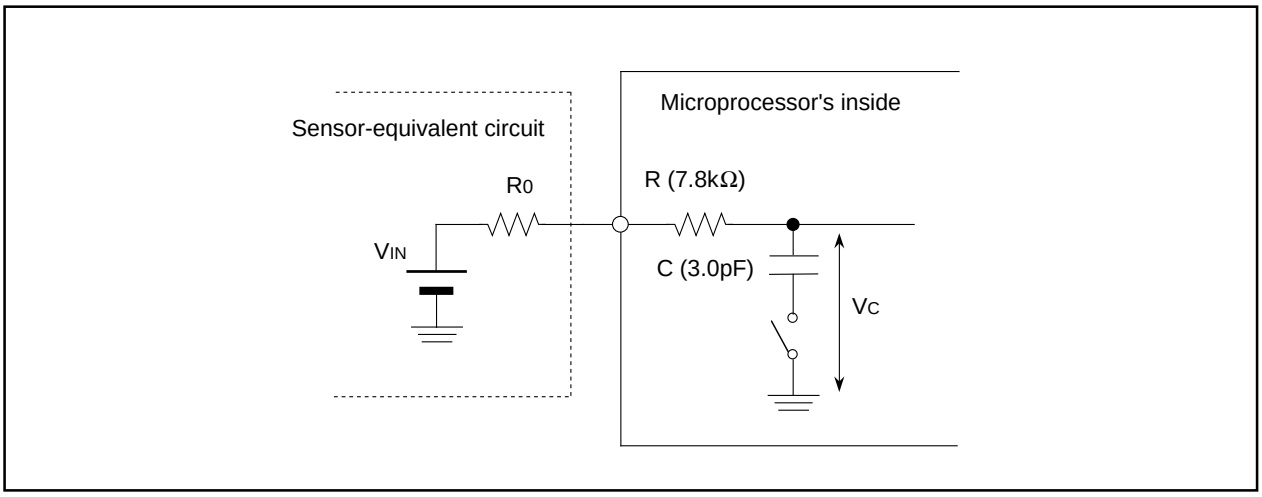


Figure 2.7.29 A circuit equivalent to the A-D conversion terminal

Tables 2.7.14. Relation between output impedance and precision (error) of A-D converter (10-bit mode) Reference value

f(Xin) (MHz)	Cycle (μs)	Sampling time (μs)	R (kΩ)	C (pF)	Resolution (LSB)	R0 (kΩ)
10	0.1	0.3 (3 x cycle, Sample & hold bit is enabled)	7.8	3.0	0.1	3.0
					0.3	4.5
					0.5	5.3
					0.7	5.9
					0.9	6.4
					1.1	6.8
					1.3	7.2
					1.5	7.5
					1.7	7.8
					1.9	8.1
10	0.1	0.2 (2 x cycle, Sample & hold bit is disabled)	7.8	3.0	0.3	0.4
					0.5	0.9
					0.7	1.3
					0.9	1.7
					1.1	2.0
					1.3	2.2
					1.5	2.4
					1.7	2.6
					1.9	2.8

Tables 2.7.15. Relation between output impedance and precision (error) of A-D converter (8-bit mode) Reference value

f(Xin) (MHz)	Cycle (μs)	Sampling time (μs)	R (kΩ)	C (pF)	Resolution (LSB)	R0 (kΩ)
10	0.1	0.3 (3 x cycle, Sample & hold bit is enabled)	7.8	3.0	0.1	4.9
					0.3	7.0
					0.5	8.2
					0.7	9.1
					0.9	9.9
					1.1	10.5
					1.3	11.1
					1.5	11.7
					1.7	12.1
					1.9	12.6
10	0.1	0.2 (2 x cycle, Sample & hold bit is disabled)	7.8	3.0	0.1	0.7
					0.3	2.1
					0.5	2.9
					0.7	3.5
					0.9	4.0
					1.1	4.4
					1.3	4.8
					1.5	5.2
					1.7	5.5
					1.9	5.8

2.8 D-A Converter Usage

2.8.1 Overview of the D-A converter usage

The D-A converter used in the M16C/62A group is based on the 8-bit R-2R technique.

(1) Output voltage

The D-A converter outputs voltage within a range from 0 V to VREF. The output voltage is determined by VREF/(256) X the D-A register contents.

The D-A converter is not effected by the Vref connection bit of the A-D converter.

(2) Conversion time

tsu = 3 µs

(3) Output from the D-A converter and the direction register

To use the D-A converter, do not set the direction register of the relevant port to output.

(4) Pins related to the D-A converter

- DA0 pin, DA1 pin Output pins of the D-A converter
- AVcc pin The power source pin of the analog section
- VREF pin Input pin of the reference voltage
- AVss pin The GND pin of the analog section

(5) Registers related to the D-A converter

Figure 2.8.1 shows the memory map of D-A converter-related registers, and Figure 2.8.2 shows D-A converter-related registers.

(6) Note

D-A output pins shared with P93 and P94. The two pins are input ports and floating at the reset.

03D8$_{16}$	D-A register 0 (DA0)
03D9$_{16}$	
03DA$_{16}$	D-A register 1 (DA1)
03DB$_{16}$	
03DC$_{16}$	D-A control register (DACON)

Figure 2.8.1. Memory map of D-A converter-related registers

D-A control register

b7 b6 b5 b4 b3 b2 b1 b0

Symbol: DACON Address: 03DC$_{16}$ When reset: 00$_{16}$

Bit symbol	Bit name	Function	R	W
DA0E	D-A0 output enable bit	0 : Output disabled 1 : Output enabled	○	○
DA1E	D-A1 output enable bit	0 : Output disabled 1 : Output enabled	○	○
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be “0”			–	–

D-A register

b7 b0

Symbol: DAi (i = 0,1) Address: 03D8$_{16}$, 03DA$_{16}$ When reset: Indeterminate

Function	R	W
Output value of D-A conversion	○	○

Figure 2.8.2. D-A converter-related registers

2.8.2 D-A Converter Operation

The following is the D-A converter operation. Figure 2.8.3 shows the set-up procedure.

Operation (1) Writing a value to the D-A register starts D-A conversion.
(2) Setting the D-Ai output enable bit to “1” outputs an analog signal on the DAi pin.
(3) The D-A converter continues outputting an analog signal until the D-A output enable bit is set to “0”.

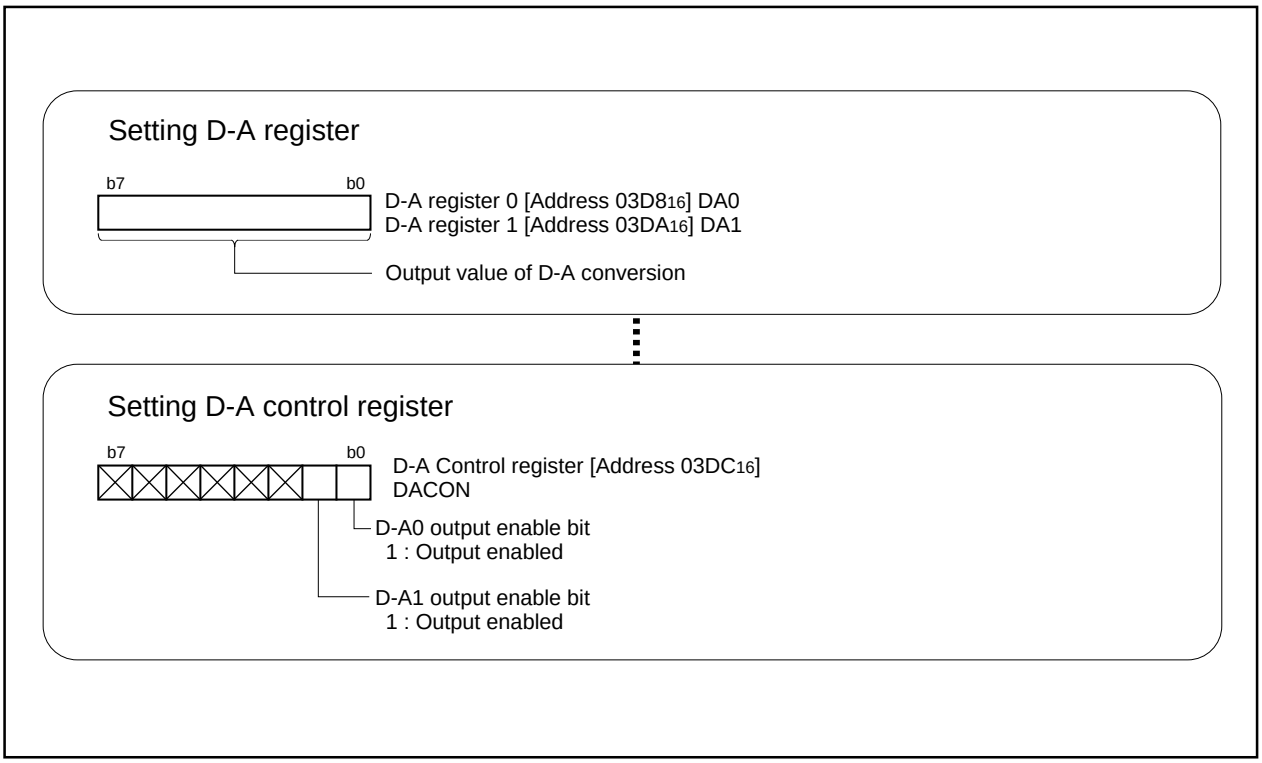


Figure 2.8.3. Set-up procedure of D-A converter

2.9 DMAC Usage

2.9.1 Overview of the DMAC usage

DMAC transfers one data item held in the source address to the destination address every time a transfer request is generated. The following is an overview of the DMAC usage.

(1) Source address and destination address

Both the register which indicates a source and the register which indicates a destination comprise of 24 bits, so that each can cover a 1M bytes space. After transfer of one bit of data is completed, the address in either the source register or the destination register can be incremented. However, both registers cannot be incremented. The links between the source and destination are as follows:

(a) A fixed address from an arbitrary 1M bytes space

(b) An arbitrary 1M bytes space from a fixed address

(c) A fixed address from another fixed address

(2) The number of bits of data transferred

The number of bit of data indicated by the transfer counter is transferred. If a 16-bit transfer is selected, up to 128K bytes can be transferred. If an 8-bit transfer is selected, up to 64K bytes can be transferred. The transfer counter is decremented each time one bit of data is transferred, and a DMA interrupt request occurs when the transfer counter underflows.

(3) DMA transfer factor

The DMA transfer factor can be selected from the following 25 factors: falling edge/two edges of $\overline{\text{INT0}}$/$\overline{\text{INT1}}$ pin, timer A0 interrupt request through timer A4 interrupt request, timer B0 interrupt request through timer B5 interrupt request, UART0 transmission interrupt request, UART0 reception interrupt request, UART1 transmission/UART1 reception interrupt request, UART2 transmission interrupt request, UART2 reception interrupt request, SI/O 3, 4 interrupt request, A-D conversion interrupt request, and software trigger.

When software trigger is selected, DMA transfer is generated by writing “1” to software DMA interrupt request bit. When other factor is selected, DMA transfer is generated by generating corresponding interrupt request.

(4) Channel priority

If DMA0 transfer request and DMA1 transfer request occur simultaneously, priority is given to DMA0.

(5) Writing to a register

When writing to the source register or the destination register with DMA enabled, the content of the register with a fixed address will change at the time of writing. Therefore, the user should not write to a register with a fixed address when the DMA enable bit is set to “1”. The contents of the register with ‘forward direction’ selected, and the transfer counter, are changed when reloaded. A reload occurs either when the transfer counter underflows, or when the DMA enable bit is re-enabled, after having been disabled.

The reload register can be written to, as in normal conditions.

(6) Reading to a register

The reload register can be read to, as in normal conditions.

(7) Switching function

(a) Switching between one-shot transfer and repeated transfer

'One-shot transfer' refers to a mode in which DMA is disabled after the transfer counter underflows. 'Repeated transfer' refers to a mode in which a reload is carried out after the transfer counter underflows. The reload is carried out for the transfer counter and on the address pointer subjected to forward direction.

The following are examples of operation in which the options listed are selected.

• A fixed address from an arbitrary 1M byte space, one-shot transfer P2-148
• An arbitrary 1M byte space from a fixed address, repeated transfer P2-150

(8) Registers related to DMAC

Figure 2.9.1 shows the memory map of DMAC-related registers, and Figures 2.9.2 and 2.9.3 show DMAC-related registers.

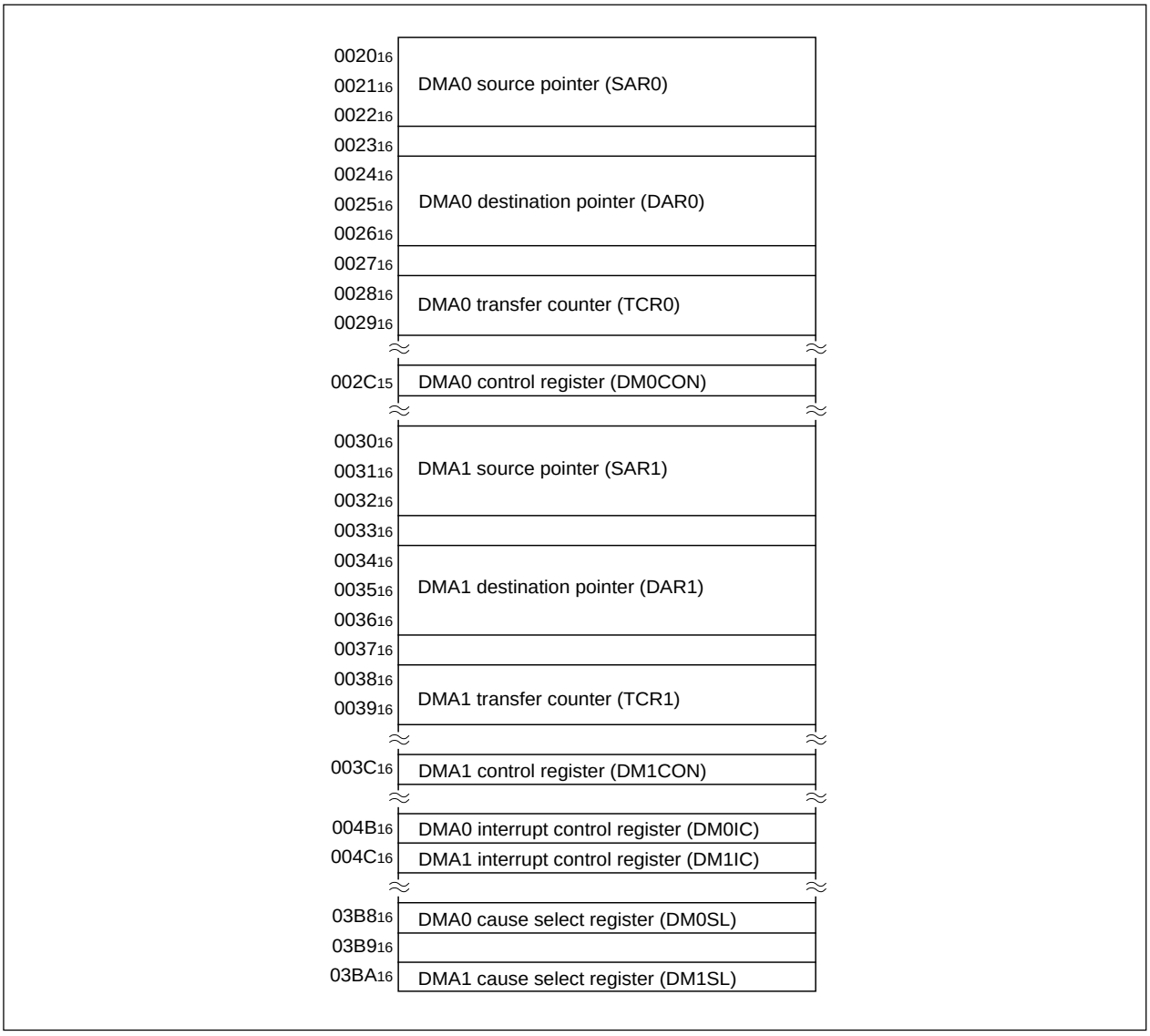
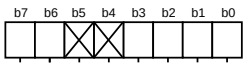


Figure 2.9.1. Memory map of DMAC-related registers

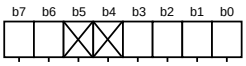
DMA0 request cause select register



Symbol: DM0SL | Address: $03B8_{16}$ | When reset: 00_{16}

Bit symbol	Bit name	Function	R	W
DSEL0	DMA request cause select bit	b3 b2 b1 b0 0 0 0 0 : Falling edge of $\overline{INT0}$ pin 0 0 0 1 : Software trigger 0 0 1 0 : Timer A0 0 0 1 1 : Timer A1 0 1 0 0 : Timer A2 0 1 0 1 : Timer A3 0 1 1 0 : Timer A4 (DMS=0) /two edges of $\overline{INT0}$ pin (DMS=1) 0 1 1 1 : Timer B0 (DMS=0) Timer B3 (DMS=1) 1 0 0 0 : Timer B1 (DMS=0) Timer B4 (DMS=1) 1 0 0 1 : Timer B2 (DMS=0) Timer B5 (DMS=1) 1 0 1 0 : UART0 transmit 1 0 1 1 : UART0 receive 1 1 0 0 : UART2 transmit 1 1 0 1 : UART2 receive 1 1 1 0 : A-D conversion 1 1 1 1 : UART1 transmit	○	○
DSEL1			○	○
DSEL2			○	○
DSEL3			○	○
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be “0”.			–	–
DMS	DMA request cause expansion select bit	0 : Normal 1 : Expanded cause	○	○
DSR	Software DMA request bit	If software trigger is selected, a DMA request is generated by setting this bit to “1” (When read, the value of this bit is always “0”)	○	○

DMA1 request cause select register

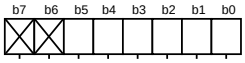


Symbol: DM1SL | Address: $03BA_{16}$ | When reset: 00_{16}

Bit symbol	Bit name	Function	R	W
DSEL0	DMA request cause select bit	b3 b2 b1 b0 0 0 0 0 : Falling edge of $\overline{INT1}$ pin 0 0 0 1 : Software trigger 0 0 1 0 : Timer A0 0 0 1 1 : Timer A1 0 1 0 0 : Timer A2 0 1 0 1 : Timer A3(DMS=0) /serial I/O3 (DMS=1) 0 1 1 0 : Timer A4 (DMS=0) /serial I/O4 (DMS=1) 0 1 1 1 : Timer B0 (DMS=0) /two edges of $\overline{INT1}$ (DMS=1) 1 0 0 0 : Timer B1 1 0 0 1 : Timer B2 1 0 1 0 : UART0 transmit 1 0 1 1 : UART0 receive 1 1 0 0 : UART2 transmit 1 1 0 1 : UART2 receive 1 1 1 0 : A-D conversion 1 1 1 1 : UART1 receive	○	○
DSEL1			○	○
DSEL2			○	○
DSEL3			○	○
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be “0”.			–	–
DMS	DMA request cause expansion select bit	0 : Normal 1 : Expanded cause	○	○
DSR	Software DMA request bit	If software trigger is selected, a DMA request is generated by setting this bit to “1” (When read, the value of this bit is always “0”)	○	○

Figure 2.9.2. DMAC-related registers (1)

DMAi control register

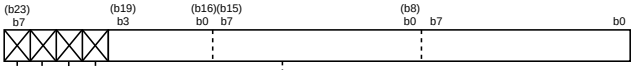


Symbol	Address	When reset
DMiCON(i=0,1)	$002C_{16}$, $003C_{16}$	$00000X00_2$

Bit symbol	Bit name	Function	R	W
DMBIT	Transfer unit bit select bit	0 : 16 bits 1 : 8 bits	○	○
DMASL	Repeat transfer mode select bit	0 : Single transfer 1 : Repeat transfer	○	○
DMAS	DMA request bit (Note 1)	0 : DMA not requested 1 : DMA requested	○	○ (Note 2)
DMAE	DMA enable bit	0 : Disabled 1 : Enabled	○	○
DSD	Source address direction select bit (Note 3)	0 : Fixed 1 : Forward	○	○
DAD	Destination address direction select bit (Note 3)	0 : Fixed 1 : Forward	○	○
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be “0”.			–	–

Note 1: DMA request can be cleared by resetting the bit.
Note 2: This bit can only be set to “0”.
Note 3: Source address direction select bit and destination address direction select bit cannot be set to “1” simultaneously.

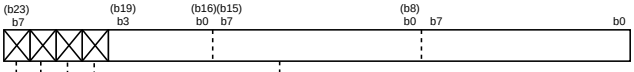
DMAi source pointer (i = 0, 1)



Symbol	Address	When reset
SAR0	0022_{16} to 0020_{16}	Indeterminate
SAR1	0032_{16} to 0030_{16}	Indeterminate

Function	Transfer address specification	R	W
• Source pointer Stores the source address	00000_{16} to $FFFFF_{16}$	○	○
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be “0”.		–	–

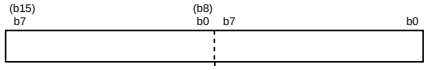
DMAi destination pointer (i = 0, 1)



Symbol	Address	When reset
DAR0	0026_{16} to 0024_{16}	Indeterminate
DAR1	0036_{16} to 0034_{16}	Indeterminate

Function	Transfer address specification	R	W
• Destination pointer Stores the destination address	00000_{16} to $FFFFF_{16}$	○	○
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be “0”.		–	–

DMAi transfer counter (i = 0, 1)



Symbol	Address	When reset
TCR0	0029_{16}, 0028_{16}	Indeterminate
TCR1	0039_{16}, 0038_{16}	Indeterminate

Function	Transfer count specification	R	W
• Transfer counter Set a value one less than the transfer count	0000_{16} to $FFFF_{16}$	○	○

Figure 2.9.3. DMAC-related registers (2)

2.9.2 Operation of DMAC (one-shot transfer mode)

In one-shot transfer mode, choose functions from the items shown in Table 2.9.1. Operations of the circled items are described below. Figure 2.9.4 shows an example of operation and Figure 2.9.5 shows the set-up procedure.

Table 2.9.1. Choosed functions

Item	Set-up	
Transfer space	**O**	Fixed address from an arbitrary 1 M bytes space
		Arbitrary 1 M bytes space from a fixed address
		Fixed address from fixed address
Unit of transfer	**O**	8 bits
		16 bits

Operation (1) When software trigger is selected, setting software DMA request bit to “1” generates a DMA transfer request signal.

(2) If DMAC is active, data transfer starts, and the contents of the address indicated by the DMAi forward-direction address pointer are transferred to the address indicated by the DMAi destination pointer. When data transfer starts directly after DMAC becomes active, the value of the DMAi transfer counter reload register is reloaded to the DMAi transfer counter, and the value of the DMAi source pointer is reloaded by the DMAi forward-direction address pointer. Each time a DMA transfer request signal is generated, 1 byte of data is transferred. The DMAi transfer counter is down counted, and the DMAi forward-direction address pointer is up counted.

(3) If the DMA transfer counter underflows, the DMA enable bit changes to “0” and DMA transfer is completed. The DMA interrupt request bit changes to “1” simultaneously.

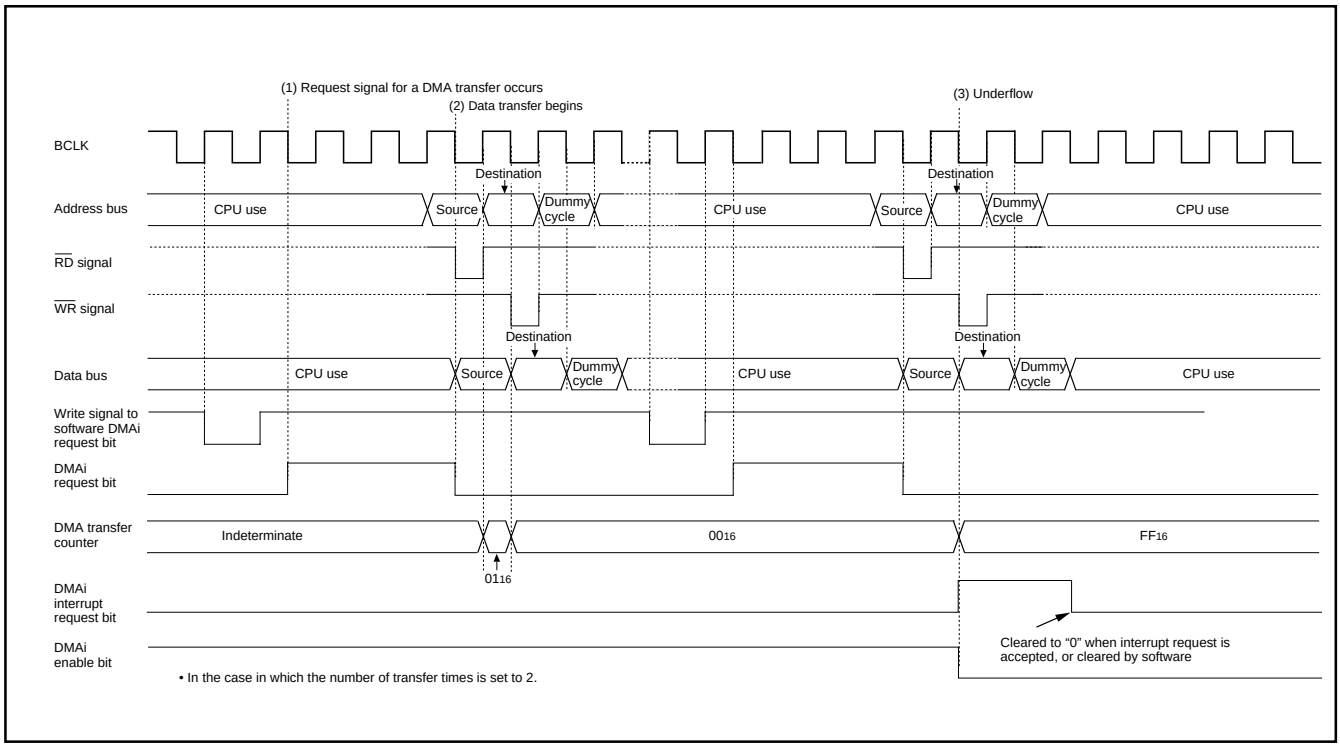


Figure 2.9.4. Example of operation of one-shot transfer mode

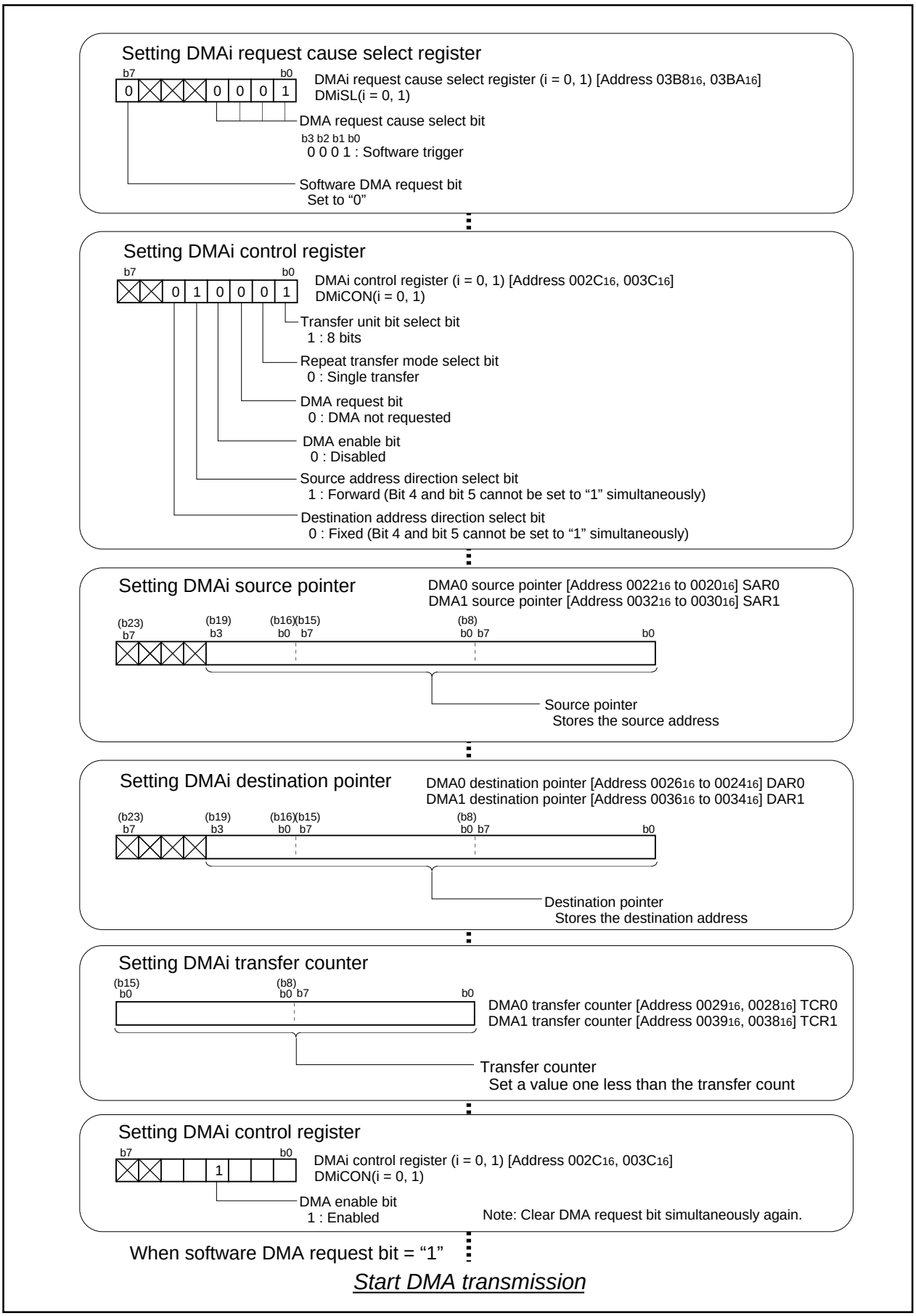


Figure 2.9.5. Set-up procedure of one-shot transfer mode

2.9.3 Operation of DMAC (repeated transfer mode)

In repeat transfer mode, choose functions from the items shown in Table 2.9.2. Operations of the circled items are described below. Figure 2.9.6 shows an example of operation and Figure 2.9.7 shows the set-up procedure.

Table 2.9.2. Choosed functions

Item	Set-up	
Transfer space		Fixed address from an arbitrary 1 M bytes space
	O	Arbitrary 1 M bytes space from a fixed address
		Fixed address from fixed address
Unit of transfer		8 bits
	O	16 bits

Operation (1) When software trigger is selected, setting software DMA request bit to “1” generates a DMA transfer request signal.

(2) If DMAC is active, data transfer starts, and the contents of the address indicated by the DMAi forward-direction address pointer are transferred to the address indicated by the DMAi destination pointer. When data transfer starts directly after DMAC becomes active, the value of the DMAi transfer counter reload register is reloaded to the DMAi transfer counter, and the value of the DMAi source pointer is reloaded by the DMAi forward-direction address pointer. Each time a DMA transfer request signal is generated, 2 byte of data is transferred. The DMAi transfer counter is down counted, and the DMAi forward-direction address pointer is up counted.

(3) Though DMAi transfer counter is underflowed, DMA enable bit is still “1”. The DMA interrupt request bit changes to “1” simultaneously.

(4) After DMAi transfer counter is underflowed, when the next DMA request is generated, DMA transfer is repeated from (1).

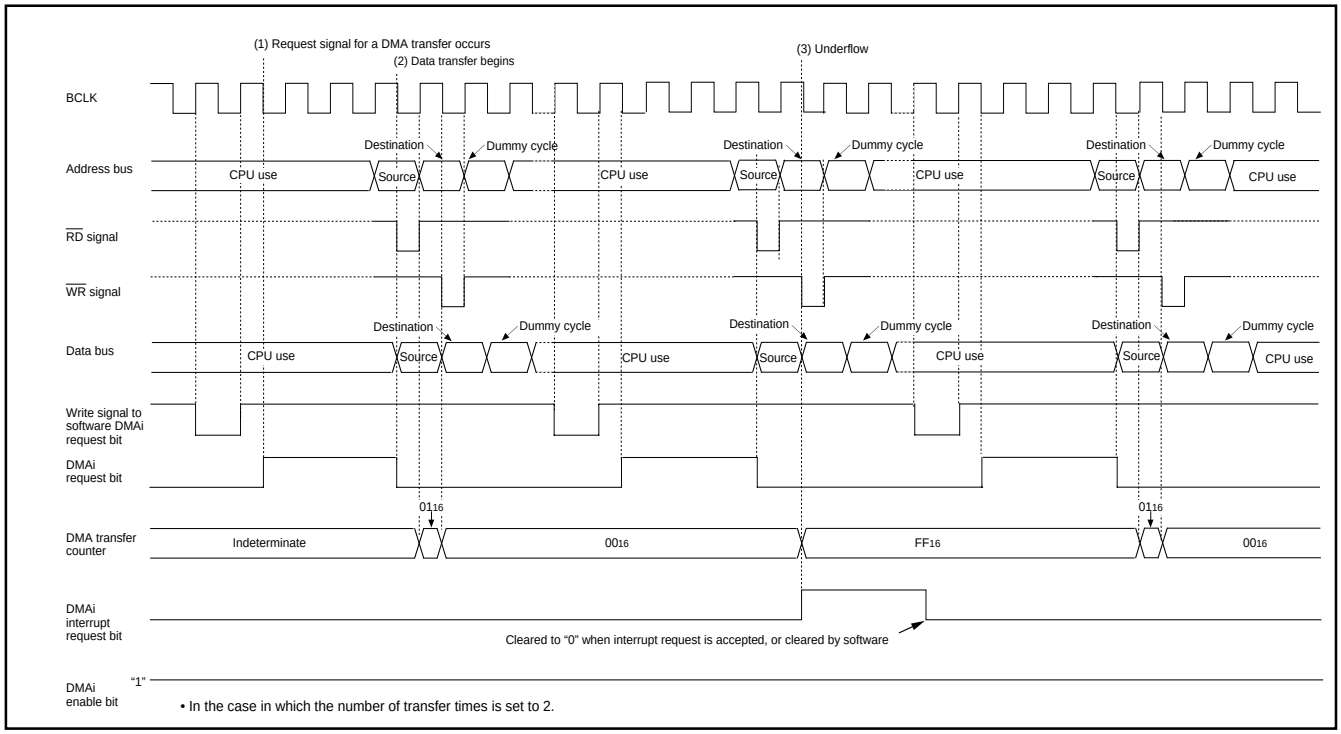


Figure 2.9.6. Example of operation of repeated transfer mode

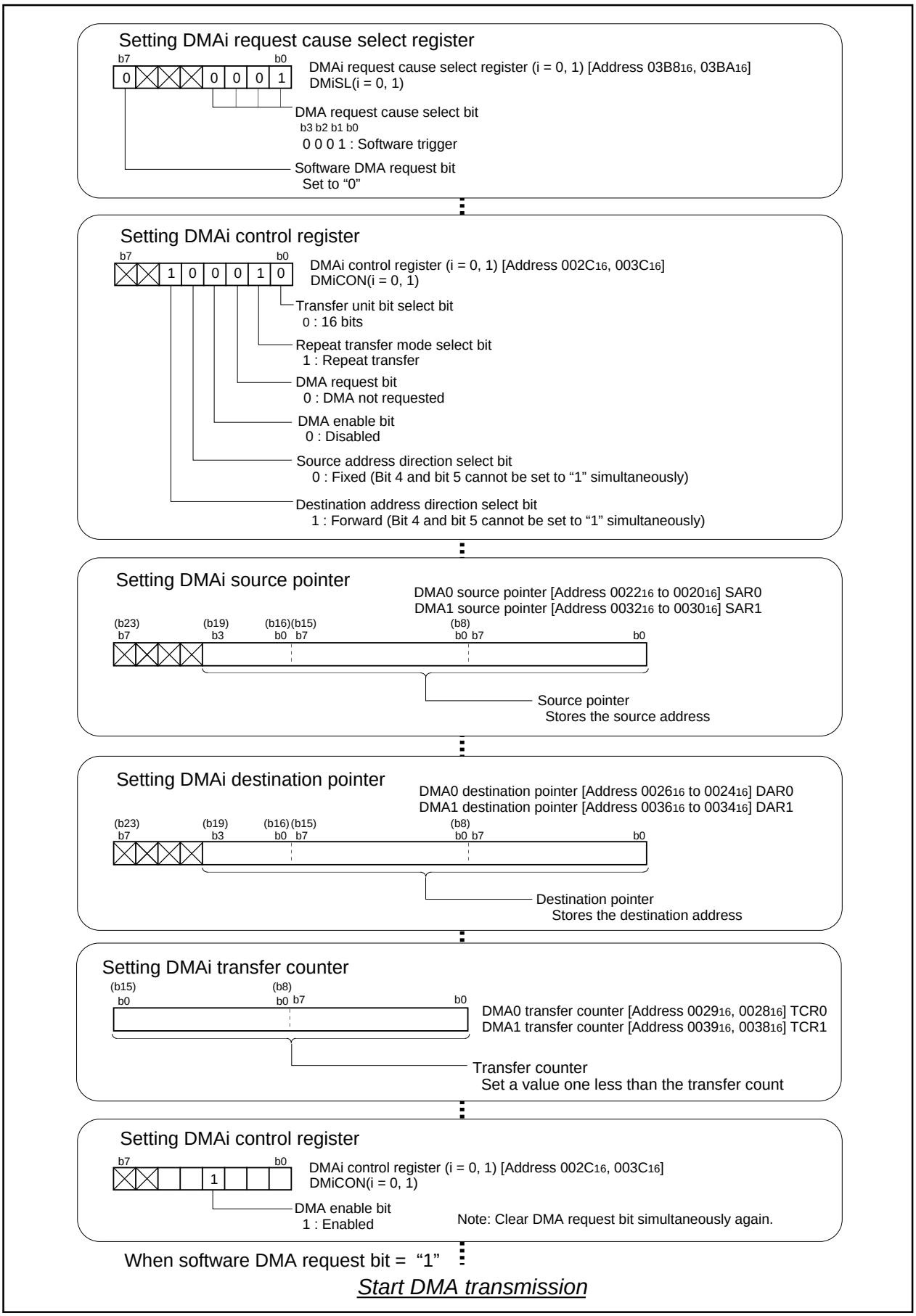


Figure 2.9.7. Set-up procedure of repeated transfer mode

2.10 CRC Calculation Circuit Usage

2.10.1 Overview of the CRC calculation circuit usage

Cyclic Redundancy Check (CRC) is a method that compares CRC code formed from transmission data by use of a polynomial generation with CRC check data so as to detect errors in transmission data. Using the CRC calculation circuit allows generation of CRC code. A polynomial counter is used for the polynomial generation of CRC_CCITT ($X^{16} + X^{12} + X^5 + 1$).

(1) Registers related to CRC calculation circuit

Figure 2.10.1 shows the memory map of CRC-related registers, and Figure 2.10.2 shows CRC- related registers.

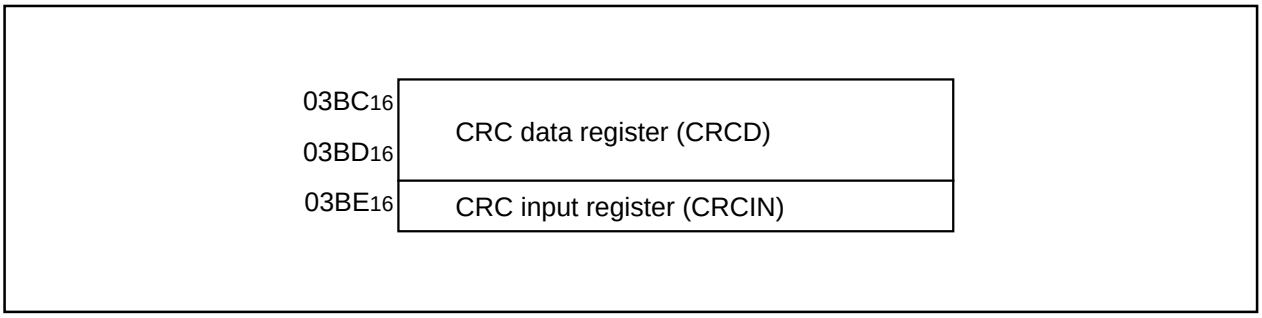


Figure 2.10.1. Memory map of CRC-related registers

CRC data register

(b15) b7	(b8) b0 b7	b0	Symbol	Address	When reset
			CRCD	03BD16, 03BC16	Indeterminate

Function	Values that can be set	R	W
CRC calculation result output register	0000₁₆ to FFFF₁₆	○	○

CRC input register

b7 b0	Symbo	Address	When reset
	CRCIN	03BE16	Indeterminate

Function	Values that can be set	R	W
Data input register	00₁₆ to FF₁₆	○	○

Figure 2.10.2. CRC-related registers

2.10.2 Operation of CRC Calculation Circuit

The following describes the operation of the CRC calculation. Figure 2.10.3 shows an example of calculation data 0123_{16} using the CRC calculation circuit.

Operation (1) The CRC calculation circuit sets an initial value in the CRC data register.
(2) Writing 1 byte data to the CRC input register generates CRC code based on the data register. CRC code generation for 1 byte data finishes in two machine cycles.
(3) The CRC calculation circuit detects an error by means of comparing the CRC-checking data with the content of the CRC data register, after the next data is written to the CRC input register.
(4) The content of CRC data register after all data is written becomes CRC code.

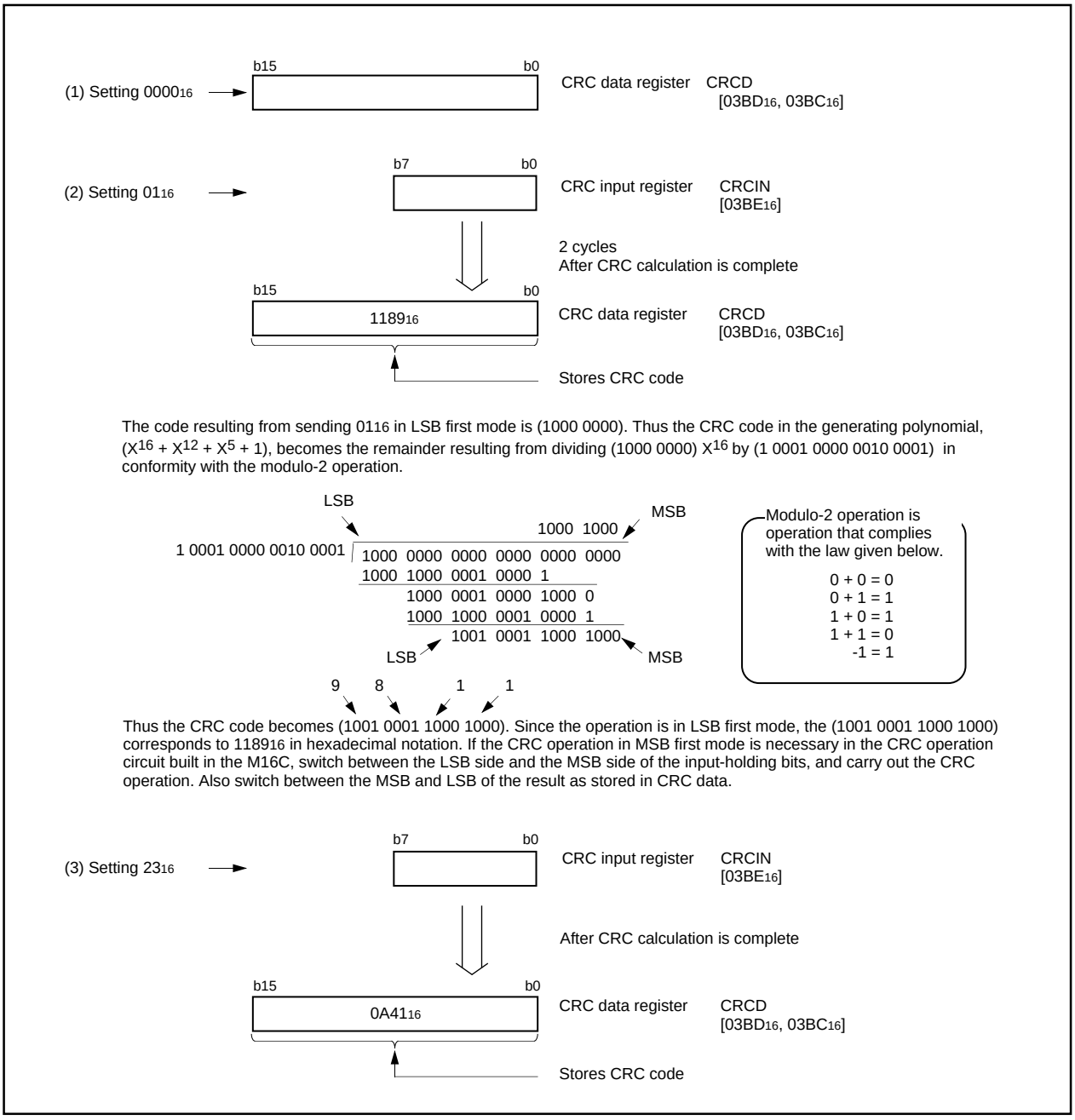


Figure 2.10.3. Calculation example using the CRC calculation circuit

2.11 Watchdog Timer Usage

2.11.1 Overview of the watchdog timer usage

The watchdog timer can detect a runaway program using its 15-bit timer prescaler. The following is an overview of the watchdog timer usage.

(1) Watchdog timer start procedure

When reset, the watchdog timer is in stopped state. Writing to the watchdog timer start register initializes the watchdog timer to 7FFF16 and causes it to start performing a down count. The watchdog timer, once started operating, cannot be stopped by any means other than stopping conditions.

(2) Watchdog timer stop conditions

The watchdog timer stops in any one of the following states:

(a) Period in which the CPU is in stopped state

(b) Period in which the CPU is in waiting state

(c) Period in which the microcomputer is in hold state

(3) Watchdog timer initialization

The watchdog timer is initialized to 7FFF16 in the cases given below, and begins a down count.

(a) When the watchdog timer writes to the watchdog timer start register while a count is in progress

(b) When the watchdog timer underflows

(4) Runaway detection

When the watchdog timer underflows, a watchdog timer interrupt occurs. In writing a program, write to the watchdog timer start register before the watchdog timer underflows. The watchdog timer interrupt occurs regardless of the status of the interrupt enable flag (I flag). In processing a watchdog timer interrupt, set the software reset bit to “1” to reset software.

(5) Watchdog timer cycle

The watchdog timer cycle varies depending on the BCLK and the frequency division ratio of the prescaler selected.

Table 2.11.1. The watchdog timer cycle

CM07	CM06	CM17	CM16	BCLK	WDC7	Period
0	0	0	0	16MHz	0	Approx. 32.8ms (Note)
					1	Approx. 262.1ms (Note)
0	0	0	1	8MHz	0	Approx. 65.5ms (Note)
					1	Approx. 524.3ms (Note)
0	0	1	0	4MHz	0	Approx. 131.1ms (Note)
					1	Approx. 1.049s (Note)
0	0	1	1	1MHz	0	Approx. 524.3ms (Note)
					1	Approx. 4.194s (Note)
0	1	Invalid	Invalid	2MHz	0	Approx. 262.1ms (Note)
					1	Approx. 2.097s (Note)
1	Invalid	Invalid	Invalid	32kHz	Invalid	Approx. 2s (Note)

Note: An error due to the prescaler occurs.

(6) Registers related to the watchdog timer

Figure 2.11.1 shows the memory map of watchdog timer-related registers, and Figure 2.11.2 shows watchdog timer-related registers.

000E16	Watchdog timer start register (WDTS)
000F16	Watchdog timer control register (WDC)

Figure 2.11.1. Memory map of watchdog timer-related registers

Watchdog timer control register

b7	b6	b5	b4	b3	b2	b1	b0
	0	0					

Symbol: WDC | Address: 000F16 | When reset: 000XXXXX2

Bit symbol	Bit name	Function	R	W
	High-order bit of watchdog timer		O	X
	Reserved bit	Must always be set to “0”	O	O
	Reserved bit	Must always be set to “0”	O	O
WDC7	Prescaler select bit	0 : Divided by 16 1 : Divided by 128	O	O

Watchdog timer start register

b7	b0

Symbol: WDTS | Address: 000E16 | When reset: Indeterminate

Function	R	W
The watchdog timer is initialized and starts counting after a write instruction to this register. The watchdog timer value is always initialized to “7FFF16” regardless of whatever value is written.	X	O

Figure 2.11.2. Watchdog timer-related registers

2.11.2 Operation of Watchdog Timer

The following is an operation of the watchdog timer. Figure 2.11.3 shows the operation timing, and Figure 2.11.4 shows the set-up procedure.

Operation (1) Writing to the watchdog timer start register initializes the watchdog timer to 7FFF16 and causes it to start a down count.

(2) With a count in progress, writing to the watchdog timer start register again initializes the watchdog timer to 7FFF16 and causes it to resume counting.

(3) Either executing the WAIT instruction or going to the stopped state causes the watchdog timer to hold the count in progress and to stop counting. The watchdog timer resumes counting after returning from the execution of the WAIT instruction or from the stopped state.

(4) If the watchdog timer underflows, it is initialized to 7FFF16 and continues counting. At this time, a watchdog timer interrupt occurs.

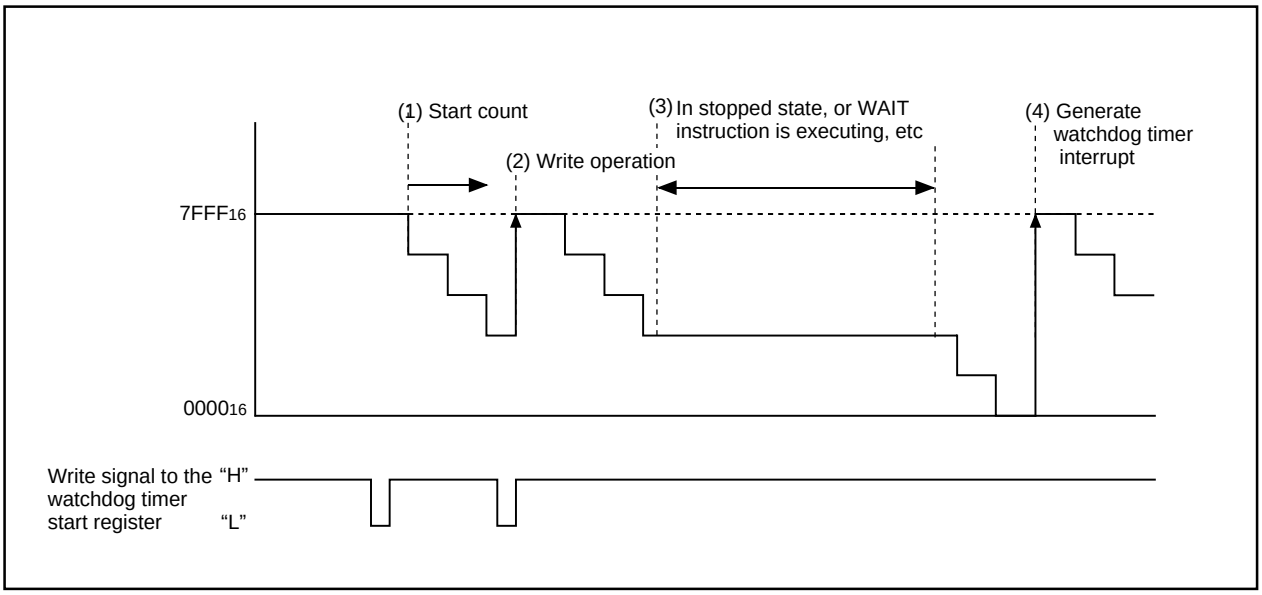


Figure 2.11.3. Operation timing of watchdog timer

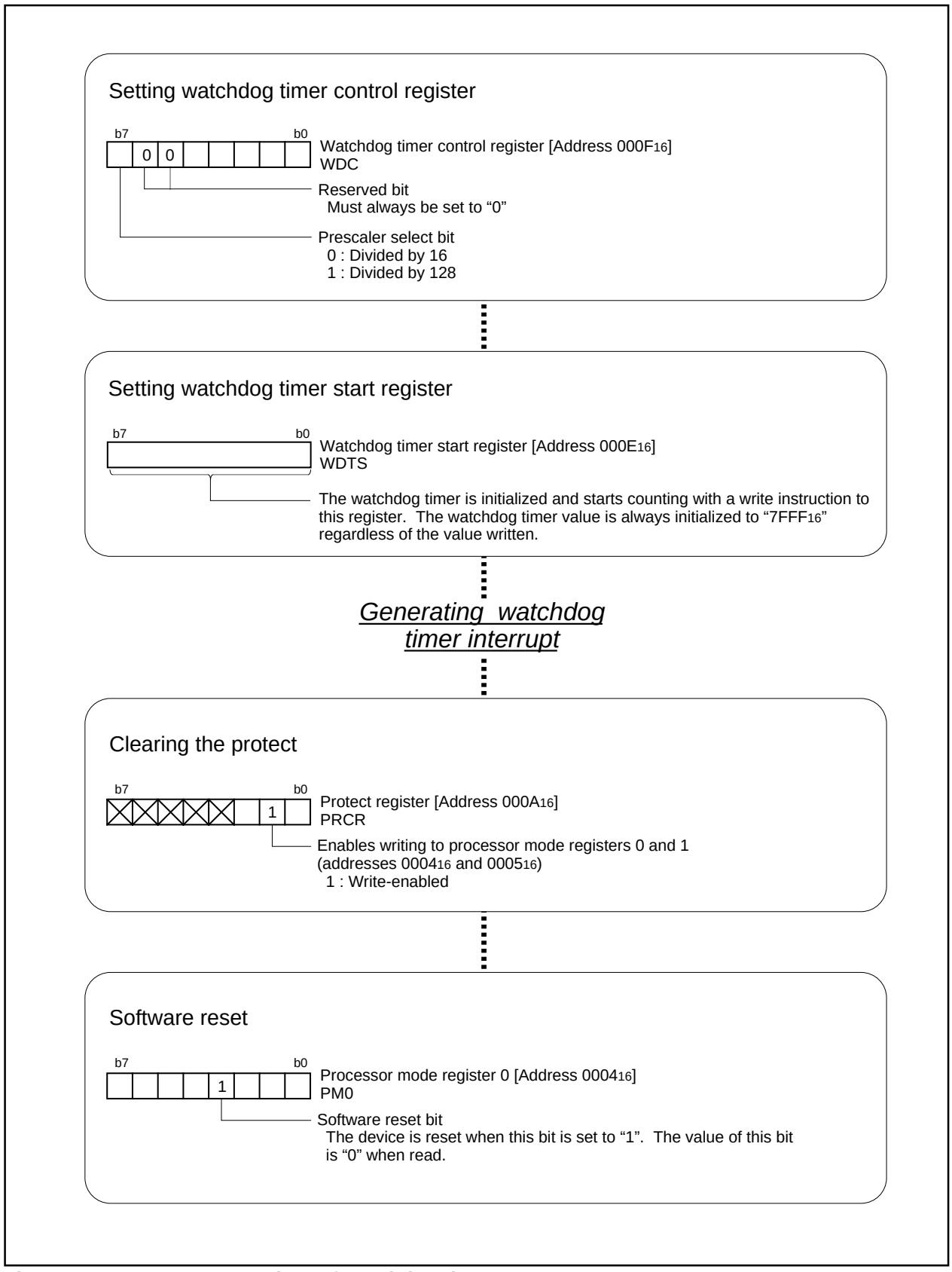


Figure 2.11.4. Set-up procedure of watchdog timer

2.12 Address Match Interrupt Usage

2.12.1 Overview of the address match interrupt usage

The address match interrupt is used for correcting a ROM or for a simplified debugging-purpose monitor. The following is an overview of the address match interrupt usage.

(1) Enabling/disabling the address match interrupt

The address match interrupt enable bit can be used to enable and disable an address match interrupt. It is affected neither by the processor interrupt priority level (IPL) nor the interrupt enable flag (I flag).

(2) Timing of the address match interrupt

An interrupt occurs immediately before executing the instruction in the address indicated by the address match interrupt register. Set the first address of the instruction in the address match interrupt register. Setting a half address of an instruction or an address of tabulated data does not generate an address match interrupt.

The first instruction of an interrupt routine does not generate an address match interrupt either.

(3) Returning from an address match interrupt

The address put in the stack when an address match interrupt occurs depends on the instruction not yet executed (the instruction the address match interrupt register indicates). The return address is not put in the stack. For this reason, to return from an address match interrupt, either rewrite the content of the stack and use the REIT instruction or use the POP instruction to restore the stack to the state as it was before the interrupt occurred and return by use of a jump instruction.

Figure 2.12.1 shows unexecuted instructions and corresponding the stacked addresses.

<Instructions whose address is added to by 2 when an address match interrupt occurs>

- 16-bit operation code instructions
- 8-bit operation code instructions given below

ADD.B:S	#IMM8,dest	SUB.B:S	#IMM8,dest	AND.B:S	#IMM8,dest
OR.B:S	#IMM8,dest	MOV.B:S	#IMM8,dest	STZ.B:S	#IMM8,dest
STNZ.B:S	#IMM8,dest	STZX.B:S	#IMM81,#IMM82,dest		
CMP.B:S	#IMM8,dest	PUSHM	src	POPM	dest
JMPS	#IMM8	JSRS	#IMM8		
MOV.B:S	#IMM,dest (However, dest = A0/A1)				

<Instructions whose address is added to by 1 when an address match interrupt occurs>

- Instructions other than those listed above

Figure 2.12.1. Unexecuted instructions and corresponding stacked addresses

(4) How to determine an address match interrupt

Address match interrupts can be set at two different locations. However, both location will have the same vector address. Therefore, it is necessary to determine which interrupt has occurred; address match interrupt 0 or address match interrupt 1. Using the content of the stack, etc., determine which interrupt has occurred according to the first part of the address match interrupt routine.

(5) Registers related to the address match interrupt

Figure 2.12.2 shows the memory map of address match interrupt-related registers, and Figure 2.12.3 shows address match interrupt-related registers.

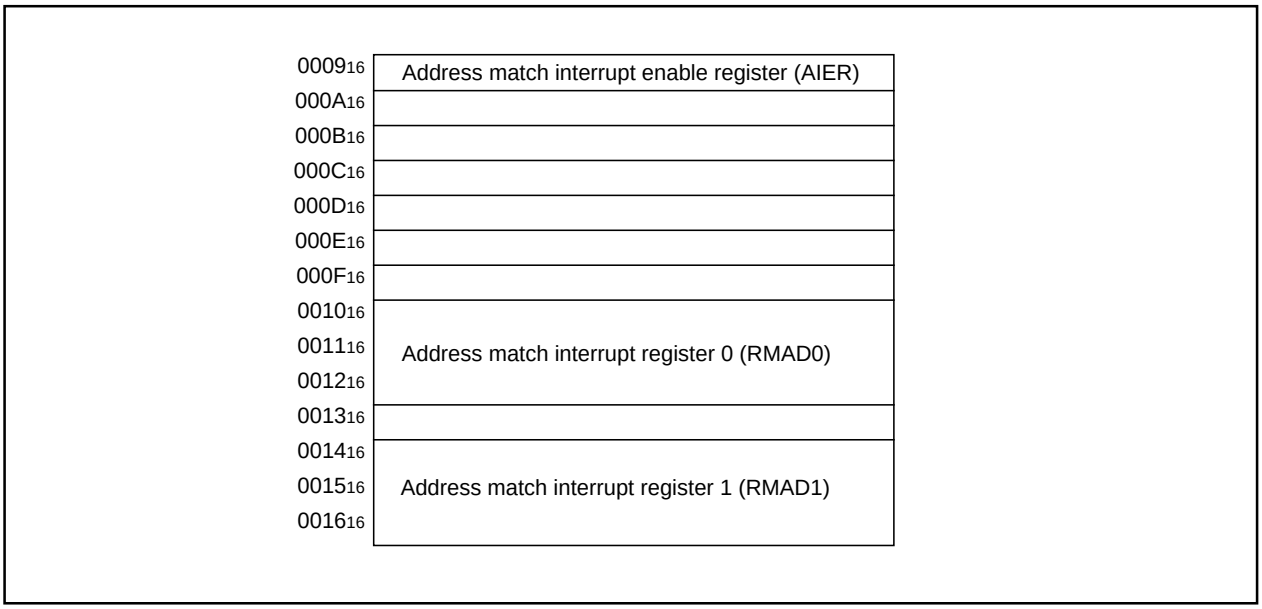


Figure 2.12.2. Memory map of address match interrupt-related registers

Address match interrupt enable register

b7 b6 b5 b4 b3 b2 b1 b0

Symbol	Address	When reset
AIER	0009₁₆	XXXXXX00₂

Bit symbol	Bit name	Function	R	W
AIER0	Address match interrupt 0 enable bit	0 : Interrupt disabled 1 : Interrupt enabled	○	○
AIER1	Address match interrupt 1 enable bit	0 : Interrupt disabled 1 : Interrupt enabled	○	○
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be indeterminated.			—	—

Address match interrupt register i (i = 0, 1)

(b23) b7 (b19) b3 (b16)(b15) b0 b7 (b8) b0 b7 b0

Symbol	Address	When reset
RMAD0	0012₁₆ to 0010₁₆	X0000016
RMAD1	0016₁₆ to 0014₁₆	X0000016

Function	Values that can be set	R	W
Address setting register for address match interrupt	00000₁₆ to FFFFF₁₆	○	○
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be indeterminated.		—	—

Figure 2.12.3. Address match interrupt-related registers

2.12.2 Operation of Address Match Interrupt

The following is an operation of address match interrupt. Figure 2.12.4 shows the set-up procedure of address match interrupt, and Figure 2.12.5 shows the overview of the address match interrupt handling routine.

Operation (1) The address match interrupt handling routine sets an address to be used to cause the address match interrupt register to generate an interrupt.
(2) Setting the address match enable flag to “1” enables an interrupt to occur.
(3) An address match interrupt occurs immediately before the instruction in the address indicated by the address match interrupt register as a program is executed.

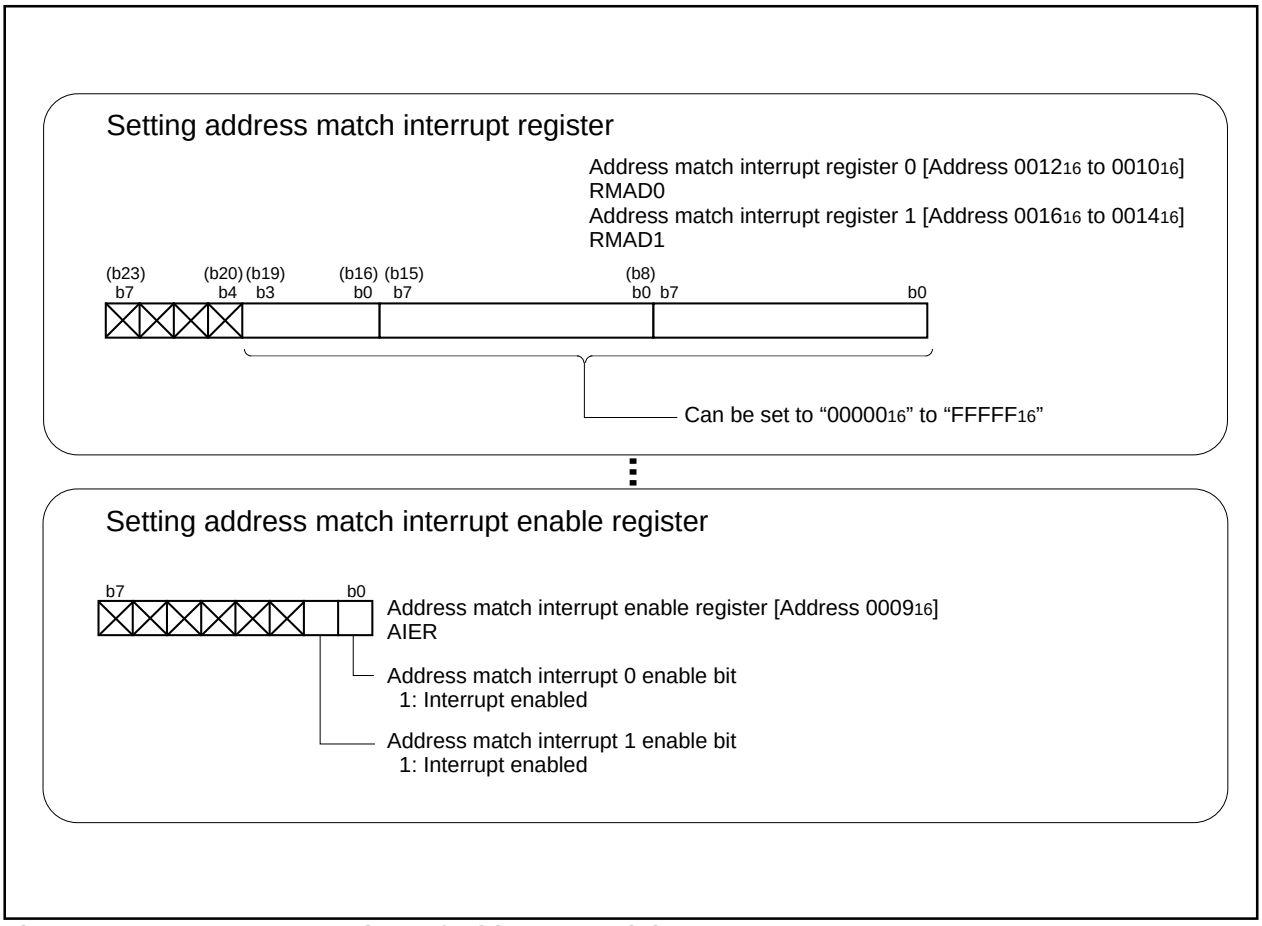
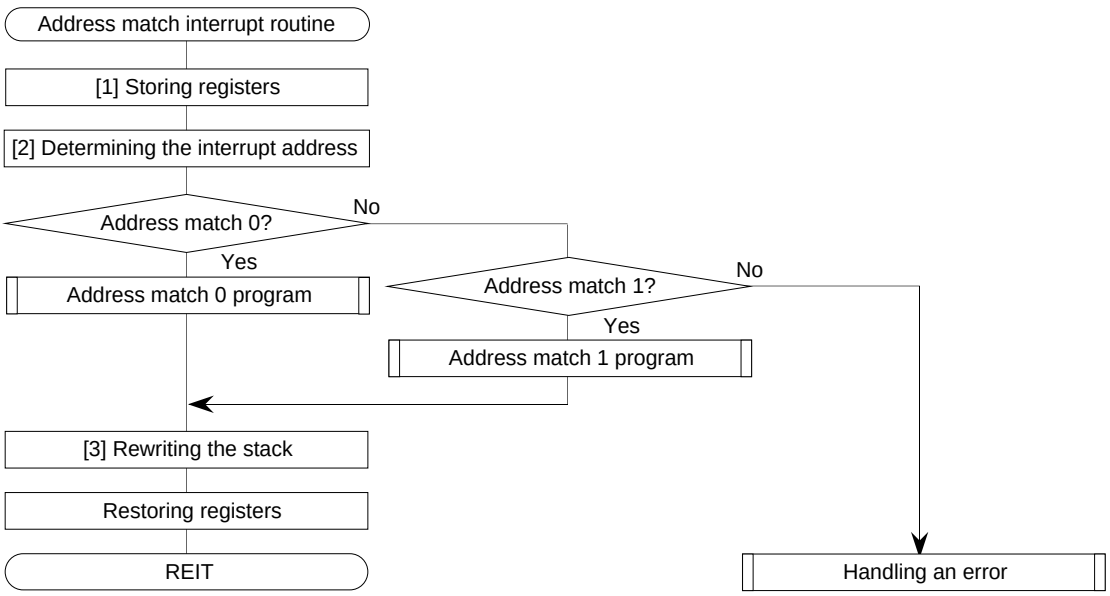


Figure 2.12.4. Set-up procedure of address match interrupt



Explanation:

[1] Storing the contents of the registers holding the main program status to be kept.

[2] Determining the interrupt address
Determining which factor generated the interrupt.

[3] Rewriting the stack
Rewriting the return address.

Figure 2.12.5. Overview of the address match interrupt handling routine

2.13 Key-Input Interrupt Usage

2.13.1 Overview of the key-input interrupt usage

Key-input interrupt occurs when a falling edge is input to P10$_4$ through P10$_7$. The following is an overview of the key-input interrupt usage:

(1) Enabling/disabling the key-input interrupt

The key-input interrupt can be enabled and disabled using the key-input interrupt register. The key-input interrupt is affected by the interrupt priority level (IPL) and the interrupt enable flag (I flag).

(2) Occurrence timing of the key-input interrupt

With key-input interrupt acceptance enabled, pins P10$_4$ through P10$_7$, which are set to input, become key-input interrupt pins ($\overline{KI_0}$ through $\overline{KI_3}$). A key-input interrupt occurs when a falling edge is input to a key-input interrupt pin. At this moment, the level of other key-input interrupt pins must be “H”. No interrupt occurs when the level of other key-input interrupt pins is “L”.

(3) How to determine a key-input interrupt

A key-input interrupt occurs when a falling edge is input to one of four pins, but each pin has the same vector address.

Therefore, read the input level of pins P10$_4$ through P10$_7$ in the key-input interrupt routine to determine the interrupted pin.

(4) Registers related to the key-input interrupt

Figure 2.13.1 shows the memory map of key-input interrupt-related registers, and Figure 2.13.2 shows key-input interrupt-related registers.

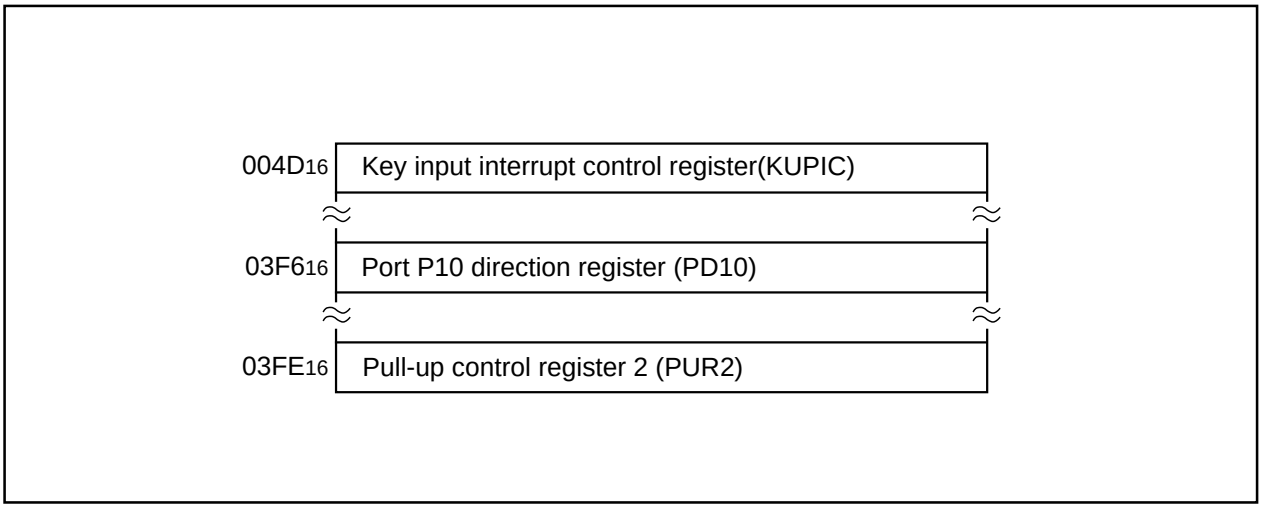
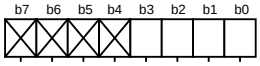


Figure 2.13.1. Memory map of key-input interrupt-related registers

Interrupt control register (Note 2)

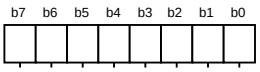


Symbol	Address	When reset
KUPIC	$004D_{16}$	$XXXX0000_2$

Bit symbol	Bit name	Function	R	W
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	○	○
ILVL1			○	○
ILVL2			○	○
IR	Interrupt request bit	0 : Interrupt not requested 1 : Interrupt requested	○	○ (Note1)
Nothing is assigned. These bits can neither be set nor reset. When read, their contents are indeterminate.			–	–

Note 1: This bit can only be accessed for reset (= 0), but cannot be accessed for set (= 1).
Note 2: To rewrite the interrupt control register, do so at a point that dose not generate the interrupt request for that register. For details, see the precautions for interrupts.

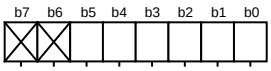
Port P10 direction register



Symbol	Address	When reset
PD10	$03F6_{16}$	00_{16}

Bit symbol	Bit symbol	Bit symbol	R	W
PD10_0	Port $P10_0$ direction register	0 : Imput mode (Functions as an input port) 1 : Output mode (Functions as an output port)	○	○
PD10_1	Port $P10_1$ direction register		○	○
PD10_2	Port $P10_2$ direction register		○	○
PD10_3	Port $P10_3$ direction register		○	○
PD10_4	Port $P10_4$ direction register		○	○
PD10_5	Port $P10_5$ direction register		○	○
PD10_6	Port $P10_6$ direction register		○	○
PD10_7	Port $P10_7$ direction register		○	○

Pull-up control register 2



Symbol	Address	When reset
PUR2	$03FE_{16}$	00_{16}

Bit symbol	Bit name	Function	R	W
PU20	$P8_0$ to $P8_3$ pull-up	The corresponding port is pulled high with a pull-up resistor 0 : Not pulled high 1 : Pulled high	○	○
PU21	$P8_4$ to $P8_7$ pull-up (Except $P8_5$)		○	○
PU22	$P9_0$ to $P9_3$ pull-up		○	○
PU23	$P9_4$ to $P9_7$ pull-up		○	○
PU24	$P10_0$ to $P10_3$ pull-up		○	○
PU25	$P10_4$ to $P10_7$ pull-up		○	○
Nothing is assigned.			–	–
In an attempt to write to these bits, write “0”. The value, if read, turns out to be “0”.			–	–

Figure 2.13.2. key-input interrupt-related registers

2.13.2 Operation of Key-Input Interrupt

The following is an operation of key-input interrupt. Figure 2.13.3 shows an example of a circuit that uses the key-input interrupt, Figure 2.13.4 shows an example of operation of key-input interrupt, and Figure 2.13.5 shows the setting procedure of key-input interrupt.

Operation (1) Set the direction register of the ports to be changed to key-input interrupt pins to input, and set the pull-up function.

(2) Setting the key-input interrupt control register and setting the interrupt enable flag makes the interrupt-enabled state ready.

(3) If a falling edge is input to either $\overline{KI_0}$ through $\overline{KI_3}$, the key-input interrupt request bit goes to “1”.

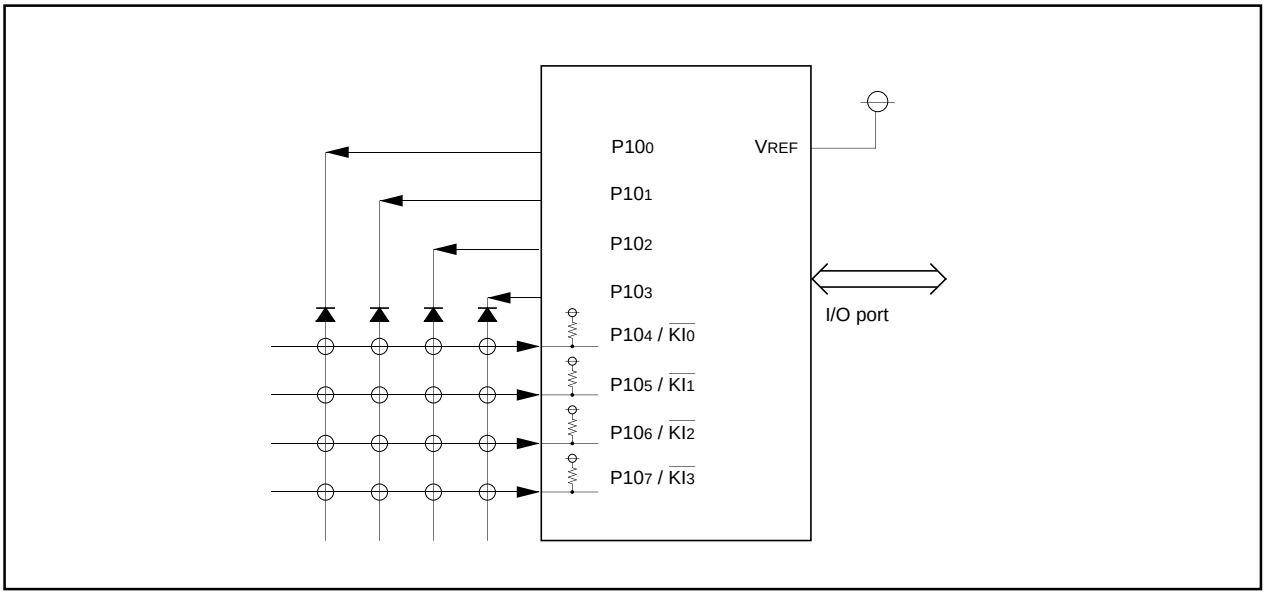


Figure 2.13.3. Example of circuit using the key-input interrupt

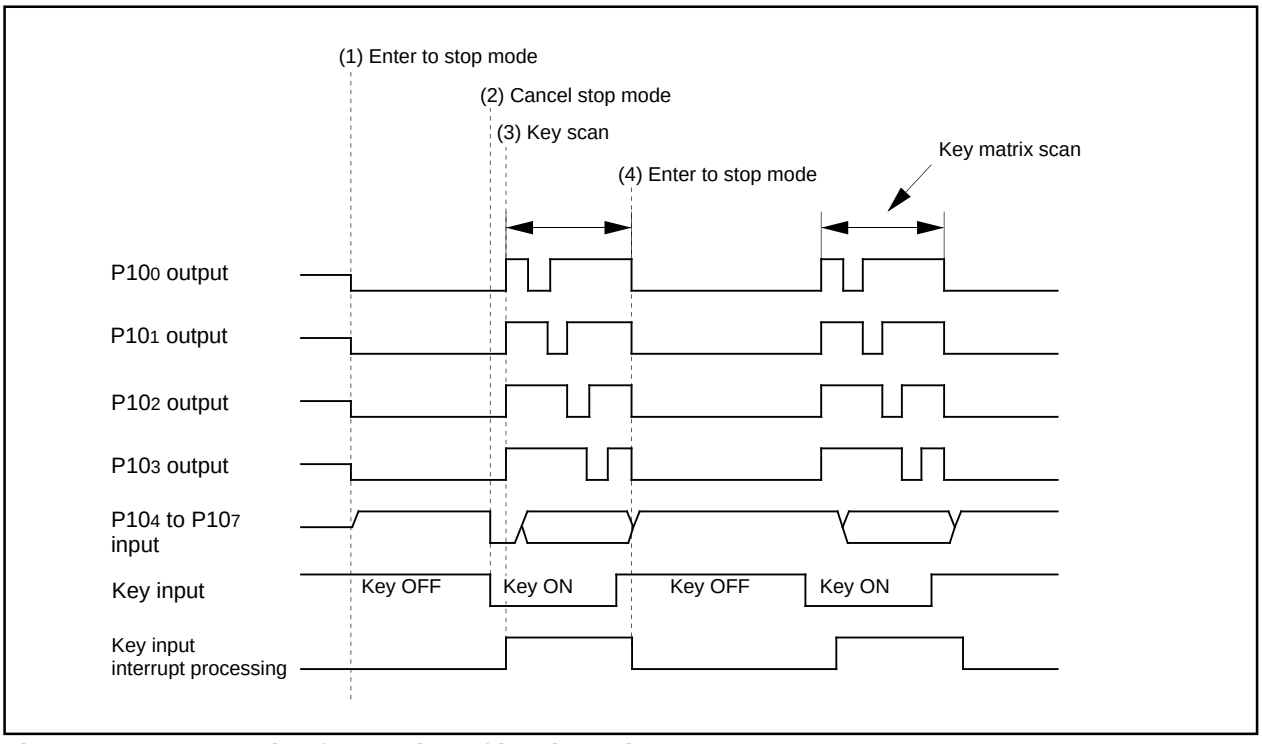


Figure 2.13.4. Example of operation of key-input interrupt

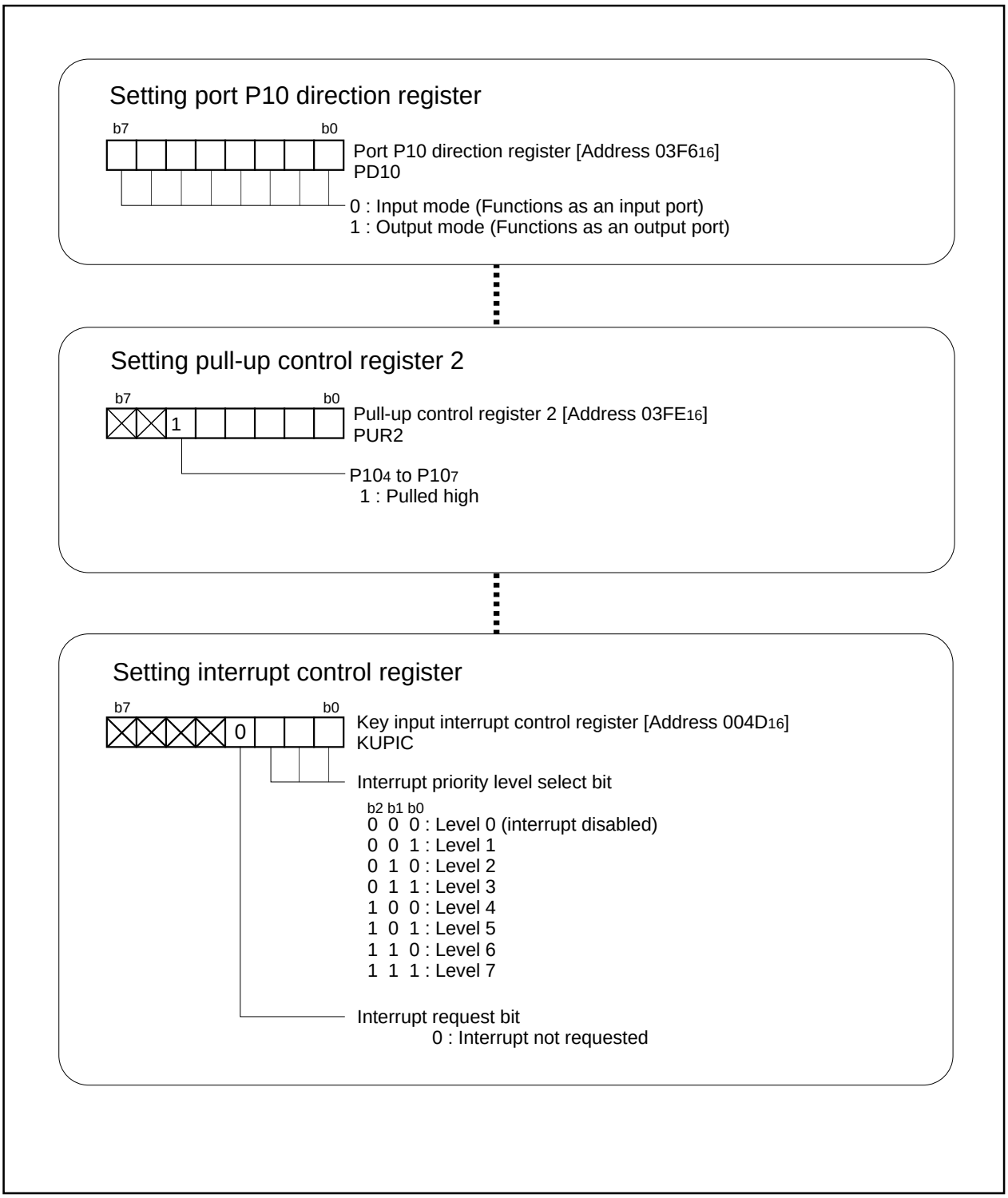


Figure 2.13.5. Set-up procedure of key-input interrupt

2.14 Multiple interrupts Usage

2.14.1 Overview of the Multiple interrupts usage

The following is an overview of the multiple interrupts usage.

(1) Interrupt control

Descriptions are given here regarding how to enable or disable maskable interrupts and how to set the priority to be accepted. What is described here does not apply to non-maskable interrupts.

Enable or disable a non-maskable interrupt using the interrupt enable flag (I flag), interrupt priority level selection bit, or processor interrupt priority level (IPL). Whether an interrupt request is present or absent is indicated by the interrupt request bit. The interrupt request bit and the interrupt priority level select bit are located in the interrupt control register of each interrupt. Also, the interrupt enable flag (I flag) and the IPL are located in the flag register (FLG).

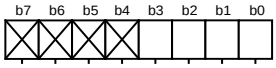
Figure 2.14.1 shows the memory map of the interrupt control registers, and Figure 2.14.2 shows the interrupt control registers.

0044$_{16}$	INT3 interrupt control register (INT3IC)
0045$_{16}$	Timer B5 interrupt control register (TB5IC)
0046$_{16}$	Timer B4 interrupt control register (TB4IC)
0047$_{16}$	Timer B3 interrupt control register (TB3IC)
0048$_{16}$	SI/O4 interrupt control register (S4IC) INT5 interrupt control register (INT5IC)
0049$_{16}$	SI/O3 interrupt control register (S4IC) INT4 interrupt control register (INT4IC)
004A$_{16}$	Bus collision detection interrupt control register (BCNIC)
004B$_{16}$	DMA0 interrupt control register (DM0IC)
004C$_{16}$	DMA1 interrupt control register (DM1IC)
004D$_{16}$	Key input interrupt control register(KUPIC)
004E$_{16}$	A-D conversion interrupt control register (ADIC)
004F$_{16}$	UART2 transmit interrupt control register (S2TIC)
0050$_{16}$	UART2 receive interrupt control register (S2RIC)
0051$_{16}$	UART0 transmit interrupt control register (S0TIC)
0052$_{16}$	UART0 receive interrupt control register (S0RIC)
0053$_{16}$	UART1 transmit interrupt control regster(S1TIC)
0054$_{16}$	UART1 receive interrupt control register(S1RIC)
0055$_{16}$	Timer A0 interrupt control register (TA0IC)
0056$_{16}$	Timer A1 interrupt control register (TA1IC)
0057$_{16}$	Timer A2 interrupt control register (TA2IC)
0058$_{16}$	Timer A3 interrupt control register (TA3IC)
0059$_{16}$	Timer A4 interrupt control register (TA4IC)
005A$_{16}$	Timer B0 interrupt control register (TB0IC)
005B$_{16}$	Timer B1 interrupt control register (TB1IC)
005C$_{16}$	Timer B2 interrupt control register (TB2IC)
005D$_{16}$	INT0 interrupt control register (INT0IC)
005E$_{16}$	INT1 interrupt control register (INT1IC)
005F$_{16}$	INT2 interrupt control register (INT2IC)

Figure 2.14.1. Memory map of the interrupt control registers

Interrupt control register (Note 2)

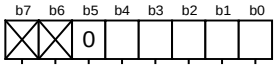
Symbol	Address	When reset
TBiIC(i=3 to 5)	0045_{16} to 0047_{16}	$XXXXX000_2$
BCNIC	$004A_{16}$	$XXXXX000_2$
DMiIC(i=0, 1)	$004B_{16}$, $004C_{16}$	$XXXXX000_2$
KUPIC	$004D_{16}$	$XXXXX000_2$
ADIC	$004E_{16}$	$XXXXX000_2$
SiTIC(i=0 to 2)	0051_{16}, 0053_{16}, $004F_{16}$	$XXXXX000_2$
SiRIC(i=0 to 2)	0052_{16}, 0054_{16}, 0050_{16}	$XXXXX000_2$
TAiIC(i=0 to 4)	0055_{16} to 0059_{16}	$XXXXX000_2$
TBiIC(i=0 to 2)	$005A_{16}$ to $005C_{16}$	$XXXXX000_2$



Bit symbol	Bit name	Function	R	W
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	○	○
ILVL1			○	○
ILVL2			○	○
IR	Interrupt request bit	0 : Interrupt not requested 1 : Interrupt requested	○	○ (Note 1)
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be indeterminate.			–	–

Note 1: This bit can only be accessed for reset (= 0), but cannot be accessed for set (= 1).
Note 2: To rewrite the interrupt control register, do so at a point that does not generate the interrupt request for that register. For details, see the precautions for interrupts.

Symbol	Address	When reset
INTiIC(i=3)	0044_{16}	$XX00X000_2$
SiIC/INTjIC (i=4, 3)	0048_{16}, 0049_{16}	$XX00X000_2$
(j=5, 4)	0048_{16}, 0049_{16}	$XX00X000_2$
INTiIC(i=0 to 2)	$005D_{16}$ to $005F_{16}$	$XX00X000_2$



Bit symbol	Bit name	Function	R	W
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	○	○
ILVL1			○	○
ILVL2			○	○
IR	Interrupt request bit	0: Interrupt not requested 1: Interrupt requested	○	○ (Note 1)
POL	Polarity select bit	0 : Selects falling edge 1 : Selects rising edge	○	○
Reserved bit		Must always be set to “0”	○	○
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be indeterminate.			–	–

Note 1: This bit can only be accessed for reset (= 0), but cannot be accessed for set (= 1).
Note 2: To rewrite the interrupt control register, do so at a point that does not generate the interrupt request for that register. For details, see the precautions for interrupts.

Figure 2.14.2. Interrupt control registers

(2) Interrupt Enable Flag (I flag)

The interrupt enable flag (I flag) controls the enabling and disabling of maskable interrupts. Setting this flag to “1” enables all maskable interrupts; setting it to “0” disables all maskable interrupts. This flag is set to “0” after reset.

The content is changed when the I flag is changed causes the acceptance of the interrupt request in the following timing:

- When changing the I flag using the REIT instruction, the acceptance of the interrupt takes effect as the REIT instruction is executed.
- When changing the I flag using one of the FCLR, FSET, POPC, and LDC instructions, the acceptance of the interrupt is effective as the next instruction is executed.

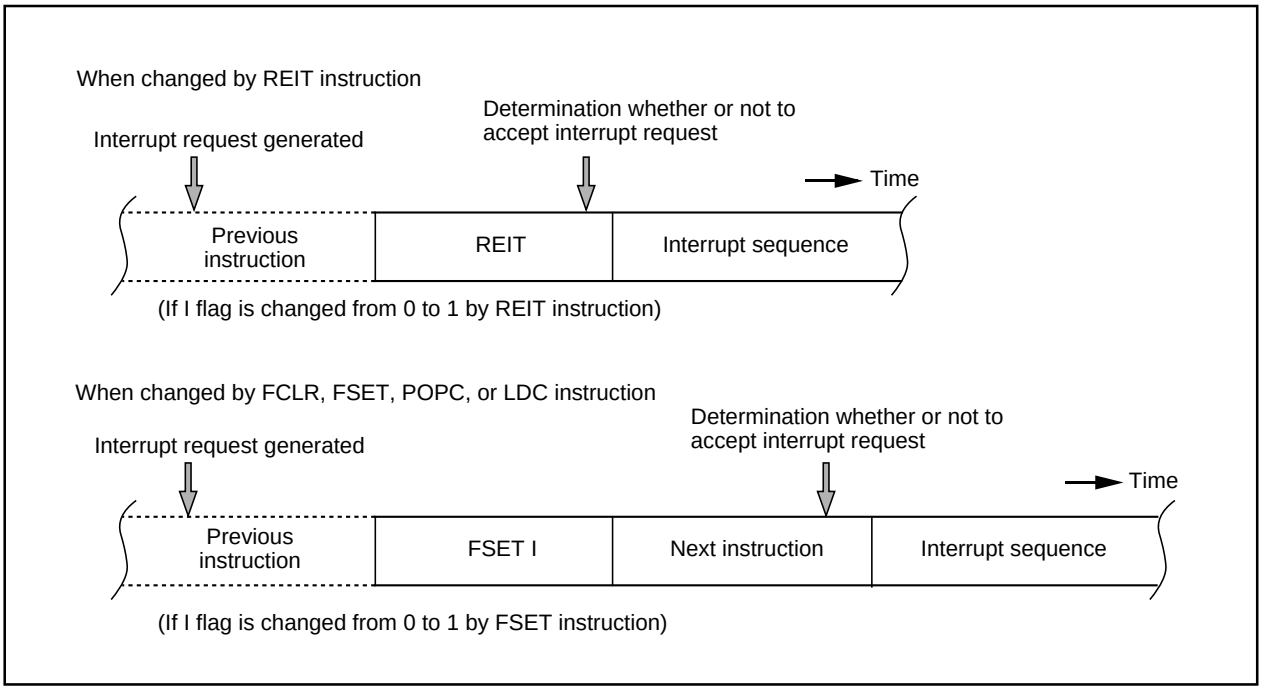


Figure 2.14.3. The timing of reflecting the change in the I flag to the interrupt

(3) Interrupt Request Bit

The interrupt request bit is set to "1" by hardware when an interrupt is requested. After the interrupt is accepted and jumps to the corresponding interrupt vector, the request bit is set to "0" by hardware. The interrupt request bit can also be set to "0" by software. (Do not set this bit to "1").

(4) Interrupt Priority Level Select Bit and Processor Interrupt Priority Level (IPL)

Set the interrupt priority level using the interrupt priority level select bit, which is one of the component bits of the interrupt control register. When an interrupt request occurs, the interrupt priority level is compared with the IPL. The interrupt is enabled only when the priority level of the interrupt is higher than the IPL. Therefore, setting the interrupt priority level to “0” disables the interrupt.

Table 2.14.1 shows the settings of interrupt priority levels and Table 2.14.2 shows the interrupt levels enabled, according to the contents of the IPL.

The following are conditions under which an interrupt is accepted:

· interrupt enable flag (I flag) = 1

· interrupt request bit = 1

· interrupt priority level > IPL

The interrupt enable flag (I flag), the interrupt request bit, the interrupt priority select bit, and the IPL are independent, and they are not affected by one another.

Table 2.14.1. Settings of interrupt priority levels

Interrupt priority level select bit	Interrupt priority level	Priority order
b2 b1 b0 0 0 0	Level 0 (interrupt disabled)	———
0 0 1	Level 1	Low
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	High

Table 2.14.2. Interrupt levels enabled according to the contents of the IPL

IPL	Enabled interrupt priority levels
IPL_2 IPL_1 IPL_0 0 0 0	Interrupt levels 1 and above are enabled
0 0 1	Interrupt levels 2 and above are enabled
0 1 0	Interrupt levels 3 and above are enabled
0 1 1	Interrupt levels 4 and above are enabled
1 0 0	Interrupt levels 5 and above are enabled
1 0 1	Interrupt levels 6 and above are enabled
1 1 0	Interrupt levels 7 and above are enabled
1 1 1	All maskable interrupts are disabled

When either the IPL or the interrupt priority level is changed, the new level is reflected to the interrupt in the following timing:

- When changing the IPL using the REIT instruction, the reflection takes effect as of the instruction that is executed in 2 clock cycles after the last clock cycle in volved in the REIT instruction.
- When changing the IPL using either the POPC, LDC or LDIPL instruction, the reflection takes effect as of the instruction that is executed in 3 cycles after the last clock cycle involved in the instruction used.
- When changing the interrupt priority level using the MOV or similar instruction, the reflection takes effect as of the instruction that is executed in 2 clock cycles after the last clock cycle involved in the instruction used.

(5) Interrupt Priority

If there are two or more interrupt requests occurring at a point in time within a single sampling (checking whether interrupt requests are made), the interrupt assigned a higher priority is accepted.

Assign an arbitrary priority to maskable interrupts (peripheral I/O interrupts) using the interrupt priority level select bit. If the same interrupt priority level is assigned, however, the interrupt assigned a higher hardware priority is accepted.

Priorities of the special interrupts, such as Reset (dealt with as an interrupt assigned the highest priority), watchdog timer interrupt, etc. are regulated by hardware.

Figure 2.14.4 shows the priorities of hardware interrupts.

Software interrupts are not affected by the interrupt priority. If an instruction is executed, control branches invariably to the interrupt routine.

Reset > $\overline{\text{NMI}}$ > $\overline{\text{DBC}}$ > Watchdog timer > Peripheral I/O > Single step > Address match

Figure 2.14.4. Hardware interrupts priorities

(6) Interrupt resolution circuit

When two or more interrupts are generated simultaneously, this circuit selects the interrupt with the highest priority level. Figure 2.14.5 shows the circuit that judges the interrupt priority level.

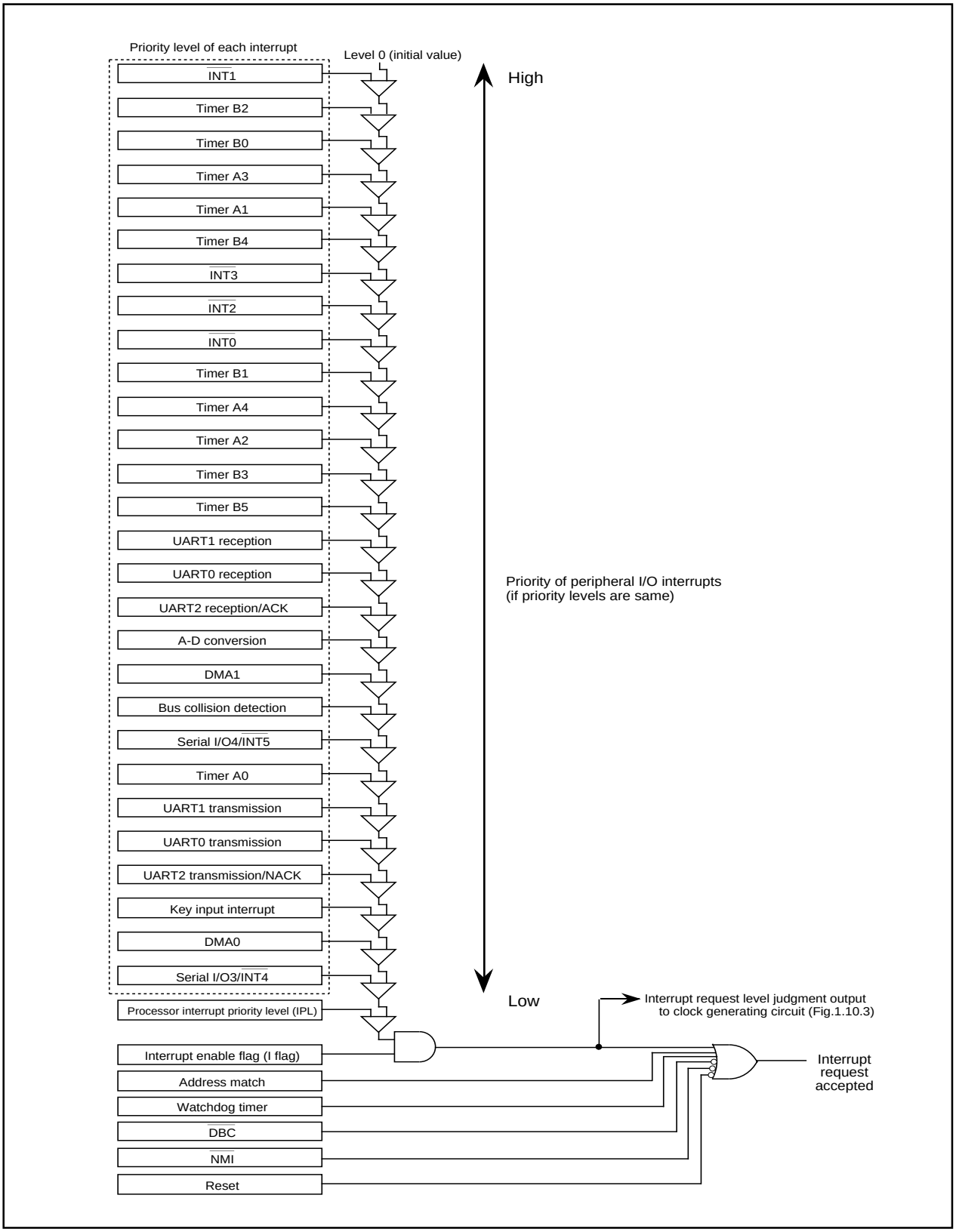


Figure 2.14.5. Interrupts resolution circuit

2.14.2 Multiple Interrupts Operation

The state when control branched to an interrupt routine is described below:

- The interrupt enable flag (I flag) is set to “0” (the interrupt is disabled).
- The interrupt request bit of the accepted interrupt is set to “0”.
- The processor interrupt priority level (IPL) is assigned to the same interrupt priority level as as signed to the accepted interrupt.

Setting the interrupt enable flag (I flag) to “1” within an interrupt routine allows an interrupt request assigned a priority higher than the IPL to be accepted.
An interrupt request that is not accepted because of low priority will be held. If the condition following is met when the REIT instruction returns the IPL and the interrupt priority is determined, then the interrupt request being held is accepted.

Interrupt priority level of the interrupt request being held > Returned the IPL

Figure 2.14.6 shows the example of the multiple interrupts operation.

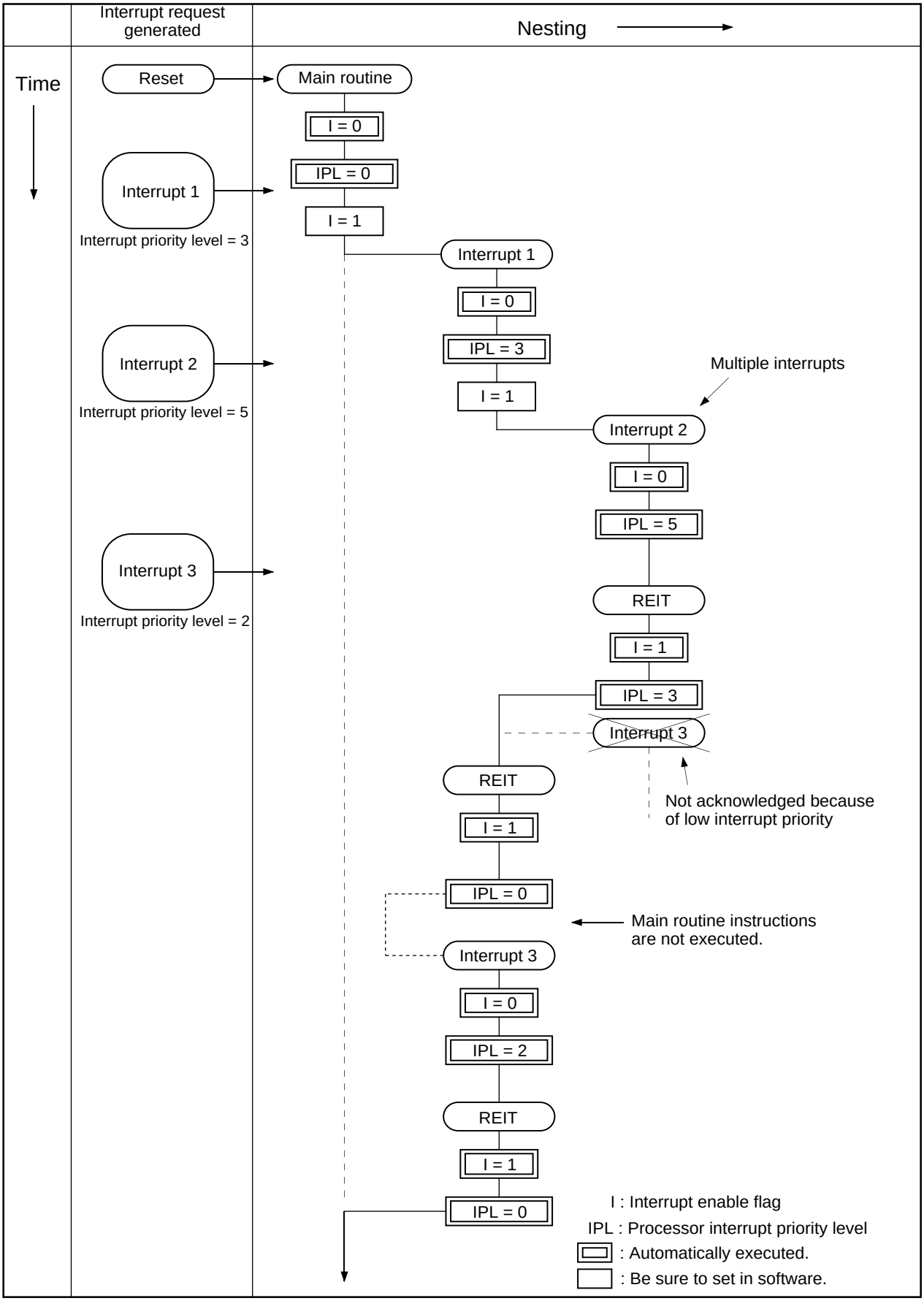


Figure 2.14.6. Example of the multiple interrupts operation

2.15 Power Control Usage

2.15.1 Overview of the power control usage

'Power Control' refers to the reduction of CPU power consumption by stopping the CPU and oscillators, or decreasing the operation clock. The following is a description of the three available power control modes:

(1) Modes

Power control is available in three modes.

(a) Normal operation mode

• High-speed mode

Divide-by-1 frequency of the main clock becomes the BCLK. The CPU operates with the BCLK selected. Each peripheral function operates according to its assigned clock.

• Medium-speed mode

Divide-by-2, divide-by-4, divide-by-8, or divide-by-16 frequency of the main clock becomes the BCLK. The CPU operates according to the BCLK selected. Each peripheral function operates according to its assigned clock.

• Low-speed mode

fc becomes the BCLK. The CPU operates according to the fc clock. The fc clock is supplied by the secondary clock. Each peripheral function operates according to its assigned clock.

• Low power consumption mode

The main clock operating in low-speed mode is stopped. The CPU operates according to the fc clock. The fc clock is supplied by the secondary clock. The only peripheral functions that operate are those with the sub-clock selected as the count source.

(b) Wait mode

The CPU operation is stopped. The oscillators do not stop.

(c) Stop mode

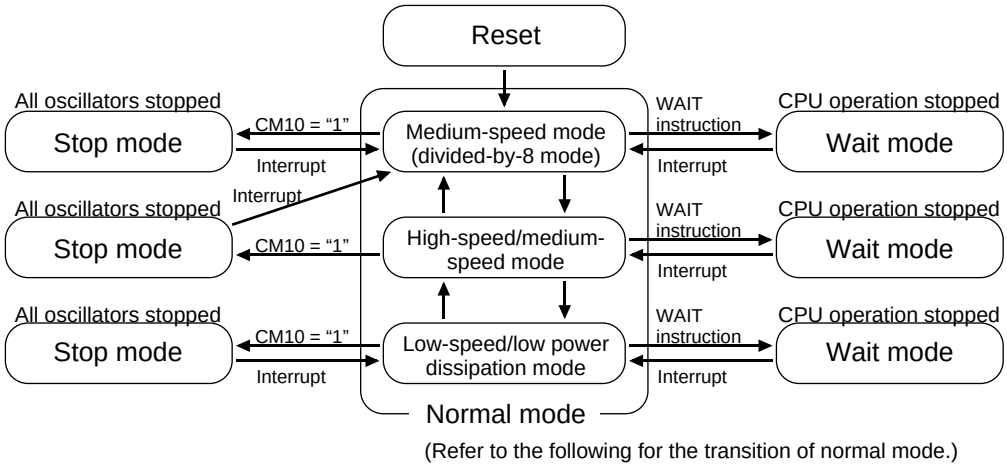
All oscillators stop. The CPU and all built-in peripheral functions stop. This mode, among the three modes listed here, is the most effective in decreasing power consumption.

Figure 2.15.1 is the state transition diagram of the above modes.

(2) Switching the driving capacity of the oscillation circuit

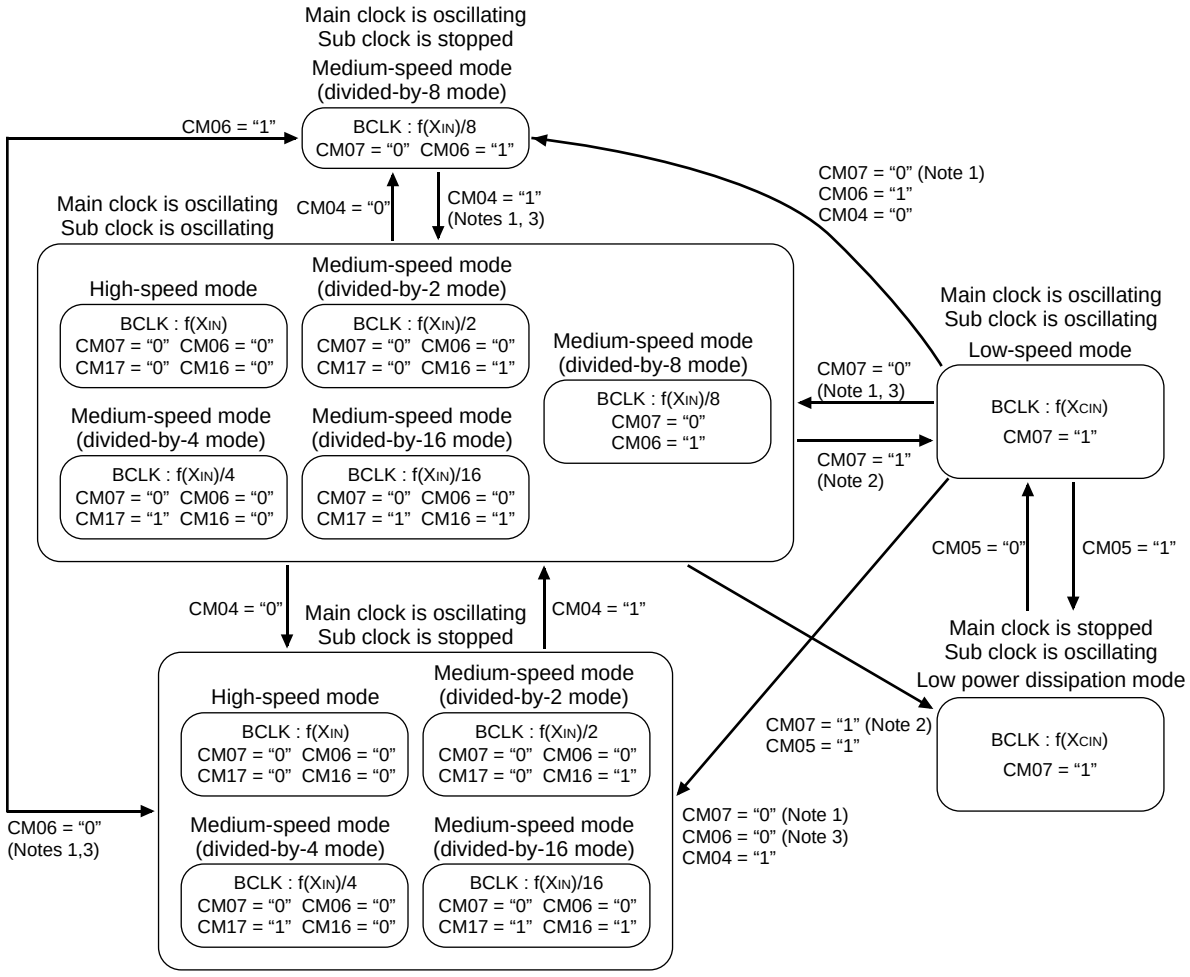
Both the main clock and the secondary clock have the ability to switch the driving capacity. Reducing the driving capacity after the oscillation stabilizes allows for further reduction in power consumption.

Transition of stop mode, wait mode



(Refer to the following for the transition of normal mode.)

Transition of normal mode



Note 1: Switch clock after oscillation of main clock is sufficiently stable.
Note 2: Switch clock after oscillation of sub clock is sufficiently stable.
Note 3: Change CM06 after changing CM17 and CM16.
Note 4: Transit in accordance with arrow.

Figure 2.15.1. State transition diagram of power control mode

(3) Clearing stop mode and wait mode

The stop mode and wait mode can be cleared by generating an interrupt request, or by resetting hardware. Set the priority level of the interrupt to be used for clearing, higher than the processor interrupt priority level (IPL), set that of the interrupt to "0" not to be used for clearing, and enable the interrupt enable flag (I flag). When an interrupt clears a mode, that interrupt is processed. Table 2.15.1 shows the interrupts that can be used for clearing a stop mode and wait mode.

(4) BCLK in returning from wait mode or stop mode

(a) Returning from wait mode

The processor immediately returns to the BCLK, which was in use before entering wait mode.

(b) Returning from stop mode

CM06 is set to "1" when the device enters stop mode after selecting the main clock for BCLK. CM17, CM16, and CM07 do not change state. In this case, when restored from stop mode, the device starts operating in divided-by-8 mode.

When the device enters stop mode after selecting the subclock for BCLK, CM06, CM17, CM16, and CM07 all do not change state. In this case, when restored from stop mode, the device starts operating in low-speed mode.

Table 2.15.1. Interrupts available for clearing stop mode and wait mode

Interrupt for clearing	Wait mode		Stop mode
	CM02 = 0	CM02 =1(Note 6), CM07 = 0, CM05 = 0	
Bus collision detection interrupt	Possible	Note 1	Note 1
DMA0 interrupt	Impossible	Impossible	Impossible
DMA1 interrupt	Impossible	Impossible	Impossible
Key input interrupt	Possible	Possible	Possible
A-D interrupt	Note 3	Impossible	Impossible
UART0 transmit interrupt	Possible	Note 1	Note 1
UART0 receive interrupt	Possible	Note 1	Note 1
UART1 transmit interrupt	Possible	Note 1	Note 1
UART1 receive interrupt	Possible	Note 1	Note 1
UART2 transmit interrupt	Possible	Note 1	Note 1
UART2 receive interrupt	Possible	Note 1	Note 1
SI/O3 interrupt	Possible	Note 4	Note 4
SI/O4 interrupt	Possible	Note 4	Note 4
Timer A0 interrupt	Possible	Note 2, Note 5	Note 2
Timer A1 interrupt	Possible	Note 2, Note 5	Note 2
Timer A2 interrupt	Possible	Note 2, Note 5	Note 2
Timer A3 interrupt	Possible	Note 2, Note 5	Note 2
Timer A4 interrupt	Possible	Note 2, Note 5	Note 2
Timer B0 interrupt	Possible	Note 2, Note 5	Note 2
Timer B1 interrupt	Possible	Note 2, Note 5	Note 2
Timer B2 interrupt	Possible	Note 2, Note 5	Note 2
Timer B3 interrupt	Possible	Note 2, Note 5	Note 2
Timer B4 interrupt	Possible	Note 2, Note 5	Note 2
Timer B5 interrupt	Possible	Note 2, Note 5	Note 2
$\overline{\text{INT0}}$ interrupt	Possible	Possible	Possible
$\overline{\text{INT1}}$ interrupt	Possible	Possible	Possible
$\overline{\text{INT2}}$ interrupt	Possible	Possible	Possible
$\overline{\text{INT3}}$ interrupt	Possible	Possible	Possible
$\overline{\text{INT4}}$ interrupt	Possible	Possible	Possible
$\overline{\text{INT5}}$ interrupt	Possible	Possible	Possible
$\overline{\text{NMI}}$ interrupt	Possible	Possible	Possible

Note 1: Can be used when an external clock is selected (UART0, UART1).
Can be used when an external clock in clock synchronous serial I/O mode is selected (UART2).
Note 2: Can be used when the external signal is being counted in event counter mode.
Note 3: Can be used in one-shot mode and one-shot sweep mode.
Note 4: Can be used when an external clock is selected.
Note 5: Can be used when count source is f_{C32}.
Note 6: When the MCU running in low-speed or low power dissipation mode, do not enter wait mode with CM02 is set to "1".

(5) Sequence of returning from stop mode

Sequence of returning from stop mode is oscillation start-up time and interrupt sequence.
When interrupt is generated in stop mode, CM10 becomes “0” and clearing stop mode.
Starting oscillation and supplying BCLK execute the interrupt sequence as follow:

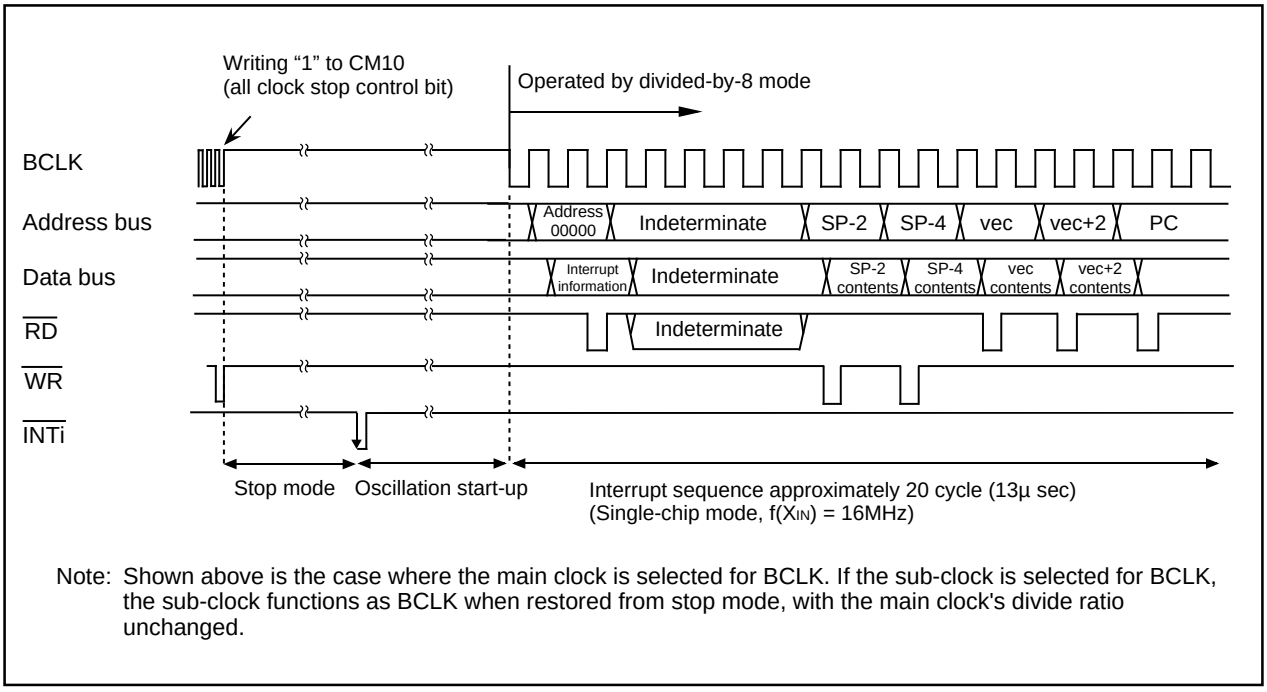
In the interrupt sequence, the processor carries out the following in sequence given:

(a) CPU gets the interrupt information (the interrupt number and interrupt request level) by reading address 00000_{16}. The interrupt request bit of the interrupt written in address 00000_{16} will then be set to “0”.
(b) Saves the content of the flag register (FLG) as it was immediately before the start of interrupt sequence in the temporary register (Note) within the CPU.
(c) Sets the interrupt enable flag (I flag), the debug flag (D flag), and the stack pointer assignment flag (U flag) to “0” (the U flag, however does not change if the INT instruction, in software interrupt numbers 32 through 63, is executed)
(d) Saves the content of the temporary register (Note) within the CPU in the stack area.
(e) Saves the content of the program counter (PC) in the stack area.
(f) Sets the interrupt priority level of the accepted instruction in the IPL.

Note: This register cannot be utilized by the user.

After the interrupt sequence is completed, the processor resumes executing instructions from the first address of the interrupt routine.

Figure 2.15.2 shows the sequence of returning from stop mode.



Note: Shown above is the case where the main clock is selected for BCLK. If the sub-clock is selected for BCLK, the sub-clock functions as BCLK when restored from stop mode, with the main clock's divide ratio unchanged.

Figure 2.15.2. Sequence of returning from stop mode

(6) Registers related to power control

Figure 2.15.3 shows the memory map of power control-related registers, and Figure 2.15.4 shows power control-related registers.

0006_{16}	System clock control register 0 (CM0)
0007_{16}	System clock control register 1 (CM1)

Figure 2.15.3. Memory map of power control-related registers

System clock control register 0 (Note 1)

b7	b6	b5	b4	b3	b2	b1	b0

Symbol	Address	When reset
CM0	000616	4816

Bit symbol	Bit name	Function	R	W
CM00	Clock output function select bit (Valid only in single-chip mode)	b1 b0 0 0 : I/O port P57 0 1 : fC output 1 0 : f8 output 1 1 : f32 output	O	O
CM01			O	O
CM02	WAIT peripheral function clock stop bit	0 : Do not stop peripheral function clock in wait mode 1 : Stop peripheral function clock in wait mode (Note 8)	O	O
CM03	XCIN-XCOUT drive capacity select bit (Note 2)	0 : LOW 1 : HIGH	O	O
CM04	Port XC select bit	0 : I/O port 1 : XCIN-XCOUT generation (Note 9)	O	O
CM05	Main clock (XIN-XOUT) stop bit (Note 3, 4, 5)	0 : On 1 : Off	O	O
CM06	Main clock division select bit 0 (Note 7)	0 : CM16 and CM17 valid 1 : Division by 8 mode	O	O
CM07	System clock select bit (Note 6)	0 : XIN, XOUT 1 : XCIN, XCOUT	O	O

Note 1: Set bit 0 of the protect register (address 000A16) to “1” before writing to this register.
Note 2: Changes to “1” when shiffing to stop mode and at a reset.
Note 3: When entering low power dissipation mode, main clock stops by using this bit. To stop the main clock, when the sub clock oscillation is stable, set system clock select bit (CM07) to “1” before setting this bit to “1”.
Note 4: When inputting external clock, only clock oscillation buffer is stopped and clock input is acceptable.
Note 5: If this bit is set to “1”, XOUT turns “H”. The built-in feedback resistor remains being connected, so XIN turns pulled up to XOUT (“H”) via the feedback resistor.
Note 6: Set port XC select bit (CM04) to “1” and stabilize the sub-clock oscillating before setting this bit from “0” to “1”. Do not write to both bits at the same time. And also, set the main clock stop bit (CM05) to “0” and stabilize the main clock oscillating before setting this bit from “1” to “0”.
Note 7: This bit changes to “1” when shifting from high-speed/medium-speed mode to stop mode and at a reset. When shifting from low-speed/low power dissipation mode to stop mode, the value before stop mode is retained.
Note 8: fC32 is not included. Do not set to “1” when using low-speed or low power dissipation mode.
Note 9: When the XCIN/XCOUT is used, set ports P86 and P87 as the input ports without pull-up.

System clock control register 1 (Note 1)

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

Symbol	Address	When reset
CM1	000716	2016

Bit symbol	Bit name	Function	R	W
CM10	All clock stop control bit (Note4)	0 : Clock on 1 : All clocks off (stop mode)	O	O
Reserved bit		Must always be set to “0”	O	O
Reserved bit		Must always be set to “0”	O	O
Reserved bit		Must always be set to “0”	O	O
Reserved bit		Must always be set to “0”	O	O
CM15	XIN-XOUT drive capacity select bit (Note 2)	0 : LOW 1 : HIGH	O	O
CM16	Main clock division select bit 1 (Note 3)	b7 b6 0 0 : No division mode 0 1 : Division by 2 mode 1 0 : Division by 4 mode 1 1 : Division by 16 mode	O	O
CM17				

Note 1: Set bit 0 of the protect register (address 000A16) to “1” before writing to this register.
Note 2: This bit changes to “1” when shifting from high-speed/medium-speed mode to stop mode and at a reset. When shifting from low-speed/low power dissipation mode to stop mode, the value before stop mode is retained.
Note 3: Can be selected when bit 6 of the system clock control register 0 (address 000616) is “0”. If “1”, division mode is fixed at 8.
Note 4: If this bit is set to “1”, XOUT turns “H”, and the built-in feedback resistor is cut off. XCIN and XCOUT turn high-impedance state.

Figure 2.15.4. Power control-related registers

2.15.2 Stop Mode Set-Up

Settings and operation for entering stop mode are described here.

Operation (1) Enables the interrupt used for returning from stop mode.
(2) Sets the interrupt enable flag (I flag) to “1”.
(3) Clearing the protection and setting all clock stop control bit to “1” stops oscillation and causes the processor to go into stop mode.

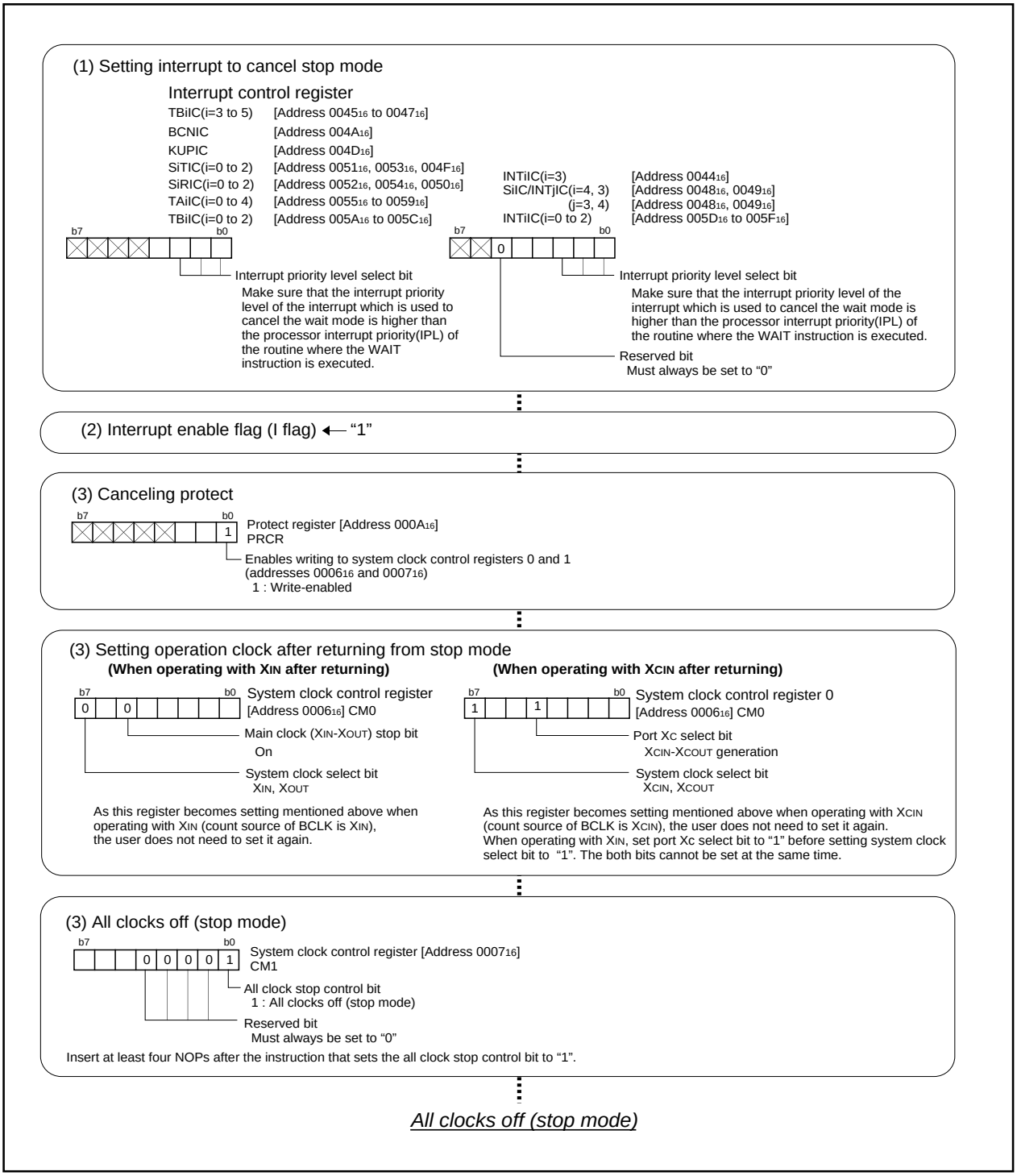


Figure 2.15.5. Example of stop mode set-up

2.15.3 Wait Mode Set-Up

Settings and operation for entering wait mode are described here.

Operation (1) Enables the interrupt used for returning from wait mode.
(2) Sets the interrupt enable flag (I flag) to “1”.
(3) Clears the protection and changes the content of the system clock control register.
(4) Executes the WAIT instruction.

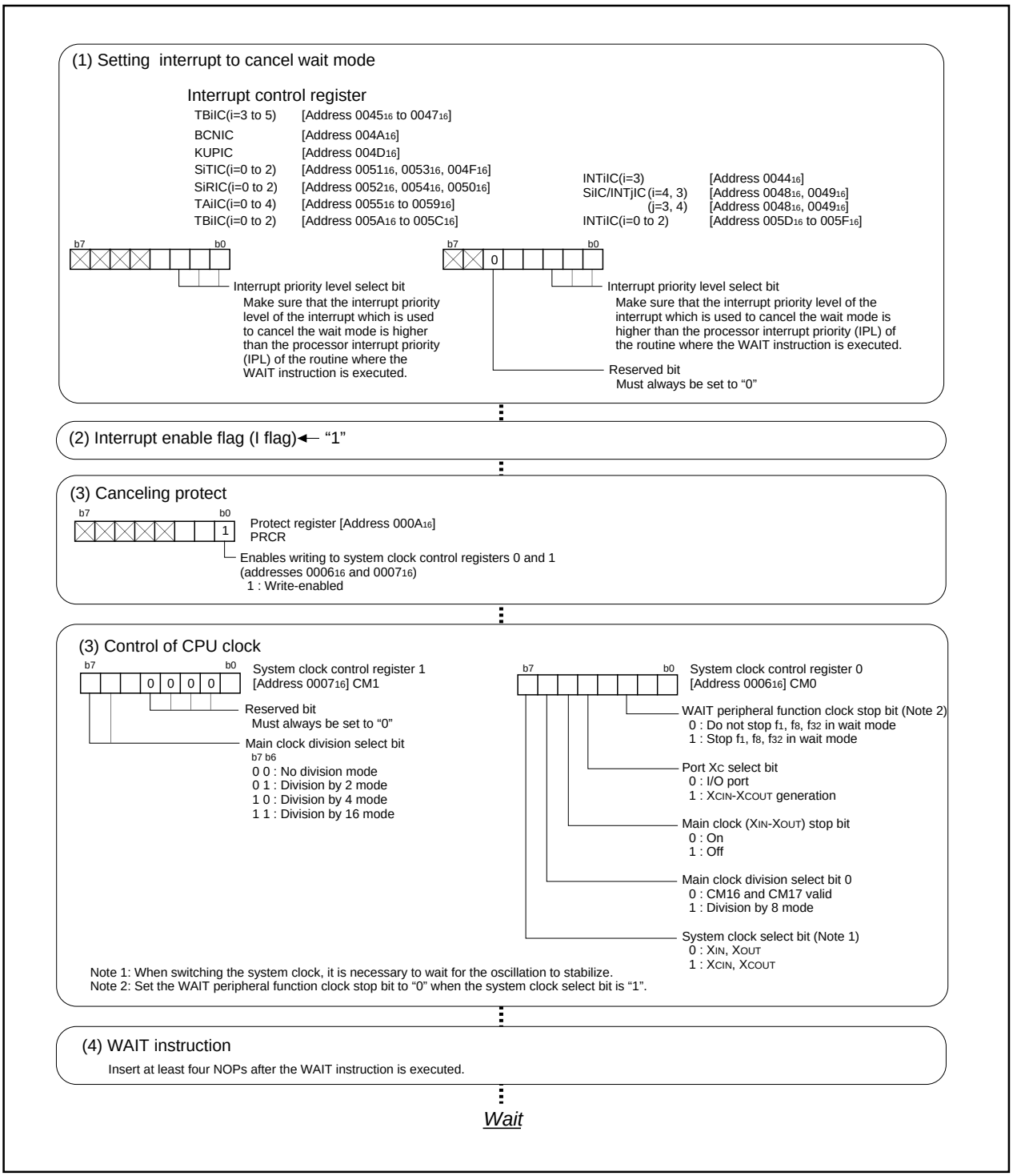


Figure 2.15.6. Example of wait mode set-up

2.15.4 Precautions in Power Control

(1) The processor does not switch to stop mode when the $\overline{NMI}$ pin is at "L" level.

(2) When returning from stop mode by hardware reset, $\overline{RESET}$ pin must be set to "L" level until main clock oscillation is stabilized.

(3) When switching to either wait mode or stop mode, instructions occupying four bytes either from the WAIT instruction or from the instruction that sets the all clock stop control bit to "1" within the instruction queue are prefetched and then the program stops. So put at least four NOPs in succession either to the WAIT instruction or to the instruction that sets the all clock stop control bit to "1".

(4) Before the count source for BCLK can be changed from XIN to XCIN or vice versa, the clock to which the count source is going to be switched must be oscillating stably. Allow a wait time in software for the oscillation to stabilize before switching over the clock.

(5) Suggestions to reduce power consumption

(a) Ports

The processor retains the state of each programmable I/O port even when it goes to wait mode or to stop mode. A current flows in active I/O ports. A pass current flows in input ports that float. When entering wait mode or stop mode, set non-used ports to input and stabilize the potential.

(b) A-D converter

A current always flows in the VREF pin. When entering wait mode or stop mode, set the Vref connection bit to "0" so that no current flows into the VREF pin.

(c) D-A converter

The processor retains the D-A state even when entering wait mode or stop mode. Disable the output from the D-A converter then work on the programmable I/O ports. Set D-A register to "00_{16}".

(d) Stopping peripheral functions

In wait mode, stop non-used peripheral functions using the WAIT peripheral function clock stop bit. However, peripheral function clock f_{C32} does not stop so that the peripherals using f_{C32} do not contribute to the power saving. When the MCU running in low-speed or low power dissipation mode, do not enter WAIT mode with this bit set to "1".

(e) Switching the oscillation-driving capacity

Set the driving capacity to "LOW" when oscillation is stable.

(f) External clock

When using an external clock input for the CPU clock, set the main clock stop bit to "1". Setting the main clock stop bit to "1" causes the XOUT pin not to operate and the power consumption goes down (when using an external clock input, the clock signal is input regardless of the content of the main clock stop bit).

2.16 Programmable I/O Ports Usage

2.16.1 Overview of the programmable I/O ports usage

Eighty-seven programmable I/O ports and one input-only port are available. I/O pins also serve as I/O pins for built-in peripheral functions.

Each port has a direction register that defines the I/O direction and also has a port register for I/O data. In addition, each port has a pull-up control register that defines pull-up in terms of 4 bits. The input-only port has neither direction register nor pull-up control bit.

The following is an overview of the programmable I/O ports usage:

(1) Writing to a port register

With the direction register set to output, the level of the written values from each relevant pin is output by writing to a port register. The output level conforms to CMOS output. Port $P7_0$ and $P7_1$ are N channel open drain. Writing to the port register, with the direction register set to input, inputs a value to the port register, but nothing is output to the relevant pins. The output level remains floating.

In memory expansion and microprocessor mode, the contents of corresponding port and direction registers of pins A0 to A19, D0 to D15, $\overline{CS0}$ to $\overline{CS3}$, $\overline{RD}$, $\overline{WRL}$/$\overline{WR}$, $\overline{WRH}$/$\overline{BHE}$, ALE, $\overline{RDY}$, $\overline{HOLD}$, $\overline{HLDA}$ and BCLK cannot be modified.

(2) Reading a port register

With the direction register set to output, reading a port register takes out the content of the port register, not the content of the pin. With the direction register set to input, reading the port register takes out the content of the pin.

(3) Effect of the protection register

Data written to the direction register of P9 is affected by the protection register. The direction register of P9 cannot be easily rewritten.

(4) Input-only port

$P8_5$ is used as the input-only port, it also serves as $\overline{NMI}$. $P8_5$ has no direction register. Pull-up cannot be set to this port. As $\overline{NMI}$ cannot be disabled, an $\overline{NMI}$ interrupt occurs if a falling edge is input to $P8_5$. Use $P8_5$ for reading the level input at this time only.

(5) Setting pull-up

The pull-up control bit allows setting of the pull-up, in terms of 4 bits, either in use or not in use. For the four bits chosen, pull-up is effective only in the ports whose direction register is set to input. Pull-up is not effective in ports whose direction register is set to output.

Do not set pull-up of corresponding pin when XCIN/XCOUT is set or a port is used as A-D input.

Pull-up can be set for P0 to P3, $P4_0$ to $P4_3$, P5 in only single-chip mode. Pull-up cannot be set for P0 to P3, $P4_0$ to $P4_3$, P5 in memory expansion and microprocessor modes. The contents of register can be changed, but the pull-up resistance is not connected.

(6) I/O functions of built-in peripheral devices

Table 2.16.1 shows relation between ports and I/O functions of built-in peripheral devices.

Table 2.16.1. Relation between ports and I/O functions of built-in peripheral devices

Port	Internal peripheral device I/O pins
P15 to P17	Input pins for external interrupt
P6	I/O pins for serial I/O communication
P70	I/O pins for serial I/O communication/Timer A I/O pin
P71	I/O pins for serial I/O communication/Timer A I/O pins/Timer B input pin
P72 to P73	I/O pins for serial I/O communication/Timer A I/O pins/Three-phase motor control output pins
P74 to P75	Timer A I/O pins/Three-phase motor control output pins
P76 to P77	Timer A I/O pins
P80, P81	Timer A I/O pins
P82 to P84	Input pins for external interrupt
P86, P87	Sub-clock oscillation circuit I/O pins
P90 to P92	Timer B input pins
P93, P94	D-A converter output pins
P95, P96	A-D converter extended input pins
P97	A-D trigger input pin
P100 to P103	A-D converter input pins
P104 to P107	A-D converter input pins / key-input interrupt function input pins

(7) Examples of working on non-used pins

Table 2.16.2 contains examples of working on non-used pins. There are shown here for mere examples. In practical use, make suitable changes and perform sufficient evaluation in compliance with you application.

(a) Single-chip mode

Table 2.16.2. Examples of working on unused pins in single-chip mode

Pin name	Connection
Ports P0 to P10 (excluding P85)	After setting for input mode, connect every pin to VSS or VCC via a resistor; or after setting for output mode, leave these pins open. (Note 1, Note 3)
XOUT (Note 2)	Open
$\overline{\text{NMI}}$	Connect to VCC via a resistor
AVCC	Connect to VCC
AVSS, VREF, BYTE	Connect to VSS

Note 1: If setting these pins in output mode and opening them, ports are in input mode until switched into output mode by use of software after reset. Thus the voltage levels of the pins become unstable, and there can be instances in which the power source current increases while the ports are in input mode.
In view of an instance in which the contents of the direction registers change due to a runaway generated by noise or other causes, setting the contents of the direction registers periodically by use of software increases program reliability.

Note 2: When an external clock is input to the XIN pin.

Note 3: Output "L" if port P70 and P71 are set to output mode.
Port P70 and P71 are N channel open drain output.

(b) Memory expansion mode, microprocessor mode

Table 2.16.3. Examples of working on unused pins in memory expansion mode or microprocessor mode

Pin name	Connection
Ports P6 to P10 (excluding $P8_5$)	After setting for input mode, connect every pin to V_{SS} or V_{CC} via a resistor; or after setting for output mode, leave these pins open. (Note 1, Note 2, Note 5)
$\overline{BHE}$, ALE, $\overline{HLDA}$	Open (Note 3)
X_{OUT} (Note 4), BCLK (Note 6)	Open
$\overline{HOLD}$, $\overline{RDY}$, $\overline{NMI}$	Connect via resistor to V_{CC} (pull-up)
AV_{CC}	Connect to V_{CC}
AV_{SS}, V_{REF}	Connect to V_{SS}

Note 1: If setting these pins in output mode and opening them, ports are in input mode until switched into output mode by use of software after reset. Thus the voltage levels of the pins become unstable, and there can be instances in which the power source current increases while the ports are in input mode.
In view of an instance in which the contents of the direction registers change due to a runaway generated by noise or other causes, setting the contents of the direction registers periodically by use of software increases program reliability.

Note 2: Make wiring as short as possible (not more than 2 cm from the microcomputer's pins) in working on non-used pins.

Note 3: When a V_{SS} level is connected to the CNV_{SS} pin, these pins are input ports until the processor mode is switched by use of software after reset. Thus the voltage levels of the pins destabilize, and there can be an increase in the power source current while these pins are input ports.

Note 4: When an external clock is input to the X_{IN} pin.

Note 5: Output "L" if port $P7_0$ and $P7_1$ are set to output mode.
Port $P7_0$ and $P7_1$ are N channel open drain output.

Note 6: When the BCLK output disable bit (bit 7 at address 0004_{16}) is set to “1”, connect to V_{CC} via a resistor (pull-up).

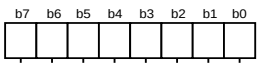
(8) Registers related to the programmable I/O ports

Figure 2.16.1 shows the memory map of programmable I/O ports-related registers, and Figures 2.16.2 to 2.16.4 show programmable I/O ports-related registers.

03E0$_{16}$	Port P0 register (P0)
03E1$_{16}$	Port P1 register (P1)
03E2$_{16}$	Port P0 direction register (PD0)
03E3$_{16}$	Port P1 direction register (PD1)
03E4$_{16}$	Port P2 register (P2)
03E5$_{16}$	Port P3 register (P3)
03E6$_{16}$	Port P2 direction register (PD2)
03E7$_{16}$	Port P3 direction register (PD3)
03E8$_{16}$	Port P4 register (P4)
03E9$_{16}$	Port P5 register (P5)
03EA$_{16}$	Port P4 direction register (PD4)
03EB$_{16}$	Port P5 direction register (PD5)
03EC$_{16}$	Port P6 register (P6)
03ED$_{16}$	Port P7 register (P7)
03EE$_{16}$	Port P6 direction register (PD6)
03EF$_{16}$	Port P7 direction register (PD7)
03F0$_{16}$	Port P8 register (P8)
03F1$_{16}$	Port P9 register (P9)
03F2$_{16}$	Port P8 direction register (PD8)
03F3$_{16}$	Port P9 direction register (PD9)
03F4$_{16}$	Port P10 register (P10)
03F5$_{16}$	
03F6$_{16}$	Port P10 direction register (PD10)
≈	≈
03FC$_{16}$	Pull-up control register 0 (PUR0)
03FD$_{16}$	Pull-up control register 1 (PUR1)
03FE$_{16}$	Pull-up control register 2 (PUR2)
03FF$_{16}$	Port control register (PCR)

Figure 2.16.1. Memory map of programmable I/O ports-related registers

Port Pi direction register (Note 1, 2)

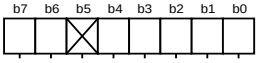


Symbol	Address	When reset
PDi (i = 0 to 10, except 8)	03E2$_{16}$, 03E3$_{16}$, 03E6$_{16}$, 03E7$_{16}$, 03EA$_{16}$ 03EB$_{16}$, 03EE$_{16}$, 03EF$_{16}$, 03F3$_{16}$, 03F6$_{16}$	00$_{16}$

Bit symbol	Bit name	Function	R	W
PDi_0	Port Pi$_0$ direction register	0 : Input mode (Functions as an input port) 1 : Output mode (Functions as an output port) (i = 0 to 10 except 8)	○	○
PDi_1	Port Pi$_1$ direction register		○	○
PDi_2	Port Pi$_2$ direction register		○	○
PDi_3	Port Pi$_3$ direction register		○	○
PDi_4	Port Pi$_4$ direction register		○	○
PDi_5	Port Pi$_5$ direction register		○	○
PDi_6	Port Pi$_6$ direction register		○	○
PDi_7	Port Pi$_7$ direction register		○	○

Note 1: Set bit 2 of protect register (address 000A$_{16}$) to “1” before rewriting to the port P9 direction register.
Note 2: In memory expansion and microprocessor mode, the contents of corresponding port Pi direction register of pins $\overline{A_0}$ to $\overline{A_{19}}$, D$_0$ to D$_{15}$, $\overline{CS_0}$ to $\overline{CS_3}$, $\overline{RD}$, WRL/$\overline{WR}$, WRH/$\overline{BHE}$, ALE, RDY, HOLD, HLDA and BCLK cannot be modified.

Port P8 direction register

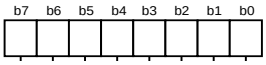


Symbol	Address	When reset
PD8	03F2$_{16}$	00X00000$_2$

Bit symbol	Bit name	Function	R	W
PD8_0	Port P8$_0$ direction register	0 : Input mode (Functions as an input port) 1 : Output mode (Functions as an output port)	○	○
PD8_1	Port P8$_1$ direction register		○	○
PD8_2	Port P8$_2$ direction register		○	○
PD8_3	Port P8$_3$ direction register		○	○
PD8_4	Port P8$_4$ direction register		○	○
	Nothing is assigned. In an attempt to write to this bit, write “0”. The value, if read, turns out to be indeterminate.		—	—
PD8_6	Port P8$_6$ direction register	0 : Input mode (Functions as an input port) 1 : Output mode (Functions as an output port)	○	○
PD8_7	Port P8$_7$ direction register		○	○

Figure 2.16.2. Programmable I/O ports-related registers (1)

Port Pi register (Note 2)

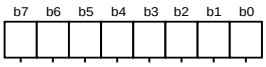


Symbol	Address	When reset
Pi (i = 0 to 10, except 8)	03E0$_{16}$, 03E1$_{16}$, 03E4$_{16}$, 03E5$_{16}$, 03E8$_{16}$ 03E9$_{16}$, 03EC$_{16}$, 03ED$_{16}$, 03F1$_{16}$, 03F4$_{16}$	Indeterminate Indeterminate

Bit symbol	Bit name	Function	R	W
Pi_0	Port Pi$_0$ register	Data is input and output to and from each pin by reading and writing to and from each corresponding bit 0 : “L” level data 1 : “H” level data (Note 1) (i = 0 to 10 except 8)	○	○
Pi_1	Port Pi$_1$ register		○	○
Pi_2	Port Pi$_2$ register		○	○
Pi_3	Port Pi$_3$ register		○	○
Pi_4	Port Pi$_4$ register		○	○
Pi_5	Port Pi$_5$ register		○	○
Pi_6	Port Pi$_6$ register		○	○
Pi_7	Port Pi$_7$ register		○	○

Note 1: Since P7$_0$ and P7$_1$ are N-channel open drain ports, the data is high-impedance.
Note 2: In memory expansion and microprocessor mode, the contents of corresponding port Pi register of pins A$_0$ to A$_{19}$, D$_0$ to D$_{15}$, $\overline{CS_0}$ to $\overline{CS_3}$, $\overline{RD}$, WRL/$\overline{WR}$, WRH/$\overline{BHE}$, ALE, $\overline{RDY}$, $\overline{HOLD}$, HLDA and BCLK cannot be modified.

Port P8 register

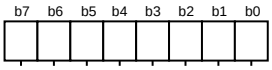


Symbol	Address	When reset
P8	03F0$_{16}$	Indeterminate

Bit symbol	Bit name	Function	R	W
P8_0	Port P8$_0$ register	Data is input and output to and from each pin by reading and writing to and from each corresponding bit (except for P8$_5$) 0 : “L” level data 1 : “H” level data	○	○
P8_1	Port P8$_1$ register		○	○
P8_2	Port P8$_2$ register		○	○
P8_3	Port P8$_3$ register		○	○
P8_4	Port P8$_4$ register		○	○
P8_5	Port P8$_5$ register		○	×
P8_6	Port P8$_6$ register		○	○
P8_7	Port P8$_7$ register		○	○

Figure 2.16.3. Programmable I/O ports-related registers (2)

Pull-up control register 0 (Note)

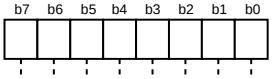


Symbol: PUR0 — Address: 03FC16 — When reset: 0016

Bit symbol	Bit name	Function	R	W
PU00	P00 to P03 pull-up	The corresponding port is pulled high with a pull-up resistor 0 : Not pulled high 1 : Pulled high	O	O
PU01	P04 to P07 pull-up		O	O
PU02	P10 to P13 pull-up		O	O
PU03	P14 to P17 pull-up		O	O
PU04	P20 to P23 pull-up		O	O
PU05	P24 to P27 pull-up		O	O
PU06	P30 to P33 pull-up		O	O
PU07	P34 to P37 pull-up		O	O

Note : In memory expansion and microprocessor mode, the content of this register can be changed, but the pull-up resistance is not connected.

Pull-up control register 1

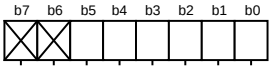


Symbol: PUR1 — Address: 03FD16 — When reset: 0016 (Note 2)

Bit symbol	Bit name	Function	R	W
PU10	P40 to P43 pull-up (Note 3)	The corresponding port is pulled high with a pull-up resistor 0 : Not pulled high 1 : Pulled high	O	O
PU11	P44 to P47 pull-up		O	O
PU12	P50 to P53 pull-up (Note 3)		O	O
PU13	P54 to P57 pull-up		O	O
PU14	P60 to P63 pull-up		O	O
PU15	P64 to P67 pull-up		O	O
PU16	P72 to P73 pull-up (Note 1)		O	O
PU17	P74 to P77 pull-up		O	O

Note 1: Since P70 and P71 are N-channel open drain ports, pull-up is not available for them.
Note 2: When the VCC level is being impressed to the CNVSS terminal, this register becomes to 0216 when reset (PU11 becomes to “1”).
Note 3: In memory expansion and microprocessor mode, the content of these bits can be changed, but the pull-up resistance is not connected.

Pull-up control register 2



Symbol: PUR2 — Address: 03FE16 — When reset: 0016

Bit symbol	Bit name	Function	R	W
PU20	P80 to P83 pull-up	The corresponding port is pulled high with a pull-up resistor 0 : Not pulled high 1 : Pulled high	O	O
PU21	P84 to P87 pull-up (Except P85)		O	O
PU22	P90 to P93 pull-up		O	O
PU23	P94 to P97 pull-up		O	O
PU24	P100 to P103 pull-up		O	O
PU25	P104 to P107 pull-up		O	O
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be “0”.			—	—
			—	—

Figure 2.16.4. Programmable I/O ports-related registers (3)

[MEMO]

Chapter 3

Examples of Peripheral functions Applications

This chapter presents applications in which peripheral functions built in the M16C/62A are used. They are shown here as examples. In practical use, make suitable changes and perform sufficient evaluation. For basic use, see Chapter 2 Peripheral Functions Usage.

Here follows the list of applications that appear in this chapter.

• 3.1 Long-period timers P2-192
• 3.2 Variable-period variable-duty PWM output P2-196
• 3.3 Delayed one-shot output P2-200
• 3.4 Buzzer output P2-204
• 3.5 Solution for external interrupt pins shortage P2-206
• 3.6 Memory to memory DMA transfer P2-208
• 3.7 Controlling power using stop mode P2-212
• 3.8 Controlling power using wait mode P2-216

[MEMO]

3.1 Long-Period Timers

Overview In this process, Timer A0 and Timer A1 are connected to make a 16-bit timer with a 16-bit prescaler. Figure 3.1.1 shows the operation timing, Figure 3.1.2 shows the connection diagram, and Figures 3.1.3 and 3.1.4 show the set-up procedure.

Use the following peripheral functions:

- Timer mode of timer A
- Event counter mode of timer A

Specifications

(1) Set timer A0 to timer mode, and set timer A1 to event counter mode.

(2) Perform a count on count source f_1 using timer A0 to count for 1 ms, and perform a count on timer A0 using timer A1 to count for 1 second.

(3) Connect a 16-MHz oscillator to XIN.

Operation (1) Setting the count start flag to “1” causes the counter to begin counting. The counter of timer A0 performs a down count on count source f_1.

(2) If the counter of timer A0 underflows, the counter reloads the content of the reload register and continues counting. At this time, the timer A0 interrupt request bit goes to “1”. The counter of timer A1 performs a down count on underflows in timer A0.

(3) If the counter of timer A1 underflows, the counter reloads the content of the reload register and continues counting. At this time, the timer A1 interrupt request bit goes to “1”.

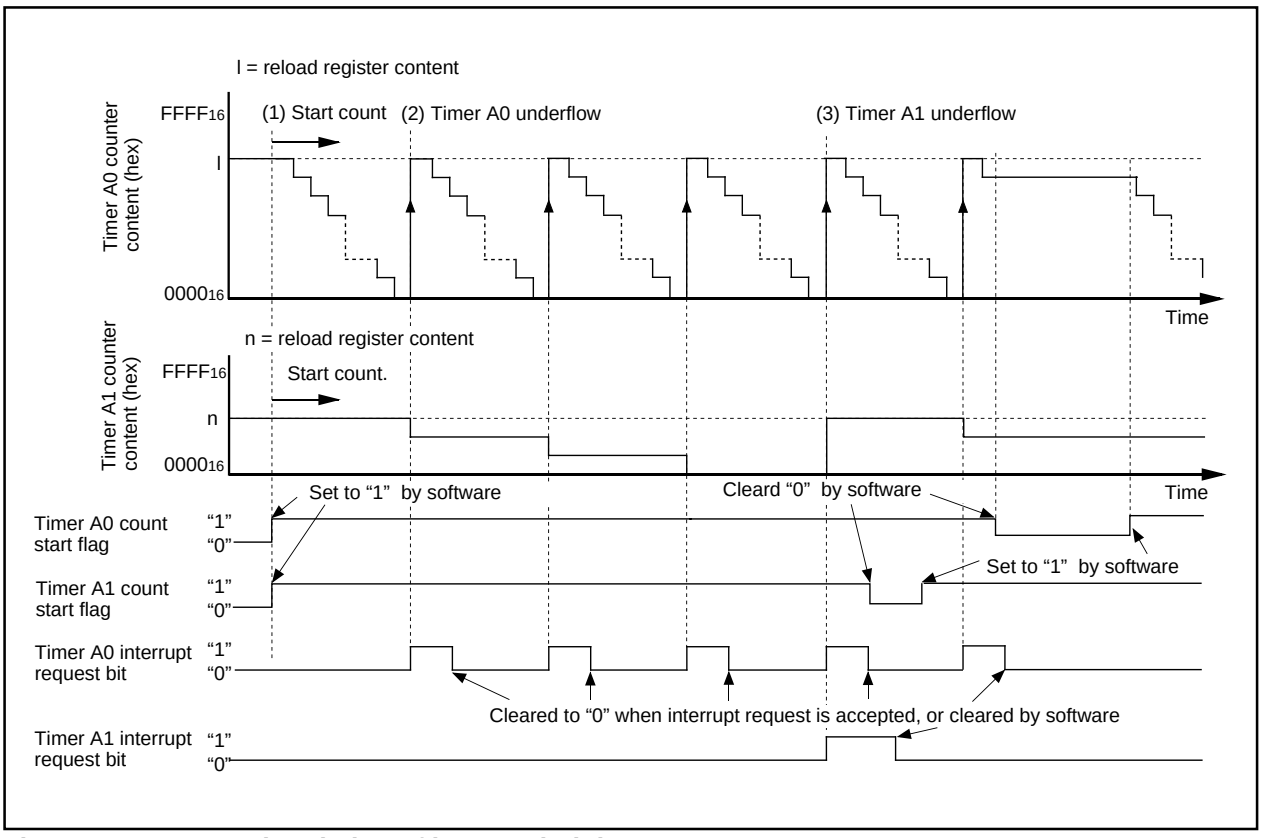


Figure 3.1.1. Operation timing of long-period timers

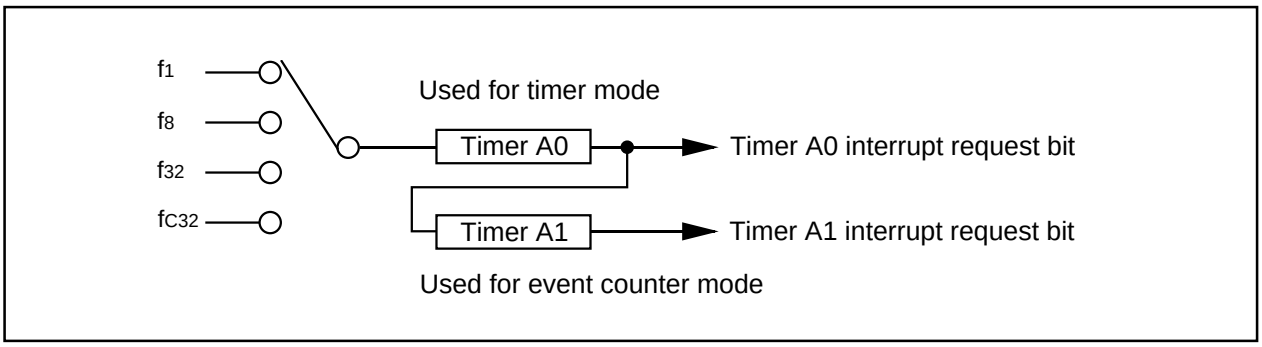
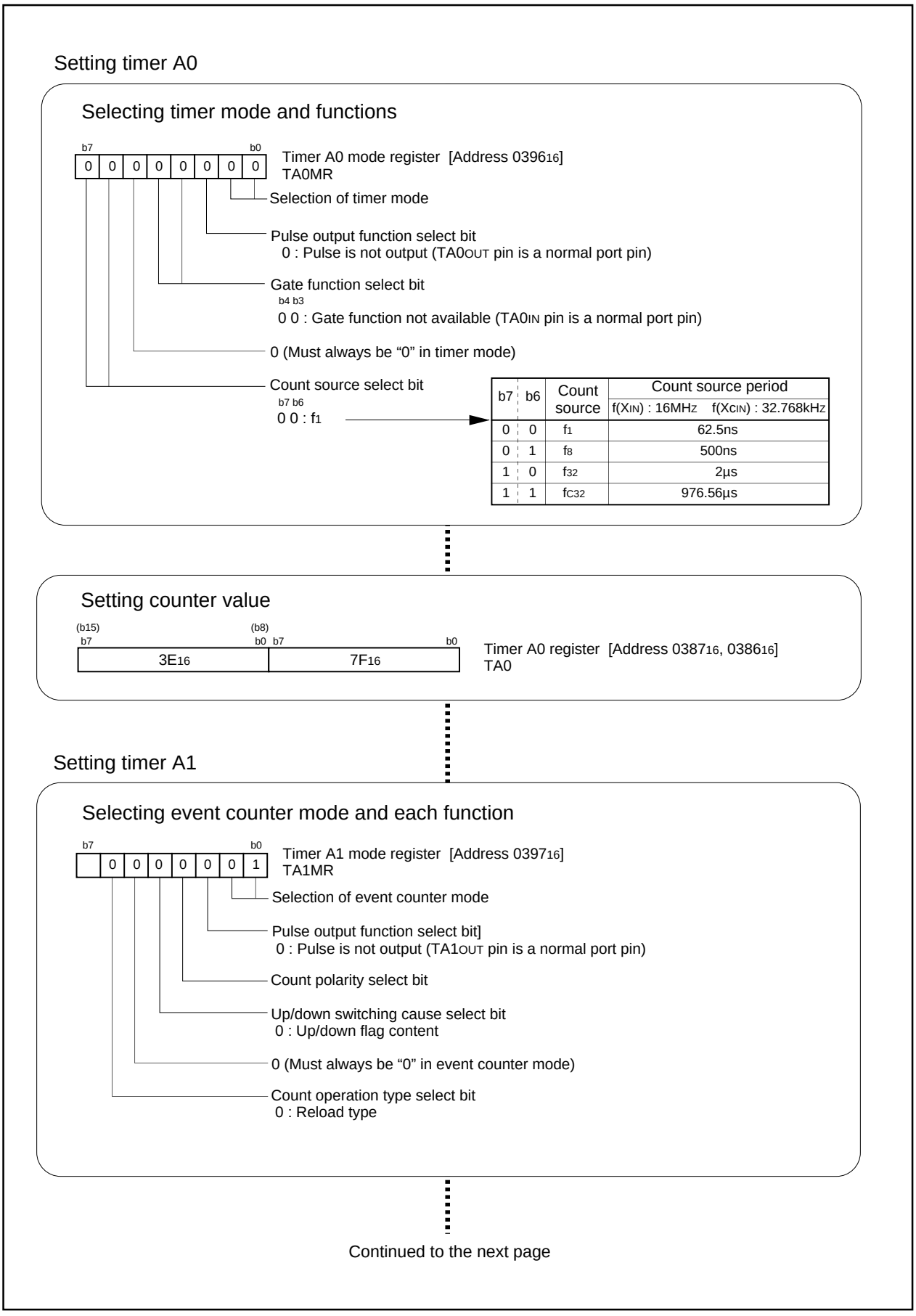


Figure 3.1.2. Connection diagram of long-period timers



Continued to the next page

Figure 3.1.3. Set-up procedure of long-period timers (1)

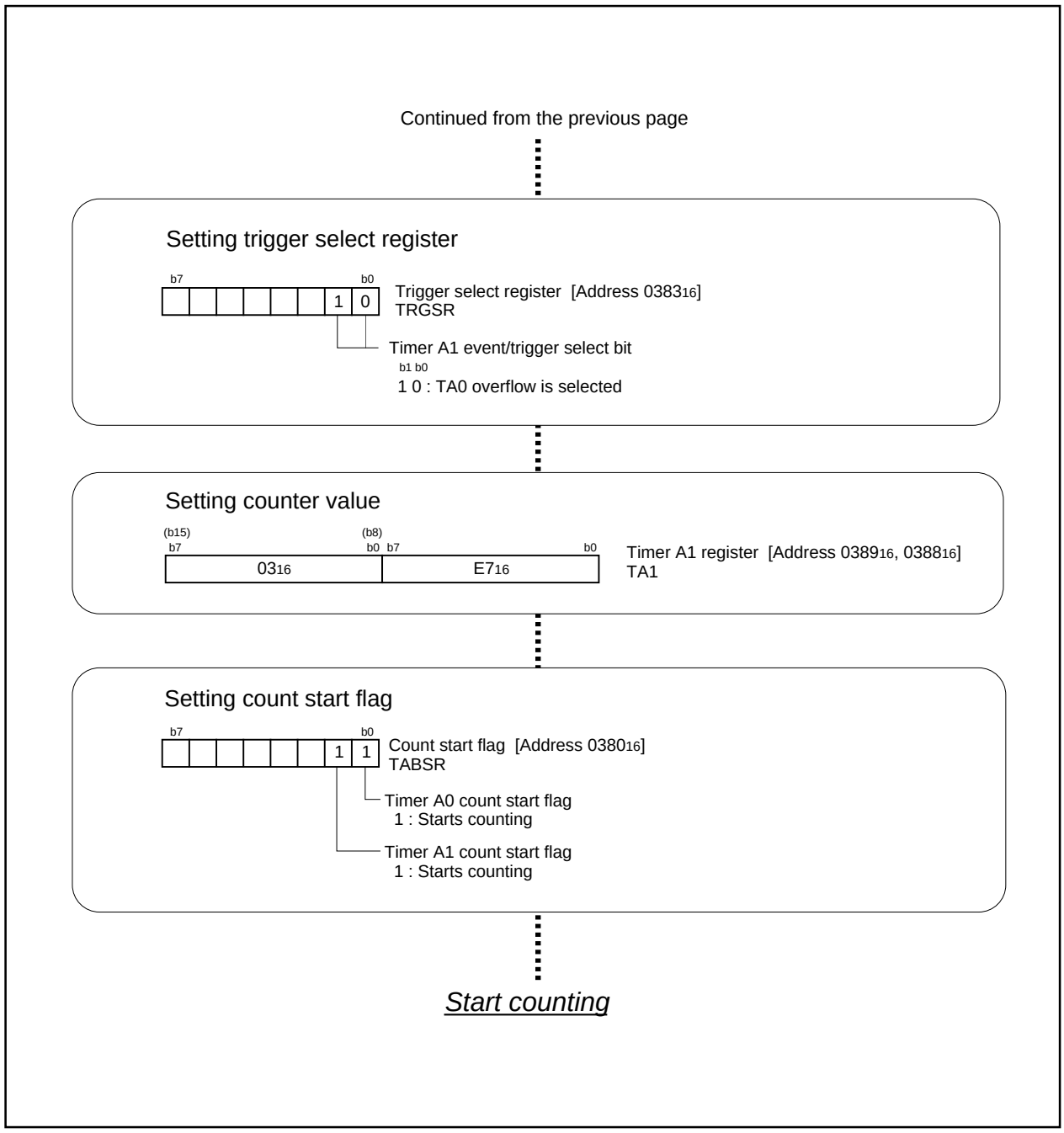


Figure 3.1.4. Set-up procedure of long-period timers (2)

3.2 Variable-Period Variable-Duty PWM Output

Overview In this process, Timer A0 and A1 are used to generate variable-period, variable-duty PWM output. Figure 3.2.1 shows the operation timing, Figure 3.2.2 shows the connection diagram, and Figures 3.2.3 and 3.2.4 show the set-up procedure.

Use the following peripheral functions:

- Timer mode of timer A
- One-shot timer mode of timer A

Specifications

(1) Set timer A0 in timer mode, and set timer A1 in one-shot timer mode with pulse-output function.

(2) Set 1 ms, the PWM period, to timer A0. Set 500 μs, the width of PWM "H" pulse, to timer A1. Both timer A0 and timer A1 use f_1 for the count source.

(3) Connect a 16-MHz oscillator to X_{IN}.

Operation (1) Setting the count start flag to "1" causes the counter of timer A0 to begin counting. The counter of timer A0 performs a down count on count source f_1.

(2) If the counter of timer A0 underflows, the counter reloads the content of the reload register and continues counting. At this time, the timer A0 interrupt request bit goes to "1".

(3) An underflow in timer A0 triggers the counter of timer A1 and causes it to begin counting. When the counter of timer A1 begins counting, the output level of the $TA1_{OUT}$ pin goes to "H".

(4) As soon as the count of the counter of timer A1 becomes "0000_{16}", the output level of $TA1_{OUT}$ pin goes to "L", and the counter reloads the content of the reload register and stops counting. At the same time, the timer A1 interrupt request bit goes to "1".

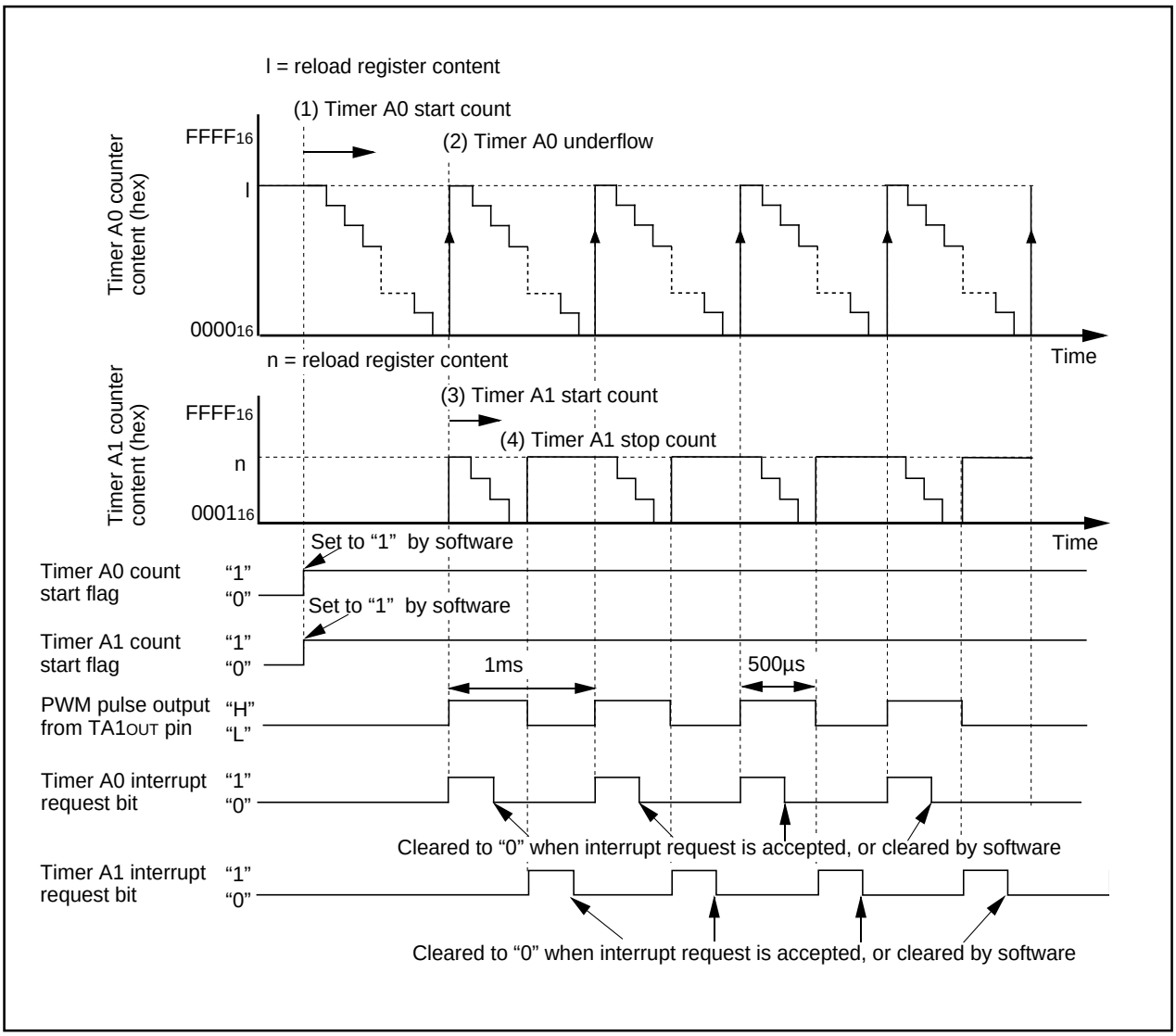


Figure 3.2.1. Operation timing of variable-period variable-duty PWM output

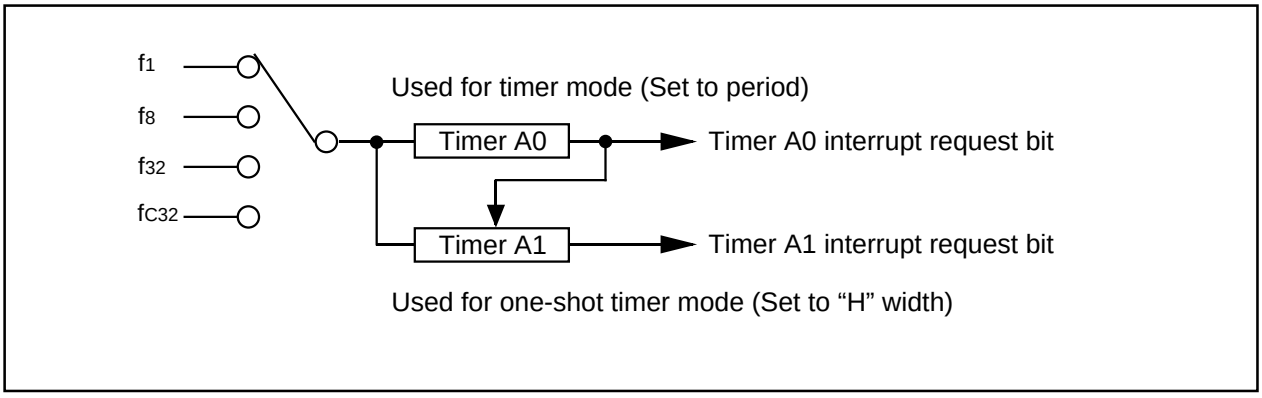


Figure 3.2.2. Connection diagram of variable-period variable-duty PWM output

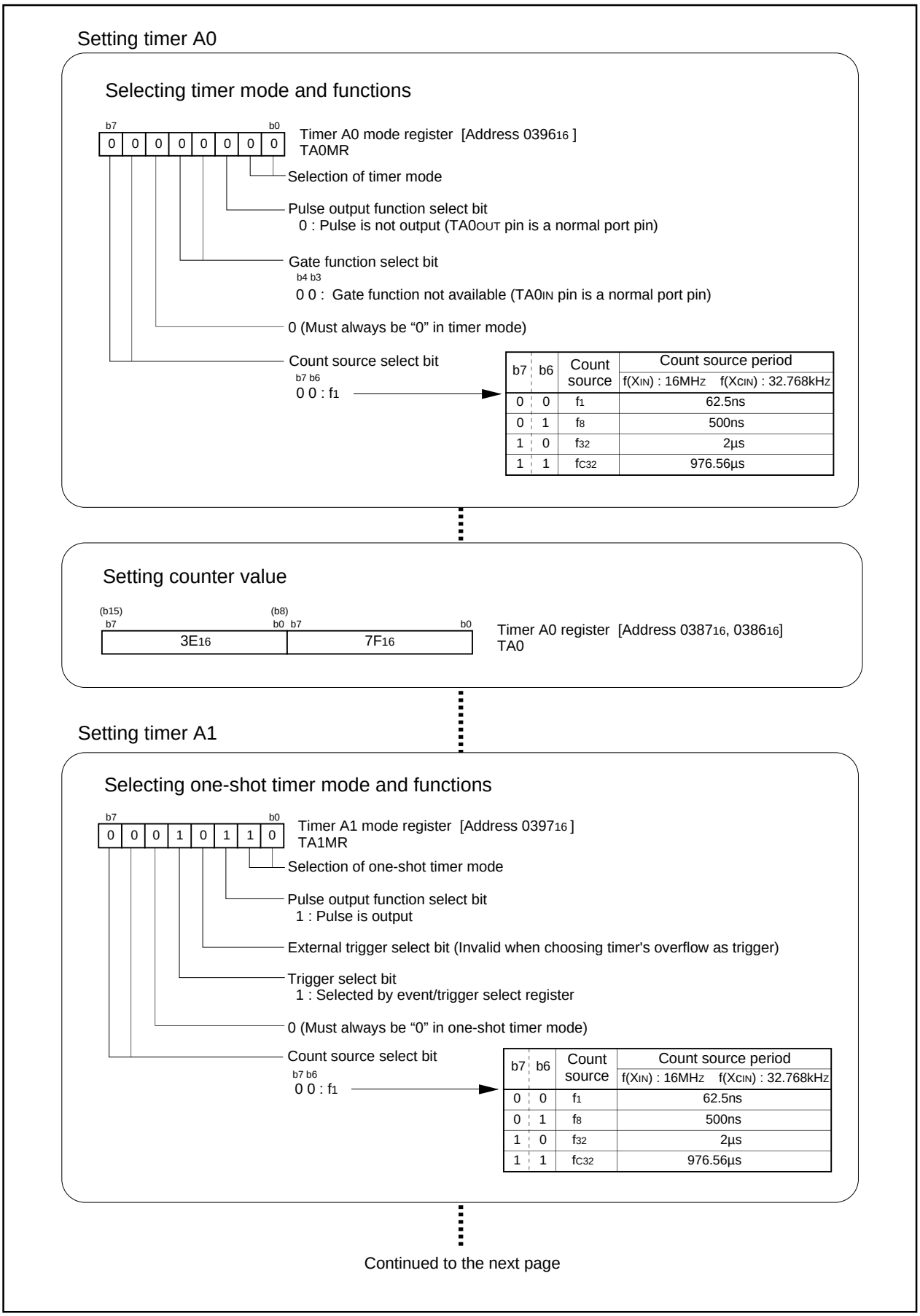


Figure 3.2.3. Set-up procedure of variable-period variable-duty PWM output (1)

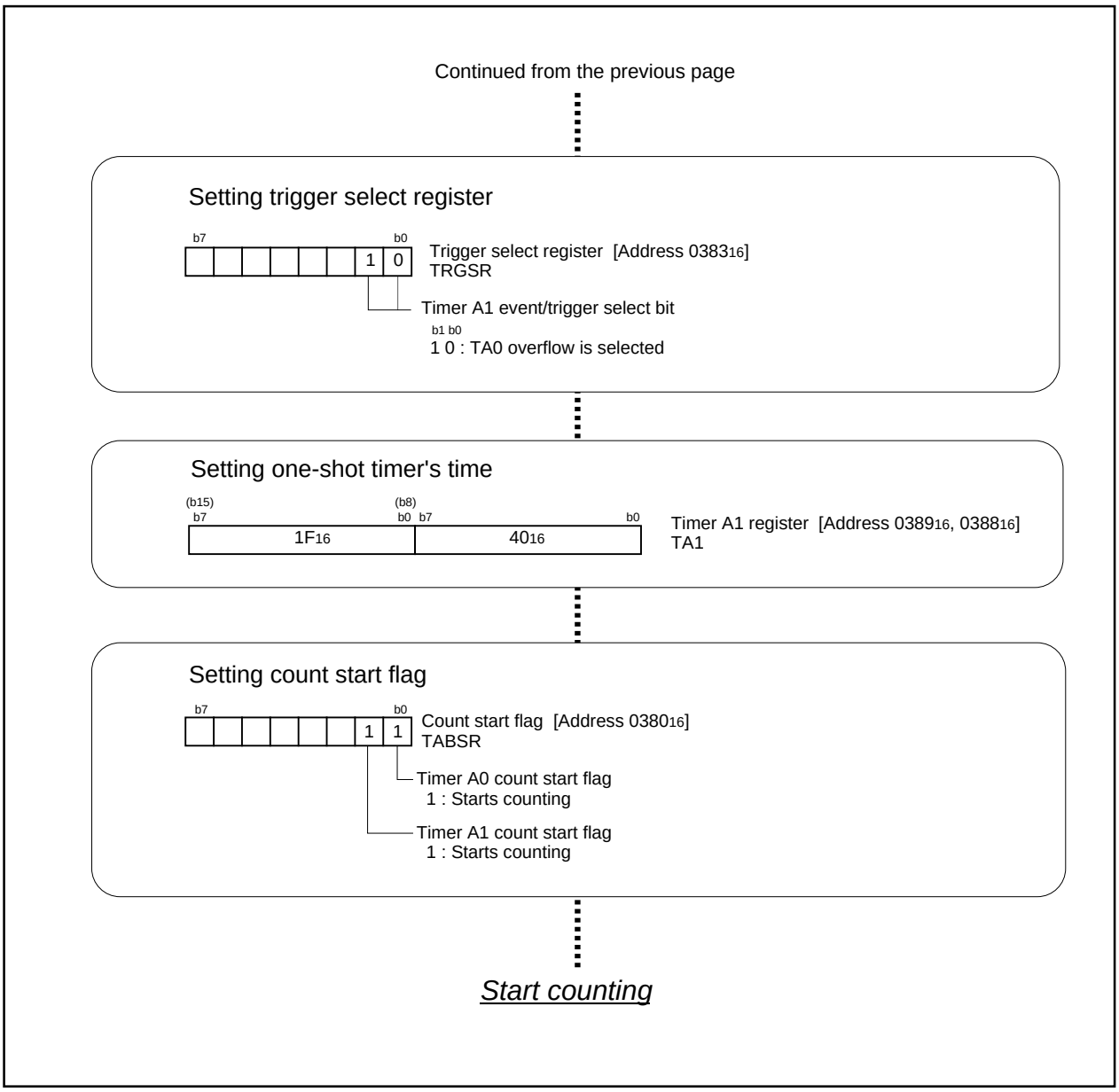


Figure 3.2.4. Set-up procedure of variable-period variable-duty PWM output (2)

3.3 Delayed One-Shot Output

Overview The following are steps of outputting a pulse only once after a specified elapse since an external trigger is input. Figure 3.3.1 shows the operation timing, Figure 3.3.2 shows the connection diagram, and Figures 3.3.3 and 3.3.4 show the set-up procedure.

Use the following peripheral function:

- One-shot timer mode of timer A

Specifications

(1) Set timer A0 in one-shot timer mode, and set timer A1 in one-shot timer mode with pulse-output function.

(2) Set 1 ms, an interval before a pulse is output, in timer A0; and set 50 μs, a pulse width, in timer A1. Both timer A0 and timer A1 use f_1 for the count source.

(3) Connect a 16-MHz oscillator to XIN.

Operation (1) Setting the trigger select bit to “1” and setting the count start flag to “1” enables the counter of timer A0 to count.

(2) If an effective edge, selected by use of the external trigger select bit, is input to the TA0IN pin, the counter begins a down count. The counter of timer A0 performs a down count on count source f_1.

(3) As soon as the counter of timer A0 becomes “0000_{16}”, the counter reloads the content of the reload register and stops counting. At this time, the timer A0 interrupt request bit goes to “1”.

(4) An underflow in timer A0 triggers the counter of timer A1 and causes it to begin counting. When timer A1 begins counting, the output level of the TA1OUT pin goes to “H”.

(5) As soon as the counter of timer A1 becomes “0000_{16}”, the output level of the TA1OUT pin goes to “L”, the counter reloads the content of the reload register, and stops counting. At this time, timer A1 interrupt request bit goes to “1”.

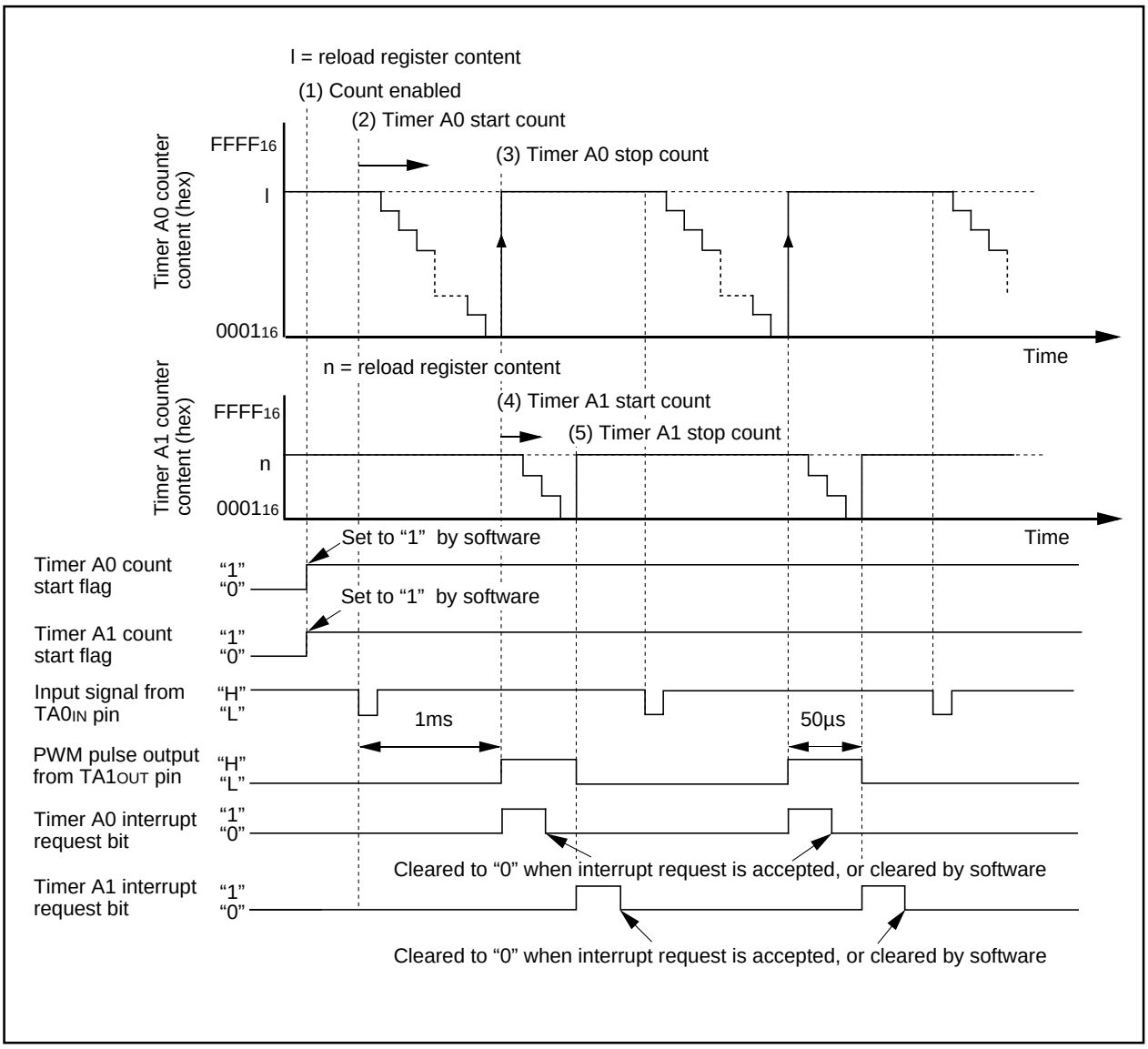


Figure 3.3.1. Operation timing of delayed one-shot output

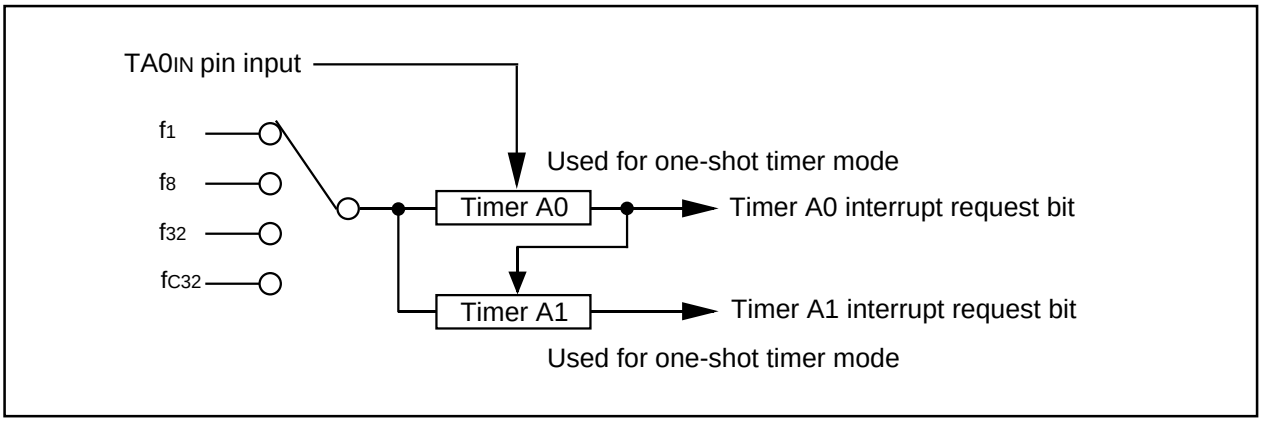


Figure 3.3.2. Connection diagram of delayed one-shot output

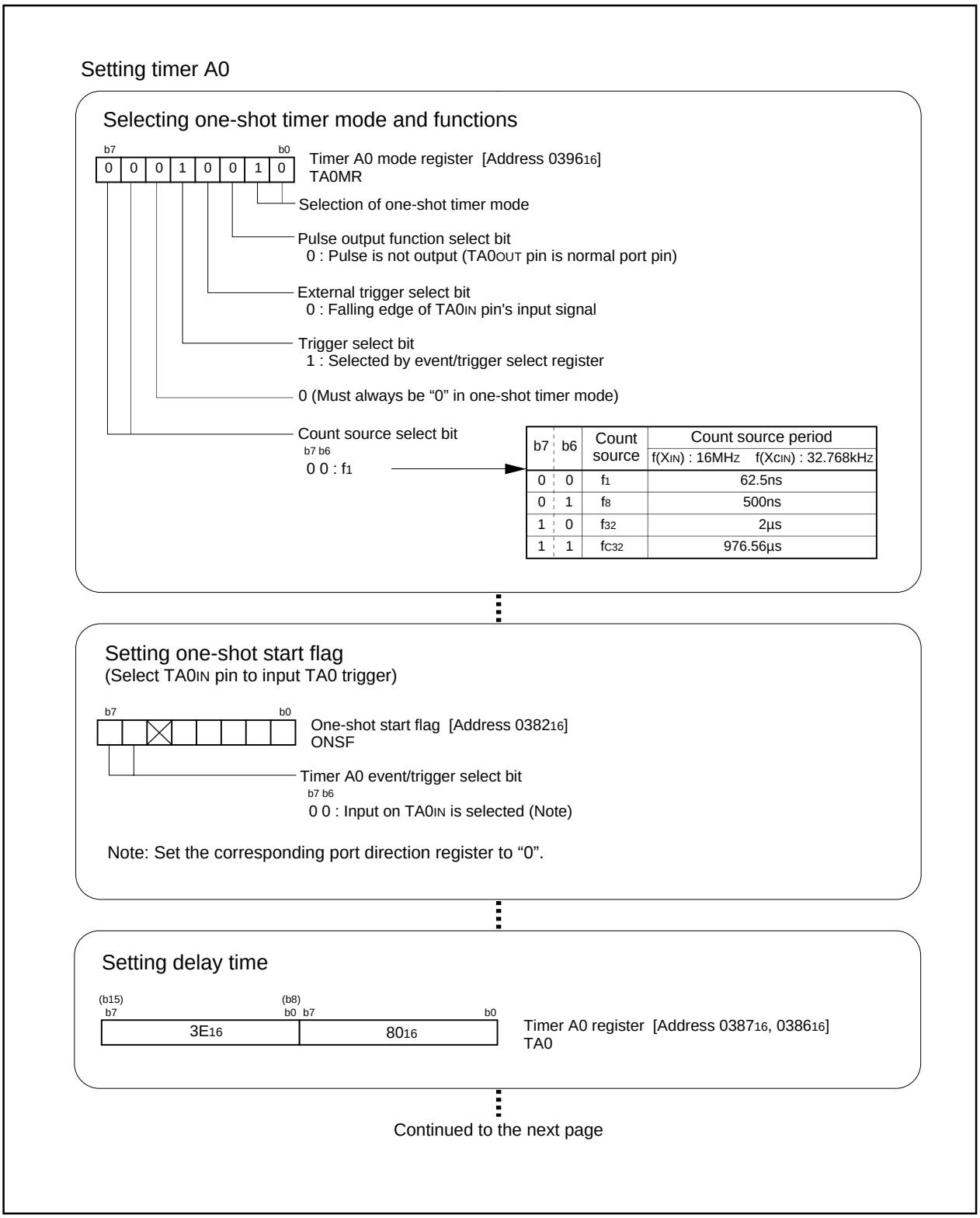


Figure 3.3.3. Set-up procedure of delayed one-shot output (1)

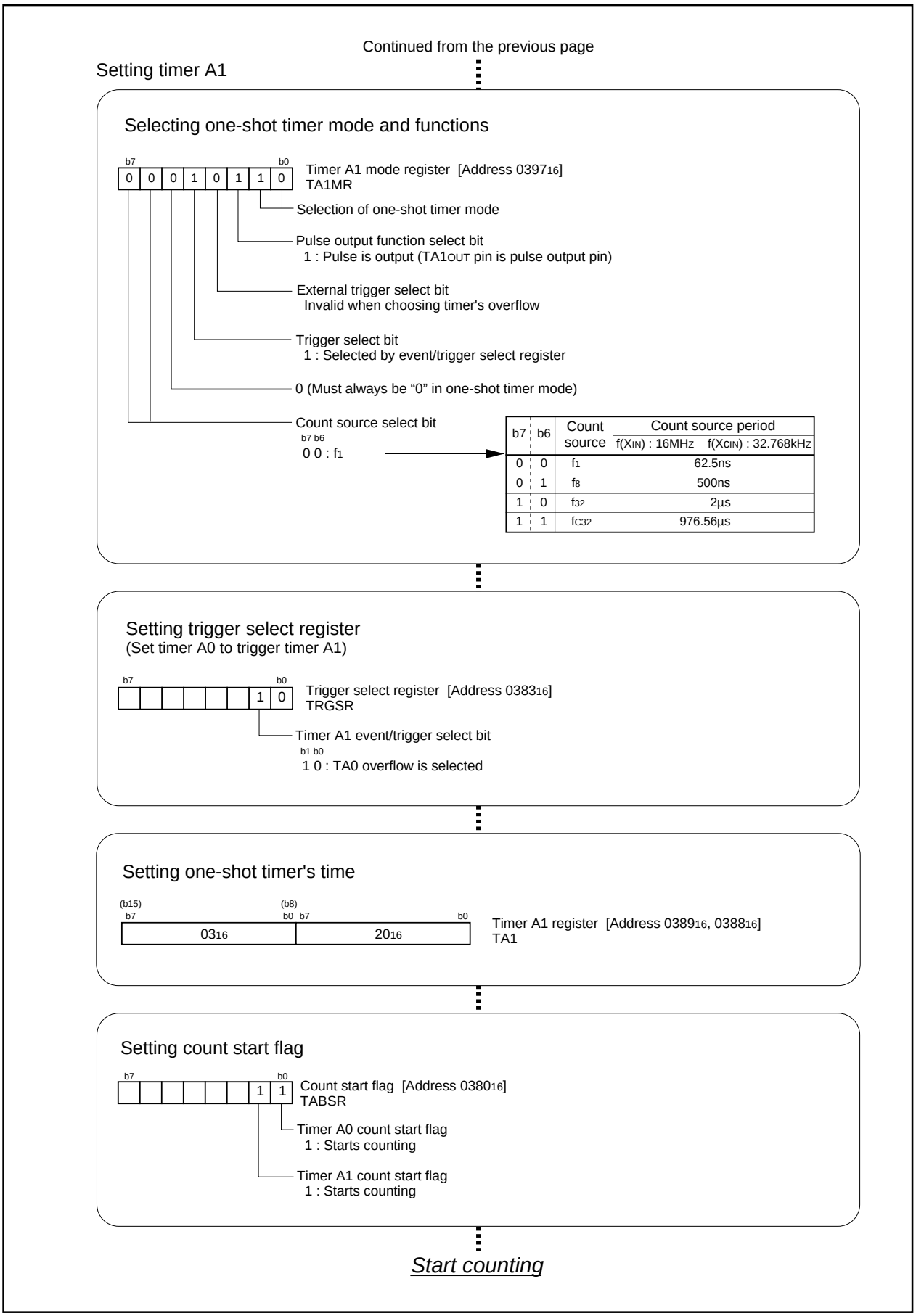


Figure 3.3.4. Set-up procedure of delayed one-shot output (2)

3.4 Buzzer Output

Overview The timer mode is used to make the buzzer ring. Figure 3.4.1 shows the operation timing, and Figure 3.4.2 shows the set-up procedure.

Use the following peripheral function:

• The pulse-outputting function in timer mode of timer A.

Specifications

(1) Sound a 2-kHz buzz beep by use of timer A0.

(2) Effect pull-up in the relevant port by use of a pull-up resistor. When the buzzer is off, set the port high-impedance, and stabilize the potential resulting from pulling up.

(3) Connect a 16-MHz oscillator to XIN.

Operation (1) The microcomputer begins performing a count on timer A0. Timer A0 has disabled interrupts.

(2) The microcomputer begins pulse output by setting the pulse output function select bit to "Pulse output effected". P7$_0$ changes into TA0OUT pin and outputs 2-kHz pulses.

(3) The microcomputer stops outputting pulses by setting the pulse output function select bit to "Pulse output not effected". P7$_0$ goes to an input pin, and the output from the pin becomes high-impedance.

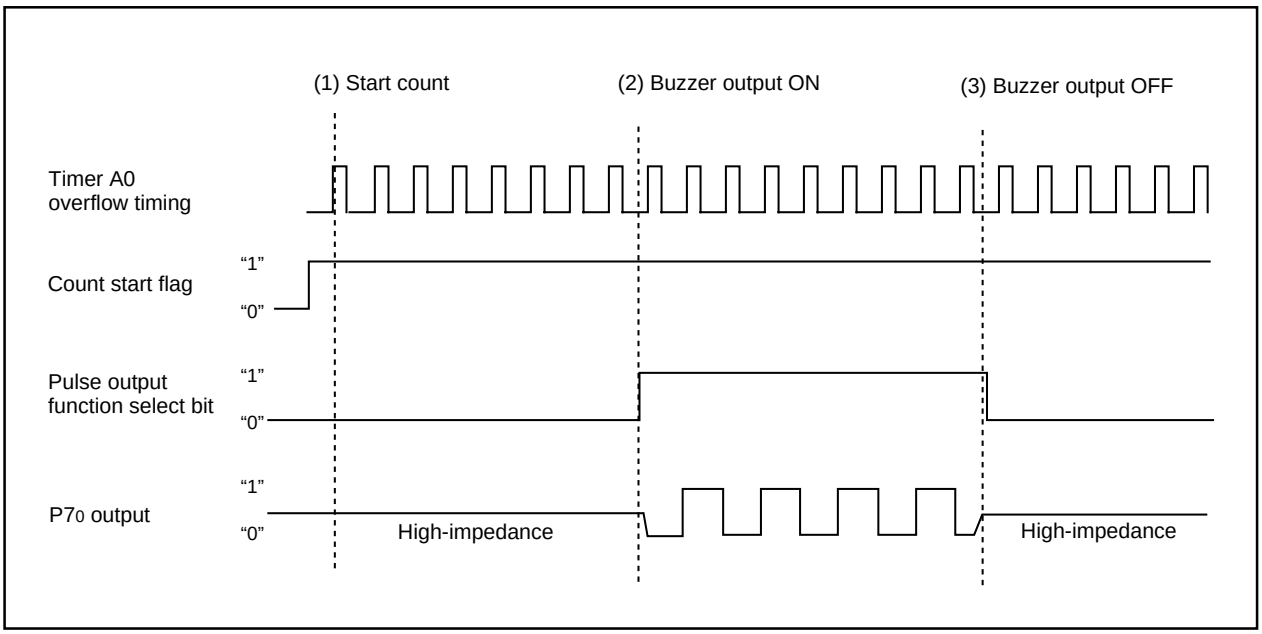


Figure 3.4.1. Operation timing of buzzer output

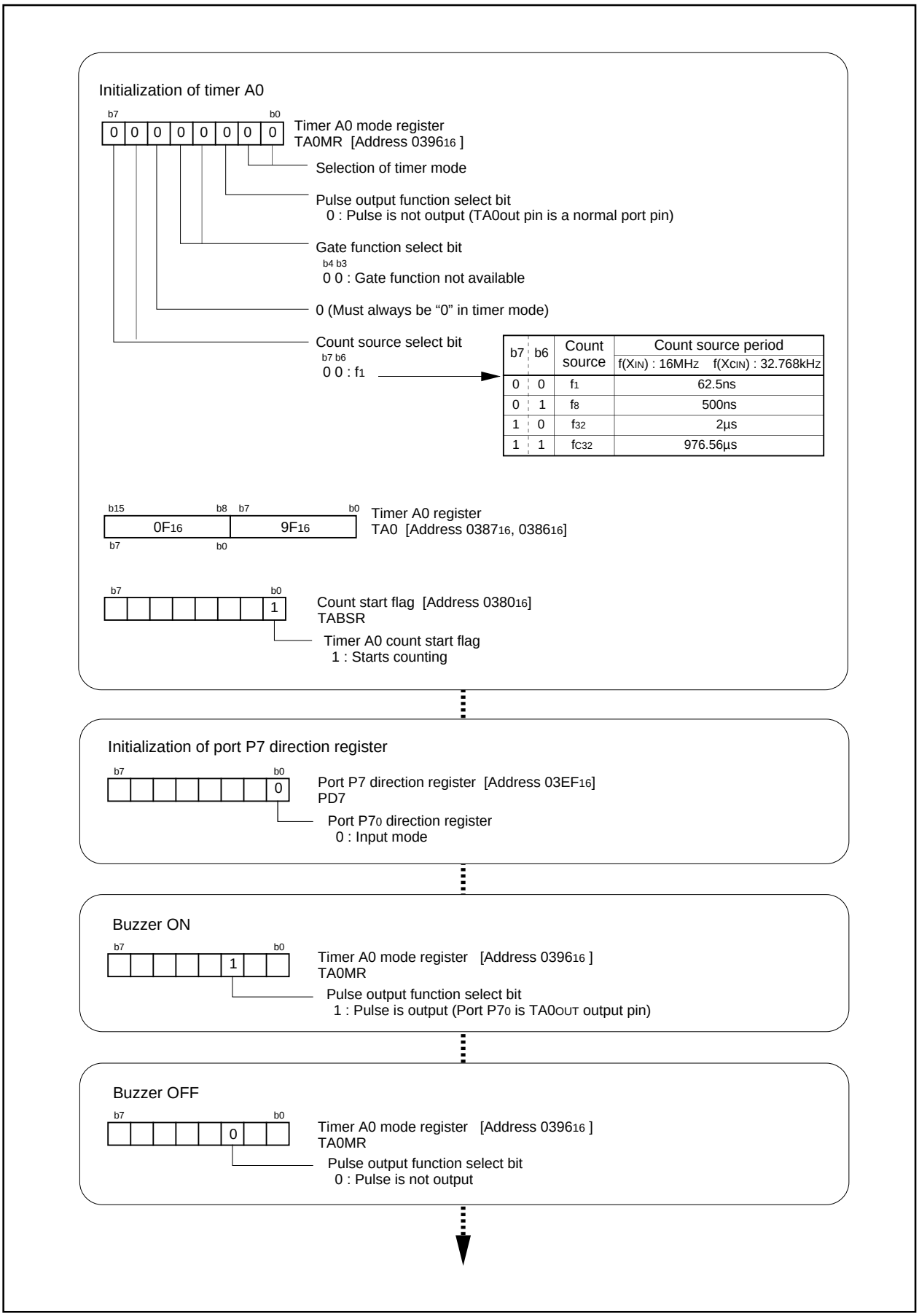


Figure 3.4.2. Set-up procedure of buzzer output

3.5 Solution for External Interrupt Pins Shortage

Overview The following are solution for external interrupt pins shortage. Figure 3.5.1 shows the set-up procedure.
Use the following peripheral function:
• Event counter mode of timer A

Specifications
(1) Inputting a falling edge to the TA0IN pin generates a timer A0 interrupt.

Operation (1) Set timer A0 to event counter mode, set timer to “0”, and set interrupt priority levels in timer A0.
(2) Inputting a falling edge to the TA0IN pin generates a timer A0 interrupt.

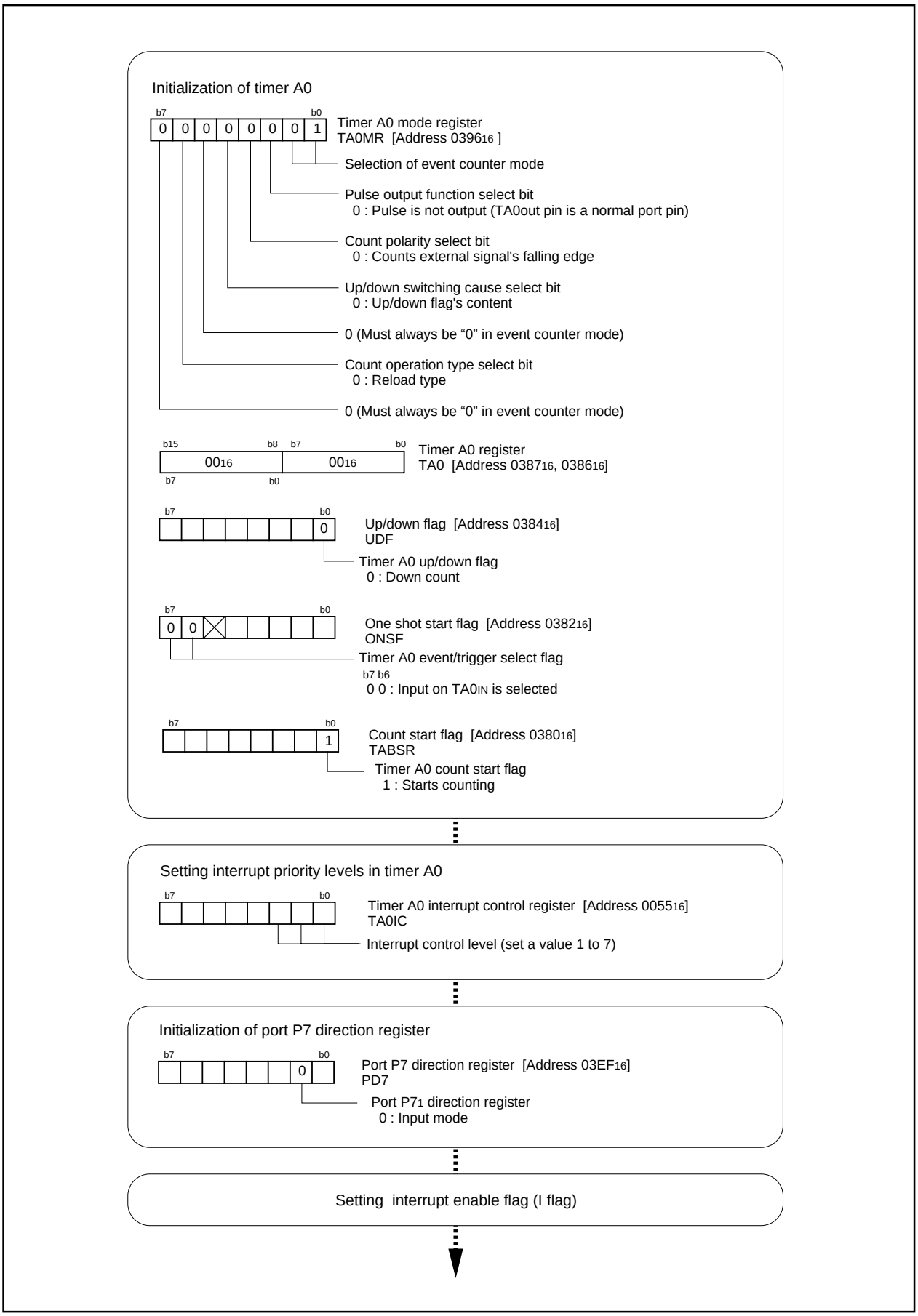


Figure 3.5.1. Set-up procedure of solution for a shortage of external interrupt pins

3.6 Memory to Memory DMA Transfer

Overview The following are steps for changing both source address and destination address to transfer data from memory to another. The DMA transfer utilizes the workings that assign a higher priority to the DMA0 transfer if transfer requests simultaneously occur in two DMA channels. Figure 3.6.1 shows the operation timing, Figure 3.6.2 shows the block diagram, and Figures 3.6.3 and 3.6.4 show the set-up procedure.

Use the following peripheral functions:

- Timer mode of timer A
- Two DMAC channels
- One-byte temporary RAM (address 0800_{16})

Specifications

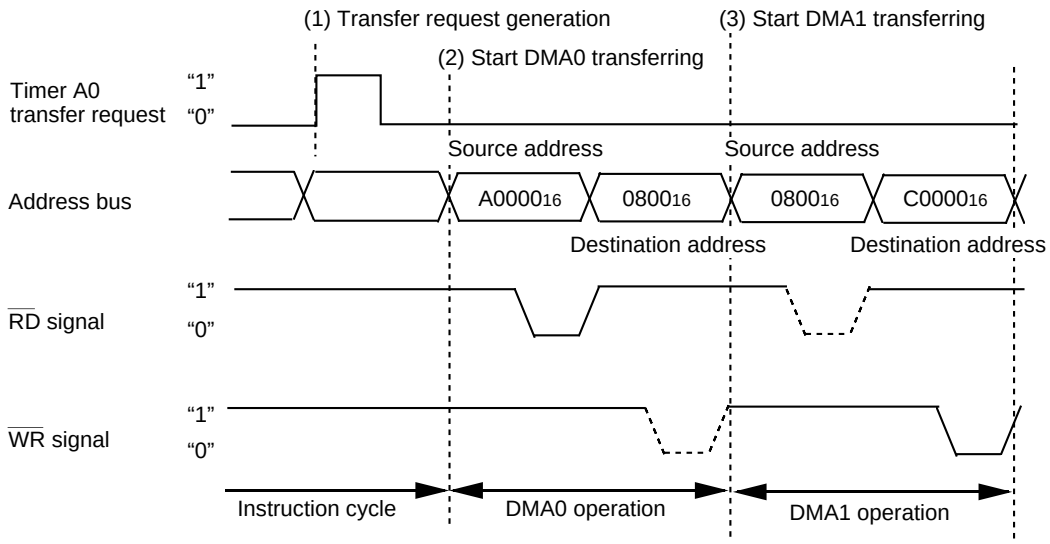
(1) Transfer the content of memory extending over 128 bytes from address $A0000_{16}$ to a 128-byte area starting from address $C0000_{16}$. Transfer the content every time a timer A0 interrupt request occurs.

(2) Use DMA0 for a transfer from the source to built-in memory, and DMA1 for a transfer from built-in memory to the destination.

Operation (1) A timer A interrupt request occurs. Though both a DMA0 transfer request and a DMA1 transfer request occur simultaneously, the former is executed first.

(2) DMA0 receives a transfer request and transfers data from the source to the built-in memory. At this time, the source address is incremented.

(3) Next, DMA1 receives a transfer request and transfers data involved from built-in memory to the destination. At this time, the destination address is incremented.



Note 1: The DMA0 operation and DMA1 operation are not necessarily executed in succession due to the a cycle steal operation.
Note 2: The instruction cycle varies from instruction to instruction.
Note 3: Since the parts of the $\overline{RD}$ and $\overline{WR}$ signals shown in short-dash lines vary in step with writing to the internal RAM, waveforms are not output to the $\overline{RD}$ and $\overline{WR}$ pins.

Figure 3.6.1. Operation timing of memory to memory DMA transfer

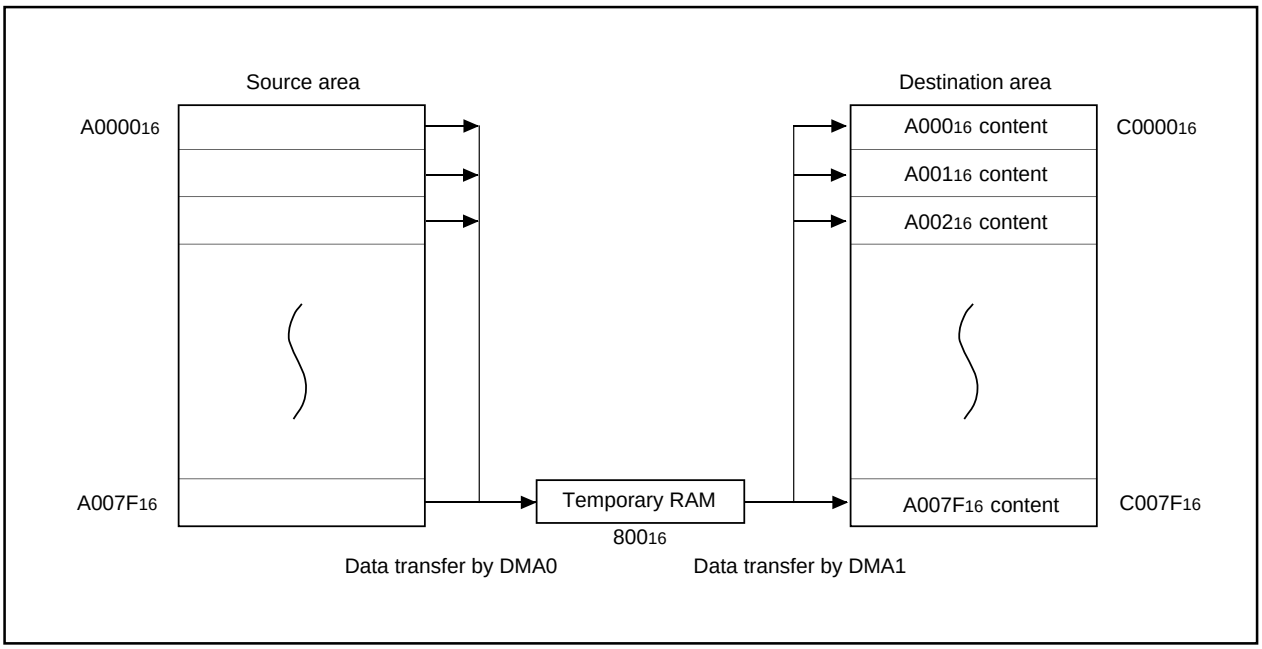


Figure 3.6.2. Block diagram of memory to memory DMA transfer

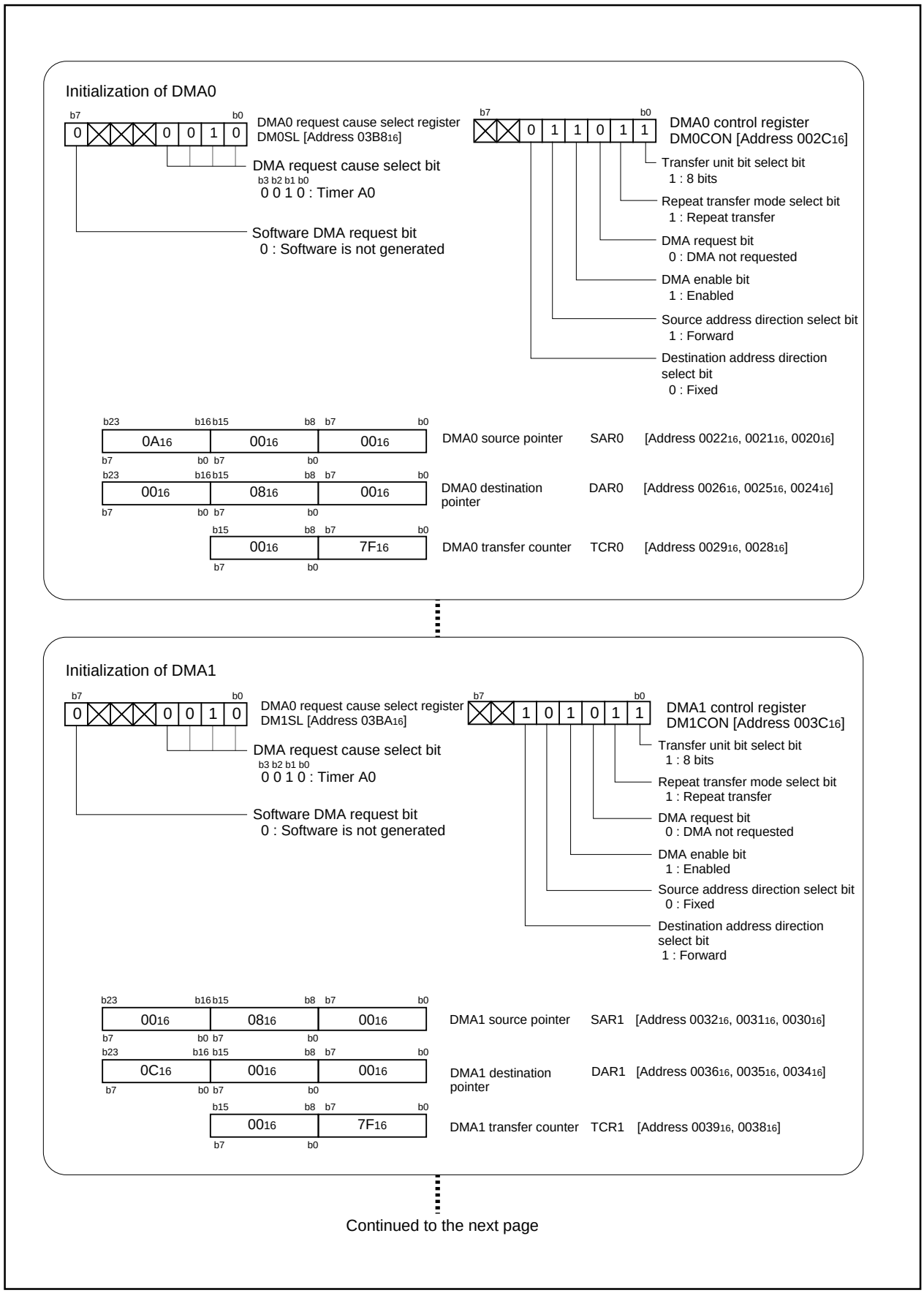


Figure 3.6.3. Set-up procedure of memory to memory DMA transfer (1)

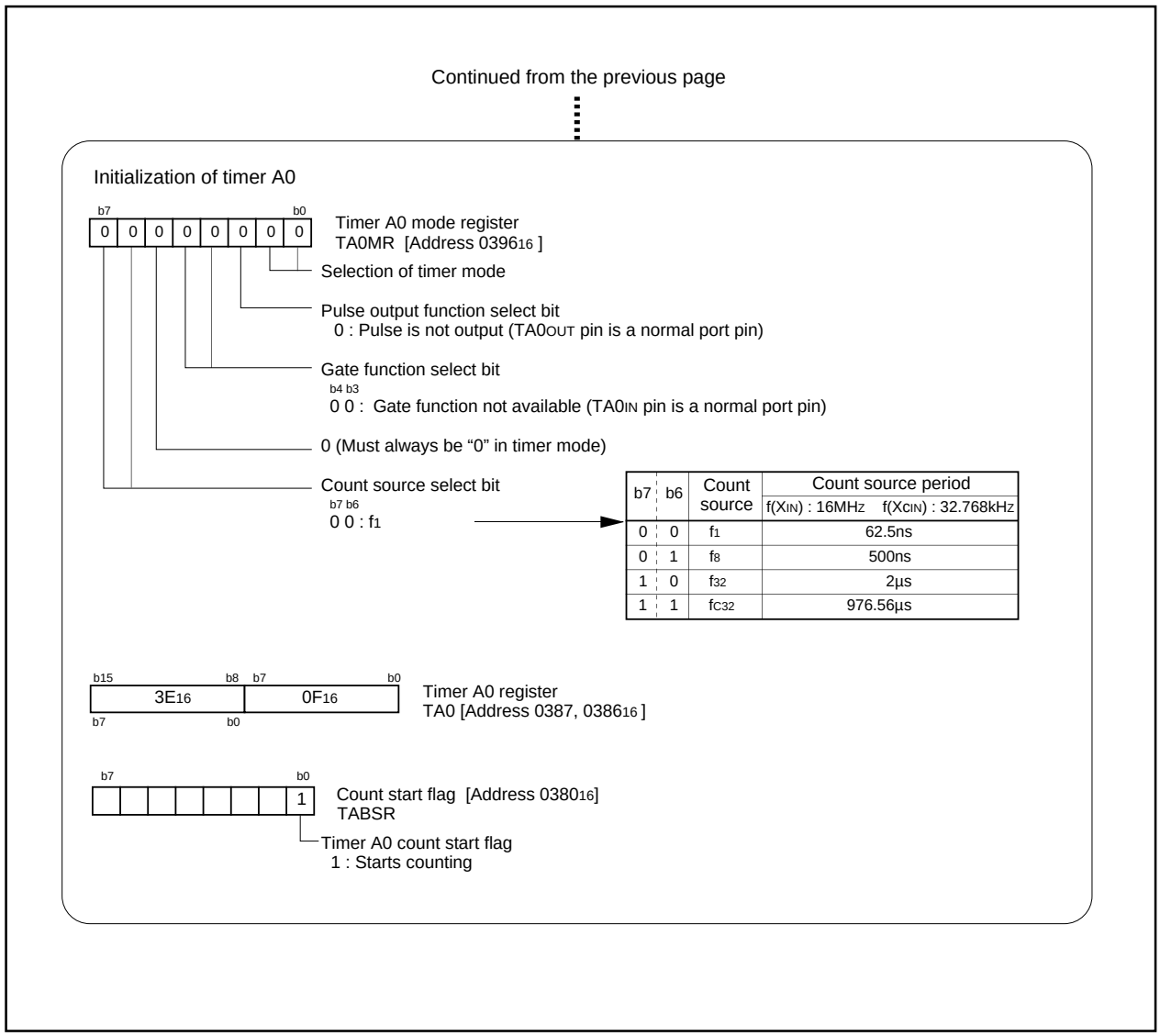


Figure 3.6.4. Set-up procedure of memory to memory DMA transfer (2)

3.7 Controlling Power Using Stop Mode

Overview The following are steps for controlling power using stop mode. Figure 3.7.1 shows the operation timing, Figure 3.7.2 shows an example of circuit, and Figures 3.7.3 and 3.7.4 show the set-up procedure.

Use the following peripheral functions:

- Key-input interrupts
- Stop mode
- Pull-up function

Specifications

(1) Use $P10_0$ through $P10_3$ for the scan output pins of a key matrix. Use the input pins ($\overline{KI_0}$ through $\overline{KI_3}$) of the key-input interrupt function for the key-input reading pins. The pull-up function is also used.

(2) If a key-input interrupt request occurs, clear the stop mode and read a key.

Operation (1) Enable a key-input interrupt and set the pull-up function to pins $\overline{KI_0}$ through $\overline{KI_3}$. Change the output of $P10_0$ through $P10_3$ to "L" and enter stop mode.

(2) If a key is pressed, "L" is input to one of pins $\overline{KI_0}$ through $\overline{KI_3}$ to clear stop mode. A key-input interrupt occurs to execute the key-input interrupt handling routine.

(3) Sequentially set $P10_0$ through $P10_3$ to "L" to determine which key was pressed.

(4) When the process to determine the key pressed is completed, change the output from $P10_0$ through $P10_3$ to "L" again and enter stop mode.

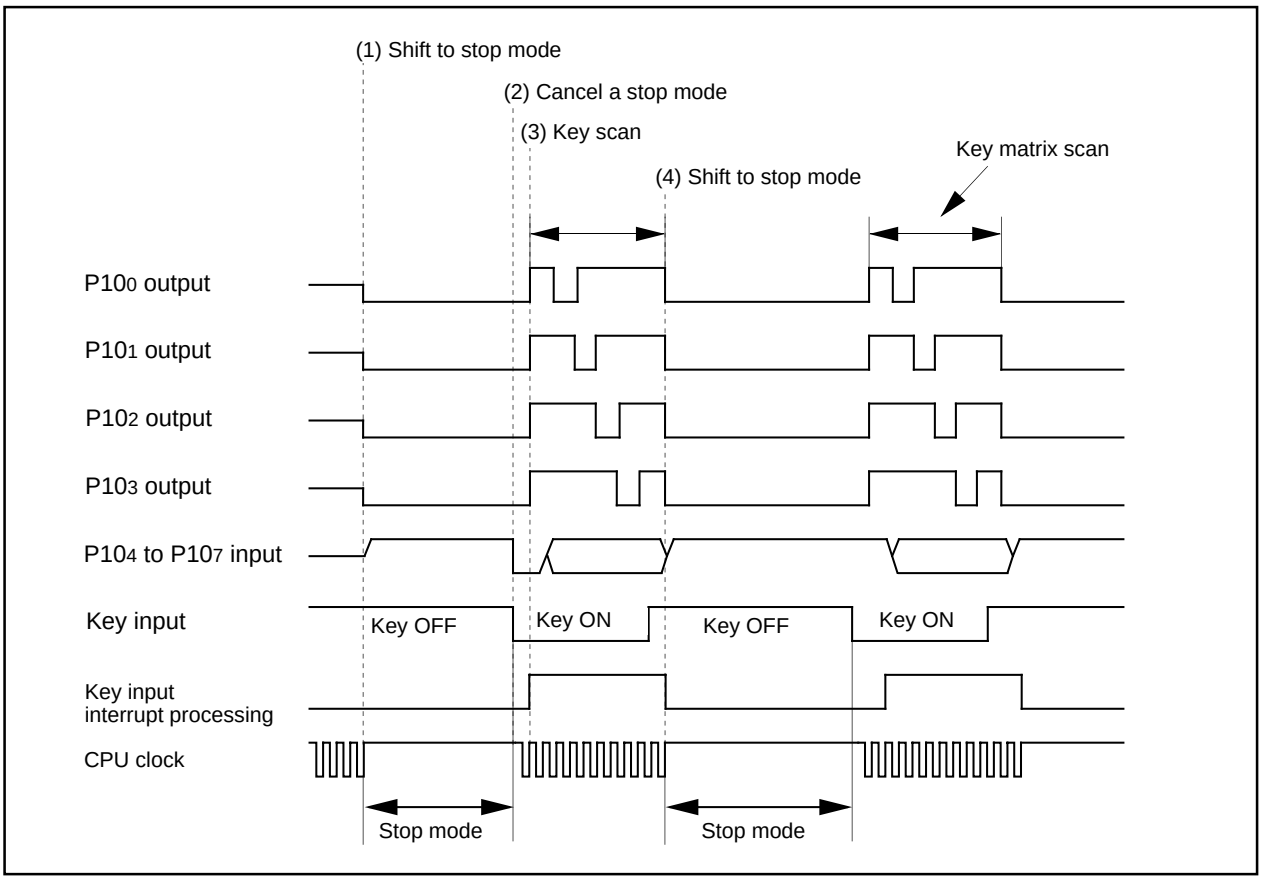


Figure 3.7.1. Operation timing of controlling power using stop mode

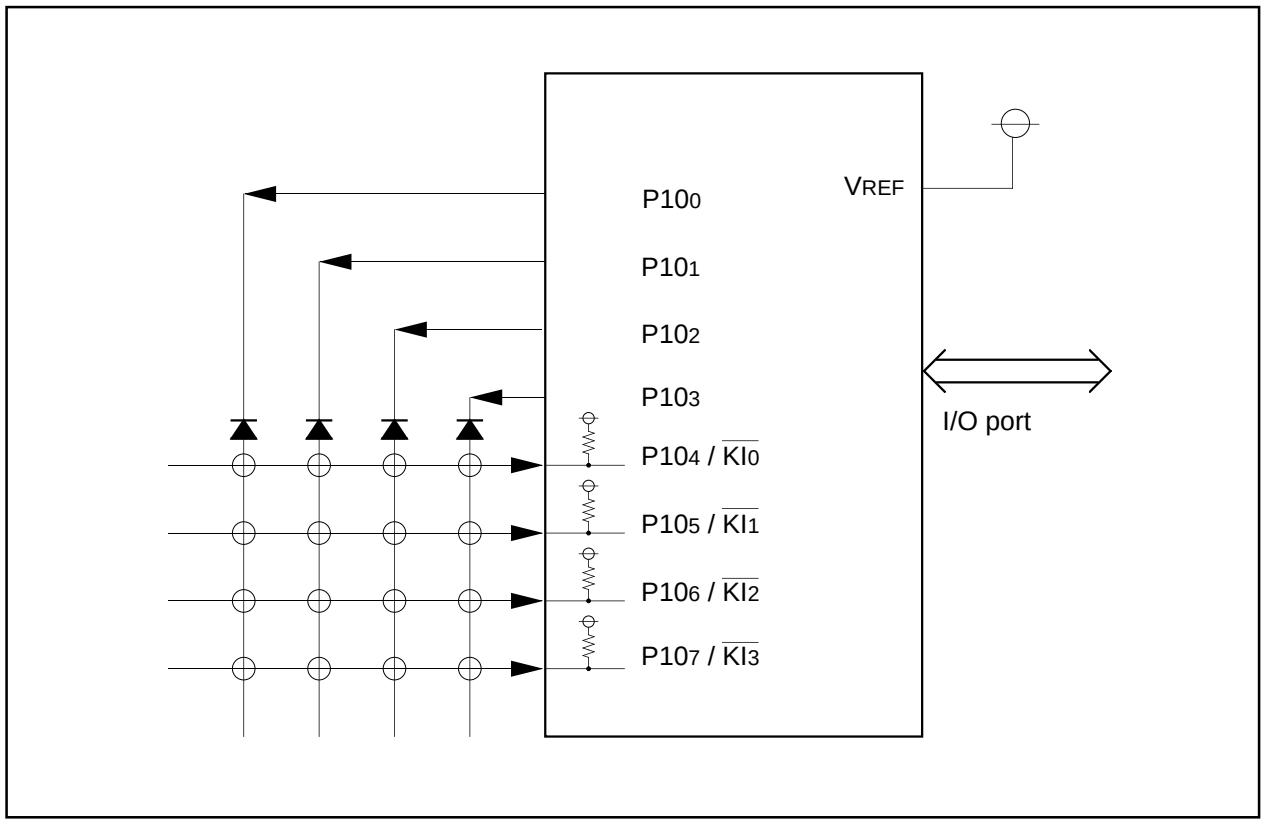


Figure 3.7.2. Example of circuit of controling power using stop mode

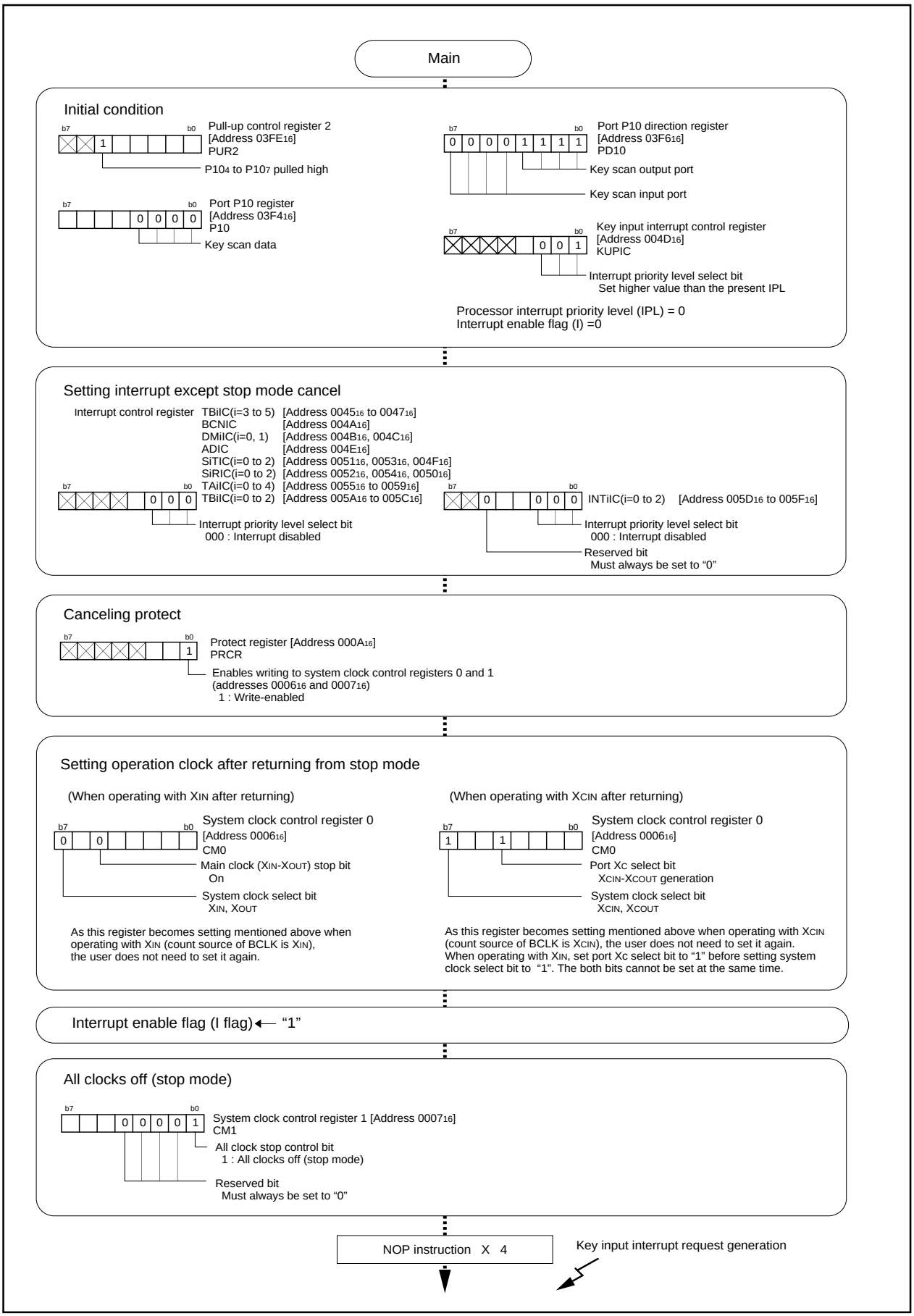


Figure 3.7.3. Set-up procedure of controlling power using stop mode (1)

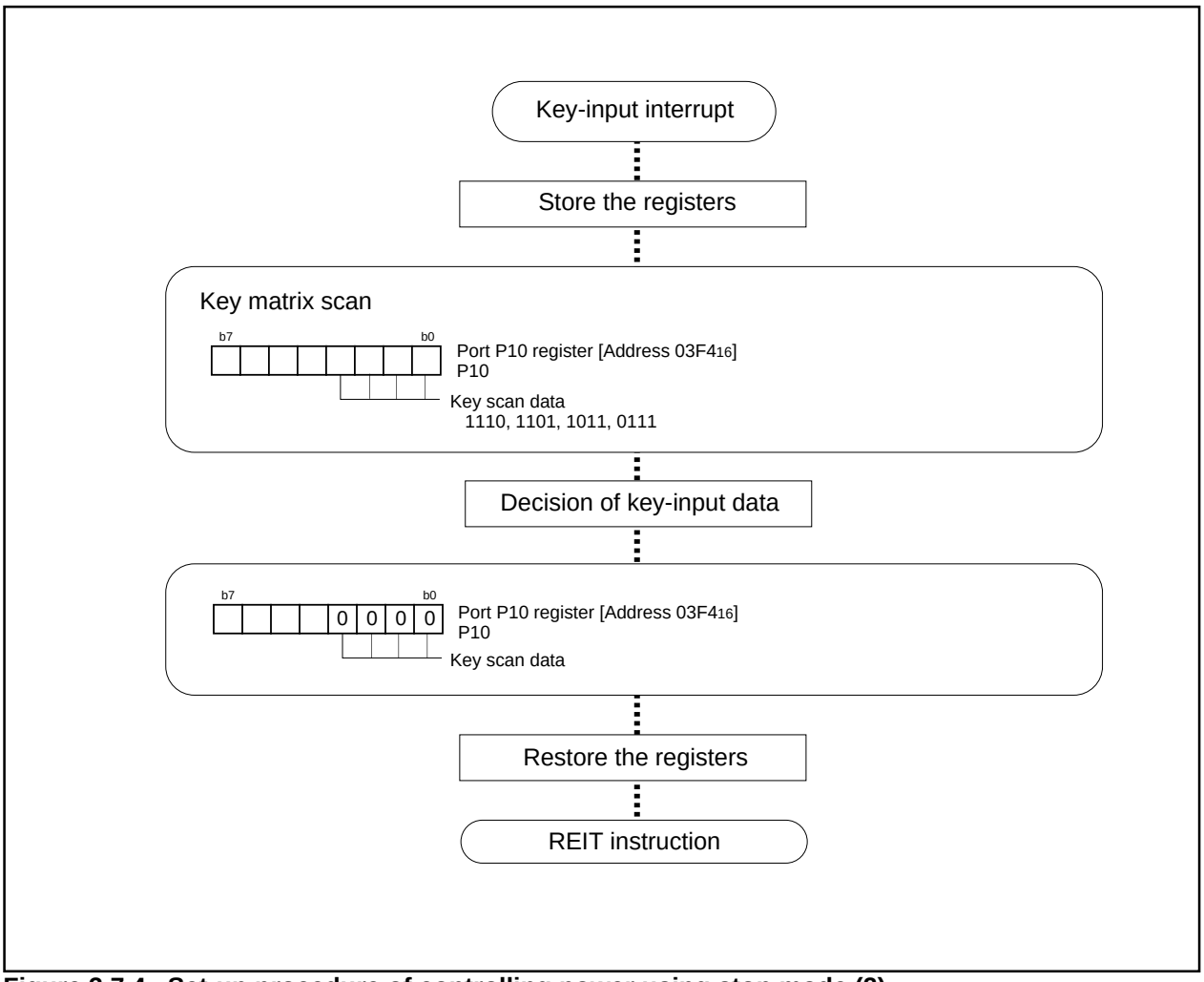


Figure 3.7.4. Set-up procedure of controlling power using stop mode (2)

3.8 Controlling Power Using Wait Mode

Overview The following are steps for controling power using wait mode. Figure 3.8.1 shows the operation timing, and Figures 3.8.2 to 3.8.4 show the set-up procedure.

Use the following peripheral functions:

• Timer mode of timer B

• Wait mode

A flag named “F-WIT” is used in the set-up procedure. The purpose of this flag is to decide whether or not to clear wait mode. If F_WIT = “1” in the main program, the wait mode is entered; if F_WIT = “0”, the wait mode is cleared.

Specifications

(1) Connect a 32.768-kHz oscillator to XCIN to serve as the timer count source. As interrupts occur every one second, which is a count the timer reaches, the controller returns from wait mode and count the clock using a program.

(2) Clear wait mode if a $\overline{\text{INT0}}$ interrupt request occurs.

Operation (1) Switch the system clock from XIN to XCIN to get low-speed mode.

(2) Stop XIN and enter wait mode. In this instance, enable the timer B2 interrupt and the $\overline{\text{INT0}}$ interrupt.

(3) When a timer B2 interrupt request occurs (at 1-second intervals), start supplying the BCLK from XCIN.

At this time, count the clock within the routine that handles the timer B2 interrupts and enter wait mode again.

(4) If a $\overline{\text{INT0}}$ interrupt occurs, start supplying the BCLK from XCIN. Start the XIN oscillation within the $\overline{\text{INT0}}$ interrupt, and switch the system clock to XIN.

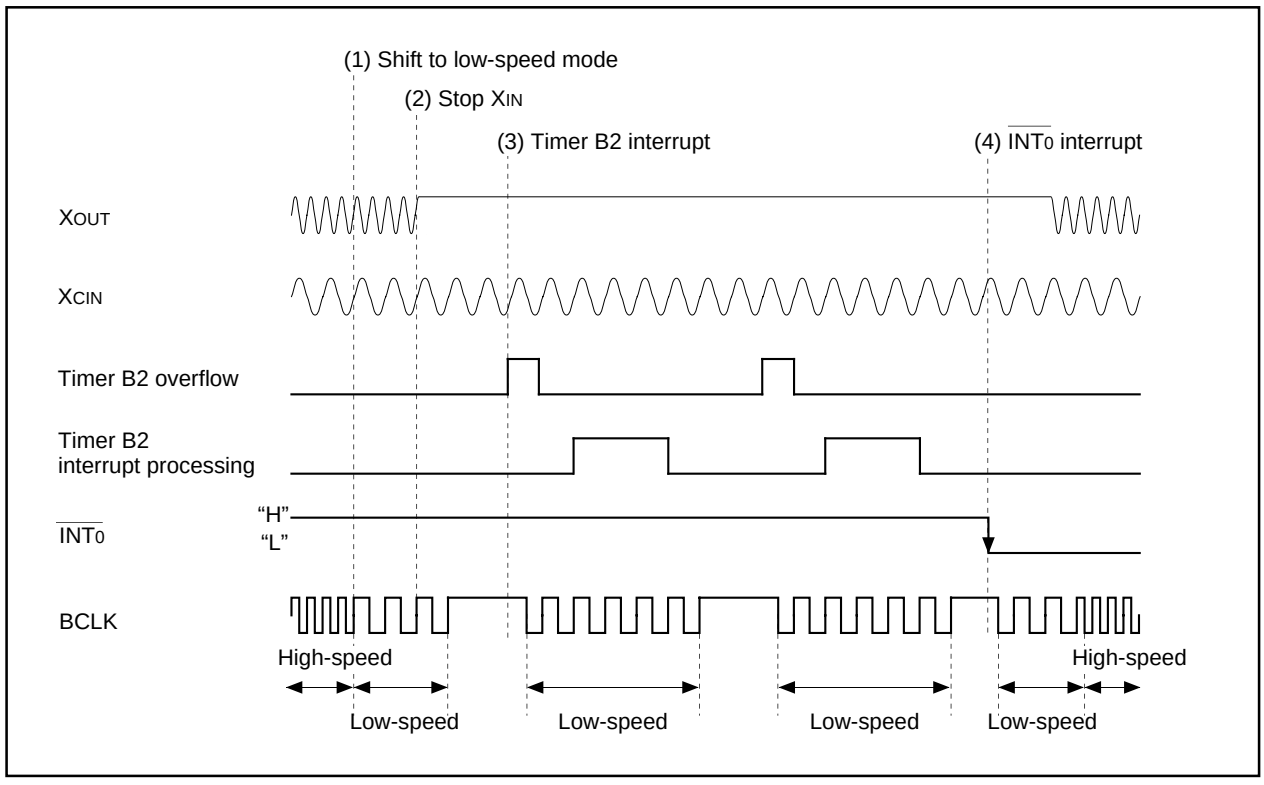


Figure 3.8.1. Operation timing of controling power using wait mode

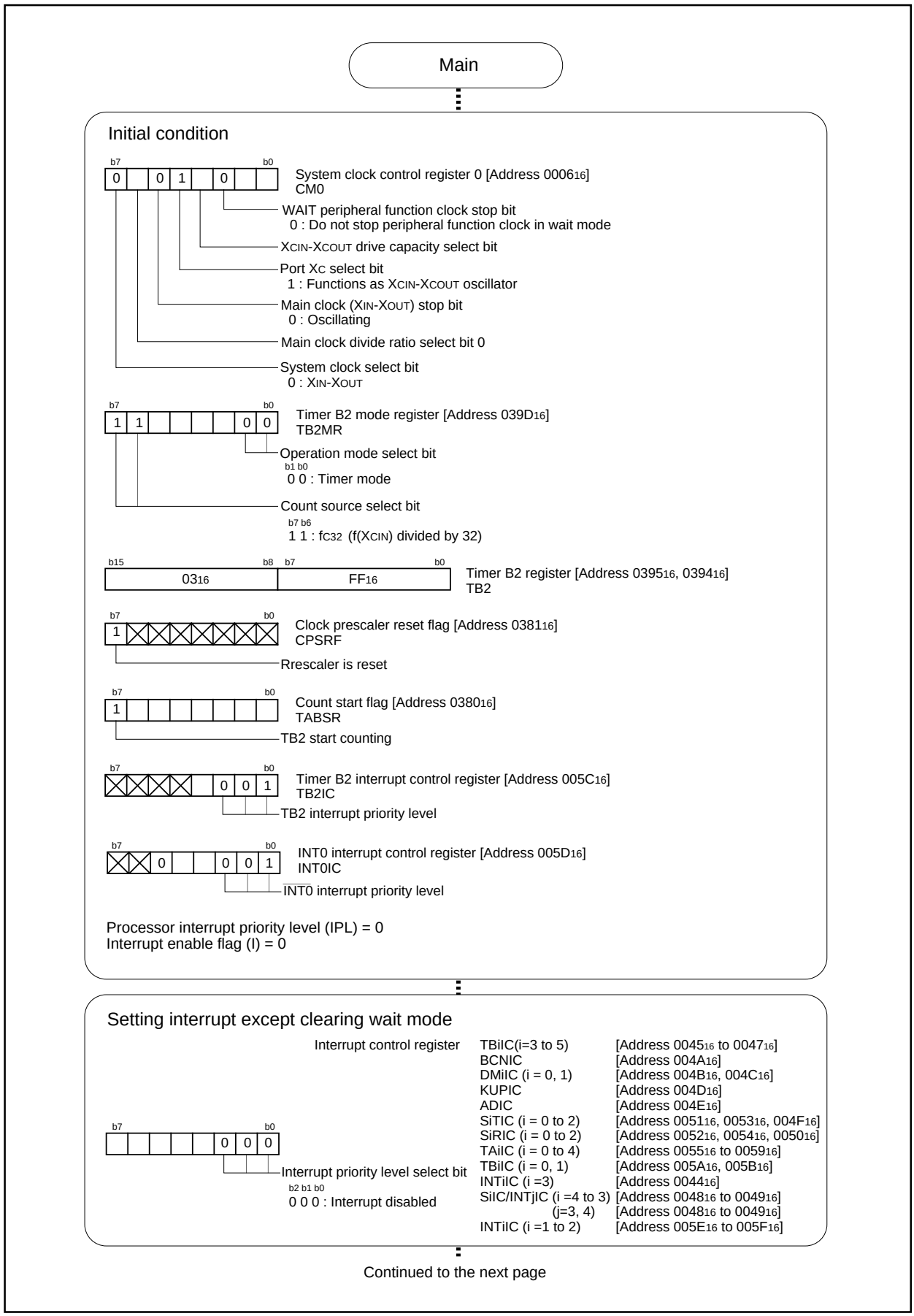


Figure 3.8.2. Set-up procedure of controlling power using wait mode (1)

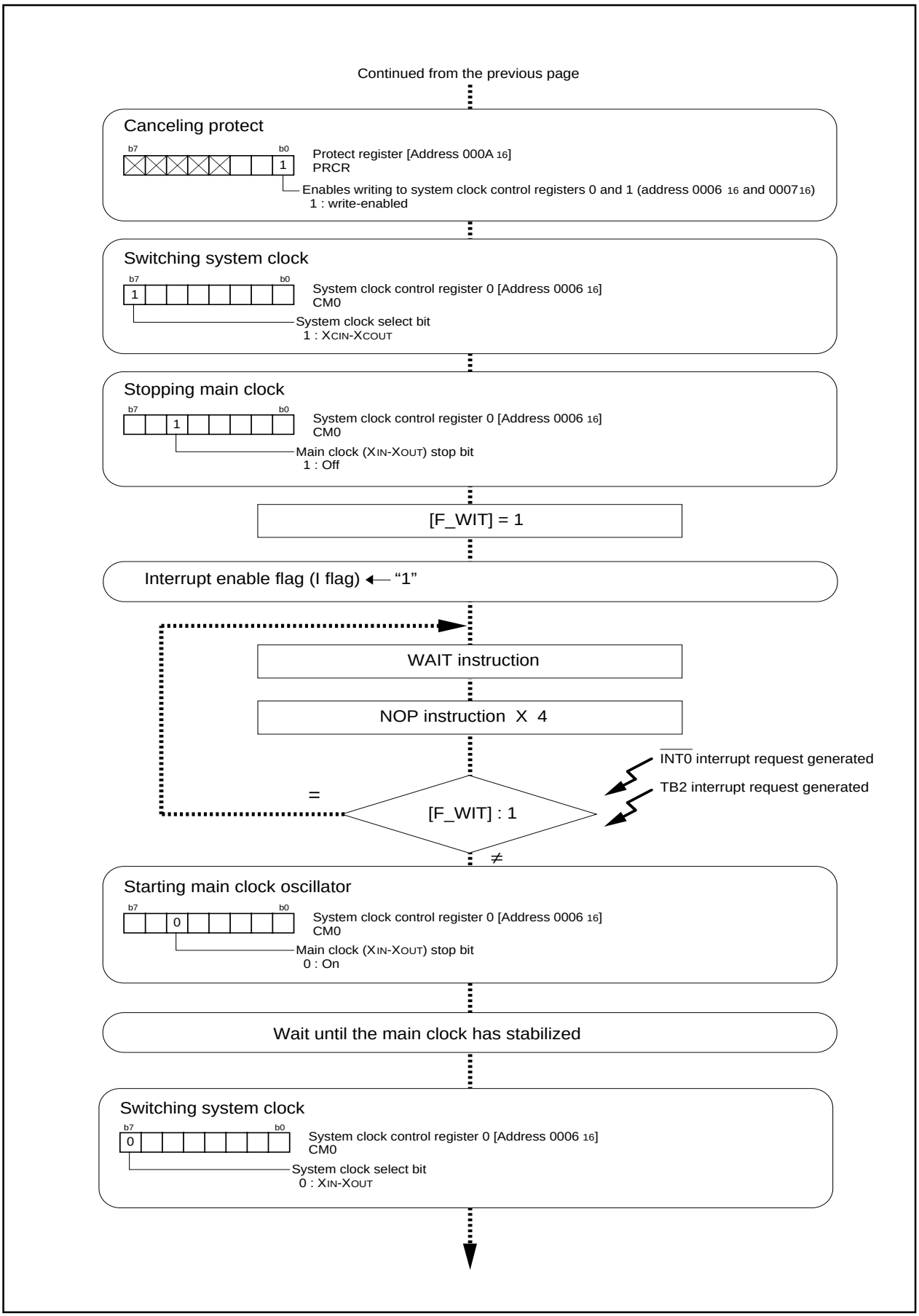


Figure 3.8.3. Set-up procedure of controlling power using wait mode (2)

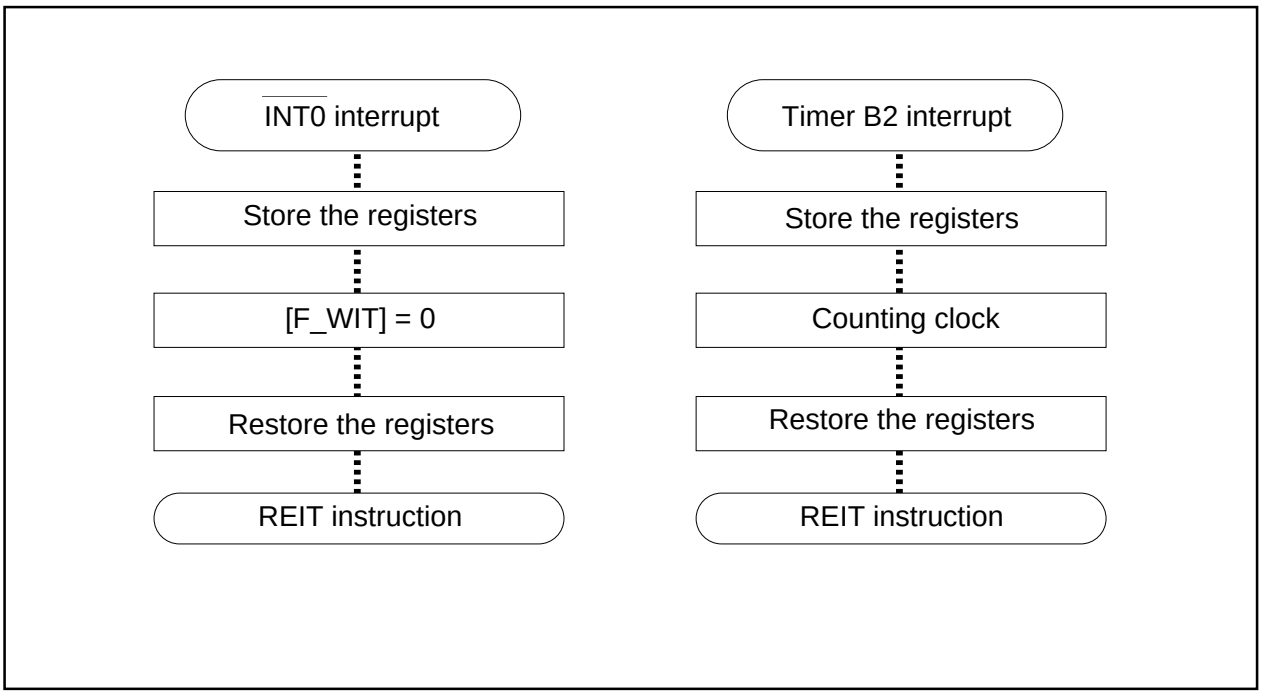


Figure 3.8.4. Set-up procedure of controlling power using wait mode (3)

[MEMO]

Chapter 4

External Buses

4.1 Overview of External Buses

Memory and I/O external expansion can be connected to microcomputer easily by using external buses. When memory expansion mode or microprocessor mode is selected for processor mode, some of the pins function as the address bus, the data bus, and as control signals and this makes the external buses be able to operate.

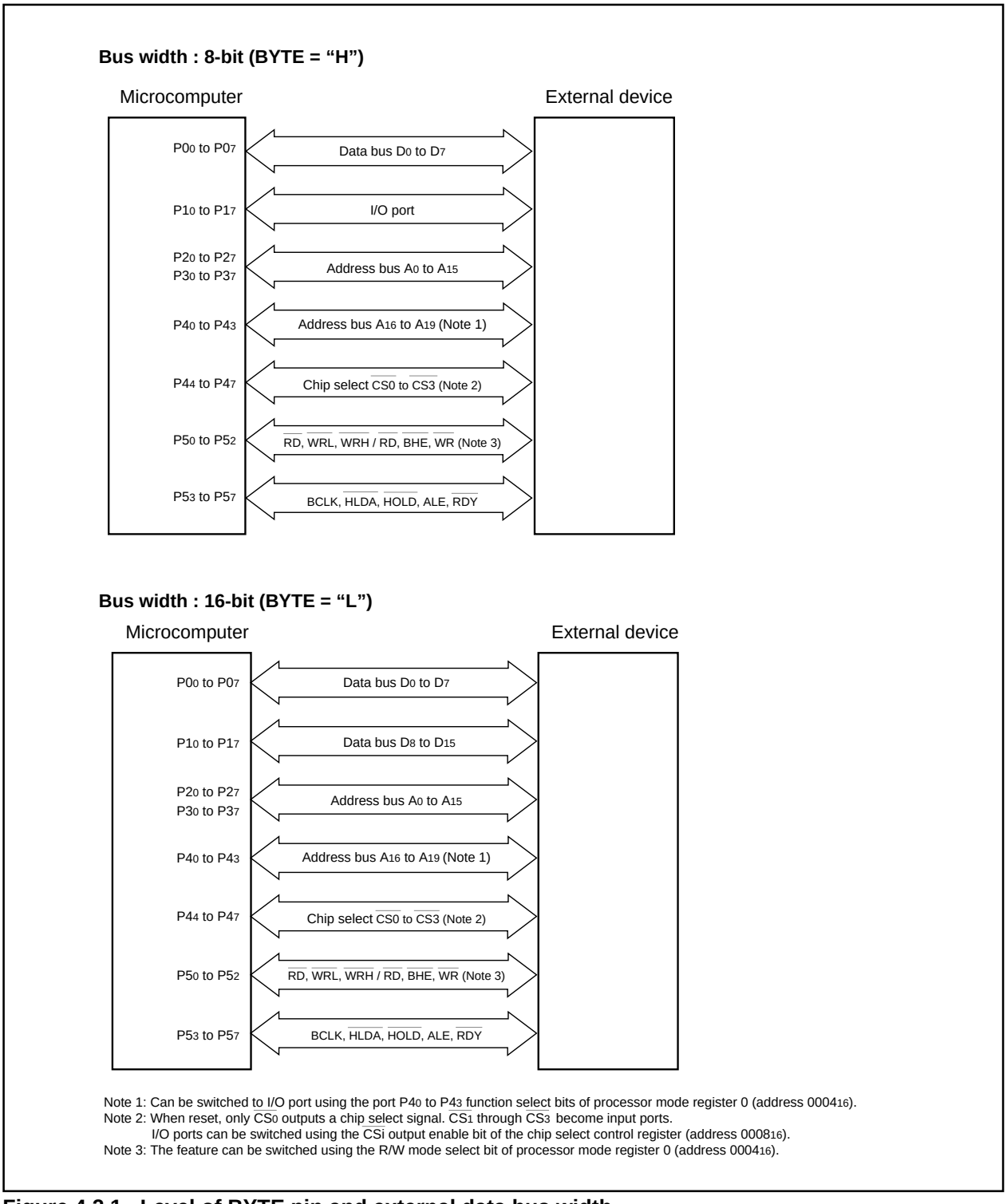
When accessing an external area, 8-bit data bus width or 16-bit data bus width can be selected, based on the BYTE pin level. 16-bit width is used to access an internal area, regardless of the level of the BYTE pin. Fix the BYTE pin either to “H” or “L” level. 8-bit and 16-bit data bus widths cannot be used together in an external area.

4.2 Data Access

4.2.1 Data Bus Width

If the voltage level input to the BYTE pin is “H”, the external data bus width becomes 8 bits, and P1$_0$ (/D$_8$) through P1$_7$ (/D$_{15}$) can be used as I/O ports (Figure 4.2.1).

If the voltage level input to the BYTE pin is “L”, the external data bus width becomes 16 bits, and P1$_0$ (/D$_8$) through P1$_7$ (/D$_{15}$) operate as a data bus (D$_8$ through D$_{15}$) (Figure 4.2.1).

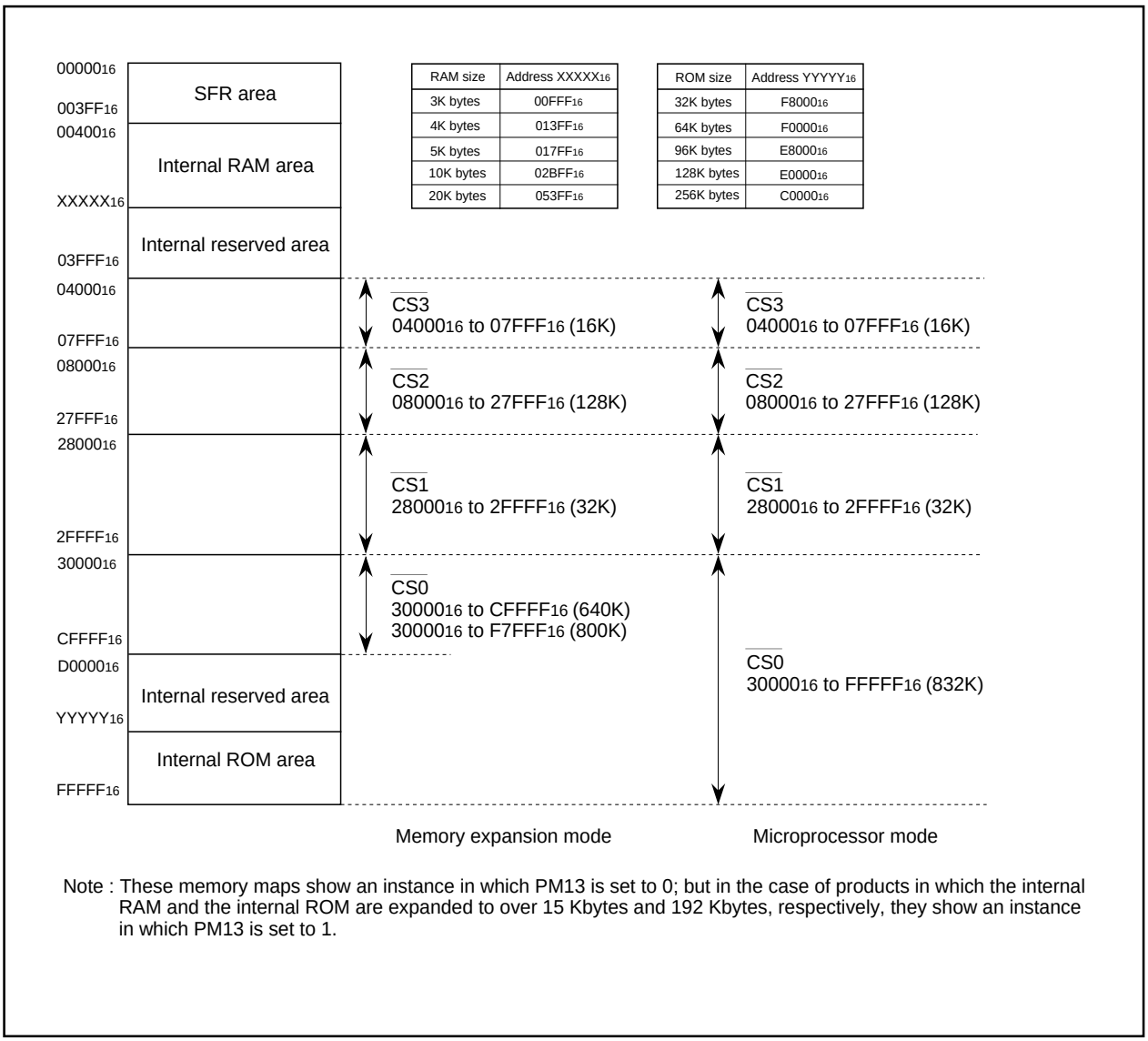


Note 1: Can be switched to I/O port using the port P4$_0$ to P4$_3$ function select bits of processor mode register 0 (address 0004$_{16}$).
Note 2: When reset, only CS$_0$ outputs a chip select signal. CS$_1$ through CS$_3$ become input ports.
I/O ports can be switched using the CSi output enable bit of the chip select control register (address 0008$_{16}$).
Note 3: The feature can be switched using the R/W mode select bit of processor mode register 0 (address 0004$_{16}$).

Figure 4.2.1. Level of BYTE pin and external data bus width

4.2.2 Chip Selects and Address Bus

Chip selects (P44/$\overline{CS0}$ through P47/$\overline{CS3}$) are output in areas resulting from dividing a 1-M byte memory space into four. To use the chip select, the chip select output must be enabled by setting the chip select control register. Figure 4.2.2 shows addresses in which chip selects become active (“L”). Since the extent of the internal area and the external area in memory expansion mode is different from those in microprocessor mode, there is a difference between areas for which $\overline{CS0}$ is output. When an internal ROM/RAM area is being accessed, no chip select is output, and the address bus does not change (the address of the external area that was accessed previously is held).



Note : These memory maps show an instance in which PM13 is set to 0; but in the case of products in which the internal RAM and the internal ROM are expanded to over 15 Kbytes and 192 Kbytes, respectively, they show an instance in which PM13 is set to 1.

Figure 4.2.2. Addresses in which chip selects turn active (“L”)

Chip select control register

b7	b6	b5	b4	b3	b2	b1	b0

Symbol: CSR
Address: 0008$_{16}$
When reset: 01$_{16}$

Bit symbol	Bit name	Function	R	W
CS0	$\overline{CS0}$ output enable bit	0 : Chip select output disabled (Normal port pin) 1 : Chip select output enabled	O	O
CS1	$\overline{CS1}$ output enable bit		O	O
CS2	$\overline{CS2}$ output enable bit		O	O
CS3	$\overline{CS3}$ output enable bit		O	O
CS0W	$\overline{CS0}$ wait bit	0 : Wait state inserted 1 : No wait state	O	O
CS1W	$\overline{CS1}$ wait bit		O	O
CS2W	$\overline{CS2}$ wait bit		O	O
CS3W	$\overline{CS3}$ wait bit		O	O

Figure 4.2.3. Chip select control register

4.2.3 Bus Types

The M16C/62A Group has two types of buses: a separate bus where separate pins are used for address output and data input/output and a multiplexed bus where pins are time- multiplexed and switched between address output and data input/output to save the number of pins used.
A separate bus is used to access devices such as ROM and RAM which have separate buses. The areas accessed via separate buses can be allocated for programs and data.
A multiplexed bus is used to access devices such as ASSPs which have multiplexed buses. The areas accessed via a multiplexed bus can only be allocated for data. Programs cannot be located in these areas.

The area accessed via a multiplex bus can be selected from three types of area $\overline{CS2}$ area, $\overline{CS1}$ area, and entire space by setting the multiplexed bus select bits (bits 4 and 5) of the processor mode register 0 (address 0004$_{16}$). However, the entire space cannot be selected when operating in the microprocessor mode.
Areas not accessed via multiplexed bus are accessed through separate buses.

When accessing an area set for access via a multiplexed bus the BYTE pin is “H” level, the data bus D_0 to D_7 is multiplexed with address bus A_0 to A_7.
If the BYTE pin is “L” level, the data bus D_0 to D_7 is multiplexed with address bus A_1 to A_8. In either case, the bus is switched between data and address separated only in time.
In the latter case, however, the addresses of connected devices are mapped into even addresses (every other addresses) of the M16C/62A. Therefore, be sure to access the M16C/62A's even addresses in length of bytes when accessing a connected device.

4.2.4 R/W Modes

The read/write signal that is output when accessing an external area can be selected between the $\overline{RD}$/$\overline{BHE}$/$\overline{WR}$ and the $\overline{RD}$/$\overline{WRH}$/$\overline{WRL}$ modes by setting the R/W mode select bit (bit 2) of the processor mode register 0 (address 0004$_{16}$). Use the $\overline{RD}$/$\overline{BHE}$/$\overline{WR}$ mode to access a 16-bit wide RAM and the $\overline{RD}$/$\overline{WRH}$/$\overline{WRL}$ mode to access an 8-bit wide RAM.
When the M16C/62A is reset, the $\overline{RD}$/$\overline{BHE}$/$\overline{WR}$ mode is selected by default. To switch over the R/W mode, change the $\overline{RD}$/$\overline{BHE}$/$\overline{WR}$ to the $\overline{RD}$/$\overline{WRH}$/$\overline{WRL}$ mode before accessing an external RAM.
Refer to the connection examples of $\overline{RD}$/$\overline{BHE}$/$\overline{WR}$ and $\overline{RD}$/$\overline{WRH}$/$\overline{WRL}$ shown in Section 4.3, "Connection Examples."

4.3 Connection Examples

4.3.1 16-bit Memory to 16-bit Width Data Bus Connection Example

Figure 4.3.1 shows an example of connecting M5M51016BTP (SRAM). In this diagram, when reset the microcomputer starts operating in single-chip mode. Change this mode to memory expansion mode in a program.

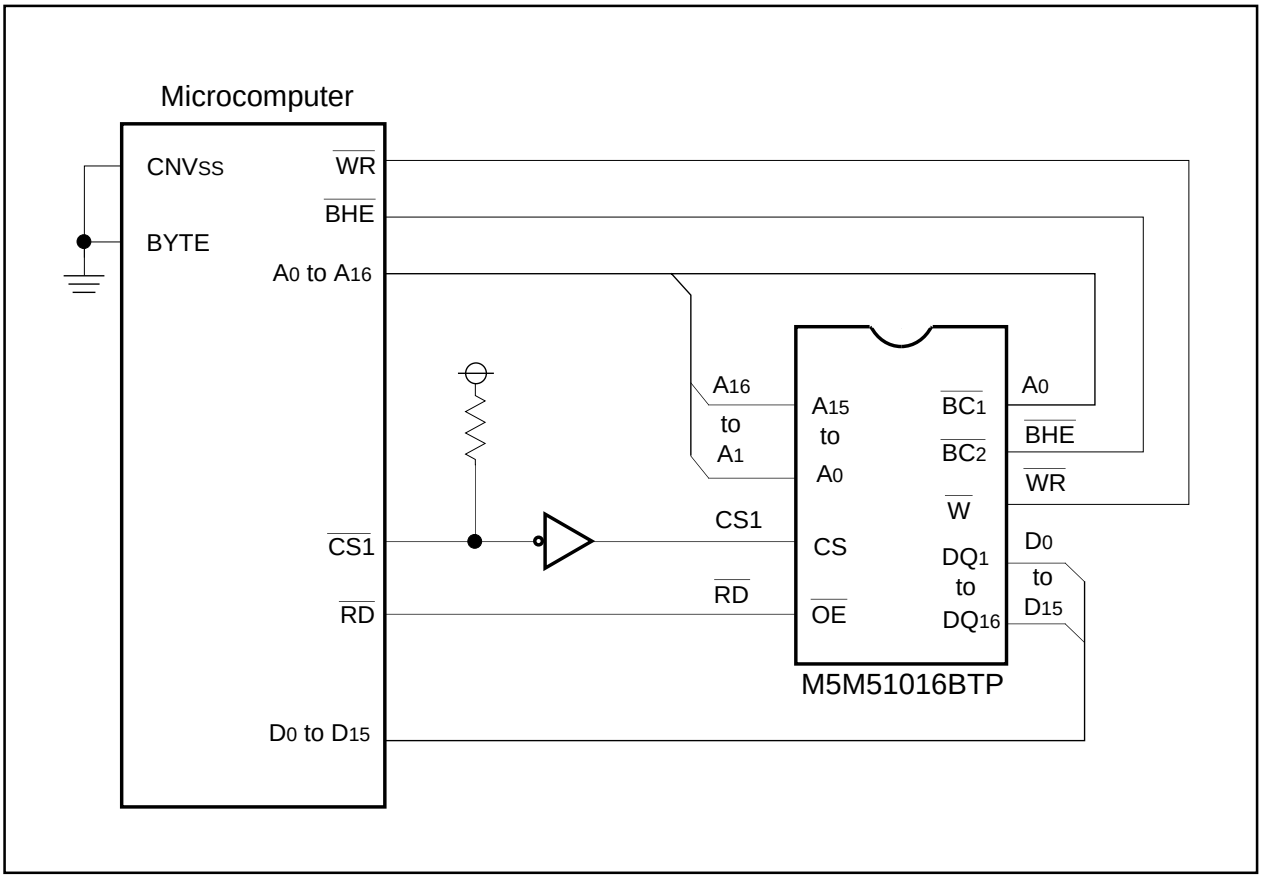


Figure 4.3.1. Example of connecting M5M51016BTP

4.3.2 8-bit Memory to 16-bit Width Data Bus Connection Example

Figure 4.3.2 shows an example of connecting two M5M5278's (SRAM) to a 16-bit data bus.In this diagram, when reset the microcomputer starts operating in single-chip mode. Change this mode to memory expansion mode in a program.

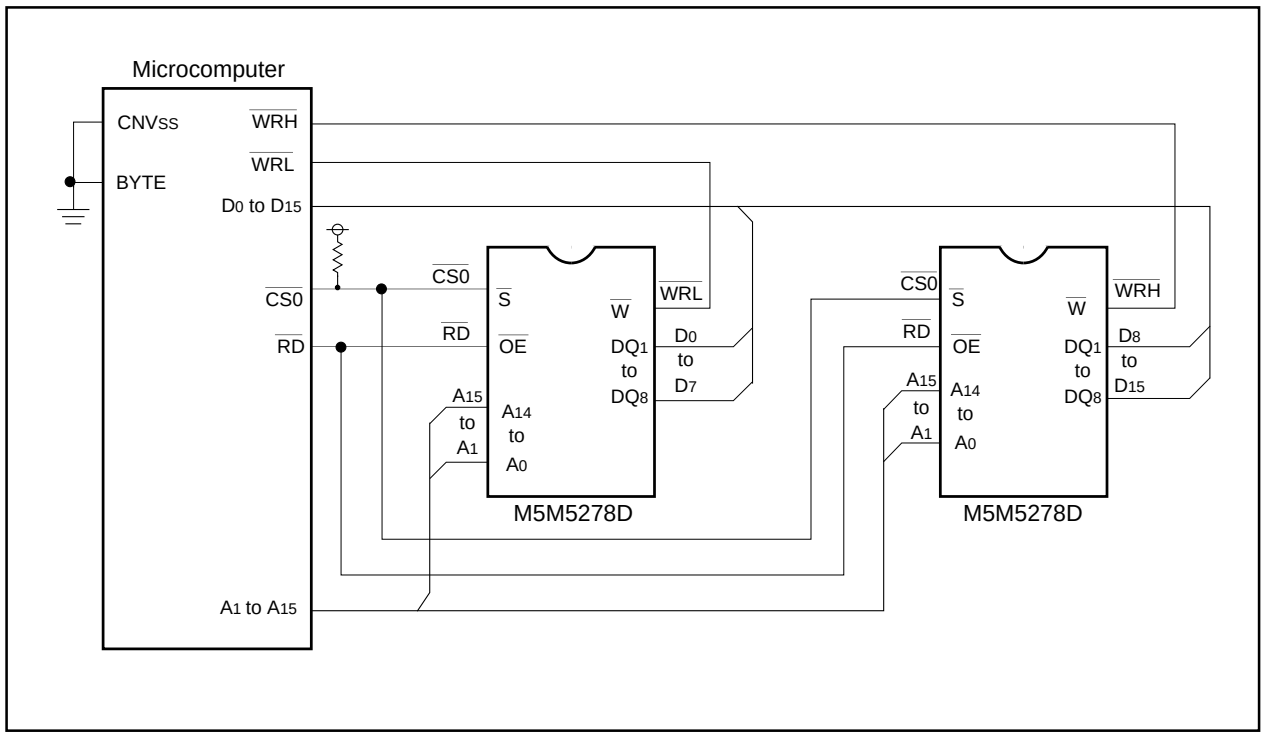


Figure 4.3.2. Example of connecting two M5M5278's to a 16-bit data bus

Figure 4.3.3 shows how to connect two Am29LV008B (flash memory). In 16-bit bus mode, the $\overline{BHE}$/$\overline{WRH}$ pin functions as $\overline{BHE}$. When connecting 8-bit flash memory chips to the 16-bit bus, make sure the microcomputer's $\overline{WRL}$ pin is connected to the $\overline{WR}$ pins on both flash memory chips, and that data is written to the flash memory in units of 16 bits beginning with an even address.

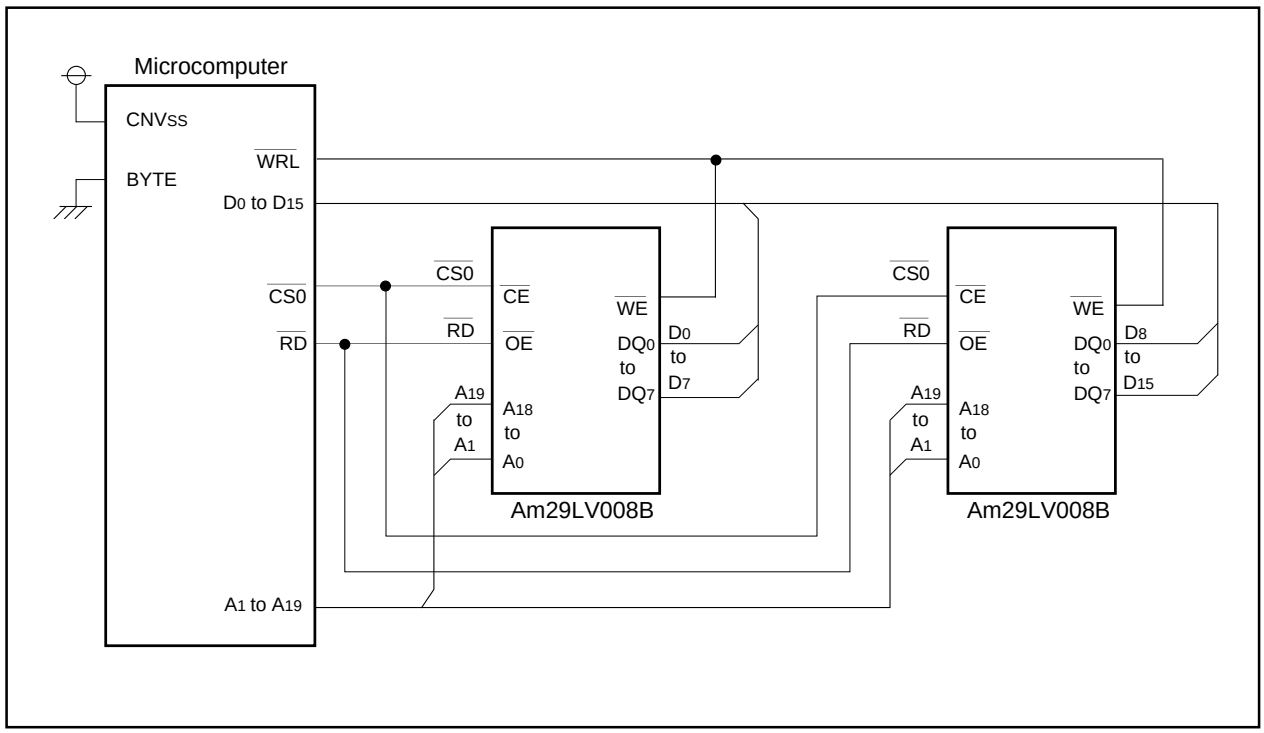


Figure 4.3.3. Example of connecting two Am29LV008B's to a 16-bit data bus

4.3.3 8-bit Memory to 8-bit Width Data Bus Connection Example

Figure 4.3.4 shows an example of connecting two M5M5278's (SRAM) to an 8-bit data bus. In this diagram, when reset the microcomputer starts operating in single-chip mode. Change this mode to memory expansion mode in a program.

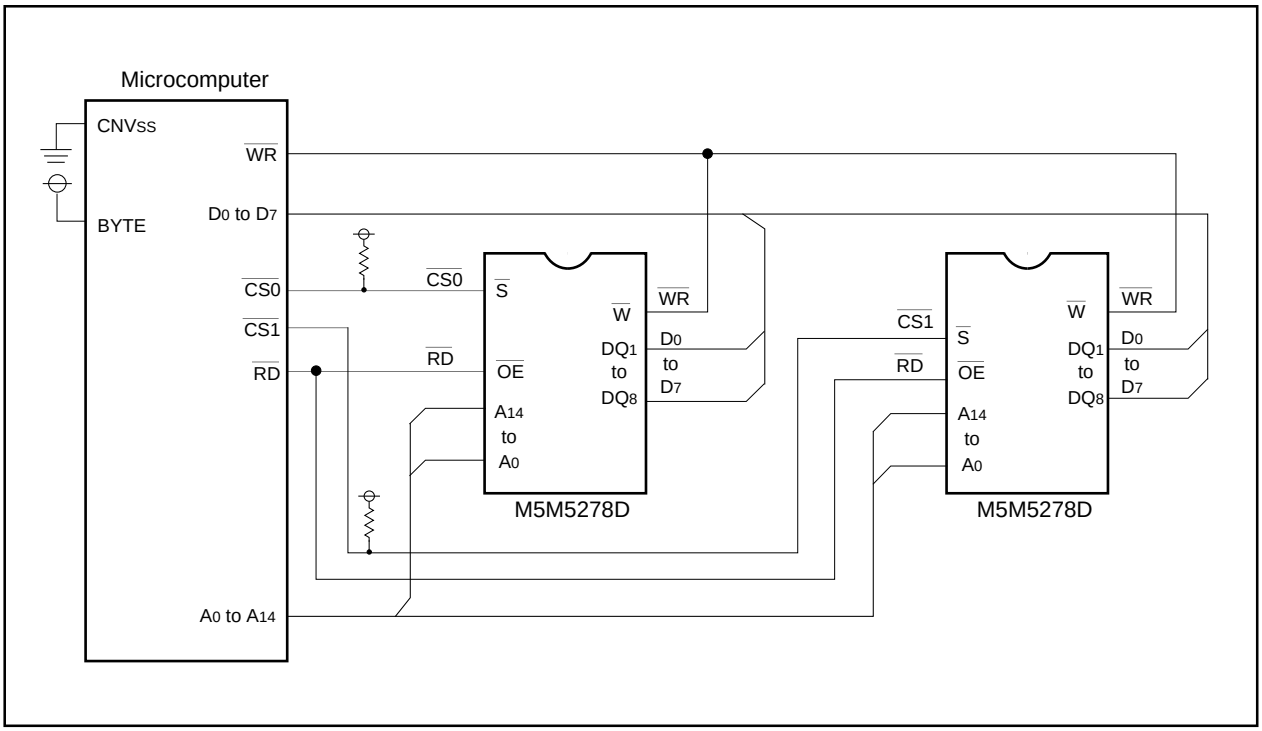


Figure 4.3.4. Example of connecting two M5M5278's to an 8-bit data bus

4.3.4 Two 8-bit and 16-Bit Memory to 16-Bit Width Data Bus Connection Example

Figure 4.3.5 shows an example of connecting M5M28F102 (16-bit flash memory) and two M5M5278's (8-bit SRAM) to a 16-bit data bus.

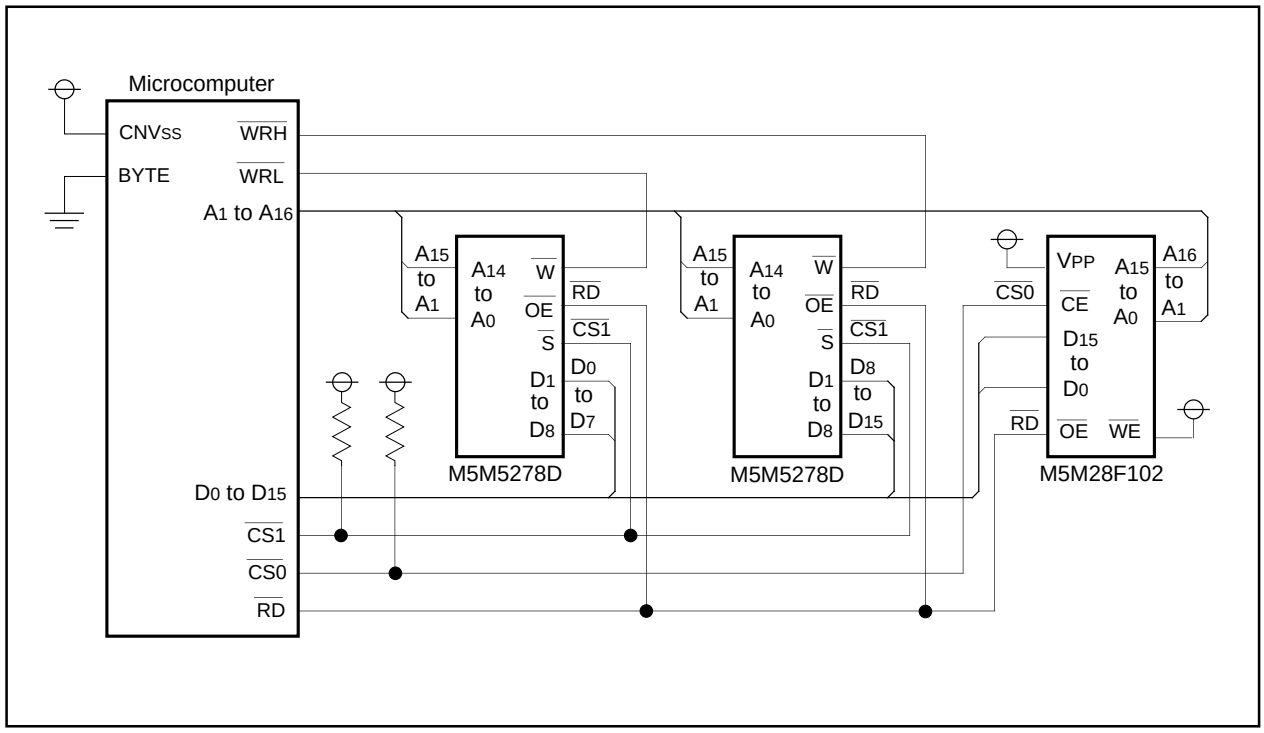


Figure 4.3.5. Example of connection of two 8-bit memories and one 16-bit memory to 16-bit width data bus

4.3.5 Chip Selects and Address Bus

When there are insufficient chip select signals, it is necessary to generate chip selects externally. Figure 4.3.6 shows an example of a connection in which the $\overline{CS2}$ (128K bytes) area is divided into four 32K byte areas.

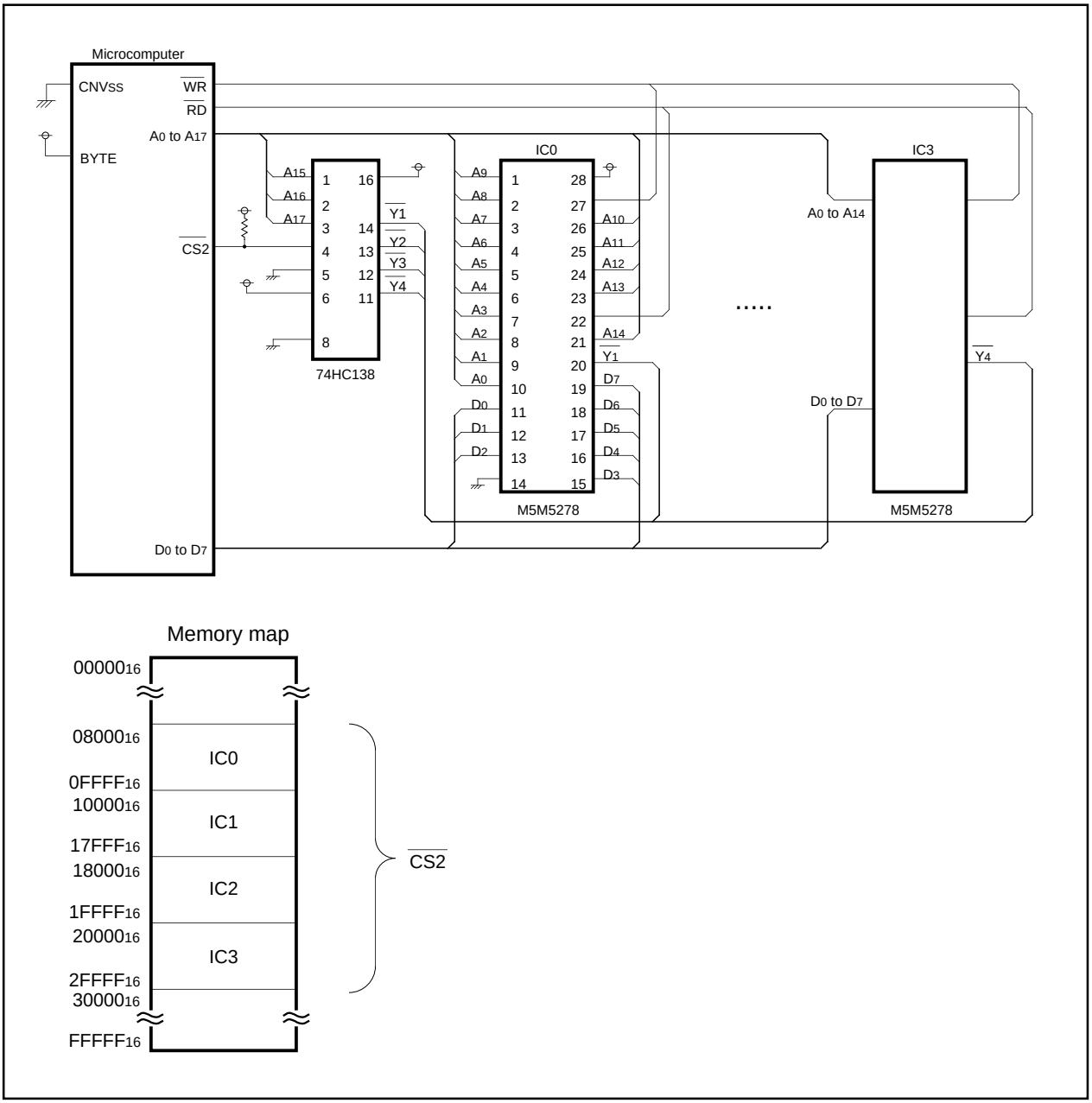


Figure 4.3.6. Chip selects and address bus

4.4 Connectable Memories

4.4.1 Operation Frequency and Access Time

Connectable memories depend upon the BCLK frequency f(BCLK). The frequency of f(BCLK) is equal to that of the BCLK , and is contingent on the oscillator's frequency and on the settings in the system clock select bits (bit 6 of address 0006_{16}, and bits 6 and 7 of address 0007_{16}).

The following are the conditional equations for the connections. Meet these conditions minimally. Figures 4.4.1 and 4.4.2 show the relation between the frequency of BCLK and memory.

(1) Read cycle time (tCR)/write cycle time (tCW)

Read cycle time (tCR) and write cycle time (tCW) must satisfy the following conditional expressions:

- **With the Wait option cleared**

  $tCR < 10^9/f(BCLK)$ and $tCW < 10^9/f(BCLK)$

- **With the Wait option selected**

  $tCR < 2 \times 10^9/f(BCLK)$ and $tCW < 2 \times 10^9/f(BCLK)$

(2) Address access time [ta(A)]

Address access time [ta(A)] must satisfy the following conditional expressions:

(a) Vcc = 5V

- **With the Wait option cleared**

  $ta(A) < 10^9/f(BCLK) - 65(ns)$*

- **With the Wait option selected**

  $ta(A) < 2 \times 10^9/f(BCLK) - 65(ns)$*

* 65(ns) = td(BCLK – AD) + tsu(DB – RD) – th(BCLK – RD)
= (address output delay time) + (data input setup time) – (RD signal output hold time)

(b) Vcc = 3V

- **With the Wait option cleared**

  $ta(A) < 10^9/f(BCLK) - 140(ns)$*

- **With the Wait option selected**

  $ta(A) < 2 \times 10^9/f(BCLK) - 140(ns)$*

* 140(ns) = td(BCLK-AD) + tsu(DB – RD) – th(BCLK – RD)
= (address output delay time) + (data input setup time) – (RD signal output hold time)

(3) Chip select access time [ta(S)]

Chip select access time [ta(S)] must satisfy the following conditional expressions:

(a) Vcc = 5V

- **With the Wait option cleared**

  $ta(S) < 10^9/f(BCLK) - 65(ns)$*

- **With the Wait option selected**

  $ta(S) < 2 \times 10^9/f(BCLK) - 65(ns)$*

* 65(ns) = td(BCLK – CS) + tsu(DB – RD) – th(BCLK – RD)
= (chip select output delay time) + (data input setup time) – (RD signal output hold time)

(b) Vcc = 3V

• With the Wait option cleared

ta(S) < 10^9/f(BCLK) – 140(ns)*

• With the Wait option selected

ta(S) < 2 X10^9/f(BCLK) – 140(ns)*

* 140(ns) = td(BCLK – CS) + tsu(DB – RD) – th(BCLK – RD)
= (chip select output delay time) + (data input setup time) – (RD signal output hold time)

(4) Output enable time [ta(OE)]

Output enable time [ta(OE)] must satisfy the following conditional expressions:

(a) Vcc = 5V

• With the Wait option cleared

ta(OE) < 10^9/(f(BCLK) X 2) – 45(ns) = tac1(RD-DB)

• With the Wait option selected

ta(OE) < 3 X 10^9/(f(BCLK) X 2) – 45(ns) = tac2(RD-DB)

(b) Vcc = 3V

• With the Wait option cleared

ta(OE) < 10^9/(f(BCLK) X 2) – 90(ns) = tac1(RD-DB)

• With the Wait option selected

ta(OE) < 3X10^9/(f(BCLK) X 2) – 90(ns) = tac2(RD-DB)

(5) Data setup time [tsu(D)]

Data setup time [tsu(D)] must satisfy the following conditional expressions:

(a) Vcc = 5V

• With the Wait option cleared

tsu(D) < 10^9/(f(BCLK) X 2) – 40(ns)*

• With the Wait option selected

tsu(D) < 10^9/f(BCLK) – 40(ns)*

* 40(ns) = td(BCLK – DB) – th(BCLK – WR)
= (data output delay time) – (WR signal output hold time)

(b) Vcc = 3V

• With the Wait option cleared

tsu(D) < 10^9/(f(BCLK) X 2) – 80(ns)*

• With the Wait option selected

tsu(D) < 10^9/f(BCLK) – 80(ns)*

* 80(ns) = td(BCLK – DB) – th(BCLK – WR)
= (data output delay time) – (WR signal output hold time)

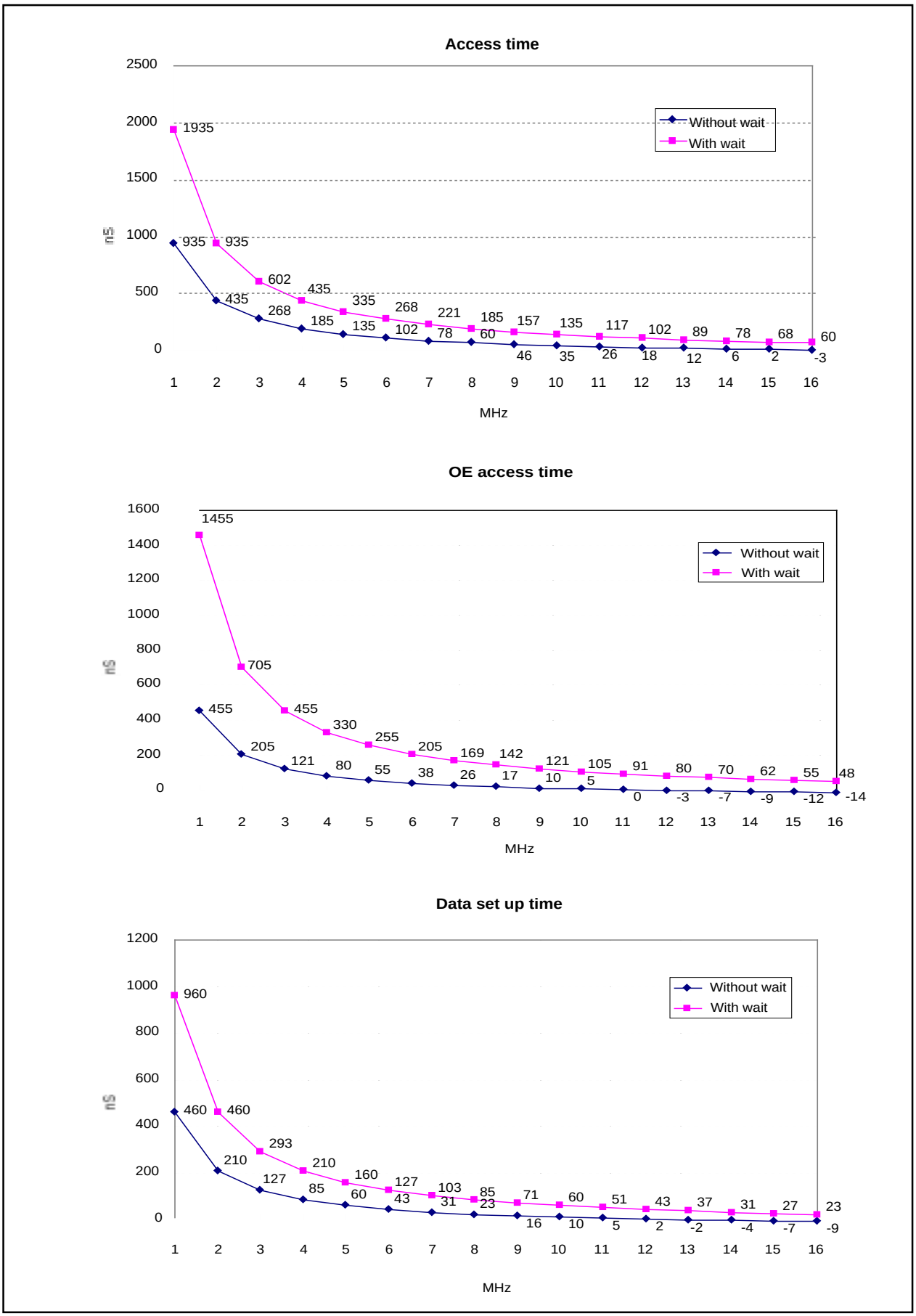


Figure 4.4.1. Relation between the frequency of BCLK and memory (Vcc = 5V)

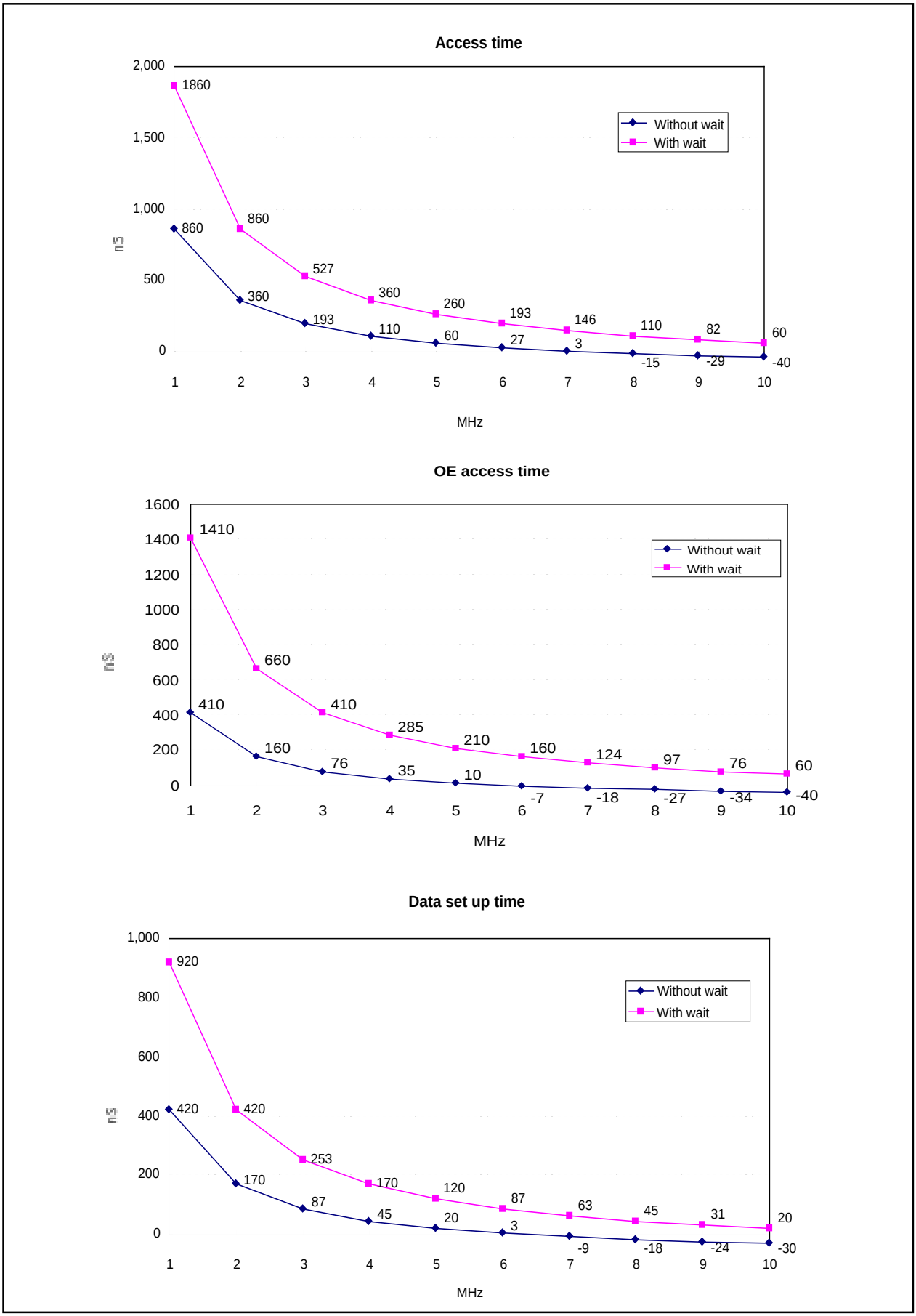


Figure 4.4.2. Relation between the frequency of BCLK and memory (Vcc = 3V)

4.4.2 Connecting Low-Speed Memory

To connect memory with long access time [ta(A)], either decrease the frequency of BCLK or set a software wait. Using the $\overline{RDY}$ feature allows you to connect memory having the timing that precludes connection though you set software wait.

(1) Using software wait

Set software wait by using either of bit 7 (PM17) of processor mode register 1 or bits 4 through 7 (CS0W through CS3W) of the chip select control register. With software wait set, if an address space is accessed in which a separate bus is selected, the bus cycle results in two cycles of BCLK; if an address space is accessed in which a multiplex bus is selected, the bus cycle results in three cycles of BCLK.

If bit 7 (PM17) of processor mode register 1 is set to “Wait selected”, the microcomputer accesses every area with this option in effect. If bit 7 (PM17) of processor mode register 1 is set to “Wait cleared”, the Wait option can be either selected or cleared, chip select by chip select, by setting bits 4 through 7 (CS0W through CS3W) of the chip select control register. Figures 4.4.3 through 4.4.5 show relation of processor mode and the wait bit (PM17, CSiW).

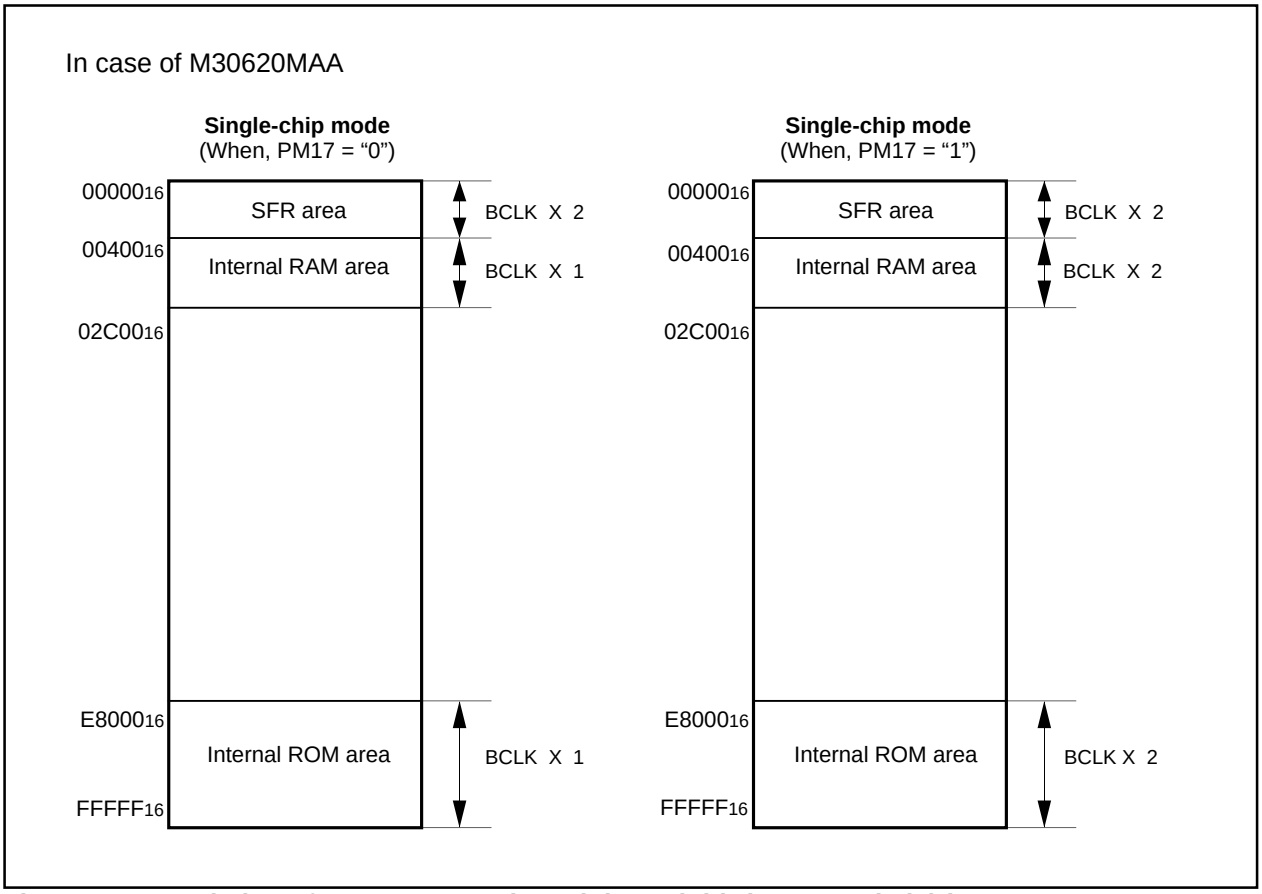


Figure 4.4.3. Relation of processor mode and the wait bit (PM17, CSiW) (1)

In case of M30620MAA

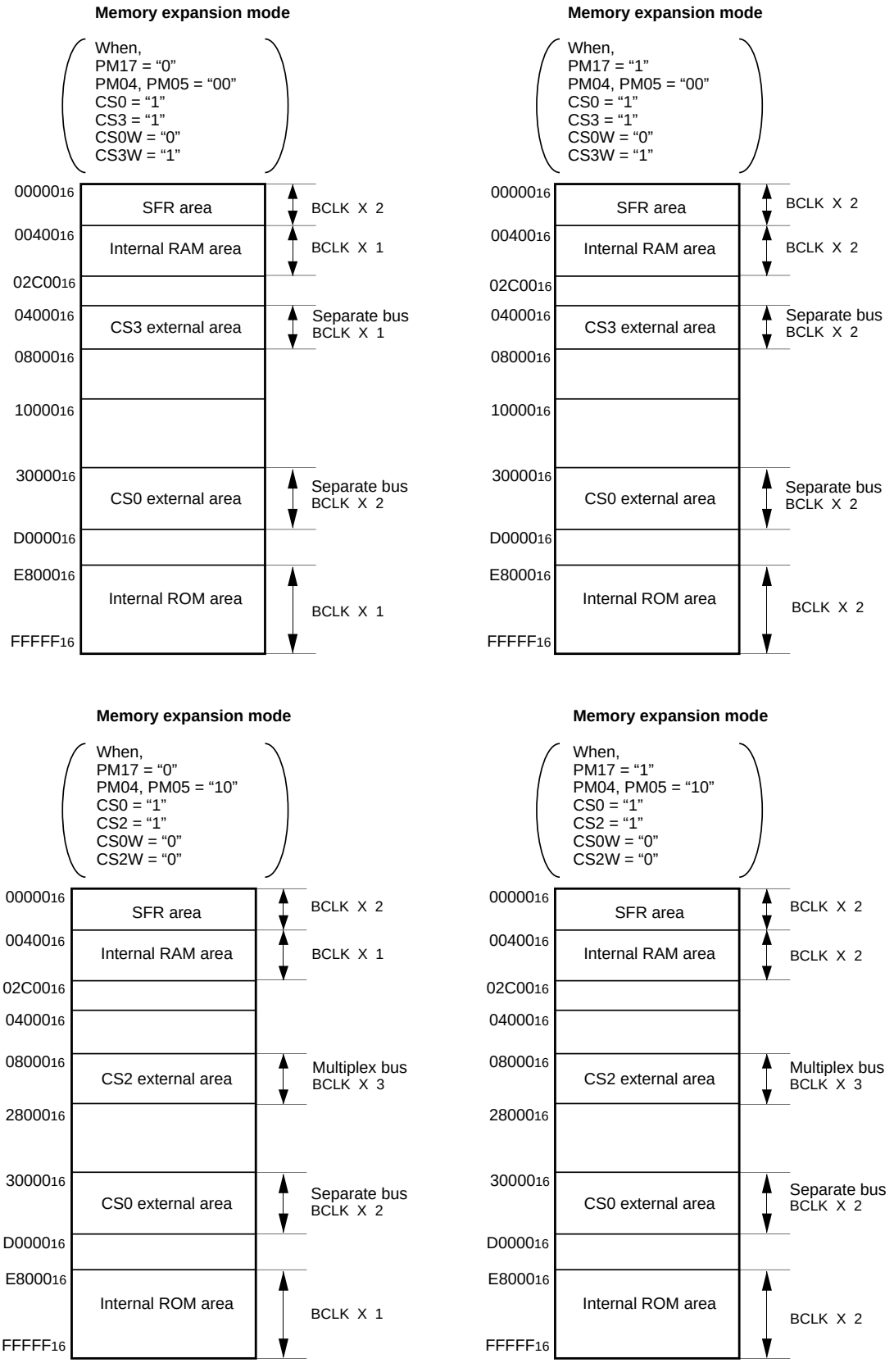


Figure 4.4.4. Relation of processor mode and the wait bit (PM17, CSiW) (2)

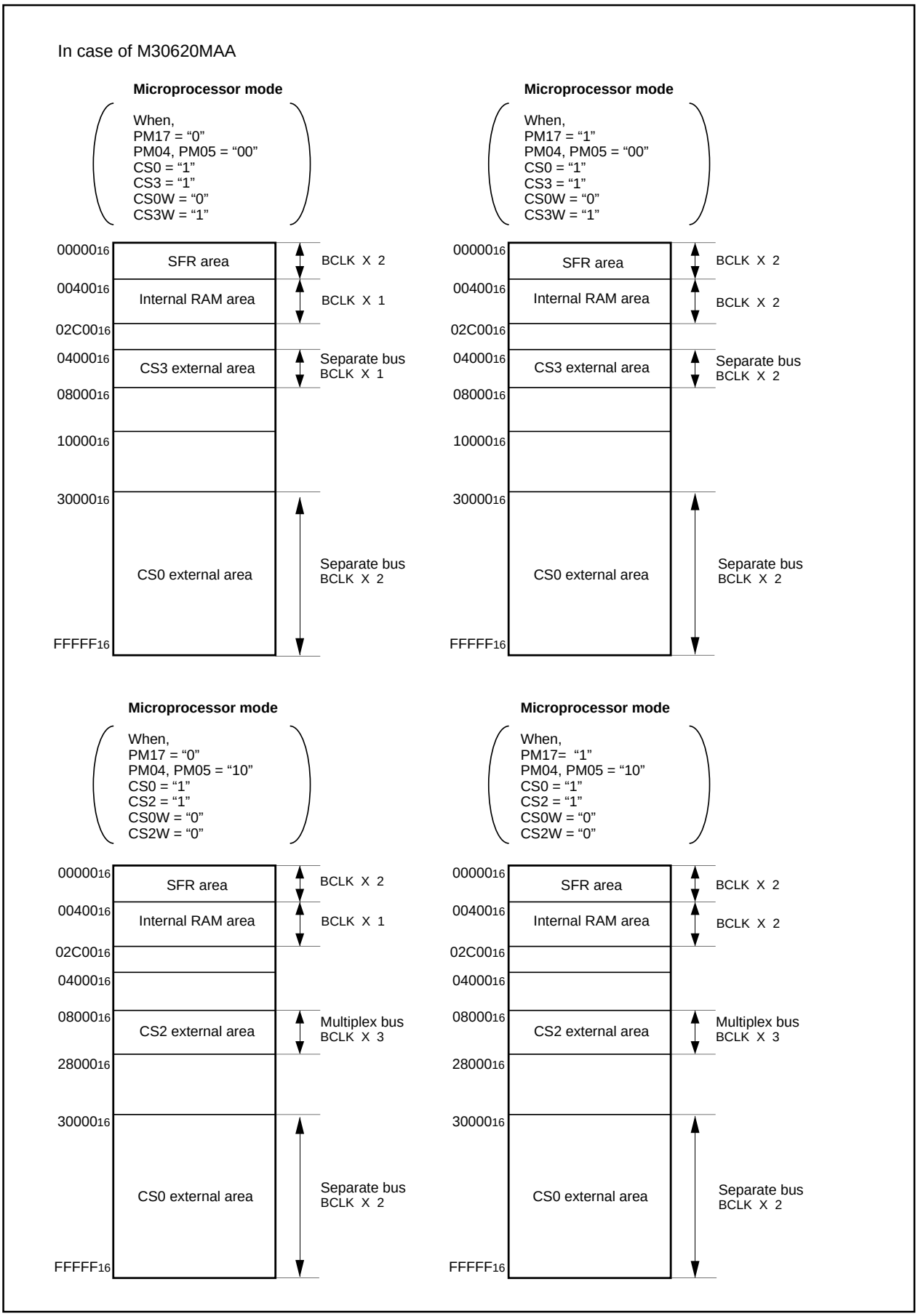


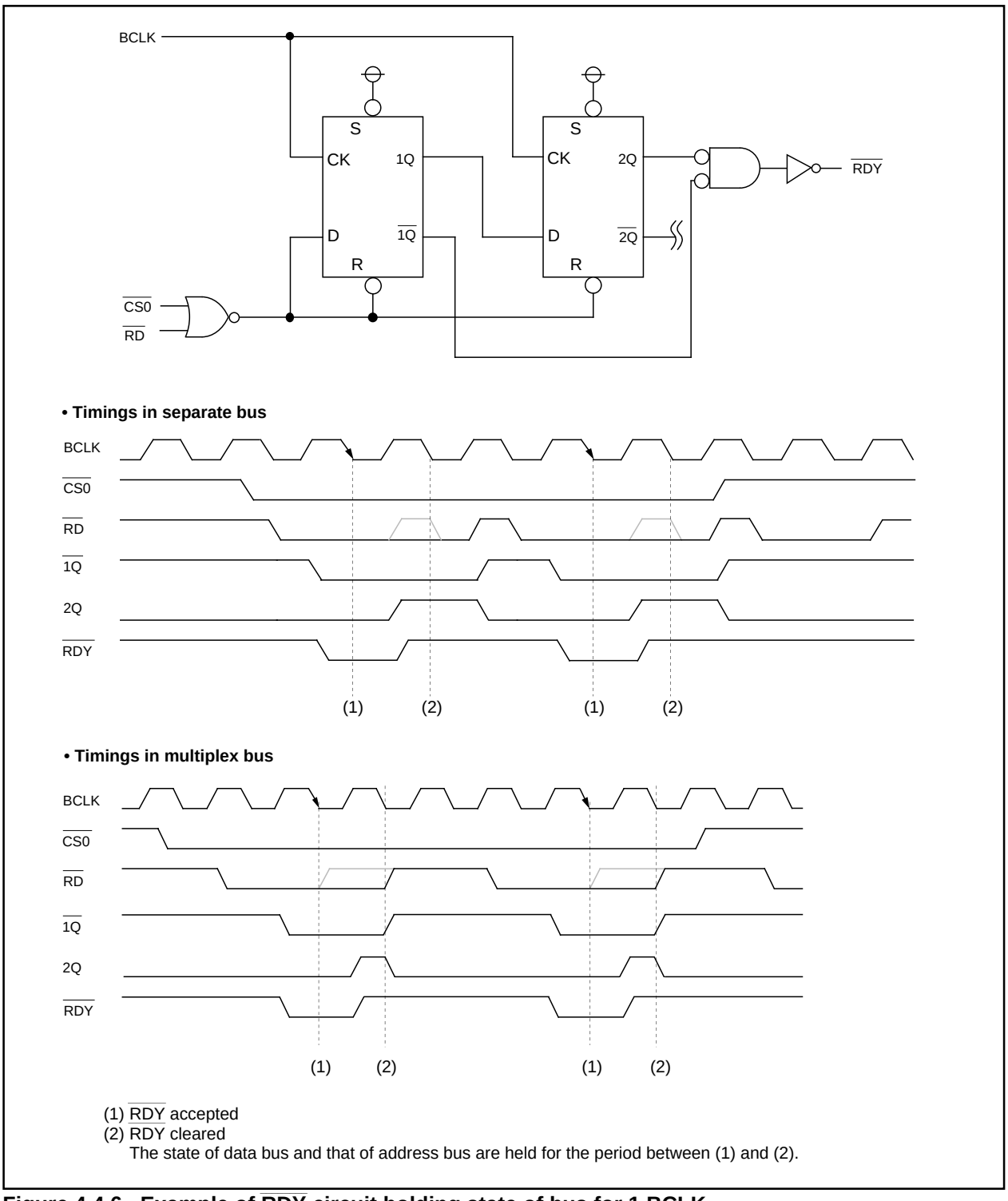
Figure 4.4.5. Relation of processor mode and the wait bit (PM17, CSiW) (3)

(2) $\overline{\text{RDY}}$ function usage

To use the $\overline{\text{RDY}}$ function, set a software wait.

The $\overline{\text{RDY}}$ function operates when the BCLK signal falls with the $\overline{\text{RDY}}$ pin at “L”; the bus does not vary for 1 BCLK, and the state at that moment is held.

The $\overline{\text{RDY}}$ function holds the state of bus for the period in which the $\overline{\text{RDY}}$ pin is at “L”, and releases it when the BCLK signal falls with the $\overline{\text{RDY}}$ pin at “H”. Figure 4.4.6 shows an example of $\overline{\text{RDY}}$ circuit that holds the state of bus for 1 BCLK.



(1) $\overline{\text{RDY}}$ accepted
(2) RDY cleared
The state of data bus and that of address bus are held for the period between (1) and (2).

Figure 4.4.6. Example of $\overline{\text{RDY}}$ circuit holding state of bus for 1 BCLK

4.4.3 Connectable Memories

Connectable memories and their maximum frequencies are given here;
M16C/62A group maximum frequency is
16MHz (without the wait) for Vcc=5V,
10MHz (with the one wait) for Vcc=3V

(1) Flash memories (Read only mode)

(a) 3V without wait

Maximum frequency (MHz)	Model No.
3.57	M5M29GB/T160BVP-80

(b) 3V with wait

Maximum frequency (MHz)	Model No.
8.33	M5M29GB/T160BVP-80

(2) SRAM

(a) 3V without wait

Maximum frequency (MHz)	Model No.
5.12	M5M54R08AJ-12 M5M54R16AJ,ATP-12

(b) 3V with wait

Maximum frequency (MHz)	Model No.
10.0	M5M54R08AJ-12 M5M54R16AJ,ATP-12

4.5 Releasing an External Bus ($\overline{\text{HOLD}}$ input and $\overline{\text{HLDA}}$ output)

The Hold feature is to relinquish the address bus, the data bus, and the control bus on M16C/62A side in line with the Hold request from the bus master other than M16C/62A when the two or more bus masters share the address bus, the data bus, and the control bus. The Hold feature is effective only in memory expansion mode and microprocessor mode.

The sequence of using the Hold feature may be:

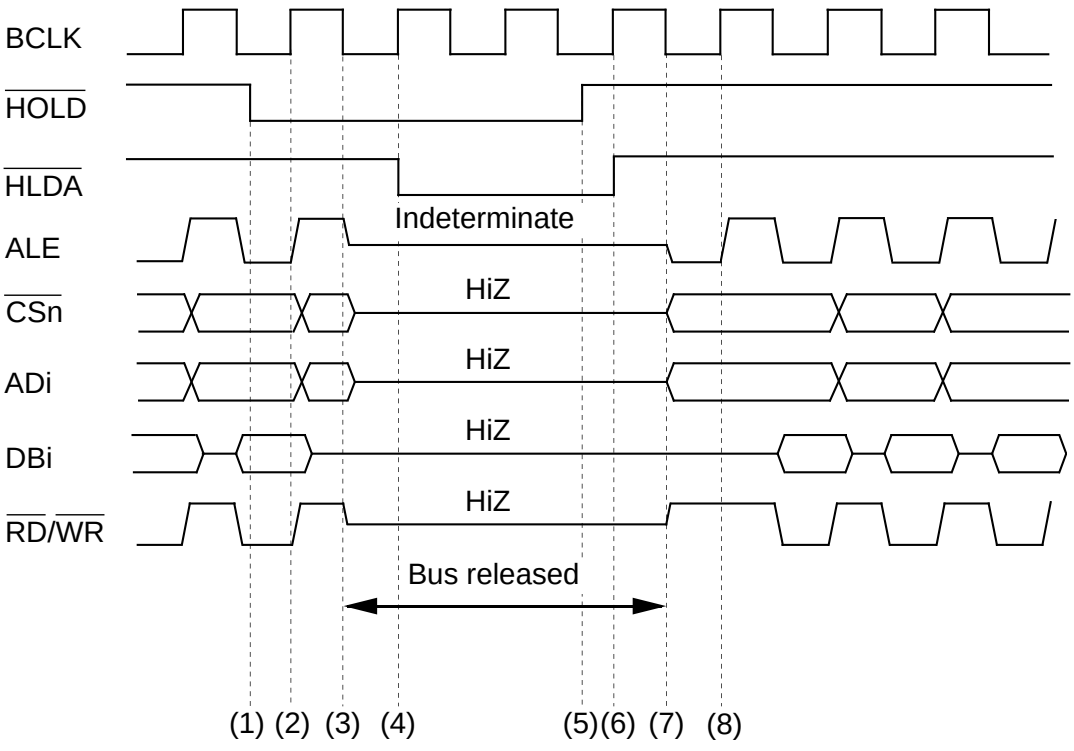
1. The external bus master turns the input level of the $\overline{\text{HOLD}}$ terminal to “L”.
2. When M16C/62A becomes ready to relinquish buses, each bus becomes high-impedance state at the falling edge of BCLK.
3. The $\overline{\text{HLDA}}$ terminal becomes “L” at the rising edge of the next BCLK.
4. The external bus master uses a bus.
5. When the external bus master finishes using a bus, the external bus master returns the input level of the $\overline{\text{HOLD}}$ terminal to “H”.
6. The output from $\overline{\text{HLDA}}$ terminal becomes “H” at the rising edge of the next BCLK.
7. Each bus returns from the high-impedance state to the former state at the falling edge of the next BCLK.

As given above, each bus invariably gets in the high-impedance state while the $\overline{\text{HLDA}}$ output is “L”. Also, M16C/62A does not relinquish buses during a bus cycle. That is, if a Hold request comes in during a bus cycle, the $\overline{\text{HLDA}}$ output become “L” after that bus cycle finishes.

In the Hold state, the state of each terminal becomes as follows.

- **Address bus A_0 to A_{19}**
  High-impedance state. The case in which A_{16} to A_{19} are used as ports $P4_0$ to $P4_3$ (64K byte address space) and the case in which A_9 to A_{19} are used as ports $P3_1$ to $P3_7$ and $P4_0$ to $P4_3$ (multiplex for the whole area) in microprocessor mode and in memory expansion mode too fall under this category.
- **Data bus D_0 to D_{15}**
  High-impedance state. The case in which D_8 to D_{15} are used as ports $P1_0$ to $P1_7$ (8-bit external bus width) and the case in which D_0 to D_{15} are used as ports $P0_0$ to $P0_7$ and $P1_0$ to $P1_7$ (multiplex for the whole area) in microprocessor mode and in memory expansion mode too fall under this category.
- **$\overline{\text{RD}}$, $\overline{\text{WR}}$, $\overline{\text{WRL}}$, $\overline{\text{WRH}}$, $\overline{\text{BHE}}$**
  High-impedance state.
- **ALE**
  An internal clock signal having the same phase as BCLK is output.
- **$\overline{\text{CS0}}$ to $\overline{\text{CS3}}$**
  High-impedance state. The case in which ports are selected by the chip selection control register too falls under this category.

Figure. 4.5.1 shows an example of relinquishing external buses.



(1) An “L” level is input to the $\overline{\text{HOLD}}$ pin.
(2) $\overline{\text{HOLD}}$ is detected.
(3) The CPU releases the bus.
(4) An “L” is output to the $\overline{\text{HLDA}}$ pin.
(5) An “H” is input to the $\overline{\text{HOLD}}$ pin.
(6) An “H” is output to the $\overline{\text{HLDA}}$ pin.
(7) The CPU not releases the bus.
(8) The CPU resumes using the bus.

Figure 4.5.1. Example of releasing the external bus

4.6 Precautions for External Bus

(1) The external ROM version can operate only in the microprocessor mode, so be sure to perform the following:
• Connect the CNVss pin to Vcc.

Chapter 5

External ROM Version

The external ROM version can operate only in the microprocessor mode.
Functions of the external ROM version differ from those of the mask ROM version in the following. therefore, only the differences are described in this chapter:
For the other functions, refer to chapters 1 to 4.

- Memory map
- Operated in only microprocessor mode

5.1 Pin Configuration

Figures 5.1.1 and 5.1.2 show the pin configrations (top view).

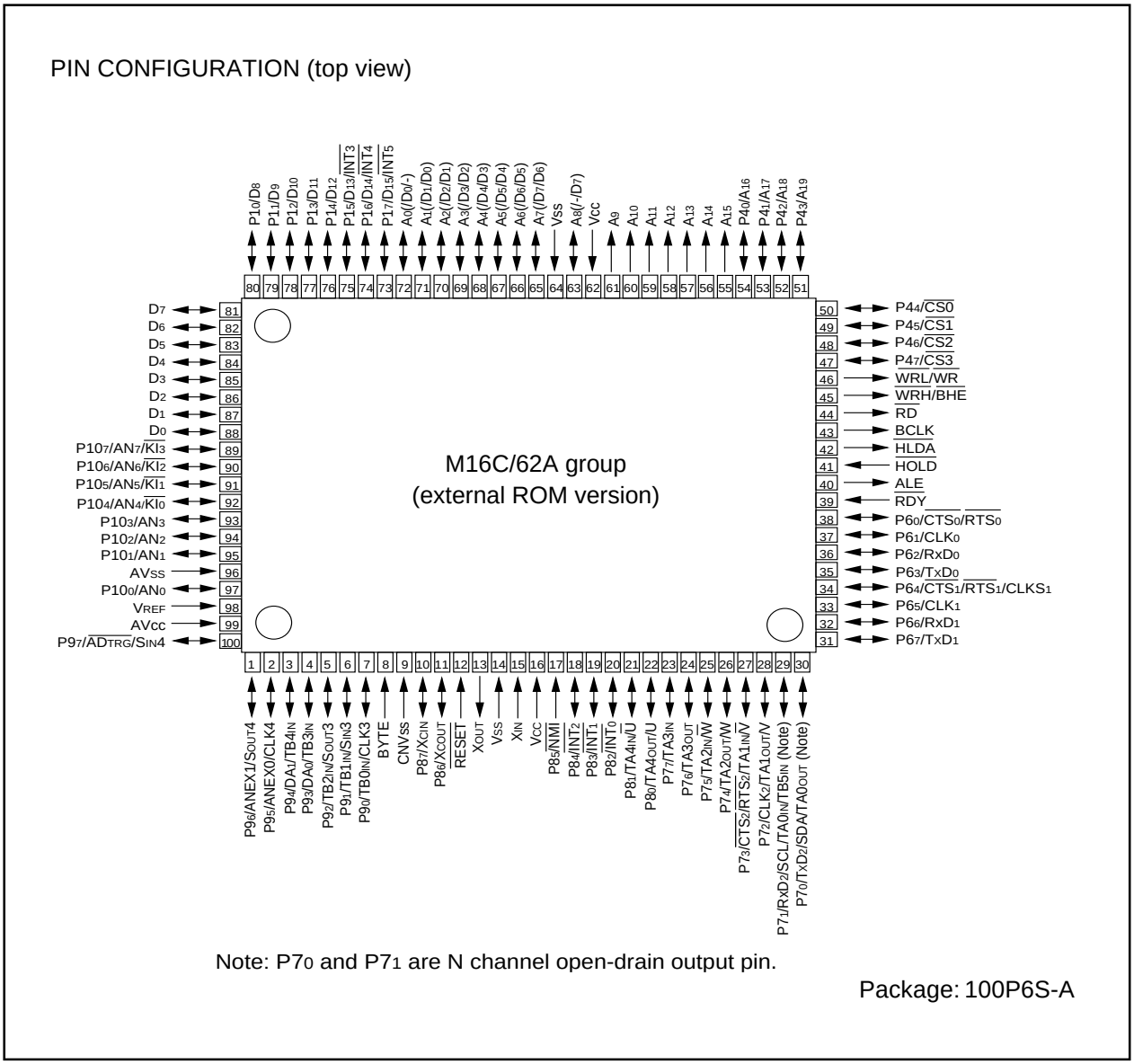
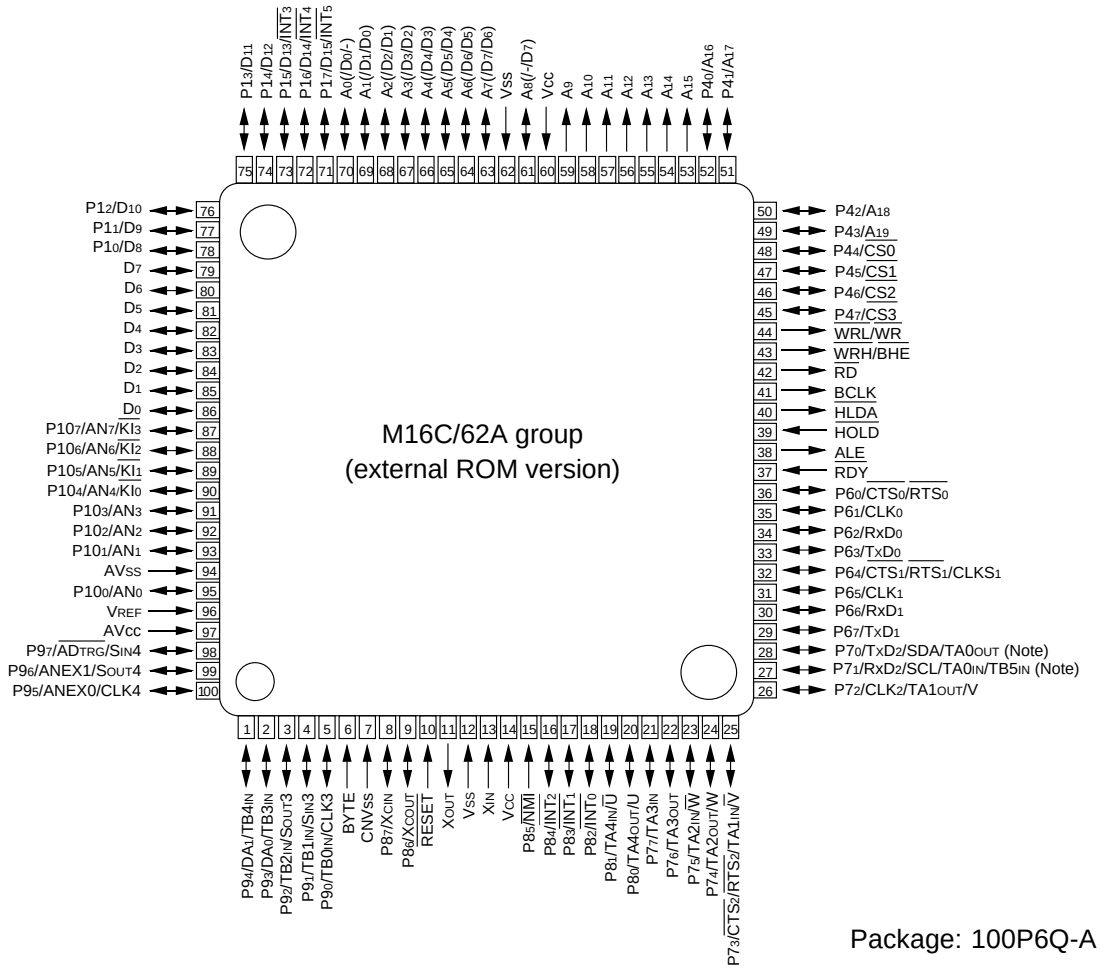


Figure 5.1.1. Pin configuration (top view)

PIN CONFIGURATION (top view)



Note: P70 and P71 are N channel open-drain output pin.

Figure 5.1.2. Pin configuration (top view)

5.2 Pin Description

Tables 5.2.1 and 5.2.2 show the pin description.

Table 5.2.1. Pin Description (1)

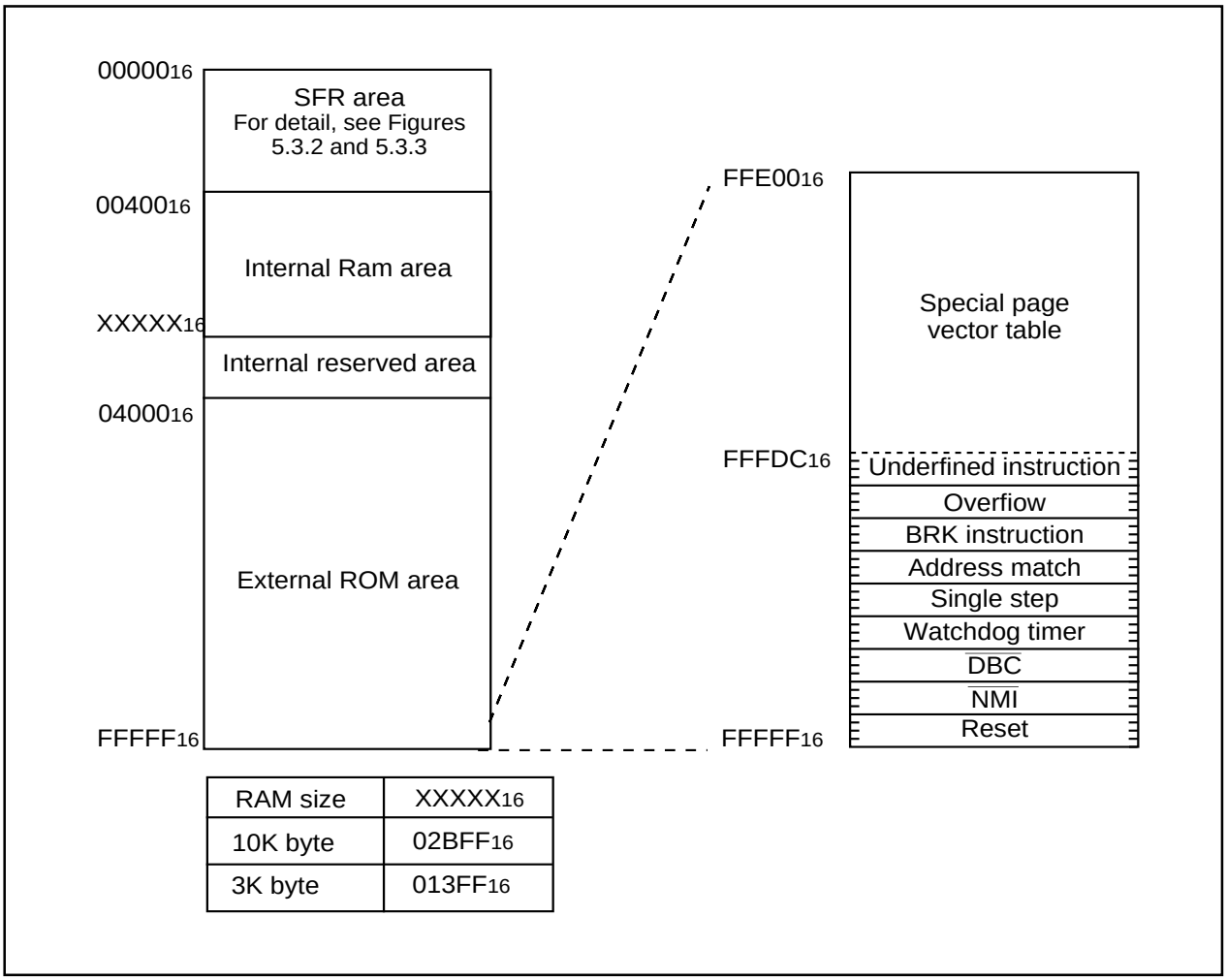
Pin name	Signal name	I/O type	Function
V_{CC}, V_{SS}	Power supply input		Supply 2.7 to 5.5 V to the V_{CC} pin. Supply 0 V to the V_{SS} pin.
CNV_{SS}	CNV_{SS}	Input	Connect this pin to V_{CC}.
$\overline{RESET}$	Reset input	Input	A "L" on this input resets the microcomputer.
X_{IN} X_{OUT}	Clock input Clock output	Input Output	These pins are provided for the main clock generating circuit. Connect a ceramic resonator or crystal between the X_{IN} and the X_{OUT} pins. To use an externally derived clock, input it to the X_{IN} pin and leave the X_{OUT} pin open.
BYTE	External data bus width select input	Input	This pin selects the width of an external data bus. A 16-bit width is selected when this input is "L"; an 8-bit width is selected when this input is "H". This input must be fixed to either "H" or "L".
AV_{CC}	Analog power supply input		This pin is a power supply input for the A-D converter. Connect this pin to V_{CC}.
AV_{SS}	Analog power supply input		This pin is a power supply input for the A-D converter. Connect this pin to V_{SS}.
V_{REF}	Reference voltage input	Input	This pin is a reference voltage input for the A-D converter.
D_0 to D_7	Data bus	Input/output	When set as a separate bus, these pins input and output data (D_0–D_7).
$P1_0$ to $P1_7$	I/O port P1	Input/output	This is an 8-bit CMOS I/O port. It has an input/output port direction register that allows the user to set each pin for input or output individually.
D_8 to D_{15}	Data bus	Input/output	When set as a separate bus, these pins input and output data (D_8–D_{15}).
A_0 to A_7	Address bus	Output	These pins output 8 low-order address bits (A_0–A_7).
A_0/D_0 to A_7/D_7	Address bus/ data bus	Input/output	If the external bus is set as an 8-bit wide multiplexed bus, these pins input and output data (D_0–D_7) and output 8 low-order address bits (A_0–A_7) separated in time by multiplexing.
A_0, A_1/D_0 to A_7/D_6		Output Input/output	If the external bus is set as a 16-bit wide multiplexed bus, these pins input and output data (D_0–D_6) and output address (A_1–A_7) separated in time by multiplexing. They also output address (A_0).
A_8 to A_{15}	Address bus	Output	These pins output 8 middle-order address bits (A_8–A_{15}).
A_8/D_7, A_9 to A_{15}	Address bus/ data bus	Input/output Output	If the external bus is set as a 16-bit wide multiplexed bus, these pins input and output data (D_7) and output address (A_8) separated in time by multiplexing. They also output address (A_9–A_{15}).
$P4_0$ to $P4_7$	I/O port P4	Input/output	This is an 8-bit I/O port equivalent to P1.
$\overline{CS_0}$ to $\overline{CS_3}$, A_{16} to A_{19}		Output Output	These pins output $\overline{CS_0}$–$\overline{CS_3}$ signals and A_{16}–A_{19}. $\overline{CS_0}$–$\overline{CS_3}$ are chip select signals used to specify an access space. A_{16}–A_{19} are 4 high-order address bits.

Table 5.2.2. Pin Description (2)

Pin name	Signal name	I/O type	Function
WRL / WR, WRH / BHE, RD, BCLK, HLDA, HOLD, ALE, RDY	WRL / WR, WRH / BHE, RD, BCLK, HLDA, HOLD, ALE, RDY	Output Output Output Output Output Input Output Input	Output WRL, WRH (WR and BHE), RD, BCLK, HLDA, and ALE signals. WRL and WRH, and BHE and WR can be switched using software control. ■ WRL, WRH, and RD selecte With a 16-bit external data bus, data is written to even addresses when the WRL signal is “L” and to the odd addresses when the WRH signal is “L”. Data is read when RD is “L”. ■ WR, BHE, and RD selected Data is written when WR is “L”. Data is read when RD is “L”. Odd addresses are accessed when BHE is “L”. Use this mode when using an 8-bit external data bus. While the input level at the HOLD pin is “L”, the microcomputer is placed in the hold state. While in the hold state, HLDA outputs a “L" level. ALE is used to latch the address. While the input level of the RDY pin is “L”, the microcomputer is in the ready state.
$P6_0$ to $P6_7$	I/O port P6	Input/output	This is an 8-bit I/O port equivalent to P1. The port can be set to have or not have a pull-up resistor in units of four bits by software. Pins in this port also function as UART0 and UART1 I/O pins as selected by software.
$P7_0$ to $P7_7$	I/O port P7	Input/output	This is an 8-bit I/O port equivalent to P6. Pins in this port also function as timer A0–A3, timer B5, or UART2 I/O pins as selected by software.
$P8_0$ to $P8_4$, $P8_6$, $P8_7$, $P8_5$	I/O port P8 I/O port $P8_5$	Input/output Input/output Input/output Input	$P8_0$ to $P8_4$, $P8_6$, and $P8_7$ are I/O ports with the same functions as P6. Using software, they can be made to function as the I/O pins for timer A4 and the input pins for external interrupts. $P8_6$ and $P8_7$ can be set using software to function as the I/O pins for a sub clock generation circuit. In this case, connect a quartz oscillator between $P8_6$ (X_{COUT} pin) and $P8_7$ (X_{CIN} pin). $P8_5$ is an input-only port that also functions for NMI. The NMI interrupt is generated when the input at this pin changes from “H” to “L”. The NMI function cannot be canceled using software. The pull-up cannot be set for this pin.
$P9_0$ to $P9_7$	I/O port P9	Input/output	This is an 8-bit I/O port equivalent to P6. Pins in this port also function as SI/O3,4 I/O pin, timer B0–B2 input pins, D-A converter output pins, A-D converter’s extended input pins, or A-D trigger input pins as selected by software.
$P10_0$ to $P10_7$	I/O port P10	Input/output	This is an 8-bit I/O port equivalent to P6. Pins in this port also function as A-D converter input pins. Furthermore, $P10_4$–$P10_7$ also function as input pins for the key input interrupt function.

5.3 Memory Map

Figure 5.3.1 shows the memory map. Figures 5.3.2 and 5.3.3 show the SFR memory map.



RAM size	XXXXX₁₆
10K byte	02BFF₁₆
3K byte	013FF₁₆

Figure 5.3.1. Memory map

0000₁₆	
0001₁₆	
0002₁₆	
0003₁₆	
0004₁₆	Processor mode register 0 (PM0)
0005₁₆	Processor mode register 1(PM1)
0006₁₆	System clock control register 0 (CM0)
0007₁₆	System clock control register 1 (CM1)
0008₁₆	Chip select control register (CSR)
0009₁₆	Address match interrupt enable register (AIER)
000A₁₆	Protect register (PRCR)
000B₁₆	
000C₁₆	
000D₁₆	
000E₁₆	Watchdog timer start register (WDTS)
000F₁₆	Watchdog timer control register (WDC)
0010₁₆	Address match interrupt register 0 (RMAD0)
0011₁₆	
0012₁₆	
0013₁₆	
0014₁₆	Address match interrupt register 1 (RMAD1)
0015₁₆	
0016₁₆	
0017₁₆	
0018₁₆	
0019₁₆	
001A₁₆	
001B₁₆	
001C₁₆	
001D₁₆	
001E₁₆	
001F₁₆	
0020₁₆	DMA0 source pointer (SAR0)
0021₁₆	
0022₁₆	
0023₁₆	
0024₁₆	DMA0 destination pointer (DAR0)
0025₁₆	
0026₁₆	
0027₁₆	
0028₁₆	DMA0 transfer counter (TCR0)
0029₁₆	
002A₁₆	
002B₁₆	
002C₁₆	DMA0 control register (DM0CON)
002D₁₆	
002E₁₆	
002F₁₆	
0030₁₆	DMA1 source pointer (SAR1)
0031₁₆	
0032₁₆	
0033₁₆	
0034₁₆	DMA1 destination pointer (DAR1)
0035₁₆	
0036₁₆	
0037₁₆	
0038₁₆	DMA1 transfer counter (TCR1)
0039₁₆	
003A₁₆	
003B₁₆	
003C₁₆	DMA1 control register (DM1CON)
003D₁₆	
003E₁₆	
003F₁₆	

0040₁₆	
0041₁₆	
0042₁₆	
0043₁₆	
0044₁₆	INT3 interrupt control register (INT3IC)
0045₁₆	Timer B5 interrupt control register (TB5IC)
0046₁₆	Timer B4 interrupt control register (TB4IC)
0047₁₆	Timer B3 interrupt control register (TB3IC)
0048₁₆	SI/O4 interrupt control register (S4IC) INT5 interrupt control register (INT5IC)
0049₁₆	SI/O3 interrupt control register (S3IC) INT4 interrupt control register (INT4IC)
004A₁₆	Bus collision detection interrupt control register (BCNIC)
004B₁₆	DMA0 interrupt control register (DM0IC)
004C₁₆	DMA1 interrupt control register (DM1IC)
004D₁₆	Key input interrupt control register (KUPIC)
004E₁₆	A-D conversion interrupt control register (ADIC)
004F₁₆	UART2 transmit interrupt control register (S2TIC)
0050₁₆	UART2 receive interrupt control register (S2RIC)
0051₁₆	UART0 transmit interrupt control register (S0TIC)
0052₁₆	UART0 receive interrupt control register (S0RIC)
0053₁₆	UART1 transmit interrupt control register (S1TIC)
0054₁₆	UART1 receive interrupt control register (S1RIC)
0055₁₆	Timer A0 interrupt control register (TA0IC)
0056₁₆	Timer A1 interrupt control register (TA1IC)
0057₁₆	Timer A2 interrupt control register (TA2IC)
0058₁₆	Timer A3 interrupt control register (TA3IC)
0059₁₆	Timer A4 interrupt control register (TA4IC)
005A₁₆	Timer B0 interrupt control register (TB0IC)
005B₁₆	Timer B1 interrupt control register (TB1IC)
005C₁₆	Timer B2 interrupt control register (TB2IC)
005D₁₆	INT0 interrupt control register (INT0IC)
005E₁₆	INT1 interrupt control register (INT1IC)
005F₁₆	INT2 interrupt control register (INT2IC)
0060₁₆	
0061₁₆	
0062₁₆	
0063₁₆	
0064₁₆	
0065₁₆	
≈	≈
032A₁₆	
032B₁₆	
032C₁₆	
032D₁₆	
032E₁₆	
032F₁₆	
0330₁₆	
0331₁₆	
0332₁₆	
0333₁₆	
0334₁₆	
0335₁₆	
0336₁₆	
0337₁₆	
0338₁₆	
0339₁₆	
033A₁₆	
033B₁₆	
033C₁₆	
033D₁₆	
033E₁₆	
033F₁₆	

Note : Locations in the SFR area where nothing is allocated are reserved areas. Do not access these areas for read or write.

Figure 5.3.2. SFR memory map (1)

0340₁₆	Timer B3, 4, 5 count start flag (TBSR)
0341₁₆	
0342₁₆	Timer A1-1 register (TA11)
0343₁₆	
0344₁₆	Timer A2-1 register (TA21)
0345₁₆	
0346₁₆	Timer A4-1 register (TA41)
0347₁₆	
0348₁₆	Three-phase PWM control register 0(INVC0)
0349₁₆	Three-phase PWM control register 1(INVC1)
034A₁₆	Three-phase output buffer register 0(IDB0)
034B₁₆	Three-phase output buffer register 1(IDB1)
034C₁₆	Dead time timer(DTT)
034D₁₆	Timer B2 interrupt occurrence frequency set counter(ICTB2)
034E₁₆	
034F₁₆	
0350₁₆	Timer B3 register (TB3)
0351₁₆	
0352₁₆	Timer B4 register (TB4)
0353₁₆	
0354₁₆	Timer B5 register (TB5)
0355₁₆	
0356₁₆	
0357₁₆	
0358₁₆	
0359₁₆	
035A₁₆	
035B₁₆	Timer B3 mode register (TB3MR)
035C₁₆	Timer B4 mode register (TB4MR)
035D₁₆	Timer B5 mode register (TB5MR)
035E₁₆	
035F₁₆	Interrupt cause select register (IFSR)
0360₁₆	SI/O3 transmit/receive register (S3TRR)
0361₁₆	
0362₁₆	SI/O3 control register (S3C)
0363₁₆	SI/O3 bit rate generator (S3BRG)
0364₁₆	SI/O4 transmit/receive register (S4TRR)
0365₁₆	
0366₁₆	SI/O4 control register (S4C)
0367₁₆	SI/O4 bit rate generator (S4BRG)
0368₁₆	
0369₁₆	
036A₁₆	
036B₁₆	
036C₁₆	
036D₁₆	
036E₁₆	
036F₁₆	
0370₁₆	
0371₁₆	
0372₁₆	
0373₁₆	
0374₁₆	
0375₁₆	
0376₁₆	UART2 special mode register 2(U2SMR2)
0377₁₆	UART2 special mode register (U2SMR)
0378₁₆	UART2 transmit/receive mode register (U2MR)
0379₁₆	UART2 bit rate generator (U2BRG)
037A₁₆	UART2 transmit buffer register (U2TB)
037B₁₆	
037C₁₆	UART2 transmit/receive control register 0 (U2C0)
037D₁₆	UART2 transmit/receive control register 1 (U2C1)
037E₁₆	UART2 receive buffer register (U2RB)
037F₁₆	

0380₁₆	Count start flag (TABSR)
0381₁₆	Clock prescaler reset flag (CPSRF)
0382₁₆	One-shot start flag (ONSF)
0383₁₆	Trigger select register (TRGSR)
0384₁₆	Up-down flag (UDF)
0385₁₆	
0386₁₆	Timer A0 register (TA0)
0387₁₆	
0388₁₆	Timer A1 register (TA1)
0389₁₆	
038A₁₆	Timer A2 register (TA2)
038B₁₆	
038C₁₆	Timer A3 register (TA3)
038D₁₆	
038E₁₆	Timer A4 register (TA4)
038F₁₆	
0390₁₆	Timer B0 register (TB0)
0391₁₆	
0392₁₆	Timer B1 register (TB1)
0393₁₆	
0394₁₆	Timer B2 register (TB2)
0395₁₆	
0396₁₆	Timer A0 mode register (TA0MR)
0397₁₆	Timer A1 mode register (TA1MR)
0398₁₆	Timer A2 mode register (TA2MR)
0399₁₆	Timer A3 mode register (TA3MR)
039A₁₆	Timer A4 mode register (TA4MR)
039B₁₆	Timer B0 mode register (TB0MR)
039C₁₆	Timer B1 mode register (TB1MR)
039D₁₆	Timer B2 mode register (TB2MR)
039E₁₆	
039F₁₆	
03A0₁₆	UART0 transmit/receive mode register (U0MR)
03A1₁₆	UART0 bit rate generator (U0BRG)
03A2₁₆	UART0 transmit buffer register (U0TB)
03A3₁₆	
03A4₁₆	UART0 transmit/receive control register 0 (U0C0)
03A5₁₆	UART0 transmit/receive control register 1 (U0C1)
03A6₁₆	UART0 receive buffer register (U0RB)
03A7₁₆	
03A8₁₆	UART1 transmit/receive mode register (U1MR)
03A9₁₆	UART1 bit rate generator (U1BRG)
03AA₁₆	UART1 transmit buffer register (U1TB)
03AB₁₆	
03AC₁₆	UART1 transmit/receive control register 0 (U1C0)
03AD₁₆	UART1 transmit/receive control register 1 (U1C1)
03AE₁₆	UART1 receive buffer register (U1RB)
03AF₁₆	
03B0₁₆	UART transmit/receive control register 2 (UCON)
03B1₁₆	
03B2₁₆	
03B3₁₆	
03B4₁₆	
03B5₁₆	
03B6₁₆	
03B7₁₆	
03B8₁₆	DMA0 request cause select register (DM0SL)
03B9₁₆	
03BA₁₆	DMA1 request cause select register (DM1SL)
03BB₁₆	
03BC₁₆	CRC data register (CRCD)
03BD₁₆	
03BE₁₆	CRC input register (CRCIN)
03BF₁₆	

Note : Locations in the SFR area where nothing is allocated are reserved areas. Do not access these areas for read or write.

Figure 5.3.3. SFR memory map (2)

03C0₁₆ 03C1₁₆	A-D register 0 (AD0)
03C2₁₆ 03C3₁₆	A-D register 1 (AD1)
03C4₁₆ 03C5₁₆	A-D register 2 (AD2)
03C6₁₆ 03C7₁₆	A-D register 3 (AD3)
03C8₁₆ 03C9₁₆	A-D register 4 (AD4)
03CA₁₆ 03CB₁₆	A-D register 5 (AD5)
03CC₁₆ 03CD₁₆	A-D register 6 (AD6)
03CE₁₆ 03CF₁₆	A-D register 7 (AD7)
03D0₁₆	
03D1₁₆	
03D2₁₆	
03D3₁₆	
03D4₁₆	A-D control register 2 (ADCON2)
03D5₁₆	
03D6₁₆	A-D control register 0 (ADCON0)
03D7₁₆	A-D control register 1 (ADCON1)
03D8₁₆	D-A register 0 (DA0)
03D9₁₆	
03DA₁₆	D-A register 1 (DA1)
03DB₁₆	
03DC₁₆	D-A control register (DACON)
03DD₁₆	
03DE₁₆	
03DF₁₆	
03E0₁₆	
03E1₁₆	Port P1 register (P1)
03E2₁₆	
03E3₁₆	Port P1 direction register (PD1)
03E4₁₆	
03E5₁₆	
03E6₁₆	
03E7₁₆	
03E8₁₆	Port P4 register (P4)
03E9₁₆	
03EA₁₆	Port P4 direction register (PD4)
03EB₁₆	
03EC₁₆	Port P6 register (P6)
03ED₁₆	Port P7 register (P7)
03EE₁₆	Port P6 direction register (PD6)
03EF₁₆	Port P7 direction register (PD7)
03F0₁₆	Port P8 register (P8)
03F1₁₆	Port P9 register (P9)
03F2₁₆	Port P8 direction register (PD8)
03F3₁₆	Port P9 direction register (PD9)
03F4₁₆	Port P10 register (P10)
03F5₁₆	
03F6₁₆	Port P10 direction register (PD10)
03F7₁₆	
03F8₁₆	
03F9₁₆	
03FA₁₆	
03FB₁₆	
03FC₁₆	Pull-up control register 0 (PUR0)
03FD₁₆	Pull-up control register 1 (PUR1)
03FE₁₆	Pull-up control register 2 (PUR2)
03FF₁₆	Port control register (PCR)

Note : Locations in the SFR area where nothing is allocated are reserved areas. Do not access these areas for read or write.

Figure 5.3.4. SFR memory map (3)

5.4 Processor Mode

The external ROM version is operated only in microprocessor mode, so be sure to perform the following:

- Connect CNVSS pin to VCC.

Figure 5.4.1 shows the processor mode register 0.

Processor mode register 0 (Note)

b7	b6	b5	b4	b3	b2	b1	b0
						1	1

Symbol: PM0 — Address: 0004_{16} — When rese: 03_{16}

Bit symbol	Bit name	Function	R	W
PM00	Processor mode bit	b1 b0 0 0: Single-chip mode 0 1: Memory expansion mode 1 0: Must not be set 1 1: Microprocessor mode	○	○
PM01			○	○
PM02	R/W mode select bit	0 : $\overline{RD}$,$\overline{BHE}$,$\overline{WR}$ 1 : RD,WRH,WRL	○	○
PM03	Software reset bit	The device is reset when this bit is set to “1”. The value of this bit is “0” when read.	○	○
PM04	Multiplexed bus space select bit	b5 b4 0 0 : Multiplexed bus is not used 0 1 : Allocated to CS2 space 1 0 : Allocated to CS1 space 1 1 : Must not be set	○	○
PM05			○	○
PM06	Port $P4_0$ to $P4_3$ function select bit	0 : Address output 1 : Port function (Address is not output)	○	○
PM07	BCLK output disable bit	0 : BCLK is output 1 : BCLK is not output (Pin is left floating)	○	○

Note: Set bit 1 of the protect register (address $000A_{16}$) to “1” when writing new values to this register.

Figure 5.4.1. Processor mode register 0

Appendix 1 Check Sheet

The following check sheet was created based on items which had been the source of problems in the past. We recommend you refer to the check sheet when troubleshooting.

☐ Are you making use of Technical News?
For the latest copy of Technical News, contact an authorized dealer.

Checks regarding register initial settings

☐ Has the initial setting been made in the interrupt stack pointer (ISP) at the top of the program?

☐ Has the initial setting been made in the user stack pointer (USP)? (Only if using the USP)

☐ Does the USP overlap the ISP area? (Only if using the USP)

☐ Is interrupt enabled after setting the ISP and USP?

☐ Is the top address of the variable interrupt vector table set in the interrupt table register (INTB)?

☐ Is interrupt enabled after setting the INTB?

☐ Has the initial setting been made in the frame base register (FB)? (Only if using the FB)

☐ Has the initial setting been made in the stack base register (SB)? (Only if using the SB)

Checks regarding the internal memory

☐ Does the RAM capacity used in the program exceed the RAM capacity of the microcomputer?

☐ Does the ROM capacity used in the program exceed the ROM capacity of the microcomputer?

Checks regarding the protect register

☐ Is writing enabled in the protect register (address $000A_{16}$) before writing in the system clock control register (addresses 0006_{16} and 0007_{16})?

☐ Is writing enabled in the protect register before writing in the processor mode register (addresses 0004_{16} and 0005_{16})?

☐ Is writing enabled in the protect register before writing in the port P9 direction register (address $03F3_{16}$)?

☐ Is writing effectuated in the port P9 direction register by the next instruction after writing is enabled in the protect register?

☐ Does not an interrupt generate between the instruction writing is enabled in the protect register and the instruction writing in the port P9 direction register?

- ☐ Does not instruction DMA transfer occur between the instruction writing is enabled in the protect register and the instruction writing in the port P9 direction register starts?
- ☐ Is writing enabled in the protect register before writing in the SI/Oi (i=3,4) control register (address 0362_{16}, 0366_{16})?
- ☐ Is writing effectuated in the SI/Oi (i=3,4) control register by the next instruction after writing is enabled in the protect register?
- ☐ Does not an interrupt generate between the instruction writing is enabled in the protect register and the instruction writing in the SI/Oi (i=3,4) control register?
- ☐ Does not instruction DMA transfer occur between the instruction writing is enabled in the protect register and the instruction writing in the SI/Oi (i=3,4) control register starts?

Checks regarding the timer

- ☐ Is the timer started after a value is set in the timer register?

Checks regarding Interrupt

- ☐ When rewrite the interrupt register, do so at a point that does not generate the interruput request?

Checks regarding low voltage and low power consumption

- ☐ When using at low voltage, have you checked recommended operating conditions and changed the wait bit (address 0005_{16}, bit 7) to “1”?
- ☐ Does the oscillator to which the count source is going to be switched be oscillating stably, before the count source for BCLK can be changed from XIN to XCIN or vice versa?
- ☐ In the low power consumption mode, does not current flow from Vref when the Vref connection bit (bit 5 in address $03D7_{16}$) is set?
- ☐ Is not voltage level of port floating in the low power consumption mode?

Checks regarding A-D converter

- ☐ Have you selected other than fAD (no dividing) for øAD when using the A-D converter at V_{CC} = 2.7V to 4.0V?
- ☐ Have you selected no sample & hold function when using the A-D converter at V_{CC} = 2.7V to 4.0V?
- ☐ Have you selected 8-bit mode when using the A-D converter at V_{CC} = 2.7V to 4.0V?
- ☐ Does øAD use it by less than 10MHz?

Appendix 2 Hexadecimal instruction CODE table

	D7 to D4	0000	0001	0010	0011	0100	0101	0110	0111
D3 to D0		0	1	2	3	4	5	6	7
0000	0	BRK	AND.B:S R0H,R0L	ADD.B:S R0H,R0L	MOV.B:S R0H,A0	BCLR:S 0,11[SB]	BNOT:S 0,11[SB]	JMP.S label	MULU.B src,dest
0001	1	MOV.B:S R0L,dsp:8[SB]	AND.B:S dsp:8[SB],R0L	ADD.B:S dsp:8[SB],R0L	MOV.B:S dsp:8[SB],A0	BCLR:S 1,11[SB]	BNOT:S 1,11[SB]	JMP.S label	MULU.W src,dest
0010	2	MOV.B:S R0L,dsp:8[FB]	AND.B:S dsp:8[FB],R0L	ADD.B:S dsp:8[FB],R0L	MOV.B:S dsp:8[FB],A0	BCLR:S 2,11[SB]	BNOT:S 2,11[SB]	JMP.S label	MOV.B:G src,dest
0011	3	MOV.B:S R0L,abs16	AND.B:S abs16,R0L	ADD.B:S abs16,R0L	MOV.B:S abs16,A0	BCLR:S 3,11[SB]	BNOT:S 3,11[SB]	JMP.S label	MOV.W:G src,dest
0100	4	NOP	AND.B:S R0L,R0H	ADD.B:S R0L,R0H	MOV.B:S R0Çk,A1	BCLR:S 4,11[SB]	BNOT:S 4,11[SB]	JMP.S label	CODE_74
0101	5	MOV.B:S R0H,dsp:8[SB]	AND.B:S dsp:8[SB],R0H	ADD.B:S dsp:8[SB],R0H	MOV.B:S dsp:8[SB],A1	BCLR:S 5,11[SB]	BNOT:S 5,11[SB]	JMP.S label	CODE_75
0110	6	MOV.B:S R0H,dsp:8[FB]	AND.B:S dsp:8[FB],R0H	ADD.B:S dsp:8[FB],R0H	MOV.B:S dsp:8[FB],A1	BCLR:S 6,11[SB]	BNOT:S 6,11[SB]	JMP.S label	CODE_76
0111	7	MOV.B:S R0H,abs16	AND.B:S abs16,R0H	ADD.B:S abs16,R0H	MOV.B:S abs16,A1	BCLR:S 7,11[SB]	BNOT:S 7,11[SB]	JMP.S label	CODE_77
1000	8	MOV.B:S R0H,R0L	OR.B:S R0H,R0L	SUB.B:S R0H,R0L	CMP.B:S R0H,R0L	BSET:S 0,11[SB]	BTST:S 0,11[SB]	JGEU/C label	MUL.B src,dest
1001	9	MOV.B:S dsp:8[SB],R0L	OR.B:S dsp:8[SB],R0L	SUB.B:S dsp:8[SB],R0L	CMP.B:S dsp:8[SB],R0L	BSET:S 1,11[SB]	BTST:S 1,11[SB]	JGTU label	MUL.W src,dest
1010	A	MOV.B:S dsp:8[FB],R0L	OR.B:S dsp:8[FB],R0L	SUB.B:S dsp:8[FB],R0L	CMP.B:S dsp:8[FB],R0L	BSET:S 2,11[SB]	BTST:S 2,11[SB]	JEQ/Z label	CODE_7A
1011	B	MOV.B:S abs16,R0L	OR.B:S abs16,R0L	SUB.B:S abs16,R0L	CMP.B:S abs16,R0L	BSET:S 3,11[SB]	BTST:S 3,11[SB]	JN label	CODE_7B
1100	C	MOV.B:S R0L,R0H	OR.B:S R0L,R0H	SUB.B:S R0L,R0H	CMP.B:S R0L,R0H	BSET:S 4,11[SB]	BTST:S 4,11[SB]	JLTU/NC label	CODE_7C
1101	D	MOV.B:S dsp:8[SB],R0H	OR.B:S dsp:8[SB],R0H	SUB.B:S dsp:8[SB],R0H	CMP.B:S dsp:8[SB],R0H	BSET:S 5,11[SB]	BTST:S 5,11[SB]	JLEU label	CODE_7D
1110	E	MOV.B:S dsp:8[FB],R0H	OR.B:S dsp:8[FB],R0H	SUB.B:S dsp:8[FB],R0H	CMP.B:S dsp:8[FB],R0H	BSET:S 6,11[SB]	BTST:S 6,11[SB]	JNE/JNZ label	CODE_7E
1111	F	MOV.B:S abs16,R0H	OR.B:S abs16,R0H	SUB.B:S abs16,R0H	CMP.B:S abs16,R0H	BSET:S 7,11[SB]	BTST:S 7,11[SB]	JPZ label	

The next instruction is arranged in each CODE.

CODE_74 : STE, MOV, PUSH, NEG, ROT, NOT, LDE, POP, SHL, SHA

CODE_75 : STE, MOV, PUSH, NEG, ROT, NOT, LDE, POP, SHL, SHA

CODE_76 : TST, XOR, AND, OR, ADD, SUB, ADC, SBB, CMP, DIVX, ROLC, RORC, DIVU, DIV, ADCF, ABS

CODE_77 : TST, XOR, AND, OR, ADD, SUB, ADC, SBB, CMP, DIVX, ROLC, RORC, DIVU, DIV, ADCF, ABS

CODE_7A : XCHG, LDC

CODE_7B : XCHG, STC

CODE_7C : MOVDir, MULU, MUL, EXTS, STC, DIVU, DIV, PUSH, DIVX, DADD, DSUB, DADC, DSBB, SMOVF, SMOVB, SSTR, ADD, LDCTX, RMPA, ENTER

CODE_7D : JMPI, JSRI, MULU, MUL, PUSHA, LDIPL, ADD, JCnd, BMCnd, DIVU, DIV, PUSH, DIVX, DADD, DSUB, DADC, DSBB, SMOVF, SMOVB, SSTR, STCTX, RMPA, EXITD, WAIT

CODE_7E : BTSTC, BMCnd, BNTST, BAND, BNAND, BOR, BNOR, BCLR, BSET, BNOT, BTST, BXOR, BNXOR

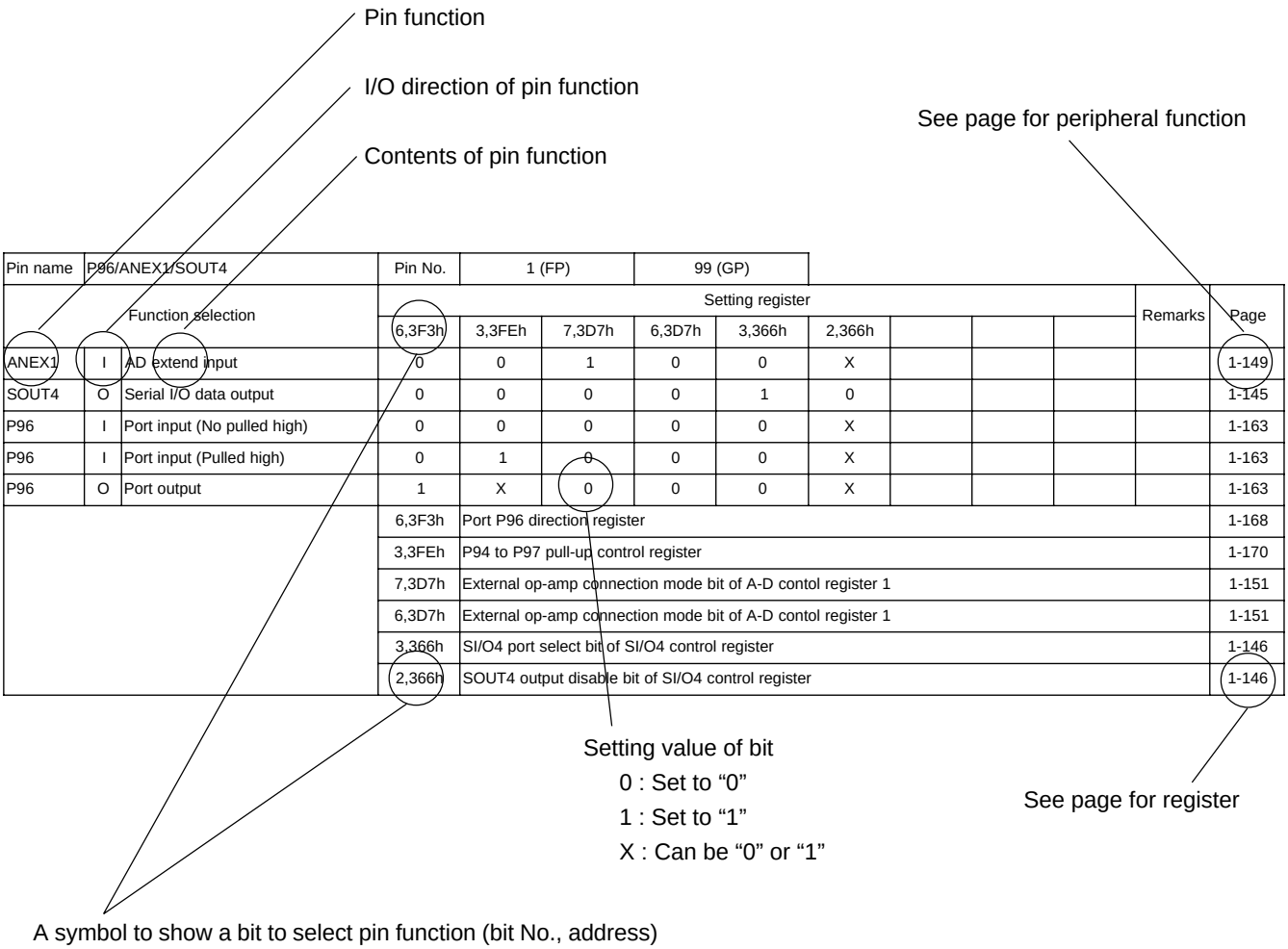
CODE_EB:SHL,FSET,FCLR,MOVA,LDC,SHA,PUSHC,POPC,INT

Appendix 2 Hexadecimal instruction CODE table

	D7 to D4	1000	1001	1010	1011	1100	1101	1110	1111
D3 to D0		8	9	A	B	C	D	E	F
0000	0	TST.B src,dest	AND.B:G src,dest	ADD.B:G src,dest	ADC.B src,dest	CMP.B:G src,dest	CMP.B:Q #IMM,dest	ROT.B #IMM,dest	SHA.B #IMM,dest
0001	1	TST.W src,dest	AND.W:G src,dest	ADD.W:G src,dest	ADC.w src,dest	CMP.W:G src,dest	CMP.W:Q #IMM,dest	ROT.W #IMM,dest	SHA.W #IMM,dest
0010	2	PUSH.B:S R0L	POP.B:S R0L	MOV.W:S #IMM,A0	INC.W A0	PUSH.W:S A0	POP.W:S A0	MOV.B:S #IMM,A0	DEC.W A0
0011	3	ADD.B:S #IMM8,R0H	AND.B:S #IMM8,R0H	INC.B R0H	MOV.B:Z #0,R0H	MOV.B:S #IMM8,R0H	STNZ #IMM8,R0H	CMP.B:S #IMM8,R0H	RTS
0100	4	ADD.B:S #IMM8,R0L	AND.B:S #IMM8,R0L	INC.B R0L	MOV.B:Z #0,R0L	MOV.B:S #IMM8,R0L	STNZ #IMM8,R0L	CMP.B:S #IMM8,R0L	JMP.W label
0101	5	ADD.B:S #IMM8,dsp:8[SB]	AND.B:S #IMM8,dsp:8[SB]	INC.B dsp:8[SB]	MOV.B:Z #0,dsp:8[SB]	MOV.B:S #IMM8,dsp:8[SB]	STNZ #IMM8,dsp:8[SB]	CMP.B:S #IMM8,dsp:8[SB]	JSR.W label
0110	6	ADD.B:S #IMM8,dsp:8[FB]	AND.B:S #IMM8,dsp:8[FB]	INC.B dsp:8[FB]	MOV.B:Z #0,dsp:8[FB]	MOV.B:S #IMM8,dsp:8[FB]	STNZ #IMM8,dsp:8[FB]	CMP.B:S #IMM8,dsp:8[FB]	INTO
0111	7	ADD.B:S #IMM8,abs16	AND.B:S #IMM8,abs16	INC.B abs16	MOV.B:Z #0,abs16	MOV.B:S #IMM8,abs16	STNZ #IMM8,abs16	CMP.B:S #IMM8,abs16	
1000	8	XOR.B src,dest	OR.B:G src,dest	SUB.B:G src,dest	SBB.B src,dest	ADD.B:Q #IMM,dest	MOV.B:Q #IMM,dest	SHL.B #IMM,dest	ADJNZ.B #IMM,dest,label
1001	9	XOR.W src,dest	OR.W:G src,dest	SUB.W:G src,dest	SBB.W src,dest	ADD.W:Q #IMM,dest	MOV.W:Q #IMM,dest	SHL.W #IMM,dest	ADJNZ.W #IMM,dest,label
1010	A	PUSH.B:S R0H	POP.B:S R0H	MOV.W:S #IMM,A1	INC.W A1	PUSH.W:S A1	POP.W:S A1	MOV.B:S #IMM,A1	DEC.W A1
1011	B	SUB.B:S #IMM8,R0H	OR.B:S #IMM8,R0H	DEC.B R0H	NOT.B:S R0H	STZ #IMM8,R0H	STZX #IMM8,#IMM8,R0H	CODE_EB	REIT
1100	C	SUB.B:S #IMM8,R0L	OR.B:S #IMM8,R0L	DEC.B R0L	NOT.B:S R0L	STZ #IMM8,R0L	STZX #IMM8,#IMM8,R0L	PUSHM src	JMP.A label
1101	D	SUB.B:S #IMM8,dsp:8[SB]	OR.B:S #IMM8,dsp:8[SB]	DEC.B dsp:8[SB]	NOT.B:S dsp:8[SB]	STZ #IMM8,dsp:8[SB]	STZX #IMM8,#IMM8,dsp:8[SB]	POPM dest	JSR.A label
1110	E	SUB.B:S #IMM8,dsp:8[FB]	OR.B:S #IMM8,dsp:8[FB]	DEC.B dsp:8[FB]	NOT.B:S dsp:8[FB]	STZ #IMM8,dsp:8[FB]	STZX #IMM8,#IMM8,dsp:8[FB]	JMPS #IMM8	JMP.B label
1111	F	SUB.B:S #IMM8,abs16	OR.B:S #IMM8,abs16	DEC.B abs16	NOT.B:S abs16	STZ #IMM8,abs16	STZX #IMM8,#IMM8,abs16	JSRS #IMM8	UND

Appendix 3 Setting register for pin function selected

The following shows the register to select pin and the setting value.
Refer to the table as follows.



Pin name	P96/ANEX1/SOUT4		Pin No.	1 (FP)		99 (GP)									
Function selection			Setting register										Remarks	Page
			6,3F3h	3,3FEh	7,3D7h	6,3D7h	3,366h	2,366h						
ANEX1	I	AD extend input	0	0	1	0	0	X						1-149
SOUT4	O	Serial I/O data output	0	0	0	0	1	0						1-145
P96	I	Port input (No pulled high)	0	0	0	0	0	X						1-163
P96	I	Port input (Pulled high)	0	1	0	0	0	X						1-163
P96	O	Port output	1	X	0	0	0	X						1-163
			6,3F3h	Port P96 direction register										1-168
			3,3FEh	P94 to P97 pull-up control register										1-170
			7,3D7h	External op-amp connection mode bit of A-D contol register 1										1-151
			6,3D7h	External op-amp connection mode bit of A-D contol register 1										1-151
			3,366h	SI/O4 port select bit of SI/O4 control register										1-146
			2,366h	SOUT4 output disable bit of SI/O4 control register										1-146

Pin name	P96/ANEX1/SOUT4		Pin No.	1 (FP)		99 (GP)							Remarks	Page
Function selection			Setting register											
			6,3F3h	3,3FEh	7,3D7h	6,3D7h	3,366h	2,366h						
ANEX1	I	AD extend input	0	0	1	0	0	X						1-149
SOUT4	O	Serial I/O data output	0	0	0	0	1	0						1-145
P96	I	Port input (No pulled high)	0	0	0	0	0	X						1-163
P96	I	Port input (Pulled high)	0	1	0	0	0	X						1-163
P96	O	Port output	1	X	0	0	0	X						1-163
			6,3F3h	Port P96 direction register										1-168
			3,3FEh	P94 to P97 pull-up control register										1-170
			7,3D7h	External op-amp connection mode bit of A-D contol register 1										1-151
			6,3D7h	External op-amp connection mode bit of A-D contol register 1										1-151
			3,366h	SI/O4 port select bit of SI/O4 control register										1-146
			2,366h	SOUT4 output disable bit of SI/O4 control register										1-146

Pin name	P95/ANEX0/CLK4		Pin No.	2 (FP)		100 (GP)							Remarks	Page
Function selection			Setting register											
			5,3F3h	3,3FEh	7,3D7h	6,3D7h	3,366h	6,366h						
ANEX0	I	AD extend input	0	0	0	1	0	X						1-149
ANEX0	O	Output when op-amp connection mode	0	0	1	1	0	X						1-149
SCLK4	O	Serial clock output	0	0	0	0	1	1						1-145
SCLK4	I	Serial clock input	0	0	0	0	1	0						1-145
P95	I	Port input (No pulled high)	0	0	0	0	0	X						1-163
P95	I	Port input (Pulled high)	0	1	0	0	0	X						1-163
P95	O	Port output	1	X	0	0	0	X						1-163
			5,3F3h	Port P95 direction register										1-168
			3,3FEh	P94 to P97 pull-up control register										1-170
			7,3D7h	External op-amp connection mode bit of A-D control register 1										1-151
			6,3D7h	External op-amp connection mode bit of A-D control register 1										1-151
			3,366h	SI/O4 port select bit of SI/O4 control register										1-146
			6,366h	Synchronous clock select bit of SI/O4 control register										1-146

Pin name	P94/DA1/TB4IN		Pin No.	3 (FP)		1 (GP)							Remarks	Page
Function selection			Setting register											
			4,3F3h	3,3FEh	1,3DCh	7,35Ch								
DA1	O	DA output	0	0	1	0								1-159
TB4IN	I	Count source input	0	0	0	0								1-90
P94	I	Port input (No pulled high)	0	0	0	0								1-163
P94	I	Port input (Pulled high)	0	1	0	0								1-163
P94	O	Port output	1	X	0	0								1-163
			4,3F3h	Port P94 direction register										1-168
			3,3FEh	P94 to P97 pull-up control register										1-170
			1,3DCh	D-A1 output enable bit of D-A control register										1-160
			7,35Ch	Event clock select bit ot timer B4 mode register										1-90

Appendix 3 Setting register for pin function selected

Pin name	P93/DA0/TB3IN		Pin No.	4 (FP)		2 (GP)							
Function selection			Setting register									Remarks	Page
			3,3F3h	2,3FEh	0,3DCh	7,35Bh							
DA0	O	DA output	0	0	1	0							1-159
TB3IN	I	Count source input	0	0	0	0							1-90
P93	I	Port input (No pulled high)	0	0	0	0							1-163
P93	I	Port input (Pulled high)	0	1	0	0							1-163
P93	O	Port output	1	X	0	0							1-163
			3,3F3h	Port P93 direction register									1-168
			2,3FEh	P90 to P93 pull-up control register									1-170
			0,3DCh	D-A0 output enable bit of D-A control register									1-160
			7,35Bh	Event clock select bit ot timer B3 mode register									1-90

Pin name	P92/TB2IN/SOUT3		Pin No.	5 (FP)		3 (GP)							
Function selection			Setting register									Remarks	Page
			2,3F3h	2,3FEh	7,39Dh	3,362h	2,362h						
TB2IN	I	Count source input	0	0	0	0	X						1-90
SOUT3	O	Serial I/O data output	0	0	0	1	0						1-145
P92	I	Port input (No pulled high)	0	0	X	0	X						1-163
P92	I	Port input (Pulled high)	0	1	X	0	X						1-163
P92	O	Port output	1	X	X	0	X						1-163
			2,3F3h	Port P92 direction register									1-168
			2,3FEh	P90 to P93 pull-up control register									1-170
			7,39Dh	Event clock select bit ot timer B2 mode register									1-90
			3,362h	SI/O3 port select bit of SI/O3 control register									1-146
			2,362h	SOUT3 output disable bit of SI/O3 control register									1-146

Pin name	P91/TB1IN/SIN3		Pin No.	6 (FP)		4 (GP)							
Function selection			Setting register									Remarks	Page
			1,3F3h	2,3FEh	7,39Ch	3,362h							
TB1IN	I	Count source input	0	0	0	0							1-90
SIN3	I	Serial I/O data input	0	0	0	1							1-145
P91	I	Port input (No pulled high)	0	0	X	0							1-163
P91	I	Port input (Pulled high)	0	1	X	0							1-163
P91	O	Port output	1	X	X	0							1-163
			1,3F3h	Port P91 direction register									1-168
			2,3FEh	P90 to P93 pull-up control register									1-170
			7,39Ch	Event clock select bit of timer B1 mode register									1-90
			3,362h	SI/O3 port select bit of SI/O3 control register									1-146

Appendix 3 Setting register for pin function selected

Pin name	P90/TB0IN/CLK3		Pin No.	7 (FP)		5 (GP)									
Function selection			Setting register										Remarks	Page
			0,3F3h	2,3FEh	7,39Bh	3,362h	6,362h							
TB0IN	I	Count source input	0	0	0	0	X							1-90
CLK3	O	Serial clock output	0	0	X	1	1							1-145
CLK3	I	Serial clock input	0	0	X	1	0							1-145
P90	I	Port input (No pulled high)	0	0	X	0	X							1-163
P90	I	Port input (Pulled high)	0	1	X	0	X							1-163
P90	O	Port output	1	X	X	0	X							1-163
			0,3F3h	Port P90 direction register										1-168
			2,3FEh	P90 to P93 pull-up control register										1-170
			7,39Bh	Event clock select bit of timer B0 mode register										1-90
			3,362h	SI/O3 port select bit of SI/O3 control register										1-146
			6,362h	Synchronous clock select bit of SI/O3 control register										1-146

Pin name	P87/XCIN		Pin No.	10 (FP)		8 (GP)									
Function selection			Setting register										Remarks	Page
			7,3F2h	1,3FEh	4,006h									
XCIN	I	Sub clock input	0	0	1									1-36
P87	I	Port input (No pulled high)	0	0	0									1-163
P87	I	Port input (Pulled high)	0	1	0									1-163
P87	O	Port output	1	X	0									1-163
			7,3F2h	Port P87 direction register										1-168
			1,3FEh	P84 to P87 pull-up control register										1-170
			4,006h	Port Xc select bit of system clock control register 0										1-39

Pin name	P86/XCOUT		Pin No.	11 (FP)		9 (GP)									
Function selection			Setting register										Remarks	Page
			6,3F2h	1,3FEh	4,006h									
XCOUT	O	Sub clock output	0	0	1									1-36
P86	I	Port input (No pulled high)	0	0	0									1-163
P86	I	Port input (Pulled high)	0	1	0									1-163
P86	O	Port output	1	X	0									1-163
			6,3F2h	Port P86 direction register										1-168
			1,3FEh	P84 to P87 pull-up control register										1-170
			4,006h	Port Xc select bit of system clock control register 0										1-39

Appendix 3 Setting register for pin function selected

Pin name	P84/$\overline{INT2}$		Pin No.	18 (FP)	16 (GP)									
Function selection			Setting register									Remarks	Page
			4,3F2h	1,3FEh									
$\overline{INT2}$	I	Interrupt input	0	X								1	1-61
P84	I	Port input (No pulled high)	0	0									1-163
P84	I	Port input (Pulled high)	0	1									1-163
P84	O	Port output	1	X									1-163
			4,3F2h	Port P84 direction register									1-168
			1,3FEh	P84 to P87 pull-up control register									1-170

Remark 1: Interrupt request bit generates by state change of a port, not by setting value of register.

Pin name	P83/$\overline{INT1}$		Pin No.	19 (FP)	17 (GP)									
Function selection			Setting register									Remarks	Page
			4,3F2h	1,3FEh									
$\overline{INT1}$	I	Interrupt input	0	X								1	1-61
P83	I	Port input (No pulled high)	0	0									1-163
P83	I	Port input (Pulled high)	0	1									1-163
P83	O	Port output	1	X									1-163
			3,3F2h	Port P83 direction register									1-168
			0,3FEh	P80 to P83 pull-up control register									1-170

Remark 1: Interrupt request bit generates by state change of a port, not by setting value of register.

Pin name	P82/$\overline{INT0}$		Pin No.	20 (FP)	18 (GP)									
Function selection			Setting register									Remarks	Page
			2,3F2h	0,3FEh									
$\overline{INT0}$	I	Interrupt input	0	X								1	1-61
P82	I	Port input (No pulled high)	0	0									1-163
P82	I	Port input (Pulled high)	0	1									1-163
P82	O	Port output	1	X									1-163
			2,3F2h	Port P82 direction register									1-168
			0,3FEh	P80 to P83 pull-up control register									1-170

Remark 1: Interrupt request bit generates by state change of a port, not by setting value of register.

Pin name	P81/TA4IN/$\overline{U}$		Pin No.	21 (FP)	19 (GP)									
Function selection			Setting register									Remarks	Page
			1,3F2h	0,3FEh	7,383h	6,383h	4,39Ah	2,348h					
TA4IN	I	Gate function level input	0	X	0	0	1	0					1-78
TA4IN	I	Count source input	0	X	0	0	X	0					1-78
TA4IN	I	External trigger input	0	X	0	0	1	0					1-78
$\overline{U}$	O	$\overline{U}$ phase output	X	X	X	X	X	1					1-96
P81	I	Port input (No pulled high)	0	0	X	X	X	0					1-163
P81	I	Port input (Pulled high)	0	1	X	X	X	0					1-163
P81	O	Port output	1	X	X	X	X	0					1-163
			1,3F2h	Port P81 direction register									1-168
			0,3FEh	P80 to P83 pull-up control register									1-170
			7,383h	Timer A4 event/trigger select bit of trigger select register									1-82
			6,383h	Timer A4 event/trigger select bit of trigger select register									1-82
			4,39Ah	Bit 4 of timer A4 mode register									1-80
			2,348h	Three phase mode select bit									1-96

Appendix 3 Setting register for pin function selected

Pin name	P80/TA4OUT/U		Pin No.	22 (FP)		20 (GP)									
Function selection			Setting register										Remarks	Page
			0,3F2h	0,3FEh	4,39Ah	2,39Ah	7,384h	2,348h						
TA4OUT	O	Pulse output	X	X	X	1	0	0					1	1-78
TA4OUT	I	Up/down polarity select input	0	X	1	0	0	0						1-78
TA4OUT	I	Two-phase pulse signal input	0	X	1	0	1	0						1-78
U	O	U phase output	X	X	X	X	X	1						1-96
P80	I	Port input (No pulled high)	0	0	X	0	0	0						1-163
P80	I	Port input (Pulled high)	0	1	X	0	0	0						1-163
P80	O	Port output	1	X	X	0	0	0						1-163
			0,3F2h	Port P80 direction register										1-168
			0,3FEh	P80 to P83 pull-up control register										1-170
			4,39Ah	Bit 4 of timer A4 mode register										1-80
			2,39Ah	Bit 2 of timer A4 mode register										1-80
			7,384h	Timer A4 two-phase pulse signal processing select bit of up/down flag										1-81
			2,348h	Three phase mode select bit										1-96

Remark 1: Can not be use when processing two-phase pulse signal.

Pin name	P77/TA3IN		Pin No.	23 (FP)		21 (GP)									
Function selection			Setting register										Remarks	Page	
			7,3EFh	7,3FDh	5,383h	4,383h	4,399h								
TA3IN	I	Gate function level input	0	X	0	0	1							1-78	
TA3IN	I	Count source input	0	X	0	0	X							1-78	
TA3IN	I	External trigger input	0	X	0	0	1							1-78	
P77	I	Port input (No pulled high)	0	0	X	X	X							1-163	
P77	I	Port input (Pulled high)	0	1	X	X	X							1-163	
P77	O	Port output	1	X	X	X	X							1-163	
			7,3EFh	Port P77 direction register										1-168	
			7,3FDh	P74 to P77 pull-up control register										1-170	
			5,383h	Timer A3 event/trigger select bit of trigger select register										1-82	
			4,383h	Timer A3 event/trigger select bit of trigger select register										1-82	
			4,399h	Bit 4 of timer A3 mode register										1-80	

Pin name	P76/TA3OUT		Pin No.	24 (FP)		22 (GP)									
Function selection			Setting register										Remarks	Page	
			6,3EFh	7,3FDh	4,399h	2,399h	6,384h								
TA3OUT	O	Pulse output	X	X	X	1	0						1	1-78	
TA3OUT	I	Up/down polarity select input	0	X	1	0	0							1-78	
TA3OUT	I	Two-phase pulse signal input	0	X	1	0	1							1-78	
P76	I	Port input (No pulled high)	0	0	X	0	0							1-163	
P76	I	Port input (Pulled high)	0	1	X	0	0							1-163	
P76	O	Port output	1	X	X	0	0							1-163	
			6,3EFh	Port P76 direction register										1-168	
			7,3FDh	P74 to P77 pull-up control register										1-170	
			4,399h	Bit 4 of timer A3 mode register										1-80	
			2,399h	Bit 2 of timer A3 mode register										1-80	
			6,384h	Timer A3 two-phase pulse signal processing select bit of up/down flag										1-81	

Remark 1: Can not be use when processing two-phase pulse signal.

Appendix 3 Setting register for pin function selected

Pin name	P75/TA2IN/$\overline{W}$		Pin No.	25 (FP)		23 (GP)								
Function selection			Setting register									Remarks	Page
			5,3EFh	7,3FDh	3,383h	2,383h	4,398h	2,348h					
TA2IN	I	Gate function level input	0	X	0	0	1	0					1-78
TA2IN	I	Count source input	0	X	0	0	X	0					1-78
TA2IN	I	External trigger input	0	X	0	0	1	0					1-78
$\overline{W}$	O	$\overline{W}$ phase output	0	0	X	X	X	1					1-96
P75	I	Port input (No pulled high)	0	0	X	X	X	0					1-163
P75	I	Port input (Pulled high)	0	1	X	X	X	0					1-163
P75	O	Port output	1	X	X	X	X	0					1-163
			5,3EFh	Port P75 direction register									1-168
			7,3FDh	P74 to P77 pull-up control register									1-170
			3,383h	Timer A2 event/trigger select bit of trigger select register									1-82
			2,383h	Timer A2 event/trigger select bit of trigger select register									1-82
			4,398h	Bit 4 of timer A2 mode register									1-80
			2,348h	Three phase mode select bit									1-96

Pin name	P74/TA2OUT/W		Pin No.	26 (FP)		24 (GP)								
Function selection			Setting register									Remarks	Page
			4,3EFh	7,3FDh	4,398h	2,398h	5,384h	2,348h					
TA2OUT	O	Pulse output	X	X	X	1	0	0				1	1-78
TA2OUT	I	Up/down polarity select input	0	X	1	0	0	0					1-78
TA2OUT	I	Two-phase pulse signal input	0	X	1	0	1	0					1-78
W	O	W phase output	X	X	X	X	X	1					1-96
P74	I	Port input (No pulled high)	0	0	X	0	0	0					1-163
P74	I	Port input (Pulled high)	0	1	X	0	0	0					1-163
P74	O	Port output	1	X	X	0	0	0					1-163
			4,3EFh	Port P74 direction register									1-168
			7,3FDh	P74 to P77 pull-up control register									1-170
			4,398h	Bit 4 of timer A2 mode register									1-80
			2,398h	Bit 2 of timer A2 mode register									1-80
			5,384h	Timer A2 two-phase pulse signal processing select bit of up/down flag									1-81
			2,348h	Three phase mode select bit									1-96

Remark 1: Can not be use when processing two-phase pulse signal.

Appendix 3 Setting register for pin function selected

Pin name	P73/$\overline{\text{CTS2}}$/$\overline{\text{RTS2}}$/TA1IN/$\overline{\text{V}}$		Pin No.	27 (FP)		25 (GP)							
Function selection			Setting register								Remarks	Page
			3,3EFh	6,3FDh	1,383h	0,383h	4,397h	4,37Ch	2,37Ch	2,348h			
TA1IN	I	Gate function level input	0	X	0	0	1	X	X	0			1-78
TA1IN	I	Count source input	0	X	0	0	X	X	X	0			1-78
TA1IN	I	External trigger input	0	X	0	0	1	X	X	0			1-78
$\overline{\text{RTS2}}$	O	$\overline{\text{RTS}}$ output	X	X	X	X	X	0	1	0		1	1-108
$\overline{\text{CTS2}}$	I	$\overline{\text{CTS}}$ input	0	X	X	X	X	0	0	0		1	1-108
$\overline{\text{V}}$	O	$\overline{\text{V}}$ phase output	X	X	X	X	X	X	X	1			1-96
P73	I	Port input (No pulled high)	0	0	X	X	X	X	X	0			1-163
P73	I	Port input (Pulled high)	0	1	X	X	X	X	X	0			1-163
P73	O	Port output	1	X	X	X	X	X	X	0			1-163
			3,3EFh	Port P73 direction register									1-168
			6,3FDh	P70 to P73 pull-up control register									1-170
			1,383h	Bit 1 of trigger select register									1-82
			0,383h	Bit 0 of trigger select register									1-82
			4,397h	Bit 4 of timer A1 mode register									1-80
			4,37Ch	CTS/RTS disable bit of UART2 transmit/receive control register 0									1-114
			2,37Ch	CTS/RTS select bit of UART2 transmit/receive control register 0									1-114
			2,348h	Three phase mode select bit									1-96

Remark 1 : Set serial I/O enabled by serial I/O mode select bit of UART2 transmit/receive mode register.

Pin name	P72/CLK2/TA1OUT/V		Pin No.	28 (FP)		26 (GP)							
Function selection			Setting register								Remarks	Page
			2,3EFh	6,3FDh	4,397h	2,397h	3,378h	2,348h					
TA1OUT	O	Pulse output	X	X	X	1	X	0					1-78
TA1OUT	I	Up/down polarity select input	0	X	1	0	X	0					1-78
CLK2	I	Serial I/O clock input	0	0	X	X	1	0				1	1-108
CLK2	O	Serial I/O clock output	X	0	X	X	0	0				1,2	1-108
V	O	V phase output	X	X	X	X	X	1					1-96
P72	I	Port input (No pulled high)	0	0	X	0	X	1					1-163
P72	I	Port input (Pulled high)	0	1	X	0	X	0					1-163
P72	O	Port output	1	X	X	0	X	0					1-163
			2,3EFh	Port P72 direction register									1-168
			6,3FDh	P70 to P73 pull-up control register									1-170
			4,397h	Bit 4 of timer A1 mode register									1-80
			2,397h	Bit 2 of timer A1 mode register									1-80
			3,378h	Internal/external clock select bit of UART2 transmit/receive mode register									1-113
			2,348h	Three phase mode select bit									1-96

Remark 1: Set serial I/O enabled by serial I/O mode select bit of UART2 transmit/receive mode register.

Remark 2: It become I/O port when UART mode selected.

Appendix 3 Setting register for pin function selected

Pin name	P71/RXD2/TA0IN/TB5IN/SCL		Pin No.	29 (FP)		27 (GP)								
Function selection			Setting register									Remarks	Page
			1,3EFh	6,3FDh	7,382h	6,382h	4,396h	7,35Dh	0,377h	3,378h			
TA0IN	I	Gate function level input	0	X	0	0	1	0	0	X			1-78
TA0IN	I	Count source input	0	X	0	0	X	0	0	X			1-78
TA0IN	I	External trigger input	0	X	0	0	1	0	0	X			1-78
TB5IN	I	Count source input	0	0	X	X	X	1	X	0			1-90
SCL	O	IIC clock output	0	0	X	X	X	0	1	0		1	1-137
SCL	I	IIC clock input	0	0	X	X	X	0	1	1		1	1-137
RXD2	I	Serial I/O data input	0	0	X	X	X	0	0	X		1	1-108
P71	I	Port input	0	0	X	X	X	0	0	X			1-163
P71	O	Port output	1	X	X	X	X	0	0	X			1-163
			1,3EFh	Port P71 direction register									1-168
			6,3FDh	P70 to P73 pull-up control register									1-170
			7,382h	Bit 7 of one-shot start flag									1-82
			6,382h	Bit 6 of one-shot start flag									1-82
			4,396h	Bit 4 of timer A0 mode register									1-80
			7,35Dh	Event clock select bit of timer B5 mode register									1-90
			0,377h	IIC mode select bit									1-137
			3,378h	Internal /external clock select bit									1-127

Remark 1 : Set serial I/O enabled by serial I/O mode select bit of UART2 transmit/receive mode register.

Pin name	P70/TXD2/TA0OUT/SDA		Pin No.	30 (FP)		28 (GP)								
Function selection			Setting register									Remarks	Page
			0,3EFh	6,3FDh	4,396h	2,396h	0,377h	6,376h					
TA0OUT	O	Pulse output	X	X	X	1	0	X					1-78
TA0OUT	I	Up/down polarity select input	0	X	1	0	0	X					1-78
TXD2	O	Serial I/O data output	X	X	X	X	0	X				1, 2	1-108
SDA	I	IIC data input	0	0	0	0	1	1				1	1-137
SDA	O	IIC data output	0	0	0	0	1	0				1, 2	1-137
P70	I	Port input	0	0	0	0	0	X					1-163
P70	O	Port output	1	X	0	0	0	X					1-163
			0,3EFh	Port P70 direction register									1-168
			6,3FDh	P70 to P73 pull-up control register									1-170
			4,396h	Bit 4 of timer A0 mode register									1-80
			2,396h	Bit 2 of timer A0 mode register									1-80
			0,377h	IIC mode select bit									1-137
			6,376h	SDA output stop bit									1-141

Remark 1: Set serial I/O enabled by serial I/O mode select bit of UART2 transmit/receive mode register.

Remark 2: N channel open-drain output.

Appendix 3 Setting register for pin function selected

Pin name	P67/TXD1		Pin No.	31 (FP)	29 (GP)									
Function selection			Setting register										Remarks	Page
			7,3EEh	5,3FDh										
TXD1	O	Serial I/O data output	X	X									1, 2	1-108
P67	I	Port input (No pulled high)	0	0										1-163
P67	I	Port input (Pulled high)	0	1										1-163
P67	O	Port output	1	X										1-163
			7,3EEh	Port P67 direction register										1-168
			5,3FDh	P64 to P67 pull-up control register										1-170

Remark 1: Set serial I/O enabled by serial I/O mode select bit of UART1 transmit/receive mode register.

Remark 2: Can be selected CMOS output or N channel open-drain output.

Pin name	P66/RXD1		Pin No.	32 (FP)	30 (GP)									
Function selection			Setting register										Remarks	Page
			6,3EEh	5,3FDh										
RXD1	I	Serial I/O data input	0	X									1	1-108
P66	I	Port input (No pulled high)	0	0										1-163
P66	I	Port input (Pulled high)	0	1										1-163
P66	O	Port output	1	X										1-163
			6,3EEh	Port P66 direction register										1-168
			5,3FDh	P64 to P67 pull-up control register										1-170

Remark 1: Set serial I/O enabled by serial I/O mode select bit of UART1 transmit/receive mode register.

Pin name	P65/CLK1		Pin No.	33 (FP)	31 (GP)									
Function selection			Setting register										Remarks	Page
			5,3EEh	5,3FDh	3,3A8h									
CLK1	I	Serial I/O clock input	0	X	1								1	1-108
CLK1	O	Serial I/O clock output	X	X	0								1, 2	1-108
P65	I	Port input (No pulled high)	0	0	X									1-163
P65	I	Port input (Pulled high)	0	1	X									1-163
P65	O	Port output	1	X	X									1-163
			6,3EEh	Port P65 direction register										1-168
			5,3FDh	P64 to P67 pull-up control register										1-170
			3,3A8h	Internal/external clock select bit of UART1 transmit/receive mode register										1-113

Remark 1: Set serial I/O enabled by serial I/O mode select bit of UART1 transmit/receive mode register.

Remark 2: It become I/O port when UART mode selected.

Appendix 3 Setting register for pin function selected

Pin name	P64/$\overline{\text{CTS1}}$/$\overline{\text{RTS1}}$/CLKS1		Pin No.	34 (FP)		32 (GP)								
Function selection			Setting register										Remarks	Page
			4,3EEh	5,3FDh	5,3B0h	4,3B0h	4,3ACh	2,3ACh	3,3A8h					
CLKS1	O	Serial I/O clock output	X	X	1	1	1	X	0				1	1-108
$\overline{\text{RTS1}}$	O	$\overline{\text{RTS}}$ output	X	X	0	0	0	1	X				1	1-108
$\overline{\text{CTS1}}$	I	$\overline{\text{CTS}}$ input	0	X	0	0	0	0	X				1	1-108
P64	I	Port input (No pulled high)	0	0	X	X	X	X	X					1-163
P64	I	Port input (Pulled high)	0	1	X	X	X	X	X					1-163
P64	O	Port output	1	X	X	X	X	X	X					1-163
			4,3EEh	Port P64 direction register										1-168
			5,3FDh	P64 to P67 pull-up control register										1-170
			5,3B0h	CLK/CLKS select bit 1 of UART transmit/receive control register 2										1-116
			4,3B0h	CLK/CLKS select bit 0 of UART transmit/receive control register 2										1-116
			4,3ACh	CTS/RTS disable bit of UART1 transmit/receive control register 0										1-114
			2,3ACh	CTS/RTS function select bit of UART1 transmit/receive control register 0										1-114
			3,3A8h	Internal/external clock select bit of UART1 transmit/receive mode register										1-113

Remark 1: Set serial I/O enabled by serial I/O mode select bit of UART1 transmit/receive mode register.

Pin name	P63/TXD0		Pin No.	35 (FP)		33 (GP)								
Function selection			Setting register										Remarks	Page
			3,3EEh	4,3FDh										
TXD0	O	Serial I/O data output	X	X									1, 2	1-108
P63	I	Port input (No pulled high)	0	0										1-163
P63	I	Port input (Pulled high)	0	1										1-163
P63	O	Port output	1	X										1-163
			3,3EEh	Port P63 direction register										1-168
			4,3FDh	P60 to P63 pull-up control register										1-170

Remark 1: Set serial I/O enabled by serial I/O mode select bit of UART0 transmit/receive mode register.

Remark 2: Can be selected CMOS output or N channel open-drain output.

Pin name	P62/RXD0		Pin No.	36 (FP)		34 (GP)								
Function selection			Setting register										Remarks	Page
			2,3EEh	4,3FDh										
RXD0	I	Serial I/O data input	0	X									1	1-108
P62	I	Port input (No pulled high)	0	0										1-163
P62	I	Port input (Pulled high)	0	1										1-163
P62	O	Port output	1	X										1-163
			2,3EEh	Port P62 direction register										1-168
			4,3FDh	P60 to P63 pull-up control register										1-170

Remark 1: Set serial I/O enabled by serial I/O mode select bit of UART0 transmit/receive mode register.

Appendix 3 Setting register for pin function selected

Pin name	P61/CLK0		Pin No.	37 (FP)		35 (GP)								
Function selection			Setting register										Remarks	Page
			1,3EEh	4,3FDh	3,3A0h									
CLK0	I	Serial I/O clock input	0	X	1								1	1-108
CLK0	O	Serial I/O clock output	X	X	0								1, 2	1-108
P61	I	Port input (No pulled high)	0	0	X									1-163
P61	I	Port input (Pulled high)	0	1	X									1-163
P61	O	Port output	1	X	X									1-163
			1,3EEh	Port P61 direction register										1-168
			4,3FDh	P60 to P63 pull-up control register										1-170
			3,3A0h	Internal/external clock select bit of UART0 transmit/receive mode register										1-113

Remark 1: Set serial I/O enabled by serial I/O mode select bit of UART0 transmit/receive mode register.

Remark 2: It become I/O port when UART mode selected.

Pin name	P60/$\overline{\text{CTS0}}$/$\overline{\text{RTS0}}$		Pin No.	38 (FP)		36 (GP)								
Function selection			Setting register										Remarks	Page
			0,3EEh	4,3FDh	4,3A4h	2,3A4h								
$\overline{\text{RTS0}}$	O	$\overline{\text{RTS}}$ output	X	X	0	1							1	1-108
$\overline{\text{CTS0}}$	I	$\overline{\text{CTS}}$ input	0	X	0	0							1	1-108
P60	I	Port input (No pulled high)	0	0	X	X								1-163
P60	I	Port input (Pulled high)	0	1	X	X								1-163
P60	O	Port output	1	X	X	X								1-163
			0,3EEh	Port P60 direction register										1-168
			4,3FDh	P60 to P63 pull-up control register										1-170
			4,3A4h	CTS/RTS disable bit of UART0 transmit/receive control register 0										1-114
			2,3A4h	CTS/RTS function select bit of UART0 transmit/receive control register 0										1-114

Remark 1: Set serial I/O enabled by serial I/O mode select bit of UART0 transmit/receive mode register.

Pin name	P57/$\overline{\text{RDY}}$/CLKOUT		Pin No.	39 (FP)		37 (GP)								
Function selection			Setting register										Remarks	Page
			7,3EBh	3,3FDh	1,004h	0,004h	1,006h	0,006h						
CLKOUT	O	CLKOUT output	X	X	0	0	Except 00							1-40
$\overline{\text{RDY}}$	I	$\overline{\text{RDY}}$ input	X	X	X	1	X	X					1	1-32
P57	I	Port input (No pulled high)	0	0	0	0	0	0						1-163
P57	I	Port input (Pulled high)	0	1	0	0	0	0						1-163
P57	O	Port output	1	X	0	0	0	0						1-163
			7,3EBh	Port P57 direction register										1-168
			3,3FDh	P54 to P57 pull-up control register										1-170
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23
			1,006h	Clock output function select bit of system clock control register 0										1-39
			0,006h	Clock output function select bit of system clock control register 0										1-39

Remark 1: When the user is using the $\overline{\text{RDY}}$, the wait bit of relevant chip selects must be set to "0".

Appendix 3 Setting register for pin function selected

Pin name	P56/ALE		Pin No.	40(FP)		38 (GP)									
Function selection			Setting register											Remarks	Page
			6,3EBh	3,3FDh	1,004h	0,004h									
ALE	O	ALE output	X	X	X	1									1-31
P56	I	Port input (No pulled high)	0	0	0	0									1-163
P56	I	Port input (Pulled high)	0	1	0	0									1-163
P56	O	Port output	1	X	0	0									1-163
			6,3EBh	Port P56 direction register											1-168
			3,3FDh	P54 to P57 pull-up control register											1-170
			1,004h	Processor mode bit of processor mode register 0											1-23
			0,004h	Processor mode bit of processor mode register 0											1-23

Pin name	P55/$\overline{\text{HOLD}}$		Pin No.	41 (FP)		39 (GP)									
Function selection			Setting register											Remarks	Page
			5,3EBh	3,3FDh	1,004h	0,004h									
$\overline{\text{HOLD}}$	I	$\overline{\text{HOLD}}$ input	X	X	X	1									1-33
P55	I	Port input (No pulled high)	0	0	0	0									1-163
P55	I	Port input (Pulled high)	0	1	0	0									1-163
P55	O	Port output	1	X	0	0									1-163
			5,3EBh	Port P55 direction register											1-168
			3,3FDh	P54 to P57 pull-up control register											1-170
			1,004h	Processor mode bit of processor mode register 0											1-23
			0,004h	Processor mode bit of processor mode register 0											1-23

Pin name	P54/$\overline{\text{HLDA}}$		Pin No.	42 (FP)		40 (GP)									
Function selection			Setting register											Remarks	Page
			4,3EBh	3,3FDh	1,004h	0,004h									
$\overline{\text{HLDA}}$	O	$\overline{\text{HLDA}}$ output	X	X	X	1									1-28
P54	I	Port input (No pulled high)	0	0	0	0									1-163
P54	I	Port input (Pulled high)	0	1	0	0									1-163
P54	O	Port output	1	X	0	0									1-163
			4,3EBh	Port P54 direction register											1-168
			3,3FDh	P54 to P57 pull-up control register											1-170
			1,004h	Processor mode bit of processor mode register 0											1-23
			0,004h	Processor mode bit of processor mode register 0											1-23

Appendix 3 Setting register for pin function selected

Pin name	P53/BCLK		Pin No.	43 (FP)		41 (GP)								
Function selection			Setting register										Remarks	Page
			3,3EBh	2,3FDh	7,004h	1,004h	0,004h							
BCLK	O	BCLK output	X	X	0	X	1							1-28
P53	I	Port input (No pulled high)	0	0	X	0	0							1-163
P53	I	Port input (Pulled high)	0	1	X	0	0							1-163
P53	O	Port output	1	X	X	0	0							1-163
			3,3EBh	Port P53 direction register										1-168
			2,3FDh	P50 to P53 pull-up control register										1-170
			7,004h	BCLK output disable bit of processor mode register 0										1-23
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23

Pin name	P52/$\overline{RD}$		Pin No.	44 (FP)		42 (GP)								
Function selection			Setting register										Remarks	Page
			2,3EBh	2,3FDh	1,004h	0,004h								
$\overline{RD}$	O	$\overline{RD}$ output	X	X	X	1								1-28
P52	I	Port input (No pulled high)	0	0	0	0								1-163
P52	I	Port input (Pulled high)	0	1	0	0								1-163
P52	O	Port output	1	X	0	0								1-163
			2,3EBh	Port P52 direction register										1-168
			2,3FDh	P50 to P53 pull-up control register										1-170
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23

Pin name	P51/$\overline{WRH}$/$\overline{BHE}$		Pin No.	45 (FP)		43 (GP)								
Function selection			Setting register										Remarks	Page
			1,3EBh	2,3FDh	2,004h	1,004h	0,004h							
$\overline{BHE}$	O	$\overline{BHE}$ output	X	X	0	X	1							1-28
$\overline{WRH}$	O	$\overline{WRH}$ output	X	X	1	X	1							1-28
P51	I	Port input (No pulled high)	0	0	X	0	0							1-163
P51	I	Port input (Pulled high)	0	1	X	0	0							1-163
P51	O	Port output	1	X	X	0	0							1-163
			1,3EBh	Port P51 direction register										1-168
			2,3FDh	P50 to P53 pull-up control register										1-170
			2,004h	R/W mode select bit of processor mode register 0										1-23
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23

Pin name	P50/$\overline{WRL}$/$\overline{WR}$		Pin No.	46 (FP)		44 (GP)								
Function selection			Setting register										Remarks	Page
			0,3EBh	2,3FDh	2,004h	1,004h	0,004h							
$\overline{WR}$	O	$\overline{WR}$ output	X	X	0	X	1							1-28
$\overline{WRL}$	O	$\overline{WRL}$ output	X	X	1	X	1							1-28
P50	I	Port input (No pulled high)	0	0	X	0	0							1-163
P50	I	Port input (Pulled high)	0	1	X	0	0							1-163
P50	O	Port output	1	X	X	0	0							1-163
			0,3EBh	Port P50 direction register										1-168
			2,3FDh	P50 to P53 pull-up control register										1-170
			2,004h	R/W select bit of processor mode register 0										1-23
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23

Appendix 3 Setting register for pin function selected

Pin name	P47/$\overline{CS3}$		Pin No.	47 (FP)		45 (GP)									
Function selection			Setting register										Remarks	Page
			7,3EAh	1,3FDh	3,008h	1,004h	0,004h							
$\overline{CS3}$	O	Chip select output	X	X	1	X	1							1-28
P47	I	Port input (No pulled high)	0 0	0 0	X 0	0 X	0 1							1-163
P47	I	Port input (Pulled high)	0 0	1 1	X 0	0 X	0 1							1-163
P47	O	Port output	1 1	X X	X 0	0 X	0 1							1-163
			7,3EAh	Port P47 direction register										1-168
			1,3FDh	P44 to P47 pull-up control register										1-170
			3,008h	CS3 output enable bit of chip select control register										1-29
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23

Pin name	P46/$\overline{CS2}$		Pin No.	48 (FP)		46 (GP)									
Function selection			Setting register										Remarks	Page
			6,3EAh	1,3FDh	2,008h	1,004h	0,004h							
$\overline{CS2}$	O	Chip select output	X	X	1	X	1							1-28
P46	I	Port input (No pulled high)	0 0	0 0	X 0	0 X	0 1							1-163
P46	I	Port input (Pulled high)	0 0	1 1	X 0	0 X	0 1							1-163
P46	O	Port output	1 1	X X	X 0	0 X	0 1							1-163
			6,3EAh	Port P46 direction register										1-168
			1,3FDh	P44 to P47 pull-up control register										1-170
			2,008h	CS2 output enable bit of chip select control register										1-29
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23

Pin name	P45/$\overline{CS1}$		Pin No.	49 (FP)		47 (GP)									
Function selection			Setting register										Remarks	Page
			5,3EAh	1,3FDh	1,008h	1,004h	0,004h							
$\overline{CS1}$	O	Chip select output	X	X	1	X	1							1-28
P45	I	Port input (No pulled high)	0 0	0 0	X 0	0 X	0 1							1-163
P45	I	Port input (Pulled high)	0 0	1 0	X 0	0 X	0 1							1-163
P45	O	Port output	1 1	X X	X 0	0 X	0 1							1-163
			5,3EAh	Port P45 direction register										1-168
			1,3FDh	P44 to P47 pull-up control register										1-170
			1,008h	CS1 output enable bit of chip select control register										1-29
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23

Appendix 3 Setting register for pin function selected

Pin name	P44/$\overline{CS0}$		Pin No.	50 (FP)		48 (GP)								
Function selection			Setting register										Remarks	Page
			4,3EAh	1,3FDh	0,008h	1,004h	0,004h							
$\overline{CS0}$	O	Chip select output	X	X	1	X	1							1-28
P44	I	Port input (No pulled high)	0 0	0 0	X 0	0 X	0 1							1-163
P44	I	Port input (Pulled high)	0 0	1 0	X 0	0 X	0 1							1-163
P44	O	Port output	1 1	X X	X 0	0 X	0 1							1-163
			4,3EAh	Port P44 direction register										1-168
			1,3FDh	P44 to P47 pull-up control register										1-170
			0,008h	CS0 output enable bit of chip select control register										1-29
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23

Pin name	P43/A19		Pin No.	51 (FP)		49 (GP)								
Function selection			Setting register										Remarks	Page
			3,3EAh	0,3FDh	6,004h	5,004h	4,004h	1,004h	0,004h					
A19	O	Address output	X	X	0	Except 11		X	1					1-28
P43	I	Port input (No pulled high)	0 0 0	0 0 0	X 0 1	X 1 Except 11	X 1	0 0 X	0 1 1					1-163
P43	I	Port input (Pulled high)	0 0 0	1 1 1	X 0 1	X 1 Except 11	X 1	0 0 X	0 1 1					1-163
P43	O	Port output	1 1 1	X X X	X 0 1	X 1 Except 11	X 1	0 0 X	0 1 1					1-163
			3,3EAh	Port P43 direction register										1-168
			0,3FDh	P40 to P43 pull-up control register										1-170
			6,004h	Port P40 to P43 function select bit of processor mode register 0										1-23
			5,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			4,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23

Pin name	P42/A18		Pin No.	52 (FP)		50 (GP)								
Function selection			Setting register										Remarks	Page
			2,3EAh	0,3FDh	6,004h	5,004h	4,004h	1,004h	0,004h					
A18	O	Address output	X	X	0	Except 11		X	1					1-28
P42	I	Port input (No pulled high)	0 0 0	0 0 0	X 0 1	X 1 Except 11	X 1	0 0 X	0 1 1					1-163
P42	I	Port input (Pulled high)	0 0 0	1 1 1	X 0 1	X 1 Except 11	X 1	0 0 X	0 1 1					1-163
P42	O	Port output	1 1 1	X X X	X 0 1	X 1 Except 11	X 1	0 0 X	0 1 1					1-163
			2,3EAh	Port P42 direction register										1-168
			0,3FDh	P40 to P43 pull-up control register										1-170
			6,004h	Port P40 to P43 function select bit of processor mode register 0										1-23
			5,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			4,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23

Appendix 3 Setting register for pin function selected

Pin name	P41/A17		Pin No.	53 (FP)		51 (GP)									
Function selection			Setting register										Remarks	Page
			1,3EAh	0,3FDh	6,004h	5,004h	4,004h	1,004h	0,004h					
A17	O	Address output	X	X	0	Except 11		X	1					1-28
P41	I	Port input (No pulled high)	0 0 0	0 0 0	X 0 1	X 1 Except 11	X 1	0 0 X	0 1 1					1-163
P41	I	Port input (Pulled high)	0 0 0	1 1 1	X 0 1	X 1 Except 11	X 1	0 0 X	0 1 1					1-163
P41	O	Port output	1 1 1	X X X	X 0 1	X 1 Except 11	X 1	0 0 X	0 1 1					1-163
			1,3EAh	Port P41 direction register										1-168
			0,3FDh	P40 to P43 pull-up control register										1-170
			6,004h	Port P40 to P43 function select bit of processor mode register 0										1-23
			5,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			4,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23

Pin name	P40/A16		Pin No.	54 (FP)		52 (GP)									
Function selection			Setting register										Remarks	Page
			0,3EAh	0,3FDh	6,004h	5,004h	4,004h	1,004h	0,004h					
A16	O	Address output	X	X	0	Except 11		X	1					1-28
P40	I	Port input (No pulled high)	0 0 0	0 0 0	X 0 1	X 1 Except 11	X 1	0 0 X	0 1 1					1-163
P40	I	Port input (Pulled high)	0 0 0	1 1 1	X 0 1	X 1 Except 11	X 1	0 0 X	0 1 1					1-163
P40	O	Port output	1 1 1	X X X	X 0 1	X 1 Except 11	X 1	0 0 X	0 1 1					1-163
			0,3EAh	Port P40 direction register										1-168
			0,3FDh	P40 to P43 pull-up control register										1-170
			6,004h	Port P40 to P43 function select bit of processor mode register 0										1-23
			5,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			4,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23

Pin name	P37/A15		Pin No.	55 (FP)		53 (GP)									
Function selection			Setting register										Remarks	Page
			7,3E7h	7,3FCh	5,004h	4,004h	1,004h	0,004h						
A15	O	Address output	X	X	Except 11		X	1						1-28
P37	I	Port input (No pulled high)	0 0	0 0	X 1	X 1	0 0	0 1						1-163
P37	I	Port input (Pulled high)	0 0	1 1	0 1	0 1	0 0	0 1						1-163
P37	O	Port output	1 1	X X	0 0	0 0	0 0	0 1						1-163
			7,3E7h	Port P37 direction register										1-168
			7,3FCh	P34 to P37 pull-up control register										1-170
			5,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			4,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23

Appendix 3 Setting register for pin function selected

Pin name	P36/A14		Pin No.	56 (FP)		54 (GP)								
Function selection			Setting register									Remarks	Page
			6,3E7h	7,3FCh	5,004h	4,004h	1,004h	0,004h					
A14	O	Address output	X	X	Except 11		X	1					1-28
P36	I	Port input (No pulled high)	0 0	0 0	X 1	X 1	0 0	0 1					1-163
P36	I	Port input (Pulled high)	0 0	1 1	0 1	0 1	0 0	0 1					1-163
P36	O	Port output	1 1	X X	0 0	0 0	0 0	0 1					1-163
			6,3E7h	Port P36 direction register									1-168
			7,3FCh	P34 to P37 pull-up control register									1-170
			5,004h	Multiplexed bus space select bit of processor mode register 0									1-23
			4,004h	Multiplexed bus space select bit of processor mode register 0									1-23
			1,004h	Processor mode bit of processor mode register 0									1-23
			0,004h	Processor mode bit of processor mode register 0									1-23

Pin name	P35/A13		Pin No.	57 (FP)		55 (GP)								
Function selection			Setting register									Remarks	Page
			5,3E7h	7,3FCh	5,004h	4,004h	1,004h	0,004h					
A13	O	Address output	X	X	Except 11		X	1					1-28
P35	I	Port input (No pulled high)	0 0	0 0	X 1	X 1	0 0	0 1					1-163
P35	I	Port input (Pulled high)	0 0	1 1	0 1	0 1	0 0	0 1					1-163
P35	O	Port output	1 1	X X	0 0	0 0	0 0	0 1					1-163
			5,3E7h	Port P35 direction register									1-168
			7,3FCh	P34 to P37 pull-up control register									1-170
			5,004h	Multiplexed bus space select bit of processor mode register 0									1-23
			4,004h	Multiplexed bus space select bit of processor mode register 0									1-23
			1,004h	Processor mode bit of processor mode register 0									1-23
			0,004h	Processor mode bit of processor mode register 0									1-23

Pin name	P34/A12		Pin No.	58 (FP)		56 (GP)								
Function selection			Setting register									Remarks	Page
			4,3E7h	7,3FCh	5,004h	4,004h	1,004h	0,004h					
A12	O	Address output	X	X	Except 11		X	1					1-28
P34	I	Port input (No pulled high)	0 0	0 0	X 1	X 1	0 0	0 1					1-163
P34	I	Port input (Pulled high)	0 0	1 1	0 1	0 1	0 0	0 1					1-163
P34	O	Port output	1 1	X X	0 0	0 0	0 0	0 1					1-163
			4,3E7h	Port P34 direction register									1-168
			7,3FCh	P34 to P37 pull-up control register									1-170
			5,004h	Multiplexed bus space select bit of processor mode register 0									1-23
			4,004h	Multiplexed bus space select bit of processor mode register 0									1-23
			1,004h	Processor mode bit of processor mode register 0									1-23
			0,004h	Processor mode bit of processor mode register 0									1-23

Appendix 3 Setting register for pin function selected

Pin name	P33/A11		Pin No.	59 (FP)		57 (GP)								
Function selection			Setting register										Remarks	Page
			3,3E7h	6,3FCh	5,004h	4,004h	1,004h	0,004h						
A11	O	Address output	X	X	Except 11		X	1						1-28
P33	I	Port input (No pulled high)	0 0	0 0	X 1	X 1	0 0	0 1						1-163
P33	I	Port input (Pulled high)	0 0	1 1	0 1	0 1	0 0	0 1						1-163
P33	O	Port output	1 1	X X	0 0	0 0	0 0	0 1						1-163
			3,3E7h	Port P33 direction register										1-168
			6,3FCh	P30 to P33 pull-up control register										1-170
			5,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			4,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23

Pin name	P32/A10		Pin No.	60 (FP)		58 (GP)								
Function selection			Setting register										Remarks	Page
			2,3E7h	6,3FCh	5,004h	4,004h	1,004h	0,004h						
A10	O	Address output	X	X	Except 11		X	1						1-28
P32	I	Port input (No pulled high)	0 0	0 0	X 1	X 1	0 0	0 1						1-163
P32	I	Port input (Pulled high)	0 0	1 1	0 1	0 1	0 0	0 1						1-163
P32	O	Port output	1 1	X X	0 0	0 0	0 0	0 1						1-163
			2,3E7h	Port P32 direction register										1-168
			6,3FCh	P30 to P33 pull-up control register										1-170
			5,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			4,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23

Pin name	P31/A9		Pin No.	61 (FP)		59 (GP)								
Function selection			Setting register										Remarks	Page
			1,3E7h	6,3FCh	5,004h	4,004h	1,004h	0,004h						
A9	O	Address output	X	X	Except 11		X	1						1-28
P31	I	Port input (No pulled high)	0 0	0 0	X 1	X 1	0 0	0 1						1-163
P31	I	Port input (Pulled high)	0 0	1 1	X 1	X 1	0 0	0 1						1-163
P31	O	Port output	1 1	X X	X 1	X 1	0 0	0 1						1-163
			1,3E7h	Port P31 direction register										1-168
			6,3FCh	P30 to P33 pull-up control register										1-170
			5,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			4,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23

Appendix 3 Setting register for pin function selected

Pin name	P30/A8(/-/D7)		Pin No.	63 (FP)		61 (GP)									
Function selection			Setting register										Remarks	Page
			0,3E7h	6,3FCh	5,004h	4,004h	1,004h	0,004h	BYTE				
A8/D7	I/O	Multiplexed bus	X X	X X	0 1	1 0	X X	1 1	L L					1-28
A8/-	O	Multiplexed bus	X X	X X	X 1	1 X	X X	1 1	H H					1-28
A8	O	Address output	X	X	0	0	X	1	X					1-28
P30	I	Port input (No pulled high)	0	0	X	X	0	0	X					1-163
P30	I	Port input (Pulled high)	0	1	X	X	0	0	X					1-163
P30	O	Port output	1	X	X	X	0	0	X					1-163
			0,3E7h	Port P30 direction register										1-168
			6,3FCh	P30 to P33 pull-up control register										1-170
			5,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			4,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23
			BYTE	BYTE pin										1-26

Pin name	P27/A7(/D7/D6)		Pin No.	65 (FP)		63 (GP)									
Function selection			Setting register										Remarks	Page
			7,3E6h	5,3FCh	5,004h	4,004h	1,004h	0,004h	BYTE				
A7/D6	I/O	Multiplexed bus	X X	X X	0 1	1 0	X X	1 1	L L					1-28
A7/D7	I/O	Multiplexed bus	X X	X X	X 1	1 X	X X	1 1	H H					1-28
A7	O	Address output	X	X	0	0	X	1	X					1-28
P27	I	Port input (No pulled high)	0	0	X	X	0	0	X					1-163
P27	I	Port input (Pulled high)	0	1	X	X	0	0	X					1-163
P27	O	Port output	1	X	X	X	0	0	X					1-163
			7,3E6h	Port P27 direction register										1-168
			5,3FCh	P24 to P27 pull-up control register										1-170
			5,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			4,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23
			BYTE	BYTE pin										1-26

Pin name	P26/A6(/D6/D5)		Pin No.	66 (FP)		64 (GP)									
Function selection			Setting register										Remarks	Page
			6,3E6h	5,3FCh	5,004h	4,004h	1,004h	0,004h	BYTE				
A6/D5	I/O	Multiplexed bus	X X	X X	0 1	1 0	X X	1 1	L L					1-28
A6/D6	I/O	Multiplexed bus	X X	X X	X 1	1 X	X X	1 1	H H					1-28
A6	O	Address output	X	X	0	0	X	1	X					1-28
P26	I	Port input (No pulled high)	0	0	X	X	0	0	X					1-163
P26	I	Port input (Pulled high)	0	1	X	X	0	0	X					1-163
P26	O	Port output	1	X	X	X	0	0	X					1-163
			6,3E6h	Port P26 direction register										1-168
			5,3FCh	P24 to P27 pull-up control register										1-170
			5,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			4,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23
			BYTE	BYTE pin										1-26

Appendix 3 Setting register for pin function selected

Pin name	P25/A5(/D5/D4)		Pin No.	67 (FP)		65 (GP)									
Function selection			Setting register										Remarks	Page
			5,3E6h	5,3FCh	5,004h	4,004h	1,004h	0,004h	BYTE				
A5/D4	I/O	Multiplexed bus	X X	X X	0 1	1 0	X X	1 1	L L					1-28
A5/D5	I/O	Multiplexed bus	X X	X X	X 1	1 X	X X	1 1	H H					1-28
A5	O	Address output	X	X	0	0	X	1	X					1-28
P25	I	Port input (No pulled high)	0	0	X	X	0	0	X					1-163
P25	I	Port input (Pulled high)	0	1	X	X	0	0	X					1-163
P25	O	Port output	1	X	X	X	0	0	X					1-163
			5,3E6h	Port P25 direction register										1-168
			5,3FCh	P24 to P27 pull-up control register										1-170
			5,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			4,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23
			BYTE	BYTE pin										1-26

Pin name	P24/A4(/D4/D3)		Pin No.	68 (FP)		66 (GP)									
Function selection			Setting register										Remarks	Page
			4,3E6h	5,3FCh	5,004h	4,004h	1,004h	0,004h	BYTE				
A4/D3	I/O	Multiplexed bus	X X	X X	0 1	1 0	X X	1 1	L L					1-28
A4/D4	I/O	Multiplexed bus	X X	X X	X 1	1 X	X X	1 1	H H					1-28
A4	O	Address output	X	X	0	0	X	1	X					1-28
P24	I	Port input (No pulled high)	0	0	X	X	0	0	X					1-163
P24	I	Port input (Pulled high)	0	1	X	X	0	0	X					1-163
P24	O	Port output	1	X	X	X	0	0	X					1-163
			4,3E6h	Port P24 direction register										1-168
			5,3FCh	P24 to P27 pull-up control register										1-170
			5,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			4,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23
			BYTE	BYTE pin										1-26

Pin name	P23/A3(/D3/D2)		Pin No.	69 (FP)		67 (GP)									
Function selection			Setting register										Remarks	Page
			3,3E6h	4,3FCh	5,004h	4,004h	1,004h	0,004h	BYTE				
A3/D2	I/O	Multiplexed bus	X X	X X	0 1	1 0	X X	1 1	L L					1-28
A3/D3	I/O	Multiplexed bus	X X	X X	X 1	1 X	X X	1 1	H H					1-28
A3	O	Address output	X	X	0	0	X	1	X					1-28
P23	I	Port input (No pulled high)	0	0	X	X	0	0	X					1-163
P23	I	Port input (Pulled high)	0	1	X	X	0	0	X					1-163
P23	O	Port output	1	X	X	X	0	0	X					1-163
			3,3E6h	Port P23 direction register										1-168
			4,3FCh	P20 to P23 pull-up control register										1-170
			5,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			4,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23
			BYTE	BYTE pin										1-26

Appendix 3 Setting register for pin function selected

Pin name	P22/A2(/D2/D1)		Pin No.	70 (FP)		68 (GP)									
Function selection			Setting register										Remarks	Page
			2,3E6h	4,3FCh	5,004h	4,004h	1,004h	0,004h	BYTE					
A2/D1	I/O	Multiplexed bus	X X	X X	0 1	1 0	X X	1 1	L L					1-28
A2/D2	I/O	Multiplexed bus	X X	X X	X 1	1 X	X X	1 1	H H					1-28
A2	O	Address output	X	X	0	0	X	1	X					1-28
P22	I	Port input (No pulled high)	0	0	X	X	0	0	X					1-163
P22	I	Port input (Pulled high)	0	1	X	X	0	0	X					1-163
P22	O	Port output	1	X	X	X	0	0	X					1-163
			2,3E6h	Port P22 direction register										1-168
			4,3FCh	P20 to P23 pull-up control register										1-170
			5,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			4,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23
			BYTE	BYTE pin										1-26

Pin name	P21/A1(/D1/D0)		Pin No.	71 (FP)		69 (GP)									
Function selection			Setting register										Remarks	Page
			1,3E6h	4,3FCh	5,004h	4,004h	1,004h	0,004h	BYTE					
A1/D0	I/O	Multiplexed bus	X X	X X	0 1	1 0	X X	1 1	L L					1-28
A1/D1	I/O	Multiplexed bus	X X	X X	X 1	1 X	X X	1 1	H H					1-28
A1	O	Address output	X	X	0	0	X	1	X					1-28
P21	I	Port input (No pulled high)	0	0	X	X	0	0	X					1-163
P21	I	Port input (Pulled high)	0	1	X	X	0	0	X					1-163
P21	O	Port output	1	X	X	X	0	0	X					1-163
			1,3E6h	Port P21 direction register										1-168
			4,3FCh	P20 to P23 pull-up control register										1-170
			5,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			4,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23
			BYTE	BYTE pin										1-26

Pin name	P20/A0(/D0/-)		Pin No.	72 (FP)		70 (GP)									
Function selection			Setting register										Remarks	Page
			0,3E6h	4,3FCh	5,004h	4,004h	1,004h	0,004h	BYTE					
A0/-	I/O	Multiplexed bus	X X	X X	0 1	1 0	X X	1 1	L L					1-28
A0/D0	I/O	Multiplexed bus	X X	X X	X 1	1 X	X X	1 1	H H					1-28
A0	O	Address output	X	X	0	0	X	1	X					1-28
P20	I	Port input (No pulled high)	0	0	X	X	0	0	X					1-163
P20	I	Port input (Pulled high)	0	1	X	X	0	0	X					1-163
P20	O	Port output	1	X	X	X	0	0	X					1-163
			0,3E6h	Port P20 direction register										1-168
			4,3FCh	P20 to P23 pull-up control register										1-170
			5,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			4,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23
			BYTE	BYTE pin										1-26

Appendix 3 Setting register for pin function selected

Pin name	P17/D15/$\overline{INT5}$		Pin No.	73 (FP)		71 (GP)									
Function selection			Setting register										Remarks	Page
			7,3E3h	3,3FCh	5,004h	4,004h	1,004h	0,004h	BYTE					
D15	I/O	Data bus	X	X	Except 11		X	1	L					1-28
$\overline{INT5}$	I	Interrupt input	0 0	X X	X X	X X	0 X	0 1	X H					1-61
P17	I	Port input (No pulled high)	0 0	0 0	X X	X X	0 X	0 1	X H					1-163
P17	I	Port input (Pulled high)	0 0	1 1	X X	X X	0 X	0 1	X H					1-163
P17	O	Port output	1 1	X X	X X	X X	0 X	0 1	X H					1-163
			7,3E3h	Port P17 direction register										1-168
			3,3FCh	P14 to P17 pull-up control register										1-170
			5,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			4,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23
			BYTE	BYTE pin										1-26

Pin name	P16/D14/$\overline{INT4}$		Pin No.	74 (FP)		72 (GP)									
Function selection			Setting register										Remarks	Page
			6,3E3h	3,3FCh	5,004h	4,004h	1,004h	0,004h	BYTE					
D14	I/O	Data bus	X	X	Except 11		X	1	L					1-28
$\overline{INT4}$	I	Interrupt input	0 0	X X	X X	X X	0 X	0 1	X H					1-61
P16	I	Port input (No pulled high)	0 0	0 0	X X	X X	0 X	0 1	X H					1-163
P16	I	Port input (Pulled high)	0 0	1 1	X X	X X	0 X	0 1	X H					1-163
P16	O	Port output	1 1	X X	X X	X X	0 X	0 1	X H					1-163
			6,3E3h	Port P16 direction register										1-168
			3,3FCh	P14 to P17 pull-up control register										1-170
			5,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			4,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23
			BYTE	BYTE pin										1-26

Pin name	P15/D13/$\overline{INT3}$		Pin No.	75 (FP)		73 (GP)									
Function selection			Setting register										Remarks	Page
			5,3E3h	3,3FCh	5,004h	4,004h	1,004h	0,004h	BYTE					
D13	I/O	Data bus	X	X	Except 11		X	1	L					1-28
$\overline{INT3}$	I	Interrupt input	0 0	X X	X X	X X	0 X	0 1	X H					1-61
P15	I	Port input (No pulled high)	0 0	0 0	X X	X X	0 X	0 1	X H					1-163
P15	I	Port input (Pulled high)	0 0	1 1	X X	X X	0 X	0 1	X H					1-163
P15	O	Port output	1 1	X X	X X	X X	0 X	0 1	X H					1-163
			5,3E3h	Port P15 direction register										1-168
			3,3FCh	P14 to P17 pull-up control register										1-170
			5,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			4,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23
			BYTE	BYTE pin										1-26

Appendix 3 Setting register for pin function selected

Pin name	P14/D12		Pin No.	76 (FP)		74 (GP)									
Function selection			Setting register										Remarks	Page
			4,3E3h	3,3FCh	5,004h	4,004h	1,004h	0,004h	BYTE					
D12	I/O	Data bus	X	X	Except 11		X	1	L					1-28
P14	I	Port input (No pulled high)	0 0	0 0	X X	X X	0 X	0 1	X H					1-163
P14	I	Port input (Pulled high)	0 0	1 1	X X	X X	0 X	0 1	X H					1-163
P14	O	Port output	1 1	X X	X X	X X	0 X	0 1	X H					1-163
			4,3E3h	Port P14 direction register										1-168
			3,3FCh	P14 to P17 pull-up control register										1-170
			5,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			4,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23
			BYTE	BYTE pin										1-26

Pin name	P13/D11		Pin No.	77 (FP)		75 (GP)									
Function selection			Setting register										Remarks	Page
			3,3E3h	2,3FCh	5,004h	4,004h	1,004h	0,004h	BYTE					
D11	I/O	Data bus	X	X	Except 11		X	1	L					1-28
P13	I	Port input (No pulled high)	0 0	0 0	X X	X X	0 X	0 1	X H					1-163
P13	I	Port input (Pulled high)	0 0	1 1	X X	X X	0 X	0 1	X H					1-163
P13	O	Port output	1 1	X X	X X	X X	0 X	0 1	X H					1-163
			3,3E3h	Port P13 direction register										1-168
			2,3FCh	P10 to P13 pull-up control register										1-170
			5,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			4,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23
			BYTE	BYTE pin										1-26

Pin name	P12/D10		Pin No.	78 (FP)		76 (GP)									
Function selection			Setting register										Remarks	Page
			2,3E3h	2,3FCh	5,004h	4,004h	1,004h	0,004h	BYTE					
D10	I/O	Data bus	X	X	Except 11		X	1	L					1-28
P12	I	Port input (No pulled high)	0 0	0 0	X X	X X	0 X	0 1	X H					1-163
P12	I	Port input (Pulled high)	0 0	1 1	X X	X X	0 X	0 1	X H					1-163
P12	O	Port output	1 1	X X	X X	X X	0 X	0 1	X H					1-163
			2,3E3h	Port P12 direction register										1-168
			2,3FCh	P10 to P13 pull-up control register										1-170
			5,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			4,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23
			BYTE	BYTE pin										1-26

Appendix 3 Setting register for pin function selected

Pin name	P11/D9		Pin No.	79 (FP)		77 (GP)								
Function selection			Setting register										Remarks	Page
			1,3E3h	2,3FCh	5,004h	4,004h	1,004h	0,004h	BYTE				
D9	I/O	Data bus	X	X	Except 11		X	1	L					1-28
P11	I	Port input (No pulled high)	0 0	0 0	X X	X X	0 X	0 1	X H					1-163
P11	I	Port input (Pulled high)	0 0	1 1	X X	X X	0 X	0 1	X H					1-163
P11	O	Port output	1 1	X X	X X	X X	0 X	0 1	X H					1-163
			1,3E3h	Port P11 direction register										1-168
			2,3FCh	P10 to P13 pull-up control register										1-170
			5,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			4,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23
			BYTE	BYTE pin										1-26

Pin name	P10/D8		Pin No.	80 (FP)		78 (GP)								
Function selection			Setting register										Remarks	Page
			0,3E3h	2,3FCh	5,004h	4,004h	1,004h	0,004h	BYTE				
D8	I/O	Data bus	X	X	Except 11		X	1	L					1-28
P10	I	Port input (No pulled high)	0 0	0 0	X X	X X	0 X	0 1	X H					1-163
P10	I	Port input (Pulled high)	0 0	1 1	X X	X X	0 X	0 1	X H					1-163
P10	O	Port output	1 1	X X	X X	X X	0 X	0 1	X H					1-163
			0,3E3h	Port P10 direction register										1-168
			2,3FCh	P10 to P13 pull-up control register										1-170
			5,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			4,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23
			BYTE	BYTE pin										1-26

Pin name	P07/D7		Pin No.	81 (FP)		79 (GP)								
Function selection			Setting register										Remarks	Page
			7,3E2h	1,3FCh	5,004h	4,004h	1,004h	0,004h	BYTE				
D7	I/O	Data bus	X	X	Except 11		X	1	X					1-28
P07	I	Port input (No pulled high)	0 0	0 0	X 1	X 1	X 0	0 1	X H					1-163
P07	I	Port input (Pulled high)	0 0	1 1	X 1	X 1	0 0	0 1	X H					1-163
P07	O	Port output	1 1	X X	X 1	X 1	0 0	0 1	X H					1-163
			7,3E2h	Port P07 direction register										1-168
			1,3FCh	P04 to P07 pull-up control register										1-170
			5,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			4,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23
			BYTE	BYTE pin										1-26

Appendix 3 Setting register for pin function selected

Pin name	P06/D6		Pin No.	82 (FP)		80 (GP)									
Function selection			Setting register										Remarks	Page
			6,3E2h	1,3FCh	5,004h	4,004h	1,004h	0,004h	BYTE				
D6	I/O	Data bus	X	X	Except 11		X	1	X					1-28
P06	I	Port input (No pulled high)	0 0	0 0	X 1	X 1	0 0	0 1	X H					1-163
P06	I	Port input (Pulled high)	0 0	1 1	X 1	X 1	0 0	0 1	X H					1-163
P06	O	Port output	1 1	X X	X 1	X 1	0 0	0 1	X H					1-163
			6,3E2h	Port P06 direction register										1-168
			1,3FCh	P04 to P07 pull-up control register										1-170
			5,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			4,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23
			BYTE	BYTE pin										1-26

Pin name	P05/D5		Pin No.	83 (FP)		81 (GP)									
Function selection			Setting register										Remarks	Page
			5,3E2h	1,3FCh	5,004h	4,004h	1,004h	0,004h	BYTE				
D5	I/O	Data bus	X	X	Except 11		X	1	X					1-28
P05	I	Port input (No pulled high)	0 0	0 0	X 1	X 1	0 0	0 1	X H					1-163
P05	I	Port input (Pulled high)	0 0	1 1	X 1	X 1	0 0	0 1	X H					1-163
P05	O	Port output	1 1	X X	X 1	X 1	0 0	0 1	X H					1-163
			5,3E2h	Port P05 direction register										1-168
			1,3FCh	P04 to P07 pull-up control register										1-170
			5,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			4,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23
			BYTE	BYTE pin										1-26

Pin name	P04/D4		Pin No.	84 (FP)		82 (GP)									
Function selection			Setting register										Remarks	Page
			4,3E2h	1,3FCh	5,004h	4,004h	1,004h	0,004h	BYTE				
D4	I/O	Data bus	X	X	Except 11		X	1	X					1-28
P04	I	Port input (No pulled high)	0 0	0 0	X 1	X 1	0 0	0 1	X H					1-163
P04	I	Port input (Pulled high)	0 0	1 1	X 1	X 1	0 0	0 1	X H					1-163
P04	O	Port output	1 1	X X	X 1	X 1	0 0	0 1	X H					1-163
			4,3E2h	Port P04 direction register										1-168
			1,3FCh	P04 to P07 pull-up control register										1-170
			5,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			4,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23
			BYTE	BYTE pin										1-26

Appendix 3 Setting register for pin function selected

Pin name	P03/D3		Pin No.	85 (FP)		83 (GP)									
Function selection			Setting register										Remarks	Page
			3,3E2h	0,3FCh	5,004h	4,004h	1,004h	0,004h	BYTE				
D3	I/O	Data bus	X	X	Except 11		X	1	X					1-28
P03	I	Port input (No pulled high)	0 0	0 0	X 1	X 1	0 0	0 1	X H					1-163
P03	I	Port input (Pulled high)	0 0	1 1	X 1	X 1	0 0	0 1	X H					1-163
P03	O	Port output	1 1	X X	X 1	X 1	0 0	0 1	X H					1-167
			3,3E2h	Port P03 direction register										1-168
			0,3FCh	P00 to P03 pull-up control register										1-170
			5,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			4,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23
			BYTE	BYTE pin										1-26

Pin name	P02/D2		Pin No.	86 (FP)		84 (GP)									
Function selection			Setting register										Remarks	Page
			2,3E2h	0,3FCh	5,004h	4,004h	1,004h	0,004h	BYTE				
D2	I/O	Data bus	X	X	Except 11		X	1	X					1-28
P02	I	Port input (No pulled high)	0 0	0 0	X 1	X 1	0 0	0 1	X H					1-163
P02	I	Port input (Pulled high)	0 0	1 1	X 1	X 1	0 0	0 1	X H					1-163
P02	O	Port output	1 1	X X	X 1	X 1	0 0	0 1	X H					1-163
			2,3E2h	Port P02 direction register										1-168
			0,3FCh	P00 to P03 pull-up control register										1-170
			5,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			4,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23
			BYTE	BYTE pin										1-26

Pin name	P01/D1		Pin No.	87 (FP)		85 (GP)									
Function selection			Setting register										Remarks	Page
			1,3E2h	0,3FCh	5,004h	4,004h	1,004h	0,004h	BYTE				
D1	I/O	Data bus	X	X	Except 11		X	1	X					1-28
P01	I	Port input (No pulled high)	0 0	0 0	X 1	X 1	0 0	0 1	X H					1-163
P01	I	Port input (Pulled high)	0 0	1 1	X 1	X 1	0 0	0 1	X H					1-163
P01	O	Port output	1 1	X X	X 1	X 1	0 0	0 1	X H					1-163
			1,3E2h	Port P01 direction register										1-168
			0,3FCh	P00 to P03 pull-up control register										1-170
			5,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			4,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23
			BYTE	BYTE pin										1-26

Appendix 3 Setting register for pin function selected

Pin name	P00/D0		Pin No.	88 (FP)		86 (GP)									
Function selection			Setting register										Remarks	Page
			0,3E2h	0,3FCh	5,004h	4,004h	1,004h	0,004h	BYTE					
D0	I/O	Data bus	X	X	Except 11		X	1	X					1-28
P00	I	Port input (No pulled high)	0 0	0 0	X 1	X 1	0 0	0 1	X H					1-163
P00	I	Port input (Pulled high)	0 0	1 1	X 1	X 1	0 0	0 1	X H					1-163
P00	O	Port output	1 1	X X	X 1	X 1	0 0	0 1	X H					1-163
			0,3E2h	Port P00 direction register										1-168
			0,3FCh	P00 to P03 pull-up control register										1-170
			5,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			4,004h	Multiplexed bus space select bit of processor mode register 0										1-23
			1,004h	Processor mode bit of processor mode register 0										1-23
			0,004h	Processor mode bit of processor mode register 0										1-23
			BYTE	BYTE pin										1-26

Pin name	P107/AN7/$\overline{KI3}$		Pin No.	89 (FP)		87 (GP)									
Function selection			Setting register										Remarks	Page
			7,3F6h	5,3FEh										
$\overline{KI3}$	I	Key input	0	X										1-62
AN7	I	A-D input	0	0										1-149
P107	I	Port input (No pulled high)	0	0										1-163
P107	I	Port input (Pulled high)	0	1										1-163
P107	O	Port output	1	X										1-163
			7,3F6h	Port P107 direction register										1-168
			5,3FEh	P104 to P107 pull-up control register										1-170

Pin name	P106/AN6/$\overline{KI2}$		Pin No.	90 (FP)		88 (GP)									
Function selection			Setting register										Remarks	Page
			6,3F6h	5,3FEh										
$\overline{KI2}$	I	Key input	0	X										1-62
AN6	I	A-D input	0	0										1-149
P106	I	Port input (No pulled high)	0	0										1-163
P106	I	Port input (Pulled high)	0	1										1-163
P106	O	Port output	1	X										1-163
			6,3F6h	Port P106 direction register										1-168
			5,3FEh	P104 to P107 pull-up control register										1-170

Pin name	P105/AN5/$\overline{KI1}$		Pin No.	91 (FP)		89 (GP)									
Function selection			Setting register										Remarks	Page
			5,3F6h	5,3FEh										
$\overline{KI1}$	I	Key input	0	X										1-62
AN5	I	A-D input	0	0										1-149
P105	I	Port input (No pulled high)	0	0										1-163
P105	I	Port input (Pulled high)	0	1										1-163
P105	O	Port output	1	X										1-163
			5,3F6h	Port P105 direction register										1-168
			5,3FEh	P104 to P107 pull-up control register										1-170

Appendix 3 Setting register for pin function selected

Pin name	P104/AN4/$\overline{KI0}$		Pin No.	92 (FP)	90 (GP)								
Function selection			Setting register									Remarks	Page
			4,3F6h	5,3FEh									
$\overline{KI0}$	I	Key input	0	X									1-62
AN4	I	A-D input	0	0									1-149
P104	I	Port input (No pulled high)	0	0									1-163
P104	I	Port input (Pulled high)	0	1									1-163
P104	O	Port output	1	X									1-163
			4,3F6h	Port P104 direction register									1-168
			5,3FEh	P104 to P107 pull-up control register									1-170

Pin name	P103/AN3		Pin No.	93 (FP)	91 (GP)								
Function selection			Setting register									Remarks	Page
			3,3F6h	4,3FEh									
AN3	I	A-D input	0	0									1-149
P103	I	Port input (No pulled high)	0	0									1-163
P103	I	Port input (Pulled high)	0	1									1-163
P103	O	Port output	1	X									1-163
			3,3F6h	Port P103 direction register									1-168
			4,3FEh	P100 to P103 pull-up control register									1-170

Pin name	P102/AN2		Pin No.	94 (FP)	92 (GP)								
Function selection			Setting register									Remarks	Page
			2,3F6h	4,3FEh									
AN2	I	A-D input	0	0									1-149
P102	I	Port input (No pulled high)	0	0									1-163
P102	I	Port input (Pulled high)	0	1									1-163
P102	O	Port output	1	X									1-163
			2,3F6h	Port P102 direction register									1-168
			4,3FEh	P100 to P103 pull-up control register									1-170

Pin name	P101/AN1		Pin No.	95 (FP)	93 (GP)								
Function selection			Setting register									Remarks	Page
			1,3F6h	4,3FEh									
AN1	I	A-D input	0	0									1-149
P101	I	Port input (No pulled high)	0	0									1-163
P101	I	Port input (Pulled high)	0	1									1-163
P101	O	Port output	1	X									1-163
			1,3F6h	Port P101 direction register									1-168
			4,3FEh	P100 to P103 pull-up control register									1-170

Appendix 3 Setting register for pin function selected

Pin name	P100/AN0		Pin No.	97 (FP)	95 (GP)									
Function selection			Setting register										Remarks	Page
			0,3F6h	4,3FEh										
AN0	I	A-D input	0	0										1-149
P100	I	Port input (No pulled high)	0	0										1-163
P100	I	Port input (Pulled high)	0	1										1-163
P100	O	Port output	1	X										1-163
			0,3F6h	Port P100 direction register										1-168
			4,3FEh	P100 to P103 pull-up control register										1-170

Pin name	P97/$\overline{\text{ADTRG}}$/SIN4		Pin No.	100 (FP)	98 (GP)									
Function selection			Setting register										Remarks	Page
			7,3F3h	3,3FEh	5,3D6h	3,366h								
$\overline{\text{ADTRG}}$	I	A-D trigger input	0	0	1	0								1-149
SIN4	I	Serial I/O data input	0	0	X	1								1-145
P97	I	Port input (No pulled high)	0	0	X	0								1-163
P97	I	Port input (Pulled high)	0	1	X	0								1-163
P97	O	Port output	1	X	X	0								1-163
			7,3F3h	Port P97 direction register										1-168
			3,3FEh	P94 to P97 pull-up control register										1-170
			5,3D6h	Trigger select bit of A-D control register 0										1-151
			3,366h	SI/O4 port select bit of SI/O4 control register										1-146

Appendix 4 A practical example of connecting to the reset IC

M62015 and M62016 are reset ICs compatible with the M16C's backup mode. Here follow these ICs' overview and characteristics together with an example of connecting to the M16C when Vcc = 3 V.

• Overview of the reset ICs

Either M62015 or M62016 detects the rising edge of the power, the falling edge, and abnormal voltage of the power of the 3-V family of microcomputer systems. It is an optimal semiconductor integrated circuit to reset or release the microcomputer system.

Either M62015 or M61016 carries out 2-step power voltage detection, and is provided with two output terminals (the forced reset signal output RESET and the interrupt handling signal output INT). Either M62015 or M62016 embeds the BiCMOS process and the low power consumption circuit, and outputs optimal signals from respective output terminals with low power consumption especially in dealing with a system that requires its RAM backup.

• Characteristics of the reset ICs

* BiCMOS process low-consumption circuit configuration

Circuit current	Icc = 3 μA (standard value normal mode Vcc = 3.0 V)
	Icc = 3 μA (standard value backup mode Vcc = 2.5 V)

* Two-step power voltage detection

Normal power detection	Vs = 2.7 V (standard value)
Power detection for backup	V_{BATT} = 2.0 V (standard value)

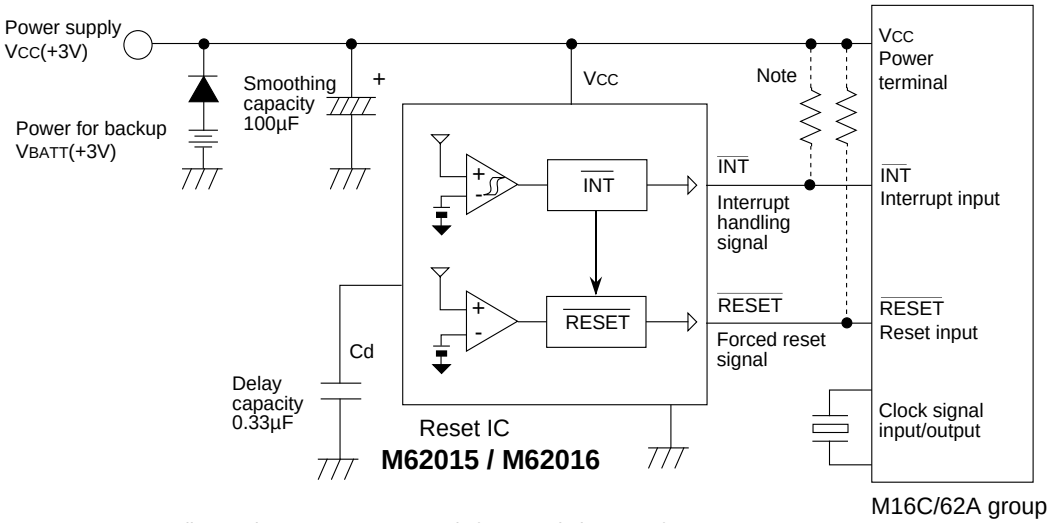
* Two outputs

Forced reset signal output	$\overline{RESET}$
Interrupt handling signal output	$\overline{INT}$

* Output form

CMOS output	: M62015
Open drain output	: M62016

• A practical example of connection



Note: Pull-up resistors are necessary only for open drain output form.

Appendix 5 Countermeasures against noise

Countermeasures against noise are described below. The following countermeasures are generally effective as countermeasures against noise, however, it is necessary not only to take measures as follows but to evaluate before actual use.

1.1 Shortest wiring length

The wiring on a printed circuit board can function as an antenna which feeds noise into the microcomputer. The shorter the total wiring length (by mm unit), the less the possibility of noise insertion into a microcomputer.

1.1.1 Package

Select the smallest possible package to make the total wiring length short.

● Reason

The wiring length depends on a microcomputer package. Use of a small package, for example QFP and not DIP, makes the total wiring length short to reduce influence of noise.

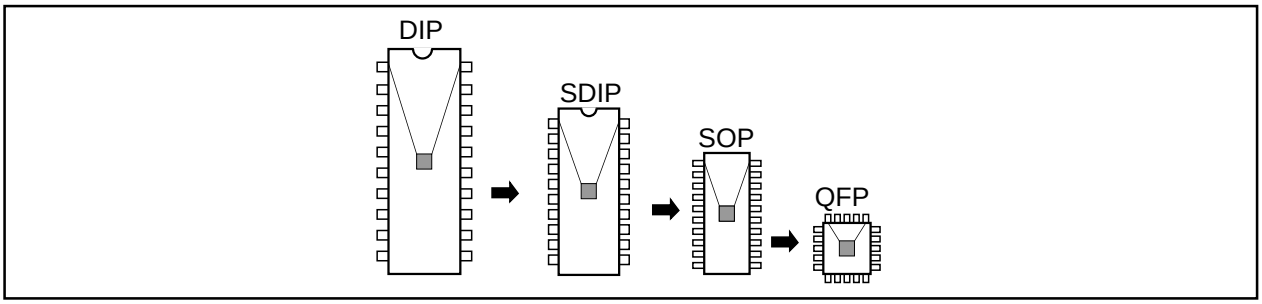


Figure 1.1.1. Selection of packages

1.1.2 Wiring for $\overline{\text{RESET}}$ pin

Make the length of wiring which is connected to the $\overline{\text{RESET}}$ pin as short as possible. Especially, connect a capacitor across the $\overline{\text{RESET}}$ pin and the V_{SS} pin or reset IC with the shortest possible wiring (within 20mm).

● Reason

The width of a pulse input into the $\overline{\text{RESET}}$ pin is determined by the timing necessary conditions. If noise having a shorter pulse width than the standard is input to the $\overline{\text{RESET}}$ pin, the reset is released before the internal state of the microcomputer is completely initialized. This may cause a program runaway.

Figure 1.1.2. Wiring for $\overline{\text{RESET}}$ pin

1.1.3 Wiring for clock input/output pins

- Make the length of wiring which is connected to clock I/O pins as short as possible.
- Make the length of wiring (within 20 mm) across the grounding lead of a capacitor which is connected to an oscillator and the Vss pin of a microcomputer as short as possible.
- Separate the Vss pattern only for oscillation from other Vss patterns. (See Figure 1.4.3.)

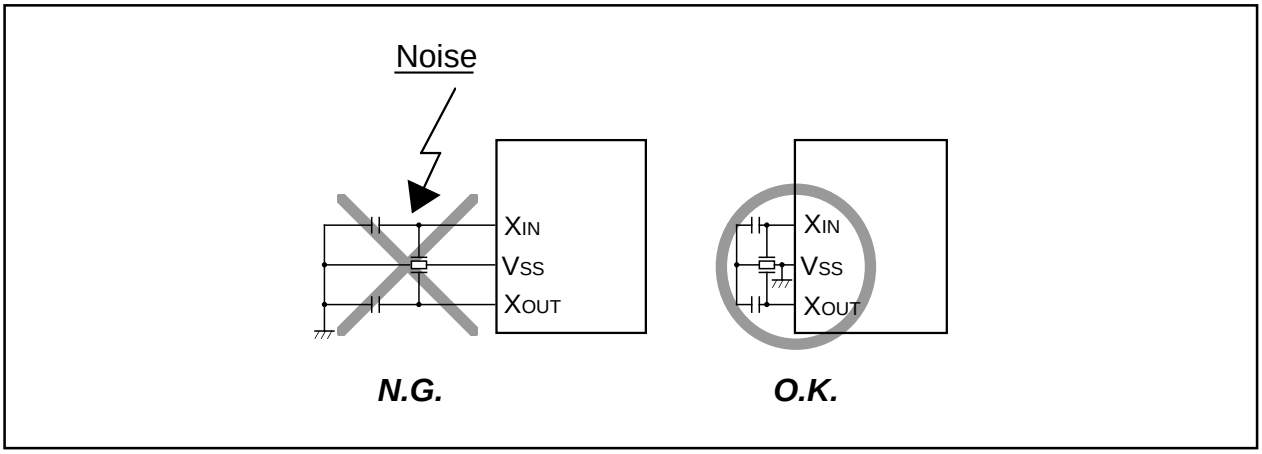


Figure 1.1.3. Wiring for clock I/O pins

● Reason

If noise enters clock I/O pins, clock waveforms may be deformed. This may cause a program failure or program runaway. Also, if a potential difference is caused by the noise between the Vss level of a microcomputer and the Vss level of an oscillator, the correct clock will not be input in the microcomputer.

1.1.4 Wiring to CNVss pin

Connect the CNVss pin to the Vss pin with the shortest possible wiring.

● Reason

The processor mode of a microcomputer is influenced by a potential at the CNVss pin. If a potential difference is caused by the noise between pins CNVss and Vss, the processor mode may become unstable. This may cause a microcomputer malfunction or a program runaway.

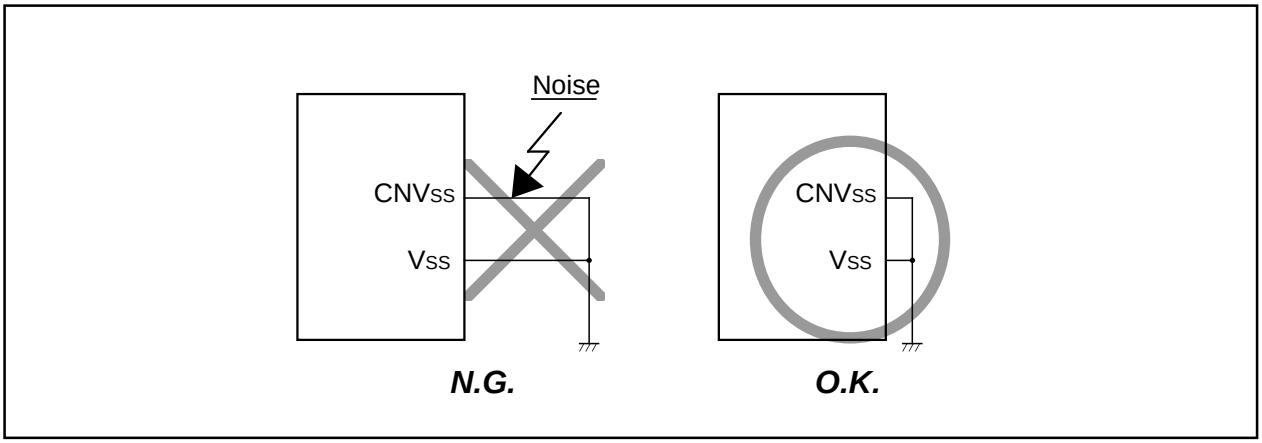


Figure 1.1.4. Wiring for CNVss pin

1.2 Connection of bypass capacitor across Vss line and Vcc line

Connect an approximately 0.1 μF bypass capacitor across the Vss line and the Vcc line as follows:

- Connect a bypass capacitor across the Vss pin and the Vcc pin at equal length.
- Connect a bypass capacitor across the Vss pin and the Vcc pin with the shortest possible wiring.
- Use lines with a larger diameter than other signal lines for Vss line and Vcc line.
- Connect the power source wiring via a bypass capacitor to the Vss pin and the Vcc pin.

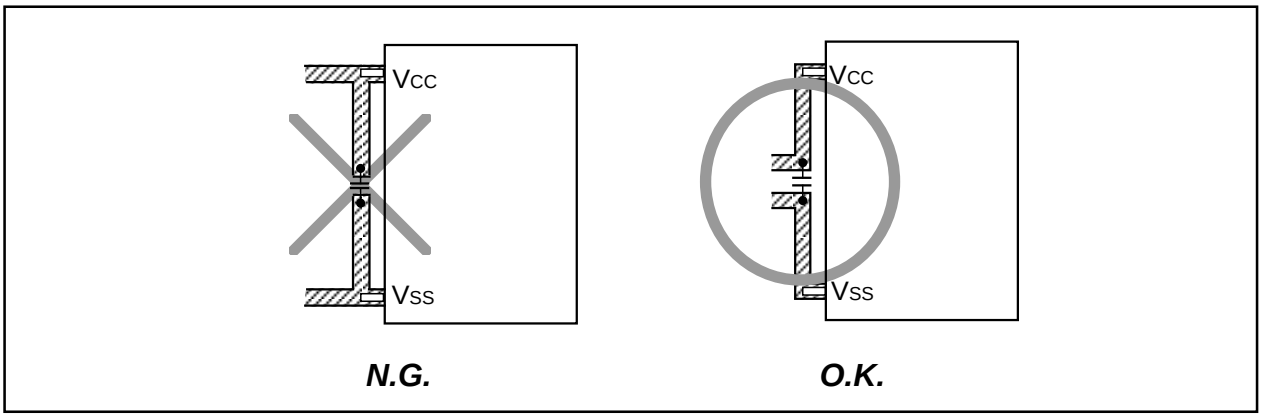


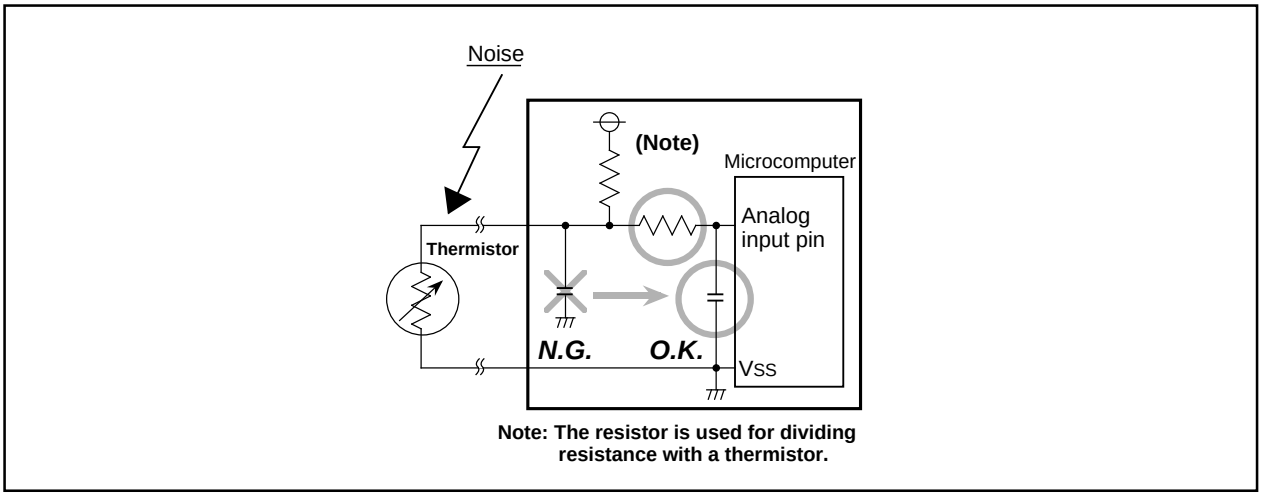
Figure 1.2.1. Bypass capacitor across the Vss line and the Vcc line

1.3 Wiring to analog input pins

- Connect an approximately 100 Ω to 1 kΩ resistor to an analog signal line which is connected to an analog input pin in series. Besides, connect the resistor to the microcomputer as close as possible.
- Connect an approximately 1000 pF capacitor across the Vss pin and the analog input pin. Besides, connect the capacitor to the Vss pin as close as possible. Also, connect the capacitor across the analog input pin and the Vss pin at equal length.

● Reason

Signals which is input in an analog input pin (such as an A-D converter/comparator input pin) are usually output signals from sensor. The sensor which detects a change of event is installed far from the printed circuit board with a microcomputer, the wiring to an analog input pin is longer necessarily. This long wiring functions as an antenna which feeds noise into the microcomputer, which causes noise to an analog input pin.



Note: The resistor is used for dividing resistance with a thermistor.

Figure 1.3.1. Analog signal line and a resistor and a capacitor

1.4 Oscillator concerns

Take care to prevent an oscillator that generates clocks for a microcomputer operation from being affected by other signals.

1.4.1 Keeping oscillator away from large current signal lines

Install a microcomputer (and especially an oscillator) as far as possible from signal lines where a current larger than the tolerance of current value flows.

● Reason

In the system using a microcomputer, there are signal lines for controlling motors, LEDs, and thermal heads or others. When a large current flows through those signal lines, strong noise occurs because of mutual inductance.

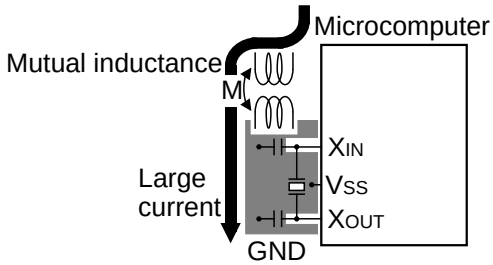


Figure 1.4.1. Wiring for a large current signal line

1.4.2 Installing oscillator away from signal lines where potential levels change frequently

Install an oscillator and a connecting pattern of an oscillator away from signal lines where potential levels change frequently. Also, do not cross such signal lines over the clock lines or the signal lines which are sensitive to noise. Or do not stretch long such signal lines parallelly to these said lines.

● Reason

Signal lines where potential levels change frequently (such as the TAOUT pin signal line) may affect other lines at signal rising edge or falling edge. If such lines cross over a clock line, clock waveforms may be deformed, which causes a microcomputer failure or a program runaway.

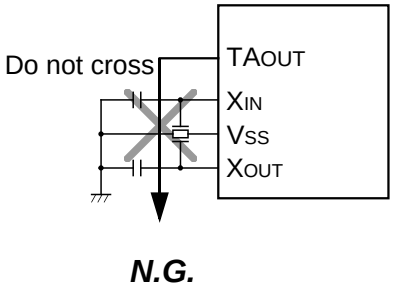


Figure 1.4.2. Wiring of signal lines where potential levels change frequently

1.4.3 Oscillator protection using Vss pattern

As for a two-sided printed circuit board, print a VSS pattern on the underside (soldering side) of the position (on the component side) where an oscillator is mounted.

Connect the VSS pattern to the microcomputer VSS pin with the shortest possible wiring. Besides, separate this VSS pattern from other VSS patterns.

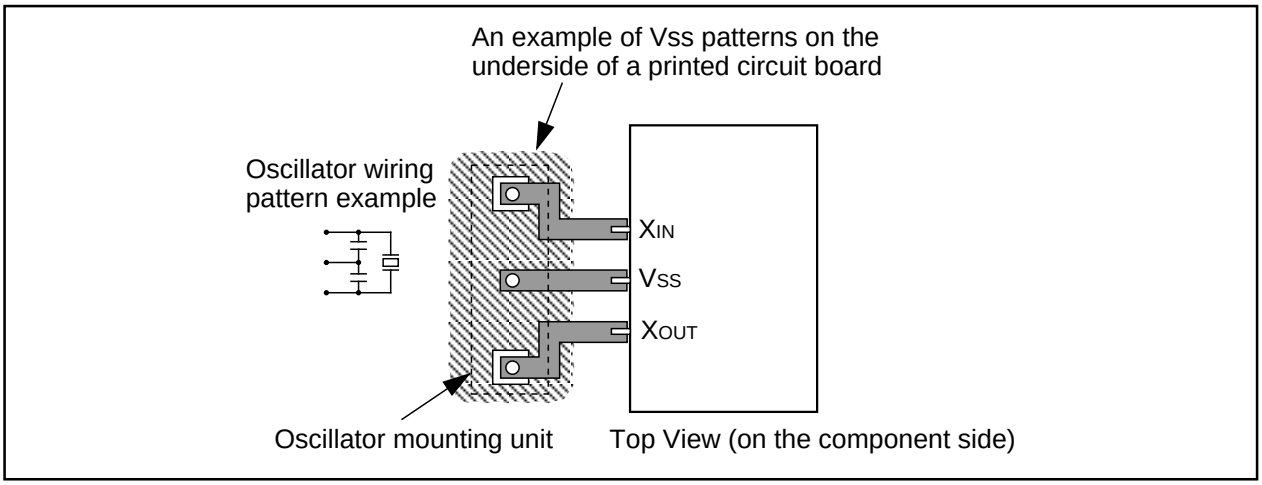


Figure 1.4.3. VSS pattern on the underside of an oscillator

1.5 Setup for I/O ports

Setup I/O ports using hardware and software as follows:

<Hardware>

- Connect a resistor of 100 Ω or more to a signal line which is connected to an I/O port in series. Besides, connect the resistor to the microcomputer as close as possible.

<Software>

- As for an input port, read data several times by a program for checking whether input levels are equal or not.
- As for an output port, since the output data may reverse because of noise, rewrite data to its port latch at fixed periods.
- Rewrite data to direction registers and pull-up control registers at fixed periods.

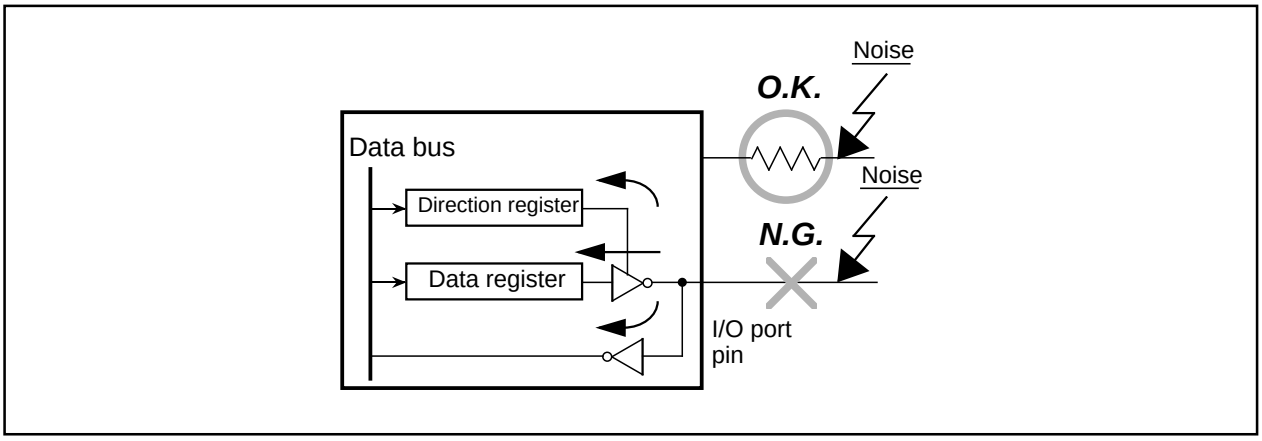


Figure 1.5.1. Setup for I/O ports

1.6 Providing of watchdog timer function by software

If a microcomputer runs away because of noise or others, it can be detected by a software watchdog timer and the microcomputer can be reset to normal operation. This is equal to or more effective than program runaway detection by a hardware watchdog timer. The following shows an example of a watchdog timer provided by software.

In the following example, to reset a microcomputer to normal operation, the main routine detects errors of the interrupt processing routine and the interrupt processing routine detects errors of the main routine.

This example assumes that interrupt processing is repeated multiple times in a single main routine processing.

<The main routine>

• Assigns a single byte of RAM to a software watchdog timer (SWDT) and writes the initial value N in the SWDT once at each execution of the main routine. The initial value N should satisfy the following condition:

N+1 ≥≥ (Counts of interrupt processing executed in each main routine)

As the main routine execution cycle may change because of an interrupt processing or others, the initial value N should have a margin.

• Watches the operation of the interrupt processing routine by comparing the SWDT contents with counts of interrupt processing after the initial value N has been set.

• Detects that the interrupt processing routine has failed and determines to branch to the program initialization routine for recovery processing in the following case:

If the SWDT contents do not change after interrupt processing.

<The interrupt processing routine>

• Decrements the SWDT contents by 1 at each interrupt processing.

• Determines that the main routine operates normally when the SWDT contents are reset to the initial value N at almost fixed cycles (at the fixed interrupt processing count).

• Detects that the main routine has failed and determines to branch to the program initialization routine for recovery processing in the following case:

If the SWDT contents are not initialized to the initial value N but continued to decrement and if they reach 0 or less.

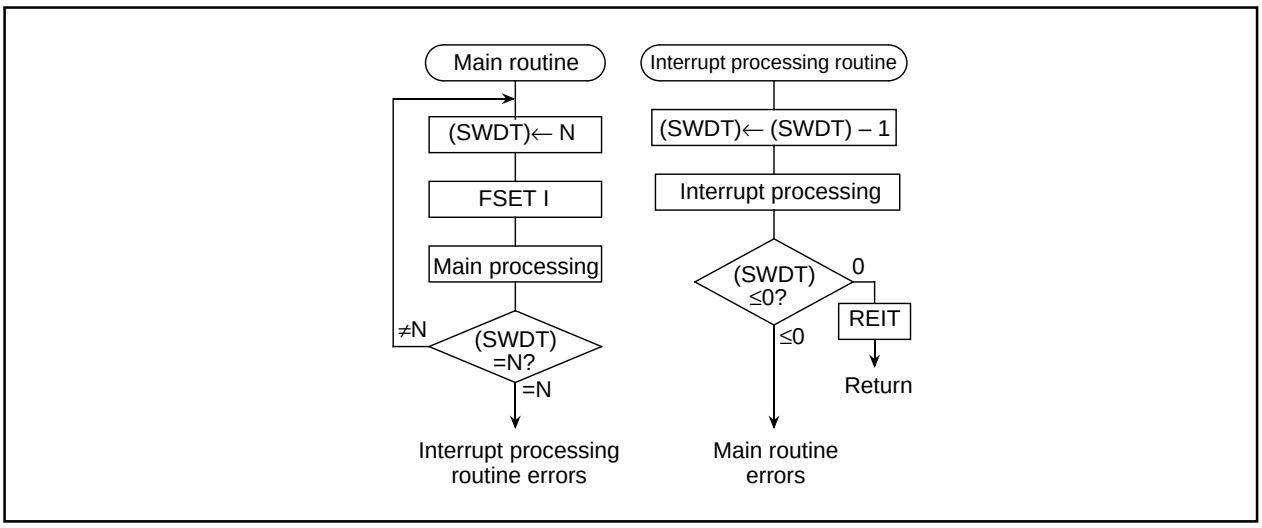


Figure 1.6.1. Watchdog timer by software

Index

A

A practical example of connecting to the reset IC 2-288
A-D Converter 1-149
A-D Converter Usage 2-110
Address Match Interrupt 1-63
Address Match Interrupt Usage 2-158
Applications 1-2

B

Block Diagram 1-5
Bus Control 1-28
Bus Settings 1-26
Buzzer Output 2-204

C

Central Processing Unit (CPU) 1-12
Check Sheet 2-254
Clock Control 1-37
Clock Generating Circuit 1-36
Clock Output 1-40
Clock-Asynchronous Serial I/O (UART) Usage 2-74
Clock-Synchronous Serial I/O Usage 2-52
Connectable Memories 2-232
Connection Examples 2-226
Controlling Power Using Stop Mode 2-212
Controlling Power Using Wait Mode 2-216
Countermeasures against noise 2-289
CPU Rewrite Mode 1-219
CRC Calculation Circuit 1-161
CRC Calculation Circuit Usage 2-152

D

D-A Converter 1-159
D-A Converter Usage 2-142
Data Access 2-223
Delayed One-Shot Output 2-200
Description 1-2
DMAC 1-68
DMAC Usage 2-144

E

Example of oscillator circuit 1-36

F

Features 1-2
Flash Memory 1-218
Functions To Inhibit Rewriting Flash Memory Version 1-233

I

$\overline{\text{INT}}$ Interrupt 1-61
Internal Reserved Area Expansion Bit (PM13) 1-24
Interrupt Control 1-51
Interrupt Priority 1-59
Interrupt resolution circuit 1-59
Interrupt Sequence 1-55
Items to be submitted when ordering masked ROM version 1-173

K

Key Input Interrupt 1-62
Key-Input Interrupt Usage 2-162

L

Long-Period Timers 2-192

M

Memory 1-11
Memory Map (External ROM version) 2-249
Memory to Memory DMA Transfer 2-208
Multiple interrupts Usage 2-166

N

$\overline{\text{NMI}}$ Interrupt 1-62

O

Operation of Functional Blocks 1-11
Outline Performance 1-217
Outline Performance (CPU Rewrite Mode) 1-220
Overview of External Buses 2-222
Overview of Interrupt 1-46
Overview of standard serial I/O mode 1 (clock synchronized) 1-240
Overview of standard serial I/O mode 2 (clock asynchronized) 1-256

P

Parallel I/O Mode 1-235
Performance Outline 1-6
Pin Configuration 1-3
Pin Configuration (External ROM version) 2-245
Pin Description (External ROM version) 2-247
Power control 1-43
Power Control Usage 2-173
Precautions for External Bus 2-242
Precautions for Interrupts 1-64
Processor Mode 1-22, 2-253
Programmable I/O Ports 1-163
Programmable I/O Ports Usage 2-181
Protect Usage 2-2
Protection 1-45

R

Releasing an External Bus ($\overline{\text{HOLD}}$ input and $\overline{\text{HLDA}}$ output) 2-241
Reset 1-15
Returning from an Interrupt Routine 1-59

S

Serial I/O 1-108
SI/O3, 4 Usage 2-106
Software Reset 1-22
Solution for External Interrupt Pins Shortage 2-206
Standard serial I/O mode 1-239
Status Transition of BCLK 1-42
Stop Mode 1-40

T

Timer 1-78
Timer A 1-80
Timer A Usage 2-6
Timer B 1-90
Timer B Usage 2-38
Timers functions for three-phase motor control 1-96

U

UART0 to 2 1-108

V

Variable-Period Variable-Duty PWM Output 2-196

W

Wait Mode 1-41
Watchdog Timer 1-66
Watchdog Timer Usage 2-154

MITSUBISHI Single-Chip Microcomputer
User's Manual
M16C/62 (M16C/62A, M16C/62M) Group Rev.1.0

June First Edition 2001

Editioned by
Committee of editing of Mitsubishi Semiconductor USER'S MANUAL

Published by
Mitsubishi Electric Corp., Kitaitami Works

This book, or parts thereof, may not be reproduced in any form without permission of Mitsubishi Electric Corporation.
©2001 MITSUBISHI ELECTRIC CORPORATION

User's Manual
M16C/62 (M16C/62A, M16C/62M) Group

Renesas Technology Corp.
Nippon Bldg.,6-2,Otemachi 2-chome,Chiyoda-ku,Tokyo,100-0004 Japan