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

Overview

The M16C/10 group (M30100 and M30102 groups) consist of single-chip microcomputers that use high-performance silicon gate CMOS processes and have a on-chip M16C/60 series CPU core. The microcomputers are housed in 32-pin plastic mold QFP or 48-pin plastic mold QFP packages. These single-chip microcomputers have both high function instructions and high instruction efficiency and feature a one-megabyte address space and the capability to execute instructions at high speed.

The M30100 and M30102 groups consist of several products that have different on-chip memory types, sizes, and packages.

Features

- Basic machine language instructions .. Compatible with the M16C/60 series
- Memory size ..... ROM/RAM (See the memory expansion diagram.)
- Shortest instruction execution time ..... 62.5 ns (when f(XIN)=16MHz)
- Power supply voltage ..... 4.2 V to 5.5V (when f(XIN)=16MHz)
  2.7 V to 5.5V (when f(XIN)=5MHz) (This is not applicable to applications for automobile use)
- Interrupts ..... 12 internal causes, 7 external causes, 4 software causes (including key input interrupts)
- 8-bit timers ..... 4 with 8-bit prescaler (PWM output of Timer Y, Z: selectable)
- 16-bit timer ..... 1 (time measurement timer)
- Serial I/O ..... UART or clock synchronization type x 2
- A-D converter ..... 10-bit X 12 channels (can be expanded to 14 channels)
- D-A converter 1
- Watchdog timer 1
- Programmable I/O ports 34
- LED drive ports 8
- Clock generation circuits ..... 3 internal circuits
  - Main clock generation circuit
    An internal feedback resistor and an externally attached ceramic resonator/quartz crystal oscillator/RC oscillator.
  - Sub clock generation circuit
    An internal feedback resistor and an externally attached ceramic resonator/quartz crystal oscillator
  - Ring oscillator

Specifications written in this manual are believed to be accurate, but are not guaranteed to be entirely free of error.
Specifications in this manual may be changed for functional or performance improvements. Please make sure your manual is the latest edition.

Applications

Home appliances, office devices, audio, automobile, other

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

Pin Configuration

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

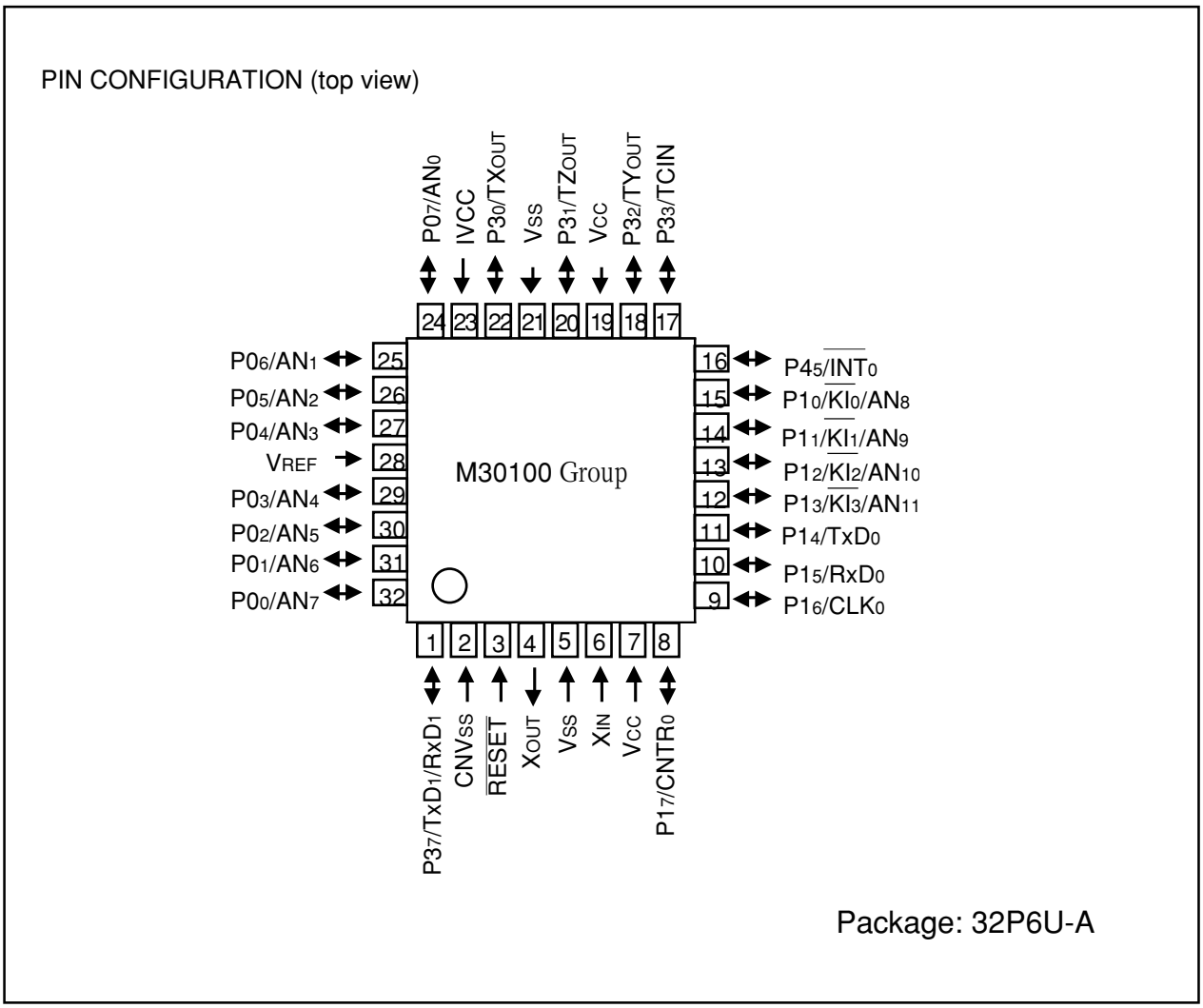


Figure 1.1.1. Pin configuration diagram (top view) of the M30100 group

PIN CONFIGURATION (top view)

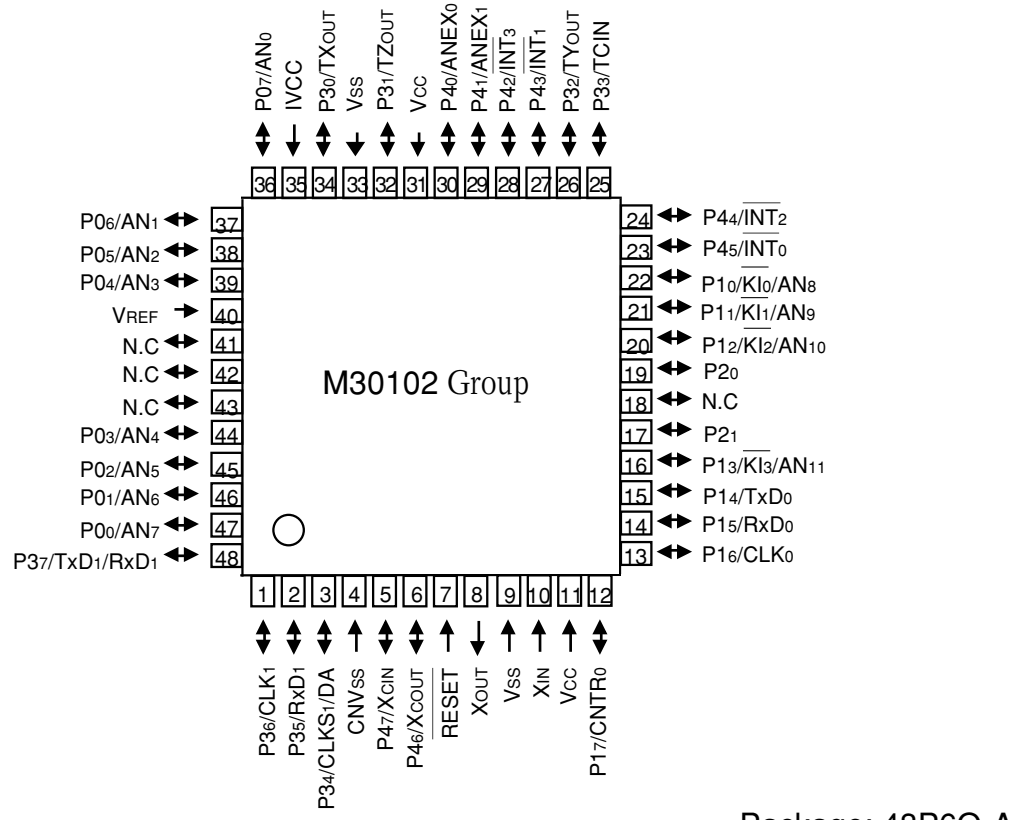


Figure 1.1.2. Pin configuration diagram (top view) of the M30102 group

Block Diagram

Figure 1.1.3 is a block diagram of the M30100 group.

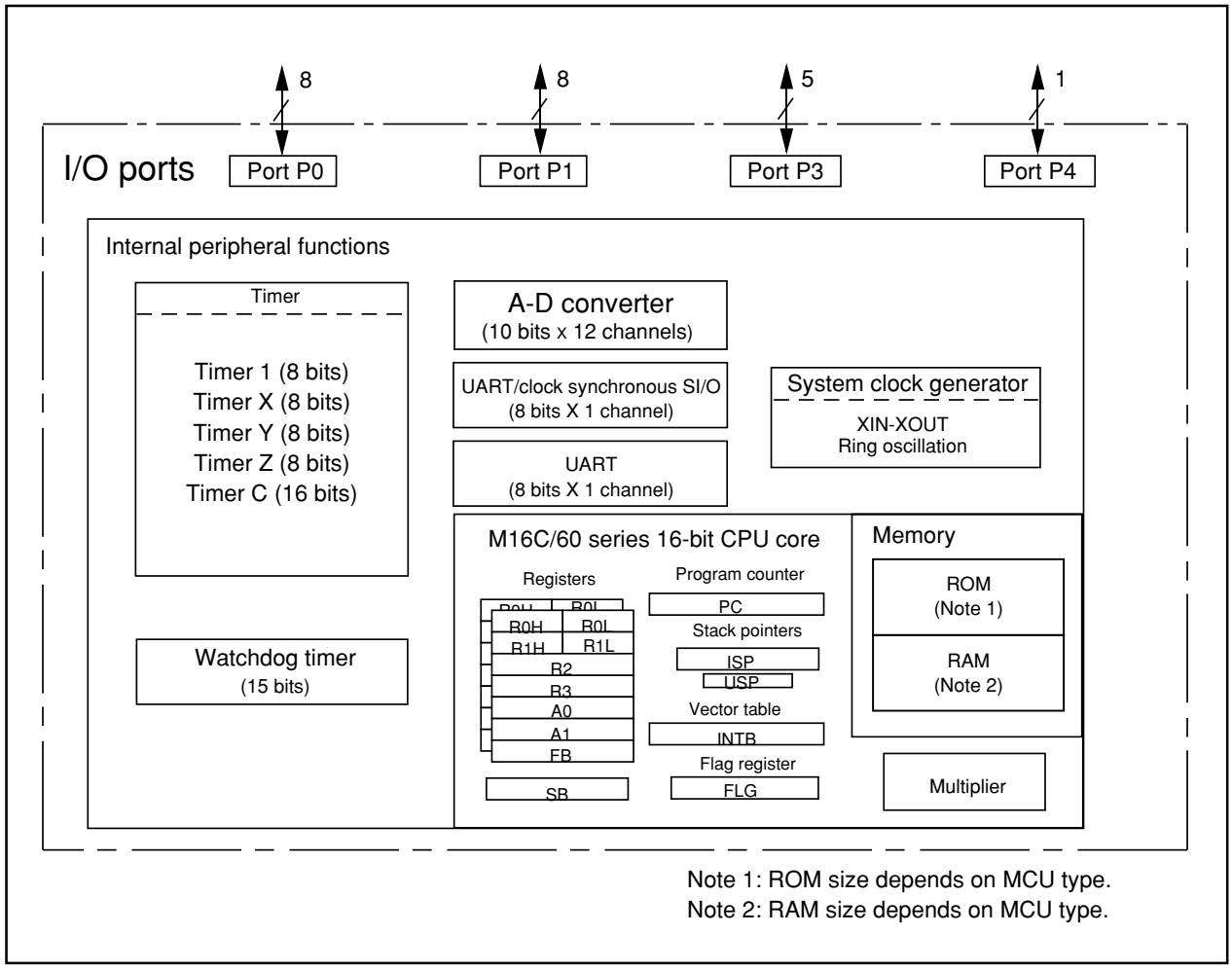


Figure 1.1.3. Block diagram for the M30100 group

Block Diagram

Figure 1.1.4 is a block diagram of the M30102 group.

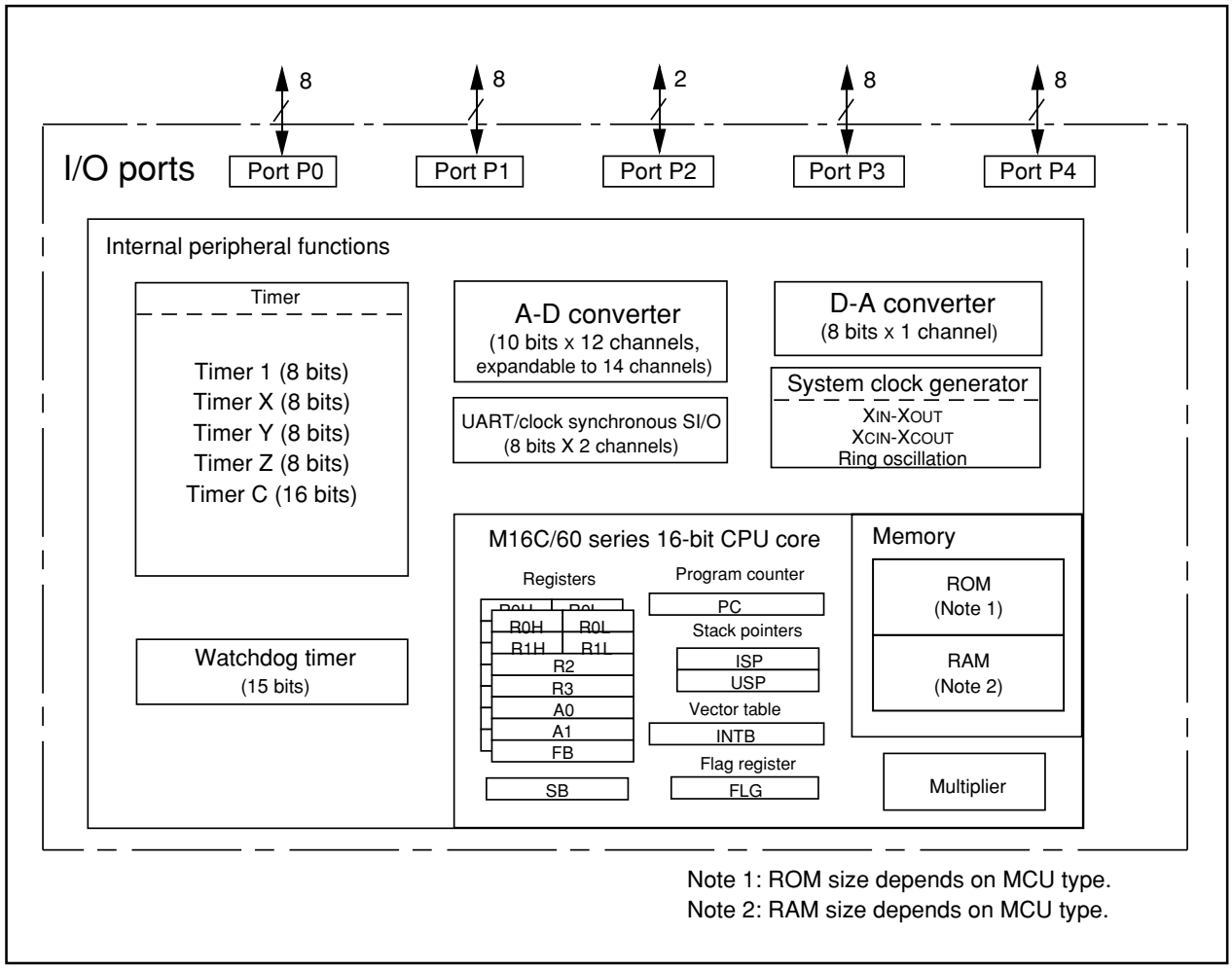


Figure 1.1.4. Block diagram for the M30102 group

Performance Overview

Table 1.1.1 gives an overview of the M16C/10 group performance specification.

Table 1.1.1. M16C/10 group performance overview

Item		Performance	
		M30100	M30102
Number of basic instructions		91 instructions	
Shortest instruction execution time		62.5 ns (when f(XIN)=16MHz)	
Memory size	ROM	See the memory expansion diagram.	
	RAM	See the memory expansion diagram.	
I/O port		P0,P1,P3,P4: 22 lines	P0 to P4: 34 lines
Multifunction timer	T1	8 bits x 1	
	TX, TY, TZ	8 bits x 3	
	TC	16 bits x 1	
Serial I/O (UART or clock synchronous)		x 2 (one is exclusively for UART)	x 2
A-D converter (maximum resolution: 10 bits)		x 12 channels	x 12 channels (Expandable up to 14 channels)
D-A converter		——	8 bits x 1
Watchdog timer		15 bits x 1 (with prescaler)	
Interrupts		12 internal causes, 7 external causes (4 for M30100), 4 software causes	
Clock generating circuits		2 internal circuits	3 internal circuits
Power supply voltage		4.2 V to 5.5V (when f(XIN)=16MHz) 2.7 V to 5.5V (when f(XIN)=5MHz) (Note)	
Power consumption		100mW (Vcc=5.0V, f(XIN)=16MHz) 12mW (Vcc=3.0V, f(XIN)=5MHz)	
I/O characteristics	I/O withstand voltage	5V	
	Output current	5mA (10mA:LED drive port)	
Device configuration		CMOS silicon gate	
Package		32-pin LQFP	48-pin LQFP

Note: This voltage is not applicable to applications for automobile use.

Mitsubishi plans to release the following products in the M16C/10 group:

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

(2) Memory size

(3) Package

32P6U: Plastic molded LQFP (mask ROM version and flash memory version)

48P6Q: Plastic molded LQFP (mask ROM version and flash memory version)

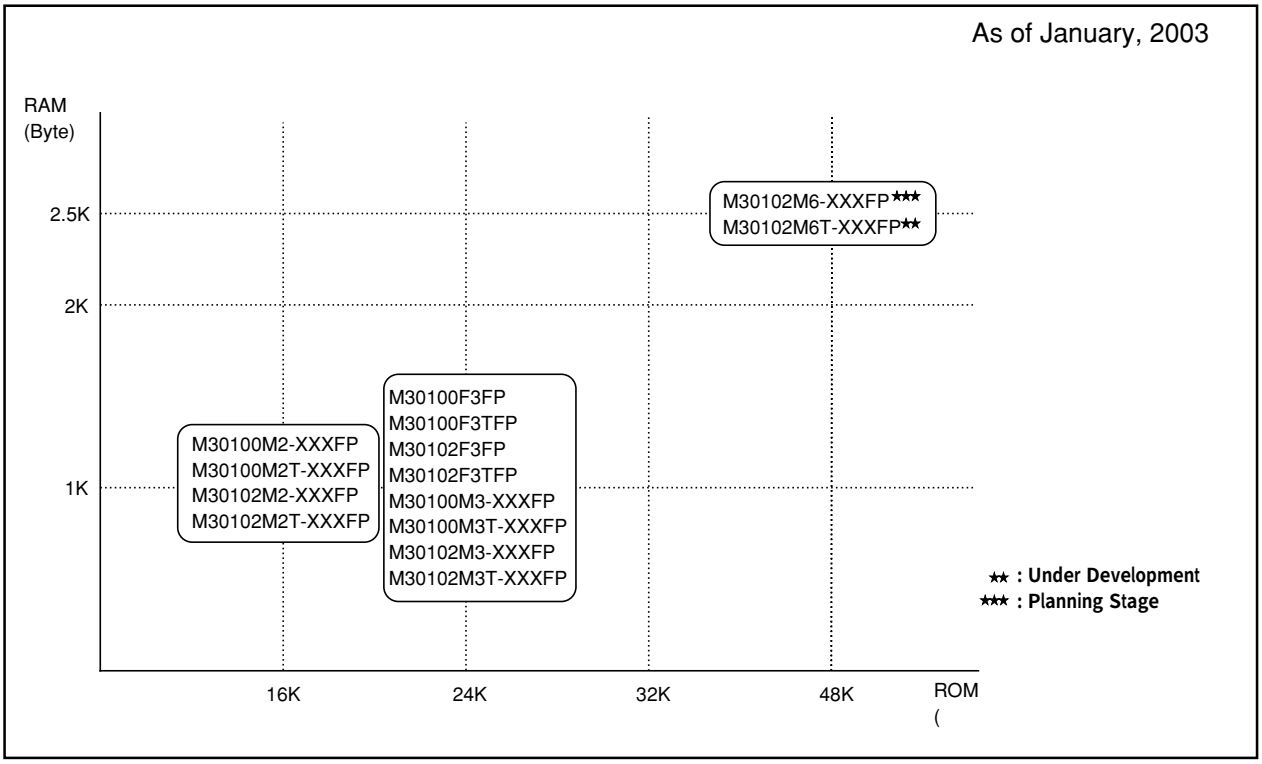


Figure 1.1.5. Memory expansion

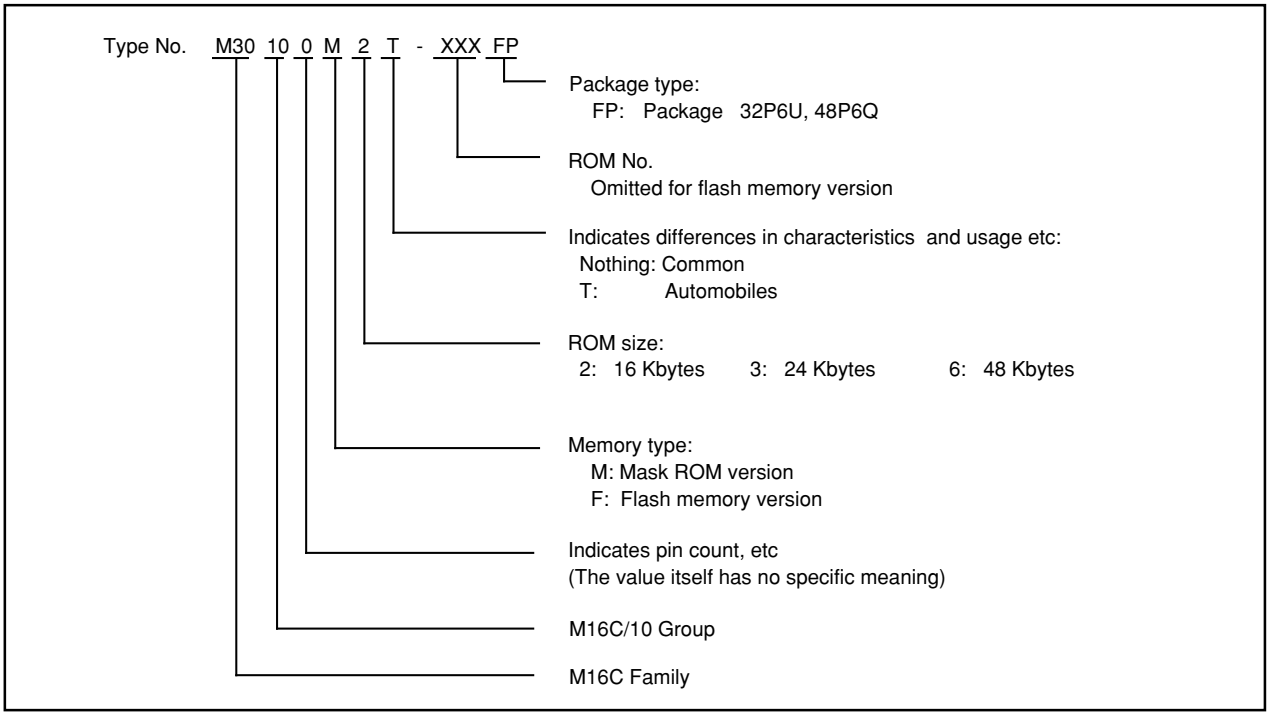


Figure 1.1.6. Type No., memory size, and package

Pin Description

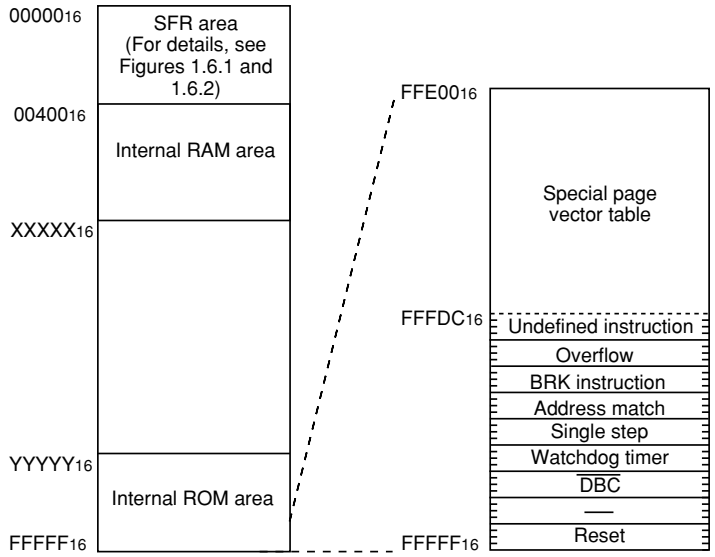
Pin name	Signal name	I/O type	Function
Vcc, VSS	Power supply input		Supply 2.7 to 5.5 V to the VCC pin. Supply 0 V to the VSS pin.
IVCC			Connect a capacitor (0.1 μF) between this pin and VSS.
CNVSS	CNVSS	Input	Connect it to the VSS pin via resistance (about 5 kΩ).
$\overline{\text{RESET}}$	Reset input	Input	An "L" on this input resets the microcomputer.
XIN XOUT	Clock input Clock output	Input Output	These pins are provided for the main clock oscillation circuit. Connect a ceramic resonator or crystal between the XIN and XOUT pins. To use an externally derived clock, input it to the XIN pin and leave the XOUT pin open.
VREF	Reference voltage input	Input	This pin is a reference voltage input for the A-D converter.
P00 to P07	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 set for input, the user can specify in units of four bits via software whether or not they are tied to a pull-up resistor. These pins are shared with analog input pins.
P10 to P17	I/O port P1	Input/output	This is an 8-bit I/O port equivalent to P0. P10 to P13 are shared with analog inputs and key input interrupts. P14 to P16 are shared with serial I/O pins. P17 is shared with timer input. Can be used as an LED drive port.
P20 to P21	I/O port P2	Input/output	This is a 2-bit I/O port equivalent to P0.
P30 to P37	I/O port P3	Input/output	This is a 8-bit I/O port equivalent to P0. P30 to P33 are shared with timer input/output. P34 to P37 are shared with serial I/O. P34 is shared with analog outputs.
P40 to P47	I/O port P4	Input/output	This is a 8-bit I/O port equivalent to P0. P40 to 41 are shared with analog inputs. P42 to P45 are shared with interrupt inputs. P46 to P47 are shared with the I/O pin of the clock oscillation circuit for the clock.

Operation of Functional Blocks

The M30100/M30102 contain the following devices on a single chip: ROM and RAM, which function as memory for storing instructions and data; a central processing unit (CPU) that executes operations; and peripheral devices, such as timers, serial I/O, an A-D converter, an D-A converter, and I/O ports.
The individual devices are described below.

Memory

Figure 1.3.1 is a memory map. The address space extends the 1M bytes from address 00000_{16} to $FFFFF_{16}$. From $FFFFF_{16}$ down is ROM. For example, in the M30100M2-XXXFP, there is 16K bytes of internal ROM from $FC000_{16}$ to $FFFFF_{16}$. The vector table for fixed interrupts such as the reset 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 M30100M2-XXXFP, there is 1K byte of internal RAM from 00400_{16} to $007FF_{16}$. 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 $000FF_{16}$. This area accommodates the control registers for peripheral devices such as I/O ports, A-D converter, serial I/O, and timers, etc. 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.



Type No.	Internal RAM area		Internal ROM area	
	Memory size	XXXXX₁₆	Memory size	YYYYY₁₆
M30100M2/M2T、M30102M2/M2T	1K byte	007FF₁₆	16K byte	FC000₁₆
M30100M3/M3T/F3/F3T、M30102M3/M3T/F3/F3T	1K byte	007FF₁₆	24K byte	FA000₁₆
M30102M6★★★ /M6T★★	2.5K byte	00DFF₁₆	48K byte	F4000₁₆

★★ : Under Development
★★★ : Planning Stage

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.

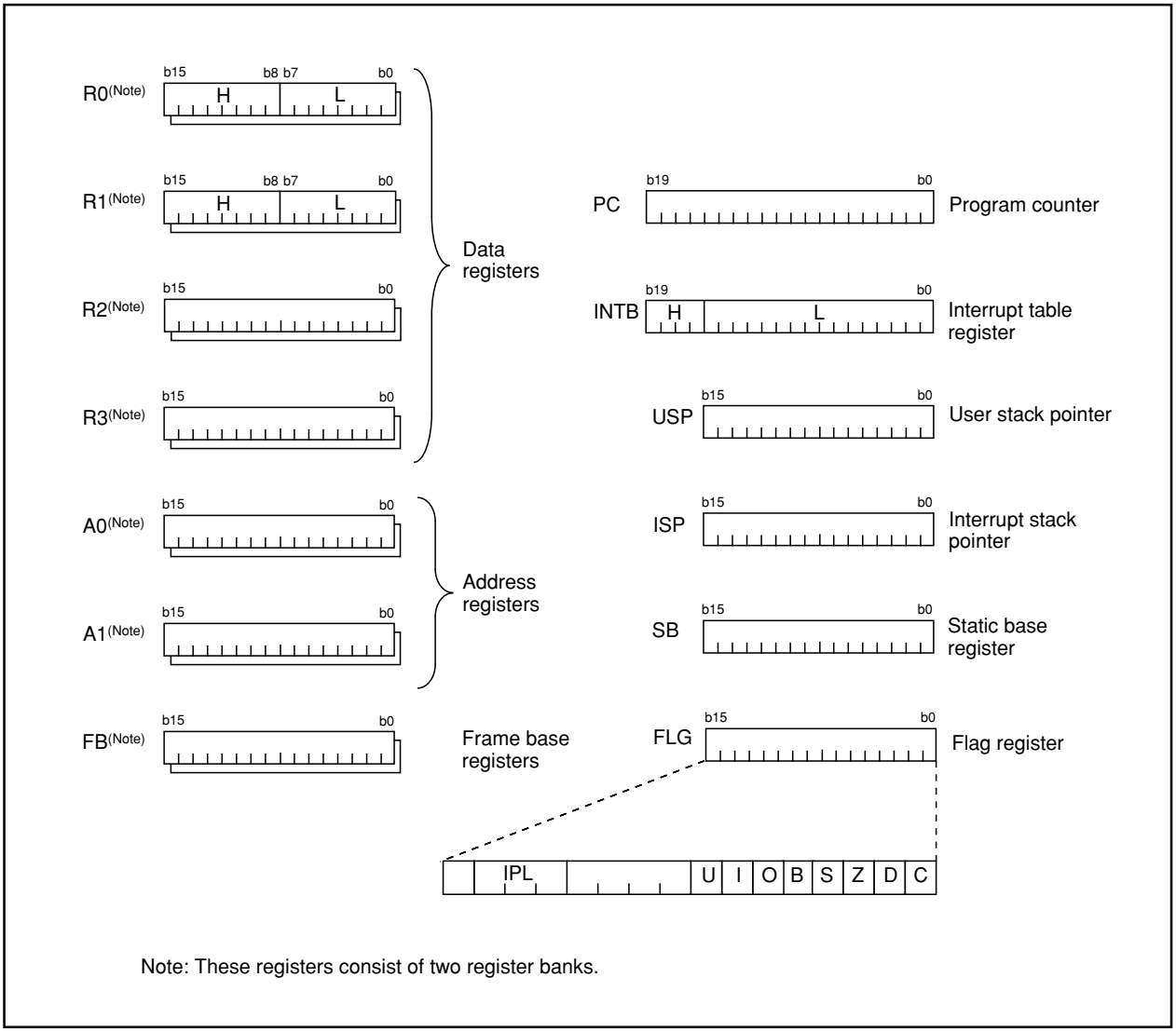


Figure 1.4.1. Central processing unit register

(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.
- **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 $\overline{\text{INT}}$ 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.

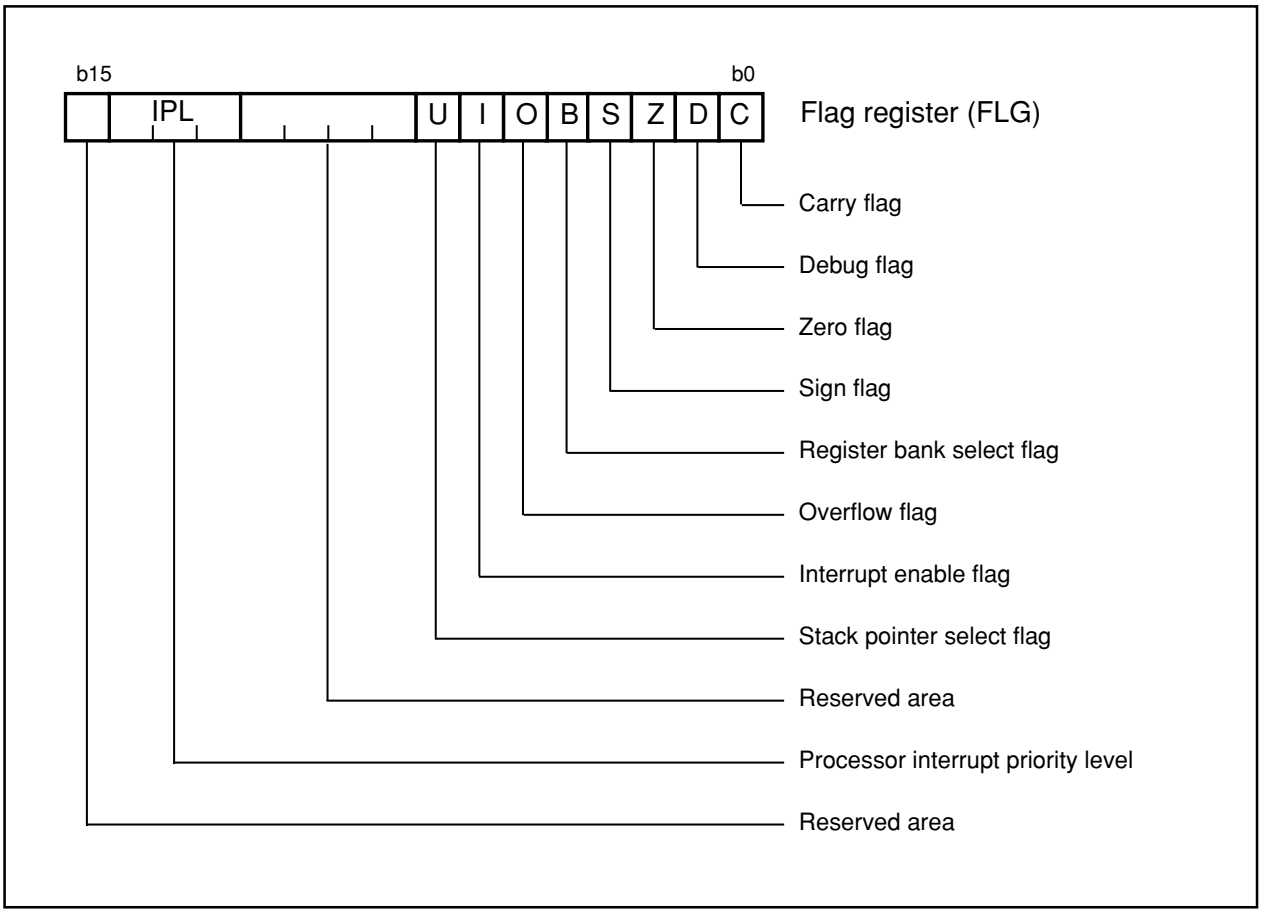


Figure 1.4.2. Flag register (FLG)

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 200 µsec. When the reset pin level is then returned to the “H” level, the reset status is cancelled and program execution resumes from the address in the reset vector table. Since the value of RAM is indeterminate when power is applied, the initial values must be set. Also, if a reset signal is input during write to RAM, the access to the RAM will be interrupted. Consequently, the value of the RAM being written may change to an unintended value due to the interruption.

Figures 1.5.1 and 1.5.2 show the example reset circuit. Figure 1.5.3 shows the reset sequence.

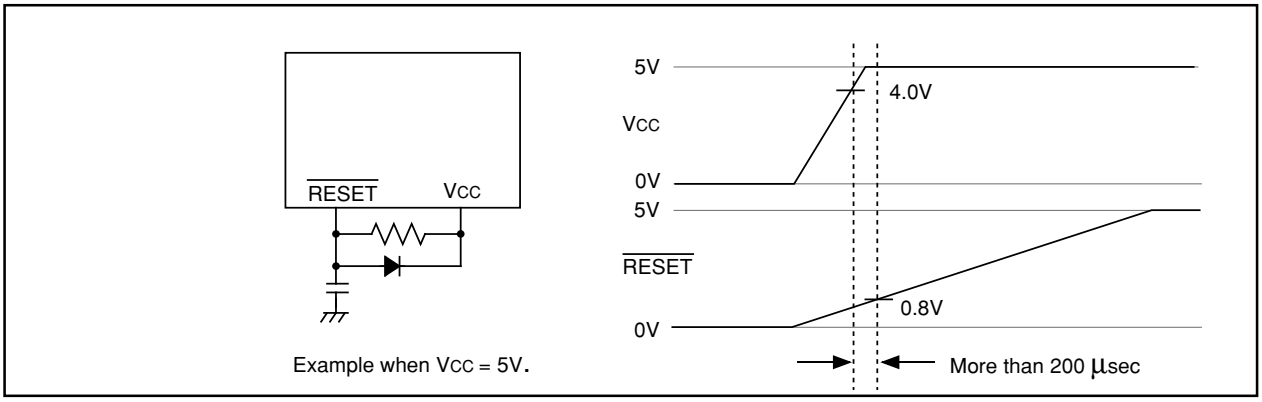


Figure 1.5.1. Example reset circuit

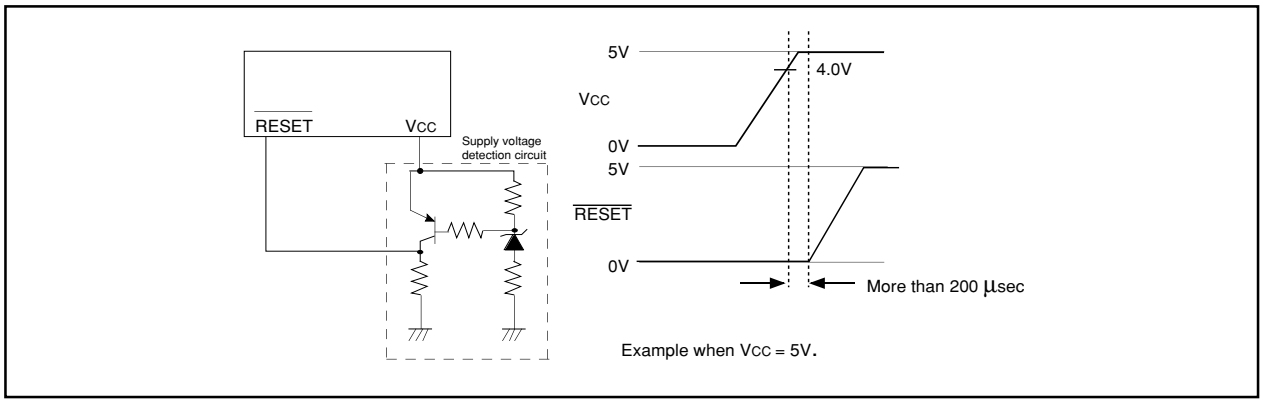


Figure 1.5.2. Example reset circuit (example voltage check circuit)

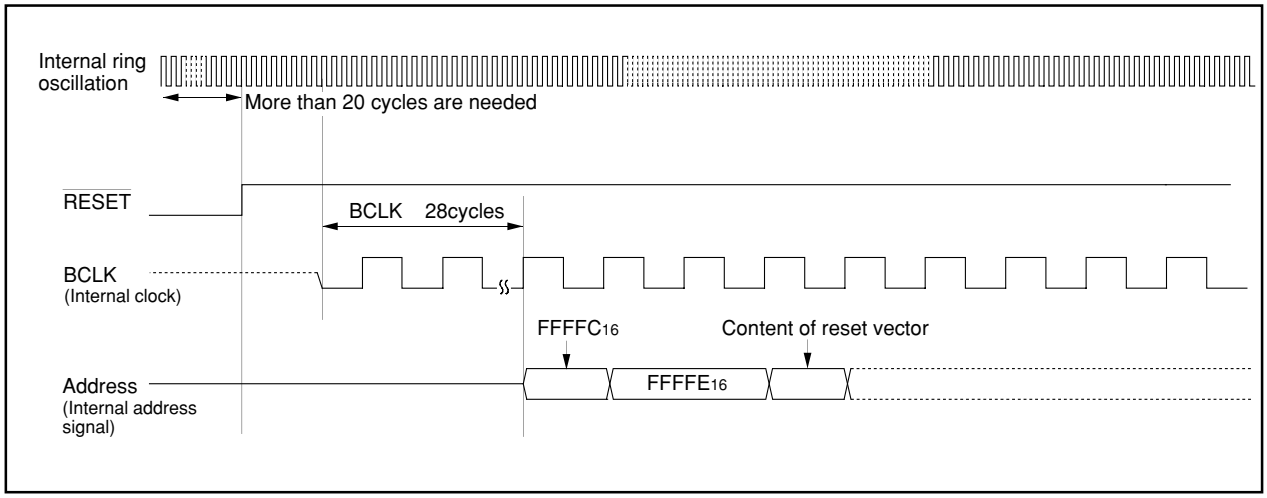


Figure 1.5.3. Reset sequence

(1) Processor mode register 0	(0004_{16})···	xxxx0xx00
(2) Processor mode register 1	(0005_{16})···	00xxxx0x0
(3) System clock control register 0	(0006_{16})···	01001000
(4) System clock control register 1	(0007_{16})···	00100000
(5) Address match interrupt enable register	(0009_{16})···	xxxxxx00
(6) Protect register	($000A_{16}$)···	xxxxx000
(7) Oscillation stop detection register	($000C_{16}$)···	00000100
(8) Watchdog timer control register	($000F_{16}$)···	000?????
(9) Address match interrupt register 0	(0010_{16})···	00_{16}
	(0011_{16})···	00_{16}
	(0012_{16})···	xxxx0000
(10) Address match interrupt register 1	(0014_{16})···	00_{16}
	(0015_{16})···	00_{16}
	(0016_{16})···	xxxx0000
(11) INT0 input filter select register	($001E_{16}$)···	xxxxx000
(12) Key input interrupt control register	($004D_{16}$)···	xxxx?000
(13) A-D conversion interrupt control register	($004E_{16}$)···	xxxx?000
(14) UART0 transmit interrupt control register	(0051_{16})···	xxxx?000
(15) UART0 receive interrupt control register	(0052_{16})···	xxxx?000
(16) UART1 transmit interrupt control register	(0053_{16})···	xxxx?000
(17) UART1 receive interrupt control register	(0054_{16})···	xxxx?000
(18) Timer 1 interrupt control register	(0055_{16})···	xxxx?000
(19) Timer X interrupt control register	(0056_{16})···	xxxx?000
(20) Timer Y interrupt control register	(0057_{16})···	xxxx?000
(21) Timer Z interrupt control register	(0058_{16})···	xxxx?000
(22) CNTR0 interrupt control register	(0059_{16})···	xxxx?000
(23) TCIN interrupt control register	($005A_{16}$)···	xxxx?000
(24) Timer C interrupt control register	($005B_{16}$)···	xxxx?000
(25) INT3 interrupt control register	($005C_{16}$)···	xxxx?000
(26) INT0 interrupt control register	($005D_{16}$)···	xx00?000
(27) INT1 interrupt control register	($005E_{16}$)···	xx00?000
(28) INT2 interrupt control register	($005F_{16}$)···	xx00?000
(29) Timer Y, Z mode register	(0080_{16})···	00000x00
(30) Prescaler Y	(0081_{16})···	FF_{16}
(31) Timer Y secondary	(0082_{16})···	FF_{16}
(32) Timer Y primary	(0083_{16})···	FF_{16}
(33) Timer Y, Z waveform output control register	(0084_{16})···	00_{16}
(34) Prescaler Z	(0085_{16})···	FF_{16}
(35) Timer Z secondary	(0086_{16})···	FF_{16}
(36) Timer Z primary	(0087_{16})···	FF_{16}
(37) Timer Y,Z output control register	($008A_{16}$)···	xxxxx000
(38) Timer X mode register	($008B_{16}$)···	x0000000
(39) Prescaler X	($008C_{16}$)···	FF_{16}
(40) Timer X	($008D_{16}$)···	FF_{16}
(41) Timer count source set register	($008E_{16}$)···	00_{16}
(42) Clock prescaler reset flag	($008F_{16}$)···	0xxxxxxx
(43) External input enable register	(0096_{16})···	00_{16}
(44) Key input enable register	(0098_{16})···	00_{16}
(45) Timer C control register 0	($009A_{16}$)···	0xx00000
(46) Timer C control register 1	($009B_{16}$)···	xxxxxx11
(47) UART0 transmit/receive mode register	($00A0_{16}$)···	00_{16}
(48) UART0 transmit/receive control register 0	($00A4_{16}$)···	00001000
(49) UART0 transmit/receive control register 1	($00A5_{16}$)···	xxxx0010
(50) UART1 transmit/receive mode register	($00A8_{16}$)···	00_{16}
(51) UART1 transmit/receive control register 0	($00AC_{16}$)···	00001000
(52) UART1 transmit/receive control register 1	($00AD_{16}$)···	xxxx0010
(53) UART transmit/receive control register 2	($00B0_{16}$)···	x0000000
(54) A-D control register 2	($00D4_{16}$)···	xxxx0000
(55) A-D control register 0	($00D6_{16}$)···	00000???
(56) A-D control register 1	($00D7_{16}$)···	00_{16}
(57) D-A control register	($00DC_{16}$)···	xxxxx0x0
(58) Port P0 direction register	($00E2_{16}$)···	00_{16}
(59) Port P1 direction register	($00E3_{16}$)···	00_{16}
(60) Port P2 direction register	($00E6_{16}$)···	xxxxxx00
(61) Port P3 direction register	($00E7_{16}$)···	00_{16}
(62) Port P4 direction register	($00EA_{16}$)···	00_{16}
(63) Pull-up control register 0	($00FC_{16}$)···	00x00000
(64) Pull-up control register 1	($00FD_{16}$)···	xxxxxx00
(65) Port P1 drive capacity control register	($00FE_{16}$)···	00_{16}
(66) Data registers (R0/R1/R2/R3)		0000_{16}
(67) Address registers (A0/A1)		0000_{16}
(68) Frame base register (FB)		0000_{16}
(69) Interrupt table register (INTB)		00000_{16}
(70) User stack pointer (USP)		0000_{16}
(71) Interrupt stack pointer (ISP)		0000_{16}
(72) Static base register (SB)		0000_{16}
(73) Flag register (FLG)		0000_{16}

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.

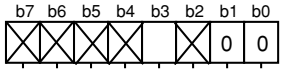
Figure 1.5.4. Device's internal status after a reset is cleared

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 almost the same effect as a hardware reset. The contents of internal RAM are preserved.

Figure 1.5.5 shows the processor mode register 0 and 1.

Processor mode register 0 (Note)

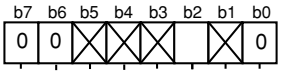


Symbol	Address	When reset
PM0	0004_{16}	$XXXX0X00_2$

Bit symbol	Bit name	Function	R	W
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.			—	—
PM03	Software reset bit	The device is reset when this bit is set to “1”. The value of this bit is “0” when read.	○	○
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be indeterminate.			—	—

Note: Set bit 1 of the protect register (address $000A_{16}$) to “1” when writing new values to this register.

Processor mode register 1 (Note 1)



Symbol	Address	When reset
PM1	0005_{16}	$00XXX0X0_2$

Bit symbol	Bit name	Function	R	W
Reserved bit		Must always be set to “0”	○	○
Nothing is assigned. In an attempt to write to this bits, write “0”. The value, if read, turns out to be indeterminate.			—	—
PM12	WDT interrupt/reset switching bit	0 : Watchdog timer interrupt 1 : Reset (Note 2)	○	○
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be indeterminate.			—	—
Reserved bit		Must always be set to “0”	○	○

Note 1: Set bit 1 of the protect register (address $000A_{16}$) to “1” when writing new values to this register.
Note 2: After setting this bit to "1", can not change to "0" by software.

Figure 1.5.5. Processor mode register 0 and 1.

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	
0009₁₆	Address match interrupt enable register (AIER)
000A₁₆	Protect register (PRCR)
000B₁₆	
000C₁₆	Oscillation stop detection register (CM2)
000D₁₆	
000E₁₆	Watchdog timer start register (WDTS)
000F₁₆	Watchdog timer control register (WDC)
0010₁₆ 0011₁₆ 0012₁₆	Address match interrupt register 0 (RMAD0)
0013₁₆	
0014₁₆ 0015₁₆ 0016₁₆	Address match interrupt register 1 (RMAD1)
0017₁₆	
0018₁₆	
0019₁₆	
001A₁₆	
001B₁₆	
001C₁₆	
001D₁₆	
001E₁₆	INT0 input filter select register (INT0F)
001F₁₆	
0020₁₆ 0021₁₆ 0022₁₆	
0023₁₆	
0024₁₆ 0025₁₆ 0026₁₆	
0027₁₆	
0028₁₆ 0029₁₆	
002A₁₆	
002B₁₆	
002C₁₆	
002D₁₆	
002E₁₆	
002F₁₆	
0030₁₆ 0031₁₆ 0032₁₆	
0033₁₆	
0034₁₆ 0035₁₆ 0036₁₆	
0037₁₆	
0038₁₆ 0039₁₆	
003A₁₆	
003B₁₆	
003C₁₆	
003D₁₆	
003E₁₆	
003F₁₆	

0040₁₆	
0041₁₆	
0042₁₆	
0043₁₆	
0044₁₆	
0045₁₆	
0046₁₆	
0047₁₆	
0048₁₆	
0049₁₆	
004A₁₆	
004B₁₆	
004C₁₆	
004D₁₆	Key input interrupt control register (KUPIC)
004E₁₆	A-D conversion interrupt control register (ADIC)
004F₁₆	
0050₁₆	
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 1 interrupt control register (T1IC)
0056₁₆	Timer X interrupt control register (TXIC)
0057₁₆	Timer Y interrupt control register (TYIC)
0058₁₆	Timer Z interrupt control register (TZIC)
0059₁₆	CNTR0 interrupt control register (CNTR0IC)
005A₁₆	TCIN interrupt control register (TCINIC)
005B₁₆	Timer C interrupt control register (TCIC)
005C₁₆	INT3 interrupt control register (INT3IC)
005D₁₆	INT0 interrupt control register (INT0IC)
005E₁₆	INT1 interrupt control register (INT1IC)
005F₁₆	INT2 interrupt control register (INT2IC)

Note: The blank area is reserved and must not be read or written.

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

0080₁₆	Timer Y, Z mode register (TYZMR)
0081₁₆	Prescaler Y (PREY)
0082₁₆	Timer Y secondary (TYSC)
0083₁₆	Timer Y primary (TYPR)
0084₁₆	Timer Y, Z waveform output control register (PUM)
0085₁₆	Prescaler Z (PREZ)
0086₁₆	Timer Z secondary (TZSC)
0087₁₆	Timer Z primary (TZPR)
0088₁₆	Prescaler 1 (PRE1)
0089₁₆	Timer 1 (T1)
008A₁₆	Timer Y, Z output control register (TYZOC)
008B₁₆	Timer X mode register (TXMR)
008C₁₆	Prescaler X (PREX)
008D₁₆	Timer X (TX)
008E₁₆	Timer count source set register (TCSS)
008F₁₆	Clock prescaler reset flag (CPSRF)
0090₁₆ 0091₁₆	Timer C counter (TC)
0092₁₆	
0093₁₆	
0094₁₆	
0095₁₆	
0096₁₆	External input enable register (INTEN)
0097₁₆	
0098₁₆	Key input enable register (KIEN)
0099₁₆	
009A₁₆	Timer C control register 0 (TCC0)
009B₁₆	Timer C control register 1 (TCC1)
009C₁₆ 009D₁₆	Time measurement register (TM)
009E₁₆	
009F₁₆	
00A0₁₆	UART0 transmit/receive mode register (U0MR)
00A1₁₆	UART0 bit rate generator (U0BRG)
00A2₁₆ 00A3₁₆	UART0 transmit buffer register (U0TB)
00A4₁₆	UART0 transmit/receive control register 0 (U0C0)
00A5₁₆	UART0 transmit/receive control register 1 (U0C1)
00A6₁₆ 00A7₁₆	UART0 receive buffer register (U0RB)
00A8₁₆	UART1 transmit/receive mode register (U1MR)
00A9₁₆	UART1 bit rate generator (U1BRG)
00AA₁₆ 00AB₁₆	UART1 transmit buffer register (U1TB)
00AC₁₆	UART1 transmit/receive control register 0 (U1C0)
00AD₁₆	UART1 transmit/receive control register 1 (U1C1)
00AE₁₆ 00AF₁₆	UART1 receive buffer register (U1RB)
00B0₁₆	UART transmit/receive control register 2 (UCON)
00B1₁₆	
00B2₁₆	
00B3₁₆	
00B4₁₆	
00B5₁₆	
00B6₁₆	
00B7₁₆	
00B8₁₆	
00B9₁₆	
00BA₁₆	
00BB₁₆	
00BC₁₆	
00BD₁₆	
00BE₁₆	
00BF₁₆	

00C0₁₆ 00C1₁₆	A-D register (AD)
00C2₁₆	
00C3₁₆	
00C4₁₆	
00C5₁₆	
00C6₁₆	
00C7₁₆	
00C8₁₆	
00C9₁₆	
00CA₁₆	
00CB₁₆	
00CC₁₆	
00CD₁₆	
00CE₁₆	
00CF₁₆	
00D0₁₆	
00D1₁₆	
00D2₁₆	
00D3₁₆	
00D4₁₆	A-D control register 2 (ADCON2)
00D5₁₆	
00D6₁₆	A-D control register 0 (ADCON0)
00D7₁₆	A-D control register 1 (ADCON1)
00D8₁₆	D-A register (DA)
00D9₁₆	
00DA₁₆	
00DB₁₆	
00DC₁₆	D-A control register (DACON)
00DD₁₆	
00DE₁₆	
00DF₁₆	
00E0₁₆	Port P0 (P0)
00E1₁₆	Port P1 (P1)
00E2₁₆	Port P0 direction register (PD0)
00E3₁₆	Port P1 direction register (PD1)
00E4₁₆	Port P2 (P2)
00E5₁₆	Port P3 (P3)
00E6₁₆	Port P2 direction register (PD2)
00E7₁₆	Port P3 direction register (PD3)
00E8₁₆	Port P4 (P4)
00E9₁₆	
00EA₁₆	Port P4 direction register (PD4)
00EB₁₆	
00EC₁₆	
00ED₁₆	
00EE₁₆	
00EF₁₆	
00F0₁₆	
00F1₁₆	
00F2₁₆	
00F3₁₆	
00F4₁₆	
00F5₁₆	
00F6₁₆	
00F7₁₆	
00F8₁₆	
00F9₁₆	
00FA₁₆	
00FB₁₆	
00FC₁₆	Pull-up control register 0 (PUR0)
00FD₁₆	Pull-up control register 1 (PUR1)
00FE₁₆	Port P1 drive capacity control register (DRR)
00FF₁₆	

Note: The blank area is reserved and must not be read or written.

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

Bus Control

During access, the memory areas (ROM, RAM, FLASH, etc.) and the SFR area have different bus cycles. As shown in Table 1.7.1, memory areas can be accessed in one cycle of the CPU operation clock BCLK. The SFR area can be accessed in two cycles of BCLK.

Table 1.7.1. Bus cycles for access areas

Area	Bus cycle
SFR	2 BCLK cycles
Internal ROM/RAM	1 BCLK cycles

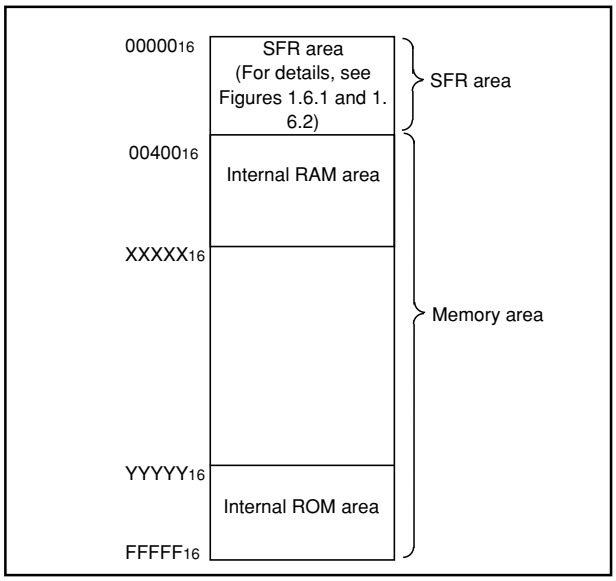
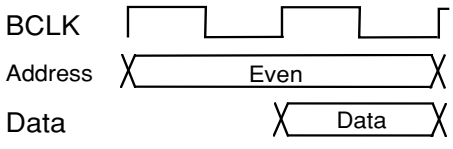
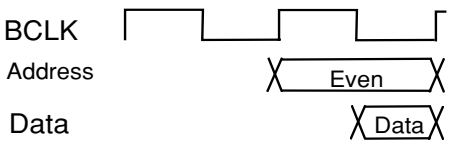
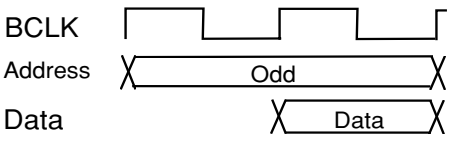
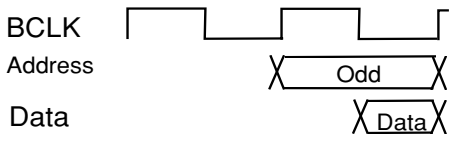
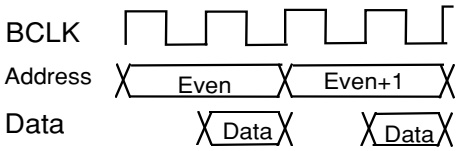
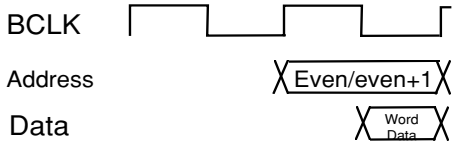
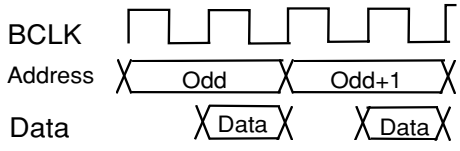
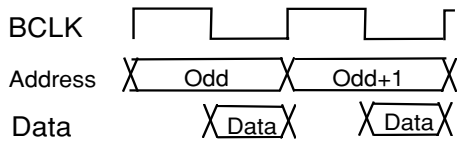


Figure 1.7.1. SFR area and memory areas

The memory areas and the SFR area also have different bus widths. The memory areas have a 16-bit bus width, while the SFR area has an 8-bit bus width. Consequently, different operations are used when the areas are accessed in word (16 bits) units. Table 1.7.2 shows the bus cycles that are necessary to access the SFR area and the memory areas.

Table 1.7.2. Cycles for access areas

Area	SFR area	Memory area
Even address byte access		
Add address byte access		
Even address word access		
Add address word access		

Clock Generating Circuit

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

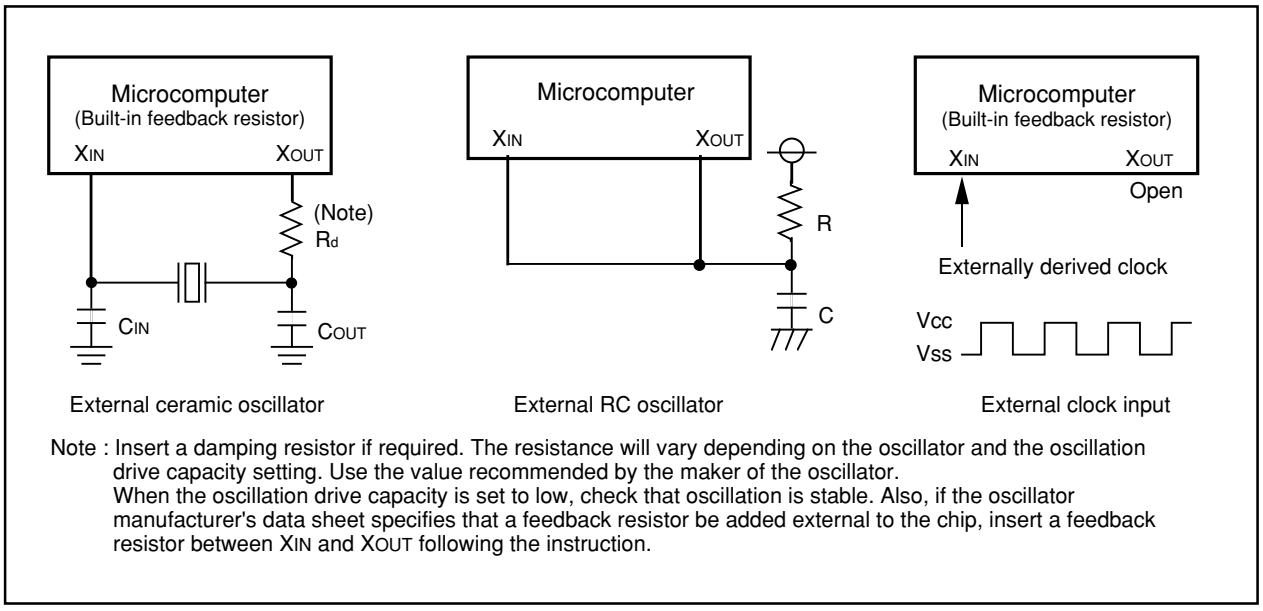
Table 1.8.1. Main clock, sub-clock, and ring oscillator generating circuits

	Main clock generating circuit	Sub clock generating circuit	Ring oscillator generating circuit
Use of clock	• CPU's operating clock source • Internal peripheral units' operating clock source	• CPU's operating clock source • Timer 1/X/Y/Z's count clock source	• CPU's operating clock source • Internal peripheral units' operating clock source • Timer Y's count clock source
Usable oscillator (Note)	Ceramic, crystal or RC oscillator	Crystal oscillator	–
Pins to connect oscillator	XIN, XOUT	XCIN, XCOUT	None (has internal pins)
Oscillation stop/restart function	Available	Available	Available
Oscillator status immediately after reset	Oscillating	Stopped	Oscillating
Other	Externally derived clock can be input		–

Note : When not using the main clock generating circuit, pull up the XIN pin and leave the XOUT pin open. Also, set the main clock stop bit (bit 5 of address 0006) to "1" (stop).

Example of oscillator circuit

Figure 1.8.1 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 1.8.2 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 1.8.1 and 1.8.2 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 1.8.1. Examples of main clock

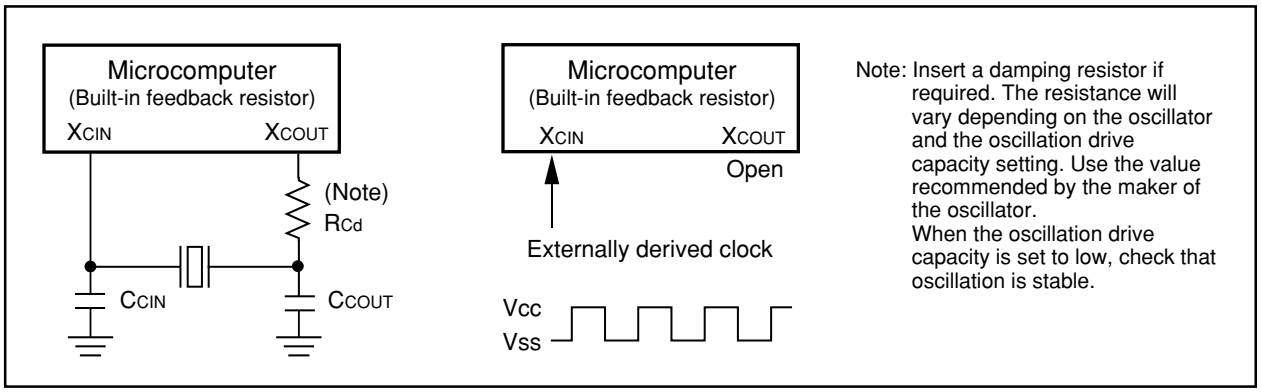


Figure 1.8.2. Examples of sub-clock

A ring oscillator is built into the microcomputer. The oscillation of the ring oscillator can be used as the BCLK by setting the main clock select bit (bit 2 of address 000C). Lower power consumption can be realized because the oscillating frequency of the ring oscillator is much lower compared to that of XIN. The frequency of the ring oscillator depends on the supply voltage and the operation temperature range. Be careful that variable frequencies and obtain the sufficient margin when designing application products.

Clock Control

Figure 1.8.3 shows the block diagram of the clock generating circuit.

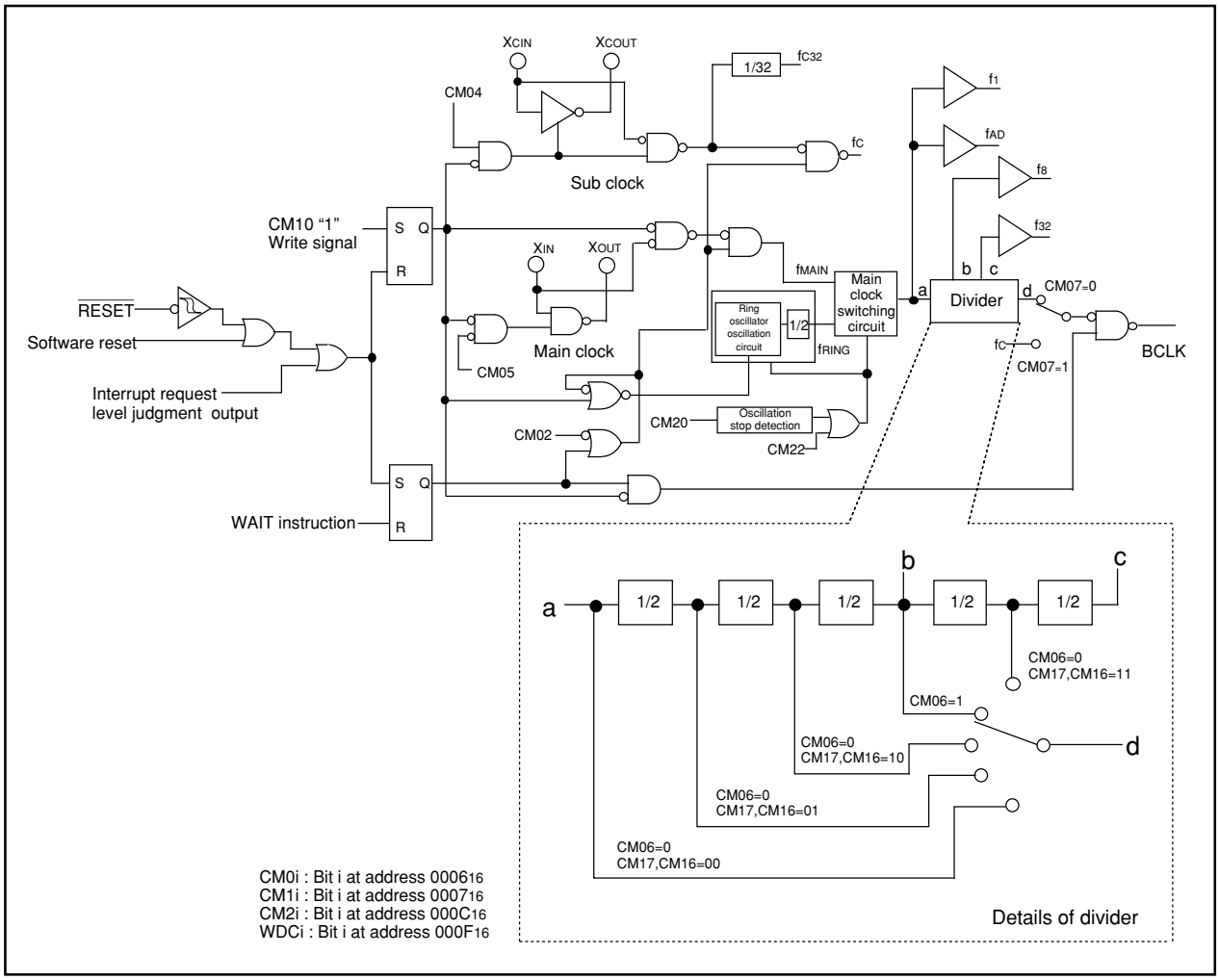


Figure 1.8.3. Clock generating circuit

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 reset, oscillation starts. The clock can be stopped using the main clock stop bit (bit 5 at address 0006$_{16}$). Stopping the clock reduces the power dissipation.

After the oscillation of the main clock oscillation circuit has stabilized, the drive capacity of the XOUT pin can be reduced using the XIN-XOUT drive capacity select bit (bit 5 at address 0007$_{16}$). Reducing the drive capacity of the XOUT pin 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 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 XCOUT pin can be reduced using the XCIN-XCOUT drive capacity select bit (bit 3 at address 0006$_{16}$). Reducing the drive capacity of the XCOUT pin reduces the power dissipation. This bit changes to "1" when shifting to stop mode and at a reset.

(3) BCLK

The BCLK is the clock that drives the CPU. The clock source for BCLK is as follows: (1) the clock derived by dividing the main clock by 1, 2, 4, 8, or 16, (2) fC, or (3) the clock derived by dividing the clock supplied by the ring oscillator circuit (fRING) by 1, 2, 4, 8 or 16. After reset, the BCLK is derived by dividing the fRING by 8. When using an external RC oscillator circuit for the main clock, 1 division of the main clock cannot be selected as BCLK.

The main clock division select bit 0(bit 6 at address 0006$_{16}$) changes to "1" when shifting from high-speed/medium-speed mode 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

a. f1, f8, f32

The clock for the peripheral devices is derived from the main clock or by dividing it by 8 or 32. The peripheral function clock is stopped as follows: (i) by stopping the main clock or (ii) by executing an WAIT instruction after setting the WAIT peripheral function clock stop bit (bit 2 at 0006$_{16}$) to "1". When using an external RC oscillator circuit for the main clock, f1 cannot be selected as the operation clock of some peripheral devices.

b. fAD

This clock has the same frequency as the main clock and is used in A-D conversion.

(5) fC32

This clock is derived by dividing the sub-clock by 32. It is used for the timer 1, timer X, timer Y and timer Z counts. Figure 1.8.6 shows the block diagram of fc32 .

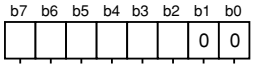
(6) fC

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

(7) fRING

This clock is supplied by the ring oscillator circuit. In the ring oscillator mode, the clock divided by the division ratio selected with the main clock division select bit 0 and bit 1(bit 6 at address 0006$_{16}$, and bit 6 and bit 7 at address 0007$_{16}$) is supplied as BCLK. Immediately after reset, 8 divisions of this clock is supplied as BCLK. The ring oscillator oscillation can be set to BCLK when oscillation stop is detected or with the main clock switching bit (bit 2 at address 000C$_{16}$).

System clock control register 0 (Note 1)

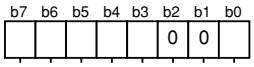


Symbol: CM0 — Address: 000616 — When reset: 4816

Bit symbol	Bit name	Function	R	W
Reserved bit		Always set to "0"	○	○
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	○	○
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 shifting to stop mode.
Note 3: This bit is used to stop the main clock when placing the device in a low-power mode. If you want to operate with XIN after exiting from the stop mode, set this bit to "0". When operating with a self-excited oscillator, set the 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 ON, so XIN turns pulled up to XOUT ("H") via the feedback resistor.
Note 6: Set port Xc select bit (CM04) to "1" before setting this bit to "1". Can not write to both bits at the same time.
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, low power dissipation or ring oscillator mode.

System clock control register 1 (Note 1)



Symbol: CM1 — Address: 000716 — When reset: 2016

Bit symbol	Bit name	Function	R	W
CM10	All clock stop control bit (Note 4)	0 : Clock on 1 : All clocks off (stop mode)	○	○
Reserved bit		Always set to "0"	○	○
Reserved bit		Always set to "0"	○	○
CM13	XIN oscillation select bit	0 : Ceramic oscillation or crystal oscillation 1 : RC oscillation	○	○
CM14	Ring oscillation stop bit	0 : Oscillation enabled 1 : Oscillation stopped (Note 5)	○	○
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/middle-speed mode to stop mode or at 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 ineffective. The mode of power control cannot be shifted to the stop mode directly from the oscillator mode.
Note 5: This bit can be set to "1" only when both the main clock switch bit (CM22) and clock monitor bit (CM23) are set to "0" . Moreover, this bit is automatically set to "0" if the main clock switch bit (CM22) is set to "1".

Figure 1.8.4. System clock control registers 0 and 1

Oscillation stop detection register (Note 5)

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

Symbol: CM2 | Address: 000C16 | When reset: 0416

Bit symbol	Bit name	Function	R	W
CM20	Oscillation stop detection bit	0: The function is invalid. 1: The function is valid. (Note 1)	○	○
CM21	Oscillation stop detection interrupt enable bit	0: Disabled 1: Enabled (Note 2)	○	○
CM22	Main clock switch bit	0: Select XIN clock. 1: Select ring oscillator. (Note 3)	○	○
CM23	Clock monitor bit (Note 4)	0: XIN oscillating normally 1: XIN stopping abnormally	○	×
Reserved bit		Always set to "0"	○	○

Note 1: Set to “0” before stopping the oscillation of the main clock (XIN-XOUT).
(stop mode, low power dissipation mode, ring oscillation mode)
An oscillation stop is detected if the oscillation of the main clock (XIN-XOUT) is stopped when the following two conditions are satisfied: (1) the oscillation stop detection function is valid and (2) CM21=1.
Note 2: Valid when CM20=1.
Note 3: CM22 bit switches to “1“ automatically if an oscillation stop is detected when both CM20 bit and CM 21 bit are “1”. CM22 bit cannot be cleared when CM23=1.
Note 4: This bit is valid when CM20 bit is “1”. Use this bit for the purpose of confirming XIN operation for oscillation stop detection interrupt execution.
Note 5: In case of writing to this register, set bit 0 of the protect register(000A16) to "1".

Clock prescaler reset flag

b7	b6	b5	b4	b3	b2	b1	b0
	✕	✕	✕	✕	✕	✕	✕

Symbol: CPSRF | Address: 008F16 | When reset: 0XXXXXXX2

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 1.8.5. Oscillation stop detection register and clock prescaler reset flag

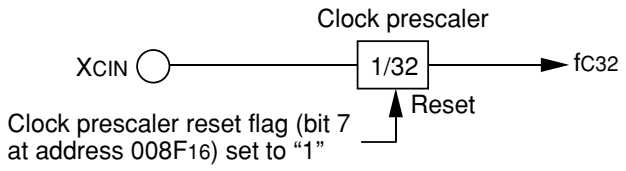


Figure 1.8.6. fC32 block diagram

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 of BCLK, f_1 to f_{32}, fc, fc_{32}, and f_{AD} stops in stop mode, peripheral functions such as the A-D converter and watchdog timer do not function. However, timer X operate provided that the event counter mode is set to an external pulse, and UART0 and UART1 function provided an external clock is selected. Table 1.8.2 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 before shifting to stop mode. If returning by an interrupt, that interrupt routine is executed. If only a hardware reset 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 or 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.

The stop mode must not be used while operating in the ring oscillator mode.

Table 1.8.2. Port status during stop mode

Pin	States
Port	Retains status before stop mode

Wait Mode

When a WAIT instruction is executed, BCLK stops and the microcomputer enters the wait mode. In this mode, oscillation continues but 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. Table 1.8.3 shows the status of the ports in wait mode.

Wait mode is cancelled by a hardware reset or interrupt. If an interrupt is used to cancel wait mode, the microcomputer restarts using as BCLK, the clock that had been selected when the WAIT instruction was executed.

Table 1.8.3. Port status during wait mode

Pin	States
Port	Retains status before wait mode

Status Transition of BCLK

Power dissipation can be reduced and low-voltage operation achieved by changing the count source for BCLK. Table 1.8.4 shows the operating modes corresponding to the settings of system clock control registers 0 and 1.

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

(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. 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 dissipation mode, 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. When using an external RC circuit for the main clock, no-division mode must not be used.

(6) Low-speed mode

f_C is used as 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.

(8) Ring oscillator mode

This mode sets the ring oscillator as BCLK. The same as when X_{IN} is the main clock, the modes are no division, 2-division, 4-division, 8-division, and 16-division.

Note: Before the count source for BCLK can be changed from X_{IN} to XC_{IN} 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.

And, be sure to shift from division by 8 mode when you change it to ring oscillator mode. Shift to other mode after you surely shift to the mode for division by 8 mode when you change it from ring oscillator mode to other mode.

Table 1.8.4. Operating modes dictated by settings of system clock control registers 0 and 1

CM22	CM17	CM16	CM07	CM06	CM05	CM04	Operating mode of BCLK
0	0	1	0	0	0	Invalid	Division by 2 mode
0	1	0	0	0	0	Invalid	Division by 4 mode
0	Invalid	Invalid	0	1	0	Invalid	Division by 8 mode
0	1	1	0	0	0	Invalid	Division by 16 mode
0	0	0	0	0	0	Invalid	No-division mode
0	Invalid	Invalid	1	Invalid	0	1	Low-speed mode
0	Invalid	Invalid	1	Invalid	1	1	Low power dissipation mode
1	0	1	0	0	Invalid	Invalid	Ring oscillator mode(divided by 2)
1	1	0	0	0	Invalid	Invalid	Ring oscillator mode(divided by 4)
1	Invalid	Invalid	0	1	Invalid	Invalid	Ring oscillator mode(divided by 8)
1	1	1	0	0	Invalid	Invalid	Ring oscillator mode(divided by 16)
1	0	0	0	0	Invalid	Invalid	Ring oscillator mode(no division)

Power Control

This section gives an overview of power control.

Modes

There are three power save modes.

(1) Normal operating mode

• High-speed mode

In this mode, one main clock cycle forms BCLK. The CPU operates on the BCLK. The peripheral functions operate on the clocks specified for each respective function.

• Medium-speed mode

In this mode, the main clock is divided into 2, 4, 8, or 16 to form BCLK. The CPU operates on the BCLK. The peripheral functions operated on the clocks specified for each respective function.

• Low-speed mode

In this mode, fc forms BCLK. The CPU operates on the fc clock. fc is the clock supplied by the subclock. The peripheral functions operate on the clocks specified for each respective function.

• Low power-dissipation mode

This mode is selected when the main clock is stopped from low-speed mode. The CPU operates on the fc clock. fc is the clock supplied by the subclock. Only the peripheral functions for which the subclock was selected as the count source continue to run.

• Ring oscillator mode

This mode sets the ring oscillator as BCLK. The ring oscillator can be set to no division, 2-divisions, 4-division, 8-division, or 16-division mode according to the settings for CM06, CM16, and CM17. Increasing the division ratio lowers power consumption. When the microcomputer is operating with the ring oscillator, the XIN clock driver can be stopped by setting the main clock stop bit to “1.” This can lower the power dissipation even more.

(2) Wait mode

CPU operation is halted in this mode. The oscillator continues to run.

(3) Stop mode

All oscillators stop in this mode. The CPU and internal peripheral functions all stop. Of all 3 power saving modes, power savings are greatest in this mode. The mode cannot be shifted to the stop mode directly from the ring oscillator mode.

Figure 1.9.1 and 1.9.2 show the transition between each of the three modes, (1), (2), and (3).

Transition of stop mode, wait mode

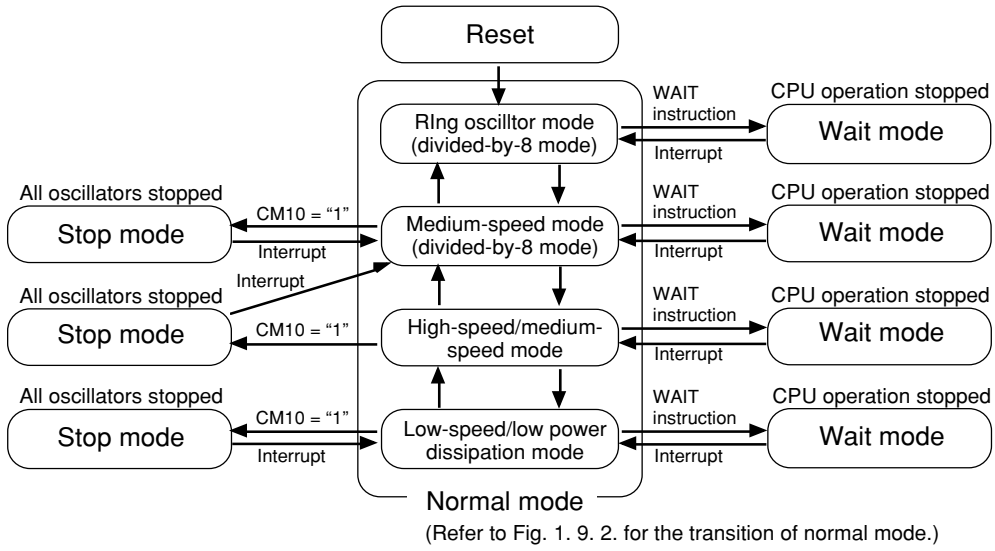


Figure 1.9.1. Clock transition (1)

Transition of normal mode

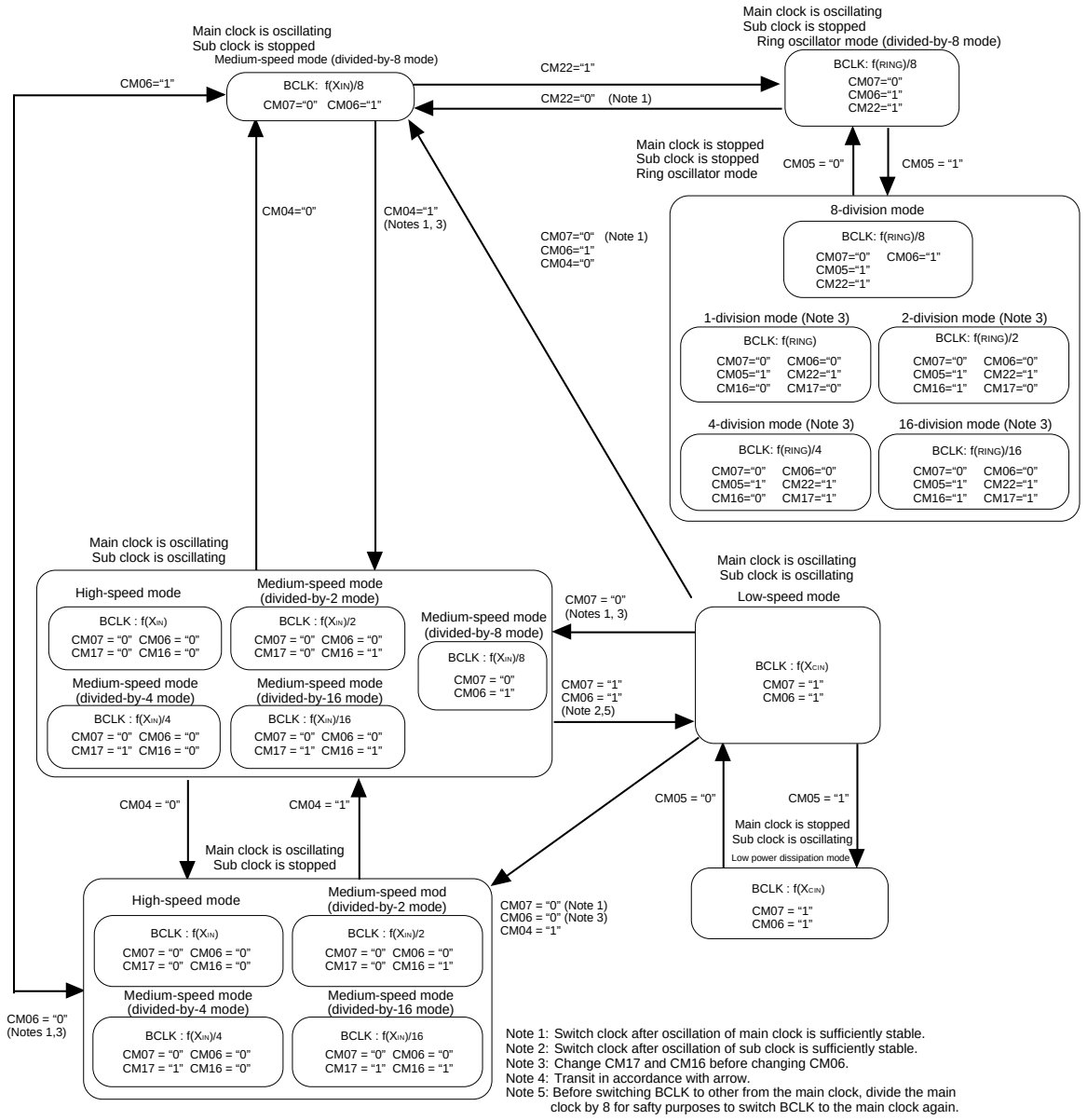


Figure 1.9.2. Clock transition (2)

Oscillation Stop Detection Function

The oscillation stop detection function detects abnormal stopping of the main clock by causes such as opening and shorting of the XIN oscillation circuit. When oscillation stop is detected, an oscillation stop detection interrupt is issued. When an oscillation stop detection interrupt is issued, the ring oscillator in the microcomputer operates automatically and is used as the main clock in place of the XIN clock. This allows interrupt processing.

The oscillation stop detection function can be enabled/disabled with bit 0 and bit 1 of the oscillation stop detection register. When this bit is set to "11_2," the function is enabled. After the reset is released, the oscillation stop detection function becomes disabled because the bit value is "00_2."

Table 1.10.1 gives an specification overview of the oscillation stop detection function, Figure 1.10.2 is a configuration diagram of the oscillation stop detection circuit and Figure 1.10.3 shows the configuration of the oscillation stop detection register.

Table 1.10.1. Specification overview of the oscillation stop detection function

Item	Specification
Oscillation stop detectable clock and frequency bandwidth	XIN ≥ 2 MHz
Enabling condition for oscillation stop detection function	When the oscillation stop detection bit (bit 0 of address $000C_{16}$) and the oscillation stop detection interrupt enable bit (bit 1 of address $000C_{16}$) are set to "1"
Operation at oscillation stop detection	• Oscillation stop detection interrupt occurs
Notes on STOP mode, low power dissipation mode, and ring oscillator mode	Before stopping the main clock (XIN-XOUT), set the oscillation stop detection enable bit to "0" to disable the oscillation stop detection function. Enable main clock (XIN-XOUT) oscillation and after the oscillation stabilizes, set the bit to "1" again.
Notes on WAIT mode	If the peripheral function clock is stopped in WAIT mode with the WAIT mode peripheral function clock stop bit (bit 2 of the address 0006_{16}), oscillation stop will be detected. Do not stop the peripheral function clock in WAIT mode.

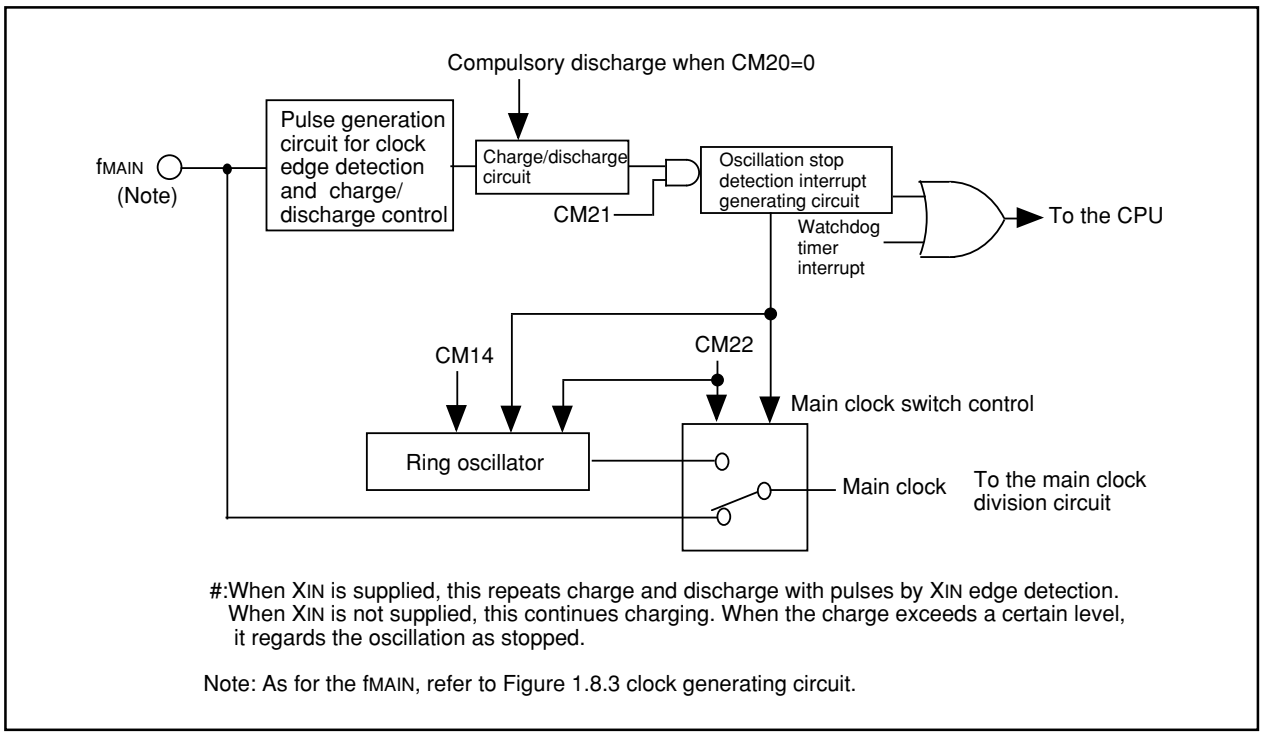


Figure 1.10.1. Oscillation stop detection circuit

Oscillation stop detection register (Note 5)

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

Symbol CM2 | Address 000C16 | When reset 0416

Bit symbol	Bit name	Function	R	W
CM20	Oscillation stop detection bit	0: The function is invalid. 1: The function is valid. (Note 1)	○	○
CM21	Oscillation stop detection interrupt enable bit	0: Disabled 1: Enabled (Note 2)	○	○
CM22	Main clock switch bit	0: Select XIN clock. 1: Select ring oscillator. (Note 3)	○	○
CM23	Clock monitor bit (Note 4)	0: XIN oscillating normally 1: XIN stopping abnormally	○	×
Reserved bit		Always set to "0"	○	○

Note 1: Set to “0” before stopping the oscillation of the main clock (XIN-XOUT).
(stop mode, low power dissipation mode, ring oscillation mode)
An oscillation stop is detected if the oscillation of the main clock (XIN-XOUT) is stopped when the following two conditions are satisfied: (1) the oscillation stop detection function is valid and (2) CM21=1.
Note 2: Valid when CM20=1.
Note 3: CM22 bit switches to “1“ automatically if an oscillation stop is detected when both CM20 bit and CM 21 bit are “1”. CM22 bit cannot be cleared when CM23=1.
Note 4: This bit is valid when CM20 bit is “1”. Use this bit for the purpose of confirming XIN operation for oscillation stop detection interrupt execution.
Note 5: In case of writing to this register, set bit 0 of the protect register(000A16) to "1".

Figure 1.10.2. Oscillation stop detection register

Oscillation stop detection bit (CM20)

You can start the oscillation stop detection by setting this bit to "1" and CM21=1 (oscillation stop detection interrupt enabled). The detection is not executed when this bit is set to "0" or in reset status. Be sure to set this bit to "0" before setting for the stop-mode. Set this bit again to "1" after release from stop-mode. Set this bit to "0" also before setting the main clock stop bit (bit 5 at 0006_{16}) to "1".

Do not set this bit to "1" if the frequency of XIN is lower than 2 MHz.

An oscillation stop is detected if CM02="1" (peripheral function clock has been set for stop in wait mode) and the mode is shifted to wait.

Oscillation stop detection interrupt enable bit (CM21)

When CM20=1 and CM21=1, an oscillation stop detection interrupt is generated if an abnormal stop of XIN is detected. The ring oscillator starts operation instead of the XIN clock which stopped abnormally. The operation goes further with the main clock supplied from the ring oscillator. For the oscillation stop detection interrupt, judgment on the interrupt condition is necessary, because this interrupt shares the vector table with watchdog timer interrupt. Figure 1.10.3 shows flow of the judgment with oscillation stop detection interrupt processing program.

Main clock switch bit (CM22)

When setting this bit to "1", the ring oscillator is selected as main clock. At this time, the ring oscillator starts simultaneously if it has been stopped (CM14=1). This bit is cleared only when CM23 is "0" (when XIN is oscillating).

If an oscillation stop is detected while both CM20 and CM21 are "1", this bit automatically switches to "1". When this bit is set to "1", the ring oscillation stop bit (bit 4 of address 0007_{16}) is automatically set to "0".

Clock monitor bit (CM23)

You can see the operation status of the XIN clock. When this bit is "0", XIN is operating correctly. You can check the oscillation status of XIN when an oscillation stop detection interrupt is generated or after reset. When oscillation stop detection is invalid (CM20="0"), the clock monitor bit is "0".

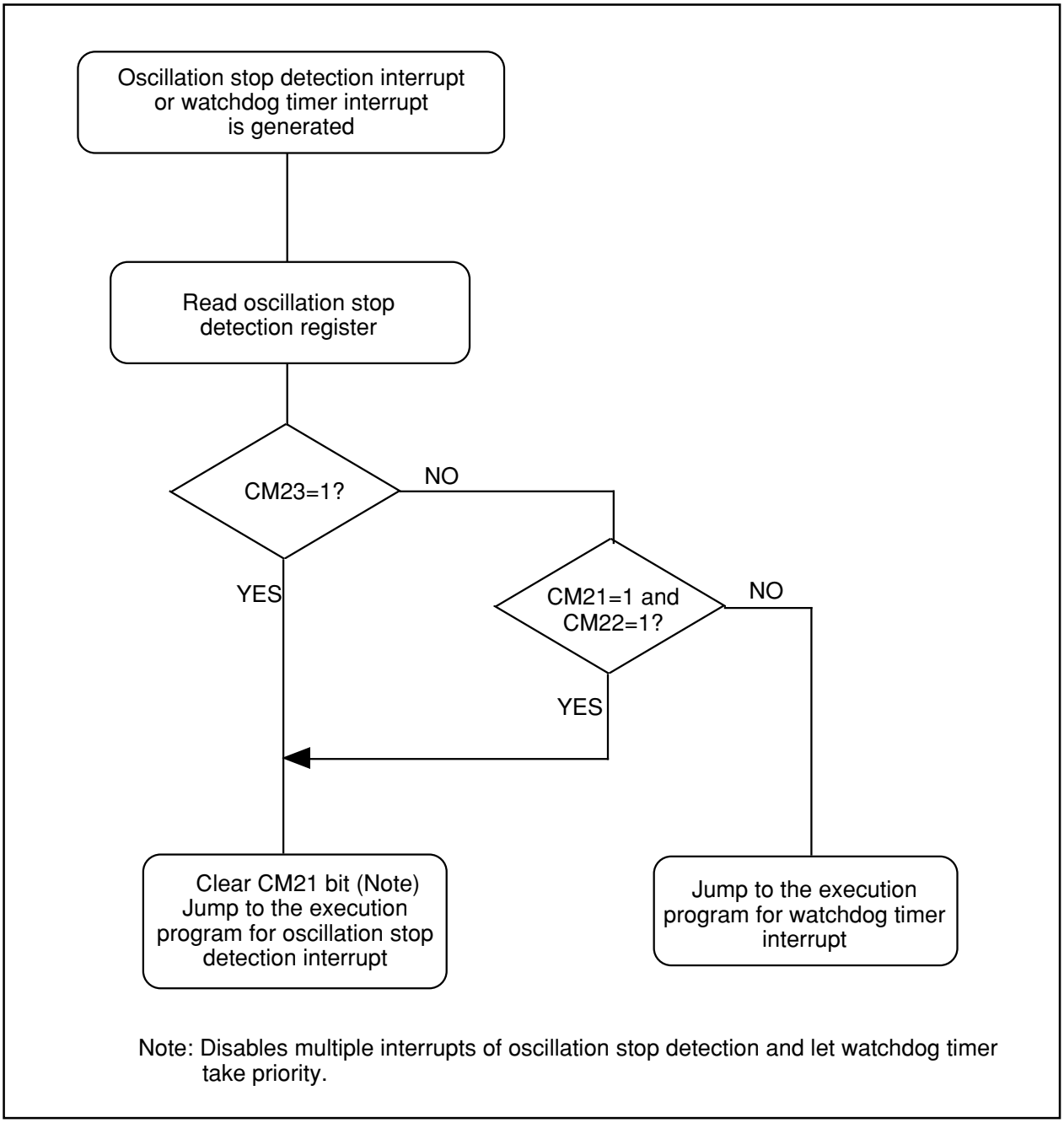


Figure 1.10.3. Flow of the judgment

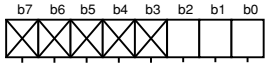
Protection

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 1.11.1 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}) and port P0 direction register (address $00E2_{16}$) can only be changed when the respective bit in the protect register is set to “1”. Therefore, important outputs can be allocated to port P0.

If, after "1" (write-enabled) has been written to bit "enables writing to port P0 direction register" (bit 2 at address $000A_{16}$), a value is written to any address, the bit automatically reverts to "0" (write-inhibited).

The system clock control registers 0 and 1 and oscillation stop detection register 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



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}) and oscillation stop detection register (address $000C_{16}$)	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 P0 direction register (address $00E2_{16}$) (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 1.11.1. Protect register

Overview of Interrupt

Type of Interrupts

Figure 1.12.1 lists the types of interrupts.

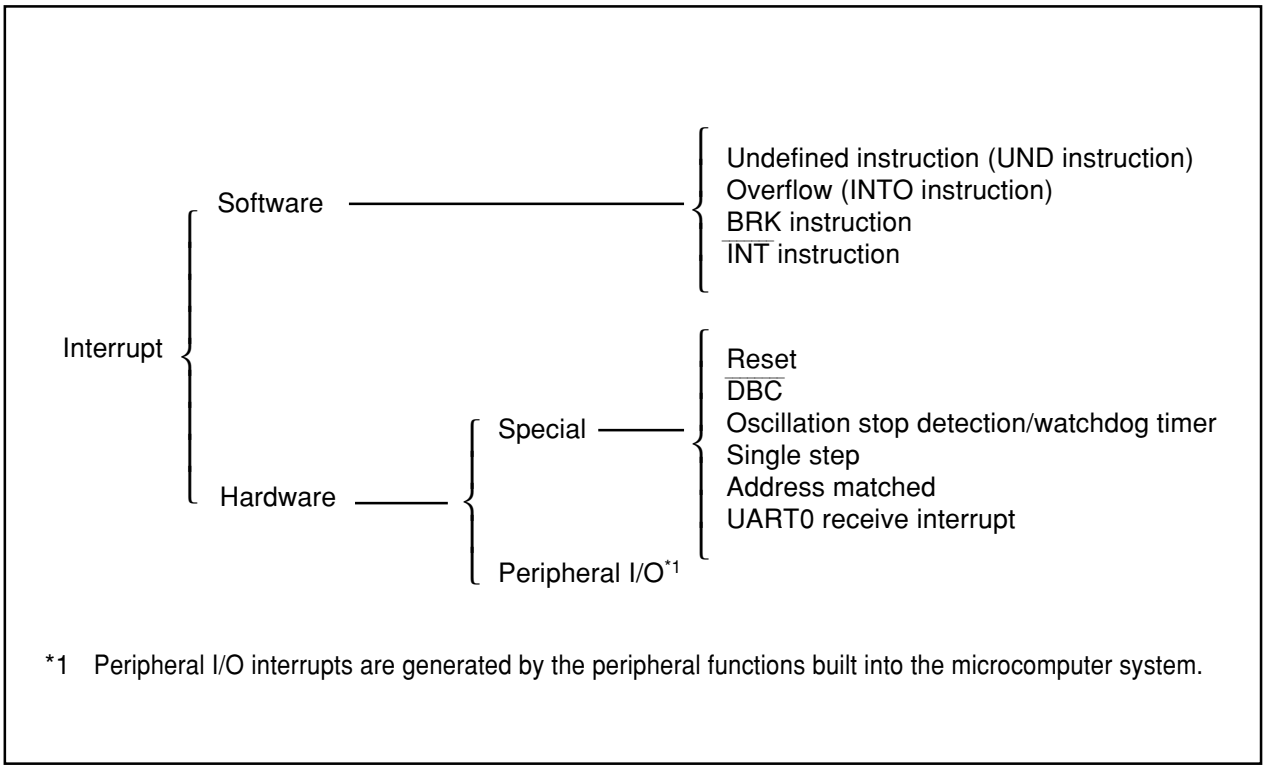


Figure 1.12.1. Classification of interrupts

- 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.

• $\overline{\text{INT}}$ interrupt

An $\overline{\text{INT}}$ interrupt occurs when assigning one of software interrupt numbers 0 through 63 and executing the $\overline{\text{INT}}$ instruction. Software interrupt numbers 0 through 31 are assigned to peripheral I/O interrupts, so executing the $\overline{\text{INT}}$ instruction allows executing the same interrupt routine that a peripheral I/O interrupt does.
The stack pointer (SP) used for the $\overline{\text{INT}}$ 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.
- **UART0 receive interrupt**
  UART0 receive interrupt occurs when UART1 is received. This interrupt can be enabled with bit 2 of the $\overline{\text{INT}}$0 input filter select register (address 001E16).
  This interrupt is exclusively for the debugger, do not use it in other circumstances.
- **$\overline{\text{DBC}}$ interrupt**
  This interrupt is exclusively for the debugger, do not use it in other circumstances.
- **Oscillation stop detection/watchdog timer interrupt**
  Generated by the oscillation stop detection or 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. The interrupt vector table is the same as the one for software interrupt numbers 0 through 31 the $\overline{\text{INT}}$ instruction uses. Peripheral I/O interrupts are maskable interrupts.

- **Key-input interrupt**
  A key-input interrupt occurs if a falling or rising edge is input to the $\overline{\text{KI}}$ pin.
- **A-D conversion interrupt**
  This is an interrupt that the A-D converter generates.
- **UART0 and UART1 transmission interrupt**
  These are interrupts that the serial I/O transmission generates.
- **UART0 and UART1 reception interrupt**
  These are interrupts that the serial I/O reception generates.
- **Timer X interrupt**
  This is an interrupts that timer X generates.
- **Timer Y interrupt**
  This is an interrupt that timer Y generates.
- **Timer Z interrupt**
  This is an interrupt that timer Z generates.
- **Timer C interrupt**
  This is an interrupt that timer C generates.
- **CNTR0 interrupt**
  This interrupt occurs if a falling or rising edge is input to the CNTR0 pin.
- **TCIN interrupt**
  This interrupt occurs if a falling edge, rising edge or both edges are input to the TCIN pin. This interrupt also occurs with the RING512.
- **$\overline{\text{INT}}$0 to $\overline{\text{INT}}$3 interrupt**
  $\overline{\text{INT}}$0 to $\overline{\text{INT}}$2 interrupts occur if any one of a rising edge, a falling edge or a both-edge is input to the $\overline{\text{INT}}$ pin. $\overline{\text{INT}}$3 inerrupt occurs if either 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 1.12.2 shows format for specifying interrupt vector addresses.

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.

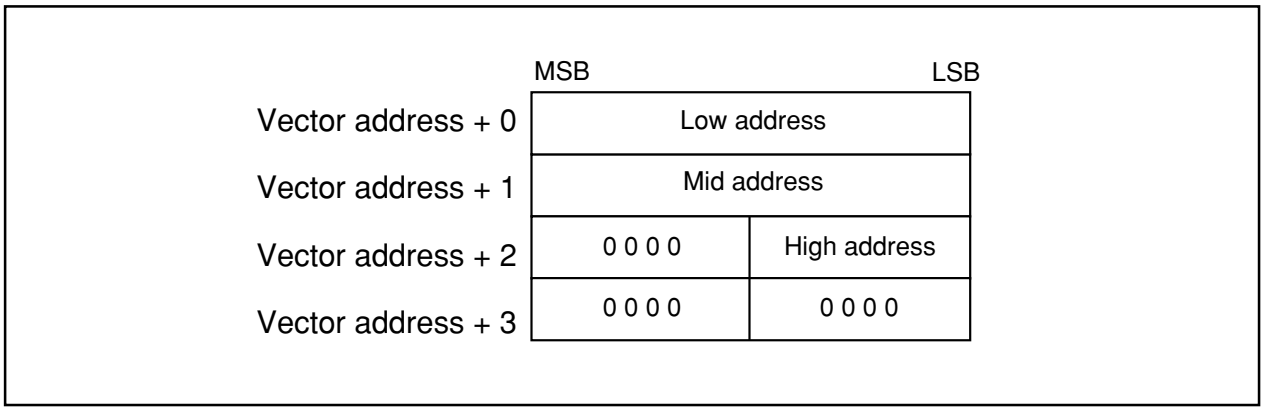


Figure 1.12.2. Format for specifying interrupt vector addresses

• 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 1.12.1 shows the interrupts assigned to the fixed vector tables and addresses of vector tables.

Table 1.12.1. Interrupt and fixed vector address

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 is filled with 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
Oscillation stop detection/ watchdog timer	FFFF016 to FFFF316	
$\overline{\text{DBC}}$ (Note)	FFFF416 to FFFF716	Do not use
UART0 receive (Note)	FFFF816 to FFFFB16	Do not use
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 1.12.2 shows the interrupts assigned to the variable vector tables and addresses of vector tables.

Table 1.12.2. Interrupt causes (variable interrupt vector addresses)

Software interrupt number	Vector table address Address (L) to address (H)	Interrupt source	Remarks
Software interrupt number 0	+0 to +3 (Note)	BRK instruction	Cannot be masked by I flag
Software interrupt number 13	+52 to +55 (Note)	Key input interrupt	
Software interrupt number 14	+56 to +59 (Note)	A-D	
Software interrupt number 17	+68 to +71 (Note)	UART0 transmit	
Software interrupt number 18	+72 to +75 (Note)	UART0 receive	
Software interrupt number 19	+76 to +79 (Note)	UART1 transmit	
Software interrupt number 20	+80 to +83 (Note)	UART1 receive	
Software interrupt number 21	+84 to +87 (Note)	Timer 1	
Software interrupt number 22	+88 to +91 (Note)	Timer X	
Software interrupt number 23	+92 to +95 (Note)	Timer Y	
Software interrupt number 24	+96 to +99 (Note)	Timer Z	
Software interrupt number 25	+100 to +103 (Note)	CNTR0	
Software interrupt number 26	+104 to +107 (Note)	TCIN	
Software interrupt number 27	+108 to +111 (Note)	Timer C	
Software interrupt number 28	+112 to +115 (Note)	$\overline{\text{INT3}}$	
Software interrupt number 29	+116 to +119 (Note)	$\overline{\text{INT0}}$	
Software interrupt number 30	+120 to +123 (Note)	$\overline{\text{INT1}}$	
Software interrupt number 31	+124 to +127 (Note)	$\overline{\text{INT2}}$	
Software interrupt number 32 to Software interrupt number 63	+128 to +131 (Note) to +252 to +255 (Note)	Software interrupt	Cannot be masked by I flag

Note : Address relative to address in interrupt table register (INTB).

Interrupt Control

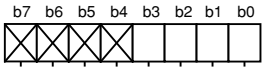
Descriptions are given here regarding how to enable or disable maskable interrupts and how to set the priority to be accepted. What is described here does not apply to non-maskable interrupts.

Enable or disable a maskable interrupt using the interrupt enable flag (I flag), interrupt priority level select bit, and 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 1.12.3 shows the interrupt control registers.

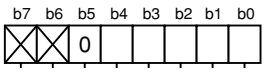
Interrupt control register

Symbol	Address	When reset
KUPIC	$004D_{16}$	$XXXXX000_2$
ADIC	$004E_{16}$	$XXXXX000_2$
SiTIC(i=0, 1)	0051_{16}, 0053_{16}	$XXXXX000_2$
SiRIC(i=0, 1)	0052_{16}, 0054_{16}	$XXXXX000_2$
T1IC	0055_{16}	$XXXXX000_2$
TXIC	0056_{16}	$XXXXX000_2$
TYIC	0057_{16}	$XXXXX000_2$
TZIC	0058_{16}	$XXXXX000_2$
CNTR0IC	0059_{16}	$XXXXX000_2$
TCINIC	$005A_{16}$	$XXXXX000_2$
TCIC	$005B_{16}$	$XXXXX000_2$
INT3IC	$005C_{16}$	$XXXXX000_2$



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 dose not generate the interrupt request for that register. For details, see the precautions for interrupts.



Symbol	Address	When reset
INTiIC(i=0, 1, 2)	$005D_{16}$, $005E_{16}$	$XX00X000_2$
	$005F_{16}$	$XX00X000_2$

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		Always 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 dose not generate the interrupt request for that register. For details, see the precautions for interrupts.

Figure 1.12.3. Interrupt control register

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 1.12.3 shows the settings of interrupt priority levels and Table 1.12.4 shows the interrupt levels enabled, according to the consist 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 1.12.3. 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 1.12.4. 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 T1IC int. priority level and int. request bit.
    NOP                   ;
    NOP
    FSET    I             ; Enable interrupts.
```

Example 2:

```
INT_SWITCH2:
    FCLR    I             ; Disable interrupts.
    AND.B   #00h, 0055h   ; Clear T1IC 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 T1IC int. priority level and int. request bit.
    POPC    FLG           ; Enable interrupts.
```

The reason why two NOP instructions 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

Changing the interrupt request bit

When attempting to clear the interrupt request bit of an interrupt control register, the interrupt request bit is not cleared sometimes. This will depend on the instruction. If this creates problems, use the below instructions to change the register.
Instructions : MOV

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 00000_{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 $\overline{\text{INT}}$ 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 1.12.4 shows the interrupt response time.

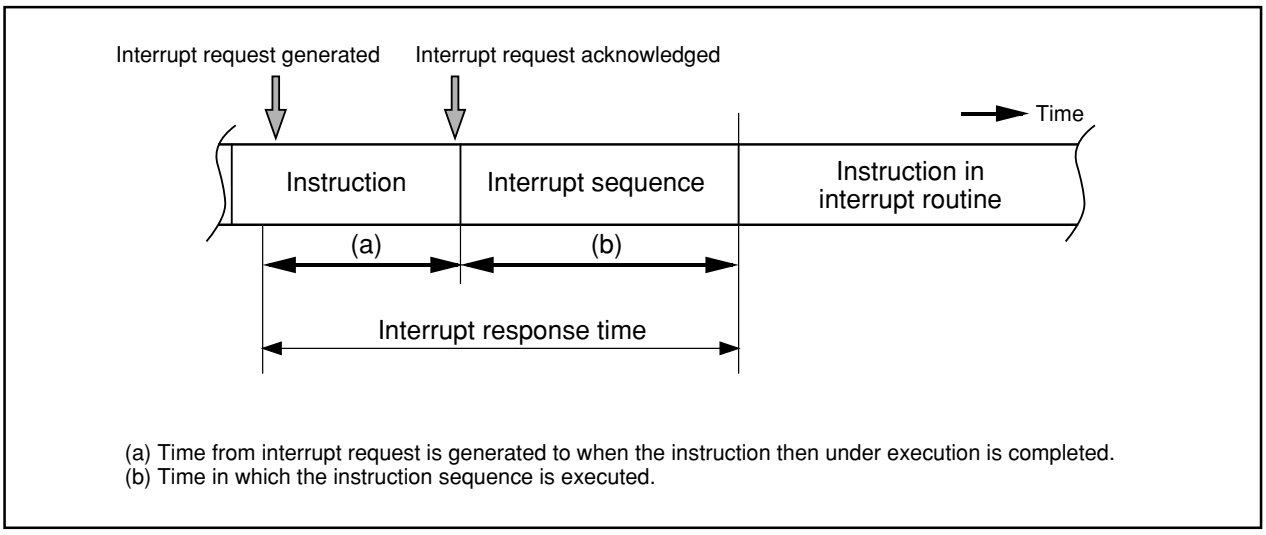


Figure 1.12.4. Interrupt response time

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 1.12.5.

Table 1.12.5. Time required for executing the interrupt sequence

Interrupt vector address	Stack pointer (SP) value	Without wait
Even	Even	18 cycles (Note 1)
Even	Odd	19 cycles (Note 1)
Odd (Note 2)	Even	19 cycles (Note 1)
Odd (Note 2)	Odd	20 cycles (Note 1)

Note 1: Add 2 cycles in the case of a $\overline{DBC}$ 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.

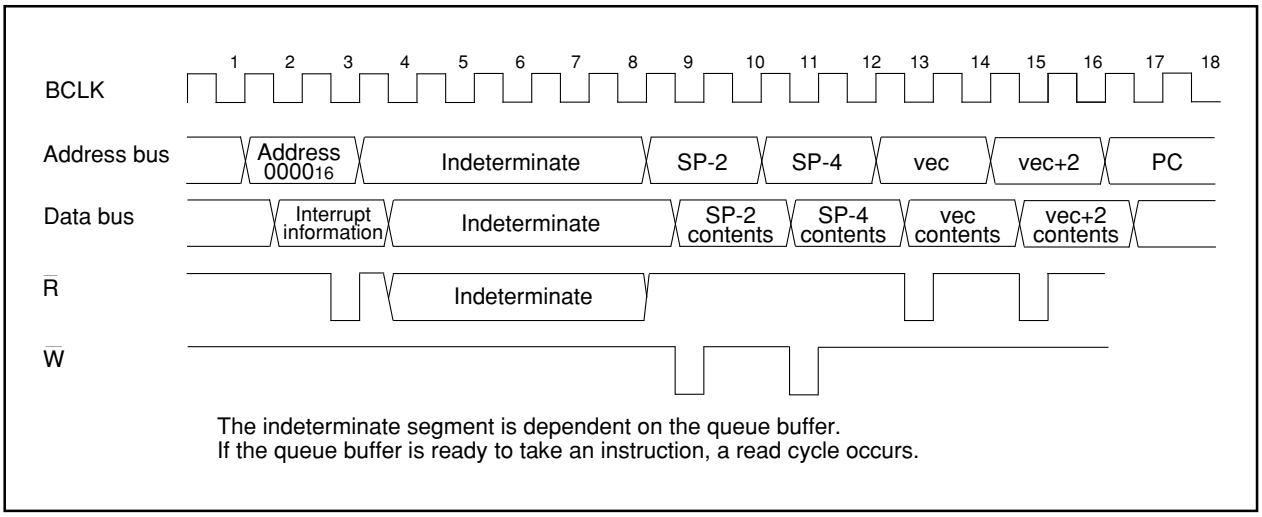


Figure 1.12.5. Time required for executing the interrupt sequence

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 1.12.6 is set in the IPL.

Table 1.12.6. Relationship between interrupts without interrupt priority levels and IPL

Interrupt sources without priority levels	Value set in the IPL
Watchdog timer	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 4 high-order bits of the program counter, and 4 high-order bits and 8 low-order bits of the FLG register, 16 bits in total, in the stack area, then saves 16 low-order bits of the program counter. Figure 1.12.6 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).

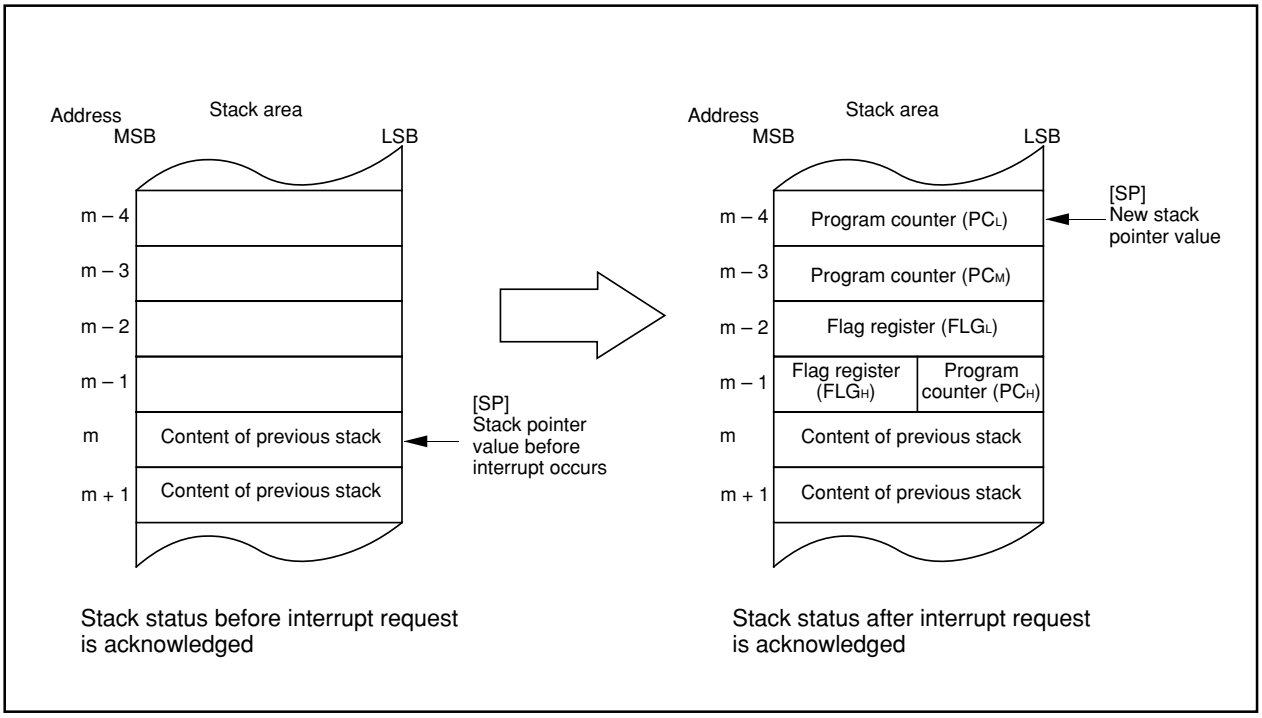


Figure 1.12.6. State of stack before and after acceptance of interrupt request

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 1.12.7 shows the operation of the saving registers.
Note: This is the stack pointer indicated by the U flag.

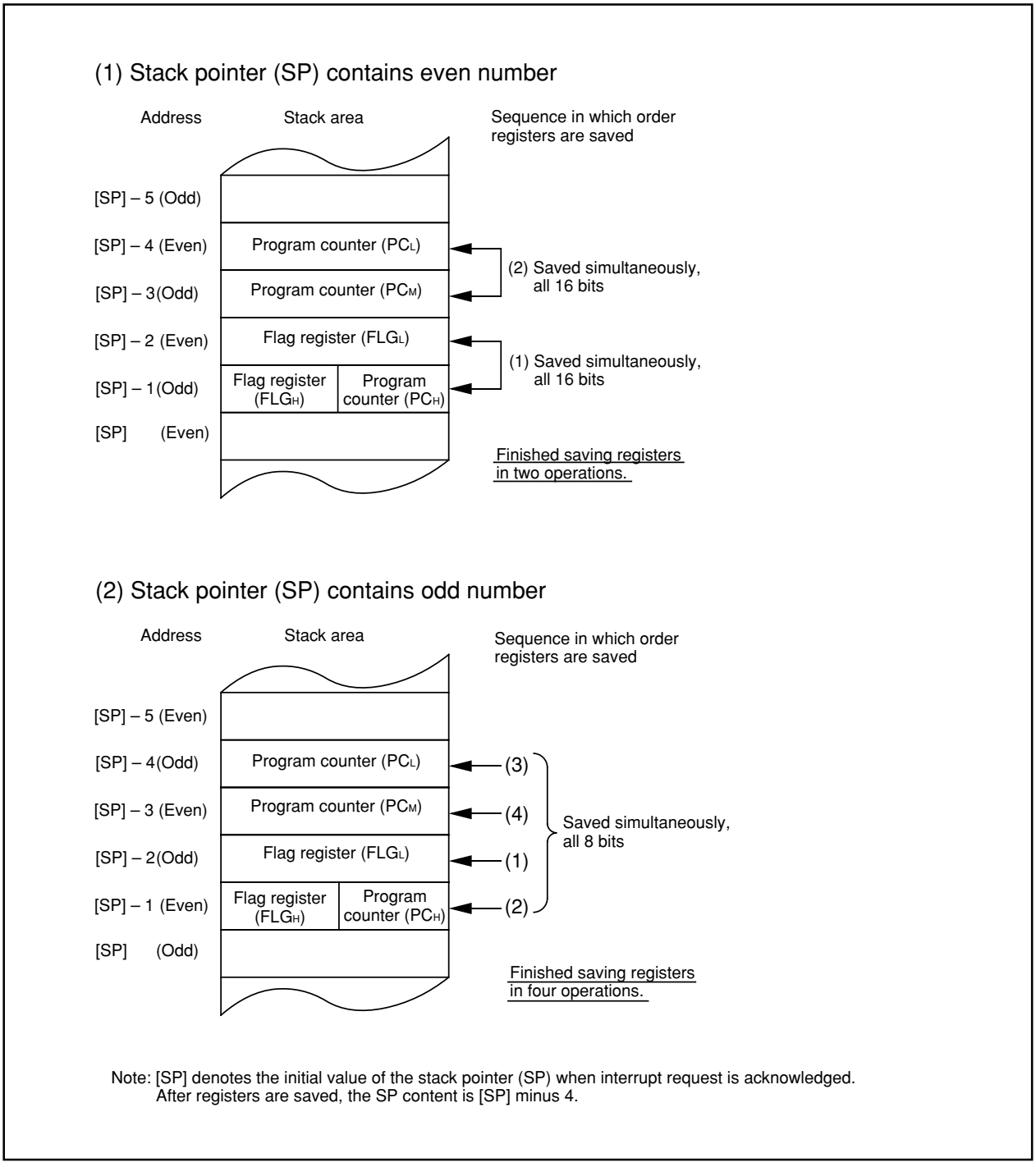


Figure 1.12.7. Operation of saving registers

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 1.12.8 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.

Interrupt Priority Level Judge Circuit

This circuit selects the interrupt with the highest priority level when two or more interrupts are generated simultaneously.

Figure 1.12.9 shows the interrupt resolution circuit.

Reset > UART0 receive > $\overline{\text{DBC}}$ > Oscillation stop detection/watchdog timer > Peripheral I/O > Single step > Address match

Figure 1.12.8. Hardware interrupts priorities

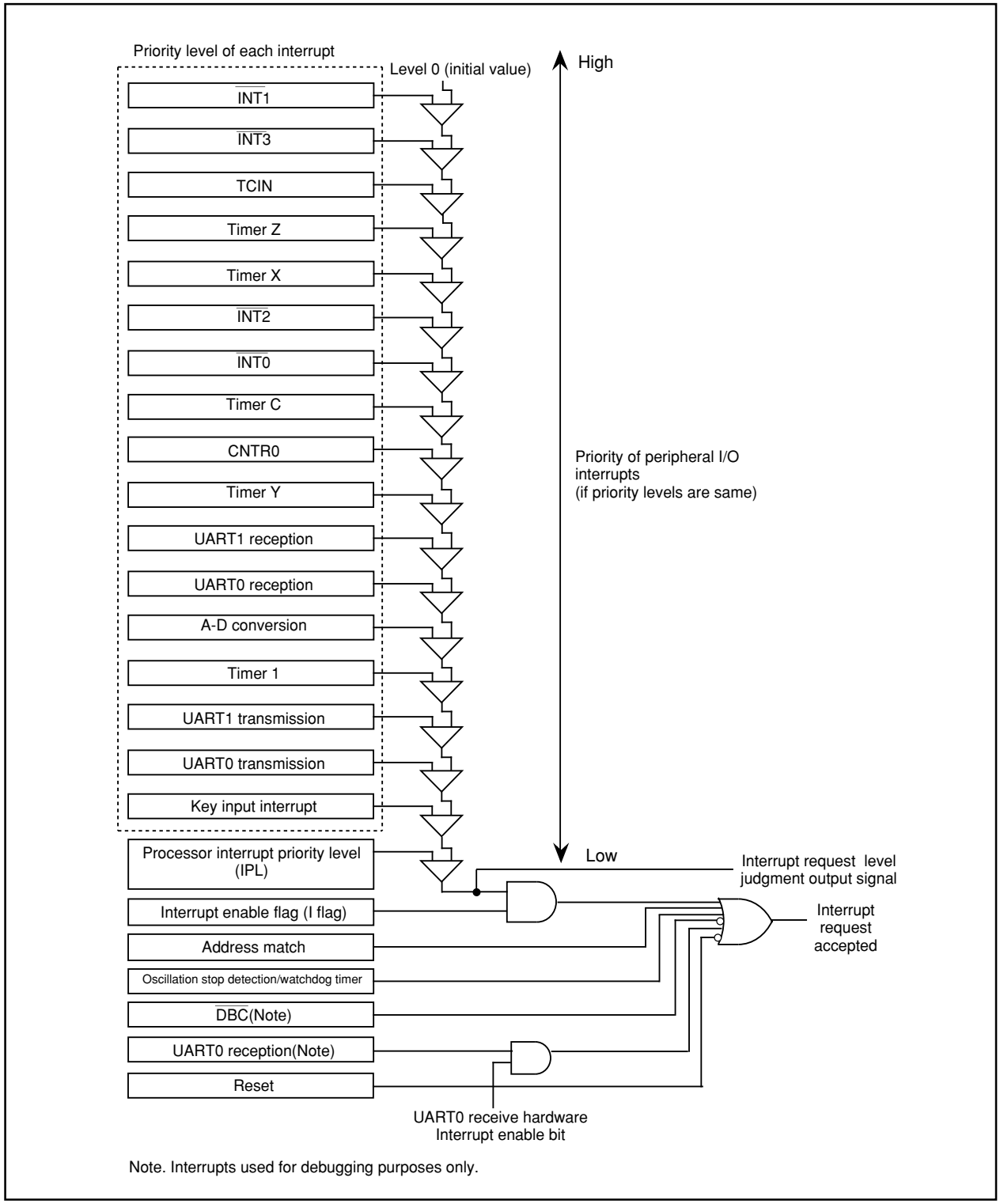


Figure 1.12.9. Interrupt resolution circuit

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

$\overline{\text{INT}}$0 to $\overline{\text{INT}}$3 are triggered by the edges of external inputs. The edge polarity of $\overline{\text{INT}}$0 to $\overline{\text{INT}}$2 is selected using the polarity select bit (bit 4 of addresses 005D16, 005E16 and 005F16). Input to INT0 is available via filter with three different sampling frequencies.

As to external interrupt input, an interrupt can be generated both at the rising edge and at the falling edge by setting the $\overline{\text{INT}}$i (i=0 to 3) input polarity select bit of the external input enable register (009616) to "1". To select both edges, set the polarity switching bit of the corresponding interrupt control register to "0" (falling edge). To select one edge, set the polarity switching bit of the corresponding interrupt control register to either "1" (raising edge) or "0" (falling edge). Please note that when one edge is selected using $\overline{\text{INT}}$3, the polarity will be a falling edge.

After setting the external input enable register, clear the interrupt request bit, and then enable the corresponding input interrupt. Moreover, you should write to the external input enable bit only under conditions where the corresponding input interrupt is disabled.

Figure 1.12.10 shows the external input related registers.

External input enable register

b7 b6 b5 b4 b3 b2 b1 b0

Symbol	Address	When reset
INTEN	009616	0016

Bit symbol	Bit name	Function	R	W
INT0EN	$\overline{\text{INT}}$0 input enable bit (Note)	0 : Disabled 1 : Enabled	○	○
INT0PL	$\overline{\text{INT}}$0 input polarity select bit (Note)	0 : One edge 1 : Two edges	○	○
INT1EN	$\overline{\text{INT}}$1 input enable bit	0 : Disabled 1 : Enabled	○	○
INT1PL	$\overline{\text{INT}}$1 input polarity select bit	0 : One edge 1 : Two edges	○	○
INT2EN	$\overline{\text{INT}}$2 input enable bit	0 : Disabled 1 : Enabled	○	○
INT2PL	$\overline{\text{INT}}$2 input polarity select bit	0 : One edge 1 : Two edges	○	○
INT3EN	$\overline{\text{INT}}$3 input enable bit	0 : Disabled 1 : Enabled	○	○
INT3PL	$\overline{\text{INT}}$3 input polarity select bit	0 : One edge 1 : Two edges	○	○

Note : This bit must be set in condition of INT0 pin one-shot trigger invalid (INOSTG="0").

$\overline{\text{INT}}$0 input filter select register

b7 b6 b5 b4 b3 b2 b1 b0

Symbol	Address	When reset
INT0F	001E16	XXXXX0002

Bit symbol	Bit name	Function	R	W
INT0F0 INT0F1	$\overline{\text{INT}}$0 input filter select bit	b1 b0 0 0 : No filter 0 1 : Filter with f1 sampling 1 0 : Filter with f8 sampling 1 1 : Filter with f32 sampling	○	○
INT0F2	UART0 receive hardware interrupt enable bit (Note)	0 : Disabled 1 : Enabled	○	○
Nothing is assigned. In an attempt to write to these bits, write "0". The value, if read, turns out to be indeterminate.			–	–

Note: Interrupts used for debugging purposes only.

Figure 1.12.10. External input related registers

$\overline{\text{INT0}}$ Input Filter

The $\overline{\text{INT0}}$ input has a digital filter which can be sampled by one of three sampling clocks. You select the sampling clock using the $\overline{\text{INT0}}$ Input Filter Select bits, bits 1 and 0.

$\overline{\text{INT0}}$ interrupt request occurs when the sampled input level matches three times.

When selecting 'sampling with filter', the value of the port P45, if read, will be the value after filtering.

Figure 1.12.11 shows the $\overline{\text{INT0}}$ input filter.

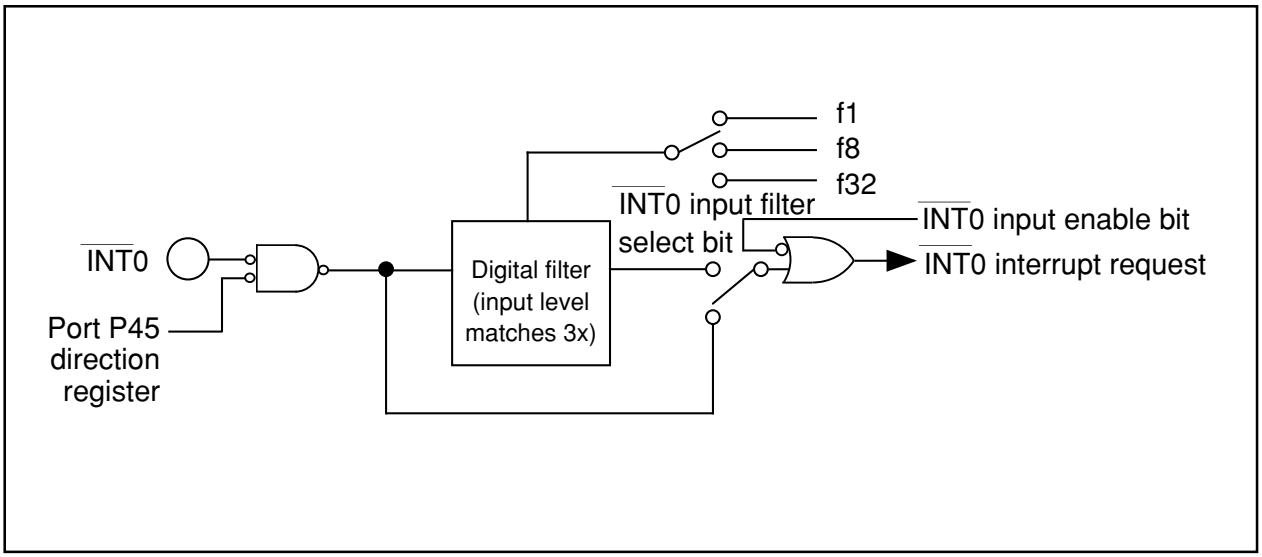


Figure 1.12.11. $\overline{\text{INT0}}$ input filter

CNTR0 interrupt

A CNTR0 interrupt is generated from the selected edge polarity, rising or falling edge, of the CNTR0 input signal. The edge polarity is selected using the CNTR0 polarity select bit (bit 2 of address 008B16). When using the CNTR0 interrupt, the port P17 direction register should be set to input.

When the pulse output mode of timer X is selected, the CNTR0 pin functions as a pulse output pin. In this case, a CNTR0 interrupt occurs by a falling or rising edge output from the CNTR0 pin. The port P17 direction register should also be set to input at this time.

Figure 1.12.12 shows the timer X mode register.

Timer X mode register

b7	b6	b5	b4	b3	b2	b1	b0

Symbol: TXMR
Address: 008B16
When reset: 000000002

Bit symbol	Bit name	Function	R	W
TXMOD0	Operation mode select bit 0, 1	b1 b0 0 0 : Timer mode or pulse period measurement mode 0 1 : Pulse output mode (Note 1) 1 0 : Event counter mode 1 1 : Pulse width measurement mode	○	○
TXMOD1			○	○
R0EDG	CNTR0 polarity switching bit (Note 2)	0 : Rising edge 1 : Falling edge	○	○
TXS	Timer X count start flag	0 : Stops counting 1 : Starts counting	○	○
TXOCNT	P30/TXOUT select bit	Function varies with each operation mode	○	○
TXMOD2	Operation mode select bit 2	0 : Except in pulse period measurement mode 1 : Pulse period measurement mode	○	○
TXEDG (Note 3)	Effectaul edge reception flag	Function varies with each operation mode.	○	○
TXUND (Note 3)	Timer X under flow flag	Function varies with each operation mode.	○	○

Note 1: In the pulse output mode, the direction register of port P17 should be set to input.
Note 2: This bit should rewrite with inhibiting the CNTR0 interrupt.
Note 3: TXEDG and TXUND were added after the product Ver.3.0 of the flash memory edition (M30100F3/M30102F3) after the product Ver.2.0 of the mask ROM edition (M30100Mx/ M30102Mx). Nothing is assigned to the product before this.

Figure 1.12.12 Timer X mode register

TCIN interrupt

A TCIN interrupt is generated from edges of a TCIN input signal or after 512 divisions of fRING.

To use TCIN input signal, set the time measurement input source switching bit (bit 7 of address 009A16) of timer C control register 0 to "0" (TCIN). The level of input to TCIN pin is sampled by one of three sampling clocks, f1, f8 or f32, selected with the digital filter clock select bit (bits 0 and 1 of address 009B16). The input level is determined when the sampled input level matches three times. (However, if the port P33 is read, the value will be the unfiltered value.) The edge polarity of an interrupt can be rising edge, falling edge, or both edges using the time measurement edge trigger select bits (bits 3 and 4 of address 009A16).

When triggered after 512 divisions of fRING, set the time measurement input source switching bit (bit 7 of address 009A16) to "1" (RING512).

Figure 1.12.13 shows the timer C control registers 0 and 1.

Timer C control register 0

b7	b6	b5	b4	b3	b2	b1	b0
	✕	✕					

Symbol	Address	When reset
TCC0	009A16	0XX000002

Bit symbol	Bit name	Function	R	W
TCC00	Time measurement control bit	0 : Time measurement disabled 1 : Time measurement enabled	○	○
TCC01 TCC02	Timer C clock select bit (Note 1)	b2 b1 0 0 : f1 (Note 2) 0 1 : f8 1 0 : f32 1 1 : Inhibit	○	○
TCC03 TCC04	Time measurement input edge trigger bit (Note 1)	b4 b3 0 0 : Rising edge 0 1 : Falling edge 1 0 : Both edges 1 1 : Inhibit	○	○
Nothing is assigned. When write, set "0". When read, their contents are "0".			—	—
TCC07	Time measurement input source switching bit (Note 1) (Note 3)	0 : TCIN 1 : RING512	○	○

Note 1: Change this bit when time measurement is disabled.
Note 2: When using an external RC circuit for the main clock, f1 cannot be selected for the Timer C clock.
Note 3: Set the ring oscillation stop bit (CM14) to “0” before setting this bit to “1”.

Timer C control register 1

b7	b6	b5	b4	b3	b2	b1	b0
✕	✕	✕	✕	✕	✕		

Symbol	Address	When reset
TCC1	009B16	XXXXXX112

Bit symbol	Bit name	Function	R	W
TCC10 TCC11	Digital filter clock select bit (Note)	b1 b0 0 0 : Cannot be used 0 1 : f1 1 0 : f8 1 1 : f32	○	○
Nothing is assigned. When write, set "0". When read, their contents are "0".			—	—

Note : Input edge becomes active when the same value from TCIN pin is sampled three times in succession.

Figure 1.12.13 Timer C control registers 0 and 1

Key Input Interrupt

When the direction register of any of P10 to P13 is set for input and the KIi (i=0 to 3) input enable bit of this port is set for enabled, if a falling or rising 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.

Figure 1.12.14 shows the block diagram of the key input interrupts. When the appropriate signal ("L" for a pin that has falling edge selected and "H" for a pin that has rising edge selected) is input to a pin for the input inhibit process has not been executed, inputs to the other pins are not detected as interrupts.

You should overwrite the KIi (i=0 to 3) input polarity select bit or the KIi (i =0 to 3) input enable bit only under conditions where the key input interrupt is disabled. After overwriting the KIi (i=0 to 3) input polarity select bit or the KIi (i=0 to 3) input enable bit, clear the interrupt request bit, and then enable the key input interrupt.

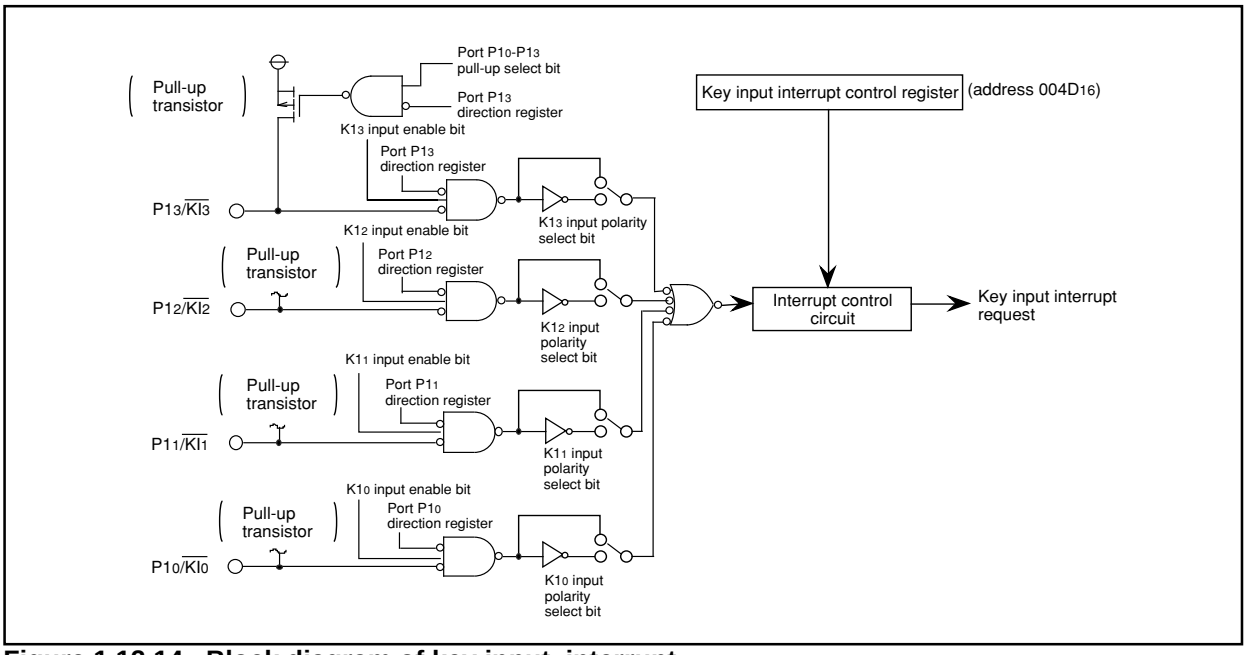


Figure 1.12.14. Block diagram of key input interrupt

Key input enable register

b7 b6 b5 b4 b3 b2 b1 b0

Symbol: KIEN
Address: 009816
When reset: 0016

Bit symbol	Bit name	Function	R	W
KI0EN	KI0 input enable bit	0 : Disabled 1 : Enabled	○	○
KI0PL	KI0 input polarity select bit	0 : Falling edge 1 : Rising edges	○	○
KI1EN	KI1 input enable bit	0 : Disabled 1 : Enabled	○	○
KI1PL	KI1 input polarity select bit	0 : Falling edge 1 : Rising edges	○	○
KI2EN	KI2 input enable bit	0 : Disabled 1 : Enabled	○	○
KI2PL	KI2 input polarity select bit	0 : Falling edge 1 : Rising edges	○	○
KI3EN	KI3 input enable bit	0 : Disabled 1 : Enabled	○	○
KI3PL	KI3 input polarity select bit	0 : Falling edge 1 : Rising edges	○	○

Figure 1.12.15. Key input enable register

Address Match Interrupt

An address match interrupt is generated immediately before the instruction at the address indicated by the address match interrupt register is executed. 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). The value of the program counter (PC) for an address match interrupt varies depending on the instruction being executed. Figure 1.12.16 shows the address match interrupt-related registers.

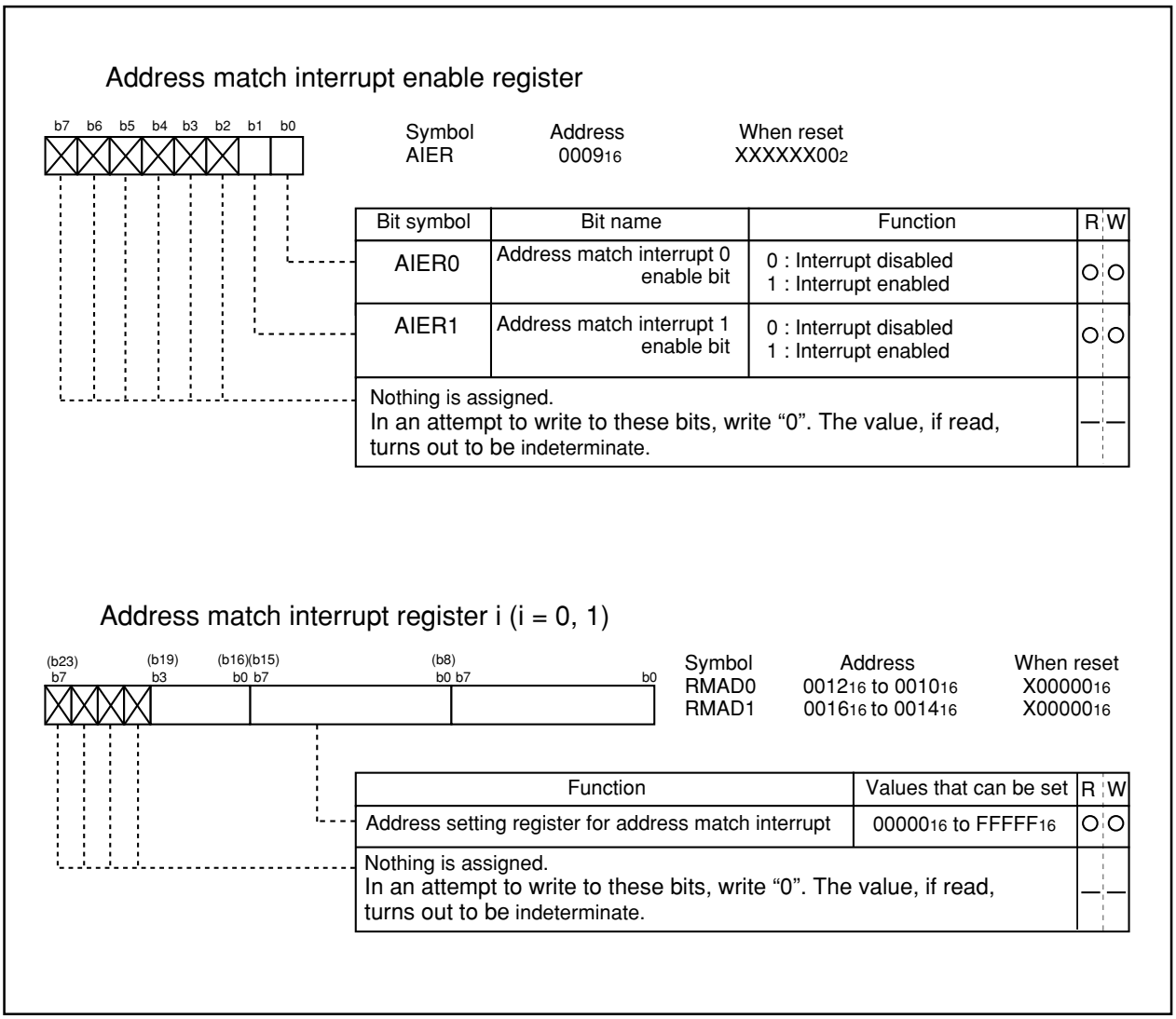


Figure 1.12.16. Address match interrupt-related registers

Precautions for Interrupts

(1) Reading address 0000016

• 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 0000016 will then be set to “0”.

Even if the address 0000016 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 0000016 by software.

(2) Setting the stack pointer

• The value of the stack pointer immediately after reset is initialized to 0000016. 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. Concerning the first instruction immediately after reset, generating any interrupts is prohibited.

(3) External interrupt

• Either an “L” level or an “H” level of at least 250 ns width is necessary for the signal input to pins $\overline{\text{INT0}}$ to $\overline{\text{INT3}}$ regardless of the CPU operation clock.

• When changing a polarity of pins $\overline{\text{INT0}}$ to $\overline{\text{INT3}}$, the interrupt request bit may become "1". Clear the interrupt request bit after changing the polarity. Figure 1.12.17 shows the switching condition of $\overline{\text{INT}}$ interrupt request.

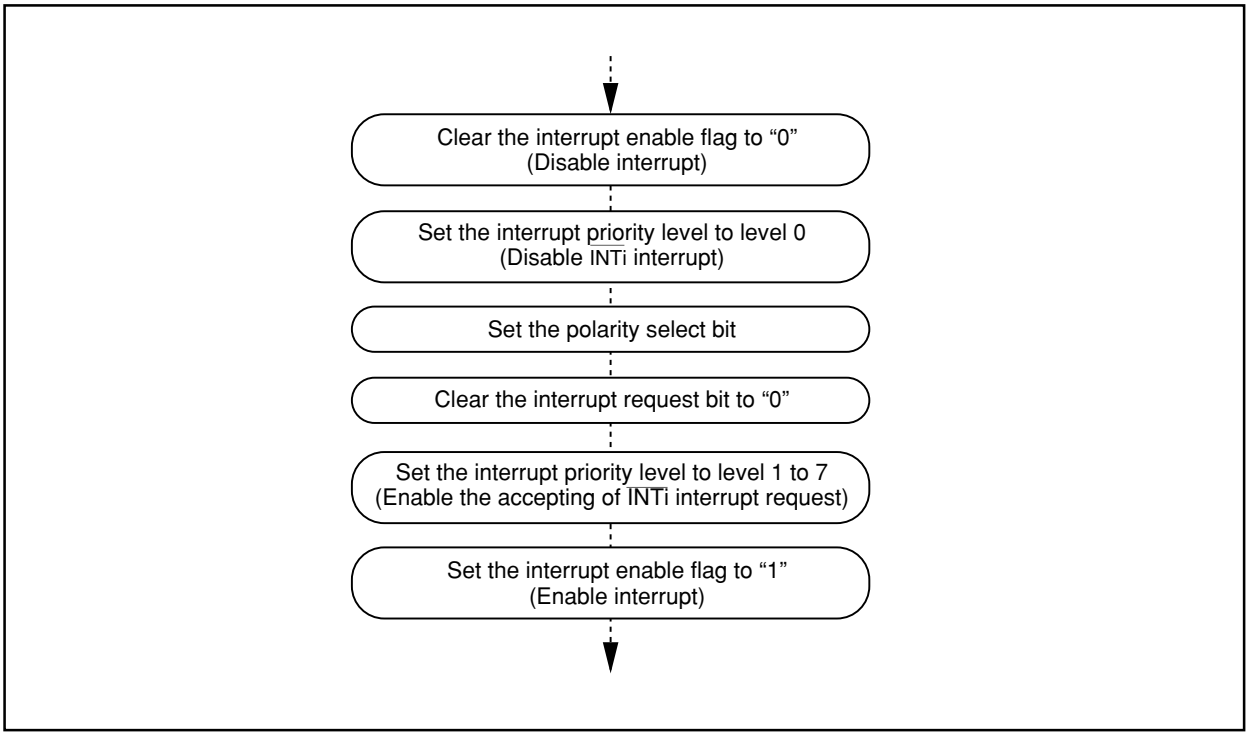


Figure 1.12.17. Switching condition of $\overline{\text{INT}}$ interrupt request

(4) Changing interrupt control register

See "Rewrite The Interrupt Control Register".

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 or reset is generated when an underflow occurs in the watchdog timer. A watchdog timer interrupt or reset is selected by bit 2 of the processor mode register 1. When XIN is selected for the BCLK, bit 7 of the watchdog timer control register (address 000F16) selects the prescaler division ratio (by 16 or by 128). When XCIN is selected as the BCLK, the prescaler is set for division by 2 regardless of bit 7 of the watchdog timer control register (address 000F16).

When XIN is selected in BCLK

$$\text{Watchdog timer cycle} = \frac{\text{Prescaler division ratio (16 or 128) x watchdog timer count (32768)}}{\text{BCLK}}$$

When XCIN is selected in BCLK

$$\text{Watchdog timer cycle} = \frac{\text{Prescaler division ratio (2) x watchdog timer count (32768)}}{\text{BCLK}}$$

For example, when BCLK is 10MHz and the prescaler division ratio is set to 16, the watchdog timer cycle is approximately 52.4 ms.

The watchdog timer is initialized by writing to the watchdog timer start register (address 000E16) 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 000E16).
Figure 1.13.1 shows the block diagram of the watchdog timer. Figure 1.13.2 shows the watchdog timer-related registers.

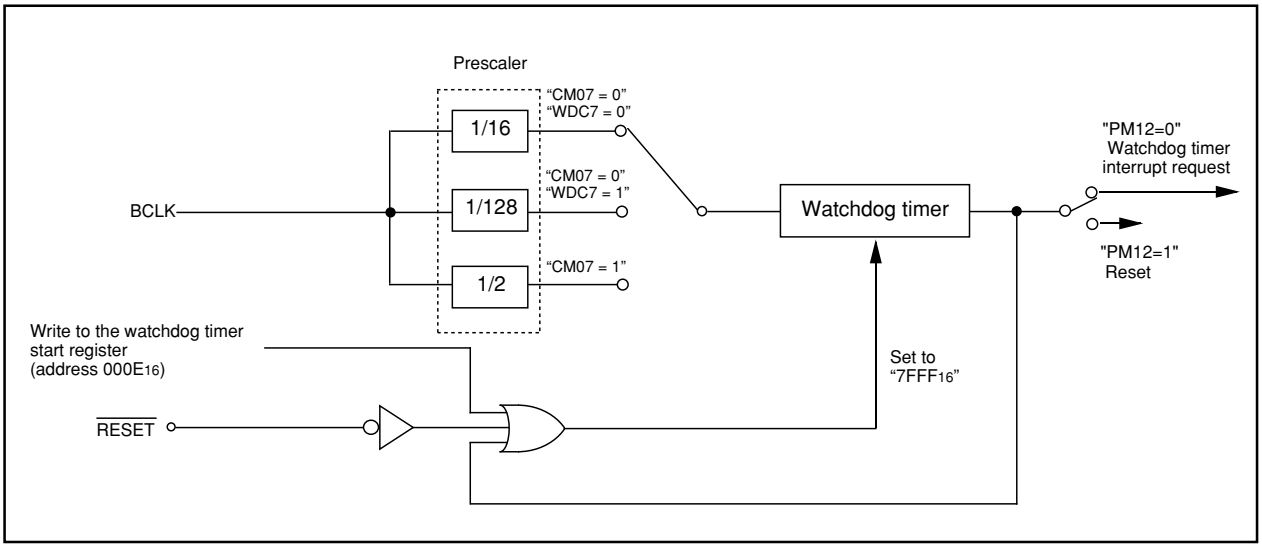


Figure 1.13.1. Block diagram of watchdog timer

Watchdog timer control register

b7	b6	b5	b4	b3	b2	b1	b0
	0	0					

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

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: WDTS
Address: $000E_{16}$
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 “$7FFF_{16}$” regardless of whatever value is written.	×	○

Processor mode register 1 (Note 1)

b7	b6	b5	b4	b3	b2	b1	b0
0	0						0

Symbol: PM1
Address: 0005_{16}
When reset: $00XXX0X0_2$

Bit symbol	Bit name	Function	R	W
Reserved bit		Must always be set to “0”	○	○
Nothing is assigned. In an attempt to write to this bits, write “0”. The value, if read, turns out to be indeterminate.			—	—
PM12	WDT interrupt/reset switching bit	0 : Watchdog timer interrupt 1 : Reset (Note 2)	○	○
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be indeterminate.			—	—
Reserved bit		Must always be set to “0”	○	○

Note 1: Set bit 1 of the protect register (address $000A_{16}$) to “1” when writing new values to this register.
Note 2: After setting this bit to "1", can not change to "0" by software.

Figure 1.13.2. Watchdog timer control and start registers

Timer

The microcomputer has four 8-bit timers and one 16-bit timer. The four 8-bit timers are Timer 1, Timer X, Timer Y, and Timer Z and each one has an 8-bit prescaler. The 16-bit timer is Timer C and has time measurement function. All these timers function independently. The count source for each timer is the operating clock that regulates the timing of timer operations such as counting and reloading.
Table 1.14.1 shows functional comparison.

Table 1.14.1. Functional comparison

		Timer1	TimerX	TimerY	TimerZ	TimerC
Configuration		8-bit timer with 8-bit prescaler	8-bit timer with 8-bit prescaler	8-bit timer with 8-bit prescaler	8-bit timer with 8-bit prescaler	16-bit free-run timer
Count		Down	Down	Down	Down	Up
Count source (Note)		•f_1 •f_8 •f_{32} •fc_{32}	•f_1 •f_8 •f_{32} •fc_{32}	•f_1 •f_8 •f_{RING} •fc_{32}	•f_1 •f_8 •TmrY underflow •fc_{32}	•f_1 •f_8 •f_{32}
Function	Timer mode	√	√	√	√	–
	Pulse output mode	–	√	–	–	–
	Event counter mode	–	√	–	–	–
	Pulse width measurement mode	–	√	–	–	–
	Pulse period measurement mode	–	√	–	–	–
	Programmable waveform generation mode	–	–	√	√	–
	Programmable one-shot generation mode	–	–	–	√	–
	Programmable wait one-shot generation mode	–	–	–	√	–
	Time measurement	–	–	–	–	√
Input pin		–	CNTR0	–	$\overline{INT0}$	TCIN
Output pin		–	CNTR0 TX_{OUT}	TY_{OUT}	TZ_{OUT}	–
Related interrupt		Tmr1 int	TmrX int CNTR0 int	TmrY int	TmrZ int	TmrC int TCIN int
Timer stop		–	√	√	√	√

Note: When using an external RC circuit for the main clock, f_1 cannot be selected for the count source.

Timer 1

Timer 1 is an 8-bit timer with an 8-bit prescaler. Figure 1.14.1 shows the block diagram of Timer 1. The timer constantly counts an internally generated count source (clock source). The count source after reset is set to f_1. The timer cannot stop counting. Table 1.14.2 shows the specifications of Timer 1 and Figure 1.14.2 shows Timer 1 related registers.

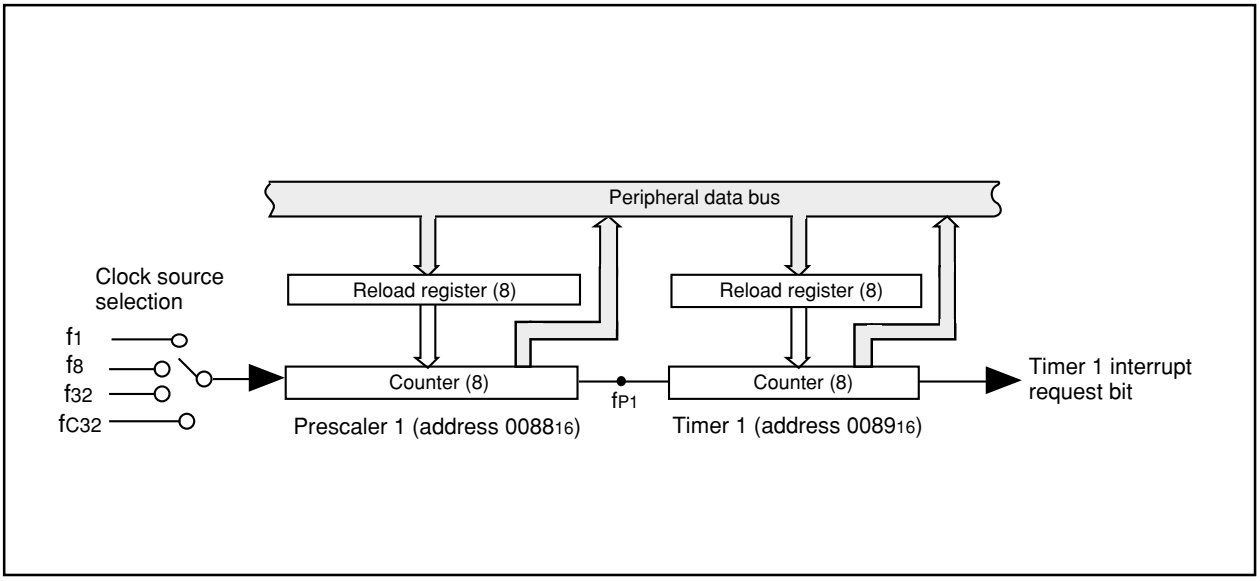


Figure 1.14.1. Block diagram of Timer 1

Table 1.14.2. Specifications of Timer 1 (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)/(m+1) n : Set value of Prescaler 1, m: Set value of Timer 1
Count start condition	After reset
Count stop condition	Disable to stop counting
Interrupt request generation timing	When Timer 1 underflows
Read from timer	Count value can be read out by reading Timer 1 register. Same applies to Prescaler 1 register.
Write to timer	When a value is written to Timer 1 register, it is written to both reload register and counter. Same applies to Prescaler 1 register.

Prescaler 1

Symbol	Address	When reset
PRE1	0088₁₆	XX₁₆

Function	Values that can be set	R	W
When set value = n, Prescaler 1 divides the internal count source by n+1	00₁₆ to FF₁₆	○	○

Timer 1

Symbol	Address	When reset
T1	0089₁₆	XX₁₆

Function	Values that can be set	R	W
When set value = m, Timer 1 divides the underflow of Prescaler 1 by m+1	00₁₆ to FF₁₆	○	○

Timer count source setting register

Symbol	Address	When reset
TCSS	008E₁₆	00₁₆

Bit symbol	Bit name	Function	R	W
TXCK0	Timer X count source select bit (Note 5)	b1 b0 0 0 : f1 (Note 1) 0 1 : f8 1 0 : f32 1 1 : fC32	○	○
TXCK1			○	○
TYCK0	Timer Y count source select bit (Note 2) (Note 5)	b3 b2 0 0 : f1 (Note 1) 0 1 : f8 1 0 : Ring oscillator output (Note 3) 1 1 : fC32	○	○
TYCK1			○	○
TZCK0	Timer Z count source select bit (Note 4) (Note 5)	b5 b4 0 0 : f1 (Note 1) 0 1 : f8 1 0 : Timer Y underflow 1 1 : fC32	○	○
TZCK1			○	○
T1CK0	Timer 1 count source select bit	b7 b6 0 0 : f1 (Note 1) 0 1 : f8 1 0 : f32 1 1 : fC32	○	○
T1CK1			○	○

Note 1: When using an external RC circuit for the main clock, f1 cannot be selected for the count source.
Note 2: The waveform extend function cannot be used when selecting f1 for count source.
Note 3: When attempting to select ring oscillator output, set the ring oscillation enable bit (CM14) of the system clock control register (address 0007₁₆) for oscillation enabled.
Note 4: The waveform extend function cannot be used when selecting Timer Y underflow and f1 for count source.
Note 5: Avoid switching a count source, while a counter is inprogress. Timer counter should be stopped before switching a count source.

Figure 1.14.2. Timer 1-related register

Timer X

Timer X is an 8-bit timer with an 8-bit prescaler. Figure 1.14.3 shows the block diagram of Timer X. Figures 1.14.4 and 1.14.5 shows the Timer X-related registers.

Timer X has the five operation modes listed as follows:

- Timer mode: The timer counts an internal count source (clock source).
- Pulse output mode: The timer counts an internal count source and outputs the pulses whose polarity is inverted at the timer the timer underflows.
- Event counter mode: The timer counts pulses from an external source.
- Pulse width measurement mode: The timer measures an external pulse's pulse width.
- Pulse period measurement mode: The timer measures an external pulse's period.

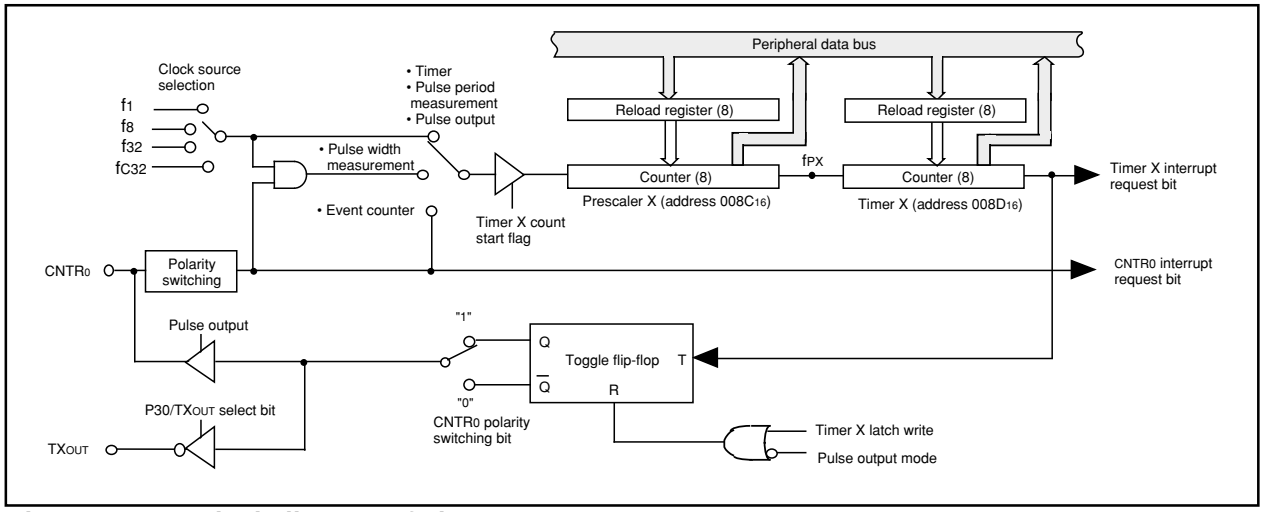


Figure 1.14.3. Block diagram of Timer X

Timer X mode register

b7	b6	b5	b4	b3	b2	b1	b0

Symbol: TXMR | Address: 008B16 | When reset: 000000002

Bit symbol	Bit name	Function	R	W
TXMOD0	Operation mode select bit 0, 1	b1 b0 0 0 : Timer mode or pulse period measurement mode 0 1 : Pulse output mode (Note 1) 1 0 : Event counter mode 1 1 : Pulse width measurement mode	○	○
TXMOD1			○	○
R0EDG	CNTR0 polarity switching bit (Note 2)	0 : Rising edge 1 : Falling edge	○	○
TXS	Timer X count start flag	0 : Stops counting 1 : Starts counting	○	○
TXOCNT	P30/TXOUT select bit	Function varies with each operation mode	○	○
TXMOD2	Operation mode select bit 2	0 : Except in pulse period measurement mode 1 : Pulse period measurement mode	○	○
TXEDG (Note 3)	Effectaul edge reception flag	Function varies with each operation mode.	○	○
TXUND (Note 3)	Timer X under flow flag	Function varies with each operation mode.	○	○

Note 1: In the pulse output mode, the direction register of port P17 should be set to input.
Note 2: This bit should rewrite with inhibiting the CNTR0 interrupt.
Note 3: TXEDG and TXUND were added after the product Ver.3.0 of the flash memory edition (M30100F3/M30102F3) after the product Ver.2.0 of the mask ROM edition (M30100Mx/ M30102Mx). Nothing is assigned to the product before this.

Figure 1.14.4. Timer X-related registers (1)

Prescaler X

b7 b0

Symbol	Address	When reset
PREX	008C16	FF16

Function	Values that can be set	R	W
• Timer mode Internal count source is counted	0016 to FF16	○	○
• Pulse output mode Internal count source is counted	0016 to FF16	○	○
• Event counter mode Externally input pulses are counted	0016 to FF16	○	○
• Pulse width measurement mode Pulse width of externally input pulses is measured (Internal count source is counted)	0016 to FF16	○	○
• Pulse period measurement mode Pulse period of externally input pulses is measured (Internal count source is counted)	0016 to FF16	○	○

Timer X

b7 b0

Symbol	Address	When reset
TX	008D16	FF16

Function	Values that can be set	R	W
Underflow of Prescaler X is counted	0016 to FF16	○	○

Timer count source setting register

b7 b6 b5 b4 b3 b2 b1 b0

Symbol	Address	When reset
TCSS	008E16	0016

Bit symbol	Bit name	Function	R	W
TXCK0	Timer X count source select bit	b1 b0 0 0 : f1 (Note 1) 0 1 : f8 1 0 : f32 1 1 : fC32	○	○
TXCK1	(Note 5)		○	○
TYCK0	Timer Y count source select bit	b3 b2 0 0 : f1 (Note 1) 0 1 : f8 1 0 : Ring oscillator output (Note 3) 1 1 : fC32	○	○
TYCK1	(Note 2) (Note 5)		○	○
TZCK0	Timer Z count source select bit	b5 b4 0 0 : f1 (Note 1) 0 1 : f8 1 0 : Timer Y underflow 1 1 : fC32	○	○
TZCK1	(Note 4) (Note 5)		○	○
T1CK0	Timer 1 count source select bit	b7 b6 0 0 : f1 (Note 1) 0 1 : f8 1 0 : f32 1 1 : fC32	○	○
T1CK1			○	○

Note 1: When using an external RC circuit for the main clock, f1 cannot be selected for the count source.
Note 2: The waveform extend function cannot be used when selecting f1 for count source.
Note 3: When attempting to select ring oscillator output, set the ring oscillation enable bit (CM14) of the system clock control register (address 000716) for oscillation enabled.
Note 4: The waveform extend function cannot be used when selecting Timer Y underflow and f1 for count source.
Note 5: Avoid switching a count source, while a counter is inprogress. Timer counter should be stopped before switching a count source.

Figure 1.14.5. Timer X-related registers (2)

(1) Timer mode

In this mode, the timer counts an internally generated count source. (See Table 1.14.3) Figure 1.14.6 shows the Timer X mode register in timer mode.

Table 1.14.3. 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)/(m+1) n : Set value of Prescaler X, m: Set value of Timer X
Count start condition	Count start flag is set (=1)
Count stop condition	Count start flag is reset (=0)
Interrupt request generation timing	When Timer X underflows [Timer X interruption]
CNTR0 pin function	Programmable I/O port, or CNTR0 interrupt input pin
TX_{OUT} pin function	Programmable I/O port
Read from timer	Count value can be read out by reading Timer X register. Same applies to Prescaler X register.
Write to timer	When a value is written to Timer X register, it is written to both reload register and counter. Same applies to Prescaler X register.

Timer X mode register

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

Symbol: TXMR
Address: $008B_{16}$
When reset: 00000000_2

Bit symbol	Bit name	Function	R	W
TXMOD0	Operation mode select bit 0, 1	b1 b0 0 0 : Timer mode	○	○
TXMOD1			○	○
R0EDG	CNTR0 polarity switching bit (Note 1)	0 : Rising edge 1 : Falling edge	○	○
TXS	Timer X count start flag	0 : Stops counting 1 : Starts counting	○	○
TXOCNT	0 : In timer mode, set to "0"		○	○
TXMOD2	0 : In timer mode, set to "0"		○	○
TXEDG	Invalid in timer mode. When write, set "0". When read, this contents is indeteminate.		○	○
TXUND	Invalid in timer mode. When write, set "0". When read, this contents is indeteminate.		○	○

Note 1: This bit should rewrite with inhibiting the CNTR0 interrupt.

Figure 1.14.6. Timer X mode register in timer mode

(2) Pulse output mode

In this mode, the timer counts an internally generated count source, and outputs from the CNTR0 pin a pulse whose polarity is inverted each time the timer underflows. (See Table 1.14.4) Figure 1.14.7 shows Timer X mode register in pulse output mode.

Table 1.14.4. Specifications of pulse output 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)/(m+1) n : Set value of Prescaler X, m: Set value of Timer X
Count start condition	Count start flag is set (=1)
Count stop condition	Count start flag is reset (=0)
Interrupt request generation timing	• When Timer X underflows [Timer X interruption] • Rising (R0EDG=0) or falling (R0EDG=1) of CNTR0 output [CNTR0 interruption] (Note)
CNTR0 pin function	Pulse output
TX_{OUT} pin function	Programmable I/O port or pulse output (Inverted waveform of the pulse output from the CNTR0 pin)
Read from timer	Count value can be read out by reading Timer X register. Same applies to Prescaler X register.
Write to timer	When a value is written to Timer X register, it is written to both reload register and counter. Same applies to Prescaler X register.
Select function	• Pulse output function Each time the timer underflows, the TX_{OUT} pin's polarity is reversed • CNTR0 polarity switching function The polarity level at starting of pulse output can be selected to be "High" or "Low" with software.

Note: When setting the timer X mode register to pulse output mode, the CNTR0 interrupt request bit becomes "1". Thus, when using an CNTR0 interrupt, the CNTR0 interrupt request bit must be set to "0" after setting the timer X mode register.

Timer X mode register

b7	b6	b5	b4	b3	b2	b1	b0
		0				0	1

Symbol TXMR — Address $008B_{16}$ — When reset 00000000_2

Bit symbol	Bit name	Function	R	W
TXMOD0	Operation mode select bit 0, 1	b1 b0 0 1 : Pulse output mode (Note 1)	○	○
TXMOD1			○	○
R0EDG	CNTR0 polarity switching bit (Note 3)	0: Output starts at "H" (Interrupt at rising edge) 1: Output starts at "L" (Interrupt at falling edge)	○	○
TXS	Timer X count start flag	0 : Stops counting 1 : Starts counting	○	○
TXOCNT	$P3_0/TX_{OUT}$ select bit	0 : Port $P3_0$ 1 : TX_{OUT} output (Note 2)	○	○
TXMOD2	0 : Set to "0" in pulse output mode		○	○
TXEDG	Invalid in pulse output mode. When write, set "0". When read, this contents is indeteminate.		○	○
TXUND	Invalid in pulse output mode. When write, set "0". When read, this contents is indeteminate.		○	○

Note 1: In the pulse output mode, the direction register of port $P1_7$ must be set to input.
Note 2: Output is set regardless of the setting of the direction register of port $P3_0$.
Note 3: This bit should rewrite with inhibiting the CNTR0 interrupt.

Figure 1.14.7. Timer X mode register in pulse output mode

(3) Event counter mode

In this mode, the timer counts an external signal fed to CNTR0 pin. (See Table 1.14.5) Figure 1.14.8 shows Timer X mode register in event counter mode.

Table 1.14.5. Specifications of event counter mode

Item	Specification
Count source	External signals fed to CNTR0 pin (Active edge is selected by software)
Count operation	• Down count • When the timer underflows, it reloads the reload register contents before continuing counting
Divide ratio	1/(n+1)/(m+1) n : Set value of Prescaler X, m: Set value of Timer X
Count start condition	Count start flag is set (=1)
Count stop condition	Count start flag is reset (=0)
Interrupt request generation timing	• When Timer X underflows [Timer X interruption] • Rising (R0EDG=0) or falling (R0EDG=1) of CNTR0 input [CNTR0 interruption]
CNTR0 pin function	Count source input
TXOUT pin function	Programmable I/O port
Read from timer	Count value can be read out by reading Timer X register. Same applies to Prescaler X register.
Write to timer	When a value is written to Timer X register, it is written to both reload register and counter. Same applies to Prescaler X register.
Select function	• CNTR0 polarity switching function The active edge of count source can be selected to be the rising or the falling edge with software.

Timer X mode register

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

Symbol: TXMR
Address: 008B16
When reset: 000000002

Bit symbol	Bit name	Function	R	W
TXMOD0	Operation mode select bit 0, 1	b1 b0 1 0 : Event counter mode	○	○
TXMOD1			○	○
R0EDG	CNTR0 polarity switching bit (Note 1)	0: Counts at rising edge (Interrupt at rising edge) 1: Counts at falling edge(Interrupt at falling edge)	○	○
TXS	Timer X count start flag	0 : Stops counting 1 : Starts counting	○	○
TXOCNT	0 : Set to "0" in event counter mode		○	○
TXMOD2	0 : Set to "0" in event counter mode		○	○
TXEDG	Invalid in event counter mode. When write, set "0". When read, this contents is indeteminate.		○	○
TXUND	Invalid in event counter mode. When write, set "0". When read, this contents is indeteminate.		○	○

Note 1: This bit should rewrite with inhibiting the CNTR0 interrupt.

Figure 1.14.8. Timer X mode register in event counter mode

(4) Pulse width measurement mode

In this mode, the timer measures the pulse width of an external signal fed to CNTR0 pin. (See Table 1.14.6) Figure 1.14.9 shows the Timer X mode register in pulse width measurement mode. Figure 1.14.10 shows an operation example in pulse width measurement mode.

Table 1.14.6. Specifications of pulse width measurement mode

Item	Specification
Count source	f_1, f_8, f_{32}, f_{C32}
Count operation	• Down count • Continuously counts the selected signal only when the measurement pulse is "H" level, or conversely only "L" level. • When the timer underflows, it reloads the reload register contents before continuing counting
Count start condition	Count start flag is set (=1)
Count stop condition	Count start flag is reset (=0)
Interrupt request generation timing	• When Timer X underflows [Timer X interruption] • Rising (R0EDG=0) or falling (R0EDG=1) of CNTR0 input [CNTR0 interruption]
CNTR0 pin function	Measurement pulse input
TX_{OUT} pin function	Programmable I/O port
Read from timer	Count value can be read out by reading Timer X register. Same applies to Prescaler X register.
Write to timer	When a value is written to Timer X register, it is written to both reload register and counter. Same applies to Prescaler X register.
Select function	• CNTR0 polarity switching function The measurement pulse input can be selected to be "H" level width or "L" level width by software.

Timer X mode register

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

Symbol: TXMR
Address: $008B_{16}$
When reset: 00000000_2

Bit symbol	Bit name	Function	R	W
TXMOD0	Operation mode select bit 0, 1	b1 b0 1 1 : Pulse width measurement mode	○	○
TXMOD1			○	○
R0EDG	CNTR0 polarity switching bit (Note 1)	0 : Measures "H" level width (Interrupt at rising edge) 1 : Measures "L" level width (Interrupt at falling edge)	○	○
TXS	Timer X count start flag	0 : Stops counting 1 : Starts counting	○	○
TXOCNT	0 : Set to "0" in pulse width measurement mode		○	○
TXMOD2	0 : Set to "0" in pulse width measurement mode		○	○
TXEDG	Invalid in pulse width measurement mode. When write, set "0". When read, this contents is indeteminate.		○	○
TXUND	Invalid in pulse width measurement mode. When write, set "0". When read, this contents is indeteminate.		○	○

Note 1: This bit should rewrite with inhibiting the CNTR0 interrupt.

Figure 1.14.9. Timer X mode register in pulse width measurement mode

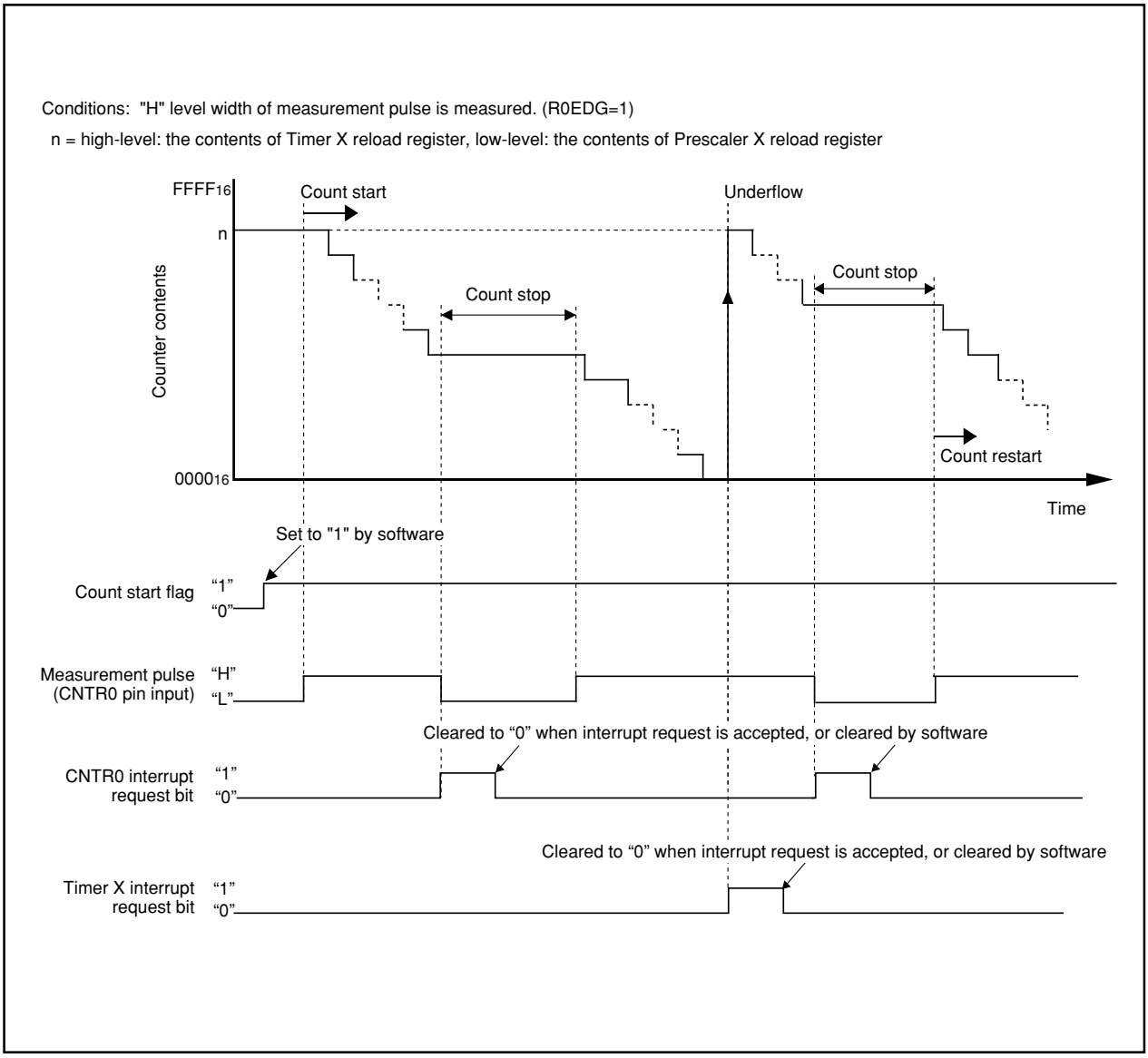


Figure 1.14.10. Operation example in pulse width measurement mode

(5) Pulse period measurement mode

In this mode, the timer measures the pulse period of an external signal fed to CNTR0 pin. (See Table 1.14.7) Figure 1.14.11 shows the Timer X mode register in pulse period measurement mode.

Table 1.14.7. Specifications of pulse period measurement mode

Item	Specification
Count source	f_1, f_8, f_{32}, f_{C32}
Count operation	• Down count • After valid edge of measurement pulse is input, the timer X reloads contents in the reload register and continues counting in underflow of the second prescaler X.
Count start condition	Count start flag is set (=1)
Count stop condition	Count start flag is reset (=0)
Interrupt request generation timing	• When Timer X underflows [Timer X interruption] • Rising (R0EDG=0) or falling (R0EDG=1) of CNTR0 input [CNTR0 interruption or Timer X interrupt]
CNTR0 pin function	Measurement pulse input (Note)
TX_{OUT} pin function	Programmable I/O port
Read from timer	When reading Timer X register, the count value of buffer for read purpose can be read out. The buffer of read purpose retains the content of Timer X register upon an active edge of measurement pulse, and starts to read the content of Timer X register by reading Timer X.
Write to timer	When a value is written to Timer X register, it is written to both reload register and counter. Same applies to Prescaler X register.
Select function	• CNTR0 polarity switching function The measurement period of pulse input can be selected to be a period from one rising edge to the next rising edge or from one falling edge to the next falling edge by software.

Note: Avoid a shorter period pulse input than double prescaler X period. Longer pulse for H width and L width than the prescaler X period should be input to the CNTR0 pin. If shorter pulse than the period is input to the CNTR0 pin, the input may be disabled.

Timer X mode register

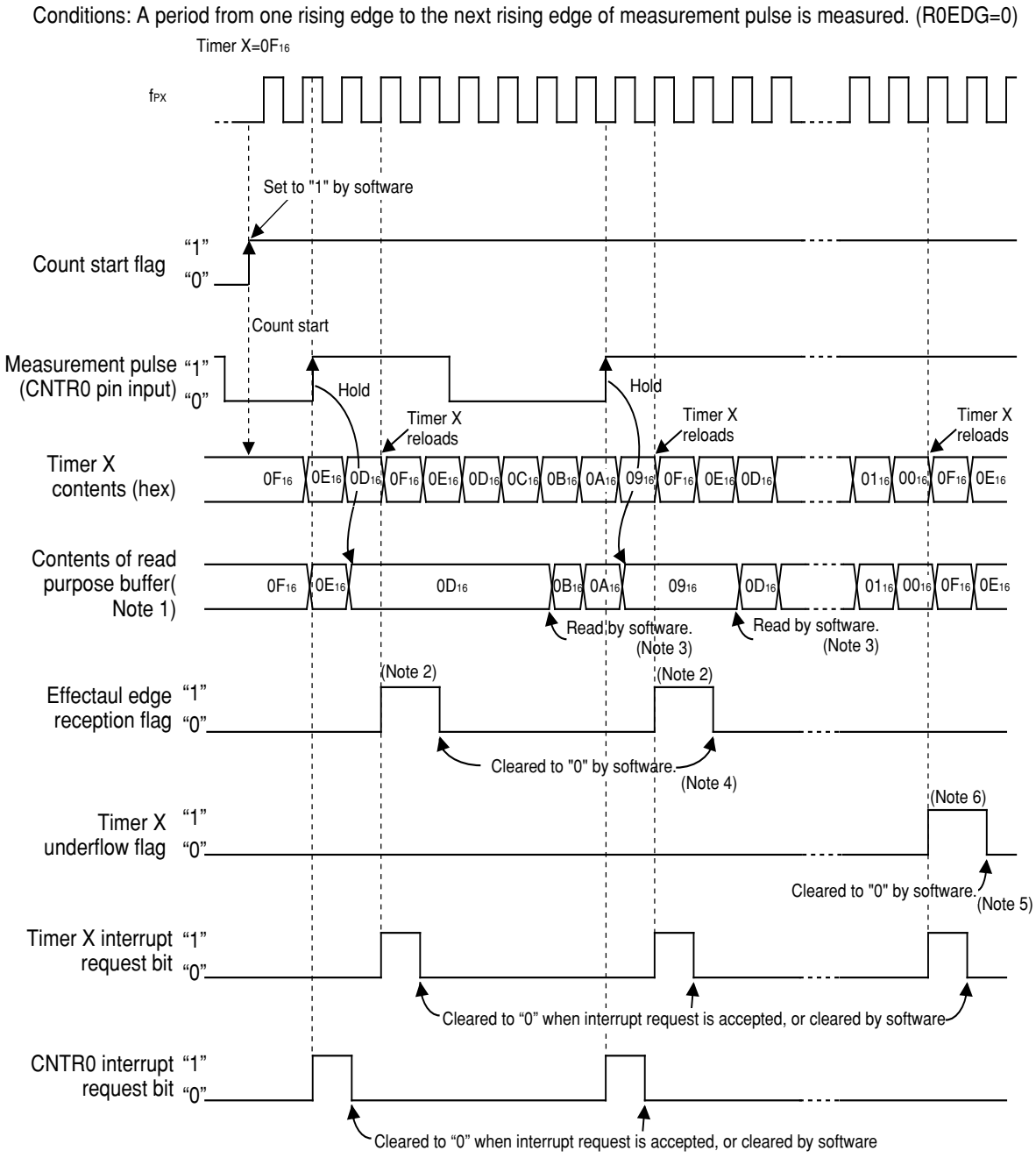
b7	b6	b5	b4	b3	b2	b1	b0
		1	0			0	0

Symbol: TXMR
Address: $008B_{16}$
When reset: 00000000_2

Bit symbol	Bit name	Function	R	W
TXMOD0	Operation mode select bit 0, 1	b1 b0 0 0 : Pulse period measurement mode	○	○
TXMOD1			○	○
R0EDG	CNTR0 polarity switching bit (Note 1)	0: Measures a measurement pulse from one rising edge to the next rising edge (Interrupt at rising edge) 1: Measures a measurement pulse from one falling edge to the next falling edge (Interrupt at falling edge)	○	○
TXS	Timer X count start flag	0 : Stops counting 1 : Starts counting	○	○
TXOCNT	0 : In pulse period measurement mode, set to "0"		○	○
TXMOD2	Operation mode select bit 2	1 : Pulse period measurement mode	○	○
TXEDG (Note 2,3)	Effectaul edge reception flag	0 : No effectual edge 1 : Effectual edge found	○	○
TXUND (Note 2,3)	Timer X under flow flag	0 : No under flow 1 : Under flow found	○	○

Note 1: This bit should rewrite with inhibiting the CNTR0 interrupt.
Note 2: TXEDG and TXUND are set to "0" by writing a "0" in a program. (Writing a "1" has no effect.)
Note 3: TXEDG and TXUND were added after the product Ver.3.0 of the flash memory edition (M30100F3/M30102F3) after the product Ver.2.0 of the mask ROM edition (M30100Mx/ M30102Mx). Nothing is assigned to the product before this.

Figure 1.14.11. Timer X mode register in pulse period measurement mode



Note 1: If timer X is read out in pulse period measurement mode, the contents of the read purpose buffer can be read.
Note 2: After an active edge of measurement pulse is input, effectaul edge reception flag(TXEDG) is set to "1" when the prescaler X underflows for the second time.
Note 3: The timer X should be read out before the next active edge is input after TXEDG is set to "1". If the timer X is not read before the next active edge is input, the value in the read purpose buffer remains unchanged and therefore is not updated on an active edge.
Note 4: When set to "0" by software, use a MOV instruction to write "0" to the bit 6 (TXEDG) in the timer X mode register ($008B_{16}$). At the same time, write "1" to the bit 7 (TXUND).
Note 5: When set to "0" by software, use a MOV instruction to write "0" to the bit 7 (TXUND) in the timer X more register ($008B_{16}$). At the same time, write "1" to the bit 6 (TXEDG).
Note 6: If the timer X underflow flag (TXUND) and TXEDG are both set to "1". In this case, the validity of TXUND sholud be judged by the contents of the read purpose buffer.

Figure 1.14.12. Operation example in pulse period measurement mode

Timer Y

Timer Y is an 8-bit timer with an 8-bit prescaler and has two reload registers - Timer Y Primary and Timer Y Secondary. Figure 1.14.13 shows the block diagram of Timer Y. Figures 1.14.14 to 1.14.16 show the Timer Y-related registers.

Timer Y has the two operation modes listed as follows:

- Timer mode: The timer counts an internal count source (clock source).
- Programmable waveform generation mode: The timer outputs pulses of a given width successively.

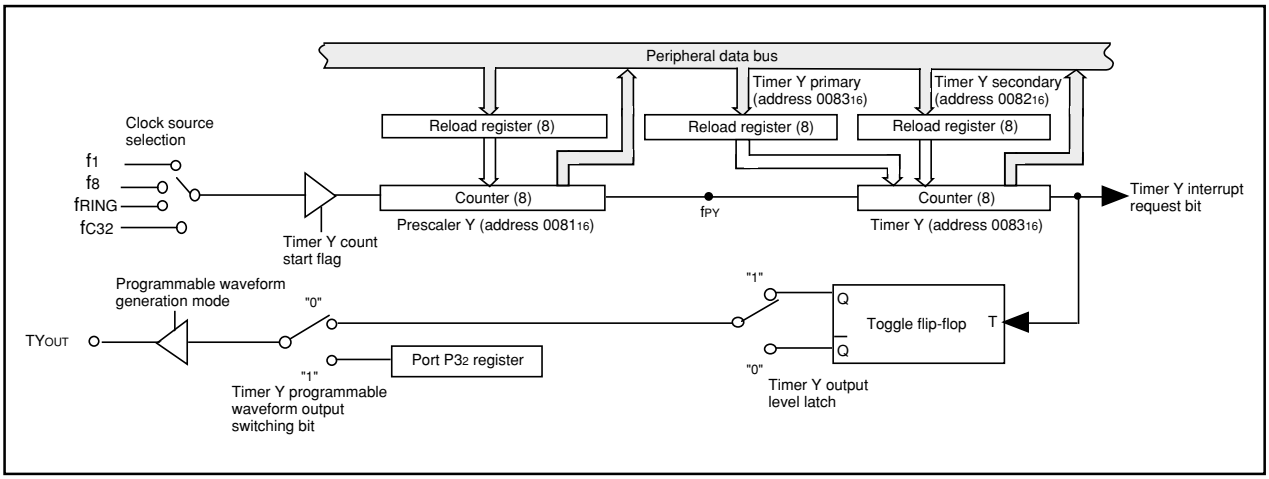


Figure 1.14.13. Block diagram of Timer Y

Timer Y, Z mode register

b7	b6	b5	b4	b3	b2	b1	b0

Symbol: TYZMR — Address: 008016 — When reset: 000000X02

Bit symbol	Bit name	Function	R	W
TYMOD0	Timer Y operation mode bit	0 : Timer mode 1 : Programmable waveform generation mode (Note 1)	○	○
Nothing is assigned. When write, set "0". When read, the content is "0".			—	—
TYWC	Timer Y write control bit	Function varies depending on the operation mode	○	○
TYS	Timer Y count start flag	0 : Stops counting (Note 2) 1 : Starts counting	○	○
TZMOD0	Timer Z operation mode bit (Note 3)	b5 b4 0 0 : Timer mode 0 1 : Programmable waveform generation mode 1 0 : Programmable one-shot generation mode 1 1 : Programmable wait one-shot generation mode	○	○
TZMOD1			○	○
TZWC	Timer Z write control bit	Function varies depending on the operation mode	○	○
TZS	Timer Z count start flag	0 : Stops counting (Note 2) 1 : Starts counting	○	○

Note 1: In programmable waveform generation mode, port P32 is set for output regardless of the value of the direction register.
Note 2: When this bit is cleared to "0", the timer reloads the content of the reload register before it stops. Read out the count value before you stop the timer.
Note 3: When timer Z operation mode bit is set for "01", "10" or "11", port P31 is set for output regardless of the value of the direction register.

Figure 1.14.14. Timer Y-related registers (1)

Prescaler Y

b7 b0

Symbol	Address	When reset
PREY	0081₁₆	FF₁₆

Function	Values that can be set	R	W
• Timer mode Internal count source is counted	00₁₆ to FF₁₆	○	○
• Programmable waveform generation mode Internal count source is counted	00₁₆ to FF₁₆ (Note)	○	○

Note: When using the waveform extend function, set the value "00₁₆" for the Prescaler Y.

Timer Y Secondary

b7 b0

Symbol	Address	When reset
TYSC	0082₁₆	FF₁₆

Function	Values that can be set	R	W
• Timer mode Invalid	———	—	—
• Programmable waveform generation mode Underflow of Prescaler Y is counted (Note 1)	00₁₆ to FF₁₆	× (Note 2)	○

Note 1: The values of Timer Y Primary and Timer Y Secondary are reloaded to the Timer Y alternately for counting.
Note 2: The count value can be read out by reading the Timer Y Primary even when the secondary period is being counted.

Timer Y Primary

b7 b0

Symbol	Address	When reset
TYPR	0083₁₆	FF₁₆

Function	Values that can be set	R	W
• Timer mode Underflow of Prescaler Y is counted	00₁₆ to FF₁₆	○	○
• Programmable waveform generation mode Underflow of Prescaler Y is counted (Note)	00₁₆ to FF₁₆	○	○

Note: The values of Timer Y Primary and Timer Y Secondary are reloaded to the Timer Y alternately for counting.

Timer Y, Z output control register

b7 b6 b5 b4 b3 b2 b1 b0

Symbol	Address	When reset
TYZOC	008A₁₆	XXXXX000₂

Bit symbol	Bit name	Function	R	W
TZOS	Timer Z one-shot start bit (Note 1)	0 : Stops one-shot 1 : Starts one-shot	○	○
TYOCNT	Timer Y programmable waveform generation output switching bit (Note 2)	0 : Outputs programmable waveform 1 : Outputs the value of P3₂ port register	○	○
TZOCNT	Timer Z programmable waveform generation output switching bit (Note 2)	0 : Outputs programmable waveform 1 : Outputs the value of P3₁ port register	○	○
Nothing is assigned. When write, set "0". When read, their contents are "0".			—	—

Note 1: The timer Z one-shot start bit is automatically cleared to "0" when the output of one-shot waveform is completed. The timer Z one-shot start bit should be set to "0" by program when the one-shot waveform output is terminated by setting the count start flag to "0" during the wave form output.
Note 2: The timer Y/Z programmable waveform generaton output switching bit is valid only when operating in programmable waveform generation mode.

Figure 1.14.15. Timer Y-related registers (2)

Timer Y, Z waveform output control register

Symbol	Address	When reset
PUM	0084₁₆	00₁₆

Bit symbol	Bit name	Function	R	W
TYPUM0	Timer Y primary waveform extension control bit	0 : No waveform extension 1 : Waveform extension (Note 1)	○	○
TYPUM1	Timer Y secondary waveform extension control bit	0 : No waveform extension 1 : Waveform extension (Note 1)	○	○
TZPUM0	Timer Z primary waveform extension control bit	0 : No waveform extension 1 : Waveform extension (Note 2)	○	○
TZPUM1	Timer Z secondary waveform extension control bit	0 : No waveform extension 1 : Waveform extension (Note 2)	○	○
TYOPL	Timer Y output level latch	Function varies depending on the operation mode	○	○
TZOPL	Timer Z output level latch	Function varies depending on the operation mode	○	○
INOSTG	$\overline{INT0}$ pin one-shot trigger control bit (Note 4) (Timer Z)	0 : $\overline{INT0}$ pin one-shot trigger invalid 1 : $\overline{INT0}$ pin one-shot trigger valid	○	○
INOSEG	$\overline{INT0}$ pin one-shot trigger polarity select bit (Note 3) (Timer Z)	0 : Edge trigger at falling edge 1 : Edge trigger at rising edge	○	○

Note 1: When setting this bit to "1", the Prescaler Y Register must be set to "00₁₆".
Note 2: When setting this bit to "1", the Prescaler Z Register must be set to "00₁₆".
Note 3: This bit is valid only when $\overline{INT0}$ input polarity select bit (bit 1 at address 0096₁₆) is "0" (one-edge).
Note 4: This bit must be set to "1", after setting $\overline{INT0}$ input enable bit (bit 0 at address 0096₁₆), $\overline{INT0}$ input polarity select bit (bit 1 at address 0096₁₆), and $\overline{INT0}$ pin one-shot trigger polarity select bit.

Timer count source setting register

Symbol	Address	When reset
TCSS	008E₁₆	00₁₆

Bit symbol	Bit name	Function	R	W
TXCK0	Timer X count source select bit	b1 b0 0 0 : f_1 (Note 1) 0 1 : f_8 1 0 : f_{32} 1 1 : f_{C32}	○	○
TXCK1	(Note 5)		○	○
TYCK0	Timer Y count source select bit	b3 b2 0 0 : f_1 (Note 1) 0 1 : f_8 1 0 : Ring oscillator output (Note 3) 1 1 : f_{C32}	○	○
TYCK1	(Note 2) (Note 5)		○	○
TZCK0	Timer Z count source select bit	b5 b4 0 0 : f_1 (Note 1) 0 1 : f_8 1 0 : Timer Y underflow 1 1 : f_{C32}	○	○
TZCK1	(Note 4) (Note 5)		○	○
T1CK0	Timer 1 count source select bit	b7 b6 0 0 : f_1 (Note 1) 0 1 : f_8 1 0 : f_{32} 1 1 : f_{C32}	○	○
T1CK1			○	○

Note 1: When using an external RC circuit for the main clock, f_1 cannot be selected for the count source.
Note 2: The waveform extend function cannot be used when selecting f_1 for count source.
Note 3: When attempting to select ring oscillator output, set the ring oscillation enable bit (CM14) of the system clock control register (address 0007₁₆) for oscillation enabled.
Note 4: The waveform extend function cannot be used when selecting Timer Y underflow and f_1 for count source.
Note 5: Avoid switching a count source, while a counter is inprogress. Timer counter should be stopped before switching a count source.

Figure 1.14.16. Timer Y-related registers (3)

(1) Timer mode

In this mode, the timer counts an internally generated count source. (See Table 1.14.8) The Timer Y secondary is unused in this mode. Figure 1.14.17 shows the Timer Y, Z mode register and Timer Y, Z waveform output control register in timer mode.

Table 1.14.8. Specifications of timer mode

Item	Specification
Count source	f_1, f_8, ring oscillator output, f_{C32}
Count operation	• Down count • When the timer underflows, it reloads the reload register contents before continuing counting (When the Timer Y underflows, the contents of the Timer Y primary reload register is reloaded.) • When a counting stops, the timer reloads the content of the reload register before it stops.
Divide ratio	1/(n+1)/(m+1) n : Set value of Prescaler Y, m: Set value of Timer Y primary
Count start condition	Count start flag is set (=1)
Count stop condition	Count start flag is reset (=0) (Note 1)
Interrupt request generation timing	When Timer Y underflows
TY_{OUT} pin function	Programmable I/O port
Read from timer	Count value can be read out by reading Timer Y primary register. Same applies to Prescaler Y register.
Write to timer	When a value is written to Timer Y Primary register, it is written to both reload register and counter or written to only reload register. Selected by software. Same applies to Prescaler Y register.
Select function	• Timer Y write control function When a value is written to Timer Y Primary register, it can be selected that the value is written to both reload register and counter or written to only reload register. Same applies to Prescaler Z register. (Note 2)

Note 1: When the count is stopped, the Timer Y interrupt request flag becomes "1" and an interrupt may occur. Thus, interrupts must be disabled before the count is stopped. Furthermore, set the Timer Y interrupt request flag to "0" before starting counting again.

Note 2: If writing to the Timer Y or prescaler Y under the following conditions being filled at the same time the Timer Y interrupt request flag becomes "1" and an interrupt occurs.

<Conditions>

• Timer Y write control bit (bit 2 of address 0080) is "0" (write to timer and reload register simultaneously)

• Timer Y count start flag (bit 3 of address 0080) is "1" (count start)

To write to the Timer Y or prescaler Y in the above state, disable interrupts before writing.

Timer Y, Z mode register

b7	b6	b5	b4	b3	b2	b1	b0
						X	0

Symbol: TYZMR
Address: 0080₁₆
When reset: 000000X0₂

Bit symbol	Bit	Function	R	W
TYMOD0	Timer Y operation mode bit	0 : Timer mode	○	○
Nothing is assigned. When write, set "0". When read, the content is indeterminate.			—	—
TYWC	Timer Y write control bit	0 : Write to timer and reload register simultaneously (Note 2) 1 : Write to reload register	○	○
TYS	Timer Y count start flag	0 : Stops counting (Note 1) 1 : Starts counting	○	○
TZMOD0	Timer Z-related bit		○	○
TZMOD1			○	○
TZWC			○	○
TZS			○	○

Note 1: When this bit is cleared to "0", the timer reloads the content of the reload register before it stops. Read out the count value before you stop the timer.
Note 2: At the TYWC bit is "0", when you write in the prescaler Y while the timer Y is counting,the timer Y reload content of the timer Y reload register.

Timer Y, Z waveform output control register

b7	b6	b5	b4	b3	b2	b1	b0

Symbol: PUM
Address: 0084₁₆
When reset: 00₁₆

Bit symbol	Bit name	Function	R	W
TYPUM0	Timer Y primary waveform extension control bit	Invalid in timer mode	○	○
TYPUM1	Timer Y secondary waveform extension control bit	Invalid in timer mode	○	○
TZPUM0	Timer Z-related bits		○	○
TZPUM1			○	○
TYOPL	Timer Y output level latch	Invalid in timer mode	○	○
TZOPL	Timer Z-related bits		○	○
INOSTG			○	○
INOSEG			○	○

Figure 1.14.17. Timer Y, Z mode register in timer mode

(2) Programmable waveform generation mode

In this mode, the microcontroller, while counting the set values of Timer Y primary and Timer Y secondary alternately, outputs from the TYOUT pin a waveform whose polarity is inverted each time Timer Y primary or Timer Y secondary underflows. (See Table 1.14.9) A counting starts by counting the set value in the Timer Y primary. Figure 1.14.18 shows Timer Y, Z mode register in programmable waveform generation mode. Figure 1.14.19 shows the operation example.

Table 1.14.9. Specifications of programmable waveform generation mode

Item	Specification
Count source	f_1, f_8, ring oscillator output, f_{C32}
Count operation	• Down count • When the timer underflows, it reloads the contents of primary reload register and secondary reload register alternately before continuing counting. • When a counting stops, the timer reloads the content of the reload register before it stops.
Divide ratio	fi/(n+1)/((m+1)+(l+1)) n : Set value of Prescaler Y, m: Set value of Timer Y primary, l: Set value of Timer Y secondary
Count start condition	Count start flag is set (=1)
Count stop condition	Count start flag is reset (=0) (Note 1)
Interrupt request generation timing	When Timer Y underflows during secondary period
TYOUT pin function	Pulse output (Note 2)
Read from timer	Count value can be read out by reading Timer Y primary register. Same applies to Prescaler Y register. (Note 3)
Write to timer	When a value is written to Timer Y primary register, it is written to only reload register. Same applies to Timer Y secondary register and Prescaler Y register. (Note 4)
Select function	• Output level latch select function The output level of a waveform being counted during primary and secondary periods is selectable. • Programmable waveform generation output switching function Can select either programmable waveform or the value of Port $P3_2$ register for output. (Note 5) • Waveform extend function (Note 6) The waveform output primary period and secondary period can each be extended 0.5 cycles of the count source Frequency when waveform extended: 2xfi/((2x(m+1))+(2x(l+1))+TYPUM0+TYPUM1) Duty: (2x(m+1)+TYPUM0)/((2x(m+1)+TYPUM0)+(2x(l+1)+TYPUM1)) m: set value of Timer Y primary, l: set value of Timer Y secondary TYPUM0: Timer Y primary waveform extension control bit TYPUM1: Timer Y secondary waveform extension control bit

Note 1: When the count is stopped, the Timer Y interrupt request flag becomes "1" and an interrupt may occur. Thus, interrupts must be disabled before the count is stopped. Furthermore, set the Timer Y interrupt request flag to "0" before starting counting again.

Note 2: When the counting stopped, the pin is the secondary period output level.

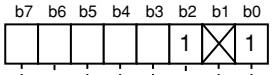
Note 3: Even when counting the secondary period, read out the Timer Y primary register.

Note 4: The set value of Timer Y secondary register and waveform extension control bits as well as Timer Y primary register are made effective by writing a value to the Timer Y primary register. The written values are reflected to the waveform output from the next primary period after writing to the Timer Y primary register.

Note 5: The output is switched in sync with timer Y secondary underflow.

Note 6: When using the waveform extend function, the Prescaler Y register must be set to "00_{16}".

Timer Y, Z mode register

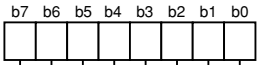


Symbol: TYZMR | Address: 0080₁₆ | When reset: 000000X0₂

Bit symbol	Bit name	Function	R	W
TYMOD0	Timer Y operation mode bit	1 : Programmable waveform generation mode (Note 1)	○	○
Nothing is assigned. When write, set "0". When read, the content is indeterminate.			—	—
TYWC	Timer Y write control bit	Set to "1" in programmable waveform generation mode	○	○
TYS	Timer Y count start flag	0 : Stops counting (Note 2) 1 : Starts counting	○	○
TZMOD0	Timer Z-related bit		○	○
TZMOD1			○	○
TZWC			○	○
TZS			○	○

Note 1: Output is set for Port P3₂ regardless of the value of the direction register.
Note 2: When this bit is cleared to "0", the timer reloads the content of the reload register before it stops. Read out the count value before you stop the timer.

Timer Y, Z waveform output control register

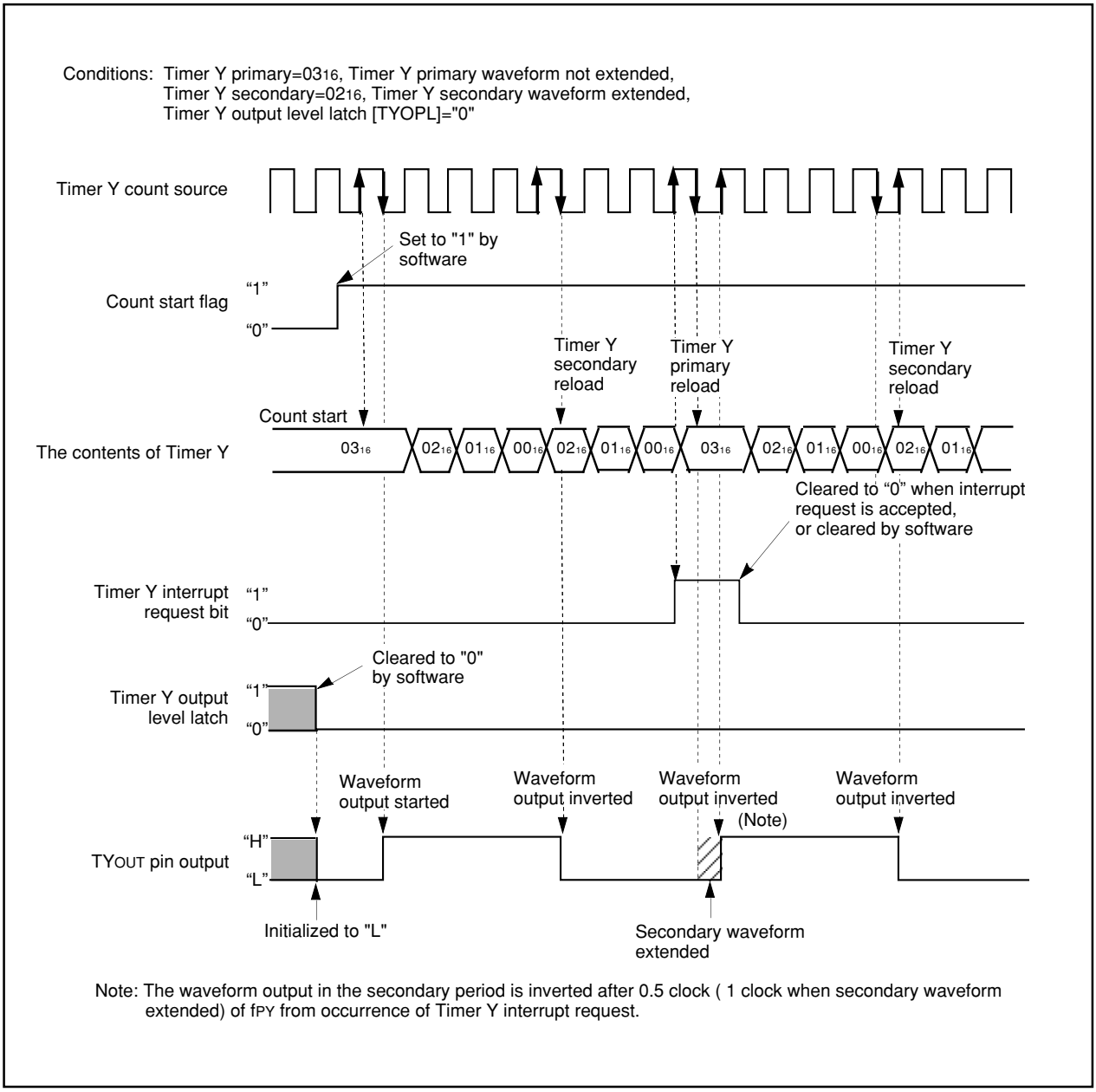


Symbol: PUM | Address: 0084₁₆ | When reset: 00₁₆

Bit symbol	Bit name	Function	R	W
TYPUM0	Timer Y primary waveform extension control bit	0 : No waveform extension 1 : Waveform extension (Note 1)(Note 2)	○	○
TYPUM1	Timer Y secondary waveform extension control bit	0 : No waveform extension 1 : Waveform extension (Note 1)(Note 2)	○	○
TZPUM0	Timer Z-related bits		○	○
TZPUM1			○	○
TYOPL	Timer Y output level latch	0 : Outputs "H" for the period set by Timer Y primary and "L" for the period set by Timer Y secondary. "L" is outputted when the timer is stopped. 1 : Outputs "L" for the period set by Timer Y primary and "H" for the period set by Timer Y secondary. "H" is outputted when the timer is stopped.	○	○
TZOPL	Timer Z-related bits		○	○
INOSTG			○	○
INOSEG			○	○

Note 1: When setting this bit to "1", the Prescaler Y Register must be set to "00₁₆".
Note 2: The waveform extend function cannot be used when selecting f_1 for count source.

Figure 1.14.18. Timer Y, Z mode register and Timer Y, Z waveform output control register in programmable waveform generation mode



Note: The waveform output in the secondary period is inverted after 0.5 clock (1 clock when secondary waveform extended) of fPY from occurrence of Timer Y interrupt request.

Figure 1.14.19. Timer Y operation example in programmable waveform generation mode

■Programmable waveform generation output switching function

When the Timer Y programmable waveform generation output switching bit (bit 1 at address 008A16) is set to 0, the output from TYOUT is inverted synchronously when the Timer Y secondary underflows. And when set to 1, the Port P32 register value is output from TYOUT synchronously when the Timer Y secondary underflows.

Timer Z

Timer Z is an 8-bit timer with an 8-bit prescaler and has two reload registers - Timer Z Primary and Timer Z Secondary. Figure 1.14.20 shows the block diagram of Timer Z. Figures 1.14.21 to 1.14.24 show the Timer Z-related registers.

Timer Z has the four operation modes listed as follows:

- Timer mode: The timer counts an internal count source (clock source) or Timer Y underflow.
- Programmable waveform generation mode: The timer outputs pulses of a given width successively.
- Programmable one-shot generation mode: The timer outputs one-shot pulse.
- Programmable wait one-shot generation mode: The timer outputs delayed one-shot pulse.

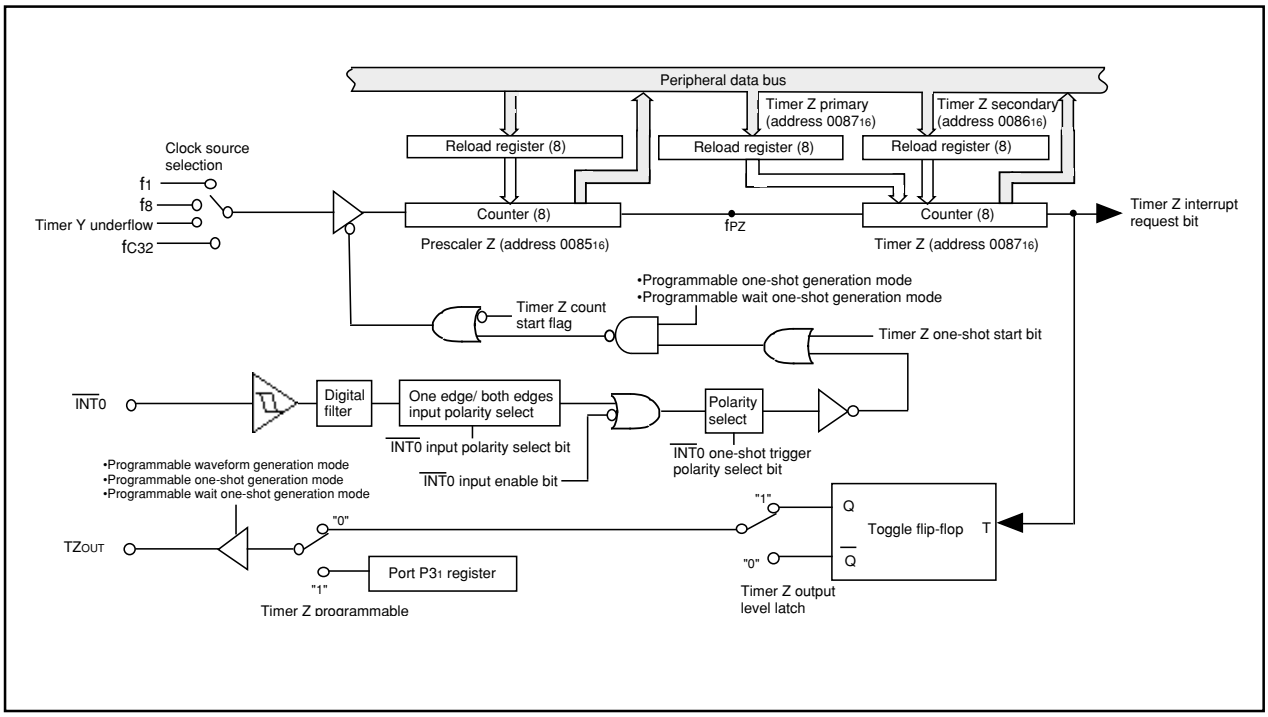
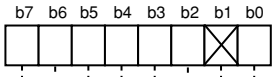


Figure 1.14.20. Block diagram of Timer Z

Timer Y, Z mode register



Symbol: TYZMR | Address: 0080₁₆ | When reset: 000000X0₂

Bit symbol	Bit name	Function	R	W
TYMOD0	Timer Y operation mode bit	0 : Timer mode 1 : Programmable waveform generation mode (Note 1)	○	○
	Nothing is assigned. When write, set "0". When read, the content is "0".		—	—
TYWC	Timer Y write control bit	Function varies depending on the operation mode	○	○
TYS	Timer Y count start flag	0 : Stops counting (Note 2) 1 : Starts counting	○	○
TZMOD0	Timer Z operation mode bit (Note 3)	b5 b4 0 0 : Timer mode 0 1 : Programmable waveform generation mode 1 0 : Programmable one-shot generation mode 1 1 : Programmable wait one-shot generation mode	○	○
TZMOD1			○	○
TZWC	Timer Z write control bit	Function varies depending on the operation mode	○	○
TZS	Timer Z count start flag	0 : Stops counting (Note 2) 1 : Starts counting	○	○

Note 1: In programmable waveform generation mode, port P3₂ is set for output regardless of the value of the direction register.
Note 2: When this bit is cleared to "0", the timer reloads the content of the reload register before it stops. Read out the count value before you stop the timer.
Note 3: When timer Z operation mode bit is set for "01", "10" or "11", port P3₁ is set for output regardless of the value of the direction register.

Figure 1.14.21. Timer Z-related registers (1)

Prescaler Z

b7 b0

Symbol	Address	When reset
PREZ	0085₁₆	FF₁₆

Function	Values that can be set	R	W
• Timer mode Internal count source or Timer Y underflow is counted	00₁₆ to FF₁₆	○	○
• Programmable waveform generation mode Internal count source or Timer Y underflow is counted	00₁₆ to FF₁₆ (Note)	○	○
• Programmable one-shot generation mode Internal count source or Timer Y underflow is counted	00₁₆ to FF₁₆ (Note)	○	○
• Programmable wait one-shot generation mode Internal count source or Timer Y underflow is counted	00₁₆ to FF₁₆ (Note)	○	○

Note: When using the waveform extend function, set the value "00₁₆" for the Prescaler Z.

Timer Z Secondary

b7 b0

Symbol	Address	When reset
TZSC	0086₁₆	FF₁₆

Function	Values that can be set	R	W
• Timer mode Invalid	———	—	—
• Programmable waveform generation mode Underflow of Prescaler Z is counted (Note 1)	00₁₆ to FF₁₆	× Note 2)	○
• Programmable one-shot generation mode Invalid	———	—	—
• Programmable wait one-shot generation mode Underflow of Prescaler Z is counted (One-shot width is counted)	00₁₆ to FF₁₆	×	○

Note 1: Each value of Timer Z Primary and Timer Z Secondary is reloaded to the Timer Z alternately for counting.
Note 2: The count value can be read out by reading the Timer Z Primary even when the secondary period is being counted.

Timer Z Primary

b7 b0

Symbol	Address	When reset
TZPR	0087₁₆	FF₁₆

Function	Values that can be set	R	W
• Timer mode Underflow of Prescaler Z is counted	00₁₆ to FF₁₆	○	○
• Programmable waveform generation mode Underflow of Prescaler Z is counted (Note)	00₁₆ to FF₁₆	○	○
• Programmable one-shot generation mode Underflow of Prescaler Z is counted (One-shot width is counted)	00₁₆ to FF₁₆	○	○
• Programmable wait one-shot generation mode Underflow of Prescaler Z is counted (Wait period is counted)	00₁₆ to FF₁₆	○	○

Note: Each value of Timer Z Primary and Timer Z Secondary is reloaded to the Timer Z alternately for counting.

Figure 1.14.22. Timer Z-related registers (2)

Timer Y, Z waveform output control register

b7	b6	b5	b4	b3	b2	b1	b0

Symbol: PUM — Address: 0084₁₆ — When reset: 00₁₆

Bit symbol	Bit name	Function	R	W
TYPUM0	Timer Y primary waveform extension control bit	0 : No waveform extension 1 : Waveform extension (Note 1)	○	○
TYPUM1	Timer Y secondary waveform extension control bit	0 : No waveform extension 1 : Waveform extension (Note 1)	○	○
TZPUM0	Timer Z primary waveform extension control bit	0 : No waveform extension 1 : Waveform extension (Note 2)	○	○
TZPUM1	Timer Z secondary waveform extension control bit	0 : No waveform extension 1 : Waveform extension (Note 2)	○	○
TYOPL	Timer Y output level latch	Function varies depending on the operation mode	○	○
TZOPL	Timer Z output level latch	Function varies depending on the operation mode	○	○
INOSTG	$\overline{INT0}$ pin one-shot trigger control bit (Note 4) (Timer Z)	0 : $\overline{INT0}$ pin one-shot trigger invalid 1 : $\overline{INT0}$ pin one-shot trigger valid	○	○
INOSEG	$\overline{INT0}$ pin one-shot trigger polarity select bit (Note 3) (Timer Z)	0 : Edge trigger at falling edge 1 : Edge trigger at rising edge	○	○

Note 1: When setting this bit to "1", the Prescaler Y Register must be set to "00₁₆".
Note 2: When setting this bit to "1", the Prescaler Z Register must be set to "00₁₆".
Note 3: This bit is valid only when $\overline{INT0}$ input polarity select bit (bit 1 at address 0096₁₆) is "0" (one-edge).
Note 4: This bit must be set to "1", after setting $\overline{INT0}$ input enable bit (bit 0 at address 0096₁₆), $\overline{INT0}$ input polarity select bit (bit 1 at address 0096₁₆), and $\overline{INT0}$ pin one-shot trigger polarity select bit.

Timer count source setting register

b7	b6	b5	b4	b3	b2	b1	b0

Symbol: TCSS — Address: 008E₁₆ — When reset: 00₁₆

Bit symbol	Bit name	Function	R	W
TXCK0	Timer X count source select bit	b1 b0 0 0 : f_1 (Note 1) 0 1 : f_8 1 0 : f_{32} 1 1 : f_{C32}	○	○
TXCK1	(Note 5)		○	○
TYCK0	Timer Y count source select bit	b3 b2 0 0 : f_1 (Note 1) 0 1 : f_8 1 0 : Ring oscillator output (Note 3) 1 1 : f_{C32}	○	○
TYCK1	(Note 2) (Note 5)		○	○
TZCK0	Timer Z count source select bit	b5 b4 0 0 : f_1 (Note 1) 0 1 : f_8 1 0 : Timer Y underflow 1 1 : f_{C32}	○	○
TZCK1	(Note 4) (Note 5)		○	○
T1CK0	Timer 1 count source select bit	b7 b6 0 0 : f_1 (Note 1) 0 1 : f_8 1 0 : f_{32} 1 1 : f_{C32}	○	○
T1CK1			○	○

Note 1: When using an external RC circuit for the main clock, f_1 cannot be selected for the count source.
Note 2: The waveform extend function cannot be used when selecting f_1 for count source.
Note 3: When attempting to select ring oscillator output, set the ring oscillation enable bit (CM14) of the system clock control register (address 0007₁₆) for oscillation enabled.
Note 4: The waveform extend function cannot be used when selecting Timer Y underflow and f_1 for count source.
Note 5: Avoid switching a count source, while a counter is inprogress. Timer counter should be stopped before switching a count source.

Figure 1.14.23. Timer Z-related registers (3)

Timer Y, Z output control register

Symbol	Address	When reset
TYZOC	$008A_{16}$	$XXXXX000_2$

Bit symbol	Bit name	Function	R	W
TZOS	Timer Z one-shot start bit (Note 1)	0 : Stops one-shot 1 : Starts one-shot	○	○
TYOCNT	Timer Y programmable waveform generation output switching bit (Note 2)	0 : Outputs programmable waveform 1 : Outputs the value of $P3_2$ port register	○	○
TZOCNT	Timer Z programmable waveform generation output switching bit (Note 2)	0 : Outputs programmable waveform 1 : Outputs the value of $P3_1$ port register	○	○
Nothing is assigned. When write, set "0". When read, their contents are "0".			—	—

Note 1: The timer Z one-shot start bit is automatically cleared to "0" when the output of one-shot waveform is completed. The timer Z one-shot start bit should be set to "0" by program when the one-shot waveform output is terminated by setting the count start flag to "0" during the wave form output.
Note 2: The timer Y/Z programmable waveform generaton output switching bit is valid only when operating in programmable waveform generation mode.

External input enable register

Symbol	Address	When reset
INTEN	0096_{16}	00_{16}

Bit symbol	Bit name	Function	R	W
INT0EN	$\overline{INT}0$ input enable bit (Note)	0 : Disabled 1 : Enabled	○	○
INT0PL	$\overline{INT}0$ input polarity select bit (Note)	0 : One edge 1 : Two edges	○	○
INT1EN	$\overline{INT}1$ input enable bit	0 : Disabled 1 : Enabled	○	○
INT1PL	$\overline{INT}1$ input polarity select bit	0 : One edge 1 : Two edges	○	○
INT2EN	$\overline{INT}2$ input enable bit	0 : Disabled 1 : Enabled	○	○
INT2PL	$\overline{INT}2$ input polarity select bit	0 : One edge 1 : Two edges	○	○
INT3EN	$\overline{INT}3$ input enable bit	0 : Disabled 1 : Enabled	○	○
INT3PL	$\overline{INT}3$ input polarity select bit	0 : One edge 1 : Two edges	○	○

Note : This bit must be set in condition of INT0 pin one-shot trigger invalid (INOSTG="0").

$\overline{INT}0$ input filter select register

Symbol	Address	When reset
INT0F	$001E_{16}$	$XXXXX000_2$

Bit symbol	Bit name	Function	R	W
INT0F0 INT0F1	$\overline{INT}0$ input filter select bit	b1 b0 0 0 : No filter 0 1 : Filter with f_1 sampling 1 0 : Filter with f_8 sampling 1 1 : Filter with f_{32} sampling	○	○
INT0F2	UART0 receive hardware interrupt enable bit (Note)	0 : Disabled 1 : Enabled	○	○
Nothing is assigned. In an attempt to write to these bits, write "0". The value, if read, turns out to be indeterminate.			–	–

Note: Interrupts used for debugging purposes only.

Figure 1.14.24. Timer Z-related registers (4)

(1) Timer mode

In this mode, the timer counts an internally generated count source or Timer Y underflow. (See Table 1.14.10) The Timer Z secondary is unused in this mode. Figure 1.14.25 shows the Timer Y, Z mode register and Timer Y, Z waveform output control register in timer mode.

Table 1.14.10. Specifications of timer mode

Item	Specification
Count source	f_1, f_8, Timer Y underflow, f_{C32}
Count operation	• Down count • When the timer underflows, it reloads the reload register contents before continuing counting (When the Timer Z underflows, the contents of the Timer Z primary reload register is reloaded.) • When a counting stops, the timer reloads the content of the reload register before stopping counting.
Divide ratio	1/(n+1)/(m+1) n : Set value of Prescaler Z, m: Set value of Timer Z primary
Count start condition	Count start flag is set (=1)
Count stop condition	Count start flag is reset (=0) (Note 1)
Interrupt request generation timing	When Timer Z underflows
TY$_{OUT}$ pin function	Programmable I/O port
$\overline{INT}$0 pin function	Programmable I/O port, or external interrupt input pin
Read from timer	Count value can be read out by reading Timer Z primary register. Same applies to Prescaler Z register.
Write to timer	When a value is written to Timer Z Primary register, it is written to both reload register and counter or written to only reload register. Selected by software. Same applies to Prescaler Z register.
Select function	• Timer Z write control function When a value is written to Timer Z Primary register, it can be selected that the value is written to both reload register and counter or written to only reload register. Same applies to Prescaler Z register. (Note 2)

Note 1: When the count is stopped, the Timer Z interrupt request flag becomes "1" and an interrupt may occur. Thus, interrupts must be disabled before the count is stopped. Furthermore, set the Timer Z interrupt request bit to "0" before starting counting again.

Note 2: If writing to the Timer Z or prescaler Z under the following conditions being filled at the same time the Timer Z interrupt request flag becomes "1" and an interrupt occurs.

<Conditions>

• Timer Z write control bit (bit 6 of address 0080) is "0" (write to timer and reload register simultaneously)

• Timer Z count start flag (bit 7 of address 0080) is "1" (count start)

To write to the Timer Z or prescaler Z in the above state, disable interrupts before writing.

Timer Y, Z mode register

b7	b6	b5	b4	b3	b2	b1	b0
		0	0			⊠	

Symbol: TYZMR
Address: 0080₁₆
When reset: 000000X0₂

Bit symbol	Bit name	Function	R	W
TYMOD0	Timer Y-related bit		○	○
	Nothing is assigned. When write, set "0". When read, the content is "0".		—	—
TYWC	Timer Y-related bits		○	○
TYS			○	○
TZMOD0	Timer Z operation mode bit	b5 b4 0 0 : Timer mode	○	○
TZMOD1			○	○
TZWC	Timer Z write control bit	0 : Write to timer and reload register simultaneously (Note 2) 1 : Write to reload register	○	○
TZS	Timer Z count start flag	0 : Stops counting (Note 1) 1 : Starts counting	○	○

Note 1: When this bit is cleared to "0", the timer reloads the content of the reload register before it stops. Read out the count value before you stop the timer.
Note 2: At the TZWC bit is "0", when you write in the prescaler Z while the timer Z is counting,the timer Z reload content of the timer Z reload register.

Timer Y, Z waveform output control register

b7	b6	b5	b4	b3	b2	b1	b0

Symbol: PUM
Address: 0084₁₆
When reset: 00₁₆

Bit symbol	Bit name	Function	R	W
TYPUM0	Timer Y-related bits		○	○
TYPUM1			○	○
TZPUM0	Timer Z primary waveform extension control bit	Invalid in timer mode	○	○
TZPUM1	Timer Z secondary waveform extension control bit	Invalid in timer mode	○	○
TYOPL	Timer Y-related bit		○	○
TZOPL	Timer Z output level latch	Invalid in timer mode	○	○
INOSTG	$\overline{INT0}$ pin one-shot trigger control bit	Invalid in timer mode	○	○
INOSEG	$\overline{INT0}$ pin one-shot trigger polarity select bit	Invalid in timer mode	○	○

Figure 1.14.25. Timer Y, Z mode register and Timer Y, Z waveform output control register in timer mode

(2) Programmable waveform generation mode

In this mode, the microcontroller, while counting the set values of Timer Z primary and Timer Z secondary alternately, outputs from the TZOUT pin a waveform whose polarity is inverted each time Timer Z primary or Timer Z secondary underflows. (See Table 1.14.11) A counting starts by counting the value set in the Timer Z primary. Figure 1.14.26 shows Timer Y, Z mode register and Timer Y, Z waveform output control register in this mode. The Timer Z operates in the same way as the Timer Y in this mode. See Figure 1.14.19 shown the Timer Y operating example in programmable waveform generation mode.

Table 1.14.11. Specifications of programmable waveform generating mode

Item	Specification
Count source	f_1, f_8, Timer Y underflow, f_{C32}
Count operation	• Down count • When the timer underflows, it reloads the contents of primary reload register and secondary reload register alternately before continuing counting. • When a counting stops, the timer reloads the content of the reload register before it stops.
Divide ratio	fi/(n+1)/((m+1)+(l+1)) n : Set value of Prescaler Z, m: Set value of Timer Z primary, l: Set value of Timer Z secondary
Count start condition	Count start flag is set (=1)
Count stop condition	Count start flag is reset (=0) (Note 1)
Interrupt request generation timing	When Timer Z underflows during secondary period
TZOUT pin function	Pulse output (Note 2)
INT0 pin function	Programmable I/O port, or external interrupt input pin
Read from timer	Count value can be read out by reading Timer Z primary register. Same applies to Prescaler Z register. (Note 3)
Write to timer	When a value is written to Timer Z primary register, it is written to only reload register. Same applies to Timer Z secondary register and Prescaler Z register. (Note 4)
Select function	• Output level latch select function The output level of an waveform being counted during primary and secondary periods is selectable. • Programmable waveform generation output switching function Can select either programmable waveform or the value of Port $P3_1$ register for output. (Note 5) • Waveform extend function(Note 6) The waveform output primary and secondary periods can each be extended 0.5 cycles of the count source Frequency when waveform extended: 2xfi/((2x(m+1))+(2x(l+1))+TZPUM0+TZPUM1) Duty: (2x(m+1)+TZPUM0)/((2x(m+1)+TZPUM0)+(2x(l+1)+TZPUM1)) m: set value of Timer Z primary, l: set value of Timer Z secondary TZPUM0: Timer Z primary waveform extension control bit TZPUM1: Timer Z secondary waveform extension control bit

Note 1: When the count is stopped, the Timer Z interrupt request flag becomes "1" and an interrupt may occur. Thus, interrupts must be disabled before the count is stopped. Furthermore, set the Timer Z interrupt request bit to "0" before starting counting again.

Note 2: When the counting stopped, the pin is the secondary period output level.

Note 3: Even when counting the secondary period, read out the Timer Z primary register.

Note 4: The set value of Timer Z secondary register and waveform extension control bits as well as Timer Z primary register are made effective by writing a value to the Timer Z primary register. The written values are reflected to the waveform output from the next primary period after writing to the Timer Z primary register.

Note 5: The switching of output is synchronized with a timer Z secondary underflow.

Note 6: When using the waveform extend function, the Prescaler Z register must be set to "00_{16}".
When selecting Timer Y underflow and f_1 for the count source, the waveform extend function cannot be used.

Timer Y, Z mode register

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

Symbol: TYZMR
Address: 0080₁₆
When reset: 000000X0₂

Bit symbol	Bit name	Function	R	W
TYMOD0	Timer Y-related bit		○	○
	Nothing is assigned. When write, set "0". When read, the content is "0".		—	—
TYWC	Timer Y-related bits		○	○
TYS			○	○
TZMOD0	Timer Z operation mode bit	b5 b4 0 1 : Programmable waveform generation mode (Note 1)	○	○
TZMOD1			○	○
TZWC	Timer Z write control bit	Set to "1" in programmable waveform generation mode	○	○
TZS	Timer Z count start flag	0 : Stops counting (Note 2) 1 : Starts counting	○	○

Note 1: When selecting programmable waveform generation mode, output is set for Port $P3_1$ regardless of the value of the direction register.
Note 2: When this bit is cleared to "0", the timer reloads the content of the reload register before it stops. Read out the count value before you stop the timer.

Timer Y, Z waveform output control register

b7	b6	b5	b4	b3	b2	b1	b0

Symbol: PUM
Address: 0084₁₆
When reset: 00₁₆

Bit symbol	Bit name	Function	R	W
TYPUM 0	Timer Y-related bits		○	○
TYPUM 1			○	○
TZPUM0	Timer Z primary waveform extension control bit	0 : No waveform extension 1 : Waveform extension (Note)	○	○
TZPUM1	Timer Z secondary waveform extension control bit	0 : No waveform extension 1 : Waveform extension (Note)	○	○
TYOPL	Timer Y-related bit		○	○
TZOP L	Timer Z output level latch	0 : Outputs "H" for the period set by Timer Z primary and "L" for the period set by Timer Z secondary. "L" is outputted when the timer is stopped. 1 : Outputs "L" for the period set by Timer Z primary and "H" for the period set by Timer Z secondary. "H" is outputted when the timer is stopped.	○	○
INOSTG	$\overline{INT0}$ pin one-shot trigger control bit	Invalid in programmable waveform generation mode	○	○
INOSEG	$\overline{INT0}$ pin one-shot trigger polarity select bit	Invalid in programmable waveform generation mode	○	○

Note : When setting this bit to "1", the Prescaler Z Register must be set to "00₁₆".

Figure 1.14.26. Timer Y, Z mode register and Timer Y, Z waveform output control register in programmable waveform generation mode

(3) Programmable one-shot generation mode

In this mode, upon software command or external trigger input (input to the $\overline{INT0}$ pin), the microcomputer outputs the one-shot pulse from the TZOUT pin. (See Table 1.14.12) When a trigger occurs, the timer starts operating from the point only once for a given period equal to the set value of the Timer Z primary. Timer Z secondary is unused in this mode. Figure 1.14.27 shows the Timer Y, Z mode register and Timer Y, Z waveform output control register in this mode. Figure 1.14.28 shows the Timer Z operation example in this mode.

Table 1.14.12. Specifications of programmable one-shot generating mode

Item	Specification
Count source	f_1, f_8, Timer Y underflow, f_{C32}
Count operation	• Downcounts the set value of Timer Z primary • When the timer underflows, it reloads the contents of reload register before stopping counting. • When a counting stops, the timer reloads the contents of the reload register before it stops.
Divide ratio	1/(n+1)/(m+1) n : Set value of Prescaler Z, m: Set value of Timer Z primary
Count start condition	• Timer Z one-shot start bit is set (=1) (Note 1) • Valid trigger is input to $\overline{INT0}$ pin (Note 2)
Count stop condition	• When reloading is completed after count value was set to "00$_{16}$" • When Count start flag is reset (=0) (Note 3) • Timer Z one-shot start bit is reset (=0) (Note 3)
Interrupt request generation timing	When count value becomes "00$_{16}$"
TZOUT pin function	Pulse output
$\overline{INT0}$ pin function	Programmable I/O port, external interrupt input pin, or external trigger input pin
Read from timer	Count value can be read out by reading Timer Z primary register. Same applies to Prescaler Z register.
Write to timer	When a value is written to Timer Z primary register, it is written to only reload register. Same applies to Prescaler Z register. (Note 4)
Select function	• Output level latch select function The output level of one-shot pulse waveform is selectable. • $\overline{INT0}$ pin one-shot trigger control function and polarity select function The trigger input from the $\overline{INT0}$ pin can be set to valid or invalid. Also, the valid trigger's polarity can be chosen to be the rising edge, falling edge, or rising and falling both edges. • Waveform extend function The one-shot pulse waveform can be extended 0.5 cycles of the count source (Note 5) Frequency when waveform extended: 2/(n+1)/(2x(m+1)+TZPUM0) n: set value of Prescaler Z, m: set value of Timer Z primary TZPUM0: Timer Z primary waveform extension control bit

Note 1: Count start flag must have been set to "1".

Note 2: Count start flag must have been set to "1", $\overline{INT0}$ input enable bit [INT0EN] to "1", and $\overline{INT0}$ one-shot trigger control bit to "1".

Note 3: When the count is stopped by writing 0 to the count start flag or Timer Z one-shot start bit, the Timer Z interrupt request flag becomes "1" and an interrupt may occur. Thus, interrupts must be disabled before the count is stopped. Furthermore, set the Timer Z interrupt request bit to "0" before starting counting again.

Note 4: Each set value becomes effective by writing to the Timer Z primary register. And the set values are reflected collectively beginning with the next one-shot pulse after writing to the Timer Z primary.

Note 5: When using the waveform extend function, the Prescaler Z register must be set to "00$_{16}$".
When selecting Timer Y underflow and f_1 for the count source, the waveform extend function cannot be used.

Timer Y, Z mode register

b7	b6	b5	b4	b3	b2	b1	b0	Symbol	Address	When reset
	1	1	0			X		TYZMR	0080_{16}	$000000X0_2$

Bit symbol	Bit name	Function	R	W
TYMOD0	Timer Y-related bit		○	○
	Nothing is assigned. When write, set "0". When read, the content is "0".		—	—
TYWC	Timer Y-related bits		○	○
TYS			○	○
TZMOD0	Timer Z operation mode bit	b5 b4 1 0 : Programmable one-shot generation mode (Note 1)	○	○
TZMOD1			○	○
TZWC	1: Set to "1" in programmable one-shot generation mode		○	○
TZS	Timer Z count start flag	0 : Stops counting (Note 2) 1 : Starts counting	○	○

Note 1: When selecting programmable one-shot generation mode, output is set for Port $P3_1$ regardless of the value of the direction register.
Note 2: When this bit is cleared to "0", the timer reloads the content of the reload register before it stops. Read out the count value before you stop the timer.

Timer Y, Z waveform output control register

b7	b6	b5	b4	b3	b2	b1	b0	Symbol	Address	When reset
								PUM	0084_{16}	00_{16}

Bit symbol	Bit name	Function	R	W
TYPUM0	Timer Y-related bits		○	○
TYPUM1			○	○
TZPUM0	Timer Z primary waveform extension control bit	0 : No waveform extension 1 : Waveform extension (Note 1)	○	○
TZPUM1	Timer Z secondary waveform extension control bit	Invalid in programmable one-shot generation mode	○	○
TYOPL	Timer Y-related bit		○	○
TZOPL	Timer Z output level latch	0 : Outputs "H" level one-shot pulse. "L" is outputted when the timer is stopped. 1 : Outputs "L" level one-shot pulse "H" is outputted when the timer is stopped.	○	○
INOSTG	$\overline{INT0}$ pin one-shot trigger control bit	0 : $\overline{INT0}$ pin one-shot trigger invalid 1 : $\overline{INT0}$ pin one-shot trigger valid (Note 3)	○	○
INOSEG	$\overline{INT0}$ pin one-shot trigger polarity select bit (Note 2)	0 : Edge trigger at falling edge 1 : Edge trigger at rising edge	○	○

Note 1: When setting this bit to "1", the Prescaler Z Register must be set to "00_{16}".
Note 2: This bit is valid only when $\overline{INT0}$ input polarity select bit (bit 1 at address 0096_{16}) is "0" (one-edge).
Note 3: When changing this bit to "1", set the $\overline{INT0}$ input filter select bit (bits 0, 1 of address $1E_{16}$).

Figure 1.14.27. Timer Y, Z mode register and Timer Y, Z waveform output control register in programmable one-shot generation mode

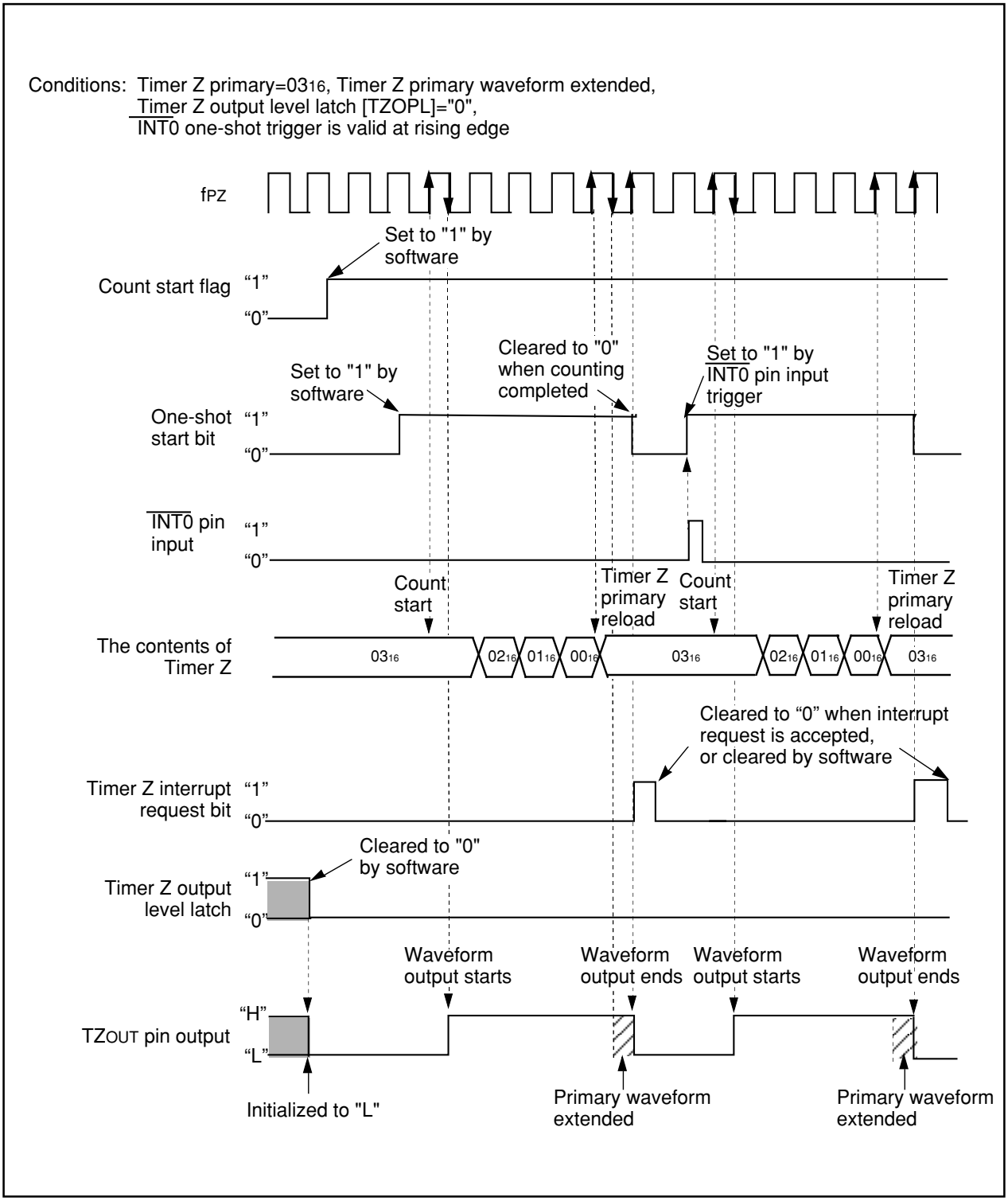


Figure 1.14.28. Operation example in programmable one-shot generation mode

(4) Programmable wait one-shot generation mode

In this mode, upon software command or external trigger input (input to the $\overline{INT0}$ pin), the microcomputer outputs the one-shot pulse from the TZ_{OUT} pin after waiting for a given length of time. (See Table 1.14.13) When a trigger occurs, from this point, the timer starts outputting pulses only once for a given length of time equal to the Timer Z primary set value after waiting for a given length of time equal to the Timer Z primary set value. Figure 1.14.29 shows the Timer Y, Z mode register and Timer Y, Z waveform output control register in this mode. Figure 1.14.30 shows the Timer Z operation example in this mode.

Table 1.14.13. Specifications of programmable wait one-shot generating mode

Item	Specification
Count source	f_1, f_8, Timer Y underflow, f_{C32}
Count operation	• Downcounts the set value of Timer Z primary • When Timer Z primary underflows, the contents of Timer Z secondary is reloaded before continuing counting. • When Timer Z secondary underflows, the contents of Timer Z primary is reloaded before stopping counting. • When a counting stops, the timer reloads the contents of the reload register before it stops.
Wait time	fi x (n+1) x (m+1), n : Set value of Prescaler Z, m: Set value of Timer Z primary
One-shot pulse output time	fi x (n+1) x (l+1), n : Set value of Prescaler Z, l: Set value of Timer Z secondary
Count start condition	• Timer Z one-shot start bit is set (=1) (Note 1) • Valid trigger is input to $\overline{INT0}$ pin (Note 2)
Count stop condition	• When reloading is completed after count value at counting Timer Z secondary was set to "00$_{16}$" • When Count start flag is reset (=0) (Note 3) • Timer Z one-shot start bit is reset (=0) (Note 3)
Interrupt request generation timing	When count value at counting Timer Z secondary becomes "00$_{16}$"
TZ_{OUT} pin function	Pulse output
$\overline{INT0}$ pin function	Programmable I/O port, external interrupt input pin, or external trigger input pin
Read from timer	Count value can be read out by reading Timer Z primary register. Same applies to Prescaler Z register.
Write to timer	When a value is written to Timer Z primary register, it is written to only reload register. Same applies to Prescaler Z register. (Note 4)
Select function	• Output level latch select function The output level of one-shot pulse waveform is selectable. • $\overline{INT0}$ pin one-shot trigger control function and polarity select function The trigger input from the $\overline{INT0}$ pin can be set to valid or invalid. Also, the valid trigger's polarity is selectable: rising edge, falling edge, or rising and falling both edges. • Waveform extend function Waiting time and one-shot pulse waveform can each be extended 0.5 cycles of the count source (Note 5) Waiting time when waveform extended: fi x (n+1) x (2x(m+1)+TZPUM0)/2 One-shot pulse output time when waveform extended: fi x (n+1) x (2x(l+1)+TZPUM1)/2 n: set value of Prescaler Z, m: set value of Timer Z primary, l: set value of Timer Z secondary TZPUM0: Timer Z primary waveform extension control bit, TZPUM1: Timer Z secondary waveform extension control bit

Note 1: Count start flag must have been set to "1".
Note 2: Count start flag must have been set to "1", $\overline{INT0}$ input enable bit [INT0EN] to "1", and $\overline{INT0}$ one-shot trigger control bit to "1".
Note 3: When the count is stopped by writing 0 to the count start flag or Timer Z one-shot start bit, the Timer Z interrupt request flag becomes "1" and an interrupt may occur. Thus, interrupts must be disabled before the count is stopped. Furthermore, set the Timer Z interrupt request bit to "0" before starting counting again.
Note 4: Each set value becomes effective by writing to the Timer Z primary register. And the set values are reflected collectively beginning with the next one-shot pulse after writing to the Timer Z primary.
Note 5: When using the waveform extend function, the Prescaler Z register must be set to "00$_{16}$".
When selecting Timer Y underflow and f_1 for the count source, the waveform extend function cannot be used.

Timer Y, Z mode register

b7	b6	b5	b4	b3	b2	b1	b0	Symbol	Address	When reset
	1	1	1					TYZMR	0080_{16}	$000000X0_2$

Bit symbol	Bit name	Function	R	W
TYMOD0	Timer Y-related bit		○	○
	Nothing is assigned. When write, set "0". When read, the content is "0".		—	—
TYWC	Timer Y-related bits		○	○
TYS			○	○
TZMOD0	Timer Z operation mode bit	b5 b4 1 1 : Programmable wait one-shot generation mode (Note 1)	○	○
TZMOD1			○	○
TZWC	1: Set to "1" in programmable wait one-shot generation mode		○	○
TZS	Timer Z count start flag	0 : Stops counting (Note 2) 1 : Starts counting	○	○

Note 1: When selecting programmable wait one-shot generation mode, output is set for Port $P3_1$ regardless of the value of the direction register.
Note 2: When this bit is cleared to "0", the timer reloads the content of the reload register before it stops. Read out the count value before you stop the timer.

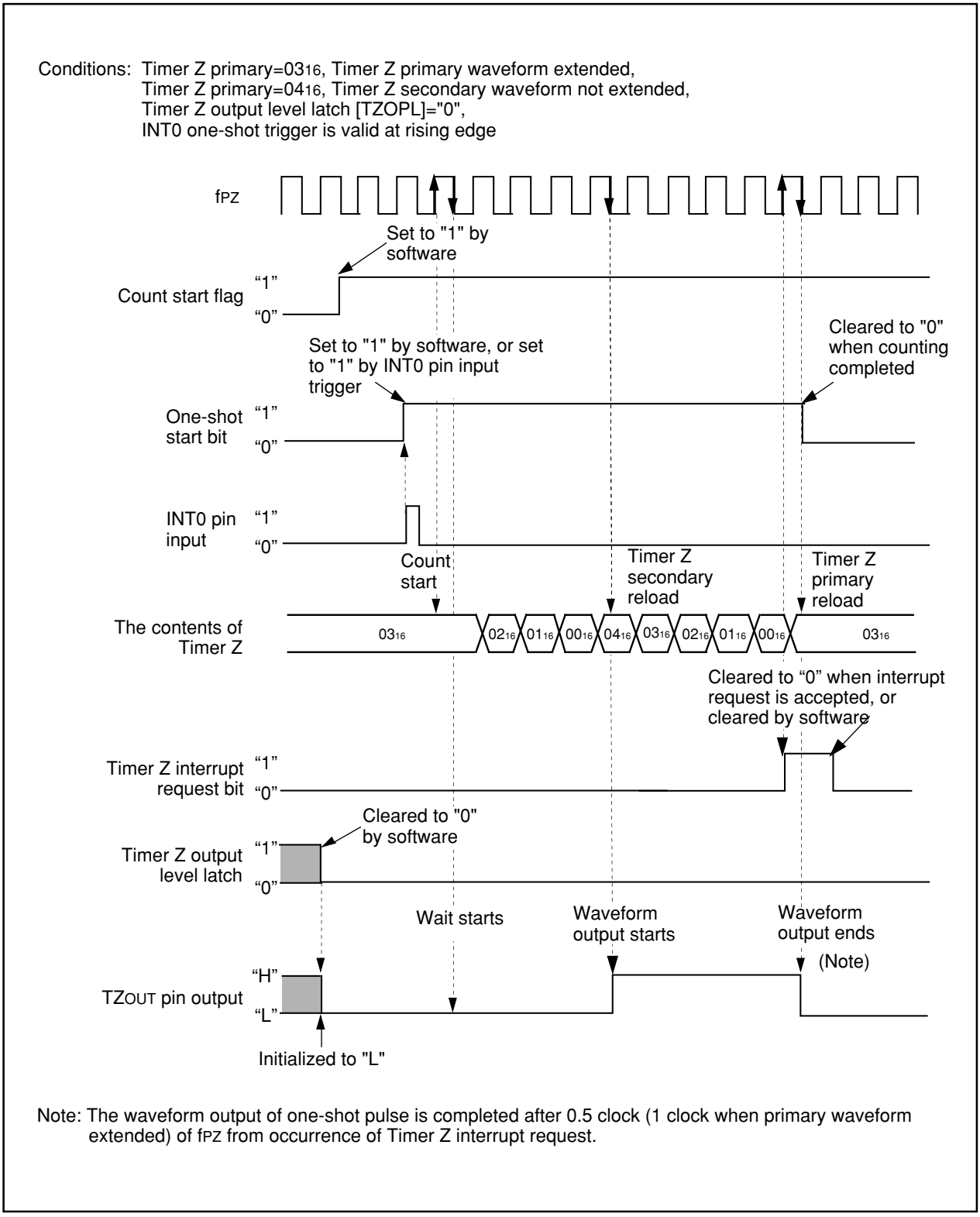
Timer Y, Z waveform output control register

b7	b6	b5	b4	b3	b2	b1	b0	Symbol	Address	When reset
								PUM	0084_{16}	00_{16}

Bit symbol	Bit name	Function	R	W
TYPUM0	Timer Y-related bits		○	○
TYPUM1			○	○
TZPUM0	Timer Z primary waveform extension control bit	0 : No waveform extension 1 : Waveform extension (Note 1)	○	○
TZPUM1	Timer Z secondary waveform extension control bit	0 : No waveform extension 1 : Waveform extension (Note 1)	○	○
TYOPL	Timer Y-related bit		○	○
TZOPL	Timer Z output level latch	0 : Outputs "H" level one-shot pulse. "L" is outputted when the timer is stopped. 1 : Outputs "L" level one-shot pulse "H" is outputted when the timer is stopped.	○	○
INOSTG	$\overline{INT}0$ pin one-shot trigger control bit	0 : $\overline{INT}0$ pin one-shot trigger invalid 1 : $\overline{INT}0$ pin one-shot trigger valid (Note 3)	○	○
INOSEG	$\overline{INT}0$ pin one-shot trigger polarity select bit (Note 2)	0 : Edge trigger at falling edge 1 : Edge trigger at rising edge	○	○

Note 1: When setting this bit to "1", the Prescaler Z Register must be set to "00_{16}".
Note 2: This bit is valid only when $\overline{INT}0$ input polarity select bit (bit 1 at address 0096_{16}) is "0" (one-edge).
Note 3: When changing this bit to "1", set the $\overline{INT}0$ input filter select bit (bits 0, 1 of address $1E_{16}$).

Figure 1.14.29. Timer Y, Z mode register and Timer Y, Z waveform output control register in programmable wait one-shot generation mode



Note: The waveform output of one-shot pulse is completed after 0.5 clock (1 clock when primary waveform extended) of fPZ from occurrence of Timer Z interrupt request.

Figure 1.14.30. Operation example in programmable wait one-shot generation mode

Timer C

Timer C is a 16-bit free-running timer. Figure 1.14.31 shows the block diagram of Timer C. The Timer C uses an edge input to TCIN pin or the output of 512 fRING divisions as trigger to latch the timer count value and generates an interrupt request. The TCIN input has a digital filter and this prevents an error caused by noise or so on from occurring. Table 1.14.14 shows Timer C specifications. Figure 1.14.32 shows Timer C-related registers. Figure 1.14.33 shows an operation example of Timer C and timer measurement register.

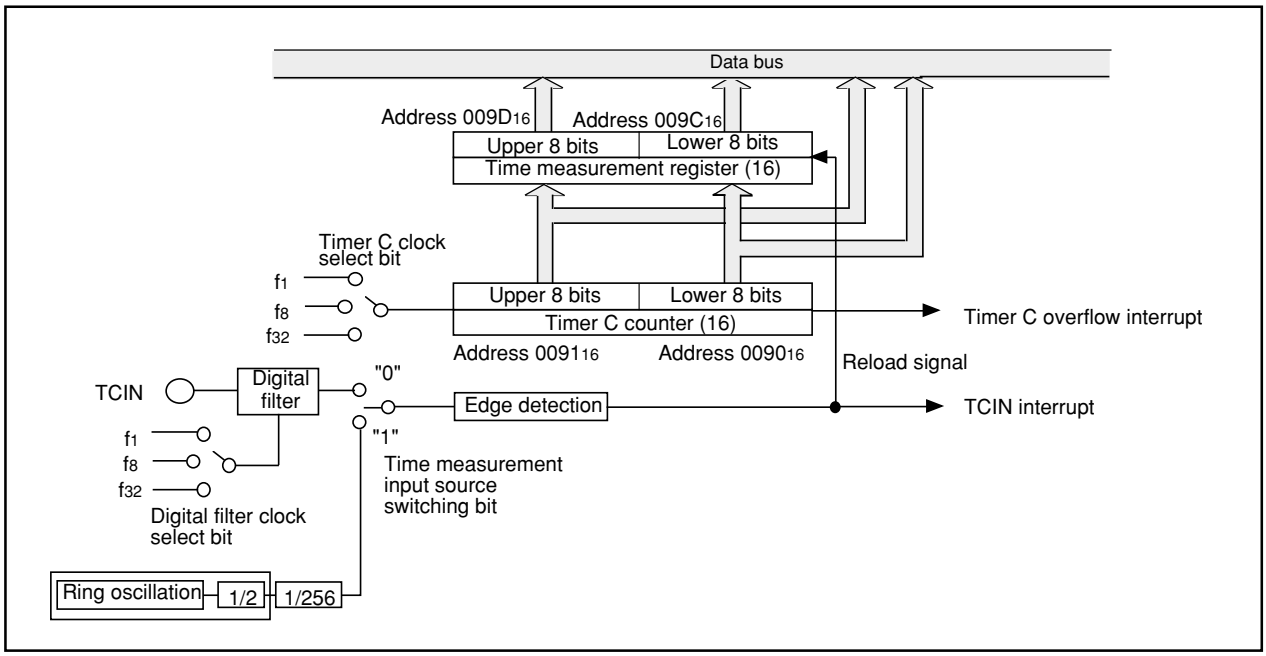


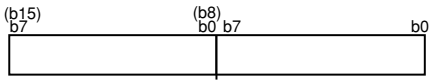
Figure 1.14.31. Block diagram of Timer C

Table 1.14.14. Specifications of Timer C

Item	Specification
Count source	f1, f8, f32
Count operation	• Up count • Transfer counter value to time measurement register at active edge of measurement pulse • Do not reset counter value even if active edge is detected
Count start condition	• Time measurement control bit is set (=1)
Counter stop condition	• Time measurement control bit is reset (=0)
Interrupt request generation timing	• When active edge of measurement pulse is input [TCIN interrupt] • When the time underflows [Timer C interrupt]
TCIN pin function	Measurement pulse input
Count value reset timing	When time measurement control bit is reset (=0)
Read from timer (Note)	• Count value can be read out by reading Timer C. (Note) • Count value at measurement pulse active edge input can be read out by reading time measurement register.
Write to timer	Cannot write to Timer C and time measurement register
Select function	• Measurement pulse active edge: selectable (rising edge/falling edge/both edges) • Measurement pulse: selectable (input from TCIN pin/512 divisions of fRING) • Digital filter sampling frequency: selectable (f1/f8/f32)

Note: The Timer C and the timer measurement register must be read in word-size.

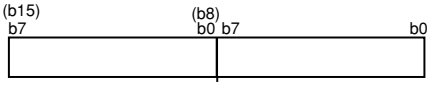
Timer C



Symbol	Address	When reset
TC	0091_{16}, 0090_{16}	Indeterminate

Function	R	W
Internal count source is counted	○	×

Time measurement register

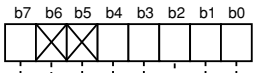


Symbol	Address	When reset
TM	$009C_{16}$, $009D_{16}$	Indeterminate

Function	R	W
When active edge of measurement pulse is input, the count value of Timer C is stored (Note)	○	×

Note: When time measurement is disabled, the value is indeterminate.
After enabling time measurement, the value is indeterminate until the first trigger is generated.

Timer C control register 0

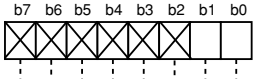


Symbol	Address	When reset
TCC0	$009A_{16}$	$0XX00000_2$

Bit symbol	Bit name	Function	R	W
TCC00	Time measurement control bit	0 : Time measurement disabled 1 : Time measurement enabled	○	○
TCC01 TCC02	Timer C clock select bit (Note 1)	b2 b1 0 0 : f1 (Note 2) 0 1 : f8 1 0 : f32 1 1 : Inhibit	○	○
TCC03 TCC04	Time measurement input edge trigger bit (Note 1)	b4 b3 0 0 : Rising edge 0 1 : Falling edge 1 0 : Both edges 1 1 : Inhibit	○	○
Nothing is assigned. When write, set "0". When read, their contents are "0".			—	—
TCC07	Time measurement input source switching bit (Note 1) (Note 3)	0 : TCIN 1 : RING512	○	○

Note 1: Change this bit when time measurement is disabled.
Note 2: When using an external RC circuit for the main clock, f_1 cannot be selected for the Timer C clock.
Note 3: Set the ring oscillation stop bit (CM14) to “0” before setting this bit to “1”.

Timer C control register 1



Symbol	Address	When reset
TCC1	$009B_{16}$	$XXXXXX11_2$

Bit symbol	Bit name	Function	R	W
TCC10 TCC11	Digital filter clock select bit (Note)	b1 b0 0 0 : Cannot be used 0 1 : f1 1 0 : f8 1 1 : f32	○	○
Nothing is assigned. When write, set "0". When read, their contents are "0".			—	—

Note : Input edge becomes active when the same value from TC_{IN} pin is sampled three times in succession.

Figure 1.14.32. Timer C-related register

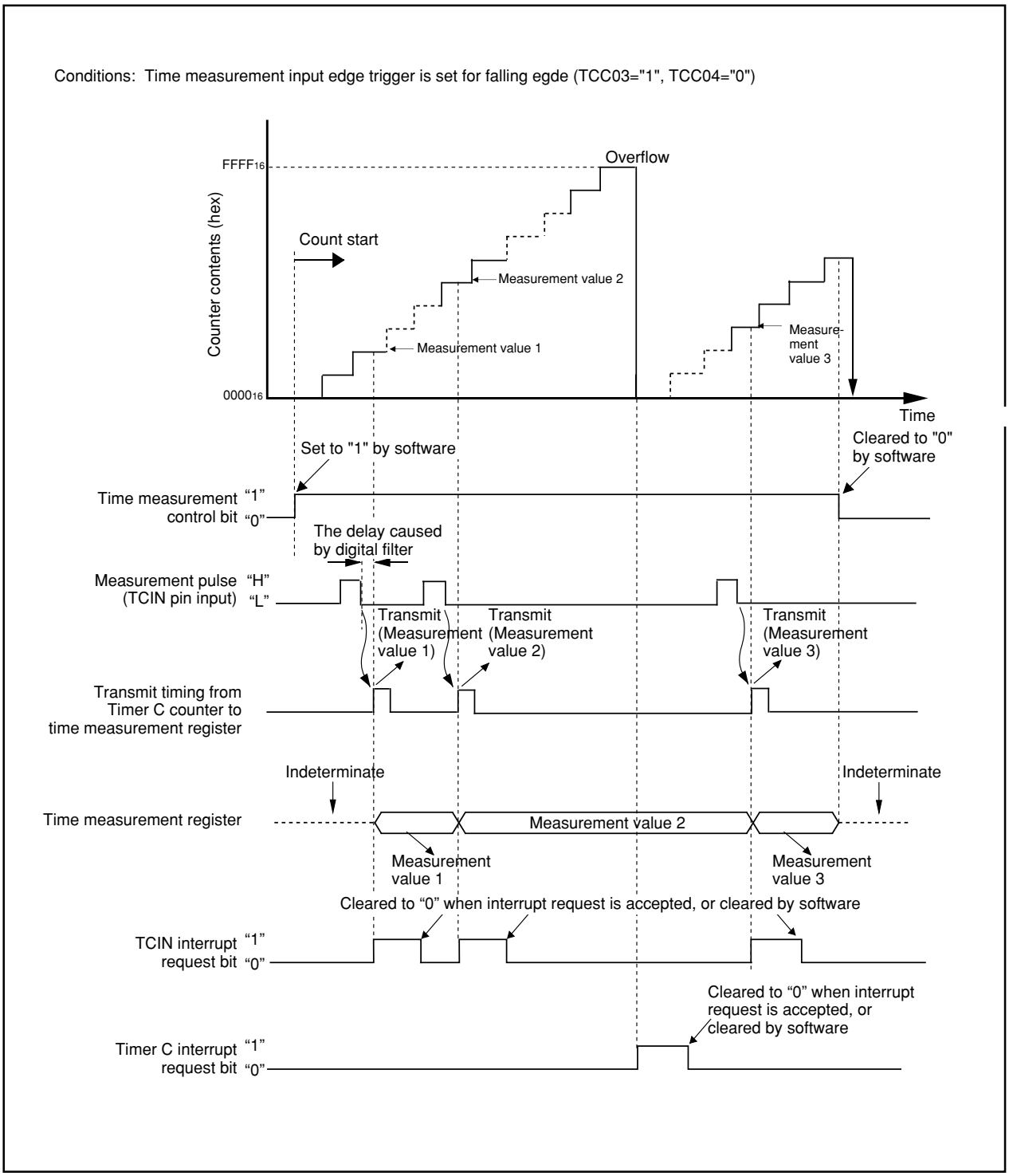


Figure 1.14.33. Operation example of Timer C and time measurement register

Serial I/O

Serial I/O is configured as two channels: UART0 and UART1. UART0 and UART1 each have an exclusive timer to generate a transfer clock, so they operate independently of each other.

Figure 1.15.1 shows the block diagram of UARTi (i=0,1). Figure 1.15.2 shows the block diagram of the transmit/receive unit.

UART0 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 $00A0_{16}$ and $00A8_{16}$) determine whether UART0 is used as a clock synchronous serial I/O or as a UART. Although a few functions are different, UART0 and UART1 have almost the same functions.

Figures 1.15.3 through 1.15.5 show the registers related to UARTi.

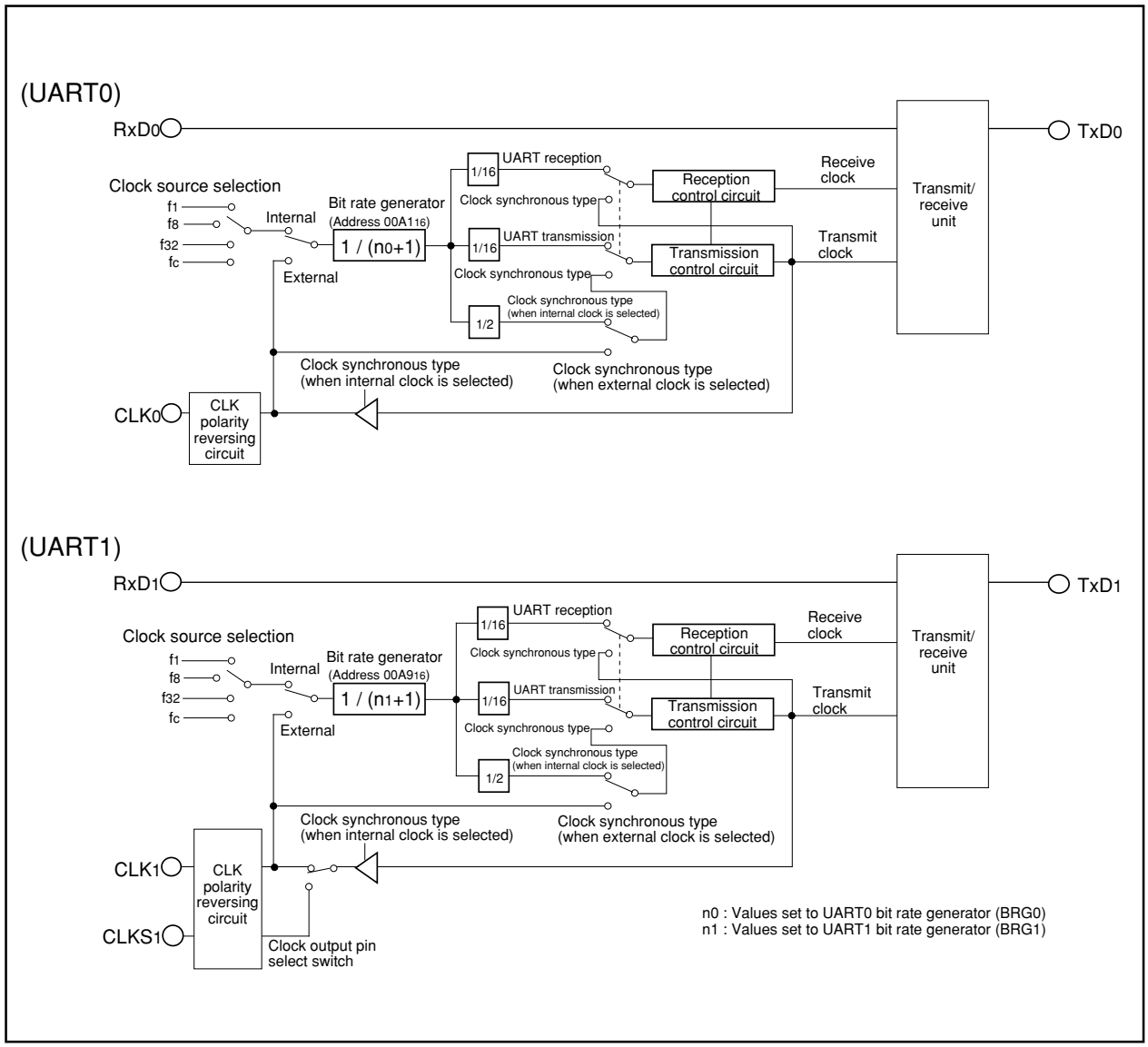


Figure 1.15.1. Block diagram of UARTi (i = 0, 1)

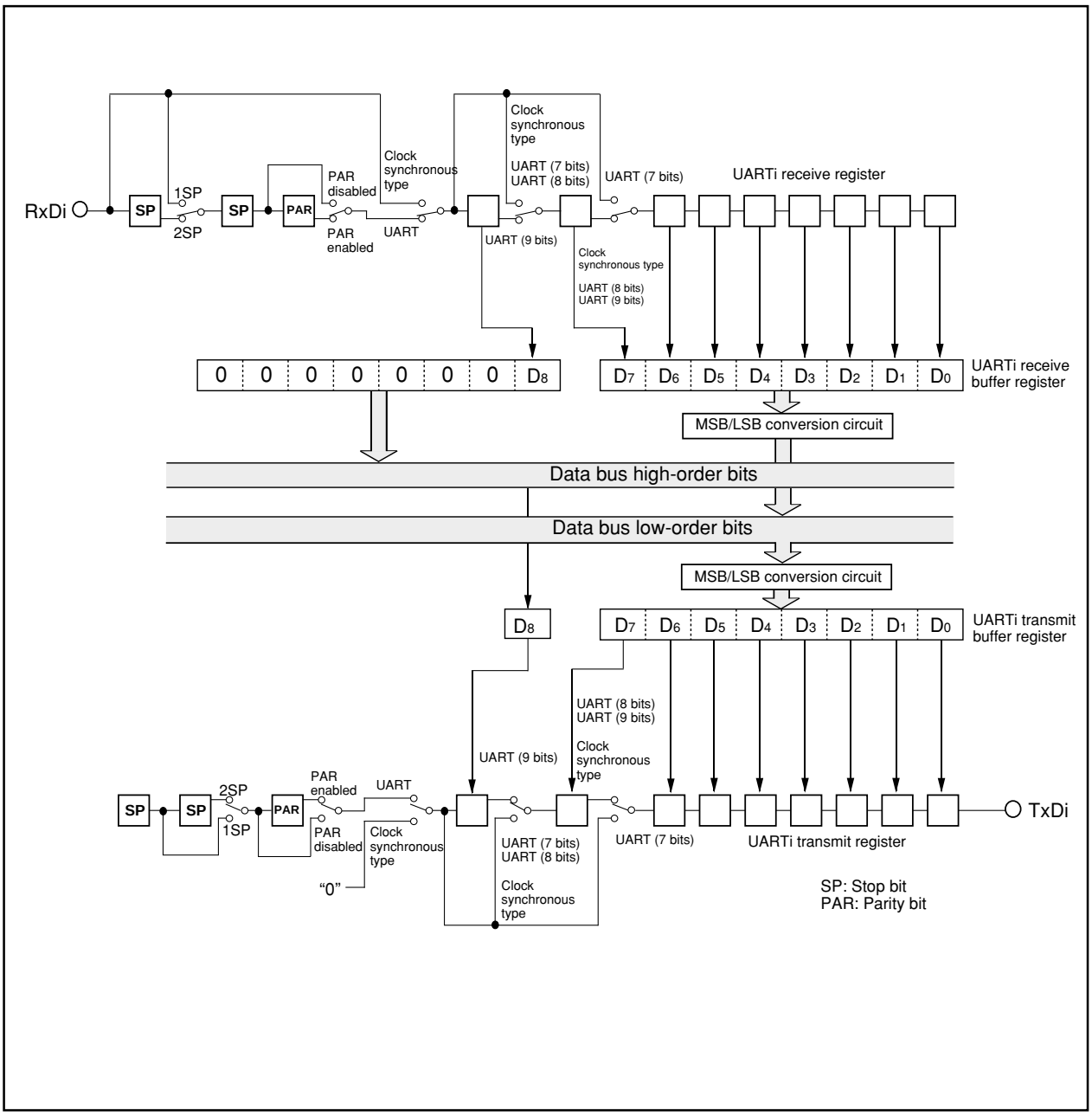
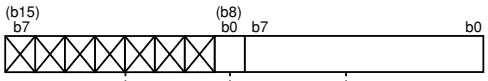


Figure 1.15.2. Block diagram of transmit/receive unit

UARTi transmit buffer register

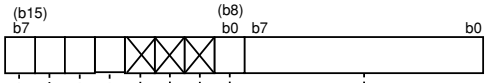


Symbol	Address	When reset
U0TB	$00A3_{16}$, $00A2_{16}$	Indeterminate
U1TB	$00AB_{16}$, $00AA_{16}$	Indeterminate

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

Note: When transfer data length is 9-bit long, write high-byte first then low-byte with byte-size.

UARTi receive buffer register

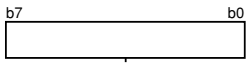


Symbol	Address	When reset
U0RB	$00A7_{16}$, $00A6_{16}$	Indeterminate
U1RB	$00AF_{16}$, $00AE_{16}$	Indeterminate

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 indeterminate.				—	—
OER	Overrun error flag (Note)	0 : No overrun error 1 : Overrun error found	0 : No overrun error 1 : Overrun error found	○	×
FER	Framing error flag (Note)	Invalid	0 : No framing error 1 : Framing error found	○	×
PER	Parity error flag (Note)	Invalid	0 : No parity error 1 : Parity error found	○	×
SUM	Error sum flag (Note)	Invalid	0 : No error 1 : Error found	○	×

Note: Bits 15 through 12 are set to "0" when the serial I/O mode select bits (bit 2 to 0 at addresses $00A0_{16}$ and $00A8_{16}$) are set to "000_2" or receive enable bit 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 $00A6_{16}$, and $00AE_{16}$) is read out or when this register is read out in word-size. When reading data from the UARTi receive buffer, data should be read high-byte first then low-byte using byte-size.

UARTi bit rate generator



Symbol	Address	When reset
U0BRG	$00A1_{16}$	Indeterminate
U1BRG	$00A9_{16}$	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}	×	○

Figure 1.15.3. 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: 00A0$_{16}$, 00A8$_{16}$ — 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 : Inhibited 0 1 1 : Inhibited 1 1 1 : Inhibited	○	○
SMD1		b2 b1 b0 0 0 0 : Serial I/O invalid 0 1 0 : Inhibited 0 1 1 : Inhibited 1 1 1 : Inhibited		○	○
SMD2				○	○
CKDIR	Internal/external clock select bit	0 : Internal clock 1 : External clock (Note)	0 : Internal clock 1 : External clock	○	○
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	○	○
Reserved bit		Must always be set to "0"		○	○

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

UARTi transmit/receive control register 0

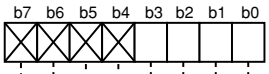
b7	b6	b5	b4	b3	b2	b1	b0
					0		

Symbol: UiC0(i=0,1) — Address: 00A4$_{16}$, 00AC$_{16}$ — When reset: 08$_{16}$

Bit symbol	Bit name	Function (Note) (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 : fc is selected	b1 b0 0 0 : f_1 is selected 0 1 : f_8 is selected 1 0 : f_{32} is selected 1 1 : fc is selected	○	○
CLK1				○	○
Reserved bit		Must set to "0".		○	○
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)	○	×
Nothing is assigned. In an attempt to write to this bit, write "0". The value, if read, turns out to be "0".				–	–
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"	○	○

Figure 1.15.4. Serial I/O-related registers (2)

UARTi transmit/receive control register 1

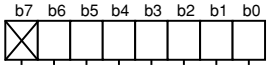


Symbol: UiC1(i=0,1) — Address: 00A5$_{16}$,00AD$_{16}$ — When reset: 02$_{16}$

Bit symbol	Bit name	Function (Note 1) (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 (Note)	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".				—	—

Note : As for the UART1, set the RXD$_1$ input port select bit before setting this bit to reception enabled.

UART transmit/receive control register 2



Symbol: UCON — Address: 00B0$_{16}$ — When reset: 00$_{16}$

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	Invalid	○	○
U1RRM	UART1 continuous receive mode enable bit	0 : Continuous receive mode disabled 1 : Continuous receive mode enable	Invalid	○	○
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 1)	0 : Normal mode (CLK output is CLK0 only) 1 : Transfer clock output from multiple pins function selected	Fixed to "0"	○	○
RXD1EN	RxD$_1$ input port select bit (Note 2)	0 : P3$_7$ 1 : P3$_5$	0 : P3$_7$ 1 : P3$_5$	○	○
Nothing is assigned. In an attempt to write to these bits, write "0". The value, if read, turns out to be "0".				—	—

Note 1: 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 00A0$_{16}$) = "0".
Note 2: For P3$_7$, select "0" for data receive, and "1" for data transfer. And set the direction register of port P3$_7$ to input ("0") when receiving.

Figure 1.15.5. Serial I/O-related registers (3)

(1) Clock synchronous serial I/O mode

The clock synchronous serial I/O mode uses a transfer clock to transmit and receive data. (See Table 1.15.1.) Figure 1.15.6 shows the UARTi transmit/receive mode register.

Table 1.15.1. Specifications of clock synchronous serial I/O mode

Item	Specification
Transfer data format	• Transfer data length: 8 bits
Transfer clock	• When internal clock is selected (bit 3 at address 00A0$_{16}$,00A8$_{16}$ = "0") : fi/ 2(n+1) (Note 1) fi = f1, f8, f32, fc • When external clock is selected (bit 3 at address 00A0$_{16}$,00A8$_{16}$ = "1") : Input from CLKi pin
Transmission start condition	• To start transmission, the following requirements must be met: – Transmit enable bit (bit 0 at address 00A5$_{16}$,00AD$_{16}$) = "1" – Transmit buffer empty flag (bit 1 at addresses 00A5$_{16}$,00AD$_{16}$) = "0" • Furthermore, if external clock is selected, the following requirements must also be met: – CLKi polarity select bit (bit 6 at address 00A4$_{16}$,00AC$_{16}$) = "0": CLKi input level = "H" – CLKi polarity select bit (bit 6 at address 00A4$_{16}$,00AC$_{16}$) = "1": CLKi input level = "L"
Reception start conditio	• To start reception, the following requirements must be met: – Receive enable bit (bit 2 at address 00A5$_{16}$,00AD$_{16}$) = "1" – Transmit enable bit (bit 0 at address 00A5$_{16}$,00AD$_{16}$) = "1" – Transmit buffer empty flag (bit 1 at address 00A5$_{16}$,00AD$_{16}$) = "0" • Furthermore, if external clock is selected, the following requirements must also be met: – CLKi polarity select bit (bit 6 at address 00A4$_{16}$,00AC$_{16}$) = "0": CLKi input level = "H" – CLKi polarity select bit (bit 6 at address 00A4$_{16}$,00AC$_{16}$) = "1": CLKi input level = "L"
Interrupt request generation timing	• When transmitting – Transmit interrupt cause select bit (bit 0 and bit 1 at address 00B0$_{16}$) = "0": Interrupts requested when data transfer from UARTi transfer buffer register to UARTi transmit register is completed – Transmit interrupt cause select bit (bit 0 and bit 1 at address 00B0$_{16}$) = "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
Select function	• CLK polarity selection Whether transmit data is output/input 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 transfer clock can be chosen by software to be output from one of the two pins set • RxD1 input pin selection UART1 RxD1 can be chosen by software to be input to one of the two pins set

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

UARTi transmit/receive mode registers

b7	b6	b5	b4	b3	b2	b1	b0	Symbol	Address	When reset
0					0	0	1	UiMR (i=0,1)	00A0$_{16}$, 00A8$_{16}$	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”.

Figure 1.15.6. UARTi transmit/receive mode register in clock synchronous serial I/O mode

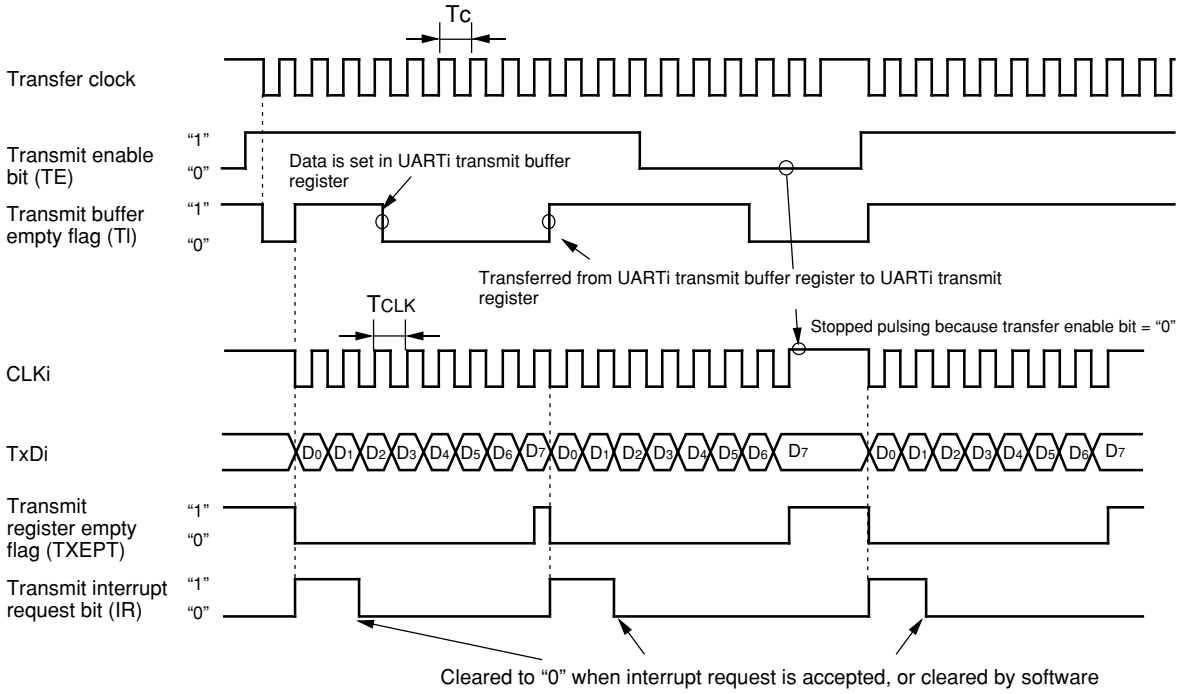
Table 1.15.2 lists the functions of the input/output pins during clock synchronous serial I/O mode. This table shows the pin functions when the transfer clock output from multiple pins is not selected. Note that for a period from when the UARTi operation mode is selected to when transfer starts, the TxDi pin outputs an “H”. (If the N-channel open-drain is selected, this pin is in floating state.)

Table 1.15.2. Input/output pin functions in clock synchronous serial I/O mode

Function	Pin name	Method of selection	Remarks
Serial data output	TxD$_0$ (P1$_4$)	——	Port P1$_4$ cannot be used as an I/O port even when performing only serial data input but not serial data output)
	TxD$_1$ (P3$_7$)	RxD$_1$ input pin select bit (bit 6 at address 00B0$_{16}$)= “1”	Port P3$_7$ cannot be used as an I/O port even when performing only serial data input but not serial data output)
Serial data input	RxD$_0$ (P1$_5$)	Port P1$_5$ direction register (bit 5 at address 00E1$_{16}$)= “0”	Port P1$_5$ can be used as an I/O port when performing only serial data output but not serial data input)
	RxD$_1$ (P3$_5$)	Port P3$_5$ direction register (bit 5 at address 00E7$_{16}$)= “0” RxD$_1$ input pin select bit (bit 6 at address 00B0$_{16}$)= “1”	Port P3$_5$ can be used as an I/O port when performing only serial data output but not serial data input)
	RxD$_1$ (P3$_7$)	Port P3$_7$ direction register (bit 7 at address 00E7$_{16}$)= “0” RxD$_1$ input pin select bit (bit 6 at address 00B0$_{16}$)= “0”	When setting Port P3$_7$ as RxD$_1$, serial data output cannot be performed. Port P3$_5$ can be used as an I/O port.
Transfer clock output	CLKi (P1$_6$, P3$_6$)	Internal/external clock select bit (bit 3 at addresses 00A0$_{16}$ and 00A8$_{16}$) = “0”	
Transfer clock input	CLKi (P1$_6$, P3$_6$)	Internal/external clock select bit (bit 3 at address 00A0$_{16}$ and 00A8$_{16}$) = “1” Ports P1$_6$ and P3$_6$ direction register (bit 6 at address 00E3$_{16}$ and 00E7$_{16}$) = “0”	

(When transfer clock output from multiple pins is not selected)

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

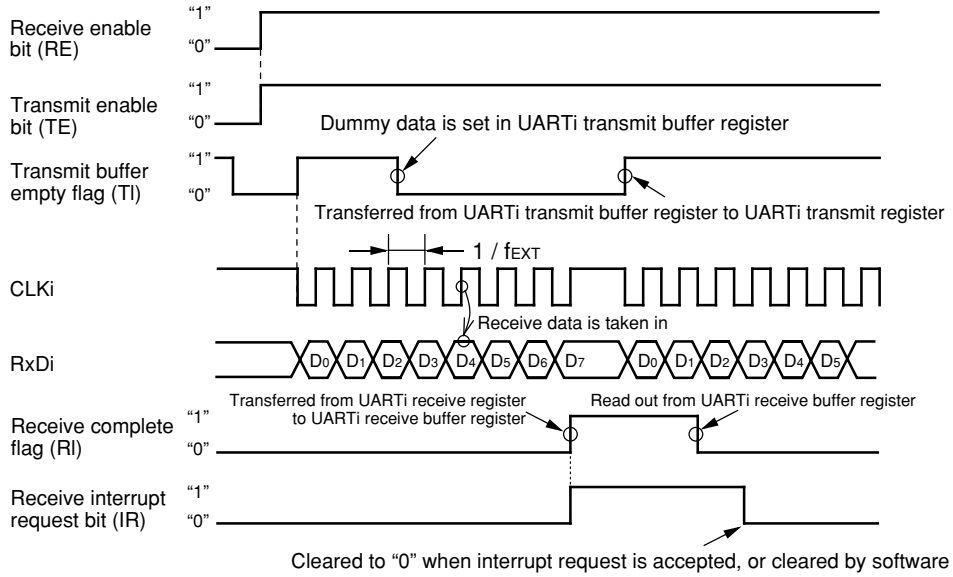


Shown in ( ) are bit symbols.

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

$Tc = T_{CLK} = 2(n + 1) / fi$
fi: frequency of BRGi count source (f1, f8, f32, fc)
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.
- CLK polarity select bit = "0".

fEXT: frequency of external clock

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

Figure 1.15.7. Typical transmit/receive timings in clock synchronous serial I/O mode

(a) Polarity select function

As shown in Figure 1.15.8, the CLK polarity select bit (bit 6 at addresses 00A4$_{16}$ and 00AC$_{16}$) allows selection of the polarity of the transfer clock.

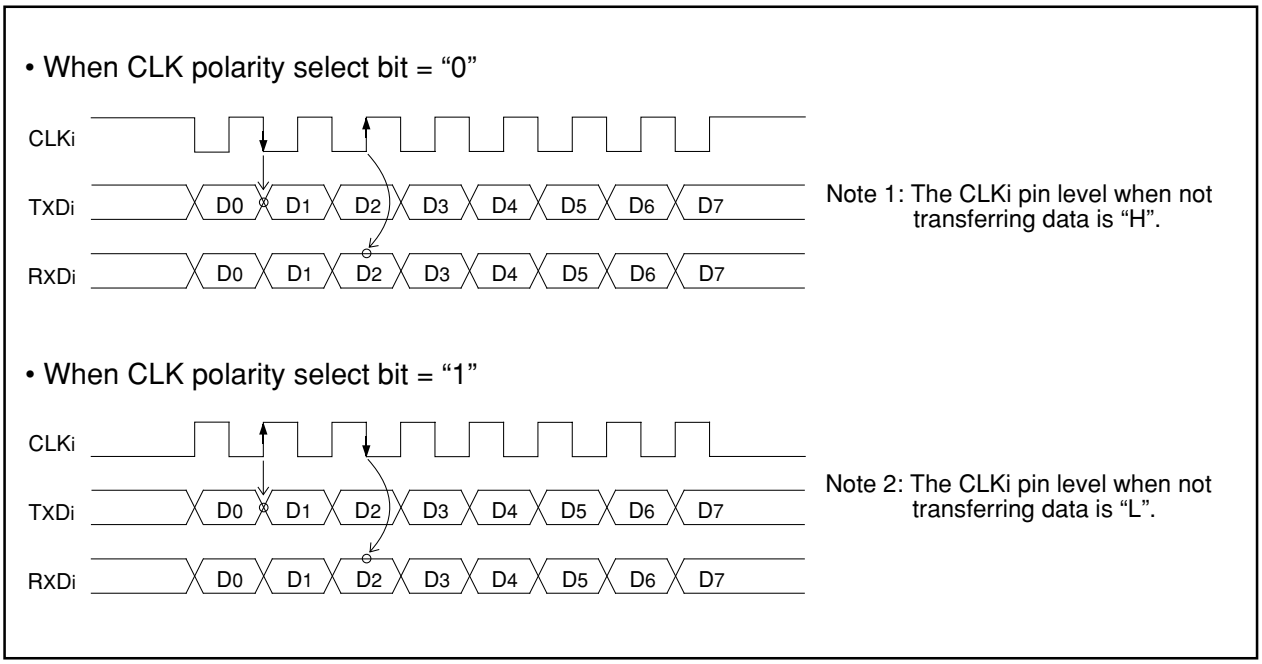


Figure 1.15.8. Polarity of transfer clock

(b) LSB first/MSB first select function

As shown in Figure 1.15.9, when the transfer format select bit (bit 7 at addresses 00A4$_{16}$ and 00AC$_{16}$) = “0”, the transfer format is “LSB first”; when the bit = “1”, the transfer format is “MSB first”.

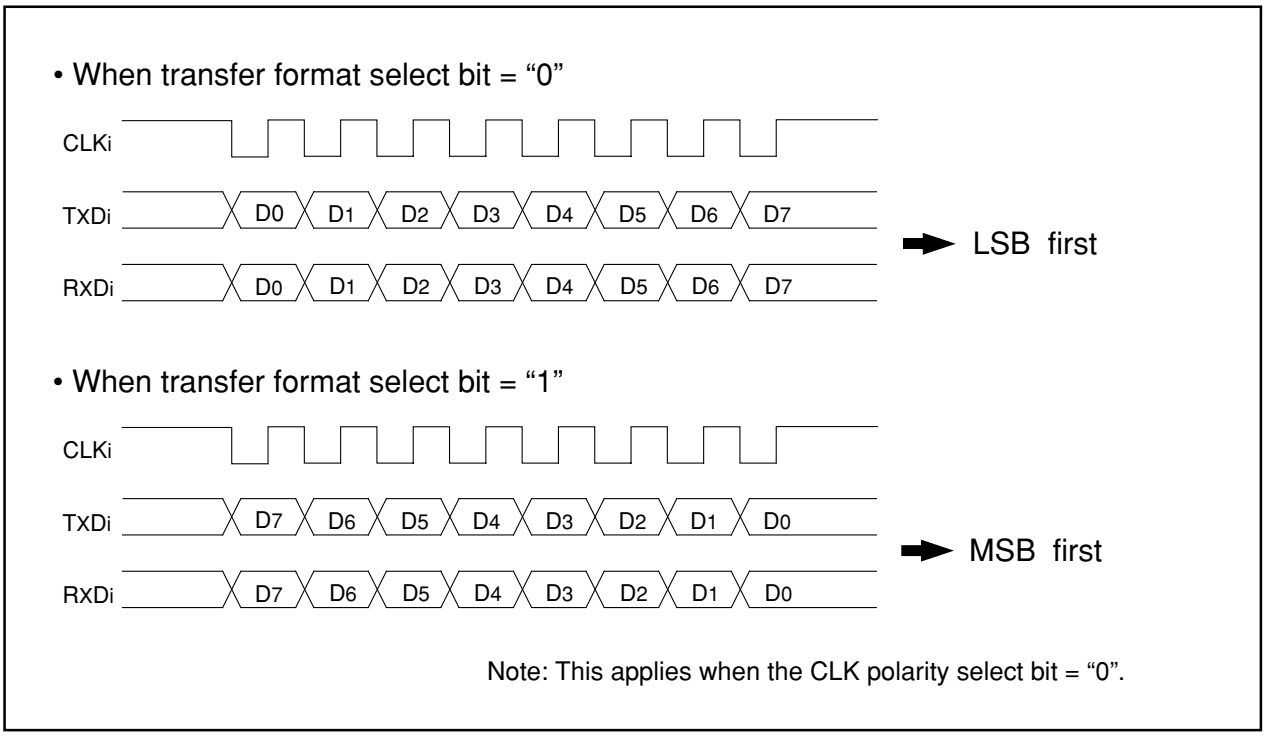


Figure 1.15.9. Transfer format

(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 00B0$_{16}$). (See Figure 1.15.10.)
The multiple pins function is valid only when the internal clock is selected for UART1.

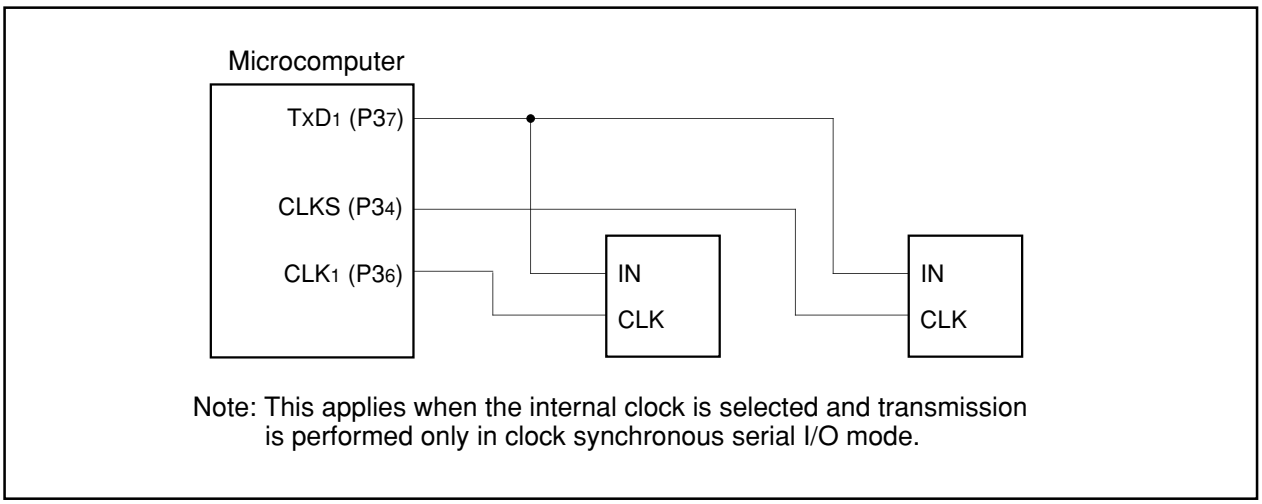


Figure 1.15.10. The transfer clock output from the multiple pins function usage

(d) Continuous receive mode

If the continuous receive mode enable bit (bits 2 and 3 at address 00B0$_{16}$) 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) RxD1 input pin selection function (UART1)

This function allows the setting two RxD1 input pins and choosing one of the two to input serial data by using the RxD1 input pin select bit (bits 6 at address 00B0$_{16}$).
When selecting "1" (P35) for RxD1 input pin select bit, P37 functions as TxD1 output pin. When selecting "0" (P37), serial data output cannot be performed. However, P35 can be used as an input/output port.

(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. (See Table 1.15.3.) Figure 1.15.11 shows the UARTi transmit/receive mode register.

Table 1.15.3. Specifications of UART Mode

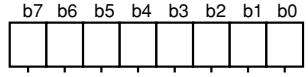
Item	Specification
Transfer data format	• Character bit (transfer data): 7 bits, 8 bits, or 9 bits as selected • Start bit: 1 bit • Parity bit: Odd, even, or nothing as selected • Stop bit: 1 bit or 2 bits as selected
Transfer clock	• When internal clock is selected (bit 3 at addresses $00A0_{16}$, $00A8_{16}$ = "0") : $f_i/16(n+1)$ (Note 1) $f_i = f_1, f_8, f_{32}, f_C$ • When external clock is selected (bit 3 at addresses $00A0_{16}$="1") : $f_{EXT}/16(n+1)$ (Note 1) (Note 2)
Transmission start condition	• To start transmission, the following requirements must be met: - Transmit enable bit (bit 0 at addresses $00A5_{16}$, $00AD_{16}$) = "1" - Transmit buffer empty flag (bit 1 at addresses $00A5_{16}$, $00AD_{16}$) = "0"
Reception start condition	• To start reception, the following requirements must be met: - Receive enable bit (bit 2 at addresses $00A5_{16}$, $00AD_{16}$) = "1" - Start bit detection
Interrupt request generation timing	• When transmitting - Transmit interrupt cause select bits (bits 0,1 at address $00B0_{16}$) = "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 $00B0_{16}$) = "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
Select function	• RxD_1 input pin selection UART1 RxD_1 can be chosen by software to be input to one of the two pins set

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.

UARTi transmit / receive mode registers



Symbol: UiMR(i=0,1) — Address: 00A0$_{16}$, 00A8$_{16}$ — 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 1)	○	○
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	○	○
Reserved bit		Must always be set to "0"	○	○

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

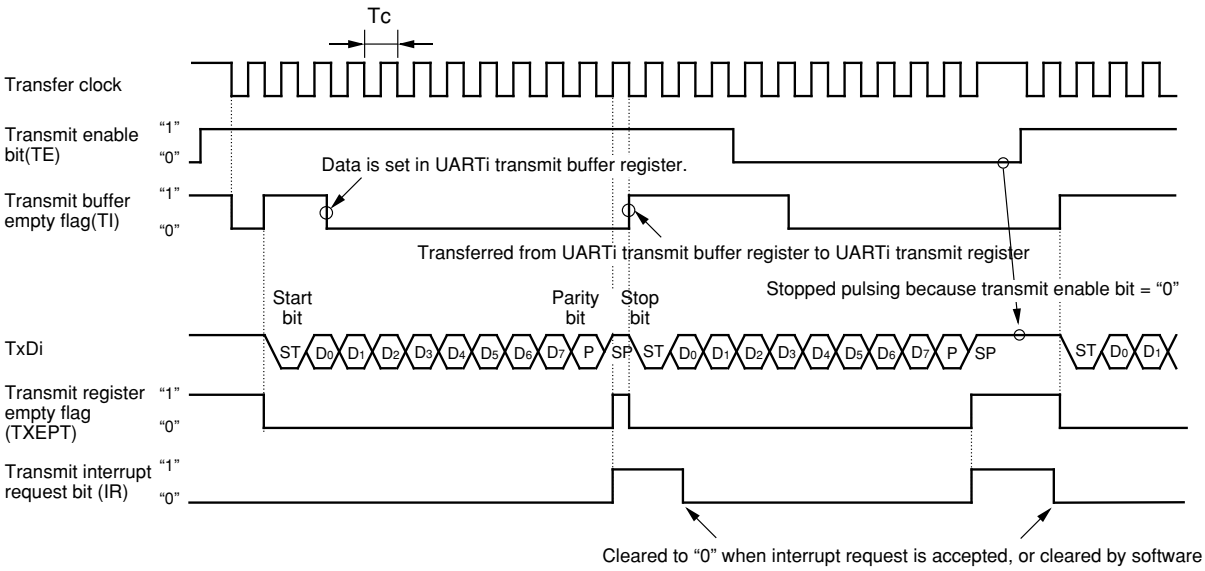
Figure 1.15.11. UARTi transmit/receive mode register in UART mode

Table 1.15.4 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 a "H". (If the N-channel open-drain is selected, this pin is in floating state.)

Table 1.15.4. Input/output pin functions in UART mode

Function	Pin name	Method of selection	Remarks
Serial data output	TxD$_0$ (P1$_4$)	——	Port P1$_4$ cannot be used as an I/O port even when performing only serial data input but not serial data output)
	TxD$_1$ (P3$_7$)	RxD$_1$ input pin select bit (bit 6 at address 00B0$_{16}$)= "1"	Port P3$_7$ cannot be used as an I/O port even when performing only serial data input but not serial data output)
Serial data input	RxD$_0$ (P1$_5$)	Port P1$_5$ direction register (bit 5 at address 00E1$_{16}$)= "0"	Port P1$_5$ can be used as an I/O port when performing only serial data output but not serial data input)
	RxD$_1$ (P3$_5$)	Port P3$_5$ direction register (bit 5 at address 00E7$_{16}$)= "0" RxD$_1$ input pin select bit (bit 6 at address 00B0$_{16}$)= "1"	Port P3$_5$ can be used as an I/O port when performing only serial data output but not serial data input)
	RxD$_1$ (P3$_7$)	Port P3$_7$ direction register (bit 7 at address 00E7$_{16}$)= "0" RxD$_1$ input pin select bit (bit 6 at address 00B0$_{16}$)= "0"	When setting Port P3$_7$ as RxD$_1$, serial data output cannot be performed. Port P3$_5$ can be used as an I/O port.
Transfer clock input	CLKi (P1$_6$, P3$_6$)	Internal/external clock select bit (bit 3 at address 00A0$_{16}$ and 00A8$_{16}$) = "1" Ports P1$_6$ and P3$_6$ direction register (bit 6 at address 00E3$_{16}$ and 00E7$_{16}$) = "0"	Ports P1$_6$ and P3$_6$ can be used as an I/O port when not performing transfer clock input. In this case, set the internal/external clock select bit to "0".

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

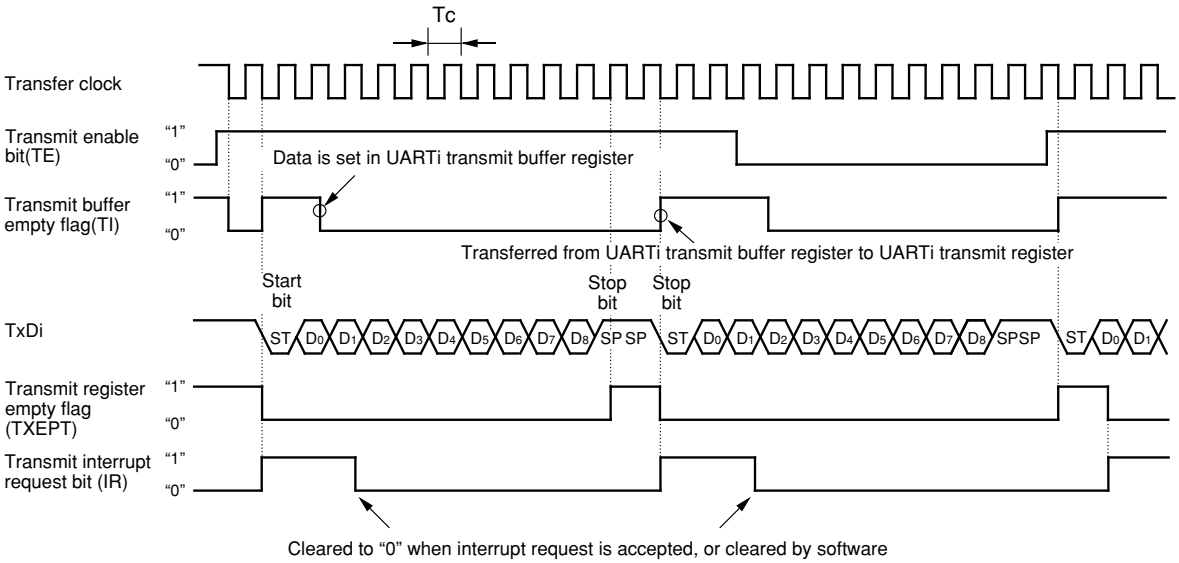


Shown in ( ) are bit symbols.

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 or 16 (n + 1) / fEXT
fi : frequency of BRGi count source (f1, f8, f32, fc)
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.
- 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, fc)
fEXT : frequency of BRGi count source (external clock)
n : value set to BRGi

Figure 1.15.12. Typical transmit timings in UART mode

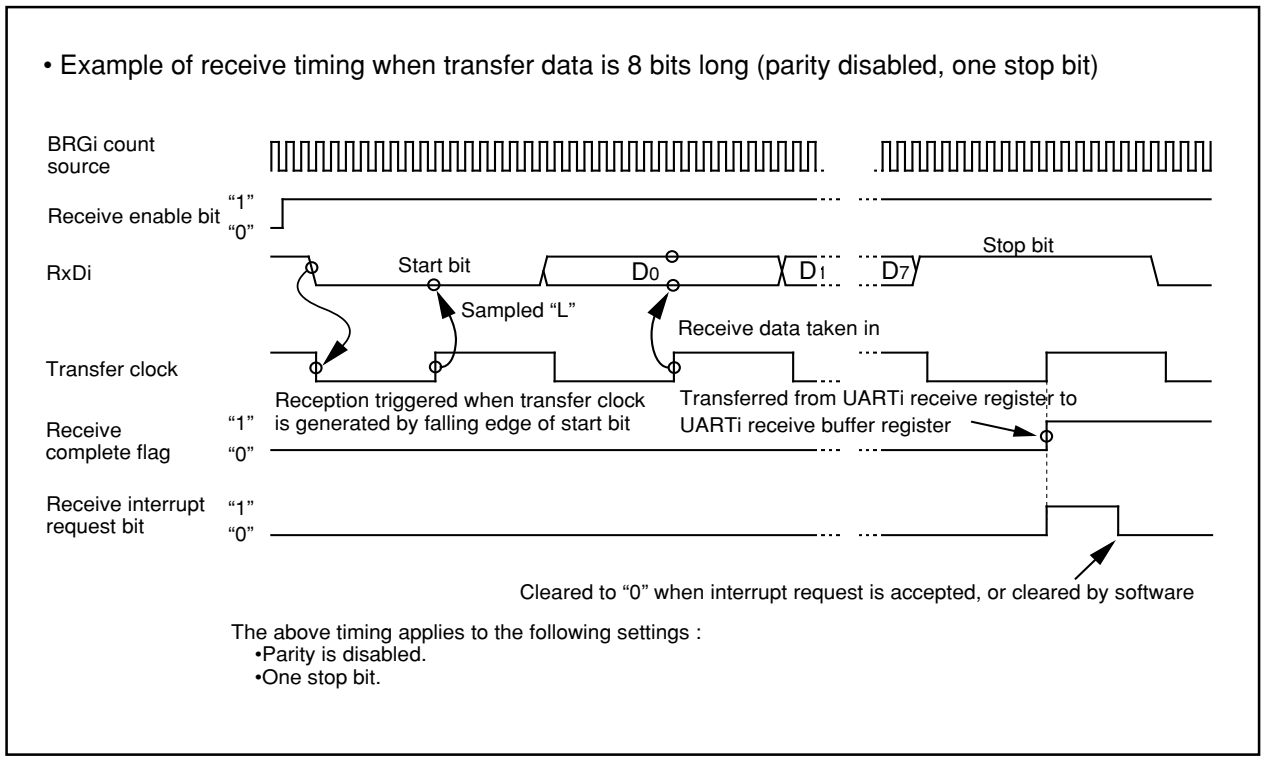


Figure 1.15.13. Typical receive timing in UART mode

(a) RxD_1 input pin selection function (UART1)

This function allows the setting two RxD_1 input pins and choosing one of the two to input serial data by using the RxD_1 input pin select bit (bits 6 at address $00B0_{16}$).

When selecting "1" ($P3_5$) for RxD_1 input pin select bit, $P3_7$ functions as TxD_1 output pin. When selecting "0" ($P3_7$), serial data output cannot be performed. However, $P3_5$ can be used as an input/output port.

A-D Converter

The A-D converter consists of one 10-bit successive approximation A-D converter circuit with a capacitive coupling amplifier. Pins $P0_0$ to $P0_7$, $P1_0$ to $P1_3$, $P4_0$ and $P4_1$ 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 $00D7_{16}$) 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 connecting to V_{REF}.

The result of A-D conversion is stored in the A-D registers. 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 1.16.1 shows the performance of the A-D converter. Figure 1.16.1 shows the block diagram of the A-D converter, and Figures 1.16.2 and 1.16.3 show the A-D converter-related registers.

Table 1.16.1. Performance of A-D converter

Item	Performance
Method of A-D conversion	Successive approximation (capacitive coupling amplifier)
Analog input voltage (Note 1)	0V to 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_{11} input : ±3LSB $ANEX_0$ and $ANEX_1$ 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 and repeat mode (Note 3)
Analog input pins	12 pins (AN_0 to AN_{11}) + 2 pins ($ANEX_0$ to $ANEX_1$)
A-D conversion start condition	• Software trigger A-D conversion starts when the A-D conversion start flag changes to “1”
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 f_{AD} if (X_{IN}) exceeds 10MHz, and make ϕ_{AD} equal to or lower than 10MHz. Also if Vcc is less than 4.2V or an external RC circuit is used for the main clock, divide f_{AD} and make ϕ_{AD} equal to or lower than $f_{AD}/2$.
Without sample and hold function, set the ϕAD frequency to 250kHz min.
With the sample and hold function, set the ϕAD frequency to 1MHz min.

Note 3: In repeat mode, only 8-bit mode can be used.

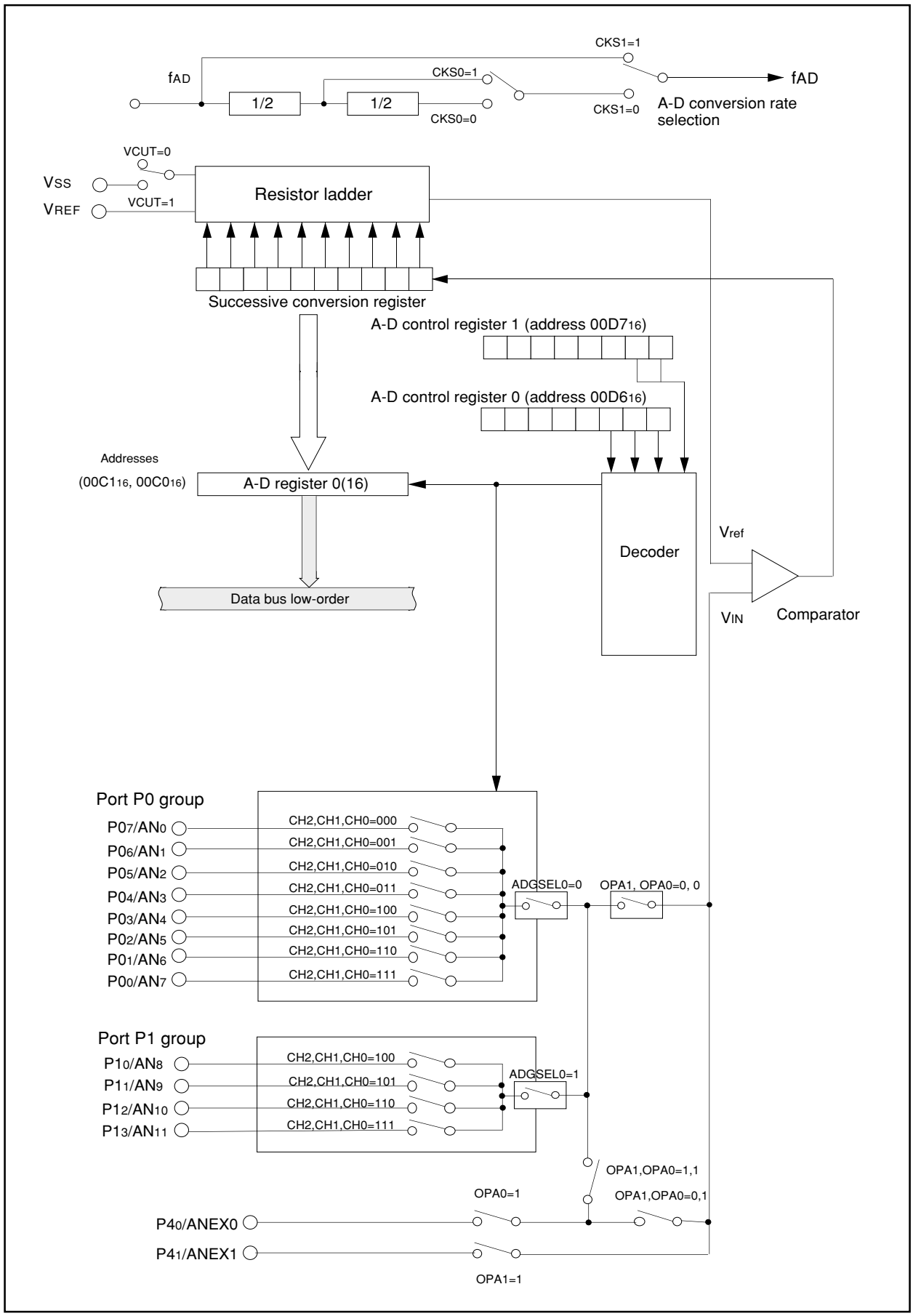


Figure 1.16.1. Block diagram of A-D converter

A-D control register 0 (Note 1)

b7	b6	b5	b4	b3	b2	b1	b0
		0					

Symbol	Address	When reset
ADCON0	$00D6_{16}$	$00000XXX_2$

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, AN_8 is selected 1 0 1 : AN_5, AN_9 is selected 1 1 0 : AN_6, AN_{10} is selected 1 1 1 : AN_7, AN_{11} is selected (Note 2, 3)	○	○
CH1				
CH2				
MD0	A-D operation mode select bit 0	0 : One-shot mode 1 : Repeat mode (Note 2)	○	○
ADGSEL0	A-D input group select bit	0 : Port P0 group is selected 1 : Port P1 group is selected	○	○
Set this bit to “0”.			○	○
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.
Note 3: AN_4 to AN_7 and AN_8 to AN_{11} are selected by the A-D input group select bit.

A-D control register 1 (Note 1)

b7	b6	b5	b4	b3	b2	b1	b0
					0	0	0

Symbol	Address	When reset
ADCON1	$00D7_{16}$	00_{16}

Bit symbol	Bit name	Function	R	W
Set this bit to “0”. When read, their values are “0”.			○	○
BITS	8/10-bit mode select bit (Note 2)	0 : 8-bit mode 1 : 10-bit mode	○	○
CKS1	Frequency select bit 1 (Note 3)	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 : $ANEX_0$ and $ANEX_1$ are not used 0 1 : $ANEX_0$ input is A-D converted 1 0 : $ANEX_1$ input is A-D converted 1 1 : External op-amp connection mode	○	○
OPA1			○	○

Note 1: If the A-D control register is rewritten during A-D conversion, the conversion result is indeterminate.
Note 2: In repeat mode, only 8-bit mode can be used.
Note 3: When $f(X_{IN})$ is over 10 MHz, the ϕ_{AD} frequency must be under 10 MHz by dividing.

Figure 1.16.2. A-D converter-related registers (1)

A-D control register 2 (Note)

b7	b6	b5	b4	b3	b2	b1	b0
				0	0	0	

Symbol	Address	When reset
ADCON2	$00D4_{16}$	$XXXX0000_2$

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		Always 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: If the A-D control register is rewritten during A-D conversion, the conversion result is indeterminate.

A-D register

Symbol	Address	When reset
AD	$00C0_{16}$	Indeterminate
	$00C1_{16}$	Indeterminate

(b15) b7 (b8) b0 b7 b0

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 The value, if read, turns out to be indeterminate.	×	×
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be indeterminate.	—	—

Figure 1.16.3. 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. (See Table 1.16.2.) Figure 1.16.4 shows the A-D control register in one-shot mode.

Table 1.16.2. 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") • 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_{11}, as selected
Reading of result of A-D converter	Read A-D register

A-D control register 0 (Note 1)

b7	b6	b5	b4	b3	b2	b1	b0	Symbol	Address	When reset
		0		0				ADCON0	$00D6_{16}$	$00000XXX_2$

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, AN_8 is selected 1 0 1 : AN_5, AN_9 is selected 1 1 0 : AN_6, AN_{10} is selected 1 1 1 : AN_7, AN_{11} is selected (Note 2, 3)	O	O
CH1			O	O
CH2			O	O
MD0	A-D operation mode select bit 0	0 : One-shot mode (Note 2)	O	O
ADGSEL0	A-D input group select bit	0 : Port P0 group is selected 1 : Port P1 group is selected	O	O
Set this bit to "0".			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 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.
Note 3: AN_4 to AN_7 and AN_8 to AN_{11} are selected by the A-D input group select bit.

A-D control register 1 (Note 1)

b7	b6	b5	b4	b3	b2	b1	b0	Symbol	Address	When reset
		1			0	0	0	ADCON1	$00D7_{16}$	00_{16}

Bit symbol	Bit name	Function	R	W
Set this bit to "0". When read, their values are "0".			O	O
BITS	8/10-bit mode select bit	0 : 8-bit mode 1 : 10-bit mode	O	O
CKS1	Frequency select bit 1 (Note 2)	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	b7 b6 0 0 : $ANEX_0$ and $ANEX_1$ are not used 0 1 : $ANEX_0$ input is A-D converted 1 0 : $ANEX_1$ 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: When f(X_{IN}) is over 10 MHz, the ϕ_{AD} frequency must be under 10 MHz by dividing.

Figure 1.16.4. 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. (See Table 1.16.3.) Figure 1.16.5 shows the A-D control register in repeat mode.

Table 1.16.3. Repeat mode specifications

Item	Specification
Function	The pin selected by the analog input pin select bit is used for repeated 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	One of AN_0 to AN_{11}, as selected (Note)
Reading of result of A-D converter	Read A-D register (at any time)

Note : AN_4 to AN_7 can be used in the same way as for AN_8 to AN_{11}.

A-D control register 0 (Note 1)

b7	b6	b5	b4	b3	b2	b1	b0	Symbol	Address	When reset
		0		1				ADCON0	$00D6_{16}$	$00000XXX_2$

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, AN_8 is selected 1 0 1 : AN_5, AN_9 is selected 1 1 0 : AN_6, AN_{10} is selected 1 1 1 : AN_7, AN_{11} is selected (Note 2, 3)	○	○
CH1			○	○
CH2			○	○
MD0	A-D operation mode select bit 0	1 : Repeat mode (Note 2)	○	○
ADGSEL0	A-D input group select bit	0 : Port P0 group is selected 1 : Port P1 group is selected	○	○
Set this bit to "0".			○	○
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.
Note 3: AN_4 to AN_7 and AN_8 to AN_{11} are selected by the A-D input group select bit.

A-D control register 1 (Note 1)

b7	b6	b5	b4	b3	b2	b1	b0	Symbol	Address	When reset
		1		0	0	0	0	ADCON1	$00D7_{16}$	00_{16}

Bit symbol	Bit name	Function	R	W
Set this bit to "0". When read, their values are "0".			○	○
BITS	8/10-bit mode select bit (Note 2)	0 : 8-bit mode	○	○
CKS1	Frequency select bit 1 (Note 3)	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 : $ANEX_0$ and $ANEX_1$ are not used 0 1 : $ANEX_0$ input is A-D converted 1 0 : $ANEX_1$ input is A-D converted 1 1 : External op-amp connection mode	○	○
OPA1			○	○

Note 1: If the A-D control register is rewritten during A-D conversion, the conversion result is indeterminate.
Note 2: In repeat mode, only 8-bit mode can be used.
Note 3: When $f(X_{IN})$ is over 10 MHz, the ϕ_{AD} frequency must be under 10 MHz by dividing.

Figure 1.16.5. A-D conversion register in repeat mode

• Sample and hold

Sample and hold is selected by setting bit 0 of the A-D control register 2 (address 00D4$_{16}$) to "1". When sample and hold is selected, the rate of conversion of each pin increases. As a result, a 28 øAD cycle is achieved with 8-bit resolution and 33 øAD 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.

• Extended analog input pins

In one-shot mode and repeat mode, the input via the extended analog input pins ANEX$_0$ and ANEX$_1$ can also be converted from analog to digital.

When bit 6 of the A-D control register 1 (address 00D7$_{16}$) is "1" and bit 7 is "0", input via ANEX$_0$ is converted from analog to digital.

When bit 6 of the A-D control register 1 (address 00D7$_{16}$) is "0" and bit 7 is "1", input via ANEX$_1$ is converted from analog to digital.

• External operation amp connection mode

In this mode, multiple external analog inputs via the extended analog input pins, ANEX$_0$ and ANEX$_1$, 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 00D7$_{16}$) is "1" and bit 7 is "1", input via AN$_0$ to AN$_{11}$ is output from ANEX$_0$. The input from ANEX$_1$ is converted from analog to digital and the result stored in the A-D register. The speed of A-D conversion depends on the response of the external operation amp. Do not connect the ANEX$_0$ and ANEX$_1$ pins directly. Figure 1.16.6 is an example of how to connect the pins in external operation amp mode.

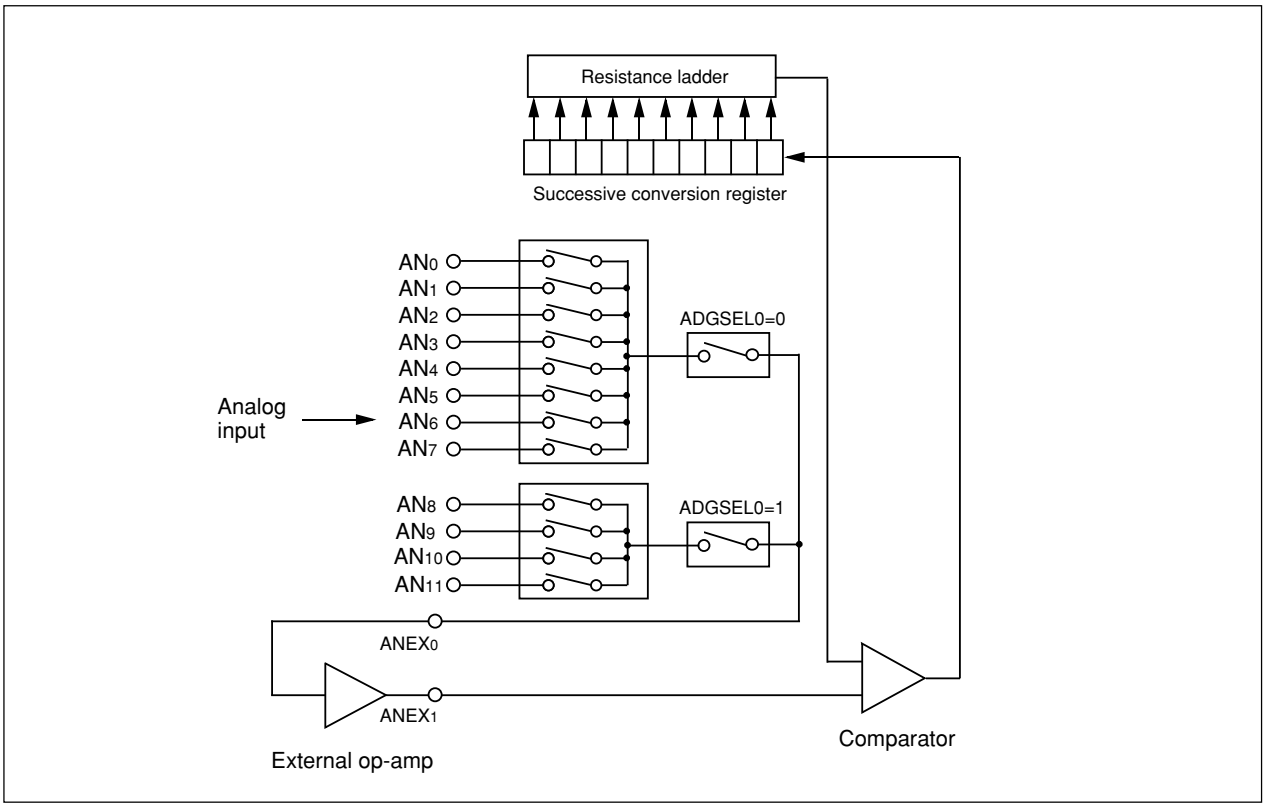


Figure 1.16.6. Example of external op-amp connection mode

D-A Converter

This is an 8-bit, R-2R type D-A converter. The microcomputer contains one independent D-A converter of this type. D-A conversion is performed when a value is written to the corresponding D-A register. Bit 0 (D-A output enable bit) 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 D-A output is set for enabled, the corresponding port is inhibited to be pulled up.

Output analog voltage (V) is determined by a set value (n : decimal) in the D-A register.

$V = V_{REF} \times n / 256$ (n = 0 to 255)

V_{REF}: reference voltage

Table 1.17.1 lists the performance of the D-A converter. Figure 1.17.1 shows the block diagram of the D-A converter, Figure 1.17.2 shows the D-A control register and Figure 1.17.3 shows D-A converter equivalent circuit.

Table 1.17.1. Performance of D-A converter

Item	Performance
Conversion method	R-2R method
Resolution	8 bits
Analog output pin	1 channel

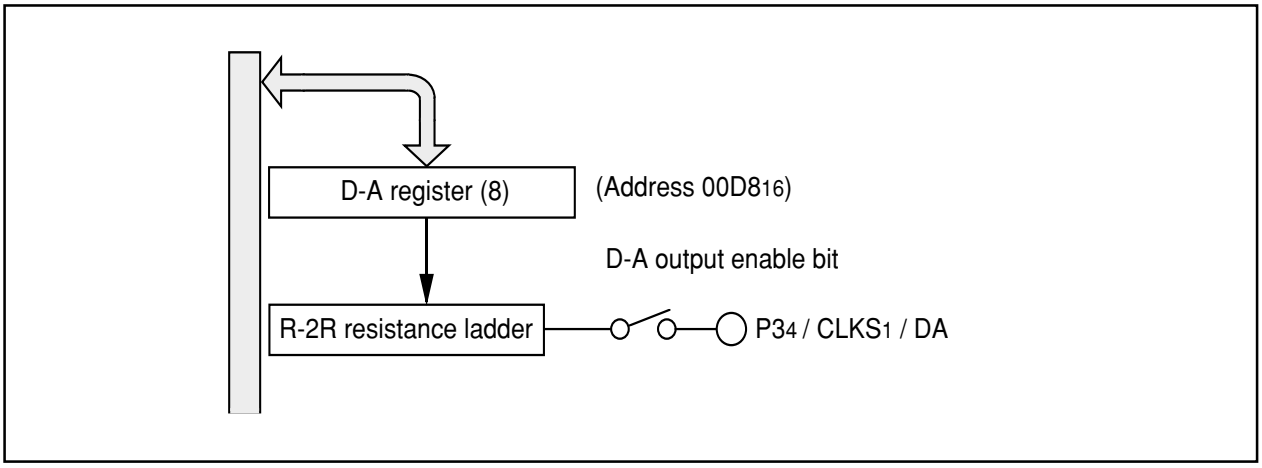


Figure 1.17.1. Block diagram of D-A converter

D-A control register

b7	b6	b5	b4	b3	b2	b1	b0
					0		

Symbol	Address	When reset
DACON	$00DC_{16}$	00_{16}

Bit symbol	Bit name	Function	R	W
DAE	D-A output enable bit	0 : Output disabled 1 : Output enabled	○	○
Nothing is assigned. When write, set "0". When read, the value of this bit is "0".			—	—
Reserved bit		Must always set to "0"	—	—
Nothing is assigned. When write, set "0". When read, the value of these bits is "0".			—	—

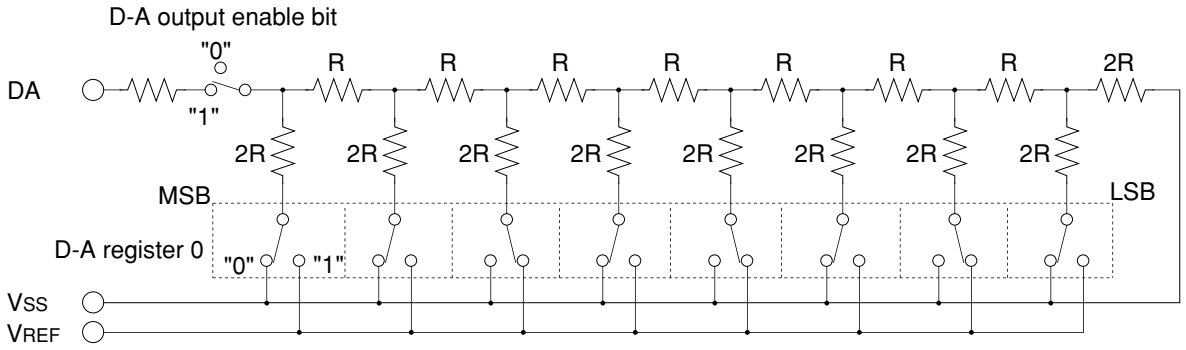
D-A register

b7	b0

Symbol	Address	When reset
DA	$00D8_{16}$	Indeterminate

Function	R	W
Output value of D-A conversion	○	○

Figure 1.17.2. D-A control register



Note 1: In the above figure, the D-A register value is "$2A_{16}$".
Note 2: To save power when not using the D-A converter, set the D-A output enable bit to "0" and the D-A register to "00_{16}", and prevent current flowing to the R-2R resistance.

Figure 1.17.3. D-A converter equivalent circuit

Programmable I/O Ports

There are 34 programmable I/O ports: P0 to P4 (when M30102). 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. The port P1 allows the drive capacity of its N-channel output transistor to be set as necessary. The port P1 can be used as LED drive port if the drive capacity is set to “HIGH”.

Figures 1.18.1 to 1.18.4 show the programmable I/O ports.

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 1.18.5 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.

(2) Port registers

Figure 1.18.6 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.

(3) Pull-up control registers

Figure 1.18.7 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.

(4) Port P1 drive capacity control register

Figure 1.18.7 shows a structure of the port P1 drive capacity control register.

This register is used to control the drive capacity of the port P1's N-channel output transistor. Each bit in this register corresponds one for one to the port pins.

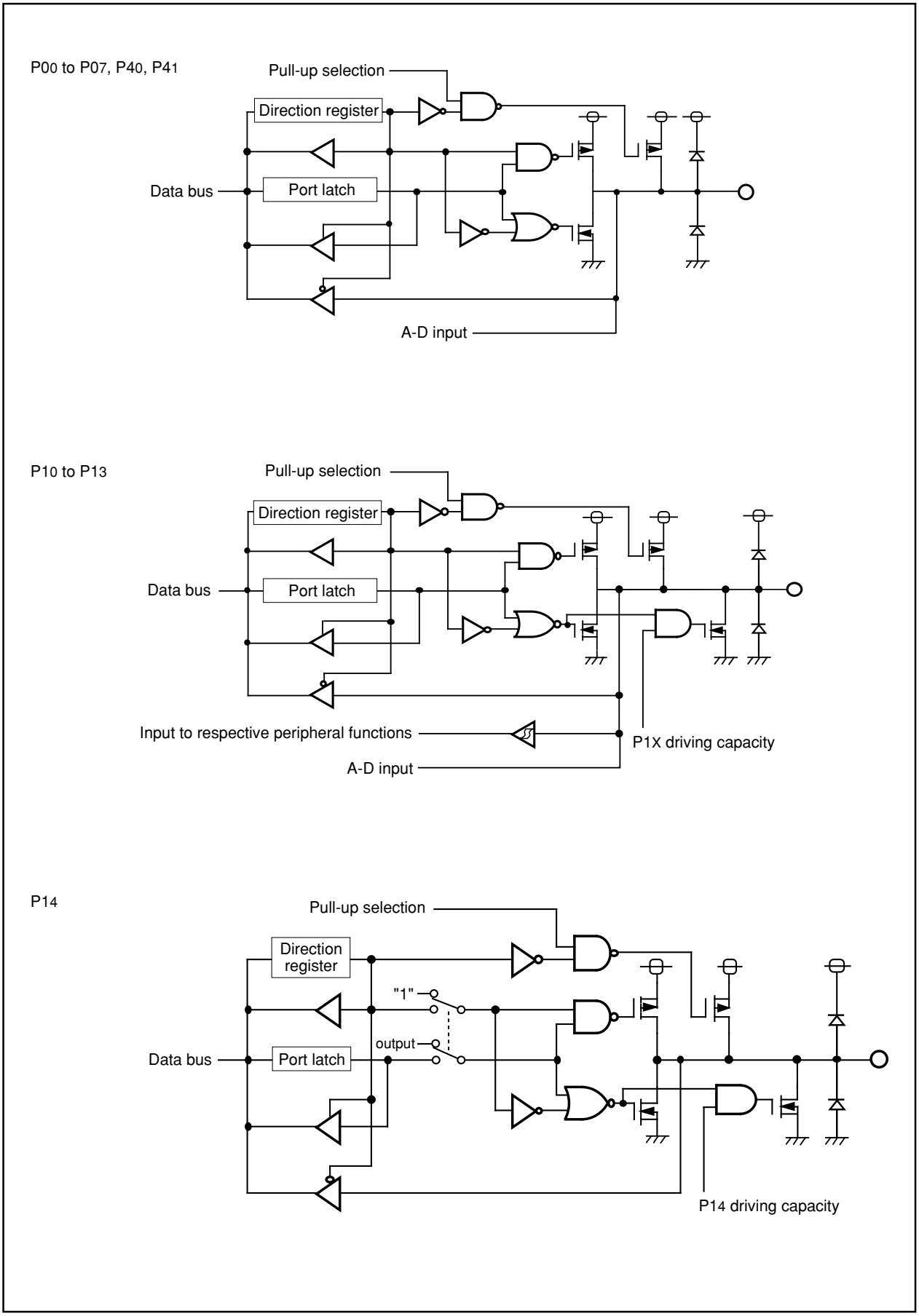
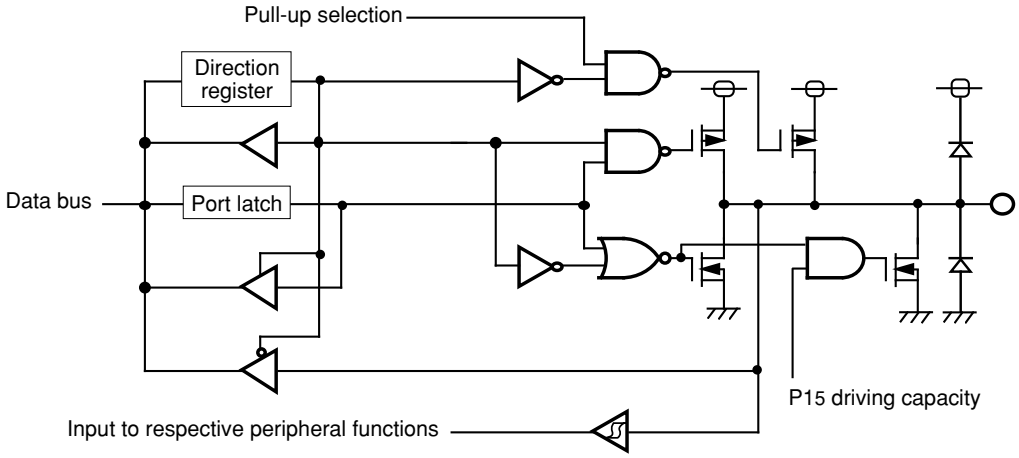
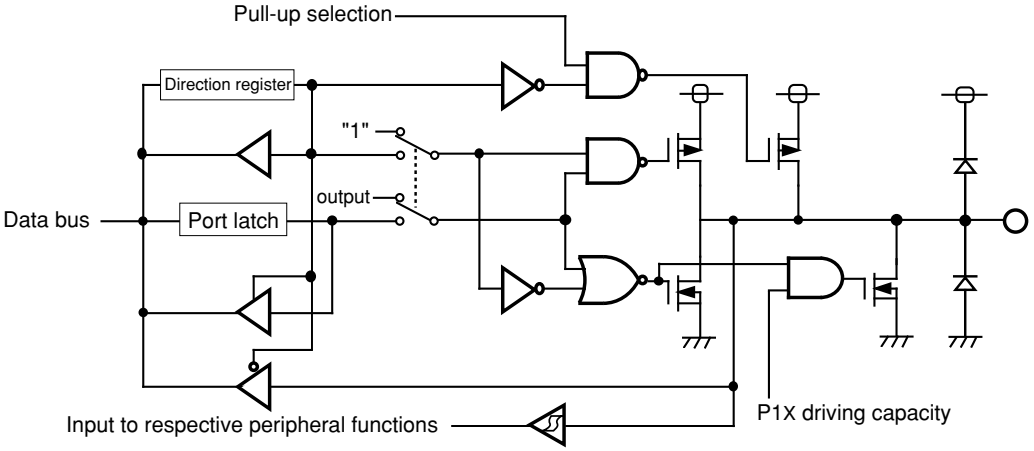


Figure 1.18.1. Programmable I/O ports (1)

P15



P16, P17



P20, P21

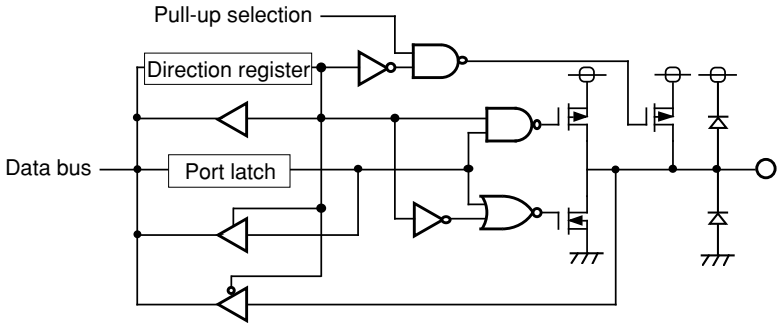


Figure 1.18.2. Programmable I/O ports (2)

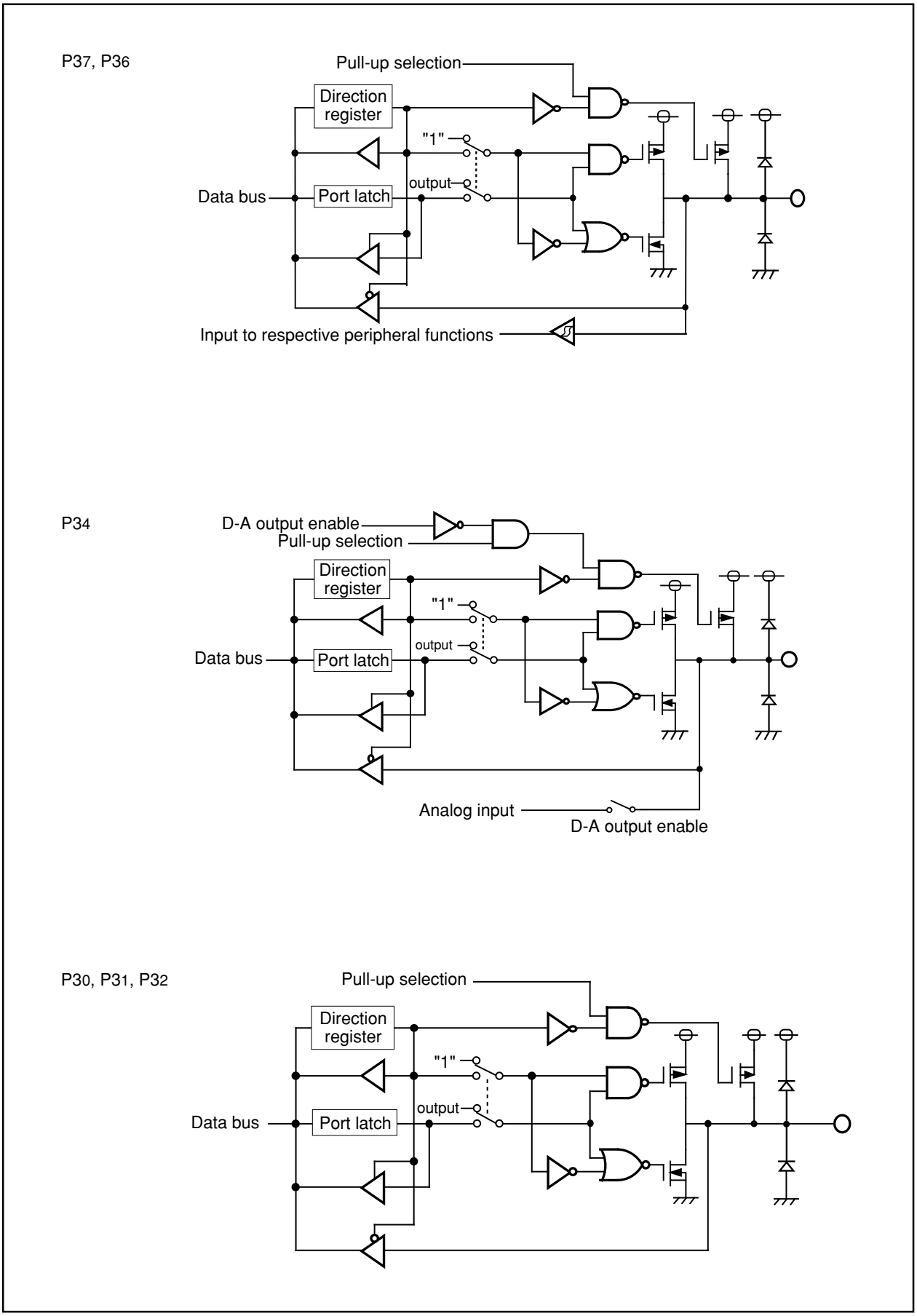


Figure 1.18.3. Programmable I/O ports (3)

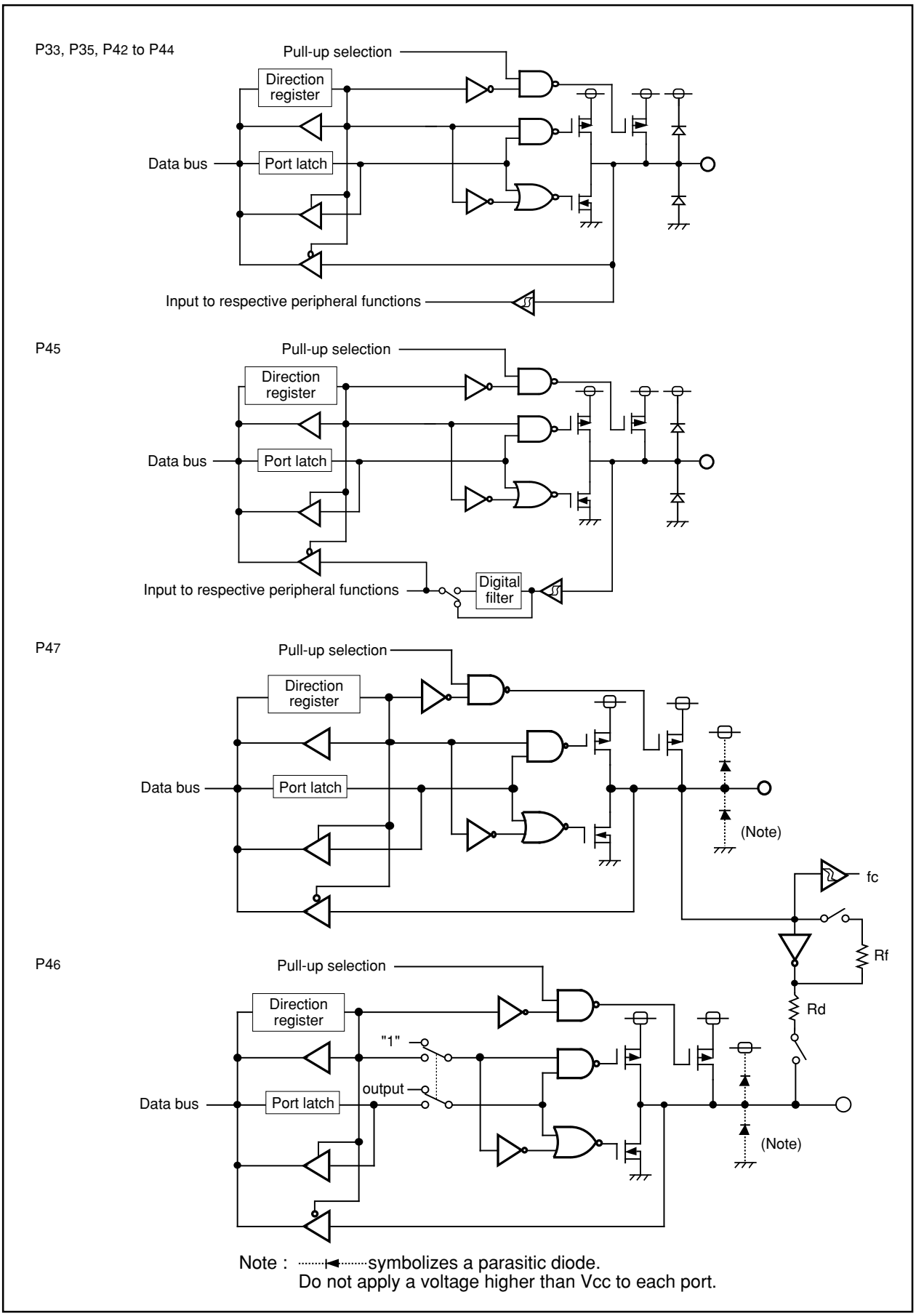
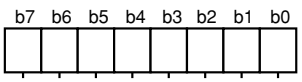


Figure 1.18.4. Programmable I/O ports (4)

Port Pi direction register (Notes 1, 2)



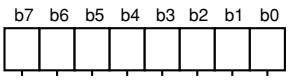
Symbol	Address	When reset
PDi (i = 0 to 4)	00E2$_{16}$, 00E3$_{16}$, 00E6$_{16}$, 00E7$_{16}$, 00EA$_{16}$	00$_{16}$ 00$_{16}$

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)	○	○
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 P0 direction register.
Note 2: Nothing is assigned in bit 2 to bit 7 of port P2 direction register. When write, set “0”. When read, their contents are “0”.

Figure 1.18.5. Direction register

Port Pi register (Note)



Symbol	Address	When reset
Pi (i = 0 to 4)	00E0$_{16}$, 00E1$_{16}$, 00E4$_{16}$, 00E5$_{16}$, 00E8$_{16}$	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	○	○
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 : Nothing is assigned in bit 2 to bit 7 of port P2 register. When write, set “0”. When read, their contents are “0”.

Figure 1.18.6. Port register

Pull-up control register 0

b7	b6	b5	b4	b3	b2	b1	b0

Symbol	Address	When reset
PUR0	$00FC_{16}$	$00X00000_2$

Bit symbol	Bit name	Function	R	W
PU00	$P0_0$ to $P0_3$ pull-up	The corresponding port is pulled high with a pull-up resistor 0 : Not pulled high 1 : Pulled high	○	○
PU01	$P0_4$ to $P0_7$ pull-up		○	○
PU02	$P1_0$ to $P1_3$ pull-up		○	○
PU03	$P1_4$ to $P1_7$ pull-up		○	○
PU04	$P2_0$, $P2_1$ pull-up		○	○
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be indeterminate.			—	—
PU06	$P3_0$ to $P3_3$ pull-up	The corresponding port is pulled high with a pull-up resistor 0 : Not pulled high 1 : Pulled high	○	○
PU07	$P3_4$ to $P3_7$ pull-up		○	○

Pull-up control register 1

b7	b6	b5	b4	b3	b2	b1	b0

Symbol	Address	When reset
PUR1	$00FD_{16}$	$XXXXXX00_2$

Bit symbol	Bit name	Function	R	W
PU10	$P4_0$ to $P4_3$ pull-up	The corresponding port is pulled high with a pull-up resistor 0 : Not pulled high 1 : Pulled high	○	○
PU11	$P4_4$ to $P4_7$ pull-up		○	○
Nothing is assigned. In an attempt to write to these bits, write “0”. The value, if read, turns out to be indeterminate.			—	—

Port P1 drive capacity control register

b7	b6	b5	b4	b3	b2	b1	b0

Symbol	Address	When reset
DRR	$00FE_{16}$	00_{16}

Bit symbol	Bit name	Function	R	W
DRR0	Port $P1_0$ drive capacuty	Set P1 N-channel output transistor drive capacity 0 : LOW 1 : HIGH	○	○
DRR1	Port $P1_1$ drive capacuty		○	○
DRR2	Port $P1_2$ drive capacuty		○	○
DRR3	Port $P1_3$ drive capacuty		○	○
DRR4	Port $P1_4$ drive capacuty		○	○
DRR5	Port $P1_5$ drive capacuty		○	○
DRR6	Port $P1_6$ drive capacuty		○	○
DRR7	Port $P1_7$ drive capacuty		○	○

Figure 1.18.7. Pull-up control register

Example connection of unused pins

Table 1.18.1. Example connection of unused pins

Pin name	Connection
Ports P0 to P4	After setting for input mode, connect every pin to VSS (pull-down) via a resistor; or after setting for output mode, leave these pins open. (Note 1)
XOUT (Note 2)	Open
VREF	Connect to VSS
XIN (Note 3)	Connect to VCC (pull-up) via a resistor

Note 1: Connect unused pins as described above. If connected otherwise, power supply current may increase due to flow-through current on Schmitt circuit in the port.
Note 2: With external clock input to XIN pin, or the main clock oscillation circuit isn't used.
Note 3: When the main clock oscillation circuit isn't used, connect XIN pin to VCC (pull-up), leave XOUT pin open.or set main clock stop bit (bit 5 at address 0006_{16}) to "1"(STOP).

Precautionary Notes in Using the Device

Serial I/O

(1) When reading data from the UARTi receive buffer in the clock asynchronous serial I/O mode, data should be read high-byte first then low-byte using byte-size. If data is read as low-byte then high-byte or in word-size, the framing error and parity error flags are cleared.

A code example is shown below.

```
MOV.B    00A7H. R0H    ; Read the high-byte of UART0 receive buffer register
MOV.B    00A6H. R0L    ; Read the low-byte of UART0 receive buffer register
```

(2) When writing data to the UARTi transmit buffer register in the clock asynchronous serial I/O mode with 9-bit transfer data length, data should be written high-byte first then low-byte using byte-size.

A code example is shown below.

```
MOV.B    #XXH, 00A3H    ; Write the high-byte of UART0 transmit buffer register
MOV.B    #XXH, 00A2H    ; Write the low-byte of UART0 transmit buffer register
```

A-D Converter

(1) Only write to each bit (except bit 6) of the AD Control Register 0, or each bit of the AD Control Register 1, or bit 0 of the AD Control Register 2 when AD conversion is stopped (before a trigger occurs).
When the Vref Connection Bit is changed from “0” to “1”, wait 1 μs or longer before starting AD conversion.

(2) When changing AD operation mode, select an analog pin again.

(3) One Shot Mode
Read the AD register only after confirming AD conversion is completed, which can be determined by using the AD conversion interrupt.

(4) Repeat Mode
Use the undivided main clock as the internal CPU clock when using this mode. The main clock can be divided by an internal divider circuit but make sure that you use main clock when using this mode.

(5) If A-D conversion is forcibly terminated while in progress by setting the ADST bit of ADCON0 register to 0 (A-D conversion halted), the conversion result of the A-D converter is indeterminate. If the ADST bit is cleared to 0 in a program, ignore the value of AD register.

Stop and Wait Mode

(1) You must put at least four NOPs after a stop (All-Clock Stop Bit to "1") or a wait instruction. When switching to a stop or wait mode, 4 instructions are prefetched after the stop or wait instruction. And so, ensure that at least four NOPs follow the stop or wait instruction.

Stop Mode

(1) After returning from stop mode, an unexpected operation may occur (for example, undefined instruction interrupt, BRK instruction interrupt, etc.).
Execute a JMP.B instruction after an instruction to write data to the all clock stop control bit. A program example is described as follows:

```
        MOV.B:S    #21H, CM1  ; writing to the all clock stop control bit  to “1”(stop mode)
        JMP.B      L1
    L1 :
        NOP
        NOP
        NOP
        NOP
```

Interrupts

(1) Reading Address 0 by Firmware

- Please do not read address 0 by firmware. In the CPU's interrupt processing sequence, when a maskable interrupt occurs, the interrupt information (interrupt no. and interrupt request level) are read from address 0. This read in turn, clears the interrupt request bit to "0" even pending with higher request level. Reading address 0 by firmware may cause interrupt cancellation or unexpected interrupts so please do not read address 0 by firmware.

(2) Stack Pointer

- Set the value of the stack pointer before accepting interrupts. Immediately after a reset, the value of the stack pointer is 0000_{16}. Accepting an interrupt before setting a value of the stack pointer may produce unpredictable results (runaway program, etc.) Make sure that you set the value of the stack pointer before accepting interrupts.

(3) External interrupts

- Clear the interrupt request bit to "0" when the INT0 - INT3 polarity is changed. The reason being is that an interrupt request may be generated when the polarity is changed.

(4) Rewriting the Interrupt Control Register

- When rewriting the Interrupt Control Register, do it at a point where it does not generate an interrupt request for that register. If there is a possibility that an interrupt may occur, disable the interrupt before rewriting. Examples are shown below.

Example 1:

```
    INT_SWITCH1:
        FCLR    I               ; Disable interrupts.
        AND.B   #00h, 0055h     ; Clear T1IC int. priority level and int. request bit.
        NOP                     ;
        NOP
        FSET    I               ; Enable interrupts.
```

Example 2:

```
INT_SWITCH2:
    FCLR    I           ; Disable interrupts.
    AND.B   #00h, 0055h ; Clear T1IC 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 T1IC int. priority level and int. request bit.
    POPC    FLG         ; Enable interrupts.
```

Note: The reason why two NOP instructions or dummy read were inserted before the FSET I for ex. 1 & 2 is to prevent interrupt enable flag from being set, due to the effects of instruction queue, before the rewritten value of the interrupt control register takes effect.

• When an instruction to rewrite the interrupt control register is executed while the interrupt is disabled, depending on the instruction used for rewriting, there are times the interrupt request bit is not set even if an interrupt request for that register has been generated. If this creates a problem, please use any of the instructions below to rewrite the register.
Instructions : AND, OR, BCLR, BSET

• Changing the interrupt request bit
When attempting to clear the interrupt request bit of an interrupt control register, the interrupt request bit is not cleared sometimes. This will depend on the instruction. If this creates problems, use the below instructions to change the register.
Instructions : MOV

Noise

(1) Bypass Capacitor between VCC and VSS Pins

• Insert a bypass capacitor (at least 0.1 μF) between VCC and VSS pins as noise and latch-up countermeasures. In addition, make sure that connecting lines are the shortest and widest possible.

(2) Port Control Registers Data Read Error

• During severe noise testing, mainly power supply system noise, and introduction of external noise, the data of port related registers may changed. As a firmware countermeasure, it is recommended to periodically re-set the port registers, port direction registers and pull-up control registers. However, you should fully examine before introducing the re-set routine as conflicts may be created between this re-set routine and interrupt routines (i. e. ports are switched during interrupts).

(3) CNVss pin wiring

• In order to improve the pin tolerance to noise, insert a pull down resistance (about 5 kΩ) between CNVss and Vss, and placed as close as possible to the CNVss pin.

Timer 1

(1) Even if the prescaler 1 and Timer 1 are read out simultaneously in word-size, these registers are read byte-by-byte in the microcomputer. Consequently, the timer value may be updated during the period these two registers are being read.

Timers X, Y and Z

(1) These timers stop counting after reset. Therefore, set values to Timer (X, Y, Z) and prescaler (X, Y, Z) before starting counting.

(2) Even if prescaler (X, Y, Z) and Timer (X, Y, Z) are read out simultaneously in word-size, these registers are read byte-by-byte in the microcomputer. Consequently, the timer value may be updated during the period these two registers are being read.

Timer X

(1) Using in the timer X pulse period measurement mode, the effectaul edge rception flag and the timer X under flow flag are setted to "0" by writing a "0" in a program. Writing a "1" has no effect. Write "1" in the other flag by using the MOV instruction when you make the flag of either one side "0" by program. (The clearance of the flag which isn't intend can be prevnted.)

Example:

```
MOV.B    #10XXXXXXB,008Bh
```

(2) When changing to the timer X pulse period measurement mode from other mode, the contents of the effectaul edge rception flag and the timer X under flow flag are indetermind. Write "0" in the effectaul edge rception flag and the timer X under flow flag before starting the timer.

Timer Y

(1) When count is stopped by writing "0" to the timer Y count start flag, the timer reloads the value of reload register and stops. Therefore, the timer count value should be read out before the timer stops.

(2) When count is stopped by writing "0" to the timer Y count start flag, the timer Y interrupt request flag becomes "1" and an interrupt may occur. Thus, disable interrupts before the timer stops. Furthermore, set the Timer Y interrupt request flag to "0" before starting the timer again.

Timer Z

(1) When count is stopped by writing "0" to the timer Z count start flag, the timer reloads the value of reload register and stops. Therefore, the timer count value should be read out before the timer stops.

(2) When count is stopped by writing "0" to the timer Z count start flag (all modes) or by writing "0" to the one-shot start bit (programmable one-shot generation mode/programmable wait one-shot generation mode), the timer Z interrupt request flag becomes "1" and an interrupt may occur. Thus, disable interrupts before the timer stops. Furthermore, set the Timer Z interrupt request flag to "0" before starting the timer again.

Timer C

(1) When reading out the timer C or timer measurement register, use a word-size instruction. Even if the Timer C is read out in word-size, the timer value is not updated during the period the high-byte and low-byte are being read.

Example:

```
MOV.W    0091H,R0    ; Read out timer C
```

Electrical characteristics

Table 1.19.1. Absolute maximum ratings

Symbol	Parameter		Condition	Rated value	Unit
V_{CC}	Supply voltage			- 0.3 to 6.5	V
V_I	Input voltage	$\overline{RESET}$, V_{REF}, X_{IN} $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$, $P2_1$, $P3_0$ to $P3_7$, $P4_0$ to $P4_7$, CNVss		- 0.3 to Vcc + 0.3	V
V_O	Output voltage	$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$, $P2_1$,$P3_0$ to $P3_7$, $P4_0$ to $P4_7$,X_{OUT}		- 0.3 to Vcc + 0.3	V
		IV_{CC}		- 0.3 to 3.6V	V
P_d	Power dissipation		Ta = 25 °C	300	mW
T_{opr}	Operating ambient temperature			- 20 to 85 (Note 1)	°C
T_{stg}	Storage temperature			- 40 to 150 (Note 2)	°C

Note 1: Extended operating temperature version: -40 to 85 °C. When flash memory version is program/erase mode: 25±5 °C.
Specify a product of -40 to 85°C to use it.

Note 2: Extended operating temperature version: -65 to 150 °C.

Note 3: For M30100 (32-pin version), $P2_0$, $P2_1$, $P3_4$ to $P3_6$, $P4_0$ to $P4_4$, $P4_6$ and $P4_7$ are not accessed to external pins.

Table 1.19.2. Recommended operating conditions (Note 1)

Symbol	Parameter			Standard Min	Standard Typ.	Standard Max.	Unit
Vcc	Supply voltage			2.7 (Note 1)	5.0	5.5	V
Vss	Supply voltage				0		V
V_{IH}	HIGH input voltage	$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$, $P2_1$, $P3_0$ to $P3_7$, $P4_0$ to $P4_7$, X_{IN}, RESET, CNV_{SS},		0.8Vcc		Vcc	V
V_{IL}	LOW input voltage	$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$, $P2_1$, $P3_0$ to $P3_7$, $P4_0$ to $P4_7$, X_{IN}, RESET, CNV_{SS},		0		0.2Vcc	V
$I_{OH\ (peak)}$	HIGH peak output current	$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$, $P2_1$, $P3_0$ to $P3_7$, $P4_0$ to $P4_7$,				- 10.0	mA
$I_{OH\ (avg)}$	HIGH average output current	$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$, $P2_1$, $P3_0$ to $P3_7$, $P4_0$ to $P4_7$,				- 5.0	mA
$I_{OL\ (peak)}$	LOW peak output current	$P0_0$ to $P0_7$, $P2_0$, $P2_1$, $P3_0$ to $P3_7$, $P4_0$ to $P4_7$,				10.0	mA
$I_{OL\ (peak)}$		$P1_0$ to $P1_7$	HIGH POWER			20.0	mA
			LOW POWER			10.0	
$I_{OL\ (avg)}$	LOW average output current	$P0_0$ to $P0_7$, $P2_0$, $P2_1$, $P3_0$ to $P3_7$, $P4_0$ to $P4_7$,				5.0	mA
$I_{OL\ (avg)}$		$P1_0$ to $P1_7$	HIGH POWER			10.0	mA
			LOW POWER			5.0	
f (X_{IN})	Main clock input oscillation frequency (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
f (X_{CIN})	Subclock oscillation frequency				32.768	50	kHz

Note 1: For applications for automobile use, this value is 4.2V.

Note 2: Unless otherwise noted: Vcc = 2.7V to 5.5V, Ta = – 20 to 85°C

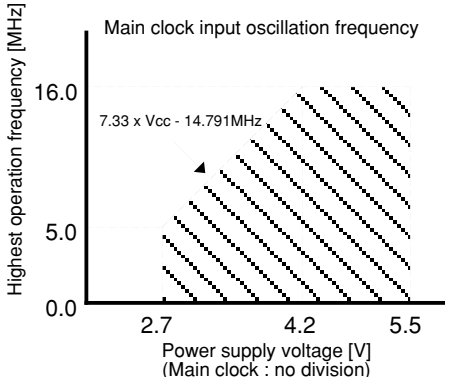
Note 3: The average output current is an average value measured over 100ms.

Note 4: Keep output current as follows:

The sum of port $P0_0$ to $P0_3$, $P1_3$ to $P1_7$, $P2_0$, $P3_4$ to $P3_7$, $P4_6$ to $P4_7$ I_{OL} (peak) is under 60 mA. The sum of port $P0_0$ to $P0_3$, $P1_3$ to $P1_7$, $P2_0$, $P3_4$ to $P3_7$, $P4_6$ to $P4_7$ I_{OH} (peak) is under 60 mA. The sum of port $P0_4$ to $P0_7$, $P1_0$ to $P1_2$, $P2_1$, $P3_0$ to $P3_3$, $P4_0$ to $P4_5$ I_{OL} (peak) is under 60 mA. The sum of port $P0_4$ to $P0_7$, $P1_0$ to $P1_2$, $P2_1$, $P3_0$ to $P3_3$, $P4_0$ to $P4_5$ I_{OH} (peak) is under 60 mA.

Note 5: Relationship between main clock oscillation frequency and supply voltage is shown as below.

Note 6: For M30100 (32-pin version), $P2_0$, $P2_1$, $P3_4$ to $P3_6$, $P4_0$ to $P4_4$, $P4_6$ and $P4_7$ are not accessed to external pins.



Electrical characteristics (Vcc = 5V)

Vcc = 5V

Table 1.19.3. (1) Electrical characteristics
(Unless otherwise noted: Vcc = 5V, Vss = 0V at Ta = 25°C, f(XIN) = 16MHz)

Symbol	Parameter		Measuring condition		Standard Min.	Standard Typ.	Standard Max.	Unit
V_{OH}	HIGH output voltage	$P0_0$ to $P0_7$,$P1_0$ to $P1_7$,$P2_0$,$P2_1$, $P3_0$ to $P3_7$,$P4_0$ to $P4_7$	I_{OH} = - 5 mA		3.0			V
			I_{OH} = - 200 µA		4.7			V
V_{OH}	HIGH output voltage	X_{OUT} HIGH POWER	I_{OH} = - 1 mA		3.0			V
		X_{OUT} LOW POWER	I_{OH} = - 0.5 mA		3.0			V
V_{OH}	HIGH output voltage	X_{COUT} (Note) HIGH POWER	No load	Flash memory		3.3		V
			No load	Mask ROM		3.0		V
		X_{COUT} (Note) LOW POWER	No load	Flash memory		3.3		V
			No load	Mask ROM		1.6		V
V_{OL}	LOW output voltage	$P0_0$ to $P0_7$,$P2_0$,$P2_1$, $P3_0$ to $P3_7$,$P4_0$ to $P4_7$	I_{OL} = 5 mA				2.0	V
			I_{OL} = 200 µA				0.45	V
V_{OL}	LOW output voltage	$P1_0$ to $P1_7$ HIGH POWER	I_{OH} = 10 mA				2.0	V
		$P1_0$ to $P1_7$ LOW POWER	I_{OH} = 5 mA				2.0	V
V_{OL}	LOW output voltage	X_{OUT} HIGH POWER	I_{OH} = 1 mA				2.0	V
		X_{OUT} LOW POWER	I_{OH} = 0.5 mA				2.0	V
V_{OL}	LOW output voltage	X_{COUT} HIGH POWER	No load			0		V
		X_{COUT} LOW POWER	No load			0		V
V_{T+} -V_{T-}	Hysteresis	CNTR0,TCIN, $\overline{INT_0}$ to $\overline{INT_3}$,CLK0,CLK1 RxD0, RxD1,KI0 to KI3,P45			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 $P1_7$,$P2_0$,$P2_1$, $P3_0$ to $P3_7$,$P4_0$ to $P4_7$, X_{IN} $\overline{RESET}$, CNVss	V_I = 5V				5.0	µA
I_{IL}	LOW input current	$P0_0$ to $P0_7$,$P1_0$ to $P1_7$,$P2_0$,$P2_1$, $P3_0$ to $P3_7$,$P4_0$ to $P4_7$, X_{IN} $\overline{RESET}$, CNVss	V_I = 0V				-5.0	µA
R_{PULLUP}	Pull-up resistor	$P0_0$ to $P0_7$,$P1_0$ to $P1_7$,$P2_0$,$P2_1$, $P3_0$ to $P3_7$,$P4_0$ to $P4_7$	V_I = 0V		30.0	50.0	167.0	kΩ
R_{fXIN}	Feedback resistor	X_{IN}				1.0		MΩ
R_{fXCIN}	Feedback resistor	X_{CIN}				6.0		MΩ
V_{RAM}	RAM retention voltage		When clock is stopped		2.0			V
R_{OSC}	Oscillation frequency of Ring oscillator		Mask ROM		300	600	1200	kHz
			Flash memory		300	600	1200	kHz

Note: The V_{OH} standard values of X_{COUT} differ between flash memory version and mask ROM version.
Therefore, please note that the oscillation constants of sub clock may differ between these versions.

VCC = 5V

Table 1.19.3. (2) Electrical characteristics
(Unless otherwise noted: VCC = 5V, VSS = 0V at Ta = 25°C, f(XIN) = 16MHz))

Symbol	Parameter	Measuring condition			Standard			Unit
					Min.	Typ.	Max.	
Icc	Power supply current	I/O pin has no load	Mask ROM	f(XIN)=16MHz Square wave, no division		20.0	36.0	mA
			Flash memory	f(XIN)=16MHz Square wave, no division		18.0	36.0	mA
			Mask ROM	Ring oscillator mode No division		800		µA
			Flash memory	Ring oscillator mode No division		1300		µA
			Mask ROM	Ring oscillator mode When a WAIT instruction is executed		100		µA
			Flash memory	Ring oscillator mode When a WAIT instruction is executed		400		µA
			Mask ROM	f(XCIN)=32kHz Square wave		50		µA
			Flash memory	f(XCIN)=32kHz Square wave		700		µA
			Mask ROM	f(XCIN)=32kHz When a WAIT instruction is executed		6		µA
			Flash memory	f(XCIN)=32kHz When a WAIT instruction is executed		350		µA
			Mask ROM	Ta=25°C when clock is stopped			2	µA
				Ta=85°C when clock is stopped			20	
			Flash memory	Ta=25°C when clock is stopped		300	600	µA
				Ta=85°C when clock is stopped		300	600	

V_{CC} = 5V

Table 1.19.4. A-D conversion characteristics (Note 1)

Symbol	Parameter		Measuring condition		Standard			Unit
					Min.	Typ.	Max.	
–	Resolution		V_{REF} =V_{CC}				10	Bits
–	Absolute accuracy	Sample & hold function not available	V_{REF} =V_{CC} = 5V				±3	LSB
		Sample & hold function available(10bit)	V_{REF} =V_{CC}= 5V	AN_0 to AN_{11} input			±3	LSB
				$ANEX_0$, $ANEX_1$ input, external op-amp connected mode			±7	LSB
		Sample & hold function available(8bit)	V_{REF} = V_{CC} = 5V				±2	LSB
R_{LADDER}	Ladder resistance		V_{REF} =V_{CC}		10		40	kΩ
t_{CONV}	Conversion time(10bit)		f(X_{IN})=10MHz, $ø_{AD}$=f_{AD}=10MHz		3.3			µs
t_{CONV}	Conversion time(8bit)		f(X_{IN})=10MHz, $ø_{AD}$=f_{AD}=10MHz		2.8			µs
t_{SAMP}	Sampling time		f(X_{IN})=10MHz, $ø_{AD}$=f_{AD}=10MHz		0.3			µs
V_{REF}	Reference voltage		f(X_{IN})=10MHz, $ø_{AD}$=f_{AD}=10MHz		2		V_{CC}	V
V_{IA}	Analog input voltage		f(X_{IN})=10MHz, $ø_{AD}$=f_{AD}=10MHz		0		V_{REF}	V

Note 1: Unless otherwise noted: V_{CC} = V_{REF} =5V, V_{SS} = 0V at Ta = 25°C, f(X_{IN}) = 16MHz

Note 2: Divide the f_{AD} if f(X_{IN}) exceeds 10MHz, and make AD operation clock frequency ($ø_{AD}$) equal to or lower than 10MHz.

Table 1.19.5. D-A conversion characteristics (Note 1)

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	(Note 2)			1.5	mA

Note 1: Unless otherwise noted: V_{CC} = V_{REF} =5V, V_{SS} = 0V at Ta = 25°C, f(X_{IN}) = 16MHz

Note 2: The A-D converter's ladder resistance is not included.

When D-A register contents are not "00$_{16}$", the current I_{VREF} always flows even though V_{REF} may have been set to be unconnected by the A-D control register.

Vcc = 5V

Timing requirements (Unless otherwise noted: Vcc = 5V, Vss = 0V at Ta = 25°C)

Table 1.19.6. XIN input

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(XIN)}$	XIN input cycle time	62.5		ns
$t_{wH(XIN)}$	XIN input HIGH pulse width	30		ns
$t_{wL(XIN)}$	XIN input LOW pulse width	30		ns

Table 1.19.7. CNTR0 input

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(CNTR0)}$	CNTR0 input cycle time	100		ns
$t_{wH(CNTR0)}$	CNTR0 input HIGH pulse width	40		ns
$t_{wL(CNTR0)}$	CNTR0 input LOW pulse width	40		ns

Table 1.19.8. TCIN input

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TCIN)}$	TCIN input cycle time	400 (Note1)		ns
$t_{wH(TCIN)}$	TCIN input HIGH pulse width	200 (Note2)		ns
$t_{wL(TCIN)}$	TCIN input LOW pulse width	200 (Note2)		ns

Note1 : Use the greater value,either (1/ digital filter clock frequency x 6) or min. value.
Note2 : Use the greater value,either (1/ digital filter clock frequency x 3) or min. value.

Table 1.19.9. 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 1.19.10. External interrupt $\overline{INTi}$ input

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{w(INH)}$	$\overline{INTi}$ input HIGH pulse width	250 (Note1)		ns
$t_{w(INL)}$	$\overline{INTi}$ input LOW pulse width	250 (Note2)		ns

Note1 : When the $\overline{INT0}$ input filter select bit selects the digital filter, use the $\overline{INT0}$ input HIGH pulse width to the greater value,either (1/ digital filter clock frequency x 3) or min. value.
Note2 : When the $\overline{INT0}$ input filter select bit selects the digital filter, use the $\overline{INT0}$ input LOW pusle width to the greater value,either (1/ digital filter clock frequency x 3) or min. value.

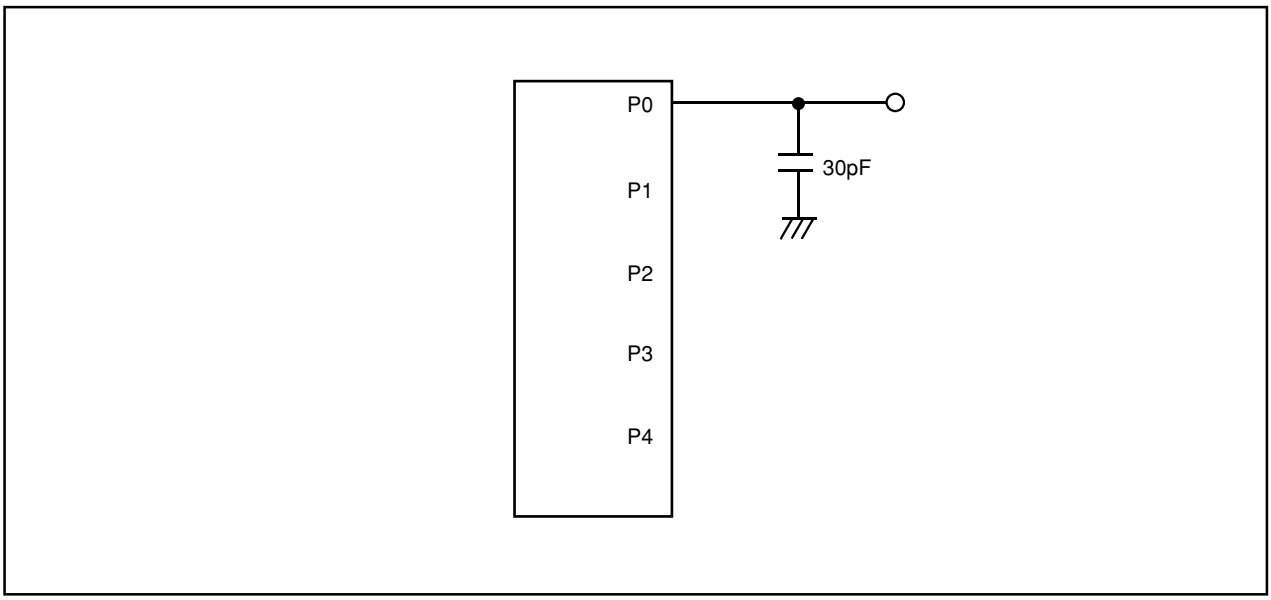
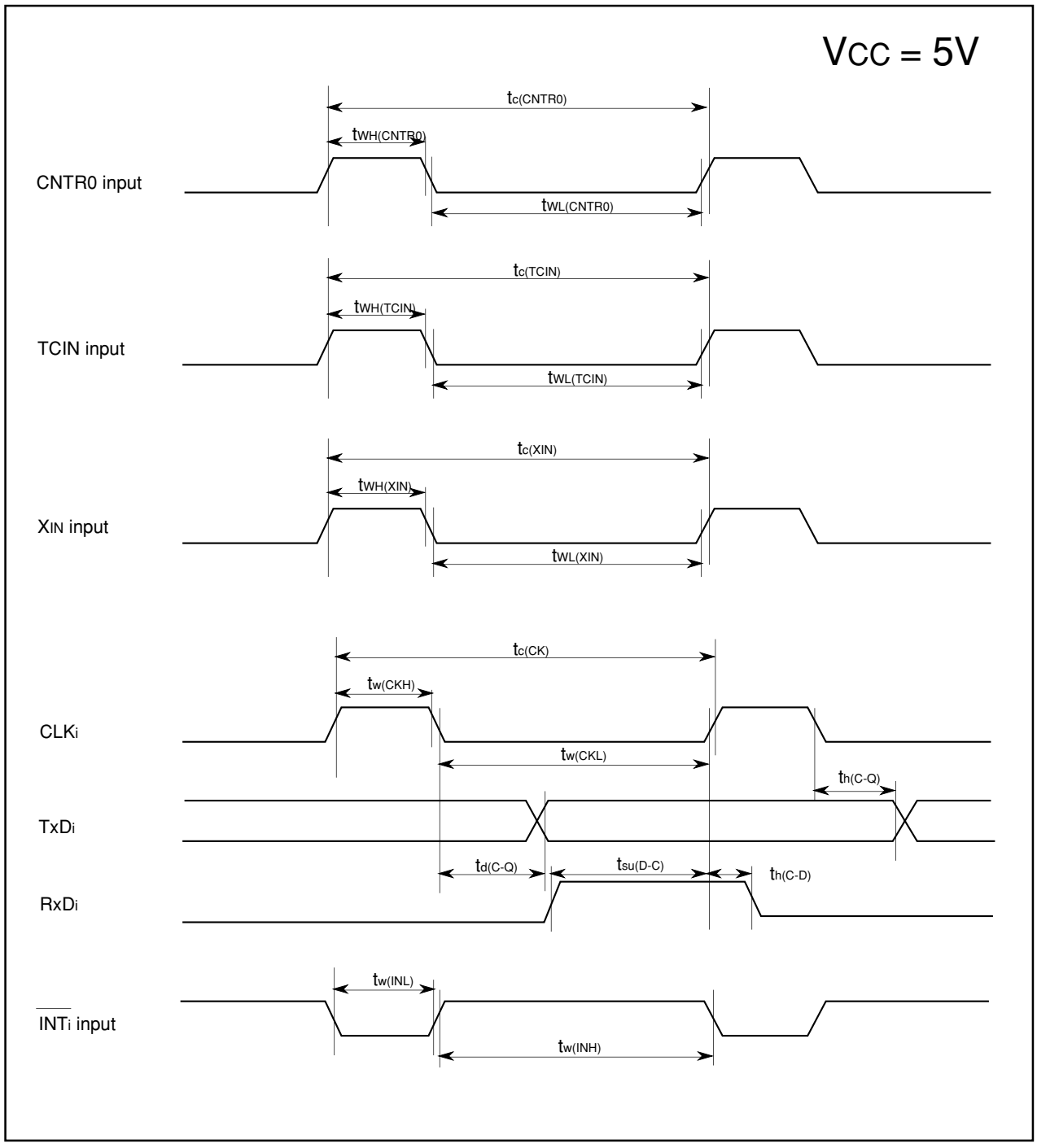


Figure 1.19.1. Port P0 to P4 measurement circuit

Figure 1.19.2. Vcc=5V timing diagram

Electrical characteristics (Vcc = 5V)



Vcc = 3V

Table 1.19.11. (1) Electrical characteristics
(Note: Unless otherwise noted: Vcc = 3V, Vss = 0V at Ta = 25°C, f(XIN) = 5MHz)

Symbol	Parameter		Measuring condition		Standard			Unit
					Min.	Typ.	Max.	
V_{OH}	HIGH output voltage	$P0_0$ to $P0_7$,$P1_0$ to $P1_7$,$P2_0$,$P2_1$, $P3_0$ to $P3_7$,$P4_0$ to $P4_7$	I_{OH} = - 1 mA		2.5			V
V_{OH}	HIGH output voltage	X_{OUT} HIGH POWER	I_{OH} = - 0.1 mA		2.5			V
		X_{OUT} LOW POWER	I_{OH} = - 50 µA		2.5			
V_{OH}	HIGH output voltage	X_{COUT} (Note) HIGH POWER	No load	Flash memory		Vcc		V
			No load	Mask ROM		3.0		
		LOW POWER	No load	Flash memory		Vcc		
			No load	Mask ROM		1.6		
V_{OL}	LOW output voltage	$P0_0$ to $P0_7$,$P2_0$,$P2_1$, $P3_0$ to $P3_7$,$P4_0$ to $P4_7$	I_{OL} = 1 mA				0.5	V
V_{OL}	LOW output voltage	$P1_0$ to $P1_7$ HIGH POWER	I_{OH} = 2 mA				0.5	V
		LOW POWER	I_{OH} = 1mA				0.5	
V_{OL}	LOW output voltage	X_{OUT} HIGH POWER	I_{OH} = 0.1 mA				0.5	V
		LOW POWER	I_{OH} = 50 µA				0.5	
V_{OL}	LOW output voltage	X_{COUT} HIGH POWER	No load			0		V
		LOW POWER	No load			0		
V_{T+} -V_{T-}	Hysteresis	CNTR0,TCIN, $\overline{INT_0}$ to $\overline{INT_3}$,CLK0,CLK1 RxD0, RxD1,KI0 to KI3,P45			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 $P1_7$,$P2_0$,$P2_1$, $P3_0$ to $P3_7$,$P4_0$ to $P4_7$, XIN RESET, CNVss	V_I = 3V				4.0	µA
I_{IL}	LOW input current	$P0_0$ to $P0_7$,$P1_0$ to $P1_7$,$P2_0$,$P2_1$, $P3_0$ to $P3_7$,$P4_0$ to $P4_7$, XIN RESET, CNVss	V_I = 0V				-4.0	µA
R_{PULLUP}	Pull-up resistor	$P0_0$ to $P0_7$,$P1_0$ to $P1_7$,$P2_0$,$P2_1$, $P3_0$ to $P3_7$,$P4_0$ to $P4_7$	V_I = 0V		66.0	120.0	500.0	kΩ
R_{fXIN}	Feedback resistor	XIN				3.0		MΩ
R_{fXCIN}	Feedback resistor	XCIN				10.0		MΩ
V_{RAM}	RAM retention voltage		When clock is stopped		2.0			V
R_{OSC}	Oscillation frequency of Ring oscillator		Mask ROM		150	300	600	kHz
			Flash memory		250	500	1000	

Note: The V_{OH} standard values of X_{COUT} differ between flash memory version and mask ROM version.
Therefore, please note that the oscillation constants of sub clock may differ between these versions.

VCC = 3V

Table 1.19.11. (2) Electrical characteristics (Unless otherwise noted: VCC = 3V, VSS = 0V at Ta = 25°C, f(XIN) = 5MHz)

Symbol	Parameter	Measuring condition			Standard			Unit
					Min.	Typ.	Max.	
Icc	Power supply current	I/O pin has no load	Mask ROM	f(XIN)=5MHz Square wave, no division		4.0	8.0	mA
			Flash memory	f(XIN)=5MHz Square wave, no division		8.0	14.0	mA
			Mask ROM	Ring oscillator mode No division		200		µA
			Flash memory	Ring oscillator mode No division		1000		µA
			Mask ROM	Ring oscillator mode When a WAIT instruction is executed		40		µA
			Flash memory	Ring oscillator mode When a WAIT instruction is executed		350		µA
			Mask ROM	f(XCIN)=32kHz Square wave		30		µA
			Flash memory	f(XCIN)=32kHz Square wave		550		µA
			Mask ROM	f(XCIN)=32kHz When a WAIT instruction is executed		4		µA
			Flash memory	f(XCIN)=32kHz When a WAIT instruction is executed		300		µA
			Mask ROM	Ta=25°C when clock is stopped			2	µA
				Ta=85°C when clock is stopped			20	
			Flash memory	Ta=25°C when clock is stopped		250	500	µA
				Ta=85°C when clock is stopped		250	500	

V_{CC} = 3V

Table 1.19.12. A-D conversion characteristics (Note)

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, ø$_{AD}$=f$_{AD}$/2			±2	LSB
R_{LADDER}	Ladder resistance		V_{REF} =V_{CC}	10		40	kΩ
t_{CONV}	Conversion time(8-bit)			14.0			µs
V_{REF}	Reference voltage			2.7		V_{CC}	V
V_{IA}	Analog input voltage			0		V_{REF}	V

Note: Unless otherwise noted: V_{CC} = V_{REF} =3V, V_{SS} = 0V at Ta = 25°C, f(X_{IN}) = 7MHz

Table 1.19.13. D-A conversion characteristics (Note 1)

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	(Note 2)			1.5	mA

Note 1: Unless otherwise noted: V_{CC} = AV$_{CC}$ =V_{REF} = 3V, V_{SS} = AV$_{SS}$ =0V at Ta = 25°C, f(X_{IN}) = 7MHz

Note 2: The A-D converter's ladder resistance is not included.

When D-A register contents are not "00$_{16}$", the current I_{VREF} always flows even though V_{REF} may have been set to be unconnected by the A-D control register.

Vcc = 3V

Timing requirements (Unless otherwise noted: Vcc = 3V, Vss = 0V at Ta = 25°C)

Table 1.19.14. XIN input

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(XIN)}$	XIN input cycle time	143		ns
$t_{wH(XIN)}$	XIN input HIGH pulse width	70		ns
$t_{wL(XIN)}$	XIN input LOW pulse width	70		ns

Table 1.19.15. CNTR0 input

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(CNTR0)}$	CNTR0 input cycle time	300		ns
$t_{wH(CNTR0)}$	CNTR0 input HIGH pulse width	120		ns
$t_{wL(CNTR0)}$	CNTR0 input LOW pulse width	120		ns

Table 1.19.16. TCIN input

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TCIN)}$	TCIN input cycle time	1200(Note1)		ns
$t_{wH(TCIN)}$	TCIN input HIGH pulse width	600(Note2)		ns
$t_{wL(TCIN)}$	TCIN input LOW pulse width	600(Note2)		ns

Note1 : Use the greater value,either (1/ digital filter clock frequency x 6) or min. value.
Note2 : Use the greater value,either (1/ digital filter clock frequency x 3) or min. value.

Table 1.19.17. 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 1.19.18. External interrupt $\overline{INTi}$ input

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{w(INH)}$	$\overline{INTi}$ input HIGH pulse width	380(Note1)		ns
$t_{w(INL)}$	$\overline{INTi}$ input LOW pulse width	380(Note2)		ns

Note1 : When the $\overline{INT0}$ input filter select bit selects the digital filter, use the $\overline{INT0}$ input HIGH pulse width to the greater value,either (1/ digital filter clock frequency x 3) or min. value.
Note2 : When the $\overline{INT0}$ input filter select bit selects the digital filter, use the $\overline{INT0}$ input LOW pusle width to the greater value,either (1/ digital filter clock frequency x 3) or min. value.

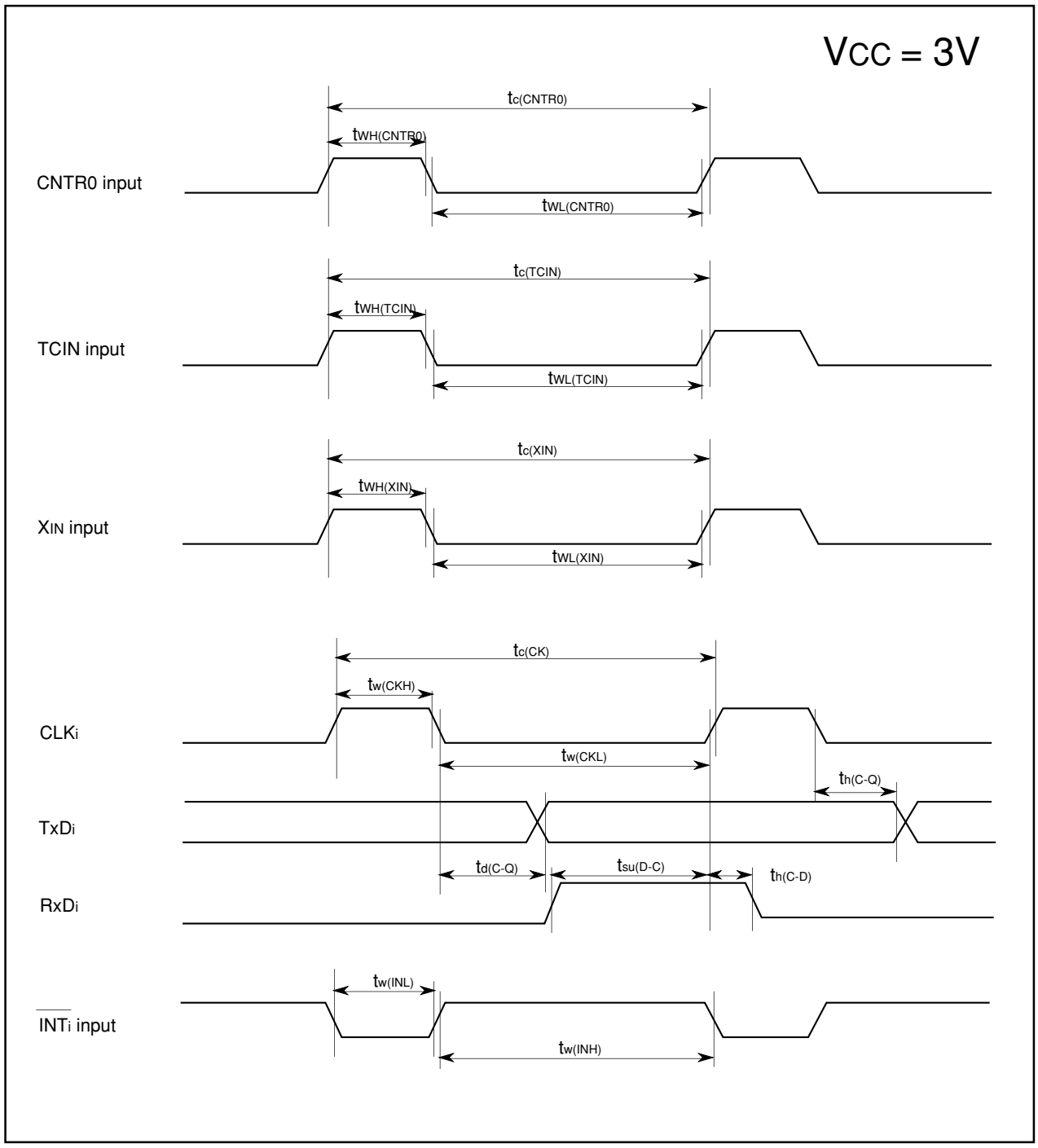


Figure 1.19.3. Vcc=3V timing diagram

Outline Performance

Table 1.20.1 shows the outline performance of the M16C/10 (flash memory version).

Table 1.20.1. Outline performance of the M16C/10 (flash memory version)

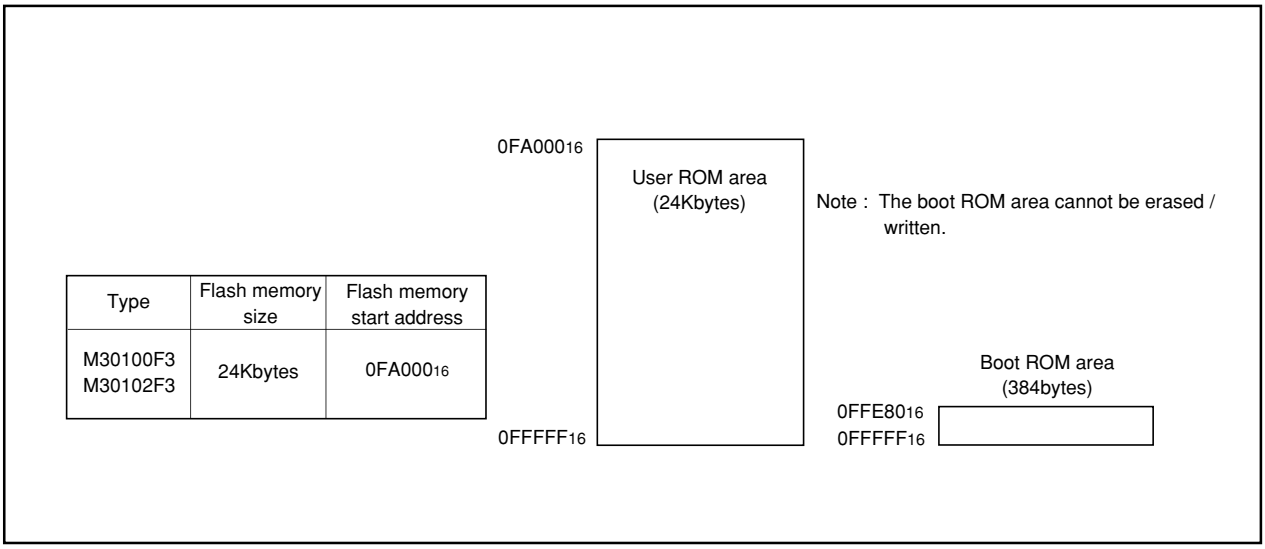
Item		Performance
Power supply voltage		4.2V to 5.5V when f(XIN)=16MHz
Program/erase voltage		Vcc=5.0V±10%
Flash memory operation mode		Standard serial I/O
Erase block division	User ROM area	One division (24 Kbytes)
	Boot ROM area	One division (384 bytes) (Note)
Program method		Collective program
Erase method		Collective erase
Program/erase count		100 times
Data retention		10 years

Note: The boot ROM area contains a control program which is used to communicate with a dedicated external device (writer). This area cannot be erased nor programmed.

Flash Memory

The M16C/10 (flash memory version) contains the flash memory that can be rewritten with a single voltage. For this flash memory, one mode is available in which to read, program, and erase: standard serial I/O mode in which the flash memory can be manipulated using a decicated external device (writer).

Figure 1.20.1 shows the on-chip flash memory. 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 communications with the dedicated external device (writer) in the standard serial I/O mode. This boot ROM area cannot be erased nor rewritten.



Type	Flash memory size	Flash memory start address
M30100F3 M30102F3	24Kbytes	0FA000₁₆

Figure 1.20.1. Block diagram of flash memory version

Pin functions (Flash memory standard serial I/O mode)

Pin	Name	I/O	Description
VCC,VSS	Power input		Apply program/erase protection voltage (5V±10%) to Vcc pin and 0 V to Vss pin.
IVCC	IVCC		Connect a capacitor (0.1μF) to Vss pin.
CNVSS	CNVSS	I	Connect to Vcc.
$\overline{\text{RESET}}$	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	
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.

Mode setup method

Signal	Value
CNVss	Vcc
$\overline{\text{RESET}}$	Vss to Vcc

* Vcc = 5V±10%

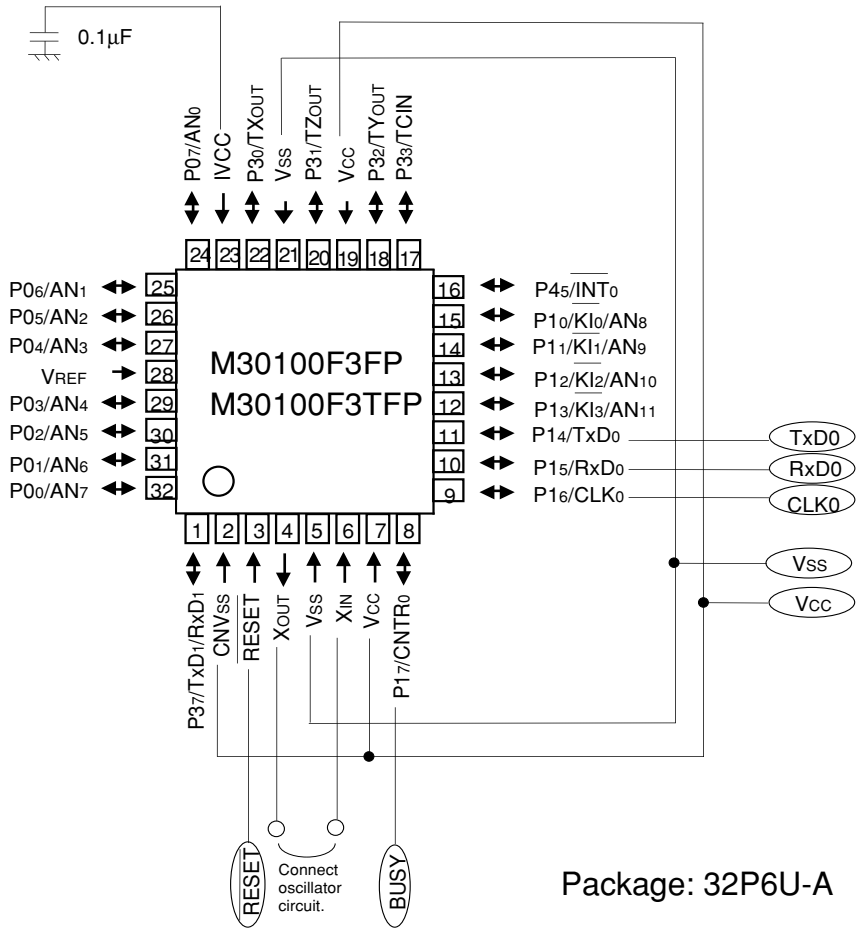


Figure 1.20.2. Pin connections for serial I/O mode (1)

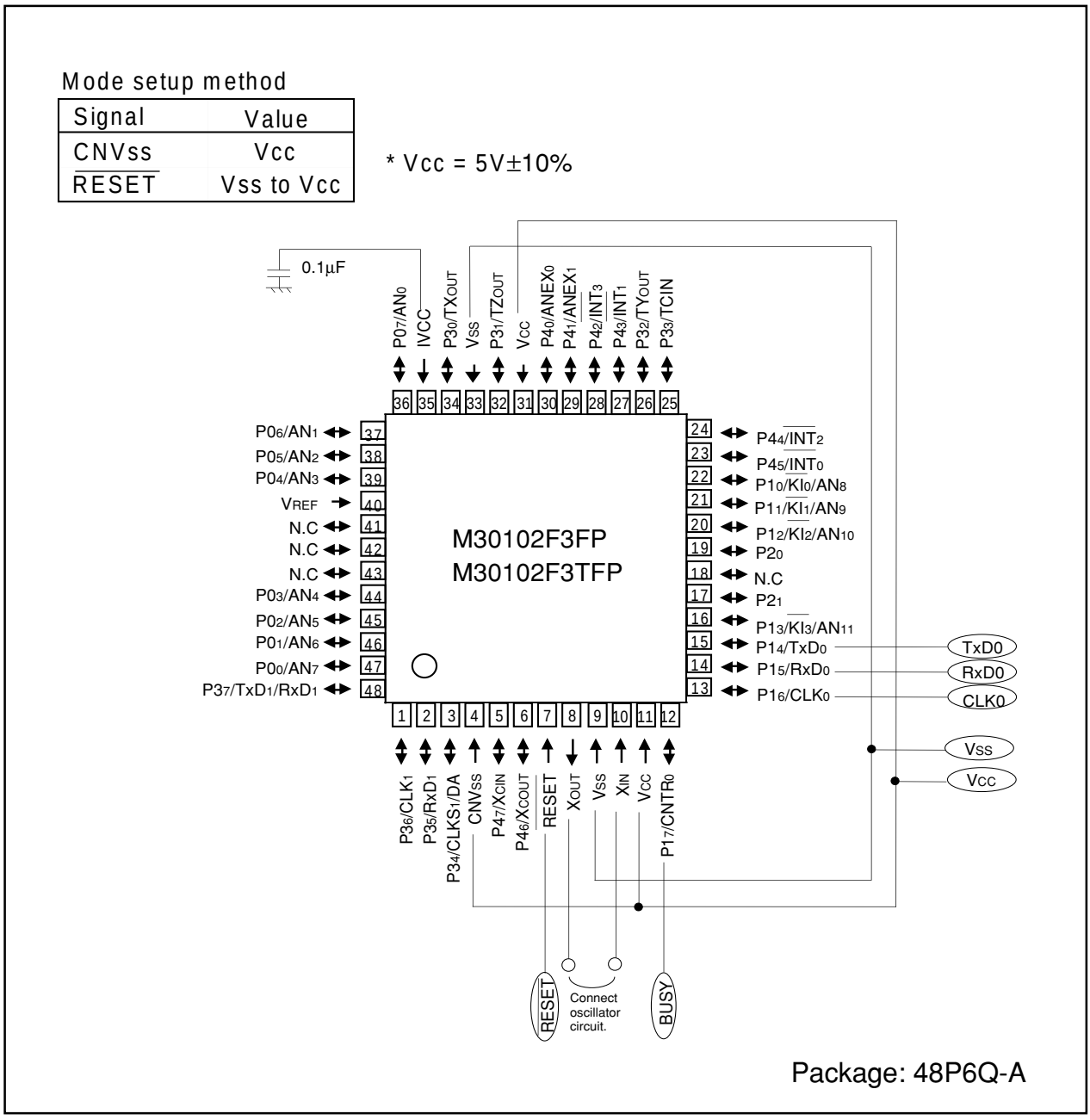


Figure 1.20.3. Pin connections for serial I/O mode (2)

Standard serial I/O mode

The standard serial I/O mode inputs and outputs the control functions, 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 dedicated external device (writer).

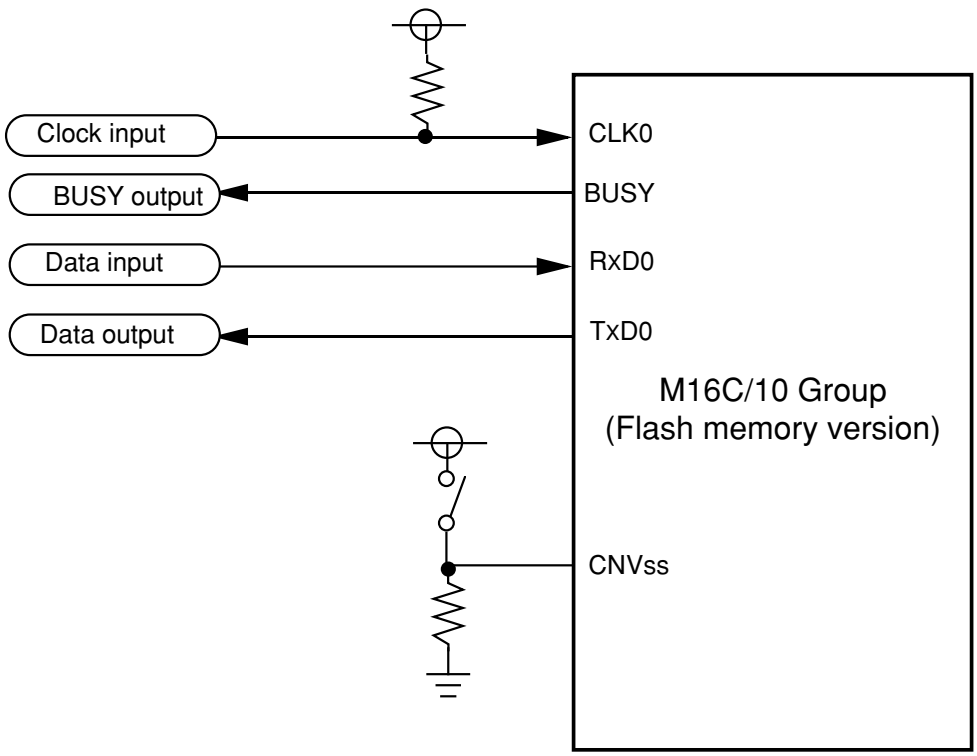
In the standard serial I/O mode, the CPU controls rewrite to the flash memory and communication with the dedicated external device (writer). This mode starts when the reset is released, which is done when the CNVss pin is "H" level. (In the ordinary microprocessor mode, set CNVss pin to "L" level.)
This control program for communications with the dedicated external device (writer) is written in the boot ROM area when the product is shipped from Mitsubishi. Accordingly, make note of the fact that the boot ROM area cannot be erased nor rewritten. Figures 1.20.2 and 1.20.3 show the pin connections for the standard serial I/O mode. The communications with the dedicated external device (writer) uses UART0. Standard serial I/O switches between mode 1 (clock synchronized) and mode 2 (clock asynchronized) according to the level of CLK_0 pin when the reset is released.

To use standard serial I/O mode 1 (clock synchronized), set the CLK_0 pin to "H" level and release the reset. The operation uses the four UART0 pins CLK_0, RxD_0, TxD_0 and BUSY.

To use standard serial I/O mode 2 (clock asynchronized), set the CLK_0 pin to "L" level and release the reset. The operation uses the two UART0 pins RxD_0 and TxD_0. The BUSY pin should be open.

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 dedicated external device (writer), therefore see the dedicated external device (writer) manual for more information.

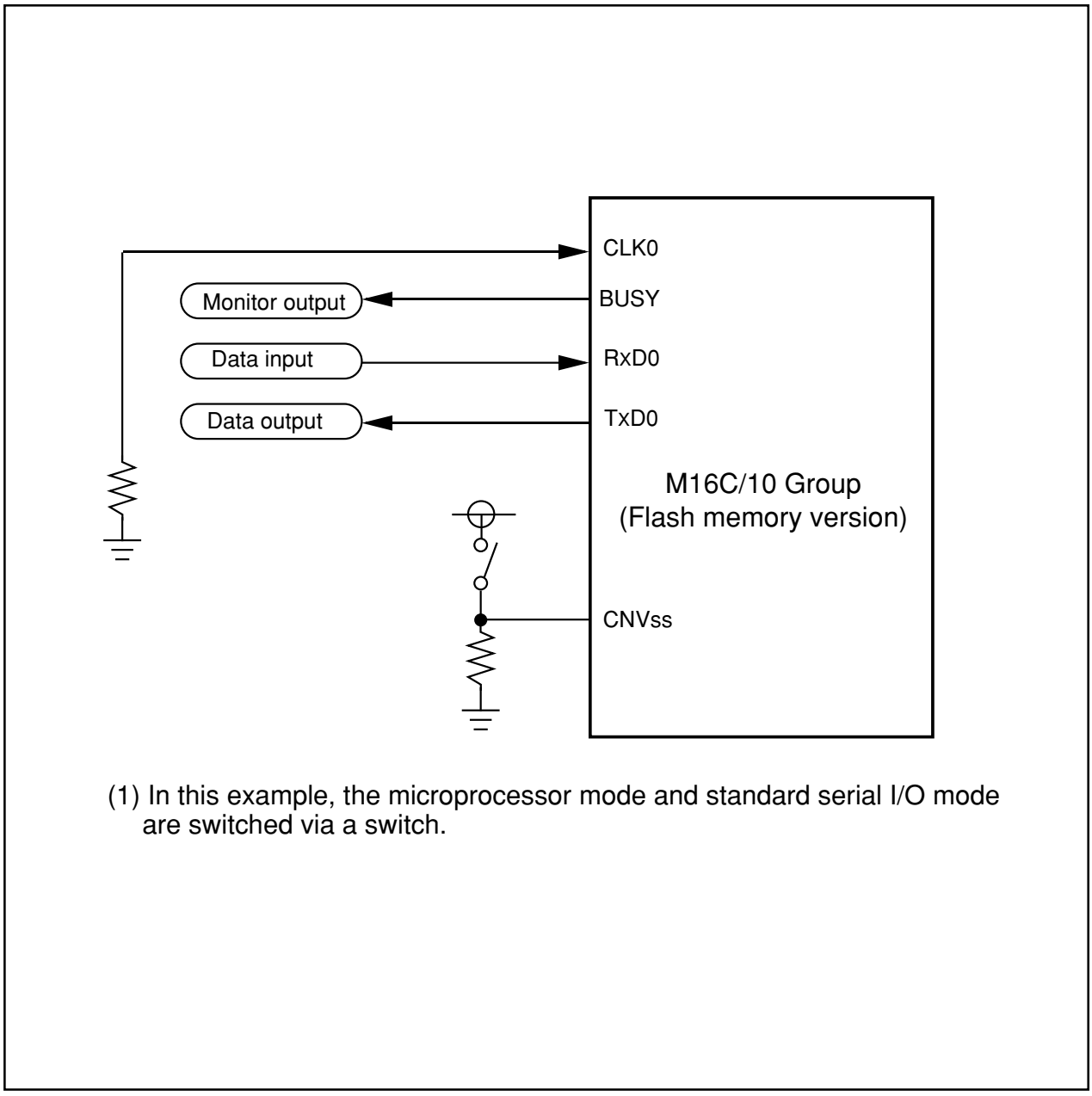


(1) Control pins and external circuitry will vary according to the dedicated external device (writer). For more information, see the dedicated external device (writer) manual.
(2) In this example, the microprocessor mode and standard serial I/O mode are switched via a switch.

Figure 1.20.4. Example circuit application for the standard serial I/O mode 1

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.

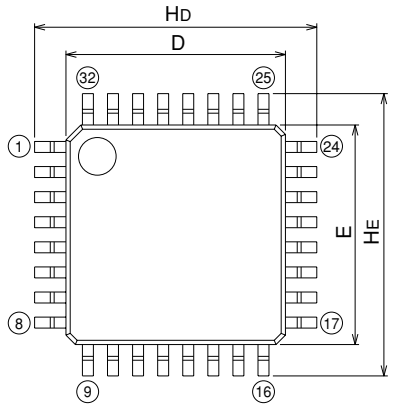
Figure 1.20.5. Example circuit application for the standard serial I/O mode 2

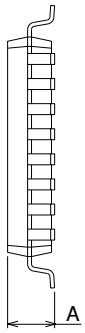
Package

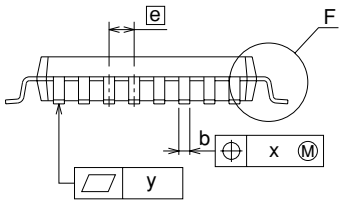
32P6U-A (MMP)

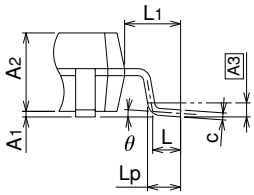
Plastic 32pin 7×7mm body LQFP

EIAJ Package Code	JEDEC Code	Weight(g)	Lead Material
LQFP32-P-0707-0.80	–		Cu Alloy

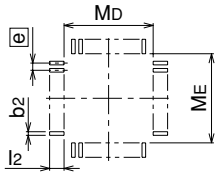








Detail F



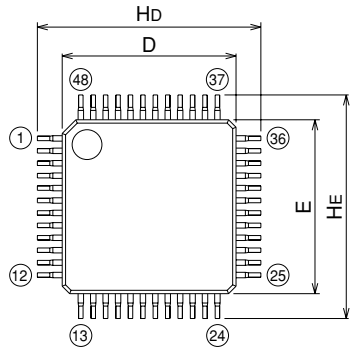
Recommended Mount Pad

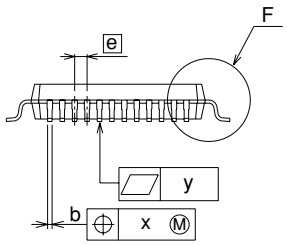
Symbol	Dimension in Millimeters		
	Min	Nom	Max
A	–	–	1.7
A1	0	0.1	0.2
A2	–	1.4	–
b	0.32	0.37	0.45
c	0.105	0.125	0.175
D	6.9	7.0	7.1
E	6.9	7.0	7.1
e	–	0.8	–
HD	8.8	9.0	9.2
HE	8.8	9.0	9.2
L	0.3	0.5	0.7
L1	–	1.0	–
Lp	0.45	0.6	0.75
A3	–	0.25	–
x	–	–	0.2
y	–	–	0.1
θ	0°	–	10°
b2	–	0.5	–
l2	1.0	–	–
MD	–	7.4	–
ME	–	7.4	–

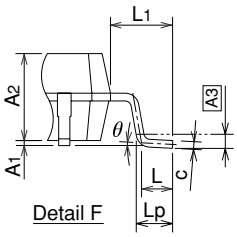
48P6Q-A (MMP)

Plastic 48pin 7×7mm body LQFP

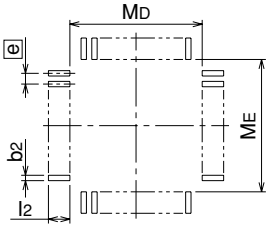
EIAJ Package Code	JEDEC Code	Weight(g)	Lead Material
LQFP48-P-77-0.50	–	–	Cu Alloy







Detail F



Recommended Mount Pad

Symbol	Dimension in Millimeters		
	Min	Nom	Max
A	–	–	1.7
A1	0	0.1	0.2
A2	–	1.4	–
b	0.17	0.22	0.27
c	0.105	0.125	0.175
D	6.9	7.0	7.1
E	6.9	7.0	7.1
e	–	0.5	–
HD	8.8	9.0	9.2
HE	8.8	9.0	9.2
L	0.35	0.5	0.65
L1	–	1.0	–
Lp	0.45	0.6	0.75
A3	–	0.25	–
x	–	–	0.08
y	–	–	0.1
θ	0°	–	8°
b2	–	0.225	–
l2	1.0	–	–
MD	–	7.4	–
ME	–	7.4	–

Rev.	Date	Description	
		Page	Summary
B	04/20/01	All pages	Figure and Table numbers are revised.
		1	Features are partly revised.
		1	Page numbers of Table of Contents are partly revised.
		2 - 4	Figure 1.1.1 to Figure 1.1.3 are partly revised.
		5 - 7	Figure 1.1.4 to Figure 1.1.6 are partly revised.
		8	Table 1.1.1 is partly revised.
		9	Figure 1.1.7 is partly revised.
		9	Figure 1.1.8 is partly revised.
		10	Pin description is partly revised.
		11	Explanation of “Memory” is partly revised.
		15	Explanation of “Reset” is partly revised.
		15	Figure 1.5.2 (example reset circuit for voltage check circuit)is added.
		15	Figure 1.5.3 is partly revised.
		16	Figure 1.5.4 is partly revised.
		17	Explanation of “Software Reset” is partly revised.
		17	Processor mode register 0 in Figure 1.5.5 is partly revised. Note 2 is deleted.
		17	Processor mode register 1 is added to Figure 1.5.5.
		18 - 19	Figure 1.6.1 and Figure 1.6.2 are partly revised.
		21	Table 1.8.1 is partly revised.
		21	Figure 1.8.1 is partly revised. External RC oscillator is added.
		24	Figure 1.8.4 is partly revised.
		25	Figure 1.8.5 is added.
		26	Explanation of “Stop Mode” is partly revised.
		27	Table 1.8.4 is partly revised.
		29	Figure 1.9.1 is partly revised.
		31	Explanation of “Oscillation Stop Detection Function” is partly revised.
		31	Table 1.10.1 is partly revised.
		32	Figure 1.10.1 is partly revised.
		32	Figure 1.10.2 is partly revised. Note 2 and Note 3 is partly revised. Note 5 is deleted.
		33	Explanation of “Oscillation stop detection interrupt enable bit (CM21) partly revised.
		33	Operation select bit (CM27) is deleted.
		35	Figure 1.11.1 is partly revised.
		38	“UART1 receive interrupt” in (1) Special interrupts is partly revised.
		38	“Timer C interrupt” in (2) Peripheral I/O interrupt is partly revised.
		40	Table 1.12.2 is partly revised.
		42	Figure 1.12.3 is partly revised.
		46	Table 1.12.5 is partly revised.
		50	Figure 1.12.9 is partly revised.
		51	Explanation of “$\overline{\text{INT}}$ interrupt” is partly revised.
		51	External interrupt enable register in Figure 1.12.10 is partly revised.
		53	Explanation of “Key interrupt” is partly revised.
		53	Figure 1.12.13 is partly revised.
		53	Figure 1.12.14 is partly revised.
		56	Explanation of “Watchdog Timer” is partly revised.
		58	Explanation of “Timer 1” is partly revised.
		59	Figure 1.14.2 is partly revised.
		59	Figure 1.14.3 is partly revised. Note 1 and Note 2 are added.
		60	Explanation of “(2) Pulse output mode” is partly revised.
		60	Explanation of “(4) Pulse width measure mode” is partly revised.

Rev.	Date	Description	
		Page	Summary
		61	Explanation of “(5) Pulse period measure mode” is partly revised.
		61	Explanation of precaution is partly revised.
		61	Figure 1.14.4 is partly revised.
		62	Explanation of Timer Y is partly revised.
		63	Explanation of “(2) Programmable waveform generation mode” is partly revised.
		64	Figure 1.14.5 is partly revised.
		65	Figure 1.14.6 is partly revised.
		65	Note 1 and Note 2 are added to Timer Y, Z mode register in Figure 1.14.6.
		66	Explanation of Timer Z is partly revised.
		66	Figure 1.14.7 is partly revised.
		67	Figure 1.14.9 is partly revised.
		68	Explanation of “(2) Programmable waveform generation mode” is partly revised.
		72	Explanation of “Timer C” is partly revised.
		72	Figure 1.14.10 is partly revised.
		72	Note 1 is added to Timer C control register 0 in Figure 1.14.10.
		73	Figure 1.14.11 is partly revised.
		73	Figure 1.14.12 is partly revised.
		74	Table 1.14.1 and its Note are partly revised.
		75	Figure 1.15.1 is partly revised.
		78	Note is added to UARTi transmit/receive mode register in Figure 1.15.4 is partly revised.
		78	UARTi transmit/receive control register 0 in Figure 1.15.4 is partly revised.
		78	Note 1 and Note 2 of UARTi transmit/receive control register 0 in Figure 1.15.4 are deleted .
		79	Note 1 is added to UARTi transmit/receive control register 1 in Figure 1.15.5.
		79	UARTi transmit/receive control register 2 is added to Figure 1.15.5.
		81	Table 1.15.2 is partly revised.
		84	Figure 1.15.10 is partly revised.
		85	Table 1.15.3 is partly revised.
		86	Table 1.15.4 is partly revised.
		89	Explanation of “A-D Converter” is partly revised.
		89	Table 1.16.1 is partly revised.
		90	Figure 1.16.1 is partly revised.
		91	A-D control register 0 in Figure 1.16.2 is partly revised.
		92	Figure 1.16.3 is partly revised.
		94	“Input pin” and Note in Table 1.16.3 are revised.
		94	Figure 1.16.5 is partly revised.
		95	Explanation of “Extended analog input pins” is partly revised.
		95	Explanation of “External operation amp connection mode” is partly revised.
		96	Explanation of “D-A Converter” is partly revised.
		96	Figure 1.17.1 is partly revised.
		97	Figure 1.17.3 is partly revised.
		98	Explanation of “Programmable I/O Ports” is partly revised.
		100	Figure 1.18.2 is revised.
		101	Figure 1.18.3 is revised.
		102	Figure 1.18.4 is revised.
		103	Figure 1.18.5 is revised.
		104	Figure 1.18.6 is revised.
		105	Table 1.18.1 is revised.
B1	05/15/01	15	Figure 1.5.3 is partly revised.
		19	Figure 1.6.2 is partly revised.
		31	Table 1.10.1 is partly revised.

Rev.	Date	Description	
		Page	Summary
		32	Figure 1.10.1 is partly revised.
		51	Explanation of “$\overline{\text{INT}}$ interrupt” is partly revised.
		89	Note 3 is added to Table 1.16.1.
		91	Figure 1.16.2 is partly revised.
		92	Figure 1.16. 3 is partly revised.
		93	Table 1.16.2 is partly revised.
		94	Table 1.16.3 is partly revised.
		94	Figure 1.16.5 is partly revised.
C1	11/20/01	01	Features are partly revised.
		01	Page numbers of Table of Contents are partly revised.
		02 - 05	Figure 1.1.1 to 1.1.4 are partly revised.
		08	Table 1.1.1 is partly revised
		09	Explanation of (3) package if partly revised.
		09	Figure 1.1.7 is partly revised.
		10	Pin description is partly revised.
		11	Figure 1.3.1 is partly revised.
		15	Explanation of reset is partly revised.
		15	Figure 1.5.3 is partly revised.
		16	Figure 1.5.4 is partly revised.
		19	Figure 1.6.2 is partly revised.
		21	Figure 1.8.1 is partly revised.
		22	Explanation is partly revised.
		22	Figure 1.8.3 is partly revised.
		23	Explanation of (1)main clock, (3)BCLK and (7)f_{RING} are partly revised.
		24	Note (2) and (5) of register CM1 in Figure 1.8.4 are partly revised.
		25	Register CM2 in Figure 1.8.5 is partly revised.
		26	Explanation of stop mode is partly revised.
		27	Explanation of “status transition of BCLK”, (3)division by 8 mode, (5)no-division mode and (8)ring oscillation mode are partly revised.
		27	Note is added.
		29	Explanation of power control is partly revised.
		31	Figure 1.9.2 is partly revised.
		32	Explanation of oscillation stop detection function is partly revised.
		32	Table 1.10.1 is partly revised.
		33	Figure 1.10.1 and 1.10.2 are partly revised.
		34	Explanation of CM20 to CM22 are partly revised.
		36	Explanation of protection is partly revised.
		39	Explanation of “UART1 receive interrupt” of (1)special interrupts is partly revised.
		39	Explanation of “$\overline{\text{INT0}}$ to $\overline{\text{INT3}}$ interrupt” of (2)peripheral I/O interrupts is partly revised.
		41	Table 1.12.2 is partly revised.
		45	Program examples are partly revised.
		52	Explanation of $\overline{\text{INT}}$ interrupt is partly revised.
		52	Figure 1.12.10 is partly revised.
		53	Figure 1.12.11 is partly revised.
		54	Explanation of key input interrupt is partly revised.
		54	Figure 1.12.13 and 1.12.14 is partly revised.
		56	Explanation of (1)reading address 0000_{16} is partly revised.
		59 - 72	“Latch” used for reload register related are changed to “reload register.”
		59 - 72	Expression of counter content is changed from “00h” to “00_{16}.”

Rev.	Date	Description	
		Page	Summary
		60	Figure 1.14.3 is partly revised.
		61	Explanation of (3) event counter mode is partly revised.
		61	Explanation of (4) pulse width measure mode is partly revised.
		62	Explanation of (5) pulse period measure mode is partly revised.
		62	Figure 1.14.4 is partly revised.
		64	Explanation of (2) programmable waveform generation mode is partly revised.
		65	Explanation of “use of the waveform extend function” is added.
		65	Last paragraph in precaution is partly revised.
		66	Figure 1.14.6 is partly revised.
		67	Figure 1.14.7 is partly revised.
		68	Figure 1.14.8 and 1.14.9 are partly revised.
		69	Explanation of (2) programmable waveform generation mode is partly revised.
		70	Explanation of (3) programmable one-shot generation mode is partly revised.
		71	Explanation of (4) programmable wait one-shot generation mode is partly revised.
		72	Explanation of “use of the waveform extend function” is added.
		72	Explanation of “change of set count values” is partly revised.
		72	Last paragraph in precaution is partly revised.
		73	Figure 1.14.10 is partly revised.
		76	Figure 1.15.1 is partly revised.
		77	Figure 1.15.2 is partly revised.
		78	Figure 1.15.3 is partly revised.
		79	Figure 1.15.4 is partly revised.
		80	Figure 1.15.5 is partly revised.
		81	Table 1.15.1 is partly revised.
		83	Figure 1.15.7 is partly revised.
		85	(e) RxD_1 input pin selection function (UART) is added.
		86	Table 1.15.3 is partly revised.
		89	Figure 1.15.13 is partly revised.
		89	(b) RxD_1 input pin selection function (UART) is added.
		90	Table 1.16.1, Note 2 is partly revised.
		91	Figure 1.16.1 is partly revised.
		92	Figure 1.16.2 is partly revised.
		94	Figure 1.16.4 is partly revised
		95	Table 1.16.3 is partly revised.
		95	Figure 1.16.5 is partly revised
		96	Explanation of extended analog input pins is partly revised.
		96	Figure 1.16.6 is partly revised.
		99	Explanation of programmable I/O ports is partly revised.
		100 - 105	Figure 1.18.1 to 1.18.6 are partly revised.
		106	Table 1.18.1 is partly revised.
		107 - 108	Explanation of usage of precaution is added.
		109 - 119	Section of electric characteristics is added.
D	July/08/02	1	Explanation of overview is partly revised.
		3(ver.C)	Figure 1.1.2 is deleted.
		6(ver.C)	Figure 1.1.5 is deleted.
		6	Table 1.1.1 is partly revised.
		7	(3)Package is partly revised.
		7	Figures 1.1.7 and 1.1.8 are partly revised.
		8	Explanation on CNVss of pin description is partly revised.
		9	Explanation of operation of functional blocks is partly revised.

Rev.	Date	Description	
		Page	Summary
		9	Explanation of memory is partly revised.
		9	Figure 1.3.1 is partly revised.
		14	Figure 1.5.4 is partly revised.
		18	Figure 1.7.1 is partly revised.
		19	Table 1.8.1 is partly revised.
		20	Figure 1.8.3 is partly revised.
		21	Explanation of (1) main clock is partly revised.
		21	Explanation of (5) fc_{32} is partly revised.
		23	Figure 1.8.5 is partly revised.
		23	Figure 1.8.6 is added.
		30	Explanation of oscillation stop detection function is partly revised.
		31	Figure 1.10.2 is partly revised.
		35	Figure 1.12.1 is partly revised.
		37	Explanation of UART0 receive interrupt of (1)special interrupts is partly revised.
		37	Explanation of CNTR0 interrupt and TCIN interrupt are added to (2)peripheral I/O interrupts instead of CNTR0 and TCIN interrupt.
		38	Table 1.12.1 is partly revised.
		49	Figure 1.12.8 is partly revised.
		49	Figure 1.12.9 is partly revised.
		51	Explanation of INT0 input filter is partly revised.
		51	Figure 1.12.11 is partly revised.
		53(rev.C)	Explanation of UART0 Receive Hardware Input and Figure 1.12.12 are deleted.
		52	Explanation of CNTR0 interrupt and Figure 1.12.13 are added.
		53	Explanation of TCIN interrupt and Figure 1.12.14 are added.
		57	Figure 1.13.1 is partly revised.
		58	PM1 register is added to Figure 1.13.2.
		59-94	Timer: Full-fledged revision
		97	Note is added to UARTi transmit buffer register in Figure 1.15.3.
		98	UARTi transmit/receive control register 0 of Fig 1.15.4 is partly revised.
		99	Note of UARTi transmit/receive control register 1 of Fig 1.5.5 is partly revised.
		99	Note 2 is added to UART transmit/receive control register 2 in Fig 1.15.5.
		100	Table 1.15.1 Note 1 is partly revised.
		101	Table 1.15.2 is partly revised.
		102	Figure 1.15.7 is partly revised.
		104	Explanation of (e) is partly revised.
		105	Note 1 of Table 1.15.1 is partly revised.
		106	Table 1.15.4 is partly revised.
		108	Explanation of (b) is partly revised.
		113	Figure 1.16.4 is partly revised.
		114	Figure 1.16.5 is partly revised.
		118	Figure numbers are revised.
		119-122	Figures 1.18.1 to 1.18.4 are partly revised.
		123	Figure 1.18.5 is partly revised.
		124	Figure 1.18.6 is partly revised.
		126	Note 1 is added to Table 1.18.1.
		127	(2) is added to explanation of precautionary note of serial I/O.
		129	(3) is added to explanation of precautionary note of noise.
		129	Explanation of precautionary notes of Timers Y, Z and C are added.
		130	Notes 2 and 3 of Table 1.19.2 are partly revised.
		131	Note 5 of Table 1.19.2 is partly revised.

Rev.	Date	Description	
		Page	Summary
		132-133	Table 1.19.3 (1) and (2) are partly revised.
		138-139	Table 1.19.11 (1) and (2) are partly revised.
		143-150	Section “Flash memory version” is added.
D1	Aug/09/02	1	Explanation of overview is partly revised.
		1	Power supply voltage in Features is partly revised.
		1	Flash memory version is added to the table of contents.
		2	Fig 1.1.1 is partly revised
		3	Fig 1.1.2 is partly revised.
		6	Table 1.1.6 is partly revised.
		13	Figs 1.5.1 and 1.5.2 are partly revised.
		16	Note is added to Figs 1.6.1 and 1.6.2.
		19	Note is added to Table 1.8.1.
		19	Note 2 is added to Fig 1.8.1.
		20	Explanation of ring oscillator is partly revised.
		20	Fig 1.8.3 is partly revised.
		21	Explanation of (3)BCLK is partly revised.
		21	Explanation of (4) peripheral function clock is partly revised.
		22	Note 8 for CM0 and Note 1 for CM1 in Fig 1.8.4 are partly revised.
		24	Explanation of stop mode is partly revised.
		25	Explanation of (5) no-division mode is partly revised.
		27	Explanation of (3) stop mode is partly revised.
		43	Explanation of changing the interrupt request bit is added.
		56	Explanation of changing the interrupt request bit is added.
		57	Explanation of WDT is partly revised.
		59	Explanation of timer is partly revised.
		59	Note is added to Table 1.14.1.
		60	Fig 1.14.1 is partly revised.
		61	TCSS register in Fig 1.14.2 is partly revised.
		62	Fig 1.14.3 is partly revised.
		63	TCSS register in Fig 1.14.5 is partly revised.
		69	Table 1.14.7 partly revised.
		69	Note is added to Table 1.14.7.
		69	Note is deleted from Fig 1.14.11.
		70	Fig 1.14.13 is partly revised.
		71	TYZOC register in Fig 1.14.15 is partly revised.
		72	TCSS register in Fig 1.14.16 is partly revised.
		73	Note 1 and Note 2 are added to Table 1.14.8.
		75	Explanation of (2) programmable waveform generation mode is partly revised.
		75	Notes of Table 1.14.9 are partly revised.
		77	Note is added to Fig 1.14.19.
		78	Fig 1.14.20 is partly revised.
		81	TCSS register in Fig 1.14.23 is partly revised.
		82	TYZOC register in Fig 1.14.24 is partly revised.
		83	Note 1 and Note 2 are added to Table 1.14.10.
		85	Explanation of (2) programmable waveform generation mode is partly revised.
		85	Notes of Table 1.14.11 are partly revised.
		87	Notes of Table 1.14.12 are partly revised.
		88	Note 3 of PUM in Fig 1.14.27 is partly revised.
		89	Fig 1.14.28 is partly revised.

Rev.	Date	Description	
		Page	Summary
		90	Explanation of (4) programmable wait one-shot generation mode is partly revised.
		90	Notes of Table 1.14.13 are partly revised.
		92	Fig 1.14.30 is added.
		93	Table 1.14.14 are partly revised.
		94	TCC0 register in Fig 1.14.32 is partly revised.
		95	Fig 1.14.33 is partly revised.
		98	Notes of UiTB register and UiRB register in Fig 1.15.3 are partly revised.
		99	UiMR register in Fig 1.15.4 is partly revised.
		106	Sleep mode is deleted from select function on Table 1.15.3.
		107	Fig 1.15.11 is partly revised.
		109	Explanation of (a) sleep mode is deleted.
		110	Note 2 in Table 1.16.1 is partly revised.
		127	Explanation of serial I/O is partly revised.
		127	Explanation #5 is added to A-D converter.
		128	Explanation of stop mode is added.
		129	Explanation of changing the interrupt request bit is added.
		130	Explanation of Timer 1 and Timer X, Y, Z are added.
		130	Explanation of #2 of Timer Y is partly revised.
		130	Explanation of #2 of Timer Z is partly revised.
		130	Explanation of #1 of Timer C is partly revised.
		132	Note 1 is added to Table 1.19.2.
		133-134	Table 1.19.3 (1) and (2) are partly revised.
		139-140	Table 1.19.11 (1) and (2) are partly revised.
		147	Fig 1.20.1 is partly revised.
		148	Fig 1.20.2 is partly revised.
E	Dec/20/02	1	Table of Contents is partly revised.
		2	Fig 1.1.1 is partly revised.
		3	Fig 1.1.2 is partly revised.
		13	Explanation of Reset and Fig.1.5.1 are partly revised.
		19	Table1.18.1 is partly revised.
		20	Fig 1.8.2 and Fig 1.8.3. are partly revised.
		22	Fig 1.8.4 is partly revised.
		25	Note is partly revised.
		26	Table 1.8.4 is partly revised.
		49	Fig 1.12.9 is partly revised.
		50	Explanation of INT interrupt is partly revised.
		52	Note 2 and Note 3 are added to Table 1.12.12.
		53	Fig 1.12.13. is partly revised.
		54	Header is partly revised.
		54	Explanation of key input interrupt is partly revised.
		55	Header is partly revised.
		57	Explanation of WDT and Fig 1.13.1 are partly revised.
		61	Note 5 is added to Table 1.14.2.
		62	Fig 1.14.4 is partly revised. Note 2 and Note 3 are added to Fig 1.14.4.
		63	Note 5 is added to Table 1.14.5.
		64	Fig 1.14.6 is partly revised. Note 1 is added to Fig 1.14.6.
		65	Fig 1.14.7 is partly revised. Note 3 is added to Fig 1.14.7.
		66	Fig 1.14.8 is partly revised. Note 1 is added to Fig 1.14.8.
		67	Fig 1.14.9 is partly revised. Note 1 is added to Fig 1.14.9.
		69	Table 1.14.7 is partly revised. Note is partly revised.

Rev.	Date	Description	
		Page	Summary
E	Dec/20/02	69	Fig 1.14.11.is partly revised. Note 1 and Note 2 are added to Fig 1.14.11.
		70	Fig 1.14.12 is added.
		73	Fig 1.14.16 is partly revised. Note 5 is added to Fig 1.14.16.
		75	Note 2 is added to Fig 1.14.17.
		79	Fig 1.14.20 is partly revised.
		82	Note 4 is added to Timer Y,Z waveform output control register in Fig 1.14.23.
		82	Note 5 is added to Timer count source setting register in Fig 1.14.23.
		83	Note 2 is partly revised to Timer Y,Z output control register in Fig 1.14.24.
		83	Note is added to External input enable register in Fig 1.14.24.
		85	Note 2 is added to Fig 1.14.25.
		89	Fig 1.14.27 is partly revised.
		95	Fig 1.14.32 is partly revised.
		104	Fig 1.15.7 is partly revised.
		110	Fig 1.15.13 is partly revised.
		117	Fig 1.16.6 is partly revised.
		127	Table 1.18.1 is partly revised and Note 3 is added.
		131	Explanation of Timer X is added.
		135	Table 1.19. 3 (1) is partly revised.
		136	Table 1.19. 3 (2) is partly revised.
		138	Table 1.19.6, Table 1.19.7 and Table 1.19.8 is changed to Table.1.19.7, Table 1.19.8 and Table 1.19.6.
		138	Note 1 and Note 2 are added to Table 1.19.8.
		138	Note 1 and Note 2 are added to Table 1.19.10.
		141	Table 1.19. 11 (1) is partly revised.
		142	Table 1.19. 11 (2) is partly revised.
		144	Table 1.19.14, Table 1.19.15 and Table 1.19.16 is changed to Table.1.19.15, Table 1.19.16 and Table 1.19.14.
		144	Note 1 and Note 2 are added to Table 1.19.16.
		144	Note 1 and Note 2 are added to Table 1.19.18.
		149	Fig 1.20.2 is partly revised.
		150	Fig 1.20.3 is partly revised.
		154	Package is added.
E1	Feb/13/03	6	Table 1.1.1 value of power consumption
		7	Fig 1.1.5 is partly revised.
		9	Fig 1.3.1 is partly revised.
		29	Fig 1.9.2 is partly revised.
		134	Table 1.19.2 IOL(peak)/IOL(avg)
		136	Table 1.19.3(2) Icc
		142	Table 1.19.11(2) Icc

Renesas Technology Corp.
Nippon Bldg.,6-2,Otemachi 2-chome,Chiyoda-ku,Tokyo,100-0004 Japan

Printed in Japan (ROD) II
New publication, effective February. 2003.
Specifications subject to change without notice.

RENESAS
Renesas Technology Corp.