Description

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

The M16C/60 group includes a wide range of products with different internal memory types and sizes and various package types.

Features

- Memory capacity ..... ROM (See Figure 1.4. ROM Expansion)
  RAM 10K bytes
- Shortest instruction execution time ..... 100ns (f(X_{IN})=10MHz)
- Supply voltage ..... 4.0 to 5.5V (f(X_{IN})=10MHz)
  2.7 to 5.5V (f(X_{IN})=7MHz with software one-wait)
- Low power consumption ..... 18mW (f(X_{IN})=7MHz, with software one-wait, Vcc = 3V)
- Interrupts ..... 17 internal and 5 external interrupt sources, 4 software interrupt sources; 7 levels (including key input interrupt)
- Multifunction 16-bit timer ..... 5 timers + 3 timers
- Serial I/O (UART or clock synchronous) ..... 2 channels
- DMAC ..... 2 channels (trigger: 15 sources)
- A-D converter ..... 10 bits X 8 channels
  (Expandable up to 10 channels)
- D-A converter ..... 8 bits X 2 channels
- CRC calculation circuit ..... 1 circuit
- Watchdog timer ..... 15 bits
- Programmable I/O ..... 87 lines
- Input port ..... 1 line ($P8_5$ shared with $\overline{NMI}$ pin)
- Memory expansion ..... Available (to a maximum of 1M bytes)
- Chip select output ..... 4 lines
- Clock generating circuit ..... 2 built-in clock generation circuits
  (built-in feedback resistor, and external ceramic or quartz oscillator)

Applications

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

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

Pin Configuration

Figures 1.1 and 1.2 show the pin configurations (top view).

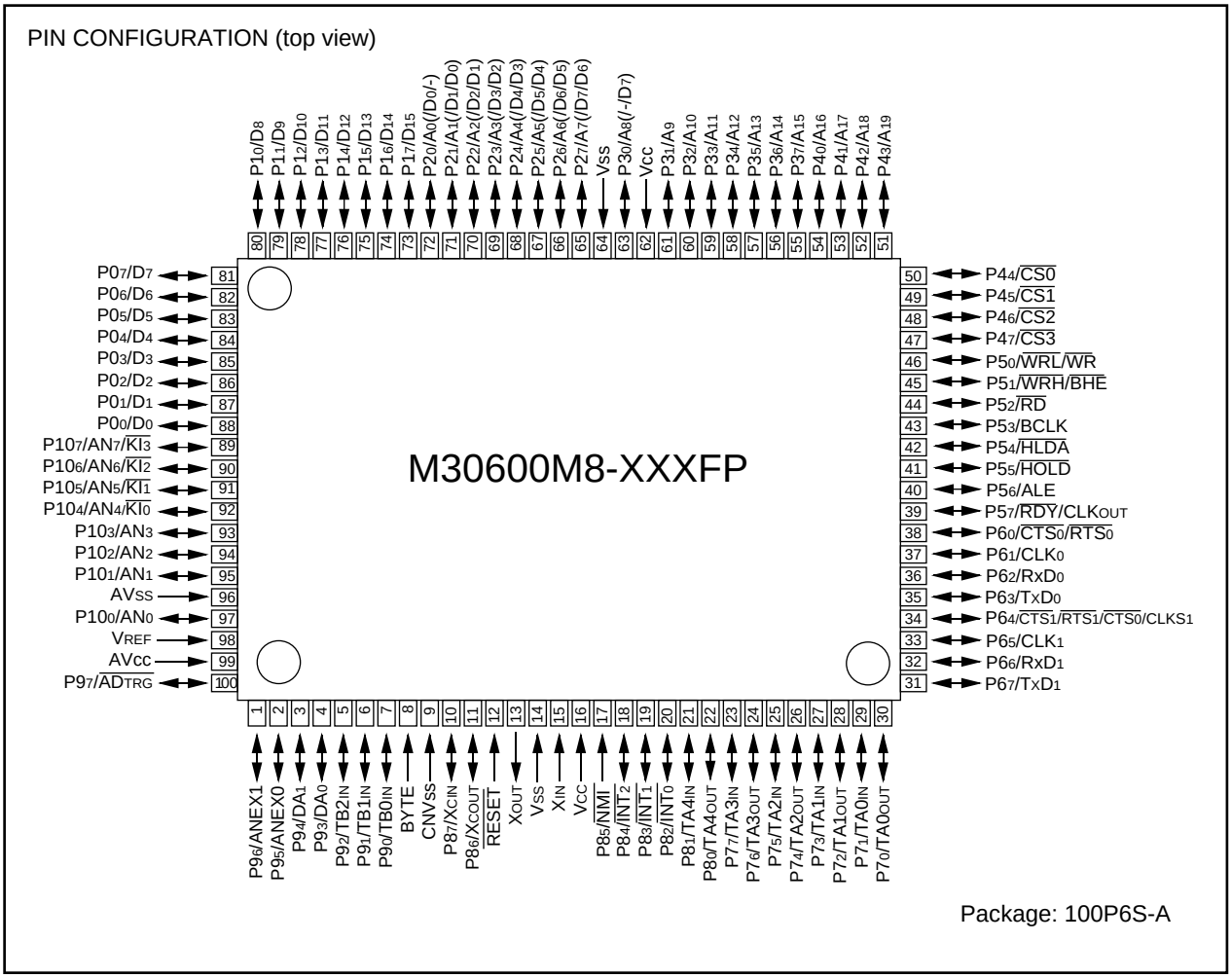


Figure 1.1. Pin configuration (top view)

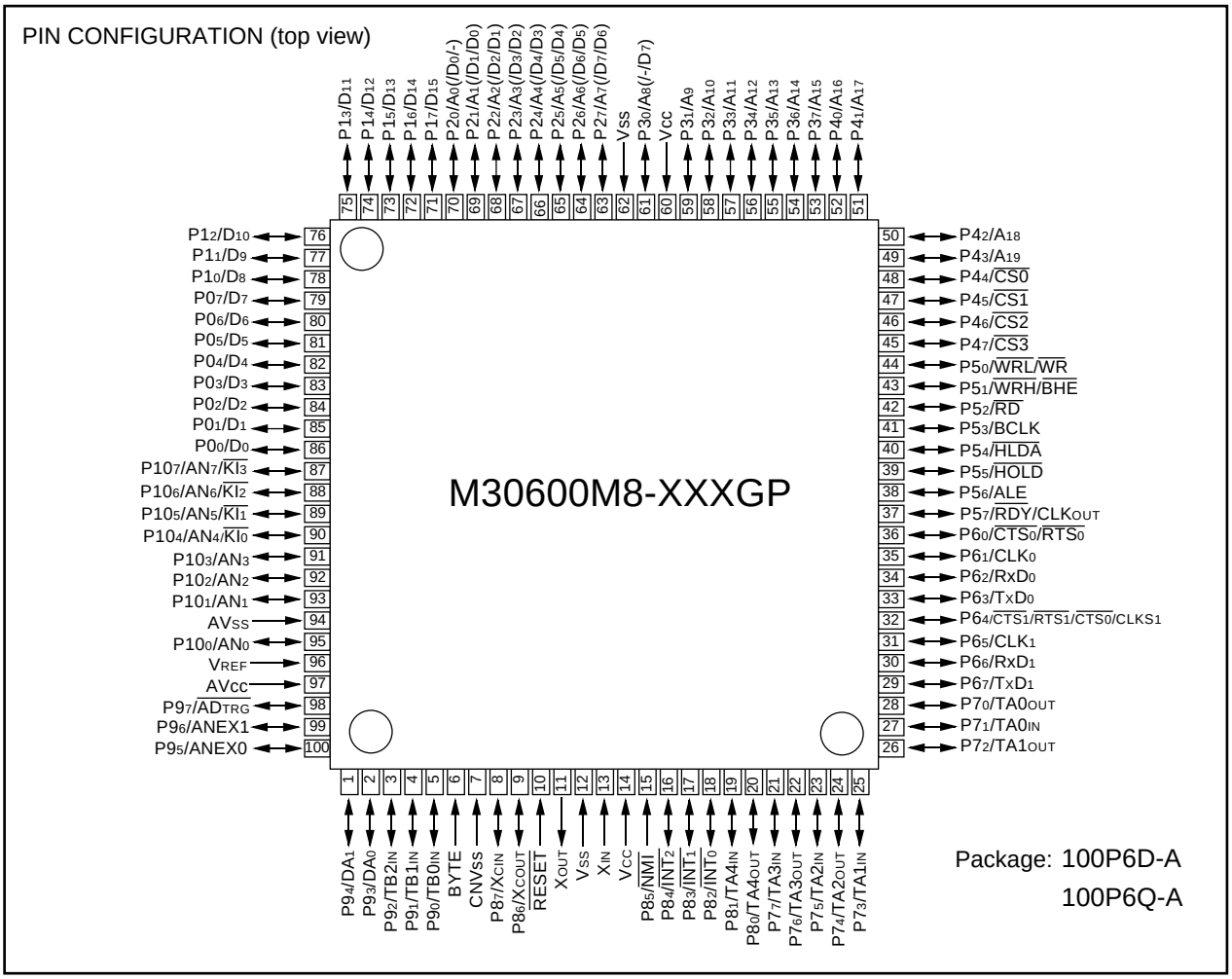


Figure 1.2. Pin configuration (top view)

Block Diagram

Figure 1.3 is a block diagram of the M16C/60 group.

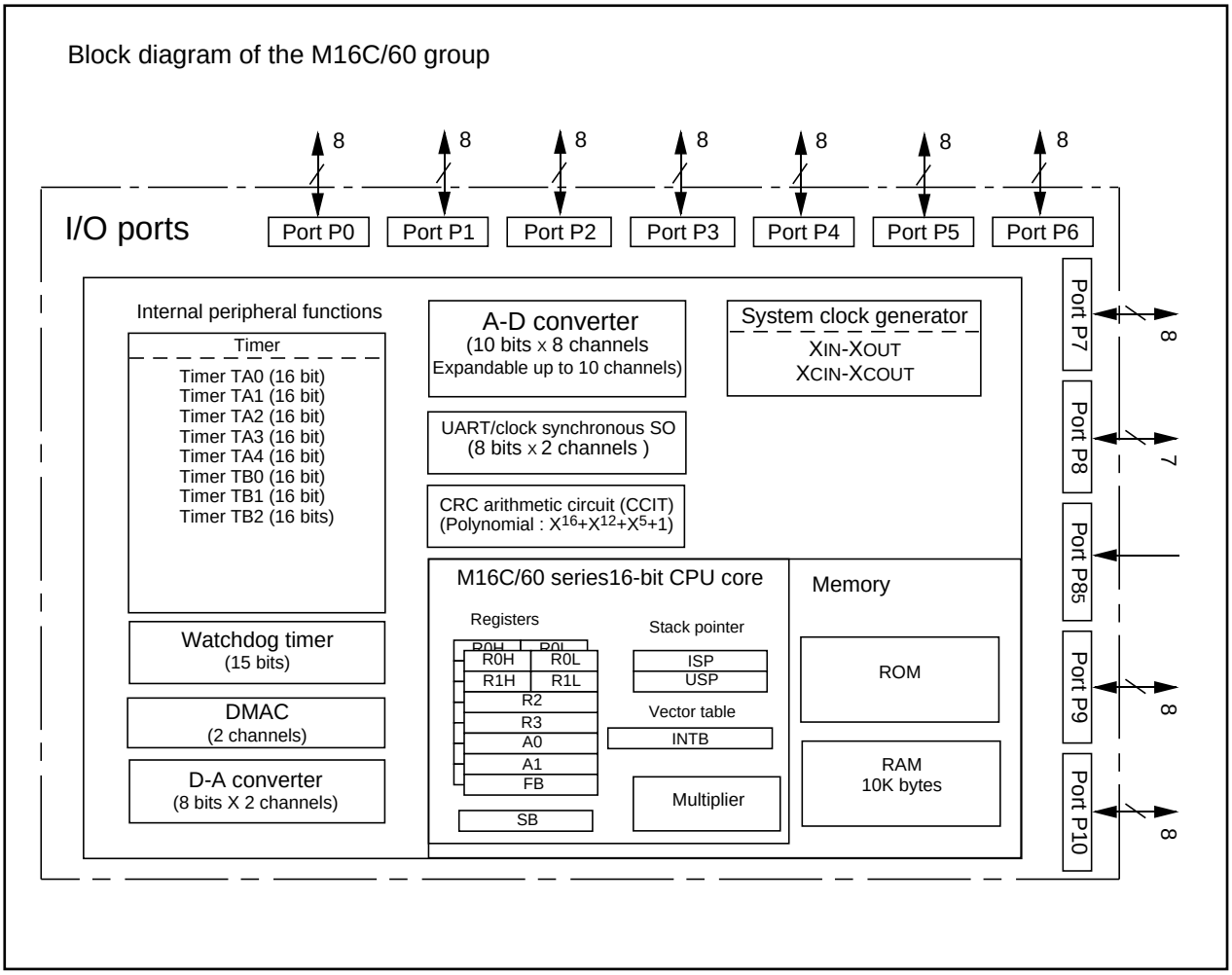


Figure 1.3. Block diagram of M16C/60 group

Performance Outline

Table 1.1 is a performance outline of M16C/60 group.

Table 1.1. Outline performance of M16C/60 group

Item		Performance
Number of basic instructions		91 instructions
Shortest instruction execution time		100ns(f(X_{IN})=10MHz)
Memory capacity	ROM	(See Figure 1.4. ROM expansion)
	RAM	10K bytes
I/O port	P0 to P10 (except $P8_5$)	8 bits X 10, 7 bits X 1
Input port	$P8_5$	1 bit X 1
Multifunction timer	TA0,TA1,TA2,TA3,TA4	16 bits X 5
	TB0,TB1,TB2	16 bits X 3
Serial I/O	UART0,UART1	(UART or clock synchronous) X 2
A-D converter		10 bits X 8 channels (expandable up to 10 channels)
D-A converter		8 bits X 2
DMAC		2 channels (trigger: 15 factors)
CRC calculation circuit		1 circuit (Generator polynomial: $X^{16} + X^{12} + X^5 + 1$)
Watchdog timer		15 bits X 1 (with prescaler)
Interrupt		17 internal and 5 external sources, 4 software sources, 7 levels
Clock generating circuit		2 built-in clock generation circuits (built-in feedback resistor, and external ceramic or quartz oscillator)
Supply voltage		4.0 to 5.5V (f(X_{IN}) = 10MHz) 2.7 to 5.5V(f(X_{IN}) = 7MHz with software one-wait)
Power consumption		18mW (f(X_{IN}) = 7MHz with software one-wait,Vcc=3V)
I/O characteristics	I/O withstand voltage	5V
	Output current	5mA
Memory expansion		Available (to a maximum of 1M bytes)
Operating ambient temperature		– 40 to 85^oC
Device configuration		CMOS silicon gate
Package		100-pin plastic mold QFP

Mitsubishi plans to release the following products in the M16C/60 group:

(1) Support for mask ROM version, external ROM version, one-time PROM version, and EPROM version

(2) ROM capacity

(3) Package

100P6S-A : Plastic molded QFP (mask ROM version and one-time PROM version)

100P6D-A/100P6Q-A : Plastic molded QFP (mask ROM version and one-time PROM version)

100D0 : Ceramic LCC (EPROM version)

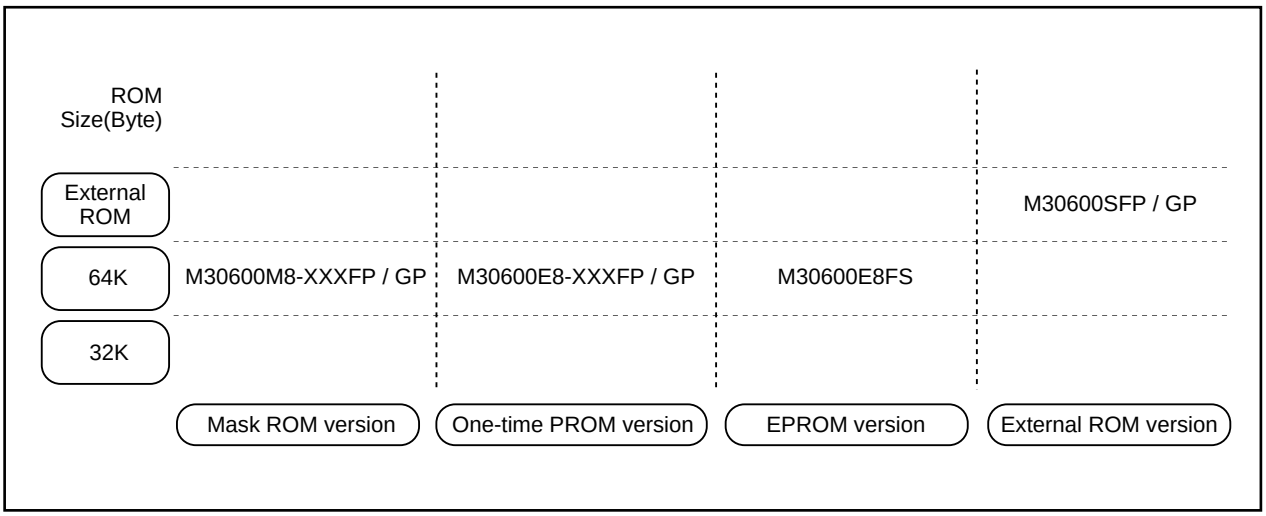


Figure 1.4. ROM expansion

The M16C/60 group products currently supported are listed in Table 1.2.

Table 1.2. M16C/60 group

Feb. 1997

	ROM Capacity	RAM Capacity	Package Type	Remarks
M30600M8-XXXFP	64K bytes	10K bytes	100P6S-A	Mask ROM version
M30600M8-XXXGP	64K bytes	10K bytes	100P6D-A/100P6Q-A	Mask ROM version
M30600E8-XXXFP	64K bytes	10K bytes	100P6S-A	One-time PROM version
M30600E8-XXXGP	64K bytes	10K bytes	100P6D-A/100P6Q-A	One-time PROM version
M30600E8FP	64K bytes	10K bytes	100P6S-A	One-time PROM version (blank)
M30600E8GP	64K bytes	10K bytes	100P6D-A/100P6Q-A	One-time PROM version (blank)
M30600E8FS	64K bytes	10K bytes	100D0	EPROM version
M30600SFP	–	10K bytes	100P6S-A	External ROM version
M30600SGP	–	10K bytes	100P6D-A/100P6Q-A	External ROM version

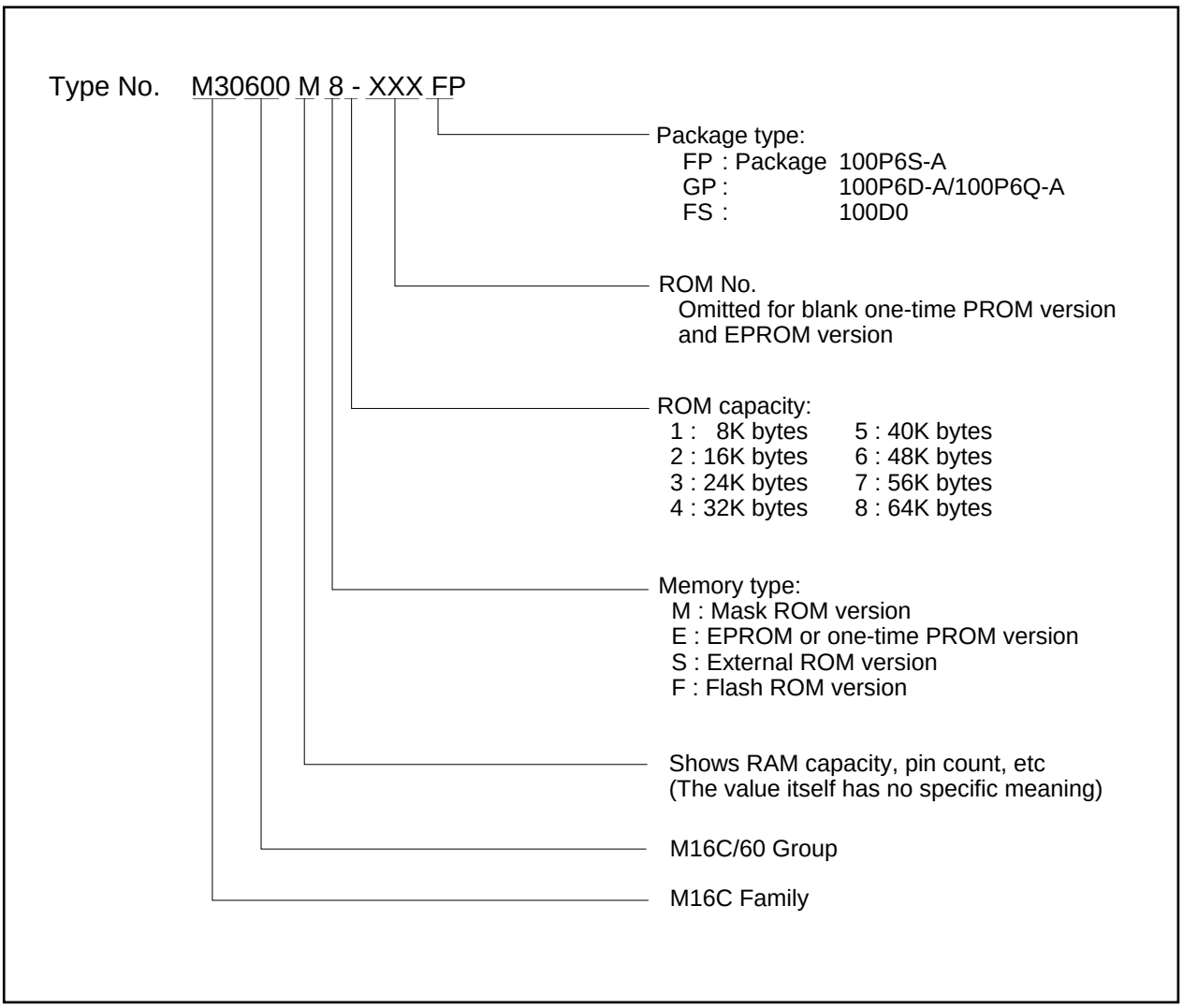


Figure 1.5. Type No., memory size, and package

Pin Description

Pin name	Signal name	I/O type	Function
V_{CC}, V_{SS}	Power supply input		Supply 2.7 to 5.5 V to the V_{CC} pin. Supply 0 V to the V_{SS} pin.
CNV$_{SS}$	CNV$_{SS}$	Input	This pin switches between processor modes. Connect it to the V_{SS} pin when operating in single-chip or memory expansion mode. Connect it to the V_{CC} pin when in microprocessor mode.
$\overline{\text{RESET}}$	Reset input	Input	A "L" on this input resets the microcomputer.
X_{IN} X_{OUT}	Clock input Clock output	Input Output	These pins are provided for the main clock generating circuit. Connect a ceramic resonator or crystal between the X_{IN} and the X_{OUT} pins. To use an externally derived clock, input it to the X_{IN} pin and leave the X_{OUT} pin open.
BYTE	External data bus width select input	Input	This pin selects the width of an external data bus. A 16-bit width is selected when this input is "L"; an 8-bit width is selected when this input is "H". This input must be fixed to either "H" or "L". When operating in single-chip mode, connect this pin to V_{SS}.
AV$_{CC}$	Analog power supply input		This pin is a power supply input for the A-D converter. Connect this pin to V_{CC}.
AV$_{SS}$	Analog power supply input		This pin is a power supply input for the A-D converter. Connect this pin to V_{SS}.
V_{REF}	Reference voltage input	Input	This pin is a reference voltage input for the A-D converter.
$P0_0$ to $P0_7$	I/O port P0	Input/output	This is an 8-bit CMOS I/O port. It has an input/output port direction register that allows the user to set each pin for input or output individually. When 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.
D_0 to D_7		Input/output	When set as a separate bus, these pins input and output data (D_0–D_7).
$P1_0$ to $P1_7$	I/O port P1	Input/output	This is an 8-bit I/O port equivalent to P0.
D_8 to D_{15}		Input/output	When set as a separate bus, these pins input and output data (D_8–D_{15}).
$P2_0$ to $P2_7$	I/O port P2	Input/output	This is an 8-bit I/O port equivalent to P0.
A_0 to A_7		Output	These pins output 8 low-order address bits (A_0–A_7).
A_0/D_0 to A_7/D_7		Input/output	If the external bus is set as an 8-bit wide multiplexed bus, these pins input and output data (D_0–D_7) and output 8 low-order address bits (A_0–A_7) separated in time by multiplexing.
A_0, A_1/D_0 to A_7/D_6		Output Input/output	If the external bus is set as a 16-bit wide multiplexed bus, these pins input and output data (D_0–D_6) and output address (A_1–A_7) separated in time by multiplexing. They also output address (A_0).
$P3_0$ to $P3_7$	I/O port P3	Input/output	This is an 8-bit I/O port equivalent to P0.
A_8 to A_{15}		Output	These pins output 8 middle-order address bits (A_8–A_{15}).
A_8/D_7, A_9 to A_{15}		Input/output Output	If the external bus is set as a 16-bit wide multiplexed bus, these pins input and output data (D_7) and output address (A_8) separated in time by multiplexing. They also output address (A_9–A_{15}).
$P4_0$ to $P4_7$	I/O port P4	Input/output	This is an 8-bit I/O port equivalent to P0.
$\overline{CS_0}$ to $\overline{CS_3}$, A_{16} to A_{19}		Output Output	These pins output $\overline{CS_0}$–$\overline{CS_3}$ signals and A_{16}–A_{19}. $\overline{CS_0}$–$\overline{CS_3}$ are chip select signals used to specify an access space. A_{16}–A_{19} are 4 high-order address bits.

Pin Description

Pin name	Signal name	I/O type	Function
P5$_0$ to P5$_7$	I/O port P5	Input/output	This is an 8-bit I/O port equivalent to P0. In single-chip mode, P5$_7$ in this port outputs a divide-by-8 or divide-by-32 clock of X_{IN} or a clock of the same frequency as X_{CIN} as selected by software.
$\overline{WRL}$ / $\overline{WR}$, $\overline{WRH}$ / BHE, $\overline{RD}$, BCLK, $\overline{HLDA}$, $\overline{HOLD}$, ALE, $\overline{RDY}$		Output Output Output Output Output Input Output Input	Output $\overline{WRL}$, $\overline{WRH}$ ($\overline{WR}$ and $\overline{BHE}$), $\overline{RD}$, BCLK, $\overline{HLDA}$, and ALE signals. $\overline{WRL}$ and $\overline{WRH}$, and $\overline{BHE}$ and $\overline{WR}$ can be switched using software control. ■ $\overline{WRL}$, $\overline{WRH}$, and $\overline{RD}$ selected With a 16-bit external data bus, data is written to even addresses when the $\overline{WRL}$ signal is “L” and to the odd addresses when the $\overline{WRH}$ signal is “L”. Data is read when $\overline{RD}$ is “L”. ■ $\overline{WR}$, $\overline{BHE}$, and $\overline{RD}$ selected Data is written when $\overline{WR}$ is “L”. Data is read when $\overline{RD}$ is “L”. Odd addresses are accessed when $\overline{BHE}$ is “L”. Use this mode when using an 8-bit external data bus. While the input level at the $\overline{HOLD}$ pin is “L”, the microcomputer is placed in the hold state. While in the hold state, $\overline{HLDA}$ outputs a “L” level. ALE is used to latch the address. While the input level of the $\overline{RDY}$ pin is “L”, the microcomputer is in the ready state. BCLK outputs a clock with the same cycle as the internal clock ϕ.
P6$_0$ to P6$_7$	I/O port P6	Input/output	This is an 8-bit I/O port equivalent to P0. Pins in this port also function as UART0 and UART1 I/O pins as selected by software.
P7$_0$ to P7$_7$	I/O port P7	Input/output	This is an 8-bit I/O port equivalent to P0. Pins in this port also function as timer A0–A3 I/O pins as selected by software.
P8$_0$ to P8$_4$, P8$_6$, P8$_7$, P8$_5$	I/O port P8 I/O port P8$_5$	Input/output Input/output Input/output Input	P8$_0$ to P8$_4$, P8$_6$, and P8$_7$ are I/O ports with the same functions as P0. Using software, they can be made to function as the I/O pins for timer A4 and the input pins for external interrupts. P8$_6$ and P8$_7$ can be set using software to function as the I/O pins for a sub clock generation circuit. In this case, connect a quartz oscillator between P8$_6$ (X_{COUT} pin) and P8$_7$ (X_{CIN} pin). P8$_5$ is an input-only port that also functions for $\overline{NMI}$. The $\overline{NMI}$ interrupt is generated when the input at this pin changes from “H” to “L”. The $\overline{NMI}$ function cannot be canceled using software. The pull-up cannot be set for this pin.
P9$_0$ to P9$_7$	I/O port P9	Input/output	This is an 8-bit I/O port equivalent to P0. Pins in this port also function as Timer B0–B2 input pins, D-A converter output pins, A-D converter’s extended input pins, or A-D trigger input pins as selected by software.
P10$_0$ to P10$_7$	I/O port P10	Input/output	This is an 8-bit I/O port equivalent to P0. Pins in this port also function as A-D converter input pins. Furthermore, P10$_4$–P10$_7$ also function as input pins for the key input interrupt function.

Operation of Functional Blocks

The M16C/60 group accommodates certain units in a single chip. These units include ROM and RAM to store instructions and data and the central processing unit (CPU) to execute arithmetic/logic operations. Also included are peripheral units such as timers, serial I/O, D-A converter, DMAC, CRC calculation circuit, A-D converter, and I/O ports.

The following explains each unit.

Memory

Figure 1.6 is a memory map of the M16C/60 group. The address space extends the 1M bytes from address 00000_{16} to $FFFFF_{16}$. From $FFFFF_{16}$ down is ROM. (In the M30600M8-XXXFP, there is 64K bytes of internal ROM from $F0000_{16}$ to $FFFFF_{16}$.) The vector table for fixed interrupts such as the reset and $\overline{NMI}$ are mapped to $FFFDC_{16}$ to $FFFFF_{16}$. The starting address of the interrupt routine is stored here. The address of the vector table for timer interrupts, etc., can be set as desired using the internal register (INTB). See the section on interrupts for details.

10K bytes of internal RAM is mapped to the space from 00400_{16} to $02BFF_{16}$. In addition to storing data, the RAM also stores the stack used when calling subroutines and when interrupts are generated.

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

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

In memory expansion mode, the spaces between $02C00_{16}$ and $03FFF_{16}$, and between $D0000_{16}$ and $EFFFF_{16}$ are reserved and cannot be used. Likewise, the space between $02C00_{16}$ and $03FFF_{16}$ is reserved when in microprocessor mode.

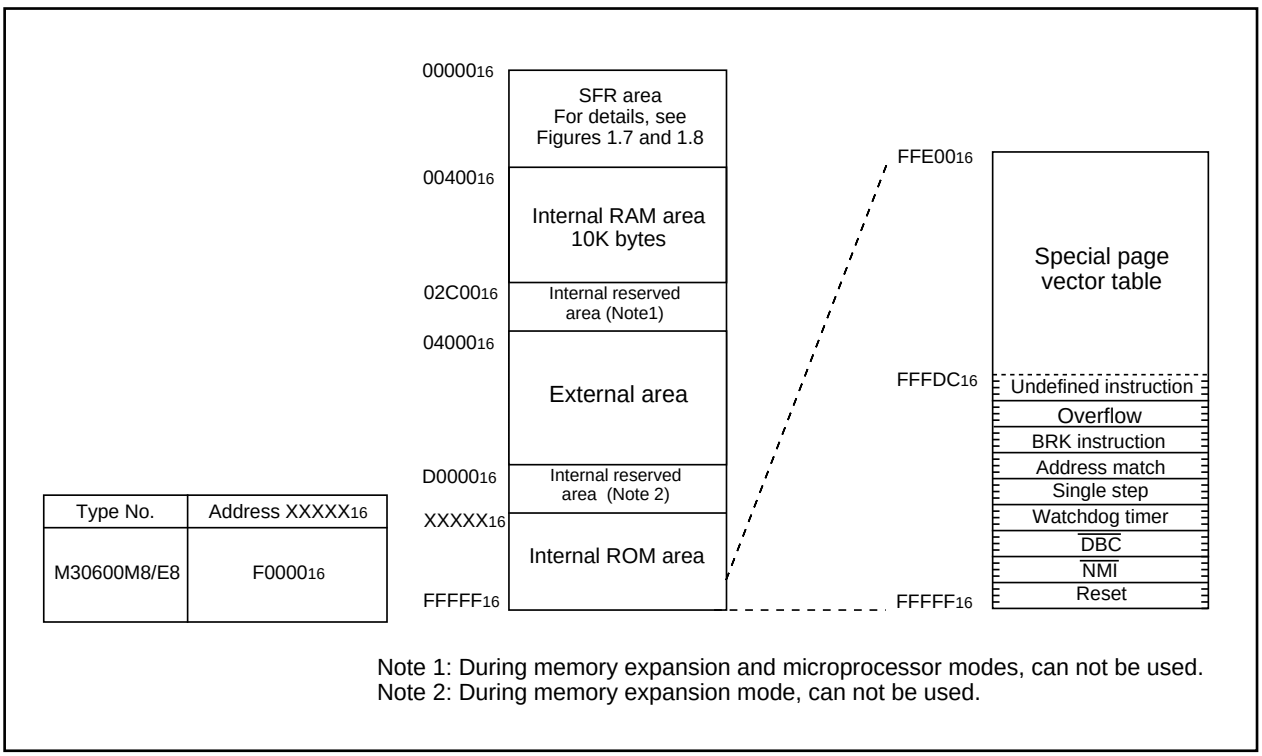


Figure 1.6. Memory map

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

0040₁₆	
0041₁₆	
0042₁₆	
0043₁₆	
0044₁₆	
0045₁₆	
0046₁₆	
0047₁₆	
0048₁₆	
0049₁₆	
004A₁₆	
004B₁₆	DMA0 interrupt control register (DM0IC)
004C₁₆	DMA1 interrupt control register (DM1IC)
004D₁₆	Key input interrupt control register (KUPIC)
004E₁₆	A-D conversion interrupt control register (ADIC)
004F₁₆	
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 A0 interrupt control register (TA0IC)
0056₁₆	Timer A1 interrupt control register (TA1IC)
0057₁₆	Timer A2 interrupt control register (TA2IC)
0058₁₆	Timer A3 interrupt control register (TA3IC)
0059₁₆	Timer A4 interrupt control register (TA4IC)
005A₁₆	Timer B0 interrupt control register (TB0IC)
005B₁₆	Timer B1 interrupt control register (TB1IC)
005C₁₆	Timer B2 interrupt control register (TB2IC)
005D₁₆	INT0 interrupt control register (INT0IC)
005E₁₆	INT1 interrupt control register (INT1IC)
005F₁₆	INT2 interrupt control register (INT2IC)

Figure 1.7. Location of peripheral unit control registers

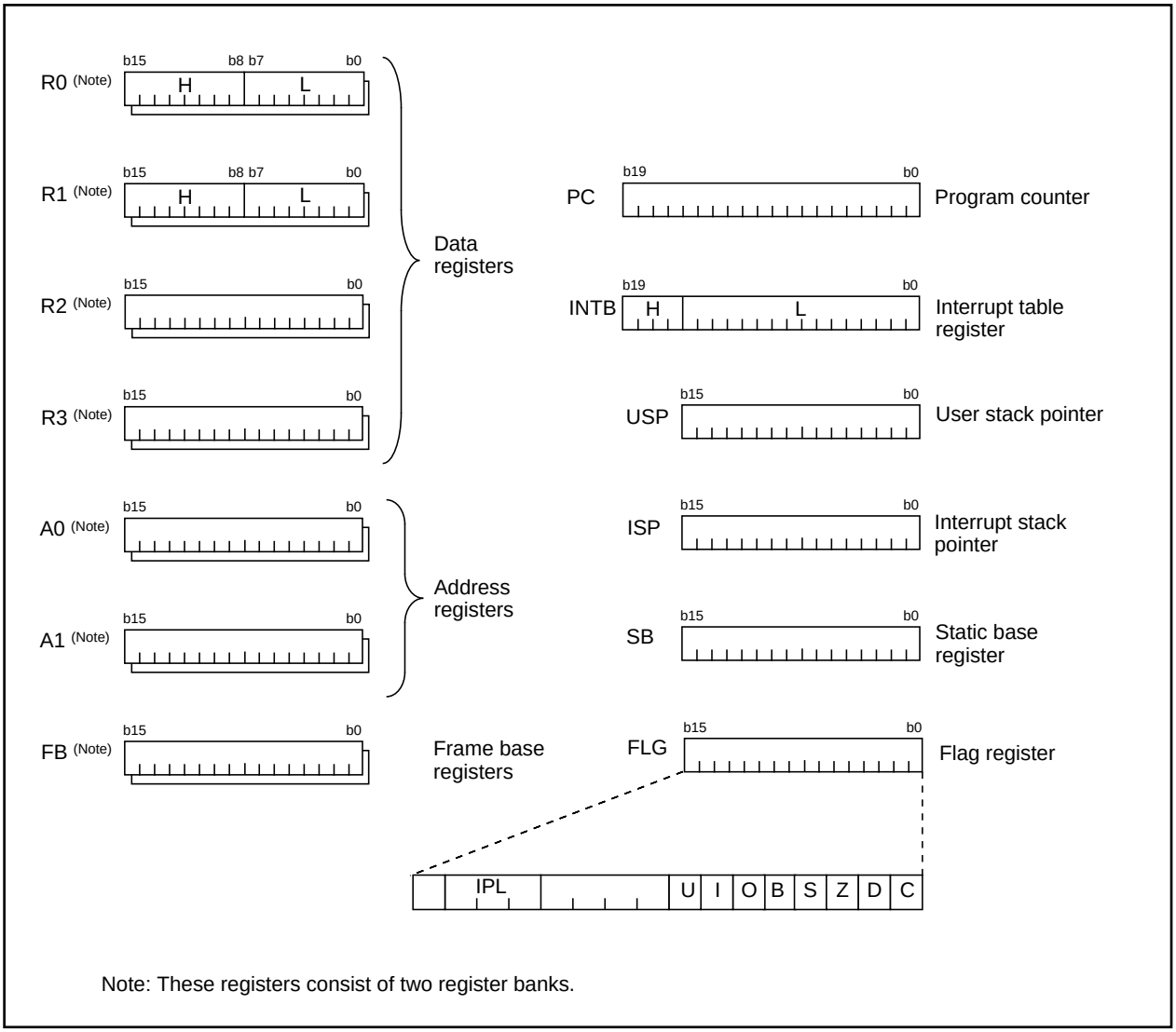
038016	Count start flag (TABSR)
038116	Clock prescaler reset flag (CPSRF)
038216	One-shot start flag (ONSF)
038316	Trigger select register (TRGSR)
038416	Up-down flag (UDF)
038516	
038616	Timer A0 (TA0)
038716	
038816	Timer A1 (TA1)
038916	
038A16	Timer A2 (TA2)
038B16	
038C16	Timer A3 (TA3)
038D16	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

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

Figure 1.8. Location of peripheral unit control registers

Central Processing Unit (CPU)

The CPU has a total of 13 registers shown in Figure 1.9. Seven of these registers (R0, R1, R2, R3, A0, A1, and FB) come in two sets; therefore, these have two register banks.



Note: These registers consist of two register banks.

Figure 1.9. Configuration of 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 be used 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 1.10 shows a configuration of the flag register (FLG). The following explains the function of each flag:

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

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

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

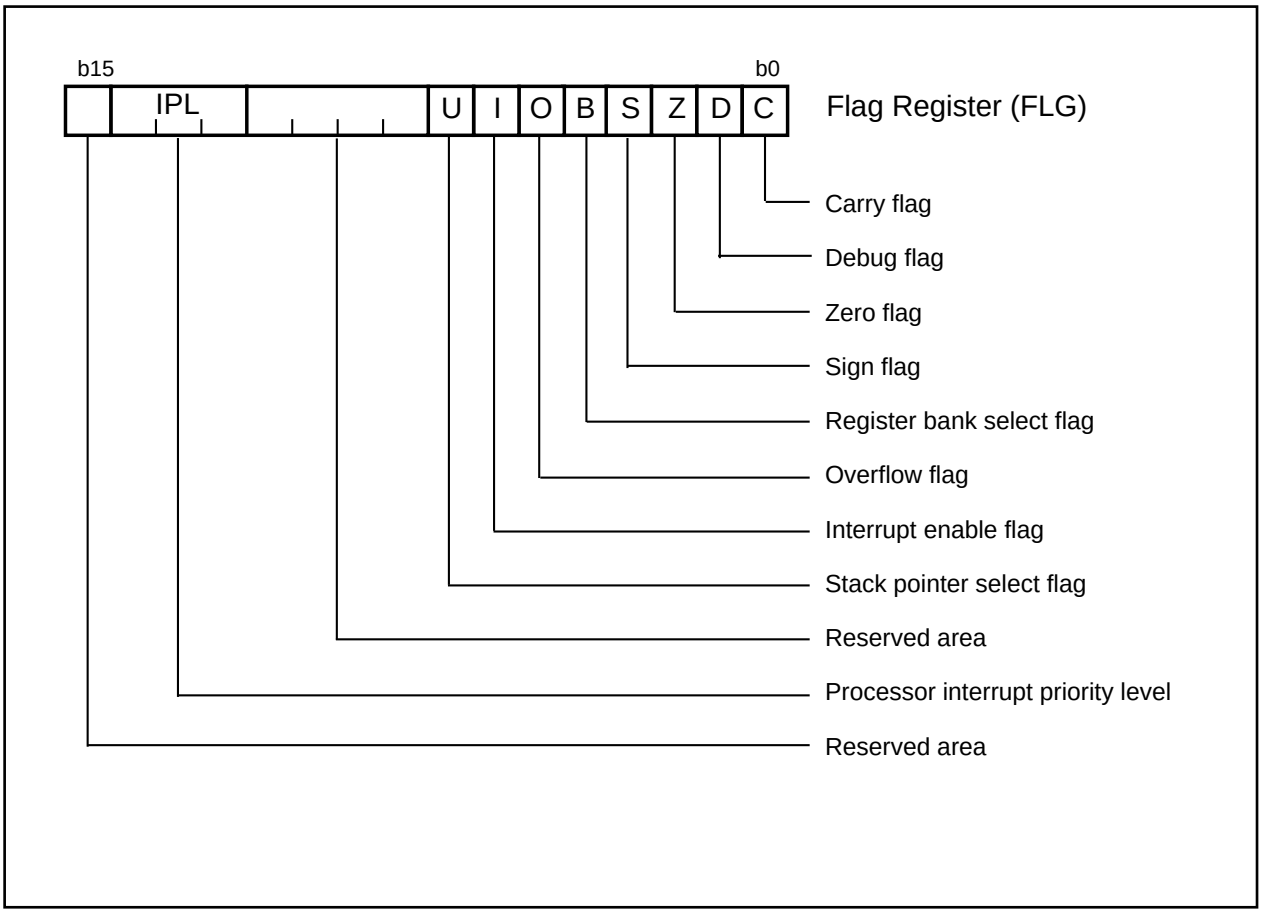


Figure 1.10. Configuration of flag register (FLG)

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” (0.2 V_{CC} max.) for at least 2µs. When the reset pin level is then returned to the “H” level while main clock is stable, the reset status is cancelled and program execution resumes from the address in the reset vector table.

Figure 1.11 shows an example reset circuit. Figure 1.12 shows the reset sequence.

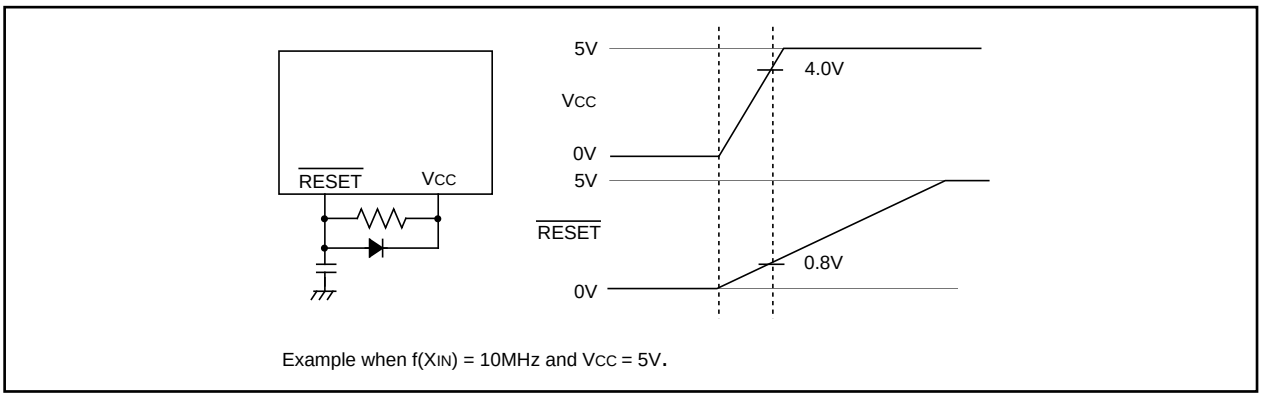


Figure 1.11. Example reset circuit

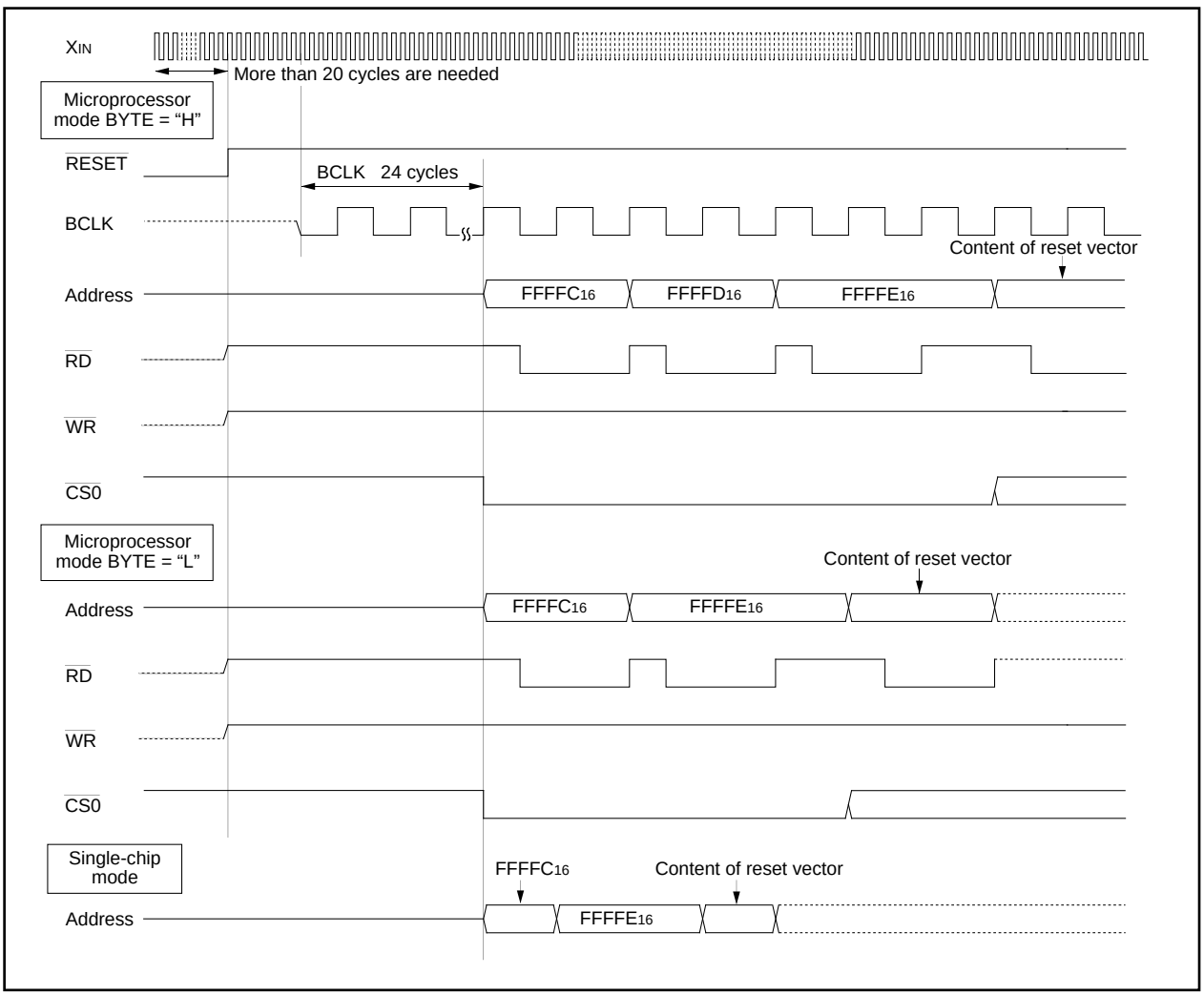


Figure 1.12. Reset sequence

Table 1.3 shows the statuses of the other pins while the $\overline{\text{RESET}}$ pin level is “L”. Figure 1.13 shows the internal status of the microcomputer immediately after the reset is cancelled.

Table 1.3. Pin status when $\overline{\text{RESET}}$ pin level is “L”

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

(1) Processor mode register 0 (Note)	(0004$_{16}$)···	00$_{16}$
(2) Processor mode register 1	(0005$_{16}$)···	0 x x x x x x 0
(3) System clock control register 0	(0006$_{16}$)···	0 1 0 0 1 0 0 0
(4) System clock control register 1	(0007$_{16}$)···	0 0 1 0 0 0 0 0
(5) Chip select control register	(0008$_{16}$)···	0 0 0 0 0 0 0 1
(6) Address match interrupt enable register	(0009$_{16}$)···	x x x x x x 0 0
(7) Protect register	(000A$_{16}$)···	x x x x x 0 0 0
(8) Watchdog timer control register	(000F$_{16}$)···	0 0 0 ? ? ? ? ?
(9) Address match interrupt register 0	(0010$_{16}$)···	00$_{16}$
	(0011$_{16}$)···	00$_{16}$
	(0012$_{16}$)···	x x x x 0 0 0 0
(10) Address match interrupt register 1	(0014$_{16}$)···	00$_{16}$
	(0015$_{16}$)···	00$_{16}$
	(0016$_{16}$)···	x x x x 0 0 0 0
(11) DMA0 control register	(002C$_{16}$)···	0 0 0 0 0 ? 0 0
(12) DMA1 control register	(003C$_{16}$)···	0 0 0 0 0 ? 0 0
(13) DMA0 interrupt control register	(004B$_{16}$)···	x x x x ? 0 0 0
(14) DMA1 interrupt control register	(004C$_{16}$)···	x x x x ? 0 0 0
(15) Key input interrupt control register	(004D$_{16}$)···	x x x x ? 0 0 0
(16) A-D conversion interrupt control register	(004E$_{16}$)···	x x x x ? 0 0 0
(17) UART0 transmit interrupt control register	(0051$_{16}$)···	x x x x ? 0 0 0
(18) UART0 receive interrupt control register	(0052$_{16}$)···	x x x x ? 0 0 0
(19) UART1 transmit interrupt control register	(0053$_{16}$)···	x x x x ? 0 0 0
(20) UART1 receive interrupt control register	(0054$_{16}$)···	x x x x ? 0 0 0
(21) Timer A0 interrupt control register	(0055$_{16}$)···	x x x x ? 0 0 0
(22) Timer A1 interrupt control register	(0056$_{16}$)···	x x x x ? 0 0 0
(23) Timer A2 interrupt control register	(0057$_{16}$)···	x x x x ? 0 0 0
(24) Timer A3 interrupt control register	(0058$_{16}$)···	x x x x ? 0 0 0
(25) Timer A4 interrupt control register	(0059$_{16}$)···	x x x x ? 0 0 0
(26) Timer B0 interrupt control register	(005A$_{16}$)···	x x x x ? 0 0 0
(27) Timer B1 interrupt control register	(005B$_{16}$)···	x x x x ? 0 0 0
(28) Timer B2 interrupt control register	(005C$_{16}$)···	x x x x ? 0 0 0
(29) INT0 interrupt control register	(005D$_{16}$)···	x x 0 0 ? 0 0 0
(30) INT1 interrupt control register	(005E$_{16}$)···	x x 0 0 ? 0 0 0
(31) INT2 interrupt control register	(005F$_{16}$)···	x x 0 0 ? 0 0 0
(32) Count start flag	(0380$_{16}$)···	00$_{16}$
(33) Clock prescaler reset flag	(0381$_{16}$)···	0 x x x x x x x
(34) One-shot start flag	(0382$_{16}$)···	0 0 x 0 0 0 0 0
(35) Trigger select flag	(0383$_{16}$)···	00$_{16}$
(36) Up-down flag	(0384$_{16}$)···	00$_{16}$
(37) Timer A0 mode register	(0396$_{16}$)···	00$_{16}$
(38) Timer A1 mode register	(0397$_{16}$)···	00$_{16}$
(39) Timer A2 mode register	(0398$_{16}$)···	00$_{16}$
(40) Timer A3 mode register	(0399$_{16}$)···	00$_{16}$
(41) Timer A4 mode register	(039A$_{16}$)···	00$_{16}$
(42) Timer B0 mode register	(039B$_{16}$)···	0 0 ? x 0 0 0 0
(43) Timer B1 mode register	(039C$_{16}$)···	0 0 ? x 0 0 0 0
(44) Timer B2 mode register	(039D$_{16}$)···	0 0 ? x 0 0 0 0
(45) UART0 transmit/receive mode register	(03A0$_{16}$)···	00$_{16}$
(46) UART0 transmit/receive control register 0	(03A4$_{16}$)···	0 0 0 0 1 0 0 0
(47) UART0 transmit/receive control register 1	(03A5$_{16}$)···	0 0 0 0 0 0 1 0
(48) UART1 transmit/receive mode register	(03A8$_{16}$)···	00$_{16}$
(49) UART1 transmit/receive control register 0	(03AC$_{16}$)···	0 0 0 0 1 0 0 0
(50) UART1 transmit/receive control register 1	(03AD$_{16}$)···	0 0 0 0 0 0 1 0
(51) UART transmit/receive control register 2	(03B0$_{16}$)···	x 0 0 0 0 0 0 0
(52) DMA0 cause select register	(03B8$_{16}$)···	00$_{16}$
(53) DMA1 cause select register	(03BA$_{16}$)···	00$_{16}$
(54) A-D control register2	(03D4$_{16}$)···	x x x x x x x 0
(55) A-D control register 0	(03D6$_{16}$)···	0 0 0 0 0 ? ? ?
(56) A-D control register 1	(03D7$_{16}$)···	00$_{16}$
(57) D-A control register	(03DC$_{16}$)···	00$_{16}$
(58) Port P0 direction register	(03E2$_{16}$)···	00$_{16}$
(59) Port P1 direction register	(03E3$_{16}$)···	00$_{16}$
(60) Port P2 direction register	(03E6$_{16}$)···	00$_{16}$
(61) Port P3 direction register	(03E7$_{16}$)···	00$_{16}$
(62) Port P4 direction register	(03EA$_{16}$)···	00$_{16}$
(63) Port P5 direction register	(03EB$_{16}$)···	00$_{16}$
(64) Port P6 direction register	(03EE$_{16}$)···	00$_{16}$
(65) Port P7 direction register	(03EF$_{16}$)···	00$_{16}$
(66) Port P8 direction register	(03F2$_{16}$)···	0 0 x 0 0 0 0 0
(67) Port P9 direction register	(03F3$_{16}$)···	00$_{16}$
(68) Port P10 direction register	(03F6$_{16}$)···	00$_{16}$
(69) Pull-up control register 0	(03FC$_{16}$)···	00$_{16}$
(70) Pull-up control register 1	(03FD$_{16}$)···	00$_{16}$
(71) Pull-up control register 2	(03FE$_{16}$)···	00$_{16}$
(72) Data registers (R0/R1/R2/R3)	···	0000$_{16}$
(73) Address registers (A0/A1)	···	0000$_{16}$
(74) Frame base register (FB)	···	0000$_{16}$
(75) Interrupt table register (INTB)	···	00000$_{16}$
(76) User stack pointer (USP)	···	0000$_{16}$
(77) Interrupt stack pointer (ISP)	···	0000$_{16}$
(78) Static base register (SB)	···	0000$_{16}$
(79) Flag register (FLG)	···	0000$_{16}$

The content of other registers and RAM is undefined when the microcomputer is reset. The initial values must therefore be set.
x : Nothing is mapped to this bit
? : Undefined

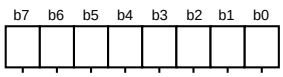
Note : When the V_{CC} level is applied to the CNV_{SS} pin, it is 03$_{16}$ at a reset.

Figure 1.13. 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 1.14 shows a configuration of processor mode register 0 and 1.

Processor mode register 0 (Note 1)

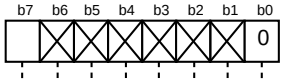


Symbol	Address	When reset
PM0	0004_{16}	00_{16} (Note 2)

Bit symbol	Bit name	Function	R	W
PM00	Processor mode bit	b1 b0 0 0: Single-chip mode 0 1: Memory expansion mode 1 0: Inhibited 1 1: Microprocessor mode	○	○
PM01			○	○
PM02	R/W mode select bit	0 : $\overline{RD}$,$\overline{BHE}$,$\overline{WR}$ 1 : $\overline{RD}$,$\overline{WRH}$,$\overline{WRL}$	○	○
PM03	Software reset bit	The device is reset when this bit is set to "1". The value of this bit is "0" when read.	○	○
PM04	Multiplexed bus space select bit	b5 b4 0 0 : Multiplexed bus is not used 0 1 : Allocated to CS2 space 1 0 : Allocated to CS1 space 1 1 : Allocated to entire space (Note 4)	○	○
PM05			○	○
PM06	Port $P4_0$ to $P4_3$ function select bit (Note 3)	0 : Address output 1 : Port function (Address is not output)	○	○
PM07	BCLK output disable bit	0 : BCLK is output 1 : BCLK is not output (Pin is left floating)	○	○

Note 1: Set bit 1 of the protect register (address $000A_{16}$) to "1" when writing new values to this register.
Note 2: If the V_{CC} voltage is applied to the CNV_{SS}, the value of this register when reset is 03_{16}. (PM00 and PM01 are both set to "1".)
Note 3: Valid in microprocessor and memory expansion modes.
Note 4: In microprocessor mode, multiplexed bus for the entire space cannot be selected. In memory expansion mode, when multiplexed bus for the entire space is selected, address bus range is 256 bytes in each chip select.

Processor mode register 1 (Note)



Symbol	Address	When reset
PM1	0005_{16}	$0XXXXXX0_2$

Bit symbol	Bit name	Function	R	W
Reserved bit		Must always be set to "0"	○	○
Nothing is assigned. These bits can neither be set nor reset. When read, their contents are indeterminate.			—	—
PM17	Wait bit	0 : No wait state 1 : Wait state inserted	○	○

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

Figure 1.14. Configuration of processor mode register 0 and 1

Processor Mode

(1) Types of Processor Mode

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

- **Single-chip mode**

  In single-chip mode, only internal memory space (SFR, internal RAM, and internal ROM) can be accessed. Ports P0 to P10 can be used as programmable I/O ports or as I/O ports for the internal peripheral functions.

- **Memory expansion mode**

  In memory expansion mode, external memory can be accessed in addition to the internal memory space (SFR, internal RAM, and internal ROM).

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

- **Microprocessor mode**

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

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

(2) Setting Processor Modes

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

Regardless of the level of the CNV$_{SS}$ pin, changing the processor mode bits selects the mode. Therefore, never change the processor mode bits when changing the contents of other bits. Also do not attempt to shift to or from the microprocessor mode within the program stored in the internal ROM area.

- **Applying Vss to CNVss pin**

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

- **Applying Vcc to CNVss pin**

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

Figure 1.15 shows the memory maps applicable for each of the modes.

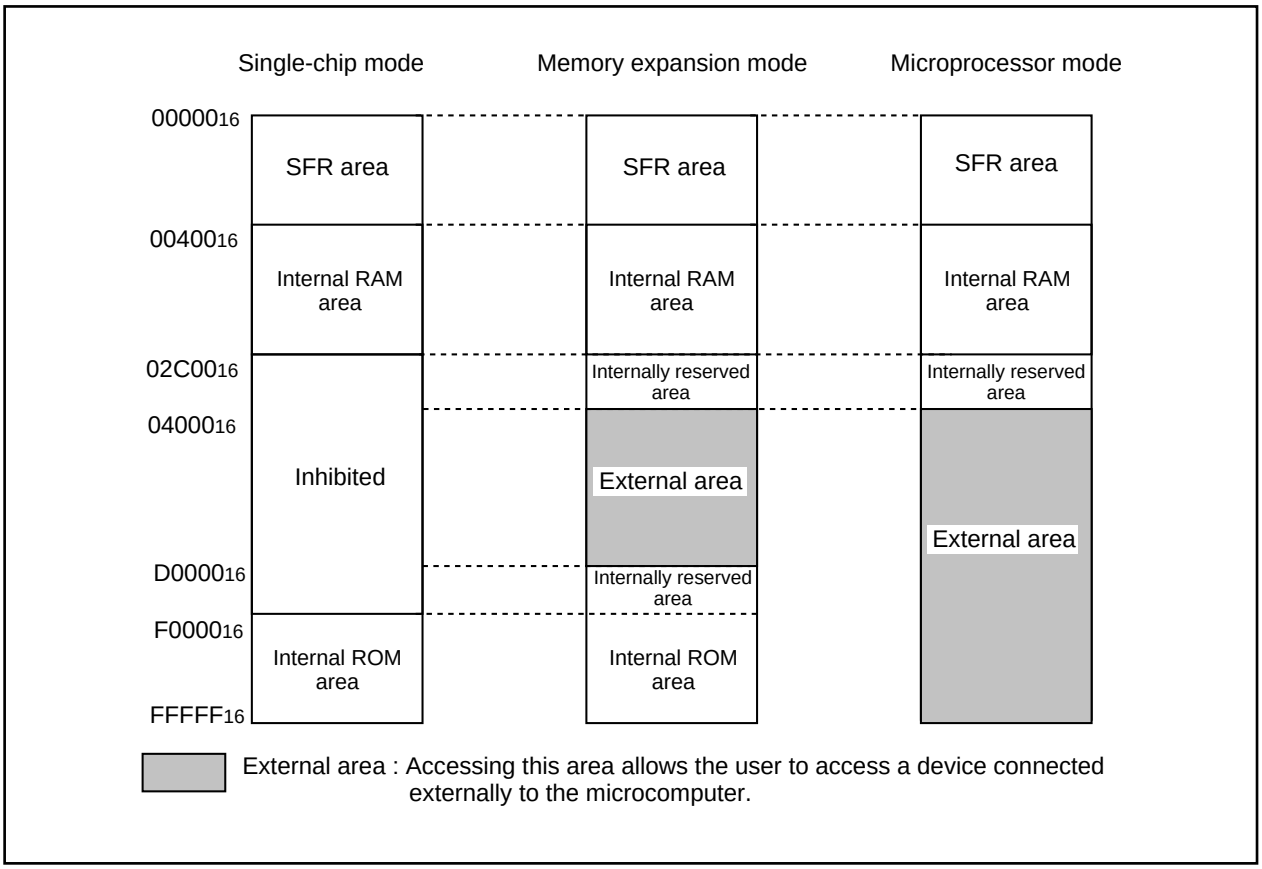


Figure 1.15. Memory maps in each processor mode

Bus Settings

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

Table 1.4 shows the factors used to change the bus settings.

Table 1.4. Factors for switching bus settings

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

(1) Selecting external address bus width

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

(2) Selecting external data bus width

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

(3) Selecting separate/multiplex bus

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

- **Separate bus**

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

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

- **Multiplex bus**

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

  With a 16-bit data bus selected (BYTE pin = “L”), the 8 bits from D_0 to D_7 are multiplexed with A_1 to A_8. D_8 to D_{15} are not multiplexed. In this case, the external devices connected to the multiplexed bus are mapped to the microcomputer’s even addresses (every 2nd address). To access these external devices, access the even addresses as bytes.

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

  Before accessing the multiplex bus, always set the CSi wait bit of the chip select control register to “0”.

Table 1.5. Pin functions for each processor mode

Processor mode	Single-chip mode	Memory expansion mode/microprocessor modes				Memory expansion mode
External bus type		Multiplexed bus and separate bus		separate bus		Multiplexed bus (Note 1)
Multiplexed bus space select bit		"01", "10"		"00"		"11" (Note 2)
Data bus width BYTE pin level		8 bits = "H"	16 bits = "L"	8 bits = "H"	16 bits = "L"	8 bits = "H"
$P0_0$ to $P0_7$	I/O port	Data bus	Data bus	Data bus	Data bus	I/O port
$P1_0$ to $P1_7$	I/O port	I/O port	Data bus	I/O port	Data bus	I/O port
$P2_0$	I/O port	Address bus /data bus (Note 3)	Address bus	Address bus	Address bus	Address bus /data bus
$P2_1$ to $P2_7$	I/O port	Address bus /data bus (Note 3)	Address bus /data bus (Note 3)	Address bus	Address bus	Address bus /data bus
$P3_0$	I/O port	Address bus	Address bus /data bus (Note 3)	Address bus	Address bus	I/O port
$P3_1$ to $P3_7$	I/O port	Address bus	Address bus	Address bus	Address bus	I/O port
$P4_0$ to $P4_3$ Port $P4_0$ to $P4_3$ function select bit = 1	I/O port	I/O port	I/O port	I/O port	I/O port	I/O port
$P4_0$ to $P4_3$ Port $P4_0$ to $P4_3$ function select bit = 0	I/O port	Address bus	Address bus	Address bus	Address bus	I/O port
$P4_4$ to $P4_7$	I/O port	$\overline{CS}$ (chip select) or programmable I/O port (For details, refer to "Bus control")				
$P5_0$ to $P5_3$	I/O port	Outputs $\overline{RD}$, $\overline{WRL}$, $\overline{WRH}$, and BCLK or $\overline{RD}$, $\overline{BHE}$, $\overline{WR}$, and BCLK (For details, refer to "Bus control")				
$P5_4$	I/O port	$\overline{HLDA}$	$\overline{HLDA}$	$\overline{HLDA}$	$\overline{HLDA}$	$\overline{HLDA}$
$P5_5$	I/O port	$\overline{HOLD}$	$\overline{HOLD}$	$\overline{HOLD}$	$\overline{HOLD}$	$\overline{HOLD}$
$P5_6$	I/O port	ALE	ALE	ALE	ALE	ALE
$P5_7$	I/O port	$\overline{RDY}$	$\overline{RDY}$	$\overline{RDY}$	$\overline{RDY}$	$\overline{RDY}$

Note 1: In memory expansion mode, do not select a 16-bit multiplex bus.
Note 2: In microprocessor mode, muitiplexed bus for the entire space cannot be selected.
In memory expansion mode, when muitiplexed bus for the entire space is selected, address bus range is 256 bytes in each chip select.
Note 3: Address bus when in separate bus mode.

Bus Control

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

(1) Address bus/data bus

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

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

Both the address and data bus retain their previous states when internal ROM or RAM is accessed. Also, when a change is made from single-chip mode to memory expansion mode, the value of the address bus is undefined until external memory is accessed.

(2) Chip select signal

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

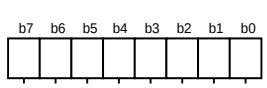
In microprocessor mode, only $\overline{CS0}$ outputs the chip select signal after the reset state has been cancelled. $\overline{CS1}$ to $\overline{CS3}$ function as input ports. Therefore, when using $\overline{CS1}$ to $\overline{CS3}$, external pull-up resistors are required. Figure 1.16 shows the configuration of the chip select control register.

The chip select signal can be used to split the external area into as many as four blocks. Table 1.6 shows the external memory areas specified using the chip select signal.

Table 1.6. External areas specified by the chip select signals

Chip select	Specified address range	
	Memory expansion mode	Microprocessor mode
$\overline{CS0}$	9000_{16} to $CFFFF_{16}$(256K)	9000_{16} to $FFFFF_{16}$ (448K)
$\overline{CS1}$	1000_{16} to $8FFFF_{16}$(512K)	1000_{16} to $8FFFF_{16}$ (512K)
$\overline{CS2}$	0800_{16} to $0FFFF_{16}$ (32K)	0800_{16} to $0FFFF_{16}$ (32K)
$\overline{CS3}$	0400_{16} to $07FFF_{16}$ (16K)	0400_{16} to $07FFF_{16}$ (16K)

Chip select control register



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

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

Figure 1.16. Configuration of chip select control register

(3) Read/write signals

With a 16-bit data bus (BYTE pin ="L"), bit 2 of the processor mode register 0 (address 0004_{16}) select the combinations of $\overline{RD}$, $\overline{BHE}$, and $\overline{WR}$ signals or $\overline{RD}$, $\overline{WRL}$, and $\overline{WRH}$ signals. With an 8-bit data bus (BYTE pin = "H"), use the combination of $\overline{RD}$, $\overline{WR}$, and $\overline{BHE}$ signals. (Set bit 2 of the processor mode register 0 (address 0004_{16}) to "0".) Tables 7 and 8 show the operation of these signals.

After a reset has been cancelled, the combination of $\overline{RD}$, $\overline{WR}$, and $\overline{BHE}$ signals is automatically selected. When switching to the $\overline{RD}$, $\overline{WRL}$, and $\overline{WRH}$ combination, do not write to external memory until bit 2 of the processor mode register 0 (address 0004_{16}) has been set (Note).

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

Table 1.7. Operation of $\overline{RD}$, $\overline{WRL}$, and $\overline{WRH}$ signals

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

Table 1.8. Operation of $\overline{RD}$, $\overline{WR}$, and $\overline{BHE}$ signals

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

(4) ALE signal

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

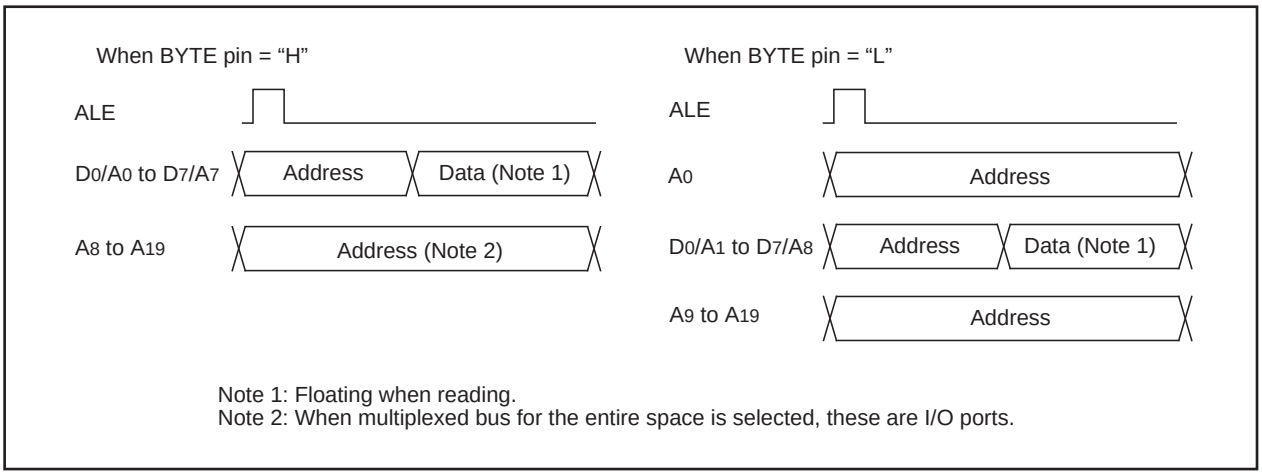


Figure 1.17. ALE signal and address/data bus

(5) Ready signal

The ready signal facilitates access of external devices that require a long time for access. As shown in Figure 1.18, inputting "L" to the $\overline{RDY}$ pin at the falling edge of BCLK causes the microcomputer to enter the ready state. Inputting "H" to the $\overline{RDY}$ pin at the falling edge of BCLK cancels the ready state. Table 1.9 shows the microcomputer status in the ready state. Figure 1.18 shows an example of the $\overline{RD}$ signal being extended using the $\overline{RDY}$ signal.

Ready is valid when accessing the external area during the bus cycle in which the software wait is applied.

Table 1.9. Microcomputer status in ready state (Note)

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

Note: The ready signal cannot be received immediately prior to a software wait.

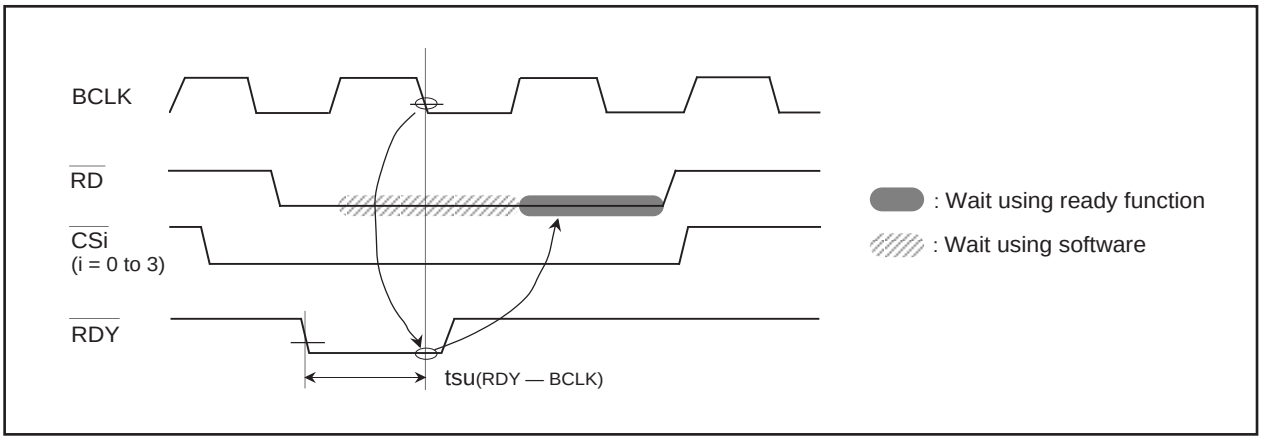


Figure 1.18. Example of $\overline{RD}$ signal extended by $\overline{RDY}$ signal

(6) Hold signal

The hold signal is used to transfer the bus privileges from the CPU to the external circuits. Inputting “L” to the $\overline{\text{HOLD}}$ pin places the microcomputer in the hold state at the end of the current bus access. This status is maintained and “L” is output from the $\overline{\text{HLDA}}$ pin as long as “L” is input to the $\overline{\text{HOLD}}$ pin. Table 1.10 shows the microcomputer status in the hold state.

Table 1.10. Microcomputer status in hold state

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

(7) BCLK output

The output of the internal clock ϕ can be selected using bit 7 of the processor mode register 0 (address 0004_{16}) (Note). The output is floating when bit 7 is set to “1”.

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

(8) Software wait

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

A software wait is inserted in the internal ROM/RAM area and in the external memory area by setting the wait bit of the processor mode register 1. When set to “0”, each bus cycle is executed in one BCLK cycle. When set to “1”, each bus cycle is executed in two or three BCLK cycles. After the microcomputer has been reset, this bit defaults to “0”. When set to “1”, bits 4 to 7 of the chip select control register are invalid and a wait is applied to all external memory areas (two or three BCLK cycles). When V_{CC} is in the range 2.7V to 4.0V, set the wait bit to “1”. However, this is not necessary if the oscillation frequency is less than 3MHz.

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

The SFR area is always accessed in two BCLK cycles regardless of the setting of these control bits. Also, the corresponding bits of the chip select control register must be set to “0” if using the multiplex bus to access the external memory area.

Table 1.11 shows the software wait and bus cycles. Figure 1.19 shows example bus timing when using software waits.

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

Table 1.11. Software waits and bus cycles

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

Note: Always set to “0”.

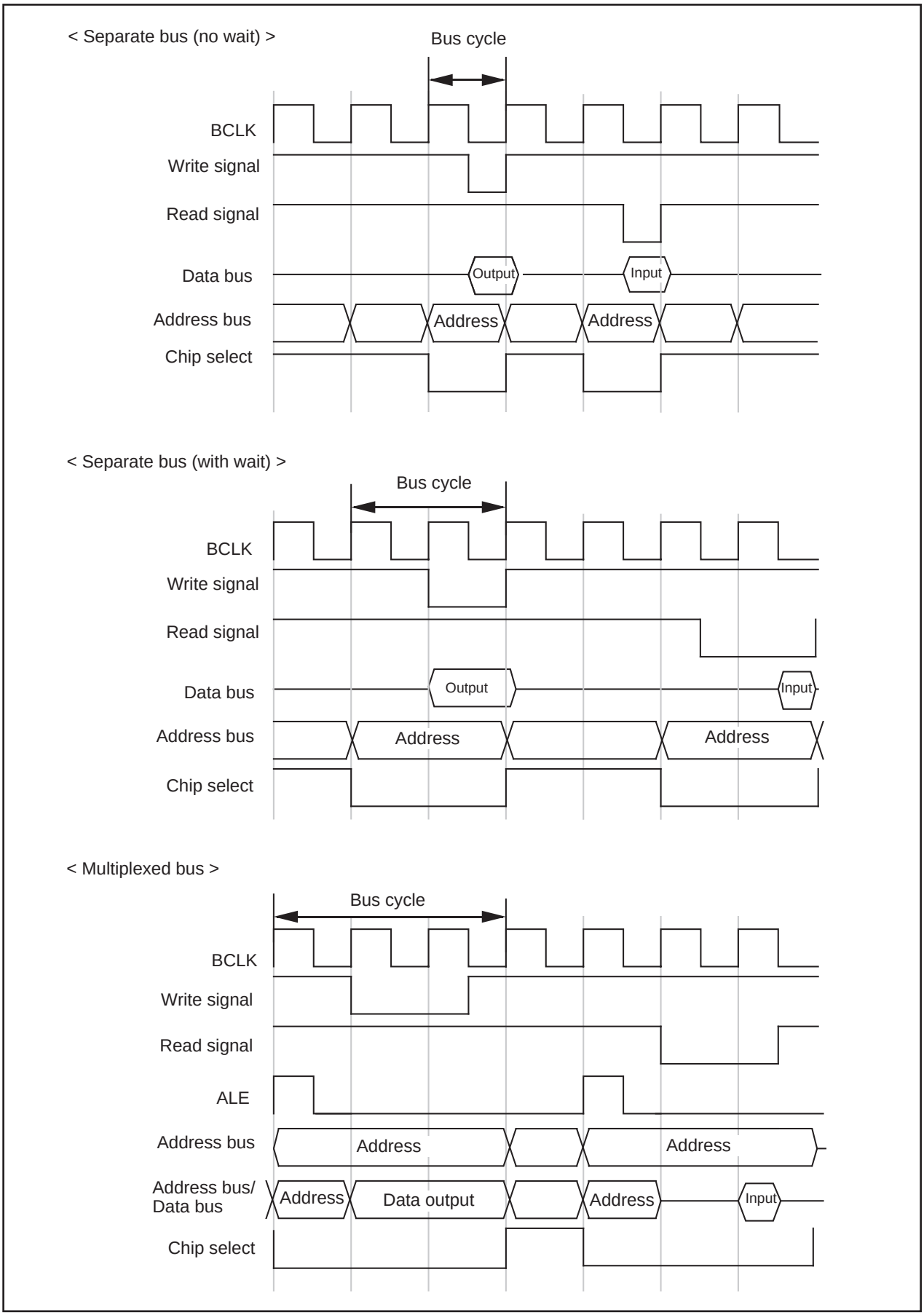


Figure 1.19. Typical bus timings using software wait

Clock Generating Circuit

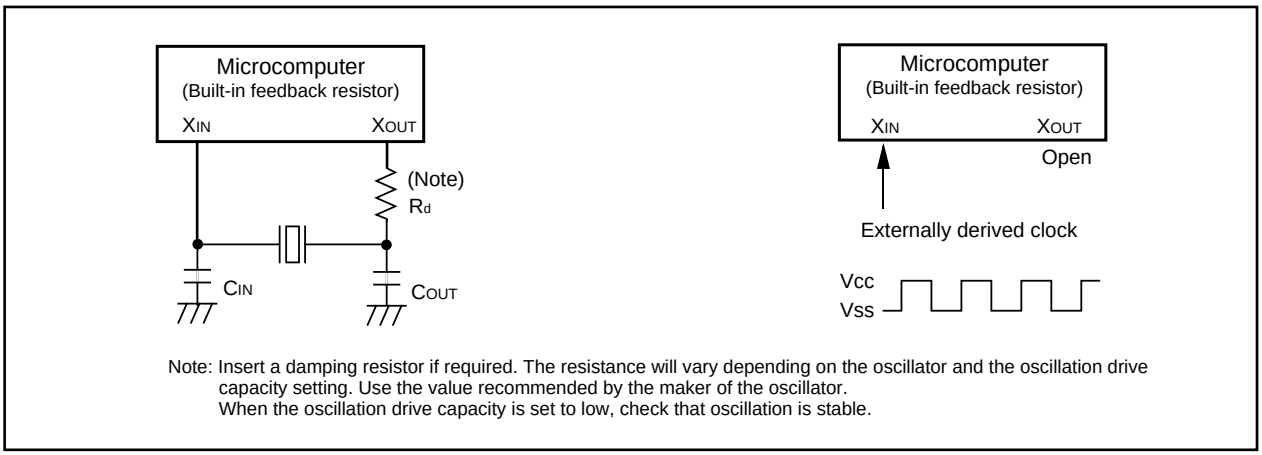
The clock generating circuit contains two oscillator circuits that supply the operating clock sources to the CPU and internal peripheral units.

Table 1.12. Main clock and sub clock generating circuits

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

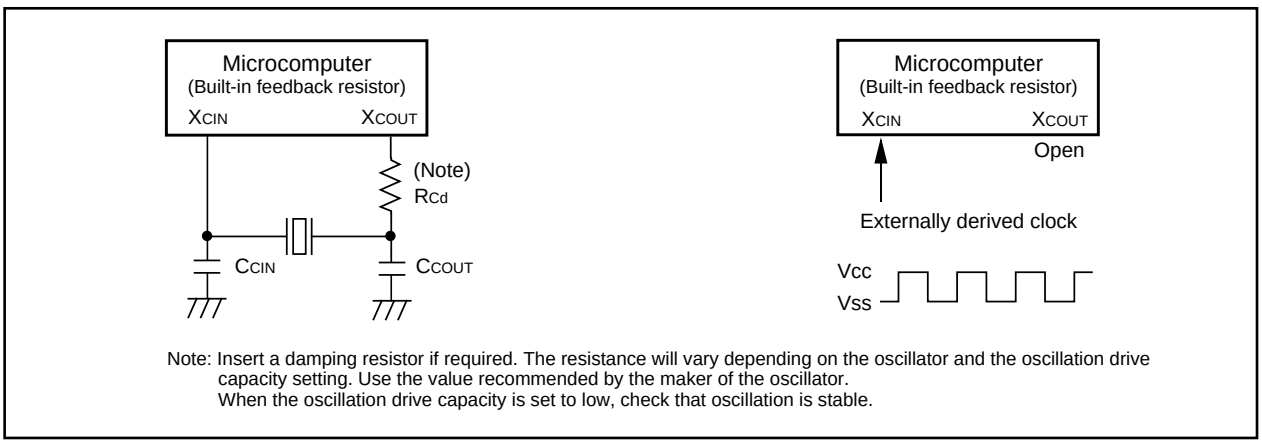
Example of oscillator circuit

Figure 1.20 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 1.21 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 1.20 and 1.21 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.

Figure 1.20. Examples of main clock



Note: Insert a damping resistor if required. The resistance will vary depending on the oscillator and the oscillation drive capacity setting. Use the value recommended by the maker of the oscillator.
When the oscillation drive capacity is set to low, check that oscillation is stable.

Figure 1.21. Examples of sub clock

Clock Control

Figure 1.22 shows a block diagram of the clock generating circuit.

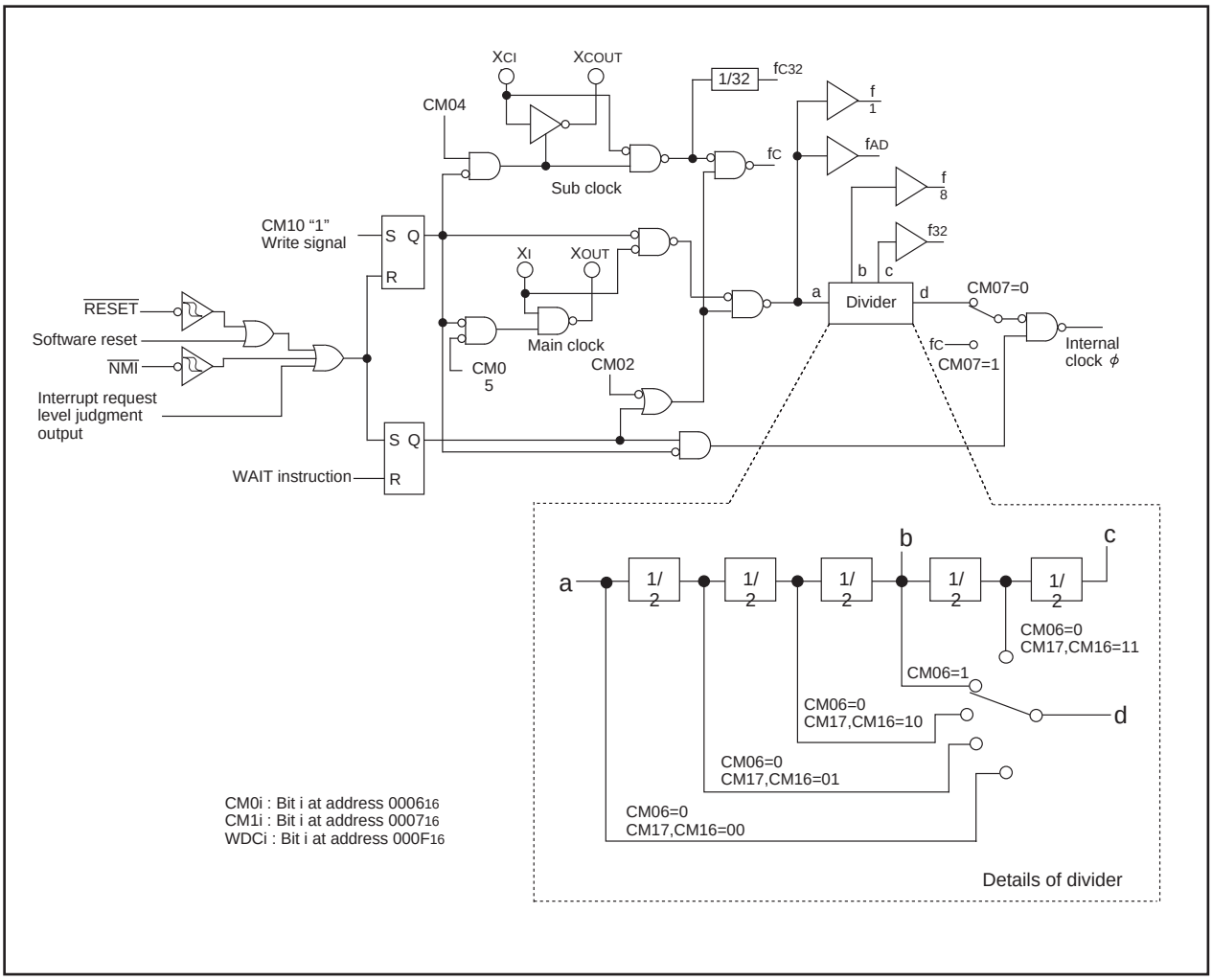


Figure 1.22. Clock generating circuit

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

(1) Main clock

The main clock is generated by the main clock oscillation circuit. After a reset, the clock is divided by 8 to the internal clock ϕ. 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 defaults to "1" when shifting to stop mode and after a reset.

(2) Sub clock

The sub clock is generated by the sub clock oscillation circuit. No sub clock is generated after a reset. After oscillation is started using the port Xc select bit (bit 4 at address 0006_{16}), the sub clock can be selected as the internal clock ϕ 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) Internal clock ϕ

The internal clock ϕ is the clock that drives the CPU, and is either the main clock or fc or is derived by dividing the main clock by 2, 4, 8, or 16. The internal clock ϕ is derived by dividing the main clock by 8 after a reset.

When shiffing to stop mode, the main clock division select bit (bit 6 at 0006_{16}) is set to "1".

(4) Peripheral function clock

- **f_1, f_8, f_{32}**

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 by stopping the main clock or by setting the WAIT peripheral function clock stop bit (bit 2 at 0006_{16}) to "1" and then executing a WAIT instruction.

- **f_{AD}**

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

(5) f_{C32}

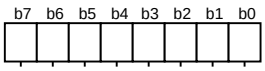
This clock is derived by dividing the sub clock by 32. It is used for the timer A and timer B counts.

(6) f_C

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

Figure 1.23 shows the configuration of system clock control registers 0 and 1.

System clock control register 0 (Note 1)

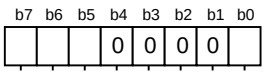


Symbol: CM0 | Address: 0006₁₆ | When reset: 48₁₆

Bit symbol	Bit name	Function	R	W
CM00	Clock output function select bit	b1 b0 0 0 : I/O port P5₇ 0 1 : fC output 1 0 : f8 output 1 1 : f32 output	O	O
CM01			O	O
CM02	WAIT peripheral function clock stop bit	0 : Do not stop f1, f8, f32 in wait mode 1 : Stop f1, f8, f32 in wait mode	O	O
CM03	XCIN-XCOUT drive capacity select bit (Note 2)	0 : LOW 1 : HIGH	O	O
CM04	Port XC select bit	0 : I/O port 1 : XCIN-XCOUT generation	O	O
CM05	Main clock (XIN-XOUT) stop bit (Note 3) (Note 4)	0 : On 1 : Off	O	O
CM06	Main clock division select bit 0 (Note 2)	0 : CM16 and CM17 valid 1 : Division by 8 mode	O	O
CM07	System clock select bit (Note 5)	0 : XIN, XOUT 1 : XCIN, XCOUT	O	O

Note 1: Set bit 0 of the protect register (address 000A16) to "1" before writing to this register.
Note 2: Changes to "1" when shifting to stop mode.
Note 3: When entering power saving mode, main clock stops using this bit. When returning from stop mode and operating with XIN, set this bit to "0". When main clock oscillation is operating by itself, set system clock select bit (CM07) to "1" before setting this bit to "1".
Note 4: When inputting external clock, only clock oscillation buffer is stopped and clock input is acceptable.
Note 5: Set port Xc select bit (CM04) to "1" before writing to this bit. The both bits can not be written at the same time.

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	0 : Clock on 1 : All clocks off (stop mode)	O	O
Reserved bit		Always set to "0"	O	O
Reserved bit		Always set to "0"	O	O
Reserved bit		Always set to "0"	O	O
Reserved bit		Always set to "0"	O	O
CM15	XIN-XOUT drive capacity select bit (Note 2)	0 : LOW 1 : HIGH	O	O
CM16	Main clock division select bit1 (Note 3)	b7 b6 0 0 : No division mode 0 1 : Division by 2 mode 1 0 : Division by 4 mode 1 1 : Division by 16 mode	O	O
CM17				

Note 1: Set bit 0 of the protect register (address 000A16) to "1" before writing to this register.
Note 2: Changes to "1" when shiffing to stop mode.
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.

Figure 1.23. Configuration of system clock control registers 0 and 1

Clock Output

In single-chip mode, the clock output function select bits (bits 0 and 1 at address 0006_{16}) enable f_8, f_{32}, or fc to be output from the $P5_7$/CLK_{OUT} pin. When the WAIT peripheral function clock stop bit (bit 2 at address 0006_{16}) is set to "1", the output of f_8 and f_{32} stops when a WAIT instruction is executed.

Stop Mode

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

Because the oscillation of internal clock ϕ, 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 A and timer B operate provided that the event counter mode is set to an external pulse, and UART0 and UART1 functions provided an external clock is selected. Table 1.13 shows the status of the ports in stop mode.

Stop mode is cancelled by a hardware reset or interrupt. If an interrupt is to be used to cancel stop mode, that interrupt must first have been enabled.

When shifting to stop mode, the main clock division select bit 0 (bit 6 at 0006_{16}) is set to "1".

Table 1.13. Port status during stop mode

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

Wait Mode

When a WAIT instruction is executed, the internal clock ϕ stops and the microcomputer enters the wait mode. In this mode, oscillation continues but the internal clock ϕ 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 1.14 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 internal clock ϕ the clock that had been selected when the WAIT instruction was executed.

Table 1.14. Port status during wait mode

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

Status Transition Of Internal Clock φ

Power dissipation can be reduced and low-voltage operation achieved by changing the count source for internal clock φ. Table 1.15 shows the operating modes corresponding to the settings of system clock control registers 0 and 1.

After a reset, operation defaults to division by 8 mode. When shifting to stop mode, the main clock division select bit 0 (bit 6 at address 0006_{16}) is set to "1". The following shows the operational modes of internal clock φ:

(1) Division by 2 mode

The main clock is divided by 2 to obtain the internal clock φ.

(2) Division by 4 mode

The main clock is divided by 4 to obtain the internal clock φ.

(3) Division by 8 mode

The main clock is divided by 8 to obtain the internal clock φ. Note that oscillation of the main clock must have stabilized before transferring from this mode to another mode.

(4) Division by 16 mode

The main clock is divided by 16 to obtain the internal clock φ.

(5) No-division mode

The main clock is used as internal clock φ.

(6) Low-speed mode

f_C is used as internal clock φ. 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 internal clock φ and the main clock is stopped.

Table 1.15. Operating modes dictated by settings of system clock control registers 0 and 1

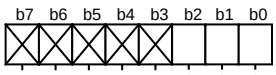
CM17	CM16	CM07	CM06	CM05	CM04	Operating mode of internal clock φ
0	1	0	0	0	Invalid	Division by 2 mode
1	0	0	0	0	Invalid	Division by 4 mode
Invalid	Invalid	0	1	0	Invalid	Division by 8 mode
1	1	0	0	0	Invalid	Division by 16 mode
0	0	0	0	0	Invalid	No-division mode
Invalid	Invalid	1	Invalid	0	1	Low-speed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

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 1.24 shows the configuration of the protect register. The values in the processor mode register 0 (addresss 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 P9 direction register (address $03F3_{16}$) can only be changed when the respective bit in the protect register is set to “1”. Therefore, important outputs can be allocated to port P9.

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

Protect register



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

Bit symbol	Bit name	Function	R	W
PRC0	Enables writing to system clock control registers 0 and 1 (addresses 0006_{16} and 0007_{16})	0 : Write-inhibited 1 : Write-enabled	○	○
PRC1	Enables writing to processor mode registers 0 and 1 (addresses 0004_{16} and 0005_{16})	0 : Write-inhibited 1 : Write-enabled	○	○
PRC2	Enables writing to port P9 direction register (address $03F3_{16}$) (Note)	0 : Write-inhibited 1 : Write-enabled	○	○
Nothing is assigned. These bits can neither be set nor reset. When read, their contents are 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 1.24. Configuration of protect register

Interrupts

Tables 1.16 and 1.17 show the interrupt sources and vector addresses. When an interrupt is received, the program is executed from the address shown by the respective interrupt vector.

The vector addresses for the interrupts in Table 1.16 are fixed (interrupt vector addresses). These interrupts are not affected by the interrupt enable flag (I flag) (non-maskable interrupts).

The vector table addresses for the interrupt in Table 1.17 are variable, being determined as relative to the fixed address in the interrupt table register (INTB) (variable interrupt addresses). These interrupts can be enabled or disabled using the interrupt enable flag (I flag) (maskable interrupts). 64 vectors can be set in the interrupt table register (INTB). Any software interrupt Nos. 0 to 63 can be assigned to each vector. By using the INT instruction to specify a software interrupt No., the program can be executed starting at the address indicated by the respective vector. The BRK instruction interrupt has interrupt vectors in both the fixed vector addresses and variable vector addresses. When the contents of FFFE4$_{16}$ to FFFE7$_{16}$ are all "FF$_{16}$", the program is executed from the address shown in the BRK instruction interrupt vector in the variable vector addresses.

Specify the starting address of the interrupt program in the interrupt vector. Figure 1.25 shows the format for specifying the address.

Table 1.16. Interrupt factors (fixed interrupt vector addresses)

Interrupt source	Vector table addresses Address (L) to address (H)	Remarks
Undefined instruction	FFFDC$_{16}$ to FFFDF$_{16}$	Interrupt on UND instruction
Overflow	FFFE0$_{16}$ to FFFE3$_{16}$	Interrupt on INTO instruction
BRK instruction	FFFE4$_{16}$ to FFFE7$_{16}$	If the vector is filled with FF$_{16}$, program execution starts from the address shown by the vector in the variable vector table
Address match	FFFE8$_{16}$ to FFFEB$_{16}$	There is an address-matching interrupt enable bit
Single step (Note)	FFFEC$_{16}$ to FFFEF$_{16}$	Do not use
Watchdog timer	FFFF0$_{16}$ to FFFF3$_{16}$	
$\overline{\text{DBC}}$ (Note)	FFFF4$_{16}$ to FFFF7$_{16}$	Do not use
$\overline{\text{NMI}}$	FFFF8$_{16}$ to FFFFB$_{16}$	External interrupt by input to $\overline{\text{NMI}}$ pin
Reset	FFFFC$_{16}$ to FFFFF$_{16}$	

Note: Interrupts used for debugging purposes only.

Figure 1.25. Format for specifying interrupt vector addresses

Table 1.17. 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 11	+44 to +47 (Note)	DMA0	
Software interrupt number 12	+48 to +51 (Note)	DMA1	
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 A0	
Software interrupt number 22	+88 to +91 (Note)	Timer A1	
Software interrupt number 23	+92 to +95 (Note)	Timer A2	
Software interrupt number 24	+96 to +99 (Note)	Timer A3	
Software interrupt number 25	+100 to +103 (Note)	Timer A4	
Software interrupt number 26	+104 to +107 (Note)	Timer B0	
Software interrupt number 27	+108 to +111 (Note)	Timer B1	
Software interrupt number 28	+112 to +115 (Note)	Timer B2	
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).

(1) Interrupt control registers

Peripheral I/O interrupts have their own interrupt control registers. Table 1.18 shows the addresses of the interrupt control registers. Figure 1.26 shows the configuration of the interrupt control registers.

The interrupt request bit is set by hardware to “0” when an interrupt request is received. The interrupt request bit can also be set by software to “0”. (Do not set to “1”.)

$\overline{INT0}$, $\overline{INT1}$, and $\overline{INT2}$ are triggered by the edges of external inputs. The edge polarity is selected using the polarity select bit. (Other interrupts are described elsewhere.)

An interrupt must first be enabled before it can be used to cancel stop mode.

Table 1.18. Addresses in interrupt control register

Interrupt control register	Symbol name	Address	Interrupt control register	Symbol name	Address
DMA0 interrupt control register	DM0IC	004B16	DMA1 interrupt control register	DM1IC	004C16
Key input interrupt control register	KUPIC	004D16	A-D interrupt control register	ADIC	004E16
UART0 transmit interrupt control register	S0TIC	005116	UART0 receive interrupt control register	S0RIC	005216
UART1 transmit interrupt control register	S1TIC	005316	UART1 receive interrupt control register	S1RIC	005416
Timer A0 interrupt control register	TA0IC	005516	Timer A1 interrupt control register	TA1IC	005616
Timer A2 interrupt control register	TA2IC	005716	Timer A3 interrupt control register	TA3IC	005816
Timer A4 interrupt control register	TA4IC	005916	Timer B0 interrupt control register	TB0IC	005A16
Timer B1 interrupt control register	TB1IC	005B16	Timer B2 interrupt control register	TB2IC	005C16
$\overline{INT0}$ interrupt control register	INT0IC	005D16	$\overline{INT1}$ interrupt control register	INT1IC	005E16
$\overline{INT2}$ interrupt control register	INT2IC	005F16			

(2) Interrupt priority

The order of priority when two or more interrupts are generated simultaneously is determined by both hardware and software.

The interrupt priority levels determined by hardware are reset > $\overline{NMI}$ > $\overline{DBC}$ > wacthdog timer > peripheral I/O interrupts > single-step > address matching interrupt.

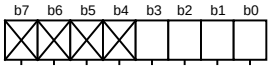
The interrupt priority levels determined by software are as the interrupt priority levels are set in the interrupt control registers.

Figure 1.27 shows the circuit that judges the interrupt priority level. When two or more interrupts are generated simultaneously, this circuit selects the interrupt with the highest priority level. However, if the interrupts have the same priority level, the interrupt is selected according to the priority set in the circuit.

The selected interrupt is accepted only when the priority level is higher than the processor interrupt priority level (IPL) in the flag register (FLG) and the interrupt enable flag (I flag) is “1”. Note that the reset, $\overline{NMI}$, $\overline{DBC}$, watchdog timer, single-step, address-match, BRK instruction, overflow, and undefined instruction interrupts are generated regardless of the interrupt enable flag (I flag).

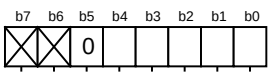
Interrupt control register

Symbol	Address	When reset
DMiIC(i=0,1)	$004B_{16}$, $004C_{16}$	$XXXXX000_2$
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}	$XXXXX000_2$
TAiIC(i=0 to 4)	0055_{16} to 0059_{16}	$XXXXX000_2$
TBiIC(i=0 to 2)	$005A_{16}$ to $005C_{16}$	$XXXXX000_2$



Bit symbol	Bit name	Function	R	W
ILVL0	Interrupt priority level select bit	b2 b1 b0 0 0 0 : Level 0 (interrupt disabled) 0 0 1 : Level 1 0 1 0 : Level 2 0 1 1 : Level 3 1 0 0 : Level 4 1 0 1 : Level 5 1 1 0 : Level 6 1 1 1 : Level 7	○	○
ILVL1			○	○
ILVL2			○	○
IR	Interrupt request bit	0 : Interrupt not requested 1 : Interrupt requested	○	○ (Note)
Nothing is assigned. These bits can neither be set nor reset. When read, their contents are indeterminate.			–	–

Note: This bit can only be accessed for reset (= 0), but cannot be accesd for set (= 1).



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

Bit symbol	Bit name	Function	R	W
ILVL0	Interrupt priority level select bit	b2 b1 b0 0 0 0 : Level 0 (interrupt disabled) 0 0 1 : Level 1 0 1 0 : Level 2 0 1 1 : Level 3 1 0 0 : Level 4 1 0 1 : Level 5 1 1 0 : Level 6 1 1 1 : Level 7	○	○
ILVL1			○	○
ILVL2			○	○
IR	Interrupt request bit	0: Interrupt not requested 1: Interrupt requested	○	○ (Note)
POL	Polarity select bit	0 : Selects falling ede 1 : Selects rising edge	○	○
Reserved bit		Always set to “0”	○	○
Nothing is assigned. These bits can neither be set nor reset. When read, their contents are indeterminate.			–	–

Note: This bit can only be accessed for reset (= 0), but cannot be accesd for set (= 1).

Figure 1.26. Configuration of interrupt control register

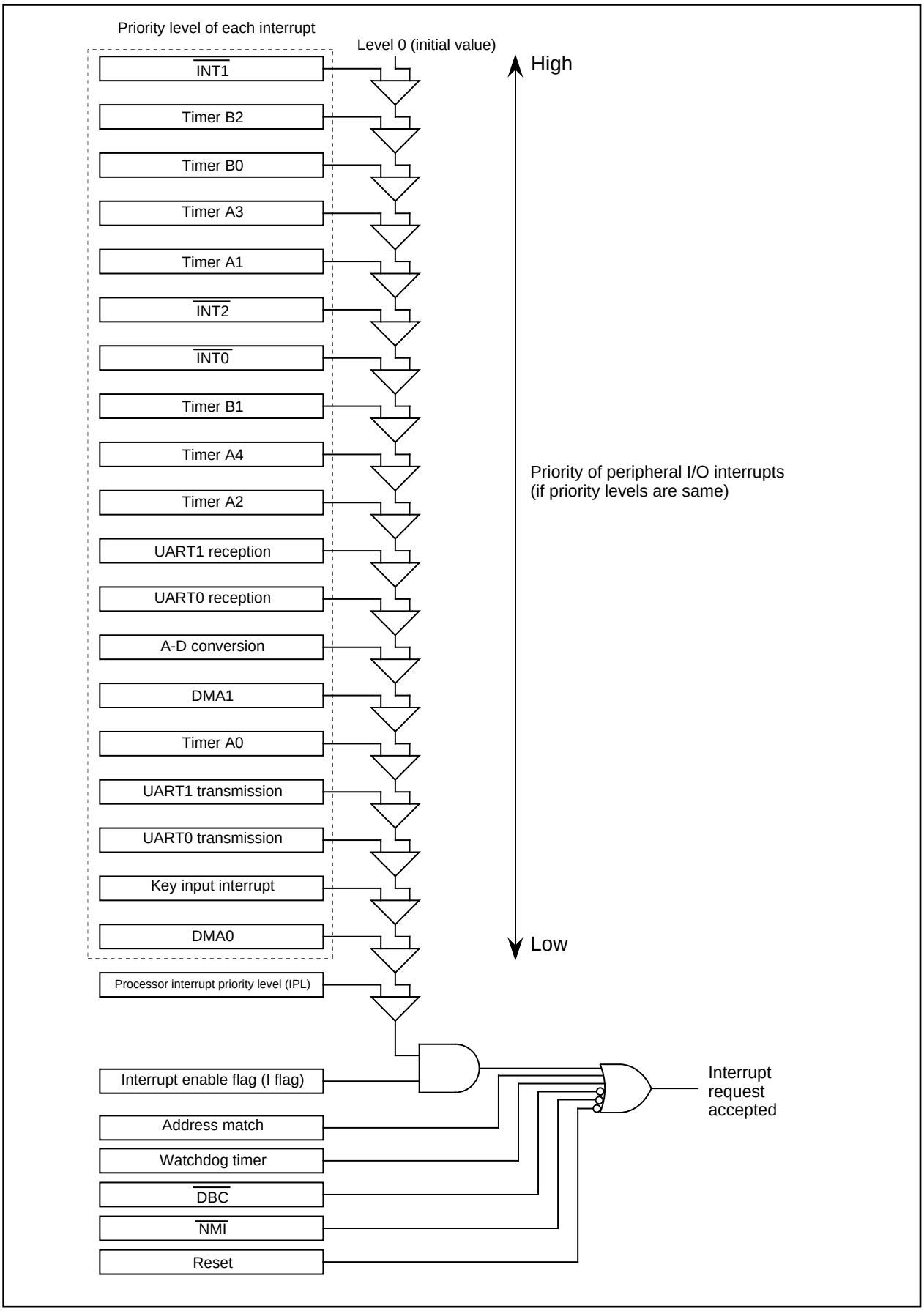


Figure 1.27. Interrupt resolution circuit

(3) Flag changes

When an interrupt request is received, the stack pointer select flag (U flag) changes to “0” and the flag register (FLG) and program counter (PC) are saved to the stack area indicated by the interrupt stack pointer (ISP). Thereafter, the interrupt enable flag (I flag) and debug flag (D flag) change to “0” and the processor interrupt priority level (IPL) at the flag register (FLG) is replaced by the priority level of the received interrupt. However, when interrupt requests are received for software interrupt Nos. 32 to 63, the flag register (FLG) and program counter (PC) are saved to the stack shown by the stack pointer select flag (U flag) at the time the interrupt was received. The stack pointer select flag (U flag) does not change. The value of the processor interrupt priority level (IPL) in the flag register (FLG) differs in the case of reset, $\overline{\text{NMI}}$, $\overline{\text{DBC}}$, watchdog timer, single-step, address-match, BRK instruction, overflow, and undefined instruction interrupts. Table 1.19 shows how the IPL changes when interrupt requests are received.

Table 1.19. Change of IPL state when interrupt requests are accepted

Interrupt	Change of IPL
Reset	Level 0 (“000_2”) is set
$\overline{\text{NMI}}$	Level 7 (“111_2”) is set
$\overline{\text{DBC}}$	Does not change
Watchdog timer	Level 7 (“111_2”) is set
Single step	Does not change
Address match	Does not change
Software interrupt	Does not change

NMI Interrupt

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

This pin cannot be used as a normal port input.

Notes:

(1) When not intending to use the NMI function, be sure to connect the NMI pin to VCC. Because the NMI interrupt is non-maskable, it cannot be disabled.

(2) When the NMI pin input is “L”, do not set the microcomputer in stop mode or wait mode. The NMI interrupt is triggered by the falling edge, so the “L” level does not need to be maintained longer than necessary.

Key Input Interrupt

If the direction register of any of P104 to P107 is set for input and a falling edge is input to that port, a key input interrupt is generated. A key input interrupt can also be used as a key-on wakeup function for cancelling the wait mode or stop mode. However, if you intend to use the key input interrupt, do not use P104 to P107 as A-D input ports. Figure 1.28 shows a block diagram of the key input interrupt. Note that if an “L” level is input to any pin that has not been disabled for input, inputs to the other pins are not detected as an interrupt.

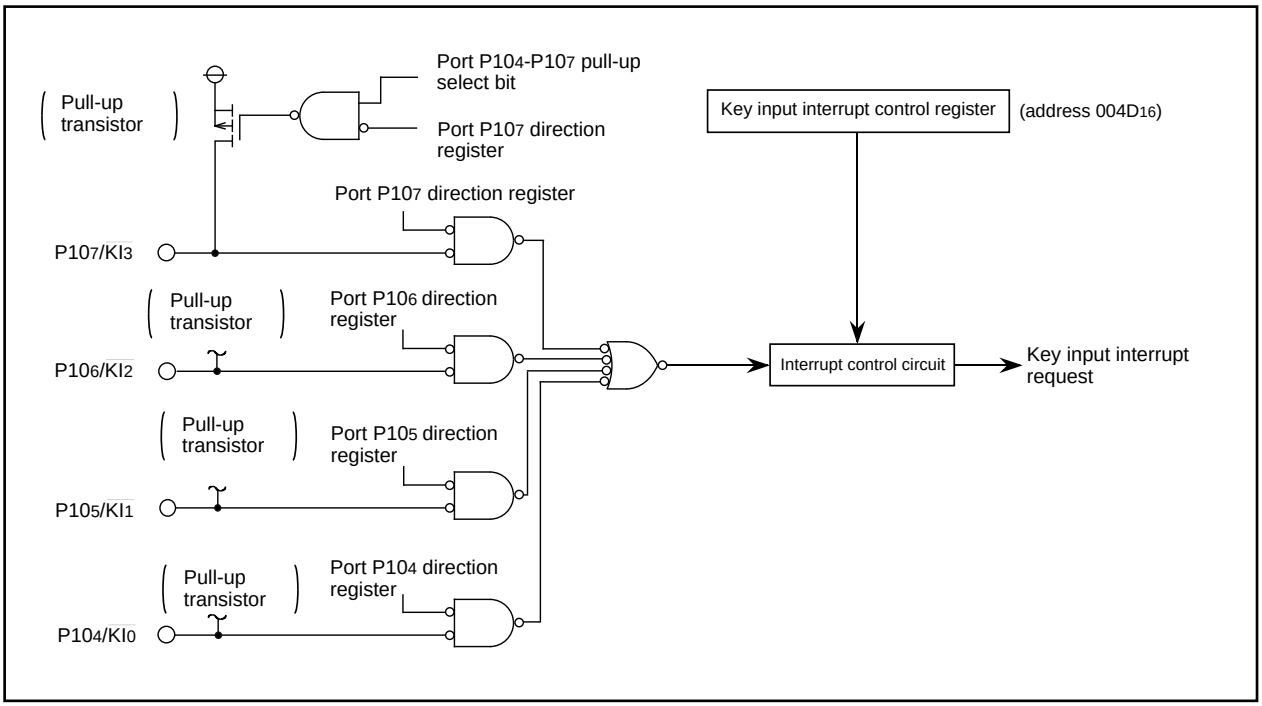


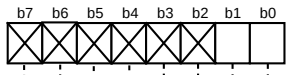
Figure 1.28. Block diagram of key input interrupt

Address Match Interrupt

An address match interrupt is generated when the address match interrupt address register contents match the program counter value. Two address match interrupts can be set, each of which can be enabled and disabled by an address match interrupt enable bit. Address match interrupts are not affected by the interrupt enable flag (I flag) and processor interrupt priority level (IPL).

Figure 1.29 shows a configuration of address match interrupt-related registers.

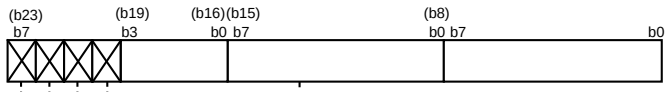
Address match interrupt enable register



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

Bit symbol	Bit name	Function	R	W
AIER0	Address match interrup 0 enable bit	0: Interrupt disabled 1: Interrupt enabled	○	○
AIER1	Address match interrupt 1 enable bit	0: Interrupt disabled 1: Interrupt enabled	○	○
Nothing is assigned. These bits can neither be set nor reset. When read, their contents are indeterminate.			—	—

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



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

Function	Values that can be set	R	W
Address setting register for address match interrupt	00000_{16} to $FFFFF_{16}$	○	○
Nothing is assigned. These bits can neither be set nor reset. When read, their contents are indeterminate.		—	—

Figure 1.29. Configuration of address match interrupt-related registers

Watchdog Timer

The watchdog timer has the function of detecting when the program is out of control. The watchdog timer is a 15-bit counter which down-counts the clock derived by dividing the internal clock ϕ using the prescaler. A watchdog timer interrupt is generated when an underflow occurs in the watchdog timer. When XIN is selected for the internal clock ϕ, 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 internal clock ϕ, the prescaler is set for division by 2 regardless of bit 7 of the watchdog timer control register (address 000F16). Table 1.20 shows a periodic table for the watchdog timer.

Table 1.20. Watchdog timer periodic table (XIN = 10MHz, XCIN = 32kHz)

CM07	CM06	CM17	CM16	Internal clock ϕ	WDC7	Period
0	0	0	0	10MHz	0	Approx. 52.4ms (Note)
					1	Approx. 419.2ms (Note)
0	0	0	1	5MHz	0	Approx. 104.9ms (Note)
					1	Approx. 838.8ms (Note)
0	0	1	0	2.5MHz	0	Approx. 209.7ms (Note)
					1	Approx. 1.68s (Note)
0	0	1	1	0.625MHz	0	Approx. 838.8ms (Note)
					1	Approx. 6.71s (Note)
0	1	Invalid	Invalid	1.25MHz	0	Approx. 419.2ms (Note)
					1	Approx. 3.35s (Note)
1	Invalid	Invalid	Invalid	32kHz	Invalid	Approx. 2s (Note)

Note: Error is generated by the prescaler.

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 1.30 shows a block diagram of the watchdog timer. Figure 1.31 shows the configuration of the watchdog timer-related registers.

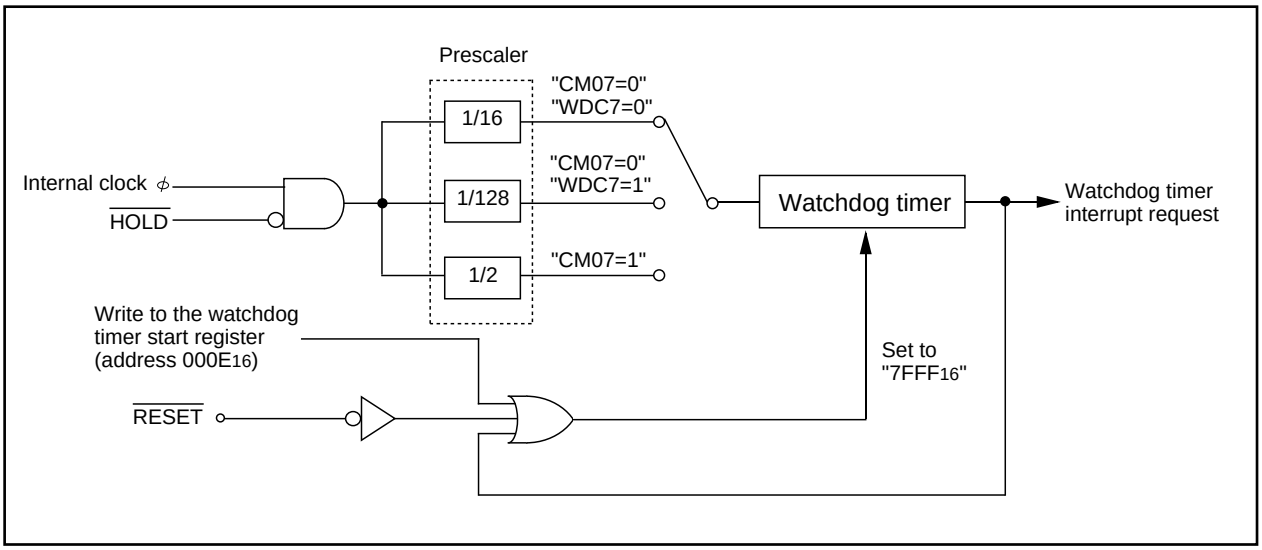


Figure 1.30. Block diagram of watchdog timer

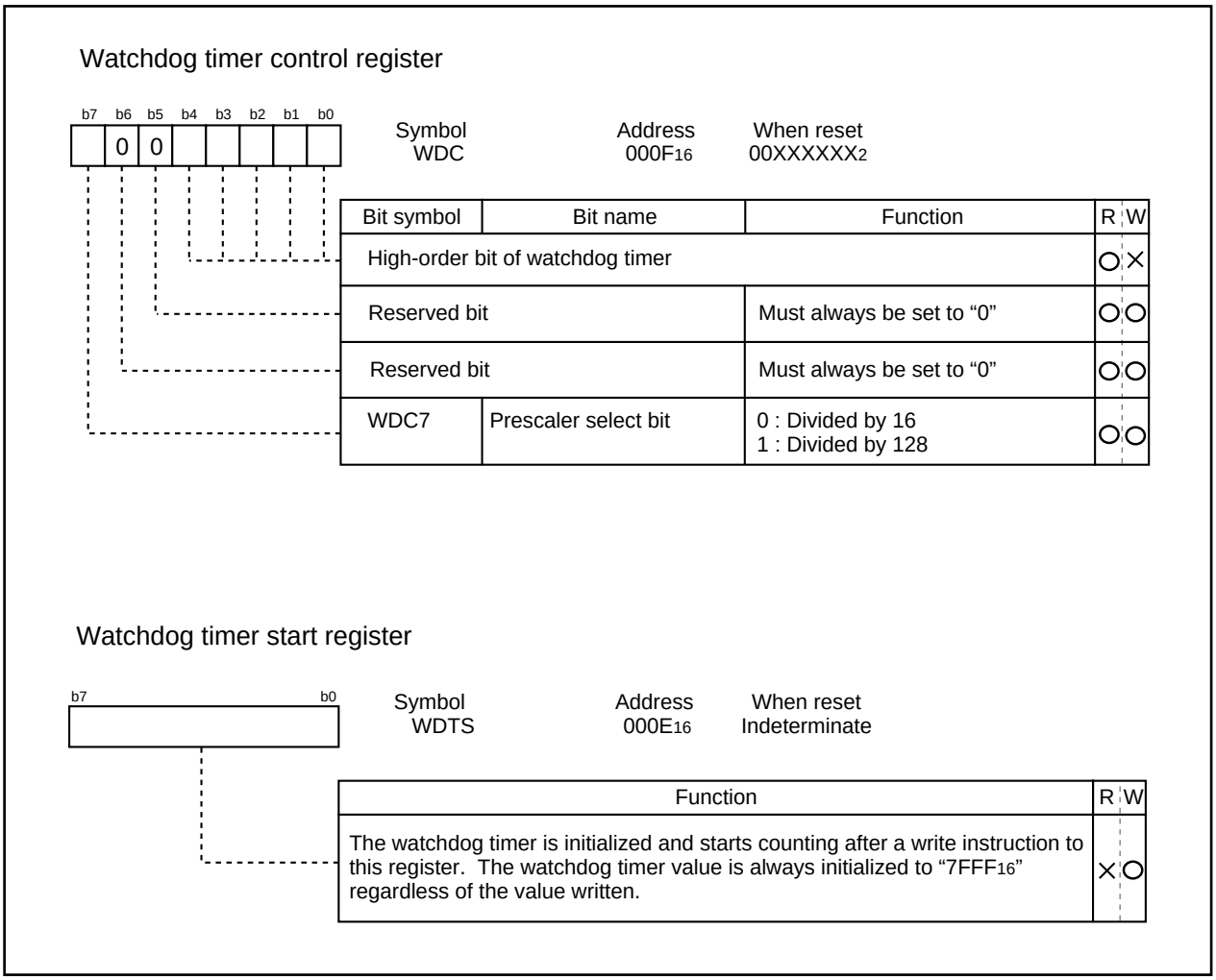


Figure 1. 31. Configuration of watchdog timer control and start registers

DMAC

This microcomputer has two DMAC (direct memory access controller) channels that allow data to be sent to memory without using the CPU. Table 1.21 shows the DMAC specifications. Figure 1.33 shows a block diagram of the DMAC. Figures 1.34 and 1.35 show the configuration of the registers used by the DMAC.

Table 1.21. DMAC specifications

Item	Specification
No. of channels	2 (cycle steal method)
Transfer memory space	• From any address in the 1M bytes space to a fixed address • From a fixed address to any address in the 1M bytes space • From a fixed address to a fixed address (Note that DMA-related registers [0020$_{16}$ to 003F$_{16}$] cannot be accessed)
Maximum No. of bytes transferred	128K bytes (with 16-bit transfers) or 64K bytes (with 8-bit transfers)
DMA request factors (Note)	Falling edge of $\overline{\text{INT0}}$ or $\overline{\text{INT1}}$ ($\overline{\text{INT0}}$ can be selected by DMA0, $\overline{\text{INT1}}$ by DMA1) Timer A0 to timer A4 interrupt requests Timer B0 to timer B2 interrupt requests UART0 transmission and reception interrupt requests UART1 transmission and reception interrupt requests A-D conversion interrupt requests Software triggers
Channel priority	DMA0 takes precedence if DMA0 and DMA1 requests are generated simultaneously
Transfer unit	8 bits or 16 bits
Transfer address direction	forward/fixed (forward direction cannot be specified for both source and destination simultaneously)
Transfer mode	• Single transfer The DMA enable bit is cleared and transfer ends when an underflow occurs in the transfer counter • Repeat transfer When an underflow occurs in the transfer counter, the value in the transfer counter reload register is reloaded into the transfer counter and the DMA transfer is repeated
DMA interrupt request generation timing	When an underflow occurs in the transfer counter
DMA startup	• Single transfer Transfer starts when the DMA is requested after “1” is written to the DMA enable bit • Repeat transfer Transfer starts when the DMA is requested after “1” is written to the DMA enable bit Transfer starts when the DMA is requested after an underflow occurs in the transfer counter
DMA shutdown	• When “0” is written to the DMA enable bit • When, in simple transfer mode, an underflow occurs in the transfer counter
Forward address pointer and reload timing for transfer counter	When DMA transfer starts, the value of whichever of the source or destination pointer that is set up as the forward pointer is reloaded into the forward address pointer. The value in the transfer counter reload register is reloaded into the transfer counter.
Writing to register	Registers specified for forward direction transfer are always write enabled. Registers specified for fixed address transfer are write-enabled when the DMA enable bit is “0”.
Reading the register	Can be read at any time. However, when the DMA enable bit is “1”, reading the register set up as the forward register is the same as reading the value of the forward address pointer.

Note: DMA transfer is not effective to any interrupt.

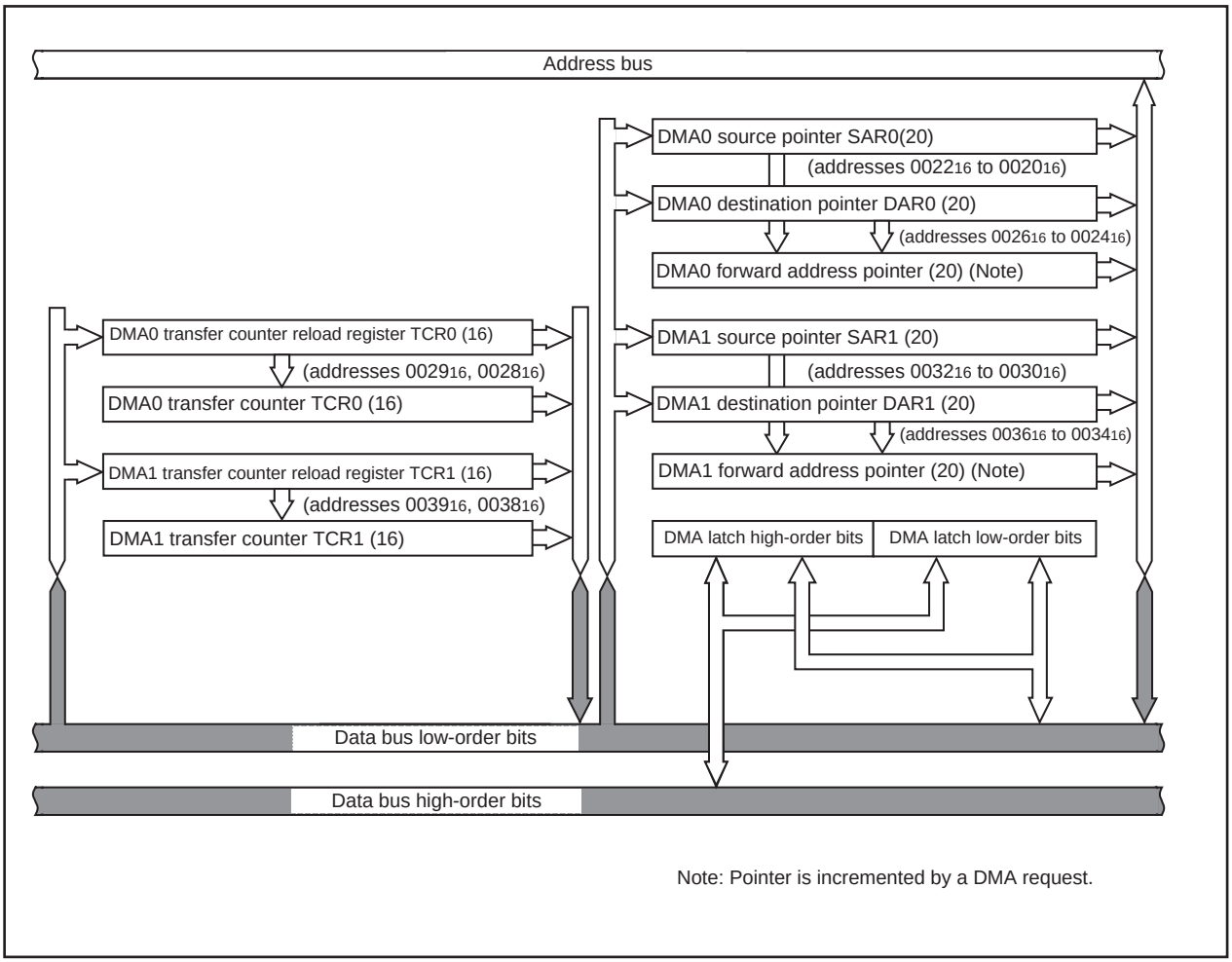
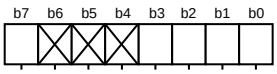


Figure 1.33. Block diagram of DMAC

DMAi request cause select register

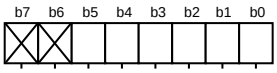


Symbol: DMiSL(i=0,1) | Address: 03B8$_{16}$,03BA$_{16}$ | When reset: 00$_{16}$

Bit symbol	Bit name	Function	R	W
DSEL0	DMA request cause select bit	b3 b2 b1 b0 0 0 0 0 : Falling edge of $\overline{INT0}$ / $\overline{INT1}$ pin (Note) 0 0 0 1 : Software trigger 0 0 1 0 : Timer A0 0 0 1 1 : Timer A1 0 1 0 0 : Timer A2 0 1 0 1 : Timer A3 0 1 1 0 : Timer A4 0 1 1 1 : Timer B0 1 0 0 0 : Timer B1 1 0 0 1 : Timer B2 1 0 1 0 : UART0 transmit 1 0 1 1 : UART0 receive 1 1 0 0 : UART1 transmit 1 1 0 1 : UART1 receive 1 1 1 0 : A-D conversion 1 1 1 1 : Inhibited	○	○
DSEL1			○	○
DSEL2			○	○
DSEL3			○	○
Nothing is assigned. These bits can neither be set nor reset. When read, the value of these bits is “0”.			–	–
DSR	Software DMA request bit	If software trigger is selected, a DMA request is generated by setting this bit to “1” (When read, the value of this bit is always “0”)	○	○

Note: Address 03B8$_{16}$ is for $\overline{INT0}$; address 03BA$_{16}$ is for $\overline{INT1}$.

DMAi control register



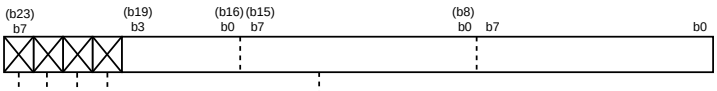
Symbol: DMiCON(i=0,1) | Address: 002C$_{16}$, 003C$_{16}$ | When reset: 00000X00$_2$

Bit symbol	Bit name	Function	R	W
DMBIT	Transfer unit bit select bit	0 : 16 bits 1 : 8 bits	○	○
DMASL	Repeat transfer mode select bit	0 : Single transfer 1 : Repeat transfer	○	○
DMAS	DMA request bit (Note 1)	0 : DMA not requested 1 : DMA requested	○	○ (Note 2)
DMAE	DMA enable bit	0 : Disabled 1 : Enabled	○	○
DSD	Source address direction select bit (Note 3)	0 : Fixed 1 : Forward	○	○
DAD	Destination address direction select bit (Note 3)	0 : Fixed 1 : Forward	○	○
Nothing is assigned. These bits can neither be set nor reset. When read, the value of these bits is “0”.			–	–

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

Figure 1.34. Configuration of DMAC register (1)

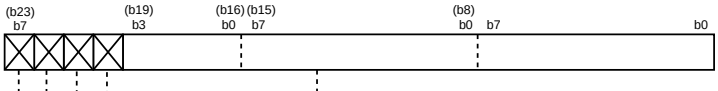
DMAi source pointer (i = 0, 1)



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

Function	Transfer count specification	R	W
• Source pointer Stores the source address	00000_{16} to $FFFFF_{16}$	○	○
Nothing is assigned. These bits can neither be set nor reset. When read, the value of these bits is “0”.		—	—

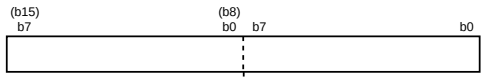
DMAi destination pointer (i = 0, 1)



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

Function	Transfer count specification	R	W
• Destination pointer Stores the destination address	00000_{16} to $FFFFF_{16}$	○	○
Nothing is assigned. These bits can neither be set nor reset. When read, the value of these bits is “0”.		—	—

DMAi transfer counter (i = 0, 1)



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

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

Figure 1.35. Configuration of DMAC register (2)

(1) Transfer cycle

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

(a) Effect of source and destination addresses

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

(b) Effect of BYTE pin level

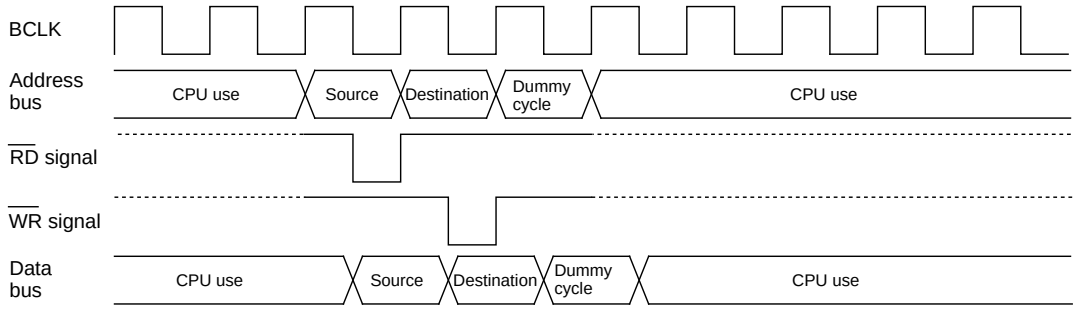
When transferring 16-bit data over an 8-bit data bus (BYTE pin = “H”) in memory expansion mode and microprocessor mode, the 16 bits of data are sent in two 8-bit blocks. Therefore, two bus cycles are required for reading the data and two are required for writing the data. Also, in contrast to when the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

(c) Effect of software wait

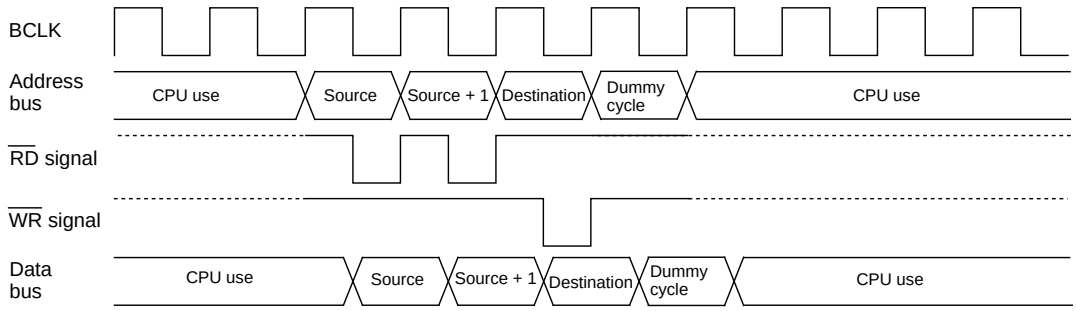
When the SFR area or a memory area with a software wait is accessed, the number of cycles is increased for the wait by 1 bus cycle. The length of the cycle is determined by BCLK.

Figure 1.36 shows an example of the transfer cycles for a source read. For convenience, the destination write cycle is shown as one cycle and the source read cycles for the different conditions are shown. In reality, the destination write cycle is subject to the same conditions as the source read cycle, with the transfer cycle changing accordingly. When calculating the transfer cycle, remember to apply the respective conditions to both the destination write cycle and the source read cycle. For example (2) in Figure 1.36, if data is being transferred in 16-bit units on an 8-bit bus, two bus cycles are required for both the source read cycle and the destination write cycle.

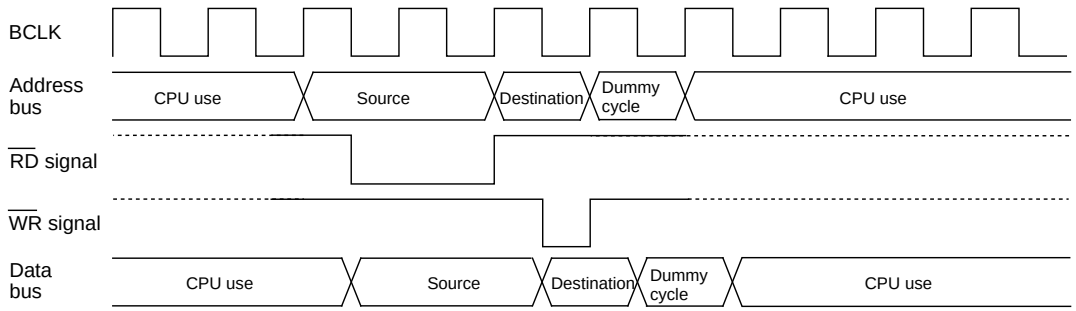
(1) 8-bit transfers
16-bit transfers from even address and the source address is even.



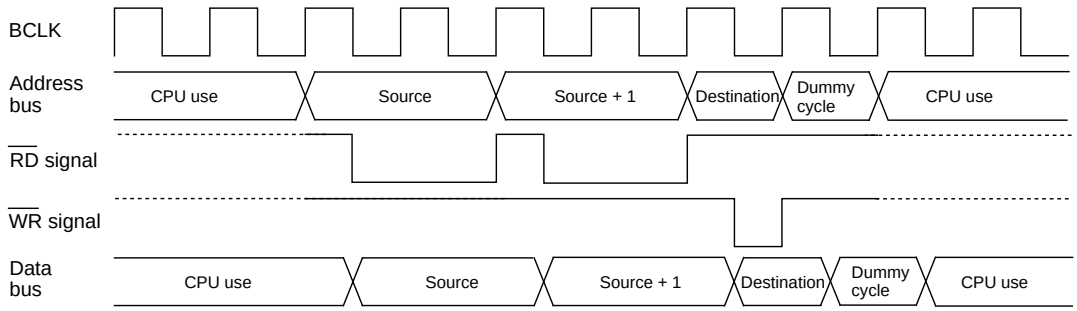
(2) 16-bit transfers and the source address is odd
Transferring 16-bit data on an 8-bit data bus (In this case, there are also two destination write cycles).



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



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



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

Figure 1.36. Example of the transfer cycles for a source read

(2) DMAC transfer cycles

Any combination of even or odd transfer read and write addresses is possible. Table 1.22 shows the number of DMAC transfer cycles.

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

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

Table 1.22. No. of DMAC transfer cycles

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

Coefficient j, k

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

Timer

There are eight 16-bit timers. These timers can be classified by function into timers A (five) and timers B (three). All these timers function independently. Figure 1.37 shows a block diagram of timers.

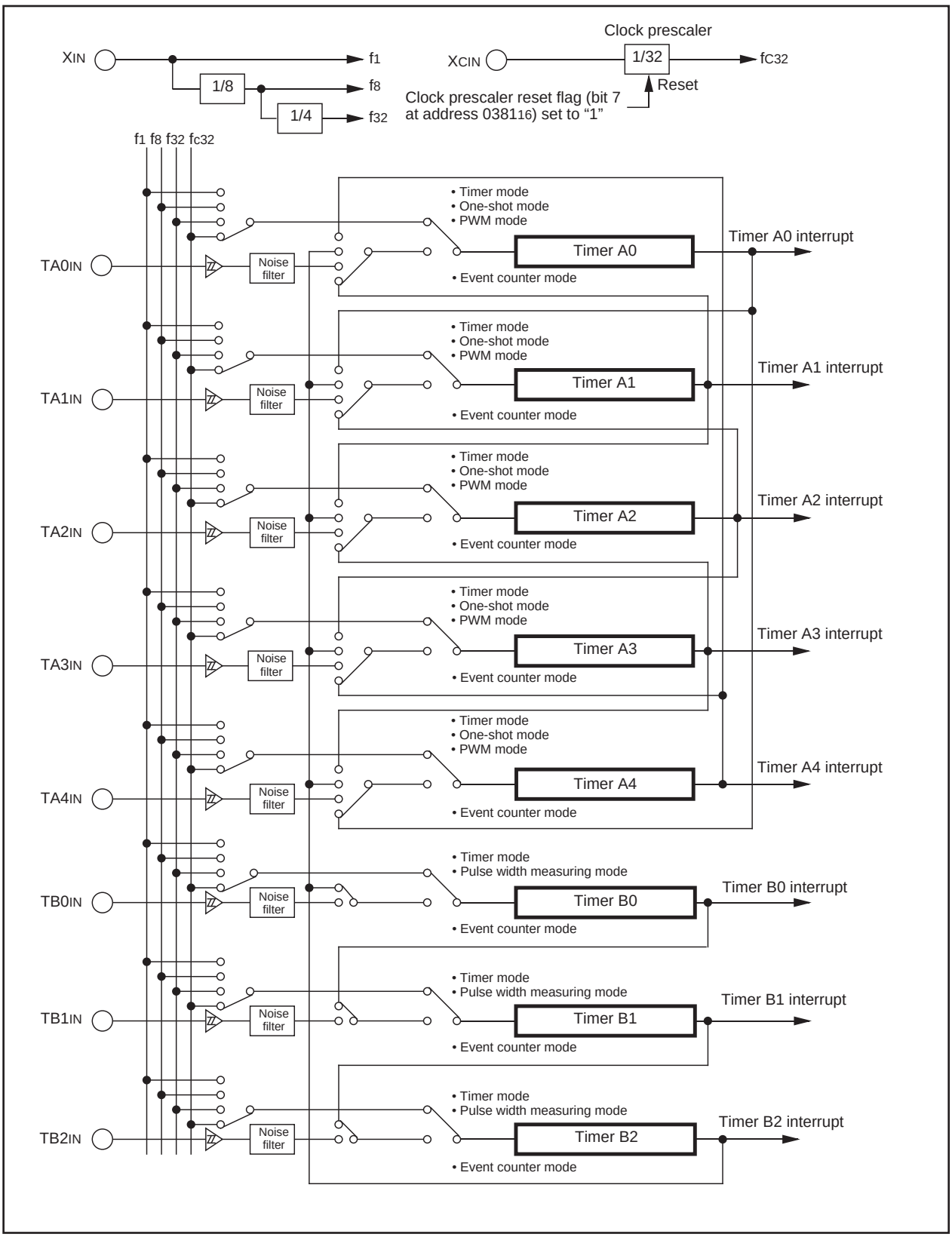


Figure 1.37. Block diagram of timer

Timer A

Figure 1.38 shows a block diagram of timer A. Figures 1.39 to 1.41 show configuration of timer A-related registers.

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

Timer A has the four operation modes listed as follows:

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

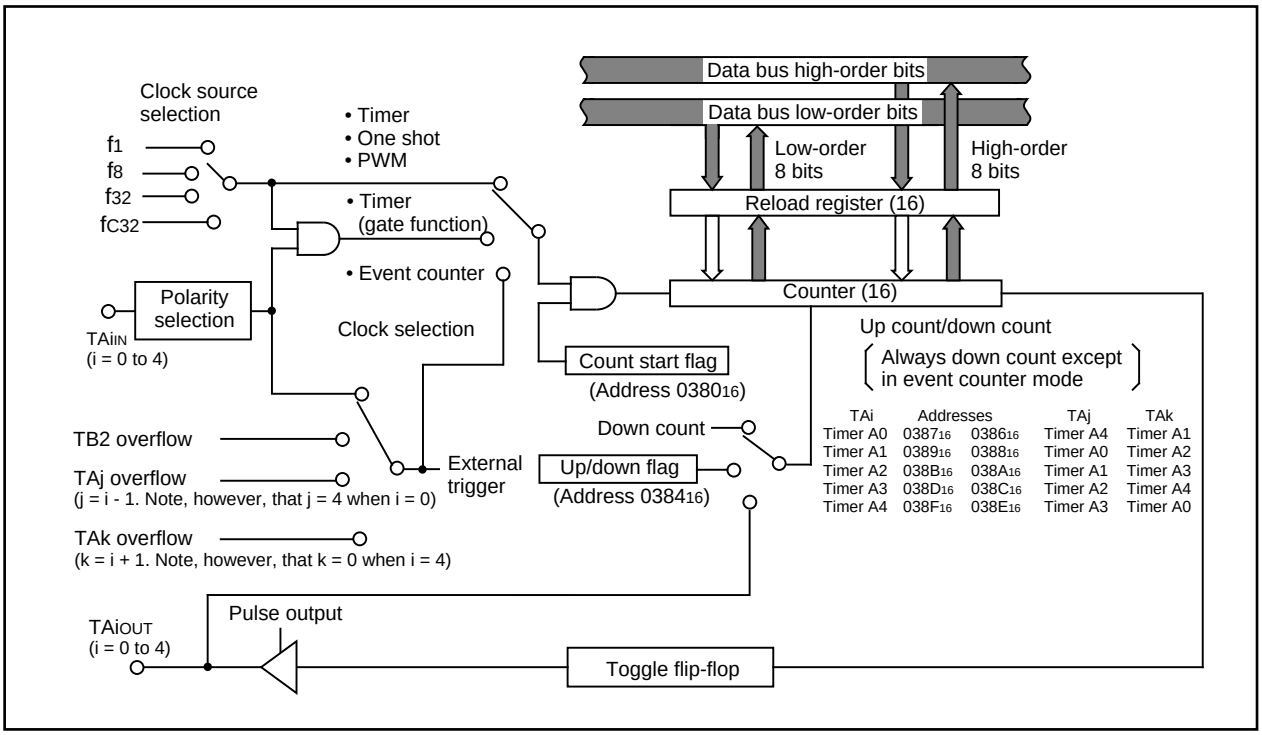


Figure 1.38. Block diagram of timer A

Timer Ai mode register

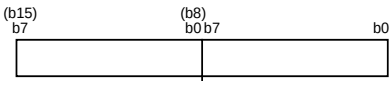
b7 b6 b5 b4 b3 b2 b1 b0

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

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

Figure 1.39. Configuration of timer A-related registers (1)

Timer Ai register (Note)



Symbol	Address	When reset
TA0	0387$_{16}$,0386$_{16}$	Indeterminate
TA1	0389$_{16}$,0388$_{16}$	Indeterminate
TA2	038B$_{16}$,038A$_{16}$	Indeterminate
TA3	038D$_{16}$,038C$_{16}$	Indeterminate
TA4	038F$_{16}$,038E$_{16}$	Indeterminate

Function	Values that can be set	R	W
• Timer mode Counts an internal count source	0000$_{16}$ to FFFF	O	O
• Event counter mode Counts pulses from an external source or timer overflow	0000$_{16}$ to FFFF$_{16}$	O	O
• One-shot timer mode Counts a one shot width	0000$_{16}$ to FFFF$_{16}$	X	O
• Pulse width modulation mode (16-bit PWM) Functions as a 16-bit pulse width modulator	0000$_{16}$ to FFFE$_{16}$	X	O
• Pulse width modulation mode (8-bit PWM) Timer low-order address functions as an 8-bit prescaler and high-order address functions as an 8-bit pulse width modulator	00$_{16}$ to FE$_{16}$ (Both high-order and low-order addresses)	X	O

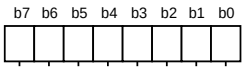
Note: Read and write data is in 16-bit units.

Count start flag

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

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

Up/down flag

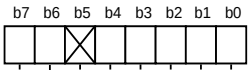


Symbol	Address	When reset
UDF	0384$_{16}$	00$_{16}$

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

Figure 1.40. Configuration of timer A-related registers (2)

One-shot start flag

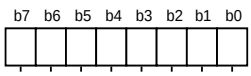


Symbol: ONSF | Address: 0382_{16} | When reset: $00X00000_2$

Bit symbol	Bit name	Function	R	W
TA0OS	Timer A0 one-shot start flag	1: Timer start When read, the value is "0"	O	O
TA1OS	Timer A1 one-shot start flag		O	O
TA2OS	Timer A2 one-shot start flag		O	O
TA3OS	Timer A3 one-shot start flag		O	O
TA4OS	Timer A4 one-shot start flag		O	O
Nothing is assigned. This bit can neither be set nor reset. When read, the content is indeterminate.			–	–
TA0TGL	Timer A0 event/trigger select bit	b7 b6 0 0 : Input on $TA0_{IN}$ is selected (Note) 0 1 : TB2 overflow is selected 1 0 : TA4 overflow is selected 1 1 : TA1 overflow is selected	O	O
TA0TGH			O	O

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

Trigger select register

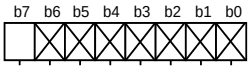


Symbol: TRGSR | Address: 0383_{16} | When reset: 00_{16}

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

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

Clock prescaler reset flag



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

Bit symbol	Bit name	Function	R	W
Nothing is assigned. These bits can neither be set nor reset. When read, their contents are indeterminate.			–	–
CPSR	Clock prescaler reset flag	0 : No effect 1 : Prescaler is reset (When read, the value is "0")	O	O

Figure 1.41. Configuration of timer A-related registers (3)

(1) Timer mode

In this mode, the timer counts an internally generated count source. (See Table 1.23.) Figure 1.42 shows the configuration of the timer Ai mode register in timer mode.

Table 1.23. Specifications of timer mode

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

Timer Ai mode register

b7	b6	b5	b4	b3	b2	b1	b0
		0				0	0

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

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

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

Figure 1.42. Configuration of timer Ai mode register in timer mode

(2) Event counter mode

In this mode, the timer counts an external signal or an internal timer's overflow. Timers A0 and A1 can count a single-phase external signal. Timers A2, A3, and A4 can count a single-phase and a two-phase external signal. Table 1.24 lists timer specifications when counting a single-phase external signal. Figure 1.43 shows the configuration of the timer Ai mode register in event counter mode.

Table 1.25 lists timer specifications when counting a two-phase external signal. Figure 1.44 shows the configuration of the timer Ai mode register in event counter mode.

Table 1.24. Timer specifications in event counter mode (when not processing two-phase pulse signal)

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

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

Timer Ai mode register

b7	b6	b5	b4	b3	b2	b1	b0
		0				0	1

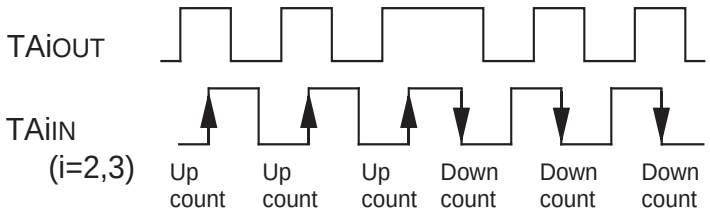
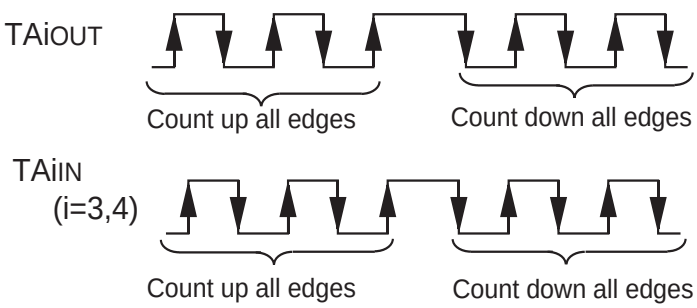
Symbol: TAiMR(i = 0, 1) — Address: 0396_{16}, 0397_{16} — When reset: 00_{16}

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

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

Figure 1.43. Configuration of timer Ai mode register in event counter mode

Table 1.25. Timer specifications in event counter mode (when processing two-phase pulse signal with timers A2, A3, and A4)

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

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

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

b7	b6	b5	b4	b3	b2	b1	b0
		0				0	1

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

Bit symbol	Bit name	Function	R	W
TMOD0	Operation mode select bit	b1 b0 0 1 : Event counter mode	○	○
TMOD1			○	○
MR0	Pulse output function select bit	0 : Pulse is not output (TAiOUT pin is a normal port pin) 1 : Pulse is output (Note 1) (TAiOUT pin is a pulse output pin)	○	○
MR1	Count polarity select bit (Note 2)	0 : Counts external signal's falling edges 1 : Counts external signal's rising edges	○	○
MR2	Up/down switching cause select bit	0 : Up/down flag's content 1 : TAiOUT pin's input signal (Note 3)	○	○
MR3	0 (Must always be “0” in event counter mode)		○	○
TCK0	Count operation type select bit	0 : Reload type 1 : Free-run type	○	○
TCK1	Two-phase pulse signal processing operation select bit (Note 4)(Note 5)	0 : Normal processing operation 1 : Multiply-by-4 processing operation	○	○

Note 1: The settings of the corresponding port register and port direction register are invalid
Note 2: This bit is valid when only counting an external signal.
Note 3: Set the corresponding port direction register to “0”.
Note 4: This bit is valid for timer A3 mode register.
For timer A2 and A4 mode registers, this bit can be “0 ”or “1”.
Note 5: When performing two-phase pulse signal processing, make sure the two-phase pulse signal processing operation select bit (address 0384$_{16}$) is set to “1”. Also, always be sure to set the event/trigger select bit (addresses 0382$_{16}$ and 0383$_{16}$) to “00”.

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

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

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

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

Note 1: This bit is valid for timer A3 mode register.
For timer A2 and A4 mode registers, this bit can be “0” or “1”.
Note 2: When performing two-phase pulse signal processing, make sure the two-phase pulse signal processing operation select bit (address 0384$_{16}$) is set to “1”. Also, always be sure to set the event/trigger select bit (addresses 0382$_{16}$ and 0383$_{16}$) to “00”.

Figure 1.44. Configuration of timer Ai mode register in event counter mode

(3) One-shot timer mode

In this mode, the timer operates only once. (See Table 1.26.) When a trigger occurs, the timer starts up and continues operating for a given period. Figure 1.45 shows the configuration of the timer Ai mode register in one-shot timer mode.

Table 1.26. Timer specifications in one-shot timer mode

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

Timer Ai mode register

b7	b6	b5	b4	b3	b2	b1	b0
		0				1	0

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

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

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

Figure45. Configuration of timer Ai mode register in one-shot timer mode

(4) Pulse width modulation (PWM) mode

In this mode, the timer outputs pulses of a given width in succession. (See Table 1.27.) In this mode, the counter functions as either a 16-bit pulse width modulator or an 8-bit pulse width modulator. Figure 1.46 shows the configuration of the timer Ai mode register in pulse width modulation mode. Figure 1.47 shows an example of how a 16-bit pulse width modulator operates. Figure 1.48 shows an example of how an 8-bit pulse width modulator operates.

Table 1.27. Timer specifications in pulse width modulation mode

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

Timer Ai mode register

b7	b6	b5	b4	b3	b2	b1	b0
					1	1	1

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

Bit symbol	Bit name	Function	R	W
TMOD0	Operation mode select bit	b1 b0 1 1 : PWM mode	○	○
TMOD1			○	○
MR0	1 (Must always be fixed to "1" in PWM mode)		○	○
MR1	External trigger select bit (Note 1)	0: Falling edge of TAiIN pin's input signal (Note 2) 1: Rising edge of TAiIN pin's input signal (Note 2)	○	○
MR2	Trigger select bit	0: Count start flag is valid 1: Selected by event/trigger select register	○	○
MR3	16/8-bit PWM mode select bit	0: Functions as a 16-bit pulse width modulator 1: Functions as an 8-bit pulse width modulator	○	○
TCK0	Count source select bit	b7 b6 0 0 : f_1 0 1 : f_8 1 0 : f_{32} 1 1 : f_{C32}	○	○
TCK1			○	○

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

Figure 1.46. Configuration of timer Ai mode register in pulse width modulation mode

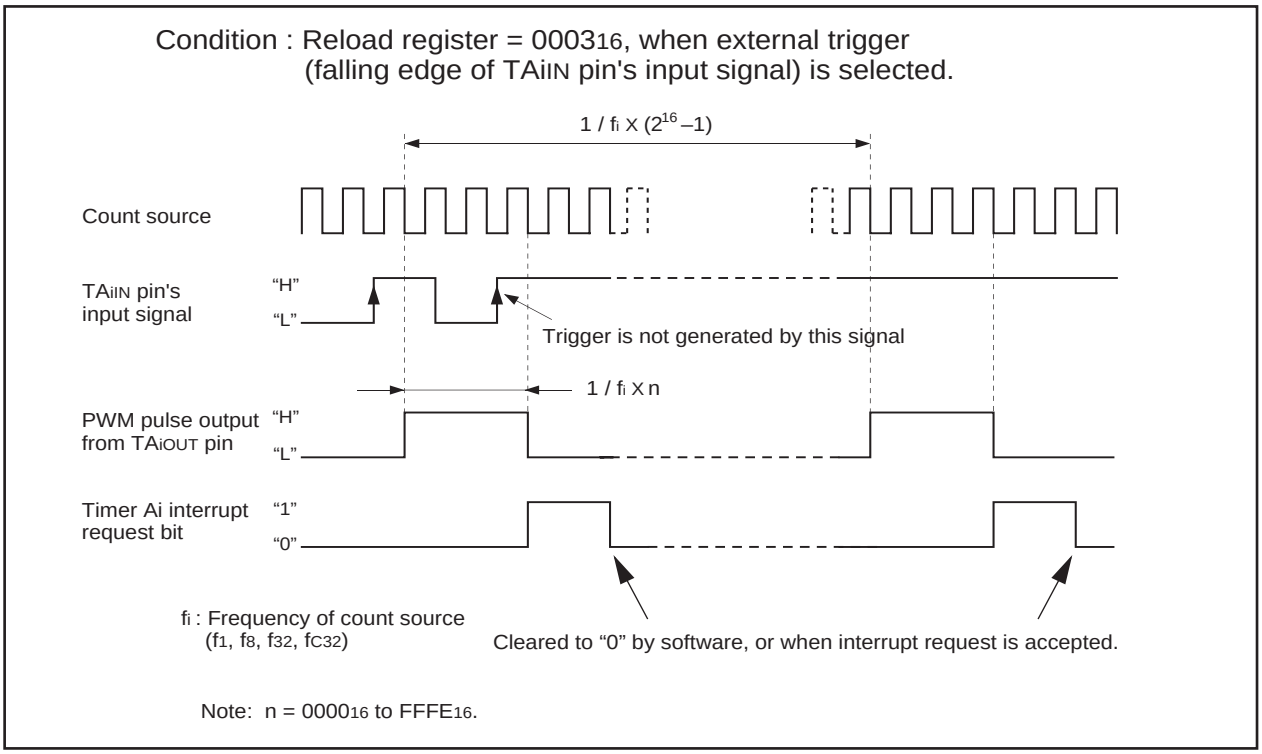


Figure 1.47. Example of how a 16-bit pulse width modulator operates

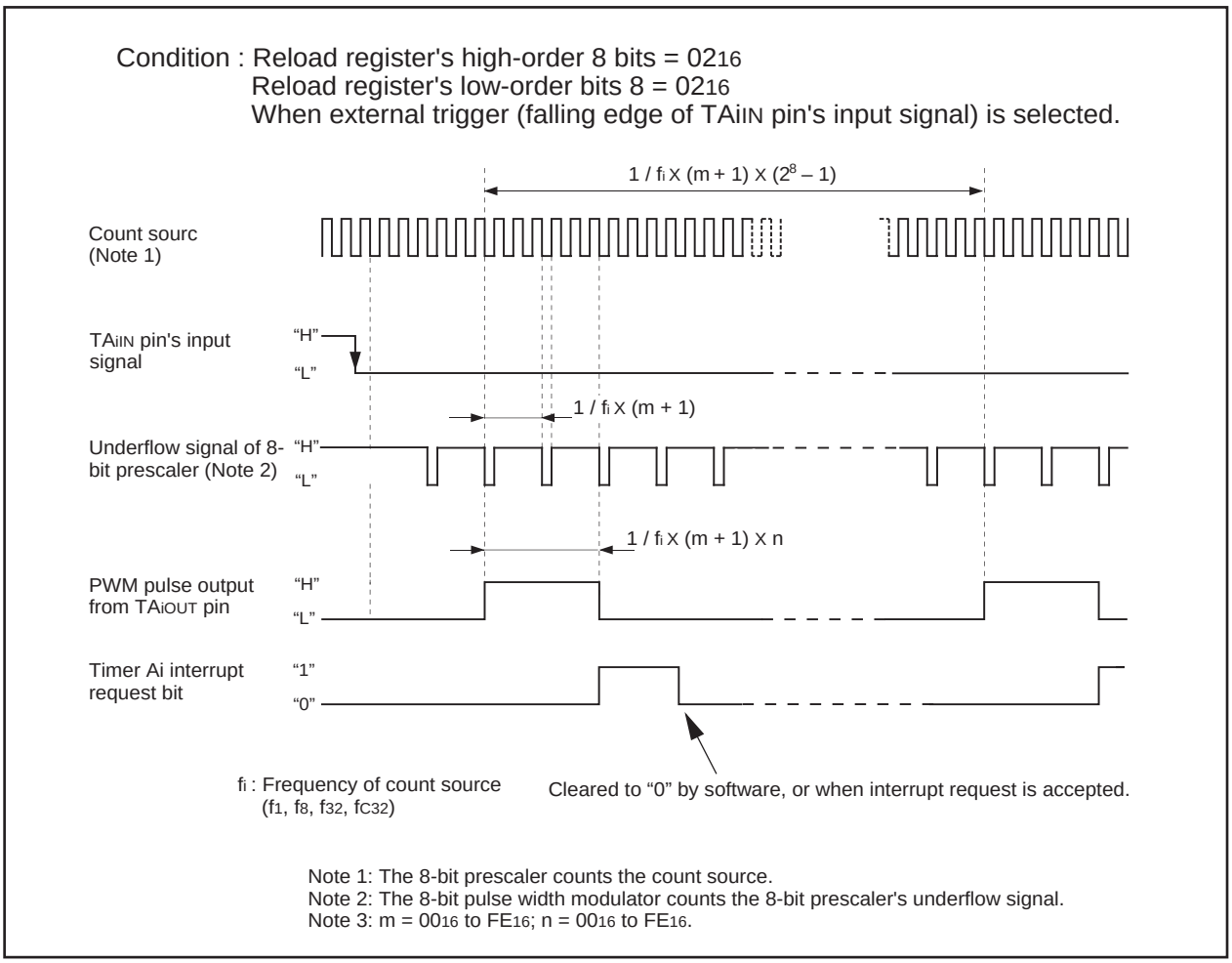


Figure 1.48. Example of how an 8-bit pulse width modulator operates

Timer B

Figure 1.49 shows a block diagram of timer B. Figures 1.50 and 1.51 show configuration of timer B-related registers.

Use the timer Bi mode register (i = 0 to 2) bits 0 and 1 to choose the desired mode.

Timer B has three operation modes listed as follows:

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

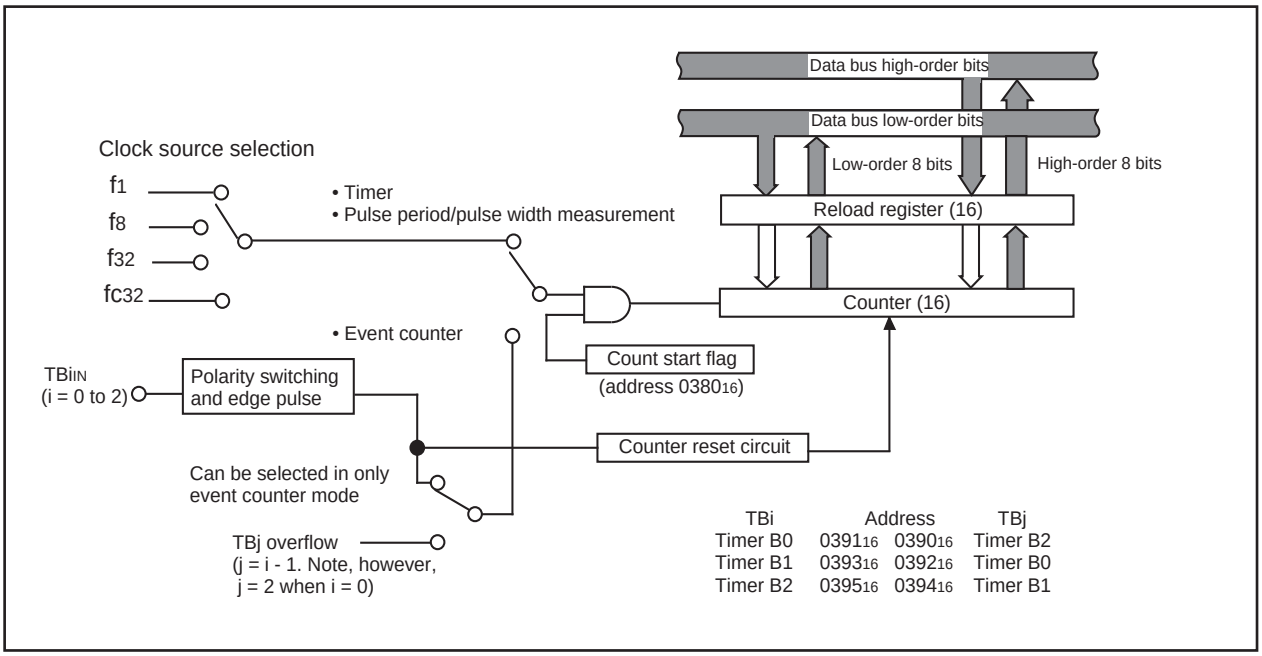


Figure 1.49. Block diagram of timer B

Timer Bi mode register

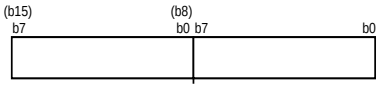
Symbol	Address	When reset
TBiMR(i = 0 to 2)	039B16 to 039D16	00XX00002

Bit symbol	Bit name	Function	R	W
TMOD0	Operation mode select bit	b1 b0 0 0 : Timer mode 0 1 : Event counter mode 1 0 : Pulse period/pulse width measurement mode 1 1 : Inhibited	○	○
TMOD1			○	○
MR0	Function varies with each operation mode		○	○
MR1			○	○
MR2			○ (Note 1) × (Note 2)	○ ×
MR3			○	×
TCK0	Count source select bit (Function varies with each operation mode)		○	○
TCK1			○	○

Note 1: Timer B0.
Note 2: Timer B1, timer B2.

Figure 1.50. Configuration of timer B-related registers (1)

Timer Bi register (Note)

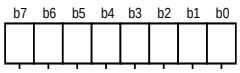


Symbol	Address	When reset
TB0	039116, 039016	Indeterminate
TB1	039316, 039216	Indeterminate
TB2	039516, 039416	Indeterminate

Function	Values that can be set	R	W
• Timer mode Counts the timer's period	000016 to FFFF16	O	O
• Event counter mode Counts external pulses input or a timer overflow	000016 to FFFF16	O	O
• Pulse period / pulse width measurement mode Measures a pulse period or width	——	O	X

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

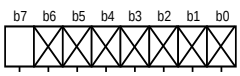
Count start flag



Symbol	Address	When reset
TABSR	038016	0016

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

Clock prescaler reset flag



Symbol	Address	When reset
CPSRF	038116	0XXXXXXX2

Bit symbol	Bit name	Function	R	W
Nothing is assigned. These bits can neither be set nor reset. When read, their contents are indeterminate.			—	—
CPSR	Clock prescaler reset flag	0 : No effect 1 : Prescaler is reset (When read, the value is "0")	O	O

Figure 1.51. Configuration of timer B-related registers (2)

(1) Timer mode

In this mode, the timer counts an internally generated count source. (See Table 1.28.) Figure 1.52 shows the configuration of the timer Bi mode register in timer mode.

Table 1.28. Timer specifications in timer mode

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

Timer Bi mode register

b7	b6	b5	b4	b3	b2	b1	b0
						0	0

Symbol: TBiMR(i=0 to 2) Address: $039B_{16}$ to $039D_{16}$ When reset: $00XX0000_2$

Bit symbol	Bit name	Function	R	W
TMOD0	Operation mode select bit	b1 b0 0 0 : Timer mode	○	○
TMOD1			○	○
MR0	Invalid in timer mode Can be “0” or “1”		○	○
MR1			○	○
MR2	0 (Fixed to “0” in timer mode ; i = 0)		○ (Note 1)	○
	Nothing is assigned (i = 1,2). This bit can neither be set nor reset. When read, its content is indeterminate.		× (Note 2)	×
MR3	Invalid in timer mode. This bit can neither be set nor reset. When read in timer mode, its content is indeterminate.		○	×
TCK0	Count source select bit	b7 b6 0 0 : f_1 0 1 : f_8 1 0 : f_{32} 1 1 : f_{C32}	○	○
TCK1			○	○

Note 1: Timer B0.
Note 2: Timer B1, timer B2.

Figure 1.52. Configuration of timer Bi mode register in timer mode

(2) Event counter mode

In this mode, the timer counts an external signal or an internal timer's overflow. (See Table 1.29.) Figure 1.53 shows the configuration of the timer Bi mode register in event counter mode.

Table 1.29. Timer specifications in event counter mode

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

Timer Bi mode register

b7	b6	b5	b4	b3	b2	b1	b0
						0	1

Symbol: TBiMR(i=0 to 2) Address: 039B16 to 039D16 When reset: 00XX00002

Bit symbol	Bit name	Function	R	W
TMOD0	Operation mode select bit	b1 b0 0 1 : Event counter mode	○	○
TMOD1			○	○
MR0	Count polarity select bit (Note 1)	b3 b2 0 0 : Counts external signal's falling edges 0 1 : Counts external signal's rising edges 1 0 : Counts external signal's falling and rising edges 1 1 : Inhibited	○	○
MR1			○	○
MR2	0 (Fixed to "0" in event counter mode; i = 0)		○ (Note 2)	○
	Nothing is assigned (i = 1, 2). This bit can neither be set nor reset. When read, its content is indeterminate.		× (Note 3)	×
MR3	Invalid in event counter mode. This bit can neither be set nor reset. When read in event counter mode, its content is indeterminate.		○	×
TCK0	Invalid in event counter mode. Can be "0" or "1".		○	○
TCK1	Event clock select	0: Input from TBiIN pin (Note 4) 1: TBj overflow (j = i-1; however, j = 2 when i = 0)	○	○

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

Figure 1.53. Configuration of timer Bi mode register in event counter mode

(3) Pulse period/pulse width measurement mode

In this mode, the timer measures the pulse period or pulse width of an external signal. (See Table 1.30.) Figure 1.54 shows the configuration of timer Bi mode register in pulse period/pulse width measurement mode. Figure 1.55 shows the operation timing when measuring a pulse period. Figure 1.56 shows the operation timing when measuring a pulse width.

Table 1.30. Timer specifications in pulse period/pulse width measurement mode

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

Note 1: An interrupt request is not generated when the first effective edge is input after the timer has started counting.

Note 2: The value read out from the timer Bi register is indeterminate until the second effective edge is input after the timer.

Timer Bi mode register

b7	b6	b5	b4	b3	b2	b1	b0
						1	0

Symbol: TBiMR (i=0 to 2) — Address: $039B_{16}$ to $039D_{16}$ — When reset: $00XX0000_2$

Bit symbol	Bit name	Function	R	W
TMOD0	Operation mode select bit	b1 b0 1 0 : Pulse period / pulse width measurement mode	○	○
TMOD1			○	○
MR0	Measurement mode select bit	b3 b2 0 0 : Pulse period measurement (Interval between measurement pulse's falling edge to falling edge) 0 1 : Pulse period measurement (Interval between measurement pulse's rising edge to rising edge) 1 0 : Pulse width measurement (Interval between measurement pulse's falling edge to rising edge, and between rising edge to falling edge) 1 1 : Inhibited	○	○
MR1			○	○
MR2	0 (Fixed to "0" in pulse period/pulse width measurement mode; i = 0)		○ (Note 2)	○
	Nothing is assigned (i = 1, 2). This bit can neither be set nor reset. When read, its content is indeterminate.		× (Note 3)	×
MR3	Timer Bi overflow flag (Note 1)	0 : Timer did not overflow 1 : Timer has overflowed	○	×
TCK0	Count source select bit	b7 b6 0 0 : f_1 0 1 : f_8 1 0 : f_{32} 1 1 : f_{C32}	○	○
TCK1			○	○

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

Figure 1.54. Configuration of timer Bi mode register in pulse period/pulse width measurement mode

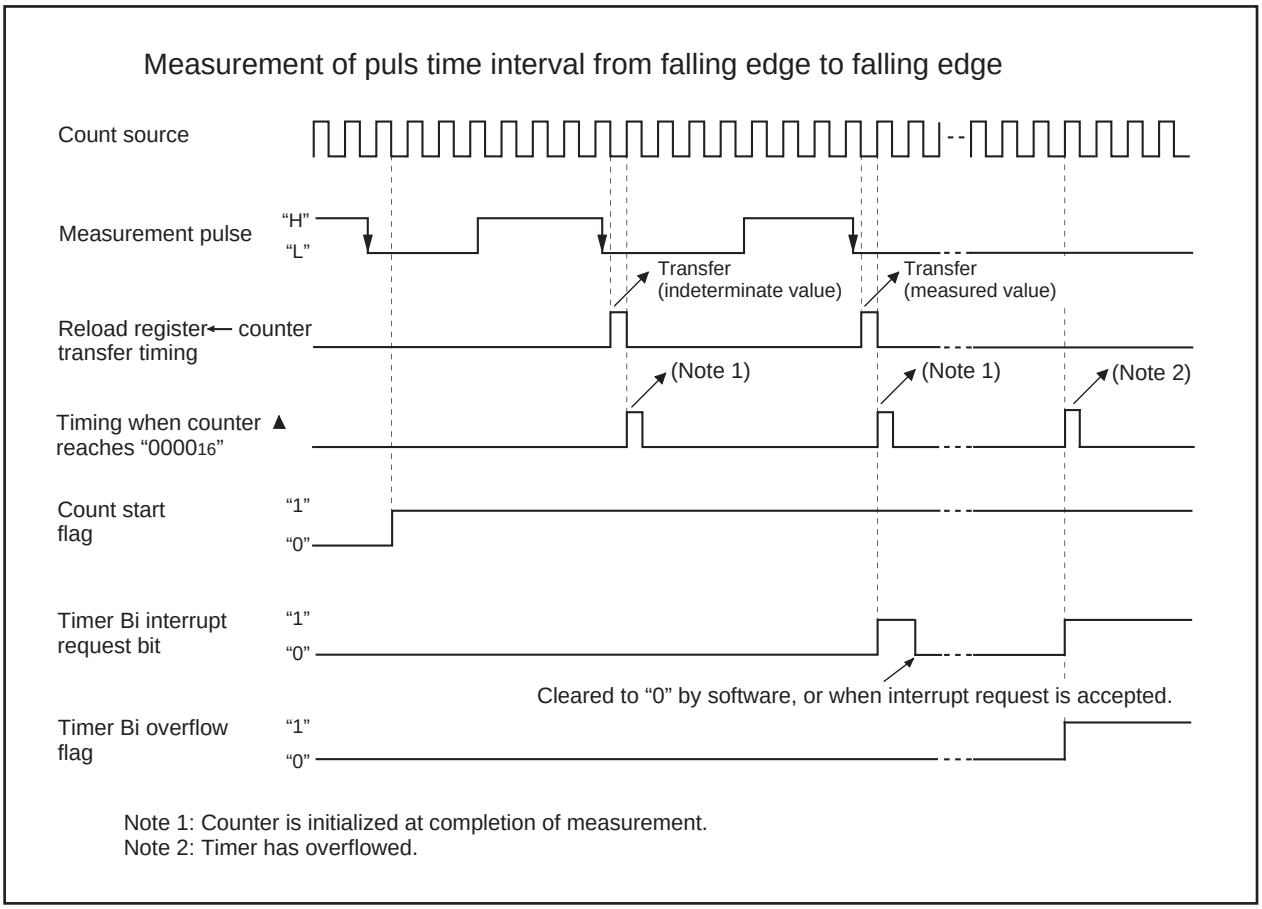


Figure 1.55. Operation timing when measuring a pulse period

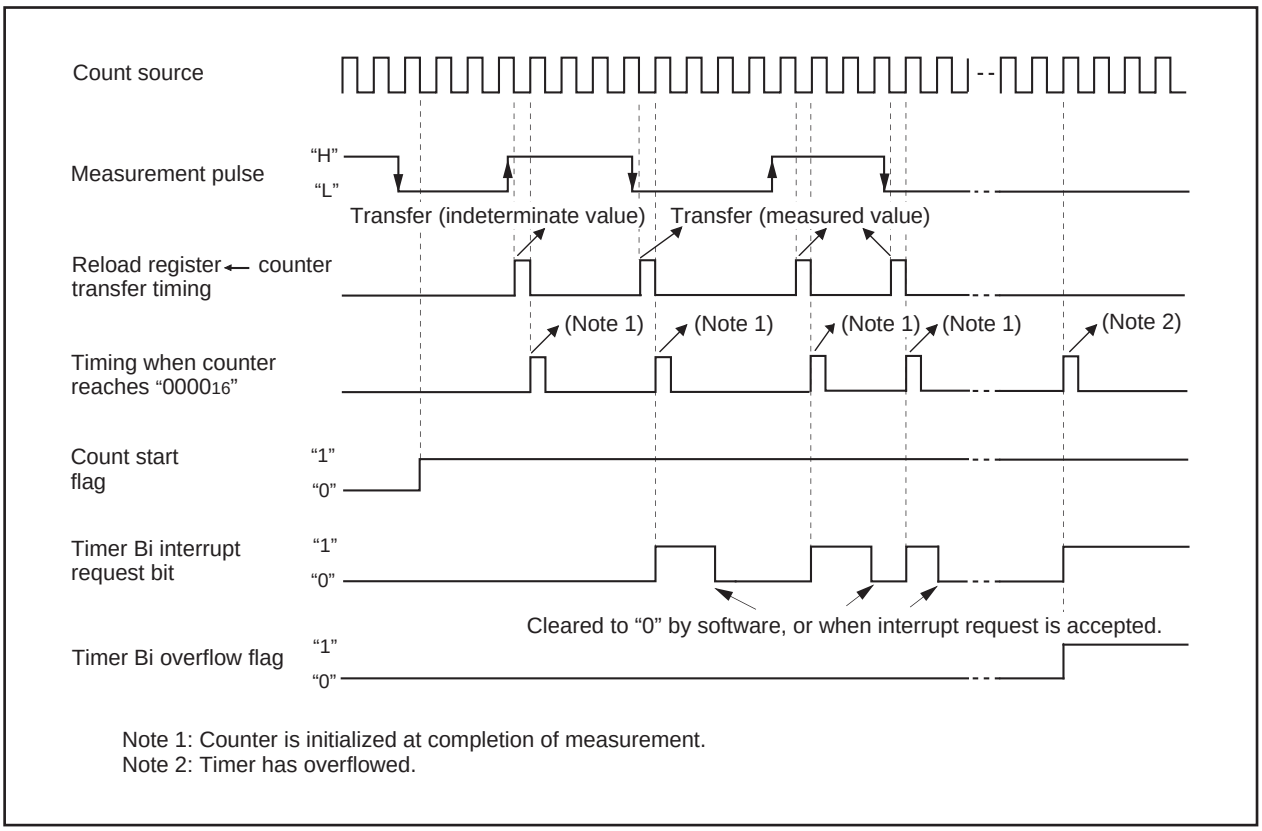


Figure 1.56. Operation timing when measuring a pulse width

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 1.57 shows a block diagram of UART0 and UART1. Figure 1.58 shows a block diagram of the transmit/receive unit.

UARTi (i = 0, 1) has two operation modes: a clock synchronous serial I/O mode and a clock asynchronous serial I/O mode (UART mode). The contents of the serial I/O mode select bits (bits 0 to 2 at addresses $03A0_{16}$ and $03A8_{16}$) determine whether UARTi 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. Figure 1.59 through 61 show configuration of UARTi-related registers.

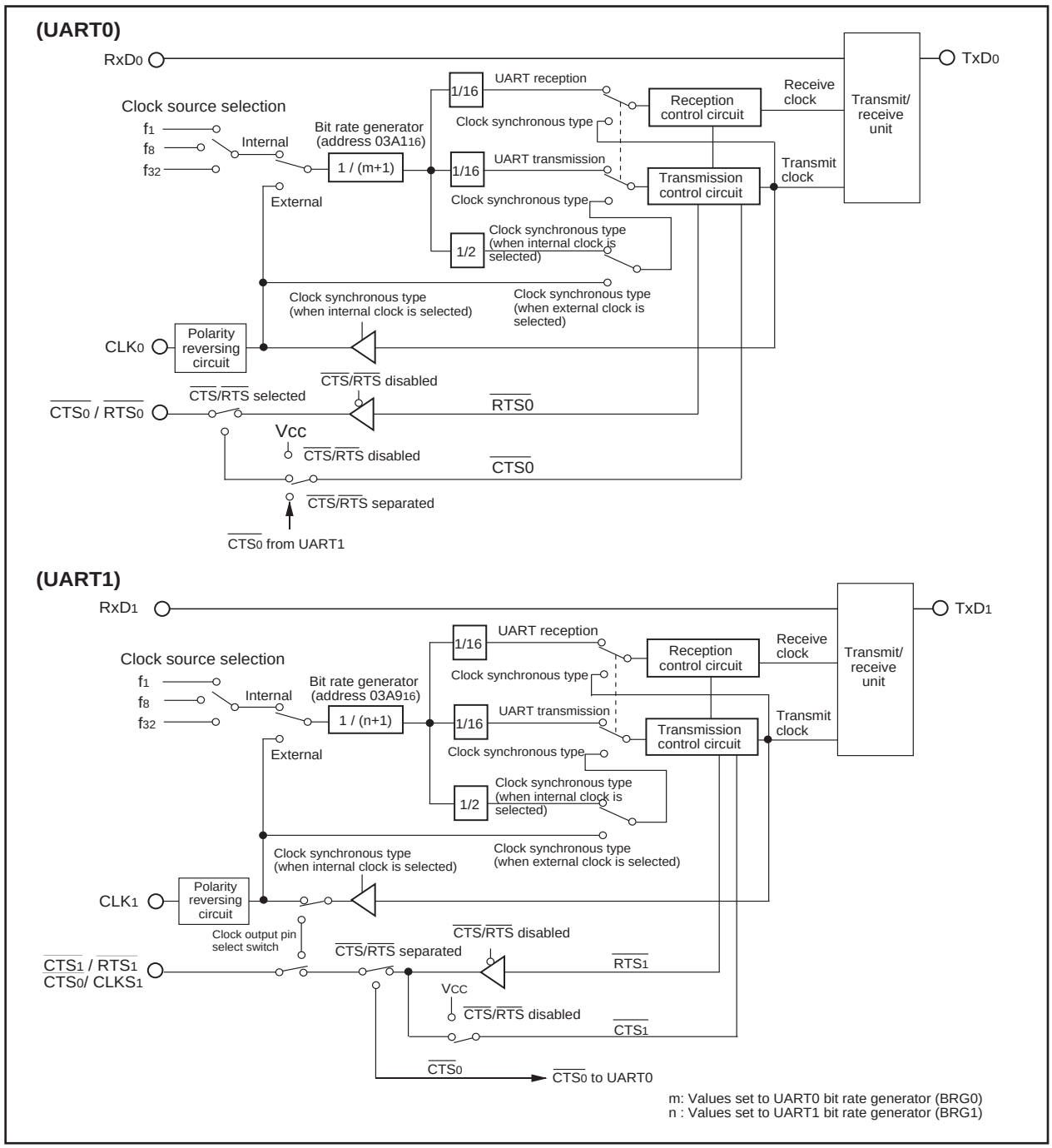


Figure 1.57. Block diagram of UARTi (i = 0, 1)

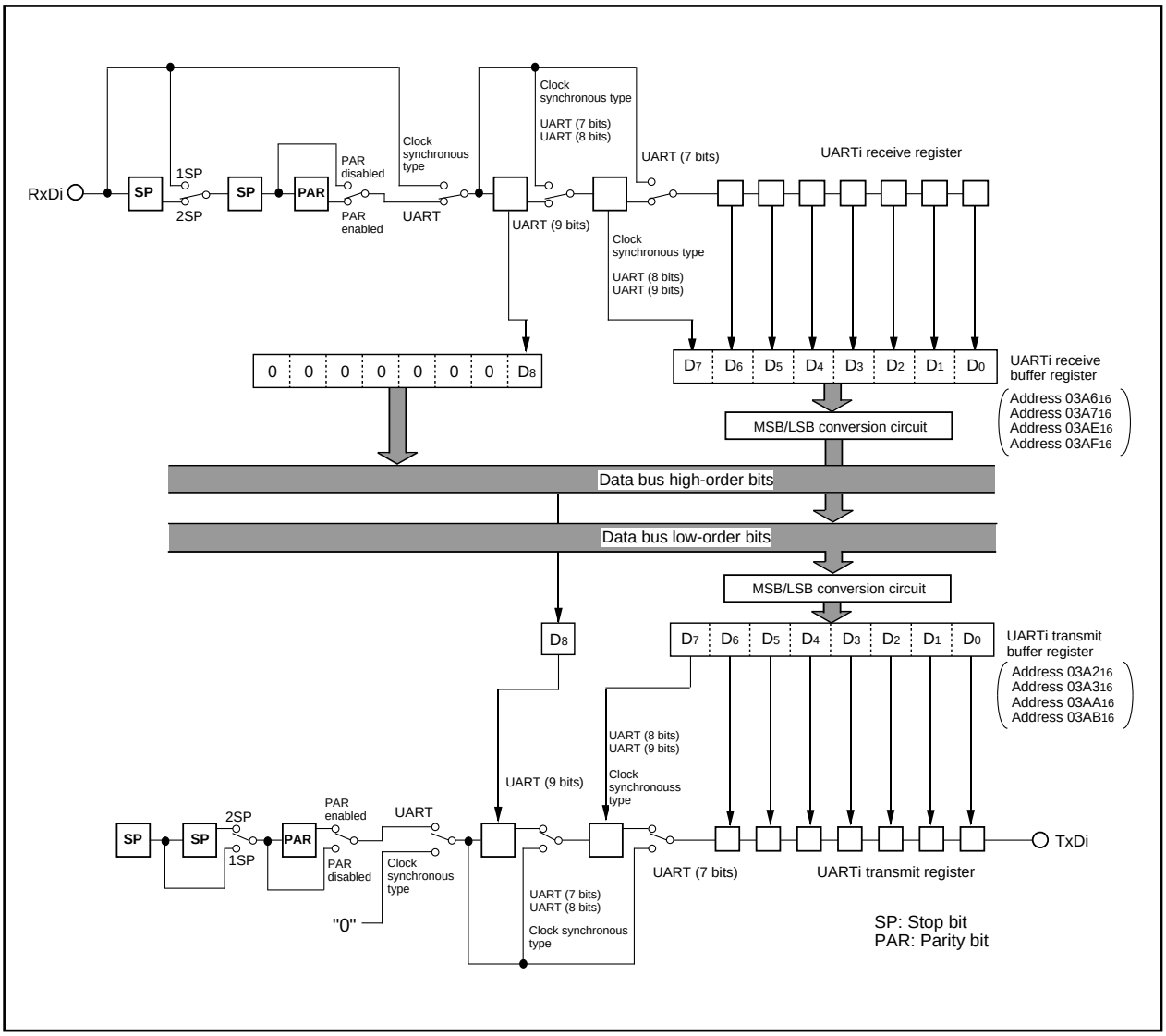
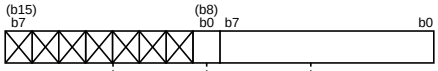


Figure 1.58. Block diagram of transmit/receive unit

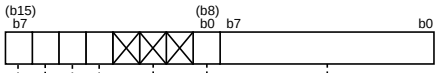
UARTi transmit buffer register



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

Function	R	W
Transmission data	×	○
Nothing is assigned. These bits can neither be set nor reset. When read, their contents are indeterminate.	–	–

UARTi receive buffer register

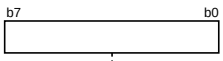


Symbol	Address	When reset
U0RB	$03A7_{16}$, $03A6_{16}$	Indeterminate
U1RB	$03AF_{16}$, $03AE_{16}$	Indeterminate

Bit symbol	Bit name	Function (During clock synchronous serial I/O mode)	Function (During UART mode)	R	W
—	———	Reception data	Reception data	○	×
Nothing is assigned. These bits can neither be set nor reset. When read, the value of these bits is 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 bit (bits 2 to 0 at addresses $03A0_{16}$ and $03A8_{16}$) are set to “000_2” or the receive enable bit is set to “0”.
(Bit 15 is set to “0” when bits 14 to 12 all are set to “0”.) Bits 14 and 13 are also set to “0” when the lower byte of the UARTi receive buffer register (addresses $03A6_{16}$ and $03AE_{16}$) is read out.

UARTi bit rate generator

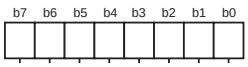


Symbol	Address	When reset
U0BRG	$03A1_{16}$	Indeterminate
U1BRG	$03A9_{16}$	Indeterminate

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

Figure 1.59. Configuration of serial I/O-related registers (1)

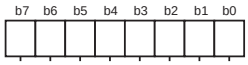
UARTi transmit/receive mode register



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

Bit symbol	Bit name	Function (During clock synchronous serial I/O mode)	Function (During UART mode)	R	W
SMD0	Serial I/O mode select bit	Must be fixed to 001	b2 b1 b0 1 0 0 : Transfer data 7 bits long 1 0 1 : Transfer data 8 bits long 1 1 0 : Transfer data 9 bits long 0 0 0 : Serial I/O invalid 0 1 0 : 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	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	○	○
SLEP	Sleep select bit	Must always be “0”	0 : Sleep mode deselected 1 : Sleep mode selected	○	○

UARTi transmit/receive control register 0



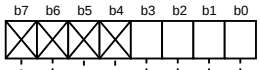
Symbol: UiC0 (i=0,1) | Address: $03A4_{16}$, $03AC_{16}$ | When reset: 08_{16}

Bit symbol	Bit name	Function (During clock synchronous serial I/O mode)	Function (During UART mode)	R	W
CLK0	BRG count source select bit	b1 b0 0 0 : f_1 is selected 0 1 : f_8 is selected 1 0 : f_{32} is selected 1 1 : Inhibited	b1 b0 0 0 : f_1 is selected 0 1 : f_8 is selected 1 0 : f_{32} is selected 1 1 : Inhibited	○	○
CLK1				○	○
CRS	$\overline{CTS}$/$\overline{RTS}$ function select bit	Valid when bit 4 = “0” 0 : $\overline{CTS}$ function is selected (Note 1) 1 : $\overline{RTS}$ function is selected (Note 2)	Valid when bit 4 = “0” 0 : $\overline{CTS}$ function is selected (Note 1) 1 : $\overline{RTS}$ function is selected (Note 2)	○	○
TXEPT	Transmit register empty flag	0 : Data present in transmit register (during transmission) 1 : No data present in transmit register (transmission completed)	0 : Data present in transmit register (during transmission) 1 : No data present in transmit register (transmission completed)	○	×
CRD	$\overline{CTS}$/$\overline{RTS}$ disable bit	0 : $\overline{CTS}$/$\overline{RTS}$ function enabled 1 : CTS/RTS function disabled (P6$_0$ and P6$_4$ function as programmable I/O port)	0 : $\overline{CTS}$/$\overline{RTS}$ function enabled 1 : CTS/RTS function disabled (P6$_0$ and P6$_4$ function as programmable I/O port)	○	○
NCH	Data output select bit	0 : TXDi pin is CMOS output 1 : TXDi pin is N-channel open-drain output	0: TXDi pin is CMOS output 1: TXDi pin is N-channel open-drain output	○	○
CKPOL	CLK polarity select bit	0 : Transmit data is output at falling edge of transfer clock and receive data is input at rising edge 1 : Transmit data is output at rising edge of transfer clock and receive data is input at falling edge	Must always be “0”	○	○
UFORM	Transfer format select bit	0 : LSB first 1 : MSB first	Must always be “0”	○	○

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

Figure 1.60. Configuration of serial I/O-related registers (2)

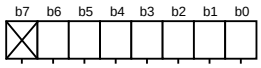
UARTi transmit/receive control register 1



Symbol: UiC1(i=0,1) — Address: $03A5_{16}$, $03AD_{16}$ — When reset: 02_{16}

Bit symbol	Bit name	Function (During clock synchronous serial I/O mode)	Function (During UART mode)	R	W
TE	Transmit enable bit	0 : Transmission disabled 1 : Transmission enabled	0 : Transmission disabled 1 : Transmission enabled	○	○
TI	Transmit buffer empty flag	0 : Data present in transmit buffer register 1 : No data present in transmit buffer register	0 : Data present in transmit buffer register 1 : No data present in transmit buffer register	○	×
RE	Receive enable bit	0 : Reception disabled 1 : Reception enabled	0 : Reception disabled 1 : Reception enabled	○	○
RI	Receive complete flag	0 : No data present in receive buffer register 1 : Data present in receive buffer register	0 : No data present in receive buffer register 1 : Data present in receive buffer register	○	×
Nothing is assigned. These bits can neither be set nor reset. When read, the value of these bits is “0”.				—	—

UART transmit/receive control register 2



Symbol: UCON — Address: $03B0_{16}$ — When reset: $X0000000_2$

Bit symbol	Bit name	Function (During clock synchronous serial I/O mode)	Function (During UART mode)	R	W
U0IRS	UART0 transmit interrupt cause select bit	0 : Transmit buffer empty (TI = 1) 1 : Transmission completed (TXEPT = 1)	0 : Transmit buffer empty (TI = 1) 1 : Transmission completed (TXEPT = 1)	○	○
U1IRS	UART1 transmit interrupt cause select bit	0 : Transmit buffer empty (TI = 1) 1 : Transmission completed (TXEPT = 1)	0 : Transmit buffer empty (TI = 1) 1 : Transmission completed (TXEPT = 1)	○	○
U0RRM	UART0 continuous receive mode enable bit	0 : Continuous receive mode disabled 1 : Continuous receive mode enable	Invalid	○	○
U1RRM	UART1 continuous receive mode enable bit	0 : Continuous receive mode disabled 1 : Continuous receive mode enabled	Invalid	○	○
CLKMD 0	CLK/CLKS select bit 0	Valid when bit 5 = “1” 0 : Clock output to CLK1 1 : Clock output to CLKS1	Invalid	○	○
CLKMD 1	CLK/CLKS select bit 1 (Note)	0 : Normal mode (CLK output is CLK1 only) 1 : Transfer clock output from multiple pins function selected	Must always be “0”	○	○
RCSP	Separate $\overline{\text{CTS}}$/$\overline{\text{RTS}}$ bit	0 : $\overline{\text{CTS}}$/$\overline{\text{RTS}}$ shared pin 1 : $\overline{\text{CTS}}$/$\overline{\text{RTS}}$ separated	0 : $\overline{\text{CTS}}$/$\overline{\text{RTS}}$ shared pin 1 : $\overline{\text{CTS}}$/$\overline{\text{RTS}}$ separated	○	○
Nothing is assigned. This bit can neither be set nor reset. When read, its content is indeterminate.				—	—

Note: When using multiple pins to output the transfer clock, the following requirement must be met:
• UART1 internal/external clock select bit (bit 3 at address $03A8_{16}$) = “0”.

Figure 1.61. Configuration of 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. Table 1.31 lists the specifications of the clock synchronous serial I/O mode. Figure 1.62 shows a configuration of the UARTi transmit/receive mode register.

Table 1.31. 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 addresses 03A0$_{16}$, 03A8$_{16}$ = "0") : fi/ 2(n+1) (Note 1) fi = f1, f8, f32 • When external clock is selected (bit 3 at addresses 03A0$_{16}$, 03A8$_{16}$ = "1") : Input from CLKi pin (Note 2)
Transmission/reception control	• $\overline{\text{CTS}}$ function/$\overline{\text{RTS}}$ function/$\overline{\text{CTS}}$, $\overline{\text{RTS}}$ function chosen to be invalid
Transmission start condition	• To start transmission, the following requirements must be met: – Transmit enable bit (bit 0 at addresses 03A5$_{16}$, 03AD$_{16}$) = "1" – Transmit buffer empty flag (bit 1 at addresses 03A5$_{16}$, 03AD$_{16}$) = "0" – When $\overline{\text{CTS}}$ function selected, $\overline{\text{CTS}}$ input level = "L" • Furthermore, if external clock is selected, the following requirements must also be met: – CLKi polarity select bit (bit 6 at addresses 03A4$_{16}$, 03AC$_{16}$) = "0": CLKi input level = "H" – CLKi polarity select bit (bit 6 at addresses 03A4$_{16}$, 03AC$_{16}$) = "1": CLKi input level = "L"
Reception start condition	• To start reception, the following requirements must be met: – Receive enable bit (bit 2 at addresses 03A5$_{16}$, 03AD$_{16}$) = "1" – Transmit enable bit (bit 0 at addresses 03A5$_{16}$, 03AD$_{16}$) = "1" – Transmit buffer empty flag (bit 1 at addresses 03A5$_{16}$, 03AD$_{16}$) = "0" • Furthermore, if external clock is selected, the following requirements must also be met: – CLKi polarity select bit (bit 6 at addresses 03A4$_{16}$, 03AC$_{16}$) = "0": CLKi input level = "H" – CLKi polarity select bit (bit 6 at addresses 03A4$_{16}$, 03AC$_{16}$) = "1": CLKi input level = "L"
Interrupt request generation timing	• When transmitting – Transmit interrupt cause select bit (bits 0, 1 at address 03B0$_{16}$) = "0": Interrupts requested when data transfer from UARTi transfer buffer register to UARTi transmit register is completed – Transmit interrupt cause select bit (bits 0, 1 at address 03B0$_{16}$) = "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
Select function	• 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 (Note 4) UART1 transfer clock can be chosen by software to be output from one of the two pins set • Separate $\overline{\text{CTS}}$/$\overline{\text{RTS}}$ pins (Note 4) UART0's $\overline{\text{CTS}}$ and $\overline{\text{RTS}}$ pins each can be assigned to separate pins

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

Note 2: Maximum 5 Mbps.

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 is not set to "1".

Note 4: The transfer clock output from multiple pins and the separate $\overline{\text{CTS}}$/$\overline{\text{RTS}}$ pins functions cannot be selected simultaneously.

UARTi transmit/receive mode register

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

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

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

Figure 1.62. Configuration of UARTi transmit/receive mode register in clock synchronous serial I/O mode

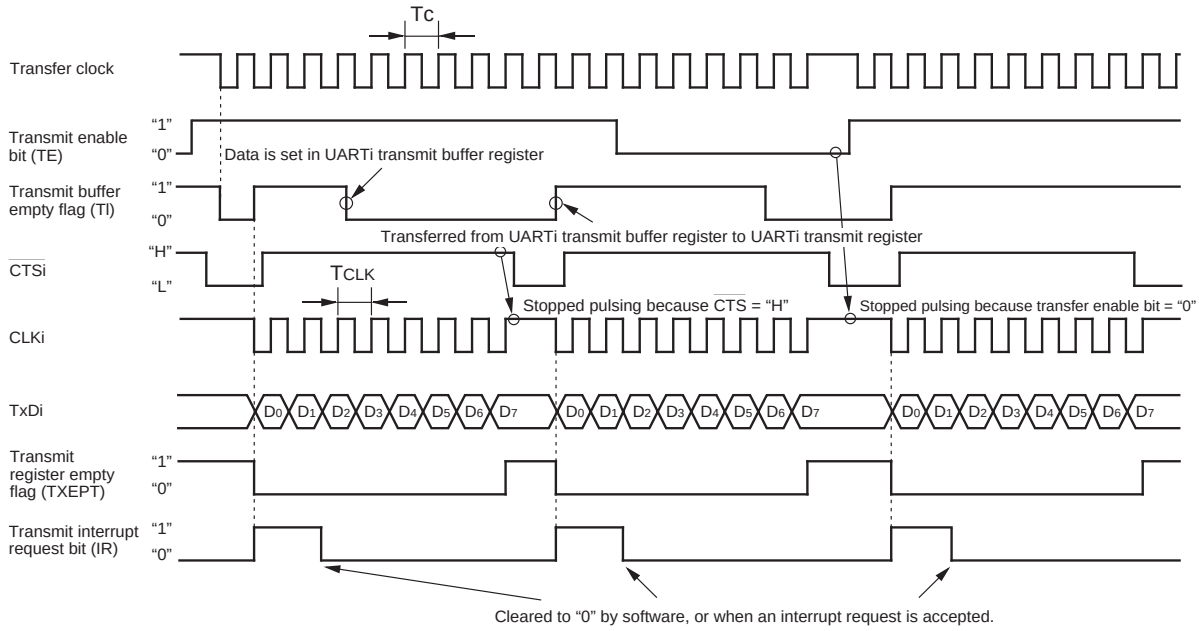
Table 1.32 lists the functions of the input/output pins during clock synchronous serial I/O mode. This table shows the pin functions when the transfer clock output from multiple pins and the separate $\overline{CTS}$/$\overline{RTS}$ pins functions are not selected. Note that for a period from when the 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 1.32. Input/output pin functions in clock synchronous serial I/O mode

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

(When transfer clock output from multiple pins and separate $\overline{CTS}$/$\overline{RTS}$ pins functions are not selected)

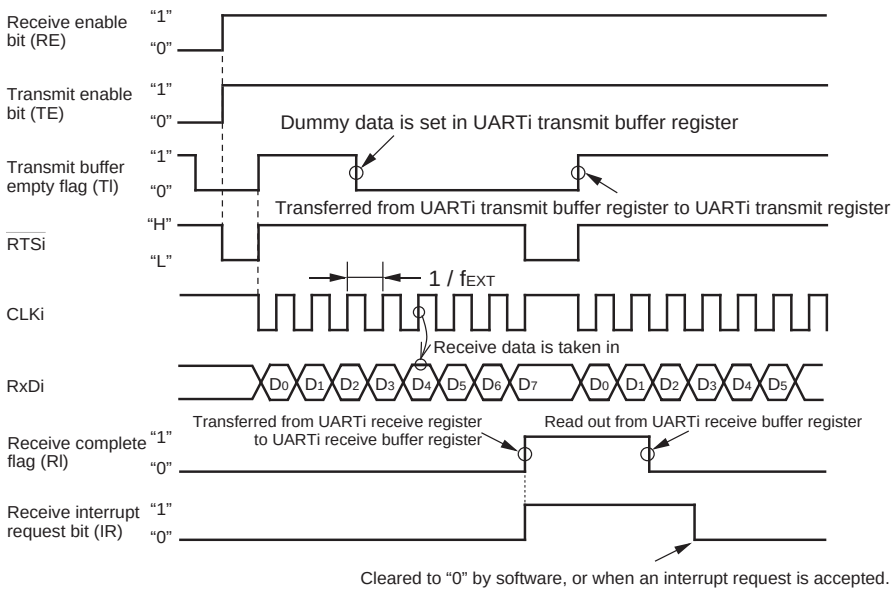
• Example of transmit timing (when internal clock is selected)



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

$Tc = T_{CLK} = 2(n + 1) / fi$
fi: frequency of BRGi's count source (f1, f8, f32)
n: value set to BRGi

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



fEXT: frequency of external clock

Shown in ( ) are bit symbols.
The above timing applies to the following settings.
- External clock is selected.
- RTS function is selected.
- CLK polarity select bit = "0".

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

Figure 1.63. Typical transmit/receive timings in clock synchronous serial I/O mode

(a) Polarity select function

As shown in Figure 1.64, the CLK polarity select bit (bit 6 at addresses 03A4$_{16}$, 03AC$_{16}$) allows selection of the polarity of the transfer clock.

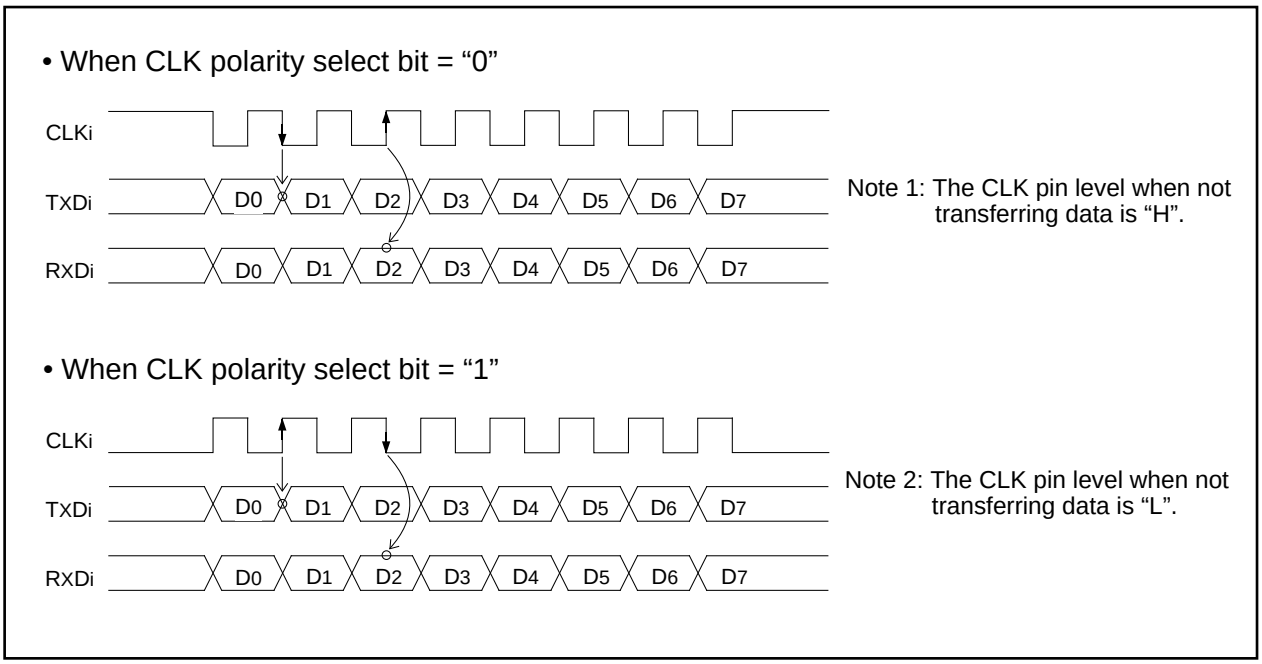


Figure 1. 64. Polarity of transfer clock

(b) LSB first/MSB first select function

As shown in Figure 1.65, when the transfer format select bit (bit 7 at addresses 03A4$_{16}$, 03AC$_{16}$) = “0”, the transfer format is “LSB first”; when the bit = “1”, the transfer format is “MSB first”.

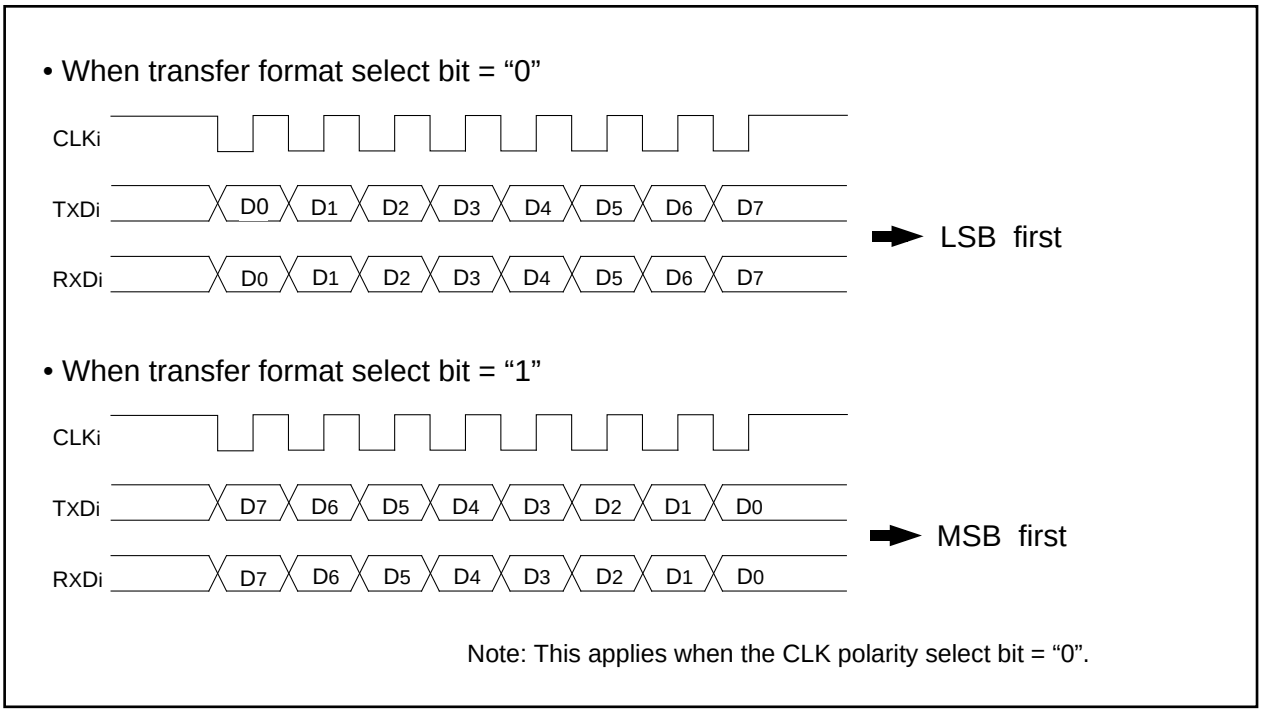


Figure 1. 65. Transfer format

(c) Transfer clock output from multiple pins function

This function allows the setting two transfer clock output pins and choosing one of the two to output a clock by using the CLK and CLKS select bit (bits 4 and 5 at address 03B0$_{16}$). (See Figure 1.66.) The multiple pins function is valid only when the internal clock is selected for UART1. Note that when this function is selected, UART1 $\overline{\text{CTS}}$/$\overline{\text{RTS}}$ function cannot be used.

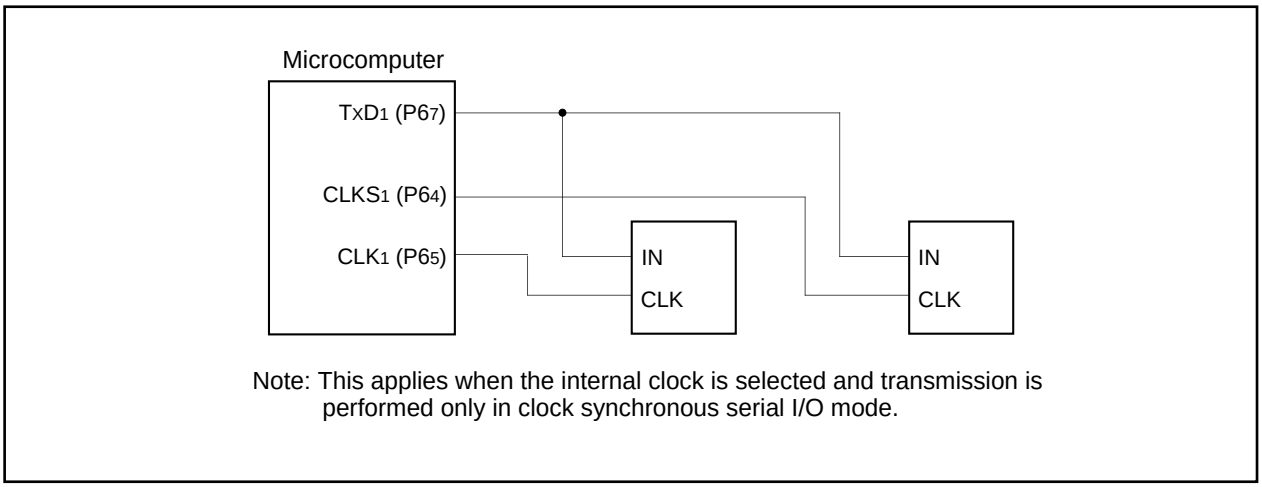


Figure 1. 66. The transfer clock output from the multiple pins function usage

(d) Continuous receive mode

If the continuous receive mode enable bit (bits 2 and 3 at address 03B0$_{16}$) is set to "1", the unit is placed in the 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) Separate $\overline{\text{CTS}}$/$\overline{\text{RTS}}$ pins function

This function works the same way as in the clock asynchronous serial I/O (UART) mode. The method of setting and the input/output pin functions are both the same, so refer to select function in the next section, "(2) Clock asynchronous serial I/O (UART) mode." Note that this function is invalid if the transfer clock output from the multiple pins function is selected.

(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. Table 1.33 lists the specifications of the UART mode. Figure 1.67 shows the configuration of the UARTi transmit/receive mode register.

Table 1.33. 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 $03A0_{16}$, $03A8_{16}$ = "0") : fi/16 (n+1) (Note 1) fi = f_1, f_8, f_{32} • When external clock is selected (bit 3 at addresses $03A0_{16}$, $03A8_{16}$="1") : f_{EXT}/16 (n+1) (Note 1) (Note 2)
Transmission/reception control	• $\overline{CTS}$ function/$\overline{RTS}$ function/$\overline{CTS}$, $\overline{RTS}$ function chosen to be invalid
Transmission start condition	• To start transmission, the following requirements must be met: - Transmit enable bit (bit 0 at addresses $03A5_{16}$, $03AD_{16}$) = "1" - Transmit buffer empty flag (bit 1 at addresses $03A5_{16}$, $03AD_{16}$) = "0" - When $\overline{CTS}$ function selected, $\overline{CTS}$ input level = "L"
Reception start condition	• To start reception, the following requirements must be met: - Receive enable bit (bit 2 at addresses $03A5_{16}$, $03AD_{16}$) = "1" - Start bit detection
Interrupt request generation timing	• When transmitting - Transmit interrupt cause select bits (bits 0,1 at address $03B0_{16}$) = "0": Interrupts requested when data transfer from UARTi transfer buffer register to UARTi transmit register is completed - Transmit interrupt cause select bits (bits 0, 1 at address $03B0_{16}$) = "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	• Separate $\overline{CTS}$/$\overline{RTS}$ pins UART0's $\overline{CTS}$ and $\overline{RTS}$ pins can each be assigned to separate pins • Sleep mode selection This mode is used to transfer data to and from one of multiple slave microcomputers

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 is not set to "1".

UARTi transmit/receive mode register

b7	b6	b5	b4	b3	b2	b1	b0

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

Bit symbol	Bit name	Function	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	○	○
STPS	Stop bit length select bit	0 : One stop bit 1 : Two stop bits	○	○
PRY	Odd/even parity select bit	Valid when bit 6 = "1" 0 : Odd parity 1 : Even parity	○	○
PRYE	Parity enable bit	0 : Parity disabled 1 : Parity enabled	○	○
SLEP	Sleep select bit	0 : Sleep mode deselected 1 : Sleep mode selected	○	○

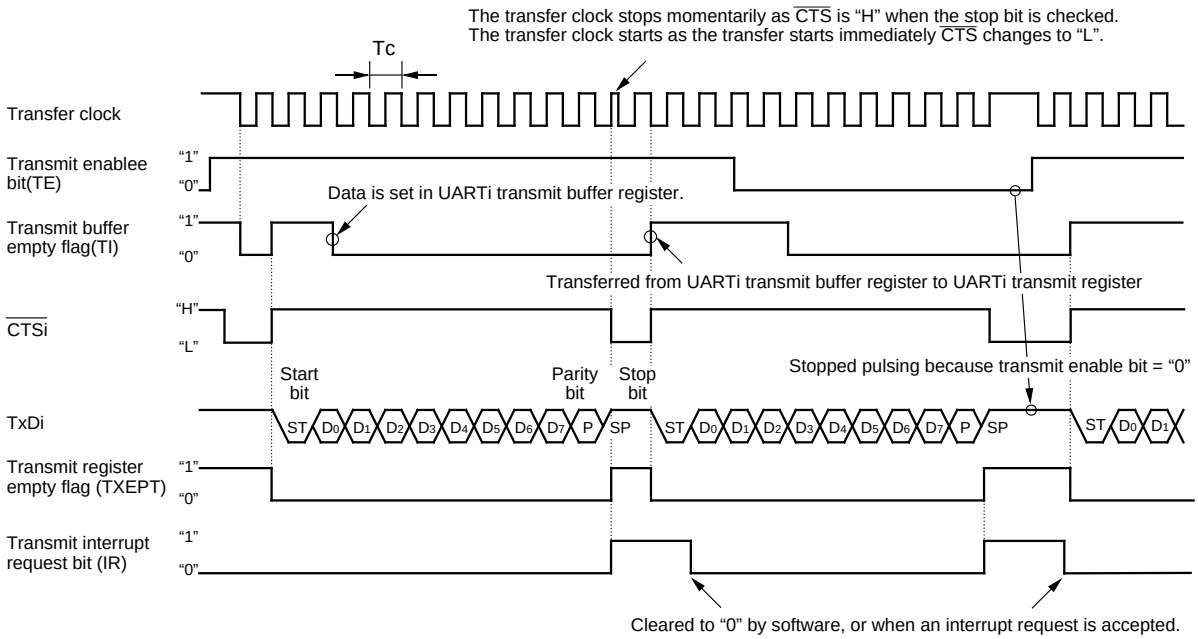
Figure 1.67. Configuration of UARTi transmit/receive mode register in UART mode

Table 1.34 lists the functions of the input/output pins during UART mode. This table shows the pin functions when the separate $\overline{CTS}/\overline{RTS}$ pins function is not selected. Note that for a period from when the UARTi operation mode is selected to when transfer starts, the TxDi pin outputs a "H". (If the N-channel open-drain is selected, this pin is in floating state.)

Table 1.34. Input/output pin functions in UART mode (when separate $\overline{CTS}/\overline{RTS}$ pins function is not selected)

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

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

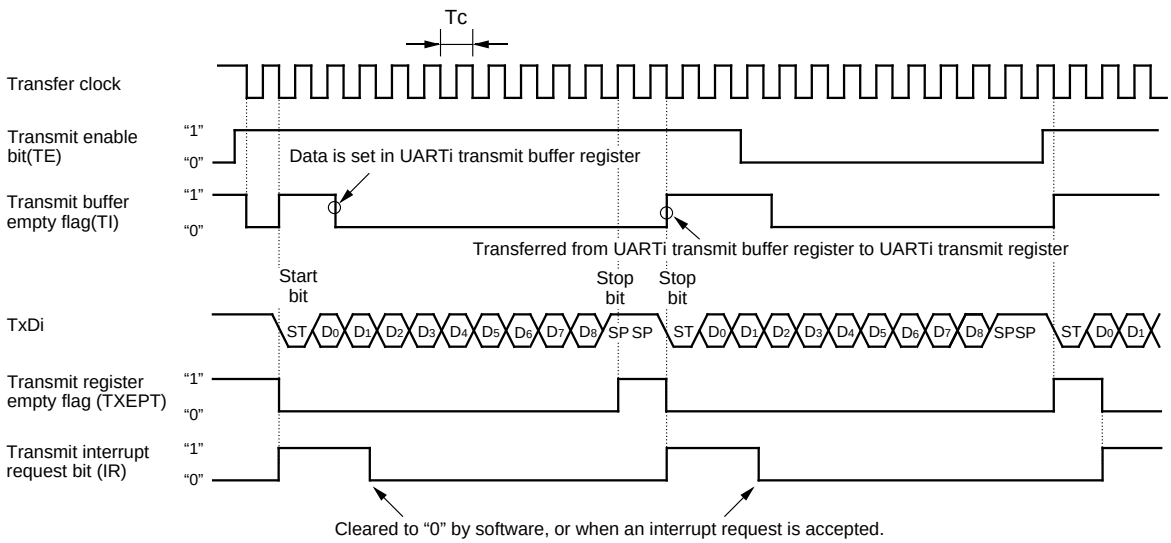


Shown in ( ) are bit symbols.

The above timing applies to the following settings :
- Parity is enabled.
- One stop bit.
- CTS function is selected.
- Transmit interrupt cause select bit = "1".

Tc = 16 (n + 1) / fi or 16 (n + 1) / f_{EXT}
fi : frequency of BRGi's count source (f_1, f_8, f_{32})
f_{EXT} : frequency of BRGi's count source (external clock)
n : value set to BRGi

• Example of transmit timing when transfer data is 9 bits long (parity disabled, two stop bits)



Shown in ( ) are bit symbols.

The above timing applies to the following settings :
- Parity is disabled.
- Two stop bits.
- CTS function is disabled.
- Transmit interrupt causes select bit = "0".

Tc = 16 (n + 1) / fi or 16 (n + 1) / f_{EXT}
fi : frequency of BRGi's count source (f_1, f_8, f_{32})
f_{EXT} : frequency of BRGi's count source (external clock)
n : value set to BRGi

Figure 1.68. Typical transmit timings in UART mode

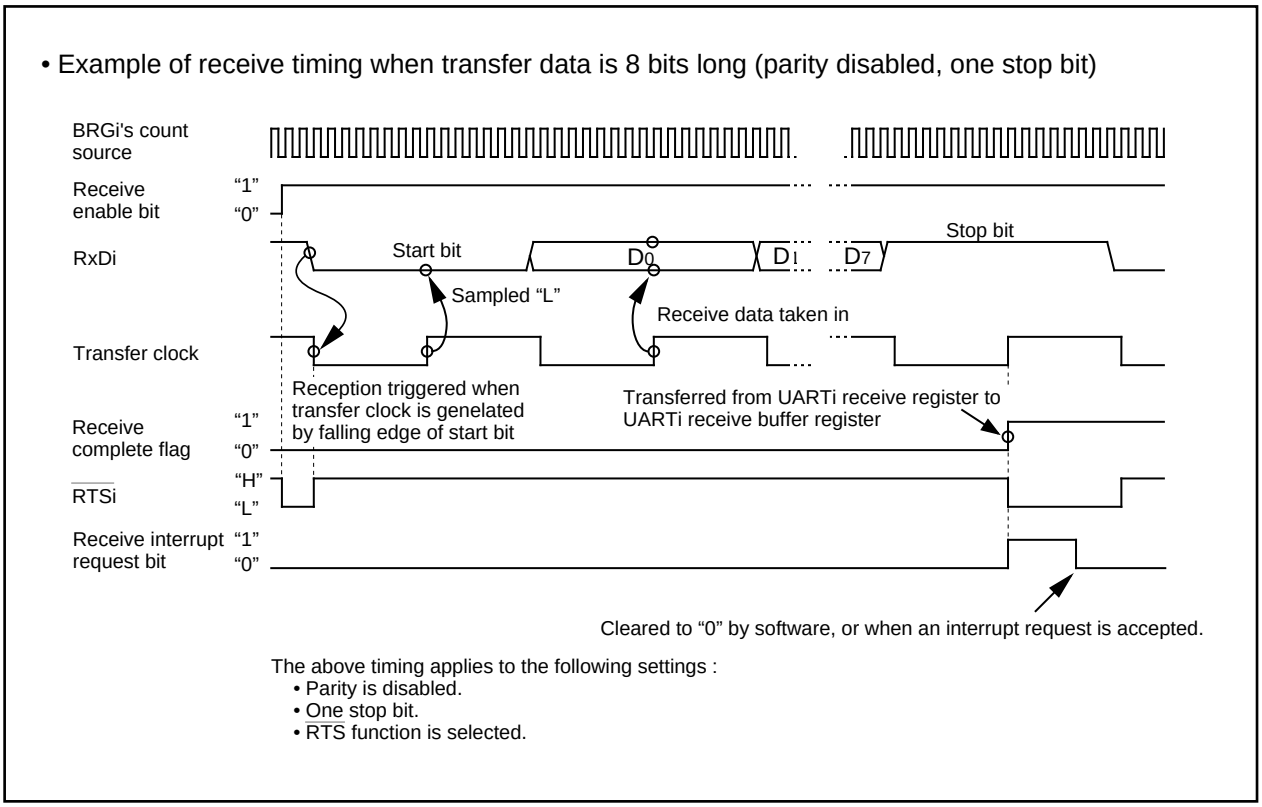


Figure 1.69. Typical receive timing in UART mode

(a) Separate CTS/RTS pins function

With the separate CTS/RTS bit (bit 6 at address 03B0$_{16}$) is set to "1", the unit outputs/inputs the CTS and RTS signals on different pins. (See Figure 1.70.) This function is valid only for UART0. Note that if this function is selected, the CTS/RTS function for UART1 cannot be used.

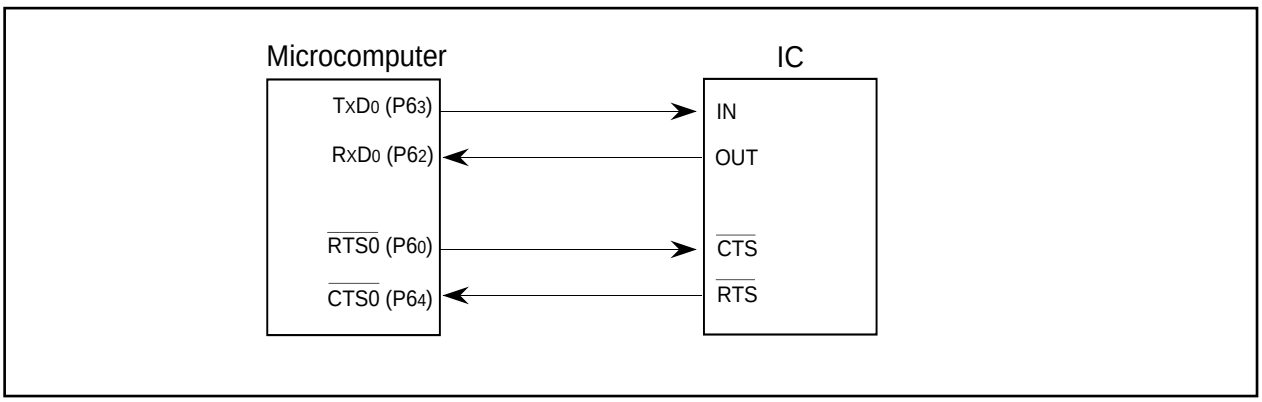


Figure 1.70. Example for the separate CTS/RTS pins function usage

(b) Sleep mode

This mode is used to transfer data between specific microcomputers among multiple microcomputers connected using UARTi. The sleep mode is selected when the sleep select bit (bit 7 at addresses 03A0$_{16}$, 03A8$_{16}$) is set to "1" during reception. In this mode, the unit performs receive operation when the MSB of the received data = "1" and does not perform receive operation when the MSB = "0".

A-D Converter

The A-D converter consists of one 10-bit successive approximation A-D converter circuit with a capacitive coupling amplifier. Pins $P10_0$ to $P10_7$, $P9_5$, and $P9_6$ also function as the analog signal input pins. The direction registers of these pins for A-D conversion must therefore be set to input. The Vref connect bit (bit 5 at address $03D7_{16}$) can be used to isolate the resistance ladder of the A-D converter from the reference voltage input pin (V_{REF}) when the A-D converter is not used. Doing so stops any current flowing into the resistance ladder from V_{REF}, reducing the power dissipation. When using the A-D converter, start A-D conversion only after setting bit 5 of $03D7_{16}$ to connect V_{REF}.

The result of A-D conversion is stored in the A-D registers of the selected pins. When set to 10-bit precision, the low 8 bits are stored in the even addresses and the high 2 bits in the odd addresses. When set to 8-bit precision, the low 8 bits are stored in the even addresses.

Table 1.35 shows the performance of the A-D converter. Figure 1.71 shows a block diagram of the A-D converter, and Figures 1.72 and 1.73 show configurations of the A-D converter-related registers.

Table 1.35. Performance of A-D converter

Item	Performance
Method of A-D conversion	Successive approximation (capacitive coupling amplifier)
Analog input voltage (Note 1)	0V to AV_{CC} (V_{CC})
Operating clock ϕ_{AD} (Note 2)	V_{CC} = 5V f_{AD}/divide-by-2 of f_{AD}/divide-by-4 of f_{AD}, $f_{AD}=f(X_{IN})$ V_{CC} = 3V divide-by-2 of f_{AD}/divide-by-4 of f_{AD}, $f_{AD}=f(X_{IN})$
Resolution	8-bit or 10-bit (selectable)
Absolute precision	V_{CC} = 5V • Without sample and hold function ±3LSB • With sample and hold function (8-bit resolution) ±2LSB • With sample and hold function (10-bit resolution) AN_0 to AN_7 input : ±3LSB ANEX0 and ANEX1 input (including mode in which external operation amp is connected) : ±7LSB V_{CC} = 3V • Without sample and hold function (8-bit resolution) ±2LSB
Operating modes	One-shot mode, repeat mode, single sweep mode, repeat sweep mode 0, and repeat sweep mode 1
Analog input pins	8pins (AN_0 to AN_7) + 2pins (ANEX0 and ANEX1)
A-D conversion start condition	• Software trigger A-D conversion starts when the A-D conversion start flag changes to “1” • External trigger (can be retriggered) A-D conversion starts when the A-D conversion start flag is “1” and the $\overline{ADTRG}$/$P9_7$ input changes from “H” to “L”
Conversion speed per pin	• Without sample and hold function 8-bit resolution: 49 ϕ_{AD} cycles, 10-bit resolution: 59 ϕ_{AD} cycles • With sample and hold function 8-bit resolution: 28 ϕ_{AD} cycles, 10-bit resolution: 33 ϕ_{AD} cycles

Note 1: Does not depend on use of sample and hold function.

Note 2: Without sample and hold function, set the ϕ_{AD} frequency to 250kHz min.
With the sample and hold function, set the ϕ_{AD} frequency to 1MHz min.

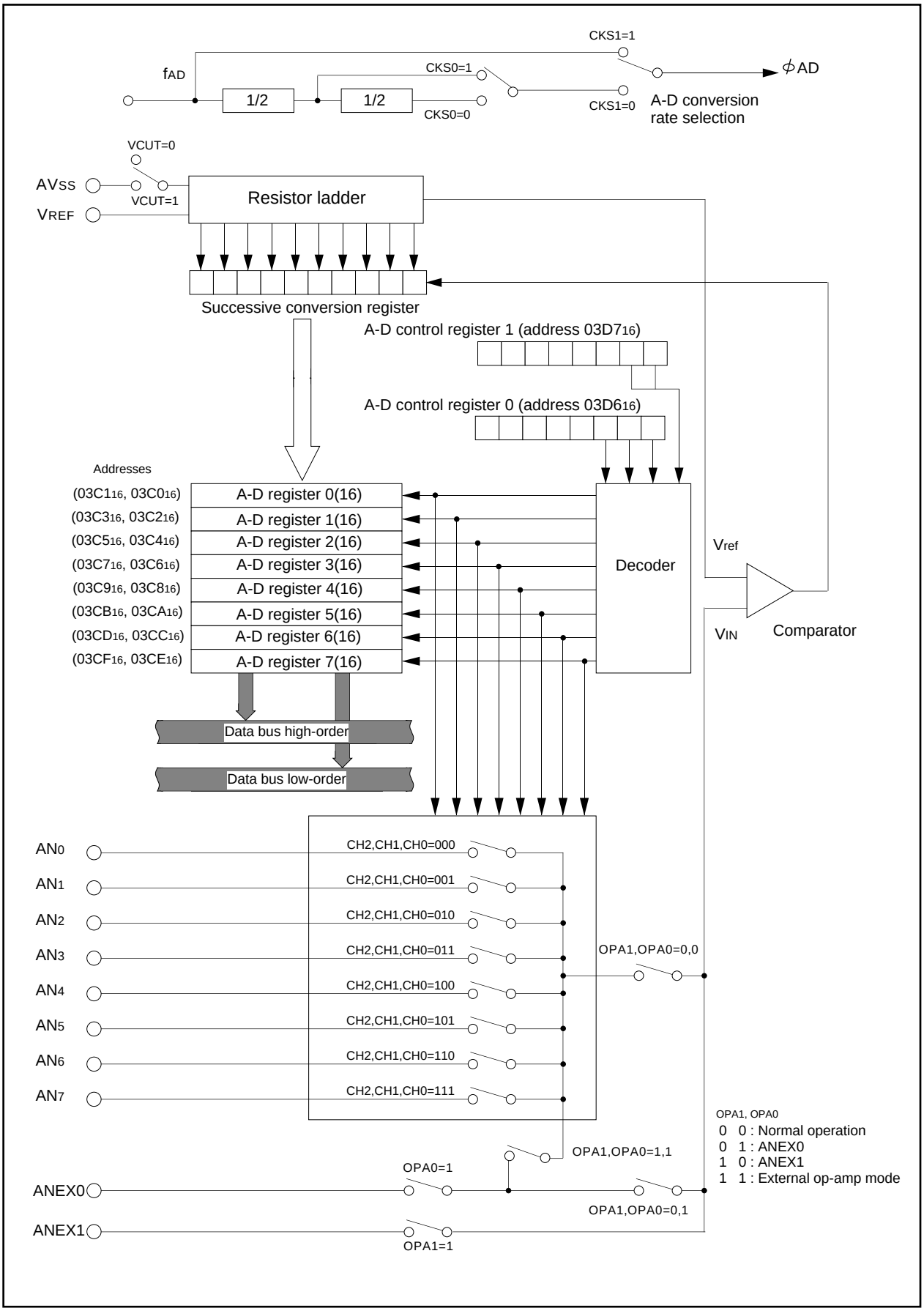


Figure 1.71. Block diagram of A-D converter

A-D control register 0 (Note)

b7	b6	b5	b4	b3	b2	b1	b0

Symbol	Address	When reset
ADCON0	$03D6_{16}$	$00000XXX_2$

Bit symbol	Bit name	Function	R	W
CH0	Analog input pin select bit	b2 b1 b0 0 0 0 : AN_0 is selected 0 0 1 : AN_1 is selected 0 1 0 : AN_2 is selected 0 1 1 : AN_3 is selected 1 0 0 : AN_4 is selected 1 0 1 : AN_5 is selected 1 1 0 : AN_6 is selected 1 1 1 : AN_7 is selected	○	○
CH1			○	○
CH2			○	○
MD0	A-D operation mode select bit 0	b4 b3 0 0 : One-shot mode 0 1 : Repeat mode 1 0 : Single sweep mode 1 1 : Repeat sweep mode 0 Repeat sweep mode 1	○	○
MD1			○	○
TRG	Trigger select bit	0 : Software trigger 1 : AD_{TRG} trigger	○	○
ADST	A-D conversion start flag	0 : A-D conversion disabled 1 : A-D conversion started	○	○
CKS0	Frequency select bit 0	0 : $f_{AD}/4$ is selected 1 : $f_{AD}/2$ is selected	○	○

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

A-D control register 1 (Note)

b7	b6	b5	b4	b3	b2	b1	b0

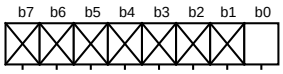
Symbol	Address	When reset
ADCON1	$03D7_{16}$	00_{16}

Bit symbol	Bit name	Function	R	W
SCAN0	A-D sweep pin select bit	When single sweep and repeat sweep mode 0 are selected b1 b0 0 0 : AN_0, AN_1 (2 pins) 0 1 : AN_0 to AN_3 (4 pins) 1 0 : AN_0 to AN_5 (6 pins) 1 1 : AN_0 to AN_7 (8 pins)	○	○
SCAN1		When repeat sweep mode 1 is selected b1 b0 0 0 : AN_0 (1 pin) 0 1 : AN_0, AN_1 (2 pins) 1 0 : AN_0 to AN_2 (3 pins) 1 1 : AN_0 to AN_3 (4 pins)	○	○
MD2	A-D operation mode select bit 1	0 : Any mode other than repeat sweep mode 1 1 : Repeat sweep mode 1	○	○
BITS	8/10-bit mode select bit	0 : 8-bit mode 1 : 10-bit mode	○	○
CKS1	Frequency select bit 1	0 : $f_{AD}/2$ or $f_{AD}/4$ is selected 1 : f_{AD} is selected	○	○
VCUT	Vref connect bit	0 : Vref not connected 1 : Vref connected	○	○
OPA0	External op-amp connection mode bit	b7 b6 0 0 : ANEX0 and ANEX1 are not used 0 1 : ANEX0 input is A-D converted 1 0 : ANEX1 input is A-D converted 1 1 : External op-amp connection mode	○	○
OPA1			○	○

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

Figure 1.72. Configuration of A-D converter-related registers (1)

A-D control register 2 (Note)



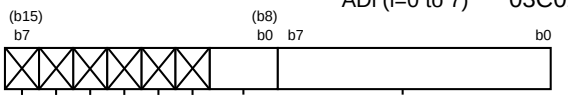
Symbol: ADCON2 | Address: $03D4_{16}$ | When reset: $XXXXXXX0_2$

Bit symbol	Bit name	Function	R	W
SMP	A-D conversion method select bit	0 Without sample and hold 1 With sample and hold	○	○
Nothing is assigned. These bits can neither be set nor reset. When read, their content is “0”.			–	–

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

A-D register i

Symbol: ADi (i=0 to 7) | Address: $03C0_{16}$ to $03CF_{16}$ | When reset: Indeterminate



Function	R	W
Eight low-order bits of A-D conversion result	○	×
• During 10-bit mode Two high-order bits of A-D conversion result	○	×
• During 8-bit mode When read, the content is indeterminate	×	×
Nothing is assigned. These bits can neither be set nor reset. When read, their content is “0”.	–	–

Figure 1.73. Configuration of A-D converter-related registers (2)

(1) One-shot mode

In one-shot mode, the pin selected using the analog input pin select bit is used for one-shot A-D conversion. Table 1.36 shows the specifications of one-shot mode. Figure 1.74 shows the configuration of the A-D control register in one-shot mode.

Table 1.36. One-shot mode specifications

Item	Specification
Function	The pin selected by the analog input pin select bit is used for one A-D conversion
Start condition	Writing "1" to A-D conversion start flag
Stop condition	• End of A-D conversion (A-D conversion start flag changes to "0", except when external trigger is selected) • Writing "0" to A-D conversion start flag
Interrupt request generation timing	End of A-D conversion
Input pin	One of AN_0 to AN_7, as selected
Reading of result of A-D converter	Read A-D register corresponding to selected pin

A-D control register 0 (Note)

b7	b6	b5	b4	b3	b2	b1	b0
			0	0			

Symbol: ADCON0 — Address: $03D6_{16}$ — When reset: $00000XXX_2$

Bit symbol	Bit name	Function	R	W
CH0	Analog input pin select bit	b2 b1 b0 0 0 0 : AN_0 is selected 0 0 1 : AN_1 is selected 0 1 0 : AN_2 is selected 0 1 1 : AN_3 is selected 1 0 0 : AN_4 is selected 1 0 1 : AN_5 is selected 1 1 0 : AN_6 is selected 1 1 1 : AN_7 is selected	○	○
CH1			○	○
CH2			○	○
MD0	A-D operation mode select bit 0	b4 b3 0 0 : One-shot mode	○	○
MD1			○	○
TRG	Trigger select bit	0 : Software trigger 1 : AD_{TRG} trigger	○	○
ADST	A-D conversion start flag	0 : A-D conversion disabled 1 : A-D conversion started	○	○
CKS0	Frequency select bit 0	0: $f_{AD}/4$ is selected 1: $f_{AD}/2$ is selected	○	○

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

A-D control register 1 (Note)

b7	b6	b5	b4	b3	b2	b1	b0
		1			0		

Symbol: ADCON1 — Address: $03D7_{16}$ — When reset: 00_{16}

Bit symbol	Bit name	Function	R	W
SCAN0	A-D sweep pin select bit	Invalid in one-shot mode	○	○
SCAN1			○	○
MD2	A-D operation mode select bit 1	0 : Any mode other than repeat sweep mode 1	○	○
BITS	8/10-bit mode select bit	0 : 8-bit mode 1 : 10-bit mode	○	○
CKS1	Frequency select bit1	0 : $f_{AD}/2$ or $f_{AD}/4$ is selected 1 : f_{AD} is selected	○	○
VCUT	Vref connect bit	1 : Vref connected	○	○
OPA0	External op-amp connection mode bit	b7 b6 0 0 : ANEX0 and ANEX1 are not used 0 1 : ANEX0 input is A-D converted 1 0 : ANEX1 input is A-D converted 1 1 : External op-amp connection mode	○	○
OPA1			○	○

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

Figure 1.74. Configuration of A-D conversion register in one-shot mode

(2) Repeat mode

In repeat mode, the pin selected using the analog input pin select bit is used for repeated A-D conversion. Table 1.37 shows the specifications of repeat mode. Figure 1.75 shows the configuration of the A-D control register in repeat mode.

Table 1.37. 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_7, as selected
Reading of result of A-D converter	Read A-D register corresponding to selected pin

A-D control register 0 (Note)

b7	b6	b5	b4	b3	b2	b1	b0
			0	1			

Symbol: ADCON0 — Address: $03D6_{16}$ — When reset: $00000XXX_2$

Bit symbol	Bit name	Function	R	W
CH0	Analog input pin select bit	b2 b1 b0 0 0 0 : AN_0 is selected 0 0 1 : AN_1 is selected 0 1 0 : AN_2 is selected 0 1 1 : AN_3 is selected 1 0 0 : AN_4 is selected 1 0 1 : AN_5 is selected 1 1 0 : AN_6 is selected 1 1 1 : AN_7 is selected	○	○
CH1			○	○
CH2			○	○
MD0	A-D operation mode select bit 0	b4 b3 0 1 : Repeat mode	○	○
MD1			○	○
TRG	Trigger select bit	0 : Software trigger 1 : $\overline{AD_{TRG}}$ trigger	○	○
ADST	A-D conversion start flag	0 : A-D conversion disabled 1 : A-D conversion started	○	○
CKS0	Frequency select bit 0	0 : $f_{AD}/4$ is selected 1 : $f_{AD}/2$ is selected	○	○

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

A-D control register 1 (Note)

b7	b6	b5	b4	b3	b2	b1	b0
		1			0		

Symbol: ADCON1 — Address: $03D7_{16}$ — When reset: 00_{16}

Bit symbol	Bit name	Function	R	W
SCAN0	A-D sweep pin select bit	Invalid in repeat mode	○	○
SCAN1			○	○
MD2	A-D operation mode select bit 1	0 : Any mode other than repeat sweep mode 1	○	○
BITS	8/10-bit mode select bit	0 : 8-bit mode 1 : 10-bit mode	○	○
CKS1	Frequency select bit 1	0 : $f_{AD}/2$ or $f_{AD}/4$ is selected 1 : f_{AD} is selected	○	○
VCUT	Vref connect bit	1 : Vref connected	○	○
OPA0	External op-amp connection mode bit	b7 b6 0 0 : ANEX0 and ANEX1 are not used 0 1 : ANEX0 input is A-D converted 1 0 : ANEX1 input is A-D converted 1 1 : External op-amp connection mode	○	○
OPA1			○	○

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

Figure 1.75. Configuration of A-D conversion register in repeat mode

(3) Single sweep mode

In single sweep mode, the pins selected using the A-D sweep pin select bit are used for one-by-one A-D conversion. Table 1.38 shows the specifications of single sweep mode. Figure 1.76 shows the configuration of the A-D control register in single sweep mode.

Table 1.38. Single sweep mode specifications

Item	Specification
Function	The pins selected by the A-D sweep pin select bit are used for one-by-one A-D conversion
Start condition	Writing "1" to A-D converter start flag
Stop condition	• End of A-D conversion (A-D conversion start flag changes to "0", except when external trigger is selected) • Writing "0" to A-D conversion start flag
Interrupt request generation timing	End of A-D conversion
Input pin	AN_0 and AN_1 (2 pins), AN_0 to AN_3 (4 pins), AN_0 to AN_5 (6 pins), or AN_0 to AN_7 (8 pins)
Reading of result of A-D converter	Read A-D register corresponding to selected pin

A-D control register 0 (Note)

b7	b6	b5	b4	b3	b2	b1	b0
			1	0			

Symbol ADCON0 — Address $03D6_{16}$ — When reset $00000XXX_2$

Bit symbol	Bit name	Function	R	W
CH0	Analog input pin select bit	Invalid in single sweep mode	○	○
CH1			○	○
CH2			○	○
MD0	A-D operation mode select bit 0	b4 b3 1 0 : Single sweep mode	○	○
MD1			○	○
TRG	Trigger select bit	0 : Software trigger 1 : $\overline{ADTRG}$ trigger	○	○
ADST	A-D conversion start flag	0 : A-D conversion disabled 1 : A-D conversion started	○	○
CKS0	Frequency select bit 0	0 : $f_{AD}/4$ is selected 1 : $f_{AD}/2$ is selected	○	○

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

A-D control register 1 (Note 1)

b7	b6	b5	b4	b3	b2	b1	b0
		1			0		

Symbol ADCON1 — Address $03D7_{16}$ — When reset 00_{16}

Bit symbol	Bit name	Function	R	W
SCAN0	A-D sweep pin select bit	When single sweep and repeat sweep mode 0 are selected b1 b0 0 0 : AN_0, AN_1 (2 pins) 0 1 : AN_0 to AN_3 (4 pins) 1 0 : AN_0 to AN_5 (6 pins) 1 1 : AN_0 to AN_7 (8 pins)	○	○
SCAN1			○	○
MD2	A-D operation mode select bit 1	0 : Any mode other than repeat sweep mode 1	○	○
BITS	8/10-bit mode select bit	0 : 8-bit mode 1 : 10-bit mode	○	○
CKS1	Frequency select bit 1	0 : $f_{AD}/2$ or $f_{AD}/4$ is selected 1 : f_{AD} is selected	○	○
VCUT	Vref connect bit	1 : Vref connected	○	○
OPA0	External op-amp connection mode bit (Note 2)	b7 b6 0 0 : ANEX0 and ANEX1 are not used 0 1 : ANEX0 input is A-D converted 1 0 : ANEX1 input is A-D converted 1 1 : External op-amp connection mode	○	○
OPA1			○	○

Note 1: If the A-D control register is rewritten during A-D conversion, the conversion result is indeterminate.
Note 2: Neither '01' nor '10' can be selected with the external op-amp connection mode bit.

Figure 1.76. Configuration of A-D conversion register in single sweep mode

(4) Repeat sweep mode 0

In repeat sweep mode 0, the pins selected using the A-D sweep pin select bit are used for repeat sweep A-D conversion. Table 1.39 shows the specifications of repeat sweep mode 0. Figure 1.77 shows the configuration of the A-D control register in repeat sweep mode 0.

Table 1.39. Repeat sweep mode 0 specifications

Item	Specification
Function	The pins selected by the A-D sweep pin select bit are used for repeat sweep A-D conversion
Start condition	Writing "1" to A-D conversion start flag
Stop condition	Writing "0" to A-D conversion start flag
Interrupt request generation timing	None generated
Input pin	AN_0 and AN_1 (2 pins), AN_0 to AN_3 (4 pins), AN_0 to AN_5 (6 pins), or AN_0 to AN_7 (8 pins)
Reading of result of A-D converter	Read A-D register corresponding to selected pin (at any time)

A-D control register 0 (Note)

b7	b6	b5	b4	b3	b2	b1	b0
			1	1			

Symbol: ADCON0 — Address: $03D6_{16}$ — When reset: $00000XXX_2$

Bit symbol	Bit name	Function	R	W
CH0	Analog input pin select bit	Invalid in repeat sweep mode 0	○	○
CH1			○	○
CH2			○	○
MD0	A-D operation mode select bit 0	b4 b3 1 1 : Repeat sweep mode 0	○	○
MD1			○	○
TRG	Trigger select bit	0 : Software trigger 1 : AD_{TRG} trigger	○	○
ADST	A-D conversion start flag	0 : A-D conversion disabled 1 : A-D conversion started	○	○
CKS0	Frequency select bit 0	0 : $f_{AD}/4$ is selected 1 : $f_{AD}/2$ is selected	○	○

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

A-D control register 1 (Note 1)

b7	b6	b5	b4	b3	b2	b1	b0
		1			0		

Symbol: ADCON1 — Address: $03D7_{16}$ — When reset: 00_{16}

Bit symbol	Bit name	Function	R	W
SCAN0	A-D sweep pin select bit	When single sweep and repeat sweep mode 0 are selected b1 b0 0 0 : AN_0, AN_1 (2 pins) 0 1 : AN_0 to AN_3 (4 pins) 1 0 : AN_0 to AN_5 (6 pins) 1 1 : AN_0 to AN_7 (8 pins)	○	○
SCAN1			○	○
MD2	A-D operation mode select bit 1	0 : Any mode other than repeat sweep mode 1	○	○
BITS	8/10-bit mode select bit	0 : 8-bit mode 1 : 10-bit mode	○	○
CKS1	Frequency select bit 1	0 : $f_{AD}/2$ or $f_{AD}/4$ is selected 1 : f_{AD} is selected	○	○
VCUT	Vref connect bit	1 : Vref connected	○	○
OPA0	External op-amp connection mode bit (Note 2)	b7 b6 0 0 : ANEX0 and ANEX1 are not used 0 1 : ANEX0 input is A-D converted 1 0 : ANEX1 input is A-D converted 1 1 : External op-amp connection mode	○	○
OPA1			○	○

Note 1: If the A-D control register is rewritten during A-D conversion, the conversion result is indeterminate.
Note 2: Neither "01" nor "10" can be selected with the external op-amp connection mode bit.

Figure 1.77. Configuration of A-D conversion register in repeat sweep mode 0

(5) Repeat sweep mode 1

In repeat sweep mode 1, all pins are used for A-D conversion with emphasis on the pin or pins selected using the A-D sweep pin select bit. Table 1.40 shows the specifications of repeat sweep mode 1. Figure 1.78 shows the configuration of the A-D control register in repeat sweep mode 1.

Table 1.40. Repeat sweep mode 1 specifications

Item	Specification
Function	All pins perform repeat sweep A-D conversion, with emphasis on the pin or pins selected by the A-D sweep pin select bit Example : AN_0 selected $AN_0 \rightarrow AN_1 \rightarrow AN_0 \rightarrow AN_2 \rightarrow AN_0 \rightarrow AN_3$, etc
Start condition	Writing "1" to A-D conversion start flag
Stop condition	Writing "0" to A-D conversion start flag
Interrupt request generation timing	None generated
Input pin	AN_0 (1 pin), AN_0 and AN_1 (2 pins), AN_0 to AN_2 (3 pins), AN_0 to AN_3 (4 pins)
Reading of result of A-D converter	Read A-D register corresponding to selected pin (at any time)

A-D control register 0 (Note)

b7	b6	b5	b4	b3	b2	b1	b0
			1	1			

Symbol ADCON0 — Address $03D6_{16}$ — When reset $00000XXX_2$

Bit symbol	Bit name	Function	R	W
CH0	Analog input pin select bit	Invalid in repeat sweep mode 1	○	○
CH1			○	○
CH2			○	○
MD0	A-D operation mode select bit 0	b4 b3 1 1 : Repeat sweep mode 1	○	○
MD1			○	○
TRG	Trigger select bit	0 : Software trigger 1 : AD_{TRG} trigger	○	○
ADST	A-D conversion start flag	0 : A-D conversion disabled 1 : A-D conversion started	○	○
CKS0	Frequency select bit 0	0 : $f_{AD}/4$ is selected 1 : $f_{AD}/2$ is selected	○	○

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

A-D control register 1 (Note 1)

b7	b6	b5	b4	b3	b2	b1	b0
		1			1		

Symbol ADCON1 — Address $03D7_{16}$ — When reset 00_{16}

Bit symbol	Bit name	Function	R	W
SCAN0	A-D sweep pin select bit	When repeat sweep mode 1 is selected b1 b0 0 0 : AN_0 (1 pin) 0 1 : AN_0, AN_1 (2 pins) 1 0 : AN_0 to AN_2 (3 pins) 1 1 : AN_0 to AN_3 (4 pins)	○	○
SCAN1			○	○
MD2	A-D operation mode select bit 1	1 : Repeat sweep mode 1	○	○
BITS	8/10-bit mode select bit	0 : 8-bit mode 1 : 10-bit mode	○	○
CKS1	Frequency select bit 1	0 : $f_{AD}/2$ or $f_{AD}/4$ is selected 1 : f_{AD} is selected	○	○
VCUT	Vref connect bit	1 : Vref connected	○	○
OPA0	External op-amp connection mode bit (Note 2)	b7 b6 0 0 : ANEX0 and ANEX1 are not used 0 1 : ANEX0 input is A-D converted 1 0 : ANEX1 input is A-D converted 1 1 : External op-amp connection mode	○	○
OPA1			○	○

Note 1: If the A-D control register is rewritten during A-D conversion, the conversion result is indeterminate.
Note 2: Neither '01' nor '10' can be selected with the external op-amp connection mode bit.

Figure 1.78. Configuration of A-D conversion register in repeat sweep mode 1

(a) Sample and hold

Sample and hold is selected by setting bit 0 of the A-D control register 2 (address 03D4$_{16}$) to "1". When sample and hold is selected, the rate of conversion of each pin increases. As a result, a 28 f $_{AD}$ cycle is achieved with 8-bit resolution and 33 f $_{AD}$ with 10-bit resolution. Sample and hold can be selected in all modes. However, in all modes, be sure to specify before starting A-D conversion whether sample and hold is to be used.

(b) Extended analog input pins

In one-shot mode and repeat mode, the input via the extended analog input pins, ANEX0 and ANEX1, can also be converted from analog to digital.

When bit 6 of the A-D control register 1 (address 03D7$_{16}$) is "1" and bit 7 is "0", input via ANEX0 is converted from analog to digital. The result of conversion is stored in A-D register 0.

When bit 6 of the A-D control register 1 (address 03D7$_{16}$) is "0" and bit 7 is "1", input via ANEX1 is converted from analog to digital. The result of conversion is stored in A-D register 1.

(c) External operation amp connection mode

In this mode, multiple external analog inputs via the extended analog input pins, ANEX0 and ANEX1, can be amplified together by just one operation amp and used as the input for A-D conversion.

When bit 6 of the A-D control register 1 (address 03D7$_{16}$) is "1" and bit 7 is "1", input via AN$_0$ to AN$_7$ is output from ANEX0. The input from ANEX1 is converted from analog to digital and the result stored in the corresponding A-D register. The speed of A-D conversion depends on the response of the external operation amp. Do not connect the ANEX0 and ANEX1 pins directly. Figure 1.79 is an example of how to connect the pins in external operation amp mode.

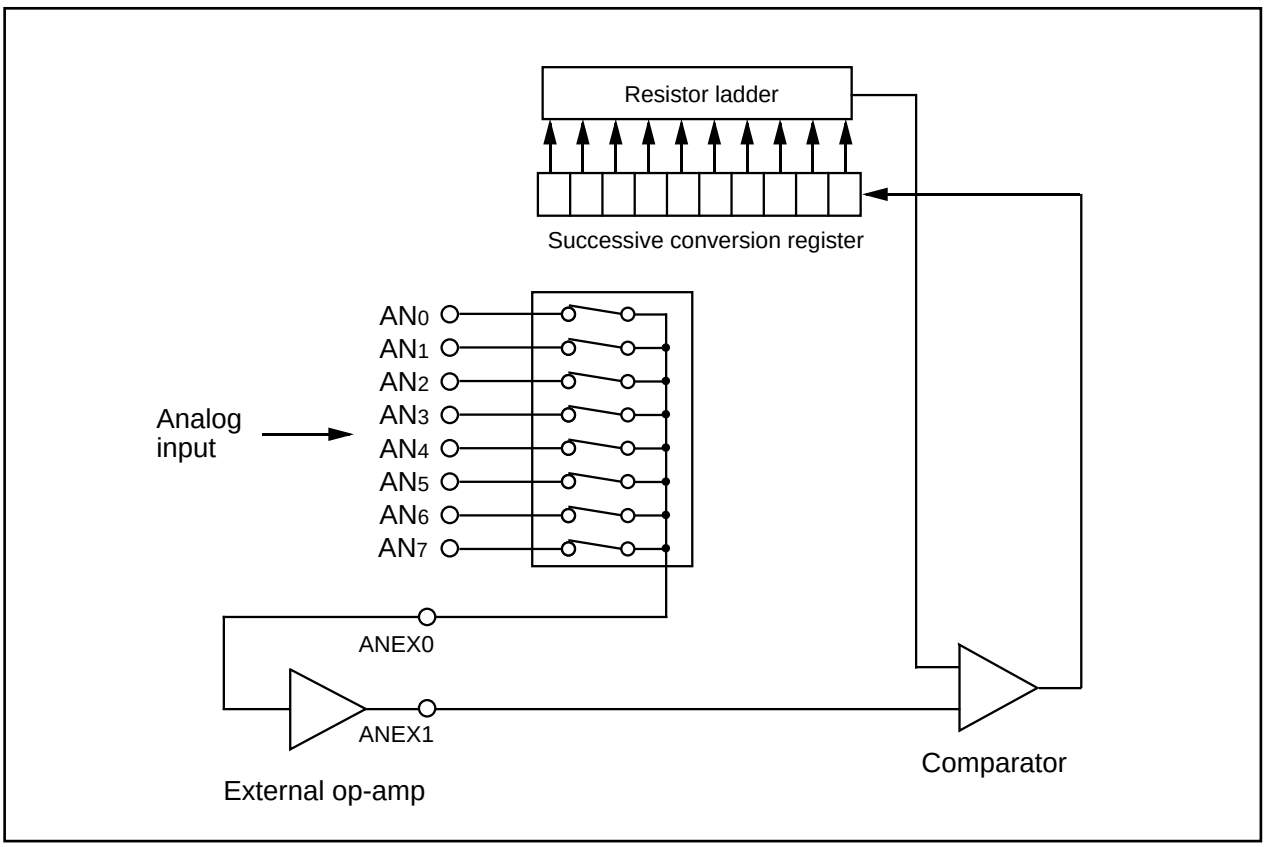


Figure 1.79. Example of external op-amp connection mode

D-A Converter

This is an 8-bit, R-2R type D-A converter. The microcomputer contains two independent D-A converters of this type.

D-A conversion is performed when a value is written to the corresponding D-A register. Bits 0 and 1 (D-A output enable bits) of the D-A control register decide if the result of conversion is to be output. Do not set the target port to output mode if D-A conversion is to be performed.

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 1.41 lists the performance of the D-A converter. Figure 1.80 shows a block diagram of the D-A converter. Figure 1.81 shows the configuration of the D-A control register.

Table 1.41. Performance of D-A converter

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

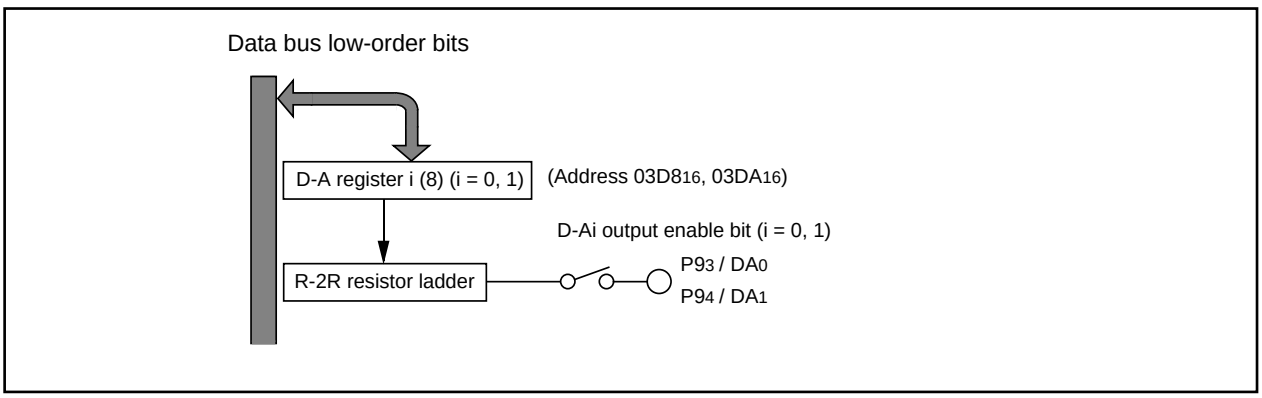


Figure 1.80. Block diagram of D-A converter

D-A control register

b7 b6 b5 b4 b3 b2 b1 b0

Symbol: DACON | Address: 03DC16 | When reset: 0016

Bit symbol	Bit name	Function	R	W
DA0E	D-A0 output enable bit	0 : Output disabled 1 : Output enabled	○	○
DA1E	D-A1 output enable bit	0 : Output disabled 1 : Output enabled	○	○
Nothing is assigned. These bits can neither be set nor reset. When read, the value of these bits is “0”.			—	—

D-A register

b7 b0

Symbol: DAi (i = 0,1) | Address: 03D816, 03DA16 | When reset: Indeterminate

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

Figure 1.81. Configuration of D-A control register

CRC Calculation Circuit

The Cyclic Redundancy Check (CRC) calculation circuit detects an error in data blocks. The microcomputer uses a generator polynomial of CRC_CCITT ($X^{16} + X^{12} + X^5 + 1$) to generate CRC code.

The CRC code is a 16-bit code generated for a block of a given data length in multiples of 8 bits. The CRC code is set in a CRC data register each time one byte of data is transferred to a CRC input register after writing an initial value into the CRC data register. Generation of CRC code for one byte of data is completed in two machine cycles.

Figure 1.82 shows a block diagram of the CRC circuit. Figure 1.83 shows the configuration of CRC-related registers.

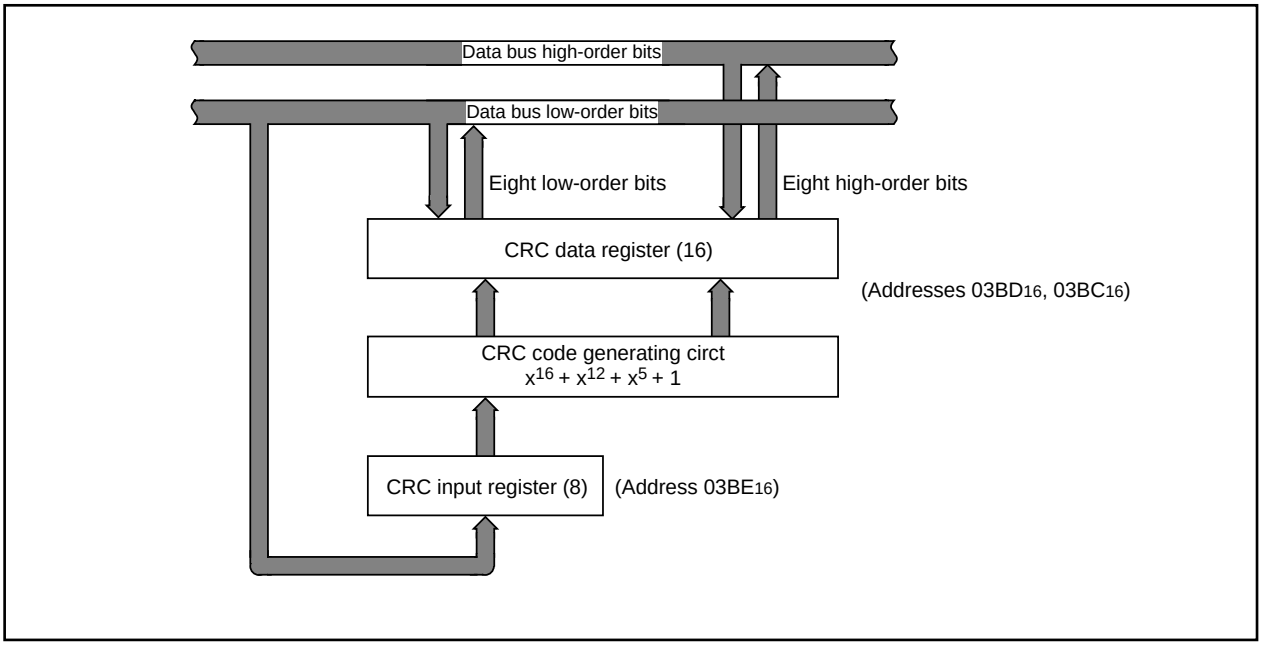


Figure 1.82. Block diagram of CRC circuit

CRC data register

(b15) (b8) b7 b0 b7 b0

Symbol	Address	When reset
CRCD	03BD16, 03BC16	Indeterminate

Function	Values that can be set	R	W
CRC calculation result output register	0000₁₆ to FFFF₁₆	○	○

CRC input register

b7 b0

Symbo	Address	When reset
CRCIN	03BE16	Indeterminate

Function	Values that can be set	R	W
Data input register	00₁₆ to FF16	○	○

Figure 1.83. Configuration of CRC-related registers

Programmable I/O Ports

There are 87 programmable I/O ports: P0 to P10 (excluding P8$_5$). Each port can be set independently for input or output using the direction register. A pull-up resistance for each block of 4 ports can be set. P8$_5$ is an input-only port and has no built-in pull-up resistance.

Figures 1.84 and 1.85 show the configuration of 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 1.86 shows a configuration of 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.

Note: There is no direction register bit for P8$_5$.

(2) Port registers

Figure 1.87 shows a configuration of 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 1.88 shows a configuration of pull-up control registers.

The pull-up control register can be set to apply a pull-up resistance to each block of 4 ports. When ports are set to have a pull-up resistance, the pull-up resistance is connected only when the direction register is set for input.

However, in memory expansion mode and microprocessor mode, P0 to P5 operate as the bus and the pull-up control register setting is invalid.

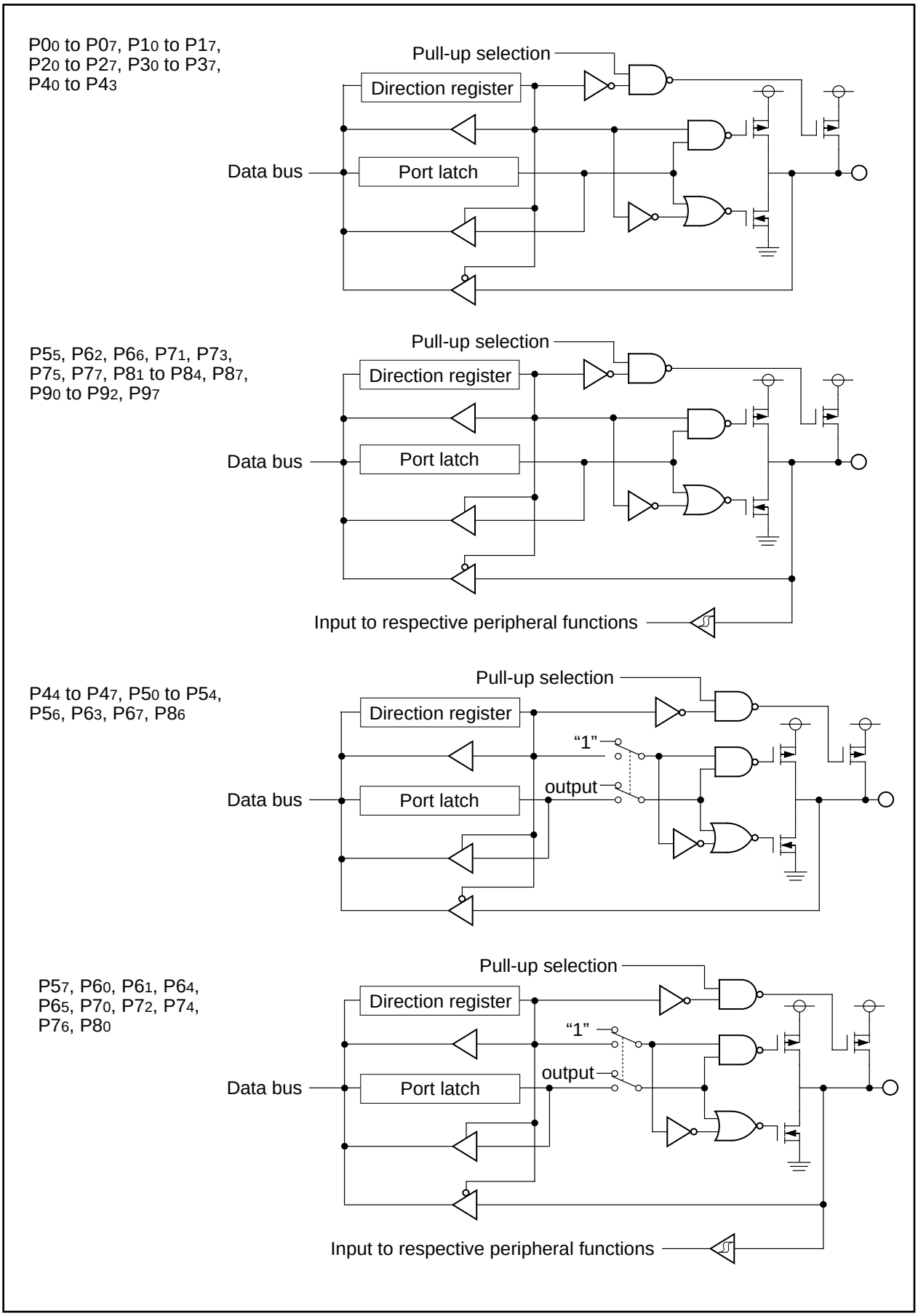


Figure 1.84. Configuration of programmable I/O ports (1)

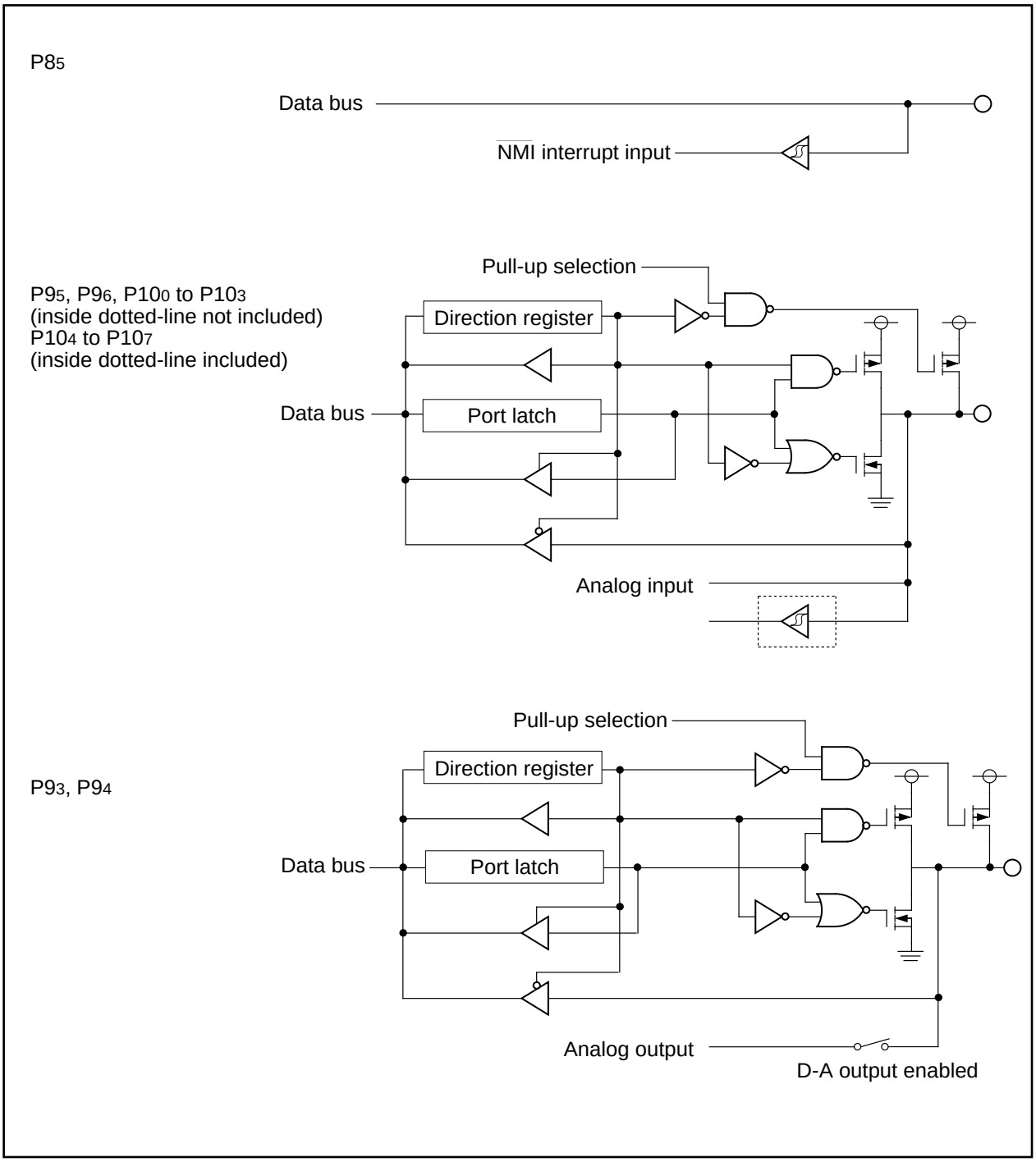
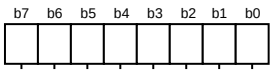


Figure 1.85. Configuration of programmable I/O ports (2)

Port Pi direction register (Note)

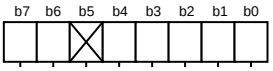


Symbol	Address	When reset
PDi (i = 0 to 10, except 8)	03E2$_{16}$, 03E3$_{16}$, 03E6$_{16}$, 03E7$_{16}$, 03EA$_{16}$ 03EB$_{16}$, 03EE$_{16}$, 03EF$_{16}$, 03F3$_{16}$, 03F6$_{16}$	00$_{16}$

Bit symbol	Bit name	Function	R	W
PDi_0	Port Pi$_0$ direction register	0 : Input mode (Functions as an input port) 1 : Output mode (Functions as an output port) (i = 0 to 10 except 8)	○	○
PDi_1	Port Pi$_1$ direction register		○	○
PDi_2	Port Pi$_2$ direction register		○	○
PDi_3	Port Pi$_3$ direction register		○	○
PDi_4	Port Pi$_4$ direction register		○	○
PDi_5	Port Pi$_5$ direction register		○	○
PDi_6	Port Pi$_6$ direction register		○	○
PDi_7	Port Pi$_7$ direction register		○	○

Note: Set bit 2 of protect register (address 000A$_{16}$) to “1” before rewritting to the port P9 direction register.

Port P8 direction register

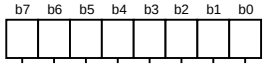


Symbol	Address	When reset
PD8	03F2$_{16}$	00X00000$_2$

Bit symbol	Bit name	Function	R	W
PD8_0	Port P8$_0$ direction register	0: Input mode (Functions as an input port) 1: Output mode (Functions as an output port)	○	○
PD8_1	Port P8$_1$ direction register		○	○
PD8_2	Port P8$_2$ direction register		○	○
PD8_3	Port P8$_3$ direction register		○	○
PD8_4	Port P8$_4$ direction register		○	○
Nothing is assigned. This bit can neither be set nor reset. When read, its content is indeterminate.			–	–
PD8_6	Port P8$_6$ direction register	0 : Input mode (Functions as an input port) 1 : Output mode (Functions as an output port)	○	○
PD8_7	Port P8$_7$ direction register		○	○

Figure 1.86. Configuration of direction register

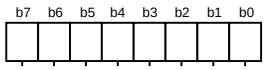
Port Pi register



Symbol	Addres	When reset
Pi (i = 0 to 10, except	03E0$_{16}$, 03E1$_{16}$, 03E4$_{16}$, 03E5$_{16}$, 03E8$_{16}$ 03E9$_{16}$, 03EC$_{16}$, 03ED$_{16}$, 03F1$_{16}$, 03F4$_{16}$	Indeterminate Indeterminate

Bit symbol	Bit name	Function	R	W
PDi_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 (i = 0 to 10 except 8)	○	○
PDi_1	Port Pi$_1$ register		○	○
PDi_2	Port Pi$_2$ register		○	○
PDi_3	Port Pi$_3$ register		○	○
PDi_4	Port Pi$_4$ register		○	○
PDi_5	Port Pi$_5$ register		○	○
PDi_6	Port Pi$_6$ register		○	○
PDi_7	Port Pi$_7$ register		○	○

Port P8 register

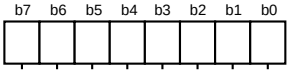


Symbol	Address	When reset
P8	03F0$_{16}$	Indeterminate

Bit symbol	Bit name	Function	R	W
PD8_0	Port P8$_0$ register	Data is input and output to and from each pin by reading and writing to and from each corresponding bit (except for P8$_5$) 0 : “L” level data 1 : “H” level data	○	○
PD8_1	Port P8$_1$ register		○	○
PD8_2	Port P8$_2$ register		○	○
PD8_3	Port P8$_3$ register		○	○
PD8_4	Port P8$_4$ register		○	○
PD8_5	Port P8$_5$ register		○	×
PD8_6	Port P8$_6$ register		○	○
PD8_7	Port P8$_7$ register		○	○

Figure 1.87. Configuration of port register

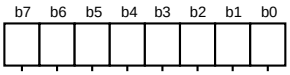
Pull-up control register 0



Symbol	Address	When reset
PUR0	03FC16	0016

Bit	Bit	Function	R	W
PU00	P00 to P03 pull-up	The corresponding port is pulled high with a pull-up resistor 0 : Not pulled high 1 : Pulled high	○	○
PU01	P04 to P07 pull-up		○	○
PU02	P10 to P13 pull-up		○	○
PU03	P14 to P17 pull-up		○	○
PU04	P20 to P23 pull-up		○	○
PU05	P24 to P27 pull-up		○	○
PU06	P30 to P33 pull-up		○	○
PU07	P34 to P37 pull-up		○	○

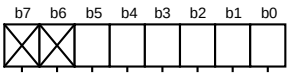
Pull-up control register 1



Symbol	Address	When reset
PUR1	03FD16	0016

Bit symbol	Bit name	Function	R	W
PU10	P40 to P43 pull-up	The corresponding port is pulled high with a pull-up resistor 0 : Not pulled high 1 : Pulled high	○	○
PU11	P44 to P47 pull-up		○	○
PU12	P50 to P53 pull-up		○	○
PU13	P54 to P57 pull-up		○	○
PU14	P60 to P63 pull-up		○	○
PU15	P64 to P67 pull-up		○	○
PU16	P70 to P73 pull-up		○	○
PU17	P74 to P77 pull-up		○	○

Pull-up control register 2



Symbol	Address	When reset
PUR2	03FE16	0016

Bit symbol	Bit name	Function	R	W
PU20	P80 to P83 pull-up	The corresponding port is pulled high with a pull-up resistor 0 : Not pulled high 1 : Pulled high	○	○
PU21	P84 to P87 pull-up (Except P85)		○	○
PU22	P90 to P93 pull-up		○	○
PU23	P94 to P97 pull-up		○	○
PU24	P100 to P103 pull-up		○	○
PU25	P104 to P107 pull-up		○	○
Nothing is assigned. Theses bits can neither be set nor reset. When read, their contents are "0".			—	—
			—	—

Figure 1.88. Configuration of pull-up control register

Table 1.42. Example of connection of unused pins in single-chip mode

Pin name	Connection
Ports P0 to P10 (excluding P8$_5$)	Specify output mode, and leave these pins open; or specify input mode, and connect to V_{SS} via resistor (pull-down)
X_{OUT} (Note)	Open
AV_{CC}, $\overline{NMI}$	Connect to V_{CC}
AV_{SS}, V_{REF}, BYTE	Connect to V_{SS}

Note: With external clock input to X_{IN} pin.

Table 1.43. Example of connection of unused pins in memory expansion and microprocessor mode

Pin name	Connection
Ports P6 to P10 (excluding P8$_5$)	Specify output mode, and leave these pins open; or specify input mode, and connect to V_{SS} via resistor (pull-down)
$\overline{BHE}$, ALE, $\overline{HLDA}$, X_{OUT} (Note), BCLK	Open
$\overline{HOLD}$, $\overline{RDY}$	Connect via resistor to V_{CC} (pull-up)
AV_{CC}, $\overline{NMI}$	Connect to V_{CC}
AV_{SS}, V_{REF}	Connect to V_{SS}

Note: With external clock input to X_{IN} pin.

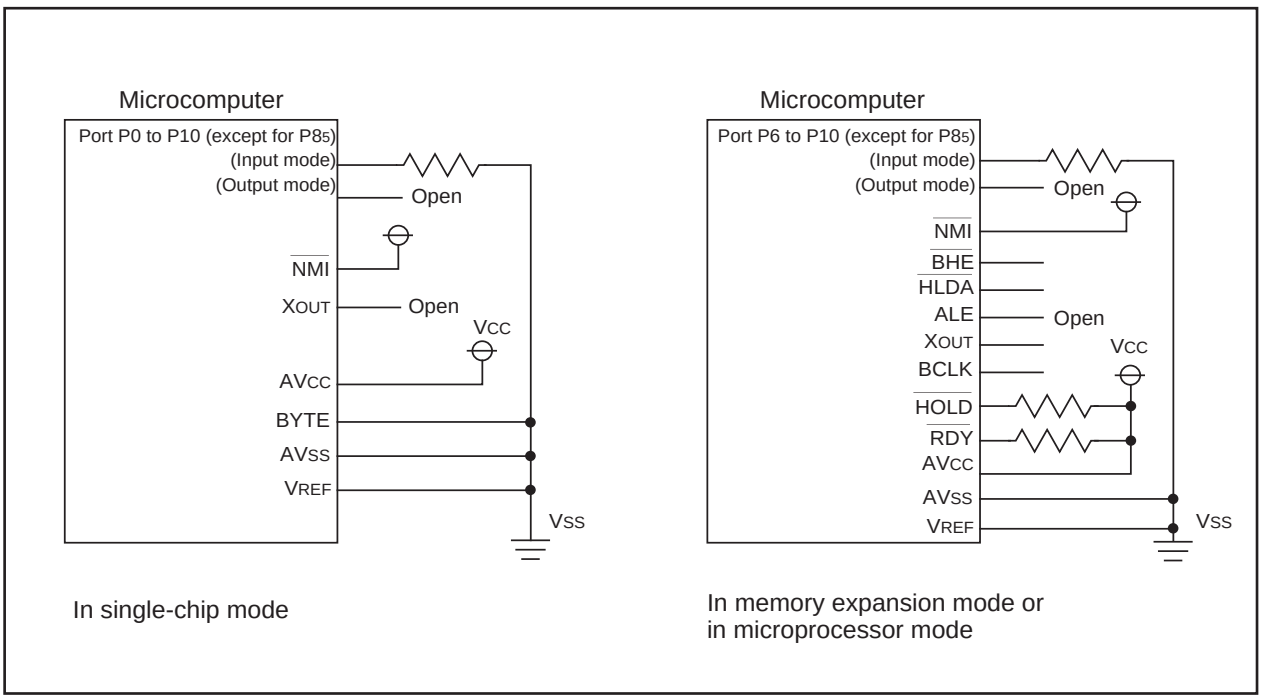


Figure 1.89. Example of connection of unused pins

Items to be submitted when ordering masked ROM version

Please submit the following when ordering masked ROM products:

(1) Mask ROM confirmation form

(2) Mark specification sheet

(3) ROM data : EPROM (3 sets)

Items to be submitted when ordering data to be written to ROM

Please submit the following when ordering data to be written to one-time PROM products at the factory:

(1) ROM writing order form

(2) Mark specification sheet

(3) ROM data : EPROM (3 sets)

Table 1.44. Absolute maximum ratings

Symbol	Parameter	Condition	Rated value	Unit
Vcc	Supply voltage		–0.3 to 7	V
AVcc	Analog supply voltage		–0.3 to 7	V
V_I	Input voltage $\overline{RESET}$, CNVss, BYTE, $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_7$, $P9_0$ to $P9_7$, $P10_0$ to $P10_7$, V_{REF}, X_{IN}		–0.3 to Vcc + 0.3 (Note 1)	V
V_O	Output voltage $P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ to $P3_7$, $P4_0$ to $P4_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_4$, $P8_6$,$P8_7$, $P9_0$ to $P9_7$, $P10_0$ to $P10_7$, X_{OUT}		–0.3 to Vcc + 0.3	V
P_d	Power dissipation	Ta=25 ˚C	300	mW
T_{opr}	Operating ambient temperature		–40 to 85	˚C
T_{stg}	Storage temperature		–65 to 150	˚C

Note 1: When writting to EPROM, only CNVss is –0.3 to 13 (V).

Table 1.45. Recommended operating conditions (referenced to Vcc = 2.7V to 5V at Ta = –40 to 85°C unless otherwise specified)

Symbol	Parameter		Standard			Unit
			Min	Typ.	Max.	
Vcc	Supply voltage		2.7	5.0	5.5	V
AVcc	Analog supply voltage			Vcc		V
Vss	Supply voltage			0		V
AVss	Analog supply voltage			0		V
V_{IH}	HIGH input voltage	$P3_1$ to $P3_7$, $P4_0$ to $P4_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_7$, $P9_0$ to $P9_7$, $P10_0$ to $P10_7$, X_{IN}, $\overline{RESET}$, CNV_{SS}, BYTE	0.8Vcc		Vcc	V
V_{IH}	HIGH input voltage	$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ (during single-chip mode)	0.8Vcc		Vcc	V
V_{IH}	HIGH input voltage	$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ (during memory expansion and microprocessor modes)	0.5Vcc		Vcc	V
V_{IL}	LOW input voltage	$P3_1$ to $P3_7$, $P4_0$ to $P4_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$, $P7_0$ to $P7_7$, $P8_0$ to $P8_7$, $P9_0$ to $P9_7$, $P10_0$ to $P10_7$, X_{IN}, $\overline{RESET}$, CNV_{SS}, BYTE	0		0.2Vcc	V
V_{IL}	LOW input voltage	$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ (during single-chip mode)	0		0.2Vcc	V
V_{IL}	LOW input voltage	$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$, $P3_0$ (during memory expansion and microprocessor modes)	0		0.16Vcc	V
$I_{OH (peak)}$	HIGH peak output current	$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$,$P3_0$ to $P3_7$, $P4_0$ to $P4_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$,$P7_0$ to $P7_7$, $P8_0$ to $P8_4$,$P8_6$,$P8_7$,$P9_0$ to $P9_7$,$P10_0$ to $P10_7$			–10.0	mA
$I_{OH (avg)}$	HIGH average output current	$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$,$P3_0$ to $P3_7$, $P4_0$ to $P4_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$,$P7_0$ to $P7_7$, $P8_0$ to $P8_4$,$P8_6$,$P8_7$,$P9_0$ to $P9_7$,$P10_0$ to $P10_7$			–5.0	mA
$I_{OL (peak)}$	LOW peak output current	$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$,$P3_0$ to $P3_7$, $P4_0$ to $P4_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$,$P7_0$ to $P7_7$, $P8_0$ to $P8_4$,$P8_6$,$P8_7$,$P9_0$ to $P9_7$,$P10_0$ to $P10_7$			10.0	mA
$I_{OL (avg)}$	LOW average output current	$P0_0$ to $P0_7$, $P1_0$ to $P1_7$, $P2_0$ to $P2_7$,$P3_0$ to $P3_7$, $P4_0$ to $P4_7$, $P5_0$ to $P5_7$, $P6_0$ to $P6_7$,$P7_0$ to $P7_7$, $P8_0$ to $P8_4$,$P8_6$,$P8_7$,$P9_0$ to $P9_7$,$P10_0$ to $P10_7$			5.0	mA
f (X_{IN})	Main clock input oscillation frequency	Vcc =4.0V to 5.5V			10	MHz
		Vcc = 2.7V to 5.5V (with wait)			7	MHz
f (X_{CIN})	Subclock oscillation frequency			32.768	50	kHz

Note 1: The mean output current is the mean value within 100ms.

Note 2: The total I_{OL} (peak) for ports P0, P1, P2, $P8_6$, $P8_7$, P9, and P10 must be 80mA max. The total I_{OH} (peak) for ports P0, P1, P2, $P8_6$, $P8_7$, P9, and P10 must be 80mA max. The total I_{OL} (peak) for ports P3, P4, P5, P6, P7, and $P8_0$ to $P8_4$ must be 80mA max. The total I_{OH} (peak) for ports P3, P4, P5, P6, P7, and $P8_0$ to $P8_4$ must be 80mA max.

Vcc = 5V

Table 1.46. Electrical characteristics (referenced to Vcc = 5V, Vss = 0V at Ta = 25°C, f(XIN) = 10MHz unless otherwise specified)

Symbol	Parameter		Measuring condition		Standard Min.	Standard Typ.	Standard Max.	Unit
VOH	HIGH output voltage	P00 to P07,P10 to P17,P20 to P27, P30 to P37,P40 to P47,P50 to P57, P60 to P67,P70 to P77,P80 to P84, P86,P87,P90 to P97,P100 to P107	IOH = –5mA		3.0			V
VOH	HIGH output voltage	P00 to P07,P10 to P17,P20 to P27, P30 to P37,P40 to P47,P50 to P57	IOH = –200µA		4.7			V
VOH	HIGH output voltage	XOUT	HIGHPOWER	IOH = –1mA	3.0			V
			LOWPOWER	IOH = –0.5mA	3.0			
VOL	LOW output voltage	P00 to P07,P10 to P17,P20 to P27, P30 to P37,P40 to P47,P50 to P57, P60 to P67,P70 to P77,P80 to P84, P86,P87,P90 to P97,P100 to P107	IOL = 5mA				2.0	V
VOL	LOW output voltage	P00 to P07,P10 to P17,P20 to P27, P30 to P37,P40 to P47,P50 to P53	IOL = 200µA				0.45	V
VOL	LOW output voltage	XOUT	HIGHPOWER	IOL = 1mA			2.0	V
			LOWPOWER	IOL = 0.5mA			2.0	
VT+-VT-	Hysteresis	HOLD, RDY, TA0IN to TA4IN, TB0IN to TB2IN, INT0 to INT2, ADTRG, CTS0, CTS1, CLK0, CLK1			0.2		0.8	V
VT+-VT-	Hysteresis	RESET			0.2		1.8	V
VT+-VT-	Hysteresis	XIN			0.2		0.8	V
IIH	HIGH input current	P00 to P07,P10 to P17,P20 to P27, P30 to P37,P40 to P47,P50 to P57, P60 to P67,P70 to P77,P80 to P87, P90 to P97,P100 to P107, XIN, RESET, CNVss, BYTE	VI = 5V				5.0	µA
IIL	LOW input current	P00 to P07,P10 to P17,P20 to P27, P30 to P37,P40 to P47,P50 to P57, P60 to P67,P70 to P77,P80 to P87, P90 to P97,P100 to P107, XIN, RESET, CNVss, BYTE	VI = 0V				–5.0	µA
VRAM	RAM retention voltage		When clock is stopped		2.0			V
Icc	Power supply current		When reset in single-chip mode, the output-only pins are open and other pins are Vss	f(XIN) = 10MHz Square wave, no division		19.0	38.0	mA
				f(XIN) = 10MHz Square wave, division by 8		4.2		mA
				f(XCIN) = 32kHz When a WAIT instruction is executed		4.0		µA
				Ta = 25°C when clock is stopped			1.0	µA
				Ta = 85°C when clock is stopped			20.0	

Vcc = 5V

Table 1.47. A-D conversion characteristics (referenced to Vcc = AVcc = VREF = 5V, Vss = AVss = 0V at Ta = 25°C, f(XIN) = 10MHz unless otherwise specified)

Symbol	Parameter		Measuring condition		Standard			Unit
					Min.	Typ.	Max.	
–	Resolution		VREF = Vcc				10	Bits
–	Absolute accuracy	Sample & hold function not available	VREF = Vcc = 5V				±3	LSB
		Sample & hold function available(10bit)	VREF =Vcc = 5V	AN0 to AN7 input			±3	LSB
				ANEX0, ANEX1 input, External op-amp connection mode			±7	LSB
		Sample & hold function available(8bit)	VREF = Vcc = 5V				±2	LSB
RLADDER	Ladder resistance		VREF =Vcc		10		40	kΩ
tCONV	Conversion time(10bit)				3.3			µs
tCONV	Conversion time(8bit)				2.8			µs
tSAMP	Sampling time				0.3			µs
VREF	Reference voltage				2		Vcc	V
VIA	Analog input voltage				0		VREF	V

Table 1.48. D-A conversion characteristics (referenced to Vcc = 5V, Vss = AVss = 0V, VREF = 5V at Ta = 25°C, f(XIN) = 10MHz unless otherwise specified)

Symbol	Parameter	Measuring condition	Standard			Unit
			Min.	Typ.	Max.	
–	Resolution				8	Bits
–	Absolute accuracy				1.0	%
tsu	Setup time				3	µs
RO	Output resistance		4	10	20	kΩ
IVREF	Reference power supply input current	(Note)			1.5	mA

Note: This applies when using one D-A converter, with the D-A register for the unused D-A converter set to "00$_{16}$".
The A-D converter's ladder resistance is not included.

Vcc = 5V

Timing requirements (referenced to Vcc = 5V, Vss = 0V at Ta = 25°C unless otherwise specified)

Table 1.50. External clock input

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t_c	External clock input cycle time	100		ns
$t_{w(H)}$	External clock input HIGH pulse width	40		ns
$t_{w(L)}$	External clock input LOW pulse width	40		ns
t_r	External clock rise time		15	ns
t_f	External clock fall time		15	ns

Table 1.51. Memory expansion and microprocessor modes

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{ac1(RD-DB)}$	Data input access time (no wait)		(Note)	ns
$t_{ac2(RD-DB)}$	Data input access time (with wait)		(Note)	ns
$t_{ac3(RD-DB)}$	Data input access time (when accessing multiplex bus area)		(Note)	ns
$t_{su(DB-RD)}$	Data input setup time	40		ns
$t_{su(RDY-BCLK)}$	$\overline{RDY}$ input setup time	30		ns
$t_{su(HOLD-BCLK)}$	$\overline{HOLD}$ input setup time	40		ns
$t_{h(RD-DB)}$	Data input hold time	0		ns
$t_{h(BCLK-RDY)}$	$\overline{RDY}$ input hold time	0		ns
$t_{h(BCLK-HOLD)}$	$\overline{HOLD}$ input hold time	0		ns
$t_{d(BCLK-HLDA)}$	$\overline{HLDA}$ output delay time		40	ns

Note: Calculated according to the BCLK frequency as follows:

$$t_{ac1(RD-DB)} = \frac{10^9}{f(BCLK) \times 2} - 45 \quad [ns]$$

$$t_{ac2(RD-DB)} = \frac{3 \times 10^9}{f(BCLK) \times 2} - 45 \quad [ns]$$

$$t_{ac3(RD-DB)} = \frac{3 \times 10^9}{f(BCLK) \times 2} - 45 \quad [ns]$$

VCC = 5V

Timing requirements (referenced to VCC = 5V, VSS = 0V at Ta = 25°C unless otherwise specified)

Table 1.52. Timer A input (counter input in event counter mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TA)}$	TAiIN input cycle time	100		ns
$t_{w(TAH)}$	TAiIN input HIGH pulse width	40		ns
$t_{w(TAL)}$	TAiIN input LOW pulse width	40		ns

Table 1.53. Timer A input (gating input in timer mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TA)}$	TAiIN input cycle time	400		ns
$t_{w(TAH)}$	TAiIN input HIGH pulse width	200		ns
$t_{w(TAL)}$	TAiIN input LOW pulse width	200		ns

Table 1.54. Timer A input (external trigger input in one-shot timer mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TA)}$	TAiIN input cycle time	200		ns
$t_{w(TAH)}$	TAiIN input HIGH pulse width	100		ns
$t_{w(TAL)}$	TAiIN input LOW pulse width	100		ns

Table 1.55. Timer A input (external trigger input in pulse width modulation mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{w(TAH)}$	TAiIN input HIGH pulse width	100		ns
$t_{w(TAL)}$	TAiIN input LOW pulse width	100		ns

Table 1.56. Timer A input (up/down input in event counter mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(UP)}$	TAiOUT input cycle time	2000		ns
$t_{w(UPH)}$	TAiOUT input HIGH pulse width	1000		ns
$t_{w(UPL)}$	TAiOUT input LOW pulse width	1000		ns
$t_{su(UP-TIN)}$	TAiOUT input setup time	400		ns
$t_{h(TIN-UP)}$	TAiOUT input hold time	400		ns

Vcc = 5V

Timing requirements (referenced to Vcc = 5V, Vss = 0V at Ta = 25°C unless otherwise specified)

Table 1.57. Timer B input (counter input in event counter mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TB)}$	TBiIN input cycle time (counted on one edge)	100		ns
$t_{w(TBH)}$	TBiIN input HIGH pulse width (counted on one edge)	40		ns
$t_{w(TBL)}$	TBiIN input LOW pulse width (counted on one edge)	40		ns
$t_{c(TB)}$	TBiIN input cycle time (counted on both edges)	200		ns
$t_{w(TBH)}$	TBiIN input HIGH pulse width (counted on both edges)	80		ns
$t_{w(TBL)}$	TBiIN input LOW pulse width (counted on both edges)	80		ns

Table 1.58. Timer B input (pulse period measurement mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TB)}$	TBiIN input cycle time	400		ns
$t_{w(TBH)}$	TBiIN input HIGH pulse width	200		ns
$t_{w(TBL)}$	TBiIN input LOW pulse width	200		ns

Table 1.59. Timer B input (pulse width measurement mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TB)}$	TBiIN input cycle time	400		ns
$t_{w(TBH)}$	TBiIN input HIGH pulse width	200		ns
$t_{w(TBL)}$	TBiIN input LOW pulse width	200		ns

Table 1.60. A-D trigger input

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(AD)}$	$\overline{ADTRG}$ input cycle time (trigger able minimum)	1000		ns
$t_{w(ADL)}$	$\overline{ADTRG}$ input LOW pulse width	125		ns

Table 1.61. 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 1.62. External interrupt $\overline{INTi}$ inputs

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

VCC = 5V

Switching characteristics (referenced to VCC = 5V, VSS = 0V at Ta = 25°C, CM15 = "1" unless otherwise specified)

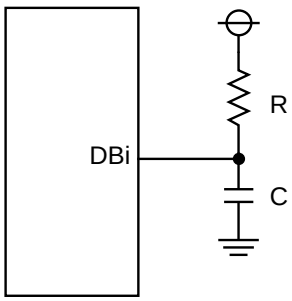
Table 1.63. Memory expansion mode and microprocessor mode (no wait)

Symbol	Parameter	Measuring condition	Standard		Unit
			Min.	Max.	
$t_{d(BCLK-AD)}$	Address output delay time	Figure 1.90		25	ns
$t_{h(BCLK-AD)}$	Address output hold time (BCLK standard)		4		ns
$t_{h(RD-AD)}$	Address output hold time (RD standard)		0		ns
$t_{h(WR-AD)}$	Address output hold time (WR standard)		0		ns
$t_{d(BCLK-CS)}$	Chip select output delay time			25	ns
$t_{h(BCLK-CS)}$	Chip select output hold time (BCLK standard)		4		ns
$t_{d(BCLK-ALE)}$	ALE signal output delay time			25	ns
$t_{h(BCLK-ALE)}$	ALE signal output hold time		– 4		ns
$t_{d(BCLK-RD)}$	RD signal output delay time			25	ns
$t_{h(BCLK-RD)}$	RD signal output hold time		0		ns
$t_{d(BCLK-WR)}$	WR signal output delay time			25	ns
$t_{h(BCLK-WR)}$	WR signal output hold time		0		ns
$t_{d(BCLK-DB)}$	Data output delay time (BCLK standard)			40	ns
$t_{h(BCLK-DB)}$	Data output hold time (BCLK standard)		4		ns
$t_{d(DB-WR)}$	Data output delay time (WR standard)		(Note 1)		ns
$t_{h(WR-DB)}$	Data output hold time (WR standard) (Note 2)		0		ns

Note 1: Calculated according to the BCLK frequency as follows:

$$td(DB - WR) = \frac{10^9}{f(BCLK) \times 2} - 40 \quad [ns]$$

Note 2: This is standard value shows the timing when the output is off, and doesn't show hold time of data bus.
Hold time of data bus is different by capacitor volume and pull-up (pull-down) resistance value.
Hold time of data bus is expressed in
$t = -CR \times \ln (1 - V_{OL} / V_{CC})$
by a circuit of the right figure.
For example, when $V_{OL} = 0.2V_{CC}$, C = 30pF, R = 1kW, hold time of output "L" level is
$t = -30pF \times 1kW \times \ln (1 - 0.2V_{CC} / V_{CC})$
$= 6.7ns$.



VCC = 5V

Switching characteristics (referenced to VCC = 5V, VSS = 0V at Ta = 25°C, CM15 = "1" unless otherwise specified)

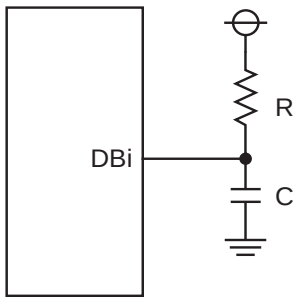
Table 1.64. Memory expansion mode and microprocessor mode (with wait, accessing external memory)

Symbol	Parameter	Measuring condition	Standard		Unit
			Min.	Max.	
$t_{d(BCLK-AD)}$	Address output delay time	Figure 1.90		25	ns
$t_{h(BCLK-AD)}$	Address output hold time (BCLK standard)		4		ns
$t_{h(RD-AD)}$	Address output hold time (RD standard)		0		ns
$t_{h(WR-AD)}$	Address output hold time (WR standard)		0		ns
$t_{d(BCLK-CS)}$	Chip select output delay time			25	ns
$t_{h(BCLK-CS)}$	Chip select output hold time (BCLK standard)		4		ns
$t_{d(BCLK-ALE)}$	ALE signal output delay time			25	ns
$t_{h(BCLK-ALE)}$	ALE signal output hold time		– 4		ns
$t_{d(BCLK-RD)}$	RD signal output delay time			25	ns
$t_{h(BCLK-RD)}$	RD signal output hold time		0		ns
$t_{d(BCLK-WR)}$	WR signal output delay time			25	ns
$t_{h(BCLK-WR)}$	WR signal output hold time		0		ns
$t_{d(BCLK-DB)}$	Data output delay time (BCLK standard)			40	ns
$t_{h(BCLK-DB)}$	Data output hold time (BCLK standard)		4		ns
$t_{d(DB-WR)}$	Data output delay time (WR standard)		(Note 1)		ns
$t_{h(WR-DB)}$	Data output hold time (WR standard) (Note 2)		0		ns

Note 1: Calculated according to the BCLK frequency as follows:

$$td(DB - WR) = \frac{10^9}{f(BCLK)} - 40 \quad [ns]$$

Note 2: This is standard value shows the timing when the output is off, and doesn't show hold time of data bus.
Hold time of data bus is different by capacitor volume and pull-up (pull-down) resistance value.
Hold time of data bus is expressed in
$t = -CR \times \ln (1 - V_{OL} / V_{CC})$
by a circuit of the right figure.
For example, when $V_{OL} = 0.2V_{CC}$, C = 30pF, R = 1kW, hold time of output "L" level is
$t = -30pF \times 1kW \times \ln (1 - 0.2V_{CC} / V_{CC})$
$= 6.7ns$.



VCC = 5V

Switching characteristics (referenced to VCC = 5V, VSS = 0V at Ta = 25°C, CM15 ="1" unless otherwise specified)

Table 1.65. Memory expansion mode and microprocessor mode (with wait, accessing external memory, multiplex bus area selected)

Symbol	Parameter	Measuring condition	Standard		Unit
			Min.	Max.	
$t_{d(BCLK-AD)}$	Address output delay time	Figure 1.90		25	ns
$t_{h(BCLK-AD)}$	Address output hold time (BCLK standard)		4		ns
$t_{h(RD-AD)}$	Address output hold time (RD standard)		(Note)		ns
$t_{h(WR-AD)}$	Address output hold time (WR standard)		(Note)		ns
$t_{d(BCLK-CS)}$	Chip select output delay time			25	ns
$t_{h(BCLK-CS)}$	Chip select output hold time (BCLK standard)		4		ns
$t_{h(RD-CS)}$	Chip select output hold time (RD standard)		(Note)		ns
$t_{h(WR-CS)}$	Chip select output hold time (WR standard)		(Note)		ns
$t_{d(BCLK-RD)}$	RD signal output delay time			25	ns
$t_{h(BCLK-RD)}$	RD signal output hold time		0		ns
$t_{d(BCLK-WR)}$	WR signal output delay time			25	ns
$t_{h(BCLK-WR)}$	WR signal output hold time		0		ns
$t_{d(BCLK-DB)}$	Data output delay time (BCLK standard)			40	ns
$t_{h(BCLK-DB)}$	Data output hold time (BCLK standard)		4		ns
$t_{d(DB-WR)}$	Data output delay time (WR standard)		(Note)		ns
$t_{h(WR-DB)}$	Data output hold time (WR standard)		(Note)		ns
$t_{d(BCLK-ALE)}$	ALE signal output delay time (BCLK standard)			25	ns
$t_{h(BCLK-ALE)}$	ALE signal output hold time (BCLK standard)		– 4		ns
$t_{d(AD-ALE)}$	ALE signal output delay time (Address standard)		(Note)		ns
$t_{h(ALE-AD)}$	ALE signal output hold time (Adderss standard)		50		ns
$t_{d(AD-RD)}$	Post-address RD signal output delay time		0		ns
$t_{d(AD-WR)}$	Post-address WR signal output delay time		0		ns
$t_{dZ(RD-AD)}$	Address output floating start time			8	ns

Note: Calculated according to the BCLK frequency as follows:

$$th(RD - AD) = \frac{10^9}{f(BCLK) \times 2} \quad [ns]$$

$$th(WR - AD) = \frac{10^9}{f(BCLK) \times 2} \quad [ns]$$

$$th(RD - CS) = \frac{10^9}{f(BCLK) \times 2} \quad [ns]$$

$$th(WR - CS) = \frac{10^9}{f(BCLK) \times 2} \quad [ns]$$

$$td(DB - WR) = \frac{10^9 \times 3}{f(BCLK) \times 2} - 40 \quad [ns]$$

$$th(WR - DB) = \frac{10^9}{f(BCLK) \times 2} \quad [ns]$$

$$td(AD - ALE) = \frac{10^9}{f(BCLK) \times 2} - 25 \quad [ns]$$

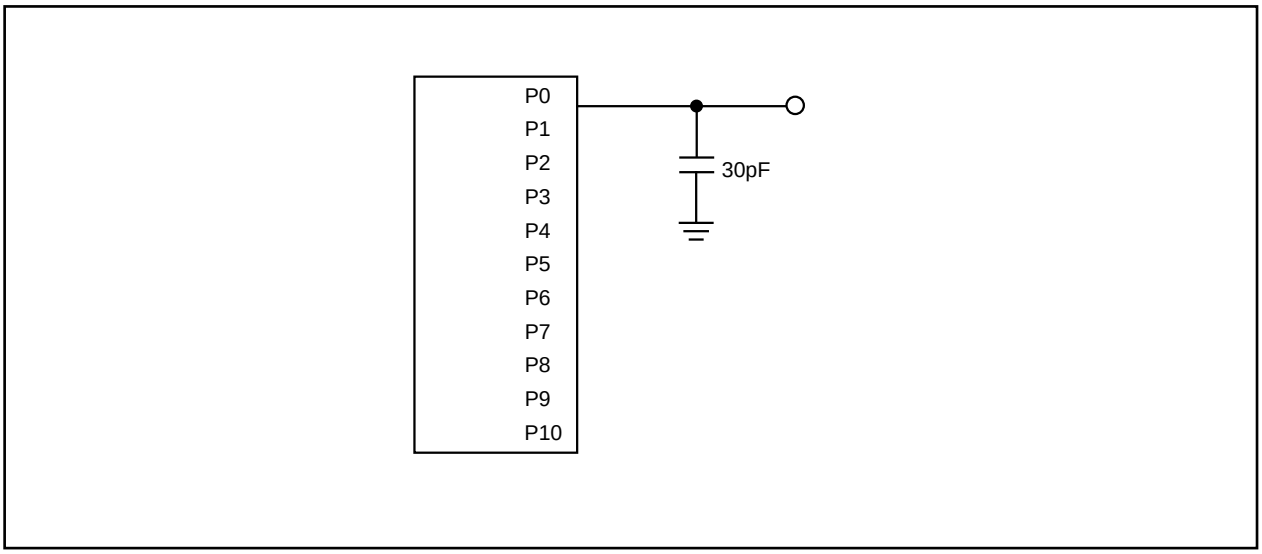
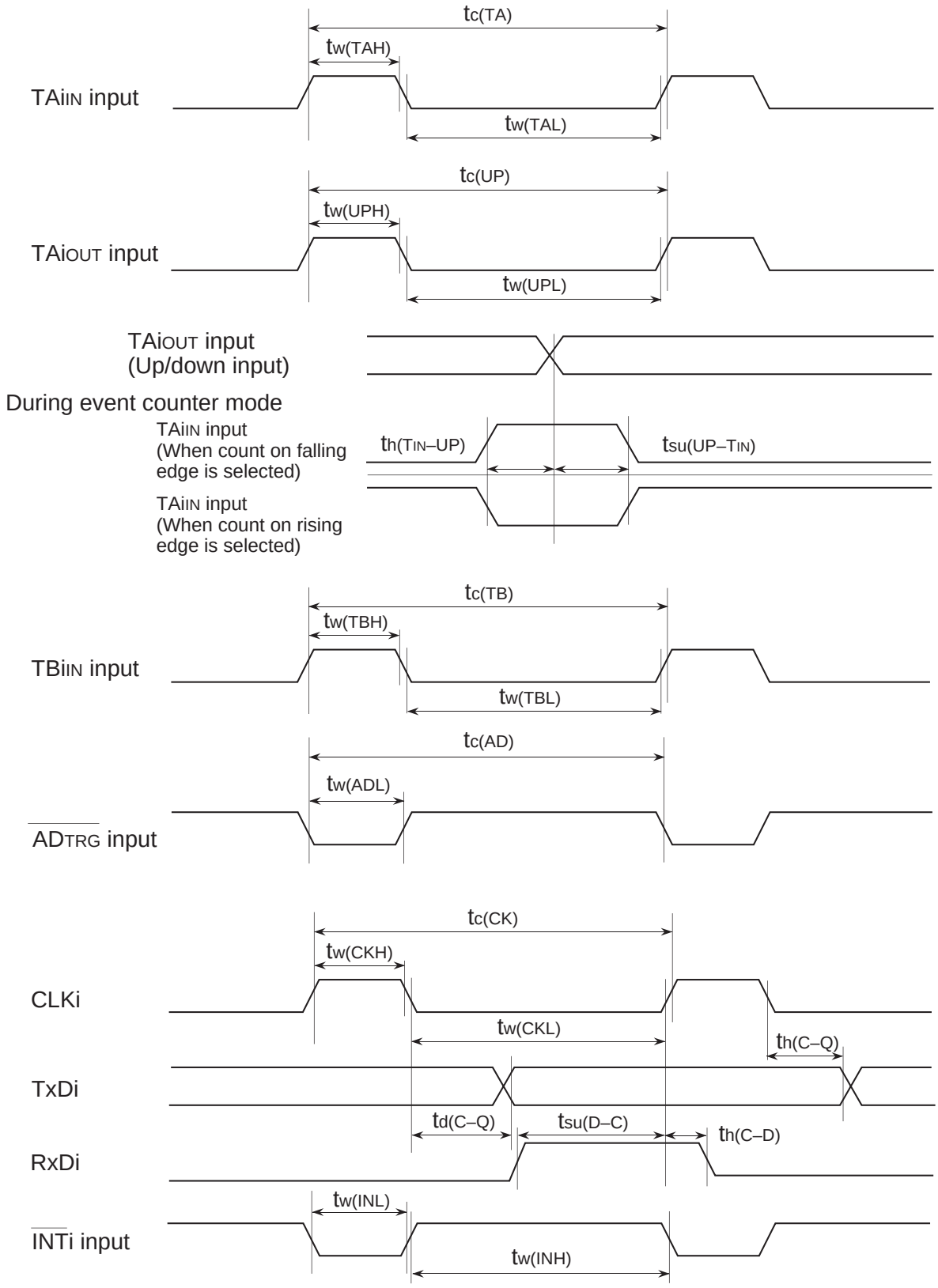


Figure 1.90. Port P0 to P10 measurement circuit

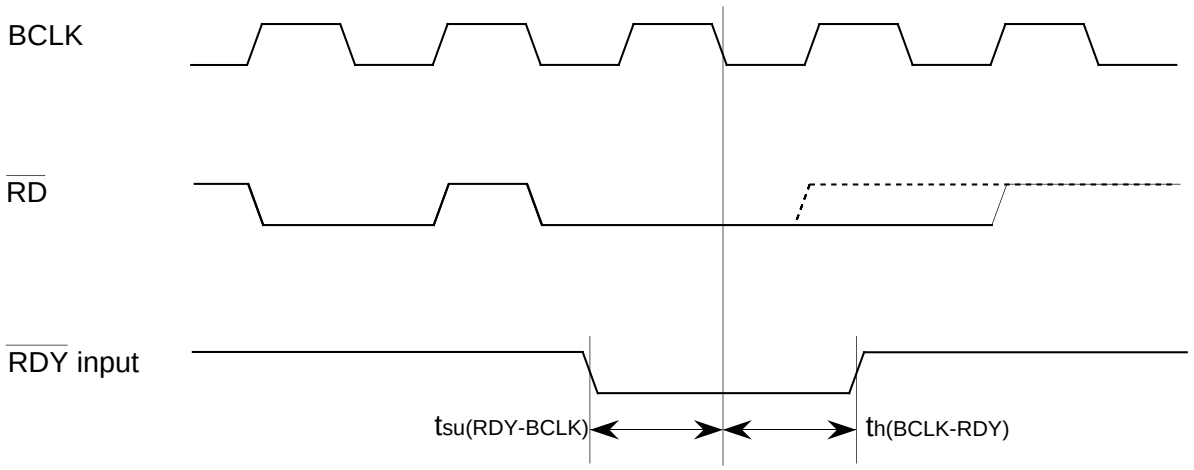
Vcc = 5V


tc(TA)
tw(TAH)
TAiIN input
tw(TAL)
tc(UP)
tw(UPH)
TAiOUT input
tw(UPL)
TAiOUT input
(Up/down input)
During event counter mode
TAiIN input
(When count on falling
edge is selected)
th(TIN–UP)
tsu(UP–TIN)
TAiIN input
(When count on rising
edge is selected)
tc(TB)
tw(TBH)
TBiIN input
tw(TBL)
tc(AD)
tw(ADL)
ADTRG input
tc(CK)
tw(CKH)
CLKi
tw(CKL)
th(C–Q)
TxDi
td(C–Q)
tsu(D–C)
th(C–D)
RxDi
tw(INL)
INTi input
tw(INH)

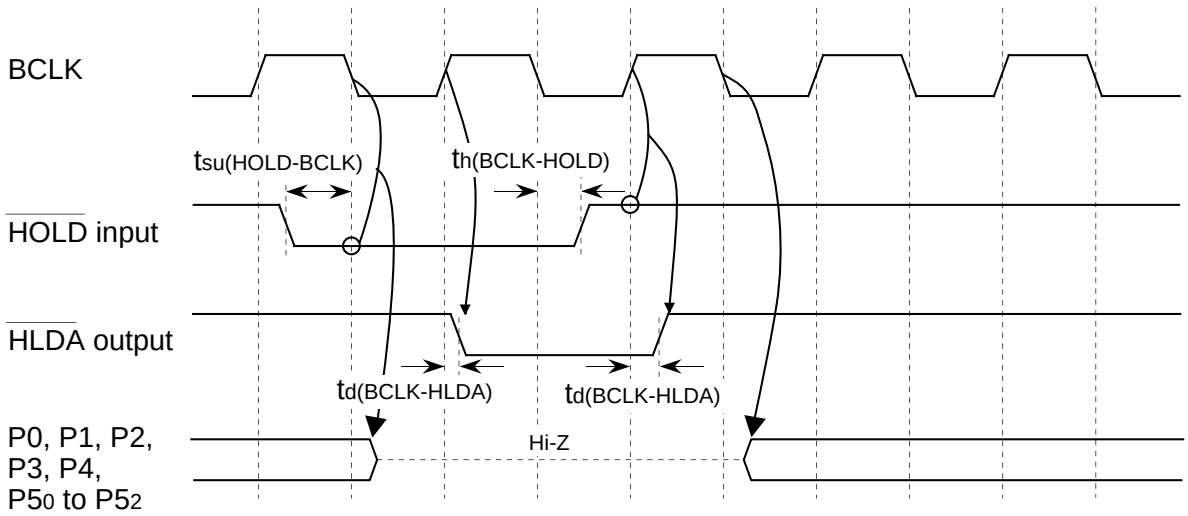
V_{CC} = 5V

Memory Expansion Mode and Microprocessor Mode

(Valid only with wait)



(Valid with or without wait)



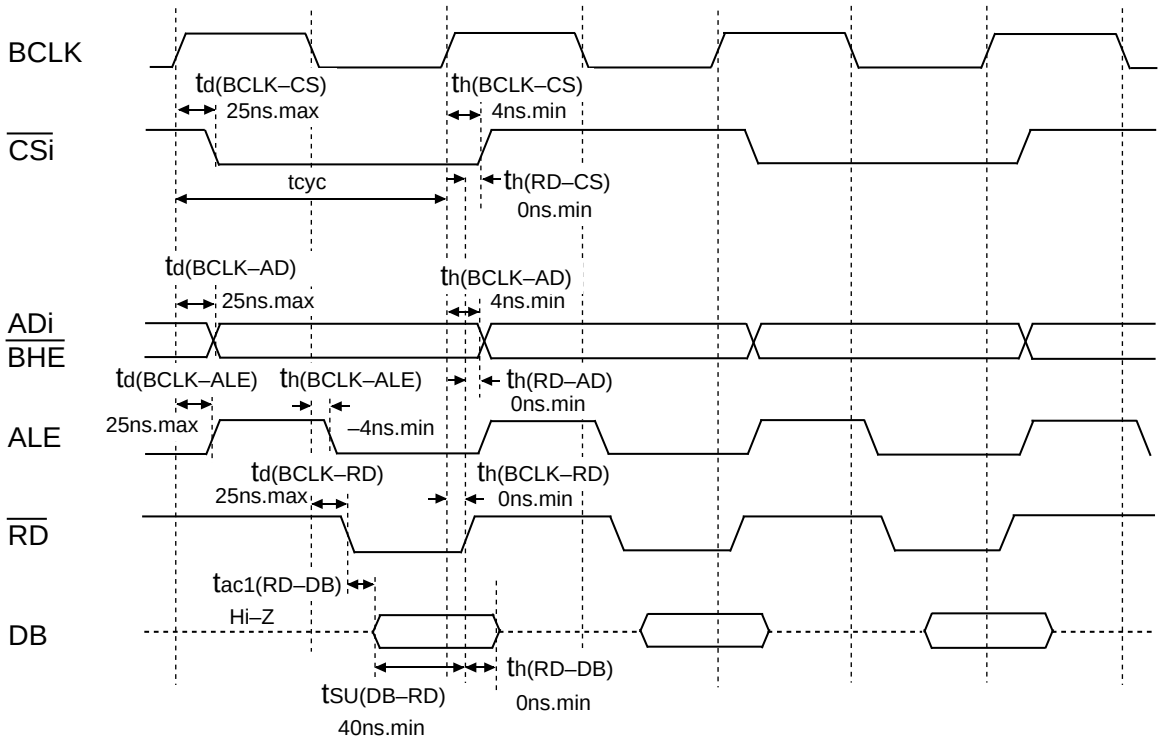
Note: The above pins are set to high-impedance regardless of the input level of the BYTE pin and bit (PM06) of processor mode register 0 selects the function of ports P40 to P43.

Measuring conditions :
- V_{CC} = 5V
- Input timing voltage : Determined with V_{IL} = 1.0V, V_{IH} = 4.0V
- Output timing voltage : Determined with V_{OL} =2.5V, V_{OH} = 2.5V

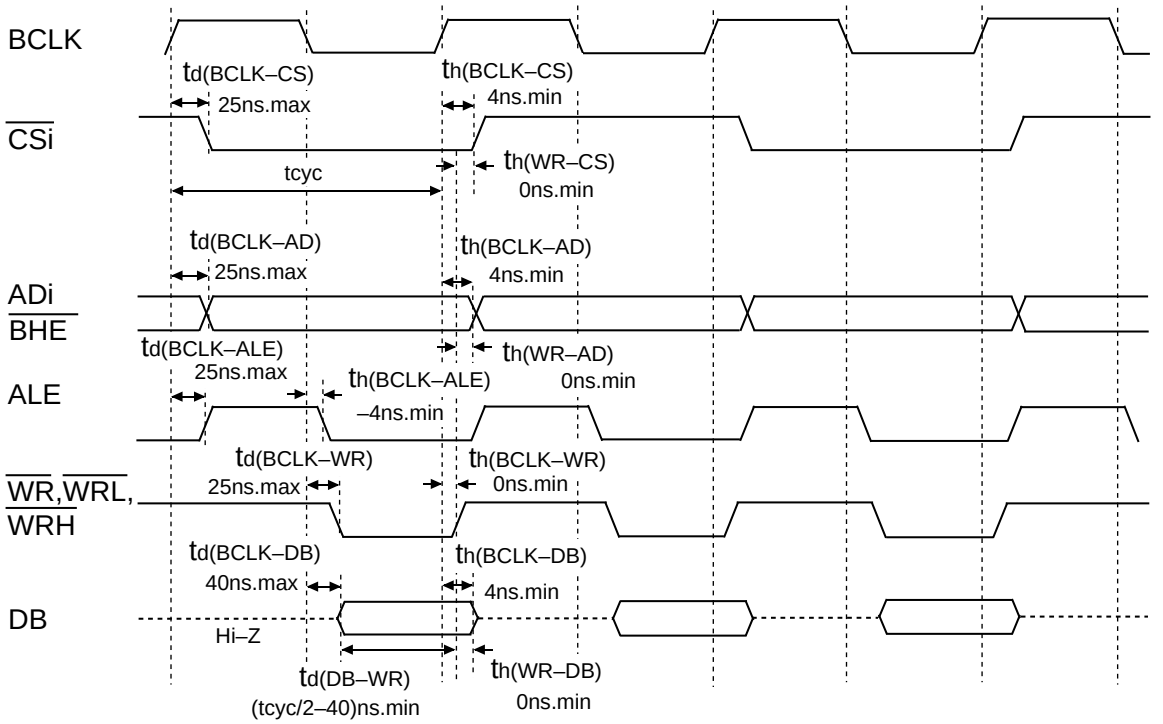
Memory Expansion Mode and Microprocessor Modes (With no wait)

VCC = 5V

Read timing



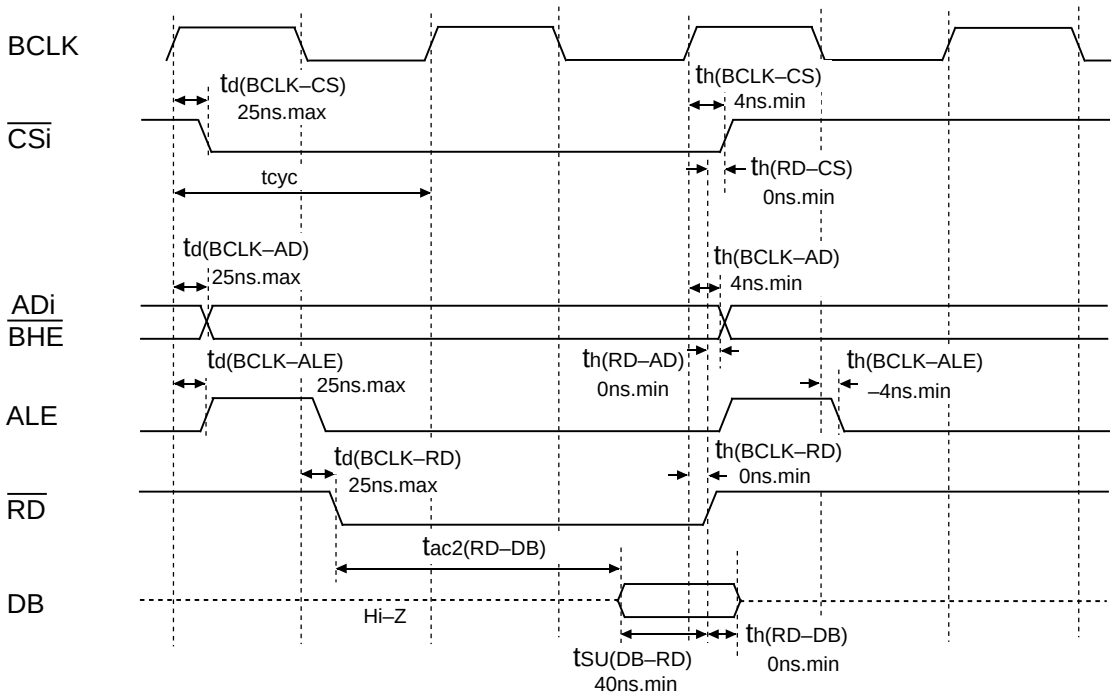
Write timing



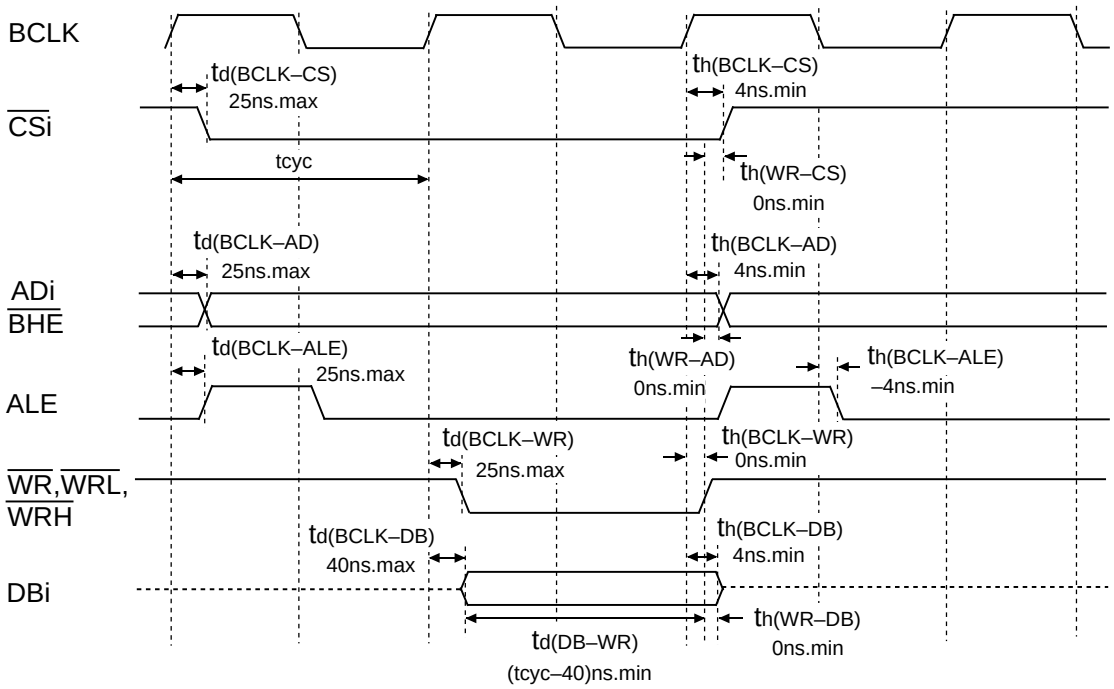
Memory Expansion Mode and Microprocessor Modes
(When accessing external memory area with wait)

VCC = 5V

Read timing



Write timing

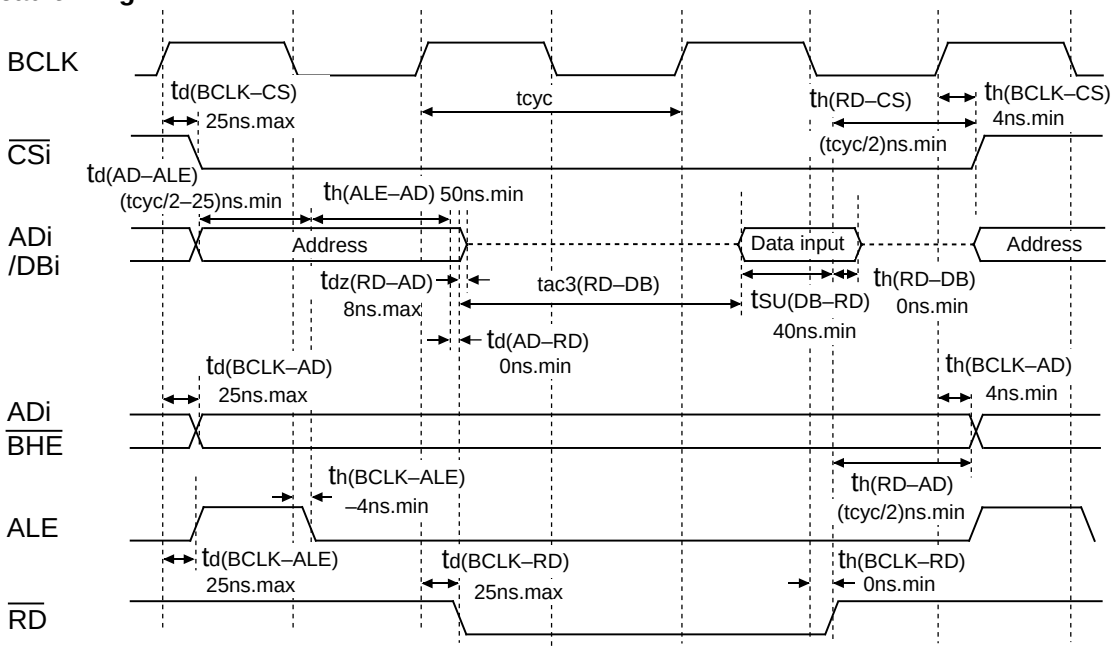


Measuring conditions :
- VCC = 5V
- Input timing voltage : Determined with VIL = 0.8V, VIH = 2.5V
- Output timing voltage : Determined with VOL = 0.8V, VOH = 2.0V

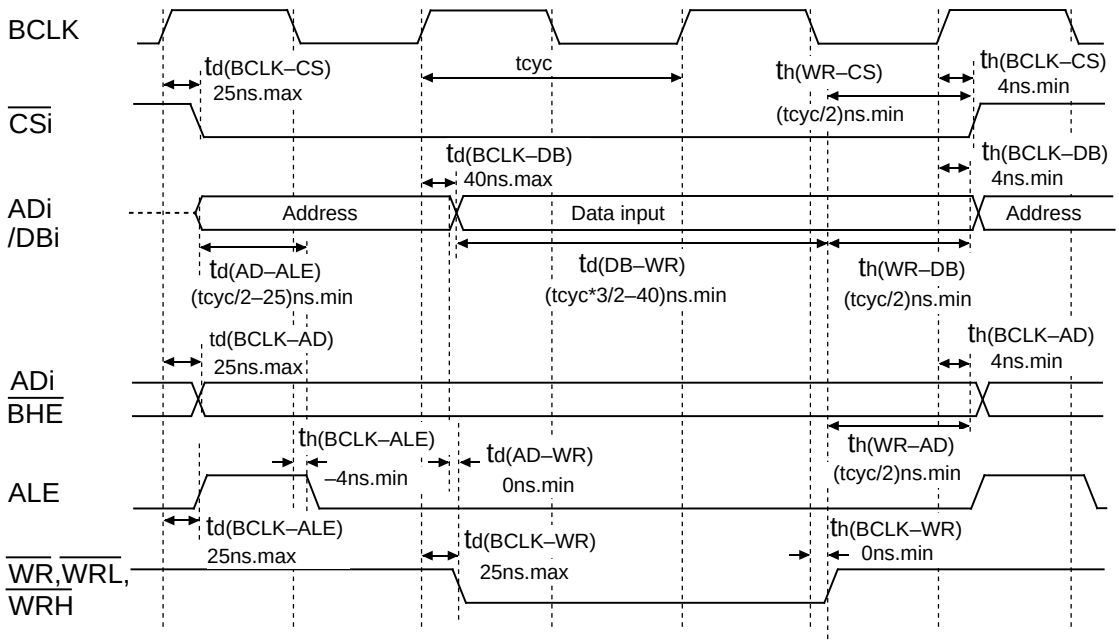
VCC = 5V

Memory Expansion Mode and Microprocessor Mode
(When accessing external memory area with wait, and select multiplexed bus)

Read timing



Write timing



Measuring conditions :
- VCC = 5V
- Input timing voltage : Determined with VIL = 0.8V, VIH = 2.5V
- Output timing voltage : Determined with VOL = 0.8V, VOH = 2.0V

Vcc = 3V

Table 1.66. Electrical characteristics (referenced to Vcc = 3V, Vss = 0V at Ta = 25°C, f(XIN) = 7MHz with wait)

Symbol	Parameter		Measuring condition		Standard			Unit
					Min.	Typ.	Max.	
VOH	HIGH output voltage	P00 to P07,P10 to P17,P20 to P27, P30 to P37,P40 to P47,P50 to P57, P60 to P67,P70 to P77,P80 to P84, P86,P87,P90 to P97,P100 to P107	IOH = –1mA		2.5			V
VOH	HIGH output voltage	XOUT HIGHPOWER	IOH = –0.1mA		2.5			V
		XOUT LOWPOWER	IOH = –50µA		2.5			
VOL	LOW output voltage	P00 to P07,P10 to P17,P20 to P27, P30 to P37,P40 to P47,P50 to P57, P60 to P67,P70 to P77,P80 to P84, P86,P87,P90 to P97,P100 to P107	IOL = 1mA				0.5	V
VOL	LOW output voltage	XOUT HIGHPOWER	IOL = 0.1mA				0.5	V
		XOUT LOWPOWER	IOL = 50µA				0.5	
VT+-VT-	Hysteresis	$\overline{HOLD}$, $\overline{RDY}$, TA0IN to TA4IN, TB0IN to TB2IN, $\overline{INT0}$ to $\overline{INT2}$, $\overline{ADTRG}$, $\overline{CTS0}$, $\overline{CTS1}$, CLK0, CLK1			0.2		0.8	V
VT+-VT-	Hysteresis	$\overline{RESET}$			0.2		1.8	V
VT+-VT-	Hysteresis	XIN			0.2		0.8	V
IIH	HIGH input current	P00 to P07,P10 to P17,P20 to P27, P30 to P37,P40 to P47,P50 to P57, P60 to P67,P70 to P77,P80 to P87, P90 to P97,P100 to P107, XIN, $\overline{RESET}$, CNVss, BYTE	VI = 3V				4.0	µA
IIL	LOW input current	P00 to P07,P10 to P17,P20 to P27, P30 to P37,P40 to P47,P50 to P57, P60 to P67,P70 to P77,P80 to P87, P90 to P97,P100 to P107, XIN, $\overline{RESET}$, CNVss, BYTE	VI = 0V				–4.0	µA
VRAM	RAM retention voltage		When clock is stopped		2.0			V
Icc	Power supply current		When reset in single-chip mode, the output-only pins are open and other pins are VSS	f(XIN) = 7MHz Square wave, no division		6.0	15.0	mA
				f(XIN) = 7MHz Square wave, division by 8		1.6		mA
				f(XCIN) = 32kHz When a WAIT instruction is executed. Oscillation capacity High (Note)		2.8		µA
				f(XCIN) = 32kHz When a WAIT instruction is executed. Oscillation capacity Low (Note)		0.9		µA
				Ta=25°C when clock is stopped			1.0	µA
				Ta=85°C when clock is stopped			20.0	

Note: With one timer operated using fc32.

Vcc = 3V

Table 1.67. A-D conversion characteristics (referenced to Vcc = AVcc = VREF = 3V, Vss = AVss = 0V at Ta = 25°C, f(XIN) = 7MHz unless otherwise specified)

Symbol	Parameter		Measuring condition	Standard			Unit
				Min.	Typ.	Max	
–	Resolution		VREF = VCC			10	Bits
–	Absolute accuracy	Sample & hold function not available (8 bit)	VREF = VCC = 3V, ϕ_{AD} = f(XIN)/2			± 2	LSB
RLADDER	Ladder resistance		VREF = VCC	10		40	kΩ
tCONV	Conversion time(8bit)			14.0			µs
VREF	Reference voltage			2.7		VCC	V
VIA	Analog input voltage			0		VREF	V

Table 1.68. D-A conversion characteristics (referenced to Vcc = 3V, Vss = AVss = 0V, VREF = 3V at Ta = 25°C, f(XIN) = 7MHz unless otherwise specified)

Symbol	Parameter	Measuring condition	Standard			Unit
			Min.	Typ.	Max	
–	Resolution				8	Bits
–	Absolute accuracy				1.0	%
tsu	Setup time				3	µs
RO	Output resistance		4	10	20	kΩ
IVREF	Reference power supply input current	(Note)			1.0	mA

Note: This applies when using one D-A converter, with the D-A register for the unused D-A converter set to "00_{16}".
The A-D converter's ladder resistance is not included.

VCC = 3V

Timing requirements (referenced to Vcc = 3V, Vss = 0V at Ta = 25°C unless otherwise specified)

Table 1.69. External clock input

Symbol	Parameter	Standard		Unit
		Min.	Max.	
t_c	External clock input cycle time	143		ns
$t_{w(H)}$	External clock input HIGH pulse width	60		ns
$t_{w(L)}$	External clock input LOW pulse width	60		ns
t_r	External clock rise time		18	ns
t_f	External clock fall time		18	ns

Table 1.70. Memory expansion and microprocessor modes

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{ac1(RD-DB)}$	Data input access time (no wait)		(Note)	ns
$t_{ac2(RD-DB)}$	Data input access time (with wait)		(Note)	ns
$t_{ac3(RD-DB)}$	Data input access time (when accessing multiplex bus area)		(Note)	ns
$t_{su(DB-RD)}$	Data input setup time	80		ns
$t_{su(RDY-BCLK)}$	$\overline{RDY}$ input setup time	60		ns
$t_{su(HOLD-BCLK)}$	$\overline{HOLD}$ input setup time	80		ns
$t_{h(RD-DB)}$	Data input hold time	0		ns
$t_{h(BCLK-RDY)}$	$\overline{RDY}$ input hold time	0		ns
$t_{h(BCLK-HOLD)}$	$\overline{HOLD}$ input hold time	0		ns
$t_{d(BCLK-HLDA)}$	$\overline{HLDA}$ output delay time		100	ns

Note: Calculated according to the BCLK frequency as follows:

$$t_{ac1(RD-DB)} = \frac{10^9}{f(BCLK) \times 2} - 90 \quad [ns]$$

$$t_{ac2(RD-DB)} = \frac{3 \times 10^9}{f(BCLK) \times 2} - 90 \quad [ns]$$

$$t_{ac3(RD-DB)} = \frac{3 \times 10^9}{f(BCLK) \times 2} - 90 \quad [ns]$$

VCC = 3V

Timing requirements (referenced to VCC = 3V, VSS = 0V at Ta = 25°C unless otherwise specified)

Table 1.71. Timer A input (counter input in event counter mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TA)}$	TAiIN input cycle time	150		ns
$t_{w(TAH)}$	TAiIN input HIGH pulse width	60		ns
$t_{w(TAL)}$	TAiIN input LOW pulse width	60		ns

Table 1.72. Timer A input (gating input in timer mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TA)}$	TAiIN input cycle time	600		ns
$t_{w(TAH)}$	TAiIN input HIGH pulse width	300		ns
$t_{w(TAL)}$	TAiIN input LOW pulse width	300		ns

Table 1.73. Timer A input (external trigger input in one-shot timer mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TA)}$	TAiIN input cycle time	300		ns
$t_{w(TAH)}$	TAiIN input HIGH pulse width	150		ns
$t_{w(TAL)}$	TAiIN input LOW pulse width	150		ns

Table 1.74. Timer A input (external trigger input in pulse width modulation mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{w(TAH)}$	TAiIN input HIGH pulse width	150		ns
$t_{w(TAL)}$	TAiIN input LOW pulse width	150		ns

Table 1.75. Timer A input (up/down input in event counter mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(UP)}$	TAiOUT input cycle time	3000		ns
$t_{w(UPH)}$	TAiOUT input HIGH pulse width	1500		ns
$t_{w(UPL)}$	TAiOUT input LOW pulse width	1500		ns
$t_{su(UP-TIN)}$	TAiOUT input setup time	600		ns
$t_{h(TIN-UP)}$	TAiOUT input hold time	600		ns

Vcc = 3V

Timing requirements (referenced to Vcc = 3V, Vss = 0V at Ta = 25°C unless otherwise specified)

Table 1.76. Timer B input (counter input in event counter mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TB)}$	TBiIN input cycle time (counted on one edge)	150		ns
$t_{w(TBH)}$	TBiIN input HIGH pulse width (counted on one edge)	60		ns
$t_{w(TBL)}$	TBiIN input LOW pulse width (counted on one edge)	60		ns
$t_{c(TB)}$	TBiIN input cycle time (counted on both edges)	300		ns
$t_{w(TBH)}$	TBiIN input HIGH pulse width (counted on both edges)	160		ns
$t_{w(TBL)}$	TBiIN input LOW pulse width (counted on both edges)	160		ns

Table 1.77. Timer B input (pulse period measurement mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TB)}$	TBiIN input cycle time	600		ns
$t_{w(TBH)}$	TBiIN input HIGH pulse width	300		ns
$t_{w(TBL)}$	TBiIN input LOW pulse width	300		ns

Table 1.78. Timer B input (pulse width measurement mode)

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TB)}$	TBiIN input cycle time	600		ns
$t_{w(TBH)}$	TBiIN input HIGH pulse width	300		ns
$t_{w(TBL)}$	TBiIN input LOW pulse width	300		ns

Table 1.79. A-D trigger input

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(AD)}$	$\overline{ADTRG}$ input cycle time (trigger able minimum)	1500		ns
$t_{w(ADL)}$	$\overline{ADTRG}$ input LOW pulse width	200		ns

Table 1.80. 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 1.81. External interrupt $\overline{INTi}$ inputs

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

VCC = 3V

Switching characteristics (referenced to VCC = 3V, VSS = 0V at Ta = 25°C, CM15 = "1" unless otherwise specified)

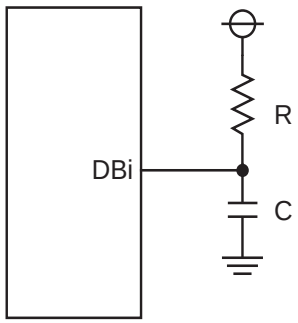
Table 1.82. Memory expansion and microprocessor modes (with no wait)

Symbol	Parameter	Measuring condition	Standard		Unit
			Min.	Max.	
$t_{d(BCLK-AD)}$	Address output delay time	Figure 1.90		60	ns
$t_{h(BCLK-AD)}$	Address output hold time (BCLK standard)		4		ns
$t_{h(RD-AD)}$	Address output hold time (RD standard)		0		ns
$t_{h(WR-AD)}$	Address output hold time (WR standard)		0		ns
$t_{d(BCLK-CS)}$	Chip select output delay time			60	ns
$t_{h(BCLK-CS)}$	Chip select output hold time (BCLK standard)		4		ns
$t_{d(BCLK-ALE)}$	ALE signal output delay time			60	ns
$t_{h(BCLK-ALE)}$	ALE signal output hold time		–4		ns
$t_{d(BCLK-RD)}$	RD signal output delay time			60	ns
$t_{h(BCLK-RD)}$	RD signal output hold time		0		ns
$t_{d(BCLK-WR)}$	WR signal output delay time			60	ns
$t_{h(BCLK-WR)}$	WR signal output hold time		0		ns
$t_{d(BCLK-DB)}$	Data output delay time (BCLK standard)			80	ns
$t_{h(BCLK-DB)}$	Data output hold time (BCLK standard)		4		ns
$t_{d(DB-WR)}$	Data output delay time (WR standard)		(Note 1)		ns
$t_{h(WR-DB)}$	Data output hold time (WR standard) (Note 2)		0		ns

Note 1: Calculated according to the BCLK frequency as follows:

$$td(DB - WR) = \frac{10^9}{f(BCLK) \times 2} - 80 \quad [ns]$$

Note 2: This is standard value shows the timing when the output is off, and doesn't show hold time of data bus.
Hold time of data bus is different by capacitor volume and pull-up (pull-down) resistance value.
Hold time of data bus is expressed in
$t = -CR \times \ln (1 - V_{OL} / V_{CC})$
by a circuit of the right figure.
For example, when $V_{OL} = 0.2V_{CC}$, C = 30pF, R = 1kW, hold time of output "L" level is
$t = -30pF \times 1kW \times \ln (1 - 0.2V_{CC} / V_{CC})$
= 6.7ns



VCC = 3V

Switching characteristics (referenced to VCC = 3V, VSS = 0V at Ta = 25°C, CM15 = "1" unless otherwise specified)

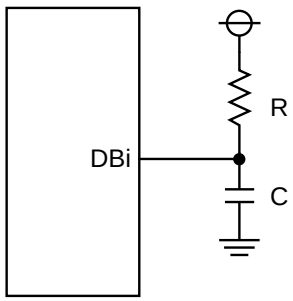
Table 1.83. Memory expansion and microprocessor modes (when accessing external memory area with wait)

Symbol	Parameter	Measuring condition	Standard		Unit
			Min.	Max.	
$t_{d(BCLK-AD)}$	Address output delay time	Figure 1.90		60	ns
$t_{h(BCLK-AD)}$	Address output hold time (BCLK standard)		4		ns
$t_{h(RD-AD)}$	Address output hold time (RD standard)		0		ns
$t_{h(WR-AD)}$	Address output hold time (WR standard)		0		ns
$t_{d(BCLK-CS)}$	Chip select output delay time			60	ns
$t_{h(BCLK-CS)}$	Chip select output hold time (BCLK standard)		4		ns
$t_{d(BCLK-ALE)}$	ALE signal output delay time			60	ns
$t_{h(BCLK-ALE)}$	ALE signal output hold time		– 4		ns
$t_{d(BCLK-RD)}$	RD signal output delay time			60	ns
$t_{h(BCLK-RD)}$	RD signal output hold time		0		ns
$t_{d(BCLK-WR)}$	WR signal output delay time			60	ns
$t_{h(BCLK-WR)}$	WR signal output hold time		0		ns
$t_{d(BCLK-DB)}$	Data output delay time (BCLK standard)			80	ns
$t_{h(BCLK-DB)}$	Data output hold time (BCLK standard)		4		ns
$t_{d(DB-WR)}$	Data output delay time (WR standard)		(Note 1)		ns
$t_{h(WR-DB)}$	Data output hold time (WR standard) (Note 2)		0		ns

Note 1: Calculated according to the BCLK frequency as follows:

$$td(DB - WR) = \frac{10^9}{f(BCLK) \times 2} - 80 \quad [ns]$$

Note 2: This is standard value shows the timing when the output is off, and doesn't show hold time of data bus.
Hold time of data bus is different by capacitor volume and pull-up (pull-down) resistance value.
Hold time of data bus is expressed in
$t = -CR \times \ln (1 - V_{OL} / V_{CC})$
by a circuit of the right figure.
For example, when $V_{OL} = 0.2V_{CC}$, C = 30pF, R = 1kW, hold time of output "L" level is
$t = -30pF \times 1kW \times \ln (1 - 0.2V_{CC} / V_{CC})$
$= 6.7ns$.



VCC = 3V

Switching characteristics (referenced to VCC = 3V, VSS = 0V at Ta = 25°C, CM15 = "1" unless otherwise specified)

Table 1.84. Memory expansion and microprocessor modes (when accessing external memory area with wait, and select multiplexed bus)

Symbol	Parameter	Measuring condition	Standard		Unit
			Min.	Max.	
$t_{d(BCLK-AD)}$	Address output delay time	Figure 1.90		60	ns
$t_{h(BCLK-AD)}$	Address output hold time (BCLK standard)		4		ns
$t_{h(RD-AD)}$	Address output hold time (RD standard)		(Note)		ns
$t_{h(WR-AD)}$	Address output hold time (WR standard)		(Note)		ns
$t_{d(BCLK-CS)}$	Chip select output delay time			60	ns
$t_{h(BCLK-CS)}$	Chip select output hold time (BCLK standard)		4		ns
$t_{h(RD-CS)}$	Chip select output hold time (RD standard)		(Note)		ns
$t_{h(WR-CS)}$	Chip select output hold time (WR standard)		(Note)		ns
$t_{d(BCLK-RD)}$	RD signal output delay time			60	ns
$t_{h(BCLK-RD)}$	RD signal output hold time		0		ns
$t_{d(BCLK-WR)}$	WR signal output delay time			60	ns
$t_{h(BCLK-WR)}$	WR signal output hold time		0		ns
$t_{d(BCLK-DB)}$	Data output delay time (BCLK standard)			80	ns
$t_{h(BCLK-DB)}$	Data output hold time (BCLK standard)		4		ns
$t_{d(DB-WR)}$	Data output delay time (WR standard)		(Note)		ns
$t_{h(WR-DB)}$	Data output hold time (WR standard)		(Note)		ns
$t_{d(BCLK-ALE)}$	ALE signal output delay time (BCLK standard)			60	ns
$t_{h(BCLK-ALE)}$	ALE signal output hold time (BCLK standard)		– 4		ns
$t_{d(AD-ALE)}$	ALE signal output delay time (Address standard)		(Note)		ns
$t_{h(ALE-AD)}$	ALE signal output hold time(Address standard)		50		ns
$t_{d(AD-RD)}$	Post-address RD signal output delay time		0		ns
$t_{d(AD-WR)}$	Post-address WR signal output delay time		0		ns
$t_{dZ(RD-AD)}$	Address output floating start time			8	ns

Note: Calculated according to the BCLK frequency as follows:

$$th(RD - AD) = \frac{10^9}{f(BCLK) \times 2} \quad [ns]$$

$$th(WR - AD) = \frac{10^9}{f(BCLK) \times 2} \quad [ns]$$

$$th(RD - CS) = \frac{10^9}{f(BCLK) \times 2} \quad [ns]$$

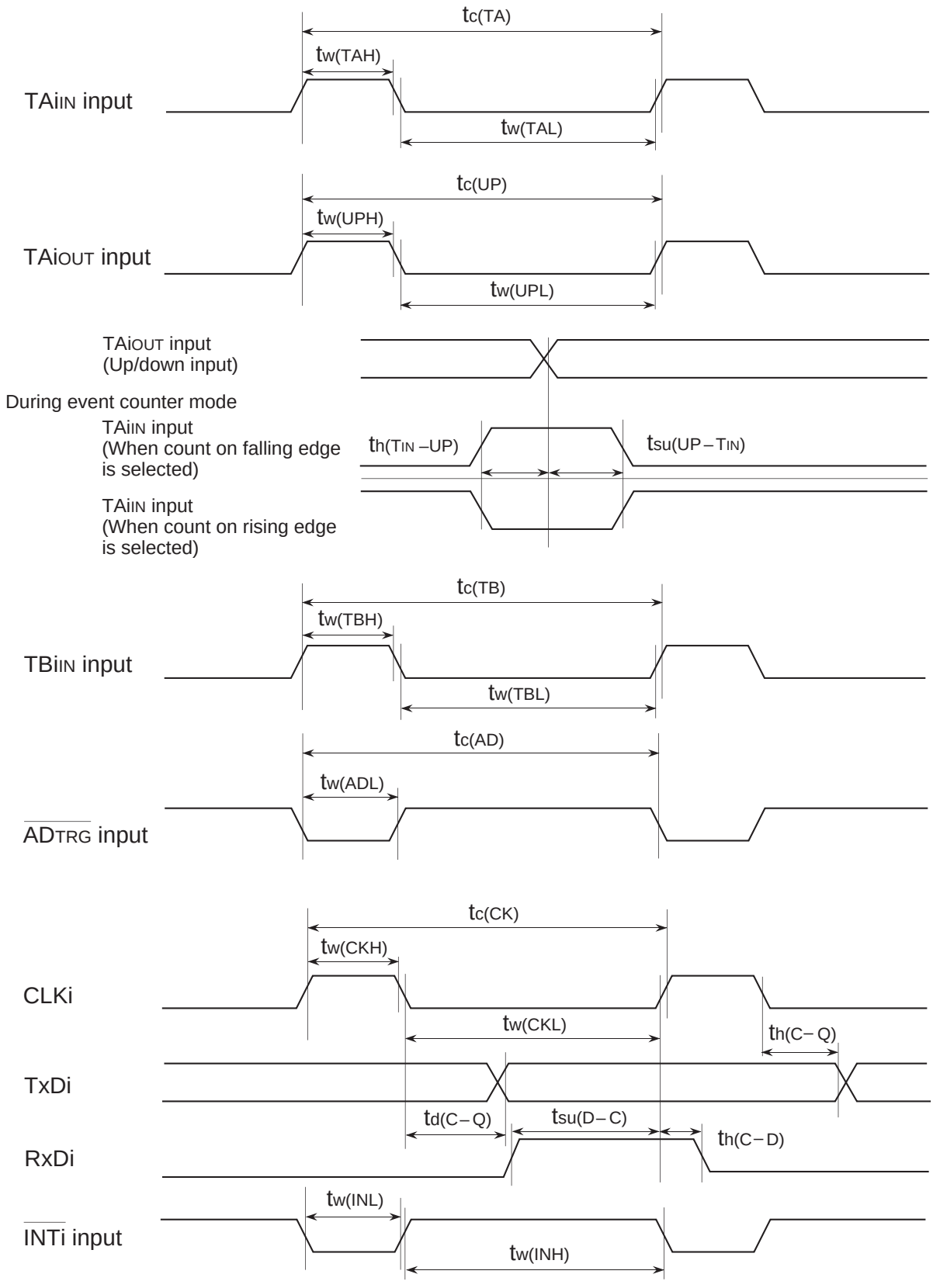
$$th(WR - CS) = \frac{10^9}{f(BCLK) \times 2} \quad [ns]$$

$$td(DB - WR) = \frac{10^9 \times 3}{f(BCLK) \times 2} - 80 \quad [ns]$$

$$th(WR - DB) = \frac{10^9}{f(BCLK) \times 2} \quad [ns]$$

$$td(AD - ALE) = \frac{10^9}{f(BCLK) \times 2} - 60 \quad [ns]$$

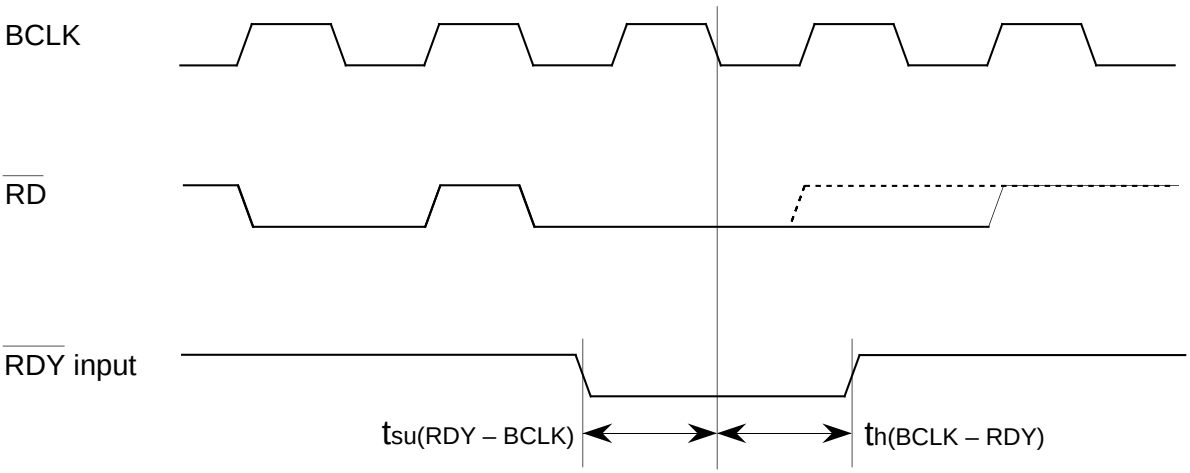
Vcc = 3V


tc(TA)
tw(TAH)
TAiIN input
tw(TAL)
tc(UP)
tw(UPH)
TAiOUT input
tw(UPL)
TAiOUT input
(Up/down input)
During event counter mode
TAiIN input
(When count on falling edge
is selected)
th(TIN –UP)
tsu(UP – TIN)
TAiIN input
(When count on rising edge
is selected)
tc(TB)
tw(TBH)
TBiIN input
tw(TBL)
tc(AD)
tw(ADL)
ADTRG input
tc(CK)
tw(CKH)
CLKi
tw(CKL)
th(C – Q)
TxDi
td(C – Q)
tsu(D – C)
th(C – D)
RxDi
tw(INL)
INTi input
tw(INH)

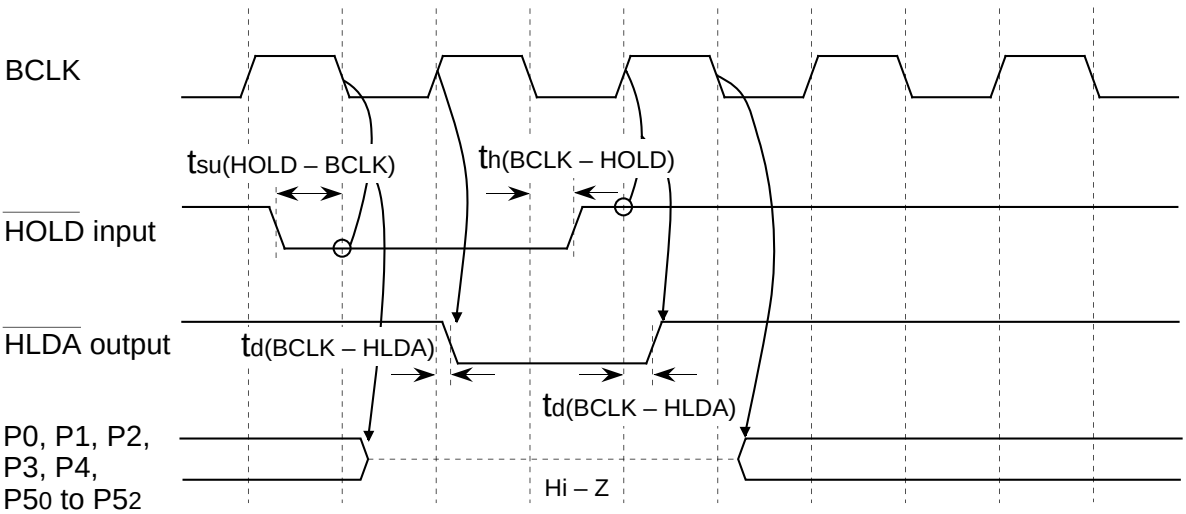
V_{CC} = 3V

Memory Expansion Mode and Microprocessor Mode

(Valid only with wait)



(Valid with or without wait)



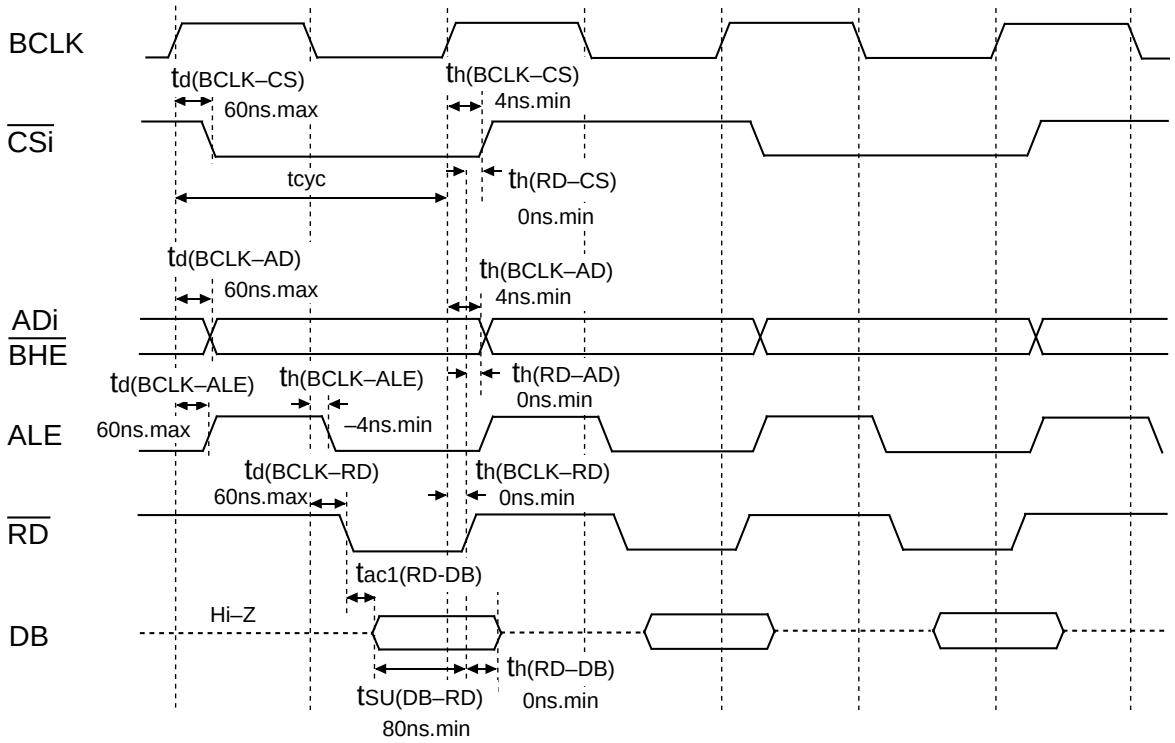
Note: The above pins are set to high-impedance regardless of the input level of the BYTE pin and bit (PM06) of processor mode register 0 selects the function of ports P40 to P43.

Measuring conditions :
- V_{CC} = 3V
- Input timing voltage : Determined with V_{IL} = 0.6V, V_{IH} = 2.4V
- Output timing voltage : Determined with V_{OL} = 1.5V, V_{OH} = 1.5V

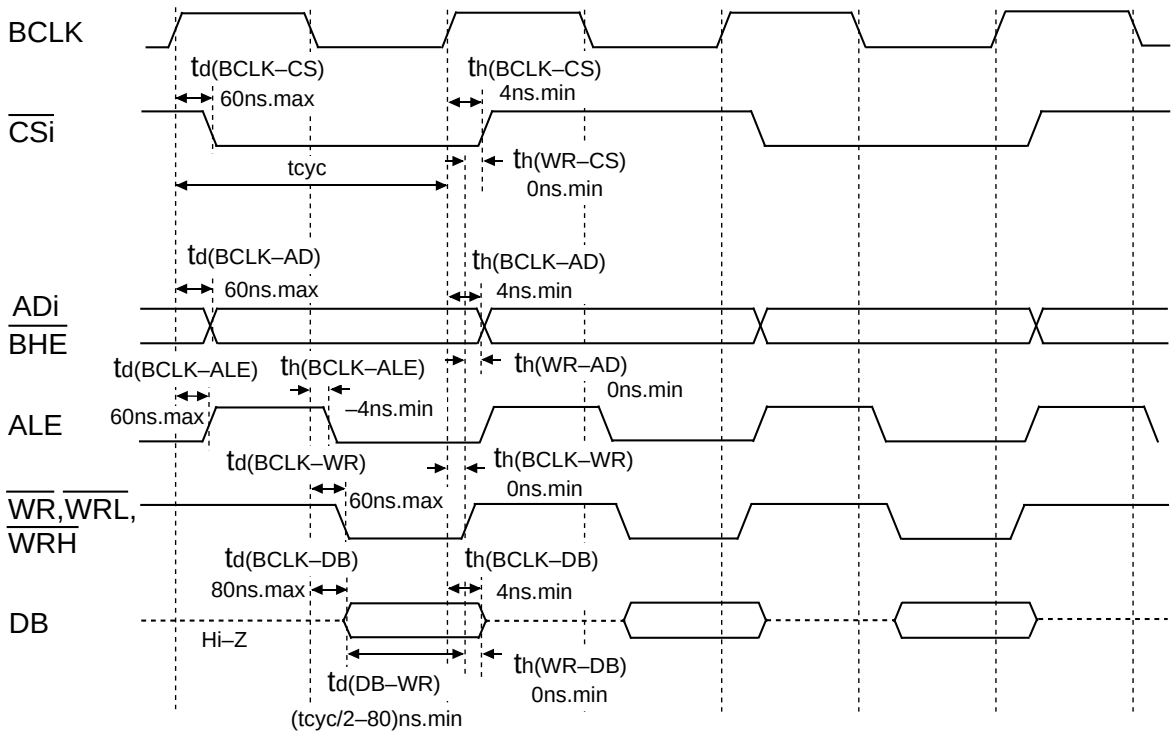
Memory Expansion Mode and Microprocessor Mode
(With no wait)

VCC = 3V

Read timing



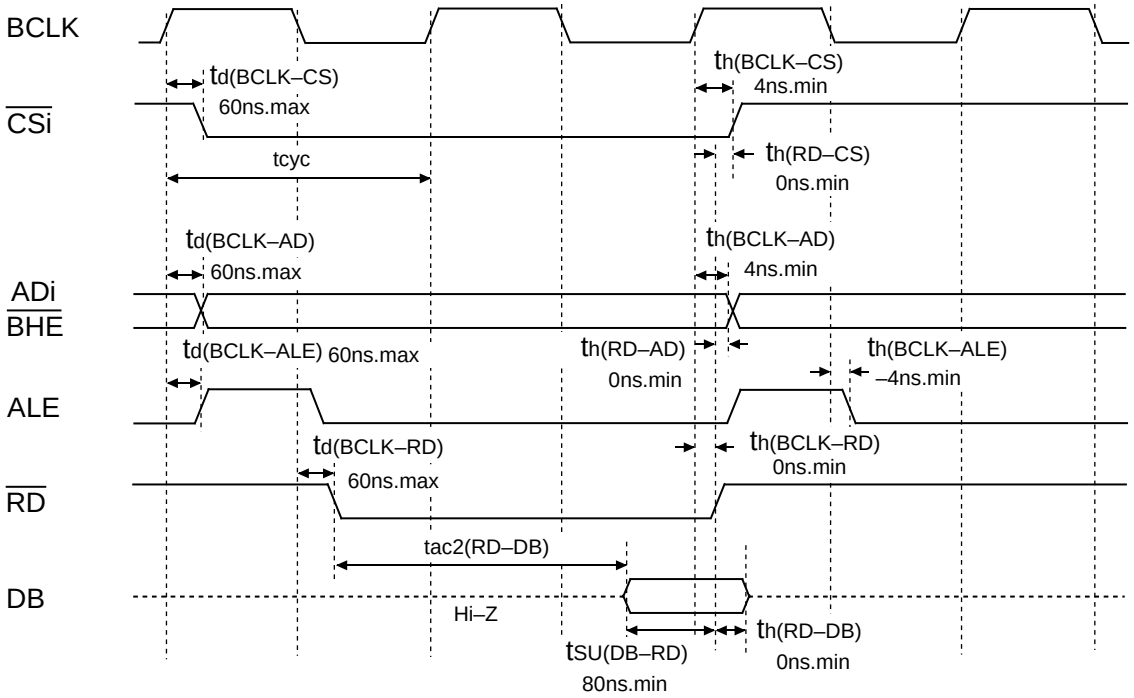
Write timing



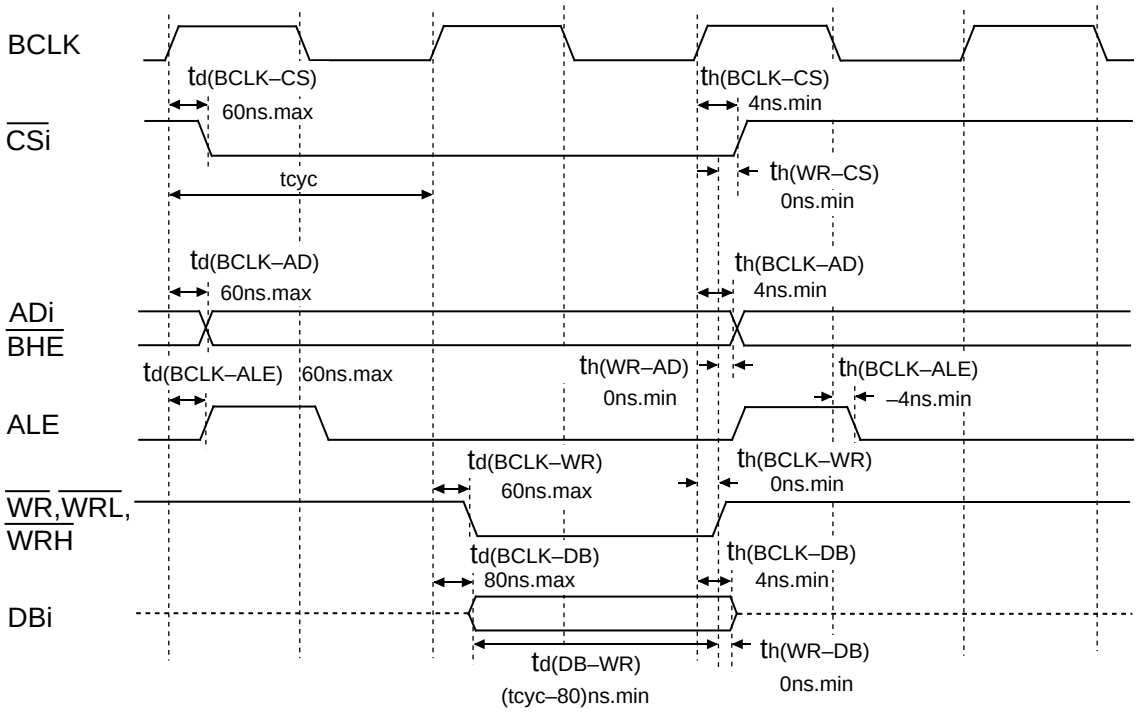
Memory Expansion Mode and Microprocessor Mode
(When accessing external memory area with wait)

V_{CC} = 3V

Read timing



Write timing

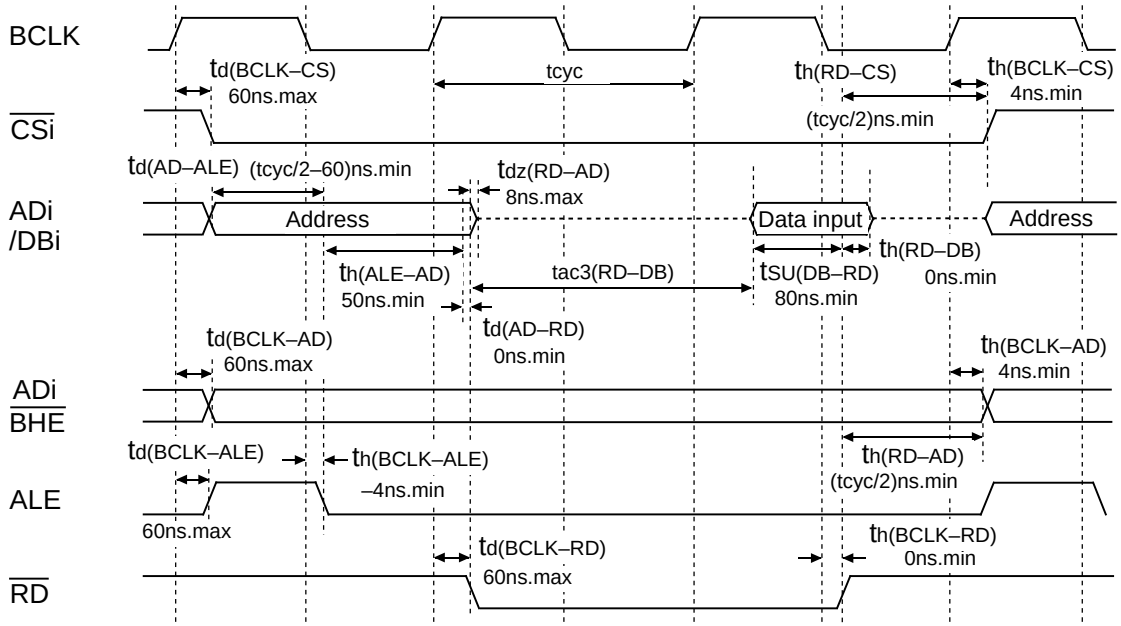


Measuring conditions :
- V_{CC} = 3V
- Input timing voltage : Determined with V_{OL} = 0.48V, V_{OH} = 1.5V
- Output timing voltage : Determined with V_{IL} = 1.5V, V_{IH} = 1.5V

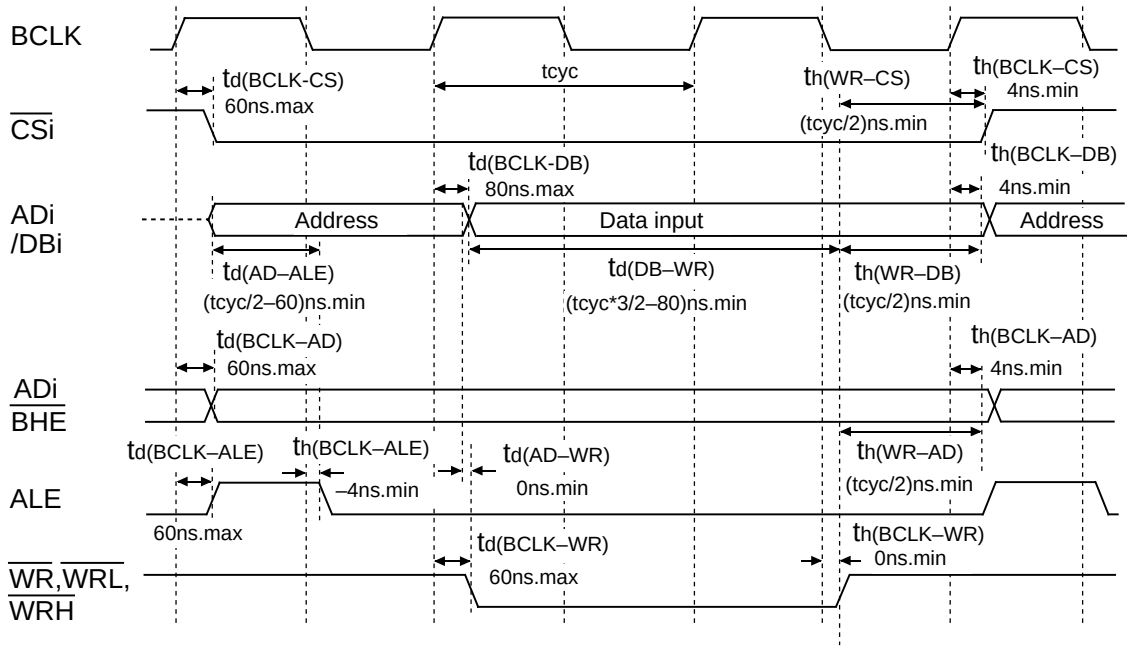
VCC = 3V

Memory Expansion Mode and Microprocessor Mode
(When accessing external memory area with wait, and select multiplexed bus)

Read timing



Write timing



Measuring conditions :
- VCC = 3V
- Input timing voltage : Determined with VOL = 0.48V, VOH = 1.5V
- Output timing voltage : Determined with VIL = 1.5V, VIH = 1.5V

GZZ─ SH00─ 39B <67A1>

Mask ROM number	

MITSUBISHI ELECTRIC SINGLE-CHIP 16-BIT MICROCOMPUTER M30600M8-XXXFP/GP
MASK ROM CONFIRMATION FORM

Receipt	Date :	
	Section head signature	Supervisor signature

Note : Please complete all items marked ※.

※	Customer	Company name	TEL ( )	Issuance signature	Submitted by	Supervisor
		Date issued	Date :			

※1. Check sheet

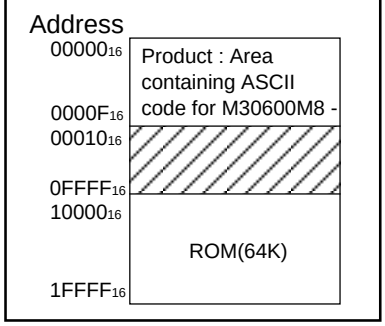
Please specify the name of the product being ordered and the EPROM being supplied.
We require 3 sets of EPROMs per pattern (please mark the box).
Mitsubishi will create the mask using the data on the EPROMs supplied, providing the data is the same on at least two of those sets. Mitsubishi will, therefore, only accept liability if there is any discrepancy between the data on the EPROM sets and the ROM data written to the product.
Please carefully check the data on the EPROMs being submitted to Mitsubishi.

Microcomputer type No. : ☐ M30600M8-XXXFP ☐ M30600M8-XXXGP

Checksum code for total EPROM area : | | | | | (hex)

EPROM type :

☐ **27C101**



(1) Write "FF16" to the lined area.
(2) The area from 0000016 to 0000F16 is for storing data on the product type name.
The ASCII code for 'M30600M8-' is shown at right.
The data in this table must be written to address 0000016 to 0000F16.
Both address and data are shown in hex.

Address		Address	
00000₁₆	' M ' = 4D16	00008₁₆	' ─ ' 2D16
00001₁₆	' 3 ' = 3316	00009₁₆	FF16
00002₁₆	' 0 ' = 3016	0000A₁₆	FF16
00003₁₆	' 6 ' = 3616	0000B₁₆	FF16
00004₁₆	' 0 ' = 3016	0000C₁₆	FF16
00005₁₆	' 0 ' = 3016	0000D₁₆	FF16
00006₁₆	' M ' = 4D16	0000E₁₆	FF16
00007₁₆	' 8 ' = 3816	0000F₁₆	FF16

GZZ─ SH00─ 39B <67A1>

Mask ROM number	

MITSUBISHI ELECTRIC SINGLE-CHIP 16-BIT MICROCOMPUTER M30600M8-XXXFP/GP
MASK ROM CONFIRMATION FORM

The ASCII code for the type No. can be written to EPROM addresses 0000_{16} to $0000F_{16}$ by specifying the pseudo-instructions for the respective EPROM type shown in the following table at the beginning of the assembler source program.

EPROM type	27C101
Code entered in source program	△* = △ $00000 △ .BYTE△' M30600M8-'

Note: The ROM cannot be processed if the type No. written to the EPROM does not match the type No. in the check sheet.

※ 2. Mark specification

The mark specification differs according to the type of package. After entering the mark specification on the separate mark specification sheet (for each package), attach that sheet to this masking check sheet for submission to Mitsubishi.
For the M30600M8-XXXFP, submit the 100P6S mark specification sheet. For the M30600M8-XXXGP, submit the 100P6Q mark specification sheet.

※ 3. Usage Conditions

For our reference when of testing our products, please reply to the following questions about the usage of the products you ordered.

(1) Which kind of XIN-XOUT oscillation circuit is used?
☐ Ceramic resonator ☐ Quartz-crystal oscillator
☐ External clock input ☐ Other ( )
What frequency do you use?
f(XIN) = ☐ MHz

(2) Which kind of XCIN-XCOUT oscillation circuit is used?
☐ Ceramic resonator ☐ Quartz-crystal oscillator
☐ External clock input ☐ Other ( )
What frequency do you use?
f(XCIN) = ☐ kHz

(3) Which operation mode do you use?
☐ Single-chip mode ☐ Memory expansion mode
☐ Microprocessor mode

※ 4. Special item (Indicate none if there is no specified item)

GZZ-SH00-40B <67A1>

ROM number	

MITSUBISHI ELECTRIC SINGLE-CHIP 16-BIT MICROCOMPUTER M30600E8-XXXFP/GP
ROM WRITING ORDER FORM

Receipt	Date :	
	Section head signature	Supervisor signature

Note : Please complete all items marked※

※	Customer	Company name	TEL ( )	Issuance signature	Submitted by	Supervisor
		Date issued	Date :			

※ 1. Check sheet

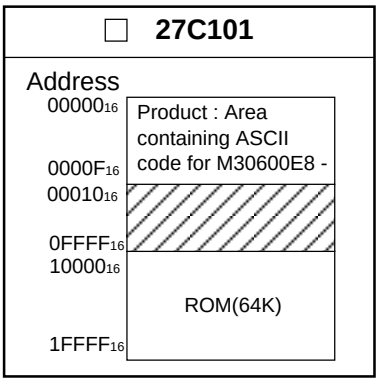
Please specify the name of the product being ordered and the EPROM being supplied.
We require 3 sets of EPROMs per pattern (please mark the box).
Mitsubishi will write to ROM using the data on the EPROMs supplied, providing the data is the same on at least two of those sets. Mitsubishi will, therefore, only accept liability if there is any discrepancy between the data on the EPROM sets and the ROM data written to the product.
Please carefully check the data on the EPROMs being submitted to Mitsubishi.

Microcomputer type No. : ☐ M30600E8-XXXFP ☐ M30600E8-XXXGP

Checksum code for total EPROM area : | | | | | (hex)

EPROM type :

☐ **27C101**



(1) Write "FF16" to the lined area.
(2) The area from 00000₁₆ to 0000F16 is for storing data on the product type name.
The ASCII code for 'M30600E8-' is shown at right.
The data in this table must be written to address 00000₁₆ to 0000F16.
Both address and data are shown in hex.

Address			Address	
00000₁₆	' M '	= 4D16	00008₁₆	'—' 2D16
00001₁₆	' 3 '	= 3316	00009₁₆	FF16
00002₁₆	' 0 '	= 3016	0000A16	FF16
00003₁₆	' 6 '	= 3616	0000B16	FF16
00004₁₆	' 0 '	= 3016	0000C16	FF16
00005₁₆	' 0 '	= 3016	0000D16	FF16
00006₁₆	' E '	= 4516	0000E16	FF16
00007₁₆	' 8 '	= 3816	0000F16	FF16

GZZ-SH00-40B <67A1>

ROM number	

MITSUBISHI ELECTRIC SINGLE-CHIP 16-BIT MICROCOMPUTER M30600E8-XXXFP/GP
ROM WRITING ORDER FORM

The ASCII code for the type No. can be written to EPROM addresses 00000_{16} to $0000F_{16}$ by specifying the pseudo-instructions for the respective EPROM type shown in the following table at the beginning of the assembler source program.

EPROM type	27C101
Code entered in source program	△* =△ $00000 △.BYTE△ 'M30600E8-'

Note: The ROM cannot be processed if the type No. written to the EPROM does not match the type No. in the check sheet.

※ 2. Mark specification

The mark specification differs according to the type of package. After entering the mark specification on the separate mark specification sheet (for each package), attach that sheet to this masking check sheet for submission to Mitsubishi.
For the M30600E8-XXXFP, submit the 100P6S mark specification sheet. For the M30600E8-XXXGP, submit the 100P6Q mark specification sheet.

※ 3. Usage Conditions

For our reference when of testing our products, please reply to the following questions about the usage of the products you ordered.

(1) Which kind of X_{IN}-X_{OUT} oscillation circuit is used?
☐ Ceramic resonator ☐ Quartz-crystal oscillator
☐ External clock input ☐ Other ( )
What frequency do you use?
$f(X_{IN})$ = [ ] MHz

(2) Which kind of X_{CIN}-X_{COUT} oscillation circuit is used?
☐ Ceramic resonator ☐ Quartz-crystal oscillator
☐ External clock input ☐ Other ( )
What frequency do you us?
$f(X_{CIN})$ = [ ] kHz

(3) Which operation mode do you use?
☐ Single-chip mode ☐ Memory expansion mode
☐ Microprocessor mode

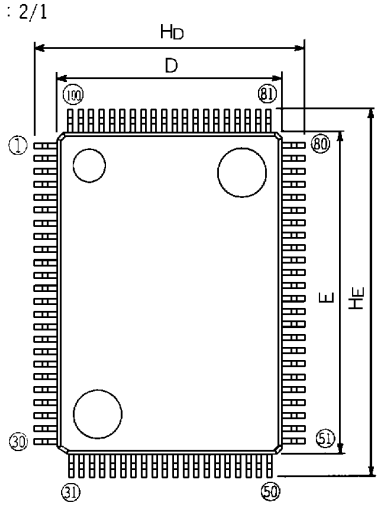
※ 4. Special item (Indicate none if there is no specified item)

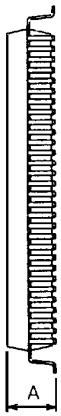
100P6S−A

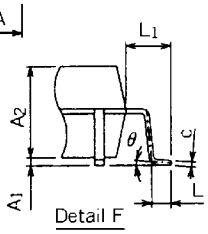
Plastic 100pin 14×20mm body QFP

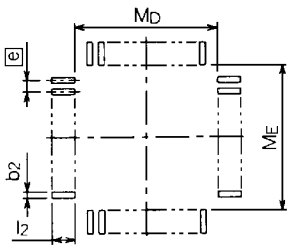
EIAJ Package Code	JEDEC Code	Weight(g)	Lead Material
QFP100-P-1420-0.65	−	1.58	Alloy 42

Scale : 2/1









Recommended Mount Pad

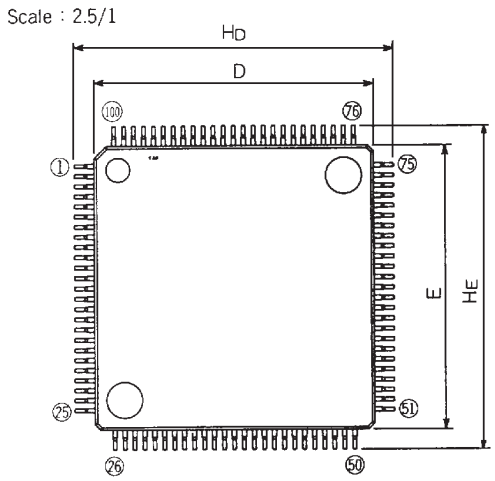
Symbol	Dimension in Millimeters		
	Min	Nom	Max
A	−	−	3.05
A_1	0	0.1	0.2
A_2	−	2.8	−
b	0.25	0.3	0.4
c	0.13	0.15	0.2
D	13.8	14.0	14.2
E	19.8	20.0	20.2
e	−	0.65	−
H_D	16.5	16.8	17.1
H_E	22.5	22.8	23.1
L	0.4	0.6	0.8
L_1	−	1.4	−
y	−	−	0.1
θ	0°	−	10°
b_2	−	0.35	−
l_2	1.3	−	−
M_D	−	14.6	−
M_E	−	20.6	−

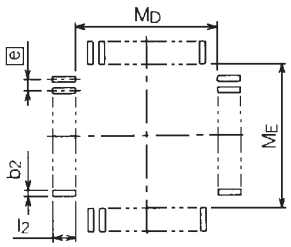
100P6D−A

Plastic 100pin 14×14mm body LQFP

EIAJ Package Code	JEDEC Code	Weight(g)	Lead Material
LQFP100-P-1414-0.50	—	0.62	Alloy 42

Scale : 2.5/1





Recommended Mount Pad

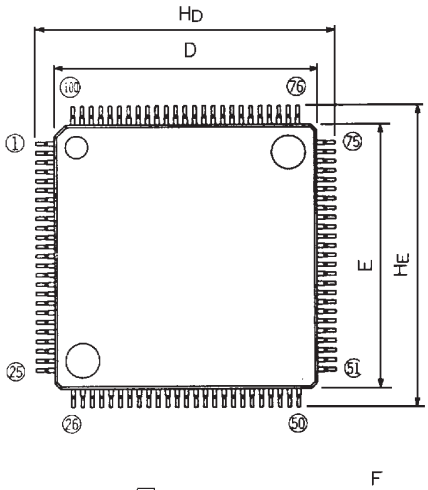
Symbol	Dimension in Millimeters		
	Min	Nom	Max
A	—	—	1.7
A_1	0	0.1	0.2
A_2	—	1.4	—
b	0.13	0.18	0.28
c	0.105	0.125	0.175
D	13.9	14.0	14.1
E	13.9	14.0	14.1
e	—	0.5	—
H_D	15.8	16.0	16.2
H_E	15.8	16.0	16.2
L	0.3	0.5	0.7
L_1	—	1.0	—
y	—	—	0.1
θ	0°	—	10°
b_2	—	0.225	—
l_2	1.0	—	—
M_D	—	14.4	—
M_E	—	14.4	—

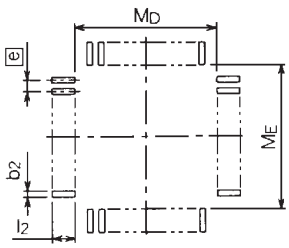
100P6Q−A

Plastic 100pin 14×14mm body LQFP

EIAJ Package Code	JEDEC Code	Weight(g)	Lead Material
LQFP100-P-1414-0.50	—		Cu Alloy

Under Development





Symbol	Dimension in Millimeters		
	Min	Nom	Max
A	—	—	1.7
A_1	0	0.1	0.2
A_2	—	1.4	—
b	0.13	0.18	0.28
c	0.105	0.125	0.175
D	13.9	14.0	14.1
E	13.9	14.0	14.1
e	—	0.5	—
H_D	15.8	16.0	16.2
H_E	15.8	16.0	16.2
L	0.3	0.5	0.7
L_1	—	1.0	—
y	—	—	0.1
θ	0°	—	10°
b_2	—	0.225	—
l_2	1.0	—	—
M_D	—	14.4	—
M_E	—	14.4	—